

IPv6 Enabled Smart Home Using Arduino

Mahesh Patil¹, Sumathy Subramaniam² and Rajendra Hegadi³

¹Walchand Institute of Technology, Solapur, Maharashtra, India

²School of Information Technology and Engineering, VIT University, Vellore, India

³Indian Institute of Information Technology, Dharwad, India

Abstract—Internet of Things (IoT) is a widely emerging technology making our lives smarter with smart applications. One of its applications is smart home where any device or object at home can be made smart and can be controlled via internet or in a local network. All recent projects in smart home are using IPv4 for addressing, as we know IPv4 addresses are getting depleted at faster rate as there is increase in connected devices. This problem of IPv4 depletion can be solved by using IPv6 protocol for addressing. IPv6 protocol stack can't be embedded directly into the microcontroller as microcontroller has limited hardware capability to run existing IPv6 protocol stack. This paper proposes a solution to build a smart control centre with Arduino yun to facilitate IPv6 addressing then using this smart control centre to control devices over wifi interface through web browser on laptops, desktops and smart phones.

Keywords—*arduino; arduino yun; IPv6; IoT; microcontroller; openWRT; smart home*

I. INTRODUCTION

It's good to hear that all our devices at home can be made smart resulting in smart home. Our daily life now can be made easier by adding microcontrollers to devices for machine to machine communication. This communication can be achieved by using internet protocol where we can successfully provide an IP address to each device and this IP addressing makes the communication easier. Any connected device in the internet can be controlled and communicated with the help of IP address. There are two types of IP addresses viz. IPv4 and IPv6 where IPv4 is still in use. Very soon there will be problem of shortage of IPv4 addresses as it is only 32 bit addressing system with 4.3 billion no of addresses possible. Pool of IPv4 address is getting depleted at faster rate as there is increase in connected devices in the internet [3]. So to solve this issue, successor of IPv4 i.e. IPv6 protocol with 128 bit addressing must be used which supports 3.4×10^{38} network addresses. APNIC has already advised to organizations to start IPv6 development process now to avoid issues with running out of IPv4 space [6]. There are lots of ongoing projects in IoT where people use different kind of microcontrollers suitable for their projects. One of the trending microcontroller based board is Arduino. A question arises in mind that why Arduino when there are many such kind of projects which can be used in IoT. Raspberry Pi is also trending nowadays in the applications of IoT. Raspberry Pi board has a good hardware configuration where Raspberry Pi 2 recently launched has 900MHz quad core CPU with 1 GB ram and Linux operating system running on it. It can be called a microprocessor rather than microcontroller and microprocessors are costlier than microcontrollers. Raspberry Pi 2 can be used in such situations where there is requirement of lot of processing

rather than controlling, and to build a smart home for controlling things needs microcontrollers.

Arduino is an open source platform founded by the Arduino community. There are few Arduino based microcontrollers which can be used in the applications of IoT. Arduino boards are cheap in cost and have limited hardware capability. Most of the Arduino boards use 8-bit microcontroller with 32 KB of flash memory, with such a limited hardware it's difficult to use existing IPv6 protocol stack with these Arduino boards as the flash memory and RAM is not sufficient to run existing IPv6 protocol stack. A solution to this problem can be addressed using Arduino yun.

II. RELATED WORK

Researchers are contributing a lot towards the applications of IoT in recent days. Bernd Michael Dörge [1] has built an automatic door locking system using an Atmel raven board which has 8-bit microcontroller with 16KB ram and 128KB flash, this system works with IPv6 over low power wireless personal area network (6LOWPAN) and an web based interaction for controlling. Here author has achieved to port the uIPv6 stack of contiki os in Atmel raven board to make it IPv6 enabled door locking system. The three authors [2] have proposed a safe building system to escape from the building in emergency situations. To make this system IPv6 enabled they have ported the uIPv6 stack of contiki os to Octopus II sensor device and this sensor device can be accessed using their information management system. Home appliances can also be controlled by developing a smart control centre [5] which operates at 433 MHz frequency of wireless sensor and actuator network to connect sensor devices and electrical switches. This smart control centre uses ARM cortex- M3 32 bit RISC microcontroller operating at 24MHz frequency and can be operated over wifi networks using mobile phones or computers. Data collected by sensors are displayed on smart home information system from where devices can be operated.

