http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (17): 4068-4072, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.4068.4072

A Design Method of Universal Wireless Sensor Networks Node

[1]Xu Lihua, [2]Wang Yihuai and [2]Li Cuixia
[1]School of Computer Engineering, Suzhou Vocational University, Suzhou, 215104, China
[2]School of Computer Science and Technology, Soochow University, Suzhou, 215006, China

Abstract: It is a new design method of embedded system to design hardware and software based on component during the development process of embedded system. The method can improve the reusability and portability of embedded system significantly. For the extensive application of wireless sensor networks, the design method of a universal wireless sensor networks node, with which hardware can be cut and software can be configured, is proposed in this study. According to the design method of component-based, the hierarchical design model of the software is improved. The hardware component design and software component design is discussed and the development details of bottom software component is given. It is proved that the node can be applied quickly, that improves the efficiency of wireless sensor networks' application development, because of the hardware can easily be extended and the software with clearly interface.

Key words: Embedded system, wireless sensor networks, component, internet of things, reusability

INTRODUCTION

Wireless Sensor Networks (WSN), which is a key technology of the perceptual layer in system structure of internet of things (IOT) (Zheng *et al*., 2011), become a research hotspot of embedded system applications in recent years. Application development in IOT is very complex (Ma, 2011), because it contains many technical fields, which involves WSN technology and specific application areas. So it is very necessary to provide a convenient, efficient, low-cost and universal module of WSN nodes for application engineers of IOT. With the module, engineers can complete the application development quickly, which improves the efficiency of development and popularize the application of WSN.

With the development of embedded technology, hardware design in embedded system becomes more and more complexly, software code increases rapidly, the design scale and development method change radically. Un-reusable hardware and un-portable software of traditional design method in embedded system are serious problems (Jozwiak *et al*., 2010), which increase the system development cycle and reduce the development efficiency. And that increase the difficulty of maintaining the system in the later stages. It is inevitable in embedded systems to increase the reusability and portability of hardware (Hasan and Bird, 2011) and software. The concept of component (Lau and Wang, 2007) comes from software engineering. The most important feature of component is reusability. Introduce the component-based design method, which commonly used in software, into the design process of embedded systems, particularly hardware design, can cut down the system development cycle and improve efficiency.

According to the method of hardware and software co-design in embedded system (Wiangtong *et al*., 2005), a universal module of WSN nodes is proposed with hardware components based on MC13213, which is the new single-chip Zigbee platform (Wu *et al*., 2010). The design is based on module and component, which means it have general functions, not for specific projects or fields. The node can be applied quickly to the secondary development for it can be extended easily.

COMPONET-BASEDEMBEDDEDSYSTEMDESIGN

Embedded system can be divided into software system and hardware system. Component-based embedded system design includes the software component design and the hardware component (Arato *et al*., 2005) design.

Embedded Software Component (ESC) (Lind and Heldal, 2012) is a set of standardized, packaged, portable, reusable software components with embedded characteristic elements. According to the relevant with hardware, embedded software component can be divided into high-level software component and the bottom software component.

Based on the component require for encapsulation, descriptive, portability and reusability (Qureshi and Hussain, 2008), Embedded Hardware Component (EHC) is defined as one or more hardware modules, with

Corresponding Author: Xu Lihua, School of Computer Engineering, Suzhou Vocational University, Suzhou, 215104, China

supporting circuit and functional description encapsulated into a reusable hardware entity and a range of standardized input/output interface provided. Users can use the hardware component in different application systems without the internal details such as the specific circuit, for hardware component itself is an entity.

DESIGN AND ANALYSIS OF THE UNIVERSAL WSN NODE

Design ideas of the universal WSN node are shown in Fig. 1. The hardware consists of the core component layer and the intermediate component layer. The core component layer includes core processing unit and its minimum system. The intermediate component layer consists of input/output(I/O) function component and communication component.

