Prepaid Energy System

Sahin Okur, Michael Cuervo, Md S. Rahaman, and Youssef Ojeil.

University of Central Florida, Department of Computer and Electrical Engineering, Orlando, Florida, 32816, U.S.A.

***Abstract* — This paper presents the design of a prepaid energy system that facilitates mobile payment made by smart phone using NFC technology. In addition to making payments, the smart phone can be used to access metrology data collected from a power meter and to receive alerts such as notification of payment, low balance from the designed system. The designed system monitors the power usage and turns on/off electricity supply based on the remaining balance. Alerts will be sent to the smart phone if the balance drops below a threshold.**

***Index Terms* — Microcontroller, Simple Mail Transfer Protocol, Wi-Fi Protected Access, Radio-Frequency Identification, Near-Field Communication,**

I. INTRODUCTION

The prepaid energy system is a pre-payment solution that will allow consumers to pay their electric utility bill as they go. It utilizse radio-frequency identification (RFID) technology so that users can add credit to the system just by using their smart phones and an app like Android Pay. The system is also connected to the internet to send real-time data and alerts directly to the users' smart phones. The user will have an extensive amount of data and statistics regarding their usage right in the palm of their hands.

The inspiration behind this project comes from the issue of consumers defaulting on utility payments. Utility companies provide their services to the customer and at the end of the month the customer does not pay be it due to financial crisis or irresponsibility. This results in a waste of energy and a loss of revenue. A scenario of this would be in a college town where many homeowners rent out their rooms for college students. As we know, college students are usually financially burdened, but with the pre-paid system, paying the utility bills becomes less of a burden. Residents would have the option of monitoring their energy usage and determine how much energy they are actually using on an hourly or daily basis. In addition, the prepaid system would allow them to budget their energy costs by paying in advance, or by paying in increments, similar to financing. With the phone app, they can monitor their usage and receive low balance alerts from within their home. By helping the residents, the prepaid system makes it easier for the property owners and cause fewer complications between them and the power companies. This will be very beneficial to the power industry and will allow them to obtain their profits on time.

If residents fail to replenish their energy meter balance, the power will automatically be cut-off. If this happens, the prepaid system will still be operable so that the residents can add money to their accounts and instantly turn the power back on. By having a system that automatically regulates the power, we eliminate the need for the power companies to send a technician out, and we save them the resources and labor.

This is also useful for residents that stay at their home temporarily throughout the year. If a resident decides to leave their house due to work, and in advance knows that he will only be staying in the house for a miniscule margin of time, they can decide to purchase less power for the following days.

Other times that this can come in handy is for people with vacation homes. Year around they are not always living in their house. If the house is used for two weeks out of the whole year, it is redundant to bill the people for the remaining fifty weeks when the person is simply not using the house.

The system is designed to measure and regulate three phase power systems. However, it can easily be scaled down for a one-phase power system, which is what is used for most residential houses. This flexibility allows both commercial and residential areas to make use of the prepaid system.

II. DESIGN

A. Microcontroller Unit

The microcontroller unit for our system lies at the center of our design. It is responsible for communication between the RFID transceiver, power meter, relay, and smartphone. We decided to go with TI's CC3200 wireless microcontroller (MCU) for our project since it features a network processor and robust MCU within a single chip. The applications MCU features an ARM Cortex-M4 core running at 80 MHz as well as 256KB of RAM. The network processor features another ARM Cortex core that offloads the application MCU. It runs at a supply voltage range of 3.3V so that we can maintain our one voltage rail.

Communication between the CC3200 and RFID transceiver is done via SPI. Once the RFID transceiver detects that the user wants to make a payment, it sends an interrupt to wake up the MCU. Data is sent and received through the Master Out Slave In (MOSI) and Master In Slave Out (MISO) lines into and out of the transceiver's FIFO until the process is complete. When the MCU successfully receives the payment, it adds it to the user's current balance and flashes the LED to notify the user. At this time, the MCU will connect to an Simple Mail Transfer Protocol (SMTP) server to send the user a text message and e-mail confirmation.

