Chapter 4 Review

- Static scheduling
  - Done in compiler
- Loop unrolling
  - Reduces loop overhead
  - Produces straight line code without control dependencies
  - Improves scheduling possibilities

Chapter 4 Review

- Static Branch Prediction
  - Predict taken
  - Predict based on branch direction
  - Profile-based
- VLIW
  - Very Long Instruction Word
  - Static scheduling, in-order-execution
  - All parallelism and hazards detected by compiler

Chapter 4 Review

- Loop-Level Parallelism
- Loop-carried dependencies
  - GCD test
  - Dependence analysis to detect dependencies
  - We then saw examples of code transformations to remove loop carried dependencies

Chapter 4 Review

- Algebraic and arithmetic compiler optimizations
  - Back substitution
  - Tree height reduction
- Software pipelining
  - Reorganize loops to interleave instructions from different iterations

Chapter 4 Review

- Hardware support
  - conditional instructions
  - Execute an instruction conditionally on the result of a test
  - Can be used to replace branches, replacing a control dependence with a data dependence
  - Conditional moves very common
- Hardware support for compiler speculation
  - Need support for correct exception behavior
  - Need support to be able to speculatively reorder loads and stores
- Example of a VLIW machine
  - Transmeta Crusoe