AC 2008-1311: WIRELESS SENSOR NETWORKS UTILIZING THE IEEE 802.15.4 STANDARD IN AN ECET CURRICULUM

David Loker, Pennsylvania State University-Erie
David R. Loker received the M.S.E.E. degree from Syracuse University in 1986. In 1984, he joined General Electric (GE) Company, AESD, as a design engineer. In 1988, he joined the faculty at Penn State Erie, The Behrend College. His research interests include wireless sensor networks, data acquisition systems, and communications systems.

Wireless Sensor Networks Utilizing the IEEE 802.15.4 Standard in an ECET Curriculum

Abstract

Recently our Electrical Engineering Technology Baccalaureate Program at Penn State Erie, The Behrend College, was expanded to the Electrical and Computer Engineering Technology (ECET) Baccalaureate Program with options in both Electrical Engineering Technology (EET) and Computer Engineering Technology (CMPET). Based upon the TAC of ABET criteria for accrediting engineering technology programs, the ECET program must satisfy the criteria for both EET and CMPET programs. Thus networking concepts need to be included in both program options.

In this paper, several laboratory applications utilizing low-rate wireless personal area network (LR-PAN) technology are presented. This material can be included within one of several typical courses in EET curriculums (such as a communications or microcontroller course) as well as a networking course within CMPET curriculums.

The device used for the laboratory applications is the MaxStream XBee wireless module, since it is compatible with the IEEE 802.15.4 standard. It uses a radio transceiver operating in the Industrial, Scientific, and Medical (ISM) RF band at frequencies from 2.400GHz to 2.484GHz. The standard device has a maximum outdoor line-of-sight range of 300 feet and a maximum data throughput of 250kbps with a UART interface. The device is intended to be used in conjunction with the internal RS-232 port found in most microcontrollers and computers for applications in wireless sensor networks and remote data collection.

Several laboratory projects utilizing the MaxStream XBee wireless module are presented. The first lab is designed to introduce this module to the students. The objectives include interfacing the module to the computer serial port, using HyperTerminal to communicate with the module, configuring the module through AT commands, and verifying functionality of the module through file transfer. The second lab is designed to introduce remote data acquisition. Students will design a terminal program utilizing LabVIEW, integrate a temperature sensor, and perform remote data collection. The third lab will introduce students to personal area networking (PAN). The objective is the establishment of a wireless sensor network. It will utilize a star network configuration with temperature sensors and the development of LabVIEW software for remotely collecting temperature data and performing statistical analysis on the data.

There are several primary objectives for the presentation of these applications within this paper. First, it presents suitable networking material to be included in one or more courses within the EET option of an ECET program as required by ABET. It can also be utilized in a typical networking course within CMPET curriculums. Second, it provides a resource to aid instructors interested in introducing wireless RF technology within their courses.

Wireless Personal Area Networks

The IEEE 802.11 Working Group[1] defined wireless local area networks (WLANs) as a complement to the IEEE 802 wired local area networks (LANs). These networks were created for high-speed data communications. Wireless personal area networks (WPANs) were designed to function in a personal operating space, which extends up to 10m in all directions. Data rates are not greater than 250Kbps. WPANs were designed to carry data over short distances and involve little or no infrastructure. This enables devices utilizing WPAN technology to be small, power efficient, and inexpensive.

The IEEE 802.15 Working Group[2] defined three classes of WPANs that are distinguished by categories that include data rate, battery drain, and quality of service (QoS). High-data rate WPANs (IEEE Standard 802.15.3) are used for multimedia applications that require very high QoS. Medium-data rate WPANs (IEEE Standard 802.15.1 – Bluetooth) are meant as cable replacements for consumer electronic devices such as mobile phones and PDAs. Low-data rate WPANs (LR-WPANs – IEEE Standard 802.15.4) are designed for applications that include such areas as wireless sensors, home automation, automotive sensing, precision agriculture, etc.

Wireless Sensor Networks

Wireless sensor networks (WSNs) focus on enabling communication to sensors and actuators without the use of wires. The development of WSNs is primarily the result of three areas. First, there is a need to reduce the cost of sensor installation, which includes materials, labor, and testing. Then, cables require the usage of connectors that can get disconnected or break due to the need to access adjacent devices. Lastly, there is the desire to gather more frequent data on a large number of systems to improve industrial operations.

WSNs provide flexible environments without the need for connectors and with improved operator safety. Some important characteristics for the design and implementation of WSNs are mentioned below.[3-4]

- *Power Consumption*: Some applications may require the usage of batteries. Thus, energy conservation is needed. A common solution is the usage of power cycling, in which the duty cycle of operation for the device is lowered.

