Raspberry Pi based Energy Management System

Jayanth S, Poorvi MB
Jayanthsuresh09@gmail.com, poorvimb83@gmail.com
Department of Electronics and Communication Engineering
School of Engineering and Technology, Jain University
Bangalore, India

Sunil MP
mp.sunil@jainuniversity.ac.in
Department of Electronics and Communication Engineering
School of Engineering and Technology, Jain University
Bangalore, India

Abstract— **In this research we are presenting a system which addresses the energy wastage arising from carelessness and negligence by a person when he leaves his house without turning off all the unnecessary appliances. The proposed system uses the Bluetooth communication in the user's phone to determine whether the user is at home or not. It is not necessary for the user's phone to be in discoverable mode for its functionality, it should be just turned on. When a user is out of reach for a certain duration, the system immediately turns off unnecessary devices such as lights and fans. Multiple Bluetooth devices can also be used in case of multiple users. The user can control the appliances manually through switches when at home.**

Keywords—Bluetooth; Raspberry Pi;PuTTY;Win32DiskImager

I. Introduction

The uses of electricity have increased from the last decade. With the growth of technology, the need for electricity has increased. Electricity is necessary to run our computers, lights, fans, air conditioners, washing machines, water heaters, street lights and more. We have numerous applications but we produce limited electric power. Since generation of more electric power to meet the demand is difficult, an easier solution would be to manage the available electric power efficiently. The energy management deals with the conservation of resources and our concern is the electric power. Energy management also has an added advantage of reduced resource cost due to increase in efficient usage and decrease in wastage.

As the technology advances, more and more devices are brought home by a person which requires power to operate, which results in crisis of electric power. In the recent years, several systems have been designed to overcome the crisis by efficient energy management. But these systems are expensive and are not easy to install. Because of this, people do not opt for such systems for their homes. To overcome such problems, we are presenting a new system which is of low cost, easy to install and can be easily upgraded. The presented design uses the famous Raspberry Pi along with a Bluetooth dongle. The credit card sized computer offers low cost and compact size, whereas the Bluetooth offers very low power consumption.

The Raspberry Pi has a Bluetooth dongle connected to it. The Raspberry Pi searches for nearby Bluetooth devices. If the user's device is found, the mains is turned on. If not, the mains supply is turned off to the appliance. This offers another advantage apart from energy management. It offers safety, the unexpected short circuits which causes fires are eliminated.

The Bluetooth technology was used because of two reasons. Firstly, availability, most people have mobile phone which has the Bluetooth function embedded into them. It is not necessary to have a smart phone to have Bluetooth, most mobile phones have Bluetooth. Because of its wide usage, Bluetooth was the obvious choice. Secondly, low power, unlike other radio communication techniques, Bluetooth offers very little power consumption, especially during idle state. Also our system does not require the mobile device to be paired. Hence the Bluetooth does not significantly reduce the duration of usage of the mobile device between charges. As we can see, due to these reasons Bluetooth was the optimum choice

In our prototype, we have used the Raspberry Pi Model B shown in Fig 5, which offers a 700 MHz processor, which can be overclocked to 1 GHz. The Raspberry Pi comes along with a 512MB RAM. We have HDMI and composite video out which is of less importance in the presented design. The Pi also comes along with two USB ports which can be used for the Bluetooth dongle. The Raspberry Pi requires 700mA at 5V to operate. It includes a 26 pin expansion header consisting of GPIOs which is necessary for interfacing. The system can be upgraded to work using IOT by utilising the available Ethernet port on the Raspberry Pi or by using a Wi-Fi dongle.

II. Literature review

Energy conservation is the main goal of energy management. Most of the energy management system aims at the grid level and the industrial level. Industries require lot of power and hence energy management plays an important role. With the increase in usage of various electrical appliances by households, the crisis for power remains. To overcome this crisis many solutions were presented.

Firstly, the transition from incandescent bulbs to Compact florescent lamps. This transition resulted in enormous energy saving. The CFL consumes fraction of the power consumed by the incandescent bulbs to produce the same luminescence. Apart from less power consumption, the CFL offered longer lifespan.

Secondly, the transition from CFL to the LED. This transition is not as effective as the previously discussed situation, since the cost of LED's are higher than CFL's. LED's higher lifespan compared to both the incandescent and

978-1-5090-4556-3/16/$31.00 ©2016 IEEE

the CFL. The CFL and the LED consume approximately the same power to produce the required luminescence.

