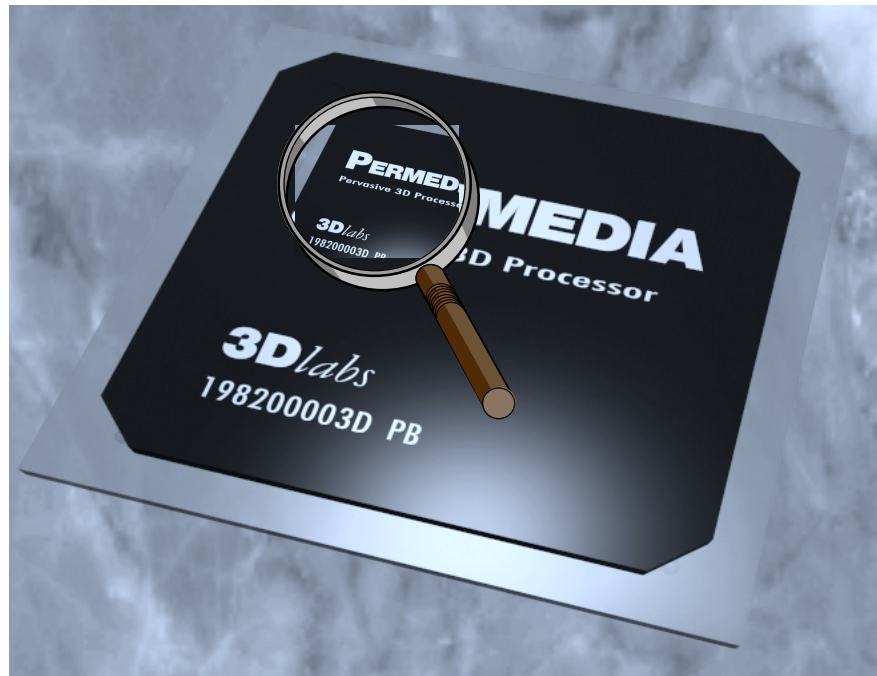


PERMEDIA and GLINT Delta

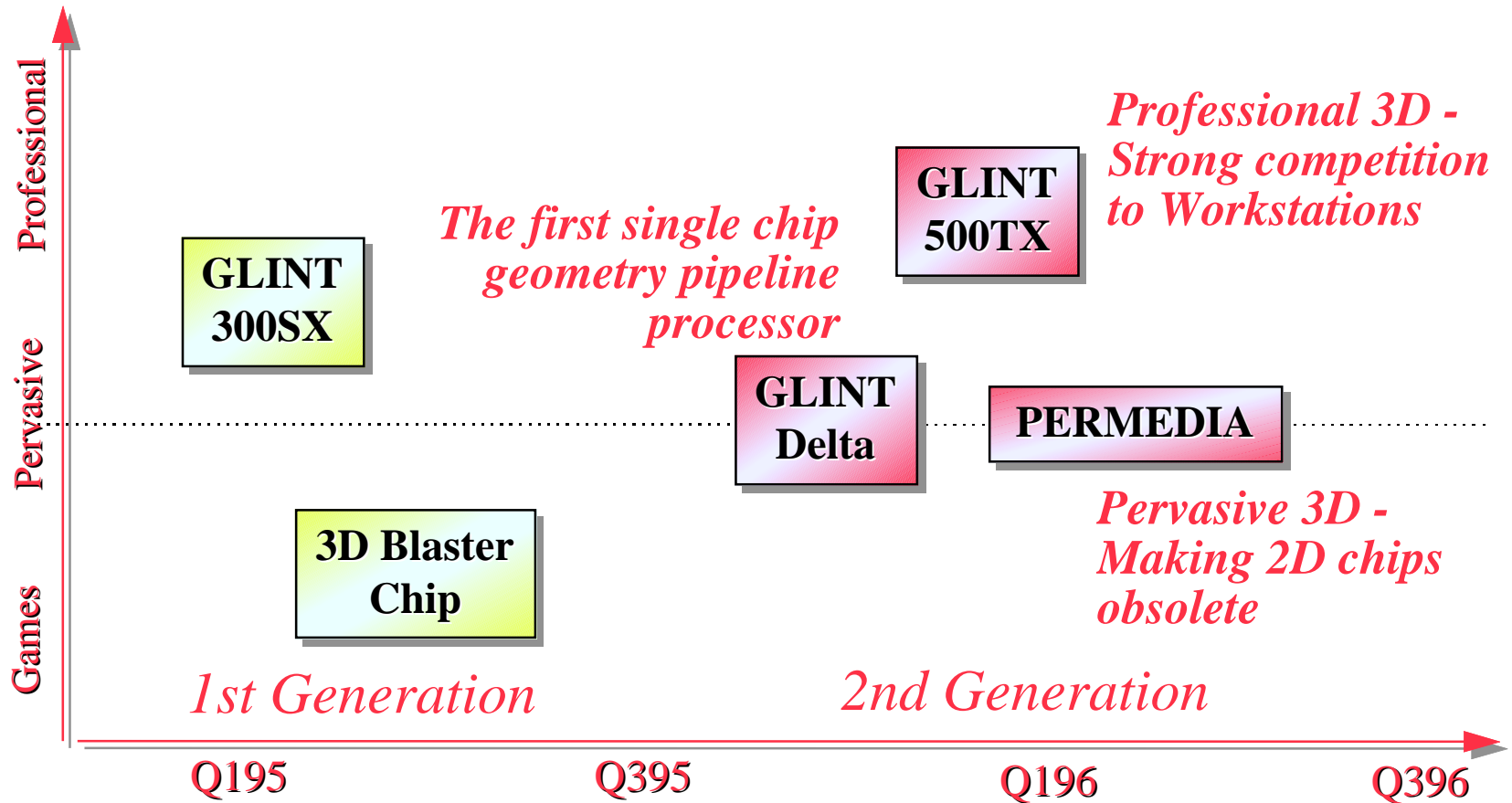
New Generation Silicon for 3D Graphics



Neil Trevett
Vice President Marketing
(408) 436 3455
www.3dlabs.com