Using the base idea of internet of things there are many protocols developed for system application and one of them is KNX protocol. A KNX based smart home was proposed [8] to implement home automation over the internet and on local network. In this proposal authors used ETS software to construct a model home and they implanted KNX based devices in each room to monitor temperature, lighting control and power consumption, from this software a JSON file is configured and sent to android device for controlling on android user interface.

Apart from directly implementing the modules on hardware, they can be virtualized and demonstrated on

network. The idea of IOT gadget control was proposed [9] to work on network virtually; here instances of objects were virtualized based on objects of java script and deploying them on the network, there is central access point running OpenWRT on it. Smart home gadgets are connected to this access point and are accessed through web management portal.

III. SYSTEM ARCHITECTURE

Smart Home devices can be controlled via a control panel by the user on a mobile phone or an android tablet; these devices will be connected to the internet or to the local network available at home. Arduino microcontroller will be using either wifi interface or Ethernet to connect to local network. Objects or devices and sensors in the home will be connected to Arduino yun.

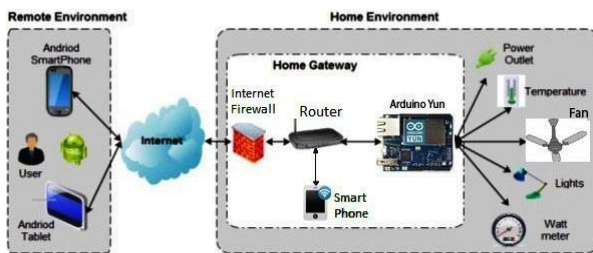


FIGURE I. SMART HOME SYSTEM ARCHITECTURE

Firewall settings can be configured on both home router and on Arduino yun which will be additional advantage of having 2 level securities in this design. If the smart home controller is connected to local home network then internet connectivity is not required, internet connectivity is only required when the user will move out of home and will to control home devices remotely.

Delay in the control operation performed will depend on what kind of network user is adopting, if user is connected in local home network then operations are performed quickly.

A. Hardware Components

Hardware components in this smart home are divided in three parts viz. Arduino yun, sensor modules, and output components like LED. To implement IPv6 Arduino yun is used; Arduino yun is developed by the Arduino community. Yun is a small sized microcontroller based on Arduino platform; it has on board wifi, SD card slot, Ethernet controller which is all connected to a microprocessor running Linux based OpenWRT operating system on it. Below figure shows the features of Yun board.

It has a second processor which is actually a microcontroller providing 20 input and output pins including pulse width modulation pins. This microcontroller communicates to Linux based processor via a bridge library and it supports python distribution. It uses embedded c for programming the board and these programs are written in Arduino IDE. Yun has 400 MHz Linux processor on board with 64MB DDR2 RAM and 16MB of flash memory featuring OpenWRT installation with card reader, USB host controller along with Ethernet and Wifi interface. There is an 8 bit microcontroller with 32KB of flash memory connecting

all input and output pins.

Relays are used to switch the high power devices where these relays are controlled through the digital pins of Arduino. A 5v relay module must be used for Arduino to work. A relay can be called as an electromagnetic switch; following is the image displaying a 4 channel 5v relay module for Arduino, it has onboard optocoupler which divides the whole circuit and ensures the microcontroller safety.

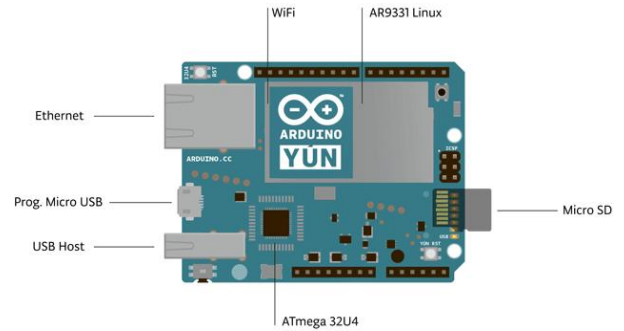


FIGURE II. ARDUINO YUN FEATURES

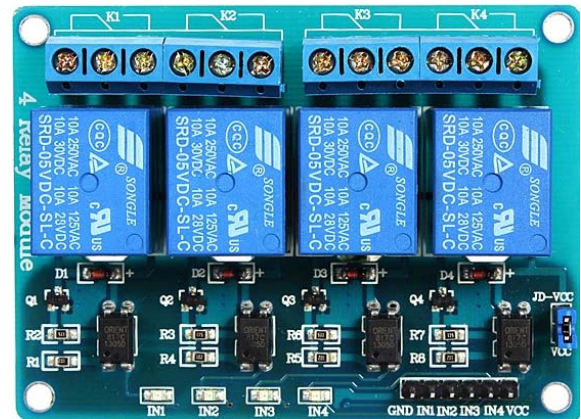


FIGURE III. RELAY MODULE (4 CHANNEL)

