
AC 2012-3329: IMPLEMENT SMART SENSORS WITH WIRELESS COMMUNICATION PROTOCOLS AND EMBEDDED MICROCONTROLLERS IN A CAPSTONE PROJECT DESIGN

Dr. Steve C. Hsiung, Old Dominion University

Steve Hsiung is an Associate Professor of electrical engineering technology at Old Dominion University. Prior to his current position, Hsiung had worked for Maxim Integrated Products, Inc., Seagate Technology, Inc., and Lam Research Corp., all in Silicon Valley, Calif. Hsiung also taught at Utah State University and California University of Pennsylvania. He earned his B.S. degree from National Kauhsiung Normal University in 1980, M.S. degrees from University of North Dakota in 1986 and Kansas State University in 1988, and Ph.D. degree from Iowa State University in 1992. Hsiung can be reached at shsiung@odu.edu.

Mr. Cory Terell Gleason, Old Dominion University

Cory Terell Gleason graduated from ODU in 2012 and was in the U.S. Navy from 2003 until 2007.

Mr. Lacides Agustin Osorio, Norfolk Ship Support Activity Regional Maintenance Center

Born in Colombia, South America and raised in Queens, New York. he graduated from Brooklyn Technical High School in 1983, and enlisted in the U.S. Navy. In 2009, he retired from the Navy after 25+ years of dedicated service, and also graduated from Old Dominion University with a bachelor's of science degree in electrical engineering technology. He currently works as an Electrical Engineering Technician at Norfolk Ship Support Activity Regional Maintenance Center, and continues to be a part of the Electrical Engineering Technology Department at Old Dominion University.

Mr. Charles H. McGinn, Old Dominion University

Implement Smart Sensors with Wireless Communication Protocols and Embedded Microcontrollers in a Capstone Project Design

Introduction

Wireless communication has become popular and widely used in our daily lives. Their applications are: Cellular Wireless for telephone systems, data collection, voice communication, and other mobile or extremely remote devices, Bluetooth for low-power applications in short range and moderate data bandwidth, Proprietary ISM (industrial, scientific, medical) protocols used in open frequency bands from 260 to 470 MHz, 902 to 928 MHz, and 2.4GHz, 802.11/WiFi in wireless data communications, 802.15/ZigBee for mesh networks of sensors and controllers, and Z-Wave for low speed wireless protocol of home electronics devices to intercommunicate using reliable protocol that easily travels through walls, floors, and cabinets¹.

Sensors with embedded intelligence and integrated with cost effective wireless protocols have been recognized as smart sensors in many applications, such as smart home appliances, home automation, green technology in energy conservation and harvesting, and remote data logging etc². This application project is implemented in the classification between Proprietary ISM, ZigBee, and Z-Wave wireless applications. It is built based on the MRF24J40MA (2.4GHz RF modules) that follows the IEEE 802.15.4TM-2003 rules⁷, standards, and software protocols designs with SPI (Serial Peripheral Interface)⁹ interfacing to a PIC16F877A microcontroller. The project uses three 2.4 GHz RF modules (MRF24J40MA), and three PIC16F877A units hosted in three previously developed low cost PIC microcontroller training systems³. The sensor stations are designed as Slave units and responsible for conditioning and reporting temperature, humidity, and atmospheric pressure. The control unit is categorized as a Master station and responsible for interacting with user/host to decide when, where, and how to report the data to the inquiries. In addition to sensors on the Slaves station, there are assistances from real time clock and external serial EEPROM devices to provide time stamped real time data for future inquiry from the Master. All the communications between the Master and multiple Slaves are through wireless RF signals with customized software protocol designs⁴.

Review of Smart Sensor Applications

The sensors are essential elements in many control system applications. They have been used almost anywhere in the past. Many of the applications are made through wires between the sensors and control element/module. These are either passive or active devices that have no intelligence but provide either resistance, voltage, or current signals per sensing conditions and most of them require conditioning circuits to deliver the needed signals. Thanks to the wireless and digital technologies, those sensors can now become intelligent with the assistance of microcontroller in an embedded system design. The smart sensor applications have become a new trend of design in product development. There are different standards such as Cellular, Bluetooth, Proprietary ISM, WiFi, ZigBee, Z-Wire, etc. in an embedded smart sensor design to meet the application objectives that are based on the power consumption, communication range, data rate, reliability, and cost. Table 1 presents these comparisons¹.