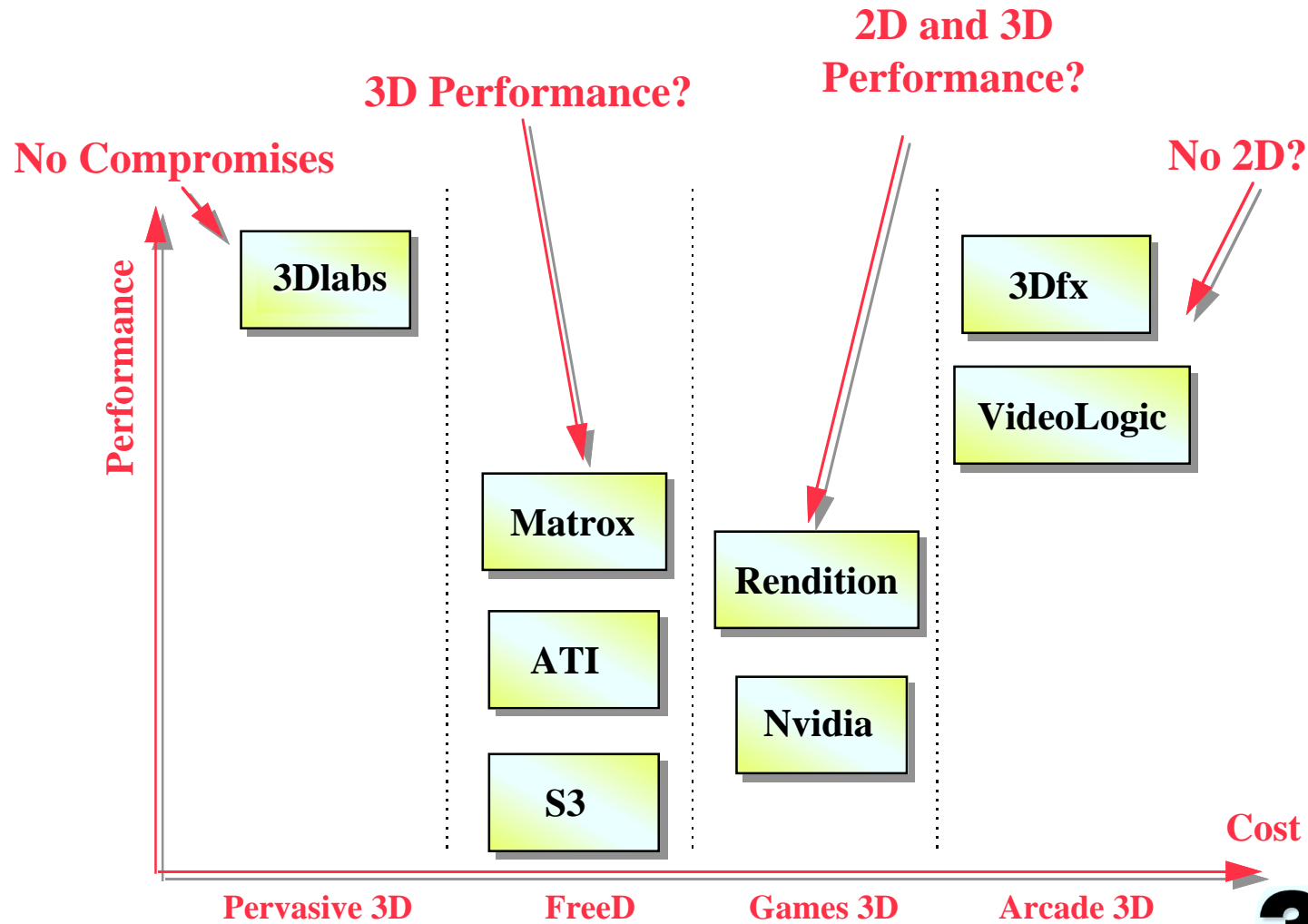
3Dlabs

3Dlabs New Generation Silicon



Alternative Approaches...

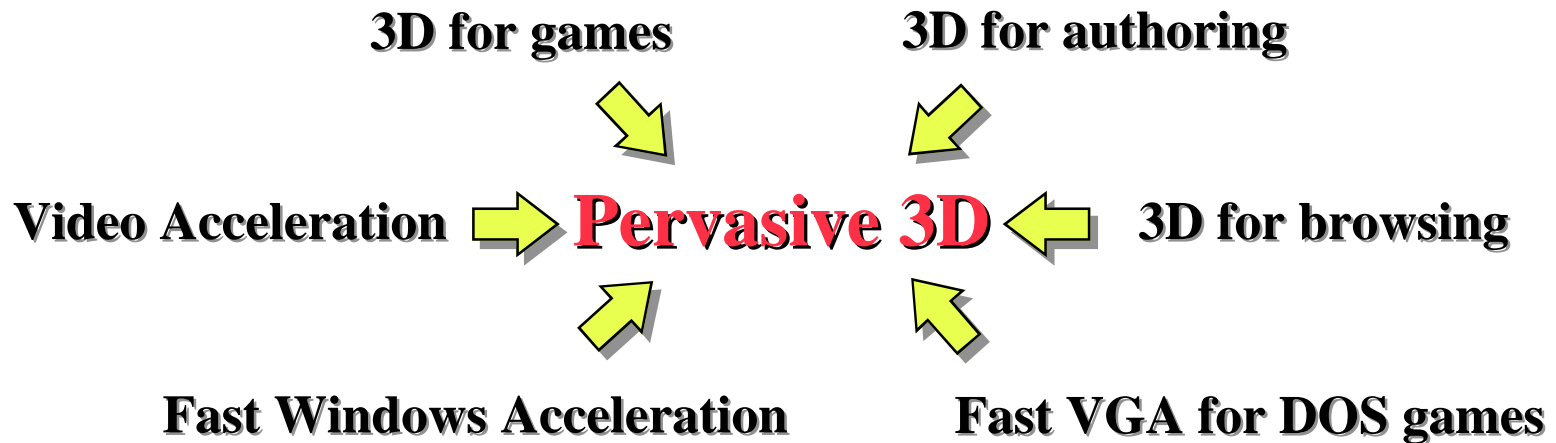
... to low-cost 3D silicon



3Dlabs

Pervasive 3D - the Challenge

No compromises!



