Real-Time Wireless Network of Mobile Sensor Nodes Based on ZigBee Protocol: Design and Implementation

Abdulbasit M. Sabaawi[a*], Abdullah A. Mohamed[b]

[a]*Communications and Signal Processing Department, Newcastle University, UK*
[b]*Communication Department, Collages of Electronics Engineering, University of Mosul, Iraq*
[a]*Email: abd_albaset87@yahoo.com*
[b]*Email: Abdullah.4Mohamed@Gmail.com*

Abstract

Wireless sensor networks are an evolving technology for a wide range of environments due to its low-cost and importance that has been implemented by the recent delivery of the IEEE 802.15.4 standard (ZigBee standard) for application layers. The benefit of this module is to develop the designing skills on the wireless networks using ZigBee protocol that is providing a standardized base set of solutions for control systems and sensor .In this paper, a wireless network consisting of four mobile sensor nodes is designed and implemented. Thermal sensor is used in each node to measure the temperature. After the temperature was measured correctly, the system is modified in terms of having a better reliability by implementing the CRC technique. Furthermore, a TDMA and CSMA algorithms are applied to the nodes in a way that the four nodes of the network must be able to transmit and receive the data without any collision. The system is applied on a modern embedded board is PICDEM-Z BOARD. This board has excellent features such as high performance core PLCI8F4620, memory and Rf transceiver work with ZigBee protocol. The results show high flexibility and reliability in the measuring and exchanging the data between all the nodes within one second in real time.

Keywords: Wireless network sensor; Zigbee; IEEE 802.15.4 standard; Microchip PICDEM-Z.

1. Introduction

In recent years, many applications such as target tracking and remote environmental monitoring depend on a wireless sensor network.

----

* Corresponding author.

Because of the sensors that are intelligent, cheaper and smaller, they are equipped with wireless interfaces [1]. Hence, they can form a network and communicate with each other. In order to implement the sensor network, PICDEM-Z BOARD has been used.

1.1. PICDEM-ZBOARD

The microchip provided for the implementation is PICDEM-Z board. The board contains a micro-controller PLCI8F4620 which performs various functions as shown in figure 1. This 40-pin micro-controller has various kinds of specific features such as it is coded in c language and it provides EEPROM. The RF transceiver used in this board is MRF24J40 transceiver. MRF24J40 is a complete IEEE 802.15.4 radio and it operates in the frequency band of 2.4GHZ.It uses a Zigbee protocol to provide a solution for the wireless sensor networks. It is typically a modem that is used to transmit M2M data. The ICD-2 port in the microchip is used to burn the program from the computer to the circuit using ICD-2 microchip. The RS 232-serial port in the board helps to decode the data from the microchip to the computer through the RS 232 serial wire. HyperTerminal is used in the software part to get the data which is decoded from the circuit through the RS-232 Serial port [2].
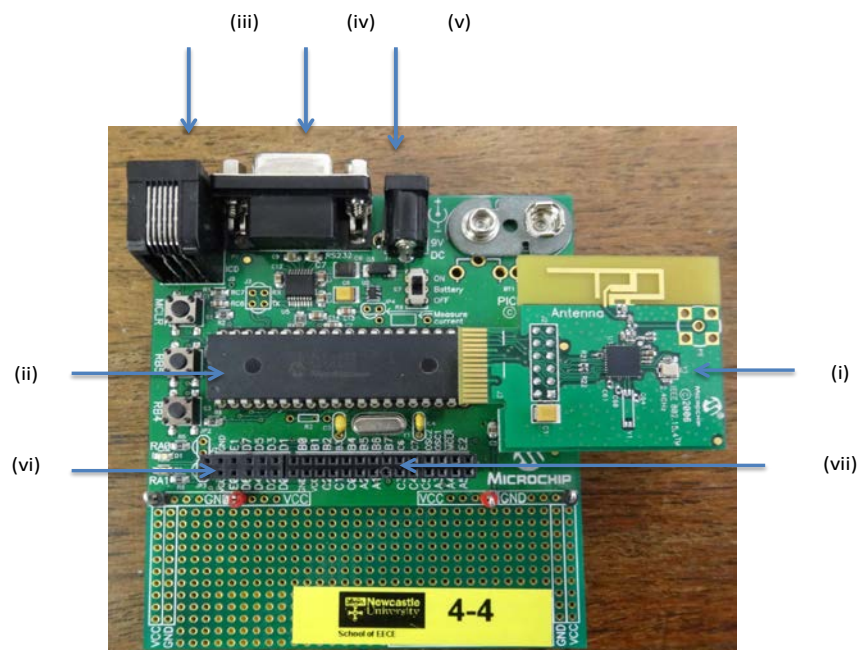

Figure 1: Microchip PICDEM-Z

(i)	MRF24J40 transceiver	(v) Power source port
(ii)	PIC18F4620	(vi) Connector J7
(iii)	ICD-2 port	(vii) Connector J6
(iv)	RS232 serial port	

1.2. ZigBee protocol

ZigBee plays a vital role in the implementation. It provides a high level communication protocols that is used to create the personal area network. Zigbee is based upon IEEE 802.15 standard. It is used in the applications that require a low-data rate, long-battery life and secure networking. It has a fixed rate of 250 Kbits/sec which is applicable for the periodic or intermediated data or a single signal transmission from a sensor or input device [3]. The channels in the ZigBee ranges from 11to 26, each have a separate center frequency as shown in figure 2. The ZigBee channels are designed in the 2.4GHz Industrial, Scientific, medical (ISM) radio bands (portion of radio spectrum) as shown in table 1.
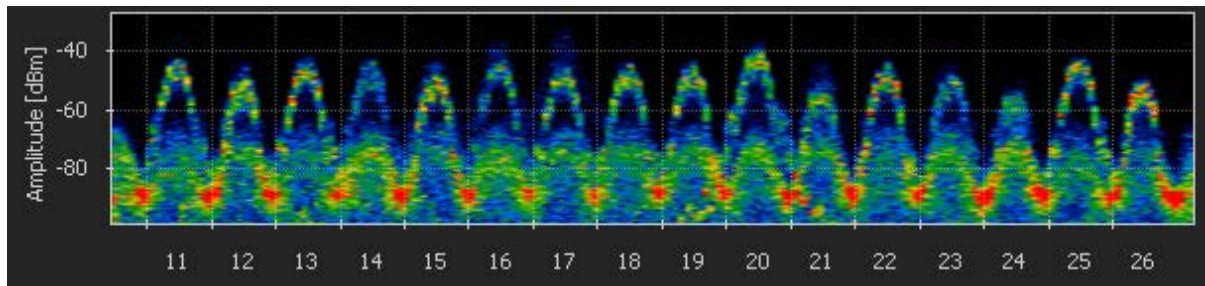

