Pak. J. Biotechnol. Vol. 13 (special issue on Innovations in information Embedded and Communication Systems) Pp. 199- 202 (2016)

MODBUS TO M-BUS PROTOCOL CONVERSION GATEWAY WITH LOW CONVERSION LATENCY FOR BUILDING MANAGEMENT SYSTEMS

Vasanth Dhinakaran [1] *and* GugaPriya G[2]

SENSE Department, VIT Chennai
vasanth.dhinakarans2014@vit.ac.in; gugapriya.G@vit.ac.in

ABSTRACT

Modbus is a common industrial field control bus protocol which is not only used in a wide variety of industrial applications like SCADA systems but also commonly used in building management systems(BMS) across the globe.Buildings generally use BTU,Water,Energy meters that operate on M-bus standard.M-bus devices cannot communicate directly with a Modbus based Building Management System(BMS).The network level incompatibility of a M-bus device to communicate with a Modbus master can be overcome by using a protocol conversion device having low conversion delay time and suitable voltage adjustment circuits.My project will focus on building a protocol conversion gateway for the M-bus slaves to communicate with a Modbus based master with minimum time delay using RS232 based serial medium for communication.

Index Terms: M-bus protocol, Modbus protocol, protocol nversion,BMS,BAS,Building Automation System,Building Management System.

I. INTRODUCTION

Nowadays we might come across Building Automation Systems(BAS) /Building Management System(BMS) using common industrial SCADA systems that use modbus and profibus protocols for their operation. An interesting observation to be made in this context is usage of m-bus meters as slave devices on the building. Meter bus(m-bus) is a popular European protocol used in buildings and industry as a high speed protocol for standalone readout of consumption meters like electricity, water and gas..A set of different protocols like modbus and m-bus requires hardware and software level changes to communicate wz accomplish the task. My project focuses on the software level details involved in changing the modbus to mbus protocol .This paper is divided into four sections ,the first section will focus on the technique involved in obtaining the incoming frames from the mbus slave section and identifying the data sender's id and classifying the kind of data obtained after scrutinising the mbus frame within the protocol conversion board. The second section of this paper is the protocol conversion part ,wherein the data from the m-bus slave is extracted and enclosed within a modbus response frame. The third section of this paper will focus on the latency involved in the entire transaction between the Building Automation System (BAS)/ Building Management System (BMS) and m-bus slave devices connected in a daisy chain configuration. The final section of this paper will focus on the hardware implementation used and the scope of improvements in the protocol conversion library such as securing the modbus frame being generated by the protocol conversion board using Tiny Encryption Algorithm.The scope of this project is vast with many befitting stakeholders,for example : The need to add a m-bus based European meter in a modbus based SCADA network.Other advantages of this modbus to m-bus protocol conversion library is that it can be easily implement encryption modules within the converted frames and also verify the incoming encrypted frames after decrypting it. This project uses an Atmega 2560 low cost development board for implementing the Modbus to m-bus protocol converter a modbus master simulator software modscan32 for simulating the BMS's master section. Two m-bus slaves have been used as a part of the slave section. Each of the m-bus slaves operate on predefined m-bus long, short and acknowledgement frames and are connected using the common daisy chain configuration with line termination resistances at the transmit and receive sections.

II.M-BUS FRAMES & DATA INTERPRETATION

M-bus frames are divided into short frame, long frame, control frame and acknowledgement frames.The protocol converter board has to understand the structure and significance of these frames to perform the protocol conversion in a successful manner. A brief overview of the four different frames:-

A. Acknowledgement frame/Single character frame:-

The acknowledgement frame or single character frame is used to acknowledge the incoming frames from the master. A master device that operates on modbus protocol does not need separate frames for the purpose of acknowledgement. Hence the E5 single character frame from the m-bus slave device is generally discarded during continuous operation and a write response without any data or null is sent to the modbus master simulator from the protocol conversion board.

E5

B. Short frame:-

The short frame uses only two fields the control and acknowledgement fields. This frame could be used to identify the m-bus slave meter that is sending sending the responses to the modbus master. Hence this frame's converted value will translate into a simple write response frame with the meter id as the write data in predefined modbus mapped locations

Modbus master request: For reading acknowledgement

S.ID	0x03	*S.ADDRESS*	*LENGTH*	*CRC*

M-bus response:Short Frame

10h	C Field	A-Field	Checksum (CS)	16h

Equivalent modbus response:Single write

M.ID	0x06	*S.ADDRESS*	*LENGTH*	*DATA=1*	*CRC*

M.ID Refers to the id of the master which is usually set as one. The next frame 0x06 is the modbus code for single write which is followed by the starting address for setting data as 1 in predefined locations for 255 m-bus slaves. The start address location is set depending on the slave which transmits the short frame.

