Simulation of radio networks

Yu, Liyi

SIMULATION OF RADIO NETWORKS

Liyi Yu

Centre for Communication systems
School of Engineering
University of Durham

A THESIS UNDERTAKEN IN ENGINEERING SCHOOL OF THE UNIVERSITY OF DURHAM IS SUBMITTED FOR THE DEGREE OF MASTER OF SCIENCE (MSc).

JANUARY 2005

Declaration

I hereby declare that this thesis is a record of work undertaken by myself and supervised by Jim Swift, that has not been the subject of any previous application for a degree, and that all sources of information have been duly acknowledged.

Abstract

Radio communication such as licensed mobile phone systems are widely used, and have been over the past five years. This thesis focuses on simulation of radio networks. The aim is to develop a simulation and modelling tool that will allow a wide range of radio systems to be studied at system level, including where the radio element is connected to and interacting with point to point networks. The aim of the work is that the simulation should be as close as possible to real systems. Chapter 1 first introduces the applications, advantages and disadvantages of radio, following a discussion of the differences between radio networks and point to point networks. Chapter 2 briefly introduces the OSI model and the modified model used in this thesis. Radio transceiver chips play the role of the physical layer of radio devices and chapter 3 introduces three popular radio chips. Chapter 4 shows the necessity to simulate radio networks, and introduces C++ as a language for implementing simulators. The design of the simulation tool is also outlined. A local sensor system is simulated in chapter 5. Chapter 6 simulates a WLAN (Wireless Local Area Network) that adopts 802.11b standard. Chapter 7 gives the conclusion and future work in this area.

Acknowledgements

The author would like to acknowledge the invaluable help of his supervisor, Jim Swift for his teaching and suggestions during the work on this project. Thanks must also be expressed to the persons who help the author a lot: Peter Baxendale who taught network communication; David Clark who is always patient to answer the author's questions; Monish Chaudhary who always discusses the network communication with the author; Yuwen Wu who encourages the author in study and life. Finally, thanks must go to his parents. Without their support, the author can not get the opportunity to study abroad and finish the work.

Contents

Chapter 1 Introduction

The uses of communications have increased very rapidly over the past ten years. China's Ministry of Information Industry (MII) claimed that the number of mobile phone users was more than 310 million in China by the end of the July of 2004. Nokia, the biggest mobile phone company said that the number of worldwide mobile phone users would reach 2 billion by 2007. According to the usage and population statistics provided by Internet World Stats [1], the number of internet users was more than 812 million at the end of September 2004. Many millions of messages are exchanged every day in the world.

The main usages of communication can be divided into two categories: domestic and commercial applications.

1.1 Domestic communication usages

1.1.1 Domestic Applications

Domestic usage mainly means private communication. The communication may be between two people whose distance may be thousands of kilometres or few hundred meters. The communication could also be between one person and one machine such as online shopping, online music, online movie or reading news from some web sites. The communication can be made by mobile phone, fixed phone, e-mail or some other software such as MSN (Microsoft MSN message) for chatting or IE (Internet Explorer) for browsing websites.

1.1.2 Drivers for Domestic Usage

a) Low cost of electronics: The costs of mass produced electronics are low. Hence private uses of the internet are also economic, for example the costs of e-mails are usually cheaper than the costs of letters.

b) High labour cost: An electronic security system for a house can work 24 hours per day without salary after the initial installation cost, but a watchman would be paid for per hour and may get double pay for holidays.

c) Electronic manufacture development: The development may increase the quality of products and the efficiency of producing products that means factories may need fewer workers to produce more products in the same time unit than before. Due to this, the cost of the mass produced products becomes lower. In 1998 the main domestic desktop PC ran at a clock speed of 586MHz or less, but six years later the speed of the latest Intel CPU achieves 3.4GHz and the price of the desktop is lower than before.

d) Electronic communication makes people's life more convenient and easier. No matter where the person is, the e-mail sent by him/her may achieve the destination in few seconds. Usually with a letter, more than one day will be spent in delivering from one city to another city if both are in the same country and seven days from one country to another country. People holding a mobile phone can contact their friends or family members at anytime and anywhere within the network coverage.

1.2 Commercial communication usages

1.2.1 Commercial Applications

Modern communication is widely used in commercial applications as well, for example supermarkets adopt wireless or wired bar-code recording systems. When the goods are checked out, systems will make a record and send related information to a central computer in the store which can make an order to distribution centres for restocking goods. Goods will be delivered to the markets overnight. These automatic information exchanges improve the efficiency. Many factories adopt the wired or wireless building monitor systems. A Central computer will keep the information for the doors, windows or temperature of rooms by getting reports from related monitors. Online video meetings save money for international companies. Managers can have meetings with others in different countries in their offices rather than go to the same city by train or air. Satellites provide wireless communication services such as satellite TV and satellite phone by exchanging data with devices on the ground.

1.2.2 Drivers for Commercial Applications

The key factors to Commercial companies are money and efficiency.

a) Money: Cost of electronic equipment is often lower than the cost of labour. It is the same driver as domestic applications such as building monitors and online video meetings. People inside a forest or on the mountain may get mobile phone signals from satellites, while it is not viable for any company to build base stations in these areas of low population density.

b) Efficiency: For many calculations, information storage and memory capacity of a computer is better than a human being's. Supermarket staffs would have to spend more time in reading bar-code and calculating the price without computer, but bar-code readers will read the code and show the price on the screen in few seconds.

Besides the two factors mentioned above, the limit of the human being is another reason. Electronic devices can work under the conditions which exceed the limit of human being, such as satellites or some detection system working under great water pressure or high temperature.

1.3 Communication Medium

All communication data in the form of electromagnetic waves is transported by physical medium which may be wired such as optical fibre, twisted pair, and coaxial cable or wireless such as space. Different medium support different data rates and have different costs for installation. No matter which medium is used, the signal strength will decrease with distance, so signal repeaters are used in long distance transmission. The distance between each repeater depends on the type of medium.

1.3.1 Wired Medium

a) Optical fibre: The basic components for an optical transmission system are a light source, optical fibre and detector. In the transmitter electronic signals are converted to light signals. For instance 1 bit is indicated by a pulse of light and zero for no pulse of light. Light signals are passed by the optical fibre from transmitter to receiver. The receiver detector receives the light signals and the light signals are converted to be electronic signals.

Optical fibres are made of glasses. Light signals follow the optical principle in transmission in optical fibre shown in figure 1.1.

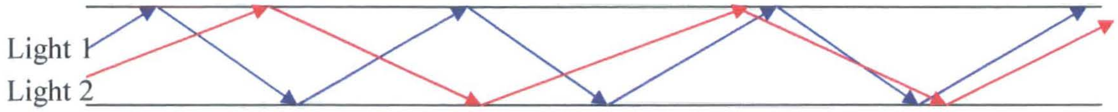


Figure 1.1 Light Transmission

Different light rays with different angles of incidence that are bigger than the critical angles are in total reflected at different angles of reflection. Each ray is called a different mode. The optical fibres with the property shown in figure 1.1 are named multimode fibres [2].

If the diameter of optical fibre is reduced so small that the light signals are propagated in a straight line, this optical fibre is called single mode fibre. Single mode fibres are widely used in longer distance transmission.

The maximum data rate optical fibre supports can be few tens of Gbps. The repeaters are used in every tens of kilometres.

b) Coaxial Cable: For an electronic communication system based on coaxial cable, it does not need signal conversion between light and electrical used in optical systems. The signals passed in coaxial cables are electrical. A coaxial cable consists of four parts. The inner part is a copper wire. The second part surrounding the inner part is a layer of insulation. The insulation is surrounded by a grounded shield of braided wire. The outer part is the plastic covering. Figure 1.2 shows an example of a coaxial cable.

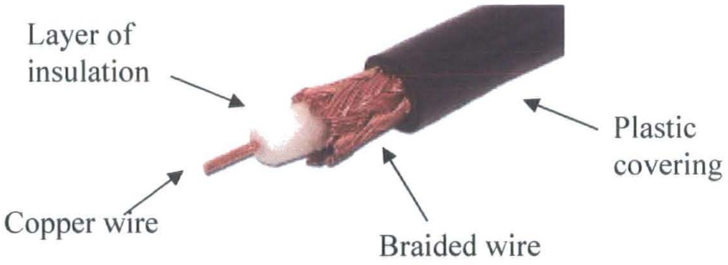


Figure 1.2Coaxial Cable

Coaxial cable is widely used in TV system. The maximum data rates supported by coaxial cable is a few hundreds Mbps. The repeater is needed for every few kilometres.

c) Twisted Pair: The signals passed in twisted pairs are electronics too. Twisted pair is made of two insulated cooper wires which are twisted around each other in a helical form in a single cable. The purpose of twisting the wires is to reduce the wires crosstalk. The greater the number of twist, the more crosstalk is reduced.

Twisted pairs can be divided into two types: shielded twisted pair (STP) and unshielded twisted pair (UTP).

STP giving the better protection from interference is widely used in token ring networks. UTP is commonly used for telephone line and computer networking. There are various types of UTP. The twists per centimetre are one part of the specification of the type. Category 3 and category 5 are two popular types of twisted pair. Category 5 with more twists is more suitable for long distances transmission.

The maximum data rate twists pair supports can be 100 Mbps for Ethernet. The repeaters are used in every few kilometres.

1.3.2 Wireless Medium

Wireless medium means space. Information in the form of electromagnetic waves spreads from one place to another place by space. WordNet dictionary gives the definition of electromagnetic waves: radiation consisting of waves of energy associated with electric and magnetic fields resulting from the acceleration of an electric charge.

At the transmitter station an information carrying signal (base band signal) modulates one parameter of the carrier signal before transmission:

Electromagnetic wave can be described in follow equation:

$$\alpha \sin(\omega * t + \theta); \qquad (1)$$

Here α is the amplitude of the carrier signal, ω ($2*\pi*$f) is the angular frequency and θ refers the phase of the carrier signal. AM which modulates α and FM modulating ω are two examples of modulation methods that are commonly used in radio communication. The carrier frequency is higher than the base band frequency. It is chosen to give desired transmission property. The reasons (a, b) for using carrier frequency modulation are following:

a) Antenna can efficiently radiate energy of the electromagnetic wave whose wavelength is related to the antenna physical dimensions. (More details about the relation between the efficiency of radiation energy and antenna size can be found in [3].) If the transmitter transmits the base band signals directly assuming the base band frequency is high enough, it has to install a long antenna because of the low frequency of base band signal. It is not easy in practice in real system, so higher carrier frequencies are used for transmission.

b) According to the frequency used, the antenna and related radio circuit of the receiver select the wanted signal from various signals operating on different frequencies. Then transmitter amplifies the output power of electromagnetic wave to the desired power level for transmission.

The related receivers in the signal coverage will pick up the electromagnetic waves, demodulate them and convert them to 1 bits or 0 bits based on some code method if they are digital signals. In a vacuum devices operating on different frequencies propagate their carrier at the same speed which is about $3.0*10^{8}$ m/s.

Depending on frequency range, the electromagnetic spectrum could be divided into seven categories of radio, microwave, infrared, visible light, Ultraviolet light, X-rays, and gamma rays. For instance the radio waves which extend from 30 KHz (wavelength 10km) to 300GHz (0.1m). The first three categories are used for transmitting information.

Within countries laws, frequencies can be divided into two categories: licence frequency bands that people should pay for using and unlicence frequency bands that are free for using. Unlicence frequency bands include ISM (Industrial, Scientific, Medical) bands which may be different from country to country. For instance in United States the frequency bands (902 MHz-928MHz, 2.4GHz-2.4835GHz and 5.735GHz-5.860GHz) are free to be used by devices with power under 1 watt.

One example of licence free standards is blue tooth. The standard is designed at the personal entertainments area, for example linking the personal radio to the headset without wires. 802.11b is another standard. Computers using 802.11b can access the near wireless networks.

Wireless signals strength decrease very rapidly with distance. Equation 2 [4] shows the fundamental relation between received power Pr and distance d in free space propagation.

$$Pr = Pt*Gt*Gr*\lambda*\lambda\ /\ (16*d*d*\pi*\pi) \qquad (2)$$

Here Pr and Pt are the received power and transmitted power; Gt and Gr are the transmitter and receiver antenna gains; d is the distance between the transmitter and receiver; Greek letter λ is called wavelength that is defined to be the distance between two consecutive wave crests or wave troughs. The fundamental relation between f (frequency), λ and c (speed of light in vacuum) is:

$$c = f*\lambda \qquad (3)\ [4]$$

If the parameters except d are constant, received power is decreased by distance squared. If the parameters except λ are constant, then the higher the frequency is, the lower the received power is.

1.3.3 Advantages and Disadvantages of Wireless Medium

1) Compared with wired transmission medium such as twisted pair, a wireless medium do not need cabling for penetrating buildings. On the other hand fixed devices limit the service coverage. For instance a fixed phone can only be used in a part of the room it is installed in.

2) The cost of unlicensed frequency is free to use, but the cost of wired medium such as optical fibre can not be ignored.

Every thing has two sides. The characteristics of the electromagnetic waves also bring a lot of problems. For an obstruction such as a building between the transmitter and receiver, the signal can arrive at the receiver via different paths of different lengths. When the signals are summed at the receiver, the phase differences due to the different path lengths can reduce or increase the signal strength. Adjacent channels may also interfere with each other. In digital systems these problems can lead to large bit error rates.

However, more and more communication applications are adopting wireless medium such as cordless phone, mobile phone, and radio (wireless) computer networks.

Radio networks are used in an ever increasing range of applications, playing an increasing role in people lives. Market research analysts are predicting that wireless network technologies will eventually become more widespread than the various wired solutions. A large technology consulting company Strategy Analytics has forecasted that 19 percent of the households in the U.S. and 15 percent of European households are expected to have wireless home networks by the year 2005 [5].

Radio networks usually range from licensed mobile phone systems, where handsets communicate over several kilometres to their base stations, to industrial or domestic unlicensed systems where each unit has a radio range of up to 200m. Radio networks are different from wire networks. For instance, they propagate information by electro-magnetic waves whose properties may cause near-far effect, shadow fading. Wire networks are fixed but radio systems have mobility.

1.4 Radio communication networks

1.4.1 Key factors for radio communication

The key factors of radio networks are frequency and signal strength.

a) Frequency:

Interference at the same frequency: when difference devices use the same frequency at the same time, they may interfere with each other if in range. The receiver may miss the signal or receive corrupt messages. For wireless systems, when one device is communicating with another device, other in range devices should not use the channel (frequency band) during the communication time.

b) Signal strength:

Adjacent channel interference: A wireless system can be divided into several groups operating on different frequencies. Though devices in different groups use different frequencies at the same time, an interference called adjacent channel interference may be caused. Near-far effect is one reason for causing adjacent channel interference. The strength of signal decreases with the distance increasing between transmitters and receivers. If one transmitter's power is too strong, it may cause significant interference to the receiver that is far away from its target transmitter. Details and example are shown in chapter 5.3.

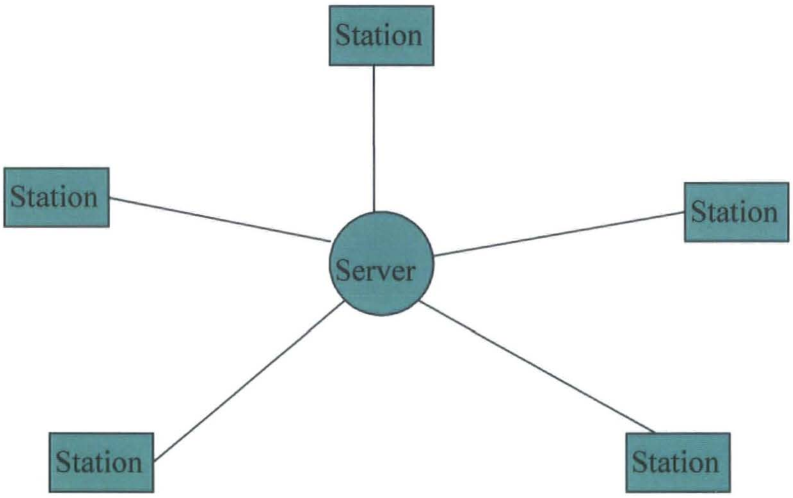
Multipath fading: Due to obstructions, the received signal can arrive at the receiver via different path with different phases which cause the fluctuation of the signal amplitude. The signal strength may be decreased because of the different phases. It could be divided into two categories of selected frequency fading and selected time fading.

1.4.2 Differences between radio networks and point to point networks

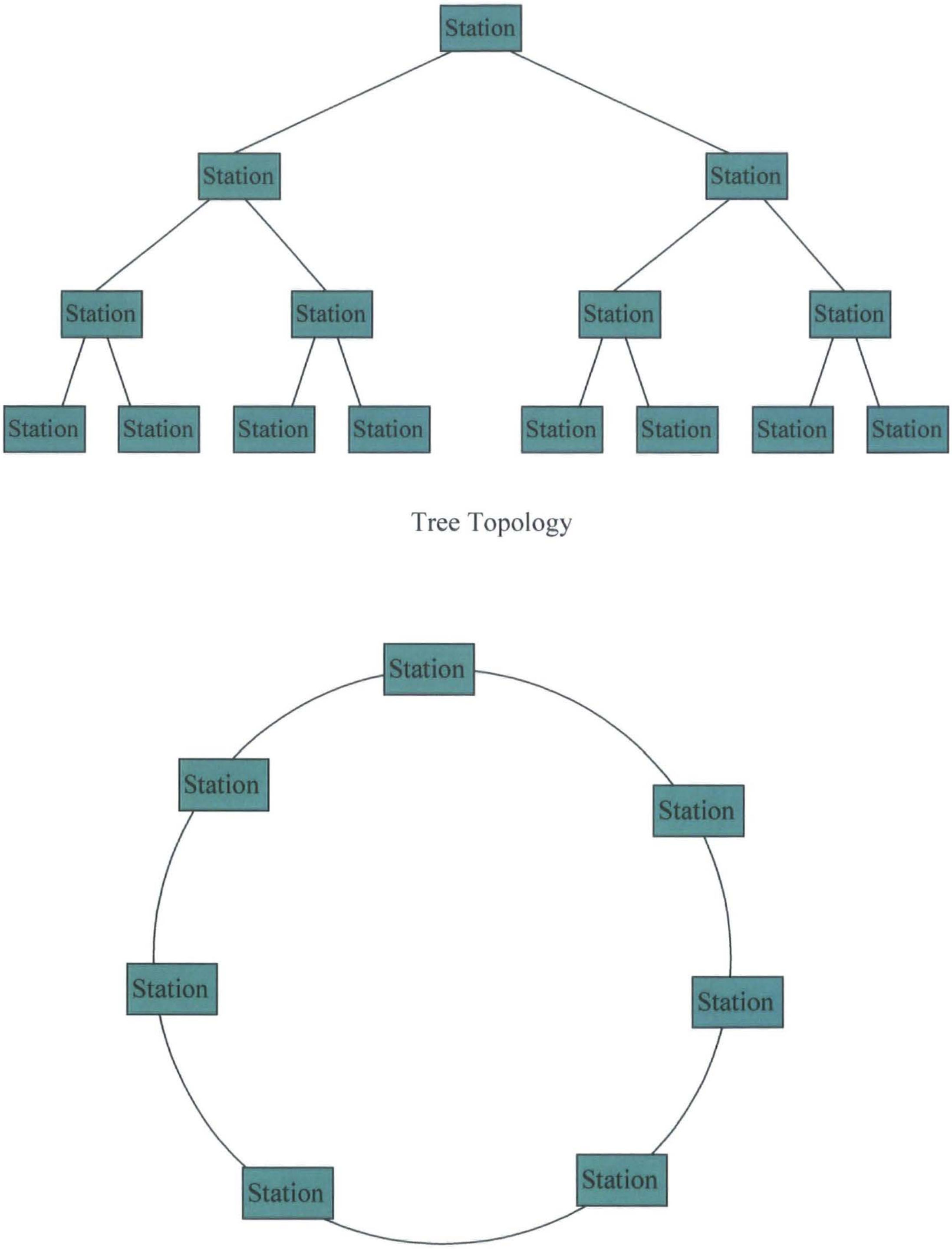
1.4.2.1 Point-to-point networks

Point-to-point networks define that only two stations can be connected over the link. The fixed phone line is one of the application examples. A phone accesses the phone line by dial-up. Once the phone connects to the target phone, the circuit only connects to those two phones assuming a circuit switched network. The circuit is occupied by the two phones until the communication is over.

The topologies of the point-to-point networks can be star topology, tree topology, and ring topology in figure 1.4 networks topologies.



Star Topology



Tree Topology

Ring Topology

Figure 1.4 Networks topologies

Mesh topology means it mixes more than one topology in its topology such as mixing star topology and tree topology. Different topology may bring different advantages and disadvantages. These details can be found in [6] [7].

1.4.2.2 Radio networks (Broadcast networks)

Radio networks are one kind of broadcast networks. Broadcast networks defines that one link can connect more than two exchangers and information sent by one of the exchanges may be received by other exchangers connecting to the link. For radio networks, the link means the frequency the devices operating on.

1.4.2.3 Differences in coupling options

a) For point-to-point networks only device connecting to other end of link can get information, but for radio networks, radio devices in the coverage operating on the same frequency will pick up information sent by other radio devices. Radio devices should only deal with the information intended for them.

b) Wired networks are not necessarily point-to-point networks. Bus topologies as shown in figure 1.5 are broadcast networks.

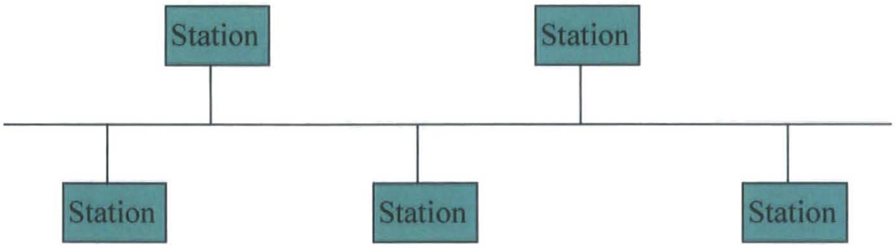


Figure 1.5 Bus Topology

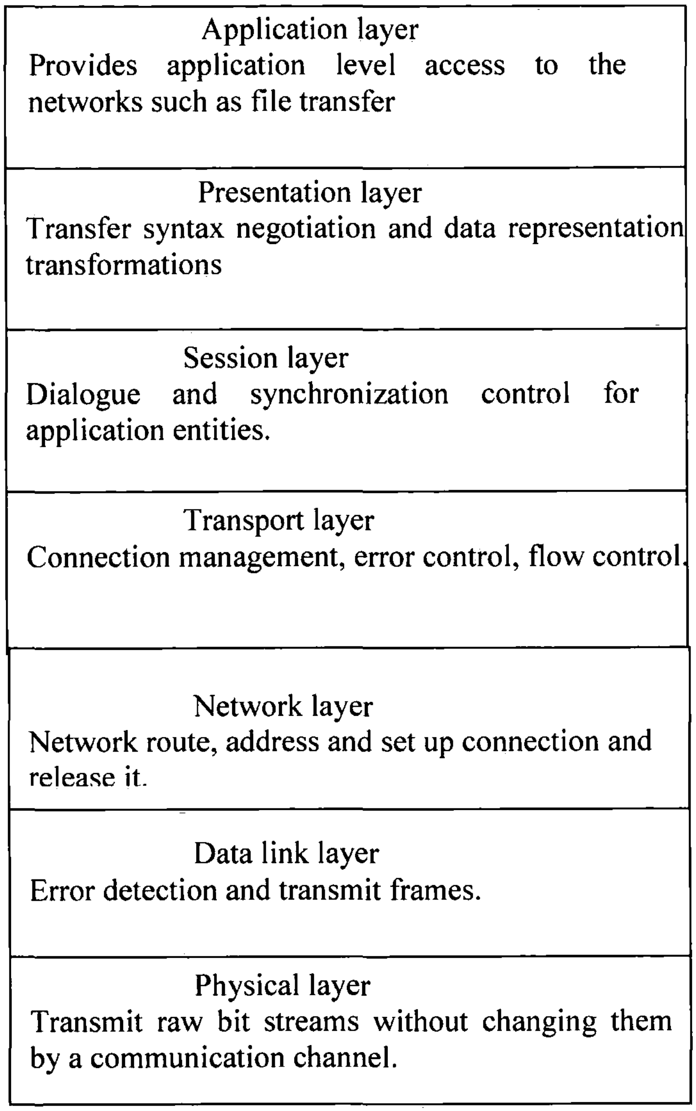
Wireless networks also may not be broadcast networks such as microwave system. Microwave spreads very directionally. Due to this property, it can be considered as a point-to-point network.

Radio networks may connect to wired networks which may be point-to-point networks, but this thesis only considers the radio part and the wired part is assumed to be ideal. It means there are no corrupted or lost messages in wired networks. Chapter 6 WLAN is based on this assumption.

Chapter 2 OSI model

It is difficult for networks using different specifications to communicate with each other and it is easy to make the networks be incompatible because of using different hardware and software implementation. Due to this, the OSI (Open System Interconnection) reference model was promulgated as an international standard for exchanging data between different systems by International Organization for Standardization (ISO).

The network functions for dealing with data are divided into seven layers by the OSI model. They are Physical layer, Data link layer, Network layer, Transport layer, Session layer, Presentation layer and Application layer. The OSI model also specifies the particular functionality for each layer to communicate with the same layers software or hardware on other network devices and the information passed between each layer in one device. Figure 2.1 shows the seven layers and related functions.



Application layer
Provides application level access to the networks such as file transfer

Presentation layer
Transfer syntax negotiation and data representation transformations

Session layer
Dialogue and synchronization control for application entities.

Transport layer
Connection management, error control, flow control.

Network layer
Network route, address and set up connection and release it.

Data link layer
Error detection and transmit frames.

Physical layer
Transmit raw bit streams without changing them by a communication channel.

Figure 2.1 OSI model

2.1 The physical layer

2.1.1 Introduction

It is the lowest layer of the OSI Model. Its purpose for transmission is to transmit the raw bit streams generated by the higher layers, and for reception, to receive the bits, possibly corrupted, and pass them to the higher layers.

It is the physical layer that controls how the information is represented for transmission. This could be as a voltage level, where 0 is represented as below V1 volts and 1 as above V2 volts. For example, in radio systems where the data stream modulates a carrier, frequency, phase or amplitude can be modulated. For frequency modulation, one frequency could represent 0, and another frequency could represent 1. More complex schemes could use four signal states, such as phase, thereby transmitting 2 bits at once.

With respect to radio devices, great care has to be taken over the use of the radio spectrum and power, because it is common to all radio devices. Ideal radio devices should use the smallest possible region of the spectrum, and cause minimum interference to adjacent frequencies.

2.1.2 The Maximum data rate of a channel

In 1948, Shannon [8] calculated the maximum data rate (C) of a limited bandwidth (W) noise channel.

$$C=W*\log_2(1+S/N) \quad \text{bits/second} \qquad (1)$$

S/N is the signal-to-noise ratio. Here S is the signal power and N is the noise power.

2.1.3 Transmission medium

This thesis focuses on wireless simulation, so the transmission medium that will be discussed is electromagnetic waves.

As it is mentioned in chapter 1.3.2, the electromagnetic spectrum could be divided into seven categories and the firs three categories that are radio, microwave, infrared are used for transmitting information.

In a vacuum devices operating on different frequencies propagate their carrier at the same speed which is about $3.0*10^8$ m/s.

In free space [4] the radio signal power calculation built in simulation tool is as following:

$$Pr = Pt*Gt*Gr*\lambda*\lambda \,/\, (16*d*d*\pi*\pi) \qquad (2)$$

The meaning of each parameter such as Pr in the above equation has been introduced in chapter 1.3.2.

2.1.4 Transmission method

A signal is normally modulated by a carrier signal with higher frequency before transmission. The modulation method is divided into two methods which are analogue modulation and digital modulation.

Analogue modulation uses the analogue signal to directly vary a parameter of the carrier signal. There are three formats: amplitude modulation (AM), frequency modulation (FM) and phase modulation (PM).

Digital modulation uses the digital signal to vary the carrier signal. For instance at the transmitter, the digital signal 1 and 0 can be represented by two different frequencies f1 and f2 for transmission. That is the basic concept of FSK modulation. Typically the digital modulations are FSK (Frequency shift keying), PSK (Phase shift keying), GMSK (Gaussian Minimum shift keying) and QPSK (Quadrature PSK). More details about AM, FM, FSK, PSK, GMSK and QPSK are in [9].

2.2 The data link layer

2.2.1 Introduction

When the source transmits, the data link layer creates the boundary of the data frame passed by the higher layer and adds some necessary information into it such as control information that can be medium address information, error control, etc. The source then waits to receive reply sent by the destination. For reception, this layer detects the boundary and errors of the bit streams from the physical layer, and generates the ACK (acknowledgement) if necessary.

