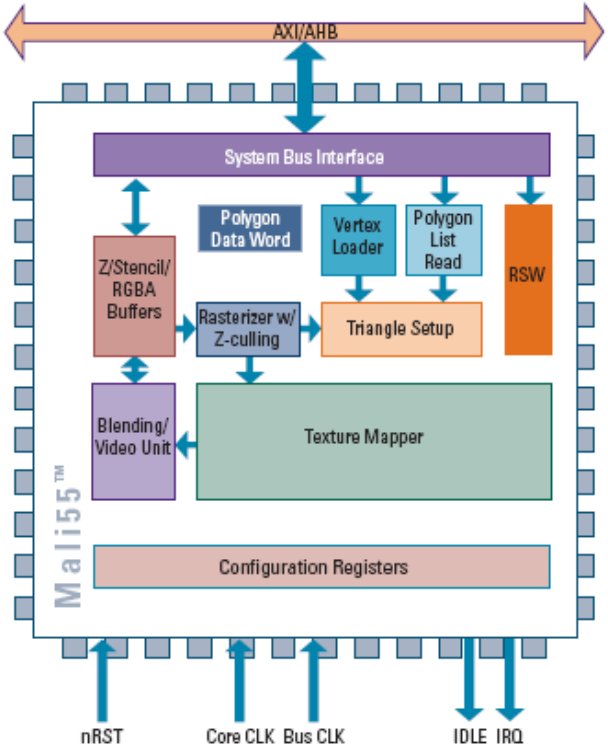
Overview of ARM Mali embedded graphics solution

Yi Yong-il

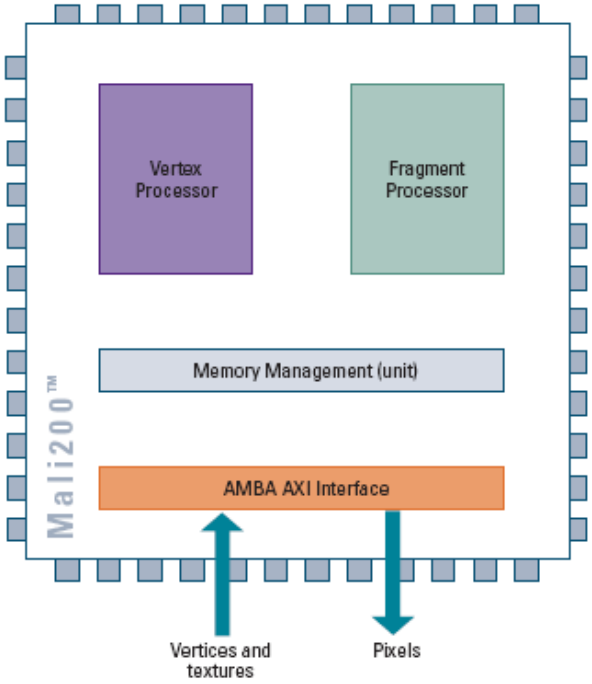
Mali graphics solution

- There are two products
  - Mali 200
    - Console-quality gaming
    - High geometry & pixel processing
    - Full OpenGL ES 2.0 & OpenVG 1.1 support
  - Mali 55
    - General purpose
    - Optimized pixel rendering & software geometry processing
    - Ultra small, cost-effective implementation
    - OpeIGL ES 1.1 & OpenVG 1.0 support

Mali graphics solution



Mali 55



Mali 200

Size @ 90nm	Geometry Performance	Pixel Performance
1.4mm²	1M tri/s	100M pix/s

Size @ 90nm	Geometry Performance	Pixel Performance
7mm²	9M tri/s	275M pix/s

Feature

- 3D Graphics
  - 4X / 16X FSAA
  - Flat / Gouraud Shading
  - Perspective Correct Texturing
  - Point Sampling / Bilinear / Trilinear Filtering
  - Mipmapping
  - Multi Texturing
  - Dot3 Bump Mapping
  - Alpha Blending
  - Stencil Buffering (4-bit)
  - Point Sprites
  - 2 bit per texel Texture Compression (FLXTC)
  - 4 bit per texel Texture Compression (ETC)

