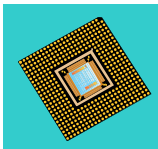


COMPAQ

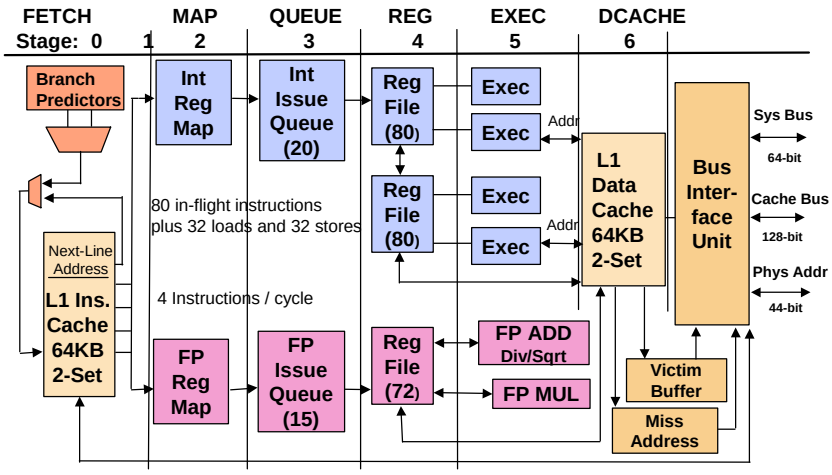
The Alpha 21264 Microprocessor: Out-of-Order Execution at 600 MHz

R. E. Kessler
Compaq Computer Corporation
Shrewsbury, MA

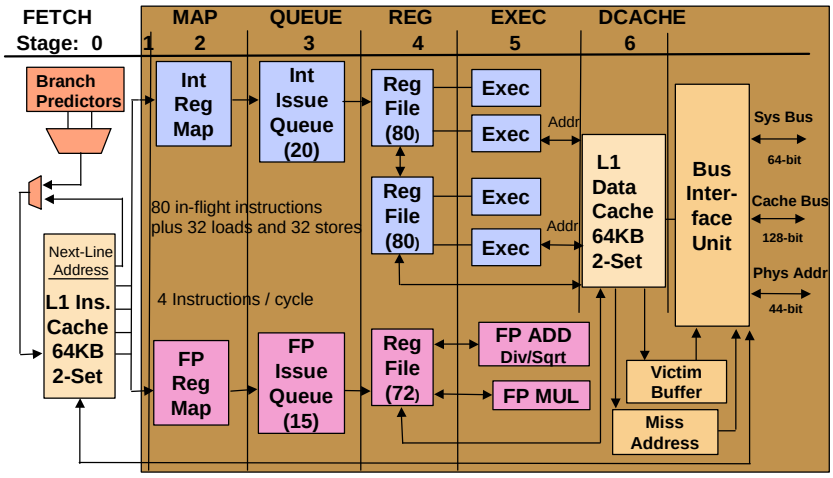
Some Highlights

- **Continued Alpha performance leadership**
  - 600 MHz operation in 0.35u CMOS6, 6 metal layers, 2.2V
  - 15 Million transistors, 3.1 cm^2, 587 pin PGA
  - Specint95 of 30+ and Specfp95 of 50+
  - Out-of-order and speculative execution
  - 4-way integer issue
  - 2-way floating-point issue
  - Sophisticated tournament branch prediction
  - High-bandwidth memory system (1+ GB/sec)

