Procurement Executive - Ministry of Defence

AWRE, Aldermaston

AWRE REPORT NO. 037/76

# The Establishment of the Blacknest Seismological Database on the Rutherford Laboratory System 360/195 Computer

C Blamey

ISBN 0 85518111 7

CONTENTS

## SUMMARY

In order to assess the problems which might arise from monitoring a comprehensive test ban treaty by seismological methods, an experimental monitoring operation is being conducted. This work has involved the establishment of a database on the Rutherford Laboratory 360/195 system computer. The database can be accessed in the UK over the public telephone network and in the USA via ARPANET.

## 1. INTRODUCTION

In order to assess the problems which might arise from monitoring a comprehensive test ban treaty by seismological methods, an experimental monitoring operation was initiated in October 1974 at AWRE, Blacknest.

It was envisaged that the experiment would include:-

(a) On site recording and preliminary determinations in several countries.

(b) Communication of data to data centres (and between data centres).

(c) Data handling operations in data centres.

(d) Analysis operations in data centres.

(e) Issuing of prompt assessments of seismic disturbances which appear to be explosions from areas of interest.

Data are supplied on a routine basis from three overseas array stations: GBA (India), WRA (Australia) and YKA (Canada) (the collaborating organisations in the countries concerned are listed in section 2) and from two UK stations. These are the array EKA (South Scotland) and the single seismometer station WOL (South England).

A principal requirement is for a database into which these incoming data are stored in some fixed format with immediate access for analysis and for data file transfer. The Blacknest Data Analysis Centre (BDAC) database is maintained on the Rutherford Laboratory (Didcot, Oxfordshire, England) 360/195 computer system via remote terminals. It can be accessed internally in the UK by computer terminals connected to Remote Job Entry work stations, of which there are about 40, or via the public telephone network, or externally in the US (for example) by users of the Advanced Research Project Agency computer network, ARPANET.

A feature adopted from the outset has been the issue of current data in message and bulletin form to a number of seismological organisations not linked to ARPANET.

This report describes the computer programs written to investigate data handling (see (c) above) and the composition of an event list for the five stations (see (e) above). The work described in this report is in the nature of a pilot study. When the detailed requirements for the database have been more explicitly defined, then the method of its operation will, in all probability, be subjected to review.

## 2. THE SOURCES OF THE DATA

The geographic co-ordinates of each of the seismometer array stations and WOL, from which the data are obtained, are given in table 1. Brief notes on each station are also provided. For a more comprehensive description, see reference [1].

TABLE 1

| Station | Code | Location ° ' " |
|---|---|---|
| Gauribidanur, South India | GBA | 13 36 15 N<br>77 26 10 E |
| Warramunga, Central Australia | WRA | 19 56 52 S<br>134 21 03 E |
| Yellowknife, North-West Canada | YKA | 62 29 34 N<br>114 36 17 W |
| Eskdalemuir, South Scotland | EKA | 55 19 59 N<br>03 09 33 W |
| Wolverton, South England | WOL | 51 18 46 N<br>01 13 22 W |

Of the four arrays, only Yellowknife provides best beam data, but the aspect of the project described here can be developed as long as prompt data are reliably available; only the quantity of data is affected by summing the array. Developments are underway to provide digital capability at the array stations of GBA, WRA and EKA [2] of Seismic Array Station Processors (SASPs) which will provide best beam data similar to those now produced at YKA.

### GBA (India)

The signals are recorded on a helicorder from one seismometer in a 20 element short period array. The records are analysed by GBA staff and selected data are radioed to Trombay and thereafter telexed to Blacknest by courtesy of the Bhabha Atomic Research Centre (Seismology Section) in Trombay.

WRA (Australia)

The signals obtained from one seismometer in a 20 element short period array are recorded on a helicorder. The records are despatched by air to Canberra where the records are analysed and selected data are telexed to Blacknest. These data are provided by courtesy of the Research School of Physical Sciences (Department of Seismology) at the Australian National University.

YKA (Canada)

This station uses a digital processing computer system CANSAM [3] to delay and sum the outputs from the 18 short period seismometers in the array. Events detected above some predetermined level are listed in the CANSAM detection bulletin. This at present is forwarded to Blacknest by airmail. These data are provided by courtesy of the Department of Energy, Mines and Resources, Earth Physics Branch, Division of Seismology and Geothermal Studies, Ottawa.

EKA (Scotland)

For this array station the signals recorded on a helicorder are obtained from the undelayed sum of the outputs of the innermost eight seismometers of a 20 element short period array. The records are analysed locally by the station staff and the analysis results are telexed to AWRE.

WOL (England)

The signals from a single short period seismometer of this standard station are recorded on a helicorder. The records are analysed at Blacknest.

So far the only information used from each seismogram has been the P wave onset time, the amplitude of the largest wave in the first few seconds of ground motion and the corresponding wave period. These data enable the location and size of the seismic sources to be estimated.

## 3. THE BLACKNEST DATABASE AND ITS LINKS

### 3.1 Description of the database

The database is a structured system of files resembling an inverted tree, within the computer system, which can be accessed both by the originators and by accredited external users. Users can enter the files via a main directory (see table 2) containing sub-directories, a file named HELP, and a few other of the more important files. The HELP file contains the basic information about the recording stations and the data contained in the files, and should enable a new user to use the database.

Each station has a sub-directory (see the example given in table 3) devoted to all the monthly data files both current and archived. Other sub-directories contain files in which are kept the programs needed to operate the database.

TABLE 2

Files of Main Directory

(The content and use of these files is explained in later sections.)

ARCHBULL
BULLETIN
BULLFORM
OUT
EKA
GBA
HELP
HOUSEKEP
INDEX76
MAIL
MASTER
MESSBOX
IN
PRELIDET
PREVBULL
SWD
WOL
WORK
WRA
YKA

TABLE 3

Portion of Sub-Directory WOL

| | | | |
|---|---|---|---|
| JAN76 | 3W | 52 | 1/ 1/79 (A) |
| FEB76 | 3W | 124 | 1/ 1/79 (A) |
| MAR76 | 3W | 125 | 1/ 1/79 (A) |
| APR76 | 3W | 147 | 1/ 1/79 (A) |
| MAY76 | 3W | 174 | 1/ 1/79 (A) |
| JUN76 | 3W | 119 | 1/ 1/79 (A) |
| JUL76 | 21998 | 0 | |
| AUG76 | 22005 | 0 | |

## 3.2 The Rutherford Laboratory computer complex

The user's manual, CIGAR [4], is available from the computer section of the laboratory. The Blacknest database is resident within the 360/195 system. To use the database requires the input of commands to the system through an interactive (file handling, job submission

and output retrieval) editor known as ELECTRIC [5]. Several universities in the UK have access to this computer complex and hence are able to access the Blacknest database.

### 3.3 ARPANET

This is a computer network [6] which links the principal computers in the US. The UK node exists at a Terminal Interface Processor (TIP) in University College, London which provides a link via a communications satellite to UK HOST computers for users of the network in the USA. The Rutherford Laboratory 360/195 computer is the British HOST computer for the ARPA network. Any accredited user of the network should be able to access any HOST computer for which permission to access has been granted.

## 4. THE OPERATION OF THE DATABASE

### 4.1 Daily logging of current data - the LOADER suite of programs

The number of lines of data/station received by telex average about 10 a day. Each line of data is of the form day (of month), hour, minute, second, tenths of second, amplitude, period. There are minor variations in this format.

Since these data are manually entered into the computer system via a terminal, it is essential to enter them on a compressed format and reformat within the computer. This has necessitated the writing of programs in FORTRAN IV, ELECTRIC and 360 JOB CONTROL languages. This particular suite of programs will be referred to as the LOADER program (see appendix A).

A brief explanation of the operation of the LOADER program is that it reads into the computer a file which contains the names of the stations which are to be involved in the loading (see the example shown in table 4 - GBA out of a possible set WOL, EKA, GBA, WRA, YKA). Table 5 shows the file of data (compressed) prepared for entry for the station GBA. The program checks the submitted data for various forms of error, writes rejected data entries and/or comments into an errors file (see table 6) and reformats the valid entries (see table 7).

The LOADER program is submitted to the computer using the command EXEC FL=LOAD, MONTH=AUG76 (say). LOADER uses about one second of central processor time.

TABLE 4

Station to be Loaded

GBA

TABLE 5

Compressed Data File for GBA

| | | |
|---|---|---|
| 15171137 | 21 | 0.7 |
| 15184123 | 44 | 0.8 |
| 15224345 | 34 | 0.8 |
| 16052423 | 53 | 0.7 |
| 16124003 | 25 | 0.7 |
| 16141304 | 86 | 1.3 |
| 16161932 | | |
| 16164456 | | |
| 16174528 | 7 | 0.7 |
| 16122443 | 31 | 0.9 |
| 16182539 | 39 | 0.8 |
| 16182939 | 10 | 0.9 |
| 16184141 | | |
| 16190126 | | |
| 16192352 | | |
| 16205549 | 26 | 1.0 |
| 16223759 | 11 | 0.8 |
| 16225757 | | |
| 16232509 | 12 | 0.7 |
| 17000324 | 9 | 1.0 |
| 17011945 | | |
| 17021618 | 18 | 1.0 |
| 17033436 | 18 | 0.9 |
| 17042745 | | |
| 17052712 | | |
| 17062222 | 7 | 0.6 |
| 17064602 | | |
| 17080152 | | |

## TABLE 6

## The Errors File

```
TIME: 15 51 23   DATE: 19 AUG 76

STATION WOL
-----------
***** NOTHING TO APPEND TO WOL FILE

STATION EKA
-----------
***** NOTHING TO APPEND TO EKA FILE

STATION GBA
-----------
DATE (GBA) TO     FILE COMPLETED

STATION WRA
-----------
***** NOTHING TO APPEND TO WRA FILE

STATION YKA
-----------
***** NOTHING TO APPEND TO YKA FILE
```

## TABLE 7

## The Reformated Data in the Database File

| | | | |
|---|---|---|---|
| 15 1711 37 | | 21 | 0.7 |
| 15 1841 23 | | 44 | 0.8 |
| 15 2243 45 | | 34 | 0.8 |
| 16 0524 23 | | 53 | 0.7 |
| 16 1224 43 | | 31 | 0.9 |
| 16 1240 03 | | 25 | 0.7 |
| 16 1413 04 | | 86 | 1.3 |
| 16 1619 32 | | | |
| 16 1644 56 | | | |
| 16 1745 28 | | 7 | 0.7 |
| 16 1825 39 | | 39 | 0.8 |
| 16 1829 39 | | 10 | 0.9 |
| 16 1841 41 | | | |
| 16 1901 26 | | | |
| 16 1923 52 | | | |
| 16 2055 49 | | 26 | 1.0 |
| 16 2237 59 | | 11 | 0.8 |
| 16 2257 57 | | | |
| 16 2325 09 | | 12 | 0.7 |
| 17 0003 24 | | 9 | 1.0 |
| 17 0119 45 | | | |
| 17 0216 18 | | 18 | 1.0 |
| 17 0334 36 | | 18 | 0.9 |
| 17 0427 45 | | | |
| 17 0527 12 | | | |
| 17 0622 22 | | 7 | 0.6 |
| 17 0646 02 | 9 | | |
| 17 0801 52 | | | |

## 4.2 Monthly archiving of older data - the HOUSEKEEPING suite of programs

The allotted number of ELECTRIC blocks of computer storage for the personal identifier under which the Blacknest database is maintained is strictly limited, so it is necessary to put older files of data into archive. Any archived file of data can, however, be restored overnight.

The policy that has been adopted regarding the quantity of data made immediately accessible to users is to keep a minimum of one month, and, because the older data are archived at the end of each month, the amount actually available for each station can rise to a maximum of nearly two months.

At the end of each month a HOUSEKEEPING program called CLEAROUT is executed and puts the oldest month's data files for all stations into archive.

Another HOUSEKEEPING program called UPDATE is then executed which enters the new month's files for all stations into the database.

The form of the job submission of the program CLEAROUT to the computer is EXEC FL=CLEAROUT, MONTH=JUL76 (say). Similarly, for program UPDATE it is EXEC FL=UPDATE, MONTH=AUG76 (say). Both these programs take less than one second of central processor time.

# 5. USES OF THE DATABASE

## 5.1 Internal use

Preliminary analyses (eg, location of the epicentre of a seismic source) are made regularly, using the onset times of P waves. If a source is found to lie in an area of interest, body ($m_b$) and surface wave (Ms) magnitudes are estimated.

The results are then added to two files within the main directory of the database. These files are named INDEX76 (see table 8) and PRELIDET (see table 9). In the file INDEX76 a single line entry is made for each event, with the most recent entry first and the oldest entry last. In the file PRELIDET additional details are given on exactly the same events with the order of entries as in file INDEX76.

The single line entries in the file INDEX76 each contain the following information about the seismic sources: month, day, year, name of area, hour, minute, second, tenths of second, latitude and longitude. The entries in the file PRELIDET additionally contain headings for each item of information and for each station, and each event, if available, the month, day, arrival time (hours, minutes, seconds, tenths of seconds), amplitude (nanometres), period (seconds), body wave magnitude and surface wave magnitude.

## TABLE 8

## Portion of File INDEX76

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| AUG | 28 | 76 | E.KAZAKH | 02 | 56 | 57.3 | 49.9N | 78.9E |
| JUL | 29 | 76 | W.KAZAKH | 04 | 59 | 57.1 | 48.3N | 48.5E |
| JUL | 28 | 76 | E.CAUCASUS | 20 | 17 | 39 | 43.5N | 46.0E |
| JUL | 28 | 76 | TADZHIK/SINK. | 18 | 24 | 22.2 | 39.7N | 73.3E |
| JUL | 23 | 76 | E.KAZAKH | 02 | 32 | 57.4 | 49.7N | 78.0E |
| JUL | 08 | 76 | TURKMEN | 23 | 36 | 7.8 | 40.7N | 57.0E |
| JUL | 04 | 76 | E.KAZAKH | 02 | 56 | 57.3 | 49.8N | 78.8E |
| JUN | 09 | 76 | E.KAZAKH | 03 | 02 | 57.0 | 50.0N | 79.0E |
| MAY | 18 | 76 | TURKMEN SSR | 08 | 57 | 23.6 | 39.2N | 62.4E |
| MAY | 18 | 76 | W.RUSSIA | 04 | 19 | 27.8 | 50.6N | 36.6E |
| MAY | 17 | 76 | TURKMEN SSR | 02 | 58 | 37.2 | 38.9N | 62.4E |
| APR | 21 | 76 | E.KAZAKH | 05 | 02 | 57.1 | 49.8N | 78.8E |
| APR | 21 | 76 | E.KAZAKH | 04 | 57 | 56.5 | 49.8N | 78.4E |
| MAR | 20 | 76 | E.KAZAKH | 04 | 03 | 39.0 | 50.0N | 77.4E |
| MAR | 17 | 76 | NEVADA | 14 | 45 | 1.0 | 38.1N | 116.9W |
| MAR | 17 | 76 | NEVADA | 14 | 15 | 0.8 | 38.1N | 116.9W |
| MAR | 14 | 76 | NEVADA | 12 | 30 | 0.9 | 38.2N | 117.0W |
| MAR | 09 | 76 | NEVADA | 14 | 00 | 0.9 | 38.2N | 117.0W |
| FEB | 14 | 76 | NEVADA | 11 | 30 | 5.3 | 39.6N | 118.0W |
| FEB | 12 | 76 | NEVADA | 14 | 45 | 4.9 | 39.7N | 118.4W |
| FEB | 04 | 76 | NEVADA | 14 | 40 | 1.9 | 38.2N | 116.8W |
| FEB | 04 | 76 | NEVADA | 14 | 20 | 5.2 | 39.6N | 118.0W |
| JAN | 15 | 76 | E.KAZAKH | 04 | 46 | 54.7 | 50.1N | 79.3E |
| JAN | 10 | 76 | S.SINKIANG | 12 | 51 | 19.9 | 42.0N | 83.3E |
| JAN | 03 | 76 | NEVADA | 19 | 15 | 0.9 | 38.1N | 116.9W |

