**Universidad de Córdoba**

Curso 2010/2011

**Iniciación y pruebas de rendimiento con CUDA**

**Diseño de Procesadores y Evaluación de Computadores**

**Raúl Pérula Martínez**

[88x31.png](http://creativecommons.org/licenses/by-sa/4.0/)

This work by [Raúl Pérula-Martínez](http://educatech.sytes.net/raul/) is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/).

Índice de contenido

[Introducción 1](#_Toc288045003)

[Qué es CUDA 1](#_Toc288045004)

[Definición 1](#_Toc288045005)

[Utilidad 1](#_Toc288045006)

[Objetivos 4](#_Toc288045007)

[Características técnicas específicas del equipo 5](#_Toc288045008)

[CPU 5](#_Toc288045009)

[Especificaciones 5](#_Toc288045010)

[GPU 7](#_Toc288045011)

[Especificaciones 7](#_Toc288045012)

[Pruebas de rendimiento 10](#_Toc288045013)

[Aspectos iniciales 10](#_Toc288045014)

[Pruebas en CPU y GPU 13](#_Toc288045015)

[Bibliografía 14](#_Toc288045016)

[Apéndice A: Instalación de CUDA 15](#_Toc288045017)

[Instalar CUDA 15](#_Toc288045018)

[Requisitos 15](#_Toc288045019)

[Instalación 15](#_Toc288045020)

[CUDA SDK 17](#_Toc288045021)

[Prueba de los Ejemplos 17](#_Toc288045022)

[Apéndice B: Código fuente 19](#_Toc288045023)

# Introducción

## Qué es CUDA



### Definición

CUDA, de nVidia, es una extensión del lenguaje C que permite utilizar el poder de computación en paralelo de las tarjetas gráficas para propósitos de lo más variado ofreciendo una mejora de hasta 100x en la eficiencia.

### Utilidad

En la actualidad, decenas de miles de desarrolladores, científicos, estudiantes, creadores de juegos e investigadores escriben aplicaciones que aprovechan la potencia de procesamiento de la GPU en campos tan dispares como la producción de videojuegos con efectos de física, el análisis de riesgos en el mercado de valores, el análisis de datos sísmicos y la predicción meteorológica.

También el mundo académico ha reconocido el increíble potencial de esta arquitectura computacional. El cálculo en la GPU basado en CUDA se ha incluido en los planes de estudios de más de 200 universidades, entre ellas, el MIT, Harvard, Cambridge, Oxford, los Institutos Indios de Tecnología, la Universidad Nacional de Taiwán y la Academia China de las Ciencias.

**CUDA en investigación**

****

Los virus, causantes de numerosas enfermedades, son los organismos más pequeños que produce la naturaleza. Debido a su simplicidad y su pequeño tamaño, los biólogos eligieron un virus para su primer intento de simular una forma de vida completa a través de un sistema informático y para ello seleccionaron uno de los más pequeños del planeta, el virus satélite del mosaico del tabaco. El proyecto consistió en simular el virus en una gota de agua salada utilizando un programa de dinámica molecular llamado NAMD (Nanoscale Molecular Dynamics) desarrollado por la Universidad de Illinois en Urbana-Champaign.

El rendimiento de NAMD se ha multiplicado por 12 al acelerar la aplicación con CUDA™ en un cluster de GPUs instalado en el Centro Nacional de Aplicaciones de Supercomputación (NCSA), lo que ha proporcionado un incremento de la velocidad de hasta 330 veces con respecto al uso de la CPU. Los investigadores contemplan este avance con optimismo ya que consideran que ayudará a la medicina moderna a comprender mejor las enfermedades virales y tratarlas con más eficacia.

[www.ks.uiuc.edu/Research/namd/](http://www.ks.uiuc.edu/Research/namd/)

**CUDA en video y fotografía**



A medida que se extiende el uso de los reproductores multimedia, aumenta el deseo de los consumidores de ver sus vídeos en estos dispositivos y también la frustración por las largas esperas que conlleva el proceso de conversión de formatos. Por ejemplo, convertir una película de 120 minutos, puede tardar seis o más horas si se utiliza la CPU del sistema. Badaboom es un programa de transcodificación de vídeo de Elemental que convierte formatos de vídeo estándar en formatos reproducibles en el iPod y otros dispositivos portátiles.