The Implementation of the Sensors with MRF24J40MA Modules

The MRF24J40MA is a transceiver that runs on 2.4 GHz license free radio frequency⁵, a product manufactured by Microchip Inc. that follows the IEEE 802.15.4 standard⁷. The price of the module is at \$9.95 in a single quantity. It has low current consumption – at TX 23mA - RX

19mA - Sleep mode of 2 μ A, Hardware CSMA-CA Mechanism, Automatic ACK response Hardware security engine for AES-128 with CTR, CCM, and CBC-MAC modes with Automatic packet retransmit⁸. This module includes everything needed to perform 2.4 GHz wireless transmitter and receiver functions with a choice of 16 different channels and provides several ready available features, such as an antenna, resonator, Slave SPI interface, interrupt driven power saving mode, and operates on 3.3VDC power source⁸. It is packaged in a 12-pin interface module. All the user needs to do is provide a 3.3-volt supply and address the module with SPI protocols. This article demonstrates the use of the PIC training system, a system designed and built by a previous funded project for distance and on-campus teaching and learning³, with a built-in plug for the required power and SPI interface communication⁹. The designs are based on prior developed wireless communication protocols⁴ following the IEEE 802.15.4 standards between one Master module and two Slave modules on three PIC training systems hosting three PIC16F877A PICs as the main MCUs (MicroController Units)¹⁰ that monitor three different sensor elements: (1) temperature, (2) pressure, and (3) humidity with a serial real time clock chip (RTC)¹² and data logged in an external EEPROM¹³.

Wireless Standard	Power	Range (meter)	Data Rate (bit per second)	Data Reliability	Cost
Cellular	High	1-100m	14.4Kbps-3.6Mbps	High	High
Bluetooth	Medium		1-3Mbps	High	Medium
Proprietary ISM	Varies	Varies	Varies	Varies	Varies
802.11/WiFi	Medium	100m-Several Km	10-100+ Mbps	High	Medium
802.15/ZigBee	Low	10-100m	20-250 Kbps	Medium	Low
Z-Wave	Low	15-150m	9.6-40 Kbps	Medium	Low

Table 1. The Comparison of Wireless Standards

The Master station has a LCD module and a 3*4 matrix keypad and responsible for user interaction interface for the logged data by the Slave. The two Slaves sensor stations are the remote units that are solely monitoring the temperature, pressure, and humidity with the date and time stamped in the external EEPROM. All communications/interfaces between the modules in the training systems are based on the SPI standard⁹. The wireless communications made between a Master and two Slaves are implemented with the MRF24J40MA transceivers. Photos 1 and 2 demonstrate the setup of the Master and Slave sensor stations.

The Hardware Design of Smart Sensor Applications

The hardware designs use three PIC 16F877A microcontrollers and three MRF24J40MA RF modules hosted in three PIC training systems⁴ that was designed and developed by the author in a previous project. Data from the remote sensor units can be gathered and transmitted wirelessly then displayed on a LCD in a base control unit. Two different types of units are setup in this project: (1) a Master that is responsible for the interface between the user and system inquiring and responding via keypad and LCD input and output displaying, (2) two Slave units to communicate with the Master control unit. Each of the Slave trainers is identically wired. The only difference is the PANID and addresses that is manipulated with software. The LCD is used to indicate the received data that the Master transmits. There is no need of the keypad and LCD on the Slaves, as only the Master interfaces with the user.



