Chapter 4 (Module D): Multiplexing and Multiple Access Techniques

In this chapter, we will investigate methods to sharing the available communications resources of a transmission medium among multiple sources on a single high-speed link to enhance the utilization of the link. First, we will look at the efficiency problem when using high bitrate communication links for low demand terminals to motivate the share of communication resources among multiple data streams through multiplexing and multiple access techniques.

Then, focusing on multiplexing techniques, we will study the principles and operations of Time and Frequency Division Multiplexing (FDM and TDM) to partition the available transmission resources in time and frequency domains. Various use cases of FDM and TDM will be presented to illustrate the use of these techniques in present systems, including Asymmetric Digital Subscriber Line, Wavelength Division Multiplexing and SONET.

Next, the focus will be turned to multiple access where a common medium can be shared among multiple users. The two mechanisms of multiple access, namely channelization and access control, are discussed. In channelization, we will examine the three basic schemes: Frequency-, Time-, and Code-Division Multiple-Access, i.e., TDMA, FDMA, CDMA. In access control, we will discuss both centralized and distributed approaches in demand-assignment, random-access: ALOHA, slotted-ALOHA, Carrier-Sense Multiple-Access (CSMA), and collision-free multiple access: bit mapping, and token ring. Characteristics and performance of each method will be identified and highlighted.

4.1 Introduction to Multiplexing (MUX) and Multiple-Access (MA) Techniques

Consider a communications link to support transmission of multiple input signals from node A to node B. One trivial approach is to provide multiple physical connections between A and B, each supports one input signal. In this way, the connection only needs to have enough communication resources, e.g., bandwidth, just for each individual input signal. However, the multiple physical connections can be too costly for installation, operation and maintenance, especially over a long distance and/or in a difficult environment. To reduce the costs, another approach could be to use just *one* connection with available communications resources sufficient to support multiple input signals. This cost-effective approach requires a process (called *multiplexing*) to combine (or *multiplex*) multiple input signals at the sending node A for transmission and to properly separate (or *de-multiplex*) them at the receiving node B. Conceptually, the available communications resources of the common transmission medium are divided into a number of sub-channels, each can accommodate one input signal in an efficient manner.

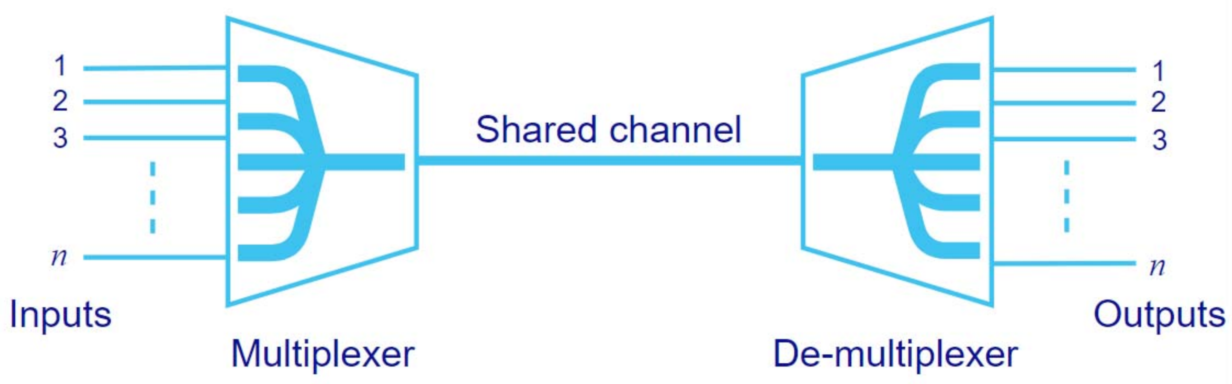


Figure 1: Multiplexing.

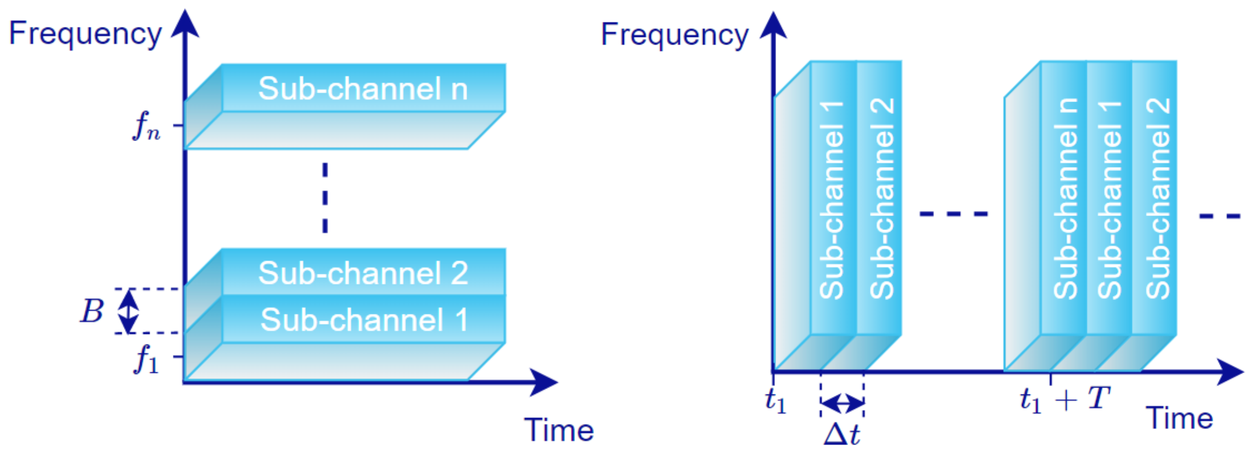
In a *multiplexing* technique, the two ends of the shared link are connected to a *multiplexer* (MUX) on the sending node and a de-multiplexer (DEMUX) on the receiving node as illustrated in Figure 1. Multiple ingress data flows are concentrated at the multiplexer that allocates the available resource portions to these flows depending on their needs and the capacity of the shared channel. The multiplexer outputs a single multiplexed data stream which carries all the input data flows. At the receiving node, the de-multiplexer follows the corresponding multiplexing rules to separate the individual data flows from each other as independent outputs. Depending on the employed resource allocation mechanism, the allocated resource portions for each input could be different from one another. In addition, the shared resource portion for each input could also be dynamically changed over time to support more data flows or to increase the transmission efficiency. For instance, if a data flow does not have any information to transfer, it would be more efficient to use some of its resource portions to support the data transfer of a new data flow. As such, there should be some signalling and synchronization between the multiplexer and the de-multiplexer so that the multiplexed data signals can be successfully separated at the other end of the channel. In Figure 1, the communications link is depicted as a single transmission channel; however, in practice, this single transmission channel can compose of many high-speed physical media, bundling together as one logical link. Due to the use of the multiplexer and de-multiplexer, multiplexing techniques are mainly used in backbone links where many data flows are aggregated.

Different from multiplexing techniques, in *multiple-access* techniques, no multiplexer and de-multiplexer are presented. *Multiple access* (or *channel access*) refers to a scheme to allow multiple communications devices connected to the same medium (or channel) for transmission and resource sharing. At first, the available communications resources must be divided into multiple sub-channels to be accessed by multiple communications devices. Secondly, there must be an established control procedure for multiple communications devices to follow in accessing a sub-channel. In other words, the participating communications devices must agree on some ways for allocating communication resources among themselves to be able to share the channel efficiently. The resource allocation and accessing scheme can be controlled in a centralized manner by a single scheduler. In this setup, participating communications devices must send their demands to the scheduler and the scheduler will perform the assignments for all participating communications devices based on the available resources, and their demands, in consideration of other possible parameters such as device priority, quality-of-service (QoS) requirements,… Participating communications devices must wait for their assignments before accessing to their allocated sub-channels. While such a *centralized* access control approach can provide good robustness and stability, its demand-assignment procedure can introduce noticeable and annoying delay, especially for short data transmissions. Alternatively, to reduce or eliminate the demand-assignment procedure, the resource allocation can also be conducted in a *distributed* manner where the participating devices follow some predefined procedures to determine the shared communication resources should be used by which device at any instance.

4.2 Multiplexing techniques

Multiplexing generally refers to the mechanism to combine multiple input signals for transmission over a common medium (or channel) to effectively and efficiently use its available communications resources. By integrating multiple low-speed sub-channels into one high-speed channel for transmission, the high-speed channel is effectively and efficiently utilized. By using a single multiplexed line, a communications link between the sending and receiving nodes can avoid maintaining multiple lines, to effectively save the operating costs.

As illustrated in Figure 1, multiplexing requires the use of a pair of multiplexer and de-multiplexer at the two ends of a communication link. In multiplexing, there are two main approaches in partitioning the communication resources. One can partition the communication resources based on frequency. In other words, the available large bandwidth of a high-speed link can be divided into multiple smaller frequency slots, or sub-channels, and each can be allocated to the appropriate data flow. On the other hand, the available communications resources of the communication link can be divided in the time domain where each data flow occupies a portion of transmitting time, or timeslot. According to these two main multiplexing schemes, namely *Frequency-Division Multiplexing* (FDM) and *Time-Division Multiplexing* (TDM) as illustrated in Figure 2.



a) FDM b) TDM

Figure 2: FDM and TDM

4.2.1 Frequency Division Multiplexing (FDM)

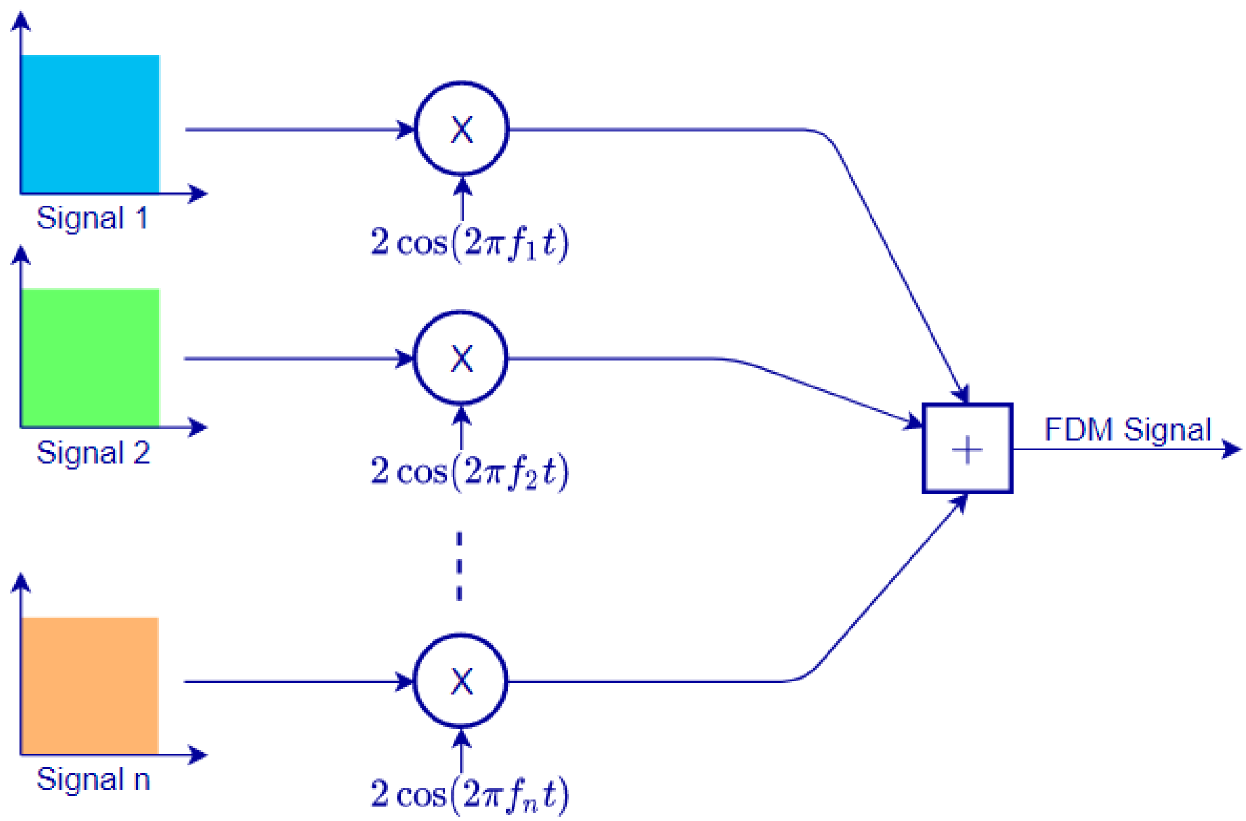


Figure 3: FDM Multiplexer structure

In Frequency-Division Multiplexing (FDM), the available bandwidth is divided into multiple non-overlapping frequency slots (or sub-channels), each accommodate one input signals into some of these sub-channels. Figure 3 illustrates the structure of an FDM Multiplexer (MUX) where the n input signals are assumed to be originally at baseband. Each baseband input signal is modulated by a sub-carrier to produce a modulated signal with the center frequency f_i and bandwidth B_i. Filtering at baseband as well as passband to confine the individual modulated signal within the specified frequency slot. These modulated signals with non-overlapping spectra are then added together to construct the combined FDM signal.

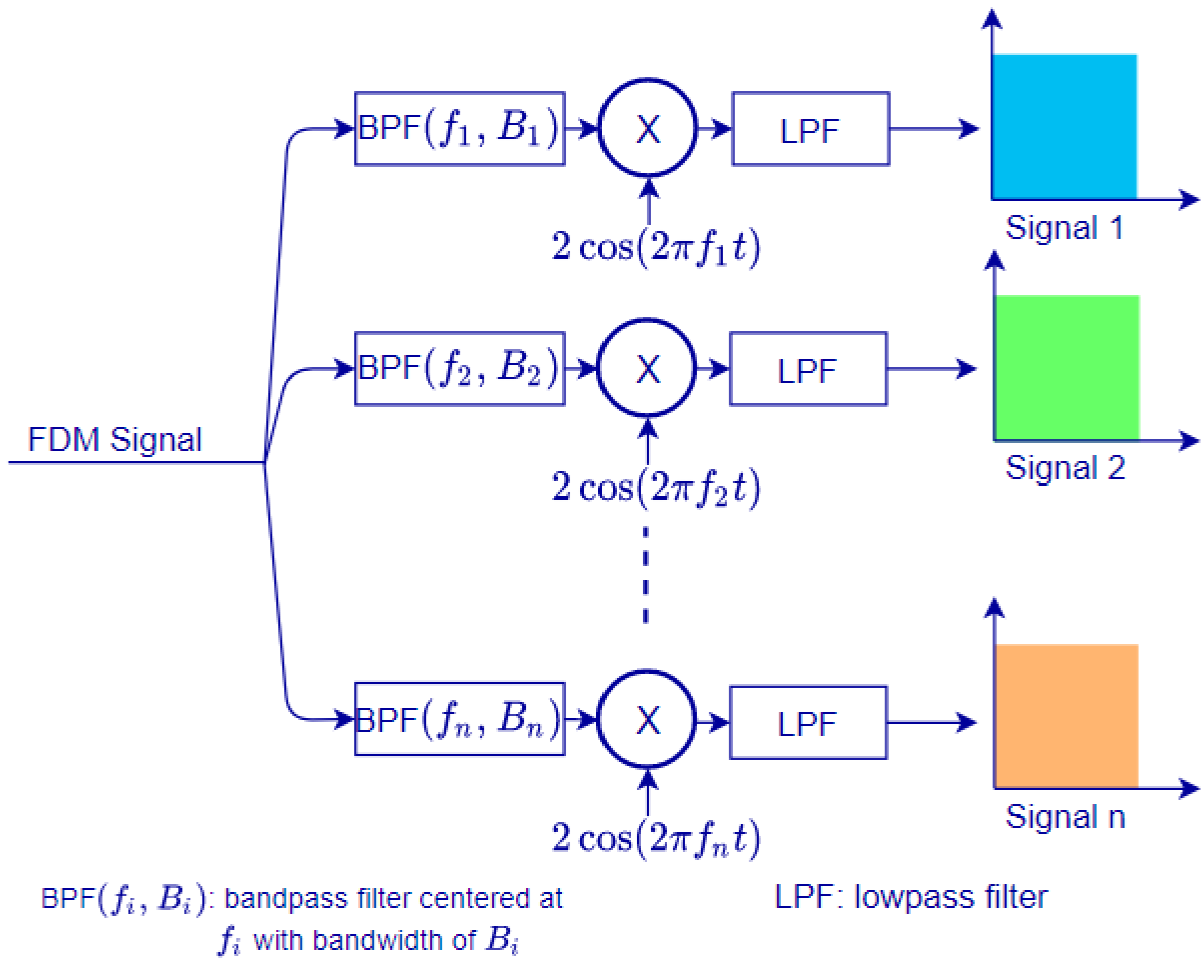


Figure 4: FDM de-multiplexer structure

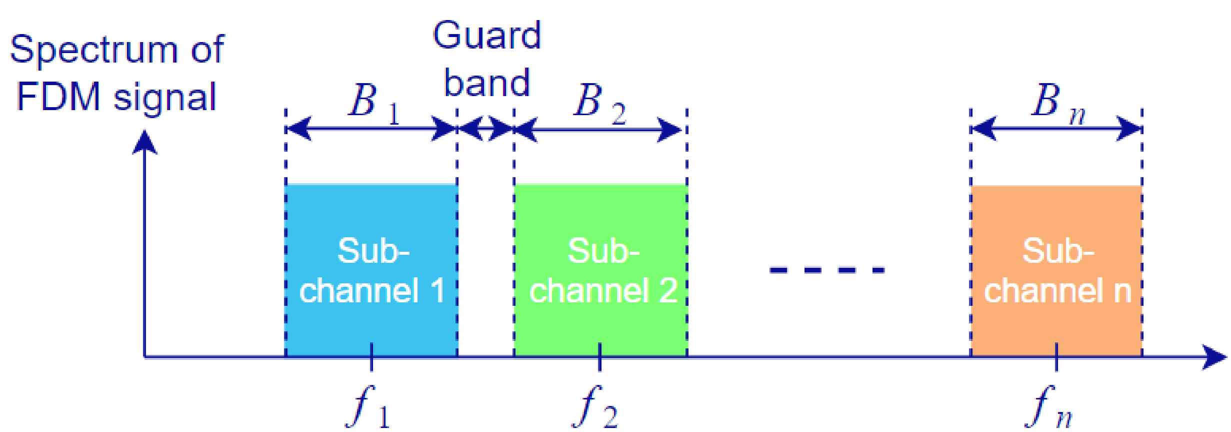


Figure 5: FDM signal spectrum

Figure 4 illustrates the structure of an FDM de-multiplexer (DEMUX). First, the combined FDM signal is filtered using bandpass filters centered at f_i with bandwidth of B_i

to separate the frequency sub-channels from each other. Since the transition slope of the bandpass filter frequency response cannot be infinite, a non-zero guard band is needed between two adjacent sub-channels as shown in Figure 5 that depicts the spectrum of an FDM signal. It is noted that these guard bands do not carry information and thus impose an overhead on an FDM scheme, i.e., with the use of these guard bands, the required bandwidth for an FDM link should be larger than the sum of bandwidths of all the input signals.

After bandpass filtering, each filtered signal will be demodulated using the same subcarrier frequency f_i and then further filtered by a lowpass filter to produce the originally baseband signal.

From the above discussion, it is apparent that FDM multiplexing is relatively simple to implement: the FDM MUX and DEMUX must operate with the same sub-carrier frequencies. However, FDM assumes non-overlapping sub-channels with unused overhead guard bands. This reduces bandwidth efficiency, especially with relatively small sub-channel bandwidths B_i's.

4.2.2 Orthogonal Frequency Division Multiplexing (OFDM)

As previously discussed, the traditional FDM approach is based on the bandpass filtering process to separate the input signals and hence needs non-overlapping sub-channels with unused overhead guard bands. As a result, guard bands become the limiting factor of the achieved bandwidth efficiency. For better bandwidth efficiency, it is desired to remove the guard bands, and this can be done by Orthogonal Frequency Division Multiplexing (OFDM). Indeed, signal separation in OFDM makes use of the *orthogonality* rather than the *non-overlapping* as follows.

