J. E. Griffith

Poughkeepsie, New York
December 12, 1955

PROJECT STRETCH
FILE MEMO # 12

COMPANY CONFIDENTIAL

SUBJECT: Indexing
By: G. M. Amdahl, E. M. Boehm, J. E. Griffith

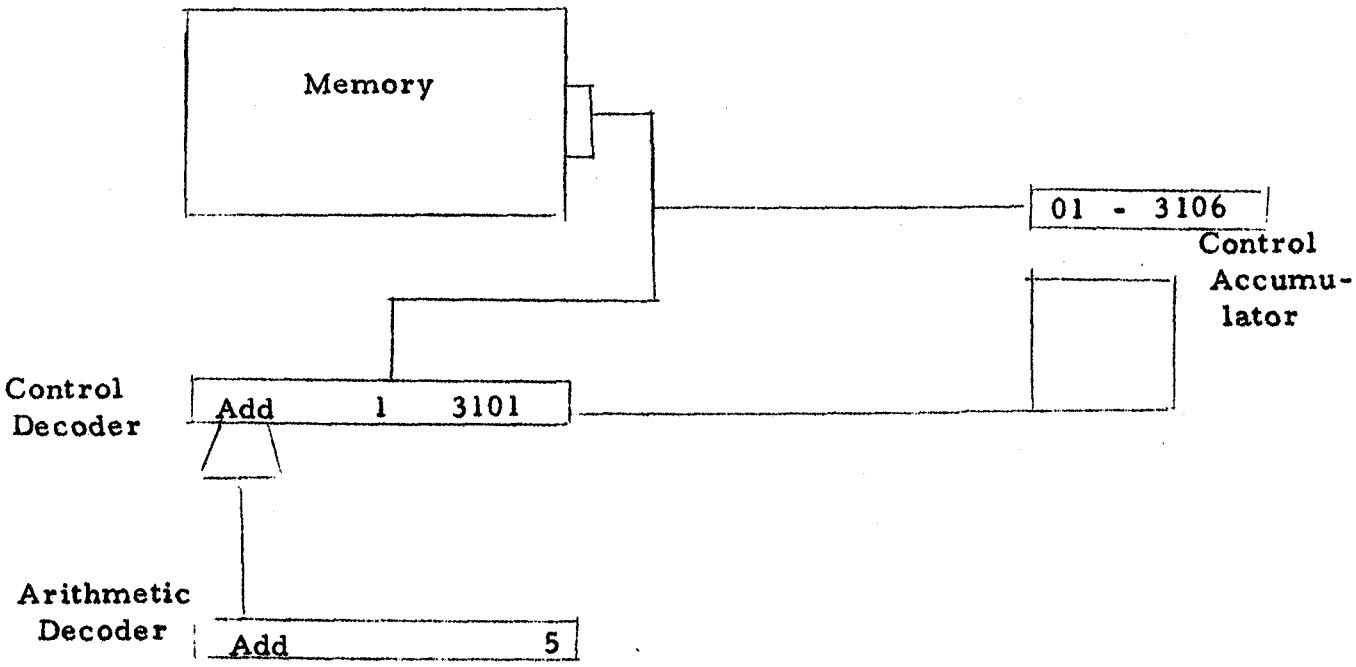
The description of Stretch's indexing system will be divided into three sections. Section 1 will describe the logical operation of indexing in address modification. Section 2 will describe the operation of the index registers, the control accumulator and the types of instructions provided to utilize the indexing system. The final section will describe the system's modularity, and outline features which may be included in the system at a future date.

Section 1.

An instruction normally consists of three parts - an operation, an index tag and an address. The index tag is used to alter the "effective" memory address before executing the instruction. Before the memory "read" of the add instruction takes place, the contents

Operation	Tag	Address
Add	1	3101

of index register 1 are algebraically added to the address part of the instruction. The resulting sum is the "effective" memory address used by the instruction. Since the index register quantity may be positive or negative, the sum may also be a negative quantity. However, the sign of the sum is disregarded, and only the absolute quantity is used in making the memory reference.



