HACKATHON
HEWLETT PACKARD ENTERPRISE **– KSHITIJ'18**

Topic-Smart Energy Management System & Controlling Electrical Devices through mobile application

Team details-
Name- Oyster
Member 1- Sholi Singh Jindal.
Email-id Sholissj@gmail.com **contact number -9454618621**
College – Meerut Institute of Engineering & Technology,Meerut

Introduction to idea-Controlling Electrical devices requires switching them on/off , the data generated during process -like time of switching devices on/off,change in power consumption, change in power factor etc is used to manage Electrical energy consumption & efficient creation(by Electricity suppliers,DISCOMs).

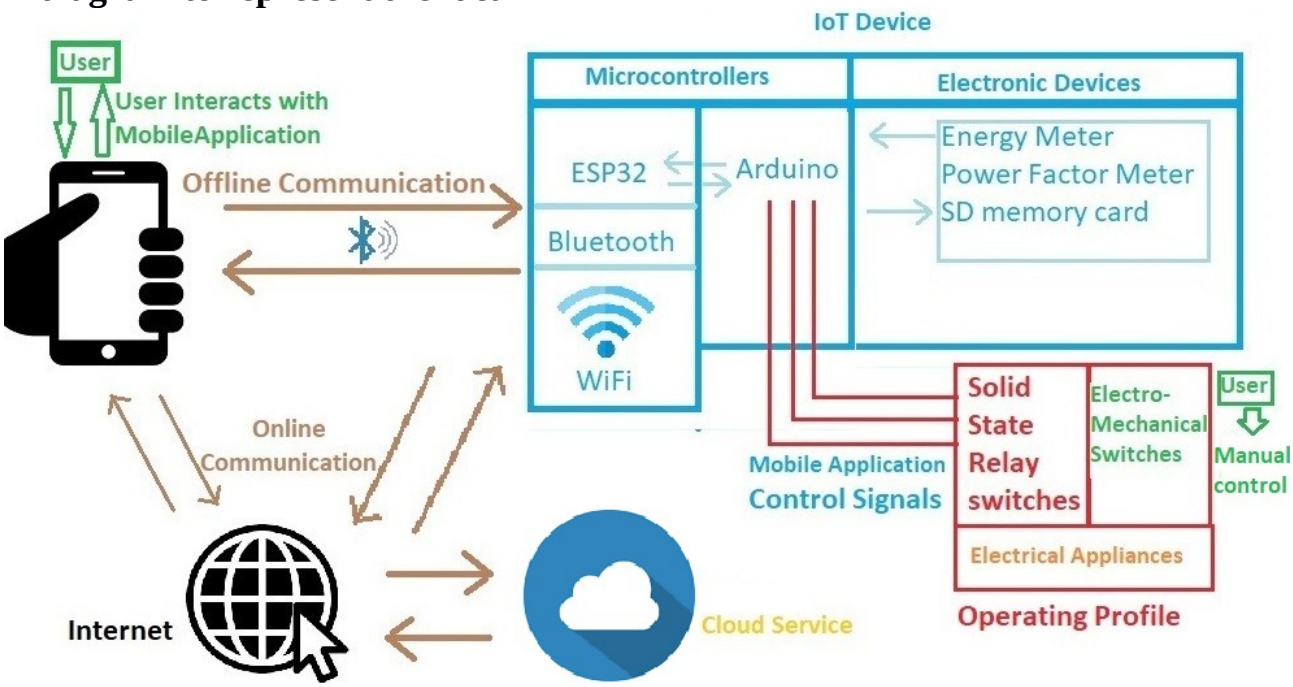
My idea is to make an IoT system which comprises of

1. **a mobile application**
2. **A micro-controller Arduino + ESP32 + SD memory Card or Raspberrry Pi**
3. **Electronic switches – GTO Thyristor or Solid State Relay switches.**
4. **Electric Energy Meter (calculates energy & power factor of total circuit)**
5. **Cloud Storage/Database for storing database**

