Whitepaper

The Benefits of Quad Core CPUs in Mobile Devices

Table of Contents

Brief History of Multi-core CPUs

Multi-core processing has been at the heart of the computing revolution for more than a decade. The transition began with server CPU manufacturers adopting multi-core processor architectures to address growing performance demands, combined with the dramatic increase in power consumption of single core processors running at high frequencies. Multi-core CPUs can operate at lower frequency, often consuming less power, and completing work much faster by running tasks in parallel compared to single core predecessors.

Over the years, multi-core technology transitioned to desktop CPU's and gaming consoles. Some consoles included as many as eight CPU cores to enable immersive, complex, and realistic gaming environments. Notebooks and laptop PCs then began to integrate multi-core CPUs to further enable high performance computing environments for users on the go.

And just recently, at the beginning of 2011, multi-core CPUs - beginning with Tegra 2 - started shipping in volume in tablets and smartphones, enabling support for higher quality mobile applications as well as longer battery life. Examples include HD video playback, 3D gaming, multi-tasking, and 3D interfaces which were already stretching the capabilities of existing single core mobile processors.

When Tegra 2 and dual-core CPUs were initially being discussed – well before device availability – they were assumed to be power hungry, not useful for mobile devices, and largely dismissed. However, after Tegra 2 was launched, these concerns were proven untrue, and every other industry leader followed suit with their own dual-core CPU product lines.

In order to further increase performance, extend battery life and stay within mobile power budgets, mobile processors will continue to increase the number of CPU cores.

Many benefits of symmetrical multiprocessing in mobile devices are described in a previously published whitepaper[1], which largely focused on the advantages of dual-core Tegra 2 compared to single core mobile processors.

NVIDIA's Project Kal-El processor implements a novel new **Variable Symmetric Multiprocessing (vSMP)** technology. Not previously disclosed publicly, vSMP includes a fifth CPU core (the "Companion" core) built using a special low power silicon process that executes tasks at low frequency for active standby mode, music playback, and even video playback. The four main "quad" cores are built in a standard silicon process to reach higher frequencies, while consuming lower power than dual core solutions for many tasks. All five CPU cores are identical ARM Cortex A9 CPUs, and are individually enabled and disabled (via aggressive power gating) based on the work load.

[1] See Whitepaper title 'Benefit of Multiple CPU cores in Mobile Devices'

NVIDIA's Project Kal-El vSMP architecture extends the benefits of dual-core processors by providing:

- Lower power consumption, Higher performance per watt
- Faster Web page load times
- Higher Performance for Demanding Applications
- Faster Multitasking
- Higher Quality Gaming

Lower Power Consumption, Higher Performance per Watt

As described in the whitepaper titled "***Variable SMP – A Multi-Core CPU Architecture for Low Power and High Performance***", one of the primary benefits of quad core CPUs is delivering ***lower*** power consumption than dual core and single core CPUs.

A common misconception is that a multi-core CPU consumes more power than a single core CPU and causes significant reduction in battery life. On the contrary, due to variable symmetric multiprocessing, the main quad core CPU architecture of Project Kal-El is more power efficient and delivers higher performance per watt than competing single and dual core processors.

In order to meet peak performance demands in a multitasking environment, a single core CPU not only runs at higher clock frequencies and voltages than a multi-core CPU, but also takes longer periods of time to complete a given task. Multi-core CPUs are able to use symmetrical multiprocessing and distribute workload across multiple CPU cores. Due to workload sharing, each CPU core can run at lower frequency and voltage to complete a multi-threaded task, or multiple tasks in a multi-tasking scenario. Also, due to lower operating frequency and voltage, each core consumes significantly lower power and offers much higher performance per watt compared to single core CPUs.

Additional detail on how multi-core CPUs are able to achieve lower power consumption than single core CPUs is given in the white paper titled "***Benefits of Multiple CPU Cores in Mobile Devices***".

Faster Web Page Load Times

Mobile devices that have a multi-core CPU with SMP support will be able to deliver a fast desktop PC-style Web browsing experience. Modern browsers such as Google Chrome and Mozilla Firefox are now multi-threaded and capable of spawning several concurrent processes. Each page tab in a Chrome browser is a separate process, and each process manages its own set of threads. Both the processes and threads are highly parallelizable. Figure 1 below shows the utilization of each core of a quad core CPU system while browsing the popular New York Times Website. From the figure it is seen that the browser uses all four cores of the quad core CPU, and this parallel processing results in a Web browsing experience that is much faster than on a dual core CPU-based mobile device. Quad Core CPUs also deliver higher performance for browsers that support tabbed browsing. Figure 2 below shows the CPU utilization across the four CPU cores when multiple tabs are opened in a Web browser. In fact, tabbed browsing uses significant CPU processing and as seen in the figure, may even saturate all four cores of a quad core CPU.

