TABLE of CONTENTS

TABLE of TABLES

TABLE of FIGURES

EXECUTIVE SUMMARY

1. This Document…
2. About NOMS

PROJECT DESCRIPTION

1. Motivation & Goals
2. Objectives
3. Requirements and Specifications

RESEARCH FOR DESIGN

1. Existing & Similar Products
   1. Beyond WBYMW1
   2. Nathan Broadbent’s version
   3. Hobart’s Combi Oven
   4. Knight Watch Security
   5. Smidge
2. Relevant Technologies
   1. Microcontrollers
   2. Barcode and QR Code Scanners
   3. LCD Touchscreens
3. Strategic Components
   1. Single-Band vs. Dual-Band Wi-Fi
   2. Embedded Wi-Fi vs. Installed Wi-Fi Shields
   3. Single vs. Multiple Embedded Systems
4. Possible Architectures

PROJECT HARDWARE & SOFTWARE DETAILS

1. Power Management
   1. Power Systems
      1. Basis of Design
      2. Food Preparation Research
      3. Heating Food with the Microwave
   2. Electrical Research
      1. Linear Power Supply Design
      2. Full Wave Rectifier
      3. Circuit Design Full Wave Rectifier
      4. Multiple Winding Transformers
      5. Full Wave Bridge Rectifier Circuit Design
      6. Voltage Doubler
      7. Villard Circuit
      8. Electric Motor
      9. Induction Motor
      10. Synchronous Motor
      11. Brushed DC Motor
      12. Brushless DC Motor
      13. Stepper DC Motor
      14. Magnetron
      15. Power Supply Stability
      16. Power Output Requirements
   3. AC Components
      1. Magnetron
      2. Turntable
      3. Fan Motor
      4. Lights
      5. AC/DC Transformer
      6. Buzzer
      7. Summary of High Power Consumption Devices
   4. DC Components
      1. Controllers
      2. Camera
      3. Touchscreen
      4. Relay
      5. Solid-State Relays vs. Electro-Mechanical Relays
      6. Relay Switch Selection Process
      7. Regulator
      8. Summary of Low Power Consumption Devices
   5. Power Systems Testing
      1. High Voltage Power Protection and Feedback
      2. Low Voltage Power Monitoring and Feedback
      3. Low Voltage Power Step-Down
2. System Controller
   1. Requirements and Specifications
   2. Circuit Card Assembly
3. Embedded Wi-Fi
   1. Functional Requirements/Design Considerations
   2. Existing Systems Choices
      1. XBee Wi-Fi Module
      2. CC3000 SimpleLink Wi-Fi Module from TI
   3. Subsystem Overview
   4. Configuration of the CC3000
   5. Parts Acquisition
4. Online Database
   1. Functional Requirements
   2. Existing Database Choices
   3. Creating a Server
   4. Integration of the Database with the Wi-Fi SOC
   5. Acquiring and IP Address
   6. Querying the Database
5. UPC Codes
   1. Motivation for Use
   2. UPC Standards
   3. EAN-13 Specification and Composition
   4. UPC Capture
      1. Grayscale Conversion
      2. Filter
      3. Black and White Conversion
      4. Deskew
      5. Crop
      6. UPC Decoding
      7. Check Digit
   5. Programming Considerations
6. QR Codes
   1. Motivation for Use
   2. QR Standard
   3. QR Specification
   4. QR Composition
   5. QR Generation
   6. QR Code Capture
   7. QR Decoding
7. User Interface
8. System Display & Touchscreen
   1. Requirements
   2. LCD Selection
   3. Programming the LCD
   4. LCD & System Controller Interconnect

SUMMARY OF HARDWARE AND SOFTWARE

1. Hardware
2. Software

PROTOTYPE CONSTRUCTION

1. Parts Acquisition & Bill of Materials
2. PCB Vendor & Assembly
3. Final Coding Plan

PROTOTYPE TESTING

1. Test Environments
2. Specific Testing
3. Conclusion

ADMINISTRATIVE CONTENT

1. Budget
2. Budget Discussion
3. The Inspiration for NOMS
4. Milestone Discussion
5. Semester 1: Senior Design 1 Milestones
6. Semester 2: Senior Design 2 Milestones
7. About Us
8. History of Easily-Prepared Food

APPENDICIES

1. Permissions
2. References

TABLE of TABLES

|  |  |  |
| --- | --- | --- |
| Table 1 | Member Contribution | Page 4 |
| Table 2 | Frozen Food Research | Page 15 |
| Table 3 | Microwave Data | Page 16 |
| Table 4 | Magnetron Power Data | Page 27 |
| Table 5 | Turntable Power Data | Page 28 |
| Table 6 | Fan Motor Power Data | Page 28 |
| Table 7 | Light Power Data | Page 29 |
| Table 8 | AC/DC Transformer Power Data | Page 30 |
| Table 9 | Buzzer Power Data | Page 30 |
| Table 10 | Summary of High Power Consumption | Page 31 |
| Table 11 | Microcontroller Power Data | Page 32 |
| Table 12 | Camera Power Data | Page 32 |
| Table 13 | Touch Screen Power Data | Page 32 |
| Table 14 | Regulator Power Data | Page 35 |
| Table 15 | Summary of Low Power Consumption | Page 35 |
| Table 16 | Microcontroller Comparison | Page 41 |
| Table 17 | Number Sets A, B, and C | Page 56 |
| Table 18 | Implicit Digit Encoding | Page 57 |
| Table 19 | Cook Data Format | Page 61 |
| Table 20 | Interconnection Protocol | Page 77-79 |
| Table 21 | Table of mainboard connections | Page 82 |
| Table 22 | Budget | Page 91 |
| Table 23 | Semester 1 Milestones | Page 94 |

TABLE of FIGURES

|  |  |  |
| --- | --- | --- |
| Figure 1 | High Level State Transition Diagram | Page 1 |
| Figure 2 | NOMS Sub-Systems Diagram | Page 3 |
| Figure 3 | Typical Kitchen Electric Layout | Page 13 |
| Figure 4 | AC Waveform | Page 19 |
| Figure 5 | Full Wave Rectifier Output | Page 19 |
| Figure 6 | Full Wave Rectifier Circuit Design | Page 20 |
| Figure 7 | Multiple Transformer Windings | Page 20 |
| Figure 8 | Full Wave Bridge Rectifier Circuit | Page 21 |
| Figure 9 | Full Wave Bridge Rectifier Output | Page 21 |
| Figure 10 | Villard Circuit Design | Page 22 |
| Figure 11 | Villard Circuit Output | Page 22 |
| Figure 12 | CEMBA and ITIC Curve | Page 26 |
| Figure 13 | Microwave Circuit Schematic | Page 36 |
| Figure 14 | High Voltage Protection Circuit | Page 38 |
| Figure 15 | Low Voltage Monitor Package | Page 39 |
| Figure 16 | Low Voltage Power Step Down | Page 40 |
| Figure 17 | Wi-Fi Block Diagram | Page 46 |
| Figure 18 | Database Structure | Page 49 |
| Figure 19 | Database Structure Flexibility | Page 50 |
| Figure 20 | Database Communications | Page 52 |
| Figure 21 | Network Topology | Page 53 |
| Figure 22 | Peer to Peer | Page 53 |
| Figure 23 | EAN-13 Symbology | Page 55 |
| Figure 24 | EAN-13 Composition | Page 56 |
| Figure 25 | QR Code Structure | Page 60 |
| Figure 26 | QR Code Layout | Page 61 |
| Figure 27 | Home Screen | Page 63 |
| Figure 28 | Manual Entry | Page 64 |
| Figure 29 | Scanning | Page 65 |
| Figure 30 | Override | Page 66 |
| Figure 31 | Save Custom | Page 67 |
| Figure 32 | Cook Screen | Page 68 |
| Figure 33 | Log Screen | Page 68 |
| Figure 34 | Servings | Page 69 |
| Figure 35 | System Settings | Page 70 |
| Figure 36 | Appearance | Page 71 |
| Figure 37 | Date/Time | Page 71 |
| Figure 38 | Time Entry | Page 72 |
| Figure 39 | Month Entry | Page 72 |
| Figure 40 | Day Entry | Page 72 |
| Figure 41 | Year Entry | Page 72 |
| Figure 42 | Profiles | Page 73 |
| Figure 43 | Name Entry | Page 73 |
| Figure 44 | Email Entry | Page 73 |
| Figure 45 | MyFitnessPal | Page 73 |
| Figure 46 | UI Flow | Page 74 |
| Figure 47 | Board Schematic | Page 83 |
| Figure 48 | PCB Layout | Page 83 |
| Figure 49 | Budget | Page 91 |
| Figure 50 | Semester 2 Milestones | Page 95 |

Executive Summary

The Nutritional Object-Identifying Microwave System (NOMS) is an improved microwave that automatically cooks food items based on identifying UPC/QR codes located on the food packaging. Current microwaves require users to input cook times and power levels in order to perform food preparation. We believe that as technology has advanced there has been a severe oversight in the realm of cookware. There are automated home systems, automated traffic signals, automated systems in cars, but no automated systems in the household kitchen. NOMS corrects this oversight in a simple and profitable way. Figure 1 shows a simple high-level state-transition diagram of the NOMS.

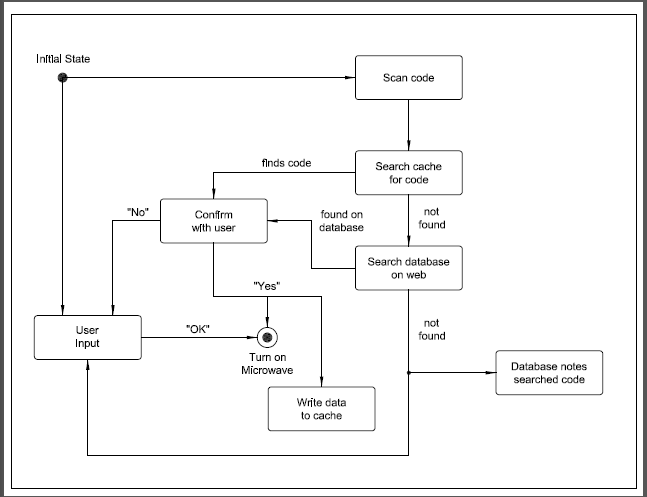


Figure 1 – High-Level State-Transition Diagram

NOMS makes use of cameras designed to look for UPC codes and/or QR codes. The user may choose to either scan the food in using one of the cameras or just enter the time and power level traditionally on the LCD screen. For the former, once a code has been identified the system searches the recently used codes in the cache for cook time and power level. If the code is found then the user will be asked if they want to cook the food according to the time and power levels found. If the user chooses ‘Yes’ then the system will turn on the microwave’s motor drive, magnetron, and lights at the specified level and time. If the user does not want to use that time or power level they will be able to use the traditional method of inputting time and power at which point the new information will be saved in the cache for future use. If, however, the correct code is not found in the cache then the system will send the code out to the NOMS website to find the proper cook time and power level. If the code is found, the Wi-Fi shield will send the data back to the system for use. If the code is not found the Wi-Fi shield will send a message back to the system showing that no valid data was found. The online database will also track searches that are not successful in order to provide future maintainability. It is at the point that the database online also does not provide cook time and power level that the user will have to resort to the traditional method of using the microwave.

Whenever the microwave has been turned on to heat food up and there is a code that was not found in the cache the system will save all of those settings and the code into the cache for future reference. In this way NOMS will be able to “learn” the users’ eating habits and be adaptable for the individual user.

1.0 This Document…

In the next section of this document, the Project Description, we will continue to further describe NOMS. We will summarize the origin of this project idea, the motivation and goals of the team and its project, and the accompanying requirements and specifications. The project description is what the group will use as a skeleton for the project. It provides a reference point that will be followed by each group member so that the completed project will be a visible, functional representation of what has been described.

In Research For Design we will show some existing technologies that we have researched in order to get a starting point for our project. We will show relevant work and products, components that we have chosen and a justification for why each component was selected for NOMS. Finally we will discuss the architecture we have selected for the design of NOMS

The Project Hardware and Software Details section will consist of the majority of this report, being the research aspect. As our group looked at how to take on the idea for NOMS we decided to break the project up into distinct parts for simplification. This helped us to divide up the individual workloads and to ensure that we didn’t miss any important steps in designing and developing NOMS. We chose to discuss these research topics in the specified order based on the flow of how NOMS will work. We begin by discussing the power management system which will in turn power the microcontroller. This microcontroller will have the Wi-Fi chip added onto it, first as a plug-and-play device and then finally surface mounted. The Wi-Fi chip will interface between the microcontroller and the online database so the database section will come next. After this we will discuss the camera and optics system which is where the EAN-13 codes will be coming from. These EAN-13 codes will be derived from the decoded UPC and QR codes which are further discussed. Finally we will end the research discussion with a brief discussion on the LCD touchscreen which will serve as the user interface allowing the user to use the NOMS system.

Following the research aspects of this document we will give a Summary of Hardware and Software that will be used in the design process of NOMS.

Then we will give a discussion on the prototype plans that we have created in terms of the Prototype Construction and then the Prototype Testing plan.

At the end of this document there will be some Administrative Content that will cover the budget and milestones of NOMS’ design followed by the Appendices with all the permissions, datasheets, and software.

2.0 About NOMS

The group decided in the early stages of planning and design to take a functional microwave and make additions to it in order to achieve our goal for the automated system. NOMS has been divided into modular pieces and each group member is specifically responsible for one or more tasks/subsystems. The diagram to follow (Figure 2) shows all subsystems, each of which is in communication with the Central Controller or System on Chip (SOC).

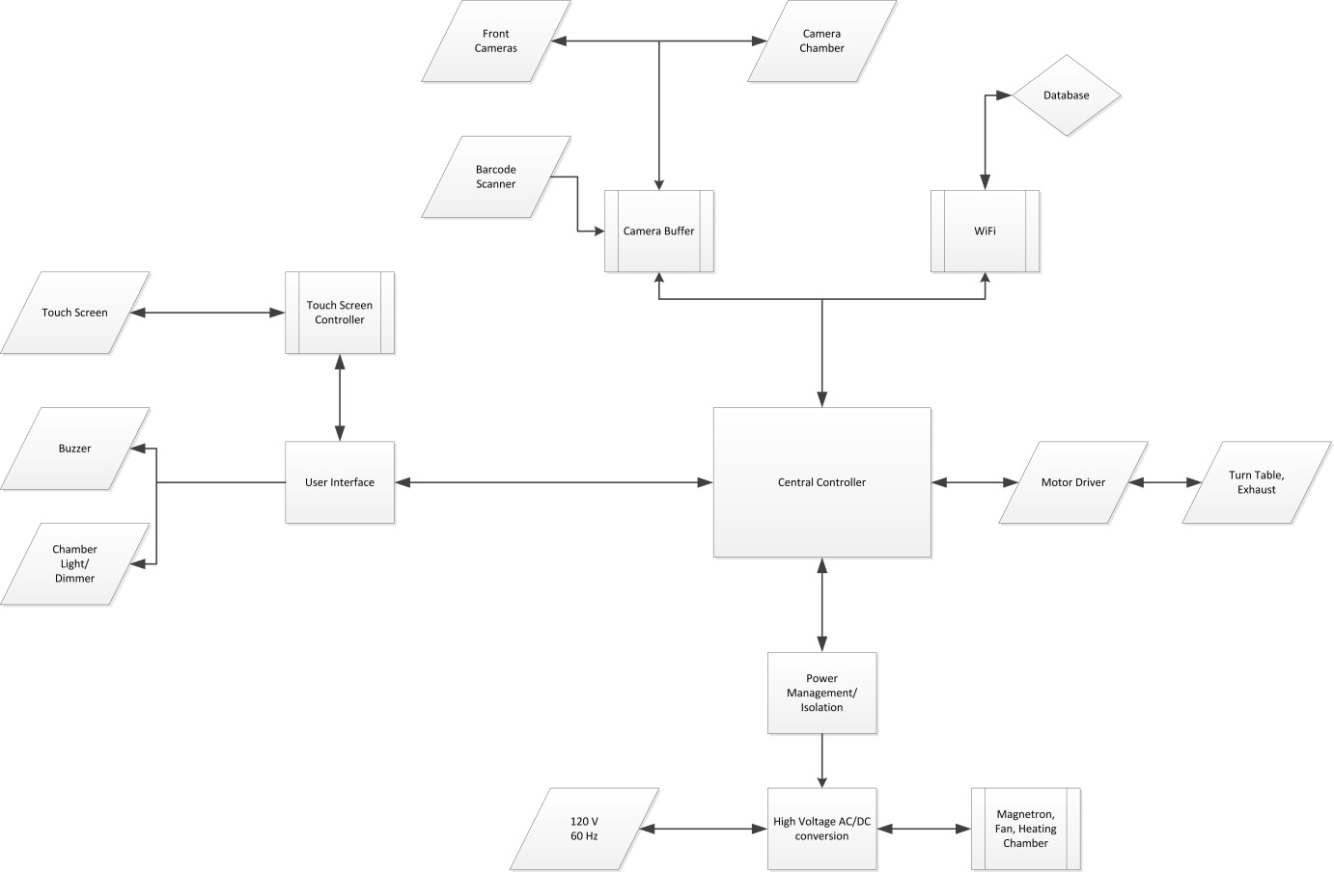


Figure 2– NOMS Subsystems Diagram

Table 1 contains the distribution of tasks to individuals in the group with the section of this document that each topic is discussed. While each member is specifically responsible for particular tasks it is understood that these tasks will overlap in some areas requiring collaboration and teamwork.

|  |  |
| --- | --- |
| Nakiesa Faraji-Tajrishi | Power Systems (4.5) |
| Tim Miller | Wi-Fi (4.1) and Online Database (4.2) |
| Jared Wach | User Interface and SOC (4.3-4.4) |

Table 1 - Member Contribution

The final section of this document (5) is the Administrative Content covering group dynamics such as milestones to be reached and financial decisions involved in budgeting.

Project Description

1.0 Motivation & Goals

Originally the idea for The Nutritional Object-Identifying Microwave System (NOMS) was one of the group members’ own, Jared Wach. Having worked with Jared in several other classes prior to Senior Design the group members decided it would benefit each person to work together to achieve our goals. Once each person agreed to be in the group we discussed different ideas for a project, finally choosing to go with Jared’s idea for the automatic microwave.

In addition the group had various degrees of experience working with electrical systems, programmable microcontrollers, and wireless internet. Each member found a particular area of interest in the project such that their theoretical knowledge could be put to use. The computer engineers found particular interest in the embedded System on Chip and integrating a wireless internet component into it. The electrical engineers found similar interest in devising a way to power each added component and building the embedded system itself.

The group was also motivated by the desire to design something challenging. The idea for NOMS seemed like just the right balance of feasibility and difficulty. The idea to create an automated microwave didn’t sound too different from previous Senior Design projects we heard about which confirmed to us that this project had the potential to be a success. In addition, had we chosen to design something simpler we might not have had the opportunity to learn as much or apply the previous knowledge we’ve accrued in our college careers. We look forward to being able to complete this project, look back on our hard work, and see the fruit of our labors in the complete NOMS Senior Design project.

One further motivation and goal of the group was to create something we had never seen before. To this point we still have not found another Senior Design project identical to this one nor have we found any products like it in the market. This gave NOMS a uniqueness that we felt would add to the value of this project. The value of this project may also transcend sentimentality to the point of profitability. Were we to market this idea it seems possible and probable that companies would be interested in using this idea to advertise their products to users. Though we do not presently intend to market this project this does add to the value of the project.

2.0 Objectives

Our objectives were to create a functional, automated microwave that made use of wireless internet and an online database created and maintained by the group. Our goal was to design a system that is intuitive enough that the average user would be able to use NOMS without too much previous instruction. In addition, our objective was to create a system that not only worked well but also had aesthetic appeal. Finally, we created a system that was as cost-effective as possible. It was certainly possible to create this system with expensive parts that were very simple to implement and use, however our goal was to keep this system as cheap as possible to show its potential for commercial production, which also kept the project itself as challenging as possible.

3.0 Requirements & Specifications

One aspect of NOMS that we focused on specifically was the energy efficiency of the system. In order to maximize efficiency we paid special attention to how much power was used by each added subsystem such as the LCD touchscreen, the embedded system on chip, and the normal microwave parts. Once each part’s power needs were known we matched our power output from the microwave’s pre-existing system with our additional modifications.

Another requirement for NOMS was that it would be easy to use. This may have somewhat vague-sounding as a requirement but we focused upon a way we could verify that NOMS was easy enough for the common user to use. This was an important attribute to focus on since we could have made a working microwave that was not particularly intuitive. However, after completing the project and testing it on the public we have decided this was not the case.

NOMS made use of an LCD touchscreen for interface between the user and the system. This display was both an output and an input and was designed to be pleasing to the eye and easy to use as discussed previously. The LCD was embedded into the front of the microwave with the assistance of a bezel. We placed the LCD touchscreen where the touchpad was previously before we tore the microwave apart. Our LCD that we chose was smaller than a three inch wide by four inch tall screen. The LCD is controlled by the system on chip (SOC).

NOMS uses wireless communication between the system on chip (SOC) and an online database. The wireless communication takes place from the SOC and was installed onto the system by the design team. We utilized the TI CC3000 for our wireless internet communication.

In order for the wireless communication to work there needed to be a working online database with a corresponding database management system. There were several choices for this database and its managements system as discussed further in the Hardware and Software Details section. This database needed to keep track of unique EAN-13 codes which identified each food item. Each food item contains multiple instructions for cook time and power levels to achieve optimal cooking of the food item. For example a Hot Pocket may need to be cooked at full power for one minute, then 30 more seconds at half power, followed by a full minute with the magnetron disabled for cooling.

In order to search the online database for the cooking instructions of a given food item the item needs to have the EAN-13 code for that item on the packaging. This code will be scanned in from a camera on the outside of the microwave that we installed into the front panel of the microwave. There was one camera installed on the microwave. The camera was installed on the outside of the microwave just below the LCD touchscreen. The second camera that was a “like-to-have” but not a must would have been installed on the inside of the microwave where it would have been embedded into the roof of the microwave so it could look down on the food. This would have been useful for food items with EAN-13 codes that could be read from the top of the food item. The implementation of this camera was not pursued as the design process took the entire semester and we did not have the additional time to advance this idea.

Research for Design

1.0 Existing & Similar Products

1.1 Beyond WBYMW1 850-Watt Microwave Oven with Barcode Scanning

A company named “Beyond” released a product similar to what we are creating that they called the “Beyond Microwave Oven”. This product made use of a barcode scanner. The website that sells Beyond’s microwaves boasts that it has “Over 4000 UPCs preprogrammed”. It can also store new barcodes that aren’t preprogrammed and has an LCD screen.

The major differences between this product and our project are:

* NOMS could use QR codes as well as barcode scanning
* NOMS uses cameras instead of a barcode scanner
* NOMS can directly use internet each time a code is not found so that no updating is required

In addition, we previously stated that we know of no existing smart microwaves that are on the market. This still remains true given that this product has been discontinued for unknown reasons. The website no longer sells Beyond’s smart microwaves and there are only a few to be found on bargain appliance websites. The fact that this product did not catch on may be cause for concern, though the appliance itself seems to have worked very well as seen in many comment sections about the appliance. It is possible that this product was created a little before the time that people were ready for it, in which case we expect to see more like it on the market in the near future as technology continues to expand and “smart” appliances continue to become more popular in homes. [1]

1.2 Nathan Broadbent’s version

Again we discovered that someone else had a similar idea for a “smart microwave”. This time a gentleman by the name of Nathan Broadbent took the idea he read about on a popular social website and decided that he could build it. Mr. Broadbent redesigned the touchpad, added voice command control, and implemented wireless internet into the system. He used a barcode scanner, an online database, and a microcontroller called Raspberry Pi. [2]

The major differences between Nathan Broadbent’s design and ours are:

* NOMS uses cameras instead of a barcode scanner
* NOMS will not use a Raspberry Pi, but will instead upon request of the client (Dr. Richie) use a System on Chip (SOC)
* Wi-Fi will be installed on the System on Chip

1.3 Hobart’s Combi Oven with Barcode Scanners

In the search for similar technologies we also discovered that this idea has gone to other appliances like the oven. A company named Hobart created their take on a “smart” oven called a Combi Oven with Barcode Scanners. This appliance also uses a barcode scanner and their system will even create a barcode for a recipe not found in their database. The oven has many features that are common to ovens like cooking modes and gas/electric types. This product does appear to be built for commercial use however. [3]

Hobart’s oven is similar to NOMS in many ways but NOMS:

* Can use QR codes as well as barcode scanning
* Is for both commercial and household use

1.4 Knight Watch Security

In addition to realms outside of the University of Central Florida there are a few similar projects from previous Senior Design classes that we were able to look at and learn from.