Photo 1. Master Station Trainer Board Setup

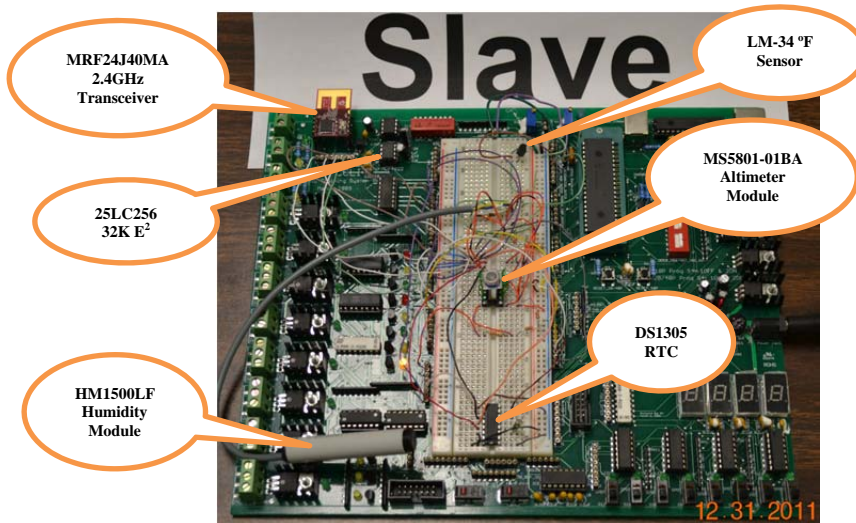
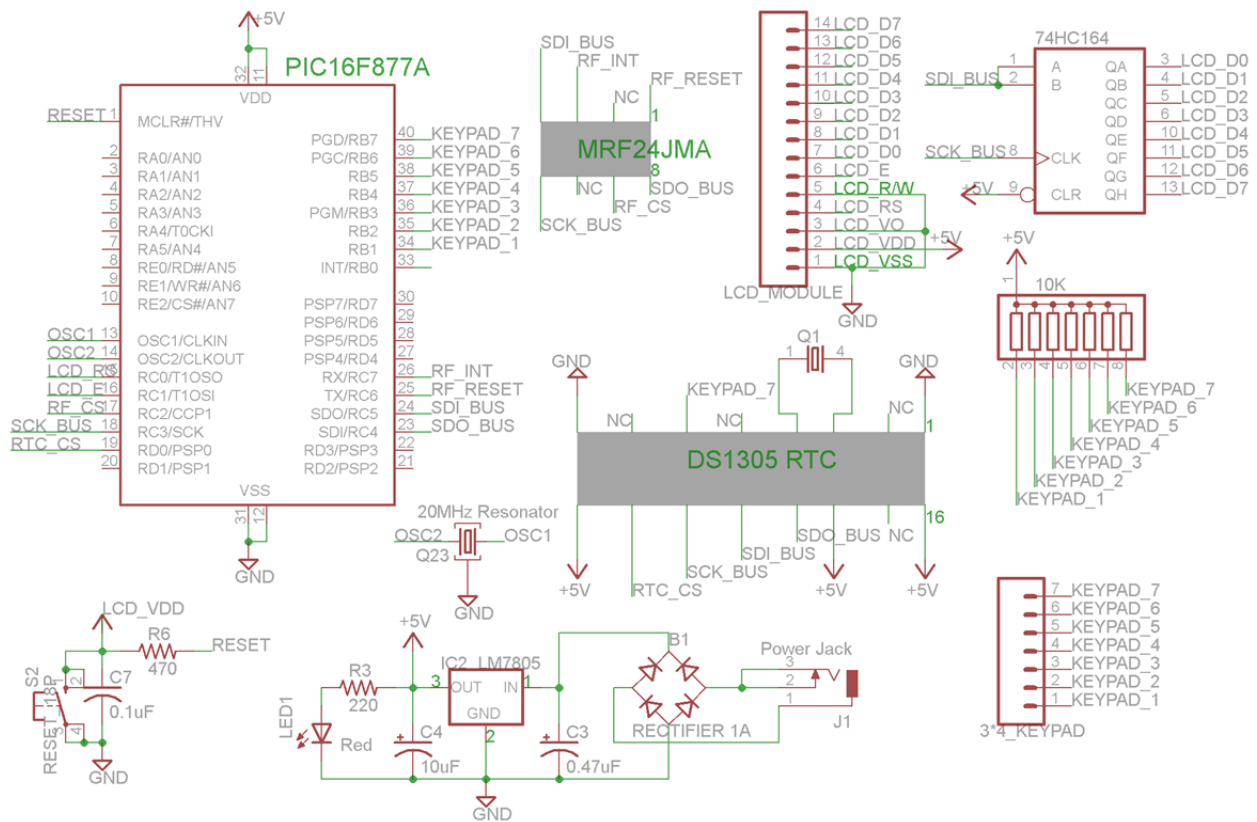


