國立交通大學

電機與控制工程學系

碩士論文

研發無線感測網路用於居家健康照護

Development of Wireless Sensor Network for Home Health-Care Application

研 究 生：林俊良

指導教授：陳右穎　博士

中華民國九十五年六月

研發無線感測網路用於居家健康照護

Development of Wireless Sensor Network for Home Health-Care Application

研究生：林俊良　　Student：Jun-Liang Lin

指導教授：陳右穎　　Advisor：Dr. You-Yin Chen

國立交通大學

電機與控制工程系

碩士論文

A Thesis
Submitted to Department of Electrical and Control Engineering
College of Electrical Engineering
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in
Electrical and Control Engineering
June 2006
Hsinchu, Taiwan, Republic of China

中華民國九十五年六月

博碩士論文授權書

本授權書所授權之論文為本人在__________________大學(學院)____________系所
______________組____________學年度第______學期取得______士學位之論文。

論文名稱：____

指導教授：____

1.□同意　　□不同意

本人具有著作財產權之上列論文全文(含摘要)資料，授予行政院國家科學委員會科學技術資料中心(或改制後之機構)，得不限地域、時間與次數以微縮、光碟或數位化等各種方式重製後散布發行或上載網路。

本論文為本人向經濟部智慧財產局申請專利(未申請者本條款請不予理會)的附件之一，申請文號為:______________，註明文號者請將全文資料延後半年再公開。

2.□同意　　□不同意

本人具有著作財產權之上列論文全文(含摘要)資料，授予教育部指定送繳之圖書館及國立交通大學圖書館，基於推動讀者間「資源共享、互惠合作」之理念，與回饋社會及學術研究之目的，教育部指定送繳之圖書館及國立交通大學圖書館得以紙本收錄、重製與利用；於著作權法合理使用範圍內，不限地域與時間，讀者得進行閱覽或列印。

本論文為本人向經濟部智慧財產局申請專利(未申請者本條款請不予理會)的附件之一，申請文號為:______________，註明文號者請將全文資料延後半年再公開。

3.□同意　　□不同意

本人具有著作財產權之上列論文全文(含摘要)，授予國立交通大學與台灣聯合大學系統圖書館，基於推動讀者間「資源共享、互惠合作」之理念，與回饋社會及學術研究之目的，國立交通大學圖書館及台灣聯合大學系統圖書館得不限地域、時間與次數，以微縮、光碟或其他各種數位化方式將上列論文重製，並得將數位化之上列論文及論文電子檔以上載網路方式，於著作權法合理使用範圍內，讀者得進行線上檢索、閱覽、下載或列印。

論文全文上載網路公開之範圍及時間 –

本校及台灣聯合大學系統區域網路：　年　月　日公開

校外網際網路：　年　月　日公開

上述授權內容均無須訂立讓與及授權契約書。依本授權之發行權為非專屬性發行權利。依本授權所為之收錄、重製、發行及學術研發利用均為無償。上述同意與不同意之欄位若未鉤選，本人同意視同授權。

研究生簽名：　　　　　　　　　　　　　　學號：

（親筆正楷）　　　　　　　　　　　　　　（務必填寫）

日期：民國　　年　　月　　日

1.本授權書請以黑筆撰寫並影印裝訂於書名頁之次頁。

國家圖書館博碩士論文電子檔案上網授權書

本授權書所授權之論文為本人在__________________大學(學院)____________系所
______________組____________學年度第______學期取得______士學位之論文。

論文名稱：____

指導教授：____

□同意　　□不同意

本人具有著作財產權之上列論文全文(含摘要)，以非專屬、無償授權國家圖書館，不限地域、時間與次數，以微縮、光碟或其他各種數位化方式將上列論文重製，並得將數位化之上列論文及論文電子檔以上載網路方式，提供讀者基於個人非營利性質之線上檢索、閱覽、下載或列印。

上述授權內容均無須訂立讓與及授權契約書。依本授權之發行權為非專屬性發行權利。依本授權所為之收錄、重製、發行及學術研發利用均為無償。上述同意與不同意之欄位若未鉤選，本人同意視同授權。

研究生簽名：　　　　　　　　　　　學號：
（親筆正楷）　　　　　　　　　　　（務必填寫）

日期：民國　　　年　　　月　　　日

---

1. 本授權書請以黑筆撰寫，並列印二份，其中一份影印裝訂於附錄三之一(博碩士論文授權書)之次頁；另一份於辦理離校時繳交給系所助理，由圖書館彙總寄交國家圖書館。

國立交通大學

論文口試委員會審定書

本校 電機與控制工程 學系碩士班 林俊良 君

所提論文 研發無線感測網路用於居家健康照護

合於碩士資格標準、業經本委員會評審認可。

口試委員：
曾煜棋
黃育綸
張淵仁
陳右穎

指導教授：陳右穎 教授

系 主 任：楊谷洋 教授

中 華 民 國 九十五 年 六 月二十四日

研發無線感測網路用於居家健康照護

學生：林俊良　　　　　　　　　　指導教授：陳右穎 博士

國立交通大學電機與控制工程學系 碩士班

摘　要

無線感測網路尚未廣泛地應用於居家健康照護服務上，大多數在健康照護相關方面的研究都只著重於點對點的傳輸，而沒有採取網路的架構。如此，將會使系統的擴充性受到限制。然而少數具有使用網路架構的研究大都採用藍芽的無線傳輸技術，藍芽是無線個人區域網路的標準之一，其 Piconet 網路最多只容許八個裝置加入，網路定址能力將受到限制。為了實現無線感測網路，使其具有高度網路定址能力，本研究透過 ZigBee 發展無線感測網路。ZigBee 為無線個人區域網路，其建構在 IEEE 802.15.4 的標準之上，具低功率消耗，高度網路定址能力的優點。

本研究所實現之居家健康照護系統是由多個無線生理信號感測器、路由器與個人區域網路仲裁器所組成的。開發此系統之硬體平台包含了一個微控制器 (MSP430F1611, TI, USA) 與 ZigBee 射頻晶片(UZ2400, UBEC, Taiwan)。本研究已完成 ZigBee 網路層之開發，並建構無線感測網路為一個樹狀三階層式之架構。透過無線生理感測器量測病人之心電圖與血氧濃度傳送至路由器；路由器負責分配每個無線感測器可擁有的傳送時間，使用階層式的路由方式將封包傳送給個人區域網路仲裁器。個人區域網路仲裁器接收路由器傳送之封包並取出生理信號資料，以 RS232 傳送至個人電腦。

居家健康照護上常伴隨大量生理信號的處理，由於無線感測網路在管理與控制會面臨棘手的問題，例如感測裝置如何做有效的管理及資料時效性的問題, 因此本研究採用 web-based 的管理方式，並結合 web 2.0 的概念, 使生理信號圖形能夠流暢地顯示於網頁上。其技術主要結合非同步 Java 描述語言、可延伸性標示語言、可縮放向量圖形與資料庫，具有跨平台、不受空間限制與整合性等優點。

本研究已建構完具有遠端監控與管理能力之無線感測網路系統，其優點為具有高定址能力、低功率消耗、快速反應時間。將本研究應用於居家健康照護系統上，可讓病人在家裡就可以透過此系統讓醫生即時監看居家病人的健康情形，但是在病人隱私權上面卻是相當有道德爭議，未來居家健康照護系統須加強資料安全以保護病人隱私。

關鍵字: 無線感測網路; ZigBee;心電圖;血氧濃度;可縮放向量圖形。

Development of Wireless Sensor Network for Home Health-Care Application

Student：Jun-Liang Lin Advisors：Dr. You-Yin Chen

Degree Program of Electrical and Control Engineering
National Chiao Tung University

ABSTRACT

The Wireless Sensor Network (WSN) is an underdeveloped area today in the health-care service industry. Many studies in health-care issues focus on implementation of peer-to-peer communication without network topology, which refers to the extension of system limits. Furthermore, most studies adopt Bluetooth, which is a wireless standard for the Wireless Personal Area Network (WPAN). Bluetooth has limited addressing capability; a Piconet of Bluetooth contains eights devices at most. For the above reasons this study uses ZigBee, a wireless standard based on IEEE 802.15.4, which is a WPAN with low power consumption and high addressing capability.

The home-care system proposed in this study consists of sensor devices, a router, Personal Area Network (PAN) coordinators and a remote database server. The sensor device, router and PAN coordinator are all developed on the ZigBee module, which consists of a micro-controller (MCU; MSP430F1611, TI, USA) and a ZigBee RF chip (UZ2400, UBEC, Taiwan). Furthermore, the ZigBee network (NWK) layer and application (APP) layer are built on MCU for a WSN with three-level topology. For the sensor devices, physiological signals of the electrocardiogram (ECG) and (Oxygen Saturation) SpO_2 are measured from patients and then transmitted to the router. The router is responsible for scheduling a sending time for each associated device and using hierarchical routing to forward data to a specific device. On the PAN coordinator side, data is extracted from received packages and then passed to the personal computer through RS232.

Due to the large number of sensor devices in the WSN, management and querying present a number of problems. In this study a web-based management method is used to manage physiological information in the WSN. To display physiological signals on the web interface smoothly and rapidly, the latest design of web 2.0, which contains Ajax (Asynchronous Java Script + XML) and Scalable Vector Graphing (SVG), is adopted. This is a high-performance web based technology that is cross-platform, has no limitations of space and integrates the WSN and database servers.

The WSN with web-based management has been implemented successfully with many advantages, including higher addressing capability, lower power consumption, and reduced waking time. The system proposed in this study has the potential to bring healthcare out of the hospital and into the patient's home, replace nurses with sensor devices, and collect sensitive information. This will create many ethical considerations, which must be dealt with in order to ensure that any changes made in the health-care industry are for the good of the patients.

Keywords: Wireless Sensor Network, ZigBee, Electrocardiogram, Oxygen Saturation, Scalable Vector Graph

誌　　謝

在碩士班的求學過程中，首先要感謝我的指導老師陳右穎博士，在我求學生涯中，老師除了在研究上對我細心指導，在待人處事上也給我許多的教導，使得我的研究不僅有所收穫，並且能夠順利完成本論文；此外還要感謝我的口試委員：曾煜棋、黃育綸與張淵仁教授，他們在口試中給我許多寶貴的意見，讓我成長茁壯。對於所有教導過我的所有師長們，讓我在兩年的學習的過程中獲益良多，我更是懷著感恩的心，謝謝您們的教導。

另外，我還要感謝台灣新竹的達盛電子提供我一個完善的實驗平台與諮詢，讓我能夠有優良的學習成果。對於達盛電子的簡嘉瑋經理、黃國彰經理、俊嘉、鈞泰以及所有達盛電子的員工們的熱心協助與關心，我充滿了無限的感激與謝意。更要感謝交通大學給我良好的學習與研究環境，讓我可以在這兩年中成長與茁壯。我還要感謝實驗室各位學長姐弟妹們，在實驗室裡的生活有了他們的幫忙以及扶持，使得我在這兩年能夠過得充實且多采多姿，感謝碩仁、信賢、毓廷學弟妹們的幫忙與支持，更要感謝修宏學長、家禎學姊的鼓勵與指導。對於同居的富全，在研究上彼此互相幫忙與協助，我充滿了感激的心。此外還要感謝所上助教們的幫忙和提供的設備資源，還有電控所 94 級的所有同學以及一些默默關心我的朋友們，在這天時地利人和的配合之下順利完成碩士學位。

最後，我要感謝我親愛的家人，我的父母與我的弟弟，尤其感謝我父母親從小到大的栽培，由於他們的支持、鼓勵、教誨以及無限的愛，才能讓我今天得以順利完成本論文而取得碩士學位，僅以本論文獻給我最親愛的家人。

Contents

List of Tables

List of Figures

Chapter 1 Introduction

The wireless sensor network is comprised of many sensor devices, each of which communicates and operates with its neighbors by wireless transmission. Each sensor device includes a microcontroller unit, a memory unit and an RF chip. The sensor device is responsible for measuring physiological signals, processing the measured data, and then sending the measured data to a specific device wirelessly. Since the sensor device with the characteristic of low power consumption is used to perform long-term monitoring, we must survey the current wireless technology and select the best candidate for low power consumption. According to our requirements, we have chosen the ZigBee wireless transmission standard [1], which is a wireless personal area network that creates a wireless sensor network of tree topology using hierarchy routing with web-based management for usage convenience.

1.1. Motivation

As the population of elderly people has increased in recent decades, the number of persons aged 60 is projected to be almost two billion by 2050 [2]. Thus heath related issues are becoming more and more important for this population. To increase medical quality, home health-care is a useful solution to achieve long-term monitoring of chronic disease effectively. The wiring of patients to obtain physiological signals has many drawbacks, including high cost of developments and maintenance. However, such drawbacks are lessened using wireless instead of wired connections to allow patients to move freely while carrying only a few small nodes. Now, in this study we propose a wireless sensor network solution that can support mobility and flexibility of sensor nodes in a network, thus providing many advantages while replacing wired with wireless logic links. Since advancement in wireless communication, sensor technology and information technology in general provides opportunities to in the area of home health-care service, which is very promising application to improve the quality of life. The home-care service usually provides real-life long term monitoring of a patient’s health that is useful for the assessment of treatments at home. This study focuses on the design and implementation of a wireless sensor network that has a tree topology of three levels using hierarchical routing with web-based management. The wireless sensor network has many advantages, including low power consumption and high addressing space. The wireless sensor network is capable of formatting a network automatically, accepting several sensor devices and collecting data from existing devices in the network.

Many studies [3] [4] [5] of home health-care issues use Bluetooth as the wireless personal area network. However, Bluetooth has many drawbacks, including limited addressing capability, higher power consumption than ZigBee and longer waking time than ZigBee.

The wireless sensor network can be used for collecting the necessary physiological signals of a patient, which are used to determine existing health problems and help predict such problems in the future. Because wireless sensor networks have a very large number of sensor nodes, their management and querying is a very troublesome problem. This study proposes a solution to this, which is to implement web-base management. This has the advantages of flexibility, ease of development and convenience of access. To determine existing health problems and help predict such problems in the future, we must measure the physiological signal of a patient. This study first implements an electrocardiogram (ECG) senor, since the ECG provides signals to monitor the most vital biological processes of the patient. Secondly, we select a commercial SpO_2 sensor that provides oximeter information containing SpO_2 and the heart rate.

1.2. Review of the Literature

Recently, in health-care related research Tura et al. [4] have focused on the implementation of a network for managing home-care activities. The device of this network can measure blood oxygen saturation, the heart rate, the respiration rate and the patient's quantity of movement. Measured data are stored into a multimedia card and then transmitted to a PC through Bluetooth. At the PC end the measured data is finally transmitted to the server center through the internet. The device uses a 16-bit microcontroller of the Mitsubishi M16C/62 family. There are many reasons for choosing this micro-controller, including compact design, smaller board dimensions and lower power consumption. For sensors they use the OEM pulse oximeter board, which is the NONIN OEM II module, to measure plenty of physiological signals of blood oxygen saturation and heart rate. The respiration rate is measured using the Respiratory Effort Monitoring System (RESP-EZ) from the EPM systems module. The patient's quantity of movement is measured through a dual axis thermal accelerometer called MEMSIC MXD205.

In terms of the hardware design for the wireless sensor, some papers [6] [7] have focused on integrating the hardware and communication interface such that each sensor device can communicate with other sensor devices and enter various physiological values into a database.

For network topology some studies have focused on the constructing suitable network topology for physiological signal transmission. One study focused on the implementation of wireless sensors network using Bluetooth, which consists of sensor nodes, relay nodes and a control node. It allowed all nodes to communicate with each other with a Bluetooth module, sensor and relay nodes detecting certain events and reporting the events to the control node. Its network is a tree topology based on Piconet where each Piconet has one master and up to seven slaves.

Korhonen et al. [8] proposed methods to implement health monitoring in homes of the future home. These methods included models of remote monitoring, comparing wearable sensors with environmental sensors, system architecture that contained sensor devices, a personal gateway or home gateway that was a mobile device, such as mobile phone used to transfer data from a sensor device to a remote database through the internet. In this study the authors presented some useful models for health monitoring and discussed the technical requirements for the health monitoring system based on wearable and ambient sensors, which measured health related data in the daily environment of the patient. To make these methods more affordable, health-monitoring needed to support the use of the generic platforms such as Bluetooth or ZigBee. The gateway used generic hardware such as a mobile phone or Personal Digital Assistant (PDA) as a personal gateway or a Digital TV as a home gateway.

Since the wireless sensor network is comprised of a very large number of sensor nodes, their management and querying of the wireless sensor network was a troublesome problem. Web-based management was useful solution for management and querying of the wireless sensor network. Hwang et al. [9] implemented a wireless sensor gateway that was responsible for processing a user's query from a web page and transferring the user's query into a sensor query. Finally the wireless sensor gateway sent the sensor query to the wireless sensor network that was generated by a sensor network emulator.

Tan et al. [3] focused on the design and implementation of a home health-care system using Bluetooth, web service and Global System for Mobile communication (GSM) short message service. In this study the authors implemented a mobile ECG unit that comprised a sensor, transmitter and receiver. The ECG physiological signal was passed to a remote database of the health web server unit based on the SOAP for diagnosis through the internet. This study used the web service to enhance the flexibility of architecture and convenience of usage, which improved medical quality and reduced medical costs.

A system called Telemedicine [10] has been developed to improve the quality of home-based care and medical treatment. The system supports 24-hour real-time monitoring in a patient's home by reliable sensors, which are responsible for measuring the physiological signals of the patient and sending this data to the patient's computer through wireless communication. The computer analyzes and stores the data.

For medical care, a project called CodeBlue [10] was developed by the division of engineering and applied science at Harvard University. The project explored the application of a wireless sensor network to a range of medical applications including pre-hospital, in-hospital emergency care, disaster response and stroke patient rehabilitation. The project integrated medical sensors with lower power wireless networks, which used wireless ad-hoc routing protocols for critical care, security, robustness and prioritization. The hardware architecture of the project focused on many aspects, including ultra-lower-power sensing, computation and communication. The project also implemented 3D location tracking using radio signal information and adaptive resource management, congestion control and bandwidth allocation in the wireless sensor network.

Another project [10] has been initiated to integrate body sensors with the internet. Sensors were attached to the body of a patient and connected with other sensor nodes through wireless transmission. The wireless body sensor network comprised several body sensor units and one body control unit. The body control unit was responsible for collecting data from all body sensor units and sending this data to the internet. Body sensor units were responsible for measuring the necessary physiological signal of patients. This data could be used for disease diagnosis or disease prediction. The project was suitable for hospital applications, home care services, healthcare centers and sports medicine.

1.3. Objective

Home health-care is a useful solution to improve medical quality by using wireless technology to transmit physiological signals to a hospital. The hospital uses this data to do long-term monitoring. Doctors can use this data to make diagnoses or predict future diseases. To implement the home health-care system for multiple users, the wireless sensor network is need. The wireless sensor network is comprised of many sensor nodes, and has lower power consumption, high addressing capability and low transmission latency.

1.4. Thesis Organization

This study is organized as follows. Chapter 2 briefly introduces the rationale of ECG and SpO_2 and describes implementation methods for ECG sensor and SpO_2 sensors. Chapter 3 illustrates the implementation methods of hardware and firmware for the wireless sensor network. Chapter 4 describes the design method of web-based management. In Chapter 5 we show experimental results of the whole system. These experimental results include the wireless sensor network and the web-based management system. Chapter 6 compares the wireless sensor network in this study with existing research, and describes related advanced applications. Chapter 7, the final chapter of this study, provides some conclusions and deals with another great challenge of this area of research. We also provide directions of future research for the extended applications.

Chapter 2 Sensors Design

To measure physiological signals of patients this study used two sensors: ECG and SpO_2 sensors. In this Chapter the ECG sensor and SpO_2 sensor will be described. The basic principles and circuit designs of the two sensors will also be illustrated.

2.1. Overview of the System Design

This study develops a wireless senor network system with web-based management, which is a tree topology of three levels with hierarchical routing. The wireless sensor network is comprised of a Personal Area Network (PAN) coordinator, routers and end devices. In the wireless personal area network, there are two types of wireless technology, Bluetooth and ZigBee. Compared to Bluetooth, ZigBee has lower power consumption, smaller size of protocol stacks and higher addressing capability. This study uses ZigBee wireless technology to develop the wireless sensor network.

The end device of the wireless sensor network includes the ECG sensor and the SpO_2 sensor. A micro-controller unit with a ZigBee RF chip selects one of the routers, which is the shortest distance between routers and the end device. When the device had joined a router, the device remains sleeping until it receives a start command from the router. To reduce power consumption, the router sends a start command to the associated devices every 15 seconds, alternately, according to its neighbor table. When the device has received a start command, it starts conversion of ADC and sends the results of the conversion to the router. When the router has received data from the associated device, it sends data to the associated PAN coordinator immediately. The PAN coordinator sends it to the UART that was connected with the UART of a personal computer (PC).

Since the wireless sensor network has many sensor devices, it is very difficult to manage and control. To solve this problem we use a remote database as the interface between the wireless sensor network and a web page. The system architecture is flexible. Users can monitor and control the wireless sensor network easily via the internet.

To achieve this goal we develop a PC application that is responsible for receiving data from

UART and then storing it to a remote database by an open database connectivity application interface (ODBC API).

Finally, this study develops a web page, which is comprised of XML, PHP, java script and Scalable Vector Graphs (SVG) to show the physiological curve. Doctors can access and manage the wireless sensor network conveniently via the internet. The system structure is shown in Fig. 2-1, which includes 3 patients in a net. The user (professional medical personnel) can monitor the ECG or SpO_2 signals via the internet.

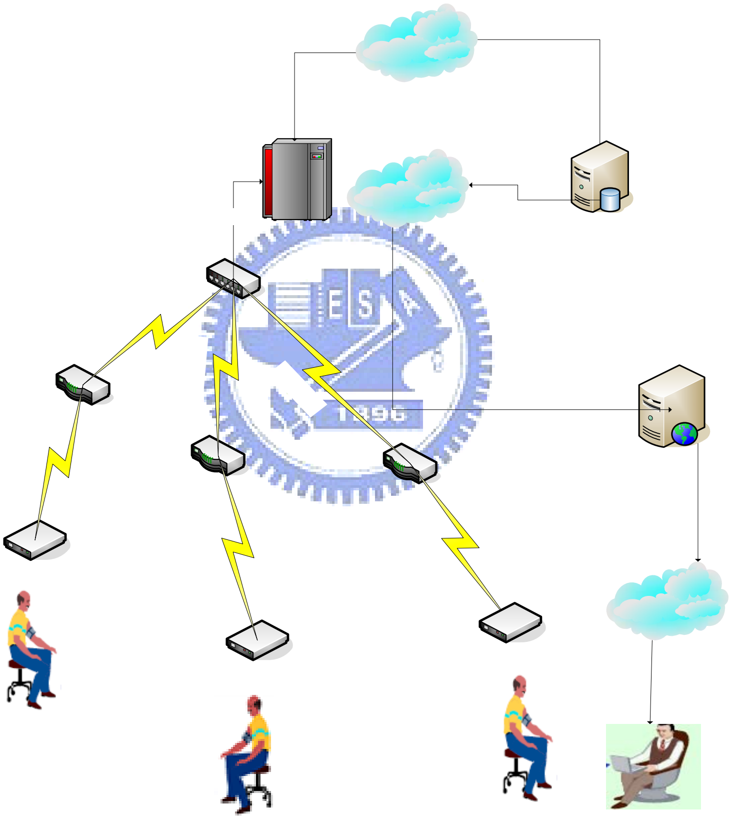


Fig. 2-1 Structure of the whole wireless sensor network with web-based management

2.2. Sensors

The sensors are responsible for providing an interface between physical signals and the digital world. In the case of home-care, the sensor is used to measure physiological signals. This study uses two sensors: ECG and SpO_2 sensors, which will be discussed below.

2.2.1. ECG Sensor

The measurement of the ECG involves the connection of between twelve and fifteen leads to a patient's chest, arms and right leg via adhesive foam pads. It records a short sampling of the heart's electrical activity between different pairs of electrodes. This study develops a sensor board that provided continuous ECG monitoring by measuring the differential across a single pair of electrodes.

2.2.1.1. ECG

The ECG measures the electrical activity of heart. The beating heart generates an electric signal that can be used as a diagnostic tool for examining the functions of the heart. This electric activity of the heart can be approximately represented as a vector quantity. Cardiologists have developed a simple model to represent the electric activity of the heart. In this model, the heart consists of an electric dipole located in the partially conducting medium of the thorax. This dipole moment, knew as the cardiac vector, is shown in Fig. 2-2

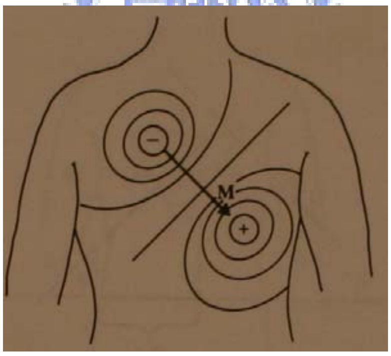


Fig. 2-2 Cardiac vector diagram [11]

The cardiac vector is defined as including 12 leads to form the exact ECG. However, in our case this is a portable device that cannot employ the whole 12 leads in our sensor, so we choose only lead II (shown in Fig. 2-3), which is considered to be a typical example of ECG monitoring. The lead II ECG waveform is also considered to provide typical clinical data for diagnosing heart disease. In fact, hospitals widely use the portable lead II ECG machine in the emergency ward.

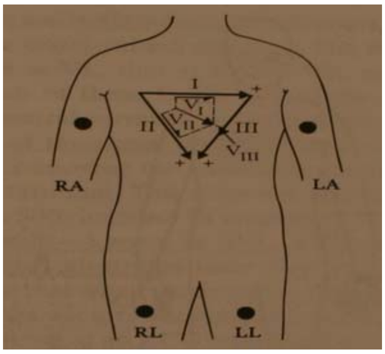


Fig. 2-3 Lead II ECG diagram [11]

2.2.1.2. ECG Waveform

There are two premises in the ECG waveform. One is that the cardiac muscle is formed by excitable nerves that express electrical signals (voltage). The electrical signals are termed the ‘action potential’ of the cardiac nerve cells. The other premise is that the cardiac muscle (the atrium and the ventricle) cells systole and diastole together, which produces every beat of the heart. If they do not systole and diastole together, the heart beat is irregular -- so called cardiac fibrillation -- and the person die.

The explanation of the ECG waveform is that electric stimulation is activated by the SA node; the SA node expresses an action potential, as shown in the first curve of Fig. 2-4. Then the *AV* node receives the stimulation and expresses its action potential, which is shown on the third curve of Fig. 2-4. Then the signal (electric stimulation by the SA node) keeps conducting to the atrial muscle and then the ventricular muscle.

Cardiologists have determined that the *P* wave of the ECG waveform is mainly contributed by the atrial muscle, as shown on the second curve of Fig. 2-4. The *QRS* complex and the *T* wave are mainly contributed by the ventricular muscle, seen in the last curve of Fig. 2-4. Because the cardiac muscle is formed by excitable nerves, the electrical stimulus from the SA node is propagated to the other part of the heart. According to the difference between propagation time and action potential of every part of the heart, the ECG wave form is decomposed to the *P* wave, *QRS* complex, and *T* wave. Every component of the ECG waveform is shown in Fig. 2-4. Note that the action potential of bundle branches lags behind the action potential of the *AV* node by 100 ms., so it takes 100 ms. to pump blood from the ventricle to the atrium.

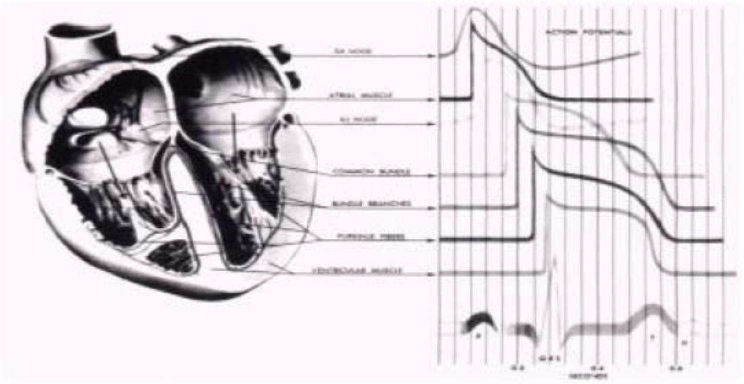


Fig. 2-4 Components of ECG waveform diagrams [11]

2.2.1.3. Measurement for ECG Signal

Because the ECG has very small signals, at the range of a few *mV* (usually less than 10) it is often interfered with by the 60 Hz noise created by the power line or the human body. In this case, we knew that signal conditioning is very important for bio-potential measuring. The best-fitting conditioning for the bio-potential signal can make it much simpler to do further signal processing. Therefore, it is necessary to employ an instrumentation amplifier to reduce the 60 Hz noise and to amplify the ECG signals we are interested in. Then, we filter the low frequency DC noise by a low pass filter and amplify the signal by a gain and filter stage. The fault often occurs that the output waveform is very sensitive to motions like breathing and even slight movement of the human body, so it is necessary to add an anti-motion artifact stage to isolate the signal from motion artifacts. To create clearer signals with less 60 Hz noise interference on the baseline of the ECG waveform, we apply a DRL (Driven Right Leg) circuit to reduce the 60 Hz noise. The block diagram of the ECG sensor is shown in Fig. 2-5.