The data frame which includes data and control information can range from a few hundred to a few thousand bytes. This layer attempts to detect the errors by some detection method such as CRC (Cyclic Redundancy Check) [10]. During transmitting data frames, if the receiver detects some errors in the data frame, it will drop it. If the timeout set in transmitter expires before an ACK is received, the transmitter will retransmit that message. In some systems, retransmission may be done in higher layer as well.

The services provided by the data link layer can be divided into three kinds: unacknowledged connectionless service used by the most LANs, acknowledged connectionless service that is useful for unreliable channels and acknowledged connection-oriented service. The difference between the second one and the last one is that frames should arrive in order in the last services.

According to the particular simulation system, the data link layer chooses one kind of the services for the higher layer. For example, in chapter 5 Local sensor system simulation the data linker layer of each device provides the acknowledged connectionless service for the higher layer.

2.2.2 The medium access sublayer

If the physical layer uses a shared medium, the data link layer may have a sublayer named MAC (Medium Access Control). The MAC layer controls the access to the available or shared medium to transmit the next frame to avoid contention.

For systems, there are two methods for accessing a channel. One method is called static channel assignment such as FDM (frequency division multiplexing) or TDM (time division multiplexing) [11]. In the case of TDM, 'slot' is more often used than the word 'channel'. In static channel assignment, users access the channel by using the assigned frequency in FDM or the allocated time slot in TDM, so there should be no interference between users.

The other method is dynamic assignment channel that is better than static in dealing with bursty traffic [12]. For dynamic channel assignment, if there is more than one transmitter to transmit at the same time, it may cause collision. For coping with this kind of problem, the MAC checks the availability of the channel by some protocol such as CSMA (Carrier Sense Multiple Access) introduced in chapter 6.

2.3 The network layer

The main task of this layer is to decide the route from the source to the destination.

Due to whether the service is reliable or not, this layer provides two classes of service to the transport layer. One is connectionless service in which the source and destination should do error control and flow control themselves. The other one is connection - oriented service in which the network layer builds a virtual circuit before sending packets and provides flow control. When a virtual circuit is set up, all packets will follow this route. But for connectionless service, packets may follow different routes to reach the destination device. The Internet has a connectionless network layer and the ATM networks have a connection-oriented network layer.

In chapter 6 WLAN simulation, because the wired line is assumed to be a part of Internet, the network layer provides network address for each device and connectionless service for transport layer.

2.4 The transport layer

The main functions of the transport layer are multiplexing, demultiplexing, flow control and to set up network connection.

The function of multiplexing is to get the data coming from different ports together with some identifier information to create segment and pass them to the network layer.

The function of delivering the different data to the related socket is called demultiplexing.

Port number used in UDP (User Datagram Protocol) introduced in chapter 6 is one example of identifier information. According to the port number, transport layer in the receiver passes the data to the particular process via the related port.

Flow control plays a very important role in the transport layer. The function is to keep a fast transmitter from overrunning a slow receiver.

Usually the transport layer sets up different network connections for different transport connections required by the session layer.

The difference between this layer and the lower layers is that the transport layer is an end to end layer. For the lower layers the protocols are between each device and may be processed by the routers. The protocols of the transport layer are between the source device and the destination device.

2.5 The session layer

The functions provided by the session layer are the following: connect, exchange data, release connection, and recover after error. The data exchanged in this layer is in dialogue units.

Connect: Source session layer establishes a logical communication path (session connection) with the destination session layer. The data is transmitted in dialogue units by this path.

Release connection: connection will be released after transmission.

Recover after error: Source session layer inserts synchronization points in a dialogue. If the errors happen, it will resume the dialogue from an agreed synchronization point.

2.6 The presentation layer

There is a number of common abstract data syntax to be used by the application entities. Different application entities may use different memory organization. For instance a value of type integer may be represented in an 8-bits, 16-bits or 32-bits form and different byte ordering depending on the system. For instance in a 32-bits system, the byte ordering for representation of 1032_{10} can be big endian or little endian [13] (figure 2.2):

Big endian	00000000_2	00000000_2	00000100_2	00001000_2
Address	00	01	02	03
Little endian	00000000_2	00000000_2	00000100_2	00001000_2
Address	03	02	01	00

Figure 2.2 Big Endian and Little Endian

It is necessary that the message exchanged have a common memory organization for both application entities.

The function of this layer is to select the appropriate data representation or do necessary data conversions for maintaining the compatibility between two application entities. It manages the abstract data types and allows higher-level information to be defined and exchanged. In some applications, this layer has another function which provides the security for the transmitting data. The source presentation layer may encrypt the data and then only the destination presentation layer knows the method to decrypt it.

2.7 The application layer

The main difference between the application layer and other layers in OSI model is that the application layer does not provide services to any other OSI layers. The application layer is an interface to the end users and provides services for users outside of OSI model that may be human or software to access the networks

The application layer defines the action taken on the transmission or reception of a message: how to send a request, how to respond to a request.

HTTP (Hypertext transfer protocol), FTP (File Transfer Program) and Electronic mail are three examples of applications of this layer.

2.8 TCP/IP model

Actually the OSI model that is a reference model is rarely used in real system. Another model name TCP/IP model is widely used in the world. The differences between these two models in architecture are shown in figure 2.3.

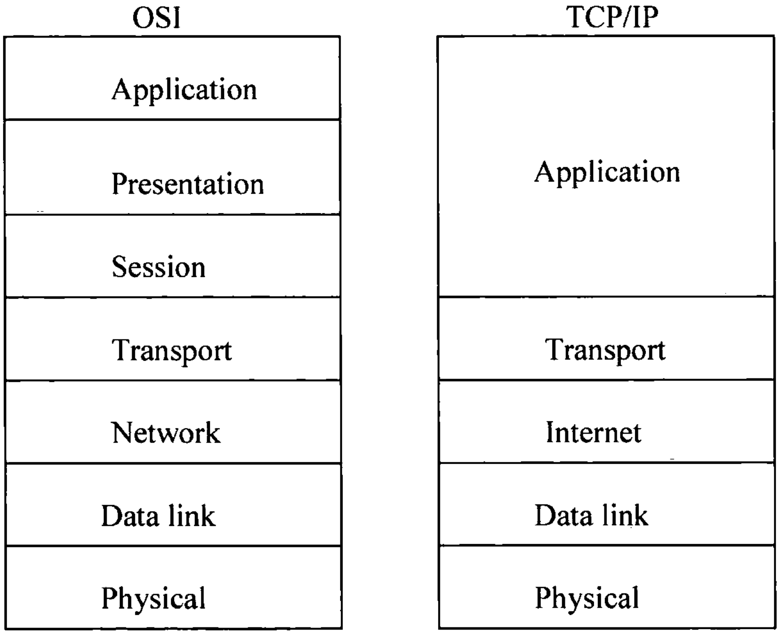


Figure 2.3 OSI and TCP/IP

Strictly speaking there are no data link layer and physical layer in TCP/IP model, because "TCP/IP model does not define the underlying network medium and physical connectivity; what is running below layer 3 is largely transparent [14]." More details about TCP/IP model can be found in [14] and [15]. Most standards such as 802.11b used in TCP/IP model still uses data link layer when they mention the layer below Internet layer.

For the work covered in this thesis, the network model used in this thesis is a five layers model in figure 2.4 that can be considered as a modified OSI model.

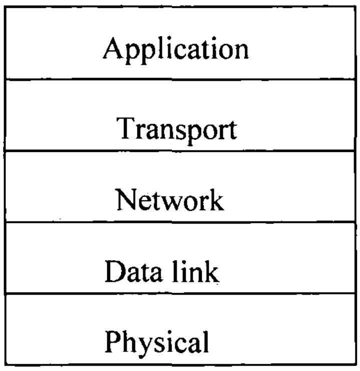


Figure 2.4 Five Layers Model

2.9 Simulation

Chapter 5 is simulating a simple system called local sensor system. The network structure used in this system is assumed to be simpler than the model shown in figure 2.4. It only contains physical layer, data link layer and application layer.

In chapter 6 the modified OSI model is simulated. 802.11b protocol is simulated in data link layer, UDP (User Datagram Protocol) is used in transport layer and IP (Internet Protocol) in network layer. The related details about the protocols used in simulation will be introduced in the chapter 5 and chapter 6.

Chapter 3 Radio chips

Radio transceiver chips play a key role in radio equipments. For radio systems, they act like "pipes" which pick up or transmit packets possibly without processing them. For reception, they receive the radio signal, and convert it into an appropriate electrical signal, which is a commonly digital signal today. They may or may not have any storage capacity within them. For transmission, they perform the reverse process. The design aim of these chips is to require the minimum of external components, such as filters and aerial compensation circuits.

3.1 BiM-418-40

BiM-418-40 [16] which is a low power UHF (UltraHigh Frequency) data chip not only can receive packets, but also transmit packets. This kind of chip is called transceiver.

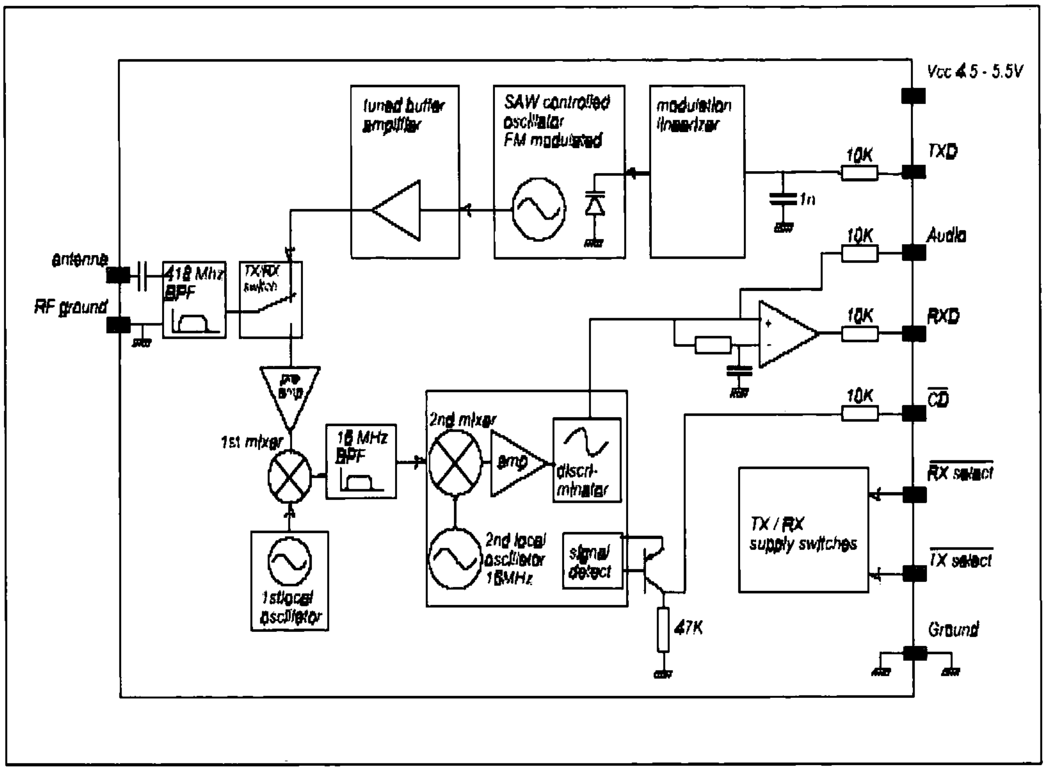


Figure 3.1 BiM-418-40 block diagram

Figure 3.1 shows the block diagram of BiM-418-40. More details can be found in [16]. The range of BiM-418-40 is about 30 meters indoor and 120 meters outdoor.

The advantages of the BiM-418-40 are data rates near 40kbit/s, and fast mode (TX/RX) changeover of less than 1 ms. Due to these factors, the manufacturer claims that a great number of radio applications adopt the BiM-418-40 and target applications are Medium speed computer networks, High integrity wireless Fire / Security alarms, Building environment control / monitoring, Vehicle alarm systems, Remote meter reading and Authorization / Access control.

The main difference with other low cost chips is that the BiM-418-40 is a single frequency chip which can only operate on unlicensed frequency (Licence Exempt) band 418MHz.

3.1.1 Data format

The maximum bit rate based on the three methods is 40kb/s. The three methods that will be discussed later are FEC coding, Byte coding and Bit coding.

For the BiM-418-40, the data format at transmission is Preamble, Control, Address, Data, CRC.

Preamble: For real systems, the first part of the packet received is the preamble which allows the receiver to lock to the incoming bit stream. The RX mode of the BiM-418-40 needs time to configure some functions such as setting the data slicer for stably receiving incoming data. The preamble time should be more than 3 ms.

Control: It may be bytes or a bit that indicates the beginning of the message. Usually it also contains the information for decoding such as byte count, repeater control.

Address: This information may be the 16/24 bit address of the receiver, transmitter or repeater.

Data: Usually the maximum of the user data is 256 bytes.

CRC: Decoder in the transceiver checks the correctness of the packet by this information. It may be 16/24 CRC or checksum of control-address-data fields.

Chips such as the BiM-418-40 are not suitable for some radio systems operating on multi bands such as frequency hopping spread spectrum systems (FHSS). Companies design and manufacture other kinds of radio frequency chips to satisfy the requirements. The AT86RF211 and the CC1020 created by two different companies are typical chips of this type. They are also low cost, low power consumption, high level of integration, high sensitivity.

3.2 AT86RF211

AT86RF211 [17] which is designed and manufactured by Atmel Corporation is also a radio transceiver chip. Figure 3.2 [17] is the block diagram of AT86RF211.

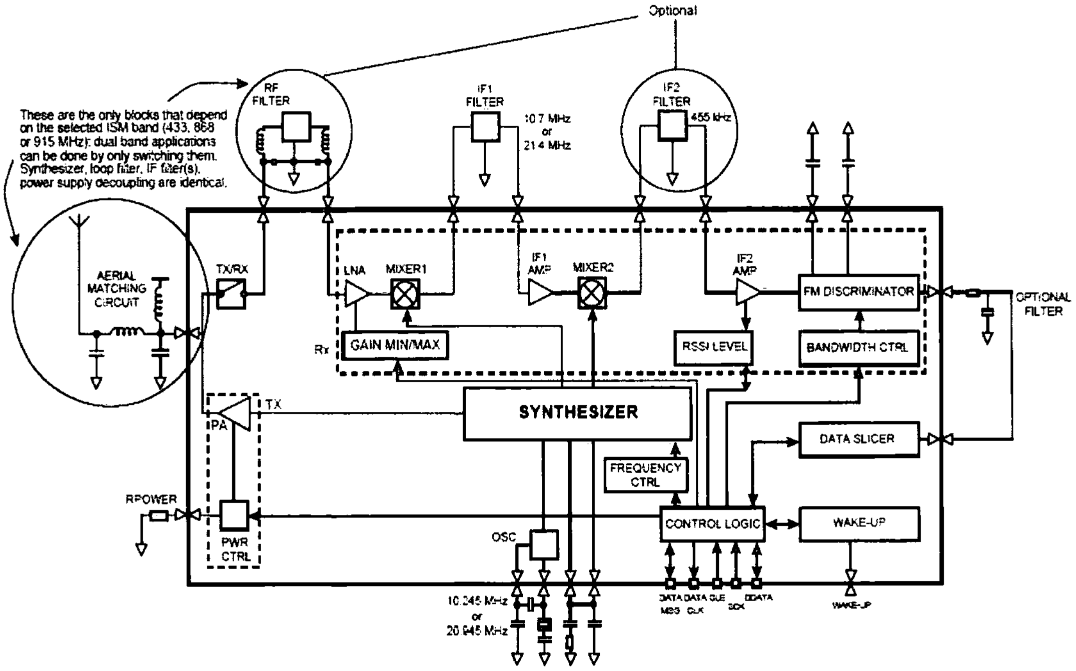


Figure 3.2

AT86RF211 operates from 400 MHz to 950MHz and can therefore use the licence-free ISM (Industrial-Scientific-Medical) bands in this range. Its coverage can be divided into short range, long range (hundreds of meter) and very long range (more than 1 km) compared with other low cost radio chips' coverage.

The manufacturer claims that the AT86RF211 is widely used in many radio systems such as alarms, radio modems and high –tech toys and also aims it for battery operated systems, as it can be powered with only 2.4V.

The users access the chips control and configuration registers to setup the chip to be reception-mode, transmit-mode or wake-up-model by a three-wire serial interface [17].

The AT86RF211 adopts FSK modulation technique, so it is the different frequencies that represent different information for transmission. For transmission mode, it needs two different frequencies registers (F0 & F1), or (F2 & F3) to represent code “0” and code “1”.

In reception mode, only one frequency needs to be programmed. The difference between F0 and F1 or F2 and F3 is two times the deviation. In reception, the chip tunes to the center frequency (F0+F1)/2 or (F2+F3)/2. Because the deviation is very small for instance the center frequency is 868.8 MHz, but the deviation is only 4 KHz, the antenna can pick up the packets operating on F0 and F1 or F2 and F3.

3.3 CC1020

The CC1020 [18] that is designed by Chipcon in 0.35 um CMOS is a single chip transceiver. Figure 3.3 [18] shows the block diagram of this chip.

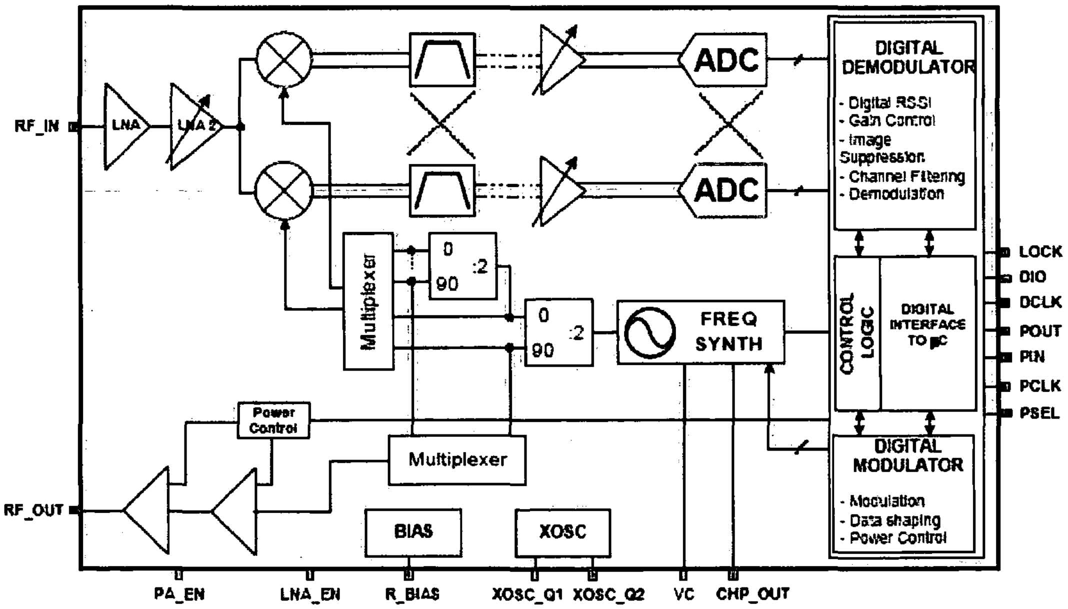


Figure 3.3 Block Diagram

Compared with other wireless chips, the CC1020 is especially suited for narrowband systems, for instance with channel widths of 25 KHz operating under EN 300 220(European Standard (Telecommunications series)). CC1020 can operate on frequencies between 424 – 470 MHz and 848 - 940 MHz.

The manufacturer claims that typical applications for the CC1020 are one-way and two-way RF systems that require narrowband such as wireless alarm and security, low-power telemetry, remote keyless entry and home automation systems.

The CC1020 is configured via a simple 4-wire SPI-compatible interface (PDI, PDO, PCLK and PSEL) [18].

3.4 Differences and Common properties

According the above information, table 3.4 is produced to show the key features of the three radio frequency chips.

	Frequency range	Data rate	Data modulation	Suitable for frequency hopping systems	Wake-up mode	Transce-iver
AT86RF211	400MHz-950MHz	up to 64bp/s	FSK	Yes	Yes	Yes
CC1020	424MHz-470MHz And 848MHz-940MHz	up to 153. 6bp/s	ASK, FSK and GFSK	Yes	Yes	Yes
BiM-418-40	418MHz	up to 40kb/s	diect transfer of analogue data	No	Yes	Yes

Table 3.4 Properties of Three Radio chips

3.4.1 Encoding Scheme

There are three coding methods used in both of AT86RF211 and BiM-418-40 for transmission. All the three coding method have the 50:50 mark:space, as this mark:space ratio will help chip to get stable average voltage from the signal.

The first scheme is bit coding. Each bit is composed of two parts. The first part is the bit and another part is complement. A "1" is composed of 1 and 0 and a "0" is composed of 0 and 1. The order of 1 and 0 can not be changed. The correctness of the transmission and 50:50 mark:space are guaranteed by this way that is also called Bi-phase or Manchester coding. Each Manchester bit is composed of bit "1" and its complement "0" or bit "0" and its complement "1" (figure 3.5). By this way, the mark:space will be 50:50. On the other hand, the Manchester bit rate is half of the general bit rate because two general bits responds one Manchester bit.

Two general bits: "1" and "0"

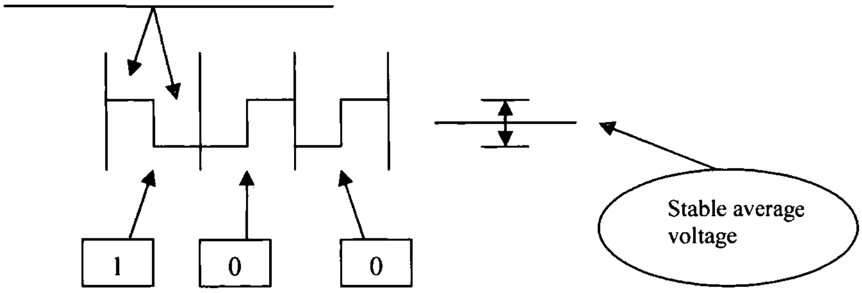


Figure 3.5 Manchester coding

The second one is byte coding. In one byte of this mode, each byte must contain four zeros and four ones which can guarantee the mark:space to be 4:4 equaling to 50:50. There are 256 possible combinations of 8 bits, but only 70 combinations satisfy the condition. That means the patterns used the byte coding are limited.

The below table 3.6 [19] has showed all the codes except 0Fh & F0h for minimizing consecutive 0 or 1's.

17	1B	1D	1E	27	2B	2D	2E	33	35	36	39	3A	3C	47	4B	4D
4E	53	55	56	59	5A	5C	63	65	66	69	6A	6C	71	72	74	78
87	8B	8D	8E	93	95	96	99	9A	9C	A3	A5	A6	A9	AA	AC	B1
B2	B4	B8	C3	C5	C6	C9	CA	CC	D1	D2	D4	D8	E1	E2	E4	E8

Table 3.6 70 Combinations

The data rate in byte mode is one eighth of the general bit rate because eight general bits is coded to one byte (Figure 3. 7).

Eight general bits: 10010110

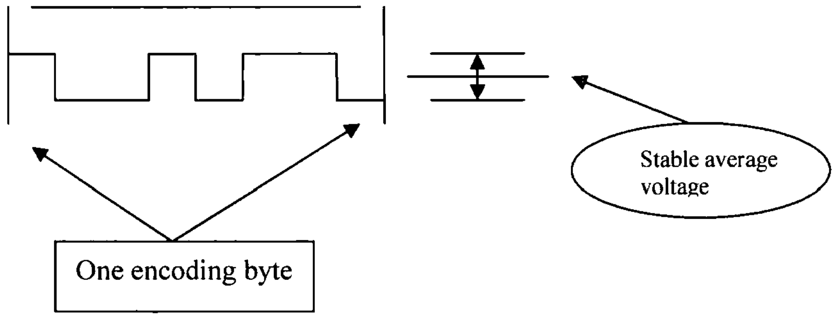


Figure 3.7 Byte coding

For byte coding, as only 68 of the possible 256 states are used, it is approximately equivalent to using only 6 out of the 8 bits for data. For bit coding, the efficiency is 50% (1/2) and for this coding, it is approximately 75% (6/8).

The last method is FEC (Forward Error Correction) coding clamed by chip manufacturer. The transmitted data unit is a byte and each byte is send twice. The first time is to send the byte and the second is to send its complement. This coding method increases the ability to prevent the isolated burst errors. The receiver adds the byte and its complement. If the result is not equal to zero, the error is detected. The receiver may drop the byte and send a message to request retransmission or drop the byte and wait for retransmission.

The data rate in FEC coding is one sixteenth of the general bit rate because transmitting sixteen bits just finishes transmit one FEC coding data (Figure 3.8). The efficiency is also 50%.

Sixteen general bits: 10010110 and its complement 01101010

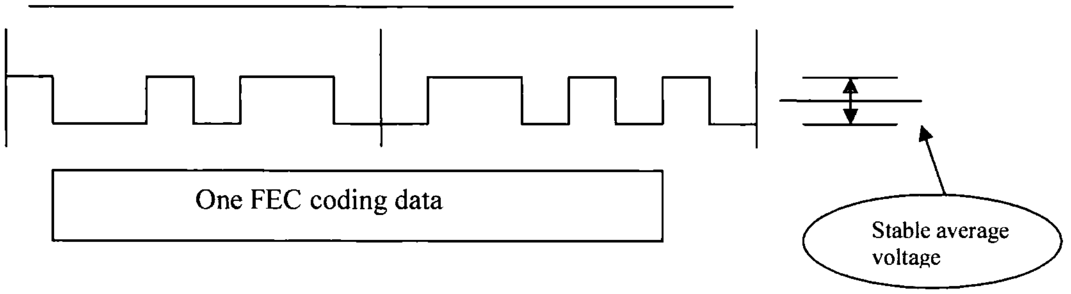


Figure 3.8 FEC Coding

The CC1020 uses NRZ (Non Return Zero) coding mode to modulate data besides the Manchester coding mode.

NRZ coding: The high-level volt is encoded to "1" and the low-level volt is encoded to "-1"figure 3.9.

The efficiency is 100%.

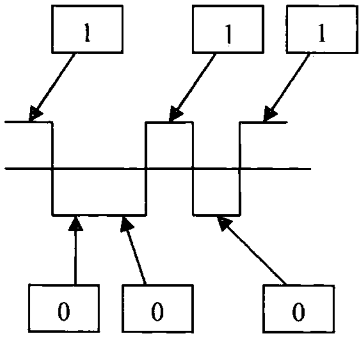


Figure 3.9 NRZ Coding

3.4.2 Wake-up mode

Beside transmission mode and reception mode, there is another mode called wake-up mode used in these chips. Wake up mode belongs to reception mode. In this mode, the chip will be in a sleeping state that is very low power mode. The internal timer of the chip will wake up itself periodically to receive expected transmissions. If it picks up a correct message, it will stop current mode and change to reception mode or transmission mode. The purpose for designing wake up mode is to save power consumption. For example for the AT86RF211 in reception mode the supply current is 29mA, but in sleeping state only 3uA. That is one reason for the battery operated systems adapting this type of chips. The main flaw of the wake up mode is time delay. If one device want to communicate with another device in sleeping state, that device has to wait for it to wake up by receiving a wake up message.

3.4.3 Wake up message format

Different chips may have different wake up message format. For the AT86RF211, the wake up message is composed by four parts: header (10 bits), address (from 0 to 20 bits), data (up to 32 bits), and stop (4 bits). Only the header is mandatory and other three parts are option.

Each message just contains one header. The receiver begins to detection the message until the message aborts or ends. If a message has more than one header, the second header would be considered as the address of the message and the message will be lost. On the other hand each message must be sent each time.

The Header is a predefined sequence which is '1010100001'. The header can not be changed and acts like synchronization. The Header is a recognition word (that is required to start any wake-up procedure), also used to lock the target frequency.

The data length can be fixed or variable. By the first way, data length is set from 1 to 32 bits without using stop field. By the second way, the data length is variable from 1 to 32 bits with using stop field.

The stop field is also a predefined sequence. If the data length is variable, the sender must add it after the last data bit of the data. It can be '1111' if the last data is '1' and be '0000' for the last data '0'.

3.5 Simulation

The simulation should be very careful of some parameters such as preamble. The functions of preamble have been discussed above. Although it does not have the same affection in the simulation, a preamble should be added into the packet timing. The main reason is that preamble also consumes transmission time and reception time. These may affect the precision of simulation time. The states of the radio chips are also very important. During the sleeping state, the device may not receive any signal.

Some radio chips such as BiM-418-40, CC1020 depend on the software to detect the information format, but some radio chips such as CC2420 not only receive the radio signal and convert it into electrical signal, but also detect some parts the packets format by themselves. The more details about simulation will be discussed in next chapter.

Chapter 4 Simulation Tool

4.1 Advantages of software simulation

Software simulation has developed very rapidly over the past few decades. It has many advantages for development over trying to work with real system for testing and evolution of designs.

a) Compared with the cost and time in installation and testing of a real system, the cost and time of a software simulation are lower and shorter. The larger the number of devices in a real system, the higher the cost will be. In the case of simulation, the cost of an additional instance of a piece of equipment which has already been modelled is negligible.

