The Memrowave: Microwaving the future

Darren Armstrong, Jack Gulick, Joseph Serritella, and Winston Todd

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* — The advent of microwave ovens over a half century ago, cutting the overall time needed to prepare dishes. Meals could be prepared with minimal effort and on the go. Demand for quicker, faster, and simpler appliances produced what is now the Memrowave. The Memrowave approaches the demand by integrating barcode scanning, database storage, Wi-Fi connectivity, and standard microwave features, producing a smart microwave. The approach uses barcode scanning to quickly determine cook times and product information. The Memrowave uses the retrieved preparation time to control magnetron power, thus, cooking the product. By combing these features a simpler microwave is created.**

***Index Terms* — Home automation, microcontrollers, microwave ovens, query processing, multimedia databases, wireless LAN, bar codes**

I. Introduction

Microwave ovens were introduced to the average consumer over a half century ago, cutting the time needed to prepare meals. Demand for quicker, faster, and now smarter appliances is the driving force behind production of the Memrowave. The Memrowave is a UCF engineering Senior Design project. The main purpose of the Memrowave is to produce a smarter and more autonomous microwave capable of cooking multiple foods with ease and limited user input.

Integration of barcode scanning, database storage, Wi-Fi connectivity, and the standard features of a microwave oven will automate the Memrowave. An automatic timer is incorporated, through the use of barcode scanning, into the system which will make the user input much simpler and more efficient. Barcode scanning the cooking configurations are based on the code of the food package. Product information can be sent directly to the Memrowave, allowing it to begin the cooking process once the door is closed. This approach eliminates the hassle of identifying proper cook times and different power levels of a product.

Along with barcode scanning, a food product database will be setup to log all the specific information needed to cook foods. Both local and internet databases will make use of a product's UPC code to index the information. Indexed codes are referenced when the product is recalled to cooking. Non-local databases require an internet connection, to accomplish this Wi-Fi connectivity is integrated into the system. A Wi-Fi connection will allow the Memrowave to send notifications to the user's phone or mobile device, granting the user freedom to roam as food is being prepared.

All the Memrowave's features are incorporated though an interactive LCD touch screen. The main user interface will be displayed on the LCD touchscreen fabricated to the front of the microwave. The user will be able to access all the smart features of the Memrowave, as well as using the Memrowave as a regular microwave.

A. Competing designs

Other senior design teams and appliance manufacturers have made efforts to make appliances smarter by designing mobile applications that serve as a remote control. The mobile applications have been demonstrated to work with washing machines, refrigerators, and dishwashers, to report status changes and when the user input is needed. Competing smart microwave designs are accomplished through touch-type interfaces and numerous preset configurations to cook multiple foods.

B. Component overview.

The basic components used in designing the Memrowave include a magnetron, internal lights, internal beeper, power switch, turntable motor, and power supply. The more complex components include the processing unit, printed circuit board, camera module, WIFI module, and LCD touchscreen. Divisions between hardware and software form the basis of the design plan. Hardware includes the powering devices, operation of switches, PCB assembly, and final enclosure assembly. Controlling of the multiple peripherals, constructing the user interface, web database development, and barcode scanning implementation are focused on the software portion of the design. Overall the Memrowave will be the go to appliance for preparing foods when time is limited, through the ease and consistency of an autonomous system.

II. DESIGN REQUIREMENTS

Discussed are the objectives set out for the Memrowave project. In this section the features included in the Memrowave are outlined in greater detail.

A. Automatic Timing and Power Levels

After a user scans a product's barcode, the Memrowave must automatically set the timer and power level, provided the Memrowave has access to this information. If, for whatever reason, the Memrowave cannot determine the appropriate cook time or power level, it will ask the user to enter this information; however, this should only occur once, because the Memrowave will store this information for the next time this product is scanned.

B. Local Database

The Memrowave must continue to work in situations where there is no internet access. To achieve this, we would need to include a local product database. This database would store all of the relevant information for a particular food product, including the UPC, name, description, preparation instructions, cook time, and power level.

