4. SEQUENTIAL FILES

4.1 Description

A sequential file is one that is processed serially from the first record in file through to the last. All files can be processed sequentially regardless of any other use. A file that is used as an indexed random file could be used as a sequential file in another task for listing, reporting or statistical purposes. A log file containing changes to an on-line file is both generated and accessed later as a purely sequential file. This chapter discusses the characteristics and handling of 'sequential files' without regard to any other way of using them.

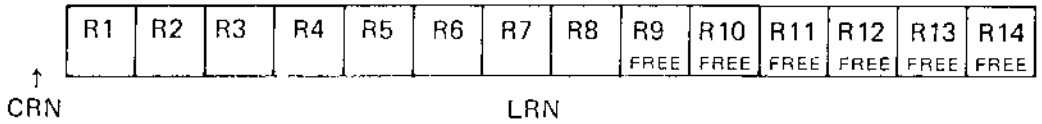

Figure 4.1 A sequential file

Consider the file shown in figure 4.1. For sequential processing all accesses to this file should be serial. When the file is opened the CRN is zero, and the LRN is R8, the last used record in the file. The sequential read instruction is issued and the first record in the file is read into the user's buffer. The CRN is 0 before the instruction and set to 1 after the instruction, thus pointing to the current record being processed. The next time a sequential read is issued, record R2 is read into the user's buffer and the CRN is updated to 2. The instruction makes no reference to a record number or to any particular record, it can only access the record after the one pointed to by the CRN.

The sequential write instruction acts in a slightly different manner. If a new record was being added to the file, it would be placed after R8 and the LRN would be updated in memory to R9, and so on for as many new records were being added to the file. The CRN is not used or updated by the sequential write instruction.

Suppose the file was to be updated and that the change was to record R6. The sequence of instructions for this are shown below:

1. Assign the file (CRN = 0, LRN = 8).
2. Read sequential until the relevant record is formed (CRN now = 6).
3. Process the data in the required record.
4. Sequential write the changed record. This will go to position 9 and the LRN is updated in memory to 9.
5. The 'old' version of the record still exists and will have to be deleted — 'get currency data file' is the instruction to access the CRN for the user program.
6. Random delete using the CRN as the logical record number. R6 is now set to FREE.
7. Close the file. The LRN will be updated on disk to 9.

If the file being processed is normally used as a random file or indexed random file, then *great care must be taken to ensure that the order of records is not changed*. Problems could arise in later processing when the record is being accessed by its logical record number.

4.2. Creating a sequential file

The rest of this chapter assumes that the file is being created specifically for the purpose of sequential processing. If it is to be used for random or indexed random processing refer to the following chapters.

The file can only be created on a TOSS formatted disk, i.e. one that has been created by the utility Create Volume. The creation of the file must be performed in two stages. Firstly, the file space is set up by the utility program Create File (CRF); see the Utilities Reference Manual, M08, or the TOSSUT utility in M11 DOS6800 Reference Manual, for a detailed description. Secondly, the actual records must be written to the file. A sequential file could contain records that are present in a predetermined sequence and any operations on the file should preferably be performed in a sequential manner.

1st Stage

The sequence of operations for CRF is as follows:

1. Call CRF utility under the TOSS utilities Monitor, or as a subroutine by the application, or via the DOS utility TOSSUT.
2. CRF requests a number of parameters. Most parameters can be given as required but for a sequential file, two parameters are obligatory.
   To 'File organization' reply 'S', and to 'Number of index files' reply '0'.
3. CRF searches the volume(s) for free extents large enough to hold the stated file size.
4. The file is set up with the required number of records, all of which contain space characters. Each record is set to 'FREE' status. The LRN is set to zero for this file.

2nd Stage

The actual records must be written to the file. For a purely sequential file, the records may be required in an order determined by the value of one of the data items. If the records have been prepared off-line on punched cards, or cassette, it is possible they may not be in the 'correct' order, so the input file must be sorted according to the required key before the file is released for use, for details of the SORT utility see the Utilities Reference Manual, M08.

When the file is available for use, it can be processed with the sequential instructions or with the random instructions presuming that the record key used for random access can be directly related to the record's position in the file. The sequence of records in the file could be determined alphabetically like a name and address file but this key could not be reduced to a numerical value to give a logical record number. If the file is required to be accessed by a task using the indexed random file, it must be set up according to the instructions described in Chapter 4. After set up, the file can then be used for sequential processes as well.

4.3 Instructions

A definitive description of these instructions is contained in the relevant language reference manuals, either Assembler Programmer's Reference Manual, M06, or CREDIT Programmer's Reference Manual, M04.
These instructions are briefly:

ASSIGNING THE FILE

When the file is 'assigned', the file name is linked to a file code declared in the same task. If the file is to be used only by that task it must be assigned with the TC parameter = 1, or if the file is to be accessible by more than one task it must be assigned as a common file, with TC = 0.
After successful assignment, the file is available to the assigning task.

SEQUENTIAL READ

After the file has been assigned, the CRN for this file in this task is set to zero. Whenever this instruction is used the CRN is updated by one and that record number is read. It is not possible to specify a particular record or a previous record. The current record can be put under exclusive access if required. If the record is already under exclusive access to another task it will not be read.

SEQUENTIAL WRITE

The record will be written to the file immediately after the record pointed to by LRN. The LRN will be incremented by one to the last record written by sequential write.
If a system crash occurs during the run of this task, the new LRN will be lost but the new records still exist. These records could be read with a random read, deleted, then rewritten with a sequential write. The LRN will then be correct.
If the record status is 'free' it will be set to 'used'. If it is 'used' sequential write is not allowed. Random delete, see chapter 3 will have to be used to delete a 'used' record.
The CRN is not changed by sequential write.

CLOSING THE FILE

A 'close file' instruction is used to indicate that the file is no longer required by that task, the LRN is updated and saved on the volume. The close action applies only to the task that issued it, and the file is still available to other tasks that are using it at that time, unless the file was assigned as common, in which case the file is no longer available to any tasks.