CS 308 Embedded Systems Lab

Systems Requirements Specification

Power Saver Servant

Submitted in the partial fulfilment of the requirements for the degree of

CS 308 Embedded Systems Lab

By

Vihang Gosavi vihang@cse.iitb.ac.in 09005016

Ashish Yadav ashishyadav@cse.iitb.ac.in 09005018

Under the guidance of

Prof. Kavi Arya

Department of Computer Science and Engineering

Indian Institute of Technology, Bombay

Table of Contents

Chapter 1

Introduction

1.1 Introduction and Rationale

The purpose of this document is to present a detailed description of Power saver servant. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for the developers of the system.

1.2 The Idea

Energy is the most critical factor responsible for human growth and success. In the era where energy resources are failing to provide the day to day need of mankind, it's highly essential to save whatever energy we have. In households, lots of energy is wasted just because of human ignorance, power saver servant serves the need of time and by saving energy, not only helps to fight the energy crisis but also saves you some more money.

Power saver servant is a smart robot which can move around your house and detect wherever the energy is being wasted. Any appliance which is left on while not in use, this robot will smartly turn it off and also informs you of it! It will also act as a surveillance robot, you can remotely check what all is happening in your house, over your android phone! It is a genuine step towards future where your robot servant does all your tasks.

1.3 References

1. **E-Yantra Website**
http://www.e-yantra.org/

This website will provide us with great help by providing us the API's for various parts of our project.

2.**ATMega16 datasheet**

3. **Android Developer's Guide**

1.4 Definitions, Acronyms & Abbreviations

Term Definitions:

Bot	The electro-mechanical machine that is guided by the user/ or by predefined algorithms
User	Owner of the robot who controls the bot , and commands the bot to perform tasks
Smart socket	Electronic device which is inserted into normal electricity socket. All gadgets which were meant to be inserted in the socket are inserted in this (fig. 3 for details).
Light sensor	Sensors which are used to detect whether lights in any room are on or off (by detecting the luminance in any room). (mounted on the bot)(fig 4)
Fan sensor (wind sensor)	Sensors used to detect whether fans in any room are on or not. (mounted on the bot)(fig. 1 for details)
Motion sensor	Sensors mounted on bot which is used to detect presence of person in any particular room. (fig. 5)
Camera	Camera of an android phone which is mounted on the bot.
Controller	It's an android application installed on user's phone.
AVR Studios	IDE for c programming for the robot.
SDK	SDK Software Development Kit
Android Phone	Phone having Android Operating system designed by Google corps. (fig. 2)

