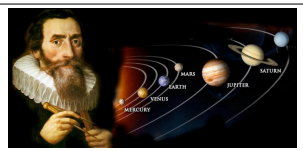
Kepler

Manuel Ujaldón
Nvidia CUDA Fellow
Dpto. Arquitectura de Computadores
Universidad de Málaga

Índice de contenidos [25 diapositivas]

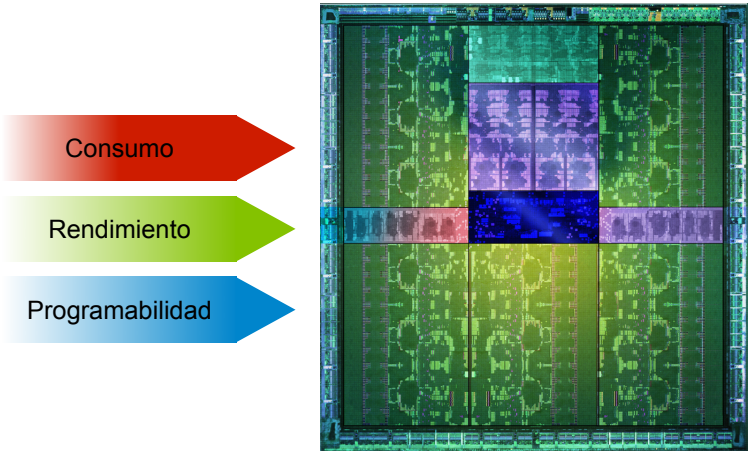
1. Presentación de la arquitectura

Kepler, Johannes (1571-1630)



- Autor de las leyes del movimiento planetario.
  - **Primera ley:** Las órbitas de los planetas son planas. El sol está en el plano de la órbita. La trayectoria del planeta respecto del sol es una elipse en la que el sol ocupa uno de los fotos.
  - **Segunda ley:** El radio vector que une al sol y el planeta barre áreas iguales en tiempos iguales. Un planeta se mueve más rápidamente en su perihelio que en su afelio, y mientras más excéntrica sea su órbita, mayor será la diferencia de velocidad entre sus extremos.
  - **Tercera ley:** Los cuadrados de los períodos de revolución en torno al sol son proporcionales a los cubos de los semiejes mayores de las órbitas. La velocidad media con que un planeta recorre su órbita disminuye a medida que el planeta está más lejos del sol. La influencia que el sol ejerce sobre los planetas disminuye con la distancia.

Nuestra Kepler también tiene 3 leyes

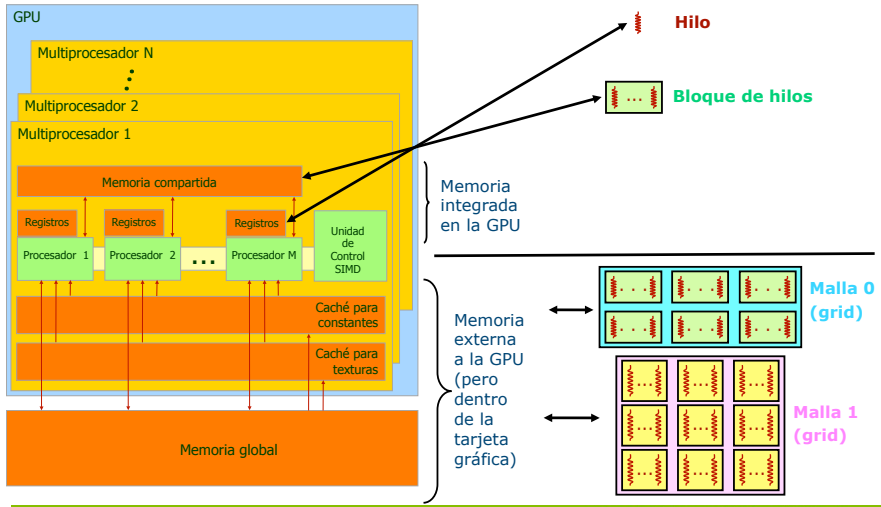


Resumen de sus rasgos más sobresalientes

- **Fabricación:** 7100 Mt. integrados a 28 nm. por TSMC.
- **Arquitectura:** Entre 7 y 15 multiprocesadores SMX, dotados de 192 cores cada uno.
  - El número de multiprocesadores depende de la versión [GKxxx].
- **Aritmética:** Más de 1 TeraFLOP en punto flotante de doble precisión (formato IEEE-754 de 64 bits).
  - Los valores concretos dependen de la frecuencia de reloj de cada modelo (normalmente, más en las GeForce y menos en las Tesla).
  - Con sólo 10 racks de servidores, podemos alcanzar 1 PetaFLOP.
- **Diseño:**
  - Paralelismo dinámico.
  - Planificación de hilos.

