GPUs móviles

Daniel Flores Tafur
Javier Sanchez Martinez

0. Abstract

En este artículo vamos a presentar la situación general en el mercado de las GPUs móviles. Presentaremos los aspectos más importantes de la arquitectura general y, también, cada una de las plataformas más importantes en el mercado.

1. La era post-PC

Cada vez más gente habla de así llamada "era post-PC", que se caracteriza fundamentalmente por dos cosas:

1. La maduración del mercado de los PCs "clásicos", donde ya no se observa el crecimiento que antes siempre caracterizaba este mercado.
2. La aparición del nuevo mercado de smartphones y tablets que tiene un crecimiento espectacular y a largo plazo va a afectar enormemente la industria actual.

De hecho, este nuevo mercado tiene los mismos rasgos que tenía el mercado de los PCs cuando ha aparecido a finales de la década de 1970. Entonces, el mercado de PCs se caracterizaba por la competencia entre varias plataformas y, incluso, arquitecturas distintas. A mediados de la década de 1990 han surgido dos claros ganadores de esta guerra de estándares y plataformas – la arquitectura x86 de los procesadores, inventada por Intel, y el sistema operativo Windows de Microsoft. Como el resultado, estas dos empresas se han convertido en, prácticamente, monopolistas del sector y han obtenido beneficios enormes comparando con los beneficios obtenidos por la competencia.

Entrando en la década de 2000, poco a poco aparecen los nuevos tipos de ordenadores móviles – las tablets, entonces caracterizadas por el uso de Windows como su sistema operativo y la misma interfaz que en un ordenador de mesa o un portátil. Por otro lado, en el terreno de los teléfonos móviles han empezado a aparecer modelos cada vez más sofisticados, que permitían conectarse al Internet, instalar programas y se caracterizaban por tener un sistema operativo cada vez más avanzado.

No obstante, este mercado de las tablets y los teléfonos avanzados era un mercado muy específico, con una base de clientes formada, principalmente, por sector empresarial y que se caracterizaba por precios muy altos y funcionalidad bastante restringida. El verdadero paso adelante se ha dado en el año 2007 con la aparición del iPhone de Apple y, más tarde, con la aparición de su competencia en forma de los teléfonos con Android, sistema operativo de Google. En 2010, con la salida de iPad de Apple, que se ha convertido en una tablet con una popularidad sin precedentes, ya se ha consolidado el hecho de que el mercado de la informática ha entrado en una nueva era que ahora llamamos "era post-PC".

La era post-PC no se supone la desaparición de los PCs clásicos. Lo único que refleja es que el mercado de los PCs clásicos se ha madurado y no va a tener ya un crecimiento tan amplio como ha tenido hace una década. En cambio, el mercado de los smartphones y las tablets tiene un crecimiento espectacular, comparado con el crecimiento del mercado de los PCs cuando este apareció. Este hecho, seguramente, va a afectar a toda la industria dado que las tablets tienen rasgos muy distintos de los PCs, principalmente la necesidad de ahorrar constantemente la energía. Esto lleva a que todos los componentes de una tablet o un smartphone deben ser muy eficientes. Por otro lado también afecta al diseño del sistema operativo, que, por ejemplo, deja en el segundo plano el multitasking en favor de la filosofía "una aplicación por una pantalla". También, teniendo en cuenta que, posiblemente, en el futuro habrá más tablets y smartphones que los PCs clásicos, es posible que el número de desarrolladores de aplicaciones móviles vaya a ser mayor que el número de desarrolladores de aplicaciones para PCs de sobremesa o portátiles. Esto, seguramente, va a afectar muchísimo el estilo de programación y los metodologías de desarrollo actuales.

En cuanto al tema de los aceleradores gráficos, cabe destacar que cada vez tienen más importancia en un dispositivo móvil. Actualmente los dispositivos móviles de alta gama se caracterizan por tener resoluciones de pantalla enormes, que a veces sobrepasan la resolución típica de un ordenador de sobremesa. Así, por ejemplo, la resolución de la pantalla Retina Display, utilizada en el iPad de cuarta generación, es de unos 2048 por 1536 pixeles. A parte de esto, muchos dispositivos móviles tienen en su portafolio de aplicaciones los juegos 3D que tienen cada vez gráficos más realistas, que se aproximan en calidad a gráficos a juegos de consolas o incluso juegos de ordenador de sobremesa.

Entonces, un acelerador grafico móvil debe resolver una tarea de complejidad muy alta – por un lado debe ser muy eficiente y por otro debe proporcionar una potencia cada vez mayor. Por lo tanto es un mercado muy dinámico y con grado de competencia e innovación muy alto. A diferencia del mercado maduro de los aceleradores gráficos para PCs, que tiene, principalmente, 2 jugadores importantes – AMD y NVidia (y, también, quizás, Intel con sus aceleradores en el mismo chip con el CPU), el mercado de las GPUs móviles tiene, como mínimo, 5 jugadores de importancia elevada a parte de algunos más pequeños, pero con mucho potencial.

Cabe destacar, que aunque el mercado está representado por plataformas, o sea por una combinación de CPU + GPU, nosotros solo centraremos en la parte de los GPUs sin entrar mucho en las particularidades de la arquitectura de un CPU móvil.

2. Las APIs para GPUs móviles

Las principales APIs para aceleradores gráficos de ordenadores son DirectX de Microsoft y OpenGL de Khronos Group. Para los aceleradores móviles, la API más utilizada es OpenGL ES (OpenGL for Embedded Systems). Como hemos visto durante el curso, los aceleradores son una implementación en hardware de lo que definen los APIs. Para aceleradores móviles, este principio se conserva. No obstante, hay que destacar, que a diferencia de los aceleradores para ordenadores, algunos aceleradores móviles pueden soportar una determinada versión de OpenGL ES de un modo incompleto, lo que nunca pasa con las versiones de APIs para ordenadores. Por otro lado, hay que destacar que últimamente muchos aceleradores móviles a parte de OpenGL ES también soportan las versiones "base" de OpenGL y algunos, también, DirectX de Microsoft, con lo que, efectivamente, no hay diferencia en API entre estos aceleradores móviles avanzados y un acelerador para un ordenador de sobremesa. Probablemente, en el futuro no tan lejano, los aceleradores móviles y los aceleradores para ordenadores siempre utilizaran la misma API.

No obstante, por ahora nos centraremos en OpenGL ES que sigue siendo la API estándar para la mayoría de los GPUs móviles.

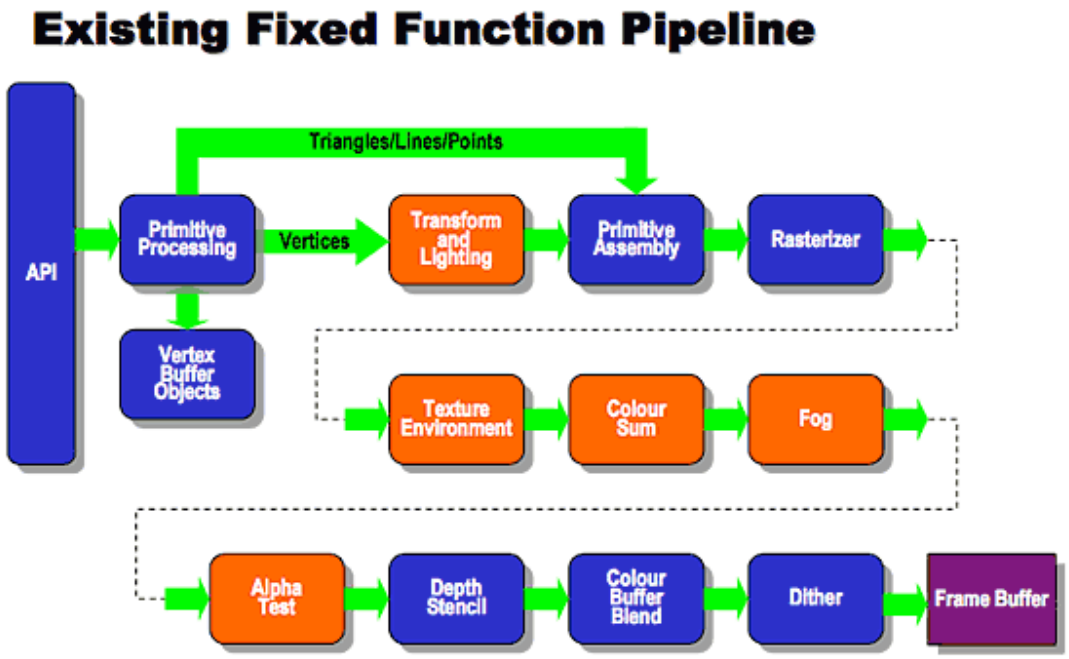
Generalmente, OpenGL ES es lo mismo que OpenGL estándar, pero con ciertas limitaciones que reflejan su naturaleza de una API adaptada para aceleradores móviles.

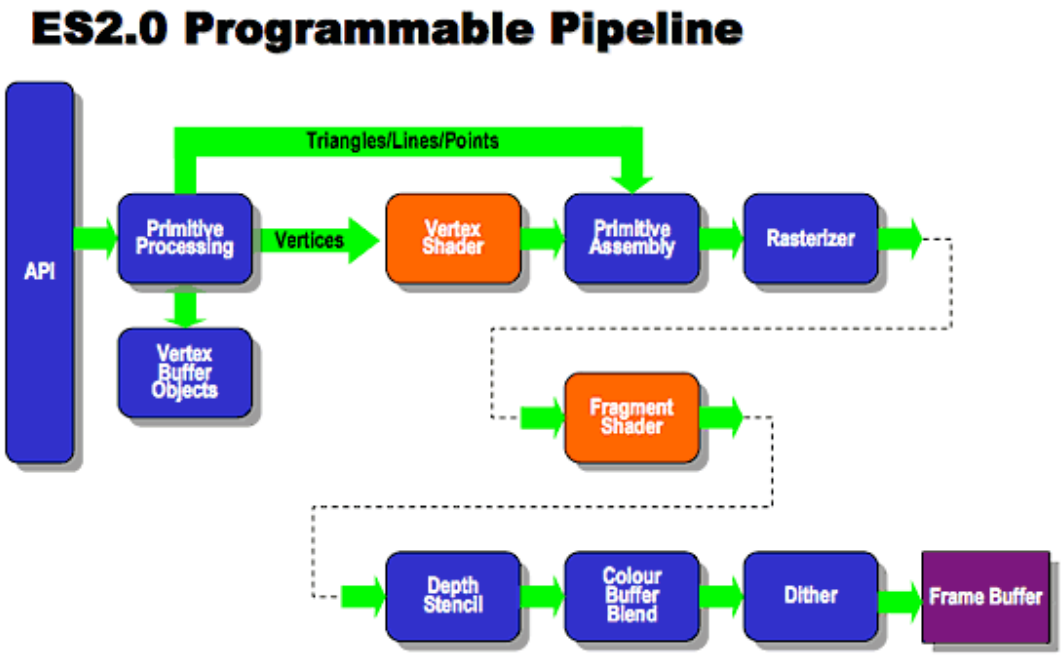
La primera versión, Open GL ES 1.0 es una versión reducida de OpenGL 1.1. Entre otras reducciones, esta adaptada para procesadores que no tienen FPU (unidad de procesamiento en coma flotante) y no tiene funciones con alta carga computacional y en el sistema de memoria, como ciertos modos de renderizado (por ejemplo, el renderizado con antialiasing). Tampoco estaban soportadas funciones avanzadas como texturas en 3D, operaciones con bitmaps y listas de display y feedback. Como podemos ver, OpenGL ES 1.0 es una API introducida antes de la era post-PC, y se puede ver que refleja las capacidades relativamente modestas de aceleradores de aquella época. Es una interfaz muy ligera y, también, básica. Todavía se utiliza en algunos dispositivos de gama baja.

