Advances in Robotics & Automation
ISSN: 2168-9695

Jalamkar and Selvakumar, Adv Robot Autom 2016, 5:2
DOI: 10.4172/2168-9695.1000149

Research Article | Open Access

Use of Internet of Things in a Humanoid Robot – A Review

Jalamkar D and Selvakumar AA*

School of Mechanical and Building sciences, VIT University, Chennai, India

Abstract

Internet of things is becoming the most growing technology in recent days. Main idea behind the IoT is to extract the various values from various sensors which are attached to various objects by connecting them to the network and automating the actions performed by the object or a system. In this review paper, a study is made to understand the importance of use of IoT in humanoid robots. In order to appreciate this growing technology, this technology should be implemented to as many objects as possible so that everything will get connected in future. Along with other hardware developments, 5G internet technology is also getting developed rapidly. Use of IoT in humanoid will make many things easier to monitor and control in many ways which is discussed further in this paper. Case studies are also involved in this study to get the brief idea about how the hardware development is being done in this area. Comparison of various hardwares available to make the system work is done considering the factors like cost, ease of development, simplicity, etc.

Keywords: Humanoid; IoT; Internet of things; Control and automation; 5G internet

Introduction

Internet of Things is all about connecting various devices by using internet and letting them communicate with each other and the remote server. Large amount of data transfer takes place between various things and the server [1]. So it also demands the continuous high speed internet connectivity. "With 5G technology, getting and staying connected will get easier", said Aicha Evans, Intel's Corporate Vice President and General Manager of the Communication and Devices Group [2]. After introduction of 5G, around 50 billion things will be connected and IoT will become more efficient, faster and effective [2]. Even today with 3G and 4G internet, IoT is working because the connected devices are very less. According to IDC, worldwide market of internet of things is going to grow up around $1.7 trillion up to 2020. Other areas in which IoT will play a very important role is industrial plants and healthcare industry. IoT is being used to monitor the various parameters of the machineries in the industries which will help to predict the breakdowns. On other hand in healthcare industry, wearable tech devices are already here to keep database of body functionality [3]. Robotics is also the very much important area to consider for implementation of the IoT. Connecting various robots in the industry, remote places, homes, etc. to internet is also important. It is possible to monitor and control all the robots at different locations centrally. Current research is leading the robotics field to use the internet thus giving birth to the new term "Internet of Robotics". Role of IoT in the area of robotics is to ease the control and communication of human and robot. This can be classified into two types of robots such as Industrial Robots and Domestic robot like a humanoid robot [4]. This paper especially concentrates on using IoT in humanoid robots.

Motivation

There are several important reasons for developing a humanoid robot with IoT capability. Humanoid robots are mostly being developed as domestic robots which will stay in house, act like humans in house, do household tasks, etc. Implementation of IoT in humanoid will make this job a lot easier for the robot as well as for humans. Home security is the most important task that needs to be performed by the robot in absence of the house owner. Multiple tasks in this area like intruder detection, avoiding disastrous situations like fire, attending the visitor at door, etc. All these tasks can be monitored by the owner from anywhere in the world. Even when in home, the owner will be updated with all the surrounding parameters and situations with help of various sensors on the robot. With IoT implemented to the robot, one can control the robot and have access to all data on the mobile phone anytime anywhere. This makes it even easier with a dedicated mobile application which may be on android, ios, or any other operating system. IoT can be seen as revolutionary technology of future which have potential to make every job easy and almost everything accessible for human beings.

Available IoT Technologies

Smart phones and tablets are the most important and common tools that are required to communicate in the field of IoT.

Currently many hardware developments are taking place in this area. Some of them are Wi-Fi shields for various microprocessor or microcontrollers, RFID tags [5], NFC enabled devices, Ethernet Shields, etc. RFID are wireless microchips which are used to tag any object and to make it readable [6]. RFID reader is located centrally and objects are tagged by using RFID tags. Every tag contains some information which is read by the reader. Most of the readers are smartphones (Figure 1) or handheld devices which are connected to the internet for Iot applications [7]. Since RFID is a short range communication technology, it may not be suitable for all the applications like a humanoid application which is being discussed in this paper.

Ethernet shield (Figure 2) tacked on arduino uno microcontroller is also an example of hardware related to the IoT. Arduino is connected to the internet by using ethernet shield and other sensors and actuators

***Corresponding author:** Selvakumar AA, School of Mechanical and Building sciences, VIT University, Chennai, India, Tel: 0416 224 3091; E-mail: arockia.selvakumar@vit.ac.in

Received March 30, 2016; **Accepted** April 27, 2016; **Published** April 30, 2016

Citation: Jalamkar D, Selvakumar AA (2016) Use of Internet of Things in a Humanoid Robot – A Review. Adv Robot Autom 5: 149. doi:10.4172/2168-9695.1000149

Figure 1: RFID reader and RFID tag.