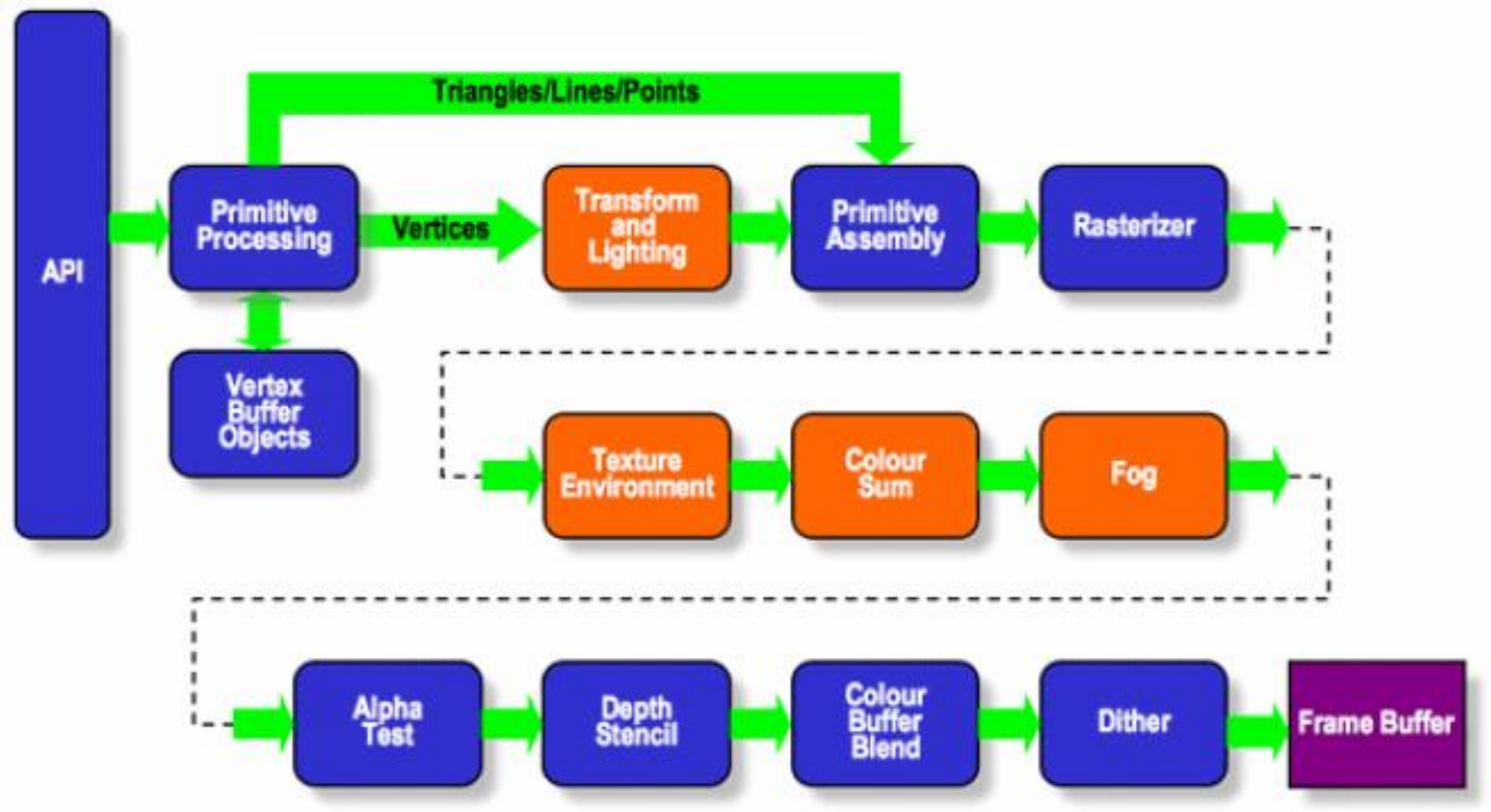
Feature

- 2D Graphics
  - Lines, Squares, Triangles, Points
  - Vector Graphics
  - Arbitrary Rotation / Scaling
  - Alpha Blending
  - Multitexture BitBLT