- 3D performance must be much greater than software only
 - Software = 5 million texture-mapped pixels per second
 - Hardware should deliver >25 million bilinear-filtered texture-mapped pixels per second

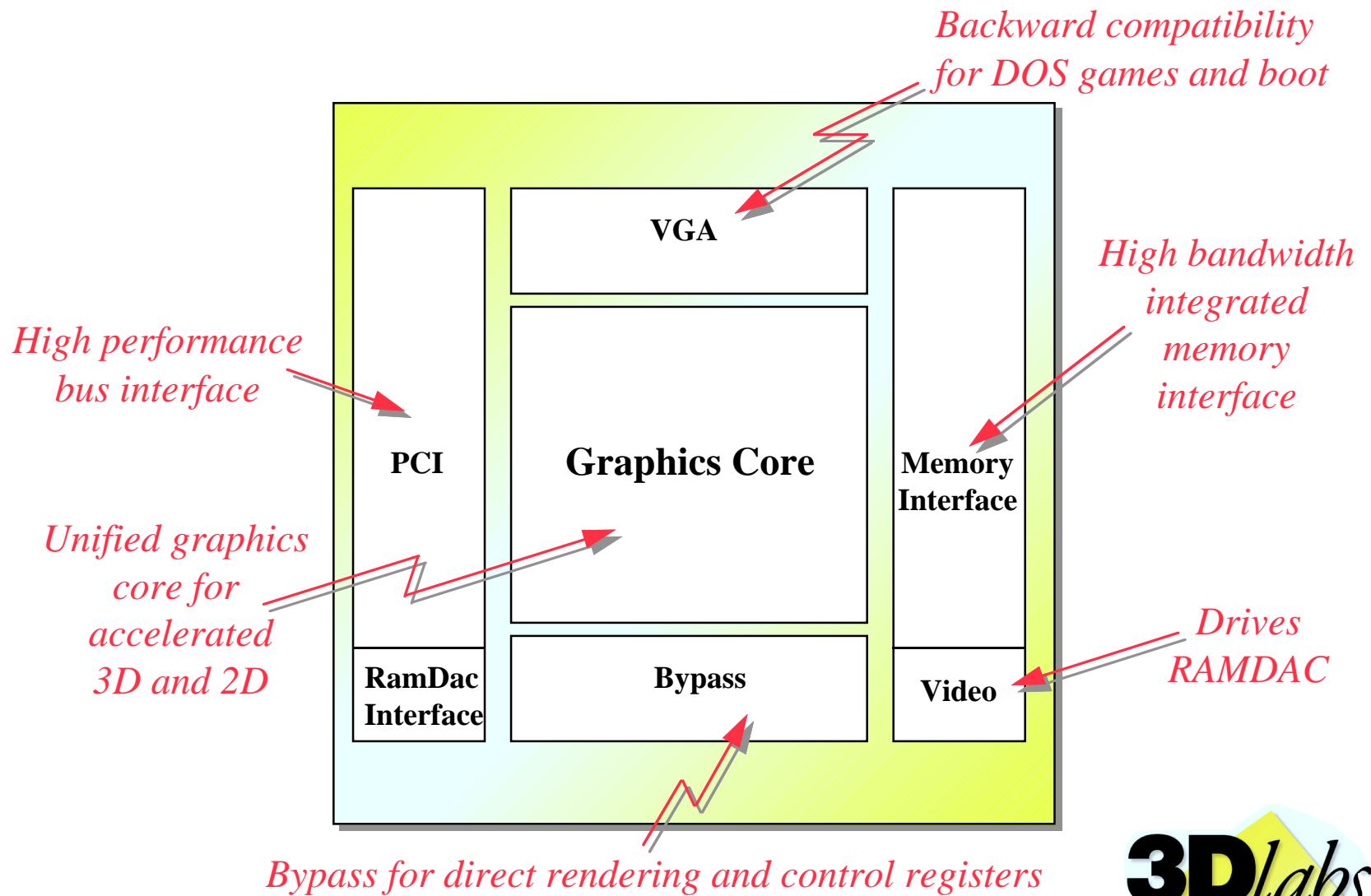
3Dlabs



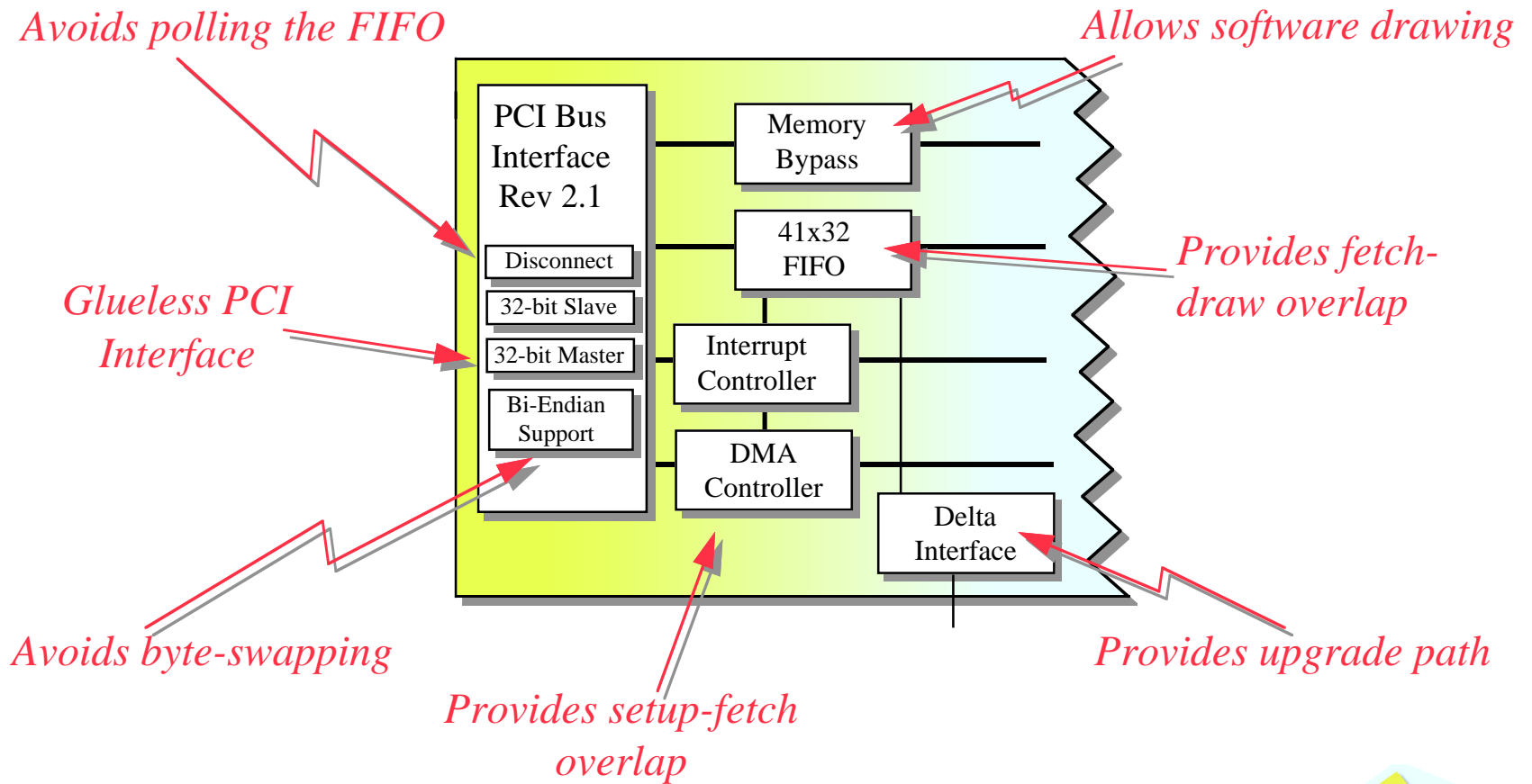
PERMEDIA Design Targets

- Robust 100% pixel functionality of all key 3D APIs
 - Direct 3D, OpenGL, Heidi, QuickDraw 3D, QuickDraw 3D RAVE
- No 2D compromises
 - >30 Million Winmarks
- Fast 3D performance
 - Balanced performance for both textures and polygons
 - 600,000 textured polygons/second
 - 30 Million bilinear filtered texture-mapped pixels /second
- Low cost
 - Selling on boards costing <\$200 (2MBytes)

