Server Based Load Analysis of Smart Meter System

Mr. S. Kranthi Kumar[1] Siva Shankar Mahanty[2], Ratnala Jyotshna[3], Jaswanth Pudi[4] Sai Kumar Tammireddi[5]

[1]*Associate Professor, Department of Electrical & Electronics Engineering, Raghu Engineering College, Visakhapatnam.*

[2]*Student, Department of Electrical & Electronics Engineering, Raghu Engineering College, Visakhapatnam.*

[3]*Student, Department of Electrical & Electronics Engineering, Raghu Engineering College, Visakhapatnam.*

[4]*Student, Department of Electrical & Electronics Engineering, Raghu Engineering College, Visakhapatnam.*

[5]*Student, Department of Electrical & Electronics Engineering, Raghu Engineering College, Visakhapatnam.*

---***---

Abstract - The electricity demand is adding with the growth of population and with the use of different appliances in the homes. So, there's a need for consumers to track their diurnal operation and understand the consumption patterns to save and control these coffers. Smart cadence along with Advanced Metering structure(AMI) is a realistic and effective result for this. before procedure which put to profitable use of one- way dispatches to gather cadence data, were mentioned to as Automated Meter Reading(AMR) Systems. This design aims at analyzing the performance of the proposed smart cadence systems, effective transmission and how serviceability explore new developments for the benefit of consumers as well as themselves by ever covering energy consumption. This is achieved by using PLC modems for remote monitoring and control of energy measures. By this way we can bring down mortal sweats demanded to outline cadence readings which are till now recorded by visiting every home collectively. As a result, the consumption patterns at the serviceability are studies and cargo analysis is made so that this can help in maintaining other systems associated with energy operation. To study and dissect the cargo consumption patterns. By this way an estimate on the energy consumption can be made and therefore have a control on its operation.

***Key Words*:** Smart Meter; Smart Grid; Advanced Metering Infrastructure(AMI); Information Rights Management(IRM); Power line communication(PLC); Arduino UNO;Energy consumption.

1.INTRODUCTION

The energy extremity around the world stems from high consumptive patterns in the domestic and artificial sectors. Due to growing energy demands, the grid stability becomes a great challenge. With rising electricity costs, it's important that homes and diligence cover the energy operation to cut down the energy operation in all possible ways to save costs. A smart energy cadence is a developing conception which give the capability to track and control energy consumption. piecemeal from tracking the energy consumption, smart energy measures record a variety of data points on consumption which will help to dissect the consumption pattern to read the future energy demand. The first ever smart energy cadence was constructed by Thomas Alva Edison(1847- 1931). As a colonist and innovator, Edison believed that electricity should be vended as a commodity just like gas. thus, he developed the world's first electrical cadence(T.A.Edison, 1881). The technology has fleetly bettered with the invention ofmicro-controllers which paved the way for digital slice(Bulbenkiene V etal., 2011, Omijeh BO etal., 2013). The digital slice fashion was the first step advancement towards the present- day smart metering system. There have been developments in smart energy metering systems designed for domestic operations, using Arduinomicro-controller system and Ethernet communication system(Aswathy and Shanthi, 2013). Reducing the specialized andnon-technical losses of electricity in smart energy metering system was bandied(Ilieet.al, 2020). The new algorithm proposed in this paper identifies the distribution bumps with energy losses through the smart metering system. The error performance analysis of 400V distribution grid using PLC technology for smart metering systems was bandied(Dzemo etal., 2020).

2. MODELLING AND HARDWARE DESIGNING

2.1ESP32 (esp32) Module Interaction:

Today, people need Mobile Phones for many things like talking, internet, multimedia etc. All these services must be made available to the user on the go i.e. while the user is mobile. With the help of these wireless communication services, we can transfer voice, data, videos, images etc.