Consider N data input streams organized in such a way that each has the same *symbol* rate of $1/T$ symbols/s where T is the symbol duration. The available frequency band is divided into N equally spaced sub-bands (or sub-channels) denoted by their center frequencies, $f_k = f_0 + \frac{k}{T}$, $k = 0,1,2,\dots,N-1$, where f_0 is the starting center-frequency, and $\frac{1}{T}$ is the sub-channel frequency spacing. Based on the Nyquist theorem on zero-ISI transmission, each sub-channel with a bandwidth $\frac{1}{T}$ can support one data input stream. As a result, for each symbol duration T, we group N symbols $X[k], k = 0,1,2,\dots,N-1$, from N data input streams to send over N sub-channels using N harmonically-related complex exponential sub-carriers ~~$\varphi_k(t) = e^{j2\pi\frac{k}{T}t}$, i.e., the *multiplexed* (or combined) signal in a symbol interval is $x(t)=\sum_{k=0}^{N-1} X[k]\varphi_k(t)$.~~ The *multiplexed* signal in a symbol interval is further sampled in N equal time intervals $\Delta t = \frac{T}{N}$ so that the *discrete-time* samples $x[n] = x(n\Delta t)$, $n = 0,1,2,\dots,N-1$, can be represented as $x[n] = \sum_{k=0}^{N-1} X[k]\varphi_k[n]$, where the *discrete-time* $\varphi_k[n] = \varphi_k(n\Delta t) = e^{j2\pi\frac{k}{T}n\Delta t} = e^{j\frac{2\pi}{N}kn}$. It can be verified that

$$\sum_{n=0}^{N-1} \varphi_k[n]\varphi_{k'}^*[n] = \sum_{n=0}^{N-1} e^{-j2\pi\frac{(k-k')}{N}n} = \begin{cases} 0, k \neq k' \\ N, k = k' \end{cases}, \text{ i.e., } \varphi_k[n]\text{'s are } \textit{orthogonal}. \qquad (4.1)$$

It follows that

$$\frac{1}{N}\sum_{n=0}^{N-1} x[n]e^{-jk\frac{2\pi}{N}} = \sum_{n=0}^{N-1}\left(\sum_{k'=0}^{N-1} X[k']e^{-jk'\frac{2\pi}{N}}\right)e^{-jk\frac{2\pi}{N}} = NX[k] \quad (4.2)$$

In fact, $x[n]$ is the inverse discrete-*frequency* (or discrete) Fourier transform (IDFT) of $X[k]$, and correspondingly, $X[k]$ is the discrete Fourier transform (DFT) of the discrete-*time* $x[n]$ for $n, k \in \{0,1,2, \dots, N-1\}$, i.e.,

$$X[k]=\frac{1}{N}\sum_{n=0}^{N-1} x[n]e^{-jk\frac{2\pi}{N}} \overset{\text{DFT}}{\longleftrightarrow} x[n]=\sum_{n=0}^{N-1} X[k]e^{j\frac{2\pi}{N}kn} \quad (4.3)$$

DFT is the appropriate Fourier representation for digital computer realization because it is discrete and of finite length in both the time and frequency domains. Fast Fourier transform (FFT) is a low-complexity fast DFT algorithm applicable for $N = 2^r$.

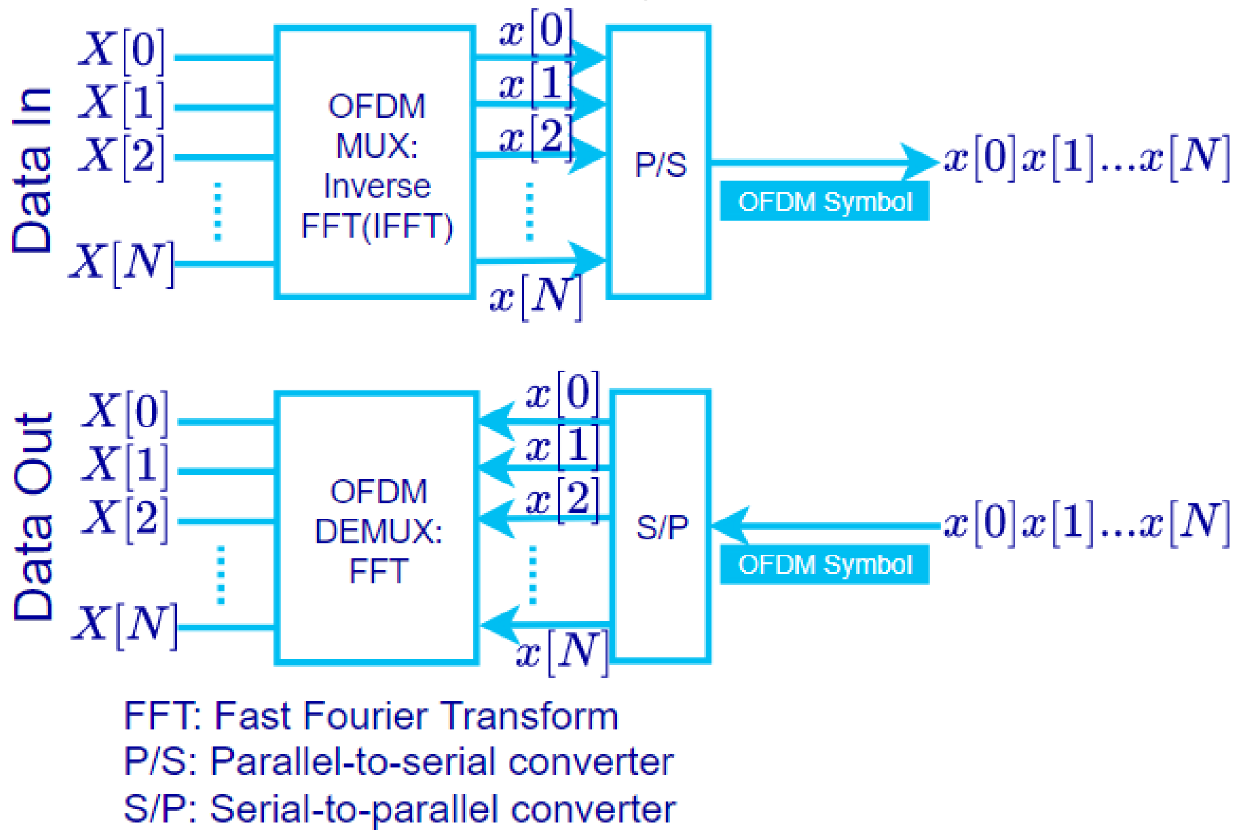


Figure 6: OFDM MUX and DEMUX

The above derivation shows that the separation (or extraction) of $X[k], k = 0,1,2, \dots, N-1$, of N data input streams from the multiplexed signal $x[n]$ can be successfully done by using the *orthogonality* to avoid inefficiency in spectral usage due to overhead guardbands in FDM. As such, the resulting multiplexing scheme is called Orthogonal Frequency Division Multiplexing (OFDM). The OFDM MUX and DEMUX block diagrams using FFT are shown in Figure 6. It is noted that the symbol of the k^{th} input stream $X[k]$ is generally represented as a *complex* value $X[k] = X_R[k] + jX_I[k]$, e.g., $X[k]$ represents a M-PSK or M-QAM symbol. The symbol $X[k]$ of the k^{th} data input stream is formed by m_k information bits and hence, has 2^{m_k} possible values, and the symbol sizes, m_k's, are not necessarily the same for all N data input streams.

OFDM has been widely used in both wireline and wireless communications. One key reason is that the number of sub-channels and the symbol sizes in different sub-channels can be flexibly selected to adapt to the channel frequency response as follows. Figure 7 illustrates an example frequency response of a frequency-selective fading channel. It indicates that the channel response *varies* at various frequencies. The entire channel bandwidth can be divided into a number of frequency slots with a sufficiently smaller bandwidth Δf so that channel frequency response across Δf is approximately flat. It is obvious that a signal transmitted over such a frequency-selective fading channel suffers different levels of attenuation (i.e., distortions) if its bandwidth is larger than Δf. Using OFDM, this wideband signal can be split into a number of narrowband signals, each with bandwidth of Δf, hence suffers only frequency-flat fading (attenuation). Furthermore, the symbol size of a narrowband signal can be selected in accordance with the channel response its frequency slot: a higher frequency response (i.e., lower attenuation) can accommodate a larger symbol size. In other words, if the transmitting end knows in advance the sub-channel frequency response (possibly from the receiving end via a feedback mechanism), it can adapt an appropriate modulation scheme for each frequency slot, for instance, BPSK with 1 bit/symbol for a frequency slot suffers from a deep (severe) fade and 256-QAM with 8 bits/symbol, for a frequency slot enjoys a very good channel response.

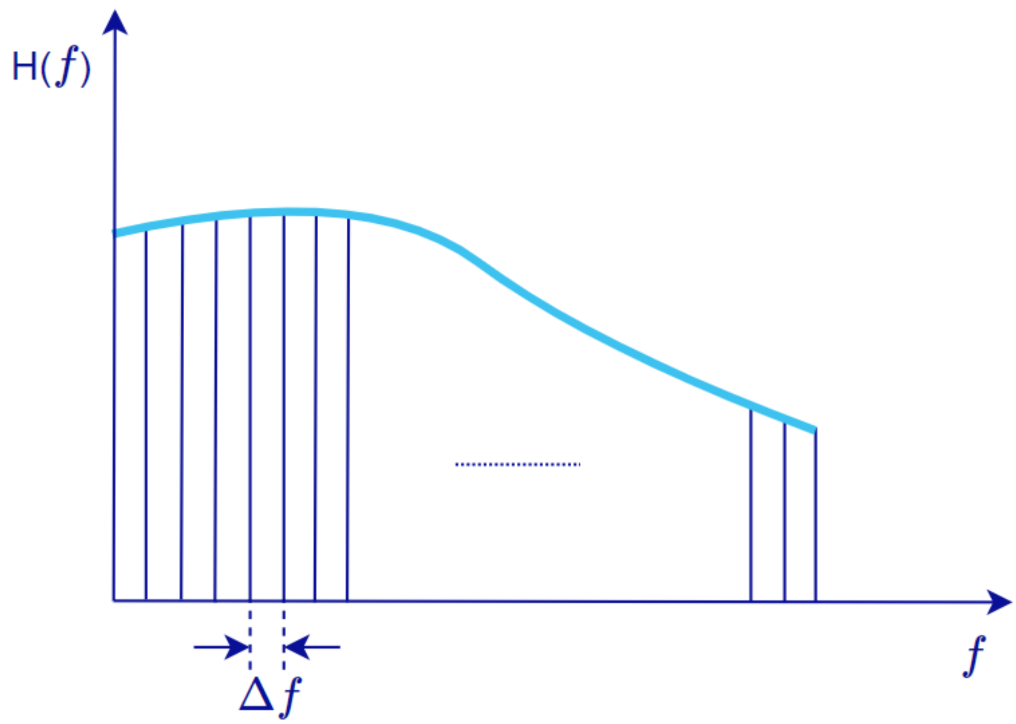
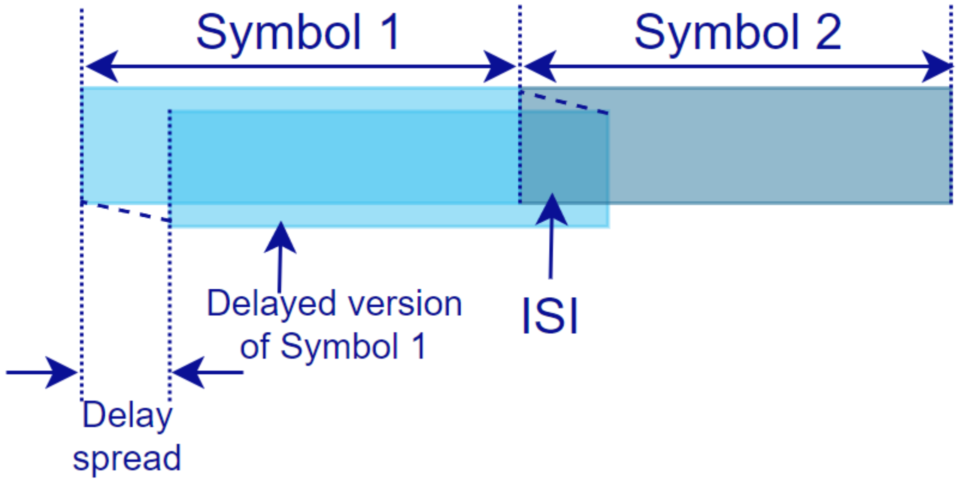
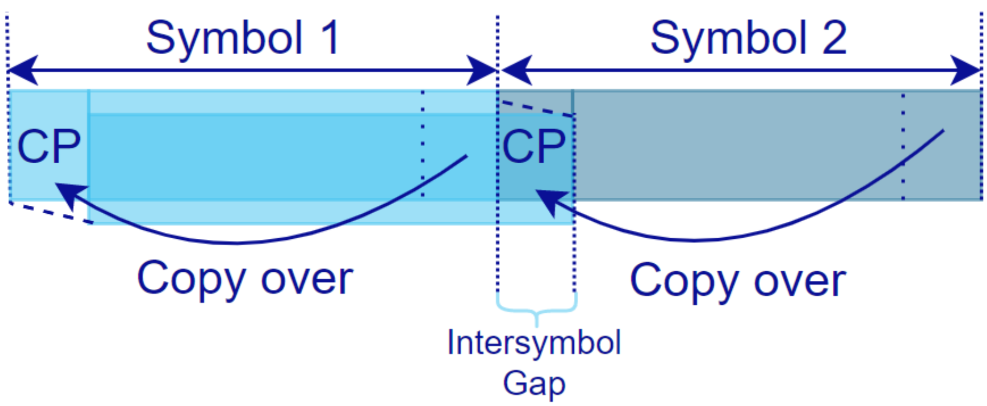


Figure 7: Example of the frequency response of a frequency-selective fading channel.

Frequency-selective fading exhibits delay spread in time domain, i.e., different frequency components of the signal arrive with different delays, causing interference between adjacent symbols with the next symbol, called *intersymbol interference* (ISI). As illustrated in Figure 8 a), the delayed version of one symbol due to frequency-selective fading overlaps with the next symbol, causing ISI. The size of the overlapping portion depends on the delay spread of the channel. To avoid ISI, a gap (or a guard space) not smaller than the channel *delay spread* must be inserted between adjacent OFDM symbols. Selection of an appropriate guard space must also consider the convolution of the channel impulse response with the transmitted signal. For this guard space, it was found that the tail of the OFDM symbol can be copied to the beginning of the same OFDM symbol. This so-called *Cyclic Prefix* (CP) shown in Figure 8 b) can avoids ISI because it acts as a guard space between successive OFDM symbol and it converts the linear convolution with the channel impulse response into a cyclic convolution.



a) Intersymbol Interference (ISI) due to delay spread.



b) The insert of CP

Figure 8: ISI and Cyclic Prefix to avoid ISI.

Figure 9 illustrates a simplified block diagram of OFDM transmitter and receiver wih CP insertion and removal. At the transmitter, N input data streams, each uses a modulation scheme such as QAM. The N input symbols are fed into an IFFT processor to produce an OFDM symbol in time domain. The tail of the OFDM symbol is copied to the beginning of the same OFDM symbol as a CP with length longer than the channel delay spread. At the receiver, the reverse process is applied. The CP will be removed from the received signal to reconstruct the OFDM symbol in time domain. The OFDM symbol is converted into the frequency domain by an FFT processor to recover the original data.

As discussed above, OFDM brings about many advantages including low-cost FFT/IFFT implementation, high spectral efficiency, simple frequency equalization, can mitigate the effect of ISI and selective frequency fading. However, it is not without disadvantages. First, the advantages of OFDM lies behind the orthogonality between its sub-carriers. As such, OFDM is very sensitive to frequency errors. If a sub-carrier frequency at the receiver is offset from that of the transmitter, the orthogonal property of its subcarriers is no longer valid. This can lead to the interference between the subcarriers, namely *Inter-Carrier Interference* (ICI), which will greatly penalize the received BER. The frequency offsets can also be caused by Doppler shift when the relative movement speed between the transmitter and receiver is high. Second, the combined OFDM signal exhibits a very high *peak to average power ratio* (PAPR), indicating a large variation in signal envelope. This large variation is very problematic in a practical system since it demands Analog-to-Digital and Digital-to-Analog Converters with large dynamic range and high-power amplifiers with large linear range, which is very costly. Failing to meet these demands will cause signal distortion, increase in noise floor and ICI in the output OFDM signals.

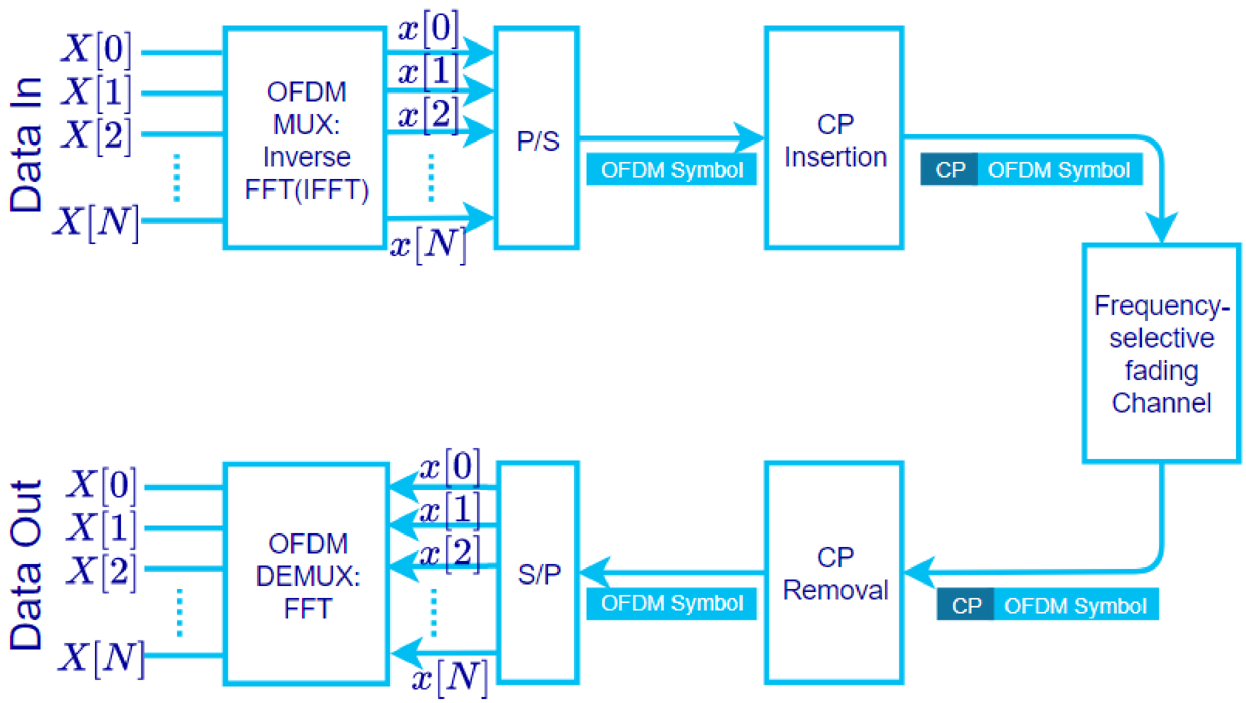


Figure 9: OFDM transmitter and receiver

As an illustrative example of OFDM, let consider ADSL (Asymmetric Digital Subscriber Line), a last-mile connection technology that provide high-speed data communication to residential users over the existing twisted-pair telephone lines. The existing twisted-pair copper lines in the telephone network are originally used for transmitting voice signal over the bandwidth from 0 to 4KHz. Nevertheless, the supported bandwidth of these very simple, thin wires is not restricted only to this frequency band. In fact, this twisted-pair cable can support the bandwidth of up to a few MHz but for a long time, this extra bandwidth was abandoned. Making use of this potential, ADSL technologies were developed to transmit high-speed data on the unused spectrum of the telephone lines. However, it is important to reckon that this surplus bandwidth reduces quickly with distance due to attenuation. Hence, the offered ADSL speed depends heavily on the distance from home to the nearest service point. A typical coverage radius of ADSL service is within 5km from the nearest service point, depending on the quality of the telephone line infrastructure.