Figure 2: Topographic view of Zigbee Channel [4]

Table 1: ZigBee channels in the 2.4GHz ISM Band [4]

ZigBee Channel	Center Frequency (MHz)	Control4
11	2405	0
12	2410	1
13	2415	2
14	2420	3
15	2425	4
16	2430	5
17	2435	6
18	2440	7
19	2445	8
20	2450	9
21	2455	10
22	2460	11
23	2465	12
24	2470	13
25	2475	14

1.3. Objectives

- A protocol is to be designed such that a minimum of four nodes within the same network can maintain communication without any collisions.

- A peer-to-peer network should be established with full coverage, such that each node can receive an update at least once from all corresponding nodes per second.

- The communication channel will be operating on a single frequency.

- The probability of an incorrect reading must be < 1/1000. Hence an error correction technique is to be implemented to ensure high data integrity.

- Each node must be able to display all the readings from the respective nodes within the network via a serial I/O port in a format (text) that is easy to interpret.

2. System Design

In this design, it has been assigned the task of designing, testing and implementing a network of mobile sensors using four nodes. As shown in figure 3, each node consists of a microprocessor, wireless transceiver, I/O port and a number of sensors all linked together.
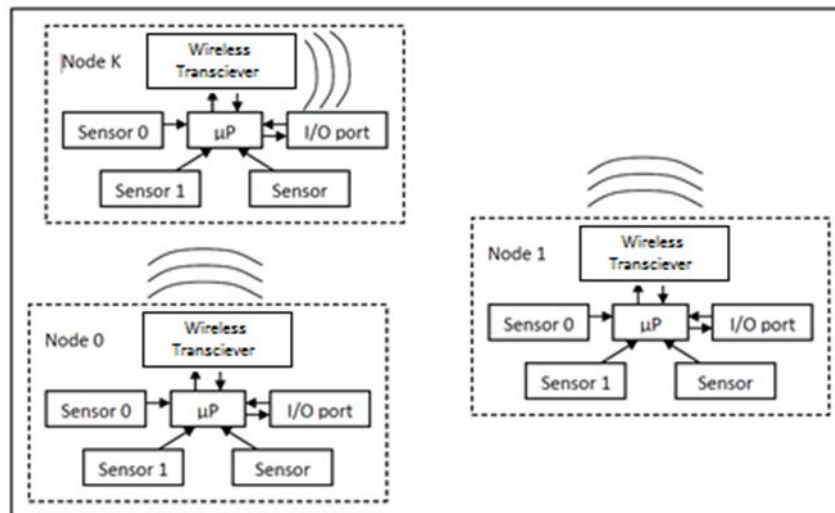

Figure 3: Architecture of Wireless Sensor Network [5]

Based on the objectives and the background theory, various designs were made in the project. These designs were in phases. A phase was a design based on achieving a certain overall operation. A list of these phases is stated below:

1. **Initial Communication Phase**: - This is an initial phase with the intention of gaining familiarization with the communication aspect of the overall design. It should be stressed that this is not the actual final design on which the sensors communication system will be based on. However certain elements on the final design will be based on this and the understanding of how the final communication system will work shall be derived from this.

2. **Temperature Phase**: - This phase should cover how the sensor element of the design should work in terms of taking measurements and being able to display them accurately. This will be done with respect

to the Temperature entity.

3. **Error detection Phase**: - In this phase, an error detection technique was implemented to ensure that accurate and error-free readings were obtained at the nodes while rejecting those with errors. This was done to minimize inaccurate readings as a result of errors picked up during transmission. An example of this detection is using Cyclic Redundancy Check(CRC).

4. **Node network traffic Phase**: - This phase dealt with the coordination of data traffic between all 4 nodes to maintain fast and consistent data traffic while preventing collisions. This can be implemented by using Time Division Multiple Access (TDMA) and Collision Sense Multiple Access (CSMA).

The system design can be divided into two parts, namely the hardware design and Implementation and the software design.

2.1. Hardware Design and Implementation

2.1.1 The Microchip PICDEM-Z board

The Microchip PICDEM-Z board is used in this work as shown in figure 1. This is the main hardware that houses the microprocessors, wireless transceiver and various input and output ports. In future, this board will have a sensor attached to it. In this project, 4 such boards were used. They each had an ID tag. Their tag names were 4-1, 4-2, 4-3 and 4-4 respectively.

In this board, a 3.3V power supply was connected to the 4-4 board via the power source port (see (V) on figure 1) and the modified software from the software phase was burnt into the board via the ICD-2 port (see (iii) on figure 1). This procedure was repeated with board 4-1. This board is designed to use the ZigBee protocol and will communicate with other boards over the ISM frequency band and based on the software design, channel 14 was used and with respect to Table 1 , the boards are operating at 2.42 GHz.

While board 4-3 and 4-2 were still active, the serial port (see (iv) on figure 1) on board 4-3 was connected to the PC and the HyperTerminal software on the PC was used to observe the received data. This was repeated with a connection to board 4-2.

