Exercises (pg. 54-59)- 1.2; 1.4; 1.10.1; 1.10.2; 1.12.1 thru 1.12.3

[1.2] The eight great ideas in computer architecture are similar to ideas from other fields. Match the eight ideas from computer architecture, “Design for Moore’s Law”, “Use Abstraction to Simplify Design”, “Make the Common Case Fast”, “Performance via Parallelism”, “Performance via Pipelining”, “Performance via Prediction”, “Hierarchy of Memories”, and “Dependability via Redundancy” to the following ideas from other fields:

a. Assembly lines in automobile manufacturing

1. Performance via Pipelining

b. Suspension bridge cables

1. Dependability via Redundancy

c. Aircraft and marine navigation systems that incorporate wind information

d. Express elevators in buildings

1. Make the Common Case Fast

e. Library reserve desk

1. Hierarchy of Memories

f. Increasing the gate area on a CMOS transistor to decrease its switching time

g. Adding electromagnetic aircraft catapults (which are electrically-powered as opposed to current steam-powered models), allowed by the increased power generation offered by the new reactor technology

h. Building self-driving cars whose control systems partially rely on existing sensor systems already installed into the base vehicle, such as lane departure systems and smart cruise control systems

1. Sldfj

[1.4] Assume a color display using 8 bits for each of the primary colors (red, green, blue) per pixel and a frame size of 1280×1024.

a. What is the minimum size in bytes of the frame buffer to store a frame?

1. 8 bits \* 3 primary colors = 24 bits/pixel = 3 bytes/pixel
2. Frame size of 1280×1024 = 1310720 pixels/frame
3. The minimum size in bytes= 1310720\*3 = 3,932,160 bytes

b. How long would it take, at a minimum, for the frame to be sent over a 100 Mbit/s network?

1. 3,932,160 bytes = 31.45728 Megabits
2. Network speed= 100 Mbit/second = 10 Mbit/millisecond
3. (31.45728 Megabits)/(10 Mbit/ms)
4. At a minimum, it would take 3.145728 milliseconds, which equals 0.003145728 seconds.

[1.10] Assume a 15 cm diameter wafer has a cost of 12, contains 84 dies, and has 0.020 defects/cm2
. Assume a 20 cm diameter wafer has a cost of 15, contains 100 dies, and
has 0.031 defects/cm2
.

[
1.10.1] Find the yield for both wafers.

1. Yield for 15 cm diameter wafer:
   1. 84 dies per wafer = wafer area/die area = **π(7.52) /die area**
   2. Die area = **π(7.52) / 84** ≈ 2.10374508053
   3. Yield = 0.95921653407
2. Yield for 20 cm diameter wafer:
   1. 100 dies per wafer = wafer area/die area = **π(102) /die area**
   2. Die area = **π(102) / 100** ≈ **π** ≈ 3.14159265359
   3. Yield = 0.9092888491

[1.10.2] Find the cost per die for both wafers.

1. Cost per die for 15 cm diameter wafer:
   1. Cost per die = cost per wafer/dies per wafer \* yield
   2. Cost per die = 12/(84\*0.95921653407)
   3. Cost per die = $0.14893106799 ≈ $0.15
2. Cost per die for 20 cm diameter wafer:
   1. Cost per die = cost per wafer/dies per wafer \* yield
   2. Cost per die = 15/(100\*0.9092888491)
   3. Cost per die = $0.1649640817 ≈ $0.16

[1.12] Section 1.10 cites as a pitfall the utilization of a subset of the performance equation as a performance metric. To illustrate this, consider the following two processors. P1 has a clock rate of 4 GHz, average CPI of 0.9, and requires the execution of 5.0E9 instructions. P2 has a clock rate of 3 GHz, an average CPI of 0.75, and requires the execution of 1.0E9 instructions.

[1.12.1]One usual fallacy is to consider the computer with the largest clock rate as having the largest performance. Check if this is true for P1 and P2.

While P1 has a larger clock rate than P2 (4 GHz > 3 GHz), P2 has better performance (. Thus, the notion that the computer with the largest clock rate has the best performance is not true for P1 and P2.

[1.12.2 ] Another fallacy is to consider that the processor executing the largest number of instructions will need a larger CPU time. Considering that processor P1 is executing a sequence of 1.0E9 instructions and that the CPI of processors P1 and P2 do not change, determine the number of instructions that P2 can execute in the same time that P1 needs to execute 1.0E9 instructions.

P2 can execute instructions in the time that it takes P1 to execute instructions.

[1.12.3] A common fallacy is to use MIPS (millions of instructions per second) to compare the performance of two different processors, and consider that the processor with the largest MIPS has the largest performance. Check if this is true for P1 and P2.