Memory Organization of a Single-Chip Video Signal Processing System with Embedded DRAM

Jörg Hilgenstock, Klaus Herrmann, Peter Pirsch

Laboratorium für Informationstechnologie
Universität Hannover, Germany
E-mail: hilgenst@mst.uni-hannover.de

Abstract

A programmable single-chip multiprocessor system for video coding applications has been developed. It integrates four processing elements, on-chip DRAM, and application-specific interfaces. The integrated DRAM is primarily used as frame buffer and makes external memory for most applications obsolete. For fast access to local data segments also static RAM is integrated in each processing element.

1. Introduction

Video coding is a key feature of many different multimedia applications. Several different standards have been adopted for applications with different requirements in terms of image quality, bandwidth, and latency, like video telephony [1] or digital television [2]. These standards utilize hybrid coding techniques [3]. They combine computation intensive low level tasks, like motion estimation or discrete cosine transform, with data dependent medium level tasks, like variable length coding, variable threshold or quantization.

To provide flexibility for processing of different video coding schemes a programmable architecture is a possible solution. But most programmable general-purpose DSPs do not provide sufficient processing power for video coding or do have modules which cannot be utilized for video coding. Therefore, the programmable video signal processing architecture AxPe [4] has been developed and implemented. It is a coprocessing architecture which is optimized for hybrid video coding and provides enough processing power for video telephony applications. For more demanding applications it can be used as processing element (PE) in a homogeneous multiprocessor system [5].

To guarantee a high degree of utilization of the processing resources the memory bandwidth is an important issue in this multiprocessor architecture. To achieve a fast data access, frame memory can be integrated in the processor. Modern semiconductor processes enable monolithic implementation of high speed standard cell logic as well as dense DRAM. With on-chip memory a lower latency access can be achieved and wider buses are possible than to external RAM devices. Furthermore embedded DRAM leads to a smaller system size and cost and decreases the power consumption, which is especially advantageous for mobile applications.

In the following section a monolithic video signal processing system with embedded DRAM and application specific interfaces is introduced. Section 3 describes the organization of the integrated memories and section 4 presents how this architecture is used in video coding applications. The implementation of the chip is described in section 5.

2. System Overview

The computational requirements for a video encoder/decoder according to MPEG4 [6] (simple profile @ CCIR601 resolution) can be fulfilled by four processing elements AxPe. For a complete system a frame buffer and interfaces for video and coded data are required additionally. To fulfill these needs a monolithic video signal processing system has been developed. It consists of four AxPe, integrated memory and application specific interfaces. The memory is divided into four parts and is distributed among the four processing elements, so that the system consists of four identical subsystems (Figure 1). The segmentation of the memory allows to connect each processing element with a private bus to the memory in order to increase the overall memory bandwidth and to simplify arbitration of memory requests. A drawback of this solution is that communication between the subsystems via the memory is not possible like in a shared memory approach. Nevertheless the subsystems can exchange data via a dedicated interface. Furthermore video and host interfaces are integrated to allow direct connection of the device to external standard hardware.

