Chapter 12
Sequential File Update

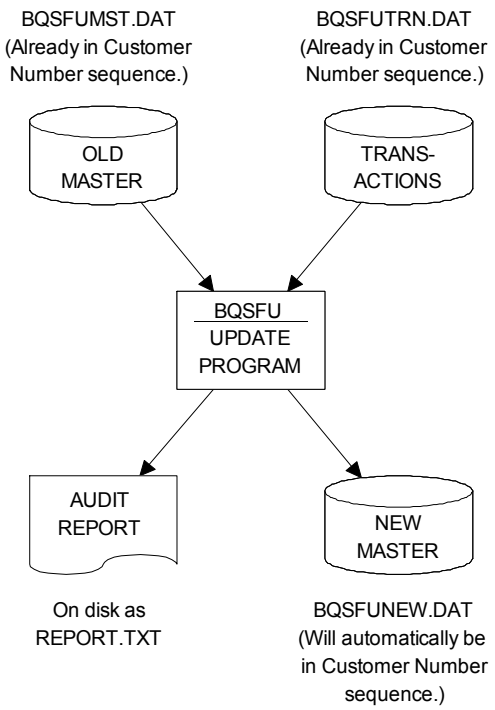
Objectives

Upon completion of this chapter you will be able to:

- List the three transaction types used in a sequential file update program,
- Given simulated `MASTER` and `TRANSACTION` files, determine the contents of the new `MASTER` file,
- Given simulated `MASTER` and `TRANSACTION` files, determine the contents of the audit report and reconcile the counts contained therein,
- Describe the processing required for `MASTER LOW`, `TRANSACTION LOW`, and `MATCH` conditions in a sequential file update program, and
- Design and code a program which uses update logic to apply transactions to a master file.

Introduction

In the preceding chapter we looked at the logic required to process two input files. Those programs created reports containing data from both files. The matching logic discussed there is common to many business applications. In this chapter we take that logic one step further, by designating one file as a master file and the other as a transaction file. Rather than simply create a report, the transactions are used to update the master file. This procedure, known as a **sequential file update**, is a very common application. Similar logic is used in other business applications as well such as merge/purge processing in direct marketing and applying payments to receivables in accounting.



The system flowchart for the sequential file update program is shown to the right.

Note: the purpose of this chapter is to illustrate update logic. "Perfect" data is presumed; that is, no field-level editing (such as to verify that a zip code is numeric) is shown.

To keep our example simple, the input (old) master file, transaction file, and output (new) master file all have the same record layout:

Field Nbr	Field Name	Description	Begins	Ends	Len	Format
1	NBR	Customer number	1	5	5	ZD
2	LNAME	Last name	6	15	10	CH
3	FNAME	First name	16	25	10	CH
4	ADDR	Street address	26	40	15	CH
5	CITY	City	41	50	10	CH
6	STATE	State	51	52	2	CH
7	ZIP	Zip code	53	57	5	CH
8		Unused	58	59	2	
9	ACD	Transaction code	60	60	1	A/C/D
10	CRLF	PC/370 Only	61	62	2	CR/LF

Our input master file, `BQSFUMST.DAT`, is as follows:

```
         1         2         3         4         5         6
12345678901234567890123456789012345678901234567890123456789012345678901
11224BINFORD   DAN       469 N 400 E    DESOTO    TX75115  A
12111ARIAS     IDA       4028 ELMO LOOP MERCED    CA95340  A
32555RYAN      RICHARD   914 FIFTH ST   NORMAL    IL61761  A
41499HILMER    DEBBIE    21175 FELIPA   BUENA PARKCA90620  A
55123JOSEPHSON PEGGY     248 MICHIGAN   JAMESTOWN NY14701  A
61626HAVLIK    CHERYL    551 WASHINGTON WHITTIER  CA90605  A
77271CARPENTER LOIS      326 BEACH      BERWYN    IL60650  A
81288BLACK     KATHY     618 S ANZA     PASADENA  CA91106  A
81997FOOTE     APRIL     635 BURNS      CAROL STRMIL60187  A
94993DIXSON    RICHARD   1021 BROWN     CHICAGO   IL60612  A
```

The transaction code in column 60 may be one of three types: `A` for adds, `C` for changes, or `D` for deletes. On the master file(s), the transaction code indicates the most recent processing applied to this record. This code is `A` for all of the above, indicating that no changes have taken place since these records were added. Our transaction file, `BQSFUTRN.DAT`, is as follows:

```
         1         2         3         4         5         6
12345678901234567890123456789012345678901234567890123456789012345678901
12111                  2211 APRICOT   MODESTO   CA95356  C
41499                                                    D
55123AMBROSE   FRANK     220 BARRETT    ROCKFORD  IL61103  A
61627QUALLS    CHERYL    201 N EIGHTH   WHITTIER  CA90605  C
81228                                                    D
82446AMICCI    BRUNO     17397 BARCELON CORVALLIS OR97330  A
```

The key field, customer number, is required on all transactions. For adds, all other fields would be provided as well. For changes, only those fields to be changed are filled in. Finally, for deletes, only the customer number and transaction code (`D`) are necessary.

All of the above transactions are *syntactically* correct, but as we will see, not all are *logically* correct.

Both input files must be in key sequence for the same reasons as discussed in the previous chapter. Given the nature of the processing, the output file will automatically be in key sequence as well. We begin by reading one record from the master file and one record from the transaction file. If the key fields do not match, then we process the record with the lower key. For example, the data shown above will be processed as follows:

OLD MASTER	TRANS KEY	TRANS TYPE	ACTION	NEW MASTER
11224	12111	C	Master Low - Write the master record to New Master and read the next Old Master record.	11224
12111	12111	C	Keys Equal - This is an attempt to change an existing record. Write changed record to New Master and read next record from both files.	12111
32555	41499	D	Master Low - Write the master record to New Master and read the next Old Master record.	32555
41499	41499	D	Keys Equal - This is an attempt to delete an existing record. Simply do not write the record to the New Master. Read the next record from both files.	
55123	55123	A	Keys Equal - This is an attempt to add a record with the same key as an existing record. This is an error. Nevertheless, the existing Master record is kept: write it to the New Master and read the next record from both files.	55123
61626	61627	C	Master Low - This would appear to be a transcription error, but an error nonetheless. Write the master record to the New Master and read the next Old Master record.	61626
77271	61627	C	Transaction Low - This is an attempt to change a record which does not exist. This is an error. Read the next Transaction record.	
77271	81228	D	Master Low - Write the master record to New Master and read the next Old Master record.	77271
81288	81228	D	Transaction Low - This would appear to be a transcription error, but an error nonetheless. Read the next Transaction record.	
81288	82446	A	Master Low - Write the master record to New Master and read the next Old Master record.	81288
81997	82446	A	Master Low - Write the master record to New Master and read the next Old Master record.	81997
94993	82446	A	Transaction Low - This is an attempt to add a record which does not exist. This is valid. Write the Transaction record to the New Master and read the next Transaction record.	82446
94993	EOF	n/a	Transaction file at EOF - Treat Old Master as Master Low - Write the record to the New Master and read the next Old Master record.	94993
EOF	EOF	n/a	Both files at EOF - Done.	