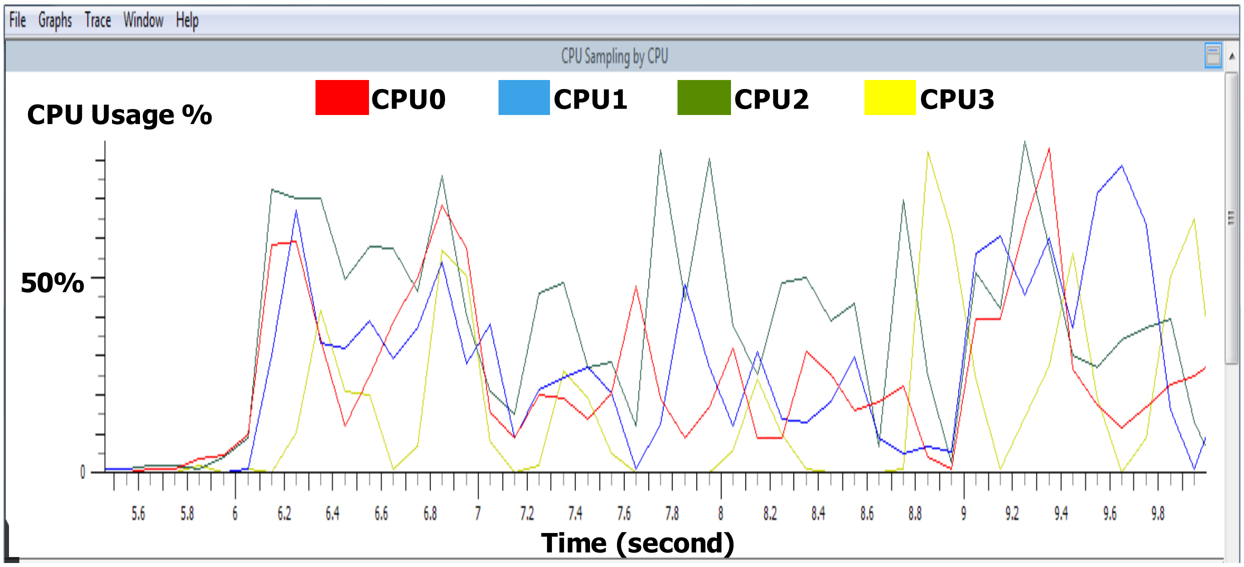

Figure 1 CPU utilization on Quad core CPU system during Web browsing[2]

Figure 2 Quad Core CPU utilization during tabbed Web browsing

[2] CPU utilization reported for Chrome Browser

.

On a quad core CPU based system, the operating system will be able to allocate multiple Web scripts across the four CPU cores and deliver much faster execution of JavaScript heavy pages. Results from Moonbat, a Web based JavaScript benchmark, show that a quad core CPU delivers almost fifty percent faster Web browsing performance compared to dual core CPU based mobile processors.

