

### The IP Re-Use Challenge

A leading Japanese consumer products manufacturer was faced with the challenge of developing a derivative design platform from an existing design. The objective of the first derivative design was to achieve a specified system performance while executing diverse audio processing algorithms.

The resulting implementation(s) had to be usable in different end products and re-usable over multiple end product generations.

The existing central processing unit (CPU) was already operating at capacity, and a faster CPU could not deliver the requisite performance.

The company's analysis demonstrated that the requisite system performance could be achieved simply by offloading the audio algorithms from the CPU, provided that each algorithm was executed with the same performance as it had been on the CPU. Any performance acceleration would be a bonus.

The implementation of each algorithm in a dedicated fixed function hardware block could not meet the product's multi-product and multi-generational re-usability objectives.

A programmable solution was therefore necessary, but the cost-sensitive nature of the company's products precluded the deployment of multiple general purpose processors.

### Application Coprocessor Re-Use

The manufacturer selected CriticalBlue's Cascade Application Coprocessor synthesis solution to execute the offloaded algorithms.

A Cascade-synthesized application coprocessor offers the programmability of a processor with the parallel processing performance of a fixed function

hardware block. Consequently, it can execute algorithms for which it was not originally designed, affording a high level of re-usability.

The manufacturer synthesized an application coprocessor for each of two audio algorithms – an MP3 decoder and an MPEG1 Layer 2 encoder. Each coprocessor was synthesized using compiled binary executable software code developed for single CPU operation. To speed time to results, the code was used "as is" with no optimization.

The MP3 coprocessor achieved an acceleration of 2.13x over the original CPU implementation, while the MPEG1 coprocessor achieved an acceleration factor of 1.62x (see table 1) at the same clock rate.

The manufacturer then executed each algorithm on the other's coprocessor, achieving acceleration multiples of 1.18x and 1.19x, respectively.

As required, therefore, each coprocessor executed its own algorithm and that of the other coprocessor with a greater performance than that of the CPU at the same clock rate, and simultaneously liberated the CPU to execute other operations.

Consequently, the two coprocessors can be re-used inter-changeably, as required by the demands of the given end product.

Of course, code optimization can greatly improve performance, as can the use of custom functional units to execute serial operations. However, the company's objectives were achieved without such development.

### In Only Three Days

The entire project – encompassing the synthesis of two coprocessors – required three engineer-days of effort.

	Coprocessor Re-Use and Interchangeability	
	MP3 Decoder	MPEG1 Layer 2 Encoder
Algorithm		
Lines of Original Code	~9,700	~16,700
Code Optimized?	No	No
Custom Functional Unit?	No	No
Boost vs. CPU	2.13x	1.62x
Boost vs. CPU on the other's coprocessor	1.18x	1.19x
Effort (in days)	2	1

**Table 1: Coprocessor Reprogrammability and Re-use**