At the same time, simulation is much more readily changed than a real system. For example the position of each device is simulated by parameters that are easy for users to set and change, but in real system large number of devices may make the installation complex and costly.

b) In a software simulation, results are easier collected than in a real system. For testing, evolution and analysis of a real system, data needs to be collected from many points in the network, ideally, at the same sample instance. For simulation and real systems, results can be displayed in any way designer wishes, but in a real system it is harder and more expensive to collect the results. For example in a real system which includes ten radio devices, there may be thousands messages exchanging per minute. Users have to check the record of the device one by one for collecting the results, but the software can do that by printing the wanted results into one word document, excel worksheet or other forms for analysis. Second the simulation results can be reproduceable, but real networks do not have this property.

c) Software simulation can help designer to improve the systems and test new ideas. First when a real system is operating, some problems that are unpredictable to the designers may be caused. Designers have to reinstall the system to attempt to correct the problems. Simulation is a better way to solve this kind of problem. Based on the property of software simulation mentioned in point b, problems can be identified by analysing the simulation results. Second it is much easier for a designer to test new function in software simulation than to do it in practice. A real system may install new hardware for new function, but the software may achieve the same purpose by building a new model or adjusting some parameters. For example a radio network including five radio devices uses CSMA/CA (discussed in chapter 6) access method to access the radio channel and someone produced another access method which may be better than CSMA/CA. Designer may spend few days or weeks in adding or changing some parts of his/her simulation for testing the new communication protocol in software simulation, but if the designer wants to do the same testing in the real networks, he/she may need the new radio devices which adopt the new protocol for testing.

d) Software simulation also can be used for training. Engineers can change the parameters to see the change of the simulation results from which engineers can know which parts are important for the systems and which ones can be ignored. Engineers can understand how to operate the real system easily by using the simulation.

e) Software simulation can reduce the risk of investment. For instance simulation results of networks can show the possible traffic, the numbers of messages exchanged and the power consumption (important for battery powered equipment) of devices in months or years. According to these results, investors can calculate the detail costing and charging algorithm for their users to avoid fail use of their investments.

4.2 Network Simulator (NS)

There is a popular public domain network simulation tool named network simulator. NS is developed by the Network Research Group at the Lawrence Berkeley Nation Laboratory. It is developed under C++ and TCL (Tool Command Language). Users can create a simulation network such as network topology and traffic source by using the NS commands. Both wired and wireless network and related protocol such as TCP can be simulated by NS. There are more details about NS in [20] and [21].

Although NS is a good network simulator, it was felt to have some disadvantages for this work.

1) NS is a very general simulation tool. Google website will list hundreds of pages about the articles that use NS to do simulation.

2) NS is a large system. Only the installation program files have been approximate 144 Mbits.

3) It is complex to add new functionality. Before adding a new functionality, users should be familiar with using NS, the structure of NS and both of C++ and TCL languages. As mentioned in point 2, NS is a big system, so it may take users considerable time to add new functionality.

In order to deeply understand the network structure, related protocols and avoid the disadvantages of NS for this thesis, a new simulation tool developed under C++ is created.

4.3 C++ computer language

C++ computer language written by Bjarne Sroustrup at Bell Labs during 1983-1985 has been widely used in academia and industry since then. It is one of the dominant computer languages commonly used by programmers to simulate communication systems.

a) It includes a lot of useful libraries.

b) It supports object-oriented programming. In C++ language the target real world entity is represented by an object that contains necessary information about the entity to be simulated. The necessary information is in the form of data， and functions known as methods or codes to manipulate the data. For instance, in order to simulate the radio networks a radio device is represented by a related object which contains the frequency used, position, receiving function, received power calculation and transmission function. The object used in the simulation of local sensor system in chapter 5 is as following:

```
 class DEVICE
{
  private:
       static DEVICE *listhead;
       DEVICE *link;
 protected:
       char   name[20];
       double fl[N];
       double fh[N];
       int   chanid[N];
       void linkin(void);
       int status;
```

```
      double  timeout;
      struct communication_list cmu;
 public :
     struct property  r;
     virtual void timestep(double, double, double);
     virtual void transmitting(char*, char , double );
     virtual void receiving(char* ,double ,double);
     void setTP(double,double,double);
     void setfrequency(int,double*,double*);
     void setchanid(int,int);
     void setname(char *);
     char  *getname(void);
     int   getchanid(int);
     int   checkid(int);
     int   getstatus(void);
     static DEVICE *finddevice(char *);
     static DEVICE *getlisthead(void);
     DEVICE *getlistnext(void);
     DEVICE(void);
 };
```

c) It supports new class for data abstraction. A class is an extension to the concept of structures in C. The main difference between class and structure is that class contains code which a structure does not. In a class the members are divided into private, protected and public members. The instances of a class type are called objects. An example is shown in b above. More details about class can be read in [22].

d) Inheritance: programmers can write new classes by deriving from a previous one. The derived class inherits all the member variables and the member functions of the base class or classes. This capability promotes code re-use. For example in the simulation of a simple system, class BASE_STATION is a derived class of base class DEVICE.

```
class BASE_STATION : public DEVICE

{

public:
    BASE_STATION(char * , char *) ;
     void  timestep(double, double ,double);
     void  transmitting(char * , char ,double);
     void  receiving(char*,double, double);
} ;
```

e) It provides function overload. In C++, several functions in a particular scope can be declared with the same name. The compiler determines by the type and number of the parameters which function is to be called. For example a class may have two member functions name testing (). The parameters of the first testing can be (char s[10], char t, double y), and the second testing function may be (char s[10], char t, int y).

f) It provides virtual function which implement late (run time) binding. A virtual function is a function member of a class declared using the virtual keyword. A pointer to a derived class object may be assigned to a base class pointer, and a virtual function called through the pointer. The most prominent reason why a C++ virtual function will be used is to have a different functionality in the derived class. It can be called from the base class, without the base class even knowing the number of derived classes. For instance, in base class DEIVICE shown in b above, there are three declared virtual functions.

```
    virtual void timestep(double, double, double);
    virtual void transmitting(char*, char , double );
    virtual void receiving(char* ,double ,double);
```

In class SLAVE:

```
class SLAVE : public DEVICE
{
```

```
public:
      SLAVE(char *com , char *line ) ;
       void  timestep(double, double, double );
       void transmitting(char * , char ,double);
       void  receiving(char*,double,double);
} ;
```

In class BASE_STATION:

```
class BASE_STATION : public DEVICE
{
public:
      BASE_STATION(char * , char *) ;
       void  timestep(double,double ,double);
       void  transmitting(char * , char ,double);
       void  receiving(char*,double,double);
} ;
```

When the function is called such as timestep () in main (), program accesses the object by pointer and does not know whether the object is a slave or the base_station.

```
for (p=DEVICE::getlisthead(); p!=NULL; p=p->getlistnext())
   {
    p->timestep(final_time, time_slot, current_time);
   }
```

More details about C and C++ can be found in [23], [24] and [25].

4.4 Simulation Structure

4.4.1 Device

In this simulation tool, device is the base class. The derived classes are named slave, base station or controller. In chapter local sensor system there is no controller and slaves are waiting for base station's command to access the channel. It means the base station decides which slave is to use the channel. In chapter 6 WLAN case study, the 'base station' becomes the access point, proving the gateway between the wired and radio networks. The slaves become stations defined in 802.11 and access the channel by CSMA/CA. The Controller is connected to the base station by a wired network. The communications between stations and controller are via base stations. More details about case study are discussed in chapter 5 and chapter 6.

Figure 4.1 shows the basic properties of a device

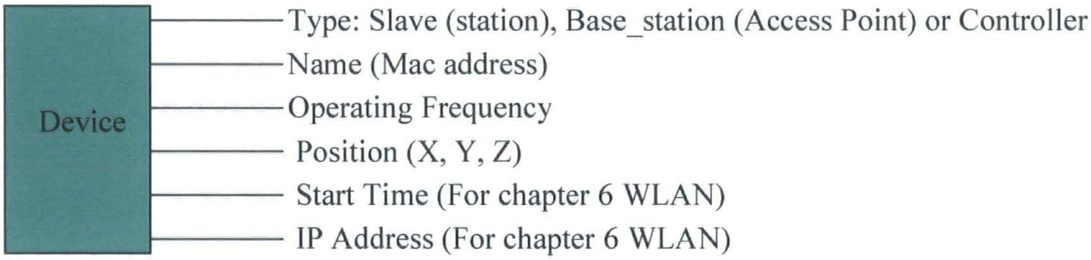

Device
Type: Slave (station), Base_station (Access Point) or Controller
Name (Mac address)
Operating Frequency
Position (X, Y, Z)
Start Time (For chapter 6 WLAN)
IP Address (For chapter 6 WLAN)

Figure 4.1 Device Setting

The member functions and all parameters of a device used in WLAN simulation are following:

```
class DEVICE{
private:
      static DEVICE *listhead;
```

```
     DEVICE *link;
protected:
     char  name[20];
     char  ip[40];
     double fl[N];
     double fh[N];
     int   chanid[N];
     void linkin(void);
     double self_time;
     int status;
     double  timeout;
     struct communication_list cmu;
     message *self_msg;
public :
    struct property  r;
    virtual void connect(DEVICE*,int);
    virtual void PHY_layer(message*,double,double ,double);
    virtual void DL_layer(message*,double,double ,double);
    virtual void IP_layer(message*,double,double ,double);
     virtual void Trans_layer(message*,double,double ,double);
    virtual void ALY_layer(message*,double,double ,double);
     virtual void BC_transmitting(message*,double,double,double);
     virtual void BC_receiving(char*,double,double);
virtual void timestep(double, double, double);

virtual void transmitting(message*,double,double,double);

virtual void receiving(char* ,double ,double);

virtual  member_list getmeb(void);

void setTP(double,double,double,double);

void setfrequency(int,double*,double*);

void setchanid(int,int);

void setname(char *);

void setip(char *);

char  *getname(void);

char  *getip(void);

int    getchanid(int);

int    checkid(int);

int    getstatus(void);

double MTM (double , double);

static DEVICE *finddevice(char *);

static DEVICE *getlisthead(void);

DEVICE *getlistnext(void);

DEVICE(void);

   };
```

4.4.2 Commands for creating simulation

CREATE CONTROLLER C1 -------> A controller named C1 is created.

CREATE BASE_STATION AP1 -------> A base station (access point) named AP1 is created.

CREATE SLAVE S1 -------> A slave (station) called S1 is created.

SETIP C1 193.168.168.1 -------> The IP address of C1 is 193.168.168.1.

SETFREQ AP1 0 2424.5 2449.5 ------->The high and low frequencies for channel [0] of the AP1 are set to are 2424.5 & 2449.5. Every device has three channels for using in the model, but only one is used in this study.

SETTP AP1 2 3 4 0-------> The position of AP1 is (2 m, 3 m, 4m) and the time for AP1 starting is at 0 s.

CONNECT AP2 0 C1 1 -------> Communication between access point and controller is by wire. This command means that the port [0] of AP1 connects the port [1] of C1. This command is only used in chapter 6 WLAN.

START 0 0 0.16 0.000001 -------> (0 0 0.16) represents (hour/ minute/second). The sum of the three values is the total simulation time. Here the simulation time is 0.16s. The last parameter 0.000001s is the simulation time step .

The controller used in this thesis is assumed to be wired device, but in other system it may also do transmission and reception by wireless. Due to this reason, the wireless property is also kept in controller for future study.

4.4.3 Transmission and Reception

Figure 4.2 draws the distribution of a radio system that contains five radio devices

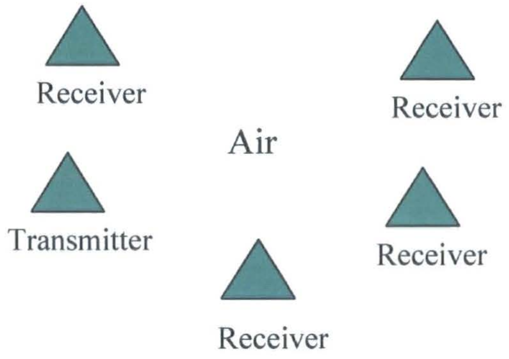


Figure 4.2 Distribution of a radio system

1) In Transmitter:

Each device has a time function called timestep() which checks whether the device wants to do transmission at each time step.

If a device wants to transmit information, a message structure for carrying the information will be created by the related layer and passed to lower layers (figure 4.3).

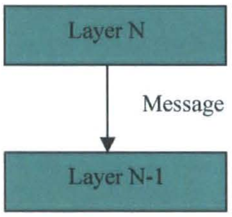


Figure 4.3 Message Passed from N Layer to (N-1) Layer

Each layer has layer functions to deal with the message. For example in WLAN case each device has five layers (application layer, transport layer, network layer, data link layer and physical layer). If the message is created by application layer such as data message and passed to transport layer, the transport layer will add port number into the message and pass it to the lower layer- network layer. The data link layer will calculate the message transmitting time by CSMA/CA (mentioned in chapter 6). The timestep function will check the transmission time. If it is time to transmit the message, transmitting function () will be called. That function will put the message into the message list.

2) Air:

The air function is simulated in main (). In main() a function message list will be checked by every time step. If it is time for one message to start transmission, a channel list will be built according to the message. According to the source name of the message, the device is found out to be the transmitter in channel list and then other devices operating on the same frequency will be the receivers in channel list (figure 4.4).

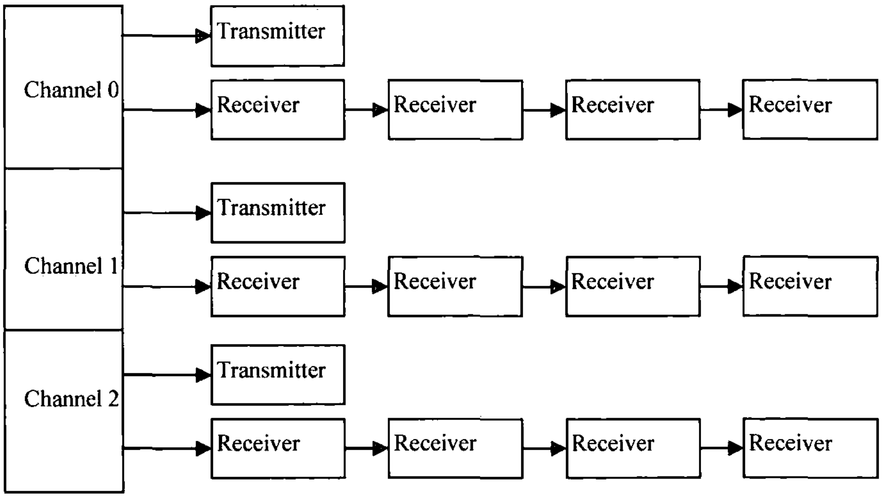


Figure 4.4 Channel Structure

According to the channel list, Collision function will check whether there is more than one device using the same channel. If there is, the messages will be corrupted by each other; adjacent channel interference function will check whether there is interference. More details about interference are introduced in local sensor system study.

If there is noise in message transmission time, the message will be corrupted by the noise.

3) In Receiver:

At the end of transmission, receives in channel list will pick up the message by receiving function. The message is passed to the lowest layer called physical layer which calculates the received power. If the received power is bigger than power sensitivity, the message will be passed to upper layer- data link layer (figure 4.5). Data link layer will deal with the message in related function such as detecting error and checking the MAC address. If there is an error in the message, all receivers will drop the message. If the receiver is the destination of the message, the device will pass the message without errors to upper layer. If it is not, the receiver will drop it.

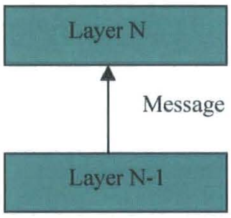


Figure 4.5 Message Passed from (N-1) layer to N layer

Depending on the protocol device using, receiver may send reply to the transmitter or not. If the receiver wants to send a reply, it will become a transmitter in transmission time.

4.4.4 Flow chart

Figure 4.6 and figure 4.7 show the flow of a message from transmission and reception.

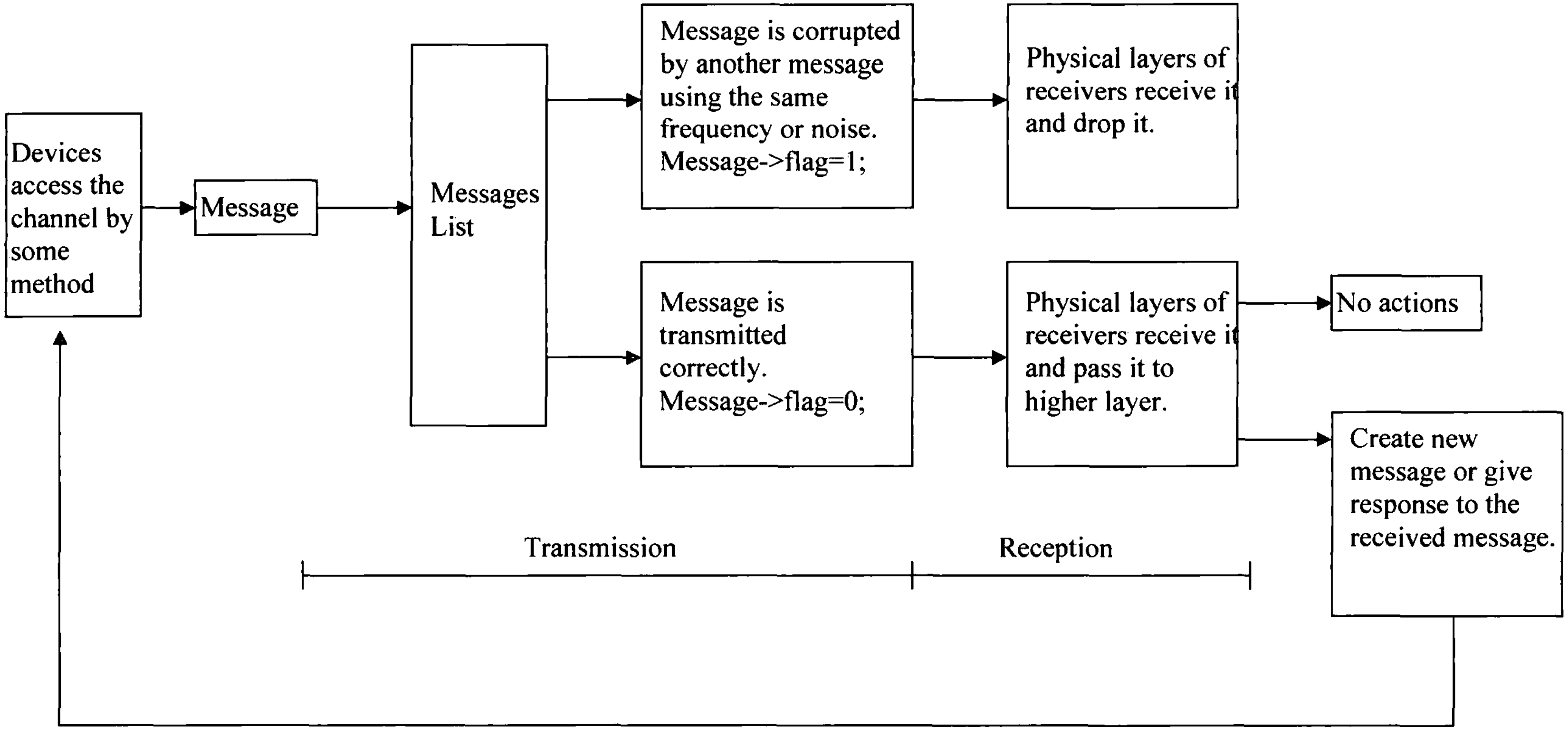

Devices access the channel by some method
Message
Messages List
Message is corrupted by another message using the same frequency or noise. Message->flag=1;
Physical layers of receivers receive it and drop it.
Message is transmitted correctly. Message->flag=0;
Physical layers of receivers receive it and pass it to higher layer.
No actions
Create new message or give response to the received message.
Transmission
Reception

Figure 4.6 Flow Chart 1

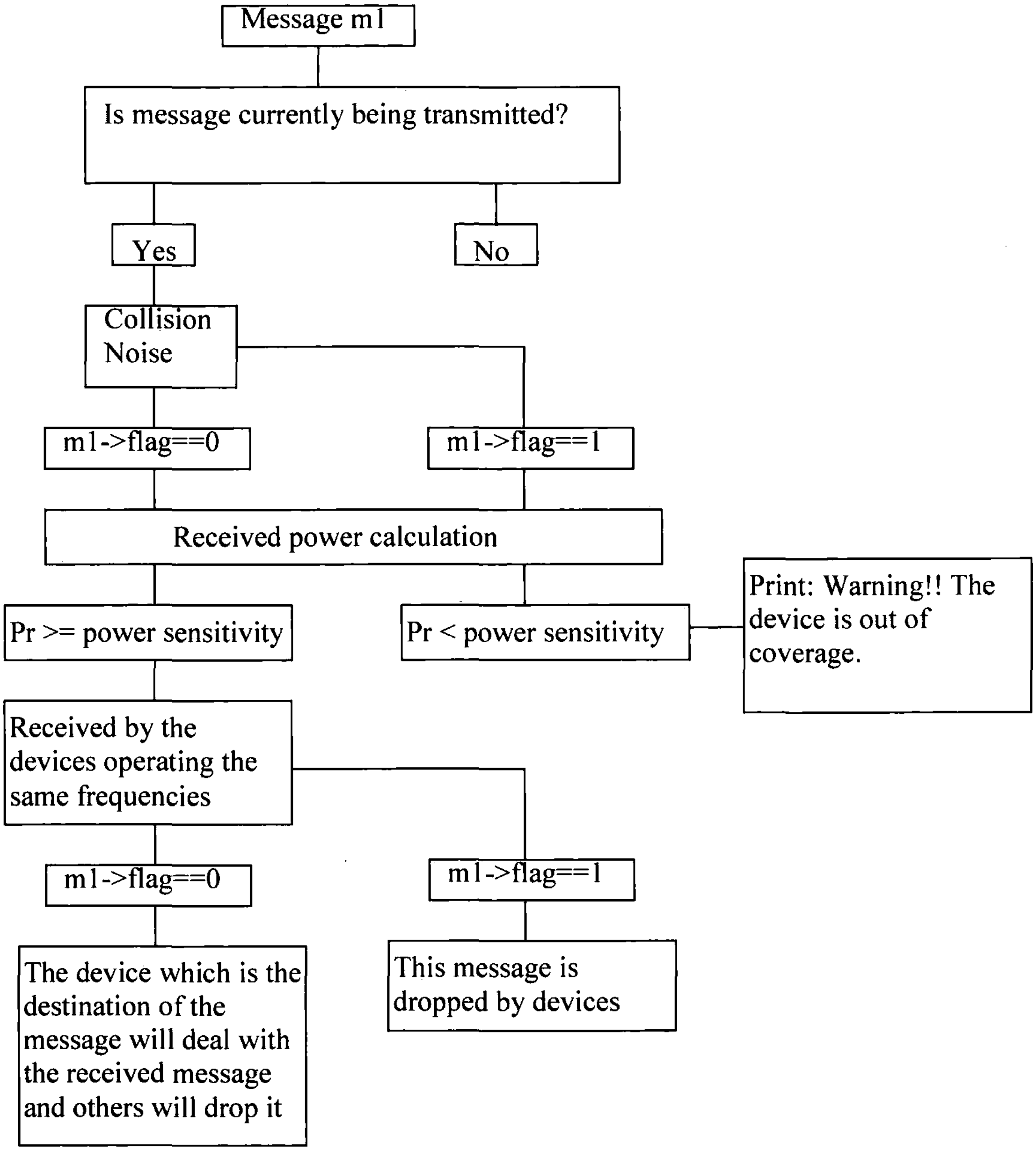

Message m1
Is message currently being transmitted?
Yes
No
Collision Noise
m1->flag==0
m1->flag==1
Received power calculation
Pr >= power sensitivity
Pr < power sensitivity
Print: Warning!! The device is out of coverage.
Received by the devices operating the same frequencies
m1->flag==0
m1->flag==1
The device which is the destination of the message will deal with the received message and others will drop it
This message is dropped by devices

Figure 4.7 Flow Chart 2

4.4.5 Display of result

C++ complier provides some commands such as 'grep' for getting the simulation results. For example, users can put all the results into a temporary file whose name is also chose by users by typing "sim wireless.cfg>result.cpp". Wireless is the configure file name and result is assumed to be the temporary file named result. Then users get the target device result such as station 1 by typing “grep S1 result.cpp>S1.cpp”. After that file S1.cpp will contain all communication process of device 1 such as registration. If users just want see the data transmission and reception, they need to type “grep Data_Reply S1.cpp>DR1.cpp”. The file DR1.cpp will show how many Data_Replys the station 1 received in total. According to these results, related tables and figures are produced such as Table 6.23.

The simulation program is not trivial. For example, the simulation program of chapter 6 WLAN is approximately 3250 lines in total.

Chapter 5 Local sensor system simulation

5.1 Introduction

This section considers the simulation of the local sensor systems such as wireless alarm systems or radio security systems.

5.1.1 Physical distribution

The devices distribution is shown in figure 5.1.

■ Base station

▲ Slave

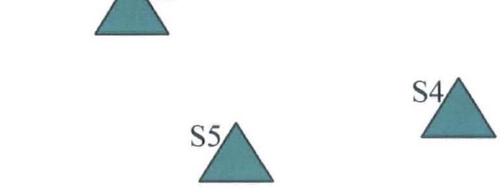


Figure 5.1 Physical Distribution

The simulation scenario chosen consists of one base station and five slaves operating on one channel. Each device only has three layers that are application layer, data link layer and physical layer. The system is not trivial, but the results are not too complex to interpret.

5.1.2 Configure file

```
CREATE BASE_STATION  B1
CREATE SLAVE S1
CREATE SLAVE S2
CREATE SLAVE S3
CREATE SLAVE S4
CREATE SLAVE S5

SETFREQ B1 0 868.1 868.3
SETFREQ S1 0 868.1 868.3
SETFREQ S2 0 868.1 868.3
SETFREQ S3 0 868.1 868.3
SETFREQ S4 0 868.1 868.3
SETFREQ S5 0 868.1 868.3

SETTP B1  2 3 4
SETTP S1  5 7 4
SETTP S2  7 9 3
SETTP S3  6 8 2
SETTP S4  3 7 4
SETTP S5  2 5 3

START 0 0 50 0.001
EXIT
```

CREATE SLAVE S1 -------> A station (slave) named S1 is created.

SETFREQ S1 0 868.1 868.3 ------->The high and low frequencies for channel [0] of the S1 are set to 868.1 & 868.3 (MHz). Every device has three channels for using in the model, but only one is used in this study.

SETTP S1 5 7 4 -------> The position of S1 is (5 m, 7 m, 4 m) and the time for S1 starting is not set here. All the devices will start to work at the beginning.

START 0 0 50 0.001 -------> (0 0 50) represents (hour/ minute/second). The sum of the three values is the total simulation time. Here the simulation time is 50s. The last parameter 0.001s is the time step.

5.1.3 Communication protocol

The communication protocol used in this system is very simple. The Base station sends a message to the slaves and waits for the response from an addressed slave, setting a time out to allow for the detection of lost or corrupt messages. If the addressed slave receives the message without errors, it will send an ACK message back, which also contains any relevant status information.

When the base station correctly receives the ACK, one communication sequence is successfully finished. If some errors are detected in the messages by the slave, typically by finding a bad CRC check, the slave will drop the message and wait for a new one. If there is no response from the target slave or the base station receives the ACK with some errors, the base station will send the message again triggered by the time out function.

a) Channel access method:

In order to avoid collisions, the base station polls each slave in turn. It means that only one slave can use the channel at a time. Communication is between the base station and the slaves.

b) Timeout:

The timeout is made up of two parts. The first part is twice the message transmission time plus a time for the device to process message and the second part is a random time for avoiding repetitive noise (introduced later). The random time value makes the time out value to be not constant.

c) Collision:

If there is more than one device to transmit on the same channel at a particular time, it may cause collision. Due to collision, receivers may receive a message with errors or lose the message. If there is no responce from the receiver, the transmitter will retransmit that message by time out function. The collision can be classified into three main categories, beginning, middle, and end collisions.

Beginning collision may happen when there is more than one device adopting the same frequency to start transmission at the same time.

Middle collision may be caused by one message's starting transmission time which overlaps with the transmitting time of another transmitting message and both of them operate on the same frequency.

End collision may happen when one message's start time is the same as another's end time and both of them are carried by the same frequency.

If there is a weak noise during the transmission time and the transmission power is very strong, there may be no middle collision and receivers can get the message correctly.

If there is a weak noise before the transmission, the beginning collision may be caused and the transmitting message may be corrupted.

In order to guarantee the quality of the radio communication and effective usage of the wireless resources, the communication networks should try their best to reduce collisions.

5.1.4 Message format

```
struct message
{
char  sourcename[20];
char  destination[20];
int   flag;.
char   stream;
double length;
double start_time;
double end_time;
};
```

Sourcename: It is the physical name of transmitter.

