High Performance Application Specific Architectures

D. Abramson

Division of Information Technology,
C.S.I.R.O.
723 Swanston St
Carlton, 3053

Department of Computer Systems Engineering
Royal Melbourne Institute of Technology
P.O. Box 2476V, Melbourne 3001, Australia.

Contact Information: davida@mel.dit.csiro.au

Application Specific Architectures have enormous potential for high performance in certain selected areas. There are many examples where they exceed the performance of general purpose computers. However, in times when micro-processors are becoming as fast as traditional supercomputers, the role of such machines needs re-examination. This mini-track of the 26th Hawaii International Conference on System Sciences, concerns High Performance Application Specific Architectures. As an introduction to the mini track we will examine a few issues related to the successful implementation of such machines. We conclude that under appropriate conditions such machines will continue to be more than competitive.

1 Why are modern micro-processors high performance?

Modern microprocessors gain their super computer performance from four main areas; through advances in micro-electronics technology, multiple functional units, superscalar instruction issue and high speed memories in caches and register files.

1.1 Technology

Advances in micro electronic technology have allowed dramatic improvement in the performance of micro-processors. The small design rules used in modern chips allow large amounts of high speed logic to be packed into one device, removing delays introduced by input/output buffers. Most high speed processor chips now incorporate integer and floating point arithmetic units, large register files, instruction and data caches and complex issue logic on the one device. However, technology alone cannot provide unlimited performance gains, and architectural techniques which incorporate concurrency are necessary.

1.2 Multiple Functional Units

Having more than one functional unit allows the processor to execute more than one instruction concurrently. Modern microprocessors like the DEC Alpha and the Intel i860 utilise multiple integer and floating point arithmetic units, thus providing significant overlap of instructions.

1.3 Superscalar design

Superscalar instruction issue logic allows the processor to maintain high efficiency in the use of multiple function units. Modern micro processors allow up to two or three instructions to be issued concurrently, providing there are sufficient resources available and there is sufficient parallelism in the program.

1.4 Cache Memories & Registers

Very high speed memories are required in order to allow the processor to issue and execute multiple instructions concurrently. Cache memories, and large windowed register files, provide the illusion of large amounts of high speed memory providing the program exhibits sufficient temporal and spatial locality.

2 Performance in Application Specific Architectures

The architectural features described above allow enormous performance gains in general purpose computers. However, by their nature such machines must support many algorithms and thus cannot increase the performance of one algorithm at the expense of another. There are a number of areas where special purpose designs have advantages, namely by using massive concurrency, using dedicated data pathways, providing dedicated address generation hardware, making use of high speed

0-8186-1060-3425/93 $03.00 © 1993 IEEE

microprocessors as building blocks and utilising high speed interconnections.

2.1 Massive Concurrency

Many algorithms have more concurrency than can be exploited by using a general purpose processor. For example, 512 by 512 pixel images contain 256k pixels, which can often be processed simultaneously. Further, the operations may be much simpler than those in general purpose programs, thus much of the power of the arithmetic units is wasted. To use the image processing example again, most pixels can be represented using one byte integers, thus a 64 bit floating point unit is not required; further many 8 bit integer units can be built using the same area of silicon as one 64 bit floating point unit.

2.2 Dedicated Data Pathways

In a general purpose computer the data flow of a program is expressed by the configuration of instructions and the way that they move data from one memory cell to another. Often this must be performed through general purpose registers and the cache. General purpose data alignment hardware and barrel shifters are also on the path of the data, thus limiting the performance of the code. In an application specific architecture the algorithm can be mapped onto the correct amount of hardware, thus speeding information flow along the critical path. A side effect is that less hardware is required making the data pathways faster and smaller than in a general purpose computer.

2.3 Address Computation

A significant portion of the execution time of programs which address multi-dimensional arrays, is devoted to address computation. Even optimising compilers cannot remove all address calculations, which in the worst case require multiple additions and multiplications. Further, if more than one array is being addressed then separate computations must be performed. Even though they are often data independent these computations are usually serialised due to lack of functional units. Special purpose computers can provide multiple parallel address computation units to meet the peak demand required by the algorithm. Further, by scaling address ranges to powers of 2, the additions, multiplications and shifting operations performed by general purpose computers can be totally removed. Addresses can then be delivered directly to dedicated memories.

2.4 Dedicated Control

Many algorithms that require high performance are actually quite simple. Image processing operators utilise a number of very simple arithmetic functions applied iteratively. Similarly, simulated annealing algorithms usually require a number of simple operations like cost computation and exponential table lookup to be repeated iteratively. These algorithms benefit from simple, high speed, dedicated control units. Programmable logic for state machine control, or even microcoded control, can provide a low cost control mechanism without the need for instruction caches and buffers, decode logic and prefetching hardware.