Even with the introduction of CFL and LED, the power crisis was not completely addressed. Hence the necessity for the Energy Management increased. The other reason is the wastage of power by an individual by his negligence and carelessness. This power could have been effectively used to provide electricity to the villages or used effectively.

III. Proposed System with Block Diagram

The block diagram of the proposed design is illustrated in the figure A. The proposed system consists of mainly 3 blocks.

Firstly, The Raspberry Pi acts as the brains of the system. It has a program written in Python. When the program executes, the Raspberry Pi decides when and what device to turn on. The Raspberry Pi is connected to the Bluetooth Module, power supply and the relay board.

Secondly, USB Bluetooth dongle is connected to the Raspberry Pi using the female ports available on the Raspberry Pi. The USB Bluetooth dongle should be Bluetooth 2.0 or greater.

Thirdly, the relay board is necessary to control the electrical appliances. The relay board provides magnetic isolation between the Raspberry Pi and the Mains. The mains power supply can be 110-240 V.

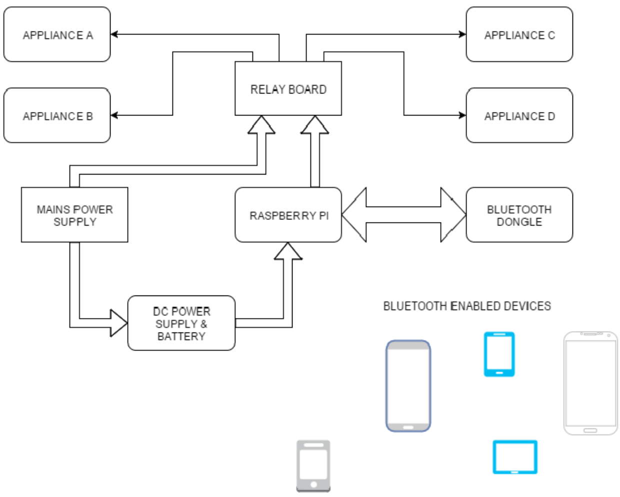

Fig.1. Block diagram of the proposed design.

The flowchart of the presented design is shown in Fig.2. When the system is powered on, the USB Bluetooth dongle driver is loaded by the Raspberry Pi. The program is written in python and is executed. When the program is executed, the raspberry pi initializes its general purpose input output, also known as GPIO. The Raspberry Pi initially sets all the outputs to low, in order to prevent any accidents. The Raspberry Pi then checks the timestamp and prints it on the monitor for reference. This is visible only if a monitor is attached or a SSH connection is established to the Raspberry Pi. Now The Raspberry Pi searches for all the nearby Bluetooth devices. As soon as the Raspberry Pi detects a Bluetooth device, it compares the found device's Bluetooth address with the user's Bluetooth address. If there is a match, the Raspberry Pi turn on all or the required electrical appliances connected to it through the relay board. The power to these appliances remain as long as the Bluetooth device is in range. It is to be noted that the Raspberry Pi controls only the power to the device. It can be manually turned off using the preexisting switches of the respective devices. When the user forgets to turn off any of the appliance and goes out for some reason, the Raspberry Pi detects this by the out of range Bluetooth device. All of us carry mobiles with us when we step out of the house, hence the system cannot find the user's Bluetooth device anymore. When this happens the system cuts the power supply to all the devices connected through its relay board. Even if the pre-existing switches are used, the devices can't be turned on. The system continues to scan for nearby devices with a certain delay. The delay can be varied depending on the situation. In the prototype, we have assumed an optimum delay of 10 Seconds. Once the user's Bluetooth device is found, the Raspberry Pi energizes the relay to control provide power back to all the connected appliances.

