<table>
<thead>
<tr>
<th>CONTENTS</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>1</td>
</tr>
<tr>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td>Purpose and scope of report</td>
<td>2</td>
</tr>
<tr>
<td>Relation to goals</td>
<td>2</td>
</tr>
<tr>
<td>National Water Data Storage and Retrieval System</td>
<td>2</td>
</tr>
<tr>
<td>Description of the data-management system</td>
<td>4</td>
</tr>
<tr>
<td>Computer languages</td>
<td>4</td>
</tr>
<tr>
<td>Variable-length records</td>
<td>5</td>
</tr>
<tr>
<td>Defining new record types</td>
<td>6</td>
</tr>
<tr>
<td>Data-file storage</td>
<td>7</td>
</tr>
<tr>
<td>Control records for COBOL programs</td>
<td>7</td>
</tr>
<tr>
<td>Audit trail</td>
<td>9</td>
</tr>
<tr>
<td>Data file descriptions</td>
<td>10</td>
</tr>
<tr>
<td>Master data file</td>
<td>10</td>
</tr>
<tr>
<td>Working data files</td>
<td>12</td>
</tr>
<tr>
<td>Ground-water data file</td>
<td>12</td>
</tr>
<tr>
<td>Water-quality data file</td>
<td>13</td>
</tr>
<tr>
<td>Water-level data file</td>
<td>14</td>
</tr>
<tr>
<td>Well-construction data file</td>
<td>15</td>
</tr>
<tr>
<td>Water-use data file</td>
<td>15</td>
</tr>
<tr>
<td>Aquifer-test data file</td>
<td>16</td>
</tr>
<tr>
<td>Well-log data file</td>
<td>18</td>
</tr>
<tr>
<td>Geophysical data file</td>
<td>18</td>
</tr>
<tr>
<td>Utility computer programs</td>
<td>19</td>
</tr>
<tr>
<td>Polygon-definition program—POLYGON</td>
<td>19</td>
</tr>
<tr>
<td>Program operation</td>
<td>21</td>
</tr>
<tr>
<td>Equations of the line segments</td>
<td></td>
</tr>
<tr>
<td>Program input</td>
<td>23</td>
</tr>
<tr>
<td>Program output</td>
<td>24</td>
</tr>
<tr>
<td>Error messages</td>
<td>25</td>
</tr>
<tr>
<td>Computer programs for editing and updating the ground-water master file</td>
<td>26</td>
</tr>
<tr>
<td>Master file edit program—EDIT-1</td>
<td>27</td>
</tr>
<tr>
<td>Audit-trail initiation</td>
<td>27</td>
</tr>
<tr>
<td>Testing records for inclusion in the master file</td>
<td>27</td>
</tr>
<tr>
<td>EDIT-1 input data</td>
<td>29</td>
</tr>
<tr>
<td>Control records</td>
<td>29</td>
</tr>
<tr>
<td>Free-field format change records</td>
<td>30</td>
</tr>
<tr>
<td>Polygon-input records</td>
<td>31</td>
</tr>
<tr>
<td>Program output</td>
<td>31</td>
</tr>
<tr>
<td>Output change file</td>
<td>31</td>
</tr>
<tr>
<td>Output pull file</td>
<td>32</td>
</tr>
<tr>
<td>Output line-printer file</td>
<td>32</td>
</tr>
<tr>
<td>Component-format edit program—EDIT-2</td>
<td>32</td>
</tr>
<tr>
<td>Program input</td>
<td>33</td>
</tr>
<tr>
<td>Control records</td>
<td>33</td>
</tr>
<tr>
<td>Input data records</td>
<td>33</td>
</tr>
</tbody>
</table>
Computer programs for editing and updating the ground-water master file—Continued

<table>
<thead>
<tr>
<th>Program output</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volume serial number</td>
<td>35</td>
</tr>
<tr>
<td>Julian date</td>
<td>35</td>
</tr>
<tr>
<td>Site-identification number</td>
<td>35</td>
</tr>
<tr>
<td>Change-record set sequence number</td>
<td>36</td>
</tr>
<tr>
<td>Record-type identifier</td>
<td>36</td>
</tr>
<tr>
<td>Component identifier</td>
<td>36</td>
</tr>
<tr>
<td>Data length</td>
<td>36</td>
</tr>
<tr>
<td>Data-type code</td>
<td>36</td>
</tr>
<tr>
<td>Component value</td>
<td>36</td>
</tr>
</tbody>
</table>

Output record format | 36

Error messages—Control-001 | 37
Error messages—Control-002 | 38

Component-value edit program—EDIT-3

Program output | 40
Update change file | 42
Output error file | 42
Error messages—edit errors | 42
Error messages—programmed stops | 44

Master file update preparation program—UPDATE-1

Program input | 45
Program output | 45

Master file update program—UPDATE-2

Valid update transactions | 47
A) dd transaction | 47
M) odify transaction | 47
D)elete transaction | 47
Null values | 49
Program input | 49
Program output | 49
Updated master ground-water data file | 49
Error output file | 49

Programs for data retrieval and reporting

Ground-water site inventory (GWSI) data retrieval program—BASIN-3

Program input | 50
Control records | 50
Polygon-description file | 50
Ground-water data file | 50
Program output | 50
Site inventory report | 52
Site data output file | 52
Error messages | 53

Water-level history program—BASIN-4

Program input | 55
Control records | 57
Polygon-description file | 57
Water-level data file | 57
Programs for data retrieval and reporting—Continued

Water-level history programs—BASIN-4—Continued

<table>
<thead>
<tr>
<th>Program output</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error messages</td>
<td>59</td>
</tr>
<tr>
<td>Areal plot of selected ground-water site data—BSNPLT</td>
<td>60</td>
</tr>
<tr>
<td>Time-span selection parameters</td>
<td>61</td>
</tr>
<tr>
<td>Numeric selection parameters</td>
<td>62</td>
</tr>
<tr>
<td>Alpha-coded selection parameters</td>
<td>63</td>
</tr>
<tr>
<td>Site-labeling options</td>
<td>63</td>
</tr>
<tr>
<td>Map-border options</td>
<td>64</td>
</tr>
<tr>
<td>Program input</td>
<td>65</td>
</tr>
<tr>
<td>Control records</td>
<td>65</td>
</tr>
<tr>
<td>Program output</td>
<td>68</td>
</tr>
<tr>
<td>Site-data report and plot-option listing</td>
<td>68</td>
</tr>
<tr>
<td>Plot description</td>
<td>68</td>
</tr>
<tr>
<td>Selection information listing</td>
<td>68</td>
</tr>
<tr>
<td>Site-data listing</td>
<td>68</td>
</tr>
<tr>
<td>Error messages</td>
<td>69</td>
</tr>
</tbody>
</table>

References cited ................................................... 71

Attachments ........................................................ 72

A. COBOL record descriptions for WATSTORE record types
B. Program listings

_________________________________________________

ILLUSTRATIONS

_________________________________________________

<table>
<thead>
<tr>
<th>Figures</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Flow chart showing relations between programs and data files in the data-management system</td>
<td>3</td>
</tr>
</tbody>
</table>
| 2-3. Sketches showing:  
  2. Process of creating a new master-file record type with COBOL record description | 8 |
| 3. Sample polygons | 20 |
| 4-11. Flow charts showing:  
  4. General system logic for program POLYGON | 22 |
| 5. General system logic for program EDIT-1 | 28 |
| 6. General system logic for program EDIT-2 | 34 |
| 7. General system logic for program EDIT-3 | 41 |
| 8. General system logic for program UPDATE-1 | 46 |
| 9. General system logic for program UPDATE-2 | 48 |
| 10. General system logic for program BASIN-3 | 51 |
| 11. General system logic for program BASIN-4 | 56 |
| 12. Diagram showing border options for areal plot of ground-water site data using program BSNPLT | 66 |
### TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Record-identification numbers in use by WATSTORE</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>WATSTORE identification numbers and record titles used in the Swab/RASA ground-water master file</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>Example of &quot;slice table&quot; of line segment equations for the polygon shown in figure 3</td>
<td>21</td>
</tr>
<tr>
<td>4</td>
<td>Parameter codes and their meaning as printed by the BASIN-3 program</td>
<td>54</td>
</tr>
</tbody>
</table>
A DATA-MANAGEMENT SYSTEM FOR USE IN GROUND-WATER MODELING AND RESOURCE EVALUATION

By

Randall L. Fields and Edward F. Vetter

ABSTRACT

A data-management system was developed for use by the Southwest Alluvial Basins, Regional Aquifer-System Analysis study. Of the more than 100 computer programs developed during the study, the 9 programs described in this report form the nucleus of this ground-water data-management system. Five of the programs are used to edit and update changes to the system's master data file. The remaining programs include a polygon-definition program, two programs for data retrieval and reporting, and a program for plotting areal distributions of ground-water data.

INTRODUCTION

This report describes a data-management system developed for use by the Southwest Alluvial Basins, Regional Aquifer-System Analysis (Swab/RASA) study. The study area includes about 84,000 square miles in southern and central Arizona and small areas in California, Nevada, New Mexico, and Mexico (Anderson, 1980, p. 3). The area was divided into 72 basins for modeling purposes. Available data on which to base the models and the general analyses of the hydrologic systems were numerous in some basins and sparse in others. A major task of the Swab/RASA study was the accumulation and management of existing data on ground-water quantity and quality. The system was developed for use on local computer facilities where the data would be available for relatively quick and inexpensive access.

The large amount of available ground-water data present a significant data-management problem. In order to use the information effectively for model development, resource evaluation, and data reporting, an integrated data-handling, analysis, and plotting system is needed. This computer-based system would consist of a set of files for the different types of data; plotting and statistical routines; and retrieval, reporting, and editing programs (Anderson, 1980, p. 16-17).

This data-management system is used to supplement the U.S. Geological Survey's National Water Data Storage and Retrieval System (WATSTORE) by including a large amount of data from other sources.
The Swab/RASA data-management system represents an attempt to include all the reliable ground-water data for Arizona in a single data system.

**Purpose and Scope of Report**

The purpose of this report is to provide program documentation that will enable a user to implement and run the programs as the foundation of a ground-water data-management system. The scope of this report is limited to descriptions of the programs and data files and operational considerations for implementing the system.

**Relation to Goals**

One of the goals of the Swab/RASA study is to accumulate, analyze, and systematically file available ground-water data for use in ground-water modeling and ground-water resource evaluation. A data-management system that could provide efficient and relatively inexpensive access to ground-water data using local computer resources was necessary. During the 4 years of the Swab/RASA study, more than 100 computer programs were developed for use in the evaluation of ground-water data. The data were used to construct three-dimensional digital models of basins within the study area. The nine programs described in this report provide the updating and retrieval of the data file and areal plotting of data distributions.

The data used in the study are indexed by a master data file and stored in eight working files. The working files are segregated by general type of data—water-level data, water-quality data, and well-log data. The programs, which allow editing of changes to the master file, updating of the master file, and retrieval and reporting of all well-site data, form the nucleus of the data-management system. Programs that provide basic graphic representation of the data in the form of areal plots also are included. The relations between the system's programs and data files are shown in figure 1.

**National Water Data Storage and Retrieval System**

The National Water Data Storage and Retrieval System (WATSTORE) was established in November 1971 to modernize the U.S. Geological Survey's existing water-data processing procedures and techniques and to provide for the effective and efficient management of its release of information. The system is operated and maintained on the central computer facilities of the U.S. Geological Survey in Reston, Virginia.
Figure 1.--Relations between programs and data files in the data-management system.
The WATSTORE system consists of several files in which data are grouped and stored by common characteristics and frequency of data collections. The system is also designed to allow for the inclusion of additional data files. Files are maintained for the storage of (1) surface-water, water-quality, and ground-water data measured on a daily or continuous basis, (2) annual peak values for streamflow stations, (3) chemical analyses for surface-water and ground-water sites, and (4) geologic and inventory data for ground-water sites. An index file of sites for which data are stored in the system is also maintained (U.S. Geological Survey, 1976).

DESCRIPTION OF THE DATA-MANAGEMENT SYSTEM

The programs in this ground-water data-management system can be divided into three major parts: (1) utility computer programs, (2) computer programs for editing and updating, and (3) programs for data retrieval and reporting. The only utility program described, POLYGON, is used to define polygons for data retrieval. Three programs edit changes to be made to the master file, and two programs perform the actual updating of the master file. Two data-retrieval programs and one graphics program, which is used to plot areal distributions of data within a given area, are discussed. The data-management system was developed for batch processing.

Computer Languages

The choice of high-level computer languages to be used in the development of this data-management system was limited by several factors. The factors included (1) the standardization of the language(s), (2) efficiency of the language(s), and (3) portability of the programs.

The criteria that the language(s) used must be standardized by the American National Standards Institute (ANSI) narrowed the field of available languages to the Common Business Oriented Language (COBOL) and the FORMula TRANslating (FORTRAN) language. All programs were written in Control Data Corporation* (CDC) FORTRAN IV Extended, version 4.6 and (or) CDC COBOL, version 4.7.

In terms of efficiency of the language(s) used, FORTRAN has long been the workhorse of the scientific community for numerical computing. COBOL, however, was designed for handling large amounts of data input and output and has several string-handling and table- or array-handling capabilities.

*The use of trade names is for descriptive purposes only and does not constitute an endorsement by the U.S. Geological Survey.
COBOL and FORTRAN, either as basic or extended languages, are available on virtually all but the smallest computer systems. Thus, portability of programs written in these languages presents fewer problems, and the programs can be moved from one computer system to another with minor modifications.

All the editing, updating, and retrieval programs in the master file are written in COBOL. FORTRAN subroutines are used by two of the editing programs to handle the conversion of latitudes and longitudes from degrees, minutes, and seconds to decimal degrees. The polygon-definition program and the areal-plot program are written in FORTRAN with no COBOL interfaces. Great care was taken during the development of the data-management system programs to insure that all source-code statements adhere to the ANSI standards for each language.

### Variable-Length Records

All data files maintained by this system are made up of records with a variable-length or trailer-type record format. Each record consists of two major parts—a 22-character fixed-length record header and a variable-length part, or trailer, that may contain from 1 to 999 characters.

The fixed-length record header contains a site-identification number; a three-digit WATSTORE-type record identifier; a three-digit number, the value of which determines the length of the variable part of the record; and a one-character security or access code. The record-type identifiers currently in use in the WATSTORE system are listed in table 1. The fixed-length record header for a variable-length record has the following format:

<table>
<thead>
<tr>
<th>Position</th>
<th>Variable</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-15</td>
<td>KEYID</td>
<td>9(15)</td>
<td>Site identifier</td>
</tr>
<tr>
<td>16-18</td>
<td>RNUM</td>
<td>999</td>
<td>Record identifier or repeating group number</td>
</tr>
<tr>
<td>19-21</td>
<td>LNUM</td>
<td>999</td>
<td>Length of variable portion in characters</td>
</tr>
<tr>
<td>22</td>
<td>SESAFE</td>
<td>X</td>
<td>Access and (or) security code; may be used to limit access to data record</td>
</tr>
</tbody>
</table>

The variable-length trailer part of the record contains as many character positions as are indicated in the length field of the record header. For example, if the length field of the record header has a value of 18, the variable-length trailer will have 18 character positions.
Table 1.--Record-identification numbers in use by WATSTORE

<table>
<thead>
<tr>
<th>Record number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>GWSI entry record</td>
</tr>
<tr>
<td>001</td>
<td>QW header record</td>
</tr>
<tr>
<td>002</td>
<td>QW name record</td>
</tr>
<tr>
<td>003</td>
<td>QW parameter record</td>
</tr>
<tr>
<td>042</td>
<td>Lift data</td>
</tr>
<tr>
<td>047</td>
<td>Major pump data</td>
</tr>
<tr>
<td>055</td>
<td>Standby power data</td>
</tr>
<tr>
<td>058</td>
<td>Construction data</td>
</tr>
<tr>
<td>072</td>
<td>Hole dimensions</td>
</tr>
<tr>
<td>082</td>
<td>Openings data</td>
</tr>
<tr>
<td>076</td>
<td>Casings data</td>
</tr>
<tr>
<td>164</td>
<td>Minor repairs</td>
</tr>
<tr>
<td>090</td>
<td>Geohydrologic unit data</td>
</tr>
<tr>
<td>094</td>
<td>Aquifer data</td>
</tr>
<tr>
<td>098</td>
<td>Hydraulic data</td>
</tr>
<tr>
<td>105</td>
<td>Coefficients</td>
</tr>
<tr>
<td>114</td>
<td>Quality network</td>
</tr>
<tr>
<td>121</td>
<td>Level network</td>
</tr>
<tr>
<td>127</td>
<td>Pumpage network</td>
</tr>
<tr>
<td>134</td>
<td>Flow data</td>
</tr>
<tr>
<td>146</td>
<td>Pump-production data</td>
</tr>
<tr>
<td>158</td>
<td>Owners</td>
</tr>
<tr>
<td>171</td>
<td>Springs</td>
</tr>
<tr>
<td>183</td>
<td>Remarks</td>
</tr>
<tr>
<td>186</td>
<td>Site visits</td>
</tr>
<tr>
<td>180</td>
<td>Other data</td>
</tr>
<tr>
<td>189</td>
<td>Other identifiers</td>
</tr>
<tr>
<td>192</td>
<td>Field water quality</td>
</tr>
<tr>
<td>198</td>
<td>Available log data</td>
</tr>
<tr>
<td>203</td>
<td>Well group</td>
</tr>
<tr>
<td>208</td>
<td>Pond-Tunnel-Drain</td>
</tr>
<tr>
<td>212</td>
<td>Cooperator data</td>
</tr>
<tr>
<td>219</td>
<td>Laterals</td>
</tr>
<tr>
<td>250</td>
<td>Miscellaneous values</td>
</tr>
<tr>
<td>277</td>
<td>State water use</td>
</tr>
<tr>
<td>269</td>
<td>Observation-well heading</td>
</tr>
<tr>
<td>320</td>
<td>Measuring point</td>
</tr>
</tbody>
</table>

Defining New Record Types

Data from sources other than U.S. Geological Survey can be entered into all WATSTORE data files, except the quality-of-water file.
A great deal of time and money, however, must be spent editing, reformating, and reentering the data in USGS formats. The use of variable-length records allows non-WATSTORE data to be entered directly into the data files in the same format as that in which the data are received. This is accomplished by creating a record header (see section entitled "Variable-Length Records") for the new data and attaching the new data to a record header.

A new record type is defined in the following manner: A record-type number not in use by the WATSTORE system is assigned to the new data record, and the length of the data part of the record is placed in the length field of the record header. After the new record has been defined, a COBOL description of the new record is written for insertion into the programs that will handle the new record type. The process of creating a new record type from non-USGS sources is illustrated in figure 2.

Data-File Storage

Data files in the data-management system are stored on magnetic tape. This storage medium was chosen because of the large sizes of the data files and the high cost of storing large amounts of data on an on-line storage device, such as magnetic disk. Although magnetic tapes do not allow the immediate access that magnetic disks allow, they do provide access times that are suitable for the needs of ground-water modeling and resource-evaluation studies.

Control Records for COBOL Programs

All COBOL programs developed as part of the Swab/RASA study make use of five control records for (1) controlling a program's function, (2) inputting the main and secondary titling information, and (3) signaling an end of the program's input data. The control records are identified by the letters CNL in positions 1 through 3 and a three-digit number in positions 4 through 6. The numbers 001 through 009 are reserved for controlling program functions.

In addition to the function-control number, the first control record may contain an optional two-digit number in positions 7-8, indicating the number of data files to be read by the program; however, this number is used only in the EDIT-1 and EDIT-2 programs. The remaining 72 positions of the first control record are available for comments or a brief description of the data being processed.

Three control records are used to carry information for printed page headings. These three title records are identified by the numbers 010, 011, and 012 in positions 4-6. Control record 010 is used for the principal title for the printed listing or report. There are 74 character
Figure 2.--Process of creating a new master-file record type with COBOL record description.
positions available for the main title. The next record, 011, is a name record used to identify the person or agency requesting the run and the person or agency that is the source of the data. The third record, 012, is used to place a subtitle on the listing or report.

The last common control record contains 999 in positions 4 through 6 and indicates to the program that all input data have been processed. The formats for these five control records are:

<table>
<thead>
<tr>
<th>Record number</th>
<th>Positions</th>
<th>Variable name</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-3</td>
<td>CNL-ID</td>
<td>XXX</td>
<td>Must be CNL</td>
</tr>
<tr>
<td></td>
<td>4-6</td>
<td>CNL-NUM</td>
<td>999</td>
<td>Must be 001 through 009</td>
</tr>
<tr>
<td></td>
<td>7-8</td>
<td>CNL-FLE</td>
<td>99</td>
<td>Number of files to be read from input</td>
</tr>
<tr>
<td></td>
<td>9-80</td>
<td>CNL-REST</td>
<td>X(72)</td>
<td>Available for comments or brief description of data or program run</td>
</tr>
<tr>
<td>2</td>
<td>1-3</td>
<td>CNL-ID</td>
<td>XXX</td>
<td>Must be CNL</td>
</tr>
<tr>
<td></td>
<td>4-6</td>
<td>CNL-NUM</td>
<td>999</td>
<td>Must be 010</td>
</tr>
<tr>
<td></td>
<td>7-80</td>
<td>TITLEX</td>
<td>X(74)</td>
<td>Main title for printed page headings</td>
</tr>
<tr>
<td>3</td>
<td>1-3</td>
<td>CNL-ID</td>
<td>XXX</td>
<td>Must be CNL</td>
</tr>
<tr>
<td></td>
<td>4-6</td>
<td>CNL-NUM</td>
<td>999</td>
<td>Must be 011</td>
</tr>
<tr>
<td></td>
<td>7-36</td>
<td>GNAME</td>
<td>X(30)</td>
<td>Person initiating program run</td>
</tr>
<tr>
<td></td>
<td>37-66</td>
<td>CNAME</td>
<td>X(30)</td>
<td>Person or agency requesting program run</td>
</tr>
<tr>
<td></td>
<td>67-80</td>
<td>NOT USED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1-3</td>
<td>CNL-ID</td>
<td>XXX</td>
<td>Must be CNL</td>
</tr>
<tr>
<td></td>
<td>4-6</td>
<td>CNL-NUM</td>
<td>999</td>
<td>Must be 012</td>
</tr>
<tr>
<td></td>
<td>7-66</td>
<td>SRCAGC</td>
<td>X(60)</td>
<td>Subtitle for printed page headings</td>
</tr>
<tr>
<td></td>
<td>67-80</td>
<td>NOT USED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1-3</td>
<td>CNL-ID</td>
<td>XXX</td>
<td>Must be CNL</td>
</tr>
<tr>
<td></td>
<td>4-6</td>
<td>CNL-NUM</td>
<td>999</td>
<td>Must be 999</td>
</tr>
<tr>
<td></td>
<td>7-80</td>
<td>NOT USED</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Audit Trail**

To insure that all changes to the master file can be accounted for, an audit trail is established by the first edit program. This audit trail is updated by each successive run of one of the edit or update
programs. Audit-trail information includes the Julian dates of the present processing run and the previous processing run and the output-sequence numbers for each record accessed during the present and previous processing runs.

The audit trail allows a user to trace any given change entered for edit and update processing from its original input through the final update processing or from the final update processing back to an original input record. Audit information also includes counts of records input to a program, the number of records output, the number of lines printed, the number of input files read, and the number of records written to error files.

DATA FILE DESCRIPTIONS

The data files maintained by this system include the master data file and eight working files (fig. 1):

- Ground-water data file
- Water-quality data file
- Water-level data file
- Well-construction data file
- Water-use data file
- Aquifer-test data file
- Well-log data file
- Geophysical data file

Data contained in the working files are consolidated into a single record type from one or more of the records contained in the master file for a given site. The working files contain several tens of thousands of records less than the master file and are used for data retrieval and reporting whenever possible to reduce processing time and cost.

Master Data File

The ground-water master data file contains available data for all ground-water sites within the State of Arizona for which reliable data could be collected. Currently, 26 WATSTORE record types are in use in this file (table 2). In addition, data records from city, county, State, and other Federal agencies and from private firms are added as the data are available.

The master file began as a magnetic tape copy of the WATSTORE Ground-Water Site Inventory (GWSI) file for Arizona. The original tape was reviewed to insure that all sites for which data were present had a GWSI entry record (WATSTORE record type = 000). This check is necessary because the WATSTORE system requires each site to have a header record in its data files before any other data and (or)
Table 2.--WATSTORE identification numbers and record titles used in the Swab/RASA ground-water master file

<table>
<thead>
<tr>
<th>WATSTORE code</th>
<th>Total length</th>
<th>Variable length</th>
<th>Record description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>251</td>
<td>229</td>
<td>General site data</td>
</tr>
<tr>
<td>001</td>
<td>83</td>
<td>61</td>
<td>Water-quality header record</td>
</tr>
<tr>
<td>002</td>
<td>87</td>
<td>65</td>
<td>Water-quality name record</td>
</tr>
<tr>
<td>003</td>
<td>94</td>
<td>72</td>
<td>Water-quality parameter record</td>
</tr>
<tr>
<td>022</td>
<td>53</td>
<td>31</td>
<td>Opening data</td>
</tr>
<tr>
<td>042</td>
<td>47</td>
<td>25</td>
<td>Lift data</td>
</tr>
<tr>
<td>047</td>
<td>217</td>
<td>195</td>
<td>Major pump data</td>
</tr>
<tr>
<td>055</td>
<td>29</td>
<td>7</td>
<td>Standby power supply</td>
</tr>
<tr>
<td>058</td>
<td>86</td>
<td>64</td>
<td>Well-construction data</td>
</tr>
<tr>
<td>072</td>
<td>41</td>
<td>19</td>
<td>Dimensions of the hole constructed</td>
</tr>
<tr>
<td>076</td>
<td>41</td>
<td>19</td>
<td>Casing data</td>
</tr>
<tr>
<td>090</td>
<td>81</td>
<td>59</td>
<td>Geohydrologic unit descriptors</td>
</tr>
<tr>
<td>094</td>
<td>40</td>
<td>18</td>
<td>Aquifer data</td>
</tr>
<tr>
<td>114</td>
<td>38</td>
<td>16</td>
<td>Water-quality network data</td>
</tr>
<tr>
<td>121</td>
<td>37</td>
<td>15</td>
<td>Water-level data</td>
</tr>
<tr>
<td>127</td>
<td>38</td>
<td>16</td>
<td>Pumpage or withdrawal data</td>
</tr>
<tr>
<td>134</td>
<td>72</td>
<td>50</td>
<td>Production data, flowing</td>
</tr>
<tr>
<td>146</td>
<td>74</td>
<td>52</td>
<td>Production data, pumped</td>
</tr>
<tr>
<td>158</td>
<td>112</td>
<td>90</td>
<td>Owner-identification record</td>
</tr>
<tr>
<td>171</td>
<td>74</td>
<td>52</td>
<td>Other spring data</td>
</tr>
<tr>
<td>180</td>
<td>64</td>
<td>42</td>
<td>Other data available</td>
</tr>
<tr>
<td>183</td>
<td>70</td>
<td>48</td>
<td>Pertinent remarks</td>
</tr>
<tr>
<td>186</td>
<td>70</td>
<td>48</td>
<td>Site visit data</td>
</tr>
<tr>
<td>189</td>
<td>102</td>
<td>80</td>
<td>Other site identification</td>
</tr>
<tr>
<td>198</td>
<td>38</td>
<td>16</td>
<td>Available log data</td>
</tr>
<tr>
<td>234</td>
<td>40</td>
<td>18</td>
<td>Water-level data schedule</td>
</tr>
</tbody>
</table>

NOTE: Detailed descriptions of all record types can be found in the WATSTORE User's Guide, Volume 2, Chapter 2.
records can be entered for a particular site. The same method is used in this data-management system. A type "000" record must exist for each site in the master data file before any other record types can be entered in the file.

The next step was to obtain magnetic tape copies of the water-quality and water-level files for the State of Arizona. Checks were made of the water-quality tapes to insure that each site that had water-quality data also had a water-quality header record. The water-level tape was checked against the GWSI tape to insure that every site that had a water-level measurement also had a GWSI entry record in the GWSI file. These four tapes—the ground-water site inventory tape, the water-quality header file tape, the water-quality history file tape, and the water-level tape—were then sorted into descending order using the site-identification numbers. The site-identification number is composed of the latitude and longitude of the site—degrees, minutes, and seconds (DMS)—and a two-digit sequence number to differentiate wells that have the same latitude and longitude. The sorting of records is performed on descending latitude, descending longitude, ascending site-identifier, ascending record-type identifier, and sequence number. The four files were merged into a single ground-water master data file. This first copy of the master file contained about 267,000 records that represented about 26,000 ground-water sites.

**Working Data Files**

*Ground-Water Data File*

The working data file is composed of records of a fixed-length structure that have a length of 270 characters. These records can be handled by the programs as either fixed-length or variable-length records. Each record is a summary of the available ground-water data for a given site. The contents of the ground-water record were determined by an analysis of requests for information that have been handled by the Water Resources Division's Arizona District data-processing unit. Where multiple values may exist for a particular data element in the master file, the most recent value is stored in the ground-water working data file. The file is in descending order by the site-identification number. The format of the ground-water working data file is:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-15</td>
<td>9(15)</td>
<td>Site-identification number</td>
</tr>
<tr>
<td>16-18</td>
<td>999</td>
<td>Record-type identification number</td>
</tr>
<tr>
<td>19-21</td>
<td>999</td>
<td>Length of variable portion</td>
</tr>
<tr>
<td>22</td>
<td>X</td>
<td>Data security or access code</td>
</tr>
<tr>
<td>23-42</td>
<td>X(20)</td>
<td>Local site identifier</td>
</tr>
<tr>
<td>43-48</td>
<td>999999</td>
<td>Latitude of site (decimal degrees)</td>
</tr>
<tr>
<td>Field ID</td>
<td>Code</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>49-55</td>
<td>999v9999</td>
<td>Longitude of site (decimal degrees)</td>
</tr>
<tr>
<td>56</td>
<td>X</td>
<td>Type of site</td>
</tr>
<tr>
<td>57-66</td>
<td>X(10)</td>
<td>Ground-water area code</td>
</tr>
<tr>
<td>67-69</td>
<td>999</td>
<td>County code</td>
</tr>
<tr>
<td>70-76</td>
<td>9(5)v99</td>
<td>Altitude of site</td>
</tr>
<tr>
<td>77</td>
<td>X</td>
<td>Primary use of site</td>
</tr>
<tr>
<td>78</td>
<td>X</td>
<td>Primary use of water</td>
</tr>
<tr>
<td>79-85</td>
<td>9(5)v99</td>
<td>Depth of hole</td>
</tr>
<tr>
<td>86-92</td>
<td>9(5)v99</td>
<td>Depth of well</td>
</tr>
<tr>
<td>93-99</td>
<td>9(5)v99</td>
<td>Water level (most recent)</td>
</tr>
<tr>
<td>100-107</td>
<td>X(8)</td>
<td>Date water level measured</td>
</tr>
<tr>
<td>108</td>
<td>X</td>
<td>Method of measurement</td>
</tr>
<tr>
<td>109</td>
<td>X</td>
<td>Site status</td>
</tr>
<tr>
<td>110-117</td>
<td>X(8)</td>
<td>Date of construction completion</td>
</tr>
<tr>
<td>118-127</td>
<td>X(10)</td>
<td>Other site identifiers</td>
</tr>
<tr>
<td>128-142</td>
<td>X(15)</td>
<td>Assigner of other identifiers</td>
</tr>
<tr>
<td>143-150</td>
<td>X(8)</td>
<td>Date of most recent chemical analysis</td>
</tr>
<tr>
<td>165</td>
<td>X</td>
<td>Type of finish</td>
</tr>
<tr>
<td>166-172</td>
<td>9(5)v99</td>
<td>Depth to top of casing</td>
</tr>
<tr>
<td>173-179</td>
<td>9(5)v99</td>
<td>Depth to bottom of casing</td>
</tr>
<tr>
<td>180-183</td>
<td>9(3)v99</td>
<td>Diameter of the casing</td>
</tr>
<tr>
<td>185-191</td>
<td>9(5)v99</td>
<td>Depth to top of opening</td>
</tr>
<tr>
<td>192-198</td>
<td>9(5)v99</td>
<td>Depth to bottom of opening</td>
</tr>
<tr>
<td>199</td>
<td>X</td>
<td>Type of opening</td>
</tr>
<tr>
<td>200-207</td>
<td>X(8)</td>
<td>Date of production measurement</td>
</tr>
<tr>
<td>208-214</td>
<td>9(5)v99</td>
<td>Discharge</td>
</tr>
<tr>
<td>215-220</td>
<td>9(4)v99</td>
<td>Drawdown</td>
</tr>
<tr>
<td>221</td>
<td>X</td>
<td>Method of production data measurement</td>
</tr>
<tr>
<td>222-228</td>
<td>9(5)v99</td>
<td>Production water level</td>
</tr>
<tr>
<td>229-235</td>
<td>9(5)v99</td>
<td>Static water level</td>
</tr>
<tr>
<td>236-242</td>
<td>9(5)v99</td>
<td>Specific capacity</td>
</tr>
<tr>
<td>243</td>
<td>X</td>
<td>Type of lift</td>
</tr>
<tr>
<td>244-251</td>
<td>X(8)</td>
<td>Available log types</td>
</tr>
<tr>
<td>252-259</td>
<td>X(8)</td>
<td>Geohydrologic unit identifier</td>
</tr>
<tr>
<td>260</td>
<td>X</td>
<td>Contributing unit</td>
</tr>
<tr>
<td>261-266</td>
<td>9(5)</td>
<td>Number of water-level measurements</td>
</tr>
</tbody>
</table>

**Water-Quality Data File**

The water-quality working data file is sorted in descending order using the site-identification number. Each record is of variable-length structure and may range from 30 to 631 characters in length.
Each record represents a single site and the working-file data is a composite of all the water-quality data for that site for a particular date in the master file. The water-quality working file records have the following format:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-15</td>
<td>9(15)</td>
<td>Site-identification number</td>
</tr>
<tr>
<td>16-18</td>
<td>999</td>
<td>Record-type identification number</td>
</tr>
<tr>
<td>19-21</td>
<td>999</td>
<td>Length of variable portion</td>
</tr>
<tr>
<td>22</td>
<td>X</td>
<td>Data security or access code</td>
</tr>
<tr>
<td>23-24</td>
<td>99</td>
<td>Number of parameters in this record (0-50)</td>
</tr>
<tr>
<td>25-30</td>
<td>X(6)</td>
<td>Date of most recent analysis</td>
</tr>
<tr>
<td>31-36</td>
<td>9(5)</td>
<td>Parameter code 1</td>
</tr>
<tr>
<td>37-41</td>
<td>v9(5)</td>
<td>Parameter value 1</td>
</tr>
<tr>
<td>42-43</td>
<td>99</td>
<td>Parameter value exponent 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>620-624</td>
<td>9(5)</td>
<td>Parameter code 50</td>
</tr>
<tr>
<td>625-629</td>
<td>v9(5)</td>
<td>Parameter value 50</td>
</tr>
<tr>
<td>630-631</td>
<td>99</td>
<td>Parameter value exponent 50</td>
</tr>
</tbody>
</table>

A single water-quality record may contain values for at least 1 but not more than 50 chemical parameters. A parameter value in the working file is the arithmetic mean of all values for that parameter for a given site in the master file. The water-quality working file record has a value for every constituent that was recorded for a given site in the master file even though a particular constituent may not appear in every analysis.

**Water-Level Data File**

This sequential file contains all water-level measurement records (WATSTORE record type = 234) that have been entered into the master file. The records in this file are 40 characters in length and are fixed-length in structure. The format of the record allows them to be processed as either fixed-length or variable-length records. The file is sorted into descending order using the site-identification number as the
primary sort key and the date measured as the secondary sort key. Water-level records have this format:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th><strong>WATSTORE code</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1- 15</td>
<td>9(15)</td>
<td>C1</td>
<td>15-digit site-identification number</td>
</tr>
<tr>
<td>16- 18</td>
<td>999</td>
<td></td>
<td>Record-type identification number</td>
</tr>
<tr>
<td>19- 21</td>
<td>999</td>
<td></td>
<td>Length of variable portion</td>
</tr>
<tr>
<td>22</td>
<td>X</td>
<td></td>
<td>Data security or access code</td>
</tr>
<tr>
<td>23- 30</td>
<td>X(8)</td>
<td>C235</td>
<td>Date on which the water level was measured</td>
</tr>
<tr>
<td>31</td>
<td>X</td>
<td>C236</td>
<td>Accuracy of the date</td>
</tr>
<tr>
<td>32- 37</td>
<td>9(4)v99</td>
<td>C237</td>
<td>Water-level measurement</td>
</tr>
<tr>
<td>38</td>
<td>X</td>
<td>C238</td>
<td>Site status at time of measurement</td>
</tr>
<tr>
<td>39</td>
<td>X</td>
<td>C239</td>
<td>Method of measurement</td>
</tr>
<tr>
<td>40</td>
<td>X</td>
<td>C240</td>
<td>Accuracy of the measurement</td>
</tr>
</tbody>
</table>

**Well-Construction Data File**

The well-construction data file contains as much reliable data pertaining to the construction and alteration of wells as can be obtained from contractors' records.

**Water-Use Data File**

The water-use data file is composed of 90-character records. The records in this file represent power consumption and reported pumpage data received from public and private sources. Each record contains data for type of power used, amount of power used, estimated or measured pumpage, and the primary use of the water pumped. The file is sorted into descending order using the site-identification number and the
year of pumpage as the primary and secondary sort keys, respectively. Water-use records have the following format:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-15</td>
<td>9(15)</td>
<td>Site-identification number</td>
</tr>
<tr>
<td>16-18</td>
<td>9(3)</td>
<td>Record number</td>
</tr>
<tr>
<td>19-21</td>
<td>9(3)</td>
<td>Length of variable part</td>
</tr>
<tr>
<td>22</td>
<td>X(1)</td>
<td>Security or access code</td>
</tr>
<tr>
<td>23</td>
<td>X(1)</td>
<td>Quadrant</td>
</tr>
<tr>
<td>24</td>
<td>X(1)</td>
<td>Half-quadrant</td>
</tr>
<tr>
<td>25-26</td>
<td>X(2)</td>
<td>Township</td>
</tr>
<tr>
<td>27</td>
<td>X(1)</td>
<td>Half-township</td>
</tr>
<tr>
<td>28-29</td>
<td>X(2)</td>
<td>Range</td>
</tr>
<tr>
<td>30</td>
<td>X(1)</td>
<td>Half-range</td>
</tr>
<tr>
<td>31-32</td>
<td>X(2)</td>
<td>Section</td>
</tr>
<tr>
<td>33</td>
<td>X(1)</td>
<td>Quarter-section</td>
</tr>
<tr>
<td>34</td>
<td>X(1)</td>
<td>Quarter-quarter-section</td>
</tr>
<tr>
<td>35</td>
<td>X(1)</td>
<td>Quarter-quarter-quarter-section</td>
</tr>
<tr>
<td>36</td>
<td>X(1)</td>
<td>Quarter-quarter-quarter-quarter-section</td>
</tr>
<tr>
<td>37-39</td>
<td>X(3)</td>
<td>Ground-water area code</td>
</tr>
<tr>
<td>40-45</td>
<td>X(6)</td>
<td>Owning agency</td>
</tr>
<tr>
<td>46-48</td>
<td>X(3)</td>
<td>Energy source</td>
</tr>
<tr>
<td>49-54</td>
<td>X(6)</td>
<td>Meter number</td>
</tr>
<tr>
<td>55-69</td>
<td>X(15)</td>
<td>Account number</td>
</tr>
<tr>
<td>70-71</td>
<td>X(2)</td>
<td>Revenue class</td>
</tr>
<tr>
<td>72-73</td>
<td>X(2)</td>
<td>Reporting year</td>
</tr>
<tr>
<td>74-81</td>
<td>9(8)</td>
<td>Power usage</td>
</tr>
<tr>
<td>82-83</td>
<td>X(2)</td>
<td>Type of power used</td>
</tr>
<tr>
<td>84-88</td>
<td>9(4)v9</td>
<td>Divisor</td>
</tr>
<tr>
<td>89-90</td>
<td>X(2)</td>
<td>Year divisor was determined</td>
</tr>
<tr>
<td>91</td>
<td>X(1)</td>
<td>Source of divisor</td>
</tr>
<tr>
<td>92-97</td>
<td>9(6)</td>
<td>Pumpage, in acre-feet per year</td>
</tr>
<tr>
<td>98-99</td>
<td>X(2)</td>
<td>Primary use of water</td>
</tr>
</tbody>
</table>

Aquifer-Test Data File

The aquifer-test data file is composed of data from numerous public and private sources. The records are of variable-length structure and contain the results of from one to six individual aquifer tests or steps of a variable-discharge step-drawdown test. The file is sorted into ascending alphabetic order using the basin or ground-water-area identifier. Within a particular ground-water area, the data are sorted in descending order using the site-identification number and the date of testing as the primary and secondary sort keys. The aquifer-test records are 332 characters in length and have the following format:
<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-15</td>
<td>9(15)</td>
<td>Site-identification number of pumped well</td>
</tr>
<tr>
<td>16-18</td>
<td>9(3)</td>
<td>Record number (800)</td>
</tr>
<tr>
<td>19-21</td>
<td>9(3)</td>
<td>Length of variable portion</td>
</tr>
<tr>
<td>22</td>
<td>X(1)</td>
<td>Security or access code</td>
</tr>
<tr>
<td>23-36</td>
<td>X(14)</td>
<td>Local site identifier of pumped well</td>
</tr>
<tr>
<td>37-42</td>
<td>X(6)</td>
<td>Basin identifier</td>
</tr>
<tr>
<td>43-44</td>
<td>X(2)</td>
<td>Type of test</td>
</tr>
<tr>
<td>45-52</td>
<td>9(8)</td>
<td>Beginning date of test</td>
</tr>
<tr>
<td>53-56</td>
<td>9(4)</td>
<td>Static water level, in feet below land surface</td>
</tr>
<tr>
<td>57-60</td>
<td>9(4)</td>
<td>Discharge measurement 1, in gallons per minute</td>
</tr>
<tr>
<td>61-65</td>
<td>9(4)v9</td>
<td>Duration of discharge 1, in hours</td>
</tr>
<tr>
<td>66-69</td>
<td>9(4)</td>
<td>Discharge measurement 2, in gallons per minute</td>
</tr>
<tr>
<td>70-74</td>
<td>9(4)v9</td>
<td>Duration of discharge 2, in hours</td>
</tr>
<tr>
<td>75-78</td>
<td>9(4)</td>
<td>Discharge measurement 3, in gallons per minute</td>
</tr>
<tr>
<td>79-83</td>
<td>9(4)v9</td>
<td>Duration of discharge 3, in hours</td>
</tr>
<tr>
<td>84-87</td>
<td>9(4)</td>
<td>Discharge measurement 4, in gallons per minute</td>
</tr>
<tr>
<td>88-92</td>
<td>9(4)v9</td>
<td>Duration of discharge 4, in hours</td>
</tr>
<tr>
<td>93-96</td>
<td>9(4)</td>
<td>Discharge measurement 5, in gallons per minute</td>
</tr>
<tr>
<td>97-101</td>
<td>9(4)v9</td>
<td>Duration of discharge 5, in hours</td>
</tr>
<tr>
<td>102-105</td>
<td>9(4)</td>
<td>Discharge measurement 6, in gallons per minute</td>
</tr>
<tr>
<td>106-110</td>
<td>9(4)v9</td>
<td>Duration of discharge 6, in hours</td>
</tr>
<tr>
<td>111-114</td>
<td>9(4)</td>
<td>Production water level, in feet below land surface</td>
</tr>
<tr>
<td>115-121</td>
<td>X(7)</td>
<td>Source of data</td>
</tr>
<tr>
<td>122</td>
<td>X(1)</td>
<td>Accuracy of water-level measurement</td>
</tr>
<tr>
<td>123</td>
<td>X(1)</td>
<td>Accuracy of discharge measurement</td>
</tr>
<tr>
<td>124-128</td>
<td>9(5)</td>
<td>Depth of pumped well, in feet</td>
</tr>
<tr>
<td>129-133</td>
<td>9(5)</td>
<td>Altitude of pumped well</td>
</tr>
<tr>
<td>134-138</td>
<td>9(5)</td>
<td>Depth to top of open section, in feet below land surface</td>
</tr>
<tr>
<td>139-143</td>
<td>9(5)</td>
<td>Depth to bottom of open section, in feet below land surface</td>
</tr>
<tr>
<td>144</td>
<td>X(1)</td>
<td>Type of opening</td>
</tr>
<tr>
<td>145-149</td>
<td>9(5)</td>
<td>Radial distance of pumped well, from observation well, in feet</td>
</tr>
<tr>
<td>150-164</td>
<td>9(15)</td>
<td>Site identifier of observation well</td>
</tr>
<tr>
<td>165-171</td>
<td>9(7)</td>
<td>Transmissivity 1, in feet squared per day</td>
</tr>
<tr>
<td>172-177</td>
<td>v9(6)</td>
<td>Storage coefficient 1</td>
</tr>
</tbody>
</table>
Well-Log Data File

This file is made up of data from drillers' logs. The data are coded from copies of the driller's log and contain the driller's comments and measurements concerning the lithology encountered during construction of the well. Programs were developed as part of the Swab/RASA study whereby weighted average specific yield, hydraulic conductivity, and transmissivity could be estimated on the basis of data from drillers' logs. This in turn provided preliminary data for input to hydrologic models. The file is sorted into descending order using the site-identification number and date as primary and secondary sort keys, respectively.

Geophysical Data File

This file contains digitized data for the following types of geophysical logs:

- Gamma
- Neutron
- Density
- Resistance
- Resistivity, short
- Resistivity, long
- SP
- Caliper
- Temperature
The file is sorted into descending order using the site-
identification number, ascending alphabetic order using the log-type code,
and descending order using the date of the log.

UTILITY COMPUTER PROGRAMS

Polygon-Definition Program—POLYGON

POLYGON is a FORTRAN program for computing a mathematical
description of a given polygon. This description is used as a means of
determining the existence of data sites within an area defined by a
polygon for use in master file edit, update, and retrieval programs. The
program can handle polygons defined by as many as 1,999 points.

Program Operation

The program reads a series of points of latitude and longitude
that define a boundary of a polygon as shown in figure 3A. The points
must be input in sequential order around the periphery of the polygon in
either a clockwise or counter-clockwise order. After all points have been
read and stored in an array or table, the program closes the polygon by
copying the coordinates of the first point into the position after the
coordinates for the last point. For example, if a polygon is defined by
67 points, the coordinates of the first point will be copied to array
position 68 to close the polygon.

The next step is to mathematically define the straight line
segments that connect the points that make up the polygon's boundary. Consecutive points are paired to form directed line segments. The point
that has the lower latitude of a pair becomes the beginning point of the
line segment and the point with the higher latitude becomes the ending
point of the line segment. If the high and low latitudes are equal, the
point with the westernmost longitude becomes the line segment's beginning
point. The mathematical equation for each line segment is computed. The
line segments are then sorted in ascending order using the latitude
of the beginning point as the sort key.

The program now views the polygon as a series of slices. Each
slice is examined from left to right, additional points of corresponding
latitude are added on the opposite side of the polygon, and additional line
segments are defined in order to more completely define the polygon's
shape. The addition of these extra points and line segments complete the
definition of the polygon as a series of slices made up of trapezoids and
triangles as shown in figure 3B. The line segments are sorted a final
time in descending order by latitude and longitude of the beginning
points, and a polygon-definition table is written to the program's output
storage device (table 3).
A. As described by input data points.

B. As defined by the POLYGON program as a series of slices composed of triangles and trapezoids.

Figure 3.--Sample polygons.
Table 3. Example of "slice table" of line-segment equations for the polygon shown in figure 3

<table>
<thead>
<tr>
<th>Sample Polygon Input Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>32160011000003300011100002</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Equations of the Line Segments</th>
</tr>
</thead>
</table>

The equation for each directed line segment is solved by the program. Given the coefficients for a line segment and either a latitude or longitude from any point along that line, the missing coordinate can be computed. For example, a polygon may be plotted in such a way that a portion of the polygon boundary would fall outside the limits of the plotting surface (top, bottom, left, or right side). The program that produces the plotter's instructions has enough information to determine at what point along the boundary of the polygon the line to be drawn would intersect the limits of the plotting area or surface. The longitude (sides) or latitude (top or bottom) of the plot area are known by the plotting program. The coefficients for the line segment that intersects one of these four boundaries are also known. The plot program has enough information to compute the unknown latitude or longitude of the point of intersection of the polygon boundary and the plot area boundary. The general system logic of POLYGON is illustrated in figure 4.

The line-segment equation is of the form $Ax + By = C$. Solution for the coefficients $A$, $B$, and $C$ of each line segment in the POLYGON program is accomplished by solving the following equations:

$$A = y_2 - y_1$$

$$B = x_2 - x_1$$

$$C = (x_2y_1 - x_1y_2) + (x_1y_2 - x_2y_1)$$
Figure 4.--General system logic for program POLYGON.
where

\[ x_1 = \text{longitude of the line's beginning point}, \]
\[ x_2 = \text{longitude of the line's ending point}, \]
\[ y_1 = \text{latitude of the line's beginning point}, \]
\[ y_2 = \text{latitude of the line's ending point}. \]

To solve for an unknown longitude (x):

\[ x = \frac{C - By}{A} \]

where

\[ y = \text{known latitude and} \]
\[ A \neq 0. \]

To solve for an unknown latitude (y):

\[ y = \frac{Ax - C}{-B} \]

where

\[ x = \text{known longitude and} \]
\[ B \neq 0. \]

NOTE: Longitudes between 0° and 180° W. are expressed as negative values.

Program Input

Input to POLYGON consists of a polygon-identification record and a latitude-longitude record for each point to be defined on the polygon boundary. Latitudes and longitudes are entered in degrees, minutes, and seconds. The format for the polygon-identification record and the latitude-longitude records are:
Program Output

POLYGON produces two output files: (1) a printed listing of the line-segment table and (2) a disk file containing the line-segment table (table 3). Each output record represents a single directed line segment and has the following format:

<table>
<thead>
<tr>
<th>Position</th>
<th>Variable</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>1X</td>
<td>Blank</td>
</tr>
<tr>
<td>2-7</td>
<td>LA1</td>
<td>I6</td>
<td>Latitude, DMS of beginning point</td>
</tr>
<tr>
<td>8-14</td>
<td>LN1</td>
<td>I7</td>
<td>Longitude, DMS of beginning point</td>
</tr>
<tr>
<td>15-20</td>
<td>LA2</td>
<td>I6</td>
<td>Latitude, DMS of ending point</td>
</tr>
<tr>
<td>21-27</td>
<td>LN2</td>
<td>I7</td>
<td>Longitude, DMS of ending point</td>
</tr>
<tr>
<td>28-33</td>
<td>KY</td>
<td>I6</td>
<td>Latitude, in decimal degrees of beginning point</td>
</tr>
<tr>
<td>34</td>
<td>I1</td>
<td>A1</td>
<td>Sign of latitude</td>
</tr>
<tr>
<td>35-41</td>
<td>KX</td>
<td>I7</td>
<td>Longitude, in decimal degrees of beginning point</td>
</tr>
<tr>
<td>42-47</td>
<td>KW</td>
<td>I6</td>
<td>Latitude, in decimal degrees of ending point</td>
</tr>
<tr>
<td>48</td>
<td>I2</td>
<td>A1</td>
<td>Sign of latitude value</td>
</tr>
<tr>
<td>49-55</td>
<td>KZ</td>
<td>I7</td>
<td>Longitude, in decimal degrees of ending point</td>
</tr>
<tr>
<td>56</td>
<td>I3</td>
<td>A1</td>
<td>Sign of coefficient A</td>
</tr>
<tr>
<td>57-64</td>
<td>KA</td>
<td>I8</td>
<td>Coefficient A of line equation</td>
</tr>
<tr>
<td>65</td>
<td>I4</td>
<td>A1</td>
<td>Sign of coefficient B</td>
</tr>
<tr>
<td>66-73</td>
<td>KB</td>
<td>I8</td>
<td>Coefficient B of line equation</td>
</tr>
<tr>
<td>74</td>
<td>I5</td>
<td>A1</td>
<td>Sign of coefficient C</td>
</tr>
<tr>
<td>75-81</td>
<td>KC</td>
<td>I8</td>
<td>Coefficient C of line equation</td>
</tr>
<tr>
<td>82-85</td>
<td>I</td>
<td>I4</td>
<td>Sequence number of line segment</td>
</tr>
</tbody>
</table>
POLYGON converts decimal degrees and the coefficients to integer format in order to simplify their handling by COBOL programs. Latitudes and longitudes are multiplied by 10,000 prior to output. Coefficients A and B are multiplied by 10,000,000 and coefficient C is multiplied by 10,000 prior to their being output by the program.

Error Messages

PLY01 THE CURRENT LINES ARE NOT MATCHED nnnnnn

Two line segments have been encountered that are not consecutive. The value of 'nnnnn' indicates the line number of the first line segment of the mismatched pair.

PLY02 LATITUDES OF THE SECOND POINTS ARE OUT OF SEQUENCE

The first line segment of a consecutive pair of line segments should always have the lower latitude when the ending points of the line segments are compared.

PLY04 THERE IS A BREAK IN THE POLYGON nnnnn

This condition is detected in the program's sort routine. It occurs when all the following conditions are present: beginning latitude of line segment 'I' = beginning latitude of line segment 'I+1' AND the ending longitude of line segment 'I' is greater than both the beginning longitude of line segment 'I+2' and the ending longitude of line segment 'I+1'. This indicates there is no connection between line segment 'I' and line segment 'I+1'. The value 'nnnnnn' indicates the sequence number of the equation for line segment 'I'.

PLY05 DUPLICATE LINE IN TABLE i, I, nl, kl, number

The line segment being inserted into the table has the same beginning and ending points as the previous line that was inserted into the line segment table.

PLY06 LAST TABLE INDEX NOT CORRECT nnnnn mmmmm

Indicates that an error has occurred while the line segment table was being expanded to accept a new line segment. The value 'nnnnnn' is the last valid table-index value, 'mmmmm' is the table-index value that caused the error.
Either of these two errors occur when the value of the beginning latitude of line segment 'I+2' is equal to 999.999 indicating that the program has attempted to sort records beyond the end of the line segment table. This error condition may never occur once the program is installed on a given machine. These messages would indicate that the number of line segments to be sorted by the program differed from the actual number of line segments encountered during the sorting operation. The number of line segments to be sorted is greater than the actual number of line segments. They were used during program development and would be useful in testing the program after or during conversion or program modification.

COMPUTER PROGRAMS FOR EDITING AND UPDATING THE GROUND-WATER MASTER FILE

Changes to the master file include addition of data, changes of existing data, and deletion of existing data from the master file. Editing of the changes to the master file is a process of five major steps:

1. Read the changes into a temporary data set and assign initial audit-trail information to each record.
2. Sort the changes into descending order using the site identifier as the primary sort key.
3. Check the site identifier of each change record to insure that all records to be changed, deleted, or added are within the boundaries of the district.
4. Convert each free-field format record into a set of formatted change records.
5. Check the data fields of the change records for valid codes and (or) the reasonableness of data values.

After change data have been fully edited, three steps are involved in the actual updating of the master file:
(1) All records in the master file that belong to a site that has a change in the change file are written to a temporary data file for update processing.

(2) Changes are made to the temporary-change master file.

(3) The changed master is merged with the records from the old master file that were not changed, which creates a new updated copy of the master file.

Master File Edit Program—EDIT-1

This program is the first in a series of five used to edit and update the master ground-water data file. The program reads change records in the WATSTORE free-field format as shown below, initiates an audit trail, checks them for location within the geographic area represented by a polygon and stores the changes for further processing. The general system logic for EDIT-1 is illustrated in figure 5.

Audit-Trail Initiation

Three data fields are added to each of the input records that serve as an audit trail. These three items are (1) a sequence number, (2) the Julian date, and (3) a change-file identifier or volume serial number (VSN).

Testing Records for Inclusion in the Master File

The latitude and longitude of each record are tested against the boundaries of a polygon. If a record's latitude or longitude are outside the polygon, a message is printed indicating that the record has been dropped from the output change file. If the latitude and longitude are within the boundaries of the polygon, a message is printed indicating that the record has been stored in the output change file.
Baseline/polygon description file created by POLYGON.

Change records that fall within the boundaries of the polygon.

Figure 5.--General system logic for program EDIT-1.
EDIT-1 Input Data

Input data consists of three record types:

— Control records
— Free-field format change records
— Polygon-description file from the polygon program

Processing under Control-001, the program (1) reads the input free-field format data, (2) assigns a file identifier or VSN read from the input control records, (3) inserts the current Julian date into each record, (4) assigns an input sequence number to each change record, and (5) prints a listing of all input data with a message indicating whether the record was stored or dropped on output. The program, under Control-001, can also be used to delete selected records from the output file.

Processing under Control-002, the program checks the location of each site to insure that the site is within the boundaries of the polygon represented by the master file. The polygon boundary must be defined by the POLYGON program.

Control-003 allows the user to write selected records to a disk or tape file. Records can be selected by either their input sequence number or by their site-identification numbers. This function is used to aid in making corrections to the change data.

Control records.—A minimum of seven control records are required for the proper operation of EDIT-1. The first control record determines the function performed and must be one of either CNL001, CNL002, or CNL003. The next three control records, 010, 011, and 012, contain information used to title the printed output as described in the section entitled "Control Records for COBOL Programs." Two control records follow the 012 record and are required for all processing with EDIT-1. These records are a volume serial number (VSN) record and a files control record. The VSN record has a six-digit plus one-alphabetic-character identifier that is used to identify the tape or disk volume that has the input change file on it. This VSN will become an integral part of the audit trail established under control-001. The files control record has a six-digit number that passes the number of files to be read from the volume identified by the VSN record. The formats for the VSN and files records are:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VSN Record</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-6</td>
<td>9(5)</td>
<td>Numeric part of the volume serial number</td>
</tr>
<tr>
<td>7-12</td>
<td></td>
<td>Not used</td>
</tr>
</tbody>
</table>
If control-001 is being used to delete records from the input change files, the record numbers are entered on delete records that follow the files record. These delete records have the following format:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-6</td>
<td>X(4)</td>
<td>Not used</td>
</tr>
<tr>
<td>7-12</td>
<td>9(6)</td>
<td>Sequence number of the record to be deleted. Sequence numbers are obtained from the printed listing from each run of EDIT-1.</td>
</tr>
<tr>
<td>13-80</td>
<td>Not used</td>
<td></td>
</tr>
</tbody>
</table>

The original input files should be run through EDIT-1 under control-001 until all records with obvious errors and all blank or 'junk' records are eliminated from the input files. One delete record must be in the control record set for each record to be deleted.

The last record in the control deck has the same format as the delete record and has 9's in positions 7 through 12. This record signals EDIT-1 that all control records have been processed.

Free-field format change records.--The WATSTORE type of free-field input is used in this system in order not to duplicate data entry. Positions 1 through 19 of this record type are of fixed format, and positions 20 through 80 are used to carry the change data in free-field format as described in the WATSTORE User's Guide, Volume 2, Chapter 2. The first four positions of the records are ignored by the program. Positions 5 through 19 MUST contain the 15-digit site-identification number.

The free-field format change records have the following format when read by EDIT-1:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>X(4)</td>
<td>Not used by EDIT-1 ignored</td>
</tr>
<tr>
<td>5-19</td>
<td>9(15)</td>
<td>Site-identification number</td>
</tr>
<tr>
<td>20-80</td>
<td>X(61)</td>
<td>Free-field format data</td>
</tr>
</tbody>
</table>
A complete description of the free-field input-data format can be found in the WASTORE User's Guide, Volume 2, Chapter 2, pages B7-B12 and B98-B102.

**Polygon-input records.**--The polygon-definition program, POLYGON, produces a file that provides EDIT-1 with a mathematical description of the polygon for which data are being processed. The format of the polygon-definition records is:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X(1)</td>
<td>Blank</td>
</tr>
<tr>
<td>2-7</td>
<td>9(6)</td>
<td>Latitude, DMS of beginning point</td>
</tr>
<tr>
<td>8-14</td>
<td>9(7)</td>
<td>Longitude, DMS of beginning point</td>
</tr>
<tr>
<td>15-20</td>
<td>9(6)</td>
<td>Latitude, DMS of ending point</td>
</tr>
<tr>
<td>21-27</td>
<td>9(7)</td>
<td>Longitude, DMS of ending point</td>
</tr>
<tr>
<td>28-33</td>
<td>99V9(4)</td>
<td>Latitude, in decimal degrees of beginning point</td>
</tr>
<tr>
<td>34</td>
<td>X(1)</td>
<td>Sign of latitude</td>
</tr>
<tr>
<td>35-41</td>
<td>999V9(4)</td>
<td>Longitude, in decimal degrees of beginning point</td>
</tr>
<tr>
<td>42-47</td>
<td>99V9(4)</td>
<td>Latitude, in decimal degrees of ending point</td>
</tr>
<tr>
<td>48</td>
<td>X(1)</td>
<td>Sign of latitude value</td>
</tr>
<tr>
<td>49-55</td>
<td>999V9(4)</td>
<td>Longitude, in decimal degrees of ending point</td>
</tr>
<tr>
<td>56</td>
<td>X(1)</td>
<td>Sign of coefficient A</td>
</tr>
<tr>
<td>57-64</td>
<td>9V9(7)</td>
<td>Coefficient A of line equation</td>
</tr>
<tr>
<td>65</td>
<td>X(1)</td>
<td>Sign of coefficient B</td>
</tr>
<tr>
<td>66-73</td>
<td>9V9(7)</td>
<td>Coefficient B of line equation</td>
</tr>
<tr>
<td>74</td>
<td>X(1)</td>
<td>Sign of coefficient C</td>
</tr>
<tr>
<td>75-81</td>
<td>9(4)V9(4)</td>
<td>Coefficient C of line equation</td>
</tr>
<tr>
<td>82-85</td>
<td>9(4)</td>
<td>Sequence number of line segment</td>
</tr>
</tbody>
</table>

**Program Output**

Output from the program consists of three files—the change file, a pull file, and the line-printer listing. All three files contain essentially the image of the input records with only a few additions.

**Output change file.**--The output change file contains images of the input data record, in addition to the current Julian date, an input sequence number, and a file-identification or volume serial number. This is the file that is passed on to the next processing step or program as the input-data file.
Output pull file.--This file contains exact images of the input data records. By removing erroneous and (or) blank records from this file, errors can be easily corrected and blank records deleted. After all corrections have been made, this file is re-edited to verify the corrections that were made. If the file contains only blank records and tape or file labels, the entire pull file can be discarded, only change records need be kept for correction.

Output line-printer file.--This file contains the paged listing of the input change file (55 lines per page). These records are images of the input change records with the addition of the audit-trail fields and a message that indicates whether the record was stored in the output file or deleted on input. An example of the line-printer listing from EDIT-1 program is shown below.

Component-Format Edit Program—EDIT-2

This program reads the free-field change records, each of which contains changes to a WATSTORE repeating group, and reformats them into records containing only a single component change. The program groups these single component change records into change sets. The transition from free-field format to a one-component-per-change-record format is shown on the following page.
Free-field format change record:
R=234$T=A$235#1200801976$237=61.0$239=V$

Change set created by EDIT-2 from above record:

8083C822580001325908112464701D234235AA00101200801976
8083C822580002325908112464701D234237AA620461.0
8083C822580003325908112464701D234239AA0001V

Each component in the input record is examined for adherence to the rules for editing and updating of the WATSTORE GWSI file. The components that do not meet the rules as described in the WATSTORE User's Guide are written in an error file and processing continues until an end-of-file marker is reached in the input data file. The general system logic for EDIT-2 is shown in figure 6. Components are written, one per record, to allow easy access to any individual component in the change file and to simplify editing and updating of each component.

Program Input

Input to EDIT-2 consists of a deck of control records or 80-position record images and the output file from EDIT-1, Control-002.

Control records.--A minimum of four control records are required for the operation of EDIT-2. The first control record must have CNL001 or CNL002 in positions 1 through 6. This first record determines the function performed by the program. The next three records, CNL010, CNL011, and CNL012, contain information used in titling the output and to aid in tracking the listings. Under Control-003, the program requires at least one additional record with a record locator on it. These locators may be individual sequence numbers or site identifiers, or they may be ranges of sequence numbers or site identifiers. The last control record should be an end-of-processing indicator with 9's in positions 1 through 6.

Input data records.--Input records to EDIT-2 are contained in the change file output by the EDIT-1 program. Input data read by EDIT-2 has the following format:
Figure 6.---General system logic for program EDIT-2.
<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-6</td>
<td>X(6)</td>
<td>Volume serial number (VSN)</td>
</tr>
<tr>
<td>7-15</td>
<td>9(9)</td>
<td>Input sequence number</td>
</tr>
<tr>
<td>16-30</td>
<td>X(15)</td>
<td>Site-identification number</td>
</tr>
<tr>
<td>31-91</td>
<td>X(61)</td>
<td>Free-field change data</td>
</tr>
<tr>
<td>92-96</td>
<td>9(5)</td>
<td>Julian date</td>
</tr>
</tbody>
</table>

EDIT-2 reads the input change file of free-field format change records. Each input record is broken down into one or more records; each of these records contain the transaction type and data-component value for each WATSTORE type value pair in the input record. The program continues to add records to a change set being created from the input data until one or more of the following conditions are met: (1) a new site-identification number is read, (2) a new record-type identifier is read, or (3) a new transaction code is read. This process of converting free-field change records into change-record sets continues until an end-of-file marker is read by the program.

**Program Output**

Output records generated by EDIT-2 contain the following information:

- Original volume serial number
- Current Julian date
- Record's sequence number within its change-record set
- Record type that the component belongs to
- Component identifier
- Length of the change data in characters
- Component change value
- Output sequence number
- Transaction-type code
- Type-of-data code

**Volume serial number.**—This is the 1- to 7-character file identifier or volume serial number of the original input change file or tape.

**Julian date.**—The date on which the program was run is in Julian calendar format (yyddd).

**Site-identification number.**—This field is the 15-digit site-identification number from the input record.
Change-record set sequence number.--Each change-record set is composed of one or more change records. The record set has one change record for each component in the original input data record.

Record-type identifier.--The record-type identifier is three digits and is used in the WATSTORE GWSI file or assigned to a new record type by the user.

Component identifier.--The component identifier is a three-digit WATSTORE code that identifies a specific component field within a record type.

Data length.--This field is a count of the number of characters contained in the component-value field. It is used only for alpha and alphanumeric data.

Data-type code.--This two-digit code describes the type of data contained in the component-value field. All alpha and alphanumeric data have a data-type code of 99. Codes for numeric data are of the form nd, where n is the total number of digits which can be in the numeric item, and d is the number of digits which occur to the right of the decimal point. For example, a water-level measurement can have a length of six digits of which two digits are to the right of the decimal point. The data-type code for this water-level component is 62.

Component value.--This field is a 45-character table or array. It is processed as alphanumeric and contains as many characters as specified in the data length field for alpha and alphanumeric data. It contains as many digits as the n portion of the data-type field for numeric data. In the case of a component having no value, or having a missing value, the component value field will contain the word NULL.

Output Record Format

The output data file from EDIT-2 has the following format:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>X(5)</td>
<td>Original file identification or VSN</td>
</tr>
<tr>
<td>6-10</td>
<td>9(5)</td>
<td>Julian date</td>
</tr>
<tr>
<td>11-13</td>
<td>999</td>
<td>Sequence number of record within a change set</td>
</tr>
<tr>
<td>14-28</td>
<td>9(15)</td>
<td>Site-identification number</td>
</tr>
</tbody>
</table>
37

29-31 999 Record-type identification number
32-34 999 Component identification number
35 X Transaction code
36-37 99 Type of data
38-39 99 Length of data in characters
40-84 X(45) Change data
85-90 9(5) Output sequence number

Error Messages—Control-001

SKIPPING TO NEXT RECORD - FIRST CHARACTER NOT *R*

The first non-blank character in the input data field of the first change record for a new site or record type was not an 'R'. A new change record for a site must begin with 'R='.

RECORD FORMAT IN ERROR

The first non-blank character encountered after an 'R=' is not an equal sign (=).

RECORD NUMBER NOT NUMERIC

One or more of the first three non-blank characters following an 'R=' is not a numeric character.

T NOT FOUND

The letter 'T' was not the first non-blank character encountered following an 'R=' value pair.

TRANSACTION FORMAT ERROR

The first non-blank character encountered following a 'T=' is not an equal sign (=).

NO VALID TRANSACTION CODE

A valid transaction code (A, D, or M) was not the first non-blank character following a 'T=' value pair.

INCORRECT END OF TRANSACTION CODE

A character that is not a valid value-pair delimiter ($ or *) was encountered immediately following the transaction code.
1ST CHARACTER OF *C* NOT NUMERIC

The first character of a data-value pair was not a numeric character (0-9, =, -).

SKIPPING REST OF RECORD - TOO MANY *C* POSITIONS

A component identifier in a value pair had more than three digits. Check for a missing equal sign or pound sign (= or #).

NON-NUMERIC ENCOUNTERED - LOOKING FOR REST OF *C*

The second or third character of the component identifier of a data-value pair is not a numeric value (0-9, +, -)

COMPONENT DATA LONGER THAN 45 CHARACTERS

A component value was encountered that was longer than 45 characters. This length is an arbitrary value and may be changed to suit your needs. CAUTION: If the length is changed in EDIT-2, it must be changed in all programs that follow EDIT-2.

LOCAL WELL NUMBER DATA FIELD STARTS WITH OTHER THAN '1

The local well identification number (C=12) must be enclosed in single quotes.

END OF DATA CHARACTER MISSING

A data value-pair delimiter ($ or *) was not found before the next record identifier was encountered.

Error Messages—Control-002

760M2 NO DATA INPUT

An end-of-file marker was encountered on the first attempt to read from the input data file.

