ISSN 2250-3153

Cross Layer Transport Layer Approach for Multihop Wireless Sensor Network

Vaishali Rajput*, Tanaji Khadtare, Atul Rohankar*****

* Information Technology Department, Sinhgad Institute of Technology & Science, Pune
** Information Technology Department, Sinhgad Institute of Technology & Science, Pune
** Information Technology Department, Sinhgad College of Engineering, Pune

Abstract- The traditional network is layered designed. Most network architectures of the radio network are also layered. In the layered design, each layer just is responsible for its own task. This can make the design more easily, and the network architecture adapted more application. But this also may make the design not efficient. Because each layer does its only job and does not know the other layer's status. For example, in many applications of wireless sensor networks, the MAC layer will shutdown the transceiver when the MAC layer thinks no data packet to send. But if when the node shutdown the transceiver, the routing layer thinks it is time to send a hello packet or something else, the node must reopen the transceiver. This does not save the energy. It is a waste of energy, the frequently open and shutdown operation will consume a lot of energy. But if the design is cross-layer, this situation could be avoided. When the MAC layer want shutdown the transceiver, it will check that whether it is time for the routing layer to send the hello message. Then the conflict is avoided. The radio network is application-dependent. So according to the actual application, the cross-layer design will make an efficient routing protocol for the wireless sensor networks.

Index Terms- Cross Layer, RTS-CTS, Autoack, Jennic

I. INTRODUCTION

The recent advances in micro-electro-mechanical system, wireless communication technology, and digital electronics have enabled the development of low-cost, low-power, small size distributed devices. Such small devices are called sensor nodes. Sensor nodes, which comprise sensing, data processing, and wireless communicating components, are capable of local processing and transmitting information wirelessly to base stations thus establishing a Radio Network. However the standardization has not been established yet. The 802.15.4 RFC is drafted for MAC/PHY layers. The Data link layer, Network layer and all rest of the layers are yet to be addressed. It is for sure that the upper layers cannot be as distinct as the OSI layer model but will have cross layer functionality due to the nature of desired small radio networks

Radio Network presents many new technical challenges for the research communities. This application addresses some important cross layer design of transport layer issues which are not handled yet.

Problem Definition

We have worked on following guidelines

- Real time study of the system overhead for bulk/file data transfer
- Devise solutions for addressing these problems.
- The protocol should have strict congestion control with little congestion control implementation overhead.
- The protocol should work with guaranteed QoS.
- The implementation code should be small (program and data memory efficient)
- Make use of the MAC/PHY layer support mechanism like AUTOACK (cross layer approach)
- The design should be generic enough to be used for other applications.

II. SYSTEM ARCHITECTURE

The system is divided into three components:

1. Master Device.
2. Network Processing Device.
3. End Node Device.

SETUP 1: End devices directly connected with Master

Figure 2.1 Mater-End device connectivity

SETUP 2: End devices connected with Master via NPDs.

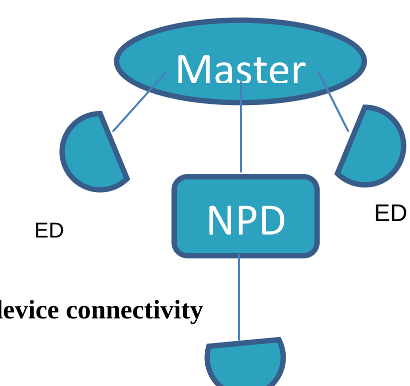


Figure 2.2 Mater-End device connectivity

III. Hardware Architecture

The Controller board is supplied with Jennic evaluation kits, which provide a complete hardware environment for designers to develop IEEE 802.15.4 and ZigBee applications, and to accelerate time-to-market. The hardware provides a stable platform which allows designers to rapidly develop and test applications based around Jennic wireless microcontroller products.

3.1 Wireless Microcontroller

Applications that transfer data wirelessly tend to be more complex than wired ones. Wireless protocols make stringent demands on frequencies, data formats, and timing of data transfers, security and other issues. Application development must consider the requirements of the wireless network in addition to the product functionality and user interfaces. To minimize this complexity, Jennic provides a series of software libraries that control the transceiver and peripherals of the JN513x. These libraries, with functions called by an Application Programming Interface (API) remove the need for the developer to understand wireless protocols and greatly simplify the programming complexities of power modes, interrupts and hardware functionality.

