



## Case Study

# Optimizing Cyberlink PowerDVD 10\* Improves Battery Life

## Authors

Manuj Sabharwal and Gael Hofemeier, Software Engineers, Software Solutions Group, Intel Corporation

## Introduction

Low battery life is one of the most serious issues currently plaguing mobile devices in general and Ultrabook™ devices and tablets specifically. Users have become accustomed to streaming multimedia content to their mobile devices “on-demand” from content servers in the cloud. Because these devices have limited battery capacity, energy efficiency is important. Cyberlink PowerDVD 10\* (PowerDVD\*) is one of the top players in the industry for HD, and 3D movie playback. This app is often included as a pre-bundled application from OEMs. In this case study, we showcase how Intel and Cyberlink collaborated to optimize the PowerDVD\* application to give best-in-class experience on Intel devices.

First, we'll talk about the challenges that Cyberlink encountered when adding content streaming features to PowerDVD and the tools and techniques Intel used to improve the power consumption of PowerDVD.

Then, we'll discuss the power consumption profile of a Cyberlink PowerDVD streaming media application and its impact on battery life for mobile devices. We also provide an analysis of PowerDVD behavior to identify issues such as decoding on CPU, large numbers of context switches, high interrupt rates, etc., causing increased power consumption. Finally, we'll provide the data that shows the reduced power consumption following optimization.

The optimization was a huge success. The Intel team was able to make the following improvements to PowerDVD:

- Package C0 reduced to 20% from 100% during media playback
- Reduced SoC power from ~6 W to ~1.8W using Intel® Power Gadget
- Intel® VTune™ analyzer reported CPU utilization of 25% down from 70%
- The Windows\* Performance Analyzer showed frequent wakeups (5 Msec) vs. 10 msec wake up frequency for local or streaming media playback frequency of 10%.

## Definitions



Acronym	Definition
BLA	Battery Life Analyzer
GPU	Graphics processing unit
WPA	Windows Performance Analyzer
DLNA Server	Digital Living Network Alliance Server
HD	High density
SoC	System on Chip
FPS	Frames per second
SDK	Software development kit
SKU	Stock Keeping Unit

## The Challenges of Optimizing Battery Life

PowerDVD offers new features for organizing, streaming media, mobile devices, and social media. In addition to functioning on a client, the latest software can turn a device into a DLNA server and stream multimedia content from a PC across a network to other devices. It can also stream content from external content servers. Adding content streaming came with a price, however. New capabilities, such as HD streaming, required running more processes, consuming much more memory and CPU cycles. This took a toll on battery life. We needed to answer the following questions:

1. What is the power consumption from PowerDVD during a 1080p streaming media playback?
2. Why was PowerDVD able to playback only an hour of media on a fully charged battery?

After two months and three iterations of analysis and validation, the engineering teams improved battery life by making the following changes:

- Offloaded graphics to the GPU (using the Intel® Media SDK)
- Removed the sleep loop calls from two threads
- Used an overlay to reduce extra memory copies

The following describes the process and tools that resulted in the optimized version of PowerDVD.

## Optimization of Cyberlink PowerDVD for Power Consumption

Test System Configuration:

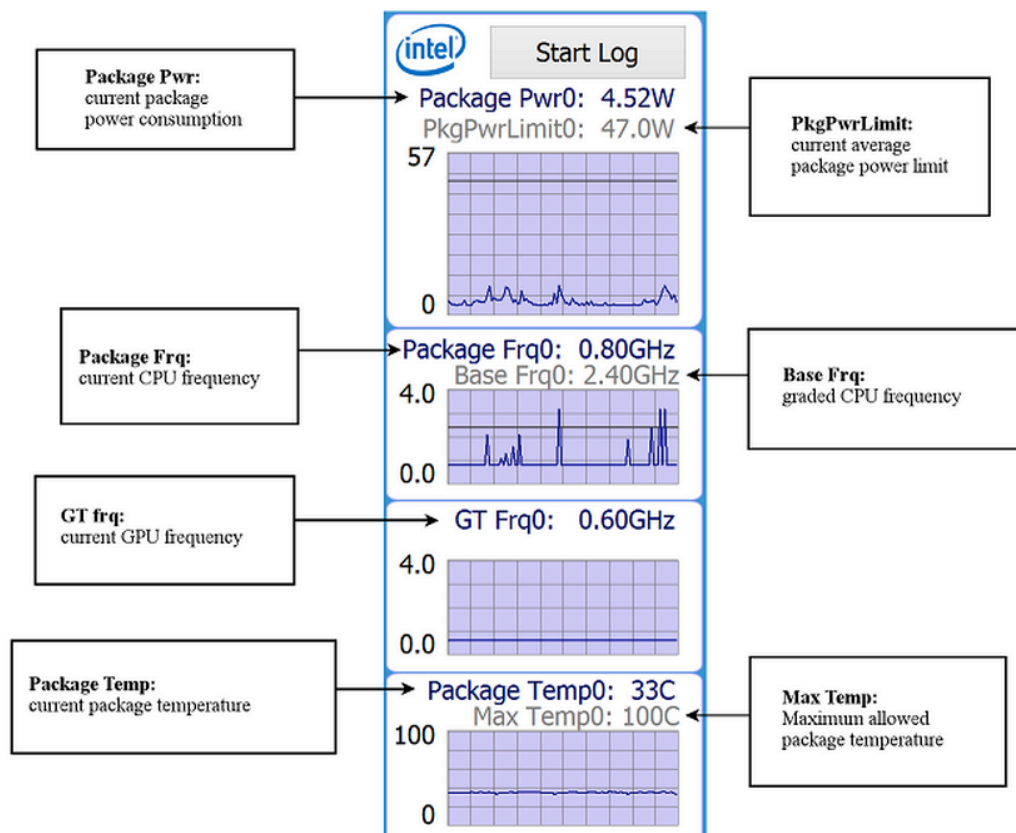
- 4th generation Intel® Core™ i7 processor
- Lenovo Yoga\* 2 Pro
- CPU speed : 1.4 GHz non-turbo frequency
- Memory 4 GB display : 1920x1080p HD panel
- Cyberlink PowerDVD 10 and Cyberlink PowerDVD 12

Validation and analysis showed:

- Package C0 was pegged 100% during media playback, while we expected it to be at 20%.
- Intel Power Gadget showed SoC power to be ~6 W. It should be ~1.7 W on a 4th generation Intel processor.
- Intel VTune results revealed no offloading of graphics to the GPU and high CPU utilization of 70% (we expected about 10%)
- The Windows Performance Analyzer tests revealed frequent wakeups (5 msec). The normal frequency is 10 msec with audio playback.