## TABLE 9

## Portion of File PRELIDET

| DATE | REGION | ORIGIN TIME | LAT | LONG |
|---|---|---|---|---|
| AUG 28 76 | E.KAZAKH | 02 56 57.3 | 49.9N | 78.9E |

| STN | MONTH | DAY | ARRIVAL TIME HOURS | MINS | SECS | AMP | PER | MB | MS |
|---|---|---|---|---|---|---|---|---|---|
| WOL | AUG | 28 | 03 | 05 | 42.4 | 117. | 0.7 | 6.1 | 3.8 |
| EKA | | 28 | 03 | 05 | 34.3 | 45 | 0.5 | 5.9 | |
| GBA | | 28 | 03 | 04 | 4.0 | | | | |
| WRA | | 28 | 03 | 09 | 36.1 | 101.1 | 0.9 | 6.0 | |

| DATE | REGION | ORIGIN TIME | LAT | LONG |
|---|---|---|---|---|
| JUL 29 76 | W.KAZAKH | 04 59 57.1 | 48.3N | 48.5E |

| STN | MONTH | DAY | ARRIVAL TIME HOURS | MINS | SECS | AMP | PER | MB | MS |
|---|---|---|---|---|---|---|---|---|---|
| WOL | JUL | 29 | 05 | 06 | 24.9 | 273 | 0.5 | 6.4 | 3.8 |
| EKA | | 29 | 05 | 06 | 27.6 | 108 | 0.6 | 6.0 | |
| GBA | | 29 | 05 | 07 | 52.0 | 40 | 0.6 | 6.2 | |
| WRA | | | | | | | | | |

| DATE | REGION | ORIGIN TIME | LAT | LONG |
|---|---|---|---|---|
| JUL 28 76 | E.CAUCASUS | 20 17 39 | 43.5N | 46.0E |

| STN | MONTH | DAY | ARRIVAL TIME HOURS | MINS | SECS | AMP | PER | MB | MS |
|---|---|---|---|---|---|---|---|---|---|
| WOL | JUL | 28 | 20 | 24 | 12.1 | 28 | 0.7 | 5.2 | |
| EKA | | 28 | 20 | 24 | 20.6 | 14 | 0.7 | 5.0 | |
| GBA | | 28 | 20 | 25 | 19 | 23 | 0.9 | 4.9 | |
| WRA | | | | | | | | | |

| DATE | REGION | ORIGIN TIME | LAT | LONG |
|---|---|---|---|---|
| JUL 28 76 | TADZHIK/SINK. | 18 24 22.2 | 39.7N | 73.3E |

| STN | MONTH | DAY | ARRIVAL TIME HOURS | MINS | SECS | AMP | PER | MB | MS |
|---|---|---|---|---|---|---|---|---|---|
| WOL | JUL | 28 | 18 | 33 | 30 | 41 | 0.9 | 5.4 | |
| EKA | | 28 | 18 | 33 | 29.2 | 11 | 0.8 | 4.9 | |
| GBA | | 28 | 18 | 30 | 01 | 39 | 0.9 | 5.2 | |
| WRA | | 28 | 18 | 36 | 46.9 | 14.1 | 0.8 | 5.0 | |

## 5.2 External use

As already stated, organisations connected to the ARPANET can access the data in the existing format [7].

By arrangement, users within the UK may retrieve data via normal datacom systems over the public telephone network.

## 5.3 The composition of an event list - the BULLETIN suite of programs

A complete event list is issued every two weeks and is made up of two parts. Part 1 gives chronological lists of seismic events for each station where data are contained in the Blacknest database. Part 2 gives the analysis of selected events for the same two-week period.

This particular suite of computer programs (see appendix B), which has been written to compose each event list, is the BULLETIN program. This program has been written so that the minimum of computer terminal entry work, typically a few minutes, is required.

As an indication of the amount of entry work involved, see table 10. This shows a possible set of input data to produce an event list. The entries are line 1, event list number, year; line 2, the start day date (inclusive) and the finish day date (inclusive); line 3, the first month, the second month (if any); the entry in line 4 indicates that no data for station YKA are to be included.

TABLE 10

Typical Input Parameters Used to Produce a Particular Event List

```
15 76
19 01
JUL AUG
NOYKA
ERRATA
ADDENDA
PART2BUL 001 044
```

Entries in lines 5 and 6 indicate that there are errata and addenda sections to be included. This requires that relevant errata and addenda information are put in two files with the names ERRATA and ADDENDA which are resident in the system. The final entry (line 7 in table 10) shows that there is to be a part 2 to the event list and it is to consist of the events contained in lines 1 to 44 of the file PRELIDET.

The relevant database files (for part 1) are concatenated and the specified range of data is selected. Similarly, the specified portion of the analysis file PRELIDET is selected for part 2 and the two portions are brought together to compose the completed current event list. The BULLETIN program is submitted to the computer with the command EXEC FL=BULLETIN(P8), MONTH1=JUL76, MONTH2=AUG76 where, in this

particular situation, the span of the event list extends from one month into the next. If the event list is contained within a single month, then the submission would be (for example) EXEC FL=BULLETIN(P8), MONTH1=SEP76, MONTH2=SEP76. The BULLETIN program takes a few seconds of central processor time to execute.

The BULLETIN program has been arranged so that, even if the ELECTRIC system is not available to the 360/195 computer, a copy of the current event list is always written on the permanent disc facilities within the 360/195 system. It can be copied from there to the file named BULLETIN in the main directory for the Blacknest identifier at some later time. In order to do this use is made of the facility within the ELECTRIC system of copying from the 360/195 system to an ELECTRIC file using the command COPY FL=BULLETIN, FROMDSN=XA3W.BULLETIN, VOL=RHELO3. Normally, however, this is not necessary because the process of copying into ELECTRIC is done by the computer.

Before a new event list is composed, the old one is archived in a sub-directory ARCHBULL in the main directory.

## 6. THE DEVELOPMENT OF THE DATABASE

### 6.1 Useful procedures

A problem encountered in the development of the database is that of keeping adequate records of the rapidly changing files and their contents. These procedures have proved to be helpful:-

(a) Use of alpha numeric names for program files.

(b) The facility of being able to order (alphabetically) the files in the sub-directories.

(c) Retention of archived copies of programs in a sub-directory named MASTER providing security against accidental deletion of any program while working on it.

The files in the main directory of the database can only be alphabetically ordered by the 360/195 advisory staff. However, execution of the HOUSEKEEPING program ORDERSUB will order any sub-directory.

For example, provided that all the files in the sub-directory MASTER are in the archived state (for more details see the program listing in appendix C), then the job submission EXEC FL=ORDERSUB, DIRECTOR=MASTER will result in the sub-directory MASTER being ordered alphabetically.

### 6.2 Future aims

(a) To widen the seismological information contained in the data by including details of other signals and waveform description.

(b) To transfer data via ARPANET.

(c) To explore the possibility of developing a program to extract groups of associated onset times from the data routinely available to Blacknest for the following:-

(i) An international network of array stations (GBA, WRA, YKA, EKA).

(ii) A UK national network [8] of single seismometer stations.

Subsequently to locate the epicentre of the source for each group of onset times. At present the association of times is done manually.

## 7. CONCLUSIONS

The work involved in this report has provided experience in the operation of a seismological database. Although the quantity of data dealt with on a daily basis is not large, the experiment has already provided material of interest to current activities of the Disarmament Unit of the Foreign and Commonwealth Office and when complete it is expected to make a significant contribution to its work for the banning of nuclear tests.

## 8. ACKNOWLEDGMENTS

I wish to thank the following organisations and personnel for contributing to the work underlying this report:-

(a) Rutherford Laboratory, Computing and Automation Division

Mr D Parker and Mr P J Hemmings for the speedy resolution of computer programming problems as they arose in the course of this work and their comments on the draft report.

(b) University College, London

Professor P T Kirstein, Mrs S Wilbur and Mr H Gamble for their helpful co-operation.

(c) The four seismological stations GBA, WRA, YKA and EKA for contributing data

GBA - Mr G Varghese and staff.
WRA - Professor A Hales and staff, in particular Mrs J Weekes.
YKA - Dr K Whitham and staff.
EKA - Mr G McKenzie and staff.

(d) Blacknest Data Analysis Centre

Dr H I S Thirlaway and Mr F H Grover for constructive criticism of the draft report, Mrs P Warburton for assistance with program corrections, and Mr P G Gibbs for supplying the WOL station data.

## REFERENCES


1. "Handbook for Stations which Provide Seismograms to the Blacknest Seismological Centre, United Kingdom". AWRE Report 44/47/29 (January 1974)

2. F A Key, T G Lea and A Douglas: "Seismic Array Station Processors". AWRE Report 036/76

3. D H Weichert and M Henger: "The Canadian Seismic Array Monitor Processing System (CANSAM)". Bulletin of the Seismological Society of America, 66, (4), 1381 (1976)

4. "Computer Introductory Guide and Reference". Science Research Council, Rutherford Laboratories, Chilton, Didcot, Oxfordshire

5. "ELECTRIC Users Manual". SRC, RHEL 72-003

6. T Johnson: "Electronic Post for Switching Data". New Scientist, 70, (1000), 351 (13 May 1976)

7. J S Derr, M A Carlson and A W Egdorf: "Seismic Data Flow at the National Earthquake Information Service". IEEE 1976 National Telecommunications Conference, 3, 46.4-1 - 46.4-4 (November 1976)

8. F H Grover: Private Communication

## APPENDIX A

## LOADER PROGRAMS

PROGRAM LISTINGS
**********

GENERAL NOTE WHICH APPLIES TO THE PROGRAMS IN THE THREE APPENDICES

ELSEND AND ELDIRE ARE PROCEDURES USED FOR SENDING MESSAGES TO ELECTRIC FROM A PROGRAM RUNNING IN THE CENTRAL COMPUTER.

APPENDIX 1

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

THE LOADER SUITE OF PROGRAMS.

| | SUB-DIRECTORY NAME | | ELECTRIC FILE NAME |
|---|---|---|---|
| A.(JOB CONTROL + ELECTRIC) PROGRAMS. | | | |
| (1)LOADER | IN | | LOAD |
| (2)LOADER.ED | IN | | LOAD.ED |
| (3)LOADER* | MASTER | | LOAD50 |
| (4)LOADER.ED* | MASTER | | LOAD50.ED |
| B. ELECTRIC PROGRAMS. | | | |
| (1)DATA | IN | | DATA |
| (2)DATA.ED | IN | | DATA.ED |
| (3)DATA* | MASTER | | DATA50 |
| (4)DATA.ED* | MASTER | | DATA50.ED |
| C. FORTRAN 4 PROGRAMS. | | BINARY VERSION** | |
| (1)MAIN | MASTER | XA3WUP50 | UNPACO50 |
| (2)CHARRV | MASTER | XA3WDC50 | DECODE50 |
| (3)INTERP | MASTER | XA3WLI50 | LININT50 |
| (4)MAIN(2) | MASTER | XA3WSC50 | SCAN50 |
| (5)MONTOR | MASTER | XA3WMT50 | MONTOR50 |
| (6)SCANFL | MASTER | XA3WSF50 | SCANFL50 |
| D. MARKER FOR END OF DATA FILES (SEE APPENDIX 2). | | | |
| (1)ENDFILE | IN | | ENDFILE |

E.OTHER PROGRAMS NOT INCLUDED SEE 360 COMPUTER MANUAL LISTED IN REFERENCES.

(1)CLOCK
(2)CLFILE

F. PERMANENT DATASETS ON THE 360/195 SYSTEM.

| DATASET NAME | CONTENTS |
|---|---|
| (1)XA3W.STATLOAD | LAST STATIONS WHOSE FILES WERE APPENDED TO. |
| (2)XA3W.WOLDATA | REPLICA OF LATEST WOL DATA FILE IN ELECTRIC. |
| (3)XA3W.EKADATA | REPLICA OF LATEST EKA DATA FILE IN ELECTRIC. |
| (4)XA3W.GBADATA | REPLICA OF LATEST GBA DATA FILE IN ELECTRIC. |
| (5)XA3W.WRADATA | REPLICA OF LATEST WRA DATA FILE IN ELECTRIC. |
| (6)XA3W.YKADATA | REPLICA OF LATEST YKA DATA FILE IN ELECTRIC. |

NOTES * INDICATES THAT THESE PROGRAMS (ARCHIVED IN SUB-DIRECTORY MASTER) ARE DUPLICATE COPIES OF THE DATABASE PROGRAMS IN ROUTINE USE.