Al utilizar el sistema de procesamiento CUDA™ de las GPUs NVIDIA, Badaboom puede realizar ese proceso hasta 18 veces más rápido, con lo que la conversión se realiza en minutos en lugar de horas. Además, durante ese tiempo la CPU está libre para realizar otras tareas como leer el correo electrónico o navegar por la web.

**CUDA en medicina**



La capacidad para generar imágenes detalladas en muy corto espacio de tiempo es particularmente crítica en exploraciones del cáncer de mama. TechniScan, empresa que desarrolla sistemas automatizados de generación de imágenes por ultrasonido, ha trasladado su algoritmo de cálculo en CPU a un sistema basado en CUDA™ y las GPU NVIDIA® Tesla™.

El nuevo sistema basado en CUDA es capaz de procesar el algoritmo de Techniscan en un tiempo inferior a 20 minutos, tres veces menos de lo que tardaba el sistema anterior. Una vez que la FDA haya dado la aprobación definitiva a este dispositivo de investigación, los pacientes podrán recibir sus resultados con una sola visita.

Más información en [www.techniscanmedicalsystems.com](http://www.techniscanmedicalsystems.com) [6].

# Objetivos

Con este trabajo se quiere intentar mejorar los resultados computacionales obtenidos por un programa con gran carga en ejecución como es la multiplicación de matrices aplicando métodos de transformación de código para la optimización vistos en la asignatura Diseño de Procesadores y Evaluación de Configuraciones tales como **loop tiling** o **array padding (inter-array o intra-array)** y comparar los resultados con los obtenidos en la ejecución paralela realizada por la GPU, programando en CUDA.

También se intentará buscar las dimensiones adecuadas para optimizar los tamaños de bloque y el número de hilos de ejecución en CUDA para que el tiempo de ejecución en la GPU sea el mínimo.

# Características técnicas específicas del equipo

## CPU



Intel® Core™2 Duo Processor T7300 (4M Cache, 2.00 GHz, 800 MHz FSB)

### Especificaciones

|  |  |
| --- | --- |
| **Essentials** | |
|  | |
| Status | Launched |
| Launch Date | Q2'07 |
| Processor Number | T7300 |
| # of Cores | 2 |
| # of Threads | 2 |
| Clock Speed | 2 GHz |
| L2 Cache | 4 MB |
| Bus/Core Ratio | 10.0 |
| FSB Speed | 800 MHz |
| FSB Parity | No |
| Instruction Set | 64-bit |
| Embedded Options Available | No |
| Supplemental SKU | No |
| Lithography | 65 nm |
| Max TDP | 35 W |
|  |  |
| **Package Specifications** |  |
|  |  |
| TJUNCTION | 100°C |
| Package Size | 35mm x 35mm |
| Processing Die Size | 143 mm2 |
| # of Processing Die Transistors | 291 million |
| Sockets Supported | PBGA479, PPGA478 |
| Halogen Free Options Available | No |
| Advanced Technologies | |
| Intel® Turbo Boost Technology | No |
| Intel® Hyper-Threading Technology | No |
| Intel® Virtualization Technology (VT-x) | Yes |
| Intel® Trusted Execution Technology | No |
| Intel® 64 | Yes |
| Idle States | Yes |
| Enhanced Intel SpeedStep® Technology | Yes |
| Intel® Demand Based Switching | No |
| Execute Disable Bit | Yes |

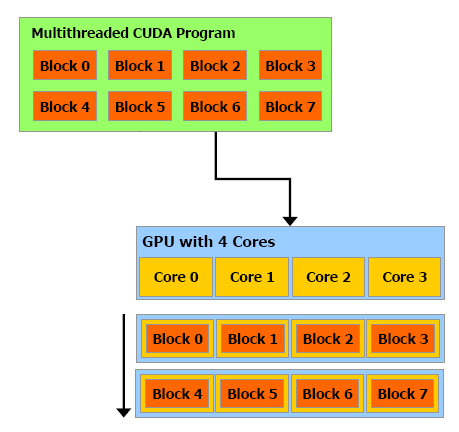
|  |  |
| --- | --- |
| **Architecture / Microarchitecture** | |
|  | |
| Processor core | Merom |
| Core steppings | B0 (QLYV)  E1 (QXJL, SLA45)  G0 (SLAMD) |
| Manufacturing process | 0.065 micron |
| Data width | 64 bit |
| Number of cores | 2 |
| Floating Point Unit | Integrated |
| Level 1 cache size | 2 x 32 KB instruction caches  2 x 32 KB write-back data caches |
| Level 2 cache size | shared 4 MB |
|  |  |
| Features | MMX instruction set  SSE  SSE2  SSE3  Supplemental SSE3  EM64T technology  Execute Disable Bit technology  Virtualization Technology  Dynamic Acceleration technology |
|  |  |
| Low power features | Stop Grant mode  Sleep mode  Deep Sleep mode  Deeper Sleep mode  Enhanced Deeper Sleep mode  Dynamic Cache sizing  Enhanced SpeedStep technology  Dynamic FSB frequency switching |
|  |  |
| **Electrical/Thermal parameters** | |
|  | |
| V core (V) | 1.0375 - 1.3 |
| Minimum/Maximum operating temperature (°C) | 0 - 100 |
| Minimum/Maximum power dissipation (W) | 7.1 (Enhanced Deeper Sleep mode)/ 53.3 |
| Thermal Design Power (W) | 35 |
|  |  |
| **Notes on Intel LF80537GG0414M** | |
|  | |
| Bus frequency is 200 MHz. Because the processor uses Quad Data Rate bus the effective bus speed is 800 MHz. | |

