

**ECE445**

**Spring 2013**

**Design Review**

**Easy Cooking Programmable  
Electric Stove**

Team 18:

Chng, Xiong Kai Benjamin

ArdyWinoto

Cheng Han, Lee

Professor:

Paul Scott Carney

TA:

Dennis Yuan

## Contents

I.	<b>Introduction</b> .....	2
II.	<b>Design</b> .....	4
	2.1 Block Diagrams.....	4
	2.2 Block Descriptions.....	5
	2.3 Component Schematics, Simulations and Calculations.....	10
III.	<b>Requirements and Verification</b> .....	23
	3.1 Requirements and Verification.....	23
	3.2 Contingency Plan.....	27
	3.3 Tolerance and Analysis.....	28
IV.	<b>Ethical Issues</b> .....	29
V.	<b>Safety</b> .....	31
VI.	<b>Cost and Schedule</b> .....	33
	4.1 Cost Analysis.....	33
	4.2 Schedule.....	34
VII.	<b>Citations</b> .....	36

# **I. Introduction**

## **a. Motivation**

Cooking is traditionally a manual process that often requires the cook to follow a set of instructions. However, this process is highly prone to fluctuations in execution. In the case of an inexperienced cook, this process is also daunting and prone to errors. As such, we decided to design a system that partially automates the cooking process to address these issues.

## **b. Benefits and Features**

The goal of this project is to create a simple and safe interface that guides a person through the process of cooking as well as make the cooking process more convenient. Thus, the stove will be partially automated with instructions that are displayed to prompt and facilitate the cooking process.

Furthermore, we intend to assimilate smart-phones into the interface of the stove so as to take advantage of the data-sharing capabilities inherent in such platforms. As such, our design will allow a user to launch an Android application, choose a recipe from a pre-existing database of recipes, transfer it via Bluetooth to our cooking device. The user will then be able to start the cooking process with most of the required heat settings automated. As mentioned earlier, the user will also be prompted with the cooking instructions on a LCD.

### **Benefits**

- Provides more convenience in the cooking process as the user does not have to change the heat level manually.
- Simple and intuitive user interface
- Guides users through the cooking process via instruction prompts on LCD
- Potentially increase the success rate of cooking a dish for inexperienced cooks.

## **Features**

- Single burner electric stove, AC input.
- Power management unit which connects to wall outlet, controls power input to the stove, and powers controller and peripherals.
- Temperature IR sensors.
- Proximity IR sensors.
- LCD for cooking instructions/status.
- Microcontroller to process the state of the system and provide necessary outputs.
- Buttons to control the state of the system/advance to next cooking stage.
- Bluetooth connectivity to extract electronic recipe from an Android device that has the app we created, 'SmartStove'

## II. Design

### 2.1 Block Diagrams

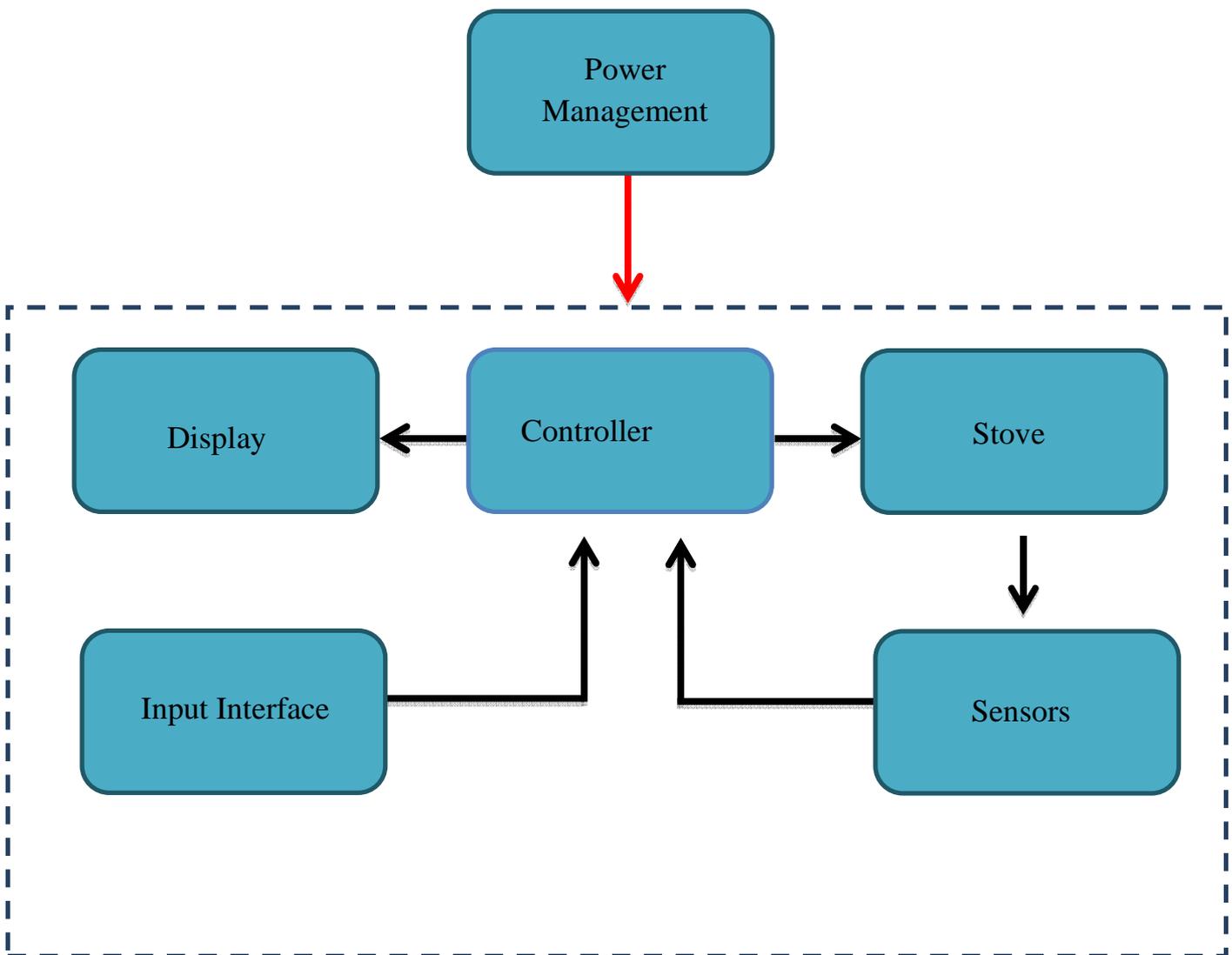


Figure 1: Block Diagram of the Design

## 2.2 Block Descriptions

### - Overall Schematic

For our system, the Power Management module is isolated from the rest of the system as illustrate by the dotted box. Every component in the dotted box will have power supplied from the Power Management module. All power flow is represented by the sole red arrow.

Looking at the controller model, the arrow going to the display represents the recipe steps that will be sent over to the Display module to be displayed on the LCD. The arrow going from the controller module to the stove module represents the PWM signal. The arrow coming on from the Input Interface Module represents the recipe data that will be transmitted to the controller via USB. The controller will also receive sensor signals from the Sensors module as shown by the arrow going from the Sensors module to the Controller module. Lastly, the arrow going from the Stove module to the sensors module represents analog inputs from measuring temperature and location of stove.

### - Controller Module

The controller module is responsible for receiving the recipe from the input module and processing the information received. In addition, it should respond to buttons presses from the user. It will also receive signals from the proximity sensors and the temperature sensor. The controller will process all these information and output a PWM signal to control the stove.

#### Master State Machine

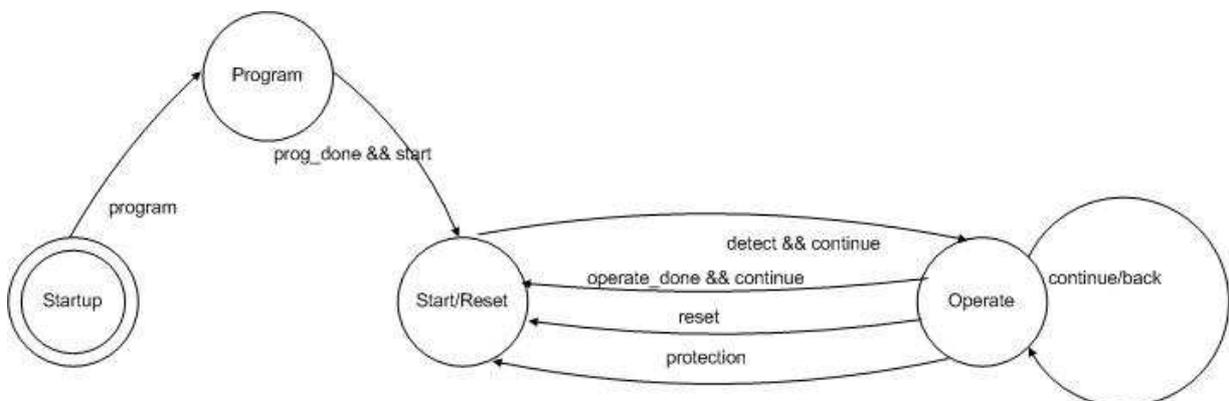


Figure 2: State Diagram of Controller Module

Control Signal Descriptions:

No.	Control Signal	Description
1.	program	“program” button is pressed
2.	start	“start” button is pressed
3.	reset	“reset” button is pressed
4.	continue	“continue” button is pressed
5.	back	“back” button is pressed
6.	prog_done	Flag set when the Arduino programming is finished
7.	detect	Flag set when a cooking vessel is detected in the last 10 seconds
8.	operate_done	Flag set when the recipe is finished
9.	protection	Flag set when stove exceeds maximum temperature (200 degrees C)

**Table 1: List of Control Signals**

State Descriptions:

1. Startup
  - Initial state upon turning on device.
  - Controller, LCD active.
  - LCD prompts user to program controller via Bluetooth.
2. Program
  - Receive new recipe from Bluetooth module and use it to program the controller.
  - Controller, LCD, Bluetooth powered active.
  - LCD displays programming progress, prompts user to send data/continue.
  - Bluetooth module sets flag when programming is done. Until flag is set, button presses are not registered.
3. Start/Reset
  - All systems ready for cooking process.
  - Controller, LCD, sensors active.
  - LCD displays recipe title, status, and sensor output.
  - Only possible to transition to operate if cooking vessel is detected.
4. Operate
  - State used for cooking process.
  - Controller, LCD, sensors, stove active (heat controlled by PWM).
  - LCD displays cooking instructions, cooking progress, and food temperature.
  - Continue/back goes to identical “Operate” state, with same behavior but different requirements, with as many “Operate” states as a given recipe requires.

