J. Cocke 1957

DATA ORDERING BY CONTROL WORD TECHNIQUES

by

G. A. Blaauw
7000 Series Planning

C. A. Scalzi
Engineering Planning

September 1, 1957

IBM Product Development Laboratory
Poughkeepsie, New York

This memorandum describes the use of control words in basic data ordering procedures, such as file merging and file maintenance. The memorandum is tutorial in nature. It is intended to serve as an introduction to subsequent memoranda which will describe further applications and evaluate the methods presented against present day procedures.

I BASIC INDEX TECHNIQUES

The stored program of any machine using the techniques to be described makes use of instructions and control words. In this presentation the 750 will be used as an example. However, the methods to be presented can be adapted to any computer with the same general control word concept. Instructions are assumed to be of the single address type. Control words are used in conjunction with instructions and may be indices, record words, or routine words. A control word is called an index when it is used with the address portion of an instruction to determine the effective address of that instruction. A control word may also be used to define a group of consecutive words in storage. A group of words so defined is called a record, and the control word used in this manner is called a record word. A control word may also be used in conjunction with branch instructions to accomplish subroutine control. When applied in this way, the control word is called a routine word. It should be understood that the three names given to control words are descriptive of their use. The format or layout of all these words is the same.

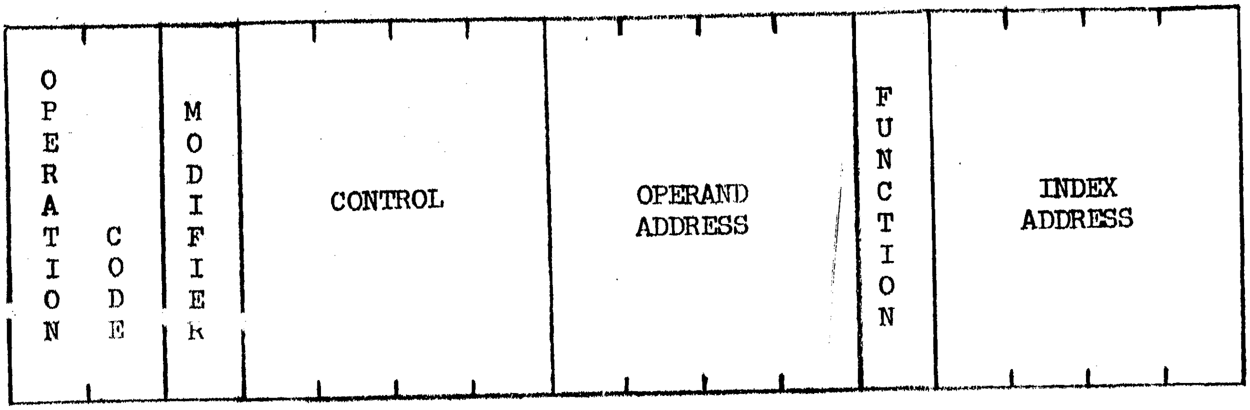
The format of a typical instruction is shown in Figure 1. Each instruction permits the use of an index word in conjunction with the address portion of the instruction. The address portion is known as the Operand. The address to which the instruction applies is determined by the index, the operand, and the index function of the instruction. This function specifies the manner in which the index and operand are to be used.

For example, when the function "index operand" is specified in an instruction, it causes the "working" address of the word addressed by the index portion of the instruction to be added to the operand portion of the instruction. This sum is used as the effective address of the instruction. When the "modify" function is specified, storage is addressed by the "working" address of the index word specified. The operand portion of the instruction is then added to the "working" address of the index, and the sum replaces the "working" address. Another function, called "reset", will be defined at a later point in this memorandum.

The format of a typical control word is shown in Figure 2. A control word is composed of four distinct parts. Three of these contain addresses which are called the working, end, and reset addresses. The fourth section contains the control condition which determines the use and operation of the control word. Some of the possible conditions will be described later.

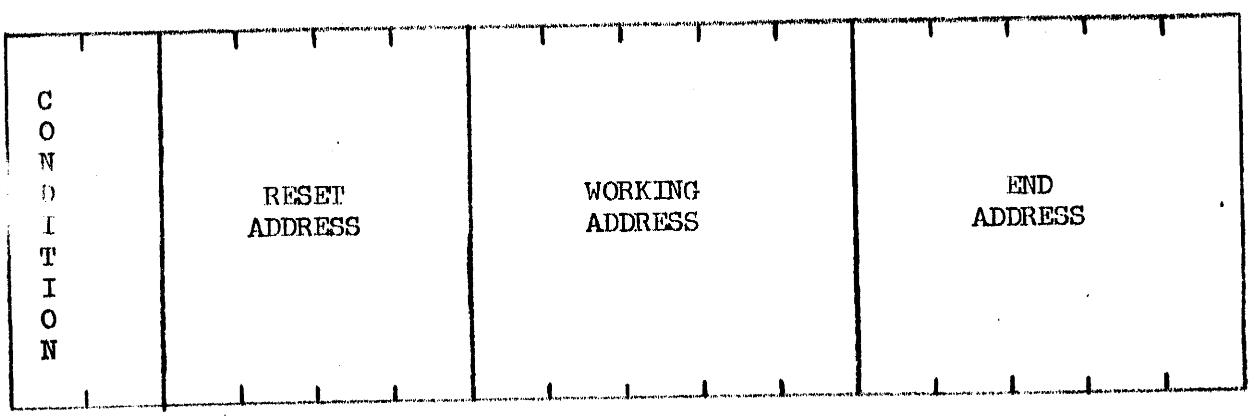
The working and end addresses are used to define a group of consecutive words in storage. They are the address of the first and last word respectively of the group. It is possible that this group of words may be only part of a record or that it may contain several records, depending upon the particular application. For convenience of description, however, a group of words so defined by a control word will be considered a record. This nomenclature does not restrict the generality of the methods which will be described.

