Implementation and Optimization of software MPEG-2 decoder for CELL REGZA

HsiaoChan Wu, Ryuji Sakai
hsiaochan.wu@toshiba.co.jp ryuji.sakai@toshiba.co.jp
Core Technology Center, TOSHIBA Corporation

1. Introduction

CELL REGZA™, equipped with Cell Broadband Engine™ * (Cell BE), is the latest Digital TV developed by Toshiba. On CELL REGZA, we realized various features which are never implemented on Digital TV before. Such as the 8-channel display, the high speed channel switching, etc. These features heavily rely on MPEG-2 decoder and on large amount of data transmission between CPU and memory, therefore, how to implement the MPEG-2 decoder and how to utilize the bandwidth between CPU and memory become important issues on CELL REGZA.

In the first part of this paper, we describe some architecture characteristics of Cell BE and then, we focus on discussing the issues of implementing MPEG-2 decoder on Cell BE and the method we used to utilize the performance of Cell BE. In the last part, we demonstrate the evaluation result.

2. Background

Cell BE is a hetero-generous multi-core processor(Fig. 1) including 1 PowerPC Processor Element (PPE) which is designed to achieve balanced performance on a wide range of applications, and 8 Synergistic Processor Elements (SPE) which are designed to achieve outstanding computation performance. However, without appropriate strategies dealing with the architectural features on Cell BE, the maximum computation performance would not be exploited.

Unlike PPE, SPE can only access directly to its on chip 256KB local storage. For accessing main memory, SPE has to issue DMA access command to Memory Flow Controller (MFC) so that MFC can read from main memory to the local storage or write back to main memory from local storage. Besides, DMA access is restricted to 128-byte alignment that means starting address of DMA read/write needs to be aligned to 128-byte boundary. For implementing MPEG-2 decoder on SPE, there are two issues we have to deal with:

I. Bandwidth limitation:
On CELL REGZA, there are various applications running simultaneously on SPEs and each of them occupies the memory bandwidth. CELL REGZA may execute up to 8 MPEG-2 decoders concurrently, therefore, to minimize the usage of bandwidth becomes an important issue.

II. Latency:
With the memory access restriction of SPE, DMA latency happens whenever a DMA command is issued. If it is not well organized, DMA latency will be the bottleneck of performance. Besides, by eliminating DMA latency, processing time can be shortened so that the SPE resource can be scheduled for other applications.

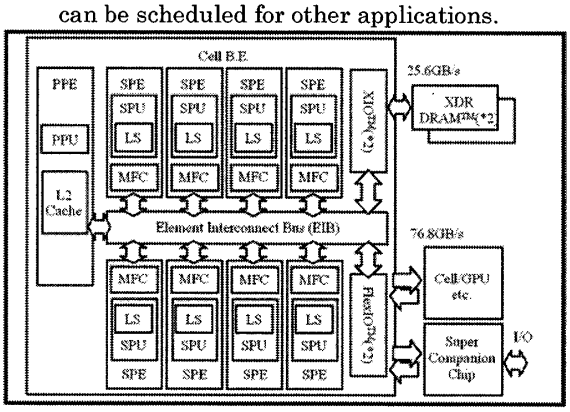

Fig. 1: Block diagram of Cell B.E.

3. Implementation and Optimization

Among all MPEG-2 decoding processes, Motion Compensation (MC) needs lots of memory access between main memory and SPE local storage. To deal with latency issue, we modified the scan order, named "16 Byte Return scan order" (16R scan order), for decoded pictures and used "DMA Queue" in order to reduce the memory access latency while processing MC.

Generally, picture data is arranged in raster scan order by which pixels are stored in a continuous memory space from left to right, top to bottom. However, in MPEG-2, pictures are decoded by Macroblock (MB) which is a 16x16 pixel area of a picture, and when writing back the decoded MB to main memory in raster scan order, SPE needs to issue 32 DMA commands for Y, U and V blocks because rows of MB are not continuous in memory space with arranging in raster scan order. Although DMA commands can be reduced by list DMA command of MFC, the overhead of maintaining the list DMA data structure still takes certain SPE time. On the other hand, by 16R scan order, SPE can read needed reference area by at most 4 DMA issues and write back decoded MB by 2 DMA commands so that the number of DMA commands can be reduced. Moreover, using 16R scan order can also minimize the amount of transferred data while MC.