### Operate State Specifications:

1. **Maximum/Desired Temperature**  
The appropriate/maximum temperature for a single cooking step. A zero maximum/desired temperature means that no temperature is set, with the maximum operating temperature as the limit (200 degrees C)
2. **Desired Heat Level**  
Heat level required for a single cooking step, controlled by setting the duty cycle of a PWM output which goes into an electromechanical relay that controls the stove.
3. **Prompt Message**  
Message displayed on LCD.
4. **Advance Condition**  
Required condition to advance to the next state. The conditions are based on time, temperature, or continue/back button presses.

### - **Power Management Module**

The power management module is responsible for converting the output voltage from the wall outlet ( $120V_{\text{rms}}$ ) into the appropriate voltages for the various modules. To convert the voltage into the appropriate ratings, we will implement an AC/DC converter by first stepping down the voltage output from the wall outlet to  $20V_{\text{rms}}$ . We will then rectify and filter this voltage to obtain a DC profile of about  $28V_{\text{DC}}$ . Since most of the other modules are powered by the FPGA controller or Arduino boards, we then implement a Buck Converter to step down the voltage from  $28V_{\text{DC}}$  to  $7V_{\text{DC}}$  to power these boards. The output from these Buck Converters is then connected to the other modules as a power source.

### - **Stove Module**

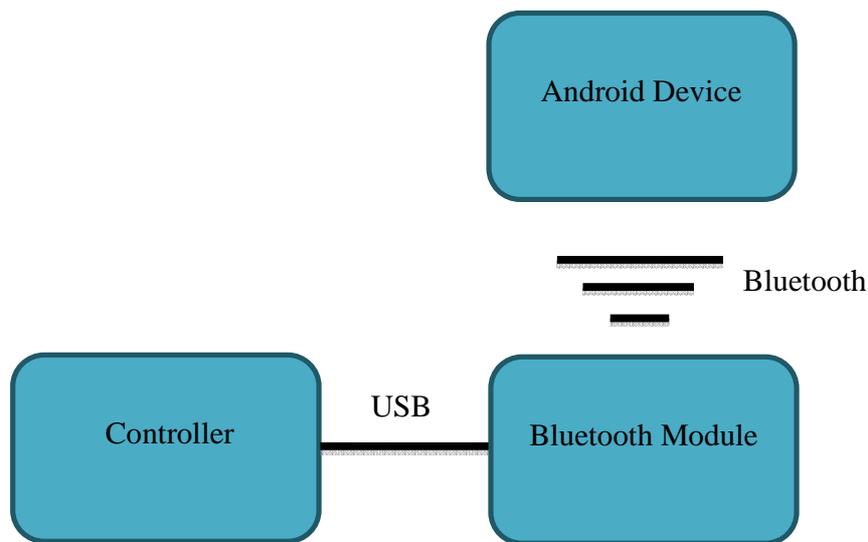
The stove module consists primarily of the electric burner which is the medium for heat to be transferred into the cooking vessel. This module will directly draw power from the wall outlet since the burner requires 1000W to operate, bypassing the AC/DC converter in the power management module. Furthermore, the stove module will be fitted with an electromechanical relay that receives the PWM signal from the controller module. This serves as a PWM mechanism to control the amount of current within the stove and hence, regulate the temperature of the stove and the cooking vessel.

- **Display Module**

LCD receives input from the controller module and displays instructions, status and food temperature. The food temperature displayed on the LCD should be close to real-time as appropriate measures are to be taken if the correct food temperature is not attained.

- **Input Module**

The input interface module consists of buttons and the Bluetooth Shield which is connected to the Arduino, which is then connected to the controller module via USB. The Android device will send data via Bluetooth to the Bluetooth module which consists of the Arduino and a Bluetooth shield. Signals from the Bluetooth module as well as button presses will be sent to the controller module via USB to be processed.



**Figure 3: Block Diagram of Input Module**

- **Sensor Module**

The sensor module controls the proximity and temperature sensors, conditions the signals from the sensors, and sends them to the controller module. The following sensors are used:

- Two IR proximity sensors sense the presence of a cooking vessel on the stove and communicate with the Arduino via two ADC ports.

- One IR temperature sensor monitors the temperature of the heating element on the stove and communicates with the Arduino using a two-wire SMBus protocol.
- One IR temperature sensor monitors the temperature of the food in the cooking vessel and communicates with the Arduino using a two-wire SMBus protocol.

## 2.3 User Interface

The user interface will consist of the display module and the input module. The monitor will display cooking instructions and useful information in the form of recipe title and details, time elapsed, and burner/food temperatures.

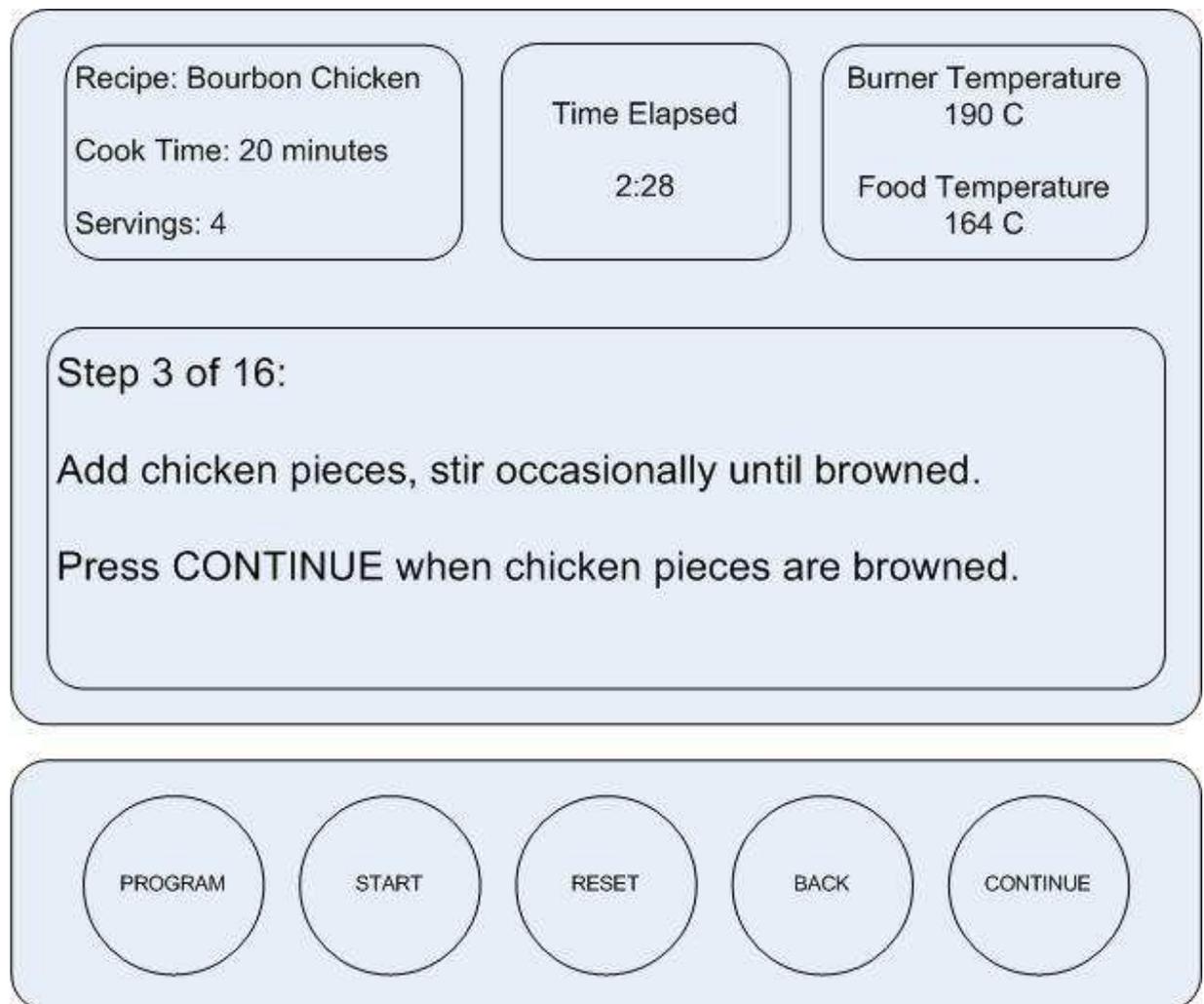


Figure 4: User Interface

## 2.4 Component Schematics, Simulations and Calculations

### - Power Management Module

The main purpose of the power management module is to convert the AC source from the wall outlet into the appropriate DC source for the various components. As mentioned earlier, we will first step down the voltage from the wall outlet to  $20V_{rms}$ . We will then rectify and filter this voltage into a  $28V_{DC}$  source. Finally, the  $28V_{DC}$  source will be stepped down to meet the voltage specifications of the other components.

