Journal of Engineering Science and Technology
Vol. 17, No. 5 (2022) 3619 - 3636

DESIGNING AN EMBEDDED WIRELESS SENSOR NETWORK PLATFORM USING SINGLE BOARD COMPUTER APPLICATION

SAJA ZAID HASAN[1], SAJJAD HASAN SHARQI[2],
TAYSER SUMER GAAZ[3,*], AHMED A. AL-AMIERY[4]

[1]Information Technology Center, Ministry of Electricity, Baghdad, Iraq
[2]Planning and Studies Office, Ministry of Electricity, Baghdad, Iraq
[3]Technical College Al-Musaib, Al-Furat Al-Awsat Technical University, Babil, Iraq
[4]Energy and Renewable Energies Technology Center, University of Technology, Baghdad, Iraq
*Corresponding Author: taysersumer@atu.edu.iq

Abstract

The aim of the recent research on wireless sensor network (WSN) technology is provide solutions to many applications. This manuscript concerns designing a wireless sensor network system that facilitates open-source hardware platforms, ZigBee and Raspberry Pi. The embedded system architecture combines the gateway node of the wireless sensor network, database server, and web server in one single-board computer (SBC) hardware platform. The design reduces power consumption and cost by developing better scalability and lesser sensors units to improve sending and receiving data. The use of Raspberry Pi in the design enhances flexibility and reduces the cost. Further, using the Python code in programming minimizes the codes of the components and avoids errors. The choice of the Raspbian operating system for Raspberries has resulted in higher quality, better stability, and a smoother scalable operating system. Raspberry Pi acts as a base station for the ZigBee protocol. Additionally, data can be transferred remotely at the base station via the website. The advantage of using python in the programming makes the work on the Raspberry easier and faster, and the language learning becomes easier than before. The other advantage of this technique is to allow sending the data to any Raspberry by clicking special buttons on the GUI, capability to work with both multicast- and unicast-based communication schemes, and to work as router and end tag(E52) to enable sending the real-time data to the nearest router. Analytically, the new design provides a maximum transmit power of 20 dBm12 compared to 18 dBs and increase the output power increases by a little more than 2 dB.

Keywords: Embedded, Design, SBC, ZigBee.

1. Introduction

In recent years, there has been development in the design and production of numerous embedded systems for various applications such as remote monitoring, alerting, controlling, and many other functions, which led to good works and tasks [1]. The embedded systems are designed in both hardware and software directions in one component to perform a particular task comparably with general computers. Wireless sensor networks (WSN) have critical applications such as environment monitoring and tracking [2]. These systems are composed of sensors that are equipped with wireless interfaces to communicate with each other. However, a wireless sensor network based on Single Board Computers (SBC) is a cost-effective, low-power, well utilized, and efficient. Such systems can keep track of several occurrences by taking in many different inputs in real-time [3].

Computers composed of different units with variant functions are known as SBC. In the past and present, SBC embedded systems have used as an essential, multi-purpose processing module that can function as an independent unit since the boards are equipped with Ethernet, Secured Digital (SD) card, and a USB/Serial converter, which allow preliminary and additional development to be done without influencing the main part of the module [4, 5]. New paragraph: use this style for beginning a new paragraph.

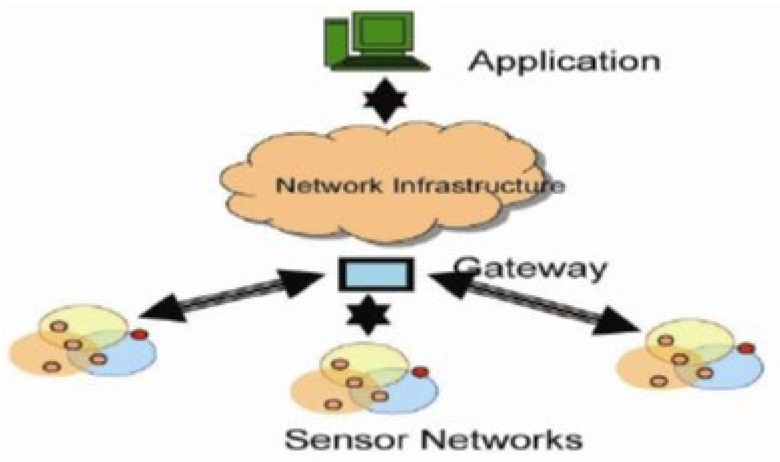
A Raspberry Pi is a type of SBC with a small size of a small credit card. The Raspberry Pi has several generations, the first is known as Raspberry Pi 1, and the second generation is known as the Raspberry Pi 2. Currently, the research is on the third Raspberry Pi 3 and Raspberry Pi 4 [6]. The latest generation was developed to work on an 8 GB variant, but it took longer than expected as an LPDDR4 RAM package with 8GB and on a 64-bit version had to be specifically designed for the Raspberry Pi [7].

In addition to Raspberry, ZigBee (developed using IEEE 802.15.4 wireless standard) is another open standard protocol that allows true operability between systems. ZigBee is a simpler system with smaller power requirements, more robust and reliable, secured, less expensive, energy efficiency with lower latency, and provides better and efficient wireless connectivity infrastructure [8].

ZigBee provides a wide range of applications in wireless personal area networks, and it has been considered as one of the most popularly deployed technologies for home automation and monitoring systems. ZigBee is a wireless technology that is commonly used to create or build small-scale and low-power consumption digital networks with a communication distance of 10-100 meters. To each better coverage, the protocol's two bottom layers are based on the IEEE 802.15.4 standard, supporting three different network topologies, namely, star, tree, and - the more robust and resilient to failures - mesh.