The third address, the reset address of a control word, will be discussed in greater detail since its use makes possible the record handling techniques which are the subject of this memorandum.



INSTRUCTION WORD

FIGURE 1



CONTROL WORD

RESET ADDRESS

In contrast to the working and end addresses, which refer to the record they define, the reset address refers to another control word. This second control word is the next one to be used. In other words, a series of control words can be established in which the reset address of each control word refers to the address of the control word which is next in the series. When a control word is used with an instruction, a "reset index" function may be specified. The reset operation causes the current control word to be replaced by the next control word in the series. Subsequent instructions addressing this location will use the new control word. The following example illustrates this principle.

A series of control words is shown in Figure 3. Each control word is represented by a rectangle. The letter to the left of each rectangle represents the address of the control word in storage. Inside each rectangle the lower case letter is the reset address, and the capital letter represents the working and end address. Since the capital letter actually represents the definition of a record in storage, each record will be referred to by this letter in its control word.

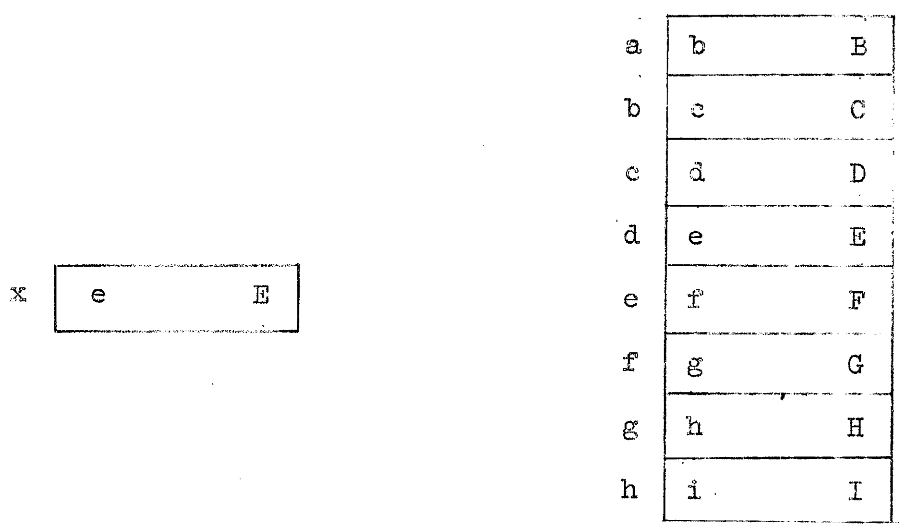


Figure 3

In Figure 3 the first control word is stored in location a. This word defines record B. Its reset address is b which means that the next control word in the series is stored in location b. The series of control words shown in the figure establishes a series or group of records. These record words need not be stored sequentially in storage since their logical order is dictated by the reset addresses. The records they define are independent in that they may have any desired length and may be placed anywhere in storage. It will be shown that the records in all likelihood are distributed in a random fashion throughout memory. This is called a "scattered" group of records. This ability to maintain scattered records and scattered control words is emphasized since it will be used extensively in the record handling techniques to be described.

On the left side of Figure 3 a working location, x, is shown. This location contains one of the control words in the series. During a processing operation the control word in x is used as an index in order to obtain or change the individual data of the record. At a given phase of the operation, x may contain e-E as shown in the Figure.

Once record E is processed, the computation may proceed to the next record which in this case is F. This is achieved by resetting location x. Since the control word in x has a reset address e, the contents of x will be replaced by the control word at e which is f-F. The processing of record F can now be performed using the same program as was used for record E since the instructions in this program refer for indexing to location x whose content is now f-F. Obviously this procedure can be continued for all the records in the group.

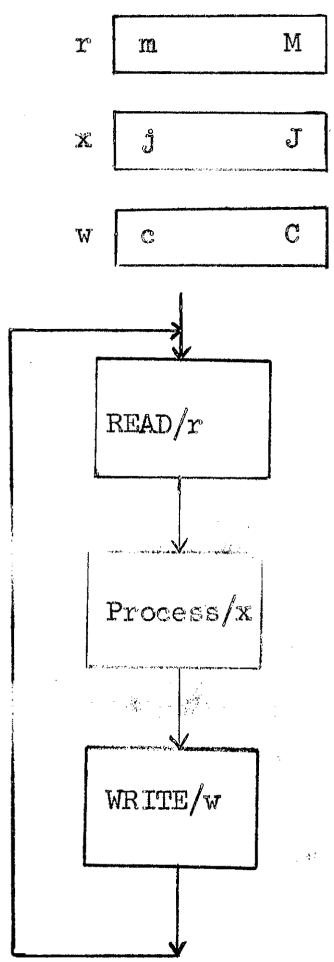
RESET OPERATION

The reset operation can be specified in two ways: as a "programmed reset" and as an "automatic reset." The programmed reset operation is specified by the index function of an instruction. This mode of resetting was used in the preceding example in order to reset the contents of location x. A transmit operation can be used to illustrate automatic reset. Transmit instructions are used prior to and following the processing of records. For example, they are used to read data from tape to storage, or to write data from storage on tape. In order to read the records B, C, D, etc., (defined by the control words in Figure 3) from tape to storage, a READ/CONTROL instruction is given. In this instruction a working location r is specified. At the start of this operation r should contain the record word b-B. The read operation will then proceed reading the first record from tape to the storage locations specified by record word b-B. When area B has been filled, the word in r is reset automatically, and b-B is replaced by c-C. The information now read from tape is stored in locations belonging to C. This process continues in the reading of records D, E, etc. The reset operations which are involved in this read operation are automatic. Only the original READ/CONTROL instruction is given. Reading proceeds by moving data from tape to the locations specified by the string of record words. The data from tape may be stored in record areas which are scattered throughout memory.

