Low Cost Network for Internet of Things

Fernando M. Almeida, Admilson R. L. Ribeiro, Marco T. Chella
Department of Computing
Federal University of Sergipe
Avenida Marechal Rondom, S/N, Jardim Rosa Elze
CEP 49.100–000, São Cristovão, SE – Brazil
fernando.m.al.91@gmail.com, {admilson,chella}@ufs.br

ABSTRACT

The extension of Internet communication in human-human, human-machine and machine-to-machine will transform the Internet into the Internet of Things. A low-cost network is presented and implemented, allowing embedded systems to provide services over the Internet. The use case presented in this work implements the proposed architecture and the used components have low-cost and easy access allowing the reproduction of the use case. The codes used are also provided.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication

General Terms

Design, Experimentation.

Keywords

Internet of Things, Home Automation, Service Oriented Architecture, Low-cost network, REST

1. INTRODUCTION

Nowadays, the main form of communication on the Internet is human-human, however the trend is that this form of communication will extend to human-human, human-machine and machine-to-machine. These new forms of communication will transform the Internet into the Internet of Things [1]. The Internet of Things will impact many aspects of the day of the users.

An example of an application that uses this impact is the home automation [2]. The Home Automation integrates electro-mechanical systems in order to automate tasks in the home. The Internet of Things can assist home automation with intercommunication of home devices or with external communication through the Internet.The text should be in two 8.45 cm (3.33") columns with a .83 cm (.33") gutter.

One of the architectures that enable machine-to-machine communication is the service-oriented architecture, since SOA enables a machine can find out the services provided by other machines.

Service-oriented architecture is an architectural style that supports the service-orientation paradigm, and that paradigm deals with the construction and integration of software solutions composed of modular elements called services [3].

Some of the solutions include the mote TelosB[1], which costs 77,00 euros, and the mote Z1[2], which costs 95,00 euros. Their's high cost for 1 mote dificult access to students and hobbists. Their boards were not designed to make easy the assembly of new components, it is necessary to use specific cables or solder them. In addition it is only possible to buy them in their manufactures' stores.

This work presents a network of low-cost sensors and actuators based on a home automation network applying concepts of Internet of Things.

The network can be assembled with components easily accessible and programmed with Open Source software, aiming to ease of reproduction of this project. It should allow communication of the Internet by itself, providing access to the services provided in the network to other machines or humans.

The low cost and easy access to components is a requirement of the projected network, so it could be replicated allowing people to apply some concepts of Internet of Things in their projects.

Section 2 presents the proposed architecture that allows a low-cost implementation of the concepts of Internet of Things, section 3 shows how service provision was implemented on the local network. The section 4 presents the use case for the validation of the proposed architecture.

2. NETWORK ARCHITECTURE

The network architecture proposed has three types of nodes: the local node that has sensors and actuators, the master node, which acts as a bridge between local nodes and the border node, and the border node, which serves as a bridge between the Internet and the local network.

The local nodes communicate with each other and with the master node through wireless communication modules. In addition to the wireless module, local nodes have also a processor and, optionally,

SForum'14, September 1–5, 2014, Aracaju, Sergipe, Brazil.

[1] http://www.advanticsys.com/shop/mtmcm5000msp-p-14.html

[2] http://webshop.zolertia.com/product_info.php/cPath/23/products_id/32

sensors and actuators provided as services. Possibly the local node should act as a router, to make possible to the packets to arrive their destination.

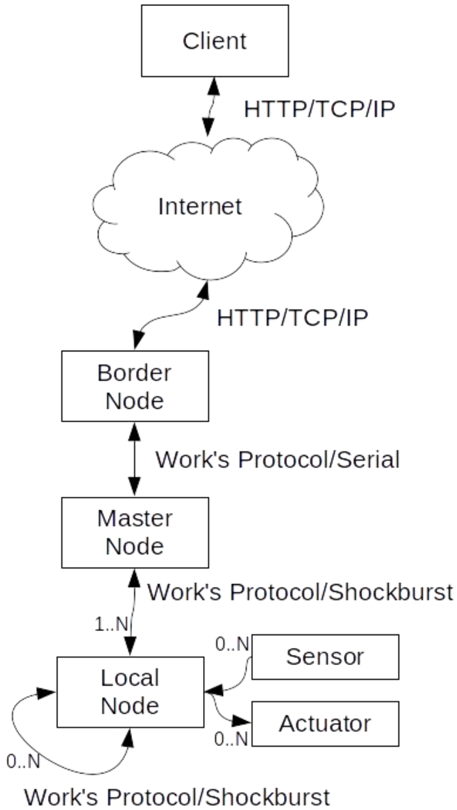


Fig. 1 Proposed architecture of Network

The border node is responsible for interfacing the local network with the Internet, your presence is important to treat the TCP[4]/IP[5] (Transmission Control Protocol/Internet Protocol) stack and other Internet standards from the local nodes. The communication between the border node and the local nodes is made through the master node.

The master node is responsible for communication between border and local nodes. Without the master node, the border node must have the wireless interface, but with the master node, the border node must have another interface, for example USB (Universal Serial Bus).