### First Step - Validation

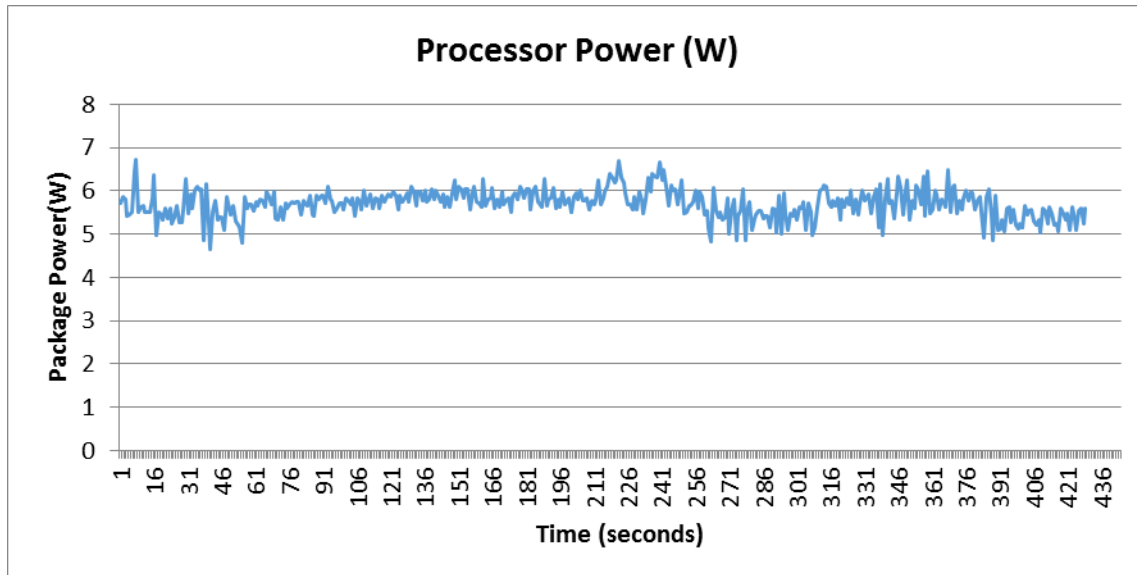
To understand and address PowerDVD's impact on battery life, we used Intel Power Gadget and Battery Life Analyzer (BLA) to validate the application's SoC power usage. Figure 1 shows the Intel Power Gadget's UI on a Windows platform.



**Figure 1. Intel® Power Gadget UI on Windows\* Platform**

As part of our validation of PowerDVD, we used Intel Power Gadget to determine power impacts during playback. Figure 2 shows the power output Intel Power Gadget recorded.

PowerDVD's power usage was ~6 W of SoC power during playback. Intel recommends a maximum of ~2.0 W on 4th generation Intel processors (low power processors typically used in Ultrabook devices).



**Figure 2.** Processor Power Usage during PowerDVD\* Playback

To gain deeper insight into what other activities were affecting power, we used the Battery Life Analyzer (BLA) tool to understand the impact of media playback on residencies. Understanding residency is important as changing the SoC SKU can impact power.

BLA is a power management analysis tool developed by Intel to identify issues that impact battery life. BLA helps to identify a wide range of issues during software analysis such as:

- Software CPU utilization
- OS timer resolution changes
- Frequent C state transitions
- Excessive ISR/DPC activity

Figure 3 shows package residency during 1080p HD video playback using Cyberlink PowerDVD.



C-State Counters	Affinity Mask	Last Period (%)	Average (%)
Package0 C0 - C1	0x0000000f	100.00	99.94
Package0 C2	0x0000000f	0.00	0.06
Package0 C3	0x0000000f	0.00	0.00
Package0 C6	0x0000000f	0.00	0.00
Package0 C7	0x0000000f	0.00	0.00
Package0 C8	0x0000000f	0.00	0.00
Package0 C9	0x0000000f	0.00	0.00
Package0 C10	0x0000000f	0.00	0.00
Core0 C0 - C1	0x00000003	91.99	90.40
Core0 C3	0x00000003	0.00	0.00
Core0 C6	0x00000003	0.00	0.00
Core0 C7 - C10	0x00000003	8.01	9.60
Core1 C0 - C1	0x0000000c	88.92	91.69
Core1 C3	0x0000000c	0.00	0.00
Core1 C6	0x0000000c	0.00	0.00
Core1 C7 - C10	0x0000000c	11.08	8.31

**Figure 3. Package Residency during 1080p HD Video Playback using PowerDVD\***

The package residency includes CPU, Graphics, and UnCore events. More time in package C0 results in higher SoC power. Expected package C0 for Cyberlink PowerDVD 1080p playback is ~20% on 4th generation U-Processor. As we can see from Figure 3, package residency is far higher than it should be.

Both Intel Power Gadget and BLA confirmed higher power usage and ~4 hrs. of battery life on 42 Whr (Watt-hours) battery capacity with ~6 W SoC+3 W of display and 2+ W for other components.

Our next step was to analyze the application for power optimization.

## Second Step - Analysis

For the analysis phase, we used two tools:

- [Intel VTune Analyzer](#)
- [Windows Performance Analyzer](#)

The following tables summarize the results of the analysis, which showed definite room for improvement.

**Table 1. Intel® Power Gadget and BLA Results**

Actual Results	Expected Results
Package C0 is pegged at 100% during media playback	Package C0 should be at 20% during media playback
SoC power using Intel® Power Gadget is ~6 W	SoC power should be ~1.7 W on 4th generation

