Senior Design: Project Alfred
The Building Master

Group 6:

Alec Seketa

Brandon Becker

Ben Bush

Table of Contents

1.0 Executive Summary

Project Alfred is a team. A hardware and software team adopting old technology, new innovation, and organization to aid business in day to day industry and activities. As the name states it is designed be the business butler, or a smart building, to archive, retrieve, and aid company information. In order to achieve our goal, this prototype is constructed from many complex different elements, products, and programs created by our team or another company. To this end, this summary provided a brief overview of what the product along with, contents and topics to be discussed in creating the building master.

To begin with, the project will be split into two separate entities the boxes at the door and the server computer that the information will be backed up on. Initial, the boxes will be placed outside each major location or door in the building. As for which location to use, that is for the company to decide. Some examples may include, major storage rooms, hospital critical patient rooms, electrical rooms, and conference rooms or even all the rooms in the building that the particular company would want to apply. Given that, the company has laid out each of its most important locations in the building the system will monitor the enterprise taking place in those rooms. Under those circumstances, a main frame on a computer will monitor each boxes location and aid the other boxes in coordinating the company's material. Anything from meeting times, to inventory, shipment dates, and patient records will be available to authorized personal at anytime and anywhere in the building. With this intention, Alfred will revolutionize company's ability gathering, storage and retrieval to efficiently organize a buildings data for the digital age.

In order to create Alfred, the group has provided this document with the instructions to design, build, and test the prototype product. In addition, this document provides schematics and materials of products used in its creation as well as other product designs referenced when constructing this prototype. As a result, we have provided a breakdown each individual sub systems as well as, layouts of their individual composition. Some subsystems include, power systems, networking devices and software, embedded controls of the major operating systems as well as accessories used by the device. While, some of these sub systems have been purchased by other companies, the prototype requires premade products combined together to create our main appliance. Therefore, each subsystem used has been carefully researched and designed to show purpose and application in Alfred. Subsequently, proving the components need in creating the final product. With regards to the subsystems chosen, this report also provides a blue print and instructions on how each subsystem works together and fits into the final product. As mentioned, a full overview of the components programs and products applied when creating the prototype.

However, the company should remember this a tool, not an automated system and will require maintenance to keep Alfred working the way it was intended to help the business market. In short, we have provided a user friendly base for record keeping and innovative building management with the proper care can benefit enterprises with a numerous amounts of applications.

2.0 Project Description

In the next sections the paper will discussing the founding of project Alfred. Provided is the goals of why companies will use Alfred in day to day activities, as well as, the different objectives need to accomplish a working prototype. The section is wrapped up with the different types of components that will be used in the construction of the prototype.

2.1 Project Motivation and Goals

In the beginning, while working as a project manager at Disney World Company I began most of my days with showing contractors where the equipment and rooms that would be used during construction on new projections or refurbishing facilities. Eventually, after doing this many weeks I found one thing in common with all of the projects: most of us had no idea where anything was or where to start looking. That being the case, we would spend hours trying different doors guessing which key would work for what door. Consequently, I would have to contact the maintenance manager of every ride a day ahead of schedule to locate things for me, taking time out of his busy day, finding one resource after another. However, some days I couldn't get ahold of the maintenance manager cause he was too busy, sick, or out on vacation. Thus, making me fend for myself using outdate drawings and information. Above all, I asked why this information is not available and if it is why is it not up to date. As a result, I as every engineer in the history of mankind said, "there has to be a better way to do this!" Then, the answer came to me while watching Batman one night. In other words, I saw a particular butler telling everyone with procession direct locations of each room that the guest was to meet with Master Bruce Wayen. "That's it! Why not put a butler in every building?" I questioned. Of course, putting a man in a building creates no electrical challenges plus hiring a butler seemed to old school. Nevertheless, I said what if I could put an electronic butler in every building. Accordingly, the building master was born.

The team wants to create a better system for buildings in order to store, organize, and location of information of buildings and the rooms that lie within them. In particular, this system, Alfred, would know the ins and outs of the building as if it were the companies own personalized butler. With this in mind, the operation would start with creating a small device at every door with a touch screen that would manage and store the entire room's data. Since, there are so many rooms with so many applications each box will be set up with a touch screen user interface to be tailored to each room's significant design. Because of this, authorized users can have easy access to the information, as well as be able to setup and control which aspects the room needs whether it be identifying electrical equipment in the electrical room, to a disease and treatments in a hospital patients room. Once all the boxes are set to the desired purpose they would all tie into one system or the main server at the front desk that will know the location of every room in the building as well as its directions and small sample access to what is inside the room. Our goal: to help the people navigate, manage, and be informed about their buildings to increase efficiency, information, and organization in the work place.

To the future: we believe this as a base system and the sky is the limit, and just like the IPhone is the starting point to many more application. We believe that this system can be improved, added to, or customized for any personalized use. For example, we can add a lock system to each door controlled by the box to have a pass key to open the door, or a temperature gage to monitor the room climate or check for fires, and automated ordering system which notifies the main server when you're running low on products in storage. With this intention, this prototype will be the start in developing a new base design for the first smart building.

2.2 Project Objectives

Initially, the object of the project is to create a working prototype that will simulate the entire system that would be used in creating the building project. The project is divided into two parts of the product, the box or appliance that will be near the door of the room with a user interface and the mainframe which will be used to monitor all boxes in the building. Additionally, the main frame and appliance boxes are divided into both hardware and software components which will be necessary in creating the system as a whole. With this in mind, we have many objectives to accomplish to complete the design and have many different subsystems working together for each piece.

2.21Appliance Objectives

The Appliance also known as the prototype box, is the box that will be used to represent the all the box that will be at the doors in a company. Normally a company would own more than one box and each one would be stationed at a door where the company would like to keep records. But considering cost and design, the prototype is one box that is designed to be copied and distributed throughout the company to connect to a company's main frame.

2.211 Appliance Hardware Objectives

The appliance box will be stationed at every door that the administrator suggests would be necessary for appliance. The list below is the objectives that that will be achieved for the hardware part of the appliance box:

- The appliance will be small portable out of the way in a safe location where it can be access regularly without taking up to much wall space.

- The appliance has enough storage to hold the information of the entire contents and location of the room as well as, any data that the user would like to store for extra information. IE: meeting times, patient records, inventory
- The appliance will have a user friendly input/output device that is clear and able to see for most people even those with disability with aids (such as glasses) can read and operate efficiently.
- The appliance will have a device (such as camera) that will allow for easy recording of data going and out of the room.
- The appliance will have low power device such that it can fit in the product and can be tapped in on site to any 120 volt 60 Hz AC outlet used standard in the United States.
- The appliance will have a network adaptation to allow to be taped into by any wireless device used by the company and subsequently any others who have access to the devices. Furthermore, this adaptation will be largely added to allow communication between Alfred main server and device.
- The appliance devices will have a RFID reader to allow for RFID tracking and input to be used when taking inventory or even a key to access information.

2.212 Appliance Software Objectives

Project Alfred appliance boxes will come with numerous software and is designed to work with the mainframe to coordinate, organize, and record all the data going in and out of the room. The following list is the objectives for the software of project Alfred:

- The appliance software will be easy to use read and understand. The program will be constructed to work with a touch screen interface that is easy for the user to read and navigate.
- The appliance software will have a catalog program. This program will keep track of inventory of the room (items in the room and quantity), in addition the device will be able to keep track of other operation happening in the rooms. For example, the program will record meeting times, patient records and dates of delivery, withdraw and maintenance of anything with in the room.
- The appliance software will have a search function, allowing the users of the device to easily locate what they are looking for in the room and search the catalog efficiently.
- The appliance will work with the mainframe software to keep track of real time. Allowing for accurate record keeping of activities taken place in the room.
- The software will include the latest applications of Barcode and QR code software. This will work with both the camera and catalog coding to keep

accurate records of items delivered or removed from the room. The User will be able to program in their own codes or have it done already for them for custom barcode record keeping.

- The appliance software will allow for a power saving function that will turn off or put the device to sleep when not in use.
- The software will work with the mainframe as a whole to retrieve and send data to the major program in charge.
- The mainframe user will be able to access any box at will with in the network it is hooked too. It will be able to update the appliance whenever needed.
- The appliance software will notify when the room is low on inventory or a change has occurred in the room in order for the user to take appropriate action

2.22 Mainframe objectives

The mainframe is the hardware and software the prototype boxes will be controlled by as well as managed. The mainframe can be one computer or a huge server, designed to be a backup for all the boxes storage and a brain of the full design. The List below shows the objectives needed for hardware and software of this device provided by companies. As for the prototype we will busing one of our computers

2.221 Mainframe hardware objectives

The mainframe will not be a provided device. The mainframe is to be any computer the company wants to designate as the master. Alfred's mainframe computer can be a back room computer, the front desk computer, or the manager's computer. Alfred's packaging will included a RFID to the server for the administrator to use as a passcode system for the device. Due to this factor the computer will need to fit the qualifications below to be used for project Alfred Design.

- The main frame hard ware will a computer or server computer that will be located in an accessible area for the administrator of Alfred to be able to update, work on, and use as they see fit.
- The main frames storage will be large enough to be able to house all of the appliances information onto the server.
- This server is also to be used as a backup for when one box goes down to store the data of the appliance that needs repair.
- The hardware of the main frame will have a large networking device to allow for communication from itself to other devices. It will act as the master hardware for appliance slave devices of the Alfred Network system.

2.222 Mainframe software objectives

Mainframe software will be provided to the computer to allow for the installation of Alfred on the main computer server of the companies choosing. The software installation will require the computer to be compatible to Windows 7 and the products software will provide:

- The computer software will be compatible with the new operating systems on the market examples include Microsoft Windows 7 and later.
- The computer networking software will be set up as master of the other boxes and will be able to remote access any box it needs.
- The computer software will be able to transfer memory from the small boxes to its own storage for back up purposes and information retrieval purposes.
- The main frame will also be notified when a change has occurred in the system and will keep records as well as time/dates to keep it well informed.

2.3 Project Requirements and Specifications

This section provides a brief overview of the hard ware and software specifications that are needed to meet all the objects listed in previous section. Each of these specifications have been broken down into major subsystems. More information and individual break down of each subsystem has been provided in later portions of this document.

Mounting equipment and hardware for door box appliance: Mounting equipment will be big enough to house all the equipment but small enough to not be inconvenient to add near the door. We want to add a screen for use so if the person is unable to tap into it with wireless access they can update it on location. Due to safety we prefer to place them near the door not on the door or the swing location to keep people from hitting those who go in and out of the door. We want an easy mounting system that can be installed on any door. Standard materials can be used since this will be for inside of buildings we are not worried about too many weather hazards. However the materials should be shock proof to avoid any accidents.

2.31 Subsystem 1: Embedded Controls & Storage

Processor (appliance box only): We will be using a Dual core ARM processor that will be running between 1200-1400 MHz. An ARM Mali-400 or similar graphics processor will be used to allow for a rich color display for the user input. We expect to use 1024 MB Ram to handle data input, storage and networking between devices more smoothly. Similar to the ones used in Smart phone’s we believe the technology to be used to allow for low power consumption with all the features we need to program. To program the processor we can use. For the operating system, we could use the RISC OS or the android OS platform, Python or C++ will be used to code the control devices

Storage Catalog (hard drive): Each one to store, download, and update information at any time, for this reason we recommend an actually hard drive to store large amounts of data to maximize uses thought out the building. For the main box. We went with a 16 GB SSD for the small box. The storage software will be written using RISC OS or the android OS platform, Python or C++. This will allow for the creation of a search function to look for all of the things cataloged in the building as well as update the information, quantity, or details on the items or things that are in the room. Also we would like it to store the date and keep track of time for the program. So it will store the dates and times things have been cataloged in the room. Furthermore, the main box will keep track of all records so you can access all information on the time. As a result, the storage needs to be reliable and sturdy to keep up with constant reading an writing.

2.32 Subsystem 2: Camera Input & Output

Camera input and output (appliance only): The three current ways of keeping inventory and cataloging in the United States is the two scanning technologies (QR and Barcode) and the RFID technology. In order to utilize all of these technologies efficiently and without taking up a lot of space is to install a camera mounted to the box. For an example, a camera to be use would be something like: iPhone Backup Camera (iOS) works. With application programs found on the market today, the camera will allow for reading of all different barcodes and could be used in cataloging personalized codes in inventory stock rooms.

Barcode (appliance only): The barcode scanning has been around for decades. As the oldest and easiest way to look up catalog, and keep track of information the appliance boxes will be provided with barcode scanning applications. This will work in conjunction with the camera and the cataloging programs to easily scan in and keep track what goes in and out of the room.

QR Code (appliance only): Due to the rise and popularity of the new QR codes, we will also provide the appliance boxes with the latest in QR code technology. This will provide the user with an alternative way of scanning in products as well as, optimize the box to allow for items that do not have a barcode to scan.

RFID reader: To allow for process involving RFID chips. It is the latest and newest technology being used by big companies such as Disney to keep track of and store information. RFID chips can be used in conjunction with cards as password protecting programs, as well as keeps track of what ever has been scanned by it without having to use a barcode. Therefore, allowing for maximum record keeping without having to lift a box.

2.33 Subsystem 3: Networking

Networking: So we can tap into without using the boxes and to help locate the box with another device that has Wi-Fi, be a IPhone or tablet. Also helpful so you can update the door files and mainframe via the door device. Okay so for the small boxes we agreed that they need to be able to send and receive data. Best way to do this will likely be a wireless adapter. We went with some pretty expensive usb adapters. They use the newest 802.11ac standards so they are as fast as they get. We can easily downgrade to less expensive ones. We don't know if the basic software will just run or we will need to find freeware or program them ourselves. Ideally I'd see us hooking everything up to a computer setting up the network and then just running. Since we talked about being able to bounce the signal off other boxes we could use software to turn it into a wireless repeater when it doesn't have data to transmit. We can easily patrician it for the different subsystems. For the main Box we are going to want to set it up as a Router. That way it will be the main network hub. We should be able to do that with a wireless network adapter and the proper Distros software.Linux has a bunch of freeware when it comes to this so we may want to go that route.

2.34 Subsystem 4: Display Interface

We will use a capacitive touch screen display that is 6-9" large. 720x1280. This gives us the best response time with a low cost with the best display. A good example would be CFAF480272G-043T-CTS Display screen W/ built in Solomon SSD1963 controller

2.35 Subsystem 5: Power

AC to DC converter (appliance box only): To start with we will be creating an AC to DC converter from scratch or order a premade one to be implemented into the box. We believe the box will be placed on the wall near the door or somewhere inside the room. It will then need to tap into the buildings power which will be standard USA AC 120 volt outlet. All of the components inside will be DC so we need to convert the building power to our own use. We can estimate it will need the same amount of power as a ipad or phone and will use a 5 watt power adapter 100 – 240 ac Voltage (50-60 HZ) convert to 5 volt 1 amp DC.

Power Devices (appliance box only): We will be going with a battery device similar to those in smart phones and tablets. The design will ensure the device still running if the power goes out. A power saving program written in the processor to go into sleep mode when not in use. We guess a battery will be a good product, because of its small quantity which will work well will low power applications such as these.

3.0 Research Related to Project

Many ideas in the modern world are adaptations of other ideas for a specific purpose. Items may be adapted for a specific need such as lower power consumption, faster performance, or greater storage capacity. For this reason, we looked at similar and relevant technology that mirrors some of the hardware and functions included in Alfred.

3.1 Existing Similar Projects and Products

The following technology has similar functionality to how Alfred will work. Some of these items will have more features than Alfred. We have used these devices and software as inspiration for the designing of the Building Master.

3.11 Smartphone

Although the term ‘smartphone’ wasn’t coined until 1995, the first smartphone was created in 1992 and introduced in 1993 by IBM [1]. This smartphone was a hybrid phone together with a PDA that included an address book, calendar, calculator, note pad, appointment scheduler and touch screen display. The phone also had the capability to send and receive faxes, cellular pages, and e-mails. The device used single touch technology through which the user could input data via finger, stylus, or by a mini keyboard on the unit. There was also a basic predictive stylus input screen keyboard feature included in the software of the device. These smartphones could also be upgraded to run third-party applications, which meant buying extra hardware or memory specifically for the app.

Smartphones in the present can now communicate via a mobile network, wireless signal, Bluetooth, and near-field communication. This allows for a plethora of information to be exchanged and accessed for anyone on the go in a variety of ways. Most smartphones are also outfitted with rich, high-definition multi-touch screen displays that allows for users to play games, browse the internet, watch videos and do so much more in addition to simply making phone calls. Third-party applications can now be downloaded over wireless connection and no longer require one to upgrade the device to do so.

The most popular and highest selling operating system on smartphones is the Android platform developed by Google. Android OS is open source and allows for low-cost and customizable features for high tech devices. The open source system allows for developers and hobbyist to create applications to serve the needs of the many.

3.12 Handheld Data Management Computers

Handheld Data Management computers are commonly used in retail and warehouse settings for mobile management of data. These mobile computers communicate via WIFI connection to a main computer system that tracks changes in inventory, placement within the store or warehouse, general knowledge of the item and so much more. Input is typically done through one or a combination of three methods; touchscreen virtual input, keyboard/keypad manual input, and/or barcode scanner.

One big downside to these devices is the battery life. The operating system in combination with the wireless network adapter and barcode scanner typically put a larger strain on the energy consumption throughout the day. Some devices overtime will have a rough time keeping up with the day to day work in a warehouse environment. This causes the business to need extra units on hand in case the battery life doesn't hold up for an entire shift.

Another downside to these devices comes from the fact that all the data is stored in a server and no information is stored on the device. If the device isn't able to connect to the network, the device is not able to update inventory changes. If anything is done while the network is down, the user must input the changes at a later time. Because the change is not done instantly, this may cause inventory discrepancies if the worker forgets what was done or forgets to update all together.

3.13 Data Management Software

Rapid Inventory Management is an android based management program. This program allows the user to input data into a google drive spreadsheet manually or by use of a barcode scanner. The program allows for quick stock quantity updates, search function, data import/export using CSV files that support Microsoft Excel, XML files and Google Spreadsheets and has built in Camera barcode scanner. The program can be set to update internal memory of the device and/or may be set to update a database external to the device. When an item is initially scanned, the user is prompted to input a description, quantity, note and label. After the initial scan, each subsequent scan of the same barcode will prompt the user to edit the initial information or create another entry. The downside to this program is the lack of customization available to the user. Also, the program does not have the function to look up items online and auto fill for the user. [2]

Microsoft Access is a database management system and a part of the Microsoft Office Business Suite. Access allows developers and data architects to develop and build application software. Access stores data based on its Jet Database Engine and can import or reference data stored in other applications and databases. This program offers great customization for users. Microsoft Access does not currently support mobile technology at this time. Mobile programs and applications can be built to store and retrieve data from a computer that is running Microsoft Access. Due to Access being compatible with the Microsoft Office Business Suite, many companies and organizations use it for their database needs. [3]

3.2 Relevant Technologies

In this section we will look at technology that is relevant to the components that make up Alfred. This technology has help us to understand the internal workings of Alfred's components and how we may implement them into our design.

3.21 Item Identification Technology

The two most common ways of identification by computers or other electronic devices are by barcode or radio frequency. Barcodes are implemented in many industries as a means of identifying items within their business. By identifying items electronically, businesses are able to use the information in more ways than simple identification. For instance, retail outlets and grocery stores are able to maintain accurate stock information with ease due to the introduction of barcodes on merchandise. In addition, larger retail stores greatly benefit from barcodes helping to expedite the shoppers experience at checkout. These reduced wait times allow for the business to serve more customers throughout the day, increasing sales. Postal and shipping providers are able to use barcodes to track customer packages throughout the delivery process. With one scan, they can find out where the package originated, which processing centers it has visited, and where it will ultimately end up. In combination with web services, this tracking service has been upgraded so that customers may track their packages online at their own convenience.

Where barcodes are good for identification, some businesses prefer using RFID technology. RFIDs can be used as security features. Where barcodes can be seen and easily replicated, RFID signals are harder to reproduce without specialized technology. Many offices now use keycards with RFID chips to allow keyless entry for workers into the building or into secure rooms within the building. Employers are able to set time restrictions in addition to clearance levels on specific doors. Recently Disney has begun using RFIDs inside wristbands to link guests to their season passes. Rides in the theme park are also able to recognize when guests are riding and can link any pictures or videos from the ride to the users account. The user is then able to view and purchase the pictures and videos from the comfort of their own home without having to wait in lines at the park or carry around claim tickets [4]. Retail outlets have begun tagging dresses with RFID chips and putting smart displays inside dressing rooms. As the consumer tries on clothes, the smart screen can make suggestions to alternative choices that he or she might also enjoy.

3.22 Camera Technology

Cameras are able to do so much in this day and age. Cameras come standard on most cell phones and laptops to allow users to take photos and video to remember moments. In addition, these cameras can provide video calling to connect users virtually face to face over vast distances.   In sports cameras are used by referees to review plays and make calls. Cameras are used for security purposes as well. Businesses are able to review tapes when a theft is suspected in hopes to catch the culprit red handed. With pictures and videos now being stored digitally, image processing has increased the capabilities of the camera. Cameras with the proper software can recognize images such as barcodes eliminating the need for barcode specific scanners. With biometrics, cameras are used for fingerprint scanners and facial recognition. As cameras have grown in usefulness, they have increasing become integrated in new technology.

3.23 Wireless Communication Technology

The modern world has seen a huge shift to wireless communication over the past few decades. With several styles of wireless connections available, the opportunity of use is endless. Wireless mesh networks for instance are used to connect military laptops during operation in the field. Mesh networks are being implemented by organizations that help bring the internet to children at home so they may study for school. This program allows the children to exchange files and access the internet without a wired connection in their area. Utility companies are able to outfit meters with mesh network technology. Usage data is then transferred between units as it makes its way back to the company for billing. [5]

Bluetooth technology has application for uses that range from convenience to security. Bluetooth technology is most commonly known for phone applications. Wireless headsets utilize Bluetooth connection for use in the office or personal cell phone calls. Some automobiles also come with Bluetooth technology to allow for hands free calling or streaming music from a cell phone. For gamers, Bluetooth technology is used in wireless controllers to communicate with the main console. Bluetooth works only while in close proximity environments. For this reason, the connections can be used to determine when one device strays too far from the other connected device. This can be helpful in raising a ‘man overboard’ alarm on commercial ships.

Wifi is a commonly used type of wireless LAN connection for both personal and professional use. Wifi is most widely known for, and commonly mistaken for only, connecting laptops, tablets, smartphones, gaming systems and other devices to the internet. Wifi connections, however, can be created to connect many devices together while not connecting to the internet. An example of this would be connecting a laptop to a wireless printer. Gamers have also made use of wireless LANs to connect many laptops together to play games without the hassle of Ethernet cables or other physical connections. Wireless LANs are very useful for any project that requires a need to link many computers wirelessly at high speeds.

3.3 Strategic Components

Components are not always chosen on performance alone. In addition to performance, the choice may be based on factors such as price, availability, compatibility, ease of implementation and familiarity. In this section we will discuss some of these factors and the strategy behind how we will choose our components.

3.31 Embedded Controls

The embedded control is the most important part of any design. Above all else, it must meet the minimum requirements for all of the components inside Alfred. Our biggest decision with the embedded controller is whether to choose a processor or microcontroller. When talking about performance alone, the processor is the clear victor for computing needs but Alfred may not require the power the processor brings to the table. If we choose to use an operating system such as Windows, IOS, or Linux, we will definitely need a processor. However, if we build our own environment, we can use one or more microcontrollers together to suit the needs of the hardware.

Members in our group currently have a few types of microcontrollers readily available from our classes here at the university and various projects we have taken on ourselves. This means that in price comparison and availability, the microcontroller will be better than the processor as we will not have to invest any more money in the project nor worry about the possibility of stock delays. In addition, because a processor does not have RAM, ROM and other peripherals on the chip, the cost of going with a processor is pushed even higher when considering the extras needed for function. The processor will be a bit harder to implement as it requires being permanently attached to our PCB professionally. Microcontrollers on the other hand are much simpler to connect to a board and switch out given the board has the proper connections. In addition, we are much more familiar with programming and implementing microcontrollers than we are with processors.

3.32 Camera

The imaging component for Alfred will be used as a barcode scanner that will be able decipher a variety of barcode types. The camera will also catalog items in storage by taking pictures for easy reference. The biggest performance indicator with cameras is the resolution of the image. The resolution of the camera does not have to be the highest quality for the catalog purpose of Alfred as it is merely a tool for reference for the user. For the purpose of barcode and QR scanning however, we do realize that better resolution of the camera will allow Alfred to scan with fewer problems when dealing with size or quality issues. The camera will also need an auto/manual focus feature in addition to the resolution quality. If the camera is not able to focus properly the image may still turn out blurry even if the camera is capable of taking high resolution photos. If this happens, the software will have trouble correctly identifying codes from the blurry image.