In order to evaluate the appropriate DC voltage we need to convert to, we compiled a table of estimated voltage and current ratings for the various components in our system.

Component	Voltage and Current Rating
Temperature Sensors	5V @ 1.5mA
Proximity Sensors	5V @ 30mA
Bluetooth Shield	2.8 - 3.5V @ 3 - 100 mA
LCD	2 - 3.6V @ 18 - 28 mA
Arduino	7 - 12V @ 40 - 50 mA

**Table 2: Voltage and Current Rating of System Components**

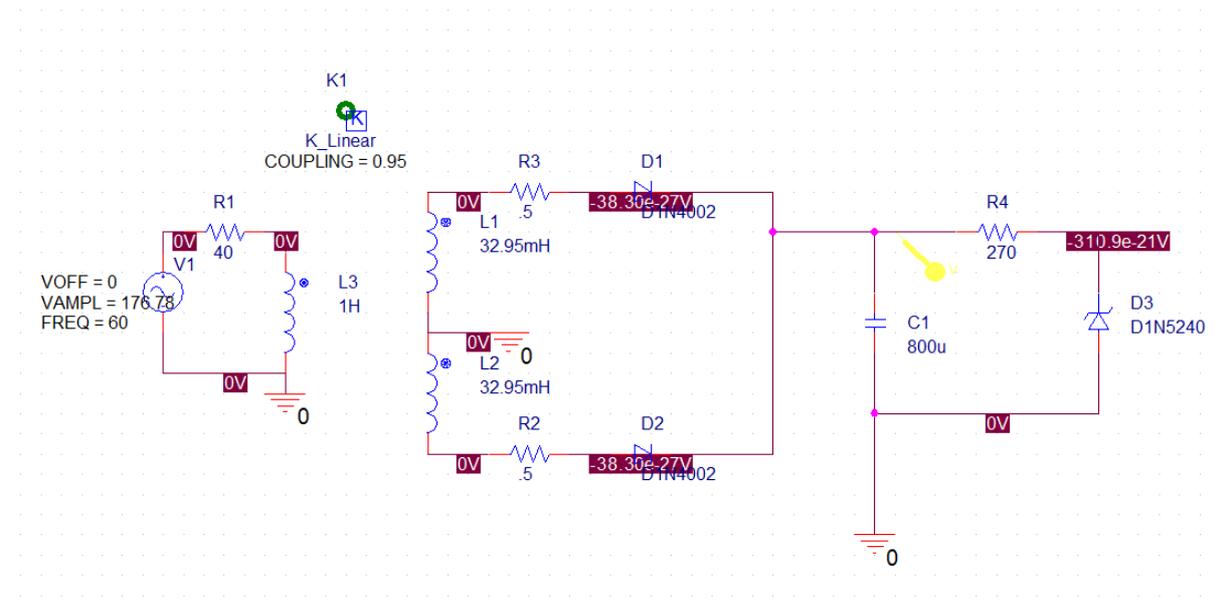
As can be seen from the table above, most of the components have a voltage rating that does not exceed 12V. Furthermore, we note that most of the components, such as the sensors and bluetooth shield, will be connected to the Arduino board. Hence, most of the components can be directly powered by the controller boards. We thus decided to build a Buck Converter that converts the  $28V_{DC}$  source into a  $7V_{DC}$ . We chose to step down to a value of 7V instead of the minimum of 5V because the controllers will further regulate the input voltage to a value lesser than 5V.

In view the ratings analysis performed above, we decided to implement one AC/DC converter and 3 Buck Converters (for the 3 Arduino boards) in the power management module. We will also be using the following components to reduce the complexity of our design:

- 241-4-20 Power Transformer
- LM2576 Simple Switcher

## AC/DC Converter

The AC/DC converter will convert the voltage of  $120V_{\text{rms}}$  from the wall output to a voltage of  $28V_{\text{DC}}$ . We decided to follow the center-tapped full wave transformer due to its simplicity of implementation. The schematic of the AC/DC converter is as follows:



**Figure 5: AC/DC converter schematic**

Using the NI reference for basic calculations of a Buck Converter [2], we note that the minimum required capacitance for the filter is

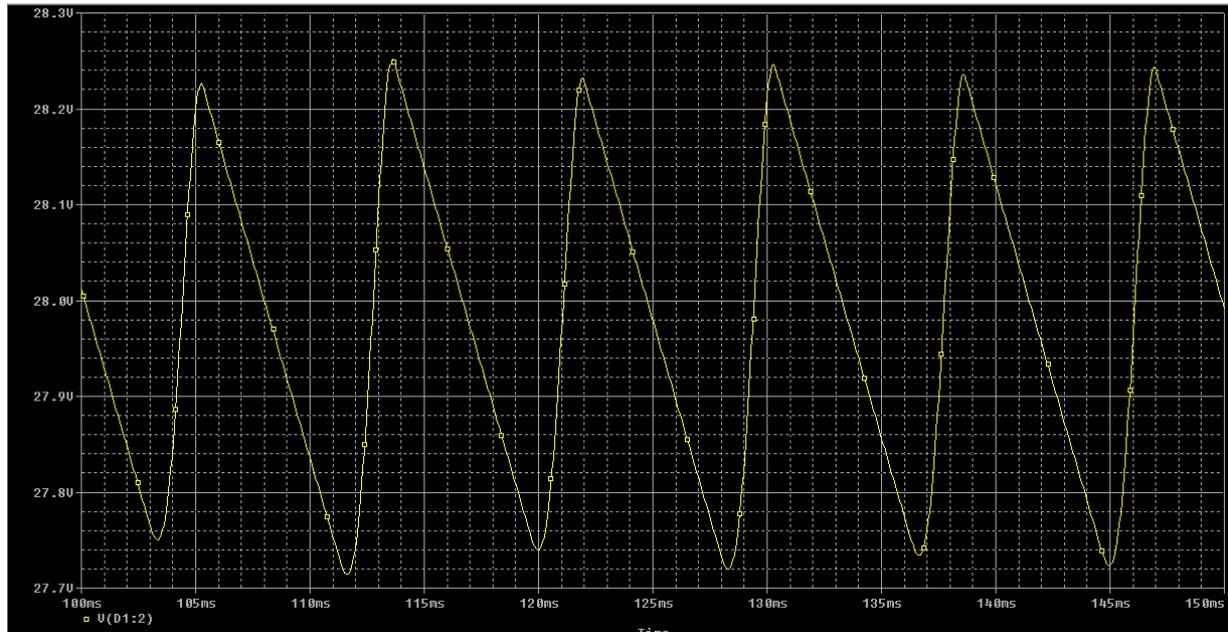
$$C_{min} = \frac{\Delta I_L}{8f_s \Delta V_{out}}$$

where  $\Delta I_L$  is the estimated inductor ripple,  $f_s$  is the switching frequency of the converter and  $\Delta V_{out}$  is the desired output voltage ripple. In our system, we specify a desired output voltage ripple that is 1% of the expected output voltage. We also note that the switching frequency of the converter is 50Hz from the datasheet. Although  $\Delta I_L$  is an unknown, we can approximate it as 0.3 times of the maximum output current from the NI reference [2]. As such, the minimum required capacitance is

$$C_{min} = \frac{0.3(300mA)}{8(50Hz)(28)(0.01)} = 795\mu F$$

From our calculations, we choose to use a capacitor that is rated with a capacitance of  $800\mu F$ . Furthermore, a Schottky diode with an average forward current rating of 300mA is recommended and this can be easily obtained.

Given the specifications of the transformer, capacitor and diode, we proceed to do a simulation of our circuit. The simulation results are as shown:



**Figure 6: Waveform of Regulated Voltage**

## **Buck Converter**

The Buck Converter will convert the input voltage of  $28V_{DC}$  to an output voltage of  $7V_{DC}$ . In order to implement the switching function of a standard Buck Converter, we will be using a LM2576 Series Simple Switcher 3A Step-Down Voltage Regulator. The schematic of the AC/DC converter is as follows:

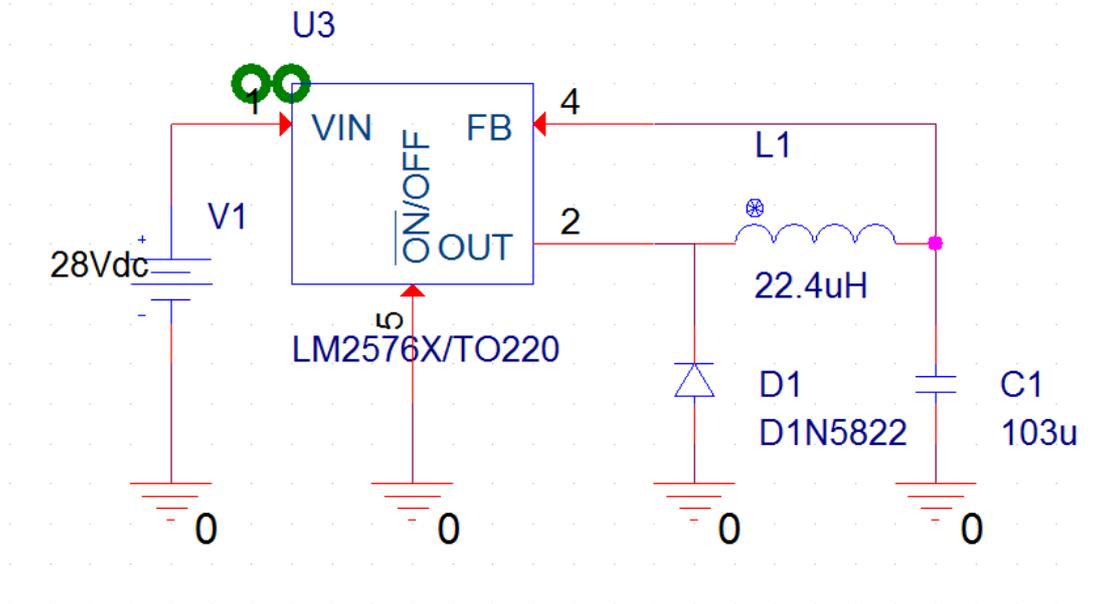


Figure 7: Buck Converter Schematic

Using the NI reference for basic calculations of a Buck Converter [1], we note that the minimum required capacitance for the converter is

$$C_{min} = \frac{\Delta I_L}{8f_s \Delta V_{out}}$$

where  $\Delta I_L$  is the estimated inductor ripple,  $f_s$  is the switching frequency of the converter and  $\Delta V_{out}$  is the desired output voltage ripple. As with the AC/DC converter, we specify a desired output voltage ripple that is 1% of the expected output voltage. We also approximate  $\Delta I_L$  as 0.3 times of the maximum output current of 3A for the LM2576 Simple Switcher. Furthermore, the datasheet specified the use of 52kHz in calculating the specifications of the Buck Converter. As such, the minimum required capacitance is

$$C_{min} = \frac{0.3(3A)}{8(52kHz)(7)(0.01)} = 30.9\mu F$$

From our calculations, we choose to use a capacitor that is rated with a capacitance of  $33\mu F$ .