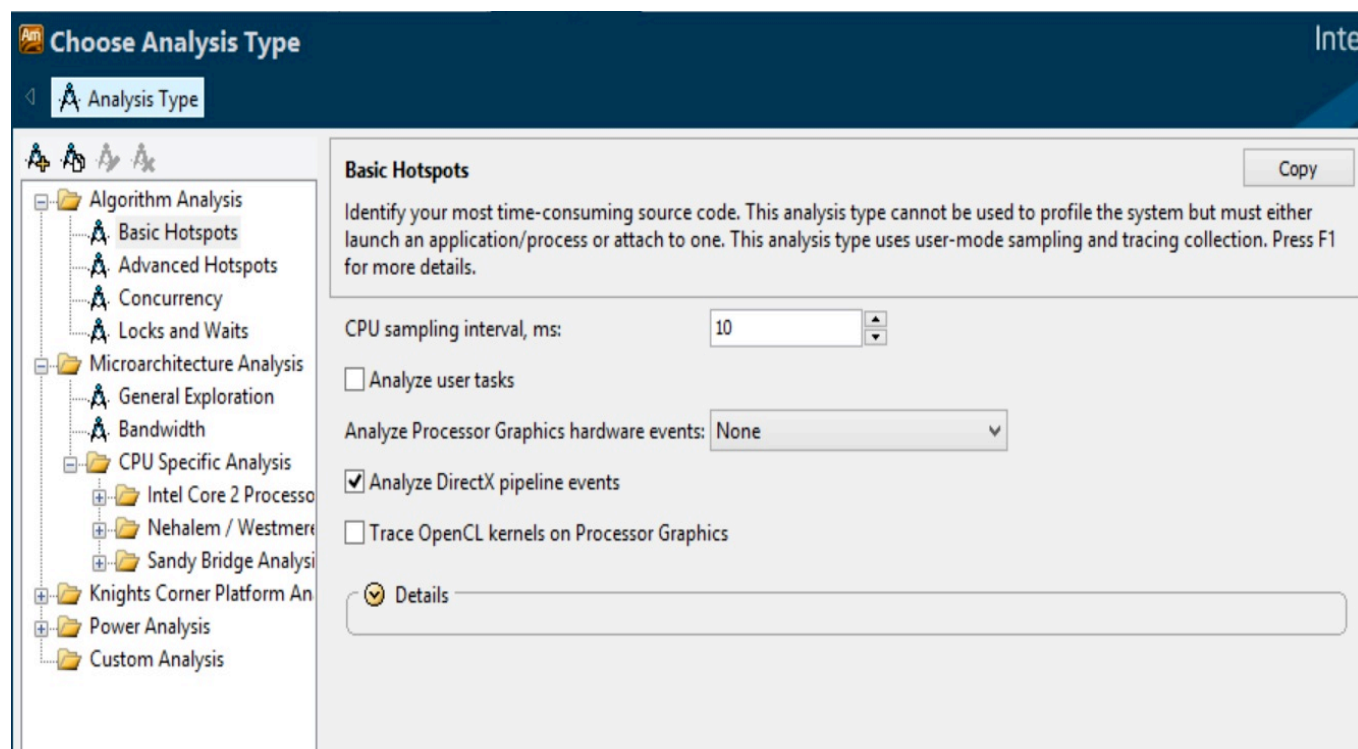
	Intel processor
--	-----------------

**Table 2. Intel® Vtune™ and WPA Results**

[Put column title here]	[Put column title here]
Intel VTune results	1. Since the app had no codecs, there was no offloading to graphics 2. High CPU utilization (70% vs. the expected 10%)
Windows Performance Analyzer	Frequent wakeups (5 msec) occurred– expected frequency is 10 msec with audio playback

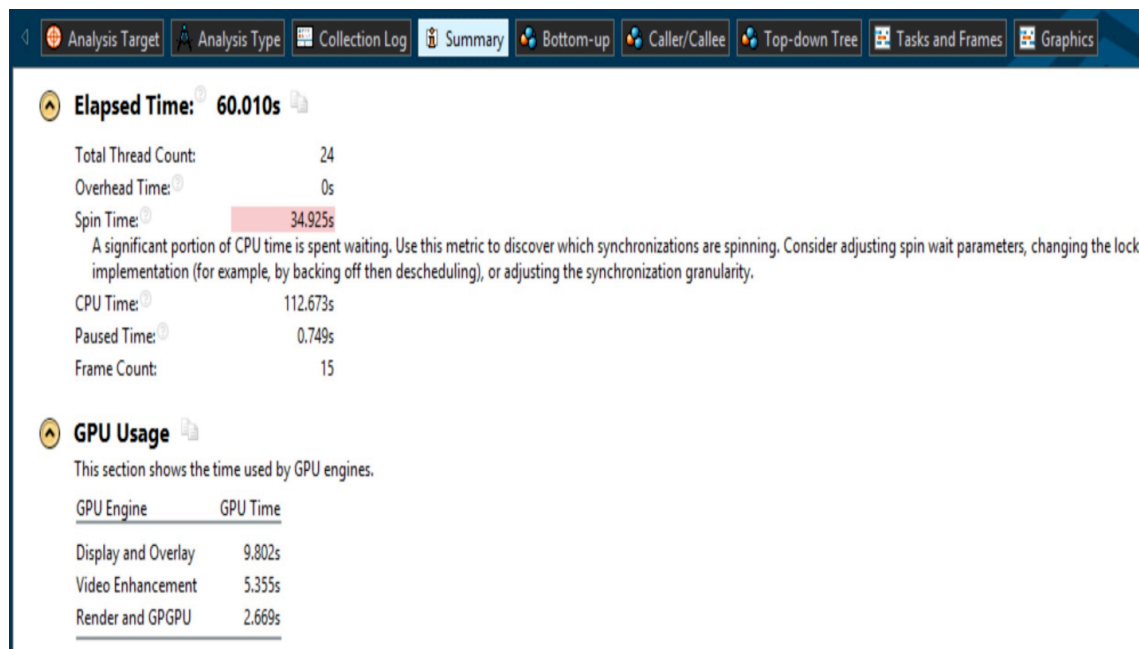
The next figures provide a walkthrough of some of the important screenshots from our analysis.

Intel VTune analyzer was used to validate the PowerDVD application for the presence of spin waits, the presence of hardware acceleration, and hotspots (a micro-architecture issue). Figure 4 shows the steps for collecting the graphics call stacks.



**Figure 4. VTune™ UI for Analyzing DirectX\* Pipeline Events**

Figure 5 shows the VTune summary with significant time spent in spin loop. GPU Usage shows no codec usage. Most of the time spent in the GPU is for display and other pre-processing algorithms during playback.



**Figure 5.** VTune™ Summary showing Spin Loop time

Digging deeper into the analysis, Intel VTune shows high CPU utilization during media playback, and instances where VSync (the red highlights in Figure 5) and GPU software queue are not occurring every ~33 msec (30 FPS playback). This analysis shows software glitches during media playback.

