Smart Energy Monitor

A project report submitted in partial fulfillment of
the requirements for the degree of

Bachelor of Engineering

by

Shabbir Ahmed (Roll No. 7173)
Dylan Dcruz (Roll No. 7304)
Jonathan Pereira (Roll No. 7337)

Under the guidance of
Prof. Jayen Modi

DEPARTMENT OF ELECTRONICS
Fr. Conceicao Rodrigues College of Engineering, Bandra (W), Mumbai - 400050
University of Mumbai

March 17, 2018

Internal Approval Sheet

CERTIFICATE

This is to certify that the project entitled **"Smart Energy Monitor"** is a bonafide work of **Shabbir Ahmed(7173), Dylan Dcruz(7304), Jonathan Pereira(7337)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering** in **Electronics**

(Name and sign)
Supervisor/Guide

(Name and sign)
Head of Department

(Name and sign)
Principal

Approval Sheet

Project Report Approval

This project report entitled by **Smart Energy Monitor** by **Shabbir Ahmed, Dylan Dcruz, Jonathan Pereira** is approved for the degree of Bachelor of Engineering

Examiners

1.____________________________

2.____________________________

Date:

Place:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Shabbir Ahmed (Roll No. 7173) **(sign)**________________

Dylan Dcruz (Roll No. 7304) **(sign)**________________

Jonathan Pereira (Roll No. 7337) **(sign)**________________

Date: March 17, 2018

Abstract

In 2014, the Electricity Consumption per capita in India was 805.6KWh which is equivalent to 637.43 kg of oil per capita. Over 58 percent of this electricity is produced from non renewable sources of energy. Our dependence on production of energy from non renewable sources of energy makes India both a major greenhouse gas emitter and one of the most vulnerable countries in the world to projected climate change. The country is already experiencing changes in climate and the impacts of climate change, including water stress, heat waves and drought, severe storms and flooding, and associated negative consequences on health and livelihoods. It is imperative that India rapidly adopts renewable sources of energy like solar and wind. But in addition to that it is also the responsibility of the Indian people to monitor their energy consumption and reduce their carbon footprint. The Smart Energy Monitor helps the Indian consumer to reduce and monitor their household energy consumption by providing insights to consumption of electricity by individual electrical appliances. The Smart Energy Monitor connects directly to your electricity panel and uses a mobile app to tell you what devices and appliances are drawing power and when. The monitor listens to the electronic signature of each device and uses algorithms to identify them and monitor their power consumption. It also presents real-time and historical usage for each device. It will help the consumer track energy inefficient appliances and also their monthly usage. From these insights the consumer can reduce their electricity consumption thereby reducing their carbon footprint.

Acknowledgments

We have great pleasure in presenting the report on **"Smart Energy Monitor"**. I take this opportunity to express my sincere thanks towards the guide Prof. Jayen Modi, C.R.C.E, Bandra (W), Mumbai, for providing the technical guidelines, and the suggestions regarding the line of this work. We enjoyed discussing the work progress with him during our visits to department.

We thank Dr. Deepak V. Bhoir, Head of Electronics Dept., Principal and the management of C.R.C.E., Mumbai for encouragement and providing necessary infrastructure for pursuing the project.

We also thank all non-teaching staff for their valuable support, to complete our project.

Shabbir Ahmed(Roll No. 7173)

Dylan Dcruz(Roll No. 7304)

Jonathan Pereira(Roll No. 7337)

Date: March 17, 2018

Contents

List of Figures

Chapter 1

Introduction

The Smart Energy Monitor is based on the concept of Non Intrusive Load Monitoring (NILM) which is a process for analyzing changes in the voltage and current going into a house and deducing what appliances are used in the house as well as their individual energy consumption. Electric meters with NILM technology are used by utility companies to survey the specific uses of electric power in different homes. NILM is considered a low-cost alternative to attaching individual monitors on each appliance. The system can measure both reactive power and real power. Hence two appliances with the same total power draw can be distinguished by differences in their complex impedance. For example, a refrigerator electric motor and a pure resistive heater can be distinguished in part because the electric motor has significant changes in reactive power when it turns on and off, whereas the heater has almost none.
NILM systems can also identify appliances with a series of individual changes in power draw. These appliances are modeled as finite state machines. A dishwasher, for example, has heaters and motors that turn on and off during a typical dish washing cycle. These will be identified as clusters, and power draw for the entire cluster will be recorded. Hence "dishwasher" power draw can be identified as opposed to "resistor heating unit" and "electric motor". Thus designing a energy monitoring unit using this NLIM has many benefits. The overview of the system is as follow:

Figure 1.1: Basic Block diagram of the System

1.1 Motivation

How many planets would it take to support our lifestyle? As blunt as it may sound, the truth stands unchanged, staring at the face of the unknown future of the whole planet. It didn't take us long to open our hearts (and homes) to the amazing changes that technology brought into our lives. Amidst the ease and comfort, everything else seems to be collateral damage to us now. One such commodity is electricity.Electricity consumption rates from non renewable sources of energy is increasing at an alarming rate by the hour. Our increasing dependency on these replenishing sources calls for innovative ways to tap on our consumption rates, preferably on a daily basis. Electrical appliances used on a daily basis are monitored ambiguously, hence an increase in the prices may not be found. The outcome of this project, the Smart Energy Monitor is aimed to help the average Indian consumer monitor their appliance usage on a daily basis hence keep track of their consumption.

1.2 Objectives

1. To disaggregate load appliances using minimum hardware efficient and accurate classification algorithms.

2. To provide the user with Real-time monitoring and alerts on their smartphone.

3. To track the power consumption of individual load appliances and provide the user with an estimate of their monthly electricity bill.

Chapter 2

Literature Review

2.1 Power Measurement

Most current sensors can measure current in two directions. this means is that if we sample fast enough and long enough, we sure to find the peak in one direction and the peak in another direction. With both peaks known, it is a matter of knowing the shape of the waveform to calculate the current. In the case of line or mains power, we know that waveform to be a Sine wave. Hence the expression of AC current will be in a value known as RMS.

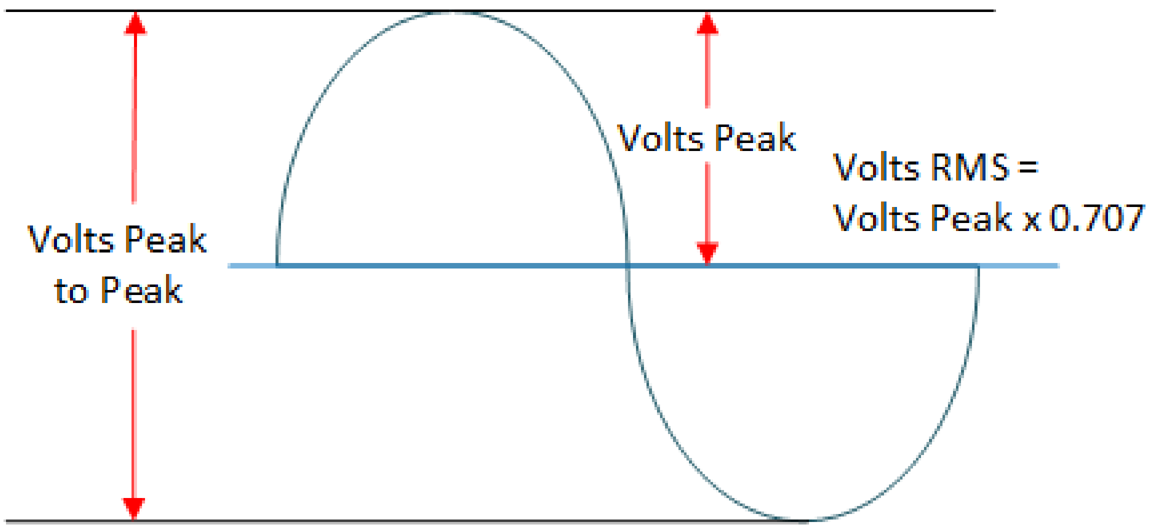

Figure 2.1: Measurement of Vrms

Conversion for a sine wave with a zero volt offset (like that in mains or line power) is performed as follows:

1. Find the peak to peak voltage (Volts Peak to Peak)

2. Divide the peak to peak voltage by two to get peak voltage (Volts Peak)

3. Multiply the peak voltage by 0.707 to yield rms volts (Volts RMS)

Having Calculated RMS voltage, is simply a matter of multiplying by the scale factor of the particular current sensor to yield the RMS value of the current being measured which can then be multiplied by the AC Voltage value in order to give the value of the Total Power Drawn.

$$I_{rms} = \frac{V_{pp} \times 0.707 \times ScaleFactor}{2} \tag{2.1}$$

$$P = I_{rms} \times 230 \tag{2.2}$$

2.2 Disaggregation Algorithms

Most NLIM systems provide implementations of two common benchmark disaggregation algorithms: Steady State Analysis and Combinatorial Optimisation(CO).

2.2.1 Steady State Analysis

The NILM methods based on steady-state analysis make use of steady-state features that are derived under the steady-state operation of the appliances. Real power (P) and Reactive power (Q) are two of the most commonly used steady state signatures in NILM for tracking On/Off operation of appliances. The real power is the amount of energy consumed by an appliance during its operation. If the load is purely resistive then the current and voltage waveforms will always be in phase and there will be no reactive energy. For a purely reactive load the phase shift will be 90 degrees, and there will be no transfer of real power. On the other hand, due to inductive and capacitive elements of the load, there is always a phase shift between current and voltage waveforms that generates or consumes a reactive power respectively.