2. Los cores y su organización

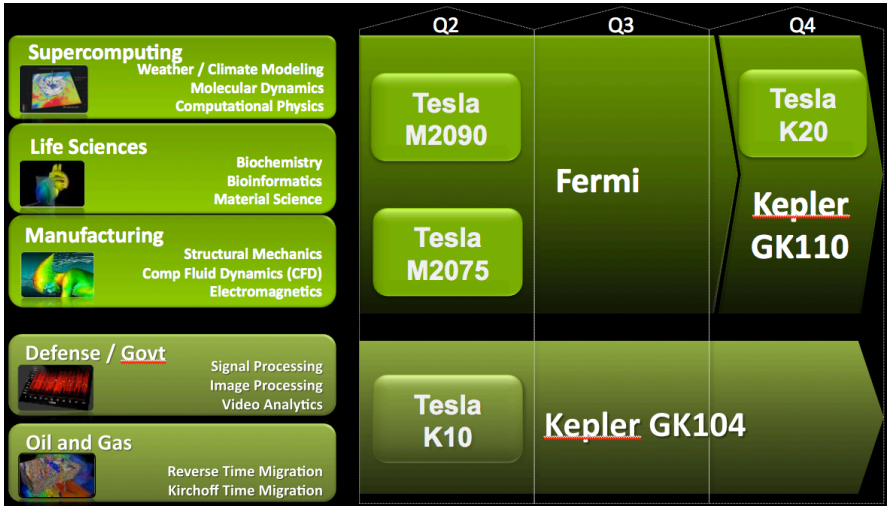
Un breve recordatorio de CUDA



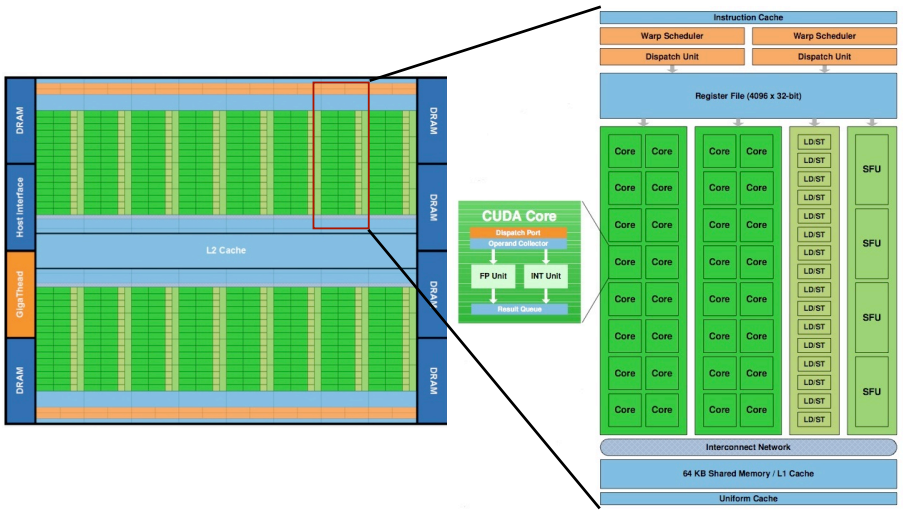
... y de cómo va escalando la arquitectura

Arquitectura	G80	GT200	Fermi GF100	Fermi GF104	Kepler GK104	Kepler GK110
Marco temporal	2006-07	2008-09	2010	2011	2012	2013
CUDA Compute Capability (CCC)	1.0	1.2	2.0	2.1	3.0	3.5
N (multiprocs.)	16	30	16	7	8	15
M (cores/multip.)	8	8	32	48	192	192
Número de cores	128	240	512	336	1536	2880

