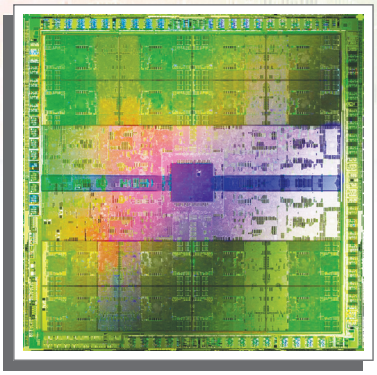


GPU Computing
از ســـیر تا پـــیاز

موتورهای پردازش موازی داخل چیپ‌های گرافیکی آنها را برای انجام طیف وسـیعی از پردازش‌های متنوع آماده کرده‌اند ولی افسوس! کجایند برنامه‌هایی که بتوانند از این همه قدرت بالقوه استفاده کنند؟

هومن سیاری
Sayyari@ComputerNews.ir

تا چند سـال پیش که صنعت گیم‌های کامپیوتری چندان رشـد نکرده بود CPU مجبور بود که بار سـنگین گیم‌های آن زمان را بر دوش بکشد. چیپ‌های گرافیکی فقط کارهای ساده و اولیه‌ای مثل عملیات ساخت تصویر ارسالی به مانیتور، سوییچ بین حالت متنی و گرافیکی و یا سـوییچ بیـن حالت نمایش ۱۶ رنگ یا ۲۵۶ رنگ را انجام می‌دادند. در آن زمان حتی سیسـتم‌عامل‌ها هم نیازی به گرافیک نداشتند چرا که متداول‌ترین سیسـتم‌عامل آن زمان یعنی DOS هم در یک محیط متنی و بدون نیاز به هیچ پردازش گرافیکی اجرا می‌شد.

بعـد از ظهور سیسـتم‌عامل ویندوز، چیپ‌هـای گرافیکی کمی رشـد کردند چرا که این سیسـتم‌عامل نیاز بـه پردازش گرافیکی بالاتری داشـت و خودش هم در یـک محیط گرافیکی نمایش داده می‌شـد. در این زمان بود که بحث اسـتفاده از Windows Accelerator برای رسـیدن به راندمان بالاتر گرافیکی مطرح شـد. سـپس کارت‌های گرافیکی متنوعی کـه از قابلیت فوق پشـتیبانی می‌کردند وارد بـازار شـدند از آن جمله می‌تـوان بـه 3dfx Voodoo و Rendition V1000 اشاره کرد. اما امروزه شرایط عوض شده است.

چیپ‌های گرافیکی می‌توانند پردازش‌های سـنگینی را به صورت موازی اجرا کنند و یا محاسـبات ممیز شـناور با دقت مضاعف را انجام دهنـد. پردازش گرافیکی به چیپ‌هـای ۳بعدی جدید امروزی اجازه می‌دهد کـه فقط به کارهای گرافیکی فکر نکنند بلکه آماده انجام هر کار دیگری هم باشند!

در این مقاله، منظور از **پردازش گرافیکی** انجام هر گونه پردازش اعم از گرافیکی و غیر گرافیکی توسط چیپ گرافیکی (GPU) و نه پردازنده اصلی (CPU) است. ریشـه پـردازش گرافیکی به یک تحقیق آکادمیک به نـام **GPGPU** که مخفف General Purpose computing on Graphics Processing Units اسـت برمی‌گـردد. در واقـع GPGPU بـه مفهـوم انجـام هـر نوع پـردازش (و نه فقط پردازش‌های گرافیکی) توسط چیپ‌های گرافیکی است.

اوایل، GPGPU با مشـکلات زیادی دست و پنجه نرم می‌کرد چرا که چیپ‌های گرافیکی برای انجام عملیات ممیز شناور با DirectX پایین‌تر از نسخه ۹، مشکل داشـتند اما در دوران DirectX 11 این فناوری جان تازه‌ای گرفته اسـت و آماده هر گونه هنرنمایی است!