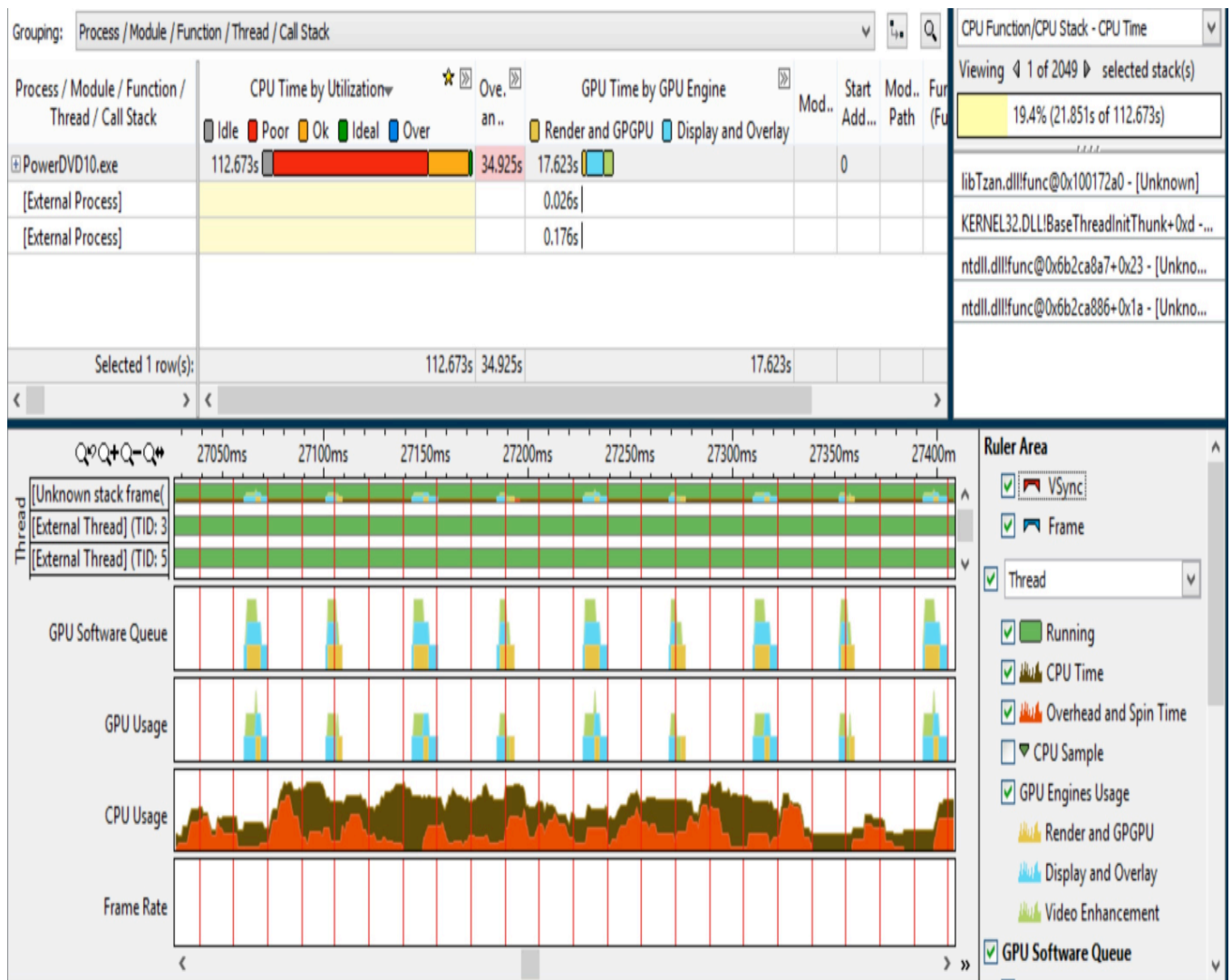


Figure 6. VTune™ Summary Report

Looking at Figure 7, the summary report confirms an inconsistent frame rate over time. The FPS varies for 30 FPS movie playback between 0-60 FPS. The chart shows the total number of frames executed in an application with a specific frame rate. A high number of slow or fast frames signals a performance bottleneck. The goal is to optimize the code to keep the frame rate constant, for example, from 30 to 60 FPS.

## Frame Rate Chart

This histogram shows the total number of frames in your application executed with a specific frame rate. High number of slow or fast frames signals a performance bottleneck. Explore the data provided in the Bottom-up, Top-down Tree, and Timeline panes to identify code regions with the high/slow frame rate. Try to optimize your code to keep the frame rate constant (for example, from 30 to 60 frames per second).

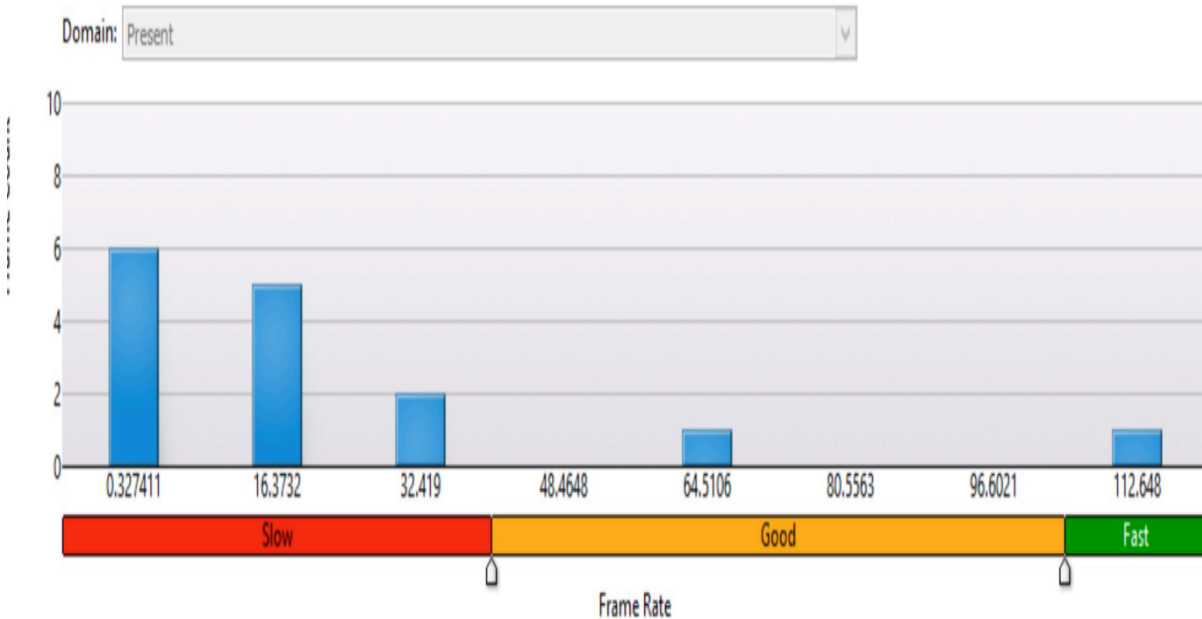
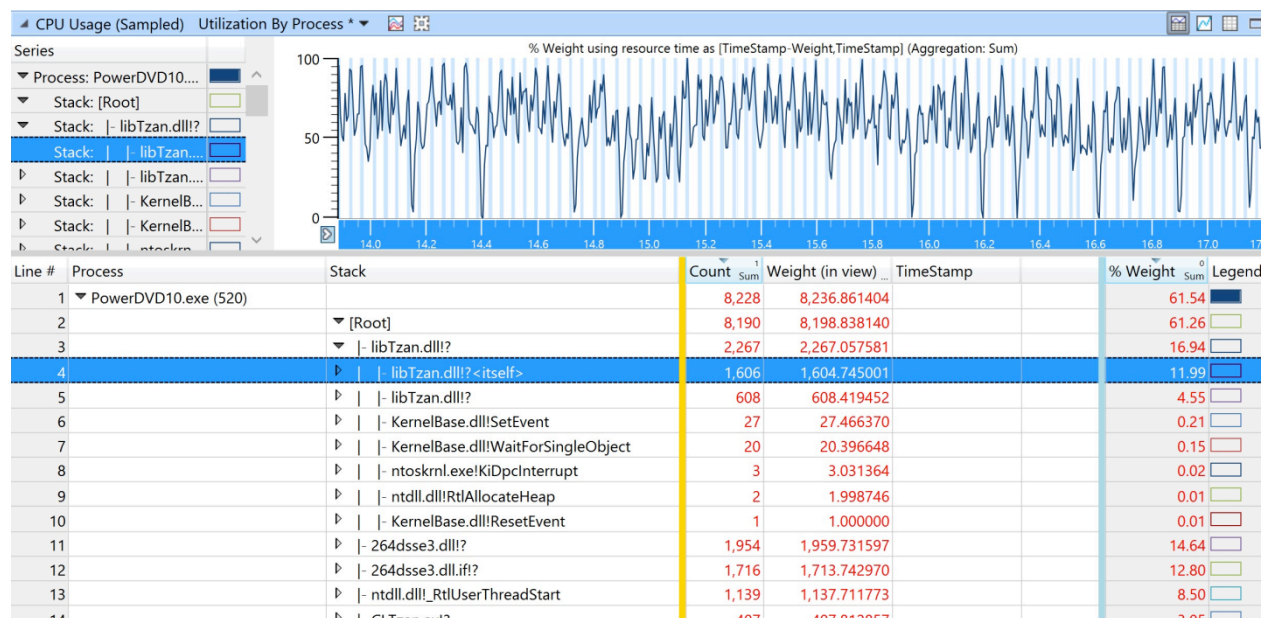