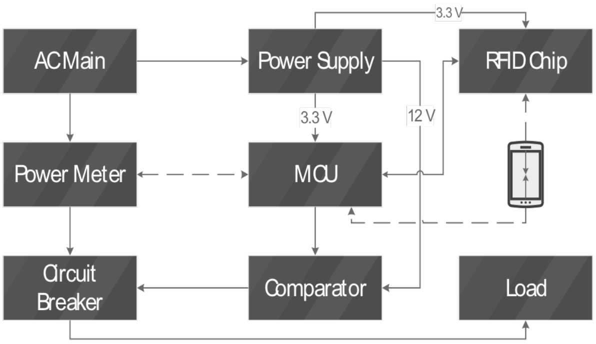


Fig. 1. Prepaid Energy System Block Diagram

Communication between the CC3200 and power meter is done via Wi-Fi. The power meter is interfaced to a CC3100 network processor that enables the power meter to host its own wireless network. The CC3200 retrieves metrology data from the meter at one minute intervals. When the timer interrupt is initiated on the MCU, it will disconnect from its current wireless network and connect to the wireless network of the power meter. The wireless network on the power meter is secured using Wi-Fi Protected Access 2 (WPA2) so that only the CC3200 will have access with pre-installed credentials. The network processor on the power meter hosts its own Hypertext Transfer Protocol (HTTP) web server which stores and displays the metrology data. To retrieve the data, the CC3200 acts as an HTTP client to the power meter's server and uses HTTP GET commands. GET commands are requests sent by the client to the server for specific token values. On the power meter side, when the

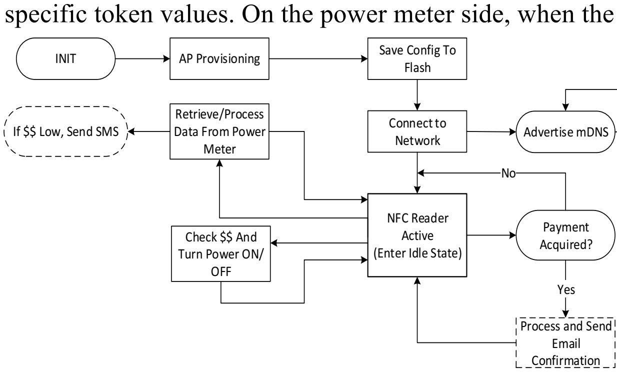


Fig. 2. Software Flow Diagram

server receives these requests, it responds by replacing the token with the values stored in memory. For example, if the CC3200 sends a request for the token "voltage," the power meter will replace this token with "120" and respond. The CC3200 then stores the values in memory to be displayed later, and deducts from the user's balance, if power has been consumed. When completed, the CC3200 reconnects to the internet and sleeps until the next interrupt. Conditional statements are set up so that if the user's balance drops below a specified threshold, the user will receive a text message alert.

Furthermore, the CC3200 hosts its own HTTP web server so that the user can access its webpages from their smartphone, or any web browser. These HTML web pages are stored within the flash memory of the CC3200. They display the following information: total kWh, current kWh usage, average kWh usage, estimated time remaining based on current load, estimated time remaining based on average load, relay status, number of payments made, and current balance. The smartphone retrieves the data similar to the power meter, by acting as a client and sending HTTP GET commands to the CC3200 server.

Finally, if the user's balance does drop to $0, the CC3200 toggles HIGH its GPIO pin connected to the relay to disconnect the load.

The software that accomplishes all of this features over 10,000 lines of code, utilizing extensive libraries including TI's SimpleLink and NFC libraries. The software also features a real-time operating system (RTOS) which allows us to prioritize tasks and handle interrupts efficiently. The main.c source file includes over 2,000 lines of code and handles the calling of the Wi-Fi, NFC, and HTTP functions.