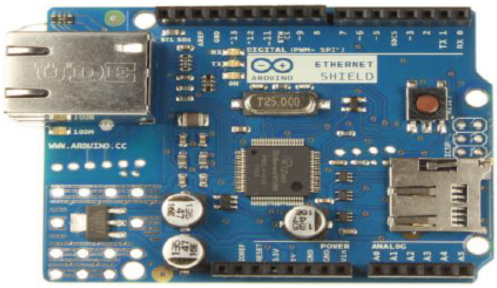

Figure 2: Ethernet shield.

connected to the arduino [8]. But disadvantage of this system is that it is not truly wireless. Ethernet shield needs to be connected by a cable to provide internet connection.

Most of the hardwares are compatible with arduino microcontroller. The Arduino is a very good and versatile microcontroller and development environment that can be used to control a variety of devices very effectively, it is also able to read many kinds of data from variety of sensors available today [9]. It is simple, cheap, easy to use and program and it has many software libraries and supporting extension hardware available ready to use. It is the open hardware platform for working and experimenting with many projects related to the Internet of Things.

ZigBee is a networking technology in wireless platform for data transmission. Zigbee module (Figure 3) is used along with the microcontroller for internet connectivity. It has advantages such as low power consumption, small and compact in size, moderate cost. A zigbee network which consists of cluster mesh and star is ideal for application of IoT [10]. Suggested structure of zigbee network can form a complex network which has further more advantages like self-healing.

Use of GSM shields (Figure 4) is also an option in parallel with above mentioned methods [11]. But with the use of GSM shield with microcontroller, there comes a problem with continuous network connectivity. If SIM card loses the signal which is very frequent and common problem, the system may stop working. As most of the GSM shields work with GPRS, bandwidth problem may occur where high bandwidth is required for specific applications. Another problem with GSM is costly internet plans with less validity.

This disadvantage of GSM can be avoided by making use of wired broadband internet connection to a wi-fi router and creating a hotspot for Wi-Fi module.

Wi-Fi module like esp. 8266 is recently developed hardware which is updated from version 1 to the latest version 14. This hardware is the most promising in terms of connecting anything to the internet. Various versions of esp. 8266 comes in different configurations for different applications. They vary in terms of number of GPIOs, cost, size, processing speed, memory, etc. [12]. This is very cheap module with system on chip. Esp8266 can work independently without connecting to any microcontroller or microprocessor. ESP 01 (Figure 5) is the most widely used module for small applications where less GPIOs are needed. This is cheapest version available today.

Figure 3: Zigbee module.

Figure 4: GSM shield.

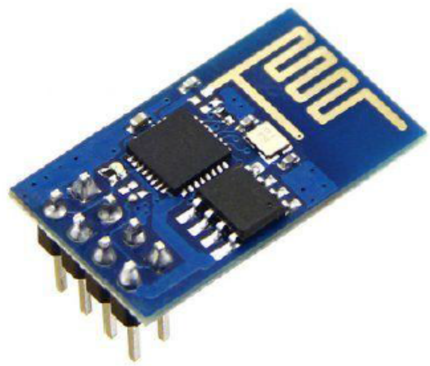

Figure 5: ESP8266 01.

ESP 12 is widely used where more GPIOs are needed. Developing IoT applications with ESP 12 is much simpler than ESP 01 because of many design improvements that are made lately (Figure 6).

Best advantage with this module is that multiple modules can be connected to each other in order to increase the number of GPIOs which is the need of a humanoid robot application. This way it is possible to connect large number of sensors and actuators for large applications. Among all the discussed hardware above, ESP8266 module family seems the most promising and cheapest option for IoT applications.

Table 1 shows the comparison between various versions of ESP8266 module. Selection of any version can be made as per the various requirements of GPIOs, cost, size, memory, speed, etc. [12].

Protocols for IoT

Standard protocols are required for the operation if IoT. Like internet protocols IoT also needs some protocols. MQTT and CoAP are two most promising protocols for IoT. Both protocols can be used for long range applications and both are open standard [13].

MQTT protocol

MQTT is a messaging protocol which stands for Message Queuing Telemetry Transport. This protocol is very good for remote location messaging. MQTT has the client/ server architecture (Figure 7) in which every sensor and actuator is connected to the server or broker. There are many brokers available for IoT applications with various feature sets. Messaging is done in the form of chunk data to the server.

Figure 6: ESP8266 12.

