THE APPLICATION OF PHASE COMPUTERS TO AUTOMATIC TRACKING

By B.R.Gaines

Digital Computers The variety of possible data-processing techniques is very great indeed, but one would hardly become aware of this through analysis of hardware currently in use. A certain configuration in which a single and very versatile processor coupled to a large passive store is multiplexed through all the necessary steps of a computation using the store to hold uniformly the data, partial computations and multiplex sequence, is ubiquitous - the stored-program digital computer has no competitors. It is the intention of this paper to demonstrate that this monopoly is, to some extent, artificial, and that the digital computer as we know it now is only one of a family of digital computers, other members of which may be better suited to particular computations; the application of phase computers to the least-squares smoothing and prediction equations used in automatic tracking is taken as an important illustration.

The advantages of the stored-program, single-processor digital computer are well-known: in particular a repertoire of basic logical operations enables it to simulate, albeit inefficiently, almost any other computational device, and its single-operation, sequential mode of computation enables this simulation to be programmed very easily. Another advantage which is not often made explicit is that the immense development cost of the digital computer has been spread thinly over many activities, and its present cost is very low relative to that of other computing hardware. The disadvantages of the digital computer in real-time operation are becoming known: in particular that versatility does not imply flexibility - changing a complex computation is difficult because it has been split into so many minor interacting parts, and that multiplexing a single processor causes a fall in bandwidth with the size of problem that makes multi-computer systems mandatory for the simulation of relatively small but fast systems. There are many applications where these limitations are not yet major obstacles, however, and the wrong tool may be used because it is available, cheap and satisfies the initial requirements; the danger is that research becomes increasingly directed to fitting the job to the computer, and simple and elegant solutions in alternative hardware may be missed.

Early digital computers were developed as university facilities for which flexibility was of prime importance; vacuum-tube hardware was massive, unreliable and produced vast quantities of heat; drum and core stores were far cheaper than the equivalent vacuum-tube store; hence we have a single multiplexed processor and a large passive store containing both program and data. One may question whether these forces which moulded the development of conventional digital computers are still present today - solid-state technology is still in its infancy, but has already changed the size, speed, reliability and cost of processing and storage techniques by orders of magnitude - real-time operation is becoming increasingly important, although it is still a small market compared with business and scientific

---

Standard Telecommunication Laboratories, London Rd., Harlow.

computing - peripheral equipment is often far more expensive than the 'main computer'. Our work at STL is concerned with the questions, 'What kind of computing system would have developed, not as a university facility restricted by the hardware of the fifties, but for the prime objective of: telephone switching ?, adaptive control ?, character recognition ?, automatic tracking ?, and so on.' One outcome of this work has been the development of the Stochastic Computer [1] for the digital simulation of linear and nonlinear analog computations, particularly the multiplication and storage involved in adaptive control, and another outcome has been the development of the Phase Computer for sampled-data logical and numerical computation; it is the latter and its application to the equations of automatic tracking which are discussed in the remainder of this paper.

The Phase Computer The main danger in attempting to escape from the confines of present computer techniques is that hardware may be produced which is very powerful, yet virtually uncontrollable. Anyone who has attempted to patch the mode control and analog/digital interface of a hybrid computer will know the difficulties in utilizing a multi-processor computer. As soon as the timing of many events in a multi-dimensional state-space have to be considered, the conceptual problems of simulation become very great. One is faced with the software problems of conventional digital computers increased by orders of magnitude; it is essential not to increase complexity of computation at the expense of simplicity in programming. Thus in a multi-processor computer one must achieve a drastic reduction in the number of control and output lines to and from each processor - ideally there should be only one of each. In the phase computer individual computing elements have only one or two control lines and one or two output lines - a basic processor containing three elements and capable of handling two or three sixteen-bit words has three control lines and eight output lines; this not only simplifies the programming of a multi-processor phase computer, but also makes it feasible to package a complete processor in a standard integrated circuit header.