** THE BINARY PROGRAMS ARE STORED ON DISK IN THE SEISMOLOGY GROUP LIBRARY NAMED ULIB.ISC.

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

```
/*PRIORITY     12
//XA3WLOAD      JOB (ACCT,ID,0-06),BLAMEY
/*ROUTE  PRINT ELECTRIC
//*
//* ------------------------------------------------------------------
//*
//*                        LOADER PROGRAM
//*                        **************
//*
//* (360 JOB CONTROL AND ELECTRIC LANGAUGES ) PROGRAM USED TO LOAD DATA
//* TO THE BLACKNEST DATA ANALYSIS CENTRE DATA FILES ON THE 360/195
//* SYSTEM COMPUTER AT RUTHERFORD LABORATORY ,DIDCOT ,OXON ,UK.
//*
//* NOTE (1)  THIS ELECTRIC FILE LOAD HAS A CORRESPONDING EDIT FILE
//*           LOAD.ED.
//*
//* NOTE (2)  THE FORTRAN PROGRAMS USED BY THIS PROGRAM ARE IN THE
```

```
//*            FORM OF LOAD MODULES (I.E. IN BINARY). THE FORTRAN
//*            PROGRAMS ARE COMPILED SEPERATELY AND WRITTEN INTO
//*            THE COMPUTER LIBRARY FROM WHENCE THE LOADER PROGRAM
//*            CALLS THEM UP.
//*
//* NOTE (3)   INPUT REQUIREMENTS OF THE LOADER PROGRAM.
//*            ----------------------------------------
//*            THE NAMES OF THE FILES INTO WHICH INFORMATION AND DATA
//*            ARE TO BE PUT (ALL IN SUB-DIRECTORY 'IN') ARE AS FOLLOWS -
//*            (A)S INTO WHICH THE STATION NAMES (THE COMPLETE SET
//*               IS WOL,EKA,GBA,WRA,YKA) WHOSE DATA ARE TO BE LOADED
//*               IN THE FORM OF ONE ENTRY PER LINE STARTING IN COLUMN 1.
//*
//*            (B)WOLM,EKAM,GBAM,WRAM,YKAM WHICH ARE THE CORRESPONDING
//*               FILES INTO WHICH COMPRESSED DATA ARE PUT, STARTING IN
//*               COLUMN 1 WITH A LAYOUT IN EACH LINE OF DAYONSET TIME
//*               XAMPLITUDEXPERIOD WHERE THE X INDICATES A SINGLE SPACE
//*               BETWEEN DATA ENTRIES.
//*
//* NOTE (4)   EXECUTION OF THE PROGRAM IS BY SUBMITTING THE COMMAND -
//*            ------------------------
//*            EXEC FL=LOAD,MONTH=NOV76
//*            ------------------------
//*               (MONTH = NAME OF FILES IN THE DATABASE TO WHICH DATA
//*                IS TO BE APPENDED.)
//*
//* ------------------------------------------------------------------
//*
//* STAGE 1. REFORMAT THE DATA ENTRIES.
//*
//* ------------------------------------------------------------------
//*
//* ********************
// EXEC FHLG,REGION.G=57K
//* ********************
//L.LIB  DD  DSN=ULIB.ISC,DISP=SHR
//L.SYSIN DD *
   INCLUDE LIB(XA3WUP50,XA3WDC50,XA3WLI50)
   ENTRY MAIN
//* NOTE   CARD10=(RECFM=FB,LRECL=80,BLKSIZE=800)
//G.FT11F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCA1,DISP=(NEW,PASS),DCB=CARD10
//G.FT12F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCA2,DISP=(NEW,PASS),DCB=CARD10
//G.FT13F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCA3,DISP=(NEW,PASS),DCB=CARD10
//G.FT15F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCA5,DISP=(NEW,PASS),DCB=CARD10
//G.FT16F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCA6,DISP=(NEW,PASS),DCB=CARD10
//G.FT21F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCB1,DISP=(NEW,PASS),DCB=CARD10
//G.FT22F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCB2,DISP=(NEW,PASS),DCB=CARD10
//G.FT23F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCB3,DISP=(NEW,PASS),DCB=CARD10
//G.FT25F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCB5,DISP=(NEW,PASS),DCB=CARD10
//G.FT26F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCB6,DISP=(NEW,PASS),DCB=CARD10
//G.SYSIN  DD  *
/*
//*
//* ------------------------------------------------------------------
//* STAGE 2. WRITE REFORMATTED DATA TO BLACKNEST DATABASE FILES.
//*          THEN PUT DATA ENTRIES IN RESERVE FILES OVERWRITING THE
//*          OLDEST DATA ENTRIES.
//* ------------------------------------------------------------------
//*
//*   ****************************
//S1  EXEC  PGM=IEFBR14,REGION=4K
//*   ****************************
//* ----DELETE STATION PERMANENT DATASETS BEFORE REFRESHING THEM.
//DEL   DD VOL=REF=RHEL03,DISP=(OLD,DELETE),DSN=XA3W.STATLOAD
//DEL   DD VOL=REF=RHEL03,DISP=(OLD,DELETE),DSN=XA3W.WOLDATA
//DEL   DD VOL=REF=RHEL03,DISP=(OLD,DELETE),DSN=XA3W.EKADATA
//DEL   DD VOL=REF=RHEL03,DISP=(OLD,DELETE),DSN=XA3W.GBADATA
//DEL   DD VOL=REF=RHEL03,DISP=(OLD,DELETE),DSN=XA3W.WRADATA
//DEL   DD VOL=REF=RHEL03,DISP=(OLD,DELETE),DSN=XA3W.YKADATA
/*
//*
//* ************************
//  EXEC  ELSEND,REGION=60K
//* ************************
//G.FT11F001 DD DSN=&&SCA1,DISP=(OLD,PASS),UNIT=WORK
//G.FT12F001 DD DSN=&&SCA2,DISP=(OLD,PASS),UNIT=WORK
//G.FT13F001 DD DSN=&&SCA3,DISP=(OLD,PASS),UNIT=WORK
//G.FT15F001 DD DSN=&&SCA5,DISP=(OLD,PASS),UNIT=WORK
//G.FT16F001 DD DSN=&&SCA6,DISP=(OLD,PASS),UNIT=WORK
//G.SYSIN  DD  *
LOGIN ID=ID,ACCT=ACCT,KEY=XXXX
SETD C=A.WOL
STREAM FT=11,NCARDS=1,L1=1,L2=9999
APPEND FL=MONTH
$$
SETD C=A.EKA
STREAM FT=12,NCARDS=1,L1=1,L2=9999
APPEND FL=MONTH
$$
SETD C=A.GBA
STREAM FT=13,NCARDS=1,L1=1,L2=9999
APPEND FL=MONTH
```

```
&&
SETD C=A.WRA
STREAM FT=15,NCARDS=1,L1=1,L2=9999
APPEND FL=MONTH
&&
SETD C=A.YKA
STREAM FT=16,NCARDS=1,L1=1,L2=9999
APPEND FL=MONTH
&&
SETD C=A.IN
DELETE FL=WOLMRB
DELETE FL=EKAMRB
DELETE FL=GBAMRB
DELETE FL=WRAMRB
DELETE FL=YKAMRB
COPY FL1=WOLMRA,FL2=WOLMRB
COPY FL1=EKAMRA,FL2=EKAMRB
COPY FL1=GBAMRA,FL2=GBAMRB
COPY FL1=WRAMRA,FL2=WRAMRB
COPY FL1=YKAMRA,FL2=YKAMRB
DELETE FL=WOLMRA
DELETE FL=EKAMRA
DELETE FL=GBAMRA
DELETE FL=WRAMRA
DELETE FL=YKAMRA
COPY FL1=WOLM,FL2=WOLMRA
COPY FL1=EKAM,FL2=EKAMRA
COPY FL1=GBAM,FL2=GBAMRA
COPY FL1=WRAM,FL2=WRAMRA
COPY FL1=YKAM,FL2=YKAMRA
CLEAR FL=WOLM
CLEAR FL=EKAM
CLEAR FL=GBAM
CLEAR FL=WRAM
CLEAR FL=YKAM
COPY FL=A.IN.S,TODSN=XA3W.STATLOAD,VOL=RHEL03,DISP=NEW
COPY FL=A.WOL.MONTH,TODSN=XA3W.WOLDATA,VOL=RHEL03,DISP=NEW
COPY FL=A.EKA.MONTH,TODSN=XA3W.EKADATA,VOL=RHEL03,DISP=NEW
COPY FL=A.GBA.MONTH,TODSN=XA3W.GBADATA,VOL=RHEL03,DISP=NEW
COPY FL=A.WRA.MONTH,TODSN=XA3W.WRADATA,VOL=RHEL03,DISP=NEW
COPY FL=A.YKA.MONTH,TODSN=XA3W.YKADATA,VOL=RHEL03,DISP=NEW
CLEAR FL=S
LOGOUT
/*
//*
//* ------------------------------------------------------------------
//* STAGE 3. SCAN EACH DATABASE FILE (TO WHICH DATA HAS BEEN APPENDED)
//*          FOR MISSING DAYS. WRITE APPROPIATE COMMENTS TO THE ERRORS
//*          FILE IF NECESSARY.
//* ------------------------------------------------------------------
//*
//* ********************
// EXEC FHLG,REGION.G=25K
//* ********************
//L.LIB  DD  DSN=ULIB.ISC,DISP=SHR
//L.SYSIN DD *
  INCLUDE LIB(XA3WSC03,XA3WMT01,XA3WSF02)
  ENTRY MAIN
//G.FT30F001  DD  DSN=XA3W.STATLOAD,DISP=OLD,VOL=REF=RHEL03
//G.FT31F001  DD  DSN=XA3W.WOLDATA,DISP=OLD,VOL=REF=RHEL03
//G.FT32F001  DD  DSN=XA3W.EKADATA,DISP=OLD,VOL=REF=RHEL03
//G.FT33F001  DD  DSN=XA3W.GBADATA,DISP=OLD,VOL=REF=RHEL03
//G.FT34F001  DD  DSN=XA3W.WRADATA,DISP=OLD,VOL=REF=RHEL03
//G.FT35F001  DD  DSN=XA3W.YKADATA,DISP=OLD,VOL=REF=RHEL03
//*
//G.FT41F001 DD UNIT=WORK,SPACE=(TRK,(2,1),RLSE),
// DSN=&&SCH1,DISP=(NEW,PASS),DCB=CARD10
//G.FT42F001 DD UNIT=WORK,SPACE=(TRK,(2,1),RLSE),
// DSN=&&SCH2,DISP=(NEW,PASS),DCB=CARD10
//G.FT43F001 DD UNIT=WORK,SPACE=(TRK,(2,1),RLSE),
// DSN=&&SCH3,DISP=(NEW,PASS),DCB=CARD10
//G.FT44F001 DD UNIT=WORK,SPACE=(TRK,(2,1),RLSE),
// DSN=&&SCH4,DISP=(NEW,PASS),DCB=CARD10
//G.FT45F001 DD UNIT=WORK,SPACE=(TRK,(2,1),RLSE),
// DSN=&&SCH5,DISP=(NEW,PASS),DCB=CARD10
//*
/*
//* ************
//  EXEC ELSEND
//* ************
//G.FT21F001 DD DSN=&&SCB1,DISP=(OLD,PASS),UNIT=WORK
//G.FT22F001 DD DSN=&&SCB2,DISP=(OLD,PASS),UNIT=WORK
//G.FT23F001 DD DSN=&&SCB3,DISP=(OLD,PASS),UNIT=WORK
//G.FT25F001 DD DSN=&&SCB5,DISP=(OLD,PASS),UNIT=WORK
//G.FT26F001 DD DSN=&&SCB6,DISP=(OLD,PASS),UNIT=WORK
//G.FT41F001 DD DSN=&&SCH1,DISP=(OLD,PASS),UNIT=WORK
//G.FT42F001 DD DSN=&&SCH2,DISP=(OLD,PASS),UNIT=WORK
//G.FT43F001 DD DSN=&&SCH3,DISP=(OLD,PASS),UNIT=WORK
//G.FT44F001 DD DSN=&&SCH4,DISP=(OLD,PASS),UNIT=WORK
//G.FT45F001 DD DSN=&&SCH5,DISP=(OLD,PASS),UNIT=WORK
//G.SYSIN  DD  *
LOGIN ID=ID,ACCT=ACCT,KEY=XXXX
SETD C=A.IN
DELETE FL=ERRORSB
COPY FL1=ERRORSA,FL2=ERRORSB
DELETE FL=ERRORSA
COPY FL1=ERRORS,FL2=ERRORSA
CLEAR FL=ERRORS
STREAM FT=21,NCARDS=1,L1=1,L2=9999
APPEND FL=ERRORS
&&
STREAM FT=41,NCARDS=1,L1=1,L2=9999
```

```
APPEND FL=ERRORS
&&
STREAM FT=22,NCARDS=1,L1=1,L2=9999
APPEND FL=ERRORS
&&
STREAM FT=42,NCARDS=1,L1=1,L2=9999
APPEND FL=ERRORS
&&
STREAM FT=23,NCARDS=1,L1=1,L2=9999
APPEND FL=ERRORS
&&
STREAM FT=43,NCARDS=1,L1=1,L2=9999
APPEND FL=ERRORS
&&
STREAM FT=25,NCARDS=1,L1=1,L2=9999
APPEND FL=ERRORS
&&
STREAM FT=44,NCARDS=1,L1=1,L2=9999
APPEND FL=ERRORS
&&
STREAM FT=26,NCARDS=1,L1=1,L2=9999
APPEND FL=ERRORS
&&
STREAM FT=45,NCARDS=1,L1=1,L2=9999
APPEND FL=ERRORS
&&
LOGOUT
/* END OF JOB XA3WLOAD

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 LOADER.ED
 ---------

&S LB= 1,LN=  78,FL=3WMAINDR.IN       .DATA     (  0,***,DG),NM=G1
&P LB= 1,LN= 111,C1= 11,C2= 15,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN= 115,C1= 11,C2= 15,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN= 119,C1= 11,C2= 15,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN= 123,C1= 11,C2= 15,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN= 127,C1= 11,C2= 15,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN= 156,C1= 15,C2= 19,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN= 157,C1= 15,C2= 19,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN= 158,C1= 15,C2= 19,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN= 159,C1= 15,C2= 19,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN= 160,C1= 15,C2= 19,CH=AN,DF=NO,NM=MONTH

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 DATA
 ----
      DATA IS AN EMPTY FILE.

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 DATA.ED
 -------

&A LB= 1,LN=   0,FL=3WMAINDR.IN       .S        (  0,***,NO)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .ENDFILE  (  0,***,NO)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .WOLM     (  0,***,YS)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .ENDFILE  (  0,***,NO)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .EKAM     (  0,***,YS)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .ENDFILE  (  0,***,NO)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .GBAM     (  0,***,YS)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .ENDFILE  (  0,***,NO)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .WRAM     (  0,***,YS)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .ENDFILE  (  0,***,NO)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .YKAM     (  0,***,YS)
&A LB= 1,LN=   0,FL=3WMAINDR.IN       .ENDFILE  (  0,***,NO)

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
C
C*************************************************************
C
C                        PROGRAM MAIN
C                        ************
C THIS IS A FORTRAN 4 PROGRAM USED TO LOAD DATA TO THE BLACKNEST
C DATABASE FILES.
C NOTE  COMMENTS HAVE BEEN INSERTED IN ALL PROGRAMS TO AID THE
C READER .
C
C*************************************************************
C
C
      DIMENSION ALPHAC(24,100),NCHARN(10),SPARE(24),STNIN(9)
      DIMENSION WRADAT(24,100),TAG1(100),TAG2(100),PEROD(32),RELMAG(32)
C
      DATA NCHARN/4H0    ,4H1    ,4H2    ,4H3    ,4H4    ,4H5    ,4H6    ,
     14H7    ,4H8    ,4H9    /
      DATA DECIPT/4H.    /,SBLANK/4H    /
      DATA AA/4HA    /,AB/4HB    /,AC/4HD    /
      DATA AN/4HN    /,AC/4HG    /,AT/4HT    /,AU/4HU    /
      DATA AF/4HF    /,AI/4HI    /,AL/4HL    /
C
C-----DATA SOURCE FIG 7.10 WRA CURVE BLACKNEST INTERNAL NOTE AG/157
      DATA PEROD/0.2,0.25,0.3,0.35,0.4,0.45,0.5,0.55,0.6,0.65,0.7,
     10.75,0.8,0.85,0.9,0.95,1.0,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9,
     12.0,2.1,2.2,2.3,2.4,2.5/
C
      DATA RELMAG/8.0,6.8,5.8,5.1,4.47,3.95,3.55,3.1,2.8,2.45,2.2,
     11.95,1.74,1.53,1.33,1.15,1.0,0.72,0.53,0.39,0.29,0.21,0.155,
```

```
     10.113,0.084,0.060,0.044,0.033,0.024,0.018,0.014,0.010/
C
      INTEGER SPARE,PREVHR,PREVMN,PREVSC
      INTEGER ALPHAC,COMMA,DECIPT,SBLANK,ZERO,PREVDY
      INTEGER AA,AB,AD,AE,AF,AI,AL,AN,AO,AT,AU
      INTEGER DP,CH,SB,U1,U2,U3,C1,C2
      INTEGER YWOL,YEKA,YGBA,YWRA,YYKA
      INTEGER WRADAT,R1,R2,R3,CW,DW
C
      REAL*8  SECS,AMP,PERIOD,PRVSEC,RV,TIME,DATE
      REAL*8  STNIN,ENDFIL/8HFILE-END/,WASTE
      REAL*8  STNWOL/8HWOL      /,STNEKA/8HEKA      /,STNGBA/8HGBA      /
      REAL*8  STNWRA/8HWRA      /,STNYKA/8HYKA      /
      REAL*8  MICRCV,MAGNIF,HTCAL,PERCD,RELMAG,RELMGN,TAG1,TAG2
C
C
      CALL CLOCK(TIME,DATE)
C
      N0=NCHARN(1)
      N1=NCHARN(2)
      N2=NCHARN(3)
      N3=NCHARN(4)
      N4=NCHARN(5)
      N5=NCHARN(6)
      N6=NCHARN(7)
      N7=NCHARN(8)
      N8=NCHARN(9)
      N9=NCHARN(10)
      DP=DECIPT
      SB=SBLANK
C
C-----READ IN STATIONS TO BE LOADED
      NUE=21
      WRITE (NUE,789)TIME,DATE
 789  FORMAT(/2X,'TIME    ',A8,5X,'DATE  ',A8)
      IK=1
 312  READ 310,STNIN(IK)
 310  FORMAT(A8)
      WRITE (6,313)STNIN(IK)
 313  FORMAT(2X,A8)
      IF(STNIN(IK).EQ.ENDFIL)GO TO 311
      IK=IK+1
      IF(IK.GT.6)GO TO 309
      GO TO 312
C
 309  WRITE (NUE,308)
 308  FORMAT(2X,'--NO. OF STATIONS GREATER THAN 5'
     1,4X,'--LOADING REJECTED')
      GO TO 20
C
C-----CHECK STATION SUBMISSIONS(WOL,ETC.)
C-----LOADED IN ANY ORDER
 311  IF(IK.GT.1)GO TO 3110
      WRITE (NUE,3111)
 3111 FORMAT(2X,5H*****,1X,
     1'NOTHING WAS PUT IN FILE S , SO NOTHING WAS ADDED'/8X,
     1'TO THE ARPA-NET FILES. TO RECOVER THE SITUATION'/8X,
     1'QUICKLY ZA ETC  FROM THE RESERVE FILES '/8X,
     1'WOLMRA,EKAMRA,--- TO THE CURRENT FILES WOLM,---'/8X,
     1'ADD ANY NEW DATA TO THE REVELANT DATA FILE'/8X,
     1'WOLM,--  AND RERUN THE LOADER.')
      GO TO 3110
 3110 YWOL=0
      YEKA=0
      YGBA=0
      YWRA=0
      YYKA=0
C
      ICNT=IK-1
      IF(ICNT.EQ.0)GO TO 1001
      DO 316 I=1,ICNT
      IF(STNIN(I).EQ.STNWOL)YWOL=1
      IF(STNIN(I).EQ.STNEKA)YEKA=1
      IF(STNIN(I).EQ.STNGBA)YGBA=1
      IF(STNIN(I).EQ.STNWRA)YWRA=1
      IF(STNIN(I).EQ.STNYKA)YYKA=1
 316  CONTINUE
C
C-----PREPARE TO LOAD WOL DATA
 1001 NUD=11
      NUE=21
      KL=1
      IF(YWOL.EQ.1)GO TO 333
      WRITE (NUE,120)
      WRITE (NUE,124)
      WRITE (NUD,315)
 315  FORMAT(10X)
      REWIND NUD
      WRITE (NUE,3150)
 3150 FORMAT(2X,5H*****,1X,'NOTHING TO APPEND TO WOL FILE')
      READ 310,WASTE
      WRITE (6,313)WASTE
      GO TO 102
C
C-----PREPARE TO LOAD EKA DATA
 1002 NUD=12
      NUE=22
      KL=2
      IF(YEKA.EQ.1)GO TO 333
      WRITE (NUE,121)
      WRITE (NUE,124)