PERMEDIA Architecture

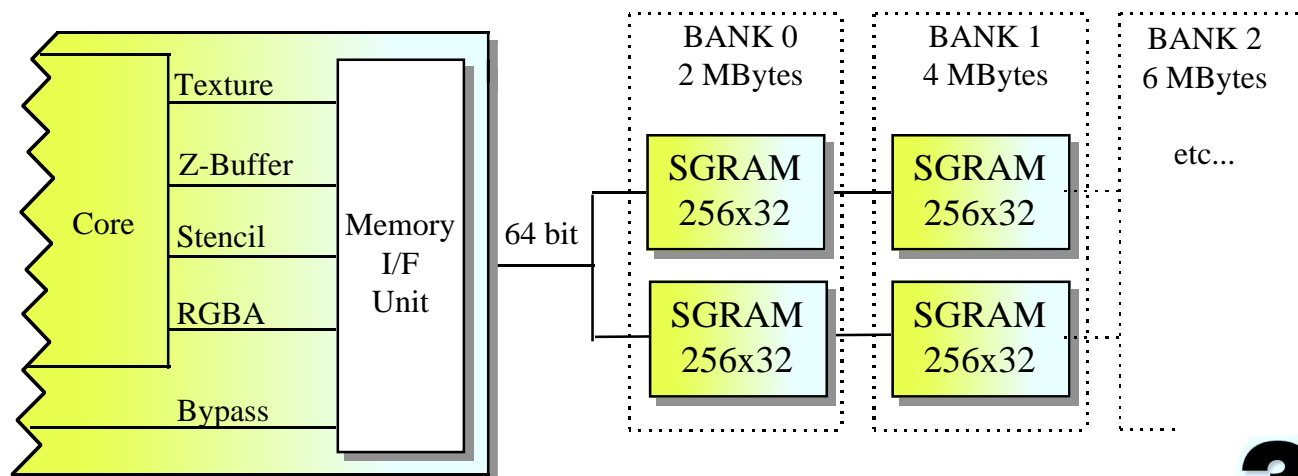


PERMEDIA Host Interface



PERMEDIA Memory Interface

- SGRAM for next generation graphics
 - Good random access speed - vital for texture mapping
 - Block fills - very important for clearing buffers
 - Write-per-bit mask, needed for per-window double buffering
 - Upgrade path to 100 MHz and beyond
- 2 to 8 MBytes
 - Up to 4 pages open at any time
 - E.g. front color buffer, back color buffer, depth buffer, texture buffer

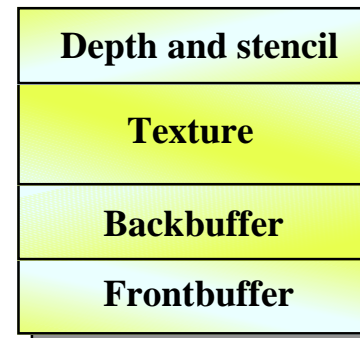


3Dlabs

Consolidated Memory

All buffers in same physical memory

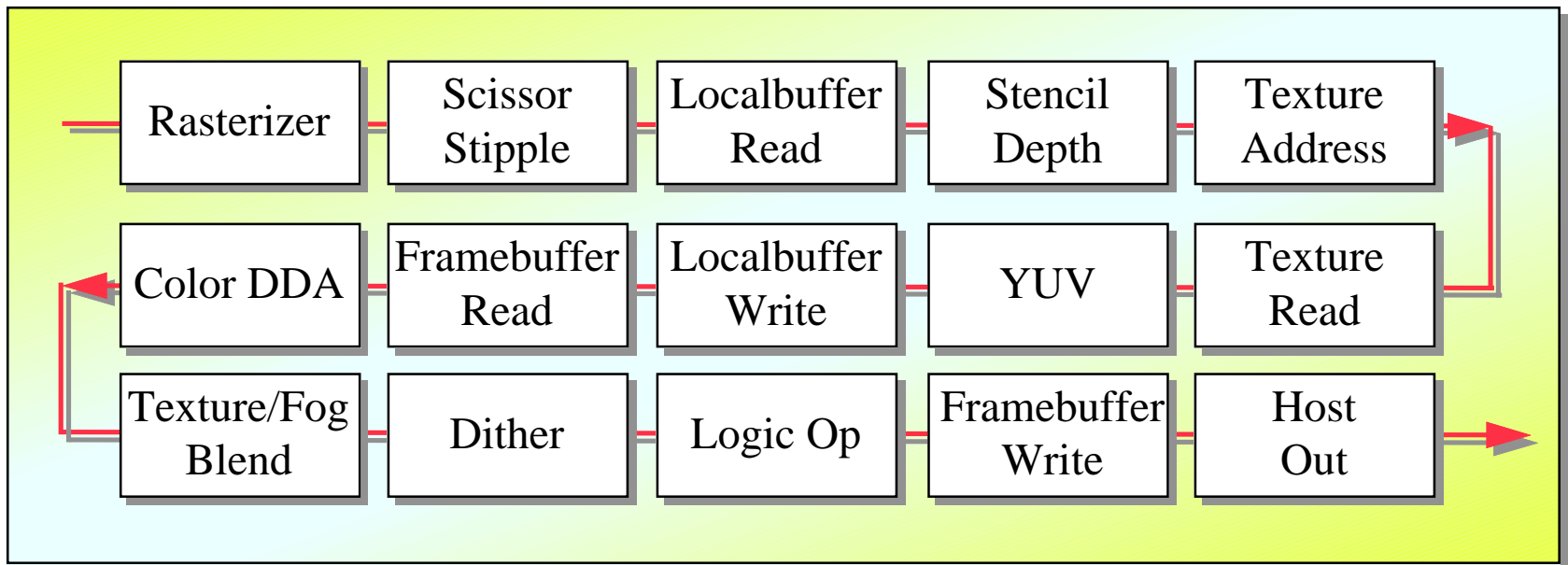
- Efficient and Flexible
 - Dynamically allocate color and depth buffers
 - Any spare memory available for textures
 - Trade resolution for depth buffer, color depth for texture space etc.
- All data in same memory, scope for optimization, e.g.
 - Clear depth buffer with framebuffer block fills
- Use texture operations on *any* image
 - Full scene anti-aliasing
 - Video texture-mapping
 - 3D sprite processing