The output from the program is as follows. Of course, the report will usually be formatted more completely rather than showing a card-image of the records as was done here. But our purpose here is to demonstrate update logic and the output has intentionally been kept simple.

```
A:\MIN>bqsfu
BQSFU ... Begin execution
BQSFU ... Audit list on REPORT.TXT
BQSFU ... Normal end of program

A:\MIN>type report.txt
               Name & Address Update Program                  Page   1
                       Audit Listing

----+----1----+----2----+----3----+----4----+----5----+----6  MESSAGES

12111ARIAS     IDA       4028 ELMO LOOP MERCED    CA95340  A  BEFORE CHANGE
12111ARIAS     IDA       2211 APRICOT   MODESTO   CA95356  C  AFTER CHANGE

41499HILMER    DEBBIE    21175 FELIPA   BUENA PARKCA90620  A  RECORD DELETED

55123JOSEPHSON PEGGY     248 MICHIGAN   JAMESTOWN NY14701  A  RECORD ON FILE
55123AMBROSE   FRANK     220 BARRETT    ROCKFORD  IL61103  A  ADD UNSUCCESSFUL

61627QUALLS    CHERYL    201 N EIGHTH   WHITTIER  CA90605  C  CHNG NOT ON FILE

81228                                                     D  DLTE NOT ON FILE

82446AMICCI    BRUNO     17397 BARCELON CORVALLIS OR97330  A  ADD SUCCESSFUL

               Name & Address Update Program                  Page   2
                       Audit Listing

----+----1----+----2----+----3----+----4----+----5----+----6  MESSAGES

Transactions In                   6
Transactions Rejected             3
Old Masters In                   10
Old Masters Deleted               1
Old Masters Changed               1
New Masters Added                 1
New Masters Out                  10
```

The output (new) master file, `BQSFUNEW.DAT`, is as follows:

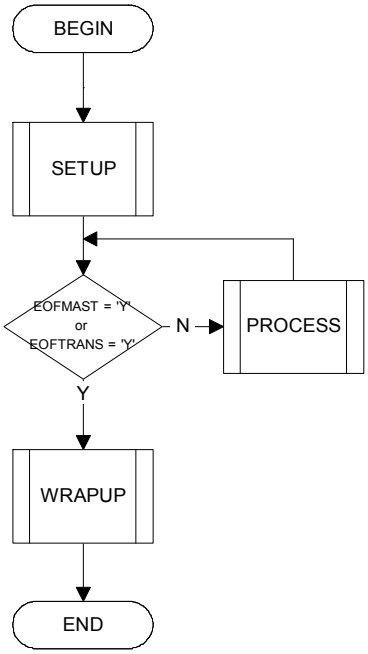
```
          1         2         3         4         5         6
12345678901234567890123456789012345678901234567890123456789012345678901234567890
11224BINFORD   DAN       469 N 400 E    DESOTO    TX75115  A
12111ARIAS     IDA       2211 APRICOT   MODESTO   CA95356  C
32555RYAN      RICHARD   914 FIFTH ST   NORMAL    IL61761  A
55123JOSEPHSON PEGGY     248 MICHIGAN   JAMESTOWN NY14701  A
61626HAVLIK    CHERYL    551 WASHINGTON WHITTIER  CA90605  A
77271CARPENTER LOIS      326 BEACH      BERWYN    IL60650  A
81288BLACK     KATHY     618 S ANZA     PASADENA  CA91106  A
81997FOOTE     APRIL     635 BURNS      CAROL STRMIL60187  A
82446AMICCI    BRUNO     17397 BARCELON CORVALLIS OR97330  A
94993DIXSON    RICHARD   1021 BROWN     CHICAGO   IL60612  A
```

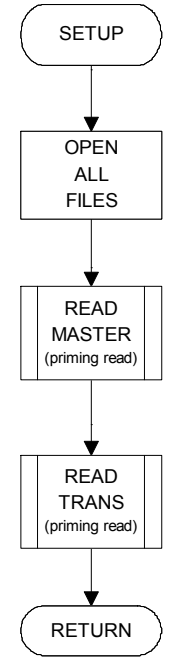
The program name is `BQSFU.MLC`. The program flowchart, assembler code, and notes follow.

The Mainline Structure

The mainline structure of the update program is the same as we saw in the previous chapter: we continue to process records until either the old master file or the transaction file is at EOF. The BAL code follows:

```
         BAL   R10,SETUP
MAIN     EQU   *
         CLI   EOFMAST,C'Y'
         BE    EOJ
         CLI   EOFTRANS,C'Y'
         BE    EOJ
         BAL   R10,PROCESS
         B     MAIN
EOJ      EQU   *
         BAL   R10,WRAPUP
```





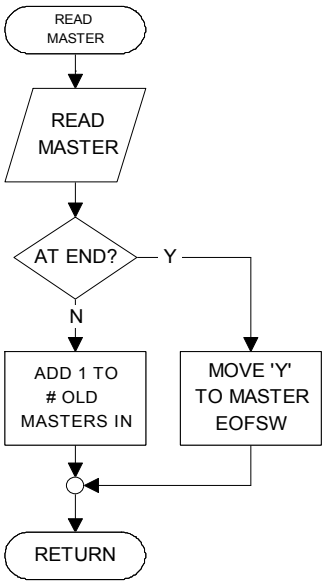
The `SETUP` Routine

Within the `SETUP` routine, we open all files and read the first record from the input master file and the transaction file:

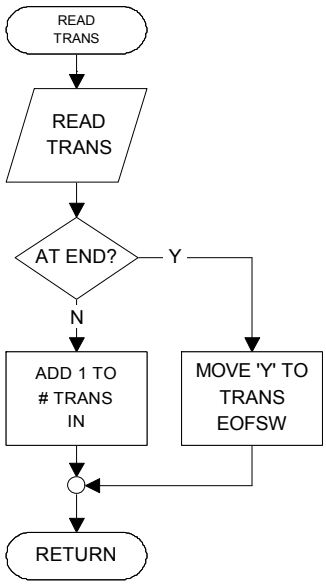
```
SETUP    EQU   *
         ST    R10,SVSETUP
         OI    MASTERIN+10,X'08'
         OI    TRANSIN+10,X'08'
         OI    MASTEROT+10,X'08'
         OI    REPORT+10,X'08'
         OPEN  MASTERIN
         OPEN  TRANSIN
         OPEN  MASTEROT
         OPEN  REPORT
         BAL   R10,READMST
         BAL   R10,READTRN
         L     R10,SVSETUP
         BR    R10
```

The READ Routines

We will have two READ routines: one for the input master file and one for the transaction file. In this program we have added the logic to include a record count for each file. These counts are incremented within the READ routines. They will be displayed within the WRAPUP routine.




```
READMST  EQU   *
         ST    R10,SVREADM
         GET   MASTERIN,IREC
         AP    #OLDIN,=P'1'
         B     READMX
ATENDMST EQU   *
         MVI   EOFMAST,C'Y'
READMX   EQU   *
         L     R10,SVREADM
         BR    R10
```

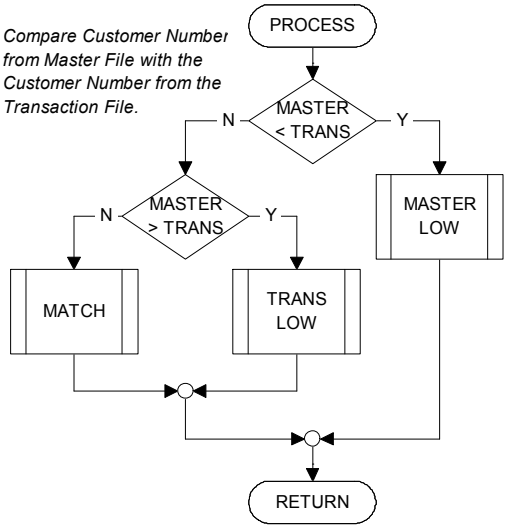



```
READTRN  EQU   *
         ST    R10,SVREADT
         GET   TRANSIN,TREC
         AP    #TRANSIN,=P'1'
         B     READTX
ATENDTRN EQU   *
         MVI   EOFTRANS,C'Y'
READTX   EQU   *
         L     R10,SVREADT
         BR    R10
```

The PROCESS Routine

At the time we perform the PROCESS routine, we have one record from the old master file and one record from the transaction file. We then compare the customer number from these two records. As in the previous chapter, there are three possible conditions as a result of this compare:

- the customer number on the MASTER is less than the customer number on the TRANSACTION,
- the customer number on the MASTER is greater than the customer number on the TRANSACTION, or
- the customer number on the MASTER is equal to the customer number on the TRANSACTION.



The first condition would indicate that we have a master for which there is no transaction: the master record would be kept as is.