When a user scans a food item using the barcode scanner on the Memrowave, it will first search the local database for an entry with a matching UPC. If the UPC is not found in the local database, the Memrowave will then attempt to connect to the internet and retrieve this information from the master product database, which is made available through some web API. If the product cannot be found in the master database or if the Memrowave does not have internet access, then the user is asked to manually enter the cook time of the product. This manual entry will be saved in the local product database for later use. A flowchart of this process is shown in Fig 1.

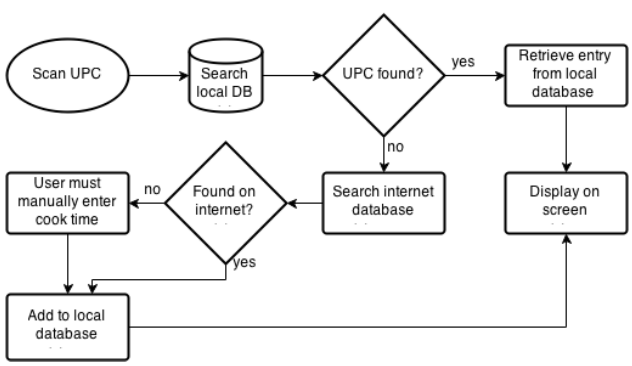

Fig 1. UPC database process flow. A barcode is scanned and processed by the Memrowave. If the UPC exists the product is cooked. If absent the product is added to the database.

C. Touch-Screen Interface

The Memrowave must feel like a real smart device, so it needs a touchscreen interface. The touchscreen must be large enough to read without straining, and it must be large enough for the user to easily interact with the software; however, it cannot be too large that it will not fit into the control panel of the microwave. An LCD in the range of 4" to 7" diagonally should meet these size requirements. The user interface will have a clean, 'smart' look. It must be easy to use, and the control elements must be large enough to select easily on the touchscreen. The touch interface is quick and responsive.

The interface has a well-defined, intuitive layout and all of the Memrowave's settings are accessible from one central location within the software. The settings are stored using non-volatile storage, so that they do not reset after a power-cycle.

D. SMS Notifications

With a normal microwave, the user must wait within earshot of the microwave to hear when it is finished. With the Memrowave, we'd like to enable the user to know exactly when the microwave finishes cooking, even out of earshot. To achieve this, we decided that the Memrowave will have an SMS notification feature. The SMS notification feature will allow the Memrowave to send text messages to the user's cell-phone. These messages will keep the user informed about the state of the Memrowave as it cooks the food. For instance, if the user is defrosting food, the Memrowave will send an SMS notification instructing the user to flip over the food to defrost the other side. This feature could also be used to notify the user when their food is finished cooking, when the food's cool down time has been reached, when to stir the food, etc. Basically, this feature can be used to send any message from the Memrowave to the user. The SMS messages will be sent over WiFi using Twilio [1], an SMS gateway service that allows text messages to be sent to mobile phones via standard http requests.

E. Manual Cooking

Sometimes a user may not want to use the smart features of the Memrowave, or the smart features may not work well for certain types of cooking and recipes. For cases such as these, standard manual operation features are necessary, so the Memrowave must provide these features. The manual operation features include basic cook time selection, start and stop buttons, power level selection, and any of the basic features you might find on a standard microwave.

III. Hardware

The Memrowave is a conjunction of many different technologies into one 'smart' device. These technologies include magnetron control, ARM microprocessors, computer vision, display technologies, etc. In the following sub-sections, we will go into detail about each hardware component used when designing the Memrowave.

A. ARM Microprocessor

While most ARM microprocessors will have adequate computing power for the Memrowave, there are two general and fundamental requirements that are used to determine which ARM microprocessor was used. The first requirement was that it must be capable of running Android out of the box, preferably Android 4.x or higher. Android was selected to aid software development such as the user interface and food preparation database. The second requirement was that the microprocessor must be available in an experimenter board that can in turn easily connect to add-on expansion boards similar to Arduino shield add-on boards.