Alpha 21264: Block Diagram



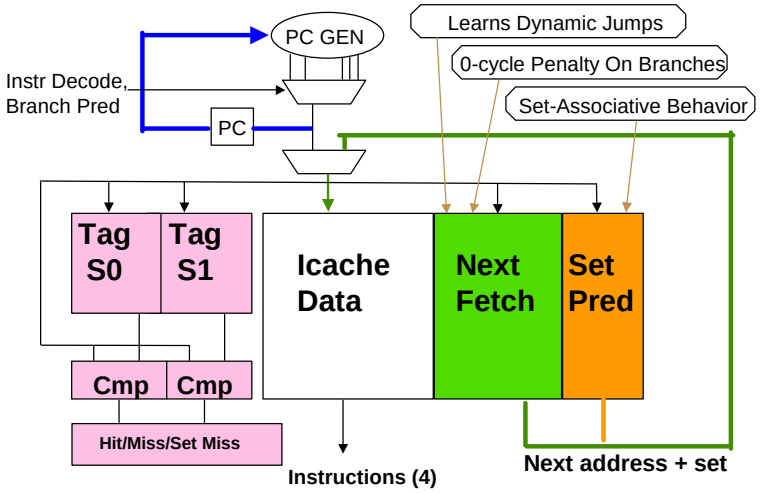
Alpha 21264: Block Diagram



21264 Instruction Fetch Bandwidth Enablers

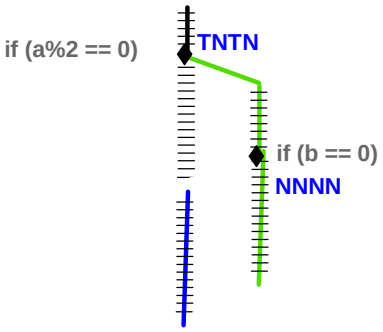
- **The 64 KB two-way associative instruction cache supplies four instructions every cycle**
- **The *next-fetch* and *set predictors* provide the fast cache access times of a direct-mapped cache and eliminate bubbles in non-sequential control flows**
- **The instruction fetcher speculates through up to 20 branch predictions to supply a continuous stream of instructions**
- **The tournament branch predictor dynamically selects between *Local* and *Global* history to minimize mispredicts**

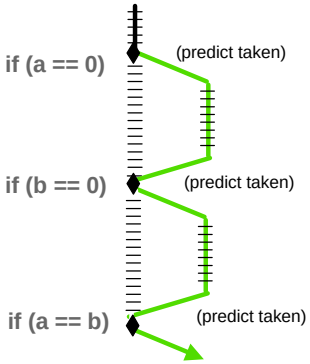
Instruction Stream Improvements



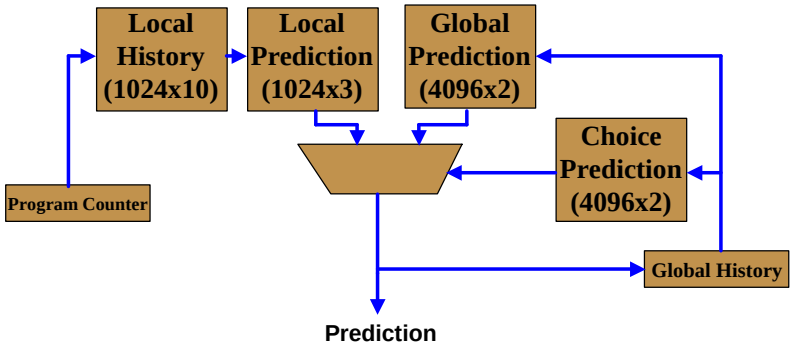
Fetch Stage: Branch Prediction

- Some branch directions can be predicted based on their past behavior: *Local Correlation*
- Others can be predicted based on how the program arrived at the branch: *Global Correlation*