0-7695-0104-4/99 $10.00

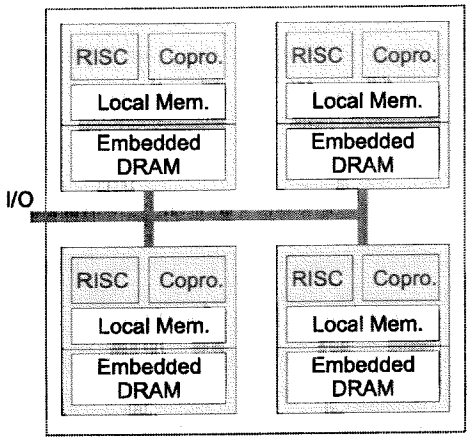

Figure 1. Monolithic multiprocessor system consisting of four subsystems

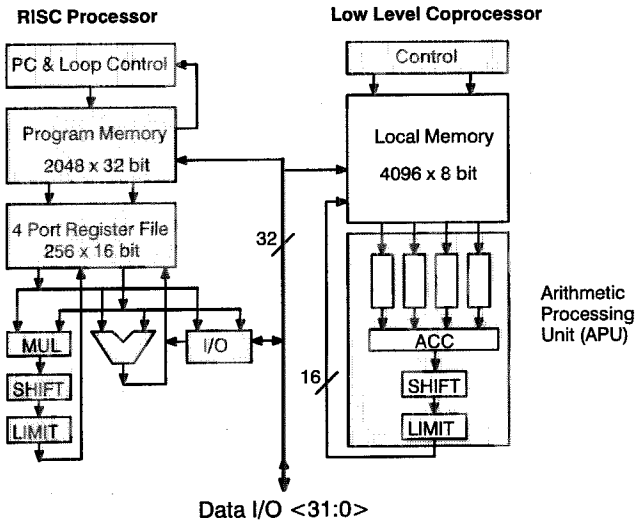

Figure 2. Architecture of the processing element AxPe

2.1. Video Processing Element AxPe

To achieve good efficiency in video coding applications, a processing architecture must be adapted to the envisaged algorithms. Hybrid video coding schemes consume most of the processing power in deterministic convolution-type low level tasks, but also employ medium and high level operations. Based on this characteristic the coprocessing architecture AxPe has been developed. It consists of a RISC core and a coprocessor. The RISC processor is used for computation of the medium level tasks and performs several control tasks. For fast processing of convolution-type low level tasks the microprogrammable coprocessor is used. This architecture has been already implemented in the video signal processor AxPe1280V [4] and will be used as building block in the presented multiprocessor system.

The structure of the architecture is shown in figure 2. The RISC processor consists of a programmable ALU, a multiply/shift/limit unit, and an integrated program memory. At 100 MHz a peak performance of 100 MIPS (Mega Instructions per Second) is achieved by the RISC.

The coprocessor features a fourfold parallel ALU/multiply pipeline in combination with a common multioperand accumulator and a shifter/limiter. These four datapaths of the coprocessor provide a peak performance of 1.6 GOPS (Giga Arithmetic Operations per Second) at 100 MHz. The operation of the coprocessor as well as all I/O transfers are controlled by the RISC.

3. Embedded Memory

To reduce the number of I/O cycles for loading and storing of video data, each subsystem is equipped with two on-chip data memories (Figure 3). The first level memory is the local memory, which can directly be accessed by the RISC and the datapaths of the coprocessor. It is build of static RAM. The second level memory is primarily used as frame memory and consists of dynamic RAM. This two level memory hierarchy has been chosen, because it provides a high bandwidth especially for block-based video processing. After loading an image block (e.g. 16*16 pixels) for processing into the local memory the processing element has a high bandwidth and low latency access to the data. This reduces the number of accesses to the DRAM significantly, so that the DRAM can also serve access requests from the video or communication interface.

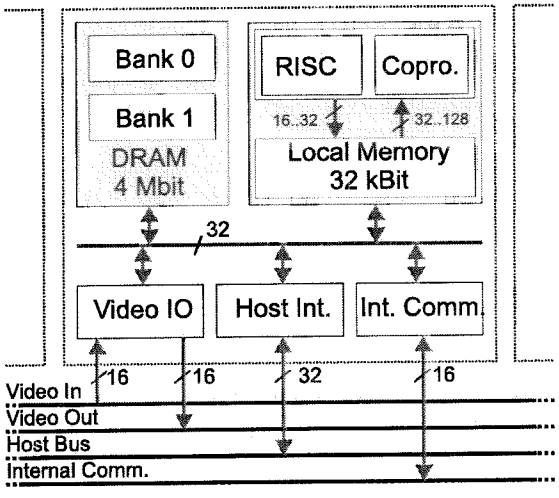

Figure 3. Subsystem

3.1. Local Memory

The local memory supports on-chip storage of input data and intermediate results of the arithmetic processing unit. Furthermore, the local memory serves as a 32 bit memory mapped interface between the coprocessor, the RISC, and the embedded DRAM.

