WAVER-SCALE
MASSIVELY PARALLEL COMPUTING MODULES
FOR
FAULT-TOLERANT SIGNAL AND DATA PROCESSING

R. M. Lea
Aspex Microsystems Ltd.
Brunel University
United Kingdom

Introduction

The WASP project was started at Brunel University in October 1983. From January 1985 to December 1989 the project was funded under a UK Alvey (VLSI) contract, involving Plessey, GEC, ICL and Middlesex Polytechnic; which included the design (at Brunel University), fabrication (by Plessey) and evaluation (at Brunel University) of defect/fault-tolerant test chips and three WASP technology demonstrator wafers (viz. WASP 1, WASP 2a and WASP 2b). Following the successful demonstration of these proof-of-principle wafers, application-oriented WASP devices are planned for development by Aspex Microsystems. Indeed, the strategy is to develop a definitive WASP application demonstrator, with a 3-phase stepping-stone sequence of experimental prototypes (WASP 3, WASP 4 and WASP 5), under a US SDIO-IS&T contract.

The application-pull and generic architecture for the WASP programme have been described [1,2] and the practical progress of WASP 1/2 programme has been reported [3,4] in previous workshops and conferences. This paper describes current progress in the the WASP 3/4/5 programme.

WASP device architecture

A WASP device is a WSI implementation of an ASP (Associative String Processor) substring and, as such, it constitutes a fundamental building block for the assembly of SIMD Massively Parallel Computer (MPC) components.

The WASP floor-plan is composed from 3 different "chip" blocks; implementing wafer-interfacing Data Routers (WIDRs), ASP substrings and wafer-interfacing Control Routers (CR/WIs).
Extrapolation, from the results of WASP experimental development, suggests that, with standard 1µm CMOS (double-layer metal) fabrication technology, 7 320-processor ASP substring blocks could be accommodated within each of 6 WASP rows on a 57mm x 57mm wafer. Thus, a single WASP device accommodates 13,440 processors.

Since WASP devices must tolerate defects in manufacture, as well as faults in service, not all of the implemented processors will be harvested. Indeed, the harvest will vary from wafer to wafer. Nevertheless, when WASP devices are concatenated to form a single ASP substring, it is the average (rather than the actual) harvest which is important. Moreover, it is unrealistic to expect all wafers to be usable. In practice, WASP defect/fault-tolerant circuitry targets harvests of at least 8,192 processors (i.e. a 61% harvest) on all wafers passing standard parametric probe-tests.

The W/DR and CR/WI blocks incorporate defect/fault-tolerant routing to connect ASP substring rows to a common Data Interface (DI) and a common Control Interface (CI) respectively. In addition, both these blocks incorporate ports to effect row-to-row extension of ASP substrings. Moreover, the W/DR and CR/WI blocks incorporate double-buffers to support efficient pipelining of continuous input data streams, fully overlapped with parallel processing in the ASP substrings.

The MPC modules can be assembled with WASP devices interfaced to local or global memories and local controllers or directly to a Data Communications Network and control bus.

**WASP defect/fault-tolerance**

Each of the W/DR, ASP substring and CR/WI blocks integrates a defect/fault-tolerant design: such that 16-processor blocks within the 320-processor ASP substring blocks, individual ASP substring blocks, entire ASP substring rows and entire WASP devices can be bypassed.

Since processors are identical and content-addressed, not only defects in manufacture, but also faults in service, can be "repaired" by not including them in ASP substrings. Moreover, such hierarchical bypassing allows both small-scale and larger clusters of defects/faults to be similarly tolerated.

The WASP power distribution network is also defect/fault-tolerant. Each ASP substring row is independently powered; the power-tracks running along the rows and being fed from external connections at both ends within the W/DR and CR/WI blocks. In addition, each block contains a power isolator, such that bypassed blocks and rows can be isolated from the power network.

The creation of a working ASP substring starts with the simultaneous testing of all processor blocks, with those processor blocks failing to generate the correct test-signature being bypassed.
The testing strategy for subsequent ASP substring building depends on inter-processor communication timing constraints; since bypassing groups of 3 or more adjacent blocks, to achieve neighbour activation, will require more than a single time-slot. Therefore, in order to maintain the highest speed, ASP substring blocks including such processor block groups must be bypassed, even though working processor blocks could have been harvested. Similarly, ASP substring rows including such groups of ASP substring blocks must also be bypassed, even though working ASP substring blocks could have been harvested. Nevertheless, by allowing extra time-slots for inter-processor communication, the processor harvest could be increased at the expense of some degradation in processing speed.

In summary, testing and automatic (re)configuration of a modular ASP system, according to a (pre)selected timing strategy, will yield a working set of ASP substrings for subsequent parallel processing.

It should be noted that ASP substrings incorporate no specifically redundant processors; all processors are available for useful work. Thus, ASP modules provide an unprecedented opportunity for second-generation MPCs which require dynamic allocation of processors for the flexible and progressive trade-off of parallel processing power for increasing fault-tolerance. Indeed, fault-tolerant configurations of ASP modules could be assembled for application as highly-reliable, self-repairable and gracefully-degradable MPCs.

For example, data input could be regularly suspended, for a few milliseconds in every minute, to allow testing and ASP substring reconfiguration. Moreover, multiple-buffered ASP configurations could avoid such interruption of data input.

**WASP packaging**

The size (57mm x 57mm) of the WASP device has been specifically chosen to satisfy the packaging constraints for both of the following two application environments.

- **printed-circuit board**: single WASP device packaged in a standard 184-pin 2.5" x 2.5" Kovar can or ceramic equivalent (originally developed for multi-chip hybrid assemblies)

- **aerospace**: four WASP devices (arrayed 2 x 2) on a 5" x 5" light-weight SEM-E compatible substrate.

SEM-E modules are based on a 6.4" x 5.88" thermal conduction plate; with a substrate, supporting microelectronic circuitry, attached to each side. In service, the plate is clamped along its top and bottom edges to heat sinks. Currently, power dissipation is limited, by the thermal resistance of the clamps, less than 50W; whereas the Silicon wafers and plate could support much higher thermal conduction. The racking pitch for SEM-E modules is 0.6".
CONCLUSIONS

The progress of WSI ASP hardware and software research heralds the development of highly-versatile and fault-tolerant WASP modules for the construction real-time signal and data processing systems; with significant benefits in size, weight, power and cost compared with existing parallel computing implementations.

Progress to date suggests that cost-effectiveness figures-of-merit of up to 100 Tera-OPS/ft³, 1 Giga-OPS/W and 10 Mega-OPS/$ can be achieved with WASP modules.

More significantly, the results of application benchmarks and, especially, of VLSI and WSI ASP device and system prototyping are providing proof-of-principle that application-specific parallel computing hardware can be cost-effectively built with WASP modules, with advantages in application flexibility, fault-tolerance and simple scalability.

In conclusion, WSI MPC modules offer the prospect of breaking through the cost-effectiveness barrier currently impeding the wider commercial exploitation of massively parallel computing systems.

ACKNOWLEDGEMENTS

The author gratefully acknowledges enthusiastic contributions to ASP-based projects from past and present members of the Parallel Computing Technology group and Aspex Microsystems Ltd. at Brunel University and the support of the UK Alvey initiative and the sponsorship of the US SDIO Innovative Science and Technology Office (administered by the Office of Naval Research) and US SDIO Sensors Office (administered by Rome Laboratory).

REFERENCES