Figure 7. VTune™ analysis of Frame Rates

Next, we used the Windows Performance Analyzer (WPA) tool to analyze the application for wakeup activities, interrupts, and context switches. Figure 8 shows using CPU-based Intel® SSE instructions for H264 decode. It is more efficient to offload this work on to the GPU than to run it on the CPU.



**Figure 8. WPA Analysis of Wakeup Activities, Interrupts, and Context Switches**

WPA also shows wakeup activities from PowerDVD during playback. Figure 9 displays the two PowerDVD threads, both running at 10 msec. The two threads are not coalesced, which causes the overall system to wake up at a 5 msec timer interval. Figure 10 shows the call stack with sleep loop Win32\* API being called every 10 msec interval.



**Figure 9. WPA thread analysis**

NewProcess	NewThreadId	NewThreadStack	Count	ReadyingProcess	Waits (us)	R
Cyberlink PowerDVD10			27		60,865.250	
	2160	[Root]	12		10,137.451	
		ntdll.dll!_RtlUserThreadStart	12		10,137.451	
		ntdll.dll!_RtlUserThreadStart	12		10,137.451	
		kernel32.dll!BaseThreadInitThunk	12		10,137.451	
		MSVCR90.DLL!_endthreadex	12		10,137.451	
		Cyberlink PowerDVD10	12		10,137.451	
		Kernelbase.dll!Sleep	12		10,137.451	
		Kernelbase.dll!SleepEx	12		10,137.451	
		ntdll.dll!ZwDelayExecution	12		10,137.451	
		ntdll.dll!LdrInitializeThunk	12		10,137.451	
		ntdll.dll!_LdrpInitialize	12		10,137.451	
		wow64.dll!Wow64LdrpInitialize	12		10,137.451	
		wow64.dll!RunCpuSimulation	12		10,137.451	
		wow64cpu.dll!Thunk2ArgNSpNSpReloadState	12		10,137.451	
		wow64cpu.dll!CpupSyscallStub	12		10,137.451	
		ntkrnlmp.exe!KiSystemServiceExit	12		10,137.451	
		ntkrnlmp.exe!NtDelayExecution	12		10,137.451	



**Figure 10. WPA call stack with sleep loop analysis**

Table 3 reveals significant reduction in package residency after optimization.

**Table 3. Validating Package Residency after Optimization**

C-state Counters	Average (%) Before Optimization	Average (%) After Optimization
PackageC0-C1	100%	20.18%
PackageC0-C2	0%	8.29%
PackageC0 C3	0%	0.19%
PackageC0 C6	0%	1.91%
PackageC0 C7	0%	69.43%

## Optimization Results/Validation

The following tables show the “before” and “after” results:

**Table 4. Intel® Power Gadget and BLA: Before and After<sup>1</sup>**

Before Optimization	After Optimization
Package C0 is pegged at 100% during media playback	Package C0 is reduced to 20%
SoC power is ~6 W	SoC power reduced to ~1.8 W on test system

**Table 5. Intel® VTune™ Amplifier and WPA Results: Before and After<sup>1</sup>**

	Before	After
<b>Intel® VTune™ Amplifier</b>	<ul style="list-style-type: none"> <li>Since the app had no codecs, there was no offloading to graphics</li> <li>High CPU utilization (70% vs. the expected 10%)</li> </ul>	<ul style="list-style-type: none"> <li>Video codecs now reported</li> <li>CPU utilization decreased by 25%</li> </ul>
<b>Windows Performance Analyzer</b>	Frequent wakeups (5 msec) – expected frequency is 10 msec with audio playback	Sleep thread removed – reduced wakeups by 2x (5 msec to 10 msec)
<b>Battery Life Analyzer</b>	Package residency 100%	Package residency ~20%

<sup>1</sup>Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark\* and MobileMark\*, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. **For more information go to <http://www.intel.com/performance>**

We optimized by:

1. Offloading to Intel® HD Graphics using Intel Media SDK
2. Optimizing Win32 API calls that cause periodic wakeup on CPU
3. Using an overlay to save one memory copy per frame

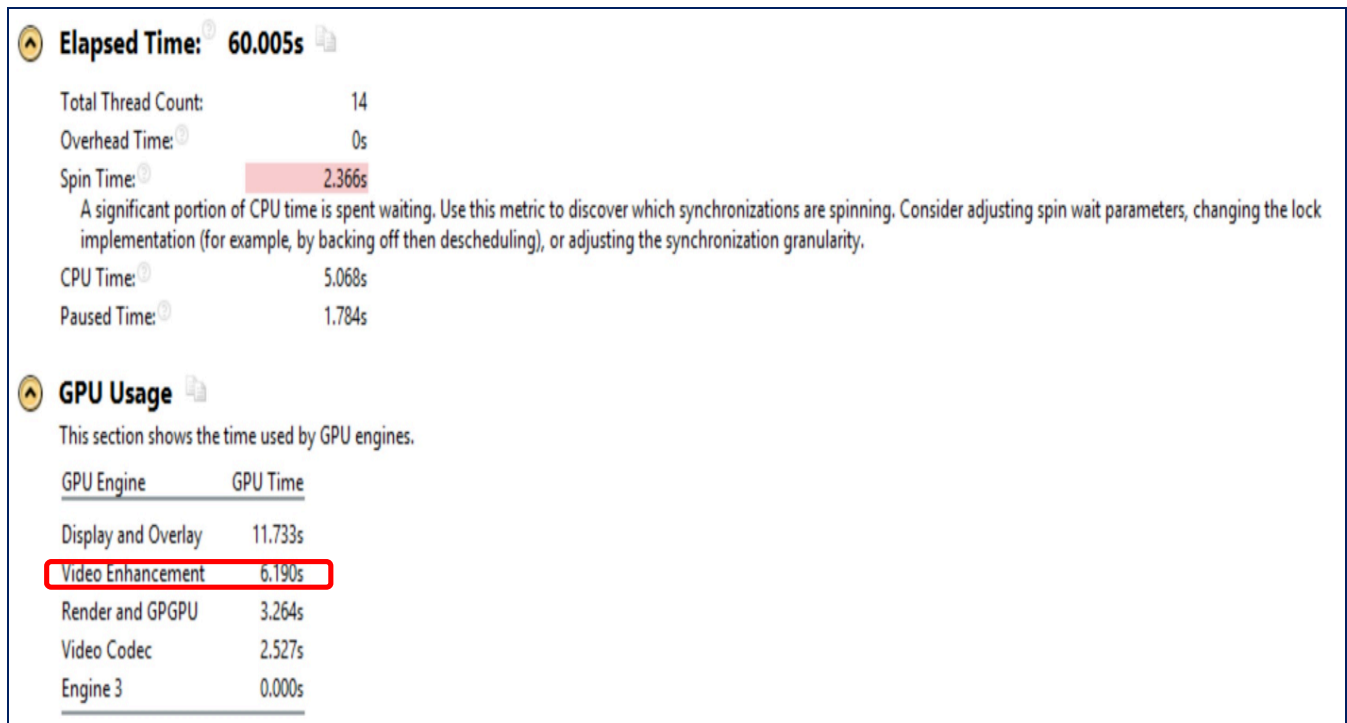
The first task was to use the [Intel Media SDK](#) for offloading decode to graphics which will provide better efficient/watt usage of Intel HD graphics. The pseudo code in Figure 11 provides an example of a simple use of Intel Media SDK to offload a stream of frame to graphics.