Destination: It is the physical name of the target receiver.

Flag: If it is set to 0, it means there is no error in message. If it is set to 1, it means there some errors that may be caused by noise, collision or interference.

Stream: It is the code that transmitter wants to send to receiver. According the code, the receiver can take related application.

Length: It is the length of message and calculated in byte.

Start time: It is the time when the message is to be sent.

End time: It is the time when the transmission ends.

5.2 No noise situation

For ideal communication environment there is no noise to affect the radio signal. The data rate is assumed to 0.8Mbit/s. The result is showed in Figure 5.2 No_noise. There is no failed transmission.

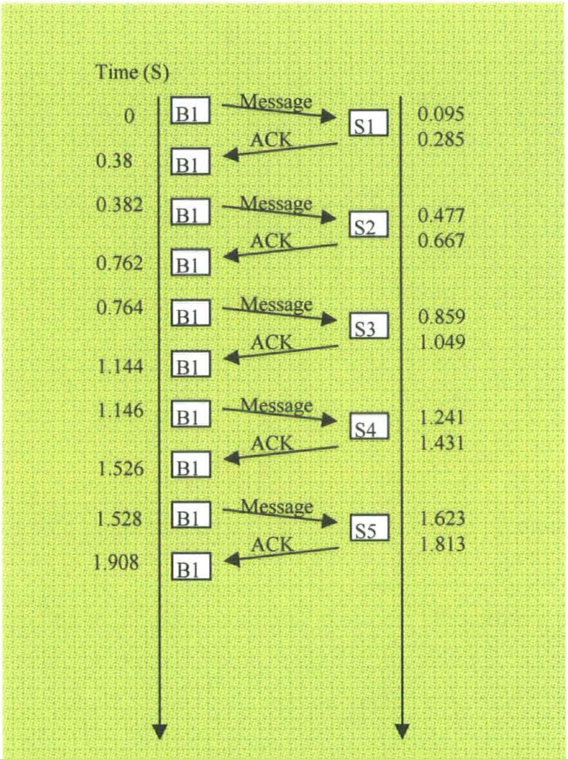


Figure5.2 No_noise

The message sent by base station to slaves may be a command which may ask slave to open a window or shut down a door. The slave detects whether the window or the door is close, does related command and creates an ACK. The time spent in doing above actions is assumed to 0.19s which is twice of the transmission time (2*0.095s). Base station receives the ACK sent by slave and then sends a message to another slave. The response time is very short and is assumed to 2 ms here. In this simulation the base station does not send messages to slaves periodically, and only performs one successful communication with each slave in whole simulation time. It means if the message is corrupted by noise, base station will retransmit it to the slave until it receives a correct ACK from the target slave.

5.3 Noise and interference

Noise such as repetitive noise and interference such as adjacent channel interference mentioned later and collision are two main factors affecting the radio communications. Messages may be corrupted by them. In order to be close to the real system, both noise and interference are simulated in this simple case.

5.3.1 Noise

For radio communication systems the unwanted radio signals are considered as noise. If the noise operates on the frequency that is in the wanted signal's frequency range, the noise will add to the wanted signal at various points between the source and destination. That mixed signal's voltages and phase will be different from the target signal, so the errors are caused by the noise. An example is given in the below figure 5.3.

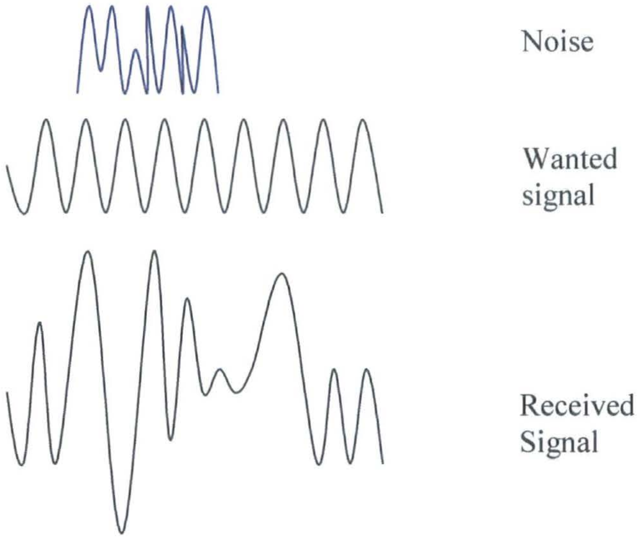


Figure 5.3 Noise Effect

a) Source

The source of noise is usually divided into two categories: external and internal noise.

The internal noise is produced by the receiver itself. The typical internal noise is thermal noise [26]. This type of noise is caused by the thermal movement of the free electrons and ions in a conductor. Due to the movement, the resistor's potential varies randomly. Because this noise is dependent on temperature, it is defined as thermal noise. It is also named Johnson noise or white noise [27].

The external noise is produced by the receiver's environment. Man-made noise and nature's noise are the main external noises.

Man-made noise is often produced by electric motors, engine ignition systems and power lines that supply the energy for most electronic systems within the receiver's antenna range. Man-made noise occurs randomly at frequencies up to minimum radio frequency.

Nature's noise consists of atmospheric noise and space noise. The atmospheric noise naturally happens in the earth's atmosphere. Lightning discharges are one of the most dominant instances of this. Its frequency content includes the entire radio spectrum but intensity is inversely related to frequency. Some testing shows that it does play a serious effect on the frequencies above 20MHz.

The space noise is generated by the stars in outer spaces. The typical example is solar noise caused by the sun. Space noise extends from about 15MHz up to 100GHz.

More details about noise can be read in [28] and [29].

b) Signal to Noise Ratio

In order to measure the noise effect to the signal, the signal to noise ratio is defined as a standard.

Signal/Noise Ratio = Signal Power / Noise Power;

The ratio is also expressed in decibel units. The conversion from watts to decibel:

Signal/Noise Ratio [dB] = 10 * Log10 (Signal Power [W] / Noise Power [W]) (1)

If the ratio is positive, it means the signal is more powerful than the noise; if it is negative, the noise will be more powerful than the signal and the receiver may not receive the wanted signal; in practice the signal to noise ratio should be big enough to overcome the effects of noise and this ratio is a limiting factor to the radio system. For example if a system using some 802.11b card wants to achieve an 11 Mbps data rate, the minimum signal/noise ratio (also named S/N ratio) is 16dB [30].

In this simulation study, it is assumed that the noise power from a distant noise source is constant over the communication area. That means no matter where the slave is, the noise will affect its radio communication.

5.3.2 Interference:

As well as noise, different radio systems interfere with each other.

Adjacent channel interference and near-far effect:

The strength of a radio signal decreases with the distance increasing between transmitters and receivers. An unwanted signal, transmitted close to a receiver, can corrupt a weak signal from a distance transmitter. This effect is named near-far effect. A radio system can be divided into several groups operating on different frequencies. Though devices in different groups use different frequencies at the same time, it may also cause interference called adjacent channel interference in figure 5.5 because of near-far effect. For instance in figure 5.4, transmitter T1 and receiver R1 operate on frequency F1 and transmitter T2 and receiver R2 use frequency F2. F1 is the adjacent frequency to F2 as shown in figure 5.5. If T1 sends message to R1 and T2 transmits information to R2 at the same time, receiver R1 may lose the signals sent by T1 or receive wrong message because of the adjacent channel interference of T2.

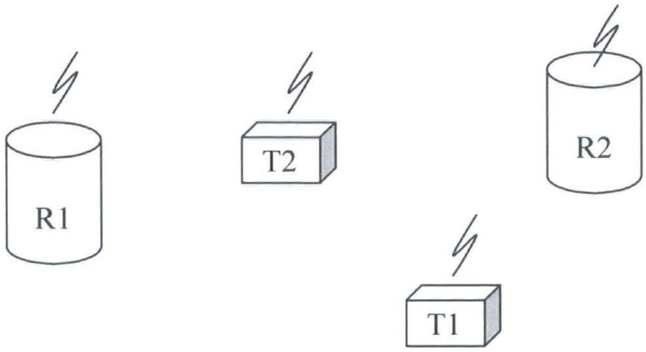


Figure 5.4 Near-far effect

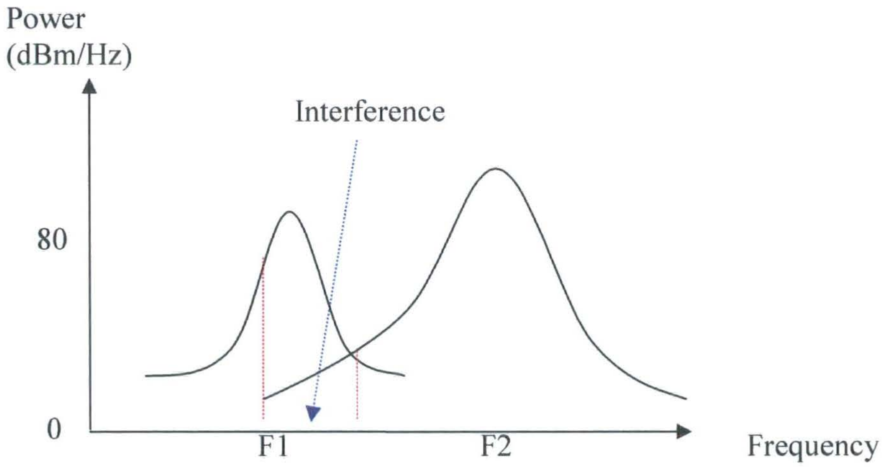


Figure 5.5 adjacent channel interference

Real systems allow adjacent transmission, but at a level typically more than 80 db below the power in the designated band.

In this simulation tool the noise is divided into two groups: repetitive noise for instance every two minutes there will be a noise causing by electric motors or engine ignition systems, and random noise such as thermal noise.

Adjacent channel interference is assumed to happen when the two channels are too close such that the difference between the two channels is less than 40 KHz in this example.

5.4 Repetitive noise situation

In this example the repetitive noise is added for simulating the real radio environment. It is assumed that the repetitive noise starts at 0.02 s, the period is 0.4s and every burst lasts 0.01 s.

Based on the simulation results, Figure 5.6 Re_noise (Repetitive Noise) is produced. The red line means the repetitive noise affected the communication in that time and the information was corrupted by the noise.

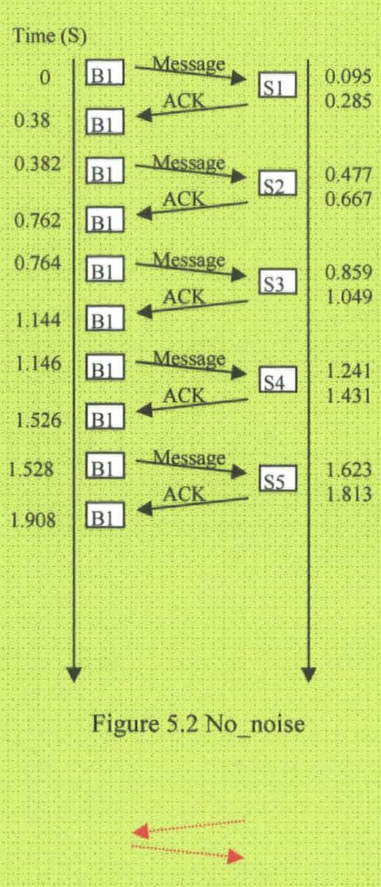


Figure 5.2 No_noise

The information is corrupted by noise.

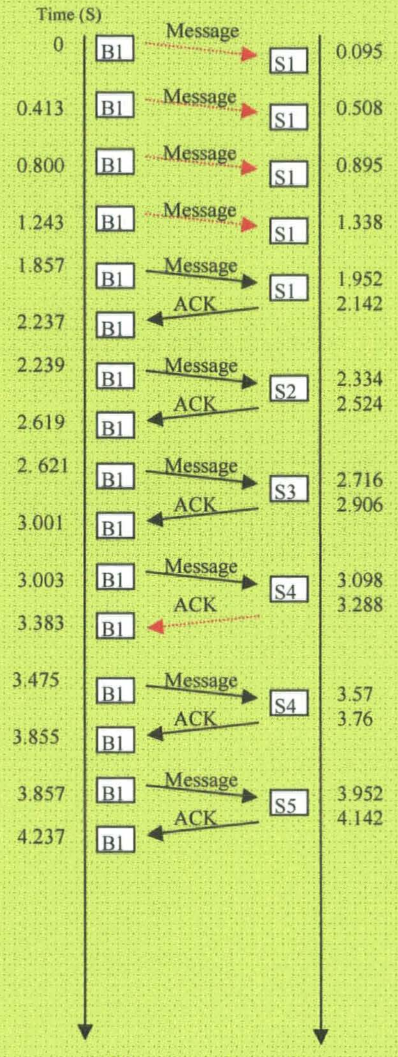


Figure 5.6 Re_noise

The first affected communication was about 0.095 s when base station was sending a message to slave 1. Slave 1 dropped the message with errors and gave no response. The retransmission was triggered by timeout function. When timeout expired, base station would retransmit the message at a random time that is the sum of the current time, one time step and a random time for avoiding noise or collision.

Random time= random()*one_time step;

Here random() will give a random value between 0 and CW (Contention Window). Contention window [31] is also called Backoff Window. According to wireless standard such as 802.11, here the range of CW is also set from 31 to 1023. The CW is 31 (2^5-1) at the first time for retransmission and is changed to 63 (2^6-1) at next retransmission time. If the CW is 1023, it will be changed to 31 in next retransmission.

Here the base station could not finish the communication with slave 1 until 2.237s. Before base station successfully finished the communication with slave 1, it failed four times. The situations for other slaves were better than slave 1. Finally base station spent 4.237 s in communicating with these five slaves.

For the no noise situation, the total communication time is only 1.908 seconds, but for this scenario with a repetitive noise it need 4.237 seconds to finish the same task. Time for transmission is extended and battery power was consumed by retransmitting the information.

5.5 Random noise situation

In this example the random noise replaces the repetitive noise. It may interfere with the communication at any time and also may not happen during the communication time.

Compared with the constant period of the repetitive noise, the period of the random noise is uncertain. Here the range of random period is set from 0 to 0.4s and every burst lasts 0.01 s.

First, in order to test whether the simulation tool can simulate random noise, simulation time is extended to 400s which is one thousand times of the maximum random period. Based on the results, random noise period distribution is drawn in figure 5.7.

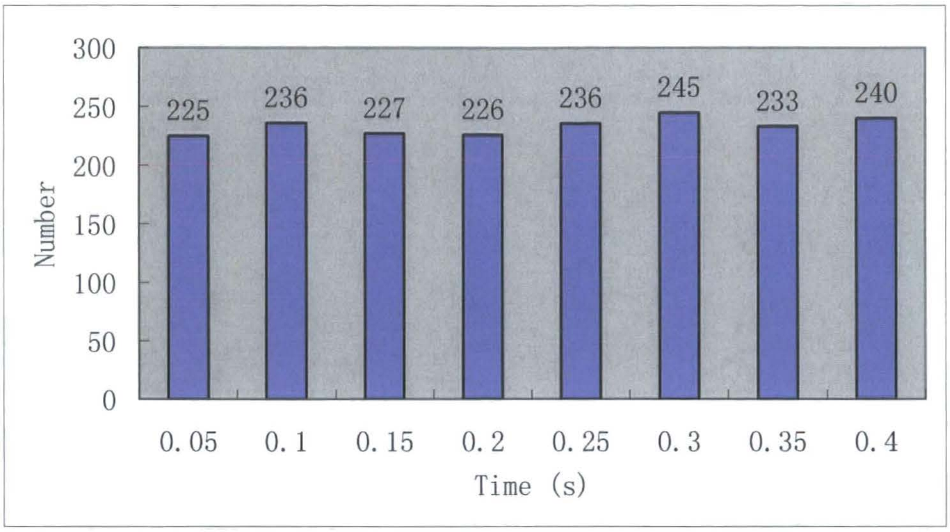


Figure 5.7 Random Noise Period Distribution

In figure 5.7 the X axis is the noise period categories and Y axis is the number of bursts whose period belonging to related the period category. Figure 5.7 shows the number of the noise period in different categories. For example there are 225 noises whose period are between 0 and 0.05s, 236 noises whose periods are between 0.05 and 0.1s, and 240 noises whose periods are between 0.35s and 0.4s. The number in each category is nearly same. It indicates the noise period is a random value.

Then the simulation time is changed to 50s which is the same as was used in the repetitive noise simulation. The whole simulation result is drawn in figure 5.8 Ran_noise (Random Noise).

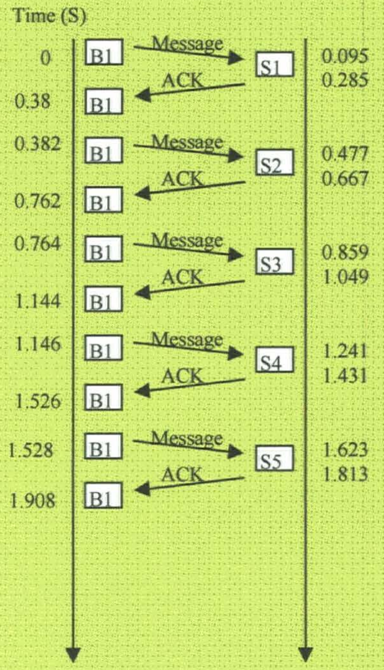


Figure1 No_noise

The information is corrupted by noise.

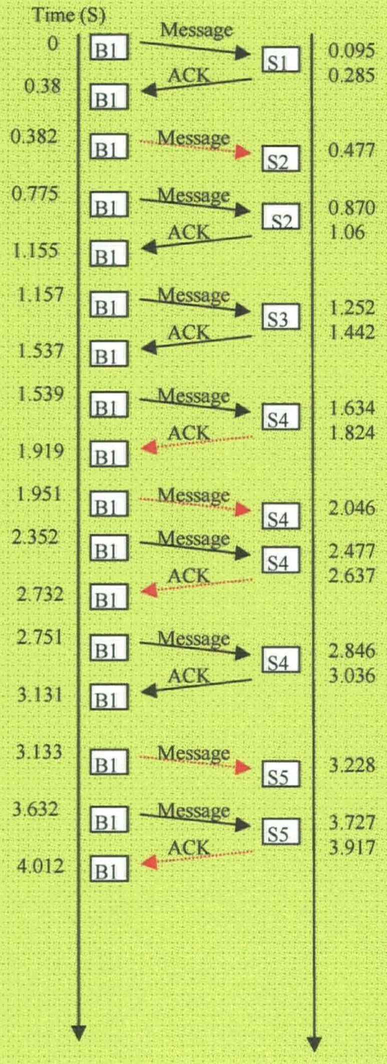


Figure 5.8 Ran_noise (a)

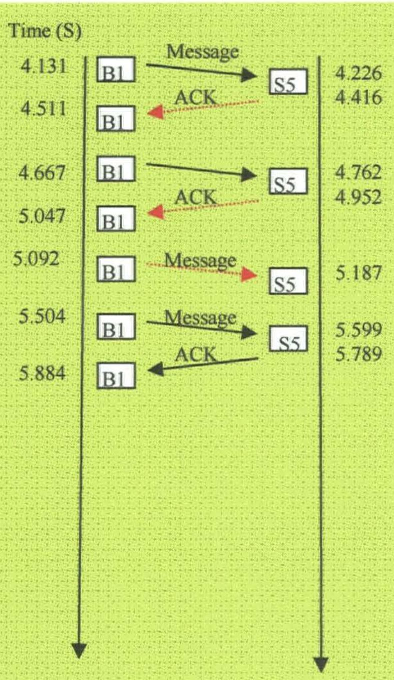


Figure 4 Ran_noise (b)

From the Figure 5.8 Ran_noise, the first random noise which corrupted the communication between Slave 2 and the Base station 1 happened between 0.382s and 0.477s. Slave 2 received a message with errors and dropped it. After a time out, the base station 1 sent message to slave 2 again at 0.775s and got the ACK sent by slave 2 at 1.06s. In this situation not all slaves' communications were affected by the random noise. Some slaves such as slave 1 and slave 3 successfully finished the communication with base station in the first time. Transmission and reception of slave 4 and slave 5 were frequently affected by random noise. They received a minimum of two messages damaged by random noise. In this situation, base station spent 5.884 s that is more than three times of the time spent in no noise situation in finishing communication with all slaves.

5.6 Adjacent channel interference situation

In this simulation scenario, the two channels are assumed to interfere with each other because of the adjacent channel interference. Because the devices operating on the licence free frequency band are usually low power transmitter and with high power sensitivity, here transmission power is set to 1 w and power sensitivity is set to -100db. Figure 5.9 devices distribution shows the physical position and received power of each device.

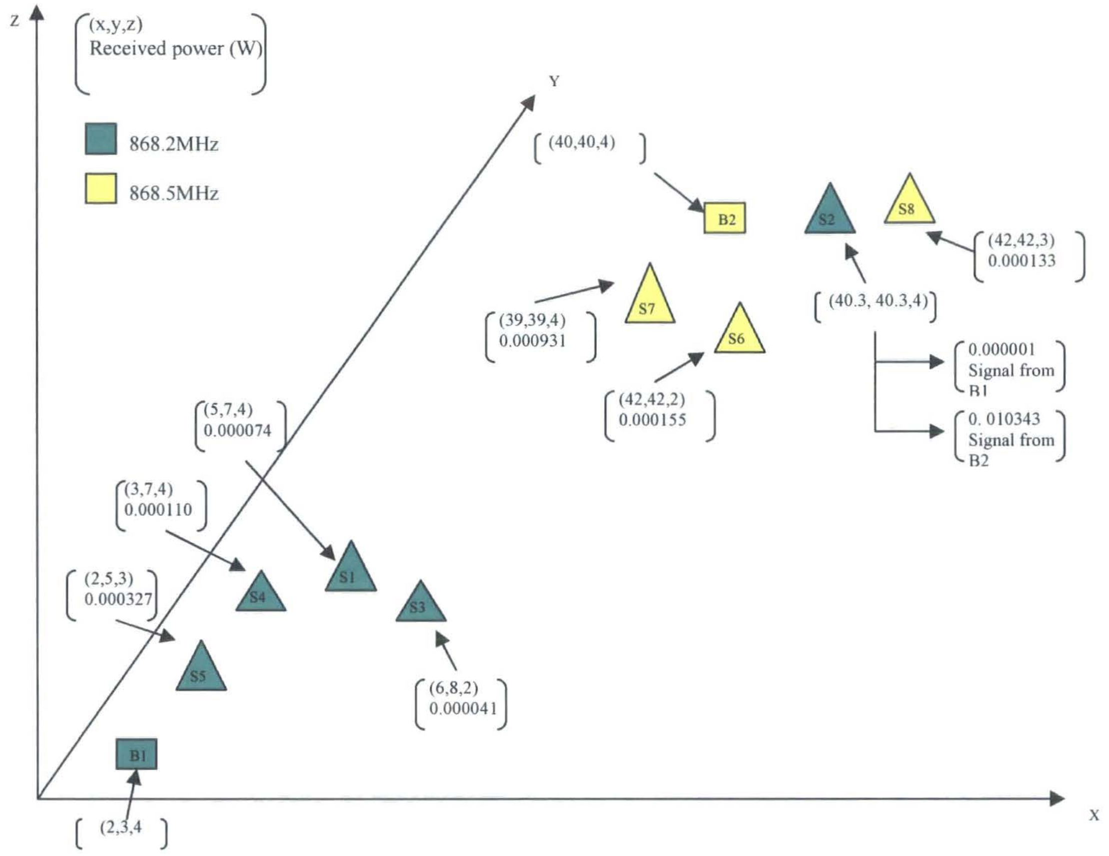


Figure 5.9 Devices distribution

For the above three examples (No noise, Repetitive noise and Random noise), the configure files are the same. The configure file of this example is different from them because of adding another base station and some slaves operating at a difference frequency band.

The related configure file is:

```
CREATE BASE_STATION  B1
CREATE BASE_STATION  B2
CREATE SLAVE S1
CREATE SLAVE S2
CREATE SLAVE S3
CREATE SLAVE S4
CREATE SLAVE S5
CREATE SLAVE S6
CREATE SLAVE S7
CREATE SLAVE S8

SETFREQ B1 0 868.1 868.3
SETFREQ B2 0 868.4 868.6
SETFREQ S1 0 868.1 868.3
SETFREQ S2 0 868.1 868.3
SETFREQ S3 0 868.1 868.3
SETFREQ S4 0 868.1 868.3
SETFREQ S5 0 868.1 868.3
SETFREQ S6 0 868.4 868.6
SETFREQ S7 0 868.4 868.6
SETFREQ S8 0 868.4 868.6

SETTP B1  2 3 4
SETTP B2  40 40 4
SETTP S1  5 7 4
SETTP S2  40.3 40.3  4
SETTP S3  6 8 2
SETTP S4  3 7 4
SETTP S5  2 5 3
SETTP S6  42 42 2
SETTP S7  39 39 4
SETTP S8  43 42 3

START 0 0 50 0.001
EXIT
```

Here two base stations and eight slaves are created. The commands for creating these stations are the same as the no noise configure. Base station 1, Slave 1, Slave 2, Slave 3, Slave 4 and Slave 5 operate on the same frequency band whose centre frequency is 868.2MHz. Base station 2, Slave 6, Slave 7 and Slave 8 adopt the same channel whose centre frequency is 868.5MHz. 868.2MHz and 868.5MHz are assumed to be two adjacent channel frequencies. It means the adjacent channel interference may affect the communication in this scenario.

Slave 2 is close to Base station 2 and far away from Base station 1. The signal sent by Base station 1 was received by Slave 2 with very low power. If there is no adjacent channel interference, the signal will be received without error. Figure 5.10 No_interference lists all the communication process between the base stations and slaves. Because of using different frequencies bands, the two radio groups B1 and B2 did not disturb each other such that B1 communicated with S1 from 0s to 0.38s and B2 also communicated with S6 during the same time.

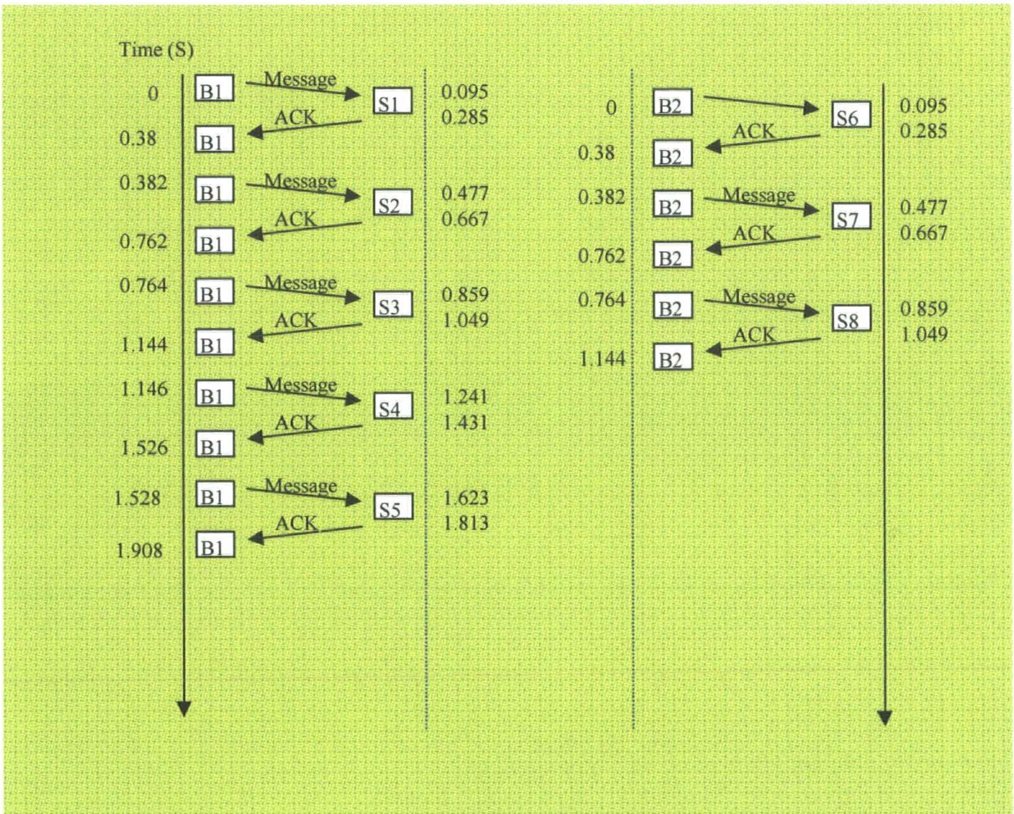


Figure 5.10 No_interference

In fact slave 2 is so close to Base station 2 operating on the adjacent channel that Base station 2 will affect the radio communication between Base station 1and Slave2. In this simulation if the received power of wanted signal (Pw) and the received power of the adjacent channel signal (Pa) achieve the following condition:

20 long10 (Pw/Pa) >= 40;

The wanted signal will be affected by the adjacent channel signal.