761M2 END OF RUN

An end-of-file marker was encountered while attempting to read a control record image. This is normal end of processing.
763M2 MULTIPLE RANGE PARAMETERS

All values on the locator control record are zero or a blank record image has been read.

764M2 SEQUENCE COUNT WITH OTHER VALUES

An ending sequence number has been read from a range control record and a value was encountered in the 'NUMSTR' field of the same range control record.

765M2 SITE ID RANGE WITH OTHER VALUES

Both a range of site identifiers AND a range of record numbers have been specified for retrieval by the control record read by EDIT-2. Select either a range of record sequence numbers or a range of site identifiers, BUT NOT BOTH.

766M2 SEQUENCE COUNT INVALID END

A range of change-record sequence numbers has been read by the program for retrieval. The beginning value of the range to be retrieved is larger than the ending sequence number. Change the range numbers on the control record so that the beginning sequence number is less than the ending sequence number.

767M2 SITE ID RANGE INVALID - END LOW

The ending site identifier of the specified range of change records to be retrieved is larger than the beginning site identifier specified on the control record. Since the change file is sorted in descending order on the site-identifier field, the beginning site identifier on the retrieval record must be larger than the ending site identifier.

768M2 NO RECORDS PRINTED NUMFST = nnnnn
               CNRECN = mmmmm

The beginning sequence number read from the retrieval control record is greater than the number of records contained in the change record file. Correct the range of records to be retrieved on the control record or check to see that you have the correct change file attached to the program.
The ending record sequence number of the range of records to be retrieved from the change file is larger than the number of records in the change file. Check to see that you have the correct change file attached to the program or change the ending value in your retrieval request.

**Component-Value Edit Program—EDIT-3**

Program EDIT-3 performs editing of the component-value field to insure that reasonable and (or) valid data values are being entered into the master data file. The general system flow chart describing EDIT-3 is included as figure 7.

EDIT-3 reads a complete set of change records for a particular site and record type. This change set is stored in a table so that all the components contained in the change set can be examined. If the change set represents a new record type to be added to a given site or the addition of a new site, each of the component identifiers is examined to insure that all mandatory components are present in the change set. Mandatory components are checked in the paragraphs "STA000" through "STA320" in the Procedure Division of EDIT-3. A component's status as mandatory or optional may be changed by adding or deleting the checks in the appropriate "STAnnn" paragraph, where the "nnn" corresponds to the record-type identifier. Each component value is edited to insure that only reasonable values or valid codes are entered into the master file. If the change set represents a change to data values already in the file, each component is edited for reasonable data values and valid codes.

EDIT-3 conforms to all edits as described in the WATSTORE User's Guide, Volume II, Chapter 2. If no errors are encountered in the editing of a change set, the entire change set is written to the output-update file; otherwise, the entire change set is written to an error file for correction and re-editing.

**Program Input**

Input to the program consists of the output changes from the component-field format-edit program, EDIT-2. EDIT-3 has only one function, and therefore no control records are required as input. Input-data records to EDIT-3 have the following format:
Figure 7.--General system logic for program EDIT-3.
Program Output

Output from EDIT-3 is unchanged from the input records in terms of record format. Output from EDIT-3 consists of an output-update change file, an error file, and a printed listing of all input change sets that contain errors.

Update change file.--The output-update change file contains change sets in which every member record has passed the value edits for its particular component value.

Output error file.--If an error is encountered in the component value of any member of a change set, the entire change set is written to the error file. Editing of component values continues through the entire change set even though an error may have been detected in a previous member record of the change set.

Error Messages—Edit Errors

INVALID RECORD NUMBER - RNUM = nnn

A record number has been encountered that is neither a valid WATSTORE identifier nor a valid user-defined record identifier. Correct the record number or insert a new record description in the program with tests for the new record identification in paragraphs 'EXAM-CHANGE' and 'DIST-CHECK'.
REQUIRED DATA FIELD MISSING

A mandatory component field is missing from the change record set.

CHANGE > 45 CHARACTERS

The DLEN field of a change record has a value greater than 45. This message is a warning only. The update programs will only insert the first 45 characters into the updated record component. If characters are present beyond the 45-character length limit or the length defined for the component in the update programs, they will be truncated.

DISTRICT CODE IS NOT '04' - AZ

The value for the district-code component (C=6) is not '04'.

INVALID ALPHA CODE IN CHANGE

An alphanumeric character was encountered in a numeric data item.

STATE CODE NOT '04' - AZ

The value for the district-code component (C=6) is not '04'.

COUNTY CODE IS NOT VALID CODE

INVALID QUAD IN LOCAL ID

The first character of the local identification number is not 'A', 'B', 'C', or 'D'.

INVALID LOCAL IDENTIFIER

ALTITUDE OUTSIDE ACCEPTABLE RANGE - 0:12633

DATE IS NOT NUMERIC VALUE
DEPT OF HOLE IS NOT VALID

A depth-of-hole value was encountered that is negative or greater than 3,000 feet or less than depth of well.

DEPT OF WELL IS NOT VALID

A depth-of-well value was encountered that is negative or greater than 3,000 feet or greater than depth of hole.

ENTRY NUMBER IS NOT NUMERIC

An entry number used to identify multiple entries of the same record type for a given site has a non-numeric character.

INVALID LITHOLOGIC CODE

An invalid lithologic code has been encountered that is not identified in the data description—'GEO-CODES'.

SOURCE AGENCY NOT VALID

An invalid source-agency code has been encountered.

DATE IS LATER THAN TODAY'S DATE

A date has been found that places the value at some time in the future.

MANDATORY FIELD IS BLANK

A mandatory component field is null or has all blank characters.

QW CODE NOT VALID STORET CODE

A field water-quality code has been found that is not defined in the QW-CODES data description in EDIT-3.

Error Messages—Programmed Stops

STOP5 FILEA-UPCHNG EMPTY
*** CHECK FOR ATTACH OF CHANGE FILE IN JCL***

The change file was in an end-of-file condition on the first attempt to read it.
ERROR IN RECORD SEQUENCE STOP8 AZERRS
KTEST = KEYIDDD =

The input change file has at least one record out of sequence. The change file should have the same sort order (descending site identification) as the master data file.

Master File Update Preparation Program—UPDATE-1

This program separates the master ground-water data file into two files. One file contains the records for sites not having changes in the update change file; the other contains all records for each site having changes in the update change file (fig. 8).

UPDATE-1 reads the update change file and the master file. If the site-identification number of the change record matches the site-identification number of the master file record, all the records with the same site identifier are written from the master file to a change volume of the master file. The change file is then read until a new site identifier is encountered; the master file is read until the new site identifier is either located or not found. A site is considered 'not found' when the site identifier read from the master file is less than the site identifier of the current change record. If no change records are in the change file for a given site, the master file records for that site are written to an unchanged volume of the master file. These two volumes of the master file will be merged after the updating of the master file change volume is completed.

Program Input

Input to UPDATE-1 consists of the current volume of the master ground-water data file and the update change file that was output by the component edit program, EDIT-3.

Program Output

Two master file volumes are created by UPDATE-1. These two files contain variable-length records of the same format and description as discussed in the section entitled "Master Data File." Appendix A contains the COBOL record descriptions of the WATSTORE record types contained in the master file. By splitting the master file into two files, only those sites for which change records are present in the update change file will be accessed by the final update program. This reduces access time and processing costs and helps to maintain the integrity of the data for those sites not being changed.
Figure 8.--General system logic for program UPDATE-1.
This program makes the changes indicated in the update change file to the records contained in the change volume of the master groundwater data file. Figure 9 shows the general system flow chart for UPDATE-2.

UPDATE-2 reads the update data file and the change master file from UPDATE-1. Each record in the change master file for which there is a change-record set in the update data file is updated and then written to an updated master volume.

Valid Update Transactions

The program recognizes three transaction codes as valid: A)dd, M)odify, and D)elete.

A)dd transaction.--A new site will be added only when an R=000 record for that site does not exist in the master change file and the first record for that site in the update change file is an R=000 record.

A new record will be added for a site only when an R=000 record exists in the master change file and no record with the same record-type identifier exists except where multiple records of the same type are allowed.

A component field will be added only when an R=000 record and the record owning the component to be added already exist in the master change file. The component value in the owning record must be a null value in order to add the component.

M)odify transaction.--Records and (or) components will be modified only where all prerequisites for modification exist. The prerequisites for modifying the contents of a given record or component are described in detail in the WATSTORE User's Guide, Volume II, Chapter 2.

D)elete transaction.--A site will be deleted only if an R=000 record exists in the master change file. If the delete is valid, all records for that site will be deleted from the master file.
Master file records to be updated (from UPDATE-1).

From UPDATE-1.

Replaces last copy of master file. Last copy becomes backup file.

Figure 9.--General system logic for program UPDATE-2.
A record will be deleted only where the site has an R=000 record in the master change file and the record to be deleted is present in the master change file. If other records have any dependency upon the record being deleted, those dependent records also will be deleted.

A component can be deleted only when an R=000 record exists for the site, the record that owns the component exists, and the component to be deleted has a value other than null. Deleting a component on which other records or components are dependent will cause those records or components to be deleted also.

Null values.--A component change record which contains the word NULL in the data field, causes the program to assign the COBOL literal 'Low-Values' (000 octal) to the component field in the master file record.

Program Input

Input to UPDATE-2 consists of the update change file from EDIT-3 and the master change file output by the master file preparation program, UPDATE-1.

Program Output

The program's output consists of an updated master change file and an error file. A listing is produced of the errors with appropriate error messages.

Updated master ground-water data file.--This file contains all the records from the input change master file and the valid changes made to them except for the sites, records, and component fields that have been deleted.

Error output file.--The error file contains all change-record sets and (or) individual change records that contain invalid transactions or transactions that cannot be completed because modification prerequisites were not present. These errors are noted on the output listing and may be rerun with the program after corrections have been made to them.
PROGRAMS FOR DATA RETRIEVAL AND REPORTING

Ground-Water Site Inventory (GWSI) Data Retrieval Program—BASIN-3

BASIN-3 is a COBOL program designed to provide an inventory of selected data for all ground-water sites in a given geographic area. Data are retrieved and stored for each site located within the boundaries of a given polygon. BASIN-3 reads consecutive slices of the input polygon and the input data file. If the location of the site read falls within the boundaries of the polygon, data for that site are written to the output files. If the site location falls below the current slice, the program reads the next slice of the polygon. The program stops when all input data have been read, or a record is read whose site location falls below the lowest point of the polygon. The general system logic is shown in figure 10.

Program Input

Input to the program consists of two control records, a polygon-description file, and a ground-water data file.

Control records.--The program is initially controlled by two user-supplied control records. The control records required for the operation of BASIN-3 are control records 001 and 010.

Polygon-description file.--The polygon-description file is read from a disk file by the program. This disk file is created by the polygon-definition program, POLYGON, and contains the polygon-definition table.

Ground-water data file.--Data records input to BASIN-3 may be either the variable-length records of the master file or the 270-character records of the working ground-water data file. The length of the variable portion of the record is defined by LNUM (positions 19-21) of each record. The format of the variable portion is defined by RNUM and a detailed description of each WATSTORE record type can be found in the WATSTORE User's Guide, Volume 2, Chapter II, pages B14 through B97.1.

Program Output

BASIN-3 produces two output files: (1) a printed report of available site data, which is shown in the following example, and (2) a disk file of the available site data for use with other programs.
Figure 10.--General system logic for program BASIN-3.
Site-inventory report.--The site-data listing contains all the available data for the parameters for each site located within the boundaries of the polygon read by the program. The last page of the well-site report (table 4) lists all the valid codes and their meanings for the following parameters: (1) source-of-depth data, (2) methods of measurement, (3) log types, (4) use of water, (5) type of finish, and (6) type of opening.

Site-data output file.--The data file written to disk storage for use by other programs is a compact version of the printed report. The records in this file are 105 characters in length and have the following format:

<table>
<thead>
<tr>
<th>Position</th>
<th>Variable</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>CNREC</td>
<td>9(4)</td>
<td>Sequence number</td>
</tr>
<tr>
<td>5-19</td>
<td>KEYID</td>
<td>X(15)</td>
<td>Site identifier</td>
</tr>
<tr>
<td>20-39</td>
<td>LCWLID</td>
<td>X(20)</td>
<td>Local site identifier</td>
</tr>
<tr>
<td>40-44</td>
<td>DEDEPTH</td>
<td>9(5)</td>
<td>Depth of well</td>
</tr>
<tr>
<td>45-50</td>
<td>ALTDEP</td>
<td>X</td>
<td>Source of depth data</td>
</tr>
<tr>
<td>46-50</td>
<td>ALTITUDE</td>
<td>9(5)</td>
<td>Altitude of site above or</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>below sea level</td>
</tr>
</tbody>
</table>
51- 55  PPDISH    9(5)  Rate of discharge
56- 59  PPYEAR    9(5)  Year of discharge
60- 63  FCYEAR    X(4)  Year construction was completed
64     WTRUSE    X    Primary use of water
65- 72  INLOGS    X(8)  Available log types
73- 79  OPNTOP    9(5)V99 Depth to top of the open section
80     OPNTYP    X    Type of opening
81- 85  CDIAM     999V99 Diameter of casing
86     FINISH    X    Type of finish
87- 92  WTRLVL    -9(4)V99 Most recent water-level measurement
93- 96  WLYEAR    X(4)  Year of most recent water-level measurement
97- 99  WLCNT     999 Number of water-level measurement records available
100-105 SPCCAP    9(4)V99 Specific capacity

Error Messages

BSN3 CONTROL INPUT FILE EMPTY

Indicates that no data are present in the input stream on the first read attempt.

BSN3 ID MISSING OR INCORRECT IN CONTROL RECORD

A control record has been read that has an identification that was not recognized by the program. Check the control records for one of the following conditions:

—Control records not in required sequence
—Control record(s) not present
—'CNL' is not in positions 1-3

STOP4 SOURCE ID RECORDS MISSING

An end-of-file marker was encountered while trying to read the second control record. Check to insure that both control records are present in the input job stream.
Table 4. Parameter codes and their meaning as printed by the BASIN-3 program

<table>
<thead>
<tr>
<th>Depth Source</th>
<th>Water Levels Measure Method (MLM)</th>
<th>Available Log Types</th>
<th>Use of Water</th>
<th>Type Openings</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td>S RPTING AGENCY</td>
<td>A AIRLINE</td>
<td>A TIME</td>
<td>A AIR CONDITIONING</td>
<td>F FRACTURE</td>
<td>C POROUS, CONCRETE</td>
</tr>
<tr>
<td>O DRILLER</td>
<td>C CAL AIRLINE</td>
<td>B COLLAR</td>
<td>B BOTTLING</td>
<td>L LOUVERED</td>
<td>F GRAVEL W/PERF</td>
</tr>
<tr>
<td>O OWNER</td>
<td>E ESTIMATED</td>
<td>C CALIPER</td>
<td>C COMMERCIAL</td>
<td>M MESH</td>
<td>G GRAVEL, SCREEN</td>
</tr>
<tr>
<td>A OTHER GOV'T</td>
<td>G PRESSURE GAGE</td>
<td>D DRILLER'S</td>
<td>D DEWATER</td>
<td>P PERFORATED</td>
<td>H HORIZ GALLERY</td>
</tr>
<tr>
<td>R OTHER REPORTED</td>
<td>H CAL PRESSURE GAGE</td>
<td>E ELECTRIC</td>
<td>E POWER</td>
<td>R WIRE WOUND</td>
<td>G OPEN HOLE</td>
</tr>
<tr>
<td>L LOGS</td>
<td>L GEOPHYSICAL LOGS</td>
<td>F FLUID CONDUCT</td>
<td>F FIRE</td>
<td>S SCREN(UNK)</td>
<td>P PERFORATED</td>
</tr>
<tr>
<td>G GEOLOGIST</td>
<td>M MANOMETER</td>
<td>G GEOLOGIST</td>
<td>H DOMESTIC</td>
<td>T SAND, POINT</td>
<td>S SCREEN</td>
</tr>
<tr>
<td>Z OTHER</td>
<td>R REPORTED</td>
<td>H MAGNETIC</td>
<td>I IRRIGATION</td>
<td>W WALLED</td>
<td>T SAND, POINT</td>
</tr>
<tr>
<td>S STEEL TAPE</td>
<td>T ELECTRIC TAPE</td>
<td>J GAMMA RAY</td>
<td>K MINING</td>
<td>X OPEN HOLE</td>
<td>W WALLED</td>
</tr>
<tr>
<td>Y CAL ELECTRIC TAPE</td>
<td>Z OTHER</td>
<td>K DIPMETER</td>
<td>M MEDICINAL</td>
<td>Z OTHER</td>
<td>X OPEN HOLE</td>
</tr>
<tr>
<td>Z OTHER</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Z OTHER</td>
</tr>
</tbody>
</table>
STOP5  DATA INPUT FILE END INCORRECT
EOF ENCOUNTERED AT READB OR READV

An end-of-file marker was encountered while reading the polygon-description file before an end-of-polygon indicator was read from the file. Check the last record in the polygon-definition table to see that it contains 9's in positions 1-15.

STOP6  SECOND RECORD OF PAIR MISSING

Each slice in the polygon-definition table is defined by at least two line-segment records. A slice may be defined by more than two records, but the records are always some multiple of two. If this error occurs, re-run the polygon-definition program, PLGN01.

STOP7  FIRST RECORD NOT '000' OR '001'

The first record encountered for a site was neither a GWSI-type header record nor a water-quality type header record.

STOP8  FILE OUT OF SEQUENCE

The data file being read has a record out of sequence. The data file must be sorted in descending order by the site identifier.

STOP9  RECORD TYPE NOT FOUND

A record was read from the data file that was neither a WATSTORE record type nor a user-defined record type. Insert a record description of the record into the program and add a test for the record type number in {paragraphs}.

Water-Level History Program—BASIN-4

BASIN-4 reads a ground-water data file or water-level working file and a polygon-description file to produce a water-level history for all sites that fall within the given polygon. The general system logic of BASIN-4 is shown in figure 11.

BASIN-4 reads consecutive slices of the input polygon and the input water-level file. If the location of the site read falls within the
Figure 11.--General system logic for program BASIN-4.
boundaries of the polygon, all water levels for that site are written to the output files. If the site location falls below the current slice, the program reads the next slice of the polygon. The program stops when all water-level data has been read or a water-level record is read for which the site location falls below the lowest point of the polygon.

Program Input

The program requires the following input data: (1) two control records, (2) a polygon-description file, and (3) a water-level working file or ground-water data file.

Control records.--The program is initially controlled by two user-supplied control records. The control records required for the operation of BASIN-3 are control records 001 and 010.

Polygon-description file.--The polygon-description file is read from a disk file by the program. This disk file is created by the polygon-definition program, POLYGON, and contains the polygon description as described in the section covering POLYGON in this report.

Water-level data file.--Water levels may be read from either the ground-water master file or from a water-level working file. In either case, the water-level records have the following format:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>WATSTORE code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1- 15</td>
<td>9(15)</td>
<td>C1</td>
<td>15-digit site-identification number</td>
</tr>
<tr>
<td>16- 18</td>
<td>999</td>
<td></td>
<td>Record-type identification number</td>
</tr>
<tr>
<td>19- 21</td>
<td>999</td>
<td></td>
<td>Length of variable portion</td>
</tr>
<tr>
<td>22</td>
<td>X</td>
<td></td>
<td>Data security or access code</td>
</tr>
<tr>
<td>23- 30</td>
<td>X(8)</td>
<td>C235</td>
<td>Date on which the water level was measured</td>
</tr>
<tr>
<td>31</td>
<td>X</td>
<td>C236</td>
<td>Accuracy of the date</td>
</tr>
<tr>
<td>32- 37</td>
<td>9(4)V99</td>
<td>C237</td>
<td>Water-level measurement</td>
</tr>
<tr>
<td>38</td>
<td>X</td>
<td>C238</td>
<td>Site status at time of measurement</td>
</tr>
<tr>
<td>39</td>
<td>X</td>
<td>C239</td>
<td>Method of measurement</td>
</tr>
<tr>
<td>40</td>
<td>X</td>
<td>C240</td>
<td>Accuracy of the measurement</td>
</tr>
</tbody>
</table>
Program Output

Output from the program consists of a printed water-level history for each site within the polygon and a compressed water-level data file that can be used as input to other programs, such as plot programs.

A water-level history for a given site includes the following data:

- Output sequence number of the selected site within this particular retrieval
- Site-identification number
- Local site-identification number or site name
- Site type, such as well, spring
- Altitude of the site
- Depth of the well
- All water-level measurements available for the site
- Date of each water-level measurement

The output water-level data file contains the same information as the printed listing, plus a two-digit record identifier that is used by a hydrograph program included as part of a ground-water modeling package used by the Swab/RASA study. Format of the output water-level data record is:

<table>
<thead>
<tr>
<th>Position</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>99</td>
<td>Record identifier for hydrograph program, must be '88'</td>
</tr>
<tr>
<td>3-6</td>
<td>9(4)</td>
<td>Sequence number of the site</td>
</tr>
<tr>
<td>7-10</td>
<td>X(4)</td>
<td>Blank</td>
</tr>
<tr>
<td>11-25</td>
<td>9(15)</td>
<td>Site-identification number</td>
</tr>
<tr>
<td>26-45</td>
<td>X(20)</td>
<td>Local site identifier</td>
</tr>
<tr>
<td>46-50</td>
<td>9(5)</td>
<td>Altitude of the site</td>
</tr>
<tr>
<td>51-55</td>
<td>9(5)</td>
<td>Depth of well below land surface</td>
</tr>
<tr>
<td>56-62</td>
<td>9(4)V99</td>
<td>Water-level measurement</td>
</tr>
<tr>
<td>63-70</td>
<td>9(8)</td>
<td>Date of measurement</td>
</tr>
</tbody>
</table>

The water-level history report printed by the BASIN-4 program is of the following type:
<table>
<thead>
<tr>
<th>REC NUM</th>
<th>SITE IDENTIFIER</th>
<th>SITE ALTITUDE</th>
<th>WELL DEPTH</th>
<th>WATER LEVEL HISTORY</th>
<th>LEVEL</th>
<th>DATE</th>
<th>DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>324945 1111945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>324945 1111937</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>324945 1111937</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>324945 1111937</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>324945 1111937</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>324945 1111937</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td>324945</td>
<td></td>
</tr>
</tbody>
</table>

Error Messages

**BSN4 CONTROL INPUT FILE EMPTY**

Indicates that no data are present in the input stream on the first read attempt.

**BSN4 ID MISSING OR INCORRECT IN CONTROL RECORD**

A control record has been read that has an identification that was not recognized by the program. Check the control records for one of the following conditions:

—Control records not in required sequence
—Control record(s) not present
—'CNL' is not in positions 1-3

**STOP4 SOURCE ID RECORDS MISSING**

An end-of-file marker was encountered while trying to read the second control record. Check to insure that both control records are present in the input job stream.
An end-of-file marker was encountered while reading the polygon-description file before an end-of-polygon indicator was read from the file. Check the last record in the polygon description to see that it contains 9's in positions 1-15.

Each slice in the polygon description is defined by at least two line-segment records. A slice may be defined by more than two records, but the records are always some multiple of two. If this error occurs, re-run the polygon-definition program, PLGN01.

The first record encountered for a site was neither a GWS1-type header record nor a water-quality type header record.

The data file being read has a record out of sequence. The data file must be sorted in descending order by the site identifier.

A record was read from the data file that was neither a WATSTORE record type nor a user-defined record type.

BSNPLT is a FORTRAN program designed to plot the areal location of selected ground-water sites. Thirteen site-selection parameters are available in BSNPLT. These parameters allow the user to select only those sites having the specific individual characteristics of interest for study. Virtually any combination of available parameters may
be used. For example, a plot may be needed of only those wells having all the following characteristics:

- Constructed after 1951
- Depth of well greater than 1,000 feet
- Water levels greater than 400 feet
- Depth to top of the opening greater than 300 feet
- Sites with drillers' logs and (or) electric logs
- Primary water use is municipal
- Number of water-level records greater than 5

The program would select all sites within the polygon that meet all these specifications.

Selection parameters are grouped into three classes:

(1) Time-span parameters:
   - Year of construction
   - Year of water-level measurement

(2) Numeric parameters:
   - Depth of well
   - Water-level measurement
   - Depth to top of open section
   - Number of water-level measurements
   - Altitude of water level
   - Specific capacity
   - Casing diameter

(3) Alpha-coded parameters:
   - Log types available
   - Type of water use
   - Type of finish

Time-Span Selection Parameters

Selection of a site by a time-span parameter is governed by the following rules:

A. If the value in the beginning-year field is zero or blank and the ending-year field is greater than zero or blank, a site is selected only if its value for the time-span parameter is less than or equal to the value in the ending-year field: For example, if the beginning-year field is zero or blank and the ending-year field is the current calendar year, the entire period of record is selected.
B. If the beginning-year field is greater than zero or blank and the ending-year field is greater than zero or blank, a site is selected only if its value for the time-span parameter is greater than or equal to the beginning-year field and less than or equal to the ending-year field.

C. If the beginning-year field is zero or blank and the ending-year field is zero or blank, the parameter is not used in site selection.

D. If the beginning-year field is greater than zero or blank and the ending-year field is zero or blank, a site is selected only if its value for the parameter is greater than or equal to the beginning-year field.

For example,

<table>
<thead>
<tr>
<th>Begin</th>
<th>End</th>
<th>Site data</th>
<th>Program</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1960</td>
<td>1961</td>
<td>REJECT</td>
</tr>
<tr>
<td>1960</td>
<td></td>
<td></td>
<td>ACCEPT</td>
</tr>
<tr>
<td>1930</td>
<td></td>
<td></td>
<td>ACCEPT</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td></td>
<td>ACCEPT</td>
</tr>
<tr>
<td>1960</td>
<td>0</td>
<td>1961</td>
<td>ACCEPT</td>
</tr>
<tr>
<td>1960</td>
<td></td>
<td>1960</td>
<td>ACCEPT</td>
</tr>
<tr>
<td>1960</td>
<td>1930</td>
<td></td>
<td>REJECT</td>
</tr>
<tr>
<td>1960</td>
<td></td>
<td>0</td>
<td>REJECT</td>
</tr>
<tr>
<td>1950</td>
<td>1960</td>
<td>1961</td>
<td>REJECT</td>
</tr>
<tr>
<td>1950</td>
<td>1960</td>
<td>1960</td>
<td>ACCEPT</td>
</tr>
<tr>
<td>1950</td>
<td>1960</td>
<td>1950</td>
<td>ACCEPT</td>
</tr>
<tr>
<td>1950</td>
<td>1960</td>
<td>1952</td>
<td>ACCEPT</td>
</tr>
<tr>
<td>1950</td>
<td>1960</td>
<td>1930</td>
<td>REJECT</td>
</tr>
</tbody>
</table>

Numeric Selection Parameters

Selection of a site based on the numeric parameters is governed by the following rules:

A. If the input parameter is positive (+), a site is selected only if its value for the parameter is greater than or equal to the selection parameter.

B. If the input parameter is negative (-), a site is selected only if its value for the parameter is less than or equal to the selection parameter.

C. If the input parameter = 0, the parameter is not used in the selection process.
Example:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Site data</th>
<th>Program</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td>400</td>
<td>ACCEPT</td>
</tr>
<tr>
<td>400</td>
<td>500</td>
<td>ACCEPT</td>
</tr>
<tr>
<td>400</td>
<td>395</td>
<td>REJECT</td>
</tr>
<tr>
<td>-400</td>
<td>400</td>
<td>ACCEPT</td>
</tr>
<tr>
<td>-400</td>
<td>500</td>
<td>REJECT</td>
</tr>
<tr>
<td>-400</td>
<td>395</td>
<td>ACCEPT</td>
</tr>
</tbody>
</table>

Alpha-Coded Selection Parameters

Selection of a site based on the alpha-coded parameters is governed by the following rules:

A. If the selection parameter is left blank, the parameter is not used in the selection process.

B. If one or more codes are entered in the selection parameter, a site will be selected only if it has one or more matching codes for that parameter.

NOTE: All alpha-coded parameters used for selection are single-character codes.

For example,

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Site data</th>
<th>Program</th>
</tr>
</thead>
<tbody>
<tr>
<td>D G</td>
<td>AUG</td>
<td>ACCEPT</td>
</tr>
<tr>
<td>D G</td>
<td>SAD</td>
<td>ACCEPT</td>
</tr>
<tr>
<td>D G</td>
<td></td>
<td>REJECT</td>
</tr>
<tr>
<td>D G</td>
<td>UA</td>
<td>REJECT</td>
</tr>
</tbody>
</table>

Site-Labeling Options

The manner in which a site is labeled on the plot is dependent on the value of the variables NID, NID1, and NID2 on control record 4.
If NID is

Zero: No label is plotted with the site symbol.

Blank: No label is plotted with the site symbol.

1: A single user-selected value will be plotted on the right side of the symbol. The value used for the label is determined by the value of NID1.

2: Two user-selected values will be plotted on the right side of the marker, one above the other. The values used are dependent upon the values of NID1 and NID2. The value for NID1 is plotted over the value for NID2.

For example,

NID = 1 and NID1 = 12     NID = 2; NID1 = 1, NID2 = 12

  • 1520                     • 413
  • 1520

The codes for NID1 and NID2 are

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sequence number on output listing</td>
</tr>
<tr>
<td>2</td>
<td>Depth of well</td>
</tr>
<tr>
<td>3</td>
<td>Altitude of site</td>
</tr>
<tr>
<td>4</td>
<td>Year of construction</td>
</tr>
<tr>
<td>5</td>
<td>Water-level measurement</td>
</tr>
<tr>
<td>6</td>
<td>Year of water-level measurement</td>
</tr>
<tr>
<td>7</td>
<td>Specific capacity</td>
</tr>
<tr>
<td>8</td>
<td>Depth to top of the open section</td>
</tr>
<tr>
<td>9</td>
<td>Diameter of the casing</td>
</tr>
<tr>
<td>10</td>
<td>Number of available water levels</td>
</tr>
<tr>
<td>11</td>
<td>Altitude of bottom of well</td>
</tr>
<tr>
<td>12</td>
<td>Altitude of water level</td>
</tr>
<tr>
<td>13</td>
<td>Altitude of top of the open section</td>
</tr>
</tbody>
</table>

Map-Border Options

Four border options are available to the user. Selection of a particular border option is made using the variable NOLNE on control record 3. The codes for borders and their meanings are
NEITHER or ' ': The plot area will have neither a boxed border nor polygon outlines. The corners of the plot area will be drawn and labeled.

MAP: A boxed or neatline border is drawn around the entire plot area and the outlines of polygons are not drawn.

BASIN: Corners of the plot area are drawn and labeled, and the outlines of all polygons input will be drawn.

BOTH: Both a boxed or neatline border and all polygon outlines are drawn.

Examples of the output of these four options are shown in figure 12.

Program Input

Input to BSNPLT consists of four control records, a BASIN-3 output data file, and a polygon-description file. The polygon file is optional (see section entitled "Map-Border Options").

Control records.--The following four control records are required for the operation of the program.

<table>
<thead>
<tr>
<th>Record</th>
<th>Position</th>
<th>Variable</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-2</td>
<td>IDC1</td>
<td>I2</td>
<td>Control record number must be '01'</td>
</tr>
<tr>
<td></td>
<td>3-80</td>
<td>TITLE</td>
<td>7A10,A8</td>
<td>Title for plot; center is position 40</td>
</tr>
<tr>
<td>2</td>
<td>1-2</td>
<td>IDC2</td>
<td>I2</td>
<td>Control record number must be '02'</td>
</tr>
<tr>
<td></td>
<td>3-4</td>
<td>LLAD</td>
<td>I2</td>
<td>Low latitude of plot, in degrees</td>
</tr>
<tr>
<td></td>
<td>5-6</td>
<td>LLAM</td>
<td>I2</td>
<td>Low latitude of plot, in minutes</td>
</tr>
<tr>
<td></td>
<td>7-8</td>
<td>LLAS</td>
<td>I2</td>
<td>Low latitude of plot, in seconds</td>
</tr>
<tr>
<td></td>
<td>9-11</td>
<td>LLND</td>
<td>I3</td>
<td>Low longitude of plot, in degrees</td>
</tr>
<tr>
<td></td>
<td>12-13</td>
<td>LLNM</td>
<td>I2</td>
<td>Low longitude of plot, in minutes</td>
</tr>
<tr>
<td></td>
<td>14-15</td>
<td>LLNS</td>
<td>I2</td>
<td>Low longitude of plot, in seconds</td>
</tr>
<tr>
<td></td>
<td>16-17</td>
<td>KHLAD</td>
<td>I2</td>
<td>High latitude of plot, in degrees</td>
</tr>
<tr>
<td></td>
<td>18-19</td>
<td>KHLAM</td>
<td>I2</td>
<td>High latitude of plot, in minutes</td>
</tr>
<tr>
<td></td>
<td>20-21</td>
<td>KHLAS</td>
<td>I2</td>
<td>High latitude of plot, in seconds</td>
</tr>
<tr>
<td></td>
<td>22-24</td>
<td>KHLND</td>
<td>I3</td>
<td>High longitude of plot, in degrees</td>
</tr>
<tr>
<td></td>
<td>25-26</td>
<td>KHLNM</td>
<td>I2</td>
<td>High longitude of plot, in minutes</td>
</tr>
<tr>
<td></td>
<td>27-28</td>
<td>KHLNS</td>
<td>I2</td>
<td>High longitude of plot, in seconds</td>
</tr>
<tr>
<td></td>
<td>29-30</td>
<td>KCMD</td>
<td>I3</td>
<td>Central meridian, in degrees</td>
</tr>
</tbody>
</table>
Figure 12.--Border options for areal plot of ground-water site data using program BSNPLT.
32-33  KCMM  I2  Central meridian, in minutes
34-35  KCMS  I2  Central meridian, in seconds
36-37  KHPROJ  I2  Type of map projection
38-44  KSC  I7  Map scale of plot
45-46  NWELLS  I2  Well-site plot switch
47-48  NBRDR  I2  Plotter pen number to be used for polygon boundaries
49-50  NHT  I2  Height of labels; inch/100
51-52  NSHT  I2  Height of symbols; inch/100

3  1-2  IDC3  I2  Control record number must be '03'
3-12  ATIC1  A10  'TIC MARKS'
13-15  ATIC2  A10  'AT'
16-17  ITIC  I2  Tic mark interval in full minutes
18-27  ATIC3  A10  'MIN MARKS'
28-35  ATIC4  A10  'VALS ON'
36-37  INTIC  I2  Full minute on which tic marks are to be drawn
38-47  ATIC5  A10  'MIN MARKS'
48-57  ATIC6  A10  'OUTLINE OF'
58-59
60-66  NOLNE  A7  Map border option

4  1-2  IDC4  I2  Control record number must be '04'
3-5  NID  I1  Number of site-label parameters
4-7  NID1  I2  First label parameter code
6-7  NID2  I2  Second label parameter code
8-11  CNYR1  F4.0  Beginning year of construction
12-15  CNYR2  F4.0  Ending year of construction
16-19  WLYR1  I4  Beginning water-level year
20-23  WLYR2  I4  Ending water-level year
24-28  XDEPTH  F5.0  Depth of well
29-33  WTRLVL  F5.0  Water level
34-38  OPNTOP  F5.0  Depth to top of opening
39  NOPN  I1  Number of type of opening codes to be searched for
40-42  IOPN  3I1  Type of opening codes to be selected
43  NLOG  I1  Number of log types to be searched for
44-51  ILOG  8A1  Log types to be selected
52  IWUSE  I1  Number of water uses to be searched for
53-55  IWUSE  3I1  Water uses to be selected
56  NFIN  I1  Number of finish types to be searched for

1Only Lambert Conformal Conic Projection is active.
Program Output

Output from the program consists of a plotter-command file (punch file) and a printed listing of the options used and sites selected for plotting.

**Site-data report and plot-option listing.**—The listing printed from the program has three main parts:

—Descriptive information about the plot
—Listing of all selection parameters
—Data for selected sites

**Plot description.**—The first page of the listing provides information about the physical characteristics of the plot. This information includes:

—Plot title
—Map projection used
—Map scale of the plot
—High and low latitudes and longitudes represented by the four corners of the plot area
—Longitude that represents the central meridian of the plot area
—Height, in inches of the site symbols
—Height, in inches of the site labels
—Whether or not well sites that met selection criteria were plotted
—Plotter-pen number used to plot the polygon outlines, if plotted
—Interval and placement of tic marks, in minutes

**Selection information listing.**—The second page of the listing identifies the site-label options used, if any, the time ranges used in selection, if any, and the limits of physical parameters used in selection, if any.

**Site-data listing.**—Only those sites that meet all selection criteria are written to the listing and the output data file. The only
difference between the data listed by BSNPLT and those listed by BASIN-3 is the presence of two sequence numbers for each site in the BSNPLT listing. The first number is the sequence number from the BSNPLT program, and the second is the sequence number from the BASIN-3 program.

Error Messages

EITHER THE DATA FILE WAS EMPTY OR ALL SITES OUT OF RANGE

Input data file reached an at-end condition on the first read attempt or all sites in the input data file were outside the plot boundaries.

JOB ABORTED -- TOO LARGE

Width of the plot will exceed 33 inches; reduce the distance between high and low longitudes or increase the value of the map scale.

STRIP CONTROL OUT OF BOUND

The program plots a band of data that roughly corresponds to the width of a slice from the polygon-definition table. The width of the plot band is either negative or very large. The most likely cause is a central meridian outside the range between the low and high longitudes.

FILE OUT OF SEQUENCE

The program assumes that the input ground-water data is sorted in descending order using the site-identification number.

EOF WHILE READING OPTIONS AND CONTROLS

The input data file was in an at-end condition on the first read attempt.

FIRST CONTROL RECORD NOT '01' ID = __

The characters '01' do not appear in positions 1 and 2 of the first control record.
SECOND CONTROL RECORD NOT '02' ID = 

The characters '02' do not appear in positions 1 and 2 of the second control record.

THIRD CONTROL RECORD NOT '03' ID = 

The characters '03' do not appear in positions 1 and 2 of the third control record.

FOURTH CONTROL RECORD NOT '04' ID = 

The characters '04' do not appear in positions 1 and 2 of the fourth control record.

LOW LATITUDE OUTSIDE STATE RANGE

LOW LONGITUDE OUTSIDE STATE RANGE

HIGH LATITUDE OUTSIDE STATE RANGE

HIGH LONGITUDE OUTSIDE STATE RANGE

The high and low latitudes and (or) longitudes for the State of Arizona are tested against the input plot corners, in the four if-statements following line 200 in subroutine NREAD. Either comment these if-statements or change the latitude and (or) longitude values.

LOW LONGITUDE GREATER THAN HIGH LONGITUDE

LOW LATITUDE GREATER THAN HIGH LATITUDE

Check the sequence of low and (or) high latitudes and longitudes on the control records.

NID IS GREATER THAN 2

A value greater than 2 has been entered in position 3 of control record 4.

NID1 HAS INVALID VALUE

NID1 (positions 4-5, control record 4) has a value less than zero or greater than 13.
NID2 HAS INVALID VALUE

NID2 (positions 6-7, control record 4) has a value less than zero or greater than 13.

BEGIN CONSTRUCTION YEAR IS GREATER THAN ENDING CONSTRUCTION YEAR

BEGIN WATER-LEVEL YEAR IS GREATER THAN ENDING WATER-LEVEL YEAR

Check the sequence of values for years on the control records

NLOG HAS INVALID VALUE

Number of log-type codes is less than zero or greater than 8.

NWUS HAS INVALID VALUE

Number of water-use codes is less than zero or greater than 3.

NFIN HAS INVALID VALUE

Number of finish codes is less than zero or greater than 3.

NOPN HAS INVALID VALUE

Number of type-of-opening codes is less than zero or greater than 3.

REFERENCES CITED


ATTACHMENTS
ATTACHMENT A

COBOL record descriptions for WATSTORE record types.
01 RO-GENERAL-SITE-DATA.
 03 RO-KEYID.
    05 RO-KEY-LATITUDE PIC 9(6).
    05 RO-KEY-LONGITUDE PIC 9(7).
    05 RO-KEY-DUP-NUM PIC 999.
 03 RO-REC-NUM PIC 999.
 03 90-KEY-LENGTH PIC 999.
 03 R0-ACCESS-CODE PIC X.
 03 C2-SITE-TYPE PIC X.
 03 C3-DATA-RELIABILITY PIC X.
 03 C4-SOURCE-AGENCY PIC X(5).
 03 C5-PROJECT-NUM.
    05 C5-BASIN-1 PIC X(7).
    05 C5-HASIN-2 PIC X(6).
 03 C6-DISTRICT PIC 999.
 03 C7-STATE PIC 999.
 03 C8-COUNTY PIC 999.
 03 C9-LATITUDE PIC 9(6).
 03 C10-LONGITUDE PIC 9(7).
 03 C11-LAT-LONG-ACURACY PIC X.
 03 C12-LOCAL-NUMBER.
    05 C12-LOCAL-SITE PIC X(14).
    05 C12-LOCAL-REST PIC X(6).
 03 C13-LAND-MAP-LOC PIC X(30).
 03 C14-LOCATION-SCALE PIC X(30).
 03 C15-MAP-SCALE PIC 9(6).
 03 C16-ALTITUDE PIC 9(5)V99.
 03 C17-ALTITUDE-METHOD PIC X.
 03 C18-ALTITUDE-ACCURACY PIC XXX.
 03 C19-TOPO-SETTING PIC X.
 03 C20-OWDC-HYDRO-UNIT PIC X(8).
 03 C21-DATE-CONSTRUCTED.
    05 C21-MONTH PIC XX.
    05 C21-DAY PIC XX.
    05 C21-YEAR PIC XXXX.
 03 C23-SITE-USE PIC X.
 03 C24-WATER-USE PIC X.
 03 C25-SECOND-WATER-USE PIC X.
 03 C26-THIRD-WATER-USE PIC X.
 03 C27-HOLE-DEPTH PIC 9(5)V99.
 03 C28-WELL-DEPTH PIC 9(5)V99.
 03 C29-WELL-DEPTH-SOURCE PIC X.
 03 C30-WATER-LEVEL PIC 9(5)V99.
 03 C31-WATER-LEVEL-DATE PIC 9(8).
 03 C32-DATE-ACCRCY-WL PIC X.
 03 C33-WATER-LEVEL-SOURCE PIC X.
 03 C34-MEAS-METHOD-WL PIC X.
 03 C35-PUMP-USED PIC X.
 03 C36-GENHYRD-O-DATA-SOURCE PIC X.
 03 C37-SITE-STATUS PIC X.
 03 C40-LAST-UPDATE PIC X(8).

A-1
********** LIFT DATA **********

01 R42-LIFT-DATA.
  03 R42-KEYID.
    05 R42-KEY-LATITUDE PIC 9(6).
    05 R42-KEY-LONGITUDE PIC 9(7).
    05 R42-KEY-DUP-NUM PIC 99.
  03 R42-REC-NUM PIC 999.
  03 R42-REC-LENGTH PIC 999.
  03 R42-ACCESS-CODE PIC X.
  03 C43-LIFT-TYPE PIC X.
  03 C254-LIFT-ENTRY-NO PIC 999.
  03 C44-INTAKE-SETTING PIC 9(5).
  03 C45-POWER-TYPE PIC X.
  03 C38-LIFT-DATE PIC X(10).
  03 C46-HORSEPOWER PIC 9(4)V9.

********** MAJOR PUMP DATA **********

01 R47-MAJOR-PUMP-DATA.
  03 R47-KEYID.
    05 R47-KEY-LATITUDE PIC 9(6).
    05 R47-KEY-LONGITUDE PIC 9(7).
    05 R47-KEY-DUP-NUM PIC 99.
  03 R47-REC-NUM PIC 999.
  03 R47-REC-LENGTH PIC 999.
  03 R47-ACCESS-CODE PIC X.
  03 C43-LIFT-TYPE PIC X.
  03 C48-MANUFACTURER PIC X(8).
  03 C49-serial-no PIC X(12).
  03 C50-POWER-COMPANY PIC X(12).
  03 C51-ACCOUNT PIC X(10).
  03 C52-METER PIC X(12).
  03 C53-CONSUMPTION PIC 99V9(5).
  03 C54-PUMP-MAINTAINER PIC X(12).
  03 C254-LIFT-ENTRY-NO PIC 999.
  03 C255-ADDL-LIFT PIC 999.
  03 C268-PUMP-CAPACITY PIC 99999.

A-2
01 R55-STANDBY-POWER-DATA.
   03 R55-KEYID.
      05 R55-KEY-LATITUDE PIC 9(6).
      05 R55-KEY-LONGITUDE PIC 9(7).
      05 R55-KEY-DUP-NUM PIC 99.
   03 R55-REC-NUM PIC 999.
   03 R55-REC-LENGTH PIC 999.
   03 R55-ACCESS-CODE PIC X.
   03 C43-LIFT-TYPE PIC X.
   03 C56-STANDBY-POWER-TYPE PIC X.
   03 C57-STANDBY-HORSEPOWER PIC 9(4)V99.
   03 C254-LIFT-ENTRY-NO PIC 999.

01 R58-WELL-CONST-DATA.
   03 R58-KEYID.
      05 R58-KEY-LATITUDE PIC 9(6).
      05 R58-KEY-LONGITUDE PIC 9(7).
      05 R58-KEY-DUP-NUM PIC 99.
   03 R58-REC-NUM PIC 999.
   03 R58-REC-LENGTH PIC 999.
   03 R58-ACCESS-CODE PIC X.
   03 C59-CONST-SEQ-NO PIC 9(3).
   03 C60-DATE-COMPLETED PIC X(10).
   03 C64-CONST-DATA-SOURCE PIC X.
   03 C63-CONTRACTOR PIC X(12).
   03 C65-CONST-METHOD PIC X.
   03 C66-FINISH PIC X.
   03 C67-SEAL-TYPE PIC X.
   03 C68-SEAL-BOTTOM PIC 9(4).
   03 C69-DEVELOPMENT-METHOD PIC X.
   03 C70-DEVELOPMENT-DURATION PIC 999.
   03 C71-SPECIAL-TREATMENT PIC X.
********** DIMENSION OF HOLE CONSTRUCTED **********

01 R72-DIMENSION-HOLE-CONST.
  03 R72-KEYID.
    05 R72-KEY-LATITUDE PIC 9(6).
    05 R72-KEY-LONGITUDE PIC 9(7).
    05 R72-KEY-DUP-NUM PIC 99.
  03 R72-REC-NUM PIC 999.
  03 R72-REC-LENGTH PIC 999.
  03 R72-ACCESS-CODE PIC X.
  03 C79-CONST-SEQ-NU PIC 999.
  03 C80-DATE-COMPLETED PIC X(8).
  03 C73-HOLE-TOP PIC 9(5)V99.
  03 C74-HOLE-BOTTOM PIC 9(5)V99.
  03 C75-HOLE-DIAMETER PIC 999V99.

********** CASING DATA **********

01 R76-CASING-DATA.
  03 R76-KEYID.
    05 R76-KEY-LATITUDE PIC 9(6).
    05 R76-KEY-LONGITUDE PIC 9(7).
    05 R76-KEY-DUP-NUM PIC 99.
  03 R76-REC-NUM PIC 999.
  03 R76-REC-LENGTH PIC 999.
  03 R76-ACCESS-CODE PIC X.
  03 C59-CONST-SEQ-NU PIC 999.
  03 C60-DATE-COMPLETED PIC X(10).
  03 C77-CASING-THICKNESS PIC 999V99.
  03 C78-CASING-BOTTOM PIC 9(5)V99.
  03 C79-CASING-DIAMETER PIC 9(3)V99.
  03 C80-CASING-MATERIAL PIC X.
********** OPENINGS SCHEDULE **********

01 R82-OPENINGS-SCHEDULE.
   03 R82-KEYID.
      05 R82-KEY-LATITUDE PIC 9(6).
      05 R82-KEY-LONGITUDE PIC 9(7).
      05 R82-KEY-DUP-NUM PIC 99.
   03 R82-REC-NUM PIC 999.
   03 R82-REC-LENGTH PIC 999.
   03 R82-ACCESS-CODE PIC X.
   03 C59-CONST-SEQ-NO PIC 999.
   03 C60-DATE-COMPLETE PIC X(10).
   03 C83-OPENING-TOP PIC 9(5)V99.
   03 C84-OPENING-BOTTOM PIC 9(5)V99.
   03 C85-OPENING-TYPE PIC X.
   03 C86-SCREEN-MATERIAL PIC X.
   03 C87-OPENING-DIAMETER PIC 999V99.
   03 C88-OPENING-WIDTH PIC 99V99.
   03 C89-OPENING-LENGTH PIC 999V99.

********** GEOHYDROLOGIC UNIT DESCRIPTION **********

01 R90-GEOHYDRO-UNIT-DESCRIPTION.
   03 R90-KEYID.
      05 R90-KEY-LATITUDE PIC 9(6).
      05 R90-KEY-LONGITUDE PIC 9(7).
      05 R90-KEY-DUP-NUM PIC 99.
   03 R90-REC-NUM PIC 999.
   03 R90-REC-LENGTH PIC 999.
   03 R90-ACCESS-CODE PIC X.
   03 C256-GEOHYDRO-SEQ-NO PIC 999.
   03 C91-TOPT-DEPTH PIC 9(5)V99.
   03 C92-BOTTOM-DEPTH PIC 9(5)V99.
   03 C93-GEOHYDRO-UNIT PIC X(8).
   03 C304-GEOHYDRO-C-UNIT PIC X.
   03 C96-LITHOLOGY PIC X(4).
   03 C97-LITHOLOGIC-MODIFIER PIC X(16).
********** AQUIFER DATA **********

01 R94-AQUIFER-DATA.
  03 R94-KEYID.
   05 R94-KEY-LATITUDE PIC 9(6).
   05 R94-KEY-LONGITUDE PIC 9(7).
   05 R94-KEY-DUP-NUM PIC 99.
  03 R94-REC-NUM PIC 999.
  03 R94-REC-LENGTH PIC 999.
  03 R94-ACCESS-CODE PIC X.
  03 C25b-GEOHYDRO-SEQ-NO PIC 999.
  03 C95-AQUIFER-DATE PIC X(10).
  03 C126-AQUIFER-STATIC-LEVEL PIC 9(5)V99.
  03 C132-AQUIFER-CONTRIBUTION PIC 999.

********** WATER QUALITY DATA COLLECTION **********

01 R114-QW-DATA-COLLECT.
  03 R114-KEYID.
   05 R114-KEY-LATITUDE PIC 9(6).
   05 R114-KEY-LONGITUDE PIC 9(7).
   05 R114-KEY-DUP-NUM PIC 99.
  03 R114-REC-NUM PIC 999.
  03 R114-REC-LENGTH PIC 999.
  03 R114-ACCESS-CODE PIC X.
  03 C115-QN-BEGIN-YEAR PIC 9999.
  03 C116-QN-END-YEAR PIC 9999.
  03 C117-QN-DATA-SOURCE PIC X(5).
  03 C118-QN-FREQUENCY PIC X.
  03 C257-QN-NETWORK PIC X.
  03 C120-QN-ANALYSIS-TYPE PIC X.
  03 C307-QN-ANAL-AGENCY PIC X(5).
**WATER LEVEL DATA COLLECTION**

01 R121-WL-DATA-COLLECT.
  03 R121-KEYID.
    05 R121-KEY-LATITUDE   PIC 9(6).
    05 R121-KEY-LONGITUDE  PIC 9(7).
    05 R121-KEY-DUP-NUM    PIC 99.
  03 R121-REC-NUM         PIC 999.
  03 R121-REC-LENGTH      PIC 999.
  03 R121-ACCESS-CODE     PIC X.
  03 C122-LN-BEGIN-YEAR   PIC 9999.
  03 C123-LN-END-YEAR     PIC 9999.
  03 C124-LN-DATA-SOURCE  PIC X(5).
  03 C125-LN-FREQUENCY    PIC X.
  03 C258-LN-NETWORK      PIC X.

**WATER PUMPAGE/WITHDRAWAL DATA COLLECTION**

01 R127-PUMP-WITHDRAWAL-DATA.
  03 R127-KEYID.
    05 R127-KEY-LATITUDE   PIC 9(6).
    05 R127-KEY-LONGITUDE  PIC 9(7).
    05 R127-KEY-DUP-NUM    PIC 99.
  03 R127-REC-NUM         PIC 999.
  03 R127-REC-LENGTH      PIC 999.
  03 R127-ACCESS-CODE     PIC X.
  03 C128-PN-BEGIN-YEAR   PIC 9(4).
  03 C129-PN-END-YEAR     PIC 9(4).
  03 C130-PN-DATA-SOURCE  PIC X(5).
  03 C131-PN-FREQUENCY    PIC X.
  03 C132-PN-NETWORK      PIC X.
  03 C133-PN-DATA-METHOD  PIC X.
********** PRODUCTION DATA **********

R = 134 -- FLOWING
R = 146 -- PUMPED

01 R134-46-PRODUCTION-DATA.
  03 R134-46-KEYID.
    05 R134-46-KEY-LATITUDE PIC 9(6).
    05 R134-46-KEY-LONGITUDE PIC 9(7).
    05 R134-46-KEY-DUP-NUM PIC 99.
  03 R134-46-REC-NUM PIC 999.
  03 R134-46-REC-LENGTH PIC 999.
  03 R134-46-ACCESS-CODE PIC X.
  03 C147-PUMP-SEQ-NO PIC 999.
  03 C150-PUMP-DISCHARGE PIC 9(5)V99.
  03 C151-PUMP-DISCHRG-SOURCE PIC X.
  03 C152-PUMP-DISCHRG-METHOD PIC X.
  03 C153-PUMP-PROD-LEVEL PIC 9(5)V99.
  03 C154-PUMP-STATIC-LEVEL PIC 9(5)V99.
  03 C155-PUMP-LEVEL-SRC PIC X.
  03 C309-PUMP-DRAWDOWN PIC 999V99.
  03 C272-PUMP-SPECIFIC-CAP PIC 9(5)V99.
  03 C156-PUMP-LEVEL-METHOD PIC X.
  03 C157-PUMP-PERIOD PIC 9(4)V99.

********** OWNER IDENTIFICATION **********

01 R158-OWNER-IDENTIFICATION.
  03 R158-KEYID.
    05 R158-KEY-LATITUDE PIC 9(6).
    05 R158-KEY-LONGITUDE PIC 9(7).
    05 R158-KEY-DUP-NUM PIC 99.
  03 R158-REC-NUM PIC 999.
  03 R158-REC-LENGTH PIC 999.
  03 R158-ACCESS-CODE PIC X.
  03 C159-OWNERSHIP-DATE PIC X(10).
  03 C161-LAST-NAME PIC X(10).
  03 C162-FIRST-NAME PIC X(8).
  03 C163-MIDDLE-INITIAL PIC X.
********** OTHER SPRING DATA **********

01 R171.
  03 C1-KEYID.
    05 R171-KEY-LATITUDE PIC 9(6).
    05 R171-KEY-LONGITUDE PIC 9(7).
    05 R171-KEY-DUP-NUM PIC 99.
  03 R171-REC-LENGTH PIC 999.
  03 R171-ACCESS-CODE PIC X.
  03 C172-SPRING-NAME PIC X(20).
  03 C173-SPRING-TYPE PIC X.
  03 C174-PERMANENCE PIC X.
  03 C175-DISCHARGE-SPHERE PIC X.
  03 C176-IMPROVEMENTS PIC X.
  03 C177-NO-SPRING-OPENINGS PIC 999.
  03 C178-FLOW-VARIABILITY PIC 999.
  03 C179-FLOW-VAR-ACCURACY PIC X.

********** OTHER DATA AVAILABLE **********

01 R180-OTHER-DATA-AVAIL.
  03 R180-KEYID.
    05 R180-KEY-LATITUDE PIC 9(6).
    05 R180-KEY-LONGITUDE PIC 9(7).
    05. R180-KEY-DUP-NUM PIC 99.
  03 R180-REC-LENGTH PIC 999.
  03 R180-ACCESS-CODE PIC X.
  03 C312-OD-ENTRY-NO PIC XXX.
  03 C181-TYPE-DATA PIC X(10).
  03 C181-DATA-LOCATION PIC X.
  03 C261-DATA-FORMAT PIC X.

A-9
********** PERTINENT REMARKS **********

01 R183-PERTINENT-REMARKS.
  03 R183-KEYID.
    05 R183-KEY-LATITUDE PIC 9(6).
    05 R183-KEY-LONGITUDE PIC 9(7).
    05 R183-KEY-DUP-NUM PIC 99.
  03 R183-RFC-NUM PIC 999.
  03 R183-REC-LENGTH PIC 999.
  03 R183-ACCESS-CODE PIC X.
  03 C185-PR-SEQ-NO PIC 999.
  03 C185-PERT-REMARK PIC X(40).

********** SITE VISIT DATA **********

01 R186-SITE-VISIT-DATA.
  03 C1-KEYID.
    05 R186-KEY-LATITUDE PIC 9(6).
    05 R186-KEY-LONGITUDE PIC 9(7).
    05 R186-KEY-DUP-NUM PIC 99.
  03 R186-REC-NUM PIC 999.
  03 R186-REC-LENGTH PIC 999.
  03 R186-ACCESS-CODE PIC X.
  03 C187-INVENTORY-DATE PIC X(10).
  03 C188-INVENTORY-PERSON PIC X(10).
********** OTHER SITE IDENTIFICATION NUMBERS **********

01 R189-OTHER-SITE-ID-NOS.
  03 R189-KEYID.
    05 R189-KEY-LATITUDE PIC 9(6).
    05 R189-KEY-LONGITUDE PIC 9(7).
    05 R189-KEY-DUP-NUM PIC 99.
  03 R189-REC-NUM PIC 999.
  03 R189-RFC-LENGTH PIC 999.
  03 R189-ACCESS-CODE PIC X.
  03 C190-OTHER-ID PIC X(10).
  03 C191-OTHER-ID-ASSIGNEK PIC X(15).

********** FIELD WATER QUALITY MEASUREMENT **********

01 R192-FWQ-MEASUREMENT.
  03 R192-KEYID.
    05 R192-KEY-LATITUDE PIC 9(6).
    05 R192-KEY-LONGITUDE PIC 9(7).
    05 R192-KEY-DUP-NUM PIC 99.
  03 R192-REC-NUM PIC 999.
  03 R192-REC-LENGTH PIC 999.
  03 R192-ACCESS-CODE PIC X.
  03 C193-FWQ-SAMPLE-DATE PIC X(10).
  03 C195-FWQ-GEOHYDRO-UNIT PIC X(8).
  03 C196-FWQ-PARAMETER PIC 9(5).
  03 C197-FWQ-MEASUREMENT PIC 9(6)V9.
********** AVAILABLE LOG DATA **********

01 R198-AVIAL-LOG-DATA.
  03 R198-KEYID.
    05 R198-KEY-LATITUDE PIC 9(6).
    05 R198-KEY-LONGITUDE PIC 9(7).
    05 R198-KEY-DUP-NUM PIC 99.
  03 R198-REC-NUM PIC 999.
  03 R198-REC-LENGTH PIC 999.
  03 R198-ACCESS-CODE PIC X.
  03 C199-LOG-TYPE PIC X.
  03 C200-LOG-TOP PIC 9(5)V99.
  03 C201-LOG-BOTTOM PIC 9(5)V99.
  03 C202-LOG-SOURCE PIC X.

********** WATER LEVEL MEASUREMENT SCHEDULE **********

01 R234-WL-MEASUREMENT.
  03 R234-KEYID.
    05 R234-KEY-LATITUDE PIC 9(6).
    05 R234-KEY-LONGITUDE PIC 9(7).
    05 R234-KEY-DUP-NUM PIC 99.
  03 R234-REC-NUM PIC 999.
  03 R234-REC-LENGTH PIC 999.
  03 R234-ACCESS-CODE PIC X.
  03 C235-WL-MEAS-DATE PIC X(10).
  03 C238-WL-REMARK PIC X.
  03 C239-WL-MEAS-METHOD PIC X.
  03 C240-WL-TIME-PICK PIC X.
01 R320-MEASURING-POINT.
  03 R320-KEYID.
    05 R320-KEY-LATITUDE PIC 9(b).
    05 R320-KEY-LONGITUDE PIC 9(7).
    05 R320-KEY-DUP-NUM PIC 99.
  03 R320-REC-NUM PIC 999.
  03 R320-KFC-LENGTH PIC 999.
  03 R320-ACCESS-CODE PIC X.
  03 C321-MP-BEGIN-YEAR PIC 9(4).
  03 C322-MP-END-YEAR PIC 9(4).
  03 C323-MP-HEIGHT PIC 999V99.
  03 C324-MP-REMARK PIC X(40).
ATTACHMENT B

Program listings.
PROGRAM PLGNO1(INPUT, OUTPUT, TAPE5=INPUT, TAPE6=OUTPUT,
$TAPE15, TAPE16)


DO 500 I=1,2000
C........ZERO OUT ALL TABLES
C........
EQA(I) = 0.
EQB(I) = 0.
E QC(I) = 0.
LA1(I) = 0
LN1(I) = 0
LA2(I) = 0
LN2(I) = 0
X(I) = 0.0
Y(I) = 0.0
Z(I) = 0.0
W(I) = 0.0
500 CONTINUE
C........STORE INITIAL VALUES FOR SETTING UP THE HIGH AND
C........LOW LATITUDES AND LONGITUDES
C........
LATL = 999999
LONL = -999999
LATH = 0
LONH = 0
C........READ THE BASIN IDENTIFIER RECORD
C........
READ(5,600) BCODE, BEXPL1, BEXPL2, BEXPL3, BEXPL4, BEXPL5
600 FORMAT(26X,A8,A10,A6)
I=1
C........READ AND STORE THE VERTICES IN THE SEQUENCE AS READ
C........STORE BOTH FIXED AND FLOATING POINT VALUES
C........
C........THE FOUR "IF" STATEMENTS SELECT THE HIGH AND LOW
C........VALUES FOR LATITUDE AND LONGITUDE
C........
18 READ(5,610) NLAD, NLAM, NLAS, MLOD, MLOM, MLOS
IF(EQF(5))14, 13
601 FORMAT(I3,1X,I2,1X,I2,2X,I3,1X,I2,1X,I2)
C........ 13 WRITE(6,602) NLAD, NLAM, NLAS, MLOD, MLOM, MLOS
C........602 FORMAT(1X,13,1X,12,2X,13,1X,12,1X,12)
13 LA1(I) = NLAD*10000+NLAM*100+NLAS
LN1(I) = -(MLOD*10000+MLOM*100+MLOS)
XLOD = MLOD
XLOM = MLOM
XLOS = MLOS
X(I) = -(XLOD+(XLOM/60.0)+(XLOS/3600.0))
YLD = NLAD
YLM = NLAM
YLAS = NLAS

Y(I) = YLA1 + (YLAM/60.0) + (YLAS/3600.0)
 IF (LA1(I) .LT. LA1(L)) LA1(L) = LA1(I)
 IF (LA1(I) .GT. LATH) LATH = LA1(I)
 IF (LN1(I) .GT. LONL) LONL = LN1(I)
 IF (LN1(I) .LT. LONH) LONH = LN1(I)
 I = I + 1
 GO TO 1

C
C.. SET NUMBER OF POINTS TO "NUMVER"
C.. STORE FIRST VERTEX IN THE LAST PLACE TO CLOSE POLYGON
C..  
14 NUMVER = I - 1
   K = I
   LA1(I) = LA1(1)
   LN1(I) = LN1(1)
   X(I) = X(1)
   Y(I) = Y(1)
C
   DO 501 I = 1, K
   C.. WRITE (6, 603) LA1(I), Y(I), LN1(I), X(I), I
C.. 603 FORMAT(1X, "LA1 = ", F11.8, 
C.. 9", LN1 = ", F13.8, " INDEX ", I5)
   C
   501 CONTINUE
   C.. WRITE(6, 604) LATL, LATH, LONL, LONH
   C.. 604 FORMAT(1X, "LOW LAT = ", I9, " HIGH LAT = ", I6, " LOW LNG = ",
C.. 9I9, " HIGH LNG = ", I9)
   C
   DO 605 FORMAT(1X, "FOLLOWING IS THE TABLE OF THE MATCHED PAIRS")
   C
C.. LINE SEGMENTS DEFINED BY PUTTING SECOND POINTS IN TABLE
C..  
DO 502 I = 1, NUMVER
   LA2(I) = LA1(I + 1)
   LN2(I) = LN1(I + 1)
   Z(I) = X(I + 1)
   W(I) = Y(I + 1)
C
   WRITE(6, 606) X(I), Y(I), Z(I), W(I), I
C.. 606 FORMAT(1X, " X(I) = ", F13.8, " Y(I) = ", F12.8,
C.. 9" Z(I) = ", F13.8, " W(I) = ", F12.8, I5)
   502 CONTINUE
   C
   WRITE(6, 607)
   C.. 607 FORMAT(1X, "TABLE WITH LOWEST LATITUDE FIRST")
   C
C.. REARRANGE POINTS SO LINE SEGMENTS ARE DIRECTED FROM
C.. LOWER TO HIGHER LATITUDE
C..
   DO 103 I = 1, NUMVER
   C
   WRITE(6, 104) X(I), Y(I), Z(I), W(I), I
   C.. 104 FORMAT(1X, " X(I) = ", F13.8, " Y(I) = ", F12.8,
C.. 9" Z(I) = ", F13.8, " W(I) = ", F12.8, I5)
   103 CONTINUE
   DO 503 I = 1, NUMVER
   IF (W(I) .GT. Y(I)) GO TO 503
   TEMP = Y(I)
   Y(I) = W(I)
   B-2
w(I) = TEMP
TEMP = X(I)
x(I) = Z(I)
z(I) = TEMP
KEMP = LA1(I)
LA1(I) = LA2(I)
LA2(I) = KEMP
KEMP = LN1(I)
LN1(I) = LN2(I)
LN2(I) = KEMP

503 CONTINUE
C....... DO 504 I=1,NUMVER
C....... WRITE(6,608)X(I),Y(I),Z(I),w(I),I
C....... 504 CONTINUE
C....... ROUTINE TO ELIMINATE PARALELLS
C....... NUM = NUMVER
I = 0
505 IF (I+1.GE.NUMVER) GO TO 507
I = I + 1
IF (Y(I).NE.W(I)) GO TO 505
C....... THIS DROPS THE LINE AND REDUCES NUMBER OF LINES
C....... WRITE(6,609)X(I),LN1(I),Y(I),LA1(I),Z(I),LN2(I),w(I),LA2(I)
NUMVER = NUMVER - 1
J = I
C....... LAST LINE IN TABLE TEST
C....... 506 IF (J.GT.NUMVER) GO TO 505
X(J) = X(J+1)
Y(J) = Y(J+1)
z(J) = Z(J+1)
W(J) = W(J+1)
LA1(J) = LA1(J+1)
LA2(J) = LA2(J+1)
LN1(J) = LN1(J+1)
LN2(J) = LN2(J+1)
J = J + 1
GO TO 506
C....... THE SORT ROUTINE COMPARES TWO LINES AND MOVES
C....... THE ONE WITH THE LOWEST FIRST LATITUDE TO THE TOP
C....... IF TWO FIRST LATITUDES ARE EQUAL THEN IT CHECKS THE
C....... LONGITUDES OF THE FIRST POINTS AND MOVES THE LARGEST TO
C....... THE TOP POSITION
C.......
507 K = NUMVER - 1
LSSW = -23
DO 508 I = 1, K
IF (Y(I) .LT. Y(I+1)) GO TO 508
IF (Y(I) .EQ. Y(I+1)) GO TO 51
50 TEMP = Y(I)
Y(I) = Y(I+1)
Y(I+1) = TEMP
TEMP = X(I)
X(I) = X(I+1)
X(I+1) = TEMP
TEMP = Z(I)
Z(I) = Z(I+1)
Z(I+1) = TEMP
TEMP = W(I)
W(I) = W(I+1)
W(I+1) = TEMP
KEMP = LN1(I)
LN1(I) = LN1(I+1)
LN1(I+1) = KEMP
KEMP = LA1(I)
LA1(I) = LA1(I+1)
LA1(I+1) = KEMP
KEMP = LN2(I)
LN2(I) = LN2(I+1)
LN2(I+1) = KEMP
KEMP = LA2(I)
LA2(I) = LA2(I+1)
LA2(I+1) = KEMP
LSSW = -23
GO TO 508
1508 IF (Z(I) .LT. Z(I+1)) GO TO 50
GO TO 508
51 IF (X(I) .EQ. X(I+1)) GO TO 1508
IF (X(I) .LT. X(I+1)) GO TO 50
GO TO 508
508 CONTINUE
IF (LSSW .LT. 0) GO TO 507
C.....THIS IS THE END OF THE SORT ROUTINE
C.....NOW THE SORTED TABLE IS PRINTED OUT THIS WILL BE
C.....MADE INACTIVE AFTER THE PROGRAM IS TESTED
C.....NOW THE COEFFICIENTS OF THE LINES ARE COMPUTED AND STORED
C.....DO 509 I=1, NUMVER
C.....WRITE(6,610)X(I),LN1(I),Y(I),LA1(I),Z(I),LN2(I),W(I),LA2(I)
C.....509 CONTINUE
DO 510 I=1, NUMVER
EQA(I) = W(I) - Y(I)
EQB(I) = X(I) - Z(I)
EQC(I) = X(I)*W(I) - Z(I)*Y(I)
510 CONTINUE

C...... THIS PRINTS OUT THE FIXED POINT VALUES OF THE POINTS FOR
C...... EACH LINE AND THE COMPUTED COEFFICIENTS IN FLOATING POINT
C...... THIS TO BE DEACTIVATED AFTER PROGRAM TEST
C......
C...... DO 511 I=1, NUMVER
C...... WRITE(6, 611) LN1(I), LA1(I), LN2(I), LA2(I), EQA(I), EQB(I), EQC(I)
C...... 611 FORMAT (1X, 4I8, 3F13.8)
C...... 511 CONTINUE
C......
C...... PUT IN TABLE END INDICATORS IN THE LAST PLUS ONE
C...... POSITION IN THE TABLE SO THAT TABLE END AND
C...... PROGRAM END CAN BE RECOGNIZED
C......
I = NUMVER + 1
LA1(I) = 9999999
Y(I) = 999.999
X(I) = 999.999
Z(I) = 999.999
W(I) = 999.999
I = -1

C...... THIS IS THE CONTROLLING ROUTINE FOR PRODUCING THE
C...... ADDITIONAL LINE SEGMENTS TO PRODUCE SLICES OF THE
C...... POLYGON AS A SET OF TRAPEZIIDS OR TRIANGLE
C......
512 I = I*2
LLSW = +23
IF (I.GT. NUMVER) GO TO 900
513 IF (Y(I), NE. Y(I+1)) GO TO 800
IF (W(I), NE. W(I+1)) GO TO 514
IF ((W(I), GT. Y(I+2)), AND. (W(I+1), GT. Y(I+2))) GO TO 517
IF ((W(I), GT. Y(I+2)), AND. (Y(I+2), LT. 999.9)) GO TO 802
IF (Y(I+2), EQ. 999.999) GO TO 900
IF (X(I), GE. X(I+1)) GO TO 512
520 TEMP = Y(I)
Y(I) = Y(I+1)
Y(I+1) = TEMP
TEMP = X(I)
X(I) = X(I+1)
X(I+1) = TEMP
TEMP = Z(I)
Z(I) = Z(I+1)
Z(I+1) = TEMP
TEMP = W(I)
w(I) = W(I+1)
w(I+1) = TEMP
KEMP = LN1(I)
LN1(I) = LN1(I+1)
LN1(I+1) = KEMP
KEMP = LA1(I)
LA1(I) = LA1(I+1)
LA1(I+1) = KEMP
KEMP = LN2(I)
LN2(I) = LN2(I+1)
LN2(I+1) = KEMP
KEMP = LA2(I)
LA2(I) = LA2(I+1)
LA2(I+1) = KEMP
TEMP = EQA(I)
EQA(I) = EQA(I+1)
EQA(I+1) = TEMP
TEMP = EQB(I)
EQB(I) = EQB(I+1)
EQB(I+1) = TEMP
TEMP = EQC(I)
EQC(I) = EQC(I+1)
EQC(I+1) = TEMP
IF (LLSW) 750,750,512
C........
C........THIS ROUTINE IS ENTERED WITH Y(I) = Y(I+1) AND
C........W(I) NOT = TO W(I+1) TEST WILL BE MADE AND A BRANCH
C........TO 517 "IF ((W(I).GT.Y(I+2)).AND.(W(I+1).GT.Y(I+2)))"
C........BRANCH TO 534 TAKEN IF W(I).GT.W(I+1)
C........
514 IF ((W(I).GT.Y(I+2)).AND.(W(I+1).GT.Y(I+2))) GO TO 517
IF (W(I).GT.W(I+1)) GO TO 534
IF (W(I).NE.Y(I+2)) GO TO 898
TX = 0.
TY = W(I)
T7 = Z(I+1)
TW = W(I+1)
TA = EQA(I+1)
TB = EQB(I+1)
TC = EQC(I+1)
KX = 0
KY = LA2(I)
KZ = LN2(I+1)
KW = LA2(I+1)
LA2(I+1) = LA2(I)
W(I+1) = W(I)
IF (TA.EQ.0.0) GO TO 803
TX = (TC-TB*TY)/TA
Z(I+1) = TX
KD = TX
XKD = KD
TX = TX-XKD
KS = TX*3600.*.5
KM = KS/60
KS = KS-(KM*60)
KX = KD*10000+KM*100+KS
LN2(I+1) = KX
TX = Z(I+1)
GO TO 515

B-6
534 IF (Y(I+2) .EQ. 999.9990) GO TO 834
    IF (W(I+1) .NE. Y(I+2)) GO TO 899
    TX = 0.0
    TY = W(I+1)
    TZ = Z(I)
    TW = W(I)
    TA = EQA(I)
    TB = EQB(I)
    TC = EQC(I)
    KX = 0
    KY = LA2(I+1)
    KZ = LN2(I)
    KW = LA2(I)
    LA2(I) = LA2(I+1)
    W(I) = W(I+1)
    IF (TA .EQ. 0.0) GO TO 803
    TX = (TC - TB * TY) / TA
    Z(I) = TX
    KD = TX
    XKD = KD
    TX = TX - XKD
    KS = TX * 3600. + .5
    KM = KS / 60
    KS = KS - (KM * 60)
    KX = KD * 10000 + KM * 100 + KS
    LN2(I) = KX
    TX = Z(I)
    GO TO 515
C......
C......THIS ROUTINE IS ENTERED WITH Y(I) = Y(I+1)
C......AND W(I) > Y(I+2)
C......
C......
C......10.0  4  11.0  7
C......10.1  4  11.0  9  I+1
C......10.0  4  11.1  11  I+2
C......11.0  4  10.0  14  I+3
C......
517 IF (Y(I+2) .NE. Y(I+3)) GO TO 802
    IF (Y(I+2) .EQ. Y(I)) GO TO 1517
    IF (Y(I+2) .LT. Y(I)) GO TO 802
    TR = Y(I+2)
    KR = LA1(I+2)
    LLSW = -23
    MI = I - 1
730 IF (MI .GE. I+1) GO TO 519
    MI = MI + 1
    IF (W(MI) .LE. TR) GO TO 730
    TX = 0.0
    TY = TR
    TZ = Z(MI)
    TW = W(MI)
    TA = EQA(MI)
    TB = EQB(MI)
    TC = EQC(MI)
KX = 0  
KY = KR  
KZ = LN2(MI)  
KW = LA2(MI)  
LA2(MI) = KR  
Z(MI) = KR  
rt(MT) = TR  
IF (TA.EQ.0.0) GO TO 803  
TX = (TY+TB*TY)/TA  
Z(MI) = TX  
KD = TX  
XKD = KD  
TX = TX-XKD  
KS = TX*3600.+5  
KM = KS/60  
KS = KS-(KM*60)  
KX = KD*10000+KM*100+KS  
LN2(MI) = KX  
TX = Z(MI)  
GO TO 515  
519 MI = I  
KN = I +2  
  
C.......FIND LAST LINE TO BE SORTED  
C.......  
717 IF (Y(KN).NE.Y(KN+1)) GO TO 749  
KN = KN +1  
GO TO 717  
C.......KN = INDEX OF LAST LINE TO BE SORTED  
C.......  
C......SORT ALL LINES WITH EQUAL FIRST LATITUDES  
C......BY THEIR FIRST LONGITUDES  
C.......  
749 NVER = +23  
I = MI +2  
750 IF (X(I).GT.X(I+1)) GO TO 751  
IF((X(I).EQ.X(I+1)).AND.(Z(I).GE.Z(I+1)))GO TO 751  
NVER = -23  
GO TO 520  
751 IF (I.GE.KN-1) GO TO 752  
I = I +1  
GO TO 750  
752 IF (NVER.LT.0) GO TO 749  
C.......  
C......END OF SORT BY FIRST LONGITUDES  
C.......  
C.......RETURN TO CONTROL ROUTINE  
C.......  
I = MI  
GO TO 512
C NOW FIND LOWEST OF SECOND LATITUDES
C...... AND THE NEXT FIRST LATITUDE
C...... AND USE AS THE SECOND LATITUDE FOR
C...... ALL THE LINES STARTING WITH THE
C...... SAME FIRST LATITUDE
C...... THIS WILL PRODUCE A NEW SET OF LINES
C......
C...... Y(I)=Y(I+1)=Y(I+2)=Y(I+3)
C......
C...... W(I) > Y(I) AND W(I+1) > Y(I)
C......
C......
1517 KN = I
1518 IF (Y(KN).NE.Y(KN+1)) GO TO 1519
   KN = KN+1
   GO TO 1518
1519 LLSW = 0
   MI = I
   TR = 99.9
761 IF (TR.LT.W(I)) GO TO 762
   TR = W(I)
   KR = LA2(I)
762 IF (I.GE.KN) GO TO 1763
   I = I +1
   GO TO 761
1763 IF (TR.LT.Y(KN-H)) GO TO 763
   TR = Y(KN+1)
   KR = LA1(KN+1)
763 I = MI -1
764 I = I +1
   IF (TR.EQ.W(I)) GO TO 765
   TX = 0.0
   TY = TR
   TZ = Z(I)
   TW = W(I)
   TA = EQA(I)
   TR = EQB(I)
   TC = EQC(I)
   KX = 0
   KY = KR
   KZ = LN2(I)
   KW = LA2(I)
   LA2(I) = KR
   W(I) = TR
   IF (TA.EQ.0.0) GO TO 803
   TX = (TC-TB*TY)/TA
   Z(I) = TX
   KD = TX
   XKD = KD
   TX = TX-XKD
   KS = TX*3600.+5
   KM = KS/60
   KS = KS-(KM*60)
\[ L_2 = K_\times \]
\[ L_1 = Z(I) \]

GO TO 515

765 IF (TR.GT.w(I)) GO TO 802
IF (I.LT.KN) GO TO 764
I = I-1
GO TO 512

C........THIS ROUTINE FINDS THE LOCATION
C........IN THE TABLE FOR THE NEW LINE
C........THE TABLE MUST FIRST BE SHIFTED DOWN
C........BY ONE LINE TO MAKE ROOM FOR THE NEW LINE
C........