La siguiente versión era OpenGL ES 1.1 derivada de la OpenGL 1.5. Era más avanzada en cuanto a sus funcionalidades que OpenGL ES 1.0 y se apoyaba mucho más en la aceleración gráfica por hardware, mientras que OpenGL ES 1.0 se suponía el renderizado por software (utilizando la CPU) y una aceleración hardware muy básica. Esto ha permitido incluir algunas funciones que estaban excluidas de OpenGL ES 1.0. No obstante, la nueva versión era completamente compatible con la versión anterior. Más tarde salió el Extension Pack para OpenGL ES 1.1 que era opcional y tenía algunas funciones más avanzadas como cube maps, frame buffer objects, matrix palettes más grandes, y mejorados texturing y stencil modos. Con este pack también se mejoraba la portabilidad de aplicaciones, ya que se podía utilizar una

función de este pack que funcionaba en todas las implementaciones compatibles en vez una función para una implementación específica.

En el marzo del 2007 ha salido la versión OpenGL ES 2.0. Esta basada en OpenGL 2.0, pero también cambia el pipeline fijo por el pipeline programmable, como en OpenGL 3.1. Como el resultado, no es compatible con las versiones anteriores de OpenGL ES.





Como podemos ver, OpenGL 2.0 es una API mucho más versátil que APIs anteriores y que aprovecha las posibilidades que ofrecían los aceleradores gráficos más avanzados. En este punto muchos de los aceleradores ya estaban utilizando los shaders unificados, exactamente como los aceleradores para ordenadores, así que no nos debe sorprender el hecho de que esta API estaba utilizando una pipeline programable en vez de la pipeline fija.

Finalmente, en agosto del 2012 ha salido la especificación de OpenGL ES 3.0, la versión más moderna de esta API. Esta versión introduce mejoras y nuevas funciones en el pipeline para efectos visuales avanzados. También introduce el estándar ETC2/EAC de comprensión de texturas en alta calidad, el nuevo lenguaje de programación de shaders con soporte completo para operaciones en coma flotante y muchas otras funciones avanzadas, como texturas 3D.

En general, OpenGL ES es muy similar a la OpenGL para los ordenadores. Un programador familiar con OpenGL no debe tener ningún problema a la hora de pasar al OpenGL ES dado que su principio es exactamente el mismo que en OpenGL. Como en OpenGL, es una pipeline que se puede ver como una máquina de estados. Desde el punto de vista de programador, OpenGL ES es la colección de comandos que permiten especificar los objetos geométricos en dos o tres dimensiones junto con otros comandos que permiten definir como estos objetos se renderizan en el framebuffer.

Así, el típico programa escrito en OpenGL empieza con una serie de comandos para definir la ventana en la cual va a dibujar los objetos. Luego se define el contexto que se asocia a la

ventana. Después el programador ya puede utilizar los comandos para dibujar simples objetos geométricos, como puntos, líneas o polígonos. A parte de estos comandos, existen comandos que permiten definir la iluminación, el color y otros parámetros que afectan como se dibujan los objetos definidos. Finalmente, hay una serie de comandos que permiten leer o escribir los píxeles concretos.

Como podemos ver, no hay ninguna diferencia conceptual en esta API con la API para ordenadores y cualquier persona que ha programado en el OpenGL puede dar por hecho que ya sabe programar en OpenGL ES. La única diferencia está en que OpenGL ES tiene ciertas limitaciones respecto a la OpenGL "base", pero no hace falta hacer nada especial aparte de mirar el manual de referencia para saber qué es lo qué se puede hacer y qué no se puede.

3. Arquitectura de GPUs móviles

Si entramos en detalle de los procesadores de "smartphones" se observa que los núcleos de procesamiento reales son sólo una parte del total del sistema que constituye la base de todos los teléfonos modernos. Junto con dichos núcleos de procesamiento y otros subsistemas del SoC, también se encuentra la unidad de procesamiento gráfico o GPU, en una proximidad muy cercana al procesador.

El sistema encastado es un pequeño chip que se utiliza en la placa base de un smartphone, y como la GPU está en realidad dentro de este chipset, encontrar físicamente a la GPU mientras se observa el interior de un teléfono es casi imposible.

Esto es completamente diferente a un ordenador de sobremesa o portátil, que habitualmente se utiliza una solución de doble chip. Por ejemplo, la CPU conectada a la placa base y el procesador gráfico (GPU) está unido a una placa separada que se une después a la placa base. Los dos componentes fundamentales, en realidad, son físicamente muy lejos.

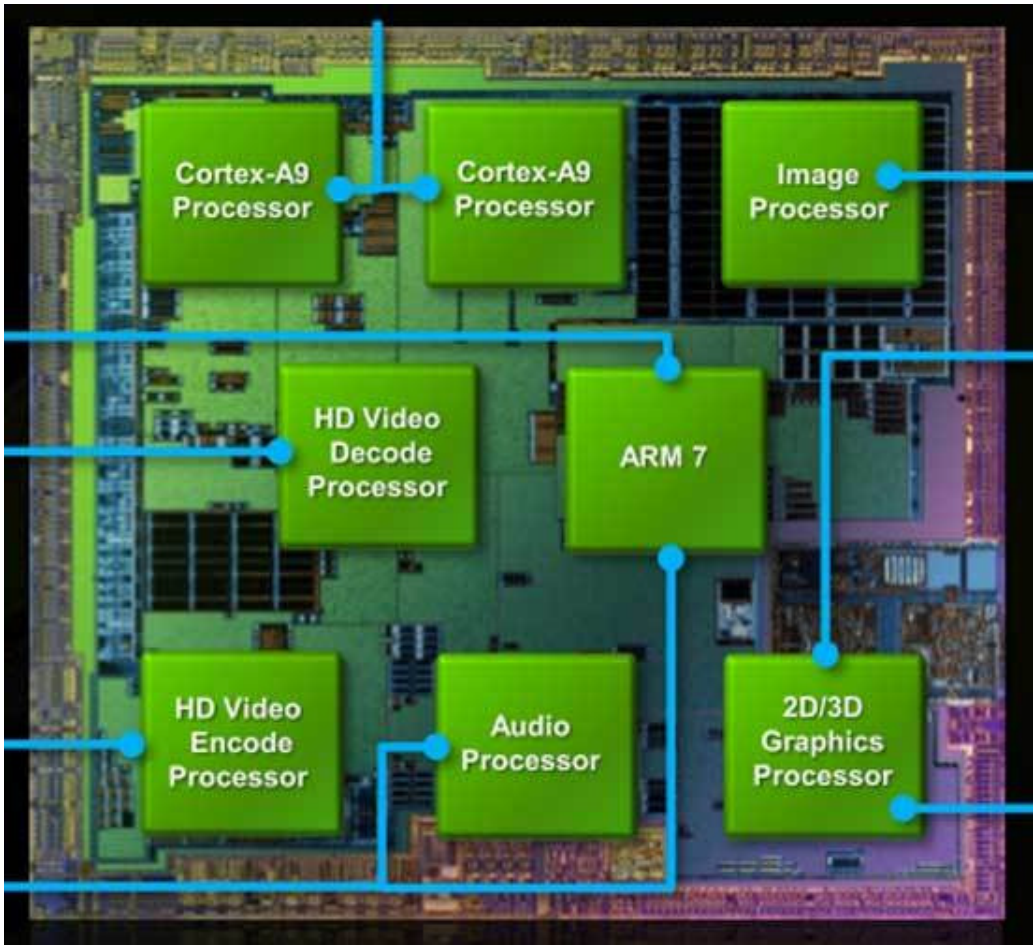
Por supuesto, hay una razón de por qué los dos chips en un smartphone se encuentran tan cerca. En primer lugar, no tienen una gran cantidad de espacio interior para trabajar, y así tener componentes críticos empaquetados permite que la placa base del dispositivo sea más pequeña y la batería a ser grandes. En segundo lugar, el encastado de las dos unidades como una sola reduce la salida de calor del dispositivo, ya que es más localizada y puede ahorrar energía a través de la estrecha integración de los dos. Por último, se ahorra los costes de fabricación para producir un chip en lugar de dos.

Uso de las GPU en dispositivos móviles

El uso de la GPU depende de varios factores: la estructura del sistema encastado y también el sistema operativo utilizado en el dispositivo. En el primer caso, si el SoC no tiene un chip de

codificación multimedia dedicado entonces la GPU puede ser utilizado para manejar videos de alta resolución. También existe la posibilidad de que las tareas compatibles se descarguen a la GPU, lo que permite reducir el trabajo y consumo de los núcleos de CPU más intensivos.

Cuando se trata del sistema operativo las cosas son mucho más complejas. En primer lugar la GPU se utiliza exclusivamente para toda la representación 3D en los juegos y aplicaciones. Los núcleos de procesamiento Cortex simplemente no están diseñados para manejar este tipo de tareas y en todos los sistemas operativos la GPU se hará cargo de la CPU para manejar la prestación más eficiente. La CPU le ayudará a cabo para ciertos cálculos, mientras que la prestación de los modelos 3D en la pantalla (sobre todo para los juegos), pero el trabajo principal será realizado por el chip gráfico.



En cuanto a la renderización hay dos extremos "Immediate-Mode-Rendering" (IMR) y "Tile-Based-Deferred-Rendering" (TBDR). Entre estos extremos se encuentra el "Tile Based Immediate Mode Rendering" (TBIMR), que se utiliza por absolutamente todas las GPUs excepto PowerVR, que utilizan TBDR.

Las primeras GPU se basaban en IMR: aquí la CPU establece los parámetros necesarios directamente en la GPU y un acceso de escritura a un cierto registro desencadenado el inicio del proceso de renderizado. La rasterización triángulo se realiza en un solo paso y la CPU tiene que esperar hasta que la GPU está lista de nuevo para emitir los siguiente triángulo, obviamente, este enfoque provoca una sobrecarga de sincronización tanto en la CPU como en la GPU.

En todas las GPU modernas la rasterización se realiza con un enfoque basado en mosaico. Los "SW-driver" almacenan tanto como sea posible de la escena y renderizan todos los triángulos sección por sección en el framebuffer. El "tile-buffer" se encuentra en el chip de memoria, lo que conduce a reducir el consumo de ancho de banda de memoria externa.