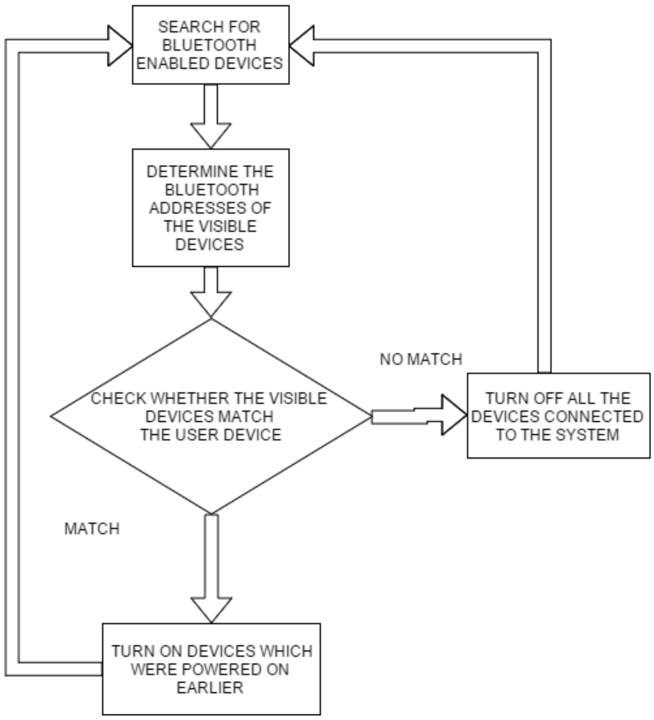

Fig.2. Flow Chart of the proposed design.

IV. Hardware Implementation and its Components

A. The Raspberry Pi

The heart of the proposed design is the Raspberry Pi, whose technical specifications were discussed earlier. In the design we make use of only one USB port and the GPIO. The USB port is necessary to connect the Bluetooth 2.0 dongle to the Raspberry Pi. The GPIO's on the Raspberry Pi are 3.3V logic. Unlike most microcontrollers which work on 5V, The

Raspberry Pi's GPIO can take a maximum input of 3.3V and can produce an output voltage of maximum 3.3V. This should be considered while connecting any input peripherals to the Raspberry Pi. The Raspberry Pi's output can be connected to an optocoupler to isolate the Raspberry Pi. Else the Raspberry Pi can be used to drive a sugar cube relay directly with the help of a transistor. In our design, we have used BC547 NPN transistor as a switch to drive the Relay. This ensures that the GPIO of the Raspberry Pi are not damaged due to high current.

Fig.3. Raspberry Pi Model B

B. Micro SD card

To get started with the Raspberry Pi it is necessary to have a minimum of 4GB micro SD card for the Raspberry Pi to load the OS. The memory card is loaded with the Raspbian wheezy, a debian OS for the Raspberry Pi. For the Model B, micro SD cards are nor supported and hence an adapter is also necessary.

Fig.4. A 4GB Micro SD card along with its adapter.

C. Bluetooth dongle

The main idea of the presented design depends on the Bluetooth technology. Hence we require an USB Bluetooth dongle of version 2.0 or greater. The prototype uses a Leoxsys USB Bluetooth module. This dongle provides the necessary hardware for the Raspberry Pi to communicate with the nearby Bluetooth devices.

Fig.5. A USB Bluetooth Dongle

Fig 6 illustrates the proposed design. The design is simple as it requires a relay board and a Bluetooth dongle. An individual relay can be used to control all the devices or multiple relays can be used to control high current appliances. Since the Raspberry Pi offers multiple GPIO pins, it is easier to add multiple relays. The proposed design consists of an individual relay driven by a NPN BC547 transistor. All the devices that are necessary to be controlled are connected in series with the system and the mains supply. The sugar cube relay offers two type of connections along with a common connection. The two connections are 'normally open' and 'normally closed' also referred to as NO and NC. The normally closed connection and the common connection of the relay are used. This is used because the electrical appliances would still work in the absence of the power to the Raspberry Pi or when there is a snitch in the system.

Fig.6. Proposed system

V. SOFTWARE IMPLEMENTATION AND ITS TOOLS

The Raspberry Pi requires an operating system in the micro SD card to boot. To install the operating system, first we require a minimum 4 GB memory card. The memory card is inserted to a Windows operating system and is formatted. Win32 Disk Imager is necessary to create a bootable drive of the operating system image file onto the micro SD card. The Raspbian Wheezy operating system is downloaded from the official Raspberry Pi website. The source path of the downloaded file is selected on the Win32 Disk Imager software. The destination drive, micro SD card is selected and the image is written on the memory card. Once this process is complete, the memory card is removed from the computer and is inserted back to the Raspberry PI.

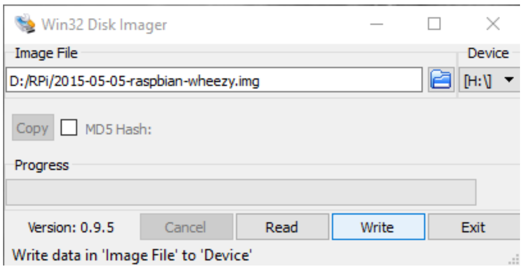

Fig.7. Win32 Disk Imager software

It is now time to power on the Raspberry Pi with a 5V 700mA adapter or a power bank. Once the Raspberry Pi powers on, we can configure it using keyboard, mouse and a monitor or using SSH.