نسل جدید پردازش گرافیکی خیلی گسترده‌تر شده است. ویژگی DirectCompute تعبیه‌شـده در DirectX 11 تقریبا تمامی کارت‌هـای مبتنی بر DirectX 11 را پشتیبانی می‌کند. OpenCL هم از پلتفرم چند سیستم‌عاملی پشتیبانی می‌نماید.

ما می‌خواهیم به هر یک از سـازندگان اصلی سخت‌افزار یا APIها نگاهی بیندازیم تا ببینیم آنها برای پشتیبانی از پردازش گرافیکی چه کرده‌اند؟

بررسی سخت‌افزاری‌ها

اگر بخواهیم به جستجوی سازندگان چیپ‌های گرافیکی که از GPGPU پشتیبانی می‌کننـد بپردازیم خیلی زود به جواب می‌رسـیم. پاسـخ کاملا واضح اسـت فقط ۲ شـرکت AMD و nVIDIA چنیـن چیپ‌هـای گرافیکـی تولیـد می‌کننـد. اجازه بدهید هر یک از آنها را به صورت مجزا مورد بررسی قرار دهیم:

nVIDIA : TESLA و CUDA

اولین تلاش برای دستیابی به GPGPU توسط nVIDIA انجام شد.

شـواهدی وجود دارد که نشـان می‌دهد این شـرکت با چیپ‌هـای گرافیکی اولیه سـری GeForce 256 کـه حتـی فاقـد قابلیـت سـایه‌زنی قابـل برنامه‌ریزی (Programmable shaders) بودند هم سـعی در پیاده‌سـازی GPGPU داشته اسـت. البته این تلاش‌ها زمانی نتیجه داد که DirectX 9 با قدرت انعطاف‌پذیری به مراتب بالاتر در معماری سایه‌زنی قابل برنامه‌ریزیش عرضه شد.

مدل‌هـای DirectX Shader جزئیـات بسـیاری را جهت کنترل بـر روی آنچه نمایـش داده می‌شـود بـرای افراد خواسـتار توسـعه، ارائه می‌دهنـد و می‌توانند افکت‌های بسـیاری نظیر سـایه‌های پیچیده انعکاس نور، ایجاد مه و مانند آنها را بوجود آورند. مایکروسـافت با ارائه Shader Model از ابتدا قدرت مانور زیادی را در اختیار توسـعه دهندگان قرار داده اسـت که نتیجه آن ایجاد پتانسیل بالا جهت واقعی‌تر و ملموس‌تر کردن تصاویر بوده است.

Shader Model II نسـخه‌ای اسـت کـه DirectX 9 در اختیـار کاربران قرار می‌دهد. بـرای به تصویر کشـیدن صحنه‌های واقعی و باور نکردنـی در گیم‌ها و خلق سـایه‌های درست و دقیق استفاده می‌شود. فعل و انفعالات پیچیده بین منابع نور، اشـیاء و نشانه‌های گرافیکی شـامل برنامه‌هایی می‌شود که جزئیات را با دقت بیشتری بررسـی می‌کنند و در واقع در یک بازی، تمامی منابع نوری مرتبط به هر شـئ باید مورد تجزیه و تحلیل قرار بگیرند. این بدین معناست که در سیستم‌هایی کـه از پردازنده‌هـای گرافیکی اسـتفاده می‌کننـد هر زمان که یک بـازی یا یک برنامه کاربردی به بازسـازی سـایه‌ها بپردازد، Ultra Shadow II عملکرد نهایی

کار را ارتقا می‌بخشد. با استفاده از این تکنولوژی هر چه نیاز بیشتری به نورپردازی و محاسبه سایه‌ها در یک صفحه داشته باشیم بهبود قابل توجهی در نوع عملکرد حاصل خواهد شد و در حقیقت هر چه قدر ترکیبات پیچیده‌تر باشد، نتایج قابل ملاحظه و برجسته‌تری به دست خواهد آمد. در مجموع این تکنولوژی نسبت به نسخه قبلی می‌تواند میزان کارایی سایه‌زنی را تا چهار برابر افزایش دهد.