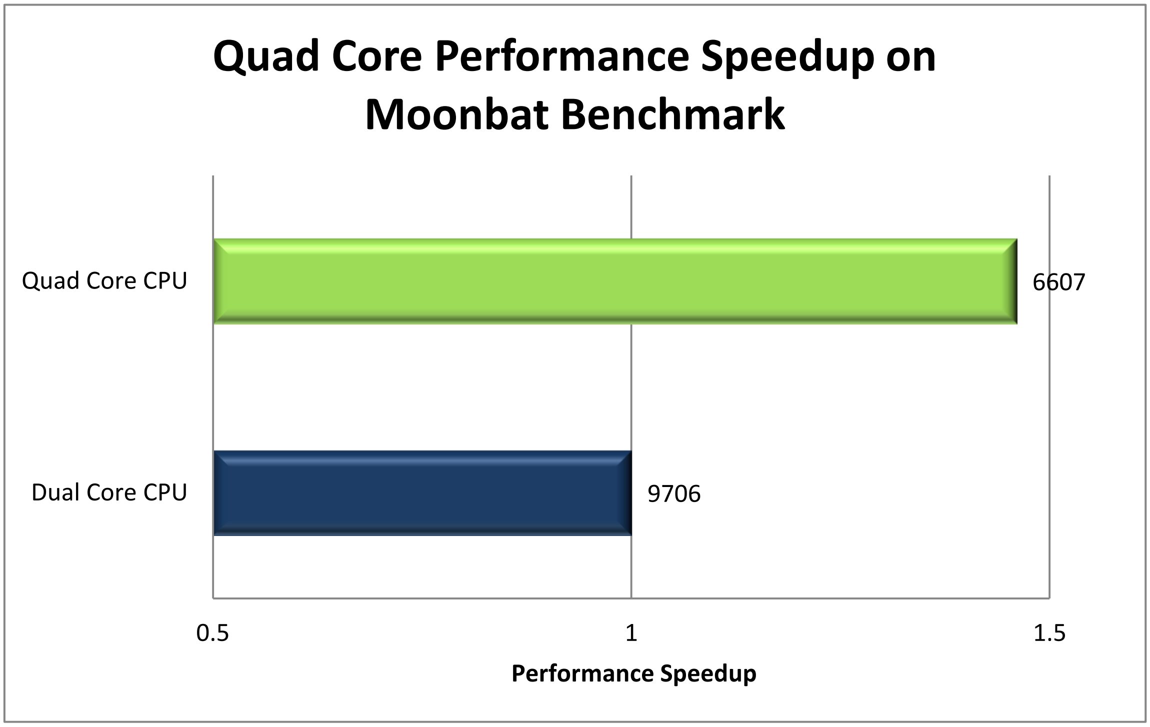

Figure 3 Quad Core Performance Advantage on Moonbat JavaScript benchmark

Higher Performance for Demanding Applications

A key benefit delivered by multi-core processors is the additional performance that they provide for demanding applications and use cases. The NVIDIA Tegra 2 processor with its dual core CPU delivered faster performance for applications such as photo editing, video transcoding, Web browsing, and multi-threaded gaming.

Quad core CPU-based mobile processors deliver even higher levels of performance for these types of applications, and enable more compelling applications never before seen on mobile devices. Examples include:

- High quality video editing
- Image processing

- Audio/video transcoding
- Physics simulations
- Numerous productivity apps
- Many forms of location-aware computing
- Facial recognition
- 3D stereo games and applications
- Virus scans
- File compression

Results from Coremark, a popular mobile CPU benchmark, are a strong indicator of performance for CPU intensive multimedia applications. For instance, Coremark shows that quad core CPUs delivers almost 2x the performance of dual core CPU based mobile processors and almost 4x the performance of single core CPUs.

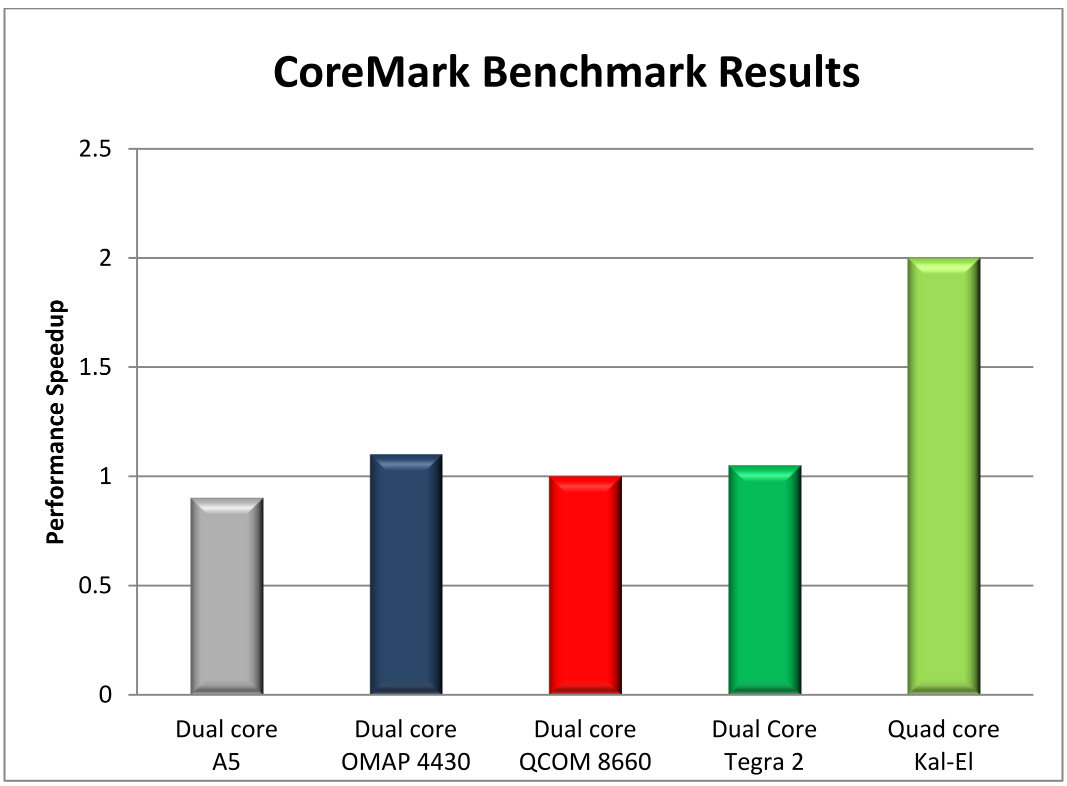

Figure 4 Coremark Multicore CPU Benchmark Results[3]