END CONDITION

When a record transmission is initiated as above, it will continue on its own. One READ/CONTROL instruction can read from tape to storage a series of records as explained above. It was shown how continuity is established by having each record word refer to the next in the series. It is also necessary to indicate when the process should be terminated. Record transmission is stopped after a record word has been used which has an "end condition." This end condition is one of the conditions which may be specified in a control word. The record described by the record word with an end condition will participate in the record transmission, but it will be the last one to be transmitted. After termination of transmission a final automatic reset takes place, initializing the working location for the next transmit operation.

In the following figures an end condition will be denoted by an asterisk. Thus, record words f-F*, k-K*, and a-A* in Figure 4 have an end condition.



a	b	B
b	c	C
c	d	D
d	e	E
e	f	F*
f	g	G
g	h	H
h	i	I
i	j	J
j	k	K*
k	l	L
l	m	M
m	n	N
n	o	O
o	a	A*

Figure 4

II SCATTER READ-COMPUTE-WRITE

An example of simultaneous read, compute, and write is illustrated in Figure 4. Three working locations are provided for the control words being used at any particular time. Location r is specified in the read instruction, x in the instructions of the compute block of the program, and w in the write instruction.

Fifteen record words are used which refer to each other by means of their reset addresses. They form a loop since the last in the series refers to the first. This series is broken into three groups of five each by the presence of end conditions in record words f-F*, k-K*, and a-A*.

A block diagram of the program to be described is shown at the lower left of Figure 4. The read and write blocks represent one instruction each. The process block may represent a series of instructions, some of which will refer to working location x as an index.

At the start of a cycle, records have been read from tape into the locations specified by G, H, I, J and K. These records are available for processing. The records in locations B, C, D, E and F have been processed and must now be written out on tape. The records in L, M, N, O and A were written on tape during the last cycle, and these record areas are now available as an input area during this cycle. The contents of working locations r, x, and w are shown in Figure 4 at some arbitrary time during this cycle. At the end of reading, r will contain b-B. When the processing is complete, x will contain l-L. At the end of writing, the contents of w will be g-G.

During the next cycle the records G, H, I, J and K must be written on tape since they were processed during the preceding cycle. This write operation requires that w (used as a record word by the write instruction) contains g-G at the start of the new cycle. Since g-G is already in w, no further programming is required. Similarly, record areas B, C, D, E and F are now available as the input area since their contents were written on tape during the last cycle. Location r (used as a record word by the read instruction) should contain b-B at the start of this cycle. Record areas L, M, N, O and A have been filled during the last cycle by reading from tape. These are the next records to be processed, and location x (used as an index by the process instructions) should contain l-L. Thus, it can be seen that at the close of each iteration, working locations r, x, and w are automatically initialized for the next iteration. As the program is continued through succeeding iterations, r, x, and w will always contain the proper control words. There is no need to reload or change the contents of these working locations.

The alphabetic notation is used to indicate that the actual records and record words may be scattered throughout storage. In the given example the records and the control words could be stored consecutively, but they need not be. Even if they are, the scatter feature will be used in the transition from a-A to b-B. This transition is an unavoidable break in the consecutive ordering. The scatter feature will be used more extensively in succeeding examples.

SUMMARY

The preceding example has shown the basic indexing technique as applied to reading, processing, and writing records. The features of this operation may be summarized as follows:

1. Each record is read into storage once.

   After the initial transmittal from tape to storage, no further record transmission takes place. There may be transmission of items in the record during the processing operation, but the record as a whole need not be moved in storage.

2. Each record is defined in storage by a control word.

   The control word may be used as a record word in reading and writing, and as an index in the processing section of the program. The use of a record word in a write instruction is equivalent to transmission of the record to an output area, and this procedure removes the necessity for actual transmission. This procedure leads to a significant saving since it is more economical to move one control word to a working location than it is to move all words of the record defined by that control word.

3. Record words generally form strings whose continuity is provided through the reset addresses of these words.

   This feature allows the convenient handling of groups of records. Both the string of record words and the records themselves may be stored in memory in scattered fashion. The method described also provides significant savings when grouped records are being handled. In order to write a group of records whose record words form a string, it is necessary only to move the record word describing the first record of the group to the working location used by the write instruction. As the write operation progresses, the record words describing the subsequent records replace the contents of the working location in the order established by the reset addresses. The operation proceeds automatically as each area is emptied until an end condition is sensed in one of the control words at which time writing will cease.

4. The use of an end condition terminates a string of record words.

Individual strings may refer to each other so that one string of record words follows another. When operation on one string is terminated, the working location used is automatically reset to the first record word in the next string. An additional saving in programming effort is gained by using this procedure since it makes unnecessary the transmission of the first record word of a string.

It should be noted that in the above example records are retained in their original order throughout the entire process. They are processed and written on output tape in the same order which they had on input tape. In the following examples it will be shown how records can be ordered, inserted, or deleted by further expanding the above-mentioned record handling techniques.

III MERGING

There are several applications in which it is desirable to combine two groups of records into one single group. A prominent example is the merging process used in sorting. Figure 5 will illustrate the merging process to be discussed. Basically this figure is a double image of Figure 3. Instead of one group of records with a working location, there are two groups of records each with its own working location, locations x and y. A third working location, z, has been added since it is necessary in this merging procedure.