In searching for previous Senior Design projects that were similar to ours we looked for any projects that made use of wireless internet. This was a good starting point for gathering ideas on how to implement our wireless internet between the SOC and online database. We did discover a group from the Senior Design Fall 2012-Spring 2013 class whose project was called Knight Watch Security. In the end though, this team had pre-existing onboard Wi-Fi on their SOC which was different from the requirements given to our group. [4]

1.5 Smidge

Lastly, we also became aware of a project similar in concept called “Smidge”. This project was a “smart” fridge. It connected to the internet, had a barcode scanner and had many other features. Dr. Richie praised the team for their design and completion when discussing our project so we knew it would be wise to consider Smidge’s approach to research and design. [5]

2.0 Relevant Technologies

2.1 Microcontrollers

Based on the project description of NOMS we needed something to control the microwave itself and its interaction with the online database among several other aspects of NOMS that needed controlling.

Microcontrollers are basically small computers that are embedded into other systems in order to perform specific tasks. Common uses of microcontrollers are in TVs, cars, and even microwaves LCD touchscreens. Each of these microcontrollers performs a specific task within its system, as was the case in NOMS. In addition microcontrollers are a low-cost, low-power, and small solution for what we were trying to accomplish. These microcontrollers are also usually outfitted with a simple programming language like “BASIC” or “Processing” which makes programming the microcontroller relatively simple.

A few examples of microcontrollers used for tasks such as what NOMS does are the Arduino, Raspberry Pi, and BeagleBone. In fact, Nathan Broadbent’s barcode scanning microwave (section 3.1) used a Raspberry Pi. For our project we will design our own embedded system based on designs in Texas Instruments products, Arduino, and Raspberry Pi. This will help guide our design process and add to the potential systems for prototyping. [6]

2.2 Barcode and QR Code Scanners

Barcodes have been in use for many years now and are also known as UPC bar codes. The barcode is a fairly straightforward tool used in tracking and identifying items. Since almost any food that requires cooking is accompanied by a UPC bar code this is a great way to identify the food items we wish to keep track of. In order to scan a UPC code we can either use a laser scanner as is common to commercial use in stores everywhere in America, which we did not use, or we could use a camera to capture an image of the barcode and use the captured image to derive the identifying information, which we implemented in this project.

QR codes are a newer identifying piece of information that accompanies some items. In addition to identifying the food item, QR codes also contain other information that can make connecting to the internet unnecessary. The main difficulty with QR codes is that they require a camera to capture the image and then process the image in order to derive the information given in each code. This can be a fairly difficult process to implement but is used widely as well. These technologies are discussed in much more detail in the Hardware and Software Details section.

2.3 LCD Touchscreens

LCD touchscreens have become very popular in today’s world. LCD screens are everywhere from TVs to kitchen appliances to our smartphones. Due to the common use of these technological tools we find it desirable to integrate one into our smart microwave. LCD touchscreens offer a variety of special features that traditional touchpads do not such as low power usage, visual appeal, and re-programmability. It is for these reasons that NOMS chose to use an LCD touchscreen. In fact, these user interfaces have also become relatively cheap and easy to buy, making them a good choice for NOMS.

The only concern for our group was that oftentimes LCD touchscreens have their own controller that requires its own programming. As this was the case, we added to the workload of the design team. We needed someone to design the LCD touchscreen subsystem and integrate it with the rest of the system. It was in our best interests to find a touchscreen that uses a programming language similar to one that the group is familiar with such as C or JAVA, but this was not the case in our project. The LCD touchscreen we purchased came with its own coding language which we needed to learn before we could use the system. This topic will be further explored in the Hardware and Software Details section.

3.0 Strategic Components

3.1 Single-Band vs. Dual-Band Wi-Fi

When the group originally decided to create NOMS we decided that it would have to make use of a wireless internet connection in order for it to be cutting-edge. We immediately began researching embedded systems with onboard Wi-Fi. As we researched we became aware of a phenomenon in which microwave ovens have been claimed to interfere with Wi-Fi. Traditionally Wi-Fi uses the 2.4 GHz frequency band which is usually reserved for that particular device. Microwave ovens, however, actually sweep across various frequencies briefly causing serious interference and data errors across Wi-Fi. This sounded like a pretty major problem for the NOMS team. We discovered that a quick solution might be to get a dual-band Wi-Fi system so that we could avoid interference while the microwave was on. A few weeks later we decided that instead of switching from single-band to dual-band Wi-Fi we could simply change the way that NOMS operates. Since Wi-Fi can be interfered with by the microwave due to the magnetron’s waves, we simply needed to make sure that the Wi-Fi does not do anything important while the microwave is on. Any messages or information gathered from the online database will need to be done BEFORE the microwave is turned on. Once the microwave is turned on the Wi-Fi needs to be in a neutral state waiting for its turn to function again. If ever NOMS were to include short advertisements on the LCD we would simply need to switch over to the dual-band Wi-Fi.

3.2 Embedded Wi-Fi vs. Installed Wi-Fi Shields

Another initial idea the team had was to start the project off with an embedded system that already had Wi-Fi built into it. This would simplify the project to some degree, but with four group members instead of three we decided it would over-simplify the work too much and wouldn’t challenge each member enough. This conclusion was reached mutually after discussion with Dr. Richie who also encouraged the team to build the embedded system instead of simply buying one. This lead to the decision to forego using only a development board as a solution for the final project. We did, however, decide to prototype NOMS at the early stages of development in order to have parallel tasks running at the same time. With a working prototype we were able to test the database management system and all power systems running to the embedded system while the SOC is being designed and developed by the team.

3.3 Single vs. Multiple Embedded Systems

One thing the NOMS team wanted to do in order to keep good testability and maintainability throughout the project was to keep tasks and subsystems as modular as possible. In order to achieve this modularity we decided at first to use multiple embedded systems: one for handling all Wi-Fi and database related tasks and the other system to handle all other tasks. The former would actually communicate with the latter which would be the main SOC. Though this compartmentalized the work to some degree it actually separated the group too much in workload and made the task much more difficult to complete, especially after deciding not to use a microcontroller like an Arduino. The conclusion of the matter is that we have decided to run all tasks through one self-designed System on Chip. This encouraged more teamwork in the group, higher productivity, and a greater probability for success.

4.0 Possible Architectures

In considering multiple choices for architectures to use in the embedded system (SOC) we considered MSP430 and ARM.

The MSP430 was initially attractive because each member of the group had actually worked with the MSP430 in a previous class. Programming the MSP430 would be something we could do rather easily having a fair amount of experience working with it. This is both a good and a bad reason to choose MSP430 since we want to be able to achieve our goals but also be challenged and put real work into NOMS.

ARM, by contrast, was a solution that the group was not too familiar with. But as we researched more about ARM’s architecture we realized that it offered more flexibility than the MSP430. There are more options in the ARM architecture than there are in MSP430’s architecture just because of the design itself. This was important since the NOMS system isn’t completely and finally designed yet and we considered the probability that the design would be changing as we implemented it. This being the case flexibility was very important to us with architecture. In addition we learned in our research of ARM that it is a highly used industry standard that would give us valuable insight and experience in the world of microcontrollers.

In conclusion, we decided that using ARM architecture would be our best solution. Based on this conclusion we decided to seek out a microcontroller of this kind.

Project Hardware & Software Details

1.0 Power Management

1.1 Power Systems

The main objective of the NOMS power system was to provide low voltage power to the clock, controllers, cameras, and Wi-Fi, as well as ensure we supplied high voltage to the magnetron, fan, the buzzer, light, turntable, transformers for the magnetron and low voltage loads. Typically utilities provide 120 V at residential wall outlets. The utilities are mandated by Public Service Commission to maintain the voltage at +/- 5%, which means the voltage runs from 114 V to 126 V. The Public Service Commission is in charge of regulating the electricity, natural gas, water and waste water, and telecommunications for each state. As far as electricity is concerned, the Public Service Commission is responsible for everything from providing reliable service to making sure they are in compliance with new construction safety codes for distribution of power. With this in mind, we can expect to design this project with consistent 120 V AC power supply coming from the wall.

For this project we used 120 V as our base power source for incoming power and high voltage side of equipment, and selected 5 V and 12 V DC for low voltage or control voltage. The 5V system is commonly used in small appliance circuits. The high voltage devices, including the magnetron, fan, lights, and turntable, high voltage transformer and AC/DC transformer are designed to operate at 120V and will not require voltage regulation.

For residential service, utilities provides 120/240 V circuit A typical house has 150amps – 200amps service that is either overhead or underground. Often multiple outlets are designed and connected to either 15 or 20 amp circuits. The National Electric Code (NEC) allows contractors and designers to connect up to 6 outlets to a single circuit. Most house hold outlets are rated at 15 amps due to cost saving measures by the contractor. Occasionally, one would encounter a 20 amp outlet in a residential home. [7]

The NEC is a group of rules and policies that govern the installation of electrical devices for residential and commercial projects. The NEC is recognized nationwide and adopted by all state and municipal governments. In the state of Florida, we have adopted the 2010 Florida Building Code, which is the governing document that is used to establish the laws for residential and construction in Florida. Within the 2010 Building code, we also adhere to the 2008 NEC Building Code. [7]

The NEC also requires each kitchen to be served by two separate circuits. Often in a new construction a separate circuit is provided for a microwave; however, in older homes there is not a designated circuit for microwave. This means that the unit will be connected to an existing kitchen circuit along with other kitchen appliances. If the circuit is not connected properly, it can cause an overload in kitchen circuits. [7] Figure 3 describes a typical kitchen electrical layout, adhering to the description listed above.

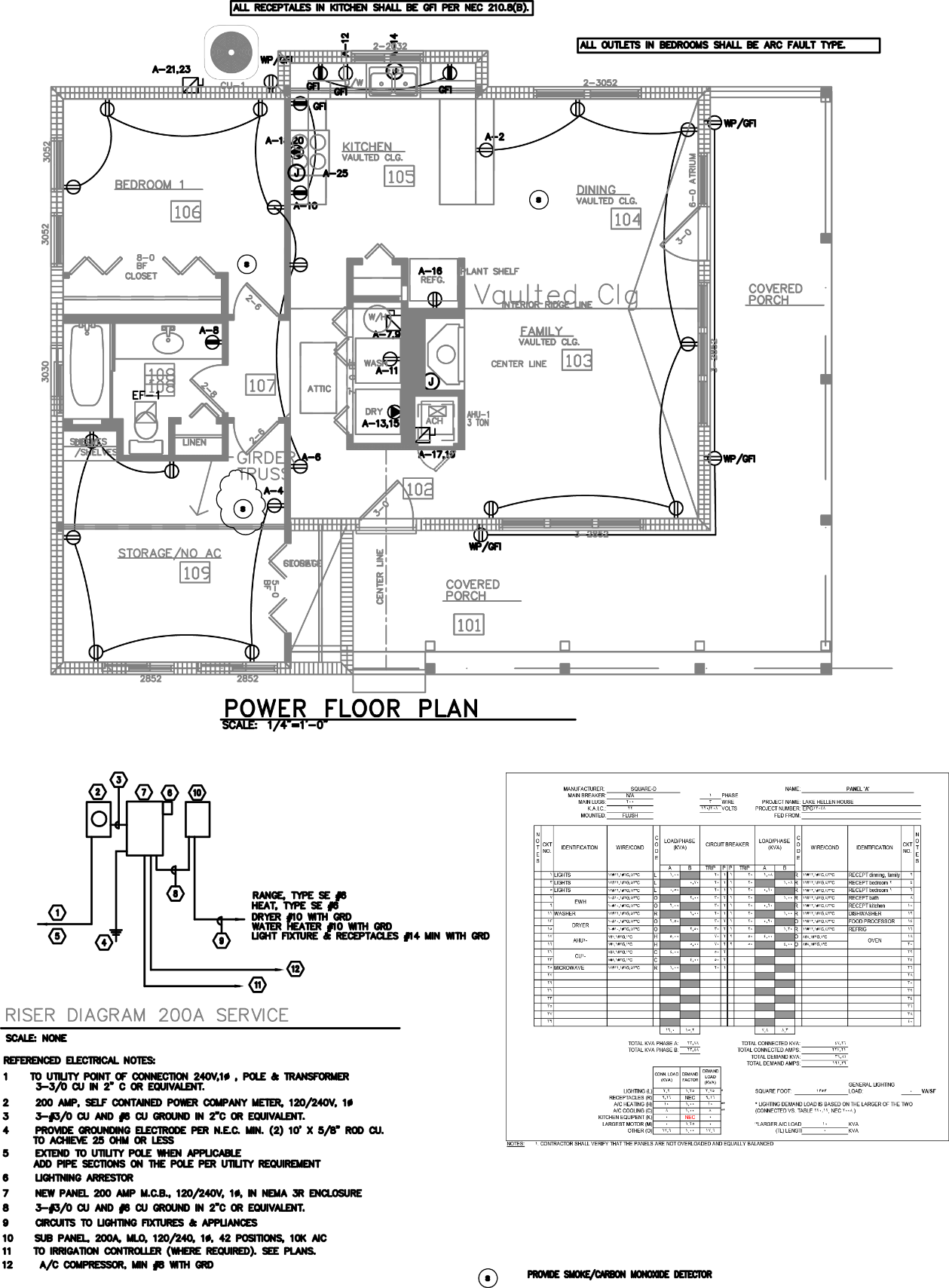


Figure 3 Typical Kitchen Electric Layout

(Figure Obtained With Permission by EPG Engineering)

The 120V wall outlet voltage is RMS value for the voltage. The peak voltage is RMS value multiplied by square root of 2, which is 170 V. This is the voltage that applied to all high voltage devices, including the AC/DC transformer. The resulting output of the transformer is 17 V AC that is fed through rectifier to create 17 V DC output. The 17 V DC flows through the regulator, where the 17 V is stepped down to 5 V.

1.1.1 Basis of Design

Food manufactures are challenged to produce a product that meets the majority of their customers’ need. This challenge is further complicated with the fact the each customer does not own a standard power output microwave to cook their meal. Therefore cooking instructions must reflect the assortment of customers whose microwaves range from using a low power microwave of 600 W to a high output power unit that produces 1250 W. In order to meet safety requirements as well as customer’s needs, most microwave food companies have established a standard of 1100 W as the basis of cooking instructions. This creates a new challenge for most customers, as often times the microwave food does not include alternative cooking methods for microwaves which do not fit the 1100 W requirements. These consumers must take out a calculator and determining the unit conversion to calculate the correct cooking time for their appliance for each specific microwavable food.

For our project we have created the NOMS system with the necessary intelligence to prevent customers from dealing with complications associated with guessing the time conversion. In order for the look-up data to accurately reflect cooking times, we designed our microwave so it was capable of converting the cooking times to account for the 1100W output instructions. With 15 A maximum circuit capacities, all design decisions accounted for the power restrictions of the microwave so we did not overload existing circuits.

All small appliances must be designed to operate with 15amp circuit, meaning that total power consumption must be less than 1800 watt (assuming 120v power base). In our design, we limited our total design watt budget to 1680 watts or 14 amps. We determined the overall watts budget by grouping the typical loads into two groups: low voltage needing 5 volt, and high voltage needing 120v.

1.1.2 Food Preparation Research

In order to determine the amount of output power NOMS required, we went to the grocery store and sampled some of the common frozen food items to see the required cooking wattage. By and large it seems that companies are creating cooking instructions to cater to people with greater than 1000 W microwaves. There were a few cases which listed alternative wattage cooking instructions, but they were in the minority. NOMS was designed as a product whose selling point was marketing the automatic preparation of food; therefore, we chose to comply with the common practices of the industry. Since most of the data we collected about microwave dinners calls for greater than 1000 watts to cook the food, we implemented this in our design.

The main concern of NOMS as an autonomous microwave was to prevent instances where frozen food was not cooked thoroughly enough and as a result, our consumers would ill from food sickness. This was a valid concern, as most of the frozen dinners found in the supermarket contain some form of meat. If the meat is not heated to proper temperatures, the consumer can fall ill due to salmonella or even E.coli. In October 2008, NBC and the Associated Press covered a story regarding an increasing problem with consumers eating undercooked microwave food. The article cited 325,000 people each year that are hospitalized from food illness. The article addressed an outbreak in salmonella and E.coli in frozen dinners within the past couple years and stated that corporations have increased their mandatory cooking power required for microwaved food as a results of these outbreaks of illness. [8]

This increase in mandatory cooking power was reflected in the data we collected in the supermarket concerning preparation instructions for frozen dinners. The data provided in table 2 comes from frozen food package cooking instructions we found during our trip to the supermarket.

|  |  |
| --- | --- |
| **Food** | **Cooking Instruction Required Microwave Wattage** |
| Garden Lites Carrot Raisin Souffle | 1000 W |
| Kashi Chicken and Chipotle BBQ | 1100 W |
| Jimmy Dean Biscuit, Sausage, Egg & Cheese | 1100 W |
| Boston Market Swedish Meatballs | 1100 W |
| Healthy Choice Chicken Parmigiana | 1100 W |
| Lean Cuisine Chicken Teriyaki Stir Fry | 1100 W |
| Marie Callender’s Sweet and Sour Chicken | 1100 W |
| Smart Ones Vegetable Fried Rice | 1100 W |
| Stouffer’s Chicken Lasagna | 1100 W |
| Brown N’Serve Lite Original Sausage Links | 1100 W |
| Moringstar Chik’n Nuggets | 1100 W |
| Kidfresh Muy Cheesy Quesadillas | 1100 W |
| Ethnic Gourmet Chicken Korma | 1000 W |
| Health is Wealth Spinach Munchees | 1000 W |
| Boca Savory Mushroom Mozzarella Veggie Patties | 600 W – 1100 W |
| Stouffer’s Veal Parmigiana | 1100 W |

Table 2 - Frozen Food Research

As clearly displayed in table 2, it appeared that most of the cooking instructions called for over 1000W output power in microwaves. With that information in mind, we decided it would be prudent to determine how common it would be for a consumer to purchase a microwave under 1000W. If most microwaves for sale currently had over 1000 W output power, we would not consider designing a microwave with less output power. We visited a local electronic store to see what wattage models were for sale on the floor. From this trip, we determined that people can purchase a microwave from 700 W to 1200 W. Typically the 700 W microwaves are well below $100 while the 1200 W microwaves tended to be pricey. Considering the number of options below 1000W output power, it is safe to assume that not all households would purchase microwaves with over 1000W output power. Therefore, NOMS was designed to work at multiple output power wattages. We chose to control the magnetron via the microcontroller by implementing duty cycles. This allowed us to cook at different power levels. The data in table 3 was obtained during our visit to the electronic store. The chart shows some of the microwaves available for consumer purchase.

|  |  |
| --- | --- |
| **Microwave Brand, Model, Output Power** | **Price** |
| Oster, OGG3701, 700W | $79.99 |
| Haier, HMC0903SEES, 900W | $99.99 |
| Franklin Chef, FC1270W, 1000W | $109.99 |
| GE, PEB7226SFSS, 1100W | $339.99 |
| Sharp, R651ZS, 1200 W | $189.99 |
| KitchenAid, KCMS2255BSS, 1200W | $539.99 |
| Whirlpool, WMC50522AS, 1200W | $269.99 |
| Panasonic, NN-SE982S, 1250W | $309.99 |

Table 3 - Microwave Data

The prices above reflect standalone microwave units and not the ones that are installed into the kitchen wall, otherwise known as “Over-the-Range”. “Over the Range” microwaves cost substantially more and were not considered for this project. It is possible that a consumer who is interested in purchasing a microwave will consider microwaves with less than 1100W output, due to cost. With the consumer in mind, we decided to create a conversion table for microwave cooking times. This was useful for the cases where the microwave output wattage does not agree with the cooking instructions. This feature makes NOMS a more universally applicable system and increases our appeal to the consumers.

In order to determine a conversion equation, we examined some products which listed cooking instructions for multiple output cooking power wattages. From this information we were able to find a relationship between the times and the output power wattage. The equation is as follows:

Logically, this makes sense, as having a lower output power would require a longer cooking time. This equation was not considered perfect; however, it was a good place to start. We tested the accuracy of this equation while we tested the microwave. Additionally, we intended to design a feature built into NOMS where if something in the cooking instructions must be changed, NOMS would have the capability to remember the changes and implement the new instructions the next time that item is placed into the microwave. We also wanted NOMS to have the ability to remember user presets. For example, if someone likes their popcorn a little less popped, the next time NOMS read that exact type of popcorn, it would remember the previously entered user settings and implement the new cooking instructions instead of referencing the manufacturer’s instructions. However, we were not able to implement these reach goals in the final product. As we designed the NOMS system we ran into limitation with RAM and realized that we would not be able to successfully implement these features. Going forward, we would be able to implement these features if we went back and chose different LCD and microcontroller components. If we ever chose to commercialize NOMS, these would be things we would consider developing further.

1.1.3 Heating Food with the Microwave

The magnetron produces microwaves by converting the current into radio frequency waves. The radio waves produced by the magnetron are sent towards spinning metal fans inside the microwave. Microwaves will reflect off metal, so this will cause the microwaves to reflect off the fan and into the cooking chamber. A typical microwave cooking chamber is lined with metal for the same design purpose: to keep the microwaves bouncing around the chamber until they are ultimately absorbed.

Microwaves are absorbed by water. Water is a polar molecule, which means one side is predominately positively charged while the other is predominately negatively charged. Indrocucing an electric field to a polarized molecule will cause the molecule to rotate. The electric field produced by the magnetron is oscillating, which means the water molecule will consistently rotate. Most microwaves attempt to operate at 2.45 GHz. This frequency oscillates at exactly the same amount of time it takes for water to rotate 180 degrees. This is considered the ideal speed since water cannot rotate any faster. [9]

The rotating water molecules are constantly bumping into the molecules around them, causing a transfer of kenetic energy. Kinetic energy is defined as the amount of work needed to accelerate a body of mass from rest, or E = ½ mv2. Kinetic energy can also be defined as temperature, as is the case for water. Increasing the kinetic energy of water will increase the temperature of the water molecule. The increase of temperature, or kinetic energy, in the water molecule will increase the internal temperature of the food.

The electric field at any point on the food is equal to the sum of the respective electric fields. If two electric fields line up perfectly so the peaks and valleys are aligned, they will create a larger electric field. The increased electric field allows the water molecules to gain more kinetic energy. The gain in kinetic energy will cause the internal temperature of the food to rise, and ultimately, get hotter. Sometimes the microwaves can interfere with each other, which causes the food to cook unevenly. If the waves line up in such a way that the peaks and valleys are on top of each other, this will cause the two to cancel and the electric field will be negated.When the electric field is negated there will not be energy available to increase the internal temperature of the food. [10]

As previously mentioned, the inside of the microwave is coated with metal. This allows for the molecules to bounce around, but it also provides oppoprtunities for the microwaves to create interferance with one and antother. The purpose of a rotating turntable in the microwave is to reduce the amount of interference the microwaves can create.

Another factor which determines cooking time is the amount of water present in a material, which is directly related to the ability of the molecules to rotate. The easier the it is for the water molecules to rotate, the quicker the food can cook. Items that are frozen have less water molecules. This creates a condition in which it is harder for the molecules to rotate, which means there will be less kinetic energy created. Items that do not have many water molecules are also difficult to cook as there are less water molecules to rotate and ultimately less kinetic energy transferrance. [11]

Electrical Research

1.2.1 Linear Power Supply Design

In order to design a linear power supply you need a transformer to step down 120 V AC, a rectifier to change the AC voltage to DC, and a capacitor to smooth out the signal. The first thing required when designing a linear power supply is to determine the output voltage and current needed. In this system, we will need 5 V at 1.8 Amps. From this, we considered a linear regulator and bridge rectifier that would handle the current requirement for our system. Finally, we need to consider a transformer that created the desired step down voltage of 5 V. The research below was necessary to aid in the decision of what circuit components were necessary to create the linear power supply design for the NOMS system.