[3] Dual core OMAP4430 running at 1GHz, dual core QCOM 8660 running at 1.2 Ghz and quad core Kal-El running at 1 Ghz. See Appendix for coremark compile settings used to obtain scores

Media applications such as Image processing, audio/video transcoding, and file compression inherently have high degrees of parallelism, and can take advantage of symmetrical multiprocessing and multi-core CPUs. Image processing on a mobile device has become a very common use case due to the inclusion of one or more cameras on most mobile devices.

For example, Photaf 3D Panorama is a highly rated Android application that enables users to automatically capture 3D panoramic photos and stitch together the captured images to be viewed immediately. The heavy image processing involved in detecting edges and stitching together the images benefit greatly from the quad core processing capabilities of Project Kal-El. Performance measurements show it is able to process and display captured panoramic images almost two times faster than on dual core CPU based mobile devices.

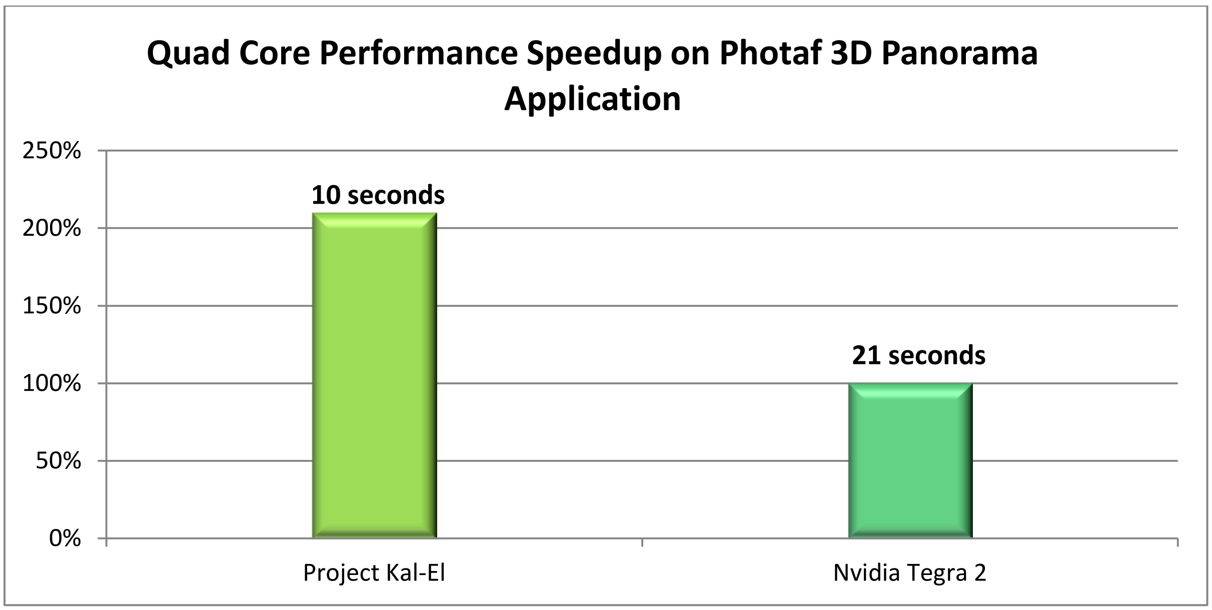

Figure 5 Performance speedup on Photaf 3D Panorama photo processing application

Linpack is a widely used CPU benchmark that gives a good measure of the performance that a processor can deliver running CPU intensive tasks such as media processing. Results from the multithreaded Linpack benchmark show that quad core Project Kal-El delivers almost sixty percent higher performance over an equivalent dual core processor. Actual applications that are tuned for quad core processors will show even higher performance gains on a quad core processor.