The main portion of the code utilizes a switch case statement structure to call the functions as seen below:

```
switch(AppState)
        {
        case CONNECT:
                AppState = NFC_READER;
                break;
        case NFC_READER:
                AppState = SEND_EMAIL;
                break;
        case SEND_EMAIL_sta:
                AppState = GET_DATA;
                break;
        }
```

Fig. 3. Case Structure Example

The program loops in a while loop until an interrupt causes the AppState variable to change. The overall flow diagram of the software can be seen below:

B. Microcontroller Unit PCB

The system design except for the power supply and RFID transceiver is implemented on its own four layer printed circuit board. The top and bottom layers were designed for signals, and the middle layers for ground and power. The RF section of the board took the highest priority and needed to be designed correctly in order to avoid performance degradation. We accomplished this by following TI's design guidelines below:

1. Place the antenna on an edge or corner of the PCB.
2. Make sure that no signals are routed across the antenna elements on all the layers of the PCB.
3. Most antennas, including the chip antenna used on the BoosterPack, require ground clearance on all the layers of the PCB. Ensure that the ground is cleared on inner layers as well.
4. Ensure that there is provision to place matching components for the antenna. These need to be tuned for best return loss once the complete board is assembled. Any plastics or casing should also be mounted while tuning the antenna as this can impact the impedance.
5. Ensure that the antenna impedance is 50 Ω as the device is rated to work only with a 50 Ω system.
6. Ensure that the antenna has a near omni-directional pattern.

C. Radio-Frequency Identification

The prepaid system will utilize Radio Frequency Identification (RFID) technology for its method of payment. The data that is transferred is similar to the data found in barcodes when scanned. However, unlike barcodes, RFID does not need to have the scanner and tag within line of sight of each other.

A specific branch of RFID that we are using is NFC, Near-Field Communication. It consists of a high frequency of 13.65 MHz and is compatible with most cell phones. As the name suggests the average distance of communication between the reader and tag is approximately 2 cm. In this subsystem, the reader chosen is a TI RFID chip called the TRF7970A. This reader supports all standards of NFC as well as modes. The three modes it operates in are Reader/Writer, Peer-to-Peer and Card emulation. The following is a table that depicts all different modes of the TRF7970A.

Reader/Writer		Card Emulation		Peer-to-Peer	
Tech	Bit Rate (kbps)	Tec h	Bit Rate (kbps)	Tech	Bit Rate (kbps)
NFC A/B (ISO 14443A/ B)	106, 212, 424, 848	NF C A/B	106	NFC A	106
NFC-F (JIS:X63 19-4)	212, 424	N/A	N/A	NFC -F	212, 424

Fig 4. Bit Rate Table

The TRF7970A is one of the most versatile chips within the RFID industry that also contains a buffer of 127 bytes as well as an ultra-low power mode of less than 1 µA. This buffer greatly helps the communication that takes place between the tag, reader and MCU since more data can be sent and stored within the buffer at once. The low power modes come in handy with the system since the chip will always be polling for different NFC tags with different standards at specific intervals. For the case of this project we have disabled the polling for different standards since we know we are using ISO14443A. This decreases the weighted code on the microcontroller tremendously and further more makes the whole system more power efficient. The method of NFC used between the devices was NDEF. This consisted of several commands the MCU had to send to the reader for a complete communication. The commands between the three devices happen sequentially within the small amount of time that the phone is within range of the reader. The functions are split into functions used for reading and writing. The functions used to read were: RATS(), NDEFApplicationSelect() ,CapabilityContainerSelect(), and SelectNDEF(). The Request application select came first followed by an NDEF application select followed by the capability container and finally the select NDEF. The function used for reading was ReadBinary(), and the function used for the wait time extension request was WTX(). Since the TRF7970A has just one buffer, every function that is written for it must first begin with the bytes 0x08 to clear the FIFO buffer followed by 0x91 to send with CRC followed by 0x3D to write continuously and finally the rest of the bytes to complete the command. The next set of bytes after the write continuous determine the amount of bytes that the RFID transceiver will send once it reaches that given function.

The subsequent function must have successfully completed before it moves on to the next and finally to the final one for a complete communication. The completion of every command can be verified if the reader receives the bytes 0x90 and 0x00. The main problem that was found within NDEF was that the tag being the phone was sending a wait time extension request with the bytes 0xF2 and 0x01 which was creating an unsynchronized communication between the two devices. The method used to solve this was to create a function that acknowledged this request and place it between the different stages of communication. Acknowledging this wait time request meant that the reader must send back those bytes.