Next, we need to calculate the required inductance for the Buck Converter. From the NI reference, the required inductance for the converter is

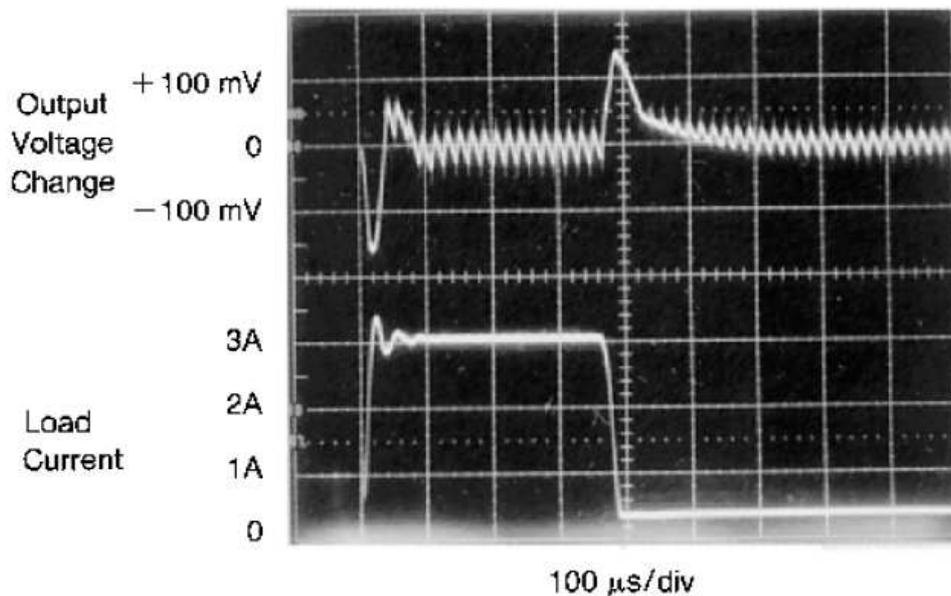
$$L = \frac{V_{in}(V_{in} - V_{out})}{\Delta I_L f_s V_{in}}$$

where  $\Delta I_L$  is the estimated inductor ripple,  $f_s$  is the switching frequency of the converter,  $V_{in}$  is the input voltage and  $V_{out}$  is the output voltage. Given an input voltage of 28V and an output voltage of 7V, the required inductance for the Buck Converter is

$$L = \frac{7(28 - 7)}{(0.3)(3)(52kHz)(28)} = 74.8\mu H$$

From our calculations, we choose to use a capacitor that is rated with an inductance of  $75\mu H$ . Furthermore, a Schottky diode with an average forward current rating of 2A is recommended and this can be easily obtained.

Although we were unable to do the simulations on PSPICE since the schematics of the LM2576 Simple Switcher is not available, we used the simulation results from the datasheet [1] as a reference in understanding and analyzing our design parameters for the Buck Converter. The following simulation is for a Buck Converter with a load current of 3A and an output voltage change of 100mV:



**Figure 8: Buck Converter Waveform**

## - Stove Module

For our stove module, we will be using an electric Single Coil Toastess THP-517 stove that consumes 1000W. Furthermore, the current supplied to the stove does not need to have a DC profile for proper operation. As such, we will bypass the AC/DC converter in the power management module and supply the stove with power from the wall outlet directly.

In order to implement the PWM feature on the stove, we will be using an electromechanical relay to control the duty cycle of the AC supply. The relay that we will be using is a single pole, double throw (SPDT) rated up to  $220V_{\text{rms}}$  @ 20A. When a current is passed through the coil, the switch will be activated, providing an effective means of controlling when the AC supply is channeled to the stove. Although the relay will produce an audible click when in operation, we intend to mute it using the metallic casing of the stove as a soundproof medium.

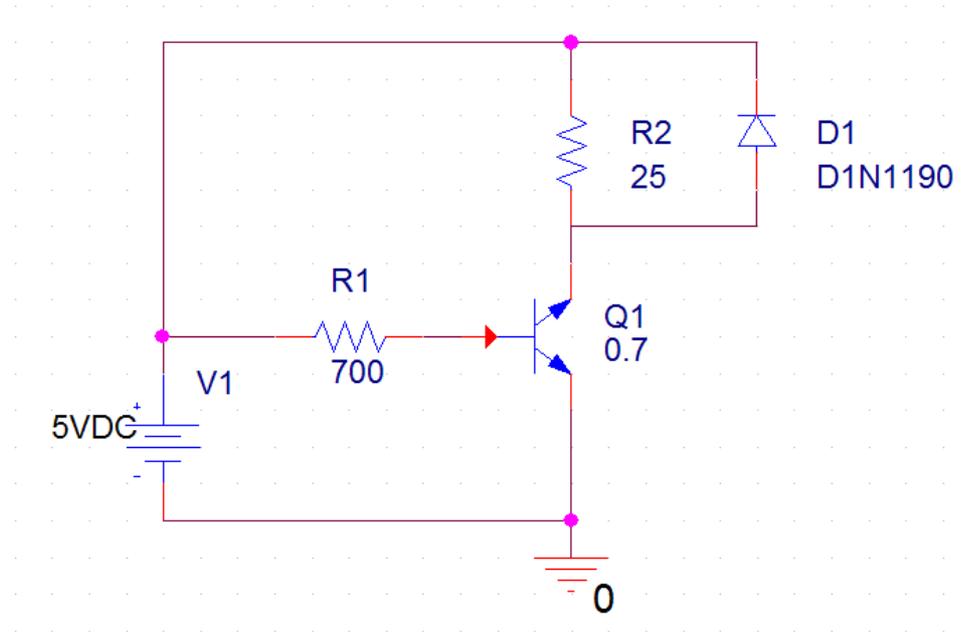
To operate the relay efficiently, we need to supply 1W to the coil. The following table shows the coil data rated at 1W for various voltage settings.

**Coil Data - 1 Watt**

Nominal Voltage	DC Resistance $\pm 10\%$ (Ohms)	Nominal Current (mA)
5	25	200
9	81	111
12	144	83
18	324	56
24	576	42
48	2,304	21
110	12,100	9

**Table 3: Rating of Coil at Various Nominal Voltage**

As we can see from the table above, the current required at 5V is 200mA. However, the output signal from the PWM pin of the Arduino board is rated 5V @ 40mA. As such, we will need to amplify the input current to 200mA. This can be achieved using a common emitter amplifier. The schematic of the common emitter amplifier is as follows:



**Figure 9: Amplifier Schematic**

As can be seen from the diagram, we model the relay as a resistor with a resistance of  $25\Omega$  as indicated in table 3. To calculate the resistance R1 from the schematic above, we note that the BJT is in saturation when a  $40\text{mA}$  signal is sent to the base. By KCL,

$$i_B = \frac{V_{in} - V_{BE}}{R1}$$

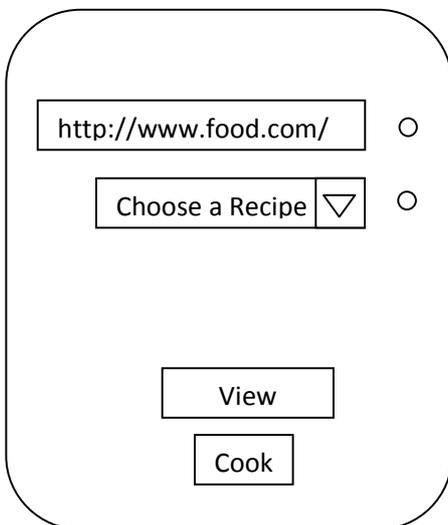
Thus, by rearranging the terms, the required resistance for our amplifier is

$$R1 = \frac{V_{in} - V_{BE}}{i_B} = \frac{5 - 0.7}{40\text{mA}} = 645\Omega$$

From our calculations, we choose to use a resistor that is rated with a resistance of  $700\Omega$ . Furthermore, a Schottky diode with an average forward current rating of  $300\text{mA}$  is recommended and this can be easily obtained.

