Multigrain Parallel Processing for JPEG Encoding on a Single Chip Multiprocessor

Takeshi Kodaka, Keiji Kimura and Hironori Kasahara

Dept. of Electrical, Electronics and Computer Engineering, Waseda University.
Okubo, Shinjuku-ku, Tokyo, Japan, 169-8555, TEL: +81-3-5286-3371
URL: http://www.kasahara.elec.waseda.ac.jp/

1 Introduction

With the recent increase of multimedia contents using JPEG and MPEG, low cost, low power consumption and high performance processors for the multimedia contents have been expected. Particularly, single chip multiprocessor architecture having simple processor cores is attracting much attention to develop such processors. This paper describes a multigrain parallel processing scheme for a JPEG encoding program on OSCAR type single chip multiprocessor and its performance.

2 Multigrain Parallel Processing[1]

OSCAR multigrain parallel processing compiler uses coarse grain task parallelism among loops, subroutines and basic blocks, loop parallelism among loop iterations and near fine grain parallelism among statement. In the coarse grain task parallel processing, a source sequential program is decomposed into three kinds of coarse grain tasks, or macrotasks (MTs), such as Repetition Block (RB), Subroutine Block (SB), and Block of Pseudo Assignment statements (BPA). After generation of macrotasks, the compiler assigns MTs onto processor-groups (PGs). If a MT assigned to a PG is a Doall loop, the MT is processed in the iteration level grain by processing elements (PEs) inside a PG. If a MT assigned to a PG is a BPA, it is decomposed into statement level near fine grain tasks and processed in parallel by PEs inside a PG.

3 JPEG Encoding using Multigrain Parallel Processing

This section describes JPEG Encoding algorithm applied multigrain parallel processing. The JPEG encoding algorithm used here is that of "jpeg-v6a" in MediaBench[2]. This JPEG encoding consists of the following six stages; 1. Decomposing the input image into 8x8 blocks, 2. Converting image to YCbCr format (YCbCr), 3. Performing the two-dimensional DCT on each block (DCT), 4. Quantizing each DCT coefficient uniformly (Quant), 5. Subtracting the quantized DC coefficient from the corresponding term in the previous block (1-D DC prediction), 6. Entropy coding the quantized coefficients using variable length codes (VLCs). These stages are continued until input data ends.

Basically, there is no data dependence among 8x8 blocks in this JPEG encoding algorithm. 1-D DC prediction stage is the only one that has data dependencies among 8x8 blocks. To use these large parallelism efficiently, 1-D DC prediction stage and DC entropy stage are put after other stages as shown in Figure 1(a). In a multigrain version of JPEG encoding program, each 8x8 block is defined as a basic processing unit and some groups of encoding stages that processes these units are defined as the macrotask (MT). In this case, YCbCr, DCT, Quant and AC Entropy Coding (AC EC) are grouped and defined as a MT (MT_a). Similarly, 1-D DC Prediction and DC Entropy Coding (DC EC) are grouped and defined as a MT (MT_b). These MTs to execute the same 8x8 block are assigned statically to the same PG considering minimization of data transfer. For instance, as shown in Figure 1(b), in the case of 2PGs, each PG has two MT_a and two MT_b respectively. At first, PG1 receives two 8x8 blocks, block1 and block2, and these blocks are processed by MT_{a1} and MT_{a2}. Similarly PG2 receives block3 and block4 and processes MT_{a3} and MT_{a4}. After finishing MT_{a2} operation in PG1, MT_{a2} in PG1 sends data needed by MT_{b3} in PG2. Then MT_{b1} and MT_{b2} are processed in PG1 and MT_{b3} and MT_{b4} are processed in PG2.

In addition to parallelism among 8x8 blocks, inside MT_a and MT_b have near fine grain parallelism[3]. These parallelism are also used by each PE inside PG. These near fine grain tasks are scheduled by the compiler that using four heuristic scheduling, CP/DT/MISF, CP/ETF/MISF, ETF/CP and DT/CP. Also the compiler chooses the best schedule automatically and generates parallel machine code for each PE. However, execution cost of entropy coding operation depends on input data. So, scheduling for entropy coding was performed using execution profile by hand.

In this paper, hierarchical multiple grains of par-

allelism, such as coarse grain parallelism among 8x8 blocks and near fine grain parallelism inside 8x8 blocks, are exploited hierarchically.

