CHAPTER 6

Bandwidth Utilization:

Solutions to Review Questions and Exercises

Review Questions

1. ***Multiplexing*** is the set of techniques that allows the simultaneous transmission of multiple signals across a single data link.
2. We discussed ***frequency-division multiplexing*** (**FDM**), ***wave-division multiplexing*** (**WDM**), and ***time-division multiplexing*** (**TDM**).
3. In ***multiplexing***, the word ***link*** refers to the physical path. The word ***channel*** refers to the portion of a link that carries a transmission between a given pair of lines. One link can have many (n) channels.
4. ***FDM*** and ***WDM*** are used to combine ***analog signals***; the bandwidth is shared. ***TDM*** is used to combine ***digital signals***; the time is shared.
5. To maximize the efficiency of their infrastructure, telephone companies have traditionally multiplexed analog signals from lower-bandwidth lines onto higher-bandwidth lines. The ***analog hierarchy*** uses voice channels (4 KHz), ***groups*** (48 KHz), ***supergroups*** (240 KHz), ***master groups*** (2.4 MHz), and ***jumbo groups*** (15.12 MHz).
6. To maximize the efficiency of their infrastructure, telephone companies have traditionally multiplexed digital signals from lower data rate lines onto higher data rate lines. The ***digital hierarchy*** uses ***DS-0*** (64 Kbps), ***DS-1*** (1.544 Mbps), ***DS-2*** (6.312 Mbps), ***DS-3*** (44.376 Mbps), and ***DS-4*** (274.176 Mbps).
7. ***WDM*** is common for multiplexing ***optical signals*** because it allows the multiplexing of signals with a very high frequency.
8. In ***multilevel TDM***, some lower-rate lines are combined to make a new line with the same data rate as the other lines. ***Multiple slot TDM***, on the other hand, uses multiple slots for higher data rate lines to make them compatible with the lower data rate line. ***Pulse stuffing TDM*** is used when the data rates of some lines are not an integral multiple of other lines.
9. In ***synchronous TDM***, each input has a reserved slot in the output frame. This can be inefficient if some input lines have no data to send. In ***statistical TDM***, slots are

dynamically allocated to improve bandwidth efficiency. Only when an input line has a slot's worth of data to send is it given a slot in the output frame.

10. In ***spread spectrum***, we spread the bandwidth of a signal into a larger bandwidth. Spread spectrum techniques add redundancy; they spread the original spectrum needed for each station. The expanded bandwidth allows the source to wrap its message in a protective envelope for a more secure transmission. We discussed ***frequency hopping spread spectrum*** (***FHSS***) and direct sequence spread spectrum (***DSSS***).
11. The ***frequency hopping spread spectrum*** (***FHSS***) technique uses *M* different carrier frequencies that are modulated by the source signal. At one moment, the signal modulates one carrier frequency; at the next moment, the signal modulates another carrier frequency.
12. The ***direct sequence spread spectrum*** (***DSSS***) technique expands the bandwidth of the original signal. It replaces each data bit with *n* bits using a spreading code.

Exercises

13. To multiplex 10 voice channels, we need nine guard bands. The required bandwidth is then B = (4 KHz) × 10 + (500 Hz) × 9 **= 44.5 KHz**
14. The bandwidth allocated to each voice channel is 20 KHz / 100 = 200 Hz. As we saw in the previous chapters, each digitized voice channel has a data rate of 64 Kbps (8000 sample × 8 bit/sample). This means that our modulation technique uses 64,000/200 = 320 bits/Hz.
15.
    a. Group level: overhead = 48 KHz − (12 × 4 KHz) = **0 Hz**.
    b. Supergroup level: overhead = 240 KHz − (5 × 48 KHz) **= 0 Hz**.
    c. Master group: overhead = 2520 KHz − (10 × 240 KHz) = **120 KHz**.
    d. Jumbo Group: overhead = 16.984 MHz − (6 × 2.52 MHz) = **1.864 MHz**.
16.
    a. Each output frame carries 1 bit from each source plus one extra bit for synchronization. Frame size = 20 × 1 + 1 = **21 bits.**
    b. Each frame carries 1 bit from each source. Frame rate = **100,000 frames/s.**
    c. Frame duration = 1 /(frame rate) = 1 /100,000 = **10 μs**.
    d. Data rate = (100,000 frames/s) × (21 bits/frame) = **2.1 Mbps**
    e. In each frame 20 bits out of 21 are useful. Efficiency = 20/21= **95%**
17.
    a. Each output frame carries 2 bits from each source plus one extra bit for synchronization. Frame size = 20 × 2 + 1 = **41 bits.**
    b. Each frame carries 2 bit from each source. Frame rate = 100,000/2 **= 50,000 frames/s.**
    c. Frame duration = 1 /(frame rate) = 1 /50,000 = **20 μs**.
    d. Data rate = (50,000 frames/s) × (41 bits/frame) = **2.05 Mbps.** The output data rate here is slightly less than the one in Exercise 16.