The two groups of records will be merged on the basis of criteria specified by the programmer. The order resulting from these criteria has been indicated by the alphabetic order of the capital letters which denote the records. Each group is ordered within itself in ascending order. It is desired to merge the two groups into one which is also ordered in ascending sequence.

The merge is achieved by combining the two strings of control words into one new series which has the required order. Thus, when the records are written on tape using these record words, they will be written in sequential order.

Working location x contains one of the record words of group I and working location y contains one of the record words of group II. At the start of the merge x contains 1-C while the content of y is 4-D as shown in Figure 5. Working location z is not used to store a record word. It contains an address and will be used as an index word. The address in z is the location at which the next record control word should be stored. In the given example it is desired to store the record word defining the first record of the merged group in a. Because of this requirement a is placed in z at the start of the operation.

GROUP I RECORD WORDS

k	l	C
l	m	G
m	n	H
n	o	K
o	p	M
p	q	N
q	r	Q
r	s	R
s	t	S*
t	u	J

GROUP II RECORD WORDS

a	b	D
b	c	E
c	d	F
d	e	I
e	f	L
f	g	O
g	h	P*
h	i	A

CONTENTS OF WORKING LOCATIONS DURING SUCCESSIVE PHASES

Start	x		y		z
1.	l	C	b	D	a
2.	m	G	b	D	l
3.	m	G	c	E	b
4.	m	G	d	F	c
5.	m	G	e	I	d
6.	n	H	e	I	m
7.	o	K	e	I	n
8.	o	K	f	L	e
9.	p	M	f	L	o
10.	p	M	g	O	f
11.	q	N	g	O	p
12.	r	Q	g	O	q
13.	r	Q	h	P*	g
14.	r	Q	i	A	h
15.	s	R	i	A	r
16.	t	S*	i	A	s
17.	u	J	i	A	t

Figure 5

The procedure to be followed in the merge is outlined below.

a. /x * low compared to /y

1. x to /z

2. reset address of x to z

3. reset x

b. /y low compared to /x

1. y to /z

2. reset address of y to z

3. reset y

Figure 5 shows the initial contents of x, y, and z in the corresponding boxes. The subsequent contents of each of these working locations is shown below these boxes. These entries illustrate how the merge proceeds step by step. The records in the two groups remain in the same locations throughout. Only the corresponding control words are transmitted internally.

As the merge starts, the control field of record C is compared to that of record D. Since C is low, 1-C should be the first in the new series of record words. Record word 1-C is moved from working location x to location a. This transmission destroys b-D, the original contents of a. However, b-D has previously been placed in working location y. The reset address of 1-C which is 1 is now placed in z to indicate the location at which the next control word in the new series must be stored. The reason for this becomes apparent if one looks ahead to the output phase of the problem. When the first phase of the merge is complete, the merged records are to be written on the output tape. Writing will begin under control of the record word at a, which is 1-C. When it automatically resets, it will be replaced by the record word at 1. Therefore, the control word describing the second record in the sequence must be stored in 1.

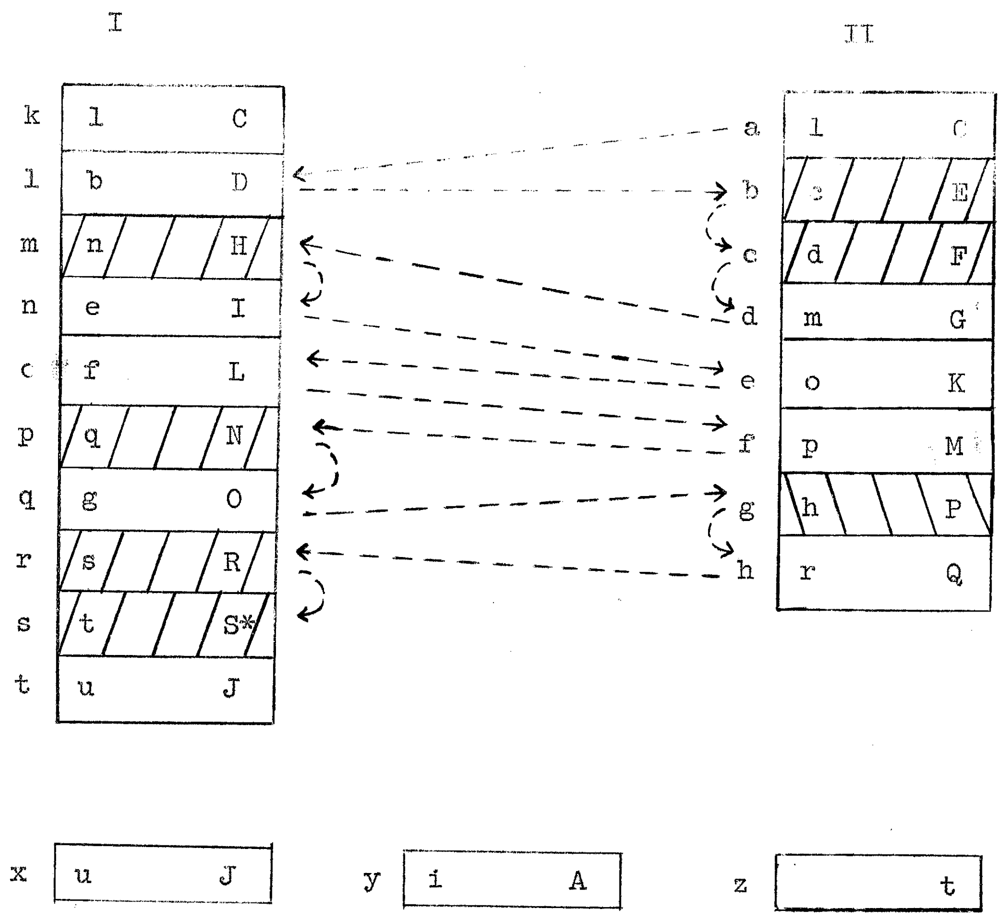
Since the record word from x has been handled, it is free to receive the next control word of group I. Location x is, therefore, reset, and its contents are replaced by the contents of 1, m-G. It should be noted that 1 is now free to receive the next record word in the series. The content of working location y remains unchanged. The new situation is found on line 2 of Figure 5. Now records G and D are compared and D is found low. Therefore,