The distinguishing feature of the phase computer is the extensive use of active differential storage throughout the computing system. Core and drum stores are passive in the sense that no operation is possible on the stored data apart from 'read' and 'over-write', whereas computational transformations of the stored data such as 'increment' or 'rotate' are possible in an active store; an array of bistables is basically a passive store, but becomes active if they are connected as a binary counter or shift register. In an active differential store two storage elements are used to represent a single variable, subject to the constraint that the same sequence of operations applied to both (active) stores does not change the stored value. Hence, if the stored value may be read out of one store (the Store) when the other store (the Reference) is in some given state, then it may equally be read out when an operation found to be an identity transformation for the Reference is applied to both stores. This principle enables an inverse operator to be applied to a Store by applying the operator to its Reference, and since one store may act as Reference to many others, or both as Store and Reference, it is not uneconomical in hardware; most importantly it enables dynamic interactions to take place between stores without loss of the stored value.

The particular form of phase computer to be described here is the Phase-count Computer intended for numerical data-processing. The basic elements are unidirectional counters which are also cross connected to act as digital modulators. These latter are devices for converting one sequence of (clocked) logic levels to another, in which the number of logic levels ON at a clock-pulse is multiplied by a fractional binary number; they are a development of the 'binary rate multiplier' first used at M.I.T. for interpolation in the automatic control of machine tools [2], and recently by Schmid in his operational hybrid computer [3]. A basic processor for the phase computer is shown in Figure 1: <A>, <B>, <C> are three (16 bit) binary counters which increment by unity at a clock pulse if their respective control lines, A, B, C, are ON; when a counter is in the state 0000000000000000 its output line, A_o, B_o, or C_o, is ON, and when in the state 1000000000000000 its output line, A_1, B_1 or C_1, is ON; the modulator lines, A_b and C_b, are ON during one complete cycle of store <B> for numbers of clock pulses equal to the counts in stores <A> and <C> respectively - if <B> counts for some fraction of a cycle then A_b and C_b will be ON for the best approximation to that fraction of the counts in <A> and <C>. In a typical configuration <B> will be used as Reference for Stores <A> and <C>, such that when B_o is ON the binary numbers in <A> and <C> are the quantities held in these Stores (negative quantities are handled in the usual 2's complement representation). This processor may be used to add and subtract, to multiply and divide, to form the square and square-root, to form the sine/cosine and inverse sine/cosine, and so on for all the basic linear and nonlinear computations. Several processors may be combined to carry out multiple operations or to process parallel data streams, and the least-squares smoothing and prediction equations for two-dimensional, sampled data using first order prediction will be used to illustrate these possibilities.

Realization of Tracking Equations Table 1 shows the standard least-squares tracking equations, and Figure 2 illustrates one channel (X-coordinate say) of a phase computer for solving them. P_N is assumed known to 10-bit precision, and its X-coordinate has been read-in as the 10 most significant bits of the 12-bit store <P>. <R>, <F>, <V> are 12-bit stores acting as Reference and Stores for the X-coordinates of F_{N-1} and V_{N-1} respectively. It is assumed for the moment that modulators are available with outputs proportional to 1/2, $1-K_v$, $1-K_p$ and $1-K_p+K_v$, respectively. Suppose the stores hold the quantities described, output line R_o is ON, and it is required to implement the up-dating equations, (1) and (2): R, P, F and V are connected to '1' until P_o or F_o comes ON; if F_o comes ON then F and V are switched to '$1-K_p$', '$1-K_v$' respectively until P_o comes ON, whereas if P_o comes ON then R, F and V are switched to'$1-K_p$', '1', '$1-K_p+K_v$' respectively until F_o comes ON; finally R, P, F and V are connected to '1' until R_o comes ON, when <F> will hold G_N and <V> will hold V_N. To implement the prediction of equation (3): R, F and V are connected to '1' until R_1 or V_1 comes ON; if R_1 comes ON then P is put OFF until V_1 comes ON, whereas if V_1 comes ON then R and F are connected to '1/2' until R_1 comes ON; finally R, F and V are connected to '1' until R_o comes ON, when <F> will hold F_{N+1} and <V> will still hold V_N.