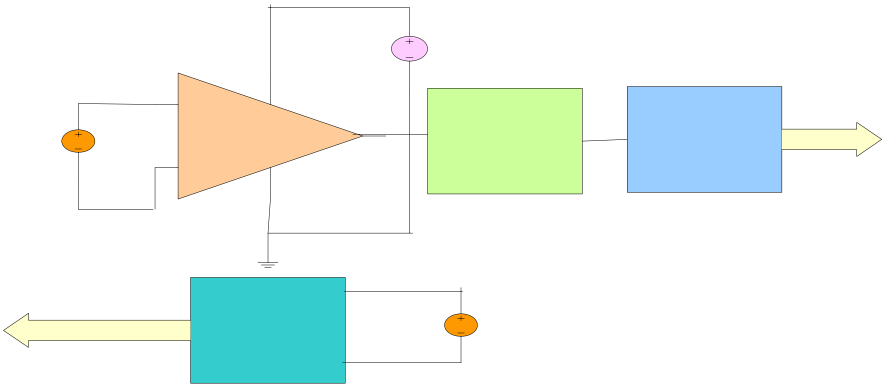


Fig. 2-5 Block diagram of ECG sensor

2.2.1.4. Circuit Design for the ECG Sensor

a. The Different Input Stage

The differential input stage [12] is shown in Fig. 2-6. Here we choose a micro-power consumption instrumentation amplifier AD627 (Analog Device). Its max supply current is only 85 µA, and it has a wide power supply range from +2.2 V to ±18 V. It also provides gain for the signal.

The gain is adjusted by the resister R^{G}, in the term:

$$R^{G} = \frac{200K\Omega}{Gain - 5}$$

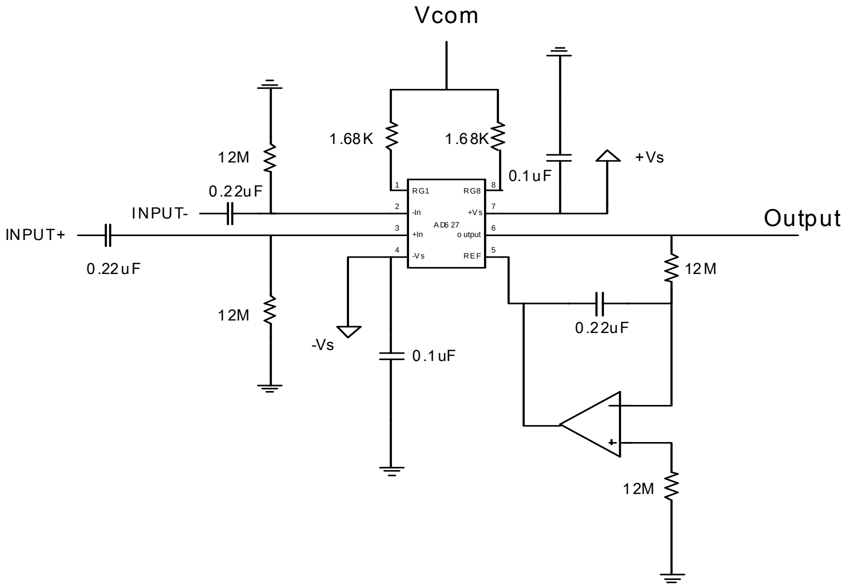


Fig. 2-6 semantic diagram of the differential input stage for the ECG sensor

We use $R^{G1} = R^{G8} = \frac{1}{2}$ with $R^{G} = 1.68$ KΩ, and the gain of the first stage is set to G^1 = 64.5.

b. The Gain and Filter Stage

In the gain and filter stage, shown in Fig. 2-7, we choose OP296 for the operation amplifier. Its max supply current is only 85 μA and consists of 2 OP amplifiers in one chip. For the sake of minimizing the scale, ICs with SMD packages are used. In this stage, the first OP amplifier serves as a gain stage where $G^2 = 20$, and the second OP serves as a low pass second-order Butterworth filter where the cut-off frequency $F_c = 75$ Hz.

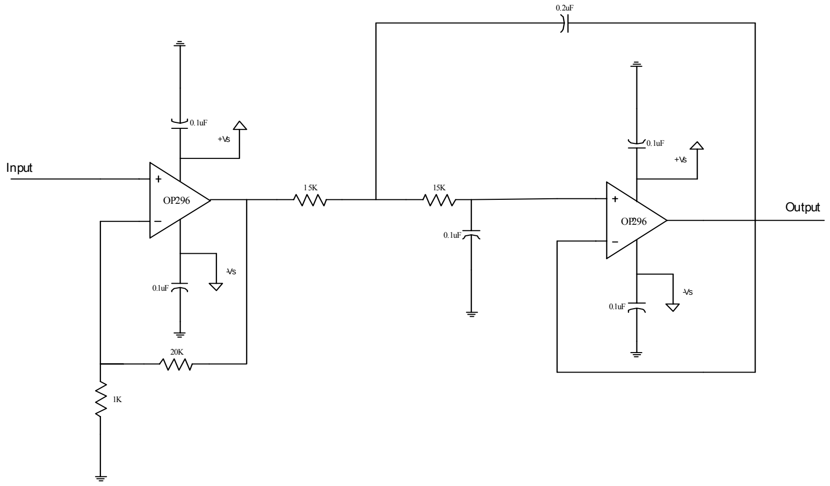


Fig. 2-7 Semantic diagram of the gain and filter stage for the ECG sensor

c. The Anti-Motion Artifact Stage

As for the anti-motion artifact stage, a second-order band-pass filter [12] is used, as in Fig. 2-8, and the resonance frequency Fr = $\frac{1}{2\pi C}\sqrt{\frac{R_1+R_3}{R_1 R_2 R_3}}$ = 10 Hz. The quality factor Q = $\pi R_2 C F_r$ =1.38. The bandwidth BW = $\frac{F_r}{Q}$ = 7.24. And the gain at resonance frequency Ar = -3.33.

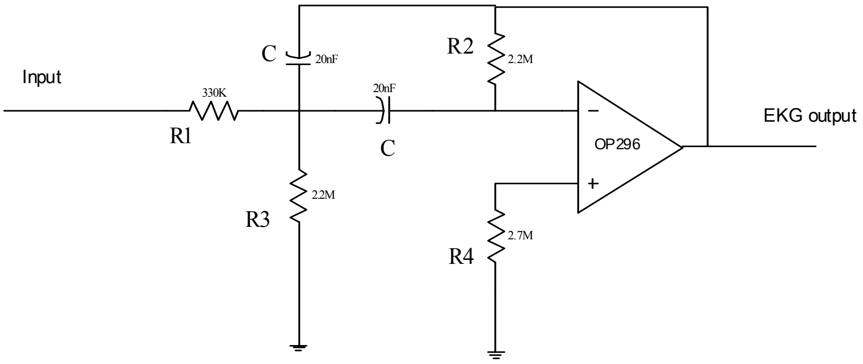


Fig. 2-8 Semantic diagram of the anti motion artifact stag for the ECG sensor

d. Driven-Right-Leg (DRL) Circuit

In the last stage, the Driven-Right-Leg (DRL) circuit [8] (Fig. 2-9), we use a small capacitor with a value of 10 nF to block the 60 Hz noise, and an auxiliary OP amp to feed the noise back to the human body by the DRL electrode.

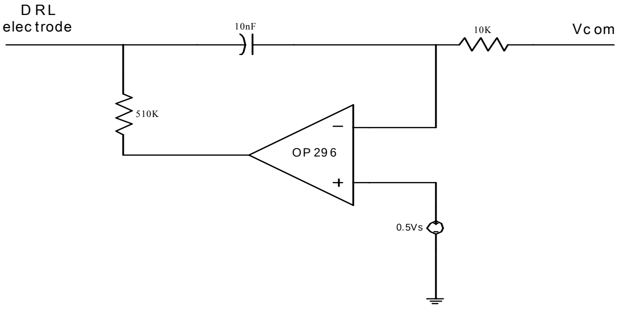


Fig. 2-9 Semantic diagram of the Driven-Right-Leg (DRL) circuit for the ECG sensor

2.2.2. SpO_2 Sensor

A commercial SpO_2 sensor is used in this study, which is an integrated pulse oximetry using NONIN OEM Ⅲ. The SpO_2 sensor sends physiological data to the UART at the rate of 9600 baud. This study combines the SpO_2 sensor with a development board, which is comprised by an MCU and ZigBee chip. The development board receives the physiological data of the SpO_2 sensor via UART, parses it to get available data and sends this data to a router wirelessly. This study chooses data of type 2 format for the SpO_2 sensor to write its parser, which was used to divide it into SpO_2 value, heart rate value and plethysmographic pulse.

2.2.2.1. Data Format of SpO_2

The data of format type 2 provides oximeter information on SpO_2, heart rate, pulse, sensor alarm, sensor disconnection, out of track, bad pulse, software firmware revision level and plethysmohraphic pulse. There are 75 frames of data sent per second. A frame consists of 5 bytes of data; the 1st byte of the frame is used for byte synchronization; the 2nd byte of the frame is the status of the SpO_2 sensor; the 3rd byte of the frame is the plethysmographic pulse value; the 4th byte of the frame may be HR, SpO_2 or software reversion.

a. SpO_2

The measurement of the SpO_2 in the range of 0 to 100 is sent three times per second in the 4th byte at frames 3, 28 and 53.

b. Heart Rate

The detection of heart rates in the range of 18 to 300 is 2 bytes. The heart rate's most significant byte is sent three times per second in the 4th byte at frames 1, 26 and 51. The heart rate's least significant byte is sent three times per second in the 4th byte at frames 2, 27 and 52.

c. Plethysmographic Pulse

The plethysmographic pulse is a representation of the IR signal; it has a range of 0 to 255 and is sent 75 times per second in the 3rd byte at all 75 frames.

2.2.2.2. SpO_2 Parser

This study develops a simple SpO_2 parser to get the heart rate, SpO_2 and SpO_2 diagram according to a data format of type 2. The data format of type 2 is described below. The flowchart of the SpO_2 parser is shown in Fig. 2-10.

a. Heart Rate

The parser gets the most significant byte of the heart rate by the 4th byte of the 1st frame, and the least significant byte of the heart rate by the 4th byte of the 2nd frame.

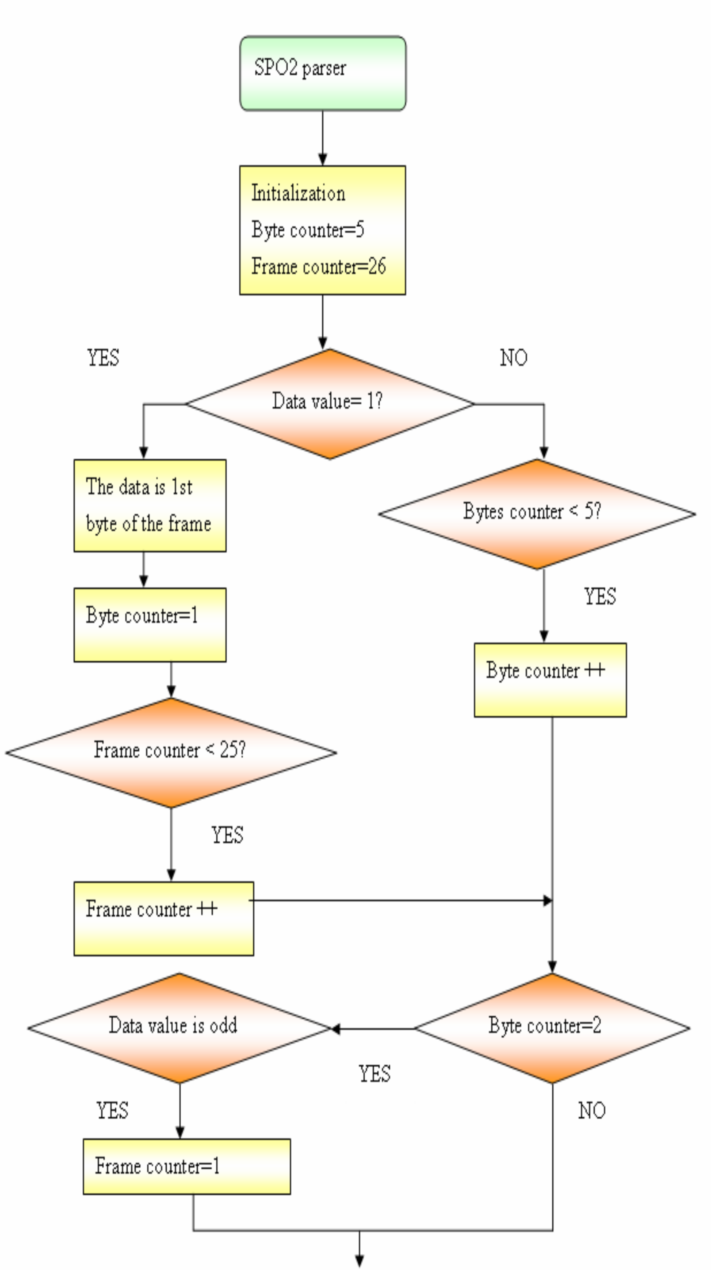
b. SpO_2

The parser gets the SpO_2 value by the 4th byte of the 3rd frame.

c. Plethysmographic Pulse Value

The parser gets the plethysmographic pulse value by the 3rd byte of the frame.

d. Flowchart of the SpO_2 Parser


SPO2 parser
Initialization
Byte counter=5
Frame counter=26
YES
NO
Data value= 1?
The data is 1st byte of the frame
Bytes counter < 5?
YES
Byte counter=1
Byte counter ++
Frame counter < 25?
YES
Frame counter ++
Data value is odd
Byte counter=2
YES
YES
NO
Frame counter=1

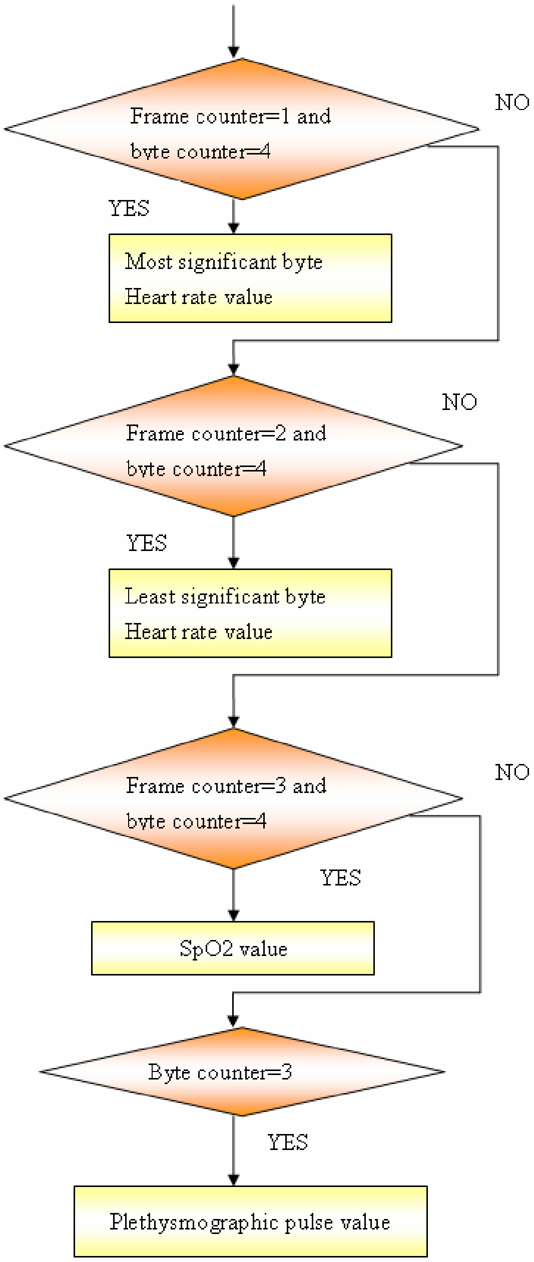


Fig. 2-10 Flowchart of the SpO_2 parser

Chapter 3 Wireless Sensor Network

3.1. Introduction to Wireless Transmission Specifications

In this chapter ZigBee and Bluetooth will be described, and the comparison between the two will be analyzed. Finally, design methods will be illustrated including hardware structure, firmware structure, message flow chart and PC application.

3.1.1. Bluetooth

The Bluetooth is a wireless personal area network that focuses on short range ad-hoc connectivity. Its operating frequency is in the Industrial-Scientific-Medical (ISM) frequency band of 2.402 GHz to 2.483 GHz. It uses a Time Division Multiplexing (TDM) technique to divide a channel into 625 micro sec slots. With Bluetooth each packet is transmitted on a different hop frequency. The Bluetooth device can be divided into two types, master and slave; a master connects with at most seven slave devices. A unit network of Bluetooth is called Piconet, as shown in Fig. 3-1, which includes a master device and a slave device.

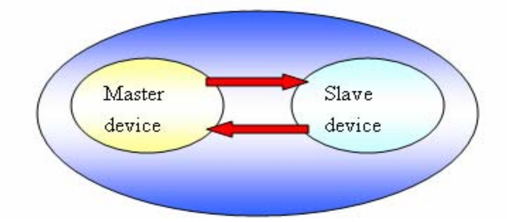


Fig. 3-1 Piconet diagram for Bluetooth

3.1.2. ZigBee

ZigBee is based on IEEE 802.15.4 wireless protocol, which focuses on sensor networks, control and home-care related applications. It has several advantages such as self-organization, lower power consumption, smaller size of protocol stacks and larger addressing space. The ZigBee can be classified into two types when accessing channels, unslotted networks and slotted networks. In unslotted networks, all devices are considered peers with respect to one another and the entire wireless resource is available.

Slotted networks comprise three time periods. The first period is the beacon frame; two beacon frames bound this structure. The second period is an active period that consists of a contention access period and a contention free period. All devices compete equally to get channel resources by using a Carrier Sensed Multiple Access with Collision Avoidance (CSMA/CA) mechanism during the contention access period. The channel resource can be allocated to specific devices during the contention free period. The third period is the inactive period. Channel access is not permitted during the inactive period. The IEEE 802.15.4 standard defines the lower two layers as Physical (PHY) and Medium Access Control (MAC) layers. ZigBee alliance builds on this foundation by providing a Network (NWK) Layer and Application (APP) layer.

3.1.2.1. Physical (PHY) Layer

The IEEE802.15.4 has three PHY layers that operate in three separate frequency ranges of 868-868.6 MHz, 902-928 MHz and 2.4-2.4835 GHz. There are 20 kb/s, 40 kb/s and 250 kb/s using the frequency bundles of 868-868.6 MHz, 902-928 MHz and 2.4-2.4835 GHz.

3.1.2.2. MAC Layer

The IEEE802.15.4 MAC layer is responsible for accessing the radio channel using two CSMA-CA mechanisms, transmitting a beacon frame, synchronization and providing a reliable transmission mechanism.

3.1.2.3. NWK Layer

The NWK Layer is responsible for implementing a mechanism that is used to join and leave a network. The NWK Layer also provided discovery and maintenance of routes between devices devolving to the NWK Layer.

3.1.3. Advantages of ZigBee

ZigBee has low power consumption and high addressing capability. ZigBee can receive 65536 devices to connect, significantly more than the seven devices of Bluetooth. ZigBee was developed to serve different applications than Bluetooth and its technology has led to optimizations in power consumption. According to the ZigBee Alliance, ZigBee has many advantages such as a very low duty cycle, long primary battery life, static and dynamic star and mesh networks. A comparison between ZigBee with Bluetooth is shown in Table 3-1.

	Operation Range	**Data rate**	**Waking time**	**Battery Life**	**System Resources**	**Network Size**
Bluetooth	10 m	250 Kbps	3 m sec	Several weeks	250 KB+	7
ZigBee	10-75 m	1 Mbps	3 sec	6 month to 2 years	4 KB- 32 KB	Unlimited 2^{64}

Table 3-1 Comparison between Bluetooth and ZigBee [1]

To compare ZigBee with Bluetooth in low power consumption accurately, this study compares Bluetooth [W7020, Lucent, USA] with ZigBee [UZ2400, UBEC, Taiwan] in terms of power consumption. The comparison is illustrated in Table 3-2, with which we can see that the power consumption of ZigBee is smaller than Bluetooth. Therefore, in this study we use ZigBee as the specification for wireless transmission.

	Sleep Mode	**TX**	**RX**
Bluetooth	2.8 V/ 50 μA	2.8 V/ 33 mA	2.8 V/ 40 mA
ZigBee	3.3 V/ 2 μA	3.3V / 22 mA	3.3V/ 18 mA

Table 3-2 Comparison between Bluetooth and ZigBee in Power consumption [15] [16]

This study uses the unslotted method to access channels in the 2450 MHz band. The 2450 MHz band provides the most channels (16 channels), highest data rate (250 kb/s), least overhead and least complexity, relative to the slotted method.

3.2. Hardware Design

The development board (DEB) used in this study is comprised of a micro-controller and a ZigBee chip (UZ2400, UBEC, Taiwan) to act as hardware platform for each device in the wireless sensor network.

3.2.1. Micro-Controller Unit (MCU)

The MCU of the development board us the Texas Instruments MSP430F1611, which incorporates a 16-bit RISC CPU, peripherals, and a flexible clock system.

3.2.2. ZigBee Chip

The ZigBee chip (UZ2400, UBEC, Taiwan) integrates a wireless RF transceiver operating at 2.4 GHz, the 802.15.4 PHY layer baseband and the MAC layer architecture. The block diagram of the DEB is shown in Fig. 3-2.

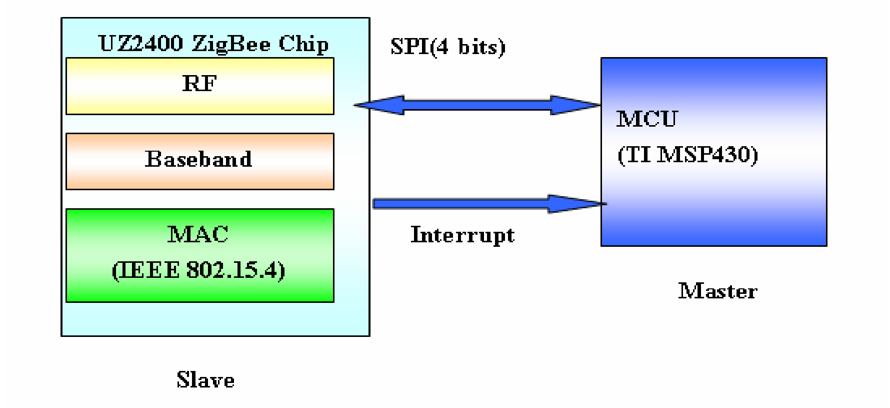
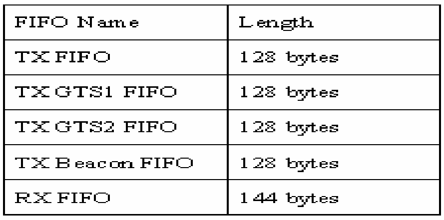


Fig. 3-2 Block diagram of the development board

The MAC of the ZigBee chip is comprised of six components, which are RX MAC, TX MAC, security, control register, FIFO and SPI interface. The FIFO of the MAC consists of five components, which are TX FIFO, RX FIFO, TX GTS1 FIFO, TXGTS2 FIFO and TX Beacon FIFO. These FIFO are listed in Table 3-3, which contains the FIFO name and its length.



FIFO Name	Length
TX FIFO	128 bytes
TX GTS1 FIFO	128 bytes
TX GTS2 FIFO	128 bytes
TX Beacon FIFO	128 bytes
RX FIFO	144 bytes

Table 3-3 Length for each FIFO

3.2.3. SPI Mode

The ZigBee chip communicates with the MCU by the SPI mode of the UART. This study uses 4-pin SPI including four serial signals, SPI enable, SPI clock, Master Input Slave Output (MISO) and Master Output Slave Input (MOSI). The MCU is the master and the ZigBee chip is the slave. If the MCU wants to read any register of the ZigBee chip, it should pull down SPI enable to zero voltage and send the register address with variable size according to the different types registered to the ZigBee chip. If the first bit of the address is 1, indicating that the register type the MCU wants to access is a long address register, it should send a 10-bit address of the register to the MCU. If the first bit of the address is 0, indicating that the register type the MCU wants to access is a short address register, it should send a 6-bit address of the register to the MCU. The last bit of the address is used to determine that the MCU wants to write or read the register. If the last bit of the address is 0, this indicates that the MCU wants to read the register. If the last bit of the address is 1, this indicates that the MCU wants to write the register. Furthermore, the ZigBee chip should send serial data to the MCU via a MISO pin if the register operation is read. If the register operation is writing, the MCU should send serial data to ZigBee via a MOSI pin.

3.3. Firmware and Software Design

The firmware of this study is a Real Time Operating System (RTOS) called CMX to interrupt the handle, provide a Timer and ZigBee protocol stack. First, the RTOS is described and the kernel functions used in the firmware are. Secondly, the ZigBee protocol stack, network layer design method based on the MAC layer, flowchart and message sequence chart are described. Finally, the design methods of the hardware and firmware are illustrated.

3.3.1. Introduction to RTOS

The firmware uses an RTOS called CMX, which is a real time multi-tasking operating system that supports many functions to develop real time multi-tasking applications. The heart of the operating system is the scheduler based on true preemption, which allows for tasks and interrupts to cause an immediate task switch. The firmware uses some tasks to handle the necessary jobs and a software timer to do periodical jobs by calling the kernel functions of the RTOS. These task related functions and cyclic timer functions are described below.

3.3.1.1. Task Related Functions

a. K_Task_Create Function

This function is used to create a task before entering the CMX RTOS. The ROM TCB task is a structural table used to describe the task ID, system stack of the task, the task’s code address and the task’s initial priority. The task is in the IDLE state when it is created.

b. K_Task_Start Function

This function is used to start a task; it puts the task into a ready state allowing it to become a running task when it is the highest priority task ready to run.

c. K_Task_Wait Function

This function is used to enable a task to suspend itself for a specific amount of time or indefinitely. The firmware uses this function to suspend an ISR task and wait for the ISR to occur. These steps are described below.

d. K_Task_Wake Function

This function is used to wake up a specific task that was put into the suspend state; it puts the task into a resume state. The firmware uses K_Task_Wait and K_Task_Wake functions in the interrupt process, this process is described as follows. The firmware selects two steps to handle the interrupt. The first step is called top halve and is responsible for doing critical tasks in order to increase sensitivity of the system. The second step is called bottom halve and is responsible for doing the remainder of the interrupt. To achieve this goal, the firmware uses a task called ISR task to handle the bottom halve; the ISR is responsible for handling the top halve. The ISR task remains in an idle state by using the K_Task_Wait kernel function of the RTOS until the ISR wakes it by using the K_TASK_WAKE kernel function of the RTOS.

e. K_OS_Start Function

This function is used to invoke the CMX RTOS once the function is called; the CMX RTOS takes control of the CPU and determines when tasks should run.

f. K_OS_Intrap_Entry Function

This function is used to save the context of the CPU registers and swap in the interrupt stack when an interrupt occurs; the interrupt's first instruction is this function.

g. K_OS_Intrap_Exit Function

This function is used to return the original sequence of the program when the interrupt has finished its code. This function is the last instruction of the interrupt's code. The firmware implements three ISRs, including the timer ISR, external ISR and ADC ISR, by writing in assembly language. The first instruction of the ISR code is the K_OS_Intrap_Entry function, and its last instruction is the K_OS_Intrap_Exit function.

3.3.1.2. Timer Related Functions

The CMX RTOS supports a software cyclic timer to execute a specific routine at the specified number of system ticks. When the timer expires, the K_Event_Signal function is performed. These details are described below.

a. K_Timer_Create Function

This function is used to sets up a cyclic timer's event function.

b. K_Timer_Start Function

This function is used to initially start a cyclic timer that has been set up by the K_Timer_Create function.

c. K_Event_Wait Function

This function is used to wait for a specific event to occur; this function is arranged in pairs with the timer related function to do a specific routine.