2.1.2. Temperature phase Design

The diagram in figure 4, illustrates the design of the temperature sensor circuit. Note that the 3.3V supply is based on the power supply of the Microchip PICDEM-Z board as indicated in the data sheet [6]. However the actual value may range from 3.1V to 3.3 V in practice, so the actual supply values must be read from each board and taken into account for the design. As the intention, is to supply the temperature sensor via the board when they are connected and operating together. Finally it is to be noted that the output of the sensor, Vo, is going to be an input onto the board as shown in figure1.
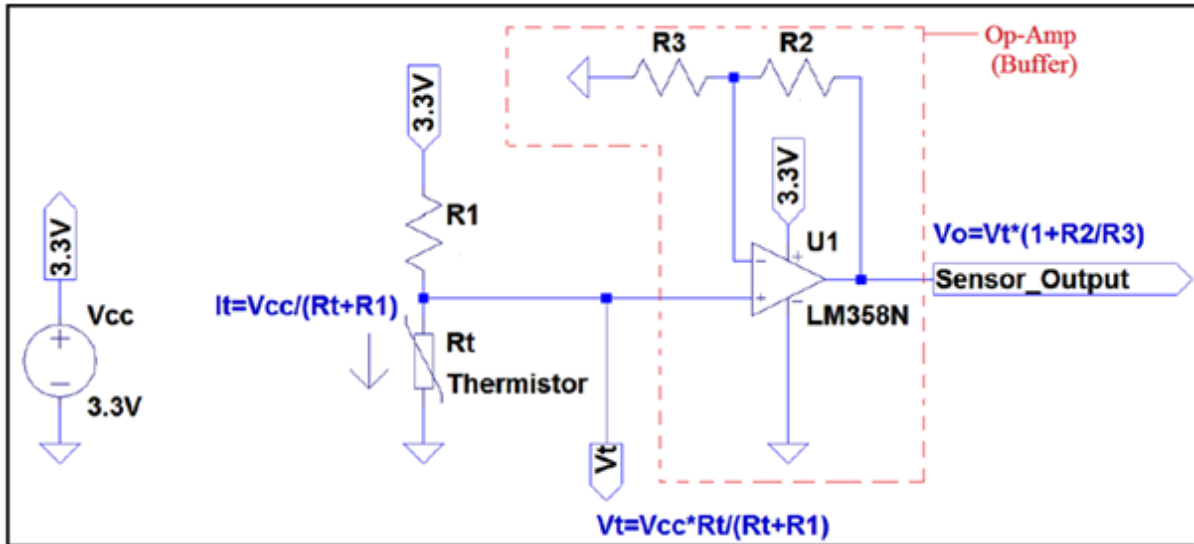
Figure 4: Circuit Diagram of the temperature sensor circuit

Based on the sensor circuit design, the sensor output (Vo) should be ideally and theoretically within the range of 1.65V at room temperature (~22℃)which is the midpoint between the maximum and minimum operating voltage (3.3V and ≈0V) of the board [6][7], while allowing an offset of ±0.3V from this value (1.65V) in practice. This is desired, so that any fluctuation in the voltage as a result of a temperature change (high or low), should be well within the operating voltage range. It is also desired as any voltage increase or decrease from this value (1.65V) depending on the temperature, should never encounter a scenario where it reaches the voltage maximum or minimum operating limits as this is bad practice and any readings going beyond this range, will give undesirable and inaccurate results. The design should ideally be able to at least read a temperature range from 5℃ to 40 ℃

The sensor should not be influenced by the ADC in the board in order to function properly, because the thermistor can get loaded by the current in the ADC, which causes the thermistor to read wrong values. So an Op-amp was used as a buffer, to isolate the sensor from the ADC (as displayed in figure 4).The Op-amp gain is arbitrary and this attribute must be exploited to maximize the dynamic range of the ADC.

Based on the design explained, the following specifications were recorded and calculated:-

Vcc Board (4-1) =3.18V   Vcc Board (4-2) =3.2V   Vcc Board (4-3) =3.18V

Vcc Board (4-4) =3.19V

Rt at room temperature =6.58 K Ohms

R1= 15K Ohms   R2=R3= 10K Ohms

Vt at room temperature=0.835V
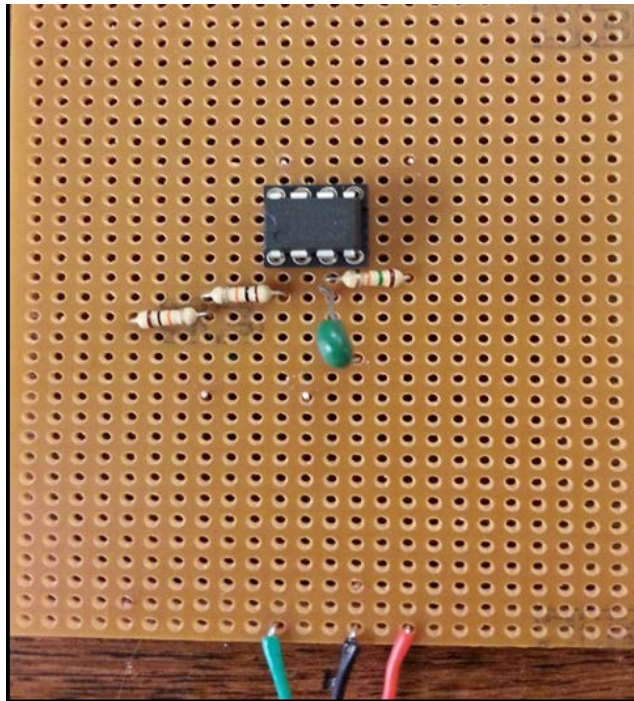
273

Vo at room temperature=1.67V

Figure 5: The hardware temperature sensor circuit

2.2. Software Design

1. The platform used to build the software is in embedded C language.
2. The code was to be modified, compiled and burned into the board via MPLAB X v1.85 software.
3. The instructions in the code were to make the transceiver send information in packets.
4. Each packet could contain a numerical value ranging from 255 (maximum packet size) to 0 (minimum packet size).
5. The packets were sent in a chronological order with respect to the packet size, ranging from the maximum to the minimum. Hence the information was counting down from (255-0)
6. The instruction also ensured that the packets were received by transceiver with the use of a channel.
7. A specific channel was used. And this was defined in the code by its Channel ID. This was to be the only channel used during the course of this project. But if another source was using the same Channel, a different Channel (Channel ID) may be used temporarily to briefly observe results. This is done as collisions will occur if another source was transmitting via the same channel. Thus giving no/undesirable results.
8. The channel ID used was the fourth ZigBee channel within the ISM band. Which according to table 1 is channel 14.
9. A specific Packet ID was also defined in the code. This was done to confirm that the data being received, it is from a source in this project. Hence identifying the information source and confirming that successful transmission has occurred.
10. The HyperTerminal start up settings applied to display the final result of the project.

