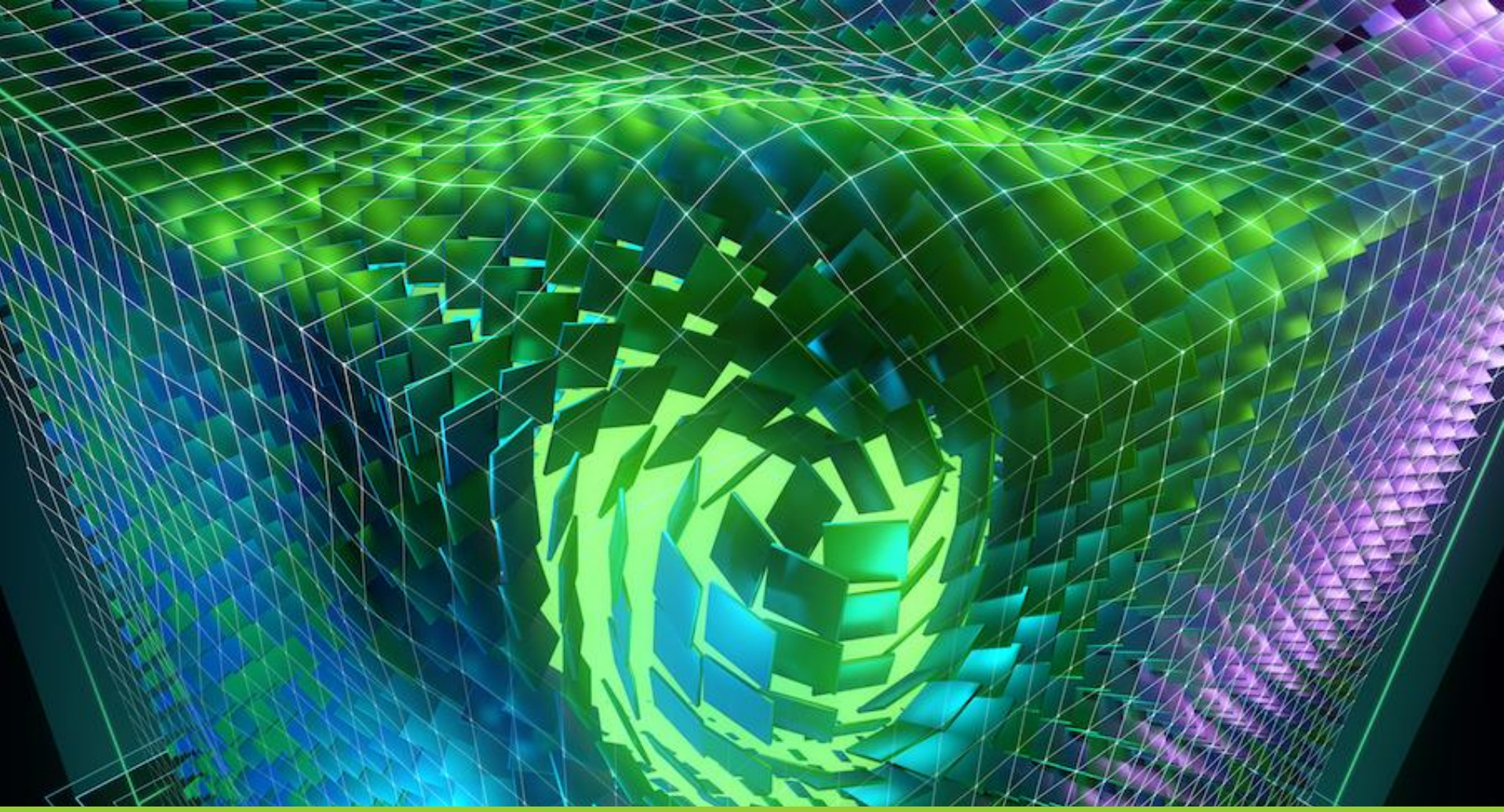


INF351 – Computación de Alto Desempeño

Jerarquía de Memorias

PROF. ÁLVARO SALINAS

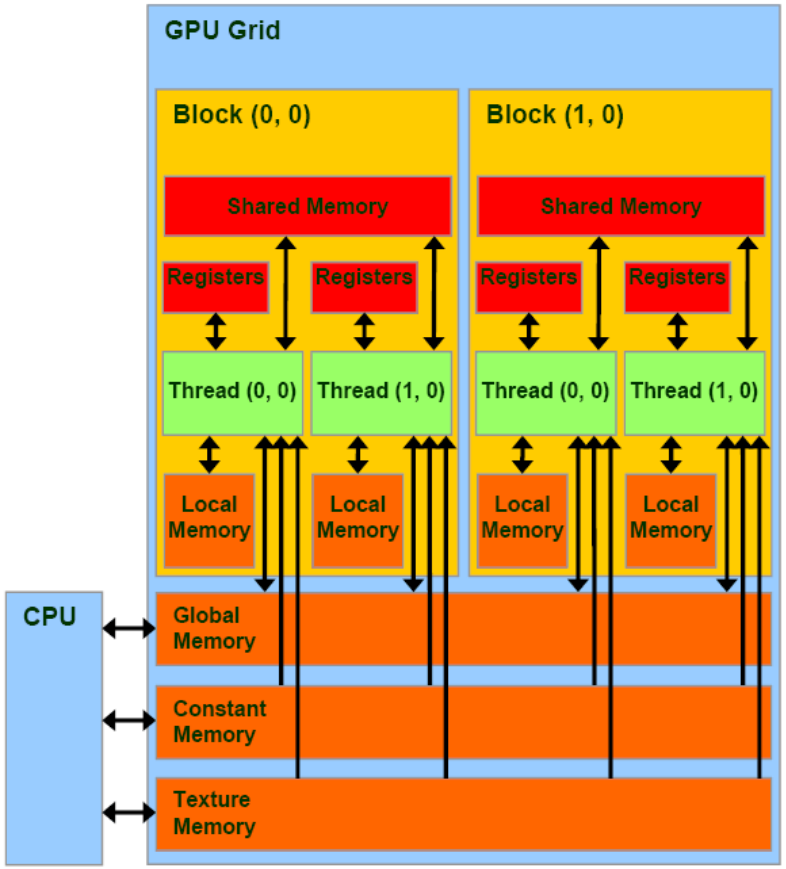
Tipos de Memoria

CUDA pone a nuestra disposición 6 diferentes tipos de memoria.

Cada uno de estos posee distintas características y optimizaciones.