1.2.2 Full Wave Rectifier

A Full Wave Rectifier converts alternating current signal (AC) to direct current signal (DC). The rectifier is composed of a system of diodes and transformers. The nature of a diode makes it ideal to use for rectification, which is the act of transforming AC voltage into DC voltage. Diodes are ideal because they only allow current to flow from the anode to cathode, but not in the reverse direction. When dealing with a larger input voltage, it is advantageous to use a power diode. Power diodes have a high forward current capability (kA) and a large reverse blocking voltage (kV). Since this project will deal with the high voltage transformer inside the microwave, we will consider power diodes within the Full Wave Rectifier. [12]

A Full Wave Rectifier takes in an AC waveform, like the one pictured in figure 4.



Figure 4 - AC Waveform

It then converts both the positive and negative inputs into a series of pulses, which means that the voltage seen by the load has less fluctuation within the signal. The negative signals are flipped onto the positive signals, creating the output pictured in figure 5.



Figure 5 - Full Wave Rectifier Output

1.2.3 Circuit Design Full Wave Rectifier

In order to achieve this output, a Full Wave rectifier is typically designed with two power diodes connected to a multiple winding transformer and load resistance. A multiple winding transformer contains more than one winding on the primary or secondary side connected by a common laminated core. The circuit diagram is pictured in figure 6.



Figure 6 - Full Wave Rectifier Circuit Design

The circuit contains two diodes so that one diode delivers current to the load while the other diode remains off. This allows for only the positive signal cycle to supply current to the load. Since the output voltage is the sum of the two waveforms, the spaces between the half waves are covered by the waveform of the other diode. The sum of the waveforms creates an output twice as large across the output resistor. [12]

1.2.4 Multiple Winding Transformers

The multiple winding transformers connect the primary windings to a high voltage or current supply and transform this input into a different secondary voltage. In order to use this kind of transformer, the voltage or current ratings must be identical in order to be connected. Figure 7 shows the design of a multiple winding transformer.



Figure 7 - Multiple Transformer Windings

The downfall of using a multiple winding transformer is a large transformer is needed to deal with the large output, which is expensive. It is also expensive to make sure the secondary windings are identical. A cheaper and more reasonable solution is to use a Full Wave Bridge Rectifier circuit, as this circuit does not require a center tapped transformer and it is also smaller circuit. [13]

1.2.5 Full Wave Bridge Rectifier Circuit Design

The Full Wave Bridge Rectifier circuit uses four diodes connected in a closed loop. The secondary winding is connected to one side of the diode bridge and the load is connected to the output. At any given time, two diodes are on for each half cycle. In the circuti diagram pictured in figure 8, D1/D2 are turned on for the negative cycle and D3/D4 are on for the positive cycle.



Figure 8 - Full Wave Bridge Rectifier Circuit

This circuit setup allows for a larger mean DC output value with less fluctuation. The capacitor shown in parallel with the load, known as the smoothing capacitor, helps create a pseudo DC output. The smoothing capacitor is determined by two factors: the working voltage needs to be less than the no load output of the rectifier and the capacitance value needs to be larger to eliminate the fluctuation on the voltage. [14] The resulting output becomes more stable, like the output in figure 9.



Figure 9 - Full Wave Bridge Rectifier Output

The downfall of this circuit is that this circuit does not work well with low voltage power; however, this will not be a problem as the NOMS microwave design deals with a high voltage power.

1.2.6 Voltage Doubler

Voltage doublers are used in AC to DC conversions. They take the input AC voltage and turn it into an output of the DC voltage twice as large as the input. A voltage doubler uses diodes as natural switches. Reference the previously discussed research in the Full Wave Rectifier section to see how diodes can be used as switches. Using a voltage doubler in a circuit allows for a smaller input power to be converted into millions of volts to power some of the high energy components inside the microwave, such as the magnetron. This works because the voltage doubler allows for a wider voltage oscillation frequency to occur from a lower input voltage. We considered using a voltage doubler for the power systems design in this project. We considered using a Villard Circuit in our power systems design, however this was not the final design decision we went with.

1.2.7 Villard Circuit

In most microwave power system design, a Villard circuit is used as a voltage doubler. A Villard circuit consists of a capacitor and a diode. The circuit is simple to design, but the designer must be wary of the high noise distortion which comes from converting the AC current signal to DC current signal. The way a Villard circuit works is that the circuit “clamps” the AC input at zero. It takes the entire input signal and shifts the negative values until the entire signal exists in the positive domain. This creates twice as large peak to peak values, since all of the values are positive. The reason this circuit is used in a microwave is because it allows the few input volts to be transformed into millions of output volts, which is necessary for powering the magnetron. When this circuit is used in a microwave oven, the direction of the diode is reversed so it can supply negative voltage. [15] The circuit design for the Villard circuit is depicted in figure 10, with the output of the circuit is included in figure 11.



Figure 10 - Villard Circuit Design



Figure 11 - Villard Circuit Output

1.2.8 Electric Motor

An electric motor is a device that converts electical energy into mechanical enegy.

1.2.9 Induction Motor

An induction motor is an alternating curent (AC) electric motor. In an induction motor, current flows through wire loops that are placed around the rotating armature. The three phase power supply creates a rotating magnetic field which turns the motor. In an induction motor the rotor rotates at a slower speed than the stator field, which means that the magnetic field creates a current in the opposite direction, causing a magnetic flux. Induction motors require slip, meaning the rotor needs to rotate slower than the AC curent , to create torque.[16]

1.2.10 Synchronous Motor

Synchronous Motors function at a steady state. They are AC motors which contain electromagnets on the stator. The rotation of the shaft is synchronized with the frequency of the supply current. [16]

1.2.11 Brushed DC Motor

Brushed motors contain permanent magnents and a spinning armature. The permant magnents are a part of the stator, or stationary part of the motor. The spinning armature is part of the rotor, or moving part of the motor. The motor runs off a direct current power source. When current runs through the coil on the armature, it acts as an electromagnet and produces a magnetic field. This field attracts and repels the magnets in the stator which causes the armature to spin. However, this will only allow the armature to spin 180 degress unless the polarities are changed on the electromagnet. The brushes change the polarity by making contact with spinning electrodes attached to the armature. [16]

There are a couple issues with this design. First, the brushes will wear out. The brushes are also restricted by the maximum speed of the motor. Brushes also restrict the amount of poles the armature can have. Since there is an electromagnetic field in the center of this motor, the system is prone to overheating and is hard to cool. This will not work within a microwave. Finally, the brushes are in constant contact with the walls of the motor, which creates sparking and electical noise.[16]

Sparking occurs when potential between the electrical field and the area around it is so great that there is an incease of free electons flying between the brush and commutator. The electric field charges the air, creating a conduction field which interracts with the electrons and ions to create a dielectric breakdown. A dielectric breakdown is when the voltage of an insulator exceeds the breakdown voltage. The excess energy charges the insulator and makes the mateial conductive. [16]

1.2.12 Brushless DC Motor

Brushless DC Motos are synchronous electric motors powered by direct curent. Unlike Brush DC Motors, Brushless motors operate without the use of brushes. Electromagnets are placed on the armature and do not move. The pemanent magnets rotate around the stator. This geneates torque, which powers the motor. Unlike induction motors, the stator and magnetic field in a synchronous and brushless motor generate the same frequency. This design prevents slippage, which is often a problem with induction motors. [17]

In a Brushless DC Motor, the permament magnets are attached to the rotor and the electromagnets are attached to the stator. While the shaft turns, computer controlled transistors are used to charge the electromagnets attached to the stator. The computer allows for better motor control. Since there are not bushes to wear out, the motor is sparks less and has less electrical noise. Another benefit of this design is that since the electromagnets are placed on the stator it is easier to control the motor as well as cool the motor down, since the rotor does not need any power.[18] For the reasons listed above it appears more prudent to use Brushless DC Motors with this project.

1.2.13 Stepper DC Motor

A Stepper motor converts electrical pulses into mechanical energy. When hit with an electical pulse, the spindle of the motor rotates in step increments. The Stepper motor breaks up a complete rotation into individual step increments. The sequence of the electrical pulses is important in determining the direction the motor shaft rotates. The speed of rotation of the shaft is also directly linked to the frequency of the pulses. The length of rotation of the motor is directly impacted by the number of electrical input pulses. The Stepper motor energizes individual electomagnets using a microcontroller. In order to turn, the first electromagnet gets a pulse of power. This magnetizes the gear’s teeth, which causes the gear to turn to align with the next electromagnet. During this process, the gear teeth are discharged and ready to interract with the next electromagnet.[19]

Some benefits of using a Stepper Motor is that the motor still has torque even if it is not rotating. The motor itself is very responsive to start and stop commands. It is also able to reverse easily. The motor does not contain brushes, which eliminates electrical noise as well as part maintance. The motor has open loop control, which means that no feedback information about the position of the motor is needed. This reduces costs and makes the motor easier to control.[20]

A Stepper Motor was ideal for this project since we used it to control the turn table in the microwave. A Stepper motor allowed for control of rotation angle, speed, position, and synchronism. This gave us percise control of the turntable motor.

1.2.14 Magnetron

A magnetron is considered a self-exciting oscillator which produces a magnetic field and electric field at the same time. This allows the magnetron to function like a pulse width modulator. The magnetron produces the magnetic field and the electric field perpendicular to one and another so they can produce a high power output. This can generate a frequency from 600-30,000 MHz. A magnetron will often only work at a fixed frequency, which could be a problem in some situations but happens to be ideal for powering a microwave machine. [21]

A magnetron consists of an anode, cathode, resonant cavities, filament leads, and a pickup loop. The anode is created out of solid copper block and serves as the positive element within the tube. The cathode, which is considered the negative element within the tube, is supported by filament leads which help keep structure as the magnetron heats up. The resonant cavities help control the output frequency of the magnetron. The electric and magnetic fields generated by the magnetron exert force upon the electrons. [21]

The magnetron is activated once current flows through the cathode and filament. Once the filament sees the current it will release electrons which travel throughout the space between the cathode and anode. Since the electrons near the cathode have negative charges, they are attracted towards the anode. Current is created around the cavity causing the cavity to act like an inductor. Once the electrons start to move away from the cathode, they are pulled by the permanent magnetic field which is perpendicular to their original path. The magnetic field is created by the permanent magnet surrounding the magnetron. The magnetic field bends the electron’s path from a straight line to a more radial path. The stronger the magnetic field, the more the path will bend. The electron’s path will also bend if the velocity of the electron increases, which will cause a crease in the field surrounding the electron. [22]

The electric field in the magnetron is created by AC to DC field interaction. The DC fields exist around the anode and spreads to the cathode. The AC field exists in the spaces between the cathode and resonant cavities. The electron spends energy entering the resonant cavities as it tries to reach the anode, creating oscillations. The oscillation of the charges within the cavities creates radiation of electromagnetic waves, which create the energy needed to cook food. [23]

1.2.15 Power Supply Stability

Power stability is determined by a power acceptability curve called the CBEMA curve, more recently renamed as the ITIC curve. CEMBA stands for Computer and Business Equipment Manufacturers Association. It is the standard engieneers must hold themselves to while designing power supplies. The curve is designed to help engineers keep their power design within bounds of acceptable power for a system. Engineers use the CBEMA curve to report on power quality. Power quality consists of problems such as voltage fluxuations and current disturbances. The rise in concern over power quality is what lead to the invention of adjustable-speed motor drives and shunt capacitors in order to reduce losses and aid in power factor correction. [23] The CEMBA curve is displayed in figure 12 on the left, and the ITIC curve (which will be discuessed later) will be displayed on the right.

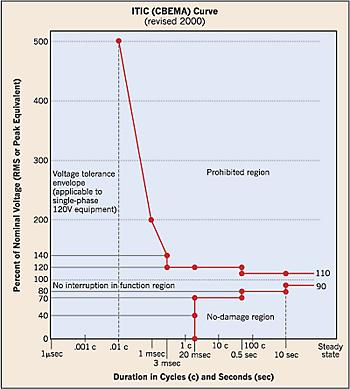
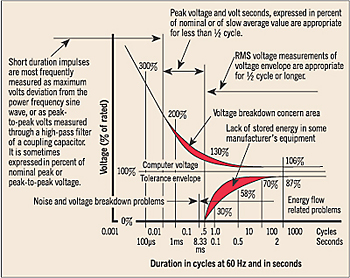


Figure 12 - CEMBA and ITIC Curve

These curve shows the relationship between the percent of voltage applied to the circuit and how it is impacted over time. Data which lies between the two curves is considerd as operating at an acceptable level. Data points which lie above the curve are considered voltage values which could lead to malfunctions within the power systems, like an overvoltage trip. Data points which lie below the curve are consdiered inadequate because the voltage is not creating sufficient energy to continue carrying the load, whivh would cause the power system to fail. [23]

The CBEMA curve was renamed in 2000 to the ITIC curve (after the Information Technology Industry Council. The reason for the new curve was to more accurately reflect the changes in technology with computers, fax machines, copiers, etc. The difference between CEMBA and the ITIC curve is that the ITIC curve has an expanded acceptable power area. The machines needed to check against this curve are also simpler to design, due to the new curve shape.This curve will be helpful for this project, as our system operated at the 120/240V 60 Hz range that the ITIC curve represents.

1.2.16 Power Output Requirements

Conducting a brief survey of microwave food, it seems that most manufacturers have standardized their cooking instruction using 1100 w power output. Variation in cooking time can produce uncooked food, or cause health issues to consumer. The microwave communicates via Wi-Fi to a database in order to pull cooking instructions. In order to successfully design the power supply for the NOMS system, we considered the overall electrical load requirement.

In the sections below, we will discuss the individual components of the high and low voltage circuits. We considered the voltage, amperage, and wattage of each component, as well as the price. All of these factors were discussed so we were able to select the most efficient and cost effective component for our power system design.

AC Components

1.3.1 Magnetron

Typical magnetron output range is from 600w to 1250w. The typical input voltage for a magnetron ranges from 4.1 kV to 4.35 kV. The magnetron is the largest component of the power system supply we will need to consider, and also the most important. The magnetron is the component responsible for generating magnetic waves which will ultimately cook the food in the microwave. For the purposes of this project, we entertained the idea of purchasing a magnetron for the NOMS design. The data in table 4 lists our top choices for magnetrons of varying voltages and output wattages. [24]

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial Number** | **Voltage** | **Current** | **Watts** |
| 10QBP0303 | 3.8 kV | 0.132 amps | 500 W |
| 10QBP1002 | 4.1 kV | .171 amps | 700 W |
| 10QBP1009 | 4.35 kV | 0.230 amps | 1000 W |
| 2M261-M32KLP | 4.35 kV | 0.287 amps | 1250 W |

Table 4 - Magnetron Power Data

The prices of these magnetrons range from $60- $180. For this project were interested in purchasing a minimum of 1000 W power output magnetron, which in the chart above is listed as the 10QBP1009. This magnetron costs approximately $75, so we can expect to spend at least that much if we can obtain this part for our microwave. If not, the next best part is the 1250 W power output, labeled 2M261-M32KLP in the table above, which will cost approximately $85. The difference in prices of the two magnetrons was enough to warrant picking one over the other, considering that we created a conversion table for cooking times.

The main factor in determining which part we chose to use in this project was current. The 1000W power output magnetron provides us with the necessary 1000 W minimum output requirement, while using less current than the 1250 W output magnetron. This could be a crucial design factor for the power systems, as we had a limited amount of current to distribute. By picking the magnetron which still has a high power output but also includes less current pull we were able to allot more power to the nonstandard components in our design, such as the cameras, touchscreen interface, and Wi-Fi.

1.3.2 Turntable

The turntable is required to heat the food evenly. The turntable rotates the food to counter the bouncing magnetic waves which are interfering with one and another, which preventing uneven distribution of microwave energy. This also allows the different particles that absorb the microwave energy more slowly than others to become excited and increase in temperature. There were the three turntables for consideration. They are found with stats in table 5. [25]

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial Number** | **Voltage** | **Current** | **Watts** |
| F63265G60AP | 120 V AC | 0.033 Amps | 4 W |
| 6549W1S011M | 120 V AC | 0.025 Amps | 3 W |
| 15QBP0851 | 120 V AC | 0.017 Amps | 2 W |

Table 5 - Turntable Power Data

The motors we were considering range in price from $20 to $33. The difference between the 3W motor and the 4 W motor is $7. As this is not much of a fluctuation between prices, we did not consider price as a determining factor. However, it seems that each of these motors have varying wattage and current. We took this into consideration for the design of our power systems. From the chart listed above, we decided to use the 15QBP0851 motor, since it has the lowest wattage and amps required of the three motors. This allowed for more freedom in power design, while still getting the job done.

1.3.3 Fan Motor

From research, we determined that a large amount of the power dissipated in the microwave system will be expelled as heat from the magnetron. The purpose of the fan in a microwave is to reduce the heat and cool the system. This will keep the other components in the microwave from overheating. It also eliminates the steam so the electronics do not get moist, thus shutting down the microwave. The fan is important to keep the other AC components running at full power. In table 6 we listed our three fan choices, as well as the information pertaining to these choices. [26]

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial Number** | **Voltage** | **Current** | **Watts** |
| AMM13-003ZSE | 120 V | .8 Amps | 96 W\* |
| OEM-10DWX1-A07 | 120 V | 0.3 Amps | 36 W\* |
| WB26X5061 | 120 V | 0.7 Amps | 39 W |

Table 6 - Fan Motor Power Data

In the two entries listed with (\*) above, we were not provided the exact wattage of the fan. In those two instances, the wattage was determined by multiplying the voltage times the current. We made this distinction because when we researched the third component, we were actually able to find the wattage listed on the motor. The wattage on this motor was most likely limited, meaning there is most likely internal resistance to prevent the motor from spinning too fast. We should assume the other two components would also have internal resistance, so the data provided above did not exactly reflect the proper wattage; however the information provided in the table above represented a theoretically correct assumption.

With that being said, we considered the WB26X5061 for the NOMS power design. We could not be certain about the exact wattage output of the other two fan motors. Like previously stated, it was very difficult to research the values of the fan motor. With our limited knowledge, we chose the fan motor in which we can account for all the relevant data. The WB2X5061 goes for $33.

1.3.4 Lights

The purpose of a light within a microwave is mostly aesthetic; however, in our project the lighting potentially bore some importance, especially if we were going to utilize the inside camera. We needed adequate lighting in the microwave chamber so the camera can read the QR code. A brighter light would create a better dynamic range of values for the images captured by the camera. This would allow easier isolation of the QR code from the rest of the image. The light needed to work while the food was cooking, any time the door was open, and when the food item was initially placed within the chamber so the internal camera could capture the image. The information in table 7 shows three of the most common microwave lights that we considered using for the NOMS system. The three lights are provided with stats, in table 7. [27]

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial Number** | **Voltage** | **Current** | **Watts** |
| WB25X10019 | 120 V | 0.167 amps | 20 W |
| 6912W1Z004B | 120 V | 0.25 amps | 30 W |
| 8206443 | 120 V | .333 amps | 40W |

Table 7 - Light Power Data

The lights considered in the table above range in prices from $3 to $5. The cost of the light was fairly inexpensive and only within $2 of each other, so this was not a deciding factor in our decision. The most important aspect to consider was the wattage. We wanted to select a light that was fairly bright, so the camera could accurately capture the QR image. Based on the lights we researched, the 40 W bulb was the best selection for the design of the NOMS system. As time went by and we lost a group member we decided to drop this aspect of the project, as the task of implementing two cameras was too great for a three person group to complete in the duration of a single semester.

1.3.5 AC/DC Transformer

The AC/DC transformer works to not only convert the wall power into something the magnetron can use, it also helps step up the voltage part of the way to reach the large required input voltage of 4.35 kV. The transformer does a lot of the high voltage work. With the help of the voltage doubler, this component plays an important part of the high voltage power design. There were three transformers we were considering after our preliminary research. The transformers and their stats are listed in the table 8. [28]

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial Number** | **Voltage** | **Current** | **Watts** |
| 16QBP0293 | 120 V | 5.833 amps | 700 W |
| 16QBP0296 | 120 V | 8.333 amps | 1000 W |
| F606Y8M00AP  (Inverter Board) | 120 V | 10.417 amps | 1250 W |

Table 8 - AC/DC Transformer Power Data

Of the three transformers listed above, the pricing was almost identical across the board, approximately $145. The only difference was that the F606Y8M00AP costs slightly more at $155, but with the ten extra dollars you are getting the entire inverter board. This simplified design eliminates some of the other components we would need to consider, such as a heat sink and voltage regulators, and also packages it in an efficient manner. We considered using this component for the purposes of the transformer for the NOMS power design.

1.3.6 Buzzer

The buzzer in the microwave is used to alert the consumer as to when the food is prepared. Most microwaves use one type of buzzer, called Piezo electric buzzers. These buzzers use less current and occupy less space than traditional buzzers, which is why they are very popular in microwave design. Space and current are two limited resources in power systems design, and the more optimized something like a buzzer can be, the more freedom we will have with the more crucial parts. Table 9 contains three Piezo electric buzzers that operate at varying currents. [29]

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial Number** | **Voltage** | **Current** | **Watts** |
| PKM24SPH3805 | 3 | 0.012 Amps | 0.036 W |
| PKM24SPH3807 | 10 | 0.010 Amps | 0.1 W |
| PKB24SPCH3601 | 15 | 0.016 Amps | 0.24 W |

Table 9 - Buzzer Power Data

From the choices listed above, we chose the PKM24SPH3807. A buzzer brings minimal value to the microwave overall. It is mostly there to alert people in the event that they walk away from the microwave that their food is prepared. It does not really add to the cooking design of the microwave, so we chose the component that needed the least amount of current. The amount of voltage that the buzzer required was not a problem, as we already designed 12 V from the regulator. Therefore, we considered the component that pulls the least current. As we ended up purchasing a LCD interface that included a buzzer, this design consideration was not implemented in our final design.

1.3.7 Summary of High Power Consumption Devices

See table 10 for all individual lower power load consumption.

|  |  |  |  |
| --- | --- | --- | --- |
| **Devices** | **Voltage** | **Current** | **Power Consumption** |
| Magnetron | 4.1 kV | 0.171 amps | 1000 W |
| Fan | 120 V | 0.7 amps | 39 W |
| Light | 240 V | 0.0833 amps | 20 W |
| Turntable | 120 V | 0.025 amps | 3 W |
| Buzzer | 10 V | 0.010 amps | 0.1 W |
| **Total** |  |  | 1062.1 W |

Table 10 - Summary of High Power Consumption

1.4 DC Components

The DC components were generated by using the rectifier circuits and transformers discussed below. Our objective was not to exceed the budget wattage of 1680 W. We were mindful of the wattage consumed by the high voltage components. All of the low voltage components were selected to satisfy the budget minus the 1062 W consumed in the previous part. After selecting our components, we verified the capacity of existing transformers and components of the DC system to ensure adequate capacity exists by summing the energy consumption of the individual components.

In this section, we needed to collaborate as teammates. Components were chosen with both power systems design specification in mind and on top of that we needed to satisfy adequate number of serial interfaces and sufficient processing power to handle image processing. These components include the controllers, camera, and touch screen. Therefore, the previously listed components include the resulting decision of the collaboration between team members, as the power design consideration of these components remains secondary to the other design specifications.

1.4.1 Controllers

When the consumer uses the touch screen, a signal is sent from the touch screen to the microcontroller. The microcontroller takes the input and sends the signal to the microwave to activate the relay switches which indirectly allows for the magnetron to turn on and off. The microcontroller also works with the Wi-Fi chip, which it uses to send signals back and forth with the online database to pull cooking instructions.