- *Range*: RF power outputs ranging from 0dBm to 20dBm are typical for these devices. Due to the limited power, this reduces the maximum transmission distance between the transmitter and receiver. This requires the usage of multi-hop network protocols that are enabled through routing algorithms.

- *Frequency Bands*: RF spectrums are regulated by governments. However, there are special unlicensed frequency bands available for usage of devices that operate within a set of rules that govern the RF output in terms of time, frequency, and amplitude. The usage of these unlicensed frequency bands is free of charge. The frequency bands are shown in Table 1.

  In the U.S., the Federal Communications Commission[5] (FCC) regulates the RF spectrum. In Europe, the European Telecommunications Standards Institute (ETSI) coordinates regulatory efforts. Other countries have their own regulatory agencies, but many of them accept either the FCC or ETSI as proof of compliance.

- *Network Topology*: Due to the limited transmit power which reduces the maximum transmission range, multihop networks are needed. In multihop networks, the message source and destination addresses are not necessarily within range, and communication may occur through intermediate devices that relay messages. This happens with devices that are configured as peer-to-peer.

- *Self-organization*: To enable ease of installation, WSNs need to be self-organizing. Thus, each sensor device can participate in the network automatically without the need for special addressing or association. This is a feature of ad hoc networks, which are collections of transceivers that create a network without the aid of any fixed infrastructure. These networks use routing protocols to transfer data from a source to a destination that may include intermediary devices.

IEEE Standard 802.15.4

The objective of the IEEE Standard 802.15.4 is to enable low range, cost, power, and data rate wireless connectivity among inexpensive devices. There are several unique features of the standard.[3-4]

- *Duty Cycle*: The battery provides the energy for the communications. In order to minimize battery cost and maximize battery life, the energy must be taken at a low-rate. The standard allows devices to operate at duty cycles less than 1%.

- *Modulation*: For low cost implementation, only data communication is specified. Also, the protocol only supports half-duplex operation. The types of modulation supported along with the bit rates are shown in Table 1.

Table 1. Frequency Bands and Modulation Parameters

Freq Band	# of Channels	Bit Rate	Modulation
868.0 – 868.6MHz	1	20Kbps	Binary phase shift keying (BPSK)
		100Kbps	Offset quadrature phase shift keying (O-QPSK)
		250KBps	Parallel sequence spread spectrum (PSSS)
902 – 928MHz	10	40Kbps	Binary phase shift keying (BPSK)
		250Kbps	Offset quadrature phase shift keying (O-QPSK)
		250Kbps	Parallel sequence spread spectrum (PSSS)
2.40 – 2.4835GHz	16	250Kbps	Offset quadrature phase shift keying (O-QPSK)

- *Direct Sequence Spread Spectrum*: Direct sequence spread spectrum (DSSS) is one way to increase the bandwidth of the transmit signal, while simultaneously reducing the power spectral density. DSSS is achieved by applying a predetermined pseudo-random digital sequence to phase modulate the carrier. The signal is recovered in the receiver by using a replica of the pseudo-random digital sequence. The replica signal is generated in the receiver by a technique that maintains coherence with the transmitted signal.

- *Transmit Power and Receiver Sensitivity*: Devices that are compliant with the standard must be capable of transmitting at -3dBm, which is within the capability of low cost system-on-a-chip (SoC) implementations. For the 868MHz and 915MHz bands, the receiver sensitivity level is at -92dBm. For the 2.4GHz band, the sensitivity is at -85dBm. These sensitivity levels enable the use of simple receiver designs to achieve compliance with the standard.

- *Network Components*: Each network contains one network coordinator, called the PAN coordinator. It defines the structure and operating mode of the network. Other devices join the network by being granted permission from the PAN coordinator.

  Each network also contains at least one network device. The device can be defined as either a full function device (FFD) or a reduced function device (RFD). The FFD implements the complete protocol set and can function as a coordinator. The coordinator provides services to other devices of the network, such as acting as a proxy for devices outside the range of the PAN coordinator. The RFD implements only a portion of the protocol set.

- *Network Topology*: The network consists of one of two basic topologies: peer-to-peer network or star network. The star network has an FFD that functions as the PAN coordinator, which acts as a hub, and additional FFD or RFD devices that act as data terminals. The PAN coordinator is the only device in the network that forms a direct link with all of the other devices.