The figure 1 shows the overview of the proposed architecture of Network. A Client communicates with the border node using HTTP. The border node translates the HTTP request into Work's Protocol request and route it to the destination. The master node is connected to one or more local nodes. A local node is connectedto zero or more local nodes, zero or more sensors and zero or more actuators.

3. PROVISION OF SERVICE

The literature provides two possibilities of guidelines for a Service Oriented Architecture, one oriented to resources, the REST (Representational State Transfer) architecture, and the other oriented to activities, the SOAP (Simple Object Access Protocol) architecture.

Recent projects of the Internet of Things use mostly the REST architecture to provide services, by the reason that SOAP architecture requests more resources of the server [6].

The methods to access a resource in the REST architecture are: GET, POST, PUT and DELETE. Other methods can be provided depending on the application protocol, but a REST implementation that uses the HTTP (Hypertext Transfer Protocol)[7] protocol, use these four methods.

Despite the local network provides their services to the Internet through the HTTP protocol, the local infrastructure does not provide suitable conditions for a verbose protocol, like HTTP. In this work a simple protocol was developed in order to enable the provision of services for the local network. The information contained in a package of this protocol are: destination address, source address, packet identifier, request or response information and extra information.

The destination address stores the address of the node that will receive the package, the source address stores the address of the node that sent the package, the package identifier avoid duplicated packets in a small time interval on the network, the request of response information indicates whether the packet is a request or a response and the extra information stores the type of request or response code.

All the request/response information is compacted into a packet. The packet structure was designed to have the main information for a REST service and fit into a 32 bytes packet. The packet header can be viewed in figure 2. The size of the header is 3 bytes, the data of packet can be as larger as 29 bytes.

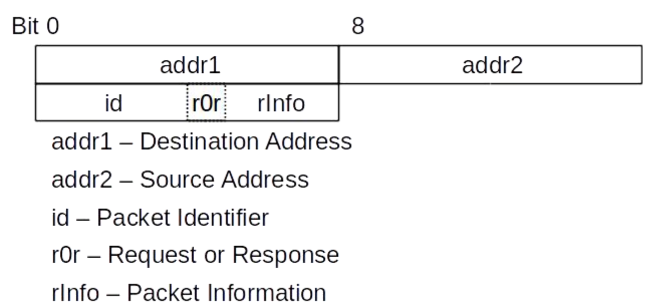


Fig. 2 Packet Header

For example, for a POST Request from node 2 to node 5 with packet id 3, the packet information will be: addr1 = 2, addr2 = 5, id = 3, rOr = 0 (Request) and rInfo = 1 (POST). The packet header in binary form will be 00000010_00000101_00110010.

4. USE CASE

The use case implemented was used to validate the proposed architecture. This case illustrated a home automation application where you can control the color and brightness of a colored LED (Light Emitting Diode). The control of this LED is supplied as a REST service and can be consumed by a browser or a local node.

The designed network has two local nodes, one of them connected to the colored LED and the other consuming the service of the LED. The local node consumer has a LDR (Light Dependent Resistor) sensor which provides the information that will be used to control the LED, this node provides the service to activate or deactivate the automatic control of the LED on the network.

The local and master nodes were implemented using the Arduino platform[3], for being a low-cost and a open source platform. The wireless module used in the network was the NRF24L01[4] by its low cost when compared to other wireless modules.

The border was implemented using the Raspberry Pi board[5] with the Raspbian OS, the HTTP server was implemented using NodeJS[6]. Communication between the master and border node was done through a serial communication emulated in a USB communication.

The block diagram of the use case can be seen in figure 3, it presents the components use and communications held between the components. The figure shows that each local node needs an microprocessor and a comunication node, but it's possible to connect sensors and actuators.

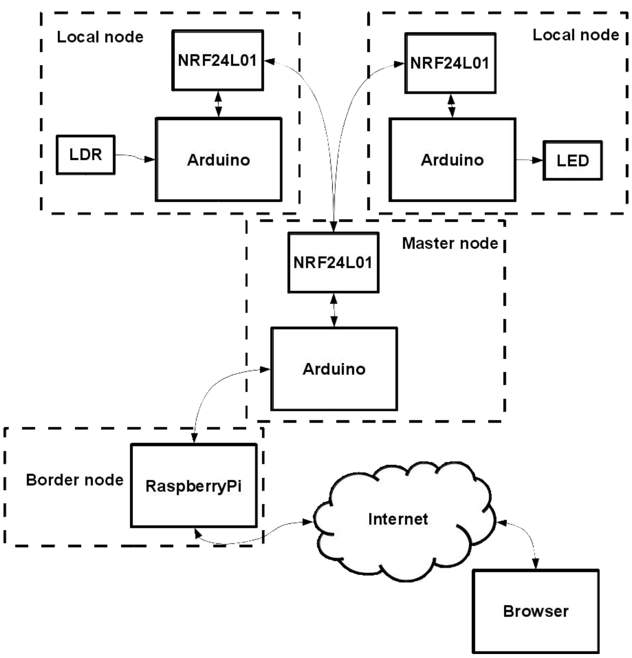


Fig. 3 Block diagram of the Use Case

[3] http://www.arduino.cc/

[4] http://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01

[5] http://www.raspberrypi.org/

[6] http://www.nodejs.org/

The client that consumes the services of the two local nodes was implemented in HTML and Javascript, the figure 4 presents its execution in a browser. The Client in the browser shows the possibility to create interfaces on any device that connects to Internet.