For creating multi-vendor interoperable products, the upper layers of the protocol stack, namely Network and Application, are described in the respective ZigBee Alliance standards, now in version 3.0. This newest version is built on the ZigBee PRO-2010 specification, which in turn is an enhanced version of the 2007 specification. Version 3.0 allows for increased interoperability (e.g., the same ZigBee 3.0 network is possible to accommodate ZigBee Home Automation, Health Care, and Light Link application profiles) as well as Internet connectivity, and thus comprises a full protocol stack by adding mesh networking and security layers along with an application framework. By using a mesh network, however, ZigBee could transfer the data over long distances. Usually, ZigBee is used for low data rates that require an extended battery life and secure networks.

A data transfer rate of 250 kbit/s has been defined for ZigBee for transmission from a connected sensor or other types of input device. The ZigBee technology intends to be less complicated and cheaper than other types of networks, for example, WPANs, Bluetooth, and Wi-Fi [9]. ZigBee aims at interoperability and easy implementation to support up to 65,000 nodes depending on the type of topologies used [10]. Figure 1 shows the wireless sensor network (a) and ZigBee technology applications (b).



(a) Wireless sensor network.

(b) ZigBee technology applications.

Fig. 1. Illustration of (a) Wireless sensor network and (b) ZigBee technology applications [8].

In the following, Table 1, a summary of the most important characteristics of three technologies of WiFi, Bluetooth and ZigBee [8].

Table 1. Summary of characteristics of WiFi, Bluetooth, and ZigBee [8].

Feature	WiFi IEEE 802.11	Bluetooth IEEE 802.15.1	ZigBee IEEE 802.15.4
Application	Wireless LAN	Cable Replacement	Control and Monitor
Frequency Bands	2.4 GHz	2.4 GHz	2.4 GHz, 868 MHz, 915 MHz
Battery Life (Days)	0.1-5	1-7	100-7,000
Nodes Per Network	30	7	65,000
Bandwidth	2-100 Mbps	1 Mbps	20-250 kbps
Range (Metres)	1-100	1-10	1-75 and more
Topology	Tree	Tree	Star, Tree, Cluster Tree, and Mesh
Standby Current	$20*10^{-3}$ amps	$200*10^{-6}$ amps	$3*10^{-6}$ amps
Memory	100 kB	100 kB	32-60 kB

The Python programming language is widely known and is one of the most popular programming languages used in computing in the scientific fields. It has proven to be a good selection for the development of algorithms and exploratory data analysis. Generally, purpose language has found application both in learning and industry [11]. This programming language is the product of many years spent in development by computer scientists. As such, there are ideas about its sureness, functional programming models, and object orientation. In comparison to other programming languages, Python has many advantages, which include being easier to write, smaller code length, fewer chances of errors occurring, suitability for use in education, not requiring initial declarations, and enabling fast development with minimal friction with the typing system of the language [12]. The purpose of this manuscript is to design and implement a solution that is capable to handle in an efficient way to reach specific goals, low consumption, and low cost in a wireless transmitter to ensure that the technology can be used with common elements such as temperature, humidity, movement, speed sensors, measures and surveillance.

2. Equipment and Devices Specifications

The embedded WSN based Raspberry is designed with ZigBee, which offers data rate and a highly reliable communication link for a specified distance, as explained in Table 1. The proposed system is composed mainly of two categories hardware and software parts. Figure 2 shows the proposed system block diagram. The project includes the connection of three raspberry pi, which can send and receive data using ZigBee and display it on GUI; therefore, the information can be monitored and controlled. Raspberry 1 and Raspberry 2 are considered the end devices. They sense the signal from their sensors and will send it to Raspberry three (coordinator), which can be read on the monitor.

3. Hardware

3.1. Raspberry Pi

A Raspberry Pi is a credit-card-sized those functions like a minicomputer. One of the most prominent advantages of using a Raspberry Pi is the simple operating system installed on it [15]. The simplest way to set up the initial configuration of a Raspberry Pi is to connect the HDMI connector to a screen and connect it to a keyboard and mouse. Also, the DVI connector alternatively could be used to connect it to a screen or monitor. SD card is essential for the operation of the Raspberry in which the operating system is installed on it, although it should be inserted in the specified slot. On the first start, Raspbian OS, which is the operating system for a Raspberry Pi will be installed by pressing the letter (*I*) on the keyboard [16]. Figure 3 shows the physical configuration of a Raspberry Pi 2.

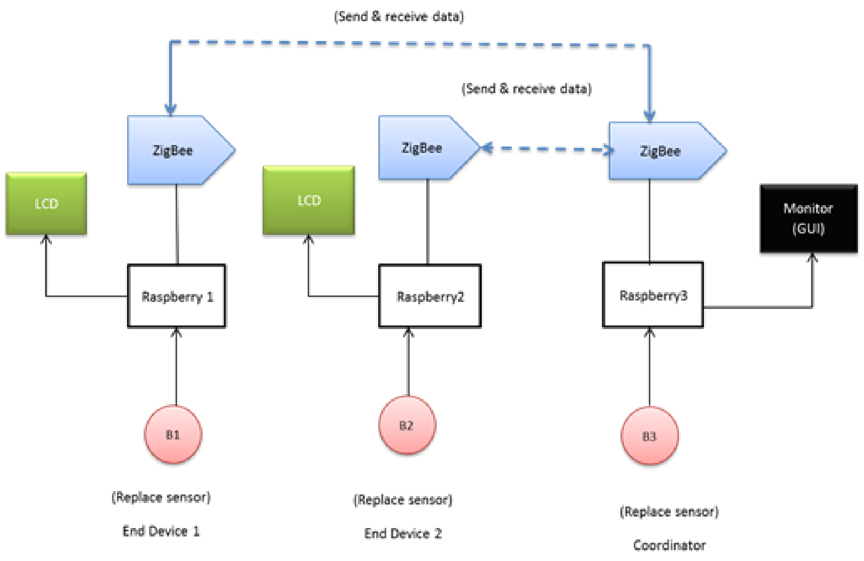


Fig. 2. The block diagram for the embedded WSN Platform [13, 14].