Wireless Communication Systems also provide different services like video conferencing, cellular telephone, paging, TV, Radio etc. Due to the need for variety of communication services, different types of Wireless Communication Systems are developed. Some of the important Wireless Communication Systems available today are:

- Television and Radio Broadcasting
- Satellite Communication
- Radar
- Mobile Telephone System (Cellular Communication)
- Global Positioning System (GPS)
- Infrared Communication
- WLAN (Wi-Fi)
- Bluetooth
- Paging
- Cordless Phones
- Radio Frequency Identification (RFID)

There are many other system with each being useful for different applications. Wireless Communication systems can be again classified as Simplex, Half Duplex and Full Duplex. Simplex communication is one way communication. An example is Radio broadcast system.

Half Duplex is two way communication but not simultaneous one. An example is walkie – talkie (civilian band radio). Full Duplex is also two way communication and it is a simultaneous one. Best example for full duplex is mobile phones.

The devices used for Wireless Communication may vary from one service to other and they may have different size, shape, data throughput and cost. The area covered by a Wireless Communication system is also an important factor. The wireless networks may be limited to a building, an office campus, a city, a small regional area (greater than a city) or might have global coverage.

2.2. Features:

- Processors:
- CPU: Xtensa dual-core (or single-core) 32-bit LX6 microprocessor, operating at 160 or 240 MHz and performing at up to 600 DMIPS
- Ultra low power (ULP) co-processor
- Memory: 520 KiB SRAM&Wireless connectivity:
- Wi-Fi: 802.11 b/g/n
- Bluetooth: v4.2 BR/EDR and BLE (shares the radio with Wi-Fi)
- Peripheral interfaces:
- 12-bit SAR ADC up to 18 channels
- 2 × 8-bit DACs
- 10 × touch sensors (capacitive sensing GPIOs)
- 4 × SPI
- 2 × I²S interfaces
- 2 × I²C interfaces
- 3 × UART
- SD/SDIO/CE-ATA/MMC/eMMC host controller
- SDIO/SPI slave controller
- Ethernet MAC interface with dedicated DMA and IEEE 1588 Precision Time Protocol support
- CAN bus 2.0
- Infrared remote controller (TX/RX, up to 8 channels)
- Motor PWM
- LED PWM (up to 16 channels)
- Hall effect sensor
- Ultra low power analog pre-amplifier
- Security:
- IEEE 802.11 standard security features all supported, including WFA, WPA/WPA2 and WAPI
- Secure boot
- Flash encryption
- 1024-bit OTP, up to 768-bit for customers
- Cryptographic hardware acceleration: AES, SHA-2, RSA,elliptic curve cryptography(ECC), random number generator (RNG)
- Power management:
- Internal low-dropout regulator
- Individual power domain for RTC
- 5 μA deep sleep current