C. Control frame:-

A control frame is used to manipulate the slave device's internal settings using the data from the CI field or the control information field. The control information field is used to differentiate between the frames and their types besides providing instructions to perform actions in master and slave.The proposed protocol conversion library will hold logical data about the control frame and will convert the m-bus frame into an equivalent modbus write or read frame.

Modbus master request: For changing setting meter settings:

S.ID	0x10	*S.ADDRESS*	*LENGTH*	*CRC*

M-bus response:Control frame

68h	L	L	68h	C	A	CI	CS	16h

D. Long frame:-

This is the most important frame of all, it contains the data logged in by the m-bus meter on the BMS/BAS system.It also contains the type of data information , for example: energy readings, water consumption readings.The converter board uses libraries that are able to distinguish between the type of the data and map the respective data into pre-defined modbus locations. The value is written on the location only if it satisfies the TES(tiny encryption standard) criteria and meets stipulated frame size and CRC values.

Modbus master request: For reading acknowledgement

S.ID	0x03	*S.ADDRESS*	*LENGTH*	*CRC*

M-bus response:Long Frame

68h	L	L	68h	C	A	CI	UD	CS	16h

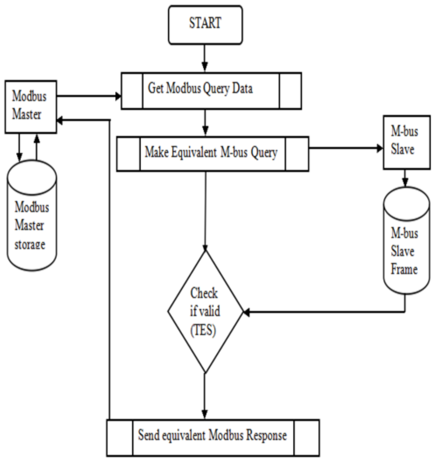
Equivalent modbus response:Multi write

M.ID	0x10	*S.ADDRESS*	*LENGTH*	*DATA=1*	*CRC*

III.PROTOCOL CONVERSION

A protocol converter device is usually used to make conversions between two different protocols communicate with each other and enable transactions between the two different systems. A protocol conversion device should contain information about the key parameters used within the protocols and mapping information for the converted frames. The library size is compact and can easily fit inside the atmega 2560 flash memory without any complicated paging logic for accessing the protocol information database. The internal settings about the m-bus devices is stored in the inbuilt EPROM of the atmega 2560.The settings stored are ID information and type of meter code information. Rest of the confidential details like billing information and consumption details are stored in the modbus master section of the system. The m-bus meter does not hold any information apart from the Manufacturing ID number and type of meter details.Rest of the details are transmitted directly to the protocol conversion device after a brief interval of time set by the modbus master device.

Figure 1:



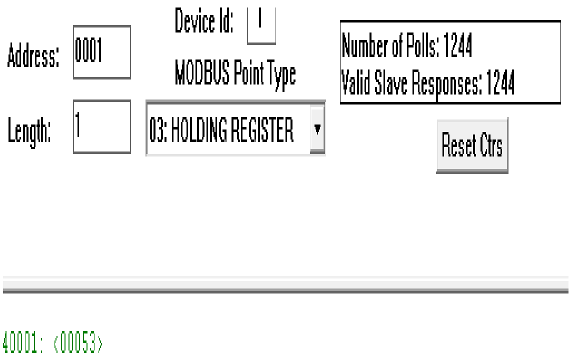
The proposed modbus to m-bus library uses a simple mechanism as shown in Figure 1. First a Modbus Query is taken into the protocol conversion board from the modbus master, and an equivalent m-bus query is sent to the m-bus slave device. The response generated by the slave is checked for TES validity,ID validity, check sum and frame format and then a equivalent modbus response is generated and transmitted to the modbus master. The atmega 2560 can handle 4 full duplex transaction channels at a time but we require only 1 full duplex channel for connecting the slave devices in a daisy chain method.The m-bus query generated by the protocol converter board is polled around the daisy chain of m-bus slaves. for showing the demo, we have used modscan 32 s the master system.Modbus master generally sends multi read(0x03) and single write(0x06) requests.