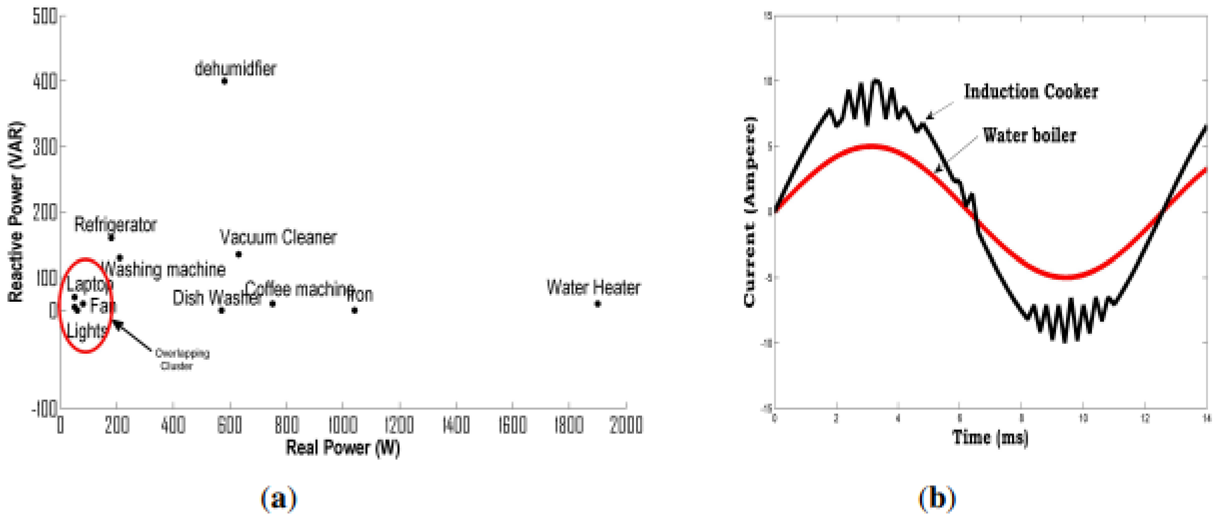

Figure 2.2: Load Distribution in PQ Plane and Current draw of Linear vs Non Linear Loads

Researchers have tried to disaggregate load using real power as a single feature and found out that high-power appliances with distinctive power draw characteristics such as electrical heaters and water pumps can be easily identified from the aggregated measurements. However this method does not take into account appliances with similar power draw characteristics. In addition, simultaneous state transitions of appliances leads to erroneous results. In order to address some of these issues, high power appliances can easily be differentiated by analyzing the step changes in real and reactive power features.

2.2.2 Combinational Optimization(CO)

Combinational Optimization finds the optimal combination of appliance states, which minimizes the difference between the sum of the predicted appliance power and the observed aggregate power, subject to a set of appliance models. Since each time slice is considered as a separate optimisation problem, each time slice is assumed to be independent. The complexity of disaggregation for T time slices is:

$$N_{Combinations} = K^N \tag{2.3}$$

where N is the number of appliances and K is the number of appliance states.
Since the complexity of CO is exponential in the number of appliances, the approach is only computationally tractable for a small number of modelled appliances. The error can be minimized by choosing the combination whose calculated power draw is the

closest to the measured power drawn.

2.2.3 K Nearest Neighbours(KNN)

In pattern recognition, the k-nearest neighbors algorithm (k-NN) is a non-parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression:

In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor. In k-NN regression, the output is the property value for the object. This value is the average of the values of its k nearest neighbors. k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms.

Both for classification and regression, it can be useful to assign weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. For example, a common weighting scheme consists in giving each neighbor a weight of 1/d, where d is the distance to the neighbor.

The neighbors are taken from a set of objects for which the class (for k-NN classification) or the object property value (for k-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

A peculiarity of the k-NN algorithm is that it is sensitive to the local structure of the data.[citation needed] The algorithm is not to be confused with k-means, another popular machine learning technique. Following example would help us to understand better : The test sample (green circle) should be classified either to the first class of blue squares or to the second class of red triangles. If k = 3 (solid line circle) it is assigned to the second class because there are 2 triangles and only 1 square inside the inner circle. If k = 5 (dashed line circle) it is assigned to the first class (3 squares vs. 2 triangles inside the outer circle).

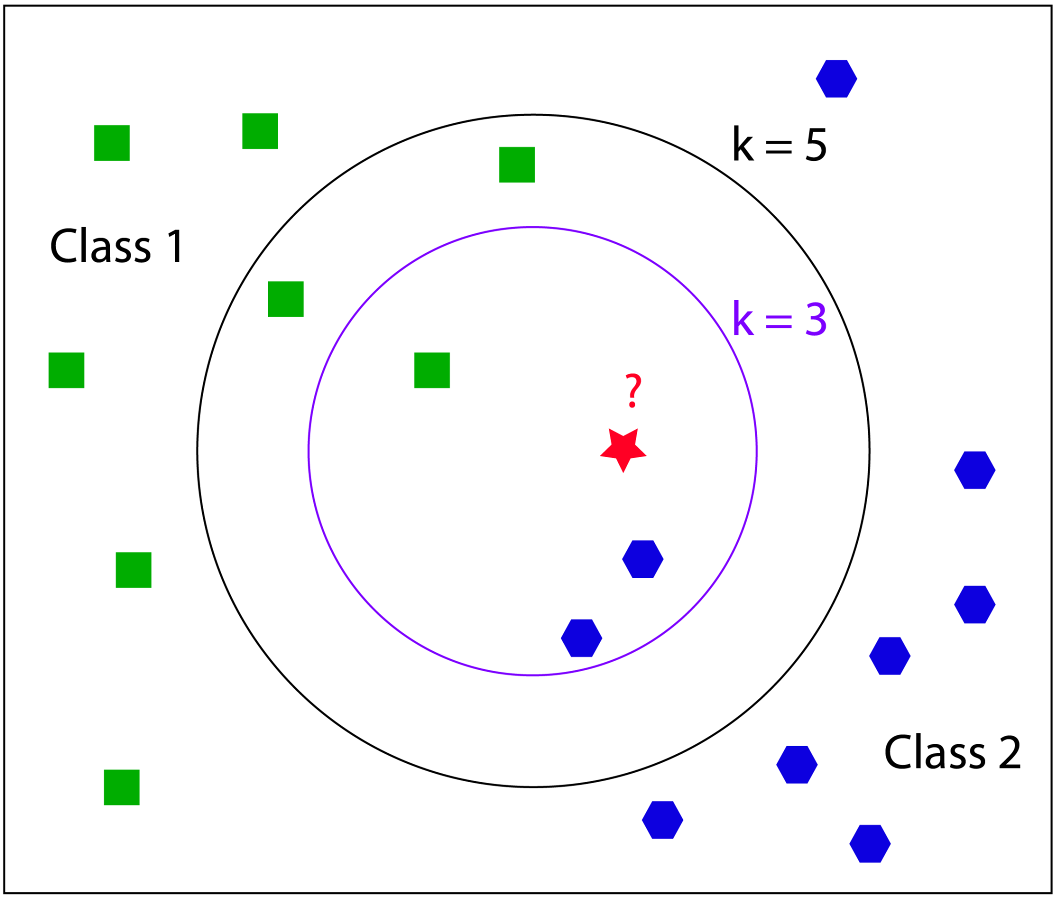

Figure 2.3: K Nearest Neighbour

2.3 Public Datasets

Apart from a common evaluation metric there is also a lack of reference dataset on which the performance of algorithm can be compared. It is quite obvious that the output of the load disaggregation algorithm is dependent on the source data, which often varies either due to difference in the number and type of appliances used in the experiment or due to the hardware used to extract the load signatures. In order to draw meaningful performance comparison of various NILM techniques, the availability of common datasets is critical. Motivated by this, recently the Reference Energy Disaggregation Data Set (REDD) and the Building-Level fUlly labeled Electricity Disaggregation dataset (BLUED) have been made publicly available in order to facilitate the researchers in the development and evaluation of new load disaggregation algorithms. The datasets contain high-frequency and low-frequency household power measurements primarily for the evaluation of steady-state as well as transient state NILM methods.

2.3.1 Reference Energy Disaggregation Dataset (REDD)

The data contains power consumption from real homes, for the whole house as well as for each individual circuit in the house (labeled by the main type of appliance on that circuit). The data is intended for use in developing disaggregation methods, which can predict, from only the whole-home signal, which devices are being used. The REDD data set contains two main types of home electricity data: high-frequency current/voltage waveform data of the two power mains (as well as the voltage signal for a single phase), and lower-frequency power data including the mains and individual, labeled circuits in the house.

2.3.2 Building-Level fUlly labeled Electricity Disaggregation (BLUED) Dataset

The BLUED dataset consists of voltage and current measurements for a single family residence in the United States, sampled at 12 kHz for a whole week. Every state transition of each appliance in the home during this time was labeled and time-stamped, providing the necessary ground truth for the evaluation of event-based algorithms.

Chapter 3

Hardware:

3.1 Raspberry Pi 3

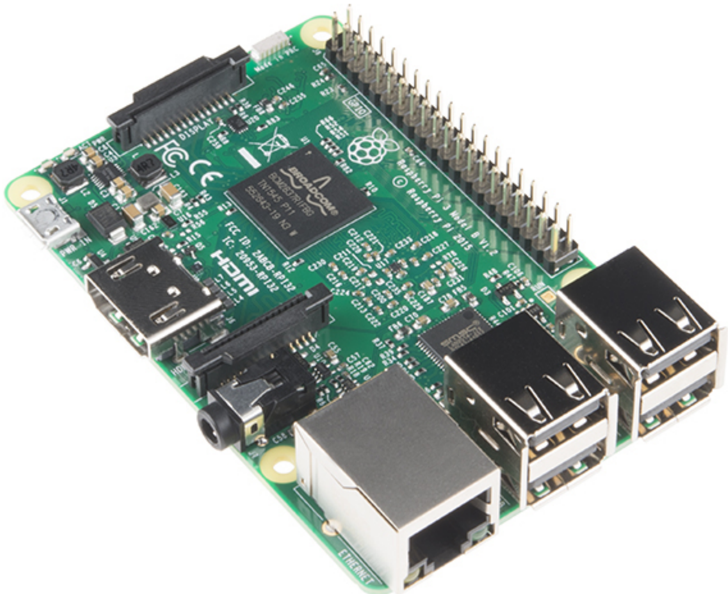

Figure 3.1: Raspberry Pi 3

The Raspberry Pi 3 is the third generation Raspberry Pi. This powerful credit-card sized single board computer can be used for many applications and supersedes the Raspberry Pi 2 Model. Whilst maintaining the popular board format the Raspberry Pi 3 Model B brings you a more powerful processer, 10x faster than the first generation Raspberry Pi. Additionally it adds wireless LAN Bluetooth connectivity making it the ideal solution for powerful connected designs. It has a HDMI (rev 1.3 1.4 Composite

RCA (PAL and NTSC).It uses Broadcom BCM2387 chipset with a 1.2GHz Quad-Core ARM Cortex-A53 and 802.11 b/g/n Wireless LAN and Bluetooth 4.1 (Bluetooth Classic and LE). It also has a 1GB LPDDR2 memory. Operating System Boots from Micro SD card, running a version of the Linux operating system or Windows 10 IoT.