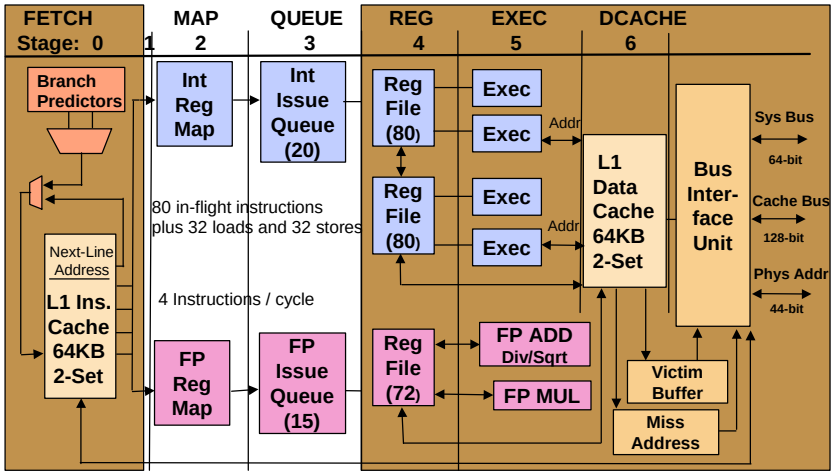




Tournament Branch Prediction



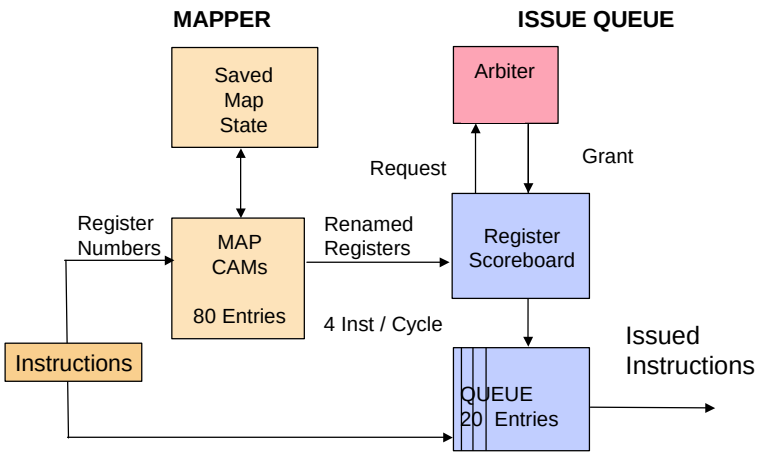
Alpha 21264: Block Diagram



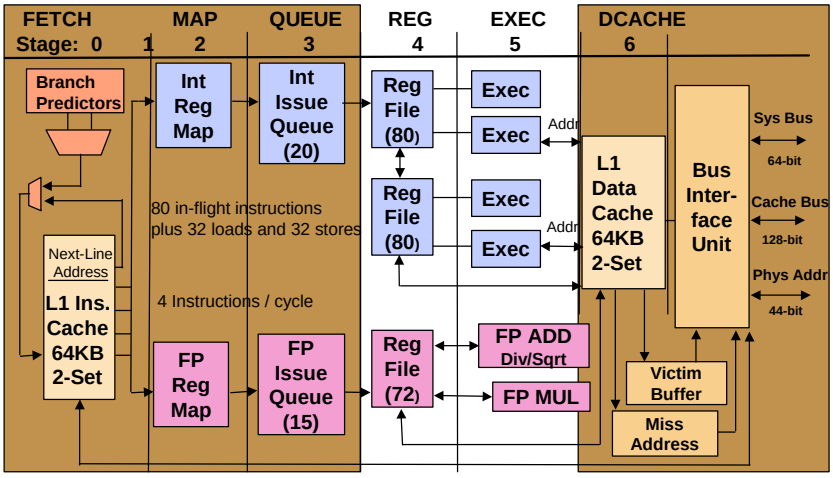
Mapper and Queue Stages

- **Mapper:**
  - Rename 4 instructions per cycle (8 source / 4 dest)
  - 80 integer + 72 floating-point physical registers
- **Queue Stage:**
  - Integer: 20 entries / Quad-Issue
  - Floating Point: 15 entries / Dual-Issue
  - Instructions issued out-of-order when data ready
    - Prioritized from oldest to youngest each cycle
  - Instructions leave the queues after they issue
    - The queue collapses every cycle as needed