  The peer-to-peer network does not have a designated network coordinator, and each device is able to form multiple direct links to other devices. As a result, many complex peer-to-peer network architectures can be supported, such as the cluster-tree network. A cluster-tree can be viewed as a hierarchical tree of network device clusters. One device in the network functions as the PAN coordinator, which behaves like the root device of the hierarchical tree. Each network cluster contains one or more coordinators (FFDs) to relay messages for other devices to other clusters. The devices within each cluster form what can be considered as parent-child relationships between each other. All outer nodes (called leaf nodes) can be RFDs, since they do not need to relay any messages.

- *Channel Access*: Each network device employs a carrier sense multiple access-collision avoidance (CSMA-CA) protocol. This protocol is based on the fact that the RF channel is shared. When multiple senders are active on the same channel, collisions occur due to mutual interference. This causes messages to be unsuccessfully delivered. In order to avoid collisions, the device listens first and then transmits only if the channel is clear.

- *Error Control*: The standard utilizes a simple handshake protocol to ensure reliable data transfer. With the exception of broadcast frames (beacons) and acknowledgment frames, each received frame is acknowledged to the transmitting device. This way the transmitter knows that its message was received without any errors. If an acknowledgment frame is not received by the transmitter, the entire frame may be retransmitted.

To detect errors in a message, redundancy bits are added to the message through cyclic redundancy checking (CRC). The CRC is calculated by dividing the message by a predetermined large prime number. The quotient is disregarded and the remainder is the CRC that is added to the message. In the receiver, the same division is used with the same prime number. The remainder of the division process should be zero if there were no errors in the message.

Introduction to the MaxStream XBee RF Module

One such device using the IEEE 802.15.4 packet radio specification is the MaxStream XBee RF wireless module[6]. This device utilizes a radio transceiver operating in the license-free Industrial, Scientific, and Medical (ISM) RF band at frequencies from 2.400GHz to 2.484GHz. The standard device has a maximum line-of-sight range of 300 feet and a maximum data throughput of 250Kbps. The device is intended to be used in conjunction with the internal RS-232 UART found in most microcontrollers and computers for the purposes of remote/embedded data collection and wireless sensor network development. It has a transmit power level of 1dBm and a receiver sensitivity of -92dBm. The device can also be configured in broadcast mode to allow wireless operation of devices that traditionally utilize RS-485 interfaces, which are common among PLCs and other industrial automation and control equipment. The module features an upgradeable firmware with 128 bit encryption for security. Possible networking topologies include point-to-point, peer-to-peer, and point-to-multipoint (star). However, it does not support mesh networking which requires full-function routers. The module is powered using a 3.3V DC source, and devices that connect to the module must use 3.3V CMOS digital levels. Thus it is required to use an RS-232 transceiver IC designed for operation at 3.3V.

Laboratory Projects

Three laboratory projects using the XBee RF module are shown. The first lab is designed to establish communication with the XBee module through the PC RS-232 port and to transfer a text file between 2 XBee modules. HyperTerminal is utilized to configure the module for Unicast operation through the AT command set and to demonstrate file transfer capabilities.

The second lab is used for the development of a LabVIEW[7] terminal program in order to use software for communication with the XBee module. Module 1 is configured as a remote sensing unit, transmitting temperature data to another module. Temperature information is collected from a temperature sensor interfaced to a PC DAQ card using LabVIEW. Module 2 is configured as a temperature collection unit. The module receives the transmitted temperature data from Module 1. The data is then sent into the PC through the serial port so that it can be displayed.

The third lab configures a wireless sensor network with multiple modules arranged in a star network configuration. Two modules are configured as end devices for remote temperature sensing. A third module is configured as the PAN coordinator. The coordinator receives transmissions from both end devices, displays the temperature sent from each unit, and stores the results in text files.

Serial Communication with the XBee RF Module

One objective for the first lab is to establish communication with the module through the PC serial port. In order to interface the module to the serial port, the schematic in Figure 1 is implemented.