LM35 temperature sensor is used to detect the temperature of the room; it has capability to measure the temperature accurately and gives the output temperature in the form of voltage proportional to Celsius. LM35 accepts the input voltage between 4volts to 30 volts as Vcc but commonly used are 5v and 12v.

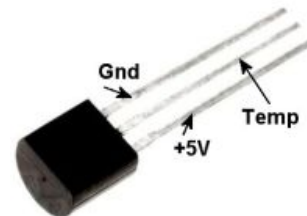


FIGURE IV. LM35 TEMPERATURE SENSOR

Devices can be switched on or off based on motion detection by using PIR motion sensor attached to the Arduino. An individual PIR sensor detects changes in the amount of infrared radiation falling upon it, which varies depending on

the temperature and surface characteristics of the objects in front of the sensor. When an object, such as a human, passes in front of the background, such as a wall, the temperature at that point in the sensor's field of view will rise from room temperature to body temperature, and then back again. The sensor converts the resulting change in the incoming infrared radiation into a change in the output voltage, and this triggers the detection. Objects of similar temperature but different surface characteristics may also have a different infrared emission pattern, and thus moving them with respect to the background may trigger the detector as well.



FIGURE V. PASSIVE INFRARED MOTION SENSOR

Light dependent resistor (LDR) sensor is used to detect the amount of brightness present in the room and trigger the microcontroller on the level of intensity; it has both analog and digital outputs, works between 3.3v to 6v.



FIGURE VI. LIGHT DEPENDENT RESISTOR

B. Arduino Software Introduction

1) OpenWRT

It is an Linux based embedded operating system to route the traffic on network, it is installed in embedded devices like routers. This operating system can be interfaced using command line over secure shell interface called SSH or web based interface. It provides package management system; around 3500 software packages are available for installation. A specific version of OpenWRT called OpenWRT Attitude Adjustment 1 is installed in Arduino yun.

2) Arduino Integrated Development Environment

It is open source software basically written in java makes the programmer easy to write the code in embedded c language and upload the code to board in one click. Codes written in this software are called sketches. It has a text area to write the sketches, menu bar, message area, and a tool bar

with button for common functions.

IV. PROPOSED SYSTEM

Arduino yun has OpenWRT installed on it and IPv6 can be configured easily on this OpenWRT operating system through installing and upgrading packages. All the smart home modules will be implemented on Arduino yun, using Arduino yun all the objects or devices at home will be controlled over a Smartphone or desktop or laptop system.

Smart home modules will accessed through a web based interface using the front end HTML5 and PHP scripting, and the control signals will be sent to Arduino board through PHP curl. Both IPv4 and IPv6 will be used to address the Arduino yun, there will be necessary configurations will be done in the OpenWRT Arduino yun to enable the IPv6 addressing capability. This Arduino yun will be connected to the home router and to rest of the internet.

Home devices can be controlled within a home network or through the internet. A PHP server will be running either on Arduino yun or on the desktop system. Web pages can be accessed through this server. Small capacity devices will be directly connected to Arduino yun for demonstration.

A. Smart Home Modules

1) Switching Devices

Any device can be switched on or off over Arduino yun which will be connected through wifi. Relays are used to switch the high power devices where these relays are controlled through the digital pins of Arduino. A 5v relay module must be used for Arduino to work. A relay can be called as an electromagnetic switch; following is the image displaying a 4 channel 5v relay module for Arduino, it has onboard optocoupler which divides the whole circuit and ensures the microcontroller safety.

There are three kinds of output points on a relay normally open (NO), normally closed (NC) and common. The common terminal is always consistent; switching is done between normally open and normally closed. At the input side VCC is the pin where 5v supply is provided and GND pin is connected to ground. IN1-IN4 or IN1 – IN8 are the input pins which are connected to the digital pins of Arduino.