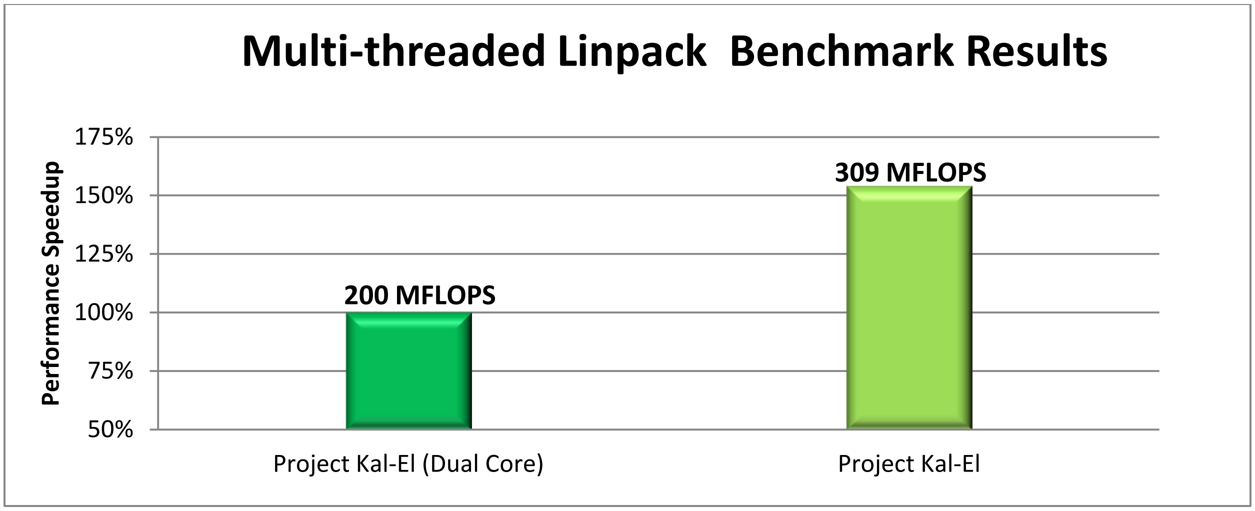

Figure 6 Performance Scaling on Multithreaded Linpack CPU benchmark[4]

Media transcoding is another use case that benefits from multiprocessing. Mobile users commonly capture and edit audio and video files on their phones before sharing them with their friends and social networks.

Handbrake, a popular video transcoding application, delivers significantly faster transcoding on a quad core CPU system compared to a dual core CPU based system. The figure below shows that a quad core CPU delivers almost a sixty percent performance speedup for Handbrake video transcoding tasks.

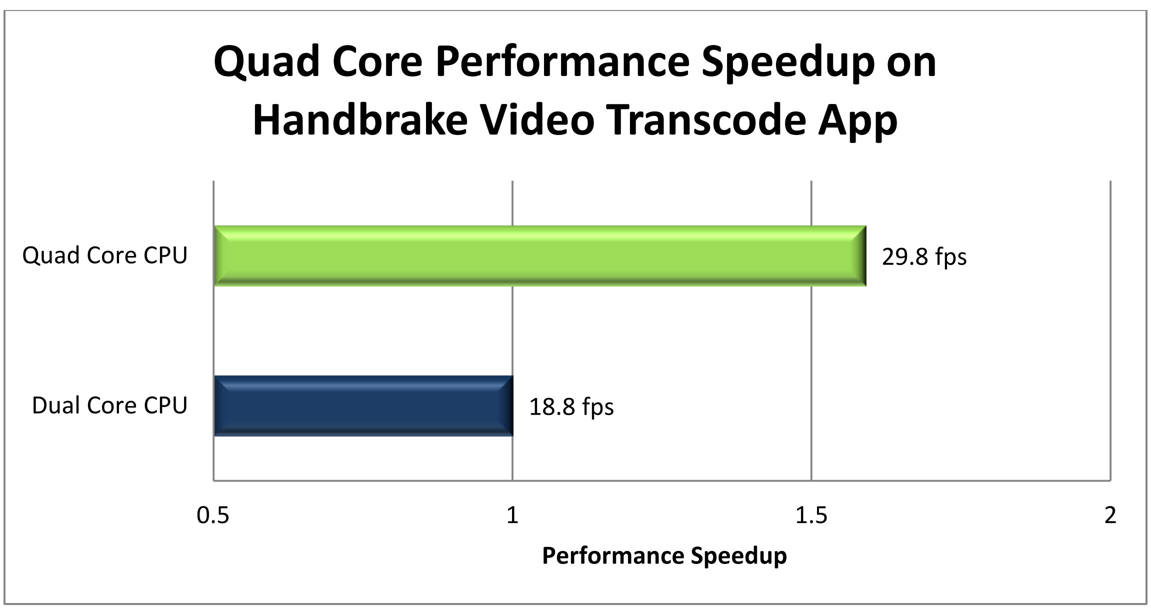

Figure 7 Quad core CPU Performance benefit on Handbrake Video Transcoding[4]

