

Processor Architecture for High Performance Video Decode

Jonah Probell

Ultra Data

High definition television (HDTV) content broadcasts over cable and from terrestrial and satellite stations are appearing in viewing markets around the world. Due to consumer demand and government mandates, HDTV capable digital consumer video equipment will be sold in high volumes in the next few years. Chipmakers, today, are designing the chips to drive these devices.

HDTV is displayed at a resolution of 1920 x 1080 pixels. That is six times the number of pixels of a standard 720 x 480 NTSC video display. To store that many more pixels on a standard DVD sized disk or broadcast that many more pixels with the same transmission bandwidth as standard definition television requires improved algorithms for video compression. Similarly, to transmit video in the limited bandwidth available to portable wireless devices requires improved algorithms for video compression.

Next generation video coding standards have been developed to deliver improved video compression. For a given display resolution, the computational requirement of a next generation codec may be as high as ten times that of MPEG2. This is compounded by the higher resolutions required by future video displays. A video processor for HDTV requires about 60 times the processing power of a video processor for NTSC television. That is nearly 9 years worth of performance scaling according to Moore's law.

MPEG2 has been a nearly universal video compression standard for many years. Several video compression standards, offering greater compression-quality, hope to displace MPEG2 in next generation video applications. H.264 succeeds developments in H.263 and MPEG4 by the ITU-T and ISO open standards bodies. By lineage, H.264 is the grandchild of MPEG2. Microsoft's Windows Media Video 9 (WMV9) standard is making a play, with lower royalties, for many of the same application spaces as H.264. Promoted by such an influential company, WMV9 cannot be ignored. On2 Technologies' next generation VP6 video compression standard has also seen adoption in royalty sensitive markets. No next generation video standard promises the near universal acceptance that MPEG2 has enjoyed. We should expect a future in which consumer video devices must support multiple video compression standards.

MPEG2 video decode, within chips for consumer video applications, has been most often implemented in fixed-function hardware. MPEG2 decode can just as easily be implemented in software and run on a software programmable processors. The reason for using fixed-function hardware is that it can achieve the required performance with a smaller die area than a software programmable processor. A software programmable processor makes sense when the application requires more variations than can be supported in a fixed-function processor. For example, consumer video devices that require support for multiple video compression standards achieve a smaller die area by using a single software programmable processor for video decode than by using multiple pieces of fixed-function hardware.

Different video compression standards are supported in such a processor by different software functions.

Because of the performance demanding complexities of next generation video compression standards neither today's blazingly fast Pentiums nor today's video optimized digital signal processors are capable of decoding bitstreams of next generation coding standards at the resolution and frame rate of HDTV. At Ultra Data I am working on developing the UD3000, the world's first software programmable processor architecture capable of decoding bitstreams of next generation video standards for HDTV. The UD3000 achieves sufficiently high performance through a multi-core processor.

A homogeneous multiprocessor consists of identical processor cores. A symmetric multiprocessor runs the same software on each processor core. Symmetric homogeneous multiprocessing can yield approximately linear system performance improvement with an increase in processor count for applications that operate on independent data. This approach is often used for IP packet header parsing in network routers because each packet is independent of all other packets.

Video compression is achieved by exploiting spatial and temporal redundancy of data in the video frame sequence. As a result, data dependencies exist in the reconstruction of the decoded video frame sequence that prevent significant performance gains by increasing processor count in a symmetric manner. Dividing the video frame spatially among symmetric processors does not work because intra prediction (prediction from neighboring pixels) and deblocking filtering along the edges of the sub-frame processed by each processor would depend on the results of other processors. Interleaving the frame sequence temporally among symmetric processors does not work because inter prediction (prediction from previously coded frames) and adaptive parameters would depend on other processors' results.

The type of multiprocessing that gives significant, though sub-linear, performance scaling with processor count for video decode is multiprocessing with a pipelined data flow through asymmetric (different software) processors. Since different processors in the data flow pipeline run different software, each can be optimized for its own processing task. This yields a heterogeneous (different hardware) multiprocessor design.

A fixed-function hardware video decoder also implements a dataflow pipeline with heterogeneous processing elements. The only difference between a fixed-function hardware video decoder and an asymmetric heterogeneous multiprocessor with a pipelined data flow is that the multiprocessor can be programmed with different software functions to implement different video compression standards.

The major steps of video decode and display are prediction, inverse transform, deblocking filtering, color space conversion, frame scaling, synchronization, stream mixing, overlays, color correction, and interlacing. Most of those steps are required by most applications, though not necessarily in the order listed. All video applications require prediction, inverse transform, and deblocking filtering. The method of prediction and inverse transform is specified by the video compression standard. For the H.264 standard, the deblocking filtering algorithm is also specified. Color space conversion, frame scaling, synchronization, stream mixing,

overlays, color correction, and interlacing are not specified by video compression standards.