515 DO 530 L=1,NUMVER
IF (TY.LT.Y(L)) GO TO 531
IF (TY.GT.Y(L)) GO TO 530
IF (TX.GT.X(L)) GO TO 531
IF (TX.LT.X(L)) GO TO 530
IF (TW.GT.W(L)) GO TO 530
IF (TW.LT.W(L)) GO TO 531
IF (TZ.GT.Z(L)) GO TO 531
IF (TZ.EQ.Z(L)) GO TO 804
530 CONTINUE

531 NL = NUMVER +1
KL = NUMVER + 2
K = L
NUMVER = NUMVER + 1
IF (LAl(NL).NE.9999999) GO TO 805

532 X(KL) = X(NL)
Y(KL) = Y(NL)
Z(KL) = Z(NL)
W(KL) = W(NL)
LAl(KL) = LAl(NL)
LA2(KL) = LA2(NL)
LN1(KL) = LN1(NL)
LN2(KL) = LN2(NL)
EQA(KL) = EQA(NL)
EQB(KL) = EQB(NL)
EQC(KL) = EQC(NL)
KL = NL
NL = NL - 1
IF (NL.GE.K) GO TO 532
X(K) = TX
Y(K) = TY
Z(K) = TZ
W(K) = TW
EQA(K) = TA
EQB(K) = TB
EQC(K) = TC....
LN1(K) = KX
LN2(K) = KZ

B-10
LA1(K) = KY
LA2(K) = KW
IF (LLSW) 730,765,513
999 STOP

C...... ERROR STOPS IF INCORRECT CONDITIONS ENCOUNTERED
C......
800 WRITE(6,850)I
850 FORMAT(1X,"THE CURRENT LINES ARE NOT MATCHED",I5)
   GO TO 900

C...... THE FIRST LINE OF A PAIR SHOULD ALWAYS HAVE THE LOWER
C...... LATITUDE IN COMPARING THE SECOND POINTS
C......
801 WRITE(6,852)I
852 FORMAT(1X,"THE LATITUDES OF THE SECOND POINTS ARE OUT OF SEQUENCE"
   $,I5)
   GO TO 900

C......
C...... TA = 0.0 INDICATES A PARALLEL LINE WHICH SHOULD HAVE
C...... BEEN ELIMINATED EARLIER
C......
803 WRITE(6,853)
853 FORMAT(1X,"A PARALLEL LINE HAS BEEN DETECTED ")
   GO TO 900

C......
C...... WITH Y(I) = Y(I+1) AND W(I) > BOTH Y(I+2) AND
C...... W(I+1) THERE IS NO CONNECTION TO THE NEXT SLICE
C...... THIS INDICATES AN ERROR CONDITION
C......
802 WRITE(6,854)I
854 FORMAT(1X,"THERE IS A BREAK IN THE POLYGON",I5)
   GO TO 900

C......
C...... LINE IN TABLE HAS COMMON FIRST POINT TO LINE BEING
C...... INSERTED AND LATITUDE OF SECOND POINTS IS ALSO EQUAL
C......
804 WRITE(6,855)I,L,NL,KL,NUMVER
855 FORMAT(1X,"DUPLICATE LINE IN TABLE",$I5)
   GO TO 900

C......
C...... TABLE MOVE NOT BEING STARTED AT RIGHT LOCATION
C......
805 WRITE(6,856)NL,KL
856 FORMAT(1X,"LAST TABLE INDEX NOT CORRECT",$I5)
   GO TO 900

C......
C...... INVALID END OF PROGRAM IN ROUTINE 514
C...... FOUND Y(I+2) = 999.9990
C......
814 WRITE(6,864)I
864 FORMAT(1X,"INVALID END OF PROGRAM IN ROUTINE 514 ",I5)
   GO TO 900

B-11
INVALID END OF PROGRAM IN ROUTINE 534

FOUND Y(I+2) = 999.9990

834 WRITE(6,884)I
884 FORMAT(1X,"INVALID END OF PROGRAM IN ROUTINE 534 ",I5)
   GO TO 900

THIS IS THE CORRECT END TO THE POLYGON PROGRAM
THE TABLE THAT IS PRINTED AT THIS TIME IS A COPY
OF THE FILE THAT WILL BE USED TO IDENTIFY THE SITES
FOR THIS BASIN

898 WRITE(6,1900)NUMVER,I
1900 FORMAT(1X,"ERROR-OUT OF SEQUENCE",2I5)
   GO TO 900
899 WRITE(6,1900)NUMVER,I
900 WRITE(6,901)NUMVER,I
901 FORMAT(1X,"NORMAL JOB TERMINATION",2I5)
   IF (LONL.LT.0.0) LONL = -LONL
   IF (LONH.LT.0.0) LONH = -LONH
   WRITE(6,902)LATL,LONL,LATH,LONH,BCODE,BEXPL1,BEXPL2,
$BEXPL3,BEXPL4,BEXPL5
902 FORMAT(1X,2(I6,I7),A8,4A10,A6)
   WRITE(6,903)LATL,LONL,LATH,LONH,BCODE,BEXPL1,BEXPL2,
$BEXPL3,BEXPL4,BEXPL5
903 FORMAT(2(I6,I7),A8,4A10,A6)
   DO 910 J=1,NUMVER
   I=NUMVER+1-J
   IF (LN1(I).LT.0.0) LN1(I) = -LN1(I)
   IF (LN2(I).LT.0.0) LN2(I) = -LN2(I)
   KY = Y(I)*10000.+0.5
   KX = X(I)*10000.+0.5
   KW = W(I)*10000.+0.5
   KZ = Z(I)*10000.+0.5
   I1 = "+"
   I2 = "+"
   IF (KX.LT.0) I1 = "-"
   IF (KX.LT.0) KX = -KX
   IF (KZ.LT.0) I2 = "-"
   IF (KZ.LT.0) KZ = -KZ
   KA = EQA(I)*10000000. + 0.5
   KB = EQB(I)* 10000000. + 0.5
   KC = EQC(I) * 10000. +0.5
   I3 = "+"
   I4 = "+"
   I5 = "+"
   IF (KA .LT. 0) I3 = "-"
   IF (KA .LT. 0) KA = -KA
   IF (KB .LT. 0) I4 = "-"
   IF (KB .LT. 0) KB = -KB
   IF (KC .LT. 0) I5 = "-"
   IF (KC .LT. 0) KC = -KC

B-12
WRITE (6, 911) LA1(I), LN1(I), LA2(I), LN2(I), KY, I1, KX, KW, I2, KZ, 
SI3, KA, I4, KB, I5, KC, I
WRITE (16, 912) LA1(I), LN1(I), LA2(I), LN2(I), KY, I1, KX, KW, I2, KZ, 
SI3, KA, I4, KB, I5, KC, I
911 FORMAT (1X, 2(I6, I7), 2(I6, A1, I7), 2(A1, I8), A1, I7, I4)
912 FORMAT (2(I6, I7), 2(I6, A1, I7), 2(A1, I8), A1, I7, I4)
910 CONTINUE
IZERO = "000000"
WRITE (16, 913) IZERO
913 FORMAT (A6, 74X)
GO TO 999
END

B-13
IDENTIFICATION DIVISION.
PROGRAM-ID. EDIT-1.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. 6600;
OBJECT-COMPUTER. 6600.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT CARD-FILE ASSIGN TO "INPUT".
SELECT LIST-FILE ASSIGN TO "OUTPUT".
SELECT FILEA ASSIGN TO TAPEIN.
SELECT SLCFLE ASSIGN TO SLC.
SELECT ACHNG ASSIGN TO ACHNG.
SELECT RNCARD ASSIGN TO RNCARD.
SELECT PULLCG ASSIGN TO PULLCG.
DATA DIVISION.
FILE SECTION.
FD CARD-FILE
LABEL RECORD OMITTED
DATA RECORD IS CNL-CARD.
01 CNL-CARD.
  02 CNLXXX.
     03 ID-CNL PIC IS XXX.
     03 ID-NUM PIC 999.
  02 DELNUM REDEFINES CNLXXX PIC 9(6).
  02 VSNA PIC X.
  02 ID-DES PIC X(73).
FD LIST-FILE
LABEL RECORD OMITTED
DATA RECORD IS LIST-LINE.
01 LIST-LINE.
  03 CC PIC X.
  03 LLPRT PIC X(132).
FD FILEA
DATA RECORD IS RSNCRD
LABEL RECORDS ARE OMITTED.
01 RSNCRD.
  03 FILLER PIC X(4).
  03 IMAGE.
     05 KEYID.
        07 KEYLAT PIC 9(6).
        07 KEYLNG PIC 9(7).
        07 KEYDUP PIC 99.
     05 VFMTIN PIC X(61).
FD SLCFLE
DATA RECORD IS SLCREC
LABEL RECORDS ARE OMITTED.
01 SLCREC.
  03 LA1 PIC 9(6).
  03 LN1 PIC 9(7).
  03 LA2 PIC 9(6).
  03 LN2 PIC 9(7).
  03 DA1 PIC 99V9999.
  03 LS1 PIC X.
03 DN1 PIC 99999999.
03 DAP PIC 99999999.
03 LS2 PIC X.
03 DN2 PIC 99999999.
03 ASIGN PIC X.
03 AM PIC 9999999999.
03 BSTGN PIC X.
03 RM PIC 99999999.
03 CSIGN PIC X.
03 CK PIC 99999999.
03 SSEQ PIC 9999.
FD ACHNG
DATA RECORD IS AUDREC
LABEL RECORDS ARE OMITTED.
01 AUDREC PIC X(96).
FD PULLCG
DATA RECORD IS PULREC
LABEL RECORDS ARE OMITTED.
01 PULREC PIC X(96).
FD RNCARD
DATA RECORD IS RECOUT
LABEL RECORDS ARE OMITTED.
01 RECOUT.
03 RVSN PIC 9(5).
03 RVSNA PIC X.
03 DSKSTR.
05 DSEQ PIC 9(9) VALUE 0.
05 KEYID.
07 KEYLAT.
09 LATDEG PIC 99.
09 LATMIN PIC 99.
09 LATSEC PIC 99.
07 KEYLNG.
09 LNGDEG PIC 999.
09 LNGMIN PIC 99.
09 LNGSEC PIC 99.
07 KEYDUP PIC 99.
05 VFMTIN PIC X(61).
03 PRD2 PIC 9(5).
COMMON-STOREGE SECTION.
01 TDA1.
03 EDA1 USAGE COMP-2 OCCURS 10 TIMES.
01 TDN1.
03 EDN1 USAGE COMP-2 OCCURS 10 TIMES.
01 TDA2.
03 EDA2 USAGE COMP-2 OCCURS 10 TIMES.
01 TDN2.
03 EDN2 USAGE COMP-2 OCCURS 10 TIMES.
01 TAM.
03 EAM USAGE COMP-2 OCCURS 10 TIMES.
01 TBM.
03 EBM USAGE COMP-2 OCCURS 10 TIMES.
01 TCK.
03 ECK USAGE COMP-2 OCCURS 10 TIMES.
WORKING-STORAGE SECTION.
77 ACCOUNT PIC 9(5) VALUE 0.
77 ALT PIC X VALUE "T".
77 ALX PIC X VALUE "X".
77 ALY PIC X VALUE "Y".
77 ALZ PIC X VALUE "Z".
77 A1 USAGE COMP-2.
77 A2 USAGE COMP-2.
77 A3 USAGE COMP-2.
77 CNRECN PIC 9(6) VALUE ZER0ES.
77 COUNTA PICTURE 9(6) VALUE 0.
77 COUNTN PIC 9(6) VALUE 1.
77 ERMSG1 PIC X(27) VALUE " SITE ID OUT OF SEQUENCE ".
77 ERMSG2 PIC X(27) VALUE " GW & QW LOCAL IDS UNEQUAL ".
77 ERRCNT PIC 9(9) VALUE 0.
77 ERRSW PIC X VALUE "Y".
77 ESW PIC 9 VALUE 0.
77 FLECNT PIC 9(6).
77 FRSW PIC X VALUE "F".
77 G1 USAGE COMP-2.
77 G2 USAGE COMP-2.
77 G3 USAGE COMP-2.
77 HSW PIC X VALUE "T".
77 INCROS PIC 9(6) VALUE 0.
77 IND PIC 99.
77 JULD PIC 9(5).
77 KDE PIC 999.
77 LAT USAGE COMP-2.
77 LATREC PIC 9(6).
77 LID PIC X(20).
77 LNCNT PIC 99 VALUE 70.
77 LNG USAGE COMP-2.
77 LOWL PIC 999999 VALUE 999999.
77 MNS PIC X VALUE "-".
77 NCDE PIC 9.
77 NDATIN PIC 9(6) VALUE 0.
77 NDETLN PIC 9(6) VALUE 0.
77 NFLEIA PIC 9(6) VALUE 0.
77 NFLEOA PIC 9(6) VALUE 0.
77 NU1 PIC 9 VALUE 1.
77 NU2 PIC 9 VALUE 2.
77 NU10 PIC 99 VALUE 10.
77 PGCNT PIC 999 VALUE 0.
77 PULMSG PIC X(6) VALUE "PULLED".
77 SEQCHK PIC 9(6) VALUE 0.
77 STRKEY PIC 9(15) VALUE 0.
77 TFILEA PIC 9(6) VALUE 0.
77 TFILEB PIC 9(6) VALUE 0.
77 TFILEC PIC 9(6) VALUE 0.
77 WRTSW PIC 99 VALUE 0.
77 X PIC 99.
77 XCODE USAGE COMP-2.
77 X6 PIC X(6) VALUE "XXXXXX".

B-16
01 WKDATE.
  03 YY PIC 99.
  03 MM PIC 99.
  03 DY PIC 99.
01 NAME-MONTH.
  03 JAN PIC XXXX VALUE " JAN".
  03 FEB PIC XXXX VALUE " FEB".
  03 MAR PIC XXXX VALUE " MAR".
  03 APR PIC XXXX VALUE " APR".
  03 MAY PIC XXXX VALUE " MAY".
  03 JUNE PIC XXXX VALUE " JUNE".
  03 JULY PIC XXXX VALUE " JULY".
  03 AUG PIC XXXX VALUE " AUG".
  03 SEPT PIC XXXX VALUE " SEPT".
  03 OCT PIC XXXX VALUE " OCT".
  03 NOV PIC XXXX VALUE " NOV".
  03 DEC PIC XXXX VALUE " DEC".
01 MONTH-TABLE REDEFINES NAME-MONTH.
  03 MONTH PIC XXXX OCCURS 12 TIMES INDEXED BY NMM.
01 JOBCNL.
  03 CC PIC X VALUE "1".
  03 PCNL.
    05 CNTYPE PIC XXX.
    05 GO-CNL PIC 999.
    05 PCNLX PIC X(74).
  03 FILLER PIC X(52) VALUE SPACES.
01 JBCNXLX.
  03 CC PIC X VALUE "0".
  03 PSTATS PIC X(14) VALUE "RECORD COUNTS".
  03 FILLER PIC X(6) VALUE SPACES.
  03 XCARD PIC X(8) VALUE "CARDS IN".
  03 CSTAT PIC ZZZZZ9.
  03 FILLER PIC X(8) VALUE SPACES.
  03 XDRLNS PIC X(12) VALUE "DETAIL LINES".
  03 DRLNS PIC ZZZZZ9.
  03 FILLER PIC X(8) VALUE SPACES.
  03 XDATIZN PIC X(7) VALUE "DATA IN".
  03 DATIN PIC ZZZZZ9.
  03 FILLER PIC X(8) VALUE SPACES.
  03 XFAIN PIC X(11) VALUE "OLD-DATA I".
  03 FADATA PIC ZZZZZ9.
  03 FILLER PIC X(8) VALUE SPACES.
  03 XNOUT PIC X(12) VALUE "FMT DATA OUT".
  03 FOUT PIC ZZZZZ9.
01 HDR1.
  03 CC1 PIC 9 VALUE 1.
  03 FILLER PIC X.
  03 LDATE PIC X(6) VALUE "DATE: ".
  03 RMTH PIC X(4).
  03 FILLER PIC X.
  03 RDY PIC Z9.
  03 XB PIC XXXX VALUE ", 19".
  03 RYR PIC XX.
  03 FILLER PIC X(21).
03 TITLEA PIC X(50).
03 FILLER PIC X(21).
03 PGE PIC X(4) VALUE "PAGE".
03 PNUM PIC ZZZ9.
03 FILLER PIC X(2).

01 HDR2.
03 CC PIC 9 VALUE 0.
03 USGS PIC X(7) VALUE " USGS: ".
03 GSNAME PIC X(20).
03 FILLER PIC XXX.
03 BASINL PIC X(7) VALUE "SOURCES".
03 FILLER PIC X.
03 SRCEID PIC X(50).
03 SRCNM PIC X(30).
03 CNLZZZ PIC X(6).
03 FILLER PIC X(8).

01 HDR3C3.
03 CC PIC X VALUE "0".
03 FILLER PIC X(8) VALUE " NUMBER".
03 FILLER PIC XXX.
03 FILLER PIC XXX VALUE "VSN".
03 FILLER PIC X(29).
03 FILLER PIC X(16) VALUE "ORIGINAL CHANGE ".
03 FILLER PIC X(17) VALUE "RECORD BETWEEN **".
03 FILLER PIC X(40).
03 FILLER PIC X(4) VALUE "DATE".
03 FILLER PIC X(12).

01 PNEWRC.
03 CC PIC X VALUE SPACE.
03 DSEQ PIC Z(9).
03 FILLER PIC X.
03 RVSN PIC 9(5).
03 RVNSA PIC X.
03 DSKSTR.
 05 FILLER PIC XXXX VALUE " **".
 05 FILLER PIC XXXX.
 05 KEYID.
   07 KEYLAT.
     09 LATDEG PIC 99.
     09 LATMIN PIC 99.
     09 LATSEC PIC 99.
   07 KEYLNG.
     09 LMGDEG PIC 999.
     09 LGMIN PIC 99.
     09 LNGSEC PIC 99.
   07 KEYDUP PIC 99.
 05 VFMTIN PIC X(61).
03 FILLER PIC XX VALUE "**".
03 FILLER PIC X(13).
03 PRO2 PIC 9(5).
03 FILLER PIC X(12).
PROCEDURE DIVISION.

NOTE-A.

NOTE THE PROGRAM STARTS WITH THIS INITIALIZATION ROUTINE THAT OPENS THE INPUT AND LISTING FILES AND THEN THE DATE STORED IN THE SYSTEM IS READ AND USED TO SET UP THE DATE IN THE FIRST HEADING LINE. NO RETURN FROM THE MAINLINE ROUTINE IS MADE TO THIS PROCEDURE.
BEGIN.
OPEN INPUT CARD-FILE.
OPEN OUTPUT LIST-FILE.
ACCEPT WKDATE FROM DATE.
ACCEPT JULD FROM DAY.
SET NMM TO MM.
MOVE MONTHLY (NMM) TO RMTH OF HDR1.
MOVE DY TO RYD OF HDR1.
MOVE YY TO RYR OF HDR1.
NOTE-B.
NOTE THIS ROUTINE READS THE CONTROL CARDS AND MOVES
THE GENERAL DESCRIPTION OF THE REPORT TO THE
HEADING AREA. THEN A BRANCH IS TAKEN TO THE
APPROPRIATE STARTING ROUTINE DEPENDING ON THE
NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD.
THERE IS NO RETURN TO THIS ROUTINE.
READC.
READ CARD-FILE AT END GO TO STOP1.
IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
ADD 1 TO INCROS.
MOVE CNL-CARD TO PCNL.
MOVE CNLXXX TO CNLZZZ.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 010 MOVE TITLEX OF DATAA
TO TITLEA OF HDR1 ELSE GO TO STOP4.
ADD 1 TO INCROS.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 011 MOVE GNAME TO GSNAME
MOVE CNAME TO SRCNAM ELSE GO TO STOP4.
ADD 1 TO INCROS.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 012 MOVE SRCAGC TO SRCEID,
MOVE GO-CNL TO CNLZZZ ELSE GO TO STOP4.
ADD 1 TO INCROS.
IF GO-CNL = 001, GO TO STARTA.
IF GO-CNL = 002, GO TO STARTB.
IF GO-CNL = 003, GO TO STARTC, ELSE GO TO STOP2.
NOTE-C.
NOTE NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE
THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING
ROUTINES.
NOTE-CNLI.
NOTE CONTROL 1 PROCESSING TO READ THE ORIGINAL DISTRICT
UPDATE FILE AND/OR DELETE SELECTED RECORDS THAT FAIL VISUAL
VERIFICATION OF LISTING. THIS MAY BE RUN SEVERAL TIMES TO
GET ALL JUNK RECORDS OUT OF THE INPUT FILE.
STARTA.
READ CARD-FILE AT END DISPLAY
"***70UM1 C1 INPUT STREAM EOF *VSN*", STOP RUN.
MOVE JULD TO PRD2 OF RECOUT. MOVE DELNUM TO RVSN OF RECOUT.
MOVE VSNA OF CNL-CARD TO RVSN OF RECOUT.
READ CARD-FILE AT END DISPLAY
"***701M1 C1 INPUT STREAM EOF *FILES?* " , STOP RUN.
MOVE DELNUM TO FLECNT.
READCD.
READ CARD-FILE AT END GO TO CLEAC.
IF DELNUM > SEQCHK, MOVE DELNUM TO SEQCHK ELSE GO TO STUP10.
READTP.
IF FRSW = "F", PERFORM STARAL.
READ FILEA AT END GO TO CLOSEA.
READTP1.
ADD 1 TO TFILEA, CNRECN, NFLEIA.
MOVE CNRECN TO CNRECP OF PRNTLN.
IF DELNUM = CNRECN,
MOVE "DELETED RECORD" TO NOTEXS OF PRNTLN ELSE
ADD 1 TO ACOUNT, MOVE "STORED RECORD" TO NOTEXS OF PRNTLN.
MOVE IMAGE OF RSNCRD TO CIMAGE OF PRNTLN.
MOVE KEYID OF RSNCRD TO KEYID OF DSKSTR OF RECOUT.
MOVE VFMTIN OF RSNCRD TO VFMTIN OF DSKSTR OF RECOUT.
MOVE CNRECN TO DSEQ OF RECOUT, DSEQ OF PRNTLN.
MOVE JULD TO RDATE OF PRNTLN, PRD2 OF RECOUT.
MOVE RVSN OF RECOUT TO RVSN OF PRNTLN.
MOVE RVSN OF RECOUT TO VSNA OF PRNTLN.
PERFORM PRTSAL. IF DELNUM NOT = CNRECN,
WRITE RECOUT, ADD 1 TO NFLEOA.
IF DELNUM > CNRECN, GO TO READTP,
ELSE GO TO READCD.
STARAL.
OPEN INPUT FILEA. OPEN OUTPUT RNCARD.
MOVE "X" TO FRWS.
CLOSEA.
ADD 1 TO COUNTA, COUNTN.
IF FLECNT = 1, GO TO CLEAC1,
ELSE SUBTRACT 1 FROM FLECNT.
PERFORM HDPRNT.
DISPLAY "END OF FILE ", COUNTA.
DISPLAY "START FILE ", COUNTN.
READ FILEA AT END GO TO EROR07.
GO TO READTP1.
CLEAC1.
CLOSE RNCARD, FILEA. GO TO STOP3.
CLEASE.
DISPLAY "999999 MISSING ON END OF DELETE FILE".
MOVE 999999 TO DELNUM.
GO TO READTP.
NOTE CONTROL 2 PROCESSING TO FIND IF SITE ID IS WITHIN
BOUNDARY OF USGS DISTRICT AS DEFINED BY POLYGON SLICE FILE.

STARTB.
OPEN INPUT SLCFLE, RNCARD. OPEN OUTPUT ACHNG.
READ SLCFLE AT END GO TO STOP21.
READ SLCFLE AT END GO TO STOP21.

START1.
READ RNCARD AT END GO TO CLSEB.
ADD NU1 TO NDATIN.
IF KEYID OF RECOUT = STRKEY, GO TO WDISK.
MOVE 0 TO WRTSW. MOVE KEYID OF RECOUT TO STRKEY.
MOVE NU2 TO XCODE. MOVE LATDEG OF RECOUT TO A1.
MOVE LATMIN OF RECOUT TO A2. MOVE LATSEC OF RECOUT TO A3.
MOVE LNGDEG OF RECOUT TO G1. MOVE LNGMIN OF RECOUT TO G2.
MOVE LNGSEC OF RECOUT TO G3. MOVE ZEROS TO LAT, LNG.
ENTER DMSDEC USING A1, A2, A3, G1, G2, G3, LAT, LNG, XCODE.

STAR2.
MOVE KEYLAT OF RECOUT TO LATREC.
IF LATREC < LOWL, GO TO STRS. MOVE ZEROS TO XCODE.
ENTER DSTEDT USING LAT, LNG, XCODE.
IF XCODE NOT = ZERO, MOVE NU1 TO WRTSW.

WDISK.
IF WRTSW = 0, MOVE JULD TO PRO2 OF RECOUT, WRITE AUDREC FROM
RECOUT, ADD NU1 TO NFLEOA, ELSE MOVE RECOUT TO RECPRT,
WRITE LIST-LINE FROM PRNTLN, ADD NU1 TO NDETLN.
GO TO START1.

STARS.
MOVE ZERO TO X.

STARS1.
EXAMINE LA1 OF SLCREC REPLACING ALL SPACES BY ZERO.
EXAMINE LN1 OF SLCREC REPLACING ALL SPACES BY ZERO.
EXAMINE LA2 OF SLCREC REPLACING ALL SPACES BY ZERO.
EXAMINE LN2 OF SLCREC REPLACING ALL SPACES BY ZERO.
EXAMINE DA1 OF SLCREC REPLACING ALL SPACES BY ZERO.
EXAMINE DN1 OF SLCREC REPLACING ALL SPACES BY ZERO.
EXAMINE DA2 OF SLCREC REPLACING ALL SPACES BY ZERO.
EXAMINE DN2 OF SLCREC REPLACING ALL SPACES BY ZERO.
EXAMINE AM OF SLCREC REPLACING ALL SPACES BY ZERO.
EXAMINE BM OF SLCREC REPLACING ALL SPACES BY ZERO.
EXAMINE CK OF SLCREC REPLACING ALL SPACES BY ZERO.
IF LATREC < LA1 OF SLCREC, GO TO STRS2.
IF X = ZERO, MOVE LA1 OF SLCREC TO LOWL.
ADD NU1 TO X. IF X > NU10, GO TO STP903.
IF LA1 OF SLCREC NOT = LOWL, GO TO STAR82.
IF LS1 = MNS, COMPUTE EDN1 (X) = - DN1,
ELSE COMPUTE EDN1 (X) = + DN1.

B-22
IF LS2 = MN3, COMPUTE EUN2 (X) = - DN2.
ELSE COMPUTE EUN2 (X) = + DN2.
IF ASIGN = MNS, COMPUTE EAM (X) = - AM.
ELSE COMPUTE EAM (X) = + AM.
IF GSIGN = MNS, COMPUTE EBM (X) = - BM.
ELSE COMPUTE EBM (X) = + BM.
IF CSIGN = MNS, COMPUTE ECK (X) = - CK.
ELSE COMPUTE ECK (X) = + CK.
COMPUTE EDA1 (X) = + DA1.
COMPUTE EDA2 (X) = + DA2.

STRA.
READ SLCFILE AT END GO TO STOP22. GO TO STRSA.
ELSE,
GO TO STOP3.
STOP RUN.

STARTC.
READ CARD-FILE INTO RANGEL AT END DISPLAY
"***760M1 NO INPUT STREAM FILE", STOP RUN.
IF NUMFST = 999999, MOVE NU1 TO WRTSW, OPEN OUTPUT ACHNG,
OPEN OUTPUT PULLCG, ELSE MOVE ZERO TO WRTSW.
OPEN INPUT RNCARD, MOVE ALT TO HSW.
MOVE NU1 TO CNRECN, NOATIN.
READ RNCARD AT END DISPLAY
"***761M1 DATA CHANGE FILE, RNCCHNG, EMPTY" STOP RUN.

STARTC.
READ CARD-FILE INTO RANGEL AT END DISPLAY
"***762M1 END OF RUN" GO TO STOP3.
EXAMINE RANGEL REPLACING ALL " " BY ZERO.
IF NUMFST NOT = ZERO, MOVE ALX TO HSW, GO TO X3.
IF NUMSTR NOT = ZERO, MOVE ALY TO HSW, GO TO X3.
IF KEYIDS NOT = ZERO, MOVE ALZ TO HSW.

X3.
IF HSW = ALT, DISPLAY
"***763M1 RANGE INCORRECT" GO TO STARTC.
IF HSW = ALX, ADD NUMSTR, NUMLST, KEYIDS, KEYIDE
GIVING KEYIDE IF KEYIDE NOT = 0 DISPLAY
"***764M1 MULTIPLE RANGE PARAMETERS" GO TO STARTC.
IF HSW = ALY, ADD NUMFST, KEYIDS, KEYIDE GIVING
KEYIDE IF KEYIDE NOT = ZERO, DISPLAY
"***765M1 SEQUENCE COUNT WITH OTHER VALUES", GO TO STARTC.
IF HSW = ALZ, ADD NUMFST, NUMSTR, NUMLST GIVING
NUMLST, IF NUMLST NOT = ZERO, DISPLAY
"***766M1 SITE ID RANGE WITH OTHER VALUES", GO TO STARTC.
IF HSW = ALY AND NUMLST < NUMSTR, DISPLAY
"***767M1 SEQUENCE COUNT INVALID END", GO TO STARTC.
IF HSW = ALZ AND KEYIDS > KEYIDS, DISPLAY
"***768M1 SITE ID RANGE INVALID END HIGH", GO TO STARTC.
IF HSW = ALY AND NUMFST < CNRECN, DISPLAY
"***769M1 NO RECORDS PRINTED NUMFST = ",NUMFST,
" CNRECN = ", CNRECN, GO TO STARTC.

STARTC.
IF WRTSW = NU1, WRITE AUDREC FROM RECOUT, ADD NU1 TO NFLEOA.
IF HSW = ALX AND CNRECN NOT > NUMFST, GO TO PRNOUT.
IF HSW = ALY AND CNRECN NOT < NUMSTR AND CNRECN NOT > NUMLST
GO TO PRNOUT.
IF HSw = ALZ AND KEYID OF RECOUT NOT > KEYIDS AND KEYID OF RECOUT NOT < KEYID, GO TO PRNUUT.

STARC3.
READ RNCARD AT END DISPLAY
"****710M1 END OF DATA REQUESTS INCOMPLETE" GO TO STOP3.
ADD NUI TO CNRECN, NDATIN.
IF HSw = ALX AND CNRECN > NUMFST GO TO STARC1.
IF HSw = AY AND CNRECN > NUMLST GO TO STARC1.
IF HSw = ALZ AND KEYID OF RECOUT < KEYID, GO TO STARC1.
GO TO STARC2.

PRNUUT.
IF LNCNT > 60, PERFORM HDPRNT, PERFORM HDCN3.
ADD NUI TO LNCNT, ND'tLN.
MOVE CORRESPONDING RECOUT TO PNEWR.
MOVE DSEQ OF RECOUT TO DSEQ OF PNEWR.
WRITE LIST-LINE FROM PNEWR.
IF WRTSW = NUI, WRITE PULREC FROM RECOUT.
GO TO STARC3.

HDCN3.
WRITE LIST-LINE FROM HDR3C3.
HDPRNT.
ADD 1 TO PGCNT.
MOVE PGCNT TO PNUM OF HDR1.
MOVE 4 TO LNCNT.
WRITE LIST-LINE FROM HDR1.
WRITE LIST-LINE FROM HDR2.

PRTSNL.
IF LNCNT > 60, PERFORM HDPRNT,
ELSE ADD 1 TO LNCNT, ND'TLN.
WRITE LIST-LINE FROM PRNTLN.
MOVE " " TO CC OF PRNTLN. ADD 1 TO ND'TLN.

STOP1.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "****710M1 CONTROL INPUT FILE EMPTY".
STOP RUN.

STOP2.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
STOP RUN.

STOP3.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY TFILB.
MOVE INCROS TO CSTAT.
MOVE NDATIN TO DATAIN.
MOVE NFLIA TO FA DATA.
MOVE ND'TLN TO DRLNS.
MOVE NFL'OA TO FOUT.
WRITE LIST-LINE FROM JBCNLX.
DISPLAY "NORMAL JOB TERMINATION".
CLOSE CARD-FILE, LIST-FILE.
STOP RUN.

STOP4.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "SOURCE ID RECORDS MISSING".
STOP RUN.
STOP5.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "DATA INPUT FILE END INCORRECT".
DISPLAY "EOF ACCURRED AT READY OR READY".
STOP RUN.

STOP6.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "SECOND CARD OF PAIR MISSING".
STOP RUN.

STOP7.
DISPLAY "VSN AND/OR DATE RECORD MISSING INPUT EMPTY".
DISPLAY "STOP7". STOP RUN.

STOP9.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "RECORD TYPE INVALID".
STOP RUN.

STOP10.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "ERROR IN RECORD SEQUENCE", "STOP10".
DISPLAY SEQCHK, " ", DELNUM.
STOP RUN.

STOP21.
DISPLAY "****921MU1-C2 SLICE FILE EMPTY".
STOP RUN.

STOP22.
DISPLAY "****922MU1-C2 CHANGE PAST END OF SLICE FILE".
STOP RUN.

STP903.
DISPLAY "****903M1 SLICE TABLE OVERFLOW". STOP RUN.

STP904.
DISPLAY "****904M1 EOF ON SLICE FILE CHANGES INVALID".
GO TO STOP3.
IDENTIFICATION DIVISION.
PROGRAM-ID. EDIT-2.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. 6600.
OBJECT-COMPUTER. 6600.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
  SELECT CARD-FILE ASSIGN TO "INPUT".
  SELECT LIST-FILE ASSIGN TO "OUTPUT".
  SELECT FILEA ASSIGN TO DELFILE.
  SELECT RNCARD ASSIGN TO ADCHNG.
  SELECT CHNGEL ASSIGN TO SRCHNG.
DATA DIVISION.
FILE SECTION.
FD CARD-FILE
  LABEL RECORD OMITTED
  DATA RECORD IS CNL-CARD.
  01 CNL-CARD.
     02 CNLXXX.
        03 ID-CN1 PIC IS XXX.
        03 ID-NUM PIC 999.
     02 DELENUM REDEFINES CNLXXX PIC 9(6).
     02 VSNA PTC X.
     02 ID-DES PIC X(73).
FD LIST-FILE
  LABEL RECORD OMITTED
  DATA RECORD IS LIST-LINE.
  01 LIST-LINE.
     03 CC PIC X.
     03 LLPRT PIC X(132).
FD FILEA
  DATA RECORD IS ERRREC
  LABEL RECORDS ARE OMITTED.
  01 ERRREC.
     03 CNREC1 PIC 9(6).
     03 KSEQ PIC 9(4).
     03 DSKSTR.
        05 DSEQ PIC 9(4).
     05 KEYID.
        07 KEYLAT.
           09 LATDEG PIC 99.
           09 LATMIN PIC 99.
           09 LATSEC PIC 99.
        07 KEYLNG.
           09 LNGDEG PIC 999.
           09 LNGMTN PIC 99.
           09 LNGSEC PIC 99.
        07 KEYDUP PIC 99.
     03 VFMTIN PIC X(61).
     03 SNDATE.
        05 VSNTA PIC 9999.
        05 VSNA PIC X.
        05 PRDATE PIC 9(5).
FD RNCARD
DATA RECORD IS RECOUT
LABEL RECORDS ARE OMITTED.
01 RECOUT.
  03 RVSN PIC 4(5).
  03 RVSNA PIC X.
  03 DSKSTR.
    05 DSEQ PIC 9(9) VALUE 0.
    05 KEYID.
      07 KEYLAT.
        09 LATDEG PIC 99.
        09 LATMIN PIC 99.
        09 LATSEC PIC 99.
      07 KEYLNG.
        09 LNGDEG PIC 999.
        09 LNGMIN PIC 99.
        09 LNGSEC PIC 99.
      07 KEYDUP PIC 99.
    05 VFMTIN PIC X(61).
  03 PRD2 PIC 9(5).
FD CHNGEL
DATA RECORD IS CNGREC
LABEL RECORDS ARE OMITTED.
01 CNGREC.
  03 VSN PIC 99999.
  03 VSNA PIC X.
  03 PRD2 PIC 9(5).
  03 KSEQ PIC 9(4).
  03 KEYID.
    05 KEYLAT.
      07 LATDEG PIC 99.
      07 LATMIN PIC 99.
      07 LATSEC PIC 99.
    05 KEYLNG.
      07 LNGDEG PIC 999.
      07 LNGMIN PIC 99.
      07 LNGSEC PIC 99.
    05 KEYDUP PIC 99.
  03 CSCDE PIC X.
  03 RNUM PIC 999.
  03 CNUM PIC 999.
  03 CCR PIC X.
  03 CCSR PIC X.
  03 TYPDAT PIC 99.
  03 DLEN PIC 99.
  03 NEWDAT PIC X(45).
  03 NSEQ PIC 9(8).
WORKING-STORAGE SECTION.
77 ACOUNT PIC 9(6) VALUE 0.
77 ALD PIC X VALUE "D".
77 ALFLAT PIC X(6).
77 ALFLNG PIC X(7).
77 ALPHA6 PIC X(6).
77 ALPHA7 PIC X(7).
77 ALR PIC X VALUE "R".

B-27
77 ALT PIC X VALUE "T".
77 ALX PIC X VALUE "X".
77 ALY PIC X VALUE "Y".
77 ALZ PIC X VALUE "Z".
77 BT PIC 999 VALUE 0.
77 BTA PIC 9(4).
77 RLK1 PIC X VALUE SPACE.
77 CMPLAT PIC 9(6).
77 CMPLNG PIC 9(7).
77 CNREC PIC 9(6) VALUE ZERODE.
77 DELSW PIC X VALUE "Z".
77 DEM1 PIC X(15) VALUE " DATA ENTRY ".
77 DEM2 PIC X(10) VALUE " UNCHECKED ".
77 DYZ PIC 9.
77 ERMG1 PIC X(27) VALUE " AZ FORMAT SORTED CHANGES ".
77 ERMG2 PIC X(27) VALUE " GW & GW LOCAL IDS UNEQUAL ".
77 ERRCT PIC 9(9) VALUE 0.
77 ERSW PIC X VALUE "Y".
77 ESW PIC 99 VALUE 0.
77 FLECNT PIC 9(6).
77 FRSW PIC X VALUE "F".
77 HSW PIC X VALUE "X".
77 INCRDS PIC 9(6) VALUE 0.
77 JULD PIC 9(5).
77 LNCT PIC 99 VALUE 70.
77 MSG1 PIC X(25) VALUE " CHECKS OK DROP NOT NEEDED ".
77 MSG2 PIC X(25) VALUE " KEYID NOT = LAT-LNG FLDS ".
77 NDATIN PIC 9(6) VALUE 0.
77 NDLTLN PIC 9(6) VALUE 0.
77 NFLEIA PIC 9(6) VALUE 0.
77 NFLEOA PIC 9(6) VALUE 0.
77 NUL PIC 9 VALUE 1.
77 PGCT PIC 999 VALUE 0.
77 SEQCHK PIC 9(6) VALUE 0.
77 TFILEA PIC 9(6) VALUE 0.
77 TFILEB PIC 9(6) VALUE 0.
77 TFILEC PIC 9(6) VALUE 0.
01 WKDATE.
  03 YY PIC 99.
  03 MM PIC 99.
  03 DD PIC 99.
01 NAME-MONTH.
  03 JAN PIC XXXX VALUE " JAN ".
  03 FEB PIC XXXX VALUE " FEB ".
  03 MAR PIC XXXX VALUE " MAR ".
  03 APR PIC XXXX VALUE " APR ".
  03 MAY PIC XXXX VALUE " MAY ".
  03 JUNE PIC XXXX VALUE " JUNE ".
  03 JULY PIC XXXX VALUE " JULY ".
  03 AUG PIC XXXX VALUE " AUG ".
  03 SEPT PIC XXXX VALUE " SEPT ".
  03 OCT PIC XXXX VALUE " OCT ".
  03 NOV PIC XXXX VALUE " NOV ".
  03 DEC PIC XXXX VALUE " DEC ".

B-28
01 MONTH-TABLE REDEFINES NAME=MONTH.
03 MUNTH PIC XXXX OCCURS 12 TIMES INDEXED BY NMM.
01 JOBCNL.
  03 CC PIC X VALUE "1".
  03 PCNL.
    05 CNTYPE PIC XXX.
    05 GOTO-CN1 PIC 999.
    05 PCNLX PIC X(74).
    03 FILLER PIC X(52) VALUE SPACES.
01 JBCNLX.
  03 CC PIC X VALUE "0".
  03 PSTATS PIC X(14) VALUE "RECURD COUNTS".
  03 FILLER PIC X(6) VALUE SPACES.
  03 XCARD PIC X(8) VALUE "CARDS IN".
  03 CSTAT PIC ZZZZ9.
  03 FILLER PIC X(6) VALUE SPACES.
  03 XORLNS PIC X(12) VALUE "DETAIL LINES".
  03 DRLNS PIC ZZZZ9.
  03 FILLER PIC X(8) VALUE SPACES.
  03 XDATIZN PIC X(7) VALUE "DATA IN".
  03 DATAIN PIC ZZZZ9.
  03 FILLER PIC X(8) VALUE SPACES.
  03 XFAIN PIC X(11) VALUE "ULO-DATA I".
  03 FADATA PIC ZZZZ9.
  03 FILLER PIC X(8) VALUE SPACES.
  03 XNOT PIC X(12) VALUE "FMT DATA OUT".
  03 FOUT PIC ZZZZ9.
01 HDR1.
  03 CC1 PIC 9 VALUE 1.
  03 FILLER PIC X.
  03 LOAD PIC X(6) VALUE "DATE: ".
  03 RMTH PIC X(4).
  03 FILLER PIC X.
  03 RDY PIC Z9.
  03 X8 PIC XXXX VALUE ", 19".
  03 RYR PIC XX.
  03 FILLER PIC X(21).
  03 TITLE PIC X(60).
  03 FILLER PIC X(21).
  03 PGE PIC X(4) VALUE "PAGE".
  03 PNUM PIC ZZZ9.
  03 FILLER PIC X(2).
01 HDR2.
  03 CC PIC 9 VALUE 0.
  03 USGS PIC X(7) VALUE " USGS: ".
  03 GSNAME PIC X(20).
  03 FILLER PIC XXX.
  03 BASINL PIC X(7) VALUE "SOURCES".
  03 FILLER PIC X.
  03 SRCED PIC X(50).
  03 SRGNM PIC X(30).
  03 CNLZZZ PIC X(6).
  03 FILLER PIC X(8).
01 HDR3.
03 CC PIC 9 VALUE 0.
03 FILLER PIC X(26).
03 ERRMSG PIC X(27) VALUE " ERRMSG NOT INITIALIZED ".
03 FILLER PIC X(45).
03 FILLER PIC X(18) VALUE "CURRECTION NOTES".
03 FILLER PIC X(16).
01 PRNTLN.
03 CC PIC X VALUE SPACE.
03 CNRTPCP PIC ZZZZZZZZ9.
03 FILLER PIC XXXX VALUE " ** ".
03 FILLER PIC X(4) VALUE "****".
03 CIMAGE PIC X(96).
03 FILLER PIC X(4) VALUE "****".
03 NOTESX PIC X(11).
01 CCARD.
03 CROSEQ PIC 9999.
03 KEYID.
 05 KEYLAT PIC 9(6).
 05 KEYLNG PIC 9(7).
 05 KEYDUP PIC 99.
03 RECTYP PIC X(4).
03 FILLER PIC X.
03 TRANSI PIC X(4).
03 FILLER PIC X.
03 CNUMBR PIC 99.
03 FILLER PIC XX.
03 GLCWLD.
 05 POSITION-1 PIC X.
 05 POSITION-2 PIC X.
 05 FILLER PIC X(18).
03 FILLER PIC X.
03 GLCWLD PIC X(20).
03 FILLER PIC X(4).
03 FILETY PIC XX.
01 DATAA.
03 ID-CNL PIC IS XXX.
03 ID-NUM PIC IS 999.
03 TITLEX.
 05 SRCAGC.
     07 GNAME PIC X(20).
     07 CNAME PIC X(30).
 05 FILLER PIC X(10).
03 FILLER PIC X(14).
01 FMTSLD.
03 CHRSLSD PIC X OCCURS 61 TIMES INDEXED BY SC.
01 KEYIOL.
03 KEYLAT.
 05 LATDEG PIC 99 VALUE 99.
 05 LATMIN PIC 99 VALUE 99.
 05 LATSEC PIC 99 VALUE 99.
03 KEYLNG.
 05 LNGDEG PIC 999 VALUE 999.
 05 LNGMIN PIC 99 VALUE 99.
 05 LNGSEC PIC 99 VALUE 99.
03 KEYDUP PIC 99 VALUE 99.
PROCEDURE DIVISION.

NOTE-A.

NOTE THE PROGRAM STARTS WITH THIS INITIALIZATION ROUTINE THAT OPENS THE INPUT AND LISTING FILES AND THEN THE DATE STORED IN THE SYSTEM IS READ AND USED TO SET UP THE DATE IN THE FIRST HEADING LINE. NO RETURN FROM THE MAINLINE ROUTINE IS MADE TO THIS PROCEDURE.

B-31
BEGIN.
OPEN INPUT CARD-FILE.
OPEN OUTPUT LIST-FILE.
ACCEPT WKDATE FROM DATE.
ACCEPT JULD FROM DAY.
SET NMM TO MM.
MOVE MONTLY (NMM) TO RMTH OF HDR1.
MOVE DY TO RY OF HDR1.
MOVE YY TO RYR OF HDR1.

NOTE-B.
NOTE THIS ROUTINE READS THE CONTROL CARDS AND MOVES THE GENERAL DESCRIPTION OF THE REPORT TO THE HEADING AREA. THEN A BRANCH IS TAKEN TO THE APPROPRIATE STARTING ROUTINE DEPENDING ON THE NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD. THERE IS NO RETURN TO THIS ROUTINE.

READC.
READ CARD-FILE AT END GO TO STOP1.
IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
ADD 1 TO INCRDS.
MOVE CNL-CARD TO PCNL.
MOVE CNLXXX TO CNLZZZ.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 010 MOVE TITLX OF DATAA TO TITLEA OF HDR1 ELSE GO TO STOP4.
ADD 1 TO INCRDS.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 011 MOVE GNAME TO GSNAME
MOVE CNAME TO SRCNAM ELSE GO TO STOP4.
ADD 1 TO INCRDS.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 012 MOVE SRCAGC TO SRCEDC, MOVE GO-CNL TO CNLZZZ ELSE GO TO STOP4.
ADD 1 TO INCRDS.
IF GO-CNL = 001, GO TO STARTA.
IF GO-CNL = 002, GO TO STARTB.
IF GO-CNL = 003, GO TO STARTC, ELSE GO TO STOP2.

NOTE-C.
NOTE NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING ROUTINES.

STARTA.
OPEN INPUT RNCARD. OPEN OUTPUT CHNGEL, FILEA.
MOVE ERMSG1 TO ERRMSG OF HDR3. MOVE ZEROS TO NSEQ OF CNGREC.

STARTA1.
READ RNCARD AT END CLOSE FILEA RNCARD CHNGEL, DISPLAY '"***700M2 END OF INPUT DATA", GO TO STOP3.
SET SC TO 0. ADD NU1 TO NOATIN, TFILEB, NFLEOA.
IF KEYID OF RECOUT > KEYIDL, DISPLAY '"***701M2 INPUT FILE OUT OF SEQUENCE", KEYID OF RECOUT,
" > ", KEYIDL GO TO STOP3.

B-32
MOVE VERBATIN OF RECOUT TO FMTSLD.
IF KEYIO OF RECOUT < KEYIDL, GO TO EDFRST.
PERFORM SRCHNB. IF CHRSLD (SC) = ALR,
SET SC DOWN BY NUI, GO TO EDFRST.
SET SC DOWN BY NUI. IF HSW = ALA, GO TO WRTERR.
IF HSW = "A", GO TO EDR1.
IF HSW = "R", GO TO EDR2.
IF HSW = "C", GO TO RNUM23.
IF HSW = ALD, GO TO TRA1.
IF HSW = "E", GO TO TRA2.
IF HSW = "F", GO TO TRA3.
IF HSW = "G", GO TO TRA4.
IF HSW = "H", GO TO CTRANS.
IF HSW = "I", GO TO CNUM23.
IF HSW = "J", GO TO CdATA.
IF HSW = "K", SET SC TO 1, GO TO PC12A.
IF HSW = "L", GO TO LCMVE.
IF HSW = "M", GO TO CDAT1.
DISPLAY "SWITCH CODE ERROR HSW = ", HSW.
STOP RUN.
EDFRST.
MOVE ZEROS TO TYPDAT OF CIMGREC. MOVE ALO TO CSCOE OF CNGREC.
MOVE NUI TO KSEW OF CNGREC. MOVE KEYID OF RECOUT TO KEYIDL.
MOVE KEYID OF RECOUT TO KEYID OF CNGREC.
PERFORM SRCHNB. IF CHRSLD (SC) NOT = "R",
DISPLAY CHRSLD (SC), HSW,
DISPLAY "SKIPPING TO NEXT RECORD FIRST CHARACTER NOT \*$\*\",
GO TO WRTERR.
MOVE "A" TO HSW.
MOVE DSEQ OF RNCARD TO NOTESX.
EDR1.
PERFORM SRCHNB. IF CHRSLD (SC) NOT = ";",
DISPLAY "RECORD FORMAT IN ERROR", GO TO WRTERR.
MOVE "B" TO HSW.
EDR2.
PERFORM SRCHNB. IF CHRSLD (SC) NOT < 0, MOVE CHRSLD (SC)
TO BT, MOVE 1 TO YY, MOVE "C" TO HSW,
ELSE DISPLAY "RECORD NUMBER NON-NUMERIC", GO TO WRTERR.
RNUM23.
PERFORM SRCHNB. IF CHRSLD (SC) = "$" OR "*",
MOVE ALO TO HSW, GO TO TRANSA.
ADD 1 TO YY. IF YY > 3, DISPLAY "RECORD CODE ERROR",
GO TO WRTERR.
IF CHRSLD (SC) NOT < 0, MULTIPLY 10 BY BT,
MOVE CHRSLD (SC) TO DYZ, ADD DYZ TO BT,
GO TO RNUM23, ELSE DISPLAY "NON-NUMERIC IN FIELD",
GO TO WRTERR.
TRANSA.
MOVE BT TO RNUM OF CNGREC.
TRA1.
PERFORM SRCHNB. IF CHRSLD (SC) NOT = "T",
DISPLAY "T NOT FOUND", GO TO WRTERR.
MOVE "E" TO HSW.

B-33
PERFORM SRCHNB. IF CHRSLD (SC) NOT = "=",
DISPLAY "TRANSACTION FORMATTED ERROR",
GO TO WRTERK.
MOVE "F" TO HSW.

PERFORM SRCHNB. MOVE "G" TO HSW.
IF CHRSLD (SC) = "A" OR "H" OR "D" OR "V",
MOVE CHRSLD (SC) TO CCK OF CNGREC, CCSR OF CNGREC,
GO TO TRA4 ELSE DISPLAY RECOUT, CHRSLD (SC),
"NO VALID TRANSACTION CODE", GO TO WRTERK.

PERFORM SRCHNB. IF CHRSLD (SC) = "S" OR "*",
MOVE "H" TO HSW, GO TO TRA5 ELSE DISPLAY
"INCORRECT END OF TRANS CODE", GO TO WRTERK.

IF (RNUM OF CNGREC = ZERO AND CCR OF CNGREC = ALDJ)
MOVE ALX TO CCR OF CNGREC, GO TO PRTCNG.

MOVE 1 TO YY.
PERFORM SRCHNB. IF CHRSLD (SC) < 0,
DISPLAY "1ST CHARACTER OF *C NON-NUMERIC",
GO TO WRTERK.
MOVE CHRSLD (SC) TO BT. MOVE "I" TO HSW.

PERFORM SRCHNB. IF CHRSLD (SC) = "=" OR ";",
MOVE "J" TO HSW, MOVE BT TO CNUM OF CNGREC,
GO TO CDATA.
ADD 1 TO YY. IF YY > 3, DISPLAY
"SKIPPING REST OF RECORD-TOO MANY *C POSITIONS",
GO TO WRTERK.
IF CHRSLD (SC) < 0, DISPLAY
"NON-NUMERIC ENCOUNTERED LOOKING FOR REST OF *C",
GO TO WRTERK.
MULTIPLY 10 BY BT. MOVE CHRSLD (SC) TO DYZ.
ADD DYZ TO BT. GO TO CNUM23.

MOVE SPACES TO DRTSLD. MOVE 0 TO BTA.
MOVE ALL ";" TO NEWDAT OF CNGREC.
PERFORM SRCHNB. SET DC TO 1.
IF CNUM OF CNGREC = 12 OR 185, GO TO PC12.

IF CHRSLD (SC) = "*" OR ";", MOVE DATSLD TO NEWDAT OF CNGREC,
MOVE BTA TO DLEN OF CNGREC, GO TO LLLCHEK.
MOVE CHRSLD (SC) TO UDAT (DC). ADD 1 TO BTA.
SET SC UP BY 1. SET DC UP BY 1.
IF DC > 45, GO TO YMVE.
IF SC > 61, MOVE "M" TO HSW, GO TO STAR1,
ELSE GO TO CDAT1.

IF CNUM OF CNGREC = 9, GO TO LATCMP.
IF CNUM OF CNGREC = 10, GO TO LNGCMP,
ELSE GO TO PRTCNG.
LATCMP.
MOVE NEWDAT OF CNGREC TO ALPHA6.
MOVE ALPHA6 TO CMPLAT.
MOVE KEYLNG OF RECOUT TO ALFLNG.
IF CMPLAT = ALFLAT, MOVE "X" TO DELSW,
ELSE MOVE "Y" TO DELSW.
GO TO PRTCNG.

LNGCMP.
MOVE NEWDAT OF CNGREC TO ALPHA7.
MOVE ALPHA7 TO CMPLNG.
MOVE KEYLNG OF RECOUT TO ALFLNG.
IF CMPLNG = ALFLNG, MOVE "X" TO DELSW,
ELSE MOVE "Y" TO DELSW.
GO TO PRTCNG.

YMEVE.
DISPLAY "OUT OVER 45 CHARACTERS".
STOP RUN.

PC12.
IF CHRSLD (SC) NOT = "'", DISPLAY
"LOCAL WELL NUMBER DATA FIELD STARTS",
" WITH OTHER THAN ' ", GO TO WRTERR,
ELSE SET SC UP BY 1.

PC12A.
MOVE CHRSLD (SC) TO UDAT (DC). ADD 1 TO BTA.
SET SC UP BY 1. SET DC UP BY 1.
IF DC > 45, GO TO XMVE.
IF SC > 61, MOVE "K" TO HSW, GO TO STARA1.
IF CHRSLD (SC) = "'", MOVE "L" TO HSW,
GO TO LCMEVE, ELSE GO TO PC12A.

LCMVE.
PERFORM SRCHNB. IF CHRSLD (SC) = "$" OR "*",
MOVE BTA TO DLEN OF CNGREC,
MOVE DATSLO TO NEWDAT OF CNGREC, GO TO PRTCNG,
ELSE DISPLAY "END OF DATA CHARACTER MISSING",
GO TO WRTERR.

XMVE.
DISPLAY "OVER 45 CHARACTERS".
STOP RUN.

PRTCNG.
MOVE RVSN OF RECOUT TO VSN OF CNGREC.
MOVE RVSNA OF RECOUT TO VSNA OF CNGREC.
MOVE JULD TO PRO2 OF CNGREC.
IF DELSW = ALZ, ADD NU1 TO NSEQ OF CNGREC.
MOVE AID TO CSCDE OF CNGREC. MOVE CNGREC TO CIIMAGE OF PRNTLN.
ADD NU1 TO CNRECN. MOVE CNRECN TO CNRECP OF PRNTLN.
MOVE ZEROS TO TYPDAT OF CNGREC.
MOVE "H" TO HSW. IF DELSW = "Z", WRITE CNGREC.
IF DELSW = "Y", MOVE MSG2 TO NEWDAT OF CNGREC,
MOVE CNGREC TO ERRREC, WRITE ERRREC.
MOVE "7" TO DELSW. ADD NU1 TO KSEQ OF CNGREC.
PERFORM PRTSNL. GO TO CTRANS.

SRCHNB.
SET SC UP BY 1. IF SC > 61, GO TO STARA1.
IF CHRSLD (SC) = BLNK1, GO TO SRCHNB.

B-35
MOVE "X" TO HSW. ADD 1 TO ACOU1.
WRITE ERRREC FROM RECOUT. GO TO STARA1.

START8.
STOP RUN.

STARTC.
OPEN INPUT CHNGEL. MOVE ALT TO HSW.
READ CHNGEL AT END DISPLAY
"****760M2 NO DATA INPUT" STOP RUN.
MOVE NUI TO CNREC, NDATIN.

STARTC.
READ CARD-FILE INTO RANGEL AT END DISPLAY
"****761M2 END OF RUN" GO TO STOP3.
EXAMINE RANGEL REPLACING ALL " " BY ZERO.
IF NUMFST NOT = ZER0, MOVE ALX TO HSW.
IF NUMSTR NOT = ZER0, MOVE ALY TO HSW.
IF KEYIDS NOT = ZER0, MOVE ALZ.
IF HSW = ALT, DISPLAY
"****762M2 RANGE INCORRECT" GO TO STARCl.
IF HSW = ALX, ADD NUMSTR, NUMLST, KEYIDS, KEYIDE GIVING KEYIDE IF KEYIDE NOT = 0 DISPLAY
"****762M2 MULTIPLE RANGE PARAMETERS" GO TO STARCl.
IF HSW = ALY, ADD NUMFST, KEYIDS, KEYIDE GIVING KEYIDE IF KEYIDE NOT = ZER0, DISPLAY
"****764M2 SEQUENCE COUNT WITH OTHER VALUES", GO TO STARCl.
IF HSW = ALZ, ADD NUMFST, NUMSTR, NUMLST GIVING NUMLST, IF NUMLST NOT = ZER0, DISPLAY
"****765M2 SITE ID RANGE WITH OTHER VALUES", GO TO STARCl.
IF HSW = ALY AND NUMLST < NUMSTR, DISPLAY
"****766M2 SEQUENCE COUNT INVALID END", GO TO STARCl.
IF HSW = ALZ AND KEYIDE < KEYIDS, DISPLAY
"****767M2 SITE ID RANGE INVALID END LOW", GO TO STARCl.
IF HSW = ALX AND NUMFST < CNREC, DISPLAY
"****768M2 NO RECORDS PRINTED NUMFST = ", NUMFST,
" CNREC = ", CNREC, GO TO STARCl.

STARTC2.
IF HSW = ALX AND CNREC NOT > NUMFST, GO TO PRNOUT.
IF HSW = ALY AND CNREC NOT < NUMSTR AND CNREC NOT > NUMLST
GO TO PRNOUT.
IF HSW = ALZ AND KEYID OF CNGREC NOT > KEYIDS AND KEYID OF CNGREC NOT < KEYIDE, GO TO PRNOUT.

STARTC3.
READ CHNGEL AT END DISPLAY
"****769M2 END OF DATA REQUESTS INCOMPLETE" GO TO STOP3.
ADD NUI TO CNREC, NDATIN.
IF HSW = ALX AND CNREC > NUMFST GO TO STARCl.
IF HSW = ALY AND CNREC > NUMLST GO TO STARCl.
IF HSW = ALZ AND KEYID OF CNGREC < KEYIDE, GO TO STARCl.
GO TO STARC2.

PRNOUT.
IF LNCNT > 60, PERFORM HDPRNT, PERFORM HDCN3.
ADD NUI TO LNCNT, NOETLN.
MOVE CORRESPONDING CNGREC TO PCNGRC.
WRITE LIST-LINE FROM PCNGRC.
GO TO STARC3.
HUCN3.
   WRITE LIST-LINE FROM HDR3C3.
HDPRTNT.
   ADD 1 TO PGCNT.
   MOVE PGCNT TO PNUM OF HDR1.
   MOVE 4 TO LNCNT.
   WRITE LIST-LINE FROM HDR1.
   WRITE LIST-LINE FROM HDR2.
PRTSNL.
   IF LNCNT > 60, PERFORM HDPRTNT,
   ELSE ADD 1 TO LNCNT.
   WRITE LIST-LINE FROM PRNTLN.
   MOVE " " TO CC OF PRNTLN.
STOP1.
   WRITE LIST-LINE FROM J0BCNL.
   DISPLAY "CONTROL INPUT FILE EMPTY".
   STOP RUN.
STOP2.
   WRITE LIST-LINE FROM J0BCNL.
   DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
   STOP RUN.
STOP3.
   WRITE LIST-LINE FROM J0BCNL.
   DISPLAY TFILE8.
   MOVE INCROS TO CSTAT.
   MOVE NDATIN TO DATAIN.
   MOVE NFLEIA TO FADATA.
   MOVE NDTLN TO DRLNS.
   MOVE NFLEOA TO FOUT.
   WRITE LIST-LINE FROM J0BCNLX.
   DISPLAY "NORMAL JOB TERMINATION".
   DISPLAY "ERRORS = ", ACOUNT.
   CLOSE CARD-FILE, LIST-FILE.
   STOP RUN.
STOP4.
   WRITE LIST-LINE FROM J0BCNL.
   DISPLAY "STOP4".
   STOP RUN.
STOP5.
   WRITE LIST-LINE FROM J0BCNL.
   DISPLAY "STOP5".
   STOP RUN.
STOP6.
   WRITE LIST-LINE FROM J0BCNL.
   DISPLAY "STOP6".
   STOP RUN.
STOP7.
   DISPLAY "STOP7". STOP RUN.
STOP8.
  WRITE LIST-LINE FROM JOBCNL.
  DISPLAY "STOP8".
  STOP RUN.