3Dlabs

PERMEDIA Pixel Core

- Hyper-pipelined function units
- Message passing protocol between units



PERMEDIA Core

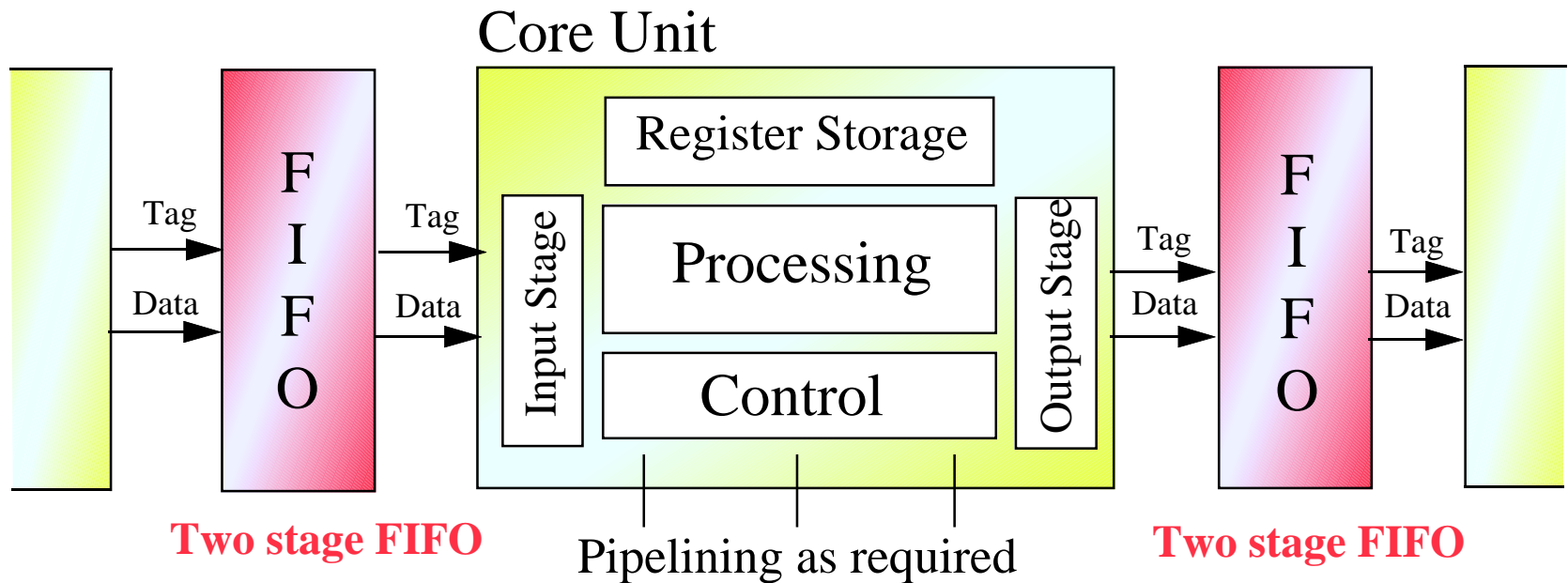


Pipeline Principles

- Each unit in the pipeline is independent
 - Can be designed, tested and synthesized separately
- Unit State Machine
 - Wait for message in input FIFO
 - If message is not relevant, pass to next unit
 - Else process message and pass on any messages as required
 - Return to waiting
- Some units know their place
 - Some units are completely self contained
 - E.g. scissor/stipple unit
 - Some units know where they are in the pipeline
 - E.g. YUV can absorb localbuffer data if the chroma test fails

Unit Pipeline Stage

- The pipeline uses a message passing paradigm
- A message is made up of a tag field and a data field
 - The tag identifies the message type



Two stage FIFO

Two stage FIFO

Tag = 9 bits

Data = 32 bits



Message Passing

- Everything that moves through the pipeline is a message
 - Messages are used to program control registers - e.g. enable texture
 - Messages are used to carry transient information e.g. texture color for current pixel
 - Messages are used as commands - e.g. start new primitive
 - Messages are used for synchronization - e.g. Sync message
- ‘Step’ messages drive the units
 - A Step message for each pixel to be plotted
 - Passive steps are pixels not to be plotted
 - If a pixel fails a test it is converted from active to passive
 - Passive steps cannot be deleted because they advance DDA units
 - Step messages hold the pixel X,Y coordinate in the data field



Unified 2D/3D Pixel Engine

- 3D is a superset of 2D
 - Don't separate them
- 2D operations use 3D pipeline and use special features
 - Texture units used for tiled blits
 - Chroma key test used for transparent blits
 - Bilinear filter used for stretch blits
- Using the 3D units is gate efficient
 - No duplication of functions
 - No compromise on performance

PERMEDIA Performance

- 30 Mpixels/sec, 600K polygons/sec, textured, bilinear, no Z
 - With full per pixel perspective correction, 16-bit framebuffer, 4-bit palletized textures, 50 displayed pixels per polygon, meshed, 640x480 at 75Hz
- 640x480 full screen bi-linear textured, x2.5 depth complexity = 40Hz frame rate
- 2 GBytes/sec Fill rate using SGRAM block fill
 - 2 GBytes/sec Color expansion
- > 30 Million Winmarks
- Video Playback performance - 30fps
 - 320x200 YUV source zoomed and filtered to 640x480x16-bit RGB

PERMEDIA Physical Characteristics

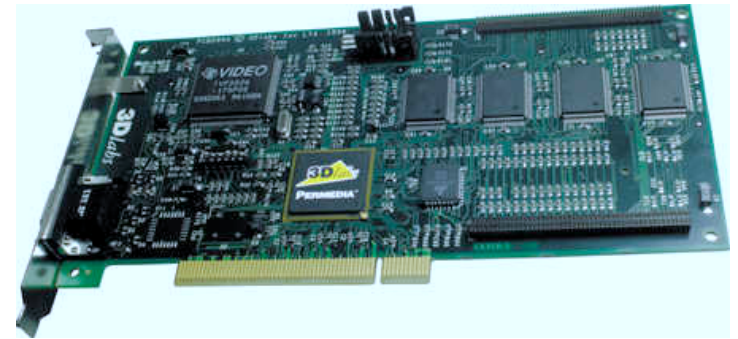
- Packaging
 - 256 pin BGA
 - Wire-bonded into a plastic BGA package
 - 3W at 3.3V
- Process
 - 0.35 μ , 4 layer metal
 - 60 MHz
- Shipping now