The coprocessor has the highest bandwidth requirements. To feed the parallel datapaths with operands it needs read access to 128 bit each clock cycle, while simultaneously writing a result of 16 bit to the local memory each clock cycle. All other transfers need a bandwidth of 32 bit per clock cycle.

The local memory was realized as dual-port static RAM. With integration of this 4096 byte local memory, parts of the processed video image can be kept locally. This is especially advantageous for motion estimation in hybrid coding applications, as it allows to carry out block matching within a search range of up to +/- 22 pixels horizontally and vertically without reloading from the DRAM.

3.2. Embedded DRAM

Although each processing element AxPe consists of integrated static RAM, an additional frame memory is necessary to provide the required storage capacity for most applications and to operate as frame buffer. Therefore an embedded DRAM of 4 MBit is integrated in each subsystem.

The DRAM in each subsystem is divided in two banks for page interleaving and is able to transfer one 32 bit word per clock cycle at a latency of 4 clock cycles.

Loading and storing operations between local memory and DRAM are controlled by a DMA controller. It supports a linear or block addressing scheme. A data transfer is initiated manually by a software command. Simulations have shown, that a transparent and autonomous cache controlling would not achieve the performance of optimized software with explicit load/store commands.

3.3. Interfaces

The video interface of each subsystem transfers video data between external devices and the embedded DRAM. It operates autonomously from the processing element after initialization with a set of parameters by software. These parameters describe the region of the frame, which is grabbed by each subsystem, so that each PE can process a different region of a larger frame. Also the DRAM start address for these video transfers is defined by parameters, which can be changed for each new frame.

Furthermore, a 32 bit host bus is integrated for transfer of coded video data and status information. The data exchange between the four subsystems is carried out by a communication interface, which employs a distributed arbitration scheme.

4. Video Processing

With the 16 MBit DRAM integrated in the multiprocessor system, three frames in CCIR601 resolution and 4:2:0 representation can be stored on-chip. To allow parallel processing of frames, each frame is divided in four slices for the four subsystems. Each subsystem can then process most of the time independently on locally stored data. For motion estimation neighboring parts of the stored image must be accessed from a different subsystem. In this case the communication interface sends a request to the subsystem, where the required data is stored, and the image data is transferred over the communication bus.

Video coding algorithms, like MPEG2/4 (simple profile), require the storage of the frame, which is currently processed, and one reference frame for motion estimation. If the system is used for video decoding this reference frame is the previous frame and identical to the frame which is read out by the video interface for display. Therefore for video decoding the storage of two frames is necessary (Figure 4).

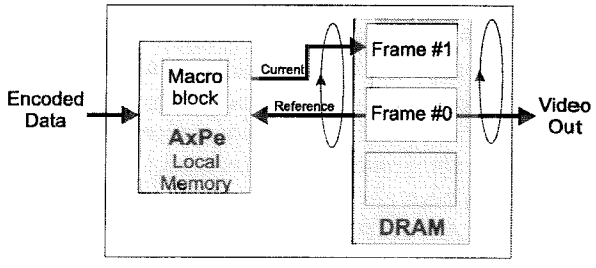

Figure 4. Memory organization for video decoding

Video encoding requires memory for storage of three frames (Figure 5). The reference frame is reconstructed by the encoder and replaces the input frame, which is currently processed, in the memory. An additional frame memory is needed for the previous reference frame and another one for the frame which is written into the memory by the video interface. If this double buffering of the input video data is not used and the data is directly written into the memory range processed by the PE, only enough memory for two frames is needed and the processing latency is reduced by one frame. On the other hand this mode requires that the processing element has to stop access to the video buffer while the video interface exchanges data with the DRAM. Thus, using only a single video input buffer leads to a de-

crease of the available time for processing by 23% compared with double buffering.