      WRITE (NUD,315)
```

```
      REWIND NUD
      WRITE (NUE,3151)
 3151 FORMAT(2X,5H*****,1X,'NOTHING TO APPEND TO EKA FILE')
      READ 310,WASTE
      WRITE (6,313)WASTE
      GO TO 103
C
C-----PREPARE TO LOAD GBA DATA
 1003 NUD=13
      NUE=23
      KL=3
      IF(YGBA.EQ.1)GO TO 333
      WRITE (NUE,122)
      WRITE (NUE,124)
      WRITE (NUD,315)
      REWIND NUD
      WRITE (NUE,3152)
 3152 FORMAT(2X,5H*****,1X,'NOTHING TO APPEND TO GBA FILE')
      READ 310,WASTE
      WRITE (6,313)WASTE
      GO TO 104
C
C-----PREPARE TO LOAD WRA DATA
 1005 NUD=15
      NUE=25
      KL=5
      IF(YWRA.EQ.1)GO TO 333
      WRITE (NUE,1250)
      WRITE (NUE,124)
      WRITE (NUD,315)
      REWIND NUD
      WRITE (NUE,3154)
 3154 FORMAT(2X,5H*****,1X,'NOTHING TO APPEND TO WRA FILE')
      READ 310,WASTE
      WRITE (6,313)WASTE
      GO TO 106
C
C-----PREPARE TO LOAD YKA DATA
 1006 NUD=16
      NUE=26
      KL=6
      IF(YYKA.EQ.1)GO TO 333
      WRITE (NUE,2100)
      WRITE (NUE,124)
      WRITE (NUD,315)
      REWIND NUD
      WRITE (NUE,2101)
 2101 FORMAT(2X,5H*****,1X,'NOTHING TO APPEND TO YKA FILE')
      READ 310,WASTE
      WRITE (6,313)WASTE
      READ (5,310,END=107)WASTE
      WRITE (6,313)WASTE
      GO TO 107
C
 333  IF(KL.EQ.1)WRITE (NUE,120)
 120  FORMAT(/1X,'STATION WCL')
      IF(KL.EQ.2)WRITE(NUE,121)
 121  FORMAT(/1X,'STATION EKA')
      IF(KL.EQ.3)WRITE(NUE,122)
 122  FORMAT(/1X,'STATION GBA')
      IF(KL.EQ.5)WRITE (NUE,1250)
 1250 FORMAT(/1X,'STATION WRA')
      IF(KL.EQ.6)WRITE (NUE,2100)
 2100 FORMAT(/1X,'STATION YKA')
      WRITE(NUE,124)
 124  FORMAT(1X,'-----------')
C
      IF(KL.NE.5)GO TO 1200
C
C-----RE-ORDER WRA DATA - PUT CALIBRATION DATA AS TAGS
C-----COUNT NUMBER OF EVENTS.
      DO 1209 K=1,100
      TAG1(K)=0.0
      TAG2(K)=0.0
 1209 CONTINUE
C
 1203 J=1
      K=0
 1220 READ 1,(WRADAT(I,J),I=1,24)
      WRITE (6,111)(WRADAT(I,J),I=1,24)
 111  FORMAT(2X,24A1)
C-----CHECK FOR BLANK CARD IMAGE.
      DO 1120 I=1,24
      IF(WRADAT(I,J).NE.SB)GO TO 1121
 1120 CONTINUE
C-----HAVE A BLANK CARD - REJECT IT , AND MAKE NOTE IN ERRORS FILE.
      WRITE (NUE,1122)
 1122 FORMAT(2X,5H*****,'HAVE A BLANK CARD IMAGE WHICH HAS BEEN'/8X,
     1'REJECTED.')
      GO TO 1220
 1121 R1=WRADAT(1,J)
      R2=WRADAT(2,J)
      R3=WRADAT(3,J)
      IF(R1.EQ.AF.AND.R2.EQ.AI.AND.R3.EQ.AL)GO TO 1200
C-----INVESTIGATE EACH LINE OF DATA TO FIND OUT IF IT
C-----CONTAINS CALIBRATION DATA.
C-----THE VERY FIRST LINE OF DATA IS A SPECIAL
C-----CASE--IT MUST CONTAIN CALIBRATION INFORMATION.
C-----DEALING INITIALLY WITH THE FIRST LINE.
C-----CHECK THAT IT IS NOT A -NO DATA- OR -OUTAGE- ,
C-----  OR AN -ONSET TIME- .
      CW=WRADAT(3,J)
```

```
      DW=WRADAT(4,J)
      IF(J.GT.1)GO TO 1231
      IF(CW.EQ.AN)GO TO 1230
      IF(CW.EQ.AO.OR.DW.EQ.AO)GO TO 1230
C-----IF BLANK IS PRESENT IN A REASONABLE POSITION
C-----ASSUME LINE CONTAINS CALIBRATION DATA.
      DO 1201 I=3,6
      IF(WRADAT(I,J).EQ.SB)GO TO 1203
 1201 CONTINUE
 1230 WRITE (NUE,1202)(WRADAT(I,J),I=1,8)
 1202 FORMAT(2X,5H*****,8A1,
     1/18X,'THIS SHOULD BE CALIBRATION DATA--HENCE'
     1/18X,'THE WHOLE OF THE WRA INPUT DATA'
     1/18X,'HAS BEEN REJECTED,')
 1066 WRITE (NUO,315)
      REWIND NUO
 1666 R1=WRADAT(1,J)
      R2=WRADAT(2,J)
      R3=WRADAT(3,J)
      IF(R1.EQ.AF.AND.R2.EQ.AI.AND.R3.EQ.AL)GO TO 106
      J=J+1
      READ 1,(WRADAT(I,J),I=1,24)
      WRITE (6,111)(WRADAT(I,J),I=1,24)
      WRITE (NUE,1260)(WRADAT(I,J),I=1,24)
 1260 FORMAT(2X,24A1)
      GO TO 1666
C-----CHECK EACH LINE AFTER FIRST IN CASE NEW
C-----CALIBRATION DATA HAS BEEN LOADED.
 1231 IF(CW.EQ.AN)GO TO 1204
      IF(CW.EQ.AO.OR.DW.EQ.AO)GO TO 1204
      KNTOST=0
      DO 1232 I=1,8
      IF(WRADAT(I,J).EQ.SB)GO TO 1232
      KNTOST=KNTOST+1
 1232 CONTINUE
      IF(KNTOST.EQ.8)GO TO 1204
C-----ASSUME LEFT WITH CALIBRATION DATA.
      GO TO 1203
C
C-----DECODE CALIBRATION DATA FOR WRA.
C-----HTCAL = MEASURED HEIGHT OF CALIBRATION PULSE IN MMS.
C-----MICROV = CALIBRATION VOLTAGE IN MICROVOLTS.
C-----FOR DETAILS SEE BLACKNEST INTERNAL NOTE AG/157.
C
C-----HOW MANY CHARACTERS IN HTCAL
 1203 KNT1=0
      DO 604 I=1,24
      IF(WRADAT(I,J).EQ.SB)GO TO 605
      KNT1=KNT1+1
 604  CONTINUE
 605  IF(KNT1.GT.0.AND.KNT1.LE.5)GO TO 606
      WRITE (NUE,607)
 607  FORMAT(2X,5H*****,1X,'TOO MANY CHARS IN HTCAL'/
     18X,'WRA DATA NOT APPENDED TO FILE')
      GO TO 1066
C-----CHECK ARE THERE ANYMORE NON-BLANK CHARACTERS IN THE LINE
C-----I.E. IS MICROV PRESENT
 606  KCP1=KNT1+1
      KNT2=0
      DO 608 I=KCP1,24
      IF(WRADAT(I,J).EQ.SB)GO TO 608
      KNT2=KNT2+1
 608  CONTINUE
      IF(KNT2.NE.0)GO TO 612
C-----INSERT A VALUE FOR MICROV SINCE IT HAS NOT BEEN GIVEN
      NMV=264
      MICROV=FLOAT(NMV)
      GO TO 613
 612  IF(KNT2.GE.5.AND.KNT2.LE.5)GO TO 615
      WRITE (NUE,614)
 614  FORMAT(2X,5H*****,1X,'NUMBER OF CHARS IN MICROV SEEMS'/
     18X,'INCORRECT WRA DATA NOT APPENDED TO FILE')
      GO TO 1066
C-----DECODE TO GET MICROV
 615  KCP2=KCP1+1
      CALL CHARRV(WRADAT,J,NCHARN,KCP2,KNT2,RV,IV)
      IF(IV.NE.9999)GO TO 617
      MICROV=RV
      GO TO 613
 617  MICROV=FLOAT(IV)
      GO TO 613
C-----DECODE TO GET HTCAL
 613  CALL CHARRV(WRADAT,J,NCHARN,1,KNT1,RV,IV)
      IF(IV.NE.9999)GO TO 616
      HTCAL=RV
      GO TO 618
      GO TO 618
C-----CHECK THAT HTCAL AND MICROV ARE REASONABLE VALUES
 618  IF(HTCAL.GT.1.0.AND.HTCAL.LT.50.0)GO TO 621
      WRITE (NUE,622)
 622  FORMAT(2X,5H*****,1X,'HTCAL LT 1 OR GT 25'/
     18X,'WRA DATA NOT APPENDED TO FILE')
      GO TO 1066
 621  IF(MICROV.GT.100.0.AND.MICROV.LT.300.0)GO TO 1240
      WRITE (NUE,624)
 624  FORMAT(2X,5H*****,1X,'MICROV LT 100 OR GT 300'/
     18X,'WRA DATA NOT APPENDED TO FILE')
      GO TO 1066
C-----COUNT NUMBER OF EVENTS AND LOAD INTO ALPHAC.
 1204 K=K+1
      NWRAEV=K
      DO 1241 I=1,24
```

```
      ALPHAC(I,K)=WRADAT(I,J)
 1241 CONTINUE
      TAG1(K)=HTCAL
      TAG2(K)=MICRCV
 1240 J=J+1
      GO TO 1220
C
C-----DECODE A LINE OF INFORMATION.
 1200 J=1
 2    IF(KL.EQ.5)GO TO 1206
 1114 READ 1,(ALPHAC(I,J),I=1,24)
 1    FORMAT(24A1)
      WRITE (6,111)(ALPHAC(I,J),I=1,24)
C
C-----REJECT ANY COMPLETE BLANK CARD IMAGES.
      DO 1111 I=1,24
      IF(ALPHAC(I,J).NE.SB)GO TO 1112
 1111 CONTINUE
C-----HAVE A BLANK CARD IMAGE.
      WRITE (NUE,1113)
 1113 FORMAT(2X,5H*****,1X,'HAVE A BLANK CARD IMAGE'/8X,
     1'WHICH HAS BEEN REJECTED.')
      GO TO 1114
 1112 U1=ALPHAC(1,J)
      U2=ALPHAC(2,J)
      U3=ALPHAC(3,J)
      IF(U1.EQ.AF.AND.U2.EQ.AI.AND.U3.EQ.AL)GO TO 130
      GO TO 601
C
C-----PARAMETER NPRT IS CONCERNED WITH PRINTING OF WRA AMP. DATA.
C-----WHEN NPRT=0 ORIGINAL AMPLITUDES TO STREAM NUE.
C-----WHEN NPRT=1 MODIFIED AMPLITUDES TO STREAM NUD.
C-----RE-SETTING OF NPRT IS DONE WITHIN THE PROGRAM.
 1206 NPRT=0
      MSKIP=0
C
C-----CHECK THAT FIRST TWO CHARACTERS ARE NUMERIC.
C-----IF SO - DECODE A DAY ,IF NOT - REJECT THE DATA-LINE.
C
      IF(J.GT.NWRAEV)GO TO 130
 601  C1=ALPHAC(1,J)
      C2=ALPHAC(2,J)
      IF(C1.EQ.N0.OR.C1.EQ.N1.OR.C1.EQ.N2.OR.C1.EQ.N3.OR.C1.EQ.N4.
     1OR.C1.EQ.N5.OR.C1.EQ.N6.OR.C1.EQ.N7.OR.C1.EQ.N8.OR.C1.EQ.N9)
     1GO TO 903
      GO TO 902
 903  IF(C2.EQ.N0.OR.C2.EQ.N1.OR.C2.EQ.N2.OR.C2.EQ.N3.OR.C2.EQ.N4.
     1OR.C2.EQ.N5.OR.C2.EQ.N6.OR.C2.EQ.N7.OR.C2.EQ.N8.OR.C2.EQ.N9)
     1GO TO 97
 902  WRITE (NUE,96)(ALPHAC(I,J),I=1,8)
 96   FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
     1'-----FIRST CHARACTER BLANK DATA-LINE REJECTED.')
      GO TO 6
 97   CALL CHARRV(ALPHAC,J,NCHARN,1,2,RV,JDAY)
C-----CHECK DAY VALUE.
      IF(JDAY.GT.0.AND.JDAY.LT.32)GO TO 43
      WRITE (NUE,44)(ALPHAC(I,J),I=1,8)
 44   FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
     1'-----DAY ERROR-LINE REJECTED')
      GO TO 6
C
C
C     TEST THE CHARACTERS IN A DATA-LINE FOR THE PRESENCE OF THE
C     CORRECT BLANKS.IF ANY OF THE CHARACTERS ARE ALPHABETIC-PASS.
C     IF AFTER THE 8.TH CHARACTER THEY ARE NUMERIC OR DECIMAL POINTS
C     THEN THERE IS A MAXIMUM NUMBER 12,TH BY WHICH A BLANK
C     SHOULD OCCUR.-IF IT DOES NOT-REJECT DATA-LINE-IF THERE IS A
C     CORRECT BLANK , THEN EITHER (A) ALL REST OF CHARACTERS FOR
C     THE LINE SHOULD BE BLANK OR (B) THERE SHOULD BE A NUMBER
C     NOT EXCEEDING 6 CHARACTERS - A SECOND BLANK - AND THEN
C     ANOTHER NUMBER NOT EXCEEDING 6 CHARACTERS .
C     IF (A) OR (B) IS TRUE - PASS . IF NOT REJECT DATA LINE.
C
 43   DO 810 IC=3,8
      CH=ALPHAC(IC,J)
      DO 557 I=1,10
      IF(CH.EQ.NCHARN(I))GO TO 810
 557  CONTINUE
      GO TO 888
 810  CONTINUE
C
C-----CHECK IF ALL REST OF CHARACTERS ARE BLANK.
      DO 823 I=9,24
      IF(ALPHAC(I,J).NE.SBLANK)GO TO 824
 823  CONTINUE
      GO TO 888
C-----CHECK FOR BLANK BETWEEN 9,TH AND 12,TH POSITIONS.
 824  IF(ALPHAC(9,J).NE.SB)GO TO 811
      IA=9
      GO TO 812
 811  IF(ALPHAC(10,J).NE.SB)GO TO 814
      IA=10
      GO TO 812
 814  IF(ALPHAC(11,J).NE.SB)GO TO 815
      IA=11
      GO TO 812
 815  IA=12
      IF(ALPHAC(12,J).EQ.SB)GO TO 812
      WRITE (NUE,813)(ALPHAC(I,J),I=1,8)
 813  FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,2X,
     1'-NO BLANK BETWEEN TIME/AMP -LINE REJECTED')
      GO TO 6
```

```
C-----FOUND FIRST BLANK-ANYMORE
 812  IL=IA+1
      NX=0
      NK=0
      DO 817 IG=IL,24
      CH=ALPHAC(IG,J)
      NX=IG
      IF(CH.NE.SB)NK=NK+1
      IF(CH.EQ.SB)GO TO 820
      IF(NK.LE.6)GO TO 817
      WRITE (NUE,8190)(ALPHAC(I,J),I=1,8)
 8190 FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,2X,
     1'-EXCESS CHARS IN AMP. OR PERIOD-LINE REJECTED')
      GO TO 6
 817  CONTINUE
C
C-----FOUND ANOTHER BLANK-ARE THERE ANYMORE NON-BLANK CHARACTERS
C-----IF NOT-ERROR
 820  IM=NX+1
      NQ=0
      DO 818 IH=IM,24
      CH=ALPHAC(IH,J)
      IF(CH.NE.SB)NQ=NQ+1
      IF(CH.EQ.SB)GO TO 889
      IF(NQ.LE.6)GO TO 818
      WRITE (NUE,8220)(ALPHAC(I,J),I=1,8)
 8220 FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,2X,
     1'-EXCESS CHARS IN AMP. OR PERIOD-LINE REJECTED')
      GO TO 6
 818  CONTINUE
C
C-----SINCE EITHER AMP/PERIOD IS PRESENT CHECK THAT
C-----PERIOD/AMP IS ALSO PRESENT . IF NOT REJECT DATA-LINE.
 889  IF(NK.EQ.0)GO TO 888
      IF(NK.NE.0.AND.NQ.EQ.0)GO TO 890
      GO TO 888
 890  WRITE (NUE,891)(ALPHAC(I,J),I=1,8)
 891  FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,2X,
     1'--EITHER AMP OR PERIOD IS MISSING LINE REJECTED')
      GO TO 6
C-----CHECK THAT THE DAYS LOADED ARE SEQUENTIAL.
 888  IF(J.EQ.1)GO TO 51
      IF((JDAY-PREVDY).LT.2)GO TO 51
      WRITE (NUE,52)(ALPHAC(I,J-1),I=1,8)
 52   FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
     1'-----DAYS ARE NOT SEQUENTIAL-PLEASE CHECK')
      GO TO 51
C
 51   DO 30 I=1,10
      IF(ALPHAC(3,J).EQ.NCHARN(I))GO TO 4
 30   CONTINUE
      IF(ALPHAC(3,J).EQ.AN.OR.ALPHAC(3,J).EQ.AO)GO TO 5
C
      WRITE (NUE,7)
 7    FORMAT(8X,'CANT RECOGNIZE CHARACTER')
      GO TO 6
C
C-----SORT OUT WHETHER DATA LOSS IS 'NO DATA' OR 'OUTAGE'.
 5    IF(ALPHAC(3,J).EQ.AN)GO TO 31
      IF(ALPHAC(3,J).EQ.AO)GO TO 18
C
C-----LOAD CHARACTERS 'NO DATA' WITH DATE INTO USUAL SLOTS IN ALPHAC.
 31   ALPHAC(3,J)=SBLANK
      ALPHAC(4,J)=AN
      ALPHAC(5,J)=AO
      ALPHAC(6,J)=SBLANK
      ALPHAC(7,J)=AD
      ALPHAC(8,J)=AA
      ALPHAC(9,J)=AT
      ALPHAC(10,J)=AA
C
      WRITE (NUD,19)(ALPHAC(I,J),I=1,10)
 19   FORMAT(8X,10A1)
      GO TO 6
C
C ----DATA LOSS---OUTAGE
C-----HOW MUCH INFORMATION HAS BEEN GIVEN ABOUT OUTAGE
C     FULL OR ABBREVIATED .
 18   IF(ALPHAC(3,J).EQ.AO.AND.ALPHAC(4,J).EQ.AU)GO TO 24
      GO TO 32
C
C-----IS IT OUTAGE A OR B .
 24   IF(ALPHAC(9,J).EQ.AA.OR.ALPHAC(10,J).EQ.AA.OR.ALPHAC(11,J).EQ.AA)
     1GO TO 33
C
C-----CHECK THAT IT,S OUTAGE B.
      IF(ALPHAC(9,J).EQ.AB.OR.ALPHAC(10,J).EQ.AB.OR.ALPHAC(11,J).EQ.AB)
     1GO TO 33
      WRITE (NUD,35)
 35   FORMAT(2X,5H*****,1X,2A1,1X,7A1,
     1'-----NOT ABLE TO IDENTIFY OUTAGE')
      GO TO 6
C
C-----ABBREVIATED INPUT---WHAT FORM OF OUTAGE IS IT , A OR B  .
 32   IF(ALPHAC(4,J).EQ.AB)GO TO 3
      IF(ALPHAC(4,J).EQ.AA)GO TO 11
      WRITE (NUD,14)(ALPHAC(I,J),I=1,9)
 14   FORMAT(2X,5H*****,1X,2A1,1X,7A1,
     1'-----NOT ABLE TO IDENTIFY OUTAGE')
      GO TO 6
C
C-----FORM OF OUTAGE = A.
```

```
 11    WRITE (NUO,17)(ALPHAC(I,J),I=1,2),(ALPHAC(I,J),I=5,13)
 17    FORMAT(8X,2A1,1X,'OUTAGE A',1X,1H(,9A1,1H))
       GO TO 6
C
C-----FORM OF OUTAGE = B .
 3     WRITE (NUO,28)(ALPHAC(I,J),I=1,2),(ALPHAC(I,J),I=5,13)
 28    FORMAT(8X,2A1,1X,'OUTAGE B',1X,1H(,9A1,1H))
       GO TO 6
C
 33    WRITE (NUO,36)(ALPHAC(I,J),I=1,20)
 36    FORMAT(8X,2A1,1X,6A1,1X,1A1,1X,11A1)
       GO TO 6
C
C-----CHECK THAT THERE ARE EXACTLY 8 CHARACTERS IN THE ONSET TIME,
C-----THE 9,TH CHARACTER BEING EITHER A DECIMAL POINT OR A BLANK.
 4     IF(ALPHAC(8,J).NE.SBLANK)GO TO 77
       WRITE(NUE,78)(ALPHAC(I,J),I=1,8)
 78    FORMAT(2X,5H*****,1X,2A1,1X,6A1,
      1'-----LESS THAN EIGHT CHARACTERS IN ONSET TIME'
      1,50X,'LINE REJECTED')
       GO TO 6
 77    IF(ALPHAC(9,J).EQ.DECIPT.OR.ALPHAC(9,J).EQ.SBLANK)GO TO 79
       WRITE (NUE,80)(ALPHAC(I,J),I=1,9)
 80    FORMAT(2X,5H*****,1X,2A1,1X,7A1,
      1'-----MORE THAN EIGHT CHARACTERS IN ONSET TIME'
      1,50X,'-LINE REJECTED')
       GO TO 6
C
C-----ONSET TIME---PROCEED TO DECODE
C      DECODE HOURS , MINUTES ,SECONDS .
 79    CALL CHARRV(ALPHAC,J,NCHARN,3,2,RV,NHR)
C-----CHECK HOURS VALUE.
       IF(NHR.LE.23)GO TO 45
       WRITE (NUE,46)(ALPHAC(I,J),I=1,8)
 46    FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
      1'-----HOURS ERROR-LINE REJECTED')
       GO TO 6
C
C-----CHECK THAT THE HOURS (FOR A GIVEN DAY) FOR TWO CONSECUTIVE ENTRIES
C-----ARE SEQUENTIAL
 45    IF(J.EQ.1)GO TO 140
       IF((JDAY-PREVDY).NE.0)GO TO 140
       IF((NHR-PREVHR).GE.0)GO TO 140
C
       WRITE(NUE,142)(ALPHAC(I,J-1),I=1,8)
 142   FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
      1'-----HOURS NOT SEQUENTIAL-PLEASE CHECK')
       GO TO 140
C
 140   CALL CHARRV(ALPHAC,J,NCHARN,5,2,RV,MINS)
C-----CHECK MINUTES VALUE.
       IF(MINS.LE.59)GO TO 47
       WRITE (NUE,48)(ALPHAC(I,J),I=1,8)
 48    FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
      1'-----MINUTES ERROR-LINE REJECTED')
       GO TO 6
C-----CHECK THAT THE MINUTES (FOR A GIVEN DAY AND HOUR)
C-----ARE SEQUENTIAL
 47    IF(J.EQ.1)GO TO 145
       IF((JDAY-PREVDY).NE.0)GO TO 145
       IF((NHR-PREVHR).NE.0)GO TO 145
       IF((MINS-PREVMN).GE.0)GO TO 145
C
       WRITE(NUE,147)(ALPHAC(I,J-1),I=1,8)
 147   FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
      1'-----MINUTES NOT SEQUENTIAL-PLEASE CHECK')