STOP9.
  WRITE LIST-LINE FROM JOBCNL.
  DISPLAY "STOP9".
  STOP RUN.

STOP10.
  WRITE LIST-LINE FROM JOBCNL.
  DISPLAY "STOP10".
  STOP RUN.
IDENTIFICATION DIVISION.
PROGRAM-ID. EUTT-3.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. 6600.
OBJECT-COMPUTER. 6600.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT CARD-FILE ASSIGN TO "INPUT".
SELECT LIST-FILE ASSIGN TO "OUTPUT".
SELECT FILEA ASSIGN TO UPCHNG.
SELECT FILEB ASSIGN TO ERCHNG.
SELECT CHANGE-FILE ASSIGN TO EDCHNG
RESERVE 10 ALTERNATE AREAS.
DATA DIVISION.
FILE SECTION.
FD CARD-FILE
LABEL RECORD OMITTED
DATA RECORD IS CNL-CARD.
01 CNL-CARD.
 02 CNLXXX.
    03 ID-CNLT PIC IS XXX.
    03 ID-NUm PIC 999.
 02 ID-DES PIC X(74).
FD LIST-FILE
LABEL RECORD OMITTED
DATA RECORD IS LIST-LINE.
01 LIST-LINE.
 03 CC PIC X.
 03 LLPRT PIC X(132).
FD FILEA
DATA RECORD IS RECOUT
LABEL RECORDS ARE OMITTED.
01 RECOUT.
 03 VSN PIC 9(5).
 03 VSNM PIC X.
 03 PRD2 PIC 9(5).
 03 KSEQ PIC 9(4).
 03 KEYID.
    05 KEYLAT.
      07 LATDEG PIC 99.
      07 LATMIN PIC 99.
      07 LATSEC PIC 99.
    05 KEYLNG.
      07 LNGDEG PIC 999.
      07 LNGMIN PIC 99.
      07 LNGSEC PIC 99.
    05 KEYDUP PIC 99.
 03 CSCDE PIC X.
 03 RNUM PIC 999.
 03 CNUM PIC 999.
 03 CCR PIC X.
 03 CCSR PIC X.
 03 TYPDAT PIC 99.

B-39
FD FILE8
DATA RECORD IS AUOREC
LABEL RECORDS ARE OMITTED.

01 AUDREC.
   03 VSN  PIC 9(5).
   03 VSNA PIC X.
   03 PRD<2 PIC 9(5).
   03 KSEQ PIC 9(4).
   03 KEYID.
      05 KEYLAT.
         07 LATDEG PIC 99.
         07 LATMIN PIC 99.
         07 LATSEC PIC 99.
      05 KEYLNG.
         07 LNGDEG PIC 999.
         07 LNGMIN PIC 99.
         07 LNGSEC PIC 99.
      05 KEYDUP PIC 99.
   03 CSCDE PIC X.
   03 RNUM PIC 999.
   03 CNUM PIC 999.
   03 CCR PIC X.
   03 CCSR PIC X.
   03 TYPDAT PIC 99.
   03 DLEN PIC 99.
   03 NSEQ PIC 9(8).

FD CHANGE-FILE
DATA RECORD IS CNGREC
LABEL RECORDS ARE OMITTED.

01 CNGREC.
   03 VSN  PIC 9(5).
   03 VSNA PIC X.
   03 PRD<2 PIC 9(5).
   03 KSEQ PIC 9(4).
   03 KEYID.
      05 KEYLAT.
         07 LATDEG PIC 99.
         07 LATMIN PIC 99.
         07 LATSEC PIC 99.
      05 KEYLNG.
         07 LNGDEG PIC 999.
         07 LNGMIN PIC 99.
         07 LNGSEC PIC 99.
      05 KEYDUP PIC 99.
   03 CSCDE PIC X.
   03 RNUM PIC 999.
   03 CNUM PIC 999.
   03 CCR PIC X.
   03 CCSR PIC X.
   03 TYPDAT PIC 99.
WORKING-STORAGE SECTION.
77 ALFAA PIC X VALUE "A".
77 ALFAB PIC X VALUE "B".
77 ALFAC PIC X VALUE "C".
77 ALFAD PIC X VALUE "D".
77 ALFAE PIC X VALUE "E".
77 ALFAF PIC X VALUE "F".
77 ALFAG PIC X VALUE "G".
77 ALFAH PIC X VALUE "H".
77 ALFAI PIC X VALUE "I".
77 ALFAJ PIC X VALUE "J".
77 ALFAK PIC X VALUE "K".
77 ALFAL PIC X VALUE "L".
77 ALFAM PIC X VALUE "M".
77 ALFAN PIC X VALUE "N".
77 ALFAO PIC X VALUE "O".
77 ALFAP PIC X VALUE "P".
77 ALFAQ PIC X VALUE "Q".
77 ALFAQ PIC X VALUE "R".
77 ALFAS PIC X VALUE "S".
77 ALFAT PIC X VALUE "T".
77 ALFAU PIC X VALUE "U".
77 ALFAV PIC X VALUE "V".
77 ALFAW PIC X VALUE "W".
77 ALFAX PIC X VALUE "X".
77 ALFAY PIC X VALUE "Y".
77 ALFAZ PIC X VALUE "Z".
77 BT PIC 9999 VALUE 0.
77 BTB PIC 999.
77 CNRECN PIC 9(6) VALUE ZEROES.
77 DCOUNT PIC 9(4) VALUE 0.
77 DELSW PIC X.
77 DECSW PIC 9 VALUE 0.
77 DECSW PIC 9 VALUE 0.
77 DGCNT PIC 999 VALUE 0.
77 EDMSG1 PIC X(22) VALUE " EDITED -C-NUMBER = ".
77 EMSG1 PIC X(22) VALUE " *** REJECT ERROR NO = ".
77 EOFSW PIC S99 VALUE -1.
77 EQSW PIC 9999 VALUE 0.
77 ERCODE PIC X.
77 ERMST PIC X(27) VALUE "DELETED CHANGES EDIT ERRORS".
77 ERMSG2 PIC X(27) VALUE " GW & QW LOCAL IDS UNEQUAL ".
77 ERNUM PIC 9(5) VALUE 0.
77 ERRCNT PIC 9(9) VALUE 0.
77 ERRSW PIC X VALUE "Y".
77 ESMSG1 PIC X(20) VALUE "FILE OUT OF SEQUENCE".
77 ESMSG2 PIC X(26) VALUE "CORRECT SEQUENCE AND RERUN".
77 ESW PIC 9 VALUE 0.
77 FLECNT PIC 9(6).
77 HIALT PIC X(6) VALUE "12633".
77 INCRRS PIC 9(6) VALUE 0.
77 KTEST PIC 9(15).
77 LNCT PIC 99 VALUE 70.
77 LOALT PIC X(6) VALUE "0".
77 MAXDEP PIC X(5) VALUE "3000".
77 NLNK1 PIC X VALUE " ".
77 NDATIN PIC 9(6) VALUE 0.
77 NDETLN PIC 9(6) VALUE 0.
77 NEGSN PIC 9 VALUE 0.
77 NFLEA PIC 9(6) VALUE 0.
77 NFLEOA PIC 9(6) VALUE 0.
77 NMM PIC 9999.
77 NUMS PIC 999.
77 NXX PIC 999.
77 NU0 PIC 9 VALUE 0.
77 NU1 PIC 9 VALUE 1.
77 NU2 PIC 9 VALUE 2.
77 NU3 PIC 9 VALUE 3.
77 NU4 PIC 9 VALUE 4.
77 NU5 PIC 9 VALUE 5.
77 NU6 PIC 9 VALUE 6.
77 NU7 PIC 9 VALUE 7.
77 NU8 PIC 9 VALUE 8.
77 NU9 PIC 9 VALUE 9.
77 NU10 PIC 99 VALUE 10.
77 NU11 PIC 99 VALUE 11.
77 NU12 PIC 99 VALUE 12.
77 NU13 PIC 99 VALUE 13.
77 NU14 PIC 99 VALUE 14.
77 NU15 PIC 99 VALUE 15.
77 NU16 PIC 99 VALUE 16.
77 NU17 PIC 99 VALUE 17.
77 NU18 PIC 99 VALUE 18.
77 NU19 PIC 99 VALUE 19.
77 NU20 PIC 99 VALUE 20.
77 NU21 PIC 99 VALUE 21.
77 NU22 PIC 99 VALUE 22.
77 NU23 PIC 99 VALUE 23.
77 NU24 PIC 99 VALUE 24.
77 NU25 PIC 99 VALUE 25.
77 NU26 PIC 99 VALUE 26.
77 NU27 PIC 99 VALUE 27.
77 N1 PIC 9.
77 N2 PIC 9.
77 N3 PIC 999.
77 N4 PIC 999.
77 N5 PIC 9(5).
77 N6 PIC 9(6).
77 PGCNT PIC 999 VALUE 0.
77 SC PIC 99.
77 RDCNT PIC 9(5) VALUE 0.
77 SE PIC 99.
77 SEQCHK PIC 9(6) VALUE 0.
77 TDYDAT PIC X(8).
77 TFILEA PIC 9(6) VALUE 0.
77 TFILE8 PIC 9(6) VALUE 0.

B-42
77 TFILEC PIC 9(6) VALUE 0.
77 UDCHK PIC X.
77 UPNUM PIC 9(5) VALUE 0.
77 VALC27 PIC X(4).
77 XBLNK1 PIC X VALUE " ".
77 XBLNK2 PIC XX VALUE SPACES.
77 XBLNK3 PIC XXX VALUE SPACES.
77 XBLNK4 PIC X(4) VALUE SPACES.
77 XBLNK5 PIC X(5) VALUE SPACES.
77 XBLNK6 PIC X(6) VALUE SPACES.
77 XBLNK7 PIC X(7) VALUE SPACES.
77 XBLNK8 PIC X(8) VALUE SPACES.
77 XBLNK9 PIC X(9) VALUE SPACES.
77 XBLNK10 PIC X(10) VALUE SPACES.
77 XBLNK11 PIC X(11) VALUE SPACES.
77 XBLNK12 PIC X(12) VALUE SPACES.
77 XBLNK13 PIC X(13) VALUE SPACES.
77 XBLNK14 PIC X(14) VALUE SPACES.
77 XBLNK15 PIC X(15) VALUE SPACES.
77 XBLNK16 PIC X(16) VALUE SPACES.
77 XBLNK17 PIC X(17) VALUE SPACES.
77 XBLNK18 PIC X(18) VALUE SPACES.
77 XBLNK19 PIC X(19) VALUE SPACES.
77 XBLNK20 PIC X(20) VALUE SPACES.
77 XBLNK21 PIC X(21) VALUE SPACES.
77 XBLNK22 PIC X(22) VALUE SPACES.
77 XBLNK23 PIC X(24) VALUE SPACES.
77 XBLNK25 PIC X(25) VALUE SPACES.
77 ZERO PIC X VALUE "0".
77 X1 PIC X(1).
77 X2 PIC X(2).
77 X3 PIC X(3).
77 X4 PIC X(4).
77 X5 PIC X(5).
77 X6 PIC X(6).
77 X7 PIC X(7).
77 X8 PIC X(8).
77 X9 PIC X(9).
77 X10 PIC X(10).
77 X11 PIC X(11).
77 X12 PIC X(12).
77 X13 PIC X(13).
77 X14 PIC X(14).
77 X15 PIC X(15).
77 X16 PIC X(16).
77 X17 PIC X(17).
77 X18 PIC X(18).
77 X19 PIC X(19).
77 X20 PIC X(20).
77 X21 PIC X(21).
77 X22 PIC X(22).
77 X23 PIC X(23).
77 X24 PIC X(24).
77 X25 PIC X(25).
77 NUMER PIC 9(11)V9(7) USAGE IS COMP.
01 SWITCHES.
 03 C-ENO PIC X VALUE "N".
 88 END-CHANGE VALUE "Y".
 03 PROC-SW PIC X VALUE "N".
 88 PROC-COMPLETE VALUE "Y".
 03 SENSW PIC X VALUE "N".
 03 RNSW PIC X VALUE "X".

01 QwCUDS.
 03 Qw1 PIC 9(5) VALUE 00010.
 03 Qw2 PIC 9(5) VALUE 00095.
 03 Qw3 PIC 9(5) VALUE 00300.
 03 Qw4 PIC 9(5) VALUE 00400.
 03 Qw5 PIC 9(5) VALUE 00405.
 03 Qw6 PIC 9(5) VALUE 00410.
 03 Qw7 PIC 9(5) VALUE 00430.
 03 Qw8 PIC 9(5) VALUE 00440.
 03 Qw9 PIC 9(5) VALUE 00445.
 03 Qw10 PIC 9(5) VALUE 00900.
 03 Qw11 PIC 9(5) VALUE 00940.
 03 Qw12 PIC 9(5) VALUE 00945.
 03 Qw13 PIC 9(5) VALUE 00950.
 03 Qw14 PIC 9(5) VALUE 01045.
 03 Qw15 PIC 9(5) VALUE 31501.
 03 Qw16 PIC 9(5) VALUE 31625.
 03 Qw17 PIC 9(5) VALUE 31673.
 03 Qw18 PIC 9(5) VALUE 71820.
 03 Qw19 PIC 9(5) VALUE 71830.

01 NUMSOUT.
 03 Num02 PIC V99.
 03 Num07 PIC V9(7).
 03 Num23 PIC 99V999.
 03 Num25 PIC 99V9(5).
 03 Num32 PIC 99V999.
 03 Num41 PIC 999V99.
 03 Num42 PIC 9999V99.
 03 Num52 PIC 9(5)V99.
 03 Num54 PIC 9(5)V9(4).
 03 Num61 PIC 9(6)V9.
 03 Num63 PIC 9(6)V999.

01 SIGNUMS.
 03 SNUM32 PIC S999V99.
 03 SNUM42 PIC S9999V999.
 03 SNUM52 PIC S99999V999.
 03 TSNUM32 PIC 999.99-.
 03 TSNUM42 PIC 9999.99-.
 03 TSNUM52 PIC 99999.99-.

01 TESTNUM.
 03 TN02 PIC .99.
 03 TN07 PIC .9(7).
 03 TN23 PIC 99.999.
 03 TN25 PIC 99.9(5).
 03 TN32 PIC 999.99.
 03 TN41 PIC 9999.9.
 03 TN42 PIC 9999.99.
03 TN52 PIC 9(5).99.
03 TN54 PIC 9(5).9999.
03 TN63 PIC 9(6).999.

01 NNUM.
03 MDIGIT PIC 9 OCCURS 11 TIMES INDEXED BY NLN.
01 ALFNUM.
03 ADIGIT PIC X OCCURS 20 TIMES INDEXED BY ALN.
01 GEO-CODES.

03 ALVM PIC X(5) VALUE "ALVM".
03 ANDR PIC X(5) VALUE "ANDR".
03 ANDRS PIC X(5) VALUE "ANDRS".
03 ARKS PIC X(5) VALUE "ARKS".
03 BSLT PIC X(5) VALUE "BSLT".
03 BMTN PIC X(5) VALUE "BMTN".
03 BLDR PIC X(5) VALUE "BLDR".
03 BLSO PIC X(5) VALUE "BLSO".
03 BLSC PIC X(5) VALUE "BLSC".
03 BRCC PIC X(5) VALUE "BRCC".
03 CLAY PIC X(5) VALUE "CLAY".
03 CLSN PIC X(5) VALUE "CLSN".
03 COAL PIC X(5) VALUE "COAL".
03 COBB PIC X(5) VALUE "COBB".
03 COSD PIC X(5) VALUE "COSD".
03 COSC PIC X(5) VALUE "COSC".
03 CLVM PIC X(5) VALUE "CLVM".
03 CGLM PIC X(5) VALUE "CGLM".
03 CQUN PIC X(5) VALUE "CQUN".
03 DIBS PIC X(5) VALUE "DIBS".
03 GLCL PIC X(5) VALUE "GLCL".
03 GNSS PIC X(5) VALUE "GNSS".
03 GRNT PIC X(5) VALUE "GRNT".
03 GRGN PIC X(5) VALUE "GRGN".
03 GRVL PIC X(5) VALUE "GRVL".
03 GRCM PIC X(5) VALUE "GRCM".
03 GRCL PIC X(5) VALUE "GRCL".
03 GRDS PIC X(5) VALUE "GRDS".
03 GRSC PIC X(5) VALUE "GRSC".
03 GRCK PIC X(5) VALUE "GRCK".
03 HROP PIC X(5) VALUE "HROP".
03 IGNS PIC X(5) VALUE "IGNS".
03 LGNT PIC X(5) VALUE "LGNT".
03 LMSN PIC X(5) VALUE "LMSN".
03 LMDM PIC X(5) VALUE "LMDM".
03 LOAM PIC X(5) VALUE "LOAM".
03 LOSS PIC X(5) VALUE "LOSS".
03 MRRL PIC X(5) VALUE "MRRL".
03 MARL PIC X(5) VALUE "MARL".
03 MRLS PIC X(5) VALUE "MRLS".
03 MMPC PIC X(5) VALUE "MMPC".
03 MUCK PIC X(5) VALUE "MUCK".
03 MUD PIC X(5) VALUE "MUD".
03 MDSN PIC X(5) VALUE "MDSN".
03 RSDM PIC X(5) VALUE "RSDM".
03 RYLT PIC X(5) VALUE "RYLT".
03 RUCK PIC X(5) VALUE "RUCK".
03 R3RL PIC X(5) VALUE "R3RL".
03 SAND PIC X(5) VALUE "SAND".
03 SOCL PIC X(5) VALUE "SOCL".
03 S6RL PIC X(5) VALUE "SU6L".
03 S6ST PIC X(5) VALUE "S6ST".
03 S6VC PIC X(5) VALUE "S6VC".
03 S6DS PIC X(5) VALUE "S6DS".
03 S6AN PIC X(5) VALUE "S6AN".
03 S6LE PIC X(5) VALUE "S6LE".
03 S6LT PIC X(5) VALUE "S6LT".
03 ST6L PIC X(5) VALUE "ST6L".
03 SLSN PIC X(5) VALUE "SLSN".
03 S6TE PIC X(5) VALUE "S6TE".
03 S6TL PIC X(5) VALUE "S6TL".
03 S6NT PIC X(5) VALUE "S6NT".
03 S6LL PIC X(5) VALUE "S6LL".
03 TILL PIC X(5) VALUE "TILL".
03 T6F PIC X(5) VALUE "T6F".

01 WKDATE.
03 YY PIC 99.
03 MM PIC 99.
03 DY PIC 99.

01 NAME-MONTH.
03 JAN PIC XXXX VALUE "JAN".
03 FER PIC XXXX VALUE "FEB".
03 MAR PIC XXXX VALUE "MAY".
03 APR PIC XXXX VALUE "APR".
03 MAY PIC XXXX VALUE "MAY".
03 JUNE PIC XXXX VALUE "JUNE".
03 JULY PIC XXXX VALUE "JULY".
03 AUG PIC XXXX VALUE "AUG".
03 SEPT PIC XXXX VALUE "SEPT".
03 OCT PIC XXXX VALUE "OCT".
03 NOV PIC XXXX VALUE "NOV".
03 DEC PIC XXXX VALUE "DEC".

01 MONTH-TABLE REDEFINES NAME-MONTH.
03 MONTHY PIC XXXX OCCURS 12 TIMES.

01 JCBCNL.
03 CC PIC X VALUE "1".
03 PCNL.
 05 CNTYPE PIC XXX.
 05 GO-CNL PIC 999.
 05 PCNLX PIC X(74).
03 FILLER PIC X(52) VALUE SPACES.

01 JCBCNLX.
03 CC PIC X VALUE "0".
03 PSTATS PIC X(14) VALUE "RECORD COUNTS".
03 FILLER PIC X(6) VALUE SPACES.
03 XCARD PIC X(8) VALUE "CARDS IN".
03 CSTAT PIC ZZZZZZ9.
03 FILLER PIC X(8) VALUE SPACES.
03 XDRRLNS PIC X(12) VALUE "DETAIL LINES".
03 DRRLNS PIC ZZZZZ9.
03 FILLER PIC X(8) VALUE SPACES.
03 XDATIZN PIC X(7) VALUE "DATA IN".
03 DATAIN PIC ZZZZ9.
03 FILLER PIC X(8) VALUE SPACES.
03 XFAIN PIC X(11) VALUE "ULD-DATA I".
03 FDATA PIC ZZZZZ9.
03 FILLER PIC X(8) VALUE SPACES.
03 XNOUT PIC X(12) VALUE "FMT DATA OUT".
03 FOUT PIC ZZZZZ9.
01 HDR1.
  03 CCI PIC 9 VALUE 1.
  03 FILLER PIC X.
  03 LDATE PIC X(6) VALUE "DATE: ".
  03 KMTH PIC X(4).
  03 FILLER PIC X.
  03 RDY PIC Z9.
  03 XB PIC XXXX VALUE ", 19".
  03 RYR PIC XX.
  03 FILLER PIC X(21).
  03 TITLEA PIC X(60).
  03 FILLER PIC X(21).
  03 PGE PIC X(4) VALUE "PAGE".
  03 PNUM PIC ZZZ9.
  03 FILLER PIC X(2).
01 HDR2.
  03 CC PIC 9 VALUE 0.
  03 USGS PIC X(7) VALUE " USGS: ".
  03 GSNAME PIC X(20).
  03 FILLER PIC XXX.
  03 FILLER PIC X(7) VALUE "SOURCES".
  03 FILLER PIC X.
  03 SRCEID PIC X(50).
  03 SRCNAM PIC X(30).
  03 CNLZZZ PIC X(6).
  03 FILLER PIC X(8).
01 HDR3.
  03 CC PIC 9 VALUE 0.
  03 PCNG PIC X(5) VALUE " CNG ".
  03 PLCWL PIC X(20) VALUE " LOCAL WELL ID ".
  03 PSITE PIC X(16) VALUE " SITE ID ".
  03 PDUP PIC XX VALUE " D ".
  03 PCR PIC X(5) VALUE " C**R ".
  03 PITEM PIC X(5) VALUE " C* ".
  03 PNDAT PIC X(25) VALUE " NEW DATA ".
  03 PCOND PIC X(25) VALUE " CONDITION ".
  03 PSRCCE PIC X(12) VALUE " SOURCE ".
  03 PCCR PIC X VALUE " D ".
  03 FILLER PIC XX VALUE ".--".
  03 PCCSR PIC X VALUE " S ".
  03 PNOTE PIC X(12) VALUE " NOTES ".
01 PRNTLN.
  03 CC PIC X VALUE SPACE.
  03 VSN PIC 9(5).
  03 VSNA PIC X.
  03 PRD2 PIC 9(5).
  03 KSEQ PIC 9(4).
  03 KEYID.
05 KEYLAT.
  07 LATDEG PIC 99.
  07 LATMIN PIC 99.
  07 LATSEC PIC 99.
05 KEYLNG.
  07 LNGDEG PIC 999.
  07 LNGMIN PIC 99.
  07 LNGSEC PIC 99.
05 KEYDUP PIC 99.
  03 CSCDE PIC X.
  03 RNUM PIC 999.
  03 CNUM PIC 999.
  03 CCR PIC X.
  03 CCSR PIC X.
  03 TYPDAT PIC 99.
  03 DLEN PIC 99.
  03 NEWDAT PIC X(45).
  03 NSEQ PIC 9(8).
  03 ERPRINT PIC X(30).
01 CCARD.
  03 VSN PIC 9(5).
  03 VSNA PIC X.
  03 PRD2 PIC 9(5).
  03 KSEQ PIC 9(4).
  03 KEYID.
05 KEYLAT.
  07 LATDEG PIC 99.
  07 LATMIN PIC 99.
  07 LATSEC PIC 99.
05 KEYLNG.
  07 LNGDEG PIC 999.
  07 LNGMIN PIC 99.
  07 LNGSEC PIC 99.
05 KEYDUP PIC 99.
  03 CSCDE PIC X.
  03 RNUM PIC 999.
  03 CNUM PIC 999.
  03 CCR PIC X.
  03 CCSR PIC X.
  03 TYPDAT PIC 99.
  03 DLEN PIC 99.
  03 NEWDAT PIC X(45).
  03 NSEQ PIC 9(8).
  03 FILETY PIC XX.
01 SITELN.
  03 CC PIC X VALUE "0".
  03 FILLER PIC X(5) VALUE "***".
  03 FILLER PIC X(7) VALUE "ERROR =".
  03 BTE PIC ZZ9.
  03 FILLER PIC X(5).
  03 FILLER PIC X(5) VALUE "***".
  03 FILLER PIC X(112).
01 DATAA.
 03 ID-CNl PIC IS XXX.
 03 ID-NUM PIC IS 999.
 03 TITLEX.
    05 SNCAGC.
      07 GNAME PIC X(20).
      07 CNAME PIC X(30).
    05 FILLER PIC X(10).
 03 FILLER PIC X(14).
01 KEYIOL.
 03 KEYLAT.
    05 LATDEG PIC 99 VALUE 99.
    05 LATMIN PIC 99 VALUE 99.
    05 LATSEC PIC 99 VALUE 99.
 03 KEYLNG.
    05 LNGDEG PIC 999 VALUE 999.
    05 LNGMIN PIC 99 VALUE 99.
    05 LNGSEC PIC 99 VALUE 99.
 03 KEVDUP PIC 99 VALUE 99.
01 KEYIDC.
 03 KEYLAT.
    05 LATDEG PIC 99.
    05 LATMIN PIC 99.
    05 LATSEC PIC 99.
 03 KEYLNG.
    05 LNGDEG PIC 999.
    05 LNGMIN PIC 99.
    05 LNGSEC PIC 99.
 03 KEVDUP PIC 99.
01 KEYIDD.
 03 KEYLAT.
    05 LATDEG PIC 99 VALUE 99.
    05 LATMIN PIC 99 VALUE 99.
    05 LATSEC PIC 99 VALUE 99.
 03 KEYLNG.
    05 LNGDEG PIC 999 VALUE 999.
    05 LNGMIN PIC 99 VALUE 99.
    05 LNGSEC PIC 99 VALUE 99.
 03 KEVDUP PIC 99 VALUE 89.
01 DELTSS.
 03 KEYLAT.
    05 LATDEG PIC 99 VALUE 99.
    05 LATMIN PIC 99 VALUE 99.
    05 LATSEC PIC 99 VALUE 99.
 03 KEYLNG.
    05 LNGDEG PIC 999 VALUE 999.
    05 LNGMIN PIC 99 VALUE 99.
    05 LNGSEC PIC 99 VALUE 99.
 03 KEVDUP PIC 99 VALUE 99.
01 ANYTAB.
 02 RECTAB OCCURS 50 TIMES.
 03 VSN PIC 9(5).
 03 VSNA PIC X.
 03 PRD2 PIC 9(5).
03 KSEQ PIC 9(4).
03 KEYID.
  05 KEYLAT.
    07 LATDEG PIC 99.
    07 LATMIN PIC 99.
    07 LATSEC PIC 99.
  05 KEYLNG.
    07 LNGDEG PIC 999.
    07 LNGMIN PIC 99.
    07 LNGSEC PIC 99.
  05 KEYDUP PIC 99.
  03 CSCOE PIC X.
  03 RNUM PIC 999.
  03 CNUM PIC 999.
  03 CCR PIC X.
  03 CCSR PIC X.
  03 TYPDAT PIC 99.
  03 DLEN PIC 99.
  03 NEWDAT PIC X(45).
  03 MSEQ PIC 9(8).
  03 ERRCODE PIC 99.

01 CHKSIT.
  03 LC1 PIC X.
  03 LC2 PIC X.
  03 LC3 PIC 99.
  03 LC4 PIC X.
  03 LC5 PIC 99.
  03 LC6 PIC X.
  03 LC7 PIC 99.
  03 LC8 PIC X.
  03 LC9 PIC X.
  03 LC10 PIC X.
  03 LC11 PIC X.
  03 RESTLC.
    05 R1 PIC X.
    05 R2 PIC X.
    05 R3 PIC X.
    05 R4 PIC X.
    05 R5 PIC X.
    05 R6 PIC X.

01 OLDMAS.
  03 VSN PIC 9(5).
  03 VSNA PIC X.
  03 PRO2 PIC 9(5).
  03 KSEQ PIC 9(4).
  03 KEYID.
    05 KEYLAT.
      07 LATDEG PIC 99.
      07 LATMIN PIC 99.
      07 LATSEC PIC 99.
    05 KEYLNG.
      07 LNGDEG PIC 999.
      07 LNGMIN PIC 99.
      07 LNGSEC PIC 99.
    05 KEYDUP PIC 99.

B-50
03 CSCDE PIC X.
03 RNUM PIC 999.
03 CNUM PIC 999.
03 CCR PIC X.
03 CCSR PIC X.
03 TYPDAT PIC 99.
03 DLEN PIC 99.
03 NEWAT PIC X(45).
03 NSEQ PIC 9(8).
03 ERRCODE PIC 99.
01 ERRCNGL.
03 VSN PIC 9(5).
03 VSNA PIC X.
03 PRD2 PIC 9(5).
03 KSEQ PIC 9(4).
03 KEYID.
05 KEYLAT.
 07 LATDEG PIC 99.
 07 LATMIN PIC 99.
 07 LATSEC PIC 99.
05 KEYLNG.
 07 LNGDEG PIC 999.
 07 LNGMIN PIC 99.
 07 LNGSEC PIC 99.
05 KEYDUP PIC 99.
03 CSCDE PIC X.
03 RNUM PIC 999.
03 CNUM PIC 999.
03 CCR PIC X.
03 CCSR PIC X.
03 TYPDAT PIC 99.
03 DLEN PIC 99.
03 NEWAT PIC X(45).
03 NSEQ PIC 9(8).
01 ALLSWS.
03 S001 PIC 9.
03 S002 PIC 9.
03 S003 PIC 9.
03 S004 PIC 9.
03 S005 PIC 9.
03 S006 PIC 9.
03 S007 PIC 9.
03 S008 PIC 9.
03 S009 PIC 9.
03 S010 PIC 9.
03 S011 PIC 9.
01 STAR-LINE.
03 FILLER PIC X VALUE SPACES.
03 FILLER PIC X(130) VALUE ALL "*".
01 EDIT-ERRORS.
03 ED-ER1 PIC X(30) VALUE "INVALID RECORD NUMBER-RNUM ".
03 ED-ER2 PIC X(30) VALUE "REQUIRED DATA FIELD MISSING".
03 ED-ER3 PIC X(30) VALUE "CHANGE < 1 OR > 25 CHAR".
03 ED-ER4 PIC X(30) VALUE "DISTRICT CODE NOT '04' - AZ".
03 ED-ER5 PIC X(30) VALUE "INVALID ALPHA CODE IN CHANGE".

B-51
01 ED-ER6 PIC X(30) VALUE "STATE CODE NOT '04' - AZ".
03 ED-ER7 PIC X(30) VALUE "COUNTY ID IS NOT VALID CODE".
03 ED-ER8 PIC X(30) VALUE "INVALID QUAD IN LOCAL ID".
03 ED-ER9 PIC X(30) VALUE "INVALID LOCAL IDENTIFIER".
03 ED-ER10 PIC X(30) VALUE "ALT OUTSIDE ACCEPTABLE RANGE".
03 ED-ER11 PIC X(30) VALUE "DATE IS NOT NUMERIC".
03 ED-ER12 PIC X(30) VALUE "DEPTH OF HOLE IS NOT VALID".
03 ED-ER13 PIC X(30) VALUE "DEPTH OF WELL IS NOT VALID".
03 ED-ER14 PIC X(30) VALUE "ENTRY NUMBER IS NOT NUMERIC".
03 ED-ER15 PIC X(30) VALUE "INVALID LITHOLOGIC CODE".
03 ED-ER16 PIC X(30) VALUE "SOURCE AGENCY NOT VALID".
03 ED-ER17 PIC X(30) VALUE "DATE IS LATER THAN TODAY DATE".
03 ED-ER18 PIC X(30) VALUE "MANDATORY FIELD IS BLANK".
03 ED-ER19 PIC X(30) VALUE "QW CODE NOT VALID STORET CODE".
03 ED-ER20 PIC X(30) VALUE "MESSAGE 20 AVAILABLE".
03 ED-ER21 PIC X(30) VALUE "MESSAGE 21 AVAILABLE".
03 ED-ER22 PIC X(30) VALUE "MESSAGE 22 AVAILABLE".
03 ED-ER23 PIC X(30) VALUE "MESSAGE 23 AVAILABLE".
03 ED-ER24 PIC X(30) VALUE "MESSAGE 24 AVAILABLE".
03 ED-ER25 PIC X(30) VALUE "MESSAGE 25 AVAILABLE".

01 ERR-MSG-TABLE REDEFINES EDIT-ERRORS.
03 ER-MSG PIC X(30) OCCURS 25 TIMES.

PROCEDURE DIVISION.
MAIN-PROC-00.
PERFORM BEGIN.
PERFORM READ-C.
PERFORM INIT-EDIT.
PERFORM EDIT-CHANGE UNTIL EOF $W > -1.
PERFORM FINALA.
BEGIN.
OPEN INPUT CARD-FILE, CHANGE-FILE.
OPEN OUTPUT LIST-FILE, FILEA, FILEB.
ACCEPT WKDATE FROM DATE.
MOVE WKDATE TO TTYDAT.
MOVE MM TO NMM.
MOVE MONTHY (NMM) TO RMTH OF HDR1.
MOVE DD TO RDY OF HDR1.
MOVE YY TO RYR OF HDR1.
READ-C.
READ CARD-FILE AT END GO TO STOP1.
IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
ADD 1 TO INCRDS.
MOVE CNL-CARD TO PCNL.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 010 MOVE TITLEX OF DATAA
TO TITLEA OF HDR1 ELSE GO TO STOP4.
ADD 1 TO INCRDS.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.

B-52
IF ID-CNL OF UAIAA NOT = "CNL" GO TU S1P4.
IF ID-NUM OF UAIAA = 011
  MOVE GNAME TU GSNAME
  MOVE CNAME TU SRCNAM
ELSE GO TO S1P4.
ADD 1 TO INCRDS.
READ CARD-FILE INTO DATAA
AT END GO TO S1P4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO S1P4.
IF ID-NUM OF DATAA = 012
  MOVE SRCAGC TO SRCE1D,
ELSE GO TO S1P4.
MOVE GO-CNLL TU CNLZZZ.
ADD 1 TO INCRDS.
INIT-EDIT.
  PERFORM CLEAR-PROC.
  PERFORM READ-CHANGE.
  IF EOFSW > -1, GO TU S1P3.
  MOVE KEYID OF CNGREC TO KEYIDL.
CLEAR-PROC.
  MOVE ALFAM TO ERRSW.
  MOVE NU0 TO NMM.
  ADD NU1 TO BT, LNCNT.
  MOVE NU0 TO DCOUNT, MM, YY, CC OF PRNLN.
  MOVE BT TO NSEQ OF PRNLN, NSEQ OF RECOUT
  MOVE ALFAN TO C-END, PROC-SW.
EDIT-CHANGE.
  PERFORM STORE-REC UNTIL C-END = ALFAY OR EOFSW > -1.
  PERFORM ALL-PROC UNTIL PROC-SW = ALFAY.
  PERFORM WHITE-LIST.
READ-CHANGE.
  READ CHANGE-FILE,
  AT END MOVE 1 TO EOFSW.
  IF EOFSW < NU1,
    ADD NU1 TO CNRECN, NFLEIA, TFILEC.
STORE-REC.
  IF NMM > 50 GO TO STOP9.
  PERFORM KSEQ-CHECK.
  IF SEQSW = ALFAY,
    ADD NU1 TO NMM
    MOVE CNGREC TO RECTAB (NMM)
    MOVE ALFAN TO C-END,
    PERFORM READ-CHANGE
  ELSE
    MOVE ALFAY TO C-END.
KSEQ-CHECK.
  IF KSEQ OF CNGREC > DCOUNT,
    MOVE KSEQ OF CNGREC TO DCOUNT,
    MOVE ALFAY TO SEQSW,
  ELSE
    MOVE ALFAN TO SEQSW.
ALL-PROC.
ADD NUI TO MM.
IF (MM = NMM OR MM > NMM)
  MOVE ALFAY TO PROC-SW
ELSE
  MOVE ALFAN TO PROC-SW.
  MOVE RECTAB (MM) TO OLOMAS.
  MOVE ZEROS TO ERCODE.
  MOVE ALFAN TO ERRSW.
  IF CCSR OF OLOMAS = ALFAA
     PERFORM DIST-CHK.
  IF (CCSR OF OLOMAS = ALFAD OR ALFAV)
     PERFORM ENDED.
ELSE
  PERFORM CHNG-EXAM.
CHNG-EXAM.
  PERFORM NEW-EXAM.
  PERFORM EXAM-CHANGE.
NEW-EXAM.
  MOVE NUO TO DECSW, DGCNT, DECNT, NEGSW.
  MOVE ZEROS TO NUMER, NUMSOUT, SIGNUMS, TESTNUM, NNUM.
  IF DLEN OF OLOMAS = 01, MOVE NEWDAT OF OLOMAS TO X1.
  IF DLEN OF OLOMAS = 02, MOVE NEWDAT OF OLOMAS TO X2.
  IF DLEN OF OLOMAS = 03, MOVE NEWDAT OF OLOMAS TO X3.
  IF DLEN OF OLOMAS = 04, MOVE NEWDAT OF OLOMAS TO X4.
  IF DLEN OF OLOMAS = 05, MOVE NEWDAT OF OLOMAS TO X5.
  IF DLEN OF OLOMAS = 06, MOVE NEWDAT OF OLOMAS TO X6.
  IF DLEN OF OLOMAS = 07, MOVE NEWDAT OF OLOMAS TO X7.
  IF DLEN OF OLOMAS = 08, MOVE NEWDAT OF OLOMAS TO X8.
  IF DLEN OF OLOMAS = 09, MOVE NEWDAT OF OLOMAS TO X9.
  IF DLEN OF OLOMAS = 10, MOVE NEWDAT OF OLOMAS TO X10.
  IF DLEN OF OLOMAS = 11, MOVE NEWDAT OF OLOMAS TO X11.
  IF DLEN OF OLOMAS = 12, MOVE NEWDAT OF OLOMAS TO X12.
  IF DLEN OF OLOMAS = 13, MOVE NEWDAT OF OLOMAS TO X13.
  IF DLEN OF OLOMAS = 14, MOVE NEWDAT OF OLOMAS TO X14.
  IF DLEN OF OLOMAS = 15, MOVE NEWDAT OF OLOMAS TO X15.
  IF DLEN OF OLOMAS = 16, MOVE NEWDAT OF OLOMAS TO X16.
  IF DLEN OF OLOMAS = 17, MOVE NEWDAT OF OLOMAS TO X17.
  IF DLEN OF OLOMAS = 18, MOVE NEWDAT OF OLOMAS TO X18.
  IF DLEN OF OLOMAS = 19, MOVE NEWDAT OF OLOMAS TO X19.
  IF DLEN OF OLOMAS = 20, MOVE NEWDAT OF OLOMAS TO X20.
  IF DLEN OF OLOMAS = 21, MOVE NEWDAT OF OLOMAS TO X21.
  IF DLEN OF OLOMAS = 22, MOVE NEWDAT OF OLOMAS TO X22.
  IF DLEN OF OLOMAS = 23, MOVE NEWDAT OF OLOMAS TO X23.
  IF DLEN OF OLOMAS = 24, MOVE NEWDAT OF OLOMAS TO X24.
  IF DLEN OF OLOMAS = 25, MOVE NEWDAT OF OLOMAS TO X25.
  IF DLEN OF OLOMAS < NU1 OR DLEN OF OLOMAS > 25,
     MOVE ALFAX TO ERRSW,
     MOVE NU3 TO ERRCODE OF OLOMAS.
EXAM-CHANGE.
IF RNUM OF OLOMAS = 000
  MOVE ALFAY TO RN Sw
  PERFORM PTR00U.

B-54
IF RNUM OF ULOMAS = 001
   MOVE ALFAY TO RNSW
   PERFORM PTR001.

IF RNUM OF ULOMAS = 002
   MOVE ALFAY TO RNSW
   PERFORM PTR002.

IF RNUM OF ULOMAS = 003
   MOVE ALFAY TO RNSW
   PERFORM PTR003.

IF RNUM OF ULOMAS = 058
   MOVE ALFAY TO RNSW
   PERFORM PTR058.

IF RNUM OF ULOMAS = 076
   MOVE ALFAY TO RNSW
   PERFORM PTR076.

IF RNUM OF ULOMAS = 198
   MOVE ALFAY TO RNSW
   PERFORM PTR198.

IF RNUM OF ULOMAS = 042
   MOVE ALFAY TO RNSW
   PERFORM PTR042.

IF RNUM OF ULOMAS = 082
   MOVE ALFAY TO RNSW
   PERFORM PTR082.

IF RNUM OF ULOMAS = 002
   MOVE ALFAY TO RNSW
   PERFORM PTR002.

IF RNUM OF ULOMAS = 001
   MOVE ALFAY TO RNSW
   PERFORM PTR001.

IF RNUM OF ULOMAS = 158
   MOVE ALFAY TO RNSW
   PERFORM PTR158.

IF RNUM OF ULOMAS = 192
   MOVE ALFAY TO RNSW
   PERFORM PTR192.

IF RNUM OF ULOMAS = 146
   MOVE ALFAY TO RNSW
   PERFORM PTR146.

IF RNUM OF ULOMAS = 186
   MOVE ALFAY TO RNSW
   PERFORM PTR186.

IF RNUM OF ULOMAS = 090
   MOVE ALFAY TO RNSW
   PERFORM PTR090.

IF RNUM OF ULOMAS = 114
   MOVE ALFAY TO RNSW
   PERFORM PTR114.

IF RNUM OF ULOMAS = 121
   MOVE ALFAY TO RNSW
   PERFORM PTR121.

IF RNUM OF ULOMAS = 189
   MOVE ALFAY TO RNSW
   PERFORM PTR189.
IF RNUM OF OLDMAS = 094,
    MOVE ALFAY TO RNSW,
    PERFORM PTR094.
IF RNUM OF OLDMAS = 183,
    MOVE ALFAY TO RNSW,
    PERFORM PTR183.
IF RNUM OF OLDMAS = 072,
    MOVE ALFAY TO RNSW,
    PERFORM PTR072.
IF RNUM OF OLDMAS = 047,
    MOVE ALFAY TO RNSW,
    PERFORM PTR047.
IF RNUM OF OLDMAS = 134,
    MOVE ALFAY TO RNSW,
    PERFORM PTR134.
IF RNUM OF OLDMAS = 171,
    MOVE ALFAY TO RNSW,
    PERFORM PTR171.
IF RNUM OF OLDMAS = 180,
    MOVE ALFAY TO RNSW,
    PERFORM PTR180.
IF RNUM OF OLDMAS = 127,
    MOVE ALFAY TO RNSW,
    PERFORM PTR127.
IF RNUM OF OLDMAS = 055,
    MOVE ALFAY TO RNSW,
    PERFORM PTR055.
IF RNUM OF OLDMAS = 098,
    MOVE ALFAY TO RNSW,
    PERFORM PTR098.
IF RNUM OF OLDMAS = 105,
    MOVE ALFAY TO RNSW,
    PERFORM PTR105.
IF RNUM OF OLDMAS = 164,
    MOVE ALFAY TO RNSW,
    PERFORM PTR164.
IF RNUM OF OLDMAS = 203,
    MOVE ALFAY TO RNSW,
    PERFORM PTR203.
IF RNUM OF OLDMAS = 208,
    MOVE ALFAY TO RNSW,
    PERFORM PTR208.
IF RNUM OF OLDMAS = 212,
    MOVE ALFAY TO RNSW,
    PERFORM PTR212.
IF RNUM OF OLDMAS = 219,
    MOVE ALFAY TO RNSW,
    PERFORM PTR219.
IF RNUM OF OLDMAS = 250,
    MOVE ALFAY TO RNSW,
    PERFORM PTR250.
IF RNUM OF OLDMAS = 277,
    MOVE ALFAY TO RNSW,
    PERFORM PTR277.
IF RNUM OF OLDMAS = 234,
    MOVE ALFAY TO RNSW,
    PERFORM PTR234.
IF RNUM OF OLDMAS = 320,
    MOVE ALFAY TO RNSW,
    PERFORM PTR320.
IF NOT RNSW = ALFAY,
    MOVE ALFAX TO ERRSW,
    MOVE NU1 TO ERCODE,
    MOVE ALFAY TO RNSW.

DIST-CHK.
MOVE ZEROS TO ALLSWS.
MOVE NU1 TO BTA, BTB.
IF RNUM OF OLDMAS = 000, PERFORM STA000 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 042, PERFORM STA042 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 047, PERFORM STA047 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 055, PERFORM STA055 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 058, PERFORM STA058 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 072, PERFORM STA072 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 076, PERFORM STA076 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 082, PERFORM STA082 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 090, PERFORM STA090 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 094, PERFORM STA094 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 114, PERFORM STA114 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 121, PERFORM STA121 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 127, PERFORM STA127 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 134, PERFORM STA134-146 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 146, PERFORM STA134-146 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 158, PERFORM STA158 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 180, PERFORM STA180 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 183, PERFORM STA183 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 186, PERFORM STA186 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 189, PERFORM STA189 VARYING BTA
    FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 192, PERFORM STA192 VARYING BTA FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 198, PERFORM STA198 VARYING BTA FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 234, PERFORM STA234 VARYING BTA FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
ELSE IF RNUM OF OLDMAS = 320, PERFORM STA320 VARYING BTA FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM.

PTR000.
IF CNUM OF OLDMAS = 002, PERFORM EC002.
IF CNUM OF OLDMAS = 003, PERFORM EC003.
IF CNUM OF OLDMAS = 004, PERFORM EC004.
IF CNUM OF OLDMAS = 005, PERFORM EC005.
IF CNUM OF OLDMAS = 006, PERFORM EC006.
IF CNUM OF OLDMAS = 007, PERFORM EC007.
IF CNUM OF OLDMAS = 008, PERFORM EC008.
IF CNUM OF OLDMAS = 011, PERFORM EC011.
IF CNUM OF OLDMAS = 012, PERFORM EC012.
IF CNUM OF OLDMAS = 013, PERFORM EC013.
IF CNUM OF OLDMAS = 014, PERFORM EC014.
IF CNUM OF OLDMAS = 015, PERFORM EC015.
IF CNUM OF OLDMAS = 016, PERFORM EC016.
IF CNUM OF OLDMAS = 017, PERFORM EC017.
IF CNUM OF OLDMAS = 018, PERFORM EC018.
IF CNUM OF OLDMAS = 019, PERFORM EC019.
IF CNUM OF OLDMAS = 021, PERFORM EC021.
IF CNUM OF OLDMAS = 023, PERFORM EC023.
IF CNUM OF OLDMAS = 024, PERFORM EC024.
IF CNUM OF OLDMAS = 027, PERFORM EC027.
IF CNUM OF OLDMAS = 028, PERFORM EC028.
IF CNUM OF OLDMAS = 029, PERFORM EC029.
IF CNUM OF OLDMAS = 030, PERFORM EC030.
IF CNUM OF OLDMAS = 031, PERFORM EC031.
IF CNUM OF OLDMAS = 033, PERFORM EC033.
IF CNUM OF OLDMAS = 034, PERFORM EC034.
IF CNUM OF OLDMAS = 266, PERFORM EC266.
IF CNUM OF OLDMAS = 267, PERFORM EC267.
IF CNUM OF OLDMAS = 301, PERFORM EC301.
IF CNUM OF OLDMAS = 302, PERFORM EC302.

PTR001. EXIT.
PTR002. EXIT.
PTR003. EXIT.
PTR042.
IF CNUM OF OLDMAS = 254, PERFORM EC254.
IF CNUM OF OLDMAS = 043, PERFORM EC043.
IF CNUM OF OLDMAS = 044, PERFORM EC044.
IF CNUM OF OLDMAS = 045, PERFORM EC045.
IF CNUM OF OLDMAS = 046, PERFORM EC046.
IF CNUM OF OLDMAS = 038, PERFORM EC038.
PTR047.
IF CNUM OF OLDMAS = 048, PERFORM EC048.
IF CNUM OF OLDMAS = 043, PERFORM EC043.
IF CNUM OF OLDMAS = 254, PERFORM EC254.
IF CNUM OF OLDMAS = 049, PERFORM EC049.
IF CNUM OF OLDMAS = 050, PERFORM EC050.
IF CNUM OF ULDMAS = 051, PERFORM EC051.
IF CNUM OF ULDMAS = 255, PERFORM EC255.
IF CNUM OF ULDMAS = 266, PERFORM EC266.

PTR055.
IF CNUM OF OLDMAS = 043, PERFORM EC043.
IF CNUM OF OLDMAS = 254, PERFORM EC254.
IF CNUM OF OLDMAS = 056, PERFORM EC056.
IF CNUM OF OLDMAS = 057, PERFORM EC057.

PTR058.
IF CNUM OF OLDMAS = 059, PERFORM EC059.
IF CNUM OF OLDMAS = 060, PERFORM EC060.
IF CNUM OF OLDMAS = 063, PERFORM EC063.
IF CNUM OF OLDMAS = 064, PERFORM EC064.
IF CNUM OF OLDMAS = 065, PERFORM EC065.
IF CNUM OF OLDMAS = 066, PERFORM EC066.
IF CNUM OF OLDMAS = 067, PERFORM EC067.
IF CNUM OF OLDMAS = 068, PERFORM EC068.
IF CNUM OF OLDMAS = 069, PERFORM EC069.
IF CNUM OF OLDMAS = 070, PERFORM EC070.
IF CNUM OF OLDMAS = 071, PERFORM EC071.

PTR072.
IF CNUM OF OLDMAS = 059, PERFORM EC059.
IF CNUM OF OLDMAS = 060, PERFORM EC060.
IF CNUM OF OLDMAS = 073, PERFORM EC073.
IF CNUM OF OLDMAS = 074, PERFORM EC074.
IF CNUM OF OLDMAS = 075, PERFORM EC075.

PTR076.
IF CNUM OF OLDMAS = 059, PERFORM EC059.
IF CNUM OF OLDMAS = 060, PERFORM EC060.
IF CNUM OF OLDMAS = 077, PERFORM EC077.
IF CNUM OF OLDMAS = 078, PERFORM EC078.
IF CNUM OF OLDMAS = 079, PERFORM EC079.
IF CNUM OF OLDMAS = 080, PERFORM EC080.

PTR082.
IF CNUM OF OLDMAS = 059, PERFORM EC059.
IF CNUM OF OLDMAS = 060, PERFORM EC060.
IF CNUM OF OLDMAS = 083, PERFORM EC083.
IF CNUM OF OLDMAS = 084, PERFORM EC084.
IF CNUM OF OLDMAS = 085, PERFORM EC085.
IF CNUM OF OLDMAS = 086, PERFORM EC086.
IF CNUM OF OLDMAS = 087, PERFORM EC087.
IF CNUM OF OLDMAS = 088, PERFORM EC088.
IF CNUM OF OLDMAS = 089, PERFORM EC089.

PTR090.
IF CNUM OF OLDMAS = 256, PERFORM EC256.
IF CNUM OF OLDMAS = 091, PERFORM EC091.
IF CNUM OF OLDMAS = 092, PERFORM EC092.
IF CNUM OF OLDMAS = 093, PERFORM EC093.
IF CNUM OF OLDMAS = 096, PERFORM EC096.
IF CNUM OF OLDMAS = 097, PERFORM EC097.
IF CNUM OF OLDMAS = 304, PERFORM EC304.

PTR094.
IF CNUM OF OLDMAS = 256, PERFORM EC256.
IF CNUM OF OLDMAS = 095, PERFORM EC095.
IF CNUM OF OLDMAS = 126, PERFORM EC126.
IF CNUM OF OLDMAS = 132, PERFORM EC132.

B-59
PTR098. EXIT.
PTR105. EXIT.
PTR114.
IF CNUM OF OLUMAS = 115, PERFORM EC115.
IF CNUM OF OLUMAS = 116, PERFORM EC116.
IF CNUM OF OLUMAS = 117, PERFORM EC117.
IF CNUM OF OLUMAS = 118, PERFORM EC118.
IF CNUM OF OLUMAS = 257, PERFORM EC257.
IF CNUM OF OLUMAS = 120, PERFORM EC120.
IF CNUM OF OLUMAS = 307, PERFORM EC307.
PTR121.
IF CNUM OF OLUMAS = 122, PERFORM EC122.
IF CNUM OF OLUMAS = 123, PERFORM EC123.
IF CNUM OF OLUMAS = 124, PERFORM EC124.
IF CNUM OF OLUMAS = 125, PERFORM EC125.
IF CNUM OF OLUMAS = 258, PERFORM EC258.
PTR127.
IF CNUM OF OLUMAS = 128, PERFORM EC128.
IF CNUM OF OLUMAS = 129, PERFORM EC129.
IF CNUM OF OLUMAS = 130, PERFORM EC130.
IF CNUM OF OLUMAS = 131, PERFORM EC131.
IF CNUM OF OLUMAS = 259, PERFORM EC259.
IF CNUM OF OLUMAS = 133, PERFORM EC133.
PTR134.
IF CNUM OF OLUMAS = 147, PERFORM EC147.
IF CNUM OF OLUMAS = 148, PERFORM EC148.
IF CNUM OF OLUMAS = 150, PERFORM EC150.
IF CNUM OF OLUMAS = 151, PERFORM EC151.
IF CNUM OF OLUMAS = 152, PERFORM EC152.
IF CNUM OF OLUMAS = 153, PERFORM EC153.
IF CNUM OF OLUMAS = 154, PERFORM EC154.
IF CNUM OF OLUMAS = 156, PERFORM EC156.
IF CNUM OF OLUMAS = 157, PERFORM EC157.
IF CNUM OF OLUMAS = 272, PERFORM EC272.
IF CNUM OF OLUMAS = 309, PERFORM EC309.
PTR146.
IF CNUM OF OLUMAS = 147, PERFORM EC147.
IF CNUM OF OLUMAS = 148, PERFORM EC148.
IF CNUM OF OLUMAS = 150, PERFORM EC150.
IF CNUM OF OLUMAS = 151, PERFORM EC151.
IF CNUM OF OLUMAS = 152, PERFORM EC152.
IF CNUM OF OLUMAS = 153, PERFORM EC153.
IF CNUM OF OLUMAS = 154, PERFORM EC154.
IF CNUM OF OLUMAS = 156, PERFORM EC156.
IF CNUM OF OLUMAS = 157, PERFORM EC157.
IF CNUM OF OLUMAS = 272, PERFORM EC272.
IF CNUM OF OLUMAS = 309, PERFORM EC309.
PTR158.
IF CNUM OF OLUMAS = 159, PERFORM EC159.
IF CNUM OF OLUMAS = 161, PERFORM EC161.
IF CNUM OF OLUMAS = 162, PERFORM EC162.
IF CNUM OF OLUMAS = 163, PERFORM EC163.
PTR164. EXIT.
PTR171. EXIT.
PTR180.
  IF CNUM OF UDOMAS = 101, PERFORM EC181.
  IF CNUM OF UDOMAS = 102, PERFORM EC182.
  IF CNUM OF UDOMAS = 261, PERFORM EC261.
  IF CNUM OF UDOMAS = 312, PERFORM EC312.

PTR183.
  IF CNUM OF ODOMAS = 311, PERFORM EC311.
  IF CNUM OF ODOMAS = 185, PERFORM EC185.

PTR186.
  IF CNUM OF ODOMAS = 187, PERFORM EC187.
  IF CNUM OF ODOMAS = 188, PERFORM EC188.

PTR189.
  IF CNUM OF ODOMAS = 190, PERFORM EC190.
  IF CNUM OF ODOMAS = 191, PERFORM EC191.

PTR192.
  IF CNUM OF ODOMAS = 193, PERFORM EC193.
  IF CNUM OF ODOMAS = 195, PERFORM EC195.
  IF CNUM OF ODOMAS = 196, PERFORM EC196.
  IF CNUM OF ODOMAS = 197, PERFORM EC197.

PTR198.
  IF CNUM OF ODOMAS = 199, PERFORM EC199.
  IF CNUM OF ODOMAS = 200, PERFORM EC200.
  IF CNUM OF ODOMAS = 201, PERFORM EC201.
  IF CNUM OF ODOMAS = 202, PERFORM EC202.

PTR203. EXIT.

PTR206. EXIT.

PTR212. EXIT.

PTR219. EXIT.

PTR234.
  IF CNUM OF ODOMAS = 235, PERFORM EC235.
  IF CNUM OF ODOMAS = 237, PERFORM EC237.
  IF CNUM OF ODOMAS = 238, PERFORM EC238.
  IF CNUM OF ODOMAS = 239, PERFORM EC239.

PTR250. EXIT.

PTR277. EXIT.

PTR320.
  IF CNUM OF ODOMAS = 321, PERFORM EC321.
  IF CNUM OF ODOMAS = 322, PERFORM EC322.
  IF CNUM OF ODOMAS = 323, PERFORM EC323.
  IF CNUM OF ODOMAS = 324, PERFORM EC324.

STA000.
  MOVE ALFAM TO CCSR OF RECTAB (BTA).
  IF CNUM OF RECTAB (BTA) = 002, ADD NU1 TO S002.
  IF CNUM OF RECTAB (BTA) = 003, ADD NU1 TO S003.
  IF CNUM OF RECTAB (BTA) = 004, ADD NU1 TO S004.
  IF CNUM OF RECTAB (BTA) = 006, ADD NU1 TO S006.
  IF CNUM OF RECTAB (BTA) = 007, ADD NU1 TO S007.
  IF CNUM OF RECTAB (BTA) = 008, ADD NU1 TO S008.
  IF BTA = NMM PERFORM CHK000.

STA042.
  MOVE ALFAM TO CCSR OF RECTAB (BTA).
  IF CNUM OF RECTAB (BTA) = 043, ADD NU1 TO S002.
  IF BTA = NMM PERFORM CHK042.

B-61
STA047.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 043, ADD NU1 TO S002.
IF BTA = NMM PERFORM CHK047.

STA055.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 043, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 056, ADD NU1 TO S003.
IF CNUM OF RECTAB (BTA) = 254, ADD NU1 TO S004.
IF BTA = NMM PERFORM CHK055.

STA058.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 059, ADD NU1 TO S002.
IF BTA = NMM PERFORM CHK058.

STA072.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 059, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 073, ADD NU1 TO S003.
IF BTA = NMM PERFORM CHK072.

STA076.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 059, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 077, ADD NU1 TO S003.
IF CNUM OF RECTAB (BTA) = 079, ADD NU1 TO S004.
IF BTA = NMM PERFORM CHK076.

STA082.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 059, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 083, ADD NU1 TO S003.
IF CNUM OF RECTAB (BTA) = 085, ADD NU1 TO S004.
IF BTA = NMM PERFORM CHK082.

STA090.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 256, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 093, ADD NU1 TO S003.
IF CNUM OF RECTAB (BTA) = 304, ADD NU1 TO S004.
IF BTA = NMM PERFORM CHK090.

STA094.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 256, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 095, ADD NU1 TO S003.
IF BTA = NMM PERFORM CHK094.

STA114.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 115, ADD NU1 TO S002.
IF BTA = NMM PERFORM CHK114.

STA121.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 122, ADD NU1 TO S002.
IF BTA = NMM PERFORM CHK121.

STA127.
MOVE ALFA1 TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 128, ADD NU1 TO S002.
IF BTA = NMM PERFORM CHK127.
STA134-146.
MOVE ALFAM TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 147, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 150, ADD NU1 TO S003.
IF BTA = NMM PERFORM CHK134-146.

STA158.
MOVE ALFAM TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 161, ADD NU1 TO S002.
IF BTA = NMM PERFORM CHK158.

STA180.
MOVE ALFAM TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 181, ADD NU1 TO S002.
IF BTA = NMM PERFORM CHK180.

STA183.
MOVE ALFAM TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 311, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 185, ADD NU1 TO S003.
IF BTA = NMM PERFORM CHK183.

STA186.
MOVE ALFAM TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 187, ADD NU1 TO S002.
IF BTA = NMM PERFORM CHK186.

STA189.
MOVE ALFAM TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 190, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 191, ADD NU1 TO S003.
IF BTA = NMM PERFORM CHK189.

STA192.
MOVE ALFAM TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 193, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 196, ADD NU1 TO S003.
IF CNUM OF RECTAB (BTA) = 197, ADD NU1 TO S004.
IF BTA = NMM PERFORM CHK192.

STA198.
MOVE ALFAM TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 199, ADD NU1 TO S002.
IF BTA = NMM PERFORM CHK198.

STA234.
MOVE ALFAM TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 235, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 237, ADD NU1 TO S003.
IF BTA = NMM PERFORM CHK234.

STA320.
MOVE ALFAM TO CCSR OF RECTAB (BTA).
IF CNUM OF RECTAB (BTA) = 321, ADD NU1 TO S002.
IF CNUM OF RECTAB (BTA) = 323, ADD NU1 TO S003.
IF BTA = NMM PERFORM CHK320.

CHK000.
IF (NU1 = S002 AND S003 AND S004 AND S006 AND S007 AND S008),
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU2 TO ERRCODE OF OLDMAS,
PERFORM WRITE-LIST.
CHK042.
  IF (NU1 = S002 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK047.
  IF (NU1 = S002 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK055.
  IF (NU1 = S002 AND S003 AND S004 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK058.
  IF (NU1 = S002)
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK072.
  IF (NU1 = S002 AND S003 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK076.
  IF (NU1 = S002 AND S003 AND S004 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK082.
  IF (NU1 = S002 AND S003 AND S004 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK090.
  IF (NU1 = S002 AND S003 AND S004 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.
CHK094.
IF (NU1 = S002 AND S003 )
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU2 TO ERRCODE OF OLDMAS
PERFORM WRITE-LIST.

CHK114.
IF (NU1 = S002 )
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU2 TO ERRCODE OF OLDMAS
PERFORM WRITE-LIST.

CHK121.
IF (NU1 = S002 )
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU2 TO ERRCODE OF OLDMAS
PERFORM WRITE-LIST.

CHK127.
IF (NU1 = S002 )
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU2 TO ERRCODE OF OLDMAS
PERFORM WRITE-LIST.

CHK134-146.
IF (NU1 = S002 AND S003 )
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU2 TO ERRCODE OF OLDMAS
PERFORM WRITE-LIST.

CHK158.
IF (NU1 = S002 )
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU2 TO ERRCODE OF OLDMAS
PERFORM WRITE-LIST.

CHK180.
IF (NU1 = S002 )
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU2 TO ERRCODE OF OLDMAS
PERFORM WRITE-LIST.

CHK183.
IF (NU1 = S002 AND S003 )
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU2 TO ERRCODE OF OLDMAS
PERFORM WRITE-LIST.
CHK186.
  IF (NU1 = S002 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK189.
  IF (NU1 = S002 AND S003 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK192.
  IF (NU1 = S002 AND S003 AND S004 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK198.
  IF (NU1 = S002 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK234.
  IF (NU1 = S002 AND S003 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

CHK320.
  IF (NU1 = S002 AND S003 )
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU2 TO ERRCODE OF OLDMAS
    PERFORM WRITE-LIST.

EC002.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (X1 = ALFAC OR ALFAD OR ALFAH OR ALFAI OR ALFAM
      OR ALFAP OR ALFAT OR ALFAW OR ALFAX)
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU5 TO ERRCODE OF OLDMAS.
  PERFORM ENDEDT.
EC003.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X1 = ALFAC OR ALFAU )
  MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
  MOVE ALFAX TO ERRSW
  MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
EC004.
MOVE 99 TO TYPDAT OF OLDMAS.
IF NEWDAT OF OLDMAS = "USGS"
  MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
  MOVE ALFAX TO ERRSW
  MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
EC005.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (NEWDAT OF OLDMAS > XRLNK10)
  MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
  MOVE ALFAX TO ERRSW
  MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
* EC006 IS NUMERIC -- RESTON PICTURE IS 99
EC006.
MOVE 00 TO TYPDAT OF OLDMAS.
MOVE X2 TO N2.
IF N2 = 04
  MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
  MOVE ALFAX TO ERRSW
  MOVE NU4 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
* EC007 IS NUMERIC -- RESTON PICTURE IS 99
EC007.
MOVE 00 TO TYPDAT OF OLDMAS.
MOVE X2 TO N2.
IF N2 = 04
  MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
  MOVE ALFAX TO ERRSW
  MOVE NU6 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
* EC008 IS NUMERIC -- RESTON PICTURE IS 999
EC008.
MOVE 00 TO TYPDAT OF OLDMAS.
MOVE X3 TO N3.
IF ( N3 = NU1 OR NU3 OR NU5 OR NU7 OR NU9 OR NU11
  OR NU13 OR NU15 OR NU17 OR NU19 OR NU21 OR NU23 OR
  NU25 OR NU27)
  MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
  MOVE ALFAX TO ERRSW
  MOVE NU7 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
EC011.
MOVE 99 TO TYPDAT OF OLDMAS.
   IF (XI = ALFAS OR ALFAF OR ALFAT OR ALFAM)
   MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
   MOVE ALFAX TO ERRSW
   MOVE NU5 TO ERRCODE OF OLDMAS.
   PERFORM ENDEDT.
EC012.
MOVE 99 TO TYPDAT OF OLDMAS.
   MOVE NEWDAT OF OLDMAS TO CHKSIT.
   IF CCSR OF OLDMAS = ALFAM,
   MOVE ALFAR TO CCSR OF OLDMAS.
   IF (LC1 = ALFAA OR ALFAB OR ALFAC OR ALFAD)
   MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
   MOVE NU8 TO ERRCODE OF OLDMAS
   MOVE ALFAX TO ERRSW.
   IF LC8 = NBLNK1 MOVE ALFAU TO LC8.
   IF LC9 = NBLNK1, MOVE ALFAU TO LC9.
   IF LC10 = NBLNK1, MOVE ALFAU TO LC10.
   IF (ALFAU = LC8 OR LC9 OR LC10), MOVE NBLNK1 TO LC11.
   MOVE SPACES TO RESTLC, NEWDAT OF OLDMAS.
   PERFORM CLCSIT.
   MOVE CHKSIT TO NEWDAT OF OLDMAS.
   PERFORM ENDEDT.
EC013.
   MOVE 99 TO TYPDAT OF OLDMAS.
   PERFORM ENDEDT.
EC014.
   MOVE 99 TO TYPDAT OF OLDMAS.
   PERFORM ENDEDT.
* EC015 IS NUMERIC -- RESTON PICTURE IS 999999
EC015.
   MOVE 00 TO TYPDAT OF OLDMAS.
   PERFORM ENDEDT.
* EC016 IS NUMERIC -- RESTON PICTURE IS 999999V99
EC016.
   MOVE 02 TO TYPDAT OF OLDMAS.
   MOVE NEWDAT OF OLDMAS TO ALFNUM.
   PERFORM ALFTONUM.
   MOVE NUMER TO NUM52.
   IF (NUM52 < LOALT OR NUM52 > HIALT)
      MOVE ALFAX TO ERRSW
      MOVE NU10 TO ERRCODE OF OLDMAS
   ELSE
      MOVE NU0 TO ERRCODE OF OLDMAS
      MOVE NUM52 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
EC017.
   MOVE 99 TO TYPDAT OF OLDMAS.
   IF (XI = ALFAA OR ALFAL OR ALFAM)
      MOVE NU0 TO ERRCODE OF OLDMAS
   PERFORM ENDEDT.
ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

EC018.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.

EC019.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAE OR ALFAF
    OR ALFAH OR ALFAM OR ALFAO OR ALFAP OR ALFAS OR ALFAT OR ALFAU OR ALFAV OR ALFAW)
    MOVE ALFAX TO ERRSW
    ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

EC020.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.

EC021.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X10 > XBLNK10)
    MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU11 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

EC022.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.

EC023.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAE OR ALFAF
    OR ALFAH OR ALFAM OR ALFAO OR ALFAP OR ALFAS OR ALFAT OR ALFAU OR ALFAV OR ALFAW)
    MOVE ALFAX TO ERRSW
    ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

EC024.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAE OR ALFAF
    OR ALFAH OR ALFAM OR ALFAO OR ALFAP OR ALFAS OR ALFAT OR ALFAU OR ALFAV OR ALFAW)
    MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC025.
   MOVE 99 TO TYPDAT OF OLDMAS.
   IF (XI = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAE
       OR ALFAF OR ALFAH OR ALFAI OR ALFAM OR ALFAN
       OR ALFAT OR ALFAR OR ALFAK OR ALFAS OR ALFAZ)
      MOVE NU0 TO ERRCODE OF OLDMAS
   ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
   EC026.
   MOVE 99 TO TYPDAT OF OLDMAS.
   IF (XI = ALFAA OR ALFAB OR ALFAC OR ALFAD OR
       ALFAE OR ALFAF OR ALFAH OR ALFAI OR ALFAM
       OR ALFAN OR ALFAK OR ALFAS OR ALFAZ)
      MOVE NU0 TO ERRCODE OF OLDMAS
   ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
* EC027 IS NUMERIC -- RESTON PICTURE IS 99999V99
EC027.
   MOVE 02 TO TYPDAT OF OLDMAS.
   MOVE NEWDAT OF OLDMAS TO ALFNUM.
   PERFORM ALFTONUM.
   MOVE NUMER TO NUM52.
   IF (NUM52 > MAXDEP OR NUM52 < NU0)
      MOVE ALFAX TO ERRSW
      MOVE NU12 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
* EC028 IS NUMERIC -- RESTON PICTURE IS 99999V99
EC028.
   MOVE 02 TO TYPDAT OF OLDMAS.
   MOVE NEWDAT OF OLDMAS TO ALFNUM.
   PERFORM ALFTONUM.
   MOVE NUMER TO NUM52.
   IF (NUM52 > MAXDEP OR NUM52 < NU0)
      MOVE ALFAX TO ERRSW
      MOVE NU13 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.