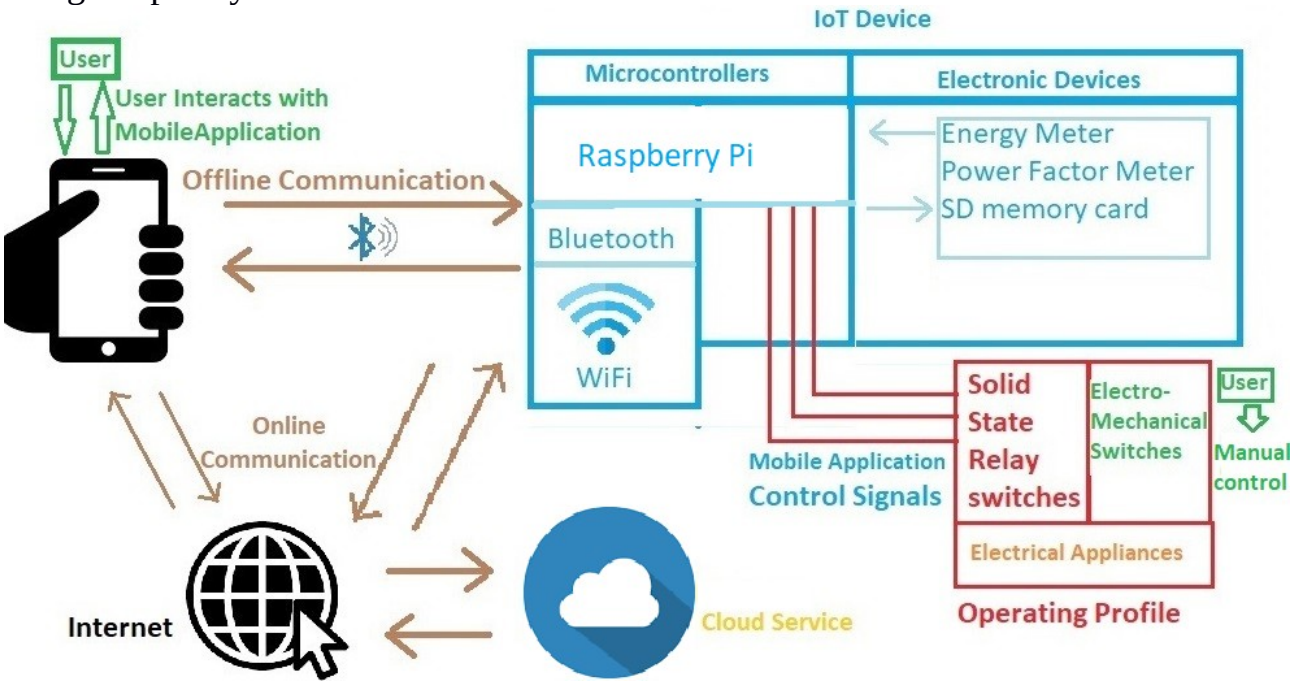
Problems it solves-

1. **Controlling Electrical devices through app.**
2. **Automation of Electrical devices.**
3. **View Electrical data like Power Consumption & Cost , Power factor of circuit.**
4. **Provides Electricity Consumption pattern on graph/table format for home users so that they can know about their usage.**
5. **View Electrical Device Electricity Consumption data like length of use & power consumed by device.**
6. **Provides Electricity Consumption Data to Electricity manufactures so that they can know the demand of electricity in real-time & produce electricity accordingly.**
7. **Calculate Electricity Availability in area.**
8. **Generates data on electricity usage which can be used for solving problems through machine learning.**

A diagram to represent the idea-



using Raspberry as Micro-Controller



The IoT system -
The IoT consists of a mobile application, a microcontroller & a database or cloud storage.

Features of mobile application
Mobile application will give user following options-

1. Controlling appliances through SSR (Solid State Relay)switches. User will get option to organize/assign switches in room & naming them(according to appliances connected).
2. Define schedule of an appliance in room.
3. Viewing Electricity data in following ways-

   a) Total power consumed & cost

   b)Graph of power consumed daily/weekly/monthly

   c)Electric power rating, length of use & consumption of appliance.

   d)View pattern of electricity usage & power factor.

Task of micro controller-

1. Receive signals from mobile app to controls switches
2. Record on/off time for each electrical device when it is switched on/off.
3. Automate Electrical Device according to user defined schedule.
4. Calculate & derive data like Total running time of an appliance,
5. Calculate & Record Electricity Data(Total Consumption & Cost, power factor of circuit, time of peak usage, maximum load achieved by house-hold circuit & the time duration of it).
6. Calculate Power Availability in house-hold circuit.
7. Sending data to mobile application.

Task of Cloud Storage/Database-Store data from all the mobile devices.

Working of IoT system-

Cloud Storage/Database-Stores data from all the mobile devices. Data includes-

1. Power Load from a house- Combination of this data from all the mobile application will provide power consumption of whole city, state or even country.
2. Electricity availability in house-hold circuit- Combination of this data from all the mobile application will provide electricity availability in an area.

3. Location of house of whom electrical data is being sent by the application.

Each application/mobile device/user can assigned a unique id to reduce redundancy of data & proper management.