D. Android Application

As mentioned earlier the TRF7970A will be the reader operating in Reader/Writer mode at 106 kbps scanning for ISO 14443A tags. The phone will also be the main hub as well as main user interface piece of this project. It will be the connection between the system and the user. The tag that is being used for this project is a phone instead of a passive NFC tag. Android has an option of HCE which stands for host card emulation. The main requirement for HCE is Android 4.4. The phone being used is a Nexus 4 Android that contains a Broadcom NFC controller. This phone is also able to operate in all three modes. In our case we had it operate in card emulation mode. Once the phone is in this mode it emulates an ISO 14443A tag which is due to its controller. The mobile app that was created for the phone was mainly to have the phone operate in card emulation mode. The bytes within the Capability Container as well as the NDEF application and AID had to match the bytes that the code within the reader contained. The final piece of the NDEF is the message. NDEF messages can hold multiple NDEF records but in our case we used one message and one record. The NDEF TNF in our case is a well-known type with the record being a text which is specified by having bytes 0x01 in the type field.

The picture below shows in simplicity how an NDEF message can hold multiple NDEF records and how each record has its own header to specify the record needed as well as the payload for the different data to send. Each payload can be of different type such as URI, text or smart posters.

Since we are not dealing with real credit we are just sending numbers from the phone to the reader as a method of payment. We want the user to receive the full experience of having to use a pre pay app and therefore the app contains text fields that emulate android pay where the user enters their credit card information and the amount they are willing to pay. Within the app one of the text boxes that was created will be used for the payment amount. Once the user enters the amount, it will be saved into a variable which is the NDEF message which will then be sent out to the reader, processed by the reader and then sent to the MCU. This guarantees the user the flexibility of paying as much as they would like to as well as use the card of their choice.

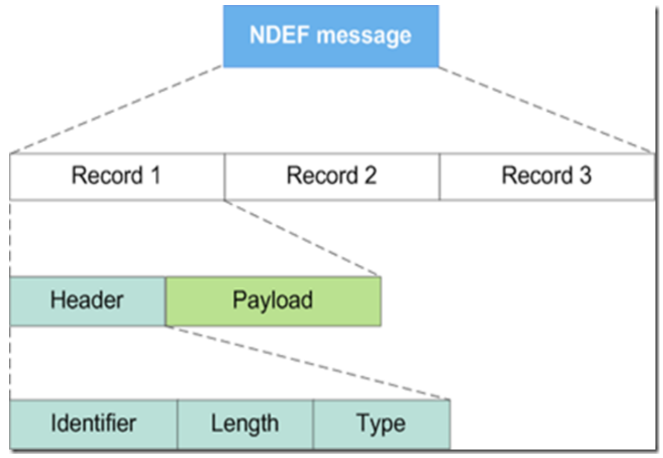


Fig 5. NDEF Message Diagram

The second purpose of using the phone is the ability for the user to receive their data on hand at any moment. Therefore, the app will have the option for the user to press a button that will have them open the server that contains all the user's power consumption thus far as well as other metrology.

E. Circuit Breaker System

The circuit breaker system is being used to control the load. As long as the customers has a significant balance the switch is allowing power to flow to the load. If the balance were to reach zero, then the circuit breaker would cut the power to the load until there is a payment made by the customer. Therefore, the circuit breaker is acting as the electrical switch that determines when the customer has power or not.

We decided on a single phase solid state relay (Fotek 75 DA) because it has a DC input control that could be easily connect and controlled by the microcontrollers.

After running tests on the relay with a load, we quickly realized that our microcontroller could only supply a maximum voltage of 3.3 V, which was not sufficient enough to trigger the relay switch. The solution we came up with was to use a comparator circuit in order to increase our output voltage of the MCU, to be able to trigger the relay which needed a 12V input to turn on.

