

Jetson hardware integrado

NVIDIA

CASTILLO CUADRA FABIAN ALONSO
Ingeniería de sistemas
Universidad industrial de Santander
Bucaramanga-Santander

DURAN DURAN JAVIER FERNANDO
Ingeniería de sistemas
Universidad industrial de Santander
Bucaramanga-Santander

Resumen- En este documento se expone a grandes rangos el hardware integrado jetson de nvidia, que gracias a su procesamiento paralelo acelerado por GPU lo hace ideal para proyectos integrados informáticas como aviones no tripulados, sistemas robóticos autónomos, la proyección de imagen médica móvil y análisis de Video inteligente (IVA).

Palabras clave-nvidia; jetson; GPU; paralelo.

Abstract—

In this document it is exposed to large ranges the integrated hardware jetson from nvidia, which makes it ideal for integrated projects with parallel processing GPU-accelerated computer as unmanned, autonomous robotic systems, the mobile medical imaging and analysis of Intelligent Video (IVA).

Keywords—nvidia; jetson; GPU; parallel.

Introducción

La computación embebida es el próximo gran avance; los incrementos de las GPU hoy en día ayudan a acelerar el ritmo de la innovación y así mismo ofrecen beneficios significativos en el campo de la visión artificial, robótica, automoción, procesamiento de imágenes de señales, seguridad de red, la medicina, y muchos otros. En este documento se aborda el sistema embebido Jetson, el cual está dividido en dos tecnologías (TX1, TK1); se expondrá esta tecnología y así mismo se nombrarán y se describirán de una forma general el uso y los componentes de esta tecnología. todos los temas tratados son con fines académicos para la materia arquitectura de computadores.

ESTADO DEL ARTE

Durante el desarrollo de la conferencia sobre GPU celebrada en san José California entre el 24 al 27 de marzo el CEO de nvidia Jen-Hsun Haung anuncio la supercomputadora más pequeña del mundo(jetson TK1), con un rendimiento de 365 gigaflops que incluye un kit de herramientas completo en C/C++, basado en CUDA, siendo diseñado para usarse en robótica, carros autónomos y en computadores que simulan reconocimiento humano u objetos físicos, aunque también fue pensado para ser usado en proyectos tipo maker o en ferias de la ciencia.

Sistema embebido

Un sistema embebido es un sistema de computación diseñado para realizar una o algunas pocas funciones dedicadas que por lo general son en tiempo real. En un sistema embebido la mayoría de los componentes se encuentran incluidos en la placa base (tarjeta de vídeo, audio, módem, etc.) y muchas veces los dispositivos resultantes no tienen el aspecto de lo que se suele asociar a una computadora. Algunos ejemplos de sistemas embebidos podrían ser dispositivos como un taxímetro, un sistema de control de acceso, la electrónica que controla una máquina expendedora o el sistema de control de una fotocopiadora entre otras múltiples aplicaciones. [1]

En el software embebido se pueden distinguir diferentes categorías: El original o básico, indispensable para el funcionamiento del aparato, el cual constituye su sistema operativo ad-hoc. El lenguaje de programación assembler, C/C++ o VHDL. Este tipo de software, de carácter eminentemente tecnológico, está incorporado en el

aparato desde la salida al mercado de este último. En general, requiere un elevado esfuerzo de creación inicial, a cargo de mano de obra de muy alta especialización. Su costo se reparte en la cantidad de equipos entregados al mercado. En la mayor parte de los casos no se modifica a lo largo de su vida activa, siendo reemplazado al aparecer un nuevo modelo de producto. Un segundo tipo es el requerido por alguno de los equipos incluidos en el aparato anterior, pero que, por condiciones específicas, como pueden ser requerimientos regulatorios de una determinada comunidad, modalidades o costumbres, se hace necesario adaptar el equipo original al uso específico requerido. Otro tipo es el dedicado a la supervisión y control de sistemas complejos cuyo funcionamiento implica la interrelación de gran cantidad de equipos o aparatos, por ejemplo, el funcionamiento y control de los distintos procesos en la industria manufacturera [2].