Board ID	pins	pitch	LEDs	Antenna	Dimensions mm
ESP-01	8	0.1"	Yes	Etched-on PCB	14.3 × 24.8
ESP-02	8	0.1"	No?	None	14.2 × 14.2
ESP-03	14	2 mm	No	Ceramic	17.3 × 12.1
ESP-04	14	2 mm	No?	None	14.7 × 12.1
ESP-05	5	0.1"	No	None	14.2 × 14.2
ESP-06	12+GND	misc	No	None	?
ESP-07	16	2 mm	Yes	Ceramic	20.0 × 16.0
ESP-08	14	2 mm	No	None	17.0 × 16.0
ESP-09	12+GND	misc	No	None	10.0 × 10.0
ESP-10	5	2 mm?	No	None	14.2 × 10.0
ESP-11	8	1.27 mm	No?	Ceramic	17.3 × 12.1
ESP-12	16	2 mm	Yes	Etched-on PCB	24.0 × 16.0
ESP-12-E	22	2 mm	Yes	Etched-on PCB	24.0 × 16.0
ESP-13	18	1.5 mm	?	Etched-on PCB	? × ?
ESP-14	22	2 mm	1	Etched-on PCB	24.3 × 16.2

Table 1: ESP8266 module family [12].

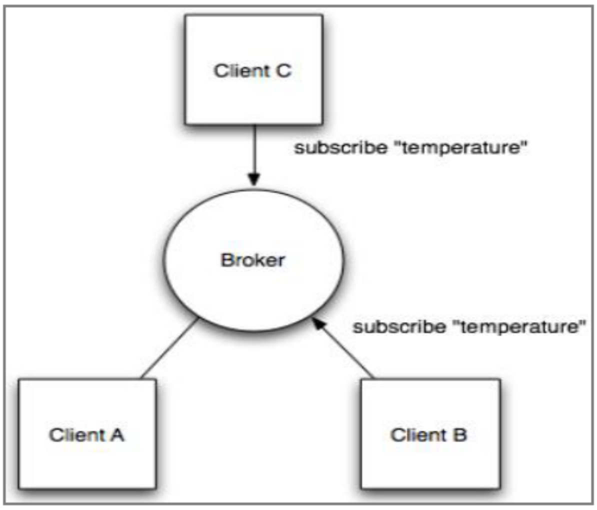

Figure 7: MQTT Architecture [13].

Server then publishes this data to the connected devices and clients [13].

It is mostly used for wireless sensor networks specifically for embedded devices [14].

There are 5 ways in which MQTT protocol works [15].

- **Connect:** Connects to the server wirelessly.
- **Disconnect:** Disconnects after finishing the task.
- **Subscribe**: Waits for subscribing the client to the broker.
- **Unsubscribe**: Client can be unsubscribed from the topics or connected things.
- **Publish**: Returns to the application thread after finishing the task and displaying value to the client.

MQTT also takes care of the security by asking username and password to connect to the broker.

CoAP protocol

CoAp stands for The Constrained Application Protocol. CoAp protocol is from "CoRE (Constrained Resource Environments) IETF group". This protocol is specifically designed for machine to machine interaction which is also a part of IoT [16]. CoAP is based on UDP. UDP enables the hardware to quickly wake-up. It transmits smaller packets of data with low overhead. This makes possible for a device to stay in a sleep mode for long time, to save the battery [17].

CoAp is a document transfer protocol similar to HTTP.

CoAP also follows client/server architecture. Client makes requests and servers send the responses back in the form of data.

But for the application of IoT, MQTT is more stable, reliable and lightweight protocol than the other protocols. MQTT is easy to get up in working network.

Case Studies

Case study on ESP8266 Version 01

Experiment: A study was done on ESP8266 01 Wi-Fi module for a small IoT application. An experiment was performed to turn on and off the LEDs connected to the module.

Connections were made as per the requirement for flashing the firmware into the chip. Firmware was custom made as per the requirement. After flashing the firmware, the module was taken out from the flash mode by changing the wiring and the code was uploaded to it.

Thingspeak.com was used as a remote server. Wi-Fi module was connected to the Thingspeak.com by using the API key and channel id provided by the server. On the other end, a smart phone with an android application was also connected to the thingspeak.com by using same API key and channel id using any internet connection.

Result: From the smart phone, the LEDs were able to turn on and off successfully. But a problem was faced regarding the response time of total operation. The response was not quick, a delay of about 8 to 20 seconds was observed between each switching and actual operation of LED. Figure 8 shows the HI / LOW status of the LEDs updated on the thingspeak.com account.

Also, this has a disadvantage of limited GPIOs. Only two LEDs were able to connect at a time. There problems are not desirable in order to use this module to implement IOT in a humanoid robot.

Case study on ESP8266 version 12

Experiment: A similar experiment was carried out with ESP8266 version 12 wi-fi module. This module is available in the form of a development board with onboard LEDs for testing, which is much easier to use than the former versions.

In this experiment, cloudmqtt.com was used as a server. Module was connected in a similar way to the server by using the key provided by the server. This was also operated by a smart phone with an android application.

Result: In this experiment the delay between the switching was reduced to almost less than a second which is desirable.