## GPU



### Especificaciones

|  |  |
| --- | --- |
| Número de multiprocesadores | 4 |
| Número total de núcleos | 32 |
| Reloj central (MHz) | 475 |
| Reloj de las unidades de sombreado (MHz) | 950 |
| Reloj de la memoria (MHz) | 700 |
| Cantidad de memoria | 512MB |
| Interfaz de memoria | 128-bit |



**Arquitectura unificada de NVIDIA®**

* Arquitectura de sombreadores unificada
* Tecnología GigaThread™
* Soporte completo de Microsoft DirectX 10
  + Sombreadores de geometría
  + Instanciación de la geometría
  + Salida de shaders a la memoria de vídeo
  + Shader Model 4.0
* Precisión de coma flotante de 128 bits en todo el canal de renderizado

**Motor NVIDIA Lumenex™**

* Antialiasing 16x en pantalla completa
* Multimuestreo de transparencias y supermuestreo de transparencias
* Filtrado anisótropo 16x independiente del ángulo
* Iluminación de alto rango dinámico (HDR) en coma flotante de 128 bits con antialiasing:
  + Filtrado y mezcla de texturas en coma flotante de 32 bits por componente
* Algoritmos de compresión avanzados sin pérdida de datos para colores, texturas y datos Z
* Soporte de compresión de mapas de normales
* Z-cull: supresión de píxeles que no pasan la prueba de profundidad de Z
* Early-Z

**Tecnología NVIDIA SLI™1**

* Tecnología de hardware y software patentada por NVIDIA que permite utilizar dos tarjetas gráficas GeForce en paralelo para multiplicar el rendimiento y mejorar la calidad de imagen en los juegos más populares del mercado

**Tecnología NVIDIA PureVideo™ HD2**

* Procesador de vídeo dedicado en el chip
* Descodificación acelerada de los formatos de alta definición H.264, VC-1, MPEG2 y WMV9
* Desentrelazado espaciotemporal avanzado
* Conforme con HDCP3
* Reducción de ruido
* Mejora de los contornos
* Corrección de errores de edición
* Telecine inverso (corrección pull-down 2:2 y 3:2)
* Adaptación a diferentes resoluciones con máxima calidad
* Corrección del color de vídeo
* Soporte de Microsoft® Video Mixing Renderer (VMR)

**Funciones avanzadas de visualización**

* Una salida DVI Dual Link para pantallas digitales planas con resoluciones de hasta 2560 x 1600
* Dos módulos RAMDAC integrados de 400 MHz proporcionan una resolución de imagen máxima de 2048 x 1536 a 85 Hz
* Codificador de HDTV integrado que proporciona salida a TV analógica (vídeo por componentes/compuesto/S-Video) con resoluciones de hasta 1080i/1080p
* Funciones de visualización multipantalla NVIDIA nView
* Procesamiento de 10 bits

**Interfaces de alta velocidad**

* Diseñada para el bus PCI Express x16
* Diseñada para memoria GDDR3 de alta velocidad y DDR2 de menor coste

**Sistemas operativos**

* Diseñada para Windows Vista
* Windows XP/Windows XP 64
* Linux

# Pruebas de rendimiento

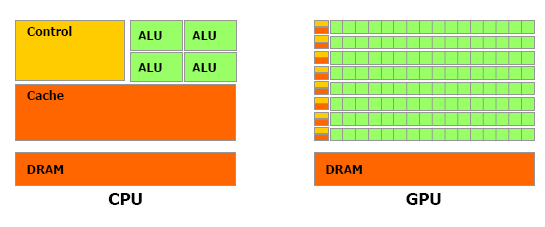
## Aspectos iniciales

Se utilizará una matriz de dimensión 2000x2000. A continuación se muestran las especificaciones de la tarjeta gráfica proporcionadas por CUDA.

