

# Smart Home Using Hardware Interfacing

12BCE034

September 3, 2014

## 1 Introduction



Home automation is the residential extension of building automation. It is automation of the home, housework or household activity. Home automation may include centralized control of lighting, HVAC (heating, ventilation and air conditioning), appliances, security locks of gates and doors and other systems, to provide improved convenience, comfort, energy efficiency and security. The popularity of home automation has been increasing greatly in recent years due to much higher affordability and simplicity through smartphone and tablet connectivity. The concept of the "Internet of Things" has tied in closely with the popularization of home automation. A home automation system integrates electrical devices in a house with each other. The techniques employed in home automation include those in building automation as well as the control of domestic activities, such as home entertainment systems, houseplant and yard watering, pet feeding, changing the ambiance "scenes" for different events (such as dinners or parties), and the use of domestic robots.

## **2 Interfaces**

### **2.1 front-end**

1. Mobile App
2. Tablet App
3. Browser Based App
4. App for Wearable

### **2.2 back-end**

1. Lighting
2. Locks, Doors, Windows
3. Curtains, Blinds
4. Surveillance
5. Multimedia and Audio Control

### **2.3 Technical Approach**

Programs sequence:

1. Switching on an LED using raspberry Pi
2. Switching on a light bulb using Raspberry Pi and relay
3. Making a LED blink using Arduino
4. Making a push button debounce to control LED using Arduino
5. Reading status of remote Xbee, to check whether door is open or closed, using Arduino
6. Sending signal to remote Xbee, to make an LED blink at every 5 seconds
7. Turn on/off a light bulb using arduino and relay
8. Make a push button debounce to control light bulb using arduino
9. Remotely control a light bulb/LED using Xbee: Push button on coordinator end and LED on the remote end.
10. Serial communication using Pi
11. Reading and writing LED status into file and accessing it using GUI.

To Cover Next:

1. Designing a DB schema
2. Deciding a general approach for the service, processes and threads
3. Making a fully functional light switch

## 3 Stages

### 3.1 Toggle Light Bulb Using R-Pi

### 3.2 Bulb Operating using Push Button with Arduino and Raspberry Pi

1. Toggle Bulb/Led using Arduino (Microcontroller)
2. Interfacing between Arduino (MCU) and R-Pi (GPU)
3. Toggle Bulb/Led using push button with R-Pi

### 3.3 Bulb operating wirelessly using ZigBee, Arduino and Raspberry pi

1. Interfacing between ZigBee and Arduino (MCU)
2. Remote LED operating using ZigBee and Arduino
3. Remote Bulb operating using Relay, ZigBee and Arduino
4. Remote Bulb operating wirelessly using Arduino and Raspberry pi

### 3.4 Combining Wireless and Pushbutton and Circuit Design for Wireless relay switch

### 3.5 ZigBee Range Testing and Mesh Topology experiment

### 3.6 DIY Voltage Regulated Adapter for 3.3, 5, 12 V

## 4 Equipments Used

### 4.1 Cables

Jumper Cables: Male to Male, Male to Female, and Female to Female

### 4.2 Adapters

**xBee Adapters** - Two 10 pin 2.0mm pitch female header to solder, One 20 pin 2.54mm pitch SIP male header (break it to two 10 pin header) to solder, 4 status indicator LEDs for Power, RSSI, Associate and mode (sleep/ON), Power requirements: 5.0V from USB or VDD pin, 3.3V generated on-board (3.3V regulator).

**Transformers** 0-12V and 500mA

**Voltage Regulator IC** 3.3V, 5V, 12V

Fuse and Socket, Soldering Iron and Wire, Round pin adapter converter

### 4.3 Processing and Controlling Units

**Raspberry Pi** 700 MHz Microprocessor and it's I/O interfaces/pins

**Arduino** Microcontroller and it's i/o pins

**IC ATmega328p** - Microcontroller IC

**Boolean Gates** Logic Gates: And, Or, Not, Nand, Nor, Xor, etc.

Microcontroller base for breadboard

### 4.4 Boards

- 4 Channel 12v Active High Relay
- 8 Channel 5v Active Low Relay
- xBee Breakout Board

### 4.5 General Electronics to build Arduino DIY

- 2 - 220 Ohm resistors
- 1 - 10k Ohm resistor
- 2 - 10 uF capacitors
- 16 MHz clock crystal
- 2 - 22 pF capacitors
- Tactile Switches

## 5 Tools

**Arduino IDE** <http://arduino.cc/en/Main/Software>

**Fritzing** [fritzing.org](http://fritzing.org)

**Processing** <http://processing.org/>