       GO TO 145
C
C      SEARCH FOR FIRST BLANK IN ORDER TO DEAL WITH THE SECONDS
C
 145   DO 8 I=1,24
       IF(ALPHAC(I,J).EQ.SBLANK)GO TO 9
 8     CONTINUE
       GO TO 6
 9     ISTORE=I
       IM7=ISTORE-7
       CALL CHARRV(ALPHAC,J,NCHARN,7,IM7,SECS,ISECS)
       GO TO 119
C-----CHECK SECONDS VALUE
 119   IF(ISECS.NE.9999)GO TO 49
 131   IF(SECS.LE.59.99)GO TO 350
       WRITE (NUE,59)(ALPHAC(I,J),I=1,10)
 59    FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,4A1,
      1'-----SECONDS ERROR-LINE REJECTED')
       GO TO 6
 49    IF(ISECS.LE.59)GO TO 350
       WRITE (NUE,67)(ALPHAC(I,J),I=1,8)
 67    FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
      1'-----SECONDS ERROR-LINE REJECTED')
       GO TO 6
C
C      FIND L THE NUMBER OF CHARACTERS BETWEEN
C      TWO BLANKS .
C
 350   IP1=ISTORE+1
       L=0
       DO 12 I=IP1,24
       IF(ALPHAC(I,J).EQ.SBLANK)GO TO 13
       L=L+1
 12    CONTINUE
       GO TO 22
 13    LAMP=L
```

```
      IF(LAMP.EQ.0)GO TO 22
C
C-----DECODE AMPLITUDE.
      CALL CHARRV(ALPHAC,J,NCHARN,IP1,L,AMP,IAMP)
C
C-----FIND NUMBER OF CHARACTERS IN PERIOD VALUE.
      L=0
      ISTART=IP1+LAMP+1
      DO 40 I=ISTART,24
      IF(ALPHAC(I,J).EQ.SBLANK)GO TO 41
      L=L+1
 40   CONTINUE
      GO TO 6
C
C-----DECODE PERIOD.
 41   CALL CHARRV(ALPHAC,J,NCHARN,ISTART,L,PERIOD,IPERCD)
      GO TO 699
C
C-----CHECK AMPLITUDE VALUE.
 699  IF(KL.EQ.5)GO TO 625
      IF(IAMP.NE.9999)GO TO 68
 626  IF(AMP.LT.10000.0)GO TO 69
 72   WRITE (NUE,70)(ALPHAC(I,J),I=1,8)
 70   FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
     1'-----AMPLITUDE ERROR-LINE REJECTED')
      GO TO 6
 68   IF(IAMP.LT.10000)GO TO 69
      GO TO 72
C-----CONVERT WRA AMPLITUDE (MMS TO NMS).
 625  IF(IAMP.NE.9999)GO TO 627
      GO TO 628
 627  AMP=FLOAT(IAMP)
      IAMP=9999
C-----MUST (J=1) COMPUTE MAGNIFICATION READY FOR
C-----CONVERTING AMPLITUDE VALUES , WHEN (J=2,3,4,---)
C-----CHECK TAG VALUES IF EITHER CHANGES RE-COMPUTE
C-----MAGNIFICATION FACTOR.
 628  IF(J.EQ.1)GO TO 1210
      IF(AMP.NE.0.0)GO TO 1213
      GO TO 22
 1213 IF(TAG1(J).EQ.TAG1(J-1).AND.TAG2(J).EQ.TAG2(J-1))GO TO 1212
      HTCAL=TAG1(J)
      MICROV=TAG2(J)
      GO TO 1216
 1210 HTCAL=TAG1(1)
      MICROV=TAG2(1)
C     PUT VALUES OF HTCAL AND MICRO IN ERRORS FILE (IF ONE OR
C     OTHER OR BOTH HAVE CHANGED).
 1216 WRITE (NUE,620)HTCAL,MICROV
 620  FORMAT(2X,'HTCAL = ',F7.2,5X,'MICROV = ',F7.2)
C-----OBTAIN MAGNIFICATION (AT 1 HZ) FACTOR FOR WRA.
      MAGNIF=(1240.0*HTCAL)/MICROV
C-----OBTAIN RELATIVE MAGNIFICATION AT GIVEN PERIOD.
C-----BUT FIRST CHECK THAT VALUE FOR WHICH INTERPOLATION
C-----IS REQUESTED IS WITHIN AVAILABLE RANGE.
C-----IF NOT REJECT LINE OF DATA WITH APPROPIATE COMMENT
C-----ALLOWED PERIOD RANGE 0.2-2.5 SECS.
 1212 IF(PERIOD.GE.0.2.AND.PERIOD.LE.2.5)GO TO 1214
      IF(PERIOD.EQ.0.0)GO TO 22
      WRITE (NUE,1215)(ALPHAC(I,J),I=1,24)
 1215 FORMAT(2X,5H*****,1X,24A1/8X,
     1'PERIOD VALUE OUTSIDE INTERPOLATION RANGE'/8X,
     1'OF 0.2-2.5 SECS HENCE LINE REJECTED.')
      GO TO 6
 1214 CALL INTERP(PEROD,RELMAG,32,PERIOD,RELMGN)
C
C-----CHECK PERIOD VALUE.
 69   IF(IPEROD.NE.9999)GO TO 73
      IF(PERIOD.EQ.0.0)GO TO 22
      IF(PERIOD.LT.3.0)GO TO 71
 76   WRITE (NUE,75)(ALPHAC(I,J),I=1,8)
 75   FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
     1'-----PERIOD ERROR-LINE REJECTED')
      GO TO 6
 73   IF(IPEROD.EQ.0)GO TO 22
      IF(IPEROD.LT.3)GO TO 71
      GO TO 76
C
C-----CHOOSE OUTPUT FORMAT.
C
C-----------------------------------------------------------------
C     FOR WRA DATA SINCE ORIGINAL AMPLITUDE DATA IS CONVERTED , FOR
C     CHECKING PURPOSES IT IS NECESSARY BEFORE DOING AMPLITUDE
C     CONVERSION TO PRINT ORIGINAL DATA OUT INTO THE ERRORS FILE.
C
C-----------------------------------------------------------------
 71   IF(KL.NE.5)GO TO 872
      NUX=NUE
      GO TO 871
 870  NUX=NUD
C-----OBTAIN WRA AMPLITUDE IN NANOMETRES ,SEE BLACKNEST NOTE AG/157
C-----APPENDIX XXIX , AND SUPPLEMENT TO APPENDIX XXX.
      AMP=((AMP/2.0)*1000.0)/(MAGNIF*RELMGN)
      GO TO 871
 872  NUX=NUD
C
 871  IF(ISECS.NE.9999)GO TO 260
      IF(IAMP.NE.9999)GO TO 261
      IF(IPEROD.NE.9999)GO TO 262
C
      IF(ALPHAC(10,J).NE.SBLANK.AND.ALPHAC(11,J).NE.SBLANK)GO TO 201
      WRITE (NUX,250)(ALPHAC(I,J),I=1,10),AMP,PERIOD
```

```
 250  FORMAT(8X,2A1,1X,4A1,1X,4A1,16X,F7.1,6X,F4.1)
      GO TO 666
 201  WRITE (NUX,270)(ALPHAC(I,J),I=1,11),AMP,PERIOD
 270  FORMAT(8X,2A1,1X,4A1,1X,5A1,15X,F7.1,6X,F4.1)
      GO TO 666
C
 262  IF(ALPHAC(10,J).NE.SBLANK.AND.ALPHAC(11,J).NE.SBLANK)GO TO 202
      WRITE (NUX,251)(ALPHAC(I,J),I=1,10),AMP,IPEROD
 251  FORMAT(8X,2A1,1X,4A1,1X,4A1,16X,F7.1,5X,I3)
      GO TO 666
 202  WRITE (NUX,271)(ALPHAC(I,J),I=1,11),AMP,IPEROD
 271  FORMAT(8X,2A1,1X,4A1,1X,5A1,15X,F7.1,5X,I3)
      GO TO 666
C
 261  IF(IPEROD.NE.9999)GO TO 263
      IF(ALPHAC(10,J).NE.SBLANK.AND.ALPHAC(11,J).NE.SBLANK)GO TO 203
      WRITE (NUX,252)(ALPHAC(I,J),I=1,10),IAMP,PERIOD
 252  FORMAT(8X,2A1,1X,4A1,1X,4A1,15X,I6,8X,F4.1)
      GO TO 666
 203  WRITE (NUX,272)(ALPHAC(I,J),I=1,11),IAMP,PERIOD
 272  FORMAT(8X,2A1,1X,4A1,1X,5A1,14X,I6,8X,F4.1)
      GO TO 666
C
 263  IF(ALPHAC(10,J).NE.SBLANK.AND.ALPHAC(11,J).NE.SBLANK)GO TO 204
      WRITE (NUX,253)(ALPHAC(I,J),I=1,10),IAMP,IPEROD
 253  FORMAT(8X,2A1,1X,4A1,1X,4A1,15X,I6,7X,I3)
      GO TO 666
 204  WRITE (NUX,273)(ALPHAC(I,J),I=1,11),IAMP,IPEROD
 273  FORMAT(8X,2A1,1X,4A1,1X,5A1,14X,I6,7X,I3)
      GO TO 666
C
C
 260  IF(IAMP.NE.9999)GO TO 264
      IF(IPEROD.NE.9999)GO TO 265
      WRITE (NUX,254)(ALPHAC(I,J),I=1,8),AMP,PERIOD
 254  FORMAT(8X,2A1,1X,4A1,1X,2A1,16X,F7.1,6X,F4.1)
      GO TO 666
 265  WRITE (NUX,255)(ALPHAC(I,J),I=1,8),AMP,IPEROD
 255  FORMAT(8X,2A1,1X,4A1,1X,2A1,18X,F7.1,5X,I3)
      GO TO 666
 264  IF(IPEROD.NE.9999)GO TO 266
      WRITE (NUX,256)(ALPHAC(I,J),I=1,8),IAMP,PERIOD
 256  FORMAT(8X,2A1,1X,4A1,1X,2A1,17X,I6,8X,F4.1)
      GO TO 666
 266  WRITE (NUX,257)(ALPHAC(I,J),I=1,8),IAMP,IPEROD
 257  FORMAT(8X,2A1,1X,4A1,1X,2A1,17X,I6,7X,I3)
      GO TO 666
C
C
 22   IF(KL.NE.5)GO TO 915
      MSKIP=MSKIP+1
      IF(NPRT.EQ.0)NUX=NUE
      IF(NPRT.EQ.1)NUX=NUD
      GO TO 916
 915  NUX=NUD
 916  IF(ISECS.NE.9999)GO TO 242
      WRITE (NUX,239)(ALPHAC(I,J),I=1,11)
 239  FORMAT(8X,2A1,1X,4A1,1X,5A1)
      GO TO 666
 242  WRITE (NUX,238)(ALPHAC(I,J),I=1,8)
 238  FORMAT(8X,2A1,1X,4A1,1X,2A1)
      GO TO 666
C
C-----CHECK THAT THE SECONDS FOR THE CONSECUTIVE
C-----ENTRIES(DAY,HR,MIN-SAME)ARE SEQUENTIAL.
C
 666  IF(KL.NE.5)GO TO 873
      NPRT=NPRT+1
      IF((NPRT.EQ.1).AND.(MSKIP.EQ.1))GO TO 22
      IF(NPRT.EQ.1)GO TO 870
 873  IF(J.EQ.1)GO TO 6
      IF((JDAY-PREVDY).NE.0)GO TO 6
      IF((NHR-PREVHR).NE.0)GO TO 6
      IF((MINS-PREVMN).NE.0)GO TO 6
      IF(ISECS.NE.9999)GO TO 667
      IF((SECS-PRVSEC).GE.0.0)GO TO 6
 669  WRITE (NUE,668)(ALPHAC(I,J),I=1,8)
 668  FORMAT(2X,5H*****,1X,2A1,1X,4A1,1X,2A1,
     1'-----SECONDS NOT SEQUENTIAL-PLEASE CHECK')
      GO TO 6
 667  IF((ISECS-PREVSC).GE.0)GO TO 6
      GO TO 669
C
 6    J=J+1
      PREVDY=JDAY
      PREVHR=NHR
      PREVMN=MINS
      PREVSC=ISECS
      PRVSEC=SECS
      IF(J.GT.200)GO TO 20
      GO TO 2
C
 130  GO TO (1020,1030,1040,1040,1060,1070),KL
 1020 WRITE (NUE,1021)
 1021 FORMAT(2X,'DATA(WOL) TO FILE COMPLETED')
      GO TO 102
 1030 WRITE (NUE,1022)
 1022 FORMAT(2X,'DATA(EKA) TO FILE COMPLETED')
      GO TO 103
 1040 WRITE (NUE,1023)
 1023 FORMAT(2X,'DATA(GBA) TO FILE COMPLETED')
      GO TO 104
```

```
 1060 WRITE (NUE,1025)
 1025 FORMAT(2X,'DATA(WRA) TO FILE COMPLETED')
      GO TO 106
 1070 WRITE (NUE,1026)
 1026 FORMAT(2X,'DATA(YKA) TO FILE COMPLETED')
      GO TO 107
 102  CALL CLFILE(11)
      CALL CLFILE(21)
      IF(KL.EQ.1)GO TO 1002
      GO TO 20
 103  CALL CLFILE(12)
      CALL CLFILE(22)
      IF(KL.EQ.2)GO TO 1003
      GO TO 20
 104  CALL CLFILE(13)
      CALL CLFILE(23)
      IF(KL.EQ.3)GO TO 1005
      GO TO 20
 106  CALL CLFILE(15)
      CALL CLFILE(25)
      IF(KL.EQ.5)GO TO 1006
      GO TO 20
 107  CALL CLFILE(16)
      CALL CLFILE(26)
C
C
 20   RETURN
      END

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

C
C*************************************************************
C
C                    SUBROUTINE PROGRAM CHARRV
C                    *************************
C THIS IS A FORTRAN 4 PROGRAM USED TO CONVERT BLOCKS OF ALPHA
C CHARACTERS INTO NUMBERS.
C
C*************************************************************
C
C
      SUBROUTINE CHARRV(ALPHAC,M,NCHARN,J,JTOT,RV,IV)
C
      DIMENSION ALPHAC(24,100),NCHARN(10),ICHAR(50)
C
      DATA COMMA/4H,   /,DECIPT/4H.   /,SBLANK/4H    /
C
      INTEGER ALPHAC,COMMA,DECIPT,SBLANK
C
      REAL*8 RV,FRACT
C
      RV=0.0
      IV=9999
C
C-----IDENTIFY CHARACTERS UP TO JTOT OR DECIMAL POINT.
C-----(WHICHEVER IS FIRST).
      IPART=0
      DO 5 L=1,JTOT
      JPLM1=J+L-1
      IF(ALPHAC(JPLM1,M).EQ.DECIPT)GO TO 2
C-----IDENTIFY CHARACTER.
      DO 1 I=1,10
      IF(ALPHAC(JPLM1,M).EQ.NCHARN(I))ICHAR(L)=I-1
 1    CONTINUE
 5    CONTINUE
      L=JTOT
C-----NO DECIMAL POINT INVOLVED.
      DO 20 I=1,JTOT
      K=JTOT-I
      IPART=IPART+10**K*ICHAR(I)
 20   CONTINUE
      GO TO 21
C
 2    LM1=L-1
      DO 6 I=1,LM1
      K=LM1-I
      IPART=IPART+10**K*ICHAR(I)
 6    CONTINUE
C
      IF(ALPHAC(JPLM1,M).EQ.DECIPT)GO TO 8
C
C-----INTEGER NUMBER.
 21   IV=IPART
      RETURN
C
C-----DECIMAL POINT BUT NOTHING AFTER IT.
 8    JPL=J+L
      IF(ALPHAC(JPL,M).NE.SBLANK.OR.ALPHAC(JPL,M).NE.COMMA)GO TO 3
      RV=FLOAT(IPART)
      RETURN
C
C-----REAL NUMBER.
 3    ICHAR(L)=DECIPT
      LDEC=L
      FRACT=0.0
      JPLEND=J+JTOT-1
      DO 4 K=JPL,JPLEND
      L=L+1
C-----IDENTIFY REMAINING CHARACTERS.
      DO 7 I=1,10
      IF(ALPHAC(K,M).EQ.NCHARN(I))ICHAR(L)=I-1
```

```
 7    CONTINUE
 4    CONTINUE
C
      L=LDEC
      DO 9 I=JPL,JPLEND
      L=L+1
      LP=L-LDEC
      FRACT=FRACT+0.1**LP*FLOAT(ICHAR(L))
 9    CONTINUE
C
      RV=FLOAT(IPART)+FRACT
C
      RETURN
      END

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

C
C*****************************************************************
C
C                    SUBROUTINE PROGRAM INTERP
C                    *************************
C THIS IS A FORTRAN 4 PROGRAM USED TO PERFORM LINEAR INTERPOLATION.
C
C*****************************************************************
C
C
      SUBROUTINE INTERP(X,Y,N,XBAR,YBAR)
C-----LINEARLY INTERPOLATES . (FOR POSITIVE X ONLY).
      DIMENSION X(N),Y(N)
C
      REAL*8 X,Y,XBAR,YBAR
C
      K=0
      IF(X(1).GT.XBAR)GO TO 1
C-----X(I) INCREASING IN VALUE.
      DO 2 I=2,N
      IF(X(I).GT.XBAR)GO TO 3
 2    CONTINUE
C
      K=1
      GO TO 99
 3    IM1=I-1
      SLOPE=(Y(I)-Y(IM1))/(X(I)-X(IM1))
      YBAR=Y(IM1)+SLOPE*(XBAR-X(IM1))
      IF(K.EQ.1)GO TO 99
      RETURN
C
C-----X(I) DECREASING IN VALUE.
 1    DO 4 I=1,N
      IF(X(I).LT.XBAR)GO TO 5
 4    CONTINUE
      K=1
      GO TO 99
 5    IM1=I-1
      SLOPE=(Y(I)-Y(IM1))/(X(IM1)-X(I))
      YBAR=Y(IM1)+SLOPE*(XBAR-X(IM1))
      IF(K.EQ.0)RETURN
 99   PRINT 6
 6    FORMAT(10X,21HLOOK AT INTERPOLATION)
      PRINT 8,XBAR
 8    FORMAT(10X,5HXBAR=,F10.5)
      PRINT 7,(X(I),Y(I),I=1,N)
 7    FORMAT(6F12.3)
      RETURN
      END

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

C
C*****************************************************************
C
C                              PROGRAM MAIN(2)
C                              ***************
C     THIS IS A FORTRAN 4 PROGRAM WHICH CALLS TWO SUBROUTINES (MONTOR
C     AND SCANFL). THE FUNCTION OF THESE SUBROUTINES IS TO CHECK THE
C     BLACKNEST DATABASE FILES FOR  MISSING DAYS . ONLY THOSE FILES TO
C     WHICH DATA HAS BEEN APPENDED AT A PARTICULAR LOADING ARE SCANNED.
C
C*****************************************************************
C
C
      DIMENSION DAY(1000),STNIN(9)
C
      INTEGER DAY
C
      REAL*8  STNIN
      REAL*8  STNWOL/8HWOL     /,STNEKA/8HEKA      /,STNGBA/8HGBA      /
      REAL*8  STNWRA/8HWRA     /,STNYKA/8HYKA      /
C
C
      IK=1
 3    READ (30,1,END=200)STNIN(IK)
 1    FORMAT(A8)
      WRITE (6,2)STNIN(IK)
 2    FORMAT(1X,A8)
      IK=IK+1
      GO TO 3
C
 200  NWOL=0
      NEKA=0
```

```
      NGBA=0
      NWRA=0
      NYKA=0
C
      DO 5 I=1,5
      IF(STNIN(I).EQ.STNWOL)NWOL=1
      IF(STNIN(I).EQ.STNEKA)NEKA=1
      IF(STNIN(I).EQ.STNGBA)NGBA=1
      IF(STNIN(I).EQ.STNWRA)NWRA=1
      IF(STNIN(I).EQ.STNYKA)NYKA=1
 5    CONTINUE
C
      NUE=41
      IF(NWOL.EQ.0)GO TO 3001
      NSTR=31
      CALL MONTOR(NSTR,DAY,NUMDAY)
      CALL SCANFL(DAY,NUMDAY,NUE,IFLAG)
      IF(IFLAG.EQ.1)GO TO 20
      GO TO 300
3001  WRITE (NUE,315)
315   FORMAT(10X)
300   REWIND NUE
20    CALL CLFILE(NUE)
      NUE=42
      IF(NEKA.EQ.0)GO TO 3002
      NSTR=32
      CALL MONTOR(NSTR,DAY,NUMDAY)
      CALL SCANFL(DAY,NUMDAY,NUE,IFLAG)
      IF(IFLAG.EQ.1)GO TO 21
      GO TO 301
3002  WRITE (NUE,315)
301   REWIND NUE
21    CALL CLFILE(NUE)
      NUE=43
      IF(NGBA.EQ.0)GO TO 3003
      NSTR=33
      CALL MONTOR(NSTR,DAY,NUMDAY)
      CALL SCANFL(DAY,NUMDAY,NUE,IFLAG)
      IF(IFLAG.EQ.1)GO TO 22
      GO TO 302
3003  WRITE (NUE,315)
302   REWIND NUE
22    CALL CLFILE(NUE)
      NUE=44
      IF(NWRA.EQ.0)GO TO 3004
      NSTR=34
      CALL MONTOR(NSTR,DAY,NUMDAY)
      CALL SCANFL(DAY,NUMDAY,NUE,IFLAG)
      IF(IFLAG.EQ.1)GO TO 23
      GO TO 303
3004  WRITE (NUE,315)
303   REWIND NUE
23    CALL CLFILE(NUE)
      NUE=45
      IF(NYKA.EQ.0)GO TO 3005
      NSTR=35
      CALL MONTOR(NSTR,DAY,NUMDAY)
      CALL SCANFL(DAY,NUMDAY,NUE,IFLAG)
      IF(IFLAG.EQ.1)GO TO 24
      GO TO 304
3005  WRITE (NUE,315)
304   REWIND NUE
24    CALL CLFILE(NUE)
C
      RETURN
      END

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

C
C*********************************************************************
C
C                             SUBROUTINE PROGRAM MONTOR
C                             *************************
C     THIS IS A FORTRAN 4 PROGRAM WHICH IS USED TO READ INTO AN ARRAY
C     THE DAYS FROM A BLACKNEST DATA BASE FILE IN PREPARATION FOR
C     SCANNING THE FILE FOR MISSING DAYS.
C
C**********************************************R**********************
C
C
      SUBROUTINE MONTOR(NSTR,DAY,NUMDAY)
C
      DIMENSION DAY(1)
C
      INTEGER DAY,PRTDAY
C
      PRTDAY=0
C
      I=1
1     READ (NSTR,2,END=200)DAY(I)
2     FORMAT(8X,I2)
      IF(PRTDAY.EQ.1)WRITE (6,2)DAY(I)
      I=I+1
      GO TO 1
C
200   NUMDAY=I
C
      RETURN
      END
```