3.2 ACS712 Module Current Sensor

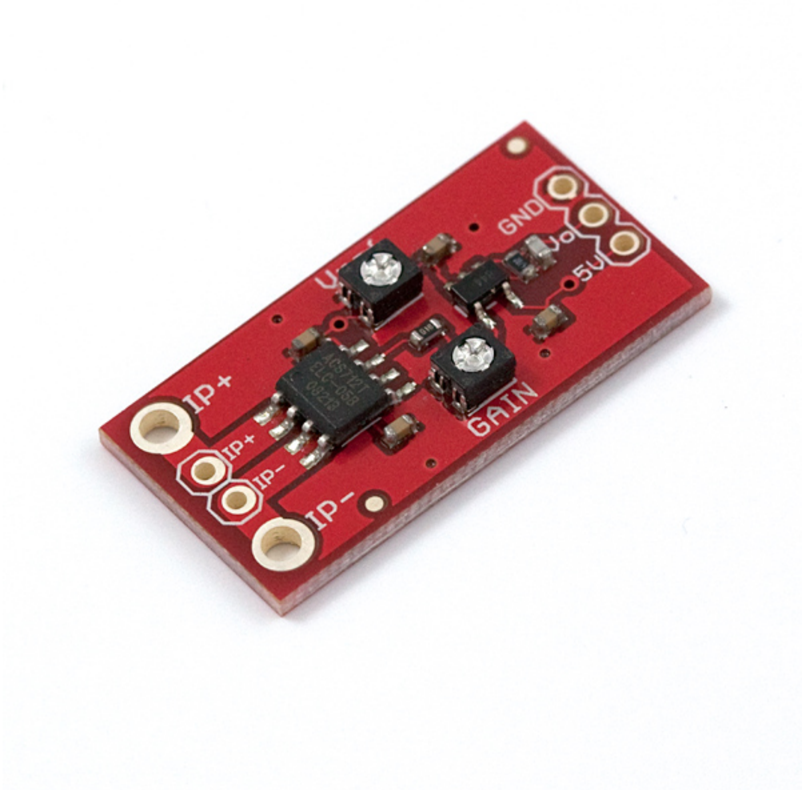

Figure 3.2: ACS712

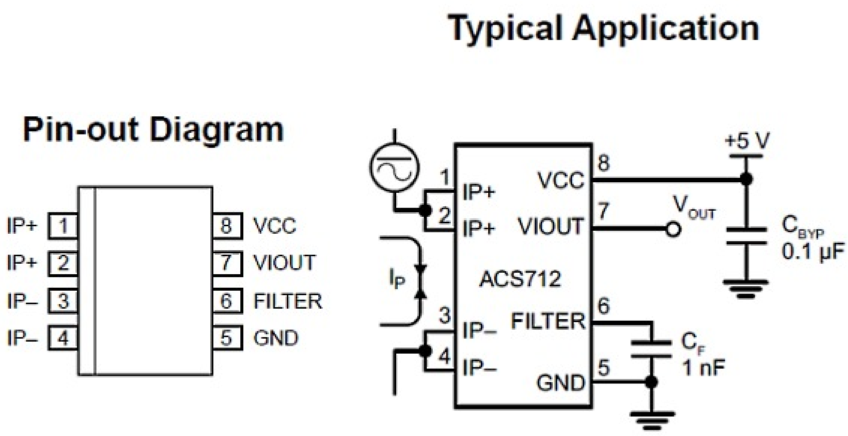

Figure 3.3: ACS712 Block Diagram

The Allegro ACS712 provides economical and precise solutions for AC or DC current sensing in industrial, commercial, and communications systems.The device consists of a precise, low-offset, linear Hall circuit with a copper conduction path located near the surface of the die. Applied current flowing through this copper conduction path generates a magnetic field which the Hall IC converts into a proportional voltage. Device accuracy is optimized through the close proximity of the magnetic signal to the Hall transducer. A precise, proportional voltage is provided by the low-offset, chopper-stabilized BiCMOS Hall IC, which is programmed for accuracy after packaging. The internal resistance of this conductive path is 1.2 mâĎę typical, providing low power loss. The thickness of the copper conductor allows survival of the device at up to 5ÃŮ overcurrent conditions. The terminals of the conductive path are electrically isolated from the signal leads (pins 5 through 8). This allows the ACS712 to be used in applications requiring electrical isolation without the use of opto-isolators or other costly isolation techniques. It requires a 4.5-5.5V power supply. It takes input current of 0-30A and produces output voltage of 2.5-5V.

3.3 ADC ADS1115

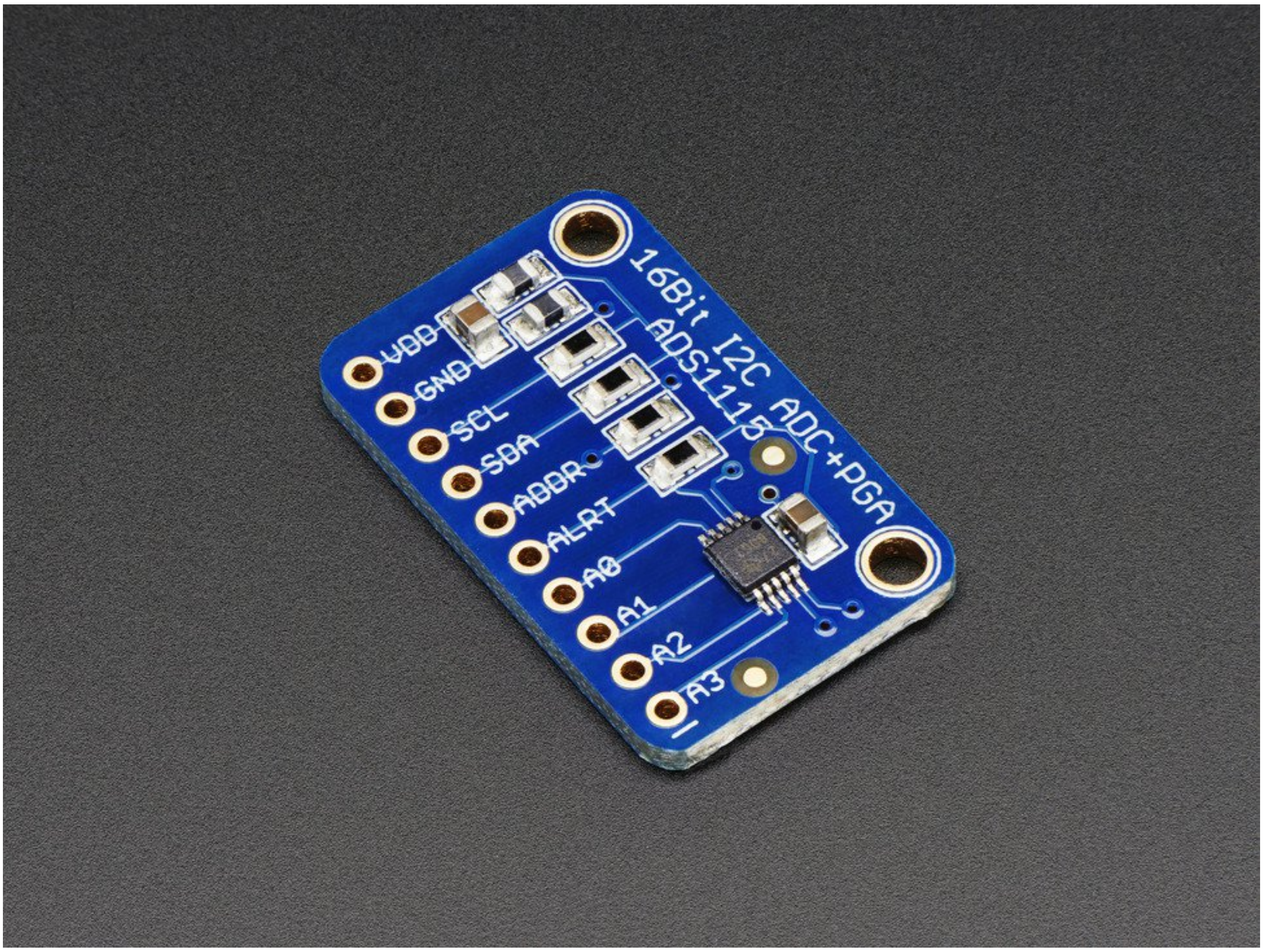

Figure 3.4: Raspberry Pi 3

For microcontrollers without an analog-to-digital converter or when you want a higher-precision ADC, the ADS1115 provides 16-bit precision at 860 samples/second over I2C. The chip can be configured as 4 single-ended input channels, or two differential channels. As a nice bonus, it even includes a programmable gain amplifier, up to x16, to help boost up smaller single/differential signals to the full range. We like this ADC because it can run from 2V to 5V power/logic, can measure a large range of signals and its super easy to use. It is a great general purpose 16 bit converter.
The chip's fairly small so it comes on a breakout board with ferrites to keep the AVDD and AGND quiet. Interfacing is done via I2C. The address can be changed to one of four options so you can have up to 4 ADS1115's connected on a single 2-wire I2C bus for 16 single ended inputs.

3.4 Current Transformer

Figure 3.5: Current Transformer

Current transformers can perform circuit control, measure current for power measurement and control, and perform roles for safety protection and current limiting. They can also cause circuit events to occur when the monitored current reaches a specified level. Current monitoring is necessary at frequencies from the 50 Hz/60 Hz power line to the higher frequencies of switchmode transformers that range into the hundreds of kilohertz.