یکی دیگر از مزایای این تکنولوژی این است که به نویسندگان نرم‌افزارهای گرافیکی این امکان را می‌دهد که سایه‌ها را با سرعت بسیار بالاتری از طریق حذف نواحی غیر ضروری بازسازی نمایند. در واقع با مشخص کردن پهنای باند یک صفحه و تمرکز بر روی نواحی که بیشتر تحت تاثیر منابع نوری هستند، نرم‌افزارنویسان می‌توانند فرآیند پردازش سایه‌ها را به طور عمده‌ای تسریع نمایند. با به کارگیری سایه‌هایی که امکان تنظیمات پارامتری را برای افزایش کارایی دارند، نویسندگان نرم‌افزارهای گرافیکی می‌توانند تصاویر گرافیکی باور نکردنی که به واقعیت بسیار نزدیک می‌باشند را خلق نمایند.

ان‌ویدیا از همان اول متوجه شده بود که GPUها پتانسیل انجام پردازش‌های سنگین را دارند. بنابراین مهندسان آن به این فکر افتادند که چگونه می‌توانند از GPU برای پردازش‌های غیر گرافیکی استفاده کنند. تا آن موقع کارت‌های گرافیک برای انجام امور گرافیکی مناسب بودند اما ساختن برنامه‌های غیر گرافیکی که بخواهند از قدرت کارت گرافیک برای پردازش مورد نیاز خود استفاده کنند مشکل بود. در برنامه‌نویسی کارت‌های گرافیکی امکان استفاده از برخی از قابلیت‌های رایج برنامه‌نویسی مثل حلقه‌ها و بازگشت‌ها میسر نبود.

مشکلات دیگر آن بود که DirectX 9 بخشی از معماری چیپ گرافیکی را قفل کرده بود این بدان معنا است که پیاده‌سازی ویژگی‌هایی که برای انجام کارهای غیر گرافیکی لازم بود خارج از استاندارد DirectX شناخته می‌شد و پشتیبانی نمی‌شد.

با وجود این‌ها به نظر می‌رسید که چیپ‌های گرافیکی کاندیدای مناسبی برای انجام پردازش‌هایی نظیر SIMD (اجرای یک دستورالعمل بر روی چند داده) و عملیات ممیز شناور محض باشند. ان‌ویدیا برای پیشرفت هر چه بیشتر GPGPU یک فریم‌ورک نرم‌افزاری جدید و کاربرپسند به نام CUDA 1.0 را طراحی کرد.

CUDA مخفف عبارت **C**ompute **U**nified **D**evice **A**rchitecture است. حالا برنامه‌نویسان قادر بودند که از زبان برنامه‌نویسی C در کنار امکاناتی که ان‌ویدیا برایشان بر اساس این زبان برنامه‌نویسی فراهم کرده بود استفاده کنند و برنامه‌های خود را بسازند. این روش خیلی راحت‌تر از روش برنامه‌نویسی بر اساس روش قبلی یعنی برنامه‌نویسی سایه‌ها است. به عبارت دیگر در این روش جدید برای نوشتن برنامه‌های عمومی نیازی به استفاده از کدهای گرافیکی نبود.

CUDA با چیپ‌های GTX Geforce 8800 و به بعد کار می‌کند. اما اولین نسل از چیپ‌های گرافیکی که اختصاصا بر اساس پردازش گرافیکی ساخته شدند Tesla 870 بودند.

ان‌ویدیا از زمان چیپ‌های ۸۸۰۰ سعی کرد که چیپ‌هایی بسازد که قابلیت GPGPU آنها کاربردی‌تر و بهتر باشد، هر چند به دنبال آن نبود که GPU را جایگزین CPU کند. CPUها هنوز در پردازش خطی یا چند رشته‌ای برتری محسوسی داشتند هر چند GPUها این پتانسیل را دارند که در پردازش‌های حجیم که به صورت موازی بر روی صدها رشته عملیاتی (Thread) که بر روی داده‌های جداگانه ولی هم نوع انجام می‌شوند بازدهی بسیار بالایی را از خود نشان دهند. این روش برنامه‌نویسی بسیار مناسب کاربردهایی مثل محاسبات علمی و یا تحلیل‌های مالی است.

