The GPS information acquisition system based on Zigbee

Zhenghua Xin[1], Liangyi Hu[2*], Guolong Chen[1], Qixiang Song[1], Hong Li[1]

[1]*School of Information and Engineering, Suzhou University, Anhui Province, 234000, China*

[2]*School of Economics and Management, Suzhou University, Anhui Province, 234000, China*

Received 1 October 2014, www.cmnt.lv

Abstract

This system designs the global positioning system (GPS) module in the CC2530 chip. So it is called the GPS wireless sensor node. It makes the GPS combine the Zigbee node. The Zigbee is the communicating protocol for the wireless sensors. The location information is sent from the terminal sensor through the Zigbee protocol to the coordinator wirelessly. The coordinator sends the GPS information to the PC or other terminal via the serial. At last, the personal computer can communicate the position information with other users via a wired or wireless network. If the GPS information timely can be transmitted by the Zigbee wireless network, then you can get the more accurate positioning system through the software in the upper host computer. It will be widely applied in the field of the industry and the agriculture.

Keywords: GPS, Zigbee, CC2530

1 Introduction

Many museums lost precious items. This system can help solve this problem. A GPS module based on Zigbee and the CC2530 chip can be put on the valuable item. The system will immediately detect whether the item moves or not in the computer. The wireless sensor net is usually including the CC2530 and the Zigbee protocol [1,2]. Global Positioning System (GPS) has been widely used all around the world. Many places are covered with WLAN signals. Therefore, comprehensive utilization of the satellite signal and WIFI signals of WLAN to realize the optimization of GPS location become possible [3]. This paper incorporates the satellite signals and wireless sensor net (WSN) signals based on Zigbee. The wireless sensors are used as the information collection terminals. The research [4] considers the intelligent perception problem of Internet of Things (IoT) based on context perception to solve the uncertain perception information appears in the perception process of intelligent wireless sensor.

In application, in view of the feature that wireless sensor network (WSN) must possess auto organization, auto-adaptation and robustness, especially, energy of WSN is very limited, this paper fully utilizes the advantages of computational intelligence, marries together both the research focuses. The paper [5] proposes some methods and ideas for applying computational intelligence to solve optimization problems of WSN. It depicts coverage problem of WSN, for the feature that this problem is the problem of multi objective optimization, under the topology control of GA to solve the problem. The wireless communication is also based on the infrared or the Bluetooth [6]. But the GPS module transmits the complex information.

Localization is one of the most important tasks for location-aware applications in wireless sensor networks (WSNs). The accuracy of positioning information gives the applications a great number of advantages. Thus, the research [7] reviews three typical localization schemes for WSNs and, then, compares their performance via computer simulation in terms of localization accuracy.

In order to precisely predict the location of forest fire, a new algorithm based on improvement of gravity centre scan method is introduced in the paper [8]. By using the neighbour nodes around the unknown nodes, it solved the problem of less signal flag nodes around unknown nodes; furthermore, it improved the coverage of nodes and suppressed error rate the location of node.

The CC 2530 chip is the true system-on-chip (SoC), the popular solution for Zigbee, IEEE 802.15.4 and RF4CE applications in things of internet. It enables robust wireless sensor network nodes to be built with very low total costs. The CC2530 combines the excellent performance of a leading RF transceiver with an industry-standard enhanced 8051 MCU, 8-KB RAM, in-system programmable flash memory, and other features [9].

Generally, two GPS modules are hard to send their tested location data to the same serial port at the same time in the design of the hardware. But through the wireless sensor network, the GPS information from different sensors can be got in the same serial port of the same coordinator. This is what the research has provided.

2 Development environment

The design of the system includes the development of software and hardware. Qt is the software development environment. The IAR-ewarm 6.10.1 is mainly used to develop the smart sensor terminals. The IAR system is the

* ***Corresponding author's*** e-mail: 40793653@qq.com

leading provider of embedded development tools and services suppliers. It includes the integrated development environment (IDE) with C/C+ + compilers and debuggers, real-time operating systems and middle-ware, development kits, the emulator of the hardware and the state machine modelling tool.