The object with current transformers is to think in terms of current transformation rather than voltage ratios. Current ratios are the inverse of voltage ratios. The thing to remember about transformers is that Pout = (Pin âĂŤ transformer power losses). With this in mind, let's assume we had an ideal loss-less transformer in which Pout = Pin. Since power is voltage times current, this product must be the same on the output as it is on the input. This implies that a 1:10 step-up transformer with the voltage stepped up by a factor of 10 results in an output current reduced by a factor of 10. This is what happens on a current transformer. If a transformer had a one-turn primary and a ten-turn secondary, each amp in the primary results in 0.1A in the secondary, or a 10:1 current ratio. It's exactly the inverse of the voltage ratio âĂŤ preserving volt times current product.

How can we use this transformer and knowledge to produce something useful? Normally, an engineer wants to produce an output on the secondary proportional to the primary current. Quite often, this output is in volts output per amp of primary current. The device that monitors this output voltage can be calibrated to produce the desired results when the voltage reaches a specified level.

A burden resistor connected across the secondary produces an output voltage proportional to the resistor value, based on the amount of current flowing through it. With our 1:10 turns ratio transformer that produces a 10:1 current ratio, a burden resistor can be selected to produce the voltage we want. If 1A on the primary produces 0.1A on the secondary, then by Ohm's law, 0.1 times the burden resistor will result in an output voltage per amp.

Many voltage transformers have adjusted ratios that produce the desired output voltage and compensate for losses. The turns-ratios or actual turns aren't the primary concern of the end-user. Only the voltage output and possibly regulation and other loss parameters may be of concern. With current transformers, the user must know the current ratio to use the transformer. The knowledge of amps in per amps out is the basis for use of the current transformer. Quite often, the end users provide the primary with a wire through the center of the transformer. They must know what secondary turns are to determine what their output current will be. Generally, in catalogues, the turns of the transformers are provided as a specification for use.

With this knowledge, the user can choose the burden resistor to produce their desired output voltage. The output current of 0.1A for a 1A primary on the 1:10 turns ratio transformer will produce 0.1 V/A across a 1Îľ burden resistor, 1V per amp across a 10Îľ burden and 10V per amp across a 100Îľ burden resistor.

The Smart Energy Monitor uses a Current Transformer with a 1:1000 turns ratio a burden resistor of 40 Ohm.

3.5 Voltage Transformer

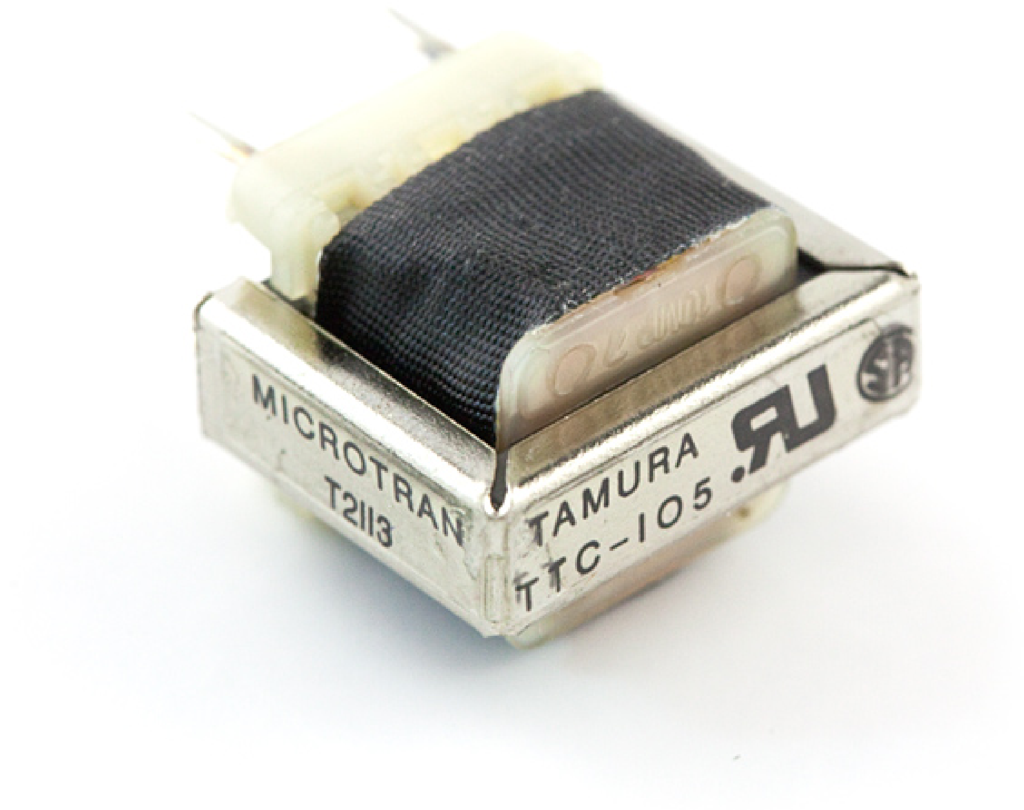

Figure 3.6: Voltage Transformer

Voltage transformers (VT), also called potential transformers (PT), are a parallel connected type of instrument transformer. They are designed to present negligible load to the supply being measured and have an accurate voltage ratio and phase relationship to enable accurate secondary connected metering. The Smart Energy Monitor uses a 230:5 turns ratio voltage transformer

3.6 Arduino

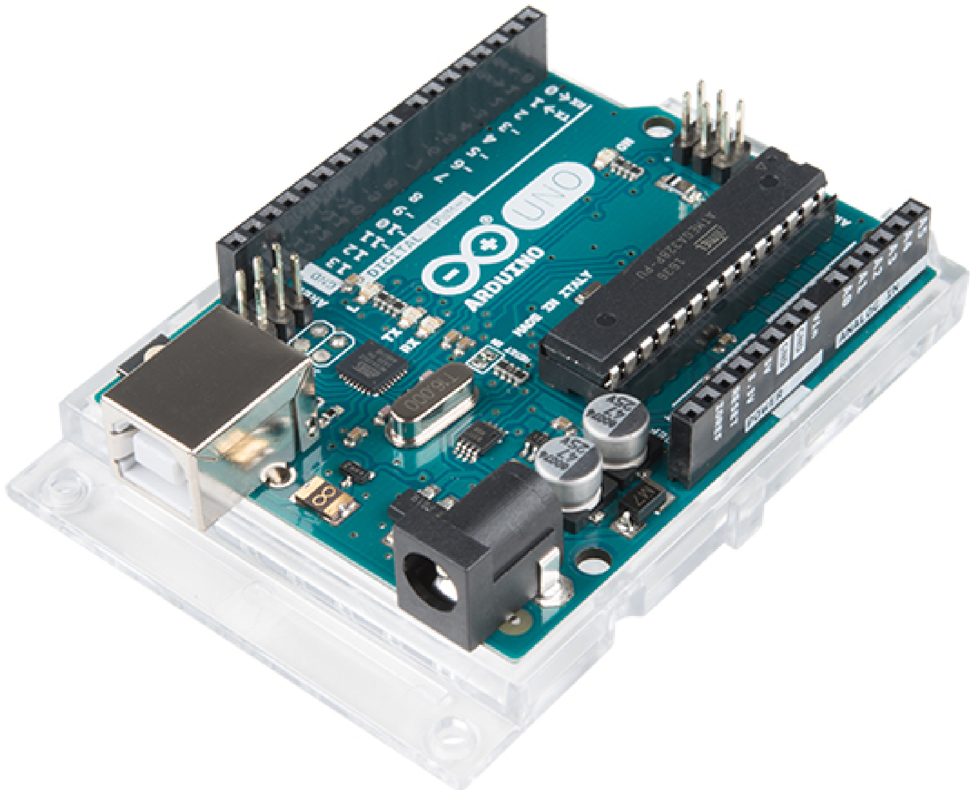

Figure 3.7: Arduino Uno

Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button.

3.7 Comparator LM339

The LM339 devices consist of four independent voltage comparators that are designed to operate from a single power supply over supplies also is possible, as long as the difference between the two supplies is 2 V to 36 V, and VCC isat least 1.5 V more positive than the input common-mode voltage. Current drain is independent of the supply voltage. The outputs can be connected to the other open-collector outputs to achieve wired-AND relationships

3.8 Operational Amplifier LM324

The LM324-N series consists of four independent, high-gain, internally frequency compensated operational amplifiers designed to operate from a single power supply over a wide range of voltages. Operation from split-power supplies is also possible and the low-power supply current drain is independent of the magnitude of the power supply voltage.

3.9 ExOR Gate 74HC86

The 74HC86 is a quad 2-input EXCLUSIVE-OR gate. Inputs include clamp diodes. This enables the use of current limiting resistors to interface inputs to voltages in excess of VCC.

Chapter 4

Design and Implementation:

4.1 Current Difference

Inorder to measure the AC Current using the current transformer we must:

1. Connect only the Live wire between the AC source and the load appliances through the current transformer.

2. Measure the peak voltage dropped across the 40 ohm burden resistor that is on the transformer output

3. Convert that voltage across the resistor to a current by applying ohms law ($I = V/R$).

4. Multiply the peak voltage by 0.707 to get and RMS Voltage across the resistor. (0.707 as a factor only applies to Sine Waves).

5. Multiply the RMS current by 1000 to get yield the value going through the wire being measured. (The current transformer has a 1000 to 1 ratio).