Photo 2. Slave Sensor Station Trainer Board Setup

The Slave unit communication and operation are accomplished by 2.4 GHz RF and SPI protocols along with 24 bit and 64 bit math routine designs. The sensors used in this project are: (1) LM-34 Precision Fahrenheit temperature sensor¹⁴, (2) HM1500LF relative humidity module¹⁵, and (3) MS5801-01BA Altimeter module¹⁶. Each sensor station reads the data from the LM-34 (temperature) and HM1500LF (humidity), followed by the ADC routine, and completed with a 24 bit math linearized calculation for data recording in an EEPROM. The MS5801-01BA Altimeter module pressure signal is sent straight to a 64 bit math routine calculation and stored in the same EEPROM with a different address location followed by the humidity data.

The PIC 16F877A is a 40 pin microcontroller with four I/O ports, SPI interface, and an integrated A/D converter¹⁰. Port A is interfaced to the temperature and humidity sensor. The sensors are connected to MCU's AN0 and AN1 of Port A as analog inputs. After the MCU's ADC conversion, a 24-bit math routine reports the accurate temperature and humidity in a 24 bit binary number. The real time data consisting of the date, minute, and second are stored in the EEPROM for RF communication to the Master depends on which interrupt has occurred through

SPI protocol. Port C is the MCU's SPI module responsible for the appropriate SPI interfacing. The SPI is implemented as the bus connections to all the modular devices on the Slave sensor station where RC3 is the serial data clock, RC4 is the serial data input, and RC5 is the serial data output. These are the bus connections that apply to all other SPI devices such as RTC (DS1305), EEPROM (25LC256), RF module (MRF24J40MA), Pressure (MS5801-01BA), and humidity (HM1500LF) sensor. Port B, RB0 is configured as the interrupt from the RF module and RB7 is the interrupt for the RTC. The RTC is used to time stamp all data prior to being stored to the EEPROM. The sensor stations are constructed identically except for the difference in address assignments. The RF and SPI communication in the Master is the same as the Slaves except Port B is used for the matrix keypad and LCD interface is made through a 74164 shift register to the same SPI bus. There is no interrupt (INT) algorithm implemented in the Master station, except polling on the RF module INT signal. The hardware wiring diagram of the Master and Slave are presented in Pictures 1 and 2 as follows:

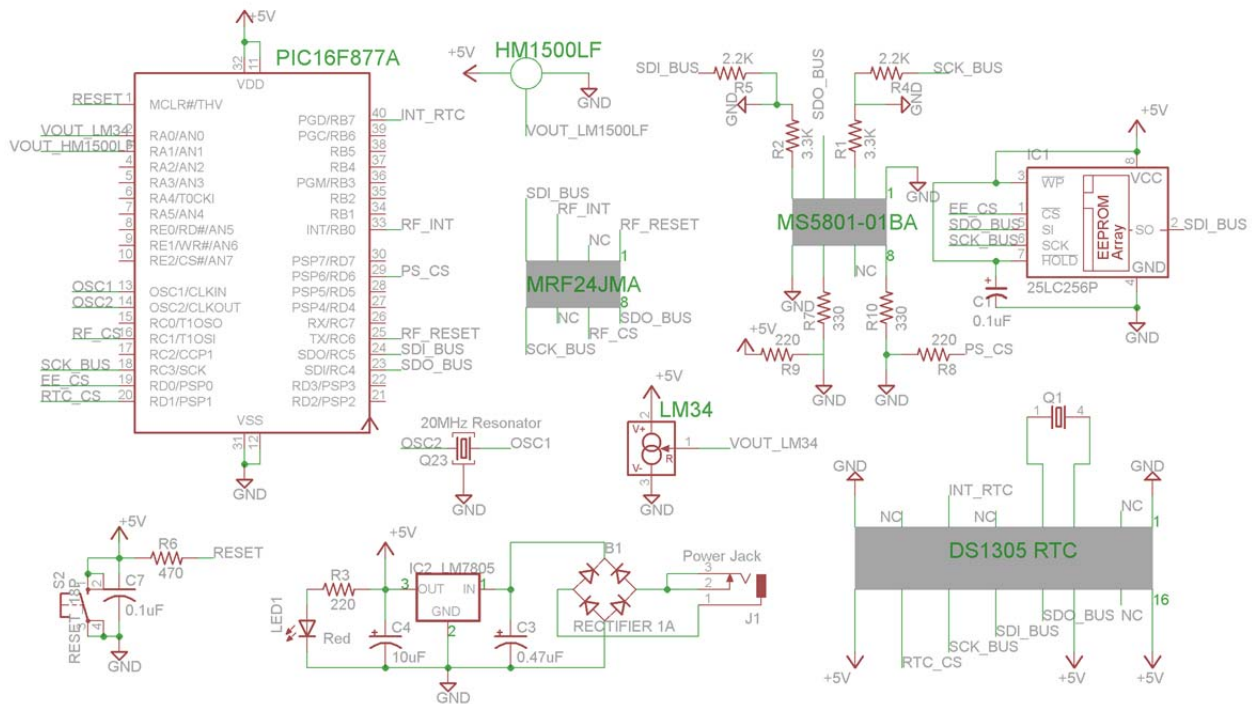


Picture 1. Hardware Schematics of the Mater Station

The Software Design of Smart Sensor Applications

The software designs are based on previous MRF24J40MA project implementations⁴. The RTC is used as an interrupt source for the Slave MCU. A five minute alarm has been programmed in the RTC set subroutine. Every five minutes the alarm will interrupt/wakeup the MCU to allow data to be read from the sensors and stored to the EEPROM. This 25LC256¹³ has 32k bytes memory and 1,000,000 erase cycles. The every 5 minute interrupts recording algorithm is designed to maximize the use of the EEPROM memory and accommodate its page boundary limit in addressing of the space. This design can record the remote data up to a week's

worth of information before it starts overwriting itself. Starting at address 0X0000 it will store date, hour, minute, second, temperature, humidity, and pressure until it has reached its last address 0X7FFF. This is a SPI EEPROM follows the same protocol as all other SPI devices on the same SPI bus. When the Master station requests timed data with its own RTC from the Slave sensor station, it will respond with associated records stored in its EEPROM and communicate through the RF modules transmitting and receiving protocols⁴. The RF module is a 2.4 GHz transceiver that allows the microcontrollers to transmit and receive wirelessly in this network. This module has four different subroutines: (1) RF initialization, (2) RF transmit, (3) RF receive, and (4) SPI subroutine⁴. This module is implemented on all three stations (one Master and two Slave units). Within the RF Initialization subroutine a PANID is programmed. This ID will only communicate with devices of identical PANIDs. Unlike the PANID, different addresses are also given to the stations for identification. This allows the Master unit to communicate to a certain Slave sensor station.