شاید این موضوع خیلی معنی‌دار باشد که ان‌ویدیا آخرین معماری خود معروف به Fermi را به عنوان یک پلتفرم پردازش توسط گرافیک (GPU compute platform) معرفی کرده است و نه یک پردازنده گرافیکی (Graphics processor)!

شکل ۱: نمایی شماتیک از معماری Fermi

معماری Fermi، پیشرفت‌های سخت‌افزاری قابل توجهی را به ارمغان آورد که در نتیجه این چیپ را برای پردازش‌های عمومی و نه صرفا گرافیکی بسیار ایده‌آل کرده است. از جمله این پیشرفت‌ها می‌توان به Atomic Memory Operations، Unified Memory Architecture و Context Switching بهتر و موارد دیگر اشاره کرد. از زمانی که Fermi معرفی شد ان‌ویدیا چندین بار پلتفرم نرم‌افزاری CUDA را آپدیت کرده است.

ان‌ویدیا به پردازش گرافیکی فقط به دید ابزاری برای انجام محاسبات مثلا استخراج نفت و یا پردازش‌های آکادمیک نگاه نمی‌کرد بلکه به دنبال آن بود که این قابلیت را کاربردی‌تر و عمومی‌تر نماید. مثلا ان‌ویدیا قابلیت PhysX را چندین سال پیش عرضه کرده بود هر چند از یک سخت‌افزار اختصاصی برای پیاده‌سازی آن استفاده نمی‌کرد ولی تلاش داشت که با APIهای طراحی شده بتواند آن را توسط GPU پیاده‌سازی کند. ان‌ویدیا همواره با شرکت‌های سازنده گیم‌های کامپیوتری کار می‌کرد و آنها را به استفاده از قابلیت‌های پردازش گرافیکی در گیم‌های جدیدشان و در کاربردهایی مثل شبیه‌سازی آب، افکت‌های لنزهای نوری، انفجار و ... تشویق می‌کرد. این شرکت همچنین با شرکت‌های مهمی مثل ArcSoft، Adobe، CyberLink و ... همکاری داشته است تا آنها قابلیت تبدیل فرمت‌ها را با استفاده از پردازش گرافیکی در محصولتشان بگنجانند.

ان‌ویدیا توانست با Tesla که بر پایه معماری Fermi طراحی شده حدود ۱۰۰ میلیون دلار در سال ۲۰۱۱ فروش کند.

AMD با قدرت در حرکت

AMD کار بر روی پردازش گرافیکی را با تاخیر شروع کرد ولی با شتاب بالا ادامه می‌دهد. ATI Stream نامی بود که این شرکت در مقابل CUDA انتخاب کرده بود. اولین کارت گرافیکی که اختصاصا برای پردازش گرافیکی طراحی شد FireStream 580s بود. البته برای اولین بار در سری Radeon کارت Radeon X1900 هم از GPGPU پشتیبانی می‌کرد. البته واقعیت آن است که تا زمانیکه کارت‌های سری Radeon HD 4000 عرضه شدند عملا AMD حرف چندانی در زمینه

GPGPU نداشت. سری Radeon HD 5000 پیشرفت قابل توجهی در این زمینه کرد و سپس در سری Radeon HD 6000 قابلیت پردازش گرافیک به اوج خود رسید.

اگر کمی دقیق‌تر نگاه کنیم متوجه می‌شویم که در ابتدا AMD سعی داشت که فقط از ان‌ویدیا تقلید کند بدون آنکه نوآوری داشته باشد اما بعد از مدتی مسیر خود را عوض کرد و به آغوش استانداردهای باز مثل OpenCL و DirectCompute پیوست!