When determining whether to build or buy a camera several factors weigh in. In terms of a monetary benefit, building a camera is less expensive than buying a premade unit and therefore helps when dealing with a tight budget. We have found a variety of mini cameras and parts, so availability will not be an issue for this component of Alfred. Our decision on processor or microcontroller will also play a big part in determining our camera selection when it comes to processing how the camera stores pictures.

3.33 Wireless Network

Wireless networks have the most performance considerations out of all of Alfred's components. The wireless network that we choose will serve two purposes for Alfred. The first will be used to link two Alfred devices together so they may access data stored in each device. The second purpose of the network is to allow the device to access and store information on an outside server. We will need a network that has a reliable connection so that edits to inventories can be updated to the server with little lag in the process and so that other Alfred units are able to view alternate inventories in real time by the user. Also, we must consider distance between Alfred units as many commercial buildings are large and may have considerable distance between the rooms Alfred will be installed on. The rate of data exchange would not be important for viewing the text inventory list but becomes more important if we include the transfer of catalog pictures for inventories as well.

The cost of implementing wireless networks will be similar for each of the types and due to many available implementation methods for each network, this will have little to no bearing on our decision making process in regards to wireless choice. That being said, some networks are easier to set up and maintain than others and that will be heavily weighed into our decision making process. We, as a group, are more familiar with wifi and Bluetooth technology and are less comfortable with mesh networking methods.

3.34 Screen Interface

Like any product, we need to be able to market Alfred. Any salesman knows that the best tactic to accomplish this goal is with a look of quality and professionalism that will attract potential buyers. This is where the display interface comes in and why we must also consider the physical appearance of this component. The screen must look sharp and be visually appealing. Size is also important, a larger interface will allow easier navigation through Alfred's data storage process while a small display will likely annoy and frustrate users. Durability is key for the product to maintain good appearance over time and make sure we keep customers coming back. We also need to consider cleanliness of the screen after repeated use, so we may look into smudge proof glass displays. The resolution of the screen must also be high enough to handle any pictures from the internal camera.

On the more technical side of things, the screen will function as both an input and output device for Alfred. We will need to consider the connections necessary to most easily install the interface. Some displays come with their own MCUs that offer easy hook ups for an added cost. There are also several type of touch technology that we must also consider when choosing the screen. The screen must be sensitive to the touch to avoid excessive wear and tear. Also, the type of touch technology should not reduce resolution in the screen.

3.35 Power

The most universal subsystem in Alfred is the power unit. All of the components of Alfred will need to be powered to function and we must find a way to get that power to each one. We propose that Alfred will be a wall mounted device that will tap into the power of the building. Our biggest concern is whether to build an AC to DC converter internally and hardwire Alfred to the building or buy a premade converter and have Alfred plugged into a wall socket. Hardwiring a unit in a commercial building would call for a professional electrician. This causes additional installation costs that may scare off potential buyers. However, having no visible wires would increase the quality feel we would like Alfred to have. Another consideration is the possible use of an internal battery. An internal batter may offer some perks such as uninterrupted work during a power outage. The battery may also allow Alfred to become mobile. If an item is large or heavy and cannot be lifted to be scanned, mobility would become a great asset to Alfred in lieu entering the information manually

3.4 Architectures and Related Diagrams

With many components able to be bought or created for Alfred, we decided to look up some architectures and diagrams of related components. This will give us a better idea of how we may approach creating specialized components for our specific purpose. This will help us in our decisions to build or buy for each section of Alfred.

3.41 Barcode Scanner

The traditional handheld barcode scanner makes use of an optical sensor to measure changes in light being absorbed by the black dashes of code and reflected by the white space between them. The light being measured is typically from a laser but can be other sources as well. The image sensor may also implore use of prisms or mirrors to scan the light over the bar code. After the image sensor reads the change in light, the internal components convert the signal to a digital input and send it through the control unit that processes the information and sends the data through the output device for the user. Below in figure 3.4.1 [6] is a functional block diagram of a barcode scanner.

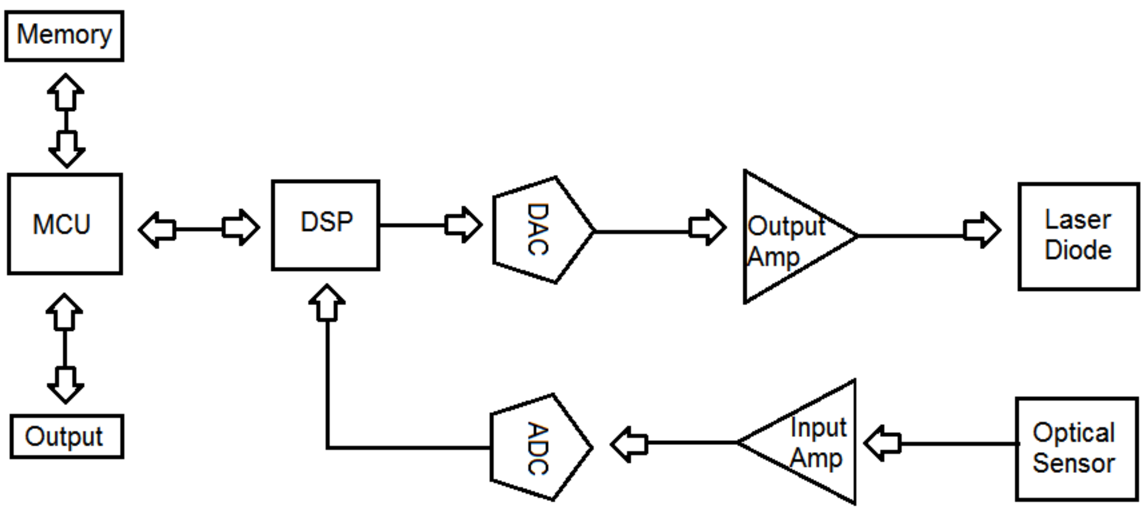


Figure 3.4.1 – Barcode Block Diagram

3.42 RFID Active Scanner

There are many types of RFID scanners that may accomplish different tasks in addition to decoding a RFID signal. An active scanner, as shown below in Figure 3.4.2, is a great example of this. The active reader not only reads signals from RFID enabled tags but also sends signals to the tags. The tags will then respond sending information back to the scanner. These scanners may therefore read multiple tags at once. This allows for filtering of information by the user to find specific tags in a group of items. The device may also ascertain relative location of items within the area. An active scanner also has the ability to update tags with new information or edit existing information on the tags.

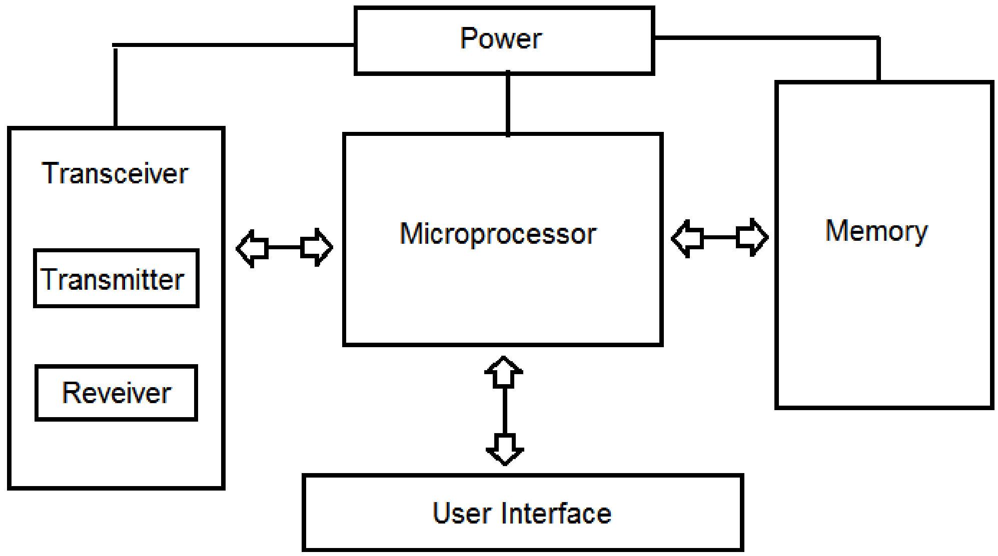


Figure 3.4.2 – RFID Block Diagram

In this design, the microprocessor handles the encoding and decoding of all tags and data for the design. Active scanners therefore heavily rely on software programming. The device also has internal memory to store entries in case of a network failure. This prevents loss of data when making updates to tags. [7]

3.43 AC to DC Converter

Buildings are supplied with alternating current, or AC, electricity by the power company. In order to obtain the necessary direct current, or DC, power for our device we will need an adapter. A basic transformer, like the one showed in Figure 3.4.3 below, is made up of a transformer, rectifier, filter, regulator and load component.

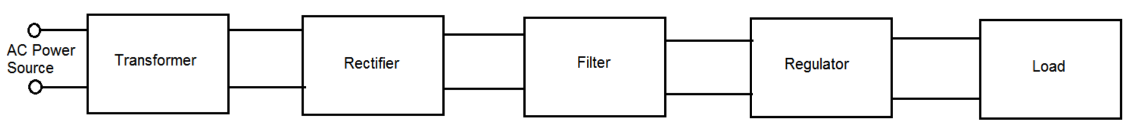


Figure 3.4.3 – AC Adapter Block Diagram

The transformer at the beginning uses inductance to translate the AC voltage from the wall to the voltage needed by the unit. The rectifier in the next step, typically a “bridge rectifier” as shown in Figure 3.4.5 [8] below, then uses diodes to make all the values of the voltage positive or the values all negative depending on what is needed. Next, the rectified signal will pass through the filter portion of the circuit. In this stage, the circuit implores capacitors that store and release energy during voltage peaks and drops. A voltage regulator is then applied to regulate the voltage within a certain value. This provides protection from an accidental surge or increasing feedback. The load on the end is the device to which you are supplying the DC voltage.

4.0 Project Hardware and Software Design

The prototype uses many different elements and products. The next section provides research into those elements and the different subsystems used by the device.

4.1 Initial Design Architectures and Diagrams

Project Alfred is broken into 5 major sub systems:

1. Embedded control sub systems
2. Camera input, RFID and cataloging input output sub systems
3. Networking and Communication
4. Screen I/O and user interface
5. Power

In order to capture all of the parts in those sub systems, project design is broken into two separate main designs: the hardware design, and the software design. The hardware design describes the overall hard ware parts that the sub systems will use to create the working prototype. On the other side, the software design controls the programing that will be necessary to control and process the user functions for the hard ware to execute. They have been broken down in the next two sections for review by the customer or product manufacture. The figure below provides the key for both the hardware and software diagrams. Each shape represents one of the main 5 subsystems used in creating project Alfred.

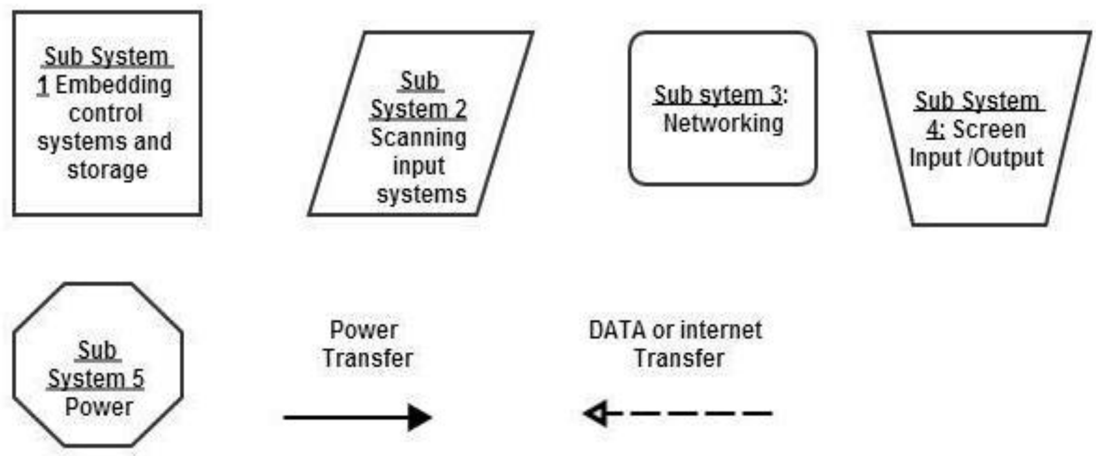


Figure 4.1.1- Hardware and software block diagram key

4.11 Hardware Design

The project has two separate pieces, the Door Box appliance and its hardware and the main frame and its hard ware. Because project Alfred works with a lab top or server computer as its mainframe the hardware for a computer comes standard and requires that of a normal computer. The computer we used to test our prototype for project Alfred has been tested and can working with the following hardware.

- Any type processor, dual core or better
- 8 GB or RAM preferred
- 250 GB of storage space (large enough to house the memory needed for holding company information.

For further information on specific necessities for computer products see individual sub systems for requirements such as hook up requirements and Networking for more information on mainframe required specs.

Alfred door box appliance comes standardly equipped with one LCD touch screen for user input/output to work with box on site. In addition it comes with one camera and RFID input for scanning and picture applications to store codes and jpeg information the user would like to input for archiving. Furthermore it's internals consist of 1 32 GB SSD and can be upgrade to large if needed, a microprocessor to store, manage, and retrieve data a wifi adapter for networking, and a DC battery for powering the device. Last, the device comes a plastic encasing to protect the insides with mounting equipment to attach to the wall, and with an AC to DC converter for wall plug in. The initial design of project Alfred uses the main sub systems below and they are connected to each as follows in the block diagram.

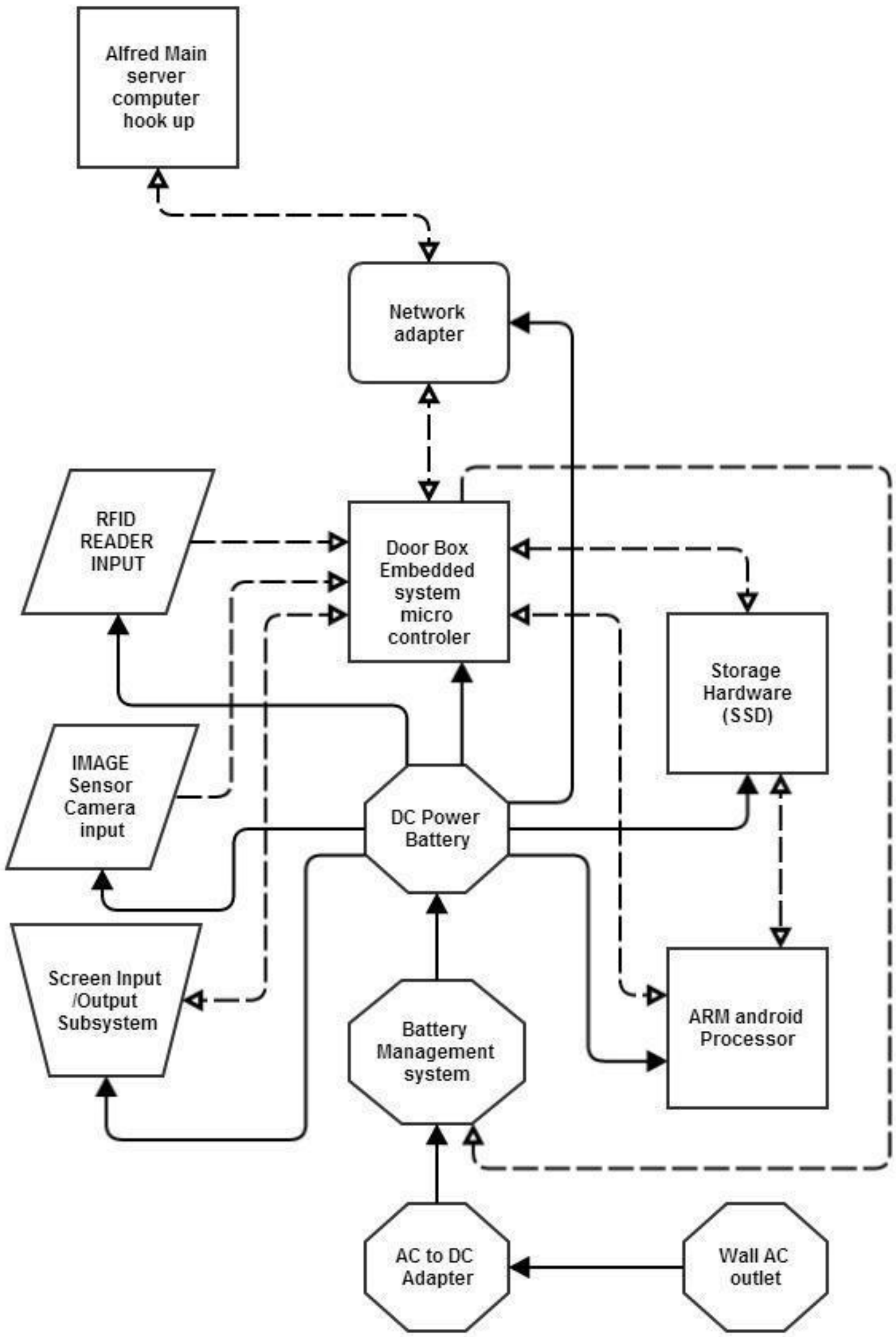


Figure 4.1.2 Hardware Block Diagram with Data and Power connections

4.12 Software Design

Project Alfred was designed to work with the software of another computer to help maintain and keep records for the whole building. The block diagram below shows a rough illustration of how project Alfred's software is interconnected together.

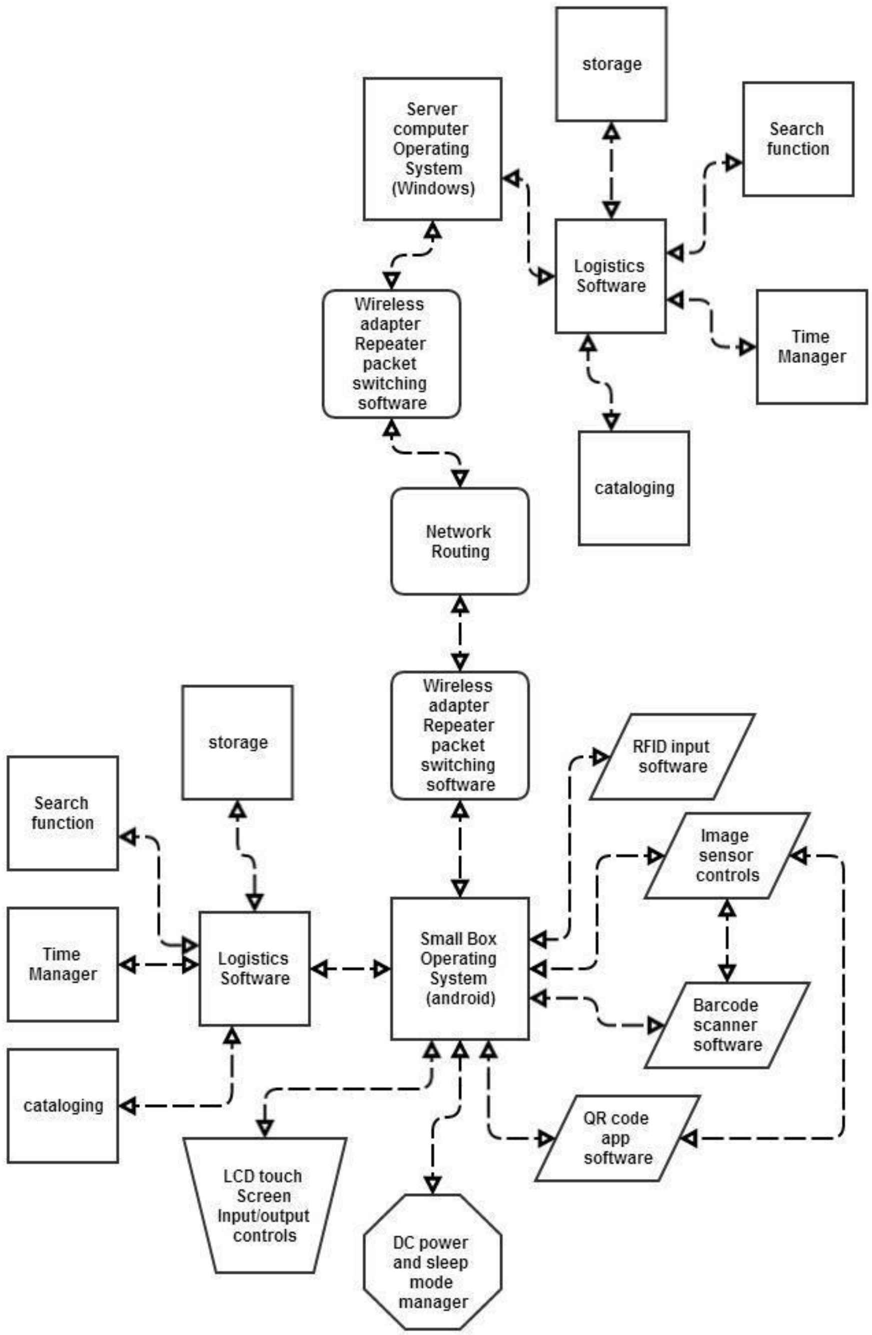


Figure 4.1.3 Software Block Diagram

The mainframe needs a standard operating system of windows 7 or better to run the programs need to run Alfred. The appliance prototype comes with its own standard software too intended to be compatible with a user's computer mainframe. The main software systems are that will be used in project Alfred include a logistics software that help with search and retrieval of data. In addition, each individual hardware piece comes included in the appliance boxes comes, with software needed to be compatible with the hardware attached to it. For this reason we will be using a higher level program language to design its software and control the functions of the device. For more information on coding design se section 6.3 for coding plan, as well as section 5 for the design summary on the software used.

4.13 Box Design

The box design is to be used at indoor locations first. Due to weather problems hazardous conditions make it difficult to design a weather proof box within budget. Furthermore, most companies are large inside building with few exceptions to the rules. The box will be designed around the same size of the screen but large depth to allow for increased components that are placed in the box. It is not meant to be a large appliance and take up much wall space, but to be a quick easy access to information on doors. The prototype is designed with in mind to be created on larger scale so if special case scenarios are needed, few design changes can be made for custom order. The product will be made out of normal anti-static plastic. Since the device is low power other than the ac to dc adapter that connects into the wall, the team is not worried about shock hazards. For this reason, materials into making the box can be normal plastics and non-static materials. The product will come in a case mounted to the wall next to the door. The product is being designed to be removed from the wall if needed for easier scanning, however the design is not intended for it to be removed from the wall.

4.14 Design Summary

In conclusion Alfred is designed as a team of products working together for one goal. The process only works if the user continuously updates the information, as well as maintains all hardware and software used by the system. This is not an automated system, but a apparatus designed to aid the user in accurate record keeping. For more information on each sub system separately, view the rest of section 4 further on in this report or see the index for locating the particular sub system needed for review. For the full design and build spectrum see the design summary in section 5.

4.2 Sub System 1: Embedded Control Systems

Embedded controls are programmable chips that interpret data from input devices, utilize logic on this data, and send control or data signals to output devices. Alfred will use microcontrollers and processors within the system to link together the input and output to facilitate overall proper function.

4.21 Processor and Micro Controllers

Alfred has a LCD display, camera, wireless communication, and several other components that must be able to transfer data and communicate effectively. For this reason, Alfred will need a microcontroller or processor that can complete multiple tasks simultaneously inside the device and handle the demands of each subsystem accordingly.

4.211 ARM Cortex A-9 Microprocessor

As Alfred's demands will be similar, the ARM Cortex A-9 processor was chosen as it is currently being used in many digital televisions, tablets, set-top boxes, and smart phones. The Cortex-A9 also comes with the TSMC 40 G Hard Macro. This macro allows for the peak performance in DMIPS while using less than 250mW per CPU. The Cortex-A9 uses a memory management unit (MMU) in place of a memory protection unit (MPU). While MMU's perform the same function as the MPU, there are more features in the MMU that are absent from the MPU. The MPU provides support for protection regions, access permissions and exports memory attributes to the system. The MMU includes the same support as the MPU but also includes support for cache control, bus arbitration and bank switching. The MPU would not be able to handle the complexity of the operating system needed for Alfred. Thumb-2 technology inside the processor also allows for 30% reduction in memory required to store instructions to help deliver peak performance of ARM code. TrustZone Technology will allow for reliable implementation of security applications Alfred may use in the commercial setting.