Ubicación de cada modelo en el mercado: Marco temporal, gama y aplicaciones

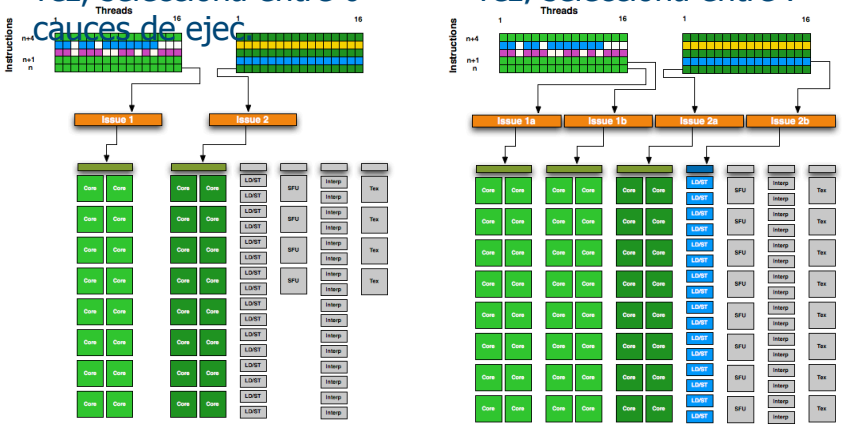


Su precursora Fermi



GF100 vs. GF104

- GF100: Emite 2 instrs. a la vez, selecciona entre 6 cauces de ejec.
- GF104: Emite 4 instrs. a la vez, selecciona entre 7



Kepler GK110: Disposición física de las UFs

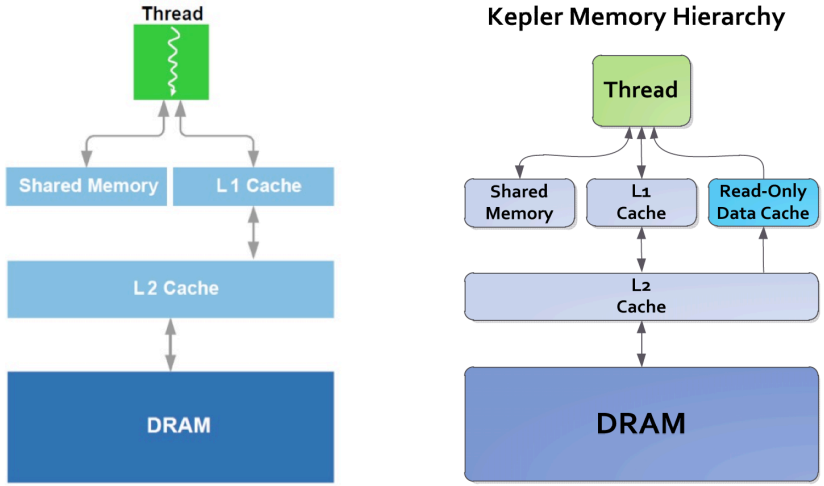
Del multiprocesador SM de Fermi GF100 al multiprocesador SMX de Kepler GK110

3. La memoria y el transporte de datos

Mejoras en la memoria y el transporte de datos

- **Memoria integrada** en cada SMX. Respecto a los multiprocesadores SM de Fermi, kepler duplica:
  - Tamaño y ancho de banda del banco registros.
  - Ancho de banda de la memoria compartida.
  - Tamaño y ancho de banda de la memoria caché L1.
- **Memoria interna (caché L2):** 1.5 Mbytes.
- **Memoria externa (DRAM):** GDDR5 y anchura de 384 bits (frecuencia y tamaño dependerán de la tarjeta gráfica).
- **Interfaz con el host:**
  - Versión 3.0 de PCI-express (el a. banda dependerá de la placa base).
  - Diálogos más directos entre la memoria de vídeo de varias GPUs.