اخیرا AMD مسیر خود را به سمت کاربردهای حرفه‌ای سوق داده است. این شرکت کارتهای اختصاصی خود را که برای پردازش گرافیکی طراحی می‌کند را تحت عنوان FireStream عرضه می‌نماید. حالا AMD قصد دارد که توانایی کارت‌های FireStream را در چیپ‌های گرافیکی Radeon که در پردازنده‌های جدید این شرکت با نام Fusion گنجانده شده‌اند قرار دهد. Fusion نام پردازنده‌های جدید شرکت AMD است که یک SoC محسوب می‌شوند یعنی هم CPU و هم GPU در کنار هم در داخل یک چیپ قرار دارند.

پردازنده‌های Fusion هم در کامپیوترهای دسکتاپ و هم موبایل کاربرد دارند. AMD به این روش mainstream GPU compute App Acceleration می‌گوید.

تعداد برنامه‌هایی که از این قابلیت AMD استفاده می‌کنند چندان بالا نیست. این برنامه‌های کم می‌توانند بر روی GPU سریعتر اجرا شوند اما راندمان متوسط CPU منجر می‌شود که نتوان مقایسه درستی بین آن و پردازنده‌های معادل اینتل انجام داد. البته AMD معتقد است که در آینده تعداد برنامه‌هایی که از این قابلیت استفاده خواهند کرد به مراتب بیشتر خواهد شد.

فرق بین اینتل و AMD در زمینه پردازش گرافیکی آن است که AMD این قابلیت را به پردازنده‌های جدید خود که همان Fusion باشند هم انتقال داده است چرا که یک پردازنده Fusion از یک چیپ گرافیکی Radeon با پشتیبانی از قابلیت GPGPU و یک CPU ساخته شده است اما اینتل در پردازنده‌های جدید خود از جمله Ivy-Bridge از یک چیپ گرافیکی به نام HD 4000 بدون پشتیبانی از قابلیت GPGPU و یکCPU استفاده کرده است.

اینتل: پلی به پردازش گرافیکی

اینتل پیشرفت پردازش گرافیکی را با دلواپسی مشاهده می‌کرد. اینتل تلاش کرد که کارت گرافیکی خودش را بسازد اما این پروژه با شکست مواجه شد. ایده‌ای که اینتل برای این کارت گرافیکی شکست خورده داشت امروزه به صورت محدود در برخی از کاربردهای پردازش گرافیکی با راندمان بالا دیده می‌شود.

از طرف دیگر اینتل مشکلاتی با گرافیک گنجانده شده در پردازنده‌های جدیدش دارد. این گرافیک‌ها در رده متوسط محسوب می‌شوند هر چند کارهایی مثل کدینگ فایل‌های ویدیویی را به خوبی انجام می‌دهند. البته کدینگ ویدیو یک کار استاندارد محسوب می‌شود و چنانچه یک کار غیر استاندارد به این چیپ گرافیکی واگذار شود عملکرد ضعیفی خواهد داشت. با این وجود با توجه به اینکه بسیاری از کاربران از پردازش گرافیکی فقط به دنبال کدینگ ویدیویی آن هستند لذا اینتل می‌تواند نیازهای آنها را پاسخ دهد.

پردازنده جدید اینتل یعنی Ivy-Bridge دارای راندمان بالایی در بخش CPU است و چیپ گرافیکی آن هم از DirectX 11 به طور کامل پشتیبانی می‌کند. این مجموعه در واقع سیگنالی است که نشان دهنده غروب کارت‌های گرافیکی رده متوسط است.

تلاش‌های اولیه برای استفاده از کارت گرافیک به عنوان یک پردازنده با مشکلات زیادی روبرو شد چرا که برنامه‌نویسان باید برنامه‌هایی را بر اساس کارت گرافیک می‌نوشتند که اصولا مربوط به کارت گرافیک نبود! یعنی آنها می‌خواستند با برنامه‌های خود کاری را انجام دهند که در حوزه وظایف آن کارت تعریف نشده بود. مثلا باید برنامه‌هایی می‌نوشتند که بتواند با استفاده از کارت گرافیک سایه‌زنی را شبیه‌سازی کند که ساختار این برنامه هیچ ارتباطی به گرافیک نداشت.