The only ARM microprocessor that meets the requirements of Android support and expansion board capability is the Texas Instruments Sitara AM335x ARM Cortex-A8 Microprocessor used in the Beaglebone Black. Figure 2 details the specifications for the Sitara AM3358 microprocessor[2, 3].

ARM CPU	ARM Cortex-A8
ARM MHz (Max.)	1000
ARM MIPS (Max.)	2000
On-Chip L1 Cache	64 KB (ARM Cortex-A8)
On-Chip L2 Cache	256 KB (ARM Cortex-A8)
On-Chip Memory	128 KB
Graphics Acceleration	3D
DRAM	1 16-bit (GPMC NAND flash NOR Flash SRAM)
USB	2
MMC/SD	3
PWM (Ch)	3
Real Time Clock	1
I2C	3
DMA (Ch)	64-Ch EDMA
IO Supply (V)	3.3

Fig 2. Specifications for Sitara AM3358 microprocessor.

The Beaglebone black will be connected to the Memrowave's PCB. Pairs of 46 pin headers on the Memrowave board will then connect the LCD touch screen display, therefore linking the LCD and Beaglebone [4]. It is important that the LCD touch screen display chosen is a Beaglebone Black expansion board, known as a "cape", or can be modified to fit the Beaglebone Black's headers. The Memrowave's PCB will be sandwiched between the LCD and the Beaglebone Black using matching 46-pin headers. The block diagram of the board layout is depicted in figure 3.

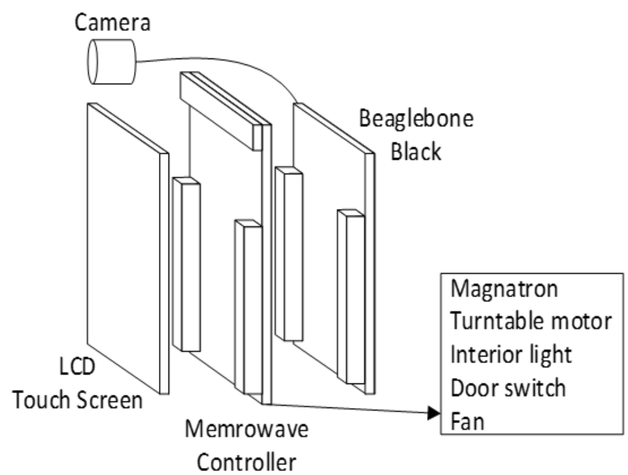

Fig 3. The Memrowave's components are placed in cascade; LCD screen, Control PCB, Beaglebone Black. This configuration was determined to be the most form fitting. Using this method allows for use of female and male headers, limiting wire use.

B. Controlling Microwave Components

The Memrowave brings together all of our subsystems through a control system. This controller will handle tasks such as controlling the magnetron, lights, turntable, and fan. The control system acts as the salve to a master. The Beaglebone, the master unit, communicates to the control PCB though I^2C, controls the magnetron and the other components. Controlling these elements are vital to the Memrowave. This section discusses the way the magnetron, lights, fan, and turntable are controlled.

Control of the microwave's preexisting electrics requires the fabrication of a PCB. The control PCB is fitted between the LCD and the Beaglebone black using two sets of 46-pin headers. The PCB consists of four main elements; a microcontroller (Specifically, the MSP430), a mechanical switching relay for the magnetron, an 8A solid state relay for all other electronics, and a MAX6958 LED driver for a 7-segment display.

The MSP430 and MAX6958 communicate with the Beaglebone black through I^2C. When the Memrowave receives a command the Beaglebone (The Master) requests the MSP430 and MAX6958 (The Slaves) to execute commands. For example when a product is scanned the Beaglebone processes the required cook time. The product information is sent to the MSP430, in turn, activing the magnetron relay and 8A relay, running the magnetron, light, fan, and turntable. The MAX chip will also receive a commands to turn on the 7-segment for a countdown timer.