To establish a SSH connection, the Raspberry Pi should be connected to the network with an Ethernet cable. A host system with Windows operating system and connected to the same network is necessary to establish SSH connection. PuTTY is run on the host computer, the IP address of the Raspberry Pi is found out from the router page and is entered. An SSH session is opened as shown in figure 8.

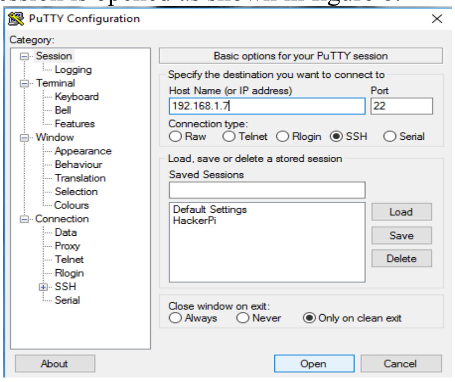

Fig.8. Connecting to the Raspberry Pi using PuTTY

When a new session is started, the raspberry pi requests login name and password. The default login name is 'pi' and the default password would be 'password'. Once the sign-in is complete, 'sudo raspi-config' is executed on the terminal and the Raspberry Pi is configured as required. The packages are similarly updated and upgraded. We also need to install the necessary drivers for the Bluetooth dongle since the operating system does not come preinstalled with the Bluetooth dongle's drivers.

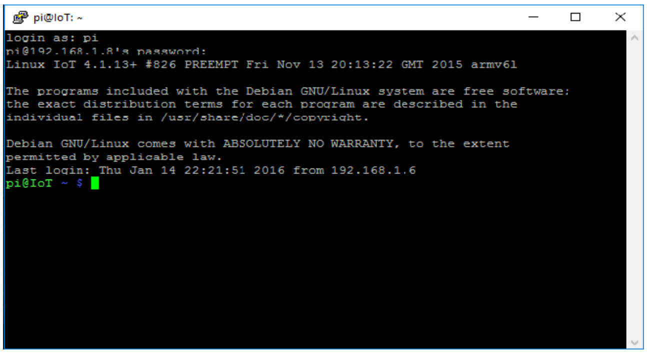

Fig.9. SSH connection to Raspberry Pi using PuTTY

To install packages super user access should be provided. The packages are installed using the apt-get command. The Bluetooth drivers are obtained from installing the Bluez package. The Raspbian Wheezy comes along with Python. The program for the proposed design is written and executed using python. The Bluetooth address of the user device is found out from the about phone section in the settings. This address is added in the program. The program is added to the list to execute at the time of boot. This way no human intervention is required by the system to function as desired once properly configured. The Ethernet connection can be removed once the SSH connection is terminated. When the Raspberry Pi boots, the program executes.

VI. Results and Discussions

In the presented design, a Bluetooth USB dongle is connected to the Raspberry Pi to allow it communicate and detect the nearby Bluetooth devices. Once the system is powered ON, the system searches for any nearby Bluetooth enabled devices with a certain delay. When a device is identified, the presented system checks the Bluetooth address of the identified device with the database. If the addresses match, the system immediately provides power to the appliance which was powered earlier. The power to these appliances remains as long as any one of the Bluetooth enabled devices, whose address is present in the database is in the vicinity. When the user forgets to turn off any appliances and leaves the building, and there are no other users in the vicinity, all the devices are shut down.

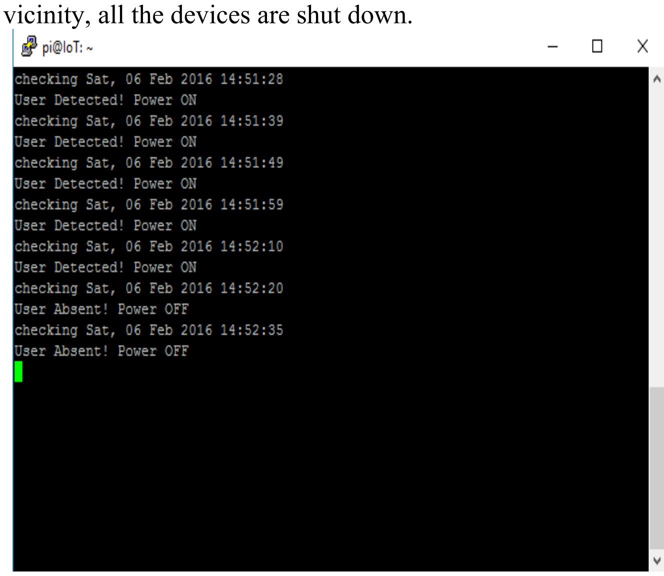

Fig.10. Graphical Output of the proposed design

Figure 10 shows the output of the python program execution. This system is designed to check the presence of the user every 10 Seconds. The Raspberry Pi can be made to run headless i.e. without a monitor attached to it to view the program execution illustrated in the figure 10. By running the raspberry Pi headless, the system is compact, easy to install, and consumes less power.

