A Scalable Media Processor Based Image Data Path for the Xerox DocuSP Platform

Norm Zeck and Bill Nelson
Xerox Corporation
Webster, New York

Abstract

The Xerox Document Services Platform (DocuSP) is the Xerox Digital Front End platform for production products. DocuSP supports over 26 different production engines that include monochrome, color, cut sheet, and continuous feed products. In particular DocuSP is one of the Digital Front Ends for the Xerox iGen3 Digital Press. This talk will describe some unique technologies in the DocuSP DFE that enable high performance printing at 100 pages per minute in full color. Unique to this controller is the concept of a scalable performance, "soft" image data path, implemented in array of low cost media processors from Equator Technologies that provide the image processing required for full performance printing on iGen3. This allows DocuSP to be an agile platform to support new imaging features without the overhead of lengthy hardware development.

Introduction

The current Xerox Document Services Platform (DocuSP) is a Digital Front End (DFE) for over 26 different production printing products. These products range from monochrome, highlight color, full color, cut sheet to continuous feed, from 60 to over 1000 ppm. DocuSP builds on the monochrome DocuTech workflow that has proven to be very effective in the production printing market. One of the features of this workflow is an architecture that supports full productivity printing for complex job content. To better enable this feature, the DocuSP architecture provides a compressed/spooled ready-to-print (RTP) file that can be used as part of a fast reprint workflow or multiple copy performance enhancement. A diagram of the image data path for DocuSP is in Figure 1. Print stream data, such as PostScript and PDF, is processed to a ready to print format. This is typically a raster format and is often compressed to reduce storage or bandwidth. The RTP data is then buffered in memory to compensate for the print engine paper path and also in parallel spooled to a disk for reprint or multiple copies. Since the print data stream can have unbounded complexity, this buffering to disk insures that copies 2-N or reprints will print at full rated speed independent of data stream complexity. To implement this in monochrome, specialized compression hardware devices were developed in the 1980's when DocuTech was developed and have also been used in DocuSP.

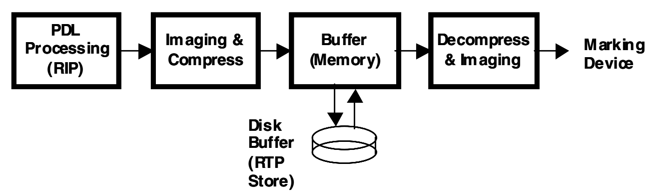

Figure 1. DocuSP Image Data Path

As this platform transitioned to color with the introduction of the Xerox DocuColor 2060/6060 and more recently the iGen3 Digital Press, the need for compression was even greater. The data rates required to enable full printing productivity on iGen3 are in excess of 400 mega bytes per second. In addition, the marking process in iGen3 has demonstrated benchmark image quality for a digital press. This capability presented high demands on the compression used in the DocuSP color platform so as to not compromise image quality in the image data path.

In addition to compression, we recognized through our experience with monochrome production printing that the requirements and workflow in color would be dynamic. As such, we needed a way to not only provide the compression capability for DocuSP but also provide a way to quickly upgrade the platform to meet new customer requirements and workflows.

The rest of this paper will describe the solution we developed in a compression technology, and the approach we took to implementing that technology in a media DSP from Equator Technologies. The architecture we developed with this media processor is also scalable by adding more processors to meet new application requirements or to configure DocuSP for higher speed print engines.

Xerox Multi-Mode Compression (XM2)

A key requirement for the compression used in DocuSP color product is the reduction of bandwidth. This is both in reducing the number of disks required to store and retrieve the RTP file, but also to reduce the bandwidth required on internal data buses in the computing platform that DocuSP runs on, in this case a SUN workstation. To meet the expected cost/performance requirements of the range of products supported by DocuSP a ratio of 15:1 was chosen. This is based on an input pixel that is 8 bits per component plus a 4 bit rendering tag that results in 48 bits per pixel that needs to be compressed for CMYK separations.

Traditional compression technologies alone would not meet this requirement and maintain image quality required. Lossless compression technologies such as dictionary coding, LZ1 or LZ2[1] can clearly meet image quality requirements, but not the compression ratio. These technologies are excellent for computer rendered objects such as text and graphics achieving typically on the order of 20:1 to 100:1 compression ratios. However, these technologies under perform for raster data from a scanner that typically contain a lot of noise. In this case compression ratios under 2:1 are typical, which is well under our 15:1 goal.