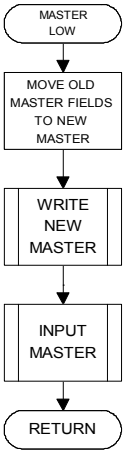
The second condition would indicate that we have a transaction for which there is no master. If the transaction type is an ADD, then this is valid. Otherwise we have an attempt to CHANGE or DELETE a record which does not exist.

The third condition would indicate that we have a transaction for an existing master. If the transaction type is CHANGE or DELETE, then this is valid. Otherwise, we have an attempt to ADD a record which already exists.

Each of these conditions will be handled in a separate routine.

```
PROCESS  EQU   *
         ST    R10,SVPROC
         CLC   INBR,TNBR
         BH    PROC2
         BL    PROC3
         BAL   R10,MATCH
         B     PROCESSX
PROC2    EQU   *
         BAL   R10,TRANSLOW
         B     PROCESSX
PROC3    EQU   *
         BAL   R10,MASTLOW
PROCESSX EQU   *
         L     R10,SVPROC
         BR    R10
```

Master Low

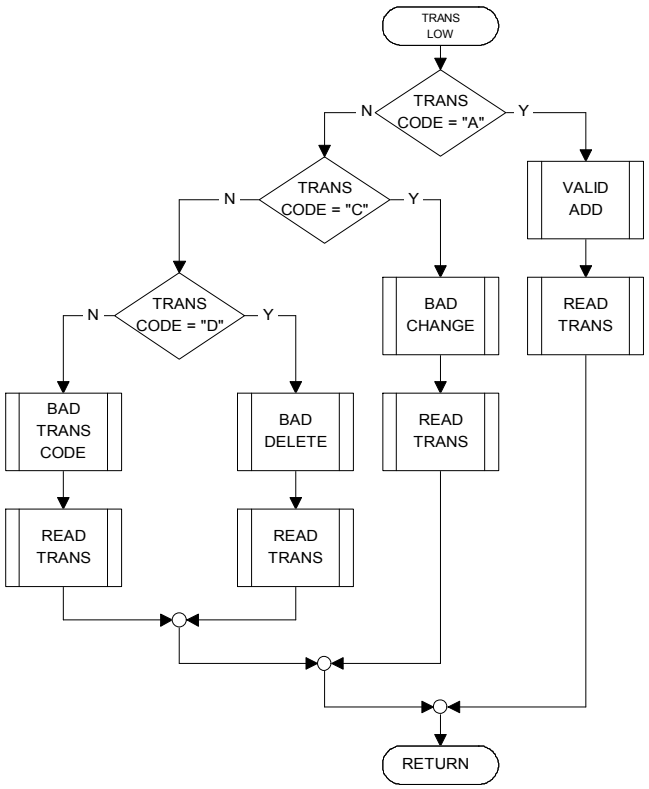


The "Master Low" condition indicates that we have a master record without a matching transaction. This is *not* an error: just write this record to the new master file and read the next old master file record.

```
MASTLOW  EQU   *
         ST    R10,SVMSTLOW
         MVC   OREC,IREC
         BAL   R10,WRITENEW
         BAL   R10,READMST
         L     R10,SVMSTLOW
         BR    R10
```

Transaction Low

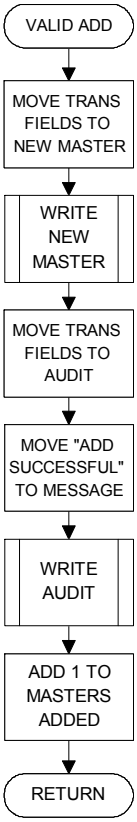
The "Trans Low" condition (aka "Master High") indicates we have a transaction without a matching master file record. Whether or not this is an error depends on the transaction type. If it is an `ADD`, then it is a valid transaction. If it is a `CHANGE` or `DELETE`, then it is an error: we cannot change or delete what isn't there.




```
TRANSLOW EQU   *
         ST    R10,SVTRNLOW
         CLI   TACD,ADD
         BE    TRANSLO2
         CLI   TACD,CHANGE
         BE    TRANSLO3
         CLI   TACD,DELETE
         BE    TRANSLO4
         BAL   R10,BADCODE
         BAL   R10,READTRN
         B     TRANSLOX
TRANSLO2 EQU   *
         BAL   R10,ADDIT
         BAL   R10,READTRN
         B     TRANSLOX
TRANSLO3 EQU   *
         BAL   R10,BADCHANG
         BAL   R10,READTRN
         B     TRANSLOX
TRANSLO4 EQU   *
         BAL   R10,BADDELET
         BAL   R10,READTRN
TRANSLOX EQU   *
         L     R10,SVTRNLOW
         BR    R10
```

Valid Add

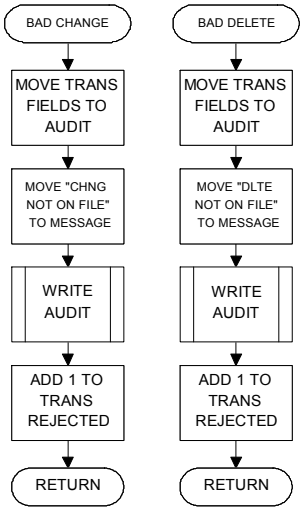
An ADD transaction with Trans Low is valid...




```
ADDIT    EQU   *
         ST    R10,SVADDIT
         MVC   ONBR,TNBR
         MVC   OLNAME,TLNAME
         MVC   OFNAME,TFNAME
         MVC   OADDR,TADDR
         MVC   OCITY,TCITY
         MVC   OSTATE,TSTATE
         MVC   OZIP,TZIP
         MVC   OACD,TACD
         MVC   OCRLF,TCRLF
         BAL   R10,WRITENEW
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,TREC
         MVC   RMSG,=CL16'ADD SUCCESSFUL'
         BAL   R10,WRITE
         AP    #ADDED,=P'1'
ADDITX   EQU   *
         L     R10,SVADDIT
         BR    R10
```

Invalid Change, Invalid Delete

A CHANGE or DELETE transaction with Trans Low is an error...

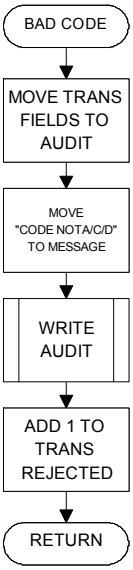



```
BADCHANG EQU   *
         ST    R10,SVBADCHG
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,TREC
         MVC   RMSG,=CL16'CHNG NOT ON FILE'
         BAL   R10,WRITE
         AP    #REJECTS,=P'1'
BADCHGX  EQU   *
         L     R10,SVBADCHG
         BR    R10

BADDELET EQU   *
         ST    R10,SVBADDEL
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,TREC
         MVC   RMSG,=CL16'DLTE NOT ON FILE'
         BAL   R10,WRITE
         AP    #REJECTS,=P'1'
BADDELX  EQU   *
         L     R10,SVBADDEL
         BR    R10
```

Bad Transaction Code

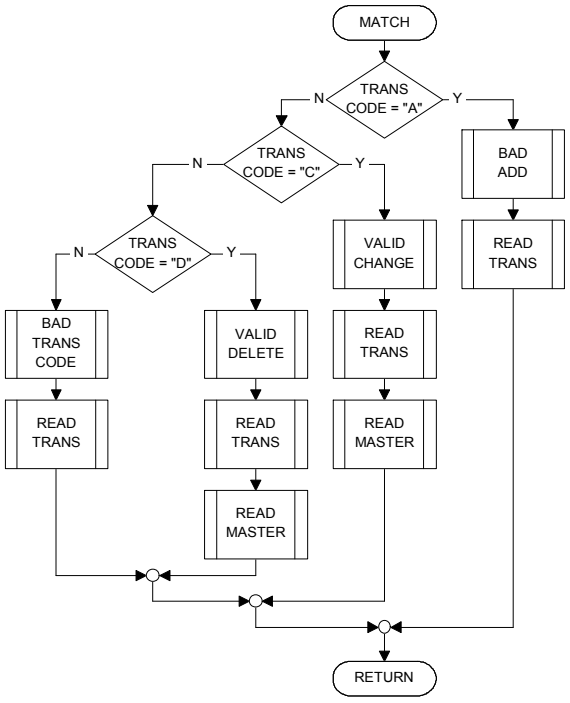
The previous chapter used two input files, but there we were not concerned with transaction codes. In this program, the transaction code *must* be A, C, or D. Anything else is an error condition...




```
BADCODE  EQU   *
         ST    R10,SVBADCOD
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,TREC
         MVC   RMSG,=CL16'CODE NOT A/C/D'
         BAL   R10,WRITE
         AP    #REJECTS,=P'1'
BADCODEX EQU   *
         L     R10,SVBADCOD
         BR    R10
```