	Adreno	GeForce ULP	Mali-400	Mali T604-	PowerVR
Unified Shader	yes	no	no	yes	yes
Render Mode	TBIMR	TBIMR	TBIMR	TBIMR	TBDR

Aparte del renderizado, otro aspecto importantísimo de la arquitectura de GPUs, es la arquitectura de los shaders. Como hemos visto en la asignatura, los shaders pueden ser unificados o separados y actualmente la mayoría de las GPUs utilizan los shaders unificados que además son programables, porque son más eficientes, ofrecen más potencia y permiten la computación masiva en paralelo con las tecnologías como CUDA y OpenCL. Con las GPUs móviles pasa lo mismo – prácticamente todas las arquitecturas que se utilizan en GPUs móviles modernas tienen shaders unificados. Hay sólo dos excepciones: las GPUs Mali de la serie 400 (pero no 600) y NVidia Tegra (incluyendo NVidia Tegra 4).

Otro aspecto importante es el uso del circuito especializado para reproducción de video y, a veces, también de audio. Aunque en las GPUs para ordenadores también están presentes estos circuitos, generalmente no se usan para reproducción de videos. En la mayoría de las ocasiones el software que reproduce los videos o bien utiliza los shaders unificados de la GPU para decodificar el video (como CoreAVC, en caso de tarjetas que soportan CUDA o OpenCL), o bien lo decodifica directamente a través de la CPU (como lo hace la mayoría de reproductores y codecs gratuitos). En los dispositivos móviles pasa lo contrario – dado que el procesador, normalmente, es bastante débil y los shaders también, lo más deseado es el uso de este circuito integrado en la GPU. La desventaja que tienen estos circuitos es que el software debe estar adaptado para el uso con la GPU particular de cada dispositivo. Aparte de esto si el circuito es incapaz de decodificar un cierto formato, entonces el video solo se puede reproducir con la ayuda

de CPU o los shaders unificados, lo que en muchas ocasiones es imposible. Dado que existen muchos formatos, es imposible diseñar un circuito que los decodifica todos... Solo los formatos más populares suelen estar soportados. Finalmente, estos circuitos ocupan espacio y teóricamente se aprovecharía mejor este espacio poniendo más shaders y decodificando los videos con ellos.

No obstante, tan solo las GPUs con la arquitectura StemCell y algunos núcleos gráficos de Broadcom no implementan estos circuitos y en vez de esto apuestan por unificar completamente la parte gráfica/computacional y la parte de video.

4. GPUs móviles más importantes

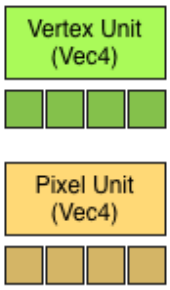
Nvidia Tegra 4

Tegra, desarrollado por Nvidia, es un system-on-a-chip para dispositivos portátiles como smartphones, tablets, personal digital assistants, y mobile Internet devices. Los chips Tegra contienen procesadores central processing unit (CPU) de Arquitectura ARM, graphics processing unit (GPU), northbridge, southbridge, y controlador de memoria en un paquete único. La serie enfatiza en el bajo consumo de energía y alto rendimiento para reproducción de audio y video.

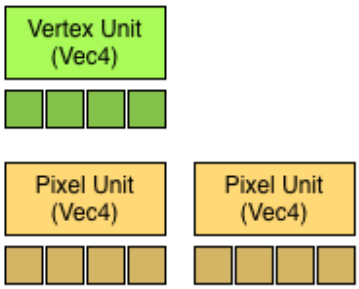
La primera versión de NVidia Tegra ha salido en febrero del 2008 e incluía soporte para OpenGL ES 2.0 y Direct3D. La Tegra 2 ha salido en 2010 con ciertos cambios en la arquitectura (pasando de GeForce ULV a GeForce ULP). La versión que ha seguido, Tegra 3 en el 2011, incluía importantes mejoras de arquitectura en su CPU, pero en GPU solo tenia cambios cuantitativos, siguiendo con la misma arquitectura. La Tegra 4 que saldrá este año, también tiene solo cambios cuantitativos.

Cabe destacar, que a diferencia de muchas otras empresas, NVidia proporciona mucha información en cuanto a la arquitectura de sus GPUs. Esto nos permite observar la evolución de Tegra desde la versión 2 hasta la versión 4.

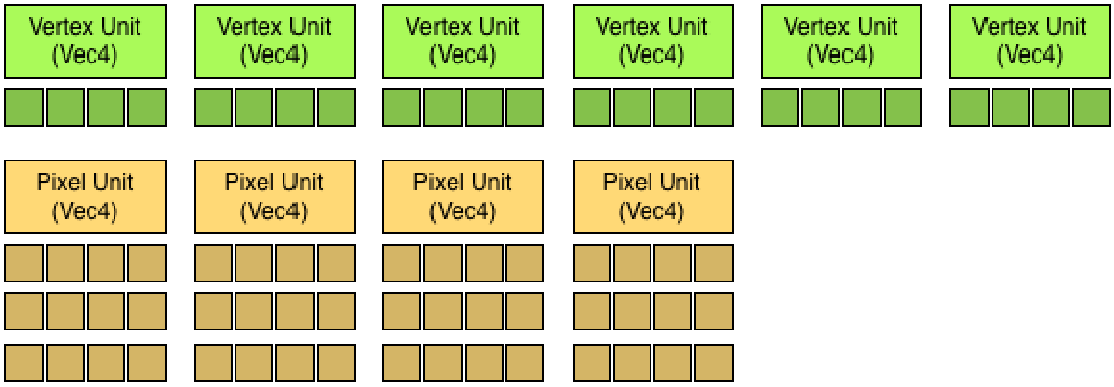
NVIDIA Tegra 2 GPU



NVIDIA Tegra 3 GPU



NVIDIA Tegra 4 GPU



Lo primero que salta a la vista observando estos esquemas es el cambio tremendo que supone el salto de Tegra 3 a Tegra 4. No debe sorprender mucho, no obstante, ya que Tegra 3 ha sido, en

muchos aspectos, la versión mejorada de Tegra 2 sin muchos cambios en la GPU desde punto de vista de la arquitectura.

El segundo aspecto importante que, quizás, es el más importante, es que NVidia Tegra, sorprendentemente, no utiliza los shaders unificados. Es decir, su arquitectura es muy distinta de la arquitectura Kepler utilizada en las GPUs de NVidia para ordenadores que ya hace tiempo tienen shaders unificados y muy avanzados (los núcleos CUDA). Es un hecho no deja de sorprender dado que todas las GPUs de sus competidores principales (Adreno, PowerVR, Mali) en sus versiones modernas tienen los shaders unificados. Una especulación por parte de los autores de este artículo consiste en que la razón de esto es que los núcleos CUDA no son suficientemente eficientes desde el punto de vista energético, por lo tanto mientras NVidia, seguramente, esta trabajando en mejorar su eficiencia, por ahora están forzados, en cierto modo, a utilizar esta arquitectura de shaders separados en shaders de los vértices y shaders de los píxeles, que es una arquitectura conceptualmente bastante antigua (basada en arquitectura NV40, introducida en 2004).

El hecho de utilizar esta arquitectura en vez de shaders unificados tiene importantes consecuencias para ciertas características de GPUs Tegra. Por ejemplo, es una de las pocas GPUs móviles que no soporta OpenCL – un hecho que da cierta pena dado el liderazgo que tiene NVidia en procesamiento en paralelo con las tarjetas para ordenadores. NVidia Tegra 4 tampoco soporta OpenGL ES 3.0, otro hecho bastante sorprendente y que seguramente tiene mucho que ver con la arquitectura utilizada en esta GPU.

No obstante, todo esto no quiere decir, que NVidia Tegra 4 es una mala GPU. Al contrario, lo más probable es que será una de las GPUs más potentes del mercado actual. El hecho de tener una arquitectura no tan moderna no implica que tenga menor potencia, especialmente en este caso, cuando el hecho de tener una buena implementación es más importante que el concepto de la arquitectura, y no cabe ninguna duda de que NVidia es una de las mejores implementaciones de la arquitectura con shaders separados. Por otro lado, NVidia dispone de muchas herramientas para desarrolladores que les permiten añadir efectos visuales muy avanzados en sus aplicaciones, muchas veces sobrepasando las posibilidades que ofrece la competencia. Por lo tanto, todo y no tener el soporte explícito de OpenGL ES 3.0, seguramente este hecho queda sobrepasado gracias a las herramientas propias de NVidia.

Por otro lado, el chip también lleva un potente circuito de facilita de reproducción y codificación de videos en alta y ultra-alta definición.

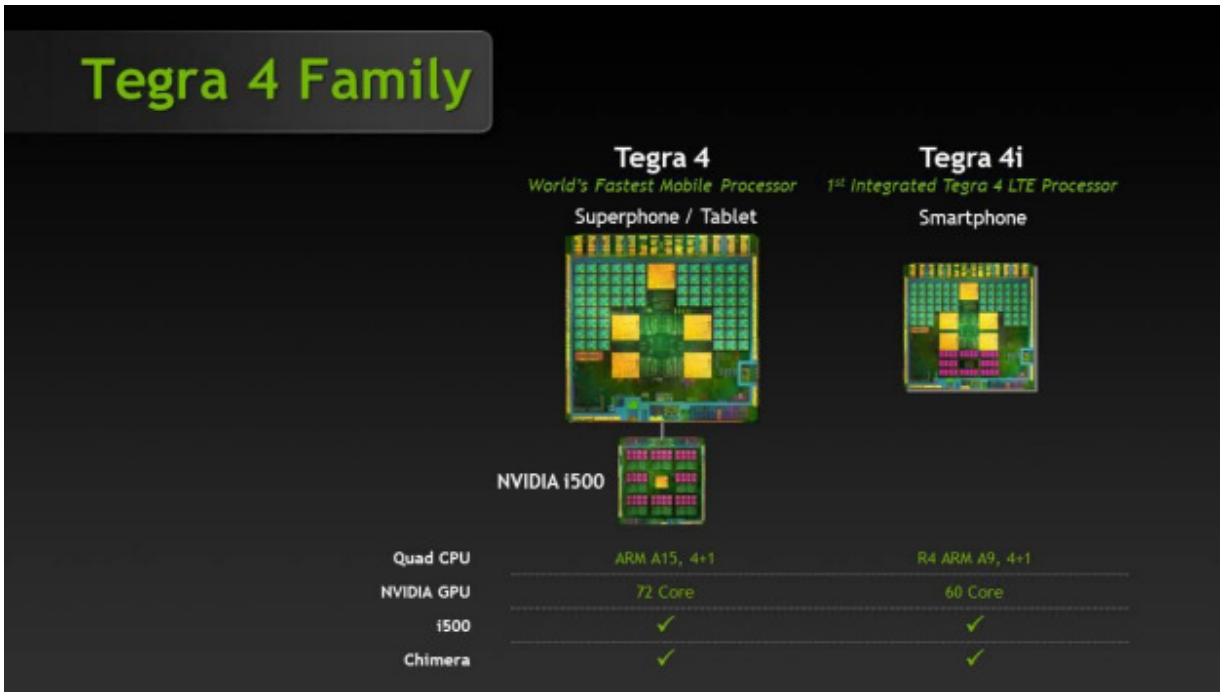
La GPU del procesador Tegra 4 acelera representaciones tanto en 2D como en 3D. Si bien la representación 2D a menudo se considera un obviedad en la actualidad, es de vital importancia para la experiencia del usuario. El motor 2D de la GPU del procesador Tegra 4 proporciona toda

la composición 2D de bajo nivel funcional, entre ellos el alfa-mezclado, dibujo lineal, escalado de vídeo, BitBLT, conversión de espacio de color, y las rotaciones de pantalla. Trabajando en conjunto con el subsistema de visualización y unidades decodificadores de vídeo, la GPU también ayuda a apoyar la salida de video 4K de alta gama.