در اینجا بود که یک نیاز اساسی به شدت احساس می‌شد و آن استانداردسازی رابط برنامه‌نویسی برای کارت‌های گرافیکی بود. البته این اتفاق در تاریخ کارت‌های گرافیکی تکرار شده بود یعنی ابتدا یک تکنولوژی معرفی می‌شد و سپس استانداردهای آن ارایه می‌شدند.

CUDA

CUDA از شرکت ان‌ویدیا یکی از اولین تلاش‌ها برای استانداردسازی برنامه‌نویسی برای پردازش گرافیکی بود. ان‌ویدیا همیشه تلاش می‌کرد که این موضوع را جا بیندازد که CUDA یک استاندارد مختص این شرکت نیست بلکه همه شرکت‌ها می‌توانند از آن استفاده کنند ولی نه AMD و نه اینتل حاضر به استفاده از آن نشدند!

CUDA با یک کامپایلر به زبان C و یک کتابخانه از توابع مربوطه کار برنامه‌نویسی برای پردازش موازی توسط کارت گرافیک را آغاز کرد. بعد از چند سال چندین کامپایلر و دیباگر از شرکت‌های دیگر در این زمینه معرفی شد و حتی برنامه‌نویسی برای CUDA در مجموعه برنامه‌نویسی قدرتمند Microsoft Visual Studio گنجانده شد.

CUDA بیشترین موفقیتش را در بازار کامپیوترهای خانگی و سوپر کامپیوترهای آکادمیک داشته است اما CUDA اهداف بزرگتری را دنبال می‌کند. شرکت معروف Adobe از CUDA در برنامه Adobe Premiere Pro استفاده می‌کند و از آن برای تبدیل فرمت‌های HD و یا برای برخی از Transitionها استفاده می‌نماید. نرم‌افزار MotionDSP از CUDA برای کاهش اثر Shaky-cam در ویدیوهای خانگی استفاده می‌کند. بسیاری دیگر از برنامه‌های سطح بالا و حرفه‌ای هم از CUDA استفاده می‌کنند تا زمان پردازش‌های خود را به میزان چشمگیری کاهش دهند. این همان چیزی است که ان‌ویدیا به شدت به دنبال آن است تا سازندگان معتبر نرم‌افزار بر اساس CUDA نرم‌افزارهای خود را طراحی کنند تا کاربران این نرم‌افزارها مجبور به خرید کارت‌های ان‌ویدیا شوند.

ATI Stream

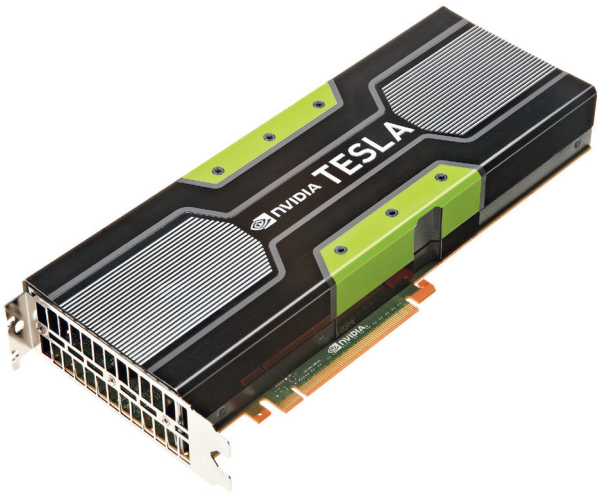
همانگونه که قبلا اشاره شد سابقه AMD در پردازش گرافیکی به مراتب کمتر از ان‌ویدیا است و این شرکت بیشتر بر روی OpenCL و DirectCompute متمرکز شده است. در واقع Stream نام روشی بود که AMD برای رقابت با CUDA انتخاب کرد اما بعد از مدتی AMD به این نتیجه رسید که استفاده از پلتفرم‌های استاندارد بسیار جذاب‌تر از روش اختصاصی خودش است!