This system is designed primarily to use the WSN-GPS-BCC2530 as the sensor node. This section is divided into the power board and sensor modules. The power supply board connects with the upper host computer through the serial port. The serial port is also as the programmer interface for the simulation program. The power board can offer the scalable interfaces linking different functional modules. In this system, it connects the GPS module and the Zigbee wireless module. 5V DC power supply and two batteries are also able to supply the power board. The power switch board will help choose the desired power supply. Each board will supply the power of four LED lights to test whether the corresponding function initializes properly or not in the slot. Next to the power board is a reset button.

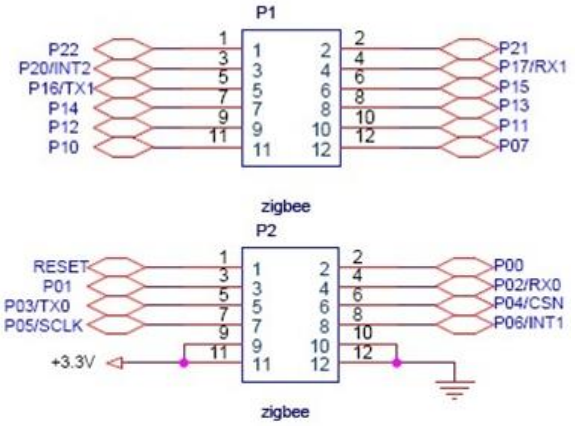


FIGURE 1 The linking pins of the Zigbee

The CC2530 drives the Zigbee to run through connectting the pins of the interface correctly.

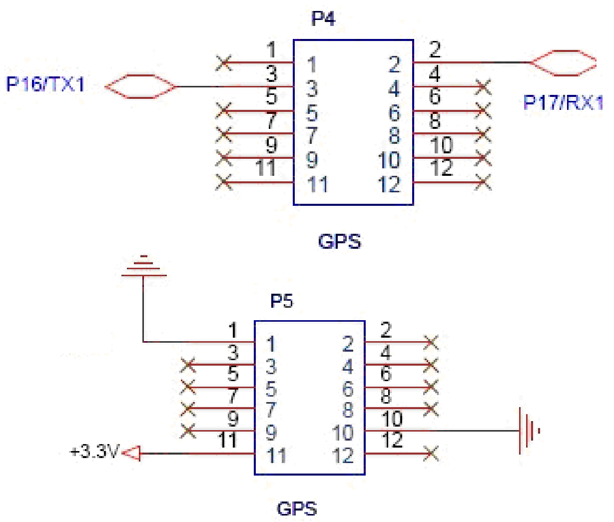


FIGURE 2 The GPS interface of the system

The pins of the is shown as the following figure. The GPS module is plugged in the CC2530 chip.