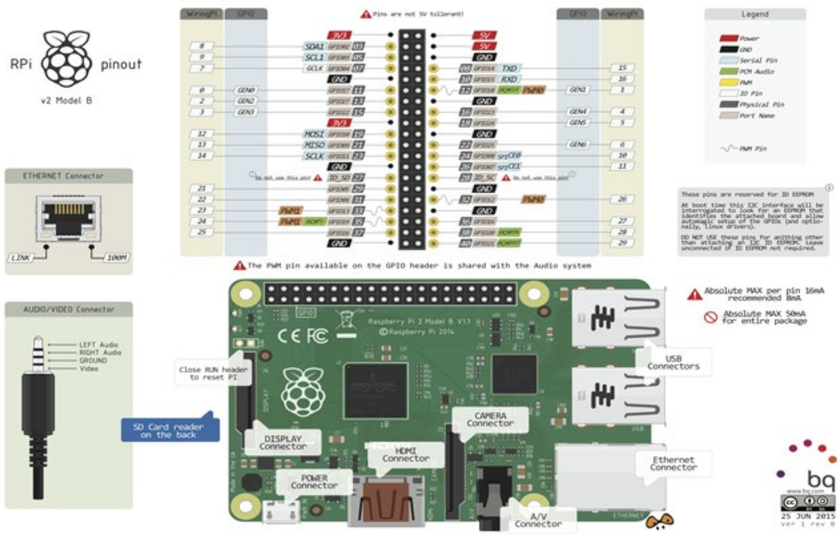


Fig. 3. Physical configuration of a Raspberry Pi 2 [16].

Raspberry Pi is equipped with a set of 40 pins called General Purpose Input/Output (GPIO), which can be used for many applications and functions as its name implies. These pins could be programmed to work as an input pin or output pin after GPIO library installation in the programs. Figure 4 displays the pin configurations of the GPIO for external connection [17]. In the pin assignment, some points are utlized for specific purposes. For example, pins 8 and 10 are used for serial communications, while other pins are used for SPI or I2C connections. SPI or I2C can still function as standard GPIO ports, however; they may also be helpful for additional sensors or outputs [18].

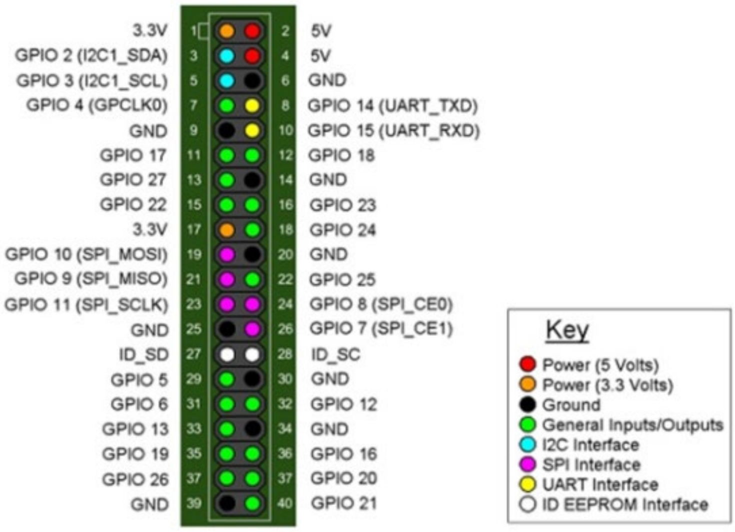


Fig. 4. GPIO pin configuration of Raspberry Pi.

The accessories include High-Definition Multimedia Interface (HDMI), which can be connected to the SBC with HDTVs and monitors. The connection to the Internet can be made using Ethernet Connector, while an A/V connector connects videos and audio.

All other connections such as camera connector, power connector (70 mA to 5 A), and display connector are using the regularly available connectors. Other devices used, Table 2, include BROADCOM BCM 2835: A 700 MHz Processor and SD Card Reader on the Back: The SD Card is the storage space or memory unit used for storing OS and for the 'Boot' task. The size could be 8 or 16 Gigabyte [16].

Table 2. Raspberry Pi comparison [19].

Raspberry Pi	RAM	Processor	USB	Ethernet	Wi-Fi	Bluetooth	HDMI	Other Video	Micro SD
Pi A+	512MB	700 MHz ARM11	1 Port	-	-	-	Yes	DSI, Composite	Yes
Pi B+	512MB	700 MHz ARM11	4 Ports	10/100Mbps	-	-	Yes	DSI, Composite	Yes
Pi 2 B	1GB	900 MHz Quad-Core ARM Cortex-A7	4 Ports	10/100Mbps	-	-	Yes	DSI, Composite	Yes

Pi 3 B	1GB	1.2 GHz, Quad-Core 64-bit ARM Cortex A53	4 Ports	10/100Mbps	802.11n	4.1	Yes	DSI, Composite	Yes
Pi 3 B+	1GB	1.4 GHz 64-bit ARM Cortex A53	4 Ports	300/Mbps/ PoE	802.11ac	4.2	Yes	DSI, Composite	Yes
Pi Zero	512MB	1 GHz single-core ARM11	1 Micro USB	-	-	-	Mini-HDMI	-	Yes
Pi Zero Wireless	512MB	1 GHz single-core ARM11	1 Micro USB	-	802.11n	4.1	Mini-HDMI	-	Yes
Pi Computer	512MB	700 MHz ARM11	1 Port	-	-	-	Yes	DSI, Composite	No
Pi Compute 3 Lite	1GB	1.2 GHz, Quad-Core 64-bit ARM Cortex A53	1 Port	-	-	-	Yes	DSI, Composite	Yes

