Available online www.jsaer.com

Journal of Scientific and Engineering Research, 2019, 6(10):44-47

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Temperature Control by Using Wireless Sensor Network (WSN) and LED

Yi-Jen Mon, Zhi-Yu Lin, Yu-Hua Peng, Shao-Min Lyu

Department of Computer Science and Information Engineering, Nanya Institute of Technology, Chung-Li, Taoyuan, 320, Taiwan, R. O. C.
monbuy@nanya.edu.tw

Abstract In this paper, the ZigBee wireless sensor network (WSN) is used to control LED such as to monitor temperature. WSN is based on Zigbee protocol which has benefits of low cost, low power consumption, etc. By using WSN's Application Program Interface (API), the LED can be controlled based on temperature easily. The example of verifying the performance of WSN will be presented in this paper. The results reveal that the method and experiments are good and satisfied.

Keywords Wireless sensor network (WSN), ZigBee, LED, Temperature, Monitor

1. Introduction

The ZigBee Alliance slogan, "Wireless Control That Simply Works," is clearly what is needed for end users and implementers, but achieving that result places a heavy burden on the developers and OEMs. Drew Gislason has the ability to take complex topics and present them in a manner that is cogent and easily digestible to OEMs and developers [1]. Today, organizations use IEEE 802.15.4 and ZigBee to effectively deliver solutions for a variety of areas including consumer electronic device control, energy management and efficiency, home and commercial building automation as well as industrial plant management [2].

We can monitor the environment for many applications by using the WSN. The benefits are reducing energy consumption such as to extend the active time of the WSN. The association of ZigBee Alliance Company have established many related protocol or standard for ZigBee. Then many standards of reliable, cost-effective, low-power wireless network have been developed. The ZigBee technology is embedded in many ranges of products which across industrial, consumer, commercial applications [3, 4]. The ZigBee defines the network layer specifications for star, tree and peer-to-peer network topologies and provides a framework for application programming in the application layer. The ZigBee is built on the so called IEEE 802.15.4 standard which defines the physical and MAC layers for WSN [5].

In this paper, the good control method and experiment for the WSN are presented. It is designed by LED to know the control method can be verified for monitoring temperature. The performances of data measured by WSN will be shown in computer screen then if the temperature higher than 28 °C the LED will be lighted, otherwise, LED will be closed. The experimental example of verifying the performance of WSN will be presented. The results reveal that the control method and experiments are good and satisfied.

2. Design Method for WSN

The ZigBee specification that you download describes how ZigBee works as a networking protocol. What is not included (at least at the time of this writing) is how applications interact through what are called Application Profiles. To access the Application Profile specifications, you must become a member of the ZigBee Alliance.

Application Profiles describe the over-the-air behavior of devices in each of the wireless control domains. In Telecommunications Applications (TA), for example, companies are creating products that can introduce location-based services such as Fig. 1. These services could advertise specials, services, or sales (at your option) based on the location of your cell phone. The cell phone company already knows where your phone is from a cell tower perspective, but that can be a 15-mile radius or more, far too broad for location-based services.

The main purpose of teaching for students are using application program interface (API) of WSN. This API can be developed to reduce the burden of program design. The Application Queue API is a very important tool to provide a queue-based interface to communicate application file and both the WSN stack and the hardware drivers. The develop environment is called Code::Blocks which can let all these APIs interacts with the IEEE 802.15.4 stack API. Meanwhile, it can interact with many Peripheral Hardware Drivers by Integrated Peripherals API. The Application Queue API handles interrupts between the Medium Access Control (MAC) sub-layer stack and integrated peripherals of wireless microcontroller to save the application time of interrupts [6].

A variety of network topologies have been developed. A star topology is used in this paper. A network must consist of a minimum of two devices. One is called Co-ordinator which is referred to as network commander. Others WSN are called End-Device which can be regarded as client or slave. Each End-Device node has a parent. The Co-ordinator node has one or many client nodes. Each node can communicate only with its parent and its client. Not all nodes may be within range of each others but information can be transferred from one node to another node until the final destination is got. The data transfer methods between network nodes have two types: (1) Transferring data from a Co-ordinator to a node. In this method, request response should be given to the node to request data when it is able to receive. (2) Transferring data from a client node to Co-ordinator node. In this method, confirm response should be sent from a data frame directly. The two data transfer methods are shown in Fig. 2.