The microcontroller we chose for this project is listed with its stats in table 11.

|  |  |  |  |
| --- | --- | --- | --- |
| **Identification** | **Voltage** | **Current** | **Watts** |
| Tiva TMC123G Launchpad | 5 V | 0.255 Amps | 1.28 W |

Table 11 - Microcontroller Power Data

After consulting with the other members of this team, we decided that the Tiva TMC123G Launchpad was the best microcontroller for the NOMS system. Since we only had 1.8 amps to allocate for the low voltage components, we chose a Launchpad with the smallest current draw, while still being capable of processing all the data and instructions required.

1.4.2 Camera

The purpose of camera within the NOMS system was to identify QR codes inside the microwave and bar codes on the outside of the microwave. The camera had its own processor, which allowed for image clean up and desaturation. It also helped increase contrast before sending the images to microcontroller. The camera we researched and decided to use for this project is listed in table 12.

|  |  |  |  |
| --- | --- | --- | --- |
| **Identification** | **Voltage** | **Current** | **Watts** |
| CMOS VGA | 0.7 | 0.0429 Amps | 0.03 W |

Table 12 - Camera Power Data

The CMOS VGA camera was definitely a power friendly component. By only drawing 0.0429 amps per camera, this helped ease the strain on the amp budget for the touch screen, which was the most important low voltage component in the system.

1.4.3 Touch Screen

The touch screen was a very important component of the NOMS system. The touch screen allowed for communication to occur between the user and the microwave. The touch screen was also the low voltage component with the largest current draw. It needed the current to not only display the information on the screen to the consumer, but to also run advertisements. The touch screen also needed to communicate with the microcontroller. After much discussion, the microcontroller we chose as a team is listed with its specs in table 13.

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial Number** | **Voltage** | **Current** | **Watts** |
| µLCD-32PTU | 5 V | 0.155 Amps | 0.76 W |

Table 13 - Touch Screen Power Data

The touch screen was the main component that we chose to design around for the low voltage power supply. We made sure to limit the rest of our budget to ensure we had enough current leftover to power this device. The touch screen was our user interface and quite literally the face of our microwave system. The fact that we were able to find such a strong device without spending too much current was a win for the NOMS system. In hindsight, we would have chosen another LCD touchscreen, as we found out that this LCD did not have sufficient RAM to execute all the things we expected from the system.

1.4.4 Relay

Within the circuit board design, we needed to implement a switch to control the large load voltage. This was crucial for communication between the microcontroller and the switch from AC to DC voltage. For this purpose, we considered using a relay. A relay uses a wire wrapped around an iron core that is activated once electricity is applied to the wire. The wire turns magnetic, creating an electromagnet. This causes the switch to change positions, either electromagnetically or with the use of a mechanical component, allowing for certain terminals to turn on while other terminals are disconnected or vice versa. The reason relays are important is that they use a small amount of current or voltage to control a large load current or voltage. There were two possible relays to consider, solid state relays (SSRs) or electro-mechanical relays (EMRs). [30]

1.4.5 Solid-State Relays vs Electro-Mechanical Relays

A solid-state relay is a purely electromagnetic switch. It does not have any moving parts. This will create a more reliable microwave, as microwaves are used often and the repeated use would cause the moving parts to break faster than an electromagnetic switch with no moving parts. Additionally, the output resistance does not degrade from constant use.

This is not true of electro-mechanical relays, which tend to be larger in size due to the moving components necessary to achieve the same switching action. The amount of time required to relay the signal to move the mechanical part makes electro-mechanical relays slower to react than solid-state relays. In electro-mechanical relays the switch is activated by the control signal. That signal causes a mechanical part to move into place so the load can switch between AC and DC. Since the mechanical part must move there is more chance for malfunctioning as well as the switch seeing wear and tear. Additionally, as with any kind of mechanical attachments, we needed to be concerned about sparking. This kind of loss can only happen within an electro-mechanical relay. [31]

Solid-state relays are typically very small. This was important for the printed circuit board design aspect of our project, as you often pay more for having a larger circuit board. Another thing to consider is that the electronics of the microwave are very dense. Therefore, it was important for us to conserve space on circuit board design in any way possible. The solid-state relay design also allowed for surface-mounted technology which freed up more space on the printed circuit board design, which ultimately saved us money. This also made our circuit board easier to mass produce, something we wanted to design for in case NOMS should ever become a commercial product. [31]

A solid-state relay does not require driver electronics and allows for bounce free switching. This allows for uninterrupted stream of voltage to be delivered. Additionally, solid-state relays tend to have improved system life-cycle costs. This means that the design of a solid-state relay allows for simpler designs that also allow you to reduce input power as well as better ways to deal with heat dissipation, which is a serious concern with microwaves. Often times a microwave will pull in around 1100 W of power but only output 700 W of power due to mostly heat losses. Heat loss is the biggest concern of power design, and especially important aspect we needed to consider while we were designing the NOMS system.

Finally, it is most important with regards to power consumption to consider which switch will consume the least amount of power. An electro-magnetic relay must heat up a coil before the switch can be activated, typically consuming 80 mW of power. The solid-state relay must also heat up a coil, but only consumes 3 mW of power.

1.4.6 Relay Switch Selection Process

It was very important for our project that we chose a switching method that:

* Saved space on our printed circuit board, which ultimately saves money
* Be able to stand up to constant use without breaking or losing output resistance, as people often use the microwave at least once a day for food preparation or consumption needs
* Account for heat losses, the main source of power loss in the microwave
* Consume the least amount of power possible, as we had a limited amount of power available. We kept in mind that the NOMS system had additional power requirements above and beyond those normally expected of microwave operations

For the purposes of this project and based on the considerations listed above we chose the solid-state switch as the better fit for the NOMS system.

1.4.7 Regulator

In a microwave, the regulator takes the voltage output of the transformer and changes it into a voltage that the rest of the small voltage components can handle. Typically, the regulator will take anywhere from 17 V to 12 V and step the voltage down to the range of 5V to 3V. It was important to consider the output current while we were deciding on a regulator. The output current limits the amount of current available to provide all of the small voltage components.

The three regulators we considered for this project are provided in table 14. [32]

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial Number** | **Output Voltage** | **Output Current** | **Output Watts** |
| L7815 | 15 V | 1.5 Amps | 22.5 |
| L7805 | 5 V | 1.5 Amps | 7.5 W |
| L7812 | 12 V | 1.8 Amps | 21.6 W |

Table 14 - Regulator Power Data

For this project, we chose the L7812. This regulator allowed us to have 1.8 output amps, which was the largest amount of output current possible for a regulator. By selecting a regulator with 1.8 output amps, this allowed us to have some freedom in our low voltage circuit design. Voltage regulators run approximately $2 each. We made sure that all the low voltage devices pulled a total of less than 1.8 amps, in order to not overload the regulator. We achieved a current load of 1.6 amps, which allowed for a safety net in our power systems design.

1.4.8 Summary of Low Power Consumption Devices

See table 15 for all individual lower power load consumption.

|  |  |  |  |
| --- | --- | --- | --- |
| **Devices** | **Voltage** | **Current** | **Power Consumption** |
| Controller | 5 V | 0.255 Amps | 1.28 W |
| Camera | 0.7 | 0.0429 Amps | 0.03 W |
| Touch Screen | 5 V | 0.155 Amps | 0.76 W |
| **Total** |  | 0.453 Amps | 2.07 W |

Table 15 - Summary of Low Power Consumption

We did not need to regulate the voltage, as the utilities are obligated to provide the power within this range. We selected all DC components to be within the range of 9V to 12V.

This picture in figure 13 represents a hypothetical microwave circuit. The schematic displays the high voltage and low voltage components discussed above. On the low voltage side, we provided 5 V and 12 V circuits for various component operations. The high voltage side provides power to the fan, light, turntable, and the magnetron. We traced an existing microwave and determined the possible connectivity in order to create figure 13.

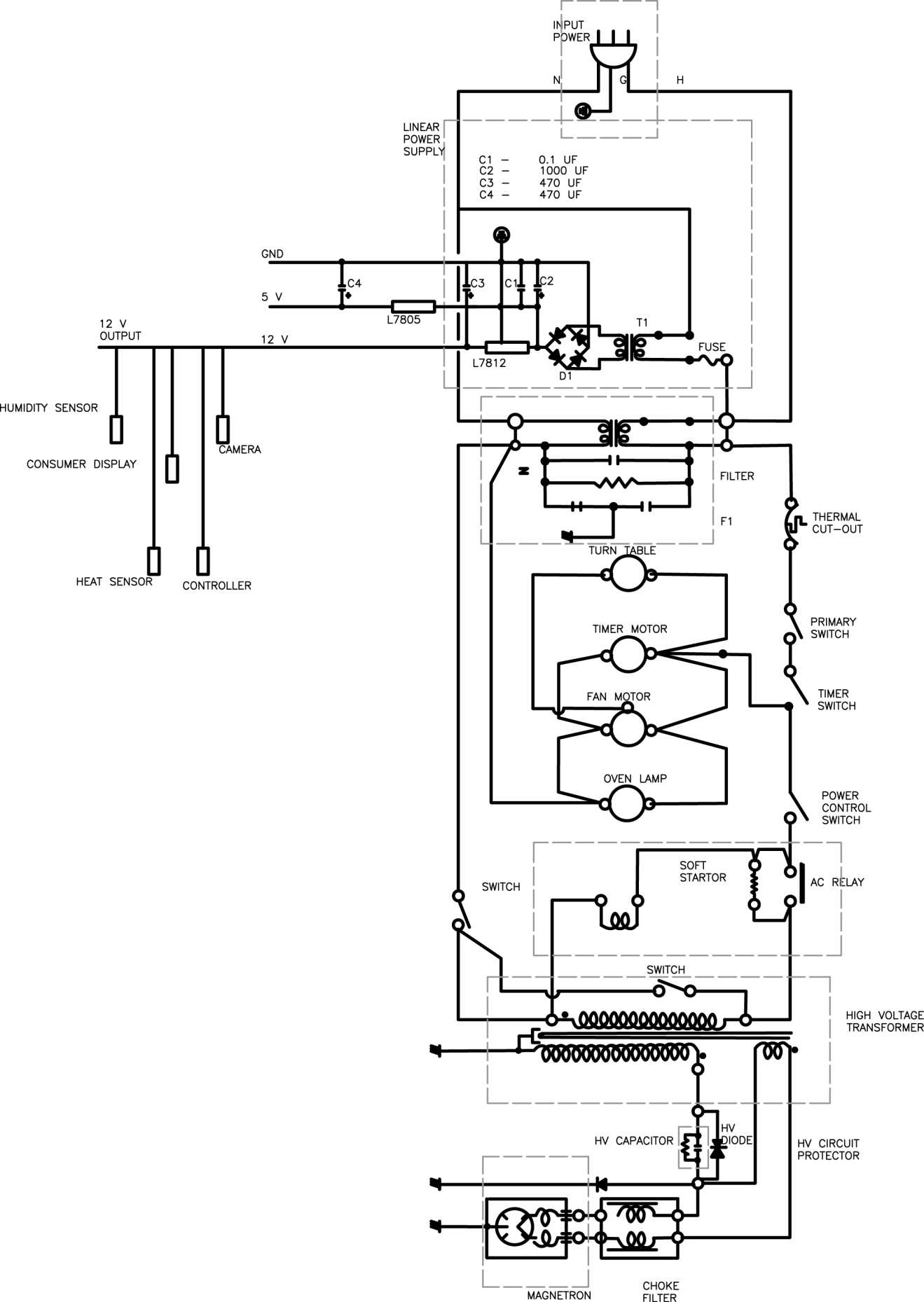


Figure 13 - Microwave Circuit Schematic

1.5 Power Systems Testing

Typically, the manufacture is required to run various tests to protect consumers against accidental damage or eliminate product liability. These tests include:

* Product Safety Test
  + Dielectric strength test
  + AC DC test
  + AC hipot
  + DC hipot
  + Arching
  + Line regulation
  + Load regulation
  + Min/max current detection
  + Grounding test
  + Polarity test
  + Ground bond test
  + Insulation resistance test
* Operator Safety Test including:
  + Tester environment
  + Operator training
  + Operator testing guideline and procedure.
* Application Test including:
  + Appliance testing
  + Motor testing
  + Transformer testing
  + Electrical component test
  + Functionally test

It was not feasible for us as students to conduct all of these tests; however, as a minimum we checked the following items during the construction of NOMS. During our assembly process we tested and inspected each device individually and once it was connected in a group, the sub system was individually ohm tested. Upon assembling our unit but prior to plugging in our unit, we inspected our device for possible damage during assembly. We inspected the plug to make sure it had a correct polarity and connection. We tested each lead for possible grounded connection. We conducted connectivity test for ground lead and run an ohm reading which determined the connectivity of power leads. We inspected and confirmed that all switches were properly positioned and we tested the on / off switch to make sure that it was functioning.

Once the appliance was plugged in we tested the voltage at input point of the transformer and open switches. The input voltage for AC/DC transformer was measured, along with DC output of the voltage regulator.

Prior to loading the magnetron, we tested the low voltage side by measuring the input amp to AC/DC transformer with all components connected. We completed this by reading the voltage and amps running through the low voltage side. Once we were satisfied with the low voltage results, we forced the magnetron to load and measured the amps and voltage for high voltage side. Lastly, we tested the microwave by cooking a meal and measured the overall amps and voltage.

1.5.1 High Voltage Power Protection and Feedback

The system implemented a high voltage power supply protection and feedback circuit allowing the system controlled to detect system faults and provide feedback to the system. The microwave is energized by the system by way of a low voltage relay. The system controller drives a pin high closing the contacts of the relay and allowing the high voltage current to flow through the system. This passes through a number of fuses and thermal protection sensors in the system. At the end of this loop is a high voltage relay. The high voltage line closes these contacts which drives a pin on the system controller high. Thus the logic of the protection circuit is that if the power pin is driven high then, after a small propagation delay, the status pin should also be driven high. If this is not the case then a general system fault has occurred and the system controller will stop operation and generate an error code which is fed to the UI. It is implemented in this method to avoid designing a circuit card and monitor explicitly for this subsystem. We do not need fine grained error codes related to the high voltage system as such errors are typically easily diagnosed by a servicer. However this rudimentary implementation is necessary so that the system controller does not continue to attempt to drive a faulty system and possibly damage itself or other components. A diagram of the circuit is depicted in figure 14. The fuse is nonspecific and in fact represents several components which are installed to protect the various high voltage (and low) components of the system.

Figure 14 - High Voltage Protection Circuit

1.5.2 Low Voltage Power Monitoring and Feedback

The system controller integrated a power monitor to ensure that all of the low voltage components were protected from damage and ensured that everything was operating within tolerances. This was implemented directly on the system controller using a Linear Technologies LTC2990 power monitor. The LTC2990 monitored voltage, current, and power draw by the low voltage subsystems. It was connected to the microcontroller via an onboard I2C bus. The unit was used to directly monitor levels as well as programmed to throw an alert when defined ranges have been exceeded. The package is shown in Figure 15.



Figure 15 - Low Voltage Monitor Package

1.5.3 Low Voltage Power Step-Down

The system required two different low voltage power levels. Many of the microwave components operate at 5V, and in fact this was the power that was provided to the system controller. However all of the integrated circuits use 3.3V signaling. It was therefore necessary to step down the voltage on the board to provide the necessary 3.3V power plane. This was done by way of an integrated Buck converter. Specifically the LTC3561 was integrated onto the system controller. It provided 3.3V at 1A to power both the system controller and the LCD unit. The converter was designed in LTSpice and is provided in Figure 16.

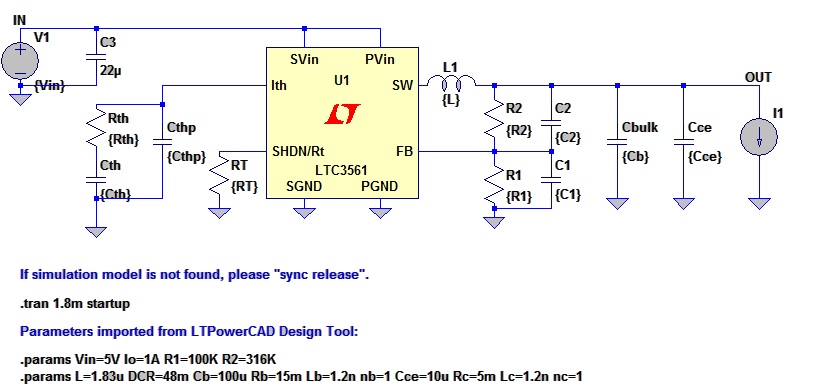


Figure 16 - Low Voltage Power Step-Down

System Controller

2.1 Requirements and Specifications

The system incorporated a centralized system controller. This handled several functions.

* Controlled the microwave magnetometer.
* Managed and controlled microwave haptics.
* Interfaced with a remote database.
* Communicated with the user interface module.
* Monitored system power

A number of systems were evaluated to fulfill this purpose. We wanted to achieve a balance of features that allowed rapid development without excess. Several cores were considered. Initially we considered using Arduino modules to implement the functionality required. These would have allowed for rapid development of the system. Prototyping would have been simple with very little design. This approach was rejected for the exact same reasons that it was considered. These modules are simply too powerful for our system and really would represent excess cost and processing overhead that was not needed for a system that can be implemented with a small embedded system. With that approach rejected we looked at components from Texas Instruments as they appeared to offer a selection of microprocessors which would meet our needs. Again a balance of features was necessary. We need several serial interfaces, and SPI, and an I2C bus. It was preferable for these components to be integrated into the microcontroller rather than having to use general purpose IO pins in conjunction with the appropriate transceivers to implement this functionality.

The two basic architectures under consideration were the MSP430 and ARM microprocessors. A quick summary of the important features are included in table 16.

|  |  |  |  |
| --- | --- | --- | --- |
|  | MSP430x5xx | TM4C123x | F28M3x |
| Operating Frequency | 25 MHz | 80 MHz | 100 MHz |
| SPI | 2 (shared) | 4 | 4 |
| I2C | 1 (shared) | 6 | 2 |
| Serial | 1 (shared) | 8 | 5 |
| GPIO | 29-87 | 12 | 88 |
| Flash | 512 KB | 256 KB | 256-512 KB(x2) |
| Ram | 66 KB | 32 KB | 16 KB(x2) |
| Floating Point | No | Yes | Yes |

Table 16 - Microcontroller Comparison

The MSP430 series of microcontrollers are low power, well suited to embedded applications. It includes several serial interface protocols but ultimately they are limited because they are shared between 2 ports (4 pins). This was not sufficient for our system. However, this limitation was offset by the generous number of general purpose IO pins. Some of these were used to implement additional serial interfaces but this increased the complexity of the circuit card. The Flash and Ram were sufficient for our purposes, or so we thought. The deciding factors were processor speed and the lack of a floating point unit. The most computationally expensive task that needed to be performed by the microcontroller was the task of deskewing of the images captured by the system cameras. This algorithm was computationally expensive and needed floating point functionality. The MSP430 series microcontrollers were not fast enough to adequately perform this task.

On the other end of the spectrum was the F28M3 series of microcontrollers. These utilize an ARM Cortex M3 core as well as a secondary unit responsible for communications. These units have a high operating frequency and a floating point unit which makes them better suited than the MSP430. The serial interfaces were sufficient for communicating with the rest of our systems. 88 general purpose IO pins was excessive for our application. Basically it was decided that this particular microprocessor really exceeded what was necessary for our application. The additional cost and the lack of a Launchpad reference board pushed us towards the TM4C123 series of microcontrollers.

The TM4C123 series of microcontrollers also utilize an ARM Cortex M3 core. It had the most integrated serial interfaces of any of the microcontrollers we considered. The number of GPIO pins was fairly limited but was decided to be sufficient for the number of functions that must be implemented on these pins. It was thought to be robust enough computationally to handle the image processing and decoding, but as we found out, this proved to be a false assumption. Since most of the data intensive storage will be moved off of the microcontroller to the LCD unit the limited onboard storage was not deemed to be an issue, as it only needed to hold the codebase of the system controller.

2.2 Circuit Card Assembly

Due to the many advanced integrated features of the selected microcontroller the circuit card assembly was fairly simple to design and integrate. The main components that need to be placed on the board were:

* The TM4C123 microcontroller
* The CC3000 SimpleLink Embedded Wi-Fi
* The LTC2990 Integrated Power Monitor
* The LTC3561 Buck Converter

In addition several pins needed to be mounted on the board to facilitate the connections between the cameras, LCD unit, and haptics of the microwave.

Embedded Wi-Fi

3.1 Functional Requirements/Design Considerations

Before we considered which embedded Wi-Fi chip to use we needed to understand the requirements of NOMS in order to make a proper selection. There were many options on the market for use and each with its own distinct features and flaws. Laying out our needs before shopping for our Wi-Fi chip helped to rule out certain options and lead us to what was a good fit for NOMS. The following were the considerations for selecting the Wi-Fi chip. They are listed in order from most important to least important to the team and were weighted accordingly in the selection process of our Wi-Fi chip to be placed on the System on Chip.

The first and most important consideration that the team looked at for NOMS’ Wi-Fi shield was its ease of operation. There were a number of horror stories from previous senior design groups that tried using an obscure part like a microcontroller or camera that ended up giving them all kinds of problems with operation. It was important to us that our design be functional, not that we blaze a new trail with materials unfamiliar to us and everyone around us. The idea for NOMS was itself original enough that part selection needed to be left to simplicity. The best way we could understand whether or not the Wi-Fi chip was easy to use outside of actual use was by its popularity among amateur users like ourselves and languages that it operated on.

Secondly, the Wi-Fi chip needed to be a low-cost solution that was affordable for the group. The design team for NOMS was self-financed meaning that we were not receiving outside funding for the design and production of the NOMS system. This was considered in light of all other existing products and their costs.

Thirdly, the reliability of the connection that we could attain through the Wi-Fi chip was of immense importance to NOMS. This provided the functionality needed to test and present the project at the committee review. Fortunately there weren’t many products that have known poor reliability. Again, the best source of judgment for this consideration aside from personal experience or experimentation was to with tried and proven materials that were popularly used on the market already.

One of the last considerations for the Wi-Fi chip was the amount of power it consumed. The NOMS system was built into a pre-existing microwave oven and used most of the parts already found in the original microwave. One of our main goals was to keep the system we created as energy efficient as possible. For this reason we needed to ensure that the materials that we used, including the Wi-Fi chip, were as efficient as possible. The Wi-Fi chip was surface-mounted onto the System on Chip. The System on Chip was programmed to make use of the Wi-Fi chip only when a user is interfacing with the LCD screen. There is a series of gates that allow for the Wi-Fi chip to be turned off while the system is in sleep-mode. This needed to be possible with the Wi-Fi chip that we selected in order to maintain the high efficiency that we were pursuing.

Finally, the Wi-Fi chip needed to be easily configured to the System on Chip. We prototyped the system initially with a pre-fabricated board bought straight from a manufacturer as-is. We plugged the Wi-Fi chip onto this prototype microcontroller in order to test each subsystem and ensure we designed the system correctly before we fabricated our own, original microcontroller. After fabrication and population, we mounted the Wi-Fi chip to the fabricated board. The Wi-Fi chip we selected was chosen to handle both our prototyping and our final installation onto the fabricated board.

3.2 Existing Systems Choices

3.2.1 XBee Wi-Fi Module

One of the earliest Wi-Fi chip options we looked at was the XBee Wi-Fi Module for OEMs. This is a product that a company called Digi produces and sells. Digi offers a wide variety of products in the wireless device market. They offer both personally tailored items and off-the-shelf devices as well. The XBee Wi-Fi Module is one such off-the-shelf device.

In considering the XBee Wi-Fi chip ease of operation was the most important thing we considered when weighing this option. This device appeared to be pretty simple to operate and program. Digi offers IP-to-device capability and device-to-cloud capability. This would be easy for us to operate from the cloud using Digi’s Device Cloud connectivity.

The second most important consideration for NOMS’ Wi-Fi chip was its cost. The XBee Wi-Fi chip is listed at $149.00 including the development board, all parts, and the chip itself. This price was a little higher than the group had originally planned to spend but we considered the cost in order to enable an effective solution.