Un correcto manejo de estas memorias puede significar un aumento importante en el rendimiento de nuestra aplicación.

A continuación daremos un breve vistazo a las distintas memorias desde la más rápida a la más lenta.

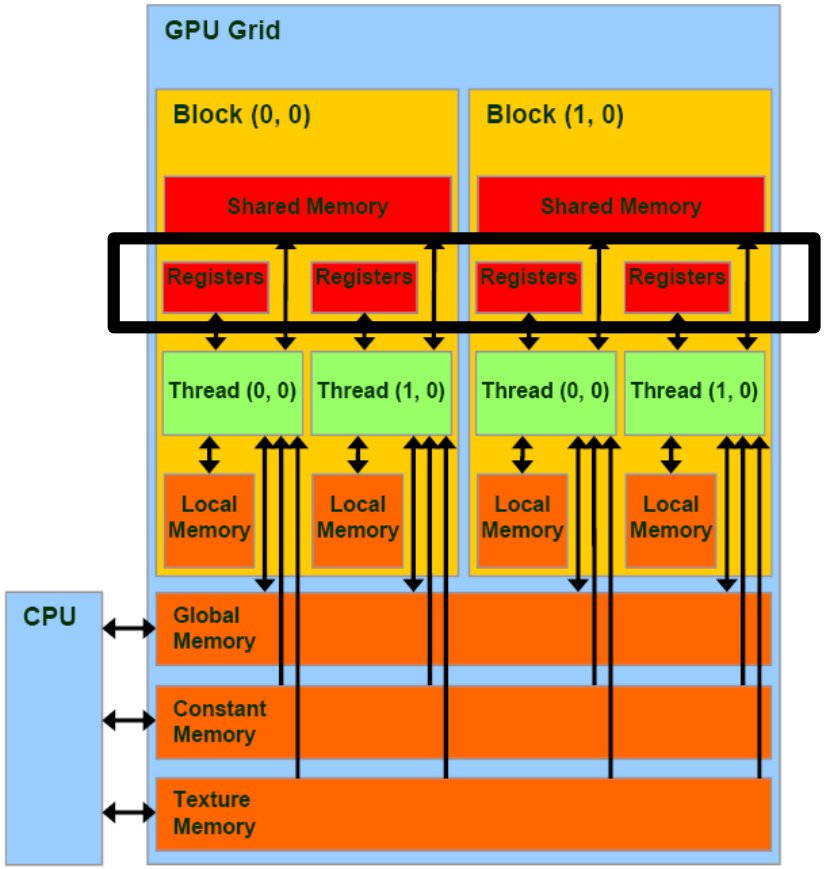


Registros

Los registros correponden a la memoria más rápida que podemos utilizar.