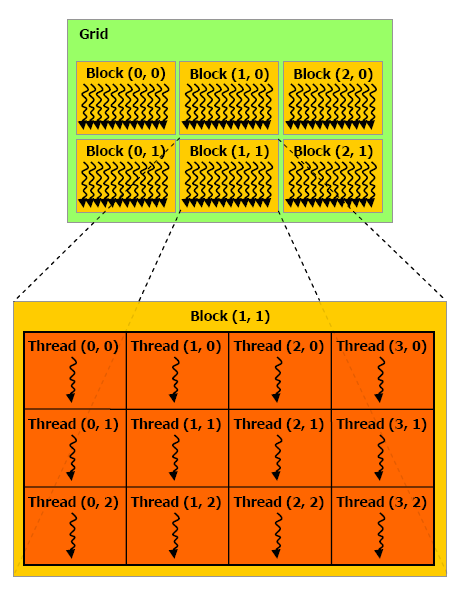
**Especificaciones proporcionadas por CUDA**

|  |  |
| --- | --- |
| CUDA Driver Version: | 3.20 |
| CUDA Runtime Version: | 3.20 |
| CUDA Capability Major/Minor version number: | 1.1 |
| Total amount of global memory: | 536543232 bytes (512 MB) |
| Multiprocessors x (Cores/MP) = Cores: | 4 (MP) x 8 (Cores/MP) = 32 (Cores) |
| Total amount of constant memory: | 65536 bytes (64 KB) |
| Total amount of shared memory per block: | 16384 bytes (16 KB) |
| Total number of registers available per block: | 8192 |
| Warp size: | 32 |
| Maximum number of threads per block: | 512 |
| Maximum sizes of each dimension of a block: | 512 x 512 x 64 |
| Maximum sizes of each dimension of a grid: | 65535 x 65535 x 1 |
| Maximum memory pitch: | 2147483647 bytes |
| Texture alignment: | 256 bytes |
| Clock rate: | 0.95 GHz |
| Concurrent copy and execution: | Yes |
| Run time limit on kernels: | Yes |
| Integrated: | No |
| Support host page-locked memory mapping: | Yes |
| Compute mode: | Default (multiple host threads can use this device simultaneously) |
| Concurrent kernel execution: | No |
| Device has ECC support enabled: | No |
| Device is using TCC driver mode: | No |

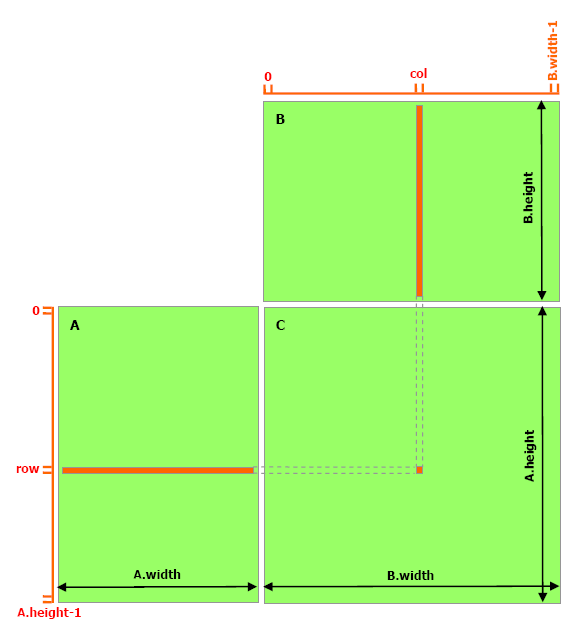
Imágenes de la organización de la memoria:



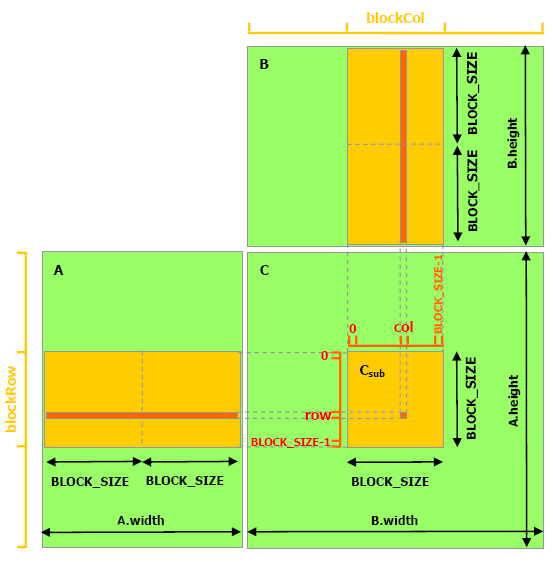
Niveles de memoria en CPU y GPU.



Definición de una rejilla y de los hilos que hay en los bloques.



Multiplicación de matrices sin compartición de memoria.



Multiplicación de matrices con compartición de memoria.

## Pruebas en CPU y GPU

# Bibliografía

1. Características técnicas de Intel Core 2 Duo Processor T7300. Página web oficial Intel. Última visita: Noviembre de 2010.

Enlace: <http://ark.intel.com/Product.aspx?id=29760>

1. Características técnicas de Nvidia GeForce 8600M GT. Página web oficial Nvidia. Última visita: Noviembre de 2010.

Enlace: <http://www.nvidia.es/page/geforce_8600M.html>

1. Cómo programar con CUDA. A.E.C. Siglo XXI. Última visita: Noviembre de 2010.

Enlace: <http://sigloxxi.fcie.uam.es/maquinaciones/como-programar-con-cuda>

1. Especificaciones técnicas y aclaraciones tecnología procesador Intel T7300. Última visita: Diciembre de 2010.

Enlace: <http://www.cpu-world.com/CPUs/Core_2/Intel-Core%202%20Duo%20Mobile%20T7300%20LF80537GG0414M%20%28BX80537T7300%29.html>

1. NVIDIA CUDA Library Documentation 3.1 Beta. Última visita: Diciembre de 2010.

Enlace: <http://developer.download.nvidia.com/compute/cuda/3_1/toolkit/docs/online/index.html>

1. CUDA en acción. Página web oficial Nvidia. Última visita: Diciembre de 2010.

Enlace: <http://www.nvidia.es/object/cuda_in_action_es.html>

# Apéndice A: Instalación de CUDA

En este apéndice se va a explicar cómo instalar CUDA en un sistema Ubuntu (Linux) 10.04 LTS paso a paso indicando en todo momento como ejecutar la instalación y las posibles dificultades que se pueden dar.

## Instalar CUDA

A continuación, se muestra qué y cómo instalar para poder empezar a usar CUDA.

### Requisitos

Lo primero que se necesita saber para usar CUDA es si se tiene una tarjeta gráfica de nVidia compatible. Para saber si vuestra tarjeta sirve, podéis consultar la [lista de tarjetas compatibles con CUDA](http://www.nvidia.com/object/cuda_learn_products.html).

En el resto de apartados se supone que estamos usando Ubuntu Linux 10.04 LTS, aunque debería servir tal cual para el resto de las principales distribuciones y prácticamente igual para Windows (aunque ese caso no se va a tratar).

### Instalación

Para poder usar CUDA se necesitan tres elementos: un driver para la tarjeta gráfica, el toolkit de desarrollo y la librería glut. Además, se va a instalar el SDK con ejemplos para poder comprobar que todo funciona bien y para ver qué tipo de cosas se pueden hacer. Todo lo necesario mencionado con anterioridad se encuentra disponible en la [página de descargas de CUDA](http://www.nvidia.es/object/cuda_get_es.html) de nVidia.

**Driver de la Gráfica**

Nota: Esta es probablemente la parte más delicada de todo el proceso.

En este caso (Ubuntu 10.04, nVidia 8600M GT), los drivers instalados dan aceleración 3D pero no soporte a CUDA. Con lo que hubo que desinstalarlos e instalar los que se ofrecen en la [página de descargas de CUDA](http://www.nvidia.es/object/cuda_get_es.html).