3.3.2. ZigBee Protocol Stack

The ZigBee protocol stack is comprised of four layers, the Physical Layer (PHY), Medium Access Control Layer (MAC), Network Layer (NWK) and Application Layer (APP). The lower two layers of the ZigBee protocol stack are IEEE 802.15.4 standards. The ZigBee protocol stack is shown in Fig. 3-3.

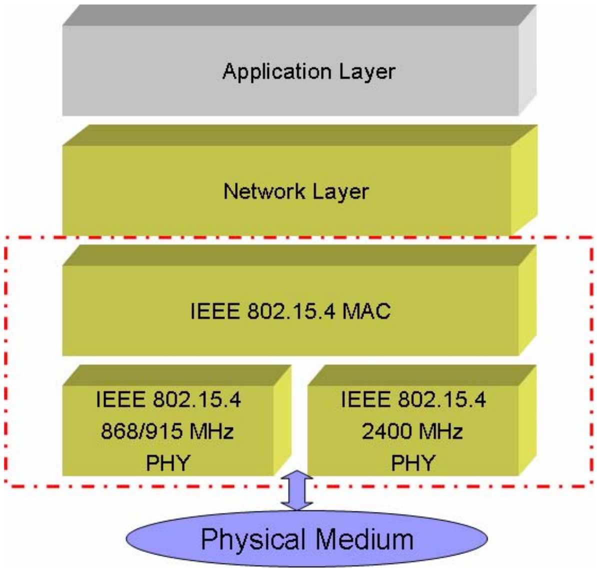


Fig. 3-3 ZigBee protocol stack diagram

3.3.2.1. Description of the IEEE 802.15.4 WPAN

The IEEE 820.15.4 consists of two devices, the Full Function Device (FFD) and the Reduce Function Device (RFD). The FFD can operate in three modes serving as a PAN coordinator, a coordinator, or a device. The RFD is used for extremely simple applications, such as a passive sensor device that only sends sensing data to the associated router or PAN coordinator periodically.

3.3.2.2. Description Concept of Primitives

A layer of the ZigBee protocol stack might have several services; each layer of ZigBee offers services to the user in the layer immediately above by building its functions on the service of the layer immediately below.

3.3.2.3. This concept is illustrated in the following figure

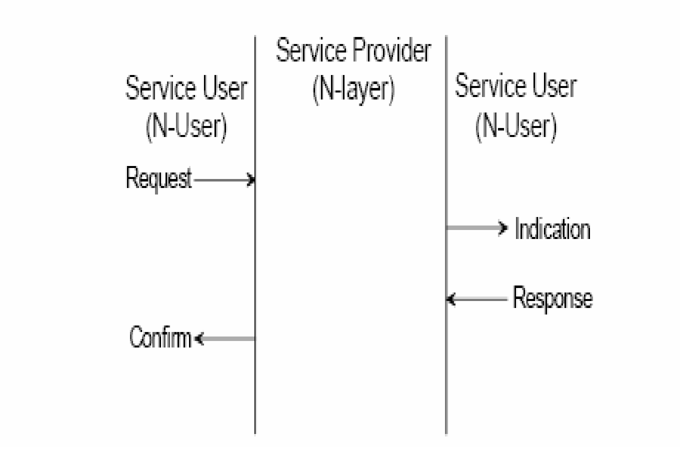
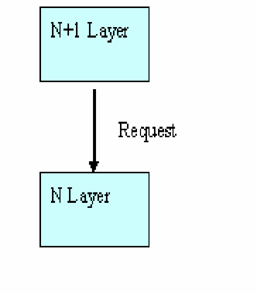


Fig. 3-4 Flow diagram of primitive concept

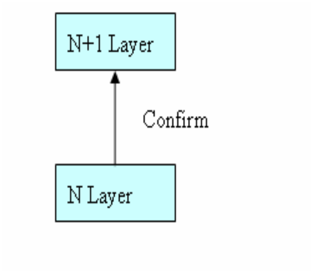
Each services of a layer can be implemented by describing the service primitives and parameters. A service might consist of one or more related primitives. Each service primitive might have zero or more parameters that convey the information required to provide the service.

3.3.2.4. A primitive can be one of four generic types:

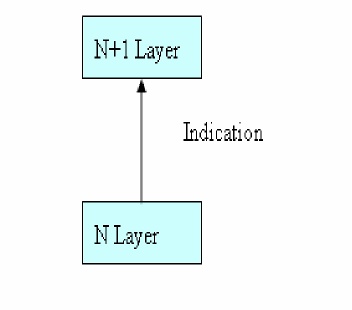
a. Request: The N+1 Layer requests a service from the N Layer.



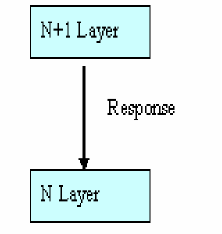
b. Confirm: The N layer issues a confirm primitive to report the result of the request when the requested service has been completed.



c. Indication: The N+1 Layer indicates that n internal N Layer event has happened by issuing an indication primitive.

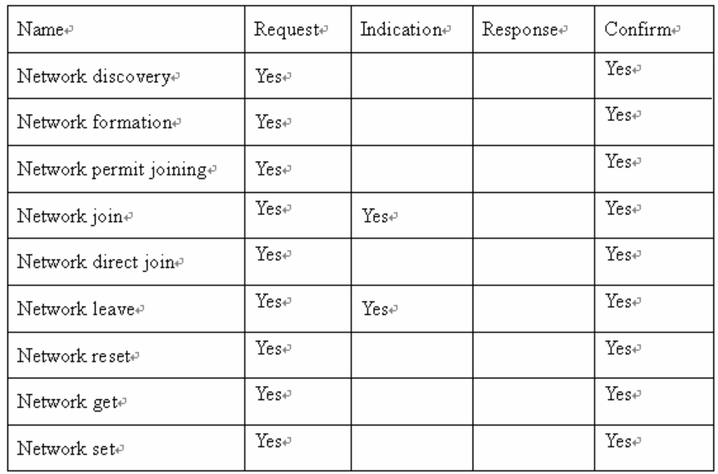


d. Response: The N+1 Layer issues a response primitive to notify the N Layer when N+1 Layer has received an indication from the N Layer.



3.3.3. Network Layer Design Based on a MAC Layer

The network layer is responsible for creating a network, processing network joining and leaving of devices, and routing frames to their destination by using a MAC layer service. This study implements primitives of a network layer according to ZigBee specifications. The network layer consists of two entities, a Network Layer Data Entity (NLDE) and a Network Layer Data Entity (NLME). The NLDE entity is responsible for network layer data services that include data request, data confirm and data indication primitives. The NLME entity is responsible for network management services that include network discovery, network joining and leaving and routing algorithms. The NLME entity comprises several primitives to achieve network management services; these primitives are listed in Table 3-4.



Name	Request	Indication	Response	Confirm
Network discovery	Yes			Yes
Network formation	Yes			Yes
Network permit joining	Yes			Yes
Network join	Yes	Yes		Yes
Network direct join	Yes			Yes
Network leave	Yes	Yes		Yes
Network reset	Yes			Yes
Network get	Yes			Yes
Network set	Yes			Yes

Table 3-4 Summary of the primitive for the NLME entity

3.3.3.1. NLDE Entity

The NLDE entity consists of three primitives, the data request primitive, data confirm primitive and data indication primitive; these primitives are described in Appendix A.

3.3.3.2. NLME Entity

The NLME entity comprises many primitives, such as network discovery, network join /leave and other primitives concerning network management; these primitives are described in Appendix A

3.3.4. Implementation of the ZigBee Protocol Stack

This ZigBee protocol stack uses an Operating System (OS) called the CMX Tiny real time multi-tasking operating system, which provides a true preemptive operating system and a wide range of functions, and uses as little RAM as possible. We port the CMX Tiny real time multi-tasking operating system on the TI MSP430 platform, and the RTOS supports many kernel functions to handle interrupts, timer, task scheduling and resource management of the hardware. First of all, the firmware system initializes the Universal Asynchronous Receiver/Transmitter (UART), the serial peripheral interface

(SPI), the micro-controller unit (MCU) and the MAC, and then it creates three tasks, the timer task, ISR task and system task. Finally, these tasks are initiated.

This study implements the network layers of the ZigBee protocol stack based on the MAC layer. When the ZigBee chip receives a packet or sends a packet, its ISR is run. The ISR is responsible for saving the event status of RX or TX and then waking the ISR task to process receive events or transmission events. The firmware employs four tasks to handle four necessary jobs, including interrupt handles, transmission of bio-signals, maintenance of the system timer and device functionality. According to ZigBee specifications, the device type is divided into three types, the end device, router and PAN coordinator. The system task is responsible for implementing the functionalities of each device in PAN.

3.3.4.1. End Device with Sensor

This device is a passive type device that is responsible for sending physiological data to the associated router. This study uses two sensors, an ECG sensor and SpO_2 sensor. The output data of the ECG sensor is an analog value, so it should been sent into the ADC of the MCU for conversion. When the conversion of ADC is completed, the ADC12 ISR is run. The ADC12 ISR is used to obtain conversion results and send them to the associated router. Because the SpO_2 sensor outputs sensing data to the UART, we should connect the UART of the end device with the UART of the SpO_2. When the end device receives data from the UART and the process of parsing had been completed, it sends data to the associated device. The hardware structure is shown in Fig. 3-6.

a. ECG Sensor Device

a.1. Flowchart Description

The firmware of the end device with ECG sensor has four tasks, including the timer task, ISR task, system task and send task, which are responsible for doing necessary jobs according to their attributes. These are illustrated below. The flowchart is shown in Fig. 3-5.

Timer Task: The timer task is responsible for waiting for one system tick and storing the TTL (time to live) of packet to the TX buffer.

ISR Task: When the ISR task starts, it enters suspend status until the ISR wakes it. The ISR task is waked when the ISR occurs. The ISR task is used to process the received packet and parse the RX frame if the interrupt is an RX event. Or, it stores the status of packet transmission into the TX buffer if the interrupt is a TX event.

System Task: The responsibility of this device is to join a router according to the results of network discovery and start the send task.

Send Task: When this task has been started, it remains in sleep mode until it receives a start command. If it receives a start command, it is awakened from sleep mode to normal mode to start a timer and the ADC. When conversion of the ADC has been completed, it runs the ADC ISR that is responsible for getting conversion results and sending them to the associated router. When the timer expires, the send task stops sending data to the router and enters sleep mode again.

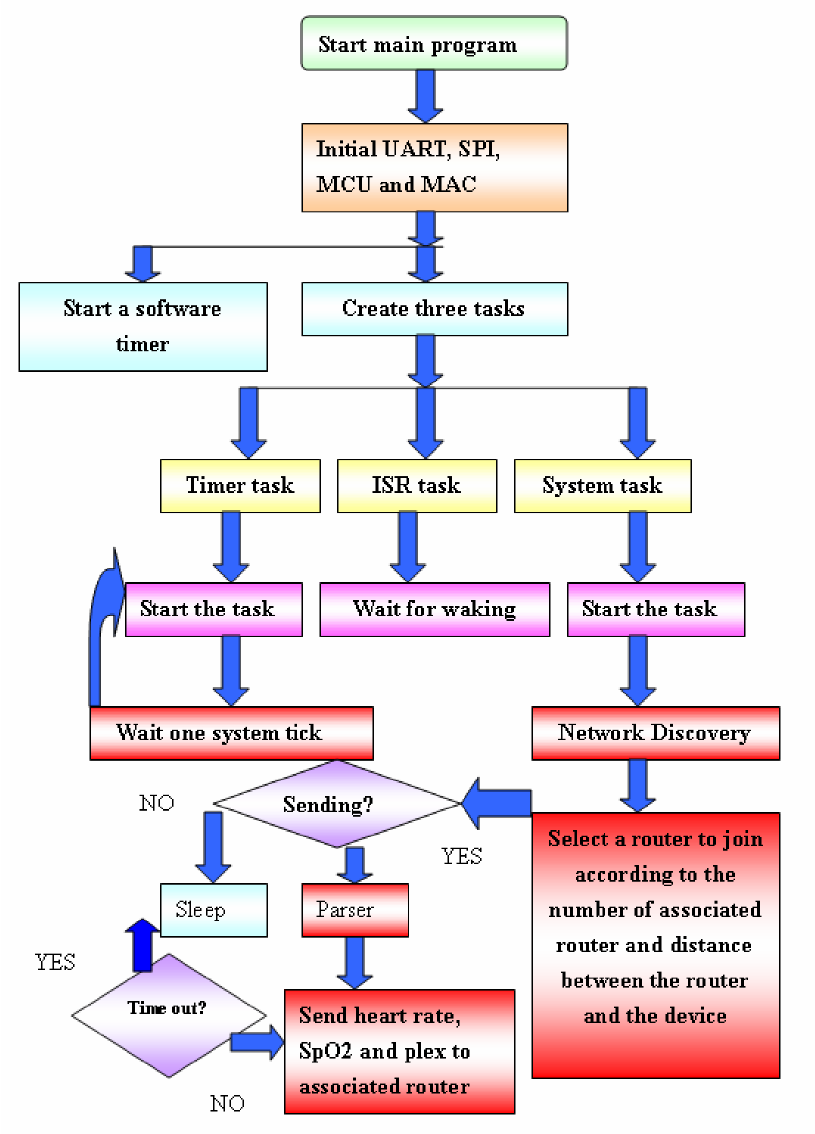


Fig. 3-5 Flowchart of end device with ECG sensor

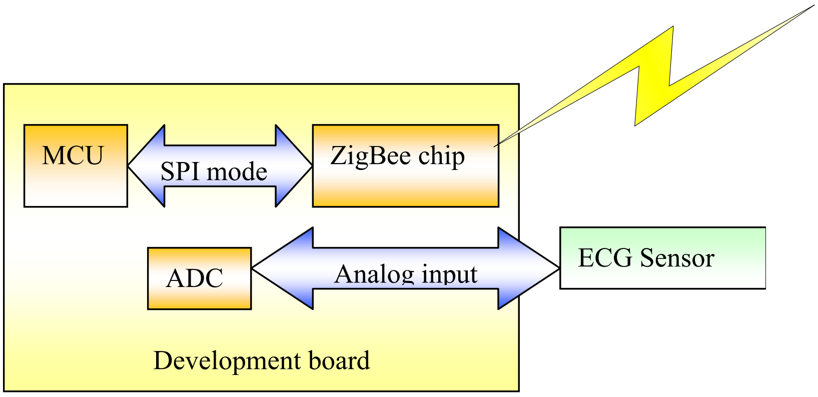


Fig. 3-6 Hardware block diagram

a.2. Message Flow of the ECG sensor device

When the end device with ECG sensor is powered on, it does network discovery by issuing the NLME-DISCOVERY-request primitive and joins a router of the network according to the discovery results returned from the NLME-DISCOVERY-confirm primitive by issuing the NLME-JOIN-request primitive. Furthermore, it remains sleeping until receiving a start command from the associated router. If the end device with ECG sensor receives a start command, it starts a timer and ADC converter that is used to convert the physiological analog signals of the ECG and sends the results of conversion to the associated router by issuing the NLME-DATA-request primitive. When the timer expires, the end device with ECG sensor stops converting and remains sleeping until receiving a start command from the associated router. The message flow chart of the ECG sensor device is shown in Fig. 3-7.

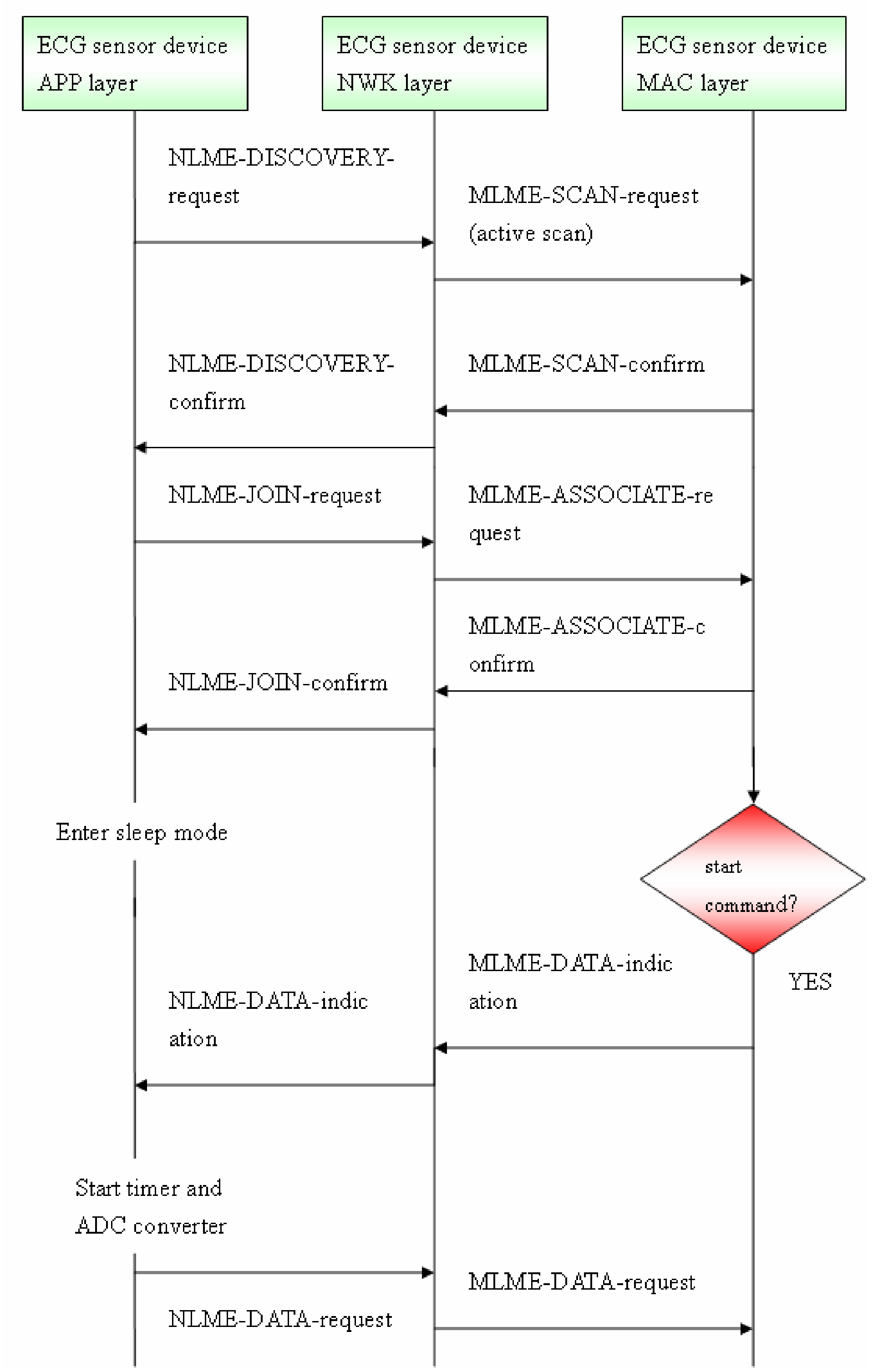


Fig. 3-7 Message flow chart of the end device with ECG sensor

b. SpO_2 Sensor Device

b.1. Flowchart Description

The firmware of the end device with SpO_2 sensor is comprised of three tasks: the timer task, ISR task and system task, which are responsible for doing the necessary jobs according to their attributes. This firmware implements a parser of data type 2 for the SpO_2; the parser is used to extract the SpO_2 value, heart rate value and plethysmographic pulse value from raw data. The flowchart of the SpO_2 sensor device is shown in Fig. 3-8, and the hardware structure of the SpO_2 sensor device is shown in Fig. 3-9.

Timer Task: The timer task is responsible for waiting for one system tick and storing the TTL of a packet to the TX buffer.

ISR Task: When the ISR task has been started, it enters suspend status until the ISR wakes it. The ISR task is awakened when the ISR occurs. The ISR task is used to process received packets and parse the RX frame if the interrupt is an RX event. Or, it stores the status of packet transmission into the TX buffer if the interrupt is a TX event.

System Task: This task first joins a router according the results of network discovery and remains sleeping until receiving a start command. If the device receives a start command, it starts a timer and polls the UART to determine whether it has received any data. If the UART has received data, the system task uses a parser for SpO_2 data type 2 to extract a SpO_2 value, heart rate value and plethysmographic pulse value and sends them to the associated router until the timer expires.

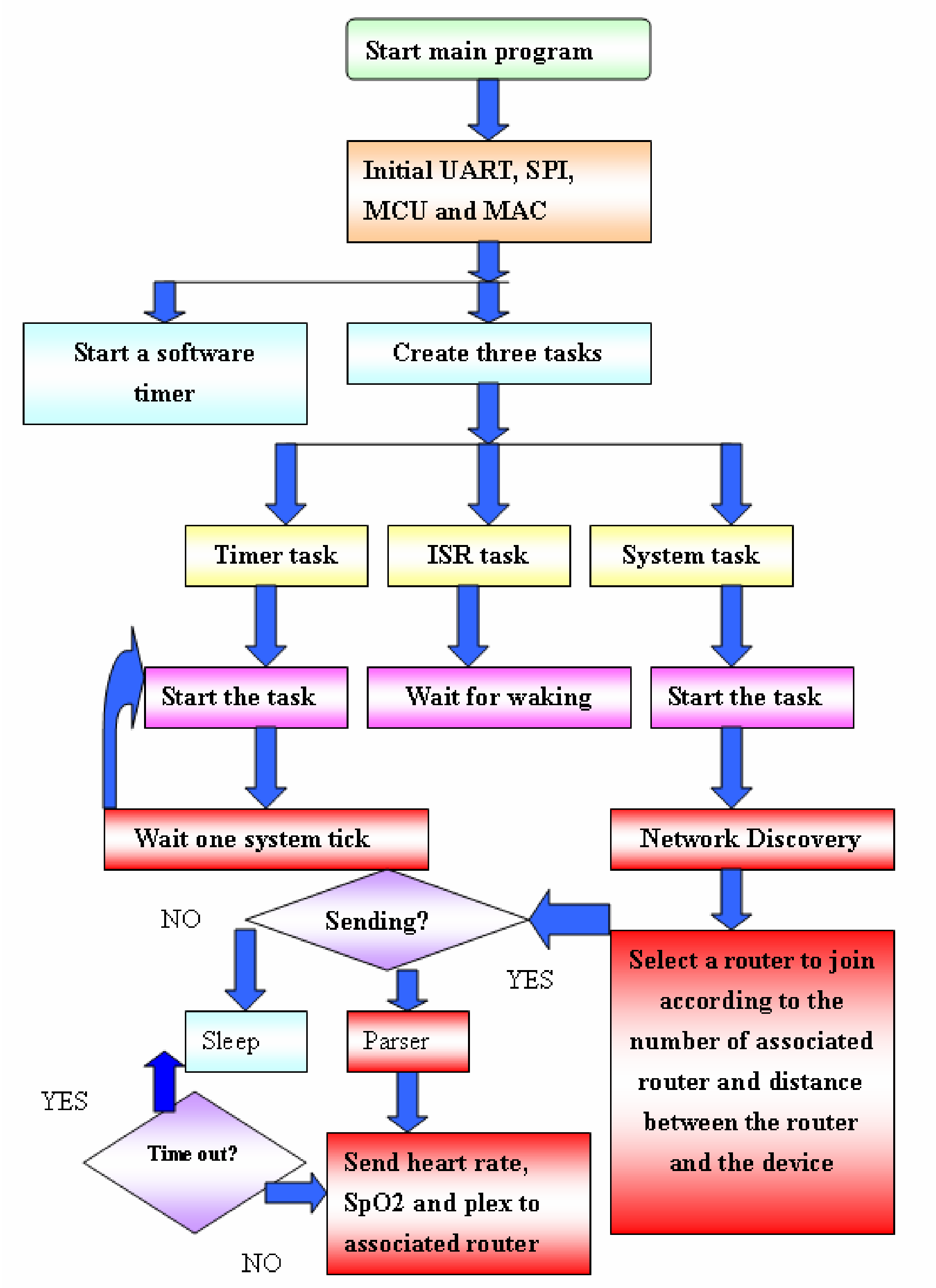


Fig. 3-8 Flowchart for the end device with SpO_2 sensor

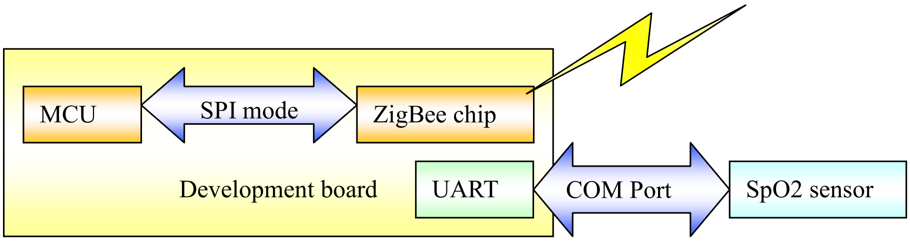


Fig. 3-9 Hardware block diagram of the end device with SpO_2 sensor

b.2. Message Flow of the SpO_2

When the end device with SpO_2 sensor is powered on, it first does network discovery by issuing the NLME-DISCOVERY-request primitive and then joins the router of a network according to the discovery results returned from the NLME-DISCOVERY-confirm primitive by issuing the NLME-JOIN-request primitive. Afterwards it remains sleeping until receiving a start command from the associated router. When the end device with SpO_2 sensor receives a start command, it starts a timer and parser that are used to extract a SpO_2 value, heart rate value and plethysmographic pulse value and send them to the associated router until the timer expires by issuing the NLME-DATA-request primitive. When the timer has expired, the end device with SpO_2 sensor stops sending and remains sleeping until receiving a start command from the associated router. The message flow chart is shown in Fig. 3-10

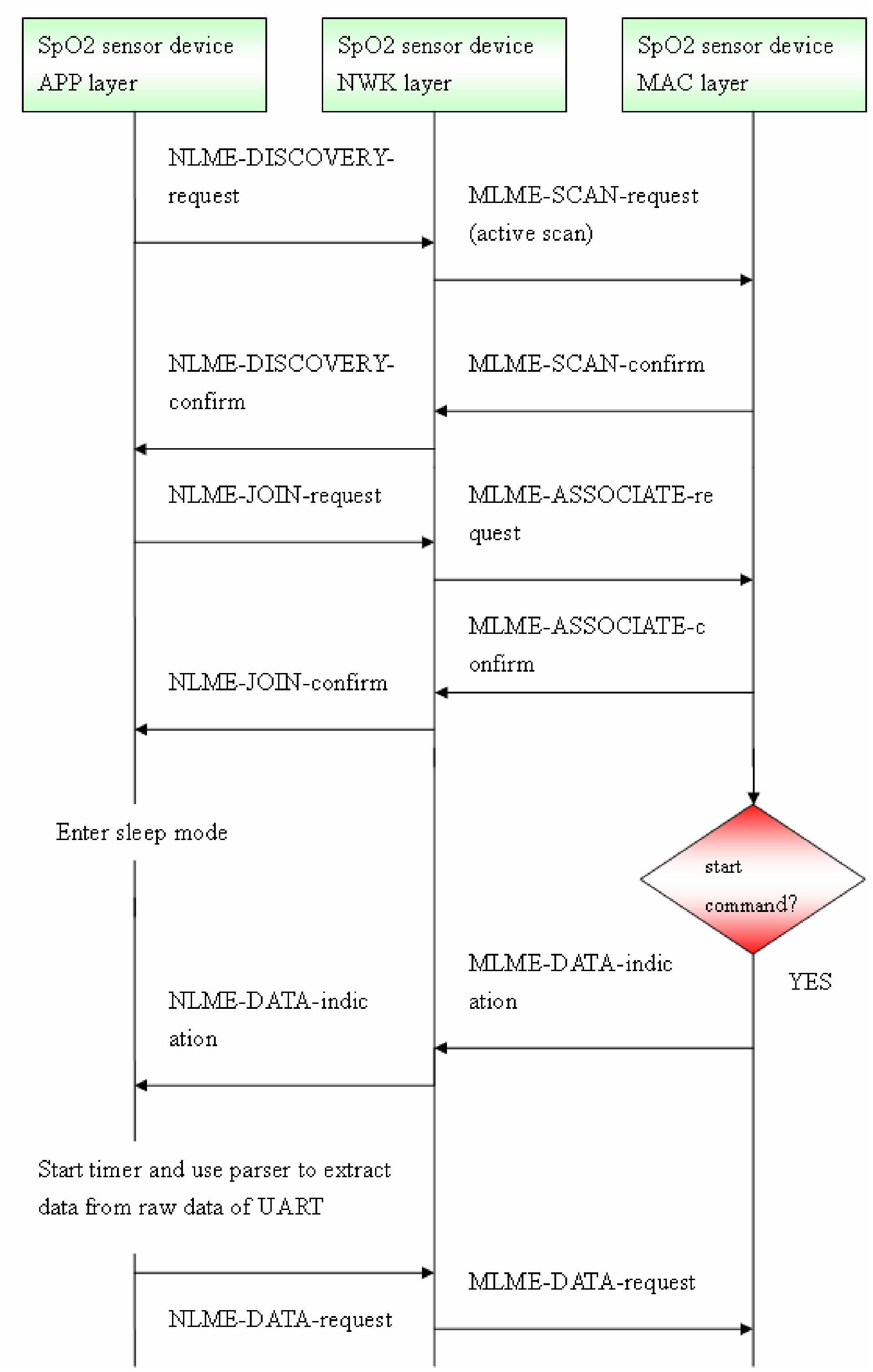


Fig. 3-10 Message flow chart for end device with SpO_2 sensor

3.3.4.2. Router

The router is responsible for scheduling sending time for each associated device according its neighbor table and using hierarchical routing to forward data to specific devices. To reduce power consumption of sensor devices, this study inserts an idle time in the middle of the sending time of two devices. The idle time is adjusted by the scheduler according to number of associated end device and end device‘s attributes that contain type of signal and signal frequency. The time slice diagram of sending times for each device is shown in Fig. 3-11.

