RISC Architectures:
Fundamentals, Design Alternatives, and Futures?
(Summary of Slides)

1. Cover Page

<u>RISC in General</u>

2. Early Statements of Risc Advantages	/* Summary of first RISC paper */
3. What is Risc Architecture	/* Common properties, nonproperties, and Variants */
4. Risc Genealogy	/* History of RISC Architectures */
5. Risc Principles	/* General and Specific Principles */
6. Risc Benefits	/* Cost & Perf. -- R3000 vs 80386 */
7. Performance Equation	/* Time/Task = (instr/task)(cycle/instr)(time/cycle) */
8. Model for Performance	/* Performance equation written a different way */
9. Making a Fast RISC Machine	/* Fast clock rate, low cycle/inst, low inst/task */
10. Why the Risc Machine Wins	/* vs CISC approaches */
11. What Factors affect RISC Performance	/* looking at CPI, clock rate,and instruction count */
12. Where does CPI advantage come from?	/* lower CPI comes from pipelining */
13. Native Instruction Execution Rates	/* Machine, Clock Rate, CPI, Native Mips */
14. Cycles per Instruction (MHz/VUP)	/* Graph showing cpi vs time for different processors */
15. Costs of RISC Approach	/* Summary of Costs associated with RISC */

<u>RISC vs VAX</u>

16. The VAX Story	/* Model, Clock, Cycles/instr, Native MIPS */
17. Instruction Set Usage - VAX and MIPS	/* for three applications look at frequency of operation types (ALU ops, moves, branches */
18. Addressing Mode Usage	/* graph showing frequency of different addressing modes */
19. VAX vs RISC Memory Bandwidth	/* Comparing memory bandwidth requirements for VAX and MIPS */

20. VAX vs RISC Memory Bandwidth	/* Graph showing data and instruction bytes */

Different RISC Implementations

21. Two Basic RISC Approaches	/* Stanford/IBM vs Berkeley */
22. Register Window Scheme	/* Graph showing register windows */
23. Blank	
24. When Do Register Windows Work?	/* Graph showing good and bad behavior for register windows */
25. Comparison of the Two Approaches	/* Compare performance of register windowing */
26. Savings from Register Windows	/* Compares reduction in instruction count versus optimization level for TEX and Spice on MIPS and Sparc */

Architecture Considerations

27. Minimizing Instruction Execution Count	/* compiler optimizations to minimize instruction count */
28. Register Allocation Example	/* Optimization achieved by using registers */
29. Register Usage	/* Graph showing fraction of references versus number of registers */
30. Optimization Affects - Instruction Counts	/* Graph showing 3 applications and the number of instructions executed using compiler optimization */
31. Optimization Affects - Instruction Frequencies	/* Graph showing 3 applications and the frequency of operation type (ALU, load/store, branches).
32. Minimizing the Cycles per Instruction	/* General comments */
33. Typical RISC Pipeline	/* Drawing showing typical pipeline with example of a load and branch delay */
34. Instruction Costs and Latencies	/* Cost of branches and memory references, and goals for instructions */
35. Architecture and Compiler Goals	/* Three architecture and compiler goals */
36. Blank	
37. Loads and Stores	/* General comments, along with comparison of different machines */
38. Optimizing the load delay	/* Consider code example */

39. Effectiveness of Load Delay Scheduling /* Graph showing fraction of loads that cause pipeline stalls for three applications */

Branch Architectures

40. Different Branch Architectures /* Compare branch architectures for various machines */

41. Delayed and Cancelling Branches /* Discussion on delayed and cancelling branches */

42. What Types of Branches Occur /* Conditional and unconditional branches for DLX, 8086, and VAX architectures */

43. Branch Costs and Delays /* branch costs and delays for various machines */

44. Scheduling a delayed branch /* Drawing showing different delay slot placements */

45. Effectiveness of simple delayed branch /* Graph showing delay slots filled and usefully filled */

46. Branch Cost in Cycles for Various Schemes /* Graph showing branch cost in cycles vs branch type */

47. Total affect of Integer Pipeline Stalls /* Graph showing 3 applications and the percentage of load delay stall cycles and branch delay stall cycles */

Floating Point on a RISC

48. Fast Floating Point on a RISC /* challenges, approaches, difficulties and solutions */

49. FP Pipeline /* Drawing showing FP pipeline */

50. Out-of-order Completion in the FP Unit /* Issues dealing with out-of-order completion in the FP Unit */

51. MIPS R2000/3000 FP Unit /* Drawing showing MIPS FP Unit */

52. FP Pipeline Stalls /* Graph showing Fraction of all cycles in execution vs types of stalls/delays */

Maximizing Clock Cycle Considerations

53. Maximizing Clock Cycle /* Issues in maximizing clock cycle */

54. Instruction fetch/decode /* Minimize instruction encoding */

55. Blank

56 Pipeline Control and Interrupt Handling /* General Comments */

Architectual Approaches

57. Architectural Approaches /* Graph showing different architectures on graph with axis of faster clock rate vs lower CPI */

58. Superscalar /* Drawing showing superscalar */

59. Superpipelined /* Drawing showing superpipeling */

60. Superscalar vs Superpipelined /* Comparison */

61. What Limits Instruction Level Parallelism/* Application parallelism and Machine parallelism */

Summary

62. Final Thoughts and Open Issues