8404121 PROCESSOR DESIGN

Excercises based on the introduction lectures

1. There is a 4-stage pipeline in a processor, with the following delays in the stages

Stage	delay [ns]
1	17
2	15
3	19
4	14

   a) What is the cycle time, if we need to allocate additional 2 ns for the clocking overhead (setup and hold times)?
   b) What is the latency of the whole pipeline (in ns) then?
   c) What are the cycle time and latency if we need to accommodate yet another ±1 ns for possible clock skew?

2. We have designed a processor, which takes 280 mm^2 on silicon (the net area without pads). In addition we know that the pad ring will take 20% of the whole chip. We have the possibility to manufacture the processor as one, two or four chips (we can partition it evenly that way). Calculate the cost of the chip (set) in each case, if the cost of any package and related final testing will be USD 50 (for all good dies), the wafer processing of a 6" (15 cm) wafer will cost USD 4000 per wafer, and the defect density is 1/cm^2. Use the following equations to approximate the number of good dies.

Total number of dies per wafer $\mathbf{N \cong (\pi / 4A) \times (d - \sqrt{A})^2}$ (where *A* is die area and *d* diameter of the wafer). Yield $\mathbf{Y = e^{-\rho DA}}$ (where ρ_D is the defect density). The number of good dies is simply $\mathbf{N_G = N \times Y}$

   a) Which partition is the most economic?
   b) What if we change to a new technology, which allows the implementation in 80 mm^2 (net area) and increases the wafer cost to USD 8000? The packaging cost and defect density do not change.

3. Which two addressing modes can be used to replace all of the following addressing modes (the rest will be worked around in software)? Consider the case when you wish to have
   a) as few instructions as possible
   b) as fast as possible address calculation (= simple operations)

Addr.mode	example code	% of addressing
register	MV **R1**, R2	8
memory	LD R1, **0xFFFF**	3
reg. indirect	ST **[R1]**, R2	40
mem. indirect	LD R1, **[0x1111]**	1
post-inc	LD R2, **[R1++]**	4
post-dec	ST **[--R1]**, R2	1
offset	LD R2, 1**2[R1]**	25
indexed	LD R3, **12[R1+R2]**	3
immediate	LD R1, **#127**	15

   Calculate the increase of code size in these cases, assuming that the transfers comprise 30% of all code.