3.2 Wireless Transceiver

The Wireless Transceiver is highly integrated and, together with the IEEE802.15.4 MAC library requires little knowledge of RF or wireless design. The Wireless Transceiver comprises a low-IF 2.45GHz radio, an O-QPSK modem, a baseband controller and a security coprocessor. The radio has a 200Ω resistive differential antenna port that includes all the required matching components on-chip, allowing a differential antenna to be connected directly to the port, minimizing the system BOM costs. The transceiver elements (radio, modem and baseband) work together to provide 802.15.4 Medium Access Control under the control of a protocol stack supplied with the device as a software library.

4.3 RISC CPU and Memory

A 32-bit RISC CPU allows software to be run on-chip, its processing power being shared between the IEEE802.15.4 MAC protocol, other higher layer protocols and the user application. The JN513x has a unified memory architecture, code memory, data memory, peripheral devices and I/O ports are organized within the same linear address space. The device contains 192kBytes of ROM, a choice of 8k, 16k, 32k or 96kBytes of RAM and a 48-byte OTP eFuse memory.

IV. Implementation and Coding

4.1 System Implementation

To demonstrate the file transfer, transport layer protocol dynamics and our solutions on the same , serial communication is used.

4.2 Working of Uart

Two hardware flow control signals are provided: Clear-To-Send (CTS) and Request-To-Send (RTS). CTS is an indication sent by an external device to the UART that it is ready to receive data. RTS is an indication sent by the UART to the external device that it is ready to receive data. Monitoring and control of CTS and RTS is a software activity, performed as part of interrupt processing. The signals do not control parts of the UART hardware, but indicate to software the state of the UART external interface.

Characters are read one byte at a time from the Receive FIFO and are written to the Transmit FIFO using vAHI_UartWriteData (). The Transmit and Receive FIFO can be cleared and reset independently of each other using vAHI_UartReset(). The status of the transmitter can be checked using u8AHI_UartReadLineStatus (), which indicates if the transmit FIFO is empty, and if there is a character being transmitted. The status of the receiver is also checked using this call, which can indicate if conditions such as parity error, framing error or break indication have occurred. It also shows if an overrun error occurred (receive buffer full and another character arrives) and if there is data held in the receive FIFO.

4.3 File Transfer with different mechanisms

4.3.1 With RTS-CTS

Suppose Node A wants to send file to Node B.

1. Node A sends enquiry packet to Node B.

2. Node B checks for the free space it its transmit queue. If space is available it sends acknowledgement to Node A and sets itself into receive state.

3. After getting acknowledgement from Node B, Node A starts sending data packets.

4. Node B copies packet data to serial buffer for output to UART.

4. After receiving each packet, Node B sends acknowledgement to Node A.

5. Node A sends next packet.

4.3.2 Without RTS-CTS

In our application we are not using RTS-CTS instead; we are using AUTOACK mechanism at transport layer. We can achieve the quality of service by using hardware generated Autoack instead of software generated RTS-CTS.

4.3.3 AUTOACK mechanism

The JN513x baseband processor can automatically construct and send the acknowledgement packet without processor intervention and hence avoid the protocol software being involved in time-critical processing within the acknowledge sequence. The JN513x baseband processor can also request an acknowledgement for packets being transmitted and handle the reception of acknowledged packets without processor intervention.

4.4 Need For Cross Layer Design

The traditional network is layered designed. Most network architectures of the radio network are also layered. There are four layers for the radio networks, the physical layer, the link layer, the network layer and the application layer. In the layered design, each layer just is responsible for its own task. This can make the design more easily, and the network architecture adapted more application. But this also may make the design not efficient. Because each layer does its only job and does not know the other layer's status. For example, in many applications of wireless sensor networks, the MAC layer will shutdown the transceiver when the MAC layer thinks no data packet to send. But if when the node shutdown the transceiver, the routing layer thinks it is time to send a hello packet or something else, the node must reopen the transceiver. This does not save the energy. It is a waste of energy, the frequently open and shutdown operation will consume a lot of energy. But if the design is cross-layer, this situation could be avoided. When the MAC layer want shutdown the transceiver, it will check that whether it is time for the routing layer to send the hello message. Then the conflict is avoided. The radio network is application-dependent. So according to the actual application, the cross-layer design will make an efficient routing protocol for the wireless sensor networks.