Master/Transaction Match

The "Match" condition indicates we have a transaction with a matching master record. Again, whether or not this is an error depends on the transaction type. If it is an ADD, then it is an error: we cannot add a record with the same key as an existing record. If it is a CHANGE or DELETE, then it is valid: we are attempting to change or delete an existing record.

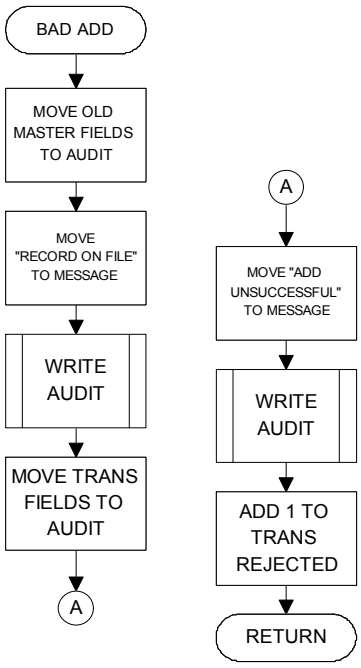


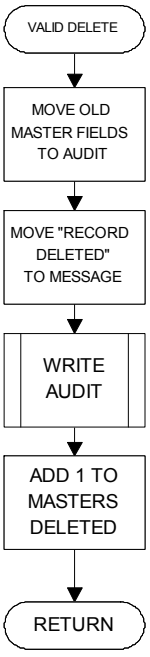

```
MATCH    EQU   *
         ST    R10,SVMATCH
         CLI   TACD,ADD
         BE    MATCH2
         CLI   TACD,CHANGE
         BE    MATCH3
         CLI   TACD,DELETE
         BE    MATCH4
         BAL   R10,BADCODE
         BAL   R10,READTRN
         B     MATCHX
MATCH2   EQU   *
         BAL   R10,BADADD
         BAL   R10,READTRN
         B     MATCHX
MATCH3   EQU   *
         BAL   R10,CHANGEIT
         BAL   R10,READTRN
         BAL   R10,READMST
         B     MATCHX
MATCH4   EQU   *
         BAL   R10,DELETEIT
         BAL   R10,READTRN
         BAL   R10,READMST
MATCHX   EQU   *
         L     R10,SVMATCH
         BR    R10
```

Invalid Add

An ADD transaction with a matching MASTER record is always invalid...

```
BADADD   EQU   *
         ST    R10,SVBADADD
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,IREC
         MVC   RMSG,=CL16'RECORD ON FILE'
         BAL   R10,WRITE
         MVC   RDATA,TREC
         MVC   RMSG,=CL16'ADD UNSUCCESSFUL'
         BAL   R10,WRITE
         AP    #REJECTS,=P'1'
BADADDX  EQU   *
         L     R10,SVBADADD
         BR    R10
```





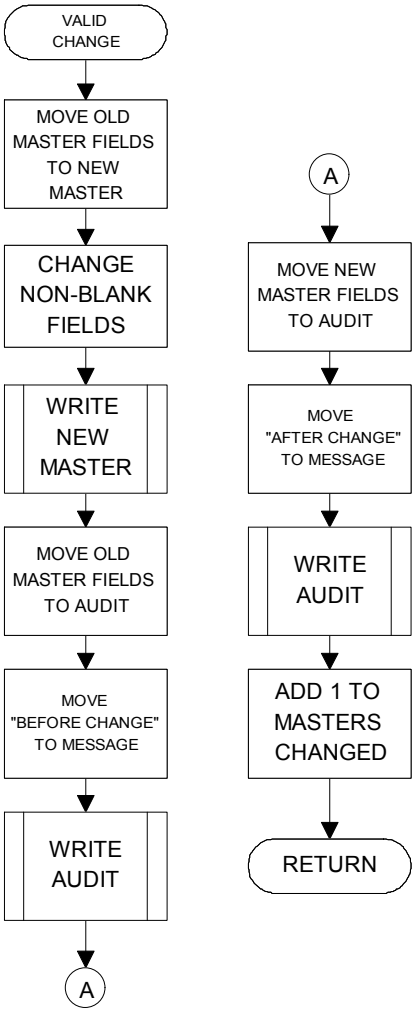
Valid Delete

A DELETE transaction with a matching MASTER record is valid. To delete a record from the old MASTER file, simply do not write it to the new MASTER file...

```
DELETEIT EQU   *
         ST    R10,SVDELIT
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,IREC
         MVC   RMSG,=CL16'RECORD DELETED'
         BAL   R10,WRITE
         AP    #DELETED,=P'1'
DELETEX  EQU   *
         L     R10,SVDELIT
         BR    R10
```

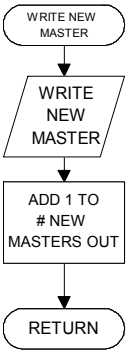
Valid Change

A CHANGE transaction with a matching MASTER record is valid. Copy the old MASTER record to the new MASTER record. Move all non-blank TRANSACTION fields to the corresponding fields on the new MASTER record. Write the new MASTER record. Show "before" and "after" images on the audit report.




```
CHANGEIT EQU   *
         ST    R10,SVCHGIT
*
* COPY EXISTING RECORD TO OUTPUT RECORD
* THEN MAKE THE REQUESTED CHANGES
*
         MVC   OREC,IREC
         CLC   TLNAME,BLANKS
         BE    CHGIT2
         MVC   OLNAME,TLNAME
CHGIT2   EQU   *
         CLC   TFNAME,BLANKS
         BE    CHGIT3
         MVC   OFNAME,TFNAME
CHGIT3   EQU   *
         CLC   TADDR,BLANKS
         BE    CHGIT4
         MVC   OADDR,TADDR
CHGIT4   EQU   *
         CLC   TCITY,BLANKS
         BE    CHGIT5
         MVC   OCITY,TCITY
CHGIT5   EQU   *
         CLC   TSTATE,BLANKS
         BE    CHGIT6
         MVC   OSTATE,TSTATE
CHGIT6   EQU   *
         CLC   TZIP,BLANKS
         BE    CHGIT7
         MVC   OZIP,TZIP
CHGIT7   EQU   *
         MVC   OACD,TACD
         BAL   R10,WRITENEW
*
* SHOW RECORD BEFORE AND AFTER CHANGES
*
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,IREC
         MVC   RMSG,=CL16'BEFORE CHANGE'
         BAL   R10,WRITE
         MVC   RDATA,OREC
         MVC   RMSG,=CL16'AFTER CHANGE'
         BAL   R10,WRITE
         AP    #CHANGED,=P'1'
CHGITX   EQU   *
         L     R10,SVCHGIT
         BR    R10
```