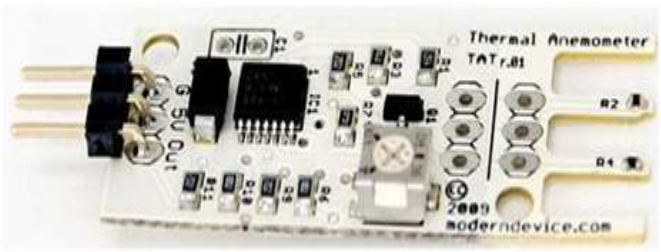


Figure1. Wind sensor

Figure2. Android Phone

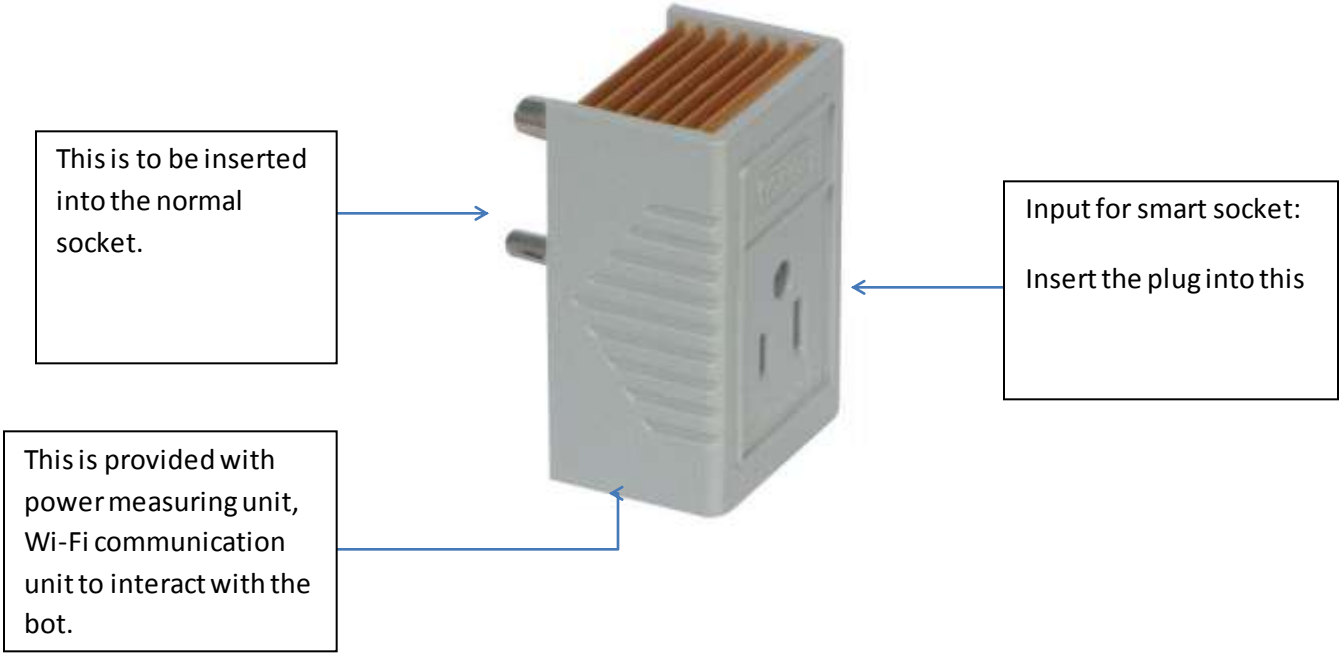


Figure3. Smart socket

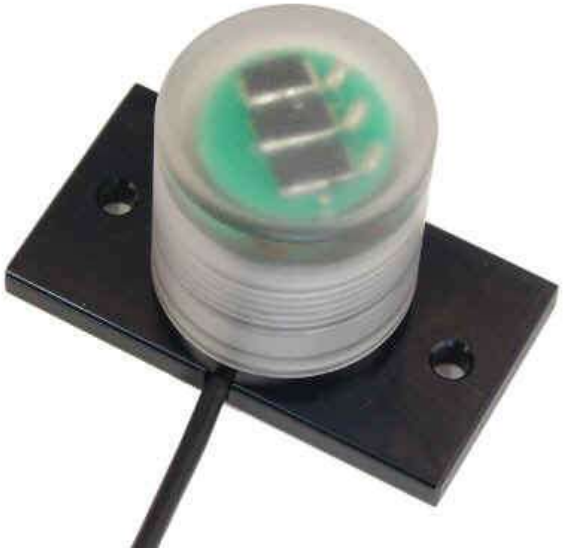


Figure4. Light Sensor

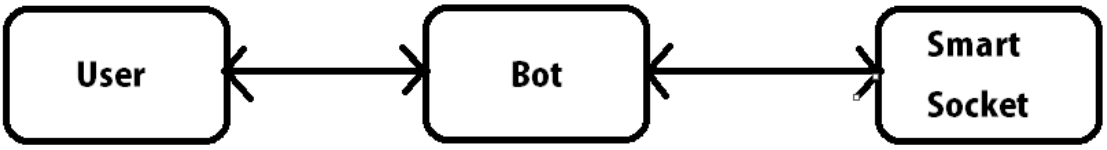
Figure5. Motion Sensor

Chapter 2

Overall Description

2.1 System Environment

The system environment for the Power saver servant consists of three parts.



1. Bot: This is an electro mechanical device with sensors, which travels to each and every room in given workspace, checks for the human presence, if not detected, takes appropriate actions. Bot follows following algorithm.

