Revision Report

Feedback comment:

Where: AR/Performance and Energy Efficiency

Comment: the illustrative learning outcome "Code a multi-threaded Python program that adds (in parallel) elements of two integer vectors" is not good. Python is NOT designed to illustrate multi-threading. In fact, it is well documented that Python's Global Interpreter Lock (GIL) prevents any real multithreading of CPU-bound tasks implemented in Python. Suggest changing this to be more general and not language-prescriptive: "Code a multi-threaded program that adds (in parallel) elements of two integer arrays"

How incorporated: As suggested, the text was changed to be more general.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: [Preamble]

Comment: Didn't Moore's law effectively end about 10 years ago?

How incorporated: revised and removed the part of Moore and Dennard Laws. This point did not contribute significantly to the preamble discussion.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: AR/Digital Logic and Digital Systems/ Functional hardware and software multi-layer architecture

Comment: I'm not sure what this means.

How incorporated: thanks for the feedback. Indeed, the topic is too general and has been removed from the text.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: AR/Machine-Level Data Representation
Follow the diagram and annotate the regions where fixed-length number representations affect accuracy and precision.

Comment: what diagram?

**How incorporated:**
Text changed to: Explain how fixed-length number representations can affect accuracy and precision.

**Why not incorporated:**

**Date considered:** 20/07/2023

---

**Feedback comment:**

Where: AR/Machine-Level Data Representation

Comment: suggestion to add a new learning outcome:

- Explain how a single adder (without overflow detection) can handle both signed (two’s complement) and unsigned (binary) input without “knowing” which format a given input is using.

**How incorporated:** Thanks for this suggestion. Learning outcome added to the main text.

**Why not incorporated:**

**Date considered:** 20/07/2023

---

**Feedback comment:**

Where: AR/Assembly Level Machine Organization

Suggestion to add a new topic:

- Fixed vs. variable-width instruction sets

**How incorporated:** thanks for the suggestion. The new topic has been added.

**Why not incorporated:**

**Date considered:** 20/07/2023

---

**Feedback comment:**

Where: AR/Functional Organization

Comment: suggestion to add “(e.g., stalls, forwarding)” as examples to the following topic:
Implementation of simple datapaths, including instruction pipelining, hazard detection and resolution

How incorporated: suggestion accepted. Text added. Thanks for the feedback.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: AR/Quantum Architectures

Comment: suggestion to add the topic:

“Measurement” (or is that implicitly covered in other bullet points?)

How incorporated: topic added to the text.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: Math Requirements

Comment: suggestion to add the topics:

- Required-> Discrete Maths: Boolean Algebra
- Required -> Logarithms
- Desired -> Limits

How incorporated: topics have been added.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: Crosscutting themes

Comment: suggestion to add bullet point:

- Separation of interface and implementation
How incorporated: suggestion accepted, bullet point added to the text.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: Course Packaging Suggestions/Computer Architecture - Introductory Course
Comment: If the course includes assembly language, a suggestion is to add Basic Programming to the prerequisites.

How incorporated: the course covers assembly language. Suggestions are accepted, and text has been added.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: Preamble
Comment: Notes from the reviewer:
Overall, I believe the additional emphasis on heterogeneous computing and the addition of quantum computing are merited and well done. The two areas that I probably would have liked to see more direct coverage would be security (perhaps cross-referencing the security KA -- I noted VM and data integrity as potential points for this) and system-level aspects (a lot of the topics seem to be presented as isolated components -- I would prefer not to see this be reflected to students).

How incorporated: we agree with the reviewer on both aspects. We will add a new knowledge unit explicitly to cover security. Subcommittee members of the systems-oriented knowledge areas (e.g. Architecture/Organization, Operating Systems, etc.) have met to discuss system-level aspects to integrate better the system-level topics. Thanks for this very useful feedback.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: Preamble
Comment: parallelism and also heterogeneity? In particular, this leads into accelerators, ML in particular, and to some extent QC

How incorporated: Suggestion accepted. Text added.
**Why not incorporated:**

**Date considered:** 20/07/2023

Feedback comment:

Where: Preamble

Comment: In the below areas, I do not see system-level topics covered (they mostly seem to be individual component topics).

Essentially, it feels like there is no unit or emphasis on how these disparate system components interact or how they relate to one another. While instructors may appreciate this, in my observation, students often struggle with this aspect and yearn for a more direct/intentional address. I don’t know if any cross-referencing with SF would be useful.

**How incorporated:** we agree with the reviewer on the system-level topics. Subcommittee members of the “systems” knowledge areas (e.g. Architecture/Organization, Operating Systems, etc) have met to discuss better integration, common definitions, and cross-referencing between systems knowledge units.

**Why not incorporated:**

**Date considered:** 20/07/2023

Feedback comment:

Where: AR/Digital Logic and Digital Systems

- Validate the timing diagram behavior of a simple processor, identifying data dependency issues.

Comment: Unclear what data dependency issues occur in simple processor design. Is this relative to a pipelined processor or an even simpler processor?

**How incorporated:** Added the word “pipelined” to be more specific on the processor design. Thanks for the feedback.

**Why not incorporated:**

**Date considered:** 20/07/2023

Feedback comment:

Where: AR/Memory Hierarchy

- Virtual memory (hardware support, cross-reference OS/Virtual Memory)
Fault handling and reliability

Reliability (cross-reference SF/Reliability through Redundancy)

Comment: Potentially, a security cross-reference would be useful here.

How incorporated: we will add a new knowledge unit focusing specifically on computer architecture aspects for security support.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: AR/Performance and Energy Efficiency

Comment: I see why prefetching is under performance and energy, but it seems as though it logically fits within the memory hierarchy topic list. My undergraduate students often intuit prefetching along with caching.

How incorporated: this topic has been re-arranged and moved to AR/Memory Hierarchy.

Why not incorporated:

Date considered: 20/07/2023

Feedback comment:

Where: AR/Quantum Architectures

Comment on the statement: a QPU is a heterogeneous multicore architecture like an FPGA or a GPU.

Comment: This is an example that leans more towards considering the whole system/set of tools and their relation.

How incorporated: agree that the example is not fully consistent with the other examples. This statement has been removed from the text.

Why not incorporated:

Date considered: 20/7/2023