IV. RESPONSE ACCURACY AND LATENCY ISSUES

A.*Modbus master and protocol converter section:-*

One of the key issues affecting a protocol conversion device is the timing and latency involved in conducting transactions between two different networks.At a polling interval of 100ms the protocol converter device exhibits cent percent accuracy as shown in the figure 2, wherein the number of polls is exactly equal to the number of valid slave responses. When the polling interval is reduced to 50ms the protocol converter device begins to distort and disrupt the frames.The test case system uses 2 m-bus slave devices in a daisy chain configuration,as we are using a daisy chain configuration there is no significant delay being produced even if we add of more m-bus slaves to the protocol during the polling operation, as shown in Figure 4.The polling interval is modified from the modscan 32 software as shown in Figure 3.The data information obtained from the m-bus slave is shown Figure 2.The type of query read(0x03) ,single write(0x06),multi write(0x10) can be determined from the drop down menu given in the point type under the display definition options of modscan32 master software

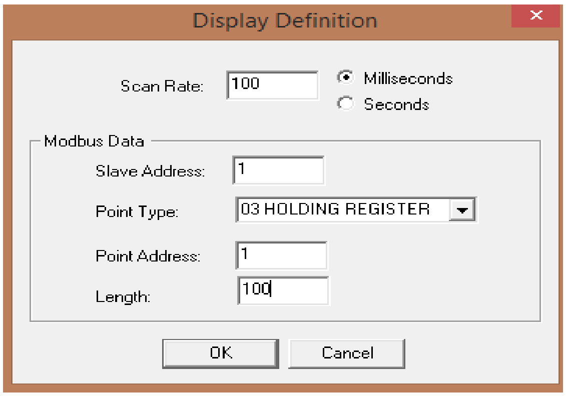
Figure 2:



Device Id: 1
Address: 0001
MODBUS Point Type
Length: 1
03: HOLDING REGISTER
Number of Polls: 1244
Valid Slave Responses: 1244
Reset Ctrs

40001: <00053>

Figure 3:

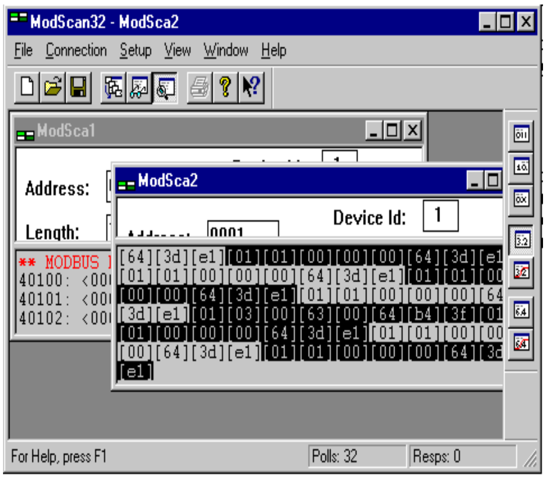


Display Definition

Scan Rate: 100 ● Milliseconds ○ Seconds

Modbus Data
Slave Address: 1
Point Type: 03 HOLDING REGISTER
Point Address: 1
Length: 100

Figure 4:



Address:
Length:
Device Id: 1
Address: 0001

```
** MODBUS I
40100: <00(
40101: <00(
40102: <00(
```

```
[64][3d][e1][01][01][00][00][00][64][3d][e1
[01][01][00][00][00][64][3d][e1][01][01][00
[00][00][64][3d][e1][01][01][00][00][00][64
[3d][e1][01][03][00][63][00][64][b4][3f][01
[01][00][00][00][64][3d][e1][01][01][00][00
[00][64][3d][e1][01][01][00][00][00][64][3d
[e1]
```

For Help, press F1 Polls: 32 Resps: 0

The white section of frames from the master simulator shown in figure 3 represent the modbus query and the dark section of frames represent the modbus response received from the protocol converter board based on the m-bus response it receives.

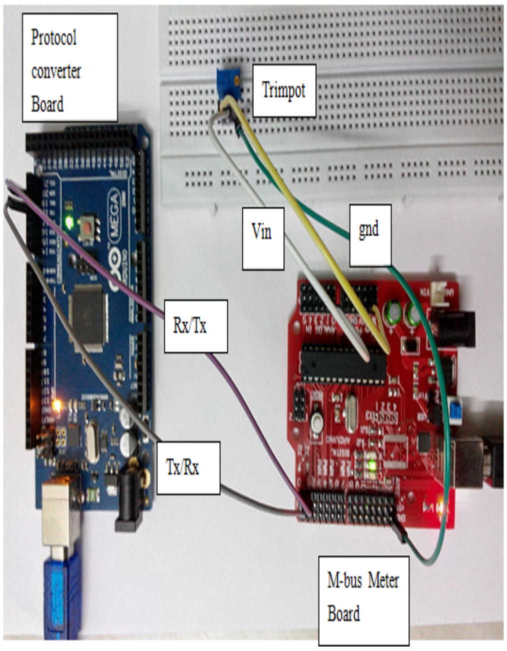
B.M-bus Slave section:-

As the proposed protocol converter device used two m-bus slave devices.Each slave device was tested to be responding within within 10 ms of delay using a separate m-bus master application by micheal rac.The TEA/TES(tiny encrption algorithm)[6]encryption was implemented using the c function modules available freely on the internet and only used on the data section. The latency involved inn calculating a TES/TEA encrypted data frame is minimal < 5 ms.Since the encrypted data frame is only formed during the long frame query by the protocol converter board latency is further reduced. Although the TEA has its own weakness and is said to be broken within a finite time period. It is practically impossible to break the encryption as each transaction is over within 100 ms and no records are kept in the slave device and the key used is a random key function. The slave section is meant to respond only during the appropriate polling calls for the appropriate m-bus slave device.