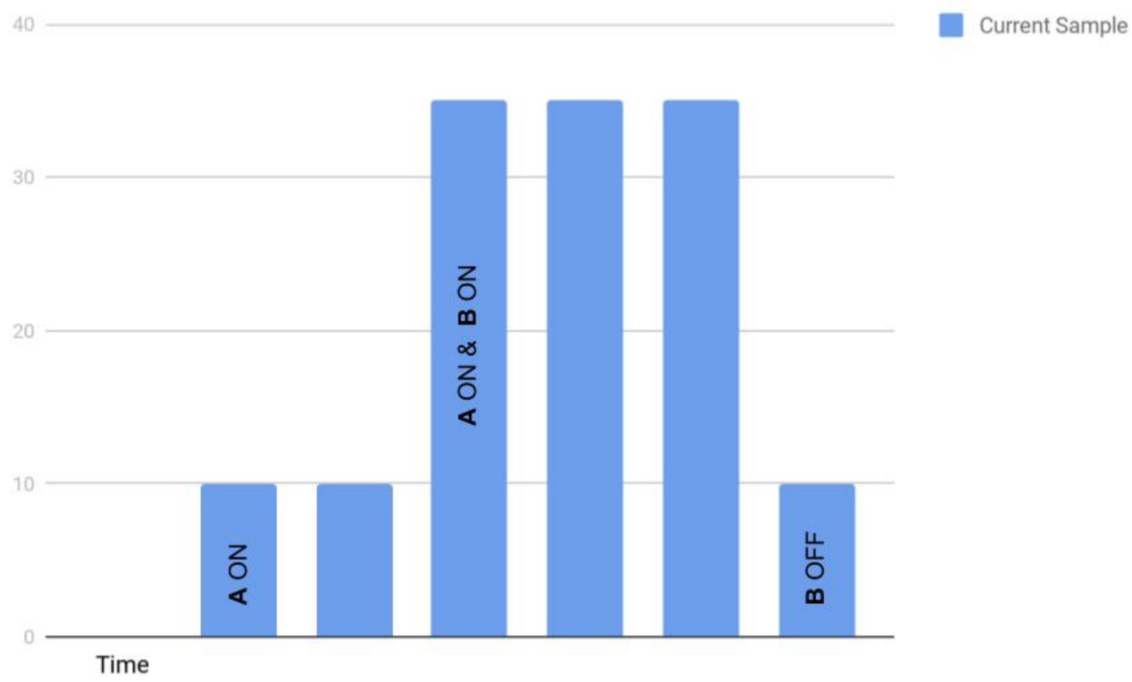

Figure 4.1: Change in Current between Conscutive Samples

The Current drawn by the load is the first parameter used to classify the various load appliances. The current transformer is used as a current sensor which will provide us with the instantaneous value of the total current drawn by all loads connected to the Smart Energy Monitor. In order to get the value of the current drawn by each individual load appliance we must calculate the difference between each of the consecutive current sample.

As shown in the above figure, we can see that when load appliance A is turned ON, the change in current is 10, when B is turned ON after A, the change in current is 25 but the total current is 35 and finally when B is turned OFF the change in current is 25 while the total current drawn is 10.

4.1.1 Averaging Consecutive Current Samples

The total change in current that occurs when a load appliance is switched ON/OFF may not be correctly reflected between the two consecutive samples. Instead the total

change in current may be reflected over more than two consecutive current samples depending on the switching speed or transient time of the load appliances. Hence, occasionally large errors are produced when measuring the current difference between any two consecutive current samples.
In order to reduce the error, the sampling rate may be adjusted accordingly. But calibrating the sampling rate may be difficult as various load appliances have different transient times as well as the time taken for different software instructions may differ with different hardware and sensors.

$$\mathrm{T}_{otal}C_{urrent} = \sum_0^N C_{urrent}S_{amples}$$

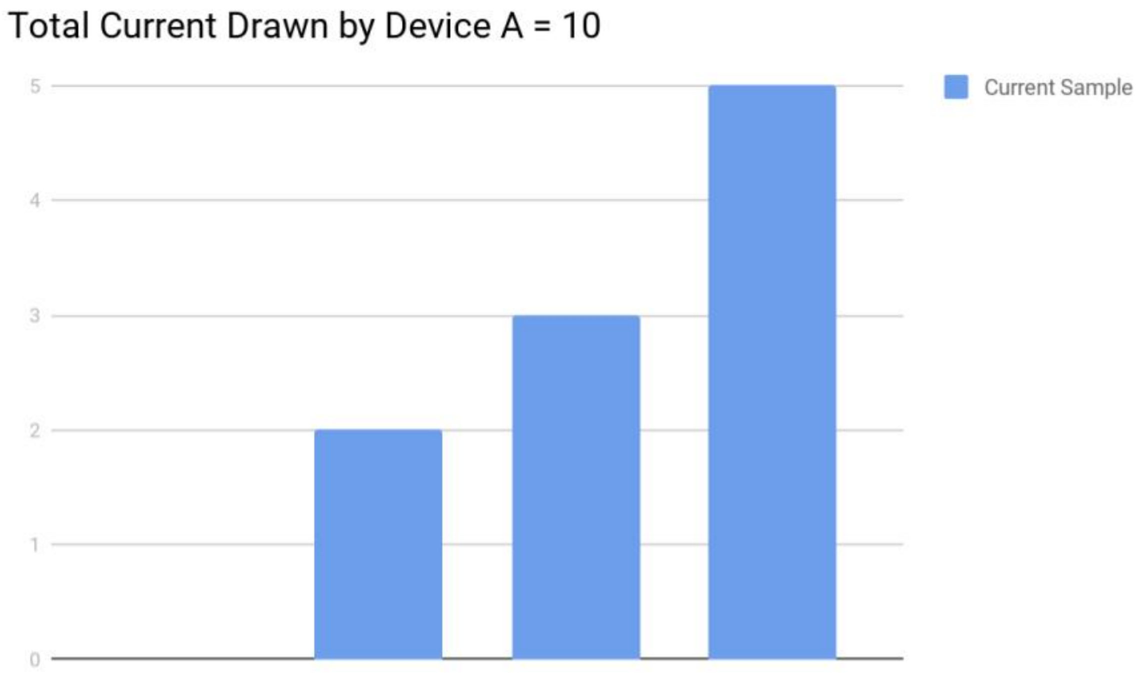

Figure 4.2: Sum of N Current Samples

A more efficient method for reducing this error will be to find the sum of N consecutive current samples such that the sum of all the error is zero(or close to zero). The user may have to wait for a very short duration before the next load appliance can be switched ON/OFF. The small error that remains will not come into effect as the classification algorithm will rule it out based on the mean and standard deviation values of all the labelled data.

As shown in the above figure, load appliance A during its transient state as a total current change of 10. But this value is not reflected between any two consecutive current samples. Instead the change in current observed between any two consecutive current samples for load appliance A is 2,3 and 5 respectively. Hence the value of error produced will be either 3 or 5. But if we take the sum of N consecutive samples where in N in this case is equal to 3, then we will get an error value of 0.

4.2 Active Power and Reactive Power

Using only the current drawn by a load as a classification parameter has the following limitation:

- If two or more completely different load appliances(different applications) draw the same amount of current, the current difference between N consecutive samples for both the devices may be equal which may produce an incorrect result.

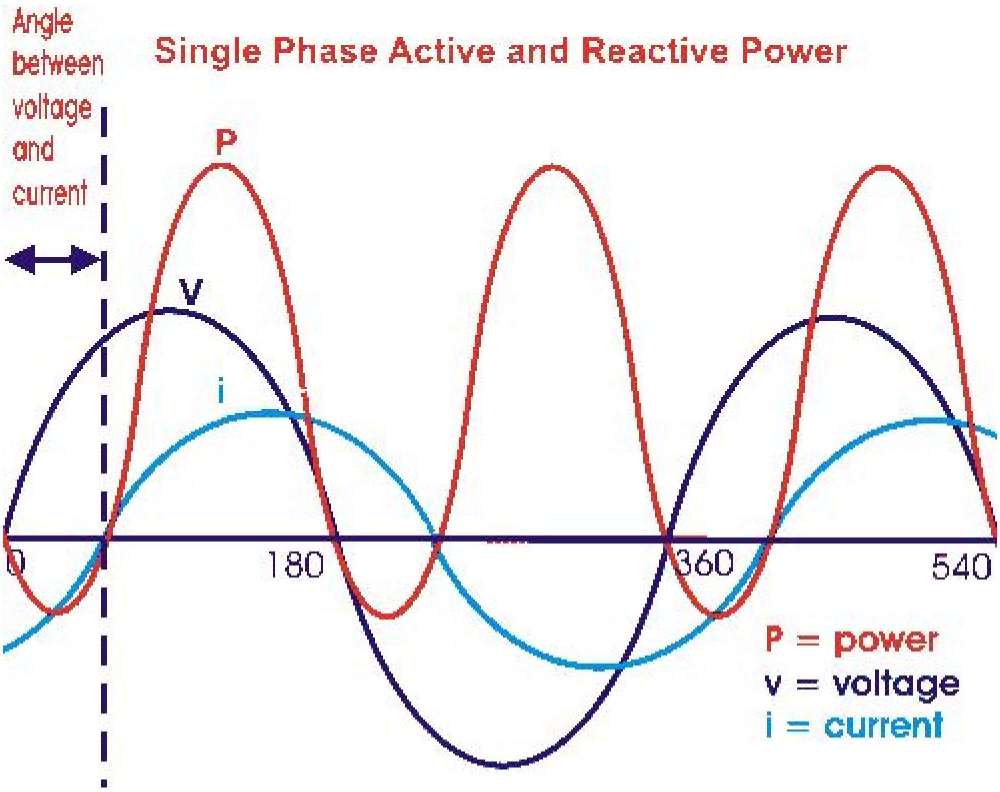

Figure 4.3: Phase Difference between Voltage and Current

Hence in order to more accurately differentiate between load appliances which draw the same amount of current, we also consider the value of active power and reactive power drawn by each individual load appliance. It will be highly unlikely that two completely different load appliances with different applications and uses have the same current drawn as well as Active Power and Reactive Power values.

4.3 Phase Angle

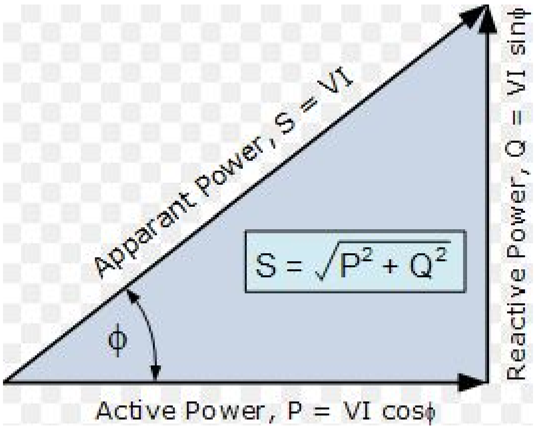

Figure 4.4: Power Triangle

$$Active\ Power = P = V_{RMS} I_{RMS} \cos\phi \tag{4.1}$$

$$Reactive\ Power = Q = V_{RMS} I_{RMS} \sin\phi \tag{4.2}$$

$$Apparent\ Power = S = \sqrt{P^2 + Q^2} \tag{4.3}$$

In order to calculate the Active Power and Reactive Power drawn by load appliances, we must first find the phase difference between the voltage and current. We do this by implementing a zero cross detector for both the voltage and current.