Características del software embebido

Los tres atributos que típicamente tienen consideraciones especiales en el desarrollo de software embebido son: confiabilidad, limitaciones en recursos de hardware y respuesta en tiempo real [4]. La figura 2 ilustra y explica dichos atributos.

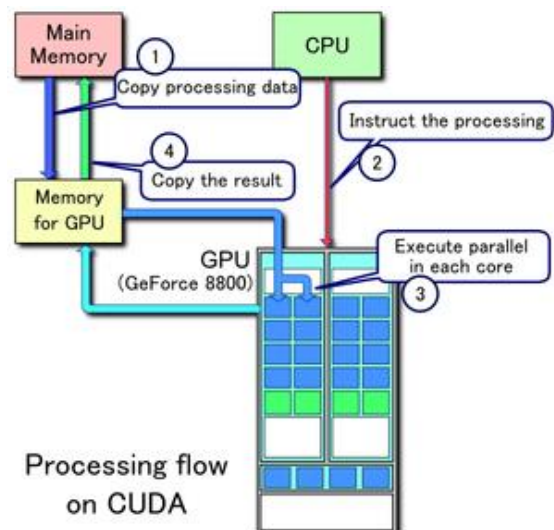
CONFIABILIDAD	Las fallas no pueden ser resueltas con ayuda del usuario (No al Ctrl-Alt-Del). Las expectativas de los usuarios son más altas. Las consecuencias de una falla pueden ser muy graves, incluso la pérdida de vidas (control industrial, aeronáutico, automotriz).
RECURSOS DE HARDWARE LIMITADOS	Memoria limitada. Energía limitada, bajo consumo de energía. Fuentes de energía alternativa inestables (celdas solares, movimiento).
RESPUESTA EN TIEMPO REAL	Las operaciones del software deben ejecutarse en periodos exactos (hard real time). Las operaciones del software deben ejecutarse en periodos limitados (soft real time).

Figura 2. Atributos del software embebido

CUDA

CUDA es una arquitectura de cálculo paralelo de NVIDIA que aprovecha la gran potencia de la GPU (unidad de procesamiento gráfico) para proporcionar un incremento extraordinario del rendimiento del sistema. [3].

La plataforma CUDA está diseñado para trabajar con lenguajes de programación como C, C++ y Fortran. Esta accesibilidad hace que sea más fácil para los especialistas en la programación paralela de utilizar recursos de la GPU, a diferencia de las soluciones anteriores de la API Direct3D y OpenGL como, lo que requería conocimientos avanzados en programación de gráficos. Además, es compatible con CUDA marcos de programación tales como OpenACC y OpenCL. [4] Cuando se introdujo por primera vez por NVIDIA, CUDA el nombre era un acrónimo de Compute Unified Device Architecture, [5] pero posteriormente NVIDIA dejó caer el uso de la sigla.



Flujo de Proceso de CUDA

Ventajas

CUDA presenta ciertas ventajas sobre otros tipos de computación sobre GPU utilizando APIs gráficas.

- Lecturas dispersas: se puede consultar cualquier posición de memoria.
- Memoria compartida: CUDA pone a disposición del programador un área de memoria de 16KB (o 48KB en la serie Fermi) que se compartirá entre threads. Dado su tamaño y rapidez puede ser utilizada como caché.
- Lecturas más rápidas de y hacia la GPU.
- Soporte para enteros y operadores a nivel de bit.[6]

usos actuales y futuros de la arquitectura CUDA

- procesamiento acelerado de los gráficos en 3D
- interconversión acelerada de formatos de archivos de vídeo
- Aceleración cifrada, descifrada y compresión
- Bioinformática, por ejemplo, NGS secuenciación de ADN Barracuda
- cálculos distribuidos, tales como la predicción de la conformación nativa de las proteínas
- simulaciones de análisis médicos, por ejemplo, de realidad virtual basado en la TC y la resonancia magnética de imágenes.
- simulaciones físicas, en particular en la dinámica de fluidos.
- formación de redes neuronales en problemas de aprendizaje automático
- Reconocimiento facial
- Computación distribuida
- La dinámica molecular [7]

Arquitectura CUDA

Un multiprocesador contiene ocho procesadores escalares, dos unidades especiales para funciones trascendentales, una unidad multihilo de instrucciones y una memoria compartida. El multiprocesador crea y maneja los hilos sin ningún tipo de overhead por la planificación, lo cual unido a una rápida sincronización por barreras y una creación de hilos muy ligera, consigue que se pueda utilizar CUDA en problemas de muy baja granularidad.