4.4.1 Cross Layer Design in this Application

We have employed reliable algorithms in each layer of the communication stack. At the MAC layer, a CSMA MAC protocol with an explicit hop-by-hop Acknowledgment for loss recovery is employed. To ensure the end-to-end reliability, the maximum number of retransmissions are estimated and used at each hop. At the transport layer, an AUTOACK mechanism of MAC layer is used. By inspecting the sequence numbers on the packets, the sink can detect which packets were lost.

4.4.2 Experimental Setting

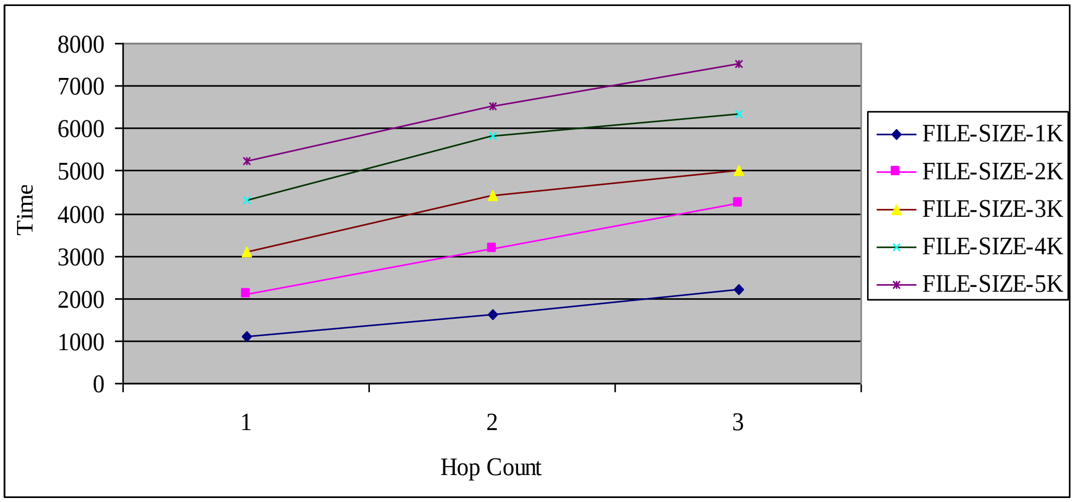
Jennic contains a controller board (featuring an LCD panel) and a number of sensor boards from the evaluation kit. The sensor boards measure temperature, humidity and light levels, and periodically send these measurements to the controller board. The controller board displays the received data on its built-in LCD panel. The interface is based on RS232. It acts as a gateway to connect the laptop and the radio wireless sensor network.

In this application, the setup was deployed within laboratory. Total 5 jennic nodes were used. One of the nodes acts as the Co-coordinator, two nodes act as Network Processing Devices and one node act as End Device. Separate code is provided for the Co-coordinator, Network Processing Devices and End Devices.

Time required for transmission of files of different sizes:

Table 4.1 Time required for transmission of files of different sizes

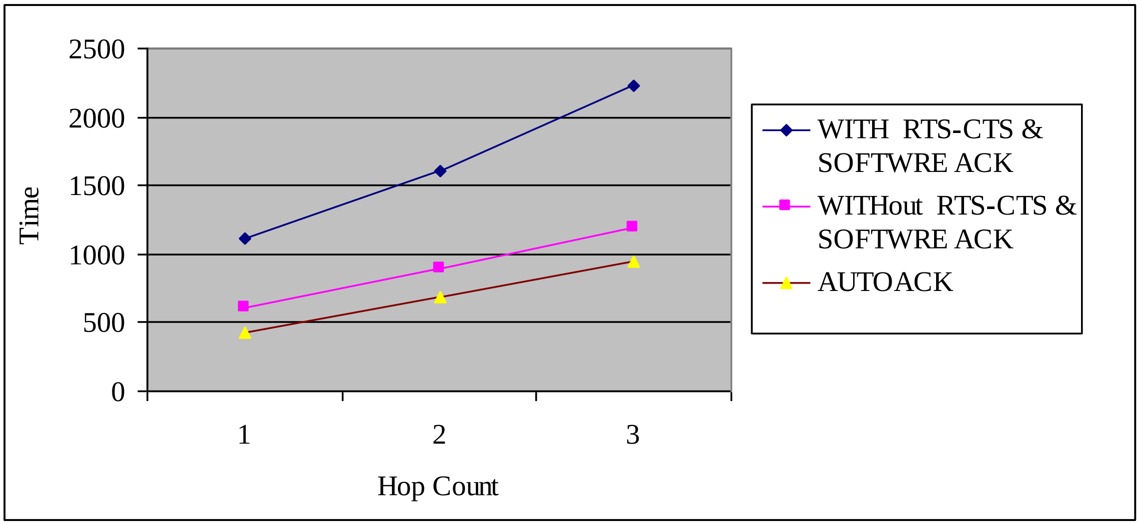
	Time required(In ms)		
File size(In Kb)	**First Hop**	**Second Hop**	**Third Hop**
1	1119	1605	2225
2	2106	3176	4240
3	3091	4421	4996
4	4302	5813	6354
5	5245	6538	7511