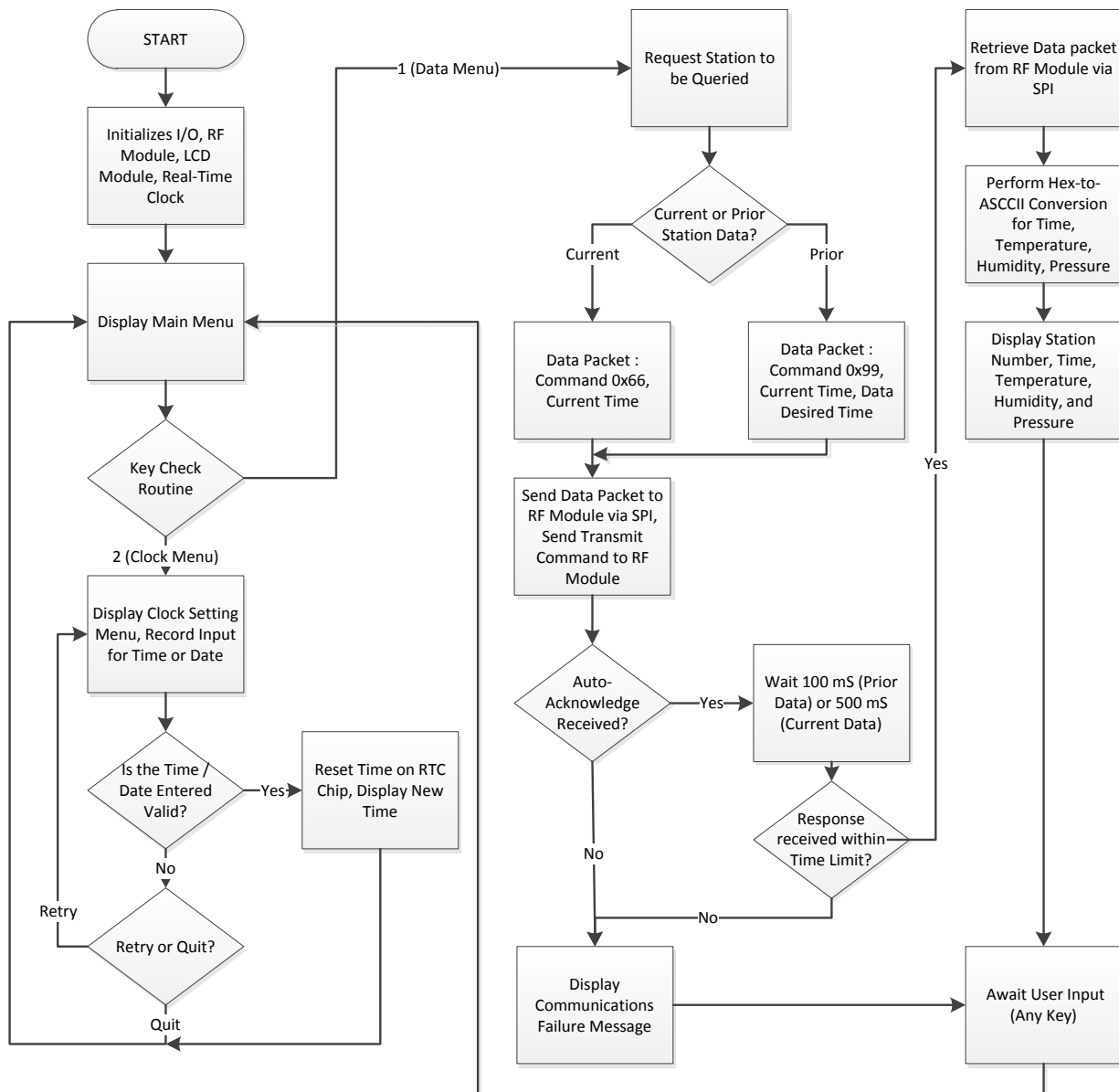
Picture 2. Hardware Schematics of the Slave Sensor Station

The main program for the Master unit has five menus: (1) a root menu, (2) a clock setting menu, (3) a current data display menu, (4) a prior data display menu, and (5) a station selection menu. The root menu displays the current time from the RTC, and requests user input to select one of the other menus. The clock setting menu allows the clock on the Master to be set with the Slave unit. After the time and day of week are entered, it validates the 24-hour clock times and dates. If the answer is yes, it asks for confirmation and the date & time will be set to that value. If it is not, it responds to the user for re-enter. The station selection menu presents a list of the current sensor stations to select the current station for data requests to be sent to.