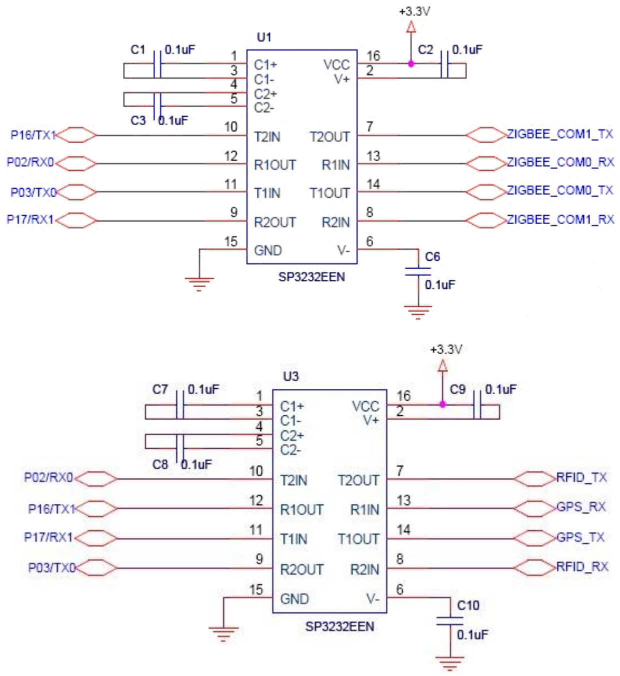


FIGURE 3 The linking pins of the Zigbee and the GPS

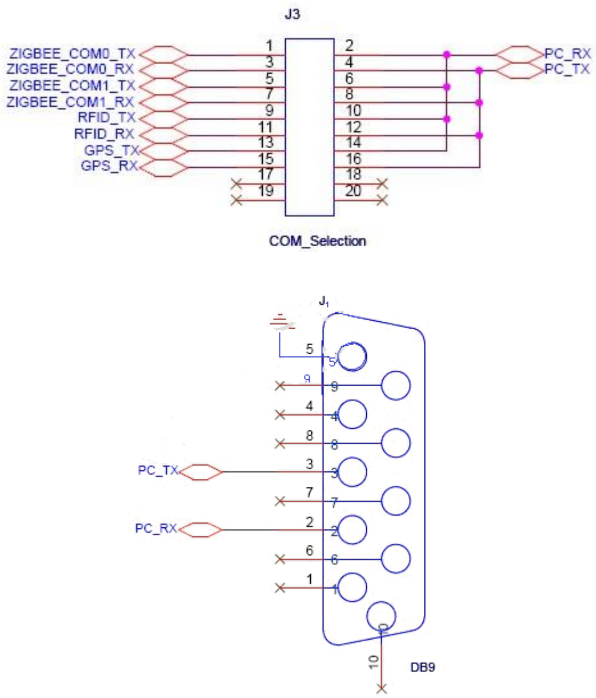


FIGURE 4 The configuration of the serial port

The system design is to send the GPS information and the temperature and humidity to the same serial port.

