The wireless networking research based on Zstack

QingXiu Wu[1, a], ShuQing Li[1, b], GuoFang Zhang[1, c]

[1]Hainan College Of Software Technology, Qionghai, Hainan Province, China

[a]wu_qingxiu@aliyun.com, [b]hnzs02@163.com, [c]hncst@aliyun.com

Keywords: Zstack; wireless network; ZigBee.

Abstract. Zstack is an industry-leading ZigBee protocol stack launched by TI company, it includes protocol stack that covers almost all of the features of mesh and network topology, and occupies a very important position in the fierce competitive zigbee field. This paper describes software framework of Zstack protocol stack, and realizes the wireless communication in the experimental platform, its purpose is to master the basic principles and methods of wireless network communication based on Zstack.

Introduction

The ZigBee alliance complete the ZigBee protocol specification, the specification defines the realization mechanism of various functions in ZigBee network, which is completed together by provided by each layer of the protocol stack called as the " primitive " operation.Because each layer of primitive operations defined by the IEEE802.15.4 standard and the ZigBee protocol specification of have as many as dozens of items, and the process of primitive operations is very complex, which lead to the use of a single task of software can not implement these complex function operation.For the complicated communication software, it must rely on the embedded operating system to implement.

The current ZigBee protocol stack software mainly includes TI company ’ZStack, Freescale company ’BeeStack, Microchip company ’MpZBee and Ember company ’EMZNet. In addition, there are open source protocol stacks that isn’t supported by Business Companies, such as: msstatePAN protocol stack, freakz protocol stack and TinyOS. For msstatePAN, freakz and TinyOS protocol stack because they aren’t have major support of big Business Companies, compared with business software ,there is a big gap in ways of software performance, development and upgrade . In many commercial software, ZStack software because of its excellent ZigBee and ZigBee PRO feature set is rated as the highest industry level of ZigBee alliance by ZigBee Testing Institute of the national technology Service Corporation (NTS). Therefore, the paper uses TI company's ZStack protocol stack to achieve network function.

Introduction of the ZStack protocol stack

ZStack protocol stack is developed by Norway semiconductor company Chipcon (have been acquired by TI), mainly used for the launch of CC2430, CC2431 and CC2530 ZigBee chip solution. ZStack protocol stack use IAR Embedded Workbench for MCS-51 of Swedish company as the software developing platform. ZStack protocol stack for the user is no entirely open source, its interior package security module, routing module and Mesh ad hoc-network module and so on, although it can not get the specific details of these module ,users can call the privided API function to realize these functions. The ZStack protocol stack designs a protocol stack scheduler called operating system abstraction layer (OSAL), as the name suggests, the abstract layer is mainly responsible for the completion of the overall scheduling of user custom application and protocol stack to the function implementation.

ZStack protocol stack ZigBee protocol divided into corresponding file group in accordance with the corresponding function, and each group is the program file realized by corresponding function. In the App group is mainly a user-defined program files, used to perform specific application; In the

HAL group, mainly including the hardware drivers of system, including the underlying hardware driver of LCD display, keyboard and so on, the user should modify this code according to their actual hardware; MAC group and ZMac group together to form the functions of MAC layer in ZigBee protocol; OSAL group is a ZStack implementation code of execution of task scheduling, including the release of memory allocation, message queue management, clock management, power management and other functions; Proflie group mainly realizes the application framework includes application layer wireless data transceiver, Registered endpoint descriptor functions; Security Group is used to achieve the safe operation of the network; Services group mainly compare and set operations of equipment short address; Tools group contain the composition and distribution of chip memory, setting operations of operation coordinator, router and terminal equipment; Group ZDO contain a variety of operation of device object of ZigBee; ZMain group contain the main function of the whole ZStack protocol stack operation. It is precisely because the TI divided ZStack protocol stack into different functional groups, it can efficiently realize the function of ZigBee network.

Design and practice of wireless network

Software design of key. Design and implementation of self starting network coordinator, let the node device automatically join network.After the establishment of the two wireless communication, data transmission mainly adopts two ways, a periodical for transmitting timing information; Another need to trigger the Flash information is transmitted through the key event, this design adopts the method of cycle timing transmission of information. The following is the key part of the design:

① The Periodic message is opened by the system timer and timing broadcast to group1 out, so in the SampleApp_ProcessEvent event handler in the following the timer code:

```
case ZDO_STATE_CHANGE:
    SampleApp_NwkState==(devastates_t)(MSGpkt->hdr.status);
    if((SampleApp_NwkState==DEV_ZB_COORD)
    ||(SampleApp_NwkState==DEV_ROUTER)
    ||(SampleApp_NwkState==DEV_END_DEVICE))
{
//Start sending the periodic message in a regular interval.
osal_start_timerEx(SampleApp_TaskID,
                              SAMPLEAPP_SEND_PERIODIC_MSG_EVT,
                              SAMPLEAPP_SEND_PERIODIC_MSG_TIMEOUT);
}
else
{
//Device is no longer in the network
}
break;
```

② When the device is added to the network after the state will change, event triggered ZDO_STATE_CHANGE for all tasks, a timer on. When the timing time, triggering events trigger event broadcast Periodic messages, SAMPLEAPP_SEND_PERIODIC_MSG_EVT, the corresponding task for SampleApp_TaskID, then again call SampleApp_ProcessEvent () to handle the SAMPLEAPP_SEND_PERIODIC_MSG_EVT event, the event handling function call SampleApp_SendPeriodicMessage () to send periodic information.

```
if (events & SAPLEAPP_SEND_PERIODI_MSG_EVT)
{
    SampleApp_SendPeriodicMessage();
    //setup to send message again in normal period(+a little jitter)
    Osal_start_timerEx(SampleApp_TaskID,SAMPLEAPP_SEND_PERIODIC_MSG_EVT,(S
AMPLEAPP_SEND_PERIODIC_MSG_TIMEOUT+(osal_rand() & 0X00FF)));
    //return unprocessed events
```

```
        return(events ∧ SAMPLEAPP_SEND_PERIODIC_MSG_EVT);
    }
```

The definition of the relevant processing function.