Working/features of mobile application-(assuming Arduino & ESP32 is used build IoT device, Both can be replaced with Rasberry Pi)

Feature 1-Switching off/on electrical appliances

Organize/assign switches in room & naming them-

User will first choose option to define rooms in application to allocate switches present in circuits.

SSR switches will be identified by number on them.

These switches when categorized in rooms will allow ease in access.

Rooms & switches will be assigned names for further ease.

Controlling Switches-

To control switches of room user can navigate to the room section , turn off/on entire power supply of room(switching off/on all appliances) or selectively switching on/off devices.

As soon as user chooses option for managing appliances

- signal is sent to Arduino via ESP32. Arduino checks status of all switches & reports the status to application.
- User is presented with an updated view of all rooms & switches. As soon as user changes state of a switch ,signal is sent to Arduino to turn the switch on/off. Arduino then does the task & reports back the status of switch to software.
- All necessary data like Switch on/off time of appliances, current power consumption is stored in memory card by Arduino.

Feature 2- Defining schedule of an Electrical appliance.

For a schedule of an appliance its start/stop time & working days of week is defined by user. For each appliance(represented by its SSR number) an option to define it's schedule will be given to user.

If any appliance(switch) is marked to be automated on schedule, its data is sent & saved in Arduino to control appliance according schedule defined by user. User can also interrupt an appliance's schedule manually, whenever this interruption is done user be notified of the predefined schedule.

Feature 3 -Viewing power consumption data

i)Total power consumed- To view power consumption data , user has option to view real-time consumption & cost of electric power used. Arduino will retrieve data from electric meter & power factor meter & sends it to application. Data can presented to user as per it's convenience. User can view total power consumed yet & the cost which has occurred.

ii)Graph of power consumed daily/weekly/monthly- Arduino will also record power consumption data on hourly basis ,store it in memory card & send it to application whenever it communicates with it. This set of data will help application to present data of power consumption graphically on daily/weekly/monthly basis.

iii) power rating,length of use & electricity consumption of appliance

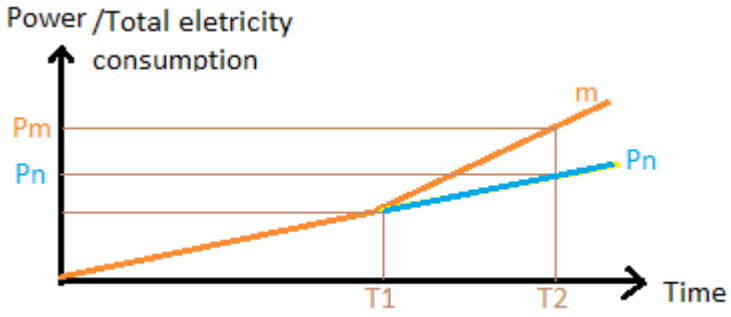
For this task user will have to switch off & switch on an appliance, Arduino will record power consumption data regularly in this case. Rise in power consumption rate over a fixed time can derive the power rating of a single device.(When a power rating is being derived no other device can be switched on/off else derivation will be aborted & user will have to start again)

Computations involved for finding power rating- Here if slope n is line representing Power Consumption without Appliance X,it is calculated when appliance is turned off(before T_1).Slope m (denoting real-time curve between T_1 & T_2) is power consumption after turning on appliance X. T_1 & T_2 time interval are calculated by Arduino.

$Pm = T_2 \times m(slope)$,

$Pn = T_2 \times n(slope)$.

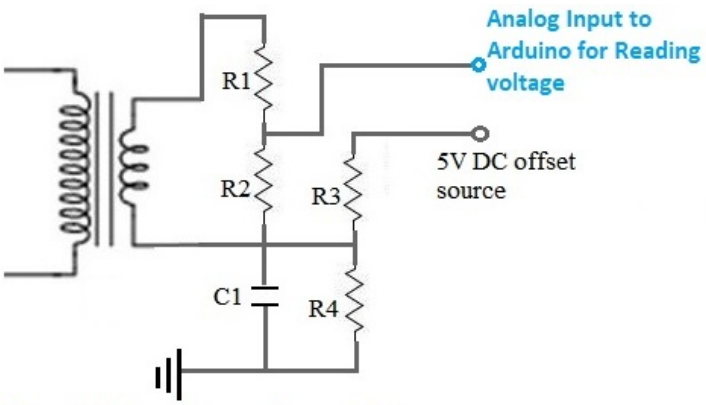
Electric Power rating of appliance X= $(Pm-Pn)/(T_2-T_1)$.