EC029.
   MOVE 99 TO TYPDAT OF OLDMAS.
   IF (XI = ALFAA OR ALFAB OR ALFAC OR ALFAD OR
       ALFAE OR ALFAF OR ALFAH OR ALFAI OR ALFAM OR
       ALFAN OR ALFAK OR ALFAS OR ALFAZ)
      MOVE NU0 TO ERRCODE OF OLDMAS
   ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
* EC030 IS NUMERIC -- RESTON PICTURE IS 99999V99
EC030.
   MOVE 02 TO TYPDAT OF OLDMAS.
   MOVE NEWDAT OF OLDMAS TO ALFNUM.
   PERFORM ALFTONUM.
MOVE NUMER TO SNUM42.
IF NOT (NEGSW = NUO), COMPUTE SNUM42 = -1 * SNUM42.
MOVE SNUM42 TO NEWDAT OF OLDMAS.
PERFORM ENDEDT.

EC031.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X10 > XBLNK10)
    MOVE NUO TO ERRCODE OF OLDMAS
ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU11 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

EC032.
MOVE 99 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.

EC033.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (XI = ALFAA OR ALFAD OR ALFAG OR ALFAL OR ALFAO
    OR ALFAR OR ALFAS OR ALFAZ)
    MOVE NUO TO ERRCODE OF OLDMAS
ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

EC034.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (XI = ALFAA OR ALFAC OR ALFAE OR ALFAG OR ALFAH OR ALFAL
    OR ALFAM OR ALFAR OR ALFAS OR ALFAT OR ALFAV OR ALFAZ)
    MOVE NUO TO ERRCODE OF OLDMAS
ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

EC036.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (XI = ALFAA OR ALFAD OR ALFAG OR ALFAL OR ALFAO
    OR ALFAR OR ALFAS OR ALFAZ)
    MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

EC037.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (XI = ALFAO OR ALFAR OR ALFAS OR ALFAH OR ALFAL
    OR ALFAP OR ALFAR OR ALFAS OR ALFAT OR ALFAV OR ALFAZ)
    MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

EC038.
MOVE 99 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.
EC045.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (XI = ALFAA OR ALFAB OR ALFAC OR ALFAJ OR ALFAP
OR ALFAR OR ALFAS OR ALFAT OR ALFAU OR ALFAZ )
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAA TO ERRSW
MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

EC044.
MOVE 00 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.

EC045.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (XI = ALFAO OR ALFAE OR ALFAG OR ALFAH OR ALFAL
OR ALFAN OR ALFAW OR ALFAZ OR XBLNK1)
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAA TO ERRSW
MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

* EC046 IS NUMERIC -- RESTON PICTURE IS 99999V99

EC046.
MOVE 02 TO TYPDAT OF OLDMAS.
MOVE NEWDAT OF OLDMAS TO ALFNUM.
PERFORM ALFTONUM.
MOVE NUMER TO NUM42.
MOVE NUM42 TO NEWDAT OF OLDMAS.
PERFORM ENDEDT.

EC048.
MOVE 99 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.

EC049.
MOVE 99 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.

EC050.
MOVE 99 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.

EC052.
MOVE 99 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.

* EC053 IS NUMERIC -- RESTON PICTURE IS 99V99999

EC053.
MOVE 05 TO TYPDAT OF OLDMAS.
MOVE NEWDAT OF OLDMAS TO ALFNUM.
PERFORM ALFTONUM.
MOVE NUMER TO NUM25.
MOVE NUM25 TO NEWDAT OF OLDMAS.

EC051.
MOVE 99 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.

EC056.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (XI = ALFAO OR ALFAE OR ALFAG OR ALFAH OR
ALFAL OR ALFAN OR ALFAW OR ALFAZ)
MOVE NU0 TO ERRCODE OF OLDMAS

B-72
ELSE
   MOVE ALFAX TO ERRSW
   MOVE NU5 TO ERRCODE OF OLDMAS.
   PERFORM ENDEDT.

  EC057.
  MOVE 02 TO TYPDAT OF OLDMAS.
  MOVE NEWDAT OF OLDMAS TO ALFNUM.
  PERFORM ALFTONUM.
  MOVE NUMER TO NUM42.
  MOVE NUM42 TO NEWDAT OF OLDMAS.
  PERFORM ENDEDT.

*   EC059 IS NUMERIC -- RESORT PICTURE IS 999

  EC059.
  MOVE 00 TO TYPDAT OF OLDMAS.
  IF (X3 > XBLNK3)
     MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
     MOVE ALFAX TO ERRSW
     MOVE NU14 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.

  EC060.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

  EC063.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

  EC064.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (X1 = ALFAA OR ALFAD OR ALFAG OR ALFAL OR ALFAO OR
    ALFAR OR ALFAS OR ALFAZ OR XBLNK1)
     MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
     MOVE ALFAX TO ERRSW
     MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.

  EC065.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAH OR
    ALFAJ OR ALFAP OR ALFAR OR ALFAT OR ALFAV OR ALFAW OR
    ALFAZ OR XBLNK1)
     MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
     MOVE ALFAX TO ERRSW
     MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.

  EC066.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (X1 = ALFAC OR ALFAB OR ALFAG OR ALFAH OR ALFAP OR ALFAS OR ALFAT OR ALFAV OR ALFAW OR
    ALFAZ OR XBLNK1)
     MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
     MOVE ALFAX TO ERRSW
     MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.

B-73
EC067.
MOVE 99 TO TYPD AT OF OLDMAS.
IF NOT (X1 = ALFAA OR ALFAC OR ALFAJ OR XBLNK)
   MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
   MOVE ALFAX TO ERRSW
   MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEOT.

* EC068 IS NUMERIC -- RESTON PICTURE IS 999

EC068.
MOVE 00 TO TYPDAT OF OLDMAS.
PERFORM ENDEOT.

EC069.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAJ OR ALFAP OR ALFAS OR ALFAZ OR XBLNK)
   MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
   MOVE ALFAX TO ERRSW
   MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEOT.

* EC070 IS NUMERIC -- RESTON PICTURE IS 999

EC070.
MOVE 00 TO TYPDAT OF OLDMAS.
PERFORM ENDEOT.

EC071.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X1 = ALFAC OR ALFAJ OR ALFAD OR ALFAM OR ALFAH OR SOLNK)
   MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
   MOVE ALFAX TO ERRSW
   MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEOT.

* EC073 IS NUMERIC -- RESTON PICTURE IS 99999999

EC073.
MOVE 02 TO TYPDAT OF OLDMAS.
IF (NEWDAT OF OLDMAS = XBLNK5)
   MOVE ALFAX TO ERRSW
   MOVE NU17 TO ERRCODE OF OLDMAS
ELSE
   MOVE NEWDAT OF OLDMAS TO ALFNUM
   PERFORM ALFTONUM
   MOVE NUMER TO NUM52
   MOVE NUM52 TO NEWDAT OF OLDMAS.
PERFORM ENDEOT.

* EC074 IS NUMERIC -- RESTON PICTURE IS 99999999

EC074.
MOVE 02 TO TYPDAT OF OLDMAS.
MOVE NEWDAT OF OLDMAS TO ALFNUM.
PERFORM ALFTONUM.
MOVE NUMER TO NUM52.
MOVE NUM52 TO NEWDAT OF OLDMAS.
PERFORM ENDEOT.

B-74
**EC075 IS NUMERIC -- RESTON PICTURE IS 99999V99**

EC075.

MOVE 02 TO TYPDAT OF OLDMAS.
MOVE NEWDAT OF OLDMAS TO ALFNUM.
PERFORM ALFTONUM.
MOVE NUMER TO NUM52.
MOVE NUM52 TO NEWDAT OF OLDMAS.
PERFORM ENDEDT.

**EC077 IS NUMERIC -- RESTON PICTURE IS 99999V99**

EC077.

MOVE 02 TO TYPDAT OF OLDMAS.
IF (NEWDAT OF OLDMAS = XBLNK5)
MOVE ALFAX TO ERRSW
MOVE NU17 TO ERRCODE OF OLDMAS
ELSE
MOVE NU0 TO ERRCODE OF OLDMAS
MOVE NEWDAT OF OLDMAS TO ALFNUM
PERFORM ALFTONUM
MOVE NUMER TO NUM42
IF (NEGSW > NU0)
COMPUTE SNUM42 = (1 * SNUM42).
IF ERRSW = ALFAM
MOVE SNUM42 TO NEWDAT OF OLDMAS.
PERFORM ENDEDT.

**EC078 IS NUMERIC -- RESTON PICTURE IS 999V99**

EC078.

MOVE 02 TO TYPDAT OF OLDMAS.
MOVE NEWDAT OF OLDMAS TO ALFNUM
PERFORM ALFTONUM
MOVE NUMER TO NUM32
MOVE NUM32 TO NEWDAT OF OLDMAS.
PERFORM ENDEDT.

**EC079 IS NUMERIC -- RESTON PICTURE IS 999V99**

EC079.

MOVE 02 TO TYPDAT OF OLDMAS.
IF (NEWDAT OF OLDMAS = XBLNK5)
MOVE ALFAX TO ERRSW
MOVE NU17 TO ERRCODE OF OLDMAS
ELSE
MOVE NEWDAT OF OLDMAS TO ALFNUM
PERFORM ALFTONUM
MOVE NUMER TO NUM32
MOVE NUM32 TO NEWDAT OF OLDMAS.
PERFORM ENDEDT.

EC080.

MOVE 99 TO TYPDAT OF OLDMAS.
IF (X1 = ALFAB OR ALFAC OR ALFAG OR ALFAI OR ALFAM OR
ALFAP OR ALFAR OR ALFAS OR ALFAT OR ALFAU OR ALFAW OR
ALFAZ OR XBLNK1)
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

B-75
* EC083 IS NUMERIC -- RESTON PICTURE IS 9999V99

    MOVE 02 TO TYPDAT OF OLDMAS.
    MOVE NEWDAT OF OLDMAS TO ALFNUM.
    PERFORM ALFTONUM.
    MOVE NUMER TO NUM52.
    MOVE NUM52 TO NEWDAT OF OLDMAS.
    PERFORM ENDEDT.

* EC084 IS NUMERIC -- RESTON PICTURE IS 9999V99

    MOVE 02 TO TYPDAT OF OLDMAS.
    MOVE NEWDAT OF OLDMAS TO ALFNUM.
    PERFORM ALFTONUM.
    MOVE NUMER TO NUM52.
    MOVE NUM52 TO NEWDAT OF OLDMAS.
    PERFORM ENDEDT.

EC085.

    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (XI = ALFAF OR ALFAL OR ALFAM OR ALFAP OR ALFAR
    OR ALFAS OR ALFAT OR ALFAX OR ALFAZ OR XBLNK1)
    MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

EC086.

    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (XI = ALFAB OR ALFAC OR ALFAG OR ALFAI OR ALFAM OR
    ALFAP OR ALFAR OR ALFAS OR ALFAT OR ALFAZ OR XBLNK1)
    MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

* EC087 IS NUMERIC -- RESTON PICTURE IS 999V99

EC087.

    MOVE 02 TO TYPDAT OF OLDMAS.
    MOVE NEWDAT OF OLDMAS TO ALFNUM.
    PERFORM ALFTONUM.
    MOVE NUMER TO NUM32.
    MOVE NUM32 TO NEWDAT OF OLDMAS.
    PERFORM ENDEDT.

* EC088 IS NUMERIC -- RESTON PICTURE IS 99V999

EC088.

    MOVE 03 TO TYPDAT OF OLDMAS.
    MOVE NEWDAT OF OLDMAS TO ALFNUM.
    PERFORM ALFTONUM.
    MOVE NUMER TO NUM23.
    MOVE NUM23 TO NEWDAT OF OLDMAS.
    PERFORM ENDEDT.
* EC089 IS NUMERIC -- RESTON PICTURE IS 999V99

```
 ECO89.
   MOVE 02 TO TYPDAT OF OLDMAS.
   MOVE NEWDAT OF OLDMAS TO ALFNUM.
   PERFORM ALFTONUM.
   MOVE NUMER TO NUM32.
   MOVE NUM32 TO NEWDAT OF OLDMAS.
   PERFORM ENDEDT.
```

```
 ECO90.
   MOVE 99 TO TYPDAT OF OLDMAS.
   PERFORM ENDEDT.
```

* EC091 IS NUMERIC -- RESTON PICTURE IS 99999V99

```
 ECO91.
   MOVE 02 TO TYPDAT OF OLDMAS.
   MOVE NEWDAT OF OLDMAS TO ALFNUM.
   PERFORM ALFTONUM.
   MOVE NUMER TO NUM52.
   MOVE NUM52 TO NEWDAT OF OLDMAS.
   PERFORM ENDEDT.
```

```
 ECO92.
   MOVE 99 TO TYPDAT OF OLDMAS.
   IF (NEWDAT OF OLDMAS = XBLNK8)
     MOVE ALFAX TO ERRSW
     MOVE NU14 TO ERRCODE OF OLDMAS.
   PERFORM ENDEDT.
```

```
 ECO93.
   IF (X8 > XBLNK8)
     MOVE NU0 TO ERRCODE OF OLDMAS
   ELSE
     MOVE ALFAX TO ERRSW
     MOVE NU11 TO ERRCODE OF OLDMAS.
   PERFORM ENDEDT.
```

```
 ECO94.
   MOVE 99 TO TYPDAT OF OLDMAS.
   IF (NEWDAT OF OLDMAS = ALVM OR ANDR OR ANDR OR ARKS
      OR BLDK OR BLSD OR BLSK OR BRCC OR CLAY OR CLSN OR COAL
      OR COBB OR COSD OR COSC OR CLVM OR CGLM OR CWUN OR DIBS
      OR GLCL OR GNSS OR GRNT OR GRGN OR GRVL OR GRCL OR GRCM
      OR GRDS OR GRSC OR GRCK OR HRPO OR IGNs OR LGNT OR LMSN
      OR LMDM OR LOAM OR LOSS OR MRBL OR MARL OR MRLS OR MMCN
      OR MUCK OR MUD OR MDSN OR RSDM OR RYLT OR ROCK OR RBLK
      OR SAND OR SDCL OR SDGL OR SOST OR SGVC OR SNDS OR SRAN
      OR SHLE OR SILT OR STCL OR SLSN OR SLTS OR RSIL OR RSTN
      OR SOIL OR SYNT OR TILL OR TUF OR XBLNK5)
     MOVE NU0 TO ERRCODE OF OLDMAS
```

B-77
ELSE
  MOVE ALFAX TO ERRSW
  MOVE NU15 TO ERRCODE OF OLDMAS.
  PERFORM ENDEDT.

EC097.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

EC115.
  MOVE 00 TO TYPDAT OF OLDMAS.
  IF (X4 > XBLNK4)
      MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU11 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.

EC116.
  MOVE 00 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

EC117.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

EC118.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (XI = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAF OR ALFAI OR ALFAM OR ALFAO OR ALFAQ OR ALFAS OR ALFAW OR ALFAZ OR XBLNK1)
      MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.

EC120.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (XI = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAE OR ALFAF OR ALFAG OR ALFAH OR ALFAJ OR ALFAK OR ALFAL OR ALFAM OR ALFAZ OR XBLNK1)
      MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.

EC122.
  MOVE 00 TO TYPDAT OF OLDMAS.
  IF (X4 > XBLNK4)
      MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU11 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.

EC123.
  MOVE 00 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.
EC124.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X4 = "USGS" OR XBLNK4)
   MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
   MOVE ALFAX TO ERRSW
   MOVE NU16 TO ERRCODE OF OLDMAS.
   PERFORM ENDEDT.
EC125.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X1 = ALFANA OR ALFANB OR ALFANC OR ALFAND OR ALFANF OR
   ALFANX OR XBLNK1)
   MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
   MOVE ALFAX TO ERRSW
   MOVE NU5 TO ERRCODE OF OLDMAS.
   PERFORM ENDEDT.
* EC126 IS NUMERIC -- RESTON PICTURE IS 99999V99
EC126.
MOVE 00 TO TYPDAT OF OLDMAS.
MOVE NEWDAT OF OLDMAS TO ALFNUM.
PERFORM ALFTONUM.
MOVE NUMER TO NUM52.
MOVE NUM52 TO NEWDAT OF OLDMAS.
   PERFORM ENDEDT.
EC128.
MOVE 00 TO TYPDAT OF OLDMAS.
IF (X4 > XBLNK4)
   MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
   MOVE ALFAX TO ERRSW
   MOVE NU11 TO ERRCODE OF OLDMAS.
   PERFORM ENDEDT.
EC129.
MOVE 00 TO TYPDAT OF OLDMAS.
   PERFORM ENDEDT.
EC130.
MOVE 99 TO TYPDAT OF OLDMAS.
   PERFORM ENDEDT.
EC131.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X1 = ALFANA OR ALFANB OR ALFANC OR ALFAND OR ALFANF OR
   ALFANX OR XBLNK1)
   MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
   MOVE ALFAX TO ERRSW
   MOVE NU5 TO ERRCODE OF OLDMAS.
   PERFORM ENDEDT.
EC132.
MOVE 00 TO TYPDAT OF OLDMAS.
   PERFORM ENDEDT.
EC133.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (X1 = ALFAC OR ALFAE OR ALFAM OR ALFAU OR ALFAZ OR XBLNK1)
    MOVE NUO TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NUS TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

EC147.
  MOVE 00 TO TYPDAT OF OLDMAS.
  IF (X3 > XBLNK3)
    MOVE NUO TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU14 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

EC148.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

* EC150 IS NUMERIC -- RESTON PICTURE IS 99999V99

EC150.
  MOVE 02 TO TYPDAT OF OLDMAS.
  IF (NEWDAT OF OLDMAS > XBLNK5)
    MOVE NU0 TO ERRCODE OF OLDMAS
    MOVE NEWDAT OF OLDMAS TO ALFNUM
    PERFORM ALFTONUM
    MOVE NUMER TO NUM52
    MOVE NUM52 TO NEWDAT OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU20 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

EC151.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (X1 = ALFAA OR ALFAQ OR ALFAU OR ALFAO
      OR ALFAS OR ALFAR OR ALFAZ OR XBLNK1)
    MOVE NUO TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NUS TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

EC152.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (X1 = ALFAB OR ALFAC OR ALFAE OR ALFAM OR ALFAO
      OR ALFAU OR ALFAS OR ALFAR OR ALFAZ OR XBLNK1)
    MOVE NUO TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NUS TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
* EC153 IS NUMERIC -- RESTON PICTURE IS 99999V99
EC153.
  MOVE 02 TO TYPDAT OF OLDMAS.
  MOVE NEWDAT OF OLDMAS TO ALFNUM.
  PERFORM ALFTONUM.
  MOVE NUMER TO NUM52.
  PERFORM ENDEDT.
* EC154 IS NUMERIC -- RESTON PICTURE IS 99999V99
EC154.
  MOVE 02 TO TYPDAT OF OLDMAS.
  MOVE NEWDAT OF OLDMAS TO ALFNUM.
  PERFORM ALFTONUM.
  MOVE NUMER TO NUM52.
  MOVE NUM52 TO NEWDAT OF OLDMAS.
  PERFORM ENDEDT.
EC156.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (XI = ALFAA OR ALFAC OR ALFAE OR ALFAG OR ALFAH OR ALFAL OR ALFAM OR ALFAR OR ALFAS OR ALFAT OR ALFAV OR ALFAZ OR XBLNK1)
  MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
  MOVE ALFAX TO ERRSW
  MOVE NU5 TO ERRCODE OF OLDMAS.
  PERFORM ENDEDT.
* EC157 IS NUMERIC -- RESTON PICTURE IS 9999V9
EC157.
  MOVE 01 TO TYPDAT OF OLDMAS.
  MOVE NEWDAT OF OLDMAS TO ALFNUM.
  PERFORM ALFTONUM.
  MOVE NUMER TO NUM41.
  MOVE NUM41 TO NEWDAT OF OLDMAS.
  PERFORM ENDEDT.
EC159.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.
EC161.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (X10 > XBLNK10)
  MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
  MOVE ALFAX TO ERRSW
  MOVE NU17 TO ERRCODE OF OLDMAS.
  PERFORM ENDEDT.
EC162.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.
EC163.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.
EC181.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (NEWDAT OF OLDMAS > XBLNK10)
  MOVE NU0 TO ERRCODE OF OLDMAS.
ELSE
    MOVE ALFAX TO ERRSW
    MOVE NU17 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.

EC182.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAC OR ALFAD OR ALFAZ OR XBLNK1)
      MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.

EC185.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.

EC187.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X10 > XBLNK10)
      MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU11 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.

EC188.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.

EC190.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X10 > XBLNK10)
      MOVE ALFAX TO ERRSW
    ELSE
      MOVE ALFAX TO ERRSW
      MOVE NU17 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.

EC191.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.

EC193.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X10 = XBLNK10)
      MOVE ALFAX TO ERRSW
      MOVE NU11 TO ERRCODE OF OLDMAS
    ELSE
      MOVE NU0 TO ERRCODE OF OLDMAS.
      IF (X10 > TDYDAT)
        MOVE ALFAX TO ERRSW
        MOVE NU18 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.

EC195.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.

EC196.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (NEWDAT OF OLDMAS = QW1 OR QW2 OR QW3 OR QW4 OR QW5
      OR QW6 OR QW7 OR QW8 OR QW9 OR QW10 OR QW11 OR QW12 OR QW13

B-82
OR QW14 OR QW15 OR QW16 OR QW17 OR QW18 OR QW19)
MOVE NG0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU18 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

* EC197 IS NUMERIC -- RESTON PICTURE IS 999999V9

EC197.
MOVE 01 TO TYPDAT OF OLDMAS.
IF (NEWDAT OF OLDMAS > XBLNK6)
MOVE NEWDAT OF OLDMAS TO ALFNUM
PERFORM ALFTONUM
MOVE NUMER TO NUM61
MOVE NUM61 TO NEWDAT OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU17 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

EC199.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAE
OR ALFAF OR ALFAG OR ALFAH OR ALFAI OR ALFAJ OR ALFAM
OR ALFAR OR ALFAS OR ALFAO OR ALFAP OR ALFAQ
OR ALFAS OR ALFAT OR ALFAU OR ALFAV OR ALFAZ)
MOVE NG0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.

* EC200 IS NUMERIC -- RESTON PICTURE IS 999999V9

EC200.
MOVE 02 TO TYPDAT OF OLDMAS.
MOVE NEWDAT OF OLDMAS TO ALFNUM.
PERFORM ALFTONUM.
MOVE NUMER TO NUM52.
MOVE NUM52 TO NEWDAT OF OLDMAS.
PERFORM ENDEDT.

* EC201 IS NUMERIC -- RESTON PICTURE IS 999999V9

EC201.
MOVE 02 TO TYPDAT OF OLDMAS.
MOVE NEWDAT OF OLDMAS TO ALFNUM.
PERFORM ALFTONUM.
MOVE NUMER TO NUM52.
MOVE NUM52 TO NEWDAT OF OLDMAS.
PERFORM ENDEDT.

EC202.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X1 = ALFAA OR ALFAD OR ALFAQ OR ALFAL OR ALFAM
OR ALFAS OR ALFAO OR ALFAP OR XBLNK1)
MOVE NG0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
EC235.
  MOVE 99 TO TYPDAT OF OLOMAS.
  PERFORM ENDEDT.
* EC237 IS NUMERIC -- RESTON PICTURE IS 9999V99
EC237.
  MOVE 02 TO TYPDAT OF OLOMAS.
  MOVE NEWDAT OF OLOMAS TO ALFNUM.
  PERFORM ALFTONUM.
  MOVE NUMER TO SNUM42.
  MOVE SNUM42 TO NEWDAT OF OLOMAS.
  PERFORM ENDEDT.
EC238.
  MOVE 99 TO TYPDAT OF OLOMAS.
  IF (XI = XBLNK1 OR ALFAA OR ALFAC OR ALFAE OR ALFAG
   OR ALFAH OR ALFAJ OR ALFAK OR ALFAO OR ALFAF OR ALFAH
   OR ALFAO OR ALFAF OR ALFAO OR ALFAM OR ALFAR OR ALFAJ
   OR ALFAO OR ALFAF OR ALFAO OR ALFAX OR ALFAZ)
   MOVE NU0 TO ERRCODE OF OLOMAS
  ELSE
    MOVE ALFAA TO ERRSW
    MOVE NU5 TO ERRCODE OF OLOMAS.
  PERFORM ENDEDT.
EC239.
  MOVE 99 TO TYPDAT OF OLOMAS.
  IF (XI = XRLNK1 OR ALFAA OR ALFAC OR ALFAE OR ALFAG
   OR ALFAH OR ALFAJ OR ALFAK OR ALFAO OR ALFAF OR ALFAH
   OR ALFAO OR ALFAF OR ALFAO OR ALFAM OR ALFAR OR ALFAJ
   OR ALFAO OR ALFAF OR ALFAO OR ALFAX OR ALFAZ)
   MOVE NU0 TO ERRCODE OF OLOMAS
  ELSE
    MOVE ALFAA TO ERRSW
    MOVE NU5 TO ERRCODE OF OLOMAS.
  PERFORM ENDEDT.
EC254.
  MOVE 00 TO TYPDAT OF OLOMAS.
  IF (XI3 > XBLNK3)
    MOVE NU0 TO ERRCODE OF OLOMAS
  ELSE
    MOVE ALFAA TO ERRSW
    MOVE NU14 TO ERRCODE OF OLOMAS.
  PERFORM ENDEDT.
EC255.
  MOVE 00 TO TYPDAT OF OLOMAS.
  PERFORM ENDEDT.
EC256.
  MOVE 00 TO TYPDAT OF OLOMAS.
  IF (XI3 > XBLNK3)
    MOVE NU0 TO ERRCODE OF OLOMAS
  ELSE
    MOVE ALFAA TO ERRSW
    MOVE NU14 TO ERRCODE OF OLOMAS.
  PERFORM ENDEDT.
EC257.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (XI = ALFAY OR XBLNK1)
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NUS TO ERRCODE OF OLDMAS.
  PERFORM ENDEDT.

EC258.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

EC259.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

EC261.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (XI = ALFAF OR ALFAM OR ALFAZ OR XBLNK1)
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NUS TO ERRCODE OF OLDMAS.
  PERFORM ENDEDT.

* EC266 IS NUMERIC -- RESTON PICTURE IS 99999V99

EC266.
  MOVE 02 TO TYPDAT OF OLDMAS.
  MOVE NEWDAT OF OLDMAS TO ALFNUM.
  PERFORM ALFTONUM.
  MOVE NUMER TO SNUM32.
  MOVE ZEROS TO NUMER.
  IF (NEGSW > NU0)
    COMPUTE SNUM32 = (-1 * SNUM32)
    MOVE SNUM32 TO NEWDAT OF OLDMAS.
  PERFORM ENDEDT.

EC267.
  MOVE 99 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

EC268.
  MOVE 00 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

* EC272 IS NUMERIC -- RESTON PICTURE IS 99999V99

EC272.
  MOVE 02 TO TYPDAT OF OLDMAS.
  MOVE NEWDAT OF OLDMAS TO ALFNUM.
  PERFORM ALFTONUM.
  MOVE NUMER TO NUM52.
  PERFORM ENDEDT.

EC301.
  MOVE 99 TO TYPDAT OF OLDMAS.
  IF (XI = ALFAA OR ALFAC OR ALFAO OR ALFAE OR ALFAG
      OR ALFAH OR ALFAM OR ALFAO OR ALFAF OR ALFAH OR ALFAS
      OR ALFAT OR ALFAU OR ALFAS OR XBLNK1)
    MOVE NU0 TO ERRCODE OF OLDMAS
  ELSE
    MOVE ALFAX TO ERRSW
    MOVE NUS TO ERRCODE OF OLDMAS.
  PERFORM ENDEDT.

B-85
EC302.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X1 = ALFAA OR ALFAC OR ALFAD OR ALFAE OR ALFAG
OR ALFAM OR ALFAO OR ALFAP OR ALFAR OR ALFAS
OR ALFAT OR ALFAU OR ALFAW OR X8LNK1)
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
EC304.
MOVE 99 TO TYPDAT OF OLDMAS.
IF (X1 = ALFAN OR ALFAP OR ALFAS OR X8LNK1)
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU5 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
EC307.
MOVE 99 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.
EC309.
MOVE 02 TO TYPDAT OF OLDMAS.
MOVE NEWDAT OF OLDMAS TO ALFNUM
PERFORM ALFTONUM
MOVE NUMER TO NUM42
MOVE NUM42 TO NEWDAT OF OLDMAS.
PERFORM ENDEDT.
EC311.
MOVE 00 TO TYPDAT OF OLDMAS.
IF (X3 > X8LNK3)
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU14 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
EC312.
MOVE 00 TO TYPDAT OF OLDMAS.
IF (X3 > X8LNK3)
MOVE NU0 TO ERRCODE OF OLDMAS
ELSE
MOVE ALFAX TO ERRSW
MOVE NU14 TO ERRCODE OF OLDMAS.
PERFORM ENDEDT.
EC321.
MOVE 00 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.
EC322.
MOVE 00 TO TYPDAT OF OLDMAS.
PERFORM ENDEDT.
EC323.
MOVE 02 TO TYPDAT OF OLDMAS.
IF (NEWDAT OF OLDMAS = X8LNK5)
MOVE ALFAX TO ERRSW
MOVE NU17 TO ERRCODE OF OLDMAS
ELSE
  MOVE NEWDAT OF OLDMAS TO ALFNUM
  PERFORM ALFTONUM
  MOVE NUMER TO SNUM42
  IF (NEG5W > NU9)
    COMPUTE SNUM42 = (SNUM42 * -1).
  MOVE SNUM42 TO NEWDAT OF OLDMAS.
  PERFORM ENDEDT.

EC324.
  MOVE 00 TO TYPDAT OF OLDMAS.
  PERFORM ENDEDT.

ENDEDT.
  MOVE ERR5W TO CCSR OF OLDMAS.
  MOVE OLDMAS TO RECTAB (MM).

CLCSIT.
  IF NOT (LC1 = "A" OR "B" OR "C" OR "D"),
    MOVE ALFAU T0 ERR5W.
  IF NOT (LC2 = "H" OR "-"), MOVE ALFAU TO ERR5W.
  IF (LC1 = "A" OR "B") AND (LC3 > 42 OR < 1),
    MOVE ALFAU TO ERR5W.
  IF (LC1 = "C") AND (LC3 > 19 OR < 1),
    MOVE ALFAU TO ERR5W.
  IF (LC1 = "D") AND (LC3 > 24 OR < 1),
    MOVE ALFAU TO ERR5W.
  IF NOT (LC4 = "H" OR "-"), MOVE ALFAU TO ERR5W.
  IF (LC1 = "A" OR "D") AND (LC5 > 32 OR < 1),
    MOVE ALFAU TO ERR5W.
  IF (LC1 = "B") AND (LC5 > 23 OR < 1),
    MOVE ALFAU TO ERR5W.
  IF (LC1 = "C") AND (LC5 > 25 OR < 1),
    MOVE ALFAU TO ERR5W.
  IF NOT (LC6 = " " OR "N" OR "E" OR "W" OR "S"),
    MOVE ALFAU TO ERR5W.
  IF LC7 > 36 OR < 1, MOVE ALFAU TO ERR5W.
  IF NOT (LC8 = ALFAU OR ALFAA OR ALFAB OR ALFAC OR ALFAD),
    MOVE ALFAU TO ERR5W.
  IF NOT (LC9 = ALFAU OR ALFAA OR ALFAB OR ALFAC OR ALFAD),
    MOVE ALFAU TO ERR5W.
  IF NOT (LC10 = ALFAU OR ALFAA OR ALFAB OR ALFAC OR ALFAD),
    MOVE ALFAU TO ERR5W.
  IF NOT (LC11 = " " OR "1" OR "2" OR "3" OR "4" OR
    "5" OR "6" OR "7" OR "8" OR "9"), MOVE ALFAU TO ERR5W.
  IF ERR5W = ALFAU
    MOVE ALFAU TO ERR5W
    MOVE NU9 TO ERRCODE OF OLDMAS.

ALFTONUM.
  PERFORM NEGTST.
  PERFORM UGCNTR VARYING ALN FROM 1 BY 1 UNTIL
    (ADIGIT (ALN) = XBLNK1).
  COMPUTE NUMS = ( 11 - UGCNT ) + 1.
  SET ALN TO DLEN OF OLDMAS.
  PERFORM UGMOVE VARYING NLN FROM 11 BY -1 UNTIL NLN < NUMS.
  EXAMINE NNUM REPLACING ALL " " BY 0.
  MOVE NNUM TO NUMER.
  EXAMINE NUMER REPLACING ALL " " BY 0.

B-87
COMPUTE DECNT = DECNT - 1.
COMPUTE NUMER = NUMER DIVIDED BY (10 ** DECNT).
COMPUTE NUMER = NUMER + .0000001.

NEG1ST.
IF (ADIGIT (1) = "-" ) ADD NU1 TO NEGSW,
MOVE NU0 TO ADIGIT (1).

ALLMOVE.
MOVE ADIGIT (ALN + 1) TO ADIGIT (ALN).

DGCTNTR.
IF (ADIGIT (ALN) = ".") ADD NU1 TO DECSW.
IF NOT (DECSW > NU0)
   ADD NU1 TO DGCTN
ELSE ADD NU1 TO DGCTN, DECNT.

DGMOVE.
SET ALN DOWN BY 1.
MOVE ADIGIT (ALN) TO NDIGIT (NLN).

CLOSEA.
MOVE 1 TO EDFSW.
MOVE ALL ZEROS TO DELTSS.

CLSEA.
IF (KEYIDD = 0000000000000000 )
   MOVE ZEROS TO DELTSS.
   MOVE ZEROS TO KEYIDC.

CLSEAD.
IF (KEYIDC = 0000000000000000)
   MOVE ZEROS TO DELTSS.
   MOVE ZEROS TO KEYIDD.

FINA.
CLOSE CHANGE-FILE, FILEA, FILEB.
GO TO STOP3.

STARTB.
OPEN INPUT CHANGE-FILE.

READ.
READ CHANGE-FILE AT END GO TO CLOSEB.
ADD NU1 TO TFILEB.
MOVE CNGREC TO LLPRT.
WRITE LIST-LINE.
GO TO READ.

CLOSEB.
GO TO STOP3.

STARTC.
DISPLAY "STARTC .".
STOP RUN.

HDPRNT.
ADD 1 TO PGCNT.
MOVE PGCNT TO PNUM OF HDR1.
MOVE 4 TO LNCNT.
WRITE LIST-LINE FROM HDR1.
WRITE LIST-LINE FROM HDR2.
WRITE LIST-LINE FROM HDR3.
WRITE-LIST.
IF LNCNT > 60,
   PERFORM HDPRNT,
ELSE
   ADD 1 TO LNCNT, NUETLN.
   ADD NU1 TO YY.
   MOVE RECTAB (YY) TO OLDMAS.
   MOVE RECTAB (YY) TO PRNTLN.
   IF (ERRSW = ALFAX)
      ADD 1 TO ERNUM,
      PERFORM ERR-MSG
      PERFORM WRTERR
   ELSE
      ADD 1 TO UPNUM
      PERFORM WRTUPD.
   WRITE-LIST-LINE FROM PRNTLN.
   MOVE SPACES TO CC OF PRNTLN.
   IF (YY = NMM OR YY > NMM) PERFORM CLEAR-PROC.
WRTERR.
   MOVE RECTAB (YY) TO AUDREC.
   MOVE ERNUM TO NSEQ OF AUDREC.
   WRITE AUDREC.
WRTUPD.
   MOVE RECTAB (YY) TO RECOUT.
   WRITE RECOUT.
ERR-MSG.
IF ERRCODE OF RECTAB (YY) = NU0 OR " "
   MOVE SPACES TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU1
   MOVE ER-MSG (NU1) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU2
   MOVE ER-MSG (NU2) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU3
   MOVE ER-MSG (NU3) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU4
   MOVE ER-MSG (NU4) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU5
   MOVE ER-MSG (NU5) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU6
   MOVE ER-MSG (NU6) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU7
   MOVE ER-MSG (NU7) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU8
   MOVE ER-MSG (NU8) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU9
   MOVE ER-MSG (NU9) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU10
   MOVE ER-MSG (NU10) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU11
   MOVE ER-MSG (NU11) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU12
   MOVE ER-MSG (NU12) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU13
   MOVE ER-MSG (NU13) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU14
    MOVE EP-MSG (NU14) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU15
    MOVE EP-MSG (NU15) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU16
    MOVE EP-MSG (NU16) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU17
    MOVE EP-MSG (NU17) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU18
    MOVE EP-MSG (NU18) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU19
    MOVE EP-MSG (NU19) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU20
    MOVE EP-MSG (NU20) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU21
    MOVE EP-MSG (NU21) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU22
    MOVE EP-MSG (NU22) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU23
    MOVE EP-MSG (NU23) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU24
    MOVE EP-MSG (NU24) TO ERPRINT OF PRNTLN.
IF ERRCODE OF RECTAB (YY) = NU25
    MOVE EP-MSG (NU25) TO ERPRINT OF PRNTLN.

STOP 1.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "CONTROL INPUT FILE EMPTY".
    STOP RUN.

STOP 2.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
    STOP RUN.

STOP 3.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "TFILEB".
    MOVE INCRDS TO CSTAT.
    MOVE NDATIN TO DATAIN.
    MOVE NFLEIA TO FAUDA.
    MOVE NDETLN TO DRLNS.
    MOVE NFEQA TO FOUT.
    WRITE LIST-LINE FROM JRCNLX.
    DISPLAY "NORMAL JOB TERMINATION".
    CLOSE CARD-FILE, LIST-FILE.
    STOP RUN.

STOP 4.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "SOURCE ID RECORDS MISSING".
    STOP RUN.

STOP 5.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOPS FILEA-UPCHNG EMPTY", "*** CHECK FOR ATTACH OF UPCHNG IN JCL".
    STOP RUN.
STOP6.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "SECOND CARD OF PAIR MISSING".
STOP RUN.

STOP7.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "STOP7 ", ESMG1.
DISPLAY "INPUT STREAM CARD RECORDS".
DISPLAY ESMG2.
DISPLAY "CORRECT SEQUENCE AND RERUN".
STOP RUN.

STOP8.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "ERROR IN RECORD SEQUENCE", "STOP8 AZERRS".
DISPLAY ERRCNG.
DISPLAY "KTEST = ", KTEST, "KEYIDD = ", KEYIDD.
STOP RUN.

STOP9.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "STOP9 ", ESMG1.
DISPLAY "INPUT AZSR-NUMSER FILE ", ESMG2.
DISPLAY "KEYID OF CNGREC = ", KEYID OF CNGREC.
DISPLAY "KEYID = ", KEYID.
STOP RUN.

STOP10.
STOP RUN.

ERROR8.
DISPLAY "***********************".
DISPLAY "ERROR IN RECORD SEQUENCE", "ERROR8 AZERRS".
DISPLAY ERRCNG.
DISPLAY "KTEST = ", KTEST, "KEYIDD = ", KEYIDD.
DISPLAY "***************".
ADD NUS TO LNCNT.

ERROR11.
DISPLAY OLDMAS.
DISPLAY "STOP 11", "RNUM OF CNGREC NOT FOUND".
DISPLAY "R=*** NOT VALID ", RNUM OF CNGREC.
DISPLAY "****** ", CNGREC.
IDENTIFICATION DIVISION.
PROGRAM-ID. UPDATE-1.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. 6600.
OBJECT-COMPUTER. 6600.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
  SELECT CARD-FILE ASSIGN TO "INPUT".
  SELECT LIST-FILE ASSIGN TO "OUTPUT".
  SELECT FILEA ASSIGN TO MASIN.
  SELECT FILEB ASSIGN TO MASOUT.
  SELECT FILEC ASSIGN TO CMAS.
  SELECT CHNGEL ASSIGN TO DCHNG.
DATA DIVISION.
FILE SECTION.
FD CARD-FILE
  LABEL RECORD OMITTED
  DATA RECORD IS CNL-CARD.
  01 CNL-CARD.
     02 CNLXXX.
        03 ID-CNLCN PIC IS XXX.
        03 ID-NUM PIC 999.
     02 DELNUM REDEFINES CNLXXX PIC 9(6).
     02 VSNA PIC X.
     02 ID-OES PIC X(73).
FD LIST-FILE
  LABEL RECORD OMITTED
  DATA RECORD IS LIST-LINE.
  01 LIST-LINE.
     03 CC PIC X.
     03 LLPRT PIC X(132).
FD FILEA
  DATA RECORD IS OLDMAS RECORD CONTAINS 23 TO 327 CHARACTERS
  RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
  CHARACTERS LABEL RECORDS ARE STANDARD.
  01 OLDMAS.
     03 KEYIDM.
        05 KYLATM.
           07 KLD PIC 99.
           07 KLM PIC 99.
           07 KLS PIC 99.
        05 KYLNGM.
           07 KLD PIC 999.
           07 KLGM PIC 99.
           07 KLGS PIC 99.
        05 KYDUPM PIC 99.
     03 RNUM PIC 999.
     03 LNUM PIC 999.
     03 SESAFE PIC X.
     03 RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
     OF OLDMAS.
FD FILEB
DATA RECORD IS NEWMAS RECORD CONTAINS 23 TO 327 CHARACTERS
RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
CHARACTERS LABEL RECORDS ARE STANDARD.
01 NEWMAS.
  03 KEYIDM.
    05 KYLATM.
      07 KLD PIC 99.
      07 KLM PIC 99.
      07 KLS PIC 99.
    05 KYLNGM.
      07 KLGD PIC 999.
      07 KLGX PIC 99.
      07 KLGS PIC 99.
    05 KYDUPM PIC 99.
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
OF OLDMAS.

FD FILEC
DATA RECORD IS CNGMAS RECORD CONTAINS 23 TO 327 CHARACTERS
RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
CHARACTERS LABEL RECORDS ARE OMITTED.
01 CNGMAS.
  03 KEYIDM.
    05 KYLATM.
      07 KLD PIC 99.
      07 KLM PIC 99.
      07 KLS PIC 99.
    05 KYLNGM.
      07 KLGD PIC 999.
      07 KLGX PIC 99.
      07 KLGS PIC 99.
    05 KYDUPM PIC 99.
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
OF OLDMAS.

FD CHNGEL
DATA RECORD IS CNGREC
LABEL RECORDS ARE OMITTED.
01 CNGRFC.
  03 VSN PIC 99999.
  03 VSNA PIC X.
  03 PRD2 PIC 9(5).
  03 KSEQ PIC 9(4).
  03 KEYID.
    05 KEYLAT.
      07 LATDEG PIC 99.
      07 LATMIN PIC 99.
      07 LATSEC PIC 99.

B-93
05 KEYLNG.
  07 LNGDEG PIC 999.
  07 LNGMIN PIC 99.
  07 LNGSEC PIC 99.
05 KEYDUP PIC 99.
  03 CSCDE PIC X.
  03 PNUM PIC 999.
  03 CNUM PIC 999.
  03 CCR PIC X.
  03 CCSR PIC X.
  03 TYPDAT PIC 99.
  03 DLEN PIC 99.
  03 NEWDAT PIC X(45).
  03 NSEQ PIC 9(8).
WORKING-STORAGE SECTION.
  77 ACOUNT PIC 9(6) VALUE 0.
  77 ALA PIC X VALUE "A".
  77 ALD PIC X VALUE "D".
  77 ALFLAT PIC X(6).
  77 ALFLNG PIC X(7).
  77 ALPHA6 PIC X(6).
  77 ALPHA7 PIC X(7).
  77 ALR PIC X VALUE "R".
  77 ALT PIC X VALUE "T".
  77 ALX PIC X VALUE "X".
  77 ALY PIC X VALUE "Y".
  77 ALZ PIC X VALUE "Z".
  77 AT PIC 999 VALUE 0.
  77 BTA PIC 9(4).
  77 BLNK1 PIC X VALUE SPACE.
  77 CMPLAT PIC 9(6).
  77 CMPLNG PIC 9(7).
  77 CNRECN PIC 9(6) VALUE ZEROES.
  77 DELSW PIC X VALUE "Z".
  77 DEM1 PTC X(15) VALUE " DATA ENTRY ".
  77 DEM2 PIC X(10) VALUE "*UNCHECKED".
  77 DYZ PIC 9.
  77 ERMSG1 PIC X(27) VALUE " AZ FORMAT SORTED CHANGES ".
  77 ERMSG2 PIC X(27) VALUE " GW & QW LOCAL IDS UNEQUAL ".
  77 ERRCNT PIC 9(9) VALUE 0.
  77 ERRSW PIC X VALUE "Y".
  77 ESH PIC 99 VALUE 0.
  77 FLCNT PIC 9(6).
  77 FRSW PIC X VALUE "F".
  77 HSW PIC X VALUE "X".
  77 INCRDS PIC 9(6) VALUE 0.
  77 JULD PTC 9(5).
  77 LNCNT PIC 99 VALUE 70.
  77 MSG1 PIC X(25) VALUE "CHECKS OK DROP NOT NEEDED".
  77 MSG2 PIC X(25) VALUE "KEYID NOT = LAT-LNG FLDS".
  77 NDATIN PIC 9(6) VALUE 0.
  77 NDETLN PIC 9(6) VALUE 0.
  77 NFLEIA PIC 9(6) VALUE 0.
  77 NFLEOA PIC 9(6) VALUE 0.
  77 NU1 PIC 9 VALUE 1.
PGCNT PIC 999 VALUE 0.
SEQCHK PIC 9(6) VALUE 0.
TFILEA PIC 9(6) VALUE 0.
TFILEB PIC 9(6) VALUE 0.
TFILEC PIC 9(6) VALUE 0.

**01 WKDATE.**
03 YY PIC 99.
03 MM PIC 99.
03 DY PIC 99.

**01 NAME-MONTH.**
03 JAN PIC XXXX VALUE " JAN".
03 FEB PIC XXXX VALUE " FEB".
03 MAR PIC XXXX VALUE " MAR".
03 APR PIC XXXX VALUE " APR".
03 MAY PIC XXXX VALUE " MAY".
03 JUNE PIC XXXX VALUE " JUNE".
03 JULY PIC XXXX VALUE " JULY".
03 AUG PIC XXXX VALUE " AUG".
03 SEPT PIC XXXX VALUE " SEPT".
03 OCT PIC XXXX VALUE " OCT".
03 NOV PIC XXXX VALUE " NOV".
03 DEC PIC XXXX VALUE " DEC".

**01 MONTH-TABLE REDEFINES NAME-MONTH.**
03 MONTHY PIC XXXX OCCURS 12 TIMES INDEXED BY NMM.

**01 JBCNL.**
03 CC PIC X VALUE "1".
03 PCNL.
05 CNTYPE PIC XXX.
05 GU-CNPL PIC 999.
05 PCNLX PIC X(74).
03 FILLER PIC X(52) VALUE SPACES.

**01 JBCNLX.**
03 CC PIC X VALUE "0".
03 PSTATS PIC X(14) VALUE "RECORD COUNTS".
03 FILLER PIC X(6) VALUE SPACES.
03 XCARD PIC X(8) VALUE "CARDS IN".
03 CSTAT PIC ZZZZZ9.
03 XDRLNS PIC X(20) VALUE " SITES TO BE CHANGED".
03 DRLNS PIC ZZZZZ9.
03 XDATIZN PIC X(15) VALUE " CHANGE RECORDS".
03 DATAIN PIC ZZZZZ9.
03 XFAIN PIC X(19) VALUE " UNCHANGED MASTERS ".
03 FDATA PIC ZZZZZ9.
03 FILLER PIC X(8) VALUE SPACES.
03 XNOUT PIC X(12) VALUE " NEW MASTER ".
03 FOUT PIC ZZZZZ9.

**01 HDR.**
03 CCl PIC 9 VALUE 1.
03 FILLER PIC X.
03 LDATE PIC X(6) VALUE "DATE: ".
03 RMTH PIC X(4).
03 FILLER PIC X.
03 ROY PIC Z9.
03 B8 PIC XXXX VALUE ", 19".
03 RYR PIC XX.
03 FILLER PIC X(21).
03 TITLEA PIC X(60).
03 FILLER PIC X(21).
03 PGE PIC X(4) VALUE "PAGE".
03 PNUM PIC ZZZ9.
03 FILLER PIC X(2).

01 HDR2.
03 CC PIC 9 VALUE 0.
03 USGS PIC X(7) VALUE " USGS: ".
03 GSNAME PIC X(20).
03 FILLER PIC XXX.
03 RASINL PIC X7 VALUE " SOURCES ".
03 FILLER PIC X.
03 SRCEID PIC X(50).
03 SRCNAM PIC X(30).
03 CNLZZZ PIC X(6).
03 FILLER PIC X(8).

01 DATAA.
03 ID-CN1 PIC IS XXX.
03 ID-NUM PIC IS 999.
03 TITLEX.
  05 SRCAGC.
   07 GNAME PIC X(20).
   07 CNAME PIC X(30).
  05 FILLER PIC X(10).
  03 FILLER PIC X(14).

01 KEYID1.
03 KEYLAT.
  05 LATDEG PIC 99 VALUE 99.
  05 LATMIN PIC 99 VALUE 99.
  05 LATSEC PIC 99 VALUE 99.

03 KEYLNG.
  05 LNGDEG PIC 99 VALUE 999.
  05 LNGMIN PIC 99 VALUE 99.
  05 LNGSEC PIC 99 VALUE 99.

01 KEYIDX.
03 KEYLAT.
  05 LATDEG PIC 99 VALUE 99.
  05 LATMIN PIC 99 VALUE 99.
  05 LATSEC PIC 99 VALUE 99.

03 KEYLNG.
  05 LNGDEG PIC 99 VALUE 999.
  05 LNGMIN PIC 99 VALUE 99.
  05 LNGSEC PIC 99 VALUE 99.

B-96
PROCEDURE DIVISION.

NOTE-A.
NOTE THE PROGRAM STARTS WITH THIS INITIALIZATION
ROUTINE THAT OPENS THE INPUT AND LISTING FILES
AND THEN THE DATE STORED IN THE SYSTEM IS READ
AND USED TO SET UP THE DATE IN THE FIRST
HEADING LINE. NO RETURN FROM THE MAINLINE
ROUTINE IS MADE TO THIS PROCEDURE.

BEGIN.
OPEN INPUT CARD-FILE.
OPEN OUTPUT LIST-FILE.
ACCEPT WKDATE FROM DATE.
ACCEPT JULD FROM DAY.
SET NMM TO MM.
MOVE MONTHY (NMM) TO RMTH OF HDR1.
MOVE DY TO ROY OF HDR1.
MOVE YY TO RYR OF HDR1.

NOTE-B.
NOTE THIS ROUTINE READS THE CONTROL CARDS AND MOVES
THE GENERAL DESCRIPTION OF THE REPORT TO THE
HEADING AREA. THEN A BRANCH IS TAKEN TO THE
APPROPRIATE STARTING ROUTINE DEPENDING ON THE
NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD.
THERE IS NO RETURN TO THIS ROUTINE.

READC.
READ CARD-FILE AT END GO TO STOP1.
IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
ADD 1 TO INCRDS.
MOVE CNL-CARD TO PCNL.
MOVE CNLXXX TO CNLZZZ.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 010 MOVE TITLEX OF DATAA
TO TITLEA OF HDR1 ELSE GO TO STOP4.
ADD 1 TO INCRDS.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 011 MOVE GNAME TO GSNAME
MOVE CNAME TO SRCNAM ELSE GO TO STOP4.
ADD 1 TO INCRDS.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 012 MOVE SRCAGC TO SRCEID,
MOVE GO-CNL TO CNLZZZ ELSE GO TO STOP4.
ADD 1 TO INCRDS.
IF GO-CNL = 001, GO TO STARTA.
IF GO-CNL = 002, GO TO STARTB.
IF GO-CNL = 003
GO TO STARTC
ELSE
GO TO STOP2.

' B-97
NOTE-C.
NOTE NO RETURNS ARE MADE TO ANY OF THE Routines BEFORE
THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING
Routines.

STARTA.
OPEN INPUT CHNGEL
FILEA.
OPEN OUTPUT FILEB
FILEC.
READ FILEA AT END DISPLAY
"****700D1 NO MASTER FILE INPUT DATA", STOP RUN.
MOVE NULL TO NFILEA.
MOVE 0 TO NDETLN, NDATIN, NFILEA.
MOVE KEYIDM OF OLOMAS TO KEYIDX.

STARTA1.
READ CHNGEL AT END DISPLAY "****701D1 END OF THE EDIT RUN"
   MOVE ALA TO HSW
   ADD NULL TO NDATIN.
   IF KEYID OF CNGREC = KEYIDL
       GO TO STARTA1.
   IF KEYID OF CNGREC = KEYIDL
       DISPLAY "****702D1 CHANGE FILE OUT OF SEQUENCE"
       STOP RUN
   ELSE
       MOVE KEYID OF CNGREC TO KEYIDL.

CMP.
   IF NDETLN > 5 GO TO STOP3.
   IF (HSW = ALA OR KEYIDM OF OLOMAS > KEYIDL)
       WRITE NEWMAS FROM OLOMAS
       ADD NULL TO NFILEA
       GO TO RDMS.
   IF KEYIDM OF OLOMAS = KEYIDL GO TO TEST-RNUM.
   IF KEYIDM OF OLOMAS = KEYIDL
       ADD NULL TO NDETLN
       WRITE CNGMAS FROM OLOMAS
       GO TO RDMS.
   IF (KEYIDM OF OLOMAS < KEYIDL AND HSW NOT = ALA)
       GO TO STARTA1.

TEST-RNUM.
   IF RNUM OF OLOMAS = RNUM OF CNGREC
       ADD NULL TO NDETLN
       WRITE CNGMAS FROM OLOMAS
   ELSE
       WRITE NEWMAS FROM OLOMAS
       GO TO RDMS.

RDMS.
READ FILEA AT END CLOSE CHNGEL FILEA FILEB FILEC
DISPLAY "****703D1 NORMAL END OF RUN"
GO TO STOP3.
IF KEYIDM OF OLOMAS > KEYIDX
DISPLAY "****704D1 MASTER FILE OUT OF SEQUENCE"
STOP RUN

B-98
ELSE
    MOVE KEYIDM OF OLDMAS TO KEYIDX.
    ADD NUI TO NFLEIA.
    GO TO CMP.

STARTB.
    STOP RUN.

STARTC.
    STOP RUN.

STOP1.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "CONTROL INPUT FILE EMPTY".
    STOP RUN.

STOP2.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
    STOP RUN.

STOP3.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY TFILEB.
    MOVE INCRDS TO CSTAT.
    MOVE NDATIN TO DATAIN.
    MOVE NFLEIA TO FADATA.
    MOVE NDETLN TO DRLNS.
    MOVE NFLEOA TO FOUT.
    WRITE LIST-LINE FROM JBCNLX.
    DISPLAY "NORMAL JOB TERMINATION".
    DISPLAY "ERRORS = ", ACOUNT.
    CLOSE CARD-FILE, LIST-FILE.
    STOP RUN.

STOP4.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP4".
    STOP RUN.
IDENTIFICATION DIVISION.
PROGRAM-ID. UPDATE-2.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. 6600.
OBJECT-COMPUTER. 6600.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-FILE ASSIGN TO "INPUT".
    SELECT LIST-FILE ASSIGN TO "OUTPUT".
    SELECT FILEA ASSIGN TO MASIN.
    SELECT FILEB ASSIGN TO MASOUT.
    SELECT FILEC ASSIGN TO CMAST.
    SELECT CHNGEL ASSIGN TO DCHNG.
DATA DIVISION.
FILE SECTION.
FD CARD-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS CNL-CARD.
 01 CNL-CARD.
    02 CNLXXX.
      03 ID-CNLI PIC IS XXX.
      03 ID-NUM PIC 999.
    02 DELNUM REDEFINES CNLXXX PIC 9(6).
    02 VSNA PIC X.
    02 ID-DES PIC X(73).
FD LIST-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS LIST-LINE.
 01 LIST-LINE.
    03 CC PIC X.
    03 LLPRT PIC X(132).
FD FILEA
    DATA RECORD IS OLDMAS RECORD CONTAINS 23 TO 327 CHARACTERS
    RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
    CHARACTERS LABEL RECORDS ARE OMITTED.
 01 OLDMAS.
    03 KEYIDM.
      05 KYLATM.
        07 KLD PIC 99.
        07 KLM PIC 99.
        07 KLS PIC 99.
      05 KYLNGM.
        07 KLGD PIC 999.
        07 KLGM PIC 99.
        07 KLGS PIC 99.
      05 KYDUPM PIC 99.
    03 RNUM PIC 999.
    03 LNUM PIC 999.
    03 SESAFE PIC X.
    03 RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
    OF OLDMAS.

B-100
FD FILEB
DATA RECORD IS NEWMAS RECORD CONTAINS 23 TO 327 CHARACTERS
RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
CHARACTERS LABEL RECORDS ARE OMITTED.
01 NEWMAS.
  03 KEYIDM.
    05 KYLATM.
      07 KLD PIC 99.
      07 KLM PIC 99.
      07 KLS PIC 99.
    05 KYLNGM.
      07 KLGD PIC 999.
      07 KLGM PIC 99.
      07 KLGS PIC 99.
    05 KYDUPM PIC 99.
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
    OF NEWMAS.

FD FILEC
DATA RECORD IS CNGMAS RECORD CONTAINS 23 TO 327 CHARACTERS
RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
CHARACTERS LABEL RECORDS ARE OMITTED.
01 CNGMAS.
  03 KEYIDM.
    05 KYLATM.
      07 KLD PIC 99.
      07 KLM PIC 99.
      07 KLS PIC 99.
    05 KYLNGM.
      07 KLGD PIC 999.
      07 KLGM PIC 99.
      07 KLGS PIC 99.
    05 KYDUPM PIC 99.
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
    OF OLDMAS.

FD CHNGEL
DATA RECORD IS CNGREC
LABEL RECORDS ARE OMITTED.
01 CNGREC.
  03 VSN PIC 99999.
  03 VSNA PIC X.
  03 PRD2 PIC 9(5).
  03 KSEQ PIC 9(4).
  03 KEYID.
    05 KEYLAT.
      07 LATDEG PIC 99.
      07 LATMIN PIC 99.
      07 LATSEC PIC 99.
05 KEYLNG.
  07 LNGDEG PIC 999.
  07 LNGMIN PIC 99.
  07 LNGSEC PIC 99.
05 KEYDUP PIC 99.
  03 CSCDE PIC X.
  03 RNUM PIC 999.
  03 CNUM PIC 999.
  03 CCR PIC X.
  03 CCSR PIC X.
  03 TYPDAT PIC 99.
  03 DLEN PIC 99.
  03 NEWDAT PIC X(45).
  03 NSEQ PIC 9(8).
WORKING-STORAGE SECTION.
  77 ACOUNT PIC 9(6) VALUE 0.
  77 ALA PIC X VALUE "A".
  77 ALD PIC X VALUE "D".
  77 ALFLAT PIC X(6).
  77 ALFLNG PIC X(7).
  77 ALM PIC X VALUE "M".
  77 ALPHA6 PIC X(6).
  77 ALPHA7 PIC X(7).
  77 ALR PIC X VALUE "R".
  77 ALT PIC X VALUE "T".
  77 ALX PIC X VALUE "X".
  77 ALY PIC X VALUE "Y".
  77 ALZ PIC X VALUE "Z".
  77 BT PIC 999 VALUE 0.
  77 BTA PIC 9(4).
  77 BLNK1 PIC X VALUE SPACE.
  77 CMPLAT PIC 9(6).
  77 CMPLNG PIC 9(7).
  77 CNRECN PIC 9(6) VALUE ZEROES.
  77 Csw PIC X VALUE "X".
  77 DELSW PIC X VALUE "Z".
  77 DEM1 PIC X(15) VALUE "DATA ENTRY".
  77 DEM2 PIC X(10) VALUE "*UNCHECKED".
  77 DSW PIC 9 VALUE 0.
  77 ERMSG1 PIC X(27) VALUE "AZ FORMAT SORTED CHANGES".
  77 ERMSG2 PIC X(27) VALUE "GW & QW LOCAL IDS UNEQUAL".
  77 ERRCNT PIC 9(9) VALUE 0.
  77 ERRSW PIC X VALUE "Y".
  77 ESW PIC 99 VALUE 0.
  77 FLECNT PIC 9(6).
  77 FRSw PIC X VALUE "F".
  77 HSW PIC X VALUE "X".
  77 INCRDS PIC 9(6) VALUE 0.
  77 JULO PIC 9(5).
  77 LNCNT PIC 99 VALUE 70.
  77 MSG1 PIC X(25) VALUE "CHECKS OK DROP NOT NEEDED".
  77 MSG2 PIC X(25) VALUE "KEYID NOT = LAT-LNG Flds".
  77 MSW PIC X VALUE "X".
  77 NDATIN PIC 9(6) VALUE 0.
01 WDATE.
   03 YY PIC 99.
   03 MM PIC 99.
   03 DD PIC 99.

01 NAME-MONTH.
   03 JAN PIC XXXX VALUE " JAN".
   03 FEB PIC XXXX VALUE " FEB".
   03 MAR PIC XXXX VALUE " MAR".
   03 APR PIC XXXX VALUE " APR".
   03 MAY PIC XXXX VALUE " MAY".
   03 JUNE PIC XXXX VALUE "JUNE".
   03 JULY PIC XXXX VALUE " JULY"
   03 AUG PIC XXXX VALUE " AUG".
   03 SEPT PIC XXXX VALUE " SEPT".
   03 OCT PIC XXXX VALUE " OCT".
   03 NOV PIC XXXX VALUE " NOV".
   03 DEC PIC XXXX VALUE " DEC".

01 MONTH-TABLE REDEFINES NAME-MONTH.
   03 MUNTHY PIC XXXX OCCURS 12 TIMES INDEXED BY NMM.

01 JOBCLNL.
   03 CC PIC X VALUE "1".
   03 PCNL.
      05 CNTYPE PIC XXX.
      05 GUCNL PIC 999.
      05 PCNLX PIC X(74).
   03 FILLER PIC X(52) VALUE SPACES.

01 JBCNLX.
   03 CC PIC X VALUE "0".
   03 PSTATS PIC X(14) VALUE "RECORD COUNTS".
   03 FILLER PIC X(6) VALUE SPACES.
   03 XCARD PIC X(8) VALUE "CARDS IN".
   03 CSTAT PIC ZZZZZ9.
   03 XDRLNS PIC X(20) VALUE " SITES TO BE CHANGED".
   03 DRLNS PIC ZZZZZ9.
   03 XDATIZN PIC X(15) VALUE " CHANGE RECORDS".
   03 DATAIN PIC ZZZZZ9.
   03 XFAIN PIC X(19) VALUE " UNCHANGED MASTERS ".
   03 FADATA PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 XNOUT PIC X(12) VALUE " NEW MASTER ".
   03 FOUT PIC ZZZZZ9.
01 HDR1.
  03 CCI PIC 9 VALUE 1.
  03 FILLER PIC X.
  03 LOADF PIC X(6) VALUE "DATE: ".
  03 RMTH PIC X(4).
  03 FILLER PIC X.
  03 RDY PIC Z9.
  03 X0 PIC XXXX VALUE ", 19".
  03 RYR PIC XX.
  03 FILLER PIC X(21).
  03 TITLEA PIC X(60).
  03 FILLER PIC X(21).
  03 PGE PIC X(4) VALUE "PAGE".
  03 PNUM PIC ZZZ9.
  03 FILLER PIC X(2).
01 HDR2.
  03 CC PIC 9 VALUE 0.
  03 USGS PIC X(7) VALUE " USGS: ".
  03 GSNAME PIC X(20).
  03 FILLER PIC XXX.
  03 BASINL PIC X(7) VALUE "SOURCES".
  03 FILLER PIC X.
  03 SRCID PIC X(50).
  03 SRCCNAM PIC X(30).
  03 CNLZZZ PIC X(6).
  03 FILLER PIC X(8).
01 DATAA.
  03 IO-CNL PIC IS XXX.
  03 ID-NUM PIC IS 999.
  03 TITLEX.
    05 SRCAGC.
      07 GNAME PIC X(20).
      07 CNAME PIC X(30).
      05 FILLER PIC X(10).
    03 FILLER PIC X(14).
01 KEYIDL.
  03 KEYLAT.
    05 LATDEG PIC 99 VALUE 99.
    05 LATMIN PIC 99 VALUE 99.
    05 LATSEC PIC 99 VALUE 99.
  03 KEYLNG.
    05 LNGDEG PIC 999 VALUE 999.
    05 LNGMIN PIC 99 VALUE 99.
    05 LNGSEC PIC 99 VALUE 99.
  03 KEYDUP PIC 99 VALUE 99.
01 KEYIDX.
  03 KEYLAT.
    05 LATDEG PIC 99 VALUE 99.
    05 LATMIN PIC 99 VALUE 99.
    05 LATSEC PIC 99 VALUE 99.
  03 KEYLNG.
    05 LNGDEG PIC 999 VALUE 999.
    05 LNGMIN PIC 99 VALUE 99.
    05 LNGSEC PIC 99 VALUE 99.
  03 KEYDUP PIC 99 VALUE 99.

B-104
01 RTW000.
05 C1-KEYID.
   07 KEY-LATITUDE PIC 9(6).
   07 KEY-LONGITUDE PIC 9(7).
   07 KEY-DUP-NUM PIC 99.
05 REC-NUM PIC 999.
05 REC-LENGTH PIC 999.
05 ACCESS-CODE PIC X.
05 C2-SITE-TYPE PIC X.
05 C3-DATA-RELIABILITY PIC X.
05 C4-SOURCE-AGENCY PIC X(5).
05 C5-PROJECT-NUM.
   07 BASIN-1 PIC X(7).
   07 BASIN-2 PIC X(6).
05 C6-DISTRICT PIC 999.
05 C7-STATE PIC 99.
05 C8-COUNTY PIC 999.
05 C9-LATITUDE PIC 9(6).
05 C10-LONGITUDE PIC 9(7).
05 C11-LAT-LONG-ACURACY PIC X.
05 C12-LOCAL-NUMBER.
   07 LOCAL-SITE PIC X(14).
   07 LOCAL-REST PIC X(6).
05 C13-LAND-NET-LOC PIC X(30).
05 C14-LOCATION-MAP-ID PIC X(30).
05 C15-MAP-SCALE PIC X(6).
05 C16-ALTITUDE PIC 9(5)V99.
05 C17-ALTITUDE-METHOD PIC X.
05 C18-ALTITUDE-ACCURACY PIC XXX.
05 C19-TOPO-SETTING PIC X.
05 C20-DWDC-HYDRO-UNIT PIC X(8).
05 C21-DATE-CONSTRUCTED.
   07 F-C-MONTH PIC XX.
   07 F-C-DAY PIC XX.
   07 F-C-YEAR PIC XXXX.
05 C22-SITE-USE PIC X.
05 C23-WATER-USE PIC X.
05 C24-SECOND-WATER-USE PIC X.
05 C25-THIRD-WATER-USE PIC X.
05 C26-DEPTH PIC 9(5)V99.
05 C27-WELL-DEPTH PIC 9(5)V99.
05 C28-WELL-DEPTH-SOURCE PIC X.
05 C29-HEIGHT PIC 9(5)V99.
05 C30-WATER-LEVEL PIC 9(8).
05 C31-WATER-LEVEL-DATE PIC 9(8).
05 C32-DATE-ACCRCY-WL PIC X.
05 C33-WATER-LEVEL-SOURCE PIC X.
05 C34-MEAS-METHOD-WL PIC X.
05 C35-PUMP-USED PIC X.
05 C36-GEHYDRO-DATA-SOURCE PIC X.
05 C37-SITE-STATUS PIC X.
05 C38-LAST-UPDATE PIC X(8).
05 C41-VERIFIED PIC X.
05 MEAS-PNT-HEIGHT PIC 999V99.
05 MEAS-PNT-DATE PIC 9(8).
05 CURECD PIC X.
PROCEDURE DIVISION.

NOTE-A.
NOTE THE PROGRAM STARTS WITH THIS INITIALIZATION ROUTINE THAT OPENS THE INPUT AND LISTING FILES AND THEN THE DATE STORED IN THE SYSTEM IS READ AND USED TO SET UP THE DATE IN THE FIRST HEADING LINE. NO RETURN FROM THE MAINLINE ROUTINE IS MADE TO THIS PROCEDURE.

BEGIN.
OPEN INPUT CARD-FILE.
OPEN OUTPUT LIST-FILE.
ACCEPT WKDATE FROM DATE.
ACCEPT JULD FROM DAY.
SET NMM TO MM.
MOVE MONTHY (NMM) TO RMTH OF HDR1.
MOVE DY TO RDIY OF HDR1.
MOVE YY TO RYR OF HDR1.

NOTE-B.
NOTE THIS ROUTINE READS THE CONTROL CARDS AND MOVES THE GENERAL DESCRIPTION OF THE REPORT TO THE HEADING AREA. THEN A BRANCH IS TAKEN TO THE APPROPRIATE STARTING ROUTINE DEPENDING ON THE NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD. THERE IS NO RETURN TO THIS ROUTINE.

NOTE-C.
NOTE NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING ROUTINES.