Fig. 3-11 Time slice diagram of sending times for each device

a. Flowchart Description

The firmware of the router performs four tasks: the timer task, ISR task, system task and switch task, which are responsible for doing the necessary jobs according to their attributes. The flowchart of the router is shown in Fig. 3-12.

Timer Task: The timer task is responsible for waiting for one system tick and storing the TTL (Time to Live) of a packet to the TX buffer.

ISR Task: When the ISR task has started, it enters into suspend status until the ISR wakes it. The ISR task is waked when the ISR occurs. The ISR task is used to process received packets and parse the RX frame if the interrupt is an RX event. Or, it shows the processing status of packet transmission if the interrupt is a TX event. If the received packet is a data frame, the router sends data to the associated PAN Coordinator wirelessly.

System Task: The responsibility of router is to schedule sending times of each associated device and wait for other devices to join. The router first does network discovery by requesting the NLME-network-discovery-request of the network layer services. The NLME-network-discovery-request issues the NLME-scan-request primitive of the MAC layer services to do active scanning, which broadcasts beacon request commands to existing devices of the POS. Finally, the router chooses a PAN Coordinator to join according to the number of associated routers and the distance between the PAN Coordinator and the router.

Switch Task: The switch task is responsible for scheduling sending time for each associated device according its neighbor table and adjusting its idle time according to the signal type and attributes of the associated end device.

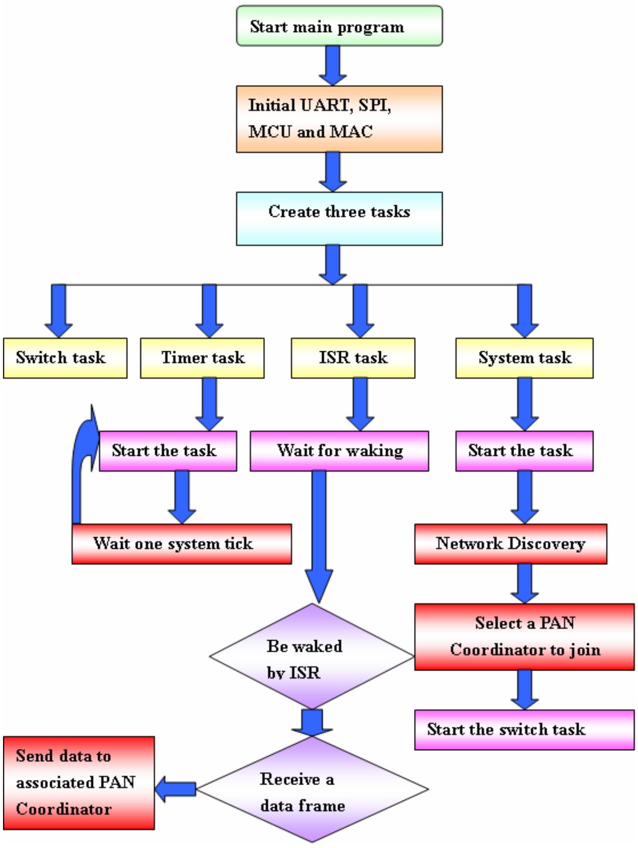


Fig. 3-12 Flowchart of the router

3.3.4.3. PAN Coordinator

The PAN Coordinator is only one device of the wireless personal area network; its responsibility is

to manage the WPAN, gather data from associated routers and
wait for other unassociated routers to join.

a. Flowchart Description

The firmware of the PAN Coordinator is comprised of three tasks: the timer task, ISR task, and system task, which are responsible for doing the necessary jobs according to their attributes. These are illustrated as follows. The flowchart of the PAN coordinator is shown in Fig. 3-13.

Timer Task: The timer task is responsible for waiting for one system tick and storing the TTL (Time to Live) of a packet to the TX buffer.

ISR Task: When the ISR task has been started, it enters into suspend status until the ISR wakes it. The ISR task is awakened when the ISR occurs. The ISR task is used to process received packets and parse the RX frame if the interrupt is an RX event. Or, it shows the processing status of packet transmission if the interrupt is a TX event.

System Task: The responsibility of the PAN Coordinator is to form the network and wait for the router to join. The PAN Coordinator first does network formation by requesting the NLME-network-formation-request of network layer services. The NLME -network- formation-request issues the NLME-scan-request primitive of MAC layer services to do energy detection and select a channel that has the maximum energy in all channels as the current channel of packet transmission. The PAN ID of the network is determined by a random number or input parameters of the NLME -network-formation-request, and then does active scanning by issuing the MLME -scan-request primitive of MAC layer services. Finally, it configures a new super-frame by issuing the MLME-start-request primitive of MAC layer services. When the PAN Coordinator has received a beacon request, it first delays a random tick time and then broadcasts a beacon frame to existing devices of the POS. Due to different arrival times of the beacon frame, the router can get all information of the existing PAN Coordinators in the POS.

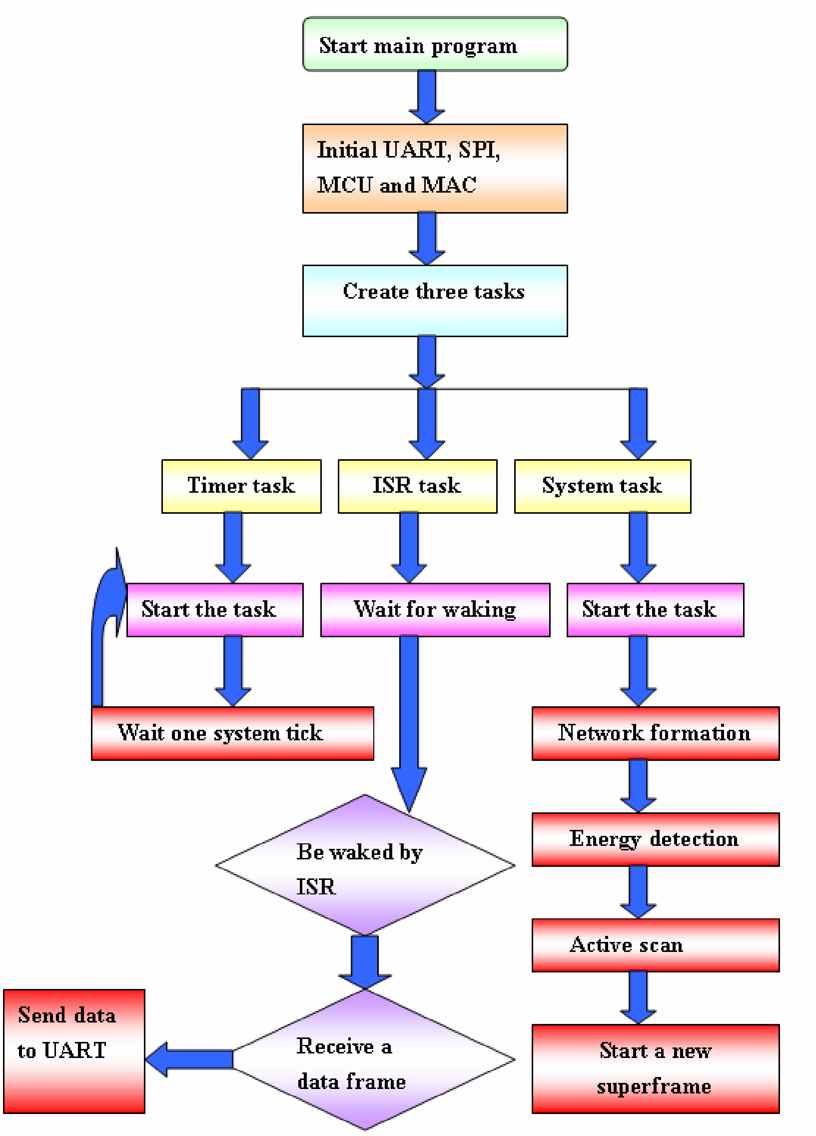


Fig. 3-13 Flowchart of the PAN Coordinator

b. Message Flow for the PAN Coordinator

When the PAN Coordinator device is powered on, it starts a new personal area network (PAN) by issuing the NLME-network-formation-request primitive and remaining asleep until receiving data. The NLME-network-formation-request primitive first does energy detection by issuing the MLME-scan-request primitive with scan type parameters set to energy detection. Once the energy detection scan is completed, its results are returned via the MLME-scan-confirm primitive. The PAN coordinator accesses the channels according to increasing energy measurements and discards those channels whose energy levels are beyond an acceptable level. Furthermore, the NLME-network -formation-request searches for existing networks by issuing the MLME-scan-request primitive with active scanning and a valid channel list.

The NLME-network-formation-request selects a suitable channel and PAN ID that does not conflict with other existing PANS. Once a suitable channel and PAN ID are determined, the NLME-network-formation chooses 0x0000 as the 16-bit network address. Then the NLME-network-formation initializes the new super-frame configuration by issuing the MLME-start-request primitive with PAN coordinator parameters set to true. The beacon order and super-frame order are set the same as those given to the NLME -network-formation-request.

Finally, the NLME-network-formation-request issues the NLME-network –formation -confirm to notify the next higher layer of the request status with the same status returned from the MLME-start-confirm primitive. The message flow chart is shown in Fig. 3-14.

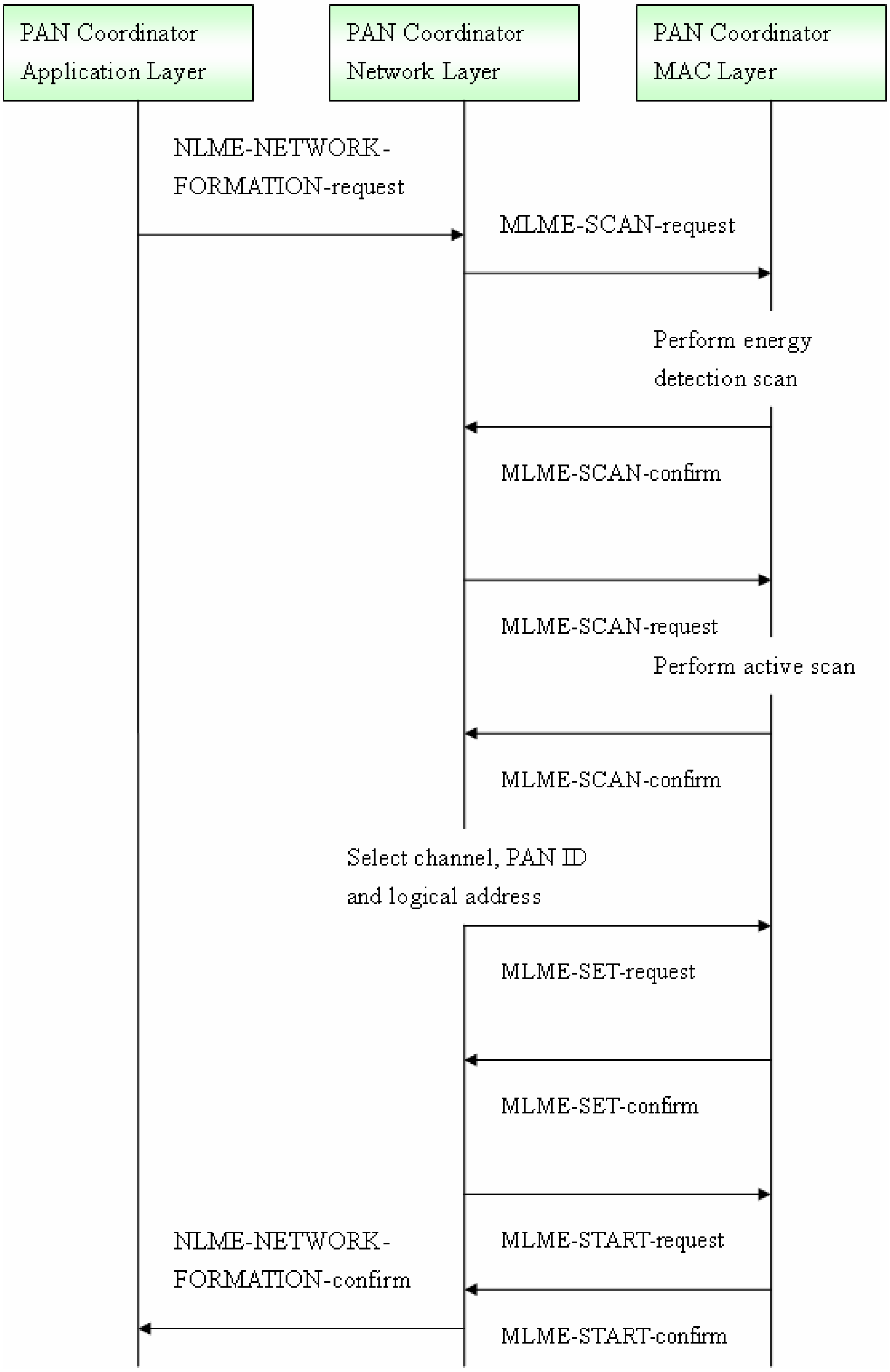


Fig. 3-14 Message flow chart for the PAN Coordinator

3.4. PC Application

The PC application is used to receive data via the UART and send data to a remote database via the ODBC API.

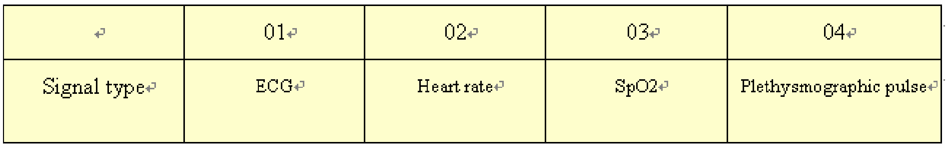
3.4.1. Development Language and Tools

The PC application is a windows application, which was developed by Microsoft visual C++ 6.0 using Microsoft Foundation Class (MFC).

3.4.2. Flowchart of PC Application

The application on a PC first finishes ODBC's initial operation of allocation for an environmental handler and used it to set ODBC's environment. After ODBC's environment is set, the ODBC's initial operation allocates a handler for database connection and uses it to connect to a remote database. When the connection of the database is completed, the ODBC's initial operation allocates a handler for SQL and uses the handler to run SQL commands.

After the above two operations are completed, the application program runs an SQL command to find the maximum number of records. When the UART receives data, the application program begins a thread to handle the data. The thread is called the receive thread, and it is responsible for processing data received from the UART, writing that data to a specific table of a remote database according to the data type, and showing its curve on a PC application. In this study, there are four types of data, which are ECG signals, heart rate, SpO_2 value and plethysmographic pulse value. They are assigned a unique identifier according to data type. These IDs for data types are listed in table. 3-5, and the flowchart of PC end application is shown in Fig. 3-15.



	01	02	03	04
Signal type	ECG	Heart rate	SpO2	Plethysmographic pulse

Table 3-5 IDs of all signal types

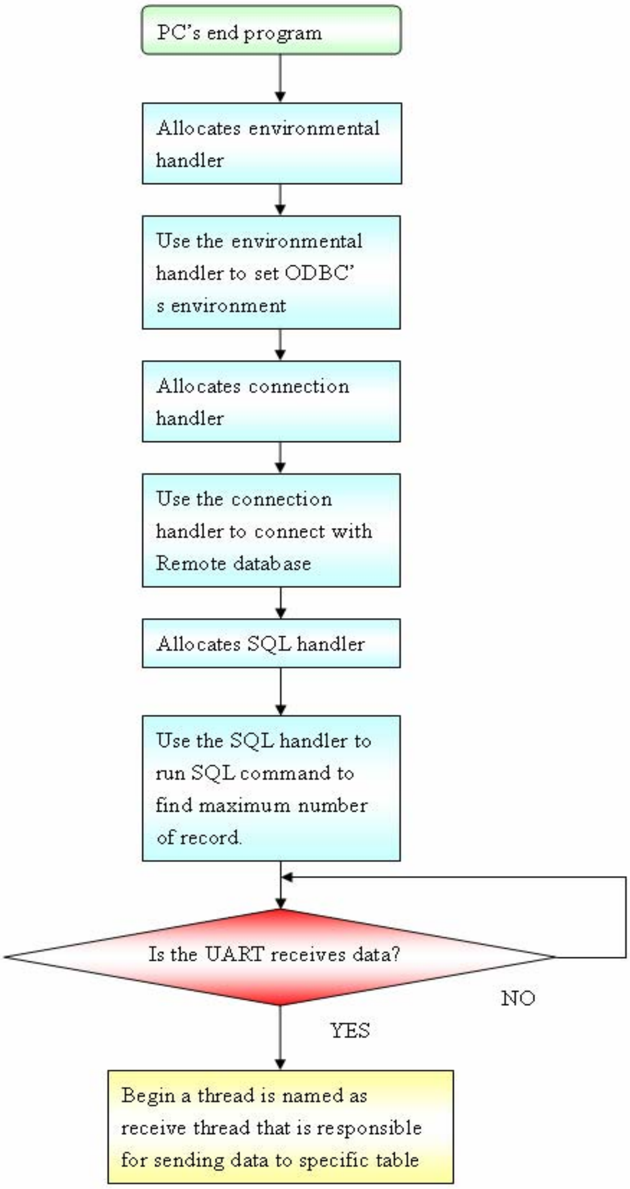


Fig. 3-15 Flowchart of PC's end program

Chapter 4 Web-based Management System

In this chapter the web architecture and design method for the database are described. The web page of this study uses web 2.0 technologies, including Ajax and SVG to show curves smoothly. These technologies are described in this chapter.

4.1. Overview of the Web-based Management System

On the PC end this study develops an application responsible for writing physiological signal values that are received from a wireless sensor network via an RS232 interface to a remote database by using an ODBC API [13]. The web page comprises XML, PHP, Ajax and SVG technology. This study uses Ajax and SVG technology to show physiological signal diagrams. The display performance of Ajax technology is better than traditional web page technology. The SVG technology supports many functions to display diagrams smoothly. The system structure is shown in Fig. 4-1, which includes a front end system and a rear end system. The front end system includes a wireless sensor network and PC application. The rear-end system comprises a web page and database. The front end system is connected with the rear end system via the internet by using an ODBC APT to write data to a remote database.

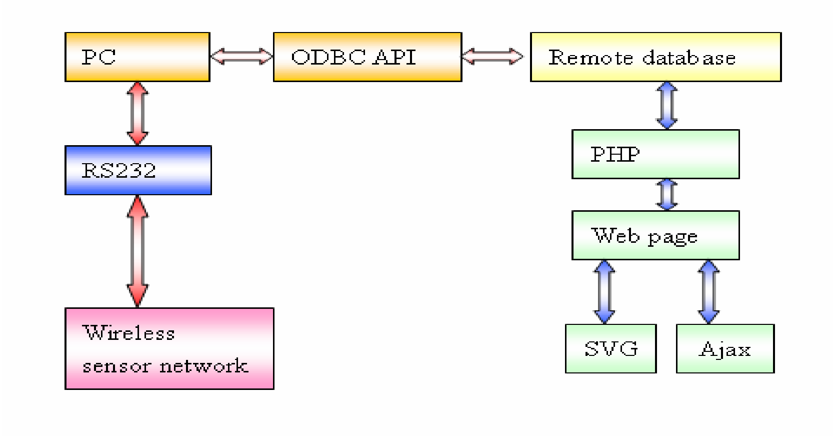


Fig. 4-1 Overview diagram of the web-based management system

4.2. Web Technology

4.2.1. Ajax (Asynchronous Java Script + XML)

Ajax comprises two technologies, asynchronous java script and XML. When the web page calls a java script function call, it triggers a call to an Ajax engine. The Ajax engine is responsible for sending HTTP request commands to a remote web server. When the web server returns a response to the Ajax engine, the Ajax engine processes it. Because the web browser does not need to wait for a response from the web server by using Ajax technology, its performance is better than traditional web pages, which must wait for a response from the web server. The model of the Ajax web page is shown in Fig. 4-2.

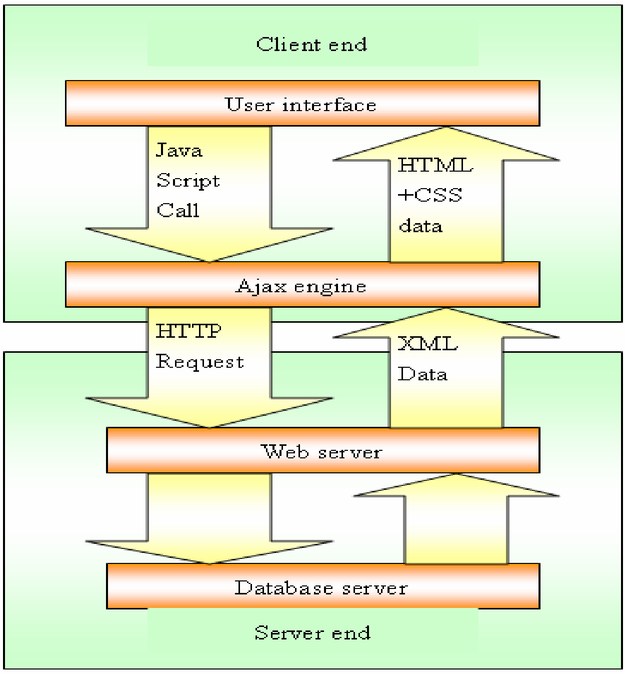


Fig. 4-2 Model of the Ajax web page

4.2.2. SVG (Scalable Vector Graphics)

This study uses the newest Scaleable Vector Graphics (SVG) technology in order to show physiological signal diagrams immediately. SVG is a text based graphics format used to describe graphics using XML grammar.

4.2.3. Web Program Design

Since the wireless sensor network is comprised by many sensor nodes, its management and control are difficult problems. To solve these problems, this study uses a database as our data interface between the web interface and the wireless sensor network. The web page uses Ajax, PHP and SVG technologies to achieve good performance while showing physiological diagrams smoothly.

4.2.3.1. Flowchart of the Web Page Program

When the sensor device is powered on, it searches for existing PANs and joins one of the routers to the PAN. When the router knows there is a device that has joined it, it sends a register message to the PAN coordinator. The PAN coordinator then sends a register message to the PC via a UART interface. At the PC end, this study develops a program to retrieve a register message from the PAN coordinator via the UART and send a register message to a remote database using ODBC API. The web page sends the period time for retrieving data back to the router. The router starts a timer with the period time to inform specific sensors that it is staring to send physiological signal data to the router. Finally, doctors can use a web browser to monitor physiological data anywhere and anytime. The flowchart of the web page program is shown in Fig. 4-3.

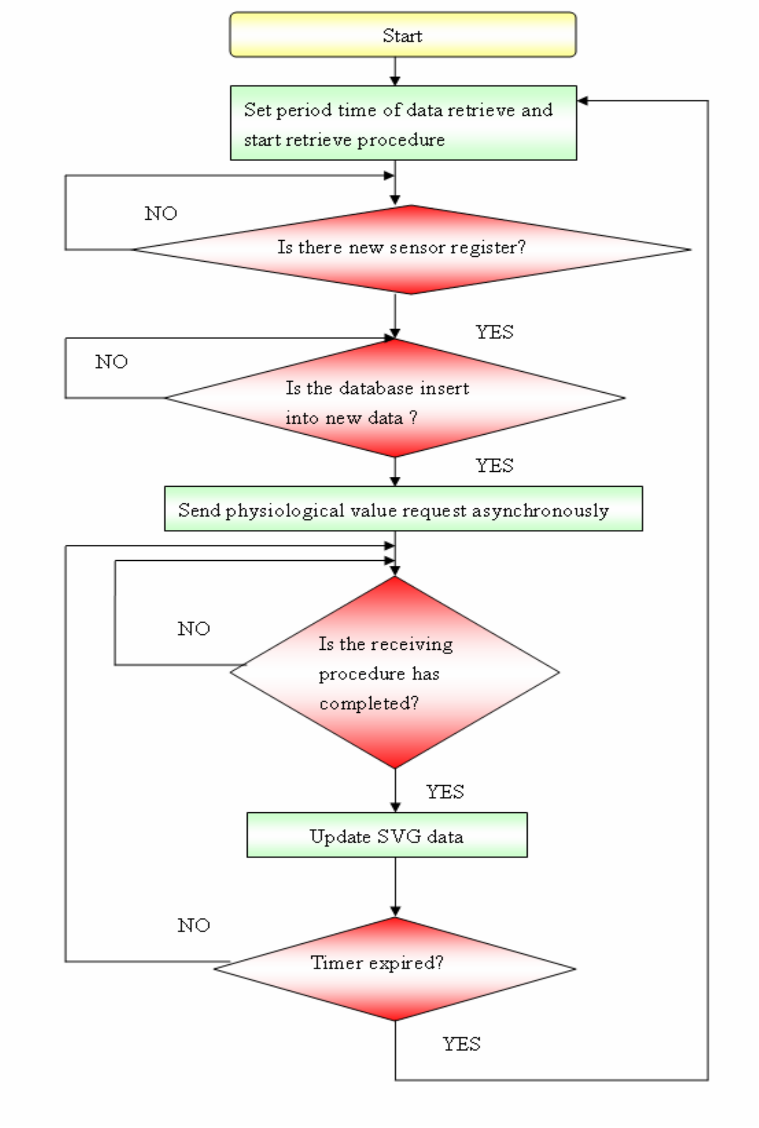


Fig. 4-3 Flowchart of the web page program using Ajax and SVG technologies

4.3. Database Design

The database is comprised of five tables: ECG, heart rate, $SpO_{2,}$ plethysmographic pulse and sensor status. The tables of ECG, heart rate, SpO_2 and plethysmographic pulse have the same fields, which contain patient ID, signal ID, most significant byte of physiological value, least significant byte of physiological value and used field. The used field records whether the physiological value has been read.

The sensor table is used to record the status of sensors. This table contains five fields: type, name, start time, end time and switch. The type field represents types of physiological signals, which are ECG, heart rate, SpO_2 value, SpO_2 diagram and status. The table formats of the ECG table, heart rate table and plethysmographic pulse table are shown in Fig. 4-4. The packet format for the sensor is shown in Fig. 4-5. The ODBC structure is shown in Fig. 4-6.

NO	Rec_time	id	Bio_id	Hi_signal	Low_signal	Voltage_x1	used

Fig. 4-4 Table format for ECG data; the fields are heart rate table, SpO_2 and plethysmographic pulse

Type	Name	Start time	End time	Switch

Fig. 4-5 Table Format for status field of the sensor

4.3.1. ODBC API

Open Database Connectivity (ODBC) is a standard interface for SQL. ODBC API supports several functions to connect and access remote databases.

4.3.2. ODBC Structure

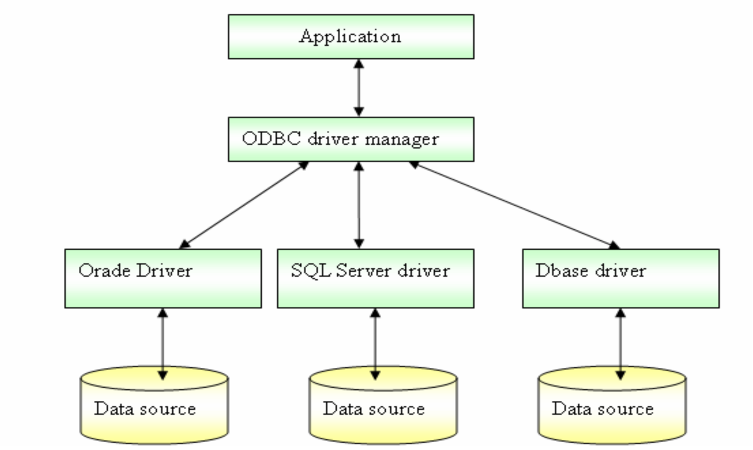


Fig. 4-6 Structural diagram of ODBC

4.3.2.1. Application

The application uses the ODBC interface to run specific tasks and sends an SQL request to a remote database by using ODBC API.