Un registro solo es visible para la hebra que lo escribió y dura hasta la muerte de dicha hebra.

Lamentablemente, no podemos controlar que variables se alojarán en esta memoria, pues es algo determinado por el compilador.

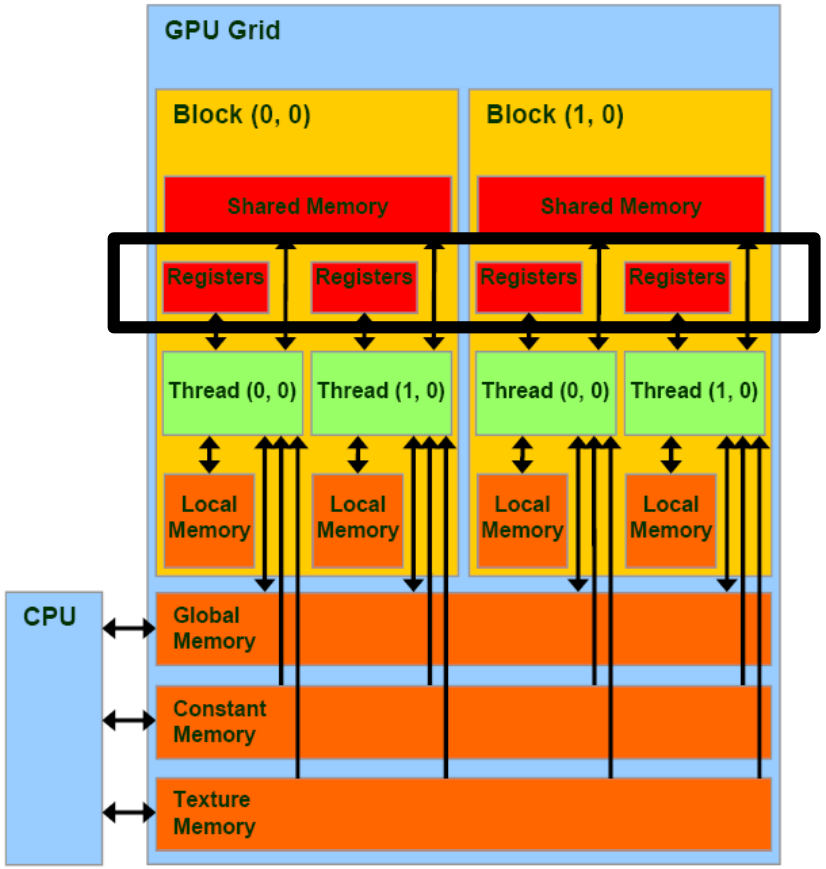


Registros

El tamaño de la memoria de registros es bastante limitado y un sobreuso de ésta puede generar graves problemas de desempeño.

Cuando se utilizan todos los registros, la memoria se “derrama” (register spilling) y las variables comienzan a almacenarse en memoria local.

Los registros máximos por SM y por hebra limitan la cantidad de bloques asignados a un SM de acuerdo a la cantidad de hebras en ese bloque.