*The notation "/x" represents the word obtained under control of x record word b-D should be second in the new series. It is placed in l (by using z as an index), replacing m-G which is now located in x. The reset address b of b-D is placed in z, and y is reset so that it now contains c-E. This leaves location b free to receive the next record word in the series. The contents of x remain unchanged. The new situation is shown on line 3.

This process continues as indicated by the listings of the step-by-step contents of x, y, and z in the figure. An arrow is used to indicate the record word which is to be placed next in the string as a result of the compare at that step of the merge. The process will terminate when one of the two groups is exhausted, i.e., when a record word containing an end condition has been stored as a result of the merge. In the given example the series of record words defining group II runs out when record word h-P* is stored at step 13. The next record word, i-A, in this series can not be merged until a new group I is encountered. The remaining record words in this part of group I must first be stored before merging is again started. Therefore, record words r-Q, s-R, and t-S* are now stored. In order to do this, it is only necessary to move r-Q to location h since the remaining record words in group I already reset to each other in the proper sequence. Following this transmittal, location x is continually reset until it contains a record word containing an end condition, in this case t-S*. The reset address, t, is placed in z, and x is reset to u-J in order to initialize the working locations for the next phase of the merge. When h-P* was stored, its end condition was removed since it no longer is the last record word of a series.

Figure 6 illustrates the situation as it exists after the first phase of the merge has been completed. The record words are still shown in two sets in order to maintain a correspondence with Figure 5. It should be pointed out, however, that these two sets actually form one merged group. The continuity and ordering of this group can be followed by means of the reset addresses. This is illustrated by the dotted line arrow between the words.

The record words which are shaded in the figure are those which are in the same storage locations after the merge as they occupied before the processing began. It is possible to arrange the merge program in such a way that advantage is taken of the fact that some control words need not be moved. Those record words which remain in their original locations are those which are in the same sequence before and after the merge. For example, record word c-E follows b-D before and after the merge and is, therefore, not displaced from location l.



I
II
k
l
m
n
c
p
q
r
s
t
l C
b D
n H
e I
f L
q N
g O
s R
t S*
u J
a
b
c
d
e
f
g
h
l C
E
d F
m G
o K
p M
h P
r Q
x
u J
y
i A
z
t

Figure 6

The shading of record word h-P is reversed to point out the fact that although it remains in its original location, it has received special treatment. The end condition, which was present at the start of the merge (see Figure 5), has been removed. This alteration of the control word, accomplished by programming, may take place in the working location y. The record word must then be placed back into g, its proper location in storage. Removing the end condition can be accomplished simultaneously with testing for its presence which is necessary in order to detect the last control word of a series.

Record word 1-C appears in Figure 6 in two locations, a and k. Its presence in location a marks it as the first record word of the new string produced by the merge. Its presence in k is redundant since k is not a member of the string. Location k is now an available location in memory. In a continuing process, its content would have been replaced by another record word in a preceding phase.

Figure 7 shows the record words of the merged group in the sequence established by their reset addresses. It should be emphasized that these words are not found consecutively in memory in the way they are shown in the figure. The order shown is the order of their use as established by the merge. The order is the same as the one depicted in Figure 6. It is seen that the record words will not be used in the alphabetic order of their storage locations. The original alphabetic order of the locations was used only to indicate an arbitrary numeric order at the start of the merge. Therefore, following the merge, the actual record word locations are just as random as they were before it began.

The series of instructions necessary for the two cases which occur in the merge process have been listed at the beginning of this section. In each case three operations are required. The two sets of three operations refer to two addresses each, either x and z or y and z. For a computer like the 750 which contains two full addresses in an instruction, it is possible to define one macro operation which does all three in one. The "control" or "second" address would contain z; the operand address, which is indexable, would contain x or y.

a	l	C
l	b	D
b	c	E
c	d	F
d	m	G
m	n	H
n	e	I
e	o	K
o	f	L
f	p	M
p	q	N
q	g	O
g	h	P
h	r	Q
r	s	R
s	t	S*
t	u	J

x	u	J
y	i	A
z		t

Figure 7

IV INSERTION AND DELETION OF RECORDS

In many applications one group of records in memory is a part of a main or master file. A second group is part of a secondary or detail file. The information in the detail file is used to update that of the master file. Updating may involve a change in a master record, the insertion of a new record in the master file, or the deletion of a record from that file. It is assumed that the records in the master and detail files are in proper sequence in their respective files. Records of the master file are called active when a detail record exists which corresponds to the master. All other records of the master file are termed inactive. When there are many more inactive records than active records, the file is said to have a low activity.

A simple example, consisting of one master file and one detail file, will be considered. The number of words in each record is the same in the detail file as in the master file. The master records will be read, processed, and written. The detail records need only be read and processed.