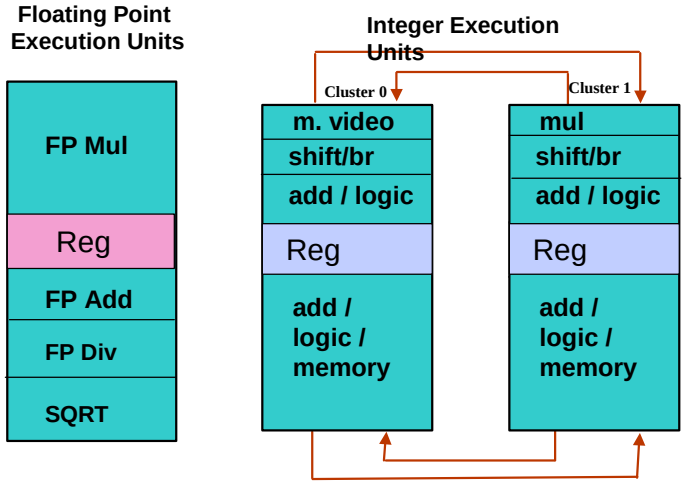
Map and Queue Stages



Alpha 21264: Block Diagram

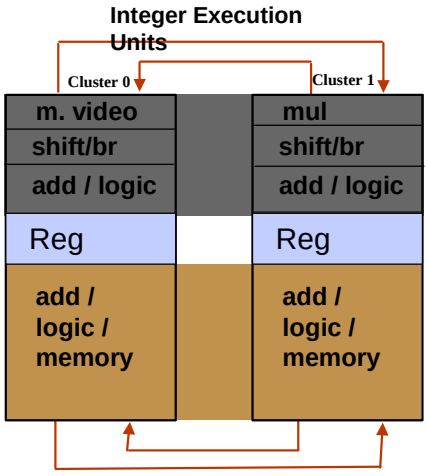


Register and Execute Stages

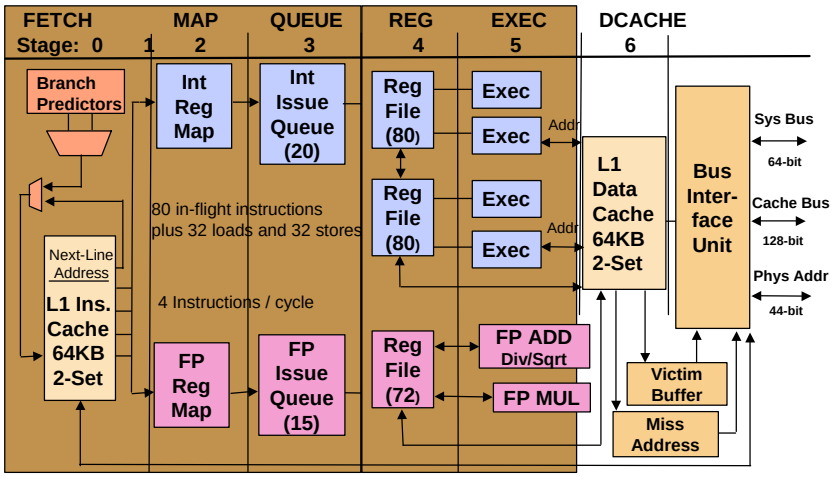


Integer Cross-Cluster Instruction Scheduling and Execution

- **Instructions are statically pre-slotted to the upper or lower execution pipes**
- **The issue queue dynamically selects between the left and right clusters**
- **This has most of the performance of 4-way with the simplicity of 2-way issue**



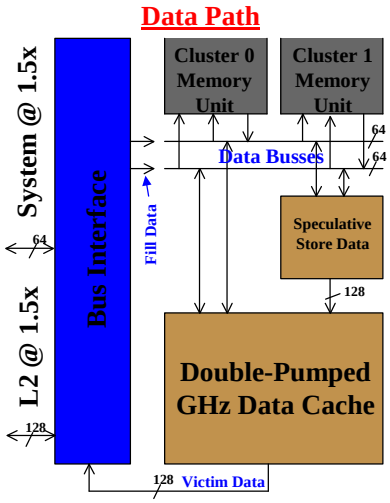
Alpha 21264: Block Diagram



21264 On-Chip Memory System Features

- **Two loads/stores per cycle**
  - Any combination
- **64 KB two-way associative L1 data cache (9.6 GB/sec)**
  - Phase-pipelined at > 1 GHz (no bank conflicts!)
  - 3 cycle latency (issue to issue of consumer) with hit prediction
- **Out-of-order and speculative execution**
  - Minimizes effective memory latency
- **32 outstanding loads and 32 outstanding stores**
  - Maximizes memory system parallelism
- **Speculative stores forward data to subsequent loads**

