



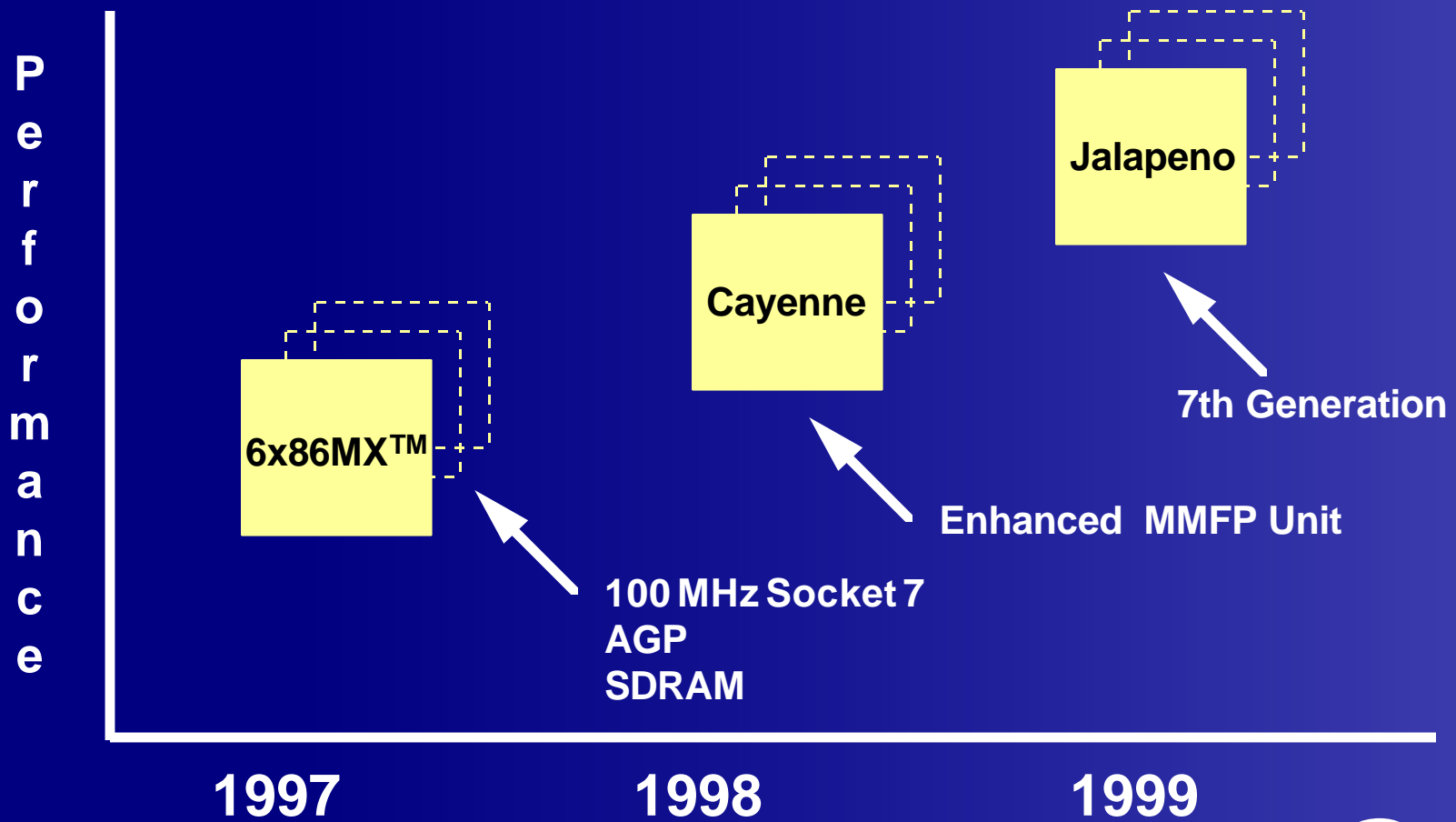
Beyond the 6x86MX™ Processor

Robert Maher
Vice President of Engineering
Cyril Corporation

www.cyril.com

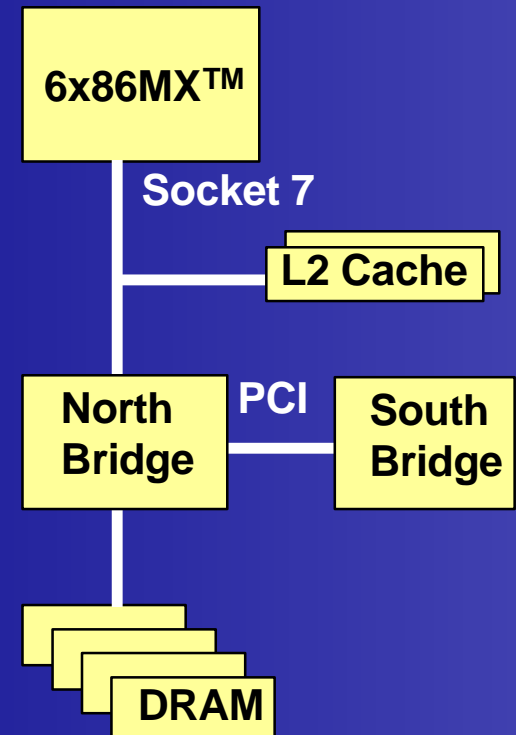
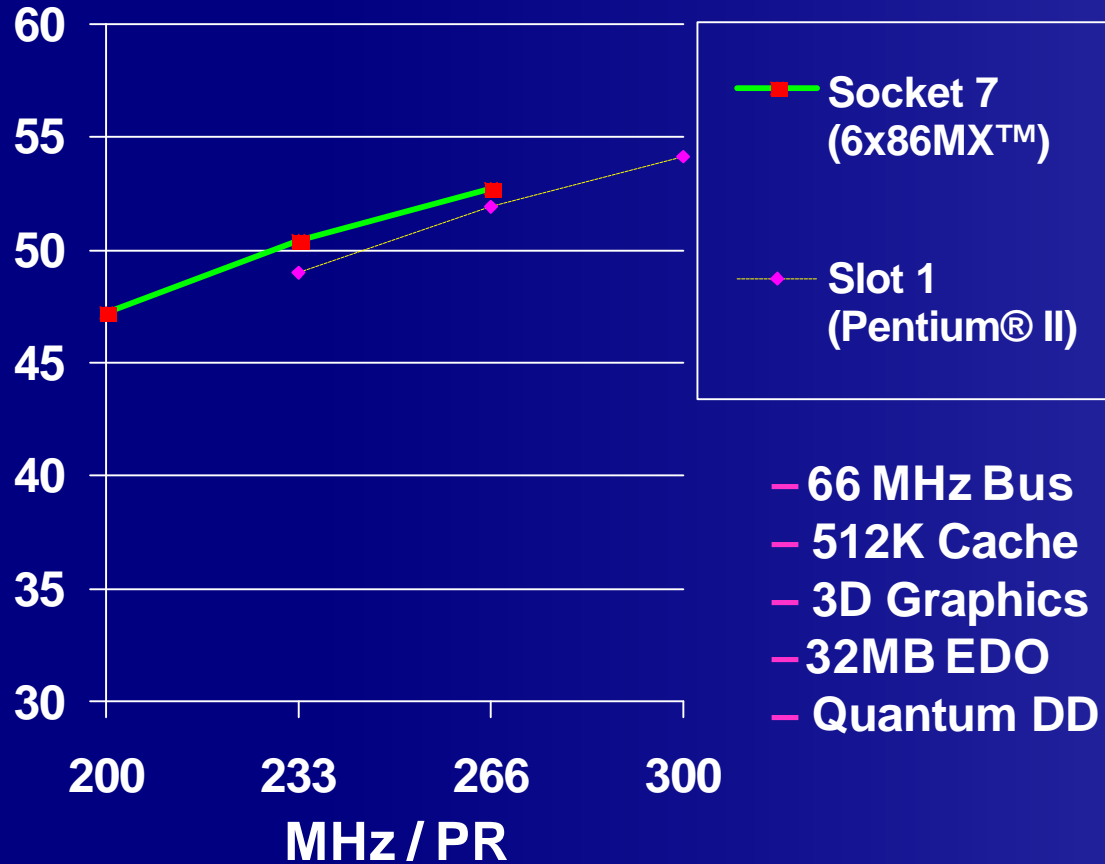
Cyril is a registered trademark and 6x86MX is a trademark of Cyril Corporation. MMX is a trademark of Intel Corporation.
All other brand or product names are trademarks or registered trademarks of their respective holders.