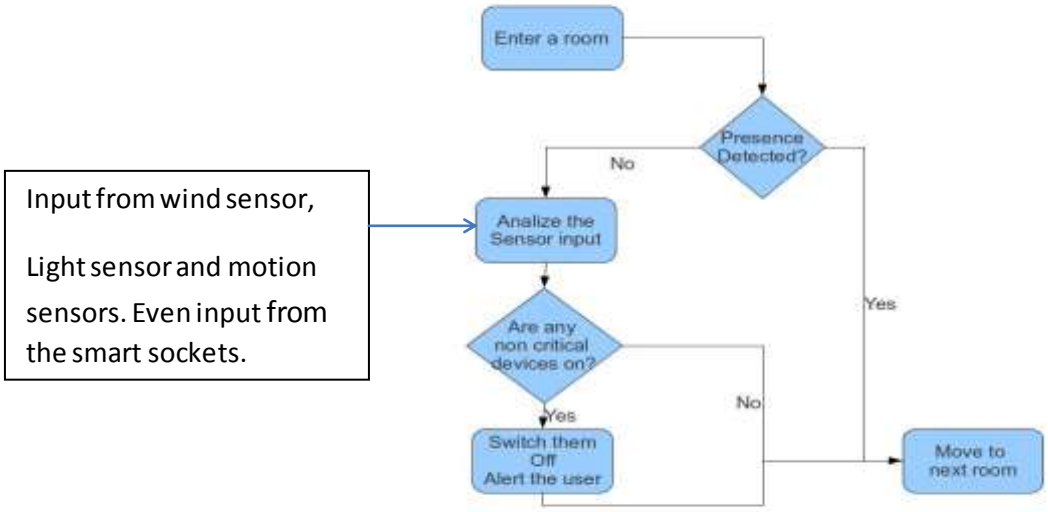


Figure6. Algorithm for bot

2. Smart socket: Smart socket communicates with the bot using Wi-Fi communication. Whenever a bot enters a room, and asks for its status , every switch replies with its current status and its priority (critical or non-critical). Bot decides what to do with the

corresponding switch, and orders its accordingly (whether to turn off or to do nothing). This device can also calculate information on total power consumed through it (either by counting total "ON" time or by using digital power measuring circuit)

3. User Interface: It is an android application on user's cell phone, which is used to command or communicate with the PSS. User can order PSS to check the house once , or to keep checking it , or even to survey around (by the use of camera of android phone mounted on PSS). The user is indicated of any power wastage or any unexpected movement on the interface, by the bot.

2.2 Product Perspective

The robot (Firebird V) is controlled remotely using the Android mobile application interface. The user is providing the input by using the user interface, like one time check , frequent check , video surveillance . The robot moves as per the user inputs in a predefined path of given workspace. In order to provide an option of making the system completely remote control (the user and the robot are not inline of sight of each other), a camera will be mounted on the robot to allow the user to view the environment of the robot.

On every round, based on a sensors' input, the bot will take appropriate action and inform the user about it. The user can decide on further actions depending on the bot's information via interface.

The bot uses light and breeze sensors to detect whether tubes/lights or fans are "ON" in the given room. It also reads every smart socket's status in the room. These serve as an input to the bot for making any sort of a decision.

2.3 Product Function

Here is a step wise description of our product's functioning.

Step 1: User commands the bot for any particular service.

Step 2: Bot takes the command and follows it.

Step 3: Bot travels room to room and for every room checks all inputs.

Step 4: Bot processes the input and takes appropriate action.

Step 5: Notifies the user about the actions taken and/or any unexpected presence is detected.

Step 6: Further awaits any command from the user.

2.4 User characteristics:

User must be friendly with the user interface. User must provide the bot with appropriate commands.

2.5 Constrains

- **Software Limitations:**
  The soft-wares AVR Studio, AVR Bootloader (developed by Atmel) are available exclusively for Windows platform thus imposing restriction on the product developed.

  Smart phones supporting android operating systems are required in order to make communication possible.

- **Hardware Limitations:**
  The given android phone must have specific minimal features on it (like camera) . They must be charged in order to make proper working.

  The bot must not be deviated from its predefined path as our implementation only works on the predefined path. (It can though be optimised for autonomous path traversing.)

2.6Assumptions

- The sensors are assumed to be calibrated to the luminance in the given environment.
- GSM network coverage for interactions. And GPRS connection (for video transmission)
- The occupancy detection must work accurately as no motion may lead to false information.
- The communication between smart sockets and bot (wireless Wi-Fi communication) must be flawless and two way communication is possible.
- There is no interference of any sort which may lead to falsified information to the bot. leading to unwanted actions.
- Smart sockets are assumed to be readymade (hardware issue).

