Chapter 2

3.

To find the maximum data rate of a 6 Mhz noiseless channel using four-level digital signals, we use Nyquist's theorem: maximum data rate = $2 \cdot H \cdot \log_2(V)$ bits/sec, where H is the bandwidth and V is the number of discrete signal levels. So the maximum data rate is $2 \cdot 6 \cdot \log_2(4) = 12 \cdot 2 = 24$ Mbps.

11.

Radio antennas work best when the diameter of the antenna is equal to the wavelength of the radio wave. To calculate the frequency of the radio wave from the wavelength, we use the relation $\lambda f = c$, where λ denotes the wavelength, f denotes the frequency, and c denotes the speed of light. If the diameters of the radio antennas range from 1 cm to 5 meters, the frequency range covers 60 Mhz to 30GHz:

$$f = \frac{c}{\lambda} = \frac{3 \times 10^{8}\, meters/sec}{10^{-2}\, meters} = 3 \times 10^{10}/sec = 30\, GHz$$

$$f = \frac{c}{\lambda} = \frac{3 \times 10^{8}\, meters/sec}{5\, meters} = 6 \times 10^{7}/sec = 60\, MHz$$

22.

A modem constellation with data points at coordinates (1, 1), (1, -1), (-1, 1), and (-1, -1) is Quadrature Phase Shift Keying (QPSK), which can be used to transmit 2 bits per symbol. A modem at 1200 baud can send 1200 symbols per second. Using QPSK, a 1200-baud modem can achieve (1200 symbols/sec)(2 bits/symbol) = 2400 bps.

53.

CDMA chip sequences:

```
station A: (-1 -1 -1 +1 +1 -1 +1 +1)
station B: (-1 -1 +1 -1 +1 +1 +1 -1)
station C: (-1 +1 -1 +1 +1 +1 -1 -1)
station D: (-1 +1 -1 -1 -1 -1 +1 -1)
```

A CDMA receiver gets chips: `(-1 +1 -3 +1 -1 -3 +1 +1)`

```
receiver A: (-1 +1 -3 +1 -1 -3 +1 +1)•(-1 -1 -1 +1 +1 -1 +1 +1)
     = (1-1+3+1-1+3+1+1)/8 = 8/8 = 1
receiver B: (-1 +1 -3 +1 -1 -3 +1 +1)•(-1 -1 +1 -1 +1 +1 +1 -1)
     = (1-1-3-1-1-3+1-1)/8 = -8/8 = -1
receiver C: (-1 +1 -3 +1 -1 -3 +1 +1)•(-1 +1 -1 +1 +1 +1 -1 -1)
     = (1+1+3+1-1-3-1-1)/8 = 0/8 = 0
receiver D: (-1 +1 -3 +1 -1 -3 +1 +1)•(-1 +1 -1 -1 -1 -1 +1 -1)
     = (1+1+3-1+1+3+1-1)/8 = 8/8 = 1
```

A value of 1 corresponds to a 1 bit. A value of -1 corresponds to a 0 bit. A value of 0 means no transmission. So station A sent a 1 bit. Station B sent a 0 bit. Station D sent a 1 bit. Station C did not transmit.

Chapter 3

13.

To use horizontal and vertical parity bits for error detection in a block of bits, we consider the method described in Question 12. In particular, one parity bit is added to each row or column of data. The lower-right corner is a parity bit that checks its row and its column. Assume for the problem, the data bits plus the parity bits make up the *n*-row, *k*-column block. Within each row or column, an error can be detected if the number of bit inversions is odd. Or equivalently, an error goes undetected if the number of bit inversions in a row or a column is even. So a transmission error with exactly 4 bit inversions is undetected if every row or column contains an even number of bit inversions. The only way this can occur is if there are exactly two rows, each with exactly two bit inversions, and there are exactly two columns, each with exactly two bit inversions. In other words, if one bit inversion is located at (row r_1, column c_1) and another is at (row r_2, column c_2), then the other two must be at (row r_1, column c_2) and (row r_2, column c_1). Assume that each pattern of 4 bit inversions is equally likely to occur. Then the number of ways any 4 bit inversions can occur in an $n \times k$ block is $\binom{n \cdot k}{4}$. By the argument above, the number of ways in which an error with 4 bit inversions is undetected is $\binom{n}{2} \cdot \binom{k}{2}$. There is an issue if a bit inversion occurs at the lower-right corner. In that case, the bottommost row plus the rightmost column together contain 3 bit inversions, which make the error detectable. So we exclude those occurrences. There are $(n-1)(k-1)$ of those. Therefore, the probability that the error with exactly 4 bit inversions is

$$\frac{\binom{n}{2} \cdot \binom{k}{2} - (n-1) \cdot (k-1)}{\binom{n \cdot k}{4}}.$$

Simulator Experiments

1.

Input:

```
10000
Machine1
The quick brown fox
.56
2000
1000
Machine2
jumped over the lazy dog
.01
1000
500
```