4.212 ARM Cortex M-4 Microcontroller

The ARM Cortex M-4 controller comes built into the TI-CC3200 chip (discussed further in this document) that will handle Alfred's wireless needs. This micro controller is designed to be highly efficient in digital signal controls as it comes ready with a variety of signal processing features. The controller has integrated sleep modes which allow for the high performance while maintaining low power consumption. Included in the M4 controller is the inclusion of the Thumb and Thumb-2 technology that was discussed earlier in the A9 processor section. The M4 Cortex also comes with three sleep modes for lower power consumption when device is idle and waiting for commands.

4.213 MSP430 G2553 Microcontroller

Alfred will also utilize the MSP430 G2553 micro controller. This specific microcontroller is optimized with five low-power modes for efficient energy consumption. It also achieves maximum code efficiency by utilizing a 16-bit RISC CPU in tandem with 16-bit registers. This version of the MSP 430 also comes with 10-bit analog-to-digital converter.

4.22 Storage, Catalog System and Search Program

4.221 Storage

There are two storage drives that were considered for Alfred's needs, hard disk drive and solid state drive. Alfred will need 16 GB of storage for data management. While both drives can boot the system, store our data and offer non-volatile memory, each has advantages and disadvantages we must consider.

Hard Disk Drives store data in magnetic coating on top of platters. While these platters spin, an arm accesses data using a read/write head on the end. Due to the moving parts of HDDs, the memory can be compromised if the unit drops or is shaken while operating. HDDs also implement fragmentation when storing data. Fragmentation occurs when data on the drive has to be split on the disk due to limited space on the drive. This can cause a slower read/write time as more information is stored on the drive. Due to the moving parts, HDDs are also very difficult to make smaller than 1.8 inches and create noise while operating.

Solid State Drives store data in interconnected flash memory chips. These chips can even be installed directly onto a systems motherboard. SSDs are more expensive than HDDs but with zero moving parts, SSDs are a safer way to store data in a commercial environment. SSDs are also faster than HDDs when booting up and stay faster throughout operation. Applications, searches and data transmission will all be faster with better overall performance. SSDs do not need to use fragmentation when storing data which increases the speed at which read/write operations take place. [9]

Due to our storage need of 16GB, we have determined we can use a solid state drive as the only disadvantage is the cost of the drive. Alfred will use a Kingston SSDNow S100 Industrial solid state disk.

SSDNow S100 Specifications [9]

- 2.5" x 2.5"
- 16GB of storage
- SATA II interface
- up to 230MB/s sustained sequential read
- up to 75MB/s sustained sequential write
- 1,000,000 hours MTBF
- 1.08 W Idle power consumption
- 2.26 W Active power consumption

4.222 Catalog System

Alfred is the butler of your warehouse and must know where things are stored at all times. It is important to have all the necessary information readily available. Therefore Alfred will catalog items based on several factors:

- Name of the item

- Description
- Quantity of item
- Room of storage
- Placement inside room – if needed
- Time and date item was stored

The user will be able to organize the list of items based on these factors. As the user taps the list names, the data will organize itself based on alphabetical or numerical order. In addition to the written data, Alfred will be able to store a photo of the item for a visual catalog of them materials being stored. This will cut down on time taken by employees searching through boxes of items only to find the item was not what they were looking for.

Alfred will have to learn from the user initially to create the catalog system. When the user scans a new barcode for instance, he/she will have to populate the information for each category except the date stamp which will always be automatic. Any subsequent scan of the same barcode will prompt Alfred to autofill information for name of item, description of item and room it is being stored in. The quantity tab will autofill with a value if more of that item is currently being stored. Specific placement inside the room will always be a manual entry. The user will also be able to add in extra categories to further catalog items if desired. Examples of extra categories include, size, weight, color, etc.

4.223 Search Bar

With an entire commercial building full of Alfred devices, a search function is necessary to finding items quickly and efficiently. The search bar will either be placed permanently at the top of the user interface screen or the user will have to push a button to call the search bar into focus. Users will be able to search by name, description, or user created category. Users may also search for multiple parameters at once. For speed purposes, the search bar will also have the option to search based on location, such as only searching the second floor storage rooms.

4.3 Sub System 2: Camera and RFID Input

To start off with, Project Alfred's appliance devices comes with a mini camera used in conjunction with applications to help keep inventory and picture records of items going in and out of the room. The camera can be utilized from the main screen of the appliance door product to take pictures of the products that go into the room. The devices storage can hold pictures in the information to be able to retrieve any images at the location device or the main frame running project Alfred. However, the camera is fixed to the device, so the user will have to maneuver what objects, or people the camera needs to scan or take pictures of. As a result you can upload your own pictures to the appliance device at will, see how to on the controls and storage for information on pictures not taken with this camera

4.31 Camera Input

A digital camera is the whole package of devices and component make ups. The many of the parts associated with a digital camera, its lenses, a programmable control chip (micro or processor), RAM and many other devices are incorporated to make a camera operate. These devices can already be found in Alfred and many other smart devices prior to making a camera a camera considering a digital camera is programmed computer function to take pictures. The true part that separates a camera from a computer is an image sensor attached to those parts then lenses are placed in front to support light refraction. That is how you create a camera small enough for low power devices such as a box. More information on image sensors is provided below.

An image sensor is a device that converts an optical image into an electronic signal. It is used mostly in digital cameras, camera modules and other imaging devices. Early analog sensors were video camera tubes; currently used types are semiconductor charge-coupled devices (CCD) or active pixel sensors in complementary metal–oxide–semiconductor (CMOS) or N-type metal-oxide-semiconductor (NMOS, Live MOS) technologies.

Today, most digital still cameras use either a CCD image sensor or a CMOS sensor. Both types of sensor accomplish the same task of capturing light and converting it into electrical signals. Each cell of a CCD image sensor is an analog device. When light strikes the chip it is held as a small electrical charge in each photo sensor. The charges are converted to voltage one pixel at a time as they are read from the chip. Additional circuitry in the camera converts the voltage into digital information. A CMOS imaging chip is a type of active pixel sensor made using the CMOS semiconductor process. Extra circuitry next to each photo sensor converts the light energy to a voltage. Additional circuitry on the chip may be included to convert the voltage to digital data.

The image sensor project Alfred is incorporating is called the MT9P031 Image Sensor from Micron Technology, INC. The image sensor is 1/2.5-inch CMOS digital image sensor with an active-pixel array of 2592H x 1944V. It incorporates sophisticated camera functions such as windowing, binning, column and row skip mode, and snapshot mode. It is programmable through a simple two-wire serial interface. The sensor can be operated in default mode or programed for frame size. Some of its features include:

- High frame rate
- Superior low-light performance
- Low dark current

- Global reset release, which starts the exposure of all rows simultaneously
- Bulb exposure mode, for arbitrary exposure times
- Snapshot mode to take frames on demand
- Horizontal and vertical mirror image
- Column and row skip modes to reduce image size without reducing field of view
- Column and row binning modes to improve image quality when resizing
- Simple two-wire serial interface
- Subframe windowed readout
- Programmable controls: gain, frame rate, frame size, exposure
- Automatic black level calibration
- On-chip PLL [19]

In, addition to the features listed above an image sensor is small relatively dimensions are in millimeter designs. Thus, an image sensor can be hooked up to an existing computer or smart device with relative easy.

The sensor is programed with the two wire serial bus which communicates with the array control, and analog signal chain, digital signal chain. The sensor is a 5 megapixel array and the control circuity resetting and reading the rows one by one in turn. Each pixel read out give a twelve bit value from the ADC (application delivery controller) and passed through to a digtal processing chain. The Block Diagram below shows the process of how the sensor works.

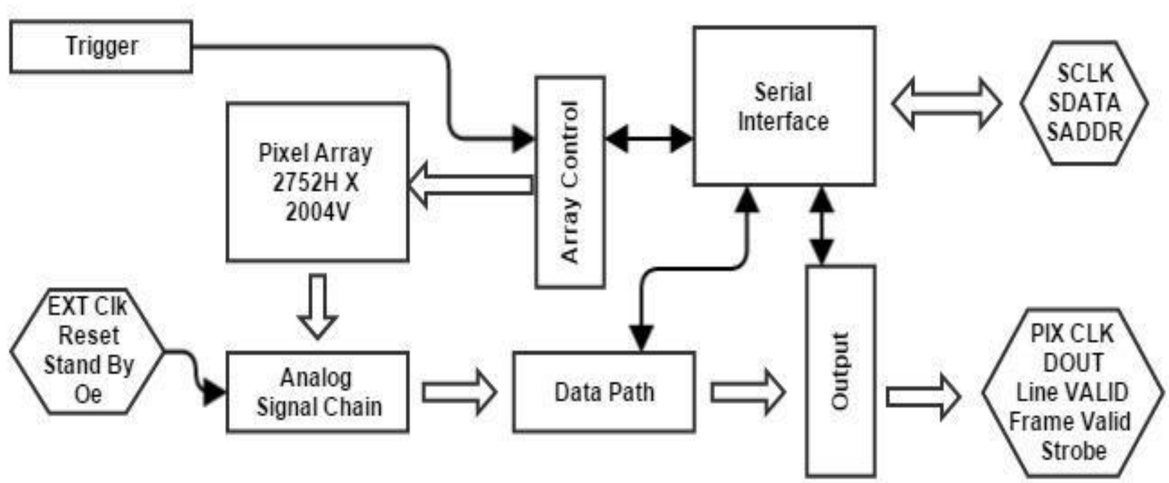


Figure 4.3.1 MT9P031 Block Diagram [20]

The block diagram above shows how the image sensor works individually from other parts in a digital camera. As said previously, a digital camera is a computer with an image sensor. Therefore, in order to process the images taken we can connect the image sensor to the controller or processor that will be used by project Alfred.

4.32 Barcode Input:

A barcode is a code consisting of a group of printed and variously patterned bars and spaces and sometimes numerals that is designed to be scanned and read into computer memory and that contains information (as identification) about the object it labels. Most barcodes used today are under the UPC or Universal Product Code. These codes are designed to have both the thin thick bars as well as a 12 digit UPC code. An example of a barcode is shown below:

Figure 4.3.2 Barcode example [21]

The first six digits of a barcode represent the manufacture identification number and provide guidelines on how to use the codes. Subsequently, the next 5 are the item number that is created by the person employed by the manufacture to keep track what the product is. The last number is the check digit, this digit let the scanner determine if it scanned the number correctly or not. Provided is the steps on how to get the check digit for the barcode example figure 4.3.2 above. [21]

1. Add together the value of all of the digits in odd positions (digits 1, 3, 5, 7, 9 and 11). **6 + 9 + 8 + 0 + 0 + 9 = 32**
2. Multiply that number by 3. **32 * 3 = 96**
3. Add together the value of all of the digits in even positions (digits 2, 4, 6, 8 and 10). **3 + 3 + 2 + 0 + 3 = 11**
4. Add this sum to the value in step 2. **96 + 11 = 107**
5. Take the number in Step 4. To create the check digit, determine the number that, when added to the number in step 4, is a multiple of 10. **107 + 3 = 110** The check digit is therefore **3**. [21]

A barcode scanner is composed of three parts: the illuminator, the decoder, and the sensor/convertor. The barcode scanner illuminates the barcode with red light using the illuminator system. The sensor/convertor part of the scanner then detects the reflected light. Once the light is detected, an analog signal is generated. This signal contains varying voltage based on the intensities of the light reflection. The analog signal is converted by the sensor into a digital signal. The digital signal is then interpreted by the decoder. The decoder then sends the information to the computer attached to the scanner.

However, the number is only part of a barcode. The barcode gets scanned into a system using the thin and thick lines above the numbers. When scanned and read by a barcode scanner the barcode gets the numbers you see below, allowing for an efficient reading without having to type in the number every time. For this reason, we have chosen this as one of our ways to keep inventory of equipment in the room such as breakers, furniture, and other things that would remain in the room so the company knows what belongs there. Barcode can also, keep track of inventory of materials that go in and out of the room such as stockroom supplies, and food for the refrigerator. The user will choose the barcode scanning application on Alfred appliance at the room desired, scan the barcode by holding up the barcode for the camera to read, and then, choose how many will go in or out of the room. Under those circumstances, the camera and software of the appliance boxes will be equipped a with a barcode scanning app. This app is found on many devices is provided by many applications stores used in smartphone and tablet technology for easy access. When prompted the camera will bring up this image:

To scan properly the user will line up the horizontal line perpendicular to the vertical lines on the barcode. Once the barcode has been scanned it will prompt the user to continue its process. Provided that Alfred also, has Wi-Fi technology, you can also scan in your desired device and search the web for more information of the object that is scanned in to the system.

To create this code, and because Alfred utilizes android technology, the Barcode reader will be created by using ZXing (Zebra Crossing) library. In other words, ZXing is an open source library that provides access to tested and functional barcode scanning on Android. With regards to ZXing, Alfred doesn't need to worry about the scanner not being installed, because it integrates the camera to do the scanning by converting camera feed to scanning feed. Thus, allowing for focus on manipulating what was scanned, rather than spending time making sure the correct item was scanned. In addition, ZXing is open source we can import the code directly into project Alfred and therefore is prefect for android scanning, but allows for change in functionality. Alfred adapts the ZXing code for its purpose in storing the information and amount of the barcode scanned by the original program.

4.33 QR Code Input

Created in Japan where they are most commonly found, Quick Response codes or QR codes for short were produced to be a quicker and easier read code by cellphones, tablets and smart devices. Qr codes were created mostly for advertising and business to describe a product, link the scanner to a website to purchase the product, or even give a coupon to the devise that just scanned the code. Moreover, a QR code is two dimensional code, which is considerably more useful than a standard barcode, because they can store and digitally present much more data than, just a company number, and an item number. Above all, other than barcode, a QR code is the most scanned code in modern technology today. Consequently Alfred has been provided with a QR scanning device, just in case the business tends to move towards the QR scanning industry, due to the more information the codes can hold.

4.331 Design of a QR Code

Unlike one-dimensional barcodes that were designed to be mechanically scanned by a narrow beam of light, a QR code is detected by a 2-dimensional digital image sensor and then digitally analyzed by a programmed processor. The processor locates the three distinctive squares at the corners of the QR code image, using a smaller square near the fourth corner to normalize the image for size, orientation, and angle of viewing. The small dots throughout the QR code are then converted to binary numbers and validated with an error-correcting code. A QR code can have a max of 7,089 characters if numeric only, to up to 1,817 characters if Kanji/kana the most common QR codes used today.

QR codes can be easily created, by using Kaywa.com or the new Google QR generator that turns URLS into QR codes for you. As a consequence, with a wealth of information with such an easy setup to create, QR codes will be used by many companies for Alfred to scan for the companies.

4.332 QR Code Scanning

A QR code scans the exact same way as a barcode except, In the case of QR code scanner, the decoder sends the information to your mobile phone instead of a computer. The app you download for your phone that is a QR code scanner contains the illuminator, which is the red light that runs across the screen when you open the app. The sensor and decoder then work to decode the QR code. Then the decoder sends the information to your phone, and you will be able to see where the QR code was supposed to take you. There are so many QR code apps that can be chose from to work with due to the popularity in Japan that the product can pick from any of the following below:

- QR Pal
- Barcode Scanner
- ShopSavvy
- ScanLife
- QuickMark
- QR Droid
- Fun 2D QR Reader
- CertainTeed QR Code Reader
- ZXing

Due to this abundance of QR code apps, all work the same and Alfred will come with a pre-installed QR code app. Due to the fact that the barcode scanner that comes with Alfred, also comes with a 1 dimensional barcode scanner and a 2 dimensional barcode scanner they will both be installed together.

4.333 User Warning

Due to the 2D image and how QR code is generated, anyone can gain access to the process of creating the codes. Therefore, it has been used to create links to malware and virus software. Alfred will be equipped with a QR scanning app, but is not responsible for what the user will scan into to the system. With that in mind please remember to only scan codes from trusted brands and sites and remove viruses as soon as the problem occurs.

4.34 RFID Input

RFID stands for Radio-Frequency Identification. The acronym refers to small electronic devices that consist of a small chip and an antenna. The chip typically is capable of carrying 2,000 bytes of data or less. The RFID device serves the same purpose as a bar code or a magnetic strip on the back of a credit card or ATM card; it provides a unique identifier for that object. And, just as a bar code or magnetic strip must be scanned to get the information, the RFID device must be scanned to retrieve the identifying information.

A significant advantage of RFID devices over the others mentioned above is that the RFID device does not need to be positioned precisely relative to the scanner. In contrast, RFID devices will work within a few feet (some up to 20 feet for high-frequency devices) of the scanner. RFID technology has been available for more than fifty years. It has only been recently that the ability to manufacture the RFID devices has fallen to the point where they can be used as a "throwaway" inventory or control device. Alien Technologies recently sold 500 million RFID tags to Gillette at a cost of about ten cents per tag. Due to this project Alfred will be incorporated with RFID technology to help keep track of products going in and out of rooms without the need of position precisely scanners (barcodes, magnetic strips).

4.341 How a RFID Works

A Radio-Frequency Identification system has three parts:

1. A scanning antenna
2. A transceiver with a decoder to interpret the data
3. A transponder - the RFID tag - that has been programmed with information.

The scanning antenna puts out radio-frequency signals in a relatively short range. The RF radiation does two things:

1. It provides a means of communicating with the transponder (the RFID tag)
2. It provides the RFID tag with the energy to communicate (in the case of passive RFID tags).

This is an absolutely key part of the technology; RFID tags do not need to contain batteries, and can therefore remain usable for very long periods of time. The scanning antennas can be permanently affixed to a surface; handheld antennas are also available. They can take whatever shape you need; for example, you could build them into a door frame to accept data from persons or objects passing through. When an RFID tag passes through the field of the scanning antenna, it detects the activation signal from the antenna. That "wakes up" the RFID chip, and it transmits the information on its microchip to be picked up by the scanning antenna. In addition, the RFID tag may be of one of two types.

- Active RFID tags have their own power source; the advantage of these tags is that the reader can be much farther away and still get the signal. Even though some of these devices are built to have up to a 10 year life span, they have limited life spans.
- Passive RFID tags, however, do not require batteries, and can be much smaller and have a virtually unlimited life span.

RFID tags can be read in a wide variety of circumstances, where barcodes or other optically read technologies are useless. The tag need not be on the surface of the object, the read time is typically less than 100 milliseconds and large numbers of tags can be read at once rather than item by item.

4.342 ID Design

Project Alfred comes with an intergraded TIDM-RFID-Transceiver. The transceiver provides reference design outlines the required components, layout considerations, and provides firmware examples to implement RFID into applications for extracting RFID data from transponders. RFID enables access control, prepayment, and inventory management. The reason this transceiver is chosen because the documentation, hardware, and example code provided, allows implement RFID functionality with an MSP430 or other MCU of choice. The transceiver comes with:

- Integrated PCB antenna
- U.FL connector allows customers to develop and connect an antenna of their own design
- Test points for SPI code development/debug
- Onboard +3VDC to +5VDC boost converter allows evaluation at full power
- Used for reading and writing RFID transponders
- Design is tested and includes firmware, GUI Software, hardware design, and user's guide [22]

This will allow for easy testing of the product with hands on integration of our embedded controller or computer with an RFID reader. The figure bellow shows the block diagram for the design of the TIDM- RFID transceiver used in project Alfred, as well as the RFID schematics:

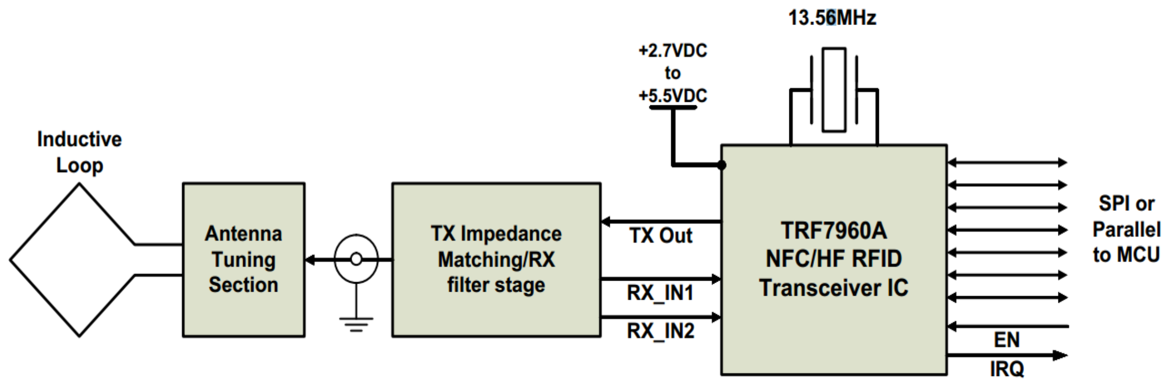


Figure 4.3.4 TIDM RFID transceiver block Diagram [22]

The RFID transceiver uses 5 volts of DC power max and will operate with any MSP 430 design. With this in mind, the project team has decided to order it integrated into the microcontroller itself for simpler design and more compact components. For more information on how the RFID transceiver connects to other parts including the MSP430 see design summary of products.

4.4 Network Sub System

Since we want to be able to access and store the data stored by Alfred the Door Master remotely, he will require networking capabilities. The data that needs to be sent will include the object's designation, amount of said object, timestamp of when object was last placed in the room, and the barcode of object if applicable. Since we want this to be easy to use, as well as easy to install, LAN and fiber cables are out of the question. The best way to do this then would be wirelessly.

4.41 Wireless Networks

We have the option of using radio signals to set up a Mesh Network, Bluetooth to setup a wireless personal area network (WPAN), or WIFI to setup a wireless local area network (WLAN) to send the data. Whichever network type we chose will need to be able to send and receive a good amount of data. We also want a large data transfer rate as well as speed in order to keep Alfred up to date at all times.

4.411 Mesh Network

The first option that is available is to setup a basic mesh network using simple radio signals. A wireless mesh network in its most basic form a group of peer to peer radio devices that transmit data from one to another until it reaches its final location.[26] Doing this it is possible to transmit data over long distances or around interference provided there are nodes available.[26] The main advantages of a wireless mesh network are:

- The more nodes the bigger and faster the network
- Convenient when cable connections are unavailable, expensive or too work intensive to install
- Self-configuring, meaning they automatically add new nodes when they pop up and remove old nodes when they leave.
- Great when direct wireless transmission are not possible
- Automatically finds the best route based on speed and reliability
- Easy install and removal

However wireless mesh networks have their disadvantages:

- Increased bandwidth usage
- Large number of spaced out nodes required to cover larger ranges

Wirelessly meshed networks have many really great advantages. However, with basic radio signals (AM or FM for example) you run a high chance of interference inside a building from not only the building walls, but also any other electronics inside. Furthermore to get a large range in the network you would need many Alfred boxes. That is very bad from a consumer standpoint. A consumer shouldn't need to buy multiple Alfred's to be able to look up the stock in the storage room on the other side of the building. There would also need an antenna to receive and transmit data attached to the central hub that was storing it for Alfred. So, in terms of using simple radio signals to build a mesh network for Alfred, it's a poor choice. The following is a visual example of a wireless mesh network.