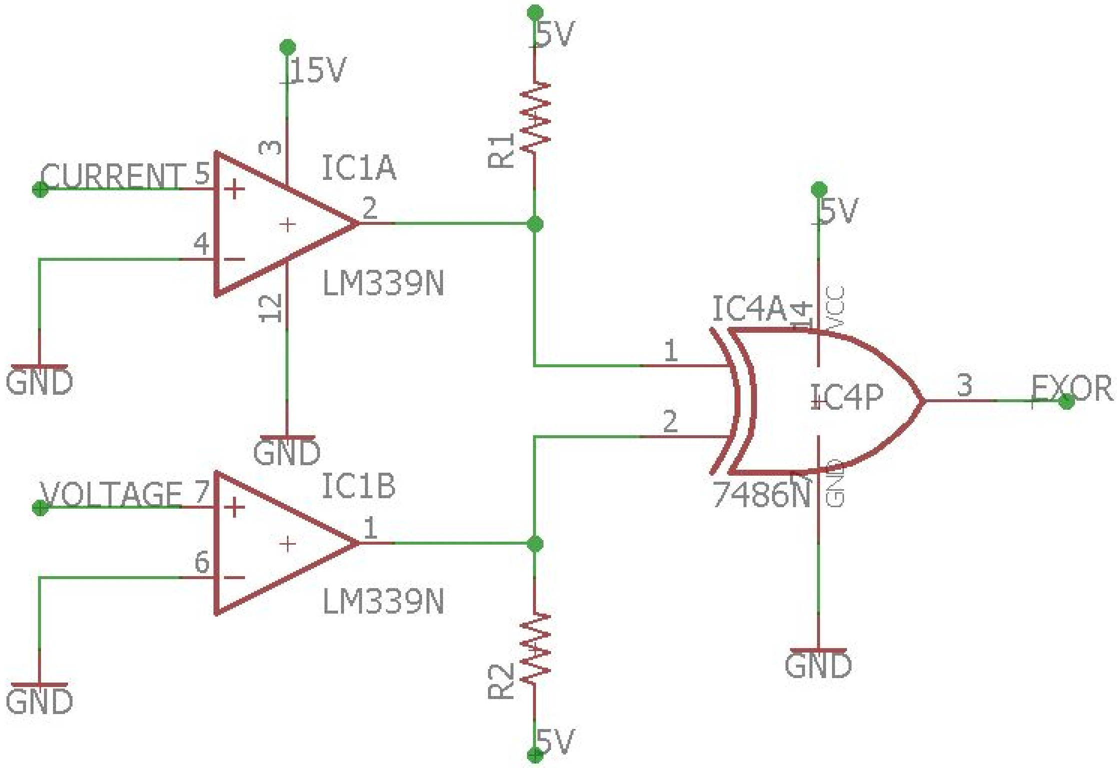

Figure 4.5: Zero Cross Detector

The zero cross detector is built using the LM339.

$$Y = (Voltage + Current) \cdot (\overline{Voltage} + \overline{Current}) \tag{4.4}$$

The output of the zero cross detectors is fed to the Ex-OR gate of the 7486 EXOR IC which will produce pulses when there is a phase shift. i.e. The two logic levels of the inputs to the EXOR gate are not equal to each other.

$$Phase\ Angle = \phi = 360 \times Frequency \times \triangle Time \tag{4.5}$$

The time period(t) of these pulses can be be used to find the phase angle between the voltage and current waveforms.

4.3.1 Current Amplification

The current transformer has a turns ratio of 1:1000. Hence if the current drawn by the load is around 1A, the output of the current transformer will be 1mA. Many load appliances draw currents far lower than 1A and hence the resulting voltage developed across the burden resistor connected to the current transformer may be lower than the input common mode voltage of the comparator.

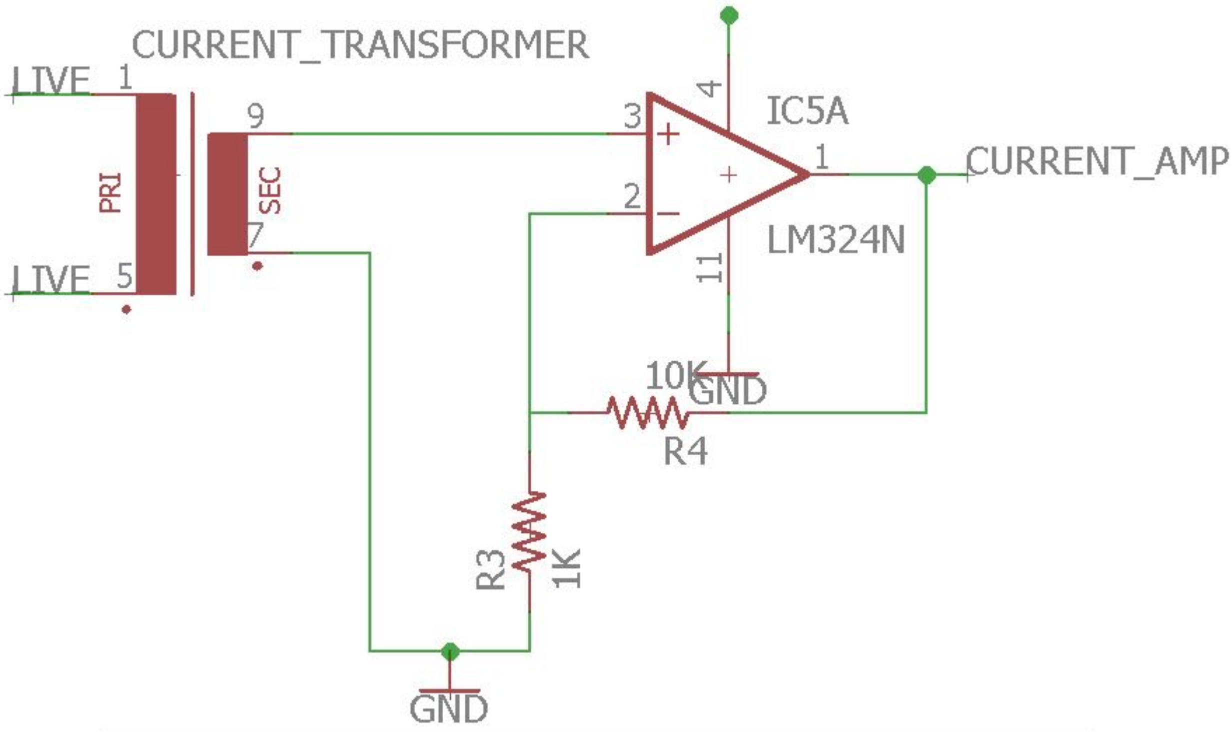

Figure 4.6: Current Signal Amplifier

In order to produce a voltage that is greater than the input common mode voltage of the comparator, output voltage across the burden resistor must be amplified. A non inverting amplifier must be used so that there is no effect on the values of phase angle between voltage and current.

$$Gain = 1 + \frac{R_f}{R_1} \tag{4.6}$$

The LM324 has an operational amplifier which can be configured as an Non Inverting amplifier with suitable gain for the purpose of amplifying the current signal.

4.4 Serial Communication

The Arduino is used as a low cost ADC which can transmit data to the Raspberry Pi. The Arduino transmits this data using Serial Communication through a USB cable. This is useful since no logic level conversion is required when compared to the I2C or GPIO communication protocols. Data such as the current drawn as well as Active Power and Reactive Power are sent to the Raspberry Pi. The Arduino is directly powered through its USB port. Hence no additional external power supply is required for the Arduino.

4.5 Naive Bayes Classification

The Naive Bayes algorithm is an intuitive method that uses the probabilities of each attribute belonging to each class to make a prediction. It is the supervised learning approach you would come up with if you wanted to model a predictive modeling problem probabilistically.
Naive bayes simplifies the calculation of probabilities by assuming that the probability of each attribute belonging to a given class value is independent of all other attributes. This is a strong assumption but results in a fast and effective method.
The probability of a class value given a value of an attribute is called the conditional probability. By multiplying the conditional probabilities together for each attribute for a given class value, we have a probability of a data instance belonging to that class.
To make a prediction we can calculate probabilities of the instance belonging to each class and select the class value with the highest probability.
Naive bases is often described using categorical data because it is easy to describe and calculate using ratios. A more useful version of the algorithm for our purposes supports numeric attributes and assumes the values of each numerical attribute are normally distributed (fall somewhere on a bell curve). Again, this is a strong assumption, but still gives robust results.

4.5.1 Creating the Dataset

The dataset is created by measuring the values of the current drawn, active power and reacive power drawn by individual load appliances. These data points are then used to identify the device. The change in these parameters must follow the following condition to be considered as valid data point:

- The increase in any classification parameter(current drawn, active power and reactive power drawn) when a load appliance is turned ON must be equal to decrease in that classification parameter when the load appliance is turned OFF.
- The values of the data points of a particular load appliance must remain the same even if other devices are active. This must apply for all combinations of load appliances and their individual states.

The valid data points have to be labelled according to the load appliance state. The data is stored in a standard CSV format table. The more validated labelled data within the dataset will make the predicted result more accurate.

4.5.2 Impelementation of Classification Algorithm

The implementation of most classification algorithms involve the following processes:

1. **Handle Data:** Load the data from CSV file and split it into training and test datasets. Split the data into a training dataset that Naive Bayes can use to make predictions and a test dataset that we can use to evaluate the accuracy of the model.

$$Training\ Data = 0.67 \times Dataset\ Size \tag{4.7}$$

$$Testing\ Data = 0.33 \times Dataset\ Size \tag{4.8}$$

   We need to split the data set randomly into train and datasets with a ratio of 67 percent train and 33 percent test (this is a common ratio for testing an algorithm on a dataset).

2. **Summarize Data:** Summarize the properties in the training dataset so that we can calculate probabilities and make predictions.

