HARVEST

~~Stretch~~

20 October 1959

SIGMA INSTRUCTION EXECUTION TIMES

Due to extensive multiplexing, the time consumed in executing on instruction is not derived by adding up the time for each operation which went into the instruction. The Sigma Computer is capable of having as many as 11 different instructions, each one of which may have different operational requirements (each such operation requiring different time durations) all being processed concurrently. This is achieved by the technique of asynchronous operation. In order to establish the speed of execution of an instruction, a concept of rate of flow must be adapted. In considering this, two basic concepts must be kept in mind:

a. To measure a rate of flow it is necessary to apply the measurement of a point in the system through which the full flow must pass (no parallel paths).

b. The general systems organization of a stored program machine is essentially a serial operation in which each instruction executed may assume that the previous instruction is completed.* This introduces the requirement on the Sigma system that information available to the program (has a memory address) cannot be modified except in the sequence in which the instructions originally appeared in memory.

Principle b gives rise to a unique point in each instruction execution at which the results are allowed to effect addressable registers i.e. Accumulator, Interrupt Register, etc. The time duration between these points is the best measure of the flow rate.

In accordance with this philosophy of performance, execution times (rate of flow) of typical instructions under average conditions are:

Floating Point Add	0.95
Floating Point Load	0.6
Floating Point Store	0.6
Floating Point Multiply	1.8

* The operation of I/O by computer instructions is the only departure from this presently introduced.

Floating Point Divide	7.0
V F L Add and Connect, BR on Bit (1 byte)	$(2 + \frac{n}{8})$ 0.5
V F L Binary Multiply	$0.5\ (14 + \frac{n}{4}) + \frac{(n + 2)}{12}$ 0.4 usec
Typical Unsuccessful Branch or	1.0
Typical Index Arithmetic Instruction	1.0
T M T, Refill	5 usec + 2-3 usec/word

n = Field length in bits

The above are subject to a 25% contingency due to circuit reliability considerations.

A program is being written to time routines for Sigma. STRAP-1 coding will be used for input. The program should be available in 3-4 months.

I received the above information from Bob Jordan of IBM on 1 Oct 1959. Some of the features available in Sigma i.e. adding the accumulator to itself, are real time consumers. The programmer should avoid using "tricks" until more information is available about Sigma. rate of flow.

Joseph E. O'Hara

JOSEPH E. O'HARA
MPRO-03