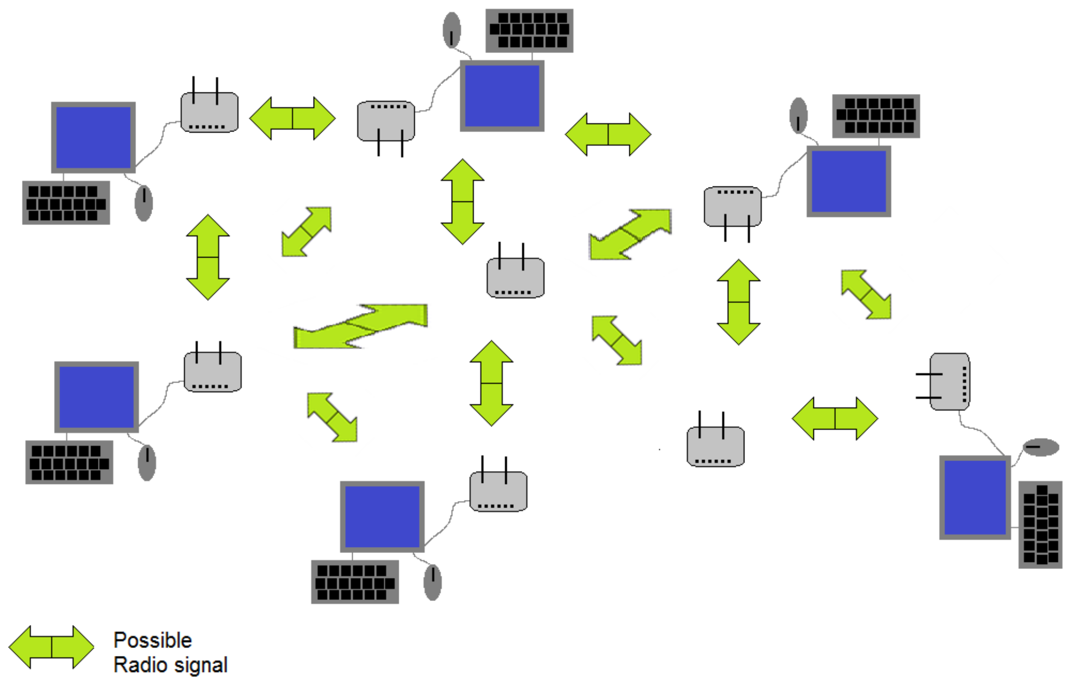


Figure 4.4.1

4.412 Personal Area Network

The next choice is to use Bluetooth to setup a wireless personal area network. A WPAN is usually a smaller network of devices connected through Bluetooth or infrared data association. Ideally, when objects that are in the WPAN are within a close enough range they can transmit data as if they were wired to each other. [27] They also have good security, being able to prevent interference or unauthorized access. The main advantages to a WPAN are:

- Very tight security
- Quick and simple setup
- Low power requirements

Since WPAN technology is still in its early stages, there are some significant disadvantages:

- Generally have a limited range of about 10 meters direct or around 2 meters if it passes through an object like a wall
- Low bandwidth

Heavy security and easy setup can really benefit Alfred. Nevertheless, the limited range of the network, especially when including interference, would really cripple Alfred. Needing to be no more than 10 meters from the room the box is in (even closer since it would likely transmit through a wall) or the central hub would defeat the purpose. Furthermore, the low bandwidth could really limit how much data Alfred could transfer as well as how quick it takes to update the user on what is stored in the room. While a WPAN is a viable choice for Alfred it has hefty limitations.

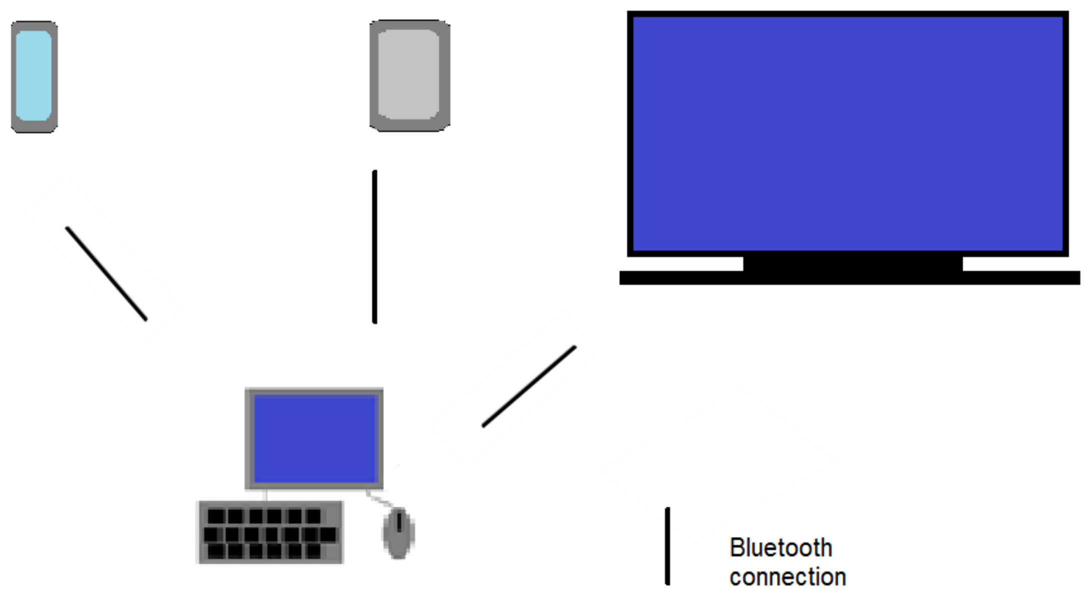


Figure 4.4.2

4.413 Local Area Network

The third option is to use a Wireless Local Area Network (WLAN). WLAN's are generally defined as a network that links at least two items using some form of wireless connection. [28] WLAN is very popular and is wide circulation, most homes, schools, and businesses have some form of WLAN operating. The most common distribution method of WLAN is a brand known as WIFI. It also allows for the users on the network to be mobile on devices such as phones and laptops, keeping up a strong connection as the user moves. Most WLAN's have a decent level of security as well. Some of the biggest advantages of WLAN are:

- Scalable network architecture
- Mobility
- Simple to moderately difficult installation
- Generally lower initial setup cost
- Access to network available anywhere in range

As with all things, there are some disadvantages as well:

- Number of usurers increase the data transfer rate decreases

- Hard to completely guarantee security
- Distance devices can be operated determined by the standard distance and is reduced by walls and other interference.

Scale, simplicity, and low cost can be a huge asset to Alfred. The mobility provided by WLAN's, as well as ease of access from different locations, would be great for the user as well. The decrease in data transfer rate could become a problem if Alfred was scaled up and used in many rooms but could be overcome by only connecting to the WLAN when it needs to transmit updates. The central hub will already have all of the data from each of Alfred's boxes, so the user can access the data at each Alfred box directly or from the data stored at the hub. The security should be strong enough for Alfred's intended purpose as well.

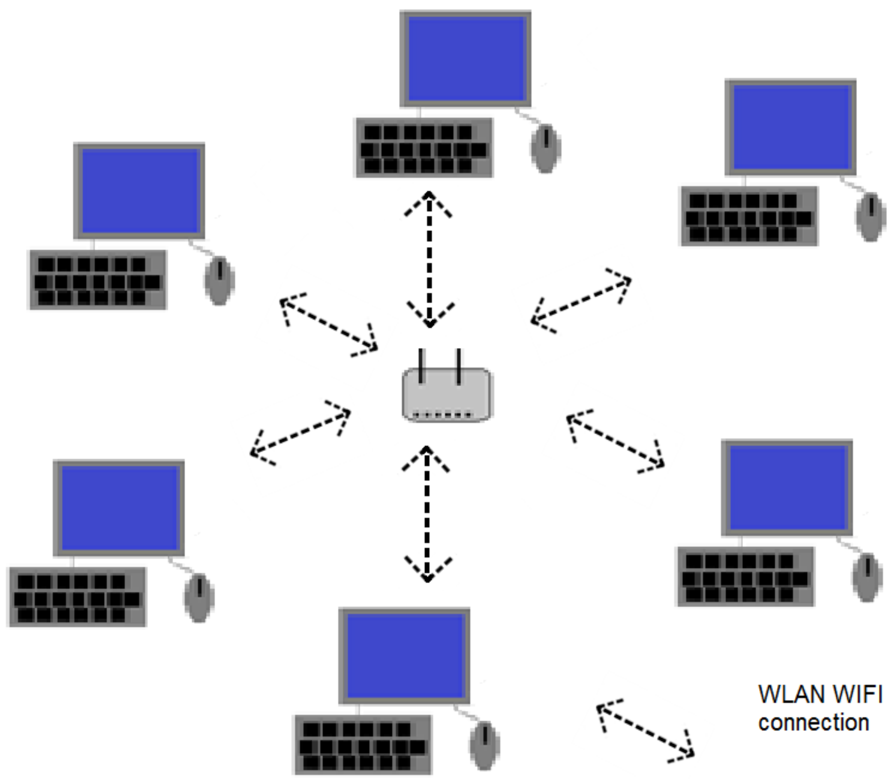


Figure 4.4.3

4.42 Networking Standard

Since we will be using WIFI as our wireless communication method we will need to use a wireless network standard. IEEE sets the standard for WIFI communications, and the standard that they use is 802.11. The purpose of these standards is to improve network throughput as well as set the standard GHz band for frequency. IEEE is constantly reviewing and updating the standards for WIFI to keep it current. There have been several version of 802.11 the most recent being 802.11ac in 2013. The next major update is 802.11ah being estimated for 2016 [29]. The diffrences in each standard can generaly be summarized by changes in frequency, bandwidth, stream data rate, allowable MIMO streams, modulation and approximate range.

4.421 802.11

Most chips currently on the market use the 802.11ac or 802.11n networking standard. In 802.11 data sent as well as managing and controlling wireless links is done through frames. Each frame has a standardized section which contains a:

- Protocol version: contains two bits, identifies version
- Type: contains two bits, used to recognize type of WLAN frame
- Subtype: contains four bits, used to help identify each frame
- ToDS and FromDS: one bit each designates if the frame is for a distribution system. All frames except control, management and IBSS network internal communications will have at least one bit set
- More Fragments: one bit, every frame has this bit set except the last one
- Retry: one bit, set when a frame Is resent
- Power Management: one bit, indicates the power status of the sender
- More Data: one bit, used to buffer frames in distributed systems
- WEP: one bit, toggles to one after frame has been decrypted
- Order: one bit, set only when strict ordering delivery method is employed.

[29] These are then followed by two bytes that contain the Duration ID field. The field will be one of three types, Duration, Contention Free Period (CFP) or Association (AID). Every frame can contain up to four MAC address fields. The first address is the receiver followed by the transmitter [29]. The third is generally used for filtering purposes. The next two byte section is the Sequence Control field. This section is used to eliminate duplicate frames. Next is the optional two byte quality of service field. This is then followed by the frame body field. The body field can very anywhere from zero to two thousand three hundred and four bytes. Finally the last four bytes are the Frame Check Sequence (FCS) also referred to as the Cyclic Redundancy Check, it is used check the correctness of the frame.

The two most common 802.11 versions on the market now are 802.11n and 802.11ac. The following table compares the 802.11n and 802.11ac standards based on by changes in frequency, bandwidth, stream data rate, allowable MIMO streams, modulation and approximate range. While 802.11ac is the newest version 802.11n is more common since it has had more time in the market. The following tables will show and make it easy to compare 802.11ac and 802.11n.

802.11	Frequency (GHz)	Bandwidth (Mhz)	Stream Data Rate (Mbit/s)	MIMO Streams	Modulation	Indoor/outdoor Range (ft)
n	2.4/5	20	7.2, 14, 21.7, 28.9, 43.3, 57.8, 65, 72.2	4	OFDM	230/820
		40	15, 30, 45, 60, 90, 120, 135, 150			

Figure 4.4.4 data taken from wiki 802.11 [29]

802.11	Frequency (GHz)	Bandwidth (Mhz)	Stream Data Rate (Mbit/s)	MIMO Streams	Modulation	Indoor/outdoor Range (ft)
ac	5	20	7.2, 14.4, 21.7, 28.9, 43.3, 57.8, 65, 72.2, 86.7, 96.3	8	OFDM	115
		40	15, 30, 45, 60, 90, 120, 135, 150, 180, 200			

		80	32.5, 65, 97.5, 130, 195, 260, 292.5, 325, 390, 433.3				
		160	65, 130, 195, 260, 390, 520, 585, 650, 780, 866.7				

Figure 4.4.5 data taken from wiki 802.11 [29]

As you can tell from the table above, 802.11n has more range than 802.11ac. However, 802.11ac has many more options when it comes to bandwidth as well as two times the number of MIMO streams. Both use OFDM for modulation.

4.422 OFDM

Orthogonal Frequency-Division Multiplexing (OFDM) is the method used by the IEEE standard 802.11 to encode digital data onto multiple carrier frequencies. [30] It is one of the most common schemes for wideband digital communication for wired and wireless. Using a digital multi-carrier modulation method, it uses a large number of tightly spaced orthogonal sub-carrier signals to carry data on several parallel channels. [30] Then each sub carrier uses a conventional modulation technique at a low symbol rate. This maintains data rates similar to normal single carrier methods in the same bandwidth. The biggest benefit of using OFDM is its ability to deal with severe channel conditions. Its low symbol rate makes a guard interval affordable which makes it possible to significantly reduce or eliminate inter-symbol interference. [30] It also yield signal to noise ratio improvements. OFDM also allows for signal frequency networks, where several adjacent transmitters can send the same signal at the same time on the same frequency. This allows for the signal from multiple far away transmitters to be combined rather than interfering with one another. The main advantages of OFDM are:

- High spectral efficiency
- Allows use of single frequency networks
- Adaptable to negative channel conditions
- No need for tuned sub channel filters

- Large narrow band co channel interference
- Strong defense against inter-symbol interference

The few disadvantages are:

- Sensitive to the Doppler Shift (the change in a frequency for an observer moving relative to its source)
- Frequency synchronization sensitivity
- Efficiency lost due to cyclic prefix/guard interval

Based on these advantages it is easy to see why OFDM is the method used by the IEEE. It is very efficient as well as accurate. Furthermore, the lack of the need for a sub channel filter is not only convenient but cost effective. Its ability to handle negative channel conditions is one of its greatest qualities, allowing it to work in areas where most methods couldn't. This makes it ideal for Alfred. Using OFDM Alfred would be able to accurately and efficiently transmit data despite poor channel conditions. Sensitivity to Doppler Shift won't have a major impact on Alfred since it is expected for the box as well as the server to not really mover often if ever.

4.423 DSSS

Direct Sequence Spread Spectrum (DSSS) is a signal modulation technique. DSSS uses phase shifts a sine wave in a way that looks random. This shift is actually creating a string of seemingly random noise to create "chips" that are used as symbols that have a significantly shorter duration than an information bit. [31]This gives it a much higher chip rate than bit rate. The receiver can then use the chips to reconstruct the signal in its original form. This has the benefit of being jamming resistant, hampers interception, multiple users can use the same channel, the timing is decided by the transmitter and receiver.

4.43 Chip Options

Since building a wireless transmitter and receiver from scratch would be very intricate and time consuming, it would be best to incorporate one already made. We will be sticking with TI chips for the most part, since we have a familiarity from using the MSP430 as well as their online support. This still leaves us a wide variety of chips to choose from, especially considering we can use Bluetooth, radio or WIFI to setup the wireless connection. We further narrowed it down to the WL18XX MOD series, the CC2520 or the CC3200. These chips are not only all low powered and capable of meeting our wireless needs, but are well within our budget. It chip has its own individual advantages and disadvantages.

The WL18XX MOD series has four different chip options all on its own. They are the WL1801 MOD, WL1805 MOD, WL1831 MOD, and WL1835 MOD. All of these chips are FCC certified and WIFI 802.11bgn capable. The WL1805 MOD and WL1835 MOD also have MIMO and MRC capabilities. Both the WL1831

MOD and WL1835 MOD have Bluetooth capabilities. [32] Since WL1835 MOD has the most option of the four it would likely be the primary choice. The best features of the WL1835 MOD are:

- Operates at the 2.4 GHz WIFI band
- Integrated Bluetooth controller (UART)
- Transmitter output power of 17.3 dbm at one DSSS
- Receiver sensitivity of -96.3 dbm at one DSSS
- WIFI : 802.11bgn, MIMO STA, AP, & Wi-Fi Direct Mode SmartConfig

[32]Having both WIFI and Bluetooth options is a real plus for Alfred. It would give the user the option to use a smaller more secure network if distance between access points was not an issue. The next option is the CC2520. The CC2520 is a second generation TI ZigBee/IEEE 802.15.4 RF transceiver. It operates in the 2.4 GHz unlicensed ISM band. This chip is a tough industrial grade chip meaning it is capable of handling an extensive temperature range. It also provides built in support for packet handling, as well as data encryption and authentication. Some of the best features for the CC2520 are:

- Noise immunity
- 103 dB link budget
- 802.15.4 MAC support
- Operates under intense temperatures -40 C to 125C
- Built in security module

[32] The noise immunity feature has a minimum 48 dB adjacent channel rejection. This would help a great deal with any local interference that could possibly be a problem. It's a very durable chip that is tested under more extreme temperatures than most. This would be useful if Alfred was used in a hot industrial setting or a very cold climate. The CC2520 comes with built in security which is needed to keep anyone with a radio transmitter from intercepting and/or hijacking the signal. The final chip option is the CC3200. One of the main features of the series is TI's Internet-on-a-chip solutions. This allows for easy WI-FI development. It is designed to be useable in a broad array of designs making it a very adaptable chip. Some of the best features of the CC3200 are:

- Secure fast connection
- Built in processor that can be used separately from the WI-FI
- Camera input
- I2S audio
- Built in power management

[32] The built in processor, while unneeded, is a nice touch, which could potentially allow us to process the camera on a different chip saving us time and processing power on the main one. The camera input and built in power management will make it even easier to setup and manage. The Internet-on-a-chip feature would make handling the data transfer between the main box and

the computer a much easier process. It also has built in security and a slightly faster connection than the WL1835 MOD.

4.431 Chip Chosen

After comparing mesh networks, WLAN, and WPAN, as well as, Bluetooth, Radio, and WIFI in the hardware sections 4.41-4.43 we have concluded that the best method to transfer data for Alfred is to use a WIFI WLAN to transmit the data to others so you can access it easily wirelessly. We chose this method over the others since radio is lacking when it comes to data transfer as well as suffers range issues. We chose WIFI instead of Bluetooth because while Bluetooth is generally easier to use and has lower power requirements, it also has limited connection options. Furthermore, WIFI has better range from the base, a faster connection, as well as decent security options. A WIFI WLAN will allow Alfred the best mix of range, security, data transfer, speed, and simplicity. It is also already widely used and understood by consumers.

For the hardware we decided to go with the TI CC3200 for our WIFI networking needs. The CC2520 was automatically eliminated since we decided to go with WLAN over a radio mesh network. We chose the CC3200 over the WL1835 MOD because while the ability to have both Bluetooth and WI-FI is nice, it is not needed, would take more time to setup up properly, and could be more difficult for the consumer to understand and setup. The CC3200 also has a slightly better connection. The real deciding factor was the Internet-on-a-chip technology that comes with the cc3200. It will make coding, setting up, and maintaining a 802.11bgn network connection much simpler and easier.

4.432 CC3200 Chip Capabilities

In this section we will go into full detail of the CC3200's features and capabilities. The CC3200 is split up into three main subsystems. The first of the subsystems is the Applications microcontroller subsystem. It contains:

- 80 MHz ARM®Cortex®-M4 core
- Embedded memory with up to 256KB, 32 Channel direct memory access
- Crypto Engine for security
- Eight bit parallel Camera interface
- Multi-Channel Audio Serial Port (McASP) that can support two I2S Channels
- SD/MMC interface
- Two universal asynchronous receivers and transmitters (UARTs)
- Serial Peripheral Interface (SPI)
- Inter-integrated circuit
- Four general purpose timers with 16 bit pulse width modulation mode (PWM)
- One watchdog timer
- Four channel 12 bit analog to digital converters (ADCs)

- 27 individually programmable, multiplexed GPIO Pins

[33] This subsystem would control any applications we chose to run off this chip. It could easily run the bar code scanner since it has the eight bit parallel camera interface. This would free up processing on the other chip. It has a solid amount of GPIO pins and the crypto engine with AES, DES, 3DES, SHA2, MD5, CRC, and Checksum is a nice to have. [33] Four timers plus the watchdog timer should be more than enough to handle any timing functions. An 80 MHZ ARM®Cortex®-M4 core processor should be powerful enough to handle the camera operations. The embedded memory can hold up to 256 KB and includes an external flash Boot loader and Peripheral Drivers in ROM. The next subsystem is the WIFI Network Processor Subsystem. It contains:

- Texas Instrument's WIFI internet on a chip
- Dedicated ARM MCU
- WIFI and internet Protocols in ROMi
- 802.11 b/g/n radio, baseband, medium access control, WIFI driver, and supplicant
- TCP/IP Stack with industry standard BSD socket application programming interface, eight simultaneous TCP or UDP sockets, and two simultaneous TLS and SSL sockets.
- Crypto Engine
- Station, AP, and WIFI Direct Modes
- WPA2 personal and enterprise security
- SimpleLink connection manager
- SmartConfig Technology
- TX Power
    - o 18.0 dBm at 1 DSSS
    - o 14.5 dBm at 54 OFDM
- RX Sensitivity
    - o -95.7 dBm at 1 DSSS
    - o -74 dBm at 54 OFDM

[33] The first thing that's important about the WIFI Networking Processor Subsystem is "WIFI Internet-On-a-Chip". [33] It is part of TI's attempt to make connectivity easier. The basic principle of it is to make the wireless communication part of the process easier for the developer. TI provides a lot of support for the product, including reference designs, code examples and online support. The chip can use 802.11 a, b, or n. Since we were looking to use either 802.11ac or 802.11n this meets that requirement. The crypto engine for security has 256 bit AES encryption which is nice. The WPA2 manager will be useful for making Alfred easy to integrate into personal and business WIFI networks. It comes with SmartConfig technology as well as an AP mode and WPS2. Furthermore, the C3200 can manage WIFI connections on its own with the SimpleLink connection manager. It can also run in station, AP and WIFI Direct Modes which has potential. The next subsystem is the Power Management

Subsystem. It contains:

- Integrated DC-DC
- Advanced low power modes
- Clock source with options of a 40 MHz crystal with an internal oscillator and 32.768 kHz crystal or external T+RTC clock
- Package and operating temperature

[33] This subsystem handles the power. Low power modes are great especially for when Alfred is not in use, which will likely be the majority of the time. These modes include hibernate which uses 4 µA, low power deep sleep (LPDS) which uses 120 µA, RX traffic which uses 59 mA at 54 OFDM, TX traffic which uses 229 mA at 54 OFDM @ maximum power, and idle connected which uses 695 µA at DTIM equals one. The Integrated DC-DC allows for a few options when it comes to supply voltage. It can use V_{BAT} wide voltage mode for voltages in-between 2.1 to 3.6V or preregulated mode which requires 1.85V. The ambient temperature can range from -40 to 85 C which should be able to handle most if not all of Alfred's needs. The low power modes are a big selling for the CC3200 and will be very useful since Alfred will always need to be running for remote access but doesn't need to be running at full power with no active use.

4.5 Sub System 4: Screen I/O Interface

Alfred will need to communicate with the user in an easy-to-use, accurate, and efficient way. Alfred will also need a touchscreen interface that is capable of displaying images back to the user in color. Touch screens can either be resistive, surface capacitive, or projected capacitive. The types of visual displays that we considered were light emitted diode, organic light emitting diode or thin film transistor. In making these decisions, we considered that Alfred's screen will need to be durable, accurate, and have good resolution for a commercial environment.

4.51 Resistive Vs. Capacitive

Resistive touch screens are comprised of multiple layers with two resistive layers in the center separated by a thin space. One layer is powered by a low voltage while the other waits to sense the voltage. When pressure is applied to the screen, the two layers make contact and the sensing panel begins to behave as a pair of voltage dividers moving quickly between both layers of material. The sensors are then able to determine the area where the screen was touched. While resistive touch screens are low in cost and high in durability when it comes to liquids, they risk damage by sharp objects and unnecessary pressure by the user. Resistive screens also have a very low resolution due to the extra layers needed for the screen.

In surface capacitance devices, one side of the insulator is coated with the conductive material. This creates a sheet of resistance across the surface of the device that in part, causes each corner of the device to have a different effective capacitance when touched. The controller then measures the change in capacitance at the four corners and calculates where the device was touched. Surface capacitance screens can become less accurate as the screens become larger. These screens are therefore often seen in basic application use, such as industry controls or store kiosks.

Projected capacitance screens on the other hand are much more accurate when compared to surface capacitance. These capacitive screens are created by etching a grid into a single conductive layer or forming a grid by etching parallel lines on two conductive layers and placing them together so the lines are perpendicular. The sensors are then able to measure the difference in effective capacitance at each intersection on the grid to more accurately determine where the device was touched. Projected capacitive screens offer better contrast and resolution over their resistive and surface capacitive counterparts. Resistive screens are generally less touch sensitive than capacitive screens. This causes the user to have to use less excessive force when operating resistive screens, which in part will allow the unit to last longer. Due to the use of Alfred in a commercial setting, the capacitive screen will allow for a longer lasting product on the job site with a better overall resolution. [11]

4.52 LED, OLED or TFT-LCD

Traditional color screens and monitor light the screen by use of the RGB color model. In this method, red, blue, and green light are added together to create many colors in the light spectrum. With new light emitting diode displays, however, light is able to be produced in any color without the need to blend colors to do so. This allows for much smaller size in display with an increase in resolution of images. LED displays also have a very quick response time when powering the unit on or while processing fast moving visuals without creating a blur effect. Another advantage to having an LED display is that because the diodes emit their own light, there is no need for additional backlighting like other screens. Without need for additional backlights, the LED screen produce less heat and draw less energy than other displays of the same size. Due to the cost of making LED displays however, most models small enough for Alfred's needs only come in the form of segment display. The models with enough pixels to render the pictures and interface necessary for Alfred are generally very costly. Therefore we have determined that LED displays will not be suitable for Alfred as the cost does not justify the advantage.