Core Roadmap



Socket 7 Today

Winstone 97[®] / Windows[®] 95
Business Applications



Socket 7: The Future - 6x86MX™

- ◆ **AGP support in Q1 1998**
- ◆ **100 MHz system bus Q1 1998**
 - **Lookaside caches: Up to 2MB support**
 - Tag RAM in north bridge
 - 5ns data RAMS
 - 3-1-1-1 line fills
- ◆ **SDRAM today, DDR SDRAM 2H 1998**
- ◆ **Firewire (IEEE 1394), Device Bay, ATA66**

The Socket 7 System

◆ Balanced system design

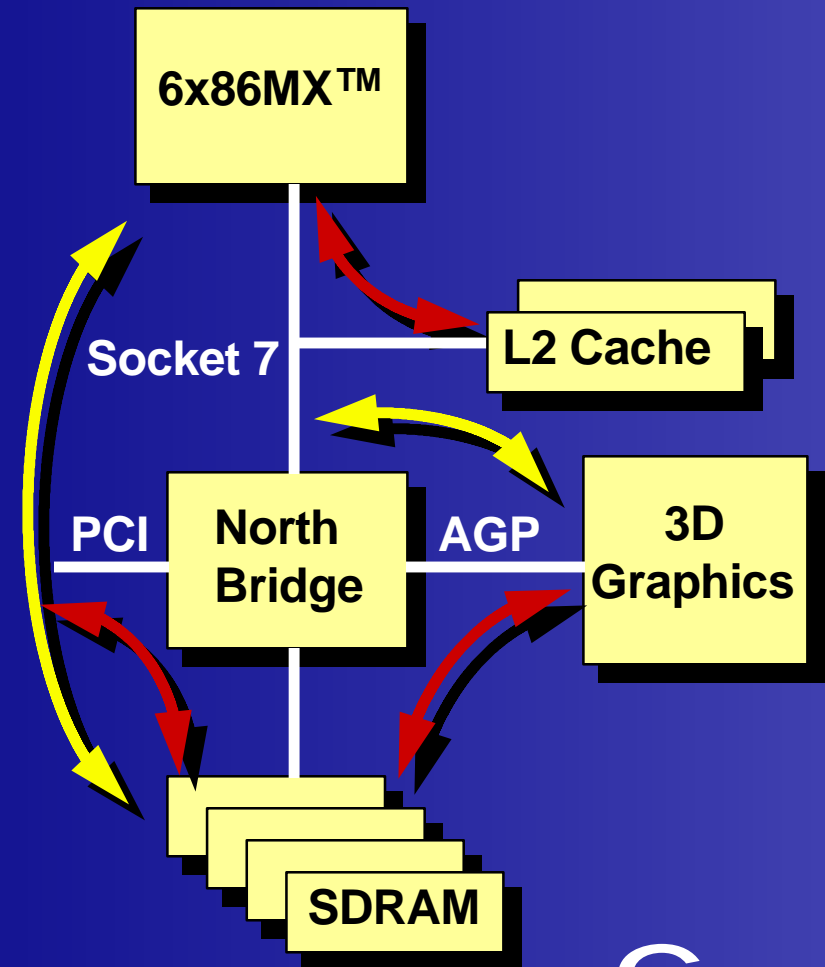
➤ Concurrency

- CPU \longleftrightarrow L2
- PCI \longleftrightarrow MEM
- AGP \longleftrightarrow MEM

➤ Concurrency

- CPU \longleftrightarrow MEM
- CPU \longleftrightarrow AGP

◆ Multimedia applications have data sets > 2MB therefore latency to main memory is critical



Socket 7 System Performance

- As core frequency improves, L2 cache at 100 MHz keeps L1 miss penalty (core clocks) consistent with 66 MHz systems
- 100 MHz system bus → 50% increase in L2 cache performance
- Low latency / High bandwidth path to main memory
 - 100 MHz Socket 7 bus
 - 100 MHz SDRAM

