HARVEST REPORT #11

Subject: Binary Merge Sorting

By: W. A. Hunt

Date: January 7, 1957

January 7, 1957

HARVEST REPORT #11

Subject: Binary Merge Sorting

The following is a description of a simple binary merge sorting system which uses the basic concepts of three continuous stream registers.

1. General assumptions:

1.1. The control data will be at the beginning of the record.

1.2. The data will be moved.

1.3. Internal sorting up to the maximum block size.

1.4. Time assumptions for internal merge sorting are one microsecond per record for control plus one microsecond per word per pass.

1.5. Time assumptions for tape passes are six microseconds per word per pass plus tape start stop time.

1.6. All items are the same length.

1.7. A program step is permissible after each "run". (A run is defined as follows: Assume that two ordered blocks of four items each are to be merged into an ordered group of eight items. The block length is four and the "run" is eight.)

2. Equipment Assumptions:

2.1. Four high speed tape units.

2.2. Two high speed tapes to operate concurrently, one in and one out.

2.3. Eight 2-microsecond memories with a minimum of $2^{16}$ words.

2.4. Three Continuous Stream Registers (CSR). Two for holding the streams to be merged and one for storing the merged stream.

2. 5. The control and equipment necessary to execute automatically one "run". (Defined in 1. 7.) This included the necessary comparing device.

3. Internal Information Flow:

3. 1. The control field of an item from each of CSR #1 and #2 will be compared and a High or Low signal will be obtained. Equal will be combined with the High or Low signal or will cause the item in one of the two CSR's to be advanced automatically into the third CSR. (It is yet to be decided whether comparing shall be done from the most significant data to the least significant or vice versa.)

3. 2. The High or Low signal will automatically cause the desired item to be advanced into CSR #3.

3. 3. These two operations will continue until a block is exhausted in either CSR #1 or #2 at which time either:

3. 3. 1. A program break-in is signaled or

3. 3. 2. The remainder of the second block is advanced without comparisons into CSR #3. The block length counters can be used to indicate that the end of a memory working area has been reached in which case 3. 3. 1. is needed. It may be desirable to program 3. 3. 2. rather than to make it automatic.

3. 4. As long as the blocks are less than the memory working area assigned to a CSR and the block is not the last block of the area, the next items to be merged are in the input CSR's and only the control information needs revisions. Advantage should be taken of this.

4. Control Information:

Control information must be specified for each of the three Continuous Stream Registers and the Logical Unit. This control information will be listed with some indication with which of the above units it is to be associated. Since CSR #1 and #2 may have identical functions, their controls may be considered identical. (It may be desirable to advance the item from one of the CSR's arbitrarily into CSR #3 as the fields are compared.) If the option, enclosed in parenthesis in the last sentence, is adopted, the controls of CSR #3 may be slightly more complicated to correct for the wrong assumption.

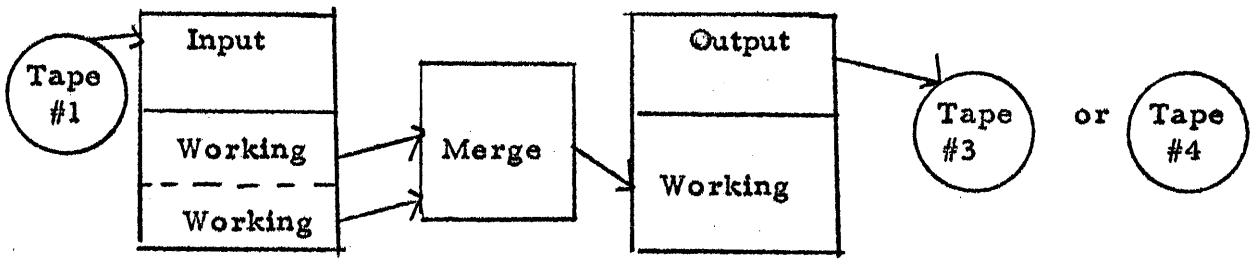
4. 1. The control information needed for each of the CSR's is:

Starting address
Byte size
Field size
Record size
Block length (may be omitted from CSR #3)

5. Working Areas in Memory:

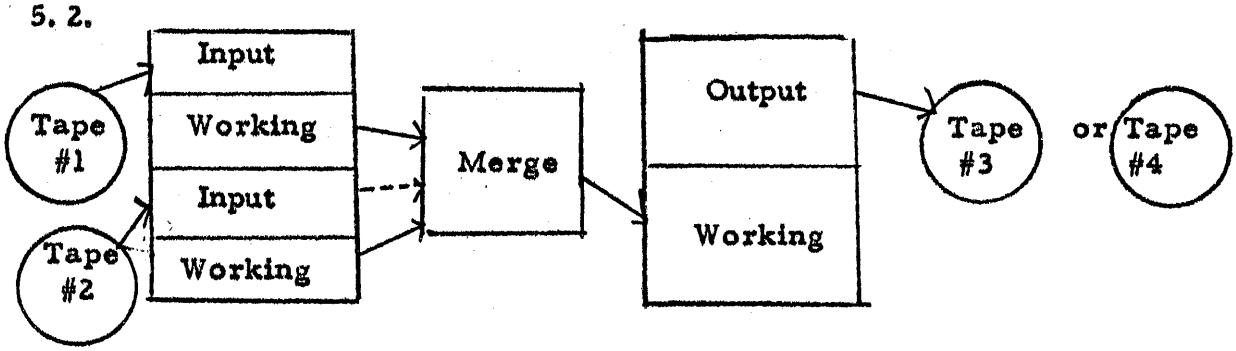
The use of memory is at the discretion of the programmer. However, to give an example, working areas for internal binary merge sorting and tape sorting are shown under 5.1. and 5.2.

5.1




These are functional areas and are not intended to be fixed areas of memory.

5.2.




The statement in 5.1. applies to this flow diagram with the addition that tapes 1 and 2 and tapes 3 and 4 are functional interchanged also.

WAH/jh W. A. Hunt

HARVEST REPORT #11, Appendix #1

Subject: Binary Merge Sorting

By: W. A. Hunt

Date: January 22, 1957

January 22, 1957

HARVEST REPORT #11, Appendix #1

Subject: Binary Merge Sorting

The following suggests a method of accomplishing a four (or eight, etc.) way merge using the basic concepts of the three continuous stream registers postulated in Special Report #6.

1. Assumptions - The internal data flow, control information required, and assumptions are the same as Special Report #6 with the exception that four additional high speed tape units are required, making a total of eight high speed tape units.

2. The method is to have four input and four output tapes (only one input and one output tape running at any given time) transferring information to and from the large memories.

The four input tapes are treated as two pairs of tapes for processing. For purposes of clarity the input tapes will be called, respectively, Tapes A, B, C, and D. Tapes A and B will be considered a pair and likewise Tapes C and D.

The operations are:

1. Read in parts of the A, B, C, and D blocks into memory. (A block is defined in 1. 7. of Special Report #6.)

2. Merge A and B into AB.

3. Merge C and D into CD.

4. Merge AB and CD into ABCD.

5. Write ABCD on one of the four output tapes.

Advantages:

1. Sorting times are less because the number of tape passes are generally halved; however, one additional tape pass may be required.

2. For items of about two words or less in length the four way merge should continue to be tape limited.

3. Only one input and one output tape transfer information simultaneously.

Disadvantages:

1. Eight tape units are required instead of four for a two way merge.

2. About 2.5 times as many index quantities are required to control the flow of data as compared to the two way merge. (This should be of no concern in this machine.)

Summary:

By passing each item through the Continuous Stream Registers twice for each tape pass, a four way merge can be achieved at about the same rate as a two way merge for items of two words or less. For items of greater than two words in length, the four way merge should give some speed increase varying with the parameters involved.

W. A. Hunt

WAH/jh

W. A. Hunt