2.3 ESP32-Architecture:

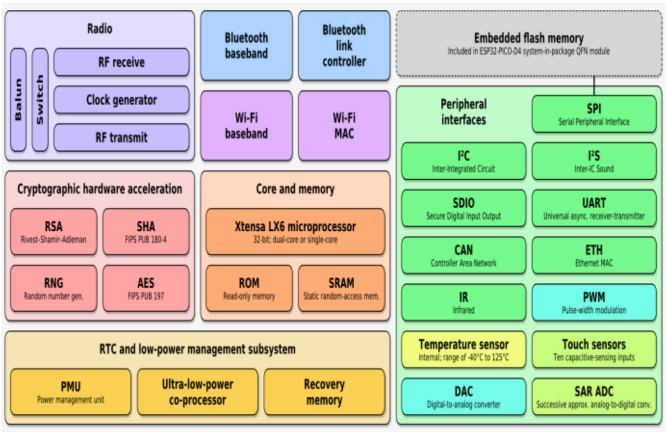


Fig-2.3 ESP32- Architecture

2.4 ESP32 Architecture-Pin Diagram:

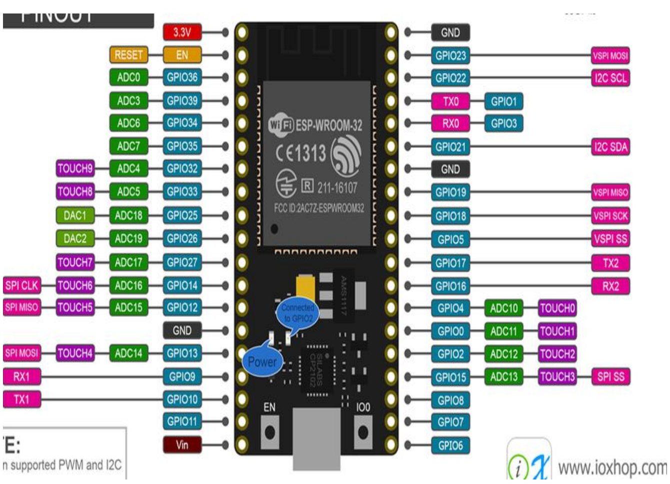
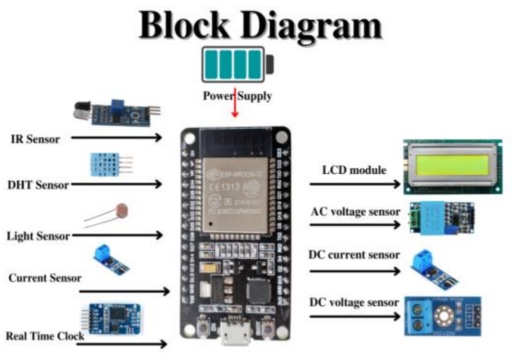
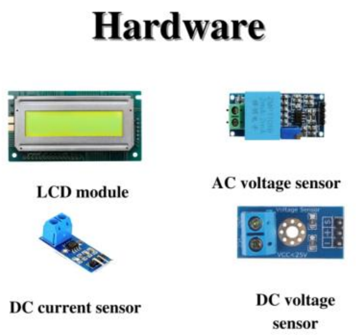
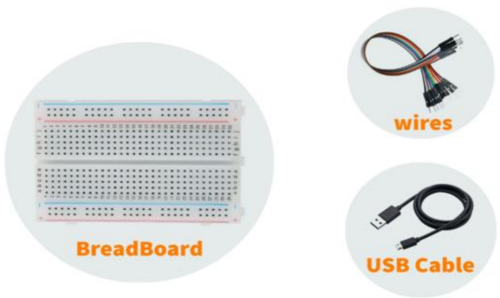