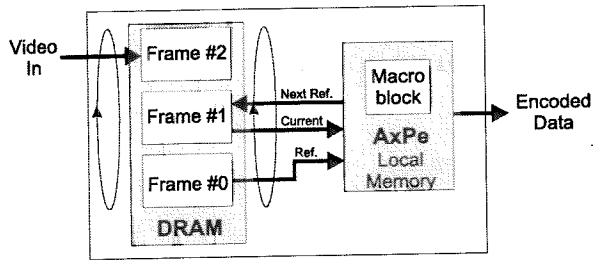

Figure 5. Memory organization for video encoding

5. Implementation

The proposed multiprocessor system is currently implemented in a 0.25 μm technology. To integrate the required memory an DRAM optimized process has been chosen, where each memory cell is built of one transistor with a single trench capacity.

Single components of the system, like the AxPe or the DRAM memory with DMA control logic, have been already implemented and verified in test chips. Based on the experience with these chips the silicon area of the final implementation in 0.25 μm technology could be ascertained (table 1).

Module	Subsystem	Multi-processor
AxPe & Local SRAM	16 mm^2	
Interfaces	5 mm^2	4 *
DMA Control	2.5 mm^2	47.5 mm^2
Emb. DRAM & Control	24 mm^2	
Pads	—	24 mm^2
Silicon Area	47.5 mm^2	214 mm^2

Table 1. Silicon area

Because the overall silicon area is dominated by the size of the embedded DRAM, the yield of the device is mainly affected by DRAM defects. To increase the yield the DRAM employs redundancy techniques in combination with a self reconfiguration logic [7].

6. Conclusion

A programmable and monolithic video signal processing system has been developed. It consists of four processing elements and 16 MBit embedded DRAM as frame memory, which is distributed among the processors. An internal communication structure allows sharing of frame memory contents between processors. This feature enables parallel processing of video compression algorithms.

The system provides enough processing power and memory to support video encoding and decoding according to MPEG2 (simple profile @ main level) or MPEG4 (simple profile, CCIR601 resolution). Additionally proprietary applications can be implemented, as the processing is controlled by software and all integrated modules are highly configurable.

Acknowledgments

The authors would like to thank Origin ASIC Centre, Hamburg, for the design of the full custom modules and the final layout of the chip. This work is supported by FhG under contract no.T/F41B/T0183/P1307.

References

[1] Draft ITU-T Recommendation H.263, "Video Coding for Low Bit Rate Communication", International Telecommunication Union, Dec. 1995.

[2] ISO/IEC JTC1/SC29/WG11 CD 13818-2, 1994.

[3] H.G. Musmann, P. Pirsch, H.-J. Grallert, "Advances in Picture Coding", Proc. IEEE, Vol. 73, No. 4, pp. 523-548, 1985.

[4] J. Hilgenstock, K. Herrmann, J. Otterstedt, D. Niggemeyer, P. Pirsch, "A Video Signal Processor for MIMD Multiprocessing", Design Automation Conference (DAC) 1998, pp. 50-55, June 1998.

[5] J. Otterstedt, K. Gaedke, K. Herrmann, M. Kuboschek, H.U. Schroeder, A. Werner, " A 16 cm^2 Monolithic Multiprocessor System Integrating 9 Video Signal-Processing Elements", IEEE International Solid-State Circuits Conference, Vol. 39, pp. 306-307, Feb. 1996.

[6] Coding of Moving Pictures and Audio, Overview of MPEG-4 Profiles and Levels, ISO/IEC JTC1/SC29/WG11, MPEG98/N2325 Jul. 1998.

[7] D. Niggemeyer, J. Otterstedt, M. Redeker, "A Defect-Tolerant DRAM employing a Hierarchical Redundancy Scheme, Built-In Self-Test and Self-Reconfiguration", IEEE International Workshop on Memory Technology, Design and Testing, pp. 33-40, Aug. 1997.