[4] Results obtained on NVIDIA Project Kal-El quad core platform. Dual core CPU scores were obtained by disabling two CPU cores on the Kal-El platform

As users begin to leverage the performance of quad core CPU-based mobile devices and use these devices as their primary computing device, developers will create more applications that leverage the power of quad core processing. The examples discussed above are just a few use cases that benefit. One can expect to see high-end applications for photo editing, video transcoding, multi-point video conferencing, multi-threaded java apps, video editing, and other scenarios become available for quad core mobile devices.

Best Multitasking Performance

Another important and highly visible benefit of quad core processing is superior multitasking that delivers high performance and uncompromised responsiveness. A variety of factors contribute to poor responsiveness. For example, a fully utilized CPU, unavailability of memory bandwidth, non-optimized driver stacks, etc., all could cause poor performance and unresponsiveness. Mobile device users typically have several applications running concurrently. For example, it is not uncommon to see applications such as Web browsers, streaming music, email syncs, social network syncs, and news feeders running concurrently on a mobile device. Under such heavy multitasking conditions, a single core CPU not only runs out of processing power to service multiple tasks, it also has to run at peak frequency to cope with the heavy workload. This results in poor performance and high power consumption.

On a quad core CPU based mobile device, the operating system can dynamically allocate workloads to the appropriate CPU core based on current CPU loads and task/activity priority. For example, if two CPU cores are busy processing long tasks such as Android application updates, file compression, and media processing the OS can immediately allocate latency sensitive tasks like touch inputs to the available third or fourth core to deliver faster responsiveness. When there are multiple tasks running at the same time, queued up request from these tasks can be serviced faster when there are multiple cores available to service these requests.

Higher Quality Gaming Experience

Gaming consoles and PCs today all use multi-core chips that support SMP technology, and most console and PC games are coded to take advantage of the multi-core and SMP capabilities of the hardware. Today, most game engines such as Unreal 3.0, Id Tech 5 and Frostbite are multithreaded, and the engines are increasingly migrating to task-processing models where the “size” of each individual job is reduced while the number of threads is increased. These threads are used for tasks such as audio, collision detection, artificial intelligence, user input processing, game strategy and network communications. All modern game platforms support many threads, and this trend will continue to increase on future platforms.

Due to the higher performance delivered by quad core processors, advanced gaming features such as real-time physics and real-time texture generation can now be used in mobile games to deliver significantly higher graphics quality and more realistic gameplay experience.

Real-time Physics Delivering Additional Realism

When elements from the physical world such as collisions, wind, water, gravity, motion, and more are included in the virtual world of a game, the natural expectation is that these behave exactly the same way as they would in the physical world. For example, collisions between two objects should result in realistic outcomes based on the mass, speed, and momentum of the colliding objects, cloth should interact with wind resulting in realistic turbulent movement along its surface, and objects should react to gravity and respond accordingly. These effects that are based on player actions within the game and real world laws add an additional level of realism to games making them more immersive and engaging. Most high end PC and console games employ these physics based effects to deliver a truly immersive and visceral gaming experience.

Implementing these special effects in a game is very performance-intensive because all the physics calculations required to implement the effects need to be done in real-time based on the real-time player actions happening in the game. The introduction of quad core mobile processors will now enable developers to use physics to create mobile games that are not only graphically rich but also very immersive. As demonstrated in the Glowball game demo[5], the collision of the ball with the barrels, the scattering of the barrels on the floor, the interaction between the ball and the cloth curtains are realistic and result in non-repetitive outcomes that are based on the speed of the ball, point of collision, and mass of the barrels.

Figure 8 Glowball Demo with Real-time Physics and Dynamic Lighting. Left side enabled, Right side disabled

[5] http://www.youtube.com/watch?v=eBvaDtshLY8

On Kal-El, the processing tasks for dynamic lighting, physics, artificial intelligence and other game related CPU processing is shared across the four cores. Due to this load sharing, none of the cores are overloaded and the processor still has additional processing headroom to handle background tasks without compromising the user experience.

The graph below shows the performance speedup that the quad core CPU based Project Kal-El platform provides for the Glowball demo over an equivalent dual core CPU based platform. The significant performance boost delivered by quad core CPU based mobile processors will bring us closer to true console and PC class games for mobile devices.