Organic light emitting diodes are becoming more popular due to the fact that the material is able to be printed directly onto a substrate which brings down production cost. Even though the cost is reduced, the displays are still very expensive to make. OLED's are superior in the fact that they are able to take on a much slimmer design due to the fact they do not need a backlight. Also, they are brighter and consumer much less energy than traditional LCD and LED displays. OLEDs do have a few drawbacks however, these types of displays generally wear out four to five times faster than LCD displays. This is due to the organic molecules degrading more quickly than the non-organic compounds. Organic materials are also sensitive to water which can prove to be very difficult when working with mobile technology. For these reasons, OLED technology seems to be better for stationary domestic products and would not be good for a commercial environment.

Thin-film transistor liquid crystal displays are widely used in most projects as they have low production cost and therefore are easy to find and implement at low cost. The disadvantage to an LCD is the need for a backlight to produce images. LCDs also have a viewing angle that that is much smaller than LED or OLED displays. Response times can also be slower causing bleeding effects in fast moving images. Alfred’s main purpose is data storage and therefore will not need to display fast moving images to accurately communicate with the user. Also, with due to our goals as stated above, we have found that a TFT-LCD screen will suffice for our needs. [12]

4.6 Sub System 5: power

As stated in previous sections, Project Alfred has two major components to the design: the main frame and the appliance box on each door. The power supplied to the main frame or computer acting as the master of the whole building will be provided by the user. Subsequently power for each box used in project Alfred will be incorporated into the lower power device. Considering that the apparatus is a low powered device project Alfred will be using a low DC power. Alfred’s power system is broken into three major categories:

1. The battery management system
2. The Battery or load of the system that will power the devices
3. The source to recharge the battery (in Alfred’s case a wall outlet connected to an AC to DC adapter.)

Even though the team has decided to use products that allow for removal from the wall, the products will still be mounted to the door. However with a rechargeable battery pack provided, the product can be easily transitioned to a movable device if the need arises. Due to it being a prototype, our main goal is getting it to function properly. Therefore, powering is the number one priority, and mobility can be added later after approval of prototype or if deemed an important function for the company using Alfred.

4.61 Battery Management System

A battery management system is any electronic system that manages a rechargeable battery, such as by protecting the battery from operating outside its Safe Operating Area, monitoring its state, calculating secondary data, reporting that data, controlling its environment, authenticating it and / or balancing it. A diagram of how the battery management system is show in the figure below.

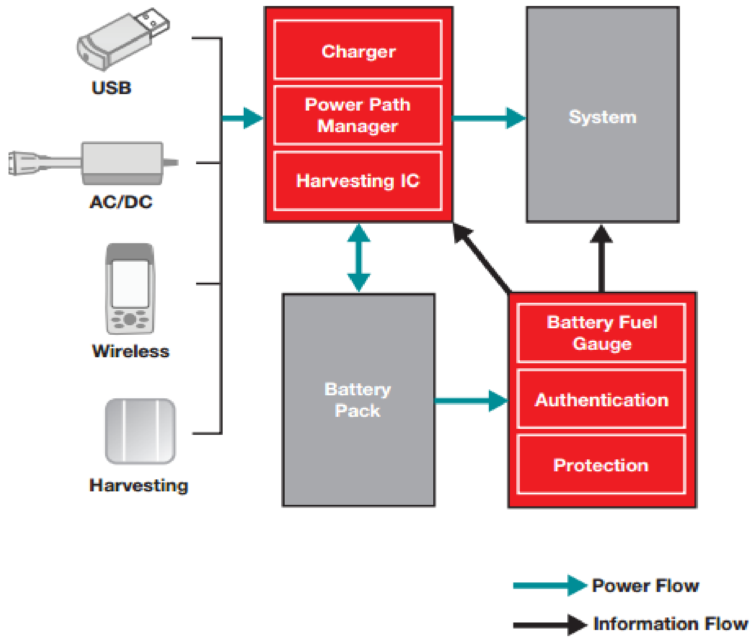


Figure 4.6.1 Power System Diagram [23]

A battery management system comes with 2 major parts those highlighted in red, the charger, power path manager, harvesting IC, and others support the ones in gray. The system in this diagram is all of Alfred's major parts that need to be powered. The battery management system for Alfred is a simple design, not requiring harvesting for solar power, or wireless transfer systems. As a result Alfred will be using a battery charging system. The system will be a Low Voltage DC from an adapter for usable currents below 1 amp. The charger used High-Current, High-Efficiency and Narrow-VDC Chargers for Single-Cell Li-Ion with Power-Path Management and USB OTG. The products found in TI are the bq2419x, bq2429x Families.

4.62 Lithium Ion Battery

A lithium-ion battery is a member of a family of rechargeable battery types in which lithium ions move from the negative electrode to the positive electrode during discharge and back when charging. Li-ion batteries use an intercalated lithium compound as one electrode material, compared to the metallic lithium used in a non-rechargeable lithium battery. The electrolyte which allows for ionic movement and the two electrodes are the consistent components of a lithium-ion cell. The battery project Alfred will be using is a A lithium-ion polymer battery, is a rechargeable battery of lithium-ion technology in a pouch format. Unlike cylindrical and prismatic cells, LiPos come in a soft package or pouch, which makes them lighter but also lack rigidity. The particular battery project Alfred will be using is a single cell Li-Polymer 3.7v 1100mAh 20C (603565) Battery. Features include:

- Single cell Li-Polymer 1100mAh battery.
- 20C continuous discharge rate of 16.5 amps with 22 amp bursts in accordance with our "Real Flight Rating" method.
- Maximum charge rate 1.1 Amps.
- Dimensions: 65mm x 35mm x 6mm
- Weight: 27 grams [23]

In addition, to protect the batter from problems that could occur we will be using a protection circuit module that comes with the battery. This will keep the battery safe from power problems such as: brown outs and black outs.

4.63 AC to DC

An AC adapter, AC/DC adapter, or AC/DC converter is a type of external power supply, often enclosed in a case similar to an AC plug. Adapters for battery-powered equipment may be described as chargers or rechargers. AC adapters are used with electrical devices that require power but do not contain internal components to derive the required voltage and power from mains power. The internal circuitry of an external power supply is very similar to the design that would be used for a built-in or internal supply. Since the team is using a TI designed battery charger it is only fitting to go with TI product for its AC to DC components. The product chosen is the PMP4390 for the ability to convert 120 AC volts to 5 volts DC and compact dimensions. Some features of the product include:

- Fully tested to comply with EMC(CE,RE) and safty requirement
- <30mW no load power over input range
- Valley control PSR controller +S.R controller for high efficiency (avg. 85% @230Vac)
- Good thermal performance with low component count and shell temperature
- Small form factor: 44mmx35mmx15mm
- High Precision CC and CV control with 0.3V cable voltage drop compensation [24]

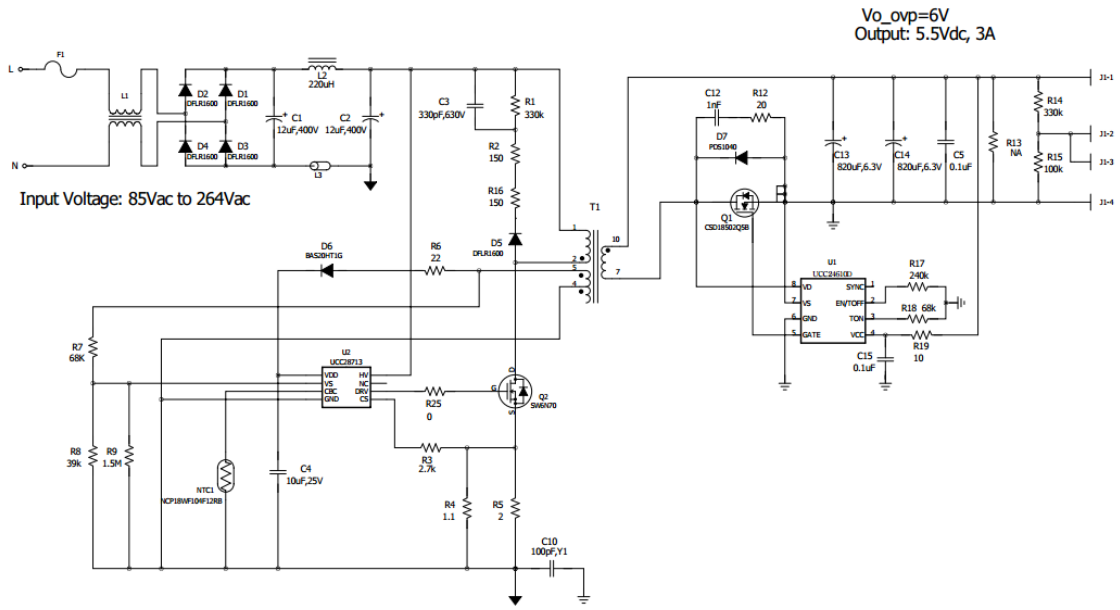


Figure 4.6.2 PMP4390 AC to DC block diagram [25]

With this adapter we can efficiently convert to DC power to send to our power charger to charge our LIPO battery pack. This product is designed to be used outside the box for portable and mobile access for future advancements and later use. Due anti-theft reasons it will be mounted to the wall along with the box if asked and both can be removed if asked. It will not be incorporated in the box.

5.0 Hardware and Software Design Summary

As Stated in previous section project Alfred is broken into 5 major sub systems:

1. Embedded control sub systems
2. Camera input, RFID and cataloging input output sub systems
3. Networking and Communication
4. Screen I/O and user interface
5. Power

Each subsystem has major parts needed to build a working prototype of project Alfred. This section summarizes the parts used in project Alfred and how they are connected together. The team has designed a diagram including all of the individual parts used as well as a key of all the parts bellow.

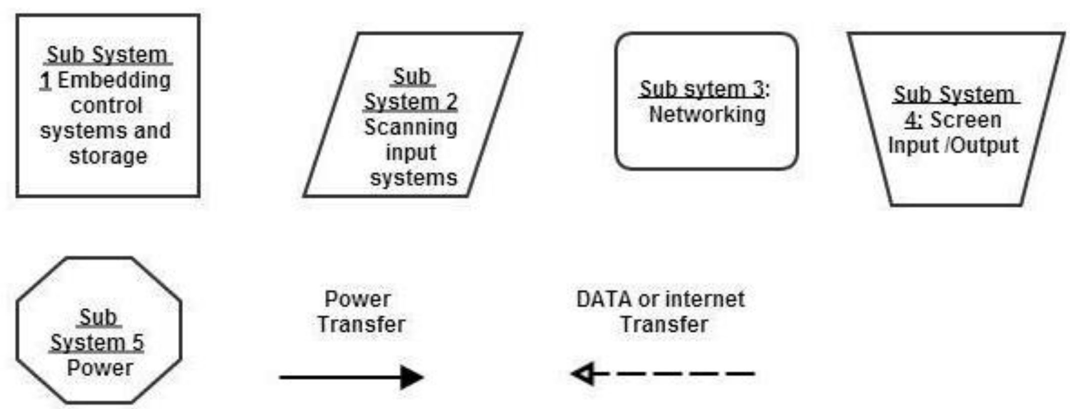


Figure 5.0.1Parts used diagram key

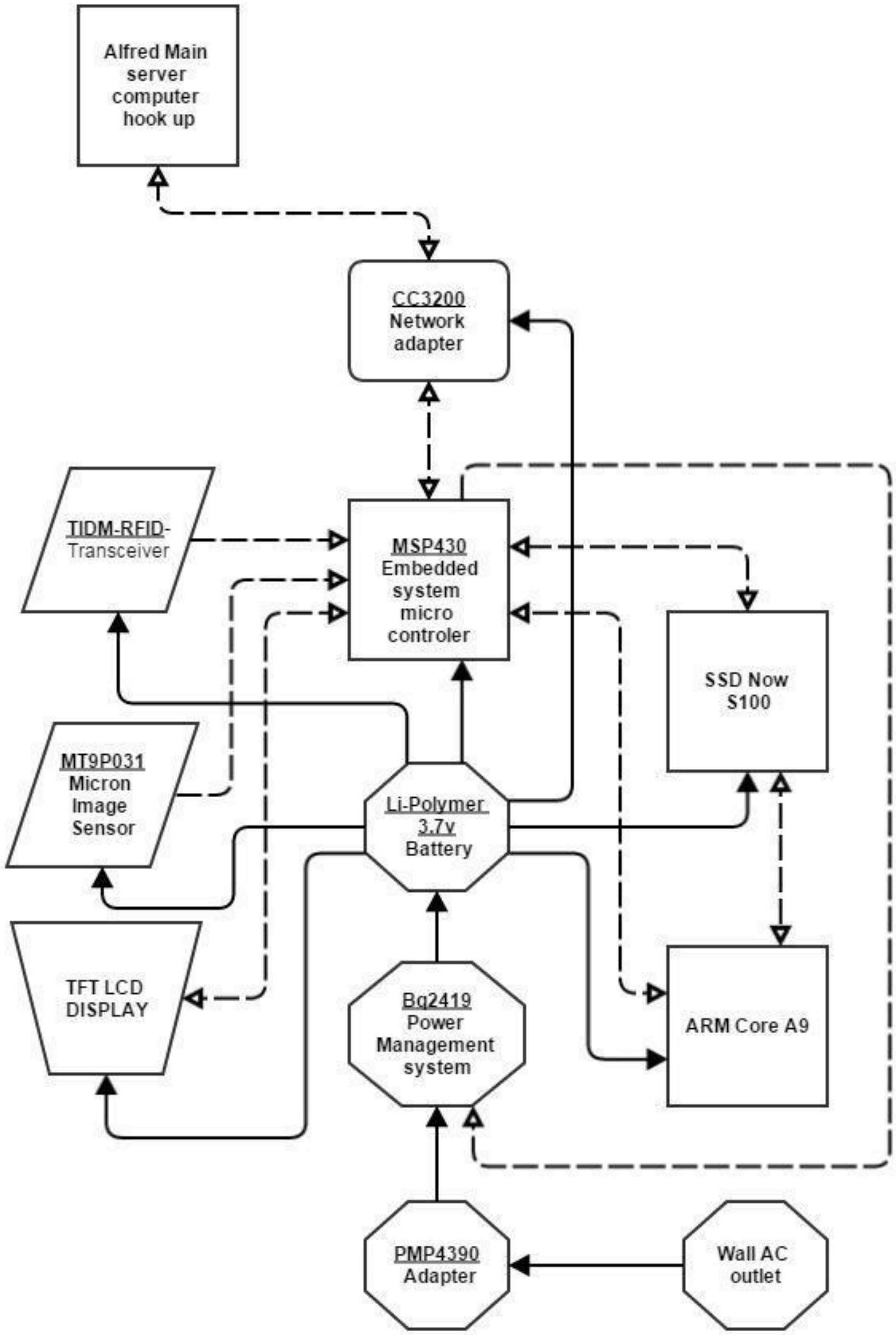


Figure 5.0.2 Parts Used

5.1 Micro controllers

In this section we will look at the internal components and specifications for the microcontroller and processors we have chosen for Alfred. This data will help facilitate proper internal connections when designing our PCB.

5.11 ARM Cortex-A9

In Figure 5.1.1 below, we see the chip diagram for the ARM Cortex A9 processor for a single core. This diagram shows the internal components that make up the Cortex processor we discussed in section 4.2.

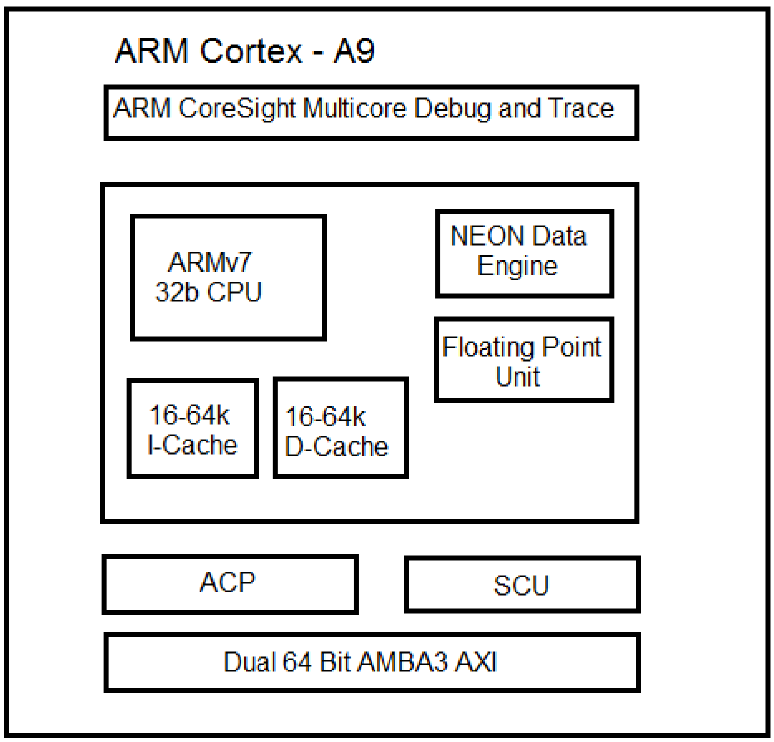


Figure 5.1.1- Chip Diagram for Cortex A9 Processor

The processor specifications for the ARM Cortex A9 processors are as follows: [13]

- ARMv7-A Architecture
- 1-4 Cores
- Memory Management Unit
- 2.5 DMIPS/MHz Dhrystone Performance per core
- Thumb/Thumb-2 Technology
- NEON Media Processing Engine
- Floating-Point Unit
- TrustZone Technology

- Optimized level 1 & 2 Cache

5.12 ARM Cortex-M4

Figure 5.1.2 below, we see the chip diagram of the microcontroller for the ARM Cortex M4.

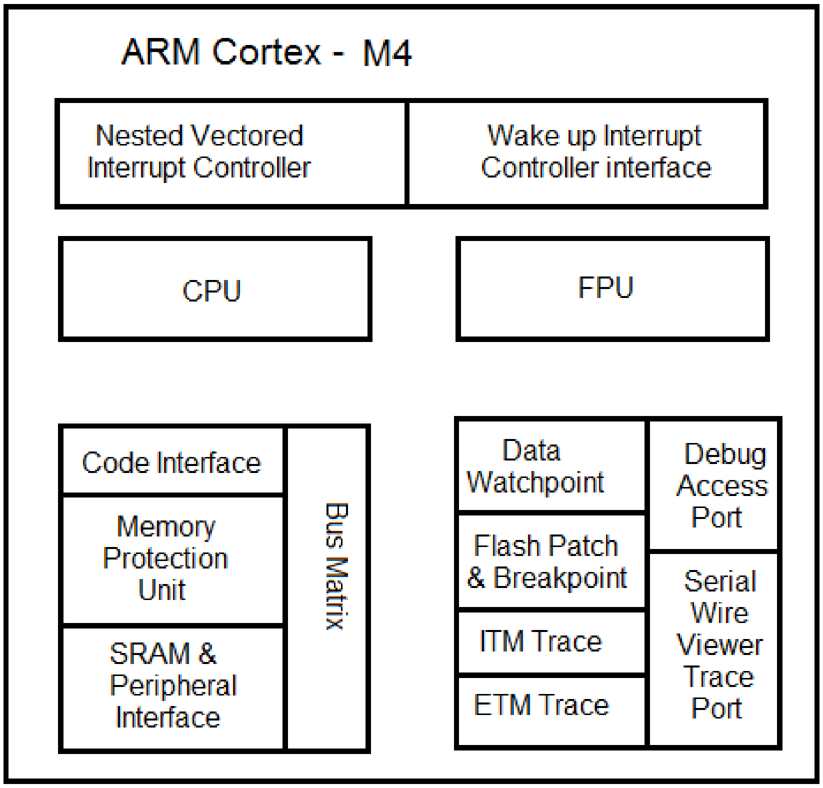


Figure 5.1.2 – Chip Diagram ARM Cortex-M4

Below is a list of the specifications and benefits that are included with the M4 controller. We see the inclusion of the Thumb and Thumb-2 technology that was discussed earlier in the A9 processor section. The M4 Cortex also comes with three sleep modes for lower power consumption when device is idle and waiting for commands.

Microcontroller Specifications [14]

- Single Core
- DSP Extensions
    - Single cycle 16/32-bit MAC
    - Single cycle 16-bit MAC
    - 8/16-bit SIMD arithmetic
    - Hardware Divide (2-12 cycles)
- Memory Protection Unit
- 3.40 CoreMark/MHz

- 1.25/1.52/1.91 DMIPS/MHZ
- Thumb/Thumb-2 Technology
- Single Precision Floating-Point Unit
- 1 to 240 Non-Maskable physical interrupts
- 8-256 Interrupt Priority Levels
- Three Sleep modes
    - WFI and WFE Instructions and Sleep on Exit
    - Sleep and Deep Sleep Signals
    - Retention Mode with ARM Power Management

5.13 MSP 430

In Figure 5.1.3 below, we see the Pinout diagram for the MSP430 G2553, 20-pin devices followed. The pin names and descriptions for how to connect the MSP 430 G2253 follow in Table 5.1.4.

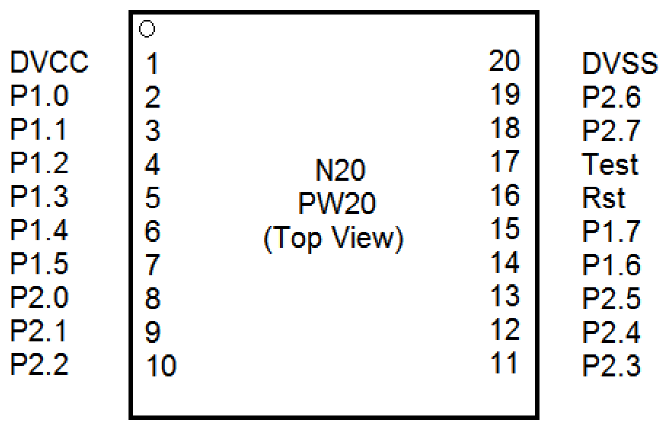


Figure 5.1.3 – Pinout diagram for MSP 430 G2553 device [15]

Name	**Pin**	**Type (I/O)**	**Description**
DVCC	1	-	Digital Supply Voltage
P1.0/TA0CLK/CLK/A0/ CA0	2	I/O	Timer0_A, clock signal TACLK input
P1.1/TA0.0/A1/CA1 UCA0RXD/UCA0SO	3	I/O	Timer0_A, capture: CCI0A input, compare: Out0 output / BSL transmit
P1.2/TA0.1/ A2/CA2	4	I/O	Timer0_A, capture: CCI1A input, compare: Out1 output
P1.3/ADC10CLK/CAO UT/VREF-/VEREF- /A3/CA3	5	I/O	ADC10, conversion clock output
P1.4/SMCLK/UCB0ST E/UCA0CLK/VREF+/V	6	I/O	SMCLK signal output

EREF+/A4/CA4/TCK			
P1.5/TA0.0/ A5/CA5/TMS	7	I/O	Timer0_A, compare: Out0 output / BSL receive
P2.0/TA1.0	8	I/O	Timer1_A, capture: CCI0A input, compare: Out0 output
P2.1/TA1.1	9	I/O	Timer1_A, capture: CCI1A input, compare: Out1 output
P2.2/TA1.1	10	I/O	Timer1_A, capture: CCI1B input, compare: Out1 output
P2.3/TA1.0	11	I/O	Timer1_A, capture: CCI0B input, compare: Out0 output
P2.4/TA1.2	12	I/O	Timer1_A, capture: CCI2A input, compare: Out2 output
P2.5/TA1.2	13	I/O	Timer1_A, capture: CCI2B input, compare: Out2 output
P1.6/TA0.1/ CA6/TDI/TCLK	14	I/O	Timer0_A, compare: Out1 output
P1.7/CAOUT /A7/CA7/TDO/TDI	15	I/O	ADC10 analog input A7
RST/NMI/SBWTDIO	16	I	Reset
TEST/SBWTCK	17	I	Selects test mode for JTAG pins on Port 1
XOUT/P2.7	18	I/O	General-purpose digital I/O pin
XIN/P2.6/TA0.1	19	I/O	General-purpose digital I/O pin
DVSS	20	-	Ground Reference

Table 5.1.4 – Pinout table for MSP 430 G2X53 device [15]

These are the MSP430 specifications:

- Supply-Voltage Range: 1.8 V to 3.6 V
- Ultra-Low Power Consumption
  - Active Mode: 230 µA at 1 MHz, 2.2 V
  - Standby Mode: 0.5 µA
  - Off Mode (RAM Retention): 0.1 µA
- Five Power-Saving Modes
- Ultra-Fast Wake-Up From Standby Mode in Less Than 1 µs
- 16-Bit RISC Architecture, 62.5-ns Instruction Cycle Time
- Basic Clock Module Configurations
  - Internal Frequencies up to 16 MHz With Four Calibrated Frequency
  - Internal Very-Low-Power Low-Frequency (LF) Oscillator
  - 32-kHz Crystal
  - External Digital Clock Source
- Two 16-Bit Timer_A With Three Capture/Compare Registers
- Up to 24 Capacitive-Touch Enabled I/O Pins
- On-Chip Comparator for Analog Signal Compare Function or Slope Analog-to-Digital (A/D) Conversion

- 10-Bit 200-ksps Analog-to-Digital (A/D) Converter With Internal Reference, Sample-and-Hold, and Autoscan
- Brownout Detector
- Serial Onboard Programming, No External Programming Voltage Needed, Programmable Code Protection by Security Fuse
- On-Chip Emulation Logic With Spy-Bi-Wire Interface

5.14 SSD1963 Microcontroller

Alfred's CFAF480272G-043T-CTS display screen also includes a built-in SSD1963 microcontroller with an 8-bit parallel interface. SSD1963 Specifications [16]

- Display features
  - Built-in 1215k byte frame buffer
  - Support TFT 18/24-bit generic RGB interface
  - Hardware rotation 0, 90, 180, 270
  - Programmable brightness, contrast and saturation
  - Backlight Control
- 8/9/16/18/24-bit MCU interface
- 4 GPIO pins
- Built-in clock generator
- Deep sleep mode
- Core supply power: 1.2V ± 0.1V
- I/O supply power: 1.65V – 3.6V
- LCD interface supply power: 1.65V - 3.6V

The SSD 1963 has an MCU interface that can be configured into two modes, 6800 and 8080. The interface connects the MCU to the SSD1963 graphics controller. The controller also comes with a Tearing Effect Signal. This signal goes high when there is to be no display. By doing so, the MCU is able to send data while avoiding tearing in the image. Below, in Figure 5.1.5, we see the internal functional block diagram for the SSD graphics controller.