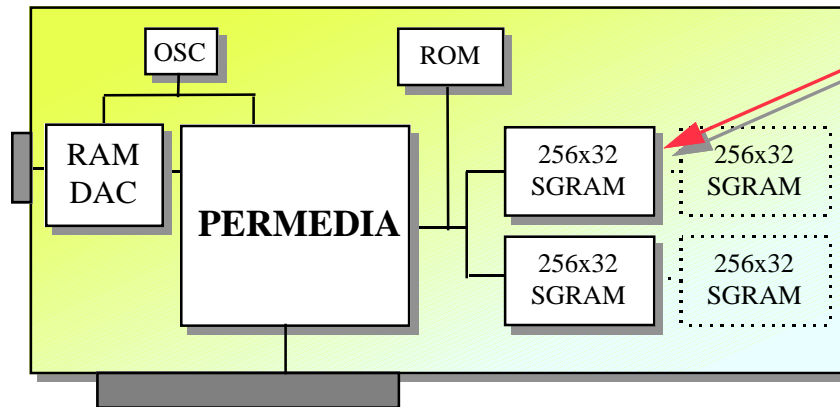
3Dlabs

Board Design

Low component count



- Single PERMEDIA Chip
 - plus SGRAM, RAMDAC, ROM
- External interfaces
 - Glueless PCI Interface
 - High performance 64-bit SGRAM Interface
 - High speed pixel port to RAMDAC



*Typically 2MBytes,
with optional
upgrade to 4MBytes*

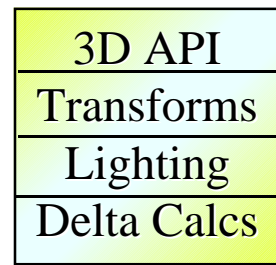
3Dlabs

But where's the bottleneck?

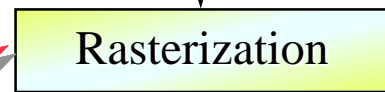
Geometry!

- The fastest Pentium Pro cannot keep PERMEDIA saturated if running the geometry in software