**Pseudo code:**

1. Do a while loop while your “bitstream” is true
  - a. Make a call to “decodeDecodeFramAsync with your bitstream as the first parameter followed by your frame, then your sync\_e var
  - b. Then call coreSyncOperation with another sync\_d var.
2. When the while loop is finished, close everything down:
  - a. Make a call to “decodeClose” to close the decode session
  - b. Make a call to “sessionClose” to close the session

**Figure 11.** *Intel® Media SDK code snippet – offloading a frame to graphics.*

Once we offloaded to graphics using the Intel Media SDK, we ran PowerDVD and measured the results using Intel VTune Amplifier. Compared to Figure 5 where we didn’t see any codec usage, we now see Video Enhancement in the summary (Figure 12).



**Figure 12.** Intel® VTune™ Amplifier Summary result

Examining other Intel VTune graphics views, we verified that by using Intel Media SDK use of frame decoded on the GPU vs. on the CPU. Figure 13 shows a batch of frames being decoded after ~20 msec on GPU. Offloading the decode work to the GPU helped to reduce CPU utilization by ~25% on the test system.

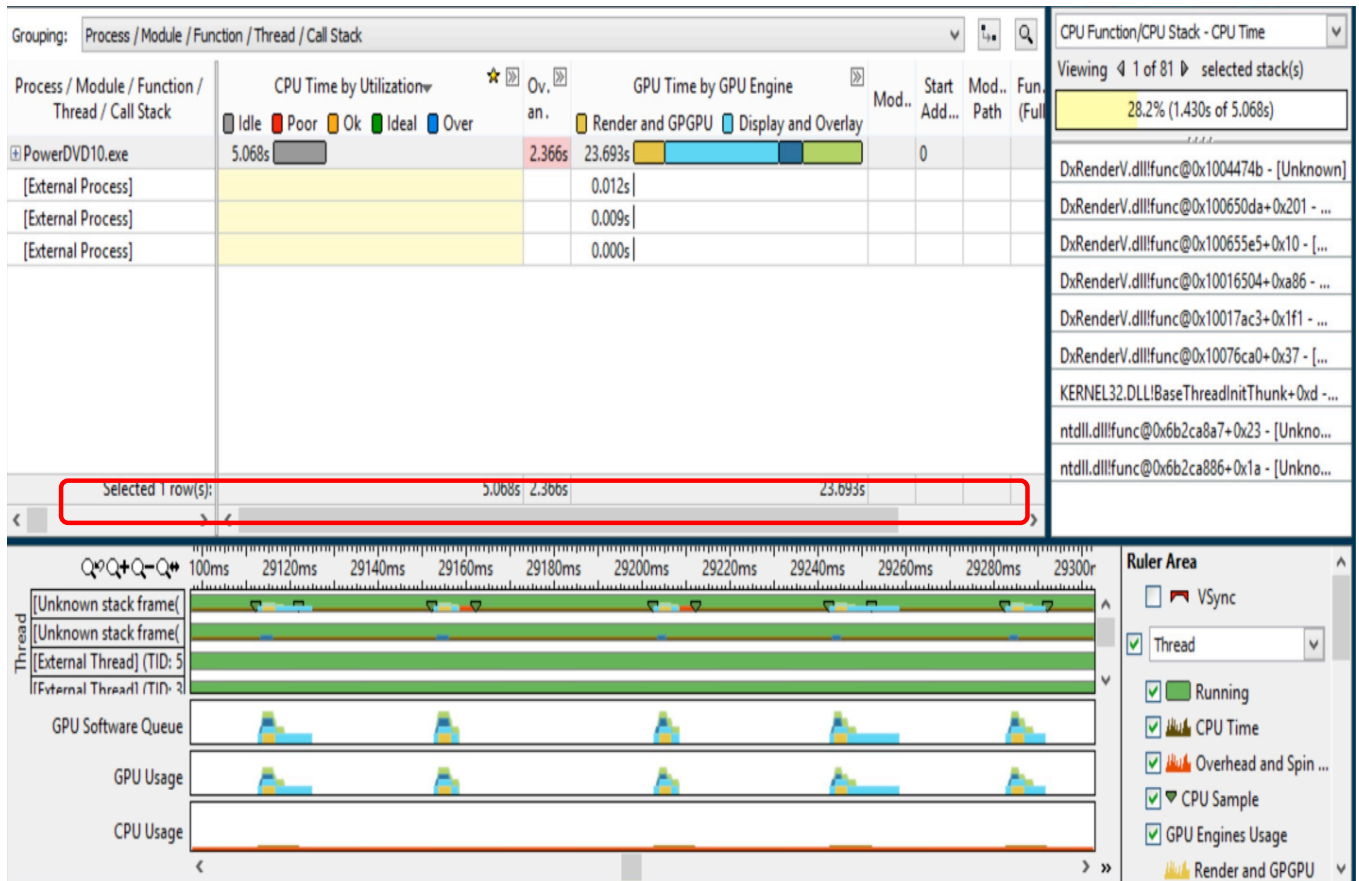


Figure 13. Frame decoding after ~20 msec on the GPU

To verify our optimization of offloading graphics, we ran Intel Power Gadget. Compared to the baseline result shown in Figure 2, we saw ~2 W of power saving just by performing graphics offloading (Figure 14).