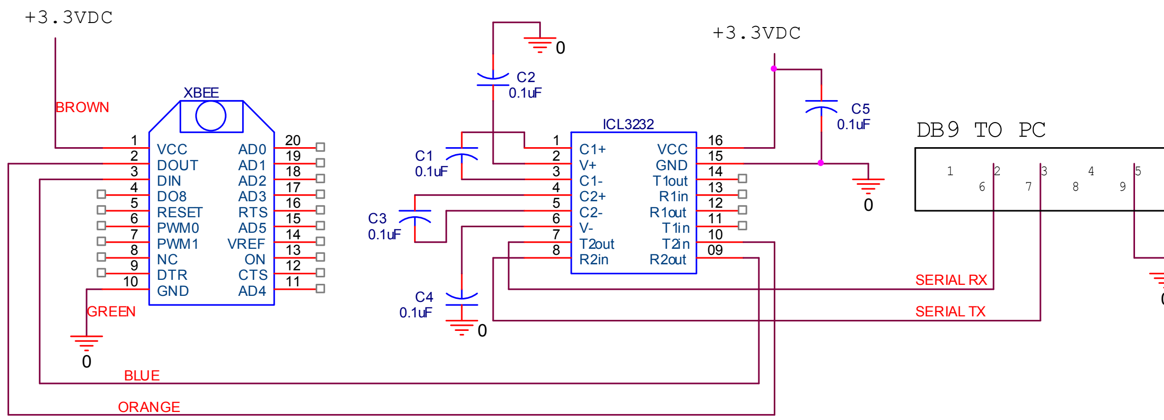

Figure 1. Schematic for XBee module

As indicated on the datasheets for the XBee module[8], only 4 connections need to be made: DIN, DOUT, VCC, and GND. It is required to use either an ICL3232[9] or similar IC to translate the RS-232 voltage level signals to and from 3.3V CMOS levels. It is not necessary to connect the hardware flow control lines for RTS and CTS.

In order to configure the module, HyperTerminal can be used to send the appropriate AT commands. An escape code sequence is required to request the module to enter the command mode. The default escape character is the plus sign '+'. In order to successfully enter the command mode, at least 1 second must elapse from the end of normal transmit traffic, followed by '+++' (3 escape characters), and then another second must elapse. The reason for the waiting period (guard time) is so that the module is certain that the command mode is desired, and not the transmission of those characters. The module will remain in command mode until either 10 seconds elapse with no commands sent, or the module is instructed to exit command mode, whichever comes first. The guard times, escape character, and command mode timeout can be changed if required. Operating parameters of the module can be changed once in the command mode by using AT commands similar to those used by a telephone modem. However, the changes will not remain until a command is sent to write the parameters to nonvolatile memory, and some changes will not take effect until the module is reset. The AT commands are not case-sensitive. AT command parameters are displayed and entered in hexadecimal, and leading zeros are not needed. When the AT command is executed successfully, the module will respond with an OK message. For this lab, the following AT commands are used for one module.

- +++ (Escape sequence to enter command mode)
- ATRE<CR> (Resets the module to factory defaults)
- ATCT FFFF<CR> (Sets command mode timeout to maximum value – default is 10 sec)
- ATCH C<CR>(Sets the RF channel to C_{hex})
- ATID 1234<CR> (Sets the PAN ID to 1234_{hex})

- ATMY 1<CR> (Sets the source address to 1)
- ATDH 0<CR> (Sets destination high address to 0 for 16 bit addressing, not 64 bit)
- ATDL 2<CR> (Sets the destination low address to 1)
- ATWR<CR> (Commit the settings to nonvolatile memory)
- ATND<CR> (Node discover command to see other devices on the network)
- ATCN<CR> (Exit the command mode)

The second module is set up in a similar manner, except that it is configured for a source address of 2 and a destination low address of 1. In order for both modules to communicate with each other, they need to have the same RF channel (C_{hex}) and the network identification (1234_{hex}). The values for these parameters were chosen arbitrarily, within the constraints specified in the datasheets.

When both modules are appropriately configured, HyperTerminal is used to verify communication between the 2 modules through text messaging. In addition, HyperTerminal is used to transfer a file through the 'Zmodem with Crash Recovery' protocol.

Temperature sensing system using the XBee RF Module

The purpose of the second lab is to design a remote temperature sensing system consisting of two modules. The full-scale temperature for the system is 100˚C and the temperature resolution is 0.01˚C. An LM35 Temperature Sensor[10] is used to detect temperature, and its output voltage is 10mV/˚C. The temperature sensor is interfaced directly to the PC DAQ card. LabVIEW is used to read the voltage from the DAQ card and convert the voltage to a temperature in ˚C. The integer and remainder portions of the temperature are sent as two bytes to the serial port for transmission to Module 1, which is configured as the remote temperature sensing unit. The LabVIEW block diagram for the temperature transmitter is shown in Figure 2.

Module 2 is configured as the temperature receiving unit. Using LabVIEW, two bytes are read from this module through the serial port. Since the two bytes represent the integer and remainder portions of temperature, they are combined together to determine the temperature in ˚C and displayed on the front panel. The block diagram for the temperature receiver is shown in Figure 3.