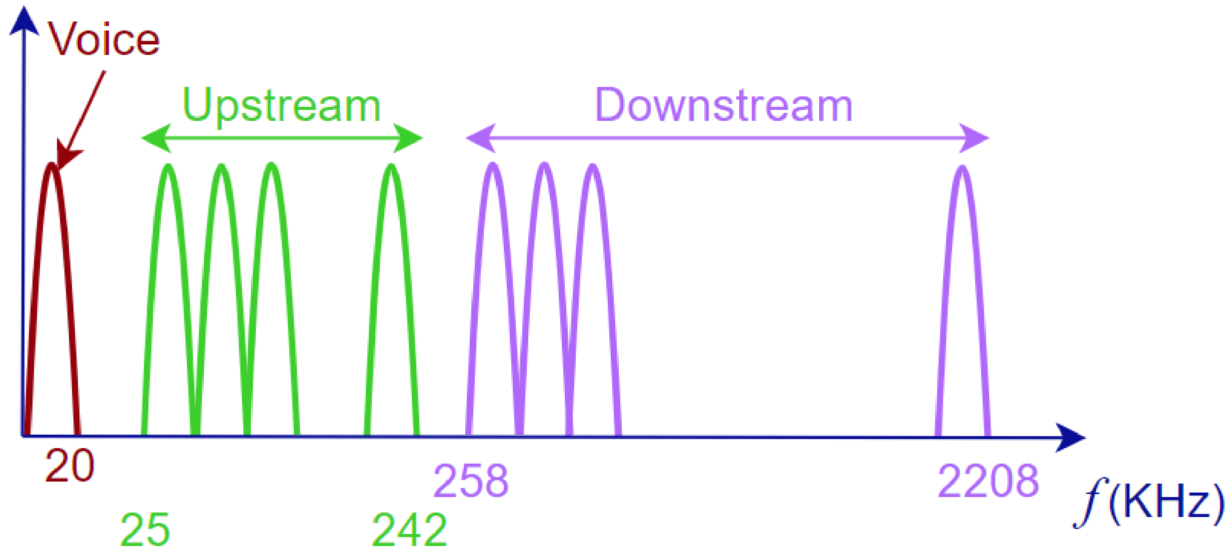


Figure 10: ADSL 2+ Annex M frequency band division.

As illustrated in Figure 10, ADSL 2+ Annex M [1] allows the use of available spectrum on a telephone line up to 2.2MHz. This spectrum is divided into three frequency ranges using FDM. The lowest frequency range of up to 20KHz is dedicated for traditional voice communication. From 25KHz, to 2.2MHz, the spectrum is divided into small sub-channels of about 4KHz each to be used with Discrete MultiTone (DMT), another name of OFDM commonly used in ADSL. To support full-duplex communications, these sub-channels are divided into two streams, the upstream that carries information from the user to the rest of the Internet and the downstream that carries information in the reverse direction. Between these two streams, some channels are left unused to prevent the interference between these two streams. Dividing the available bandwidth into multiple sub-channels can offer the following benefits:

- The service provider can flexibly adjust the portion of bandwidth between the upstream and downstream. In ADSL, typically, the downstream bandwidth is much larger (80%-90% the available bandwidth) and hence its bit rate is much higher than that of the upstream. This bandwidth allocation also accounts for the term *Asymmetric* in ADSL's name. For instance, one of the main differences in bandwidth allocation between ADSL 2+ Annex M and ADSL 2+ is the use of double the upstream bandwidth that allows the increase of the maximum upstream bit rate from about 1.4Mbps to about 3.3Mbps. For reference, the maximum downstream bit rate of ADSL Annex M is 24Mbps. It is noted that this maximum bit rate is valid only when the length of the link is within 0.5km from the nearest service point.

- By dividing the available bandwidth into multiple narrow frequency sub-channels, the channel response of each sub-channel can be flat, hence, ease the process of power allocation and equalization. Periodically, the two ends of an ADSL channel measure the sub-channel responses by sending out test signals on each of the sub-channels. The received power level of each sub-channel is measured to determine its frequency response. Based on this measurement, each sub-channel is assigned from 0 to 15 bits; sub-channel with a higher channel gain will be assigned with higher number of bits. Using a guided media, one may falsely assume that the channel response is invariant and need to be measured only once; however, in practice, this is hardly correct. As a result, this channel measurement needs to be repeated periodically.

- Using multiple parallel sub-channels allows the use of error correction codes to make the channel more robust. If the measurement of a sub-channel is not correct or is no longer valid, its carried data may experience a deep fade and hence cannot be detected. With the use of error correction codes, these lost bits may be recovered from the other successfully detected bits on other sub-channels.

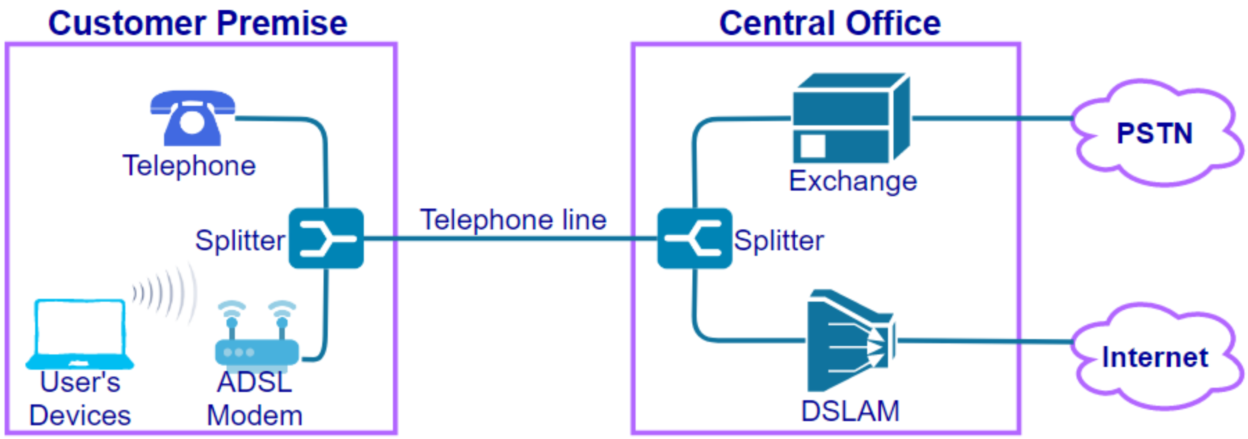


Figure 11: ADSL network structure.

In order to fit a data stream on to ADSL sub-channels, the user bit stream is divided into smaller sub-streams. Each sub-stream will then be transmitted over a sub-channel by using an appropriate QAM scheme. From Figure 9, a serial to parallel converter must be inserted at the beginning to split the input stream into multiple sub-streams. At the receiver, a parallel to serial converter is needed to re-arrange the received bits in a correct order.

Figure 11 illustrates a typical ADSL network structure. A telephone line connects the customer premise to a Central Office of the service provider. At each end of the telephone line, a splitter is used to separate the telephone and the data signals from each other. At the customer premise, a traditional telephone can connect directly to the splitter while a modem is needed to convert the ADSL signals to the appropriate LAN signals, such as wired or wireless LAN, for connecting to the user's devices. In the Central Office, the voice signal is routed to an exchange to connect to the Public Switched Telephone Network (PSTN), while the data signal is routed to a Digital Subscriber Line Access Multiplexer (DSLAM). The DSLAM does all the required signal processing similar to those in an ADSL modem to convert between ADSL signals and the information bits and connects to the rest of the Internet.

The separation of voice and data domain enables telephone companies to be able to provide Internet service to its existing customer quickly by only adding the data-domain devices without changing the existing voice network. This is one of the prime reasons that leads to the widespread of ADSL service nowadays.

4.2.3 Wavelength Division Multiplexing (WDM)

While the available bandwidth of each of the low attenuation window in an optical fiber is quite large, e.g., tens of THz, one of the main obstacles for fully exploring the capacity of fiber optics is due to the relatively lower speeds of electrical circuits, e.g., tens to hundreds of GHz and the conversion between electrical and optical signals. Wavelength Division Multiplexing (WDM) has been introduced to address this issue.

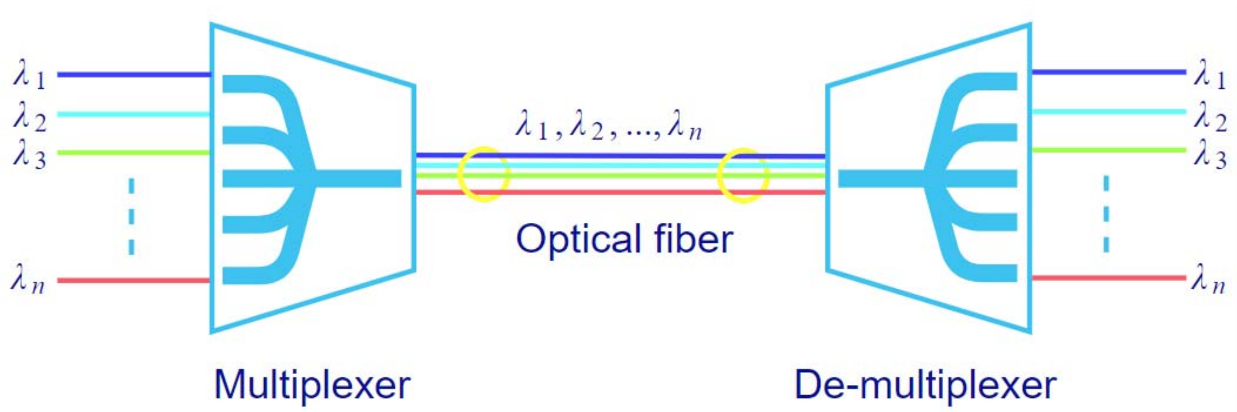


Figure 12: Wavelength Division Multiplexing.

In WDM, multiple low bandwidth sub-channels are combined together in a single optical fiber, each occupies a separated bandwidth centered at a different wavelength. In its essence, the basic principle of WDM is similar to FDM where the center wavelengths in WDM plays the role of the center frequencies in FDM.

Figure 12 illustrates the principle of a WDM system. At the transmitter, multiple data streams are modulated at different wavelengths $\lambda_1, \lambda_2, \ldots, \lambda_n$, then inputed to an optical WDM multiplexer, which combines all of these optically modulated input signals onto a single fiber for transmission. At the receiver side, an optical WDM de-multiplexer is used to separate these wavelengths from each other. As with FDM, as long as the bandwidth of each sub-channel or wavelength are non-overlapping, the multiplexing and de-multiplexing of optical wavelengths can be done non-destructively. However, different from FDM where the digital signal processing can be done on electrical signals; optical modulator, de-modulator, and filters are needed to process the optical signals and inefficient conversions between optical and electrical signals are required in WDM. As a result, the separation between the modulated wavelengths is typically large in comparison to the bandwidth of the individual channel. For instance, early WDM systems require a separation of 200GHz between the sub-channels. This large separation degrades the bandwidth efficiency of these WDM systems. With the advancements of technologies, this channel spacing is significantly reduced. Today, modern WDM systems can be deployed with a channel spacing of 50GHz or even only 12.5GHz [4]. This small channel separation allows the allocation of a very dense number of wavelengths onto the same transmission window, significantly increasing the transmission bandwidth efficiency of the optical fiber. For instance, a channel spacing of 12.5GHz allows a maximum of 560 wavelengths to be multiplexed on the 1550nm transmission window while a channel spacing of 50GHz allows only a maximum of 90 wavelengths transmitting in the same window. Due to this dense concentration of wavelengths, WDM technologies with small channel spacing are often referred to as *Dense Wavelength Division Multiplexing* (DWDM).

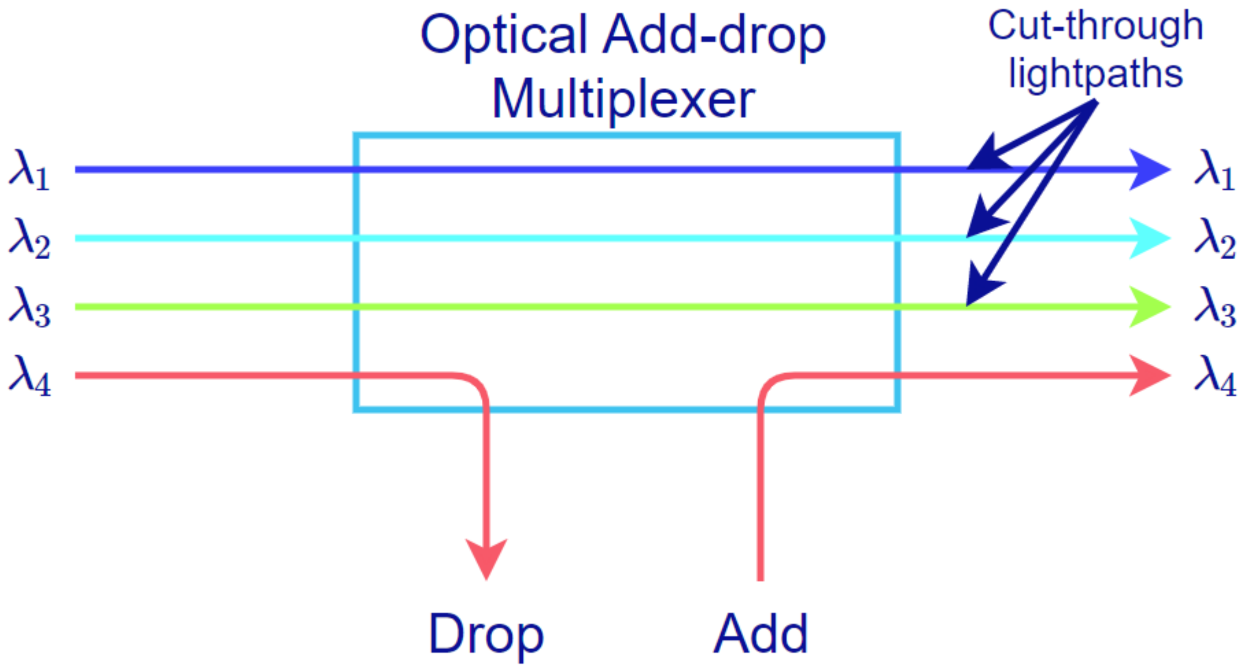
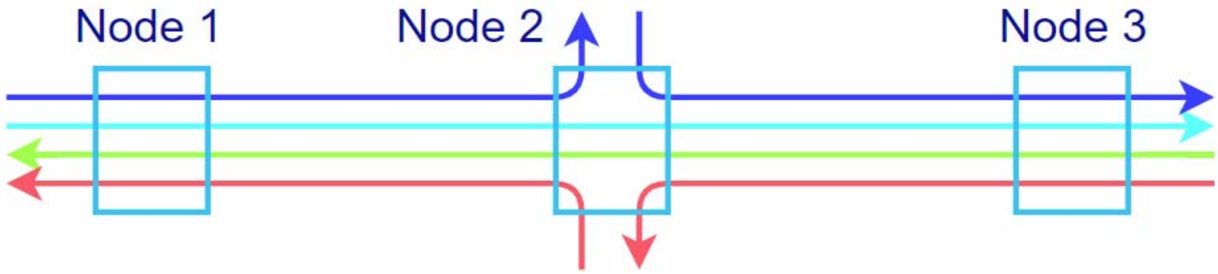


Figure 13: Optical add-drop Multiplexer

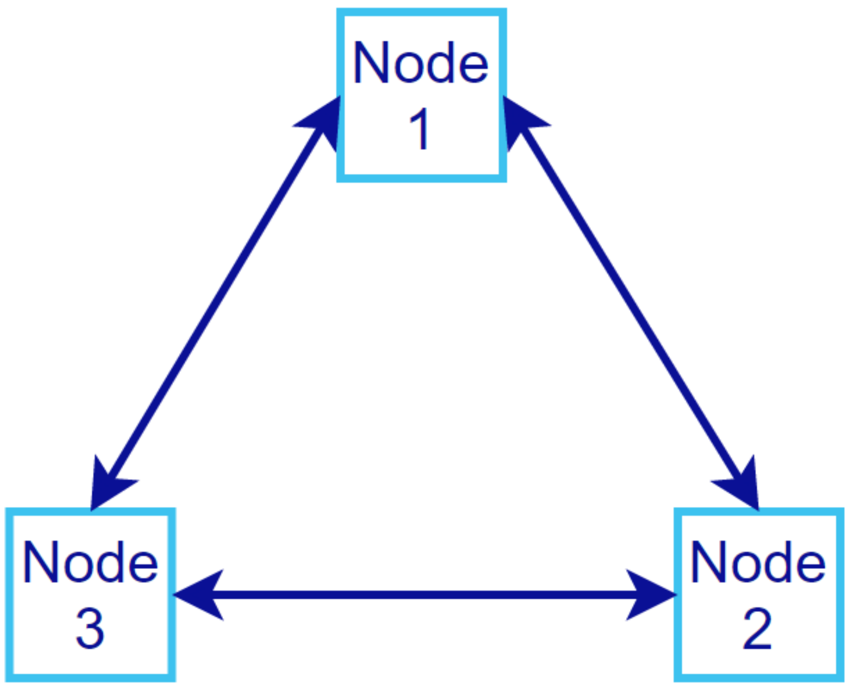
In recent years, multiple technological difficulties have been overcome to allow the processing of optical signals in the optical domain without the need to converting them to the electrical signals for processing and then back to optical signals for transmission over the optical fiber. For instance, optical amplifiers can be employed to boost directly the optical signals. More importantly, the development of optical add-drop multiplexers allows individual wavelengths to be added or dropped from an optical link entirely in the optical domain without affecting other signals.

Figure 13 illustrates an example of an optical add-drop multiplexer with four wavelengths. At this multiplexer, the information channel carries on wavelength λ_4 terminates and is "dropped" or removed while the other three wavelengths are passed through without being modified. At the output of the multiplexer, since λ_4 is available, it can be re-used to "add" or insert another information channel onto this wavelength. In this setup, a lightpath refers to a path of a wavelength between two nodes through which the wavelength is carried without being modified. As such, light paths of $\lambda_1, \lambda_2, \lambda_3$ are called *cut-through* light paths.

The use of optical add-drop multiplexer does not only allow the reuse of wavelengths, but it also allows the utilization of light paths to design flexible logical topologies that can look and behave very different from the physical topology. Figure 14 a) illustrates an example of this concept. In this figure, the physical topology between three nodes is implemented in series. With 4 wavelengths and the use of add-drop multiplexers, we drop two wavelengths and add two wavelengths at each of the nodes. The resulting logical topology is a ring topology as shown in Figure 14 b). As shown in this example, we can transform the serial physical connection to a ring topology. In practice, a physical ring topology is commonly used as it can flexibly adapt to complex logical topologies while allowing redundancy in the case of failures of physical links.



a) Physical topology and light paths.



b) Logical topology

Figure 14: Physical and logical topology with optical add-drop multiplexer.

4.2.4 Time Division Multiplexing (TDM)

Different from FDM, in TDM, many data streams share the same physical link by taking turn using the link. In TDM, each channel occupies the whole channel bandwidth but for only a limited time duration of its turn, or namely a timeslot. A timeslot may contain a bit, a byte or several bytes depending on the particular TDM standard and implementation. Figure 15 illustrates the principle of a simplified TDM system. In this system, multiple data streams are input into a TDM multiplexer, which transmits the data stream from input i for a timeslot then switchs to the next data stream in a round robin fashion. Effectively, on the physical channel between the multiplexer and de-multiplexer, the signals from the input streams are interlaced with each other. The period that a TDM signal revolve from the first timeslot through the last timeslots (timeslot n) is called a TDM frame. At the other end of the TDM link, a TDM de-multiplexer employs the reverse switching logic of the multiplexer to distribute the data in each timeslot to the appropriate output. For a TDM channel with n sub-channels, it is required that the capacity of the overall TDM channel must be at least equal to the total bit rate of all the participated sub-channels.