Each relay is connected to an output pin on the MSP430. To minimize the required output current from the MSP430 pins, a transistor switch is used [5]. Using a transistor will provide the required current gain to flip each relay without drawing too much current form the MSP430. Each component on the control PCB requires 3.3V to operate.

C. Power Supply

Since both AC and DC power is needed within the Memrowave, a way for converting AC to DC is needed. An AC DC inverter will be used to convert the 120V AC domestic outlet to 12V-20V DC. The inverter must also be able to meet our total DC current needs as well. To acquire the multiple voltages needed for the DC devices, voltage regulation will used based on the specific DC devices that we will chose, ideally a maximum of three different voltage regulators should be sufficient. Another possible way of obtaining the dried voltages is to build the voltage dividers as well on the PCB, this option would save space as well as money and more accurate voltage could be produced. The use of voltage regulator components brings the advantage of a more robust system, a current loading limits can observed much easier based on the device specifications and with voltage regulators a more precision DC voltage can be obtained.

The power supply for the DC components uses two LM2576T switching regulators, one steps down to 5 volts powering the Beaglebone and LCD. The second linear regulator steps down the voltage to 3.3 to power the control PCB. 120Vrms is applied to a 120V/20V transformer, the AC signal is then rectified to create a 20V DC signal. The signals are then stepped down to the required voltage using switching regulators. Figure 4 represents the circuit used.

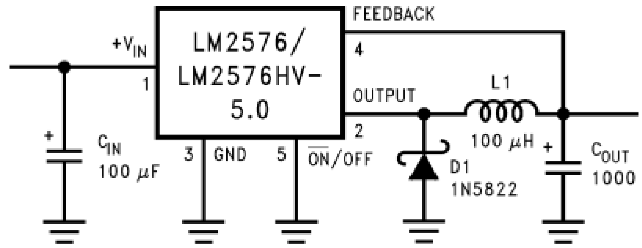

Fig 4. The Memrowave's switching regulator design. A 120Vrms signal is applied at a 120V/20V transformer, the AC signal is then rectified to create a 20V DC signal. The signals are then stepped down to the required voltage using switching regulators.

The output pin of the regulators are connected to a barrel jack for the Beaglebone and a molex connector for the control PCB. The Power supply is solder to a prototyping board.

D. Wi-Fi Interface

Internet connectivity gives the Memrowave's software access to an online products database that can be queried for items that are not found in the local database. This keeps the local database as small as possible, a desirable outcome when working with a limited amount of memory. Only the user interface software running on Android will require Internet connectivity. This connectivity will be provided via a USB Wi-Fi device attached to the Beaglebone Black's USB port. Kitchen appliances tend to remain in use for many years and are not replaced nearly as frequently as smart phones and other smart devices. It is likely that Wi-Fi standards will change during the typical lifespan of a microwave. Using a USB device has the advantage of being easy to replace or upgrade. We will be using the MediaTek MT7601 USB Wireless-N WiFi interface.

E. Camera

For the camera, we will be using the Logitech HD C270 USB webcam. It has a 1280 x 720 resolution, which has been demonstrated to be more than enough to read barcodes and QR-codes reliably. The camera is activated when the user requests to scan a product to cook. When prompted the hold the barcode to the camera. To increase efficiency a laser diode is used to help aim the barcode. The camera is connected directly with the Beaglebone black through a USB 2.0 connection.

IV. Software

The Memrowave cannot be considered a ‘smart’ application without well developed software. This section discusses the software design details and considerations for the Memrowave.

A. The Memrowave’s Operating System

Android was chosen to be the best choice for the Memrowave’s operating system. Android support for the BeagleBone Black is provided by the open-source Rowboat project, which enables Android Jelly Bean v4.2.2 to run on TI Sitara ARM Cortex-A processors.