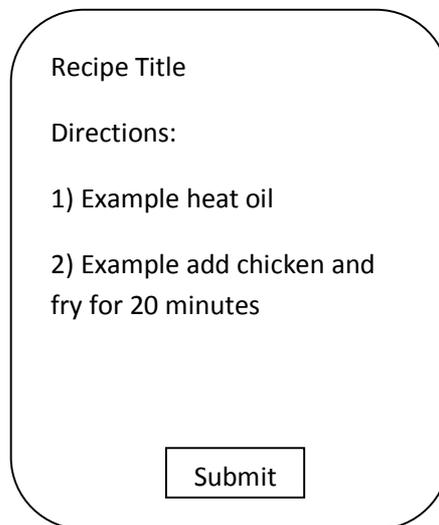
## - Input Module

We will design an app on the Android platform called “SmartStove”. The stove will consist of 6 preloaded recipes from <http://www.food.com> that the user can choose from. The app will consist of 3 different views (activities). When the user first opens the app, there will be a dropdown menu for the user to select the recipe he wants to cook and an URL field for him to view any recipe from <http://www.food.com/> with the 1<sup>st</sup> part of the URL already pre-filled. Depending on which radio button is selected, the View Recipe button will take the user to either view 1 or view 2 as shown in figures 11 and 12 below. With the URL radio button selected and a recipe URL inputted into the field, the user will be able to view the recipe and be able to help contribute a translation to our database for review by us after pressing submit. If the user chooses a recipe from the drop down menu, it will bring him to another page with the recipe steps as well as a Cook button which can be pressed if the user decides to cook this recipe.



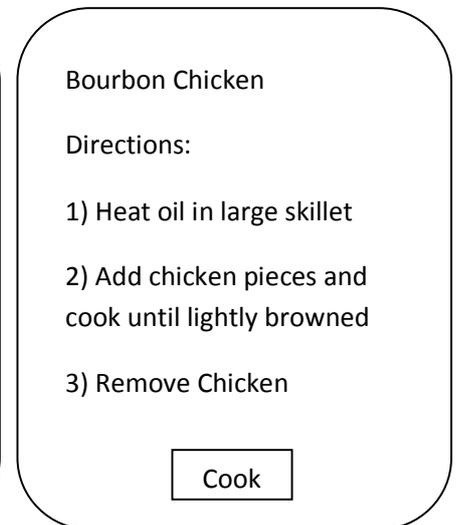
The Home Screen UI features a rounded rectangular container. At the top, there is a text input field containing "http://www.food.com/" with a radio button to its right. Below this is a dropdown menu with the text "Choose a Recipe" and a downward-pointing triangle, also with a radio button to its right. At the bottom of the container, there are two buttons: "View" and "Cook", stacked vertically.

Figure 10: Home Screen



The View 1 UI features a rounded rectangular container. At the top, it displays "Recipe Title". Below that is the text "Directions:" followed by a numbered list: "1) Example heat oil" and "2) Example add chicken and fry for 20 minutes". At the bottom of the container, there is a "Submit" button.

Figure 11: View 1



The View 2 UI features a rounded rectangular container. At the top, it displays "Bourbon Chicken". Below that is the text "Directions:" followed by a numbered list: "1) Heat oil in large skillet", "2) Add chicken pieces and cook until lightly browned", and "3) Remove Chicken". At the bottom of the container, there is a "Cook" button.

Figure 12: View 2

However, as the feature of having a user being able to submit a translated copy of a recipe is not an integral part of our project, with the most important parts lying in the

hardware design, we may or may not implement a UI that allows a user to help us translate a recipe and submit it to us for verification.

For the manual translation of the recipe to something that is easier for our code to understand, we will be following the operate state specifications closely with slight modifications.

We will break each step of the recipe into the following 3 parts:

- 1) Desired heat level
- 2) Ingredients to add/remove(LCD display)
- 3) Advance condition
  - a. Duration
  - b. Condition
  - c. User presses continue button(for subjective cooking steps such as ‘cook until lightly browned’ or ‘until well mixed and dissolved’)

Desired heat level will contain keywords such as heat\_high, heat\_medium, heat\_low and heat\_level\_oiletc to indicate the heat setting required. If it is the 1<sup>st</sup> step of the recipe, there will be a startup time for the desired heat level to be reached before the prompt message to add in the 1<sup>st</sup> ingredient comes up. For the following steps, it will include keywords above and also others such as heat\_prev(continue with the previous heat setting), heat\_reduce and heat\_increaseetc

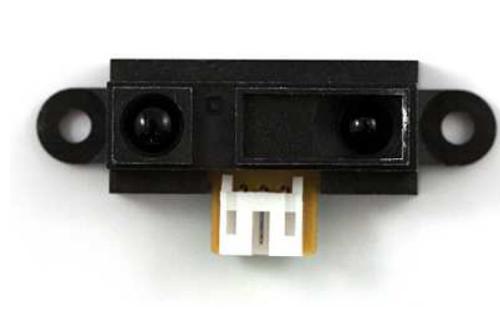
The prompt message step will simply be add or remove a certain ingredient and this will be displayed on the LCD.

For the advance condition, it could come in the form of duration, condition reached or when the user presses continue. Duration will be absolute time, such as ‘boil for 20 minutes’. Condition will be for steps such as ‘until hard boil’ or ‘until simmering’. For other cooking conditions that we deem too subjective, we will let the user decide when to continue by pressing the ‘Continue’ button.

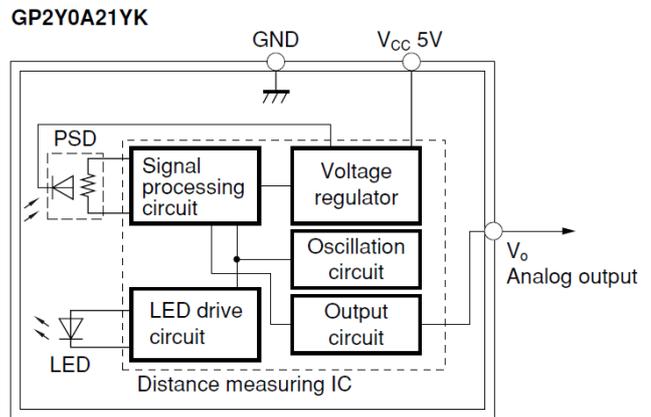
For all the steps, the LCD will display to the user what the system is currently doing.

- **Sensor Module**

**Sharp GP2Y0A21K IR Proximity Sensor**

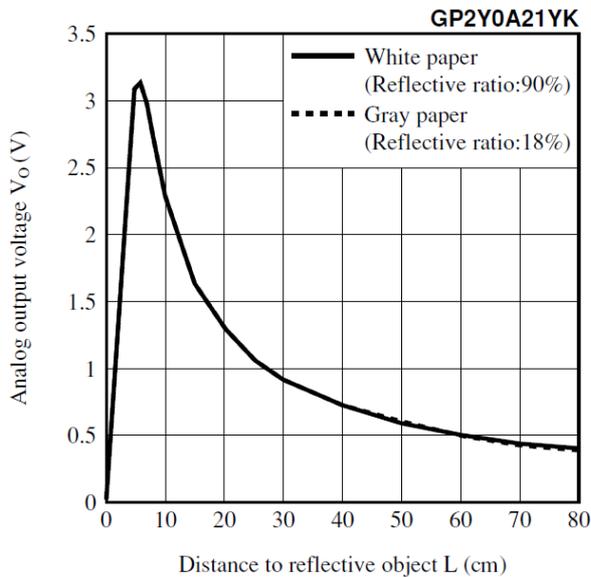


**Figure 13: Proximity Sensor**



**Figure 14:Block Diagram of Proximity Sensor**

The GP2Y0A21K consists of an IR emitter and detector and an on-package distance measuring IC. It detects the intensity of IR light reflected off a surface and calculates the distance from the sensor to the surface. The GP2Y0A21K provides an analog output voltage and will communicate with the controller via an ADC. The analog output voltage response with respect to surface distance is shown below:



**Figure 15: Analog voltage response of proximity sensor**

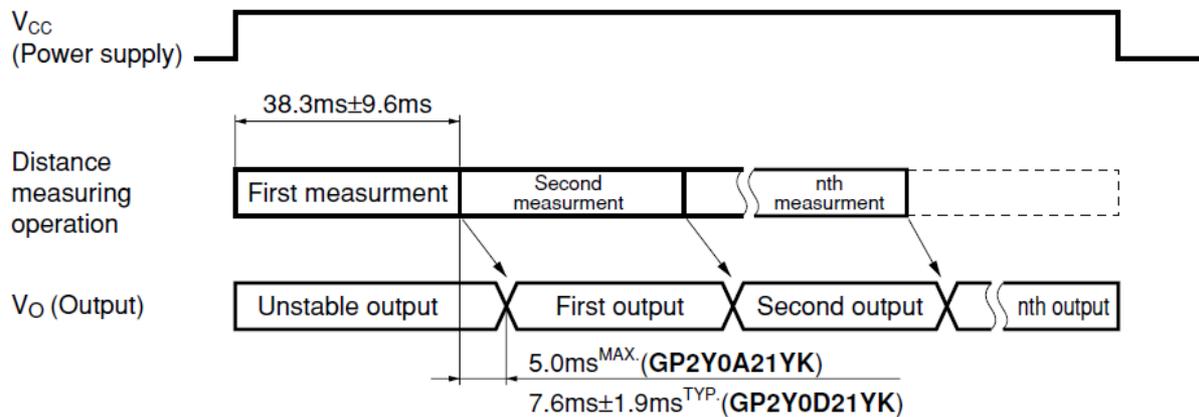
Pinouts:

1. Vcc ( 4.5-5.5V supply voltage)
2. GND
3. Vo (Analog output voltage)

GP2Y0A21K Parameters

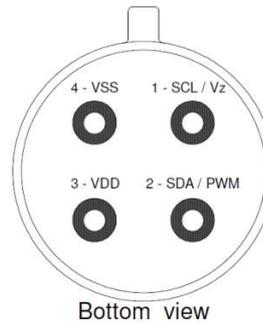
Parameter	Symbol	Conditions	Minimum	Typical	Maximum	Units
Distance Measuring Range	$\Delta L$		10	-	80	cm
Output Voltage	$V_o$	L=80 cm	0.25	0.4	0.55	V
	$V_{oh}$	High output voltage	$V_{cc}-0.3$	-	-	V
	$V_{ol}$	Low output voltage	-	-	0.6	V
Output Voltage Difference	$\Delta V$	Output change from L=80cm to L=10cm	1.65	1.9	2.15	V
Dissipation Current	$I_{cc}$	L=80cm	-	30	40	mA

**Table 3: Specifications of Proximity Sensor**



**Figure 16: GP2Y0A21K Timing Diagram**

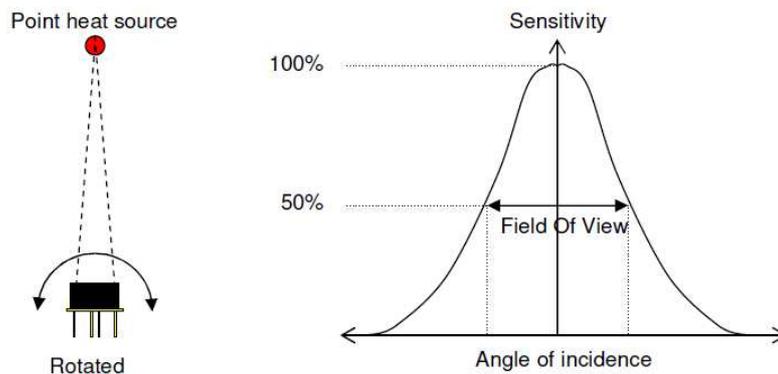
## Melexis MLX90614-ACA-000-TU IR Thermometer



**Figure 17: IR Thermometer**

**Figure 18: IR Thermometer Bottom View**

The Melexis MLX90614-ACA-000-TU is an IR thermometer for non-contact temperature measurements. It consists of an IR sensor, an IR thermopile detector IC, and a signal conditioning ASSP IC with an internal DSP and a high-resolution 17-bit ADC. It measures a weighted average of all temperatures in its field of view (FOV), with peak sensitivity located at an angle of incidence of 90 degrees.



**Figure 19: Field of View of IR Sensor**

The IR sensor consists of serial-connected thermocouples with cold junctions placed at thick chip substrate and hot junctions placed over a thin membrane. IR radiation incident on the sensor heats or cools the membrane and the thermopile detector produces an output voltage  $V_{ir}$  as follows:

$$V_{ir}(T_a, T_o) = A \times (T_o^4 - T_a^4)$$

To and Ta are object and sensor die temperatures (K) and A is the overall sensitivity of the sensor. The output voltage of the thermopile then serves as an input to the internal DSP, which calculates the corresponding ambient and object temperatures and produces digital outputs. The sensor will communicate with the controller through two wires using the SMBus communications protocol.

Pinouts:

1. SCL/Vz  
Serial clock input for 2 wire communications protocol.
2. SDA/PWM  
Digital input/output. In PWM mode, the measured object temperature is available at this pin as a pulse width modulated signal. In SMBus mode, the pin is configured as an open drain NMOS.
3. VDD (5V)
4. VSS (GND)

### III. Requirements and Verification

#### 3.1 Requirements and Verification

<b>Controller Module</b> Requirements	Verification
1) Controller module displays correct state transitions and PWM outputs with 6 recipes.	Simulate state machine using the 6 recipes and simulated button presses and sensor outputs as well as expedited time frames and monitor the states as well as the PWM signal.
2) Controller module reverts to Start/Reset state when the temperature exceeds 200 degrees C or when the cooking vessel is not on the stove for 10 seconds.	Simulate reset conditions and verify that the state machine goes to the Start/Reset state.
<b>Power Management Module</b> Requirements	Verification
241-4-20 Power Transformer should: <ul style="list-style-type: none"> <li>a. Convert <math>120V_{\text{rms}}</math> to <math>20V_{\text{rms}}</math>.</li> <li>b. Have a maximum operating current of 300mA</li> </ul>	Connect output signal from the transformer into a power probe <ul style="list-style-type: none"> <li>a. Voltage measured of output signal is <math>20V \pm 1\%</math></li> <li>b. Current measured of output signal is at maximum of <math>300\text{mA} \pm 5\%</math></li> </ul>
Rectifier and Filter should: <ul style="list-style-type: none"> <li>a. Provide a DC profile of 28V with a ripple voltage of <math>\pm 0.3V</math> from the <math>20V_{\text{rms}}</math> source.</li> </ul>	Connect output signal from the filter into an oscilloscope <ul style="list-style-type: none"> <li>a. Voltage measured of output signal is <math>28V \pm 0.3V</math></li> </ul>
Buck Converter should: <ul style="list-style-type: none"> <li>a. Convert <math>28V_{\text{DC}}</math> to <math>7V_{\text{DC}}</math> with a ripple voltage of <math>\pm 0.1V</math></li> <li>b. Have an output current of 3A.</li> </ul>	Connect output signal from Buck Converter into an oscilloscope <ul style="list-style-type: none"> <li>a. Voltage measured of output signal is <math>7V \pm 0.1V</math></li> <li>b. Current measured of output signal is <math>3A \pm 1\%</math></li> </ul>

<b><u>Stove</u></b> Requirements	Verification
Modified stove should boil water as fast as the original stove	We will measure the time taken for the original stove to boil 1L of water at various heat settings. When the modified stove is in place, we will measure the time taken for the modified stove to boil 1L of water. The time taken to boil water should be the same for both the original and modified stove with a tolerance of $\pm 5\%$
Electromechanical relay should switch effectively with 1W of power dissipated into it	Connect a 1W power source across the electromechanical relay and measure the resultant voltage to see whether the relay has been switched. This test will be performed 10 times and assert that the relay is working correctly if the relay has been switched 90% of the time.
Electromechanical relay should respond to the PWM signal with a delay tolerance of $\pm 10\%$ of the period of the PWM signal.	Connect both the output of the PWM signal and the relay into an oscilloscope. Measure the time difference between the two signals and verify that the delay is around $\pm 10\%$ of the period of the PWM signal.

<b><u>Display</u></b> Requirements	Verification
Display must show correct image when the associated signals are sent from the controller	Run a test program that displays 50 characters comprising of letters, numbers and special symbols on the LCD. The test program will run 10 times on the LCD. Observe the LCD and ensure that the set of characters are correctly displayed 90% of the time

<b><u>Input Interface</u></b> Requirements	Verification
There must be internet connection on the Android device	Open any web browser on the Android device and at the URL bar, type <code>www.google.com</code> . If the page opens, internet connection has been established. If not, proceed to settings->Wi-Fi and turn it to 'on' . Then select one of the available networks for connection.
The Android app, “Smart Stove”, must be able to display the recipe steps that the user has chosen to preview.	Open “SmartStove” then choose from the given list of recipes. Proceed to press the “View Recipe” button and check whether a new screen pops up displaying the steps of the recipe.
Android device must be able to send data over Bluetooth to the Bluetooth module and the Arduino must be able to receive the data accurately	<p>Power on the Arduino. Connect the Arduino to the laptop via USB. Open the Slave.pde code provided by the Sseed library. Upload the code then open the serial monitor.</p> <p>On the Android device, download the app “Bluetooth SPP” available on the Play Store. Open the app and look for the device with the name “SeedBTSlave”.</p> <p>We will attempt to transmit 10 ASCII characters from inside the “Bluetooth SPP” app to the Arduino. Check that the input characters from the Android device appear on the Arduino serial monitor on the laptop, vice versa.</p>

<b>Sensor Module</b> Requirements	Verification
<p>1) Proximity sensor provides an appropriate analog output according to the distance between the sensor and a given surface, with no identical outputs for two different distances above 10 cm.</p>	<p>Connect the output of the sensor to an oscilloscope. Observe the output when test surface is placed on 11 different positions perpendicular to the sensor from 10 to 15 cm, with .5 cm intervals. Verify that the output voltage at each point is within <math>\pm 10\%</math> of the values in the manufacturer-supplied voltage vs. distance plot.</p>
<p>2) Proximity sensor interfaces correctly with the Arduino and the sensor module generates correct hexadecimal values based on the analog output.</p>	<p>Monitor the readings on the Arduino and (hexadecimal) and an oscilloscope (analog) when test surface is placed on 11 different positions perpendicular to the sensor from 10 to 15 cm, with 0.5 cm intervals. Then, convert the hexadecimal values back to the corresponding scale. The converted values should match the values on the oscilloscope, with a tolerance of <math>\pm 10\%</math>.</p>
<p>3) Two proximity sensors on the left and right sides of the stove in conjunction can sense the presence of a cooking vessel centered on the stove at a typical distance of 12 cm, tolerant to a radial offset of 2 cm from the center of the stove.</p>	<p>Place sensors on opposite sides of cooking vessel with a distance of 12 cm and verify that the sensors can detect the cooking vessel. Then, move the cooking vessel 2 cm in the 0,45,90,135,180,225,270,315 degree directions with respect to the sensor on the right and verify that the cooking vessel is still detected.</p>
<p>4) Temperature sensor interfaces correctly with the Arduino and senses the temperature of an object at a distance of <math>12 \pm 2</math> cm with a tolerance of <math>\pm 5\%</math>.</p>	<p>Measure the temperature of the stove as it is turned on with no cooking vessel present using the sensor and a thermocouple and generate a plot of temperature vs. time using both sets of data after converting hexadecimal values on the Arduino to the correct scale. The results</p>

	should be within $\pm 5\%$ of each other. Repeat with temperature sensor moved 2 cm away and towards stove.
--	---