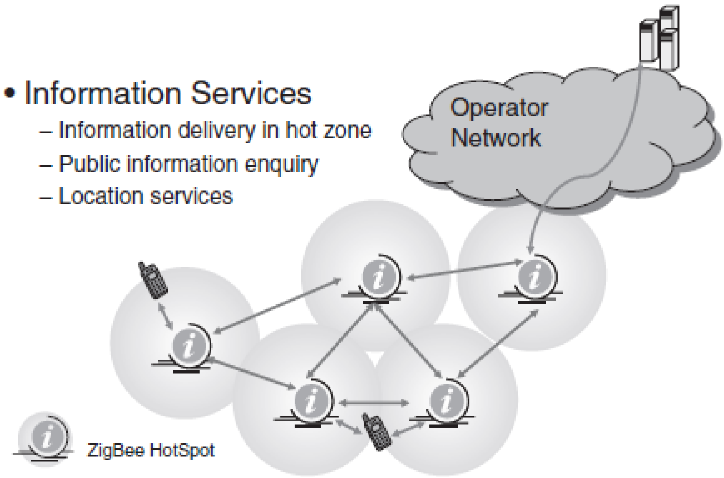


Figure 1: Telecom Application Information Services

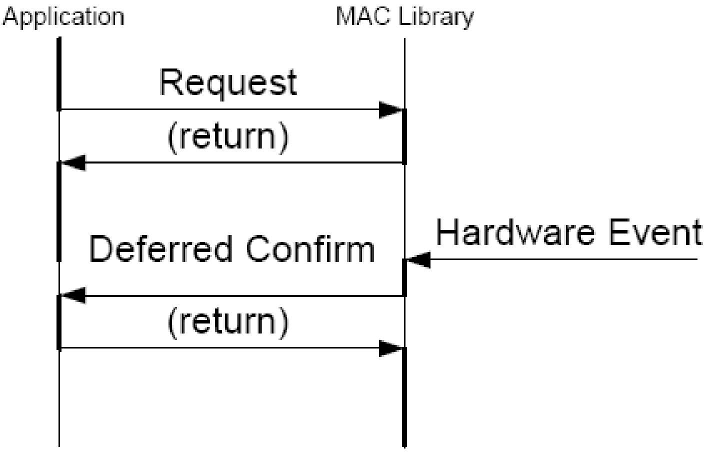


Figure 2: The diagram of WSN data transferring

3. Experiment Results

The program of WSN is developed on the free software called Code::Blocks. It can provide all the software tools and hardware required to get the first-hand experience with WSN. At first, the program of Co-ordinator is developed then the program of End-Device is developed consequently. Every network must have one and only one PAN Co-ordinator, and one of the tasks in setting up a network is to select and initialize this Co-ordinator. The network setup process is automatically started from stack initialization to data transferred. The entry-level kits contain one base development board (BDB) and one sensor development board (SDB). Each board is equipped with a high-power IEEE 802.15.4 RF module which provides much higher covering range with 2.4GHz RF antenna for easy mechanical design than normal-power RF module. For I/O expansion ports, it has 10 useful pins of GPIO include UART, ADC, DAC and Comparator. The sensor development board features temperature sensors such as Fig.3. When the temperature and humidity are greater than 28 °C, the external LED light flashes. If it is less than 25 °C, the LED light will go out and nothing will happen. This digital temperature control design can be applied to the intelligent air-conditioning switch. When the temperature reaches 28 °C, the cold air is automatically started. When the temperature drops below 25 °C, the power is automatically turned off. The experimental results are shown in Fig. 4.

The API software provides free packages to the peripheral devices single-chip compliant wireless microcontrollers. This is known as the Integrated Peripherals API. It details the calls that may be made through the API. This hardware is controlled using the MAC software stack which is built in the on-chip ROM. The developed program is shown in Fig. 5. In this paper, the WSNs are used to monitoring temperature. The experimental example of verifying the performance of WSN has been presented in this paper. The results reveal that the control method and experiments are good and satisfied.