Here Slave 2 received the wanted signal from Base station 1 with 0.000001W and got the adjacent channel signal 0.010343W. These two values satisfy the condition. Slave 2 was affected by the adjacent channel interference caused by Base station 2. From 0.382s to 0.477s, the communication between B1 and S2 was corrupted by that inference, because during that time B2 was sending message to S7 and S2 was also picking up the adjacent channel signal which is too strong to affect the signal sent by B1. B1 had to send message to S2 again at 0.795s, but B2 was transmitting message to S8 from 0.764s to 0.859s. The adjacent interference happened again and S2 lost the message sent by B1 again. Finally S2 received the message sent by B1 at 1.182s and sent ACK back at 1.277s. Because other slaves operating on the same frequency as B1 are far away enough from B2, they were not affected by the adjacent interference. Figure 5.11 shows the situation. Compared with the no interference environment, the time to finish the same task was deferred by 0.8s (2.708s-1.908s).

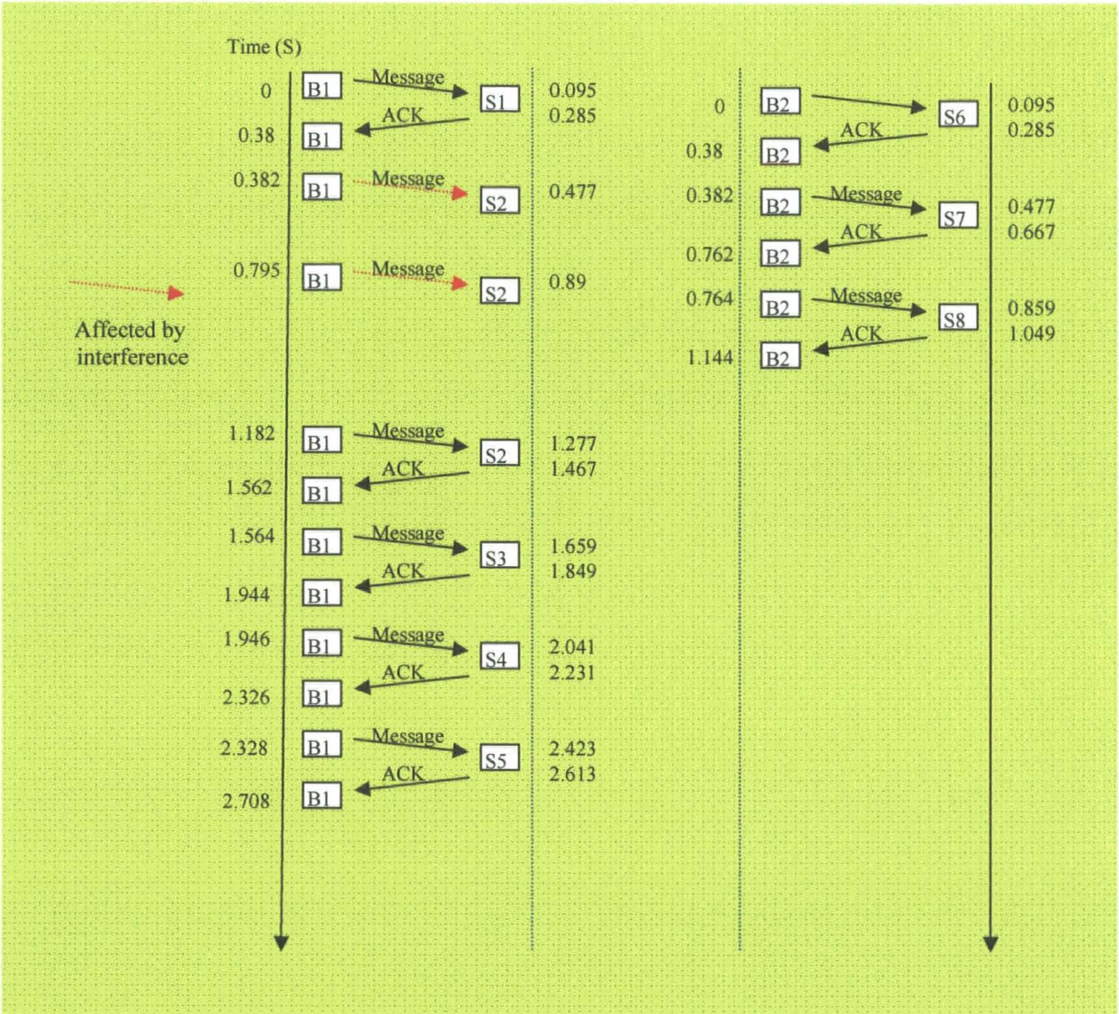


Figure 5.11 Adjacent_interference

5.7 Conclusion

The simulation tool shows the ability of simulating a local sensor system in four situations: no noise, repetitive noise, random noise and adjacent channel interference. In no noise situation, radio devices follow the simple communication protocol to transmit and receive messages. No message is corrupted by another message. Only one device can use the radio channel in a time. Slaves will send an acknowledge frame out, if they receive a message. When base station finishes communication with one slave, it will turn to next one. Base station will stop sending message to slaves when all the slaves finish one message reception and ACK transmission. The slave list is updated in base station every time step. It means a new slave will be added into the list once it turns on and operates on the frequencies the base station using. This function can be considered to be done in hardware and it is replaced by registration function in chapter 6 WLAN.

In repetitive noise, random noise and adjacent channel interference situations, the timeout function, contention window, retransmission function and message error detection of devices are tested. Simulation results are the same as what are expected.

However the local sensor system can be applied in the real system, most radio networks are connected to the wired networks such as internets. People staying in the office can know the status of his factory such as the temperature by receiving data from radio monitors. People can turn on the heating of his/her house by sending command to the radio device connecting to the internet before he/she arrives at home. Of course the radio devices in these applications are more complex than the simple radio devices that only have physical layer, data link layer and application layer. In next chapter 6 WLAN simulation, network layer and transport layer are added to the radio device and a controller which connects to the radio networks by a wire link is added. The protocol used in the data link layer is 802.11b, which is widely used in the world. Internet protocol is used in the network layer and UDP is used in the transport layer.

Chapter 6 Wireless LAN Simulation

6.1 Introduction

This case study is to simulate a typical wireless LAN that adopts the 802.11b standard. The physical distribution is shown in figure 6.1.

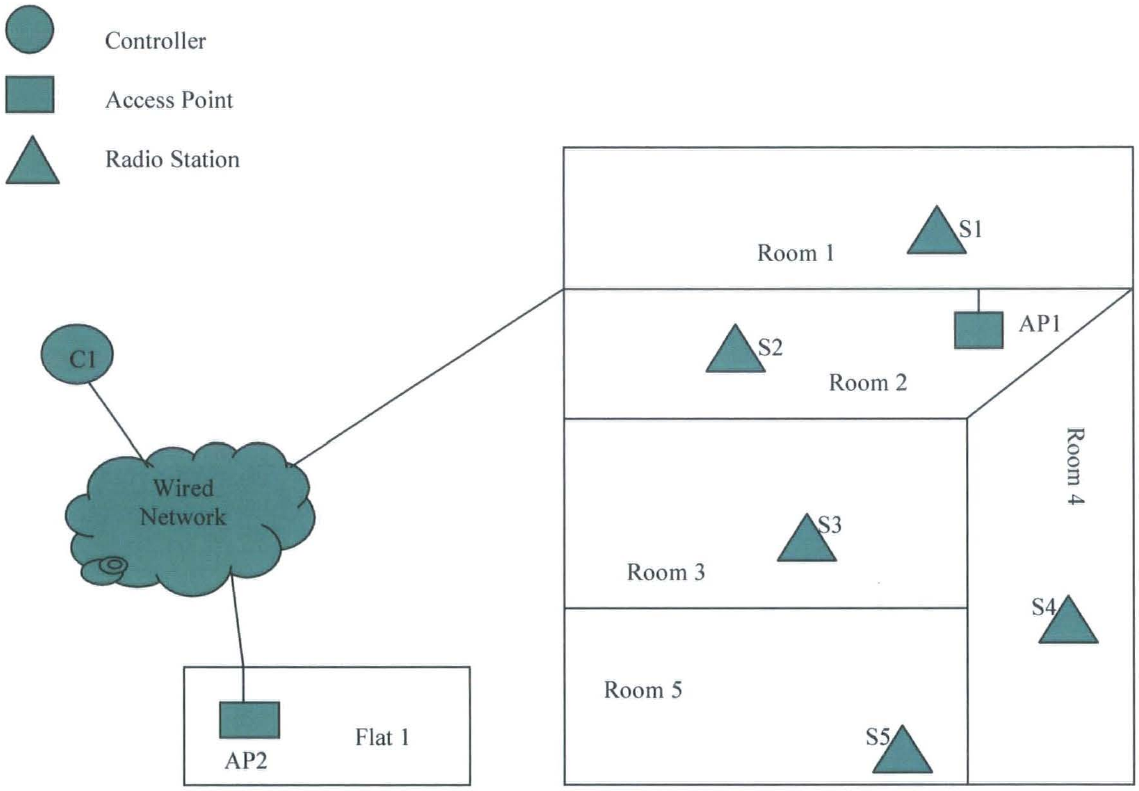


Figure 6.1 Physical Distribution

There are five rooms installing five radio stations and one access points AP1. Access point AP2 is installed in flat 1. Five stations operate on the same frequency as AP1. AP2 operates on another frequency. The reason for setting two access points using different frequency is to show that this simulation tool can simulate the scenario using more than one frequency such as adjacent channel interference. The communication between stations and access point is by radio wave. The access point connects to the controller which may be in the same building or another city by wire. This simulation tool focuses on the radio communication, so the wired communication is assumed to be ideal that means no message will be corrupted or lost. The stations play the role of monitors that may report the temperature and the status of the door or window (close or open) to the controller. The advantages of using radio stations such as low cost of equipment and installation were introduced in the first chapter.

The IEEE 802.11b standard supports the following wireless features: CSMA/CA, back off procedure, ACK procedure, retransmission of unacknowledged frames, RTS/CTS handshake, beacon generation, fragmentation, short or long preamble, distributed coordinated function (DCF) and point coordination function (PCF).

In order to understand those features and simulate this scenario, it is necessary to introduce the 802.11b standard first.

6.2 IEEE 802.11b standard

IEEE 802.11b is a standard that deals with the Physical and MAC Layer as defined by the OSI seven layers model introduced in chapter two.

6.2.1 Physical layer

IEEE 802.11b specifies that the operating frequency of the system is 2.4GHz with data rates of 1, 2, 5.5 and 11 Mbps and the system adopts the Direct Sequence Spread Spectrum (DSSS) technique

6.2.1.1 Spread Spectrum (SS)

The Shannon and Hartley channel-capacity theorem states:

$C=W*Log_2 (1+S/N);$ (1)

The communication channel capacity C can be increased by expanding the bandwidth W and/or increasing the signal-to-noise ratio.

Spread Spectrum refers to the expansion of the signal bandwidth. There are lots of techniques to achieve that condition. The common way is to add the key which can be a sequence to the communication channel. Each data bit is represented by a bit pattern, thereby increasing the bandwidth.

The formal definition of SS is: Spread spectrum in an RF communication system in which the baseband signal bandwidth is intentionally spread over a larger bandwidth by injecting a higher frequency signal [32].

Before the data is sent to the air, the SS technique is applied. Figure 6.2 shows the bandwidth effects of the spreading operation.

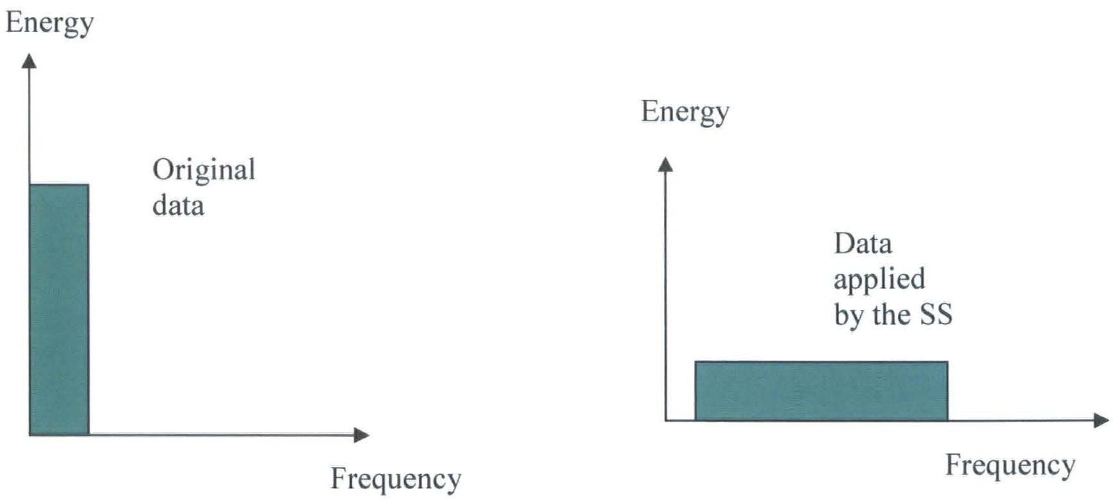


Figure 6.2 Encoding and Transmission

The figure 6.2 also shows that the SS technique uses a wider frequency band.

When the receiver gets the data, it will decode the data (figure 6.3).

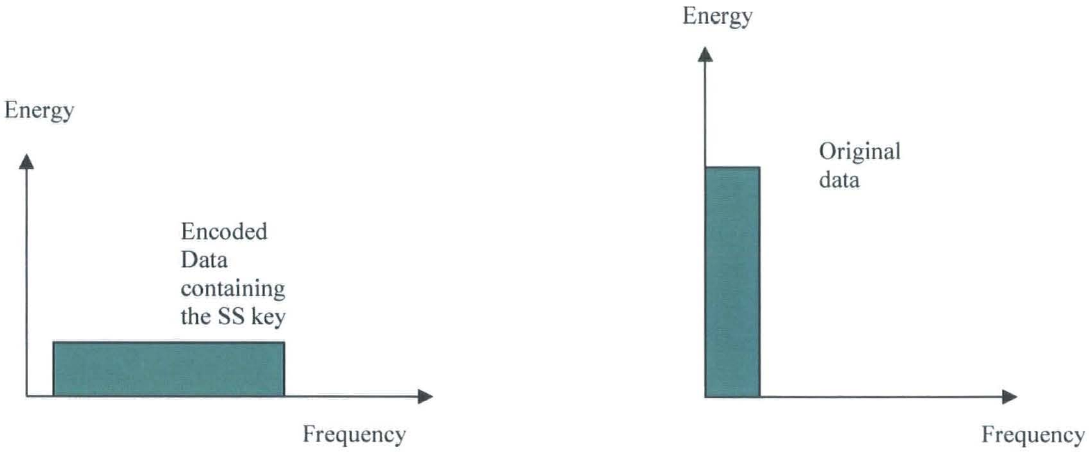


Figure 6.3 Reception and Decoding

a) The key of the SS:

To add the key to the data is the main characteristic of the SS technique. Both of the transmitter and receiver should know the key for spreading data and dispreading the data. In modern communication, the key is known as sequence.

b) The advantages of the SS:

1) Resistance to noise:

The receiver only processes the signal containing the SS key and may correct a message corrupted by noise using the redundancy.

2) Security

If the SS key is nearly a random value, it is hard for other devices to decode it.

6.2.1.2 Direct Sequence Spread Spectrum

The DSSS is one basic type of Spread Spectrum modulation techniques. DSSS does not provide resistance to noise but against partial band interference and frequency selective channel fading.

In DSSS system the key is the chip sequence. In IEEE 802.11b the chip sequence is 11 bits long. Here the bit does not mean the binary data transmitted. It means the individual components of the spreading code as chips. The information bit (bit 1 or bit 0) is combined via an XOR function with this 11-bit sequence. The result used to represent the bit "1" is the inverse of the chip sequence used to represent the bit "0". Redundancy is used by the DSSS, so some messages affected by noise may be correctly received by the receiver.

Usually the bandwidth of a channel using DSSS is 22MHz and the minimum distance between the channels is 30MHz. The total available ISM bandwidth for IEEE802.11b is

83.5MHz in the US and Europe (2.4835-2.4 GHz), 26MHz in Japan (2.497 - 2.471 GHz), so there may be only three channels working at the same time without adjacent channel interference in a certain coverage in the US and Europe. In some product the distance between used channels can be reduced to 5MHz, for example the channel allocation used in ME 102 (A product of Netgear) that can be used as an access point is shown in table 6.4 [33].

Although there are 13 available channels, the guide also recommend the users to use the channel 1, channel 6 and channel 11 for keeping the frequency spread to be 25 MHz. The data rate could be 1,2,5.5, and 11 Mbit/s.

Channel	Center Frequency	Frequency Spread
1	2412Mhz	2399.5Mhz – 2424.5Mhz
2	2417Mhz	2404.5Mhz – 2429.5Mhz
3	2422Mhz	2409.5Mhz –2434.5Mhz
4	2427Mhz	2414.5Mhz –2439.5Mhz
5	2432Mhz	2419.5Mhz – 2444.5Mhz
6	2437Mhz	2424.5Mhz –2449.5Mhz
7	2442Mhz	2429.5Mhz –2454.5Mhz
8	2447Mhz	2434.5Mhz – 2459.5Mhz
9	2452Mhz	2439.5Mhz –2464.5Mhz
10	2457Mhz	2444.5Mhz –2469.5Mhz
11	2462Mhz	2449.5Mhz – 2474.5Mhz
12	2467Mhz	2454.5Mhz – 2479.5Mhz
13	2472Mhz	2459.5Mhz – 2484.5Mhz

Table 6.4 ME 102 Frequency Allocations

6.2.1.3 Relation to simulation

The power level of any interference before correct data reception fails with the system using the DSSS will be greater than the power level of the interference with the system without using the DSSS. For the simulation the related parameters should be adjusted to model this.

In the simulation it is assume that if the difference between two adjacent channels is less than 25MHz and the log (wanted signal power/ unwanted signal power) is less than 35dB, the adjacent interference will be caused. The AP1 and the devices in its coverage operate on 2424.5MHz –2449.5MHz. There are also another two available channels that operate on 2399.5MHz – 2424.5MHz and 2449.5MHz – 2474.5MHz.

6.2.2 Data link layer

Figure 6.5 [34] shows the relationship between 802.11 and other IEEE standards.

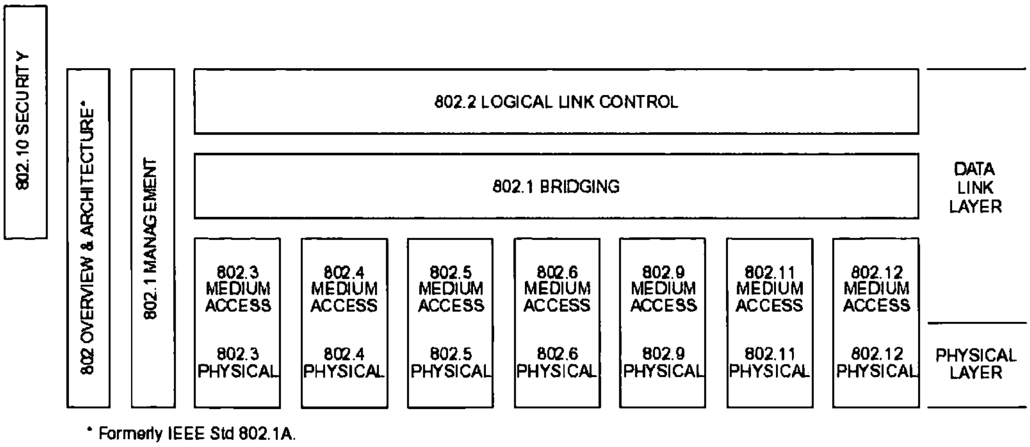


Figure 6.5 IEEE Standards

The section 6.2.1 has introduced 802.11 technology used in physical layer such as DSSS. In this section, the IEEE 802.11b defines the methods, protocols and frame formats in medium access.

6.2.2.1 CSMA/CA

The two popular methods for accessing the channel in networks are CSMA/CD and CSMA/CA.

CSMA/CD (Carrier Sense Multiple Access/ Collision Detection) is suitable for wire networks such as Ethernet. For wireless networks, the CSMA/CA (Carrier Sense Multiple Access/ Collision Avoidance) is used. The main reasons are following:

1. Collision Detection Mechanism is based on full duplex principle, which means the transmission and receiving can be done at the same time, but usually a radio station is a half duplex device that is unable to detect the channel and transmit data simultaneously. To use full duplex radio devices will increase the cost of the wireless LAN significantly.

2. One of the basic assumptions of using collision detection is that all nodes can hear each other. Due to the hidden station and exposure station problems [35], the wireless LAN may not achieve this requirement. It is assumed that there are three stations shown in figure 6.6, station A is communicating with station B and station C is out of station A's coverage. Because station C does know the channel is busy, it may send message to station B, which may disturb the communication between station A and station B. This is called hidden station problem. If station A is communicating with other stations such as station D and station B wants to send message to station C, Station B will detect that

the channel is busy and give up the transmission because of the coverage of station A. This is called exposure station problem.

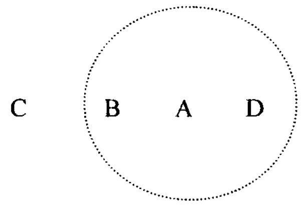


Figure 6.6 Hidden Station Problem

CSMA/CA: if the channel is busy, the transmitter will defer the transmission. If the channel is free for a specified time called DIFS (Distributed Inter Frame Space, it will be introduced in 6.2.2.3), then the transmitter is allowed to transmit.

6.2.2.2 Virtual Carrier Sense

In order to reduce the probability of collision, a mechanism named virtual carrier sense is defined.

A station transmits a short packet named RTS (Request To Send) first, the target station will send a CTS (Clear to Send) back (if the channel is free). Both RTS and CTS packets include the information of the destination and the duration of the following transaction. The RTS packet contains the source address, but the CTS packet does not. Other stations which receive these pieces of information will set their NAV (Network Allocation Vector) by themselves, which indicate the channel is busy during the following duration. This method solves the hidden station and exposure station problems.

For example:

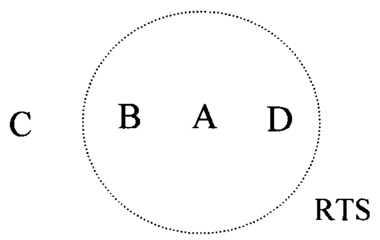


Figure 6.6 RTS Coverage

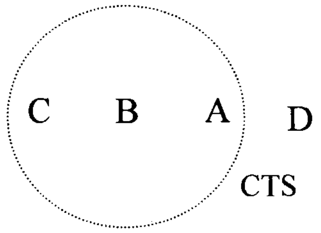


Figure 6.7 CTS Coverage

In Figure 6.6 station C is a hidden station for station A. Though station A can not notice the station C to set itself NAV by sending RTS out, station B will notice station C to set itself NAV by sending CTS out (Figure 6.7).

6.2.2.3 Inter frame spaces

The space between two frames is named inter frame space such as the space between RTS and CTS, the space between Frame and ACK. There are four types of inter frame spaces.

Slot Time: For 802.11b the slot time for the High Rate PHY shall be the sum of the RX-to-TX turnaround time (5 μ s) and the energy detection time (15 μ s). The propagation delay shall be regarded as being included in the energy detection time [36]. The value is 20 μ s in DSSS system. The slot time defined here is different from the time step used in the simulation program. The time step that is 1 μ s is used in the simulator to control the time accuracy of the simulation and slot time is 20 μ s.

SIFS- Short Inter Frame Spaces: it separates the transmission in one sequence, for instance RTS-CTS, Frame-ACK. It is the shortest Inter Frame Space. Its value is 10 μ s in DSSS system.

PIFS-Point Coordination IFS: it is used in Point Coordination system introduced later. The value is SIFS plus a slot time.

DIFS-Distributed IFS: it is used in Distributed Coordination system introduced later. The value is the PIFS plus one slot time.

EIFS-Extend IFS: it is for station to transmit a frame when previous frame was not received correctly. The value is the DIFS plus one slot time.

Figure 6.8 shows the relation of the four types inter frames spaces and gives a simple example of use.

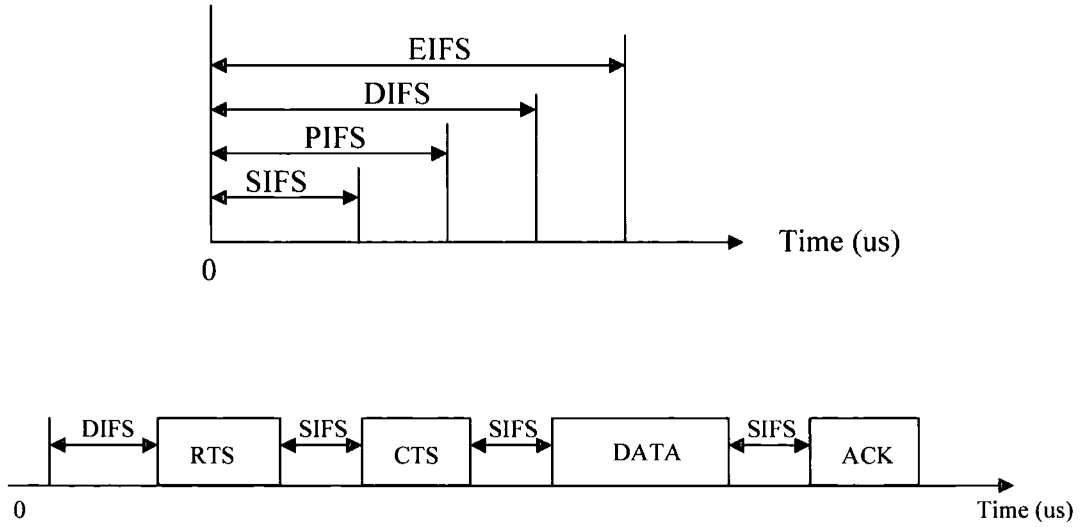


Figure 6.8 Inter Frames

6.2.2.4 Backoff Time

When more than one station wants to access the medium at the same time, a collision may be caused. Backoff time is used to solve this problem. Each station is required to take a backoff value (Figure 6.9) and wait for this number of time steps before accessing the medium.802.11b uses this method to minimize the probability of collision.

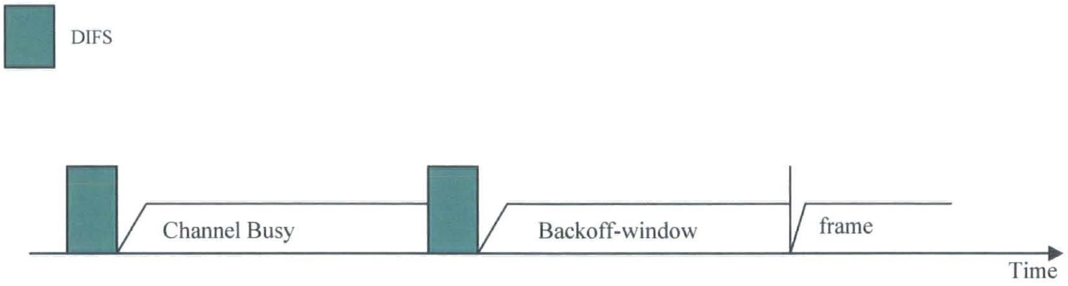


Figure 6.9 Random backoff

If one station wants to transmit, it should detect the channel for a DIFS time. If the channel is free for this DIFS time, then the station should generate a random backoff time for an additional delay before transmitting instead of transmitting immediately. At the time for transmission, the station should also ensure the channel is idle by detection. This method minimizes the probability of multiple stations transmitting at the same time.

If there is more than one station to access the channel at the same time such as sending the RTS, a collision will be caused. The station will conclude that the transmission has failed by the timeout expiration. Then they will use the random back off time method to access the channel for retransmitting.

Backoff time= Random() *aSlotTime [37]; (1)

Random() will produce a integer which is between 0 and CW (Contention Window). Here CW is a variable and CWmin <= CW <= aCWmax.

For DSSS system, the CWmin is 31, CWmax is 1023 and a slot time is 20 μ s.

Retransmission limit: Every station shall set a limit for retransmission. If the number of retransmissions reaches the limit, the station will stop trying to retransmit that message.

6.2.2.5 802.11b Frames Format

802.11b defines the format of the header of the Physical layer and the Data Link Layer frames. The format of the packet to be sent is as following:

PLCP	MPDU

PLCP: Physical Layer Convergence Protocol

MPDU: MAC Protocol Data Unit

6.2.2.5.1 PLCP

1) Long PLCP

Preamble enable synchronization format:

Preamble					
Sync 128bits	SFD 16 bits	Signal 8 bits	Service 8 bits	Length 16 bits	CRC 16 bits

Sync: A 128-bit sequence of alternating zeros and ones, which is used to synchronize the received packet timing.

SFD- Start Frame Delimiter: It consists of the 16-bit binary pattern 0000 1100 1011 1101.