B. Drivers, Libraries, Dependencies

There are a number of drivers that are needed for Android to support the hardware of the Memrowave. A driver is needed for the LCD to work properly, but luckily, this driver is included in the Android distribution we used. However, the driver for the camera is not included, so it was obtained separately and installed on the system, along with the driver for the Wi-Fi interface.

Android includes most of the libraries we used to create the. Android does not have a native library for reading barcodes or QR-codes using a camera. To handle this, we were required to use the open source ZBar multi-format barcode image processing library. This software library is licensed under the Apache License 2.0, which gives us permission to use the software in our project.

C. Communication

The microwave hardware is controlled by a circuit separate from the BeagleBone Black, with its own microcontroller. The microwave control circuit takes care of the timing, power level, magnetron control, fan control, turntable control, beeper, and the light control. The BeagleBone Black communicates with the microwave control circuit via the I^2C communication protocol, with the BeagleBone Black configured as the master and the microwave control circuit configured as the slave. The I2C master must write the I2C command to the slave device, and then read from the slave device to retrieve the result. Simultaneously, on the slave device, after receiving a command that expects some return value, the slave must write to the master.

D. Barcode scanning library

The barcode scanning will be done using ZBar barcode library to scan barcodes and QR-codes using the Memrowave’s camera. The library can decode barcodes from image data. On each frame from the Memrowave’s camera, the image data is converted from the default YUYV pixel format of the Logitech c270 to the Y800 pixel format, a greyscale format supported by ZBar. Each frame is run through ZBar’s decoding algorithm to check if a valid barcode is in frame.

E. User Interface

The primary means of interfacing with the Memrowave is a touch screen interface and not physical buttons like older microwaves. Touch screen interfaces should be reasonably familiar with the majority of users today due to the proliferation of handheld smart devices such as cellular phone and tablets. With this in mind, conventions that are used by modern communication devices, such as the swiping motion to advance through a list, will be used for the Memrowave interface where feasible. Interface design should be kept simple while also trying to avoid driving too deeply into menus or requiring too many steps to accomplish a simple task.

The first screen that a user will see is the Memrowave's main menu or home screen. This screen should remain simple with a clear focus on what makes the Memrowave unique, that is the ability to scan and automatically prepare food products.

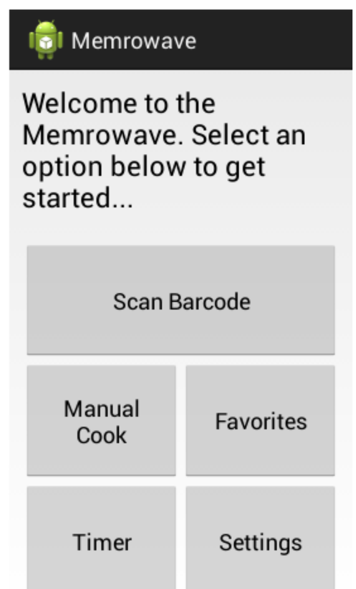

Fig 5. The first screen that a user will see is the Memrowave's main menu or home screen. This screen has four options; Scan, Favorites, Manual cook, and Settings.

Despite the automatic nature of the Memrowave, it should have a capable manual control option. Figure 6 shows the manual cook screen and the cooking screen.