In terms of reliability, we only had the reputation of Digi to go by as well as internet forums of people who had experience in using these devices. As discussed in the opening paragraph of this section, Digi is a relatively known company with success in the wireless device industry. Based on some of the forums that we looked at it seemed that XBee is a popular Wi-Fi chip with little issue in terms of reliability.

The power consumption of the XBee Wi-Fi chip kept with other chips that we explored, specifically TI’s cc3000, which will be discussed in the following section. Low power was therefore a non-issue between this option and the next.

Finally configuration of the XBee into the NOMS system and onto the System on Chip that we prototyped and eventually fabricated was a strong consideration and was the deciding factor for our group. The XBee supports both common through-hole and surface mount configuration for initial connection. As mentioned previously the XBee can be connected to a cloud that is accessed and used by the design team for NOMS.

The biggest downside to this product was that NOMS’ design team planned to use a Texas Instruments microcontroller for prototyping and the fabricated board that was built by the team would be loosely based on that prototype board. This being the case it was in our best interests and in the interest of synergy to keep with a Texas Instruments Wi-Fi chip. [45]

3.2.2 CC3000 SimpleLink Wi-Fi Module from Texas Instruments

Little to no introduction is needed for Texas Instruments and its products. Texas Instruments is one of the leading brands in the wireless device market currently. There are many valuable features that come along with buying a product from Texas Instruments. The following is a brief demonstration of those features and how they weighed into integration of the NOMS system.

The CC3000 SimpleLink Wi-Fi module was acquired in the form of a booster pack on the Texas Instruments website. This booster pack included a fairly detailed explanation of how to get started using the Wi-Fi chip and how to operate it once it was integrated into the system. In addition, Texas Instruments is so widely used that there were many forums available to access in order to learn use and troubleshooting of the device. Therefore, one of the best and most pertinent features that Texas Instruments offers was actually its popularity due to its simplicity of operation and wide use by other engineers.

Cost was the second most important consideration for the NOMS design team when considering a Wi-Fi chip. Texas Instruments’ website showed a price for the booster pack we selected between $35-41. This was already much more cost-effective than the XBee Wi-Fi module. In addition to this cost Texas Instruments offered free sampling of their products. This being the case the CC3000 made a strong case for selection based on pricing alone.

Reliability of Texas Instruments is a known feature when buying from their selection. In fact, this was one of the reasons we initially looked into Texas Instruments’ solutions for our Wi-Fi needs. The CC3000 was highly reliable in terms of functionality and connection, and if at any point a problem should come up in the use of Texas Instruments’ Wi-Fi chip there was a help line available as needed in their customer support centers and their vast library of previously answered questions. However, as we soon learned, this did not mean that we would be able to find answers to all of our questions.

The power consumption of the CC3000 was in the same realm of the XBee Wi-Fi module and was therefore not a determining factor between the two. The CC3000 used integrated crystal and power management.

Finally the configuration of the CC3000 was a very simple and easy to use process designed by Texas Instruments for its customers. The booster pack included software drivers, sample applications, an API guide, user documentation, and customer support available Monday-Friday 7AM to 7PM.

A final consideration that we gave to selecting the CC3000 was which microcontroller we decided to prototype with and what our fabricated board would be loosely based on. We decided before selecting which Wi-Fi chip to use that we would use another Texas Instruments device for the microcontroller in the prototyping phase and that it would therefore support the synergy and continuity of the NOMS system to aim for another Texas Instruments solution in terms of the Wi-Fi chip. [46]

After careful consideration of each chip and its various features and differences from the other we decided to select the CC3000 based on its broad use by the engineering community, its ease of use, its similarity to the other devices we intended to use, and its low cost for our design group. The common use that the CC3000 gets between many engineers proved to help us in our design process and integration into the NOMS system.

3.3 Subsystem Overview

The following block diagram in Figure 17 is a basic outline of the CC3000 Wi-Fi Module from Texas Instruments. On the right hand side the front end module connects with the 2.4 GHz band pass filter which connects to the antenna for input and output. There is a 26MHz Crystal and the 32.768 KHz Clock are both used for clocking, one for real-time and one for a system clock. The power enable comes into the FET which is DC/DC.

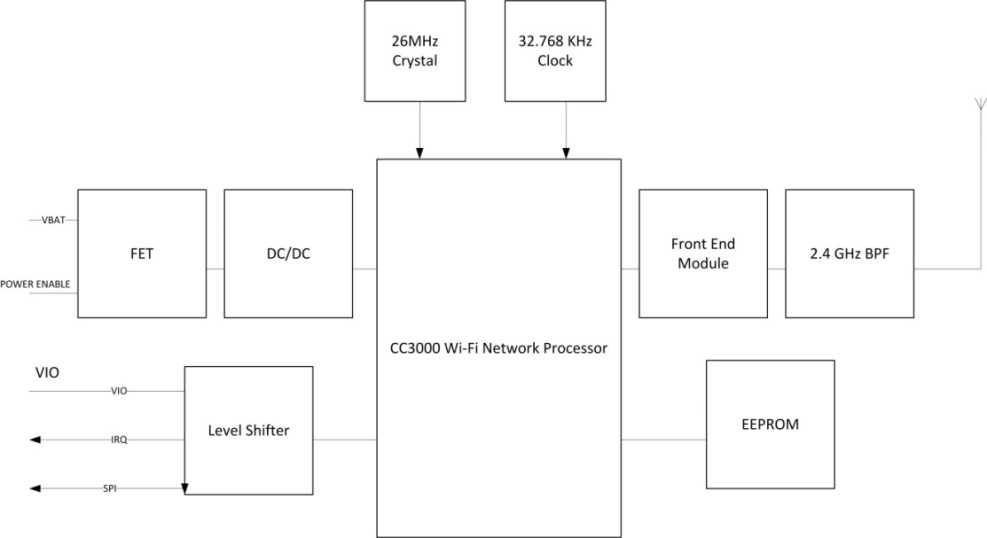


Figure 17 - Wi-Fi\_\_\_33 Block Diagram

3.4 Configuration of the CC3000

The CC3000 is connected to the microcontroller via serial peripheral interface (SPI). This is the communication standard used in the above diagram. Operation and configuration of the chip is done via a software API that is provided by Texas Instruments as discussed previously. The CC3000 can also be configured externally using direct Wi-Fi applications from a personal computer, android, or IOS device. This is known as SimpleLink Wi-Fi SmartConfig Technology and is included free with the booster pack we intend to purchase. This allowed for easy configuration for the design team and with any needed assistance from the Texas Instruments customer support center, forums, and documentation that came with the device. Configuration of the device with the prototype microcontroller was predicted to be straightforward. Upon completion of the prototype NOMS we reconfigured the CC3000 to the newly fabricated board that we based on the microcontroller prototype from Texas Instruments, which made reconfiguration to the new board nearly identical.

3.5 Parts Acquisition

In order to procure the CC3000 Wi-Fi Chip we ordered directly from the Texas Instruments website. On each product’s page there was an option to buy the product directly from the distributor or from a contract distributor. In our case there was also a third option to get a free sample from the distributor which was what we chose to do. In addition to this we purchased a second Wi-Fi chip in case of damage to the Wi-Fi chip in installation and configuration or in case we needed to test and verify a second system during completion of the project. In either case we acquired the chips directly from Texas Instruments and were financed by each member in the group independent of any outside financing. We applied for a Texas Instruments grant of $200 on the condition that we enter our final project into a design contest that Texas Instruments approves; however, we did not use this grant in the completion of our project.

Online Database

4.1 Functional Requirements

The online database used by NOMS would be accessed by means of the wireless internet connection on the SOC. It contained multiple fields including the EAN-13 code, the cook time, and the power level to cook the food item at. The EAN-13 is defined in the UPC Code sections. Databases usually make use of a particular field that is chosen as the “key” field. This means that as the database is searched it will specifically search for your item by searching through that particular field in each item until it finds the item in the database. This requires that the key in use is unique to one and only one item in the database. For NOMS this key field was the EAN-13 code. The database management system was queried with the EAN-13 code and outputs back to the SOC three fields: the cook time, item name, and the power level. For our project, the database management system was designed to be used and edited only by members of the team. We chose to design NOMS so users were not able to add to it or subtract from it. In addition the database was secured so that the user has access to it.

In addition, the database management system could keep track of EAN-13 codes searched for. There was a distinction made between items searched for and found in the database and items searched for and not found in the database. The first list could be used for marketing ideas and product research by food companies. The second list could be used by the team to see which items need to be added to the database as users scan their food items. At the time of presentation NOMS had fifteen different food items with EAN-13 codes in the database available for lookup.

Since we did not have additional time to design the system, we did not have the ability to add more features like we had initially planned, as this four person project transformed to a three person project. Initially, we thought we might have time to create new database fields for each item for nutritional information. Possible fields would be grams of fat, grams of protein, and vitamins and minerals. These fields would be useful when connected to a user’s personal profile. This would have allowed the user to track his or her nutritional intake based on microwave cooking. While not every meal is cooked in the microwave this would have been a good starting point on tracking all food cooked and eaten. In the future it would have been very easy to turn this type of food tracking for personal health profiles for use by the oven, stove, or just personal tracking in the kitchen apart from food items that require cooking such as fruit or vegetables.

3.2 Existing Database Choices

The tools used to access, change, and retrieve data in a database are called Database Management Systems (DBMS). These are software tools designed to make data analysis and acquisition simple so that greater tasks can be completed. For NOMS this meant that we needed to create a database and be able to interact with it through one of these database management systems. This allowed for simplicity in the area of acquiring cook times and power levels for use by the System on Chip (SOC). The question for NOMS was what type of database management system should we use? The following paragraphs contain brief descriptions of each option that was up for consideration.

Hierarchical databases are a popular choice for databases in today’s world. A hierarchical database uses a single root record and all other records fall beneath that one. The record that falls below another is called the child to that record. The record linked to a record below it is called the parent to that record. Similar to a binary search tree each record can have more than one child up to a predefined number of children. This concept is seen in figure 18. This can be very useful when searching for a specific record. The difficulty in using a hierarchical database management system is that editing such a tree of records requires an editing of the entire database since the rules of the hierarchical database require only one parent per record. Another consideration is that these types of database management systems are traditionally used for much larger groups of records than NOMS needed to make use of.

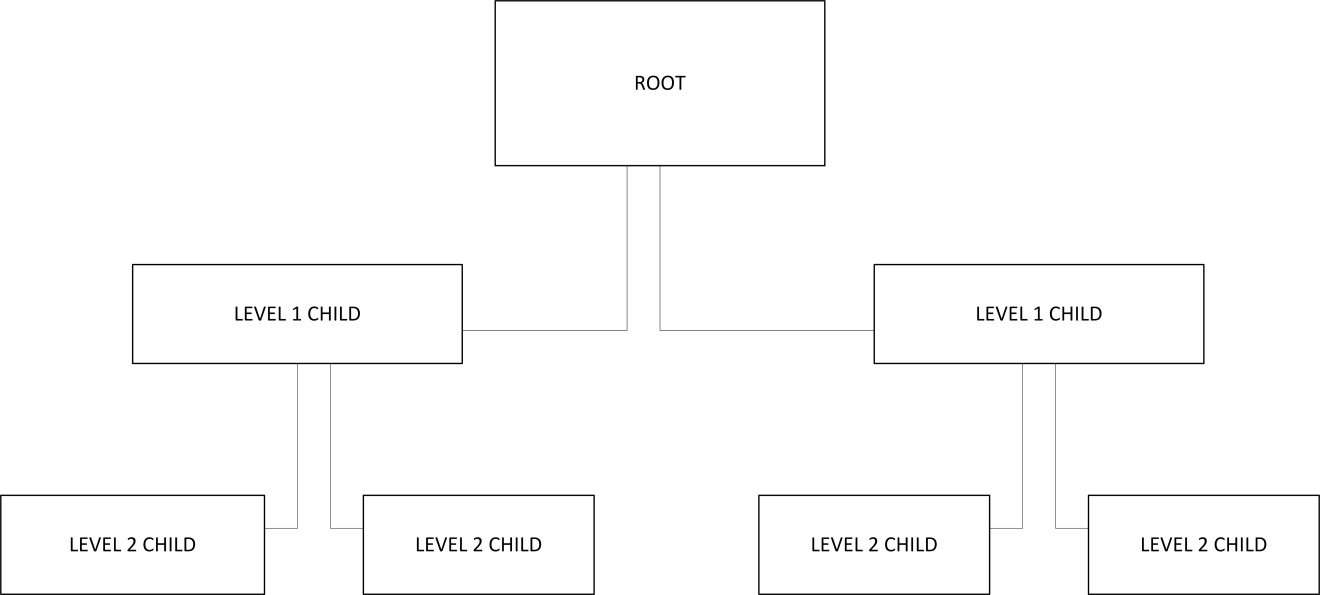


Figure 18 - Database Structure

Network databases are similar to hierarchical databases except that they allow multiple parents and relations between records. This concept is shown in the next figure (Figure 19). The difference between the two figures is small but in terms of functionality is very different. This does eliminate the difficulty offered in hierarchical database management systems, though not completely. There are still a predefined number of relationships that can take place between records. Another difficulty with these kinds of database management systems is that they must be defined before use. This would have caused complications later in the development of NOMS when a point is reached where flexibility was necessary in order to complete the project.

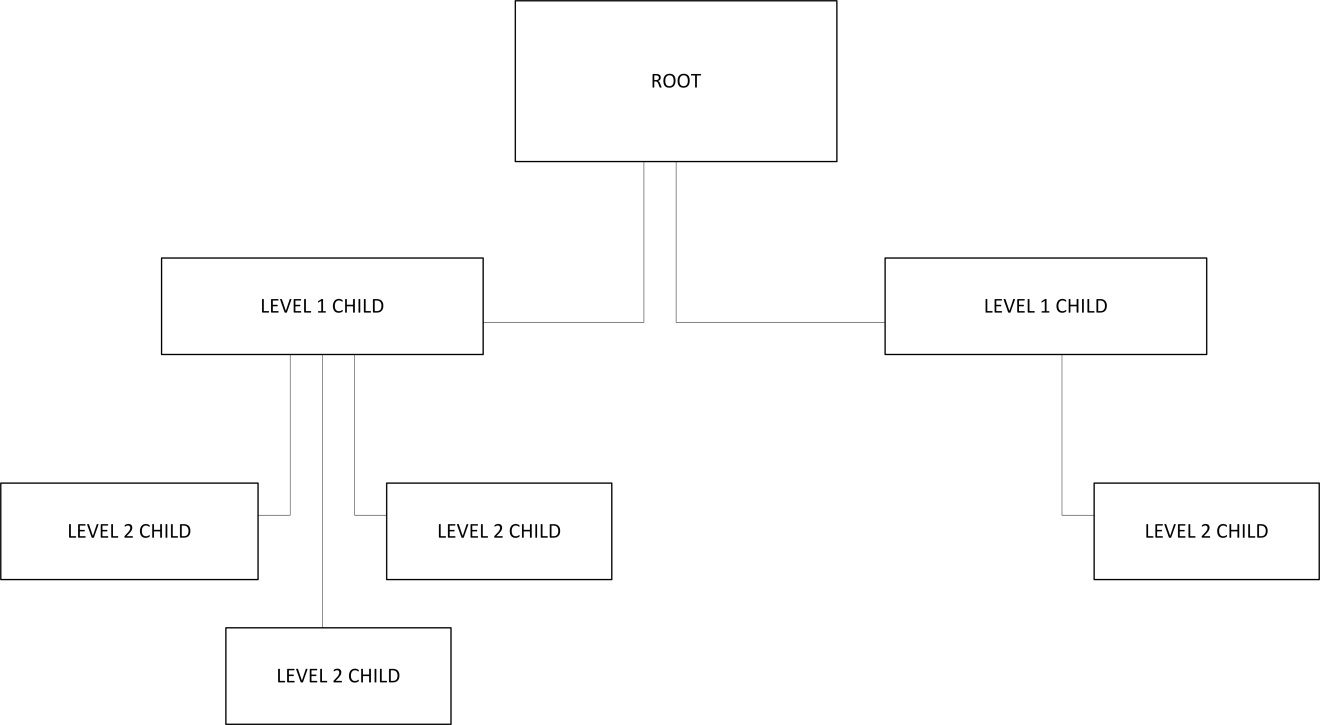


Figure 19 - Database Structure Flexibility

Relational databases are groups of records related to one another through relationship, not hierarchy. These databases also use the key concept briefly mentioned earlier. The idea of the key is that each item in the database has a unique piece of information in a field that no other item in the database will share. In our case this identifying piece of information was the UPC code itself. In this case we created two tables of data, one for cooking times and the other for power levels. Each table of data had exactly one item that corresponds to exactly one item in the other table. This was the relationship between the different records spoken of in the name “relational databases”.

Finally there was one other database management system researched called object-oriented databases. These database management systems are more advanced in what they are capable of storing in databases. These databases provide much more flexibility in the amount of data and what kind of data they can store. This is a very interesting idea but would not be necessary for what NOMS needs.

It appeared that after a brief overview of the different types of database management systems NOMS needed to make use of a relational database management system. The next step was to decide which relational DBMS to use. [38]

Microsoft Access was one possible solution for a relational database management system. It offered many features including tables, forms, reports, queries, macros, and modules. Access is a well-known commonly used database management system and would have been fairly easy to work with. It has many releases since 1992 with a 2013 version available. Access also supports C, C++, Java, and Visual Basic which was very useful in terms of development since these were the most well-known programming languages by the group members handling the database. Another nice feature of Access was that it uses GUIs for frontend data manipulation and acquisition. In comparison to other database management systems Access was a pretty popular choice and does most everything that was needed. The downsides to this software was that Access was mainly commercial, was made for individual use normally, and more importantly for our project it was not free. Prices start in the $100’s. [39]

SQLite was another popular database management system that is available for use. It was released originally in 2000 and is currently at version 3.8.1. SQLite’s source code was in the public domain, was pretty self-sustaining with minimal assistance from outside sources online, didn’t need setup or installation for use, and was very widely used for relational database management systems. The biggest downside was that it didn’t use a server but was more of an embedded system type of database management system. This didn’t really fit NOMS since we wanted the database to be online rather than within the system itself. [40]

MySQL is an Oracle product created in 1995. It has open source licensing and ranks 2nd in database engines by db-engines.com. MySQL’s website boasts that successful companies such as Facebook, Google, and Adobe all use MySQL to power their high volume websites. Something else that made this product stand out above the others is that it could be used on several different operating systems including Windows, Linux, and OS X. The original plan for NOMS was to use Windows but this versatility was helpful in the development phase. In addition MySQL supports many of the same programming languages as Microsoft Access and supports several others as well. This added to the value of such a database management system. The major problem with MySQL was the cost of buying and using such a database management system was not only not free but much more costly than Microsoft Access. This was a major sticking point for the group as NOMS was funded almost entirely by the group members themselves. [41]

Microsoft SQL Server was the last database management system that will be considered for use in NOMS. This product, like others already discussed, was built more for large companies doing a lot of data analysis and acquisition. Microsoft SQL Server is designed to handle large amounts of traffic boasting 1 million commands per second possible. It is a highly secure tool, widely used, and recognized throughout the business world as a top competitor for database management systems. The server operating system is Windows only which did not affect NOMS either way. One major downside was that it does not support any form of C. Being such a successful competitor in the database management system market we weren’t sure we’d be able to afford this solution but Microsoft offered a free version for developers like us called Microsoft SQL Server Express Edition. This ended up being the solution we chose. [42]

4.3 Creating a Server

In order to connect the embedded Wi-Fi with the online database that contains all of the food items and their corresponding fields we needed to have access to a server. A server is a computer that holds the database and allows access by clients such as our NOMS system. The database was created and maintained on one of the group member’s personal laptops. A port is selected on the computer that is used for communication for a particular database management system. The concept of using a laptop as a server is shown in the following figure (Figure 20).

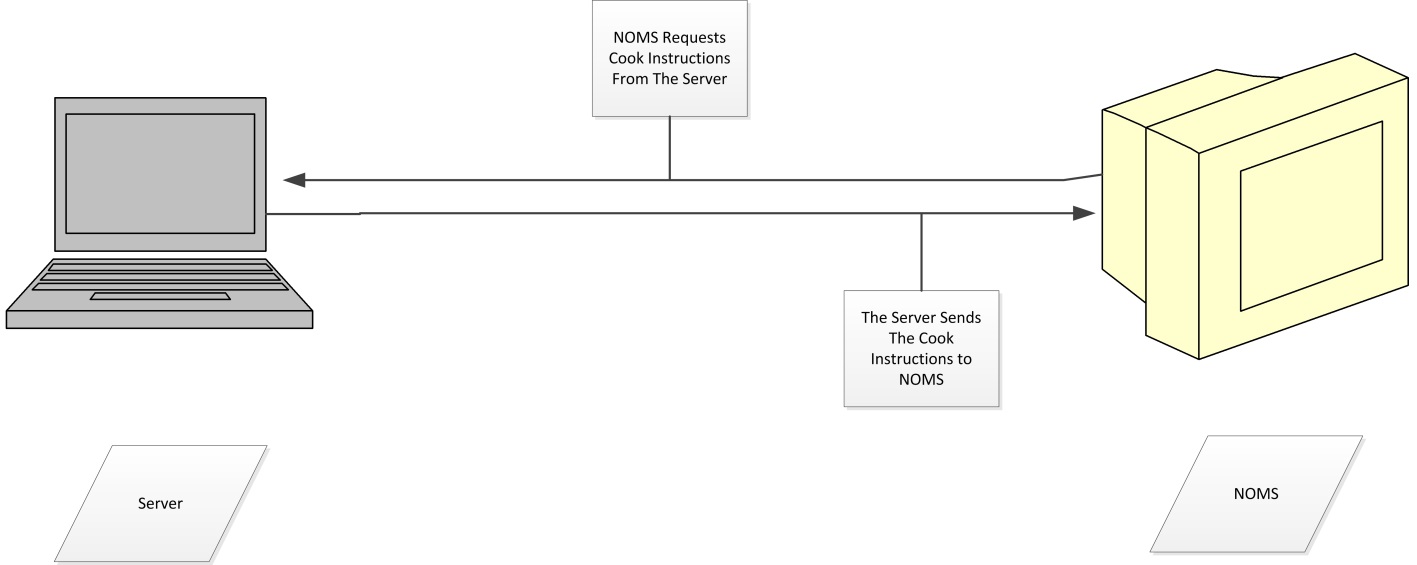


Figure 20 - Database Communications

4.4 Integration of the Database with the Wi-Fi SOC

Once a connection was made the Wi-Fi chip needed to follow programmed instructions in order to query the database and search for food items and returned values for cooking times and power levels.

It was discovered in implementing the Wi-Fi that the server computer containing the database and the microcontroller would need to be on the same network in order to have access to the server and send and receive data to the appropriate ports. The following figure is an example of a server-client relationship.

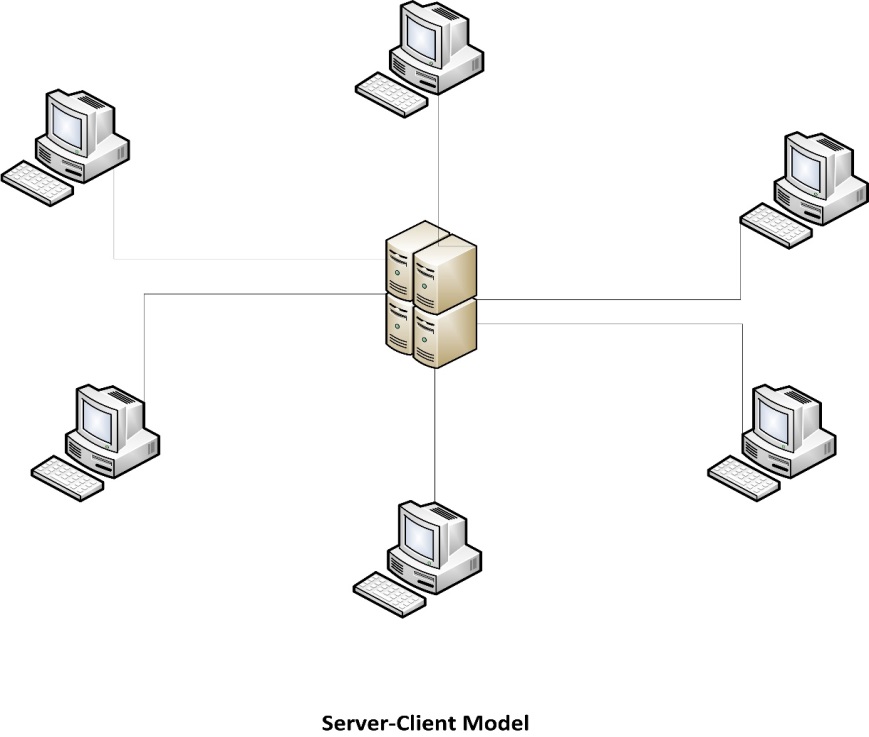


Figure 21 - Network Topology

The next figure shows the Peer to Peer form of networking that we used if we did not have each device across the same network. This was allowable due to the wireless capabilities of the CC3000.