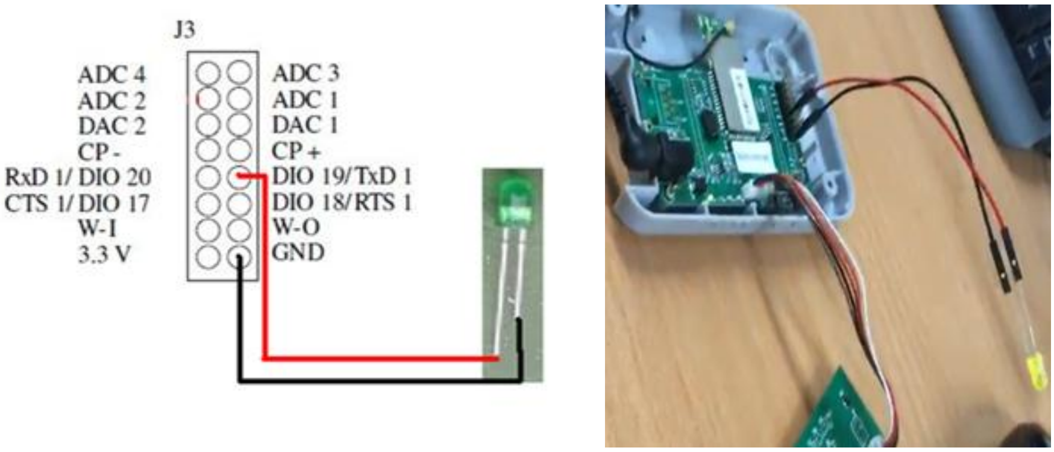


Figure 3: The development boards of WSN

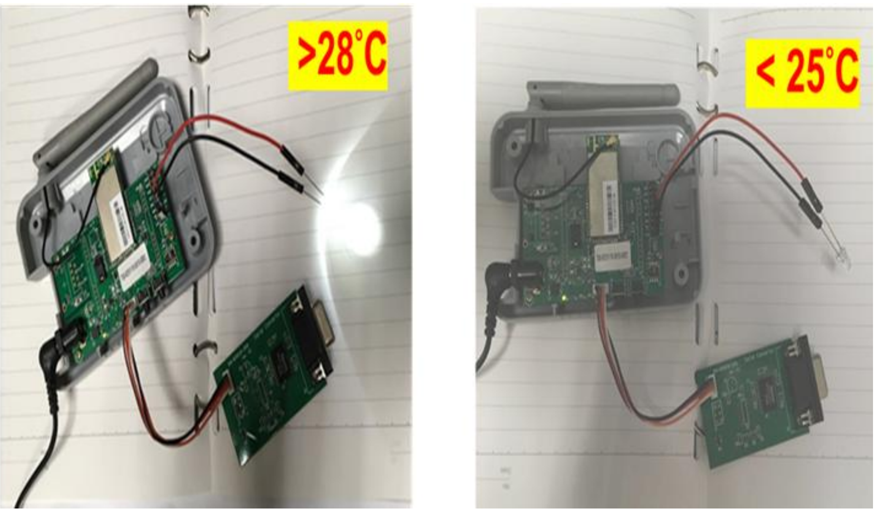
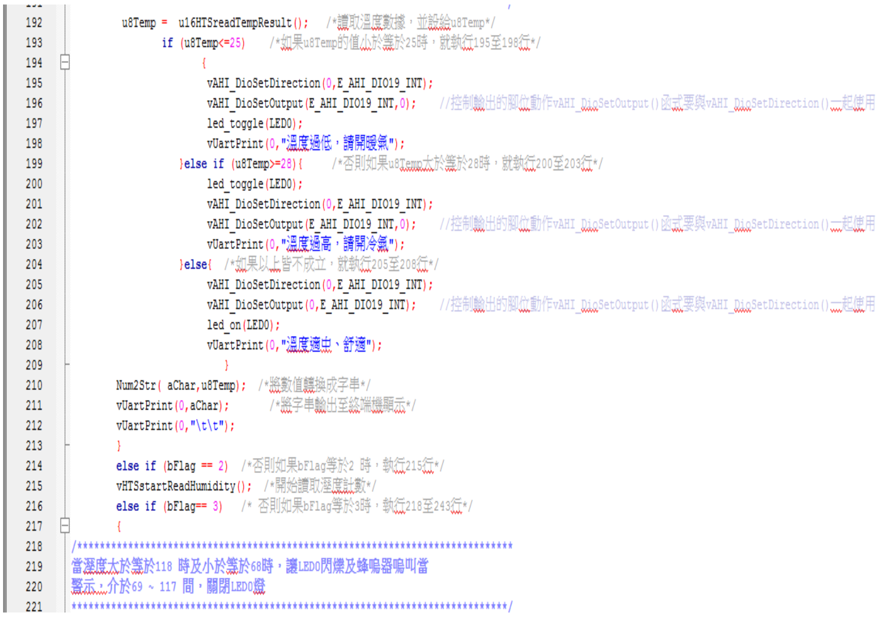