Nvidia Jetson TK1

Jetson TK1 es la plataforma de desarrollo de NVIDIA Linux embebido que ofrece un Tegra SOC K1 (CPU + GPU + ISP en un solo chip) , viene pre -instalado con Linux4Tegra OS (básicamente Ubuntu 14.04 con los controladores pre configurados); Posee cuatro núcleos a 2,3 GHz ARM Cortex - A15 CPU y la GPU revolucionaria Tegra K1 , la Jetson TK1 incluye características similares a los de un Frambuesa Pi , pero también algunas características orientadas PC - tales como SATA , mini - PCIe y un ventilador para permitir la continua operación bajo fuertes cargas de trabajo. [8]



Jetson TK1 Development board vista superior

Nvidia Jetson TK1 Developer kit

El Kit de desarrollo de TK1 Jetson está diseñado alrededor de la revolucionaria 192-core NVIDIA Tegra K1 procesador móvil, junto con 2 GB de RAM, 16 GB de almacenamiento a bordo y numerosos periféricos y puertos de la entrada-salida. Tegra K1 se basa en la misma arquitectura GPU de NVIDIA Kepler utilizada en supercomputadores y sistemas de computación de alto rendimiento alrededor del mundo, además de esto el kit de desarrollo de Jetson ofrece una plataforma de NVIDIA CUDA completamente funcional e incluye el paquete de soporte de Junta, CUDA 6, OpenGL 4.4 y el toolkit de NVIDIA VisionWorks. Con una completa suite de desarrollo y herramientas de generación de perfiles, además de fuera de la caja soporte para cámaras y otros periféricos. [9]

Características del hardware

Dimensiones: 5'' * 5''[board]

Tegra K1 SOC (CPU + GPU + ISP en un solo chip, con un consumo de energía típico entre 1 a 5 vatios):

GPU: NVIDIA Kepler GPU " GK20a " con 192 núcleos CUDA SM3.2 (hasta 326 GFLOPS)

CPU: NVIDIA " 4 - Plus - 1 " CPU 2.32GHz ARM de cuatro núcleos Cortex-A15 -A15 con la corteza de ahorro de batería sombra núcleos

DRAM: 2 GB DDR3L 933MHz x16 EMC utilizando anchura de datos de 64 bits

Almacenamiento: 16 GB rápida eMMC 4,51 (ruteado a SDMMC4)

USB 3.0

USB 2.0

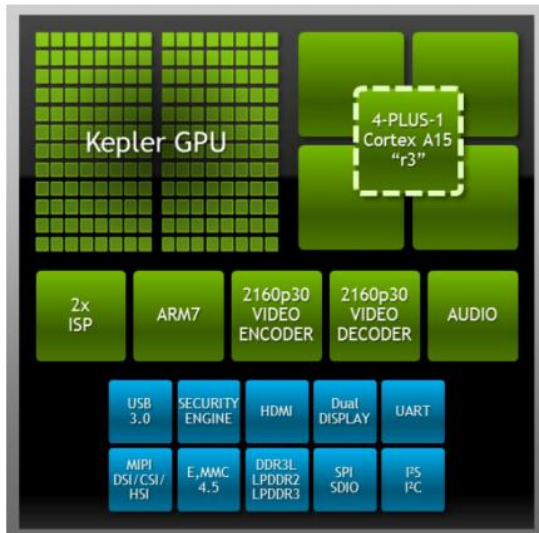
HDMI

Audio: un códec de audio Realtek HD ALC5639 con micrófono en línea y tomas de salida (una ruta a DAP2)

Ethernet: un puerto / 100BASE - T Gigabit LAN Realtek RTL8111GS 10/100 utilizando PEX

SATA: un puerto de tamaño completo que soporta 2,5 " y 3,5" discos, pero no es de acoplamiento activo.