Lossy compression technologies such as transform coding with coefficient quantization can meet the compression rate goals by adjustment of the quantization tables. However on pixel data that has high frequency components such as sharp edges, this technology does not often meet the image quality requirements. Note that these sharp edge features can exist in both computer rendered data and raster data that is included in a page description language such as PostScript or PDF.

Xerox has long used a technology called segmentation to help with the processing of data in digital printing. The DocuTech series products use segmentation in the scanner to help with processing and rendering different content. For this application, we looked at developing segmentation that would be optimized for color compression. Unlike segmentation for scanning applications where the content type was important such as text, graphics, halftone and so on, we were only concerned about the compression rate and image quality. This lead to development of a segmenter concept that was designed not for image type, but designed as how to best compress that particular data with specific optimized compression technologies. In this way we were able to insure the image quality with the segmenter and with optimized compression technologies obtain the compression ratio required. This compression technology is what we call Xerox Multi-Mode or XM2 compression in that multiple compression technologies can be used on a document determined by the image quality and compression goals of the system.

Table 1. Lossless compression optimization by vector ordering of pixels

Document	Pixel Ordering	Compression Ratio
Text	Vector	29.38:1
	Scan line	22.27:1
Mixed Text and Graphics	Vector	17.39:1
	Scan line	11.67:1

While the segmentation allows us to apply different compression to areas of the document, we also realized that we needed optimized compression to achieve our goals. An example of one such optimization is ordering of pixels in a block format as a vector such that there is high correlation between pixels. In this way we can apply standard sliding window dictionary compression technology to the reordered data to achieve better lossless results. Table 1 shows example improvements on text and mixed content documents.

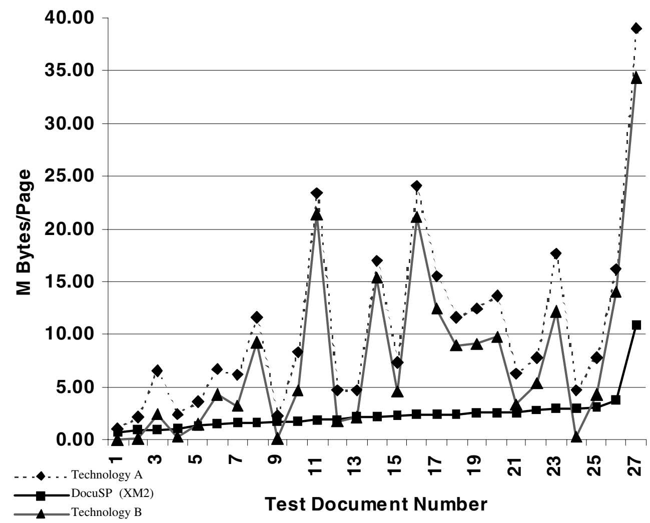

Figure 2. XM2 Compression Technology Benchmark. Average Mbytes per page for a set of color test documents. Smaller average size is better, resulting in lower bandwidth requirements.

This technology has proven to work quite well in the compression application for digital printing. As a comparative example of the value this technology provides, the graph in Figure 2 shows benchmarking we have done with XM2 compared to two leading technologies that provide similar capabilities as DocuSP in digital color printing products. The graph shows a plot of average page size in mega bytes per page for a set of test documents. The test documents range from simple, standard, complex with complexity defined as a measure of content type with mixed content, in particular difficult images in the complex set of documents. The plots show that XM2 typically achieves the lowest file size for the document set. Most notable is that XM2 provides a very uniform file size that actually allows us to cost optimize the design of the DocuSP platform for a particular print engine and still ensure full productivity.

Media Processor Image Data Path

Early in the design of the XM2 compression technology we developed a full software implementation in the C programming language on a SUN workstation. This implementation was acceptable for technology development, but ran at glacial speeds. To prove out the technology from an image quality view an ASIC based decompression and print interface card was developed. In this mode, segmentation and compression was done in slow software and decompression and printing in hardware. This allows us to drive early prototype print engines and validate both image quality and complete some of the system integration. The ASIC decompression board, while very effective and well designed, was also very inflexible when it came to rapidly adapting to new requirements. Changes to ASICS can involve a year or more in turn around. We were also embarking on an ASIC design for the compression part of the system that had a similar design cycle.

While we had often looked at specialized processors for implementing image path technologies such as Digital Signal Processors (DSP), we found them lacking in performance. There is not sufficient computational power and parallelism in the architectures to achieve good cost/performance with a limited number of accelerator cards. In particular, accelerator cards would have to be added on the PCI bus in motherboards or workstations and there is a limited number of PCI slots in most configurations. In addition, most DSPs have to be programmed in assembly language to get the full value of the architecture that would reduce our productivity significantly.

