Silaero

HYBRID-FPGA

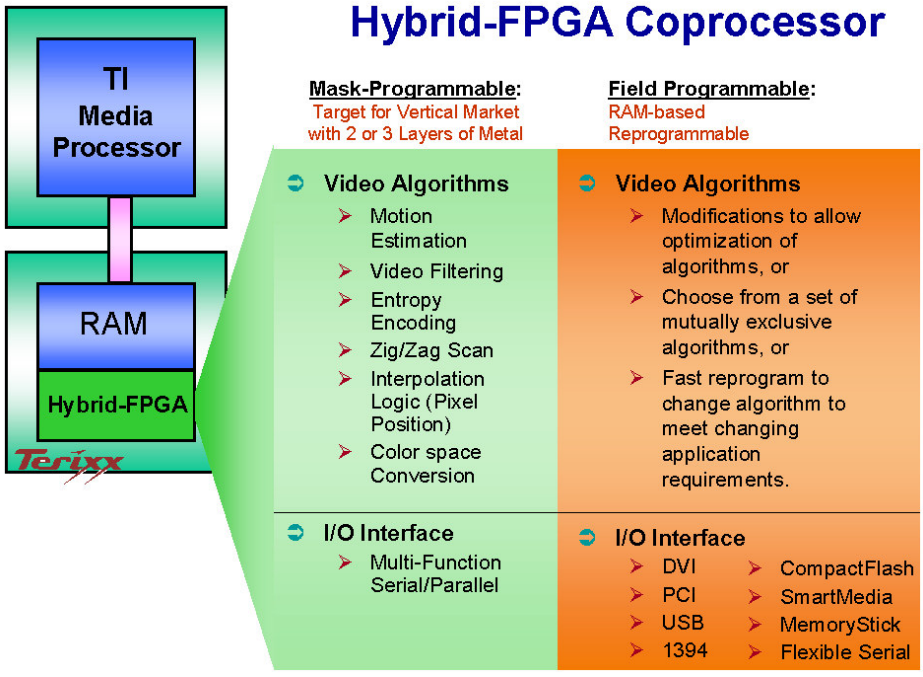
AS A

CO-PROCESSOR FOR CONVENTIONAL SOFTWARE DSPS

OVERVIEW

While one proposed solution for Hybrid-FPGA describes a single device containing a conventional software programmable DSP plus a tightly-coupled layer of configurable logic implemented with our patented Hybrid-FPGA technology, a more loosely-coupled solution is possible. The Hybrid-FPGA combines elements of both ASIC and FPGA capabilities – creating a form of "Application-Specific FPGA". Hybrid-FPGA provides field-programmable functionality for both algorithm acceleration and I/O controller functions.

Hybrid-FPGA technology can be used to create coprocessor devices that interface with standard DSPs such as the TI 6X series Media Processors. The combination of a Hybrid-FPGA based coprocessor with a TI DSP can be compared to the DSP alone or to the combination of the DSP and a fixed function coprocessor constructed with conventional ASIC technology. These different solutions can be compared for both algorithm acceleration level, and I/O function adaptability.



In comparing the performance of a Hybrid-FPGA based coprocessor with a fixed function coprocessor, there are two perspectives:

1) What level of performance can be achieved for identical algorithms;

2) What level of performance can be achieved when modifications are required for a specific algorithm, or alternately a different algorithm is required.

It should not be surprising that for identical algorithm implementations, a fixed coprocessor constructed with conventional ASIC technology will be somewhat faster than our flexible architecture. Of course that is the point – a fixed coprocessor frequently can't

support a broad menu of hardware algorithms or be programmably fine-tuned as desired by the customer. Thus, this discussion will focus on the performance and adaptability of our Hybrid-FPGA technology compared to a conventional hard-wired coprocessor. We will also speculate on the achievable level of acceleration compared with a software programmable DSP.

For this discussion, we will not focus on the specific task division between the software DSP processor device and the Hybrid-FPGA based coprocessor device. This is typically done interactively as part of the development process to ensure the final solution extracts the maximum performance and utility from the combined solution.

ALGORITHM ACCELERATION

Compute-intensive functions such as motion-estimation and video filtering are not only substantially accelerated by a hardware coprocessor – they are also functions where the creativity and skill of a video processing engineer can produce higher value solutions that are better differentiated from their competition.

Motion Estimation

The flexibility of the Hybrid-FPGA will enable significant variations in motion estimation algorithms to be supported, including variations in block sizes and block matching/comparison algorithms, as well as overlapping block searches. Regarding different block sizes, a good example of variation is the move toward H.264 where variable block sizes as small as 4x4 must be supported. A hard-wired coprocessor architecture aimed at traditional MPEG would need to be respun for H.264, rather than just reprogramming it.

While MPEG uses the DCT transform function, an ever-increasing number of non-MPEG video algorithms, as well as JPEG-2000, are based on the wavelet function. When wavelets are used, the motion estimation function is considerably different, again requiring a hardware change for a conventional coprocessor. With the Hybrid-FPGA, it would even be possible to support both MPEG-style video (for video clips) and JPEG-2000 still picture compression with the same coprocessor used in a digital camera.

Video Filtering

While MPEG is a strict standard for the decoder allowing less design freedom, the designer has many degrees of freedom in the encoder design, especially with regard to filtering. Filtering may be used for input noise filtering, output filtering with regard to the motion vectors, and/or within the encode loop itself. Pre-filtering is often used for noise reduction or contrast enhancement. Post filtering or loop filtering can remove blocking and ringing artifacts that may be visible.