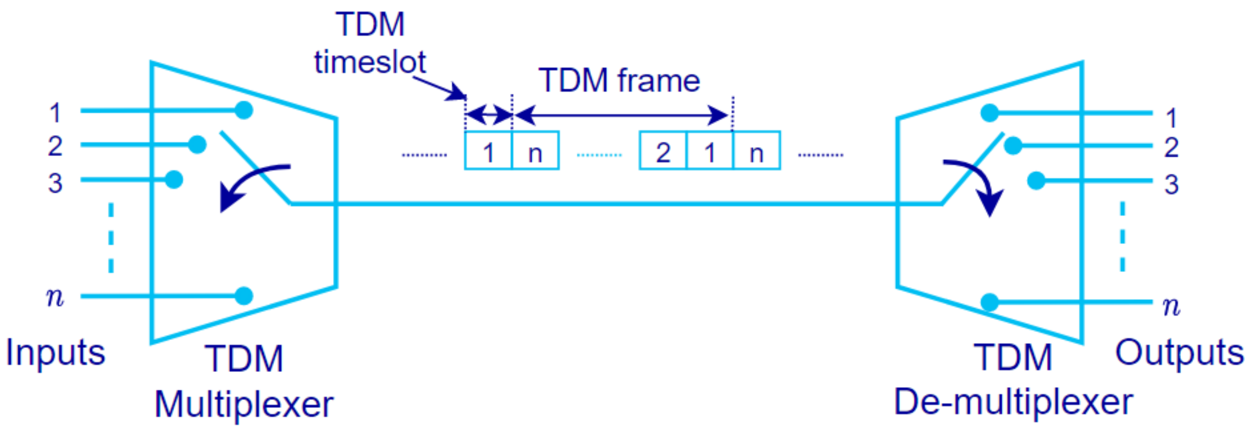


Figure 15: Time Division Multiplexing

It is noted that the TDM structure in Figure 15 only illustrates the basic principle of TDM and it hides some implementation details. For instance, in a TDM system, since the data rate of the output is significantly larger than that of the inputs of the multiplexer, each input of the multiplexer is connected to a buffer that enables the data rate conversion as well as storing the information bits from the input when it is not yet its turn to send data. Similarly, buffers are also used at the TDM de-multiplexer for the same purpose. In addition, depending on the implementation, a guard time may be used between the consecutive timeslots to avoid interference, similar to the concept of guard bands in FDM. Besides, within a TDM frame, there may exist non-information bits for link control purposes, such as for the synchronization between the two sides.

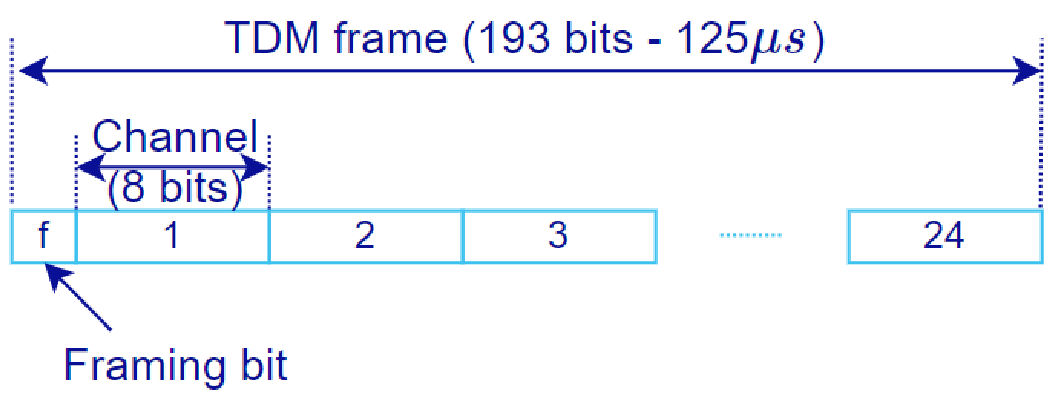


Figure 16: A T1 frame.

TDM was widely used in digital telephone system to multiplex multiple voice channels onto a high-speed trunk line. In a traditional voice system, each sub-channel sends 8 bits of voice data every 125μs, resulting in a bit rate of 64Kbps per voice channel. In the US and Japan, 24 of these voice channels are time multiplexed together to construct a T1 frame, where each channel occupies a timeslot in the T1 frame, as illustrated in Figure 16. In addition, an additional bit is added to the beginning of each T1 frame to facilitate synchronization at the receiver. For instance, these bits are inserted into each frame and follow a certain pattern. At the receiver, the receiver will search for this pattern in order to determine the beginning of a frame. Even after synchronization is successful, the de-multiplexer continues to check for these framing bits to make sure that it is still in synch with the multiplexer. Besides, the de-multiplexer has to repeat the synchronization process if it loses track of these framing bit. The total number of bits in a T1 frame is 193 bits and the bit rate of a T1 frame is:

$$R_{T1} = \frac{1 + 24 \times 8}{125 \times 10^{-6}} = 1.544\ Mbps$$

Table 4.1: North America and ITU data rate hierarchies

North American hierarchy		European hierarchy (ITU)	
T1	1.544 Mbps	**E1**	2.048 Mbps
T2	6.612 Mbps	**E2**	8.448 Mbps
T3	44.736 Mbps	**E3**	34.368 Mbps
T4	274.176 Mbps	**E4**	139.264 Mbps

Following the same TDM principle, a TDM hierarchy is constructed. A T2 frame is constructed from 3 T1 frames and have a bit rate of 6.312Mbps, A T3 frame is constructed from 7 T2 frames or 28 T1 frames with a bit rate of 44.736Mbps, etc. In Europe, a similar TDM hierarchical structure was also constructed by the ITU. However, the bit rates that are defined in ITU standard is slightly different than that in the US as illustrated in Table 4.1. The lowest TDM signal level of this hierarchy is an E1 signal that can support 30 voice channels with a bit rate of 2.048Mbps. Each of the higher signal level is combined from 4 signals of its previous level.

Synchronous Optical NETwork (SONET) Multiplexing

Table 4.2: SONET and SDH data rate hierarchies

SONET signal	SDH signal	Data rate (Mbps)
STS-1/OC-1	-	51.84
STS-3/OC-3	STM-1	155.52
STS-9/OC-9	STM-3	466.56
STS-12/OC-12	STM-4	622.08
STS-48/OC-48	STM-16	2488.32
STS-192/OC-192	STM64	9953.28
STS-768/OC-768	STM-256	38486.01

When optical fiber is introduced, telephone companies were amazed by its potential capacity. As a result, during the 1980s, optical fibers were widely deployed in the backbone long-range connections. However, at the time, the supported devices and signalling schemes were solely based on proprietary solutions. As the optical network grows, there was a need for compatibility and interoperation between different networks. In response to this demand, the SONET standard was developed in North America. Latter on, ITU-T also developed a similar set of standards for using in Europe, namely **Synchronous Digital Hierarchy (SDH)**.

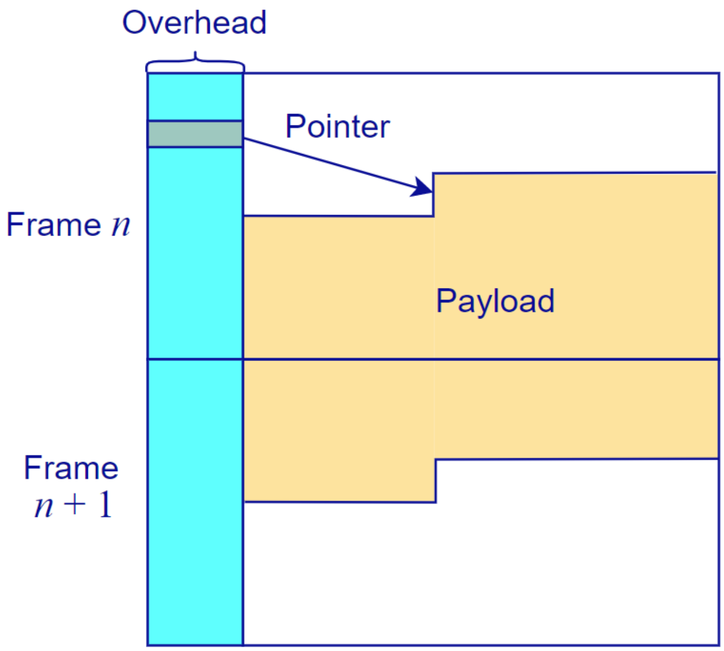


Figure 17: SONET payload can span over two frames.

The SONET specifications define standardized optical signals, synchronous frame structures and capabilities for operations, administration and maintenance. Regrading the digital data rates, SONET defines a hierarchical set of data rates with the basic data rate in the hierarchical is 51.84Mbps (Synchronous Transport Signal Level 1: STS-1, or Optical Carrier 1: OC-1). Operating based on a very accurate synchronous clock, a higher level signal in the hierarchy can be constructed by stacking multiple lower level signals. The signal hierarchy of SONET and SDH is shown in Table 4.2.

SONET organizes data into 810-byte blocks, namely frames. A SONET frame is arranged into a two-dimension array of 90 columns and 9 rows. The first three columns in a SONET frame are used for synchronization and management purposes and is divided into section overhead (first three rows) and line overhead (last 6 rows). Since SONET is a synchronous transmission scheme, it generates data frames continuously back-to-back without any gap and even when there is no user data. However, when user data arrive, they can be inserted into a being shaped up frame and can span over two frames. In order to identify the beginning of the data block, a pointer to the first byte of the data is written in the first three bytes of the line overhead as illustrated in Figure 17. Using this pointer also provides the ability for SONET to flexibly add/drop low data rate signals to/from a SONET signal without having to de-multiplexed the whole multiplexed stream. For more information about SONET standard, interested user can refer to the SONET homepage [5].

4.3 Multiple Access (MA) techniques

Multiple access (or *channel access*) refers to a scheme to allow multiple communications devices connected to the same medium (or channel) for transmission and resource sharing. A multiple-access (MA) scheme consists of two mechanisms: *channelization* and *access control*.

Channelization refers to the way to share the available communications resources of the common transmission medium. The available communications resources are essentially divided into multiple sub-channels to be accessed by multiple communications devices. In this context, *channelization* has a similar concept as multiplexing and is provided by the physical layer.

Access control refers to *how* a communications device (or user) can access a sub-channel, including various issues such as addressing, sub-channel assignment and related resolution protocols. In this context, it is also known as medium access control (MAC), which is a sub-layer in the data link layer of the OSI model and a component of the link layer of the TCP/IP model.

4.3.1 Channelization

Similar to multiplexing, channelization in multiple access divides the channel resources into small partitions or sub-channels for users to access. To this end, three major methods of channelization for multiple access are identified: the partition can be done in the frequency domain, time domain or code domain.

Frequency Division Multiple Access (FDMA)

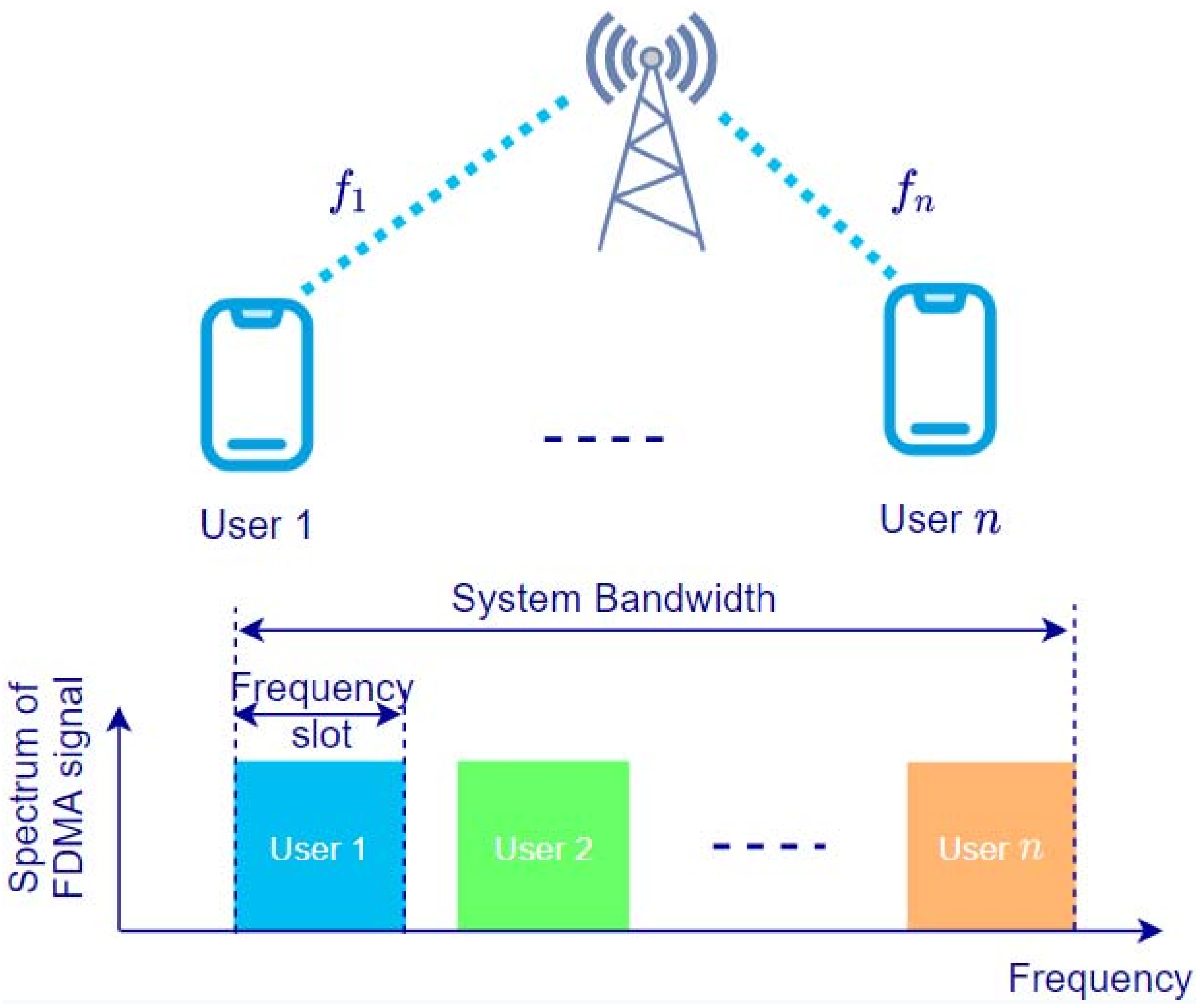


Figure 18: Principle of FDMA systems.

In this scheme, the available bandwidth of the common transmission medium is divided into smaller, *non-overlapping* frequency slots (as sub-channels) with *guard bands*, similar to FDM. As illustrated in Figure 18, the system bandwidth is divided into n sub-channels with bandwidth B_i and center frequency $f_i, i = 1,2, \ldots, n$. To avoid interference between the frequency slots, a guard band must be inserted between the adjacent frequency slots. Each of these frequency slots (sub-channels) can be used by one user only and the sending user has to modulate its passband signal centered at f_i with spectrum confined within B_i. The intended receiving user has to use a bandpass filter centered at f_i with bandwidth B_i to extract the signal and perform the corresponding demodulation and detection. For simplicity, each frequency slot in FDMA has the same bandwidth, i.e., $B_i = B$, $i = 1,2, \ldots, n$.

Orthogonal Frequency Division Multiple Access (OFDMA)

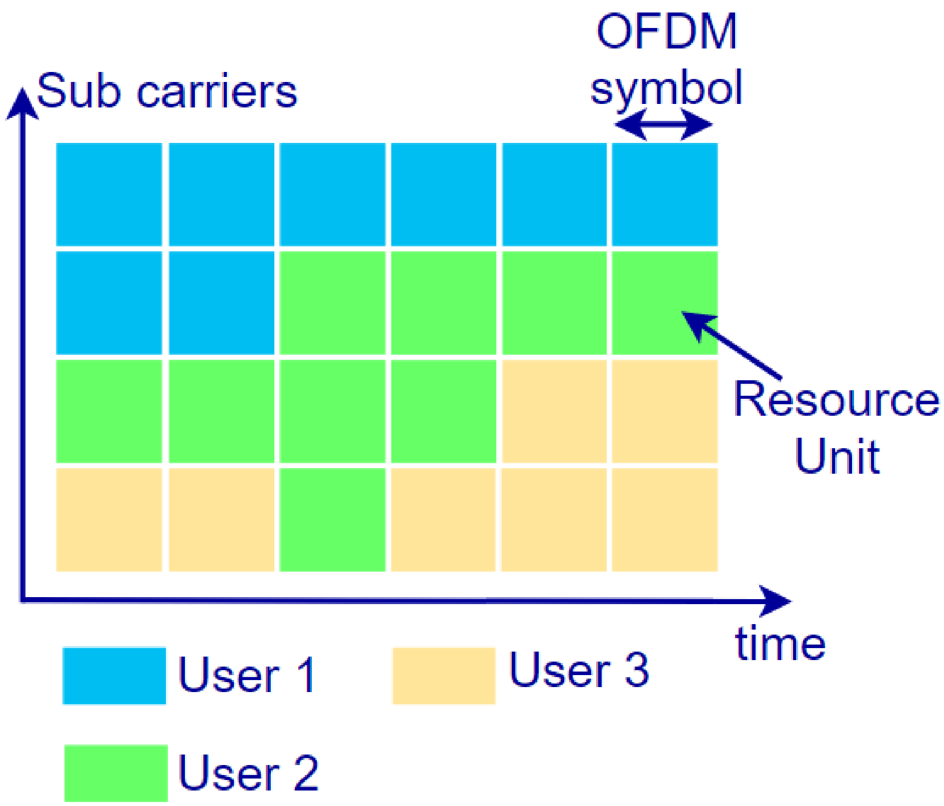


Figure 19: OFDMA resource units

To achieve a better spectral efficiency by eliminating guard bands, an Orthogonal Frequency Division Multiplexing (OFDM) structure can be used instead of non-overlapping FDM for accessing. The resulting access scheme is called Orthogonal Frequency Division Multiple Access (OFDMA) that needs more stringent synchronization and coordination than non-overlapping FDMA. For example, OFDMA and its variants have been recently used in cellular communications.

In OFDMA, the entire spectrum is divided into resource units (RU) that consists of a block of OFDM subcarriers spanning a duration of an OFDM symbol, as illustrated in Figure 18. These RUs are allocated to users by a centralized resource manager, such as a base station, based on their traffic demands. In this way, OFDMA can greatly enhance the resource usage efficiency by allowing multiple users transmitting at different speeds to share the same communication channel simultaneously instead of having only one user using the channel at a time with the traditional OFDM. However, as illustrated in Figure 20, in order to support OFDMA, all involved devices must not only be tightly synchronized with each other so that the combined signal appears as a single OFDM symbol at the receiver for de-modulation but also finely tuned in the transmit power of different RUs so that the power levels of different subcarriers arrive at the receiver(s) similarly. In addition, to be able to allocate RUs to users in an efficient way, the resource manager must be aware of the data availability at the users, which significantly increases the communication overhead and system complexity.

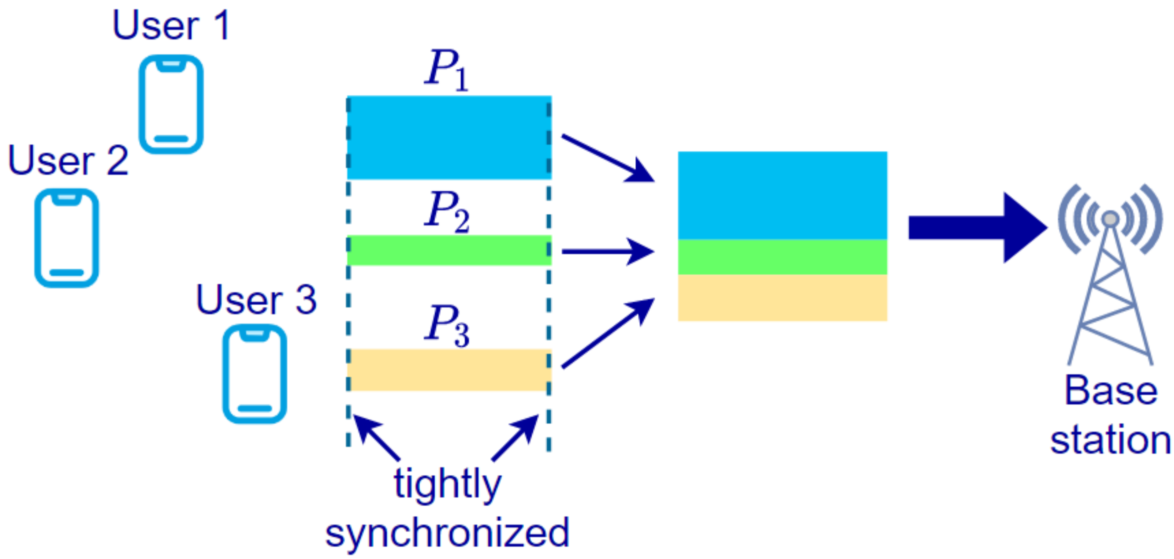


Figure 20: Synchronization and power control in OFDMA.

