

BACnet Protocol Implementation on Lab View and Device Development using Arduino

Suyash Barodiya¹, Nital Patel² and Kapil Purabiya³

^{1,2,3}Control & Automation, Institute of Technology, Nirma University, Ahmedabad
 E-mail: ¹suyash.barodiya@gmail.com, ²nital.patel@nirmauni.ac.in, ³purabiyakapil@gmail.com

Abstract—BACnet (Building automation and control networks) protocol implementation on LabVIEW (Laboratory Virtual Instrument Engineering Workbench) platform is presented in this paper. Project has been started to eliminate the use of BACnet gateway, which translate the information for non-BACnet devices. LabVIEW platform has been chosen for implementing the BACnet protocol because of its popularity in field of instrumentations and ease of use. Direct communication between any BACnet device and system like scada can provide better performance in terms of lower propagation delay and additional cost reduction. For the testing of protocol, one temperature sensing BACnet device has also been created and the validation of device is also analyzed in this paper. The paper comprise of implementation work and results.

Index terms : BACnet; LabVIEW; Communication; Protocol implementation.

1. INTRODUCTION

Building automation became more popular in building industries, which consist the automatic centralize control of HVAC (Heating ventilation and air conditioning) system [1]. For establishing the communication between HVAC devices and controlling system, a specific protocol has been developed by the people from Cornell University, which is BACnet, due to the advantage of interoperability and interchangeability this protocol is being widely used in building automation industries for data exchange and Communication over the network. LabVIEW platform is used for protocol implementation because of its versatility. LabVIEW is a product of National Instruments which uses dataflow programming, where the contents are represented in graphical form, it consist two windows front panel and block diagram, combination of both is called VI (virtual instruments), also it contains predefined functional blocks which is called pallets, which makes it more easy and interesting to use[2].

2. BACNET PROTOCOL

BACnet an open source protocol but it is not free, it is not a hardware or electronic chip, it is a just a chargeable book of standard which is approved by ASHRAE (American Society of Heating, Refrigerating, and Air Conditioning Engineers).

This book is comprise of objects, properties, protocol architecture and set of rules, which tells how the protocol supposed to work on embedded systems [3].

BACnet is a four layer architecture shown in fig. 1, in which physical, data link, network and application layers of the OSI model are present.

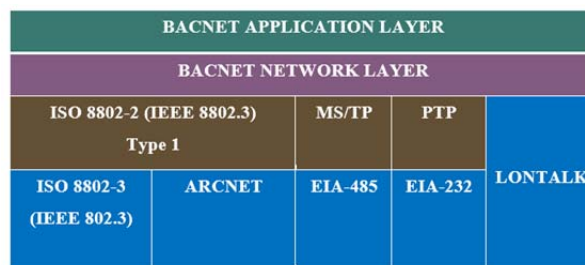


Fig. 1: BACnet Architecture

Information inside the BACnet device organized in form of, set of objects with their properties. This Object oriented approach makes this protocol, user friendly compare to the other protocol which consist very complex sequences structure.

There are total 54 objects are present today and they are still growing, as developers are evolving the new objects and ideas. Some objects are analog input, analog output, analog value, digital input, digital output, and digital value and so on and properties are read and write properties etc. [3]. These objects and properties can be used by various application according to their needs.

3. BACNET DEVICE USING ARDUINO

Among the six lane media types, BACnet I/P has been chosen to develop as a temperature sensing device, using arduino and ethernet shield. Arduino is an open source platform which is used for research and project work, which consist microcontroller and IDE (Integrated development environment) for creating the program [4], and ethernet shield

is an add-on, which allows arduino to communicate over the I/P (Internet protocol).

For creating the application layer in arduino, BACnet open source stacks has been used, and for the temperature monitoring, one wire library is used. Project is compiled with eclipse IDE and avr plugin [8].

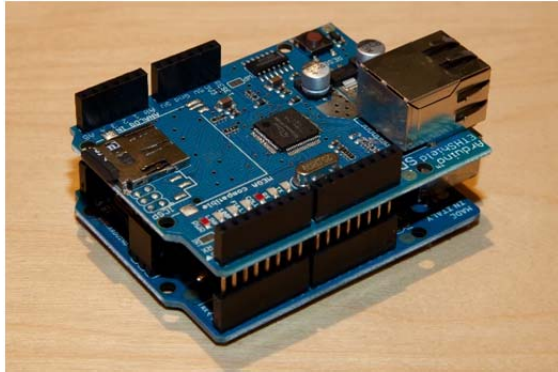


Fig. 2: BACnet I/P Device using Arduino

Fig. 2 shows arduino Uno board at bottom with ethernet shield on top. In this device the temperature is measured by one wire sensor and its o/p is received by arduino on digital pin no. 2, and then it sends the data over the BACnet I/P network by ethernet shield.