2) Manual Light Brightness Control

Instead of having different LED bulbs for different kinds of brightness it is better to control the brightness of same LED bulb by using PWM output of Arduino.

Pulse Width Modulation, or PWM, is a technique for getting analog results with digital means. Digital control is used to create a square wave, a signal switched between on and off. This on-off pattern can simulate voltages in between full on (5 Volts) and off (0 Volts) by changing the portion of the time the signal spends on versus the time that the signal spends off. The duration of "on time" is called the pulse width. To get varying analog values, you change, or modulate, that pulse width. If you repeat this on-off pattern fast enough with an LED for example, the result is as if the signal is a steady voltage between 0 and 5v controlling the brightness of the

LED.

3) *Manual Fan Speed Control*

Now, it's also possible to control the speed of fan by using PWM output of Arduino. Electrical fans work on high voltage so these fans need IGBT to drive them. User can adjust variable fan speeds from 0 to 100% depending on the requirement. In this module fan speed cannot be demonstrated because the output current of Arduino board is too low (40 mA) to operate a small mini fan; this module will be demonstrated using an LED instead of fan.

4) *Adaptive Fan Speed Control*

It is possible to change the speed of fan adaptively where speed of fan changes according to the change in the temperature of room. LM35 temperature sensor is used to detect the temperature of the room; it has capability to measure the temperature accurately and gives the output temperature in the form of voltage proportional to Celsius. LM35 accepts the input voltage between 4volts to 30 volts as Vcc but commonly used are 5v and 12v. In this module adaptive fan speeds are demonstrated using LED instead of fan due to low current on the output terminal which causes improper operation, using LED difference can be visualized.

5) *Adaptive Light Brightness Control*

In this module an LDR (light dependent resistor) sensor is used to detect the amount of brightness present in the room. Using the value of brightness the function auto adjusts the brightness of the light operated through Arduino. This adaptive nature always keeps the room at perfect brightness.

Arduino gets the brightness input from light LDR sensor and passes this input to the desired function. LDR is connected to analog pin of Arduino. This sensing of brightness will be in loop.

6) *Motion Based Switching*

Now, the devices can be switched on or off based on motion detection by using PIR motion sensor attached to the Arduino. An individual PIR sensor detects changes in the amount of infrared radiation falling upon it, which varies depending on the temperature and surface characteristics of the objects in front of the sensor. When an object, such as a human, passes in front of the background, such as a wall, the temperature at that point in the sensor's field of view will rise from room temperature to body temperature, and then back again. The sensor converts the resulting change in the incoming infrared radiation into a change in the output voltage, and this triggers the detection. Objects of similar temperature but different surface characteristics may also have a different infrared emission pattern, and thus moving them with respect to the background may trigger the detector as well.

V. SYSTEM IMPLEMENTATION

This project implements IPv4 and IPv6 addressing on Arduino yun over OpenWRT operating system. Network related operations are implemented on Linux based processor and other project modules are implemented on

microcontroller.

A. *IPv6 Configuration*

Arduino yun has OpenWRT attitude adjustment 1 installed on it which is custom version of OpenWRT specially targeted for Arduino yun.

Packages necessary to be installed:

1. ip6tables
2. kmod-ip6tables
3. radvd
4. ipv6-support

To provide native IPv6 address one must get IPv6 from ISP. DHCP also can be configured using the package dhcpv6.

/etc/config/network file must be edited in order to configure IPv6 on yun.

After editing the file in VI editor save that file and the network must be restarted.

/etc/init.d/network restart

To configure firewall: go to directory /etc/config and edit firewall file.

To check whether the firewall settings are correctly configured execute the following command:

cat /proc/sys/net/ipv6/conf/all/forwarding

Output of the above command should be 1, then it is verified that firewall is correctly configured.

B. *Circuit Diagrams for Module Connections*

An overall system circuit diagram can be proposed but to test modules separate circuit diagram is required for all modules so each module will be implemented separately.