In the example above, the instruction will be executed as Add the contents of memory location 5. The tag position of an instruction will correspond to two decimal digits, and may refer to any index register between 1 and 99. When the contents of an index register are brought into the control accumulator from memory, the number of the index register is held in the control accumulator. Holding the number of the index register in the control accumulator will decrease the number of memory references, where one index register is referred to several times without an intervening reference to another index register. The first instruction in a sequence to refer to an index register will capture one memory reference to bring the instruction into the control decoder and a second reference to bring the contents of the specified index register into the control accumulator. Succeeding instructions referring to the same index register will capture one memory reference to bring the instruction into the control decoder and will use the contents of the control accumulator for the index reference.

After an instruction has been brought into the control decoder, the tag position of the instruction is compared with the index number in the control accumulator. If the two numbers match, the address of the instruction is added to the contents of the control accumulator. When the two numbers do not match, a memory reference is made to transmit the contents of the desired index register, and its number into the control accumulator before the addition takes place.

Zero in the tag part indicates that the address part of the instruction is the "effective" address (no address modification). Before the tag part of an instruction is compared with the index number in the control accumulator, the tag is tested for zero. If a zero tag is found, no comparison is made, and nothing is added to the address part of the instruction.

Not all of the 15 decimal positions of the Stretch instruction word are needed to specify the operation, the address, and one of the 99 index registers. The unused positions are used to increase the flexibility of the indexing system by making one of the unused positions a control digit that alters the interpretation of the unassigned positions. One logical extension of the indexing system is permitting a full address tag. When a full address tag is used, the three positions adjacent to the tag position are combined with

Control	Opn.	Full add. Tag		Address
		Tag		
1	3	3	2	5

the tag to form a 5 digit address. When a full address tag is given, the contents of the specified memory location are brought into the control accumulator, but no index register number is assigned to the control accumulaotr, and the number previously held in the control accumulator is wiped out.

A further logical extension of the indexing system is double indexing. By placing the proper control digit in an instruction the unassigned positions are interpreted as index register tags. A maximum of three index registers may be referred to in a single instruction.

1	3	2	2	2	5
Control					
Operation code		Secondary tags		Tag	Address

When an instruction of this type is encountered, the contents of the tag part are compared with the number in the index register and if necessary, a memory reference made. The contents of the control accumulator and the address part of the instruction are added and transmitted to the address part of the control decoder. The contents of the index register specified by the secondary tag are brought into the control accumulator and added to the address part of the control decoder.

Control	Opn		Secondary Tag	Tag	Address
X	Add	03	16	02	01 1 1 3

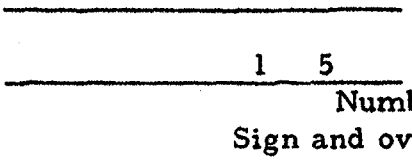
C (03) - 119
C (16) + 420
C (02) + 03

At conclusion of indexing operation

Control Decoder			Control Accumulator		
X	Add	1417	03	-	00119

Section #2

When an index register is used in address modification, the quantity is brought into the control accumulator for use, and also is retained in memory. Each of the 99 index registers is assigned a memory location. The sign, the status of overflow and the fine decimal digit quantity are held in the index register



In addition to bringing an index register into the control accumulator for address modification, index register contents may be altered and tested in the control accumulator. If the c.a. is loaded from an index register, the number of the index register is held in the control accumulator. When loading from other memory locations, the no. position of the c.a. is left blank. At no time is the control accumulator to associate a value with an index register unless the value is duplicated in the memory location of the index register. When the contents of the control accumulator are altered, any index number associated with the control accumulator is erased. If the contents of the control accumulator are stored in an index register the number of the index register is inserted in the control accumulator. To insure that operations in the main arithmetic unit do not alter the index register currently in the control accumulator, each instruction that writes in memory is tested when it is in the control decoder. If the address of the "write" instruction is the same as the index number in the control accumulator, the number in the control accumulator is erased. Writing from the control accumulator into general storage will not change any indexing number in the control accumulator.