3 System architecture

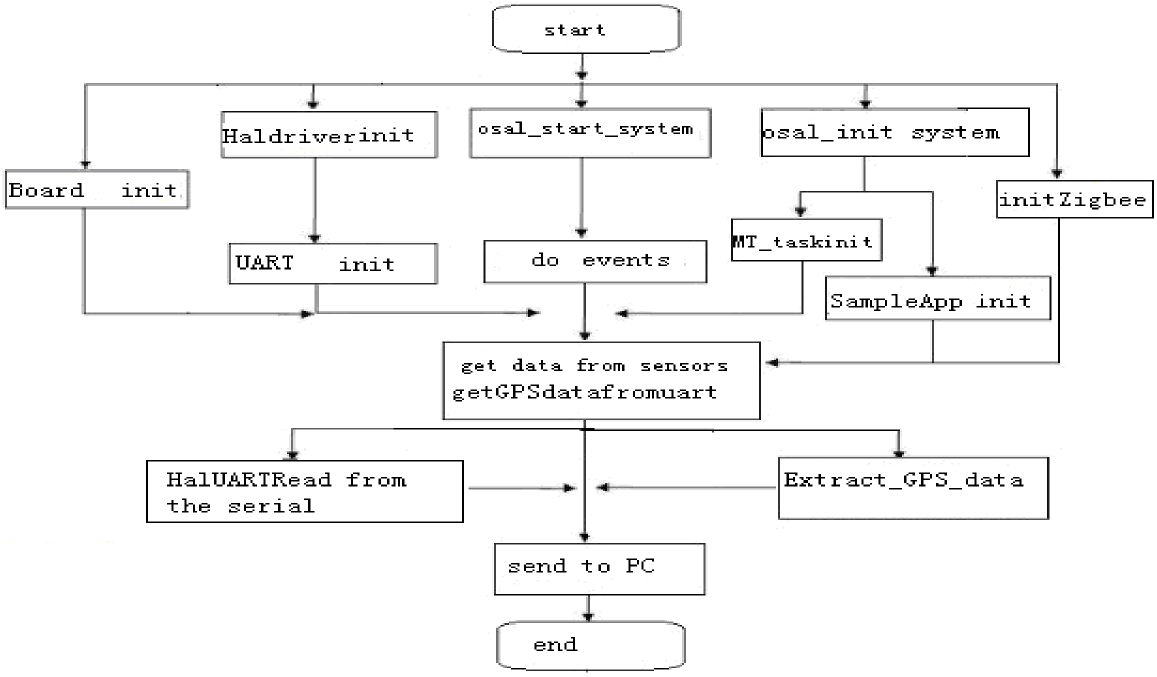


FIGURE 5 The system flow of the coordinator

After this system is powered normally, it will first perform the initialization of hardware. The detailed execution process is as the following figure. It includes the function Init_board to init the motherboard, the function HalDriverInit for driver initialization, Init_lcd for the lcd initialization, Init_led for the led initialization and software initialization function zgInit wireless Zigbee initialization and osal_init_system. After initialization is complete hardware and software system function is executed immediately osal osal_start_system ().

In the following figure, the module of the left is the GPS module embedded in the CC2530 chip. The other module is the coordinator module. The GPS sensor is connected with the coordinator based on the Zigbee in the wireless way. The coordinator is linking with the upper host computer through the RS232 serial shown as the following Figure 6.

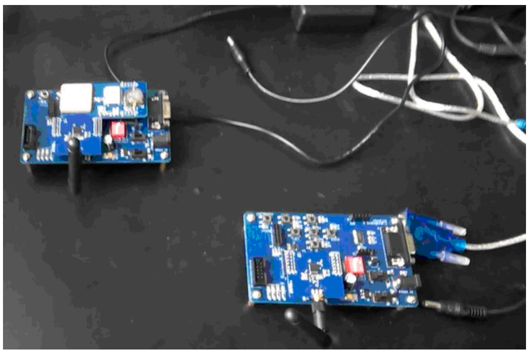


FIGURE 6 The physical GPS sensor and the coordinator

The Zigbee protocol uses the IEEE802.15.4-2003 agreement. The coordinator takes charge of the formation of network automatically. After initialization, the coordinator is starting the entire network. It is also the first one equipment in the Zigbee network. It selects a channel and a network ID, and then starts the whole network. Coordinator can also be used to help establish the binding network security layer and application layer. Each GPS sensor node will automatically join the network. The data frame structure from the node to the coordinator is the16-byte MAC address, 4-byte network address of the node, the 4-byte parent node network address, 2-byte parity and a byte trailing. The terminal node can be waked up or in the sleep state.

Positioning accuracy of GPS module depends on many aspects, such as the satellite clock error and the track error, the number of visible GPS satellites and their geometric distribution, solar radiation, atmospheric, multi-path effects and so on.

The function of the system SampleApp_GetGpsDataFromUart () can get the GPS information. The HalUARTRead () function is to read data from the cache of the serial port UART. it polls the p_0 and p_1 serial. In the process, the length of the data packet SENSOR_DATA is 35bit. It adopts the common structure. The first two bits are the message header. The next 3-bit is the command header. Then 4-bit sensor data, 19-bit GPS position information, 4-bit address of the network node, 1-bit check code, and the last two bits finish the transmission of the message. With the help of the serial port, the received GPS data is shown in the Figure 7.

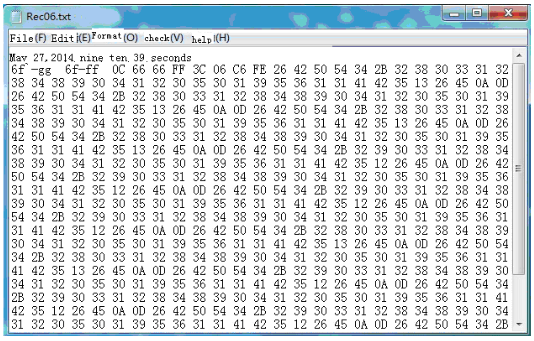


FIGURE 7 The received data from GPS

The coordinator sends the information from its serial port to the personal computer through the RS232 interface. The system design achieves the common manipulation for the serial port to get GPS data and the temperature and humidity data by HalUARTOpen (). The function opens port0 or port1 orderly according to the configuration parameters. The HalUARTRead () function reads data from the cache. And the corresponding HalUARTWrite () function is used to write data back to the cache by port0 or port1. The design not only solves only one serial port hardware interface can be used to get the data from the temperature and humidity module and GPS module. And the temperature and humidity module and GPS module comes from different nodes.