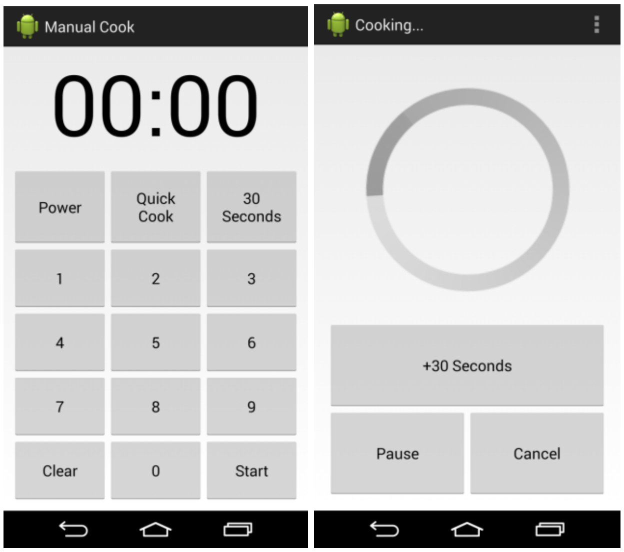

Fig 6. Allows the user to manually set cook time and power level like a standard microwave. During cooking the screen displays a circle to denote time.

Lastly, the Memrowave displays the product information when scanned. This is shown in figure 7.

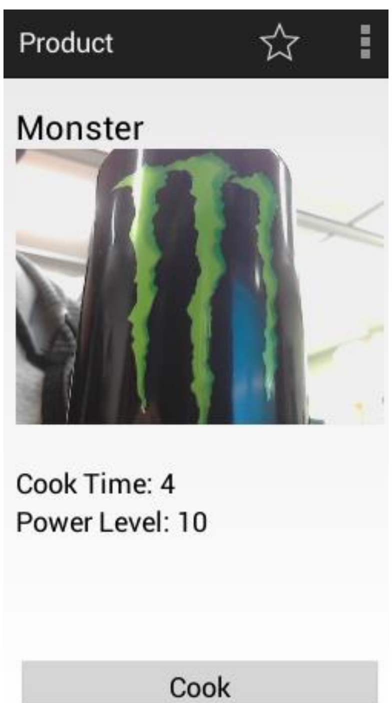

Fig 7. The screen display for a given product. Provides information such as Cooking time and Power Levels.

V. PCB Design

The Memrowave's control PCB was designed using EagleCAD. The design uses a two layer board with two main IC's. The PCB was designed to fit a 20 pin MSP430 and a 16 pin MAX6958. All components used on the PCB were chosen as though-hole mounts. This allowed for ease of soldering and replacement. The PCB mimics the layout of the Beaglebone black. This similar size allowed for easy alignment of the 46-pin headers. The schematic design was obtained from sparkfun using their open source libraries. Figure 8 depicts the final PCB.

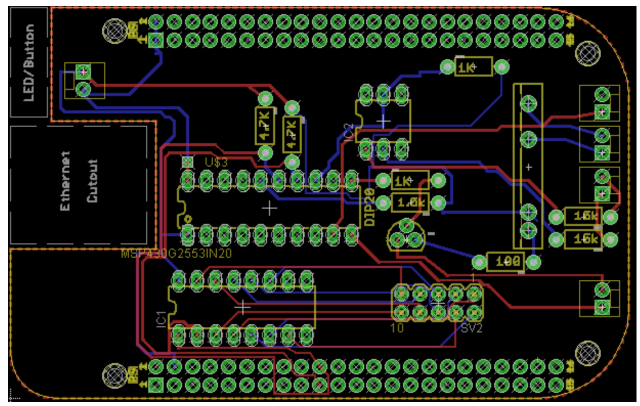

Fig 8. The final PCB layout design. The board consist of 4 molex connectors to serve as intputs for; 3.3V line, I/O for the magnetron relay, I/O for light relay, and I/O for the door switch.

As shown in the PCB only one relay is mounted to the board, which is a smaller 8A relay. The magnetron relay was removed from the design to isolate the high current of the magnetron's transformer.

The PCB was sent out for fabrication approximately 8 weeks into development, the group decided to use OshPark as the vendor. OshPark was chosen because for each ordered board 3 will be made. Having three boards was seen as a major advantage, especially due to inexperience with PCB fabrication.

The fabricated PCB design contained one major error. The MAX6958 LED driver chip's ground was isolated from common ground. The problem was resolved by soldering a jumper wire to the MAX's ground pin and the input gound from the power supply. Figure 9 depicts the final PCB from OshPark.