START A.
OPEN INPUT CHNGEL, FILEA.
OPEN OUTPUT FILEB, FILEC.
READ FILEA AT END DISPLAY
"***700D2 NO MASTER FILE INPUT DATA", MOVE ALA TO MSW,
MOVE ZEROS TO KEYIDM OF OLDMAS.
MOVE NU1 TO NFLEIA. MOVE 0 TO NDATLN, NDATIN, NFLEOA.
MOVE KEYIDM OF OLDMAS TO KEYIDX.

RDCHNG.
READ CHNGEL AT END DISPLAY
"***70102 END OF CHANGE INPUT", MOVE ALA TO CSW, GO TO CMC.
IF KEYID OF CNGREC > KEYIDL, ADD NU1 TO TFILEC, DISPLAY
"***70202 CHANGE FILE OUT OF SEQUENCE", TFILEC,
STOP RUN ELSE ADD NU1 TO NDATIN.
IF KEYID OF CNGREC = KEYIDL AND RNUM OF CNGMAS = RNUM OF CNGREC, GO TO PCHG.
IF KEYID OF CNGREC = KEYIDL AND RNUM OF CNGMAS NOT = RNUM OF CNGREC, MOVE ZEROS TO KEYIDL.
IF KEYID OF CNGREC < KEYIDL, MOVE KEYID OF CNGREC TO KEYIDL.
IF DSW = 1, PERFORM EDTMAC.

CMPMC.
IF KEYIDX = KEYIDL AND RNUM OF OLDMAS = RNUM OF CNGREC,
MOVE OLDMAS TO CNGMAS, GO TO PCHG.
IF KEYIDX = KEYIDL AND RNUM OF OLDMAS < RNUM OF CNGREC,
GO TO ROMAS.
IF KEYIDX > KEYIDL, GO TO RDCHNG.
IF KEYIDX < KEYIDL, GO TO RDCHNG.
RDMS.

IF MSW = ALA, GO TO CMC.
READ FILEA AT END DISPLAY
"****70301 END OF MASTER FILE", MOVE ALA TO MSW,
   GO TO CMC.
IF KEYIDM OF OLDMAS > KEYIDX, DISPLAY
"****704D1 MASTER FILE OUT OF SEQUENCE", STOP RUN
   ELSE MOVE KEYIDM OF OLDMAS TO KEYIDX.
   GO TO CMPMC.

CMC.

IF DSW = 1 AND CSW = ALA, PERFORM EDTMAC.
DISPLAY "OLDMAS ", OLDMAS.
DISPLAY "NEWMAS ", NEWMAS.
DISPLAY "CNGMAS ", CNGMAS.
DISPLAY "CNGREC ", CNGREC.
IF ( CSW = ALA AND MSW = ALA ) GO TO STOP3.
IF CSW = ALA, MOVE ZEROS TO KEYIOIL, GO TO RDMAS.
IF MSW = ALA, MOVE ZEROS TO KEYIDX, GO TO RDCHNG.
DISPLAY "****728D2 INCORRECT END OF FILES". STOP RUN.

P000.

MOVE CNGMAS TO RTW000.
IF CNUM OF CNGREC = 012 AND CCSR OF CNGREC = ALM, MOVE NU1
   TO DSW, MOVE NEWDAT OF CNGREC TO C12-LOCAL-NUMBER.
   IF CNUM OF CNGREC = 012 AND CCSR OF CNGREC = ALD, MOVE NU1
   TO DSW, MOVE SPACES TO C12-LOCAL-NUMBER.
IF CNUM OF CNGREC = 014 AND CCSR OF CNGREC = ALM, MOVE NU1
   TO DSW, MOVE NEWDAT OF CNGREC TO C14-LOCATION-MAP-ID.
   IF CNUM OF CNGREC = 014 AND CCSR OF CNGREC = ALD, MOVE NU1
   TO DSW, MOVE SPACES TO C14-LOCATION-MAP-ID.
IF CNUM OF CNGREC = 015 AND CCSR OF CNGREC = ALM, MOVE NU1
   TO DSW, MOVE NFWDAT OF CNGREC TO C15-MAP-SCALE.
   IF CNUM OF CNGREC = 015 AND CCSR OF CNGREC = ALD, MOVE NU1
   TO DSW, MOVE SPACES TO C15-MAP-SCALE.
MOVE RTW000 TO CNGMAS. GO TO RDCHNG.

P001.

GO TO RDCHNG.

P002.

GO TO RDCHNG.

P003.

GO TO RDCHNG.

P042.

GO TO RDCHNG.

P047.

GO TO RDCHNG.

P055.

GO TO RDCHNG.

P058.

GO TO RDCHNG.

P072.

GO TO RDCHNG.

P076.

GO TO RDCHNG.

P082.

GO TO RDCHNG.

B-107
B-108
DISPLAY "PCHG" NDATIN.
IF RNUM OF CNGREC = 000, GO TO P000.
IF RNUM OF CNGREC = 001, GO TO P001.
IF RNUM OF CNGREC = 002, GO TO P002.
IF RNUM OF CNGREC = 003, GO TO P003.
IF RNUM OF CNGREC = 042, GO TO P042.
IF RNUM OF CNGREC = 047, GO TO P047.
IF RNUM OF CNGREC = 055, GO TO P058.
IF RNUM OF CNGREC = 058, GO TO P058.
IF RNUM OF CNGREC = 072, GO TO P072.
IF RNUM OF CNGREC = 076, GO TO P076.
IF RNUM OF CNGREC = 082, GO TO P082.
IF RNUM OF CNGREC = 090, GO TO P090.
IF RNUM OF CNGREC = 094, GO TO P094.
IF RNUM OF CNGREC = 098, GO TO P098.
IF RNUM OF CNGREC = 105, GO TO P105.
IF RNUM OF CNGREC = 114, GO TO P114.
IF RNUM OF CNGREC = 121, GO TO P121.
IF RNUM OF CNGREC = 127, GO TO P127.
IF RNUM OF CNGREC = 134, GO TO P134.
IF RNUM OF CNGREC = 146, GO TO P146.
IF RNUM OF CNGREC = 158, GO TO P158.
IF RNUM OF CNGREC = 164, GO TO P164.
IF RNUM OF CNGREC = 171, GO TO P171.
IF RNUM OF CNGREC = 180, GO TO P180.
IF RNUM OF CNGREC = 183, GO TO P183.
IF RNUM OF CNGREC = 186, GO TO P186.
IF RNUM OF CNGREC = 189, GO TO P189.
IF RNUM OF CNGREC = 192, GO TO P192.
IF RNUM OF CNGREC = 198, GO TO P198.
IF RNUM OF CNGREC = 203, GO TO P203.
IF RNUM OF CNGREC = 208, GO TO P208.
IF RNUM OF CNGREC = 212, GO TO P212.
IF RNUM OF CNGREC = 219, GO TO P219.
IF RNUM OF CNGREC = 234, GO TO P234.
IF RNUM OF CNGREC = 250, GO TO P250.
IF RNUM OF CNGREC = 277, GO TO P277.
IF RNUM OF CNGREC = 320, GO TO P320.
DISPLAY "****729D2 INVALID RECORD TYPE", RNUM OF CNGREC.
DISPLAY CNGREC. GO TO ROCHNG.

EDTMAC.

DISPLAY "EDTMAC" NDATIN.
IF RNUM OF CNGMAS = 000, PERFORM E000.
IF RNUM OF CNGMAS = 001, PERFORM E001.
IF RNUM OF CNGMAS = 002, PERFORM E002.
IF RNUM OF CNGMAS = 003, PERFORM E003.
IF RNUM OF CNGMAS = 042, PERFORM E042.
IF RNUM OF CNGMAS = 047, PERFORM E047.
IF RNUM OF CNGMAS = 055, PERFORM E058.
IF RNUM OF CNGMAS = 058, PERFORM E058.
IF RNUM OF CNGMAS = 072, PERFORM E072.
IF RNUM OF CNGMAS = 076, PERFORM E076.
IF RNUM OF CNGMAS = 082, PERFORM E082.
IF RNUM OF CNGMAS = 090, PERFORM E090.
IF RNUM OF CNGMAS = 094, PERFORM E094.
IF RNUM OF CNGMAS = 098, PERFORM E098.
IF RNUM OF CNGMAS = 105, PERFORM E105.
IF RNUM OF CNGMAS = 114, PERFORM E114.
IF RNUM OF CNGMAS = 121, PERFORM E121.
IF RNUM OF CNGMAS = 127, PERFORM E127.
IF RNUM OF CNGMAS = 134, PERFORM E134.
IF RNUM OF CNGMAS = 146, PERFORM E146.
IF RNUM OF CNGMAS = 158, PERFORM E158.
IF RNUM OF CNGMAS = 164, PERFORM E164.
IF RNUM OF CNGMAS = 171, PERFORM E171.
IF RNUM OF CNGMAS = 180, PERFORM E180.
IF RNUM OF CNGMAS = 183, PERFORM E183.
IF RNUM OF CNGMAS = 186, PERFORM E186.
IF RNUM OF CNGMAS = 189, PERFORM E189.
IF RNUM OF CNGMAS = 192, PERFORM E192.
IF RNUM OF CNGMAS = 198, PERFORM E198.
IF RNUM OF CNGMAS = 203, PERFORM E203.
IF RNUM OF CNGMAS = 208, PERFORM E208.
IF RNUM OF CNGMAS = 212, PERFORM E212.
IF RNUM OF CNGMAS = 219, PERFORM E219.
IF RNUM OF CNGMAS = 234, PERFORM E234.
IF RNUM OF CNGMAS = 250, PERFORM E250.
IF RNUM OF CNGMAS = 277, PERFORM E277.
IF RNUM OF CNGMAS = 320, PERFORM E320.
WRITE NEWMAS FROM CNGMAS. MOVE ZERO TO DSW.
E000.
  DISPLAY CNGMAS.
E001.
  DISPLAY CNGMAS.
E002.
  DISPLAY CNGMAS.
E003.
  DISPLAY CNGMAS.
E042.
  DISPLAY CNGMAS.
E047.
  DISPLAY CNGMAS.
E055.
  DISPLAY CNGMAS.
E058.
  DISPLAY CNGMAS.
E072.
  DISPLAY CNGMAS.
E076.
  DISPLAY CNGMAS.
E082.
  DISPLAY CNGMAS.
E090.
  DISPLAY CNGMAS.
E094.
  DISPLAY CNGMAS.
E098.
  DISPLAY CNGMAS.
E105.
   DISPLAY CNGMAS.
E114.
   DISPLAY CNGMAS.
E121.
   DISPLAY CNGMAS.
E127.
   DISPLAY CNGMAS.
E134.
   DISPLAY CNGMAS.
E141.
   DISPLAY CNGMAS.
E158.
   DISPLAY CNGMAS.
E164.
   DISPLAY CNGMAS.
E171.
   DISPLAY CNGMAS.
E180.
   DISPLAY CNGMAS.
E183.
   DISPLAY CNGMAS.
E186.
   DISPLAY CNGMAS.
E189.
   DISPLAY CNGMAS.
E192.
   DISPLAY CNGMAS.
E198.
   DISPLAY CNGMAS.
E203.
   DISPLAY CNGMAS.
E208.
   DISPLAY CNGMAS.
E212.
   DISPLAY CNGMAS.
E219.
   DISPLAY CNGMAS.
E234.
   DISPLAY CNGMAS.
E250.
   DISPLAY CNGMAS.
E277.
   DISPLAY CNGMAS.
E320.
   DISPLAY CNGMAS.
STARTb.
   STOP RUN.
STARTC.
   STOP RUN.
STOP1.
   WRITE LIST-LINE FROM JOBCNL.
   DISPLAY "CONTROL INPUT FILE EMPTY".
   STOP RUN.

B-111
STOP2.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
    STOP RUN.

STOP3.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY TFILEB.
    MOVE INCROS TO CSTAT.
    MOVE NDATIN TO DATAIN.
    MOVE NFLEIA TO FAADATA.
    MOVE NDETLN TO DRLNS.
    MOVE NFLEOA TO FOUT.
    WRITE LIST-LINE FROM JBNCNLX.
    DISPLAY "NORMAL JOB TERMINATION".
    DISPLAY "ERRORS = ", ACOUNT.
    CLOSE CARD-FILE, LIST-FILE.
    STOP RUN.

STOP4.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP4".
    STOP RUN.
IDENTIFICATION DIVISION.
PROGRAM-ID. BASIN3.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. CDC CYBER-175.
OBJECT-COMPUTER. CDC CYBER-175.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT CARD-FILE ASSIGN TO "INPUT".
SELECT LIST-FILE ASSIGN TO "OUTPUT".
SELECT FILE3 ASSIGN TO MGQWIN
RESERVE 4 ALTERNATE AREAS
ORGANIZATION IS SEQUENTIAL.
SELECT PLTFLE ASSIGN TO PLTDSK
ORGANIZATION IS SEQUENTIAL.
SELECT ERRFLE ASSIGN TO ERDdisk.
DATA DIVISION.
FILE SECTION.
FD CARD-FILE
LABEL RECORD OMITTED
DATA RECORD IS CNL-CARD.
01 CNL-CARD.
  03 ID-CNl PIC IS XXX.
  03 ID-Num PIC 999.
  03 ID-DES PIC X(74).
FD LIST-FILE
LABEL RECORD OMITTED
DATA RECORD IS LIST-LINE.
01 LIST-LINC.
  03 CC PIC X.
  03 LLPRl.
    05 FILLER PIC X.
    05 P0 PIC X(10).
    05 FILLER PIC XXX.
    05 P1 PIC X(10).
    05 FILLER PIC XXX.
    05 P2 PIC X(10).
    05 FILLER PIC XXX.
    05 P3 PIC X(10).
    05 FILLER PIC XXX.
    05 P4 PIC X(10).
    05 FILLER PIC XXX.
    05 P5 PIC X(10).
    05 FILLER PIC XXX.
    05 P6 PIC X(10).
    05 FILLER PIC XXX.
    05 P7 PIC X(10).
    05 FILLER PIC XXX.
    05 P8 PIC X(10).
    05 FILLER PIC XXX.
    05 P9 PIC X(10).
  03 PIND PIC X(4).
FD FILE8
DATA RECORD IS OLDMAS
RECORD CONTAINS 23 TO 327 CHARACTERS
RECORDING MODE IS BINARY
BLOCK CONTAINS 4793 TO 5120 CHARACTERS
LABEL RECORDS ARE STANDARD.

01 OLDMAS.
  03 KEYID.
     05 KEYLAT.
        07 KLD PIC 99.
        07 KLM PIC 99.
        07 KLS PIC 99.
     05 KEYLNG.
        07 KLGD PIC 999.
        07 KLGM PIC 99.
        07 KLGS PIC 999.
     05 KEYDUP PIC 99.
  03 RNUM PIC IS 999.
  03 LNUM PIC IS 999.
  03 SESAFE PIC X.
  03 RESTRX PIC X OCCURS 1 TO 305 TIMES,
       DEPENDING ON LNUM OF OLDMAS.

FD PLTFLE
DATA RECORD IS RECOUT
LABEL RECORDS ARE OMITTED.

01 RECOUT .
  03 CNRECR PIC 9(4).
  03 KEYID.
     05 KEYLAT PIC 9(6).
     05 KEYLNG PIC 9(7).
     05 KEYDUP PIC 99.
  03 LCWLID.
     05 LCSITE PIC X(14).
     05 LCREST PIC X(6).
  03 DDEPTH PIC 9(5).
  03 IDEPTH PIC X.
  03 ALTUDE PIC 9(5).
  03 PPDISH PIC 9(5).
  03 PPYEAR PIC X(4).
  03 FCYEAR PIC X(4).
  03 WTRUSE PIC X.
  03 INLOGS PIC X(8).
  03 OPNTOP PIC 9(5)V99.
  03 OPNTYP PIC X.
  03 CDIAM PIC 999V99.
  03 FINISH PIC X.
  03 WTRLVL2 PIC -9(4)V99.
  03 WLYEAR PIC X(4).
  03 WLMETH2 PIC X.
  03 WLCNT2 PIC 99.
  03 CNTQ1O PIC 99.
  03 CNTQ20 PIC 99.
  03 CNTQ30 PIC 99.

FD ERRFLE
LABEL RECORDS ARE OMITTED
DATA RECORD IS ERREC.

01 ERREC.
  03 ERPRT PIC X(132).

B-114
WORKING-STORAGE SECTION.
  77 AC PIC 9(4).
  77 ACOuNT PIC 9(6) VALUE 0.
  77 BC PIC 9(4).
  77 BLKCNT PIC 9 VALUE 0.
  77 BT PIC 999.
  77 BTA PIC 999.
  77 CNBLNK PIC 9(6) VALUE 0.
  77 CNRECN PIC 9(6) VALUE ZEROES.
  77 CNTQW1 PIC 99 VALUE ZEROES.
  77 CNTQW2 PIC 99 VALUE ZEROES.
  77 CNTQW3 PIC 99 VALUE ZEROES.
  77 CK PIC 9(4).
  77 COUNTA PICTURE 9(3) VALUE 0.
  77 COUNTN PIC 9(6) VALUE 0.
  77 COUNT2 PIC 9(6) VALUE 0.
  77 COUNT2G PIC 9(6) VALUE 0.
  77 CTRTOT PIC 9(8) VALUE 0.
  77 DA1 PIC 9(4).
  77 DA2 PIC 9(4).
  77 DA3 PIC 9(4).
  77 DAN PIC 9(4).
  77 DCOuNT PIC 9(6) VALUE 0.
  77 DN1 PIC 9(4).
  77 DN2 PIC 9(4).
  77 EKEYID PIC 9(15) VALUE ZEROES.
  77 ELCNCT PIC 999 VALUE 000.
  77 ENTEMP PIC 999 VALUE ZEROES.
  77 EPGCNT PIC 999 VALUE 001.
  77 ERRCNT PIC 9(6) VALUE 0.
  77 ERRSW PIC X VALUE "Y".
  77 ESEQ PIC 9(5) VALUE 0.
  77 ESW PIC X VALUE "X".
  77 EXTSW PIC X VALUE "Y".
  77 ESPU PIC ZZZZZ.
  77 FRSW PIC X VALUE "F".
  77 FRTSW PIC X VALUE "F".
  77 FOCLAT USAGE IS COMP-2.
  77 FLATD USAGE IS COMP-2.
  77 FLATM USAGE IS COMP-2.
  77 FLATS USAGE IS COMP-2.
  77 FOCLNG USAGE IS COMP-2.
  77 FLNGD USAGE IS COMP-2.
  77 FLNGM USAGE IS COMP-2.
  77 FLNGS USAGE IS COMP-2.
  77 GWCOUNT PIC 9(6) VALUE 0.
  77 HCOuNT PIC 9(6) VALUE 0.
  77 HSW PIC X VALUE "X".
  77 INCRDS PIC 9(6) VALUE 0.
77 KEYIDL PIC 9(15) VALUE 0.
77 KTEST USAGE IS COMP-2.
77 LATLNA PIC 9(13).
77 LATLNB PIC 9(13).
77 LA1 PIC 9(4).
77 LA2 PIC 9(4).
77 LNCNT PIC 99 VALUE 70.
77 LN1 PIC 9(4).
77 LN2 PIC 9(4).
77 LOGSW PIC X VALUE "N".
77 LSW PIC X VALUE "X".
77 LTEST USAGE IS COMP-2.
77 LX PIC 9(4).
77 NDATIN PIC 9(6) VALUE 0.
77 NDATLN PIC 9(6) VALUE 0.
77 NFLEIA PIC 9(6) VALUE 0.
77 NFLEOA PIC 9(6) VALUE 0.
77 NHCNT PIC 9(6).
77 NLATMS PIC 9(6).  
77 NLATDC PIC 999999.
77 NLNGMS PIC 9(7).
77 NLNGDC PIC 999999.
77 NMM PIC 999.
77 NSW PIC X VALUE "X".
77 OCOUNT PIC 9(6) VALUE 0.
77 PCOUNT PIC 9(6) VALUE 0.
77 PGCNT PIC 999 VALUE 0.
77 PRTSW PIC X VALUE "F".
77 SCOUNT PIC 9(6) VALUE 0.
77 SLSW PIC X VALUE "Y".
77 TFILEB PIC 9(6) VALUE 0.
77 WLDCNT PIC 99 VALUE ZEROS.
77 WLDTMP PIC X(4) VALUE SPACES.
77 WLTEST PIC X(7).
01 LABELF.
  03 IDENTX PIC XXXX.
  03 FILLER PIC X(17).
  03 REKCNT PIC 9(6).
  03 FILLER PIC X(53).
01 WKDATE.
  03 YY PIC 99.
  03 MM PIC 99.
  03 DY PIC 99.
01 NAME-MONTH.
  03 JAN PIC XXXX VALUE " JAN".
  03 FEB PIC XXXX VALUE " FEB".
  03 MAR PIC XXXX VALUE " MAR".
  03 APR PIC XXXX VALUE " APR".
  03 MAY PIC XXXX VALUE " MAY".
  03 JUNE PIC XXXX VALUE " JUNE".
  03 JULY PIC XXXX VALUE " JULY".
  03 AUG PIC XXXX VALUE " AUG".
  03 SEPT PIC XXXX VALUE " SEPT".
  03 OCT PIC XXXX VALUE " OCT".
  03 NOV PIC XXXX VALUE " NOV".
  03 DEC PIC XXXX VALUE " DEC".

B-116
01 MONTH-TABLE REDEFINES NAME-MONTH.
   03 MONTHY PIC xxxx OCCURS 12 TIMES.
01 JOBCNL.
   03 CC PIC X VALUE "1".
   03 PCNL.
      05 CNTYPE PIC XXX.
      05 GO-CNl PIC 999.
      05 PCNLX PIC X(74).
   03 FILLER PIC X(52) VALUE SPACES.
01 JBCNLX.
   03 CC PIC X VALUE "0".
   03 PSTATS PIC X(14) VALUE "RECORD COUNTS".
   03 FILLER PIC X(6) VALUE SPACES.
   03 XCARD PIC X(8) VALUE "CARDS IN".
   03 CSTAT PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 XDLRNS PIC X(12) VALUE "DETAIL LINES".
   03 DRNLS PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 XDATIZN PIC X(7) VALUE "DATA IN".
   03 DATAIN PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 XFAIN PIC X(11) VALUE "ULD-DATA I".
   03 FADATA PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 XNOUT PIC X(12) VALUE "FMT DATA OUT".
   03 FOUT PIC ZZZZZ9.
01 HDR1.
   03 CC1 PIC 9 VALUE 1.
   03 FILLER PIC X.
   03 LDATE PIC X(6) VALUE "DATE: ".
   03 RMTH PIC X(4).
   03 FILLER PIC X.
   03 RDY PIC Z9.
   03 XR PIC XXXX VALUE ", 19".
   03 RYR PIC XX.
   03 FILLER PIC X(21).
   03 TITLEA PIC X(60).
   03 FILLER PIC X(21).
   03 PGE PIC X(4) VALUE "PAGE".
   03 PNUM PIC ZZZ9.
   03 FILLER PIC X(2).
01 HDR2.
   03 CC PIC 9 VALUE 0.
   03 USGS PIC X(7) VALUE " USGS: ".
   03 GSNAME PIC X(20).
   03 FILLER PIC XXX.
   03 BASINL PIC X(7) VALUE "SOURCES".
   03 FILLER PIC X.
   03 SRCEID PIC X(50).
   03 SRCNAM PIC X(30).
   03 FILLER PIC X(14).

B-117
HDR3.
03 CC PIC 9 VALUE 0.
03 FILLER PIC X VALUE SPACES.
03 FILLER PIC X(3) VALUE "REC".
03 FILLER PIC X(5).
03 FILLER PIC X(19) VALUE "SITE IDENTIFICATION".
03 FILLER PIC X(19).
03 FILLER PIC X(5) VALUE "DEPTH".
03 FILLER PIC X(7).
03 FILLER PIC X(5) VALUE "CONST".
03 FILLER PIC XXX.
03 FILLER PIC X(3) VALUE "LOG".
03 FILLER PIC X(5).
03 FILLER PIC X(12) VALUE "WATER LEVELS".
03 FILLER PIC XX.
03 FILLER PIC X(7) VALUE "DISCHRG".
03 FILLER PIC X(5).
03 FILLER PIC X(8) VALUE "OPENINGS".
03 FILLER PIC XX.
03 FILLER PIC X(7) VALUE "CASINGS".
03 FILLER PIC X.
03 FILLER PIC X(10) VALUE "RECORD CNT".
03 FILLER PIC X.

HDR4.
03 CC PIC X VALUE SPACES.
03 FILLER PIC X VALUE SPACES.
03 FILLER PIC X(3) VALUE "NUM".
03 FILLER PIC X(3).
03 FILLER PIC X(3) VALUE "LAT".
03 FILLER PIC X(4).
03 FILLER PIC X(4) VALUE "LONG".
03 FILLER PIC X(7).
03 FILLER PIC X(8) VALUE "LOCAL ID".
03 FILLER PIC X(12).
03 FILLER PIC X(4) VALUE "MEAS".
03 FILLER PIC X.
03 FILLER PIC XXX VALUE "SRC".
03 FILLER PIC X.
03 FILLER PIC X(3) VALUE "ALT".
03 FILLER PIC XX.
03 FILLER PIC X(4) VALUE "YEAR".
03 FILLER PIC XXXX.
03 FILLER PIC X(5) VALUE "TYPES".
03 FILLER PIC XXX.
03 FILLER PIC X(4) VALUE "MEAS".
03 FILLER PIC X.
03 FILLER PIC XX VALUE "YR".
03 FILLER PIC X.
03 FILLER PIC X(3) VALUE "MTH".
03 FILLER PIC XX.
03 FILLER PIC X(3) VALUE "GPM".
03 FILLER PIC XX.
03 FILLER PIC XX VALUE "YR".
03 FILLER PIC X.

B-118
03 FILLER PIC XXX VALUE "USE".
03 FILLER PIC X(3).
03 FILLER PIC X(3) VALUE "TOP".
03 FILLER PIC X.
03 FILLER PIC XXX VALUE "TYP".
03 FILLER PIC XX.
03 FILLER PIC X(4) VALUE "DIAM".
03 FILLER PIC X.
03 FILLER PIC X VALUE "F".
03 FILLER PIC X.
03 FILLER PIC XX VALUE "WL".
03 FILLER PIC X.
03 FILLER PIC XX VALUE "Q1".
03 FILLER PIC X.
03 FILLER PIC XX VALUE "Q2".
03 FILLER PIC X.
03 FILLER PIC XX VALUE "Q3".

01 HDR5.
03 CC PIC 9 VALUE 0.
03 FILLER PIC X(40).
03 FILLER PIC X(43) VALUE "SEE LAST PAGE FOR EXPLANATION OF ".
 "GWSI CODES".

01 ERH01.
03 CC PIC 9 VALUE 1.
03 FILLER PIC X(25) VALUE SPACES.
03 ETITLE PIC X(80).
03 FILLER PIC X(14) VALUE SPACES.
03 FILLER PIC X(6) VALUE "PAGE : ".
03 EPNUM PIC ZZ9.

01 ERH02.
03 FILLER PIC X(10) VALUE SPACES.
03 FILLER PIC X(12) VALUE "SWAB/RASA : ".
03 FILLER PIC X(14) VALUE "STAFF USE ONLY".
03 FILLER PIC X(14) VALUE SPACES.
03 FILLER PIC X(25) VALUE "LISTING OF REJECTED SITES".

01 ERH03.
03 FILLER PIC X(5) VALUE SPACES.
03 FILLER PIC X(6) VALUE "REJECT".
03 FILLER PIC XX VALUE SPACES.
03 FILLER PIC X(5) VALUE "INPUT".
03 FILLER PIC X(9) VALUE SPACES.
03 FILLER PIC X(4) VALUE "SITE".
03 FILLER PIC X(14) VALUE SPACES.
03 FILLER PIC X(5) VALUE "LOCAL".
03 FILLER PIC X(20) VALUE SPACES.
03 FILLER PIC X(10) VALUE "REASON FOR".
03 FILLER PIC X(25) VALUE SPACES.
03 FILLER PIC X(10) VALUE "CORRECTION".

01 ERH04.
03 FILLER PIC X(5) VALUE SPACES.
03 FILLER PIC X(6) VALUE "NUMBER".
03 FILLER PIC XX VALUE SPACES.
03 FILLER PIC X(6) VALUE "SEQNCE".
03 FILLER PIC X(5) VALUE SPACES.
03 FILLER PIC X(10) VALUE "IDENTIFIER".

B-119
03 FILLER PIC X(10) VALUE SPACES.
03 FILLER PIC X(7) VALUE "WELL ID".
03 FILLER PIC X(19) VALUE SPACES.
03 FILLER PIC X(9) VALUE "REJECTION".
03 FILLER PIC X(28) VALUE SPACES.
03 FILLER PIC X(5) VALUE "NOTES".
01 ERMSGS.
03 EMSG1 PIC X(25) VALUE "LATITUDE OUTSIDE POLYGON".
03 EMSG2 PIC X(25) VALUE "LONGITUDE EAST OF POLYGON".
03 EMSG3 PIC X(25) VALUE "LONGITUDE WEST OF POLYGON".
01 ERECLN.
03 FILLER PIC X(5) VALUE SPACES.
03 ERSEQ PIC ZZZZ9.
03 FILLER PIC XX VALUE SPACES.
03 INSEQ PIC ZZZZ9.
03 FILLER PIC XXX VALUE SPACES.
03 ERINT PIC X(15).
03 FILLER PIC XXX VALUE SPACES.
03 ELCID PIC X(20).
03 FILLER PIC XXX VALUE SPACES.
03 ENOTE PIC X(80).
01 DICT-HDR1.
03 FILLER PIC 9 VALUE 0.
03 FILLER PIC X(50).
03 FILLER PIC X(26) VALUE "EXPLANATION OF GWSI CODES".
03 FILLER PIC X(50).
01 DICT-HDR2.
03 FILLER PIC 9 VALUE 0.
03 FILLER PIC X(9).
03 FILLER PIC X(5) VALUE "DEPTH".
03 FILLER PIC X(14).
03 FILLER PIC X(12) VALUE "WATER LEVELS".
03 FILLER PIC X(11).
03 FILLER PIC X(9) VALUE "AVAILABLE".
03 FILLER PIC X(14).
03 FILLER PIC X(3) VALUE "USE".
03 FILLER PIC X(21).
03 FILLER PIC X(4) VALUE "TYPE".
03 FILLER PIC X(14).
03 FILLER PIC X(6) VALUE "FINISH".
03 FILLER PIC X(9).
01 DICT-HDR3.
03 FILLER PIC 9 VALUE 0.
03 FILLER PIC X(9).
03 FILLER PIC X(6) VALUE "SOURCE".
03 FILLER PIC X(12).
03 FILLER PIC X(17) VALUE "MEAS METHOD[MTH]".
03 FILLER PIC X(7).
03 FILLER PIC X(9) VALUE "LOG TYPES".
03 FILLER PIC X(12).
03 FILLER PIC X(8) VALUE "OF WATER".
03 FILLER PIC X(17).
03 FILLER PIC X(8) VALUE "OPENINGS".

B-120
03 FILLER PIC X(11).
03 FILLER PIC X(5) VALUE "[ F ]".
03 FILLER PIC X(10).
01 DICT-LN1.
03 FILLER PIC 9 VALUE 0.
03 FILLER PIC X(4).
03 FILLER PIC X(17) VALUE "S RPTING AGENCY".
03 FILLER PIC X(4).
03 FILLER PIC X(11) VALUE "A AIRLINE".
03 FILLER PIC X(13).
03 FILLER PIC X(8) VALUE "A TIME".
03 FILLER PIC X(12).
03 FILLER PIC X(19) VALUE "A AIR CONDITIONING".
03 FILLER PIC X(6).
03 FILLER PIC X(12) VALUE "F FRACTURE".
03 FILLER PIC X(6).
03 FILLER PIC X(19) VALUE "C POROUS, CONCRETE".
01 DICT-LN2.
03 FILLER PIC 9 VALUE 0.
03 FILLER PIC X(4).
03 FILLER PIC X(11) VALUE "D DRILLER".
03 FILLER PIC X(10).
03 FILLER PIC X(15) VALUE "C CAL AIRLINE".
03 FILLER PIC X(9).
03 FILLER PIC X(10) VALUE "B COLLAR".
03 FILLER PIC X(10).
03 FILLER PIC X(11) VALUE "B BOTTLING".
03 FILLER PIC X(14).
03 FILLER PIC X(12) VALUE "L LOUVERED".
03 FILLER PIC X(6).
03 FILLER PIC X(19) VALUE "F GRAVEL W/PERF".
01 DICT-LN3.
03 FILLER PIC 9 VALUE 0.
03 FILLER PIC X(4).
03 FILLER PIC X(9) VALUE "O OWNER".
03 FILLER PIC X(12).
03 FILLER PIC X(13) VALUE "E ESTIMATED".
03 FILLER PIC X(11).
03 FILLER PIC X(11) VALUE "C CALIPER".
03 FILLER PIC X(9).
03 FILLER PIC X(13) VALUE "C COMMERCIAL".
03 FILLER PIC X(12).
03 FILLER PIC X(8) VALUE "M MESH".
03 FILLER PIC X(10).
03 FILLER PIC X(17) VALUE "G GRAVEL, SCREEN".
03 FILLER PIC XX.
01 DICT-LN4.
03 FILLER PIC 9 VALUE 0.
03 FILLER PIC X(4).
03 FILLER PIC X(15) VALUE "A OTHER GOV'T".
03 FILLER PIC X(6).
03 FILLER PIC X(17) VALUE "G PRESSURE GAGE".
03 FILLER PIC X(7).
03 FILLER PIC X(13) VALUE "D DRILLER'S".

B-121
03 FILLER PIC X(7).
03 FILLER PIC X(10) VALUE "D DEWATER".
03 FILLER PIC X(15).
03 FILLER PIC X(14) VALUE "P PERFORATED".
03 FILLER PIC X(4).
03 FILLER PIC X(17) VALUE "H HORIZ GALLERY".
03 FILLER PIC XX.

01 DICT-LN5.
03 FILLER PIC 9 VALUE 0.
03 FILLER PIC X(4).
03 FILLER PIC X(18) VALUE "R OTHER REPORTED".
03 FILLER PIC X(3).
03 FILLER PIC X(21) VALUE "H CAL PRESSURE GAGE".
03 FILLER PIC X(3).
03 FILLER PIC X(12) VALUE "E ELECTRIC".
03 FILLER PIC X(8).
03 FILLER PIC X(8) VALUE "E POWER".
03 FILLER PIC X(17).
03 FILLER PIC X(14) VALUE "R WIRE WOUND".
03 FILLER PIC X(4).
03 FILLER PIC X(12) VALUE "O OPEN END".
03 FILLER PIC X(5).

01 DICT-LN6.
03 FILLER PIC 9 VALUE 0.
03 FILLER PIC X(4).
03 FILLER PIC X(8) VALUE "L LOGS".
03 FILLER PIC X(13).
03 FILLER PIC X(20) VALUE "L GEOPHYSICAL LOGS".
03 FILLER PIC X(4).
03 FILLER PIC X(17) VALUE "F FLUID CONDUCT".
03 FILLER PIC X(3).
03 FILLER PIC X(7) VALUE "F FIRE".
03 FILLER PIC X(18).
03 FILLER PIC X(15) VALUE "S SCREEN(UNK)".
03 FILLER PIC X(3).
03 FILLER PIC X(14) VALUE "P PERFORATED".
03 FILLER PIC X(5).

01 DICT-LN7.
03 FILLER PIC 9 VALUE 0.
03 FILLER PIC X(4).
03 FILLER PIC X(13) VALUE "G GEOLOGIST".
03 FILLER PIC X(8).
03 FILLER PIC X(13) VALUE "M MANOMETER".
03 FILLER PIC X(11).
03 FILLER PIC X(13) VALUE "G GEOLOGIST".
03 FILLER PIC X(7).
03 FILLER PIC X(11) VALUE "H DOMESTIC".
03 FILLER PIC X(14).
03 FILLER PIC X(14) VALUE "T SAND POINT".
03 FILLER PIC X(4).
03 FILLER PIC X(10) VALUE "S SCREEN".
03 FILLER PIC X(9).

B-122
01 DICT-LN8.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(4).
 03 FILLER PIC X(9) VALUE "Z OTHER".
 03 FILLER PIC X(12).
 03 FILLER PIC X(12) VALUE "R REPORTED".
 03 FILLER PIC X(12).
 03 FILLER PIC X(12) VALUE "H MAGNETIC".
 03 FILLER PIC X(8).
 03 FILLER PIC X(13) VALUE "I IRRIGATION".
 03 FILLER PIC X(12).
 03 FILLER PIC X(10) VALUE "W WALLED".
 03 FILLER PIC X(8).
 03 FILLER PIC X(14) VALUE "T SAND POINT".
 03 FILLER PIC X(5).

01 DICT-LN9.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(25).
 03 FILLER PIC X(14) VALUE "S STEEL TAPE".
 03 FILLER PIC X(10).
 03 FILLER PIC X(13) VALUE "I INDUCTION".
 03 FILLER PIC X(7).
 03 FILLER PIC X(22) VALUE "J INDUSTRIAL(COOLING)".
 03 FILLER PIC X(3).
 03 FILLER PIC X(13) VALUE "X OPEN HOLE".
 03 FILLER PIC X(5).
 03 FILLER PIC X(10) VALUE "W WALLED".

01 DICT-LN10.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(25).
 03 FILLER PIC X(17) VALUE "T ELECTRIC TAPE".
 03 FILLER PIC X(7).
 03 FILLER PIC X(13) VALUE "J GAMMA RAY".
 03 FILLER PIC X(7).
 03 FILLER PIC X(9) VALUE "K MINING".
 06 FILLER PIC X(16).
 03 FILLER PIC X(9) VALUE "Z OTHER".
 03 FILLER PIC X(9).
 03 FILLER PIC X(13) VALUE "X OPEN HOLE".
 03 FILLER PIC X(5).

01 DICT-LN11.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(25).
 03 FILLER PIC X(21) VALUE "Y CAL ELECTRIC TAPE".
 03 FILLER PIC X(3).
 03 FILLER PIC X(12) VALUE "K DIPMETER".
 03 FILLER PIC X(8).
 03 FILLER PIC X(12) VALUE "M MEDICINAL".
 03 FILLER PIC X(31).
 03 FILLER PIC X(9) VALUE "Z OTHER".
 03 FILLER PIC X(9).

01 DICT-LN12.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(25).
 03 FILLER PIC X(9) VALUE "Z OTHER".
 03 FILLER PIC X(15).
 03 FILLER PIC X(12) VALUE "L LATERLOG".
 03 FILLER PIC X(8).
 03 FILLER PIC X(13) VALUE "N INDUSTRIAL".

B-123
01 DICT-LN13.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(49).
 03 FILLER PIC X(12) VALUE "M MICROLOG".
 03 FILLER PIC X(8).
 03 FILLER PIC X(16) VALUE "P PUBLIC SUPPLY".
01 DICT-LN14.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(49).
 03 FILLER PIC X(11) VALUE "N NEUTRON".
 03 FILLER PIC X(9).
 03 FILLER PIC X(14) VALUE "Q AQUACULTURE".
01 DICT-LN15.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(49).
 03 FILLER PIC X(9) VALUE "O LATER".
 03 FILLER PIC X(11).
 03 FILLER PIC X(13) VALUE "R RECREATION".
01 DICT-LN16.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(49).
 03 FILLER PIC X(9) VALUE "P PHOTO".
 03 FILLER PIC X(11).
 03 FILLER PIC X(8) VALUE "S STOCK".
01 DICT-LN17.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(49).
 04 FILLER PIC X(15) VALUE "Q RADIOACTIVE".
 03 FILLER PIC X(5).
 03 FILLER PIC X(14) VALUE "T INSTITUTION".
01 DICT-LN18.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(49).
 03 FILLER PIC X(8) VALUE "S SONIC".
 03 FILLER PIC X(12).
 03 FILLER PIC X(9) VALUE "U UNUSED".
01 DICT-LN19.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(49).
 03 FILLER PIC X(7) VALUE "T TEMP".
 03 FILLER PIC X(13).
 03 FILLER PIC X(15) VALUE "Y DESALINATION".
01 DICT-LN20.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(49).
 03 FILLER PIC X(14) VALUE "U GAMMA-GAMMA".
 03 FILLER PIC X(6).
 03 FILLER PIC X(8) VALUE "Z OTHER".
01 DICT-LN21.
 03 FILLER PIC 9 VALUE 0.
 03 FILLER PIC X(49).
 03 FILLER PIC X(17) VALUE "V FLUID VELOCITY".

B-124
01 DICT=LN22.
  03 FILLER PIC 9 VALUE 0.
  03 FILLER PIC X(49).
  03 FILLER PIC X(8) VALUE "Z OTHER".
01 DBGUULNS.
  03 STRLN.
  05 FILLER PIC 9 VALUE 1.
  05 STRS PIC X(130) VALUE ALL "*".
03 DBGUG1.
  05 OB11 PIC 9 VALUE 0.
  05 OB12 PIC X(11) VALUE " SITE ID : ".
  05 LTD PIC XX.
  05 OB12 PIC X VALUE SPACES.
  05 LTM PIC XX.
  05 OB13 PIC X VALUE SPACES.
  05 LTS PIC XX.
  05 OB14 PIC XXX VALUE SPACES.
  05 LGD PIC XXX.
  05 OB15 PIC X VALUE SPACES.
  05 LGM PIC XX.
  05 OB16 PIC X VALUE SPACES.
  05 LGS PIC XX.
  05 FILLER PIC X(10) VALUE SPACES.
  05 FILLER PIC X(20) VALUE "DEBUG CALLED FROM : ".
  05 DBSRC PIC X(20).
03 DBGUG2.
  05 OB21 PIC 9 VALUE 0.
  05 OB22 PIC X(8) VALUE "TEST2 : ".
03 DBGUG3.
  05 OB31 PIC 9 VALUE 0.
  05 OB32 PIC X(10) VALUE " NLATMS : ".
  05 ELATMS PIC 9(6).
  05 OB33 PIC X(17) VALUE " LATITUDE-1(1) : ".
  05 ELATITUDE-1 PIC 9(6).
  05 OB34 PIC X(17) VALUE " LATITUDE-2(1) : ".
  05 ELATITUDE-2 PIC 9(6).
  05 OB35 PIC X(12) VALUE " NLNGMS : ".
  05 ELNGMS PIC 9(7).
  05 OB36 PIC X(11) VALUE " LONGL : ".
  05 ELOLNG PIC 9(7).
  05 OB37 PIC X(11) VALUE " HILNG : ".
  05 EHILNG PIC 9(7).
03 DBGUG4.
  05 OB41 PIC 9 VALUE 0.
  05 OB41 PIC X(8) VALUE "VCARO : ".
  05 EVSEQ PIC 9999.
03 DBGUG5.
  05 OB51 PIC 9 VALUE 0.
  05 OB52 PIC X(12) VALUE " LATITUDE-L : ".
  05 ELAT-L PIC 9(6).
  05 OB53 PIC X(14) VALUE " LNGTH-L : ".
  05 ELNG-L PIC 9(7).
  05 OB54 PIC X(14) VALUE " LATITUDE-H : ".
  05 ELAT-H PIC 9(6).
05 D855 PIC X(14) VALUE " LNTUD-H : ".
05 DLNG-H PIC 9(7).
05 D856 PIC X(13) VALUE " DECLAT-L : ".
05 EOL-L PIC 99.9999.
05 D857 PIC X(9) VALUE " LS1 : ".
05 ELS1 PIC X.
05 D858 PIC X(12) VALUE " DECLNG-L : ".
05 EDL-L PIC 999.9999.
05 DBUG6.
05 D861 PIC 9 VALUE 0.
05 D862 PIC X(9) VALUE " ASIGN: ".
05 EASIN PIC X.
05 D863 PIC X(13) VALUE " ACOEFF : ".
05 EACOF PIC 99999999.
05 D864 PIC X(11) VALUE " BSIGN : ".
05 EBSIN PIC X.
05 D865 PIC X(13) VALUE " BCUEFF : ".
05 EBCOF PIC 99999999.
05 D866 PIC X(19) VALUE "DECLAT-H: ".
05 EOL-H PIC 99999999.
05 D867 PIC X(9) VALUE " LS2 : ".
05 ELS2 PIC X.
05 D868 PIC X(12) VALUE " DECLNG-H: ".
05 EDL-L PIC 999.9999.
05 DBUG7.
05 D871 PIC 9 VALUE 0.
05 D872 PIC X(9) VALUE " CSIGN : ".
05 ECSIN PIC X.
05 D873 PIC X(12) VALUE " KNSTNT : ".
05 EKNSTNT PIC 99999999.
05 DBUG8.
05 D881 PIC 9 VALUE 0.
05 D882 PIC X(8) VALUE "LATDMS : ".
05 ELTOMS PIC 9(6).
05 D883 PIC X(13) VALUE " FDCLAT: ".
05 EFDCOLT PIC 9(6).
05 D884 PIC X(26) VALUE " LNTDMS: ".
05 ELGOMS PIC 9(7).
05 D885 PIC X(8) VALUE "LNGDMS: ".
05 EFGOMS PIC 9(7).
05 D886 PIC X(14) VALUE " FDCLNG : ".
05 EFDCLG PIC 9(7).
05 DBUG9.
05 D891 PIC 9 VALUE 0.
05 D892 PIC X(10) VALUE " SPACES.
05 D893 PIC X(10) VALUE " LATD : ".
05 ELATD PIC 99.
05 D894 PIC X(10) VALUE " LATM : ".
05 ELATM PIC 99.
05 D895 PIC X(10) VALUE " LATS : ".
05 ELATS PIC 99.
05 D896 PIC X(26) VALUE " SPACES.
05 D897 PIC X(10) VALUE " LNSD : ".
05 ELNGD PIC 999.
05 D898 PIC X(10) VALUE " LNGM : ".
05 ELNGM PIC 99.
05 D899 PIC X(10) VALUE " LNGS : ".
05 ELNGS PIC 99.

03 DEBUG10.
05 D8101 PIC 9 VALUE 0.
05 D8102 PIC X(9) VALUE SPACES.
05 D8103 PIC X(11) VALUE " FLATD : ".
05 EFLATD PIC 99.
05 D8104 PIC X(11) VALUE " FLATM : ".
05 EFLATM PIC 99.
05 D8105 PIC X(11) VALUE " FLATS : ".
05 EFLATS PIC 99.
05 D8106 PIC X(26) VALUE SPACES.
05 D8107 PIC X(11) VALUE " FLNGD : ".
05 EFLNGD PIC 999.
05 D8108 PIC X(11) VALUE " FLNGM : ".
05 EFLNGM PIC 99.
05 D8109 PIC X(11) VALUE " FLNGS : ".
05 EFLNGS PIC 99.

03 DEBUG11.
05 D8111 PIC 9 VALUE 0.
05 D8112 PIC X(8) VALUE "TEST21: ".

03 DEBUG12.
05 D8121 PIC 9 VALUE 0.
05 D8122 PIC X(8) VALUE " DA1 : ".
05 EDA1 PIC 99.
05 D8123 PIC X(17) VALUE " ACOEFS(DA1) : ".
05 ECOF-DA1 PIC .9999999999.
05 D8124 PIC X(19) VALUE " ACUEFS(DA1+1) : ".
05 ECOF-DA2 PIC .9999999999.
05 D8125 PIC X(17) VALUE " BCOEFS(DA1) : ".
05 EBCOF-DA1 PIC .9999999999.
05 D8126 PIC X(19) VALUE " BCUEFS(DA1+1) : ".
05 EBCOF-DA2 PIC .9999999999.

03 DEBUG13.
05 D8131 PIC 9 VALUE 0.
05 D8132 PIC X(10) VALUE " KTEST : ".
05 EKTEST PIC 999999.999999.
05 D8133 PIC X(12) VALUE " LTEST : ".
05 ELTEST PIC 999999.999999.
05 D8134 PIC X(17) VALUE " KNSTNS(DA1) : ".
05 EKNSTNS PIC 9999.9999.
05 D8135 PIC X(19) VALUE " KNSTNS(DA1+1) : ".
05 EKNSTNS-2 PIC 9999.9999.
05 D8136 PIC X(7) VALUE " LN1: ".
05 ELN1 PIC 9999.
05 D8137 PIC X(7) VALUE " LN2: ".
05 ELN2 PIC 9999.

01 SITELN.
03 CC PIC 9 VALUE SPACE.
03 CNRECP PIC ZZZZ9.
03 FILLER PIC X.
03 KEYID.
  05 KEYLAT PIC 9(6).
  05 FILLER PIC X.
  05 KEYLNG PIC 9(7).
  05 FILLER PIC X.
  05 KEYDUP PIC 99.
  05 FILLER PIC X.
03 LCWLID.
  05 LCSITE PIC X(14).
  05 LCREST PIC X(6).
  05 FILLER PIC X.
  03 DDEPTH PIC ZZZZZ.
  03 FILLER PIC X.
  03 IDEPTH PIC X.
  03 FILLER PIC X.
  03 ALTUDE PIC ZZZZZ.
  03 FILLER PIC X.
  03 FCYEAR PIC X(4).
  03 FILLER PIC X.
  03 INLOGS PIC X(8).
  03 FILLER PIC X.
  03 WTRLVL3 PIC -ZZZZZ.
  03 FILLER PIC X.
  03 WLYEAR PIC X(4).
  03 FILLER PIC X.
  03 WLMETH3 PIC X.
  03 FILLER PIC X.
  03 PPDISH PIC ZZZZ9.
  03 FILLER PIC X.
  03 PPYEAR PIC X(4).
  03 FILLER PIC X.
  03 WTRUE PIC X.
  03 FILLER PIC X.
  03 OPNTOP PIC ZZZZ.99.
  03 FILLER PIC X.
  03 OPNTYP PIC X.
  03 FILLER PIC X.
  03 CDIAM PIC ZZ.99.
  03 FILLER PIC X.
  03 FINISH PIC X.
  03 FILLER PIC X.
  03 WLCNT3 PIC Z9.
  03 FILLER PIC X.
  03 CNTQ1S PIC Z9.
  03 FILLER PIC X.
  03 CNTQ2S PIC Z9.
  03 FILLER PIC X.
  03 CNTQ3S PIC Z9.
01 DATAA.
  03 ID-CNl PIC IS XXX.
  03 ID-NUM PIC IS 999.
  03 TITLEX.

B-128
05 SRCGC.
  07 GNAME PIC X(20).
  07 CNAME PIC X(30).
  05 FILLER PIC X(10).
  03 FILLER PIC X(13).
01 BCARD.
  03 LULAT PIC 9(6).
  03 LOLNG PIC 9(7).
  03 HILAT PIC 9(6).
  03 HILNG PIC 9(7).
  03 BASIN1 PIC X(7).
  03 BASINX REDEFINES BASIN1.
  05 BASIN2 PIC X(6).
  05 FILLER PIC X.
  03 BTYPE PIC X.
  03 BDERSP PIC X(50).
01 VCARD.
  03 LATUDE-L PIC 9(6).
  03 LNGTUD-L PIC 9(7).
  03 LATUDE-H PIC 9(6).
  03 LNGTUD-H PIC 9(7).
  03 DECLAT-L PIC 99V9999.
  03 LS1 PIC X.
  03 DECLNG-L PIC 99V9999.
  03 DECLAT-H PIC 99V9999.
  03 LS2 PIC X.
  03 DECLNG-H PIC 99V9999.
  03 ASIGN PIC X.
  03 ACOEFF PIC 9V9999999.
  03 BSIGN PIC X.
  03 BCOEFF PIC 9V9999999.
  03 CSIGN PIC X.
  03 KNSTNT PIC 99V9999.
03 VSEQ PIC 9999.
01 SLN.
  03 CC PIC X VALUE SPACE.
  03 LATUDE-L PIC 9(6).
  03 FILLER PIC XX.
  03 LNGTUD-L PIC 9(7).
  03 FILLER PIC XX.
  03 LATUDE-H PIC 9(6).
  03 FILLER PIC XX.
  03 LNGTUD-H PIC 9(7).
  03 FILLER PIC XX.
  03 DECLAT-L PIC 99V9999.
  03 FILLER PIC XX.
  03 DECLNG-L PIC 99V9999.
  03 FILLER PIC XX.
  03 DECLAT-H PIC 99V9999.
  03 FILLER PIC XX.
  03 DECLNG-H PIC 99V9999.
  03 FILLER PIC XX.
  03 ACOEFF PIC 9.9999999.
  03 ASIGN PIC X.

B-129
03 FILLER PIC XX.
03 BCOEFF PIC 9.9999999.
03 ASIGN PIC X.
03 FILLER PIC XX.
03 KNSTNT PIC ZZZ.9999.
03 CSIGN PIC X.
03 FILLER PIC XX.
03 VSEQ PIC ZZZZ.

01 LATONE.
 03 LATUDE-1 PIC 9(6) OCCURS 50 TIMES.
01 LNGONE.
 03 LNGTUD-1 PIC 9(7) OCCURS 50 TIMES.
01 LATTWO.
 03 LATUDE-2 PIC 9(6) OCCURS 50 TIMES.
01 LNGTWO.
 03 LNGTUD-2 PIC 9(7) OCCURS 50 TIMES.
01 DATONE.
 03 DECLAT-1 PIC 99V9999 OCCURS 50 TIMES.
01 DNGONE.
 03 DECLNG-1 PIC 999V9999 OCCURS 50 TIMES.
01 DATTWO.
 03 DECLAT-2 PIC 99V9999 OCCURS 50 TIMES.
01 DNGTWO.
 03 DECLNG-2 PIC 999V9999 OCCURS 50 TIMES.
01 ACOEF.
 03 ACOEFS USAGE IS COMP-2 OCCURS 50 TIMES.
01 BCOEF.
 03 BCOEFS USAGE IS COMP-2 OCCURS 50 TIMES.
01 CCOEF.
 03 KNSTNS USAGE IS COMP-2 OCCURS 50 TIMES.
01 LATDMS.
 03 LATD PIC 99.
 03 LATH PIC 99.
 03 LATS PIC 99.
01 LNGDMS.
 03 LNGD PIC 999.
 03 LNGH PIC 99.
 03 LNGS PIC 99.
01 TYPLOG.
 03 LOGTYP PIC X OCCURS 8 TIMES.

01 RTWOOO.
 03 KEYID.
    05 KEYLAT PIC 9(6).
    05 KEYLNG PIC 9(7).
    05 KEYDUP PIC 99.
 03 RNUM PIC 999.
 03 LNUM PIC 999.
 03 SESAFE PIC X.
 03 SITETY PIC X.
 03 URELIx PIC X.
 03 REPAGY PIC X(5).
 03 PROJNU.
    05 BASIN1 PIC X(7).
    05 RASIN2 PIC X(6).

B-130
03 DISTRT PIC XXX.
03 STATEW PIC XX.
03 COUNTY PIC XXX.
03 LLACUR PIC X.
03 LCWLD.
   05 LCSITE PIC X(14).
   05 LCREST PIC X(6).
03 LCNET PIC X(30).
03 LCLMAP PIC X(30).
03 SCALEN PIC X(6).
03 ALTITUDE PIC 9(5)V99.
03 MEASR PIC X.
03 MACCUR PIC XXX.
03 TOPSET PIC X.
03 OWDCUR PIC X(8).
03 FCMNTH PIC XX.
03 FCCDAY PIC XX.
03 FCYEAR PIC XXXX.
03 FILLER PIC X.
03 SITUSE PIC X.
03 WTRUSE PIC X.
03 SITRUSE PIC X.
03 TWTRUS PIC X.
03 HDEPTH PIC 9(5)V99.
03 WDEPTH PIC 9(5)V99.
03 WDSRCE PIC X.
03 WTRLVL PIC 9(5)V99.
03 WDAT.
   05 WLMNTH PIC XX.
   05 WLLDAY PIC XX.
   05 WLYEAR PIC XXXX.
03 WLOADCR PIC X.
03 WLSRCE PIC X.
03 WLMETH PIC X.
03 PUMPUS PIC X.
03 GHSRCE PIC X.
03 XTATUS PIC X.
03 LUPDAT PIC X(8).
03 VERIFY PIC X.
03 MSHGHT PIC 999V99.
03 MSDATE PIC 9(8).
03 CURECD PIC X.
01 RTG001.
  03 KEYID.
   05 KEYLAT PIC 9(6).
   05 KEYLNG PIC 9(7).
   05 KEYDUP PIC 99.
03 RNUM PIC 999.
03 LNUM PIC 999.
03 SESAFE PIC X.
03 LATLNG.
   05 LATITUDE PIC 9(6).
   05 LNTUD PIC 9(7).
   05 DUPERK PIC 99.
03 STATEC PIC XX.
03 DISTRT PIC XX.
03 CONTYC PIC XXX.
03 SITECD PIC XX.
03 HYDROC PIC X(8).
03 TDRAIN PIC X(7).
03 CORAIN PIC X(7).
03 DATUMX PIC X(8).
03 WDEPTH PIC 9(7)V99.
03 EPACDE PIC X.

01 RTQ0002.
03 KEYID.
   05 KEYLAT PIC 9(6).
   05 KEYLNG PIC 9(7).
   05 KEYDUP PIC 99.
03 RNUM PIC 999.
03 LNUM PIC 999.
03 SESAFE PIC X.
03 LCWLD.
   05 LCSITE PIC X(14).
   05 LCREST PIC X(6).
03 FILLER PIC X(28).
03 GEOLUC PIC X(8).
03 AOTYPE PIC X.
03 FILLER PIC X(6).
03 EPACDE PIC X.

01 RTW058.
03 KEYID PIC 9(15).
03 RNUM PIC 999.
03 LNUM PIC 999.
03 SESAFE PIC X.
03 EMTNUM PIC 999.
03 ENTD.
   05 DTDAY PIC XX.
   05 DTMNTH PIC XX.
   05 DTYEAR PIC XXXX.
03 DTACCR PIC X.
03 DRTLNAM PIC X(12).
03 FILLER PIC X(28).
03 DATSRC PIC X.
03 METHOD PIC X.
03 FINISH PIC X.
03 BOTTOM PIC 999.
03 OVMETH PIC X.
03 SPCLTR PIC X.

01 RTW076.
03 KEYID PIC 9(15).
03 RNUM PIC 999.
03 LNUM PIC 999.
03 SESAFE PIC X.
03 CSGTOP PIC 9(5)V99.
03 CSGBOT PIC 9(5)V99.
03 CSGDIAM PIC 999V99.
03 CSGMTL PIC X.
03 CSGTHK PIC 99V999.

B-132
01 RTW082.
  03 KEYID PIC 9(15).
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  04 OPNTOP PIC 9(5)V99.
  03 OPNBMT PIC 9(5)V99.
  03 OPNTYP PIC X.
  03 TYPMTL PIC X.
  03 OPNDIAM PIC 999V99.
  03 OPNWID PIC 99.999.
  03 OPNLNG PIC 999.99.

01 RTW198.
  03 KEYID PIC 9(15).
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 TYPELG PIC X.
  04 BDEPTH PIC 9(5)V99.
  03 EDEPTH PIC 9(5)V99.
  03 SRCLOG PIC X.

01 RTW146.
  03 KEYID PIC 9(15).
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 PPSQNO PIC 999.
  03 PPDAT.
    07 PPMNTH PIC XX.
    07 PPDAY PIC XX.
    07 PPYEAR PIC XXXX.
  03 PPDATC PIC X.
  03 PPDISH PIC 9(5)V99.
  03 PPDISS PIC X.
  03 PPMETH PIC X.
  03 PPPWL PIC 9(5)V99.
  03 PPSWL PIC 9(5)V99.
  03 PPLSRC PIC X.
  03 PPLMTH PIC X.
  03 PPPERD PIC 9(4)V9.
  03 PPSCAP PIC 9(5)V99.
  03 FILLER PIC XXX.

01 RTW234.
  03 KEYID PIC 9(15).
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 WLDAT.
    05 WLMNTH PIC XX.
    05 WLDAY PIC XX.
    05 WLYEAR PIC XXXX.
03 DTACCR PIC X.
03 WTRLVL1 PIC S9(4)V99.
03 WLSTAT PIC X.
03 WLMETH1 PIC X.
03 MEASAC PIC X.

01 RTEST.
  03 KEYID.
    05 KEYLAT PIC 9(6) VALUE 340900.
    05 KEYLNG PIC 9(7) VALUE 1133650.
    05 KEYDUP PIC 99.
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 LATLNG.
    05 LATITUDE PIC 9(6).
    05 LNGTUD PIC 9(7).
    05 DUPERK PIC 99.
  03 STATEC PIC XX.
  03 DISTRT PIC XX.
  03 CONTYC PIC XXX.
  03 SITECO PIC XX.
  03 HYDROC PIC X(8).
  03 TDRAIN PIC X(7).
  03 CDRAIN PIC X(7).
  03 DATUMX PIC X(d).
  03 WDEPTH PIC 9(7)V99.
  03 EPACDE PIC X.

01 LNTEST.
  03 CC PIC X VALUE SPACE.
  03 FILLER PIC X(8) VALUE " KTEST= ".
  03 KTESTE PIC ZZZ.9(8).
  03 FILLER PIC X(8) VALUE " LTEST= ".
  03 K1E PIC X(12) VALUE " KNSTNS-1 = ".
  03 KN1E PIC ZZZ.9(8).
  03 L1E PIC X(12) VALUE " KNSTNS-1 = ".
  03 LNE1 PIC ZZZ.9(8).

PROCEDURE DIVISION.
DECLARATIVES.
SFILEB SECTION.
  USE AFTER STANDARD ENDING FILE LABEL
  PROCEDURE ON FILEB.
LFILEB.
  MOVE FILE-LABEL TO LABELF.
  MOVE TFILEB TO REKCNT.
  MOVE LABELF TO FILE-LABEL.
  DISPLAY FILE-LABEL, TFILEB, REKCNT.
END DECLARATIVES.
*NOTE-A.*

THE PROGRAM STARTS WITH THIS INITIALIZATION
ROUTINE THAT OPENS THE INPUT AND LISTING FILES
AND THEN THE DATE STORED IN THE SYSTEM IS READ
AND USED TO SET UP THE DATE IN THE FIRST
HEADING LINE. NO RETURN FROM THE MAINLINE
ROUTINE IS MADE TO THIS PROCEDURE.

BEGIN.
OPEN INPUT CARD-FILE.
OPEN OUTPUT LIST-FILE.
ACCEPT WKDATE FROM DATE.
MOVE MM TO NMM.
MOVE MONTHLY (NMM) TO RMTH OF HDR1.
MOVE YY TO RYR OF HDR1.
MOVE SPACES TO wLDTMP.
MOVE ZEROS TO ENTEMP.

*NOTE-B.*

THIS ROUTINE READS THE CONTROL CARDS AND MOVES
THE GENERAL DESCRIPTION OF THE REPORT TO THE
HEADING AREA. THEN A BRANCH IS TAKEN TO THE
APPROPRIATE STARTING ROUTINE DEPENDING ON THE
NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD.
THERE IS NO RETURN TO THIS ROUTINE.

READC.
READ CARD-FILE AT END GO TO STOP1.
IF ID-CNGL OF CNL-CARD NOT = "CNL" GO TO STOP2.
IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
ADD 1 TO INCRDS.
MOVE CNL-CARD TO PCNL.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNGL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 010 MOVE TITLEX OF DATAA
TO TITLEA OF HDR1 ELSE GO TO STOP4.
ADD 1 TO INCRDS.
IF GO-CNGL = 001, GO TO STARTA, ELSE GO TO STOP2.

STARTA.
OPEN INPUT FILEB.
OPEN OUTPUT ERRFILE.
OPEN OUTPUT PLTFILE.

READA.
READ CARD-FILE INTO BCARD AT END GO TO STOP5.
ADD 1 TO INCRDS.

*NOTE-C.*

NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE
THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING
ROUTINES. A RETURN TO READA1 IS MADE FROM THE
ROUTINE ENDFILE WHEN A TAPE RECORD IS LESS THAN THE
LOWEST LATITUDE IN THE CURRENT BASIN AND THERE
IS NOT A TERMINATION RECORD Signals BY 000000 IN
THE FIRST SIX POSITIONS OF THE CURRENT RECORD.
READB1.
    MOVE BDESRP OF BCARD TO SRCEID OF HQR2.
    MOVE BASIN1 OF BCARD TO BASINL OF HQR2.

*NOTE-D.*
* THE ROUTINES FROM HERE TO THE ROUTINE *
* FINOPT READ AND STORE THE SLICES FOR THE *
* BASIN WHILE EDITING AND COMPUTING THE *
* COEFFICIENTS FOR THE LINE SEGMENTS. *
* EXIT IS TAKEN WHEN THE FIRST RECORD *
* OF THE NEXT SLICE IS READ AND A BRANCH *
* IS TAKEN TO THE ROUTINE, FINDPT. *
* THE RETURN IS TO READB1 SINCE THE FIRST *
* RECORD OF THE NEXT SLICE HAS NOT BEEN *
* STORED IN THE TABLE. *

READV.
    READ CARD-FILE INTO VCARD AT END GO TO STOPS.
    ADD 1 TO INCARDS.

READV1.
    IF FRSW = "F", PERFORM CLRVSS VARYING LA1 FROM 1 BY
        1 UNTIL LA1 = 51,
        MOVE "Y" TO FRSW, SET LA1 TO 1,
        MOVE LATUDE-L OF VCARD TO NHOLD.
    IF LATUDE-L OF VCARD < NHOLD, GO TO FINOPT.

STRVCD.
    IF LATUDE-L OF VCARD > NHOLD, GO TO STOPS.
    EXAMINE LATUDE-L OF VCARD REPLACING ALL " " BY 0.
    MOVE LATUDE-L OF VCARD TO LATUDE-1 (LA1).
    EXAMINE LNGTUD-L OF VCARD REPLACING ALL " " BY 0.
    MOVE LNGTUD-L OF VCARD TO LNGTUD-1 (LA1).
    EXAMINE LATUDE-H OF VCARD REPLACING ALL " " BY 0.
    MOVE LATUDE-H OF VCARD TO LATUDE-2 (LA1).
    EXAMINE LNGTUD-H OF VCARD REPLACING ALL " " BY 0.
    MOVE LNGTUD-H OF VCARD TO LNGTUD-2 (LA1).
    EXAMINE DECLAT-L OF VCARD REPLACING ALL " " BY 0.
    MOVE DECLAT-L OF VCARD TO DECLAT-1 (LA1).
    EXAMINE DECLNG-L OF VCARD REPLACING ALL " " BY 0.
    MOVE DECLNG-L OF VCARD TO DECLNG-1 (LA1).
    EXAMINE DECLAT-H OF VCARD REPLACING ALL " " BY 0.
    MOVE DECLAT-H OF VCARD TO DECLAT-2 (LA1).
    EXAMINE DECLNG-H OF VCARD REPLACING ALL " " BY 0.
    MOVE DECLNG-H OF VCARD TO DECLNG-2 (LA1).
    EXAMINE ACOEFF OF VCARD REPLACING ALL " " BY 0.
    MOVE ACOEFF OF VCARD TO ACOEFS (LA1).
    IF ASIGN OF VCARD = "-", COMPUTE ACOEFS (LA1) =
        ACOEFS (LA1) - 2 * ACOEFS (LA1),
    ELSE MOVE "+" TO ASIGN OF VCARD.
    EXAMINE BCOEFF OF VCARD REPLACING ALL " " BY 0.
    MOVE BCOEFF OF VCARD TO BCOEFS (LA1).
    IF BSIGN OF VCARD = "-", COMPUTE BCOEFS (LA1) =
        BCOEFS (LA1) - 2 * BCOEFS (LA1),
    ELSE MOVE "+" TO BSIGN OF VCARD.
    EXAMINE KNSTNT OF VCARD REPLACING ALL " " BY 0.
    MOVE KNSTNT OF VCARD TO KNSTNS (LA1).
IF CSIGN OF VCARD = "-", COMPUTE KNSTNS (LA1) =
   KNSTNS (LA1) - 2 * KNSTNS (LA1),
   ELSE MOVE "+" TO CSIGN OF VCARD.
SET LN2 TO LA1.
SET LA1 UP BY 1.
GO TO READV.

*NOTE-E*
* THIS IS THE TAPE READ ROUTINE THAT CONTROLS THE MAIN
* PROGRAM. THE KEY ROUTINES ARE READTAPE, BRNALL,
* CHNG01, CHNG02, TEST2.
* THIS PROCEDURE TESTS THE FIRST RECORD TAPE
* SWITCH FOR THE SITE BEING OUTSIDE THE BASIN
* AND CONTINUES TO READ THE MASTER FILE UNTIL
* A SITE WITHIN THE BASIN IS FOUND.
* RECORDS ARE PROCESSED FOR THAT SITE UNTIL
* A CHANGE OF SITE KEYID IS DETECTED. A
* BRANCH IS MADE TO CHNG01 WHERE THE OUTPUT
* FOR THE PREVIOUS IS MADE IF APPLICABLE.
* THEN IN CHNG02 THE LAST SITE RECORDS ARE
* CLEARED AND THE NEW SITE IS SET UP FOR
* SEARCH AND PROCESSING.
FINDPT.
MOVE "F" TO FRSW.
IF (SLSW = "X") MOVE "Y" TO SLSW, GO TO TEST2.
GO TO READTAPE.
READTAPE.
READ FILEB AT END GO TO CLOSEA.
ADD 1 TO TFILEB, NFLEIA.
IF KEYID OF OLDMAS NOT = KEYIDL, GO TO CHNG01.