DirectCompute

DirectCompute با فریم‌ورک DirectX 11 عرضه شد و بنابراین فقط در ویندوز ویستا و ۷ و ۸ قابل استفاده است. بنابراین بر روی ویندوزهای قدیمی‌تر مثل ویندوز XP و سایر سیستم‌عامل‌های غیر مایکروسافتی قابل اجرا نیست. همچنین بر روی ویندوز فون۷ و ۸ و Xbox 360 نیز کار نمی‌کند.

DirectCompute بر روی تمام GPUهایی که می‌توانند از DirectX 11 پشتیبانی کنند اجرا می‌شود. این GPUها شامل سری NVidia GTX 400 و مدل‌های جدیدتر از آن و نیز سری AMD Radeon HD 5000 و مدل‌های جدیدتر از آن می‌گردند. اینتل هم تنها در پردازنده‌های Ivy Bridge از DirectX 11 پشتیبانی می‌کند.

مزیت کلیدی DirectCompute آن است که تنها از یک زبان سایه‌زنی پیشرفته به نام HLSL برای برنامه‌نویسی پردازش گرافیکی استفاده می‌کند. این موضوع باعث می‌شود که کار برنامه‌نویسان به مراتب ساده‌تر شود. مزیت دیگر هم این



شکل۲: کارت گرافیک Tesla

شکل ۳: مقایسه APIهای رایج

CUDA	OpenCL	DirectCompute
Nvidia proprietary	Developed by a consortium led by Khronos Group	Developed by Microsoft
Oldest GPU compute platform software	Strong implementation on Mac OS X	Part of DirectX 11
Lots of support in high-performance computing	Available on most OS platforms	Graphics-like API
Available on multiple OS platforms	Runs on Nvidia, AMD, Intel (CPU). Intel GPU support in Ivy Bridge	Strong performance on Windows Vista, 7, 8
Mostly Nvidia H/W with CPU fallback		Runs on GPUs; no CPU fallback

است که این کدهای نوشته شده می‌توانند به راحتی بر روی طیف وسیعی از کارت‌های گرافیکی اعم از AMD و NVidia اجرا شوند بدون اینکه نیاز به تغییری در کد نوشته شده وجود داشته باشد.

البته باید به این نکته هم توجه کرد که کدی که بر اساس DirectCompute نوشته شده است اگر بر روی سیستمی اجرا شود که گرافیک آن از DirectX 11 پشتیبانی نمی‌کند دچار خطا شده و اجرا نمی‌گردد. بنابراین برنامه‌نویسان باید در برنامه‌های خود این نکته را هم مد نظر داشته باشند و کد دیگری هم بنویسند تا در شرایط فوق جایگزین کد مبتنی بر DirectCompute شود و البته توسط CPU اجرا گردد!

OpenCL

در ابتدا OpenCL توسط اپل توسعه داده شد و سپس اپل آن را در قالب یک استاندارد باز به یک کمیته به نام Khronos Group واگذار کرد. اپل نام آن را برای خود ثبت کرده است ولی استفاده از آن را آزاد گذاشته است.

OpenCL تقریبا بر روی همه پلتفرم‌های موجود از CPUهای کامپیوترهای قدیمی تا GPUهای تجهیزات موبایل مثل گوشی‌های هوشمند و تبلت‌ها اجرا می‌شود. فقط توجه داشته باشید که کدی که بر اساس OpenCL نوشته می‌شود ممکن است تفاوت اجرای قابل توجهی داشته باشد زمانیکه بر روی یک GPU گوشی موبایل اجرا می‌شود نسبت به زمانیکه همان کد بر روی یک گرافیک قدرتمند مثل NVidia GTX 680 اجرا می‌گردد!