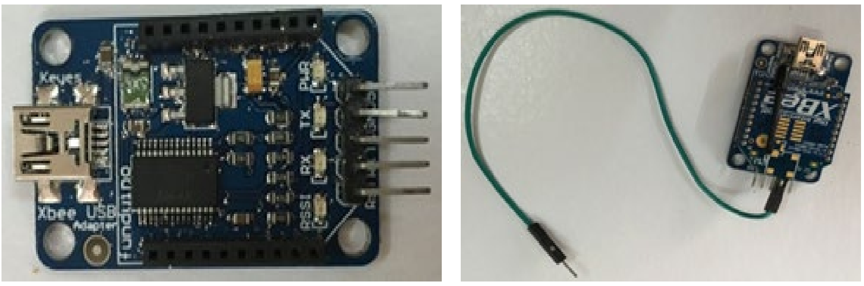
3.2. ZigBee

The short-range wireless network gateway nodes are Wi-Fi, Bluetooth, and ZigBee. The ZigBee Communication protocol is simple by design, reducing power usage and data rate compared to the other devices. Usually, this device is used as an indoor device. Therefore, the ZigBee is implemented in systems that need low data rates and very power consumption. It is necessary to avoid using a different type of ZigBee in the same network, such as using XBee pro series 1 and series 2 together. The difference lies mainly in their software configuration. The manufacturer company named XCTU software for configuration of the ZigBee. Table 3 shows the detailed specifications for the ZigBee.

Table 3. ZigBee Specifications [14].

Specification	XBee Pro (S2B)
RF Data Rate	250 Kbps
Configuration Method	API or AT
Encryption	128-bit AES
Supply Voltage	2.7 - 3.6 V dc
Transmit Current	220 mA
Receive Current	62 mA
Power-Down Current	4 uA
Receiver sensitivity	-102 dBm
Channel	16

As mentioned earlier, the configuration of ZigBee is very critical. Figure 5(a) schematically explains the programming ZigBee using a special kit called ZigBee programming kit. This kit connects the ZigBee module to any PC for testing and configurations, which is a relatively cheap and economical choice for a gateway node. The adapter has 20 ports on each side which give the availability to connect the ZigBee easily. The direction of connection of the ZigBee module for connecting the ZigBee in the kit is essential. This direction is illustrated in Fig. 5(b). The instructions and parameters for ZigBee configuration is described in the XCTU software section.



(a) (b)

Fig. 5. (a) ZigBee programming kit and (b) Connection to PC.

3.3. Liquid Crystal Display (LCD)

The LCD shows the results and data received from the raspberry or the data received by ZigBee. In this project, two raspberries are used for this reason, and the third raspberry is connected to a monitor to show the GUI to monitor the wireless sensor network. The LCD features can be outlined as follows: It has 16 Character by 2 Line Format, 4-Bit or 8-Bit Input, 5 Dots by 8 Dots Display Font, and Power Supply of 5V±10% [20]. To connect the LCD to the Raspberry board and receive data, the description of the pin is necessary; therefore, the pin description of the LCD has been shown in Table 4.

Table 4. LCD pin description [20].

No.	Symbol	Level	Function	
1	Vss	---	0 V	Power Supply
2	Vdd	---	+5 V	
3	V0	---	For LCD	
4	RS	H/L	Register Select: H:Data Input L:Instruction Input	
5	R/W	H/L	H---Read L--Write	
6	E	H/L	Enable Signal	
7	DB0	H/L	Data bus used in 8 bit transfer	
8	DB1	H/L		
9	DB2	H/L		
10	DB3	H/L		
11	DB4	H/L	Data bus for both 4 and 8 bit transfer	
12	DB5	H/L		
13	DB6	H/L		
14	DB7	H/L		
15	BLA	---	BLACKLIGHT +5V	
16	BLK	---	BLACKLIGHT 0V-	

Figure 6 shows LCD Pin Configuration for connection to a Raspberry Pi [21]. The LCD power supply is pin 1 VSS, and pin 2 VDD for ground, and +5 volt is used, respectively. However, the Pins and place defined by manufactured, therefore, including pin instructions with the component are necessary. Pin 3 acts as VE or contrast voltage which controls the contrast of the LCD screen; however, by connecting a variable resistor in which has been done in this project, the contrast could be changed. Pin 4 or register select is used to define that the data received is a data input or instruction input. Pin 5 read/write used to read from LCD or to write on it, or in other words, if showing the results is required the pin should be low or zero (write) and vice versa. Pin 6 is used to enable the LCD. Pin 7 to 14 are used to receive or read the data from Raspberry or another device. These 8 bits consider the communication pins of the device with other components. Pin 15 and 16 is to control the backlight of the device.

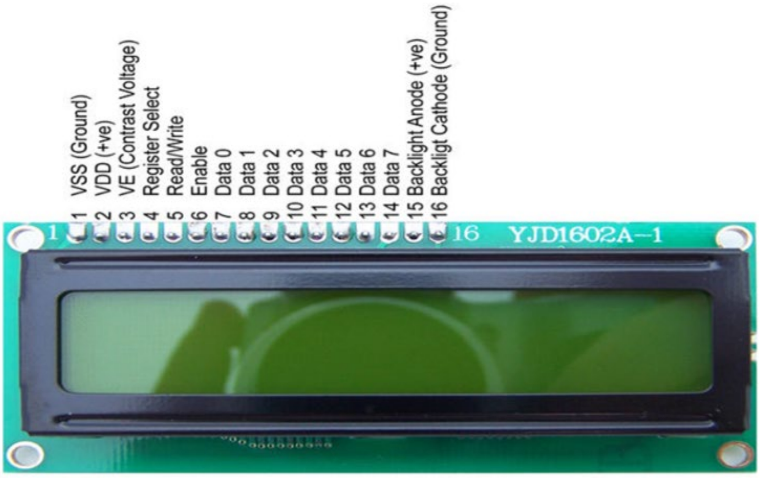


Fig. 6. LCD Pin Configuration for connection to a Raspberry Pi [21].