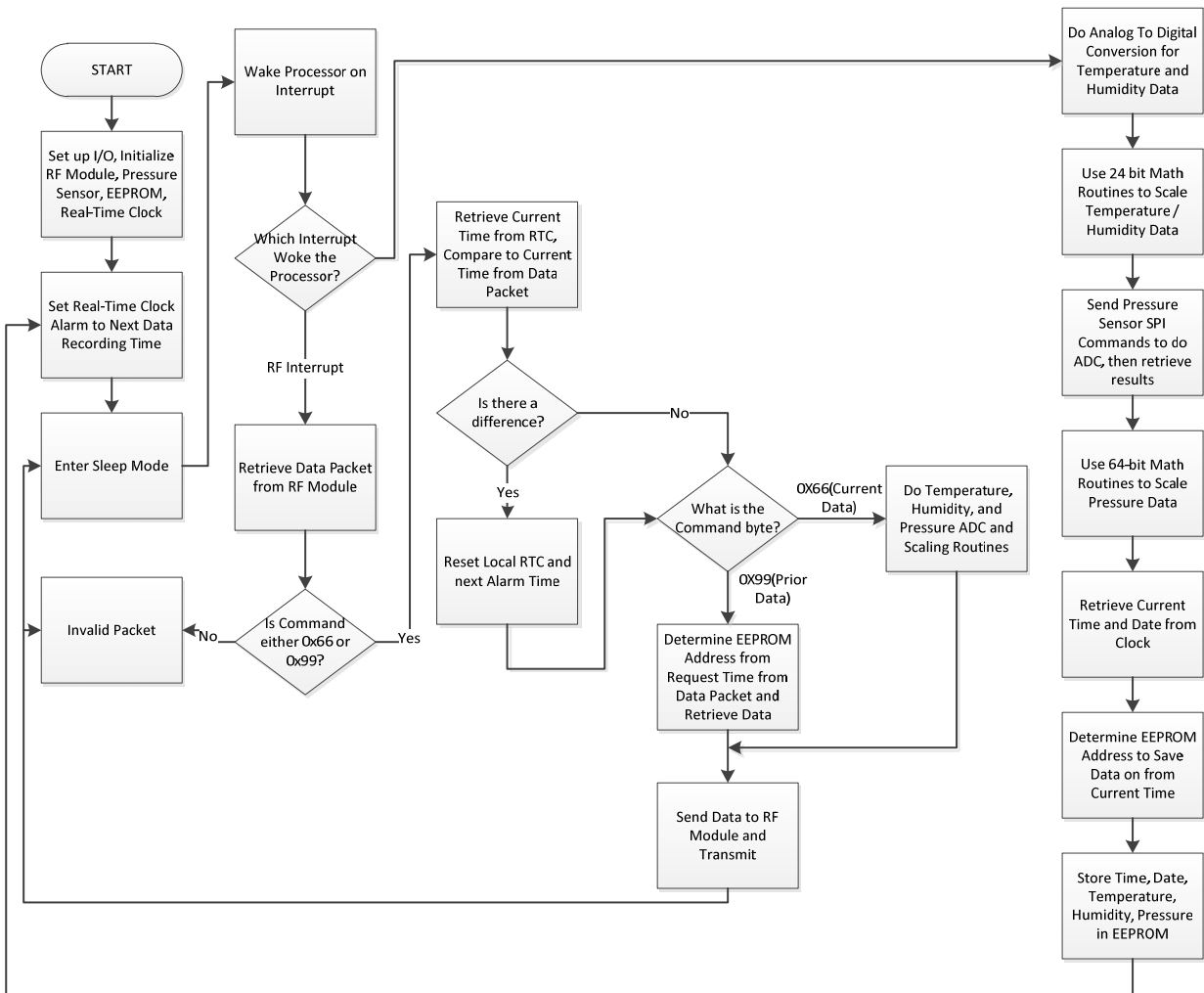
The current data display menu requests the selected station to immediately display the current conditions. The prior data display menu uses the same time entry format as setting the clock. It sends this time to the sensor station as the requested data. The sensor station uses this

time to look up prior data in the EEPROM to transmit to the master. In addition, the sensor Slave unit will not respond if the command byte is not a recognized value.

If the Master RF Module does not receive the hardware auto-acknowledge from the requested Slave sensor station, the MCU will immediately display a “Communications Failure 1” message, and wait for user’s input before returning to the main menu. When the auto-acknowledge is retrieved, the MCU will start a timer (200 mS for prior data, 500 mS for current data). If a data packet has not been received after this time out period, the display will indicate “Communications Failure 2” instead. This allows the user to determine whether the Slave sensor station is without power or powered but the MCU is not responding properly. Pictures 3 and 4 are the simplified flowchart representation of the software designs for the Master and Slave sensor stations.



Picture 3. Master Station Software Design Flowchart



Picture 4. Slave Sensor Station Software Design Flowchart

Conclusion and Recommendations

The smart sensor is capable of making decision and interacts with its surroundings and interprets the sensed physical conditions. Combining with a wireless network makes the sensor with endless possibilities. The main focus of this demonstration project is to provide the student a unique learning experience in programming artificial intelligence for use with sensors and the use of wireless communication.

This demonstration of an EET capstone project uses a 2.4GHz module that has all the needed hardware except the 3.3V power source. The PIC training system has a plug-in type of connector and needed power source for this MRF24J40MA to be used easily with only 5 lines of SPI interface to the MCU. The design is concentrated on the software and makes the changes of the application very flexible. It is currently implemented with one Master and two Slaves' communications on monitoring and reporting the temperature, pressure, and humidity in any location. The intelligence can certainly be added into the software design and more Slaves can be incorporated if the remote sensing application is needed. Every system is identical in hardware

construction but the unique address and PANID, so the ability to add more sensor stations is simple and convenient.