Para desinstalar se necesita salir del modo gráfico (pulsando [Ctrl]+[Alt]+[F1] y logueándose de nuevas) y matar el proceso de KDE, GNOME o lo que se use (se necesitan privilegios):

|  |
| --- |
| # killall kdm ó killall gdm ó killall xdm |

Una vez hecho eso, se buscan los paquetes de nvidia instalados:

|  |
| --- |
| # aptitude search nvidia | grep -E ^i |

Y para cada uno de ellos, se desinstalan:

|  |
| --- |
| # aptitude remove [nombre\_del\_paquete] |

Por suerte, la **instalación** es aún más sencilla. Una vez se haya bajado el fichero de la página de descargas (algo similar a devdriver\_3.2\_linux\_32\_260.19.14.run), sólo hay que modificar los permisos para hacerlo ejecutable y ejecutarlo. Estando en la carpeta donde se encuentra el fichero:

|  |
| --- |
| # chmod +x devdriver\_3.2\_linux\_32\_260.19.14.run  # ./devdriver\_3.2\_linux\_32\_260.19.14.run |

Y el asistente se debería encargar de todo. Si todo ha ido bien, ejecutando de nuevo el gestor gráfico kdm, gdm o xdm o reiniciando, se debería volver a ver el escritorio tal y cómo se encontraba antes.

Si ha habido algún problema, siempre se puede revertir al estado anterior:

1. Ejecutando nvidia-uninstall como root.
2. Volviendo a instalar los mismos paquetes que se habían desinstalado.
3. Tomando alguno de los backups en **/etc/X11/** y renombrándolo a **xorg.conf**.

**Toolkit de Desarrollo**

Este paso es más sencillo y seguro de instalar. Sólo hay que descargar el fichero de la página de descargas (con un nombre parecido a **cudatoolkit\_3.2.12\_linux\_32\_ubuntu10.04.run**), darle permisos de ejecución como en el paso anterior y ejecutarlo.

El asistente se encarga de casi todo y sólo pedirá la carpeta donde se desea instalar. Si se escoge una carpeta que no sea la que pone por defecto es importante que se recuerde la ubicación.

Una vez terminada la instalación, hay que añadir los ejecutables y las librerías a los paths correspondientes. Como se comenta en la página de nVidia, hay que añadir a ~/.bashrc las siguientes líneas:

|  |
| --- |
| PATH=$PATH:/bin  LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:/lib  export PATH  export LD\_LIBRARY\_PATH |

donde se encuentra la carpeta en donde se ha instalado el toolkit.

**Librería glut**

Esta librería sirve para poder programar con OpenGL y también se necesita para CUDA. Tan sólo hay que instalar un paquete (si es que no se tiene de antes):

|  |
| --- |
| # aptitude install libglut3-dev |

### CUDA SDK

Con lo que se ha instalado se puede empezar a desarrollar con CUDA. Es recomendable empezar descargando e instalando el SDK de la página de descargas, éste tendrá un nombre parecido a **gpucomputingsdk\_3.2.12\_linux.run** y se instalará igual que el toolkit.

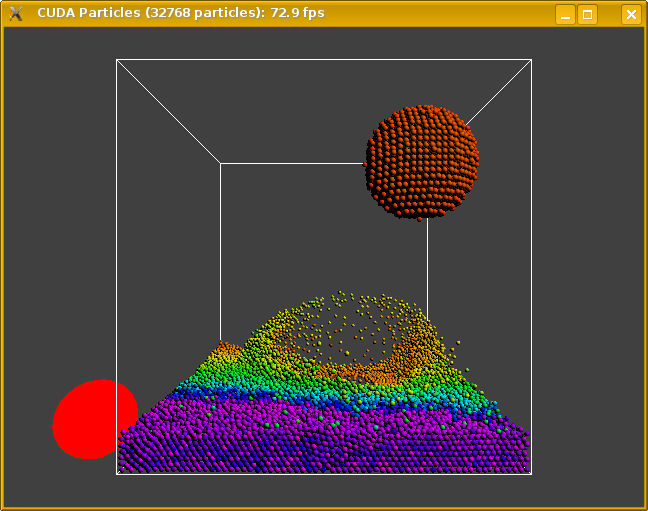
### Prueba de los Ejemplos

Para compilar los ejemplos hay que ir a la carpeta generada tras la instalación, entrar en la subcarpeta llamada C y ejecutar:

|  |
| --- |
| # make |

Si la compilación da problemas, es recomendable buscar el error que dé (por ejemplo en Google) y seguro que encontrareis la solución.

Una vez terminado todo, se podrán ejecutar los ejemplos que vienen.



# Apéndice B: Código fuente