In accordance with component-based, modular and hierarchical design ideas, the software can be divided into three layers, from top to bottom are: the application profile layer, wireless routing protocol layer and embedded software component layer.

Application profile level, which faces to the user's specific application, can implement the configuration of various functional components and the communication component and complete assembling and disassembling of the user data. Wireless routing protocol layer provides routing services for the application profile level, with its realization of the data store and forward of specific WSN. Embedded software component layer includes high-level software component and the bottom software component. The bottom software component of embedded software component layer implements the drivers of the various hardware components. High-level software component calls related services through the bottom software component interfaces.

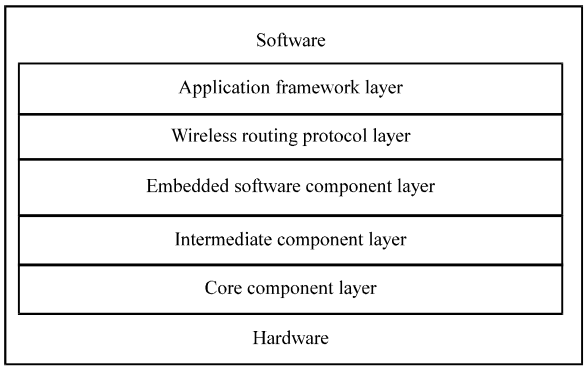

Fig. 1: Structure of universal WSN node

HARDWARE COMPONENT DESIGN

According to the component-based ideas, the hardware of the Universal WSN Node is composed by the minimum system component, the data communication component, the analog input component and output component and the switch input component and output component. The minimum system component is also called core component. The data communication component is in charge of communicating with external devices. The analog input component and output component interact with the analog signals. And the switch input component and output component interact with the switch signals. Hardware component of the universal module of WSN nodes is shown in Fig. 2.

MC13213 is used as the control chip of the core processing unit. This is a dedicated chip for WSN, with internal integration of the 8-bit microcontroller MC9S08GB60 and low power 2.4GHz radio frequency transceiver MC13192. It has the features of small, compact design, high noise immunity and works stability. MC13213 is designed with a rich interface reserved, including the analog input interface, the analog output interface, the switch input interface and the switch output interface.

The minimum system component consists of core processing unit, write and debug component and power component, which is composed by power circuit, crystal oscillator circuit, filter circuit and reset circuit. That supplies the basis work conditions for the core processing unit. The data communication component, which includes serial communication component and wireless communication component, is designed to charge with the data exchange to outside. The serial communication component is realized by serial communication interface module and the wireless communication component is realized by wireless communication module MC13192.

Generality is a key consideration in the design of input/output function component. After collected by sensor, analog signal in physical environment must be converted into a digital signal before processing by core

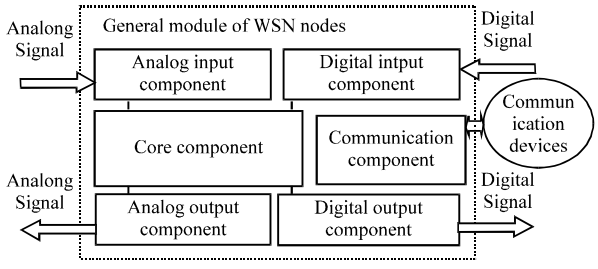

Fig. 2: Hardware component of the universal WSN node

Table 1: Resource list of the hardware component

Function	Component Name	Directions
Minimum system	Power Component	Power Circuit, Crystal Oscillator Circuit, Filter Circuit, Reset Circuit
	Write and Debug Component	BDM Write and Debug Interface
Data Communiction	Serial Communication Component	Serial Communication
	Wireless Communication Component	MC13192 Wireless Communication
Basic I/O of MC13213	Analog Input Component	8-channal A/D
	Analog Output Component	4-chaunal PWM
	Digital Input Component	8-channal GPIO
	Analog Output Component	8-channal GPIO

unit. Analog input component is responsible for implementing this process, with the internal A/D module in MC13213. The design of the component includes the sensor and the external signal conversion circuits. The core processing unit usually transfers the analog signal through the PWM module. So the design of analog output component is focused on corresponding driver conversion circuits. The switch signal can be processed directly by the core unit without conversion. But the input signal needs to isolate, otherwise it will interfere with the system. The design of output signal should also consider the amplification and interference. That is the design content of switch input component and output component.