For the validation of this device two open source software YABE (Yet another BACnet explorer) and BDT (BACnet discovery tool) has been used and tasted well.

4. IMPLEMENTATION OF PROTOCOL ON LABVIEW

BACnet protocol is very easy to use for end user but it is quite complex to implement. It was recommended by the experts, that one should start with BACnet I/P as it is very easier to write, compare to others. It's a push type communication where the data is originated from sender and then responded to by receiver [5]. In order to establish the communication between BACnet device (Client) and LabVIEW (Server), a correct frame should be receive by client, consequently it will generate some response accordingly, which should be captured by server. Device identification on network can be done by Who-is and I-am services. Server uses the Who-is service to determine the object identifier and network address of the device and I-am service is used by client to response to Who-is service request [6].

Configuration of I/P port is the first step for this communication. BACnet I/P uses the user datagram protocol (UDP), which is connection less protocol and allows one-to-many messages [7]. UDP pallet is already available in LabVIEW, so the only thing that need to be define is port which is 47808 (BAC0H), its default UDP port for broadcasting the messages [6]. Sending of Who-is command is

next step of this communication, which is an unconfirmed request and then after I-am frame will receive, which will have to decode.

Table 1: BACnet unconfirmed request format

Bit	7	6	5	4	3	2	1	0
PDU Type	x	x	x	x	0	0	0	0
Service	x	x	x	x	x	x	x	x

Table 1 shows the unconfirmed request format, it's an 8 bit frame. First frame define the PDU Type in which last 4 bits are important and remaining 4 bits must be zero, those bits are reserved by ASHRAE [6]. In our case this frame is 00010000, which defines that request is unconfirmed. Second frame define the service choice. There are total 10 unconfirmed service request but in our case we have used only 2 of them, which is Who-is (00001000) and I-am (00000000) [6].

After completing above steps one can go for other different commands for different services, like in our case we have applied some confirmed request, that is read property service, which is, for object list, object names and for analog present value, which is written inside the device. All the commands, we have applied follows the flow given in fig. 3.

4.1 Communication Flow

Fig. 3 shows that how the commands in BACnet communication spouse to work. Configuration of UDP port can be consider as first step of this flow, which is followed by network setting in personal computer which includes the

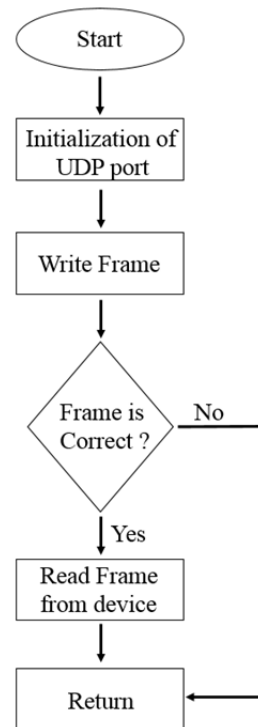


Fig. 3: BACnet communication flow

setting of IP address and subnet mask setting in server as well as in client. In our case the IP address of BACnet device (Client) is 192.168.0.185, so the PC (Server) must have the IP address within the range. After the successful initialization, if a valid command is sent on network, then device will respond accordingly. For example if server generates the correct Who-is command then client will respond with I-am command, for which server should be ready to capture the frame and if something goes wrong then whole process should be start from the beginning.

4.2 LabVIEW VI

Fig. 4 shows the LabVIEW block diagram in which UDP port is set to 47808 and different commands like Who-is, I-am, Object Identifier and read analog present value has been implemented as LabVIEW Sub-VI. UDP write pallet is used in this VI for writing the data on the network, which is 81H (BVLC type BACnet I/P), 0BH (BVLC function), 000CH (Length), 01H (Version), 20H (Control), ffffH (Destination address), 00H (MAC length), ffH (HOP count) followed by PDU type and service choice (10H & 08H) [6]. Same way UDP read pallet is used for receiving the data. In this way different commands has been implemented.