The other method we used to reduce memory

* Cell Broadband Engine is a trademark of Sony Computer Entertainment Inc.

access latency is "DMA Queue". In MC of MPEG-2, decoder forms prediction from previously decoded pictures. These processes are: decoder reads needed area of previous picture, forms prediction and combines with coefficient data, and then writes back final decoded MB to memory. If we do above steps sequentially, decoding process would become inefficient. By the memory architecture of Cell BE, DMA access and SPE computation can be overlapped. DMA Queue is designed to overlap the SPE computation time and DMA access time. (Fig.2)

In MC, we create 2 DMA Queue nodes, 1 for reading and 1 for writing, for each MB. Each node works as following steps: (1) Decode motion vector from bit stream, (2) Set reading memory addresses and callback function after DMA finished, and (3) Enqueue and starts DMA. When the queue is full, the head of the DMA Queue is dequeued, and invokes its callback function to combine the prediction with coefficient data. After finishing callback function, the write queue node is triggered and recycled into empty queue. By this method, we can overlap the calculation time with DMA access time.

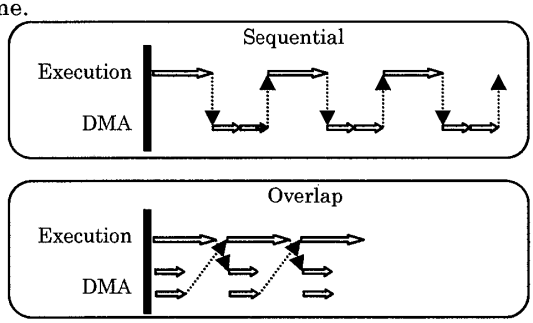

Fig. 2: Sequential and Overlap execution

Besides those methods mentioned above, we also implemented the MPEG-2 decoder with other optimization methods such as exploiting SIMD and raising IPC (Instruction Per Cycle) by pairing adequate instructions so that SPE can execute them simultaneously, and those methods also significantly contributed to the performance of our MPEG-2 decoder.

4. Evaluation

In this section, we will evaluate the performance of the MPEG-2 decoder in the following aspects: DMA latency, bandwidth usage and decoding time.

First of all, we evaluated DMA stall time with different DMA Queue sizes while decoding a territorial broadcast stream (1440x1080, 15Mbps approx.). For each DMA Queue size, we measured the proportion among DMA stall, dual issue and single issue with 1 second of execution. Measuring DMA stall time within certain interval can indicate that how many SPE time is wasted due to memory access latency. As showed in Fig.3, DMA stall took about 35% of total SPE time when DMA Queue size is 1. After we increased the DMA Queue size to 2, 4, 8 and 16, the ratio of DMA stall became lower and the single issue and dual issue rate became relatively higher. This result showed that using DMA Queue can indeed reduce the memory access latency by overlapping execution and DMA access.

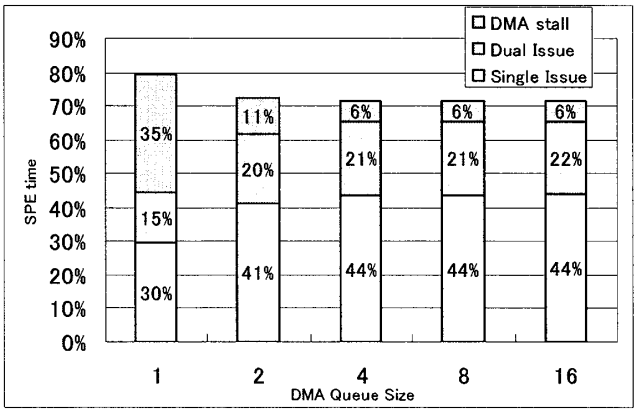

Fig. 3: DMA stall ratio

Fig. 4 illustrates the needed SPE time for real-time decoding. As we increase the number of decoding streams to 12, the memory bandwidth becomes congested and the decoding time increases. Because of the heavy usage of the bandwidth, the performance dropped while decoding 8 or 12 streams concurrently, but the result still shows that the decoder is capable of decoding 8 territorial broadcast streams currently with less than one third of total SPE resources.

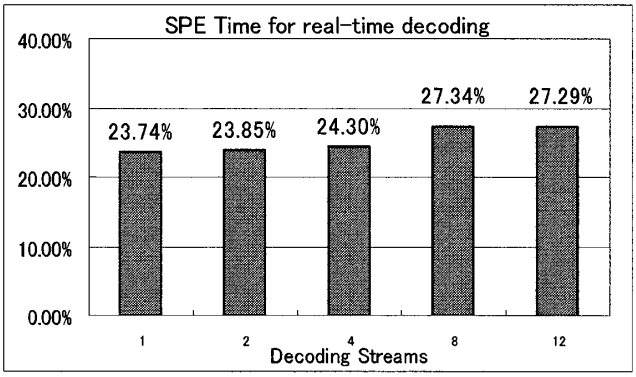

Fig.4: SPE time for real-time decoding

5. Conclusion

By the approaches proposed in this paper, we implemented a highly efficient MPEG-2 software decoder that enables CELL REGZA to realize various features which have never appeared on Digital TV before.

Reference

[1] ISO/IEC 13818-2 1995(E)
[2] Cell Broadband Engine Programming Handbook V1.1, IBM