3.4. Sensors

WSNs are an effective way for monitoring and controlling the physical ambient surroundings both in the immediate vicinity and remotely and for use for other remote applications. The central task in design is considering the energy consumption of sensor nodes. Besides that, the deployment of sensors is an essential factor in the designing of WSNs. The power consumption is high since the network has several sensor nodes. The smart way to reduce the power usage in the WSN is activating only the necessary number of sensor nodes at any time. Therefore, the WSN will save energy. In this project, a push bottom switch was used to show the function of the network instead of other types of sensors. Nevertheless, different types of sensors could be employed to perform different functions as intended [22].

3.5. Auxiliary Devices

In addition to the leading devices in the project, various components were used which are necessary for the operation of the network. The auxiliary devices used in this work include SD Card, which is used to install operation system (Raspbian) on the Raspberry Pi. The total memory has been used is 8 GB for Raspberry Pi 1 and 2., Micro USB power port which provides 5000 mA at 5 dc volts, HDMI OUT cable (High-Definition Multimedia Interface) is used to transfer the video and audio to the monitor, and an HDMI-HDMI cable is used. Keyboards and the mouse should be standard to work with the Raspberry Pi. Although Wireless keyboard/mouse requires a single USB port [23].

4. Software

The installation of Raspbian OS was utilized in the software to be used in this design. In addition, there is software such as the SD Formatter, XCTU, Disk Image, Cool Term, and Python Language.

4.1. SD Formatter

The purpose of the SD Formatter program (Fig. 7) is to enable the software to be readable in Raspberry. SD formatter is used to format the SD Memory Card, SDHC, and SDXC. It has been recommended to use the SD Formatter program to change the format of the SD card to FAT32 instead of NTFS. However, the program optimizes the card performance because the operating system provides the SD card with lower performance. Besides that, an additional advantage of this program is that it does not format the protected zone of the storage on the SD card [15].

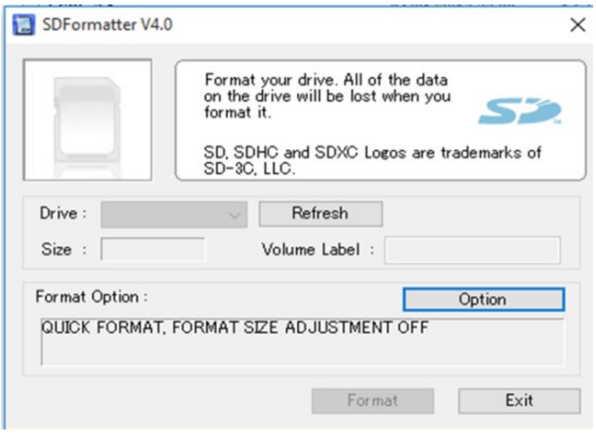


Fig. 7. SD Formatter.

4.2. XCTU

The manufacturer has assigned the XCTU program to configure the ZigBee. The characteristics of XCTU includes the Function Set, which defines the device role and select the mode of operation; PAN ID; SH and SL of 64 bits to recognize the source address; DH and DL that are send data to another ZigBee; the Power Level (PL) as defined by Tsvetanov et al. [24] and Sharma et al. [25].

4.3. Disk Image

Disk image program such other image programs is responsible for writing the files with (iso) extension to SD card; therefore, the Raspberry can read the operating system files and install it, however; this software is very useful for embedded device development or backup files and usually used with removable disks.

4.4. Cool Term

The cool term is a simple program designed to connect to the devices that work with a serial terminal such as controllers, GPS receivers, etc. In the implementation of the circuit, this program has been used to connect the ZigBee Kit to the PC.

4.5. Python Language

The python program was a successor to the ABC language at the National Research Institute for Mathematics and Computer Science in the 1980s by Guido van Rossom. Python has been focusing on readable code. The Raspberry Pi has been provided with many commands to make the language easy to learn and follow. The Python language is an open-source language that is freely available for any operating system, including Linux, Microsoft's Windows, and Apple's OS X [26, 27]. Python is a general-purpose language used in artificial intelligence (AI), numeric programming, image processing, and even biological studies. Furthermore, Python is easy and straightforward for a person to understand, yet it has among its advantages the possibility of fast and rapid changes to be made in the program as well as good performance and ability to be used for educational purposes [12].

5. Experimental Procedure

In this project, three python codes have been used; each Raspberry has an SD card responsible for saving the OS program and the python language codes. Raspberry 1 code, Raspberry 2 code, and the Graphical User Interface (GUI) which is an interface that used by users who interact with electronic devices via visual indicator representations.

5.1. Wiring Diagram

The main components of the circuit (Sensor, ZigBee, Raspberry Pi, and LC) are connected to perform the function as shown in Fig. 8.