The MCU provides a 3.3 V to the comparator circuit which steps up the voltage to 12 V which opens the relay. The comparator circuit can take the 0V volt input instead of the 3.3 provided by the MCU in order to trigger the relay. We assume the relay will be open for the majority of its operations and can conclude that energy will be saved using it this way. In addition to increasing the voltage,

a byproduct of using the comparator circuit was that it made the system more energy efficient.

Fig. 6 and Fig,7 bellow show the simulation for the comparator circuit (achieved in Mulisim Circuit Simulation Software)

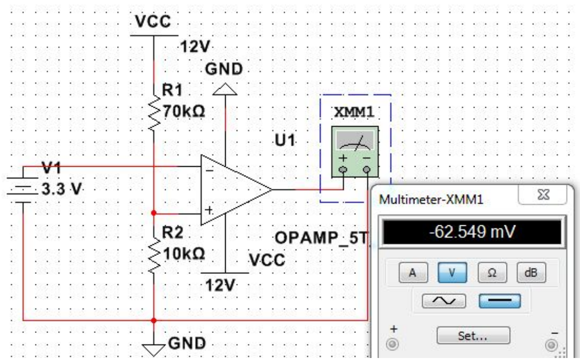


Fig 6. Circuit Simulation

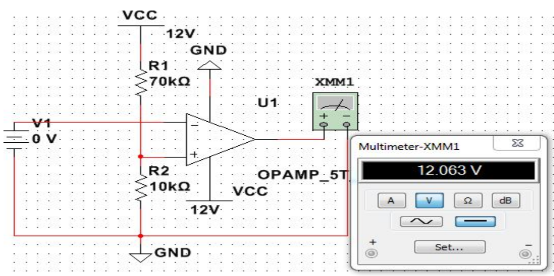


Fig 7. Circuit Simulation

F. Three-Phase Smart Meter

The Three-Phase Smart Meter measures the electric power in Watts or KWh of three-phase electrical systems. EVM430-F6779 is a three-phase electronic power meter designed by Texas Instruments. The objective of this project is to implement a prepaid energy meter system using the EVM430-F6779. Slaa577 was a resource document used to obtain the technical details of the EVM430-F6779. For testing and demonstration purposes, Phase C was used which is powered by the AC main line. Supply voltages from the power rail for the MCU and LCD are directly converted from Phase C. The EVM430-F6779 provides all the necessary sampling, current sensors, voltage sensors, and LED lighting for indication of all different types of energy measurement. Fig 8 and 9 below shows the EVM430-F6779 Power Meter connections.

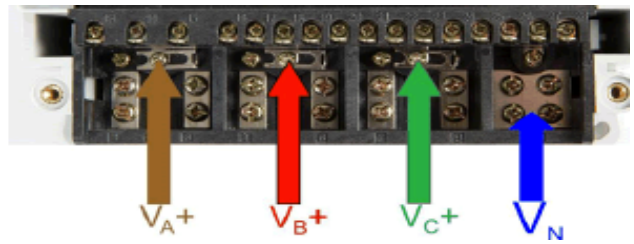


Fig 8. Voltage inputs of EVM430-F6779 Three-Phase Power Meter (Extracted form TI website)

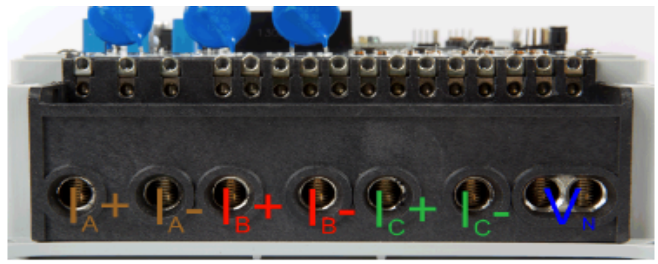


Fig 9. Current inputs of EVM430-F6779 Three-Phase Power Meter (Extracted from TI website)