Memory System (Continued)

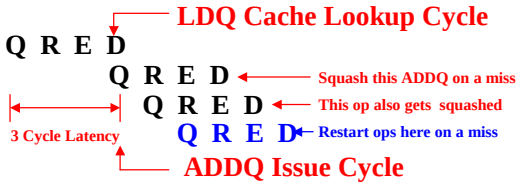


Address Path

- **New references check addresses against existing references**
- **Speculative store data that address matches bypasses to loads**
- **Multiple misses to the same cache block merge**
- **Hazard detection logic manages out-of-order references**

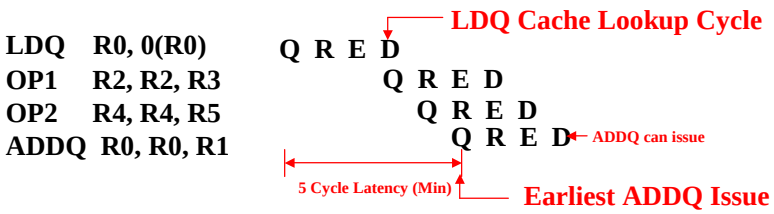
Low-latency Speculative Issue of Integer Load Data Consumers (Predict Hit)

```
LDQ   R0, 0(R0)
ADDQ  R0, R0, R1
OP    R2, R2, R3
```



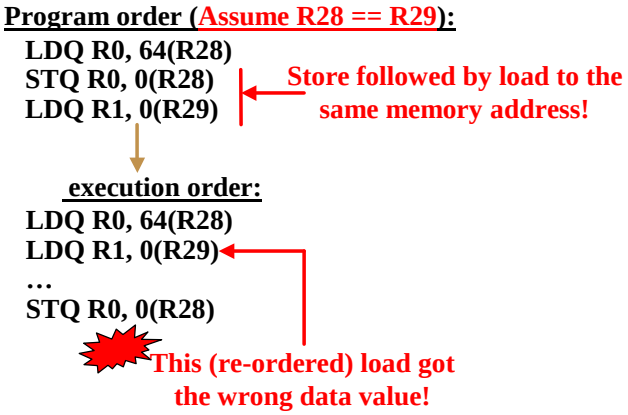
- **When predicting a load hit:**
  - The ADDQ issues (speculatively) after 3 cycles
  - Best performance if the load actually hits (matching the prediction)
  - The ADDQ issues before the load hit/miss calculation is known
- **If the LDQ misses when predicted to hit:**
  - Squash two cycles (replay the ADDQ and its consumers)
  - Force a “mini-replay” (direct from the issue queue)

Low-latency Speculative Issue of Integer Load Data Consumers (Predict Miss)

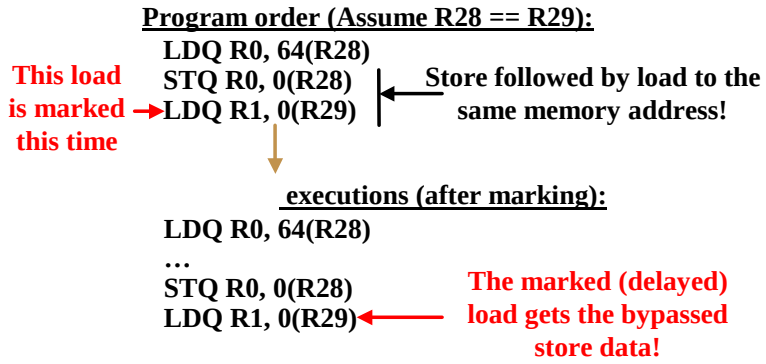


- **When predicting a load miss:**
  - The minimum load latency is 5 cycles (more on a miss)
  - There are no squashes
  - Best performance if the load actually misses (as predicted)
- **The hit/miss predictor:**
  - MSB of 4-bit counter (hits increment by 1, misses decrement by 2)