Fig. 4 Client implemented to consume the services of the Use Case

5. RESULTS

The proposed use case was successfully implemented, i.e. the provision of services on the local network to the Internet worked properly and the consumption of services on the local network also worked properly.

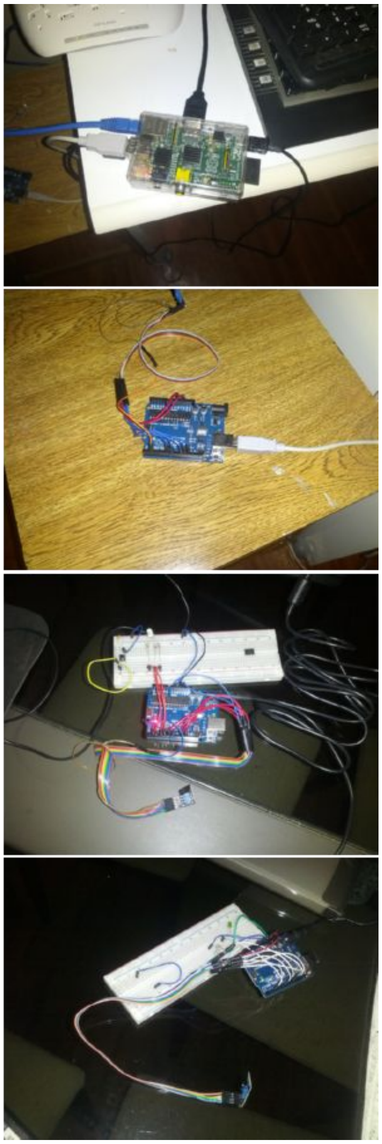
Component	Price/Unit	Qnt.	Price
RaspberryPi	U$40.000 (Newark)	1	U$40.000
Arduino Uno R3	U$29.290 (Newark)	3	U$87.87
NRF24L01	U$2.405 (DigiKey)	3	U$7.215
LED MULTICOL	U$1.430 (Newark)	1	U$1.430
LDR	U$1.900 (Newark)	1	U$1.900
1kOhm Resistor	U$0.065 (Newark)	3	U$0.195
10kOhm Resistor	U$0.065 (Newark)	1	U$0.065
		Total	**U$138.675**

Table 1 Cost of components in the Use Case (Evaluated in 05-22-2014)

The use case has a low cost, which can be observed in table 1. The components used are of easy access, allowing the experiment to be replicated with some ease.

It's possible to verify some patterns in the communication that can assist in the development of a library to help to connect the nodes in that network.

In figure 5 is possible to observe the nodes mounted, the nodes are: a) Border Node, b) Master Node, c) Local Node that is connected to the LED and d) Local Node that is connected to the LDR.



a) b) c) d)

Fig. 5 All the nodes of the network implemented in the Use Case

6. CONCLUSIONS

The concept of Internet of Things makes home automation applications more powerful, the communication between the devices in home enables various types of applications.

The network, designed and implemented, can serve as a basis for future works. Even if the network is limited, it allows communication between devices and the Internet, providing services. The use of easily accessible components and open source tools allows the reproduction of the experiment at low cost.

The codes used in this work can be found on GitHub[7].

[7] https://github.com/fernandomalmeida/tcc-domotica

7. REFERENCES

[1] Tan, L., & Wang, N. (2010, August). *Future internet: The internet of things*. In *Advanced Computer Theory and Engineering (ICACTE)*, 2010 3rd International Conference on (Vol. 5, pp. V5-376). IEEE.

[2] Atzori, L., Iera, A., & Morabito, G. (2010). *The internet of things: A survey*. Computer networks, 54(15), 2787-2805.

[3] Fugita, H. S. and Hirama, K. (2012). *SOA, modelagem, análise e design*. Rio de Janeiro: Elsevier.

[4] J. Postel, Transmission Control Protocol, RFC793, Sept 1981, http://www.ietf.org/rfc/rfc793.txt.

[5] J. Postel, Internet Protocol, RFC791, Sept 1981, http://www.ietf.org/rfc/rfc791.txt.

[6] Guinard, D., Trifa, V., Mattern, F., & Wilde, E. (2011). *From the internet of things to the web of things: Resource-oriented architecture and best practices*. In *Architecting the Internet of Things* (pp. 97-129). Springer Berlin Heidelberg.

[7] R. Fielding, J. Gettys, J. C. Mogul, H. Frystyks, L. Masinter, P. Leach, and T. Berners-Lee, Hypertext Transfer Protocol – HTTP/1.1, RFC2616, Jun 1999, http://www.ietf.org/rfc/rfc2616.txt.