Resource list of the hardware components is shown in Table 1.

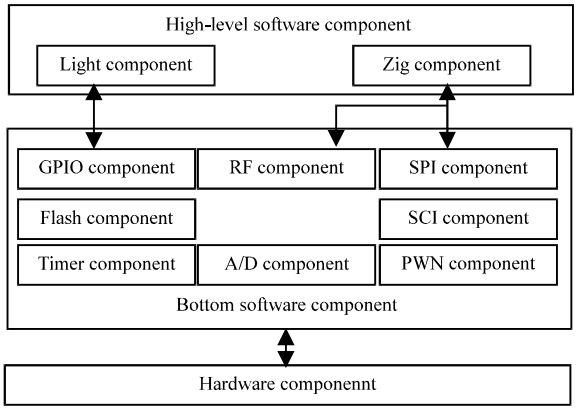

Fig. 3: Embedded software component of the universal WSN node

SOFTWARE COMPONENT DESIGN

Embedded software component of the universal WSN node: As mentioned above, the module's software is designed into three layers. The design of embedded bottom software components, on which the hardware components closely related, will be discussed in detail in this study, while only a brief description will be given for the high-level component, the wireless routing protocol layer and the application profile layer.

The bottom software component of the universal module, based on hardware component, is composed by general purpose input/output(GPIO) component, Flash component, Analog/Digital(A/D) component, Pulse Width Modulation(PWM) component, Serial Peripheral Interface(SPI) component, Timer component, Serial Communicate Interface(SCI) component and Radio Frequency(RF) component. The high-level software component is composed by Light component and wireless transmission component(Zig Component). The design of embedded software component of the universal module is shown in Fig. 3.

Design of bottom software component: According to the encapsulation method based on component, each bottom software component is designed in two files: function implementation file"*.c" and header file"*.h". The detailed design of the bottom software component of the module is shown in Table 2.

The reusability of the bottom software components and its using method is given by the example of RF component, which is in charge of wireless communication. The two files corresponded by RF component are source file "RF.c" and head file "RF.h". The source file "RF.c" involves the implementation code of all abstract performance functions, which are hardware independent. In file "RF.c", include RF_Init(), RF_Set Channel(), RF_PA Output Adjust(), RF_Send Data(), RF_Send Data By CSMACAv(), RF_Recv Data(), RF_Energy Detect(), RF_Link Quality() and RF_IrqInt Proc().

The head file "RF.h" explains the function description of component, the cautions when using component, other head files will be referenced, macro definitions of related hardware parameters and prototype declaration of the function defined in "RF.c". When the component is reused or transplanted, by modifying the related parameters in head file "RF.h", without change the code in file "RF.c", the bottom software component's reuse and the system's reorganization can be realized. When developing the WSN applications by RF component, the two file "RF.c" and "RF.h" should be

Table 2: Function list of the bottom software component

Component name	"*.h" File	"*.c" File	Function to External Interface
GPIO component	GPIO.h	GPIO.c	GPIO_Init() ,GPIO_Get() ,GPIO_Set()
Flash component	Flash.h	Flash.c	Flash_Init(), Flash_ErasePage(), Flash_Write ()
A/D component	AD.h	AD.c	AD _Init(),AD _ Ave ()
PWM component	PWM.h	PWM.c	PWM_Set (),PWM_Disable (), PWM_Enable ()
SPI component	SPI.h	SPI.c	SPI_Init (), SPI_SendOneByte(), SPI_ReceiveOneByte ()
Timer component	Timer.h	Timer.c	Timer_Init()
SCI component	SCI.h	SCI.c	SCI_Init(), SCI_Send1(), SCI_Re1 (), SCI_SendN(), SCI_ReN()
RF component	RF.h	RF.c	RF_Init(), RF_SetChannel(), RF_PAOutputAdjust(), RF_SendData(), RF_ RecvData(), RF_SendDataByCSMACAv(), RF_ EnergyDetect(), RF_LinkQuality() , RF_ IrqIntProc ()

loaded first and then only parameters related with hardware in "RF.h" need to be modified, after that the program of wireless communication is completed.