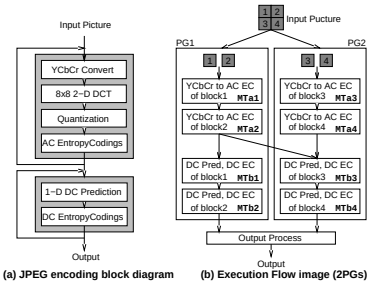

Figure 1: Parallel JPEG execution flow

4 OSCAR Type Single Chip Multiprocessor Architecture

OSCAR type single chip multiprocessor (SCM) is described here. In this evaluation, clock level detailed simulator was used.

OSCAR type SCM architecture[4] is shown in Figure 2. In this SCM architecture, each processing element (PE) has CPU, local program memory (LPM), local data memory (LDM), distributed shared memory (DSM) having two ports and data transfer unit (DTU). DTU is used for overlapping of data transfer and computation by compiler control, though this function is not used in this paper. Three buses connect these PEs in this evaluation though other interconnections like crossbar can be also used. Furthermore, this SCM has centralized shared memory (CSM) outside a chip. Processor core inside the chip is based on pipeline configuration of UltraSPARC-II architecture. In this evaluation, OSCAR type SCM has simple in-order-issue processor core that has one integer execution unit (IEU), one load-store unit (LSU), one floating-point unit (FPU) of function units and issue width is one instruction per clock.

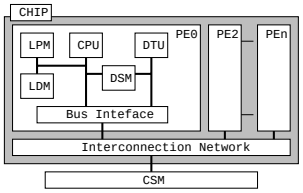

Figure 2: OSCAR type SCM architecture

5 Performance Evaluation

This section evaluates performance of multigrain parallel processing of JPEG encoding program.

The speed-up against sequential execution time on OSCAR type SCM is shown in Figure 3. mPGnPE in the Figure 3 means that the SCM has m processor-groups (PGs) for coarse grain processing and each PG has n processor-elements (PE) for near fine grain processing, and US-II means SCM has a four-issue superscalar processor core that is similar to UltraSPARC-II. When total number of processors is two, 2PG1PE gives us 1.50 times speed-up and 1PG2PE gives us 1.62 times speed-up respectively. When total number of processors is four, 4PG1PE gives us 3.36 times speed-up and 2PG2PE gives us 3.59 times speed-up respectively. In the same numbers of processors, using multigrain parallel processing gives us the better performance since multigrain parallel processing gives us good load balance both in coarse and near fine grain level. Furthermore, SCM that has a four-issue superscalar processor core gives us only 1.25 times speed-up. This result shows that OSCAR type SCM which has simple processor cores architecture contribute to scalable performance improvement.

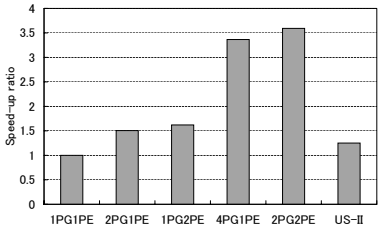

Figure 3: Evaluation result of JPEG encoding

6 Conclusions

This paper has proposed the multigrain parallel processing scheme for JPEG encoding on a single chip multiprocessor (SCM). The performance evaluation showed that OSCAR type SCM having four single-issue processor cores gave us 3.59 times speed-up against sequential execution though speed-up by a four-issue superscalar processor gave us only 1.25 times speedup.

Acknowledgments

This research has been supported by STARC. The authors thanks to Mr. Hirata (STARC), Mr. Miyata (STARC), Mr. Asano (Toshiba), Mr. Yasukawa (Toshiba), Mr. Takahashi (Fujitsu), Mr. Kurata (Sony) and Mr. Takayama (Matsushita) who are STARC project members and gave invaluable comments.

References

[1] H. Kasahara, M. Obata, and K. Ishizaka. Automatic coarse grain task parallel processing on smp using openmp. *Proc. of 13th International Workshop on Languages and Compilers for Parallel Computing (LCPC'00)*, Aug. 2000.

[2] C. Lee, M. Potkonjak, and W. H. Mangione-Smith. Mediabench: A tool for evaluating and synthesizing multimedia and communications systems. *30th International Symposium on Microarchitecture (MICRO-30)*, Nov. 1997.

[3] T. Kodaka, N. Miyashita, K. Kimura, and H. Kasahara. Near fine grain parallel processing on multimedia application for single chip multiprocessor. ARC2001-140-11, IPSJ, Jul. 2001.

[4] K. Kimura, T. Kato, and H. Kasahara. Evaluation of processor core architecture for single chip multiprocessor with near fine grain parallel processing. *Trans. of IPSJ*, 42(4), Apr. 2001.