Performance will scale with 100 MHz system bus

Beyond 6x86MX™: Cayenne Core

- ◆ **Based on 6x86MX™ integer core**
 - 4MB paging
 - Virtual mode enhancements
 - Frequency optimizations
- ◆ **64KByte L1 cache**
- ◆ **Pipelined, dual-issue Multimedia Floating Point (MMFP) Unit**
- ◆ **Enhanced Multimedia Instructions**
- ◆ **.25 micron process technology**

Cayenne Core: FPU

◆ Dual-issue floating point

- 2 FOP
- 1 load / 1 store, 1 FOP
- Instruction queue

◆ Fully pipelined floating point unit

Throughput / Latency		
	Cayenne	6x86MX™
FXCHG	0	2/2
LOAD/STORE	1/4	4/4
ADD	1/4	4/4
MULTIPLY (SP)	1/4	4/4
MULTIPLY (DP)	3/6	6/6



Cayenne Core: MMX™ Instruction Execution

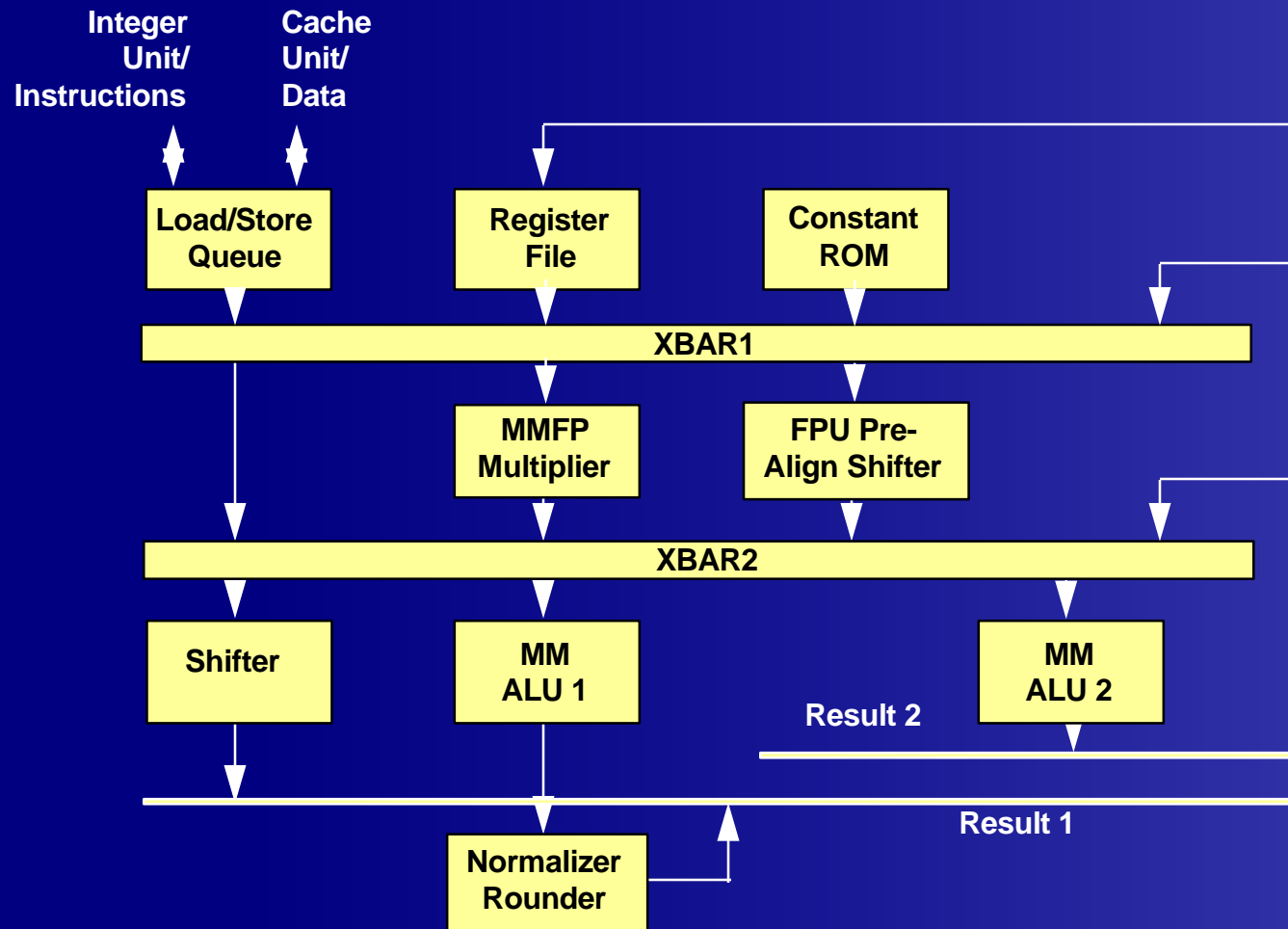
- ◆ **Dual Issue / Dual Execute**
 - 1 shift / 1 multiply
 - 2 MMX add/logical units
 - 1 load / 1 store
- ◆ **Fully pipelined**
- ◆ **Single-cycle execution**
- ◆ **Multiplies execute with 1 cycle throughput, 2 cycle latency**
- ◆ **Single-cycle Multimedia FP Instruction context switch**