High-level software component and the upper layer software: As shown in Figure 3, the high-level software component of the module includes Light component and Zig component. Light component is designed to control the state of the lights by calling GPIO component. Zig component is designed to complete the wireless communication by calling SPI component and RF component.

Something about the wireless routing protocol layer. We propose a simple wireless routing protocol for WSN (WSN-SDNET), which is simple and clear, very suitable for fixed nodes in WSN applications. The software of the wireless routing protocol layer provide appropriate interface for application profile layer to transfer data orderly.

Something about the application profile layer of WSN. Application profile layer, which is based on wireless routing protocol layer, provides services to the user application layer to achieve the configuration on nodes by various parameters. It includes the basic functional component configuration, the data communication component configuration and the network identification configuration.

OTHERS

We have used the universal WSN node during developing the project of wireless lighting brilliance modulation system and intelligent building lighting control system, etc. With silicon controlled rectifier (SCR), the digital output component of MC13213 can control the lightness. And the analog input component is in charge of collecting the lightness. So the alternating current is converted to analog signal to be processed by MC13213. The lightness can be detected and controlled wirelessly. Figure 4 shows the real product of the universal WSN node.

Fig. 4: Real product of the universal WSN node

CONCLUSION

Whether software or hardware, it has become a trend to design based on component, which can improve the embedded system reusability and portability. This study discusses the design method of a universal WSN node based on component. The design based on hardware component realizes the reusability of hardware, which can be easily extended. And the design based on software component realizes the transparent transmission of user's data, which is convenient for user to customize. Testing and tracking on related projects shows that the module has a good application prospects, with fully function and stable performance.

REFERENCES

Arato, P., Z. Adam Mann and A. Orban, 2005. Extending component-based design with hardware components. Sci. Comput. Programming, 56: 23-39.

Hasan, M.Z. and M. Bird, 2011. Energy reductions for embedded processors in reconfigurable hardware. Proceedings of the IEEE International Conference on Electro/Information Technology, May 15-17, 2011, Mankato, MN., USA., pp: 1-8.

Jozwiak, L., N. Nedjah and M. Figueroa, 2010. Modern development methods and tools for embedded reconfigurable systems: A survey. Integr. VLSI J., 43: 1-33.

Lind, K. and R. Heldal, 2012. A practical approach to size estimation of embedded software components. IEEE Trans. Software Eng., 38: 933-1007.

Lau, K.K. and Z. Wang, 2007. Software component models. IEEE Trans. Software Eng., 33: 709-724.

Ma, H.D., 2011. Internet of things: Objectives and scientific challenges. J. Comput. Sci. Technol., 26: 919-924.

Qureshi, M.R.J. and S.A. Hussain, 2008. A reusable software component-based development process model. Adv. Eng. Software, 39: 88-94.

Wiangtong, T., P.Y. Cheung and W. Luk, 2005. Hardware/software codesign: A systematic approach targeting data-intensive applications. IEEE Signal Process. Mag., 22: 14-22.

Wu, J., Q.Y. Pang and Y.H. Wang, 2010. Research and implementation of physical layer protocol basing on the single-chip ZigBee platform of MC13213. Proceedings of the International Conference on Computer Application and System Modeling, Volume 3, October 22-24, 2010, Taiyuan, pp: V3-164-V3-169.

Zheng, J., D. Simplot-Ryl, C. Bisdikian and T.H. Mouftah, 2011. The internet of things. IEEE Commun. Mag., 49: 30-31.