e. In each frame 40 bits out of 41 are useful. Efficiency = 40/41= **97.5%**. Efficiency is better than the one in Exercise 16.

18.

a. Frame size = 6 × (8 + 4) = **72 bits**.

b. We can assume that we have only 6 input lines. Each frame needs to carry one character from each of these lines. This means that the frame rate is **500 frames/s**.

c. Frame duration = 1 /(frame rate) = 1 /500 = **2 ms**.

d. Data rate = (500 frames/s) × (72 bits/frame) = **36 kbps**.

19. We combine six 200-kbps sources into three 400-kbps. Now we have seven 400-kbps channel.

a. Each output frame carries 1 bit from each of the seven 400-kbps line. Frame size = 7 × 1 = **7 bits.**

b. Each frame carries 1 bit from each 400-kbps source. Frame rate = **400,000 frames/s.**

c. Frame duration = 1 /(frame rate) = 1 /400,000 = **2.5 μs**.

d. Output data rate = (400,000 frames/s) × (7 bits/frame) = **2.8 Mbps.** We can also calculate the output data rate as the sum of input data rate because there is no synchronizing bits. Output data rate = 6 × 200 + 4 × 400 = **2.8 Mbps**.

20.

a. The frame carries 4 bits from each of the first two sources and 3 bits from each of the second two sources. Frame size = 4 × 2 + 3 × 2 = **14 bits.**

b. Each frame carries 4 bit from each 200-kbps source or 3 bits from each 150 kbps. Frame rate = 200,000 / 4 = 150,000 /3 = **50,000 frames/s.**

c. Frame duration = 1 /(frame rate) = 1 /50,000 = **20 μs**.

d. Output data rate = (50,000 frames/s) × (14 bits/frame) = **700 kbps.** We can also calculate the output data rate as the sum of input data rates because there are no synchronization bits. Output data rate = 2 × 200 + 2 × 150 = 700 kbps.

21. We need to add extra bits to the second source to make both rates = 190 kbps. Now we have two sources, each of 190 Kbps.

a. The frame carries 1 bit from each source. Frame size = 1 + 1 = **2 bits.**

b. Each frame carries 1 bit from each 190-kbps source. Frame rate = **190,000 frames/s.**

c. Frame duration = 1 /(frame rate) = 1 /190,000 = **5.3 μs**.

d. Output data rate = (190,000 frames/s) × (2 bits/frame) = **380 kbps.** Here the output bit rate is greater than the sum of the input rates (370 kbps) because of extra bits added to the second source.