Write New Master



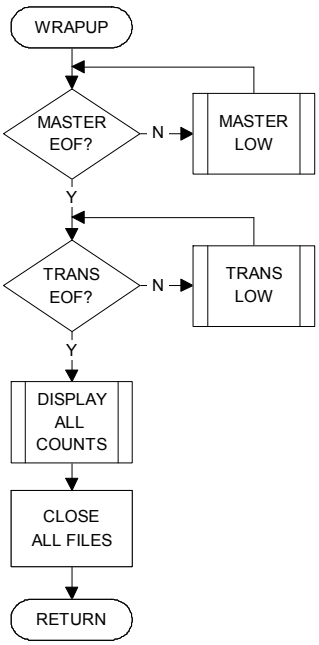
Recall that the new MASTER file is separate from the old MASTER file. There is nothing unusual in this routine. We include a count of the records written.

```
WRITENEW EQU   *
         ST    R10,SVWRITEN
         PUT   MASTEROT,OREC
         AP    #NEWOUT,=P'1'
         L     R10,SVWRITEN
         BR    R10
```

The WRAPUP Routine

The WRAPUP routine for this program is very similar to the one shown in the previous chapter. Recall that we continue to process records until the MASTER file *or* the TRANSACTION file is at EOF. Therefore, when we reach WRAPUP, it is likely that one of the these files is *not* at EOF. We will therefore execute one of the following loops:

- If the old MASTER file is not at EOF, then we process all remaining MASTER file records as unmatched; that is, master low. (Recall such records are written to the new MASTER file without changes.)
- Otherwise, if the TRANSACTION file is not at EOF, then we process all remaining TRANSACTION file records as unmatched; that is, transaction low. (Recall unmatched ADDS will be added, while unmatched CHANGES and DELETES are errors.)




```
WRAPUP   EQU   *
         ST    R10,SVWRAP
WRAPUP2  EQU   *
         CLI   EOFMAST,C'Y'
         BE    WRAPUP3
         BAL   R10,MASTLOW
         B     WRAPUP2
WRAPUP3  EQU   *
         CLI   EOFTRANS,C'Y'
         BE    WRAPUP4
         BAL   R10,TRANSLOW
         B     WRAPUP3
WRAPUP4  EQU   *
         CLOSE MASTERIN
         CLOSE TRANSIN
         CLOSE MASTEROT
         BAL   R10,DOCOUNTS
         CLOSE REPORT
         WTO   'BQSFU ... Audit list on REPORT.TXT'
         L     R10,SVWRAP
         BR    R10
```

Program Solution

The complete annotated program, `BQSFU.MLC`, follows.

```
         PRINT NOGEN
*****************************************************************
*        FILENAME:  BQSFU.MLC                                   *
*        AUTHOR  :  Bill Qualls                                 *
*        SYSTEM  :  PC/370 R4.2                                 *
*        REMARKS :  Sequential File Update Sample Program       *
*****************************************************************
         START 0
         REGS
BEGIN    BEGIN
         WTO   'BQSFU ... Begin execution'
         BAL   R10,SETUP
MAIN     EQU   *
         CLI   EOFMAST,C'Y'
         BE    EOJ
         CLI   EOFTRANS,C'Y'
         BE    EOJ
         BAL   R10,PROCESS
         B     MAIN
EOJ      EQU   *
         BAL   R10,WRAPUP
         WTO   'BQSFU ... Normal end of program'
         RETURN
*****************************************************************
*        SETUP - Those things which happen one time only,       *
*                before any records are processed.              *
*****************************************************************
SETUP    EQU   *
         ST    R10,SVSETUP
         OI    MASTERIN+10,X'08'  PC/370 ONLY - Convert all
*                                 input from ASCII to EBCDIC
         OI    TRANSIN+10,X'08'   PC/370 ONLY - Convert all
*                                 input from ASCII to EBCDIC
         OI    MASTEROT+10,X'08'  PC/370 ONLY - Convert all
*                                 output from EBCDIC to ASCII
         OI    REPORT+10,X'08'    PC/370 ONLY - Convert all
*                                 output from EBCDIC to ASCII
         OPEN  MASTERIN
         OPEN  TRANSIN
         OPEN  MASTEROT
         OPEN  REPORT
         BAL   R10,READMST
         BAL   R10,READTRN
         L     R10,SVSETUP
         BR    R10
*****************************************************************
*        HDGS - Print headings.                                 *
*****************************************************************
HDGS     EQU   *
         ST    R10,SVHDGS
         AP    PGS,=P'1'          Add 1 to page count
         MVC   HDPGS,=X'40202120' Edit pattern for page count
         ED    HDPGS,PGS          Move page count to heading
         PUT   REPORT,FORMFEED    PC/370 ONLY
```

(continued)

```
         PUT   REPORT,HD1
         PUT   REPORT,HD2
         PUT   REPORT,HD3
         PUT   REPORT,HD4
         ZAP   LNS,=P'0'          Reset line count to zero
         L     R10,SVHDGS
         BR    R10
*****************************************************************
*        PROCESS - Those things which happen once per record.   *
*****************************************************************
PROCESS  EQU   *
         ST    R10,SVPROC
         CLC   INBR,TNBR          Attempt match on customer nbr
         BH    PROC2              Transaction low
         BL    PROC3              Master low
         BAL   R10,MATCH          Otherwise a match was found
         B     PROCESSX
PROC2    EQU   *                  No master for this transaction
         BAL   R10,TRANSLOW
         B     PROCESSX
PROC3    EQU   *                  No transaction for this master
         BAL   R10,MASTLOW
PROCESSX EQU   *
         L     R10,SVPROC
         BR    R10
*****************************************************************
*        MASTLOW - No updates for this master record.           *
*                  Just write this record and go to next.       *
*****************************************************************
MASTLOW  EQU   *
         ST    R10,SVMSTLOW
         MVC   OREC,IREC          Move input to output
         BAL   R10,WRITENEW       Write new master record
         BAL   R10,READMST        Read next master record
         L     R10,SVMSTLOW
         BR    R10
*****************************************************************
*        MATCH - Transaction for existing master record.        *
*****************************************************************
MATCH    EQU   *
         ST    R10,SVMATCH
         CLI   TACD,ADD           Attempt to add?
         BE    MATCH2             Can't do it - already exists
         CLI   TACD,CHANGE        Attempt to change?
         BE    MATCH3             OK to change existing record
         CLI   TACD,DELETE        Attempt to delete?
         BE    MATCH4             OK to delete existing record
         BAL   R10,BADCODE        Error - unrecognized code
         BAL   R10,READTRN        Read next transaction
         B     MATCHX
MATCH2   EQU   *                  Attempt to add
         BAL   R10,BADADD         Can't add - it already exists
         BAL   R10,READTRN        Read next transaction
         B     MATCHX
MATCH3   EQU   *                  Attempt to change
         BAL   R10,CHANGEIT       OK to change existing record
         BAL   R10,READTRN        Read next transaction
         BAL   R10,READMST        Read next master
         B     MATCHX
MATCH4   EQU   *                  Attempt to delete
         BAL   R10,DELETEIT       OK to delete existing record
```

(continued)

```
         BAL   R10,READTRN        Read next transaction
         BAL   R10,READMST        Read next master
MATCHX   EQU   *
         L     R10,SVMATCH
         BR    R10
*****************************************************************
*        TRANSLOW - Transaction without a matching master.      *
*****************************************************************
TRANSLOW EQU   *
         ST    R10,SVTRNLOW
         CLI   TACD,ADD           Attempt to add?
         BE    TRANSLO2           OK since it doesn't exist
         CLI   TACD,CHANGE        Attempt to change?
         BE    TRANSLO3           Can't change - doesn't exist
         CLI   TACD,DELETE        Attempt to delete?
         BE    TRANSLO4           Can't delete - not there
         BAL   R10,BADCODE        Error - Unrecognized code
         BAL   R10,READTRN        Read next transaction
         B     TRANSLOX
TRANSLO2 EQU   *                  Attempt to add
         BAL   R10,ADDIT          OK to add - not there already
         BAL   R10,READTRN        Read next transaction
         B     TRANSLOX
TRANSLO3 EQU   *                  Attempt to change
         BAL   R10,BADCHANG       Can't change - doesn't exist
         BAL   R10,READTRN        Read next transaction
         B     TRANSLOX
TRANSLO4 EQU   *                  Attempt to delete
         BAL   R10,BADDELET       Can't delete - doesn't exist
         BAL   R10,READTRN        Read next transaction
TRANSLOX EQU   *
         L     R10,SVTRNLOW
         BR    R10
*****************************************************************
*        BADCODE - Bad Transaction Code                         *
*****************************************************************
BADCODE  EQU   *
         ST    R10,SVBADCOD
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,TREC
         MVC   RMSG,=CL16'CODE NOT A/C/D'
         BAL   R10,WRITE
         AP    #REJECTS,=P'1'
BADCODEX EQU   *
         L     R10,SVBADCOD
         BR    R10
*****************************************************************
*        BADADD - Bad Add Attempted                             *
*****************************************************************
BADADD   EQU   *
         ST    R10,SVBADADD
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,IREC
         MVC   RMSG,=CL16'RECORD ON FILE'
         BAL   R10,WRITE
         MVC   RDATA,TREC
         MVC   RMSG,=CL16'ADD UNSUCCESSFUL'
```