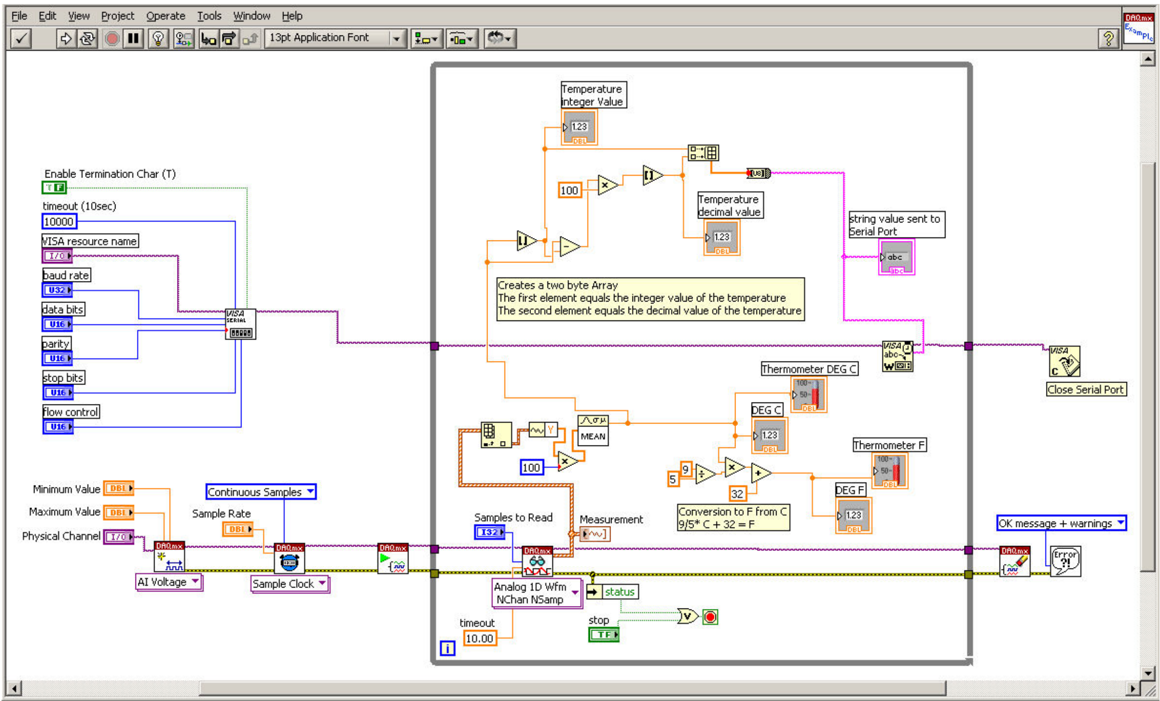

Figure 2. Temperature Transmitter LabVIEW Block Diagram

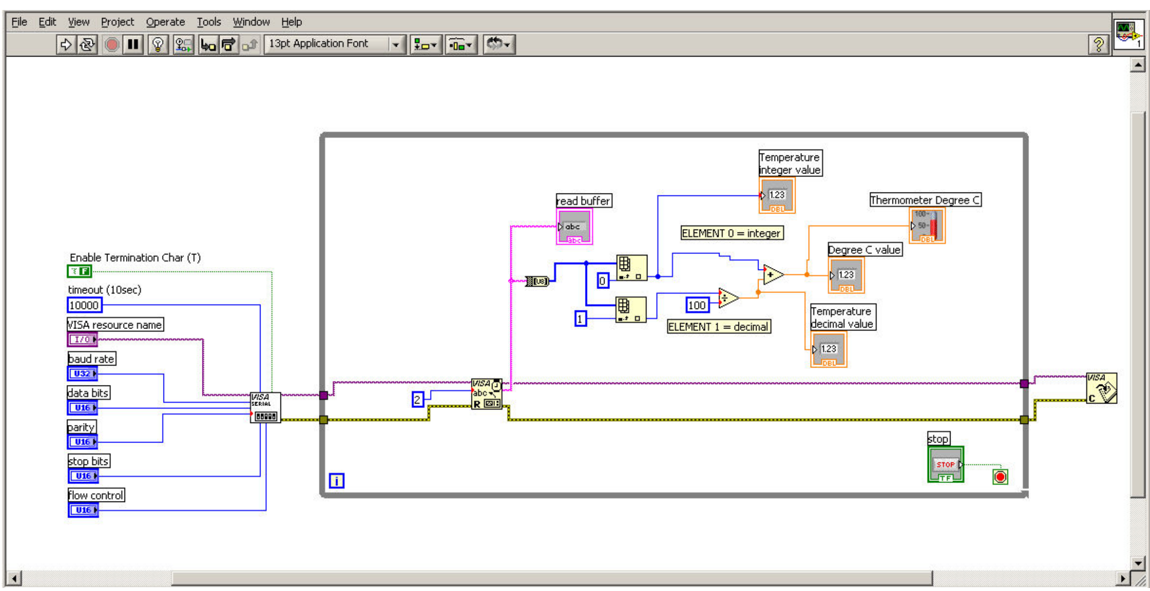

Figure 3. Temperature Receiver LabVIEW Block Diagram

Development of a Wireless Sensor Network

The purpose of the third lab is the development of a wireless sensor network (WSN) using three modules arranged in a star topology. Using HyperTerminal, one module is configured as the PAN coordinator. The same AT commands as shown previously for the first lab are used for this module, with the addition of the following commands.

- ATCE 1<CR> (Enables the unit to function as a coordinator)

- ATA2 4<CR> (Allows devices to associate with it; uses static ID and CH)
- ATNI CORD<CR> (Optional node text identifier)

The additional AT commands for each end device are below.

- ATCE 0<CR> (Enables the unit to function as an end device – not a coordinator)
- ATA1 4<CR> (End device only associates with coordinator at static ID and CH)
- ATNI ENGINE1<CR>(Optional text node identifier – the 2nd end device uses ENGINE 2)

In order to verify the proper configuration of the star network, HyperTerminal is used to enter the node discover command. The following results were received.

- FFFE (Source address – 'my' address – of the end device)
- 13A200 (Serial high address)
- 400226E3 (Serial low address)
- 3A (Signal strength in dBm; for this example is -58dBm)
- ENGINE1 (Node text identifier)

- FFFE
- 13A200
- 40022728
- 38
- ENGINE2

Association from end devices to the PAN coordinator is verified, since the 'my' addresses for the end devices were automatically changed to FFFE. The FFFE address indicates that each end device will use their unique 64-bit address (serial number) as their source address and will no longer use their previously set 16-bit address. The association can also be verified by looking at the destination low address for each end device. It should be changed to the 'my' address of the coordinator.

Each end device is designed to transmit temperature obtained from an LM35 Temperature Sensor. The sensor is interfaced directly to the PC DAQ card, as previously described in the second lab. LabVIEW is used to read the voltage from the DAQ card and convert the voltage to temperature in ˚C. The temperature, serial number of the device, node text identifier, date, and time are all sent through the PC serial port to the end device for transmission. The WSN temperature transmitter LabVIEW block diagram is shown in Figure 4.

Both end devices will be transmitting data simultaneously to the coordinator device. The coordinator LabVIEW program will use the PC serial port to read the data transmitted from both end devices, display the temperature for each end device on the front panel, save the time stamped data for each end device in a separate file, and display statistical results (maximum, minimum, and average temperatures) for each end device on the front panel. A portion of the WSN temperature receiver LabVIEW block diagram is shown in Figure 5. A sample of the results stored in the text file for end device 1 (Node text identifier ENGINE1) is shown in Figure 6.