At the time we were part way through the design of the ASIC based compression board we started to see anther type of specialized processor emerge: media processors. Media processors were like DSPs in that they offered a highly parallel internal architecture with multiple execution units, but were specialized toward having sufficient capability to compress JPEG on the order of 50 mega pixels per second for a single component image. While some of the Pentium class processors are now getting in that space for performance, the Pentium processors consume on the order of 40-70 watts power and can't easily be added to a workstation in the form of an accelerator card. PCI accelerator cards can draw at most 25 watts power in a workstation. In addition, the processor on the workstation is busy processing PDL's and other job management functions such that implementing the compression on that processor would likely reduce the overall performance of the system.

We started benchmarking a Media processor developed by Equator Technologies called at the time MAP-1000 (since then the product has evolved to the BSP-15). This processor was designed for "set top" boxes that would replace the cable box in homes with a programmable processor that would decode digital television and also provide a software upgrade capacity. Unlike the power consumption for a Pentium processor these processors were designed to consume under five watts. This low power consumption enabled PCI accelerator cards with four processors to be quite feasible.

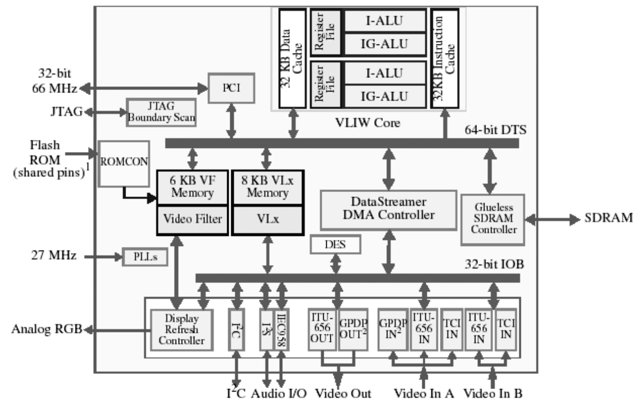

Figure 3. Equator BSP-15 Architecture Block Diagram

To highlight the features that made the Equator BSP-15 media processor different from other processors and ultimately feasible for our application, Figure 3 shows the high level architecture of the processor.[2] While the architecture has many components, there are three major elements that make this architecture unique. First are the two very long instruction word (VLIW) "Core" processor units that between them have four 32 bit arithmetic logic units (ALU)'s and two 128 bit multi media units. The multi media units provide a wealth of single instruction, multiple data (SIMD) complex instructions that allow pixel data at 8 bits per pixel to be processed up to 32 pixels per clock, or even more if the other four 32 bit ALU's are also utilized. A large register set of (128) 32 bit registers, (8) 128 bit special registers, and (32) 1 bit predicate registers adds to the utilization efficiency of the execution units.

The second important feature of the architecture is the Data Streamer. The data streamer is a programmable intelligent data mover that supports 64 independent channels. These channels operate in parallel with other elements of the device and can be used to move data from memory to memory, memory to/from I/O or the PCI bus, and most significantly be used to pre-fill the on chip cache memory. We found the capabilities of this data streamer to be much more significant that any other device we evaluated. In particular, since imaging operations often do not follow typical cache replacement algorithms, the ability to programmatically control the data movement into the cache is a very unique and useful feature. This allows us to move data into cache in parallel with processing of previous data such that when processing on the next data set is started, the data is already in the cache. This avoids the typical cache thrashing that happens in many processor architectures with imaging applications.

The last major function that we found unique is the VLx coprocessor. While the core processor provides high performance in SIMD processing, many algorithms require sequential processing that can't often be implemented in parallel. The VLx stands for Variable Length encoder/decoder (x). The VLx offloads the VLIW core processors from sequential processing tasks such as Huffman encoding. The VLx includes special purpose hardware for bitstream processing, hardware-accelerated MPEG-2 table lookup, and general purpose variable length decoding.

In addition to a unique hardware architecture that is well matched to our image data path applications, one of Equator Technologies key strengths is the complier technology. The C complier Equator provides has it's origin in technology developed at Multiflow Computer, Inc., which pioneered VLIW architecture and Trace Scheduling compiler technologies. This unique complier technology allows us to program in C and optimizes the scheduling of the multiple ALU's in the architecture. Most programmers who have examined the assembly language output of the compiler are both incredibly impressed and at a loss as to how to improve the optimization of the code over what the complier has generated.