(continued)

```
         BAL   R10,WRITE
         AP    #REJECTS,=P'1'
BADADDX  EQU   *
         L     R10,SVBADADD
         BR    R10
*****************************************************************
*        BADCHG - Bad Change Attempted                          *
*****************************************************************
BADCHANG EQU   *
         ST    R10,SVBADCHG
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,TREC
         MVC   RMSG,=CL16'CHNG NOT ON FILE'
         BAL   R10,WRITE
         AP    #REJECTS,=P'1'
BADCHGX  EQU   *
         L     R10,SVBADCHG
         BR    R10
*****************************************************************
*        BADDEL - Bad Delete Attempted                          *
*****************************************************************
BADDELET EQU   *
         ST    R10,SVBADDEL
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,TREC
         MVC   RMSG,=CL16'DLTE NOT ON FILE'
         BAL   R10,WRITE
         AP    #REJECTS,=P'1'
BADDELX  EQU   *
         L     R10,SVBADDEL
         BR    R10
*****************************************************************
*        ADDIT - Add a new record to master file                *
*****************************************************************
ADDIT    EQU   *
         ST    R10,SVADDIT
         MVC   ONBR,TNBR
         MVC   OLNAME,TLNAME
         MVC   OFNAME,TFNAME
         MVC   OADDR,TADDR
         MVC   OCITY,TCITY
         MVC   OSTATE,TSTATE
         MVC   OZIP,TZIP
         MVC   OACD,TACD
         MVC   OCRLF,TCRLF
         MVC   OREC,TREC
         BAL   R10,WRITENEW
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,TREC
         MVC   RMSG,=CL16'ADD SUCCESSFUL'
         BAL   R10,WRITE
         AP    #ADDED,=P'1'
ADDITX   EQU   *
         L     R10,SVADDIT
         BR    R10
```

(continued)

```
*****************************************************************
*        CHANGEIT - Apply changes to existing master record     *
*****************************************************************
CHANGEIT EQU   *
         ST    R10,SVCHGIT
*
*        COPY EXISTING RECORD TO OUTPUT RECORD
*        THEN MAKE THE REQUESTED CHANGES
*
         MVC   OREC,IREC
         CLC   TLNAME,BLANKS
         BE    CHGIT2
         MVC   OLNAME,TLNAME
CHGIT2   EQU   *
         CLC   TFNAME,BLANKS
         BE    CHGIT3
         MVC   OFNAME,TFNAME
CHGIT3   EQU   *
         CLC   TADDR,BLANKS
         BE    CHGIT4
         MVC   OADDR,TADDR
CHGIT4   EQU   *
         CLC   TCITY,BLANKS
         BE    CHGIT5
         MVC   OCITY,TCITY
CHGIT5   EQU   *
         CLC   TSTATE,BLANKS
         BE    CHGIT6
         MVC   OSTATE,TSTATE
CHGIT6   EQU   *
         CLC   TZIP,BLANKS
         BE    CHGIT7
         MVC   OZIP,TZIP
CHGIT7   EQU   *
         MVC   OACD,TACD
         BAL   R10,WRITENEW
*
*        SHOW THE RECORD BEFORE AND AFTER CHANGES
*
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         MVC   RDATA,IREC
         MVC   RMSG,=CL16'BEFORE CHANGE'
         BAL   R10,WRITE
         MVC   RDATA,OREC
         MVC   RMSG,=CL16'AFTER CHANGE'
         BAL   R10,WRITE
         AP    #CHANGED,=P'1'
CHGITX   EQU   *
         L     R10,SVCHGIT
         BR    R10
*****************************************************************
*        DELETEIT - Delete an existing master record            *
*                   (To delete it, just don't write it out.)    *
*****************************************************************
DELETEIT EQU   *
         ST    R10,SVDELIT
         BAL   R10,CHKLNS
         MVC   RREC,BLANKS
```

(continued)

```
         BAL   R10,WRITE
         MVC   RDATA,IREC
         MVC   RMSG,=CL16'RECORD DELETED'
         BAL   R10,WRITE
         AP    #DELETED,=P'1'
DELETEX  EQU   *
         L     R10,SVDELIT
         BR    R10
*****************************************************************
*        READMST - Read a master record.                        *
*****************************************************************
READMST  EQU   *
         ST    R10,SVREADM
         GET   MASTERIN,IREC
         AP    #OLDIN,=P'1'
         B     READMX
ATENDMST EQU   *
         MVI   EOFMAST,C'Y'
READMX   EQU   *
         L     R10,SVREADM
         BR    R10
*****************************************************************
*        READOFF - Read a transaction record.                   *
*****************************************************************
READTRN  EQU   *
         ST    R10,SVREADT
         GET   TRANSIN,TREC
         AP    #TRANSIN,=P'1'
         B     READTX
ATENDTRN EQU   *
         MVI   EOFTRANS,C'Y'
READTX   EQU   *
         L     R10,SVREADT
         BR    R10
*****************************************************************
*        CHKLNS - Check lines printed. Full page?               *
*****************************************************************
CHKLNS   EQU   *
         ST    R10,SVCHKLNS
         CP    LNS,MAXLNS
         BL    CHKLNSX
         BAL   R10,HDGS
CHKLNSX  EQU   *
         L     R10,SVCHKLNS
         BR    R10
*****************************************************************
*        WRITE - Write a single detail line.                    *
*****************************************************************
WRITE    EQU   *
         ST    R10,SVWRITE
         PUT   REPORT,RREC        Write report line
         AP    LNS,=P'1'
         L     R10,SVWRITE
         BR    R10
*****************************************************************
*        WRITE - Write a new master record.                     *
*****************************************************************
WRITENEW EQU   *
         ST    R10,SVWRITEN
         PUT   MASTEROT,OREC
```

(continued)

```
         AP    #NEWOUT,=P'1'
         L     R10,SVWRITEN
         BR    R10
*****************************************************************
*        WRAPUP - Those things which happen one time only,      *
*                 after all records have been processed.        *
*****************************************************************
WRAPUP   EQU   *
         ST    R10,SVWRAP
*                                  At this point we know that
*                                  at least one of the input
*                                  files is at EOF. Process
*                                  other file as "unmatched"
*                                  until at EOF also.
WRAPUP2  EQU   *
         CLI   EOFMAST,C'Y'
         BE    WRAPUP3
         BAL   R10,MASTLOW
         B     WRAPUP2
WRAPUP3  EQU   *
         CLI   EOFTRANS,C'Y'
         BE    WRAPUP4
         BAL   R10,TRANSLOW
         B     WRAPUP3
WRAPUP4  EQU   *
         CLOSE MASTERIN
         CLOSE TRANSIN
         CLOSE MASTEROT
         BAL   R10,DOCOUNTS
         CLOSE REPORT
         WTO   'BQSFU ... Audit list on REPORT.TXT'
         L     R10,SVWRAP
         BR    R10
*****************************************************************
*        DOCOUNTS - Show counts for audit                       *
*****************************************************************
DOCOUNTS EQU   *
         ST    R10,SVCOUNTS
         BAL   R10,HDGS
         MVC   AREC,BLANKS
         BAL   R10,WRITE
*
         MVC   ADESC,=CL25'Transactions In'
         MVC   ACOUNT,EDCOUNT
         ED    ACOUNT,#TRANSIN
         BAL   R10,WRITE
*
         MVC   ADESC,=CL25'Transactions Rejected'
         MVC   ACOUNT,EDCOUNT
         ED    ACOUNT,#REJECTS
         BAL   R10,WRITE
*
         MVC   ADESC,=CL25'Old Masters In'
         MVC   ACOUNT,EDCOUNT
         ED    ACOUNT,#OLDIN
         BAL   R10,WRITE
*
         MVC   ADESC,=CL25'Old Masters Deleted'
         MVC   ACOUNT,EDCOUNT
         ED    ACOUNT,#DELETED
```