El motor 3D es totalmente programable, e incluye la geometría y píxeles de alto rendimiento, capacidad de procesamiento que permite interfaces de usuario avanzadas 3D y juegos con calidad de consola.

La GPU también acelera el procesamiento de Flash en las páginas web y las de GPGPU de computación, tal como se utiliza en la nueva “Computational Photography Engine arquitectura NVIDIA Chimera”, que implementa en tiempo casi real, HDR foto y vídeo, fotografía, HDR de procesamiento de imagen panorámica, y "Tap-to-Track" seguimiento de la objeción.

Como se muestra en el diagrama de la familia Tegra 4, el procesador Tegra 4 incluye un subsistema con 72 GPU core. La GPU del procesador del Tegra 4 tiene 6 veces la cantidad de núcleos de procesamiento de sombreado que el procesador Tegra 3, que se traduce en más o menos el rendimiento del juego entregado 3-4x y a veces aún mayor. El procesador NVIDIA Tegra 4i utiliza la misma arquitectura de GPU como que el procesador Tegra 4, pero una variante de 60 núcleos en lugar de 72 núcleos. Incluso con los 60 núcleos que ofrece, se aprecia una asombrosa cantidad de rendimiento gráfico para los principales dispositivos smartphone.



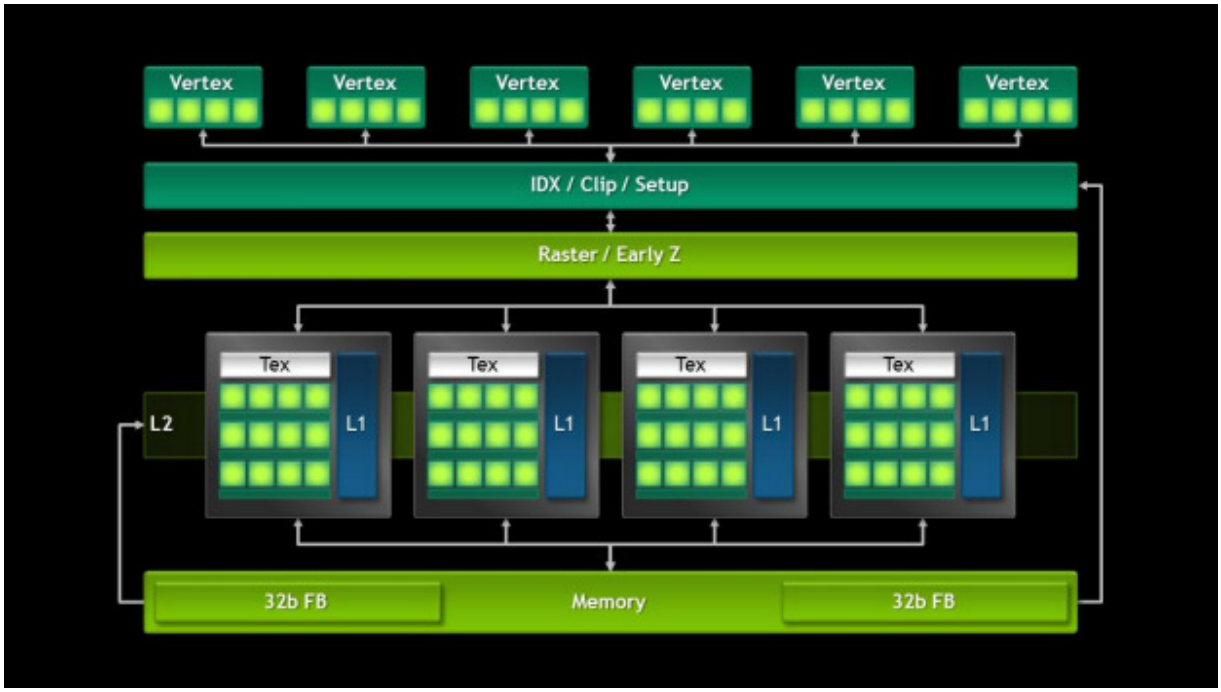
Implementación GPU Pipeline
A partir de la parte superior, los comandos de renderizado se recuperan a través de unidades finales de host / Front. A continuación, los índices y los vértices se recuperan directamente de la memoria y la caché de la unidad IDX. IDX pasa entonces vértices a los múltiples motores de procesamiento de vértices (VPEs). La unidad de IDX también soporta la creación de instancias DX9l, donde un solo comando de empate puede hacer varias instancias de un modelo, en el cual cada modelo puede utilizar un conjunto diferente de datos.

Los vértices son procesados por seis unidades de VPE en la GPU del procesador Tegra 4, cada una con un Vec4 ALU (unidad aritmética lógica) que contiene cuatro unidades MAD (Multiply-Add) (donde MAD unidades se conocen más comúnmente como Cores Vertex). El procesador Tegra 4 posee un total de 24 núcleos de vértices, que es 6 veces la cantidad de núcleos de vértices en el procesador Tegra 3.

El motor de rasterización genera fragmentos de los píxeles de las primitivas y puede proporcionar ocho fragmentos de píxeles por ciclo de reloj a las tuberías de sombreado de píxeles, similar al procesador Tegra 3. Tanto el procesador Tegra 4 como el procesador Tegra 4i ahora soportan 2x y 4x antialiasing Multisample (MSAA), 24-bit Z y el procesamiento de la plantilla de 8 bits. La Unidad de trama genera fragmentos de píxeles. La unidad “Early-Z” trabaja en conjunto con la Unidad de trama y es una versión optimizada de la aplicación utilizada en la gama alta de escritorio GeForce.

Cada uno de los cuatro shaders de fragmentos de pixeles en la GPU del procesador Tegra 4 incluye tres ALU, y cada ALU contiene cuatro unidades MAD, para un total de 48 núcleos de sombreado de píxeles (4 x 3 x 4). La MFU (Unidad multifunción) se incluye por ALU para un total de 12 unidades MFU.

Cada unidad de sombreado de píxeles también incluye una unidad de filtrado de textura capaz de realizar FP16 filtrado de texturas, que permite High Dynamic Range (HDR). Las cuatro unidades de textura tienen cada una su propia caché L1, y una caché L2 de texturas de 16K (tanto en Tegra 4 y 4i procesadores), que mejora el rendimiento mediante la reducción de textura que lee desde la memoria externa. Debido a la localidad típica de memoria de texturas, se accede por las cuatro unidades de textura.Lla combinación de L1 y L2 reduce la cantidad de texturas a las que accede fuera del chip más de un 80% en la mayoría de los casos.



Tegra 4 vs Tegra 3 GPU stats

	Tegra 4/ Tegra 3
Vertex Shader	8x
Fragment ALU	8x
Pixel Rate	2.6x
Texture Rate	2.6x
Memory Rate	2.3x
Z-Kill Rate	1.3x
Triangle Rate	1.3x

Tegra 4 - 72 Core GPU @ 672 MHz
4 pixel pipes * 3 ALUs/pipe * 4 MADS/ALU +
6 VPEs * 4 MADS/VPE

Tegra 3 - 12 Core GPU @ 520 MHz
2 pixel pipes * 1 ALU/pipe * 4 MADS/ALU +
1 VPE * 4 MADS/VPE

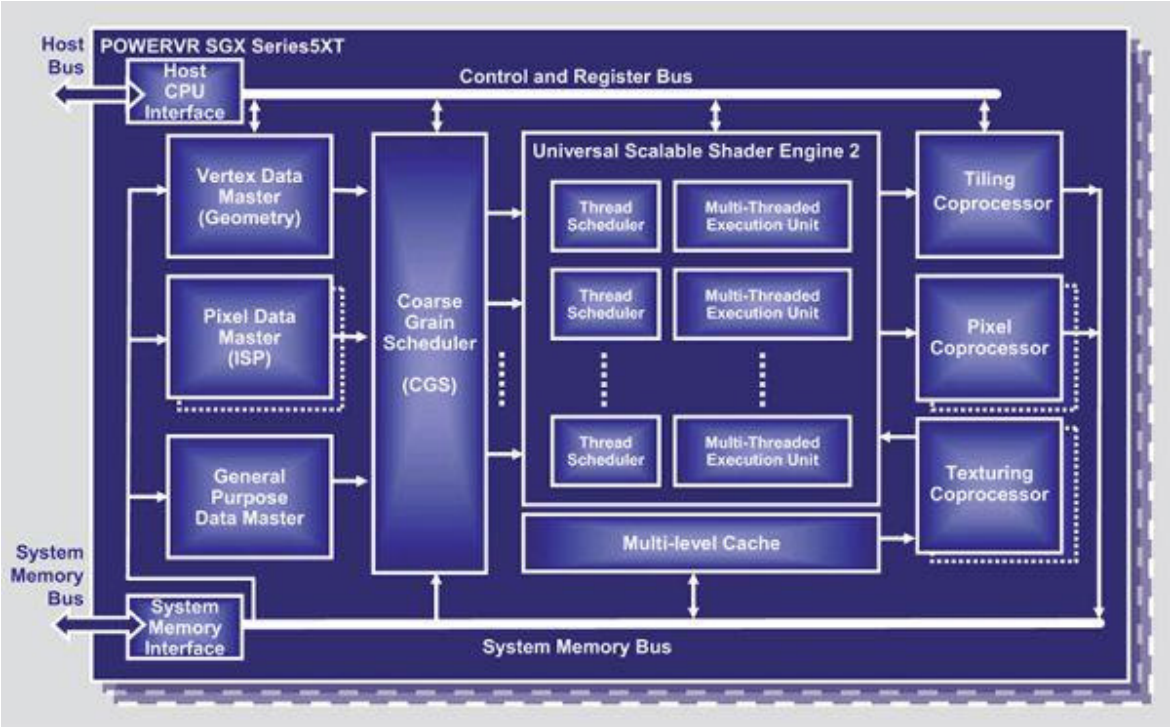
PowerVR

PowerVR es la división de la empresa británica Imagination Technologies que desarrolla una línea de GPUs móviles bajo el mismo nombre.

La línea de productos PowerVR se introdujo originalmente para competir en el mercado de PC de escritorio para aceleradores hardware 3D con un producto con una mejor relación precio / rendimiento que los productos existentes como los de 3dfx Interactive. Los cambios rápidos en ese mercado, en particular con la introducción de OpenGL y Direct3D, llevaron a una rápida consolidación, la mayoría de los jugadores más pequeños, como PowerVR, fueron empujados del mercado. PowerVR respondió introduciendo nuevas versiones con la electrónica de baja potencia que se dirige a la computadora portátil mercado. Con el tiempo esto se convirtió en una serie de diseños que podrían ser incorporados en arquitecturas adecuadas para el uso de dispositivos de mano. De esta forma, el PowerVR es un proveedor líder en el espacio móvil, que se encuentra en muchos sistemas encastados.

PowerVR no produce GPUs por sí misma, sólo las diseña y luego vende la licencia a los clientes que quieren producir un producto con su diseño. Dentro de sus clientes hay muchas empresas importantes, como Intel, Texas Instruments, Samsung y Freescale. Pero la marca de PowerVR es sobretodo conocida por un solo cliente, que es Apple, que utiliza exclusivamente las GPUs de PowerVR en sus tablets de la marca iPad y sus smartphones iPhone.