Multiple Loads have been used to view the energy measurement. The Loads are directly connected to the AC side of the solid-state relay. For demonstration purposes, a hair dryer and laptop have been chosen. The hair dryer consumes large amounts of power in a short period of time, which makes it a useful test candidate. Built in, the LCD has a resolution of 0.01. Because of the high power consumption defined in this project, the decimal point will vary easily, making the LCD an easy demonstration for presentations. The power meter has a RS-232 built in serial data transfer protocol which can prove to be useful for calibration purposes. Fig. 9 below shows the Graphic User Interface (GUI):

Fig10. GUI interface

Calibration formula:

$$Correction\ (\%) = \left(\frac{Value_{observed}}{Value_{desired}} - 1\right) * 100$$

Calibration is essential to the performance of any power meter. Therefore, it is necessary to step through the calibration process defined in the Texas Instruments slaa577 technical resource document.

The CC3100 WiFi chip can be interfaced with the power meter, allowing data transfer to the Microcontroller. The power meter has its own server that allows values of energy, phase, frequency, and even energy pulses to be retrieved. When data is collected from the power meter itself this pulse is shown. Due to Texas Instruments convention of every 6400 pulses being equivalent to 1 KWh, this value was implemented in the code to convert pulse to money and time. This conversion will provide data into the power consumption of the user and how long access to the power is granted based on the money provided. With the tap of a phone, the user can recharge at any time and increase the amount of power allotted. The power meter has a background process using an interrupt as a trigger to collect voltage and current sensors. After sufficient samples have been accumulated, the foreground process calculates final voltage, current, active, reactive, and other system parameters. The background process calculates the frequency in terms of samples per main cycle.

$$Frequency(Hz) = \frac{Sampling\ rate(\frac{sample}{second})}{Frequency(\frac{samples}{cycles})}$$

Figure 11 is a flow diagram that shows the process of data collection by the power meter,

G. Power Supply

For the power supply, there were many designs that where looked at. Some of the sites that where referenced were Texas Instruments Reference designs, Webench and Digikey. When designing the power supply addressing the required parameters is the first step. Some characteristics that are analyzed when comparing designs is the circuit's input voltage, output voltage, output current, and the efficiency.

There are three devices that need DC power source, the solid state relay, CC3200 microcontroller and RF transceiver. The relay requires a 12 volt input, while the microcontroller and RF transceiver needs 3.3 volts supplied. For the output power needed to power the devices the required current that each part draws were needed, this value changes depending on the devices use and if it is

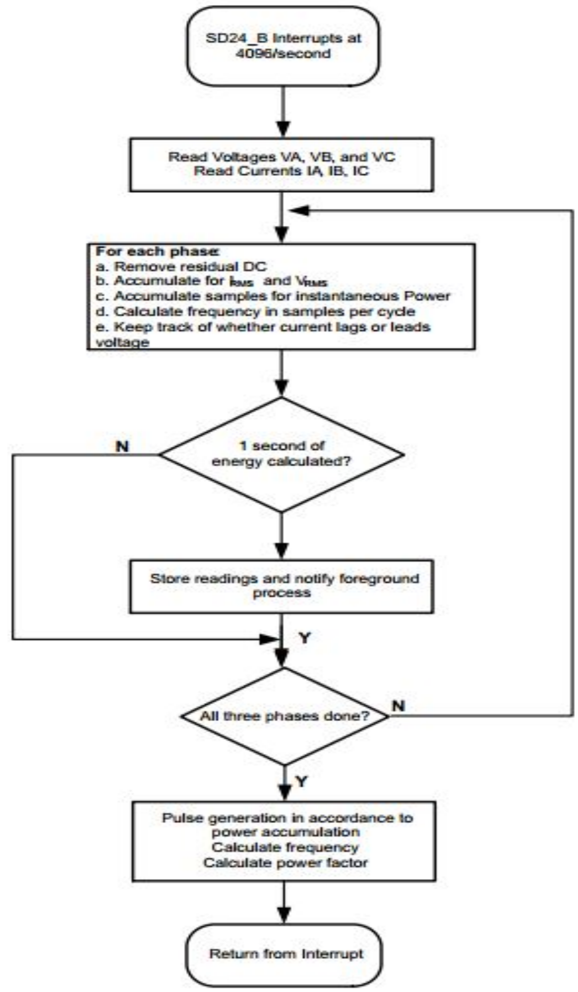


Fig. 11 shows a process flow diagram of the data collection by the power meter: (extracted from TI website)

calibrating. To insure all possible cases, the max current drawn by the microcontroller and RF transceiver was used as the benchmark for the output power. For the power supply it will be powered by the wall outlet.