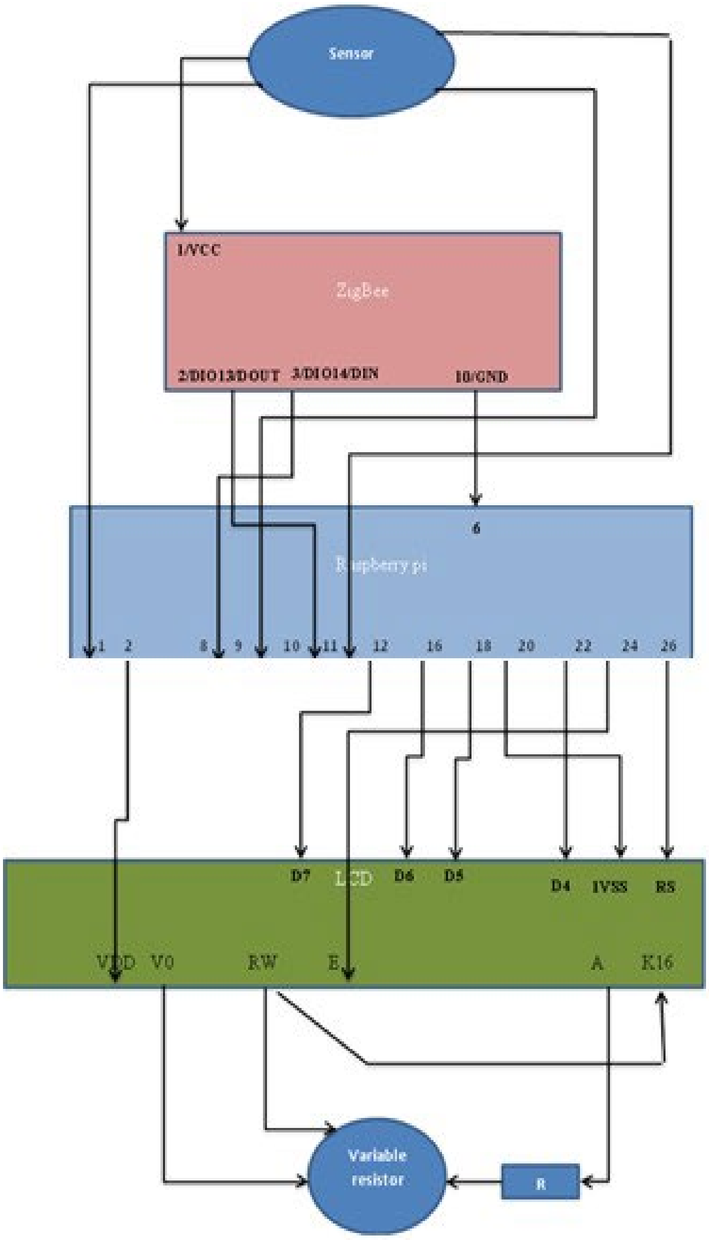


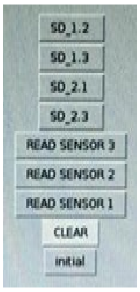
Fig. 8. Wiring diagram of WSN circuit.

5.2. Graphic User Interface (GUI)

The graphic user interface was used for monitoring and illustrating the collected data in the project. The GUI provides the real connection between the human and the device in the designed GUI. The 9 buttons provided for the user and the function of each button is illustrated in Table 5.

Table 5. The nine buttons of GUI.

Button	Function
SD_1.2	The coordinator sent sensor 2 results to raspberry 1 to show it on the LCD.
SD_1.3	The coordinator sent sensor 3 results to raspberry 1 to show it on the LCD.
SD_2.1	The coordinator sent sensor 1 results to raspberry 2 to show it on the LCD.
SD_2.3	The coordinator sent sensor 3 results to raspberry 2 to show it on the LCD.
Read Sensor 3	Read the sensor 3 data and show it on the GUI.
Read Sensor 2	Read the sensor 2 data and show it on the GUI.
Read Sensor 1	Read the sensor 1 data and show it on the GUI.
Clear	Clear the GUI screen.
Initial	Initialize the network.

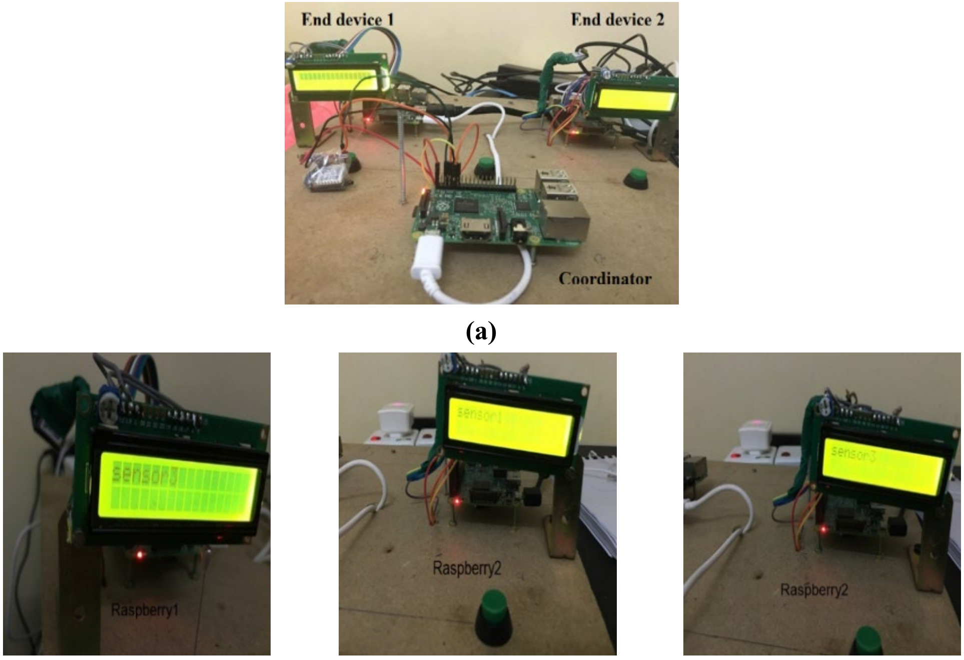


6. The Results

The results can be categorized under qualitative and quantitative contributions to communication technology. The qualitative results are shown in Figs. 9 and 10. The first part of the qualitative results shows the new design, as shown in Fig. 9(a). The new design comprises python codding, formatting the SD memory, and installing the Raspbian operating system on each Raspberry. The project circuit is connected using the pin configuration of each device and the wiring diagram of the practical circuit. In this design, the coordinator sent sensor 3 results to raspberry 1 on the LCD, Fig. 9(b). The coordinator sent sensor 2 results to raspberry 1 to be shown on the LCD, Fig. 9(c). The coordinator sent sensor 1 result to raspberry 2 to be shown on the LCD, Fig. 9(d). The information sent from Raspberry 1 and 2 to Raspberry 3 (coordinator) is carried out by the coordinator (ZigBee), which plays as receiver and transmitter in the wireless sensor network. The benefit of this technique is to allow sending the data to any Raspberry by clicking special buttons on the GUI.