Potencia: un conector de alimentación de 12V DC barril y un conector de alimentación IDE PC de 4 pines, utilizando PMIC AS3722



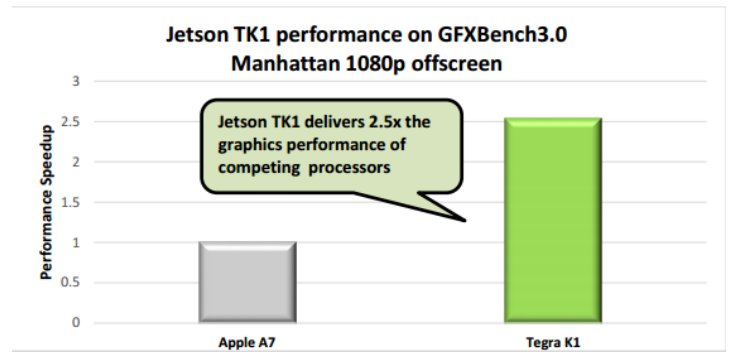
Procesador móvil NVIDIA Tegra K1 (32-bit versión)

Jetson TK1 - que ofrece un rendimiento excepcional entre poder y eficiencia

La arquitectura de la GPU Kepler en Tegra K1 es prácticamente idéntica a la arquitectura Kepler GPU utilizado en los sistemas de gama alta, pero también incluye una serie de optimizaciones para el uso del sistema móvil para conservar potencia y rendimiento de la GPU móvil líder en la industria. Mientras que el más alto de gama de Kepler en las GPU escritorio, estaciones de trabajo y supercomputadoras incluyen hasta 2.880 punto flotante precisión simple en CUDA.

Jetson TK1 con Tegra K1 incluye 192 núcleos CUDA y consume energía significativamente menor.

Cuando se compara con los procesadores móviles de la generación actual, la plataforma Tegra Jetson TK1 K1 funcionando entrega casi 2,5 veces el máximo rendimiento que su competencia en procesadores móviles (comparativa del año 2014).



Comparativo año 2014[10]

NVIDIA Jetson TX1

El Jetson TX1 es un dispositivo de bajo costo para sistemas embebidos que implementa un supercomputador en un módulo del tamaño de una tarjeta de crédito. Viene integrada con un procesador ARM de 64 bits junto con una GPU de arquitectura NVIDIA Maxwell pensada en algoritmos de redes neuronales y con una capacidad de cálculo de 1 TeraFlop.



NVIDIA Jetson TX1 Developer Kit

El kit de desarrollo del NVIDIA Jetson TX1 es una plataforma basada en el módulo TX1 diseñada para implementar soluciones de computación visual en tiempo mínimo. Esta viene con un entorno Linux y

con la gama de herramientas de desarrollo NVIDIA con numerosas interfaces de hardware.



CARACTERÍSTICAS

El módulo TX1 viene con una GPU de 256 núcleos, arquitectura Maxwell, procesador ARM Cortex – A57MPCore de cuatro núcleos, 4 GB de memoria LPDDR4, 16 GB de espacio de almacenamiento flash eMMC 5.1, conexión con dispositivos Wi-Fi 802.11ac y Bluetooth, Ethernet 10/100/1000Base-T, cámara de 5 MP y enfoque fijo con interfaz MIPI CSI. [11]

DISPOSITIVOS I/O

El dispositivo cuenta con entradas USB 3.0 tipo A, USB 2.0 Micro-AB, HDMI, M.2 tipo E, PCI-E x4, Gigabit Ethernet, SD, Datos y alimentación SATA, GPIOs, I2C, I2S, SPI, UART TTL para control de flujos, terminal de expansión de tamaño y cámara.

SOPORTE SOFTWARE

Linux4Tegra R24.1 para ARM (Ubuntu 14.04 32 bits y 64bits), JetPack 2.2, CUDA Toolkit 7, cuDNN v5, VisionWorks 1.4, OpenCV4Tegra 2.4.13, OpenGL 4.4, OpenGL ES 3.1, Vulkan, V4L2 media controller, gstreamer / OpenMAX, Tegra System, Tegra Graphics.