This experiment was also carried out to actuate multiple servo motors along with the LEDs as shown in Figure 9. A PWM signal was generated from the android application by using a slider interface. Values from 0 degree to 180 degrees were set to the PWM signal in the arduino microcontroller. By sending the signals from smart phone to the server then to the wi-fi module and then to the arduino microcontroller and finally to the servo motors, multiple servos were successfully actuated from multiple sliders in the android application.

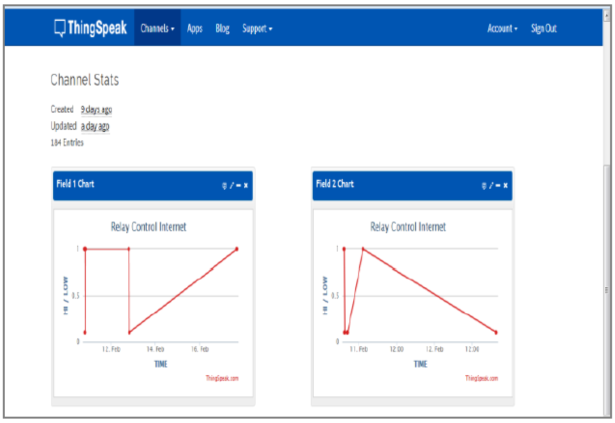

Figure 8: HI / LOW status of LED.

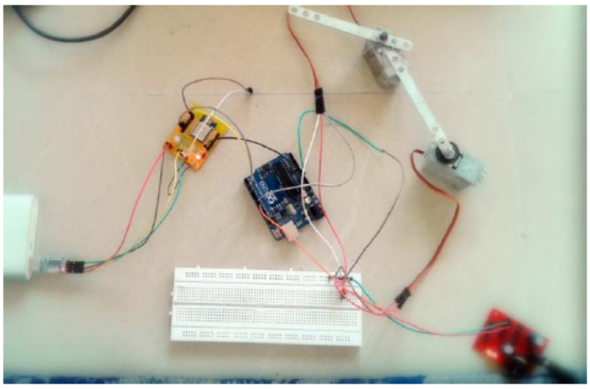

Figure 9: Actuation of servo motors and LEDs.

Conclusion

In order to implement Internet of Things in the Humanoid application, it is necessary to identify the tasks that are to be performed by the humanoid robot. Various methods can be used to implement IoT depending on the tasks. Humanoid robot mainly consists of servo motors as the main actuators.

As the number of actuators increases the need of more GPIOs also increases. Humanoid robots are very sophisticated in terms of number of sensors and actuators used. Selection of the proper IoT hardware is very important so that it should withstand the processing speeds and communication bandwidths. ESP 8266 version 12 wi-fi module is a very good hardware with many advantages as discussed earlier, for moderate level projects. Sensors attached on the humanoid robot such as temperature sensor, accelerometer sensor, PIR sensor, gyroscope, strain gauges, etc. can be successfully connected to the ESP8266 wi-fi module. Single servo at a time can also be actuated with this method. This method makes use of MQTT protocol for communication which is the most accepted and stable protocol for internet of things. Pros and cons of various methods, hardware, architectures, protocols for IoT should be considered in order to make the proper selection depending upon the need of application.

References

1. Sarma S (2016) IoT is not hype, but it's also not some magic technology. Leslie D'Monte.
2. Landau DM (2016) How 5G will Power the Future Internet of Things. Iq Intel.
3. Roy U (2016) Internet of Things: Let devices do the talking. Tech News Technology.
4. Grieco LA, Rizzo A, Colucci S, Sicari S, Piro G, et al. (2014) IoT-aided robotics applications: Technological implications, target domains and open issues. Computer Communications 54: 32-47.
5. Buschmann T, Sebastian L, Heinz U (2009) Humanoid robot lola: Design and walking control. Journal of physiology-Paris 3: 141-148.
6. Dmitri S (2016) RFID technology and internet of things. Slide share.
7. Ray Floyd (2014) RFID and the Internet of Things. Engineering.com.
8. Kadir W, Muhamad HW, Reza ES, Ibrahim BSK (2012) Internet Controlled Robotic Arm. Procedia Engineering 41: 1065-1071.
9. Doukas C (2012) Building Internet of Things with the Arduino. Create space.

10. Liguo Q, Huang Y, Tang C, Han T (2012) Node design of internet of things based on ZigBee multichannel. Procedia Engineering 29: 1516-1520.

11. https://www.arduino.cc/en/Guide/ArduinoGSMShield

12. http://www.esp8266.com/wiki/doku.php?id=esp8266-module-family

13. Toby J (2014) MQTT and CoAP, IoT Protocols. Eclipse.

14. http://mqtt.org/documentation

15. http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html

16. http://coap.technology/

17. James S (2015) MQTT and CoAP: Underlying Protocols for the IoT. Electronic design.