Fig-2.4 ESP32– Pin Diagram

3. HARDWARE IMPLEMENTATION AND RESULTS







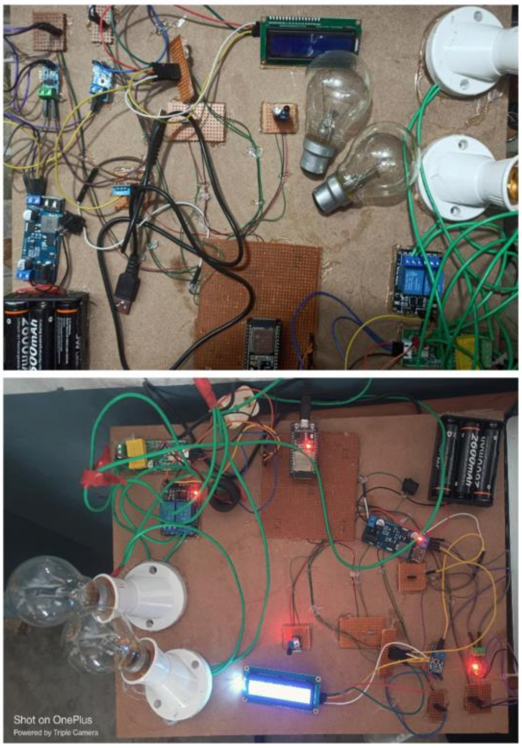
CODE:

```
#include<WiFi.h>
#include <HTTPClient.h>
#include<WiFiClient.h>
#include <PZEM004Tv30.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C  lcd(0x27,16,2);
#include <DHT.h>
#define DHTPIN 4
#define DHTTYPE DHT11
DHT dht(DHTPIN,DHTTYPE);
#if !defined(PZEM_RX_PIN) && !defined(PZEM_TX_PIN)
#define PZEM_RX_PIN 16
#define PZEM_TX_PIN 17
#endif
#if !defined(PZEM_SERIAL)
#define PZEM_SERIAL Serial2
#endif
#if defined(ESP32)
 *  ESP32 initialization
 * The ESP32 HW Serial interface can be routed to any GPIO pin
 * Here we initialize the PZEM on Serial2 with RX/TX pins 16 and 17
PZEM004Tv30 pzem(PZEM_SERIAL, PZEM_RX_PIN, PZEM_TX_PIN);
#elif defined(ESP8266)
 *  ESP8266 initialization
 * Not all Arduino boards come with multiple HW Serial ports.
 * Serial2 is for example available on the Arduino MEGA 2560 but not Arduino Uno!
 * The ESP32 HW Serial interface can be routed to any GPIO pin
 * Here we initialize the PZEM on Serial2 with default pins
//PZEM004Tv30 pzem(Serial1);
#else
 *  Arduino initialization
  * Not all Arduino boards come with multiple HW Serial ports.
 * Serial2 is for example available on the Arduino MEGA 2560 but not Arduino Uno!
 * The ESP32 HW Serial interface can be routed to any GPIO pin
 * Here we initialize the PZEM on Serial2 with default pins
PZEM004Tv30 pzem(PZEM_SERIAL);
#endif
WiFiClient client;
String tsAddress1="http://maker.ifttt.com/trigger/sendMail/json";
String tsAddress="http://api.thingspeak.com/update.json?";
String url;
String url1;
```

```
HTTPClient http;
HTTPClient http1;
 float energy;
 float price;
void setup() {
   // Debugging Serial port
   Serial.begin(115200);
// Initialize the LCD connected
  lcd.init ();
  // Turn on the backlight on LCD.
  lcd. backlight ();
  // put your setup code here, to run once:
  pinMode(34,INPUT);//LDR PIN
   pinMode(5,INPUT);//IR sensor PIN
  pinMode(18, OUTPUT); // Relay Bulb pin
  pinMode(19, OUTPUT); //Relay Bulb pin
pinMode(35,INPUT);//DC Voltage PIN
pinMode(33,INPUT);//DC Current PIN
dht.begin();
WiFi.begin("APPLETON","Christi@123i");// Provide Wifi Name and Password
while((!(WiFi.status()==WL_CONNECTED))){
delay(300);
Serial.println("Wifi Connecting..");
   // Uncomment in order to reset the internal energy counter
   // pzem.resetEnergy()}}
void loop() {
int x=analogRead(34);
  Serial.println(x);
   float v=3.3*x/4095;
  Serial.println(v);
  //delay(1000);
  float t=dht.readTemperature();
   Serial.println("temp");
   Serial.println(t);
  float h=dht.readHumidity();
   Serial.println("humid");
    Serial.println(h);
  //  delay(1000);
   // Print the custom address of the PZEM
    Serial.print("Custom Address:");
    Serial.println(pzem.readAddress(), HEX);
  // Read the data from the sensor
    float voltage = pzem.voltage();
    float current = pzem.current();
    float power = pzem.power();