APLICACIONES

Debido a la arquitectura del módulo Jetson TX1 resulta útil para el desempeño y eficiencia en

sistemas embebidos con aplicaciones de: visión por computadora, deep learning, robótica, GPU computing. Por tanto, su uso se extiende a sistemas como: drones, reconocimiento de imágenes médicas, sistemas robóticos autónomos, etc. Se presentan algunas investigaciones y experimentos que emplean la tecnología de la Jetson TX1.

A TEST DRIVE OF THE NVIDIA JETSON TX1 DEVELOPER KIT FOR DEEP LEARNING AND COMPUTER APPLICATIONS

Una investigación realizada por BDTi, una firma de tecnología que se dedica al análisis de tecnología y aplicaciones en sistemas embebidos.

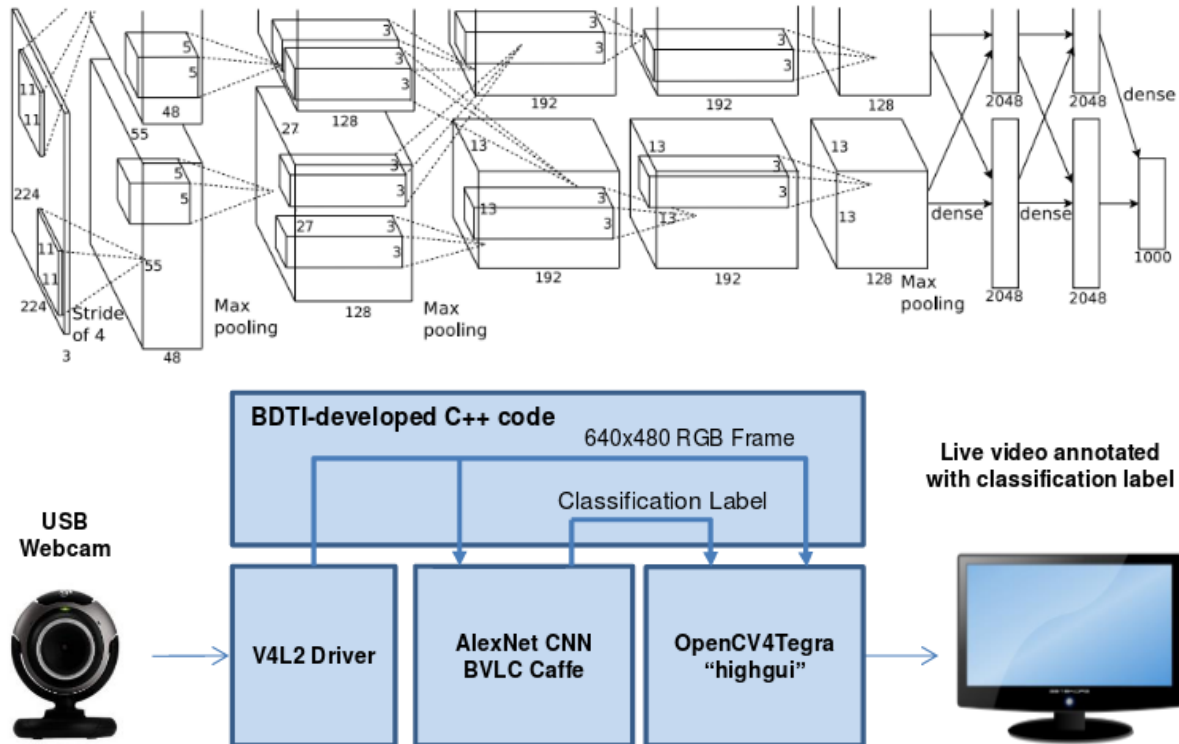
El experimento consistía en desarrollar e implementar en el módulo Jetson TX1 un algoritmo basado en deep learning, utilizando los paquetes de OpenCV y Caffe software comparado con un algoritmo clásico de visión por computador.