The control accumulator contains 5 decimal digits, a sign and an overflow indicator. Quantities will be transmitted to the control accumulator in several ways. An index register will be loaded into the control accumulator by a reference for address modification - (ADD x Y..C (X) goes into control accumulator). When an index register is loaded from memory, the quantity is also sent to the control accumulator (Load Index Register x Y... C (Y) goes into index register x and into the control accumulator). Instructions to load the control accumulator from memory will be provided.

(Load Control Accumulator x Y...Contents of memory location (.. + C ..) are loaded into the control accumulator). Data may be loaded in several forms. When signed quantities are loaded, the signs may be reversed or unchanged, and the overflow indicator will be brought into the control accumulator. When unsigned quantities are loaded, the sign of the data is assumed to be positive, and the overflow indicator is assumed off.

Addition, subtraction, multiplication and shifting can be done in the control accumulator. A signed or unsigned quantity may be added or subtracted. For signed numbers the numeric or absolute value may be used. If the result exceeds 99,999 the overflow indicator will be turned on,. When multiplication is specified, the execution of the operation will be delayed until the main arithmetic unit is free. At this time the contents of the control accumulator and the multiplier will be sent to the main arithmetic section for execution. Only the five low order digits of the product will be returned to the control accumulator. If the product exceeds this amount, the overflow indicator will be turned on. Data in the accumulator may be shifted to the left or right. Information shifted beyond the boundaries of the control accumulator is lost.

The absolute or algebraic value of the contents of an index register or the control accumulator can be tested or compared against a predetermined value. Any test may be used to set a selector or determine a transfer of control. Tests may be made for overflow (or absence of overflow), zero quantity (or non-zero quantity) or positive (or negative) sign with a transfer of control determined by the outcome of the test.

When the contents of an index register or the control accumulator are compared with a predetermined quantity, the algebraic or logical value of the two factors may be used. If an algebraic comparison is made, the status of the overflow indicator is ignored. The compare instructions will set selectors which indicate whether the index register contents are higher, lower, or equal to the compared value. Compare instructions to give conditional program skips and transfer of control will also be included.

Signed or unsigned quantities[1] may be written into storage. Information may be written into an index register from storage, the control accumulator, the program counter or from the main arithmetic accumulator. Information from storage goes into the control accumulator before it is taken to an index register. A quantity may also be stored from the control accumulator. In both cases, the number of the index register written in, is attached to the contents of the control accumulator. When data is written into general storage, the number of the x register in the control accumulator is unaltered. When a quantity is to be stored from the main arithmetic unit into an index register, the number in the control accumulator is wiped out, if the address of the write instruction refers to the index register currently in the control accumulator. The contents of the program counter may be stored in the control accumulator and an index register. Either the two values may be exchanged, or one of the values may be copied into the other location.

Unless multiplication is specified, no I/O break-in is effective until the operation in the control accumulator is completed. If multiplication is specified and is being executed, the I/O break in is delayed until the product has been transmitted to the control accumulator. If the execution of the multiplication has not been started (main arithmetic unit in use), I/O break in will be effective immediately.

---

1. Signed - 5 decimal digits + sign and overflow status
   Unsigned - 5 decimal digits

The indexing system will be operative in two modes - decimal and non-decimal. Arithmetic operations and memory addressing will be done in either mode.

Section 3.

Indexing is a modular system, with the speed of the operation determined by the components of the control adder and the index register. The organization of the indexing logic remains constant. The control adder may be serial or parallel. Previously index registers have been termed memory locations. Speed of execution of indexing operations may be increased by using high speed transistor registers as the index registers.

Several features of the indexing system have been considered and no decision reached. These features will be listed below for future study:

1) Recording final value or decrement in unused portion of index register memory location.

2) Expanding the number of index registers.

3) Assigning selectors to various conditions of the control accumulator.

4) Providing instructions which conditionally modify the contents of an index register and conditionally transfer control.

G. M. Amdahl
E. M. Boehm
J. E. Griffith