Diferencias en la jerarquía de memoria: Fermi vs. Kepler



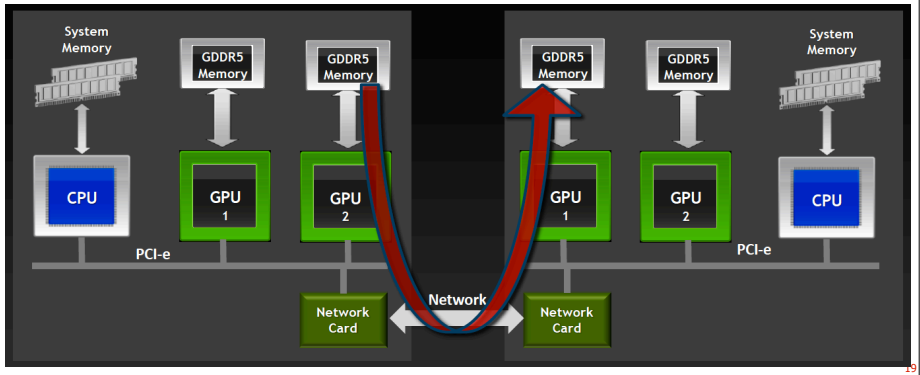
La jerarquía de memoria en cifras

Generación de GPU	Fermi		Kepler		Limi-tación	Impacto
Modelo hardware	GF100	GF104	GK104	GK110		
CUDA Compute Capability (CCC)	2.0	2.1	3.0	3.5		
Tope de registros de 32 bits / hilo	63	63	63	255	SW.	Working set
Regs. de 32 bits / Multiprocesador	32 K	32 K	64 K	64 K	HW.	Working set
Mem. compartida / Multiprocesador	16-48KB	16-48KB	16-32-48KB	16-32-48 KB	HW.	Tile size
Caché L1 / Multiprocesador	48-16KB	48-16KB	48-32-16KB	48-32-16 KB	HW.	Velocidad de acceso
Caché L2 / GPU	768 KB.	768 KB.	1536 KB.	1536 KB.	HW.	Velocidad de acceso

- Todos los modelos de Fermi y Kepler incorporan:
  - Corrección de errores ECC en DRAM.
  - Anchura de 64 bits en el bus de direcciones.
  - Anchura de 64 bits en el bus de datos por cada controlador (todos presentan 6 controladores para 384 bits, salvo GF104 que tiene 4).

GPUDirect ahora soporta RDMA [Remote Direct Memory Access]

- Esto permite transferencias directas entre GPUs y dispositivos de red, y degrada menos el extraordinario ancho de banda de la memoria de vídeo GDDR5.



4. Programabilidad: Nuevas prestaciones

Limitadores del paralelismo a gran escala

Generación de GPU	Fermi		Kepler	
Modelo hardware	GF100	GF104	GK104	GK110
CUDA Compute Capability (CCC)	2.0	2.1	3.0	3.5
Número de hilos / warp (tamaño del warp)	32	32	32	32
Máximo número de warps / Multiprocesador	48	48	64	64
Máximo número de bloques / Multiprocesador	8	8	16	16
Máximo número de hilos / Bloque	1024	1024	1024	1024
Máximo número de hilos / Multiprocesador	1536	1536	2048	2048

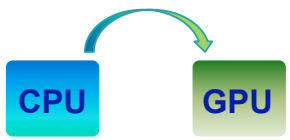
Kepler vs. Fermi: Computación a gran escala, paralelismo dinámico y planificación de hilos

Generación de GPU	Fermi		Kepler		Limitación	Impacto
Modelo hardware	GF100	GF104	GK104	GK110		
Compute Capability (CCC)	2.0	2.1	3.0	3.5		
Máxima dimensión X de la malla	2^16-1	2^16-1	2^32-1	2^32-1	Software	Tamaño del problema
Paralelismo dinámico	No	No	No	Sí	Hardware	Estructura del problema
Planificación de mallas (Hyper-Q)	No	No	No	Sí	Hardware	Planificación de hilos

Mejorando la programabilidad

¿Qué es el paralelismo dinámico?

- La habilidad para lanzar nuevos procesos (mallas de bloques de hilos) desde la GPU de forma:
  - Dinámica.
  - Simultánea.
  - Independiente.