(continued)

```
         BAL   R10,WRITE
*
         MVC   ADESC,=CL25'Old Masters Changed'
         MVC   ACOUNT,EDCOUNT
         ED    ACOUNT,#CHANGED
         BAL   R10,WRITE
*
         MVC   ADESC,=CL25'New Masters Added'
         MVC   ACOUNT,EDCOUNT
         ED    ACOUNT,#ADDED
         BAL   R10,WRITE
*
         MVC   ADESC,=CL25'New Masters Out'
         MVC   ACOUNT,EDCOUNT
         ED    ACOUNT,#NEWOUT
         BAL   R10,WRITE
*
         L     R10,SVCOUNTS
         BR    R10
*****************************************************************
*        Literals, if any, will go here                         *
*****************************************************************
         LTORG
*****************************************************************
*        File definitions                                       *
*****************************************************************
MASTERIN DCB   LRECL=62,RECFM=F,MACRF=G,EODAD=ATENDMST,
               DDNAME='BQSFUMST.DAT'
TRANSIN  DCB   LRECL=62,RECFM=F,MACRF=G,EODAD=ATENDTRN,
               DDNAME='BQSFUTRN.DAT'
MASTEROT DCB   LRECL=62,RECFM=F,MACRF=P,
               DDNAME='BQSFUNEW.DAT'
REPORT   DCB   LRECL=80,RECFM=F,MACRF=P,
               DDNAME='REPORT.TXT'
*****************************************************************
*        RETURN ADDRESSES                                       *
*****************************************************************
SVSETUP  DC    F'0'               SETUP
SVHDGS   DC    F'0'               HDGS
SVPROC   DC    F'0'               PROCESS
SVREADM  DC    F'0'               READMST
SVREADT  DC    F'0'               READTRN
SVWRITE  DC    F'0'               WRITE
SVWRITEN DC    F'0'               WRITENEW
SVWRAP   DC    F'0'               WRAPUP
SVCHKLNS DC    F'0'               CHKLNS
SVMATCH  DC    F'0'               MATCH
SVMSTLOW DC    F'0'               MASTLOW
SVTRNLOW DC    F'0'               TRANSLOW
SVCOUNTS DC    F'0'               DOCOUNTS
SVBADCOD DC    F'0'               BADCODE
SVBADADD DC    F'0'               BADADD
SVBADCHG DC    F'0'               BADCHANG
SVBADDEL DC    F'0'               BADDELET
SVADDIT  DC    F'0'               ADDIT
SVCHGIT  DC    F'0'               CHANGEIT
SVDELIT  DC    F'0'               DEELTEIT
*****************************************************************
*        Miscellaneous field definitions                        *
*****************************************************************
```

(continued)

```
EOFMAST  DC    CL1'N'             End of master file? (Y/N)
EOFTRANS DC    CL1'N'             End of transaction file? (Y/N)
EDCOUNT  DC    X'40206B2020206B202120'   BZ,ZZZ,ZZ9
PGS      DC    PL2'0'             Nbr of pages printed.
LNS      DC    PL2'20'            Lines printed on this page.
MAXLNS   DC    PL2'20'            Max nbr lines per page.
*                                 My line counts exclude hdgs.
BLANKS   DS    0CL80
         DC    CL78' ',XL2'0D25'
*****************************************************************
*        Transaction codes                                      *
*****************************************************************
ADD      EQU   C'A'
CHANGE   EQU   C'C'
DELETE   EQU   C'D'
*****************************************************************
*        Counts for audit purposes                              *
*****************************************************************
#TRANSIN DC    PL4'0'    Transactions In
#REJECTS DC    PL4'0'    Transactions Rejected
#OLDIN   DC    PL4'0'    Old Masters In
#DELETED DC    PL4'0'    Old Masters Deleted
#CHANGED DC    PL4'0'    Old Masters Changed
#ADDED   DC    PL4'0'    New Masters Added
#NEWOUT  DC    PL4'0'    New Masters Out
*****************************************************************
*        Input record definition - Master In                    *
*****************************************************************
IREC     DS    0CL62      1-62    Master record
INBR     DS    CL5        1- 5    Customer nbr
ILNAME   DS    CL10       6-15    Last name
IFNAME   DS    CL10      16-25    First name
IADDR    DS    CL15      26-40    Address
ICITY    DS    CL10      41-50    City
ISTATE   DS    CL2       51-52    State
IZIP     DS    CL5       53-57    Zip
         DS    CL2       58-59    Unused
IACD     DS    CL1       60-60    Transaction code (A/C/D)
ICRLF    DS    CL2       61-62    PC/370 only - CR/LF
*****************************************************************
*        Input record definition - Transaction                  *
*****************************************************************
TREC     DS    0CL62      1-62    Transaction record
TNBR     DS    CL5        1- 5    Customer nbr
TLNAME   DS    CL10       6-15    Last name
TFNAME   DS    CL10      16-25    First name
TADDR    DS    CL15      26-40    Address
TCITY    DS    CL10      41-50    City
TSTATE   DS    CL2       51-52    State
TZIP     DS    CL5       53-57    Zip
         DS    CL2       58-59    Unused
TACD     DS    CL1       60-60    Transaction code (A/C/D)
TCRLF    DS    CL2       61-62    PC/370 only - CR/LF
*****************************************************************
*        Output record definition - Master Out                  *
*****************************************************************
OREC     DS    0CL62      1-62    Master record
ONBR     DS    CL5        1- 5    Customer nbr
OLNAME   DS    CL10       6-15    Last name
OFNAME   DS    CL10      16-25    First name
```

(continued)

```
OADDR    DS    CL15      26-40    Address
OCITY    DS    CL10      41-50    City
OSTATE   DS    CL2       51-52    State
OZIP     DS    CL5       53-57    Zip
         DS    CL2       58-59    Unused
OACD     DS    CL1       60-60    Transaction code (A/C/D)
OCRLF    DS    CL2       61-62    PC/370 only - CR/LF
*****************************************************************
*        Output (line) definition                               *
*****************************************************************
RREC     DS    0CL80      1-80    Report record
RDATA    DC    CL60' '    1-60    Transaction Data
         DC    CL2' '    61-62
RMSG     DC    CL16' '   63-78    Audit message
RCRLF    DS    CL2       79-80    PC/370 only - CR/LF
*****************************************************************
*        Output record definition - Audit                       *
*        !!! NOTE HOW SPACE FOR RREC IS REDEFINED !!!           *
*****************************************************************
         ORG   RREC
AREC     DS    0CL80      1-87    Audit Line
ADESC    DC    CL25' '    1-25    Description on count
ACOUNT   DC    CL10' '   26-35    Count
         DC    CL43' '   36-78
ACRLF    DS    CL2       79-80    PC/370 only - CR/LF
         ORG
*****************************************************************
*        Headings definitions                                   *
*****************************************************************
FORMFEED DS    0CL80              PC/370 only
*        DC    X'0C'              EBCDIC formfeed
*        DC    CL77' '
         DC    78C'_'             For testing...
         DC    X'0D25'            EBCDIC CR/LF
HD1      DS    0CL80
         DC    CL40'               Name & Address Update Pro'
         DC    CL26'gram                 Page'
HDPGS    DC    CL4'BZZ9'
         DC    CL8' '
         DC    XL2'0D25'
HD2      DS    0CL80
         DC    CL78'                      Audit Listing'
         DC    XL2'0D25'
HD3      DS    0CL80
         DC    CL78' '
         DC    XL2'0D25'
HD4      DS    0CL80
         DC    CL40'----+----1----+----2----+----3----+----4'
         DC    CL38'----+----5----+----6  MESSAGES'
         DC    XL2'0D25'
         END   BEGIN
```

Note the use of the ORG instruction above. This allows us to "redefine" a record. The first instruction, ORG RREC, says to reset the location counter (remember: "stuff on the left") to what it was at RREC above. The second instruction, ORG (alone), says to put the location counter back to what it was prior to the first ORG. By using the ORG, we have saved 80 bytes of memory. When using ORG, it is easy to forget the "closing" ORG. Don't do it!