The record words and the necessary working locations for this application are shown in Figure 8. A group size of three records is assumed although this is purely an arbitrary choice. A small group size has been chosen to facilitate the presentation of an example. The group size is not limited by this method but rather will be determined by the characteristics of the particular application. The group of record words on the left of the figure are those which define the master records. This group forms a loop which is divided into four distinct series by the presence of an end condition in the record words d-D*, g-G*, j-J*, and a-A*. Three of these series are necessary in order to run the problem with overlapped read, write, and compute. The fourth can be considered a spare compute area. This string is necessary to take care of the possibility of a disproportionate number of deletions.

The record words shown at the upper right are those which define the detail records. This loop is divided into three strings by the end condition in p-P*, s-S*, and n-N*. Two of these strings are used for simultaneous read and compute of the detail records. In the example to be presented, it is assumed that the detail records need not be written. This assumption is arbitrary; there is nothing in the method which forces any such restriction. The third string is a spare area to cover the possibility of a disproportionate number of insertions.

Locations mr, mx, and mw contain the control words describing the master records being read, processed, and written respectively. Locations dr and dx contain the control words describing the detail records being read and processed respectively. The master working location, mz, and the detail working location, dz, are used in the manipulation of the control words during the processing. These two locations contain addresses and are used as indices and not as control words. The contents of all these locations are shown in the figure at some arbitrary time. A count is kept of the masters in the updated file so that when the group size is reached this group can be written on the output tape. Likewise, a count is kept of the processed detail records to determine when the next group must be read from the detail file tape.

Insertions and deletions are handled in such a way that those control words describing deleted master records become part of the detail record series and those which define detail records which are insertions become part of the master record string. In effect these strings are cut when insertions or deletions are encountered and tied into sections of the other string. A record word may be part of either string as the file maintenance progresses, depending on whether the last record stored in the memory area it defines was an insertion, deletion, active, or inactive record. The detail string and the master string trade memory areas for most economical operation. This is accomplished only by the manipulation of the control words. The records themselves are never moved from their original locations in storage. The procedure will be illustrated by the example to be discussed.

At the start, location mx contains b-B defining the first master. The record word n-N, describing the first detail, is located in dx. In mz is the address a, the location of the record word in mx, and in dz is the address m, the location of the record word in dx. These two locations contain the addresses to which record words will be moved if a swap of areas is dictated. They are used as indexes in the transmission of the record words.

Each case will be taken separately to illustrate the procedure.

A. Inactive Masters

In the case of an inactive master, it is desired to obtain the next master record and to leave the detail record unchanged as outlined below:

1. Reset address of mx to mz

2. Reset mx

The reset address of the present master is transmitted to mz, the master record word working location mx is reset to the record word defining the next master record, and the master record counter is advanced. The master string is not broken by this procedure. Transmitting the reset address to mz in effect updates that location for the next step in the processing. The detail series is not involved in the case of an inactive master record so no change is made in dx or dz.

MASTER RCW's

a	b	B
b	c	C
c	d	D*
d	e	E
e	f	F
f	g	G*
g	h	H
h	i	I
i	j	J*
j	k	K
k	l	L
l	a	A*

DETAIL RCW's

m	n	N
n	o	O
o	p	P*
p	q	Q
q	r	R
r	s	S*
s	t	T
t	u	U
u	n	N*

mr	i	I
mx	c	C
mw	l	L
mz		b

dr	u	U
dx	p	P*
dz		o

Figure 8

B. Active Masters

In the case of an active master, it is desired to obtain both a new master record and a new detail record as outlined below:

1. Reset address of mx to mz

2. Reset address of dx to dz

3. Reset mx

4. Reset dx

The working locations mz and dz are each updated. This is done by placing the reset address of the current master record word in mz and the reset address of the current detail record word in dz. Both mx and dx are reset to obtain the next master word and the next detail word respectively. The master counter and the detail counter are each advanced.

The strings of Figure 8 will be used to illustrate the method being described. The contents of the working locations at the start of the process are shown on line 1 of Figure 9.

	mx		dx		mz	dz
1.	b	B	n	N	a	m
2.	c	C	n	N	b	m
3.	d	D*	o	O	c	n

Figure 9

Assume that master record B is found to be an inactive record when compared to detail record N. The reset address b of b-B is placed in mz, mx is reset to c-C, and the master record counter is advanced. The contents of the working locations at this step in the problem is shown on line 2 of the figure.

Assume further that detail record N specifies a modification in master record C, i.e., master record C is an active record. The reset address c of c-C is placed in mz and the reset address n of n-N is placed in dz. Locations mx and dx are reset to d-D and o-O respectively. Line 3 of the figure indicates the situation at this step of the problem. The master and detail counters are each advanced.

C. Insertion

In the case of an insertion, the following procedure is followed:

1. dx to /mz

2. Reset address of dx to mz

3. Reset dx

4. mx to /mz

5. dx to /dz

The detail record word in dx is transmitted to the location specified by mz using mz as an index word. As a result this detail word becomes part of the master string. The reset address of this word is a location which is occupied by a member of the detail string. The next master record word, therefore, must be placed in that location. For this reason the reset address of this detail is placed in mz which dictates the placement of the master record words. Then dx is reset to obtain the next detail word. This makes its location available to receive the next record word in the master series. The master record word in mx is transmitted to that location using mz as an index. The reset address of that word is the location of the following word in the original master string. Thus, the master series is returned to its original string and remains unbroken.

Because the master string has taken over a location which was part of the original detail string, that series is now broken. The new detail word in dx is transmitted to the location specified by dz in order to mend the break. In effect this detail word is moved up in the detail series in order to fill the gap which was left when the detail describing the insertion was moved into the master string. Its reset address will close the detail string around the location taken over by the master series. Thus, through the manipulation of the control words the insertion has been made and both strings are intact. Finally, the master counter is advanced.