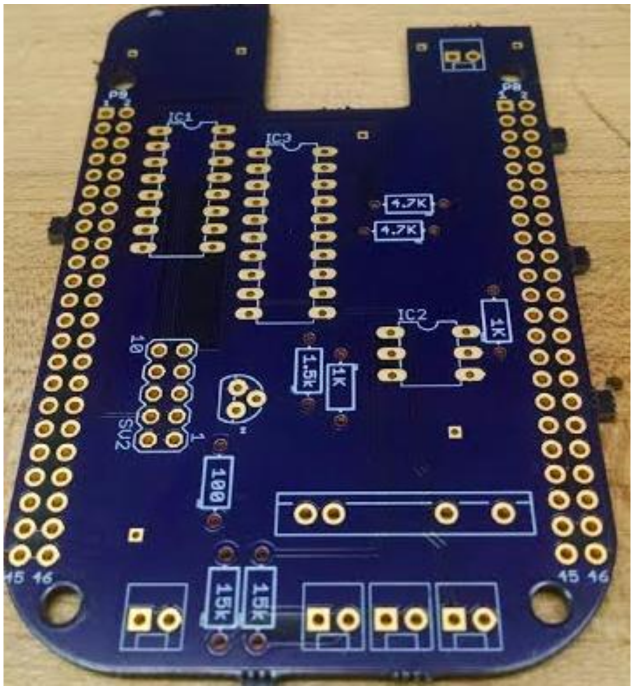

Fig 9. The final PCB layout design recived from OshPark. The board consist of 4 molex connectors to serve as intputs for; 3.3V line, I/O for the magnetron relay, I/O for light relay, and I/O for the door switch.

VI. Finance and Budget

The Memrowave's development was only possible with a strong financial backing. There were multiple options to obtain financing for our project. The first option was to seek finical backing from companies that would like to sponsor our design. Our largest sponsor options were Duke Energy and Boeing. These companies requested documentation contain information on project summary, applications, and financing requests. Using these sponsors however limited our group to only receiving reimbursements for our money invested. The Memrowave design team decided on financing the project out of pocket. The product's estimated cost was considered to be low and the design didn't fit the needs of Boeing.

The Memrowave's budget consisted; LCD screen, Wi-Fi, printed circuit boards, and more along these lines. Our design was heavily influenced by the microwave used. For instance if our design required us to remove parts from the microwave we would end up spending more on replacing the parts. However if our design allowed us to reuse the original parts such as motors and fans. This will give our budget a worst case scenario.

The budget includes the medium price listings for cameras, Wi-Fi Chips, development boards, Touch screen panels, Microcontrollers, resistors, transistors, and so on. To save time parts and the PCB were ordered from vendors that supply more than one board upon purchase. The Best PCB manufacturer for this was OCH Park because they supply 3 copies of the printed circuit board with the purchase.

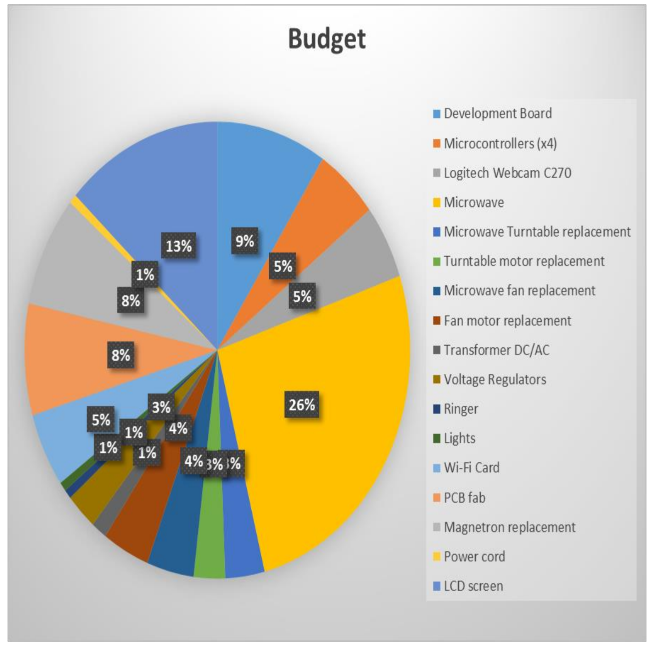

Fig 10. The Memrowave's final budget, the most expensive item was projected to be a new microwave.