The power supply was divided into two stages, the first stage has an output voltage of 12 Volts and must be able to power the second stage along with the solid state relay. The second stage has to have an output of 3.3 volts and must be able to supply power to the CC3200 microcontroller and it's peripherals such as the Radio Frequency transceiver.

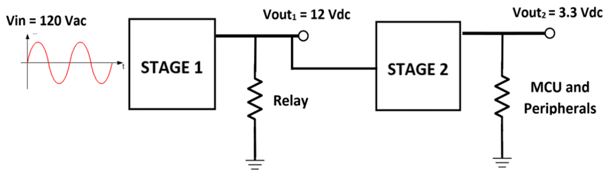
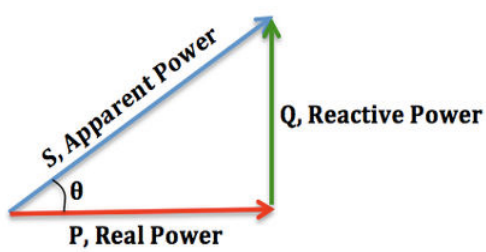


Fig12. Power Supply System

When first researching for the power supply the idea of power factor correction came up. This is a method used to optimize the effective output power. When the current and voltage phase are not synchronized the power that can used to power any loads gets diminished. In several reference designs a power factor correction circuit was implemented as an intermediate stage within the power supply system. The figure below shows the relationship between reactive, apparent, and true power. The optimal case is that the current and voltage phase, resulting in a power factor of one, and reducing the reactive power to zero.



$$\text{Reactive power} = \sqrt{(\text{Apparent power})^2 - (\text{True power})^2}$$

Fig13. Power Triangle

After reading more into this, it was noted that the use of power factor correction circuits is preferred and need when the desired load requires a large amount of power. This ranges generally starts about 20 watts. Since for this project the amount of power necessary to power the device was significantly less relative to the 20 watt mark, PFC was not necessary.

For the first stage a design from Texas Instruments reference page was used. The PMP8764 was used, it utilizes a Flyback topology. After looking at several designs they all follow a similar pattern as the architecture used when constructing these circuits. Most of the circuits had an IC chip that uses a feedback as well as pulse width modulation to regulate the output voltage. Several of these circuits start off with a full bridge rectifier, include a IC chip that has a feedback and a transformer.

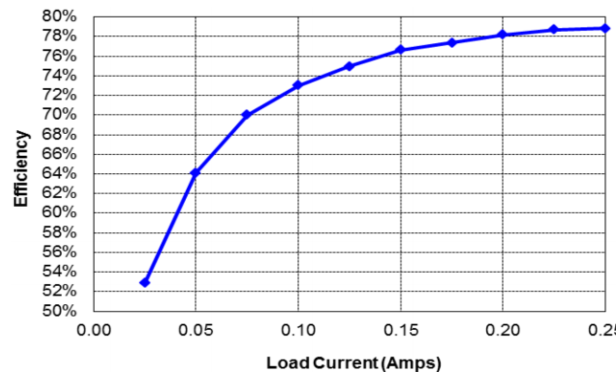


Fig14. Stage 1 Efficiency

For the second stage a DC to DC buck converter with another IC chip was used. This design was exported from Webench. This one was choose because at lower currents such as 600 mA the efficiency was in the 95% range. Another benefit with going with this design is that if there are any fluctuations with the amount of current being drawn that the efficiency still stays around the same percentage.