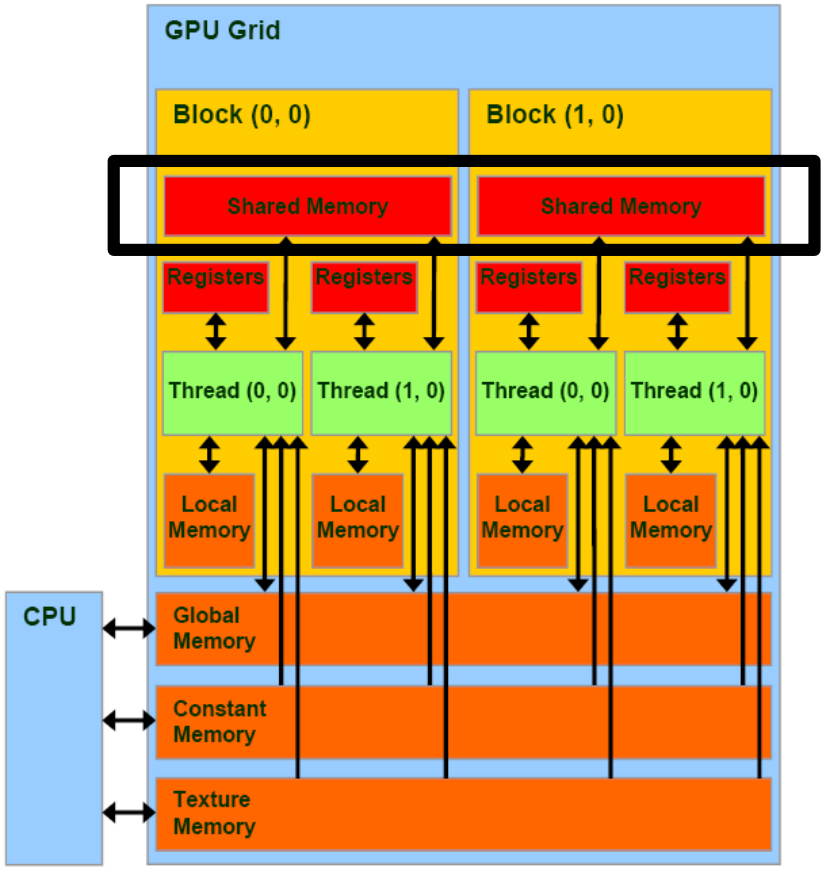


Memoria Compartida

Shared memory es la segunda memoria más rápida del dispositivo.

Es visible para un bloque completo y permite la comunicación entre las hebras que lo conforman (de ahí el nombre).

Su uso es conveniente cuando varias hebras pretenden leer la misma dirección de memoria.

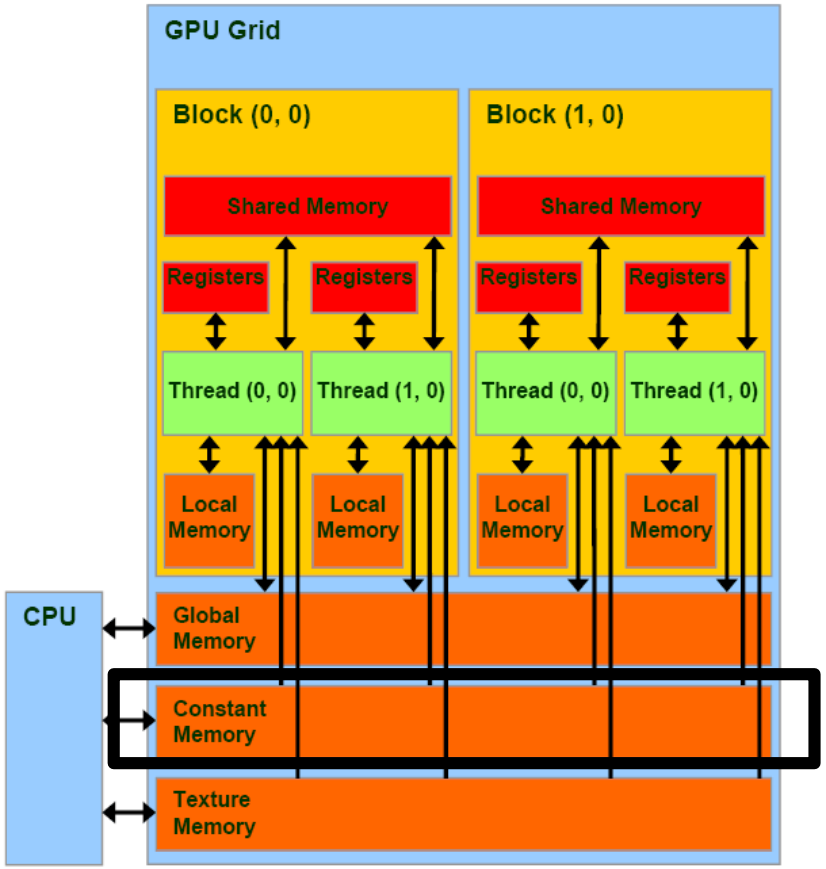


Memoria Constante

La memoria constante es visible para todas las hebras que estén en ejecución, pero solo pueden leer desde ella.