3. **Make a Prediction:** Use the summaries of the dataset to generate a single prediction.

4. **Make Test Data Predictions:** Generate predictions given a test dataset and a summarized training dataset.

5. **Evaluate Accuracy:** Evaluate the accuracy of predictions made for a test dataset as the percentage correct out of all predictions made.

6. **Tie it Together:** Use all of the code elements to present a complete and standalone implementation of the Naive Bayes algorithm.

4.5.3 Data Summarizing

The naive bayes model is comprised of a summary of the data in the training dataset. This summary is then used when making predictions.

The summary of the training data collected involves the mean and the standard deviation for each attribute, by class value.

These are required when making predictions to calculate the probability of specific attribute values belonging to each class value.
We can break the preparation of this summary data down into the following sub-tasks:

1. **Separate Data By Class:** The first task is to separate the training dataset instances by class value so that we can calculate statistics for each class. We can do that by creating a map of each class value to a list of instances that belong to that class and sort the entire dataset of instances into the appropriate lists.

2. **Calculate Mean:** We need to calculate the mean of each attribute for a class value.

$$Mean\ of\ each\ Attribute\ for\ Class\ Value = \frac{Sum\ of\ all\ Attributes}{Total\ Number\ of\ Attributes} \tag{4.9}$$

   The mean is the central middle or central tendency of the data, and we will use it as the middle of our gaussian distribution when calculating probabilities.

   We also need to calculate the standard deviation of each attribute for a class value. The standard deviation describes the variation of spread of the data, and we will use it to characterize the expected spread of each attribute in our Gaussian distribution when calculating probabilities.

$$Standard\ Deviation\ of\ each\ Attribute\ for\ Class\ Value = \sigma = \sqrt{\frac{\sum_0^n (x - \overline{Mean})^2}{n}} \tag{4.10}$$

   In the above equation n = The number of attributes. The standard deviation is calculated as the square root of the variance. The variance is calculated as the average of the squared differences for each attribute value from the mean.

3. **Summarize Dataset:** Now we have the tools to summarize a dataset. For a given list of instances (for a class value) we can calculate the mean and the standard deviation for each attribute.

4. **Summarize Dataset:** Now we have the tools to summarize a dataset. For a given list of instances (for a class value) we can calculate the mean and the standard deviation for each attribute.

5. **Summarize Attributes By Class:** We can pull it all together by first separating our training dataset into instances grouped by class. Then calculate the summaries for each attribute.

4.5.4 Making Predictions

We are now ready to make predictions using the summaries prepared from our training data. Making predictions involves calculating the probability that a given data instance belongs to each class, then selecting the class with the largest probability as the prediction.
We can divide this part into the following tasks:

1. **Calculate Gaussian Probability Density Function:** We can use a Gaussian function to estimate the probability of a given attribute value, given the known mean and standard deviation for the attribute estimated from the training data.

$$Gaussian\ Probability = \frac{1}{\sqrt{2\pi\sigma^2}} \times Exponent \tag{4.11}$$

$$Exponent = e^{-\frac{(x-Mean)^2}{2\sigma^2}} \tag{4.12}$$

   Given that the attribute summaries where prepared for each attribute and class value, the result is the conditional probability of a given attribute value given a class value.

2. **Calculate Class Probabilities:** Now that we can calculate the probability of an attribute belonging to a class, we can combine the probabilities of all of the attribute values for a data instance and come up with a probability of the entire data instance belonging to the class.

   We combine probabilities together by multiplying them.

3. **Making a Prediction:** Now that can calculate the probability of a data instance belonging to each class value, we can look for the largest probability and return the associated class. Finally, we can estimate the accuracy of the model by making predictions for each data instance in our test dataset. This will return a list of predictions for each test instance.

4. **Estimating Accuracy:**

$$Accuracy = \frac{Number\ of\ Accurate\ Predictions}{Size\ of\ the\ Testing\ Dataset} \times 100 \tag{4.13}$$

The predictions can be compared to the class values in the test dataset and a classification accuracy can be calculated as an accuracy ratio between 0 and 100

4.6 Additional Classification Parameters

The more number of parameters used in classification algorithms will further improve the quality of the dataset.

These additional parameters are non numerical values. Hence a decision tree algorithm along with a Naive Bayes classification must be implemented when considering any non numerical parameters.

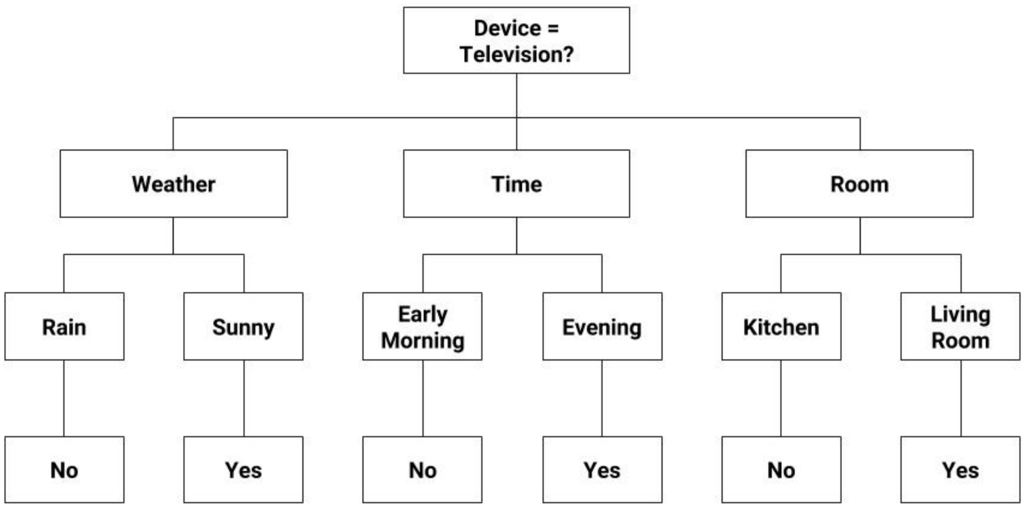

Figure 4.7: Electricity Tariff

Hence in addition to the current, active power and reactive power classification parameters, we propose the use of additional parameters such as:

4.6.1 Weather

The current weather may determine the state of the device. For example, when the temperature is low and the residents begin to feel cold during the winter, the likelihood of the Air Conditioner being on is highly unlikely. The weather data will have to be calibrated according to a particular geographic location

4.6.2 Time Day

The time of the day can also be used to determine the state of the device. For example, during the day and during the night(sleep time 1 A.M. to 6 A.M.), the likelihood of the use of high power lighting is highly unlikely.

In some cases the day of the week can be used as an additional classification parameter. For example, during weekends, more people are present at home(weekdays spent at Work,college,etc), hence the likelihood of the use of entertainment systems such as Television and Gaming systems during the midday as well as the likelihood of the use of kitchen appliances during the midday will significantly increase.

This may vary from household to household but with a large dataset of mixed households, this parameter may become very useful.

The time day data will have to be calibrated according to a particular timezone.

4.6.3 Room Location

Certain load appliances have fixed places within an household. By creating another parameter which takes into account the location of a load appliance within the household can also improve the quality of the dataset. For example, most households have refrigerators microwaves in the kitchen, water heating geysers in the bathroom, televisions home theatre systems in the living room. On the other hand, the likelihood of an Air Conditioner within a bathroom will be highly unlikely.

This will indeed require additional hardware and separate current sensors per room.

4.7 Voltage Calibration

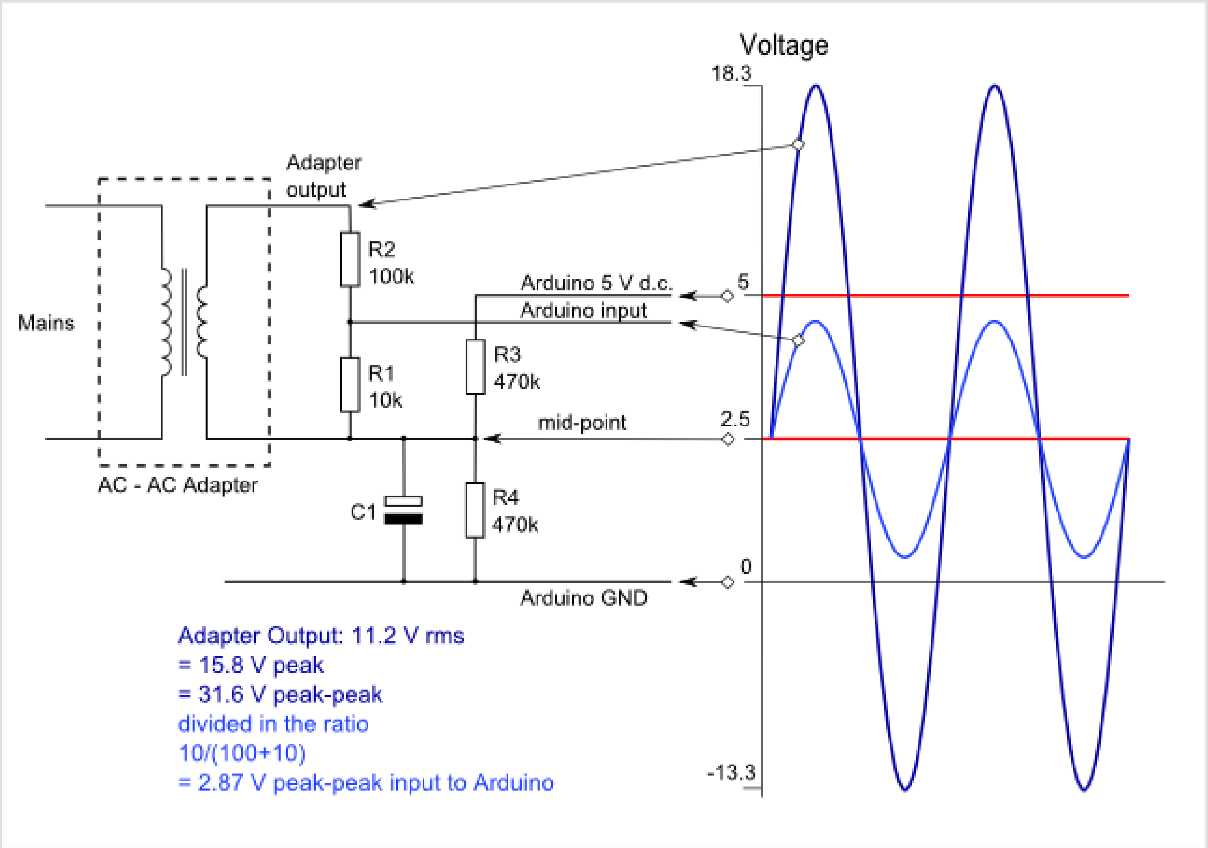

Figure 4.8: Voltage Calibration

$$Peak\ Output\ Voltage = \frac{R_1}{R_1 + R_2} \times Peak\ Input\ Voltage \tag{4.14}$$

The standard household AC voltage provided by electricity distributors varies from 220V to 240V. Hence the voltage value will vary depending on the location, electricity distributor and even in some cases voltage fluctuations. In order taken into account such voltage variations, the voltage transformer is used to calibrate the household voltage once the Smart Energy Monitor has been installed.