*1K polygons/MHz on a Pentium
Class machine
(90K polygons on a P5/90)*



*70% of the
CPU cycles
spent in setup!*

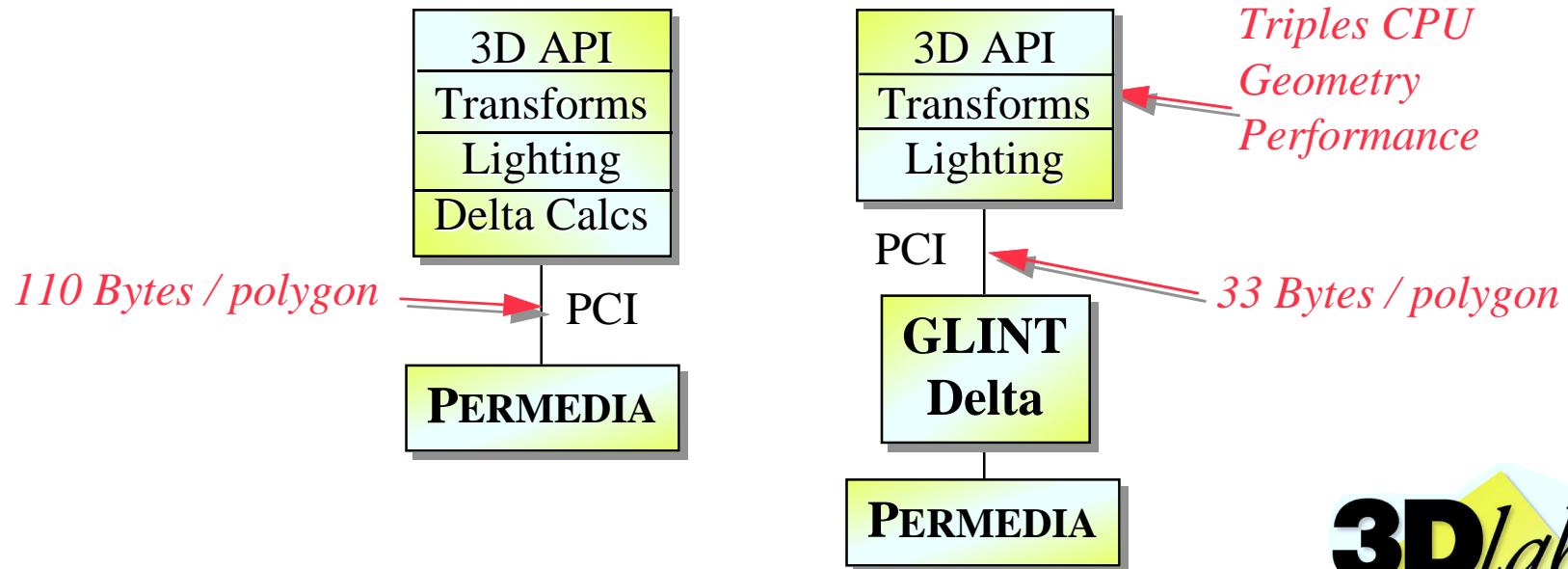


*100% of
Rasterization in
PERMEDIA
silicon*

GLINT Delta

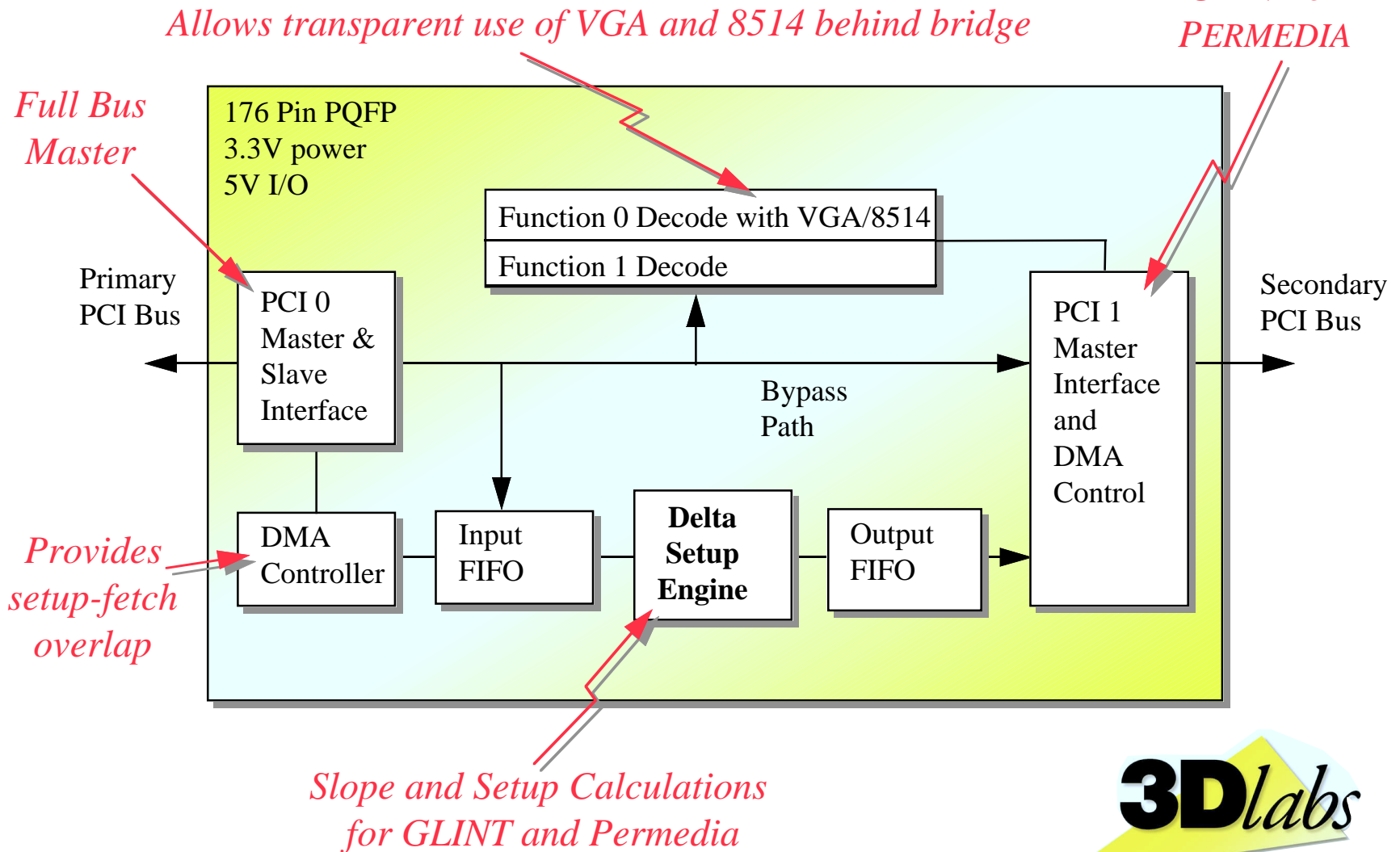
Breaking the Geometry Bottleneck

- Hardwired 3D Pipeline Processing
 - 1M vertex/sec Vertex Setup Processor
 - Performs all delta calculations and floating point conversions
 - 100 MFlop floating point processor
- Reduces PCI Bandwidth - just passing vertices - no slopes



GLINT Delta

Setup Processing in a PCI Bridge



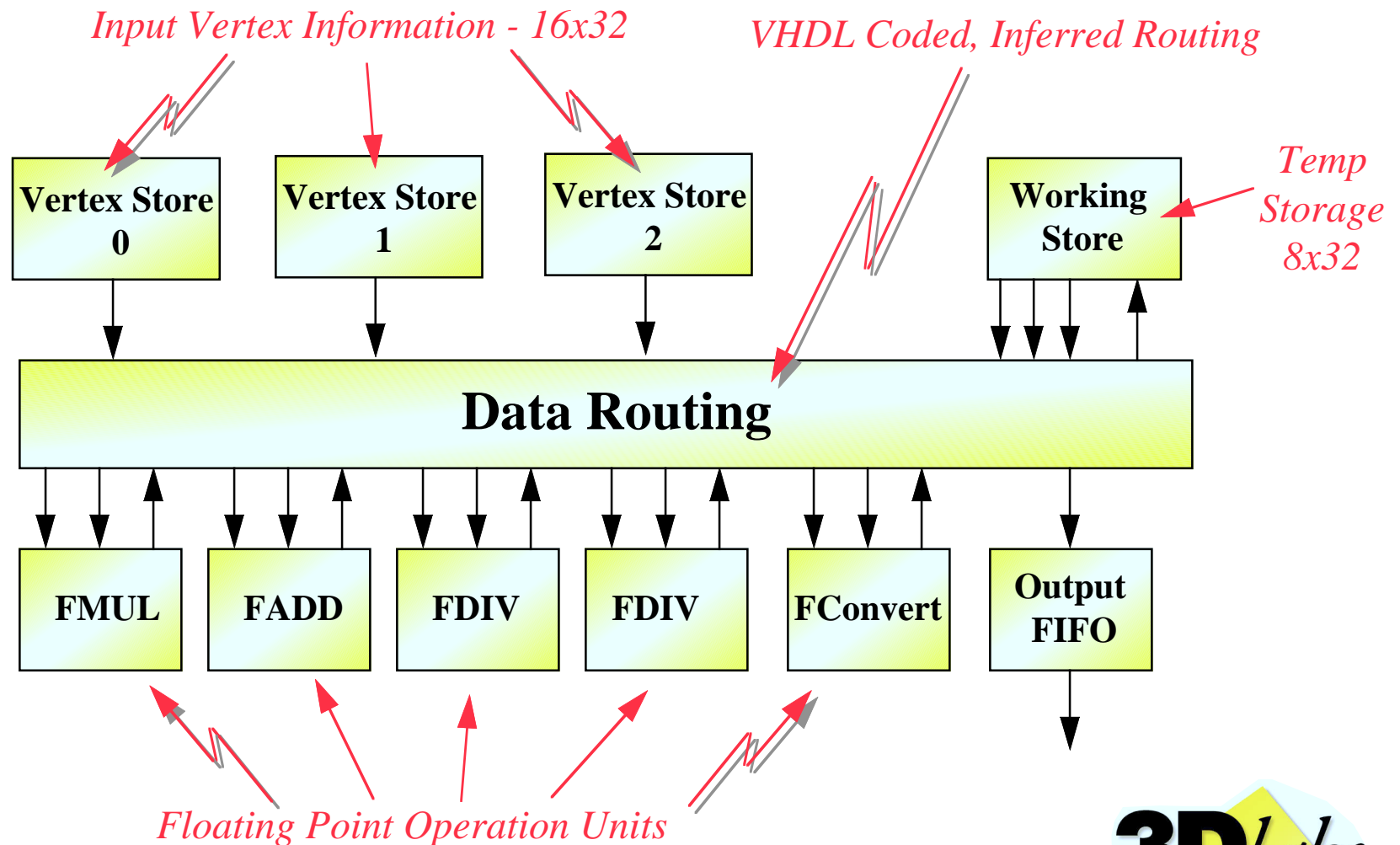
GLINT Delta

Setup Engine Functionality

- Follows the message passing architecture of PERMEDIA
 - Delta is just another unit in front of the rasterizer
- API neutral - low-level functionality
- Triangle primitive setup (AA and non-AA)
- Line primitive setup (AA and non-AA)
- Interpolation Parameters - XYZ, RGBA, F, STQ, Ks, Kd
- Accepts floating point (IEEE SP) or fixed point inputs
- Texture coordinate auto normalization
- Optional input value clamping
- High precision sub-pixel correction