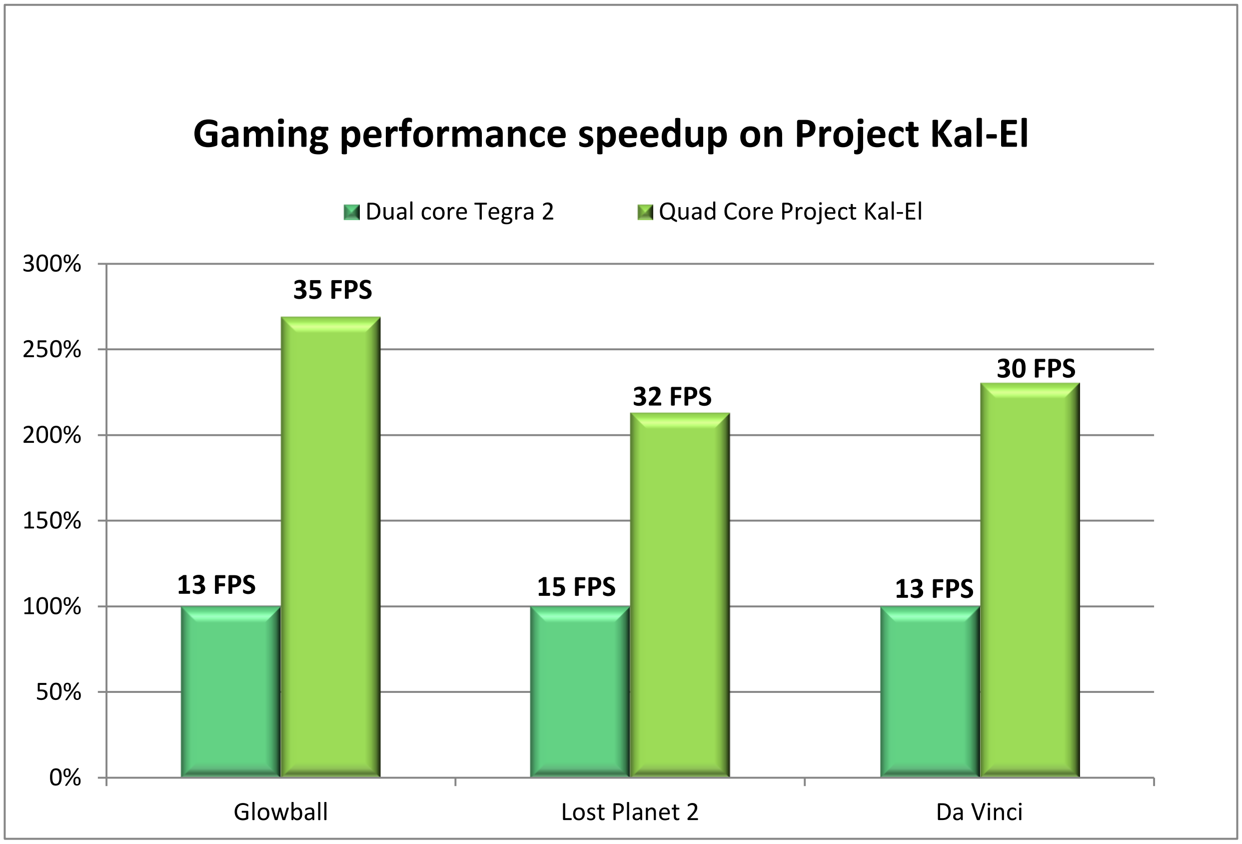

Figure 9 Performance benefit from Quad Core mobile processor for advanced games

Real-time Dynamic Texture Generation

As mobile games increase in complexity and visual richness, the corresponding file sizes of downloadable game content have also grown tremendously. Often, the game file size is so big that the time taken to download the game exceeds the fifteen minute refund time provided after a purchase for users to claim a refund if they don't like the game. This is not only creating buyer frustration, but also is demotivating buyers from purchasing advanced games that require large file downloads.

Real-time dynamic texture generation techniques allows game developers to code games such that textures required in the game are created in real-time based on the narrative and state of the game. Since the textures are generated on the fly, they do not have to be provided at the time of purchase. Using this technique, game developers can reduce game file sizes by several orders of magnitude.

Figure 10 Scene variations in games, all dynamically generated on Kal-El's 4 CPU cores

Another key benefit of dynamic texture generation is that the game can be coded such that the environment in the game changes based on player input and customization. For example, the player can be given the option to change the weather in the game and based on the chosen weather condition the associated textures that affect the scene can be dynamically generated. Thus this technique helps deliver a very interactive gaming experience without significantly increasing file size or coding complexity in the game.