Fig.11. Appliances turned off in absence of the user

Fig.12. Appliances turned on in presence of the user

This system can be used to turn off specific devices or the power supply to the entire house. It may be necessary to keep certain appliances to be running in our absence such as aquarium filters, security systems and personal computers. In such situations, only specific appliances such as lighting, fan, water heaters etc. can be controlled. By turning off the power to the Raspberry Pi, the system can be shut down without any changes to the system. This way the manual control of the electrical appliances is achieved. This feature is presented for situations when the Raspberry Pi has a snitch.

VII. CONCLUSION

The implementation of the presented design results in reducing the unnecessary power consumptions across the globe. With the development of technology, people carry multiple Bluetooth enabled devices such as the mobile phone and fitness band. This system is not restricted to only mobile phones. People who wear fitness bands to track their daily activity can use the Bluetooth address of the device to control the system. The user can turn off their Bluetooth device or devices when they are about to sleep to turn off any devices which was left turned on. Since most of the electrical appliances are turned off in the absence of the user, the accidents due to electrical fires are minimised. As we can see, this system not only helps save energy but also improves safety. The use of Raspberry Pi helps in keeping the system cost low, yet allowing it to be upgraded. The system can be incorporated with the IOT to be able to monitor the power usage across the house.

REFERENCES

[1] Jinsoo Han, Chang-Sci Choi, Wan-Ki park, Ilwoo lee, and Sang-ha Kim, "Smart Home Energy Management System Including Renewable Energy Based on ZigBee and PLC", IEEE Transactions on Consumer Electronics, Vol. 60, N0.2 2, May 2014.

[2] John C, "Maximizing Energy Saving With Enterprise Energy Management Systems", IEEE Power & Energy, 2004.

[3] Sudip Misra, Ayan Mondal, Shukla Banik, Manas Khatua, Samaresh Bera, and Mohammad S. Obaidat, "Residential Energy Management in Smart Grid: A Markov Decision Process-Based Approach", 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of things and IEEE Cyber, Physical and Social Computing.

[4] Smitha S.D, Dr. J. S. Savier, Asst.Prof. Fossy Mary Chacko, "Intelligent Control System for Efficient Energy Management in Commercial Buildings", International Conference on Microelectronics, Communication and Renewable Energy, 2013.

[5] Joao Pedro F. Trovao, Victor D. N. Santos, Carlo Henngeler Antunes, Paulo G. Pereirinha, Humberto M. Jorge, "A Real-Time Energy Management Architecture for multisource Electric Vehicles", IEEE Transactions On Industrial Electronics, VOL. 62, NO. 5, May 2015.

[6] Gopinath Shanmuga Sundaram, Bhanuprasad Patibandala, Harish Santhanam, Sindhura Gaddam, Vamsi Krishna Alla, Gautham Ravi Prakash, Shiva Chaitanya Vishwakarma Chandracha, Sindhu Boppana, and James M. Conard, "Bluetooth Communication using a Touchsreen Interface with the Raspberry Pi" , IEEE, 2013.

[7] J. Haartsen, M. Naghshinesh, J. Inouye, O. J. Joeressen, W. Allen, "BLUETOOTH- The universal radio interface for ad hoc, wireless connectivity", ACM SIGMOBILE Mobile Computing and Communications Review, Vol. 2, No. 4, pp. 38-45, October 1998.

[8] J. C. Haartsen, "Bluetooth- a new low-power radio interface providing short-range connectivity", Ericsson Radio Syst. B. V., Emmen, Netherlands, Vol. 88, Issues. 10, October 2000.

[9] J. Bray, C. F. Sturman, "Bluetooth 1.1: Connect without Cable", Pearson Education, edition 2, 2001.

[10] Powers, Shawn, "The open-source classroom: your first bite of raspberry pi.", Linux Journal 2012.224 (2012): 7