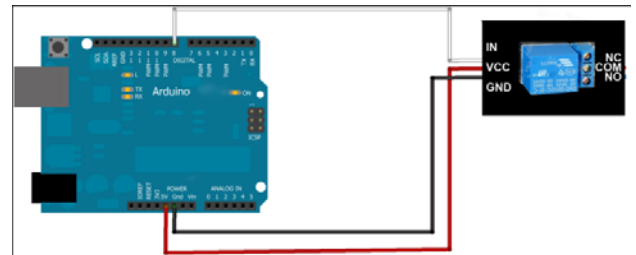


FIGURE VII. WIRING DIAGRAM FOR SWITCHING DEVICES

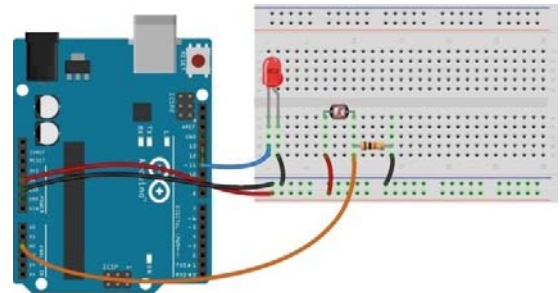


FIGURE VIII. WIRING FOR ADAPTIVE LIGHT BRIGHTNESS

Figure XIV displays the output of relay switching connected to Arduino yun. 8 input pins of relay are connected to 8 digital output pins of yun. Enlighten red light on yun ensures that bridge is connected between microcontroller and microprocessor and enlighten white light on yun ensures that wifi is on. After pressing all switch-on buttons on user interface, all the relay pins got active and relay pins were switched on, this demonstrates that relays are operationally working correctly with yun and there was successful communication between user interface and Arduino yun.



FIGURE XV. MANUAL LIGHT BRIGHTNESS CONTROL

Figure XV shows the output of manually controlling the light brightness from user interface. Figure XVI shows adaptive nature of fan speed. Real time input of temperature is taken from temperature sensor. Rotation of fan cannot be displayed in still image so in above figure LED is used to demonstrate adaptive nature; LED brightness is adjusted according to the input temperature, to see immediate visible results LM35 is kept under fire for sudden increase in temperature and LED brightness changes according to change in temperature.