4.8 Estimating Electricity Bill

Categories	Tata Power	Reliance Infra	BEST	MSEDCL
Low - End Residential				
0 - 100 units	2.49	3.92	3.2	3.36
101 - 300 units	4.13	5.56	6.38	6.05
High - End Residential				
301 - 500 units	7.31	6.46	8.94	7.92
Above 500	9.09	8.77	11.4	8.78
HT Industrial	8.02	7.50	9.59	7.01
HT Commercial	8.31	9.23	10.31	10.45

Figure 4.9: Electricity Tariff

$$Monthly\ Electricity\ Bill = Number\ of\ Units\ Consumed \times Cost\ per\ Unit \qquad (4.15)$$

By calculating the total number of units (1KWh = 1unit) consumed per month we can estimate the electricity bill for that month. The cost per unit of electricity will vary based on the electricity distributor as well as the number of units consumed.

4.9 User Interface

4.9.1 Flask Microframework

The user interface uses a Python Micro-Framework called Flask which is used to build web applications.

The Flask framework encodes the real time Python Variables into a JavaScript Object Notation(JSON) string.

The JSON string is then read into the HTML file using Asynchronous JavaScript And XML(AJAX) requests.

These requests are periodically refreshed using the Auto-Refresh function in AJAX.

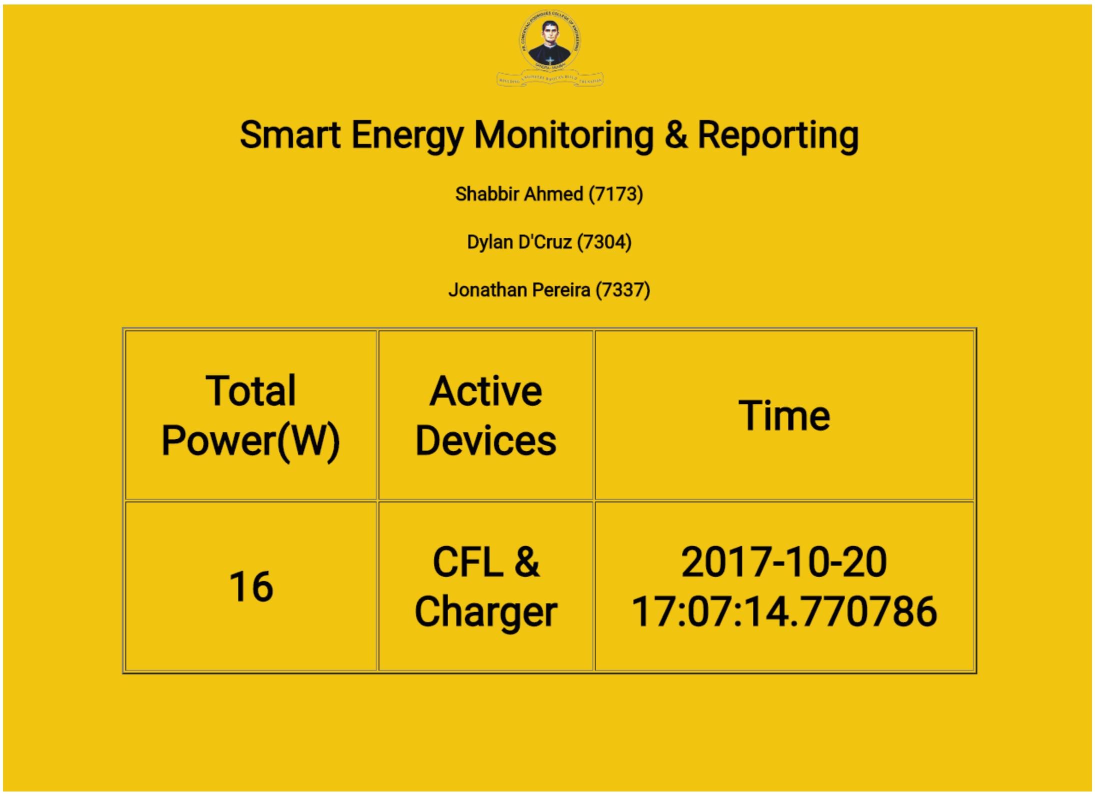

Figure 4.10: HTML-Based UI

The user interface displays various system parameters such as Total Power being Consumed, Active Devices and the Real Time Clock. Additional parameters such as Estimated Monthly Electricity bill can be added in the future.

4.9.2 Dash Framework

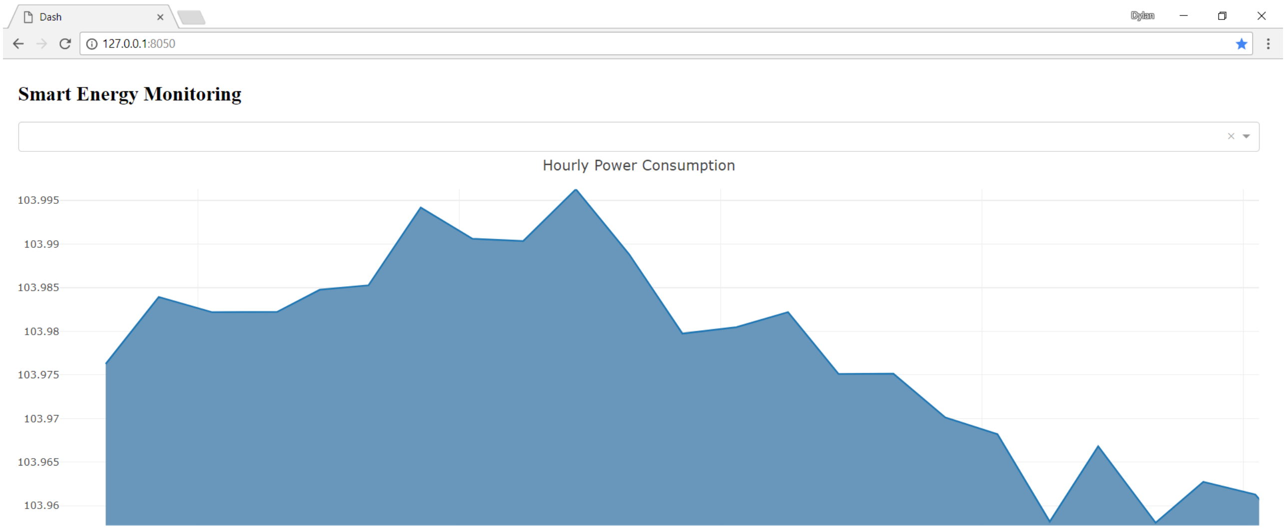

Figure 4.11: Hourly Power Consumption over a 24hr period cycle using Dash

Dash is a Python framework for building analytical web applications. There is no JavaScript required. Built on top of Plotly.js, React, and Flask, Dash ties modern UI elements like dropdowns, sliders, and graphs to your analytical Python code.
We plan on using Dash to plot various live graphs as well as a pie chart of the energy usage of all load appliances.

Chapter 5

Results

5.1 Observations

We successfully measure changes in the current drawn by the load appliances when their states change. By measuring the Phase Angle we can get Active Power and Reactive Power which further improves the accuracy of the system. We created an dataset which contains labelled data which is passed to our classification algorithm. The Naive Bayes algorithm uses the mean standard deviation of the input parameters to determine the active devices.

5.2 Drawback

The user must wait for a period of 0.5s to 1s before turning ON/OFF any load appliance. The Smart Energy Monitor is not a plug and play device. Since it deals with high levels of voltage current, it must be installed within the household by a professional electrician.

5.3 Future Scope

1. The classification accuracy can be improved through pattern analysis by monitoring the shape of the VI trajectory. Shape features such as area under the VI curve as well as peak of segments can be further analyzed.

2. Analysis of Steady State Voltage Noise such as EMI signatures can improve the detection of motor based devices like Fans, Food Mixers and Washing Machines.

3. Unsupervised learning methods such as Artificial Neural Networks(ANN) and Hidden Markov Model(HMM) have shown to perform well for the task of load

disaggregation due to their ability to incorporate in their learning, temporal as well as appliance state transition information.

Chapter 6

Conclusion

The system is able to successfully disaggregate the power signals and classify the active devices. By calculating the active time for individual load appliances, we can track the total power consumed by that device. This information can then be used to estimate the monthly electricity bill. It can also provide the user with detailed insights as to which devices are not energy efficient can help the user track their energy usage.

Bibliography

[1] Nipun Batra, Jack Kelly, Oliver Parson, Haimonti Dutta, William Knottenbelt, Alex Rogers, Amarjeet Singh and Mani Srivastava *NILMTK: An Open Source Toolkit for Non-intrusive Load Monitoring.*

[2] Ahmed Zoha, Alexander Gluhak, Muhammad Ali Imran, Sutharshan Rajasegarar *Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing: A Survey.*
`www.mdpi.com/1424-8220/12/12/16838/pdf`

[3] Henry's Bench *How to Measure AC Current with an Arduino and an ASC712.*
`http://henrysbench.capnfatz.com/henrys-bench/arduino-current-measurements/acs712`

[4] Watty - *A simple way to keep track of what goes on at home.*
`https://watty.io/`