2.2.2 Voltage to temperature conversion

Once the analogue input had been converted to digital information, the data needed to be scaled so that all the data coming in would be displayed in ºC. The thermistor used for the sensor was TCC 502 61-0410..A slope equation was applied to a small range from it to ensure a liner response, in order to determine the relationship between the resistance (Rt) and the actual temperature value as follows;

By applying the line equation formula:

$$\frac{y - y_1}{x - x_1} = \frac{y_2 - y_1}{x_2 - x_1} \Rightarrow \frac{R_t - 10k}{T - 5} = \frac{3k - 10k}{35 - 30} \Rightarrow 30(R_t - 10k) = -7k(T - 5)$$

$$R_t - 10k = -233.33(T - 5) \Rightarrow -233.33 \, T = R_t - 10000 - 1166.667$$

$$\therefore T = \frac{R_t - 1166.667}{-233.33}$$

After applying this equation, the temperature reading of the sensor was between 20 0C to 32 0C, and in order to obtain wider range of temperature readings, the range was expanded, and that is why the slope was modified until the correct temperature was obtained. The final equation after the slope modification was written as follows:

To conclude the software design of the temperature phase, code was written to map the input of the sensor (Vo) into the microcontroller, after which the ADC conversion process was instructed. Once this was completed, the digital data was then scaled to display accurate readings of temperature. In addition, the output of the transmitter was instructed via coding to transmit the data readings while the packet size countdown transmission was omitted.

2.2.3 Error detection phase

The error detection technique being used is Cyclic Redundancy Check, also commonly known as CRC. In this technique, the data (each packet being transmitted) is divided by a fixed 16-bit polynomial and the remainder is appended to the data (packet) and transmitted. Upon receiving this packet, the receiver already knows the 16-bit polynomial that was used at the receiver and repeats the division process. If the resulting remainder matches that appended into the packet via the transmitter, then the message is believed to be error free. But if there is no match, it means that there is an error.
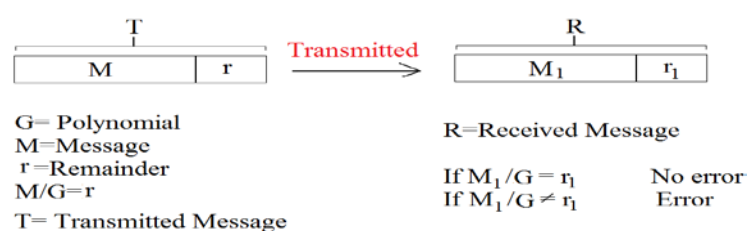

Figure 9: Simple Illustration of CRC operation

2.2.4 TDMA and CSMA implementation

In this work, token passing technique is used in order to implement the TDMA method as shown in figure 10. There are some specifications in this technique. First, the network must communicate together without any collision of RF transmissions. Second, each node must update the data in the network at least once per second. This is achieved by dividing the maximum time slot, which is one second, equally into all nodes [8]. Hence the total time is 1 second and 4 nodes were used, so the time slot for each node is 0.25 second.
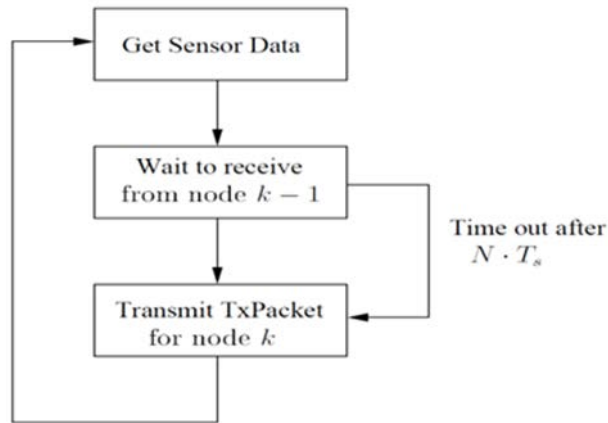

Figure 10: Token passing network algorithm

While the CSMA technique works by sensing the channel to decide whether any other node is transmitting, as illustrated in figure 11. In order to do this process the PHYGetRSSI() function was used [9]. Furthermore, all the nodes should transmit and receive the data within a period of one second. When the channel is busy, a random delay time was provided by using the rand() function in C language.
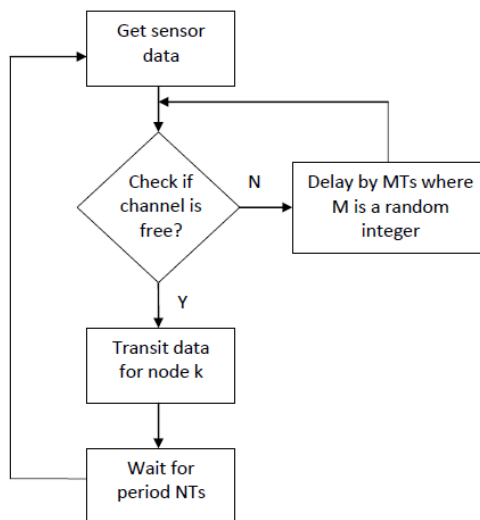