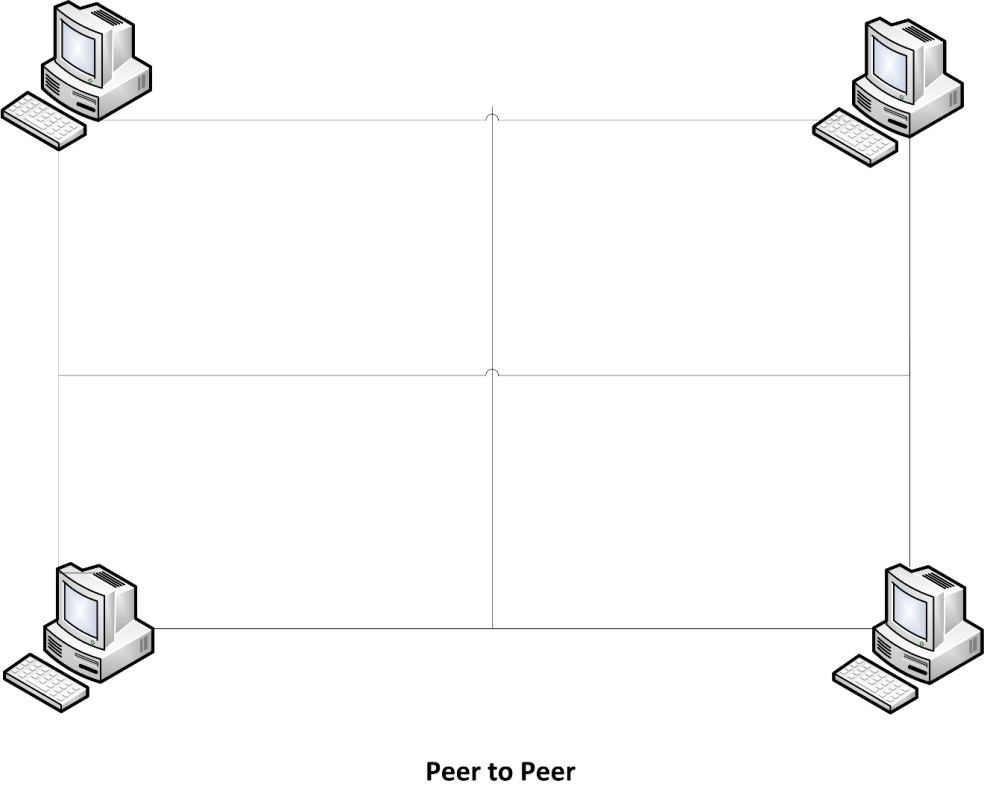


Figure 22 - Peer to Peer

In addition we discussed the database as an online database but are now showing an implementation using a local server such as a group member’s laptop. If in future development we needed to implement the database online we would simply need to lease a static IP address from an internet service provider or lease a domain name with a dynamic IP address. This would never become a problem as long as the Wi-Fi chip code always pointed to the correct name. [43]

4.5 Acquiring an IP Address

Leasing an IP address is relatively simple and straightforward as well, costing at most up to $5 per month. Internet service providers often will lease IP addresses for hosting but there are several other sites that will lease IP addresses. The main difficulty in getting an IP address is that they usually are for sale in large bundles which is not only unnecessary for NOMS but also expensive. [44]

4.6 Querying the database

We created a database and connected it to the Wi-Fi chip to the server that maintains the database, so when we need to send queries from the System on Chip to the database we were able to receive the fields requested (cooking time, power level, turntable functionality, etc.). To query the database we programmed the commands to be sent serially from the SOC where the Wi-Fi chip handles packeting and routing and returns the information requested from the server to the SOC. We did however decide to implement an onboard database so that querying across the network was no longer necessary.

UPC Codes

6.1 Motivation for Use

Established in 1973, Universal Product Codes, hereafter referred to as UPCs, have become a near universal for identifying trade items. They provide a simple system for retailers to inventory and sell. While not ideal for our purposes they provide a usable point of data that allows us to achieve our goal of automatically cooking. They are fairly limited in the amount of data that can be encoded which means that their use requires additional support systems to use them in the microwave. Despite this limitation we have decided to implement their use because the system is already in place.

6.2 UPC Standards

The governing body which established and maintains the standards regulating the use and implementation of UPCs is GS1. The relevant specifications are detailed in the GS1 general Specifications document. [33]

The most common standard in use is UPC-A which is 12 digit system of trade identification. It has been superseded by the European Article Number; hereafter referred to as EAN, and is a subset of this standard. The EAN13 specification adds a thirteenth digit to the code. UPC-A can be converted to EAN13 by simply appending a 0 to the front of the code. For this reason we restricted our discussion to the EAN13 specification.

6.3 EAN-13 Specification and Composition

A fairly detailed discussion of how the EAN-13 barcode is implemented will be necessary as we implemented systems capable of capturing and decoding these codes in order to identify what is to be cooked.



Figure 23 - EAN-13 Symbology

The EAN-13 specification encodes a Global Trade Item Number or GTIN which uniquely identifies the item being scanned. The GTIN-13 is composed of three parts; the GS1 company prefix, item reference, and check digit. The company prefix is issued by the GS1 Member Organization of which the company is a part. This portion was important for our purposes, aside from product identification, in that it allowed us to identify the manufacturer for advertising and product tracking. The item reference identifies the individual product. Finally the check digit is used for error correction. This code was relayed to the database subsystem by the SoC.

The physical composition of the EAN-13 barcode was our primary concern. It consists of:

* A left Quiet Zone
* A normal Guard Bar Pattern
* Six symbol characters from number sets A and B
* A center Guard Bar pattern
* Six symbol characters from number set C
* A normal Guard Bar Pattern
* A right Quiet Zone



Figure 24 - EAN-13 Composition

Symbol characters are encoded in three different sets known as A, B, and C as given in Table 17

.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Digit  Value | Set A element Widths | | | | Set B Element Widths | | | | Set C Element Widths | | | |
|  | S | B | S | B | S | B | S | B | S | B | S | B |
| 0 | 3 | 2 | 1 | 1 | 1 | 1 | 2 | 3 | 3 | 2 | 1 | 1 |
| 1 | 2 | 2 | 2 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 1 |
| 2 | 2 | 1 | 2 | 2 | 2 | 2 | 1 | 2 | 2 | 1 | 2 | 2 |
| 3 | 1 | 4 | 1 | 1 | 1 | 1 | 4 | 1 | 1 | 4 | 1 | 1 |
| 4 | 1 | 1 | 3 | 2 | 2 | 3 | 1 | 1 | 1 | 1 | 3 | 2 |
| 5 | 1 | 2 | 3 | 1 | 1 | 3 | 2 | 1 | 1 | 2 | 3 | 1 |
| 6 | 1 | 1 | 1 | 4 | 4 | 1 | 1 | 1 | 1 | 1 | 1 | 4 |
| 7 | 1 | 3 | 1 | 2 | 2 | 1 | 3 | 1 | 1 | 3 | 1 | 2 |
| 8 | 1 | 2 | 1 | 3 | 3 | 1 | 2 | 1 | 1 | 2 | 1 | 3 |
| 9 | 3 | 1 | 1 | 2 | 2 | 1 | 1 | 3 | 3 | 1 | 1 | 2 |

Table 17 - Number Sets A, B, and C

Since the EAN-13 Bar code consists of only 12 symbols the thirteenth digit (leftmost) is encoded by the variable parity mix of the first six symbols as given in Table 18.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Encoded  Digit | Number Sets Used of Numbering Left half of an EAN-13 Bar Code | | | | | |
|  | 1 | 2 | 3 | 4 | 5 | 6 |
| 0 | A | A | A | A | A | A |
| 1 | A | A | B | A | B | B |
| 2 | A | A | B | B | A | B |
| 3 | A | A | B | B | B | A |
| 4 | A | B | A | A | B | B |
| 5 | A | B | B | A | A | B |
| 6 | A | B | B | B | A | A |
| 7 | A | B | A | B | A | B |
| 8 | A | B | A | B | B | A |
| 9 | A | B | B | A | B | A |

Table 18 – Implicit Digit Encoding

UPC Capture

Two basic methods exist for the physical capture of the barcode. The simplest and most common method is laser scanning. They are very well suited to scanning EAN-13 barcodes but lack the flexibility to read 2D codes (such as QR). If we had chosen to use this method there were several off-the-shelf solutions suitable for our purposes however these were cost prohibitive. The second method was digital imaging. All that was ultimately needed was a simple, inexpensive, CMOS camera. This method offered the flexibility to capture 2D codes and was the primary method which we explored for capturing our barcodes. The significant disadvantage associated with this method was the necessity of processing the image so that it could be decoded.

6.4.1 Grayscale Conversion

The image captured by the camera was relayed back to the SoC. At this point some post-processing was necessary to make it possible to decode the UPC. The first step we used was Grayscale conversion of the image. Since the barcode is, by definition, black and white we removed a lot of “noise” from the image simply by performing this conversion. The format of the data packet depended on the camera that we ended up using, but regardless of the format it was converted into a simple 2d array where each pixel was represented by three 32-bit values for red, green, and blue intensity. Each of these was converted to a single 32-bit grayscale value. Some experimentation was needed to determine the best method for our application but three basic algorithms existed for performing this calculation.

Lightness method:

Average method:

Lumosity method:

It was likely that the individual algorithms above produced similarly useful results for our application and so the simplest average method was applied.

6.4.2 Filter

With the image converted to grayscale we further refined our image by filtering pixels which were not black or white. The specific thresholds we used were subject to experimentation but the essential idea was that pixels which did not lie at the extreme ends, black or white, were filtered from the image leaving primarily the barcode.

6.4.3 Black and White Conversion

There was no longer any reason to maintain any grayscale information and all pixels were converted to a simple black and white representation. This was done primarily to simplify the final 2 steps.

6.4.4 Deskew

It was advantageous to correct any image skew that was present to make it simple to decode the UPC. This was done by detecting the lines in the image using a Hough transform, calculating the slope of those lines, calculating the average skew, and rotating the image. This was the most processor intensive part of the process requiring a great deal of floating point math and required a critical algorithm which determined the necessary computational power of our SoC. However, today’s processors are generally much more powerful than is strictly necessary for performing their task and the SoCs we chose was well suited to the task.

6.4.5 Crop

Blank, or nearly blank, lines were eliminated from the image entirely leaving only the UPC for processing.

6.4.6 UPC Decoding

With the image deskewed and cropped it was fairly straightforward to decode the image. Basic edge detection techniques allowed us to detect the guard bar patterns and determine the thickness of the various bars and spaces. The GS1 specification provided a basic algorithm for decoding the UPC and was used as a starting point for our system. It is presented in section 5.2.2 of the specification document.

6.4.7 Check Digit

One final consideration that was not addressed up until this point was data validation. The final symbol in the barcode is the check digit. It is used to validate the decoded GTIN-13. It is given by the following formula:

6.5 Programming Considerations

While a good deal of code needed to be developed, the algorithms for performing many of these tasks were well established. There were a couple of C libraries specifically created recognizing and decoding UPC codes. Specifically the open source Zbar (<http://zbar.sourceforge.net/>) libraries were evaluated for use in our project. However, as the project progressed we determined we were not able to use Zbar as initially expected and had to generate original libraries in order for this project to be functional.

QR Codes

7.1 Motivation for Use

QR Codes are a 2D standard which is able to convey much more information than a simple UPC. We intended to use these codes to deliver cooking instructions directly to the microwave system without the necessity of accessing a database. The advantages were offset partially however by the relative complexity of capturing and decoding them. A laser scanning system was not sufficient and a digitizer was required. Their inclusion in the system was not finalized and further testing and evaluation would be necessary to determine both feasibility and efficacy. As this project became a three person project, we eliminated this aspect from our design.

7.2 QR Standard

The QR code system was originally invented by Toyota for vehicle tracking during manufacturing but its use has expanded well beyond that limited function. Today QR codes can be found nearly everywhere, from POS systems, to advertising, and even as login credentials. While GS1 manages its own QR based standard, it was not particularly useful for our purposes. The specific regarding QR and its use in retail ultimately does not pertain to the specific of encoding and decoding data in this format. Instead we will draw our information from the ISO/IEC 18004:2006 specification. [34]

7.3 QR Specification

While all QR codes share several features, their specific implementation can vary significantly depending on the specific type of code. There are forty different QR code types mostly differing in the amount of data which can be stored. The most common format is version 2 which can encode, at a medium level of error correction 224 bits of data. This will allow for 14, 16-bit fields, which is sufficient for our needs and will be the QR code implemented.

7.4 QR Composition

QR codes are complex and integrate several advanced features over UPC. The QR incorporates various features which allowed the code to be more easily captured and be more tolerant of damage and image distortion. These included multiple levels of error correction as well as data masking to distribute the data across the code area. Some important features of the QR structure and layout are shown in Figures 25 and 26.

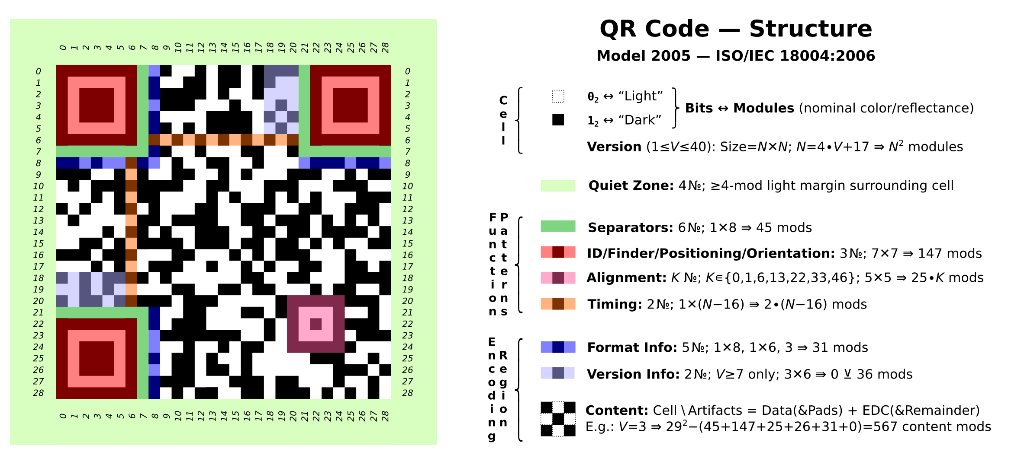


Figure 25 - QR Code Structure

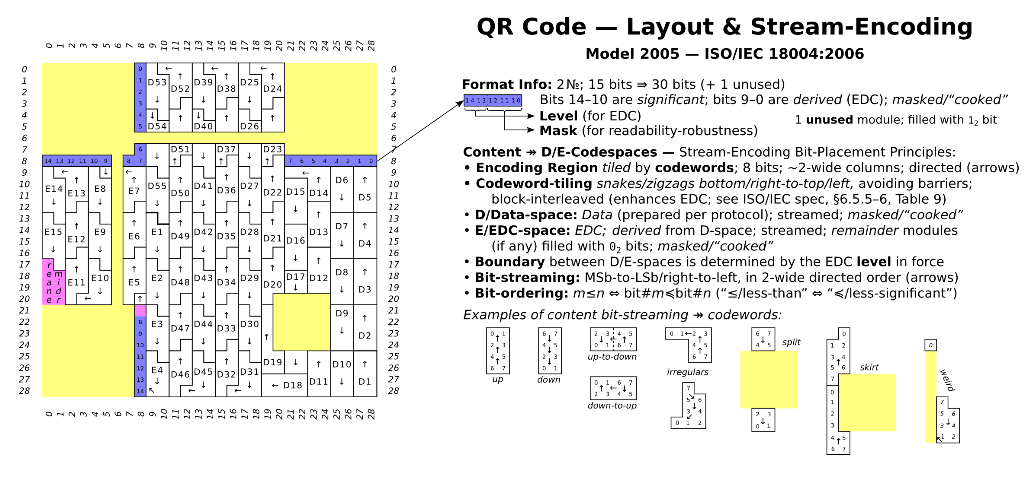


Figure 26 - QR Code Layout

7.5 QR Generation

The specifics of generating a QR code are not important for our application. There are several, freely available tools for creating QR codes encoded with whatever data you wish. We restricted our discussion instead to the specific data fields and formatting that were considered. The first two fields will be identical to those found on the UPC. The first will be a 16-bit Integer which will contain the company prefix followed by another 16-bit Integer containing the item reference. The remaining fields will encode the various cooking times and temperatures with a flag for the turntable. The first 4 bits will encode the power level from 0-10. The next 11 bits will contain the number of seconds to operate at that power. This allows for approximately half an hour of operation at a given power. Should the unlikely situation arise that a longer cooking time is needed, a second field at the same power will extend that time. The final bit will flag whether or not the turntable is to operate. The chosen QR specification can encode 14 of these 16 bit fields. This allows us to have instruction sets for multiple wattages, 4 instructions for 3 wattages.

0 16

|  |  |  |
| --- | --- | --- |
| Company Prefix | | |
| Item Reference | | |
| Power | Time | Turntable Flag |
| … | | |

Table 19 - Cook Data Format

7.6 QR Code Capture

The 2D nature of QR symbology necessitates the use of digital imaging for their capture. The methods discussed previously in the capture of UPCs apply equally well to QR code capture. However additional care must be taken when choosing the camera, its position, and focus. The data is more compact in the QR code than the UPC and a higher fidelity source image must be obtained for reliable decoding of the information. This is somewhat mitigated by the various levels of error correction inherent in the QR standard. We will attempt to balance camera performance at a medium level of error correction.

7.7 QR Decoding

Code level implementation of QR decoding algorithms is a complex task that would, in and of itself, represent a significant software design project. It is outside of the scope of this project and the use of QR codes in our application will rely heavily on the feasibility of using an existing open source library, namely the aforementioned Zbar C Library.

User Interface

When considering the design of the user interface it became immediately apparent that a simple push-button interface would not be sufficient for our needs. Migrating to a touchscreen LCD seemed to be the most efficient solution providing the necessary interactivity and feedback to meet our design goals. The user interface provided the following functions:

* Display the time while in standby
* Provided a method whereby either a UPC or QR code could be scanned
* Provided a method for manual cooking
* Provided a method for the system to be configured
* Provided a method to enter and maintain use profiles
* Provided a method for entering and maintaining custom cooking instructions
* Displayed product and status information while cooking

These features were implemented in an intuitive and dynamic way. The purpose of this system was to make the microwave experience simpler and more feature-rich. The home screen was simple and clean presenting very few options. The primary function of the microwave was to scan a product. This option was presented prominently on the screen to encourage the user to automatically select it. The home screen allowed access to manual cooking and settings. The design is shown in figure 27.



Figure 27 - Home Screen

While food scanning was the primary means of instruction entry, the option to manually enter cooking instructions was provided. It provides the traditional number pad as well as buttons to specify both the time and cooking power. We considered allowing macro instruction entry from this screen. Implementation of this feature was subject to prototype testing. Additionally, all screens with the exception of the home screen were implement two buttons at the top. These were the home button which enabled the user to immediately jump back to the home screen, as well as a simple back button for navigating back to the previous screen. With users nearly universally familiar with these functions and icons from internet navigation it seemed natural to implement them in this fashion.

The manual entry screen is shown in figure 28.



Figure 28 - Manual Entry

Upon pressing the scan button users were presented with a feed from the front-facing camera. This was designed to enable the user to position the product so that the microwave could scan the barcode. Previous products similar to this project implemented this step by way of a volumetric laser scanner. These were not rasterized and required the user to physically remove a scanning wand from the microwave and pass it over the bar code of the item to be cooked. This method was slow and awkward and delivered inconsistent results. Again the microwave we designed was intended to make things simpler, faster, and more effective rather than complicating the process. A digitizer, as implemented in our system, yielded better results. After the product was recognized the display outputs the found product as well as the total cooking time. The user is then presented with three options. They can begin cooking by pressing start and this option is presented prominently by placing it in the center. Alternatively they can override the default cooking instructions and enter custom instructions at this time. Lastly the option to rescan is presented. Additionally this was solved by having the microwave continue to scan until a selection was determined. This allowed for correction of any incorrect scans as well as allowed the user to scan a different product rather than the one the initially scanned.

The display is shown in Figure 29.



Figure 29 - Scanning

Choosing to override the default cooking instructions presents the screen in Figure 30. This screen provides a blank slate for the user to enter and store custom cooking instructions. This was important for user friendliness as well as the possibility that a product was not recognized or indexed in the database. This entry was a component of this part of the user interface so that it can automatically be associated with the scanned product. The screen presented was designed to be similar to that of the manual entry with the exception that macro instructions will be presented by default. Each set of instructions has a time and power associated with it. After an instruction line is entered the user can enter an additional line pressing next or complete the entry by pressing save.

The override is shown in Figure 30.



Figure 30 - Override

The save option will bring the user to the profile select screen shown in Figure 31. The idea was to allow fine-grained customization per user. Some may prefer their food to be hotter or colder than others in the household or may be willing to wait longer to obtain a more even cook. Thus custom instructions were designed to be saved to the individual user as well as globally for the microwave. Setting up these profiles will be discussed in more detail later.

The save custom is shown in Figure 31. We were not able to implement this in our design, but included below is our considerations for this feature.



Figure 31 - Save Custom

The actual cooking screen presented a large amount of information and figure 32 is simplified. The cooking screen showed the total cook time remaining, information regarding the current phase of cooking including the power, and time remaining of the current phase. We decided to discretely show the current time and power while the total time remaining is implemented as a countdown clock. One of the most unique (optional) features of the microwave is the ability to push information to the user from the online database. This will most likely take the form of advertising from the manufacturer of the item currently being scanned. There may also be general advertising or even advertising from competitors! This feature should be opt-out by the user but there will also be advantages to leaving it on. The advertisements should also include digital coupons for the items. These can be sent directly to the email associated with the profile of the current user (or default). They are likely to be static in nature. The resolution of the display is not sufficient for high quality video. While the system does have a speaker and would be capable of audio, it is unlikely this will be allowed or may by explicitly opt-in. Further testing will be required. Options from this screen then include the ability send the coupon, which will not navigate away from the page, as well as the ability to log the food for diet tracking.



Figure 32 - Cook Screen

Selecting to log the food gives the user the ability to select which profile to log it to. This profile needs to be associated with a diet tracking application. Currently it was associated with MyFitnessPal which is the largest diet tracking service currently in operation.

The profile selection screen is shown in Figure 33. We were not able to implement this in our design, but included below is our considerations for this feature.



Figure 33 - Log Screen

After selecting the profile to log the food to they will need to select how many servings are to be logged. This is necessary because it is common practice for products to contain multiple servings. The various ways this is done may complicate this page as it is not uncommon for a package to even contain partial servings. It’s an odd situation created by manufacturers trying to claim certain things about the nutritional contents of their food and disguising the actual values by listing it as multiple servings. Serving selection is in Figure 34.



Figure 34 – Servings

Selecting settings from the home page presented a couple of options. System settings will allow the user to adjust the colors both the background and text. The date and time should be pulled down from the servers and it will not be necessary to set these values manually but the user’s time zone and DST selection will be necessary. If someone has not connected their microwave the option is available to enter the time and date manually. The system reset will revert to factory settings erasing all profiles, caches, etc. This dialog will confirm with the user that they want to perform that function with an explicit warning that it erases all data. This screen is not depicted.