① The serial port function declaration

```
void UartInit(void);
void Uart_Baud_rate(int Baud_rate);
void UartTX_Send_String(char * Data,int len);
```

② The serial port initialization function invocation

```
#if defined(LCD_SUPPORTED)
    HalLcdWriteString( " SampleApp " ,HAL_LCD_LINE_1);
#endif
    UartInit();
    Uatt_Baud_rate(384);
```

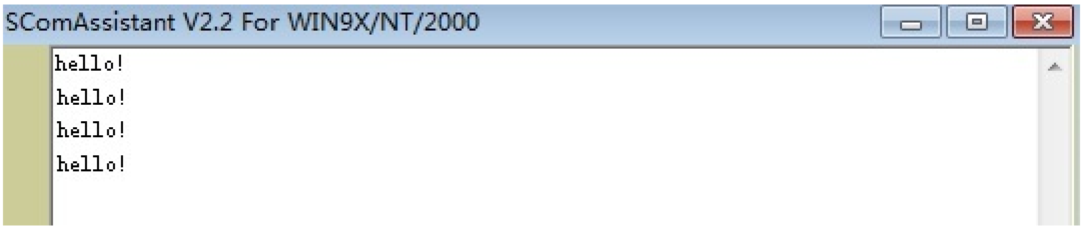
③ Add send data content is "hello!" function

```
    void SampleApp_SendPeriodicMessage(void)
    {
        uint8 buffer[11];
        buffer[0] = 0X00;
        buffer[1] = LO_UINT16(1000);
        buffer[2] = HI_UINT16(1000);
        buffer[3] = ′ h′ ；
         ……..
        if
(AF_DateRequest(&SampleApp_Periodic_DstAddr,&SampleApp_epDesc,SAMPLEAPP_PERIO
DIC_CLUSTERID,10,BUFFER,&sAMPLEaPP_TransID,AF_DISCV_ROUTE,AF_DEFAULT_R
ADIUS)==afStatus_SUCCESS)
    {
    }
    else
    {
    //Error occurred in request to send.
    }
    }
```

④ Periodic message processing function

```
    void SampleApp_MessageMSGCB(afIncomingMSGPacket_t * pkt)
    {
        uint16 flashTime;
        uint8 * pointer;
        uint8 test = 0x30;
        switch(pkt -> clusterId)
        {
        case SAMPLEAPP_PERIODIC_CLUSTERID:
                flashTime = 1000;
                HalLedBlink(HAL_LED_1,4,50,(flashTime/ 4));
                Pointer=&(pkt->cmd.Data[3]);
                UartTX_Send_String(pointer,7);
                break;
        case SAMPLEAPP_FLASH_CLUSTERID:
                flashTime=BUILD_UINT16(pkt->cmd.Data[1],pkt->cmd.Data[2]);
                HalLedBlink(HAL_LED_4,4,50,(flashTime/ 4));
                break;
        }
    }
```

Program debugging. The first to start the coordinator module, the establishment of the network after the success of LED2 light, and then start the node terminal ZigBee module, network module after the success of the LED2 light. Then node end will cycle sending broadcast message "hello!",And the LED1 flashes 4 times, the coordinator receives a message also flashes 4 times LEDl,to the serial port to send and receive data over the " hello!". At this time, such as the PC port line is connected to the serial port on the platform of RS-232-2 DB9, use the CCD_SETKEY button to choose coordinator module, open the PC serial port terminal software, set the baud rate to 38400, to receive the coordinator sends a data "hello!", As shown in Fig. 1:



```
hello!
hello!
hello!
hello!
```

Fig. 1 program debugging results

Summary

ZStack protocol stack provides a powerful function, which makes most of the application design apply and develop based on the existing open protocol. The key to learn ZStack protocol is clearing structure of ZStack,and task management methods based on polling and priority control.This paper briefly describes software framework of the Zstack protocol stack, and realizes wireless communication in the related experimental platform.

Acknowledgement

This work is supported by the Natural Science Foundation of Hainan Province in China (Research on rubber forest environment monitoring system based on the Internet of things No:614238);Science Research Project of Colleges and Universities of Hainan Province in China (Research and implementation of network security mechanism based 6LoWPAN No: Hnky2015-79) and (Research on mobile banking security access based on SSLVPN No: Hnky2015-78).

References

[1] Casilari E, Cano-García J M, Campos-Garrido G. Modeling of Current Consumption in 802.15.4/ZigBee Sensor Motes. sensors, 2010,10(6):5443-5468 .

[2] Nguyen T. Low-Power Direct Conversion Transceiver for 915 MHz Band IEEE 802.15.4b Standard Based on 0.18 um CMOS Technology. ETRI Journal, 2008,30(2):33-46.

[3] Sung T, Yang C. An adaptive joining mechanism for improving the connection ratio of ZigBee wireless sensor networks. INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS, 2010,23(2):231-251.

[4] IEEE Std. 802.15.4 Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specification for Low-Rate Wireless Personal Area Networks. 2003(5)

[5] Sung W, Tsai M. Multi-Sensor Wireless Signal Aggregation for Environmental Monitoring System via Multi-bit Data Fusion. Applied Mathematics & Information Sciences,2011,5(3):589-603.