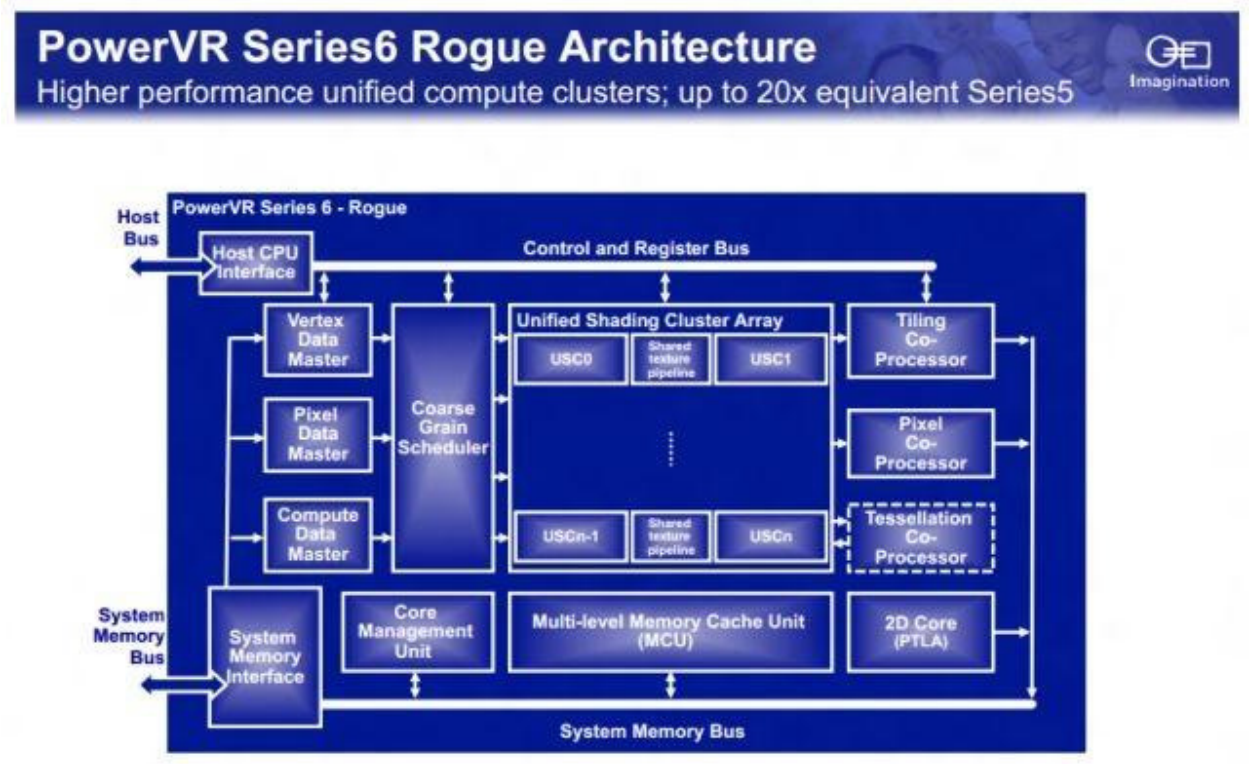
Las GPUs PowerVR tienen una larga historia. Los modelos recientes son los de la serie 5, que incorporan soporte de OpenGL ES 2.0 y, algunos, de OpenCL 1.1. Los modelos más recientes de esta serie son SGX545, SGX543, SGX544 y SGX554, todos de años 2009 y 2010. A pesar de esto hasta los tiempos más recientes eran las GPUs de referencia en cuanto al rendimiento que ofrecían.



Este hecho, posiblemente, se debe a la única arquitectura que tienen las GPUs de Imagination Technologies llamada tile-based deferred rendering o TBDR. No hay ninguna otra empresa que la utiliza en actualidad. A grandes rasgos, esta arquitectura consiste en que durante el renderizado, el cuadro de la imagen se divide en muchas partes iguales (llamadas "tiles") los cuales se renderizan pixel por pixel, intentando solo acceder a los píxeles visibles, en vez de renderizar todo el "tile" de golpe. Esto, permite, por un lado, reducir el bandwidth total necesario, que es especialmente importante en las GPUs móviles, y por otro permite realizar más eficientemente los cálculos en el z-buffer o buffer de profundidad y así ahorrar el renderizado de objetos invisibles.

Según Imagination Technologies, es esta ventaja arquitectónica que les permite siempre estar dentro de los líderes en potencia en el sector, incluso con GPUs que han salido antes de las GPUs de sus competidores.

La serie 6, la más moderna de sus GPUs, que ha salido al año pasado, continúa con esta misma arquitectura, pero también incorpora algunas mejoras.



Una de estas mejoras son el uso de así llamado "Unified Shading Cluster Array" muy similar al que usa AMD en sus tarjetas gráficas para ordenadores. La principal diferencia entre nuevos modelos de GPUs de la serie 6 será el número de "Unified Shader Clusters". Por ejemplo, el modelo G6200 los tendrá 2, mientras que G6400 los tendrá 4. Todos los modelos de la serie 6

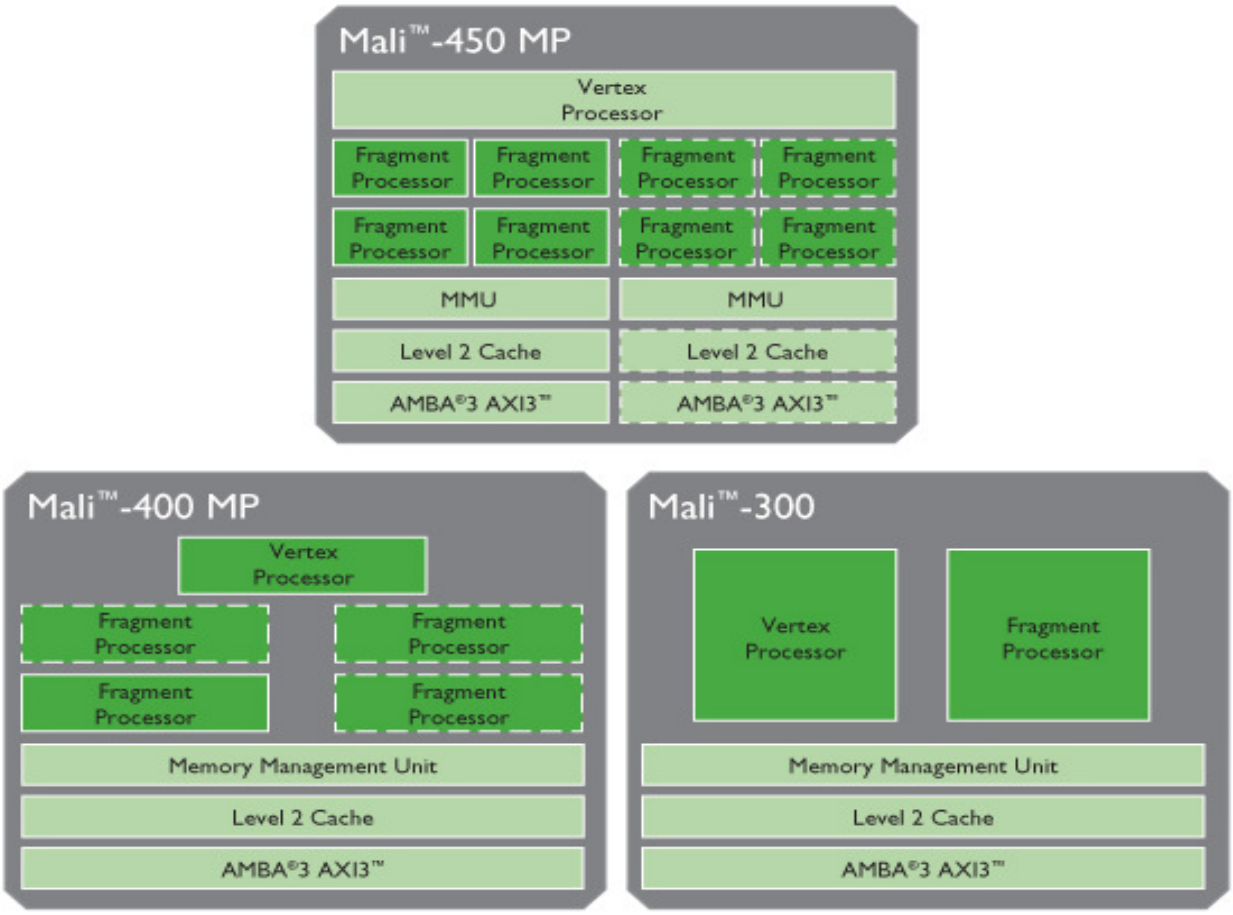
soportan OpenGL ES 3.0 y, como mínimo, Open CL 1.1. Por lo demás, mirando el esquema, parece que, en general, la nueva versión de la arquitectura es la evolución de la versión anterior.

Por ahora en el mercado no existe ningún producto con GPUs de la serie 6, pero ya hay algunas empresas muy importantes, como Texas Instruments, que han comprado la licencia para iniciar su producción. Los primeros productos con estas GPUs saldrán, probablemente, durante este año, un poco más tarde que los productos de los competidores. No obstante, dado el historial de estas GPUs que a pesar de ser un poco más antiguas podían competir con mucho éxito contra GPUs más modernas, esto no debe suponer ningún problema.

Mali

Nos referimos a Mali, como la serie de GPU's de la compañia ARM. Cabe destacar dos series diferenciadas dentro de mali: "Mali Graphics" que se especializa unicamente en gráficos, y "Mali Graphics plus GPU compute".

Mali Graphics
Los soluciones gráficas de "Mali Graphics" están basadas en la arquitectura Utgard, que permite gráficos de alto rendimiento en el área de silicio más pequeña. Los productos basados en la arquitectura escalable Utgard incluyen el Mali-300, Mali-400 MP y el Mali-450 MP. Estos productos proporcionan soluciones escalables para el mercado de masas, desde gama baja hasta gama alta. Su capacidad de gráficos proporciona un rendimiento gráfico líder en el mercado de los televisores inteligentes, tabletas y smartphones.



Las GPUs de esta serie tienen arquitectura de shaders separados, no unificados, el hecho que destaca su enfoque en aplicaciones gráficos, sin parte computacional. Como consecuencia, no soportan ni OpenGL ES 3.0, ni OpenCL. No obstante, no es una serie que se va a despedir, de hecho la última GPU de la serie – Mali-450 MP, es una GPU moderna, creada específicamente para ofrecer el mejor rendimiento por precio en aquellos productos, donde no hace falta para nada tener capacidades computacionales.