Time required for transmission of file with different mechanisms:

Table 4.2 Time required for transmission of file with different mechanisms

	Time required(In ms)			
File size (In Kb)	**First Hop**	**Second Hop**	**Third Hop**	
1.01	1119	1605	2225	With RTS-CTS & software ACK
	606	899	1194	Without RTS-CTS & with software ACK
	421	689	945	Only Autoack



V. CONCLUSION AND FUTURE SCOPE

System overhead increases with RTS-CTS and software generated ACK mechanism. AUTOACK mechanism reduces time required for block/file transfer. It ensures data reliability by generating hardware acknowledgement for received packet. Cross layer design approach is achieved by using MAC/PHY support AUTOACK mechanism. For transfer of audio/video files the mechanisms like data compression, data fusion can be used.

ACKNOWLEDGMENT

I express my deep sense of gratitude to my guide Prof.A.W.Rohankar for his invaluable guidance and unflinching help during the research work. His esteemed suggestion and encouragement during the entire work have been inspiring source of motivation for me. I also wish to express my gratitude to Prof.G.R.Pathak for his invaluable moral support.

REFERENCES

[1] A. Woo, T. Tong, and D. Culler, "Taming the underlying challenges of reliable multihop routing in sensor networks," in SenSys '03: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, (Los Angeles, CA, USA), pp. 14–27, ACM Press, 2003.

[2] R. Stann and J. Heidemann, "Rmst: Reliable data transport in sensor networks," in SNPA '03: Proceedings of the first IEEE International Workshop on Sensor Network Protocols and Applications, (Anchorage, Alaska, USA), pp. 102–112, 2003.

[3] C.-Y. Wan, A. Campbell, and L. Krishnamerthy, "Psfq: A reliable transport protocol for wireless sensor networks," in WSNA '02: Proceedings of the first ACM International Workshop on Wireless Sensor Networks and Applications, (Atlanta, Georgia, USA), pp. 862–872, 2002.

[4] S. Kim, R. Fonseca, P. Dutta, A. Tavakoli, D. E. Culler, P. Levis, S. Shenker, and I. Stoica, "Flush: A reliable bulk transport protocol for multihop wireless network," in SenSys '07: Proceedings of the 5th ACM Conference on Embedded Networked Sensor Systems, (Sydney, Australia), pp. 351–365, 2007.

[5] ¨Ozg¨ur B. Akan and I. F. Akyildiz, "Esrt: Event-to-sink reliable transport in wireless sensor networks," IEEE/ACM Transactions on Networking, vol. 13, no. 5, pp. 1003–1016, 2005.

[6] Y. G. Iyer, S. Gandham, and S. Venkatesan, "Stcp: A generic transport layer protocol for wireless sensor networks," in ICCCN '05: Proceedings of the 14th IEEE International Conference on Computer Communications and Networks, (San Diego, CA, USA), 2005.

[7] Y. Zhou and M. R. Lyu, "Port: A price-oriented reliable transport protocol for wireless sensor networks," in ISSRE '05: Proceedings of the 16th IEEE International Symposium on Software Reliability Engineering, (Washington, DC, USA), pp. 117–126, 2005.)

AUTHORS

First Author – Vaishali Rajput,ME(IT), Sinhgad Institute of Technology & Science, Pune, Kvaishu19@gmail.com

Second Author – Tanaji Khadtare,ME(IT), Sinhgad Institute of Technology & Science, Pune, tanajikhadtare@gmail.com

Third Author – Atul Rohankar, PHd (IT), Sinhgad College of Engineering, Pune, rohankar@gmail.com