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
C
C*****************************************************************
C
C                              SUBROUTINE PROGRAM SCANFL
C                              *************************
C     THIS IS A FORTRAN 4 PROGRAM WHICH IS USED TO CHECK THE FILE OF
C     DATA FOR ANY  MISSING DAYS .
C
C*****************************************************************
C
C
      SUBROUTINE SCANFL(DAY,NUMDAY,NUE,IFLAG)
C
      DIMENSION DAY(1)
C
      INTEGER DAY,DAYDIF
C
C-----NO ERROR DETECTED THEN IFLAG=0 , IF ERROR THEN IFLAG=1.
      IFLAG=0
C-----CHECK IF FILE CONTAINS ANY DATA
      IF(NUMDAY.EQ.0)GO TO 9
C-----IF FILE CONTAINS ONLY 1 VALUE IT MUST BE FIRST DAY OF MONTH.
      IF(NUMDAY.GT.1)GO TO 2
      IF(DAY(1).EQ.01)GO TO 9
      WRITE (6,6)
      WRITE (NUE,6)
 6    FORMAT(1X,'*****ONLY VALUE IN FILE IS NOT FIRST DAY OF MONTH ')
      IFLAG=1
      GO TO 99
C-----CHECK SEQUENCE OF DAYS.
C-----IGNORE MULTIPLE DAYS----FIRST VALUE SHOULD BE 01.
 2    IF(DAY(1).EQ.01)GO TO 7
      WRITE (6,4)
      WRITE (NUE,4)
 4    FORMAT(1X,'*****FIRST DAY VALUE IN FILE NOT 01****************')
      IFLAG=1
      GO TO 99
C
 7    NUMDM1=NUMDAY-1
      DO 3 I=2,NUMDM1
      DAYDIF=DAY(I-1)-DAY(I)
      IF(DAYDIF.EQ.0.OR.DAYDIF.EQ.-1)GO TO 8
      IF(DAYDIF.GT.0)GO TO 1
C-----AT LEAST A DAY MISSING
      WRITE (6,3)
      WRITE (NUE,3)
 3    FORMAT(1X,'*****DAY MISSING**********************************')
      IFLAG=1
      GO TO 8
C-----DAY OUT OF SEQUENCE
 1    WRITE (NUE,5)
      WRITE (6,5)
 5    FORMAT(1X,'*****DAY OUT OF SEQUENCE***************************')
      IFLAG=1
 8    CONTINUE
      IF(IFLAG.EQ.1)GO TO 99
C
 9    WRITE (NUE,10)
 10   FORMAT(10X)
C
 99   RETURN
      END
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