4.3.2.2. ODBC Driver Manager

The ODBC driver manager is a dynamic-link library provided by Microsoft as part of the ODBC installation. The ODBC driver manager is responsible for loading and unloading the appropriate ODBC drivers. The driver manager is also responsible for processing several initialization calls, entry points to ODBC functions for each driver, and providing parameter validation and sequence validation for the ODBC function call.

4.3.2.3. Driver

The ODBC driver is responsible for the majority of the functionality for data sources. When an application calls the ODBC function, the driver manager loads a driver to perform the necessary functions. Once the driver is loaded, a connection with a data source is established. The driver interprets the request and sends the request to a data source when an application submits a request to the data source. The ODBC driver is also responsible for converting to other formats if the data needs to be converted.

4.3.2.4. Data Source

The data source may be an Oracle or SQL server. In the study we choose the SQL server as the data source.

4.3.2.5. Data Source Name (DSN)

Before using the ODBC we assign an interface that is used to communicate with the ODBC driver manager. The interface is called Data Source Name (DSN).

4.4. ODBC API

4.4.1. SQL

The ODBC API [14] supports a standard method and many functions to connect and access remote databases by executing the SQL. In the ODBC, the running type of the SQL is divided into two types:

direct running and prepared running. Prepared running is mostly used to insert and delete data; direct running is mostly used to run analysis of the SQL.

4.4.2. Result of the SQL

When the SQL is completed, the ODBC returns a result that is a point, which therefore collects like a table. We use the point to go through all results by using ODBC API to move the point; furthermore, we also modify data of the remote database by using the point.

4.4.3. Use Procedure of ODBC API

The most important problem of ODBC API is the transformation between two different data types. The ODBC had three types of data, which are listed as follows.

(a) SQL data type in a database

(b) Data type in ODBC

(c) Data type in C language

4.4.4. Data type transfer between Database and C language

In the procedure for using ODBC API, there are two data type transfers. The first transfer of data type is between C language and ODBC; the secondary transfer of data type is between ODBC and SQL. Therefore, we know that the data type of ODBC is a bridge between C language and the SQL. The hierarchy for the transfer of data types is shown in Fig. 4-7.

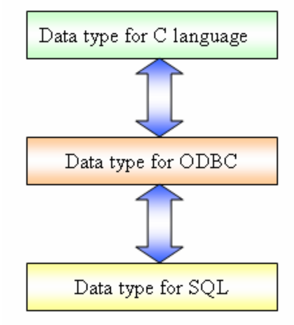


Fig. 4-7 Hierarchy for transfer of data types

Before using the functionality of ODBC, firstly an environmental handle is allocated; and then a connection handle is allocated; finally the handle for the SQL is allocated. The ODBC operation can be divided into seven steps: allocation of the handle, connection, initialization, execution of the SQL statement, fetching result, transaction and disconnection. The flowchart for ODBC operation is shown in Fig. 4-8.

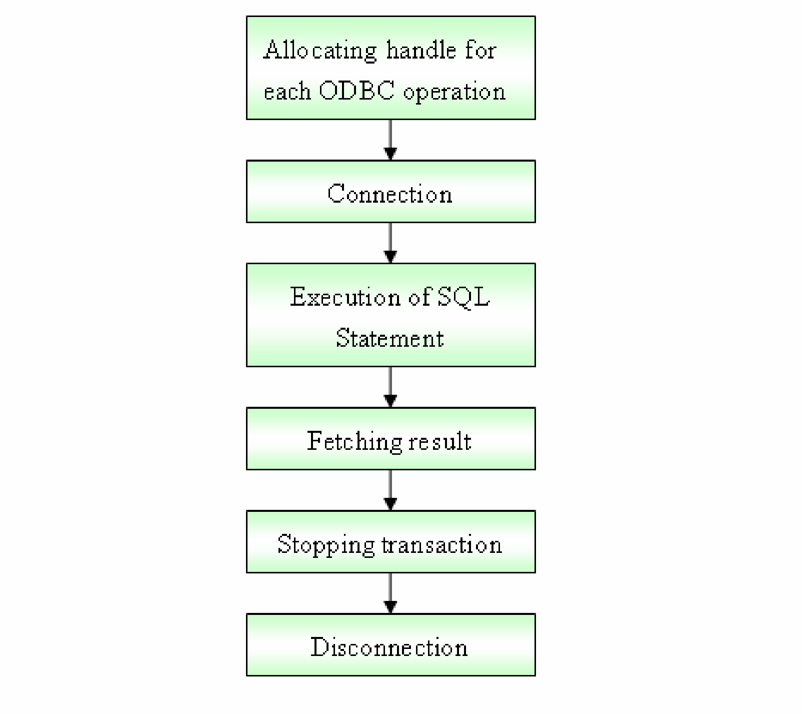


Fig. 4-8 Flowchart of ODBC operation

Chapter 5 Experimental Results

The design of the ECG sensor circuit is quite successful, as seen in Fig. 5-1, which shows an ECG signal on the oscilloscope. Filtering and compression techniques are used to remove unwanted noise. The size of the wireless ECG sensor node is acceptable as it can easily be placed on a patient without discomfort. For the SpO_2 sensor device, this study combines the SpO_2 sensor with a wireless device to send the heart rate, SpO_2 value and SpO_2 diagram to the associated router.

The implementation of a ZigBee protocol stack on the TI MSP430 functions properly with the packet format. The system is created using the CMX RTOS, which not only provides the platform by which our threads can execute, but is also very useful in performing the requirements of the ZigBee protocol.

We implement a wireless sensor network to do remote ECG and SpO_2 monitoring. We view the patients' ECG data and SpO_2 data from the remote webpage, as shown in Figs. 5-3 and 5-5. This achieves the essence of telemedicine. In the prototype of the ECG sensor device and SpO_2 sensor device using a wireless sensor network, as shown in Figs. 5-6 and 5-7, the resulting waveform viewed from the webpage is identical to the waveform of the ECG and SpO_2 sensors. Figs. 5-1 and 5-4 show a comparison between the oscilloscope waveform and the webpage waveform. This illustrates the reliability and accuracy of our system. This system provides a convenient interface for the end user (professional medical personnel or even the patient) to monitor a patient's physical condition.

5.1. ECG Waveform on a Sensor

The ECG waveform is shown by using an oscilloscope to measure the output of the ECG senor.

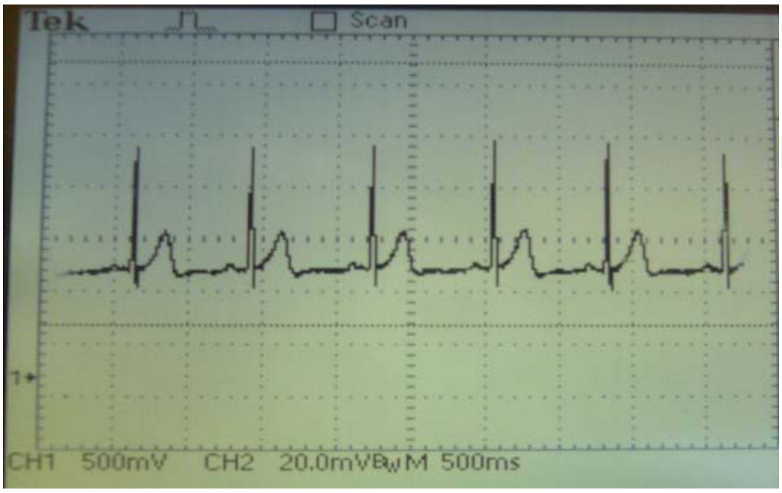
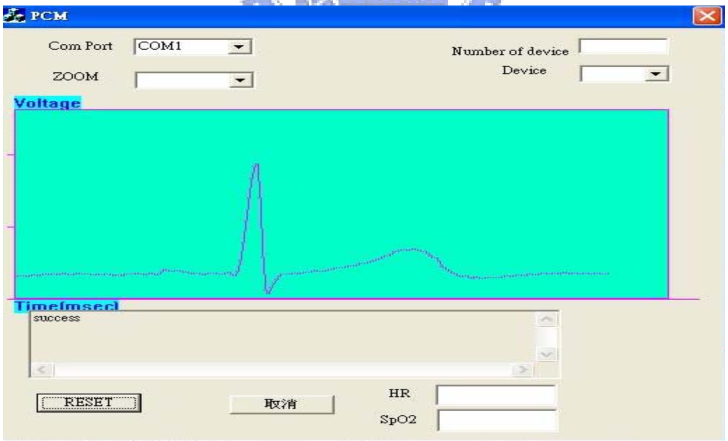


Fig. 5-1 ECG waveform on a sensor

5.2. ECG Waveform on a PC Application

The PAN Coordinator sends measured data to the UART, which is connected with the UART of a PC. The PC application is responsible for receiving this data via the UART (its curve diagram is shown in Fig. 5-2) and sending the data to a remote database using ODBC API.



Com Port COM1

success

Fig. 5-2 ECG waveform on a PC application

5.3. ECG Waveform on a Web Browser

The web program polls the remote database and retrieves physiological data from a specific table according to the type of the sensor device. If the device is an ECG sensor device, it retrieves physiological data from the ECG table and shows its curve diagram using Scalable Vector Graph (SVG) technology. A doctor can use a web browser to view physiological diagrams of patients anywhere.

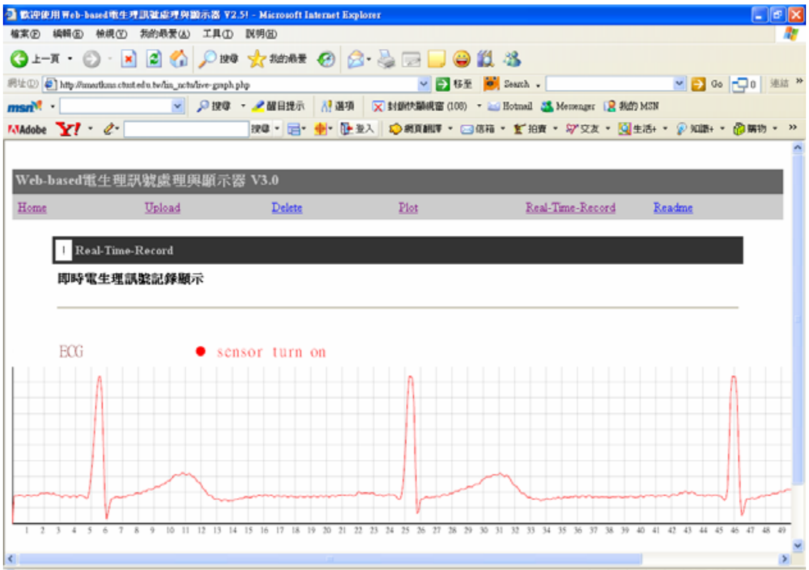
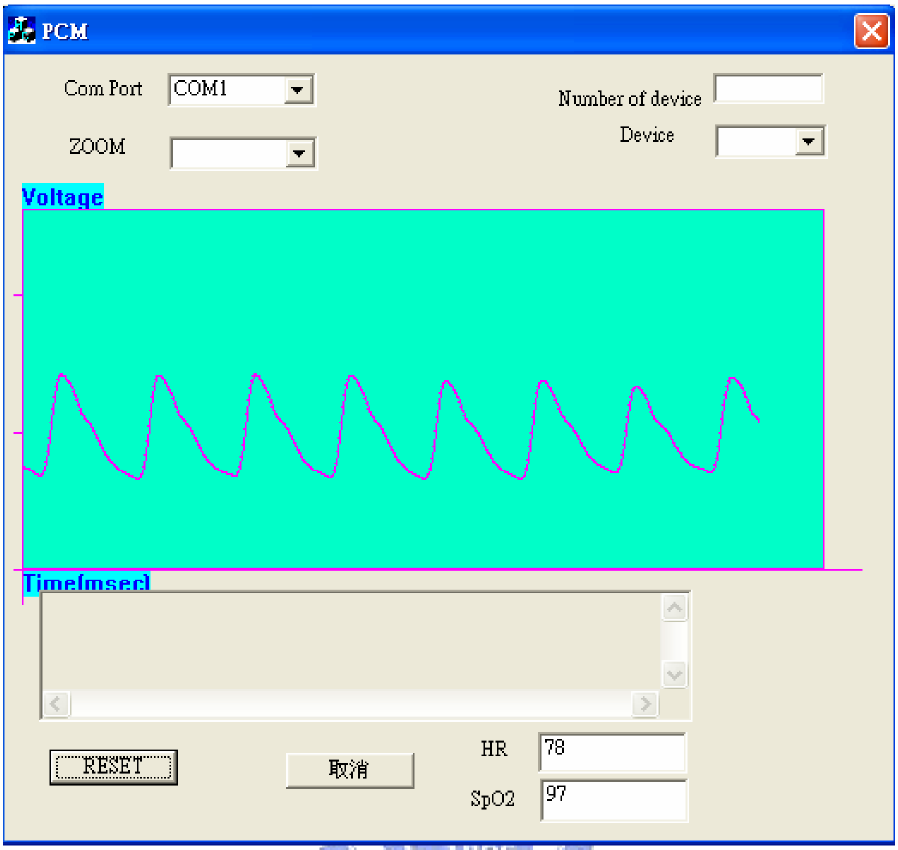


Fig. 5-3 ECG waveform on a web browser.

5.4. SpO$_2$ Sensor Device

The SpO$_2$ sensor device's measured data contains heart rate, SpO$_2$ value and SpO$_2$ diagrams. The PC application receives this data via the UART and shows its curve diagram and related information, such as heart rate and SpO$_2$ values on a form of a PC application. We can view the heart rate value, SpO$_2$ value and SpO$_2$ diagrams in a PC application, as shown in Fig. 5-4.

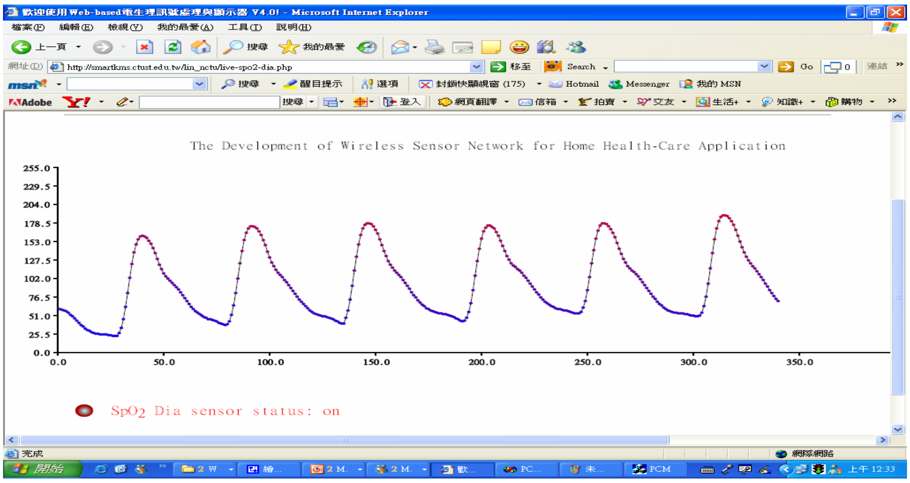


Com Port: COM1

HR: 78

SpO2: 97

Fig. 5-4 SpO$_2$ waveform on a PC application.



http://smartkms.ctust.edu.tw/lin_nctu/live-spo2-dia.php

The Development of Wireless Sensor Network for Home Health-Care Application

● SpO2 Dia sensor status: on

Fig. 5-5 SpO$_2$ waveform on a web browser.

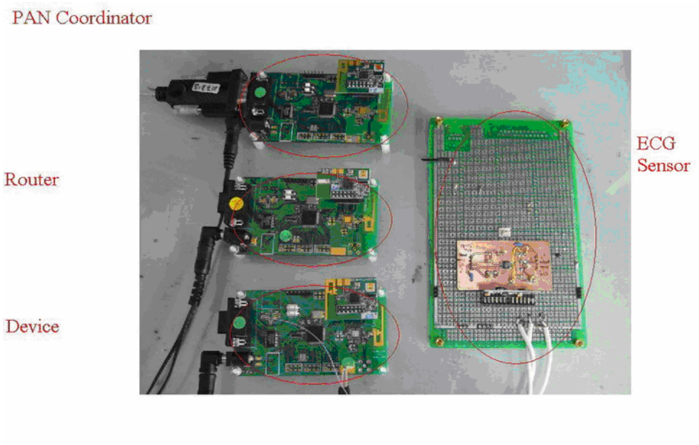


Fig. 5-6 The completed prototype of the wireless ECG monitoring system using a wireless sensor network.

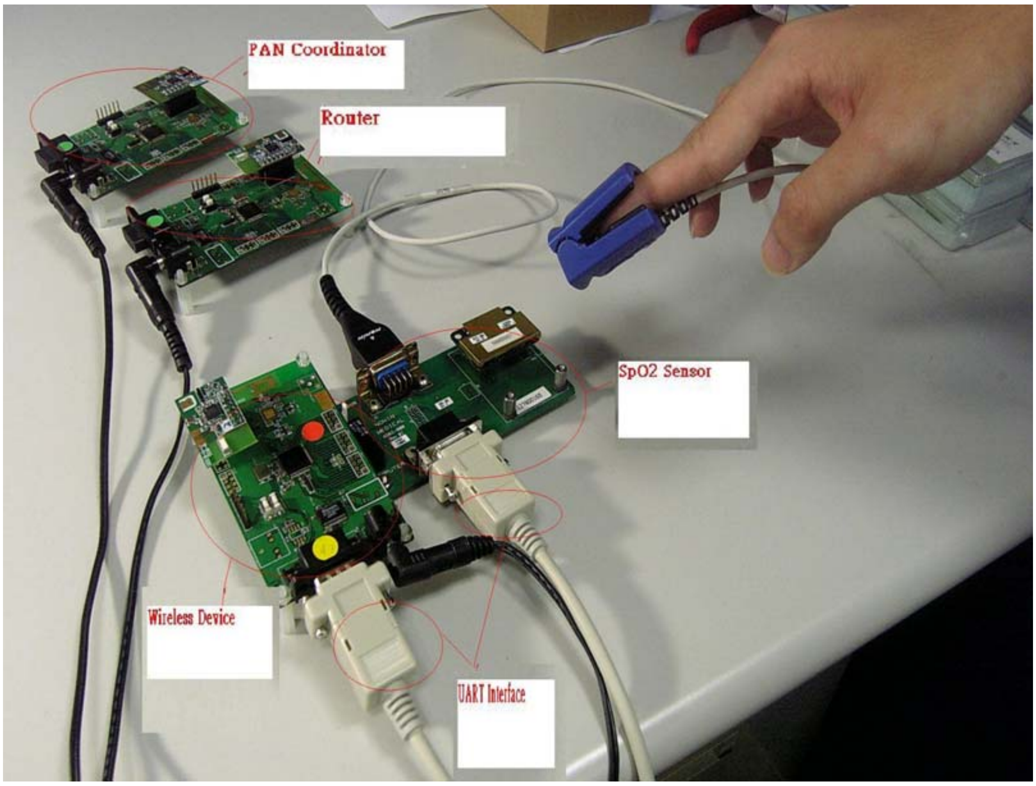
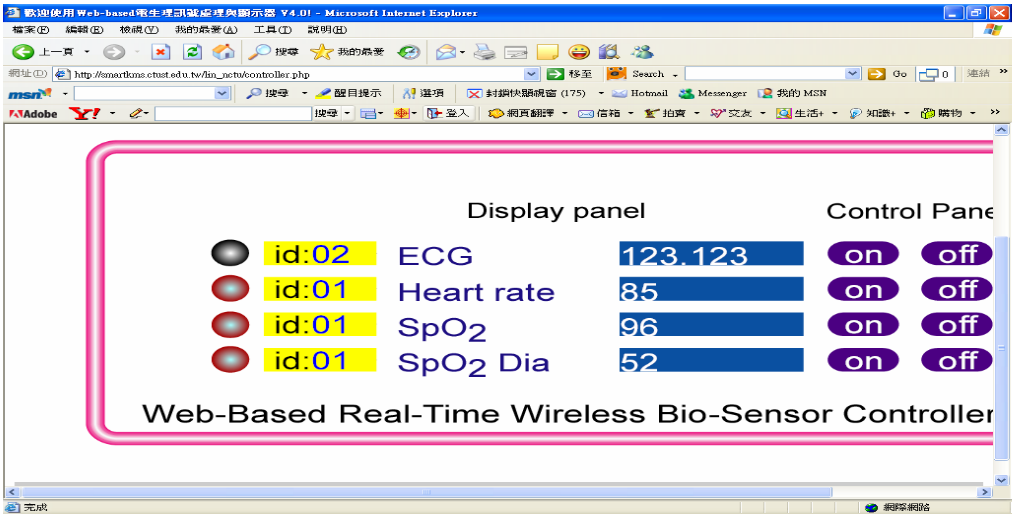


Fig. 5-7 The completed prototype of the wireless SpO$_2$ monitoring system using a wireless sensor network

5.5. Display Panel of the Wireless Sensor Network

The display panel displays information about the patient, including ECG, heart rate and SpO2. A doctor can use a web browser to monitor the health of a patient remotely.



Display panel Control Pane

id:02	ECG	123.123	on	off
id:01	Heart rate	85	on	off
id:01	SpO_2	96	on	off
id:01	SpO_2 Dia	52	on	off

Web-Based Real-Time Wireless Bio-Sensor Controller

Fig. 5-8 the display panel of the wireless sensor network for the home health-care system on a web page

5.6. Packet Sniffing

The accuracy of the firmware code is verified by using a sniffer. This study uses a Chipcon sniffer to capture MAC layer packets. These details are described in Appendix B; the MAC packet format is described in Appendix C.

Chapter 6 Discussion

In the chapter we compare the wireless sensor network with existing research and discuss our advantages and advanced application.

6.1. Comparison with Existing Studies

Many such health-care research projects have employed Bluetooth technology, one popular wireless personal area network. However, Bluetooth has many drawbacks such as limited addressing capability, excessive power consumption, and wake-up time. The wireless sensor network in this study is the ZigBee wireless transmission standard. ZigBee divides address length into 16 bits and 64 bits, where the 16-bit-length address is called network address and is used for short associations, and the 64-bit-length address is used as the MAC address. In this way, the ZigBee network provides addressing space for up to 2^{16} devices and the number of networks addressing space is up to 2^{64} devices.

Tan et al. [1] previously focused on the design and implementation of a home health-care system based on web service, Bluetooth and GSM short message service that only implemented a simple ECG Sensor, a transmitter and a receiver. The authors of that study did not implement wireless sensor network architecture. So its extension and flexibility were very poor for adding others sensor devices into its home health-care system.

The power consumption of Bluetooth is larger than ZigBee. The battery life of ZigBee is 6 month to 2 years approximately -- longer than Bluetooth. Furthermore, to reduce power consumption the sleep mode needed to be implemented. When a specific event occurs the device should been awakened from sleep mode to normal mode. In some applications longer waking time makes the network less sensitive. For this reason the waking time of wireless networks is a very important factor. The waking time of ZigBee requires only 3 ms, while the waking time of Bluetooth is 3 sec. Since the wireless sensor network is very large, its management and querying are very troublesome problems. This study implements a soft wireless sensor gateway to connect the wireless sensor network with the internet by referring related research to the wireless sensor gateway [4]. The soft wireless sensor gateway only sends related data received from the PAN Coordinator of the wireless sensor network to a remote database by using ODBC API. The soft wireless sensor gateway is a PC application that was developed using Visual C++ 6.0. Because PCs are very popular, the soft wireless sensor gateway is easily installed and it is simple to use. The soft wireless sensor gateway does not need any extra hardware; it only needs a PC, so the soft wireless sensor gateway can reduce deployment costs.

For network topology, this study implements a tree topology of three levels using a hierarchy routing method to forward packets to specific devices. This network topology has many advantages, including a fixed number of hops and low transmission latency.

Finally, this study uses the newest web 2.0 concepts to implement a web-based management and control system. This study uses Ajax and SVG technologies to show physiological signal diagrams. Using Ajax technology improves display performance of the web page compared with traditional web pages. Using SVG technology allows vector diagrams to be shown on a web page. The web-management system using Ajax and SVG technologies has many advantages, including being cross-platform, having no space constraints, being simple to use, and having good display performance on the web page. The wireless sensor network with web-based management of this study has several advantages, including high addressing capability, low power consumption, and short waking time.

6.2. Advanced Applications for the Wireless Sensor Network

6.2.1. Advanced Home-Care Applications

If home-care monitoring is applied for long-term conditions, automatic processing of the sensor data is very important. This processing should include extracting specific features for the monitored parameters and providing the necessary alarms. In the future we intend to develop automated screening and analysis methods for different kinds of physiological signals.

6.2.2. Home Automation Applications

For intelligent and comfortable homes, automation is an essential requirement. Home automation includes control of lights, alarms and other electrical appliances. To achieve the home automation we integrate several environmental sensors into our wireless sensor network with web-based management.

We use environmental sensors to detect changes in the home environment and take appropriate action according to these changes.

6.2.3. Security Issues

Since the physiological data of patients is private information, it should be protected from theft. We implement a security algorithm in the network layer to protect this private data.

Chapter 7 Conclusions and Future Research

7.1. Conclusions

This study implements the wireless sensor network with web-based management using ZigBee and web-related technologies including PHP, Java script and SVG.. The wireless sensor network has two types of sensors, the ECG sensor and SpO_2 sensor. When the sensor devices power up, they do network discovery and automatically join a router of the networks according to the results of discovery. The router of the wireless sensor network schedules the sending time of the device that has joined the network and adjusts idle time for each device according to different conditions. When the PAN coordinator receives physiological signals from the routers, it sends them to a PC via the UART interface. On the PC end, this study develops an application that is responsible for receiving the physiological signals from the UART, parsing them to get physiological signals and writing them to a remote database using ODBC API. For the web page this study develops a PHP program that is responsible for polling the remote database to get physiological signal values and showing curve diagrams by using SVG technology.

7.2. Contributions for this Study

The wireless sensor network can be used in home health-care applications. For chronic diseases long-term monitoring is needed. The wireless sensor network can record the necessary physiological signal data continuously, such as ECG, SpO_2 and heart rate. It then store this date in a remote database. Doctors can use a web browser to get physiological data of patients in order to monitor the state of patients with chronic diseases.

Doctors also can view historical data by using a web browser, and then use this data to make diagnosis or predict chronic disease. The wireless sensor network also can be used in home automation related applications. We can add sensors for home automation into the wireless sensor network, and then modify the web page program depending on the requirements of home automation applications. The web page program can provide responses to users according to different measured values of the sensors.

The architecture of the wireless sensor network is very flexible so it can be applied in different applications. One need only add sensors into the wireless sensor network and then modify the web page program according to the requirements of the application.

7.3. Location Estimation

The information about the location of sensors is needed to detect the movement of patients and the positions of patients in the home health-care system. Doctors can use this information to track patients and know the positions of the patients. As shown in recent studies [14], location estimation consists of two types of devices, reference devices and blindfolded devices. The reference device knows its position, and the blindfolded device does not know its position. The latter must use an estimation location algorithm to calculate its location. The estimation location algorithm derives the location of the blindfolded device from the positions of reference devices. In the future, we can implement the estimation location algorithm into our wireless sensor network to determine a patient’s location.

Appendix A.

A.1. NLDE Entity

A.1.1. NLDE-Data Service

The primitive is used to send data frame to specific destination in network layer. If the device is not associated currently, the primitive will issue an NLDE-data-confirm primitive with status of INVALID-REQUEST. If the device is associated device, the NLDE-data -request primitive first constructs an NPDU in order to transmit the supplied NSDU. If the destination address is 0xffff, it represent the data will be sent by using broadcast way, so the broadcast radius and broadcast mode will be considered. Once the NPDU is constructed, the NSDU is routed using routing algorithm, this study only implements hierarchy routing according to neighbor table.