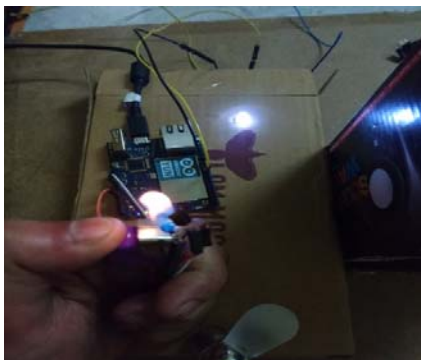


FIGURE XVI. ADAPTIVE FAN SPEED CONTROL

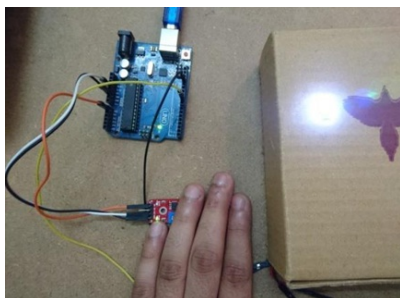


FIGURE XVII. LDR SENSOR WITH ZERO BRIGHTNESS

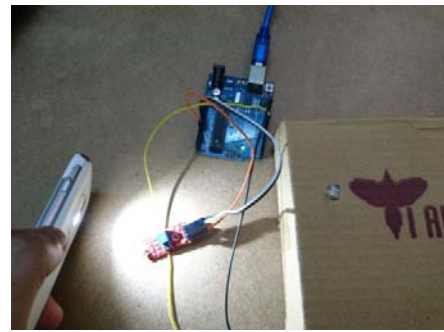


FIGURE XVIII. LDR SENSOR WITH FULL BRIGHTNESS

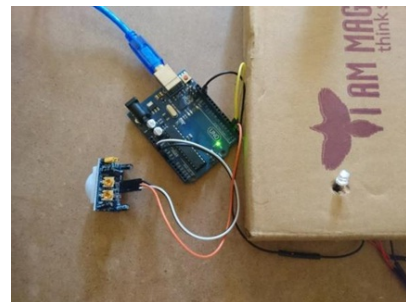


FIGURE XIX. PASSIVE INFRARED SENSOR WITH NO MOTION

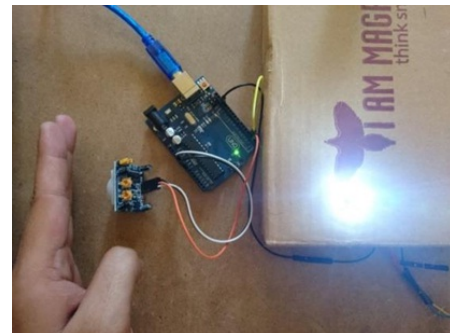


FIGURE XX. PIR SENSOR WITH SOME MOTION

VII. CONCLUSION

Need of IPv6 is very essential in the fast growing connected device environment, it's now possible to analyze a smart home network with IPv6 addressing capability using Arduino yun, IPv6 ping statistics showed us that an IPv6 smart home can be built and devices can be controlled. Smart home modules were successfully implemented on Arduino yun and the results were analyzed. Now a traditional IPv4 based smart home can be replaced by wifi enabled IPv6 smart home.

VIII. FUTURE WORK

Currently objects are controlled via web based user interface on all devices; later dedicated android application can be developed for smart phone. Adaptive modules like adaptive light brightness and adaptive fan speed can be notified to user by hardware using beep sound or by software displaying and notifying the user about changed values. Power consumption monitoring module can be added and captured power consumption values can be uploaded to cloud for

analysis. Power management can be done by automatically switching of devices when not in use.

ACKNOWLEDGEMENT

Images of the Arduino board in this paper are sourced from the Arduino official website and images of the sensors are sourced from amazon.in

REFERENCES

- [1] Bernd Michael Dörge, Thomas Scheffler "Using IPv6 and 6LoWPAN for Home Automation Networks", 2011 IEEE International Conference on Consumer Electronics - Berlin (ICCE-Berlin)
- [2] Tseng-Yi Chen, *Hsin-Wen Wei, Nien-I Hsu, Wei-Kuan Shih, "A IoT Application of Safe Building in IPv6 Network Environment", 2013 IEEE 37th Annual Computer Software and Applications Conference
- [3] S. Deering and R. Hinden, "Internet Protocol, version 6 (IPv6) specification," RFC 2460, 1998
- [4] Arduino official website: www.arduino.cc accessed on 10th January 2016
- [5] Ming Wang, Guiqing Zhang, Chenghui Zhang, Jianbin Zhang, Chengdong Li, "An IoT-based Appliance Control System for Smart Homes", 2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP) June 9 – 11, 2013, Beijing, China
- [6] <https://www.apnic.net/community/ipv4-exhaustion/exhaustion-and-network-operators>
- [7] <https://www.apnic.net/community/ipv4exhaustion/ipv4-exhaustion-details>
- [8] Gabriele De Luca, Paolo Lillo, Luca Mainetti, Vincenzo Mighali, Luigi Patrono and Ilaria Sergi "The use of NFC and Android technologies to enable a KNX-based Smart Home"
- [9] Jinhong Yang, Hyojin Park, Yongrok Kim, Jun Kyun Choi "IoT Gadget Control on Wireless AP at Home" The 11th Annual IEEE Consumer Communications and Networking Conference 978-1-4799-2355-7/14
- [10] Ming Wang, Guiqing Zhang, Chenghui Zhang, Jianbin Zhang and Chengdong Li "An IoT-based Appliance Control System for Smart Homes" 2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP) June 9 – 11, 2013, Beijing, China.
- [11] Y. Zhang, L. Ye, L. Zhu, Y. Lai, "A Solution for Low Cost and High Performance Smart Home Networking Smart Home Networking", in Proceedings of 2011 International Conference on Engineering and Industries, pp. 1-6, 2011.
- [12] B.-K. Kim, S.-H. Hong, Y.-S. Jeong, and S.-S. Eom, "The study of applying sensor networks to a smart home", in Proceedings of Fourth International Conference on Networked Computing and Advanced Information Management, pp. 676-681, 2008.
- [13] Qifen Dong, Li Yu, Huanjia Lu, Zhen Hong, Yourong Chen, "Design of Building Monitoring Systems Based on Wireless Sensor Networks," Wireless Sensor Network, Vol.2 No.9, PP.703-709, September 2010
- [14] OpenWRT official website: www.openwrt.org accessed on 15th January 2016.