Figure 11: CSMA technique

3. Results and Conclusions

3.1. The Temperature circuit

The first part of the design was measuring the temperature, and to achieve that, the output voltage of the ADC should be obtained, then the value of the resistor should be calculated as well. Finally line equation can be applied to find the relation between the temperature and the resistor in order to obtain the temperature value. This part was accomplished and the temperature was measured as shown in figure 12.
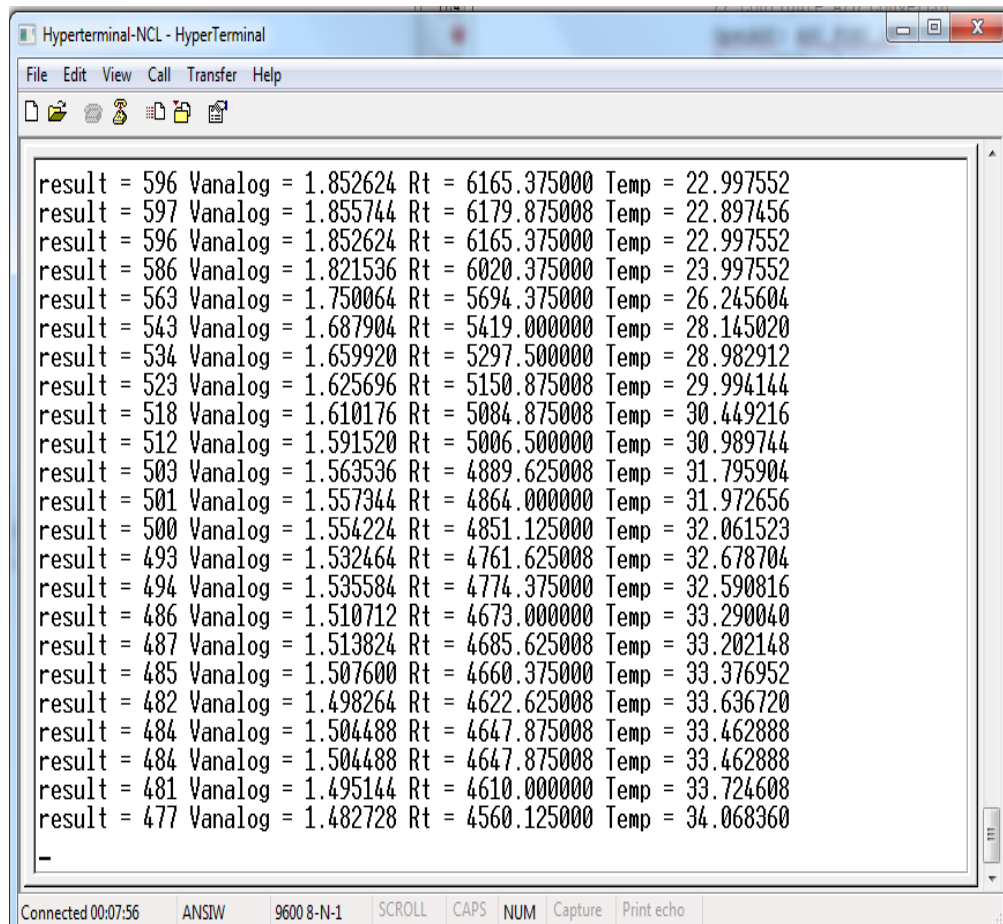


```
Hyperterminal-NCL - HyperTerminal
File  Edit  View  Call  Transfer  Help

result = 596 Vanalog = 1.852624 Rt = 6165.375000 Temp = 22.997552
result = 597 Vanalog = 1.855744 Rt = 6179.875008 Temp = 22.897456
result = 596 Vanalog = 1.852624 Rt = 6165.375000 Temp = 22.997552
result = 586 Vanalog = 1.821536 Rt = 6020.375000 Temp = 23.997552
result = 563 Vanalog = 1.750064 Rt = 5694.375000 Temp = 26.245604
result = 543 Vanalog = 1.687904 Rt = 5419.000000 Temp = 28.145020
result = 534 Vanalog = 1.659920 Rt = 5297.500000 Temp = 28.982912
result = 523 Vanalog = 1.625696 Rt = 5150.875008 Temp = 29.994144
result = 518 Vanalog = 1.610176 Rt = 5084.875008 Temp = 30.449216
result = 512 Vanalog = 1.591520 Rt = 5006.500000 Temp = 30.989744
result = 503 Vanalog = 1.563536 Rt = 4889.625008 Temp = 31.795904
result = 501 Vanalog = 1.557344 Rt = 4864.000000 Temp = 31.972656
result = 500 Vanalog = 1.554224 Rt = 4851.125000 Temp = 32.061523
result = 493 Vanalog = 1.532464 Rt = 4761.625008 Temp = 32.678704
result = 494 Vanalog = 1.535584 Rt = 4774.375000 Temp = 32.590816
result = 486 Vanalog = 1.510712 Rt = 4673.000000 Temp = 33.290040
result = 487 Vanalog = 1.513824 Rt = 4685.625008 Temp = 33.202148
result = 485 Vanalog = 1.507600 Rt = 4660.375000 Temp = 33.376952
result = 482 Vanalog = 1.498264 Rt = 4622.625008 Temp = 33.636720
result = 484 Vanalog = 1.504488 Rt = 4647.875008 Temp = 33.462888
result = 484 Vanalog = 1.504488 Rt = 4647.875008 Temp = 33.462888
result = 481 Vanalog = 1.495144 Rt = 4610.000000 Temp = 33.724608
result = 477 Vanalog = 1.482728 Rt = 4560.125000 Temp = 34.068360

Connected 00:07:56    ANSIW    9600 8-N-1    SCROLL  CAPS  NUM  Capture  Print echo
```

Figure12: the measured temperature

As the figure 12 above shows, the temperature was measured for a wide range of degrees. Also it can be noticed that when the temperature increased the Rt value will be decreased and that verifies the concept of the negative slope which the line equation was derived from.