As the chart shows the largest portion of the budget is devoted to purchasing a microwave. However we were able to use a pre-purchased microwave (which saved 26% of our budget) our largest contributor is now the LCD screen (which is 13% of our budget). One hundred percent of our own financial resources were used when creating the Memrowave. This allowed us to work at own pace without constraints from a sponsor. It also allowed the design team to keep the Memrowave upon completion. The total cost was under budget because our team was able to acquire a pre-purchased microwave that has a working magnetron, turntable, fan, and other standard microwave accessories.

VII. ENGINNEERS

Jack A. "Andy" Gulick is a 51 year old Computer Engineering student at the University of Central Florida in Orlando, Florida. In his previous 21+ year career as an electronics technician, Andy worked with airborne and ground systems for several missile and spacelift programs. His specific interest in the electronics and computer science technologies utilized in spacelift command and control systems led him to pursue his current degree. He hopes to return to working in the spacelift industry after he graduates.

Darren Armstrong, a senior student of the Electrical Engineering department of the University of Central Florida. He is currently a part of the CAAT research group at UCF. After college he would like peruse graduate studies in power engineering at USF.

Joseph Serritella, a senior student of the Electrical Engineering department of the University of Central Florida. He is currently a part of the CAAT research group at UCF and participating Honors in the Major, developing a room temperature methane sensor.

Winston Todd is a senior Computer Engineering student at the University of Central Florida. His areas of interest include application development for web and mobile platforms, digital electronics, and embedded Linux development.

VIII. CONCULSION

Creating the Memrowave has been a rewarding challenge, with many obstacles to overcome. After a considerable amount of patience and hard work the project started to become the Memrowave. The key asset to the development was excellent software components developed by each computer engineer. By Integrating barcode scanning, local and internet databases, and Wi-Fi connectivity the Memrowave was able to bridge the gap with microwaves and smart devices. The approach demonstrated the success of using barcode scanning on product UPC codes to quickly determine cook times and product information. In the end the Memrowave has proven an effective appliance bringing innovation to commonly used technology.

ACKNOWLEDGEMENT

The authors wish to acknowledge the assistance and support to all of the professors that took their times to help teach them engineering. Specifically the team would like to thank Dr. Richie for being a major help when learning the design processes.

The Team would also like to thank the committee members for taking their time to provide a review on the Memrowave.

Professors like Dr. Malocha, Dr. Chan, and Dr. Weeks played a major role in teaching the Electrical Engineering discipline to the groups EE members. Their wealth of knowledge gave insight technical projects.

The design team would like to acknowledge every student in the college of Electrical Engineering and Computer Science for their hard work and determination thought the years.

Each member of the Memrowave would like to thank their families and friends for their continued support thought the years at UCF.

All the help and support made the Memrowave possible.

REFERENCES

[1] "“Twilio - APIs for Text Messaging, VoIP & Voice in the Cloud.” Internet https://www.twilio.com/.

[2] "Beagleboard:BeagleBoneBlack Wiki." Internet: http://elinux.org/Beagleboard:BeagleBoneBlack .

[3] “"BeagleBoard."Internet: http://en.wikipedia.org/wiki/BeagleBoard.

[4] “"Liquid Crystal Display." Internet: http://en.wikipedia.org/wiki/Liquid-crystal_display.

[5]“Switching circuits” Internet: http://www.electronics-tutorials.ws/transistor/tran_7.html