Exam solutions (DAT091, DAT092, DAT093)

Lars Svensson

Oct 27, 2015

What follows are brief suggested answers/solutions. Other solutions than those listed here may be acceptable.

1. (a) *Give at least two reasons why the packaging of electronic components is of importance for system development.*

   Packaging may influence the physical size of the finished product; the cost of manufacturing; the power dissipation capability; and the peak performance, just to name a few.

   (b) *What determines the frequency of time-of-flight ringing on a PCB? How could you make use of that information when debugging your PCB?*

   A wave travels along the PCB trace until it encounters an impedance discontinuity, which creates a reflected wave traveling in the other direction. This reflected wave may in turn be reflected at another discontinuity, which would result in a twice-reflected wave traveling in the original direction; etc. The frequency is determined by the physical distance betwen reflecting impedance discontinuities, and by the speed of light in the material surrounding the PCB trace. By observing the frequency of signal ringing, it is therefore possible to estimate the distance between impedance discontinuities, which may help in spotting the causes on the board.

   (c) *Compare and contrast surface mounting and through-hole mounting of components on PCBs. When would each method be preferrable?*

   In current practice, surface mounting is the default method. Components are placed directly on the PCB and (typically) fixed in place with glue, and the whole assembly is then baked under infrared lights to melt the solder paste applied at the component connections.

   The older through-hole method may still be used for some large and heavy components, but has several disadvantages: it does not scale to very small components, it creates more problems when components are placed on both sides of a board, and it requires time-consuming drilling of the through-holes. The benefits of easier component alignment and hand-tool workability may still motivate through-hole components in prototypes.

2. (a) *Discuss how to compare the logic design of the constituent configurable logic blocks of two different FPGA designs.*

   - It is of course possible to do a pure logic comparison: how many inputs and outputs each block has, the maximum delay from inputs to outputs, etc.

- A better comparison might be to map a "benchmark" VHDL (etc) design onto each of the two FPGA types, and compare the results: which design needs a smaller number of logic blocks to implement the functionality of the VHDL design?
- Then again, the blocks of the two FPGAs may not be equally large. It is arguably more fair to include the silicon area occupied by each logic block: if one block needs twice the area of the other, you "should" need only half as many of them to implement a certain design.
- You could (should?) include also the power dissipation of the two mapped designs in your comparison.
- The considerations above implicitly assume that the two FPGAs are manufactured in similar semiconductor processses. Compensating for the difference is fraught with difficulties. . .

(b) *Selection of a microprocessor for an embedded system is often based at least in part on benchmark results. Discuss some ways in which such benchmarks can mislead the designer! How can the designer avoid these problems?*

If a software benchmark becomes popular, processor manufacturers will tend to target this benchmark with their implementation improvements. Your own software need unfortunately not benefit very much, if it is not very similar to the benchmark codes. Likewise, a benchmark software that fits in the processor cache is likely to overestimate the performance reachable for a larger software that needs to fetch much more data and instructions from main memory.

A good way to avoid these problems is to use as a benchmark the code you intend to run on the finished system; or, failing that, code that is as similar to that code as possible (for example an earlier version of the same program).

3. (a) *How may device variability in present and future semiconductor processes contribute to higher power dissipation for digital circuits? Two mechanisms for full marks.*

Device variability is typically compensated for by an increased V_{dd} margin, in order to guarantee the required performance also with slow devices. Dynamic power scales as V_{dd}^2, so this increased margin is expensive in terms of power.

An increased variability for device threshold voltages means that a larger proportion of devices will suffer from increased leakage currents due to their low $|V_T|$. This effect will increase static power dissipation.

(b) *Briefly describe two ways to combat the detrimental effects of device variability.*

The "Future" lecture brought up a few: for defect yield loss, redundancy, error-correcting codes (ECC), and built-in self test (BIST) are common; for parametric yield loss, there are circuit techniques which adapt operation to the devices at hand. Future design-for-manufacturing (DFM) approaches include increased layout regularity.

4. (a) *The choice of technology platform is usually determined by a combination of several design requirements. Briefly discuss requirements that would cause you to consider an FPGA platform and an ASIC platform, respectively, and compare and contrast*

the two cases.

An FPGA solution is likely to be physically larger and need more power for a comparable performance level (but beware that the highest performance may not be reachable with an FPGA). The cost per part will be higher for the FPGA, but the NRE cost will be worse for the ASIC.

(b) *The ASIC/FPGA consideration need not be a strict either/or choice. Briefly discuss how the two technologies can be used to complement eachother in the same project.*

An FPGA prototype may help to uncover mistakes in the original specification, before going to an expensive ASIC manufacturing round. Also, an FPGA in an early product version may be replaced with an ASIC version with a lower cost per part, provided that manufacturing volumes turn out to be large enough to justify the extra NRE cost.

5. (a) *How can the selection of implementation technology affect the power dissipation of an electronic (sub)system? Compare at least three alternative technologies.*

A digital subsystem may often be implemented with software running on a microprocessor/microcontroller. In case the computations are a bad "fit" to the processor—for example, if the operand wordlengths are much different from those supported by the processor, or if massive parallelism of simple operations is needed—an FPGA may offer better power for similar performance. A purpose-designed ASIC is likely to offer and even better power/performance tradeoff: configurability requires mutliplexers, wires, and gates that are not used in the ASIC; these need space, so all parasitics will be larger in the FPGA.

(b) *Describe the concept of clock gating. How can it improve power dissipation in a digital system? Are there any pitfalls to consider when applying it?*

Clock gating reduces dynamic power dissipation in a digital system by eliminating unnecessary signal transitions: the clock net itself is not driven, neither are the internals of the clocked registers, and also the outputs of un-clocked registers are static and inject no transitions into the next level of combinational logic.

It is necessary to verify that the input/output behavior is unchanged by clock gating; and the introduction of gates in the clock paths may influence the clock budget.

6. *The reading material of the course has included the classic paper that introduced the "waterfall" design process (or at least the name which is commonly used for it...), and also some newer publications which outline current thinking. Describe, in your own words, the design-process progress since the 1970s until present; what has changed, what remains the same, and what is driving the evolution? Feel free to include examples from industry guest lectures, if appropriate.*

This is almost an essay-style task, so no solution will be suggested here. However, for full marks, I will expect some reference to the following:

- A "clean" process such as the "waterfall", with orderly progression from task to task and no iterations, has never been anything but an idealization
- The emphasis on complete documentation has been reduced over the years
- End-user/customer involvement, while recommended already in 1970, is considered

more important today

- Activities that were considered separate in 1970 (such as testing and verification) are in current practice much more integrated with the actual design/implementation work