The System screen is described in figure 35. We were not able to implement this in our design, but included below is our considerations for this feature.



Figure 35 - System Settings

The appearance screen will allow the user to customize colors via a gradient bar and should be simple and user friendly. The date and time screen is simple as well only presenting options to adjust the time zone or enter the time and date manually. The time zone screen is not depicted but will likely only present options for the U.S. time zones.

The appearance and date/time screens are shown in figures 36 and 37. We were not able to implement this in our design, but included below is our considerations for these features. We were able to implement a running clock, and the user is able to set the time for this clock from the home screen.



Figure 36 - Appearance



Figure 37 - Date and Time

Manually entering the time and date takes several screens. It’s presented in a natural format with individual screens for time, day, month, and year.

Figure 38 - Time Entry Figure 39 - Month Entry

Figure 40 - Day Entry Figure 41 - Year Entry

User profiles are another unique feature of the system. These allow multiple users to customize their experience, their cooking preferences, and individual diet tracking by way of MyFitnessPal integration. This should be easy to set up and the only information that is necessary to do this is a name for the profile and an email address. The Email address enables the MFP integration as well as a place to send the coupons generated by the push advertising. Obviously the MyFitnessPal should be an optional component and so that part of the profile setup can be skipped though it does offer great additional benefits.

The screens associated with profile management are shown in Figures 42 through 45. We were not able to implement this in our design, but included below is our considerations for this feature.

Figure 42 - Profiles Figure 43 - Name Entry

Figure 44 - Email Entry Figure 45 - MyFitnessPal

These systems combine to provide a simple an intuitive and simple to use user interface. The user can navigate in a natural way to accomplish whatever they would like on the system. A basic overview of the user interface design flow is shown in Figure 55.

Figure 46 - UI Flow

System Display and Touch Screen

9.1 Requirements

The System Display was an essential component of this system, as such care needed to be exercised in choosing a component which would meet our needs. The display needed to be low cost and yet offer sufficient resolution to display the varying content. The main constraints were that it needed to be able to adequately display the camera feed as well as the push advertising. Additionally we wanted a simple solution for allowing the user to interact with the microwave. This implementation was considered in a couple of ways. A small screen could be integrated with a traditional keypad. This approach was simple. It would not be necessary to have a touch screen. The keypad would be easily integrated into the system controller. The cost of these simple components was low. The resolution of these small LCD units was very low and they are almost universally monochrome. Ultimately this approach was rejected in favor of a larger touch screen interface. The small screen and keypad option just did not offer the flexibility that we wanted in this design. The larger screen offered better resolution and user interface flexibility. The larger color screen offered a much more pleasing aesthetic. The advertisements were in color and much more engaging. It’s possible that video could even be played, although we did not explore this venue. The disadvantages of this option included the greater power draw as well as cost associated with the unit.

9.2 LCD Selection

There were a couple of options when considering which LCD to utilize. The least expensive option was to get a simple LCD panel with an integrated touch interface. There were endless options for a simple dumb LCD panel but using one of these units introduces additional complexity to the system controller. It would be necessary to handle all user interface at the system controller level. The controller would need to integrate and LCD driver. This made the selection of microprocessor much more limited and expensive. Additionally the complexity of coding all necessary functions onto a system controller which was already handling database management, system control, and wireless communication was simply not ideal.

The second option was to choose an LCD with an onboard controller. These units integrate a microcontroller which was used to drive LCD as well as handle some rudimentary logic functionality. Depending on how robust this system was it could be used to drive the user interface completely separately from the system controller. The obvious advantage being that database management and system control could happen, not only separately, but in parallel which resulted in a much better experience for the user. Another benefit was in power. The system controller can go, not just into a low power state, but be completely powered down and then boot by the LCD controller whenever a user event is triggered.

It was ultimately decided that the system would utilize the 4D Systems micro LCD-32PTU. This LCD has onboard logic and storage which would allow us to implement all user interface functions directly on the LCD system. It is a 3.2”, 240x320 color LCD screen. It has an integrated PICASO processor which is proprietary to 4D Systems units, a micro-SD card connector, and even an audio amplifier and speaker. Communication with the system controller took place via I2C or standard serial pins. The cost of this system was moderate and in fact the additional cost of this LCD was almost entirely offset by being able to use a much less complex microcontroller for the system controller. In hindsight, we would have chosen another LCD controller, as this one proved to have inadequate RAM for what we were trying to accomplish.

9.3 Programming the LCD

The 4D Systems LCD uses a proprietary coding language which was not easy to adopt. 4D Systems provides a robust set of visual and object oriented tools to program the system; however these tools do not work beyond their limitations. Ultimately the microcontroller ended up driving the user interface with less coding than we expected on the LCD side. This ideally allowed for rapid development and a parallel workload for prototyping the system, which would have allowed the UI and system controller code base to be developed independently and in parallel, but this was not the case in our project.

9.4 LCD and System Controller Interconnect

A few options were provided on both the system controller and the LCD module for communications between the two subsystems. A combination of these interfaces was used to enable the different functions necessary within the system. There was an I2C bus on both the LCD and System controller microcontrollers. This bus was relatively low speed and was used to send and receive process commands in between the two units. An API was written to abstract this functionality but a working set of the encoding of these commands is provided in the table. All commands implement a send and acknowledge format. If the receiving unit does not acknowledge the command after a set period of time the command will be re-sent. Multiple failed attempts will generate an error code and trigger a reset. The system controller implements a FIFO queue for storing and executing instructions. These can be explicitly managed be the LCD unit or loaded from the onboard queue in the case of a scan.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Bit String | Source | Destination |
| Execute Cooking Instructions | 0x0001 | LCD | SC |
| Execute the cooking instructions currently loaded into the queue. | | | |
|  | | | |
| Acknowledge Execute | 0x8001 | SC | LCD |
| Confirm that cooking execution has begun. | | | |
|  | | | |
| Pause Execution UI | 0x0002 | LCD | SC |
| Pause execution of cooking instructions. | | | |
|  | | | |
| Acknowledge Pause UI | 0x8002 | SC | LCD |
| Confirm that cooking has been paused. | | | |
|  | | | |
| Pause Execution SC | 0x8003 | SC | LCD |
| Pause initiated by a hardware event, most likely the door is open | | | |
|  | | | |
| Acknowledge Pause SC | 0x0003 | LCD | SC |
| Confirm that the UI has switched to pause mode. | | | |
|  | | | |
| Halt | 0x00FF | LCD | SC |
| Initiates an all stop and flushes the instruction queue. User has cancelled execution. | | | |
|  | | | |
| Acknowledge Halt | 0x80FF | SC | LCD |
| Execution stopped and queue flushed. | | | |
|  | | | |
| Hard Fault | 0xFFFF | SC | LCD |
| A serious error has occurred likely indicating hardware failure. This will be followed by a system error code. | | | |
|  | | | |
| Load Instruction | 0x0004 | LCD | SC |
| Load an instruction into the queue. The next line will be an encoded cooking instruction consisting of 4 bits encoding the power level, 11 bits encoding the time in seconds, and a flag enabling the turntable. | | | |
|  | | | |
| Acknowledge Load | 0x8004 | SC | LCD |
| Successful load confirmed. | | | |
|  | | | |
| Clear Last Instruction | 0x0005 | LCD | SC |
| Clear the last instruction in the queue. This would likely be triggered when a user is entering custom instructions and hits the back key. | | | |
|  | | | |
| Acknowledge Clear Last | 0x8005 | SC | LCD |
| Confirm that the last instruction in the queue has been cleared. | | | |
|  | | | |
| Reload Instruction Queue | 0x0006 | LCD | SC |
| User has canceled override entry. Clear the queue and reload instructions. | | | |
|  | | | |
| Confirm Reload | 0x8006 | SC | LCD |
| Reload successfully executed. | | | |
|  | | | |
| Enable Camera | 0x0007 | LCD | SC |
| Turn on the front camera and begin feed via RS422. The system controller will also automatically attempt to decode any UPC that may be found in the image. | | | |
|  | | | |
|  | | | |
| Acknowledge Camera | 0x8007 | SC | LCD |
| Camera is on and broadcasting. | | | |
|  | | | |
| Store Instruction | 0x0008 | LCD | SC |
| Store an entry in the cache associated with the current product ID and fill it with the current instruction queue. This will be followed by another word establishing which user to associate the instructions with. | | | |
|  | | | |
| Confirm Store | 0x8008 | SC | LCD |
| Store was successful. | | | |
|  | | | |
| UPC Alert | 0x8009 | SC | LCD |
| Broadcast to the LCD unit when a successful decode has occurred. There is no ACK for this command. | | | |
|  | | | |
| Fetch Product ID | 0x000A | LCD | SC |
| Requests the current product ID stored in the queue on the System Controller. | | | |
|  | | | |
| Sending Product ID | 0x800A | SC | LCD |
| This command is followed by the product ID currently in the queue. | | | |
|  | | | |
| Fetch Product Details | 0x000B | LCD | SC |
| Requests product details from the system controller that were not found on the LCD cache. This is followed by the product ID that is being requested. | | | |
|  | | | |
| Sending Product Details | 0x800B | SC | LCD |
| Will send a data packet containing a product description for the product ID requested. This should be stored on the LCD cache. | | | |
|  | | | |
| QR Alert | 0x800C | SC | LCD |
| The internal camera has detected a QR code and it has been decoded by the system controller. This alerts the UI of the event so that it can switch to the scanning output.  The System controller will also attempt to send the camera feed to the UI via the serial connection. | | | |
|  | | | |
| QR acknowledge | 0x000C | LCD | SC |
| The LCD has switched to the scan screen and is receiving the internal camera feed. | | | |
|  | | | |
| Fetch Instruction Queue | 0x000D | LCD | SC |
| Request the current instruction queue from the system control to coordinate cooking output. | | | |
|  | | | |
| Sending Instruction Queue | 0x800D | SC | LCD |
| A series of encoded cooking commands is sent back to the LCD unit from the instruction queue. | | | |

Table 20 - Interconnection Protocol

Summary of Hardware & Software

1.0 Hardware

* 4D Systems’ Micro LCD-32PTU Touchscreen LCD
* TI’s TIVA C Series TM4C123G Launchpad Microcontroller
* Fabricated Microcontroller – source to be determined
* TI’s CC3000 BoosterPack Wi-Fi chip
* Panasonic’s Luxury Full-Size 2.2 cu. ft. Genius Microwave Oven with Inverter Technology
* CMOS VGA Camera Module

2.0 Software

* Code Composer Studios - Microcontroller
* 4D Systems’ Workshop4 IDE - LCD Touchscreen
* SDK Installer (Rev.A) - Wi-Fi Chip
* Microsoft SQL Server Express Edition – Database Management System
* PCB Planner - Board Fabrication
* PCB Expedition - Board Fabrication
* DX Designer - Board Fabrication
* Microsoft Word - Project Documentation
* InDesign - Milestone Management
* Microsoft Visio - Project Documentation
* LTPOWERCAD – Power Design
* LTSPICE – Power Simulation

Prototype Construction & Coding

1. Parts Acquisition & Bill of Materials

In order to acquire the parts for our prototyped NOMS system we purchased the parts from the individual vendors themselves such as Texas Instruments, Sanyo, and DigiKey for the fabricated controller board. These parts were acquired by purchase online and delivered over the Winter break and throughout the spring 2014 semester. The fabricated board was donated as well as the population of it.

We prototyped our project using TI’s TIVA C Series TM4C123G Launchpad Microcontroller. After successfully implementing the prototype, we created our printed circuit board design using Eagle Schematic Capture.

2.0 PCB Vendor & Assembly

The controller board that we prototyped with was TI’s TIVA C Series TM4C123G Launchpad Microcontroller. This was off the shelf and did not require any additions. We designed our own controller board in parallel with the design of the other subsystems. The design of this board was loosely based on the design of the microcontroller board we used from Texas Instruments. In order to acquire this fabricated board we had a couple of options to select from.

The first option we considered is a company named Advanced Circuits. This company fabricates individual boards without a minimum size or quantity which was exactly what we need. Upon further research into this company’s offers we discovered that Advanced Circuits offers special deals for engineering students on their website. The offer that we could take advantage of if we do is the “PCB 4 Layer Full Spec for ONLY $66 Each”. This was pretty affordable for the group and a good back-up plan.[47]

The next possible option was available due to the employment of two of our group members. Jared Wach and Tim Miller both worked at Lockheed Martin as CWEP interns and were involved in electrical work. Jared Wach specifically worked on board design and has access to many design tools used in fabrication of controller boards in addition to the design process itself. It was offered that we may be able to design and fabricate our controller board at the Lockheed Martin Missiles and Fire Control facility in Orlando. This was a good option being that it would be free and included the potential help of professional engineers who would oversee the use of the board fabrication design tools.

However as the project progressed we utilized a company called Quality Manufacturing Services. This company is run by several UCF alumni and offered to populate our board design for free. In addition they also helped us find a group that would fabricate the board for free, Bare Board Group.

CADSOFT Eagle was utilized in the development of the PCB which houses both the TM4C123GH6PM Tiva series microcontroller as well as the CC3000 embedded Wi-Fi\_\_\_33. The primary design considerations were simplicity to control costs as well as signal integrity. For those reasons it was decided that all analog systems, primarily high voltage management would remain off of the board. The resulting PCB is a simple 2 layer board. The backplane is primarily a ground plane, but due to routing issues a couple of signals are also included in this plane. Most of the signals remain on the top plane as was the design goal. Antenna placement and design was also a concern. It was isolated according to design specifications included as part of the CC3000 reference design. The board is quite small and all ancillary components (resistors, capacitors, inductors) were chosen in the 0402 package. This helps keep undesirable electrical effects to a minimum. The downside to this choice is that they are so small, hand assembly simply was not an option. Therefore both board fabrication and assembly were subcontracted out.

In order to control the high voltage components while meeting the goal of isolating the mainboard from these high voltage signals an external relay assembly was used. This opto-isolated, dual channel, high voltage relay board is used to drive both the magnetron as well as the other high voltage components using simple 3.3V GPIO output from the microcontroller.

The board includes 5 connection headers for the various subsystems which are attached to the main board a table of which is given in table 21.

|  |  |
| --- | --- |
| Reference Designation | Subsystem |
| J1 | Power Connection |
| J2 | Door Switch |
| J3 | JTAG Header |
| J4 | Camera Header |
| J5 | LCD Header |

Table 21 - Table of mainboard connections

The resulting PCB schematic and board layout are presented as figures 47 and 48.

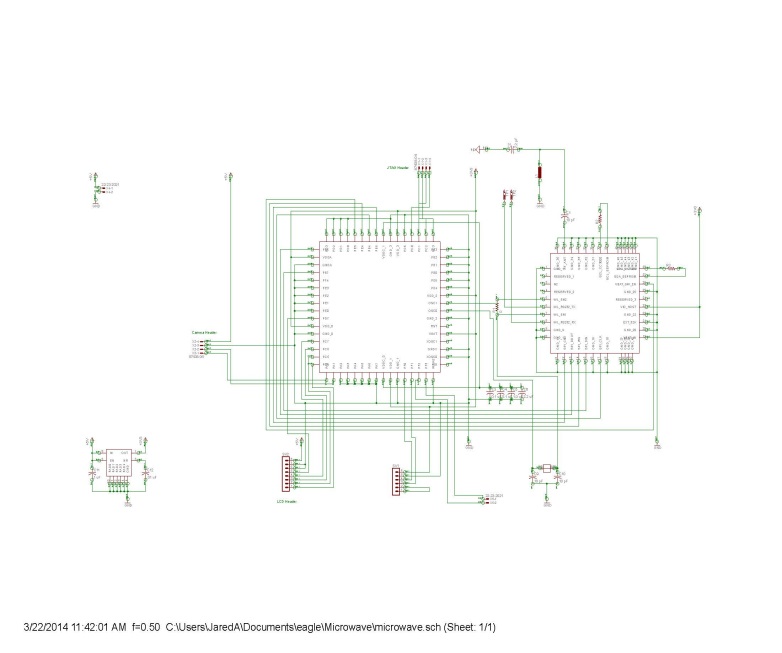


Figure 47- Board Schematic

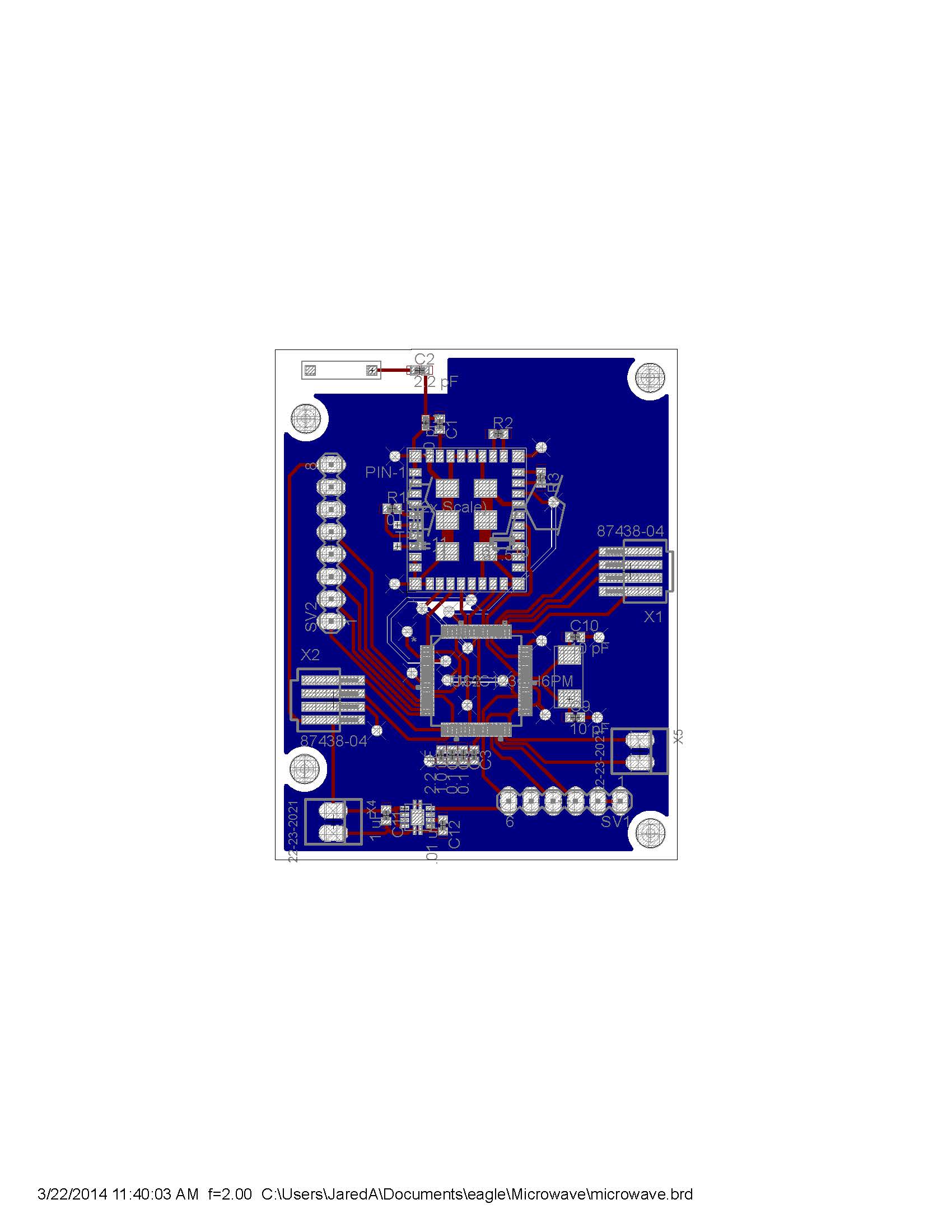


Figure 48 - PCB Layout

3.0 Final Coding Plan

The coding plan for each subsystem has been previously described in detail but will be summarily addressed here for the prototyping section. We used an FPGA to process the images captured by the camera(s) and used Verilog to write the algorithms that handled all the image processing involved. We used the development kits that came with the prototype microcontroller from Texas Instruments which followed syntax similar to C or JAVA. The same was true of the Wi-Fi chip selected as well. The LCD touchscreen had its own microcontroller and was programmed using the IDE that comes with it called Workshop. This was a free IDE.

4.0 Software

The overall software design philosophy was to have a minimal codebase. While it may have been easier to run the system on a small embedded operating system we just did not feel that the scope of the system justified the overhead. To that end, the entire system was coded as a bare bones project directly to the Tiva C series ARM processor. This introduced several difficulties along the way that were not foreseen in the planning stages of the project. The end result being that there was significantly more software coding necessary than was originally intended.

The microcontroller was actually the high point when it came to the software development for this project. It was originally assumed that this would be the most difficult, but in fact TI has provided ample software support for development on this particular microprocessor. Included as part of the development tools for this device is the TivaWare API. The provided functionality is actually quite robust ranging from powerful libraries to control the communications to embedded graphic libraries and objects for designing and driving a user interface directly from the microcontroller. All of these libraries are further embedded in ROM on the chip allowing them to be accessed quickly and without creating additional software overhead that must be loaded into the onboard FLASH memory. This portion of the software is primarily interrupt driven. The microcontroller waits in a low power state, only driving the clock, during most of the time. Interrupts are driven via UART primarily from the LCD UI unit. This is, unfortunately, where the easy development ends. The major blocks of the design are devoted to the LCD, camera, microwave, Wi-Fi, and bar code scanner.

The LCD was chosen because it offered a powerful system for driving the user interface which would be easy to design. Theoretically the 4D Systems Visi Genie environment provided a simple, visual development environment, which would allow the system to be designed with little to no coding necessary. This was desirable since as computer and electrical engineers our primary strengths are not software development. This unfortunately proved to be far from actuality. While it is true that Visi Genie provides a visual development environment which requires not coding it is extremely lacking in functionality. Even simple tasks such as running a clock proved to require convoluted workarounds to their system. In retrospect given the extensive tools provided by the microcontroller this system would not have been used and the user interface would have been driven directly by the microcontroller. The host driver for the LCD had to be written from scratch. No tools or API’s were provided by the manufacturer for ease of development. The LCD unit utilizes a message passing protocol to drive user events. The only functionality the unit provides is to trigger this message passing to the microcontroller when an item that is part of the UI has been pressed. All functionality needed to then be driven from the microcontroller to the UI via this same system. This was eventually accomplished and the UI works as desired but this was, and is not an optimal solution.

The camera presented similar issues to that of the LCD. The camera was chosen because it performs onboard image processing. The frame is captured, processed, and compressed into a JPEG onboard the device. This JPEG is then sent via UART to the microcontroller. Software libraries were supposed to be provided as a host driver for the device. Upon receipt of the device it was discovered that these libraries were not appropriate for our device. They were written in C++ and were specific to the Arduino. Very little of this code was usable and a significant effort was required to port the host driver to our microcontroller. Ultimately this device did in fact work as intended and despite the difficulties associated with porting it we are satisfied with its functionality.

The microwave underwent two iterations. The original microwave that we had procured utilized inverting technology to drive the magnetron. After much research it was discovered that it would simply not be feasible to drive this device with our system without heavily modifying the design of the system board. A second, simpler, microwave met our requirements. The magnetron can be driven on and off with a simple relay. Power levels are controlled via a duty cycle. Software specific to the microwave was extremely simple and represented the easiest portion of the design.

The Wi-Fi\_\_\_33 is the only portion of the code that worked as intended. This was not however without difficulty. The CC3000 is designed to be integrated with TI microcontrollers. A host driver is provided to this end. However finding the software proved difficult. The Tiva series of microcontrollers are quite new and the software repository for the CC3000 contained code only for the MSP430 line of microcontrollers. Calls to TI yielded no results. Finally a chance discovery in the example code provided as part of the TivaWare API for the microcontroller allowed us to locate the proper host driver for the device.