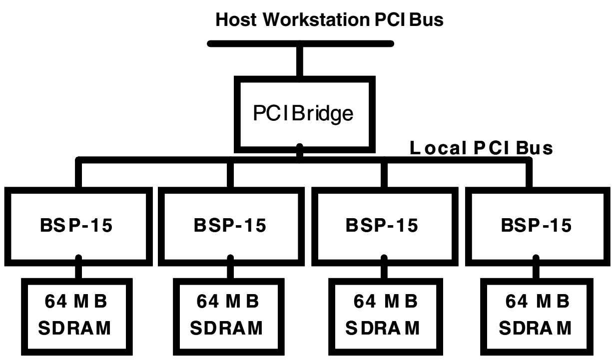

Figure 4. DocuSP Equator BSP-15 PCI Accelerator Card Block Diagram

The low power consumption of this processor ~4 watts @ 405 Mhz allowed us to design a four processor PCI card. The block diagram of this PCI accelerator card is shown in Figure 4. The card consists of a PCI bridge chip and four Equator processors that share a local PCI bus. Each Equator processor has 64 Mbytes of local memory for use as data and program storage. The block diagram also highlights the simplicity of the board design from a component view. While attention has to be paid to high frequency board design standards, the diagram is almost a 1 to 1 mapping of the all the significant devices on the board.

Applications, Performance, Scalability

To utilize the full scalability of the multi processor PCI card, the software has been designed to partition the image up into sections and allow each processor to operate on a section at a time. XM2 compression is designed to be implemented in parallel where independent sections of the image can be segmented and compressed on separate processors. This allows us to scale the performance of the platform quite easily by adding boards to the system. As an example, a single processor can segment and XM2 compress at a rate greater than 50 Mega Pixels per second (single component separations). The Equator architecture allows segmentation and compression to be programmed completely in parallel on a single processor. This includes lossless compression of a 4 bit rendering tag that is associated with each pixel. In this case, a single board can provide compression acceleration for a DocuColor 6060 at 60 pages per minute. Two boards will provide sufficient compression acceleration for iGen3 at 100 pages per minute.

While we have discussed a compression implementation that was the first significant application we required, there have been several other applications that are and will be supported through the software image data path that the Equator processors provide. These will be offered as part of future DocuSP software releases.

As an example of the rapid development that this system provides, Xerox developed an iGen3 configuration with a EFI DFE through use of a specially configured DocuSP. This approach saved having to design a hardware iGen3 interface for the EFI controller and integrate the marking engine specific interfaces with EFI. Instead, a standard, network capable general RTP format was defined that could be efficiently transcoded to the XM2 format using the Equator processors in DocuSP. In this way, we developed in a matter of a few months the code required to demonstrate an EFI connected to an iGen3 print engine with a significantly smaller integration effort. In addition, it is clear that with the original ASIC approach we were planning for the XM2 compression accelerator, new applications such as the EFI integration could not be supported in such a short time frame or with just a few person months of effort.

Conclusion

We have described an approach we developed that capitalizes on a trend in the broadcasting industry with the development of Media Processors such as the Equator BSP-15. By design of algorithms and architectures that can be partitioned in parallel, we have been able to provide scalability by adding processor boards to support higher performance printers or other applications such as the example EFI integration. In addition, this provides a platform for rapidly developing new imaging application support for DocuSP that will be offered in future releases.

References

1. Ian H. Witten, Alistair Moffat, and Timothy C. Bell, Managing Gigabytes: Compressing and Indexing Documents and Images, pg. 48-57, Van Nostrand Reinhold, New York, 1994
2. Equator MAP-CA data sheet, 6/2001, document number HWR.CA.DS.2001.06.20.

Biographies

Norm Zeck is a Technical Manager in the Wilson Center for Research and Technology in the Xerox Innovation Group. He has a BS degree from University of Buffalo and MS degree from University of Rochester, both in electrical engineering. He has worked in the Xerox research center for the past 18 years. In those years he has worked on digital imaging research and technology, in particular, halftoning, image compression, and systems architectures for document printing and scanning.

Bill Nelson is a Principal Scientist in the Wilson Center for Research and Technology in the Xerox Innovation Group. He has a BS degree from Rochester Institute of Technology and MS degree from Rensselaer Polytechnic Institute both in electrical engineering. He has worked in the Xerox research center for the past 17 years. In those years he has worked on digital imaging research and technology, in particular, contour font technology, image compression and high performance printing system design.