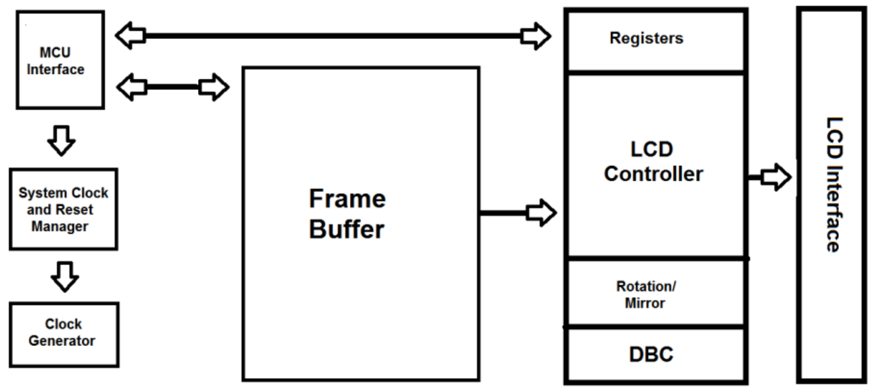


Figure 5.1.5- SSD1963 Bock Diagram [16]

5.2 Camera and RFID Inputs

The image sensor has 48 different pins classified into three main types: input for data, output for data, and supply or power for the image sensor. The Pin Description sheet provided in figure 4-… provides the functions for each individual pin in the 48 Pinout diagram.

Name	LLCC PIN	Type	Description
Reset #	16	Input	When Low, the Mtp9p031 asynchronously resets
Ext Clck	31	Input	External Input Clock
SCLK	4	Input	Serial clock: Pull to VDDQ with a 1.5k resistor
OE#	17	Input	When ender a hiz to write to frames, low normal
STANDBY#	14	Input	Standby
TRIGGER	15	Input	Snap shot trigger
SADOR	13	Input	Serial Address
SDATA	5	I/O	Serial Data
PIXCLK	32	Output	Pixel Clock
DOUT[11	33 to 45	Output	Pixel Data

:0]			
Frame_Valid	7	Output	Frane Valid activate pixels horizontal
line_Valid	8	Output	Activate pixels in each line
Strobe	9	Output	Snape shot strobe
VDD	12, 7	Supply	Digital Supply voltage
VDDQ	11, 39	Supply	I/o Supply voltage
DGND	10, 26, 46	Supply	Digital ground
VAA	23, 24	Supply	analog supply voltage
VAAPIX	1, 48	Supply	Pixel supply voltage
AGND	2, 22	Supply	analog to ground
VDPLL	25	Supply	Pll Supply voltage
Test	3, 20 ,21	none	factory use
RSVD	6	none	factory use
NC	18, 19, 27, 28, 29, 30	none	No connects

Table 5.2.1- Pin Descriptions for 48 pin diagram. [19]

The pin description table shows the normal data and power needed when plugging into each pin. The manufacture recommends around 1.8 to 2.8 volts supplied to inputs and that any wear between 10 micro amps to 72 mil amps will be provides or inputted by or into each pin.. Power consumption of the image sensor is set around 381 mW at full resolution while doing the most power intense mode of streaming. To make sure we don't cause stress to the or permanently damage the device we need to keep the voltage between 3.1 and -.3 volts along with a max 100 mA and a temperature of -30 to 70 degree Celsius.

Camera Software applications:

The main function of this camera is to provide the latest barcode and QR code scanning application software for the user to maintain and keep inventory. The user activates the camera function on Alfred and selects the desired scanning process it would like to do and hold up the sticker code to the camera. The camera will scan in the product and prompt the user to give extra information such as storage and quantity of the item. Further explanation on the QR code and the Barcode software can be found in 4.32 and 4.33.

The RFID Transceiver is the TIMD RFID by Texas Instruments. The RFID transceiver works with many of the msp430 and processor technology. As a result it will be a kit bought separately and connected separately. Below shows the schematic block diagram of the items provided by TI to be hooked up to the main board.

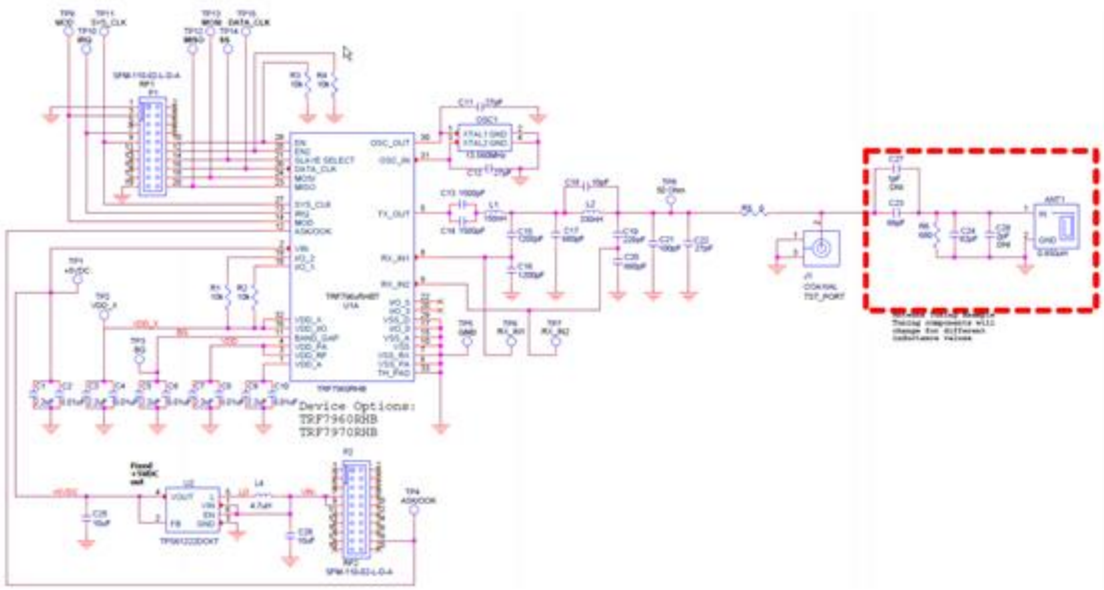


Figure 5.2.2 - TIDM RFID Schematic Diagram [22]

The RFID Transceiver comes with an input/output plug and allows for simplistic design attributes for plug and play with a board. Therefore, the product will be integrated into the main design of the board.

5.3 Network

For our network, we will be doing a WLAN network using the Texas Instruments CC3200 networking chip. The chip will be attached to the other subsystems using the pins located on it. The following section will go in depth on the specifications required to use the CC3200, such as the power requirements, current needs and voltage maximums and minimums.

5.31 Chip Specifications

The following tables contain the absolute maximum ratings for the CC3200. They are done assuming the device is operating in a free air temperature range. All measurements are taken at the device pins.

Parameters	Pins	Min	Max	Unit
V_{BAT} and V_{IO}	37, 39,44	-0.5	3.8	V
V_{IO}-V_{BAT}	10,54		0	V

Figure 5.3.1 data taken from TI Specifications [33]

Parameters	Min	Max	Unit
Digital Inputs	-0.5	V_{IO}+.5	V
RF pins	-0.5	2.1	V
Analog pins	-0.5	2.1	V
Op temp range	-40	85	C

Figure 5.3.2 data taken from TI Specifications [33]

Since V_{BAT} and V_{IO} have a max input of 3.8 V we will have to stick to that. We will likely use the since input for both in order to keep V_{IO}-V_{BAT} at zero. The maximum operating temperature is 85 degrees Celsius which is 185 degrees Fahrenheit. The next table contains the handling ratings for the CC3200.

Parameters	Min	Max	Unit
T_{STG}	-55	125	C
V_{ESD} (HBM)	-2000	2000	V
V_{ESD} (CDM)	-500	500	V

Figure 5.3.3 data taken from TI Specifications [33]

V_{ESD} (HBM) and V_{ESD} (CDM) stand for Human Body Model and Charged Device Model respectively. The storage temperature range is from -55 to 155 degrees C which is -67 to 311 degrees Fahrenheit. That really should not be a problem for Alfred since 99% of Alfred's storage will likely take place inside of buildings. The next table contains the Power on Hours specifications for the CC3200.

Conditions	POH
$T_{Ambient}$	17500

Figure 5.3.4 data taken from TI Specifications [33]

So assuming that the temperature remains at ambient levels, the CC3200 is spec'd to last for 17500 hours or about two years. However, TI notes that the CC3200 can be run reliably for ten years. I would expect to be able to get at least four years out of the device assuming no catastrophic failures. The next table is the recommended operating conditions.

Parameters	Pins	Conditions	Min	TYP	Max	Unit
V_{BAT}, V_{IO} (shorted V_{BAT})	10, 37, 39, 44,	Direct battery connection	2.1	3.3	3.6	V

	54					
V_{BAT}, V_{IO} (shorted V_{BAT})	10, 37, 39, 44, 55	Preregulated 1.85V	1.76	1.85	1.9	V
Ambient thermal slew			-20		20	C/min

Figure 5.3.5 data taken from TI Specifications [33]

This shows the min and max VIO assuming the VBAT connection is shorted and the CC3200 is running on either a battery or at a preregulated voltage of 1.85V. It should be noted that the operating temperature is limited by the variation of the frequency for the crystal. For the best performance under direct battery power conditions, the ripple on the 2.1 to 3.6 V connection needs to be no more than 300 and no less than -300 mV. For the best performance under preregulated conditions, the ripple on the 1.85V needs to be less than two percent. The following table contains the electrical characteristics for the cc3200 GPIO Pins except 29, 30, 45, 50, 52, and 53 at 25 C assuming 3.3 V at 25 degrees Celsius.

Parameters	Min	NOM	Max	Unit
C_{IN}		4		pF
V_{IH}	.65*VD		VDD + .5V	V
V_{IL}	-0.5		.35*VDD	V
I_{IH}		5		nA
I_{IL}		5		nA
V_{OH}	2.4			V
V_{OL}			0.4	V
I_{OH}	2			mA
	4			mA
	6			mA
I_{OL}	2			mA
	4			mA
	6			mA

Figure 5.3.6 data taken from TI Specifications [33]

C_{IN} is the Pin capacitance that has a NOM value of four Pico farads. VIH is the high level input voltage. This shouldn't be any higher than half a volt above VDD or sixty five percent below VDD. V_{IL} is the low level input voltage which should not fall below negative half a volt or go above thirty five percent of VDD. I_{IH} is the high level input current which has a NOM value of five nanoamps. Thus I_{IL} is the low level input current which has the exact same value as the high level input current. V_{OH} is the high level output voltage assuming VDD is three volts. The minimum high level output is 2.4 volts which means you would lose .6 volts. V_{OL} is the low level output, assuming VDD is three volts; the maximum output is four tenths of a volt. I_{OH} is the high level source current assuming V_{OH} is at its minimum of 2.4 V. Thus I_{OL} is the low level sink current assuming VOH is .4 V. Both have a 2, 4, and 6 mA drive which run at 2, 4 and 6 mA. It is recommended to run at the lowest possible drive level to reduce interference with the WLAN radio, and reduce degradation to RF sensitivity and performance. The next table contains the parameters for the GPIO Pins 29, 30, 45, 50, 52, and 53 at 25 C.

Parameters	Min	NOM	Max	Unit
C_{IN}		7		pF
V_{IH}	.65*VD		VDD + .5V	V
V_{IL}	-0.5		.35*VDD	V
I_{IH}		50		nA
I_{IL}		50		nA
V_{OH}	2.4			V
V_{OL}			0.4	V
I_{OH}	1.5			mA
	2.5			mA
	3.5			mA
I_{OL}	1.5			mA
	2.5			mA
	3.5			mA

Figure 5.3.7 data taken from TI Specifications [33]

These pins are basically the same as the last ones except, the capacitance at each pin has a NOM of seven Pico farads. The high and low level input currents NOM value is increased to 50 nano Amps. Finaly the 2, 4 and 6 mA Drives for the High level source current and low level sink are 1.5, 2.5, and 3.5 mA instead of 2, 4, and 6 mA. The next table contains the pin internal pullup and pull down specs.

Parameters	Min	NOM	Max	Unit
I_{OH} Pull-up current, (V_{OH} = 2.4 VDD = 3V)	5		10	uA
I_{OL} Pull-Down current, (V_{OH} = .4 VDD = 3V)	5			uA

Figure 5.3.8 data taken from TI Specifications [33]

This shows that the I_{OH} Pull-up current has a min of five and a max of ten uA, while the I_{OL} Pull-Down current has just a min of five uA. The next table contains the parameters and characteristics for the WLAN receiver.

Parameters	Conditions(Mbps)	Min	TYP	Max	Unit
Sensitivity	1 DSSS		-95.7		dBM
	2 DSSS		-93.6		dBM
	11 CCK		-88		dBM
	6 OFDM		-90		dBM
	9 OFDM		-89		dBM
	18 OFDM		-86		dBM
	36 OFDM		-80.5		dBM
	54 OFDM		-74		dBM
	MCS0		-89		dBM
	MCS7		-71		dBM
Maximum input level	802.11b		-4		dBM
	802.11g		-10		dBM

Figure 5.3.9 data taken from TI Specifications [33]

This shows the typical sensitivity of the reciever based on different DSSS, OFD<, CCK, MCS, and at max input on 802.11 b / 802.11 g. overall it maintains a low dBm no greater than -70 for all except the max inputs. Those increase all the way up to -4 dBm. The next table contains the WLAN transmitter parameters and characteristics.

Parameters	Condition s	Min	TYP	Max	Unit
Sensitivity	1 DSSS		18		dBM
	2 DSSS		18		dBM
	11 CCK		18.3		dBM
	6 OFDM		17.3		dBM
	9 OFDM		17.3		dBM
	18 OFDM		17		dBM
	36 OFDM		16		dBM
	54 OFDM		14.5		dBM
	MCS7		13		dBM
Transmit center frequency accuracy		-25		25	ppm

Figure 5.3.10 data taken from TI Specifications [33]

The sensitivity of the transmitter is between 17-18 dBm for all except 36 OFDM, 54 OFDM, and MCS7. These TYP measurements fall down to 16, 14.5, and 13 respectively. The transmitter accuracy can vary between -25 and 25 ppm. The next table deals with the MCU current consumption.

Parameters		Test Conditions			Min	TYP	Max	Unit
MCU Active	NWP Active	TX	1 DSSS	TX power level = 0		278		mA
				TX power level = 4		194		mA
			6 OFDM	TX power level = 0		254		mA
				TX power level = 4		185		mA
			54 OFDM	TX power level = 0		229		mA
				TX power level = 4		166		mA
		RX	1 DSSS			59		mA
			54 OFDM			59		mA
	NWP idle connected					15.3		mA

Figure 5.3.11 data taken from TI Specifications [33]

Based on the table above, the typical current consumption for the transmitter, while the MCU and NWP are active, is highest the lower the power level. The current consumption also increases for lower OFDMs and is highest when using 1 DSSS. The current consumption by the receiver is significantly lower, and consumes the same amount for OFDM and DSSS. When the NWP is idle the current consumption drops to its lowest at 15 mA. The next table is the current consumption when the MCU is active.

Parameters		Test Conditions			Min	TYP	Max	Unit
MCU Sleep	NWP Active	TX	1 DSSS	TX power level = 0		278		mA
				TX power level = 4		194		mA
			6 OFDM	TX power level = 0		254		mA
				TX power level = 4		185		mA
			54 OFDM	TX power level = 0		229		mA

				TX power level = 4		166		mA
		RX	1 DSSS			59		mA
			54 OFDM			59		mA
	NWP idle connected					15.3		mA

Figure 5.3.12 data taken from TI Specifications [33]

The typical current consumption for the transmitter, while the MCU is in sleep mode and NWP is active, is the highest when TX is at lower the power levels. The current consumption also increases for lower OFDMs and is highest when using 1 DSSS. The current consumption by the receiver is significantly lower at 59 mA, and consumes the same amount for OFDM and DSSS. When the NWP is idle the current consumption drops to its lowest at 15.3 mA. The next table is the current consumption when the MCU is active.

Parameters		Test Conditions			Min	TYP	Max	Unit
MCU LPDS	NWP Active	TX	1 DSSS	TX power level = 0		272		mA
				TX power level = 4		188		mA
			6 OFDM	TX power level = 0		248		mA
				TX power level = 4		179		mA
			54 OFDM	TX power level = 0		223		mA
				TX power level = 4		160		mA
		RX	1 DSSS			53		mA
			54 OFDM			53		mA
	NWP LPDS					0.12		mA
	NWP idle connected					0.695		mA

Figure 5.3.13 data taken from TI Specifications [33]

The typical current consumption for the transmitter, while the MCU is in LPDS mode and NWP is active mode, is the highest when TX is at lower the power levels. The current consumption also increases for lower OFDMs and is highest when using 1 DSSS. The current consumption by the receiver is significantly lower, and consumes the same amount for OFDM and DSSS. When the NWP is idle the current consumption drops to its second lowest at .695 mA. NWP LPDS mode brings the current consumption down to .12 mA. The next table is the current consumption when the MCU is active.

Parameters	Test Conditions	Min	TYP	Max	Unit
MCU Hibernate	NWP hibernate		4		uA

Peak calibration current	V_{BAT} = 3.3 V				450		mA
	V_{BAT} = 2.1 V				670		mA
	V_{BAT} = 1.85 V				700		mA

Figure 5.3.14 data taken from TI Specifications [33]

The typical current consumption for the transmitter, while the MCU is in hibernate mode and NWP is hibernate mode, is typically only 4 uA. The peak calibration current is inversely proportional to V_{BAT}, reaching a typical value of 700 mA when V_{BAT} is equal to 1.85V and a low of 450 mA at V_{BAT} is equal to 3.3V.

5.4 Screen I/O

Alfred will come equipped with a CFAF480272G-043T-CTS display screen from Crystalfontz America, INC. This screen utilizes a resolution of 480xRGBx272 for full color graphics on the TFT display. TFT was chosen as it is a liquid-crystal display variant that uses thin-film transistors to improve contrast and addressability in the image quality. This will allow Alfred to accurately render full colored pictures of any item being stored and tracked by the user. The display also comes with an LED backlight which is needed on LCD screens as discussed earlier. The overall module for the TFT will be 4.15” x 2.64” x 0.31” with a viewing area of 3.81” x 2.20” for the user interface and a 4.3” diagonal. [16]

Item	Symbol	Values			Unit
		Min	Typ	Max	
Operating Voltage	VDD	3.1	3.3	3.5	V
Input High Voltage	VIH	0.8*VDD	-	VDD	V
Input Low Voltage	VIL	0	-	0.2*VDD	V
Output High Voltage	VOH	VDD-0.3		VDD	V
Output Low Voltage	VOL	0	-	0.3	V
Current Consumption	IVCI	-	245	-	mA
Power Consumption	PLCD	-	808.5	-	mA
Power Supply Voltage	VDD	-0.5	-	5.0	V
Input Signal Voltage	LI	-0.5	-	5.0	V
Operating Temp	Topa	-20	-	70	C
Storage Temp	Tst	-30	-	80	C
Power supply for Driver	Vdd-Vss	-0.3	-	3.6	V
Input Voltage	VIN	-0.3	-	IOVCC+0.3	V

Table 5.4.1- Electrical Characteristics for LCD Interface

As show in Table 5.4.1 above, the unit may operate in temperatures between -4 and 158 degrees Fahrenheit. The electrical specifications for supply voltage are between 3V and 3.6V with a typical supply voltage of 3.3V. The TFT module will also run on a typical supply current of 213mA with a 1mm connector pitch. In order to connect the pins of the device properly to the controller, the pin names and function have been provided in Table 5.4.2 below.

PIN NO.	SYMBOL	FUNCTION
1	GND	Ground
2	VDD	Power Supply for Logic
3	B/L Enable	Backlight Control (H: ON L:OFF)
4	RS	Command/Data select
5	WR	8080 family MPU interface: Write
6	RD	8080 family MPU interface: Read
7	DB0	Data Bus
8	DB1	
9	DB2	
10	DB3	
11	DB4	
12	DB5	
13	DB6	
14	DB7	
15	CS	Chip select
16	RES	Reset
17	NC	No connection
18	NC	No connection
19	Disp ON	Display on
20	NC	No connection

Table 5.4.2 - Pin connection from display to control board [16]

The functional block diagram in figure 5.4.3 below shows the connections of display screen we have chosen. There are two components that make up the screen, the TFT LCD panel and the capacitive panel. Both components have an interface that connects into the SSD 1963 microcontroller. This controller then uses an 8 bit 8080 interface to communicate with the rest of Alfred's components. As seen below in the diagram, information is flowing bi-directionally between each block connection.