The detailed struct data of the GPGGA is as follows.

$ GPGGA, <1>, <2>, <3>, <4>, <5>, <6>, <7>, <8>, <9>, M, <10>, M, <11> , <12> * xx <CR> <LF>

The $ GPGGA means the guide symbol for the initial and statement format description for the GPS positioning data.

<1> stands fro the UTC time. Its format is hhmmss.sss.

<2> is the Latitude. Its format is ddmm.mmmm. If its first bit is zero, it will be transferred).

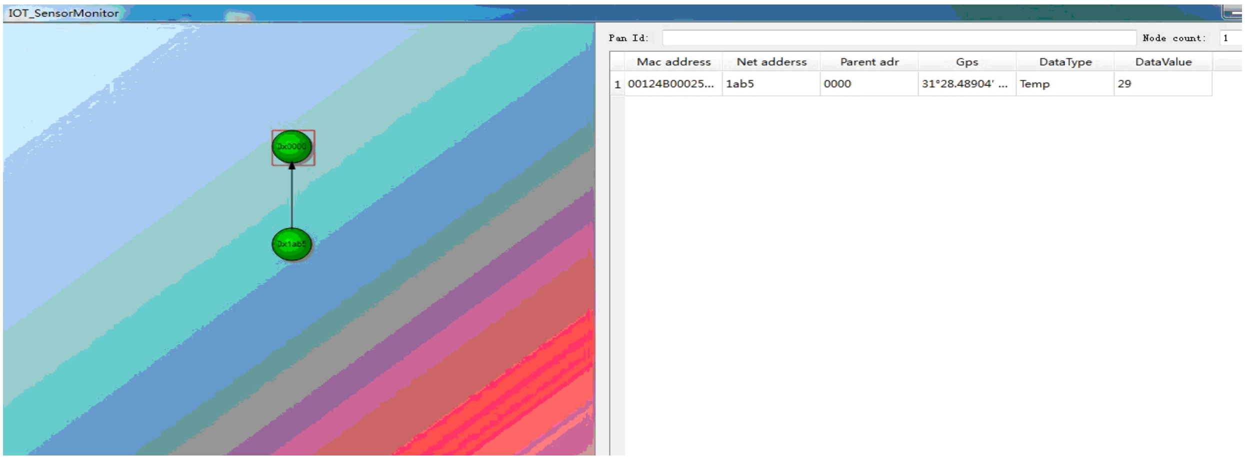


FIGURE 9 The topology displayed in the software

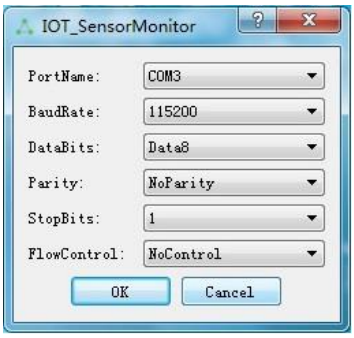


FIGURE 8 The login of serial port

<3> is the hemisphere where the Latitude is from. The N or S means the latitude or latitude.

<4> is the longitude. Its format is dddmm.mmmm. Even if its first bit is a zero, the data will be transferred).

<5> stands for the hemisphere where the Longitude is from. Its value is E or W.

<6> indicats the quality of the location. If its value is 0, the position is invalid. 1 is effective.

<7> means the number of used satellites. Its range is from 00-12. Even the first bit is zero, it will be transferred.

<8> is the accuracy of the level, from 0.5 to 99.9.