En el análisis se sigue el proceso de desarrollo completo de ambos algoritmos (deep learning y computer vision), comenzando por la configuración del módulo y luego el desarrollo y depuración de la aplicación.

Para el algoritmo de deep learning se usaron redes neuronales convolucionales para las tareas de clasificación de imágenes. A su vez, éstas redes neuronales tienden a tener un gran costo en computación y pueden ser paralelizables.

Utilizando la tecnología CUDA de NVIDIA es posible acelerar procesos a través del paralelismo. En el experimento se programó la GPU para acelerar

los procesos de entreno y reconocimiento de las redes neuronales convolucionales.



En la figura se aprecia el modelo desarrollado para el algoritmo de deep learning. En el que se usa una cámara para el reconocimiento de distintas clases de objetos.

En el algoritmo clásico de visión por computador se utiliza el módulo Jetson TX1 para implementar una típica tarea de diferencias.

Se usó herramientas integradas en el kit de desarrollo del Jetson TX1 de software libre para modificar el tamaño de los frames y alimentar la red neuronal que ha sido entrenada para reconocer distintos tipos de objetos, como es el caso de OpenCV4Tegra; una herramienta de NVIDIA que optimiza OpenCV especialmente para la arquitectura Tegra.

Del mismo modo, OpenCV4Tegra incorpora además de sofisticados algoritmos para la visión por computador optimizaciones en OpenGL y CUDA; para ser usadas por defecto en la Jetson TX1.

El experimento concluye que el rendimiento del algoritmo de deep learning fue impresionante ya que fácilmente alcanzaba los 30 fps sin haber realizado ningún tipo de optimización.

Además, el desarrollo de aplicaciones en el Developer Kit se parece más al desarrollo en pc que al típico sistema embebido. Los soportes para GPU que trae la Jetson TX1 permiten los desarrolladores se ahorren tiempo programando la GPU. [12]

Nvidia jetson pro

El kit modular avanzada NVIDIA Jetson Pro permite a los fabricantes de automóviles desarrollar y crear fácilmente gráficos de prueba intensivos y aplicaciones de visión por ordenador que van desde información y entretenimiento a bordo de vehículos, grupos de instrumentos digitales, y el controlador avanzado sistemas de asistencia (ADAS) hasta aplicaciones de vídeo y de procesamiento de imágenes, tales como la prevención de colisiones y detección de peatones.



Plataforma de desarrollo

Placa base Jetson Pro

VCM Tegra con procesador móvil Tegra para automoción

Placa adaptadora integrada (EBB) con numerosas opciones de conectividad

GPU externa compatible con NVIDIA CUDA®

Wi-Fi, módulo Bluetooth y antenas GPS

Unidad de disco mSATA de 64 GB

Pantalla táctil y cables

Fuente de alimentación y cables

Cable USB (mini-USB - USB)

Cable HDMI – DVI. [13]

FEATURES	
SoC	NVIDIA Tegra K1 VCM
Memory	2 or 4 GB
Storage	64 GB eMMC on VCM 8 GB eMMC on Baseboard
Networking	100 Mbit (ABB) GbE / AVB
Communication	Wi-Fi 802.11 a/b/g/n, AP; BT 4.0 (AW-AH691A-I5)
Modem	(add-on) NVIDIA i500 LTE Cat 3-4
Audio	AK4618VQ + AD1937 Audio Codecs
CAN	(ABB) Via MicroController [2 CAN channels, MCU]
Camera Aggregator	(ABB) 2x Maxim 9268 [8x GMSL over 2x CSI-2]
Touch Panel	Atmel Controller Sharp 10.1" EDP display
Other	AV Companion Chip

INPUT/OUTPUT AND FORM FACTOR	
USB	1 x 3.0 + 4x 2.0 type A + 1x 2.0 micro type B (ABB) 1x iAP2
Networking	100 Mbit (ABB) GbE / AVB
Storage	8 GB eMMC, mSATA, SD (VCM) 64 GB eMMC
PCIe	PCIe x 2
Camera	(ABB) GMSL x 8 OR 2x CSI-2 x4
Display Out	HDMI + mini DP [2x GMSL; up to 3 displays w/ ABB]
Display In	HDMI – In, 2x CVBS In
Audio	6x 2ch line out, 3x 2ch line in (ABB) 4x 2ch line out, 1x 2ch line in
Form Factor	178 mm x 178 mm x 66 mm