The second qualitative result, which explains the role of WSN, is described in Fig. 10. These roles are sensing and reading each sensor and showing the exact Raspberry's LCD results, as shown in Fig. 10. For example, by activating Read Sensor 1, Fig. 10(a), the result is shown on the GUI. The same scenario for reading Sensor 2, Fig. 10(b) and Read Sensor 3, Fig. 10(c). This technique can provide much more flexibility of the data exchange not only by quantity but also by enhancing the rate of data exchange.

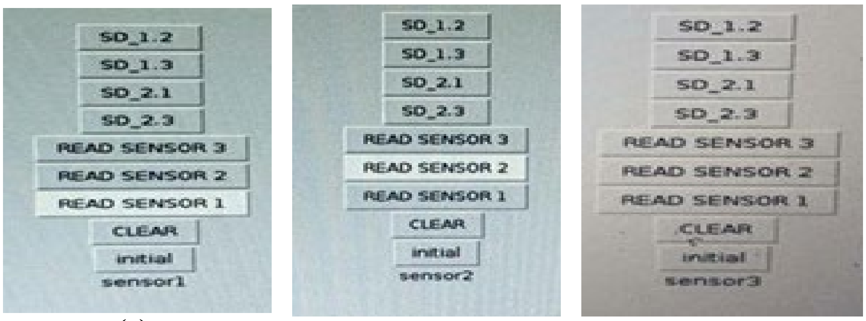
The second category of the results is the quantitative contribution. The new system can be operated at a maximum setting t which the power level could be boosted to provide a maximum transmit power of 20 dBm12 compared to 18 dB [28]. The new design was also tested for sensitivity measurements. The result was imposing by showing improvement by about 2 dB while the output power increases by little more than 2 dB compared to the results obtained by Jaladi et al. [29]. The new design also experimented with both multicast- and unicast-based communication schemes. The current design is configured as a router and end tag(E52) to enable sending the real-time data to the nearest router.



(a)

(b) (c) (d)

Fig. 9. (a) Embedded WSN Platform, (b) Coordinator and sensor 3, (c) Coordinator and sensor 2, and (d) Raspberry 2.



(a) (b) (c)

Fig. 10. GUI results via WSN transferring information (a) sensor 1, (b) sensor 2, and (c) sensor 3.

7. Discussion

The analysis of this study has shown a clear difference between Raspberry Pi 2 Model B and other versions of this device in terms of memory, CPU usage, and performance which was almost doubled compared to other versions of Raspberry models. The Raspberry Pi replaces the use of larger computer boards and, hence, reduces the total cost of the WSN designed.

Furthermore, the power usage of the network was considerably reduced by using ZigBee protocols instead of other wireless devices. Besides, the Python language is a clear and powerful object-oriented program. The use of the single-board computer in this project makes the project more cost-effective. Also, it is an easy-to-use language that makes it simple to run on any device.

Using python in the programming makes the work on the Raspberry easier and faster, and the language learning was more straightforward than previously reported. Other advantage of this technique is that it allows sending the data to any Raspberry by clicking special buttons on the GUI, capability to work with both multicast- and unicast-based communication schemes, and to work as router and end tag(E52) to enable sending the real-time data to the nearest router. Analytically, the new design provides a maximum transmit power of 20 dBm12 compared to 18 dBs and increases the output power by a little more than 2 dB.

8. Conclusions and Future Work

This study developed a wireless sensor network based on an SBC to send and receive data from sensors for data storage and display it on an LCD screen and GUI. The system could be developed further for use in many different applications such as measurement and surveillance of temperature, humidity, movement, speed, etc.

The main concept is using a small computer such as a Raspberry Pi that is cheaper than traditional desktop and laptop computers. The price of a Raspberry Pi is only $35, while the minimum price for a traditional computer can be around $300 to $400. This search provides an opportunity to work with the Raspberry Pi board, ZigBee, and LCD output device.

In this project, GUI has been successfully used for ZigBee devices to enable a quick and seamless operation. Future work could develop this system on the web frame concept to provide better flexibility to control the application and devices. Further, the GUI may be accessed on a server and have a direct link of communication between the client and gateway.

The GUI version of this project can be developed to be more convenient and user-friendly for user operation. The current wireless sensor system could allow users to monitor ambient temperature, improve the quality and control process, reduce manual data entry, and accelerate the manufacturing process in their manufacturing and storage facilities. Benefits include improved process and quality control, reduction in manual data collection, and accelerated manufacturing.

Future work can be elaborated to further experimental and studies to determine the coverage of each XBee radio transceiver in the selected deployment environment. This development is very helpful in designing the optimum deployment structure.

The Xbee module system can act as a coordinator whose quality sensor and interface capability can provide a digital output to the Raspberry Pi. The interference of the sensor node and with the base station could be used as temperature and humidity sensors. XBee can be employed as a coordinator with the assistance of UART protocol.

Abbreviations

App	Application
BD	Band Rate
DSP	Digital Signal Processor
DVI	Digital Visual Interface
FPGA	Field Programmable Gate Array
GPIO	General Purpose Input Output
GUI	Graphic User Interface
HDMI	High-Definition Multimedia Interface
LCD	Liquid Crystal Display
OS	Operating System
R/W	Read/Write
RAM	Random Access Memory
Rasp Pi	Raspberry Pi Device
RS	Register Select
SBC	Single Board Computer
SD Card	Secure Digital Card
SDHC	Secure Digital High Capacity
SDXC	Secure Digital Extended Capacity
USB	Universal Serial Bus
Vdd	Positive Voltage Supply
Vss	Negative Supply Voltage
WPAN	Wireless Personal Area Network
WSN	Wireless Sensor Network
Xbee	ZigBee