## APPENDIX B

## BULLETIN PROGRAMS

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                    THE EVENT LIST SUITE OF PROGRAMS
                    --------------------------------

                                 SUB-DIRECTORY              ELECTRIC
                                 NAME                       FILE NAME
-------------------------------------------------------------------------
 A.(JOB CONTROL +
    ELECTRIC) PROGRAMS.
-----------------------
 (1)BULLETIN                     CLT                        BULLETIN
 (2)BULLETIN.ED                  CLT                        BULLETIN.ED
 (3)BULLETIN*                    MASTER                     JCLBUL50
 (4)BULLETIN.ED*                 MASTER                     JCLBUL50.ED

-------------------------------------------------------------------------
 B. FORTRAN 4 PROGRAMS.                         BINARY
-----------------------                         VERSION
 (1)MAIN(1)                      MASTER         XA3WSTG2    SELECT50
 (2)MAIN(2)**                    MASTER         XA3WSTG3    BULBEG50
 (3)FRONTP**                     MASTER           ..        FRONT50
 (4)PAGES**                      MASTER           ..        PAGE50
 (5)HEAD**                       MASTER           ..        HEAD50
 (6)PRELIM**                     MASTER           ..        PRELIM50
 (7)CHARA1                       MASTER         XA3WND50    NEWDEC50

-------------------------------------------------------------------------
 C. MARKERS FOR END OF DATA FILES.
 --------------------------------
 (1)STOZERO                      CLT                        STOZERO
 (2)ENDBLOC                      CLT                        ENDBLOC
 (3)ENDBLOCK                     CLT                        ENDBLOCK
 (4)ENDFILE                      CLT                        ENDFILE

-------------------------------------------------------------------------
D. OTHER PROGRAMS NOT INCLUDED SEE 360 COMPUTER MANUAL LISTED IN REFERENCES.
-------------------------------------------------------------------------
 (1)CLOCK
 (2)CLFILE

-------------------------------------------------------------------------
 E. PERMANENT DATASETS ON 360/195 SYSTEM.
 ---------------------------------------
DATASET NAME            CONTENTS
------------            --------
(1)XA3W.ADDENDA         REPLICA OF LATEST EVENT LIST ADDENDA FILE IN ELECTRIC.
(2)XA3W.BULLETIN        REPLICA OF LATEST EVENT LIST BULLETIN FILE IN ELECTRIC.
(3)XA3W.ENDFILE         END OF FILE MARKER.
(4)XA3W.ERRATA          REPLICA OF LATEST EVENT LIST ERRATA FILE IN ELECTRIC.
(5)XA3W.STOZERO         END OF FILE MARKER.

-------------------------------------------------------------------------
NOTES    * INDICATES THAT THESE PROGRAMS (ARCHIVED IN SUB-DIRECTORY MASTER)
           ARE DUPLICATE COPIES OF THE DATABASE PROGRAMS IN ROUTINE USE.

  ** BINARY VERSIONS ARE ALL IN LIBRARY MEMBER XA3WSTG3
-------------------------------------------------------------------------

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

/*PRIORITY     12
  XA3WBULL  JOB (ACCT,IC,0-10),BLAMEY
/*ROUTE  PRINT ELECTRIC
//*
//* ---------------------------------------------------------------
//*
//*                    BULLETIN PROGRAM
//*                    ****************
//*
//* (360 JOB CONTROL AND ELECTRIC LANGUAGES) PROGRAM USED TO PRODUCE
//* THE SEISMOLOGICAL EVENT LIST EVERY TWO WEEKS FROM THE BLACKNEST
//* DATABASE FILES KEPT ON THE 360/195 SYSTEM COMPUTER AT RUTHERFORD
//* LABORATORY ,DIDCOT ,OXON ,UK .
//*
//* NOTE (1)  THIS ELECTRIC FILE BULLETIN HAS A CORRESPONDING EDIT FILE
//*           BULLETIN.ED.
//*
//* NOTE (2)  THIS PROGRAM CAN BE EXECUTED IN VARIOUS WAYS E.G. (A)WITH
//*           ALL FORTRAN PROGRAMS AS LOAD MODULES (IN BINARY). (B)WITH
//*           ALL PROGRAMS IN FORTRAN TO BE COMPILED INTO BINARY
//*           REPLACING THE OLDER VERSIONS IN THE COMPUTER LIBRARY
//*                                                          (C)WITH
//*           SOME PROGRAMS IN FORTRAN TO BE COMPILED AND WRITTEN INTO
//*           THE COMPUTER LIBRARY AND THE REMAINDER AS LOAD MODULES.
//*                   THE OPTIONS AVAILABLE ARE INDICATED IN THE FILE
//*           BULLETIN.ED.
//*
//* NOTE (3)  INPUT REQUIREMENTS OF THE BULLETIN PROGRAM.
//*           ---------------------------------------
//*           THE NAME OF THE FILE INTO WHICH INFORMATION AND DATA IS
//*           PUT IN SUB-DIRECTORY 'OUT' IS INPUT.
//*           THE FIRST THREE LINES OF ENTRIES IN FILE INPUT ARE
//*           ENTER IN ORDER (THE ORDER OF ENTRIES IN A PARTICULAR
```

```
//*            LINE ARE ALSO FIXED). FOR ALL LINES THE ENTRIES COMMENCE
//*            IN COLUMN 1. X BELOW INDICATES A SINGLE SPACE BETWEEN DATA
//*            ENTRIES.
//*
//*            LINE 1.
//*            EVENT LIST NUMBER X YEAR
//*
//*            LINE 2.
//*            START DAY DATE (INCLUSIVE) X FINISH DAY DATE (INCLUSIVE)
//*
//*            LINE 3.
//*            FIRST MONTH X SECOND MONTH (IF ANY)
//*
//*            LINES 4 AND ONWARDS
//*            THE REMAINDER OF THE LINE ENTRIES CAN BE IN ANY ORDER
//*            (THE ORDER OF ENTRIES IN ANY PARTICULAR LINE ARE FIXED)
//*            ONLY THE PRESENCE OF THE WORD PART2BUL IS MANDATORY.
//*            PROVISION HAS BEEN MADE FOR UP TO 3 STATIONS TO BE LEFT
//*            OUT OF ANY PARTICULAR EVENT LIST.TO LEAVE OUT A STATION
//*            CHOSE THE APPROPIATE PARAMETER FROM THE SET -
//*            NOWOL, NOEKA, NOGBA, NOWRA, NOYKA.
//*            AND INSERT AS THE FOURTH LINE (SAY), STARTING IN
//*            COLUMN 1, IN THE FILE INPUT.
//*            TO INCLUDE AN ERRATA SECTION IT IS NECESSARY TO PUT IN
//*            THE WORD ERRATA, STARTING IN COLUMN 1.
//*            SIMILARLY TO INCLUDE AN ADDENDA SECTION INSERT THE WORD
//*            ADDENDA STARTING IN COLUMN 1.
//*            IF ANALYSED DATA FROM THE FILE PRELIDET ARE TO BE INCLUDED
//*            THEN AFTER THE WORD PART2BUL (WHICH SHOULD COMMENCE IN
//*            COLUMN 1) THE APPROPIATE LINE NUMBERS FROM PRELIDET
//*            MUST BE INSERTED IN THE FORM -
//*            PART2BUL 001 024  THEN DATA FROM LINES 1 TO 24 WOULD BE
//*            INCLUDED IN THE EVENT LIST.
//*
//*  NOTE (4)  EXECUTION OF THE PROGRAM IS BY SUBMITTING THE COMMAND -
//*            ---------------------------------------------
//*            EXEC FL=BULLETIN(P8),MONTH1=OCT76,MONTH2=NOV76
//*            ---------------------------------------------
//*            WHERE MONTH1 AND MONTH2 ARE THE APPROPIATE NAMES OF THE
//*            DATA FILES IN THE DATABASE. IF THE EVENT LIST FALLS
//*            ENTIRELY IN ONE MONTH THEN MONTH2 MUST BE SET TO THE
//*            SAME PARAMETER AS MONTH1.
//*            THE COMMAND AS IT STANDS EXECUTES THE PROGRAM IN BINARY
//*            FORM. OTHER POSSIBILITIES ARE GIVEN IN THE EDIT FILE
//*            CORRESPONDING TO THIS FILE.
//*
//* ------------------------------------------------------------------
//* STAGE 1.GATHER REQUISITE ELECTRIC DATA FILES TOGETHER
//* ALSO SOME OTHER FILE PREPARATION.
//* ------------------------------------------------------------------
//*
//* **********
// EXEC ELSEND
//* **********
//DEL  DD VOL=REF=RHELO3,DISP=(OLD,DELETE),DSN=XA3W.BULLETIN
//G.SYSIN DD *
LOGIN ID=ID,ACCT=ACCT,KEY=XXXX
SETD C=A.OUT
DELETE FL=ADDS.ED
ENTER FL=ADDS.ED,LB=1
&A LN=0,FL=A.OUT.INPUT(NO)
&A LN=0,FL=A.OUT.ENDBLOCK(NO)
&A LN=0,FL=A.WOL.MONTH1(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.WOL.MONTH1(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.EKA.MONTH1(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.EKA.MONTH1(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.GBA.MONTH1(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.GBA.MONTH1(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.WRA.MONTH1(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.WRA.MONTH1(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.YKA.MONTH1(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.YKA.MONTH1(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.WOL.MONTH2(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.WOL.MONTH2(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.EKA.MONTH2(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.EKA.MONTH2(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.GBA.MONTH2(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.GBA.MONTH2(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.WRA.MONTH2(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.WRA.MONTH2(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.YKA.MONTH2(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
&A LN=0,FL=A.YKA.MONTH2(NO)
&A LN=0,FL=A.OUT.ENDBLOC(NO)
```

```
&&
COPY FL=ADDS,TODSN=XA3W.ADDS,VOL=RHEL03,DISP=NEW
COPY FL=A.OUT.ERRATA,TODSN=XA3W.ERRATA,VOL=RHEL03,DISP=OLD
COPY FL=A.OUT.ADDENDA,TODSN=XA3W.ADDENDA,VOL=RHEL03,DISP=OLD
COPY FL=A.PRELIDET,TODSN=XA3W.PRELIDET,VOL=RHEL03,DISP=NEW
DELETE FL=ERRATAR
COPY FL1=ERRATA,FL2=ERRATAR
CLEAR FL=ERRATA
DELETE FL=ADDENDAR
COPY FL1=ADDENDA,FL2=ADDENDAR
CLEAR FL=ADDENDA
CLEAR FL=INPUT
SETO C=3WMAINDR
DELETE FL=PREVBULL
RENAME FL1=BULLETIN,FL2=PREVBULL
LOGOUT
/*
//*
//* ------------------------------------------------------------------
//* STAGE 2.SELECT REQUIRED FORTNIGHTS DATA FROM THE DATA BLOCK.
//* ------------------------------------------------------------------
//*
//* ******************************************************
// EXEC FHCL,LIBRARY='ULIB.ISC',MEMBER=XA3WSTG2,CPRINT=YES,
//    REGION.C=250K
//C.SYSIN DD *
/*
//L.LIB  DD  DSN=ULIB.ISC,DISP=SHR
//L.SYSIN DD *
 INCLUDE LIB(XA3WND50)
 ENTRY MAIN
//* ********************************************************
// EXEC JOBLIB,LIBRARY='ULIB.ISC',MEMBER=XA3WSTG2,REGION.G=170K
//* ********************************************************
//* NOTE  CARD10=(RECFM=FB,LRECL=80,BLKSIZE=800)
//G.FT10F001 DD UNIT=WORK,SPACE=(TRK,2,RLSE),
// DSN=&&SCC1,DISP=(NEW,PASS),DCB=CARD10
//G.FT11F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCA1,DISP=(NEW,PASS),DCB=CARD10
//G.FT12F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCA2,DISP=(NEW,PASS),DCB=CARD10
//G.FT13F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCA3,DISP=(NEW,PASS),DCB=CARD10
//G.FT14F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCA4,DISP=(NEW,PASS),DCB=CARD10
//G.FT15F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCA5,DISP=(NEW,PASS),DCB=CARD10
//G.FT21F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCB1,DISP=(NEW,PASS),DCB=CARD10
//G.FT22F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCB2,DISP=(NEW,PASS),DCB=CARD10
//G.FT23F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCB3,DISP=(NEW,PASS),DCB=CARD10
//G.FT24F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCB4,DISP=(NEW,PASS),DCB=CARD10
//G.FT25F001 DD UNIT=WORK,SPACE=(TRK,(5,5),RLSE),
// DSN=&&SCB5,DISP=(NEW,PASS),DCB=CARD10
//G.SYSIN DD DSN=XA3W.ADDS,DISP=(OLD,DELETE),VOL=REF=RHEL03
//*
//* ------------------------------------------------------------------
//* STAGE 3.PRODUCE BULLETIN
//*
//* (A)STORE IT WITHIN SHORT TERM(I.E. LESS THAN 1 MONTH) FACILITIES OF
//* 360/195.
//*
//* (B)ATTEMPT(THIS WOULD USUALLY BE SUCCESSFUL) TO PLACE BULLETIN BACK
//* INTO APPROPRIATE ELECTRIC FILE WITHIN FRAMEWORK OF THIS JOB.
//*
//* ************************NOTE*****************************************
//* (C)IF ELECTRIC WAS NOT AVAILABLE WHEN THIS JOB WAS RUN USER COULD
//* STILL COPY BULLETIN OUT OF 360 INTO APPROPRIATE ELECTRIC FILE AT
//* SOME SUBSEQUENT TIME.
//* ------------------------------------------------------------------
//*
//* ******************************************************
// EXEC FHCL,LIBRARY='ULIB.ISC',MEMBER=XA3WSTG3,CPRINT=YES
//C.SYSIN DD *
//L.LIB  DD  DSN=ULIB.ISC,DISP=SHR
//L.SYSIN DD *
 INCLUDE LIB(XA3WND50)
 ENTRY MAIN
//* ********************************************************
// EXEC JOBLIB,LIBRARY='ULIB.ISC',MEMBER=XA3WSTG3,REGION.G=510K
//* ********************************************************
//*
//* ------------------------------------------------------------------
//* SELECTED DATA ON 360/195
//* ------------------------------------------------------------------
//*
//G.FT05F001 DD DSN=&&SCC1,DISP=(OLD,PASS)
//   DD DSN=&&SCA1,DISP=(OLD,PASS)
//   DD DSN=&&SCB1,DISP=(OLD,PASS)
//   DD DSN=XA3W.STOZERO,DISP=OLD,VOL=REF=RHEL03
//   DD DSN=&&SCA2,DISP=(OLD,PASS)
//   DD DSN=&&SCB2,DISP=(OLD,PASS)
//   DD DSN=XA3W.STOZERO,DISP=OLD,VOL=REF=RHEL03
//   DD DSN=&&SCA3,DISP=(OLD,PASS)
//   DD DSN=&&SCB3,DISP=(OLD,PASS)
//   DD DSN=XA3W.STOZERO,DISP=OLD,VOL=REF=RHEL03
//   DD DSN=&&SCA5,DISP=(OLD,PASS)
//   DD DSN=&&SCB5,DISP=(OLD,PASS)
//   DD DSN=XA3W.STOZERO,DISP=OLD,VOL=REF=RHEL03
```