Dynamic Hazard Avoidance (Before Marking)



Dynamic Hazard Avoidance (After Marking/Training)



New Memory Prefetches

- **Software-directed prefetches**
  - Prefetch
  - Prefetch w/ modify intent
  - Prefetch, evict it next
  - Evict Block (eject data from the cache)
  - Write hint
    - allocate block with no data read
    - useful for full cache block writes

21264 Off-Chip Memory System Features

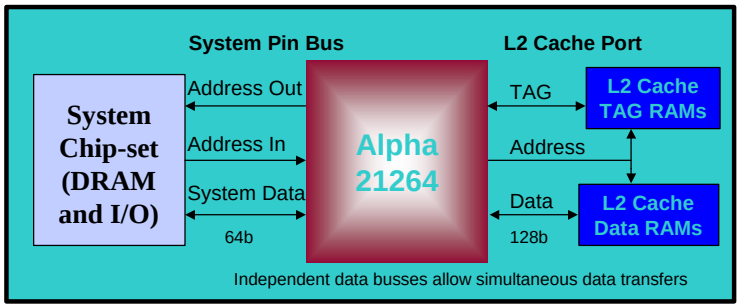
- **8 outstanding block fills + 8 victims**
- **Split L2 cache and system busses (back-side cache)**
- **High-speed (bandwidth) point-to-point channels**

  clock-forwarding technology, low pin counts
- **L2 hit load latency (load issue to consumer issue) = 6 cycles + SRAM latency**
- **Max L2 cache bandwidth of 16 bytes per 1.5 cycles**

  6.4 GB/sec with a 400Mhz transfer rate
- **L2 miss load latency can be 160ns (60 ns DRAM)**
- **Max system bandwidth of 8 bytes per 1.5 cycles**

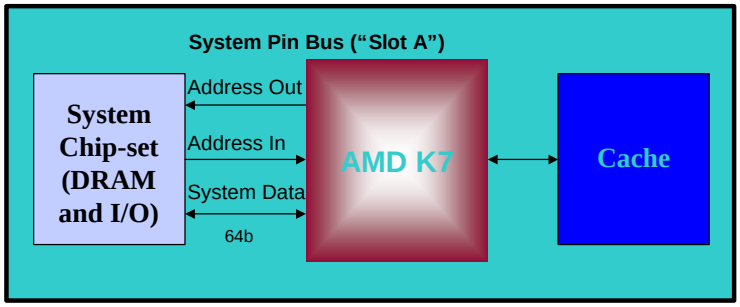
  3.2 GB/sec with a 400Mhz transfer rate

21264 Pin Bus



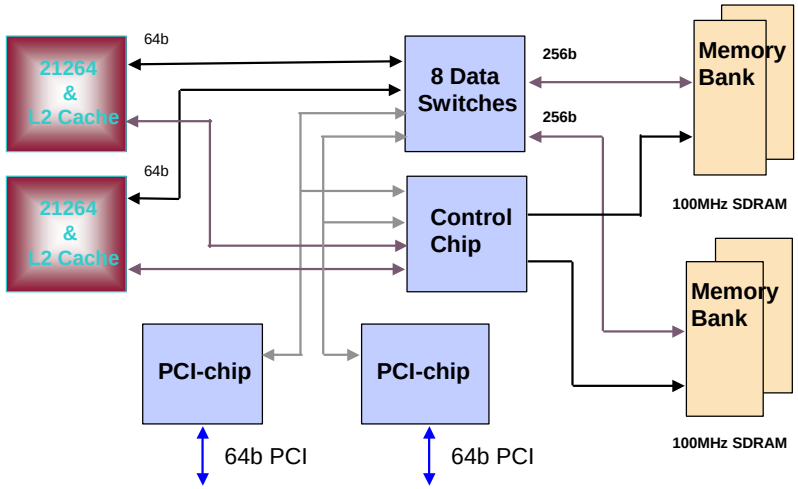
L2 Cache:	0 -16MB	Synch Direct Mapped	Peak Bandwidth:
Example 1:	Reg-Reg	133 MHz Burst RAM	2.1 GB/Sec
Example 2:	'RISC'	200+ MHz Late Write	3.2+ GB/Sec
Example 3:	Dual Data	400+ MHz	6.4+ GB/Sec