Signal: it indicates the bit rate of the MPDU such as 1Mbps, 2Mbps, 5.5Mbps, 11Mbps.

Length: it is the time in microseconds to transmit the MPDU.

The bit rate for transmitting the PLCP is always at 1 Mbps, so the time for transmitting is 0.192 ms.

2) Short PLCP (Optional):

Preamble					
Short Sync 56 bits	Short SFD 16 bits	Signal 8 bits	Service 8 bits	Length 16 bits	CRC 16 bits

Only when both of receiver and transmitter are capable of using the short PLCP, the short PLCP can be used.

6.2.2.5.2 The format of the general MPDU

Octets:	2	2	6	6	6	2	6	0-2312	4
	Frame Control	Duration/ID	Address 1	Address 2	Address 3	Sequence Control	Address 4	Frame body	CRC 16 bits

1) Frame Control Field:

Bits	2	2	4	1	1	1	1	1	1	1	1
	Protocol Version	Type	Subtype	ToDS	From DS	More flag	Retry	Pwr Mgt	More data	WEP	Order

The definitions details of the parameters in the frames are in [38].

6.2.2.5.3 The list of the individual frames format

1) Control Frames Format:

RTS Frame Format:

Octets	2	2	6	6	4
	Frame control	Duration	RA	TA	CRC
	MAC Head				

CTS Frame Format:

Octets	2	2	6	4
	Frame control	Duration	RA	CRC

MAC Head

ACK Frame Format:

Octets	2	2	6	4
	Frame control	Duration	RA	CRC

MAC Head

2) Management Frames:

Octets:	2	2	6	6	6	2	0-2312	4
	Frame Control	Duration	DA	SA	BSSID	Sequence Control	Frame body	CRC 16 bits

The frame body of the management frame can be beacon frame, disassociation frame, association request frame, association response frame, reassociation request frame, reassociation response frame, probe request frame, probe response frame, authentication frame and deauthentication.

Beacon frame format:

Time Stamp	Beacon interval	Capability information	SSID	Supported rates	FH parameter set	DS parameter set	CF parameter set	IBSS parameter set	TIM

Disassociation frame format:

Reason code

Association Request frame format:

Capability information	Listen interval	SSID	Support rates

Association Response frame format:

Capability information	Status code	Associatio ID (AID)	Support rates

Authentication frame format:

Authentication algorithm number	Authentication transaction sequence number	Status code	Challenge text

Deauthentication frame format:

Reason code

RA is the address of receiver and TA is the address of transmitter. More details about above parameters can be found in [38].

6.2.2.5.4 Relation with the simulation

a) The simulation frame format

The simulation tool is to simulate the very low level hardware operation. It is not necessary to simulate all the parameters in the frames in our simulation. The common frame format is as following:

```
struct message
{
char  sourcename[20];
char  destination[20];
char SIP[40];
char DIP[40];
int  S_port;
int  D_port;
char  type_name[20];
int   flag;
char   stream;
double  length;
int CM;
int status;
double start_time;
double end_time;
double duration;
};
```

Here the sourcename means the transmitter physical name.

The destination is the physical name of target receiver. For the beacon frame, the sourcename is set to ALL.

SIP[40] is the IP address of transmitter.

DIP[40] is the IP address of destination.

S_port is the port number used by the transmitter.

D_port is the port number used by the target receiver.

Type_name is type of the frame. According to the type_name, the packet is passed to the related layer function.

Flag: If it is set to 0, it means there is no error in message. If it is set to 1, it means there are errors that may be caused by noise, collision or interference in message. This parameter simulates the result of the CRC.

Stream: It is the code that transmitter wants to send to receiver. For example if 1101 means to open a door, the receive gets the message and will do open a door.

Length: The value of the length which is used to calculate the transmission time is the actual value of the real frame. For instance the ACK frame is 14 bytes, so the Length is 14.

CM: If the message is a wired message, it is set to 1. If the message is a wireless message, it is set to 0.

Status: 0 is for the initialization, 1 means it has been received and 2 means it has been transferred by access point.

Start time: It is the time when the message is to be sent.

End time: It is the time when the transmission ends.

Duration: It is used in some frames such as the duration time in RTS and CTS. According to it, the transmissions of other devices will be deferred for this time.

b) Time calculation and definition in IEEE 802.11b

$$\text{Time} = \text{PreambleLength} + \text{PLCPHeaderTime} + \text{LENGTH} \times 8 / \text{DATARATE}; \quad (2)$$

Here for long preamble, the value of PreambleLength is 144us and the value of the PLCPHeaderTime is 48us; for short preamble, the value is 72us and 24us. Short preamble is optional and in this work the long preamble is used. Data rate can be 1Mbps, 2Mbps, 5.5Mbps and 11Mbps.

6.2.3 DCF and PCF

There are two access methods of the IEEE 802.11b MAC. They are DCF (Distribution Coordination Function) and PCF (Point Coordination Function). Both of them have an AP in the coverage. An AP is a radio station that connects to the wired network. The AP can send the packets to the station directly and receive data directly from a radio station and wire networks. The main differences between the two are that the PCF uses a Point Coordination (PC) which controls all the stations' access to the channel, but for the DCF all the devices including the AP use the CSMA/CA method to access the channel. For example, if a system adopts the PCF, the PC will tell the device1 when it can transmit and when to stop the transmission and then it will be the turn for device 3 to use the channel. For DCF, if device 2 is communicating with the AP, other devices who want to transmit detect that the channel is busy and then they will defer their transmission until the channel is free for a DSFS.

DCF and PCF only define the access method for the radio devices. When one station communicates with another station, they may send a message via the AP or may not. It may depend on the devices type and router protocols. For instance in the wireless sensor networks, the radio devices can communicate with each other directly.

Many popular network products such as the ME 102 support the DCF. Here the simulation tool is also based on DCF and radio stations transmit messages via an access point.

6.3 Wireless LAN Simulation

6.3.1 Registration

a) Background

When a station changes from power off mode to power on, from sleep mode to active mode or just enters the cell, it should register to be a member of that cell. For registration the station needs to get synchronization information from the Access Point.

There are two methods to get the synchronization information:

Passive scanning: the AP sends the beacon frame with synchronization information periodically and the station just waits to receive the beacon frame.

Active scanning: the station sends probe request frames and waits for a probe response from the Access point.

The simulation tool uses the passive scanning. It means that the access point generates the beacon frames periodically for the station registration.

After receiving the beacon frame, the station will try to do the following process to finish registration.

1) The Authentication Process:

This is the interchange of information between the AP and the station such as providing a password.

2) The Association Process:

When the station is authenticated, then it will send its properties information to the AP such as current position. Only after this process is finished, is a station is allowed to transmit and receive data frames.

b) Simulation

To configure the simulator, access points and stations are created and their properties are set by using a series of commands such as SETTP, SETFREQUENCY, SETIP.

The configure file is as follows:

```
CREATE CONTROLLER  C1
CREATE BASE_STATION  AP1
CREATE BASE_STATION  AP2
CREATE SLAVE S1
CREATE SLAVE S2
CREATE SLAVE S3
CREATE SLAVE S4
CREATE SLAVE S5

SETIP C1  193.168.168.1
SETIP AP1 193.168.168.10
SETIP S1  193.168.168.11
SETIP S2  193.168.168.12
SETIP S3  193.168.168.13
SETIP S4  193.168.168.14
SETIP S5  193.168.168.15
SETIP AP2 193.168.168.2

SETFREQ AP1 0 2424.5 2449.5
SETFREQ S1 0 2424.5 2449.5
SETFREQ S2 0 2424.5 2449.5
SETFREQ S3 0 2424.5 2449.5
SETFREQ S4 0 2424.5 2449.5
SETFREQ S5 0 2424.5 2449.5
SETFREQ AP2 0 2449.5 2474.5

SETTP AP1  2 3 4 0
SETTP S1  5 7 4  0
SETTP S2  7 9 3  0
SETTP S3  6 8 2  0
SETTP S4  3 7 4  0
SETTP S5  2 5 3  0
SETTP AP2  60 62 4 0

CONNECT AP2 0 C1  1
CONNECT AP1 0 C1  0

START 0 15 0 0.000001
EXIT
```

CREATE BASE_STATION AP1 -------> A base station named AP1 is created.

SETIP AP1 193.168.168.10 ------->The IP address of AP1 is set to 193.168.168.10.

SETFREQ AP1 0 2424.5 2449.5 ------->The high and low frequencies for channel [0] of the AP1 are set to are 2424.5 & 2449.5. Every device has three channels for using in the model, but only one is used in this study.

SETTP AP1 2 3 4 0-------> The position of AP1 is (2 m, 3 m, 4m). The time for AP1 starting is at 0 s.

CONNECT AP2 0 C1 1 -------> Communication between access point and controller is by wire. This command means the port [0] of AP1 connects to the port [1] of C1.

START 0 15 0 0.000001 -------> (0 15 0) represents (hour/ minute/second). The sum of the three values is the total simulation time. Here the simulation time is 15 minutes. The last parameter 0.000001s (1 μ s) is the time step.

The options for all the devices are shown in table 6.10.

Name	IP address	Frequencies	Time & Position
AP1	193.168.168.10	0 2424.5 2449.5	2 3 4 0
S1	193.168.168.11	0 2424.5 2449.5	5 7 4 0
S2	193.168.168.12	0 2424.5 2449.5	7 9 3 0
S3	193.168.168.13	0 2424.5 2449.5	6 8 2 0
S4	193.168.168.14	0 2424.5 2449.5	3 7 4 0
S5	193.168.168.15	0 2424.5 2449.5	2 5 3 0

Table 6.10 Devices Properties

In order to be close to a real system, the beacon period is set to 30 seconds and the period of sending data by station is set to 3 minutes after it receives the first data reply.

After the initialization for each device, the simulation can be run and the results compared to what is expected of a real system.

As mentioned before the access point will send the beacon frame periodically. Here the period is set to 30s. The first time to broadcast the beacon frame was at 50 μ s. The stations would pick up the beacon frame and start their registration. They got the AP1's name from the beacon frame and generated the RTS frames. The time for sending RTS was the sum of one SIFS, one DIFS and backoff time. One SIFS is for device to process the beacon frame and generate the RTS. According to the 802.11 standard, the channel should be idle for one DIFS before starting a new transmission. The backoff time is a random value for minimizing the probability of collision. The stations access the channel by CSMA/CA. If AP1 receive a RTS, it will send a CTS back to the transmitter. All stations will receive the CTS, but only the target station starts to send its authentication request (Auth-request) frame and other stations will defer their transmissions for a duration time contained by the CTS. AP1 receives the authentication request and sends a response (Auth-respond) back. The target station gets the response frame and sends an association request back. Then the AP1 sends the association response (Ass-res) out after receiving the association request (Ass-req). When the target station receives the association response, it means one registration is successfully finished. The access point will keep the information of the registered station such as name and IP address. After the first station finishes its registration other stations start to register the same way as the first station. When a new device finishes its registration, the member list or map of the devices in the access point will be updated. Figure 6.11 (a,b) shows the process between AP1 and stations. In figure 6.11a all stations receive the beacon at 244 μ s. S2 became the first station to register at 346 μ s and finished it at 1661 μ s. The second station to register is S5 at 2685 μ s and it finished the process at 4000 μ s. In figure 6.11b other stations such as S4, S1 and S3 followed the same steps to finish their registrations.

After sending the first beacon frame at 50 μ s, access point would send beacon frame every 30 seconds. The second one and third one is sent at 30.000100 s and 60.000150 s. The difference between them is 30 seconds which is the sum of one period and 50 μ s (one DIFS) used in CSMA/CA method for accessing the channel.

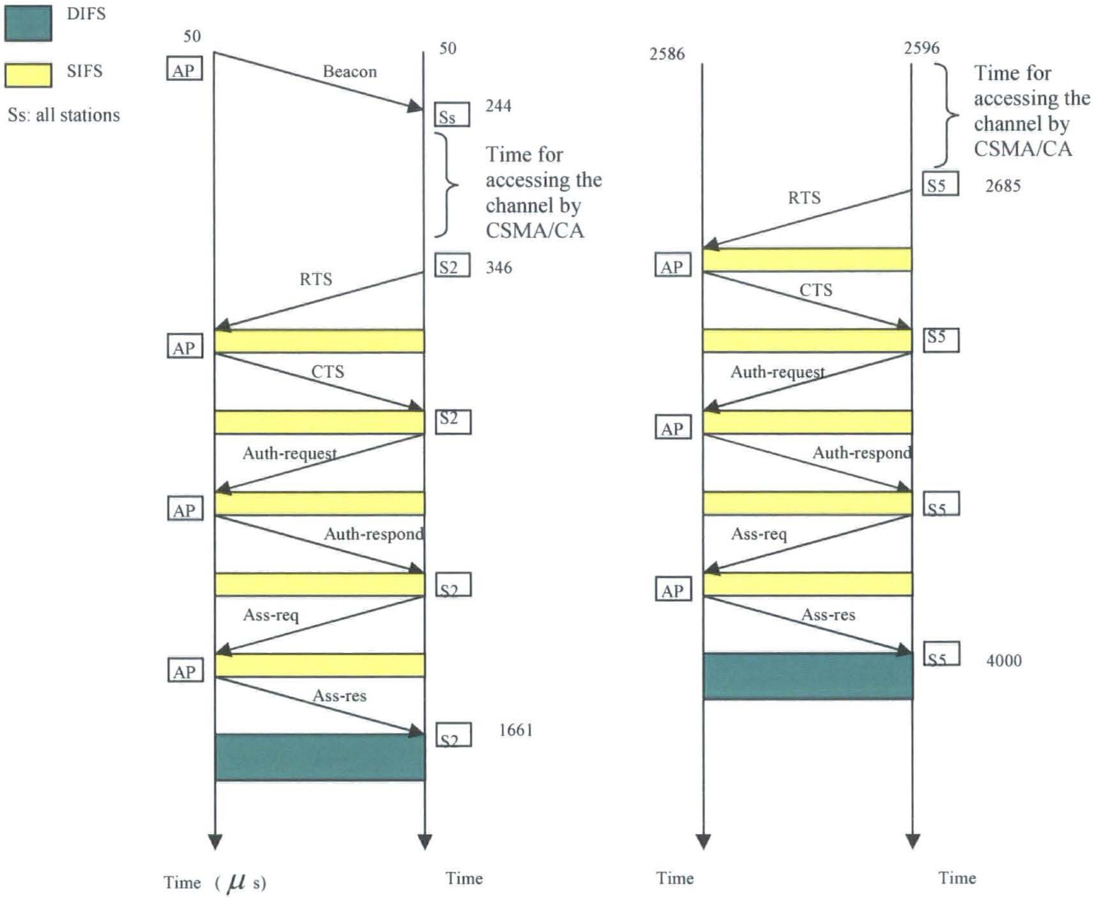


Figure 6.11 (a) Register Simulation

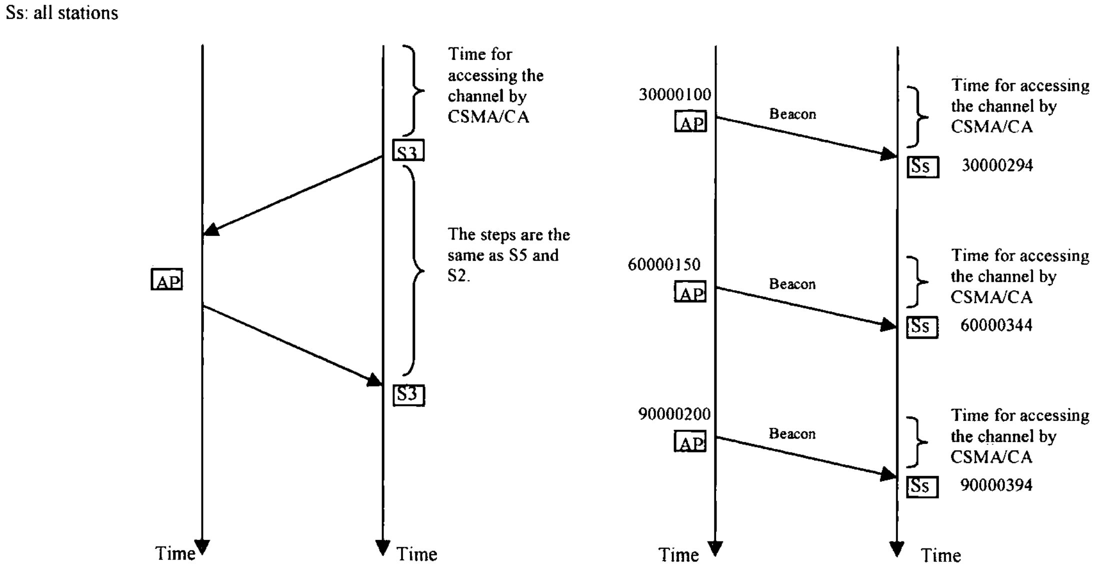
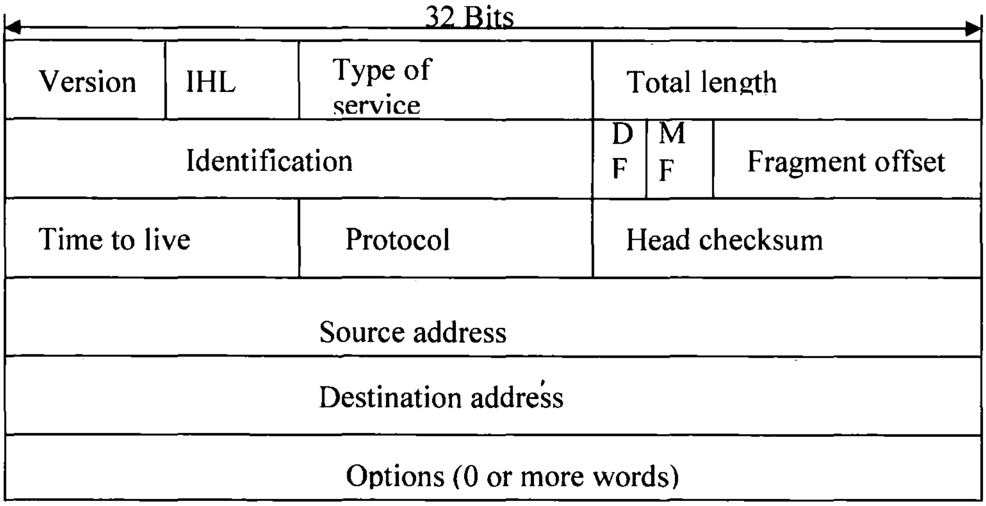

Ss: all stations
Time for accessing the channel by CSMA/CA
S3
The steps are the same as S5 and S2.
AP
S3
Time
Time
30000100
AP
Beacon
Time for accessing the channel by CSMA/CA
Ss
30000294
60000150
AP
Beacon
Time for accessing the channel by CSMA/CA
Ss
60000344
90000200
AP
Beacon
Time for accessing the channel by CSMA/CA
Ss
90000394
Time
Time

Figure 6.11 (b) Register Simulation

6.3.2 Network layer

IP (Internet Protocol) is used in network layer. Data is routed in Internet by this addressing method. There are two versions of IP (IP version 4 and IP version 6). Now IPv4 is widely used in public and private networks. This thesis also uses IPv4. The IPv4 header is showed in Figure 6.12.



Version	IHL	Type of service	Total length		
Identification			D F	M F	Fragment offset
Time to live		Protocol	Head checksum		
Source address					
Destination address					
Options (0 or more words)					

Figure 6.12 IPv4 header

The IP header has a fixed part whose length is 20 bytes and a variable length optional part. The details of the parameters in IPv4 header can be found in [39].

IP address has two main features:

1) The IP address is unique in the internet.

2) The IP address is 32-bit long.

There are five classes in IP address shown in figure 6.13.

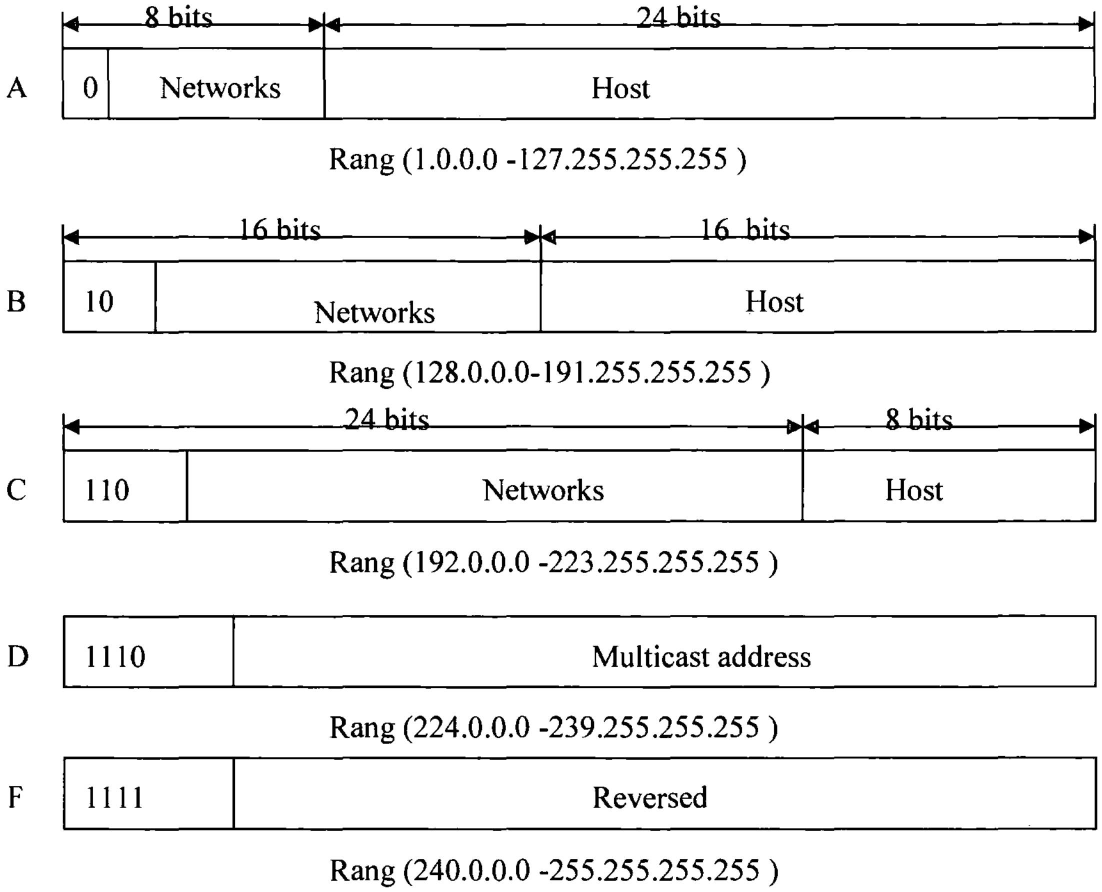

8 bits
24 bits
A
0
Networks
Host
Rang (1.0.0.0 -127.255.255.255)
16 bits
16 bits
B
10
Networks
Host
Rang (128.0.0.0-191.255.255.255)
24 bits
8 bits
C
110
Networks
Host
Rang (192.0.0.0 -223.255.255.255)
D
1110
Multicast address
Rang (224.0.0.0 -239.255.255.255)
F
1111
Reversed
Rang (240.0.0.0 -255.255.255.255)

Figure 6.13 IP address formats

More details about IP address are in [40] and [41].

In this simulation tool IP addresses for stations, access point and controllers are set before the simulation starts. The access point plays the role of a gateway. When one station finishes the registration in one access point, it should send port number request mentioned later to the controller for data transmission. The station only knows the IP address of the controller, so it will send this information that includes the IP address of the controller to the access point first. The access point receives it and sends it to the connected controller according to the IP address of the controller.

6.3.3 Transport Layer

The protocol used in this layer is UDP (User Data Protocol) [42] [43]. Compared with TCP (Transport Control Protocol) [44] [45], it is a simple transport protocol.

The principle is as following: if one station wants to send data to the controller, it should first get port number to use from the controller. The function of the port is discussed in chapter 2. Every station uses the same port number for port requests to the controller such as 1880 in this simulation. Controller will return another port number such as 1010 included in the port reply for the station to send its data.

The ideal (no noise) station port requesting with the controller is shown in figure 6.14.

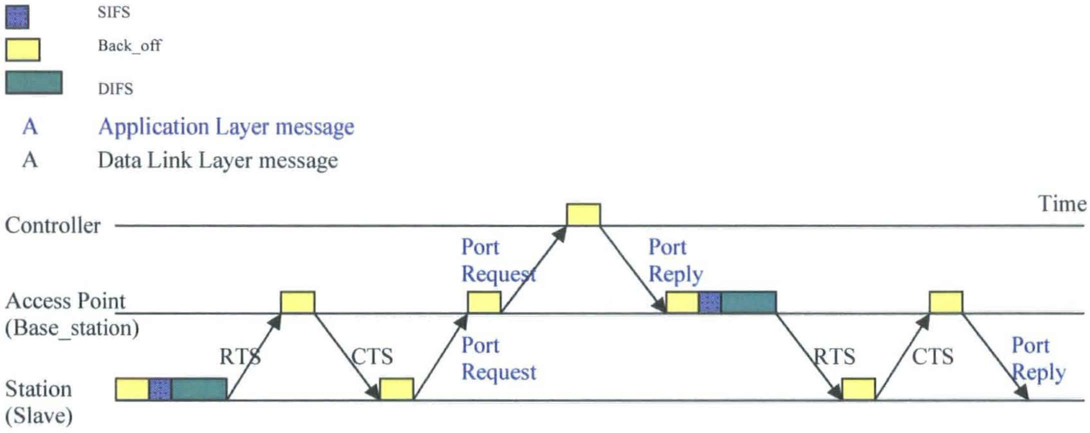


Figure 6.14 Port Request

After the registration with the access point, station S1 will send its port request that is created in Application Layer to the controller with port number 1880. First it sends RTS to the access point and access point receives the RTS and sends CTS back. Then station gets the CTS and sends the port request to the controller via the access point. Access point picks up this message and forwards it to the controller. Controller receives the port request and sends port reply back. The access point gets the port reply from the controller then sends the RTS to the target station. The target station gets the RTS and sends CTS out. The access point receives it and sends port reply to the station.

6.3.4 Application Layer

After getting the port number, Stations start to send data messages to the controller periodically on that port. The period of data in this simulation is 3 minutes. Data messages can contain various types of information. For instance if the station is a temperature monitor of a room, the data message may be the current temperature of the room and the data reply message sent by controller may be a command to increase or reduce the temperature.

The ideal data transmission and reply are shown in figure 6.15.

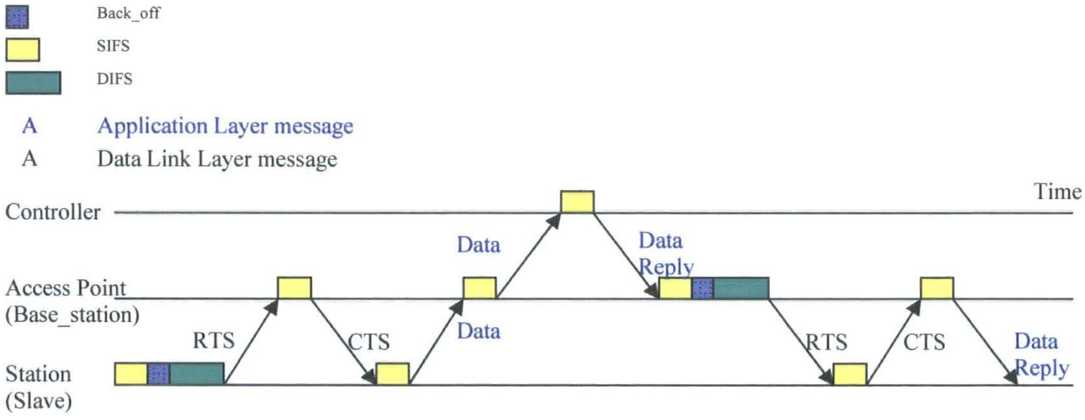


Figure 6.15 Data Transmission

6.3.5 Timeout

Radio communication can be affected by many factors such as noise. If one message is corrupted by noise, the receiver may drop it. Triggered by the timeout function, the transmitter will retransmit the message. For data link layer, it has one timeout to be sure the receiver has got the message correctly. For application layer when it creates a data message, it also sets a timeout which is defined to be h_timeout in this simulation. When the data message is sent, h_timeout starts to work. If the station does not receive any response from the controller before the h_timeout become zero, the station will send the data message again. The value of the timeout of data link layer is the sum of the time spent in sending message and receiving message and the response time such as SIFS in DCF. It is not easy to calculate the value of the h_timeout which includes the time spent in competing for the channel besides the communication time and response time, because time spent in completing for channel by CSMA/CA is an uncertain value for 802.11b. For example, when station 1 finishes transmission of a data message, after some time steps, it may be the turn for the access point to send the data reply from the controller to the station1 and it may be the turn for another station to send information to the access point. From observing one message transmission and reception time which is only few hundred microseconds in total, h_timeout is set to 6s which is a thousand times one data transmission and reception time and includes the delay time that may be a few seconds in a real network.