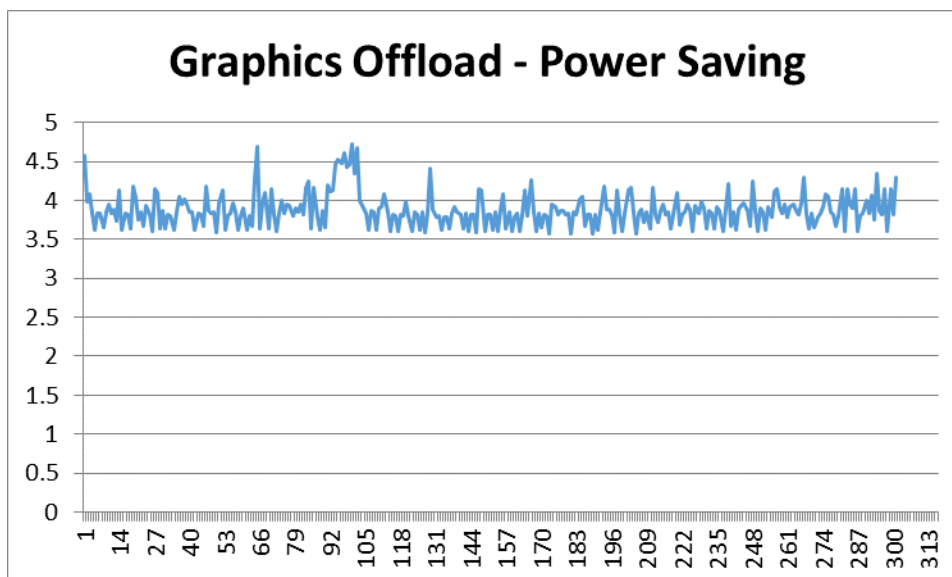
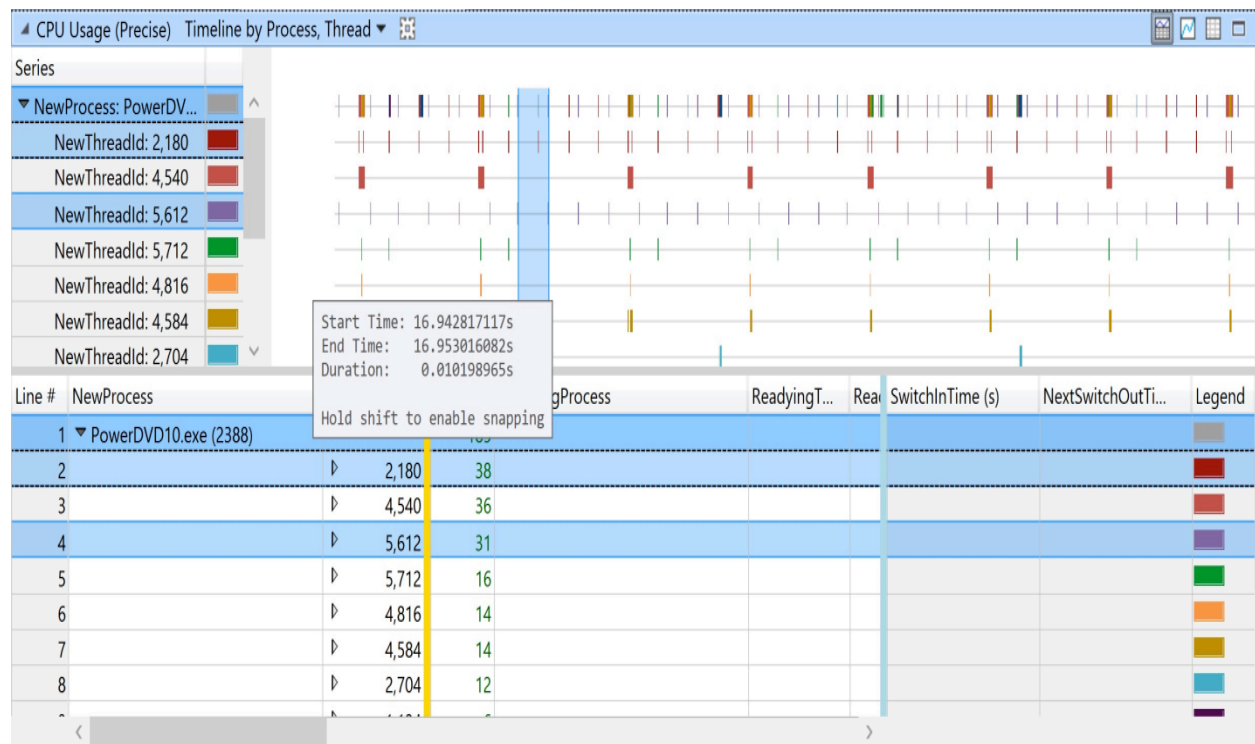


Figure 14. Power Savings resulting from Graphics Offload

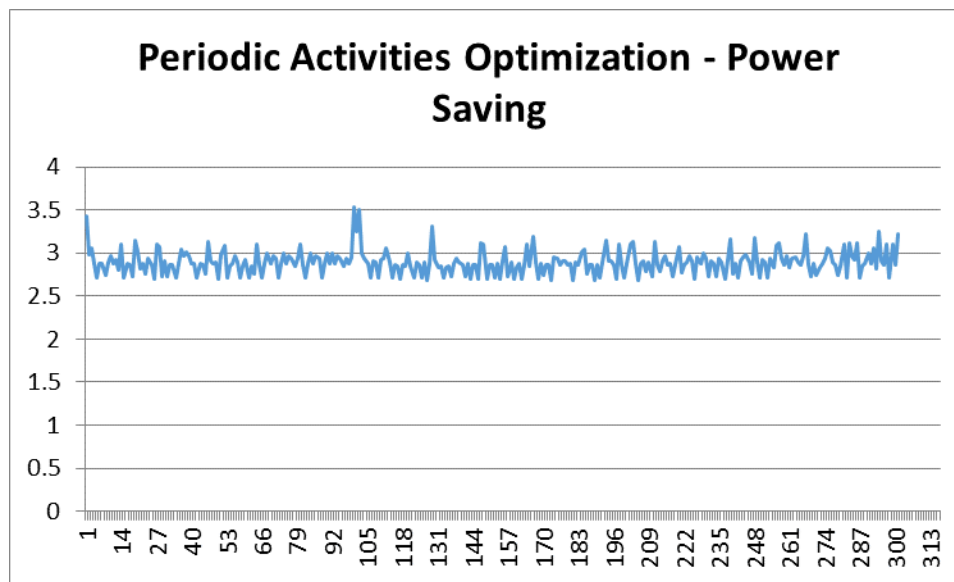
We made some good progress, but ~4 W was not low enough. As stated earlier, the goal for streaming media 1080p playback is ~1.7 W of SoC/package power.