Different filter strategies are used with respect to spatial, temporal, spatio-temporal, and even adaptive filtering. Within a filter, different combinations of algorithms may be utilized including FIR, IIR, and Median filters. Different mask sizes may be utilized. Filtering occurs at high bit rates and is best served by hardware acceleration. Changes in filtering techniques will often require hardware changes, if the designer is to have the flexibility desired.

Other Video Functions

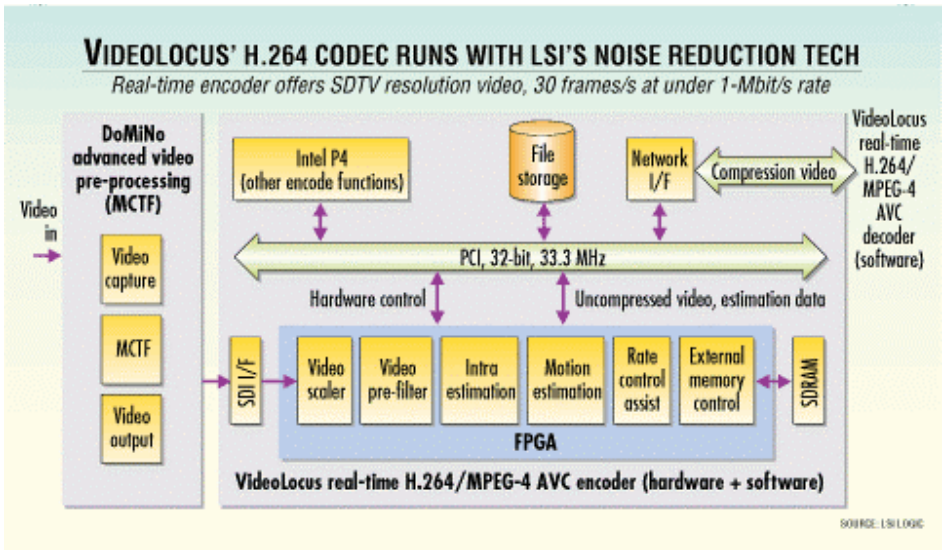
Choices also exist in other functions that lend themselves to hardware acceleration. Entropy coding is often performed by Huffman encoding; however it may also be done by arithmetic coding, necessitating a hardware change if performed by a hard-wired coprocessor.

Sequence control may also benefit from hardware reprogramming flexibility. Improved video processing may incorporate different strategies for the sequence and relationship of I, P, and B frames. More complex inter-frame relationships are also possible. The sequence in which certain algorithms are applied within the coprocessor may need to vary for optimum results in different applications. For these and other reasons, it is useful to have the primary control sequencer for the coprocessor implemented with programmable logic.

Degree of Acceleration

To speculate on the level of acceleration that is possible, we have two sources that provide some approximation. The first is an estimate from an experienced designer that on .13u, is that it would be possible to build a hardware accelerator with performance of ~40 BOPS overall and over 200 BOPS for Motion Estimation specifically.

A second reference point is based on a system architecture proposed for H.264 by VideoLocus in conjunction with LSI Logic.



A representative of VideoLocus stated that this system provides motion-estimation, intra-estimation, mode decision statistics, and video pre-processing support with a throughput rate 300x that of a 2GHz P4.

Worth noticing in this diagram is the use of the LSI Logic DoMiNo video processor chip. While in theory this device is capable of implementing a complete and relatively advanced video codec function, in this instance it is being used solely for its pre-processing capabilities. Although intended to have some flexibility, this device was obviously not able to adapt to the H.264 functionality. An FPGA was utilized for the compute intensive functions in this example.

Of course, since a number of video coprocessing functions have been designed into TI DSPs in the past, other reference points exist. As stated earlier, a Hybrid-FPGA

implementation will be somewhat slower than a fixed function hard-wired coprocessor for identical functions, but will adapt to desired customer needs. The Hybrid-FPGA solution offers more overall performance and/or higher image quality than can be achieved with a fixed function coprocessor.

I/O FUNCTION ADAPTIBILITY

The best video processing solutions often face barriers to customer acceptance, if they cannot interface with a customer's system. This can involve a number of standard general interfaces like FireWire, PCI, USB and DVI, or some form of custom parallel or serial interface. In particular, serial interfaces have increased in popularity. Many DSP devices traditionally have some form of generalized serial interface where control bits can be set to tailor the function to match up with the customer's system. Unfortunately, even with this configurability, many times incompatibilities exists. When this happens, the DSP supplier is faced with building a custom device to get the business or passing on the opportunity. The field-programmability contained in the Hybrid-FPGA can enable these custom interfaces to be easily constructed.

In the consumer world, video processing systems often must interface with some form of non-volatile memory medium. Some examples are CompactFlash, SmartMedia, MemoryStick, MMC, and SD. Each of these requires a different interface standard with a different I/O controller functionality that can be readily implemented in a Hybrid-FPGA. Of course external memory interfaces like SDRAM and DDR DRAM may also be required for some applications.

SUMMARY

We believe the combination of performance and flexibility is the real winner in tackling ever-changing customer design requirements.

Video Processing Solutions

Solution	Overall Performance	I/O Flexibility
Media Processor Alone	Lowest	Low
MP + Fixed Coprocessor	Fastest for standard algorithm	Low
MP + Hybrid-FPGA based Coprocessor	Fastest over all algorithms	High

The focus should be to meet and exceed the customer's application performance metric, while providing the flexibility required in a diverse market that often requires algorithm tweaking and changing. The Hybrid-FPGA solution will better enable designers to differentiate and add value to their products, as well as meet evolving standards. In addition, the unique ability of the Hybrid-FPGA architecture to provide configurable I/O functions is both important to manage the sheer number of chip combinations and to enable selling into a diverse market space.