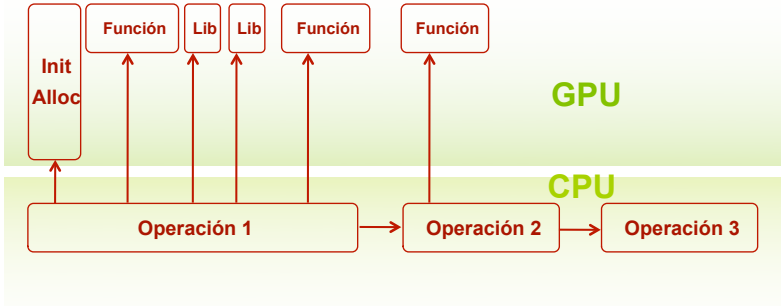


Fermi: Sólo la CPU puede generar trabajo en GPU.

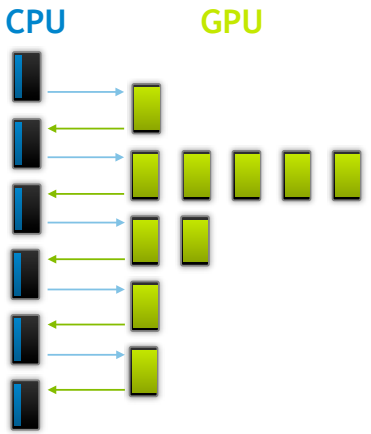
Kepler: La GPU puede generar trabajo por sí sola.

Así se hacían las cosas hasta ahora: La GPU era un mero esclavo del host o CPU

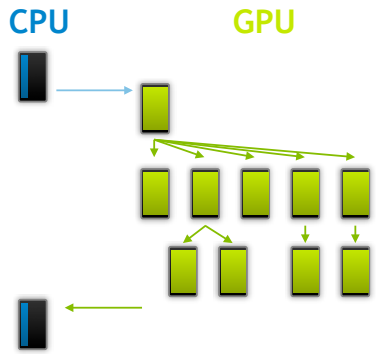
- Gran ancho de banda en las comunicaciones:
  - Externas: Superior a 10 GB/s (PCI-express 3).
  - Internas: Superior a 100 GB/s (memoria de vídeo GDDR5 y anchura de bus en torno a 384 bits, que es como un séxtuple canal en CPU).



Y así se pueden hacer a partir de ahora: Las GPUs Kepler lanzan sus propios kernels

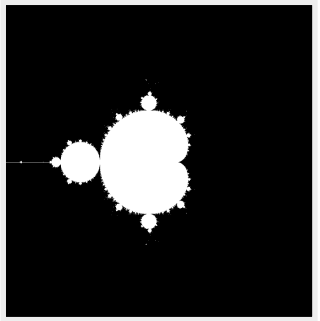


La GPU como co-procesador



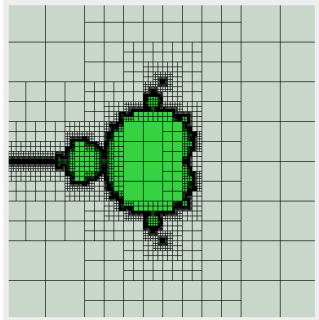
GPU autónoma: Paralelismo dinámico

Paralelismo dependiente del volumen de datos o de la "calidad computacional" de cada región



CUDA hasta 2012

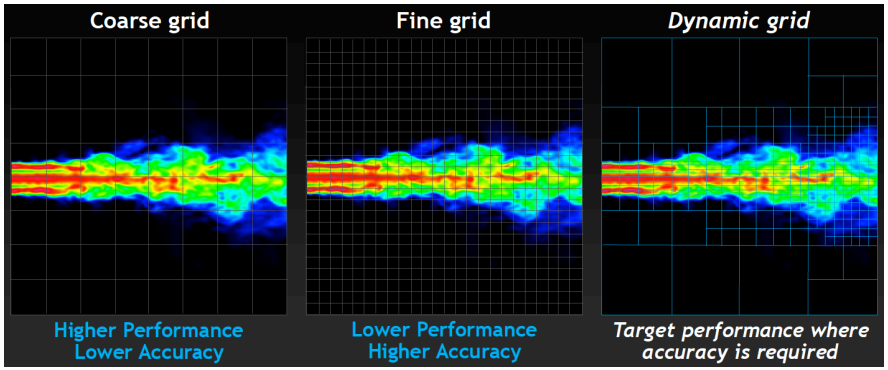
La potencia computacional se asocia a las regiones según su interés