3.2. The CRC Algorithm

After measuring the temperature, the CRC algorithm was applied in order to improve the reliability of the designed system. The CRC algorithm was implemented in the transmitter and the receiver to make sure that the received data is correct as shown in figure 13.
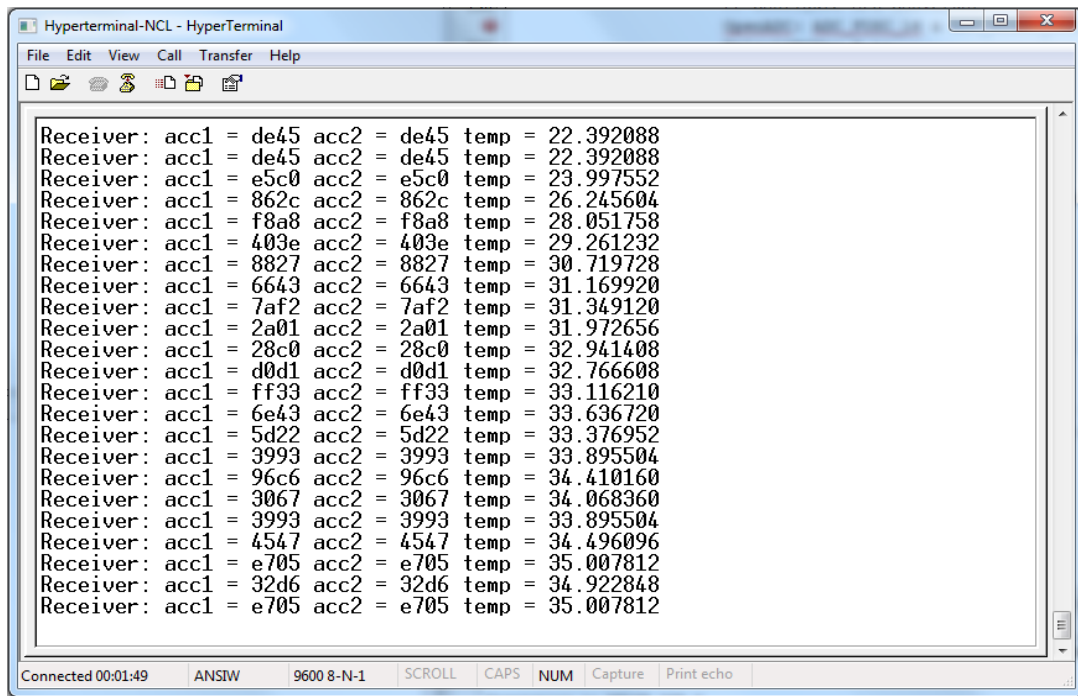
Figure 13: The temperature with the CRC algorithm

As the figure (4.2) shows; the CRC algorithm was applied to the transmitter and the receiver. The value of acc1 represents the calculated CRC in the transmitter, while the acc2 represents the value of the CRC in the receiver. It can be seen that the CRC values in the transmitter and the receiver are equal which means the data have been received without any errors, and if they were not equal, then it means an error have been occurred during the process.

3.3. The TDMA

The TDMA (Time Division Multiple Access) was the third main part in the design; this approach will divide the network into time slices. In this project, since four nodes were used, then the time specified for each node will be 0.25 second, the four nodes should transmit within one second without any collision, this was accomplished and the four nodes successfully transmitted and received the updated data each second without any problems as shown in figure 14.
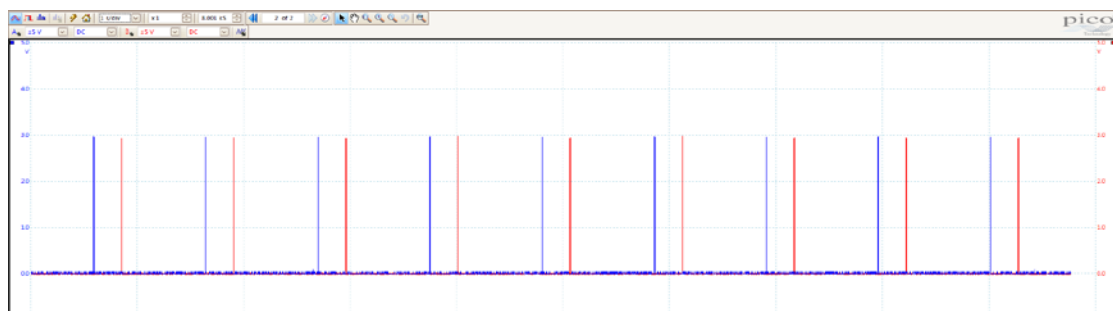

Figure 14: the TDMA for two successive nodes

As the figure 14 shows, in this case only two nodes were connected to the pico scope channels, and the time slot between each two successive nodes is 0.25 second.
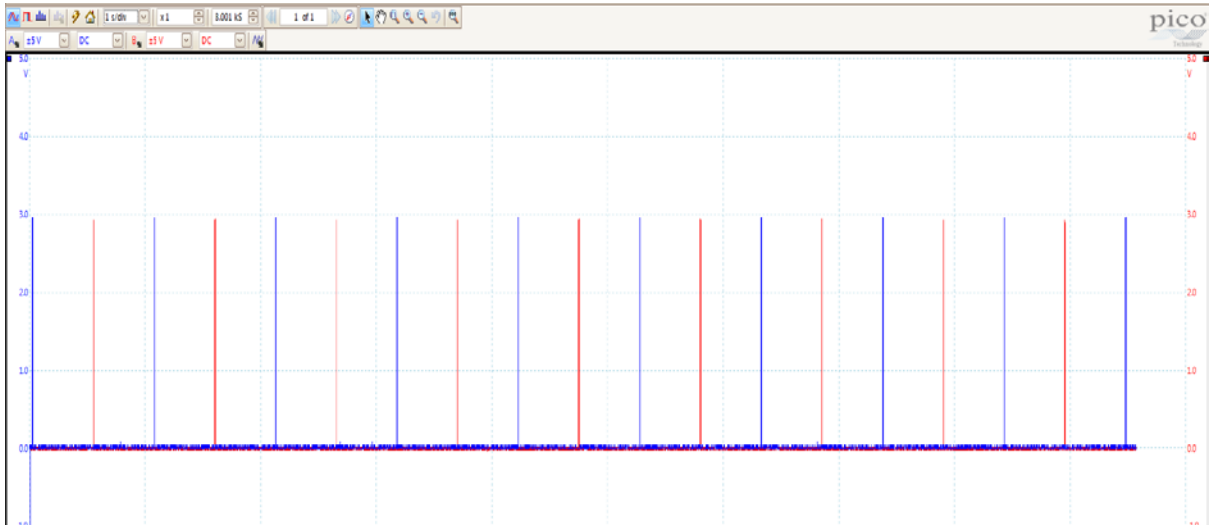