Output:

```
Machine1's DataLink Layer sent frame: {0, 1, T}
Machine2's DataLink Layer sent frame: {0, 1, j}
Machine2's DataLink Layer Received frame: {0, 1, T}
Machine2's Network Layer Received: 'T'
Machine2's DataLink Layer sent frame: {0, 0, j}
Machine1's DataLink Layer Received frame: {0, 0, j}
Machine1's Network Layer Received: 'j'
Machine1's DataLink Layer sent frame: {1, 0, h}
Machine2's DataLink Layer Received frame: {1, 0, h}
Machine2's Network Layer Received: 'h'
Machine2's DataLink Layer sent frame: {1, 1, u}
Machine1's DataLink Layer Received frame: {1, 1, u}
Machine1's Network Layer Received: 'u'
Machine1's DataLink Layer sent frame: {0, 1, e}
Machine2's DataLink Layer Received frame: {0, 1, e}
Machine2's Network Layer Received: 'e'
Machine2's DataLink Layer sent frame: {0, 0, m}
Machine1's DataLink Layer Received frame: {-1, -1, m}
Machine2's DataLink Layer sent frame: {0, 0, m}
Machine1's DataLink Layer sent frame: {0, 1, e}
Machine2's DataLink Layer sent frame: {0, 0, m}
Machine1's DataLink Layer Received frame: {-1, -1, m}
Machine2's DataLink Layer sent frame: {0, 0, m}
Machine1's DataLink Layer sent frame: {0, 1, e}
Machine2's DataLink Layer sent frame: {0, 0, m}
Machine1's DataLink Layer Received frame: {0, 0, m}
Machine1's Network Layer Received: 'm'
Machine1's DataLink Layer sent frame: {1, 0,  }
Machine2's DataLink Layer Received frame: {1, 0,  }
Machine2's Network Layer Received: ' '
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine1's DataLink Layer Received frame: {-1, -1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine1's DataLink Layer sent frame: {1, 0,  }
Machine2's DataLink Layer sent frame: {1, 1, p}
```

Here we see data being exchanged between the two machines. When an error occurs (e.g. {-1, -1 , m}), the machines keep trying until a good frame is received.

2.

Input:

```
10000
Machine1
The quick brown fox
.99
```

```
2000
1000
Machine2
jumped over the lazy dog
.0
1000
500
```

Output:

```
Machine1's DataLink Layer sent frame: {0, 1, T}
Machine2's DataLink Layer sent frame: {0, 1, j}
Machine2's DataLink Layer Received frame: {0, 1, T}
Machine2's Network Layer Received: 'T'
Machine2's DataLink Layer sent frame: {0, 0, j}
Machine1's DataLink Layer Received frame: {-1, -1, j}
Machine1's DataLink Layer sent frame: {0, 1, T}
Machine2's DataLink Layer Received frame: {0, 1, T}
Machine1's DataLink Layer sent frame: {0, 1, T}
Machine2's DataLink Layer Received frame: {0, 1, T}
Machine1's DataLink Layer sent frame: {0, 1, T}
Machine2's DataLink Layer Received frame: {0, 1, T}
Machine1's DataLink Layer sent frame: {0, 1, T}
Machine2's DataLink Layer Received frame: {0, 1, T}
```

Here Machine 1 almost never gets a good packet. Machine 1 keeps sending the same frame to Machine 2. The channel is tied up by Machine 1 so Machine 2 cannot send anything.

3.

Input:

```
10000
Machine1
The quick brown fox
.0
2000
1000
Machine2
jumped over the lazy dog
.0
200
100
```

Output:

```
Machine1's DataLink Layer sent frame: {0, 1, T}
Machine2's DataLink Layer sent frame: {0, 1, j}
Machine2's DataLink Layer Received frame: {0, 1, T}
Machine2's Network Layer Received: 'T'
Machine2's DataLink Layer sent frame: {0, 0, j}
```

```
Machine2's DataLink Layer sent frame: {0, 0, j}
Machine2's DataLink Layer sent frame: {0, 0, j}
Machine1's DataLink Layer Received frame: {0, 0, j}
Machine1's Network Layer Received: 'j'
Machine1's DataLink Layer sent frame: {1, 0, h}
Machine2's DataLink Layer Received frame: {1, 0, h}
Machine2's Network Layer Received: 'h'
Machine2's DataLink Layer sent frame: {1, 1, u}
Machine2's DataLink Layer sent frame: {1, 1, u}
Machine2's DataLink Layer sent frame: {1, 1, u}
Machine2's DataLink Layer sent frame: {1, 1, u}
Machine2's DataLink Layer sent frame: {1, 1, u}
Machine1's DataLink Layer Received frame: {1, 1, u}
Machine1's Network Layer Received: 'u'
Machine1's DataLink Layer sent frame: {0, 1, e}
Machine2's DataLink Layer Received frame: {0, 1, e}
Machine2's Network Layer Received: 'e'
Machine2's DataLink Layer sent frame: {0, 0, m}
Machine2's DataLink Layer sent frame: {0, 0, m}
Machine2's DataLink Layer sent frame: {0, 0, m}
Machine2's DataLink Layer sent frame: {0, 0, m}
Machine2's DataLink Layer sent frame: {0, 0, m}
Machine1's DataLink Layer Received frame: {0, 0, m}
Machine1's Network Layer Received: 'm'
Machine1's DataLink Layer sent frame: {1, 0,  }
Machine2's DataLink Layer Received frame: {1, 0,  }
Machine2's Network Layer Received: ' '
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine1's DataLink Layer Received frame: {1, 1, p}
Machine1's Network Layer Received: 'p'
Machine1's DataLink Layer sent frame: {0, 1, q}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
```

```
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine1's DataLink Layer sent frame: {0, 1, q}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine2's DataLink Layer sent frame: {1, 1, p}
Machine1's DataLink Layer sent frame: {0, 1, q}
Machine2's DataLink Layer Received frame: {0, 1, q}
Machine2's Network Layer Received: 'q'
Machine2's DataLink Layer sent frame: {0, 0, e}
Machine2's DataLink Layer sent frame: {0, 0, e}
Machine2's DataLink Layer sent frame: {0, 0, e}
Machine2's DataLink Layer sent frame: {0, 0, e}
Machine2's DataLink Layer sent frame: {0, 0, e}
```

Here Machine 2 is much faster than Machine 1. We see Machine 2 sends many packets because Machine 1 cannot respond in time.