    float energy = pzem.energy();
    float frequency = pzem.frequency();
    float pf = pzem.pf();
//lcd.setCursor(0,0);
 // lcd.print("H=");
  // lcd.print(h);
  //lcd.setCursor(0,1);
  //lcd.print(" & T=");
   //lcd.print(t);
   // lcd.setCursor(0,1);
  //lcd.print("V=");
  // lcd.print(voltage);
  lcd.setCursor(2,0);
  lcd.print("Units=");
  lcd.print(energy);
  lcd.setCursor(2,1);
  lcd.print("Price=");
  lcd.print(price);
   //delay(1000);
   int sensorStatus = digitalRead(5);
  if (sensorStatus == 0) {
    digitalWrite(18,HIGH);
delay(5000);}else
  { digitalWrite(18, LOW);  }
     Serial.println(sensorStatus);
 float adc = analogRead(33);
  float DCCV = adc * 3.3 / 4095.0;
  float DCcurrent = (2.5 - DCCV) / 0.1;
  if (DCcurrent > 69) {
   digitalWrite(18,HIGH);
  }else{
```

```
    digitalWrite(18,LOW);}
Serial.print("DCCurrent : ");
 Serial.println(DCcurrent);
int DCV=analogRead(35);
//Serial.println(DCV);
float DCvoltage=((3.3*DCV)/4095.0)*5;
Serial.print("DCvoltage:");
Serial.println(DCvoltage);
  // Check if the data is valid
    if(isnan(voltage)){
        Serial.println("Error reading voltage");
    } else if (isnan(current)) {
        Serial.println("Error reading current");
    } else if (isnan(power)) {
        Serial.println("Error reading power");
    } else if (isnan(energy)) {
        Serial.println("Error reading energy");
     }
     else if (isnan(frequency)) {
        Serial.println("Error reading frequency");
     }
    else if (isnan(pf)) {
        Serial.println("Error reading power factor");
     }
     else {
        // Print the values to the Serial console
        Serial.print("Voltage:   ");       Serial.print(voltage);
Serial.println("V");
        Serial.print("Current:   ");       Serial.print(current);
Serial.println("A");
        Serial.print("Power:   ");         Serial.print(power);
Serial.println("W");
        Serial.print("Energy:  ");         Serial.print(energy,3);
Serial.println("kWh");
        Serial.print("Frequency: ");   Serial.print(frequency, 1);
Serial.println("Hz");
        Serial.print("PF: ");       Serial.println(pf);
    }
    Serial.println();
    //delay(2000);
price = energy*10;
Serial.println(price);
 //delay(2000);
//https://api.thingspeak.com/update?api_key=2ORTO04IKNUMUGQV&field1=30&field2=70
if(client.connect("api.thingspeak.com",80))
{
url=tsAddress;
url+="api_key=";
url+="O7EQK3NFPHSOR2Z8";//provide write API Key
url+="&field1=";
url+=analogRead(34);
url+="&field2=";
url+=power;
url+="&field3=";
url+=energy;
url+="&field4=";
url+=price;
Serial.println(url);
http.begin(client,url);
http.GET();
http.end();
}
if(client.connect("maker.ifttt.com",80))
{
if(price>10){
url1=tsAddress1;
url1+="/with/key/";
url1+="iU2ljHaCsN5o4nYJQWsTZh47LbvqdLnwZi1MwqbHzyV";//provide write API Key
//https://maker.ifttt.com/use/buAYDCnW_1U1nEj7m1V0rF
Serial.println(url1);
http1.begin(client,url1);
http1.GET();
http1.end()
}
}
}
```