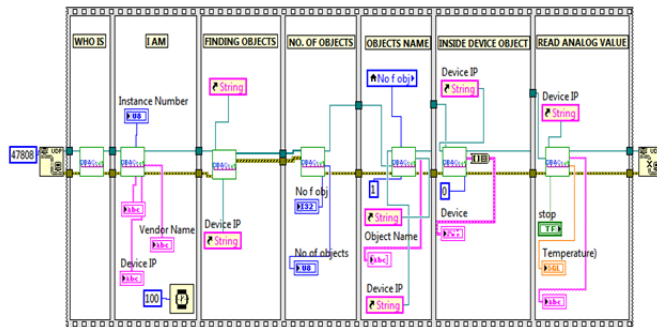


Fig. 4: LabVIEW block diagram

5. RESULT

Commands sent from the LabVIEW are correct and in result proper response has been achieved as it should be. After sending the Who-is command, the BACnet device responding with I-am, in which object identifier has segregated, and got the device object name, its vendor name and other parameters.

Fig. 5 shows the result that is achieved on LabVIEW front panel. Instance number, vendor name and device IP, representing the I-am response and different objects including device objects is achieved by confirmed request read property for object list and object names. For reading the temperature value, read analog present value command has been used, in which the present value is in form floating point which has been handled by type cast function which is already available in LabVIEW.

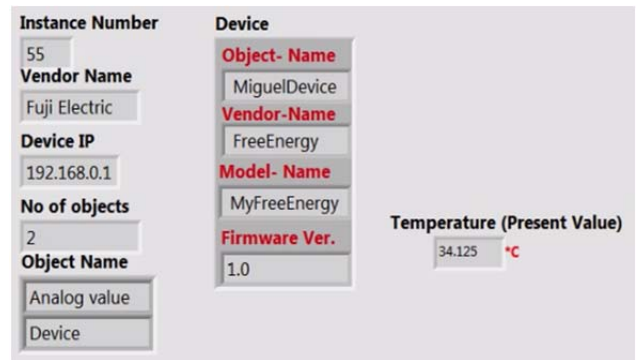


Fig. 5: LabVIEW front panel

6. CONCLUSION

Some basic commands like Who-is, I-am and read analog value, of BACnet protocol has been demonstrated on the LabVIEW platform, which has tested and verified by the temperature sensing BACnet device, without using any gateway or converter and successfully established the communication between BACnet device and LabVIEW.

The addition device has been eliminated successfully, which was the main goal of this project and by which overall system cost has reduced. It is also observed that, Transmission rate got improved by removing the gateway because the additional circuitry were talking the additional conversion time, just like the translator between sender and receiver, who know different languages. This project can be further extended for other lane-media type like MSTP (Master Slave Token Passing). Due to the versatility of LabVIEW, this VI can be used by different applications like scada systems, which are based on LabVIEW.

7. ACKNOWLEDGEMENT

Our special thanks to Mr. Steve Karg, who has written the open source BACnet stacks. We would like to thank him for their guidance and support, without him it was not possible. We would also like to thank Mr. Miguel fernandes for helping us to resolving the issues related to the BACnet device implementation using arduino.

REFERENCES

- [1] En.wikipedia.org, BACnet, 2016[Online]. Available:http://www.wikipedia.org/Buildingautomation [Accessed 9 April 2016].
- [2] En.wikipedia.org, LabVIEW, 2016 [Online].available: http://en.wikipedia.org/LabVIEW [Accessed 8 Mar 2016].
- [3] BACnet.org, BACnet, 2015. [Online]. Available: http://www.BACnet.org [Ac-cessed 10 Sep 2015].
- [4] Sparkfun.com, Arduino, 2016[Online]. Available: http://www.learn.sparkfun.com/tutorials/what-is-an-arduino [Accessed 10 April 2016].

-
- [5] En.wikipedia.org, Pull technology, 2016 [Online]. Available: http://www.wikipedia.org/pull_technology [Accessed 9 April 2016].
- [6] ASHRAE standard, "A Data Communication Protocol for Building Automation and Control Networks", 2004.
- [7] BACnet.org, BACnet/IP, 2015. [Online]. Available: <http://www.bacnet.org/Tutorial/BACnetIP> [Accessed 10 Sep 2015].
- [8] Github.com, Energy monitoring device, 2015 [Online]. Available <http://github.com/mig-l-f/bacnet-energy-monitor> [Accessed 2 Nov 2015].