References

1. Nikoukar, A.; Raza, S.; Poole, A.; Güneş, M.; and Dezfouli, B. (2018). Low-power wireless for the Internet of Things: Standards and applications. *IEEE Access,* 6, 67893-67926.
2. Kandris, D.; Nakas, C.; Vomvas, D.; and Koulouras, G. (2020). Applications of wireless sensor networks: an up-to-date survey. *Applied System Innovation,* 3(1), 14.
3. Baire, M.; Melis, A.; Lodi, M.B.; Dachena, C.; Fanti, A.; Farris, S.; Pisanu, T.; and Mazzarella, G. (2019). WSN Hardware for Automotive Applications: Preliminary Results for the Case of Public Transportation. *Electronics,* 8(12), 1483.
4. Anand, S.; and Nath, V. (2019). Study and design of smart embedded system for remote health monitoring using internet of things. In: Nath, V., Mandal, J. (eds) *Nanoelectronics, Circuits and Communication Systems . Lecture Notes in Electrical Engineering,* 511, 409-414, *Springer, Singapore.*,
5. Johnston, S.J.; Basford, P.J.; Perkins, C.S.; Herry, H.; Tso, F.P.; Pezaros, D.; Mullins, R.D.; Yoneki, E.; Cox, S.J.; and Singer, J. (2018). Commodity single board computer clusters and their applications. *Future Generation Computer Systems,* 89, 201-212.
6. Maceli, M.; and Cocciolo, A. (2017). Monitoring environmental conditions with low-cost single board computers. *Preservation, Digital Technology and Culture,* 46(4), 124-131.

7. Manu, M.M.R.; Saha, R.; Hoque, S.A.; and Hoque, A. (2019). Remote environmental data analysis using sounding rocket. *International Journal of Engineering and Technology*, 11(6), 1209-1221.
8. Aju, O.G. (2015). A survey of zigbee wireless sensor network technology: Topology, applications and challenges. *International Journal of Computer Applications,* 130(9), 47-55.
9. Kuzminykh, I.; Snihurov, A.; and Carlsson, A. (2017). Testing of communication range in ZigBee technology. P*roceedings of the 14th International Conference the Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*, Lviv, Ukraine, 133-136.
10. Buratti, C.; Conti, A.; Dardari, D.; and Verdone, R. (2009). An overview on wireless sensor networks technology and evolution. *Sensors,* 9(9), 6869-6896.
11. Hao, J.; and Ho, T.K. (2019). Machine learning made easy: A review of scikit-learn package in Python programming language. *Journal of Educational and Behavioral Statistics,* 44(3), 348-361.
12. Srinath, K. (2017). Python - The fastest growing programming language. *International Research Journal of Engineering and Technology,* 4(12), 354-357.
13. Hakim, W.M.A.; Ramli, A.F.; Basarudin, H.; Abu, M.A.; and Ahmad, I. (2020). WSN and IoT based landslide monitoring system. *Test Engineering and Management*, 83, 10926-10932.
14. Shelar, D.; Gharpure, D.; and Shaligram, A. (2017). Performance analysis of ZigBee based wireless sensor network for grain storage monitoring. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering,* 6(6), 5027-5035.
15. Howser, G. (2019). *Raspberry Pi operating system.* In *Computer Networks and the Internet*, Chapter 8, Springer, Cham, 119-149.
16. Matthews, S.J.; Adams, J.C.; Brown, R.A.; and Shoop, E. (2019). Portable parallel computing with the raspberry. *SIGCSE '18: Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, 92-97.
17. Cicolani, J. (2018). *Beginning robotics with Raspberry Pi and Arduino*. Apress Berkeley, CA., 355-363.
18. Pajankar, A. (2017). *Raspberry Pi supercomputing and scientific programming. MPI4PY, NumPy, and SciPy for Enthusiasts. Apress*., 167-171.
19. Ojha, T.; Misra, S.; and Raghuwanshi, N.S. (2015). Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges. *Computers and electronics in agriculture,* 118, 66-84.
20. VivekBabu, K.; Reddy, K.A.; Vidhyapathi, C.; and Karthikeyan, B. (2017). Weather forecasting using raspberry pi with internet of things (IoT). *ARPN Journal of Engineering and Applied Sciences,* 12(17), 5129-5134.
21. Sollu, T.S.; Bachtiar, M.; and Bontong, B. (2018). Monitoring system heartbeat and body temperature using raspberry pi. *E3S Web of Conference,* 73(12003), 1-5.
22. El Sayed, A.; Harb, H.; Ruiz, M.; and Velasco, L. (2021). ZIZO: A zoom-in zoom-out mechanism for minimizing redundancy and saving energy in wireless sensor networks. *IEEE Sensors Journal,* 21(3), 3452-3462.

23. Molloy, D. (2019). *Exploring BeagleBone: Tools and techniques for building with embedded Linux*. Wiley.
24. Tsvetanov, F.; Georgieva, I.; and Pandurski, M. (2020). Topology influence on the radio communication range of sensor networks. *Technium: Romanian Journal of Applied Sciences and Technology,* 2(1), 17-24.
25. Sharma, S.; Chand, K.; Sharma, D.; and Guha, P. (2018). Development of an early detection system for fire using wireless sensor networks and Arduino. *2018 International Conference on Sustainable Energy, Electronics, and Computing Systems (SEEMS)*, Greater Noida, India, 1-5.
26. Mandanici A.; and Mandaglio, G. (2020). Experiments and data analysis on one-dimensional motion with Raspberry Pi and Python. *Physics Education,* 55(3), 033006.
27. Vaidya, B.; Patel, A.; Panchal, A.; Mehta, R.; Mehta, K.; and Vaghasiya, P. (2017). Smart home automation with a unique door monitoring system for old age people using Python, OpenCV, Android and Raspberry pi. *Proceedings of the 2017 International Conference on Intelligent Computing and Control Systems (ICICCS),* Madurai, India, 82-86.
28. Ferdoush, S.; and Li, X. (2014). Wireless sensor network system design using Raspberry Pi and Arduino for environmental monitoring applications. *Procedia Computer Science*, 34, 103-110.
29. Jaladi, A.R.; Khithani, K.; Pawar, P.; Malvi, K.; and Sahoo, G. (2017). Environmental monitoring using wireless sensor networks (WSN) based on IOT. *International Research Journal of Engineering and Technology,* 4(1), 1371-1378.