Figure 15: TDMA for nodes (1 and 3)

The figure 15 above shows the data for nodes (1 and 3), as we can see, the time slot between them is 0.5 second, and the same graph was observed between the nodes (2 and 4).
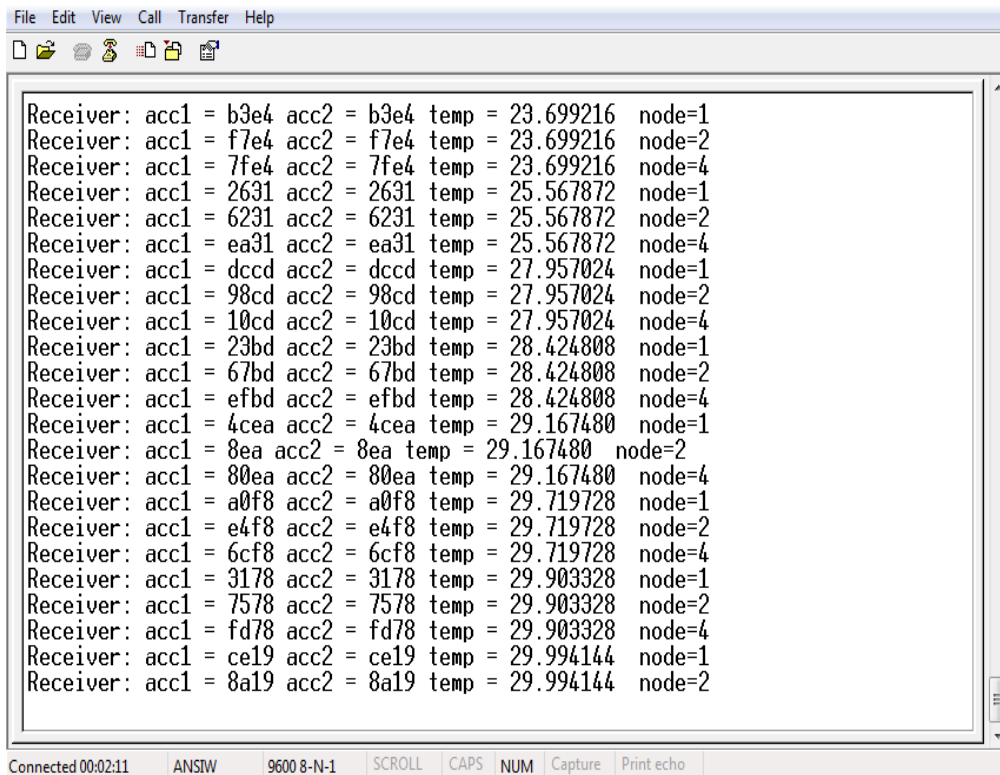

Figure 16: TDMA with CRC

As it can be observed from the figure 16, the four nodes were communicating together correctly, and the data

279

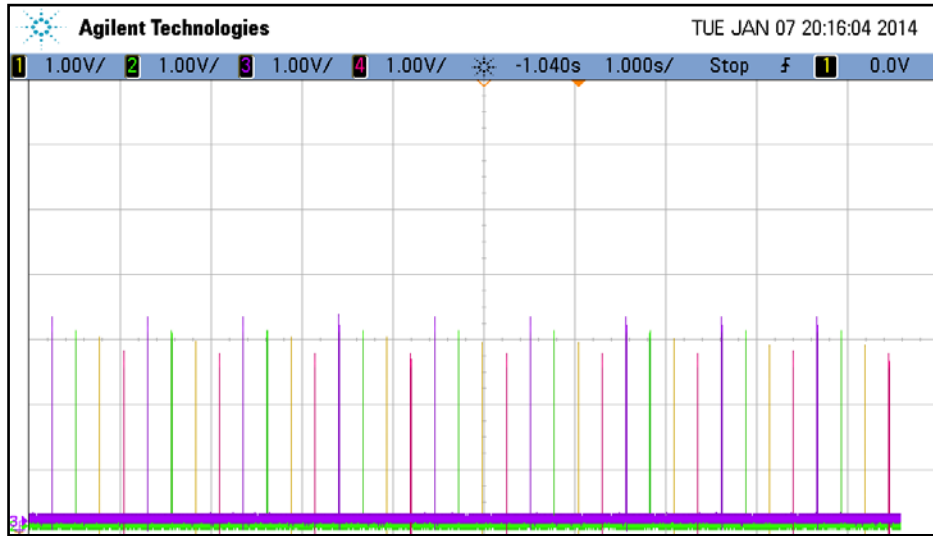was updated in each node every second.

Figure 17: TDMA after connecting the four nodes

As the figure 17 above shows; the four nodes were transmitting the data each second and the time slot between each two nodes is 0.25 second without any collision.