Time Division Multiple Access (TDMA)

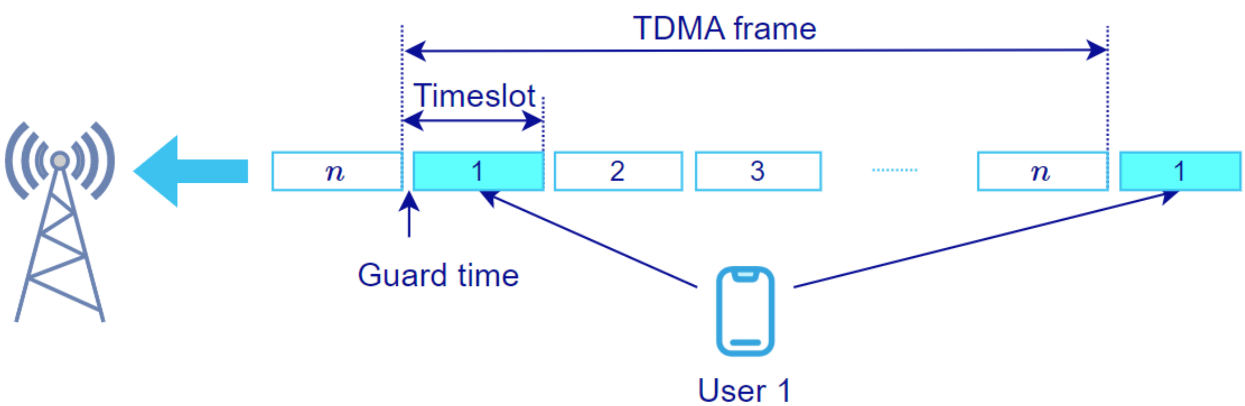


Figure 21: Principle of TDMA systems.

Similar to TDM, TDMA divides the channel resources into multiple *timeslots* (as sub-channels), i.e., In TDMA, a *TDMA frame* is divided into N non-overlapping timeslots, each can be used by *only one* user as illustrated in Figure 21. For simplicity, each timeslot normally has the same size. In TDMA, all users transmit and receive signals in a *non-continuous* manner, using the same carrier frequency, the same transmission bandwidth and the same TDMA *system transmission rate*. To avoid interference between the users, a guard time needs to be inserted between adjacent timeslots. In addition, to facilitate synchronization between the transmitter and receiver, a specific, predefined bit pattern (called pre-amble) is added at the beginning of the data segment sent in each timeslot. It can be seen that the TDMA *system transmission rate* in any timeslot is the same and represents the *highest supportable user data rate*. TDMA can easily support different and varying *user data* rates as long as they are lower than the TDMA *system transmission rate*. When its data rate increases the user need more timeslots per TDMA frame for transmission.

Code Division Multiple Access (CDMA)

In Code Division Multiple Access (CDMA), several transmitters can transmit information at the *same time* in the *same frequency band* by using different *spreading codes*, each represents a sub-channel as illustrated in Figure 22. Consider a channel shared by N users, and each user uses a specific *spreading code* (of sub-channel) for its coded signal. As a result, the received signal is the sum of N coded signals, from which the signal of a particular user of interest must be able to be extracted. The basic idea behind CDMA can be presented in a simplified manner as follows.

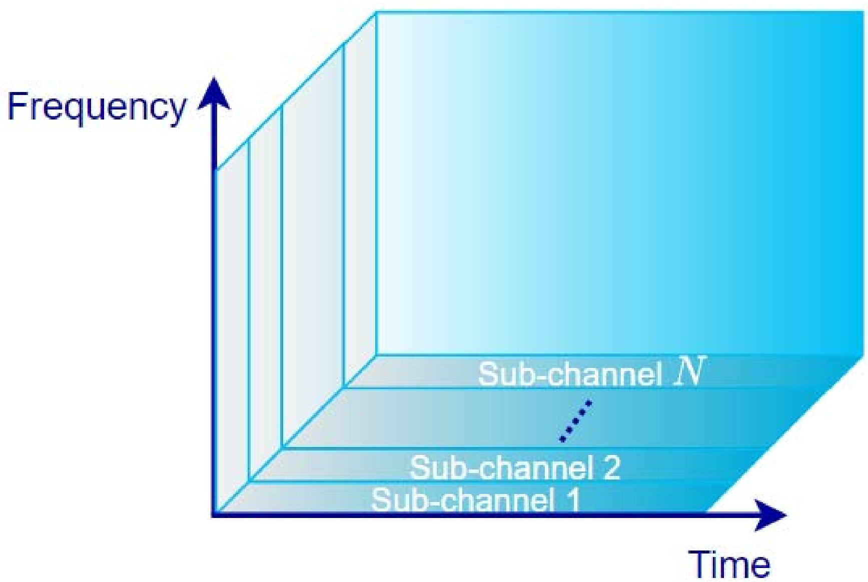


Figure 22: CDMA

As an illustrative example, each user n is assumed to have a BPSK (binary phase-shift keying) modulated signal represented by

$$s_n(t) = d_n(t) cos\omega_o t \qquad (4.4)$$

where $d_n(t) = \sum_{k=-\infty}^{+\infty} b_{n,k} r(t - kT_b)$ is its data bit stream, $b_{n,k} \in \{+1, -1\}$ is its data bit in the k^{th} bit interval, $r(t) = \begin{cases} 1, 0 < t < T_b \\ 0, elsewhere \end{cases}$, and T_b denotes the bit interval.

In L' bit intervals, user n further multiplies its L' data bits by a specific spreading code

$$\boldsymbol{c}_n = [c_{n,L-1}, c_{n,L-2}, \dots, c_{n,1}, c_{n,0}], c_{n,i} \in \{+1, -1\}, i = 0,1, \dots, L-1 \qquad (4.5)$$

where $L = L'L''$, and L'' is an integer, i.e., the spreading code is a binary sequence of length L and each element $c_{n,i}$ is called *chip* to avoid possible confusion with the data bit. The chip interval is T_c and $T_b = L''T_c$. The resulting *coded* modulated signal can be represented by

$$s_{nc}(t) = d_{nc}(t) cos\omega_o t \text{, where } d_{nc}(t) = d_n(t) c_n(t). \qquad (4.6)$$

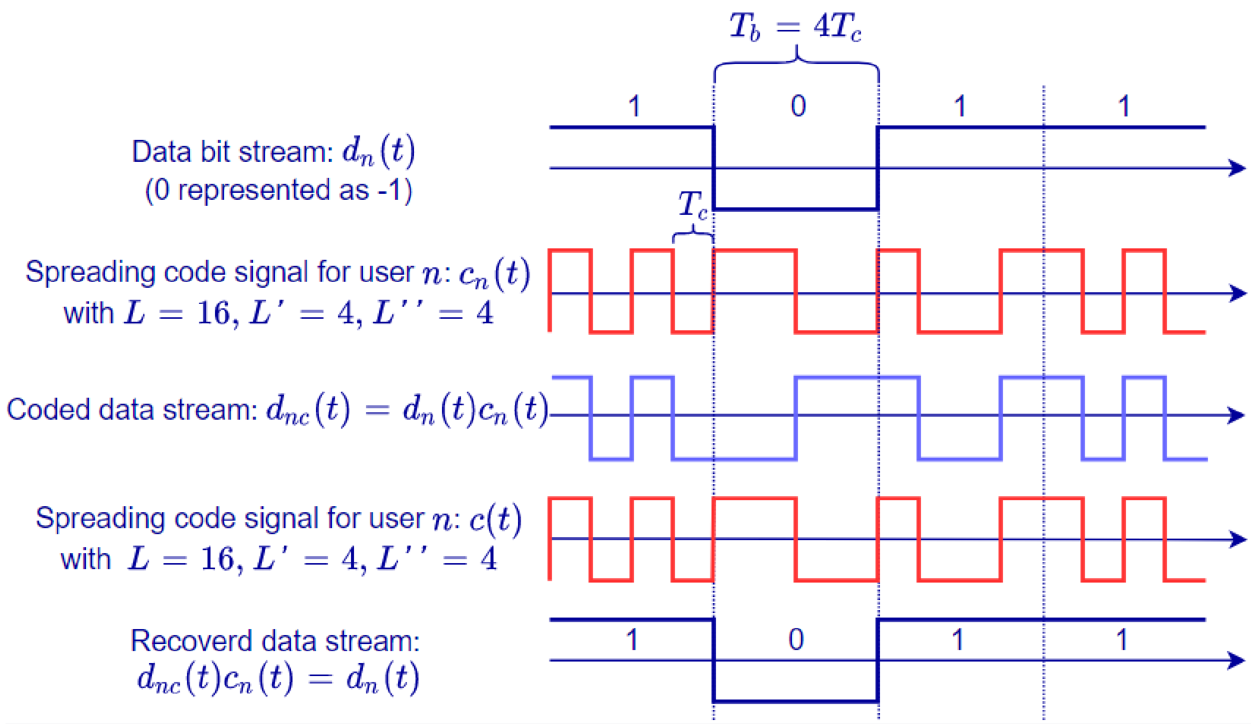


Figure 23: Example of coded and recovered data streams

The spreading code signal $c_n(t)$ can be represented as:

$$c_n(t) = \sum_{k=-\infty}^{+\infty} \sum_{l=0}^{L'-1} \sum_{i=0}^{L"-1} c_{n,l+i}\, r'(t - [l+k]T_b - [l+i]T_c)\,, r'(t) = \begin{cases} 1, 0 < t < T_c \\ 0, elsewhere \end{cases} \quad (4.7)$$

It follows that

$$d_{nc}(t) = d_n(t)c_n(t) = \sum_{k=-\infty}^{+\infty} \sum_{l=0}^{L'-1} b_{n,(k+l)} \sum_{i=0}^{L"-1} c_{n,l+i}\, r'^{(t-[l+k]T_b-[l+i]T_c)}, \quad (4.8)$$

and

$$c_n(t)c_n(t) = \sum_{k=-\infty}^{+\infty} \sum_{l=0}^{L'-1} \sum_{i=0}^{L"-1} \left[c_{n,l+i}\right]^2 r'(t - [l+k]T_b - [l+i]T_c) = 1 \quad (4.9)$$

since $\left[c_{n,l+i}\right]^2 = 1, c_{n,l+i} \in \{+1, -1\}$. Hence, $d_n(t)$ can be recovered by multiplying $d_{nc}(t)$ with $c_n(t)$, i.e.,

$$d_{nc}(t)c_n(t) = d_n(t)c_n(t)c_n(t) = d_n(t). \quad (4.10)$$

Figure 23 illustrates an example of the operations for coded and recovered data streams of user n where the spreading code $c_n(t)$ has $L = 16, L' = 4, L" = 4$.

For simplicity in the following discussions, let further assume

- The spreading code $\boldsymbol{c}_n = \left[c_{n,L-1}, c_{n,L-2}, \dots, c_{n,1}, c_{n,0}\right], c_{n,i} \in \{+1, -1\}$ is an element of a set of N *orthogonal* codes, i.e., for any $n, n' = 0,1,2, \dots, N-1$,

$$\sum_{i=0}^{L-1} c_{n,i} c_{n',i} = \begin{cases} 0, n \neq n' \\ N, n = n' \end{cases} \quad (4.11)$$

- $T_b = LT_c$ (i.e., $L' = 1, L" = L$), and

- A *synchronous* system[1] such that the composite signal in the shared channel is the sum of N coded signals and can be represented as
  $x(t) = \sum_{n=0}^{N-1} s_{nc}(t) = [\sum_{n=0}^{N-1} d_{nc}(t)] cos\omega_o t$ where the composite data stream is
  $d_x(t) = \sum_{n=0}^{N-1} d_{nc}(t) = \sum_{n=0}^{N-1} \sum_{k=-\infty}^{+\infty} b_{n,k} \sum_{l=0}^{L-1} c_{n,l}\, r'(t - kT_b - lT_c)$.

By re-arranging the first 2 sums in the above equation, the composite data stream can be rewritten as $d_x(t) = \sum_{n=0}^{N-1} d_{nc}(t) = \sum_{k=-\infty}^{+\infty} \sum_{n=0}^{N-1} b_{n,k} \sum_{l=0}^{L-1} c_{n,l} r'(t - kT_b - lT_c)$. This equation indicates that in the k^{th} bit interval, the composite data stream $d_x(t)$ contains $d_{x,k}(t) = \sum_{n=0}^{N-1} b_{n,k} \sum_{l=0}^{L-1} c_{n,l} r'(t - kT_b - lT_c)$. By multiplying $d_{x,k}(t)$ with the spreading code signal of user $n', c_n(t)$, and integrating the result over the bit interval T_b, we obtain $\int_{(k-1)T_b}^{kT_b} d_{x,k}(t) c_n(t) dt = T_b \sum_{n=0}^{N-1} b_{n,k} \sum_{l=0}^{L-1} c_{n',l} c_{n,l} = T_b b_{n',k}$ by making use of (4.11). In other words, the k^{th} data bit of user n' can be faithfully recovered from the composite data stream $d_x(t)$.

A systematic way of deriving orthogonal codes were developed by Walsh by forming an $n \times n$ square matrix in the following recursive manner

$$W_1 = [-1]; \qquad W_{2n} = \begin{bmatrix} W_n & W_n \\ W_n & \overline{W_n} \end{bmatrix} \tag{4.12}$$

The $n \times n$ Walsh matrix provide a set of n orthogonal codes of length $n = 2^m$, where each code is a row of the matrix. For instance, the Walsh matrices W_2, W_4 are

$$W_2 = \begin{bmatrix} -1 & -1 \\ -1 & 1 \end{bmatrix}, \quad W_4 = \begin{bmatrix} -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & -1 \end{bmatrix} \tag{4.13}$$

Figure 24 illustrates the Transmitter and Receiver with BPSK of a CDMA system. At the transmitter, the BPSK signal of user n, $s_n(t) = d_n(t) cos\omega_o t$, is multiplied by a spreading code signal $c_n(t)$. The bandwidth of the BPSK signal $s_n(t)$ is $1/T_b$while the resulting coded signal, $s_{nc}(t) = s_n(t) c_n(t) = d_{nc}(t) cos\omega_o t$, is also a BPSK signal but with a bandwidth of $\frac{1}{T_c} = \frac{L}{T_b}$, where T_c and $T_b = LT_c$ are the *chip* and data bit intervals, respectively. In other words, the bandwidth of the coded signal is spread by L times, and hence the coded signal is called spread-spectrum signal[2], and CDMA is also referred to as spread spectrum technique. Each sub-channel shown in Figure 22 is centered at ω_o and has a bandwidth of $1/T_c$.

At the receiver, the composite CDMA signal is the sum of all N spread-spectrum signals $x(t) = \sum_{n=0}^{N-1} s_{nc}(t) = [\sum_{n=0}^{N-1} d_{nc}(t)] cos\omega_o t$ from all N users. This composite CDMA signal is multiplied by the spreading code signal $c_n(t)$ of the intended user n, and then passed to the BPSK demodulator. The process of integrating the result over the bit interval T_b is actually performed by the so-called *integrate-and-dump filter* already included in the BPSK demodulator for detection of the data bits.

---

[1] The *synchronous* system assumption is needed to preserve the orthogonal property. This constraint is hard to achieve in practice. For this reason, pseudorandom codes and corresponding operations (beyond the scope of this chapter) are generally preferred.

[2] The spread-spectrum technique based on the multiplication with a spreading code described in this sub-section is called direct sequence spread spectrum (DSSS).

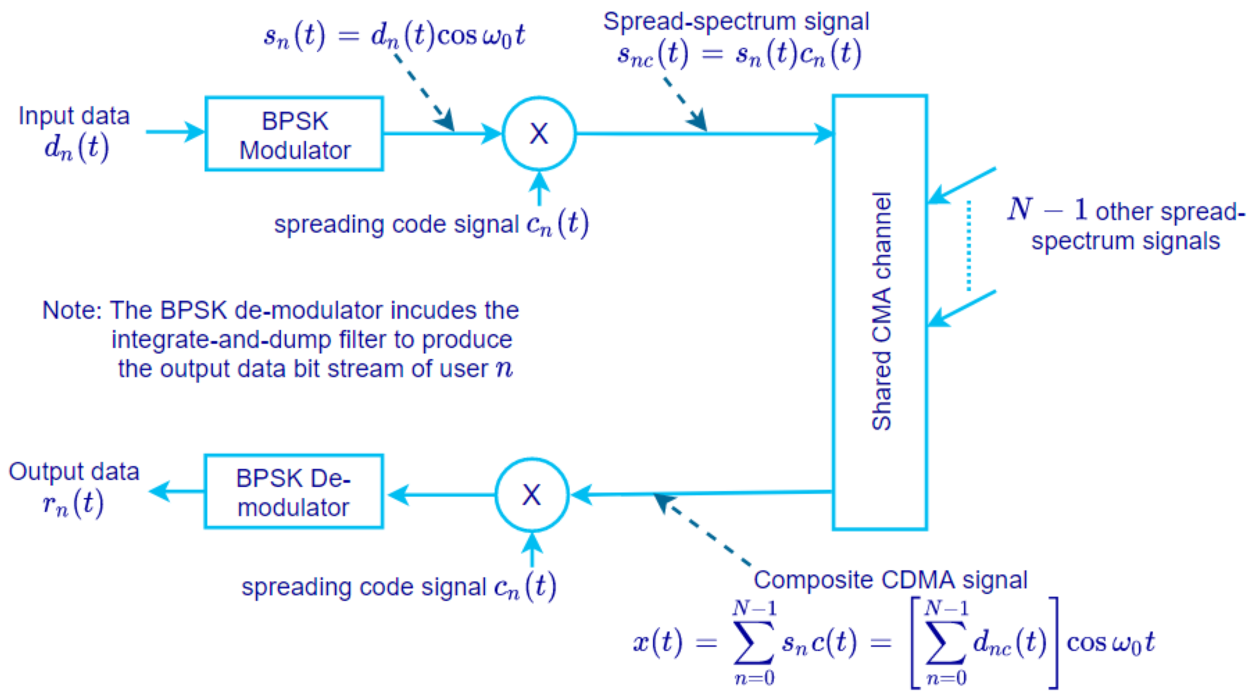


Figure 24: Transmitter and Receiver with BPSK in a CDMA system.

Advantages and disadvantages of CDMA

As discussed before, CDMA allows multiple users to transmit in the same frequency band at the same time without causing much interference to each other. Moreover, the wide bandwidth of the transmitted signal reduces the chance of jamming attack in which another station intentionally transmits onto the same frequency band to disrupt the communications. Moreover, with the use of different codes, data security can be enforced. A receiver which does not know the spreading code is unable to interpret the transmitted data. In addition, the wide bandwidth of a CDMA signal also offers frequency diversity in which frequency dependent impairments such as noise burst and multipath fading are less a critical issue. Another benefit of the wide bandwidth of CDMA is that it is possible to transmit data at a very low SNR.

On the other hand, CDMA is not without its drawbacks. First, CDMA requires a very large bandwidth in comparison to the baseband bandwidth of the data stream. Second, CDMA transmitters and receivers are more complex than those used in TDMA or FDMA. For instance, to successfully recover the data, the receiver has to correctly synchronize the spreading code with the transmitter. Moreover, if the received signal from an unintended user is much stronger than that of the intended user, the stronger signal will be dominant and can cause strong interference, making the detection of the intended signal severely degraded. This situation may occur when the unintended transmitter is much closer to the receiver than the intended one and it is commonly referred to as the "near/far" problem.

4.3.2 Access Control

Access control determines *how* a communications device (or user) can access a sub-channel, including sub-channel assignment and related resolution protocols. In particular, the participating communications devices must agree on some ways for allocating communication resources among themselves to be able to share the channel efficiently.

The resource allocation and accessing scheme can be controlled in a centralized manner by a single scheduler. In this setup, participating communication devices must report their demands to the scheduler and the scheduler will perform the assignments for all participating communications devices based on the available resources, and their demands, in consideration of other possible parameters such as device priority, quality-of-service (QoS) requirements,…

If a participating user has a *well-defined* and *fixed* periodic schedule for a long time, then a *fixed* allocation of communications resources in terms of sub-channels (in FDMA, TDMA or CDMA) can be done once in advance for the participating user to follow for a long time. This is called *fixed-assignment multiple access* scheme.

If a participating user has traffic that randomly occurs, then a *fixed* sub-channel assignment is not possible. In this case, sub-channel must be assigned on demand. The sub-channel demand-assignment (DA) can be done in *centralized* manner as follows.

A particular element of the communications system is selected to be a central scheduler for the sub-channel demand-assignment or the entire system.