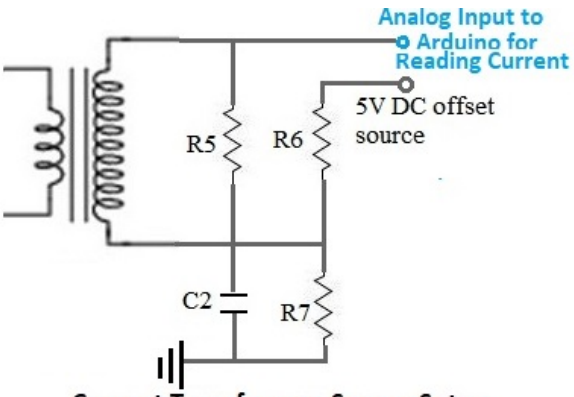
On/off time-stamp will be saved by Arduino whenever any appliance is switched on/off. This time-stamp data helps in deriving total running time of an appliance.

Product of Total running time & Derived electric rating of an appliance will give total power consumption.

iv)Power usage pattern &Power Factor of circuit- Power usage pattern is derived from electricity consumption data recorded by Arduino. Through this data peak time & max power consumption is derived. Power factor is derived from Current Transformer & Potential Transformer sensors connected to Arduino to analyze actual power usage & electrical circuit's efficiency.

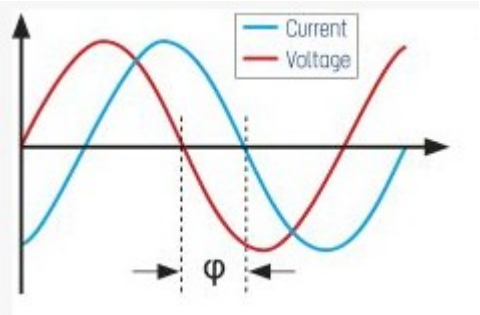


Potential Transformer Sensor Setup

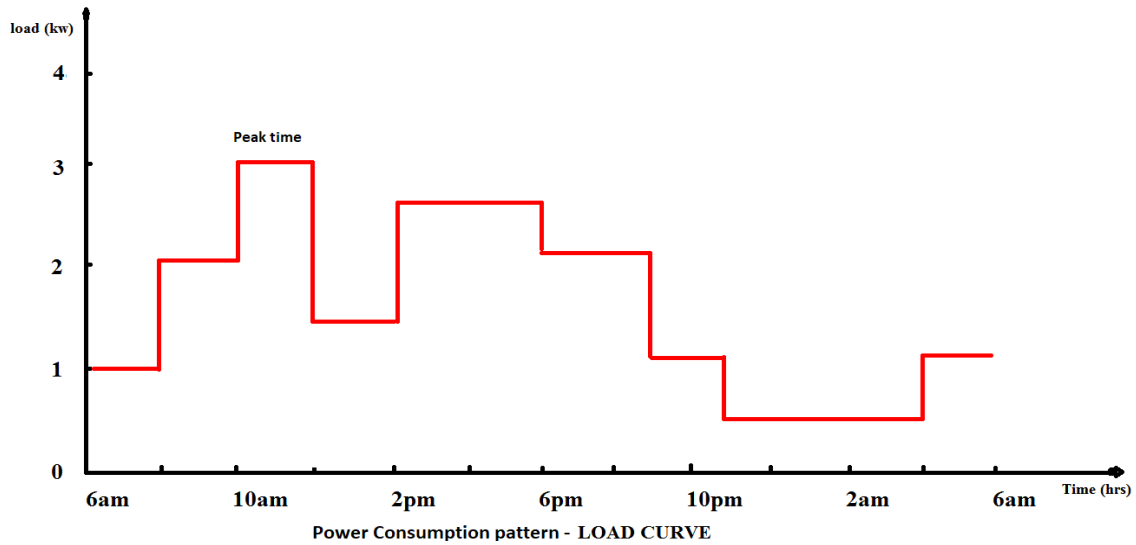


Current Transformer Sensor Setup

<u>Power factor meter</u>-Custom built Current & Potential Transformer Sensor used to find phase difference in Arduino by calculating the time difference when Current & Voltage reaches 0 value in both inputs.