The next step was to find other CPU-based optimizations. Initial analysis showed sleep loop calls from two threads (non-coalesced) waking the CPU every 5 msec. CyberLink engineers needed to remove the sleep threads from their application. However, this was one of the most difficult changes since it required modifying the structure of the application. Figure 15 shows wakeup activities increase to 10 msec after periodic activities were removed.



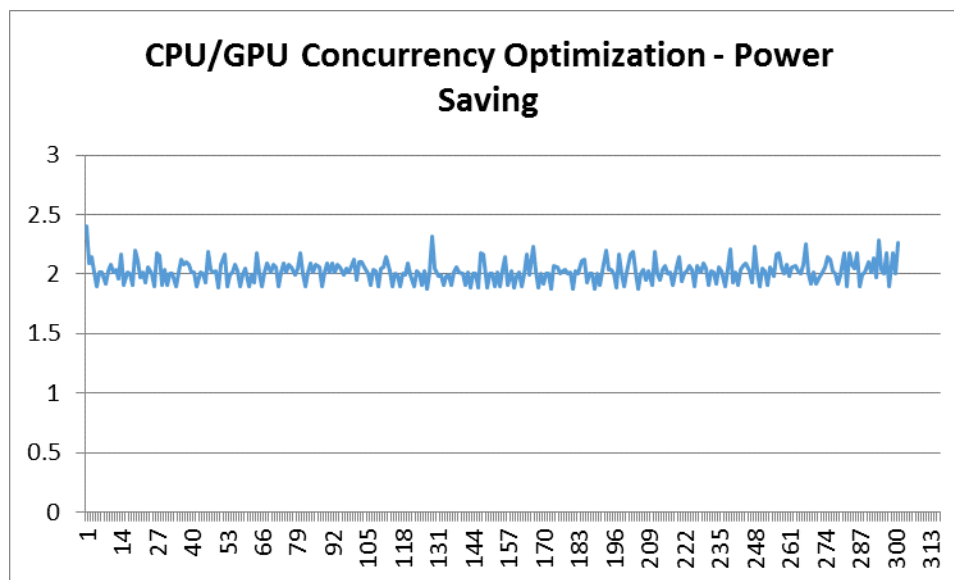
**Figure 15.** Optimized Cyberlink PowerDVD\* after removing periodic activities

Removing periodic activities revealed a ~800 mW saving. With current optimizations, 1080p HD streaming playback SoC power went from ~6 W to 2.8 W, but additional optimizations still had to be done to reach the 1.7 W goal seen in best-in-class applications.



**Figure 16.** Power Optimizations down to ~2.8 W

The next step was to reduce extra memory copies using an overlay. With the overlay, the overall package power was reduced by ~400 mW. Figure 17 shows power was reduced to ~1.8 W from ~6 W.



**Figure 17.** Cyberlink PowerDVD\* at final Power Consumption (1.8 W)

With that, the most important optimization goals had been achieved, and Intel and Cyberlink engineers deemed the project a success.

Close collaboration between Cyberlink and Intel helped to complete the optimization in two months with full validation. The final product with all optimizations was released to OEMs six months from when we started.

## Conclusion

The Intel and PowerDVD engineers used several tools including Intel VTune and Microsoft Windows Performance Analyzer to reach the optimum low-power playback. The collaboration included knowledge sharing on tools with weekly analysis/meetings to meet the battery life goal before the release deadline.

Several iterations were completed before the team was satisfied with their results (PowerDVD consumes ~1.8 W down from ~6 W.) Intel and Cyberlink engineers faced the challenge of keeping the quality of playback the same before and after optimization. Each optimization required a validation and analysis process before it could pass the Cyberlink team's internal quality tests. Thus, every change was tracked and user experience metrics (power and performance) were evaluated.

The following optimizations were found to work the best for achieving the optimization goals, but as noted above, these were accomplished over several iterations:

- Offloading graphics to the GPU (using the Intel Media SDK)
- Removing sleep loop calls from two threads
- Using an overlay to reduce extra memory copies

The combined efforts between the Intel and CyberLink PowerDVD team resulted in optimizing their streaming media playback application to reach the best-in-class goal.

## About the Authors



Manuj Sabharwal is a Software Engineer in the Software Solutions Group at Intel. Manuj has been involved in exploring power enhancement opportunities for idle and active software workloads. He has significant research experience in power efficiency and has delivered tutorials and technical sessions in the industry. He also works on enabling client platforms through software optimization techniques.



Gael Hofemeier has worked for Intel since 2000 as an Application Engineer in the Software Solutions Group at Intel. Gael's current focus is in Technology Evangelism for Business Client Apps and Technologies.



## References

1. Windows Performance Analyzer: <http://www.microsoft.com/en-us/download/details.aspx?id=30652>
2. Battery Life Analyzer: [http://downloadcenter.intel.com/Detail\\_Desc.aspx?agr=Y&DwnldID=19351](http://downloadcenter.intel.com/Detail_Desc.aspx?agr=Y&DwnldID=19351)
3. Intel® Power Gadget: <https://software.intel.com/en-us/articles/intel-power-gadget-20>
4. Cyberlink PowerDVD: [http://www.cyberlink.com/products/powerdvd-ultra/features\\_en\\_US.html?&r=1](http://www.cyberlink.com/products/powerdvd-ultra/features_en_US.html?&r=1)
5. Intel® Media SDK: <https://software.intel.com/en-us/vcsource/tools/media-sdk-clients>

## Relevant Intel Links

Intel processor numbers [http://www.intel.com/products/processor\\_number/](http://www.intel.com/products/processor_number/)

## Notices and Disclaimers

<http://legal.intel.com/Marketing/notices+and+disclaimers.htm>

## Notices

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other



Intel literature, may be obtained by calling 1-800-548-4725, or go to:  
<http://www.intel.com/design/literature.htm>

Intel, the Intel logo, Ultrabook, and VTune are trademarks of Intel Corporation in the U.S. and other countries.

\*Other names and brands may be claimed as the property of others  
Copyright© 2014 Intel Corporation. All rights reserved.



## Optimization Notice

---

### Optimization Notice

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2®, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

**NOTE:** additional notices and disclaimers may be needed, depending on the content of the collateral. Generally, they can be found in the following areas, and must be appended to the Notices section of the paper.

[General Notices](#): Notices placed in all materials distributed or released by Intel

[Benchmarking and Performance Disclaimers](#): Disclaimers for Intel materials that use benchmarks or make performance claims.

[Technical Collateral Disclaimers](#): Disclaimers that should be included in Intel technical materials that describe the form, fit or function of Intel products.

[Technology Notices](#): Notices for Intel materials when the benefits or features of a technology or program are described. Note for technology disclaimers - if every product being discussed (e.g., ACER ULV) has the particular technology/feature, then you can remove the requirements statement in the disclaimer. If you have multiple technical disclaimers, you can consolidate the "your performance may vary" statements and only put in a single "your mileage may vary".