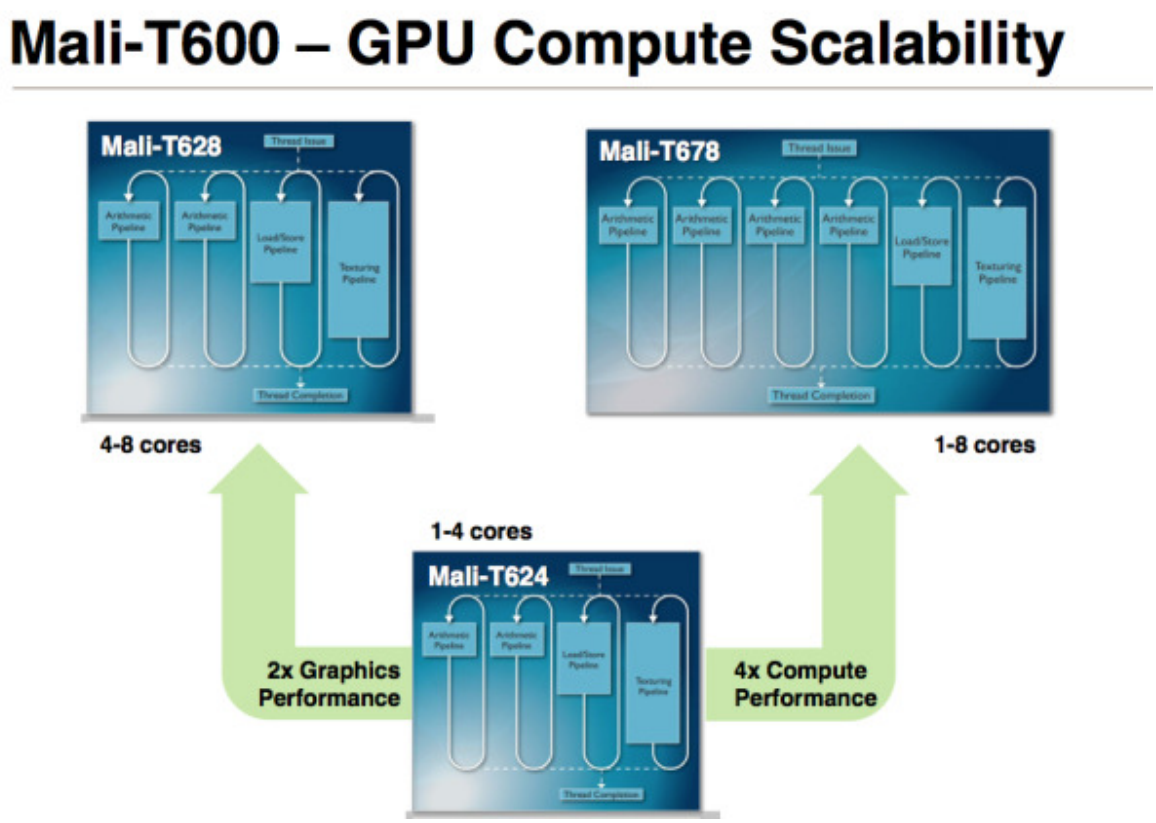
Mali-450 MP es un procesador gráfico que duplica el rendimiento de OpenGL ® ES 2.0 de la exitosa familia de productos gráficos. El Mali-450 MP GPU amplía la gama de puntos de rendimiento mediante el apoyo a la escalabilidad de hasta 8 núcleos, al tiempo que duplica el rendimiento de procesamiento de vértices.

El Mali-450 MP eleva las capacidades del anterior Mali-400 MP a nuevos niveles y, con optimización adicional arquitectónica, está diseñado para maximizar la reutilización de los recursos disponibles durante la operación de gráficos. Esto reduce al mínimo los requisitos de energía y ancho de banda del Mali-450 MP.

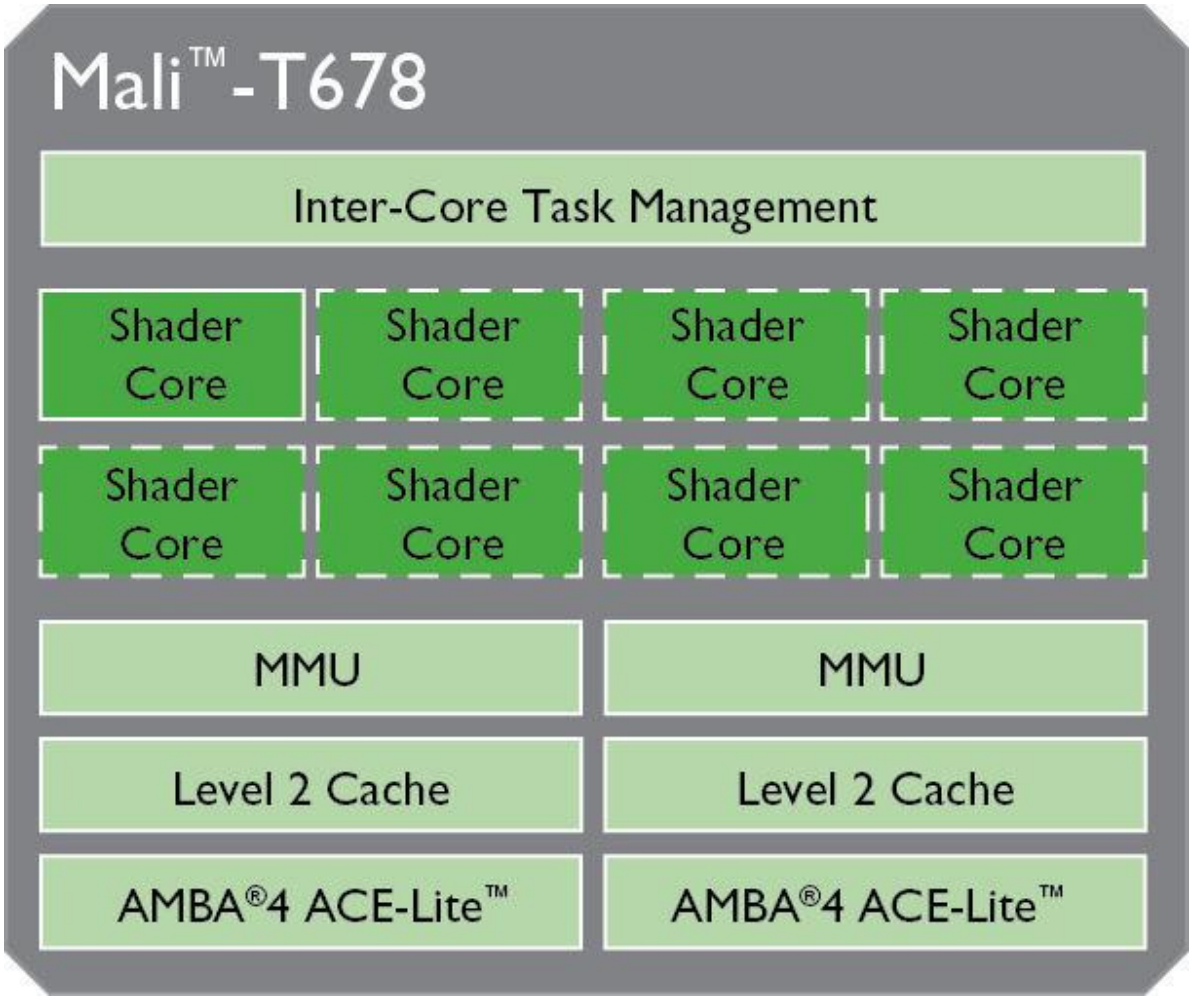
Feature	Value	Description
Anti-Aliasing	4xAA 16xAA	4x Multi-Sampling with virtually no performance drop 16xAA outperforming all implementations of comparable quality
API Support	OpenGL ES 1.1/2.0 OpenVG 1.1	Full support for next-generation and legacy 2D/3D graphics applications
Bus Interface	AMBA AXI	Compatible with a wide range of bus interconnect and peripheral IP
L2 Cache	8KB - 512KB	Configurable L2 cache optimized for graphics data traffic
Memory System	MMU	Memory Management Unit
Multi-Core Scaling	1 to 8 cores	A single IP covering a wide range of markets and price/performance points

Mali Graphics plus GPU compute

Esta serie, también llamada “serie 600” es la serie de productos más moderna de las GPUs Mali. Tiene la arquitectura “Midgard” basada en shaders unificados y trae soporte para OpenGL ES 3.0 y OpenCL 1.1. Esta pensada como una serie de GPUs de alta gama, no obstante, dentro de esta serie hay una pequeña separación en dos miniseries – una más enfocada a las aplicaciones gráficas y otra enfocada en aplicaciones de cálculo en paralelo. Este hecho refleja que es muy difícil crear aceleradores que están enfocadas al mismo tiempo tanto a los gráficos, como a la computación en paralelo.



La GPU más avanzada de la serie es Mali-T678 que esta enfocada a la computación en paralelo, no obstante sus características gráficas son virtualmente idénticas a las características de Mali-T628.



En este esquema podemos observar su composición interna. Se destaca la diferencia en arquitecturas con la línea anterior, que no tiene shaders unificados.

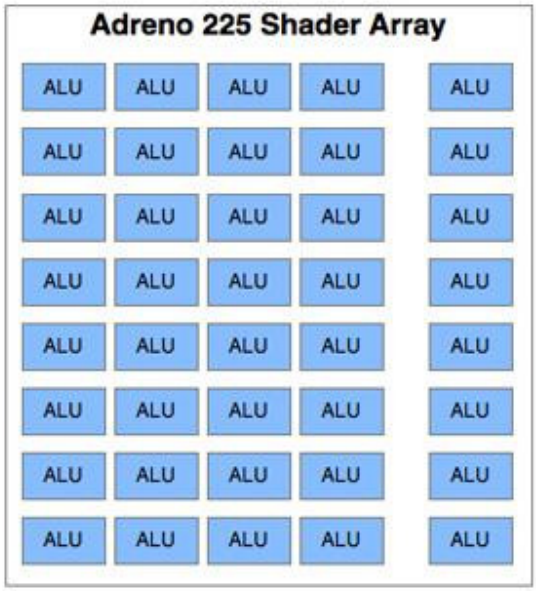
Las GPUs Mali se encuentran en muchos productos basadas en varias plataformas, la más famosa de las cuales es Samsung Exynos. Otras plataformas conocidas a los consumidores son plataformas de Allwinner y Rockchip.

Adreno

Las GPUs móviles comercializados bajo la marca “Adreno” son un producto de la empresa estadounidense Qualcomm y son parte de su plataforma “Snapdragon”. Han aparecido por primera vez en el año 2008 cuando Qualcomm compró a AMD su división de GPUs móviles junto con la línea de GPUs móviles llamada “Imageon”. Por lo tanto, los primeros Adrenos eran las mismas GPUs que “Imageon”, pero con otro nombre. De hecho, el nombre “Adreno” es un anagrama de “Radeon” – el nombre comercial de las tarjetas gráficas de AMD.