Cayenne Core: MMFP Instruction Unit

- ◆ **Enhanced MMX™ instruction execution unit**
 - **Additional data types**
 - IEEE 754 single-precision floating point
 - **Two single-precision floating point results per operation**
 - **Dual execute ➡ 4 FLOPS per clock cycle**
 - 1 FP add unit / 1 FP multiply unit

> 1 GFLOP peak performance

MMFP Instruction Unit Block Diagram



Multimedia Floating Point (MMFP) Instruction Extensions

- ◆ **Add, Sub, Multiply, Convert, and Compare operations which support FP data types**
- ◆ **Scatter/Gather operations for vectorized floating point**
 - **Gather and scatter triangle vertices for optimum parallelism**
- ◆ **Reciprocal and Reciprocal Square Root**
- ◆ **Motion Estimation Instruction**

MMFP: Software Support and Execution

- ◆ Work with Microsoft to support MMFP in retained mode Direct3D driver
- ◆ Assist 3rd-party software development: i.e., game developers: immediate mode drivers

Throughput / Latency		
	MMX™	Equiv x86FP
Load/Store/Convert	1/1	2/4
Add/Multiply	1/3	2/4
Reciprocal	3/5	48+/48+
Root Reciprocal	3/5	140+/140+

MMFP: Geometry Transforms

$$[x' \ y' \ z' \ w] = [x \ y \ z \ 1] * T_{\text{geom}} * T_{\text{view}}$$

$$T_{\text{geom}} = \begin{vmatrix} a00 & a01 & a02 & 0 \\ a10 & a11 & a12 & 0 \\ a20 & a21 & a22 & 0 \\ tx & ty & tz & 1 \end{vmatrix}$$

$$T_{\text{orth}} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T_{\text{pers}} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1/d \\ 0 & 0 & 0 & 0 \end{vmatrix}$$

- ◆ 4 x 4 matrix multiply reduces to 3x3 matrix multiply with a 3x1 vector add
- ◆ Translates 2 vertices at a time in 21 clocks
- ◆ 10.5 clocks/vertex vs. 36 clocks/vertex with standard x86 code

> 3x Performance Increase

MMFP: Lighting Calculation

$$I_d = I_i * k_d * (L \cdot N)$$

- I_i is the intensity at the light source
- k_d is a coefficient of diffuse lighting
- L is direction vector
- N is surface normal

- ◆ Processes 2 vertices in 23 clocks
- ◆ 11.5 clocks/vertex vs. 129 clocks with standard x86 code (70 clock square root)

>10 million meshed triangles / sec peak performance
(Geometry + Lighting)

Summary

- ◆ **6x86MX™ to support 100 MHz socket 7 bus with state-of-the-art system features:
AGP, SDRAM, 1394, Device Bay, ATA66**
- ◆ **Cayenne Core:**
 - **Dual-issue, pipelined MMFP Instruction Unit to enable highest performance 3D graphics, DVD, and 3D audio**
 - **Multimedia floating point (MMFP) extensions**
 - **PR300 to PR400**
 - **65 sq mm in .25 um, 5 layer metal process**
 - **2H98 production**

