Power optimization by load monitoring and control

Prasanna Rajaram

September 17, 2017

Contents

1 Introduction

In conventional power saving techniques, the user has to be vigilant in turning ON/OFF electrical loads as and when required. This ensure optimal consumption and reduce power bill for the consumer. This effort can be automated and optimized better. This project will enable the user to control loads from anywhere as long as the mobile device and the Raspberry Pi (@ home) are connected to Internet.

The intention is to retrofit this system into the existing domestic wiring scheme of the user with minimal changes as much as possible.

1.1 Purpose of this document

The purpose of this document is to list down the design specification and design details for this home automation project.

1.2 Objectives

1. The project will enable users to switch electrical loads through a mobile application

2. The load ON/OFF times shall be logged. This logged information can be used to provide the user with suggestions for power saving.

3. Using interfaces like gas detectors, one can log information if there were any gas leakage in any one of the apartments using centralized logging system.
4. Power factor correction for inductive loads and switching in/out "tank circuits" shall also be done based on the load conditions
5.
6. ....

2 Design

3 Wiring scheme

In the conventional power switching arrangement, the following modifications should be made to implement this design. The SPST switch used conventionally to switch ON/OFF loads, should be replaced with a DPST switch (commonly known as "two-way" switch). A relay with DPST switching arrangement should be connected to the new DPST switch. This two-way switch wiring scheme established by the new two-way switch and the DPST portion of the relay allows the user to switch ON/OFF loads either from the two switch or by microcontroller. The relay shall be driven by a Arduino microcontrolller. The wire leading to the load from the relay contact shall have an ACS712 (Hall effect sensor) connected in series before the load. The pupose of the Hall effect sensor is to measure the current drawn by the load. The output of the Hall effect sensor shall be wired as input to one of the analog channels of the Arduino controller. The wiring drawing is presented below

Wiring diagram - insert here

4 High level block diagram

The following portions shall make up this design

1. Wiring scheme with modified switch, relay and Hall effect sensor
2. Arduino (Micro) micro controller
3. Raspberry Pi

4.1 Arduino micro controller

The microntroller will be able to see the ON times of all the loads if either they are switched via the controller itself or directly by the DPST switch.

This information is logged at the microntroller level and passed on to Raspberry Pi.

4.2 Raspberry Pi

4.2.1 HMI

The Pi is responsible for pulling all the data logged by the microcontroller and present it on screen for the user to review. The Pi functions like an HMI. A small 7" touch enabled screen could be wired to RPi to enable users to control loads and view reports locally

4.2.2 Data analysis

The next function of Pi is to do statistical analysis of the logged information. This analysis will come out with results that could provide suggestions to the user with load profiles. This report will tell the user how much power was consumed by respective loads. It will also present the normal ON times of the load during a day. Based on this data, the user can take an informed decision about turning OFF loads at certain time of the day.

4.3 RPi Alternative

Provided the Arduino is able to communicate directly to a server over the internet (using Arduino Wifi Shield), then all data analysis and HMI functions can be passed on to application in the server machine. Data can be pulled over the network to a mobile device to control and perform HMI functions.

4.4 HMI UI Design

It is envisaged to have a one screen display which will display all types of loads on the first screen. The groups could be like lights, fans, air conditioners, heaters etc. When the user taps on one of the groups (say lights), the screen would expand to a list or table (list/tabulate all lights with their statuses). From this expanded page, the user would be able to toggle the load status.

This can further be improved based the users' usage pattern. For example, in day time of the hot months, the user would want to keep the fans running and my turn ON/OFF air conditioner based on requirement. In this case, (assuming) good ambient lighting from the sun, the user may never turn ON the lights during the day in the balcony / kitchen / hall. So depending the season of the year, time of day the list could dynamically display only the frequently used loads for control besides the exhaustive list.

5 Some special controls

Assuming that a humidity sensor and temperature sensor module are wired into the Arduino, we can collect humidity, temperature and air conditioner "turn-ON" times information based on user behaviour. This information could be used to predict when the user would want to have the air conditioner turned-ON/OFF from the humidity and temperature sensor readings.

6 Bill of material

S.No	Mandatory /Optional	Part No.	Model Name	Approx. Cost
1	M		Arduino Micro	
2	M		Arduino Wifi Shield	
3	O		Raspberry Pi	
4	M		Relay board (X relays)	
5	M		Hall effect sensor	
6	O		Humidity sensor	
7	O		Temperature sensor	
8	O		Gas (LPG) detector	
9	M		Arduino Wifi Shield	
10	M		Breadboard	
11	M		Power supply +5V 2A	
12	M		Resistors - 220 Ω - 10k Ω - ... Ω	
13	M		Capacitors - 10 µF	