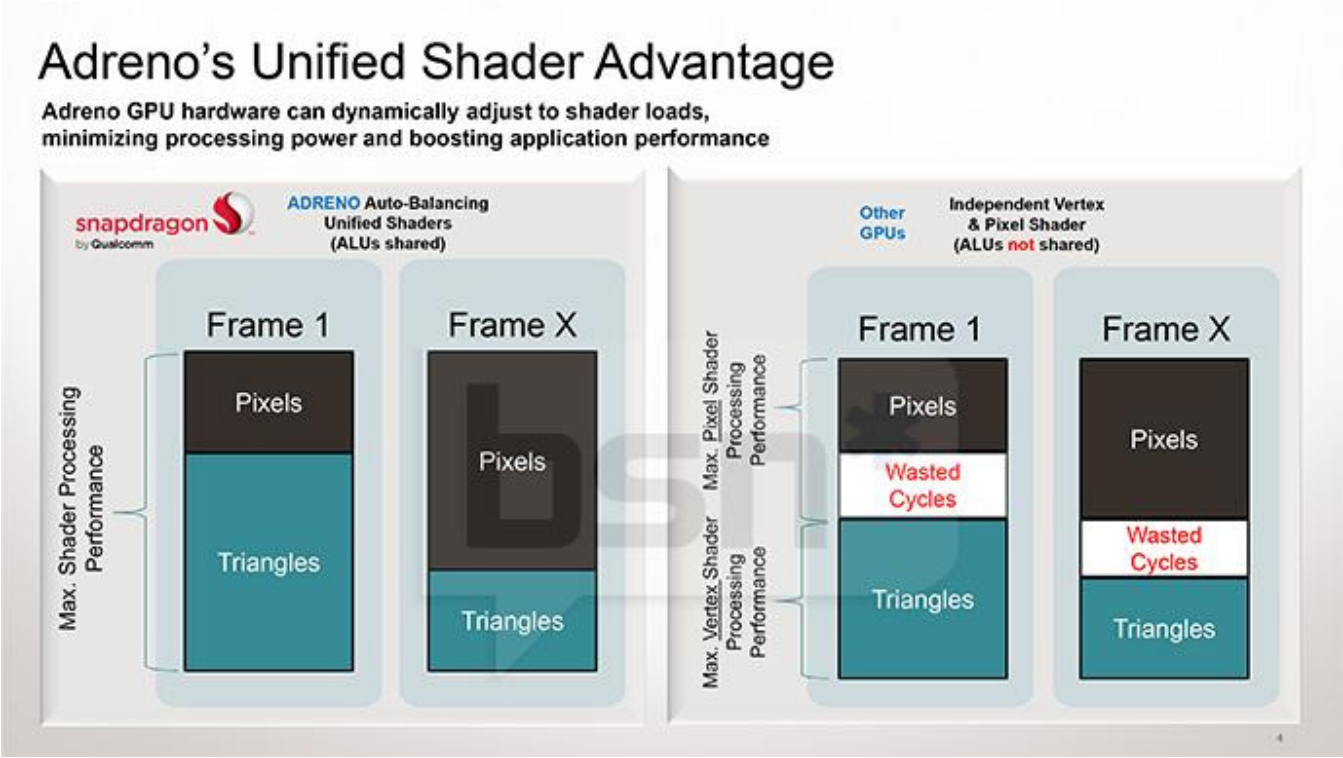
Por lo tanto, no nos debe sorprender el hecho de que los primeros Adrenos tenían la misma arquitectura que los Radeon (de hecho, Adreno 200 era lo mismo que AMD Z430). Como los Radeon, tenían una arquitectura muy avanzada que se basaba en shaders unificados y ofrecía soporte para OpenGL ES 1.1, OpenGL ES 2.0 y Direct3D, entre otras.

En las próximas GPUs de la serie, como Adreno 220, la arquitectura se ha mejorado aunque ha conservado buena parte de rasgos de los GPUs originales.

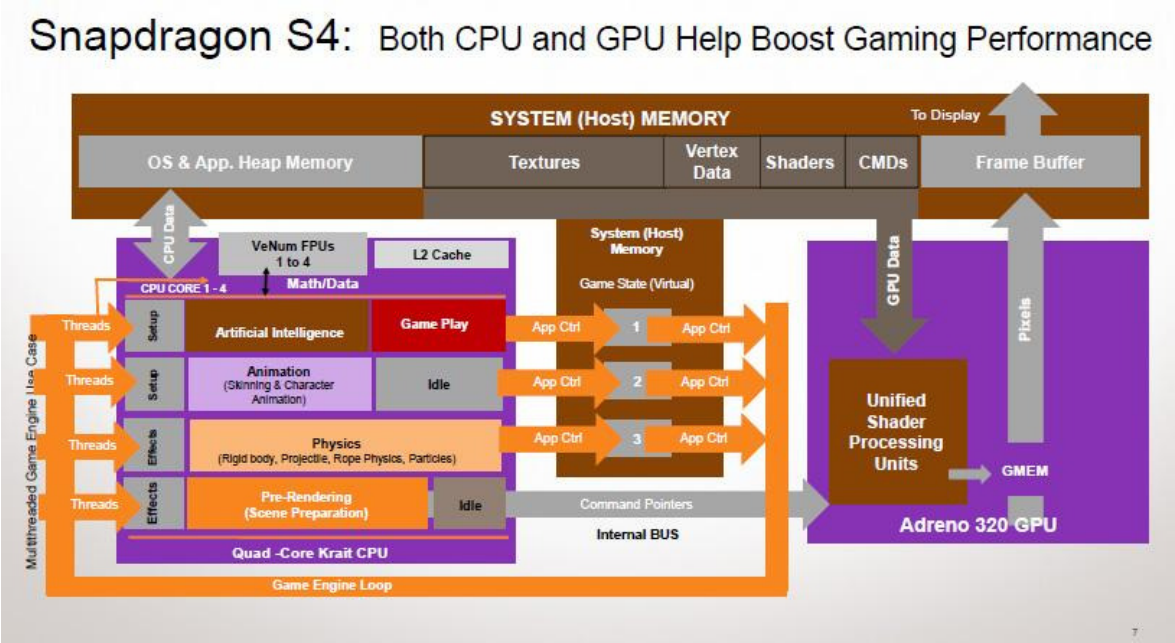


Históricamente, Qualcomm comparte muy poca información respecto a la arquitectura de sus GPUs. Una de las cosas que sabemos es que las GPUs Adreno 205 y Adreno 225 comparten la misma arquitectura y se diferencian tan sólo por algunos parámetros cuantitativos, como el número de unidades de procesamiento en paralelo y la cantidad de memoria. Adreno 225 era capaz de conseguir la potencia de 12.8 GFLOPS a frecuencia de 200 Mhz y 19.2 GFLOPS a 300 Mhz, siendo así una de la GPUs más rápidas del momento (igualada o superada por PowerVR SGX 543MP2 que ofrecía la misma potencia, pero rendía incluso mejor en aplicaciones reales).

Las próximas versiones son las GPUs de la serie 300 – Adreno 305 y Adreno 320 que según Qualcomm tienen una arquitectura completamente distinta, creada desde cero. Soporta OpenGL ES 3.0 y OpenCL 1.2 gracias a los shaders unificados lo que posibilita el uso de esta GPU como un acelerador para aplicaciones con paralelismo masivo, escritas en OpenCL 1.2.

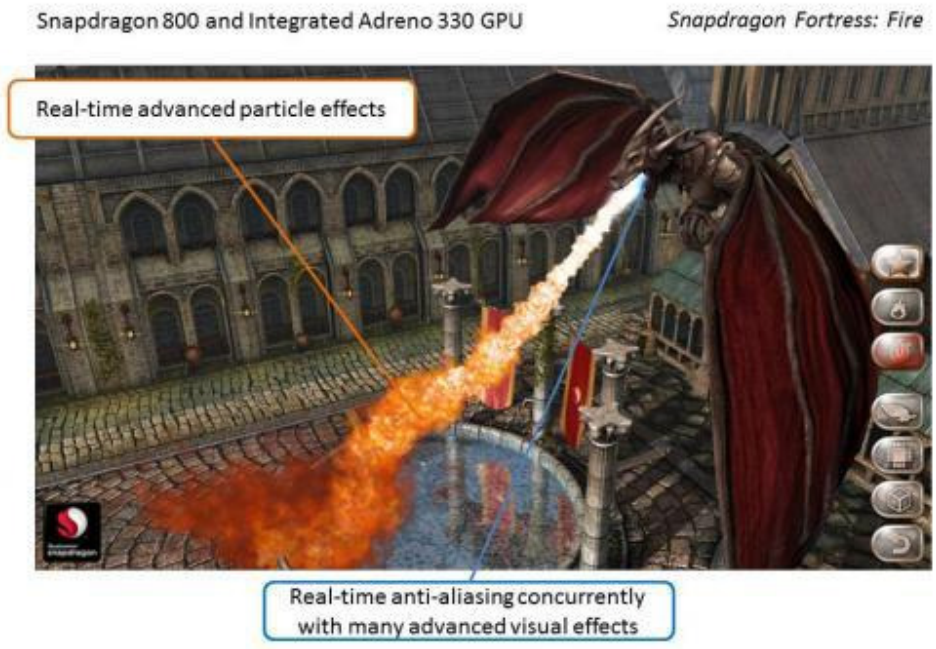


En esta imagen Qualcomm ilustra la ventaja de utilizar los shaders unificados respecto a píxel shaders y vertex shaders separados. Hay que anotar, pero, que la mayoría de sus competidores también utilizan los shaders unificados excepto NVidia.



Aquí podemos ver el esquema general de la plataforma, donde podemos destacar la fuerte integración del CPU con la GPU.

Finalmente, este año sale la nueva GPU de la serie Adreno – Adreno 330. Como es de costumbre, poco se sabe en cuanto a la arquitectura de este GPU, pero juzgando por su nombre, parece que no va tener grandes cambios en la arquitectura, sino que va a tener más cambios cuantitativos – más frecuencia, más memoria y más unidades de procesamiento. Qualcomm dice que su rendimiento será un 50% más alto que el rendimiento de Adreno 320. Mirando los videos de los demos presentados en el CES 2013, parece que el rendimiento es bastante interesante.



En esta imagen podemos apreciar la calidad de los gráficos en una demo de Qualcomm hecha para la presentación de Snapdragon 800 que incluye la GPU Adreno 330.

Entre otras cosas, hay que destacar que Adreno 330 soporta resoluciones de pantalla hasta 2560 por 2048 píxeles y es capaz de reproducir, grabar y codificar vídeos con resolución UltraHD lo que es en este caso, 3840 por 2160 píxeles (probablemente, gracias al circuito especializado integrado en GPU, no a los unidades de procesamiento en paralelo). Con todas estas características, Adreno 330 tiene pinta de ser una de las GPUs móviles más potentes que saldrán este año.

Vivante

Vivante es una empresa estadounidense fundada en 2004 por un grupo de ingenieros de NVidia con los fondos de Marvell, una empresa especializada en procesadores ARM y soluciones encastadas. No es muy conocida a nivel de consumidor, pero esta presente en muchos dispositivos, sobretodo en Asia. De las soluciones más importantes donde está presente, podemos destacar los sistemas de Freescale (plataforma i.MX6), Marvell y las plataformas de Rockchip, una empresa china. También está presente en muchos dispositivos de GoogleTV.

Muy poco se sabe de la arquitectura que tienen sus GPUs, incluso menos que de la arquitectura de Adreno. Pero sabemos que utiliza shaders unificados y que las últimas GPUs de la empresa soportan OpenGL ES 3.0, OpenGL 3.0 y OpenCL 1.2/1.1. En cuanto a su rendimiento, los tablets basados en la plataforma Freescale i.MX6 con GPU GC2000 de Vivante tienen el rendimiento comparable con NVidia Tegra 3. Soporta resoluciones de pantalla hasta 2048 por 1536 píxeles y puede reproducir videos en FullHD con frecuencia de frames de 60FPS, como, también, realizar la codificación en FullHD con frecuencia de frames de 30FPS.

