THE PLACE OF INCREMENTAL TECHNIQUES IN FUTURE COMPUTER SYSTEMS

By B.R.Gaines

Distribution of Cost, Reliability and Size in Computer Structures

The main impact of integrated circuits on the economics of computer structures has been to heighten the imbalance between the cost of the processing unit (arithmetic and control logic) and the cost of core-store and peripherals. Savings in the cost, improvements in the reliability and decreases in the size of the processor circuits are made insignificant by comparison with the cost, size and reliability of core-store and peripherals. Only if improvements in logic-circuitry fabrication may be used to decrease the size of program required and simplify the peripheral equipment do they have an appreciable impact on the hardware economics of computers.

There is a limit to the extent to which increases in processor complexity can reduce program storage requirements in a general-purpose computer facility. For a given computer and computation, the bit-pattern forming a program may be regarded as a binary word encoding the computation. The optimum coding to give a minimum mean word length depends on the distribution of the computational mix encountered by the computer. The degree of sequential decoding used in past generations of computers has been excessive on coding-theoretic grounds, but it was, of course, determined by alternative considerations of cost, size and reliability which led to the minimization of processor hardware.

Thus the utility of LSI in general-purpose computer systems depends on the extension of the processor by adding such capabilities as: multiple hardare accumulators and index registers, priority interupt systems with associated register-saving operations, autonomous data-transfer units, hardware multiply/divide and variable-length arithmetic units, and, most importantly, high-level languages features such as DO-loops.

The optimum hardware configuration for a general-purpose, facility-type computer depends on the problem mix and relative costs of increased hardware versus increased throughput. The optimum hardware configuration for other data-processing and control computer systems may be determined from the same coding-theoretic and economic considerations, but, since the constraints are different, the configurations cannot be expected to be the same. For example, in a computer dedicated to a real-time task excess speed cannot be used to raise throughput and will be wasted - the favourable trade of extra hardware for throughput does not apply, and some other advantage must be taken of the quantity of devices made available through LSI.

There is a class of real-time data-processing problems arising in automatic control, aircraft and marine navigation systems, product quality monitoring and air-traffic control where complex computations have to be carried out with a small amount of data in a comparatively long time (several milliseconds),

---

Department of Electrical Engineering Science, University of Essex.
Standard Telecommunication Laboratories, London Rd., Harlow, Essex.

and it is suggested that the optimum computing configuration for these problems may be based on a processor using incremental computing for arithmetic operations. Incremental, or 'counting', techniques have been previously confined largely to pseudo-analog machines, such as the digital differential analyser, and their utilization in the arithmetic unit of conventional computers has been neglected.

Incremental Computing Techniques - The Phase Computer

The advantages of incremental computing techniques are that they may be realized with simple arrays of conventional elements, counters, full-adders, registers and so on, all amenable to LSI, and that they may be used in simple configurations to compute, in a single step, quite complex functions which would be multi-stage sub-routines in a conventional computer, e.g. -

$$X \rightarrow X + k(Z-X) \ , \quad Y \rightarrow Y + j(Z-X) \quad ....(1)$$

or

$$X \rightarrow (X^2 + Y^2)^{1/2} \ , \quad \theta \rightarrow \tan^{-1}(X/Y) \quad ....(2)$$

These computations may be realized by simple incremental computing configurations in which they are the final state of a pseudo-analog loop defined by a differential equation. The classical DDA, however, is equivalent to a processor using entirely non-sequential encoding of the computation - the opposite extreme to that adopted in the classical general-purpose digital computer. The optimum configuration would seem to lie between these extremes and utilize a small, simplified DDA-like processor, coupling to a sequencing unit which multiplexes it through the various stages of a computation.

The Phase Computer[≠] is presented as an example of such a configuration - it is a modular system based on five elements:-

counters, which are unidirectional, ripple-through binary counters;

modulators, which are gated full-adders with associated registers, that add the contents of a counter into the register and output an incremental sequence of overflows;

input-control units, which are AND/OR gates switching one of several input lines onto a single output line;

sequencers, which are the digital equivalent of rotary switches, being counters with an output line corresponding to each state of the counter;

differentiators, which are gate/flip-flop combinations detecting changes in the most significant bits of counters, and hence indicating when the counter enters certain states.

The counters and modulators are used to set up pseudo-analog computing loops via the input-control gates. The outputs of the sequencers select the loop in operation through these gates, and the differentiators cause the sequencers, and hence the computation, to advance stage by stage according to the results obtained.

A basic three-counter, one-modulator, one-sequencer, processor may be used to transfer data between registers, add, subtract, multiply, divide, square or square-root, together with combinations of these.

≠ Gaines, B.R., Joyce, P.L., Phase Computers, Proc. 5th Int. Congr. AICA, Lausanne, 1967.