The Source Address Mode (SrcAddrMODE) and Destination Address Mode (DstAddrMode) parameter will be set to 0x02 that indicating the use of 16 bit address. The Source PAN Identifier (SrcPANID) and Destination PAN Identifier (DstPANID) parameters should be set to the current value of MAC PAN ID from the MAC PIB. If the network-wide security level specified in the NIB has a non-zero value and the Security Enable parameter has a value of true, the security processing will be applied to the frame before transmission. Finally, the NLDE-data-request will issue the MCPS-data-request to send the NPDU to destination using routing algorithm and wait for completion of transmission, when the transmission has been completed, the primitive will issue NLDE-data-confirm to report the result of transmission to upper layer. The flowchart of NLME-data-request primitive is shown in Fig. A-1.

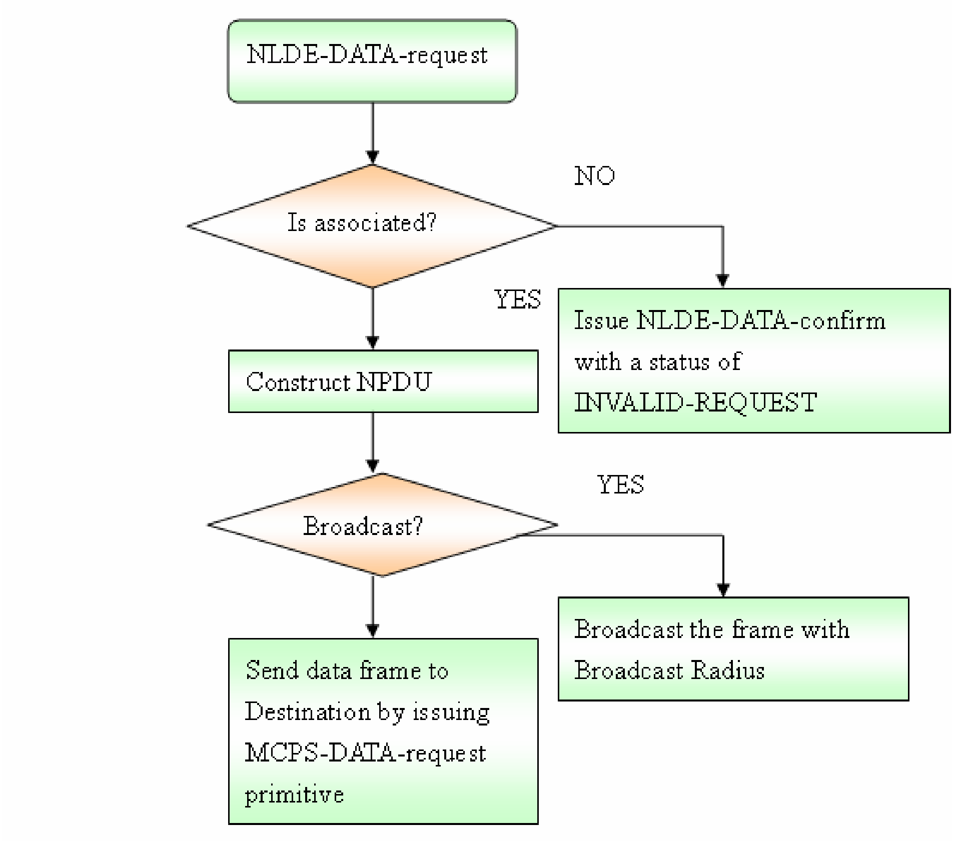


Fig. A-1 Flowchart of data service of network layer

A.2. NLME Entity

The NLME entity consists of a lot of primitives, such as network discovery, network join /leave and others about management related primitives, these is explained later.

A.2.1. NLME-Network-Discovery-Request/Confirm Primitives

The NLME-network-discovery-request primitive allows the next higher layer to request that the NWK layer discover networks currently operating within the personal operating space (POS). The NLME-network-discovery-request is used to find existing device in the POS by issuing MLME-SCAN-request primitive with active scan type parameter. When the active scan has been completed, the scan result will store in PAN descriptor that is used for recording all existing device in the POS. Finally, the NLME-network-discovery-confirm is issued, it will assemble the network descriptor list and calculate network count according to return information of MLME-scan-confirm. The flowchart of NLME-network-discovery service is shown in Fig. A-2.

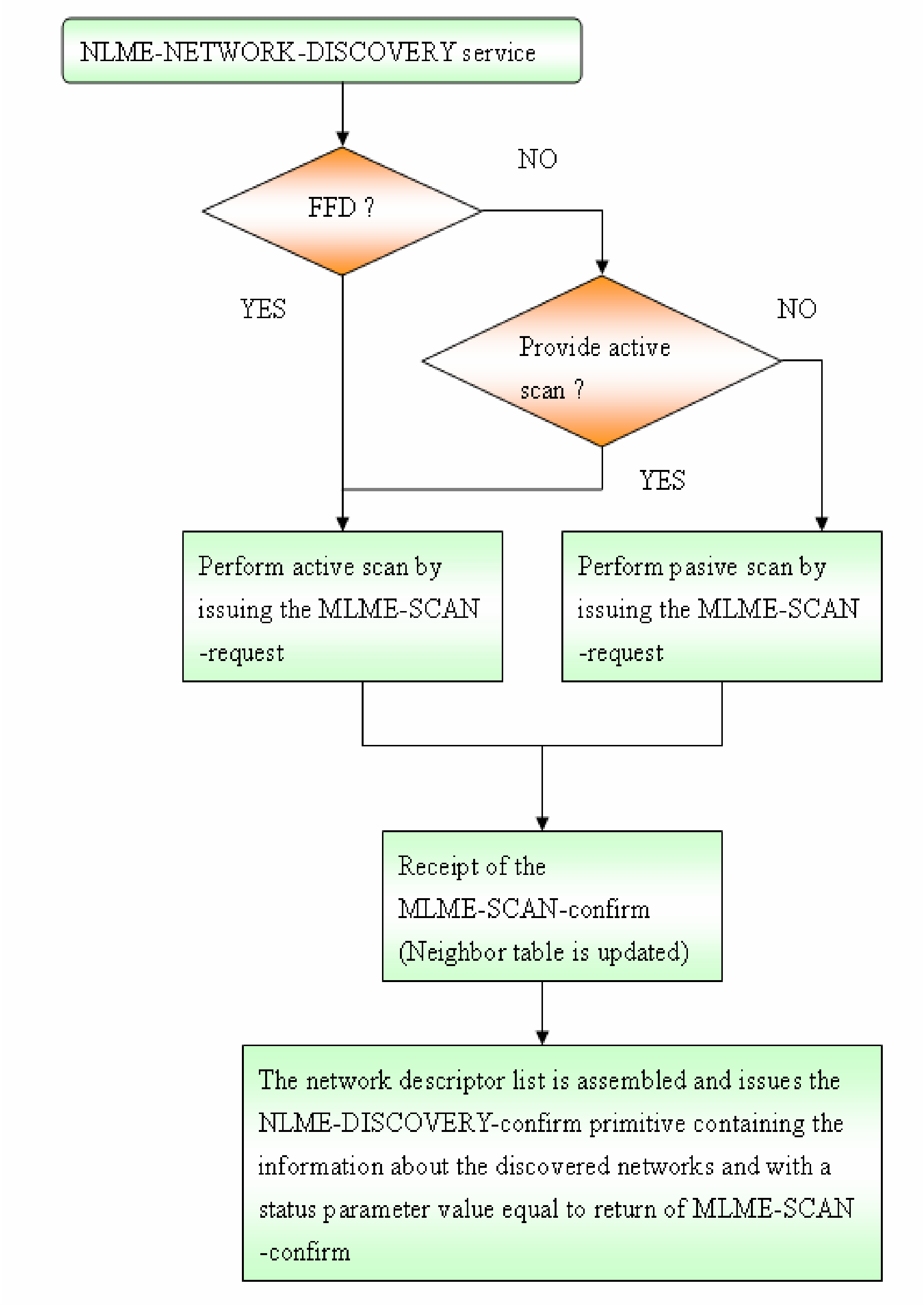


Fig. A-2 Flowchart of network discovery service of network layer

A.2.2. NLME-Network-Formation-Request/Confirm Primitives

The primitive allows the next higher layer (application layer) to request that the device starts a new network with itself as the coordinator. If the device is not capable of being a ZigBee coordinator of a network, the NLME-network-formation-request will issue the NLME-network -formation-confirm primitive with the status parameter set to INVALID_REQUEST. The NLME-network-formation-request primitive first requests the MLME-scan-request with energy detection scan parameter to perform energy detection. On receipt of the result from a successful energy detection scan, it orders the channels according to increasing energy measurement and discards those channels whose energy levels are beyond an acceptable level. And then run active scan by issuing MLME-scan-request primitive with a scan type parameter set to active scan and channel list is set to the list of acceptable channels to search for other ZigBee devices. The NLME-network-formation selects a suitable channel and PAN ID that is not conflict with other existing PANS. Once a suitable channel and PAN ID are determined, the NLME- network-formation will choose 0x0000 as the 16 bit network address, And then the NLME- network-formation will initialize the new super-frame configuration by issuing the MLME-start-request primitive with PAN coordinator parameter is be set to true, beacon order and super-frame order will be set as same those given to the NLME-network-formation-request.

Finally, the NLME-network-formation-request will issues the NLME- network-formation- confirm to notify next higher layer of the status of request with same status returned from the MLME-start-confirm primitive, the flowchart of NLME network formation is shown in Fig. A-3.

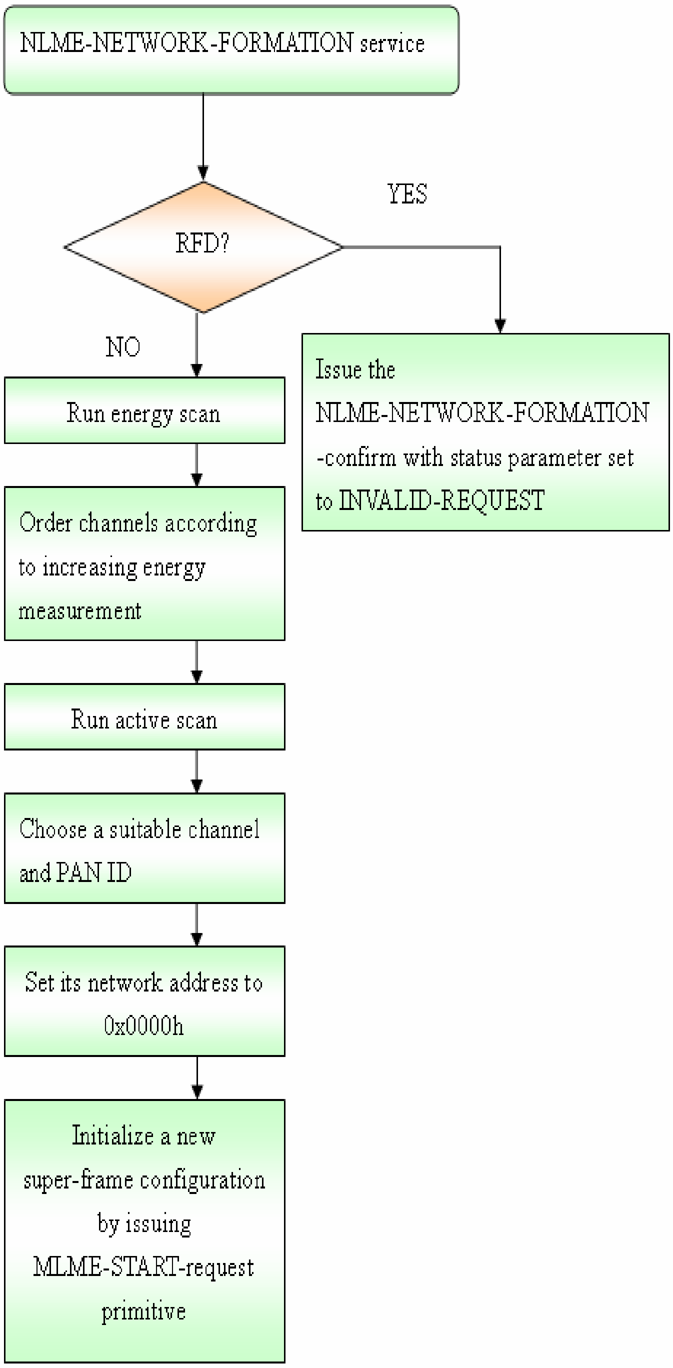


Fig. A-3 Flowchart of network formation in network layer

A.2.3. NLME-Permit-Joining-Request/Confirm Primitives

These primitives are used to define how the next higher layer of a ZigBee coordinator or router can request that devices be permitted to join its network. If the next layer of a ZigBee coordinator or router wants to define a fixed period during which it may accept devices onto its network by set its MAC layer association permit flag, it can issues the NLME- network-permit-join-request primitive with permit duration parameter. On receipt of the NLME-permit-join-request by the ZigBee end device, the NLME-permit-join-confirm returns a status of INVALID REQUEST.

If the permit duration parameter of the NLME-permit-joining-request is set to 0x00, the NLME sets the MAC PIB attribute, The MAC association permit variable is set to false by issuing the MLME-SET-request. If the permit duration parameter of the NLME -permit-joining-request is set to 0xff, the NLME sets the MAC PIB attribute, The MAC association permit variable is set to true by issuing the MLME-SET-request. If the permit duration parameter of the NLME-permit-joining-request is set to in range of 0x01 to 0xfe, the NLME sets the MAC PIB attribute, MAC association permit to true by issuing the MLME-SET-request, and then the NLME-permit-joining starts a timer to expire after permit duration seconds. If the timer has been set, the NLME will return a status of NLME by issuing the NLME-permit-joining-confirm. If the timer has expired, the NLME-permit-joining will sets MAC association permit to false by issuing the MLME-SET primitive, the flowchart of NLME permit join service is shown in Fig. A-4.

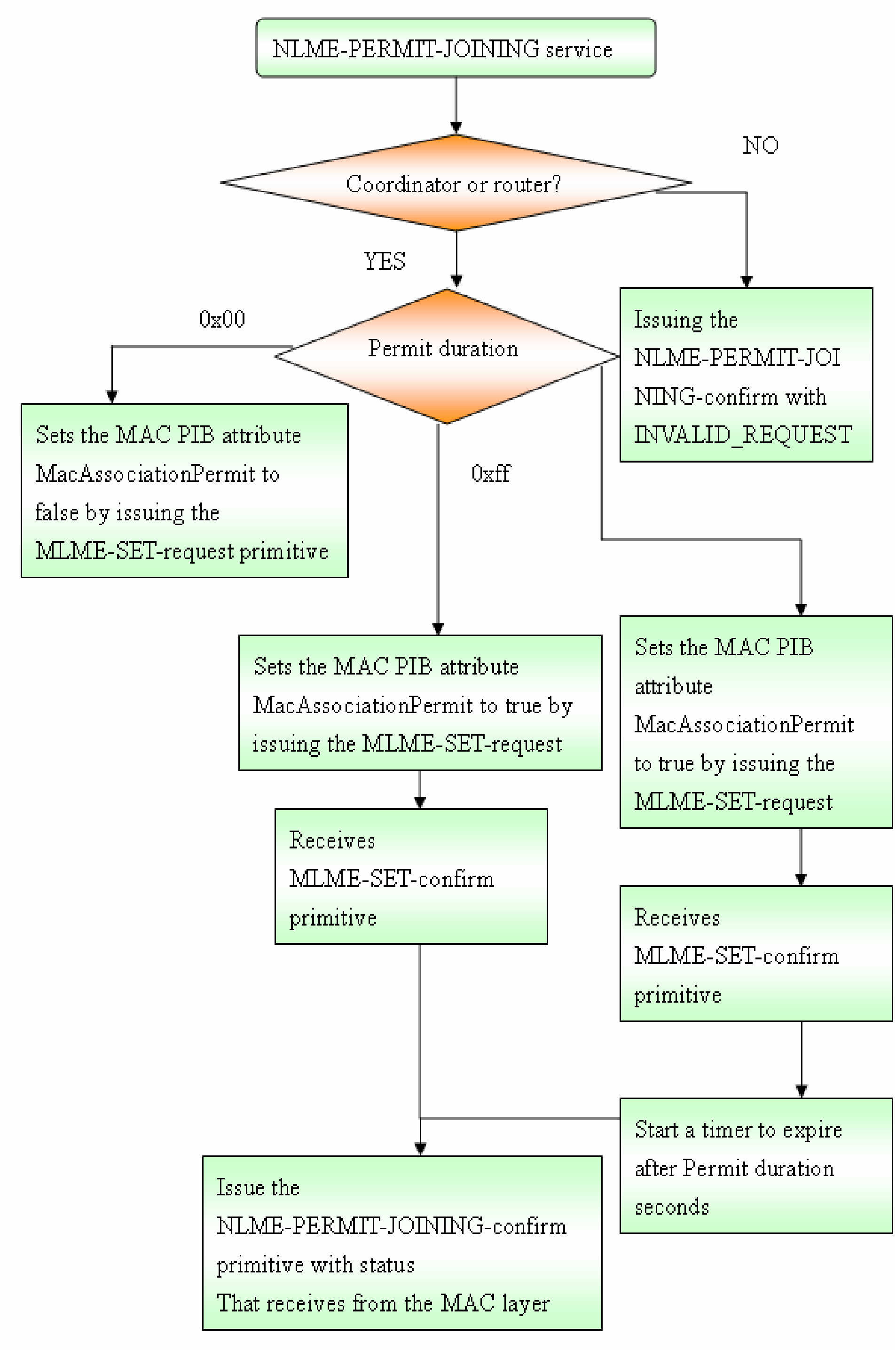


Fig. A-4 flowchart of network permits joining service

A.2.4. NLME-JOIN-Request/Confirm Primitives

These primitive is used to generate a request to join a new network using MAC layer association procedure and report status of joining. If the device is currently joined to a network, the NLME issues a NLME-join-confirm primitive with the status parameter set to invalid request. If the device is not currently joined to a network, the NLME attempts to join the network specified by the PAN ID parameter.

If the rejoin network parameter is false, and Join as router parameter is false, the NLME issues an MLME-associate-request with its coordinator address parameter set to the address of a router in its neighbor table. If the rejoin network parameter is false and Join as router parameter is true, the Device will function as a ZigBee router in the network. If the device is not joined to a network and the rejoin network parameter is equal to true, and then run orphan scan by issuing an MLME-scan-request with the scan type parameter set to indicate an orphan scan. Finally, the NLME will issues the NLME-join-confirm with status is equal to status of MLME-scan-confirm, the flowchart of NLME join service is shown in Fig. A-5.

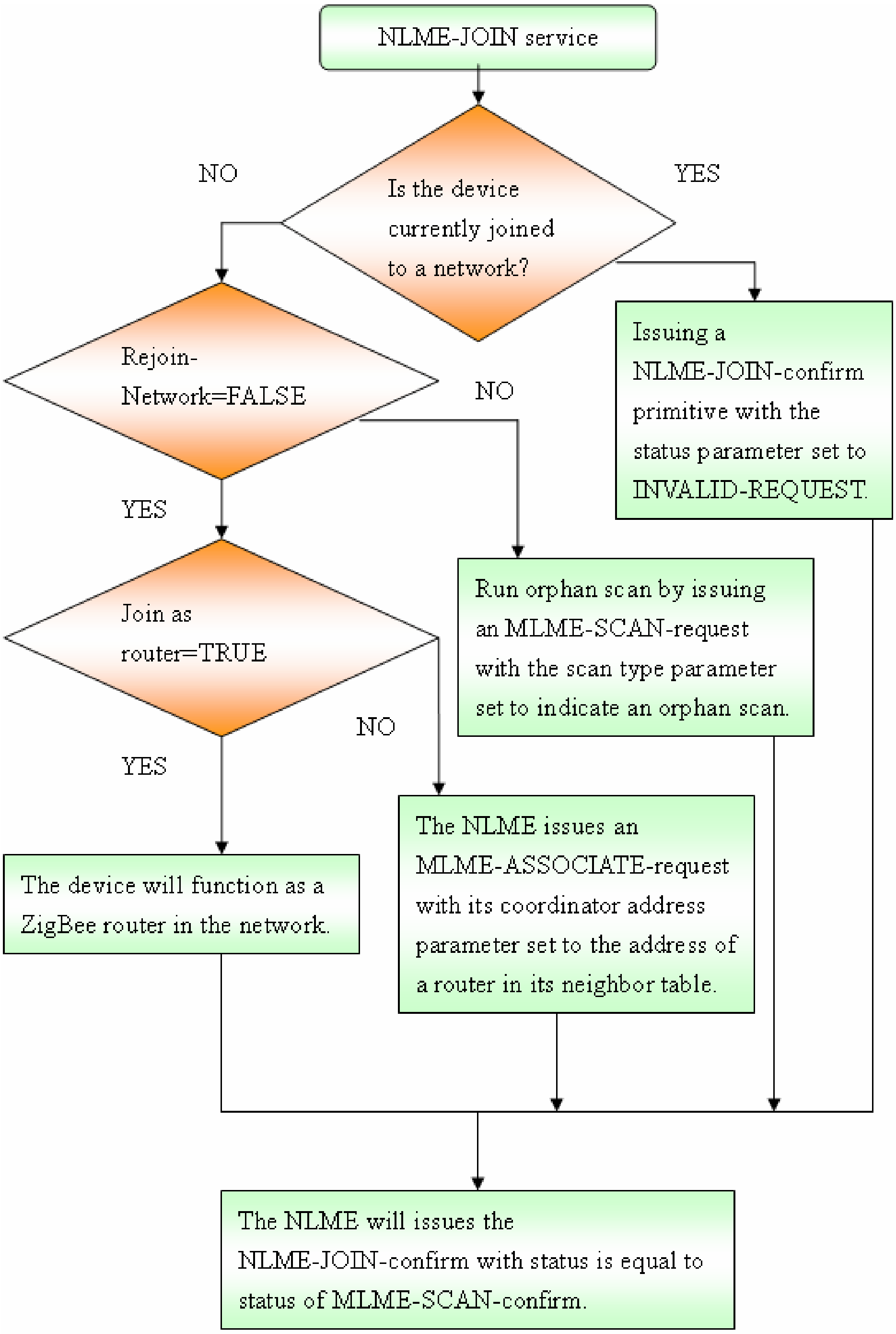


Fig. A-5 Flowchart of network joining service

A.2.5. NLME-JOIN-Indication Primitives

The primitive is used to inform the next higher layer of ZigBee coordinator or router that a new device has successfully joined it network by association. The primitive is issued by MLME-association-indication MAC layer to inform network layer that a new device has join it network by MAC Layer association.

A.2.6. The Message Flow of Establishing a New Network

The procedure of establishing a network is completed by using NLME- network- formation-request primitive. If the device is ZigBee Coordinator capable and not currently joined to a network, the procedure allows the device to establish a new network. If the device is no ZigBee coordinator capable or currently joined to a network, the procedure is not allows the device to establish a new network. The procedure first performs an energy scan over specified channels by requesting MLME-scan-request with the scan type parameter set to energy detection scan. After the energy detection scan has been completed successfully, the procedure will order the channels according to increasing energy measurement and choose an acceptable channel. And the procedure performs active scan to search other existing ZigBee devices by issuing the MLME-scan-request with the scan type parameter set to active scan. After the active scan has been completed successfully, the procedure will choose a PAN ID according to parameter of primitive or random number that are must not conflict with existing PAN ID. Once a PAN ID is selected, the procedure will select a 16 bit network address equal to 0x0000 and set the MAC short address of PIB attribute in the MAC layer. Once a network is selected, the NLME will begin operation of the new PAN by issuing the MLME-start -request primitive, the flowchart of network formation is shown in Fig. A-6.

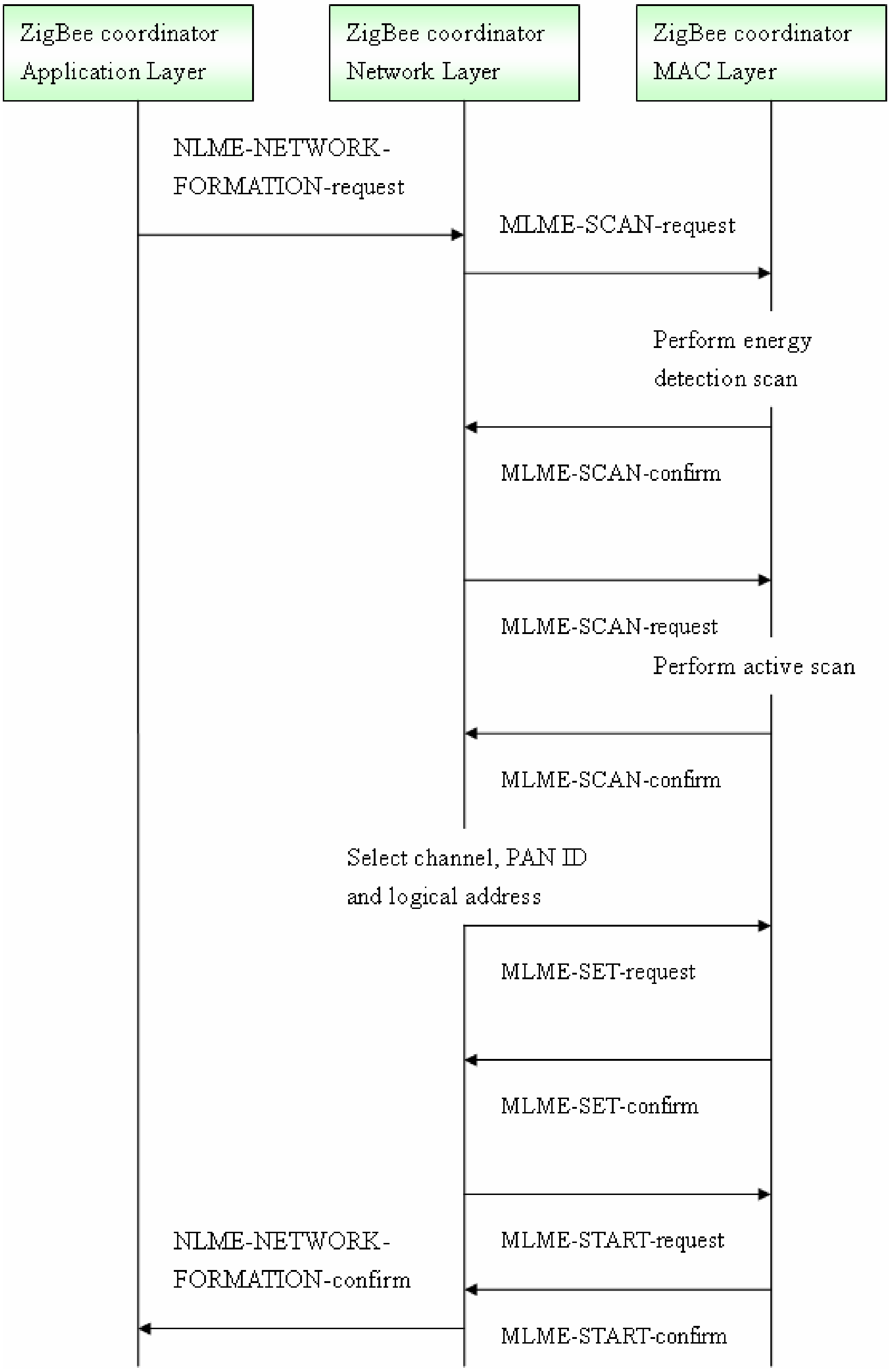


Fig. A-6 Message flow chart of establishing a new network

A.2.7. The Message Flow of Joining a Network through Association

The procedure for joining a network using the MAC layer association is first to do network discovery by issuing NLME-network-discovery-request primitive. The NLME -network-discovery-request primitive is to research existing devices in POS by issuing the MLME-scan-request with scan type set to active scan. Once the MLME-scan-request primitive has been completed, the procedure will issue the MLME-scan-confirm to inform result of network layer. The procedure issue the NLME-discovery-confirm primitive to inform application layer of discovery result. Once the network discovery has been completed, the procedure will join a network according to existing PAN of discovery result by issuing NLME-join -request with the PAN ID of the desired network. If the device is already joined a network, the procedure will terminate the procedure and notify the application layer of the illegal request by issuing the NLME-JOIN-confirm primitive with status parameter set to invalid request. If the device is not already joined a network, the procedure will join the desired network by issued the MLME-associate-request primitive with address parameter. Once the association procedure has been completed, the procedure will get result of association through MLME-associate-confirm. If the device can not join the PAN successfully, the procedure will terminate the procedure by issuing the NLME-join-confirm with the status parameter set to the value returned in the last received MLME- associate-confirm primitive. If the device can join the PAN successfully, the procedure will get a 16 bit logical address through MLME-associate-confirm primitive. Once the device has joined the network successfully and the application layer has issued a NLME- start-router-request primitive to setup its super-frame configuration and begin transmitting beacon frames. The NLME-start-router-request primitive setup its super-frame configuration by issuing the MLME-start-request, the message flow chart of joining a network through association procedure is shown in Fig. A-7.