پشتیبانی از OpenCL به طور کامل انجام شده است. AMD آنقدر شیفته OpenCL بود که حاضر شد از ATI Stream SDK خود دست بکشد و به یک SDK جدید بر مبنای پردازش موازی که اختصاصا بر اساس OpenCL طراحی شده است روی بیاورد. OpenCL حتی در وب هم خودنمایی می‌کند. یک نسخه از OpenCL به نام WebCL وجود دارد که توسط مرورگرها مورد استفاده قرار می‌گیرد. این نسخه به جاوا اسکریپت‌ها اجازه می‌دهد که کدهای OpenCL را فراخوانی کنند. این بدان معناست که شما می‌توانید کدهای پردازش گرافیکی را در داخل مرورگر خود اجرا کنید.

از طرف دیگر OpenCL هنوز در دوران طفولیت خود به سر می‌برد. ابزارها و میان‌افزارهای آن تازه معرفی شده‌اند و هنوز توسعه‌دهندگان نیاز به زمان بیشتری دارند تا کتابخانه‌ها و توابع مورد نیاز را عرضه کنند. به طور مثال در حال حاضر توسعه دهندگان بزرگ مثل مایکروسافت از OpenCL در Microsoft Visual Studio پشتیبانی نمی‌کنند.

جنگ APIها

امروزه وضعیت APIهای پردازش گرافیکی مشابه شرایط APIهای گرافیک ۳بعدی اواخر دهه ۱۹۹۰ میلادی است. پلتفرم پیشرو همان CUDA است. با وجود ادعای NVIDIA هنوز هم CUDA به صورت اختصاصی توسط خودش استفاده می‌شود. CUDA دارای سهم بزرگی از توسعه‌دهندگان و برنامه‌های کاربردی است اما واقعیت این است که هیچگاه تاریخ با استانداردهای اختصاصی مهربان نبوده است! شاید همین باعث می‌شود که NVidia تلاش می‌کند که سایر رقبا را هم به استفاده از CUDA تشویق کند هر چند تا کنون که موفق نبوده است!

البته می‌توان ادعا کرد که DirectCompute هم اختصاصی است چرا که فقط بر روی سیستم‌عامل ویندوز اجرا می‌شود و حتی ویندوزهای ماقبل از ویستا را هم در بر نمی‌گیرد هر چند ویندوز حاکم مطلق کامپیوترهای دنیاست و از طرف دیگر تمامی کارت‌های گرافیکی که از DirectX 11 پشتیبانی می‌کنند هم می‌توانند از DirectCompute بهره ببرند.

OpenCL با پشتیبانی از چندین سیستم‌عامل و پشتیبانی از طیف وسیعی از سخت‌افزارها آینده درخشان‌تری را ترسیم می‌کند. OpenCL توسط سیستم‌عامل MAC OS X به صورت خودکار پشتیبانی می‌شود و استفاده از آن مخصوصا در MAC Bookها راندمان بالاتری را نسبت به نوت‌بوک‌های ویندوزی به ارمغان می‌آورد. اما واقعیت آن است که OpenCL هنوز اول راه است و هیچ توسعه‌دهنده بزرگی به صورت رسمی برنامه‌نویسی برای آن را در محصولات خود نگنجانده است.

پردازش موازی در آینده

CPUها هرگز از گردونه خارج نخواهند شد چرا که همیشه نیاز به پردازش خطی وجود خواهد داشت و برخی از کاربردها اصولا قابل اجرا به صورت موازی نیستند. آینده اینترنت و کامپیوترها به سوی ویژوال شدن پیش می‌رود. ویدیوهای دیجیتال، عکسبرداری و بازی‌ها گام‌های اولیه به سوی ویژوال شدن هستند و اینترنت ویژوال از طریق استانداردهایی مثل WebCL و HTML5 Canvas تجربه ۳بعدی را بر روی وب عملی خواهد کرد. برنامه‌نویسان بیشتری از کدهای پردازش موازی استفاده خواهند کرد. چیپ‌های گرافیکی چه از نوع مجزا (Discrete) و چه از نوع داخلی (Intgerated) هر دو از پردازش موازی با قدرت بیشتری پشتیبانی خواهند کرد.

یادتان باشد که پردازش گرافیکی هنوز یک طفل نوپاست! ■