CUDA sobre Kepler

Generación dinámica de la carga

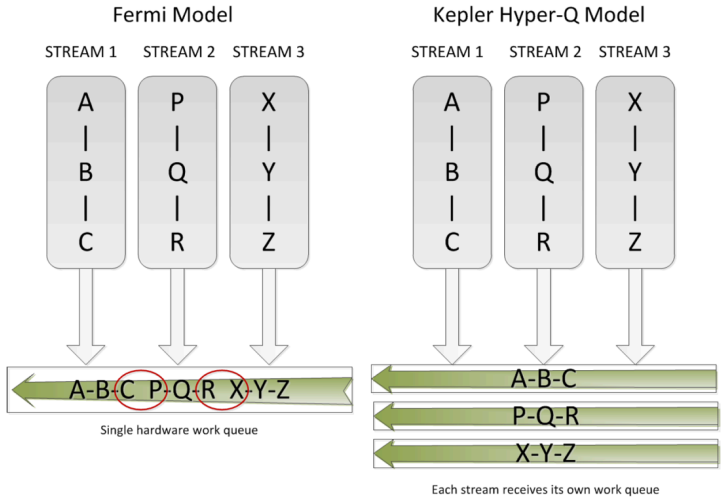
- Facilita la computación en GPU.
- Amplía el ámbito de las aplicaciones en que puede ser útil.



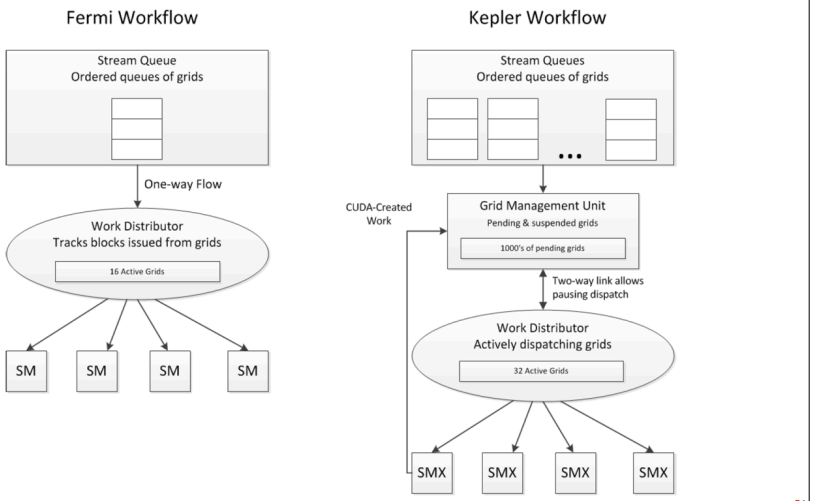
Hyper-Q

- En Fermi, diversos procesos de CPU ya podían enviar sus mallas de bloques de hilos sobre una misma GPU, pero un kernel no podía comenzar hasta que no acabase el anterior.
- En Kepler, pueden ejecutarse hasta 32 kernels procedentes de varios procesos de CPU de forma simultánea, lo que incrementa el porcentaje de ocupación temporal de la GPU.
- Veámoslo con un sencillo ejemplo...

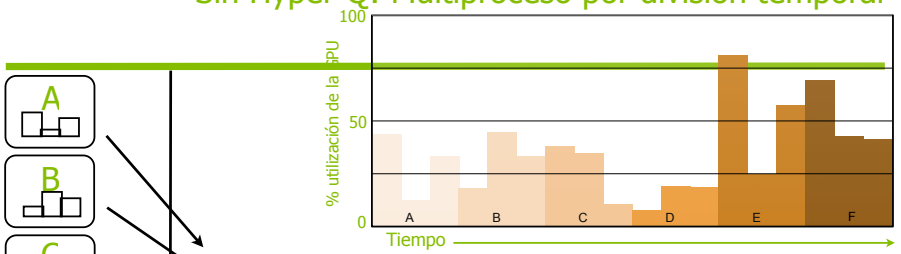
Planificación de kernels con Hyper-Q



Con Hyper-Q, una malla no ocupa toda la GPU, sino los multiprocesadores necesarios



Sin Hyper-Q: Multiproceso por división temporal



Con Hyper-Q: Multiproceso simultáneo