Reconciling the Audit Report

Let's take one more look at the audit report produced by this program:

```
                Name & Address Update Program                  Page   2
                        Audit Listing

----+----1----+----2----+----3----+----4----+----5----+----6  MESSAGES

Transactions In                   6
Transactions Rejected             3
Old Masters In                   10
Old Masters Deleted               1
Old Masters Changed               1
New Masters Added                 1
New Masters Out                  10
```

It is important that we be able to reconcile the counts in this report; that is, we need to be able to account for all transaction and master records. For example, we see that there were six transactions in. So what what happened to them? Three were rejected for errors, one was a valid delete, one was a valid change, and one was a valid add: 3 + 1 + 1 + 1 = 6.

We see that there were 10 old masters in and 10 new masters out. How do we reconcile these counts? We see that one record was deleted, so that takes us from ten down to nine. One record was changed, but changes have no affect on the number of records. Finally, one record was added, so that takes us from nine back up to ten: 10 - 1 + 1 = 10.

Exercises

1. True or false. When two files are processed using sequential file update logic...

   T F a. there are three different transaction types.
   T F b. both files must be in "key" sequence.
   T F c. the MAINLINE logic will continue until both files are at EOF.
   T F d. the SETUP routine will contain a priming read for both files.
   T F e. an ADD transaction in MATCH is an error.
   T F f. a CHANGE transaction in MATCH is an error.
   T F g. a DELETE transaction in MATCH is an error.
   T F h. an ADD transaction in TRANSLOW is an error.
   T F i. a CHANGE transaction in TRANSLOW is an error.
   T F j. a DELETE transaction in TRANSLOW is an error.
   T F k. we read the next MASTER and TRANSACTION records at the end of the PROCESS routine.
   T F l. NEW MASTERS = OLD MASTERS + valid ADDS - valid DELETES.
   T F m. TRANSACTIONS IN = TRANSACTIONS rejected + valid ADDS - valid DELETES.

2. The following table shows the key for records on the Old Master file, and the key and transaction type for records on the Transaction file. Determine which records will be written to the New Master file, and supply the counts for the Audit Report.

Old Master
112
222
317
469
558
627
731
880
914
921

Trans Key	Trans Type
222	D
223	C
496	A
558	D
628	D
731	C
808	D
914	A

New Master

Audit Report

Transactions In	
Transactions Rejected	
Old Masters In	
Old Masters Deleted	
Old Masters Changed	
New Masters Added	
New Masters Out	

Exercises

3. Which of the following audit reports indicate a program error?

```
(a)  Transactions In          35      (b)  Transactions In          23
     Transactions Rejected     5           Transactions Rejected     3
     Old Masters In           50           Old Masters In           10
     Old Masters Deleted      10           Old Masters Deleted       6
     Old Masters Changed       5           Old Masters Changed       7
     New Masters Added        15           New Masters Added         8
     New Masters Out          60           New Masters Out          12
```

4. In the sample program, BQSFU.MLC, we assumed a blank field on a change transaction indicated that the corresponding field on the master record should not be changed. But how would we indicate that a field should be changed to blanks? For example, we may know that a customer has moved to a new city, but we don't know the new zip code. One common solution to this type of problem is to designate some special character (such as dollar sign) which, if found in the first position of a field in the transaction record, indicates that blanks should be moved to the corresponding field in the master record. Change the sample program to include this feature. Test with the following transactions:

```
         1         2         3         4         5         6
12345678901234567890123456789012345678901234567890123456789012345678901234567890
12111                     2211 APRICOT   MODESTO   CA$       C    (changed)
41499                                                         D
55123AMBROSE   FRANK      220 BARRETT    ROCKFORD  IL61103   A
61627QUALLS    CHERYL     201 N EIGHTH   WHITTIER  CA90605   C
81228                                                         D
81997          KEVEN      $              WEST CHGO IL60185   C    (new)
82446AMICCI    BRUNO      17397 BARCELON CORVALLIS OR97330   A
```

5. In the sample program, BQSFU.MLC, deleted records were *physically* deleted; that is, those records were no longer included in the MASTER file. Sometimes we prefer records be *logically* deleted; that is, the records are still included in the MASTER file, but they are tagged in such a way as to indicate that they should not be processed by other programs. (This is how dBASE handles deleted records: an asterisk in the first position of the record is used to indicate logically deleted records. dBASE's PACK command is used to physically delete them.) We have included the most recent transaction code in position 60 of the master record. We could say a D in this position indicates the record is logically deleted.

   (a) Change the sample program to include this feature.
   (b) Show how the READ routine would be modified for all other programs using this file.
   (c) Write a program which would physically delete all records which are currently logically deleted (similar to dBASE's PACK).
   (d) Write a program which would "undelete" all records which are currently logically deleted (similar to dBASE's RECALL).

Exercises

6. (Refer to the Small Town Payroll database in More Datasets.) The following table describes the transactions used to update the EMPL file:

Field Nbr	Field Name	Description	Begins	Ends	Len	Format
1	NUM	Employee number	1	3	3	ZD
2	LNAME	Last name	4	13	10	CH
3	FNAME	First name	14	23	10	CH
4	DEPT	Department	24	24	1	CH
5	RATE	Pay rate	25	29	5	999V99
6	TYPE	Rate type	30	30	1	H or S
7	ACD	Transaction code	31	31	1	A/C/D
8	CRLF	PC/370 Only	32	33	2	CR/LF

Write a program which will apply the following transactions to the EMPL file.

```
         1         2         3
12345678901234567890123456789012
270                         32000 C
310DECARLO   RICHARD     30000HA
610                             D
688SMITH     GERRY      A00525HA
791                      00475 C
828GOYAK     DEBBIE     A34000SA
857                             D
```

Design a meaningful audit report with the appropriate counts. Before updating the file, verify the following limits:

Rate type	Minimum Rate	Maximum Rate
H	$4.00	$10.00
S	$200.00	$500.00

7. (Refer to the Small Town Hardware Store database in More Datasets.) The following table describes the transactions used to update the cost and sell price (only) for the TOOL file:

Field Nbr	Field Name	Description	Begins	Ends	Len	Format
1	TID	Tool ID	1	3	3	CH
2	TCOST	Tool cost each	4	8	5	999V99
3	TSELL	Tool sells for	9	13	5	999V99
4	CRLF	PC/370 Only	14	15	2	CR/LF

Write a program which will apply the following transactions to the TOOL file. Note that there are no transaction codes: records can be changed (only). Of course, a record must exist before it can be changed...

(continued)

Exercises

```
         1
1234567890123
H8      01099
PLM00375
PLX     00599
SPM0025000419
SSM0041900250
SSS00399
WSP00025
```

Design a meaningful audit report with the appropriate counts. Before updating the file, verify that the cost is less than the sell price, except for wrappers, which must have a sell price of zero. (A wrapper is indicated by a `W` in the first position of the tool ID.)