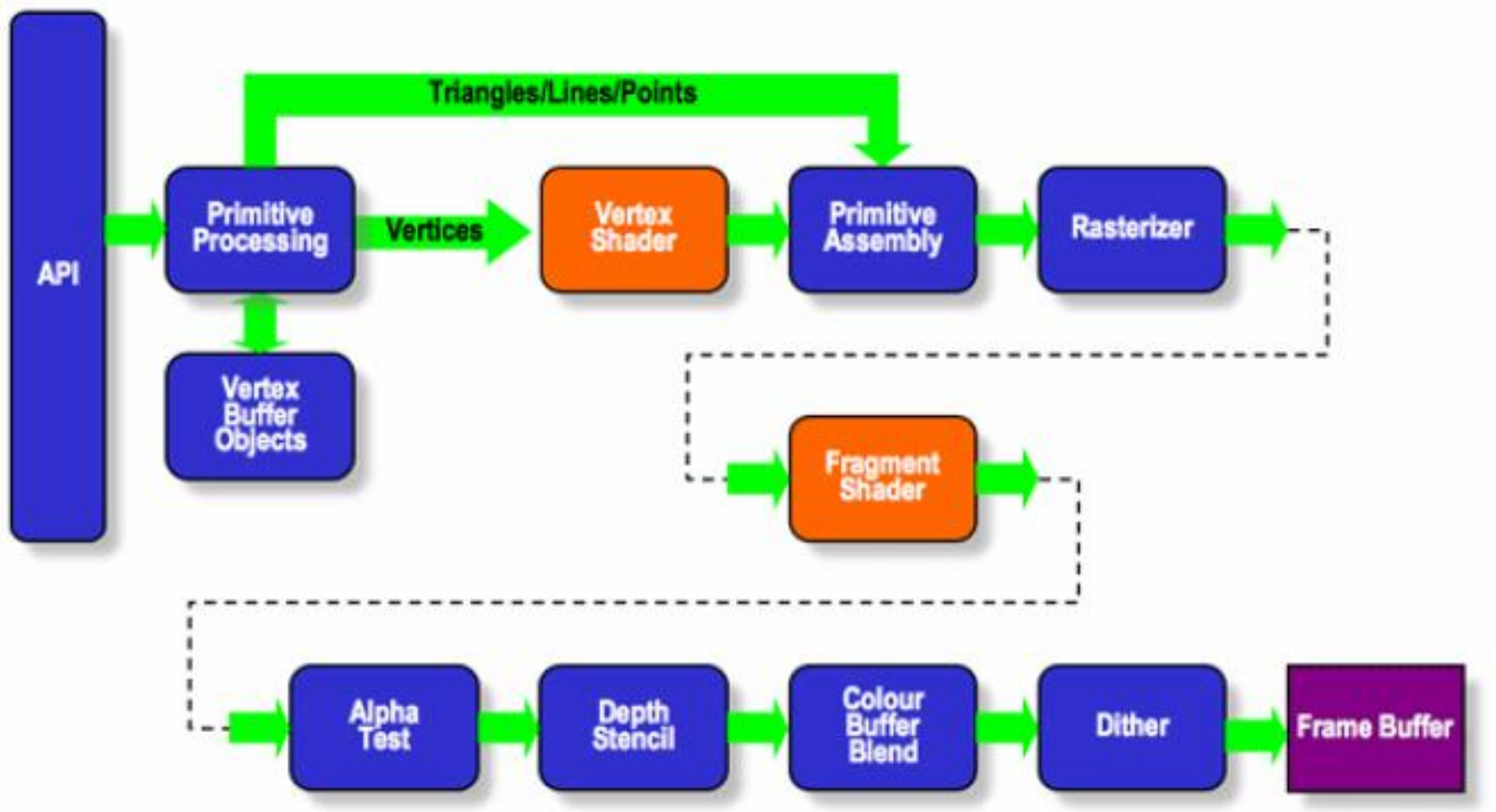
Rendering scheme

- Mali200 pixel processor + MaliGP2 programmable geometry processor
  - Complete OpenGL ES 2.0 support
- Tile based rendering
  - break the screen into tiles and render a tile at a time to an on-chip tile-memory
  - additional complexity compared with simple immediate-mode renderers
- Mali uses a unique blend of tile-based & immediate mode rendering

OpenGL ES 1.x fixed function Pipeline



OpenGL ES 2.0 Programmable Pipeline



Data Flow

- 3d objects are broken down into lists of triangles
- the vertices & commands for the MaliGP2 processor are written by the ARM processor to memory
- MaliGP2 processes the commands
- Mali200 processor then rasterizes
  - It reads data and renders a tile at a time

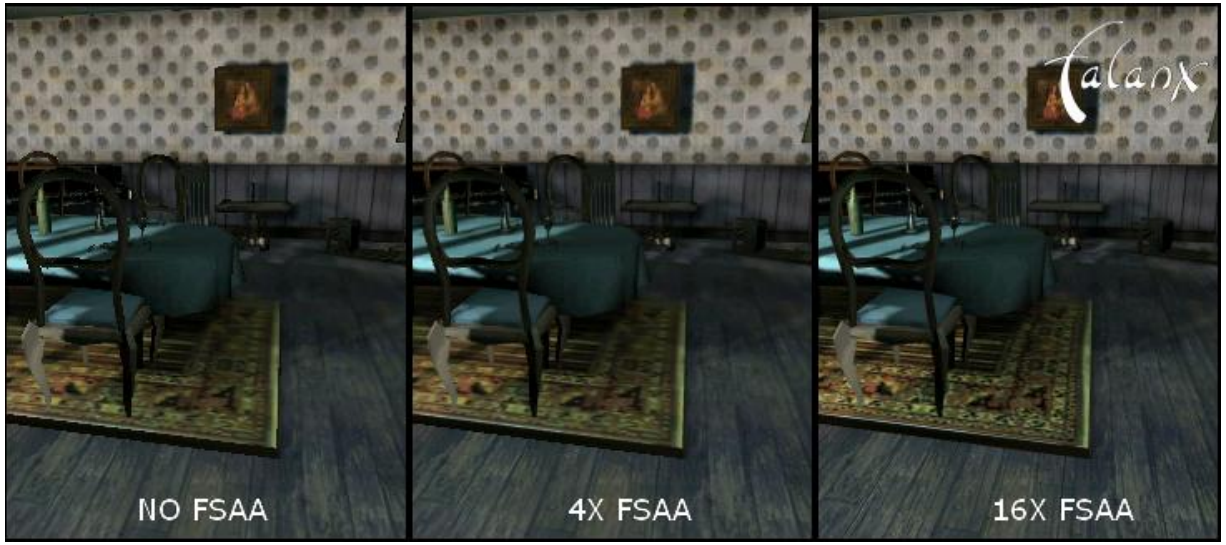
Image quality

- 4X, 16X FSAA(Full Scene Anti-Aliasing)

Image quality

- HDR (High Dynamic Range) Rendering
  - Use floating-point value for huge range of light value

Demos



Demos

Demos