V.HARDWARE IMPLEMENTATION:

Since the proposed project is more about the protocol conversion involved, many things involved in an actual modbus master and slave have been implemented in the hardware as shown in Figure 5.And the devices are not provided with line-termination resistance as only two m-bus slaves have been used.Line termination resistances are generally used as a means to filter the noises in a daisy chain based configuration.The entire cost of a protocol converter mostly lies on its software.The software libraries used have 6 different functions, each having its own part in creating the frames for the m-bus slave devices.

Figure 5:



The hardware used to implement this concept is a low cost atmega 2560 development board by arduino and the the slave section is demonstrated by a atmega 328 module.The slave section's code is a pre-set frame response using a trimpot as shown in Figure 5.

VI.CHALLENGES FACED

Testing the protocol converter's efficiency in terms of latency is a big challenge as it involves measurement of time to complete transactions in milliseconds.Finding a fairly accurate representation of an original modbus based BMS/BAS system was challenging. But modscan32 software satisfied all the requirements of a modbus master without any difficulty.Challenges faced in the m-bus section was the testing of outgoing m-bus protocol frames, which was being fed to the protocol converter board via one of the four UART channels supported by atmega 2560,the verfication of m-bus frames was done by using m-bus simulator software by micheal rac.

VII. CONCLUSION

Hence we have effectively implemented a protocol converter device for communication between modbus master to m-bus slave device using modscan32 software from wintech as a modbus master as a substitute for the actual Building Automation System(BAS) and verified its timing latency as well as its ability to support slave devices connected in a daisy chain configuration.The protocol conversion makes use of TES algorithm on the basis of the importance of critical data and infrastructure protection mentioned by references[3][2].The concept of the protocol conversion gateway for modbus to m-bus is inspired by the detailed account provided on protocol conversion gateway basics provided by reference[1].The permissibility of applying this same concept on other protocols makes this simple protocol conversion methodology more interesting and useful for industrial applications that involve parsing of data across different systems.

ACKNOWLEDGEMENT

I would like to acknowledge the support and encouragement received from my guide Prof.GugaPriya G and my peers, lecturers, programme chair and dean at SENSE department of VIT Chennai.

REFERENCES

[1] Hao Zhang ,Yannan Li ,Huiling Zhu, Development for Protocol Conversion Gateway of Profibus and Modbus Procedia Engineering **15**: 767–771 (2011)

[2] Phan, R.C.W., Authenticated Modbus Protocol for Critical Infrastructure Protection. IEEE Transactions on Power Delivery **27**(3): 1687 - 1689 (2012)

[3] Gen-Yih Liao, Yu-Jen Chen, Wen-Chung Lu, and Tsung-Chieh Cheng, Toward Authenticating the Master in the Modbus Protocol. IEEE Transactions on Power Delivery **23**(4): 2628-2629 (2008)

[4] M-bus Protocol Specification document

[5] Modbus Protocol Specification document V1.1b

[6] Mozaffari-Kermani, M., Kai Tian, R. Azarderakhsh, S. Bayat-Sarmadi, Fault-Resilient Lightweight Cryptographic Block Ciphers for Secure Embedded Systems. Embedded Systems Letters **6**(4): 89-92 (2014)

[7] Hunn, S.A.Y., binti Md Naziri, S.Z., Binti Idris, N., The development of tiny-encrption algorithm(TEA) crpto-core for mobile systems. Electronics Design, Systems and Applications (ICEDSA). IEEE Inter- national Conference Pp. 45–49 (2012).

[8] Clarke, G., and R. Deon, Practical Modern Scada Protocols: Dnp3, 60870.5 and Related Systems. Newnes. Pp. 47–51 (2004).

[9] Palmer; Shenoi, Sujeet, Eds. Critical Infrastructure Protection III. Third IFIP WG 11. 10 International Conference. Hanover, New Hampshire: Springer Pp. 87 (2009)

[10] Felser, M. and T. Sauter, Standardization of Industrial Ethernet - the next battlefield? In: Proceedings of the 5th IEEE International Workshop on Factory Communication Systems, Pp. 413-421 (2004).

[11] Wilson. C., Common Industrial Communications Protocols. The International Journal of Thermal Technology, digital edition (2011).

[12] Zheng, Y. MZ: An Ubiquitous Communication Protocol in Industrial Environment. Int. Conf. EBISS '09, Pp, 1-4 (2009).