3.4. The CSMA

In this algorithm the channel was checked before transmitting the data, if the channel was Idle; which means that there is no data transmission in that moment, then the data will be transmitted, otherwise, the transmitter will wait for a random period of time before checking the channel again, if it was Idle then the data will be transmitted, otherwise the same routine process will be repeated again until the channel will be free. This approach was achieved and the four nodes were transmitting and receiving without any collision as shown in figure 18.
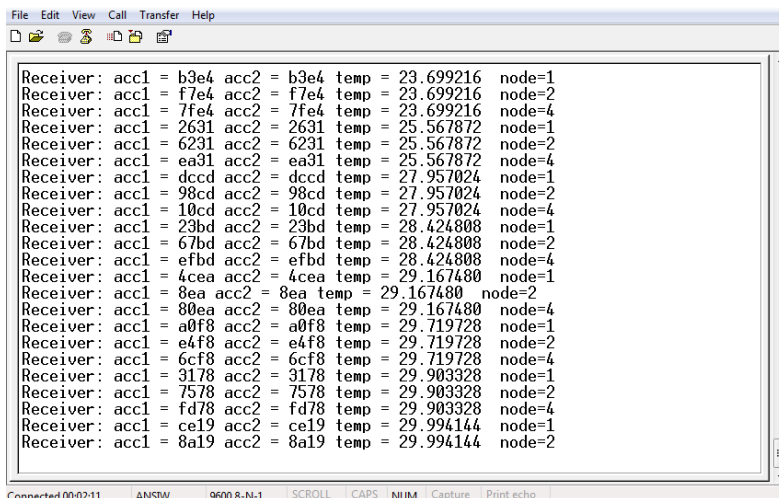

Figure 18: CSMA algorithm

As the figure 18 above shows, the four nodes were transmitting the data but in different order unlike the TDMA algorithm. After connecting the four nodes to the oscilloscope the figure 19 was obtained.
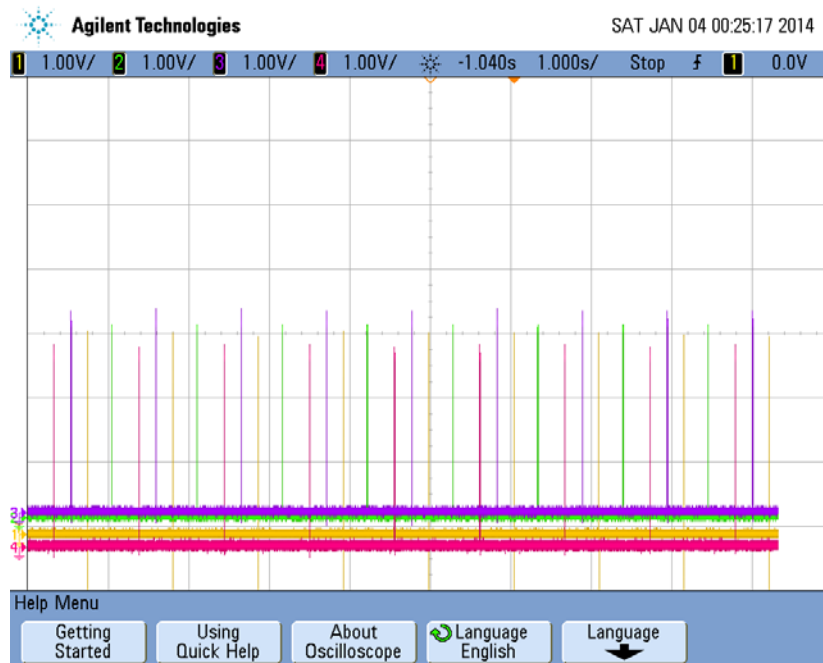

Figure 19: The CSMA after connecting the four nodes

The time slot between each two nodes was not 0.25 sec and that matched the concept of CSMA algorithm, however; each node will transmit the data within a period of one second..

4. Conclusion

It is known that the PICDEM-Z board with Zigbee protocol is suitable to achieve wireless sensor network and they achieve good result**.** In this paper, a wireless network of four nodes was implemented. These nodes communicate with each other using the 2.4 GHz frequency channel. MPLAB X v1.85 software program was used in this module to program the four nodes by writing a code in C-language. The first task of this paper was to design a circuit for measuring the temperature by using a thermal sensor. After a successful reading of the temperature, the system was modified in order to improve the reliability of the network data; this was achieved by applying the CRC technique.

The next stage of the design was to implement the TDMA technique. The TDMA is very important in order to ensure that the four nodes of the network are communicating with each other without any collision, and the four nodes should transmit and receive the data within one second, which means that a 0.25 second time slot should separate the transmission of any successive nodes. In addition, CSMA algorithm was implemented. In this algorithm the four nodes also should update the transmitted and received data during a time period of one second, but the time slot between any two successive transmissions is not restricted to be 0.25 second .

References

[1] M. T. Lazarescu, "Design of a WSN Platform for Long-Term Environmental Monitoring for IoT Applications," in IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 3, no. 1, pp. 45-54, March 2013.

[2] "PIC18F2525/2620/4525/4620 Data Sheet." Microchip. N.p., n.d. Web. 3 Dec. 2013.

[3] "ZigBee Specification FAQ." ZigBee Specification FAQ. N.p., n.d. Web. 6 Dec. 2013. <http://www.zigbee.org/Specifications/ZigBe

[4] T. de Almeida Oliveira and E. P. Godoy, "ZigBee Wireless Dynamic Sensor Networks: Feasibility Analysis and Implementation Guide," in IEEE Sensors Journal, vol. 16, no. 11, pp. 4614-4621, June1, 2016.<http://www.metageek.net/control4/zigbee-channels/>.

[5] H. Ghayvat, J. Liu, S. C. Mukhopadhyay and X. Gui, "Wellness Sensor Networks: A Proposal and Implementation for Smart Home for Assisted Living," in IEEE Sensors Journal, vol. 15, no. 12, pp. 7341-7348, Dec. 2015.

[6] "PICDEM™ Z Demonstration Kit User's Guide." Microchip. N.p., n.d. Web. 13 June 2012. <http://ww1.microchip.com/downloads/en/DeviceDoc/51524c.pdf>.

[7] "Thermal Sensor with SPI Interface." Microchip. N.p., n.d. Web. 6 Dec. 2013. <http://www.microchip.com/downloads/en>

[8] "TDMA & CDMA Technologies." TDMA and CDMA Technolgies. N.p., n.d. Web. 7 Dec. 2013. <http://www.tutorialspoint.com/gsm/tdma_

[9] "EEE8092 MSc Lab 9 to 12." EEE8092 MSc Lab 9 to 12. N.p., n.d. Web. 13 Dec. 2013. <http://www.staff.ncl.ac.uk/charalampos.tsimenidis/EEE8092/EEE8092-L4.html>.