Any participating user must report a demand to scheduler whenever it has traffic arrival to request for allocated communications resources. Based on the requests from all demanding users, and possibly other considerations such as priority, resource requirements, QoS requirements, …, the scheduler will prepare the sub-channel assignments and send them to all demanding users. During this *sub-channel setup phase*, the demanding users must wait for their assignments before accessing to their allocated sub-channels. It is worth noting that some *signalling* sub-channel or scheme must be established to support the demand-assignment exchanges.

The next *data transfer phase* is for users to access their allocated channels for transfer their data traffics.

At the end of its data transfer, the corresponding user needs to send to the scheduler a request to release its allocated sub-channels. This is the *sub-channel release* phase.

Such a so-called *demand-assignment multiple access* (DAMA) or *reservation-based multiple access* approach can provide good robustness and stability in control and hence is widely used in wire-line, wireless and satellite communications. For example, in a cellular system, the base-station takes care of the scheduling/resource allocation for all users in the cell.

As the available sub-channel is not used during the *sub-channel setup phase* and the final *sub-channel release* phase, the sub-channel utilization for a communications transaction in a DAMA scheme is reflected by the ratio of the duration of the *data transfer phase* to the total duration of the 3 phases. The sub-channel utilization gets better (higher) for relatively longer *data transfer phase*. Hence, DAMA is acceptable for long communications transactions such as voice/video communications, large data transfers, …

While DAMA with centralized control can provide good robustness and stability, its demand-assignment procedure can introduce noticeable and annoying delay, especially for short data transmissions. Alternatively, to reduce or eliminate the demand-assignment procedure, the resource allocation can also be conducted in a *distributed* manner where the participating devices follow some predefined procedures to determine the shared communication resources should be used by which device at any instance. In the following sub-sections, a few multiple-access schemes based on this alternative approach to reduce or eliminate the demand-assignment procedure in a *distributed* manner are discussed.

4.3.3 Random Access Techniques

In data communication, traffic demands are often bursty in nature with mixture of very short messages (e.g., service requests, commands) and large data transfers. An actual data transfer can be very short with possibly high-rate demands while most of the time the device stays idle. In such case, allocating dedicated communication resources to a user can become inefficient since during the user's idle period, the valuable channel resource is not utilized, hence wasted. This situation gets even worse for a large population of users. In such cases, it could be better to use a more dynamic approach to determine the use of channel resources in a time-sharing manner. This issue motivates the development of random access methods. In random access schemes, during its transmission, a device will use the entire channel bandwidth. Since the channel resources are not shared during this duration, if multiple devices attempt to use the channel at the same time, their signals will interfere with each other, resulting in a collision. Collided frames are not recoverable and must be retransmitted at a latter time. Random access schemes are normally executed in a distributed manner between the participating devices without the need for a centralized coordinator.

In this sub-section, ALOHA, Slotted-ALOHA and Carrier Sense Multiple Access (CSMA) schemes will be described to answer the questions of how a device can determine if it is its turn to use the channel, how to detect and recover from a collision when it happens and what is the performance of each scheme.

ALOHA (Pure ALOHA)

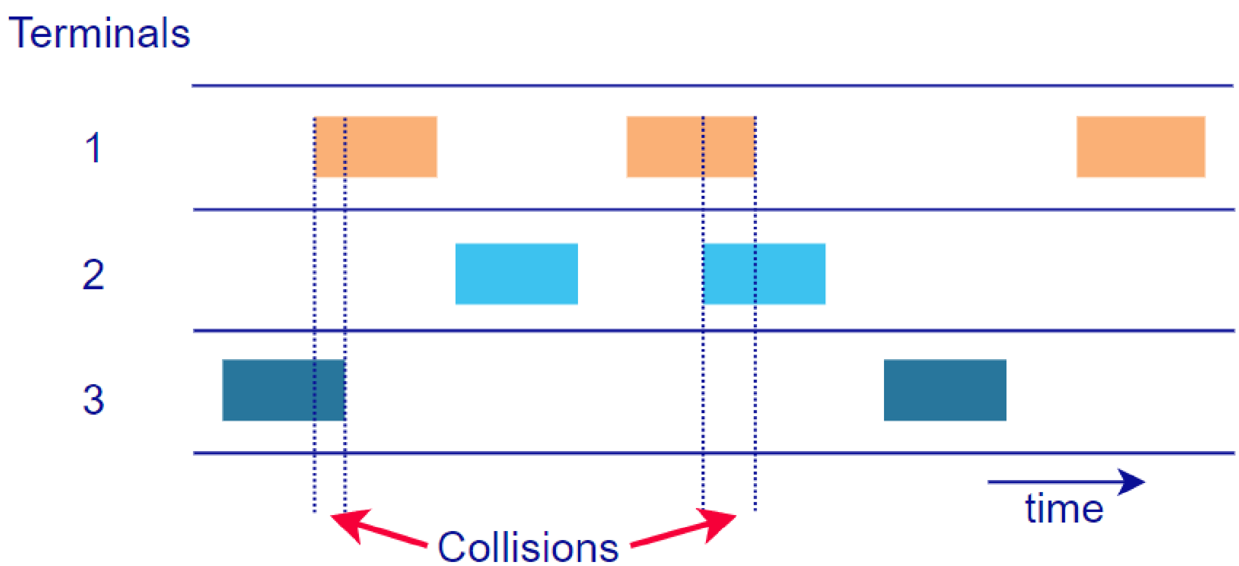


Figure 25: ALOHA and collisions in ALOHA.

ALOHA or equivalently pure ALOHA (pure ALOHA is used to differentiate the first scheme of ALOHA with its improved version, the slotted ALOHA) was a multiple access scheme developed and employed in Hawaii in the 1970s to connect the various terminals on remote islands of Hawaii to the central site at the University of Hawaii. At the time, there was no cable system to connect between these islands and building one is too expensive for the university to afford. In this context, the connection links were implemented using radio waves. In this scheme, all the remote terminals share the same frequency band for the uplink traffic to the central site and hence are prone to the issue of multiple access. The downlink traffic from the central site to the remote terminals was sent on a separated frequency band and is not subject to collisions.

In ALOHA, a terminal will start transmitting as soon as it has data in its buffer. If the periods of data transmission between two terminals overlaps, collision will occur. Figure 25 illustrates the operations of ALOHA in the case of 3 terminals and the collisions that occur in ALOHA. In order for the terminals to detect the collision, after each successfully data reception, the central computer will send an acknowledgement. As a result, after sending a data frame, a terminal will listen to the acknowledgement from the central computer, if there is none, it assumes that a collision occurred and the data must be retransmitted. However, if the terminal resends the data right after the collision, it will cause another collision since the other terminals that had their frames collided previously will also send their frames. As a result, after an occurrence of a collision, each sender will wait for a random delay before retransmitting the collided frame.

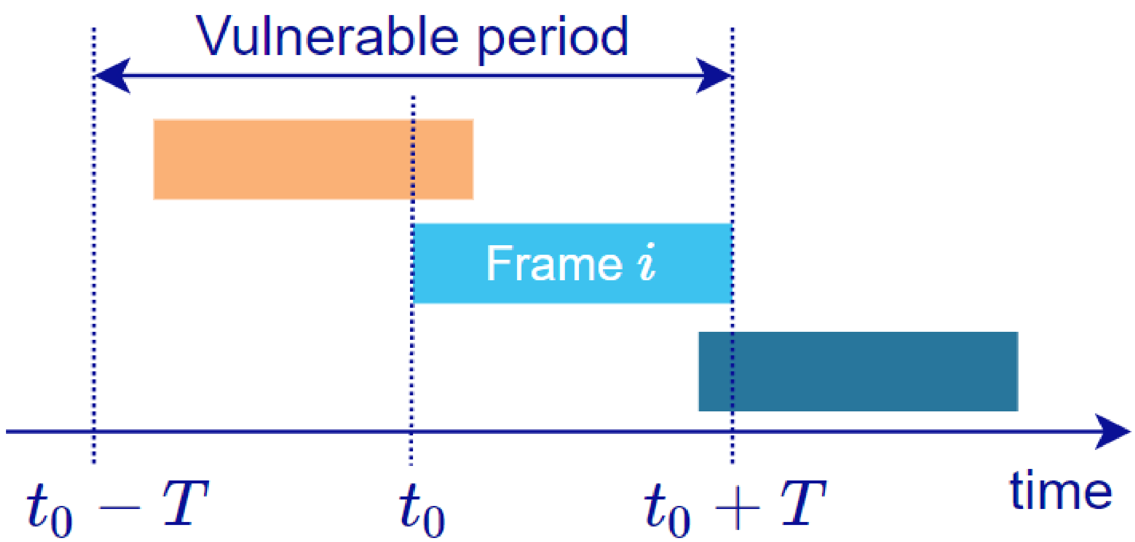


Figure 26: Vulnerable period in ALOHA.

Assuming that the packet length and the transmission rate of all the involved terminals are identical to each other. When frame i is transmitted at time t_0, it takes T seconds for the frame to be completely transmitted, T is called the frame time. Frame i will collide with any frame that was previously transmitted in a period of T seconds from its start, i.e. $[t_0 - T, t_0]$. In addition, any frame that is transmitted latter but within a period of T seconds from t_0 will also collide with it, i.e. $[t_0, t_0 + T]$. As a result, when a frame is transmitted, its vulnerable period for experiencing a collision is $2T$, from $t_0 - T$ to $t_0 + T$. The **vulnerable period** in which frame i may experience a collision with other frames before and after it is illustrated in Figure 26.

To be able to analyze the performance of ALOHA, let's further assume that the transmission attempts per frame time T follow a Poisson distribution with an average of G frames per frame time. This number of transmission attempts includes both the new frames and the retransmitted ones. In a vulnerable period $2T$ of ALOHA, the average transmission attempts would be $2G$ and the probability of k transmission occurs in this period is expressed in equation (4.14) following Poisson distribution with an average of $2G$.

$$\Pr[k \text{ transmission in } 2T] = \frac{(2G)^k e^{-(2G)}}{k!} \tag{4.14}$$

As a result, in order to have a collision-free transmission, we expect to have no transmission attempts in this vulnerable period. This event happens with the probability as expressed in equation (4.15).

$$\Pr[\text{collision} - \text{free}] = \Pr[0 \text{ transmission in } 2T] = e^{-(2G)} \tag{4.15}$$

Since the average transmission attempts in each frame time is G, the throughput S of ALOHA can be then calculated as in equation (4.16) as the multiplication between G and the probability of collision free transmission.

$$S = G \times \Pr[\text{collision} - \text{free}] = Ge^{-(2G)} \tag{4.16}$$

Equation (4.16) includes two parts that depend on G, one is linearly increased with G and one is exponentially decreased with G. When G is small, the linear term is dominant making S increases. However, as G grows bigger, the exponential term is dominant, making ALOHA throughput decreases. As a result, when the input frame rate increases from 0, we expect ALOHA throughput increases to a maximum value and then decreases. Practically, it can also be explained by the fact that as the traffic demand increases, collisions happen more frequently, decreasing the effective throughput.

To this end, it is expected that as G increases, there is a maximum throughput that is achievable under ALOHA scheme. By taking the derivative S' of S according to equation (4.16) and solve for its roots when $S' = 0$, one can easily derive that ALOHA maximum throughput is $1/2e \approx 0.18$, achievable when $G = 1/2$. As a result, the maximum channel utilization of ALOHA is 18%.

Slotted ALOHA

The 18% channel utilization of ALOHA is definitely not high and leave much for one to desire. From Figure 26, it is shown that one way to improve the channel utilization is by reducing the vulnerable period. In ALOHA, since terminals can send data anytime they want, the vulnerable period is $2T$; however, the actual transmission time of the frame is only T. If we can reduce the vulnerable period to only the frame time T, we can greatly improve the channel utilization. Motivated by this intuition, slotted ALOHA was proposed a short time after ALOHA.

In slotted ALOHA, transmissions from the terminals are conducted in discrete timeslots instead of in a send-at-will manner as in ALOHA. Each timeslot is assumed to support only one frame. The terminals synchronize with each other and send data frames only at the beginning of a slot. One way for synchronization is by using the downlink channel from the central computer to send a short tone at the beginning of a timeslot. A collision can occur when two or more terminals start their transmission in the same timeslot.

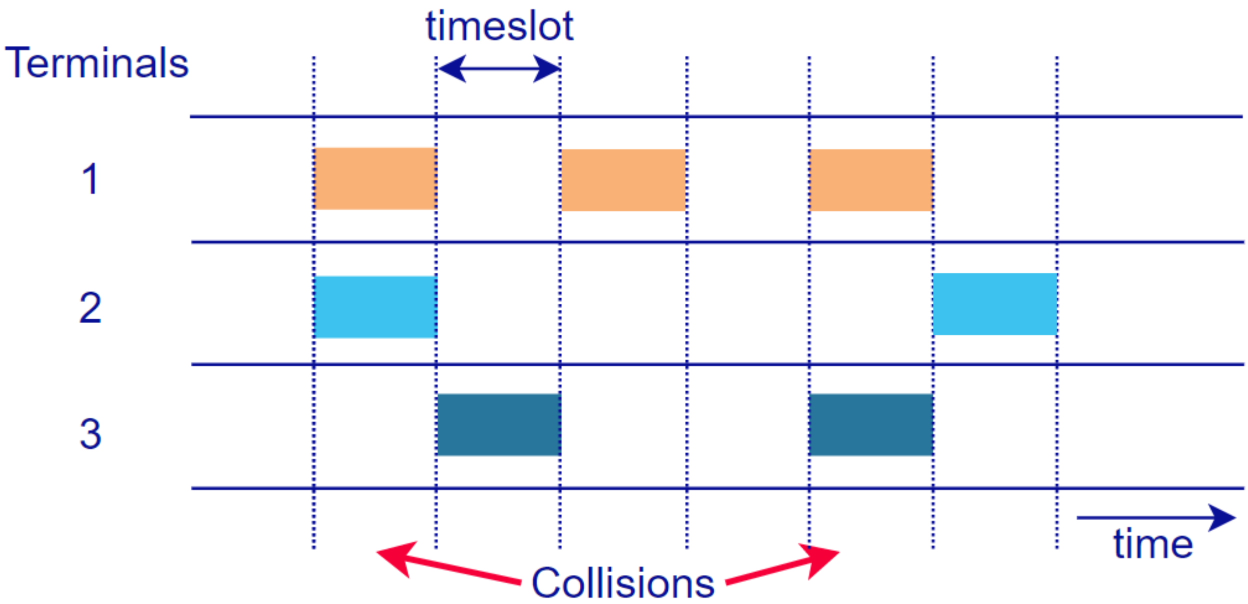


Figure 27: Slotted ALOHA and collisions in slotted ALOHA.

Figure 27 illustrates the transmissions in slotted ALOHA with 3 terminals and the collisions in this scheme. As the terminals can only transmitted at a beginning of a timeslot, it can be easily seen that the vulnerable period of slotted ALOHA reduces to only T.

Following the same assumptions and derivations as ALOHA, the average transmission attempts in one timeslot is G and hence, the probability of k transmission attempts in one timeslot is expressed as

$$\Pr[k \text{ transmission in } T] = \frac{(G)^k e^{-(G)}}{k!} \tag{4.17}$$

The probability of no collision is then

$$\Pr[\text{collision} - \text{free}] = \Pr[0 \text{ transmission in } T] = e^{-(G)} \tag{4.18}$$

And the effective throughput can be derived as

$$S = G \times \Pr[\text{collision} - \text{free}] = Ge^{-(G)} \tag{4.19}$$

By taking the derivative of (4.19), the maximum achievable throughput of slotted ALOHA can be calculated as $S = 1/e = 0.368$, or 36.8% channel utilization. This maximum throughput happens when $G = 1$. As a result, the maximum channel utilization of slotted ALOHA is doubled of that offered by ALOHA and this is achieved at a much higher input traffic load.

Figure 28 illustrates the throughputs of ALOHA and slotted ALOHA as the input traffic demand increases. It can be seen that as the input traffic increases from 0, the throughputs of both schemes increase, reaching their maximums and then decrease towards 0. It can also be shown that the maximum throughputs of ALOHA and slotted ALOHA are $1/2e$ and $1/e$ accordingly. Throughout the entire operational range of G, slotted ALOHA outperforms ALOHA by a large margin, and it can also withstand high traffic load better than ALOHA.

It is noted that the increase in throughput results from the reduce in vulnerability period.

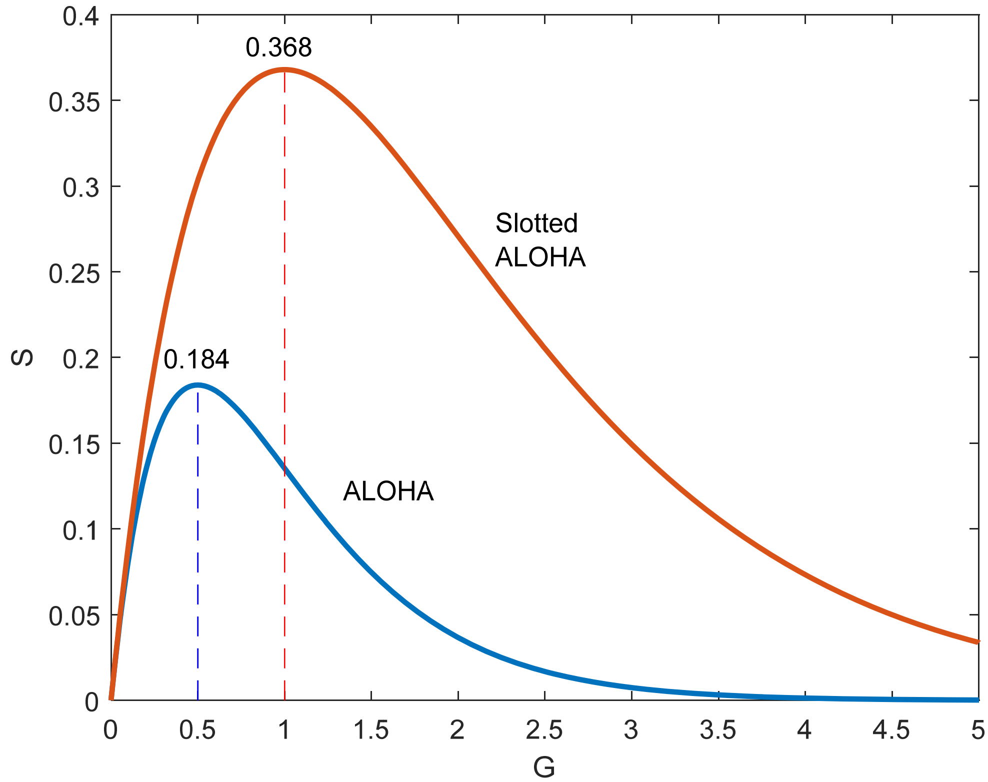


Figure 28: Throughput versus input load of ALOHA and Slotted ALOHA.

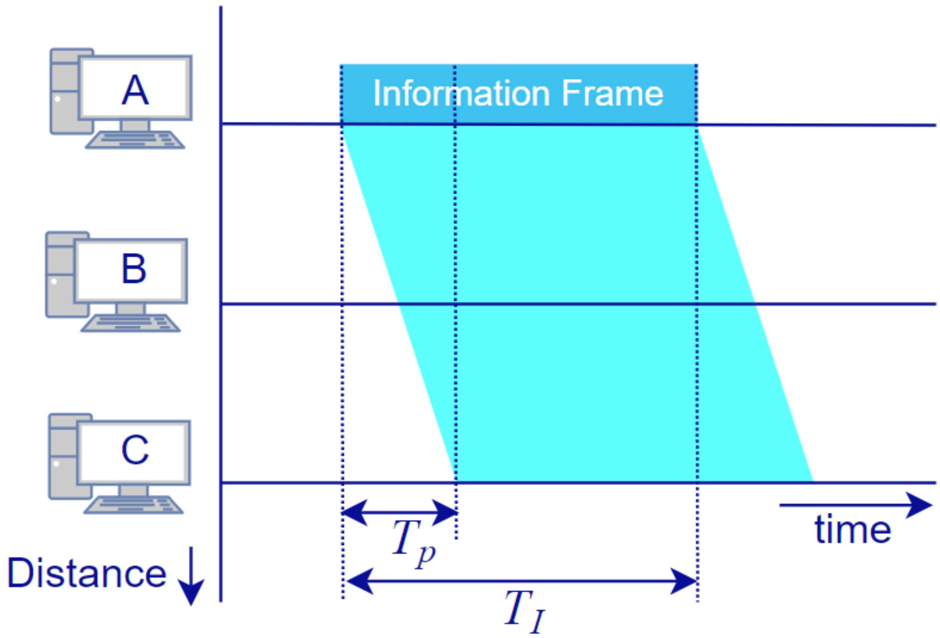
As an example, Slotted ALOHA is used as an access scheme in Radio-frequency identification (RFID) for many RFID tags.