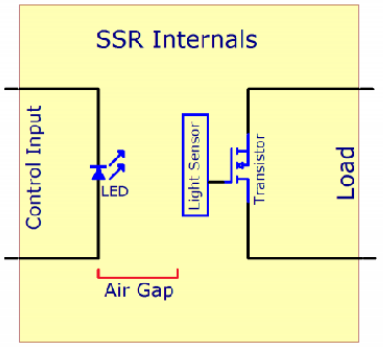
Power Usage pattern will be shown in following manner-

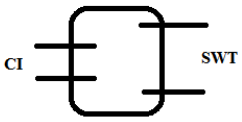


Power Consumption pattern - LOAD CURVE

A safety feature of manual override in case failures of mobile application-

Manual Overriding of Switches-To provide user an option for switching on/off appliances manually a 2-way switch is implemented at each circuit board to switch appliances control to manual or application mode. A diagrammatic explanation of Overriding implementation is given below-

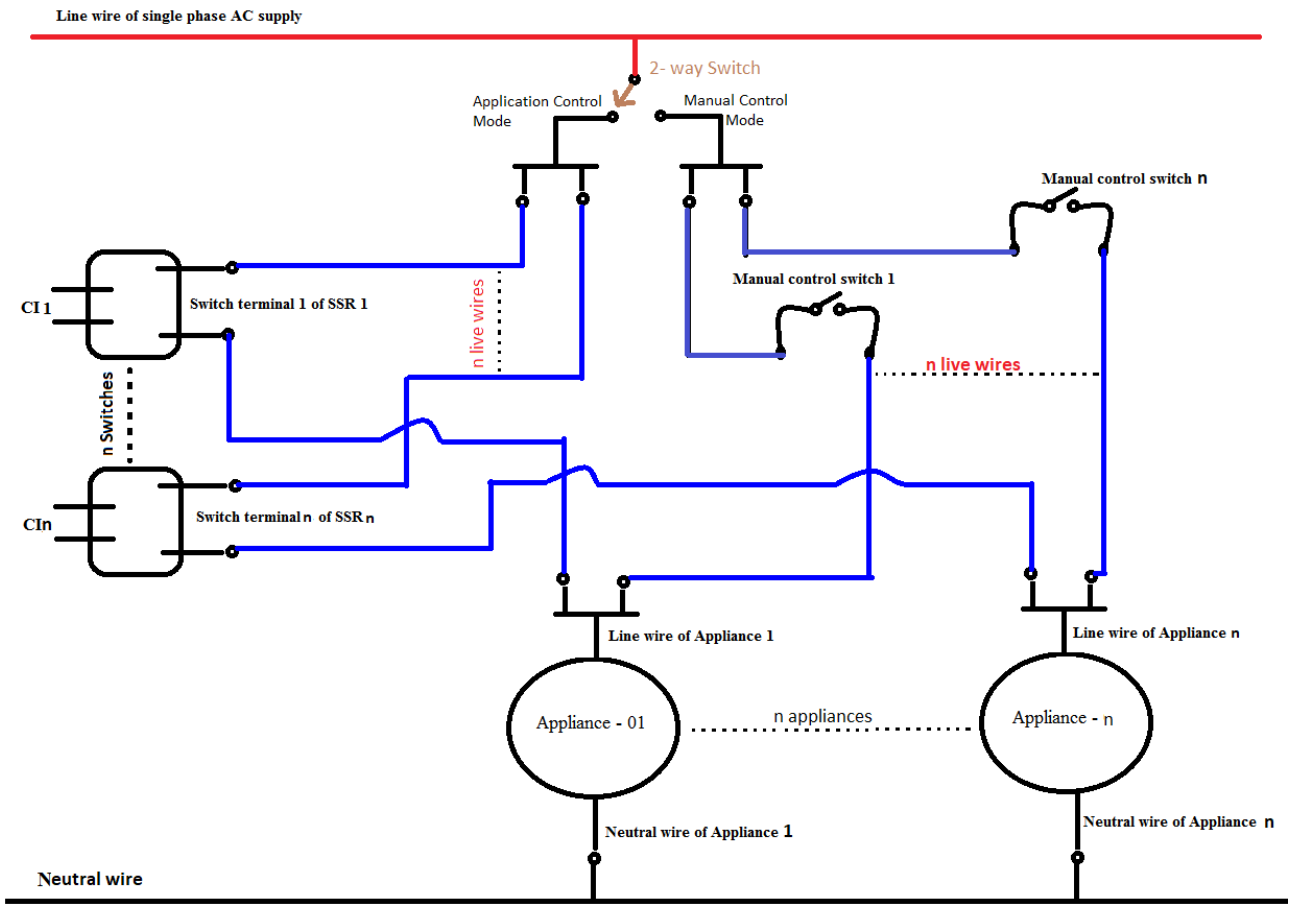




Where, CI = Control Input
SWT = Switch Terminal

 # Symbol for representing an SSR

Insight of an SSR switch



Thank You

Team Oyster(Sholi Singh Jindal)