6.4 No noise situation

6.4.1 Introduction

The simulation scenario has been introduced in the beginning of this chapter. The registration described above is only one part of the simulation. Now the whole communication process including registration, port request and data transmission will be simulated.

6.4.2 No noise situation 1

It is assumed that there is no noise affecting the radio channel and no message lost in transmission. The station should register in the related access point after it receives the beacon frame. Second, it should send its port request to the controller via the access point and the controller then replies with a port number which is used for future communication to the station. The last simulation step is to send a data message to the controller periodically.

In order to be close to the real system, the beacon period is set to 30 seconds and the period of sending data by station is set to 3 minutes after the station receives the first data reply. The data rate is 11 Mbit/s. The simulation time is set to 15 minutes.

This simulation tool spent nearly thirteen minutes in simulating the above scenario on a 2.4GHz Pentium4 PC. A part of the whole communication process is drawn in figure 6.16.

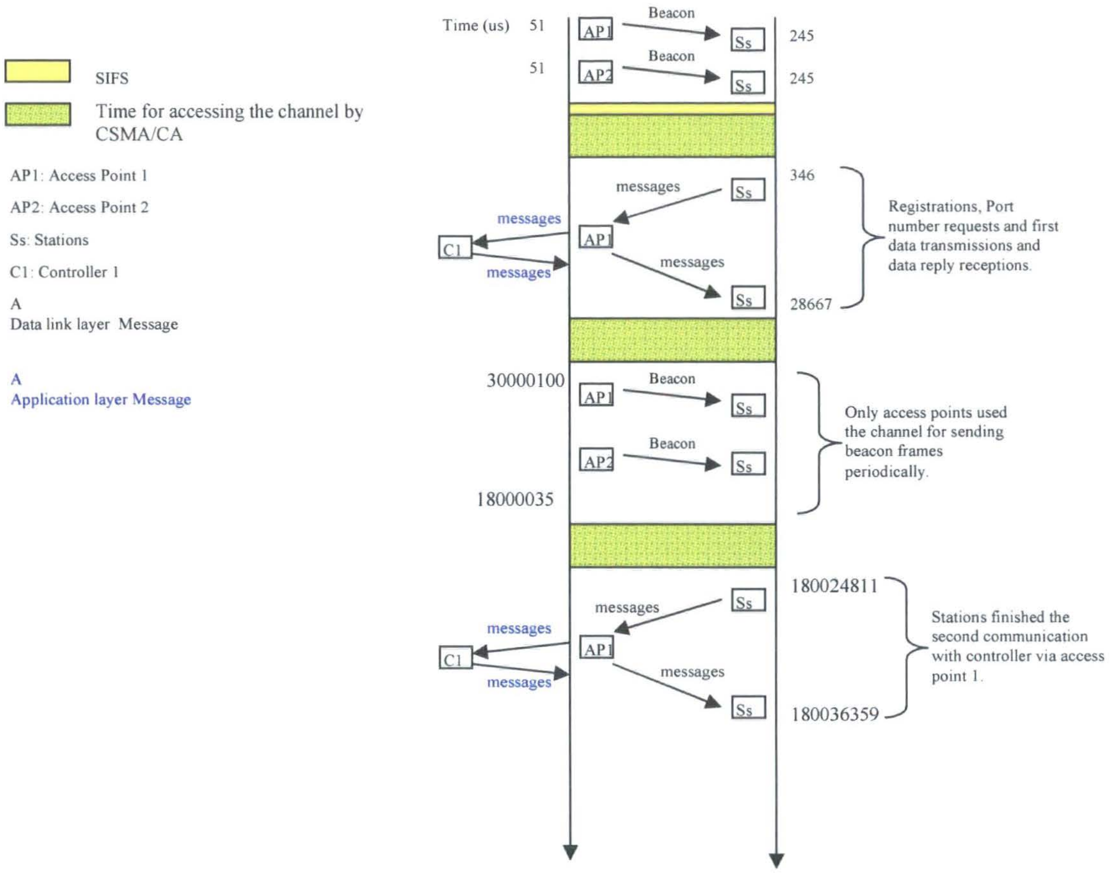

SIFS
Time for accessing the channel by CSMA/CA
AP1: Access Point 1
AP2: Access Point 2
Ss: Stations
C1: Controller 1
A
Data link layer Message
A
Application layer Message
Time (us) 51
51
AP1
AP2
Beacon
Beacon
Ss
Ss
245
245
346
messages
messages
messages
messages
C1
AP1
28667
Registrations, Port number requests and first data transmissions and data reply receptions.
30000100
Beacon
Beacon
18000035
Only access points used the channel for sending beacon frames periodically.
180024811
messages
messages
messages
messages
180036359
Stations finished the second communication with controller via access point 1.

Figure 6.16 A Part of Communication Process

Station 3 was the last one that finished the registration at 16563 μ s. All stations finished the first data transmission and data reply reception before 28668 μ s. From 28668 μ s to 3 minutes and 24810 μ s, there is no message exchanging except beacon frames sent by the access point every 30 seconds, because the data period of stations is 3 minutes. From 3 minutes and 24811 μ s to 3 minutes and 36359 μ s, all stations finished their second communication with the controller via the access point. From then on, stations did not use the channel until 6 minutes and 32056 μ s.

Each device successfully finished five data transmissions and data reply receptions in 15 minutes. The simulation tool shows its ability to simulate a real scenario of fifteen minutes duration, but the simulator has to wait for approximate thirteen minutes to get the results. In fifteen minutes, all stations only used the radio channel for less than six seconds in total. It indicates that 99% of the fifteen minutes can be considered as idle time. In order to get the result quickly and also be close to a real scenario at the same time, the idle time is reduced by reducing the period of stations sending data. The period is reduced to six seconds and h_timeout for data is set to 0.6 seconds which is hundreds times of one data transmission and reception time. The period of the beacon frame is reduced to one second. The short period is good to be used in a short simulation time. The total simulation time is reduced to thirty seconds that is also five times of the data period.

The simulation tool only spent 25 seconds in simulating 30 seconds. Because all the stations do the same steps, only one station is selected to be analysed. According to the result produced by the simulation, figures 6.17 (a,b,c and d) are drawn to show the process of station 3.

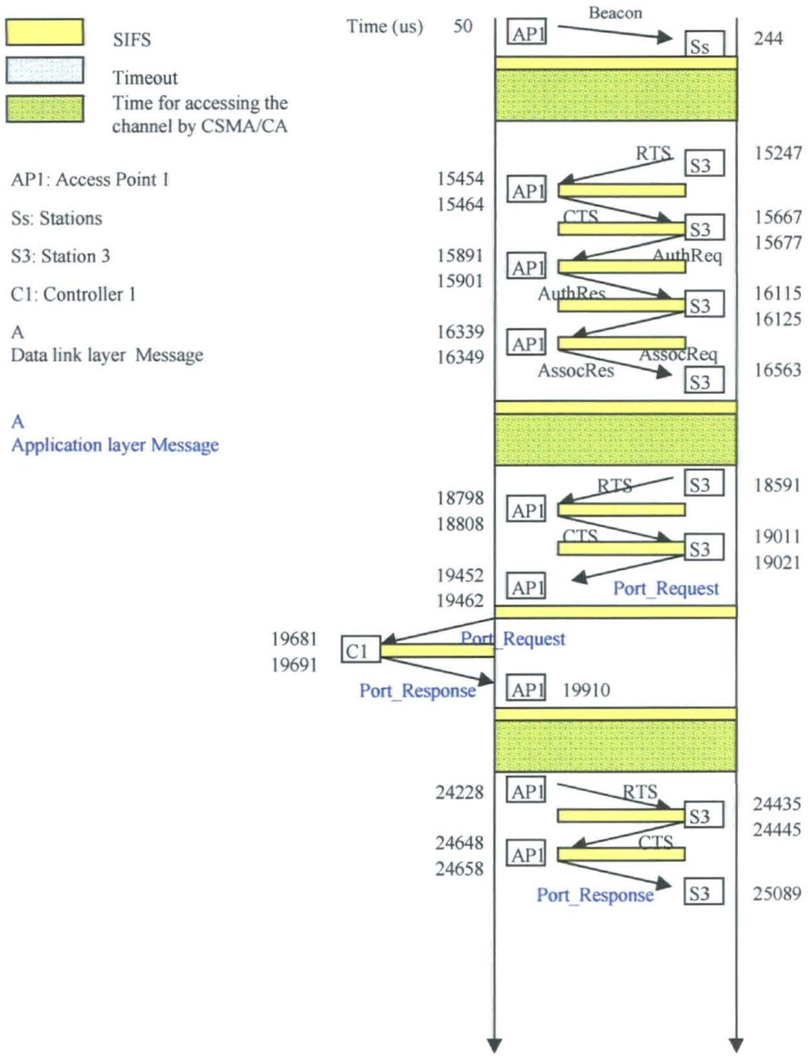


Figure 6.17a Process of Station 3

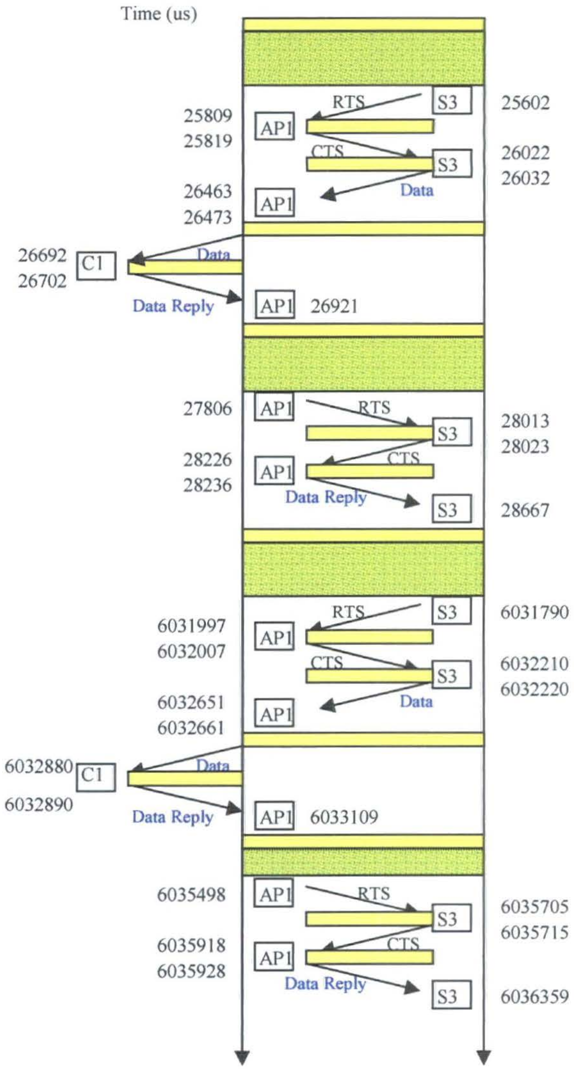


Figure 6.17 b Process of Station 3

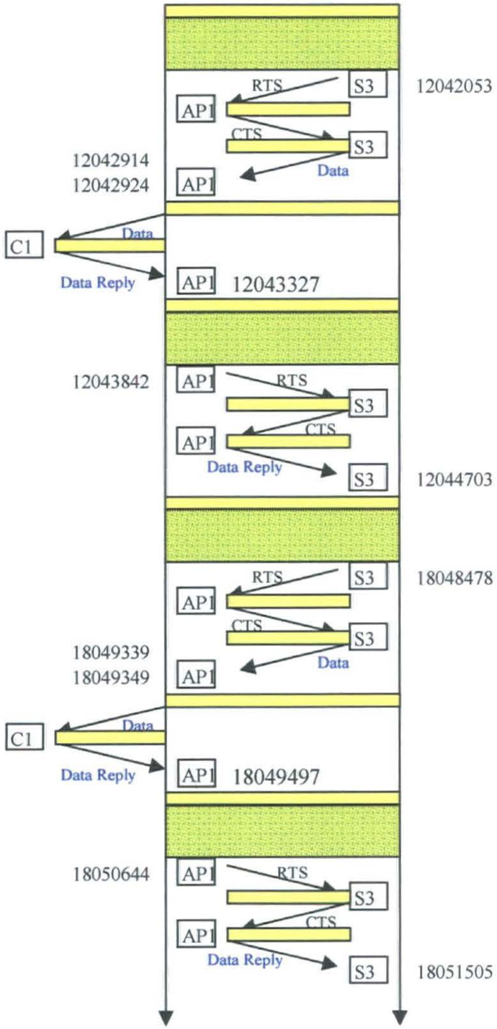


Figure 6.17 c Process of Station 3

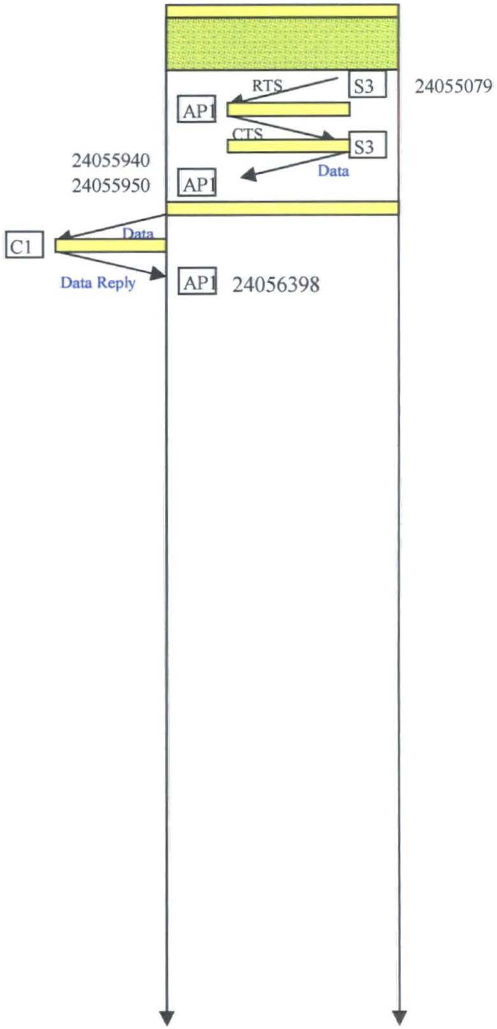


Figure 6.17 d Process of Station 3

After receiving a beacon frame at 244 μs, Station 3 got the chance by CSMA/CA to send RTS to the access point at 15247 μs. The access point received the RTS at 15454 μs and spent 10 μs which is one SIFS defined in 802.11b, in producing CTS. At 15464 μs the access point sent CTS out. Station 3 picked up the CTS and started to do its registration in this access point. Other stations deferred their transmissions according to the duration time included in the received CTS, so during the station 3's registration time there was no other station to disturb the communication. When station 3 finished its registration at 16563 μs, the radio channel was released again. The next time for station 3 to send RTS should be 60 ms more than its last finish time, because the station needs 10ms for creating the message, at least one random back off time for reducing the probability of collision and detecting whether the channel is free for 50 ms (one DIFS). At 18591 μs station 3 sent RTS out again. Here 18591 μs -15563 μs > 60 μs achieved the sending conditions. The access point sent CTS at 18808 μs for allotting the channel for station 3. The access point supported the registration service for stations and transfer function for stations and controller. For data or port messages, it plays the role of a gateway. When the access point receives a data or port message, it will forward them to the controller or target stations. When the access point received the port request from station 3 at 19452 μs, it transfers the port request message to the controller which may be a few thousand kilometres away at 19462 μs. The communication between access point and controller is assumed to be ideal and the delay time in transmission is not considered as well. The controller got this message at 19681 μs and was assumed to give response in one SIFS. At 19691 μs controller sent the port response back to the access point. The access point received the port response from the controller at

19910 μ s and got the chance to send RTS to station 3 at 24228 μ s. The difference between 24228 μ s and 19910 μ s was more than the sum of one SIFS and one DIFS. It seems that the simulation follows the CSMA/CA access method. At 25089 μ s Station 3 gets the port response transferred by the access point. From then on station 3 can send data to controller. The first data was sent at 26032 μ s and station 3 got data reply at 28667 μ s. After that station 3 started to send data to the controller periodically. The period was set to 6 seconds, so the next data was sent at 6031790 μ s and the reply returned via the access point at 6036359 μ s.

In this case the simulation time was 30s and station 3 received the last data reply at 24056398 μ s. Station 3 finished five data transmissions and receptions in total. Other stations did the same steps as station 3.

Simulation time was extended to 60s and other parameters such as period of beacon and data were kept to be the same to see whether the simulation tool will work satisfactory.

In 30s each station finished five data transmissions and receptions; in 60s the result is ten. These simulation results showed that stations can get nearly equal probability to access the channel by CSMA/CA access method. One part of communication of station 1, station 4 and station 5 are selected as an example for channel access competition based on CSMA/CA between stations in figure 6.18 CSMA/CA.

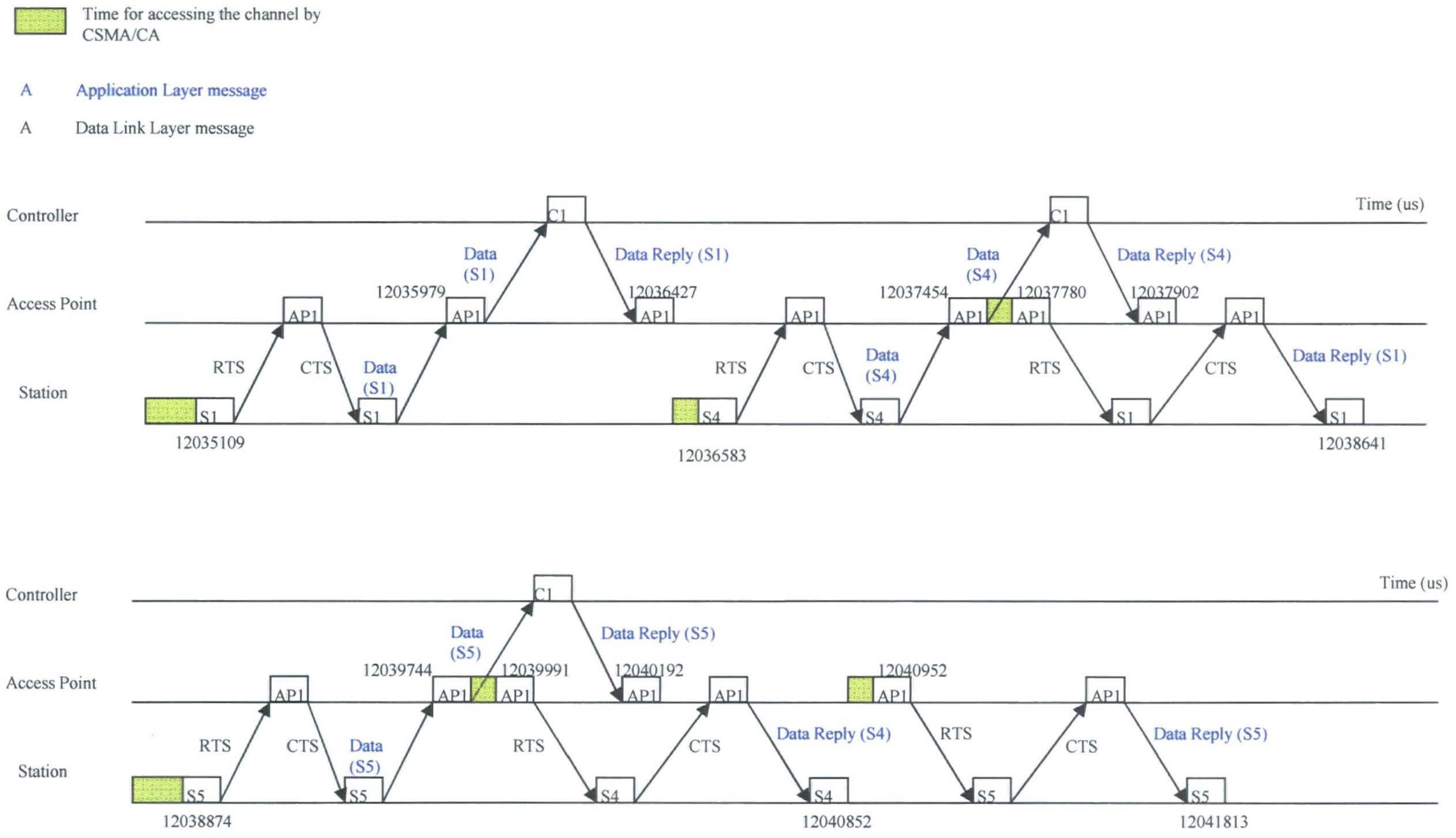


Figure 6.18 CSMA/CA

Access point received a data sent by station 1 and forwarded it to controller at 12035979 μs. Controller sent a data reply back and access point got it at 12036427 μs. Then station 4 got the chance to use the channel at 12036583 μs. Access point transferred the data sent by station 4 to controller at 12037454 μs and received the related data reply at 12037902 μs. There is a message list built in the access point, so the data reply for station 4 will not over write the data reply for station 1. Here from 12037444 μs (12037454 μs -10 μs), the radio was idle, because communication between access point and controller is by wire. Access point accessed the radio channel at 12037780 μs for sending RTS to station1. Station 1 received the data reply sent by controller via access point at 12038641 μs. Then station 5 sent RTS to access point at 12038874 μs and access point forwarded the data sent by station 5 to controller at 12039744 μs. At 12039991 μs the access point got the chance to send information and transferred the data reply to station 4. Station 4 got a data reply at 12040852 μs. Then access point accessed the channel again at 12040952 μs and station 5 received a data reply via access point at 12041813 μs.

6.4.3 No noise situation 2

In the above three simulations (30 minutes, 30 seconds and 60 seconds), all devices are set to work at the beginning of simulation. In a real scenario, maybe some devices start to work in the middle of simulation time or a certain time not the beginning. The simulation tool also can deal with this situation. Start time of devices are changed in configure file by changing related commands. The changed file is as follows:

```
CREATE CONTROLLER  C1
CREATE BASE_STATION  AP1
CREATE BASE_STATION  AP2
CREATE SLAVE S1
CREATE SLAVE S2
CREATE SLAVE S3
CREATE SLAVE S4
CREATE SLAVE S5

SETIP C1  193.168.168.1
SETIP AP1 193.168.168.10
SETIP S1  193.168.168.11
SETIP S2  193.168.168.12
SETIP S3  193.168.168.13
SETIP S4  193.168.168.14
SETIP S5  193.168.168.15
SETIP AP2 193.168.168.2

SETFREQ AP1 0 2424.5 2449.5
SETFREQ S1 0 2424.5 2449.5
SETFREQ S2 0 2424.5 2449.5
SETFREQ S3 0 2424.5 2449.5
SETFREQ S4 0 2424.5 2449.5
SETFREQ S5 0 2424.5 2449.5
SETFREQ AP2 0 2449.5 2474.5

SETTP AP1  2 3 4 0
SETTP S1  5 7 4  0
SETTP S2  7 9 3  8
SETTP S3  6 8 2  15
SETTP S4  3 7 4  25
SETTP S5  2 5 3  -1
SETTP AP2  60 62 4 0

CONNECT AP2 0 C1 1
CONNECT AP1 0 C1 0

START 0 0 30 0.000001
EXIT
```

The blue part is different from the original file. Station 2 will start to work at 8 s, station 3 will start to work at 15 s, station 4 is going to work at 25s and station 5 is set to -1s which means it will not work in the simulation time. Simulation time is 30 seconds and other parameters such as period of beacon and data are kept the same as above 30 seconds simulation. The simulation result is shown in table 6.19.

Station	Station 1	Station 2	Station 3	Station 4	Station 5
Data transmissions in 30s	5	4	3	1	0

Table 6.19 Data Transmission Results in 30s

The result of station 1 is the same as it in Table 6.19. Station 2 started to work at 8s. When it finished its first data transmission and reception, the remaining simulation time was less than 24 seconds (4* data_period); so station 2 just can finish four data transmission and reception. Station 5 did not start during the simulation time, so there was no messages exchanging between it and access point. The result is what is expected.

6.5 WLAN with Repetitive Noise

In this example repetitive noise is added for simulating the real radio environment. Repetitive noise can be divided into two groups: uncorrelated noise and correlated noise. Uncorrelated noise is the noise whose period is unrelated to the clock of the system under study, for example the period of a correlated noise can be 399.123456μs. An example of this type of uncorrelated noise would be noise from a motor, which is not related in any way to the radio system.

Correlated noise is the noise whose period is related to the clock of the system under study, for instance the period of noise is 400μs. There maybe more than one radio system operating on a particular frequency in a room or special space. They may interfere with each other. That is one example where correlated noise comes from.

6.5.1 Uncorrelated noise

The parameters of the simulation are the same as the no noise situation. Simulation time is 30 seconds, period of beacon is 1 second, period of data is 6 seconds, timeout for data is 0.6s and all devices start to work in the beginning of simulation time. The noise starts at 40μs; period of noise is set to 1599.12356456μs and every burst lasts 100μs.

There are five stations and hundreds of exchanged messages. It is not effective to analyze all the stations and all data, so one successful data transmission and data reply reception of station 3 is selected to be the measure to overall effect of the noise.

Based on the simulation results, Figure 6.20 (a, b) Re_noise1 (Repetitive Noise) are produced. The red line means the repetitive noise affected the communication at that time and the information was corrupted by the noise.

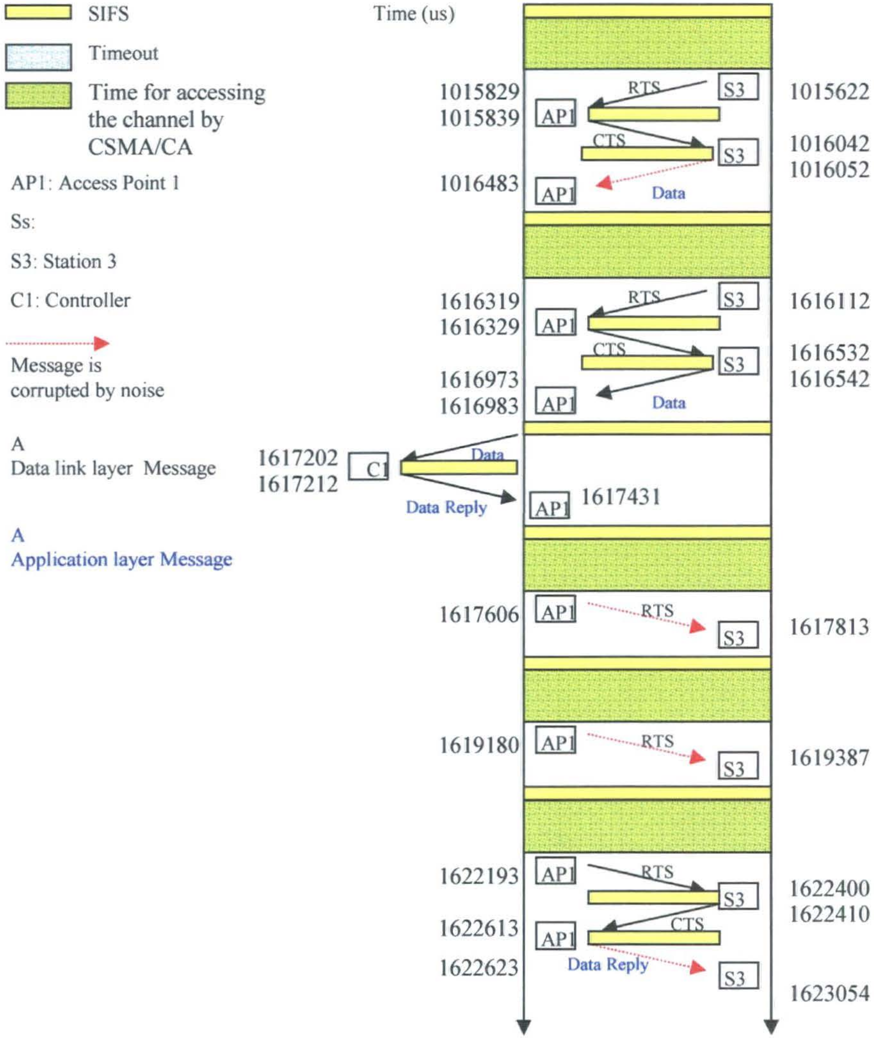


Figure 6.20a Re_noise1

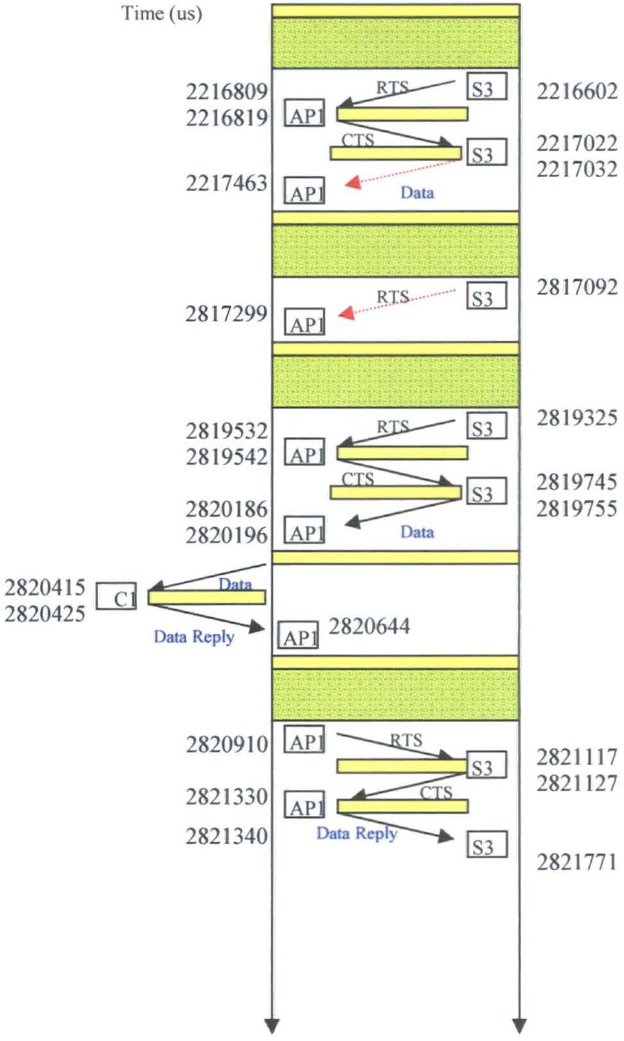


Figure 6.20b Re_noise1

After receiving the CTS sent by access point at 1.016042 s, station 3 sent data to controller via access point at 1.016052 s, but noise corrupted the message. When the timeout set in station 3 for data reply expired, station 3 had to retransmit the data and sent RTS first at 1.616112 s. Then the access point sent CTS back and station 3 sent the data to it. The access point then forwarded the data to the controller at 1.616983 s and got the data reply for station 3 at 1.617431 s. The access point got the chance to send RTS to station 3 at 1.617606 s, but the RTS is corrupted by noise. The access point could not correctly get the CTS sent by station 3 until 1.622613 s. Then the access point sent the data reply to station 3 at 1.622623 s, but station 3 picked up a message with errors because of repetitive noise. The timeout of station 3 triggered it to send the data to the controller again. Station 3 can not successfully finish this data transmission and data reply reception until 2.821771 s because of the effect of repetitive noise.