The above procedure will be illustrated by continuing the example used to illustrate the processing of inactive and active records. Figure 10 is a continuation of Figure 9. Line 3 shows the contents of the working locations up to this point.

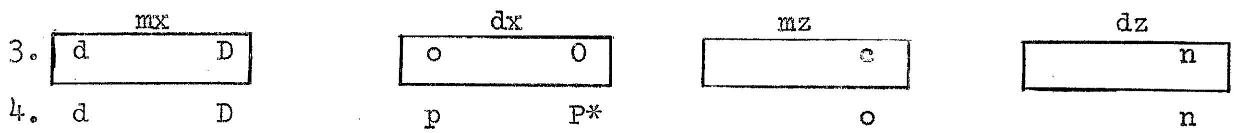


Figure 10

Assume that the detail record, O, is found to be an insertion when matched with master record D. This means that o-O should precede d-D in the new master series. It is, therefore, transmitted to c, the location specified by mz. In the output phase, when O is written and its control word resets automatically, it will be replaced by the contents of location O since that is the reset address of o-O. For this reason the next record word in the master series must be found in o. The reset address o of o-O is placed in mz which dictates the placement of the next master record word. Location dx is reset to obtain the detail word now occupying location o (p-P*). The word in mx (d-D) is transmitted to location o using mz as an index. The reset address d of d-D returns the master series to its original string. The detail string is now broken, however, and must be mended. The mending is accomplished by transmitting the word in dx (p-P*) to location n under control of dz. Finally, the master counter is advanced.

The situation at this point is shown on line 4 of Figure 10. The pertinent parts of the two strings up to this point are shown in Figure 11. The broken line shows the new sequence of the detail string and the solid line the new sequence of the master string.

Figure 11

It should be noted that record word o-O* is shown in the figure with an end condition. This has been added since O became the third processed record in the revised master string. Since group size has been reached, a write master operation is initiated before continuing the processing. The detail counter stands at one so no new details need be read at this point.

In the insertion case described above, the detail is placed before the current master record and a new detail record is obtained. In this procedure the operation continues with the current master, and no further reference is made to the inserted detail record. It may be desired, however, to follow a procedure in which the inserted detail remains available for further action against following details. In this case, action upon the current master is delayed until processing of the inserted detail is finished. An outline of this mode of operation is as follows:

1. Reset address of dx to y
2. Reset dx
3. mx to /y
4. /dz to mx
5. mx to /mz
6. dx to /dz

The following remarks may help in the understanding of this process: A new working location y is used as temporary storage for the address of the next detail record word. After the next detail record word is obtained in dx, the current master record word is placed following the original detail record word. Next the original detail record word is placed in the master working location. Finally, both master and detail string are mended by the appropriate store operations.

D. Deletion

In the case of a deletion, the next detail is obtained. The current master is stored between the old detail and the new detail. A new master record is obtained. The procedure to be followed is the same as that followed for an insertion, except that the functions of master and details are reversed and the operation is preceded by obtaining a new detail record.

This process is outlined below:

1. Reset address of dx to dz
2. Reset dx
3. mx to /dz
4. Reset address of mx to dz
5. Reset mx
6. dx to /dz
7. mx to /mz

The reset address of the detail record word defining the record to be deleted is placed in dz. A new detail record word is obtained by resetting the detail record word. The master record word is then transmitted using dz as an index. In other words, the record word describing the master to be deleted is made part of the detail string following the record word of the detail which specified the deletion. The reset address of this word is a location which is occupied by a member of the master string. The next detail record word, now in dx, therefore, must be placed in that location. For this reason the reset address of the word in mx must be placed in dz which dictates the placement of the detail record words. Following this, location mx is reset to obtain the master record word which was stored in the location which is being taken over by the new detail series. This makes that location available to receive the next record word in the detail series. The detail record word in dx is transmitted to that location using dz as an index. The reset address of this detail word is the location of the next word in sequence in the original detail string. Thus, the continuity of the detail string is restored. It should be noted that the record word defining the detail record which specified the deletion remains in its original location.

Because the detail string has taken over a location which was originally part of the master string, there is a break in the master string. The new master word in mx is transmitted to the location specified by mz in order to mend the break. In effect this master word is moved up in the master series to fill the gap which was left when the record word describing the deleted master was moved into the detail string. Its reset address will close the master string around the location which was taken over by the detail series. Thus, through manipulation of control words, the deletion has been made and both strings remain intact. The detail counter is advanced twice (once for the deleted master and once for the detail specifying the deletion).

The above procedure will be illustrated by continuing the example used in the section describing insertions. Figure 12 is a continuation of Figure 10. Line 4 shows the contents of the working locations up to this point.

	mx	dx	mz	dz
4.	d D	p P*	o	n
5.	e E	q Q	o	p d

Figure 12

Assume that detail record P specifies that record D is to be deleted from the master file. Record word d-D must be removed from the master series. It can, however, be used in the new detail string. The reset address p of the word in dx (p-P*) is transmitted to dz and dx is reset to q-Q. Then d-D, the word describing the deleted master, is transmitted to location p (using dz as an index). Thus d-D becomes part of the detail series following p-P in the detail sequence. Looking ahead to the next time new details are read from tape using the new detail string, it can be seen that, when record area D has been used, its record word d-D will reset automatically and be replaced by the contents of d, the location specified by its reset address. The next detail word in the series must, therefore, be found in d. For this reason d, the reset address of the word in mx, is placed in dz. Working location mx is reset and its contents are replaced by e-E. Record word q-Q is transmitted to location d under control of dz. The reset address q of q-Q returns the detail series to its original string. The master string is broken at this point, however, and must be mended around location d by transmitting the word in mx (e-E) to location o under control of mz. Since the reset address e of e-E is part of the original master series, that string has now been mended around location d. The detail counter is advanced twice.