También es visible desde la CPU.

Usar memoria constante solamente es beneficioso cuando las hebras de un warp intentan leer la misma dirección de memoria. Cuando acceden a posiciones diferentes, es menos eficiente que con memoria global.

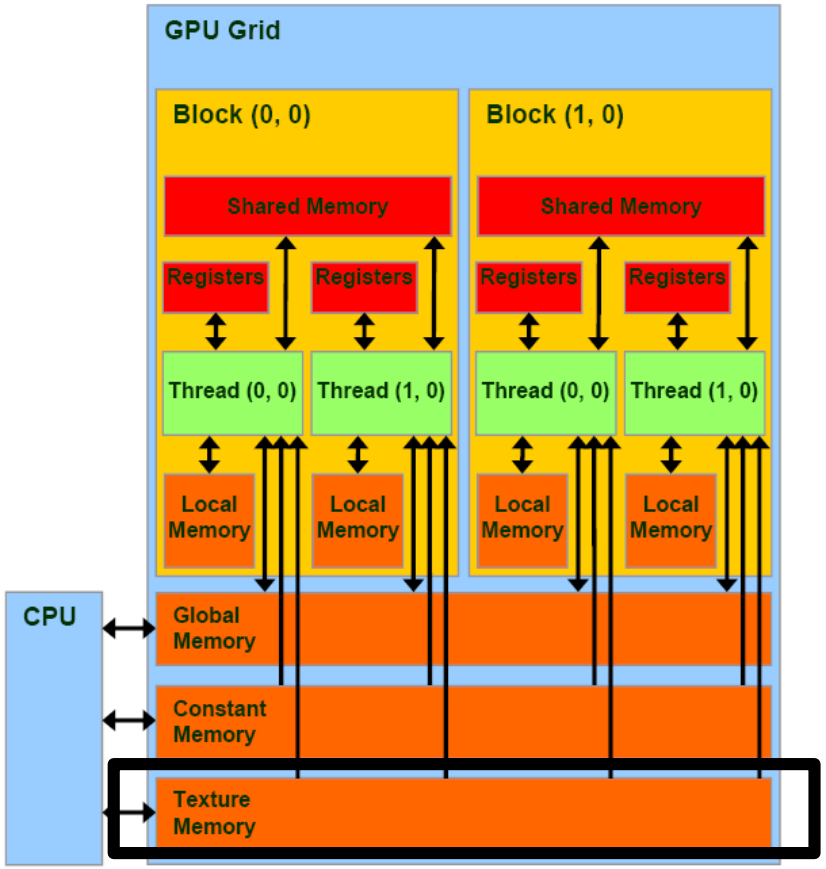


Texturas

La memoria de textura comparte varios rasgos con la memoria constante. Es visible para la CPU y todas las hebras, pero ellas solo pueden leer desde esta memoria.

La diferencia radica en que está optimizada para accesos físicamente adjacentes en un espacio 2D.

Debido a esto, se suele utilizar solo en aplicaciones muy específicas.