Table 6.21 compared the results of all stations with repetitive noise (Re_noise) and no noise.

Station	Station 1	Station 2	Station 3	Station 4	Station 5
30s (No noise)	5	5	5	5	5
30s (Re_noise)	5	5	5	5	5

Table 6.21 Results of Re noise and No Noise

Although repetitive noise affected the detail of the radio communication significantly, each station still finished five data transmissions and receptions in total. It means the system can work in the situation with this kind of noise.

In order to test the limit of this system, the noise period is changed to 799.12356456 μs to make the situation worse. Other parameters are kept the same.

Table 6.22 contains the results of all stations with no noise, repetitive noise whose period is 1599.12356456 μs and repetitive noise whose period is 799.12356456 μ s.

Station	Station 1	Station 2	Station 3	Station 4	Station 5
30s (No noise)	5	5	5	5	5
(Re_noise) 1599.12356456	5	5	5	5	5
(Re_noise) 799.12356456	0	0	0	0	0

Table 6.22 Results of Different Noise Periods

The noise was so frequent that no station can finish one successfully data transmission and reception in 30 seconds. The system can not operate under this situation. In figure 6.18 one sequence data transmission time that is the sum of an RTS transmission time, a CTS transmission time and a data transmission time is 861 μ s which is more than 800 μ s. That is why there is no station finishing one data transmission and reception. It also shows that the simulation tool works satisfactory.

6.5.2 Correlated noise

The noise period is changed to 1600 μ s and 800 μ s for simulating the correlated noise. The start time and duration of noise are the same as uncorrelated noise. Table 6.23 lists the results of two noise periods and uncorrelated noise.

Station	Station 1	Station 2	Station 3	Station 4	Station 5
30s (No noise)	5	5	5	5	5
(Re_noise) 1599.12356456	5	5	5	5	5
(Re_noise) 1600	5	4	5	5	4
(Re_noise) 799.12356456	0	0	0	0	0
(Re_noise) 800	0	0	0	0	0

Table 6.23 Uncorrelated Noise

If the noise period is so small that noise affected the radio communication too much, there is no difference between correlated noise and uncorrelated noise. For example, both in 799.12356456 μs case and 800 μs case, there is no station successfully finishing one data transmission and reception in 30 s. If the noise is not so frequent such as 1599.12356456 μs and 1600 μs, the difference between these two kinds of repetitive noises is shown in the related results. The system can deal with the repetitive noise whose period is 1599.12356456 μ s. Each station successfully finished five data transmissions and receptions in total. 1600 μs are more than one sequence data transmission time, so stations can finish data transmission and reception. The table 6.23 also indicates that the repetitive noise whose period is 1600 μs affected the system more than the last one. Station 2 and station 5 only finished four data transmissions and receptions.

The difference between correlated noise and uncorrelated noise can be more obvious, if their periods are changed to 1200 μs and 1199.12356456 μs.

Station	Station 1	Station 2	Station 3	Station 4	Station 5
30s (No noise)	5	5	5	5	5
(Re_noise) 1199.12356456	0	2	0	2	3
(Re_noise) 1200	4	5	3	4	4

Table 6.24 Results of Changed Periond

Although 1199.12356456μs can be considered to be equal to 1200μs and both of them is more than one sequence data transmission time, station 1 and station 3 can not finish even one data transmission and reception in 1199.12356456μs case. The total number of data throughput was 7 in 1199.12356456μs case, but 20 in 1200μs case (Table 6.24). Compared with no noise situation, the data throughput is reduced by 72% ((25-7)/25) in the uncorrelated case and 20% ((25-20)/25) in correlated case. The differences between these two results may be caused by following reason:

In the simulation tool the time step unit is 1 μs, so the first correlated noise and uncorrelated noise were considered to happen at the same time. When the simulation was continuous, the difference between correlated noise and uncorrelated noise became obvious such that the approximate difference between the thousandth correlated noise and uncorrelated noise is 877 μs that is more than one sequence data transmission time. As was mentioned before, the data period that is six seconds is constant, so if the data transmission is not corrupted by correlated noise, it may be corrupted by uncorrelated noise. Figure 6.25 gives the assumed situation to explain the reasons. That is why the two results are very different, although the two noise period is almost the same.

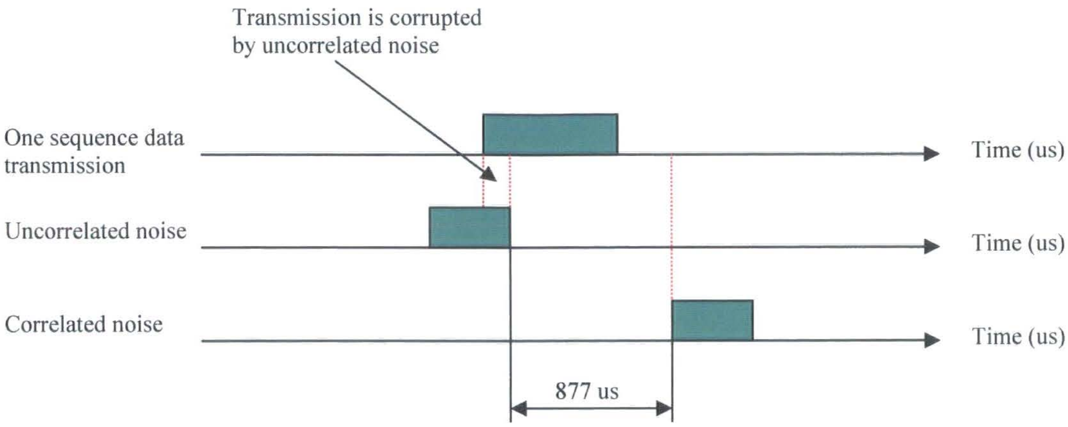

Transmission is corrupted by uncorrelated noise
One sequence data transmission
Time (us)
Uncorrelated noise
Time (us)
Correlated noise
Time (us)
877 us

Figure 6.25 Explanation for Uncorrelated noise and Correlated noise

6.5.3 The improvement

Repetitive noise has a common property that the period between each burst is a fixed value such as 1199.12356456 μ s and 1200 μ s. If the noise is not so frequent such that the period is 1199.12356456 μ s, devices may change their transmission period such as data period to reduce the probability of meeting noise.

Table 6.24 shows that the radio communication was serious damaged if the repetitive noise period is 1199.12356456 μ s. If the system has to work in such bad situation, data period is changed from 6 seconds to 6.5000001415926 s, 5.5000001415926 s, 5.8000001415926 s, 7.4000001415926s, 6.4000001415926 s, 6.500000 s and 6.800000 s to obtain a better result. Based on simulation results, table 6.26 lists two better results for operator.

Station in (Re_noise) 1199.12356456 us	Station 1	Station 2	Station 3	Station 4	Station 5
(Data period) 6 s	0	2	0	2	3
(Data period) 6.4000001415926 s	2	3	3	3	3
(Data period) 6.5000001415926 s	1	3	3	4	4

Table 6.26 Better Results by Changing Period

To change the data period is one way to increase data throughput. Compared with no noise situation, data throughput is reduced by 44% ((25-14)/25) in 6.4000001415926s case and 40% ((25-15)/25) in 6.5000001415926s case. Both of them are better than the 6 seconds data period.

The gap between two noise bursts is available for transmission. It seems that the two changed data period are much better at using the gap than the unchanged data period. That is the reason why the throughputs of 6.4000001415926s and 6.5000001415926s are more than 6 seconds.

This method for improving the throughput is based on the condition that the noise is known. In practice this condition may not be achieved. It means this method may not be useful.

6.6 WLAN with Random Noise

In this case random noise replaces the repetitive noise. The random noise may interfere with the communication at any time and may not happen during the communication time as well. Compared with the constant period of the repetitive noise, the period of the random noise is variable.

6.6.1 Random noise

Here the random period is between 1600 μs and 0. The random noise burst is also assumed to last 100 μs. Other parameters such as simulation time, period of beacon and data are the same as the no noise situation and repetitive noise situation.

Based on simulation results, table 6.27 lists the result of each station.

Station	Station 1	Station 2	Station 3	Station 4	Station 5
30s (No noise)	5	5	5	5	5
Ran_noise 1600	3	3	3	4	4

Table 6.27 Results of Random Noise

The table indicates that this kind of random noise did not affect radio communication significantly. Each station can not successfully finish five data transmission and reception. The throughput of data is reduced by 40% ((25-17)/25).

In order to investigate the increasing noise and decreasing noise, the limit of random noise period is changed to 1200 μs and 2000 μs. The related results are shown in table 6.28 (Ran_noise= Random noise).

Station	Station 1	Station 2	Station 3	Station 4	Station 5
30s (No noise)	5	5	5	5	5
(Ran_noise) 0-1600	3	3	3	4	4
(Ran_noise) 0-1200	1	0	0	0	0
(Ran_noise) 0-2000	5	4	5	4	4

Table 6.28 Ran_Noise with changing Period

When the random noise maximum period is reduced from 1600 μ s to 1200 μ s, the situation became worse. There was no station that could successfully finish one data transmission and reception except station 1. The system can not operate successfully with this level of noise.

When the maximum value is increased to 2000 μ s, data throughput was higher than before. Each station can successfully finish four data transmission and reception at least.

If the maximum of the period is increased, the probability of producing a period which is more than one sequence data transmission time is increased. That is why the larger the maximum of period is, the more the data throughput is.

6.6.2 The improvement

Data period was then changed to test whether it can improve the data throughput. Based on simulation results, three better results are shown in table 6.29

Station in (Ran_noise)	Station 1	Station 2	Station 3	Station 4	Station 5
(Data period) 6000000us	3	3	3	4	4
(Data period) 5400000.1415926 us	4	3	3	5	3
(Data period) 5400000 us	4	3	3	5	3
(Data period) 5800000.1415926 us	3	3	4	5	4

Table 6.29 Improved Results

There is no difference between 5.4000001415926 s and 5.400000s. If the data period is changed to 5.8000001415926 s, there will be two more data transmissions and receptions comparing with 6.0000000s. The reason for this improvement is the same as the reason in table 6.26.

6.7 WLAN with adjacent channel interference

In this simulation scenario, the two channels are assumed to interfere with each other because of the adjacent channel interference (mentioned in chapter 5). Because the devices operating on the licence free frequency band are usually low power transmitter and with high sensitivity receivers, here transmission power is set to 1 w and the sensitivity is set to -100db.

For the above three examples (No noise, Repetitive noise and Random noise) study, the configure files are the same. The configure file of this example is different from them because of adding another five stations operating at the same frequency band as access point 2. The blue part is the new part of following configure file.

```
CREATE CONTROLLER  C1
CREATE BASE_STATION  AP1
CREATE SLAVE S1
CREATE SLAVE S2
CREATE SLAVE S3
CREATE SLAVE S4
CREATE SLAVE S5

SETIP C1  193.168.168.1
SETIP AP1 193.168.168.10
SETIP S1  193.168.168.11
SETIP S2  193.168.168.12
SETIP S3  193.168.168.13
SETIP S4  193.168.168.14
SETIP S5  193.168.168.15

SETFREQ AP1 0 2424.5 2449.5
SETFREQ S1 0 2424.5 2449.5
SETFREQ S2 0 2424.5 2449.5
SETFREQ S3 0 2424.5 2449.5
SETFREQ S4 0 2424.5 2449.5
SETFREQ S5 0 2424.5 2449.5

SETTP AP1  2 3 4 0
SETTP S1  5 7 4  0
SETTP S2  7 9 3  0
SETTP S3  6 8 2  0
SETTP S4  3 7 4  0
SETTP S5  2 5 3  0

CONNECT AP1 0 C1 0

CREATE BASE_STATION  AP2
CREATE SLAVE S6
CREATE SLAVE S7
CREATE SLAVE S8
CREATE SLAVE S9
CREATE SLAVE S10

SETIP AP2 193.168.168.2
SETIP S6  193.168.168.22
SETIP S7  193.168.168.23
SETIP S8  193.168.168.24
SETIP S9  193.168.168.25
SETIP S10  193.168.168.26

SETFREQ AP2 0 2449.5 2474.5
SETFREQ S6 0 2449.5 2474.5
SETFREQ S7 0 2449.5 2474.5
SETFREQ S8 0 2449.5 2474.5
SETFREQ S9 0 2449.5 2474.5
SETFREQ S10 0 2449.5 2474.5

SETTP AP2  30 32 4 0
SETTP S6  32 32 2 0
SETTP S7  29 29 4 0
SETTP S8  29.5 32 3 0
SETTP S9  30 30.5 4 0
SETTP S10 34 32 3 0

CONNECT AP2 0 C1 1

START 0 0 30 0.000001
EXIT
```

Access point 1, station 1, station 2, Station 3, station 4 and station 5 operate at [2424.5 2449.5] MHz. They register with access point 1 and sent data to controller via access point1. Other stations such as station 6 operate at [2449.5 2474.5]MHz that is the same as access point 2, so those stations will register with access point 2 and send data to the controller via access point 2. In the controller, there is a member list or map. According to this map, the controller will send the data reply back via the correct access point. For example the controller sends the data reply for station 5 via access point 1.

If there is no interference, the results are listed in table 6.30.

Station in Access point 1	Station 1	Station 2	Station 3	Station 4	Station 5
No interference	5	5	5	5	5
Station in Access point 2	Station 6	Station 7	Station 8	Station 9	Station 10
No interference	5	5	5	5	5

Table 6.30 Results of No Interference

No matter which frequency the station use, each station can successfully finish five data transmissions and receptions.

In order to achieve the interference condition, position of station 2 is changed to (30 31.5 3) by command SETTP S2 30 31.5 3 0. Transmission power is increased to 8W.

Now station 2 is far away from access point 1, but very near access point 2. Simulation results that the communication between access point 1 and station 2 is affected by interference caused by nearby radio devices such as station 8 and access point 2, but the interference is not so serious. Station 2 still finished five data transmissions and receptions. Other stations are not affected by this interference, because they are not subjected to this condition. The result of each station is shown in table 6.31.

Station in Access point 1	Station 1	Station 2	Station 3	Station 4	Station 5
Interference	5	5	5	5	5
Station in Access point 2	Station 6	Station 7	Station 8	Station 9	Station 10
Interference	5	5	5	5	5

Table 6.31 Results of Interference

Although results in table 6.31 show no difference, the difference can be found in details. At 765 μs, 7048 μs, 20273 μs, 22106 μs, 6035371 μs, 12041129 μs, and 12042460 μs, S2 received messages with errors because of the adjacent channel interference. When the transmitter timed out, the message had to be retransmitted.

6.8 Conclusion

One controller, one access point and five stations constitute the configuration studied. Every device of the system follows 802.11b standard and uses IP for network layer and UDP for transport layer. The system performance is investigated in five situations: that of no noise, uncorrelated repetitive noise, correlated repetitive noise, random noise and adjacent channel interference (one more access point and another five stations are added in). In the no noise situation, the details of stations communication were drawn in figure 6.17 (a,b,c and d) for proving that simulation tool works satisfactory. Then uncorrelated repetitive noise and correlated repetitive noise are added in to the scenario. Although the period of the two noise sources can be considered to be equal such as 1199.12356456 μ s and 1200 μ s, the difference on affecting the radio communication can be very obvious. The worse situation of this system is also tested. For example, if the period of repetitive noise is 800 μ s or 799.12356456 μ s, no station can finish one data transmission and reception.

Random noise replaces repetitive noise for testing the system. The system can not cope with the random noise whose maxim period is less than 1200 μ s. If the random noise maxim value is increased to 2000 μ s, the system can work as well as the no noise situation.

There may be more than one radio system working together, so adjacent interference may be caused. In chapter 6.6, station 2 was changed to be close to access point 2 and far away from access point 1 for simulating adjacent interference. The results are that station 2 still can work well in such interference.

These simulation results can help system operators to decide whether the real system could work in a certain situation. Besides simulation of a certain system, the simulation tool also can help operators to find out how to improve the data throughput such as changing data period in chapter 6.4 and 6.5.

Chapter 7 Conclusion and future work

7.1 Conclusion

The simulation tool developed using C++ has achieved the original aim of simulating the behaviour of a given system or scenario and providing a tool for the evaluation of communication protocols.

1) This simulation tool can get low level performance information such as RTS, CTS messages exchanged in figure 6.11 (a).

2) This simulation tool can predict the interference or the low level protocol. The effects of noise that are divided into random and repetitive noise and adjacent channel interference are simulated in chapter 5 and chapter 6.

3) The simulation tool tests the design of application software. A WLAN using 802.11b standard is simulated in chapter 7 for five situations. These simulation results may help the designer to improve the operation of the system.

4) The simulation tool looks at overall system performance. Figures 6.17 (a,b,c and d) shows the details of communication process of station 3. The details of other stations such as station 1 are kept in a temporary file produced by the simulation tool.

5) The simulation tool provides information to improve the system. For example in chapter 6.5.3 and chapter 6.6.2 the data period is changed to improve the throughput.

7.2 Future work

This version of the simulation tool can be considered as the first version or base version and some parts of it need to be improved in future.

1) Noise simulation:

This simulation tool does not calculate the noise power. It is assumed that the noise will corrupt the communication no matter where the transmitter is. In a real situation, the noise power may decrease with distance. It means if the transmitter is far enough away from the noise source, it may not be affected by the noise. In future versions, noise power calculation should be added and the simulation tool should work out the level of the corruption from the noise properties and according to the noise properties such as power distribution, users can improve their system by some way such as changing devices position and data period.

2) Multipath fading

The multipath fading mentioned in chapter 1.4 has not been simulated in this work, because it is assumed that there is no obstruction between transmitter and receivers in free space. In order to be closer to the real system, it could be added in future. The received power calculation will be complex, as the received power is the sum of two or more different signals via different path such as figure 7.1.

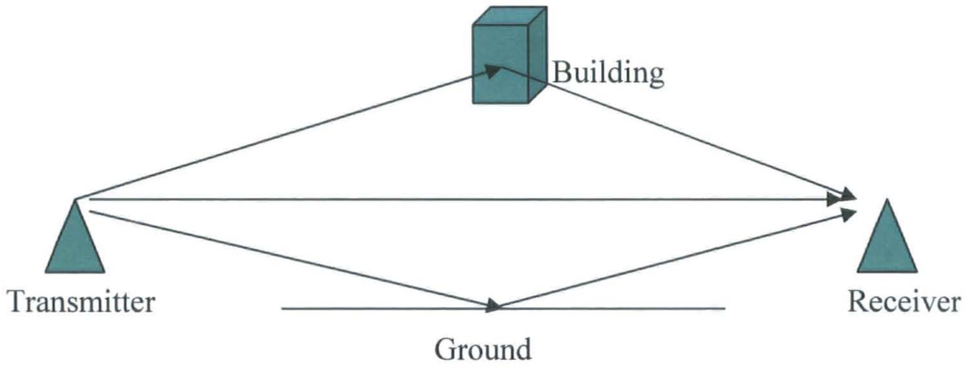


Figure 7.1 Multipath Transmission

3) Motion

In this thesis the radio devices are fixed. It means there are no motion and roaming problems in simulation. In the realistic environments, mobility is a main property of radio device such as mobile phone. The received power may be changed when the position of the device is changed. [46] gives a two-way model for simulating the mobile devices. This model can be added in future versions.

4) Roaming

If a radio device moves from one cell to another one and the connection is kept, this process is called roaming. During roaming, the device should register in the new access point and disconnect from the old one. If motion is added in future, the roaming should be considered as well. Signal power calculations are important. Devices should compare the received power from the old access point with the new one to decide whether it is time to switch access points.

5) Simulation tool structure:

In this simulation tool, 802.11b is simulated in WLAN. If users want to simulate another standard such as 802.16 [47], they may change or rewrite some parts of the program in data link layer. In future versions, it is proposed that users can choose the protocol they want to simulate in the configure file, for example 802.11, 802.15 [48] and 802.16. In the simulation model only data link layer chooses the related protocol, because the higher layers such as Application layer, Transport layer, Network layer are the same. Figure 7.2 shows the structure of the model.

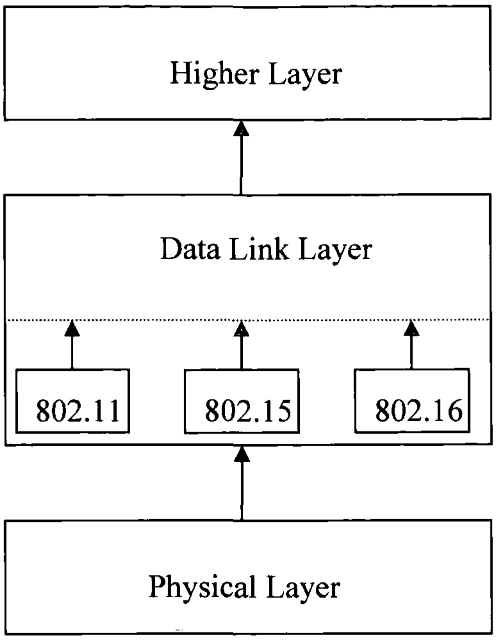


Figure 7.2 Improved Structure

If there is a new protocol, the programmer only adds a new part that simulates the protocol into the data link layer. When users choose the new protocol, the new model will be called.

6) Display of result:

As mentioned in chapter 4.4.5, users can get results by using some commands such as grep. In the beginning, these commands provide the details for each device which can help users to analyse whether the simulation tool works as what it is expected, but these commands may be inconvenient for users in following situations.

When the simulation tool follows the excepted protocol, the users only care about the data transmission and reception. In order to see the data transmission and reception, the users have to type the commands step by step. It wastes the users' time.

The simulator writes a data file containing all the results for this work 'grep' has been used to extract specific station results. In order to make the simulation tool more convenient, it is better to design a specific tool to process the results.

References list

[1] http://www.internetworldstats.com/stats.htm.

[2]John M. Senior, Optical Fiber Communications Principles and Practice (2nd ed, New York : Prentice Hall ,1992), p45.

[3] Vijay K. Garg and Joseph E. Wilkes, WIRELESS AND PERSIONAL COMMUNICTIONS SYSTEMS, (Upper Saddle River, N.J. ; London : Prentice Hall, c1996), pp.138.

[4] Theodore S. Rappaport, Wireless Communications Principles and Practice, (Upper Saddle River, N.J. ; London : Prentice Hall PTR, c2002), p71.

[5] Amit Dhir, Wireless Home Networks-DECT, Bluetooth, HomeRF, and Wireless LANs, (March 2001), p.1.

[6] William A. Shay, UNDERSTANDING DATA COMMUNICATIONS AND NETWORKS ,(Boston, Mass. ; London : PWS Pub. Co, c1995), pp.7-13.

[7] Roshan L. Sharma, Network Topology Optimization: The Art and Science of Network Design, (New York : Van Nostrand Reinhold, c1990), pp.8-13.

[8] http://www-groups.dcs.st-and.ac.uk/~history/Mathematicians/Shannon.html.

[9] D.C.Green, Radio Communication (2nd ed, Harlow: Longman, 2000), pp.28-109.

[10] James F.Kurose and Keith W.Ross, Computer Networking (2nd ed, Boston : Addison-Wesley, c2001), pp.430-432.

[11] A. Bruce Carlson, COMMUNICATION SYSTEMS An Introduction to Signals and Noise in Electrical Communication (3rd ed, New York : McGraw-Hill, 1986), pp.279-286, pp.363-368.

[12] Andrew S.Tanenbaum, Computer Networks (4th ed, Upper Saddle River, N.J. : Pearson Education/Prentice Hall PTR, c2003), pp.248-251.

[13] http://catb.org/~esr/jargon/html/B/big-endian.html.

[14]Tony Kenyon, Data Networks Routing, Security, and Performance Optimization (Amsterdam : Digital, c2002), p31.

[15] Tanenbaum (2003), pp.41-46.

[16] Radiometrix, BiM 418 or BiM 433.pdf (13 July 2001).

[17] AT86RF211 (TRX01) Datasheet.pdf (2002).

[18] CC1020 Data Sheet1.0.pdf.

[19] Radiometrix (July 2001), p12.

[20] http://www.isi.edu/nsnam/ns/

[21] http://www-nrg.ee.lbl.gov/ns/

[22] Ira Pohl, C++ by Dissection (Addison-Wesley, 2002), pp.168-173.

[23] Mike Banahan, Declan Brady and Mark Doran, The C Book Featuring the ANSIC Standard (2nd, ed, Wokingham, England ; Reading, Mass : Addison-Wesley Pub. Co,1991).

[24] Pohl (2002).

[25] Bjarne Stroustrup, The C++ Programming Language, (3rd ed, Reading, Mass. ; Harlow : Addison-Wesley, c1997).

[26] Ian Glover and Peter Grant, Digital Communications (London ; New York : Prentice Hall, 1998), pp.426-431.

[27] Carlson (3rd ed, 1988), pp.174-180.

[28] M. J. BUCKINGHAM, NOISE IN ELECTRONIC DEVIES AND SYSTEMS (Chichester [West Sussex, England] : E. Horwood ; New York : Halsted Press, 1983).

[29]http://murray.newcastle.edu.au/users/staff/eemf/ELEC351/SProjects/Fitzsummons/noiand.htm.

[30] http://www.swisswireless.org/wlan_calc_en.html.

[31] R. Bruno, M. Conti and E. Gregori, WLAN technologies for Mobile ad hoc Networks (IEEE Computer Society Washington, DC, USA, 2001), p3.

[32] http://www.atis.org/tg2k/_spread_spectrum.html.

[33] ME102 REFERENCE GUIDE.PDF, p7.

[34] ANSI/IEEE Std 802.11 (1999 Edition), p3.

[35] Tanenbaum(4th ed,2003), pp.295-296.

[36] ANSI/IEEE Std 802.11 (1999 Edition), p233.

[37] ANSI/IEEE Std 802.11 (1999 Edition), p90.

[38] ANSI/IEEE Std 802.11 (1999 Edition), pp41-44,55-73.

[39] http://www.spacerobots.org/dennis/Headers/IPv4header.htm.

[40] Tanenbaum (4th ed,2003), pp.436-448.

[41] Tony Kenyon, Data Networks Routing, Security, and Performance Optimization (Amsterdam : Digital, c2002), pp.60-80.

[42] Tanenbaum (4th ed,2003), pp.524-529.

[43] http://www.buzzle.com/editorials/8-24-2004-58325.asp.

[44] Tanenbaum (4th ed,2003), pp.532-555.

[45] http://www.geocities.com/kayakcraig/tcpip_model.html.

[46] Kaveh Pahlavan and Prashant Krishnamurthy, PRINCIPLES OF WIRELESS NETWORKS, (Upper Saddle River, N.J. ; London : Prentice Hall PTR, c2002), pp47-48.

[47] http://standards.ieee.org/getieee802/802.16.html.

[48] http://standards.ieee.org/getieee802/802.15.html.