Real-time dynamic texture generation is however a very performance intensive task and requires the performance of multi-core CPUs. The Windmill dynamic texture generation demo created by Allegorithmic uses dynamic texture generation to create in real-time a texture set that is over 300MB using only 900KB of base texture set. In other words, if a developer were to create this game without using dynamic texture generation, the file size of this game would be approximately 300MB and by using real-time dynamic texture generation techniques, the developer can reduce the size of his game to approximately 1MB. Generating this large texture set in real-time requires significant CPU processing power that cannot be met by single and dual core mobile processors. Real-time texture generation on dual core mobile processor would overload the two processing cores and result in a poor gaming experience. A quad core mobile processor would not only be able to easily handle this task but also have sufficient head room to handle other tasks in the background.

The introduction of quad core CPU based mobile devices will be an inflection point for mobile gaming. The processing capabilities delivered by quad core mobile processors will significantly accelerate development of console and PC class games for mobile devices. Games running on quad core mobile processors will deliver higher level of graphics quality and realism that sets a new standard for mobile gaming experience.

Conclusion

Desktop CPUs transitioned from single core CPU architecture to dual and quad core architectures many years ago. But consumers started to see real benefits of multi-core CPUs only a few years ago. This is because the software ecosystem required to leverage the power of multi-core CPUs was not immediately available for desktop PCs. Except for multitasking with different applications on multiple cores, truly multi-threaded browsers and applications that were capable of leveraging the multiple CPU cores only became available in quantity a few years after the introduction multi-core CPUs. Today's desktop PCs are seeing the many benefits of multi-core CPUs.

In the mobile space, the transition from single core CPUs to multi-core CPUs has been much quicker. The mobile software ecosystem is evolving from the work already done in the desktop space to enable support for multi-core mobile CPUs. Widely used mobile software already supports multitasking and multi-threading.

The Android operating system evolved from Linux, and therefore has native support for multitasking and multi-threading. Recent releases of Android 2.3 and Android 3.0/3.1/3.2 have added several features that improve the operating system's ability to leverage the processing power of multi-core CPUs.

Mobile browsers such as Firefox and Webkit are based on their desktop counterparts and therefore natively include support for multi-threading. The new browser included with Android 3.0 supports not only multi-threading, but also tabbed browsing. These browsers are able to utilize the increased processing power offered by multi-core CPUs to deliver a faster and better Web browsing experience.

Due to rapid growth in mobile gaming, developers are porting popular PC and console game engines to the mobile environment. These game engines were originally developed for multi-core desktop PC platforms, and therefore will leverage the multi-core CPUs in mobile processors to deliver immediate benefits for mobile gamers. Quad core CPUs provide significant processing power for game developers and will enable developers to include advanced physics effects, artificial intelligence, collision detection/avoidance, virtual texturing, better network playability, and more.

Quad core CPUs and variable SMP technology (read more about variable SMP in the whitepaper titled “Variable SMP – A Multi-Core CPU Architecture for Low Power and High Performance”) will enable mobile devices to further push the performance envelope, and allow application and game developers to deliver new mobile experiences, all while extending battery life for the most popular use cases.

Appendix

Coremark Compile Settings for Reported Scores
Project Kal-El (dual core mode, each core running at 1 GHz) CoreMark 1.0 : 5532 / GCC4.4.1 -O3 -mcpu=cortex-a8 -funroll-loops -falign-loops=8 -fgcse-sm -fno-tree-vectorize -marm / Heap / 4:PThreads
Project Kal-El (quad core mode, each core running at 1 GHz) CoreMark 1.0 : 11667 / GCC4.4.1 -O3 -mcpu=cortex-a8 -funroll-loops -falign-loops=8 -fgcse-sm -fno-tree-vectorize -marm / Heap / 4:PThreads
OMAP4430 (each core running at 1 GHz) CoreMark 1.0 : 5673 / GCC4.4.1 -O3 -mcpu=cortex-a8 -funroll-loops -falign-loops=8 -fgcse-sm -fno-tree-vectorize -marm / Heap / 4:PThreads
QC8660 (each core running at 1.2 GHz) CoreMark 1.0 : 5690 / GCC4.4.1 -O3 -mcpu=cortex-a8 -funroll-loops -falign-loops=8 -fgcse-sm -fno-tree-vectorize -marm / Heap / 4:PThreads

Table 1 Coremark Compile Settings for reported scores

Document Revision History

Revision Number	Notes
1.0	First Release
1.1	Added compile settings for Coremark scores reported in the paper