GLINT Delta Setup Engine

Hardwired processing



GLINT Delta Calculations

Floating Point improves robustness and visual quality

- Input parameter score-boarding
- All internal calculations in custom floating point format
 - Less dynamic range, but more precision than IEEE
- RGBAZ triangle set-up involves:
 - 41 floating point add or subtract
 - 27 floating point multiplies
 - 5 floating point divides
 - plus.. compares, clamping, fixed point/floating point conversions
- Main floating point operators are:
 - One multiplier (one pipeline stage, single cycle).
 - One adder/subtractor (single cycle)
 - Two dividers (5 cycle iterative, autonomous)
 - Four comparators
 - Float to fixed point conversion with clamping

Hard-Wired Processing

Cost-effective floating point performance

- Control is a VHDL state machine.
 - No RAM or ROM for program storage (less gates)
 - No program sequencer or instruction set (less gates)
 - No program fetch (less memory bandwidth)
- Data paths are inferred directly from VHDL
 - No general purpose routing costs
- No software maintenance
- 35 cents / MFlop

GLINT Delta

Physical Characteristics

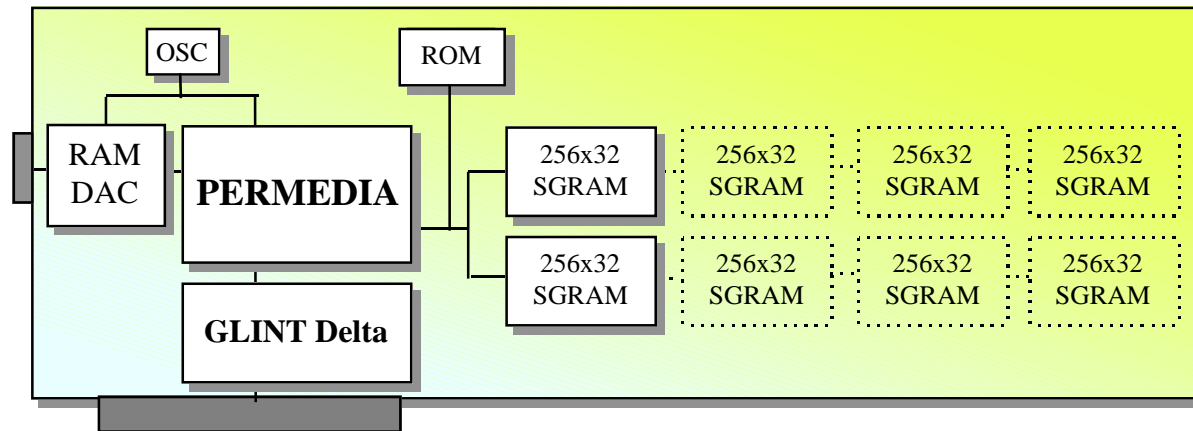
- Low cost device - 176 pin PQFP
- .45 μ , 40MHz, 3 layer metal
- Shipping now
- Performance
 - 1M Meshed Shaded, Z buffered triangles/sec
 - 2M 2D polylines/sec



Combined Board Design

Delta and PERMEDIA

- Matched Geometry and Rasterization performance
- High performance Arcade machines, VR/simulation engines, entry-level desktop OpenGL acceleration
- Sub \$350 street price



GLINT Delta

Measured Performance Increases

| Tspeed3 V3.0 OpenGL | No Delta | With Delta | X Faster With Delta |
|---|----------|------------|------------------------|
| Meshed Triangles (Z, Shaded) 50 Pixel per second | 155,146 | 238,997 | 1.54 |
| Meshed Triangles (Z, flat) 50 Pixel per second | 205,870 | 321,247 | 1.56 |
| Meshed Triangles (Z, Shaded) 25 Pixel per second | 180,744 | 427,242 | 2.36 |
| Meshed Triangles (Z, flat) 25 Pixel per second | 232,398 | 573,212 | 2.47 |
| Meshed Triangles (Z, Shaded) Small Triangles per second | 187,454 | 599,762 | 3.20 |
| Meshed Triangles (Z, flat) Small Triangles per second | 249,629 | 586,527 | 2.35 |
| Meshed Triangles (Z, Shaded) Single Pixel Triangles per second | 187,454 | 600,476 | 3.20 |
| Meshed Triangles (Z, flat) Single Pixel Triangles per second | 249,629 | 586,527 | 2.35 |
| Meshed Triangles (No Z, Shaded) 50 Pixel per second | 182,048 | 277,016 | 1.52 |
| Meshed Triangles (No Z, flat) 50 Pixel per second | 223,155 | 365,412 | 1.64 |
| Meshed Triangles (No Z, Shaded) 25 Pixel per second | 199,290 | 514,781 | 2.58 |
| Meshed Triangles (No Z, flat) 25 Pixel per second | 272,531 | 585,847 | 2.15 |
| Meshed Triangles (No Z, Shaded) Small Triangles per second | 200,159 | 646,607 | 3.23 |
| Meshed Triangles (No Z, flat) Small Triangles per second | 271,068 | 586,527 | 2.16 |
| Meshed Triangles (No Z, Shaded) Single Pixel Triangles per second | 200,079 | 646,607 | 3.23 |
| Meshed Triangles (No Z, flat) Single Pixel Triangles per second | 271,214 | 586,527 | 2.16 |



Future Directions

- More Geometry Pipeline in Hardwired Logic
 - CPUs just aren't fast enough
 - Hardwired logic is more cost-effective
- Unified Memory
 - Using system memory for texture
 - Intel's AGP - Accelerated Graphics Port
- 3D Graphics on the Motherboard
 - High integration - RAMDACs and geometry included on-chip
- Aggressive Performance Increases
 - Next generation silicon - single chip million polygon devices
- Major silicon vendors entering graphics chips market