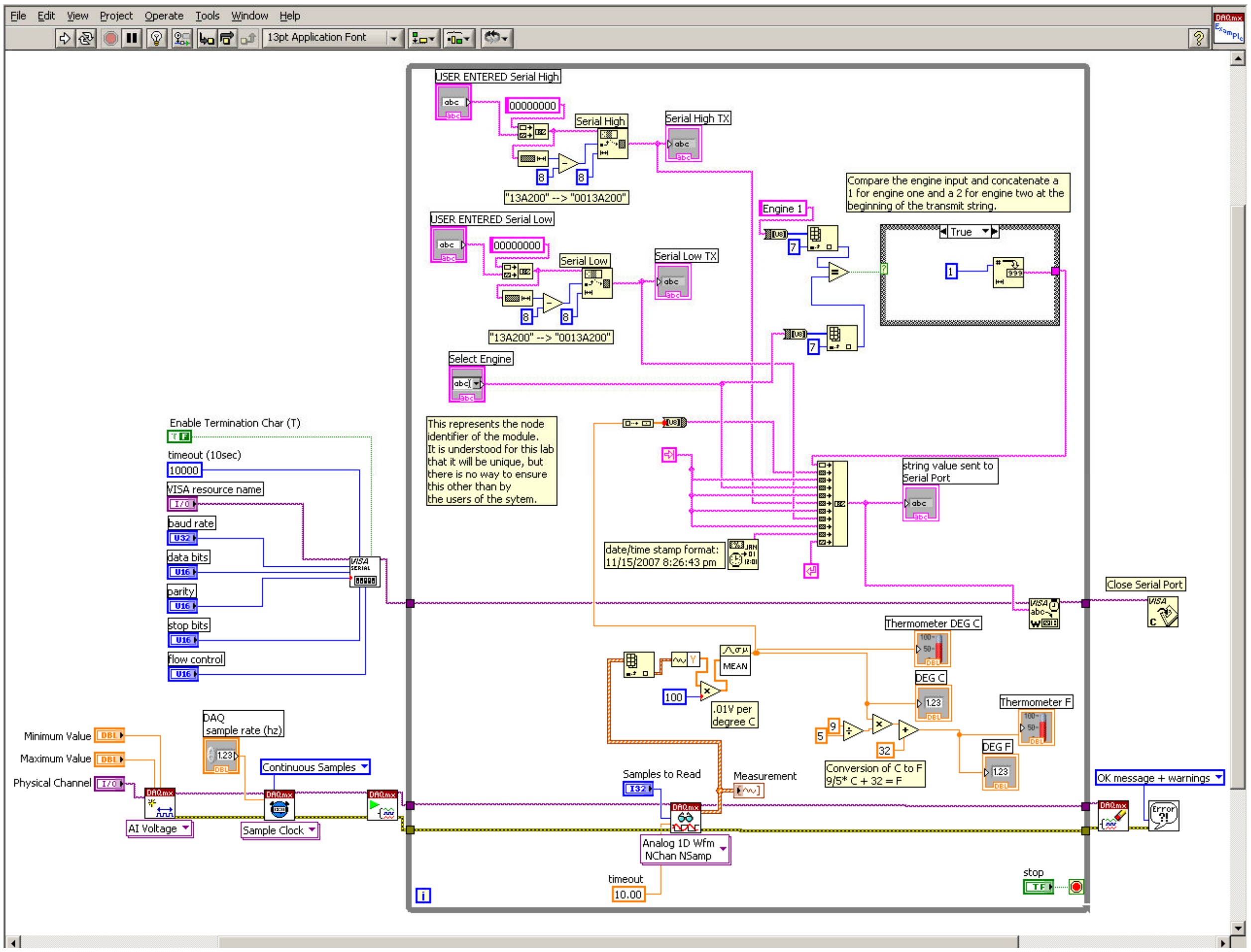

Figure 4. WSN Temperature Transmitter LabVIEW Block Diagram

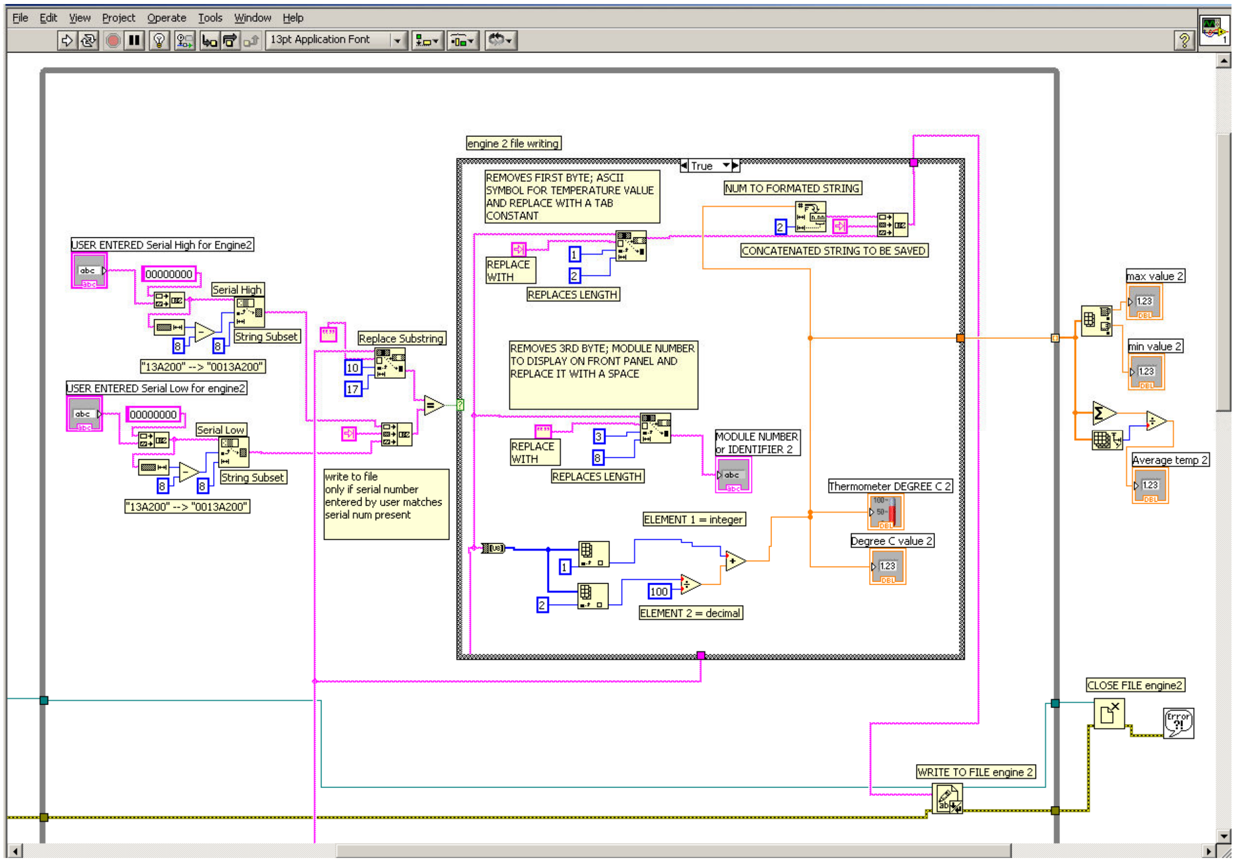

Figure 5. WSN Temperature Receiver LabVIEW Block Diagram

25.09	1	Engine 1	0013A200	40022793	11/27/2007 12:19:43 PM
25.09	1	Engine 1	0013A200	40022793	11/27/2007 12:19:44 PM
25.09	1	Engine 1	0013A200	40022793	11/27/2007 12:19:45 PM
25.09	1	Engine 1	0013A200	40022793	11/27/2007 12:19:46 PM
25.09	1	Engine 1	0013A200	40022793	11/27/2007 12:19:47 PM
25.09	1	Engine 1	0013A200	40022793	11/27/2007 12:19:48 PM
25.09	1	Engine 1	0013A200	40022793	11/27/2007 12:19:49 PM
25.09	1	Engine 1	0013A200	40022793	11/27/2007 12:19:50 PM

Figure 6. Text File for ENGINE1

Conclusions

The first lab involved the development of a point-to-point network with demonstration of file transfer capabilities using HyperTerminal. The second lab was the design of a remote temperature sensing system using the LM35 Temperature Sensor. Two XBee modules were used in the design, with one module transmitting temperature information and the other module collecting the information. Both devices were interfaced to PC serial ports and LabVIEW was used for software development. The third lab was the development of a wireless sensor network designed to monitor temperature at two remote sites. At each remote site, an XBee module was used to transmit temperature information to a third XBee module, configured as a PAN coordinator. LabVIEW was also used for software development. Each end device transmitted the temperature, serial number, node text identifier, date, and time. The coordinator device received the information, and software enabled the temperature data to be stored in files and statistical calculations to be performed on the data.