Wireless communication has been widely used for decades. The current advancements in technology have made wireless application more attractive and our daily lives heavily dependent on it. Also, these technologies make it possible to merge complex circuits into a single integrated circuit. The application of an embedded system design is aimed at using a simplified version of a microcomputer, a microcontroller, in a system design with minimal support of external components. A closer look at the new products available in the market today reveals a growing trend in the integration of systems. Certainly, there are security issues associated with wireless communication, but that can be compensated with embedded systems using intensive software protocols. Both hardware and software provide options in designing security choices to overcome the concerns. The merging of these systems is the major focus in newly developed industrial products. The reflection of the current developments in technology, with regard to computer/electronic technology curricula and project implementations, are crucial to the success of the program and the enrolled students. Engineering technology focuses on both “hands-on and minds-on” design work and the practice is to integrate existing technology products into real world applications. Through the implementation of this project, students have been provided the opportunities to learn about current technological concepts and to prepare and qualify for those high-tech jobs for today and the future.

Reference

1. Lebson, Steve (2010). “Welcome to the Wireless Menagerie: RF Bands and Protocol Choice for Embedded Development”. TechZone Wireless Solutions Magazine. Page: 10, TZW101.US, Winter.
2. Lattibeaudiere, Derrick and Venuturumilli, Abhinay (2011). “Microcontroller and Wireless Connectivity in Smart Appliances”. TechZone Wireless Solutions Magazine. Pages: 41-43, TZW111.US, Winter.
3. Hsiung, Steve, Hackworth, John, Ritz, John, and Eiland, James (2009). “Designing a Microcontroller Training Platform for Active Distance Learning Engineering and Technology Students”. Journal of Engineering Technology. Pages: 34-44, Volume 26.
4. Hsiung, Steve, Deal, Walter, Osorio, Lacides, Henderson, Mathrew (2011). “Design and Implement a Cost-Effective Wireless Communication Project on a PIC Training System”. Proceedings of the 2011 American Society for Engineering Education Annual conference & Exposition. Engineering Technology Division, Session 0574, Paper 721, Vancouver, Canada.
5. Federal Communications Commission Code of Federal Regulations Part 15.23 Title 47 (1997). Customer Service Branch, MS 1300F2, 7435 Oakland Mills Road, Columbia, MD 21046, Retrieved December 16, 2010 from: www.fcc.gov
6. Bible, Steven. “Why a Wireless RF Module? Making the Make vs. Buy Decision”. TechZone Wireless Solutions Magazine. Pages: 62-64, TZW101.US, Winter.
7. IEEE 802.15.4 Standard (2003). Retrieved January 10, 2011, from <http://www.cs.jhu.edu/~cliang4/public/datasheets/802.15.4-2003.pdf>
8. MRF24J40MA Data Sheet (2008). Retrieved January 10, 2011, from <http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en535967>
9. SPI (2010). A Synchronous Serial Data Link Standard by Motorola, Retrieved December 1, 2010 from: http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus
10. PIC16F877A Data Sheet (2004). Microchip Technology Inc., Retrieved December 5, 2010 from: <http://www.microchip.com/>
11. Delport, Vivien (2011). “Low-Power Wireless Sensor Networks”. TechZone Sensor Solutions Magazine. Pages: 52-55, TZS111.US, Spring.
12. RTC, Serial Alarm Real Time Clock, DS1305 (2009). Retrieved July 19, 2011, from

- <http://www.maxim-ic.com/datasheet/index.mvp/id/2686>
13. 25LC256 Serial SPI EEPROM Data Sheet (2007). Retrieved November 25, 2011, from <http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en023856>
 14. LM-34 Precision Fahrenheit Temperature Sensor (2000). Retrieved December 29, 2011, from <http://www.national.com/mpf/LM/LM34.html#Overview>
 15. HM1500LF Relative Humidity Module (2008). Retrieved September 10, 2011, from <http://www.meas-spec.com/downloads/HM1500LF.pdf>
 16. MS5801-01BA Altimeter module (2011). Retrieved November 20, 2011, from <http://www.servoflo.com/pressure-sensors/item/ms5801-01ba-altimeter.html>