The situation at this point is shown on line 5 of Figure 12. The parts of the two strings which have been processed up to this point are shown in Figure 13. The broken line shows the sequence of the revised detail string and the solid line the sequence of the revised master series.

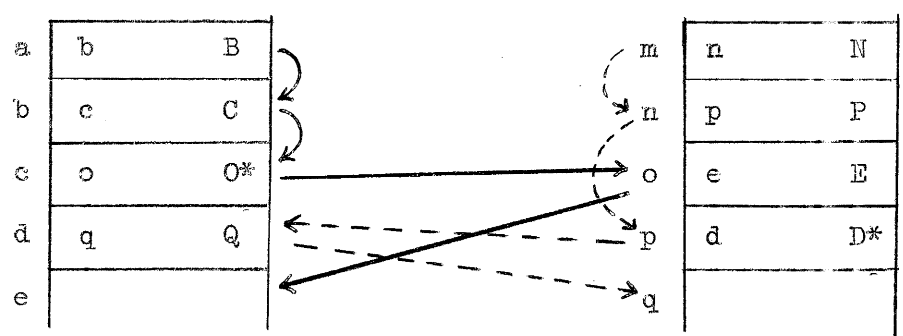


Figure 13

It should be noted that the end condition has been removed from detail record word p-P and added to d-D*. This has been done since D is now the third record area in the revised detail string. Since group size has been reached, a read detail operation is initiated before the processing continues. Since the master counter stands at zero, no master input-output operation need be initiated at this point.

E. Control

When a large number of insertions occur and only few deletions, the number of control words in the master string increases. As a result, the spare area of the master string increases in size, and the spare area of the detail string decreases in size. Conversely a large number of deletions would decrease the master spare area and increase the detail spare area. When one of the spare areas is empty, the other area has a double capacity. At that time a corrective step should be taken by which the number of control words in the two areas is made equal again. This operation consists in inserting or deleting a block of record words. It is handled in a way similar to the insertion or deletion of a single record word. The operation is accomplished by storing the reset address of the last control word of the string to be moved. The check for the necessity of inserting record words in the master string should occur just prior to the reading of new detail records. The check for the necessity of inserting record words in the detail string should occur just prior to the reading of new master records. When such an insertion occurs in the master string, master records are read instead of details. When the insertion of record words is made in the detail string, details are read instead of masters.

During the process a count is kept of the masters and the details that have been treated. When the number of processed masters reaches group size, a write operation is initiated and the process continues into the next master string in the loop. When the number of processed details reaches group size, a read operation is initiated to obtain new details and the process continues into the next string in the detail loop of control words. As a result of the counting, end conditions may be removed from some control words and added to others depending on their sequence in the revised strings.

The specific procedures described can be further refined.

The insertions and deletions have been treated as isolated occurrences with low activity. In higher activity applications it may be desirable to combine the handling of an individual insertion or deletion with subsequent processing. As a result, the total amount of control word manipulation may be reduced to obtain an economy in processing at the expense of a somewhat more complicated program.

F. Summary

The procedure described, effectively handles low activity problems with insertions and deletions. The method employs simultaneous read, write, and compute with scattered control.

The technique is basically simple. The object of the program is to maintain two strings of record words, one string containing those words describing the master records, and the other containing the record words defining the detail records. Sequence in the two strings is established by the reset addresses of the control words. When an insertion is encountered, its control word is made part of the master string and removed from the detail string. When a deletion is encountered, the record word describing the deleted master is made part of the detail series and removed from the master series.

Thus, both strings are updated as the processing dictates, and, since the input and output phases of the problem use these same strings, no movement of the records themselves is necessary. Throughout the entire procedure all records, both master and detail, remain in the memory areas into which they were originally read.

V CONCLUSION

In the preceding sections the use of control words for record handling has been explained and illustrated. File maintenance using this technique has been further explained by A. F. Deckman in Planning Evaluation File Memorandum #1. A forthcoming memorandum also by A. F. Deckman describes the use of control word techniques for internal sorting and merge sorting.

The use of control words has been described in detail in this memorandum in order to show that each of the basic record handling operations, merge, insert and delete, can be performed in a simple and direct way. Once the general method is understood, each particular operation proves to be a variation on the same theme. In fact, it can be shown that a merge operation is a special case of the general file maintenance operation, as far as record handling is concerned. Or, a file maintenance process may be considered two merges in one: Detail records are merged with master records (insertion) and master records are merged with detail records (deletion).

The specific programming techniques employed are of secondary importance compared to the general principle illustrated. The "reset" operation which has been assumed in this report and which is available in the 750, may very well be programmed in another machine. Conversely a machine code may be postulated which makes use of macro-instructions which combine several operations in one. Examples of this kind of instruction are given at the end of section III. Similar instructions can be defined for the operations listed in section IV.

While the particular instruction set is of minor importance, it should be emphasized that the availability of input-output grouping and distribution is a necessity for the methods described. This particular feature, which is also called scattered read and write cannot be replaced by programming except at the expense of a considerable amount of storage space and time. It is believed that the advantages of the record control techniques described in this report are such that grouping and distribution will be mandatory for the input-output systems of future machines and that the particular instruction set and indexing procedure of these machines should facilitate the handling of record control words.