We note that, for any set of representative problems, there is a common set S of basic operations which both machines will perform: the purely arithmetic storage-to-storage operation (excluding pseudo-arithmetics like subscript manipulation and sign change). We now regard the proposed machine P as comprising: Pl, the orthodox register-data flow configuration, together with the microprogram for executing the operations S; P2 the remaining microprogram, which includes the compiler, array manipulation, function execution, and general stack-organization facilities.

Cost

Because the pushdown is microprogrammed, there is no special circuitry in the machine, which may be regarded as typical of machines of the same size. The cost of the proposed philosophy therefore lies wholly within the P2 microprogram, which is in fact almost exactly the same size as that of P1. In a machine in the low-cost scientific field, therefore, the cost is approximately that of doubling the microprogram of a more conventional machine.

Note that the basic FORTRAN processor needs only typewriter input-output.

Performance

Machine X is provided with a single-pass FORTRAN compiler. A set of representative problems was:

(i) hand-coded for machine X,

(ii) written in FORTRAN, and

(iii) compiled on machine X.

After allowing for difference in basic circuit speeds, the execution times for the described machine were, for all problems:

(*a*) greater than the hand-coded solution times, but

(*b*) less than the compiled solution times.

The instruction storage required by machine P was at all times less than that required by either configuration of machine X. The time taken to write the programs in FORTRAN was considerably less than to hand-code them.

Efficiency

For programs which are to be run repeatedly, optimum coding for inner loops becomes increasingly important. If the program is to be restricted to FORTRAN, the P2 microprogram may have to be larger and more sophisticated. Clearly any compiler feature can be implemented in microprogram—at a price. Two factors contribute to the minimization of the P2 microprogram: the close structural resemblance between source and object program, and the fact that the compiler itself can be implemented in a language (chosen by the machine designer) which is itself independent of source and object languages.

The machine is intended for the inexperienced programmer who wishes to express and solve scientific problems, preponderantly "one-off," with a minimum of delay and cost. Time saved in writing and testing programs will more than compensate for any inefficiency in inner-loop processing. Direct communication with the machine will quicken the education process.

Conclusions

There is little doubt as to the usefulness of a machine which provides the facilities described here. Of the several design philosophies conceivable, we have studied the feasibility and practicability of incorporating the FORTRAN processing facilities in the machine circuitry. Some measure has been given of the effect of the proposed philosophy on the cost and performance of the basic computer unit. It is felt that any depreciation here is more than offset by the improved efficiency and operational facilities of the overall system.

Acknowledgements

The design and a complete simulation of the machine described formed part of an exploratory machine design study under Mr. C. E. Owen, IBM World Trade Laboratories (Great Britain) Ltd. The authors wish to thank the laboratory management for permission to publish this paper.

Errata

A simplex method for function minimization by J. Nelder and R. Mead.

The following alterations are required in the appendix to the above paper which was published in Vol. 7, p. 308. The two expressions for the information matrix should have a factor of 2 attached, and that for the variance-covariance matrix a factor of $\frac{1}{2}$. The authors are grateful to Dr. Rodes Trautman for drawing their attention to this slip.