*NOTE-WELL*
* THE BRANCH INSTRUCTIONS IN THE PROCEDURE, BRNALL,
* ARE ORDERED BY THEIR RELATIVE FREQUENCY IN THE
* WRDM MASTER FILE. REARRANGEMENT WOULD SLOW
* THE OPERATION.
BRNALL.
IF FRTSW = "X", GO TO READTAPE.
IF RNUM OF OLDMAS = 234, GO TO PTR234.
IF RNUM OF ULMAS = 000, GO TO PTR000.
IF RNUM OF OLDMAS = 001, GO TO PTR001.
IF RNUM OF OLDMAS = 002, GO TO PTR002.
IF RNUM OF OLDMAS = 003, GO TO PTR003.
IF RNUM OF OLDMAS = 058, GO TO PTR058.
IF RNUM OF OLDMAS = 076, GO TO PTR076.
IF RNUM OF OLDMAS = 198, GO TO PTR198.
IF RNUM OF OLDMAS = 042, GO TO PTR042.
IF RNUM OF OLDMAS = 082, GO TO PTR082.
IF RNUM OF OLDMAS = 002, GO TO PTR002.
IF RNUM OF OLDMAS = 001, GO TO PTR001.
IF RNUM OF OLDMAS = 158, GO TO PTR158.
IF RNUM OF OLDMAS = 192, GO TO PTR192.
IF RNUM OF OLDMAS = 146, GO TO PTR146.
IF RNUM OF OLDMAS = 186, GO TO PTR186.
IF RNUM OF OLDMAS = 090, GO TO PTR090.
IF RNUM OF OLDMAS = 114, GO TO PTR114.
IF RNUM OF OLDMAS = 121, GO TO PTR121.
IF RNUM OF OLDMAS = 189, GO TO PTR189.
IF RNUM OF OLDMAS = 094, GO TO PTR094.
IF RNUM OF OLDMAS = 183, GO TO PTR183.
IF RNUM OF OLDMAS = 072, GO TO PTR072.
IF RNUM OF OLDMAS = 047, GO TO PTR047.
IF RNUM OF OLDMAS = 134, GO TO PTR134.
IF RNUM OF OLDMAS = 171, GO TO PTR171.
IF RNUM OF OLDMAS = 180, GO TO PTR180.
IF RNUM OF OLDMAS = 127, GO TO PTR127.
IF RNUM OF OLDMAS = 055, GO TO PTR055.
IF RNUM OF OLDMAS = 098, GO TO PTR098.
IF RNUM OF OLDMAS = 105, GO TO PTR105.
IF RNUM OF OLDMAS = 164, GO TO PTR164.
IF RNUM OF OLDMAS = 203, GO TO PTR203.
IF RNUM OF OLDMAS = 208, GO TO PTR208.
IF RNUM OF OLDMAS = 212, GO TO PTR212.
IF RNUM OF OLDMAS = 219, GO TO PTR219.
IF RNUM OF OLDMAS = 250, GO TO PTR250.
IF RNUM OF OLDMAS = 277, GO TO PTR277.
GO TO STOP9.

CHNG01.
IF FRTSW = "Y", PERFORM PRTSNL.
* THE FOLLOWING CONDITION TO BE USED FOR WITHDRAWING SPECIFIC LOG
  IF FRTSW = "Y" AND LOGSW = "Y", PERFORM PRTSNL.

CHNG02.
MOVE KEYID OF OLDMAS TO KEYIDL.
MOVE SPACES TO SITELN.
MOVE SPACES TO RECOUT.
MOVE "X" TO FRTSW, HSW, NSW, LSW.
MOVE KEYLAT OF OLDMAS TO NLATMS.
MOVE KEYLNG OF OLDMAS TO NLNGMS.
GO TO TEST2.

*NOTE-P.*
* THE FOLLOWING ROUTINES ARE IN PLACE TO *
* ALLOW PROCESSING OF ANY RECORD IN THE *
* (WRDM) WATER RESOURCES MASTER TAPE FILE *
* DATA MAY BE EXTRACTED AND STORED IN THE *
* LAST 41 POSITIONS OF "SITELN" AND "RECOUT" *
* FOR STORING ON YOUR CYBER DISK FILE AND *
* PRINTING ON THE LISTING.
*
* ADDITIONAL OUTPUT MAY BE HANDLED BY *
* SETTING UP ADDITIONAL OUTPUT RECORDS *
* AND INSERTING THE PROPER INSTRUCTIONS *
* TO ADD TO THE OUTPUT *

PTR000.
MOVE "Y" TO HSW.
MOVE "N" TO LOGSW.
MOVE OLDMAS TO RTW000.

PRMOVE.
EXAMINE WDEPHT OF RTW000 REPLACING ALL " " BY 0.
EXAMINE HDEPHT OF RTW000 REPLACING ALL " " BY 0.
EXAMINE WTRLVL UF RTW000 REPLACING ALL " " BY 0.
EXAMINE PPDISH OF RTW146 REPLACING ALL " " BY 0.
EXAMINE ALTUDE OF RTW000 REPLACING ALL " " BY 0.
IF WDEPTH OF RTW000 = 0, MOVE HDEPTH OF RTW000 TO DDEPTH OF SITELN, DDEPTH OF RECOUT
ELSE
    MOVE WDEPTH OF RTW000 TO DDEPTH OF SITELN, DDEPTH OF RECOUT.
    MOVE WDSRCE OF RTW000 TO IDEPTH OF SITELN, IDEPTH OF RECOUT.
    MOVE WTRLVL OF RTW000 TO WTRLVL2, WTRLVL3.
    MOVE WLYEAR OF RTW000 TO WLYEAR OF SITELN.
    MOVE WLDATE TO WLDAT.
    MOVE WLMECH OF RTW000 TO WLMETH2, WLMETH3.
    MOVE CORR RTW000 TO SITELN.
    MOVE CORR RTW000 TO RECOUT.
    GO TO READTAPE.

P001.
    COMPUTE CNTQW1 = CNTQW1 + 1.
    IF (HSW = "Y" AND WDEPTH OF RTW000 NOT = 0) GO TO READTAPE.
    MOVE "H" TO HSW.
    MOVE OLDMAS TO RTQ001.
    EXAMINE WDEPT OF RTQ001 REPLACING ALL " " BY 0.
    IF HSW = "Y", MOVE WDEPTH OF RTQ001 TO DDEPTH OF SITELN, DDEPTH OF RECOUT, GO TO READTAPE.
    GO TO P0001.

P002.
    MOVE "N" TO NSW.
    ADD 1 TO COUNTN, CNTQW2.
    IF HSW = "Y", GO TO READTAPE.
    MOVE OLDMAS TO RTQ002.
    MOVE LCW2ID OF RTQ002 TO LCW2ID OF SITELN.
    MOVE LCW2ID OF RTQ002 TO LCW2ID OF RECOUT.
    GO TO READTAPE.

P003.
    COMPUTE CNTQW3 = CNTQW3 + 1.
    GO TO READTAPE.

P042.
    GO TO READTAPE.

P047.
    GO TO READTAPE.

P055.
    GO TO READTAPE.

P058.
    MOVE OLDMAS TO RTW058.
    IF (ENTNUM > ENTEMP),
    MOVE ENTEMP TO ENTEMP.
    MOVE FINISH OF RTW058 TO FINISH OF RECOUT, FINISH OF SITELN.
    GO TO READTAPE.

P072.
    GO TO READTAPE.

P076.
    MOVE OLDMAS TO RTW076.
    EXAMINE RTW076 REPLACING ALL " " BY 0.
    MOVE CSGDIAM TO DIAM OF RECOUT, DIAM OF SITELN.
    GO TO READTAPE.
PTR082.
MOVE OLDMAS TO RTW082.
EXAMINE OPNTOP OF RTW082 REPLACING ALL " " BY 0.
MOVE OPNTOP OF RTW082 TO OPNTOP OF RECOUT, OPNTOP OF SITELN
MOVE OPNTYP OF RTW082 TO OPNTYP OF RECOUT, OPNTYP OF SITELN.
GO TO READTAPE.

PTR090.
GO TO READTAPE.

PTR094.
GO TO READTAPE.

PTR098.
GO TO READTAPE.

PTR105.
GO TO READTAPE.

PTR114.
GO TO READTAPE.

PTR121.
GO TO READTAPE.

PTR127.
GO TO READTAPE.

PTR134.
GO TO READTAPE.

PTR146.
MOVE OLDMAS TO RTW146.
EXAMINE PPYEAR OF RTW146 REPLACING ALL " " BY 0.
EXAMINE PPDISH OF RTW146 REPLACING ALL " " BY 0.
MOVE PPYEAR OF RTW146 TO PPYEAR OF RECOUT.
IF PPYEAR OF RTW146 = 0000 MOVE SPACES TO PPYEAR OF SITELN
ELSE MOVE PPYEAR OF RTW146 TO PPYEAR OF SITELN.
MOVE PPDISH OF RTW146 TO PPDISH OF RECOUT.
MOVE PPDISH OF RTW146 TO PPDISH OF SITELN.
GO TO READTAPE.

PTR158.
GO TO READTAPE.

PTR164.
GO TO READTAPE.

PTR171.
GO TO READTAPE.

PTR180.
GO TO READTAPE.

PTR183.
GO TO READTAPE.

PTR186.
GO TO READTAPE.

PTR189.
GO TO READTAPE.

PTR192.
GO TO READTAPE.

PTR198.
MOVE OLDMAS TO RTW198.
IF LSW = "X", SET LX TO 1,
    MOVE "Y" TO LSW,
ELSE SET LX UP BY 1.
IF LX > 8, GO TO READTAPE.
MOVE TYPELG OF RTW198 TO LOGTYP (LX).

B-140
USE THE FOLLOWING STATEMENT TO PULL SPECIFIC LOG TYPES
IF (TYPELG OF RTW198 = "G" OR "I" OR "J" OR "N" OR "O" OR "S" OR "U"), MOVE "Y" TO LOGSW.
GO TO READTAPE.
PTR203.
GO TO READTAPE.
PTR208.
GO TO READTAPE.
PTR212.
GO TO READTAPE.
PTR219.
GO TO READTAPE.
PTR234.
MOVE OLDMAS TO RTW234.
MOVE WTRLVL1 TO WLTEST.
IF (CNRECn = 421 OR CNRECn < 425) MOVE 000000 TO WTRLVL1,
WTRLVL2, WTRLVL3.
IF NOT (WLTEST > "000000" OR WLTEST ="000000" OR
WLTEST = "999999") MOVE ZEROS TO WTRLVL1.
EXAMINE WLDATE REPLACING ALL " " BY 0.
EXAMINE WTRLVL1 REPLACING ALL " " BY 0.
EXAMINE WTRLVL1 REPLACING ALL "." BY 0.
COMPUTE WTLCNT = WTLCNT + 1.
IF (WLDATE > WLDTMP),
MOVE WLDATE TO WLDTMP,
MOVE WLYEAR OF RTW234 TO WLYEAR OF SITELn,
WLYEAR OF RECOUT,
IF (WTRLVL1 IS NOT NUMERIC) MOVE ZEROS TO WTRLVL1.
MOVE WTRLVL1 TO WTRLVL2, WTRLVL3,
MOVE WLMETH1 TO WLMETH2, WLMETH3.
GO TO READTAPE.
PTR250.
GO TO READTAPE.
PTR277.
GO TO READTAPE.

NOTE-T.
THIS ROUTINE TESTS THE LINE SEGMENT EQUATIONS
TO DETERMINE IF THE SITE IS WITHIN THE BASIN.
THE FRTSW SWITCH IS SET TO "Y" IF THE SITE
IS WITHIN THE BASIN. EXIT IS TAKEN TO ENDFLE.
IF THE LATITUDE OF THE SITE BEING TESTED IS
GREATER THAN THE HIGH LATITUDE OF THE
CURRENT SLICE. EXIT IS TO READTAPE IF THE
SITE IS OUTSIDE THE RECTANGLE THAT BOUNDS
THE CURRENT SLICE. IF THE SITE IS WITHIN
THE RECTANGLE BUT OUTSIDE THE BASIN EXIT
IS TAKEN TO READTAPE. ONLY IF THE SITE IS
WITHIN THE BASIN IS THE BRANCH TO BRNALL
TAKEN.

TEST2.
IF NLatMS < LATITUDE-1(1), GO TO ENDFLE.
IF NLatMS > LATITUDE-2(1), MOVE EMSG1 TO ENOTE, PERFORM DMPSIT,
GO TO READTAPE.
IF NLngMS < LONGL OF BCARD, MOVE EMSG2 TO ENOTE,
PERFORM DMPSIT, GO TO READTAPE.
IF NLNGMS > HILNG OF BCARD, MOVE EMSG3 TO ENOTE,
    PERFORM DMPST, GO TO READTAPE.
MOVE NLATMS TO LATDMS.
MOVE LATD TO FLATO.
MOVE LATH TO FLATS.
COMPUTE FDCLAT = FLATO + FLATS / 60 + FLATS / 3600.
MOVE NLATMS TO NLNGMS.
MOVE LNGO TO FLNGO.
MOVE LNGM TO FLNGM.
MOVE LNSG TO FLNGS.
COMPUTE FDCLNG = FLNOG + FLNGM / 60 + FLNGS / 3600.
COMPUTE FDCLNG = FDCLNG - 2 * FDCLNG.
SET DA1 TO 1.
TEST21.
    COMPUTE KTEST = ACOEFS (DA1) * FDCLNG + BCOEFS (DA1) * FDCLAT.
    COMPUTE LTEST = ACOEFS (DA1 + 1) * FDCLNG
                   + BCOEFS (DA1 + 1) * FDCLAT.
    IF (KTEST < KNSTNS (DA1) OR LTEST > KNSTNS (DA1 + 1))
       GO TO NEXTB, ELSE GO TO SITEIN.
NEXTB.
    IF (DA1 + 1 = LN2), GO TO READTAPE,
       ELSE SET DA1 UP BY 2, GO TO TEST21.
SITEIN.
    MOVE "Y" TO FRSTSW.
    GO TO BRNALL.
ENDFLE.
    IF LATITUDE-L OF VCARD = 000000, GO TO CLOSEA.
    IF LATITUDE-L OF VCARD < LOLAT OF BCARD,
       MOVE VCARD TO BCARD, GO TO READB1,
       ELSE MOVE "F" TO FRSTW, MOVE "X" TO SLSW, GO TO READV1.
PRTSNL.
    IF LNCNT > 60, PERFORM HDPRTN,
       ELSE ADD 1 TO LNCNT.
    MOVE IDETH OF SITELN TO IDETH OF RECOUT.
    MOVE TYPLOG TO INLOGS OF SITELN, INLOGS OF RECOUT.
    MOVE WTLCNT TO WLCTNT2, WLCTNT3.
    MOVE CNTQW1 TO CNTQ10, CNTQ1S.
    MOVE CNTQW2 TO CNTQ20, CNTQ2S.
    MOVE CNTQW3 TO CNTQ30, CNTQ3S.
ADD 1 TO CNRECN, NDETLN.
MOVE CNRECN TO CNRECP OF SITELN.
MOVE CNRECN TO CNRECR OF RECOUT.
WRITE RECOUT.
WRITE LIST-LINE FROM SITELN.
    IF BLKCNT = 5, MOVE 0 TO BLKCNT, MOVE SPACES TO LIST-LINE, WRITE LIST-LINE, ADD 1 TO LNCNT.
    MOVE SPACES TO TYPLOG, SITELN, RECOUT, WLDMP.
    MOVE ZEROS TO ENTEMP, WTLCNT, CNTQW1, CNTQW2, CNTQW3, CNTQ10,
        CNTQ1S, CNTQ20, CNTQ2S, CNTQ30, CNTQ3S, WLCTNT2, WLCTNT3,
        RTW234, RTW058, RTW076, RTW082.
    MOVE "N" TO LOGSW.

B-142
ADD 1 TO PGCNT.
MOVE PGCNT TO PNUM OF HDR1.
MOVE 0 TO BLKCNT.
MOVE 8 TO LNCNT.
WRITE LIST-LINE FROM HDR1.
WRITE LIST-LINE FROM HDR2.
WRITE LIST-LINE FROM HDR5.
WRITE LIST-LINE FROM HDR3.
WRITE LIST-LINE FROM HDR4.

DBUGGER.
WRITE ERREC FROM STRLN AFTER ADVANCING 2 LINES.
MOVE KLO TO LTD, MOVE KLM TO LTM, MOVE KLS TO LTS, MOVE KLGM TO LGM, MOVE KLGD TO LGD, MOVE KLGS TO LGS.
WRITE ERREC FROM DBGUG1 AFTER ADVANCING 2 LINES.
MOVE NLATMS TO ELATMS, MOVE LATITUDE-1(1) TO ELATITUDE-1, MOVE LATITUDE-2(1) TO ELATITUDE-2, MOVE NLNGMS TO ELNGMS, MOVE LONGLONG TO EONGLONG, MOVE HLING TO EHLING.
WRITE ERREC FROM DBGUG3 AFTER ADVANCING 2 LINES.
MOVE LATITUDE-L OF VCARD TO ELAT-L, MOVE LNGTUD-L OF VCARD TO ELNG-L, MOVE LATITUDE-H OF VCARD TO ELAT-H, MOVE LNGTUD-H OF VCARD TO ELNG-H, MOVE DECLAT-L OF VCARD TO EDL-L, MOVE LS1 TO ELS1, MOVE DECLNG-L OF VCARD TO EDLG-L.
WRITE ERREC FROM DBGUG4 AFTER ADVANCING 2 LINES.
WRITE ERREC FROM DBGUG5 AFTER ADVANCING 2 LINES.
MOVE ASIGN OF VCARD TO EASIN, MOVE ACUEFF OF VCARD TO EACOF, MOVE BSIGN OF VCARD TO EBSIN, MOVE BCUEFF OF VCARD TO EBCOF, MOVE DECLAT-H OF VCARD TO EDL-H, MOVE LS2 TO ELS2, MOVE DECLNG-H OF VCARD TO EDLG-H, MOVE CSIGN OF VCARD TO ECSIN, MOVE KNSTNT OF VCARD TO EKNSTNT.
WRITE ERREC FROM DBGUG6 AFTER ADVANCING 2 LINES.
WRITE ERREC FROM DBGUG7 AFTER ADVANCING 2 LINES.
MOVE LATDMS TO ELATDMS, MOVE FDCLAT TO ELFCLT, MOVE LNGDMS TO ELNGDMS, MOVE FDLNGD TO ELFCLG.
WRITE ERREC FROM DBGUG8 AFTER ADVANCING 2 LINES.
MOVE LATD TO ELATD, MOVE LATM TO ELATM, MOVE LATS TO ELATS, MOVE LNGD TO ELNGD, MOVE LNGM TO ELNGM, MOVE LNGS TO ELNGS.
WRITE ERREC FROM DBGUG9 AFTER ADVANCING 2 LINES.
MOVE FLATD TO EFLATD, MOVE FLATM TO EFLATM, MOVE FLATS TO EFLATS, MOVE FLNGD TO EFLNGD, MOVE FLNGM TO EFLNGM, MOVE FLNGS TO EFLNGS.
IF (FRSW = "F") SET DA1 TO 1.
IF DA1 < 1, SET DA1 TO 1.
WRITE ERREC FROM DBGUG10 AFTER ADVANCING 2 LINES.
MOVE KTEST TO EKTEST, MOVE LTEST TO ELTEST, MOVE KNSTNS (DA1) TO KNSTNS (DA1), COMPUTE DA3 = DA1 + 1, MOVE KNSTNS (DA3) TO EKNSTNS-2, MOVE LN1 TO ELN1, MOVE LN2 TO ELN2.
WRITE ERREC FROM DBGUG11.
MOVE DA1 TO EDA1, MOVE ACOEFS (DA1) TO ECOF-DA1, MOVE ACOEFS (DA3) TO ECOF-DA2, MOVE BCOEFS (DA1) TO EBCOF-DA1, MOVE BCOEFS (DA3) TO EBCOF-DA2.
WRITE ERREC FROM DBUG12.
WRITE ERREC FROM DBUG13.
CLOSEA.
  IF FRTSW = "Y", PERFORM PRTSNL.
*  IF FRTSW = "Y" AND LOGSW = "Y", PERFORM PRTSNL.
  CLOSE FILEB.
  GO TO STOP3.
CLRVSS.
  MOVE 0 TO LATUDE-1 (LA1).
  MOVE 0 TO LNGTUD-1 (LA1).
  MOVE 0 TO LATUDE-2 (LA1).
  MOVE 0 TO LNGTUD-2 (LA1).
  MOVE 0 TO DECLAT-1 (LA1).
  MOVE 0 TO DECLAT-2 (LA1).
  MOVE 0 TO DECLNG-1 (LA1).
  MOVE 0 TO DECLNG-2 (LA1).
  MOVE 0 TO ACOEFS (LA1).
  MOVE 0 TO BCOEFS (LA1).
  MOVE 0 TO KNSTNS (LA1).
PRNTSL.
  MOVE LATUDE-1 (LN1) TO LATUDE-L OF SLN.
  MOVE LNGTUD-1 (LN1) TO LNGTUD-L OF SLN.
  MOVE LATUDE-2 (LN1) TO LATUDE-H OF SLN.
  MOVE LNGTUD-2 (LN1) TO LNGTUD-H OF SLN.
  MOVE DECLAT-1 (LN1) TO DECLAT-L OF SLN.
  MOVE DECLAT-2 (LN1) TO DECLAT-H OF SLN.
  MOVE DECLNG-1 (LN1) TO DECLNG-L OF SLN.
  MOVE DECLNG-2 (LN1) TO DECLNG-H OF SLN.
  MOVE ACOEFS (LN1) TO ACUEFF OF SLN.
  MOVE BCOEFS (LN1) TO BCUEFF OF SLN.
  MOVE KNSTNS (LN1) TO KNSTNT OF SLN.
  IF ACOEFS (LN1) > 0, MOVE "+" TO ASIGN OF SLN
  ELSE MOVE "-" TO ASIGN OF SLN.
  IF BCOEFS (LN1) > 0, MOVE "+" TO BSIGN OF SLN
  ELSE MOVE "-" TO BSIGN OF SLN.
  IF KNSTNS (LN1) > 0, MOVE "+" TO CSIGN OF SLN
  ELSE MOVE "-" TO CSIGN OF SLN.
  IF LNCNT > 60 PERFORM SHPRT, ELSE ADD 1 TO LNCNT.
  WRITE LIST-LINE FROM SLN.
SHPRT.
  ADD 1 TO PGCNT.
  MOVE PGCNT TO PNUM OF HDR1.
  WRITE LIST-LINE FROM HDR1.
  WRITE LIST-LINE FROM HDR2.
  MOVE 4 TO LNCNT.
RIDICT.
  ADD 1 TO PGCNT.
  MOVE PGCNT TO PNUM OF HDR1.
  WRITE LIST-LINE FROM HDR1.
  WRITE LIST-LINE FROM HDR2.
  WRITE LIST-LINE FROM DICT-HDR1.
  WRITE LIST-LINE FROM DICT-HDR2.
WRITE LIST-LINE FROM DICT-HDR3.
WRITE LIST-LINE FROM DICT-LN1.
WRITE LIST-LINE FROM DICT-LN2.
WRITE LIST-LINE FROM DICT-LN3.
WRITE LIST-LINE FROM DICT-LN4.
WRITE LIST-LINE FROM DICT-LN5.
WRITE LIST-LINE FROM DICT-LN6.
WRITE LIST-LINE FROM DICT-LN7.
WRITE LIST-LINE FROM DICT-LN8.
WRITE LIST-LINE FROM DICT-LN9.
WRITE LIST-LINE FROM DICT-LN10.
WRITE LIST-LINE FROM DICT-LN11.
WRITE LIST-LINE FROM DICT-LN12.
WRITE LIST-LINE FROM DICT-LN13.
WRITE LIST-LINE FROM DICT-LN14.
WRITE LIST-LINE FROM DICT-LN15.
WRITE LIST-LINE FROM DICT-LN16.
WRITE LIST-LINE FROM DICT-LN17.
WRITE LIST-LINE FROM DICT-LN18.
WRITE LIST-LINE FROM DICT-LN19.
WRITE LIST-LINE FROM DICT-LN20.
WRITE LIST-LINE FROM DICT-LN21.
WRITE LIST-LINE FROM DICT-LN22.

DMPSIT.
IF EXTSW = "Y", MOVE PCNLX TO ETITLE, MOVE "X" TO EXTSW,
   PERFORM ERHEAD.
   COMPUTE ESEQ = ESEQ + 1
   MOVE ESEQ TO ERSEQ.
   MOVE TFILE8 TO INSEQ.
   MOVE KEYID OF OLDMAS TO ERIDNT.
   IF RNUM OF OLDMAS = 000, MOVE OLDMAS TO RTW000,
      MOVE LCWLID OF RTW000 TO ELCLID.
   IF RNUM OF OLDMAS = 001, MOVE OLDMAS TO RTQ001,
      MOVE SPACES TO ELCLID.
   IF RNUM OF OLDMAS = 002, MOVE OLDMAS TO RTQ002,
      MOVE LCWLID OF RTQ002 TO ELCLID.
   IF RNUM OF OLDMAS = 234, MOVE SPACES TO ELCLID.
   WRITE ERREC FROM ERECLS.
   MOVE SPACES TO ERECLN, ERREC.
   ADD 1 TO ELNCNT.
   IF ELNCNT > 60, ADD 1 TO EPGCNT, MOVE EPGCNT TO EPNUM,
      PFRFORM ERHEAD.

ERHEAD.
   MOVE ZEROS TO ELNCNT.
   WRITE ERREC FROM ERHD1.
   WRITE ERREC FROM ERHD2.
   WRITE ERREC FROM ERHD3.
   WRITE ERREC FROM ERHD4.
   MOVE SPACES TO ERREC.
   WRITE ERREC.
   WRITE ERREC.
   ADD 5 TO ELNCNT.
STOP1.
  WRITE LIST-LINE FROM JOBCTRL.
  DISPLAY "CONTROL INPUT FILE EMPTY".
  STOP RUN.

STOP2.
  WRITE LIST-LINE FROM JOBCTRL.
  DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
  STOP RUN.

STOP3.
  PERFORM PITDICT.
  WRITE LIST-LINE FROM JOBCTRL.
  DISPLAY TFILEB.
  MOVE INCRDS TO CSTAT.
  MOVE NDATIN TO DATAIN.
  MOVE NFLEIA TO FADATA.
  MOVE NDETLN TO DRLNS.
  MOVE NFLEOA TO FOUT.
  WRITE LIST-LINE FROM JBCNLY.
  DISPLAY "NORMAL JOB TERMINATION".
  DISPLAY " COUNT OF BLANKS ", CNBLNK.
  CLOSE CARD-FILE, LIST-FILE, ERRFILE.
  STOP RUN.

STOP4.
  WRITE LIST-LINE FROM JOBCTRL.
  DISPLAY "STOP4 SOURCE ID RECORDS MISSING".
  STOP RUN.

STOP5.
  WRITE LIST-LINE FROM JOBCTRL.
  DISPLAY "STOP5 DATA INPUT FILE END INCORRECT".
  DISPLAY "EOF ACCURRED AT READS OR READV".
  STOP RUN.

STOP6.
  WRITE LIST-LINE FROM JOBCTRL.
  DISPLAY "STOP6 SECOND CARD OF PAIR MISSING".
  STOP RUN.

STOP7.
  WRITE LIST-LINE FROM JOBCTRL.
  DISPLAY "STOP7 FIRST RECORD NOT 000 OR 001".
  DISPLAY KEYIDL, " 7 ERROR 7 ", KEYID OF OLDMAS.
  STOP RUN.

STOP8.
  WRITE LIST-LINE FROM JOBCTRL.
  DISPLAY "STOP8 FILE OUT OF SEQUENCE".
  DISPLAY NHOLD, "LATITUDE-L = ", LATITUDE-L OF VCARD.
  DISPLAY "LATITUDE-H", LATITUDE-H OF VCARD.
  DISPLAY KEYIDL, " 8 ERROR 8 ", KEYID OF OLDMAS.
  STOP RUN.

STOP9.
  DISPLAY "STOP9 RECORD TYPE NOT FOUND".
  DISPLAY OLDMAS.
  STOP RUN.
IDENTIFICATION DIVISION.
PROGRAM-ID. BASN04.
REMARKS. BASN04 IS USED TO OUTPUT A WATER LEVEL HISTORY REPORT FOR A PARTICULAR BASIN FROM THE SWAB/RASA MASTER FILE.

ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. CDC CYBER 175.
OBJECT-COMPUTER. CDC CYBER 175.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT CARD-FILE ASSIGN TO "INPUT".
SELECT LIST-FILE ASSIGN TO "OUTPUT".
SELECT FILEB ASSIGN TO MGQWIN
RESERVE 4 ALTERNATE AREAS
ORGANIZATION IS SEQUENTIAL.
SELECT PLTFILE ASSIGN TO PLTDSK
ORGANIZATION IS SEQUENTIAL.

DATA DIVISION.
FILE SECTION.
FD CARD-FILE
LABEL RECORD OMITTED
DATA RECORD IS CNL-CARD.
01 CNL-CARD.
  03 ID-CNLL PIC IS XXX.
  03 ID-NUM PIC 999.
  03 ID-DES PIC X(74).
FD LIST-FILE
LABEL RECORD OMITTED
DATA RECORD IS LIST-LINE.
01 LIST-LINE.
  03 CC PIC X.
  03 LLPR.
    05 FILLER PIC X.
    05 P0 PIC X(10).
    05 FILLER PIC XXX.
    05 P1 PIC X(10).
    05 FILLER PIC XXX.
    05 P2 PIC X(10).
    05 FILLER PIC XXX.
    05 P3 PIC X(10).
    05 FILLER PIC XXX.
    05 P4 PIC X(10).
    05 FILLER PIC XXX.
    05 P5 PIC X(10).
    05 FILLER PIC XXX.
    05 P6 PIC X(10).
    05 FILLER PIC XXX.
    05 P7 PIC X(10).
    05 FILLER PIC XXX.
    05 P8 PIC X(10).
    05 FILLER PIC XXX.
    05 P9 PIC X(10).
  03 PIND PIC X(4).
FD FILEB
DATA RECORD IS OLDMAS
RECORD CONTAINS 23 TO 327 CHARACTERS
RECORDING MODE IS BINARY
LABEL RECORDS ARE STANDARD.

01 OLDMAS.
  03 KEYID.
    05 KEYLAT.
      07 KLD PIC 99.
      07 KLM PIC 99.
      07 KLS PIC 99.
    05 KEYLNG.
      07 KLGD PIC 999.
      07 KLGM PIC 99.
      07 KLGS PIC 99.
    05 KEYDUP PIC 99.
  03 RNUM PIC IS 999.
  03 LNUM PIC IS 999.
  03 SESAFE PIC X.
  03 RESTRX PIC X OCCURS 1 TO 305 TIMES,
     DEPENDING ON LNUM OF OLDMAS.

FD PLTFLE
DATA RECORD IS RECOUT
LABEL RECORDS ARE OMITTED.

01 RECOUT.
  03 CCK PIC X.
  03 CNCRECR PIC 9(4).
  03 XSP PIC X(4).
  03 INDO PIC 99.
  03 KEYID.
    05 KEYLAT PIC 9(6).
    05 KEYLNG PIC 9(7).
    05 KEYDUP PIC 99.
  03 LCWLID.
    05 LCSITE PIC X(14).
    05 LCREST PIC X(6).
  03 ALTUOE PIC 9(5).
  03 RWDEPTH PIC 9(5).
  03 RWTLEV PIC 9(5)V99.
  03 RWDATE PIC 9(8).
  03 RWRST PIC X(5).

WORKING-STOREAGE SECTION.
77 ALFAA PIC X VALUE "A".
77 ALFABLNK PIC X VALUE " ".
77 ALFAY PIC X VALUE "Y".
77 ALFAX PIC X VALUE "X".
77 ALFAZ PIC X VALUE "Z".
77 CNBLNK PIC 9(6) VALUE 0.
77 CNTSL PIC 9(5) VALUE 0.
77 COUNTA PICTURE 9(3) VALUE 0.
77 COUNT2 PIC 9(6) VALUE 0.
77 COUNT2G PIC 9(6) VALUE 0.
77 CNRQUT PIC 9(5) VALUE 0.
01 LABELF.
  03 IDEXTX PIC XXXX.
  03 FILLER PIC X(17).
  03 REKCNT PIC 9(6).
  03 FILLER PIC X(53).
01 WKDATE.
   03 YY PIC 99.
   03 MM PIC 99.
   03 DY PIC 99.
01 NAME-MONTH.
   03 JAN PIC XXXX VALUE "JAN".
   03 FEB PIC XXXX VALUE "FEB".
   03 MAR PIC XXXX VALUE "MAR".
   03 APR PIC XXXX VALUE "APR".
   03 MAY PIC XXXX VALUE "MAY".
   03 JUNE PIC XXXX VALUE "JUNE".
   03 JULY PIC XXXX VALUE "JULY".
   03 AUG PIC XXXX VALUE "AUG".
   03 SEPT PIC XXXX VALUE "SEPT".
   03 OCT PIC XXXX VALUE "OCT".
   03 NOV PIC XXXX VALUE "NOV".
   03 DEC PIC XXXX VALUE "DEC".
01 MONTH-TABLE REDEFINES NAME-MONTH.
   03 MONTHY PIC xxxx OCCURS 12 TIMES.
01 J0BCN.
   03 CC PIC X VALUE "1".
   03 PCNL.
      05 CNTYPE PIC XXX.
      05 GO-CNL PIC 999.
      05 PCNLX PIC X(74).
   03 FILLER PIC X(52) VALUE SPACES.
01 JBCNXL.
   03 CC PIC X VALUE "0".
   03 PSTATS PIC X(14) VALUE "RECORD COUNTS".
   03 FILLER PIC X(6) VALUE SPACES.
   03 XCARD PIC X(8) VALUE "CARDS IN".
   03 CSTAT PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 XORLNS PIC X(12) VALUE "DETAIL LINES".
   03 DRLNS PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 XDATIZN PIC X(7) VALUE "DATA IN".
   03 DATTN PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 XFAIN PIC X(11) VALUE "OLD-DATA I".
   03 FADATA PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 XNOUT PIC X(12) VALUE "FMT DATA OUT".
   03 FOUT PIC ZZZZZ9.
01 J0BCNX2.
   03 CC PIC X VALUE "0".
   03 JC21 PIC X(10) VALUE "SITES READ".
   03 STAT2A PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 JC22 PIC X(12) VALUE "WATER LEVELS".
   03 STAT22 PIC ZZZZZ9.
   03 FILLER PIC X(8) VALUE SPACES.
   03 JC2G PIC X(13) VALUE "LINES PRINTED".
   03 STAT2G PIC ZZZZZ9.

B-150
01 HDR1.
  03 CC1 PIC 9 VALUE 1.
  03 FILLER PIC X.
  03 LDATE PIC X(6) VALUE "DATE: ".
  03 RMTH PIC X(4).
  03 FILLER PIC X.
  03 ROY PIC Z9.
  03 XB PIC XXXX VALUE ", 19".
  03 RYR PIC XX.
  03 FILLER PIC X(21).
  03 TITLEA PIC X(60).
  03 FILLER PIC X(21).
  03 PGE PIC X(4) VALUE "PAGE".
  03 PNUM PIC ZZZ9.
  03 FILLER PIC X(2).
01 HDR2.
  03 CC PIC 9 VALUE 0.
  03 USGS PIC X(7) VALUE " USGS: ".
  03 GSNAME PIC X(20).
  03 FILLER PIC XXX.
  03 BASINL PIC X(7) VALUE "SOURCES".
  03 FILLER PIC X.
  03 SRCEID PIC X(50).
  03 SRCNAM PIC X(30).
  03 FILLER PIC X(14).
01 HDR3.
  03 CC PIC 9 VALUE 0.
  03 FILLER PIC X VALUE SPACES.
  03 FILLER PIC X(3) VALUE "REC".
  03 FILLER PIC XX.
  03 FILLER PIC X(15) VALUE "SITE IDENTIFIER".
  03 FILLER PIC X(8).
  03 FILLER PIC X(10) VALUE "LOCAL SITE".
  03 FILLER PIC X(9).
  03 FILLER PIC X(4) VALUE "SITE".
  03 FILLER PIC X.
  03 FILLER PIC X(5) VALUE "ALTI-".
  03 FILLER PIC XX.
  03 FILLER PIC X(4) VALUE "WELL".
  03 FILLER PIC X(16).
  03 FILLER PIC X(19) VALUE "WATER LEVEL HISTORY".
  03 FILLER PIC X(30).
01 HDR4.
  03 CC PIC X VALUE SPACES.
  03 FILLER PIC X VALUE SPACES.
  03 FILLER PIC X(3) VALUE "NUM".
  03 FILLER PIC X(4).
  03 FILLER PIC X(3) VALUE "LAT".
  03 FILLER PIC X(4).
  03 FILLER PIC X(4) VALUE "LONG".
  03 FILLER PIC XX.
  03 FILLER PIC XXX VALUE "DUP".

B-151
03 FILLER PIC X(5).
03 FILLER PIC X(10) VALUE "IDENTIFIER".
03 FILLER PIC X(9).
03 FILLER PIC X(4) VALUE "TYPE".
03 FILLER PIC X.
03 FILLER PIC X(4) VALUE "ITUDE".
03 FILLER PIC X(3).
03 FILLER PIC X(5) VALUE "DEPTH".
03 FILLER PIC X(4).
03 FILLER PIC X(5) VALUE "LEVEL".
03 FILLER PIC X(4).
03 FILLER PIC X(4) VALUE "DATE".
03 FILLER PIC X(7).
03 FILLER PIC X(5) VALUE "LEVEL".
03 FILLER PIC X(4).
03 FILLER PIC X(4) VALUE "DATE".
03 FILLER PIC X(7).
03 FILLER PIC X(5) VALUE "LEVEL".
03 FILLER PIC X(4).
03 FILLER PIC X(4) VALUE "DATE".
03 FILLER PIC X(7).
03 FILLER PIC X(5) VALUE "LEVEL".
03 FILLER PIC X(4).
03 FILLER PIC X(4) VALUE "DATE".
03 FILLER PIC X(7).
01 SITELN.
  03 CC PIC 9 VALUE SPACE.
  03 SITEHO.
    05 CNRECP PIC ZZZ9.
    05 FILLER PIC XXX.
    05 KEYID.
      07 KEYLAT PIC 9(6).
      07 FILLER PIC X.
      07 KEYLNG PIC 9(7).
      07 FILLER PIC X.
      07 KEYDUP PIC 99.
    05 FILLER PIC XXX.
    05 LCWLID.
      07 LCSITE PIC X(14).
      07 LCREST PIC X(6).
    05 FILLER PIC XX.
    05 STYP PIC X.
    05 FILLER PIC XX.
    05 ALTUDE PIC ZZZZZ.
    05 FILLER PIC XX.
    05 SDEPTH PIC ZZZZZ.
    05 FILLER PIC X(4).
  03 LEVELT PIC X(62).
  01 LEVELN.
    03 LEVDAT OCCURS 3 TIMES.
      05 LINT PIC ZZZZZ.
      05 LPT PIC X.
      05 LDEC PIC Z.
      05 FILLER PIC XX.
      05 LMO PIC ZZ.
      05 LSYM1 PIC X.
      05 LDAY PIC ZZ.
      05 LSYM2 PIC X.
      05 LYR PIC ZZZZ.
      05 FILLER PIC XX. B-152
01 TMPLVL.
  03 INTLVL PIC 99999V.
  03 OFCLVL PIC 9.
01 DATAAA.
  03 ID-CNL PIC IS XXX.
  03 ID-NUM PIC IS 999.
  03 TITLEX.
    05 SRCAGC.
    07 GNAME PIC X(20).
    07 CNAME PIC X(30).
    05 FILLER PIC X(10).
  03 FILLER PIC X(14).
01 BCARD.
  03 LOLAT PIC 9(6).
  03 LOLNG PIC 9(7).
  03 HILAT PIC 9(6).
  03 HILNG PIC 9(7).
  03 BASIN1 PIC X(7).
  03 BASINX REDEFINES BASIN1.
    05 BASIN2 PIC X(6).
    05 FILLER PIC X.
  03 BTYPE PIC X.
  03 BDESRP PIC X(50).
01 VCARD.
  03 LATITUDE-L PIC 9(6).
  03 LONTUD-L PIC 9(7).
  03 LATITUDE-H PIC 9(6).
  03 LONTUD-H PIC 9(7).
  03 DECLAT-L PIC 99V9999.
  03 LS1 PIC X.
  03 DECLNG-L PIC 999V9999.
  03 DECLAT-H PIC 99V9999.
  03 LS2 PIC X.
  03 DECLNG-H PIC 999V9999.
  03 ASIGN PIC X.
  03 ACOEFF PIC 9V9999999.
  03 BSIGN PIC X.
  03 BCOEFF PIC 9V9999999.
  03 CSIGN PIC X.
  03 KNSINT PIC 999V9999.
  03 VSFQ PIC 9999.
01 LATONE.
  03 LATITUDE-1 PIC 9(6) OCCURS 50 TIMES .
01 LNGONE.
  03 LONTUD-1 PIC 9(7) OCCURS 50 TIMES .
01 LATTWO.
  03 LATITUDE-2 PIC 9(6) OCCURS 50 TIMES .
01 LNGTWO.
  03 LONTUD-2 PIC 9(7) OCCURS 50 TIMES .
01 DATONE.
  03 DECLAT-1 PIC 99V9999 OCCURS 50 TIMES .
01 DNGONE.
  03 DECLNG-1 PIC 999V9999 OCCURS 50 TIMES .
01 DATTWO.
  03 DECLAT-2 PIC 99V9999 OCCURS 50 TIMES.

01 DNGTWO.
  03 DECLNG-2 PIC 999V9999 OCCURS 50 TIMES.

01 ACOEF.
  03 ACOEFS USAGE IS COMP-2 OCCURS 50 TIMES.

01 BCOEF.
  03 BCOEFS USAGE IS COMP-2 OCCURS 50 TIMES.

01 CCOEF.
  03 KNSTNS USAGE IS COMP-2 OCCURS 50 TIMES.

01 LATDMS.
  03 LATD PIC 99.
  03 LATM PIC 99.
  03 LATS PIC 99.

01 LNGDMS.
  03 LNGD PIC 999.
  03 LNGM PIC 99.
  03 LNSG PIC 99.

01 TYPLOG.
  03 LOGTYP PIC X OCCURS 8 TIMES.

01 RTWO00.
  03 KEYID.
    05 KEYLAT PIC 9(6).
    05 KEYLNG PIC 9(7).
    05 KEYDUP PIC 99.
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 SITETY PIC X.
  03 DRELIX PIC X.
  03 REPAGY PIC X(5).
  03 PROJNU.
    05 BASIN1 PIC X(7).
    05 RASIN2 PIC X(6).
  03 DISTRT PIC XXX.
  03 STATEW PIC XX.
  03 COUNTY PIC XXX.
  03 LLACUR PIC X.
  03 LCLID.
    05 LCSITE PIC X(14).
    05 LCREST PIC X(6).
  03 LCLNET PIC X(30).
  03 LCLMAP PIC X(30).
  03 SCALEN PIC X(6).
  03 ALTUDE PIC 9(5)V99.
  03 MMEASR PIC X.
  03 MACCUR PIC XXX.
  03 TOPSET PIC X.
  03 UWDCUR PIC X(8).
  03 FCMNTH PIC XX.
  03 FCCDAY PIC XX.
  03 FCYEAR PIC XXXX.
  03 FILLER PIC X.
  03 SITUSE PIC X.
  03 WTRUSE PIC X.
  03 SWTRUS PIC X.
03 TWTRUS PIC X.
03 WDEPTH PIC 9(5)V99.
03 WDEPTH PIC 9(5)V99.
03 WDSRCE PIC X.
03 WTRLVL PIC 9(5)V99.
03 WLMAT.
   05 WLMMTH PIC XX.
   05 WLLDAY PIC XX.
   05 WLYEAR PIC XXXX.
03 WLDACR PIC X.
03 WLSRCE PIC X.
03 WLMETH PIC X.
03 PUMPUS PIC X.
03 GHSRCE PIC X.
03 XSTATUS PIC X.
03 LUPDAT PIC X(8).
03 VERIFY PIC X.
03 MSHGHT PIC 999V99.
03 MSDATE PIC 9(8).
03 CURECD PIC X.
01 RTW234.
  03 KEYID.
     05 KEYLAT PIC 9(6).
     05 KEYLNG PIC 9(7).
     05 KEYDUP PIC 99.
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 WLDATE.
     05 WLMMTH PIC 99.
     05 WLDAY PIC 99.
     05 WLYEAR PIC 9999.
  03 DTACCR PIC X.
  03 WTRLVL PIC 9(4)V999.
  03 WLMETH PIC X.
  03 MEASAC PIC X.
01 RTQ002.
  03 KEYID PIC 9(15).
  03 RNUM PIC 999.
  03 LNUM PIC 999.
  03 SESAFE PIC X.
  03 LCWLID PIC X(20).
  03 LCREST PIC X(44).
01 BLNKHD.
  03 HEAD PIC X(69).
  03 BLVL PIC X(62).
PROCEDURE DIVISION.
DECLARATIVES.
SFILEB SECTION.
USE AFTER STANDARD ENDING FILE LABEL
PROCEDURE ON FILEB.
LFILEB.
MOVE FILE-LABEL TO LABELF.
MOVE TFILEB TO REKCNT.
MOVE LABELF TO FILE-LABEL.
DISPLAY FILE-LABEL, TFILEB, REKCNT.
END DECLARATIVES.
*NOTE-A.
* THE PROGRAM STARTS WITH THIS INITIALIZATION
* ROUTINE THAT OPENS THE INPUT AND LISTING FILES
* AND THEN THE DATE STORED IN THE SYSTEM IS READ
* AND USED TO SET UP THE DATE IN THE FIRST
* HEADING LINE. NO RETURN FROM THE MAINLINE
* ROUTINE IS MADE TO THIS PROCEDURE.
BEGIN.
OPEN INPUT CARD-FILE.
OPEN OUTPUT LIST-FILE.
ACCEPT WKDATE FROM DATE.
MOVE MM TO NMM.
MOVE MONTHY (NMM) TO RMTH OF HDR1.
MOVE DY TO RDY OF HDR1.
MOVE YY TO RYR OF HDR1.
MOVE SPACES TO WLOTMP.
MOVE ZEROS TO ENTEMP.
*NOTE-B.
* THIS ROUTINE READS THE CONTROL CARDS AND MOVES
* THE GENERAL DESCRIPTION OF THE REPORT TO THE
* HEADING AREA. THEN A BRANCH IS TAKEN TO THE
* APPROPRIATE STARTING ROUTINE DEPENDING ON THE
* NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD.
* THERE IS NO RETURN TO THIS ROUTINE.
READC.
READ CARD-FILE AT END GO TO STOP1.
IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
ADD 1 TO INCRDS.
MOVE CNL-CARD TO PCNL.
READ CARD-FILE INTO DATAA AT END GO TO STOP4.
IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
IF ID-NUM OF DATAA = 010 MOVE TITLEX OF DATAA
TO TITLEA OF HDR1 ELSE GO TO STOP4.
ADD 1 TO INCRDS.
IF GO-CNL = 001, GO TO STARTA, ELSE GO TO STOP2.
STARTA.
OPEN INPUT FILEB.
OPEN OUTPUT PLTFLE.
READB.
READ CARD-FILE INTO BCARD AT END GO TO STOP5.
ADD 1 TO INCROS.
*NOTE-C.*
* NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE
* THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING
* ROUTINES. A RETURN TO READB1 IS MADE FROM THE
* ROUTINE ENDFLE WHEN A TAPE RECORD IS LESS THAN THE
* LOWEST LATITUDE IN THE CURRENT BASIN AND THERE
* IS NOT A TERMINATION RECORD SIGNALLED BY 000000 IN
* THE FIRST SIX POSITIONS OF THE CURRENT RECORD.

READB1.
MOVE BDESRP OF BCARD TO SRCEID OF HDR2.
MOVE BASIN1 OF BCARD TO BASINL OF HDR2.

*NOTE-D.*
* THE ROUTINES FROM HERE TO THE ROUTINE
* FINDPT READ AND STORE THE SLICES FOR THE
* BASIN WHILE EDITING AND COMPUTING THE
* COEFFICIENTS FOR THE LINE SEGMENTS.
* EXIT IS TAKEN WHEN THE FIRST RECORD
* OF THE NEXT SLICE IS READ AND A BRANCH
* IS TAKEN TO THE ROUTINE, FINDPT.
* THE RETURN IS TO READV1 SINCE THE FIRST
* RECORD OF THE NEXT SLICE HAS NOT BEEN
* STORED IN THE TABLE.

READV.
READ CARD-FILE INTO VCARD AT END GO TO STOPS.
ADD 1 TO INCGRDS.

READV1.
IF FRSW = "F", PERFORM CLRVSS VARYING LA1 FROM 1 BY
1 UNTIL LA1 = 51,
MOVE "Y" TO FRSW, SET LA1 TO 1,
MOVE LATUDE-L OF VCARD TO NHOLD.
IF LATUDE-L OF VCARD < NHOLD, GO TO FINDPT.

STRVCD.
IF LATUDE-L OF VCARD > NHOLD, GO TO STOP8.
EXAMINE LATUDE-L OF VCARD REPLACING ALL " " BY 0.
MOVE LATUDE-L OF VCARD TO LATUDE-1 (LA1).
EXAMINE LNGTUD-L OF VCARD REPLACING ALL " " BY 0.
MOVE LNGTUD-L OF VCARD TO LNGTUD-1 (LA1).
EXAMINE LATUDE-H OF VCARD REPLACING ALL " " BY 0.
MOVE LATUDE-H OF VCARD TO LATUDE-2 (LA1).
EXAMINE LNGTUD-H OF VCARD REPLACING ALL " " BY 0.
MOVE LNGTUD-H OF VCARD TO LNGTUD-2 (LA1).
EXAMINE OECLAT-L OF VCARD REPLACING ALL " " BY 0.
MOVE OECLAT-L OF VCARD TO OECLAT-1 (LA1).
EXAMINE OECLNG-L OF VCARD REPLACING ALL " " BY 0.
MOVE OECLNG-L OF VCARD TO OECLNG-1 (LA1).
EXAMINE OECLAT-H OF VCARD REPLACING ALL " " BY 0.
MOVE OECLAT-H OF VCARD TO OECLAT-2 (LA1).
EXAMINE OECLNG-H OF VCARD REPLACING ALL " " BY 0.
MOVE OECLNG-H OF VCARD TO OECLNG-2 (LA1).
EXAMINE ACOEFF OF VCARD REPLACING ALL " " BY 0.
MOVE ACOEFF OF VCARD TO ACOEFS (LA1).
IF ASIGN OF VCARD = ",", COMPUTE ACOEFS (LA1) =
ACOEFS (LA1) - 2 * ACOEFS (LA1),
ELSE MOVE "+" TO ASIGN OF VCARD.
EXAMINE BCOEFF OF VCARD REPLACING ALL " " BY 0.
MOVE BCOEFF OF VCARD TO BCOEFS (LA1).
IF 8SIGN OF VCARD = "-", COMPUTE BCOEFS (LA1) =
BCOefs (LA1) - 2 * BCOEFS (LA1),
ELSE MOVE "+" TO 8SIGN OF VCARD.
EXAMINE KNSTNT OF VCARD REPLACING ALL " " BY 0.
MOVE KNSTNT OF VCARD TO KNSTNS (LA1).
IF CSIGN OF VCARD = "-", COMPUTE KNSTNS (LA1) =
KNSTNS (LA1) - 2 * KNSTNS (LA1),
ELSE MOVE "+" TO CSIGN OF VCARD.
SET LN2 TO LA1.
SET LA1 UP BY 1.
GO TO READV.

*NOTE-E.
* THIS IS THE TAPE READ ROUTINE THAT CONTROLS THE MAIN
* PROGRAM. THE KEY ROUTINES ARE READTAPE, BRNALL,
* CHNG01, CHNG02, TEST2.
* THIS PROCEDURE TESTS THE FIRST RECORD TAPE
* SWITCH FOR THE SITE BEING OUTSIDE THE BASIN
* AND CONTINUES TO READ THE MASTER FILE UNTIL
* A SITE WITHIN THE BASIN IS FOUND.
* RECORDS ARE PROCESSED FOR THAT SITE UNTIL
* A CHANGE OF SITE KEYID IS DETECTED. A
* BRANCH IS MADE TO CHNG01 WHERE THE OUTPUT
* FOR THE PREVIOUS IS MADE IF APPLICABLE.
* THEN IN CHNG02 THE LAST SITE RECORDS ARE
* CLEARED AND THE NEW SITE IS SET UP FOR
* SEARCH AND PROCESSING.
FINQPT.
MOVE "F" TO FRsw.
IF (SLSW = "X") MOVE "Y" TO SLSW, GO TO TEST2.
GO TO READTAPE.

READTAPE.
READ FILEB INTO LIST-LINE AT END GO TO CLOSEA.
ADD 1 TO TFREB, NFLEIA.
IF KEYID OF OLDMAS NOT = KEYIDL, GO TO CHNG01.

*NOTE-WELL.
* THE BRANCH INSTRUCTIONS IN THE PROCEDURE, BRNALL,
* ARE ORDERED BY THEIR RELATIVE FREQUENCY IN THE
* WRDM MASTER FILE. REARRANGEMENT WOULD SLOW
* THE OPERATION.
BRNALL.
IF FRTSW = "X", GO TO READTAPE.
IF RNUM OF OLDMAS = 234, GO TO PTR234.
IF RNUM OF OLDMAS = 000, GO TO PTR000.
IF RNUM OF OLDMAS = 001, GO TO PTR001.
IF RNUM OF OLDMAS = 002, GO TO PTR002.
IF RNUM OF OLDMAS = 003, GO TO PTR003.
IF RNUM OF OLDMAS = 058, GO TO PTR058.
IF RNUM OF OLDMAS = 076, GO TO PTR076.
IF RNUM OF OLDMAS = 198, GO TO PTR198.
IF RNUM OF OLDMAS = 042, GO TO PTR042.
IF RNUM OF OLDMAS = 082, GO TO PTR082.
IF RNUM OF OLDMAS = 158, GO TO PTR158.
IF RNUM OF OLDMAS = 192, GO TO PTR192.
IF RNUM OF OLDMAS = 146, GO TO PTR146.
IF RNUM OF OLDMAS = 186, GO TO PTR186.
IF RNUM OF OLDMAS = 090, GO TO PTR090.
IF RNUM OF OLDMAS = 114, GO TO PTR114.
IF RNUM OF OLDMAS = 121, GO TO PTR121.
IF RNUM OF OLDMAS = 189, GO TO PTR189.
IF RNUM OF OLDMAS = 094, GO TO PTR094.
IF RNUM OF OLDMAS = 183, GO TO PTR183.
IF RNUM OF OLDMAS = 072, GO TO PTR072.
IF RNUM OF OLDMAS = 047, GO TO PTR047.
IF RNUM OF OLDMAS = 134, GO TO PTR134.
IF RNUM OF OLDMAS = 171, GO TO PTR171.
IF RNUM OF OLDMAS = 180, GO TO PTR180.
IF RNUM OF OLDMAS = 127, GO TO PTR127.
IF RNUM OF OLDMAS = 055, GO TO PTR055.
IF RNUM OF OLDMAS = 098, GO TO PTR098.
IF RNUM OF OLDMAS = 105, GO TO PTR105.
IF RNUM OF OLDMAS = 164, GO TO PTR164.
IF RNUM OF OLDMAS = 203, GO TO PTR203.
IF RNUM OF OLDMAS = 208, GO TO PTR208.
IF RNUM OF OLDMAS = 212, GO TO PTR212.
IF RNUM OF OLDMAS = 219, GO TO PTR219.
IF RNUM OF OLDMAS = 250, GO TO PTR250.
IF RNUM OF OLDMAS = 277, GO TO PTR277.
GO TO STOP9.

CHNG01.
IF (FRTSW = "Y" AND LDSW = ALFAX) PERFORM PRTSNL,
   MOVE ZEROS TO CNTSL, MOVE SPACES TO LIST-LINE,
   ADD 1 TO LNCNT, WRITE LIST-LINE.
   MOVE ALFAZ TO PRTSW.
   MOVE ALFAY TO SW0, SW1, SW2, LDSW.

CHNG02.
   MOVE KEYID OF OLDMAS TO KEYIDL.
   MOVE SPACES TO SITELN.
   MOVE SPACES TO RECOUT.
   MOVE "X" TO FRTSW, HSW, NSW, LSW.
   MOVE KEYLAT OF OLDMAS TO NLATMS.
   MOVE KEYLNG OF OLDMAS TO NLANGMS.
   GO TO TEST2.

*NOTE-P.*
* THE FOLLOWING ROUTINES ARE IN PLACE TO *
* ALLOW PROCESSING OF ANY RECORD IN THE *
* (WRDM) WATER RESOURCES MASTER TAPE FILE *
* DATA MAY BE EXTRACTED AND STORED IN THE *
* LAST 41 POSITIONS OF "SITELN" AND "RECOUT" *
* FOR STORING ON YOUR CYBER DISK FILE AND *
* PRINTING ON THE LISTING.*
* *
* ADDITIONAL OUTPUT MAY BE HANDLED BY *
* SETTING UP ADDITIONAL OUTPUT RECORDS *
* AND INSERTING THE PROPER INSTRUCTIONS *
* TO ADD TO THE OUTPUT

B-159
PTRO00.
  MOVE ALFAX TO SW0.
  SET WTLCNT TO ZERO.
  ADD 1 TO COUNTA.
  MOVE "Y" TO HSW.
  MOVE OLDMAS TO RTW000.
  IF ( SITETY OF RTW000 = "S" ),
    GO TO READTAPE.
PRMOVE.
  MOVE KEYLAT OF RTW000 TO KEYLAT OF SITELN,
   KEYLAT OF RECOUT.
  MOVE KEYLNG OF RTW000 TO KEYLNG OF RECOUT, KEYLNG OF SITELN.
  MOVE KEYDUP OF RTW000 TO KEYDUP OF RECOUT, KEYDUP OF SITELN.
  MOVE LCWLID OF RTW000 TO LCWLID OF RECOUT, LCWLID OF SITELN.
  EXAMINE ALTUDE OF RTW000 REPLACING ALL " " BY 0.
  MOVE ALTUDE OF RTW000 TO ALTUDE OF RECOUT, ALTUDE OF SITELN.
  EXAMINE WDEPTH OF RTW000 REPLACING ALL " " BY 0.
  MOVE WDEPTH OF RTW000 TO RWDEPTH.
  EXAMINE WLDAT OF RTW000 REPLACING ALL " " BY 0.
  EXAMINE WTRLVL OF RTW000 REPLACING ALL " " BY 0.
  IF (WTRLVL OF RTW000 = 0) GO TO READTAPE, ELSE
    ADD 1 TO WTLCNT,
    MOVE WTRLVL OF RTW000 TO TMPLVL,
    MOVE INTLVL TO LINT (WTLCNT),
    MOVEDECLVL TO LDEC (WTLCNT),
    MOVE "." TO LPT (WTLCNT),
    MOVE WLMNTH OF RTW000 TO LMO (WTLCNT),
    MOVE WLDAY OF RTW000 TO LDAY (WTLCNT),
    MOVE WLYEAR OF RTW000 TO LYR (WTLCNT),
    MOVE "/" TO LSYM1 (WTLCNT), LSYM2 (WTLCNT),
    MOVE ALFAX TO LSDW.
    GO TO READTAPE.
PTR001.
  MOVE ALFAX TO SW1.
  IF NOT (SW0 = ALFAX), PERFORM READTAPE,
    IF RNUM OF OLDMAS = 002, MOVE OLDMAS TO RTQ002,
      MOVE LCWLID OF RTQ002 TO LCWLID OF SITELN.
    GO TO READTAPE.
PTR002.
  MOVE "N" TO NSW.
  MOVE ALFAX TO SW2.
  IF LSDW = ALFAY, MOVE LCWLID OF RTQ002 TO LCWLID OF SITELN.
  IF LCWLID OF SITELN = ALFARLNK,
    MOVE "NO LOCAL ID AVAILABLE" TO LCWLID OF SITELN.
  GO TO READTAPE.
PTR003.
  GO TO READTAPE.
PTR042.
  GO TO READTAPE.
PTR047.
  GO TO READTAPE.
PTR055.
  GO TO READTAPE.
PTR058.
  GO TO READTAPE.
PTR072.
  GO TO READTAPE.
PTR076.
  GO TO READTAPE.
PTR082.
  GO TO READTAPE.
PTR090.
  GO TO READTAPE.
PTR094.
  GO TO READTAPE.
PTR098.
  GO TO READTAPE.
PTR105.
  GO TO READTAPE.
PTR114.
  GO TO READTAPE.
PTR121.
  GO TO READTAPE.
PTR127.
  GO TO READTAPE.
PTR134.
  GO TO READTAPE.
PTR146.
  GO TO READTAPE.
PTR158.
  GO TO READTAPE.
PTR164.
  GO TO READTAPE.
PTR171.
  GO TO READTAPE.
PTR180.
  GO TO READTAPE.
PTR183.
  GO TO READTAPE.
PTR186.
  GO TO READTAPE.
PTR189.
  GO TO READTAPE.
PTR192.
  GO TO READTAPE.
PTR198.
  GO TO READTAPE.
PTR203.
  GO TO READTAPE.
PTR208.
  GO TO READTAPE.
PTR212.
  GO TO READTAPE.
PTR219.
  GO TO READTAPE.

B-161
PTR234.

IF NOT (SW0 = ALFAX) MOVE
"NO SITE HEADER REC" TO LCWLID OF SITELN,
MOVE KEYLAT OF RTW234 TO KEYLAT OF SITELN,
MOVE KEYLNG OF RTW234 TO KEYLNG OF SITELN,
MOVE KEYDUP OF RTW234 TO KEYDUP OF SITELN.
ADD 1 TO COUNT2.
MOVE OLDMAS TO RTW234.
IF (WTLCNT = 3) ADD 1 TO CNTSL, PERFORM PRTSNL,
SET WTLCNT TO 0.
EXAMINE WLDATE OF RTW234 REPLACING ALL " " BY 0.
EXAMINE WTRLVL OF RTW234 REPLACING ALL " " BY 0.
IF (WTRLVL OF RTW234 = 0), GO TO READTAPE, ELSE
ADD 1 TO WTLCNT,
MULTIPLY WTRLVL OF RTW234 BY 0.1 GIVING WTRLVL OF RTW234,
MOVE WTRLVL OF RTW234 TO TMPLVL, RWTLEV,
MOVE INTLVL TO LINT (WTLCNT),
MOVE DECLVL TO LDEC (WTLCNT),
MOVE "." TO LPT (WTLCNT),
MOVE WLMTNTH OF RTW234 TO LMO (WTLCNT),
MOVE WLDAY OF RTW234 TO LDAY (WTLCNT),
MOVE WLYEAR OF RTW234 TO LYR (WTLCNT),
MOVE RECNT TO CNRECR.
MOVE WLDATE TO RWDATE.
EXAMINE RWDATE REPLACING ALL " " BY 0.
EXAMINE RWTLLEV REPLACING ALL " " BY 0.
MOVE SPACES TO CCR, RWRST, XSP.
MOVE 88 TO INDO.
WRITE RECOUT.
ADD 1 TO CNROUT.
MOVE "/" TO LSYM1 (WTLCNT), LSYM2 (WTLCNT).
MOVE ALFAX TO LDSW.
GO TO READTAPE.
PTR250.
GO TO READTAPE.

PTR277.
GO TO READTAPE.

*NOTE-T.*
*This routine tests the line segment equations*
*to determine if the site is within the basin.*
*The FRTSW switch is set to "Y" if the site*
*is within the basin. Exit is taken to ENDFLE*
*if the latitude of the site being tested is*
*greater than the high latitude of the*
*current slice. Exit is to READTAPE if the*
*site is outside the rectangle that bounds*
*the current slice. If the site is within*
*the rectangle but outside the basin exit*
*is taken to READTAPE. Only if the site is*
*within the basin is the branch to BRNALL*
*taken.*
TEST2.
   IF NLATMS < LATUDE-1(1), GO TO ENDFLE.
   IF NLATMS > LATUDE-2(1), GO TO READTAPE.
   IF NLNGMS < LULNG OF BCARD, GO TO READTAPE.
   IF NLNGMS > HILNG OF BCARD, GO TO READTAPE.
   MOVE NLATMS TO LATDMS.
   MOVE LATO TO FLATO.
   MOVE LATM TO FLATM.
   MOVE LATS TO FLATS.
   COMPUTE FDCLAT = FLATD + FLATM / 60 + FLATS / 3600.
   MOVE NLNGMS TO LNGDMS.
   MOVE LNGD TO FLNGD.
   MOVE LNGM TO FLNGM.
   MOVE LNGS TO FLNGS.
   COMPUTE FDCLNG = FLNGD + FLNGM / 60 + FLNGS / 3600.
   COMPUTE FDCLNG = FDCLNG - 2 * FDCLNG.
   SET DA1 TO 1.

   TEST21.
   COMPUTE KTEST = ACOEFS (DA1) * FDCLNG + BCOEFS (DA1) * FDCLAT.
   COMPUTE LTEST = ACOEFS (DA1 + 1) * FDCLNG
                  + BCOEFS (DA1 + 1) * FDCLAT.
   IF (KTEST < KNSTNS (DA1) OR LTEST > KNSTNS (DA1 + 1))
      GO TO NEXTB, ELSE GO TO SITEIN.

   NEXTB.
   IF (DA1 + 1 = LN2), GO TO READTAPE,
      ELSE SET DA1 UP BY 2, GO TO TEST21.

   SITEIN.
   MOVE "y" TO FRTSW.
   GO TO BRNALL.

   ENDFLE.
   IF LATUDE-L OF VCARD = 000000, GO TO CLOSEA.
   IF LATUDE-L OF VCARD < LOLAT OF BCARD,
      MOVE VCARO TO BCARD, GO TO REAO81,
      ELSE MOVE "F" TO FRTSW, MOVE "X" TO SLSW, GO TO READV1.

   PRTSNL.
   ADD 1 TO COUNT2G.
   IF (CNTSL NOT > 1) ADD 1 TO RECNT.
   IF LNCNT > 59, PERFORM HDPRNT,
      ELSE ADD 1 TO LNCNT.
   ADD 1 TO CNTSL, PERFORM CLRSHD.
   IF (WTLCNT = 1) PERFORM CLRLVL 2 TIMES.
   IF (WTLCNT = 2) PERFORM CLRLVL 1 TIMES.
   MOVE LEVLN TO LEVDAT,
   MOVE RECNT TO CNRRECP OF SITELN,
   IF LDSW = ALFAX,
      IF PRTSW = "A", MOVE LEVLN TO BLVL,
         WRITE LIST-LINE FROM BLNKHD
      ELSE WRITE LIST-LINE FROM SITELN.
   MOVE SPACES TO BLNKHD.
   MOVE SPACES TO LIST-LINE.
   MOVE ZEROS TO WTLCNT.
   PERFORM CLRLVL 3 TIMES.
   MOVE "A" TO PRTSW.

B-163
HDPRT1.
ADD 1 TO PGCNT.
MOVE PGCNT TO PNUM OF HOR1.
SET LNCNT TO 10.
WRITE LIST-LINE FROM HDR1.
WRITE LIST-LINE FROM HDR2.
WRITE LIST-LINE FROM HDR3.
WRITE LIST-LINE FROM HDR4.
MOVE SPACES TO LIST-LINE.
WRITE LIST-LINE.
WRITE LIST-LINE.
CLOSEA.
IF FRTSW = "Y", PERFORM PRTSNL.
CLOSE FILEB.
GO TO STOP3.
CLRVSS.
MOVE 0 TO LATUDE-1 (LA1).
MOVE 0 TO LNGTUD-1 (LA1).
MOVE 0 TO LATUDE-2 (LA1).
MOVE 0 TO LNGTUD-2 (LA1).
MOVE 0 TO DECLAT-1 (LA1).
MOVE 0 TO DECLNG-1 (LA1).
MOVE 0 TO DECLAT-2 (LA1).
MOVE 0 TO DECLNG-2 (LA1).
MOVE 0 TO ACOEFS (LA1).
MOVE 0 TO RCOEFS (LA1).
MOVE 0 TO KNSTNS (LA1).
CLRSHD.
MOVE SITELN TO BLNKHD.
MOVE SPACES TO HEAD.
CLRLVL.
ADD 1 TO WTLCNT.
MOVE ZEROS TO LINT (WTLCNT).
MOVE SPACES TO LPT (WTLCNT).
MOVE ZEROS TO LDEC (WTLCNT).
MOVE ZEROS TO LMO (WTLCNT).
MOVE SPACES TO LSYM1 (WTLCNT).
MOVE ZEROS TO LDAY (WTLCNT).
MOVE SPACES TO LSYM2 (WTLCNT).
MOVE ZEROS TO LYR (WTLCNT).
STOP1.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "CONTROL INPUT FILE EMPTY".
STOP RUN.
STOP2.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
STOP RUN.
STOP3.
WRITE LIST-LINE FROM JOBCNL.
DISPLAY TFILEB.
MOVE INCRDS TO CSTAT.
MOVE NDATIN TO DATAIN.
MOVE NFLEIA TO FADATA.
MOVE NDETLN TO DRLN$.
MOVE NFLEOA TO FOUT.
MOVE COUNTA TO STAT2A.
MOVE COUNT2 TO STAT22.
MOVE COUNT2G TO STAT2G.
WRITE LIST-LINE FROM JBCNLX.
WRITE LIST-LINE FROM JOBCLNX.
DISPLAY "NORMAL JOB TERMINATION".
DISPLAY " COUNT OF BLANKS ", CNBLNK.
CLOSE CARD-FILE, LIST-FILE, PLTFLE.
STOP RUN.

STOP4.
WRITE LIST-LINE FROM JBCNL.
DISPLAY "STOP4 SOURCE ID RECORDS MISSING".
STOP RUN.

STOP5.
WRITE LIST-LINE FROM JBCNL.
DISPLAY "STOP5 DATA INPUT FILE END INCORRECT".
DISPLAY "EOF ACCURRED AT READB OR READV".
STOP RUN.

STOP8.
WRITE LIST-LINE FROM JBCNL.
DISPLAY "STOP8 FILE OUT OF SEQUENCE".
DISPLAY NHOLD, "LATITUDE-L = ", LATITUDE-L OF VCARD.
DISPLAY "LATITUDE-H", LATITUDE-H OF VCARD.
DISPLAY KEYIDL, " 8 ERROR 8 ", KEYID OF OLDMAS.
STOP RUN.