Lastly the scanner: while this would initially appear to require significant design, this functionality was chosen with the knowledge that there is a powerful open source library available specifically for this task. ZBar is a library which can, in theory, be adapted to any system. From embedded design up to visual development environments. The underlying code is written in C and then wrappers are provided for more abstraction. What quickly became apparent upon importing it into the project was that the library really was intended to be used with an embedded operating system. The library cannot be easily built without configuration. This configuration can only be done in a Linux environment. This meant that the code had to be analyzed, modified, and adapted directly from the source code. This was a significant effort but was still preferable to writing the routines from scratch. The end result is a powerful system that can not only identify and decode bar codes from the JPEG provided by the camera but also converts it to a packed RBG format appropriate for displaying the output via the LCD.

Prototype Testing

In order to verify that the prototyped NOMS system was working correctly we had a testing plan for each phase of development with test cases and ways to verify that each test case is passed. For each subsystem, we previously identified ways in which we tested and verified that the subsystem itself is working. For example, the camera system has its own set of requirements that needed to be met each with ways to verify that the requirement was met and will working properly. Once each subsystem was tested thoroughly enough that it passed we integrated each system into the whole and verified through more testing that the entire system was working correctly and that each subsystem coordinated with the others properly. This section of the document addresses which environments used to test the prototyped NOMS system and then addresses some specific tests that we used to ensure the prototyped NOMS system worked correctly. These same tests were also used to verify that the final product worked correctly once we fabricated our own microcontroller board.

1.0 Test Environments

Code Composer Studios was used to write and test the system controller payload.

4D Systems’ Workshop4 IDE was used to write and test the LCD microcontroller’s functionality and interaction between itself and the main system controller.

DX Designer was used to create a circuit card layout in the high level basics. Testing at this level included testing for proper line termination, hanging nets, and proper system design.

PCB Expedition was used to lay out all the details of the card layout. Testing at this level was detailed and includes things like testing for noise, thermal hotspots, and more detail-oriented issues.

2.0 Specific Testing

We wrote an API that allowed the microcontroller that we prototyped with to receive packets over an I2C bus. These packets were directly routed to the LCD unit to ensure communication was correct. Verification was done visually, by inspection on the LCD itself. We checked for correct imaging, white balancing, de-skewing, and de-warping of the image.

In order to save power and result in a higher efficiency we caused the system controller to enter a sleep-mode when it had not been in use for a specified period of time. In order to achieve this we created an internal running clock, which when it reached a certain period of time that it remained dormant, will switch the device to sleep-mode.

Another event we considered was when the microwave door was opened. This caused the microcontroller to turn on from sleep mode. This assumes that the microcontroller was in sleep-mode to begin with and was tested from the previous test case.

In addition, opening the microwave door should turn the interior camera on. This could be tested very simply by checking the camera’s clock to see if it is running. Whenever the clock is running the camera is scanning for images, sending those images to the buffer and these images are eventually processed by the FPGA. The FPGA will process the images and send these processed images to the microcontroller for use. We were not able to implement this in our design, due to lost team member.

Whenever the interior camera is on it should be scanning for QR codes for a specified period of time based on hardware ability in the camera buffer. This can be tested in a similar manner as the previous test case. We were not able to implement this in our design, due to lost team member.

The LCD comes out of sleep-mode and lights up whenever the door is opened so that the user will see that NOMS is aware of the user and is ready for use. This test required a visual inspection to verify that the system was working together properly.

We ensured that proper communication was taking place from the microcontroller to the microwave system so that when the appropriate signal was sent to the relay switch, the microwave oven turned on (magnetron turns on).

The microcontroller communicates through a bus to the LCD touchscreen. We tested this communication link to ensure that the user was able to properly communicate with NOMS. We used the JTAG programming interface to verify that the microcontroller bus was sending and receiving proper signals using the Code Composer Studio.

There were a number of simple tests that took place between the LCD touchscreen and the microcontroller. When a manual instruction was given by the user on the LCD, the LCD’s own controller processed those instructions and sent them to the microcontroller. We verified that when these instructions were passed along that they were passed along correctly. The testing of the LCD’s processing took place within the subsystem’s own test procedure as seen in its own test plan. This testing took place between the bus that connected the LCD touchscreen to the main system controller in a similar way to the previous test.

An initial test we ran was the verification that the correct UPC or QR codes were being recognized as they were being scanned. In order to do this we started with a hardcoded image that was sent to the microcontroller and ran through the decoding algorithm. This image had known EAN-13 code. We verified that the algorithm worked correctly through a built-in test in the code that verified that the EAN-13 code was the same one that we already knew it should be. This test was repeated several times we ensured that there wasn’t any lucky guesswork going on in the decoding algorithm.

When the camera has successfully scanned an image of a UPC code or QR code we wanted the camera to enter into sleep-since it had fulfilled its purpose temporarily. In order to do this, however, the microcontroller needed to send the signal to it that a valid EAN-13 code was found and was being searched for in the online database. We verified that the camera had ceased scanning for images. This was done in two ways: the first was by inspection since we had the images being scanned by the camera as output on the LCD screen. The second way disregarded the LCD screen since we could technically stop outputting to the LCD while the cameras continue to scan for images anyway. This test ensured that the power to the camera’s clock was cut off. When the clock for the camera was off then the clock itself stopped scanning images.

The interior camera was not a required camera to have in the NOMS system but would still need to be tested if and when the NOMS design team decides to implement it into the system. We were not able to implement this in our design, but included below is our considerations for this feature.

Similar to the last test case, we will want the interior camera to stop scanning for any images when the magnetron is turned on since there should be no need to images once the microwave is operating and also because the radiation will disturb the process of image capture anyway. In order to ensure that this is the case we can actually also ensure that the interior camera is never on unless the microwave door is open. This will be achieved by cutting off the power to this camera from the microcontroller whenever the door is closed. The interior camera, therefore, will only be functional whenever the microwave door is open, and also only for a specified period of time based on the camera buffer.

We ensured that the cooking instructions that were received from the online database were sent from the database’s server to the microcontroller correctly. This was verified by causing the microcontroller’s algorithm to output the cook times and power levels in the development environment. We were able to verify between the two that they match.

NOMS needed to have a manual usability section on the LCD so that a user who chooses not to use the automated aspects of NOMS or has food that cannot be scanned can cook their food. These were achieved through the LCD touchscreen which was required to send the proper signals to the microcontroller indicating a cook time and power level for the microwave to use. This was verified in two ways: we inspected to see that the microwave turned on and for the allotted time, though power level was difficult to track. We also verified in the code development environment for the microcontroller that the signals received were identified correctly and lead to the correct event of turning on the microwave for the specified period of time and that the power level was set correctly as well.

3.0 Conclusion

N.O.M.S. has been a challenging project. There were many obstacles that we had to overcome. We lost a team member between semesters, which means this team and this project had to adjust to the sudden change. With regards to N.O.M.S. we found some difficulty in communication between the individual components. However, after much patience and problem solving, we were able to make all components communicate with each other, in spite of the difficulties. We also had difficulty when it came to dissecting the microwave, as most companies do not include the full electrical manual available to consumers. We had to go through multiple microwaves before we found one that we were able to understand.

There was, unfortunately, a lack of support from TI in particular that was assumed to be in place during components selection. The host drivers for the CC3000 were not available and caused a great deal of difficulty in implementation. Additionally the LCD unit suffers from finicky communications components. This means that the desired and advertised communication speeds were unable to be reached. Ultimately these difficulties affected both development time and the resulting performance of the system. However we are satisfied with the results and proud of our accomplishments on this project.

.

Operations

When users purchase the N.O.M.S. operating system they will need to complete minimal setup before the microwave will be functional. Upon plugging the microwave in, the users will need to set the time. By pressing the time button on the home scree, the users will be prompted to input the time. After setting the time the microwave is ready for operation.

In order to initiate the scanning process the user presses the prominent “Scan Now” button located on the home page. Users will take the food package and hold the barcode facing upright towards the camera located on the front panel of the microwave. The user will reference the LCD user interface for alignment feedback. The user might need to readjust the package. Once the system recognizes the product, text will appear underneath the video feedback identifying the item. The user will press start and be taken to a screen which will contain the cooking instructions. The user will either press start to activate cooking or will press manual to be taken to the manual entry screen to input cooking power and cook time.

When the user presses start, the microwave will begin cooking at the appropriate cooking time and power. While on the cook screen, N.O.M.S. has the ability to display ads relevant to the product being cooked. Companies will have a means to directly reach their clients. This has potential in the future to generate useful data for companies to purchase so they can see consumer trends.

The user will also have the option of pressing the manual cook on the home screen. This will take users to a traditional microwave screen where they can input the cooking power and time. Upon completion, N.O.M.S. buzzer will go off to let users know that their product has finished cooking.

Administrative Content

1.0 Budget

|  |  |
| --- | --- |
| LCD Touch Screen and Controller | $80 |
| Embedded Wi-Fi Chip | $40 |
| C-MOS Cameras and Controllers | $20 |
| Embedded Systems Controller Components | $15 |
| Fabricated Board | $70 |
| Microwave | $250 |
| Magnetron: 10QBP1009 | $75 |
| Turntable: 15QBP0851 | $33 |
| Fan Motor: | $33 |
| Lights | $5 |
| AC/DC Transformer: F606Y8M00AP | $155 |
| Buzzer | $2 |
| Regulator | $8 |
| **Total** | $786 |

Table 22 - Budget

Figure 49 - Budget

2.0 Budget Discussion

The financial aspect of NOMS was handled collectively by the group members themselves independent of grants or help from outside companies. We therefore selected the most cost-effective parts available to us. The previous budget list was a comprehensive list of all known costs that we needed to cover in order to complete NOMS. We have found a couple of microwaves that were free, but were not able to use them for this project, as the free microwaves had inverter technology that we were unfamiliar with and unable to grasp. We had to purchase a microwave at a local store to find a model we were sure we could work with. The list above also includes some redundant parts such as a magnetron and a microwave. Every microwave that can be bought usually contains the magnetron otherwise it is unusable. We included such redundancies in order to offset any complications that came up in the design and building stage. Lastly, the fabricated board that we created came to us free of charge. In addition to these savings, Texas Instruments offered samples on many of their products including the ones that we used in this project. We ordered multiple of particular parts to anticipate hardware failures or damage to any electronics in the process of installation, integration, or testing, which we were grateful for the foresight as we ended up needing these extra parts as we built this project.

3.0 Milestone Discussion

Milestones are an important part of any large research project. Determining proper milestones and schedules at the beginning of the semester provides a quick overview of where we need to be as the semester progresses, as well as provides an indication as to whether we are falling behind or not. As a group of three who each had jobs outside of school, it was very important that everyone remained on the same page. By having established goals and weekly meetings, we were able to keep ourselves on track, as well as our teammates. Without this structure, our project would have fallen apart very quickly as members fell behind. This would have created a very stressful working environment, and would have hindered any creative communication that could come from working within a group.

The first semester was all about deciding on a project and producing a paper. Our group was chosen quickly enough, and a project decided upon. This gave us an advantage, as we were quickly able to start brainstorming ways to make this project more than just your typical microwave project. We pulled our minds together and came up with the idea of a microwave that would be intelligent enough to prepare food, based on bar and QR codes. The additional perk of pushing the food consumption data to a fitness website came as a result of brainstorming sessions. Luckily for us, some of our group members had experience with hands on electronics, which will come in handy next semester.

The first semester’s milestones included breaking the project apart and assigning responsibilities to teammates. We then had to meet with our advisor to make sure our project provided enough design challenge. From there, our main focus was to obtain as much research as possible to aid us with the task of writing the paper towards the end of the semester. We had some deadlines of turning in preliminary research, as well as a rough draft paper submission. We needed to schedule our time to make all of these goals.

We established meetings every other week to make sure everyone was doing their part and this also gave us a chance to communicate with team members along the way. We expected this project to be challenging. We were not disappointed. This project proved to be an enormous undertaking which would not have been possible without the use of team meetings and group goals.

In the images below, we have provided the milestone discussion per senior design class.

## Semester 1: Senior Design 1 Milestones

In table 22, we discussed the milestones and the dates we planned to achieve these goals.

|  |  |  |  |
| --- | --- | --- | --- |
| **Task Name** | **Initial Date Planned** | **Date Process** | **% Successful** |
| Project decision | 08/30/13 | 08/30/13 | 100% |
| Project discussion and splits of categories for each person | 09/04/13 | 09/04/13 | 100% |
| Meeting with Dr. Richie to discuss the project | 09/22/13 | 09/22/13 | 100% |
| Start researching | 09/04/13 | 09/10/13 | 100% |
| Finished researching | 11/24/13 | 11/30/13 | 100% |
| Start paper | 09/15/13 | 09/18/13 | 60% |
| Complete 10 Pages for Rough Draft Review | 11/05/13 | 11/07/13 | 90% |
| Set date to review/ rough paper | 11/22/13 | 12/1/13 | 100% |
| Meeting 1: Research | 08/31/13 | 08/31/13 | 100% |
| Meeting 2: Research | 09/14/13 | 09/14/13 | 100% |
| Meeting 3: Research | 09/28/13 | 09/28/13 | 100% |
| Meeting 4: Research Paper | 10/26/13 | 10/26/13 | 15% |
| Meeting 5: Research Paper | 11/02/13 | 11/02/13 | 30% |
| Meeting 6: Research Paper | 11/16/13 | 11/16/13 | 60% |
| Meeting 7: Research Paper/ Final Edits | 12/01/13 | 12/01/13 | 100% |

Table 23 - Semester 1 Milestones

Our goals and milestones for Senior Design 2 on our NOMS project were all in theory of what we believed and hoped would happen. The first real milestone on our chart was the database development for our project. We ordered all of our parts during winter break, but we allotted extra time for complications. Parts acquisition occurred before the semester started.

From there, we worked on and tested all of the parts as we build them, as we did not need the entire microwave to be created in order to ensure individual components worked. We worked in teams to provide support during the construction process. We were not able to finish our project by early April, as the schedule below predicted. We finished testing this project the day before our presentation.

## Semester 2: Senior Design 2 Milestones

In figure 48 we discuss our senior design 2 milestones that we hoped to achieve, as well as the timeline we hoped to keep while completing this project. We did not keep this timeline as predicted, and had to change our goals and dates in order to keep up with the way things happened. We were ultimately able to complete our project in time for our demonstration, so we considered this a success, even if it was not as early as expected.

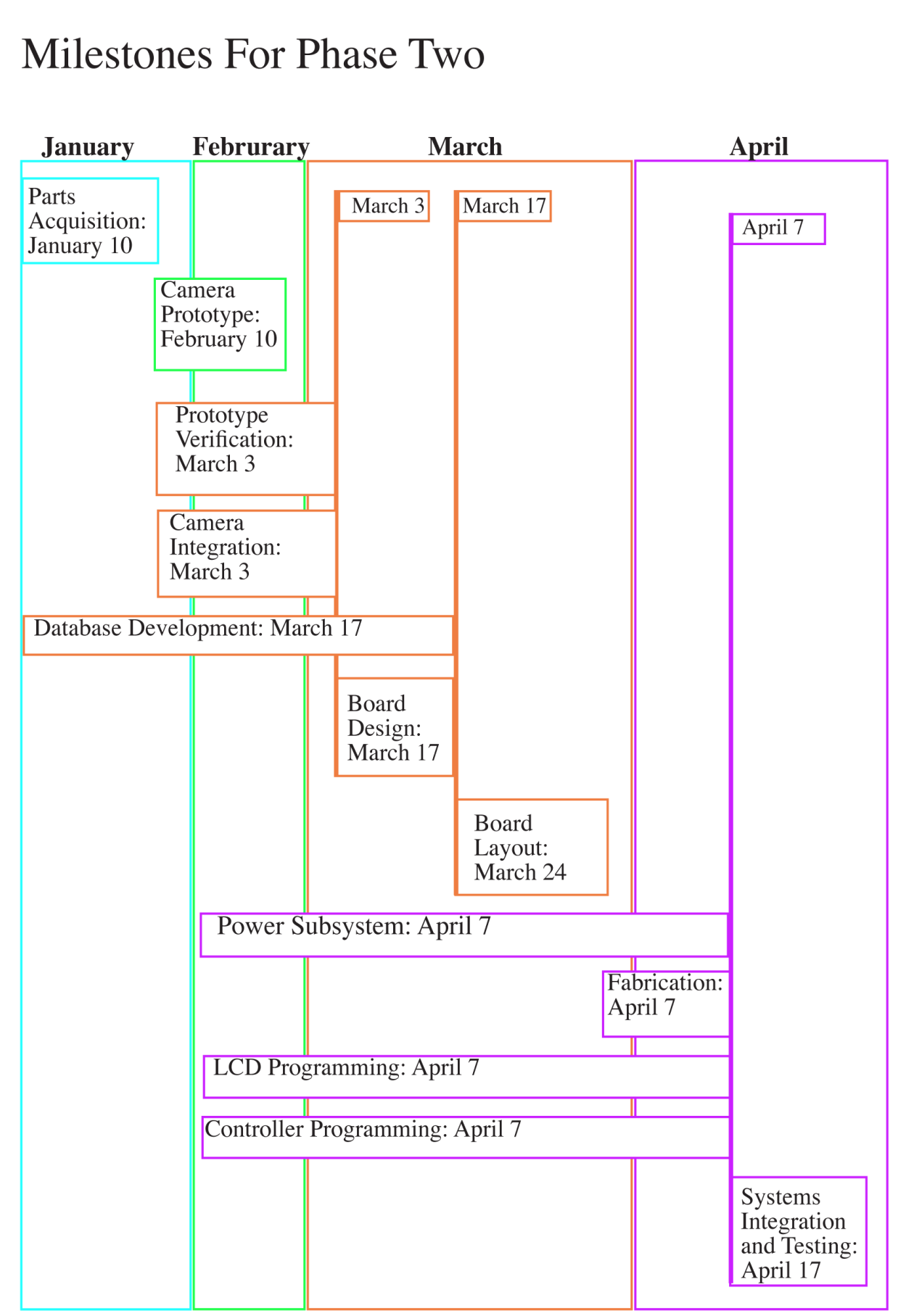


Figure 50 - Semester 2 Milestones

7.0 About Us

Nakiesa Faraji-Tajrishi

Nakiesa is an electrical engineering major in her senior year at UCF, graduating in May 2014. She comes from a line of engineers including her father who is an electrical engineering consultant for power management. Her brother is an electrical engineering major at Georgia Tech and is graduating this May as well. Nakiesa originally obtained an undergrad degree in Studio Art at Florida State University. Nakiesa currently is interning at Siemens, as an electrical engineering intern in the wind power division. Nakiesa will accept a job offer from Siemens in the coming weeks.

Tim Miller

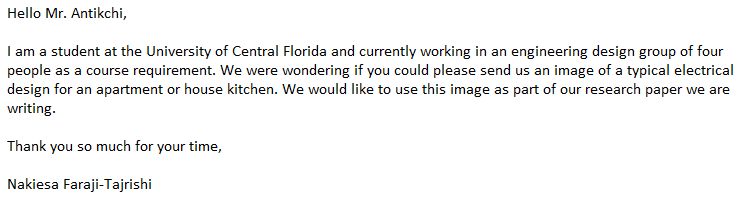
Tim is a computer engineering major in his senior year at UCF, graduating in May 2014. He received his Associates in Arts from Valencia College in 2011. Tim was a CWEP intern at Lockheed Martin in Systems Engineering and has recently accepted a fulltime position upon graduation at Lockheed Martin MFC’s site in Orlando as a Systems Engineer. Tim’s father was an electrical technician for the United States Navy and worked for NASA for over twenty years before moving on to Tektronix. His grandfather was a chemical engineer at Martin as well. Tim’s younger brother also is seeking to become a nuclear engineer for the United States Navy where he will pursue a degree from a University as well in order to further pursue a career in engineering. Coming from an engineering family Tim eventually decided to also pursue engineering as a career. Tim has accepted a job offer from Lockheed Martin.

Jared Wach

Jared is a computer engineering major in his senior year at UCF, graduating in May 2014. He previously earned his Associate of Arts from Lake Sumter Community College. He is a non-traditional student coming back after having successfully run a small business here in Central Florida. Air Central operated for nearly a decade before the recession caused the construction industry to bottom out. Seeing an opportunity to pursue his passion of computer engineering he returned to UCF where he is now finishing his degree. He does this despite the challenges of having his concurrently in the UCF College of Nursing and three young children. Jared has accepted a job offer from Lockheed Martin.

Appendix A – Permissions

Question:



Response:



Appendix B - References

[1] <http://www.smarthome.com/13041/Beyond-Microwave-Oven/p.aspx>

[2] <http://madebynathan.com/2013/07/10/raspberry-pi-powered-microwave/>

[3] <http://www.hobartcorp.com/products/cooking/combi-ovens-and-barcode-scanners/>

[4] <http://www.eecs.ucf.edu/seniordesign/fa2012sp2013/g31/>

[5] <http://www.eecs.ucf.edu/seniordesign/fa2010sp2011/g15/>

[6] <http://www.howstuffworks.com/microcontroller.htm>

[7] <http://www.esfi.org/index.cfm/page/National-Electrical-Code-%28NEC%29/pid/10860>

[8] <http://www.nbcnews.com/id/27055169/>

[9] <http://www.madehow.com/Volume-1/Microwave-Oven.html>

[10] <http://suite101.com/a/microwave-ovens-heat-food-unevenly-a35367>

[11] <http://scitech.web.cern.ch/scitech/toptech/01/MicroWaveOven/microwave_2.shtml>

[12 <http://www.eecs.tufts.edu/~dsculley/tutorial/diodes/diodes3.html>

[13] <http://www.electronics-tutorials.ws/transformer/multiple-winding-transformers.html>

[14] <http://www.electronics-tutorials.ws/diode/diode_6.html>

[15] <http://www.allaboutcircuits.com/vol_3/chpt_3/8.html>

[16] <http://electronics.howstuffworks.com/brushless-motor.htm>

[17] <http://www.anaheimautomation.com/manuals/forms/brushless-dc-motor-guide.php>

[18] <http://www.s-line.de/homepages/bosch/sensorless/node9.html>

[19] <http://www.solarbotics.net/library/pdflib/pdf/motorbas.pdf>

[20] <http://www.engineersgarage.com/articles/stepper-motors>

[21] <http://www.radartutorial.eu/08.transmitters/tx08.en.html>

[22] <http://hyperphysics.phy-astr.gsu.edu/hbase/waves/magnetron.html>

[23] <http://www.powerqualityworld.com/2011/04/cbema-curve-power-quality-standard.html>

[24] <http://www.maineelectronics.com>

[25] <http://www.repairclinic.com/Panasonic-Microwave-Motor-Parts>

[26] <http://www.repairclinic.com/Microwave-Blower-Wheel-Fan-Blade-Parts>

[27] <http://www.repairclinic.com/Microwave-Lighting-Light-Bulb-Parts>

[28] [www.erparts.com](http://www.erparts.com/)

[29] <http://www.murata.com/products/catalog/pdf/p37e.pdf>

[30] <http://www.1728.org/project3.htm>

[31] <http://www.ixysic.com>

[32] <http://www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/CD00000444.pdf>

[33] <http://www.gs1.org/docs/gsmp/barcodes/GS1_General_Specifications.pdf>

[34]http://webstore.ansi.org/RecordDetail.aspx?sku=BS+ISO%2FIEC+18004%3A200

[35]http://www.ti.com/msp430/

[36]http://www.ti.com/arm/

[37] http://www.ti.com/arm/

[38] <http://www.personal.psu.edu/glh10/ist110/topic/topic07/topic07_01.html>

[39] <http://office.microsoft.com/en-us/access-help/database-basics-HA010064450.aspx>

[40] <http://www.sqlite.org/>

[41] <http://www.mysql.com/>

[42] <https://www.microsoft.com/en-us/sqlserver/default.aspx>

[43] <http://windows.microsoft.com/en-us/windows-vista/set-up-a-computer-to-computer-ad-hoc-network>

[44] <https://my.bluehost.com/cgi/help/541>

[45] <http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/point-multipoint-rfmodules/xbee-wi-fi>

[46] <http://www.ti.com/product/cc3000>

[47] [www.4pcb.com](http://www.4pcb.com)