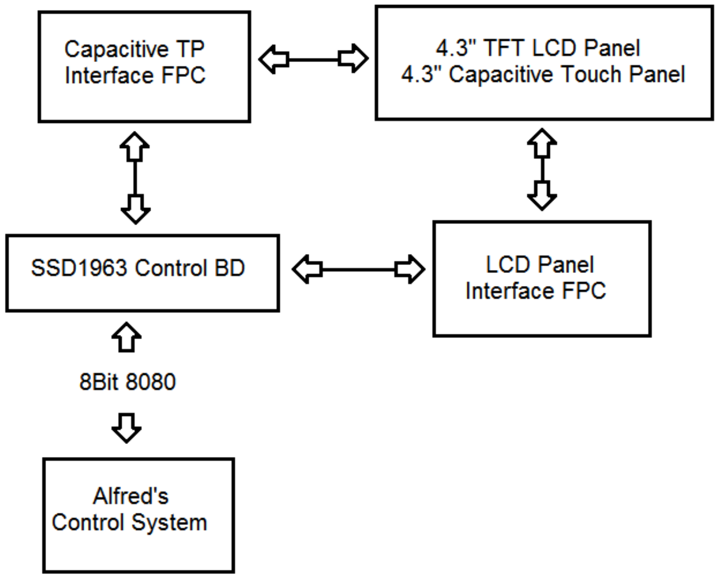


Figure 5.4.3 – Block diagram for TFT-LCD display [16]

5.5 Power

The Bq2419 is a power management system that allows for rechargeable batteries and an ac-dc adapter to be connected and power the product at the same time. Below is a diagram of the management system

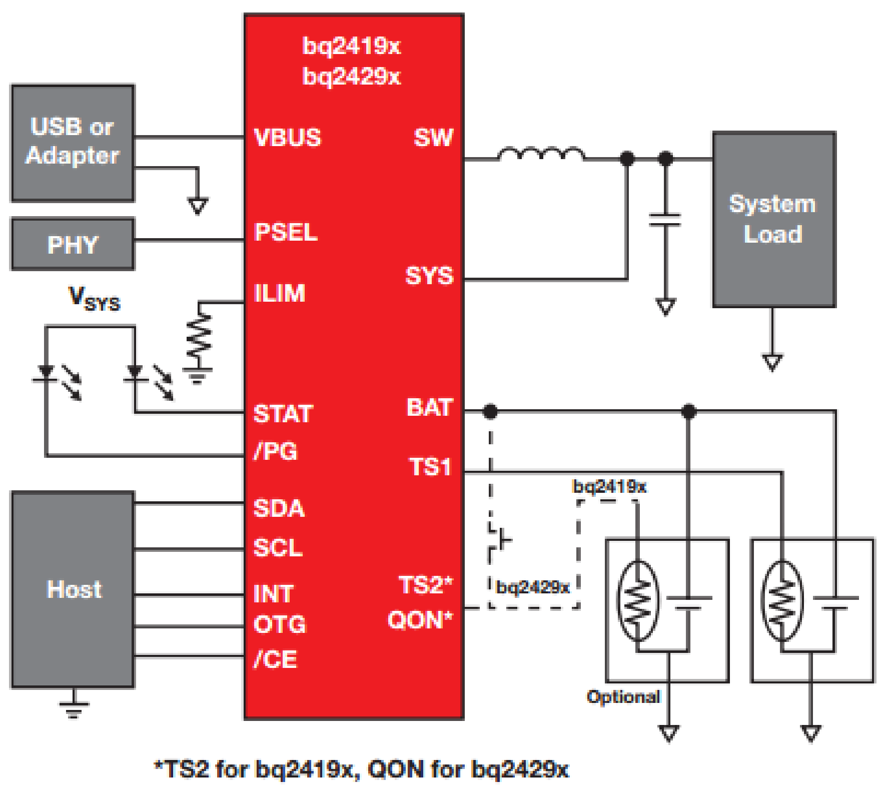


Figure 5.5.1 bq2419 and 2429x battery charger design. [23]

This charging system is ideal for tablets and fast charging applications requiring system power, which is perfect for powering Alfred's hardware. The features provides by TI of this management product are:

- NVDC-1 system with power-path management, system instant-on with no battery or deeply discharged battery
- I 2C host control or autonomous charging with default parameters
- High integration including battery FET, AC switching FETS, current sensing and compensation
- Input-voltage range: bq2419x: 3.9 to17 Vbq2429x: 5 V
- USB-compliant 2.5/4.5-A charger with 1.5-MHz switching mode and D+/D– detection or PSEL
- High charging efficiency: bq2419x: 92% at 2.5 A and 90% at 4 A bq2429x: 88% at 2 A
- High USB OTG efficiency: 90% at 1 A
- Programmable thermal-regulation
- 4 x 4-mm QFN package, pin-to-pin compatible [23]

With this design the product can use a wide selection of batteries, as well as, a wide selection of USB power supplies or AC/DC converters. With A lipo battery used in section 4.62 and a AC/DC converter from 4.63 it makes an efficient power team. The AC/DC adapter is not part of the design it is a separate part of the prototype to allow from removal from the wall.

6.0 Prototype Construction and Coding

Inside the prototype the many elements are connected together through a printed Circuit board. Along with the hardware construction the software is designed by combining different source codes. The next section provides a brief description of design work on the main board and software used.

6.1 Parts Acquisition and BOM

Project Alfred will come with multiple different parts and PCB boards. Some of the products such as the RFID transmitter come with a board and will hook up to the PCB produced by the team. For information on those parts pricing see section 8.2 for cost summary and 5.0 for design summary. The main board will be a double layer with the main chip products such as the network adapter and processor on the main board design. Most of the parts will be bought primarily from the companies which provide, the PCB vendor will be 4 PCB. Since it will be a double sided layer and probably be be between 5 to 10 square inches it will be 35 dollars for 4 board designs. For products parts that go on the boards cost, see section 8.2 for an individual breakdown of each parts cost, number, and manufacturer.

6.2 PCB Vendor and Assembly

A printed circuit board or PCB mechanically supports and electrically connects electronic components using conductive tracks, pads and other features etched from copper sheets laminated onto a non-conductive substrate. PCBs can be single sided (one copper layer), double sided (two copper layers) or multi-layer. PCB's require the additional design effort to lay out the circuit but manufacturing and assembly can be automated. Manufacturing circuits with PCBs is cheaper and faster than with other wiring methods as components are mounted and wired with one single part. Furthermore, operator wiring errors are eliminated. In order, to interconnect all of the part's, chips, and other circuit boards used in project Alfred a PCB has been designed for simpler replication. The team as accomplished the design by using a design program known as eagle CAD software.

6.21 Eagle CAD PCB Design Program

Eagle CAD PCB software is a free to use program that helps with the design and creation of PCB boards for use by manufactures. Eagle Light edition, works by creating a 4 X 3.2 inch board that allows for parts to be aligned and interconnected to each other. Can only use two signal layers with the top and bottom and with limited creation space not everything will be on the boards, but inputs can be placed on the board to hardwire other parts that do not come as separate chips. As a result we can create a design to send to the manufacture where we can create cheap boards that allow all our parts to be connected together without problems.

6.22 PCB Vendor

4PCB will be used when finalizing and creating the PCB boards for project Alfred. They are a reliable company offering free build and design products that allow us to paste our designs from Eagle CAD and get a quote for our products. In addition, to providing a custom design board we send, they will send an additional 3 boards of the same exact design. As a result, we can use the extra boards if we make an error in connecting devices, break the board with soldering mistakes, or the board is flawed design.

6.3 Final Coding Plan

Each section of Alfred must be carefully coded for a quality end result. In this section we will go in depth with how we plan to tackle each segment of code in each of the controllers for the purpose of our major subsystems. Unless otherwise specified, we plan to code Alfred in basic C language as it is the language we are most familiar and comfortable writing.

6.31 Main Screen

The main screen is the jumping off point for the user when interacting with Alfred. From this screen the user will be able to access the different features in order to store or view information. The main screen code will therefore be the root of Alfred's design. Upon start of the device, after boot procedures, the code will wait in this section until the user chooses an option on the screen. The options include the barcode scanner, manual input, the search function, view stored items and wireless communications manager.

6.32 Camera

The camera will have two uses in the project. The first use is as a barcode scanner and the second is to take photos for a visual catalog. We will therefore split the cameras code into two subroutines to deal with each case.

6.321 Barcode Scanner

The barcode scanner code will be accessed when the virtual button on the main display is selected. Once open, we will set up the display of the cameras input to the screen. We will program a red horizontal line in the center of the image. This line will have the purpose of helping the user line up the barcode for a proper scan. The camera will then activate the autofocus feature for better resolution in the barcode imaging. When the image has reached focus, the program will attempt to read the barcode. In the event no barcode was read, the program will jump back to autofocus to wait for the input section again. When a barcode is successfully read, the program will then move into the decryption phase. In this phase, the barcode is decoded by the microcontroller. The controller will then fetch the related data that matches the barcode from memory. If the program fails to find matching data, the program will jump to manual entry of the storage code. Upon success, the program will store the related information of the barcode in registers and pass the information to the screen for output. After information is input, the program will jump back to the beginning of the subroutine to allow for more barcodes to be input. The program will exit this routine upon the user selecting the 'home' button on screen.

6.322 Picture Catalog

The picture catalog function of the camera will be accessed when the program recognizes that the user selected the button in the data entry screen. Similar to the barcode scanner, the program will take the input from the camera and display it to the screen. One difference is that the red line will not be included in this portion. The camera will activate the autofocus feature, once focused the program will wait until the 'take photo' button is selected by the user. When this button is selected, the program will capture the frame that is currently being stored in the output buffer. The function will then display the picture to screen and allow for the user to take a new photo or keep the existing one. Taking a new photo will cause the program to jump back to the beginning of the subroutine to repeat the process. If the user decides to keep the photo, the information will be passed to storage. The program will exit automatically when a photo is accepted or manually if the user chooses the 'home' button on screen.

6.33 Search Function

The search function will be crucial in finding information on Alfred. This function will be accessed when selected by the user from the main screen. When called, the user will be taken to a search screen where they will be prompted with blank category bars. There will also be a selection which allows the user to search selected devices, all devices, or just the one device at which they are present. When the user selects a category bar, the virtual keyboard will appear. The program will then take the input from the user, display the input on the screen, and store the string of characters in an array. The program will then wait for either more search criteria or for the 'search' button to be pressed. When the search button is pressed, the program will then begin comparing the search criteria with stored information in the locations specified by the user. If no matches are found, a message will appear on screen informing the user. The user will then have to option to refine the search or exit back to home. Upon finding a match, the program will display the stored information to the user. The user will have the option to begin a new search or edit this entry. This program will exit if the 'home' button is pressed at any time.

6.34 Data Storage

Storing data will be integral in Alfred's mechanics so that data can be found easily by programs such as the search function. The item data will be stored in a multidimensional array. These arrays will house the information the items are cataloged by. The array will be set up in the form [N, X], where each value of N is a new item being stored and each value of X corresponds to a new category of the description. Table 5.3.1 below shows an example of the spread of information.

	Barcode #	Name	Description	Location
1st item	[0,0]	[0,1]	[0,2]	[0,3]
2nd item	[1,0]	[1,1]	[1,2]	[1,3]

3rd item	[2,0]	[2,1]	[2,2]	[2,3]
4th item	[3,0]	[3,1]	[3,2]	[3,3]

Table 5.3.1 – Data storage array

6.35 Manual Input

The manual input portion will be accessed when selected by the user from the main screen or when accessed by the barcode scanning application. If accessed by the barcode scanner, some information may be filled for the user if the item has been previously scanned, else the user must input any required information. The program will prompt the user for the information needed. The program will require the name, room location and quantity sections to be filled in. Categories such as description and specific placement in room may be skipped by the user. Once the user has entered in the appropriate data, the program will prompt the user to choose to add another entry or exit to main screen. During manual input, the user may choose to edit existing information that is already in storage. This includes adding more details, deleting existing details, updating the quantity, or deleting an entry all together. Upon an edit, or deletion, the system will prompt the user to confirm the details are correct before saving the information to memory. This will add an extra layer of protecting from including/deleting information accidently.

7.0 Project Prototype Testing

This section contains the hardware and software testing procedures for Project Alfred. It will go into detail on what kind of environment the hardware is expected to be used in and how we will simulate that environment. Next it will cover the basic testing procedure for the hardware. This will be followed by specific testing required each individual piece of hardware as well as the hardware assembled together. It will also cover the software testing environment and the procedures for specific software testing.

7.1 Hardware Testing Environment

Project Alfred the Building Master is expected and designed to be operated indoors. This eliminates the need for any testing under severe weather conditions such as rain. Project Alfred is expected to be used in a general indoor setting likely with air conditioning to maintain a stable temperature. This, however, does not mean that Alfred and its associated hardware should not be tested at more strenuous temperatures. Many businesses in different cities, states, countries, and continents could have higher humidity, higher average temperatures, or even sub zero average temperatures. Air can conditioners break down for long periods of time. Heaters can run out of gas supplies in the winter. Alfred needs to be able to maintain top performance levels despite such strenuous conditions. It needs to be able to function when everything else is failing. Alfred will be the constant in a chaotic world, a butler who is always ready to serve.

7.2 Hardware Testing

Each piece of hardware will go through individual and group testing. All hardware will need to pass a primary visual test. To pass this test each part must at the very minimum look to be in working order, have no missing pieces, and no dents or scratches. After passing the preliminary test, a more in depth review of the part will begin. It will check to see that:

- All pins are not bent, broken or missing
- Each pin is functional
- It turns on
- Stays on without any issues
- Doesn't overheat
- Can receive and send data (where applicable)
- It turns off properly

All these small tests are basic requirements to make sure we won't have any catastrophic issues that will require us to send a part back later on that could have been caught from the very beginning. Next each part will be tested on a breadboard one at a time to be sure they function. Then each part will slowly be tested in unison until it can be determined that all parts can function with each other without any issues. All chips will run a set of timing tests to check their individual speeds versus their specs. After passing these tests, each part will then be tested according to their individual purposes.

7.21 Specific Hardware Testing

After passing the basic preliminary tests, each piece of hardware will then undergo individual part specific testing. The individual tests will pertain to each specific part purpose and uses. Some parts will undertake different or more strenuous tests. The following subsections will detail the planed tests for each of the major subsystem hardware.

7.211 CC3200 Testing

The first part that will be tested is the CC3200. We will need to be tested to make sure that its WI-FI is functioning properly. We will also need to check to see that the 802.11 protocols are installed. Furthermore the camera input and processor needs to be checked. These tests will include:

- Testing that the Wi-Fi chip can send and receive data
- Testing that the camera input can receive video
- Testing that the processor function
- Testing that the processor can handle the correct load without issue
- Testing that the 802.11 protocols function properly.

If the CC3200 passes all these tests it will be ready for integration testing. It will be carefully placed back into its packaging and box and set aside until all the other hardware has arrived, been tested and is ready for integration testing. It will be removed if it is needed for further hardware or software testing.

7.212 TFT LCD Display Testing

The next part that will need testing is the TFT LCD display. We will need to check to see that the display is functioning properly. We will also need to check that it is responsive to touch in a quick and reliable manner. This part will be tested to be sure that:

- The screen has no dead pixels
- The touchscreen is responsive
- Input and output function properly
- No heating issues
- The screen’s coloring is not faulty

If the TFT LCD display passes all these tests it will be ready for integration testing. It will be carefully placed back into its packaging and box and set aside until all the other hardware has arrived, been tested and is ready for integration testing. It will be removed if it is needed for further hardware or software testing.

7.213 Micron mt90p001 Testing

The next part that will be tested will be the Micron mt90p001. It is important to make sure that the image sensor will be able to perform up to specs as well as that it is functioning properly. It also needs to be able to identify and read the barcodes that are held up to it. This part will be tested to be sure that:

- The lens is undamaged
- Input and output functions properly
- The images output are clear and understandable
- Can clearly read the barcodes it is required to
- Input and output connections are undamaged

If the Micron mt90p001 passes all these tests it will be ready for integration testing. It will be carefully placed back into its packaging and box and set aside until all the other hardware has arrived, been tested and is ready for integration testing. It will be removed if it is needed for further hardware or software testing.

7.214 ARM Core A9 Testing

The next part that will be tested will be the ARM Core A9. We need to make sure that the processor will perform up to specs. It needs to be able to handle the data flow in and out. It needs to stay within proper heat levels under intense operation. This part will be tested to be sure that:

- Can handle the computing load required
- It has no overheating issues
- Receives inputs and outputs properly
- It has been stress tested for maximum load
- Meets the specified specifications in the data sheet

If the ARM Core A9 passes all these tests it will be ready for integration testing. It will be carefully placed back into its packaging and box and set aside until all the other hardware has arrived, been tested and is ready for integration testing. It will be removed if it is needed for further hardware or software testing.

7.215 SSD NOW S100 Testing

The next part that will be tested will be the SSD NOW S100. We need to make sure that the hard drive will perform up to specs. It needs to be able to handle the data flow in and out. It needs to stay within proper heat levels under intense operation. This part will be tested to be sure that:

- Has no heating issues
- Properly receives inputs and outputs
- Can store the proper amount of data
- Can be split into separate sections

If the SSD NOW S100 passes all these tests it will be ready for testing. It will be carefully placed back into its packaging and box and set aside until all the other hardware has arrived, been tested and is ready for integration testing. It will be removed if it is needed for further hardware or software testing.

7.22 Integration Hardware Testing

After each individual part has been tested they will then be tested together on a breadboard. On part will be properly attached to the board at a time and re tested to see that it functions properly under a variety of currents and voltages using an oscilloscope, multimeter, and battery. After each part has been double checked, we will slowly combine the parts to make sure they all work together as one whole. All the inter part confections will be tested and monitored using the equipment on hand to make sure that they do not violate any levels that could cause damage or incapacitate the part(s).

7.3 Software Test Environment

Alfred's processors and network chip are TI products so naturally we will choose TI's Code Composer Studio as the software testing environment for these components. As for the operating system, Androids Eclipse ADT will be used to test and implement the overall operating system for the device as well as the applications he will be running.

7.31 Code Composer Studio

Code composer studio is an integrated development environment that offers a wide array of tools used to develop and debug embedded applications. Features include [17]

- Optimized C/C++ compiler
- Source code editor
- Project build environment
- Debugger
- Profiler

Code composer also includes support for specific processors including the MSP430 Ultra Low power MCUs and C2000 Real-time MCUs, both of which are included in Alfred. All members of our group have worked with Code Composer Studio and have familiarized ourselves with the testing environment and controls of the program. Code composer offers Grace to generate peripheral set up code very quickly. Also included is Energia which allows for importing open-source prototyping for use with the MSP430.

We will be using C-code as our primary programming language for the micro-controllers as all members in the group are most familiar with this style of programming. We will make use of the built in libraries. The following are the libraries we expect to include for the embedded controls portion of our project.

- **math.h –** The standard math library with many functions including trigonometric functions that will be useful in signal processing. It will also allow for the program to convert and modify any raw data brought in to successfully store to memory.
- **stdio.h –** The standard library for performing input and output functions. Each part of the project will utilize these functions when transferring data from one subsystem to the next.
- **stdlib.h –** This library contains several general operation functions and macros that will be used within the code. These functions and macros are will help with memory allocation, more math functions, as well as searching and sorting functions for data

- **string.h –** This library will allow for the manipulation of the physical data being input and output by the user.
- **time.h –** This library will allow for us to display time more efficiently to the user. This will help with date stamps of when items were checked in and out of the rooms.

As we will be using our own computers to run the test environments, we also have to consider the system requirements of the software. The following system requirements are for Code Composer Studio v6, the most recent installment. The minimum required memory is 2GB with 4GB being the recommended. The required disk space is 400 MB but recommends 2GB. A processor compatible to the 1GHz x86 is the minimum but recommends a dual core x86 or greater. Code Composer is able to run on Windows XP, Windows7 or Windows8.

7.32 Eclipse ADT

Eclipse Android Development tool is a fully integrated test environment used to build Android applications. Eclipse is primarily used for Java but may also be used with C, C++, JavaScript, Python and many others when loaded with plug-ins. Eclipse is able to work with LaTeX, a typesetting language, and Telnet, a networking application. Telnet is a network protocol that provides bidirectional text communication via virtual connection. The connection is not as secure as a secure shell network when operation on an open network, however, companies tend to use security protection, so this will not be an issue. Below are some features to using the Eclipse development tool. [18]

- Integrated Android project tools
    - Creation
    - Building
    - Packaging
    - Installation
    - Debugging
- Java programming language editor
- Custom XML editors
- Integrated API framework documentation

Like Code Composer, we will need to look at the system requirements for Eclipse. The following operating system requirements are for the most recent version of Eclipse. [18]

- Windows
    - XP – 32-Bit
    - Vista – 32-Bit or 64-Bit
    - Windows 7 – 32-Bit or 64-Bit
- Mac OS X
    - 10.8.5 or later
- Linux

- 64-bit distribution that runs 32-bits apps
- GNU C Library 2.11 or later
- Tested on Ubuntu 12.04

7.4 Software Specific Testing

Alfred's software will need to be rigorously tested in order to guarantee proper retrieval, transmission, and storage of data. For data retrieval, the barcode scanning program will need to be tested. For the transmission portion, the wireless communication and microcontrollers will need verification. Lastly, to test for proper storage of data, we will test our search function and manual input for accuracy.

7.41 Retrieval

In order to test that the barcode scan program is running properly and will be able to handle several scenarios, we will create a test bank of barcodes to scan. As there are never ideal scenarios when working in the real world, the barcodes will be of varying quality to ensure we maximize use of scanner vs. manual input from the user. The qualities of barcodes will be as follows:

- Low contrast
- Low quality print
- Printed on non-flat surfaces
- Printed on transparent material
- Printed on reflective material
- Small size/proportion

The barcode scanner will be connected to the MSP430 in our design. In testing the barcode scan code is working properly, we will connect the MSP430 to the Launchpad along with the barcode scanner. We will begin scanning barcodes one at a time down the test bank we have created. After we scan each barcode, we will use Code Composer Studio to verify that the MSP430 is decoding the information from the barcode properly and fetching the data from the proper location based on the decoded input.

In addition to the barcode scanner, the user may enter information directly to the touchscreen. We will again hook up the touch screen to our MSP 430 in order to test that our interface understands the users input. The user will be able to select text boxes and manually type in name, description, location, price, etc. We will test that each input is being recognized correctly.

7.42 Transmission

The user may only view data on the screen of the device but internally the data will be moving around to the different components. It is for this reason we must test the software that facilitates the data interaction to make sure the data is being shared between systems properly without data loss or corruption. We may do this by using the built in features of Code Composer Studio. We will follow the flow of data between registers. Using features such as the step in/over method, we can trace any mistakes in transmission to the source of the infraction. We will need to test the data transfer connections between the microcontrollers and the hard drive, screen interface, and wireless chip. In addition, we must also test the dataflow between each of the microcontrollers to confirm proper compatibility of the devices.

7.43 Storage

To test storage, we will need to input data of various types. We will need to verify how large the entries may be for proper storage so we do not rewrite any data incorrectly. We will test the storage with the three types of inputs that can write to our hard drive. These inputs include the manual input from the user, input from the wireless receiver to storage and the input transferred from the item list in to inventory storage. After each new entry is placed on the hard drive, we will perform checks against a master list. If we are going to monitor the addition of data, we also need to monitor the deletion of data. Alfred's data will be able to be deleted by use of a delete button. The user may also set the quantity of an item to zero to delete an item. We must check that both avenues properly dispose of the data without affecting other items. One more test will have to be done on Alfred's storage for when an item is edited. We must make sure that an edit to an items information does not alter other items in the list unless linked.

7.44 Wireless

Alfred's wireless communications will be controlled and monitored by his Android software. We will test the update functionality of Alfred over the wireless connection. We need to be sure that the proper safeguards are in place so that one Alfred does not overwrite another causing a loss of information. We also need to check that if signal is lost the update will occur instantly when signal is retained.

The first step of this procedure will be to update one device with a new data entry while we have wireless signal. We will then check the second device for the updated information by use of the search function. Next, we will disable the wireless capability of the first device and then update the inventory. We will turn on the devices signal and check the second device is able to view the new inventory. Third, we will attempt to update the information stored on the second device using the first. While both units have a signal, we will search for an item in the second device. Using the edit feature, we will alter the information for item and save the entry. The second device will then be checked for the newly edited information. Due to information for each room being stored internally at each module, an Alfred device cannot change another's inventory unless both units are connected to the network. This will also help to prevent any overwrites causing loss of information during a network outage.

8.0 Administrative Content

This section contains the administrative details of Project Alfred. It will go in-depth on the timeline for completion as well as the budget for spending. Furthermore it will go into detail on what the budget is spent on, internal goals, and their desired completion dates. The data or plans will be displayed graphically using tables or flow charts then be described in detail.

8.1 Milestone Discussion

We aim to complete Project Alfred in a timely manner. We also plan to leave ourselves with a good deal of wiggle room to account for the unexpected. We have split the project into three phases. The first phase lasts until the rough draft of the project document is due. The second lasts until the end of winter break. The final phase lasts the full spring semester. Though, we aim to be done well before the semester ends. The following figure is a flow chart of Project Alfred's initial phase planning and scheduling. The following figure contains a key of what the basic meaning behind each shape will be for the following figures within this section.

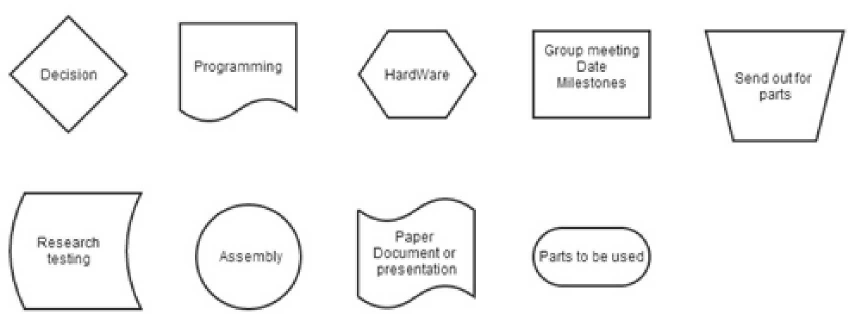


Figure 8.1.1 flowchart key

As you can see in the figure 8.1.2 at the end of this section, the first phase takes place over the course of eleven weeks. The first week was spent finalizing our design goals, picking a main processor, dividing the research between us, and setting our deadlines to keep us on track. The next week is when all the research on ac to dc converters needs to be done. In the following two weeks, the research for the main system storage device as well as the LCD screen research will be started and completed. The week of October ninth is a big milestone week. It is when we not only start and finish the mounting research but also when we review the programing we want to use to complete the software. This will help us narrow down what chips to use as well as give us an idea of the specs the chips will need. The next week will be spent checking the compatibility for all the parts. It will allow us to further eliminate hardware that won't be able to function together. The next week will be spent checking for anything important that we may have missed that we would like reviewed when we submit our rough draft to Dr. Richie. We are scheduled to meet with Dr. Richie on 11/24/14 to go over our rough draft. After which we will record any changes that need to be made and divide up the remains of the paper. This will lead to the beginning of the second phase of Project Alfred.