STOP9.
DISPLAY "STOP9 RECORD TYPE NOT FOUND".
DISPLAY OLDMAS.
STOP RUN.
PROGRAM BSNPLT (INPUT, OUTPUT, TAPE5=INPUT, TAPE6=OUTPUT
1, PUNCH, TAPE7=PUNCH, TAPE15, TAPE16, TAPE17, TAPE18, TAPE19
2, TAPE20, TAPE21, TAPE22, TAPE23, TAPE24, TAPE25, TAPE26, TAPE27
3, TAPE28, TAPE29, TAPE30, TAPE31, TAPE32, TAPE33, TAPE34, TAPE35
4, TAPE46)
C......
C...... MULTIPLE UNITS ARE USED TO ACCOMMODATE THE RUNNING
C...... OF MULTIPLE PLOTS ON ONE RUN OF THE PROGRAM
C......
DIMENSION LPROJ(4), NSTDP(2), LCMS(2), LCWLID(20)
DIMENSION KEYID(2)
C......
C...... THIS COMMON STORAGE AREA IS USED BY THE MAIN ROUTINE
C...... AND THE READ CONTROL CARD ROUTINE PRIMARILY
C......
COMMON/MALINE,JSTR,TITLE(8),XS(300),YS(300)
1, KSYM(300), XLS(300), YLS(300), FHAT, FLLAT, FHNG, FLLNG, CM, ZHT
2, SYHT, IPAGE, NPLLOT, SCALE, XHAT, YHNG, XLAT, YLLNG, TITLE2(8)
3, TITLE3(8), TLA, TLN, FITIC, LLAT, KHLAT, LLNG, KHLNG, KOLNE, LOP, NCM
4, TITLE4(8), TITLF5(8), TITLE6(8), TITLE7(8), NF, NOTN(20)
5, NDS, NBRDR
C......
COMMON/SELECT/CNYR1, CNYR2, CDIAM, ILOG(8), IFIN(3), IOPN(3), IWUSE(3)
$1, ISTSW, NFIN, NWP, NWLVL, NWUS, NWELE, SPCAP, TSTEP, TSTOPN, TSWL
2, TSWDP, TSTDIM, WLYR1, WLYR2, WTRLVL, XDEPT, OPNTP
C......
COMMON/RDR/ALT, CSMD, CNSYR, DEPT, INWU, LOGS(8), MFIN, MOPN, SPCP
1, TOPPN, WL, WLYR, WLCNT
C......
COMMON/LBLR/I, NID1, NID2, OLBL(300), OLBL2(300), BTMALT, WLALT
1, TQPALT, SEQ
C......
COMMON/SYMLD/XL, YL, KSP, HT, ANGLE, NDM
C......
C...... DEFAULT VALUES FOR THE LAMBERT CONFORMAL PROJECTION
C......
DATA PH33, PH44, AA, AB, CM, SCALE/33., 45., 251109960.5,
1250258678.8, 112., 126720./
C......
DATA LPROJ(1), LPROJ(2), LPROJ(3), LPROJ(4)/"LAMBERT CO",
1"NFORMAL CO", "NIC PROJECT", "TION /
C......
DATA NSTDP(1), NSTDP(2)/"STANDARD P", "ARALLELS "/
DATA LCMS(1), LCMS(2)/"CENTRAL ME", "RDIAN ="/
DATA BMSG/"BEGIN PLOT"/
DATA HT, ANGLE, KSP/0.08, 90.0, 1/
C......
C...... NPLOT IS INCREASED BY 1 FOR EACH PLOT RUN
C...... KDISK IS FILE CONTAINING THE SLICE TABLE FROM PLOYGON PROGRAM
C......
CALL PLOTS(0, 0, 46)
NPLOT = 0
KDISK = 20
KDISK = 15
B-166
C...... FOR MULTIPLE PLOTS:
C...... THIS IS THE RESTART POINT FOR PLOTS AFTER THE FIRST
C......
    100 EDOSW = -1.
          I = 0
          SEQ = 0.0
          KGOSW = -1
          NFIE = 5
          NDPLT = -1
          XE = 2.0
          KEYID = 999999
          FSLC = 1.0
          EOFSL = 0.0
          ITBSW = 0
          SLCNT = 0.0
          XHT = ZHT
          NBRDR = 1
C......
C...... THE VARIABLES FOR THIS CONTROL CARD READ ROUTINE
C...... ARE STORED IN THE COMMON MALINE AREA
C......
C...... ALL HEADING FOR THE PLOT IS DONE IN THE NREAD ROUTINE
C...... FOR OUTPUT AFTER THE PLOT FILE IS OPENED
C......
    CALL NREAD3
    LNCNT=6
    WRITE(6,9007)TITLE,IPAGE,NPLOT,DATE(QQ)
    WRITE(6,9008)
    WRITE(6,9121)
    WRITE(6,9122)
C......
C...... IF BASIN BOUNDARIES ARE TO BE INCLUDED IN THE PLOT
C...... READ THE SLICE TABLE HEADER LINE
C......
C...... KHOLNE MAY HAVE THE FOLLOWING CODES AND OPTIONS DEPENDING ON
C...... NOLNE FOUND ON CONTROL CARD # 3
C......
    1   "NEITHER" OR " "
    2   "BASIN  "
    3   "MAP   "
    4   "BOTH "
C......
    IF(KHOLNE .EQ. 2 .OR. KHOLNE .EQ. 4) GO TO 110
    GO TO 120
    110 READ(KDISK,9020)LLATS,LLNGS,LHLTS,LHLNS
       9020 FORMAT(2(I6,I7),54X)
    READ(KDTSK,9021)LLATS,LLNGS,LHLTS,LHLNS,DLLTS,DLLNS,DHLTS,DHLNS,
       $LS1,ARGA,LS2,ARGB,LS3,ARGC
C......
C...... DEGREES ARE CONVERTED TO PLOT COORDINATES
C...... NOTE: PROJECTION ROUTINE IS INITIALIZED HERE
C......
    120 CALL LMBINIT(FLLAT,FHLAT,AA,BB,CM)
    CALL LMRCMPT(FLLAT,FLLNG,SCALE,YLLNG,XLLAT)
    CALL LMRCMPT(FHLAT,FHLNG,SCALE,YHLNG,XHLAT)
Initialize the plot band increment and the bottom of the plot band

\[ \text{BNDINC} = 0.01 \]
\[ \text{BNDBTM} = \text{FHLAT - 0.01} \]

Set the width of the strip

\[ 0.9 < \text{WIDTH OF THE STRIP} < 1.1 \text{ INCHES} \]

\[ 200 \text{ BNDBTM} = \text{BNDBTM} - 0.002 \]
\[ \text{BNDINC} = \text{BNDINC} + 0.002 \]
\[ \text{CALL LMBCMPT(BNDBTM,CM,SCALE,YCM,XCM)} \]
\[ \text{DIFF} = \text{XCM} - \text{XHLAT} \]
\[ \text{IF (DIFF.LT.0.0.OR.DIFF.GT.1.1) GO TO 4800} \]
\[ \text{IF (DIFF.LT.0.9) GO TO 200} \]
\[ \text{BNDBTM} = \text{FHLAT} \]

Increment count of plots for this run

\[ \text{EPLOT} = \text{NPLOT} \]
\[ \text{NPLOT} = \text{NPLOT} + 1 \]
\[ \text{BPLOT} = \text{NPLOT} \]
\[ \text{YTEST} = \text{YHLNG} - \text{YLLNG} \]

Test for width of plot greater than width of plotter

\[ \text{IF (YTEST.GT.33.0) GO TO 4700} \]
\[ \text{CALL SYMBOL(0.0,18.4,0.24,BMSG,270.0,10)} \]
\[ \text{CALL NUMBER(0.0,15.8,0.24,BPLOT,270.0,-1)} \]
\[ \text{CALL SYMBOL(1.0,7.5,0.24,TITLE,90.0,78)} \]

Plot the northwest corner and print the coordinates outside the plot

\[ \text{CALL LMBCMPT(FHLAT,FHLNG,SCALE,YNW,XNW)} \]
\[ \text{XNW} = 2.0 \]
\[ \text{YNW} = 17.0 - \text{YHLNG} \]

If option is "NEITHER" skip the border plot routines

If option is "MAP" or "BOTH" plot complete borders

If option is "BASIN" only, plot corners only

\[ \text{IF (KHOLNE.LT.2) GO TO 500} \]
\[ \text{IF(KHOLNE.GT.2) GO TO 300} \]
\[ \text{CALL PLOT(XNW+0.5,YNW,3)} \]
\[ \text{CALL PLOT(XNW,YNW,2)} \]
\[ \text{CALL PLOT(XNW,YNW+0.5,2)} \]
\[ \text{CALL PLOT(XNW+0.5,YNW-0.01,3)} \]
\[ \text{CALL PLOT(XNW-0.01,YNW-0.01,3)} \]
\[ \text{CALL PLOT(XNW-0.01,YNW+0.5,2)} \]

300
\[ \text{NOM} = \text{KHLAT} \]
\[ \text{XL} = \text{XNW} + \text{HT}/2.0 \]
\[ \text{YL} = \text{YNW} - 10.0 \times \text{HT} \]
CALL SYMLDM
NDM = KHLNG
XL = XNW-HT
YL = YNW-4.0*HT
CALL SYMLDM
C...... PLOT THE NORTH EAST CORNER AND PRINT THE
C...... LATITUDE AND LONGITUDE OF THE CORNER
C......
CALL LMBCMPT(FHLAT,FLLNG,SCALE,YNE,XNE)
YNE = 17.0-YLLNG
XNE = 2.0
IF(KHOLNE .GT. 2) GO TO 400 ,
CALL PLOT(XNE,YNE-0.5,3)
CALL PLOT(XNE,YNE,2)
CALL PLOT(XNE+.01,YNE+.01,3)
CALL PLOT(XNE+.01,YNE+.01,2)
CALL PLOT(XNE+0.5,YNE+.01,2)
400 NDM = KHLAT
XL = XNE+HT/2.0
YL = YNE + HT
CALL SYMLDM
NDM = LLNG
XL = XNE-HT
YL = YNE-4.0*HT
CALL SYMLDM
LNCNT = 16
C...... IF OPTION IS "MAP" OR "BOTH" CLOSE TOP OF BORDER
C......
IF (KHOLNE.LE.2) GO TO 500
CALL PLOT(XNE,YNE,3)
CALL PLOT(XNE,YNW,2)
CALL PLOT(XNE+.01,YNW-.01,3)
CALL PLOT(XNE+.01,YNW-.01,2)
C......
C...... COMPUTE THE CENTER POINT OF THE PLOT (Y-AXIS)
C......
500 YCNT = (YNW - YNE) / 2.0 + YNW
C......
C...... COMPUTE THE BOUNDARIES AT THE SIDES OF THE PLOT
C......
550 IF (BNDBTM.LT.FLLAT) GO TO 3800
JSTR = -1
I = 0
BNDBTM = BNDBTM
BNDBTM = BNDBTM-BNDBTM
IF (BNDBTM.LT.FLLAT) BNDBTM = FLLAT-.0001
CALL LMRCMPT(BNDBTM,CM,SCALE,YINC,XINC)
XF = XINC -XHLAT+2.0
IF (TLA.LT.BNDBTM) GO TO 700
TLNP = TLN
JSTR = 0
C..... COMPUTE THE LOCATION OF TIC MARKS AND STORE THEM IN THE PLOT TABLE

C.....
600 IF (TLNP.LT.FLLNG) GO TO 700
   I = I+1
   CALL LMBCMP(TLA,TLNP,SCALE,YLNP,XLNP)
   YS(I) = 17.0-YLNP
   XS(I) = XLNP-XHLAT+2.0
   KSYM(S(I)) = 0
   TLNP = TLNP-FITIC
   IF (TLNP.LT.FLLNG) TLA = TLA-FITIC
   GO TO 600
700 IF (KHQLNE.LE.1) GO TO 900
   IF (KHQLNE.EQ.2) GO TO 800
   I = I+1
   JSTR = 0
C.....
C..... COMPUTE THE LEFT HAND BORDER AND STORE IT IN THE PLOT TABLE
C.....
   CALL LMBCMPT(BNDTOP,FHLNG,SCALE,YCM,XCM)
   XS(I) = XCM-XHLAT+2.0
   YS(I) = 17.0-YCM
   KSYM(S(I)) = 1
   CALL LMBCMPT(BND8TM,FHLNG,SCALE,YCM,XCM)
   XLS(I) = XCM-XHLAT+2.0
   YLS(I) = 17.0-YCM
   I = I+1
C.....
C..... COMPUTE THE RIGHT HAND BORDER AND STORE IT IN THE PLOT TABLE
C.....
   CALL LMBCMPT(BNDTOP,FLLNG,SCALE,YCM,XCM)
   XS(I) = XCM-XHLAT+2.0
   YS(I) = 17.0-YCM
   KSYM(S(I)) = 2
   CALL LMBCMPT(BND8TM,FLLNG,SCALE,YCM,XCM)
   XLS(I) = XCM-XHLAT+2.0
   YLS(I) = 17.0-YCM
   IF(KHQLNE.EQ.3) GO TO 900
C.....
C..... IF OPTION IS "BOTH" OR "BASIN" READ THE SLICE TABLE FROM DISK,
C..... COMPUTE THE COORDINATES OF THE LINE SEGMENT, AND STORE THEM IN THE
C..... PLOT TABLE
C.....
800 IF(EOFSL.EQ.1.0) GO TO 900
   ISLSW = 0
   IF(DLLTS.LT.BNOBTM.AND.SLCNT.GT.0.0) GO TO 900
805 IF(LS1.EQ."") ARGA = -ARGA
   IF(LS2.EQ."") ARGB = -ARGB
   IF(LS3.EQ."") ARGC = -ARGC
9022 FORMAT(1H0,F4.0,2(I6,2X,I7,2X),2(F7.4,2X,F8.4,2X),3(F10.0,2X))
   IF (DLLTS.LT.BNOBTM) GO TO 900
   I = I + 1
   CALL LMBCMP(DLLTS,DLLNS,SCALE,YLNP,XLNP)
   YS(I) = 17.0 - YLNP
   XS(I) = XLNP - XHLAT + 2.0
B-170
KSYM(I) = 3
CALL LMBCMPT(DHLTS,DHLNS,SCALE,YLNPs,XLNPs)
XLS(I) = XLNPs - XHLAT + 2.0
YLS(I) = 17.0 - YLNPs
LSTSLC = LSCR
LSCR = LLATS
IF(NSLC .EQ. 1) NSLC = 2
IF(NSLC .EQ. 2) NSLC = 1
JSTR = 0
9013 FORMAT(1HO,10X,7HNWELLSS=.I5,.I5)
READ(KDISK,9021) LLATS, LLNGS, LHLTS, LHLNS, DLLTS, DLLNS, DHLTS, DHLNS,
$LS1, ARGA, LS2, ARGB, LS3, ARGC
SLCNT = SLCNT + 1.0
9021 FORMAT(2(I6,I7),2(F6.4,1X,F7.4),3(A1,F8.0))
IF(EOF(KDISK)) 802, 803
802 EOFSL = 1.
GO TO 900
803 IF(I .LE. 250) GO TO 800
ISLSW = 1
GO TO 900
C..... IF WELLS ARE TO BE PLOTTED ENTER THE SELECTION ROUTINE
C..... OTHERWISE GO TO THE SORT ROUTINE
C.....
900 IF (NWELLS) 2400, 1000, 1000
C..... IF "GO" SWITCH IS ZERO OR POSITIVE -- GO TO THE TEST FOR A FULL TA
C.....
1000 IF(KGOSW) 1100, 1600, 1600
C.....
C..... READ IN THE FIRST SET OF SITES TO BE PLOTTED
C..... THIS IS THE ONLY READ ROUTINE THAT IS USED FOR
C..... READING THE SITE DATA FILE WHETHER THE FILE IS
C..... COMING FROM THE INPUT STREAM OR FROM A DISK FILE
C.....
1100 READ(NFILE,9003) NSEQ, LAD, LAM, LAS, LOD, LOM, LOS, KDUPL, (LCWLID(JK),
$JK=1,20), DEPTH, ALT, CNSYR, INWU, (LOGS(JK), JK=1,8), TPOPN, MOPN, CSDM
$MFIN, WL, WTLVR, WLCNT, SPCE
9003 FORMAT(I4,3I2, I3,2I2, I2,20A1,F5.0,1X,F5.0,9X,F4.0, A1,8A1,F7.2,A1
$F5.2,A1,F7.2,F4.0,1X, I2, F5.2, 15X)
C.....
C..... AT THE FIRST END OF FILE THE UNIT NUMBER IS CHANGED FROM 5 TO 16
C..... TO ALLOW READING FROM A DISK FILE, THEN THE UNIT NUMBER IS
C..... STEPPED BY 1 FOR EACH END OF FILE ON UNIT 5 "INPUT" ALSO THE
C..... EOFSW IS SET TO 0.0
C.....
1300 READ(NDISK,9003) NSEQ, LAD, LAM, LAS, LOD, LOM, LOS, KDUPL, (LCWLID(JK),
$JK=1,20), DEPTH, ALT, CNSYR, INWU, (LOGS(JK), JK=1,8), TPOPN, MOPN, CSDM
IF(EOF(NDISK)) 4300, 1350
4300 IF(EUF(NFILE)) 4300, 1350
C..... TEST TO CHECK IF THE SITE IS WITHIN THE PLOT LIMITS

1350 FLAT=FLOAT(LAD)+FLOAT(LAM)/60.+FLOAT(LAS)/3600.
FLNG=FLOAT(LOD)+FLOAT(LOM)/60.+FLOAT(LOS)/3600.
KEYID(1)=LAD*10000
KEYID(1)=KEYID(1)+LAM*100
KEYID(1)=KEYID(1)+LAS
KEYID(2)=LOD*10000
KEYID(2)=KEYID(2)+LOM*100
KEYID(2)=KEYID(2)+LOS

1400 KGOSW=+1
IF(FLAT.GT.FHLAT.OR.FLNG.GT.FHNG)GO TO 1100
ISTSW = 0
CALL SLCTN
BTMALT=ALT-DEPTH
WLALT=ALT-WL
TOPALT=ALT-TPUPN
IF(ISTSW.NE.1)GO TO 1100
ISEQ = ISEQ+1

C..... INSERT WRITE STATEMENT HERE TO LIST SELECTED SITES

WRITE(6,9004)ISEQ,NSEQ,LAD,LAM,LAS,LOD,LOM,LOS,KDUP,(LCWLID(JK),$ JK=1,20),DEPTH,ALT,CNSYR,INWU,(LOGS(JK),JK=1,8),TPUPN,MOPN,CSDM $,MFIN,WL,WTLYR,MLCNT,SPCP
9004 FORMAT(1H ,5X,I4,1X,I4,1X,3I2,1X,I3,3I2,2X,20A1,2(1X,F6.0)),1X $,F5.0,1X,A1,2X,8A1,1X,F6.0,1X,A1,F6.2,1X,A1,1X,F8.2,1X,F5.0 $,I4,1X,F6.2)
SEQ=SEQ+1.0

C..... CONVERT MAP COORDINATES TO PLOTTER COORDINATES

1500 CALL LMBCMPT(FLAT,FLNG,SCALE,Y,X)
Y=17.0-Y
X=X-XHLAT+2.0

C..... PLOTTING IS DONE IN ONE INCH WIDE BANDS OR LESS
C..... FIRST TEST IS DONE TO CHECK IF PAST THE ONE INCH MARK
C..... SECOND TEST IS FOR TABLE FULL CONDITION IF THE TABLE IS FULL
C..... THE BAND IS REDUCED IN LENGTH TO ITS PRESENT POSITION AND THE
C..... CURRENT BAND IS PLOTTED
C.....
1600 IF(X.GE.XE)GO TO 2400
IF(I.GE.298)GO TO 2300
1700 I=I+1

C..... NEXT SECTION STORES THE POINTS TO BE PLOTTED IN A TABLE UNTIL
C..... ALL POINTS FOR A GIVEN BAND HAVE BEEN READ IN OR THE TABLE IS
C..... FULL JSTR IS SET TO 0 TO INDICATE THAT A POINT HAS BEEN STORED
C.....
XS(I)=X
YS(T)=Y
KSYM(I)=4
IF(NID.EQ.0)GO TO 1900
CALL SITLBL
1900 JSTR = 0
ISEQ=IFIX(SEQ)
LNCNT=LNCNT+1
IF(LNCNT.GE.60)GO TO 4200
GO TO 1100
C...... ONCE THE STRIP IS COMPLETE, THE TABLE IS SORTED BY
C...... THE LONGITUDE COORDINATE TO REDUCE PEN TRAVEL
C......
2300 XE = XS(I)
BNDRTM = BNDDBM+BNDINC/2.0
2400 J = I+1
JSW = 0
DO 2600 JE=1,J
IF (YS(JE).LE.YS(JE+1)) GO TO 2600
XT = XS(JE)
YT = YS(JE)
TLAB1 = OBL1(JE)
TLAB2 = OBL2(JE)
KSYM1 = KSYM(JE)
XLT = XLS(JE)
YLT = YLS(JE)
XS(JE) = XS(JE+1)
YS(JE) = YS(JE+1)
OBL1(JF) = OBL1(JE+1)
OBL2(JE) = OBL2(JE+1)
KSYM(JE) = KSYM(JE+1)
XLS(JE) = XLS(JE+1)
YLS(JE) = YLS(JE+1)
OBL1(JF+1) = TLAB1
OBL2(JE+1) = TLAB2
KSYM(JF+1) = KSYM1
XLS(JF+1) = XLT
YLS(JF+1) = YLT
JSW = -1
2600 CONTINUE
IF (JSW) 2400,2800,2800
C...... THE FOLLOWING FIVE STATEMENTS MAY BE USED FOR DEBUGGING
C......
C 2650 IF(JSW) 2400,2650,2650
C 2650 DO 2700 J=1,I
C     WRITE(6,9006)I,J,SYM(J),XS(J),YS(J),XLS(J),YLS(J)
C 9006 FORMAT(" ",3I10,4F12.4)
C 2700 CONTINUE
C...... PEN LOCATION IS CHECKED AND PLOTTING IS DONE FROM
C...... THE PEN LOCATION
C......
2800 CALL WHERE(RXPAGER,RPAGEN,RFAC)
   IF (RPAGEN=17.0) 3000,2900,2900

B-173
2900 IJ = I + 1
   IE = 0
   INC = -1
   GO TO 3100
3000 IJ = 0
   IE = I+1
   INC = 1
3100 IJ = IJ + INC
   IF (IJ.EQ.IE) GO TO 550
3150 IF (KSYMS(IJ)) 3100,3300,3400
3200 CALL SYMBOL(XS(IJ),YS(IJ),0.06,3,0.0,-1)
   CALL NUMBER(XS(IJ)+.02+.1,YS(IJ)+.06,0.1,OLBL1(IJ),90.0,-1)
   GO TO 3100
C..... PLOT THE TIC MARKS
3300 CALL SYMBOL(XS(IJ),YS(IJ),0.3,3,0.0,-1)
   GO TO 3100
3400 NGOTO=KSYM(S(IJ)
   GO TO (3500,3550,3600,3650,3650,3650,3650,3650) NGOTO
C..... PLOT DOUBLE THICK LINE ON THE LEFT HAND BORDER SEGMENT
3500 CALL PLOT(XS(IJ),YS(IJ),3)
   CALL PLOT(XLS(IJ),YLS(IJ),2)
   CALL PLOT(XLS(IJ),YLS(IJ)+0.01,3)
   CALL PLOT(XS(IJ),YS(IJ)+0.01,2)
   GO TO 3100
C..... PLOT A DOUBLE THICK LINE ON THE RIGHT HAND BORDER SEGMENT
3550 CALL PLOT(XS(IJ),YS(IJ),3)
   CALL PLOT(XLS(IJ),YLS(IJ),2)
   CALL PLOT(XLS(IJ),YLS(IJ)+0.01,3)
   CALL PLOT(XS(IJ),YS(IJ)+0.01,2)
   GO TO 3100
C..... PLOT THE BASIN BOUNDARY LINE SEGMENTS
3600 IF(NBRDR .EQ. 1) GO TO 3610
   CALL PLOT(XLS(IJ),YLS(IJ),3)
   CALL PLOT(XS(IJ),YS(IJ),2)
   GO TO 3100
3610 CALL NEWPEN(2)
   CALL PLOT(XLS(IJ),YLS(IJ),3)
   CALL PLOT(XS(IJ),YS(IJ),2)
   CALL NEWPEN(1)
   GO TO 3100
C..... PLOT SITE SYMBOL AND LABEL
3650 CALL SYMBOL(XS(IJ),YS(IJ),ZHT,1,90.0,-1)
   IF(NID.EQ.0)GO TO 3100
   IF(NID.EQ.2)GO TO 3675
   CALL NUMBER(XS(IJ)+.12,YS(IJ),SYHT,OLBL1(IJ),90.0,-1)
   GO TO 3100
B-174
3675 IF(NID1.EQ.7) GO TO 3680
   SIHT=SYHT+0.02
   CALL NUMBER(XS(IJ)+0.02, (YS(IJ)+SIHT), SYHT, OLBL1(IJ)
   $,90.0,-1)
   GO TO 3690
3680 SIHT=SYHT+0.02
   CALL NUMBER(XS(IJ)+0.12, (YS(IJ)+SIHT), SYHT, OLBL1(IJ),90.0,2)
3690 CALL PLOT(XS(IJ), YS(IJ)+0.1,3)
   SIHT=SYHT*5.0
   CALL PLOT(XS(IJ), YS(IJ)+SIHT, SYHT, OLBL1(IJ)),90.0,2)
   IF(NID2.EQ.7) GO TO 3695
   SIHT=SYHT+0.02
   CALL NUMBER(XS(IJ)+0.1, (YS(IJ)+SIHT), SYHT, OLBL2(IJ),90.0,-1)
   GO TO 3100
3695 SIHT=SYHT+0.02
   CALL NUMBER(XS(IJ)+0.1, (YS(IJ)+SIHT), SYHT, OLBL2(IJ),90.0,2)
   GO TO 3100

C...... PLOT SW CORNER AND PRINT COORDINATES

3800 CALL LMBCMPT(FLLAT, FHLNG, SCALE, Y, X)
   Y = 17.0 - Y
   X = X-XHLAT+2.0
   IF(KHOLNE .GT. 2) GO TO 3900
   IF(KHOLNE.LT.2) GO TO 3900
   CALL PLOT(X, Y+.5, 3)
   CALL PLOT(X, Y, 2)
   CALL PLOT(X-.5, Y, 2)
   CALL PLOT(X+.01, Y+.5, 3)
   CALL PLOT(X+.01, Y-.01, 2)
   CALL PLOT(X-.5, Y-.01, 2)
3900 TLNP = TLN
   NDM = LLAT
   XL = X+HT/2.0
   YL = Y -10.0*HT
   CALL SYMLDM
   NDM = KHLNG
   XL = X + 2.0*HT
   YL = Y-(4.0*HT)
   CALL SYMLDM

C...... PRINT INFORMATION BLOCK AT SW CORNER

C......
   XL = X+4.0*HT
   YL = Y
   CALL SYMBOL(XL, YL, XHT, LPROJ, 90.0, 40)
   XL = XL+2.0*XHT
   YL = Y
   CALL SYMBOL(XL, YL, XHT, LCMS, 90.0, 20)

C...... PRINT CENTRAL MERIDIAN

C......
   NDM = NCM
   YL = YL+XHT*25.0
   CALL SYMLDM
XL = XL + 2.0 * XHT
YL = Y
CALL SYMBOL(XL, YL, XHT, NSTD, 90.0, 20)
YL = Y + 21.0 * XHT
NDM = LLAT
CALL SYMLDM
YL = Y + 31.0 * XHT
CALL SYMBOL(XL, YL, XHT, "AND", 90.0, 3)
NDM = KHLAT
YL = Y + 37.0 * XHT
CALL SYMLDM

C...... IF OPTION IS "MAP" OR "BOTH" CLOSE THE BOTTOM OF THE BORDER
C...... IF (KHOLNE .LE. 2) GO TO 4000
CALL LMBCMPT(FLLAT, FHLNG, SCALE, Y1, X1)
Y1 = 17.0 - Y1
X1 = X1 - XHLAT + 2.0
CALL PLOT(X1, Y1, 3)
CALL LMBCMPT(FLLAT, FLLNG, SCALE, Y, X)
Y = 17.0 - Y
X = X - XHLAT + 2.0
CALL PLOT(X, Y, 2)

4000 CALL LMBCMPT(FLLAT, FLLNG, SCALE, Y, X)
Y = 17.0 - Y
X = X - XHLAT + 2.0
IF (KHOLNE .GT. 2) GO TO 4150
IF (KHOLNE .LT. 2) GO TO 4150
CALL PLOT(X, Y - 0.5, 3)
CALL PLOT(X, Y, 2)
CALL PLOT(X - 0.5, Y, 2)
CALL PLOT(X + 0.01, Y - 0.5, 3)
CALL PLOT(X + 0.01, Y - 0.01, 2)
CALL PLOT(X - 0.5, Y - 0.01, 2)

4150 IF (KHOLNE .LT. 3) GO TO 4100
CALL PLOT(X, Y, 3)
CALL PLOT(X + 0.01, Y - 0.01, 2)

4100 NDM = LLAT
XL = X + HT / 2.0
YL = Y + HT
CALL SYMLDM
NDM = LLNG
XL = X + 2.0 * HT
YL = Y - (4.0 * HT)
CALL SYMLDM

C...... CALL ROUTINE TO OUTPUT MILEAGE SCALE
C......
XLM = XL + 1.0
YLM = 17.0
CALL SCMLE(XLM, YLM, SCALE)
HT = 0.14
X = XL + 2.0
Y = 17.0 - 6.0 * HT
XCNTR = XL + 3.5

B-176
YCNTR = 17.0
CALL SYMBOL(XCNTR+2.0,7.5,0.24,TITLE,90.0,78)
CALL SYMBOL(XCNTR+3.0,18.4,0.24," END PLOT",270.0,10)
CALL NUMBER(999.0,15.8,0.24,8PLOT,270.0,-1)
CALL WHERE(XCORD,YCORD,RFAC)
XNEW = XCORD+4.0
CALL PLOT(XNEW,0.0,0,-3)
GO TO 100

4200 IPAGE = IPAGE+1
WRITE(6,9007)TITLE,IPAGE,NPLOT,DATE(QQ)
9007 FORMAT(1H1,10X,4HUSGS,14X,7A10,A8,19X,4HPAGE,I3
1/1H0,10X,4HPLOT,I2,37X,24HSWABRASA WELL SITE PLOTS,44X,A10)
WRITE(6,9008)
9008 FORMAT(1HO,5X,3HNEW,2X,3HRPT,72X,6HTOP OF,4H TYP,9H CSNG TYP,3X
1,12HWATER LEVELS)
WRITE(6,9121)
9121 FORMAT(1H0,5X,3HSEQ,2X,3HSEQ,4X,3HLAT,4X,4HLONG,8X,8HLOCAL ID,7X
$,5HDEPTH,4X,3HALT,2X,5HCNSYR,1X,3HUSE,3X,4HLOGS,2X,7HOPENING,1X
$,3HOPN,1X,4HDIAM,1X,3HFIN,1X,4HMEAS,1X,14HYEAR CNT SPCAP)
LCNT = 6
WRITE(6,9122)
9122 FORMAT(1H ,5H )
GO TO 1100

4300 IF(KGOSW.GE.0)NWELLS=-1
IF(KGOSW)4400,2400,2400
4400 GO TO 100
4500 IF (NPLOT.GT.0) GO TO 100
WRITE(6,9009)
9009 FORMAT("0","EITHER THE DATA FILE WAS EMPTY OR SITES OUT OF RANGE")
STOP
4600 STOP
4700 WRITE(6,9010)YTEST,YHLNG,YLLNG
9010 FORMAT(" ","JOB ABORTED---TOO LARGE",3F12.4)
STOP
4800 WRITE(6,9011)DIFF,XCM,XHLAT
9011 FORMAT(" ","STRIP CONTROL OUT OF BOUND",3F12.4)
STOP
4900 WRITE(6,9012)KEYID,KEYID
9012 FORMAT(1H ,I6,I7,I2,5X,I6,21H FILE OUT OF SEQUENCE,85X)
STOP
END
SUBROUTINE NREAD3
COMMON/MALINE/JSTR,TITLE(8),XS(300),YS(300)
$,KS¥MS(300),XLS(300),YLS(300),FHLAT,FLLAT,FHLNG,FLLNG,CM,ZHT
$,SYHT,IPAGE,NPL0T,SCALE,XHLAT,YHLNG,XLLAT,YLLNG,TITLE2(8)
$,TITEL3(8),TLA,TLN,FITIC,LLAT,KHLAG,LLNG,KHOLME,LOP,NCM
$,TITLE4(8),TITLE5(8),TITLE6(8),TITLE7(8),NF,NOTN(20)
$,NDS,NBRDR
C......
COMMON/LBLR/I,NID,NID1,NID2,OLBL1(300),OLBL2(300),BTMALT,WLALT
$,TOPALT
C......
COMMON/SELCI/CNYR1,CNYR2,CDIAM,ILOG(8),IFIN(3),IOPN(3),IUSE(3)
$,ISTSW,NFIN,NLOG,NUPN,NWVL,NWUS,NEWL,SPEC,TSTDEP,TSTOPN,TSTWL
$,TSTWDP,TSTDIM,WTX1,WTY2,WTREDP,WTRLVL,XDEPTH,OPNTOP
C......
DATA ZHT,SYHT/0.06,0.06/
DATA (NOTN(I),I=1,13)/1OH SEQUENCE,10H WELL DEPTH,10H ALTITUDE,
$10H YR OF CNST,10H WATER LVL,10H WT LVL YR,10H SPEC CPTY,
$10H TOP DEPTH,10H CANAG DIAM,10H NUM WLVLs,10H ALT BOTTOM,
$10H ALT WTRLVL,10H ALT TP OPN/
C......
CONTROL CARDS ARE AS FOLLOWS
C......
TITLE I2,7410,AB
C......
LATITUDE(LOW AND HIGH) LONGITUDES (LOW AND HIGH)
C......
PROJECTION AND SCALE DATA
C......
IPAGE = IPAGE + 1
C......
READ TITLE RECORD AND PRINT FIRST HEADING LINE
C......
READ(5,9010)IDC1,TITLE
IF (EOF(5)) 9901,100
100 IF (IDC1.NE.1) GO TO 9801
NPXX = NPL0T +1
WRITE(6,9020)TITLE,IPAGE,NPXX,DATE(QQ)
C......
READ AND PRINT PLOT COORDINATES LIMITS
C......
READ(5,9030)IDC2,LLAD,LLAM,LLAS,LLND,LLNM,LLNS,KHLAD,KHLM,KHLAS
$,KHLND,KHLNM,KHLNS,KCMD,KCMM,KSMS,KHPROJ,KSC,NEWL,NBRDR,
$NHT,NHT,ISYM
IF (EOF(5)) 9901,200
200 IF (IDC2.NE.2) GO TO 9802
IF (LLAD.LT.30.0R.LLAD.GT.37) GO TO 9821
IF (KHLAG.LT.30.0R.KHLAG.GT.37) GO TO 9822
IF (LLND.LT.107.0R.LLND.GT.116)GO TO 9823
IF (KHLND.LT.107.0R.KHLND.GT.116) GO TO 9824
IF (LLND.GT.KHLND) GO TO 9825
IF (LLAD.GT.KHLAG) GO TO 9826
C......
CONVERT NOTATION HEIGHT TO DECIMAL INCHES
IF(NHT .LE. 0) GO TO 210
ZHT = NHT
ZHT = ZHT * 0.01
B-178
210 IF(NSHT LE. 0) GO TO 220  
    SYHT = NSHT  
    SYHT = SYHT * 0.01  
C..... CONVERT DEGREES, MINUTES, SECONDS TO DECIMAL LONGITUDES AND  
C..... LATITUDES  
C.....  
220 LLAT=LLAD*10000+LLAM*100+LLAS  
      KHLAT=KHLAD*10000+KHLAM*100+KHLAS  
      LLNG=LLND*10000+LLNM*100+LLNS  
      KHLNG=KHLND*10000+KHLNM*100+KHLNS  
C..... CONVERT LOW LATITUDE TO DECIMAL DEGREES  
C.....  
      FLLAD=LLAD  
      FLLAM=LLAM  
      FLLAS=LLAS  
      FLLAT=FLLAD+FLLAM/60.+FLLAS/3600.  
C..... CONVERT HIGH LATITUDE TO DECIMAL DEGREES  
C.....  
      FHLAD=KHLAD  
      FHLAM=KHLAM  
      FHLAS=KHLAS  
      FHLAT=FHLAD+FHLAM/60.+FHLAS/3600.  
C..... CONVERT LOW LONGITUDE TO DECIMAL DEGREES  
C.....  
      FLLND=LLND  
      FLLNM=LLNM  
      FLLNS=LLNS  
      FLLNG=FLLND+FLLNM/60.+FLLNS/3600.  
C..... CONVERT HIGH LONGITUDE TO DECIMAL DEGREES  
C.....  
      FHLNO=KHLNO  
      FHLNM=KHLNM  
      FHLNS=KHLNS  
      FHLNG=FHLNO+FHLNM/60.+FHLNS/3600.  
C..... PROJECTION AND SCALE DATA  
C.....  
C..... CONVERT INPUT CENTRAL MERIDIAN TO DECIMAL DEGREES  
C.....  
    IF (KCMD.EQ.0) GO TO 300  
    CMD = KCMD  
    CMM = KCMM  
    CMM = CMM/60.  
    CMS = KCMS  
    CMS = CMS/3600.0  
    CM = CMD+CMM+CMS  
    GO TO 310
C..... IF NO CENTRAL MERIDIAN IS INPUT, COMPUTE CENTRAL MERIDIAN
C.....
300 CM = (FLNGL+FHNLG)/2.0
KCMD = CM
CMD = KCMD
CMT = CM-CMD
CMM = CMT*60.0
KCMM = CMM
CMT = KCMM
CMS = (CMM-CMT)*60.0
KCMS = CMS
GO TO 310

C..... IF SCALE IS INPUT, CONVERT TO DECIMAL
C.....
310 IF (KSC.GT.0) SCALE = KSC
NCM = KCMD*10000+KCMM*100+KCMS

C..... DETERMINE WHETHER PROJECTION IS LAMBERT CONFORMAL OR MERCATOR
C.....
IF (KHPROJ.EQ.1) GO TO 320
IF (KHPROJ.EQ.2) GO TO 330
320 PROJ1 = "LAMBERT CO"
PROJ2 = "CONFORMAL"
GO TO 400
330 PROJ1 = "MERCATOR"
PROJ2 = ""
GO TO 400

C..... WRITE PLOT LIMITS AND PARAMETERS
C.....
400 WRITE(6,9040)
WRITE(6,9050)PROJ1,PROJ2
WRITE(6,9060)KS
WRITE(6,9070)LLAD,LLAM,LLAS,KHLAD,KHLAM,KHLAS
WRITE(6,9080)LLND,LLNM,LLNS,KHLND,KHLNM,KHLNS
WRITE(6,9090)KHLAD,KHLAM,KHLAS,KHLND,KHLNM,KHLNS,LLAD,LLAM,LLAS,
$KHLAD,KHLAM,KHLAS,KHLND,KHLNM,KHLNS,LLAD,LLAM,LLAS,
$LLND,LLNM,LLNS
WRITE(6,9100)KCMD,KCMM,KCMS
WRITE(6,9130)SYHT
WRITE(6,9140)ZHT
IF(NWELLS.GE.0) WRITE(6,9150)NWELLS
IF(NWELLS.LT.0) WRITE(6,9160)NWELLS
IF(N8RDR.NE.1) WRITE(6,9170)
IF(N8RDR.EQ.1) WRITE(6,9180)

C..... READ TIC MARKS AND PERIMETER OPTIONS
C.....
READ(5,9190)IDC3,ATIC1,ATTC1,ITIC,ATIC2,ATIC3,ATIC4,INTIC
$,ATIC5,ATIC6,NOLNE
IF (EOF(5)) 9803,500
500 IF (IDC3.NE.3) GO TO 9803
C...... CONVERT TIC MARK INTERVAL TO DECIMAL
C......
FITIC=ITIC
FITIC=FITIC/60.
FINTIC=INTIC
FINTIC=FINTIC/60.
C...... DETERMINE THE TYPE OF MAP OR BASIN PLOT OPTION TO BE USED
C......
IF (NOLNE.EQ."NEITHER") KHOLNE = 1
IF (NOLNE.EQ."BASIN ") KHOLNE = 2
IF (NOLNE.EQ."MAP ") KHOLNE = 3
IF (NOLNE.EQ."BOTH ") KHOLNE = 4
IF (NOLNE.EQ." ") KHOLNE = 1
WRITE(6,9200)ATIC1,ATIC2,ATIC3,ATIC4,ATIC5,ATIC6,NOLNE
MHLAM =KHLAM
IF (INTIC.EQ.0) GO TO 520
DO 510 IXK=1,60
IIN=MHLAM/INTIC
IF (MHLAM-(IIN*INTIC).EQ.0) GO TO 520
510 MHLAM = MHLAM -
520 TLD =KHLAD
TLM =MHLAM
TLA = TLD + TLM/60.
MHLNM = KHLNM
DO 530 IXK=1,60
IF (INTIC.EQ.0) GO TO 540
IIN = MHLNM/INTIC
IF (MHLNM-(IIN*INTIC).EQ.0) GO TO 540
530 MHLNM = MHLNM -
540 TLND = KHLND
TLNM = MHLNM
TLN = TLND + TLNM/60.
C...... READ OPTION CONTROL CARD # 4
C......
READ(5,9210)IDC4,NID,NID1,NID2,CNYR1,CNYR2,WLYR1,WLYR2,
$XDEPTH,WTRLVL,OPNTOP,NOPN,(IOPN(I),I=1,3),NLUG,(ILOG(N),N=1,8)
$,NWUS,(IWUSE(N),N=1,3),NFIN,(IFIN(N),N=1,3),SCAP,NWLV,WTRDEP
$,CDIM
IF(EQF(5))9800,600
600 IF(IDC4 .NE. 4) GO TO 9804
TSTDEP = ABS(XDEPTH)
TSTOPN = ABS(OPNTOP)
TSTWL = ABS(WTRLVL)
TSTWDP=ABS(WTRDEP)
TSTDIM=ABS(CDIM)
WRITE(6,9020)TITLE,IPAGE,NPXX,DATE(QQ)
WRITE(6,9220)
IF(NWELLS)610,620,620
610 WRITE(6,9230)
GO TO 1700
C...... EXAMINE SITE IDENTIFIER OPTIONS
C...... 620 IF(NIO .EQ. 0) WRITE(6,9240)
   IF(NID .NE. 1) GO TO 630
   IF(NID1 .LE. 0 .OR. NID1 .GT. 13) GO TO 9830
      WRITE(6,9250) NOTN(NID1)
C...... 630 IF(NID .LT. 2) GO TO 635
   IF(NID .NE. 2) GO TO 9831
   IF(NID1 .LE. 0 .OR. NID1 .GT. 13) GO TO 9832
      WRITE(6,9260) NOTN(NID1), NOTN(NID2)
C...... EXAMINE TIME SPAN RANGE - CONSTRUCTION YEAR
C...... 635 IF(CNYR1 .LE. 0.0 .AND. CNYR2 .LE. 0.0) WRITE(6,9270)
   IF(CNYR1 .LE. 0.0 .AND. CNYR2 .GT. 0.0) WRITE(6,9280) CNYR2
   IF(CNYR1 .GT. 0.0 .AND. CNYR2 .LE. 0.0) WRITE(6,9290) CNYR1
   IF(CNYR1 .GT. 0.0 .AND. CNYR2 .GT. 0.0 .AND. CNYR1 .GT. CNYR2) GO TO 9833
      WRITE(6,9300) CNYR1, CNYR2
C...... EXAMINE TIME SPAN RANGE - YEAR OF WATER LEVEL MEASUREMENT
C...... IF(WLYR1 .LE. 0.0 .AND. WLYR2 .LE. 0.0) WRITE(6,9310)
   IF(WLYR1 .LE. 0.0 .AND. WLYR2 .GT. 0.0) WRITE(6,9320) WLYR2
   IF(WLYR1 .GT. 0.0 .AND. WLYR2 .LE. 0.0) WRITE(6,9330) WLYR1
   IF(WLYR1 .GT. 0.0 .AND. WLYR2 .GT. 0.0 .AND. WLYR1 .GT. WLYR2) GO TO 9834
      WRITE(6,9340) WLYR1, WLYR2
C...... EXAMINE DEPTH OF WELL
C...... IF(XDEP) 640,650,660
   WRITE(6,9350) TSTDEP
   GO TO 670
C...... 650 WRITE(6,9360)
   GO TO 670
   WRITE(6,9370) TSTDEP
C...... EXAMINE WATER LEVEL MEASUREMENTS
C...... 670 IF(WTRLVL) 680,690,700
   WRITE(6,9380) TSTWL
   GO TO 710
C...... 690 WRITE(6,9390)
   GO TO 710
   WRITE(6,9400) TSTWL
C...... EXAMINE DEPTH TO TOP OF THE OPENING
C...... 710 IF(OPNTO) 720,730,740
   WRITE(6,9410) TSTOPN
   GO TO 750
C...... 730 WRITE(6,9420)
   GO TO 750
   WRITE(6,9430) TSTOPN
B-182
C... EXAMINE TYPE OF OPENING PARAMETERS
C...
745 IF(NOPN .GT. 3) GO TO 9840
   IF(NOPN .GT. 0) GO TO 747
   WRITE(6,9445)
   GO TO 750
747 WRITE(6,9447) NOPN, (IOPN(I), I=1, NOPN)
C...
C... EXAMINE LOG PARAMETERS
C...
750 IF(NLOG .GT. 8) GO TO 9835
   IF(NLOG .GT. 0) GO TO 760
   WRITE(6,9440)
   GO TO 800
760 WRITE(6,9450) NLOG, (ILOG(I), I=1, NLOG)
C...
C... EXAMINE WATER USE PARAMETERS
C...
800 IF(NWUS .GT. 3) GO TO 9836
   IF(NWUS .GT. 0) GO TO 810
   WRITE(6,9460)
   GO TO 820
810 WRITE(6,9470) NWUS, (IWUSE(I), I=1, NWUS)
C...
C... EXAMINE FINISH PARAMETERS
C...
820 IF(MFIN .GT. 3) GO TO 9837
   IF(NFIN .GT. 0) GO TO 830
   WRITE(6,9480)
   GO TO 840
830 WRITE(6,9490) NFIN, (IFIN(I), I=1, NFIN)
C...
C... EXAMINE SPECIFIC CAPACITY PARAMETER
C...
840 IF(SPCAP .GT. 9999.99) GO TO 9838
   IF(SPCAP .GT. 0.0) GO TO 850
   IF(SPCAP .EQ. 0.0) GO TO 860
   IF(SPCAP .LT. 0.0) GO TO 870
850 WRITE(6,9500) SPCAP
   GO TO 880
860 WRITE(6,9510)
   GO TO 880
870 WRITE(6,9520) SPCAP
C...
C... EXAMINE WATER LEVEL COUNT PARAMETER
C...
880 IF(NWLVL .GT. 999) GO TO 9839
   IF(NWLVL .GT. 0) GO TO 890
   WRITE(6,9530)
   GO TO 900
890 WRITE(6,9540) NWLVL
900 IF(WTRDEP.EQ.0.0) GO TO 930
   IF(WTRDEP.GT.0.0) GO TO 920
WRITE(6,9941)TSTWDP
GO TO 940
920 WRITE(6,9942)TSTWDP
GO TO 940
930 WRITE(6,9943)
940 IF(CDIAM.EQ.0.0) GO TO 960
IF(CDIAM.GT.0.0) GO TO 950
WRITE(6,9944)TSTDIM
GO TO 970
950 WRITE(6,9945)TSTDIM
GO TO 970
960 WRITE(6,9946)
970 CONTINUE
LNCNT = 60
1700 RETURN
C......
C...... ERROR STOPS
C......
9800 WRITE(6,9899)
9899 FORMAT(1H1,"***** EOF WHILE READING OPTIONS AND CONTROLS ****")
STOP
9801 WRITE(6,9811)IDC1
9811 FORMAT(1H1,31HFIRST CONTROL RECORD NOT 01 ID=,I2)
STOP
9802 WRITE(6,9812)IDC2
9812 FORMAT(1H1,31HSECOND CONTROL RECORD NOT 02 ID=,I2)
STOP
9803 WRITE(6,9813)IDC3
9813 FORMAT(1H1,31HTHIRD CONTROL RECORD NOT 03 ID=,I2)
STOP
9804 WRITE(6,9814)IDC4
9814 FORMAT(1H1,31HFORTH CONTROL RECORD NOT 04 ID=,I2)
STOP
9821 WRITE(6,9921)LLAD
STOP
9822 WRITE(6,9922)KHLAD
STOP
9823 WRITE(6,9923)LLND
STOP
9824 WRITE(6,9924)KHLND
STOP
9825 WRITE(6,9925)LLND,KHLND
STOP
9826 WRITE(6,9926)LLAD,KHLAD
STOP
9830 WRITE(6,9930)NID1
STOP
9831 WRITE(6,9931)NID
STOP
9832 WRITE(6,9932)NID2
STOP
9833 WRITE(6,9933)
STOP


```
9834 WRITE(6,9934)  
STOP  
9835 WRITE(6,9935)NLOG  
STOP  
9836 WRITE(6,9936)NWUS  
STOP  
9837 WRITE(6,9937)NFIN  
STOP  
9838 WRITE(6,9938)SPCAP  
STOP  
9839 WRITE(6,9939)NWVL  
STOP  
9840 WRITE(6,9940)NQPN  
STOP  
9901 WRITE(6,9911)  
9911 FORMAT(1H1,35HINPUT DATA FILE EMPTY* END OF RUN *)  
   IF (NPLOT.EQ.0) WRITE(6,9912)  
   IF (NPLOT.NE.0) WRITE(6,9913)NPLOT  
9912 FORMAT(1HO,39HNO PLOT OUTPUT ***CONTROLS MISSING *** )  
9913 FORMAT(1HO,I2,34HPLOTS COMPLETED NORMAL TERMINATION)  
   CALL WHERE(XFIN,YFIN,RFIN)  
   CALL PLOT(1.0,YFIN+5.0,999)  
   STOP  
C......  
C...... FORMAT STATEMENTS  
C......  
9010 FORMAT(I2,7A10,A8)  
9020 FORMAT(1H1,10X,4HUSGS,14X,7A10,A8,19X,4HPAGE,I3  
   1,1/H0,10X,4HPLOT,I2,37X,25HSWAB-RASA WELL SITE PLOTS,44X,A10)  
9030 FORMAT(I2,2(3I2,I3,2I2),I3,2I2,I2,I7,5I2)  
9040 FORMAT(1HO,51X,27HPLOT OPTIONS AND PARAMETERS,44X)  
9050 FORMAT(1HO,51X,14HPROJECTION IS ,2A10)  
9060 FORMAT(1HO,51X,14HSSCALE : 1 : ,I7)  
9070 FORMAT(1HO,30X,19HLATITUDES : LOW :,I3,1X,2(1X,I2),10X,  
\$7HIGH :,I3,2(1X,I2))  
9080 FORMAT(1H,30X,19HLONGITUDES : LOW :,I3,1X,2(1X,I2),10X,  
\$7HIGH :,I3,2(1X,I2))  
9090 FORMAT(1HO,30X,21HPLLOT LIMITS : N.W. :I3,2(1X,I2),3H :,  
\$13,2(1X,I2),8X,7HN.E. :I3,2(1X,I2),3H :,I3,2(1X,I2))  
9100 FORMAT(1H,45X,6HS.W. :I3,2(1X,I2),3H :,  
\$13,2(1X,I2),8X,7HS.E. :I3,2(1X,I2),3H :,I3,2(1X,I2))  
9110 FORMAT(1H,40X,20HCENTRAL MERIDIAN : ,I3,2(1X,I2))  
9120 FORMAT(1H,40X,I2,38HSYMBOLS WILL BE USED AS SITE LOCATORS)  
9130 FORMAT(1H,40X,21HSITE MARKERS WILL BE ,F4.2,17H INCHES IN HEIGHT)  
9140 FORMAT(1H,40X,25HSITE IDENTIFIERS WILL BE ,F4.2,17H INCHES IN WEI  
\$GH)  
9150 FORMAT(1H,40X,26HWELL SITES WILL BE PLOTTED,5X,9HNWELLS = ,I3)  
9160 FORMAT(1H,40X,29HNWELL SITES WILL BE PLOTTED,5X,9HNWELLS = ,I3)  
9170 FORMAT(1H,40X,35HSBASIN BOUNDARIES PLOTTED WITH PEN 1)  
9180 FORMAT(1H,40X,35HSBASIN BOUNDARIES PLOTTED WITH PEN 2)  
9190 FORMAT(I2,A10,A3,I2,A10,A8,I2,A10,1X,A10,1X,A7,14X)  
9200 FORMAT(1H,40X,A10,A3,I2,A10,A8,I2,A10,1X,A10,1X,A7,30X)  
9210 FORMAT(I2,12I2,4F4.0,3F5.0,I1,3A1,I1,8A1,2(I1,3A1),F5.2,I3  
\$,F5.0,F6.3)  

B-185
```
9220 FORMAT(1H,52X,26H OPTIONS FOR SITE SELECTION)
9230 FORMAT(1H,30X,43H NWELLS = -1 : ALL SELECTION OPTIONS IGNORED)
9240 FORMAT(1H,30X,24H SITE NOTATION IS OMITTED)
9250 FORMAT(1H,30X,21H SITE IDENTIFIER IS : ,A10)
9260 FORMAT(1H,30X,21H SITE IDENTIFIER IS : ,A10,6H OVER ,A10)
9270 FORMAT(1H,30X,45H YEAR OF CONSTRUCTION IS NOT USED IN SELECTION, $17H - NO RANGE GIVEN)
9280 FORMAT(1H,30X,44H SELECTED SITES WILL HAVE CONSTRUCTION YEARS, $5THRU ,F5.0)
9290 FORMAT(1H,30X,44H SELECTED SITES WILL HAVE CONSTRUCTION YEARS, $5THRU ,F5.0,11H TO PRESENT)
9300 FORMAT(1H,30X,44H SELECTED SITES WILL HAVE CONSTRUCTION YEARS, $5BETWEEN ,F5.0,4H AND,F5.0)
9310 FORMAT(1H,30X,41H SITE IDENTIFIER IS: ,A10,6H OVER ,A10)
9320 FORMAT(1H,30X,43H SELECTED SITES WILL HAVE WATER LEVEL YEARS, $5THRU ,F5.0)
9330 FORMAT(1H,30X,43H SELECTED SITES WILL HAVE WATER LEVEL YEARS, $5THRU ,F5.0,11H TO PRESENT)
9340 FORMAT(1H,30X,43H SELECTED SITES WILL HAVE WATER LEVEL YEARS, $5BETWEEN ,F5.0,4H AND,F5.0)
9345 FORMAT(1H,30X,20H PHYSICAL PARAMETERS)
9350 FORMAT(1H,30X,25H DEPTH OF WELL : DEPTHS OF,F6.0,8H OR LESS)
9360 FORMAT(1H,30X,37H DEPTH OF WELL : NOT USED IN SELECTION)
9370 FORMAT(1H,30X,25H DEPTH OF WELL : DEPTHS OF,F6.0,11H OR GREATER)
9380 FORMAT(1H,30X,30H WATER LEVELS : MEASUREMENTS OF,F6.0,8H OR LESS)
9390 FORMAT(1H,30X,30H WATER LEVELS : NOT USED IN SELECTION)
9400 FORMAT(1H,30X,30H WATER LEVELS : MEASUREMENTS OF,F6.0,11H OR GREATER)
9410 FORMAT(1H,30X,32H DEPTH TO TOP OPENING : DEPTHS OF,F6.0,8H OR LESS)
9420 FORMAT(1H,30X,47H DEPTH TO TOP OPENING : NOT USED IN SELECTION)
9430 FORMAT(1H,30X,32H DEPTH TO TOP OPENING : DEPTHS OF,F6.0,11H OR GREATER)
9440 FORMAT(1H,30X,33H LOG TYPE IS NOT USED IN SELECTION)
9445 FORMAT(1H,30X,40H TYPE OF OPENING IS NOT USED IN SELECTION)
9447 FORMAT(1H,30X,34H OPENING TYPES USED FOR SELECTION : ,3(1X,A1))
9450 FORMAT(1H,30X,31H LOG TYPES USED FOR SELECTION : ,8(1X,A1))
9460 FORMAT(1H,30X,34H WATER USE IS NOT USED IN SELECTION)
9470 FORMAT(1H,30X,32H WATER USES USED FOR SELECTION : ,3(1X,A1))
9480 FORMAT(1H,30X,36H FINISH TYPE IS NOT USED IN SELECTION)
9490 FORMAT(1H,30X,34H FINISH TYPES USED FOR SELECTION : ,3(1X,A1))
9500 FORMAT(1H,30X,29H SPECIFIC CAPACITY : SP CAP OF,F7.2,8H OR LESS)
9510 FORMAT(1H,30X,11H OR GREATER)
9510 FORMAT(1H,30X,42H SPECIFIC CAPACITY : NOT USED IN SELECTION)
9520 FORMAT(1H,30X,29H SPECIFIC CAPACITY : SP CAP OF,F7.2,8H OR LESS)
9530 FORMAT(1H,30X,42H WATER LEVEL COUNT : NOT USED IN SELECTION)
9540 FORMAT(1H,30X,29H WATER LEVEL COUNT : COUNTS OF,F5.8H OR LESS)
9921 FORMAT(1H,10X,30H LOW LATITUDE OUTSIDE STATE RANGE:30-37, $5X,9HLow LAT = ,I3)
9922 FORMAT(1H,10X,39H HIGH LATITUDE OUTSIDE STATE RANGE:30-37, $5X,10H HIGH LAT = ,I3)
SUBROUTINE LAMMB(ARG1, ARG2, ARG3, ARG4, ARG5)

C..... LAMBERT CONFORMAL PROJECTION, AFTER SCHWENN AND WOLFE (U. WISC.,
C..... CALL LMBINIT(A,B,C,D,E) -------INITIALIZES OR DEFINES PROJECTION
C..... A=STANDARD PARALLEL NO. 1 (+ = NORTH) LG. +33.
C..... B=STANDARD PARALLEL NO. 2 LG. +45.
C..... C=LENGTH OF SEMIMAJOR AXIS OF ELLIPSOID (LG. 6378206.4
C..... D=LENGTH OF SEMIMINOR AXIS OF ELLIPSOID (LG. 6356583.8
C..... E=STANDARD LONGITUDE IN DEGREES (+ FOR WEST).
C..... CALL LMBRST(E,DUM,DUM,DUM,DUM) -------SETS STANDARD LONGITUDE
C..... E=STANDARD LONGITUDE IN DEGREES.
C..... CALL LMBCMPT(P,Q,S,X,Y) ------CONVERTS LAT/LONG TO X/Y.
C..... P=LATITUDE IN DEGREES (+ FOR NORTH).
C..... Q=LONGITUDE IN DEGREES (+ FOR WEST).
C..... S=MAP SCALE (LG. 500000).
C..... X=OUTPUT X COORDINATE (INCREASING TO W OF STND LONGITUDE)
C..... Y=OUTPUT Y COORDINATE (INCREASING TO S OF APEX 0
C..... IF OUTPUT IS DESIRED IN INCHES (FOR MAP USE), C AND D MUST BE INCHES

END
ENTRY LMBINIT

ENTRY LMBINIT
STD = ARG5
AL11 = ARG1*.0174533
AL22 = ARG2*.0174533
EPS = SQRT(1.-ARG4**2/ARG3**2)
CN1 = EPS**2
AN1 = ARG3/SQRT(1.-CN1*SIN(AL11)**2)
AN2 = ARG3/SQRT(1.-CN1*SIN(AL22)**2)
CN2 = SIN(1./3600.*.0174533)
AA1 = 1./(AN1*CN2)
AA2 = 1./(AN2*CN2)
CN5 = EPS/2.
COSL1 = COS(AL11)
P1 = 90.-ARG1
P2 = 90.-ARG2
CN3 = COS(P1*.0174533)
CN4 = COS(P2*.0174533)
Q1 = TAN(P1*.0174533/2.)*((1.+EPS*CN3)/(1.-EPS*CN3))**CN5
Q2 = TAN(P2*.0174533/2.)*((1.+EPS*CN4)/(1.-EPS*CN4))**CN5
ALCN = ALOG10(COSL1/COS(AL22)/AA1*AA2)/ALOG10(Q1/Q2)
AKCN = COSL1/(AA1*CN2*ALCN*Q1**ALCN)
RCN = ALCN*EPS/2.
RETURN

ENTRY LMBRST

ENTRY LMBRST
STD = ARG1
RETURN

ENTRY LMBCMPT

ENTRY LMBCMPT
AMCN = STD-ARG2
P = (90.-ARG1)*.0174533
CN7 = EPS*COS(P)
BASE1 = AKCN*TAN(P/2.)*ALCN
BASE2 = ((1.+CN7)/(1.-CN7))*RCN
BASE3 = ALCN*AMCN*.0174533
BASE4 = BASE1*BASE2
ARG4 = -BASE4*STN(BASE3)/ARG3
ARG5 = BASE4*COS(BASE3)/ARG3
RETURN
END
SUBROUTINE SLCTN

C......
COMMON/SELT/CNYR1,CNYR2,CDIAM,ILOG(8),IFIN(3),IUPN(3),IWUSE(3)
$ ,ISTSW,NFIN,NLOG,NUPN,NWLV,NWUS,NWELLS,SPCAP,TSTDEP,TSTUPN,TSTWL
$ ,TSTWDP,TSTDIM,WLYR1,WLYR2,WTRLVL,XDEPTH,OPNGP

C......
COMMON/RUR/ALT,CSDM,CNSYR,DEPTH,INWU,LOGS(8),MFIN,MOPN,SPCP
$ ,TPOPN,WL,WTLVR,WLCTN

C......
TEST CONSTRUCTION YEAR WITHIN SPECIFIED RANGE

C......
IF((CNYR1.EQ.0).AND.(CNYR2.EQ.0))GO TO 200
IF((CNYR1.EQ.0).AND.(CNYR2.GT.0))GO TO 110
IF((CNYR1.GT.0).AND.(CNYR2.EQ.0))GO TO 150
IF((CNYR1.GT.0).AND.(CNYR2.GT.0))GO TO 170
GO TO 9900
110 IF(CNSYR.GT.CNYR2)GO TO 9900
ISTSW=1
GO TO 200
150 IF(CNSYR.LT.CNYR1)GO TO 9900
ISTSW=1
GO TO 200
170 IF(CNSYR.LT.CNYR2)GO TO 9900
ISTSW=1

C......
TEST WATER LEVEL MEASUREMENT YEAR WITHIN SPECIFIED RANGE

C......
200 IF((WLYR1.EQ.0).AND.(WLYR2.EQ.0))GO TO 300
IF((WLYR1.EQ.0).AND.(WLYR2.GT.0))GO TO 210
IF((WLYR1.GT.0).AND.(WLYR2.EQ.0))GO TO 250
IF((WLYR1.GT.0).AND.(WLYR2.GT.0))GO TO 270
GO TO 9900
210 IF(WTLYR.GT.WLYR2)GO TO 9900
ISTSW=1
GO TO 300
250 IF(WTLYR.LT.WLYR1)GO TO 9900
ISTSW=1
GO TO 300
270 IF(WTLYR.LT.WLYR2)GO TO 9900
ISTSW=1
GO TO 300

C......
TEST FOR DEPTH OF WELL WITHIN SPECIFIED RANGE

C......
300 IF(XDEPTH)310,400,350
310 IF(DEPTH.GT.TSTDEP)GO TO 9900
ISTSW=1
GO TO 400
350 IF(DEPTH.LT.TSTDEP)GO TO 9900
ISTSW=1
GO TO 400
C..... TEST DEPTH TO TOP OF OPENING FOR SPECIFIED RANGE
  400 IF(OPNTOP)410,500,450
  410 IF(TPOPN.GT.TSTOPN)GO TO 9900
    ISTSW=1
    GO TO 500
  450 IF(TPOPN.LT.TSTOPN)GO TO 9900
    ISTSW=1
    GO TO 500
C..... TEST DEPTH TO WATER FOR SPECIFIED RANGE
C.....
  500 IF(WTRDEP)510,600,510
  510 IF(ALT.LE.0.0)GO TO 9900
    IF(WL.LE.0.0)GO TO 9900
    DPWTR=ALT-WL
    IF(WTRDEP.LT.0.0)GO TO 550
    IF(DPWTR.LT.TSTWDP)GO TO 9900
    ISTSW=1
    GO TO 600
  550 IF(DPWTR.GT.TSTWDP)GO TO 9900
    ISTSW=1
C..... TEST FOR SPECIFIED LOG TYPES
C.....
  600 IF(NLOG.EQ.0)GO TO 700
    DO 690 J=1,NLOG
      DO 680 N=1,8
          IF(LOGS(N).EQ.ILOG(J))LGSW=1
      680 CONTINUE
    690 CONTINUE
    IF(LGSW.NE.1)GO TO 9900
    ISTSW=1
C..... TEST FOR WATER LEVEL WITHIN SPECIFIED RANGE
C.....
  700 IF(WTRLVL)710,600,750
  710 IF(WL.GT.TSTWL)GO TO 9900
    ISTSW=1
    GO TO 800
  750 IF(WL.LT.TSTWL)GO TO 9900
    ISTSW=1
C..... TEST FOR SPECIFIED WATER USE CODES
C.....
  800 IF(NWUS.EQ.0)GO TO 900
    DO 890 J=1,NWUS
    IF(INWU.EQ.IWUSE(J))IUSW=1
  890 CONTINUE
    IF(IUSW.NE.1)GO TO 9900
    ISTSW=1
C..... TEST FOR TYPE OF OPENINGS SPECIFIED

900 IF(NOPN.EQ.0)GO TO 1000
DO 990 J=1,NOPN
   IF(MOPN.EQ.IOPN(J))IOPSW=1
990 CONTINUE
   IF(IOPSW.NE.1)GO TO 9900
ISTSW=1

C..... TEST FOR SPECIFIED TYPE OF FINISH CODES

1000 IF(NFIN.EQ.0)GO TO 1100
DO 1090 J=1,NFIN
   IF(MFIN.EQ.IFIN(J))IFSW=1
1090 CONTINUE
   IF(IFSW.NE.1)GO TO 9900
ISTSW=1

C..... TEST FOR WATER LEVEL RECORD COUNT WITHIN SPECIFIED RANGE

1100 IF(NWLVL)1110,1200,1150
1110 IF(WLCNT.GT.IABS(NWLVL))GO TO 9900
   ISTSW=1
   GO TO 1200
1150 IF(WLCNT.LT.IABS(NWLVL))GO TO 9900
   ISTSW=1

C..... TEST FOR SPECIFIC CAPACITY WITHIN SPECIFIED RANGE

1200 IF(SPCAP)1210,1300,1250
1210 IF(SPCP.GT.ABS(SPCAP))GO TO 9900
   ISTSW=1
   GO TO 1400
1250 IF(SPCP.LT.ABS(SPCAP))GO TO 9900

C..... TEST FOR CASING DIAMETER WITHIN SPECIFIED RANGE

1300 IF(CDIAM)1310,1400,1350
1310 IF(CSDM.GT.ABS(CDIAM))GO TO 9900
   ISTSW=1
   GO TO 1400
1350 IF(CSDM.LT.ABS(CDIAM))GO TO 9900
   ISTSW=1

1400 LGSW=0
   IFSW=0
   IUSW=0
   IOPSW=0
   IF(ISTSW.EQ.1)GO TO 9910
9900 ISTSW=0
9910 RETURN
END
SUBROUTINE LABEL

COMMON/RDR/ALT,CSDM,CNSYR,DEPTH,INWU,LOGS(8),MFIN,MOPN,SPCP,TPOPN,WL,WTLR,WLCNT

COMMON/LBLR/I,NID,NID1,NID2,OLBL1(300),OLBL2(300),BTMALT,WALT,SEQ

SUBROUTINE SYMLDM

DIMENSION NOX(7)

COMMON/SYMLD/XL,YL,KSP,HT,ANGLE,NDM

MT = NDM/10000
NT = NT*10000
FND = NT/10000
CALL NUMBER(XL,YL,HT,FND,ANGLE,-1)
YL = YL + 3.0*HT
IF (FND.GE.100.0) YL = YL+HT
NM = NDM-NT
DO 31 I=1,5
  J = 6-I
  IF (I.NE.3) GO TO 30
  NOX(J) = 11
  GO TO 31
30 NMT = NM/10
  NMT = NMT*10
  NOX(J) = NM-NMT+1
  NM = NM/10
31 CONTINUE
DO 41 I=1,5
  NOXK = NOX(I)
  GO TO (10,11,12,13,14,15,16,17,18,19,20),NOXK
10 CALL SYMBOL(XL,YL,HT,"0",ANGLE,1)
  GO TO 41
11 CALL SYMBOL(XL,YL,HT,"1",ANGLE,1)
  GO TO 41
12 CALL SYMBOL(XL,YL,HT,"2",ANGLE,1)
  GO TO 41
13 CALL SYMBOL(XL,YL,HT,"3",ANGLE,1)
  GO TO 41
14 CALL SYMBOL(XL,YL,HT,"4",ANGLE,1)
  GO TO 41
15 CALL SYMBOL(XL,YL,HT,"5",ANGLE,1)
  GO TO 41
16 CALL SYMBOL(XL,YL,HT,"6",ANGLE,1)
  GO TO 41
17 CALL SYMBOL(XL,YL,HT,"7",ANGLE,1)
  GO TO 41
18 CALL SYMBOL(XL,YL,HT,"8",ANGLE,1)
  GO TO 41
19 CALL SYMBOL(XL,YL,HT,"9",ANGLE,1)
  GO TO 41
20 CALL SYMBOL(XL,YL,HT," ",ANGLE,1)
41 YL = YL + HT
RETURN
END