Carrier Sense Multiple Access (CSMA)

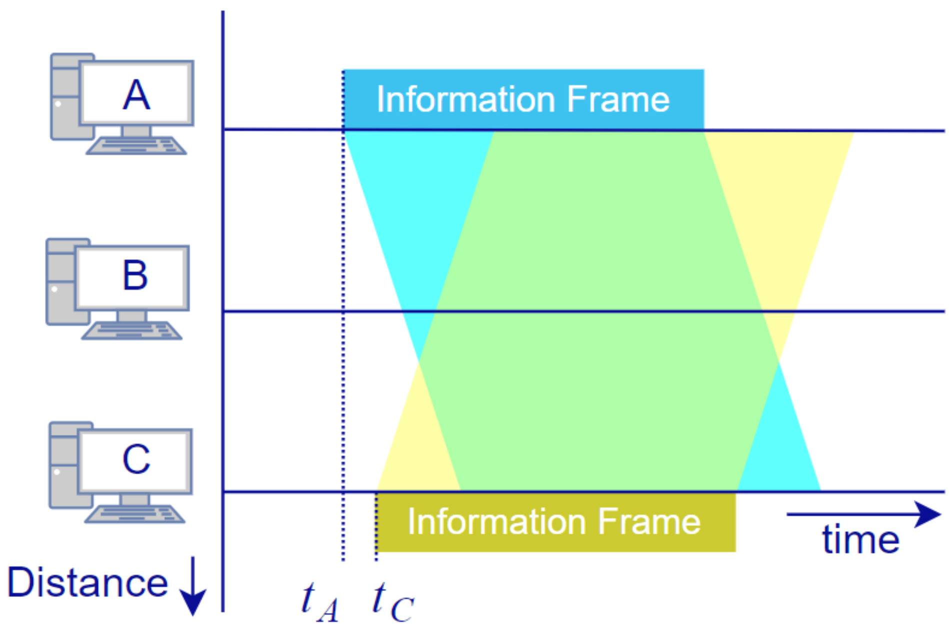
In ALOHA and slotted ALOHA, the terminals make their decisions to transmit their data independently from each other and regardless of what the others are doing. For instance, in ALOHA, when a terminal is in the process of transmitting data, if another terminal transmits data simultaneously, this transmission will only result in collision. However, being unaware of what the channel state, the ALOHA terminal may transmit anyway. Moreover, when a collision occurs, it does not worth to continue transmitting as the already transmitted portion of the frame was already collided and the entire frame needs to be retransmitted anyway.

All these wastages use of communication resources can be improved if the terminals are equipped with the ability to be aware of the current usage of the channel and act accordingly in a reasonable manner. This idea gives rise to the development of Carrier Sense Multiple Access schemes. The "Carrier Sense" part refers to the ability that the participated terminals can listen to the carriers or the current transmission ongoing on the channel. In CSMA, before transmitting on to the channel, the terminal should first listen to the current activities there. If there is any ongoing transmission, it is better not to transmit at that moment. Only in case the terminal senses that the channel is idle, it may start transmitting its data.

It is noted that while CSMA can potentially reduce the chance of collisions, it does not guarantee that no collision will happen. Collisions can still occur if two terminals both have data for transmission, they listen to the channel and wait until it is silent, then transmit simultaneously, causing a collision. Moreover, due to the physical distance between them, even when one terminal starts the transmission first, as long as the signal does not yet propagate to the other terminals, they still think that the channel is idle and can start their transmissions.



a) The propagation of signals through the transmission channel.



b) Collision in CSMA.

Figure 29: Propagation of signal through the transmission channel and collision in CSMA.

Figure 29 a) illustrates the propagation of an information frame through the channel. The channel consists of three terminals A, B and C; terminals A and C are located at the two ends of the transmission channel and B is in the middle of these two. Assuming at time t_0, A starts transmitting an information frame. The signals from A will propagate through the channel and will take a duration of T_p to propagate to the other end of the channel. Only by this time that all the terminals on the channel are aware of the transmission from A and will not interfere with it. However, it is possible that another terminal, such as C, starts transmitting before the first signal from A reaches it, hence causing a collision as depicted in Figure 29 b). As a result, the vulnerable period in CSMA is T_p.

Depending on the way a station starts its transmission when detecting that the channel is idle, CSMA is divided into three schemes. Firstly, the terminal can start transmitting as soon as it senses that the channel is idle, this scheme is called **1-persistent CSMA**. The "1" notation implies that the station will start transmitting with a probability of 1 after sensing that the channel is idle. As discussed in ALOHA scheme, start transmitting as soon as the channel is idle may not be a good idea as there is a chance that other terminals are also waiting for their transmissions, hence, causing another collision. Moreover, a transmission from any other terminal within T_p from a previous transmission will also result in a collision. As such, 1-persistent CSMA is the greediest scheme among the three and have the highest number of collisions.

The second CSMA scheme we investigate is **non-persistent CSMA**. In this scheme, when a terminal has data, it will sense the channel, if the channel is idle, it can start transmitting. However, if the channel is busy, the terminal will stop sensing the channel and wait for a random period of time before restarting the channel sensing. It can be seen that non-persistent CSMA is less greedy than 1-persistent CSMA, which results in less collision but at the cost of a longer delay.

The last CSMA scheme we study is **p-persistent CSMA**. In this scheme, when sensing the channel, if the channel is idle, the terminal will start transmitting with the probability of p and wait for another T_p with the probability of $(1-p)$. Even after this waiting period, the same procedure repeats, i.e. the terminal senses the channel, if it is idle, the terminal starts transmitting with the probability of p and wait for yet another T_p with the probability of $(1-p)$. If after the waiting period, the channel is busy, the terminal will wait for a random period of time before restarting this algorithm again, as if a collision occurs. This behaviour of waiting or transmitting based on probability makes it possible to distribute the transmissions of the participated terminals with time and hence reducing the chance of causing collision. However, it introduces a delay depending on p.

Regarding the performance of these CSMA schemes, the throughput derivations are quite lengthy and hence, are not included in this chapter. Interested users can refer to [6] for the detail derivations. As illustrated in Figure 29, the collision in CSMA schemes depends on the propagation duration T_p and in particular, the ratio between the propagation duration and the data transmission duration T_I, i.e. $a = T_p/T_I$. Figure 30 depicts the performance of 1-persistent CSMA scheme with different values of a. It can be seen that as a increases, it takes longer to transmit the signal from one end to another end of the channel, hence, the chance of causing collisions increases. This increase of collision in turns will decrease the throughput of CSMA since the collided frames need to be retransmitted. It can also be seen that when a increases, the maximum achievable throughput of CSMA may even fall behind that achieved by ALOHA and slotted ALOHA as analyzed earlier.

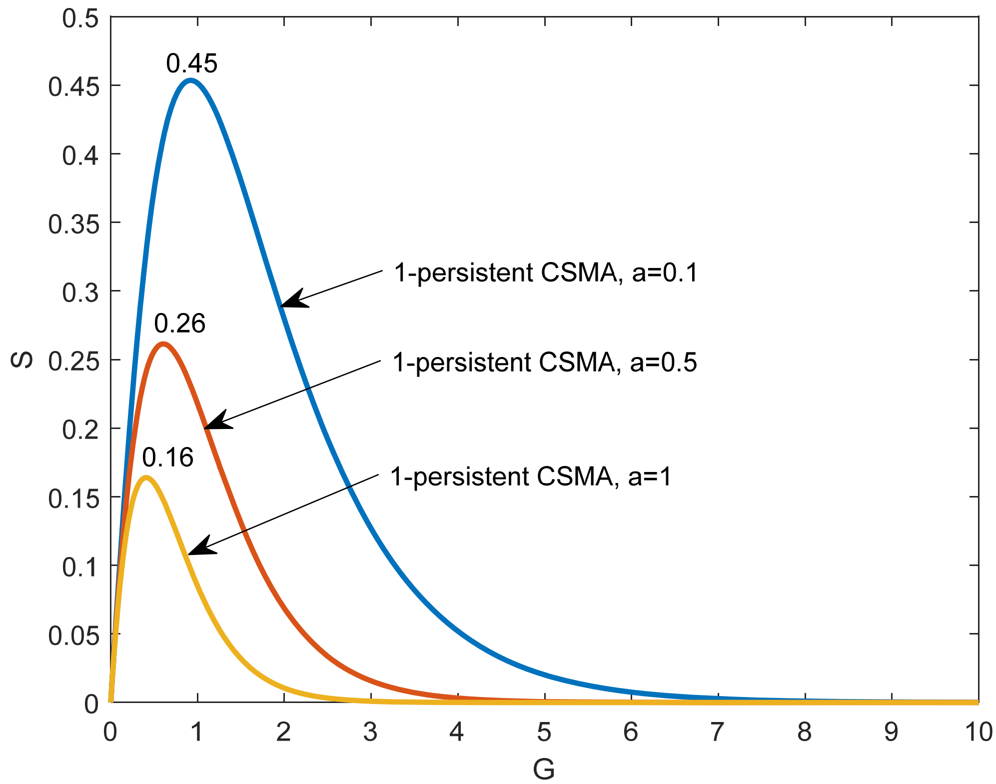


Figure 30: Throughput of 1-persisent CSMA with different values of a.

Figure 31 illustrates the throughput comparison between ALOHA, slotted ALOHA and the various schemes of CSMA including 1-persistent CSMA, non-persistent CSMA, and p-persistent CSMA with $p = 0.5$ and 0.01 when the input load increases. For all CSMA schemes, their performances are plotted in the ideal case when $a = 0$, meaning the propagation time is negligible and when one terminal starts transmitting, other terminals will be aware of its transmission immediately. It can be seen that in this case, the achievable throughput of all CSMA schemes outperforms that of ALOHA and slotted ALOHA by a wide margin. For instance, the peak throughput of 1-persistent CSMA is over 50% and that of p-persistent CSMA ($p = 0.01$) is approximately 100%. Among the three CSMA schemes, it is shown that 1-persistent CSMA has a lowest performance. It can be explained by the fact that in this scheme, the greedy approach of trying to transmit immediately when the channel is idle causes a lot of collisions especially when the load is high, i.e. more terminals have data to transmit. The high number of collisions then penalizes its throughput, hence, makes the throughput of 1-persistent CSMA falls off rapidly after reaching its maximum.

On the other hand, non-persistent CSMA approach of re-schedule the sensing whenever the channel is busy instead of continuously monitoring for idle gaps makes it a less greedy approach. As terminals only sense the channel randomly, the chance of simultaneous transmission reduces and the throughput of non-persistent CSMA continues to increase towards 100% even when the input load is very high. It is noted that this increase in throughput is the result of backlogging the data frames for a better chance of transmission; hence, it results in a very high delay.

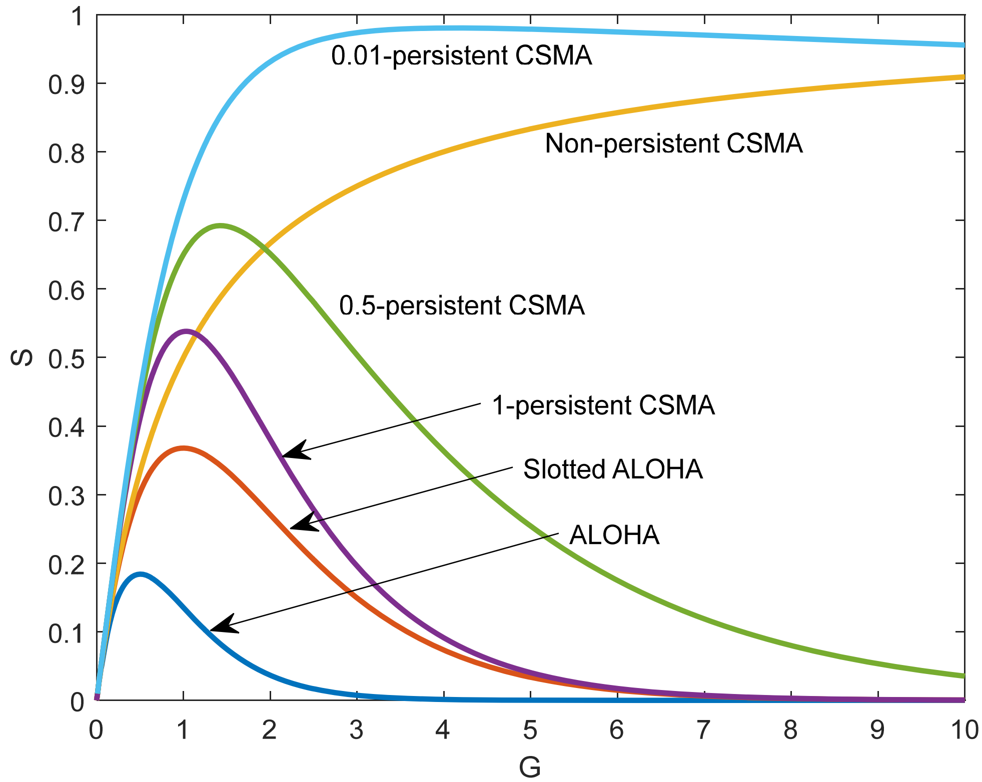


Figure 31: Throughput comparison between ALOHA, slotted ALOHA and CSMA schemes.

In the case of p-persistent CSMA, there is a parameter p that can be manipulated to compensate between the achievable throughput and delay. It is shown that in the extreme when $p \to 1$, the performance of p-persistent CSMA converges to that of 1-persistent CSMA. If we allow p to be very small, i.e. the terminal is likely to defer its transmission even when the channel is idle, decreasing the chance of collision, and making the throughput of p-persistent CSMA shoots up to approximately 100% at low load. When the input traffic load increases, the probability of collision increases even with the deferring of transmissions, making the achievable throughput decreases. Because the probability of collision depends on p, it is shown that when p decreases, the slope of the decrease in throughput is flatter.

Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

Being aware of the state of the transmission channel makes CSMA schemes more efficient than ALOHA and slotted ALOHA, especially when the transmission link is short. However, when collision occurs, participated CSMA terminals continues to transmit until the end of the frame. This extra transmission "tail" is a waste of communication resource as those bits cannot be recovered and the collided frames must be re-transmitted anyway. This wastage is very noticeable when the frame time is much larger than the transmission time ($a \ll 1$). Instead, if the collided terminals can stop their transmissions as soon as the collision happens, this wastage can be eliminated, hence increasing the achievable transmission efficiency.

This observation motivates the development of **Carrier Sense Multiple Access** with **Collision Detection (CSMA/CD)**. The "collision detection" part signifies the ability of the participated terminals to detect collisions. In CSMA/CD, during its transmission, a terminal keeps listening to the channel, if it senses that what it receives is not what it sent, it will declare that the transmitting frame experiences collision. In this case, it will stop transmitting immediately so that the channel resource is not wasted. The collided frame will be retransmitted latter. After detecting the collision, the terminal will wait for a random amount of time before sensing the channel for a retransmission. In CSMA/CD, the process of channel sensing and retransmitting can follow any of the previous CSMA schemes: 1-persistent CSMA, non-persistent CSMA or p-persistent CSMA.

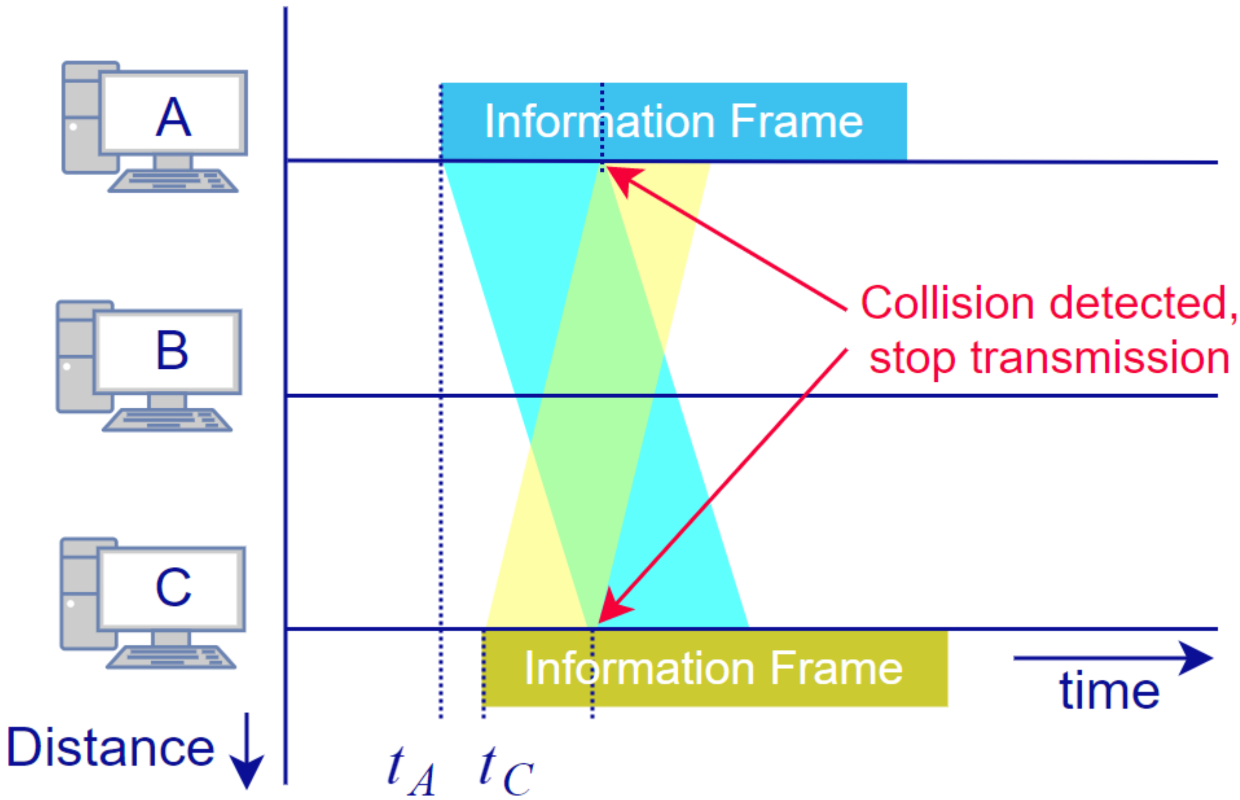


Figure 32: Collision detection in CSMA/CD.

Figure 32 illustrates the collision detection mechanism of CSMA/CD. During their transmissions, terminal A and C continuously listen for signals on the channel. If the transmitting terminals detect signals from other terminals, it will stop the transmission and hence, in comparison with Figure 29 b), a significant amount of transmission resources can be saved.

It is noted that if a terminal stops its data frame transmission before the signal from the other station reach it, it is unable to detect the collision and will falsely assume that the data frame was successfully transmitted. From Figure 32, it can be seen that the worst case scenario is in the case the two terminals that involve in the collision are located at the very two ends of the transmission channel and one starts transmitting just before the signal from the other terminal reaches it. As it takes T_p for a signal to propagate from one end to another, it takes another time duration of T_p for the first terminal to receive the signal from the second one and be able to detect the collision. As a result, it is required that that the length of a data frame must be long enough so that the transmission time is larger than a round-trip propagation duration as

$$T_I = \frac{L}{R} > 2T_p = 2 \times \frac{d}{v} \tag{4.20}$$

where L, R, d, v are the frame length, transmission rate, cable length and speed of signal propagation on the channel.

CSMA/CD is widely used in LAN protocols such as the Ethernet. In Ethernet, the random delay is determined by the **binary exponential back off algorithm**. This algorithm aims at adapting the random waiting time after a collision with the number of repeated collisions the terminal previously experienced. In particular, upon experiencing repeated collisions while transmitting a frame, with each additional collision, the terminal doubles the mean value of its random delay. For example, if n consecutive collisions already occur while a terminal tries to transmit a frame, it will wait for a random duration picking from the set of $\{0,1,\dots,2^n-1\}$ times the back off duration (defined depending on the protocol). Hence, as the number of collisions increases, the terminals that caused these collisions are likely to wait longer, reducing the probability of further collisions and giving the chance for the network to resolve the current severe contention on the channel. Depending on the implementation, after 10 to 15 consecutive unsuccessful attempts, the terminal will give up and report an error. It then depends on the upper protocols or application that the frame should be retried again or not.