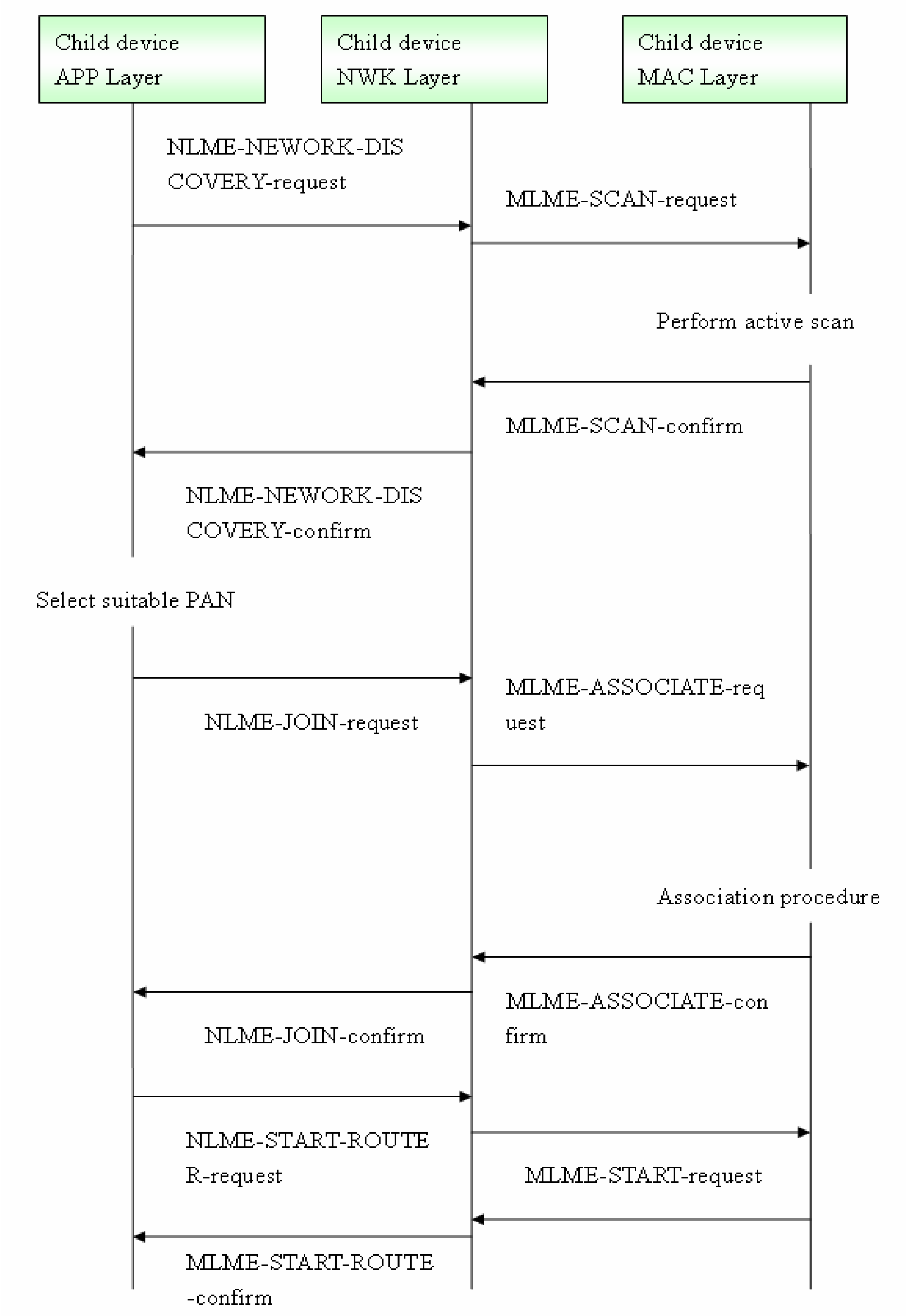


Fig. A-7 Message flow chart for network discovery.

A.2.8. NLME-LEAVE-Request/Confirm Primitives

The set of primitives define how the next layer of a device can request to leave or request that another device leaves a network if the device is not currently joined to a network, the NLME-leave-confirm primitive with a status of invalid request. If the device is currently joined to a network and with the device address parameter equal to NULL, the NLME issues the MLME-disassociate-request primitive. The NLME issues the NLME-leave-confirm primitive with status equal to status of MLME-disassociate-confirm primitive. And the NLME clear its routing table entries. If the issuing device is ZigBee coordinator or Router and the device address parameter is not equal to NULL, the NLME searches its neighbor table whether the specified device exists. If the specified device doesn’t exist, the NLME issues the NLME -leave-confirm primitive with a status of unknown device. If the specified device exists, the NLME removes it entry in the neighbor table and issues the MLME -disassociate-request primitive, on receipt of the corresponding MLME-disassociate-confirm primitive, the NLME issues the NLME-leave-confirm primitive with status returned from the MLME -disassociate-confirm primitive, the flowchart of NLME leave service is shown in Fig. A-8.

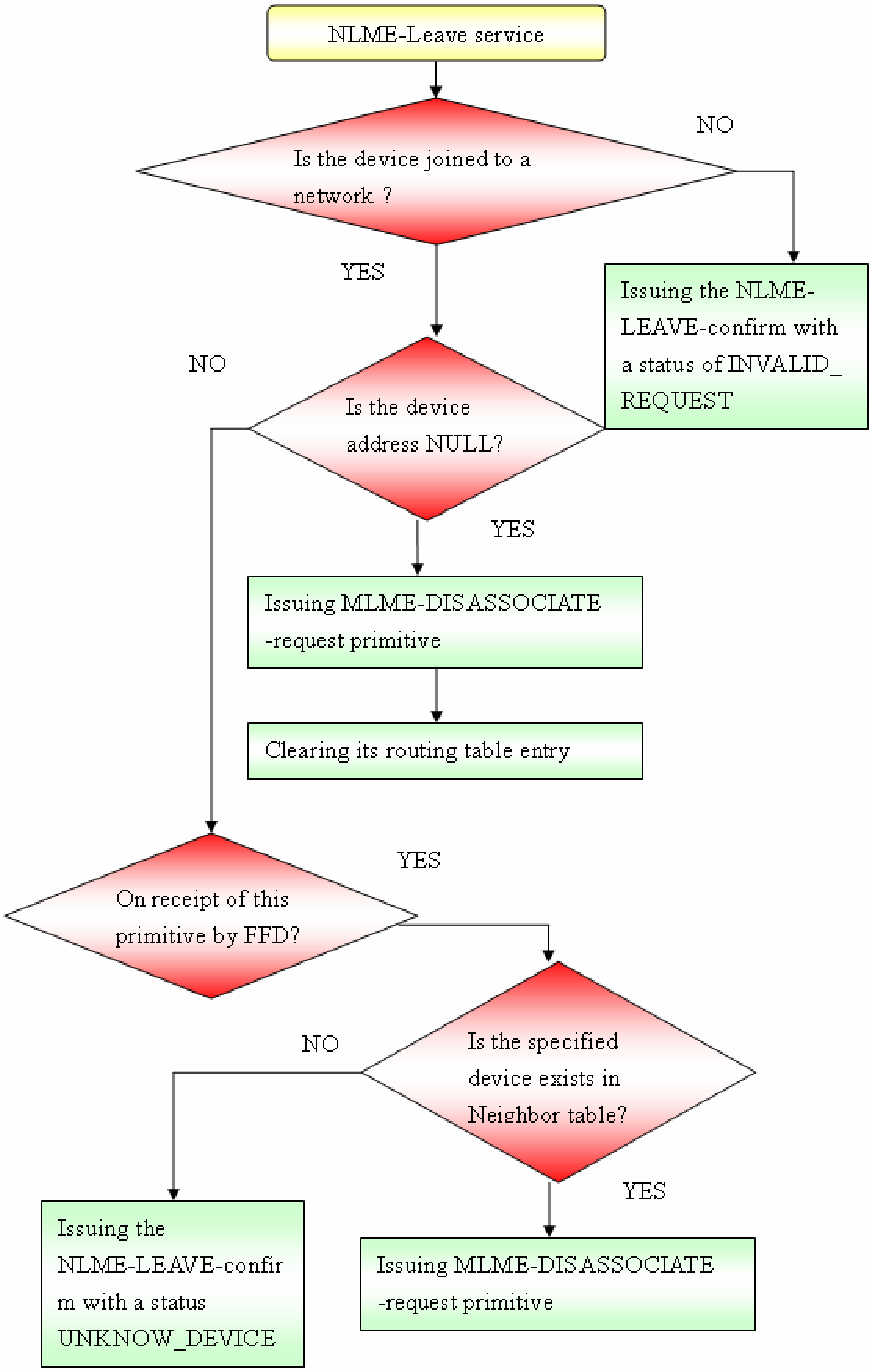


Fig. A-8 Flowchart for network leave

A.2.9. NLME-LEAVE-Indication Primitive

The primitive is used to inform application layer of coordinator or router of successful exit of one of that device’s associated children from the network. The primitive is also used to inform application layer of router or end device that the device has been successfully removed from the network by its associated router or coordinator.

A.2.10. Child Initiates its own Removal from a Network

If a child device wants to leave the network, the procedure will issues the NLME- leave-request primitive with the device address parameter set to null. If the child device is not joined the network currently, the procedure will terminate the Procedure and issues the NLME-leave-confirm with the status parameter set to invalid request. If the child device is joined the network currently, the procedure will first issues MLME- disassociate-request to send disassociated command to its parent device. The status report of MLME -disassociate-request is communicated back via the MLME-disassociate-confirm primitive. On receipt of the status of the disassociation, the procedure will notify network layer of the disassociation status by issuing the NLME-leave-confirm with the status parameter set to the status value returned in the MLME-disassociate-confirm primitive, the message flow chart for a child to initiate its own removal from a network is shown in Fig. A-9.

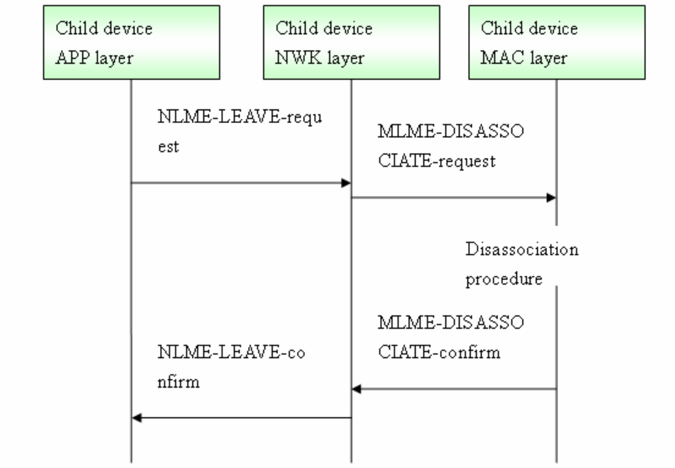


Fig. A-9 Message flow chart for a child device to initiate its own removal from a network

A.2.11. Parent Device Handles a Leave Request from its Child

When the parent device receives a disassociation request command, it will issue the MLME-DISASSOCIATE-indication primitive to inform network layer that a child device want to leave its network. When the network layer of parent device receives the Disassociation indication, the NLME will search its neighbor table in order to determine whether a child device can be found. If the child device is not found, the NLME will terminate the procedure. If the child device is found, the NLME will remove the appropriate entry from its neighbor table and inform the network layer that the child device has been removed by issuing the NLME-LEAVE-indication primitive, the message flow chart for parent to handle a leave request from its child is shown in Fig. A-10.

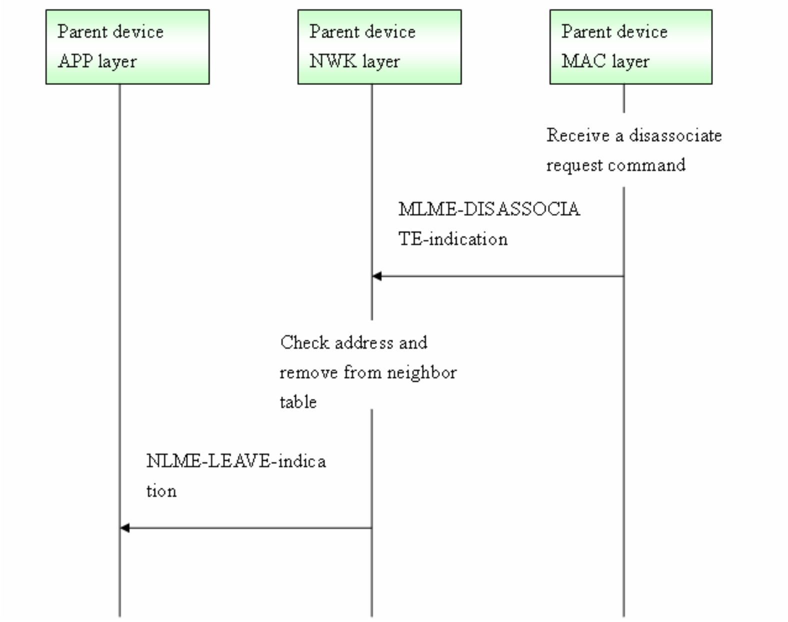


Fig. A-10 Message flow chart for parent to handle a leave request from its child

A.2.12. Parent Device Forces a Child Device Leaves its Network

The procedure for a parent device to remove a child leave its network by issuing the NLME-leave-request primitive with device address parameter set to the address of the device to be removed from the network. If the procedure is initiated on end device, the procedure will terminate the procedure and notify the network layer of the illegal request by issuing the NLME-leave-confirm primitive with the status parameter set to invalid-request. If the procedure is initiated on coordinator or router, the NLME first search its neighbor table in order to determine whether the specified device can be found. If the specified device is not found, the NLME will terminate the procedure and inform network layer of the unknown device by issuing the NLME-leave-confirm primitive with the status parameter set to unknown device. If the specified device is found, the NLME will remove the appropriate entry from the neighbor table and perform a disassociation procedure.

The status of disassociation procedure is communicated back via the MLME- disassociate -confirm primitive. On receipt of the results from the disassociation procedure, the NLME will inform the network layer of the status of its request to remove the device from the network by issuing the NLME-leave-confirm primitive with the status parameter set to the status returned in the MLME-disassociate-confirm primitive, the message flow for a parent device to force a child device leaves its network is shown in Fig. A-11.

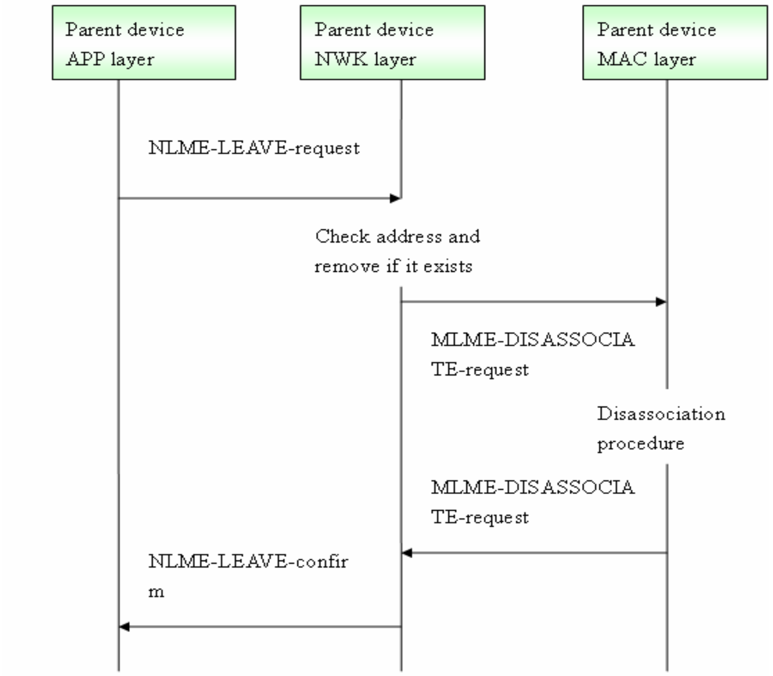


Fig. A-11 Message flow chart for a parent to force a child from its network.

A.2.13. Child Removes itself from a Network at the Request of its Parent

When the child device receives a disassociation command, the NLME will issues the NLME-disassociate-indication to inform the network layer that the child device is forced to leave its network. The NLME first compare 64 bit extended address with the extended address of disassociation request command. If the two addresses are same, the child device will inform the network layer that it always removed from the network by issuing the NLME-leave-indication. If the two addresses are not the same, the NLME will terminate the procedure, the message flow for a child to remove itself from a network at the request of its parent is shown in Fig. A-12.

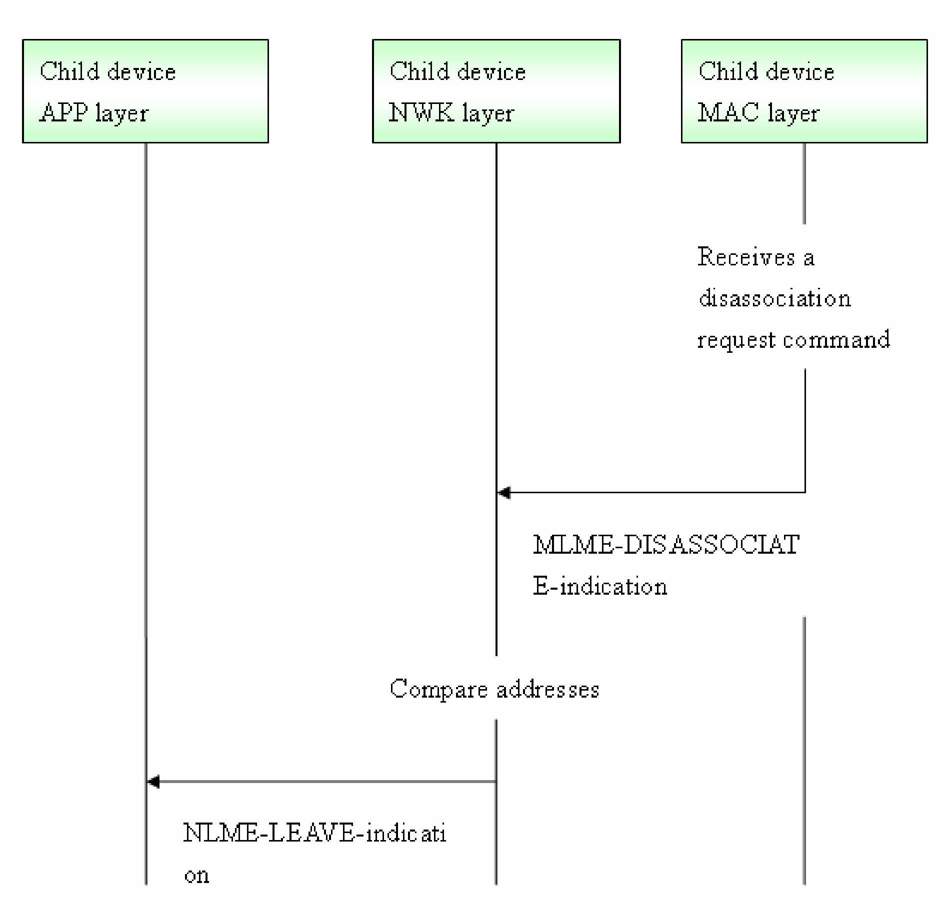


Fig. A-12 Message flow chart for a child to remove itself from a network

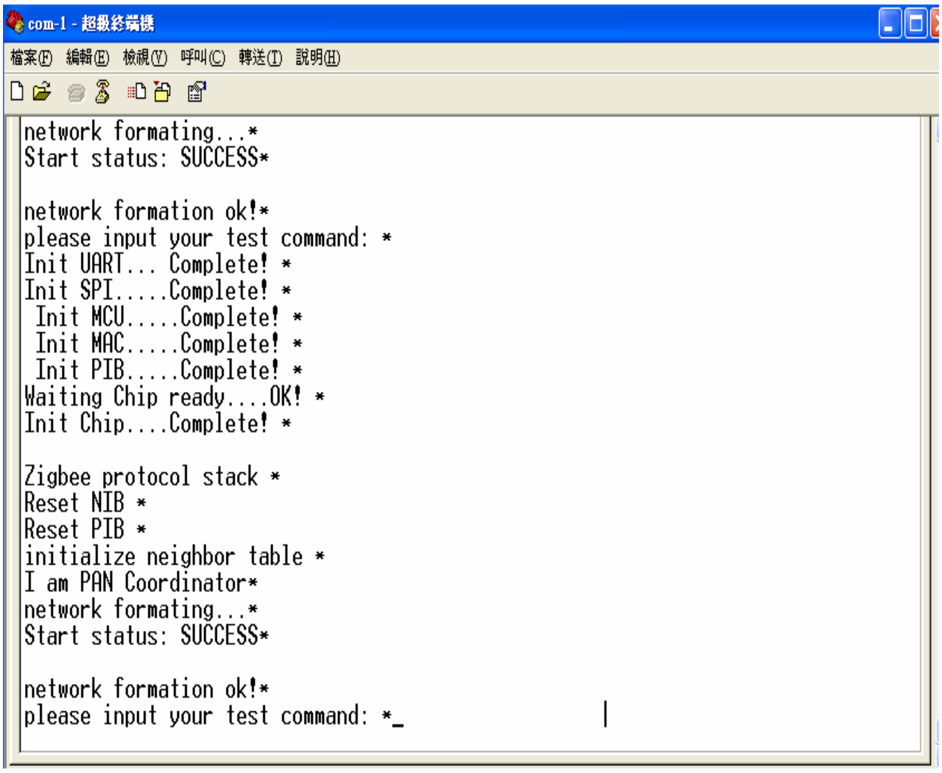
Appendix B.

To verify correctness of firmware code, the study uses the Chipcon sniffer to capture IEEE 802.14.4 MAC packets.

B.1. PAN Coordinator

B.1.1. Hyper-Terminal

The PAN coordinator is responsible for creating a network and wait for join of routers, it status of test can be seen in hyper-terminal is shown in Fig. B-1.

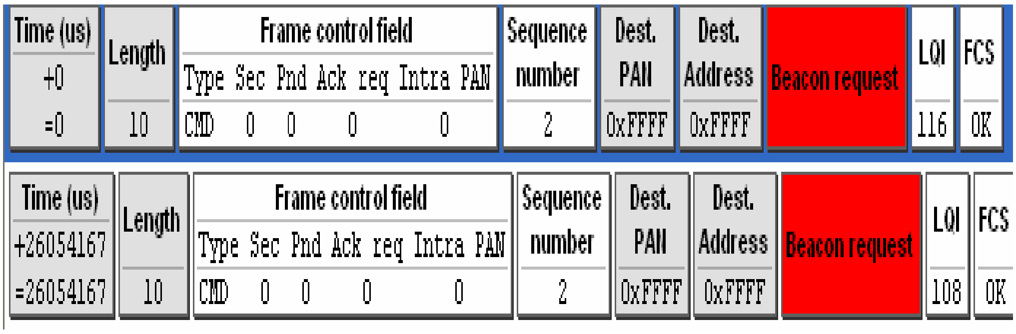


```
network formating...*
Start status: SUCCESS*

network formation ok!*
please input your test command: *
Init UART... Complete! *
Init SPI.....Complete! *
 Init MCU.....Complete! *
 Init MAC.....Complete! *
 Init PIB.....Complete! *
Waiting Chip ready....OK! *
Init Chip....Complete! *

Zigbee protocol stack *
Reset NIB *
Reset PIB *
initialize neighbor table *
I am PAN Coordinator*
network formating...*
Start status: SUCCESS*

network formation ok!*
please input your test command: *_
```

Fig. B-1 Status of PAN coordinator is shown in hyper-terminal

B.1.2. Frames Measurement

The PAN Coordinator first starts a new network by issuing NLME-NETWORK -FORMATION-request primitive, which broadcasts a beacon request command to all devices in Personal Operating Space (POS). We can see the following packet diagram, the Dest Address represents destination address is equal to 0xffff that representing is to use broadcast way to send the packet. When PAN Coordinator or Router receives the beacon request command, it will broadcast a beacon frame that contains the PAN and device information. The PAN Coordinator can get existing PAN ID to prevent new PAN ID from conflicting with existing PAN ID, Measurement of frames is shown in Fig. B-2.



Time (us)	Length	Frame control field					Sequence number	Dest. PAN	Dest. Address	Beacon request	LQI	FCS
+0		Type	Sec	Pnd	Ack req	Intra PAN						
=0	10	CMD	0	0	0	0	2	0xFFFF	0xFFFF		116	OK

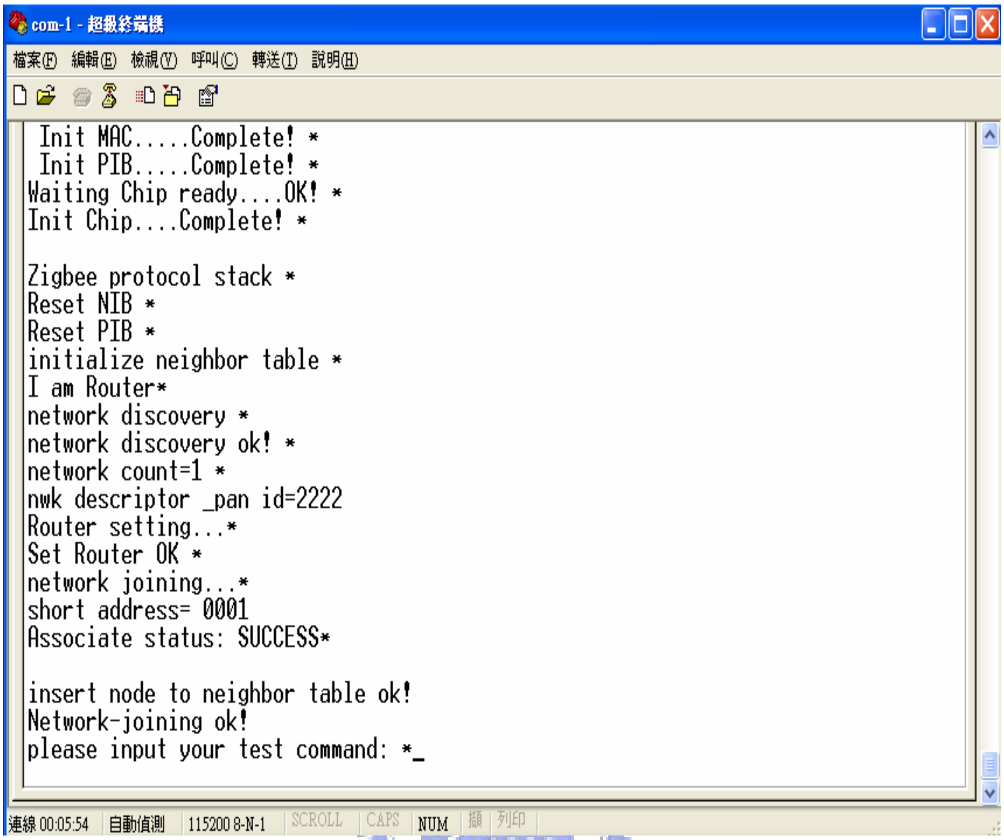
Time (us)	Length	Frame control field					Sequence number	Dest. PAN	Dest. Address	Beacon request	LQI	FCS
+26054167		Type	Sec	Pnd	Ack req	Intra PAN						
=26054167	10	CMD	0	0	0	0	2	0xFFFF	0xFFFF		108	OK

Fig. B-2 Frames Measurement for PAN coordinator using Chipcon sniffer

B.2. Router

B.2.1. Hyper-terminal

The router is responsible for scheduling sending time of associated device and forwarding data that received from associated devices to PAN Coordinator by using hierarchy routing according its neighbor table. The router first does network discovery to search existing PAN Coordinator in POS, and then select a PAN Coordinator to join its PAN. In following examples, the router searches a PAN; its PAN ID is 2222 and joins. Its PAN. It status of router can be seen in hyper-terminal is shown in Fig. B-3.