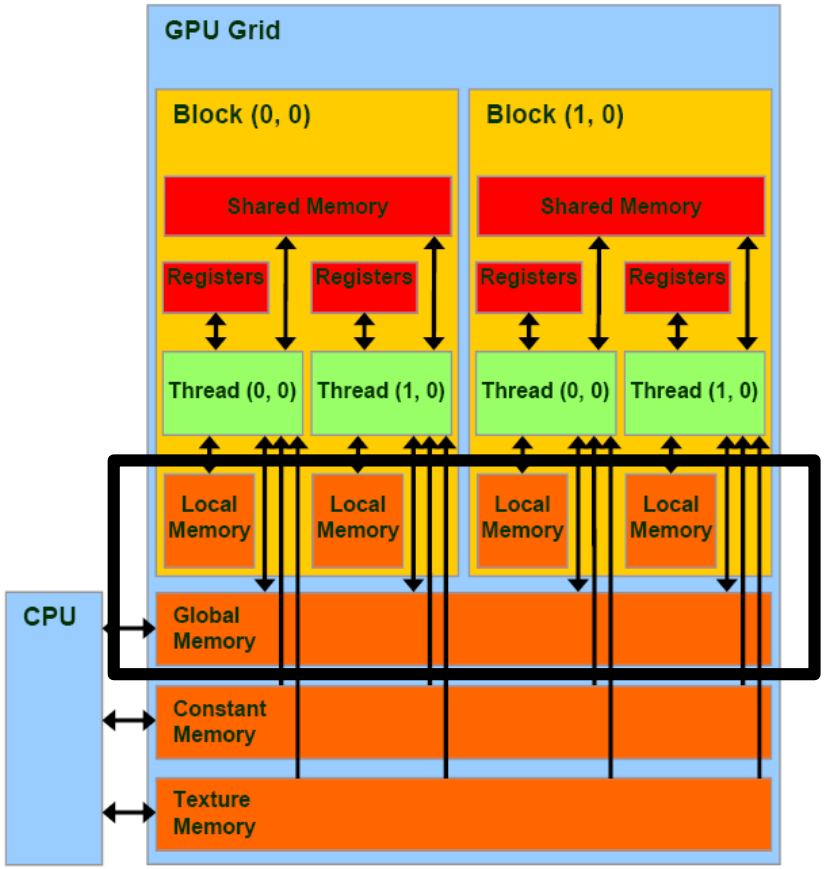


Memoria Local y Global

La memoria local presenta las mismas reglas que los registros. Solo es visible para una hebra y dura mientras esa hebra viva.

Sin embargo, es una memoria mucho más lenta.

Es aquí donde se almacenan las variables que no caben en memoria de registros.

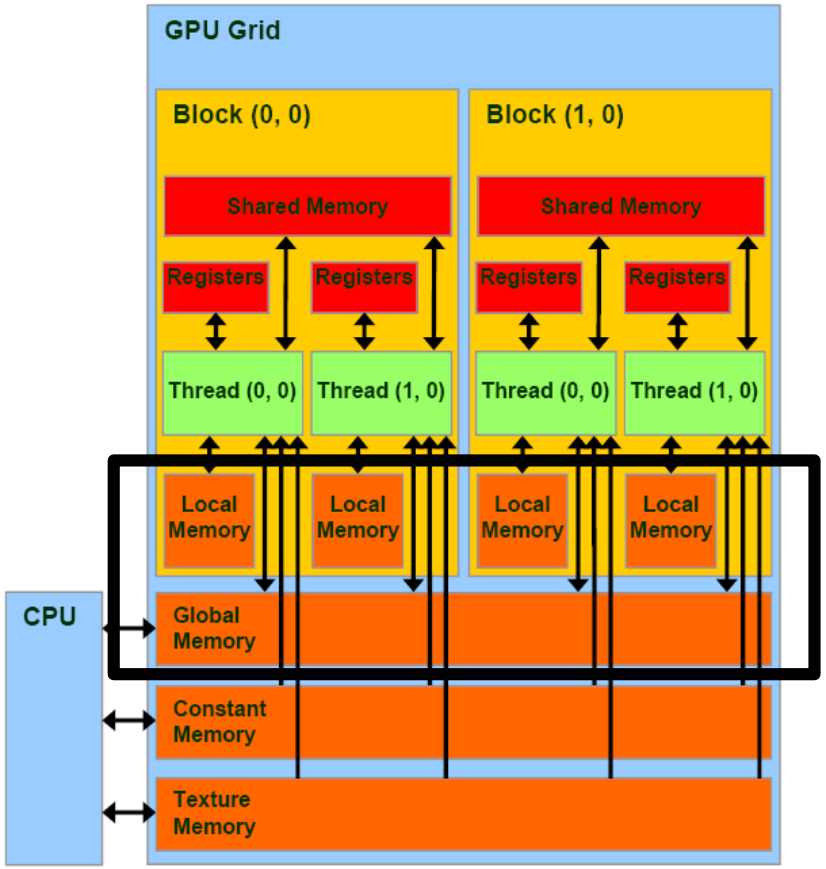


Memoria Local y Global

La memoria global, si bien es la más lenta de todas junto con la memoria local, es también la más importante.

Esta memoria es vista por todas las hebras y la CPU, al igual que la memoria constante y las texturas, pero no es de solo lectura.

Es la única memoria en la cual se pueden almacenar valores calculados en un kernel y la única que nos servirá para llevar datos a la CPU.



Espacio Físico de Memorias