<9> is the height from the antenna to above the sea level. Its range is from -9999.9 to 9999.9 meters.

<10> is the geoid's height from -9999.9 to 9999.9 meters.

<11> is the differential GPS data period (RTCM SC-104). And finally it is the number of the seconds that the RTCM transmission establishes.

<12> means the differential reference station label, from 0000-1023. If its first bit is 0, it will be transmitted.

4 The Running Results of the system

This system uses QT as the development platform. It is good at the human-computer interaction, logic operations and the user interface (UI). Good UI design is not only to allow the software to be specific, but also to make the software operate comfortably and freely. This system is a data acquisition system based on the hardware sensors. And it displays the related data on the personal computer. So the system's interface is mainly to show the network topology through the serial testing assistant.

This research develops the application software based on the Qt platform. The software displays the network topology and related information about the terminals. As shown in Figure 9, the Mac address of the child sensor node is 00124800025, its network address is 1ab5, the information of GPS is $11^{0}28.4$, and its parent node address is 0000. In this figure, there is only one GPS sensor node. In fact, many GPS child sensors are also working collaboratively and send their respective GPS information to the coordinator. And then the coordinator displayed all the GPS information in order.

Some main codes of the system are as follows.

```
void SampleApp_Init( uint8 task_id )
{
  SampleApp_TaskID = task_id;
  SampleApp_NwkState = DEV_INIT;
  SampleApp_TransID = 0;
  //Initializing the serial port
  MT_UartInit();
  //registering the task
  MT_UartRegisterTaskID(task_id);
 //Initializing the temperature sensor
  P0SEL &= 0x7f;
  //P0_7port
// Device hardware initialization can be added here or in
//main() (Zmain.c).
  // If the hardware is application specific - add it here.
// If the hardware is other parts of the device add it in
//main().
  #if defined ( BUILD_ALL_DEVICES )
  // The "Demo" target is setup to have
//BUILD_ALL_DEVICES and HOLD_AUTO_START
  // We are looking at a jumper (defined in
//SampleAppHw.c) to be jumpered
  // together - if they are - we will start up a
coordinator. //Otherwise, the device will start as a router.
  if ( readCoordinatorJumper() )
   zgDeviceLogicalType =
ZG_DEVICETYPE_COORDINATOR;
  else
   zgDeviceLogicalType =
ZG_DEVICETYPE_ROUTER;
 #endif // BUILD_ALL_DEVICES
 #if defined ( HOLD_AUTO_START )
  // HOLD_AUTO_START is a compile option that
//will surpress ZDApp
  //  from starting the device and wait for the
//application to start the device.
  ZDOInitDevice(0);
 #endif
// Setup for the periodic message's destination address
  // Broadcast to everyone
  SampleApp_Periodic_DstAddr.addrMode =
(afAddrMode_t)AddrBroadcast;
  SampleApp_Periodic_DstAddr.endPoint =
SAMPLEAPP_ENDPOINT;
  SampleApp_Periodic_DstAddr.addr.shortAddr =
0xFFFF;
 // Setup for the flash command's destination address -
//Group 1
  SampleApp_Flash_DstAddr.addrMode =
(afAddrMode_t)afAddrGroup;
  SampleApp_Flash_DstAddr.endPoint =
SAMPLEAPP_ENDPOINT;
  SampleApp_Flash_DstAddr.addr.shortAddr =
SAMPLEAPP_FLASH_GROUP;
  //The defination for the point to point comunication
  Point_To_Point_DstAddr.addrMode =
(afAddrMode_t)Addr16Bit;
  Point_To_Point_DstAddr.endPoint =
SAMPLEAPP_ENDPOINT;
  Point_To_Point_DstAddr.addr.shortAddr = 0x0000;
  Client_To_Server_DstAddr.addrMode =
(afAddrMode_t)Addr16Bit;
  Client_To_Server_DstAddr.endPoint =
SAMPLEAPP_ENDPOINT;
  Client_To_Server_DstAddr.addr.shortAddr =
0x0000;
 // Fill out the endpoint description.
  SampleApp_epDesc.endPoint =
SAMPLEAPP_ENDPOINT;
  SampleApp_epDesc.task_id =
&SampleApp_TaskID;