SOFTWARE AND SUPPORT	
OS	Vibrante Linux, Android, QNX
Graphics	OpenGL ES 3.1
Compute	NVIDIA CUDA 7.0
Multimedia	NvMedia
Vision	OpenCV
Authoring Toolchain	UI Composer™ Studio
Development Tools	NVIDIA Nsight™ and Visual Profiler
Support Model	NVIDIA Direct / NVonline
Genivi Compliant	Yes

Conclusiones

El aumento de la capacidad de cálculo a 1 Teraflop y un eficiente consumo energético, la tecnología Jetson desarrollada por NVIDIA ha permitido implementar nuevas técnicas (machine learning, computer vision) en los sistemas embebidos. Además de extender su uso a drones y sistemas autónomos.

Desarrollar aplicaciones para los módulos de NVIDIA Jetson se hace a través de una distribución de GNU/Linux lo que resulta más sencillo e intuitivo, diferenciándose de los típicos sistemas embebidos y acercándose a la programación de

escritorio. Los programadores cuentan con una gran cantidad de herramientas (Cuda, OpenCV, OpenGL, Vulkan, etc.) y algunas librerías especializadas para este sistema, todo en un módulo de bajo costo. Haciendo posible implementar soluciones y aplicaciones de una manera más sencilla para los programadores en general.

Los avances en tecnología que ha traído Jetson son un paso importante en la contribución de Nvidia en el desarrollo de aplicaciones que requieran sistemas embebidos con gran capacidad para procesar información en tiempo real,

Bibliografía

[1] Wikipedia. Sistema embebido. Disponible en https://es.wikipedia.org/wiki/Sistema_embebido

[2] Obregón, Héctor. (2007), "Embedded Software Estado Actual y Tendencias". SG Software Gurú Conocimiento en Práctica. Septiembre – Octubre. Año 3. Núm 5. Naucalpan, México, pp 20-21.

[3] NVidia. Procesamiento paralelo CUDA. Recuperado de <http://www.nvidia.es/object/cuda-parallel-computing-es.html>.

[4] Abi-Chahla, Fedy (June 18, 2008). "Nvidia's CUDA: The End of the CPU?". Recuperado de: <http://www.tomshardware.com/reviews/nvidia-cuda-gpu,1954.html>. Tom's Hardware. Retrieved May 17, 2015.

[5] Shimpi, Anand Lal; Wilson, Derek (November 8, 2006). "NVIDIA's GeForce 8800 (G80): GPUs Re-architected for DirectX 10". Recuperado de <http://www.anandtech.com/show/2116/8>. AnandTech. Retrieved May 16, 2015.

[6] Wikipedia. CUDA. Tomado de <https://es.wikipedia.org/wiki/CUDA>

[7] Wikipedia. CUDA. Tomado de <https://en.wikipedia.org/wiki/CUDA>

[8] Elinux. Jetson TK1. Tomado de http://elinux.org/Jetson_TK1

[9] Nvidia. NVIDIA Jetson TK1 Development Kit. Tomado de <http://www.nvidia.com/object/jetson-tk1-embedded-dev-kit.html>

[10] Nvidia. NVIDIA Jetson TK1 Development Kit. Tomado de http://developer.download.nvidia.com/embedded/jetson/TK1/docs/Jetson_platform_brief_May2014.pdf

[11] Elinux. Jetson TX1. Tomado de http://elinux.org/Jetson_TX1

[12] BDTi. A Test Drive of the NVIDIA Jetson TX1 Developer Kit for

Deep Learning and Computer Vision Applications. Tomado de http://www.bdti.com/MyBDTI/pubs/Nvidia_Jetson_TX1_Kit.pdf. Junio 2016

[13] Nvidia. Jetson pro. Tomado de <http://www.nvidia.com/object/jetson-automotive-development-platform.html>