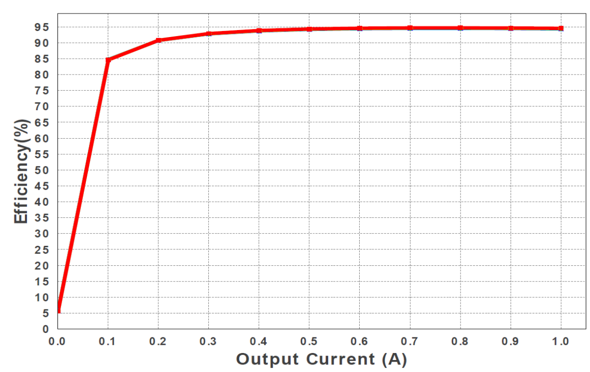


Fig15. Stage 2 Efficiency

Once the designs for the two stages was designed on. Recreating the circuits on Eagle Cad was the next step. Some of the parts in the designs when not in stock. Digikey is a large distributer of parts for Texas Instruments and many other companies. For every part there is a description and a link with the data sheets. These data sheets where often referend to when searching of alternative parts. Digikey also provides drawing for many parts. These were used when creating a footprint in Eagle Cad.

The PCB was then sent to OSH park to get manufactured. Afterward the PCB was sent to Quality Manufacturing Service to get the components soldered onto the board. Once tested the power supply soldered to a wall outlet cable and connected to the microcontroller's PCB via headers and jumper cables.

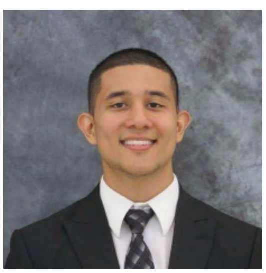
III. CONCLUSION

In conclusion, this project was an enormous learning experience for all of the team members. As electrical engineers we are accustomed to dealing with hardware so taking on a project that was software intensive was challenging. Another challenge was the time management and finding times that accommodate everyone's schedule. This project also showed us how to obtain information through independent research as opposed to the conventional way of learning from a professor in a classroom environment. Throughout the testing phase many errors occurred which exposed us to troubleshooting both hardware and software. The four of us learned a lot about programing microcontrollers, different types of wireless communications, power meters, and power circuits.

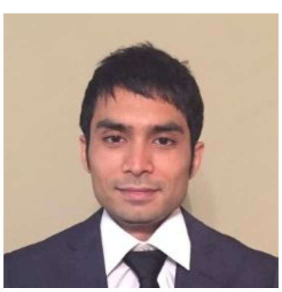
ACKNOWLEDGEMENT

The authors wish to acknowledge Texas Instruments for the proposal of the idea of this project, as well as for their sponsorship. They would also like to thank and acknowledge Dr. Chung Yong Chan for his supervision and assistance throughout the course of the project.

BIOGRAPHY



Michael Cuervo will graduate from the University of Central Florida with honors and receive his Bachelors of Science in Electrical Engineering in May of 2016. He worked as an intern at Lockheed Martin in the Logistics Department and Training Systems Team along with other System Engineers. After graduating Michael Cuervo has accepted a Test Engineering position at Texas Instruments.



Md Rahaman is currently a senior at the University of Central Florida and will receive his Bachelors of Science in Electrical Engineering in May of 2016. He has finished Associate degree from Valencia College and transfer to University of Central Florida in 2014. He is planning to continue studying in a graduate school to specialize in Microwave engineering.

Sahin Okur is currently a senior at the University of Central Florida. He plans to graduate with his Bachelor of Science in Electrical Engineering in May of 2016. He has accepted a position with Texas Instruments as an Applications Engineer.

Youssef Ojeil is currently a senior at the University of Central Florida. He plans to graduate with his Bachelor of Science in Electrical Engineering in May of 2016. His goal is to receive a degree of higher education in Electrical Engineering.

REFERENCES

[1] http://www.allaboutcircuits.com/textbook/alternating-current/chpt-11/practical-power-factor-correction/

[2] http://www.ti.com/lit/an/slaa577g/slaa577g.pdf

[3] http://www.onsemi.com/PowerSolutions/content.do?id=15067

[4] http://www.ti.com/tool/EVM430-F6779

[5] http://www.ti.com/lit/ds/symlink/cc3200.pdf

[6] http://www.digikey.com/product-search/en