22.

a. T-1 line sends 8000 frames/s. Frame duration = 1/8000 = **125 μs**.

b. Each frame carries one extra bit. Overhead = 8000 × 1 = **8 kbps**

23. See Figure 6.1.

24. See Figure 6.2.

Figure 6.1 *Solution to Exercise 23*

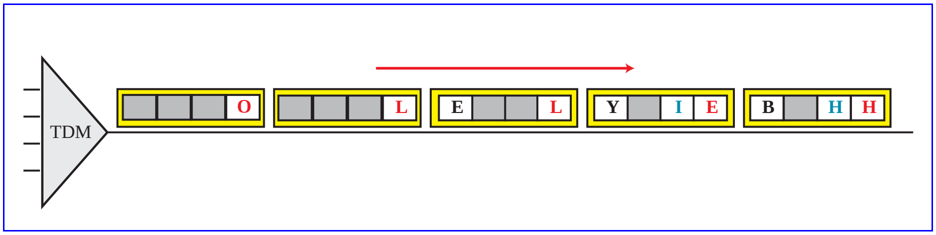
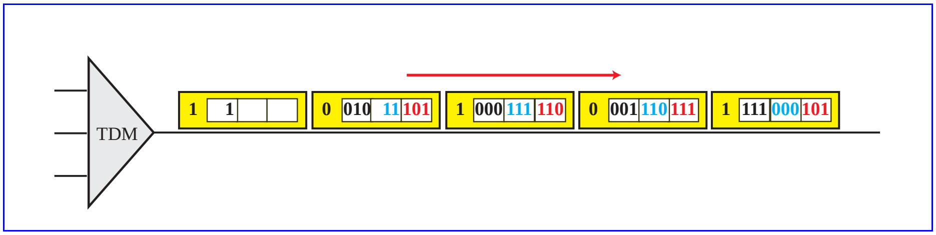
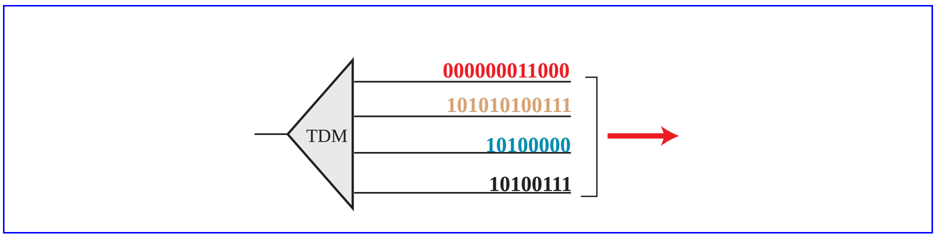


Figure 6.2 *Solution to Exercise 24*



25. See Figure 6.3.

Figure 6.3 *Solution to Exercise 25*



26.

a. DS-1 overhead = 1.544 Mbps – (24 × 64 kbps) = **8 kbps**.

b. DS-2 overhead = 6.312 Mbps – (4 × 1.544 Mbps) = **136 kbps**.

c. DS-3 overhead = 44.376 Mbps – (7 × 6.312 Mbps) = **192 kbps**.

d. DS-4 overhead = 274.176 Mbps – (6 × 44.376 Mbps) = **7.92 Mbps**.

27. The number of hops = 100 KHz/4 KHz = 25. So we need $\log_2 25 = 4.64 \approx$ **5 bits**

28.

a. 2^4 = **16 hops**

b. (64 bits/s) / 4 bits = **16 cycles**

29. Random numbers are 11, 13, 10, 6, 12, 3, 8, 9 as calculated below:

N_1	=	**11**
$N_2 = (5 + 7 \times \mathbf{11}) \bmod 17 - 1$	=	**13**
$N_3 = (5 + 7 \times \mathbf{13}) \bmod 17 - 1$	=	**10**
$N_4 = (5 + 7 \times \mathbf{10}) \bmod 17 - 1$	=	**6**

$$N_5 = (5 + 7 \times \mathbf{6}) \bmod 17 - 1 = \mathbf{12}$$
$$N_6 = (5 + 7 \times \mathbf{12}) \bmod 17 - 1 = \mathbf{3}$$
$$N_7 = (5 + 7 \times \mathbf{3}) \bmod 17 - 1 = \mathbf{8}$$
$$N_8 = (5 + 7 \times \mathbf{8}) \bmod 17 - 1 = \mathbf{9}$$

30. The Barker chip is 11 bits, which means that it increases the bit rate 11 times. A voice channel of 64 kbps needs 11 × 64 kbps = 704 kbps. This means that the bandpass channel can carry (10 Mbps) / (704 kbps) or approximately **14 channels**.