SYSC 4602 Assignment 2

Jessica Morris 100882290

September 30th, 2016

1. OC-1 speed is $51.84\,\frac{\text{Mbit}}{\text{s}}$, where one bit lasts $1.929 \times 10^{-8}\,\text{s}$. The time for the clock to drift by 1 bit is given by:

$$t_{drift} = \frac{1.929 \times 10^{-8}\,\text{s}}{1\,\frac{\text{ns}}{\text{s}}}$$
$$t_{drift} = 19.29\,\text{s}$$

   Therefore, it takes a SONET clock 19.29 seconds to drift by 1 bit at OC-1 speed. This suggests that SONET clocks should be resynchronized every 19.29 s, but resynchronizing at half of that time should be the maximum.

2. Consider that the delay for a circuit-switched network is due to the setup time, s, the propagation delay, kd, and and the transmission time, $\frac{x}{b}$. Determining the total delay for the circuit-switched network:

$$d_{total} = s + kd + \frac{x}{b}$$

   The delay for a packet-switched network, on the other hand, is determined by the propagation delay, kd, and the transmission time per hop. The transmission time per hop is given by the time for the first packet to fully transmit from source to destination, x bits over b bits per second, plus the time taken for p packets to complete $k-1$ hops over b bits per second. So, the total delay for the packet-switched network is given by:

$$d_{total} = kd + \frac{x}{b} + \frac{(k-1)p}{b}$$

   Since the two delays have kd and $\frac{x}{b}$ in common, the packet-switched network will have a smaller delay than the circuit-switched network when $s > \frac{(k-1)p}{b}$.

3. To determine which stations transmitted, and the bits they sent, we multiply each element in the station's chip sequence by the received chips, and divide by 8:

$$b_A = \frac{S \cdot A}{8} = \frac{(+1-1+3+1-1+3+1+1)}{8} = 1$$
$$b_B = \frac{S \cdot B}{8} = \frac{(+1-1-3-1-1-3+1-1)}{8} = -1$$
$$b_C = \frac{S \cdot C}{8} = \frac{(+1+1+3+1-1-3-1-1)}{8} = 0$$
$$b_D = \frac{S \cdot D}{8} = \frac{(+1+1+3-1+1+3+1-1)}{8} = 1$$

So Station A transmitted a 1, Station B transmitted a -1, Station C did not transmit, and Station D transmitted a 1.

4. (a) For $k = 2$, the second stage of connections is the bottleneck, and blocking could occur at the first stage. For $k = 2$, $nk = 8$ simultaneous connections are supported.

   For $k = 4$, since $k \geq n$, it is possible to reach a maximum of $nk = 16$ simultaneous connections as long as rearrangement is allowed.

   For $k = 10$, by the Clos theorem, since $k \geq 2n - 1$, a new connection can always be added without rearrangement. Since there are only 16 inputs and 16 outputs for the ten 4×4 second-stage switches, the maximum of 16 simultaneous connections is allowed under all circumstances.

   (b) Yes. Each input-output pair can be connected through any one of the k second-stage switches, giving k different ways to arrange the connections.

5. (a) Using byte count: 00000101 01000111 11100011 11100000 01111110 (The first byte is the byte count for the frame, which is 5 including the byte for byte count.)

   (b) Using byte stuffing: 01111110 01000111 11100011 11100000 11100000 11100000 01111110 01111110 (This is FLAG A B ESC ESC ESC FLAG FLAG.)

   (c) Using bit stuffing: 011111010 01000111 110100011 011111010 11100000 011111010

6. The probability that a 4-bit error will go undetected in a $n \times k$ bit block is given by the number of ways to arrange 4 erroneous bits in a way that will not be detected by a parity check, divided by the total number of ways to arrange 4 erroneous bits in a $n \times k$ bit block.

   Starting with the more simpler part of the fraction, the number of ways to arrange 4 erroneous bits is given by the number of 4-combinations possible in a $n \times k$ bit block:

$$\binom{n \times k}{4} = \frac{(nk)!}{4!(nk-4)!}$$

The number of ways to arrange 4 erroneous bits in a way that will not be detected by a parity check requires some deduction first. The only way the 4 error bits will pass a parity check is if they are evenly distributed through the $n \times k$ block. This means that a row with error bits must contain an even number of flipped bits, and a column with error bits must contain an even number flipped bits. There are multiple ways to arrange the error bits to achieve this. Consider a some permutations of bit block where $n = 2$ and $k = 3$, where 0 represents a correct bit and 1 represents a flipped bit:

$$1 : \begin{pmatrix} 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

$$2 : \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

$$3 : \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

Matrices 1 and 2 have undetected errors, while matrix 3 will be detected by the vertical parity check. Therefore, the flipped bits in a $n \times k$ bit block will only go undetected if they are arranged as the vertices of a rectangle. With this in mind, the number of ways 4 erroneous bits can be arranged and go undetected is given by the sum of all ways to position the top-left vertex to create a rectangle:

$$\begin{aligned} &\sum_{p=1}^{k-1}(k-p) \times \sum_{q=1}^{n-1}(n-q) \\ &= (\frac{1}{2}(k-1)k) \times (\frac{1}{2}(n-1)n) \\ &= \frac{nk(k-1)(n-1)}{4} \end{aligned}$$

So the final probability is given by:

$$P_{undetected} = \frac{nk(k-1)(n-1)}{4\frac{(nk)!}{4!(nk-4)!}}$$

7. The checksum for 1001 1100 1010 0011 is given by:

$$b_0 = 9$$
$$b_1 = 12$$
$$b_2 = 10$$
$$b_3 = 3$$
$$b_0 + b_1 + b_2 + b_3 = 34 = 4 \mod 15$$
$$4 = 0100_2$$
$$-4 = 1011_2$$
$$checksum = 1011$$

8. Given $G(x) = x^3 + 1$, or $G = 1001$, and $D = 10011101$, a 3-bit remainder must first be obtained using mod2 division:

$$\begin{array}{r} 10001100 \\ 1001\overline{)10011101000} \\ \underline{1001} \\ 1101 \\ \underline{1001} \\ 1000 \\ \underline{1001} \\ 100 \end{array}$$

This gives R = 100. The transmitted bit string is given by $< D, R >$ or 1001 1101 100.
If the third bit from the left is inverted (received bit string is 1011 1101 100), the error will be detected by the receiver, as the remainder of the mod2 division will be non-zero:

$$\begin{array}{r} 10101000 \\ 1001\overline{)10111101100} \\ \underline{1001} \\ 1011 \\ \underline{1001} \\ 1001 \\ \underline{1001} \\ 100 \end{array}$$

An example of a bit error that will not be detected by the receiver is if the two most significant bits and the two least significant bits are flipped, i.e. $E(x) = -x^7 + x^6 + x - 1$, and the received data is 0101 1110. This will not be detected by a CRC check because $E(x)$ is divisible by $G(x)$.