BURROUGHS CORPORATION
Product Planning, DPS

2115
U2111

February 16, 1961

SUBJECT: NEW CONCEPT FOR A COMPUTER

To: W. R. Lonergan, Manager, Product Planning, DPS

From: Don Knuth, Product Planning, DPS

The following ideas might prove to be very feasible for designing an inexpensive extra-fast computer for scientific applications.

Some early computers were single-address machines, others two-address, and still others with three addresses in the instructions. The idea to be introduced here allows instructions to have any number of associated addresses. Advantages of the method are its ease of implementation and the way it minimizes the time needed to fetch instructions from storage to the program register.

To illustrate the ideas, I will consider a hypothetical computer which uses the system. I really didn't spend much time putting finesse into the instructions, but I hope they do illustrate the usefulness and power of the method.

1) Registers which concern the programmer
   a) A register. 48-bits binary, used for all arithmetic operations. The layout of this register is identical with B5000 floating point words.
   b) B register. 12-bits binary; an index register.
   c) I register. 48-bits or less; this is an interrupt-select register which keeps track of all conditions under which an interrupt is desired.

2) There are 4096 locations of core memory, each 48-bits binary.

3) Program Syllables.
   a) Each word of the program is divided into three 16-bit syllables, and these normally enter the program register sequentially. Each syllable has the form:

12	34	5 - 16
Type	Index	Address, or Operation

   b) Type 00 syllables. These are operator syllables; they specify which operation will take place when future address syllables appear, but they do not necessarily initiate any action immediately.

c) Type 01, 10, 11 syllables. These cause an operation to be performed, according to the previous operator syllable which was encountered.

d) Example. Following an ADD operator syllable, a type 01 syllable will cause a "clear-add" to be performed; a type 10 will cause an "add" to be performed; and a type 11 will cause a "store" to be performed. These syllables may appear in any order or any number.

ADD, 01 X, 10Y, 11Z causes X+Y→Z
ADD, 01 X, 10Y, 10Z, 11W, 11T causes X+Y+Z→W and T.
ADD, 01 X, 11Y causes X→Y.
ADD, 10 X adds X to the accumulator (and leaves the result there)
ADD, 01 X, 10Y, 11Z, 10W, 11T, 01F, 11H causes X+Y→Z, Z+W→T, F→H.

e) The index field. An index field of 00 means the address is absolute. An index field of 01 means the address is to be B- modified (mod 4096). An index field of 10 means the 12-bit address is to be considered as a literal value, except in "store" or "jump" meanings. An index field of 11 means the entire index plus address field is to be replaced by the 14 low order bits of the word in the specified place, and the new index-field is to be interpreted again.

4) Operators. Under each operator we list what happens immediately when the operator syllable occurs, and what happens when address syllables follow.

a) ADD, ADD ABSOLUTE, SUBTRACT, SUBTRACT ABSOLUTE, MULTIPLY, DIVIDE, INTEGER DIVIDE, REMAINDER DIVIDE, LOGICAL AND, LOGICAL OR, LOGICAL NEGATION are all similar. They cause no action when the operator appears, but type 01 syllables following execute "Clear Add", type 10 syllables execute the indicated operation as on the B5000, type 11 syllables execute a "store".

b) SQUARE ROOT. When the operator occurs a square root occurs immediately. 01 syllable following does a "clear add", a 10 syllable does a "clear add absolute", and a 11 syllable does a "store."

c) ABSOLUTE. This sets the sign of the mantissa in A positive immediately. 01-syllables do "clear add absolute." 10-syllables "add", 11-syllables "store".

d) NEGATE. This changes the sign of the mantissa in A immediately. Succeding 01-syllables perform a "clear subtract". 10-syllables "add," 11-syllables "store".

e) INTEGERIZE. This converts the A register contents to an integer immediately. Succeeding 01-syllables perform a "clear add," 10-syllables set B to A, 11-syllables store.

f) B REGISTER. No immediate action. 01-syllables specify "set B to low-order 12 bits of this value". 10-syllables do the same. 11-syllables store B.

g) CHANGE B. 12 bits of this operation specify an increment, and and when the operator is sensed, B is replaced by B plus this increment (mod 4096). A carry toggle remembers whether or not this sum was greater than 4095. Following this syllable, 01-syllables "branch to this location if carry toggle on", 10-syllables "branch if carry toggle off", 11-syllables store B.

After branches, the left-most syllable of a word is executed first.

h) BRANCH UNCONDITIONALLY. No immediate action. 01-syllables, branch. 11-syllables, "enter" a subroutine. This process consists of storing the location of the next three syllables into the address field of the left-most syllable of the address specified, then transfers to the middle syllable of that word. 10-syllables merely store the location of the next three syllables into the left-most address field, does not branch.

i) BRANCH CONDITIONALLY. No immediate action. 01-syllable branch if the accummulator is positive, 10-syllables branch on zero, 11-syllables branch on negative.

j) EXPONENT. 01-syllables, set exponent of A to mantissa of value. 10-syllables, set exponent of A to exponent of this value. 11-syllables, store exponent of A as an integer.

k) STORE ADDRESS. 01-syllables, store low order 14-bits of A in left-hand syllable (10-syllables, in middle syllable, 11-syllables, in right-hand syllable) of address specified.

l) CARD READ or WRITE. The unit designation is part of the operator. 01-syllables, initiate card read or write, 10-syllables, same; 11-syllables, load format band.

m) INTERRUPTS. 01-syllables: "or" this word to I register; 10-syllables: "and" to I-register; 11-syllables: store I register.

Don Knuth /ga

Don Knuth
Product Planning, DPS

DK:ga