Las soluciones más potentes de Vivante – GC6000 y GC4000, deben proporcionar rendimiento muy por encima de GC2000, pero no hemos encontrado comentarios o tests sobre productos que las utilizan. Mayormente los productos de Vivante se encuentran en tablets de bajo coste.

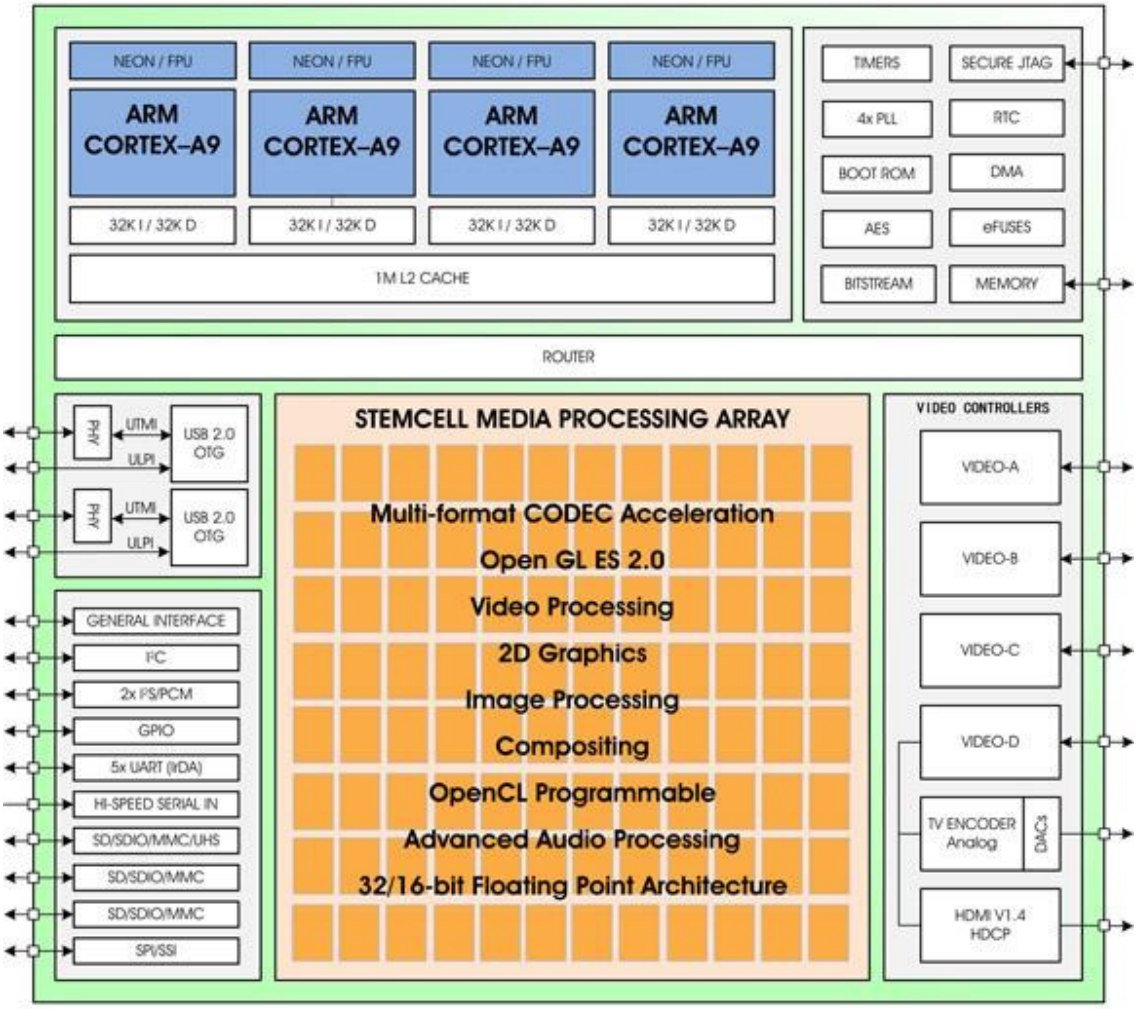
StemCell de ZiiLabs

ZiiLabs es una empresa de Singapur fundada en 1994 (entonces se llamaba “3DLabs”). Fue comprada por Creative en enero de 2009 y sus GPUs fueron utilizados, principalmente, en los reproductores y los tablets de esta empresa. En noviembre de 2012 Intel adquirió los derechos de uso de la tecnología y los patentes de ZiiLabs sin que Creative pierda el control sobre la empresa, así que es posible que la arquitectura de ZiiLabs se utilizará en los futuros productos de Intel.

Después de ser comprada por Creative, la empresa ha producido 4 plataformas que utilizaban GPUs con la arquitectura StemCell – ZMS-05, ZMS-08, ZMS-20 y ZMS-40, la más reciente.

La arquitectura StemCell es bastante peculiar, porque a diferencia de otras arquitecturas de GPUs, en tarjetas basadas en esta arquitectura el procesamiento de los videos y los gráficos es unificado gracias a los núcleos de procesamiento completamente programables, al estilo de la arquitectura CUDA. Mientras que los shaders unificados ya no son algo raro en la arquitecturas de GPUs móviles modernas, la mayoría de estas todavía tienen un circuito separado para reproducción de videos, aunque también pueden intentar a utilizar sus shaders. En cambio, en el StemCell desde principio se utilizaban estos núcleos para decodificar el stream de video en paralelo. Cabe destacar, que en esta arquitectura se intenta a procesar cuanto más con estos núcleos, más allá de solo procesamiento de gráficos y videos, por ejemplo también se utilizan para procesamiento de audio. También soportan OpenCL 1.1.

Las GPUs de primera generación – ZMS-05 y ZMS-08, tenía 24 y 64 núcleos StemCell respectivamente. En la segunda generación, ZMS-20 y ZMS-40, se ha mejorado el rendimiento del núcleo y tenía, respectivamente, 48 y 96 núcleos.



En el diagrama, podemos ver la composición interna de ZMS-40, que tanto parece a la arquitectura típica de tarjetas para ordenadores de sobremesa, como las tarjetas de NVidia y AMD.

No obstante, a pesar de ser una arquitectura prometedora, en la práctica estas GPUs no han ofrecido el rendimiento tan bueno como sus competidores. Probablemente este hecho es debido a que se tarda demasiado tiempo en “reprogramar” a los núcleos de procesamiento, con lo que el rendimiento se baja sensiblemente al ejecutar varias tareas al mismo tiempo. Por este hecho estas GPUs se utilizaban, prácticamente, solo en los productos de Creative y en pocas aplicaciones más (como en un proyector). Tampoco ha salido la nueva generación de estas GPUs para competir contra las GPUs de la nueva generación de GPUs de la competencia como NVidia Tegra 4 o Qualcomm Adreno 330.

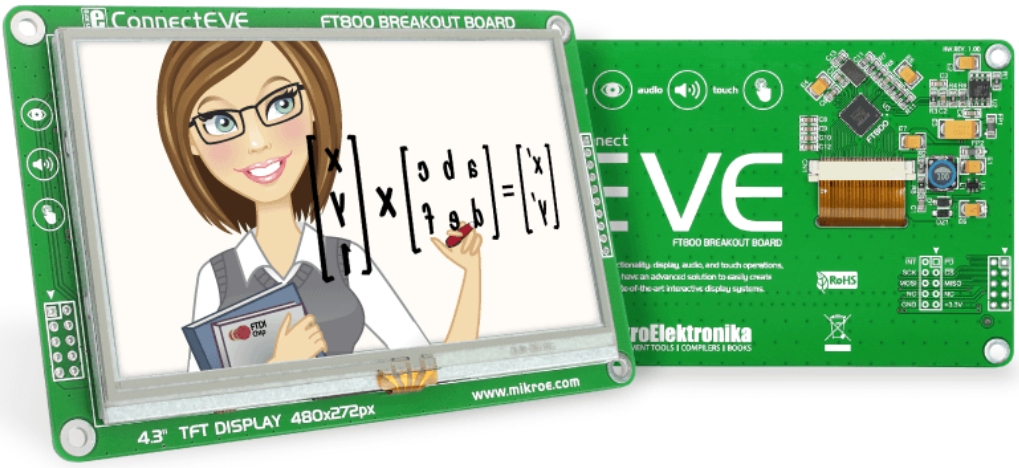
Según los tests y opiniones de consumidores, los tablets basados en estas plataformas ofrecían el rendimiento inferior a la NVidia Tegra 3 en cuanto a los gráficos y otras aplicaciones. No obstante, sí que tenían muy buen rendimiento a la hora de reproducir los videos, de hecho se destacaban muy bien en este aspecto comparado a la competencia.

SmartGPU de Vizic Technologies

Finalmente, como una curiosidad, os presentamos un acelerador grafico para microcontroladores. Si habéis programado microcontroladores en alguna asignatura como DSBM, ya sabéis que programar para obtener una interfaz gráfica o mostrar un texto en la pantalla es bastante complicado. Es la situación similar a la que tenían los primeros ordenadores de sobremesa. Y aquí es, como ha pasado en la historia con los ordenadores, cuando entran los aceleradores para microcontroladores.

SmartGPU es un pequeño controlador de pantalla que se conecta a un microcontrolador como Arduino, PIC, micros de Freescale, etc. Permite dibujar en la pantalla con llamadas muy simples como “drawLine()” o “drawRectangle()”. Mientras en la mayoría de otros controladores hay que dibujar todo punto a punto creando complicadas subrutinas para muchos casos, con este pequeño controlador se puede simplificar mucho el desarrollo de una interfaz gráfica. También libera el espacio de programa en el microcontrolador principal, porque no hace falta guardar todas estas subrutinas. También simplifica muchísimo la escritura de textos, ya que se puede pasar al acelerador directamente el texto y el tipo y tamaño de letra para mostrarlo en la pantalla.

Este acelerador viene con una pantalla con resolución 320 por 240 píxeles capaz de mostrar 262,144 colores y tiene 29 comandos para dibujar simples objetos geométricos e imágenes guardadas en la memoria. Si programáis para microcontroladores y necesitáis una interfaz gráfica podéis considerar su uso, porque puede simplificar mucho el desarrollo. Existen también otros aceleradores gráficos para microcontroladores, como, por ejemplo, FTDI Chip's EVE (Embedded Video Engine) que soporta pantallas con resolución hasta 400 por 240 píxeles y incluso tiene capacidad para mostrar objetos 3D (muy simples) y que se puede conectar hasta con un microcontrolador de 8 bits, mejorando muchísimo sus capacidades gráficas.



ConnectEVE

Nosotros hemos decidido hablar de esto, porque, al fin y al cabo, estos simples controladores también son aceleradores. Los aceleradores pueden ser tan grandes, complicados y potentes como NVidia Titan o tan pequeños y simples como SmartGPU, pero en ambos casos su tarea es la misma – liberar al procesador principal de una carga considerable y así simplificar la vida tanto al desarrollador como al usuario. Esta tarea se ha convertido en tan amplia e importante que ahora los aceleradores están presentes en todos los ámbitos de la industria informática empezando por microcontroladores y acabando en superordenadores.