### 3.2 Contingency Plan

Module	Contingency Plan
Controller	If the Arduino UNO fails to perform adequately, we will use the Raspberry Pi instead.
Power Management	If we fail to design the AC/DC converter, we will supply the stove with power from the wall outlet and for the other components, we will supply power using batteries
Stove	Purchase another model of stove if the original stove fails
Display	If the LCD fails, we will use instead a LCD display that is connected directly to the Arduino. The model number is Devantech LCD03 4x20 I2C & Serial LCD (Blue).
Input Interface	<ol style="list-style-type: none"> <li>1) If the Bluetooth communication fails, we will program the recipe straight into the controller.</li> <li>2) If we fail to device an algorithm good enough to parse and interpret recipes, we will write our own recipes formatted in a standard fashion.</li> </ol>
Sensors	<ol style="list-style-type: none"> <li>1) If the temperature sensor fails, we will instead use a thermocouple to measure the temperature of the stove in order to enforce the maximum operating temperature protection.</li> <li>2) If the proximity sensor detection scheme fails, confine the cooking vessel to a narrower detectable range on the stove.</li> </ol>

### **3.3 Tolerance and Analysis**

The component that is most susceptible to failure is the PWM mechanism embedded within the stove module. The PWM mechanism comprises of the electromechanical relay that controls the duty cycle of the AC current supplied to the coil and the amplifier that amplifies the PWM signal from the controller module to the relay. The mechanism is most likely to fail because the action of the relay is highly dependent on the amount of current drawn from the amplifier. In addition, the relay may be prone to huge delays that arise from the movement of mechanical parts in the switch.

Furthermore, the PWM mechanism is integral to the operation of our device since we need it to indirectly control the temperature of the stove. As such, we pinpoint the PWM mechanism in the stove module as the focus of our tolerance analysis.

#### Testing Focus:

We will replace the PWM input signal with a function generator that produces a square wave with an amplitude of 5V and offset of 2.5V. We will then sweep through a duty cycle from 0 to 100% with 1% intervals and record the profile of the AC supply using an oscilloscope.

#### **IV. Ethical Issues**

The purpose of our project is to simplify the cooking process for the average cook and also attempt to provide a safer cooking environment with the automatic shut off function. We also inform the user that our device is not 100% safe and there are inherent dangers in using a heated element and there is a chance of mechanical and component failures. As such, we follow the 1<sup>st</sup> code of the IEEE Code of Ethics which states:

- 1. to accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment.*

We will make claims and estimates based on data we acquired throughout the testing process and not modify or tamper with the data in any way. This follows the 3<sup>rd</sup> code of the IEEE Code of Ethics which states:

- 3. to be honest and realistic in stating claims or estimates based on available data*

We will be using Arduino boards, Seeed Bluetooth shield, temperature and proximity sensors amongst various other equipment and components. This will improve our understanding of the technology involved as well as their applications and potential consequences. This goes in line with the 5<sup>th</sup> and 6<sup>th</sup> code of the IEEE Code of Ethics which states that:

- 5. to improve the understanding of technology; its appropriate application, and potential consequences*
- 6. to maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations*

We will also take into consideration the advice and constructive criticisms for our advisors, peers and other parties of our work and to our best of our ability, acknowledge and make necessary corrections to our work. We will also cite all

sources that we have referred to our taken relevant information from. This goes in the line with 7<sup>th</sup> code of the IEEE Code of Ethics which states:

*7. to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others*

ECE445 is designed to assist our peers and faculty. This goes in line with the last code of conduct which state:

*10. to assists colleagues and co-workers in their professional development and to support them in following this code of ethics*

## V. Safety

Our project involves the supply of high AC current as well as heating elements that can reach approximately 200°C. As such, we must take necessary safety precautions when building our device in order to ensure the safety of other people working in the lab as well as ourselves, and also to protect the lab equipment and the lab room. As such, we need to pay special attention to electrical and fire hazards.

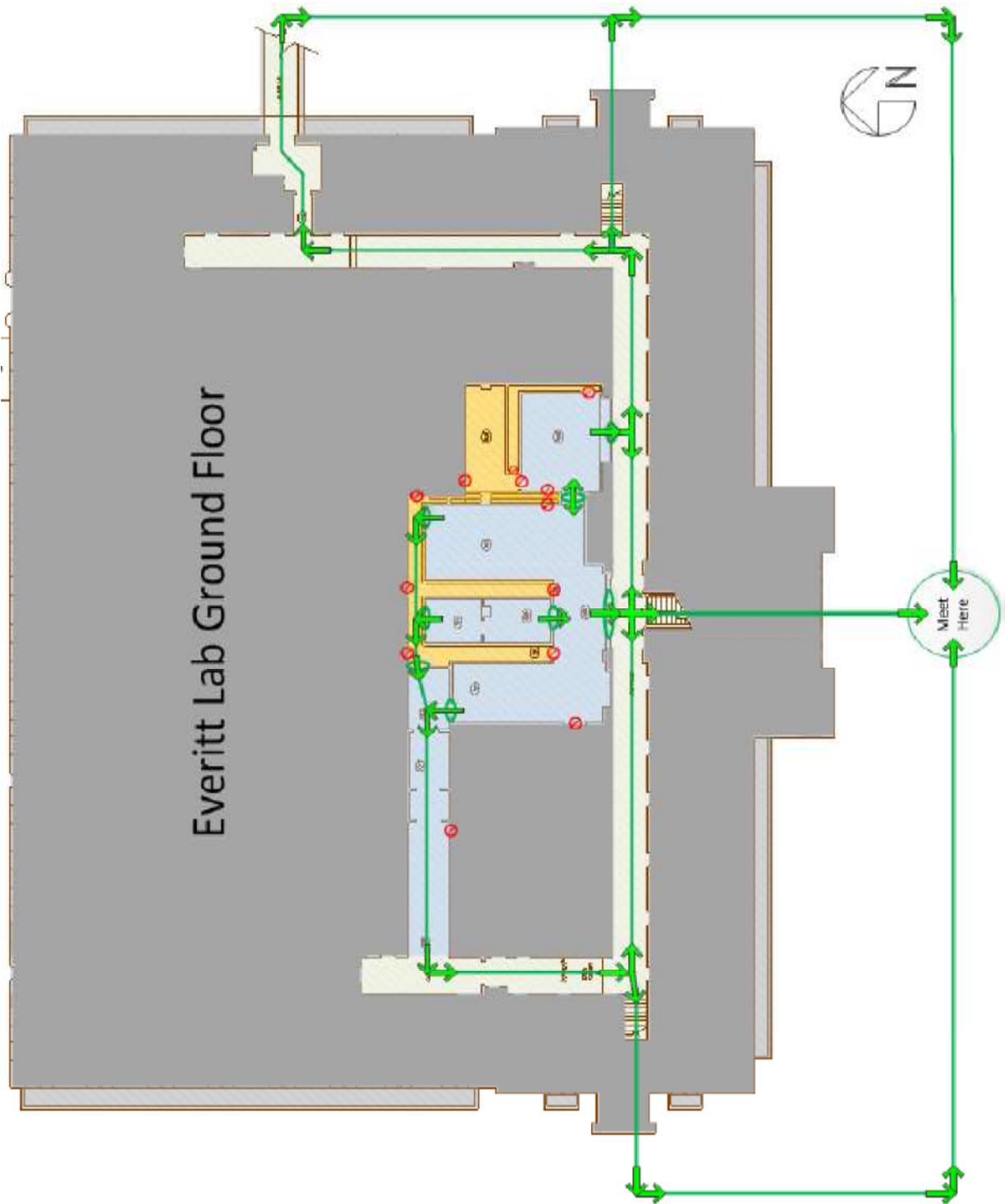
We will adhere to the following safety guidelines strictly:

- 1) When building and testing the device, there should always be at least 2 or more people present in case of an accident
- 2) Not leave any operating device unattended
- 3) Always wear safety glasses when operating/testing the stove as well as building/testing the power electronics
- 4) Always wear covered rubber-soled shoes as well as long pants when in the lab
- 5) Always ground yourself before doing any work
- 6) Be familiar with evacuation routes from the building
- 7) Be familiar with the list of phone numbers for assistance
- 8) Be familiar with the techniques for handling certain type of small fires
- 9) Do not consume any food or beverage of any kind in the laboratory
- 10) Be familiar with the Everitt evacuation route as shown on the next page

Evacuation Plan:

If the lab needs to be evacuated in case of an emergency, follow the evacuation route as shown below. When evacuating the lab:

- Do not stop to retrieve personal items
- Evaluate the safest route for exit:
  - o Check for smoke coming from under exit doors
  - o Feel exit doors before opening to ensure that fire does not exist on the other side
- Meet on the sidewalk south of the building



## VI. Cost and Schedule

### 4.1 Cost Analysis

Labor:

Name	Hourly Rate	Total Hours Invested	Total= Hourly Rate x 2.5 x Total Hours Invested
Chng, Xiong Kai Benjamin	\$40.00	150	\$15000.00
ArdyWinoto	\$40.00	150	\$15000.00
Cheng Han, Lee	\$40.00	150	\$15000.00
PCB	\$35.00	4	\$140.00
Mechanical Store	\$50.00	10	\$500.00
<b>Total</b>			<b>\$45640.00</b>

Parts:

Item	Part Number	Quantity	Unit Price	Total Cost
Single Coil Cooking Range	Toastess THP-517	1	\$30.00	\$30.00
Arduino	Arduino UNO A000066	3	\$27.20	\$81.60
IR Proximity Sensor	Sharp GP2Y0A21YK	2	\$15.00	\$30.00
IR Temperature Sensor	Melexis MLX90614ESF-ACA-000-TU	2	\$20.00	\$40.00
LCD	73-1335-ND	1	\$32.71	\$32.71
Round Push Button	MSPS103C	5	\$2.72	\$13.60
Seed Bluetooth Shield	SLD63030P	1	\$22.90	\$22.90
Center-tap Transformer	595-1050-ND	1	\$10.01	\$10.01
Electromechanical Relay	COM-10924	1	\$2.95	\$2.95
NPN BJT	NPN 2N3904	1	\$0.75	\$0.75
LM2576 Step-Down Voltage Regulator	576-1518-5-ND	3	\$3.06	\$9.18
75 $\mu$ H Inductor	M8715-ND	3	\$3.94	\$11.82
33 $\mu$ F Capacitor	445-8539-ND	3	\$0.70	\$2.10
800 $\mu$ F Capacitor	399-7544-ND	1	\$5.72	\$5.72
Schottky Barrier Diode	1N5822-TPMSCT-ND	6	\$0.58	\$3.48
700 $\Omega$ Resistor	ALSR5J-700-ND	1	\$1.55	\$1.55
			<b>TOTAL =</b>	<b>\$288.37</b>

GRAND TOTAL = LABOR + PARTS = \$45,928.37

## 4.2 Schedule

<b>Week</b>	<b>Benjamin</b>	<b>Ardy</b>	<b>Cheng Han</b>
<b>1/28</b> RFA due	<ul style="list-style-type: none"> <li>• Project Topic Discussion</li> <li>• Submit RFA</li> <li>• Discuss project/proposal details</li> </ul>	<ul style="list-style-type: none"> <li>• Project Topic Discussion</li> <li>• Discuss project/proposal details</li> </ul>	<ul style="list-style-type: none"> <li>• Project Topic Discussion</li> <li>• Discuss project/proposal details</li> </ul>
<b>2/4</b> Proposal due(Wednesday 4pm)	<ul style="list-style-type: none"> <li>• Final proposal</li> </ul>	<ul style="list-style-type: none"> <li>• Proposal Draft 2</li> </ul>	<ul style="list-style-type: none"> <li>• Proposal Draft 1</li> </ul>
<b>2/11</b>	<ul style="list-style-type: none"> <li>• Research power electronics circuits and come up with preliminary design</li> </ul>	<ul style="list-style-type: none"> <li>• Design state machines for micro-controller</li> <li>• Finalize parts</li> </ul>	<ul style="list-style-type: none"> <li>• Prepare documents for design review</li> <li>• Order parts and request for funding</li> </ul>
<b>2/18</b> Design Review sign up(Opens Monday closes Friday)	<ul style="list-style-type: none"> <li>• Design schematics for PWM and converter</li> <li>• Work on design review document</li> </ul>	<ul style="list-style-type: none"> <li>• Design and implement temperature and proximity sensor units</li> <li>• Work on design review document</li> </ul>	<ul style="list-style-type: none"> <li>• Learn Android App programming for Bluetooth</li> <li>• Work on design review document</li> </ul>
<b>2/25</b> Design Review	<ul style="list-style-type: none"> <li>• Implement power electronics and PWM for stove</li> </ul>	<ul style="list-style-type: none"> <li>• Implement state transition on controller based on button presses and sensor inputs</li> </ul>	<ul style="list-style-type: none"> <li>• Preliminary Android app that allows a user to input recipe manually</li> </ul>
<b>3/4</b>	<ul style="list-style-type: none"> <li>• Program controller to send PWM signal to the stove</li> </ul>	<ul style="list-style-type: none"> <li>• Perform reliability and accuracy testing of sensors</li> <li>• Obtain performance data of control stove</li> </ul>	<ul style="list-style-type: none"> <li>• Preliminary Android app that allows transmission of simple data to Bluetooth module</li> </ul>
<b>3/11</b> Individual Progress Reports due(Wednesday 4pm)	<ul style="list-style-type: none"> <li>• Complete individual progress report</li> <li>• Program controller to display recipe on LCD</li> </ul>	<ul style="list-style-type: none"> <li>• Complete individual progress report</li> <li>• Submit 1<sup>st</sup> draft of power circuitry PCB</li> </ul>	<ul style="list-style-type: none"> <li>• Complete individual progress report</li> <li>• Implement Bluetooth connectivity for controller</li> </ul>

<b>3/18</b> Spring Break	<ul style="list-style-type: none"> <li>• Prepare slides for mock up demo (power electronics and PWM)</li> </ul>	<ul style="list-style-type: none"> <li>• Prepare slides for mock up demo (state transitions/algorithm)</li> </ul>	<ul style="list-style-type: none"> <li>• Prepare slides for mock up demo (slides for Bluetooth transmission)</li> </ul>
<b>3/25</b> Mock-up Demos Presentation sign up(Opens Monday closes Friday)	<ul style="list-style-type: none"> <li>• Perform testing on the reliability and accuracy of sensors in affecting the stove</li> <li>• Prepare for mock up presentation</li> </ul>	<ul style="list-style-type: none"> <li>• Finish power circuitry PCB and submit 2<sup>nd</sup> draft if necessary</li> <li>• Prepare for mock up presentation</li> </ul>	<ul style="list-style-type: none"> <li>• Prepare for mock up presentation</li> <li>• Finish android app for recipe transmission via Bluetooth</li> </ul>
<b>4/1</b> Mock-up Presentations(Web sign-up, EL 159) Last day to request 1 <sup>st</sup> revision PCB(Friday)	<ul style="list-style-type: none"> <li>• Fix remaining issues with power electronics and display</li> </ul>	<ul style="list-style-type: none"> <li>• Fix remaining issues with control logic, sensors and buttons</li> </ul>	<ul style="list-style-type: none"> <li>• Ensure proper transmission of data from the Android device to the Bluetooth module</li> </ul>
<b>4/8</b> Last day to request final revision PCB(Friday)	<ul style="list-style-type: none"> <li>• Ensure completion of system including the house of power electronics and stove</li> </ul>	<ul style="list-style-type: none"> <li>• Debugging and fine-tuning of control logic, sensors and buttons</li> </ul>	<ul style="list-style-type: none"> <li>• Debugging and fine-tuning of transmission of data from the Android device to the Bluetooth module</li> </ul>
<b>4/15</b> Demo and Presentation sign up(Opens Monday closes Friday)	<ul style="list-style-type: none"> <li>• Prepare final report with emphasis on project deliverables(power electronics, display)</li> </ul>	<ul style="list-style-type: none"> <li>• Prepare final report with emphasis on project deliverables (control logic, sensors)</li> </ul>	<ul style="list-style-type: none"> <li>• Prepare final report with emphasis on Bluetooth transmission of recipe</li> </ul>
<b>4/22</b> Demos and Presentation	<ul style="list-style-type: none"> <li>• Prepare for demonstration with focus on power electronics and display</li> </ul>	<ul style="list-style-type: none"> <li>• Prepare for demonstration with focus on control logic, sensors and buttons</li> </ul>	<ul style="list-style-type: none"> <li>• Prepare for demonstration with focus on Bluetooth transmission data</li> </ul>
<b>4/29</b> Presentations Final Papers and lab notebook due(Wednesday 4pm)	<ul style="list-style-type: none"> <li>• Finish up final report</li> </ul>	<ul style="list-style-type: none"> <li>• Last check on all equipment for check out</li> </ul>	<ul style="list-style-type: none"> <li>• Prepare presentation</li> </ul>

## VII. Citations

- [1] LM2576/LM2576HV Data Sheet. Texas Instruments Inc., Texas. [Online]  
Available: <http://www.ti.com/lit/ds/symlink/lm2576.pdf>
- [2] Basic Calculation of a Buck Converter's Power Stage. Texas Instruments Inc., Texas.  
[Online] Available: <http://www.ti.com/lit/an/slva477a/slva477a.pdf>
- [3] T9A Series Relay Data Sheet. Tyco Electronics Connectivity, Pennsylvania. [Online]  
Available: [http://www.sparkfun.com/datasheets/Components/T9A\\_DS.pdf](http://www.sparkfun.com/datasheets/Components/T9A_DS.pdf)
- [4] 2N3904 Small Signal NPN Transistor Data Sheet. ST Microelectronics, Switzerland.  
[Online] Available: <http://www.sparkfun.com/datasheets/Components/2N3904.pdf>
- [5] GP2Y0A21YK/GP2Y0D21YK Data Sheet. Sharp Microelectronics, Washington. [Online]  
Available: <http://www.sparkfun.com/datasheets/Components/GP2Y0A21YK.pdf>
- [6] MLX90614 Family Data Sheet. Melexis Microelectronic Systems, Ieper, Belgium. [Online]  
Available: <http://www.melexis.com/Asset/IR-sensor-thermometer-MLX90614-Datasheet-DownloadLink-5152.aspx>
- [7] IEEE Code of Ethics [Online]  
Available: <http://www.ieee.org/about/corporate/governance/p7-8.html>
- [8] Safety. ECE444 Administrative Manual Spring 2013 [Online]  
Available: [http://fabweb.ece.uiuc.edu/lab/manual/admin\\_manual\\_sp2013.pdf](http://fabweb.ece.uiuc.edu/lab/manual/admin_manual_sp2013.pdf)
- [9] Bluetooth Shield Data Sheet [Online]  
Available: [http://www.seeedstudio.com/wiki/Bluetooth\\_Shield](http://www.seeedstudio.com/wiki/Bluetooth_Shield)
- [10] Bluetooth Shield Arduino Code [Online]  
Available: [http://www.seeedstudio.com/wiki/Bluetooth\\_Shield](http://www.seeedstudio.com/wiki/Bluetooth_Shield)
- [11] Android App Development [Online]  
Available: <http://developer.android.com/training/basics/firstapp/index.html>