  SampleApp_epDesc.simpleDesc
      = (SimpleDescriptionFormat_t
*)&SampleApp_SimpleDesc;
  SampleApp_epDesc.latencyReq = noLatencyReqs;
 // Register the endpoint description with the AF
  afRegister( &SampleApp_epDesc );
 // Register for all key events - This app will handle
all //key events
  RegisterForKeys( SampleApp_TaskID );
  // Register callback evetns from the ZDApp
  ZDO_RegisterForZDOMsg( SampleApp_TaskID,
NWK_addr_rsp );  ZDO_RegisterForZDOMsg(
SampleApp_TaskID, Match_Desc_rsp );
 // By default, all devices start out in Group 1
  SampleApp_Group.ID = 0x0001;
  osal_memcpy( SampleApp_Group.name, "Group 1",
7);
  aps_AddGroup( SAMPLEAPP_ENDPOINT,
&SampleApp_Group );
 #if defined ( LCD_SUPPORTED )
  HalLcdWriteString( "SampleApp",
HAL_LCD_LINE_1);
 #endif
 }
 void
SampleApp_GetGpsFromUart(RFSENSOR_DATA
*rfdata)// get GPS data
 {
  char GPS_BUFFER[200] = {0};
```

```
    int sum = 0, count = 0;
    while(sum < sizeof(GPS_BUFFER))
     {
     sum = HalUARTRead(0,GPS_BUFFER+count,sizeof(GPS_BUFFER));
    sum += count;
     }
    //extract effect data
    ExtractGpsData(GPS_BUFFER, rfdata->BUF.gpsData, rfdata->BUF.gpsData+9);
    }
    void ExtractGpsData(const char *src, char *latitude_data, char *longitude_data)
    {  // get the GPS data
       int m=0, n=0;
       char tmpdata[2][13] = {{'\0'},{'\0'}};
       char *p = NULL, *q = NULL;
       const char *tmp = "$GPRMC";
       if((p=strstr(src,tmp))==NULL){
          return;
       }
       q=strtok(p,",");
       while(q){
       if(n==3||n==5){
       strcpy(tmpdata[i++],q);
       }
       q=strtok(NULL,",");
       n++;
       }
       printf("%s\n", tmpdata[0]);
       printf("%s\n", tmpdata[1]);
      //fill with the latitude data
      for(m=0;m<9;m++){
         if(m!=4){
            *latitudedata = tmpdata[0][i];
            latitudedata++;
         }
      }
      //fill with the longitude data
      for(m=0;m<10;m++){
         if(m!=5){
            *longitudedata = tmpdata[1][i];
            longitudedata++;
         }
      }
    }
unsigned char check(const void * data, size_t size)
{
      unsigned char *pos = NULL;
      unsigned char *first = (unsigned char *)data;
      unsigned char ch = *first;
      for(pos = first+1; pos < first+size; ++pos)
      {
            ch ^= *pos;
      }
return ch;
}
```

5 Conclusion

This system uses the CC2530 chip as the master chip to realize the position system based on the Zigbee. It designs the related circuits. By the Zigbee protocol, the wireless sensor net about positioning is formed. On the Qt development platform, the connection between sensors is showed in the form of the diagram. And the GPS information is displayed in detail. It will be widely applied in the intelligent monitoring system.

Acknowledgements

This paper are sponsored by the Suzhou University student scientific research initiating projects (KYLXLKYB14-35, KYLXLKYB14-36, KYLXLKZD14-19), University science research project of Anhui Province (KJ2011B180, KJ2012A263, KJ2012B185), Quality project of Anhui Province (2012jyxm545) and Quality project of Suzhou University, the course construction of Internet of things Engineering under the Collaborative Innovation environment (szxyjyxm201314), The practice teaching quality guarantee system construction for the IT professional engineering (szxyzlbz201305), Anhui Province Natural Science Research Major Project, Research on P2P multi-channel video encoding key technologies (KJ2014ZD31), Anhui Province Natural Science Research Key Project, Research on vegetable supply chain traceability Platform Based on the quality and safety(KJ2014A247). The innovation training program projects for the Anhui College students, Coal Mine Personnel Positioning System based on the RFID (201310379083), The implementation of embedded alarms management system based on A900 (201310379023).