One objective of this paper was to present laboratory activities that will help meet some of the ABET networking requirements of ECET Programs as well as CMPET Programs. All three labs focus on the application of networks to computer systems. Similar wording is specified for these programs in the ABET documentation for accrediting engineering technology programs[11]. With the usage of the MaxStream XBee RF module, this material also provides a resource to aid instructors interested in including wireless technology within EET curriculums.

Recommendations

Since the author anticipates using these projects for a communications systems course within the EET option of an ECET Program, the PC RS-232 port was used for communicating with the module. It would be more practical to use a microcontroller instead of a PC, but the prerequisites for the communications course do not include a microcontroller's course.

A Series 2 XBee RF module has been introduced from MaxStream. This module is similar to the Series 1 module, but it supports mesh networking with full-function routers. This would enable more complex networking topologies.

Acknowledgement

I would like to thank Nate Steele, a senior in our program, who assisted with the design of the laboratory activities.

Bibliography

1. Web Site http://www.ieee802.org/11/.
2. Web Site http://www.ieee802.org/15/.
3. Gutierrez, J., E. Callaway, and R. Barrett, *Low-Rate Wireless Personal Area Networks*, 2nd Edition, IEEE Press, 2007.
4. Wu, Shih and Y. Tseng, *Wireless Ad Hoc Networking*, Auerbach Publications, 2007.
5. Web Site http://www.fcc.gov.
6. MaxStream XBee Series 1 RF Module, part # XB24-ACI-001.
7. National Instruments, LabVIEW, Version 8.2.
8. Web Site http://maxstream.net.
9. Intersil, part # ICL3232CP.
10. National Semiconductor, part # LM35DZ.
11. Web Site http://www.abet.org.