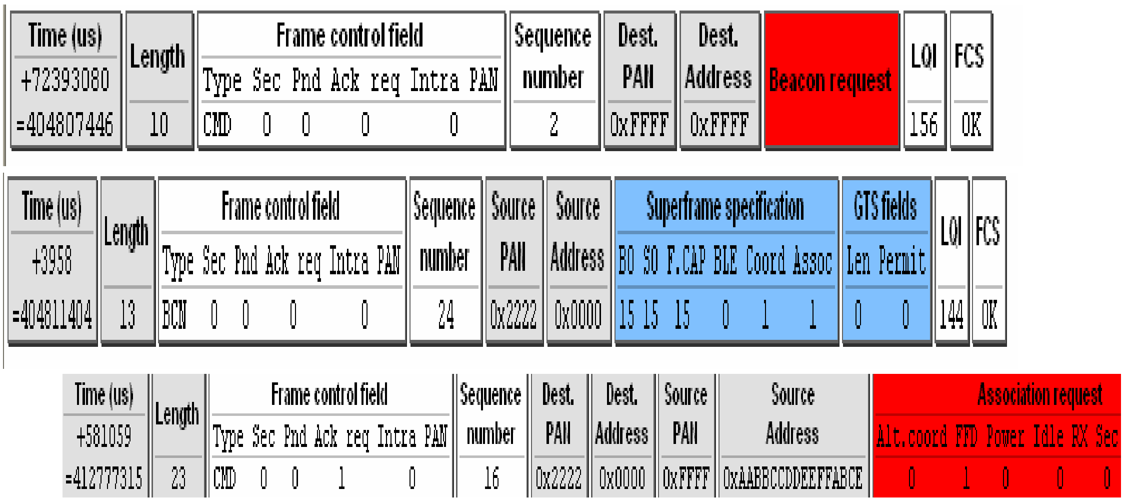


```
 Init MAC.....Complete! *
 Init PIB.....Complete! *
Waiting Chip ready....OK! *
Init Chip....Complete! *

Zigbee protocol stack *
Reset NIB *
Reset PIB *
initialize neighbor table *
I am Router*
network discovery *
network discovery ok! *
network count=1 *
nwk descriptor _pan id=2222
Router setting...*
Set Router OK *
network joining...*
short address= 0001
Associate status: SUCCESS*

insert node to neighbor table ok!
Network-joining ok!
please input your test command: *_
```

Fig. B-3 Status of router is shown in hyper-terminal

B.2.2. Frames Measurement

The router first broadcasts a beacon request command to existing PAN coordinators in POS, and get a beacon frame from PAN coordinator with PAN ID is equal to 0x2222. We can know the device is PAN coordinator according to Coord field of the beacon frame's super-frame specification is equal to 1. Since Assoc field of the beacon frame's super-frame specification is equal to 1, the PAN coordinator allows other devices to join it. The router joins the PAN coordinator by using the NLME-JOIN-Request primitive, which issues the MLME-ASSOCIATE-REQUEST primitive that sends association request command to the PAN coordinator. Measurement of frames is shown in Fig. B-4 and Fig. B-5.

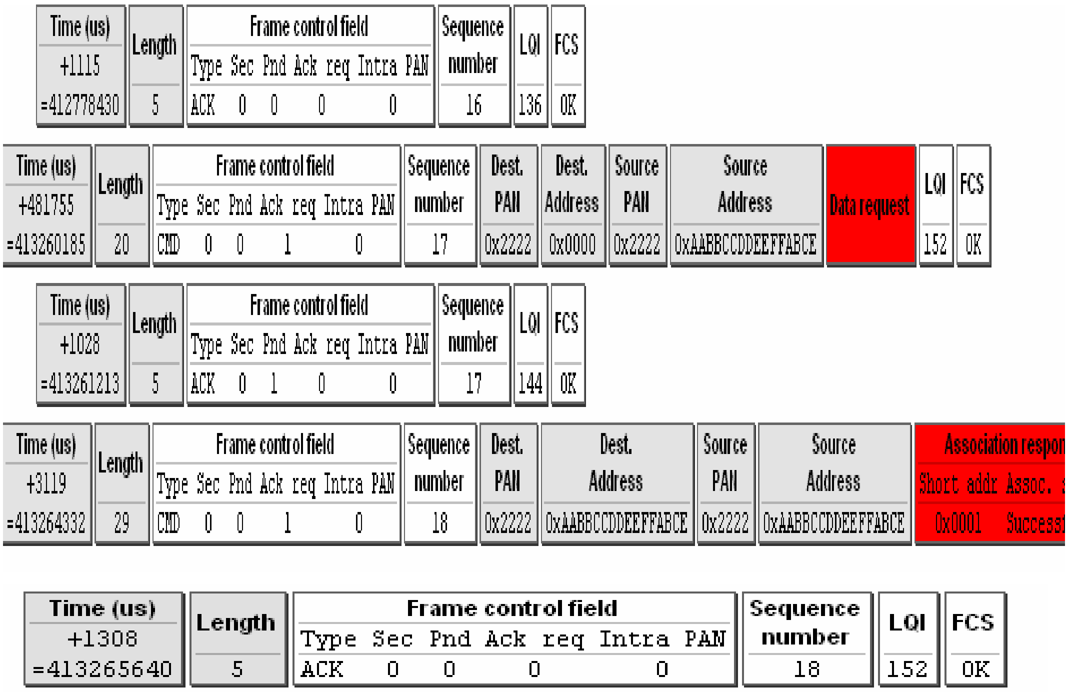


Time (us)	Length	Frame control field Type Sec Pnd Ack req Intra PAN	Sequence number	Dest. PAN	Dest. Address	Beacon request	LQI	FCS
+72393080 =404807446	10	CMD 0 0 0 0	2	0xFFFF	0xFFFF		156	OK

Time (us)	Length	Frame control field Type Sec Pnd Ack req Intra PAN	Sequence number	Source PAN	Source Address	Superframe specification BO SO F.CAP BLE Coord Assoc	GTS fields Len Permit	LQI	FCS
+3958 =404811404	13	BCN 0 0 0 0	24	0x2222	0x0000	15 15 15 0 1 1	0 0	144	OK

Time (us)	Length	Frame control field Type Sec Pnd Ack req Intra PAN	Sequence number	Dest. PAN	Dest. Address	Source PAN	Source Address	Association request Alt.coord FFD Power Idle RX Sec
+581059 =412777315	23	CMD 0 0 1 0	16	0x2222	0x0000	0xFFFF	0xAABBCCDDEEFFABCE	0 1 0 0 0

Fig. B-4 Frames measurement for router using Chipcon sniffer



Time (us)	Length	Frame control field Type Sec Pnd Ack req Intra PAN	Sequence number	LQI	FCS
+1115 =412778430	5	ACK 0 0 0 0	16	136	OK

Time (us)	Length	Frame control field Type Sec Pnd Ack req Intra PAN	Sequence number	Dest. PAN	Dest. Address	Source PAN	Source Address	Data request	LQI	FCS
+481755 =413260185	20	CMD 0 0 1 0	17	0x2222	0x0000	0x2222	0xAABBCCDDEEFFABCE		152	OK

Time (us)	Length	Frame control field Type Sec Pnd Ack req Intra PAN	Sequence number	LQI	FCS
+1028 =413261213	5	ACK 0 1 0 0	17	144	OK

Time (us)	Length	Frame control field Type Sec Pnd Ack req Intra PAN	Sequence number	Dest. PAN	Dest. Address	Source PAN	Source Address	Association respon Short addr Assoc. s
+3119 =413264332	29	CMD 0 0 1 0	18	0x2222	0xAABBCCDDEEFFABCE	0x2222	0xAABBCCDDEEFFABCE	0x0001 Successf

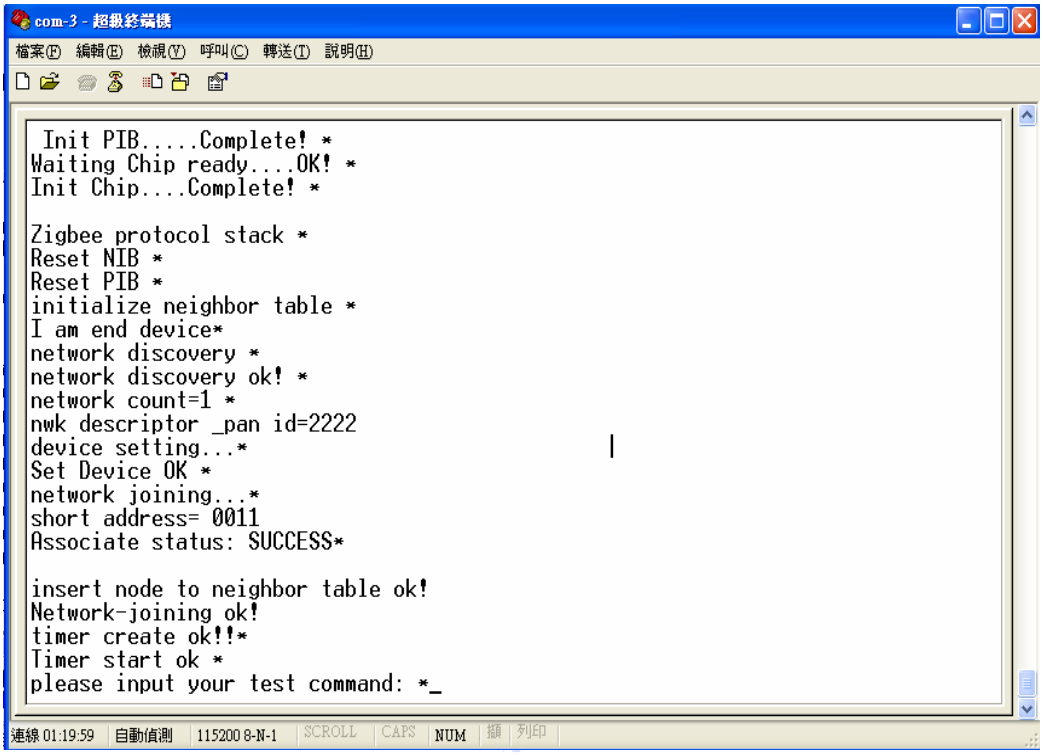
Time (us)	Length	Frame control field Type Sec Pnd Ack req Intra PAN	Sequence number	LQI	FCS
+1308 =413265640	5	ACK 0 0 0 0	18	152	OK

Fig. B-5 Frames measurement for router using Chipcon sniffer

B.3. Device

B.3.1. Hyper-Terminal

The Device is responsible for physiological data to associated router by issuing the NLME-DATA-request, which issues MLME-DATA-request to send data frame to the router. The device first does network discovery to search existing routers in POS, and then select a router to join its PAN. In following examples, the device searches a router; its PAN ID of router is 2222 and joins the router. Its status of router can be seen in hyper-terminal is shown in Fig. B-6

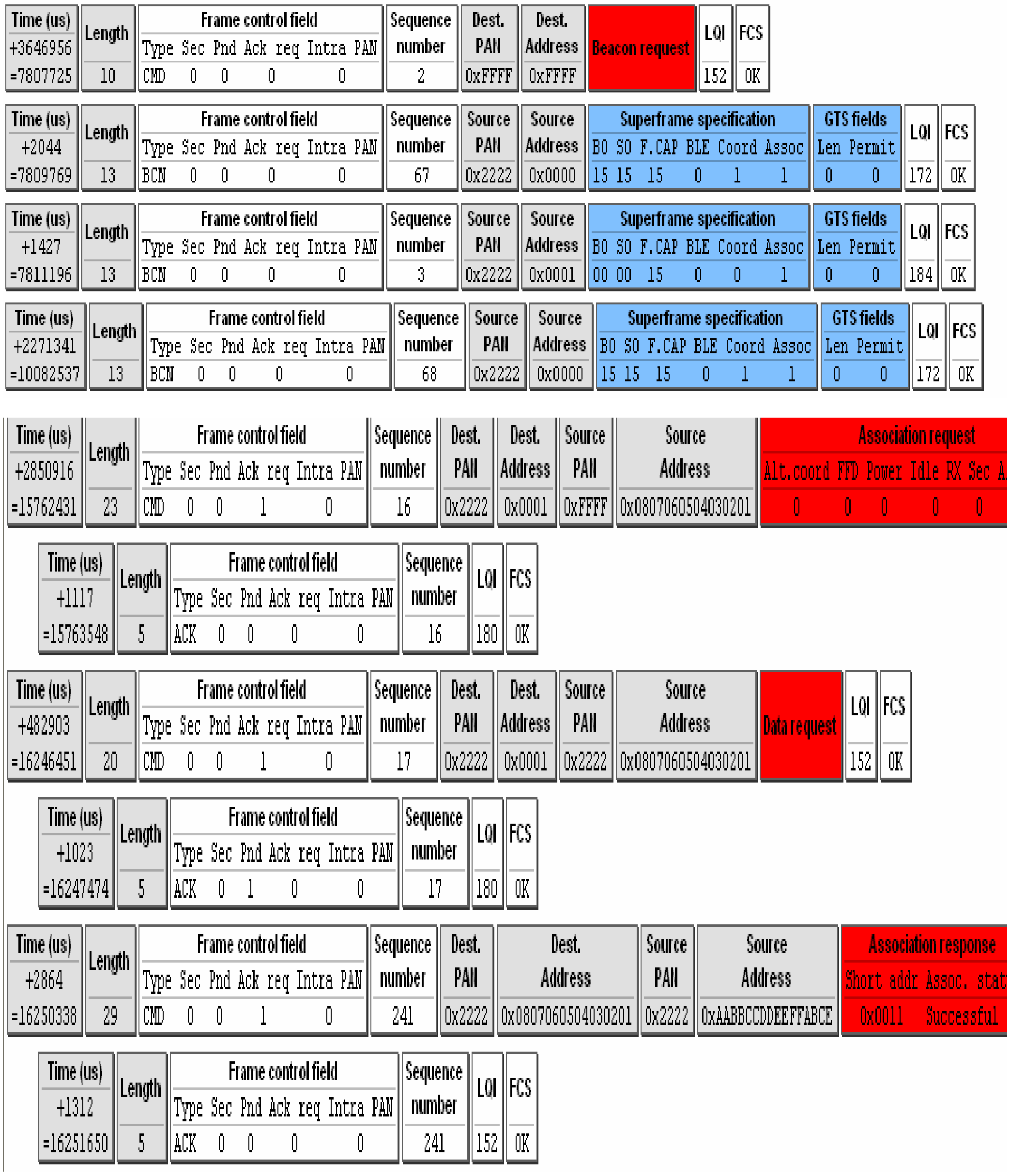


```
 Init PIB.....Complete! *
Waiting Chip ready....OK! *
Init Chip....Complete! *

Zigbee protocol stack *
Reset NIB *
Reset PIB *
initialize neighbor table *
I am end device*
network discovery *
network discovery ok! *
network count=1 *
nwk descriptor _pan id=2222
device setting...*
Set Device OK *
network joining...*
short address= 0011
Associate status: SUCCESS*

insert node to neighbor table ok!
Network-joining ok!
timer create ok!!*
Timer start ok *
please input your test command: *_
```

Fig. B-6 Status of router is shown in hyper-terminal

B.3.2. Frames Measurement

The device first broadcasts a beacon request command to existing routers in POS, and get a beacon frame from the router with PAN ID is equal to 0x2222. We can know the device is router according to Coord field of the beacon frame’s super-frame specification is equal to 0. Since Assoc field of the beacon frame’s super-frame specification is equal to 1, the router allows other devices to join it. The device joins the router by using the NLME-JOIN-Request primitive, which issues the MLME-ASSOCIATE-REQUEST primitive that sends association request command to the router, and then sends physiological data to the router, the router sends the data to PAN Coordinator. Measurement of frames is shown in Fig. B-7 and Fig. B-8.



Time (us)	Length	Frame control field					Sequence number	Dest. PAN	Dest. Address	Beacon request	LQI	FCS
		Type	Sec	Pnd	Ack req	Intra PAN						
+3646956 =7807725	10	CMD	0	0	0	0	2	0xFFFF	0xFFFF		152	OK

Time (us)	Length	Frame control field					Sequence number	Source PAN	Source Address	Superframe specification						GTS fields		LQI	FCS
		Type	Sec	Pnd	Ack req	Intra PAN				BO	SO	F.CAP	BLE	Coord	Assoc	Len	Permit		
+2044 =7809769	13	BCN	0	0	0	0	67	0x2222	0x0000	15	15	15	0	1	1	0	0	172	OK

Time (us)	Length	Frame control field					Sequence number	Source PAN	Source Address	Superframe specification						GTS fields		LQI	FCS
		Type	Sec	Pnd	Ack req	Intra PAN				BO	SO	F.CAP	BLE	Coord	Assoc	Len	Permit		
+1427 =7811196	13	BCN	0	0	0	0	3	0x2222	0x0001	00	00	15	0	0	1	0	0	184	OK

Time (us)	Length	Frame control field					Sequence number	Source PAN	Source Address	Superframe specification						GTS fields		LQI	FCS
		Type	Sec	Pnd	Ack req	Intra PAN				BO	SO	F.CAP	BLE	Coord	Assoc	Len	Permit		
+2271341 =10082537	13	BCN	0	0	0	0	68	0x2222	0x0000	15	15	15	0	1	1	0	0	172	OK

Time (us)	Length	Frame control field					Sequence number	Dest. PAN	Dest. Address	Source PAN	Source Address	Association request				
		Type	Sec	Pnd	Ack req	Intra PAN						Alt.coord	FFD	Power	Idle RX	Sec A
+2850916 =15762431	23	CMD	0	0	1	0	16	0x2222	0x0001	0xFFFF	0x0807060504030201	0	0	0	0	0

Time (us)	Length	Frame control field					Sequence number	LQI	FCS
		Type	Sec	Pnd	Ack req	Intra PAN			
+1117 =15763548	5	ACK	0	0	0	0	16	180	OK

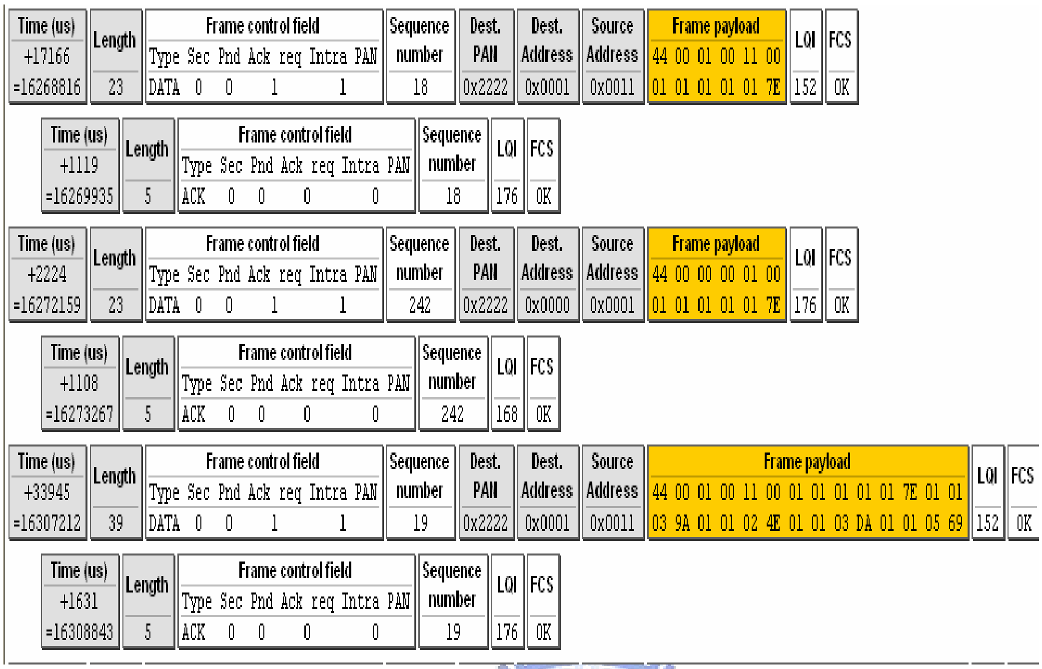
Time (us)	Length	Frame control field					Sequence number	Dest. PAN	Dest. Address	Source PAN	Source Address	Data request	LQI	FCS
		Type	Sec	Pnd	Ack req	Intra PAN								
+482903 =16246451	20	CMD	0	0	1	0	17	0x2222	0x0001	0x2222	0x0807060504030201		152	OK

Time (us)	Length	Frame control field					Sequence number	LQI	FCS
		Type	Sec	Pnd	Ack req	Intra PAN			
+1023 =16247474	5	ACK	0	1	0	0	17	180	OK

Time (us)	Length	Frame control field					Sequence number	Dest. PAN	Dest. Address	Source PAN	Source Address	Association response	
		Type	Sec	Pnd	Ack req	Intra PAN						Short addr	Assoc. stat
+2864 =16250338	29	CMD	0	0	1	0	241	0x2222	0x0807060504030201	0x2222	0xAABBCCDDEEFFABCE	0x0011	Successful

Time (us)	Length	Frame control field					Sequence number	LQI	FCS
		Type	Sec	Pnd	Ack req	Intra PAN			
+1312 =16251650	5	ACK	0	0	0	0	241	152	OK

Fig. B-7 Frames measurement for device using Chipcon sniffer



Time (us)	Length	Frame control field					Sequence number	Dest. PAN	Dest. Address	Source Address	Frame payload	LQI	FCS
+17166		Type	Sec	Pnd	Ack req	Intra PAN					44 00 01 00 11 00		
=16268816	23	DATA	0	0	1	1	18	0x2222	0x0001	0x0011	01 01 01 01 01 7E	152	OK

Time (us)	Length	Frame control field					Sequence number	LQI	FCS
+1119		Type	Sec	Pnd	Ack req	Intra PAN			
=16269935	5	ACK	0	0	0	0	18	176	OK

Time (us)	Length	Frame control field					Sequence number	Dest. PAN	Dest. Address	Source Address	Frame payload	LQI	FCS
+2224		Type	Sec	Pnd	Ack req	Intra PAN					44 00 00 00 01 00		
=16272159	23	DATA	0	0	1	1	242	0x2222	0x0000	0x0001	01 01 01 01 01 7E	176	OK

Time (us)	Length	Frame control field					Sequence number	LQI	FCS
+1108		Type	Sec	Pnd	Ack req	Intra PAN			
=16273267	5	ACK	0	0	0	0	242	168	OK

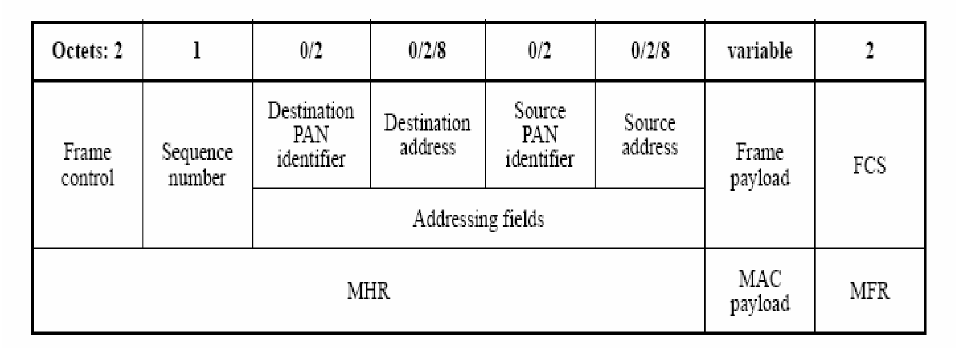
Time (us)	Length	Frame control field					Sequence number	Dest. PAN	Dest. Address	Source Address	Frame payload	LQI	FCS
+33945		Type	Sec	Pnd	Ack req	Intra PAN					44 00 01 00 11 00 01 01 01 01 01 7E 01 01		
=16307212	39	DATA	0	0	1	1	19	0x2222	0x0001	0x0011	03 9A 01 01 02 4E 01 01 03 DA 01 01 05 69	152	OK

Time (us)	Length	Frame control field					Sequence number	LQI	FCS
+1631		Type	Sec	Pnd	Ack req	Intra PAN			
=16308843	5	ACK	0	0	0	0	19	176	OK

Fig. B-8 Frames measurement for device using Chipcon sniffer

Appendix C.

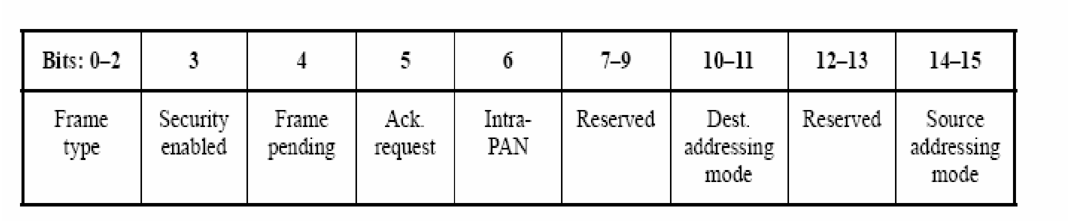
C.1. General MAC Frame Format



Octets: 2	1	0/2	0/2/8	0/2	0/2/8	variable	2
Frame control	Sequence number	Destination PAN identifier	Destination address	Source PAN identifier	Source address	Frame payload	FCS
		Addressing fields					
MHR						MAC payload	MFR

Fig. C-1 general MAC frame format diagram

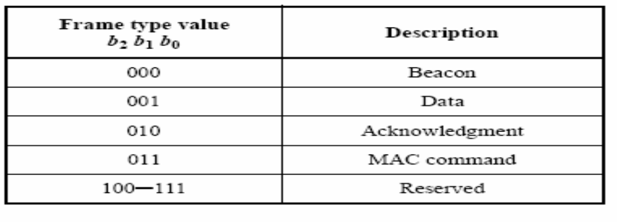
C.2. Frame Control Field Format



Bits: 0–2	3	4	5	6	7–9	10–11	12–13	14–15
Frame type	Security enabled	Frame pending	Ack. request	Intra-PAN	Reserved	Dest. addressing mode	Reserved	Source addressing mode

Fig. C-2 frame control field format diagram

C.3. Frame Type Field Format



Frame type value $b_2\ b_1\ b_0$	Description
000	Beacon
001	Data
010	Acknowledgment
011	MAC command
100—111	Reserved

Fig. C-3 frame type field format diagram

References

[1] ZigBee Alliance, ZigBee Specification, Available: http://www.zigbee.org.

[2] K. Hung and Y.-T. Zhang, “Usage of Bluetooth in wireless sensor for Tele-healthcare,” *Proc. of the Second Joint EMBS/BEMS Conf. Houston,* pp. 1881–1882, 2002.

[3] T.-H. Tan, C.-H. Pai and K.-L. Yeh, “Design and implementation of home health care system based on web service, Bluetooth, and GSM short message service,” *J. of National Taipei Univ. of Tech.,* pp. 15–25, 2004.

[4] A.Tura, M. Badanai, D. Longo and L. Quareni, “A medical wearable device with wireless Bluetooth-based data transmission,” *Measurement Sci. Rev.*, vol. 3, sec. 2, pp. 1–4, 2003.

[5] J.-C. Chien and C.-C. Tai, “The implementation of Bluetooth-based wireless phonocardio-diagnosis system,” *Proc. IEEE Inter. Conf. on Networking, Sensing and Control,* pp. 170–171, 2004.

[6] M. S. Enrique and A. M. Miguel “AC-coupled front-end for bio-potential measurements,” *IEEE Trans. Biomed. Eng.*, vol. 50, no. 3, pp. 391–395, 2003.

[7] B. B. Winter, J. G. Webster, “Driven-right-leg circuit design,” *IEEE Trans. Biomed. Eng.*, vol. 30, no. 1, pp. 62-66, 1983.

[8] I. Korhonen, J. Parkka and M. V. Gils, “Health monitoring in the home of the future,” *IEEE Eng. Med. Biol. Mag.*, pp. 66–73, 2003.

[9] K.-I. Hwang, J. In, N. Park and D.-S Eom, “A design and implementation of wireless sensor gateway for efficient querying and managing world wide web,” *IEEE Trans. Consum. Electron.,* vol. 49, no. 4, pp. 1090–1097, 2003.

[10] B. M. Thraning, “The Impact of ZigBee in A Biomedical Environment,” *Agder University College Grimstad*, pp. 6–17, 2005.

[11] J. G. Webster, Editor, “Medical Instrumentation: Application and Design,” third edition, *John Wiley & Sons, Inc*, pp. 143, pp. 235–237, 1998.

[12] U. Tietze and C. Schenk, “Electronic Circuits: Design and Applications,” *Springer-Verlag*, pp.

385–387, 1990.

[13] R. J Simon, T. Davis, J. Eaton and R. M. Goertz, “Windows 95 Multimedia and ODBC API Bible”, *Waite Group Press*, pp.1–273, 1996.

[14] N. Patwari, A. O. Hero, M. Perkins, M. Perkins, N. S. Correal and R. J. O’Dea,” Relative location estimation in wireless sensor networks,” *IEEE Trans. Signal Process*., vol. 51, no. 8, pp. 2137–2418, 2003.

[15] UBEC Inc., “UZ2400 Data Sheet”, *UBEC Inc*., pp. 9, 2005.

[16] Lucent Inc., “W7020 Data Sheet”, *Lucent Inc*, 2001.