The next figure at the end of the section, 8.1.3, shows the second phase of project Alfred. The second phase of Project Alfred begins during Thanksgiving break. All members are expected to have completed their respective parts by the first Tuesday after the break. On that Tuesday all members will meet up for final formatting of the paper. During this time the paper will be reviewed in its entirety for spelling, grammar errors, and syntax errors. At this time the paper will also be put into the correct IEEE format, all sources will be cited, and any other details that need to be taken care of before final submission. If all this is completed in a timely manner we will stop by Dr. Richie's office for a second go over before final submission. If everything checks out it will then be bound and turned in. if things need to be changed or edited, the changes will be divided amongst the team and we will meet again the next day for the final printing and binding of the paper. The paper will be handed in one hour early on Thursday for safe measure after which much alcohol will be had. After this group members will order the parts needed for their respective subsystems and group meetings will be suspended for the winter holiday. Members are expected to learn, practice and code as much coding that can be reasonably done over the winter break. There will be a synopsis meeting the Thursday before school starts. This meeting will allow us to see how far we've come on each subsystem as well as make any adjustments in planning that may be required for phase three. We will also check to see what parts have arrived and what still needs to be bought. This will lead to the beginning of the third phase of Project Alfred.

The next figure 8.1.4, shows the third phase of project Alfred. The third phase of project Alfred begins on the first day of the spring semester. The first meeting will be held on the fifteenth of January. At this meeting will again check up on the final code progress and review the original paper. This is also the arrival deadline for all parts. All parts will then be tested for faults so they can be sent back immediately if there are any issues. At the next meeting we will begin debugging the program as well as start assembling the hardware. At this time we won't make any permanent connections, just connect everything and make sure the code works with it all put together. At the next meeting we will begin dividing up and writing the project presentation. Assuming there have been no issues final assembly will begin. The next week will be hardware integration and integration testing. The next two weeks will be spent testing, troubleshooting, and bug checking Alfred. During this time we will cover every function, every scenario, and every possible error in order to make Alfred a smooth running finished product. There will be a meeting February twenty sixth to assemble all the parts of the presentation and give it a final review before submission. On March fifth, we will rerun all major checks in order to assure Alfred is still running smoothly. This should complete Project Alfred the Building Master. We have left a few weeks leeway in order to leave us wiggle room for any unforeseen hiccups or issues that will likely occur.

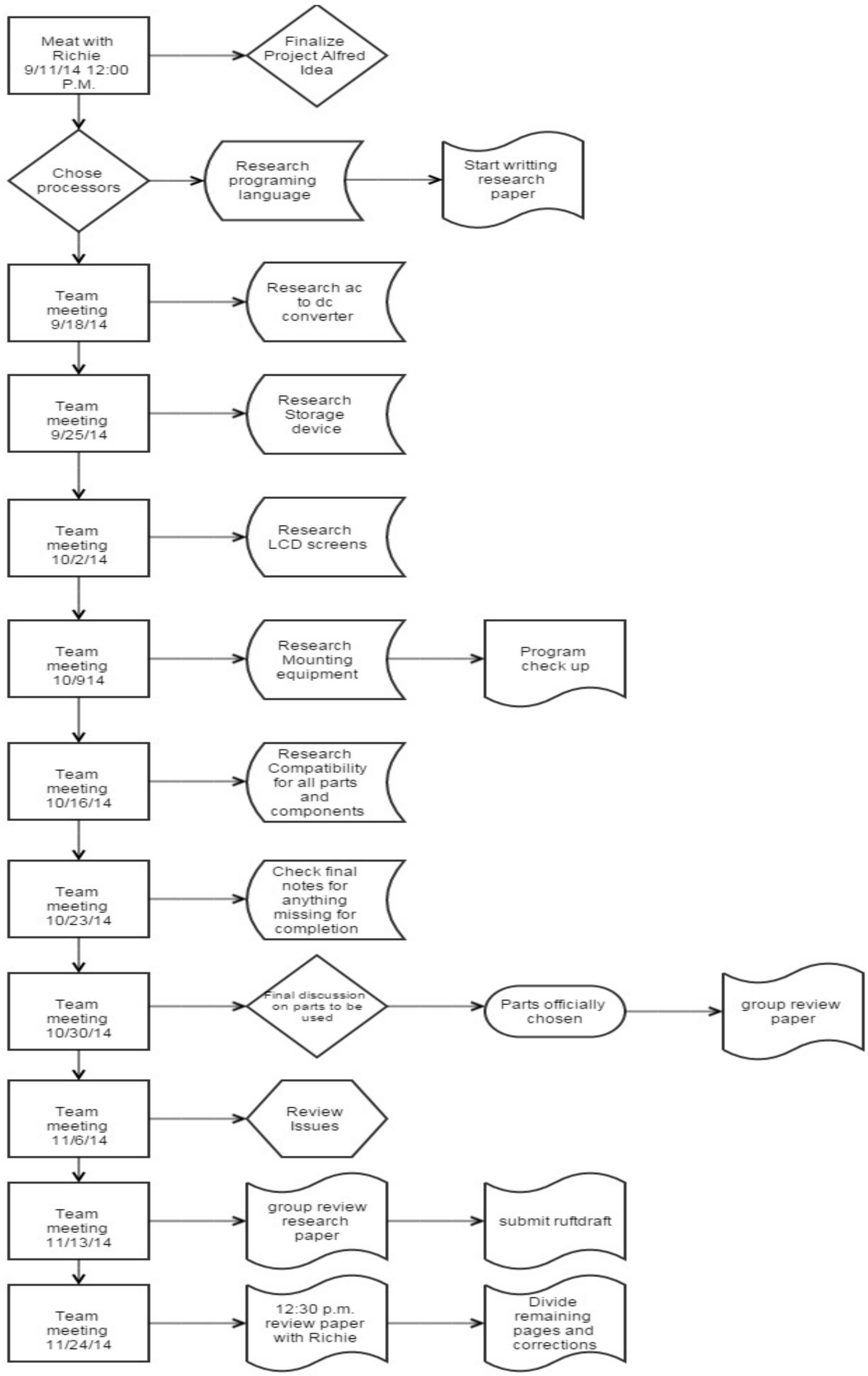


Figure 8.1.2 beginning to first review

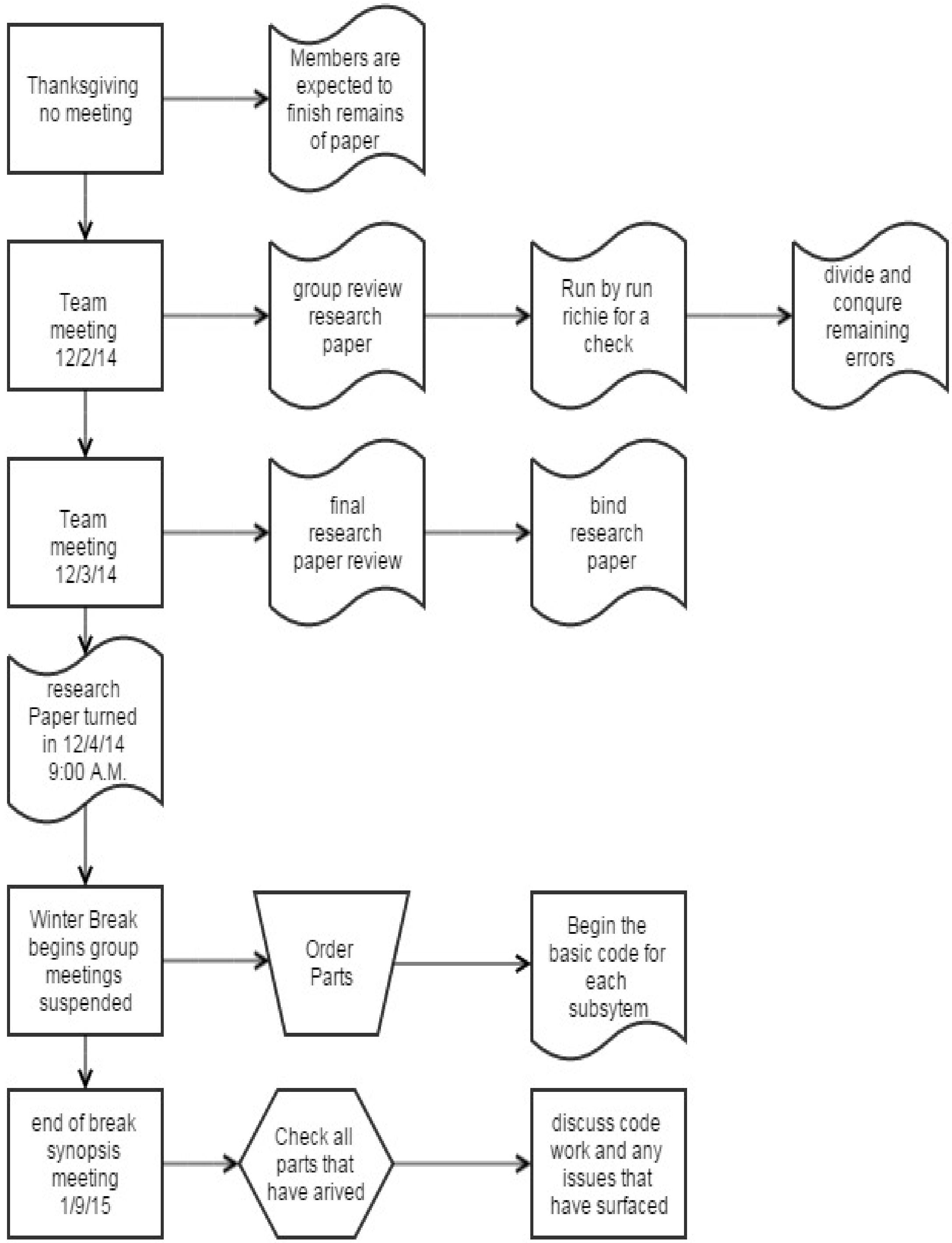


Figure 8.1.3 review to start of next semester

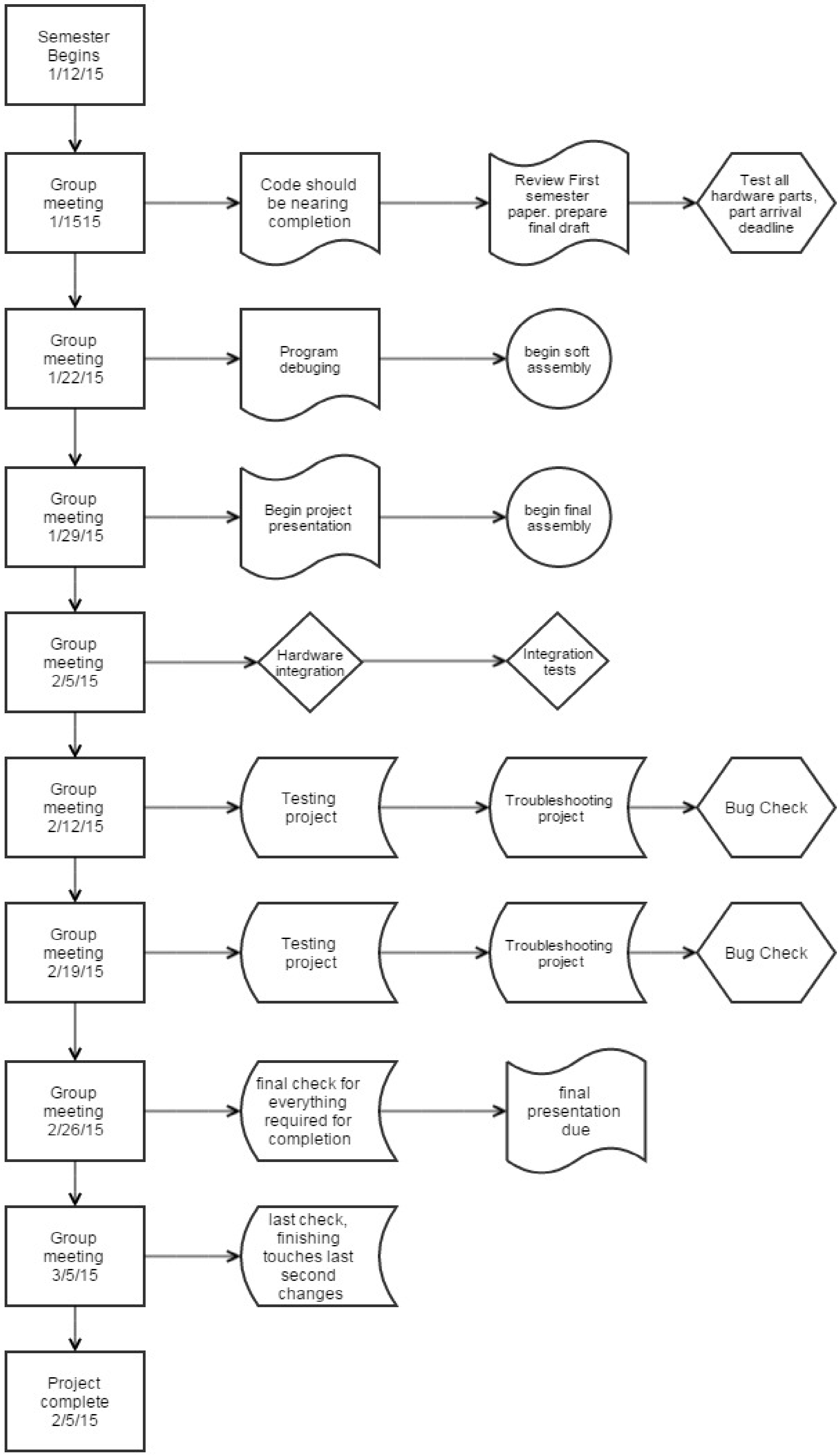


Figure 8.1.4 Second semester

8.2 Budget and Finance Discussion

In this section we will discuss the budget and finance for project Alfred. Through our sponsor, Boeing, we have received six hundred and eighteen dollars to be used as a budget for project Alfred. Most of the money will be spent acquiring the necessary parts to complete the project though some will go toward the design and building of a PCB board. The table below shows the different costs of all the products that will be used in the construction. Some of the products show in the table are provided to have more than one product. The PCB board comes with 4 boards to make sure that if an error is made that we can use another. In addition, the battery price is 6 dollars for 5 batteries. The team believes that the prototype will only need one battery, but batteries have to be bought in bulk. Otherwise the products are only bought with one part in mind that may or may not come with extra components.

subsystem	part	price
network adapter	cc3200	$18.08
micro controller	msp430	$4.30
touch screen	TFT LCD display	$62.08
SSD	SSD NOW S100	$50.00
Processor	ARM Core A9	$70.00
RFID reader	TIDM RFID	$49.99
Main Box	plastic box	$50.00
AC to DC converter	PMP 4390	$6.00
Camera	Micron	$15.20

	mt90p001	
battery power system	lithium ion polymer	$6.90
PCB Manufacturing		$60.00
product software		$120.00
miscellaneous equipment		$100.00
Total		$612.55

Figure 8.2.1 Planned Budget for parts and expenses

The cost of hardware alone is three hundred thirty two dollars and fifty five cents. This cost does not include shipping. Only accounting for the price of parts will leave us with two hundred eighty five dollars and forty five cents. Some products come with full assembly of boards and other products than just the main chips. For more information on what each item purchased includes see the section 5.0. The remaining money will be spent on miscellaneous equipment, shipping, and software licensing required for finishing the project. It is unlikely we will spend more than fifty dollars on product software but we have set an upper limit of one hundred to be safe. Building the PCB is likely going to cost at least fifty dollars. If it goes over sixty we have left room in the miscellaneous budget to dip into. The main part of the miscellaneous budget is going to be spent on small items like capacitors, resistors, and other low level parts. In addition extra parts, the miscellaneous equipment is made for contingency purposes in case we need more parts or one of the other parts break in construction.

Figure 8.2.1 table is a reference to what products will most likely be used in product of project Alfred. However, with this in mind many of the products are subject to change due to availability. Also, design changes may occur throughout the construction period to optimize the prototype on increase success of project completion. Therefore, this is representation of the design intended for production purposes and may not be exact to design. The design team deserves the right to change, replace or remove products if needed.   If this does occur the products chosen to replace components will be similar to the products that were suggested in the design phase of the document. As a result, the design will remain the same just some of the component’s manufacture or type maybe different to ensure quality and availability.

9.0 Appendix

9.1 Appendix A: References

[1] Bill Buxton, Simon, [Online]
http://research.microsoft.com/en-us/um/people/bibuxton/buxtoncollection/detail.aspx?id=40

[2] TEC-IT, Rapid Inventory App for Android, [Online}
http://www.tec-it.com/software/android/mobile-inventory/overview/Default.aspx

[3] Microsoft, Access, [Online]
http://products.office.com/en-us/access

[4] Disney, MagicBands & Cards - Frequently Asked Questions, [Online]
https://disneyworld.disney.go.com/faq/bands-cards/understanding-magic-band/

[5] Wikipedia, Wireless Mesh Networks, [Online]
http://en.wikipedia.org/wiki/Wireless_mesh_network

[6] ON Semiconductor, Barcode Scanner, [Online]
http://m.onsemi.com/PowerSolutions/appDiagram.do?appId=651

[7] Stevan Preradovic, Modern RFID Readers, [Online]
http://www.microwavejournal.com/articles/5271-modern-rfid-readers

[8] Kurt Denke, Audio Power Cables & Cords, [Online]
http://www.audioholics.com/audio-video-cables/power-cables

[9] Webmaster, Types of Storage, [Online]
http://typesofbackup.com/types-of-storage/

[10]Kingston, SSDNow Datasheet, [Online]
http://www.kingston.com/datasheets/sv100s2_us.pdf

[11] Tech Explainer, Resistive Vs Capacitive, [Online]
https://techexplainer.wordpress.com/2012/04/02/resistive-vs-capacitive-touchscreen/

[12] Geoffrey Morrison, LED LCD vs. OLED vs. Plasma, [Online]
http://www.cnet.com/news/led-lcd-vs-oled-vs-plasma/

[13] ARM, Cortex-A9 Processor, [Online]
http://www.arm.com/products/processors/cortex-a/cortex-a9.php

[14] ARM, Cortex-M4 Microcontroller, [online]

http://www.arm.com/products/processors/cortex-m/cortex-m4-processor.php

[15] Texas Instruments, MSP430G2553, [Online]
http://www.ti.com/product/msp430g2553

[16]Crystalfontz, CFAF480272G-043T-CTS, [Online]
https://www.crystalfontz.com/product/CFAF480272G043TCTS

[17] Texas Instruments, Code Composer Studio (CCS), [Online]
http://www.ti.com/tool/CCSTUDIO

[18] Android Developers, Get the Android SDK, [Online]
https://developer.android.com/sdk/index.html?hl=i

[19] Micron inc, MT9p031 Image Sensor, Boise ID, 2006

[20] Aptina Imaging, MT9p03, 2012

[21] Marchall Brian, How UPC Barcodes work, [online]
http://electronics.howstuffworks.com/gadgets/high-tech-gadgets/upc.htm"

[22]Texas Instruments, RFID Transciever Reference Design, [online]
http://www.ti.com/tool/tidm-rfid-transceiver?keyMatch=RFID&tisearch=Search-EN

[23] Texas Instruments, Power Management [online] http://www.ti.com/lsds/ti/power-management/wireless-power-receiver-solutions-overview.page

[24] all battery.com, Li-Polymer 3.7v 1100mAh 20C (603565) Battery [online]
http://www.all-battery.com/37volt-1100mah15cli-polycell.aspx

[25] Texas Instruments, Universal AC Input 15W Adapter with Energy Star 6 Efficiency Rating Reference Design [online]
http://www.ti.com/tool/pmp4390?keyMatch=AC%20to%20DC%20Adapter&tisearch=Search-EN

[26]Wikipedia, Mesh Networking, [online]
http://en.wikipedia.org/wiki/Mesh_networking

[27]Wikipedia, Mesh Personal Area Netwrok, [online]
http://en.wikipedia.org/wiki/Personal_area_network

[28]Wikipedia, Wireless LAN, [online]
http://en.wikipedia.org/wiki/Wireless_LAN

[29]Wikipedia, IEEE 802.11n-2009, [online]
http://en.wikipedia.org/wiki/IEEE_802.11n-2009

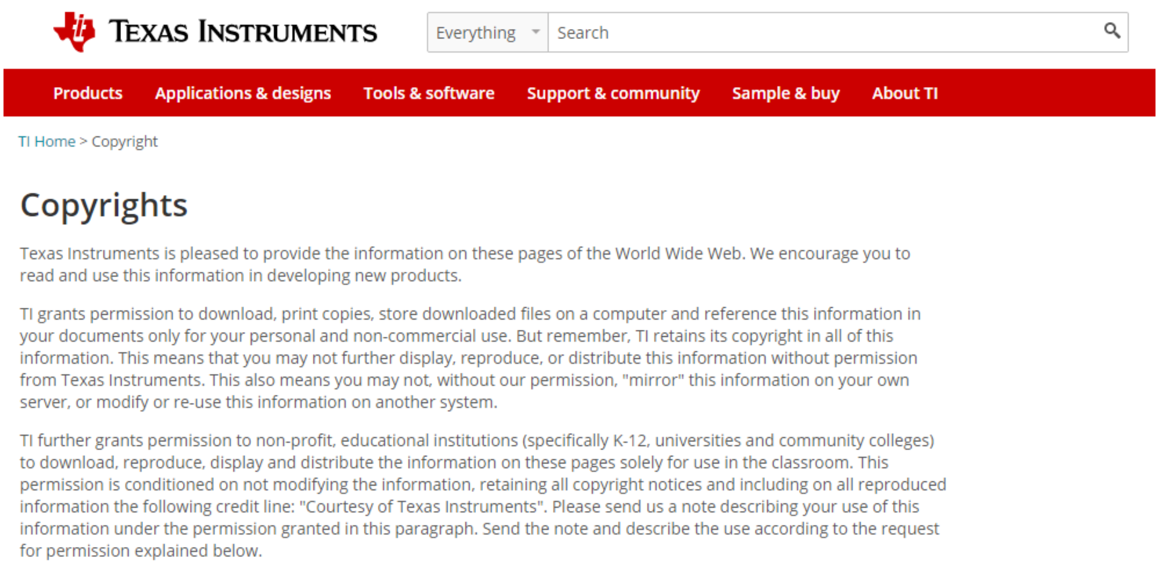
[30]Wikipedia, Orthogonal frequency-division multiplexing, [online]
http://en.wikipedia.org/wiki/Orthogonal_frequency-division_multiplexing

[31]Wikipedia, Direct-sequence spread spectrum, [online]
http://en.wikipedia.org/wiki/Direct-sequence_spread_spectrum

[32]Texas Instruments, Wireless Connectivity, [online]
http://www.ti.com/lit/sg/slab056d/slab056d.pdf

[33]Texas Instruments, CC3200, [online]
http://www.ti.com/product/CC3200/datasheet/abstract#XXXX0012474

9.2 Appendix B: Copyright Permissions



TEXAS INSTRUMENTS

Everything | Search

Products | Applications & designs | Tools & software | Support & community | Sample & buy | About TI

TI Home > Copyright

Copyrights

Texas Instruments is pleased to provide the information on these pages of the World Wide Web. We encourage you to read and use this information in developing new products.

TI grants permission to download, print copies, store downloaded files on a computer and reference this information in your documents only for your personal and non-commercial use. But remember, TI retains its copyright in all of this information. This means that you may not further display, reproduce, or distribute this information without permission from Texas Instruments. This also means you may not, without our permission, "mirror" this information on your own server, or modify or re-use this information on another system.

TI further grants permission to non-profit, educational institutions (specifically K-12, universities and community colleges) to download, reproduce, display and distribute the information on these pages solely for use in the classroom. This permission is conditioned on not modifying the information, retaining all copyright notices and including on all reproduced information the following credit line: "Courtesy of Texas Instruments". Please send us a note describing your use of this information under the permission granted in this paragraph. Send the note and describe the use according to the request for permission explained below.