References

[1] Xin Z, Chen G, Li X, Hu L, Yuan J, Gu H, Cao P 2013 Research On the ZigBee Network and Equipment Design Based on the CC2530 *Sensors & Transducers* **158**(11) 89-94

[2] Xin Z, Li H, Song Q, Lu H, Yang B, Dong Q, An H, Liu H 2013 The intelligent transportation system using the infrared sensors based on the ZigBee protocol and Eclipse *Sensors & Transducers* **158**(11) 95-9

[3] Zhang F, Wu X, Zhang S 2013 GPS Location Accuracy Improvement By Wlan *Computer Modelling and New Technologies* **17**(4)

[4] Zhang Y, Zhang F 2014 Research On The Smart Wireless Sensor Perception System And Its Application Based On Internet Of Things *Computer Modelling and New Technologies* **18(**1)

[5] Sun Z, Yang T, Shu Y 2014 Wireless Sensor Networks Optimization Covering Algorithms Based On Genetic Algorithms *Computer Modelling And New Technologies* **18**(4)

[6] Xin Z, Song Q, Hu L, et.al. 2013 The Research on the Wireless Audio Transmission System based on the Infrared *Information technology Journal* **12(**23) 7610-5

[7] Mukhopadhyay S K, Ghosh S, Chakraborty S, Das S, Mitra M, Mitra S 2014 Comparative Performance Study of Localization Schemes in

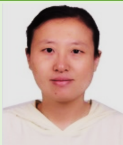
Wireless Sensor Networks *International Journal On Smart Sensing And Intelligent Systems* **8**(1) 95-102

[8] Liu D, Zhang Y 2014 The Positioning Algorithm Research for Forest Fire Prediction *International Journal of Smart Home* **8**(3) 181-19

[9] Yang Y, Xin Z, Hu L, Zhang C 2013 Research and the Implementation on the Intelligent Control System Based on the CC2530 *Journal of Suzhou University* **28**(3) 72-4

[10] Zheng J, Wu C, Li U D, Han Z, Bai L 2008 The Design of Alarming System Based on Wireless Sensor Network Technology in Intelligent Home *Journal of Shenyang Jianzhu University (Natural Science)* **24**(2) 345-8

Authors



Xin Zhenghua, P.R. China.

Current position, grades: lecture of Suzhou University.
University studies: master degree from Hebei University of Technology, China.
Scientific interest: cloud computing, internet of things.
Publications: 17 papers.

Hu liangyi, P.R. China.

Current position, grades: lecture of Suzhou University.
University studies: master degree from Yunnan University, China.
Scientific interest: industrial economics.
Publications: 10 papers.

Guolong Chen, P.R. China.

Current position, grades: professor, party secretary of Suzhou University in Anhui province.
University studies: doctor degree from Beijing Normal University, the postdoctoral from the institute of Software in China.
Scientific interest: digital computer, cloud computing.
Publications: 35 papers.
Experience: 10 national and provincial scientific research projects.

Qixiang Song, P.R. China.

Current position, grades: professor, Dean of School of Information and Engineering, Suzhou University, Anhui Province.
University studies: bachelor degree from Anhui University.
Scientific interest: physics, cloud computing.
Publications: 15 papers.

Hong LI, 1965 Anhui province, P.R. China.

Current position, grades: professor China University of Mining and Technology, Xuzhou in Jiangsu province.
University studies: master degree in computer science and technology from Hefei University of Technology, China.
Scientific interest: intelligent computing, algorithm design and analysis, data mining.
Publications: 3.