Figure 4: The diagram of experimental results



```
192             u8Temp =  u16HTSreadTempResult();   /*讀取溫度數據，並設給u8Temp*/
193                 if (u8Temp<=25)    /*如果u8Temp的值小於等於25時，就執行195至198行*/
194                     {
195                     vAHI_DioSetDirection(0,E_AHI_DIO19_INT);
196                     vAHI_DioSetOutput(E_AHI_DIO19_INT,0);    //控制輸出的腳位動作vAHI_DioSetOutput()函式要與vAHI_DioSetDirection()一起使用
197                     led_toggle(LED0);
198                     vUartPrint(0,"溫度過低，請開暖氣");
199                 }else if (u8Temp>=28){     /*否則如果u8Temp大於等於28時，就執行200至203行*/
200                     led_toggle(LED0);
201                     vAHI_DioSetDirection(0,E_AHI_DIO19_INT);
202                     vAHI_DioSetOutput(E_AHI_DIO19_INT,0);    //控制輸出的腳位動作vAHI_DioSetOutput()函式要與vAHI_DioSetDirection()一起使用
203                     vUartPrint(0,"溫度過高，請開冷氣");
204                 }else{  /*如果以上皆不成立，就執行205至208行*/
205                     vAHI_DioSetDirection(0,E_AHI_DIO19_INT);
206                     vAHI_DioSetOutput(0,E_AHI_DIO19_INT);    //控制輸出的腳位動作vAHI_DioSetOutput()函式要與vAHI_DioSetDirection()一起使用
207                     led_on(LED0);
208                     vUartPrint(0,"溫度適中、舒適");
209                         }
210         Num2Str( aChar,u8Temp);  /*將數值轉換成字串*/
211         vUartPrint(0,aChar);       /*將字串輸出至終端機顯示*/
212         vUartPrint(0,"\t\t");
213         }
214         else if (bFlag == 2)  /*否則如果bFlag等於2 時，執行215行*/
215         vHTSstartReadHumidity();  /*開始讀取溼度計數*/
216         else if (bFlag== 3)   /* 否則如果bFlag等於3時，執行218至243行*/
217         {
218 /***************************************************************************
219 當溼度大於等於118 時及小於等於68時，讓LED0閃爍及蜂鳴器鳴叫當
220 警示，介於69 ~ 117 間，關閉LED0燈
221 ***************************************************************************/
```

Figure 5: The program diagram of monitoring temperature

4. Conclusion

In this paper, the control method and experiment for temperature monitoring by WSN and LED are proposed. The control example and experiment result of verifying the performance of WSN and LED have been presented. The results reveal that the teaching method and experiments are good and satisfied.

References

[1]. D. Gilslason, "*Zigbee Wireless Networking*," Newnespress, 2008.

[2]. D. M. Han, J. H. Lim, "Smart home energy management system using IEEE 802.15.4 and zigbee," *IEEE Trans. on Consumer Electronics*, Vol. 56 , No. 3 , pp. 1403-1410 , 2010.

[3]. I. J. Su, C. C. Tsai and W. T. Sung, "Area temperature system monitoring and computing based on adaptive fuzzy logic in wireless sensor networks, *Applied Soft Computing*," vol. 12, pp. 1532-1541, 2012.

[4]. N. Golmie and I. Matta, "Applications and services in wireless networks," *Computer Communications*, vol. 28, pp. 1603-1064, 2005.

[5]. L. Aguilar, G. Licea and J. A. García-Macías, "An experimental wireless sensor network applied in engineering courses," *Computer Applications in Engineering Education*, vol. 19, pp. 777-786, 2011.

[6]. J. Yick, B. Mukherjee and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, pp. 2292–2330, 2008.