Compaq Alpha / AMD Shared System Pin Bus (External Interface)

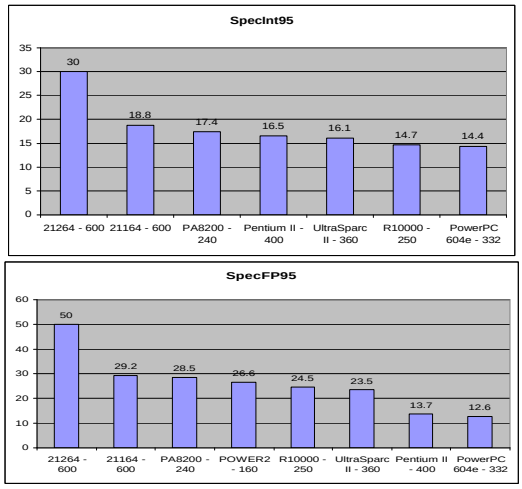


- High-performance system pin bus
- Shared system chipset designs
- This is a win-win!

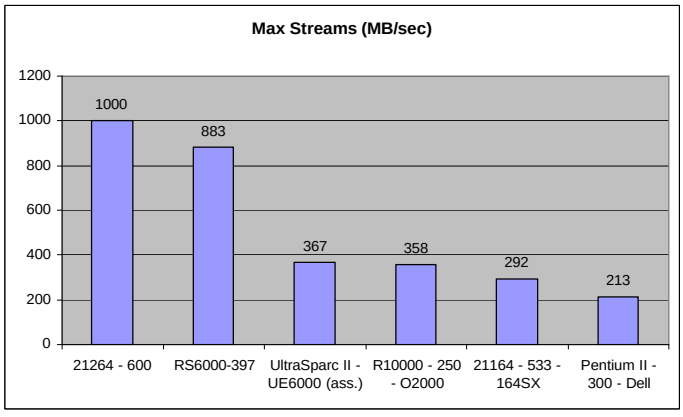
Dual 21264 System Example



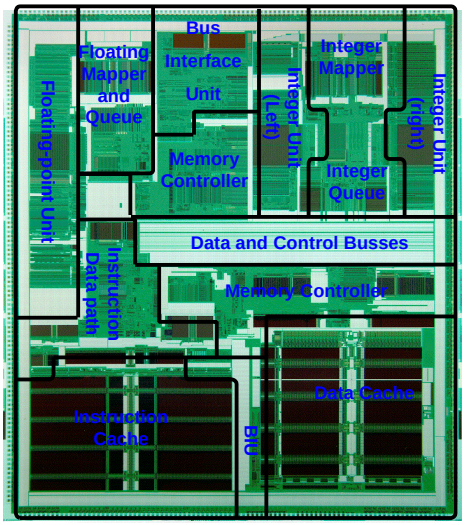
Measured Performance: SPEC95



Measured Performance: STREAMS



Alpha 21264



Summary

- **The 21264 will maintain Alpha's performance lead**
  - 30+ Specint95 and 50+ Specfp95
  - 1+ GB/sec memory bandwidth
- **The 21264 proves that both high frequency and sophisticated architectural features can coexist**
  - high-bandwidth speculative instruction fetch
  - out-of-order execution
  - 6-way instruction issue
  - highly parallel out-of-order memory system