4.3.3 Collision-free multiple access

In the previous section, random access schemes are very simple to implement, can operate in a fully distributed manner and can achieve relatively high channel utilization efficiency at low input traffic. However, as the traffic demand increases, random access schemes suffer from collisions which would severely penalize the achievable throughput. Moreover, random access schemes such as CSMA trade-offs its delay for high throughput, which may not be desirable in many cases. In this section, we will investigate some approaches that enable multiple access on a shared medium without causing collisions, including bit mapping and token passing.

Bit mapping

The first collision-free scheme that we study is bit mapping protocol. In this scheme, the transmission time is divided into epochs including two types of periods, the reservation period and the data transmission period. A **reservation period** comes at the beginning of each epoch so that each terminal can inform its demand for data transmission. Data transmission will occur in the data **transmission period**, in which each station will transmit according to the order and the reservations in the reservation period.

The reservation period is divided into multiple small reservation slots of one bit time and each terminal is assigned a dedicated reservation slot. If a terminal wants to transmit in the next transmission epoch, it broadcast its reservation bit in its assigned slot. Figure 33 illustrates the transmission epochs and their components in bit mapping protocol. Assuming that there are N terminals in the communication system. Each reservation period then consists of N reservation slots for the terminals to express their interests for transmission in the following data transmission period. For instance, in Figure 33, terminal 1, 3 and N express their interest in sending data; hence, in the following data transmission period, three frames will be transmitted from terminal 1, 3 and N, in that order, according to the reservation. Because the terminals already reserved their interests for sending data in the reservation period and transmissions will follow this reservation order, no collision will occur in data transmission period. Moreover, since the traffic demand may vary with time, a terminal may want to transmit in epoch k but not in epoch $k+1$, the length of the epoch may also vary with time. In a way, bit mapping operates similar to time division multiplexing but in a distributed manner and can adapt to the traffic demand without wasting timeslots when some terminals do not have data for transmission.

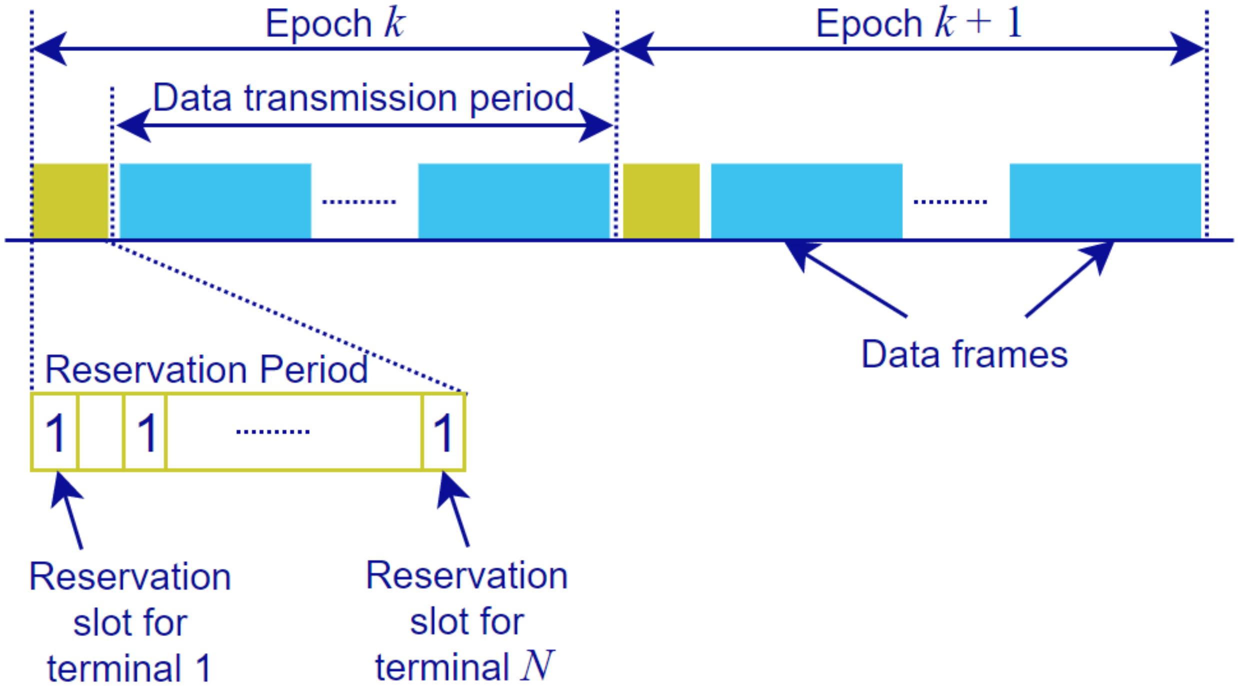


Figure 33: Components of a transmission epoch of bit mapping protocol.

Assuming that the frames are of fixed size of L bits, the number of terminals that involve in the transmission is M, the channel utilization for this epoch is expressed as

$$S = \frac{ML}{ML+N} = \frac{L}{L+\frac{N}{M}} \tag{4.21}$$

where ML is the number of data bits transmitted in the data transmission period and N is the number of bits in the reservation period.It is easy to see that this channel utilization is maximized when all the terminals want to send data in an epoch, i.e. $M = N$. In this case, the maximum channel utilization of bit mapping can be expressed as

$$S = \frac{L}{L+1} \tag{4.22}$$

Depending on the frame length, the channel utilization of bit mapping can be very high. For instance, if $L = 100$ bits, equation (4.22) yields $S = 99.01\%$, a much higher efficiency than ALOHA and most cases of CSMA. However, increasing the frame length will also lengthen the delay a terminal must wait before its transmission. It is noted that bit mapping scheme can be modified to allow more than one frame from one terminal to be transmitted in one epoch by adjusting the size of each reservation slot. However, it is noted that this scheme does not behave well in the case the number of terminals is unknown. Moreover, the delay that a terminal has to wait for its turn to transmit will also increase drastically with the number of terminals. This issue makes the scheme unsuitable for networks with a lot of terminals.

Token Ring

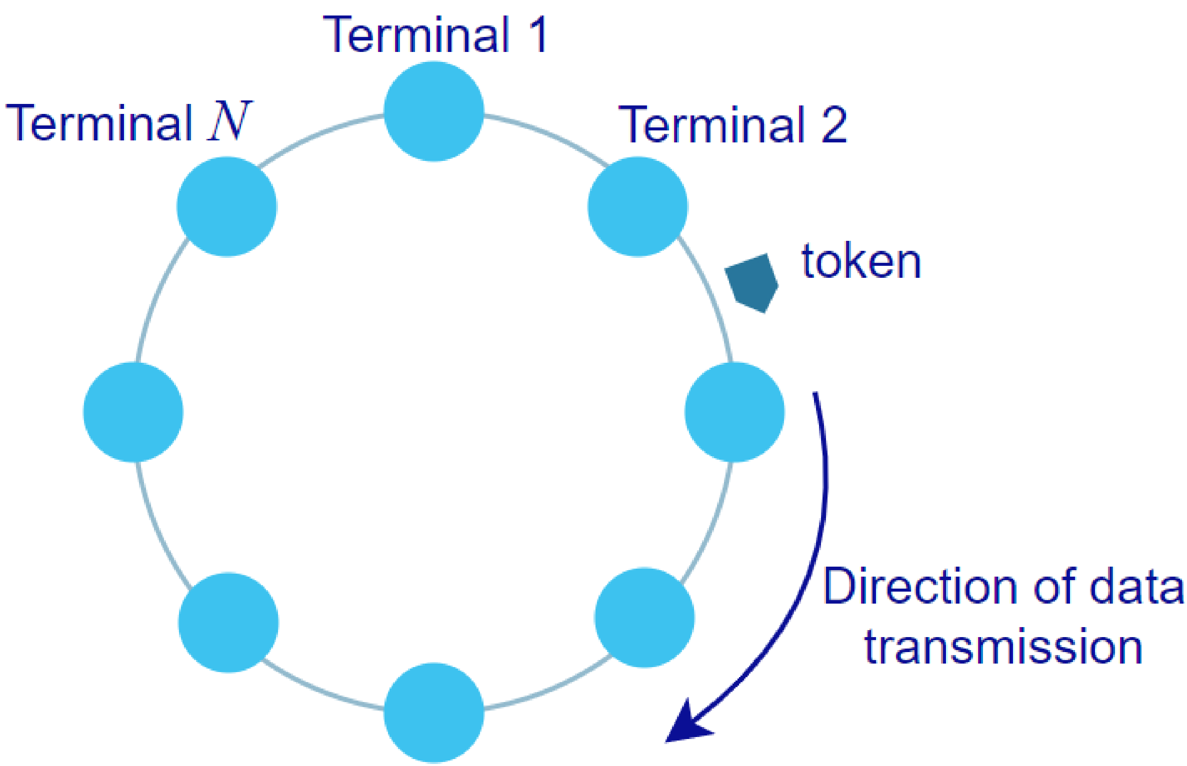


Figure 34: A Token Ring network.

The next collision-free mechanism that we will investigate is Token Ring. In this scheme, the participated terminals are connected together in a ring topology. As depicted in Figure 34. Essentially, each network card of a terminal consists of one transmission link to receive signals from the ring and one link to send signals to the ring. On a token ring, signals are transmitted in a predetermined direction and is passing from one terminal to another. In a token ring network, two of tokens are used: a free token and a busy token. These tokens are basically small messages with predefined bit patterns. A free token is used to allow a terminal to send data onto the ring. When no data is available for transmission, the free token is passing from one terminal to another around the ring.

When a terminal receives the free token and it wants to transmit data, it will keep the free token, then transmit the busy token, following by the data it wants to send. After sending the data, it puts the free token back to the ring to allow other station to send data. The busy token and the accompanied data must be removed from the ring by either the destination terminal or the source terminal depending on the implementation. One benefit that can be introduced, if they are removed by the source terminal, is that the source terminal can check this frame for error and can retransmit the frame if errors are found.

Since the signals are passing from one terminal to another, the time needed for a data frame of length L to travel one complete cycle around the ring, T, includes the propagation delay T_p and the time for each terminal to pass the frame, i.e.,

$$T = T_p + \frac{NL}{R} = T_p + NT_I \tag{4.23}$$

where $T_I = \frac{L}{R}$ is the transmission time of one frame.

Similar to bit mapping, the maximum channel efficiency is achieved when all terminals are transmitting data. In this case, the channel usage efficiency can be calculated as

$$S = \frac{NT_I}{NT_I + T_p} = \frac{1}{1 + \frac{T_p}{NT_I}} \tag{4.24}$$

From (4.24), it can be shown that the efficiency of token ring increases as the number of terminals increases. S also depends on the propagation delay and will decrease as the geographical size to the ring increases.

However, token ring is not without its issues. Firstly, the implementation of token ring is much more complicated than CSMA schemes. Moreover, the use of a single token for channel access may raise the fairness problem in which one terminal hold on to the token too long, preventing other terminals from transmitting their data. Thirdly, token is essentially only a pattern of bit that is known to the participated terminals and it can be contaminated by bit errors, which results in a token loss. Hence, an algorithm to monitor the passing of the token around the ring should be implemented to regenerate the token whenever it is no longer available. This algorithm must be operated in a distributed manner while making sure that multiple token should not be regenerated at the same time, which is not straightforward.

In addition, the ring topology is vulnerable to terminal or link failures. As depicted in Figure 34, any of these failures would break the ring topology apart and hence, disrupt the operations of the token ring network.

4.5 Conclusions

In this chapter, we studied approaches to improve the efficiency of a link by sharing the available bandwidth among many data streams. First, we looked at the issue of resource wastage when high bit rate communication links are used and the need of sharing communication resources to evade this problem. Then, we studied the basic principles of multiplexing and multiple access techniques and the differences between them to lay a foundation for the following detail discussions.

Next, focusing on the multiplexing techniques, we explored the mechanisms of TDM and FDM to divide the available communication resources in time and frequency domains. The possibilities and potentials of each scheme were considered through their many techniques. For instance, in FDM scheme, it was shown that by using orthogonal frequencies, we can achieve a very high spectral efficiency while being able to mitigate the effect of ISI at the same time. In each case, various applications were presented to illustrate how FDM and TDM can be used in practical systems. As examples, we examined ADSL Wavelength Division Multiplexing as the use cases of FDM, and SONET as a use case of TDM.

After understanding multiplexing techniques, we turned our attention to multiple access techniques to share a common medium (channel) among multiple users. We divided multiple access into 2 main mechanisms: channelization, and access control. In channelization, we discussed 3 basic schemes: FDMA (including OFDMA), TDMA, and CDMA. In access control, we discussed the centralized reservation techniques and various distributed schemes and their advantages/disadvantages along with examples. In the distributed approach, we examined random-access schemes: ALOHA, slotted ALOHA and CSMA with its various derivatives to provide a dynamic way for users to share their resources by competing with each other for transmission. As a consequence of this competition, collisions may occur when multiple users tried to capture the channel at the same time. To this end, we also investigated distributed collision-free multiple access schemes where contention can be resolved without causing collisions.

References

[1] ITU-T Recommendation G.922.5: Asymmetric Digital Subscriber Line 2 transceivers (ADSL2) – Extended bandwidth ADSL2 (ADSL2plus) https://www.itu.int/rec/T-REC-G.992.5/en

[2] IEEE P802.3ae 10Gbps Ethernet Task Force - http://www.ieee802.org/3/ae/index.html

[3] 2018 Ethernet Roadmap looks to Future Speeds of 1.6 Tbps - https://insidehpc.com/2018/03/new-2018-ethernet-roadmap-looks-future-speeds-1-6-terabits-s/

[4] ITU-T G. 694.2: Spectral Grids for WDM applications: CWDM wavelength grid - https://www.itu.int/rec/T-REC-G.694.2/en

[5] SONET homepage - http://www.sonet.com/

[6] L. Kleinrock, F. Tobagi, "Packet Switching in Radio Channels: Part I – Carrier Sense Multiple-Access Modes and their Throughput-Delay Characteristics", *IEEE Transactions on Communications*, Vol. 23 (12), pp. 1440-1416, 1975.

Review questions

R1. Why multiplexing and multiple-access are needed for communications? What are the basic differences between these two techniques?

R2. Briefly define OFDM and provide two benefits of OFDM.

R3. Compare the signal bandwidth before and after being encoded by the direct sequence spread spectrum (DSSS).

R4. Spread spectrum system is originally developed for military applications. Provide the motivation of these applications.

R5. Briefly compare FDMA, TDMA, and CDMA in terms of bandwidth for each user, cost, and guard band/time.

R6. Provide two advantages and two disadvantages of CDMA?

R7. What is the near-far problem? Describe the effect of the near-far problem in CDMA.

R8. What is the main difference between ALOHA and CSMA?

R9. What is the main difference between 1-persistent, non-persistent and p-persistent in CSMA?

R10. Briefly define CSMA/CD.

R11. Describe the hidden terminal problem in CSMA/CD. Explain how cellular networks resolve the hidden terminal problem.

R12. Between bit-mapping and token ring, which is more efficient to be implemented in a multiple access system having a large number of terminals densely deployed in a small area? Explain your answer.

R13. Briefly provide the differences between token ring and CSMA/CD.

Problems

PD1. Explain why broadcasting is preferred to multiplexing scheme in local area networks (LAN).

PD2. In a FDM multiplexing system between two offices, 5 data flows, each occupies 1.5MHz bandwidth, need to be supported. Assume that several guard bands whose length equals 10% of a signal bandwidth is used to avoid interference. Calculate the minimum bandwidth must be available in the trunk link between these two offices?

PD3. Consider a FDM system having a frequency band of 120 kHz, which is equally divided into 6 channels.

a. Calculate the bit rate of each channel in the case of no guard band. Assume that all channels achieve the same SNR level of 5 *dB*.
b. Suppose this FDM system additionally requires the guard band of 10% of the bandwidth of each channel. Calculate the bit rate of each channel.
c. When the entire frequency band is used as a single channel for TDM system with 6 users, calculate the bitrate of each user.

PD4. Consider a system having the bandwidth M is used to transmit and receive from a large number of users, say n users. Derive the bit rate provided for each user in the following two systems. Which system is more suitable in practice?

- A TDM system.
- A hybrid TDM-FDM system where the frequency band is equally divided into n channels. Each channel uses TDM transmission.

PD5. Suppose a multiple-access system in which multiple users transmit at different bit rates. Compare FDMA and TDMA when they are applied to this system.

PD6. Suppose we use a 64 kbps channel to transmit data. When the whole bandwidth is dedicated to for the data transmission, what is the minimum required signal-to-noise ratio (SNR) for a bandwidth efficiency of 1 bps/Hz? When the spread spectrum is used, find the channel bandwidth needed to achieve 64 kbps channel with SNR=0.1 and 0.01. From the obtained results, compare the spread spectrum approach to the ordinary (without spread spectrum) system in terms of the required SNR.

PD7. Compare the bit rate of signal before and after being encoded in CDMA.

PD8. In CDMA applications, Walsh codes can be modified to obtain another set of orthogonal codes. These codes consist of n codes having length n, represented as the rows of $n \times n$ matrix. The matrix is constructed recursively as

$$W_1 = [-1] \qquad\qquad W_{2n} = \begin{bmatrix} W_n & W_n \\ W_n & \overline{W_n} \end{bmatrix}$$

where the over-score denotes the logical NOT of the bits, and n is the matrix dimension. Construct the code matrices having dimension of *2* and *4*. Prove the orthogonality of these codes.

PD9. Replace 0 with -1 in the Walsh matrix and consider a CDMA system. Suppose that user 1 uses the Walsh code (-1 1 -1 1 -1 1 -1 1) and user 2 uses the code (-1 -1 1 1 -1 -1 1 1).

a. Suppose user 1 transmit bit -1 and user 2 transmits bit 1. Let the received powers from users 1 and 2 are equal, calculate the output of user 1 at the receiver.

b. Let the received power from user 2 is twice that of user 1. Suppose both users transmit bit 1. Calculate the output of user 1 at the receiver. Give a brief discussion on this result.

PD10. In a pure ALOHA system, it is observed that that a frame can be transmitted without collision with a probability of 0.2.

a. What is the average number of transmissions required to successfully transmit a frame through this system?
b. What is the throughput of this system?
c. If we want to improve the throughput of the system, should we increase or decrease the load? If each frame is 1000bits in length and a time slot is 0.1s, calculate the maximum achievable throughput of this system in terms of bps.
d. If a slotted ALOHA is used instead, should we increase or decrease the load to improve the throughput of the system? Calculate the maximum achievable throughput of this slotted ALOHA with the same assumptions as in c.

PD11. In a slotted ALOHA communication system, assume that the frame length is 1024bits and a 64kbps channel is shared by multiple users, calculate the maximum throughput of the system in frames per second.

PD12. For a multiple access system serving a small number of users, which of pure ALOHA or slotted ALOHA results a shorter delay? Explain your answer.

PD13. Compare the use of ALOHA and CSMA/CD in terms of efficiency in LAN and WAN communications? Explain and justify your answers with some numerical results, given that the maximum efficiency of ALOHA is 0.368 and that of CSMA/CD is given by $\frac{1}{1+6.4 \times \frac{TR}{L}}$, where *T, R,* and *L* denote the line propagation time, transmission line speed, and frame length, respectively.

PD14. Consider a slotted ALOHA system with *R* being the total transmission rate of frames. Derive the fraction of timeslot that is empty in the system in terms of *R*. When the system uses maximum transmission rate, calculate the fraction of timeslot that is empty. What is the meaning of obtained results in considered system?

PD15. Why is CSMA/CD impractical for WiFi but suitable for Ethernet?

PD16. Suppose that 100 terminals are connected to a hub in star topology (each terminal is dedicated a pair of line) and use token ring for the multiple access control. Specifically, this token ring system uses a *single-frame operation*. To transmit a frame, a terminal waits for a free token to arrive at its interface card. The terminal claims the token, removes it from the ring, and start transmitting its frame into the line. The frame then travels through the ring and across every interface card. The token is inserted after the last bit of the frame has returned to the sending terminal.

In this system, let assume that the frame length is 12,500 bytes, the distance from terminal to hub is 50 meters, transmission rate of line is 100 Mbps at speed 2.5×10^8 meters/second. Further, the single-frame operation results in 10 bit latency at each station, and the length of a free token are 3 bytes.

a. Calculate the effective transmission time of frame around the ring.
b. Suppose that the frame size is 10 times larger, calculate the maximum network throughput in this case.