As the next stage of computation the modulator output of <V>, V_r, could be used to form the square of the X-component of V_N in the working space <P>, together with the square of the

Y-component formed simultaneously in the other channel, and then to extract the square-root for display as the 'speed' via a BCD counter. The computation of heading, cross-track error, error in angle and error in radius may be performed either simultaneously or in sequence, dependent on the working space available. In practice K_p and K_v are effectively generated from equations (4) and (5) in terms of Q, the number of data points considered. Q-2, Q-1, Q, Q+1 are held in 3-bit modulators and updated by the turn-detection channel; this requires far less working space than a table look-up procedure. To take advantage of the greater smoothing of velocity data, three additional bits of precision in storage are added by using $8K_v$ as the multiplier in updating V_{N-1}. Using 1Mcs clock pulses the predicted position is computed in 8 msec, and speed and heading are available after a further 8msec.

Conclusions It has not been possible in the space of this paper to do more than indicate that there are digital computers, differing in many respects from conventional machines, which offer some advantages in particular applications. The solution of sampled-data smoothing equations by the phase computer is one example of the simple and natural approach possible with appropriate hardware. It is unlikely that the multi-processor computers required for real-time operation will develop on the same lines as present single-processor machines - parallel computation decreases the need for very great speed in individual operations - cheap integrated circuits relax the need for economy in hardware - the fundamental constraint in the design of these machines will be ease of use. A very attractive configuration would be a small digital computer acting as the control and sequencing unit, coupled to many processors, similar to those of the phase computer, powerful in computation, simple to control, and compatible with conventional computer techniques.

Acknowledgments I would like to thank my colleagues both at S.T.L., Harlow, and at S.R.T., Stockholm, for lengthy discussions on computers and their application in radar systems, and also our two companies for permission to publish this paper.

References [1]Gaines:'Stochastic Computers' in Encyclopaedia of Information, Linguistics and Control, Pergamon (to be pub.1967).
[2]Rosenberg:'Automatic Machine-tool Control' in Computer Control Systems Technology, Ed:Leondes, McGraw Hill 1961.
[3]Schmid:'An Operational Hybrid Computing System' IEEET-EC12 Dec'63

Table 1 $G_N = F_N + K_p(P_N - F_N)$ ...(1) $V_N = V_{N-1} + K_v(P_N - F_N)$..(2)
$F_{N+1} = G_N + V_N$ ....(3) $K_p = 2(2Q-1)/Q(Q+1)$..(4) $K_v = 6/Q(Q+1)$

Figure 1

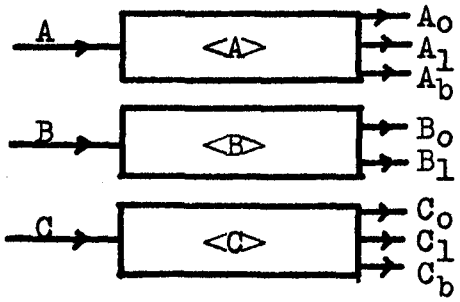
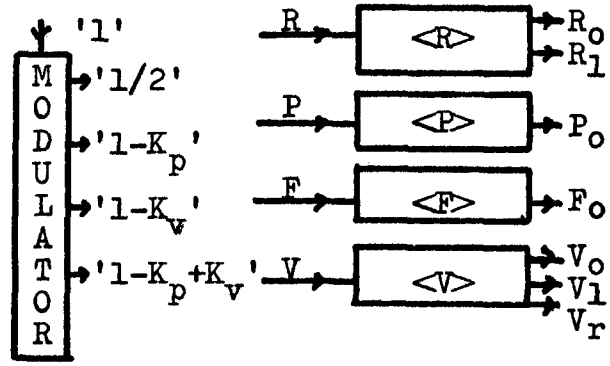
Figure 2