2.5 Micro-processors as building blocks

If a special purpose architecture is sufficiently complex then it can benefit from being constructed from general purpose micro-processors as the building blocks. This configuration gives the advantages of both dedicated architectures and general purpose multiprocessors. However, in order for the technique to be effective, the algorithm should make significant use of the functionality of the micro-processor. Applications which benefit from this approach have a lot of both large and fine-grain concurrency, and are usually floating point intensive.

2.6 Special Purpose Interconnect

Algorithms which are highly concurrent, but have a special structure, can often benefit from a special purpose machine built of general purpose micro-processors which are connected by a dedicated interconnect. This interconnect has the potential to be much cheaper than one provided by a general purpose multiprocessor because it only provides bandwidth along paths which require it. Also, issues such as cache coherence may be avoided by considering the structure and operation of the algorithm, simplifying the hardware further.

3. Comparative Issues

3.1 Price Performance

General purpose computers benefit from enormous production quantities, and thus have very low price/performance ratios. Far fewer special purpose computers are produced, and thus must have very high performance to achieve the same price/performance ratio. Further, that price performance ratio must be lower for special purpose architectures if they are to be commercially viable because advances in general purpose computers are proceeding at such a rapid rate. By considering the current rate of advancement in general purpose micro-processors, it would appear that a special purpose architecture would need a price/performance ratio initially of the order of 100 times less than that of the general purpose hardware before it would be commercially competitive. This ratio is composed of two independent factors of 10; one which provides the price/performance decrease necessary to consider special purpose architectures in the first place, and one which helps maintain a lead over general purpose machines. Thus, a special purpose architecture which starts off being 100 times more competitive than a general purpose one , will still be 10 times more competitive at the end of its product cycle.

3.2 Technology curves

Typically a micro-processor manufacturer will invest leading edge technology in the latest range of chips, because of the huge design effort required to produce a complete system. Such technology is usually not available for designers of special purpose architectures because of the lower production numbers. Thus, the price/performance ratio mentioned in the previous section must often be achieved using technology which is not state of the art. Further, if the special purpose architecture is to remain competitive, it must use upgraded technology at the same rate as the general purpose parts.

3.3 FPGAs and Design Systems

Special purpose computers are usually much simpler to build than general purpose ones, however, a new machine must be designed for each application. Advances in Field Programmable Gate Arrays (FPGAs) offer great potential for special purpose architectures because they allow the architecture to be altered without hardware changes. Further, advances in high level hardware description techniques should allow special purpose architectures to be produced easier than with current design tools. Because FPGAs can be produced in fairly large quantities, they can allow special purpose architectures to track the same technology curves as general purpose ones.

3.4 Summary

It can be seen that given a few criteria, application specific architectures can be extremely successful at solving problems. First, they must provide a lower price/performance ratio than conventional general purpose computers, even if they utilise a slower technology. Second, they should employ various architectural techniques which are not possible in general purpose processors. Finally, FPGAs and advanced design systems must be developed to allow easy prototyping of new architectures.

4. This Mini-track

This is the second year that the mini-track has been run. This year 9 papers were selected out of the 15 that were submitted and they cover a range of topics. In the table overleaf is an overview based on the criteria discussed in section 3, which shows the various architectural features of special purpose architectures that were used.

Acknowledgement

The work described in this introduction was conducted under the joint CSIRO/RMIT Program in High Performance Computation. Thanks go to the members of the program team. In particular, thanks to Rhys Francis for editing a draft of this introduction. Thanks also go to the researchers who submitted work to the mini-track.

	Paper	Architectural Feature Exploited					
		Massive Concurrency	Dedicated Data Pathways	Dedicated Address Comp.	Dedicated Control	Micro-processor Building Blocks	Special Purpose Interconn ect
1	A Parallel Image Processing and Display System (PIPADS): Hardware Architecture and Control	✓	✓	✓	✓	✓	✓
2	STORM: A Multibus System for Climate Modelling	✓	✓	✓		✓	✓
3	Optimal Mapping of DSP Applications to Architectures	✓	✓	✓	✓		✓
4	A Special Purpose Computer for Particle Dynamics Simulation Based on the Ewald Method: WINE-1	✓	✓	✓	✓		✓
5	GRAPE: Special Purpose Computer for Simluations of Many-Body Systems	✓	✓	✓	✓		✓
6	On the Projections of Affine Recurrence Equations	✓	✓		✓		✓
7	A Distributed-Memory Multi-Thread Multiprocessor Architecture for Computer Vision and Image Processing: Optimised Version of AMP	✓	✓		✓	✓	✓
8	Learning in Systolic Neural Network Engines	✓	✓	✓	✓		✓
9	A Reconfigurable data-driven Multiprocessor Architecture for Rapid Prototyping of High Throughput DSP Algorithms	✓	✓	✓	✓		✓