Results:



3. CONCLUSIONS

Energy measures are gradationally being replaced by electronic measures. Ashuge profit is lost to power theft, incorrect cadence reading and billing, the use of smart cadence analysis eliminates this issue. likewise, it'll ameliorate metering, billing effectiveness and delicacy, thereby contributing the energy in a justifiable way. Also, making it possible to use energy coffers more efficiently and furnishing real- time data useful for balancing electric outfit or loads and reducing energy outages. Consumers can fluently understand their power consumption by this means. Advance metering structure proposed gives way to Smart Grid Technology veritably soon in near future. The Power Line communication modems is been used as a devoted bias in achieving the asked pretensions. But PLC isn't so powered due to high cost needed to design transceivers at each station. Then the challenge is to lower the cost so enjoy a more effective and cheapest media of communication. Clustering in high confines has been an open problem all these times. Recent exploration show that it's preferable to use dimensionality reduction ways before clustering than clustering in the high dimension directly. therefore there's a need for good- quality, fast clustering algorithms for low-dimensional data.

REFERENCES

[1] DzemoBorovinaa, Matej Zajcb, AljoMujcicc, Andrea Tonellod, and NerminSuljanovicc, (2020), "Error performance analysis and modeling of narrow-band PLC technology enabling smart metering systems", Electrical Power and Energy Systems, Elsevier Publications, vol.116.

[2] Burunkaya M., Pars T(2017).: 'A smart meter design and implementation using ZigBee based wireless sensor network in smart grid'. 4th IEEE Int. Conf. on Electrical and Electronic Engineering (ICEEE), pp. 158–162

[3] Tamkittikhun N., Tantidham T., Intakot P.(2015), 'AC power meter design based on Arduino: multichannel single-phase approach'. IEEE Int. on Computer Science and Engineering Conf. (ICSEC), pp. 1–5

[4] Yang L., Chen X., Zhang J., ET AL.(2015): 'Cost effective and privacy preserving energy management for smart meters', IEEE Trans. Smart Grid, 6, (1), pp. 486–495

[5] Ahlem A., Hfaiedh M., Amira H.(2014): 'Design and implementation of single phase intelligent energy meter using a microcontroller interfaced to PC'. IEEE 15th Int. Conf. on Sciences and Techniques of Automatic Control and Computer Engineering (STA), pp. 191–195

[6] Arote S., Mulay G., Khaparde A.(2016): 'Design and implementation of smart three phase energy meter'. IEEE Int. Conf. on Smart Grid and Clean Energy Technologies (ICSGCE), pp. 44–49

[7] Stankovic L., Stankovic V., Liao J., ET AL.(2016): 'Measuring the energy intensity of domestic activities from smart meter data', Appl. Energy, 183, pp. 1565–1580

[8] Wang S., Luo R., Zhou H., ET AL. (2011), 'Research on reliability enhancement testing for single-phase smart meter'. IEEE Int. Conf. on Electrical and Control Engineering (ICECE), pp. 3128–3131

[9] Damavandi M., Krishnamurthy V., Martí J (2015).: 'Robust meter placement for state estimation in active distribution systems', IEEE Trans. Smart Grid, 6, (4), pp. 1972–1982

[10] Ngamchuen S., Pirak C.(2013): 'Smart anti-tampering algorithm design for single phase smart meter applied to AMI systems'. IEEE 10th Int. Conf. on Electrical Engineering/Electronics, Computer,Telecommunications and Information Technology (ECTI-CON), pp. 1–6

[11] Nsilulu T. Mbungu, Raj Naidoo, Ramesh C. Bansal, andMinneshBipath, (2017) "Optimal single phase smart meter design", The Journal of Engineering, vol.7, Iss. 13, pp.1220-1224.

[12] Haque, Md Mejbaul; Hossain, Md Kamal; Ali, Md Mortuza; Sheikh, and Md Rafiqul Islam,(2011), "Microcontroller Based Single Phase Digital Prepaid Energy Meter for Improved Metering and Billing System", International Journal of Power Electronics and Drive Systems, Vol.1, Iss.2, pp. 139-147

[13] Shashank Singh, S. L. Arunand M. P. Selvan,(2017) "Regression based approach for measurement of current in single-phase smart energy meter", 2017 IEEE Region 10 Symposium (TENSYMP), pp.1-5

[14] D.A. Shomuyiwa and J.O. Ilevbare, (2013),"Design and Implementation of Remotely- Monitored Single Phase Smart Energy Meter via Short Message Service (SMS)", International Journal on Computer Applications,, vol.74, No.9, pp. 14-22.

[15]Mayukh Roy Chowdhury, Sharda Tripathi, and Swades De, (2020) "Adaptive Multivariate Data Compression in Smart Metering Internet of Things", IEEE Transactions on Industrial Informatics, pp.1-11.