2.7 Requirement Subsets

- Motion sensing
- Android application development
- Light sensors, breeze sensors calibration.
- Wireless communication (zigbee communication)
- Timers

- Interrupts
- Transmission of Commands
- Interpretation of Commands
- Signal generation based on Commands

Chapter 3

Details

3.1 Functionality

This bot is a smaller subset of household servant bot. this bot uses sensors (just like human senses) to detect light , breeze , motion; processes them to comes up with appropriate predefined action. This bot is made with an ambition to minimize the wastage of electricity in households and also save user's money. This product is a smart investment.

3.2 Supportability

3.2.1 Basis for future work

This project gives us an insight of the future of the home appliance systems. This version can be taken upon and can be used to make home appliance use very smart and secure. Future can be seen as when Wi-Fi reaches each and every house, each of the appliances can be controlled with your cell phone being its Remote! There are endless things we can do over here, but the main service it gives you is Energy management. This product can help us calculate the electricity bills for transparency, manage our consumption. This project is a mere start to something new which can be seen in every household in near future.

3.2.2 Distinguished modules of the project

Various modules can be used and implemented for various purposes.

- Motion sensing module
- Light , breeze sensing module
- Wireless communication module (for smart sockets).
- Android development module.
- Timer and interrupt processing modules.

3.2.3 Documentation

Documentation in terms of embedded comments and other explicit write-ups will help others in upcoming future.

Documentation regarding android applications can be found on android developer's community or it will also be shared through git-hub.

3.3 Interfaces

3.3.1 User Interface

This will be an android application on user's Android cell phone. It will be provided with all appropriate commands. It will also have an notification interface which will show all the actions of the bot. It will also have video interface to live stream from the camera mounted on the bot.

3.4.2 Hardware Interfaces

- Android phone: (2 nos.) 1 for user and one for mounting on the bot.
- Firebird robot
- Light, breeze, motion sensors.
- Smart sockets

3.4.3 Software Interfaces

This section states the requirements to development the application which will use the data feed from the various interfaces and use them to control the panel or display information for the user.They are as follows:

- Windows Operating System (XP, Vista, 7)
- Atmel AVR Studio
- AVR Bootloader
- Android Operating Systems (Optional)

3.4.4 Communications Interfaces

This section states the requirements to create a communication between user and the panel. They are as follows:

- 2 ZigBee Wireless Modules
- GSM/GPRS communication. (for android applications)

Chapter 4

Quality Control

1) **Accurate occupancy detection:**

   Motion sensors must be accurate to detect any presence in the room. They must cover whole room and natural motions must be avoided. Same goes with the luminance and breeze detection.

2) **Accurate determination of the states:**

   The smart switch must be able to keep track of its state and must be able to communicate it to the bot when it asks for the same.

3) **Accurate processing of command:**

   The bot must process all commands and all the interrupts and timers must be treated according to process priorities. And system must always remain in a consistent state.

Chapter 5

Risk Management

1) Communication failure:

There are two possibilities of communication failure.

i. User to bot (or vice-versa): This can be caused if there is no GSM coverage or the android phone's battery wares off. In such a scenario bot will take any predefined action (say check once) and go to its starting position, and wait for the communication to be restored. This will be pre-programmed on the bot.

ii. Smart-socket to bot: It can be caused either because of smart socket's failure, or any kind of failure in Wi-Fi communication. In such scenario, the bot will not take into account those sockets which fail to respond. And will only take actions on those which are working properly. It will also notify the user about the failure.

2) Power Failure:

When bot is about to run out of power, it will go to its starting position (we can also implement technology similar to that of Roomba, so that it can charge itself at a charging dock), and hold there till it is charged.

3) Path Failure

If the bot goes off the course, it will try to find its original path, failing to do so, it will notify the user of it and it will stay idle over there. The path finding algorithm can be programmed on the bot. We can also provide manual control (using the video and the android application) to the user, so that (s)he can manure the robot to any designated position, from where robot can again retrace its original path.