<SIDEBAR: The steps of video decode and display, defined –

Prediction (inter) – Predicting the value of a video block from the value of one or a weighted average of more than one previously coded “reference” video blocks. Reference blocks may be shifted sub-pixel distances by interpolating values between the pixels of a reference block.

Prediction (intra) – Predicting the value of a video block from the values of selected pixels in neighboring blocks. Different modes will use different source pixels.

Inverse transform – The conversion of quantized coefficients from the spatial frequency to the spatial location domain in order to derive a block of differences used to correct prediction error.

Deblocking filter – The process of smoothing the visible artifacts that appear at the edges between blocks due to the effects of data lost in the compression process. A good deblocking filter will smooth most compression artifacts without blurring most real edges in the video image.

Color space conversion – The transformation of coded luminance and down-sampled chrominance differences into a full resolution red, green, blue representation.

Frame scaling – The increasing or decreasing of the resolution of the coded video frame to match the resolution of the display.

Synchronization – The matching, in time, of multiple video, audio, caption, and other data streams.

Stream mixing – The overlaying of one video stream on another, such as for a picture-in-picture display.

Overlays – The application of generated image data, such as on-screen display text, over the video image.

Color correction – The scaling of color values, according to a model of the display parameters, to correct for non-linearity in the representation of colors on the display.

Interlacing – The conversion between frames drawn at a given refresh rate and fields of interleaved lines drawn at twice that rate as needed for the display.

:END SIDEBAR>

Each of these steps, for most video applications, operates on structures containing 8-bit data elements. Intermediate calculations within each step require 16-bit data resolution. The internal datapaths of today’s general-purpose digital signal processors are optimized for audio, which operate on 16, 24, or 32-bit data. Such processors are not optimally efficient for video processing applications.

Furthermore, video processing operates on two-dimensional data. Video processor instruction sets are optimized for two-dimensional filters and transforms using matrix multiplication unlike general-purpose DSPs that perform linear filters and transforms using simple multiplication operations. Video processors also like to

access elements of two-dimensional arrays in horizontal and vertical sequences, unlike general-purpose DSPs.

Video decode applications have hard real-time requirements. For general-purpose processors that use caches and take interrupts it is difficult to achieve the performance required for real-time due to the non-deterministic length of time required to service cache misses and interrupts. In an asymmetric heterogeneous multiprocessor with a pipelined data flow for video it is possible to perform each off-chip reference data read early enough that the read data is available before it is needed. Since latency variation is masked by an early read, the only remaining concern is for sufficient off-chip memory bandwidth. An efficient multiprocessor architecture for video will include enough storage to hold all intermediate data structures locally in on-chip RAM to minimize off-chip bandwidth consumption.

The off-chip memory bandwidth required to decode the theoretical worst-case H.264 video bitstream exceeds the bandwidth available in a 64-bit wide 200 MHz DDR SDRAM. A large majority of the bandwidth is due to cycles spent waiting for DRAM row activation during the row address select phase of an access. Much of this delay is an unavoidable result of the fact that there is no predictability to inter prediction read sources. Optimal organization of data within memory can minimize but not eliminate cycles wasted for RAS delays. Organizing video frame buffers with square groups of pixels in consecutive DRAM addresses such that each group is the size of one DRAM row means that most block reads will fall within a single DRAM row and in the worst case will span groups stored in four different DRAM rows. This means that a block read will require at most four RAS delays.

Each of the step algorithms executed to transform a bitstream into a video display requires a different combination of decision-making and mathematical calculation. Traditional high-speed general-purpose RISC processors are efficient at the branching operations required for decision-making heavy algorithms. Traditional general-purpose DSPs, with their single instruction multiple data (SIMD) parallelism, are efficient at intense mathematical calculations. A VLIW architecture with parallel RISC and DSP execution units allows the most precise fine-grained matching of high performance mathematical operations and decision making.

When studying video processors consider the value of software programmability. Consider optimizing the mix of RISC and DSP processors for the decode and display steps required. Also consider the dataflow within the video processor and to off-chip storage. Finally, make sure that your architecture can achieve the required real-time performance for a worst-case bitstream with worst-case memory access latencies.

--JP

Jonah Probell is the Founder and Principal Engineer of Ultra Data Corporation in Waltham, Massachusetts <http://www.ultradatacorp.com>. Ultra Data licenses advanced video processor architectures, designs, and software for industry standard codecs to developers of consumer video devices.