```
//   DD DSN=&&SCA4,DISP=(OLD,PASS)
//   DD DSN=&&SCB4,DISP=(OLD,PASS)
//   DD DSN=XA3W.STOZERO,DISP=OLD,VOL=REF=RHEL03
//   DD DSN=XA3W.ERRATA,DISP=OLD,VOL=REF=RHEL03
//   DD DSN=XA3W.ENDFILE,DISP=OLD,VOL=REF=RHEL03
//   DD DSN=XA3W.ADDENDA,DISP=OLD,VOL=REF=RHEL03
//   DD DSN=XA3W.ENDFILE,DISP=OLD,VOL=REF=RHEL03
//   DD DSN=XA3W.PRELIDFT,DISP=(OLD,DELETE),VOL=REF=RHEL03
//   DD DSN=XA3W.STOZERO,DISP=OLD,VOL=REF=RHEL03
//*
//* ------------------------------------------------------------------
//* PUT BULLETIN ON 360/195
//* ------------------------------------------------------------------
//*
//G.FT08F001 DD VOL=REF=RHEL03,SPACE=(TRK,(5,5),RLSE),
// DSN=XA3W.BULLETIN,DISP=(NEW,KEEP),DCB=CARD10
//*
//* ------------------------------------------------------------------
//* PUT BULLETIN BACK INTO ELECTRIC FILE BULLETIN.
//* ------------------------------------------------------------------
//*
//* ************
//   EXEC  BLSEND
//* ************
//G.SYSIN  DD  *
LOGIN ID=ID,ACCT=ACCT,KEY=XXXX
COPY FL=BULLETIN,FROMDSN=XA3W.BULLETIN,VOL=RHEL03
CHANGE FL=BULLETIN,ACCESS=FRRR,ADDPINR=P1 P2
LOGOUT
/* END OF JOB XA3WBULL

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 BULLETIN.CD
 -----------


&G LB= 1,LN=    0,CS=ONLY( 1, 2) COMPILE/LINK STAGE 2 AND GO.
&G LB= 1,LN=    0,FS=ONLY( 1) COMPILE/LINK STAGES 1 & 2 AND GO.
&G LB= 1,LN=    0,OG=ONLY( 1, 2, 3, 4) EXECUTE WITH ALL LOAD MODULES.
&G LB= 1,LN=    0,CF=ONLY( 1, 3) COMPILE/LINK STAGE 1 AND GO.
&G LB= 1,LN=    0,P8=ONLY( 1, 2, 3) STORAGE USED GT DEFAULT P8 & GO.
&X LB= 1,LN=    0
&X LB= 1,LN=    0 NOTE  CURRENTLY CANNOT USE THE DEFAULT POSSIBILITY
&X LB= 1,LN=    0       EXEC FL=BULLETIN,MONTH1=*****,MONTH2=-----.
&X LB= 1,LN=    0       BECAUSE WITHIN THE JOB THE REGION SIZE PARAMETER
&X LB= 1,LN=    0       EXCEEDS IT,S DEFAULT VALUE.USE THE FOLLOWING INSTEAD
&X LB= 1,LN=    0       EXEC FL=BULLETIN(P8),MONTH1=*****,MONTH2=-----.
&X LB= 1,LN=    0
&X LB= 1,LN=    0 FUNCTION OF EACH OF THE LABELS.
&X LB= 1,LN=    0 ==============================
&X LB= 1,LN=    0  (1)LB1 = ORDINARY EDITS.
&X LB= 1,LN=    0  (2)LB2 = SUPPRESS FIRST FHCL.
&X LB= 1,LN=    0  (3)LB3 = SUPPRESS SECOND FHCL.
&X LB= 1,LN=    0  (4)LB4 = DEFINE JOB AS P12 AND TIME = 10 SECS.
&X LB= 1,LN=    0
&E LB= 4,LN=    1,C1= 16,C2= 16 12
&E LB= 4,LN=    2,C1= 28,C2= 29 10
&P LB= 1,LN=   96,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH1
&P LB= 1,LN=   98,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH1
&P LB= 1,LN=  100,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH1
&P LB= 1,LN=  102,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH1
&P LB= 1,LN=  104,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH1
&P LB= 1,LN=  106,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH1
&P LB= 1,LN=  108,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH1
&P LB= 1,LN=  110,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH1
&P LB= 1,LN=  112,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH1
&P LB= 1,LN=  114,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH1
&P LB= 1,LN=  116,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH2
&P LB= 1,LN=  118,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH2
&P LB= 1,LN=  120,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH2
&P LB= 1,LN=  122,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH2
&P LB= 1,LN=  124,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH2
&P LB= 1,LN=  126,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH2
&P LB= 1,LN=  128,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH2
&P LB= 1,LN=  130,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH2
&P LB= 1,LN=  132,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH2
&P LB= 1,LN=  134,C1= 18,C2= 23,CH=AN,DF=NO,NM=MONTH2
&X LB= 2,LN=  134
&X LB= 2,LN=  134  EXPLANATION OF ACTION OF FOLLOWING THREE LINES
&X LB= 2,LN=  134  NAMELY &D , &C AND &S .
&X LB= 2,LN=  134  THE &D DELETES LINES 159 TO 166 OF THE FL=BULLETIN.
&X LB= 2,LN=  134  BUT IT IS NECESSARY USING THE &C TO CANCEL THE
&X LB= 2,LN=  134  FORTRAN ROUTINE WHICH IS SUPPLIED BY THE &S
&X LB= 2,LN=  134  IMMEDIATELY FOLLOWING THE &C COMMAND ---
&X LB= 2,LN=  134  OTHERWISE A FORTRAN ROUTINE IS SUPPLIED WITH
&X LB= 2,LN=  134  NO SLOT FOR IT TO GO INTO.
&X LB= 2,LN=  134       THIS PROCEDURE ALSO APPLIES FURTHER DOWN
&X LB= 2,LN=  134  IN THIS EDIT FILE.
&X LB= 2,LN=  134
&D LB= 2,LN=  159,L2= 166
&C LB= 2,LN=  161   CCC CCC
&S LB= 1,LN=  161,DF=NO,NM=FA
&D LB= 3,LN=  211,L2= 216
&C LB= 3,LN=  212   CCC CCC
&S LB= 1,LN=  212,DF=NO,NM=F1
&C LB= 3,LN=  212   CCC CCC
&S LB= 1,LN=  212,DF=NO,NM=F2
&C LB= 3,LN=  212   CCC CCC
&S LB= 1,LN=  212,DF=NO,NM=F3
&C LB= 3,LN=  212   CCC CCC
&S LB= 1,LN=  212,DF=NO,NM=F4
&C LB= 3,LN=  212   CCC CCC
&S LB= 1,LN=  212,DF=NO,NM=F5
```

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 STOZERO
 -------

       00

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 ENDBLOCK
 --------

       99

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 ENDBLOC
 -------

99

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 ENDFILE
 -------

FILE-END

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


C*********************************************************************
C
C                    PROGRAM MAIN(1)
C                    ***************
C THIS IS A FORTRAN 4 PROGRAM WHICH SELECTS THE CHOSEN RANGE OF DATA
C FROM THE BLACKNEST DATABASE FILES.
C
C NOTE  COMMENTS HAVE BEEN INSERTED IN ALL THE PROGRAMS TO AID THE
C READER .
C
C*********************************************************************
C
C
      DIMENSION DAY(1000),LINEST(5),LINEFN(5)
      DIMENSION ALPHAC(54,650),NOENP1(5),NUMEP1(5)
      DIMENSION INFORM(10),LDATA(10),ALPHA(24),NCHARN(10),ICHAR(8)
C
      INTEGER DAYBGK,DAYFOK,DAYBEG,DAYEND,DYSTOP,DAY,DAY1,DAY2
      INTEGER YEAR,BULLNO,BLANKA,ALPHAC,ALPHA,PRTDAY
      INTEGER JAN,FEB,MAR,APR,MAY,JUN,JUL,AUG,SEP,OCT,NOV,DEC
C
      REAL*8  BLANK/8H        /,NINES/8H99      /,PART2/8HPART2BUL/
      REAL*8  ERR/8HERRATA  /,ADD/8HADDENDA /,INFORM,LDATA,RV
      REAL*8  STNWOL/8HNOWOL   /,STNEKA/8HNOEKA   /,STNGBA/8HNOGBA   /
      REAL*8  STNYKA/8HNOYKA   /,STNWRA/8HNOWRA   /
C
      DATA JAN/4HJAN /,FEB/4HFEB /,MAR/4HMAR /,APR/4HAPR /
      DATA MAY/4HMAY /,JUN/4HJUN /,JUL/4HJUL /,AUG/4HAUG /
      DATA SEP/4HSEP /,OCT/4HOCT /,NOV/4HNOV /,DEC/4HDEC /
      DATA BLANKA/4H    /,MINUS/4H-   /
      DATA NCHARN/4H0   ,4H1   ,4H2   ,4H3   ,4H4   ,4H5   ,4H6   ,
     14H7   ,4H8   ,4H9   /
C
C
C
C-----CAN OBTAIN PRINT-OUT (STREAM 6) - FOR DEBUG PURPOSES - BY SETTING
C-----PARAMETER PRTDAY=1 .
      PRTDAY=0
C
      MON1=BLANKA
      MON2=BLANKA
C
      DO 84 I=1,5
      NOENP1(I)=0
      NUMEP1(I)=0
 84   CONTINUE
C
      DO 205 J=1,10
      INFORM(J)=BLANK
      LDATA(J)=BLANK
 205  CONTINUE
C
      NOTENM=1
C
C-----BULLETIN NUMBER AND YEAR INPUT.
      READ 3,BULLNO,YEAR
 3    FORMAT(A2,1X,A2)
      WRITE (6,18)BULLNO,YEAR
 18   FORMAT(1X,A2,1X,A2)
C
C-----INPUT SEARCH INFORMATION.
      READ 16,(ALPHA(I),I=1,24)
 16   FORMAT(24A1)
      WRITE (6,166)(ALPHA(I),I=1,24)
 166  FORMAT(1X,24A1)
      CALL CHARAI(ALPHA,NCHARN,1,2,RV,DAY1)
      CALL CHARAI(ALPHA,NCHARN,4,2,RV,DAY2)
```

```
      CYSTOP=DAY2+1
      READ 15,MON1,MON2
 15   FORMAT(2A4)
      WRITE (6,17)MON1,MON2
 17   FORMAT(1X,2A4)
      IF(MON2.EQ.BLANKA)MON2=MON1
      J=1
C-----NUM1 AND NUM2 READ IN IMMEDIATELY BELOW ARE RELEVANT ONLY TO
C-----INFORM(J)=PART2BOL.
      LINE1=0
      LINE2=0
 204  READ 201,INFORM(J),NUM1,NUM2
 201  FORMAT(A8,1X,I3,1X,I3)
      IF(INFORM(J).NE.PART2)GO TO 167
      LINE1=NUM1
      LINE2=NUM2
 167  WRITE (6,202)INFORM(J),LINE1,LINE2,NOTENM
 202  FORMAT(2X,A8,5X,I5,5X,I5,5X,I5)
      IF(INFORM(J).EQ.NINES)GO TO 203
      J=J+1
      NOTENM=J
      GO TO 204
C
 203  NER=0
      NAD=0
      NWO=0
      NEK=0
      NGB=0
      NWR=0
      NYK=0
      NPART2=0
      IF(NOTENM.EQ.1)GO TO 206
      NMNM1=NOTENM-1
      DO 207 J=1,NMNM1
      IF(INFORM(J).EQ.ERR)NER=1
      IF(INFORM(J).EQ.ADD)NAD=1
      IF(INFORM(J).EQ.STNWOL)NWO=1
      IF(INFORM(J).EQ.STNEKA)NEK=1
      IF(INFORM(J).EQ.STNGBA)NGB=1
      IF(INFORM(J).EQ.STNWRA)NWR=1
      IF(INFORM(J).EQ.STNYKA)NYK=1
      IF(INFORM(J).EQ.PART2)NPART2=1
 207  CONTINUE
C
      IF(NER.NE.1)GO TO 220
      LDATA(1)=ERR
 220  IF(NAD.NE.1)GO TO 221
      LDATA(2)=ADD
 221  NSTOUT=NWO+NEK+NGB+NWR+NYK
      IF(NSTOUT.GT.3)RETURN
      IF(NSTOUT.EQ.0)GO TO 228
      N=1
      IF(NWO.EQ.0)GO TO 223
      NPL2=N+2
      LDATA(NPL2)=STNWOL
      N=N+1
 223  IF(NEK.EQ.0)GO TO 224
      NPL2=N+2
      LDATA(NPL2)=STNEKA
      N=N+1
 224  IF(NGB.EQ.0)GO TO 226
      NPL2=N+2
      LDATA(NPL2)=STNGBA
      N=N+1
 226  IF(NWR.EQ.0)GO TO 227
      NPL2=N+2
      LDATA(NPL2)=STNWRA
      N=N+1
 227  IF(NYK.EQ.0)GO TO 228
      NPL2=N+2
      LDATA(NPL2)=STNYKA
      N=N+1
C
 228  WRITE (10,230)BULLNO,YEAR
 230  FORMAT(1X,2A4)
      WRITE (10,231)(ALPHA(I),I=1,24)
 231  FORMAT(1X,24A1)
      WRITE (10,232)MON1,MON2
 232  FORMAT(1X,2A4)
      WRITE (10,233)BULLNO,MINUS,YEAR,(LDATA(I),I=1,5),LINE1,LINE2
 233  FORMAT(1X,A2,A1,A2,5X,A8,2X,A8,2X,A8,2X,A8,2X,A8,2X,I3,2X,I3)
      WRITE (10,235)(ALPHA(I),I=1,2),MON1,YEAR,MINUS,
     1(ALPHA(I),I=4,5),MON2,YEAR
 235  FORMAT(1X,5H0000 ,2A1,1X,A4,A2,1X,A1,1X,6H 2400 ,2A1,1X,A4,A2)
      CALL CLFILE(10)
C
C-----FIND NUMBER OF DAYS IN MONTH MON1.
 206  IF(MON1.NE.JAN)GO TO 40
      GO TO 61
 40   IF(MON1.NE.FEB)GO TO 41
      DIVYB4=FLOAT(YEAR)/4.0
      NUM=DIVYB4
      ANUM=DIVYB4-FLOAT(NUM)
      IF(ANUM.EQ.0.0)GO TO 42
      NDYMON=28
      GO TO 70
 42   NDYMON=29
      GO TO 70
 41   IF(MON1.NE.MAR)GO TO 43
      GO TO 61
 43   IF(MON1.NE.APR)GO TO 44
      GO TO 60
```

```
 44    IF(MON1.NE.MAY)GO TO 45
       GO TO 61
 45    IF(MON1.NE.JUN)GO TO 46
       GO TO 60
 46    IF(MON1.NE.JUL)GO TO 47
       GO TO 61
 47    IF(MON1.NE.AUG)GO TO 48
       GO TO 61
 48    IF(MON1.NE.SEP)GO TO 49
       GO TO 60
 49    IF(MON1.NE.OCT)GO TO 50
       GO TO 61
 50    IF(MON1.NE.NOV)GO TO 51
       GO TO 60
 51    IF(MON1.NE.DEC)GO TO 52
       GO TO 61
 52    RETURN
 60    NDYMON=30
       GO TO 70
 61    NDYMON=31
       GO TO 70
 70    NDMNP1=NDYMON+1
C
C-----DOES THE EVENT LIST OVERLAP INTO SECOND MONTH.
       NUMONT=1
       IF(MON2.NE.MON1)NUMONT=2
       IF(NUMONT.EQ.2)GO TO 100
C
C-----READ DAYS FROM ARPA-FILE FOR STATION .
C-----FILE OF DATA FOR A PARTICULAR MONTH.
       K=1
C-----REMOVE ANY END OF FILE MARKERS FOR STATIONS WHOSE
C-----DATA ARE NOT AVAILABLE.
       NSTR=11
       IF(NWO.EQ.0)GO TO 8
       GO TO 410
 401   NSTR=12
       IF(NEK.EQ.0)GO TO 8
       GO TO 410
 402   NSTR=13
       IF(NGB.EQ.0)GO TO 8
       GO TO 410
 403   NSTR=14
       IF(NWR.EQ.0)GO TO 8
       GO TO 410
 404   NSTR=15
       IF(NYK.EQ.0)GO TO 8
       GO TO 410
 410   READ 9,NWAST1
       WRITE (6,2)NWAST1
 2     FORMAT(8X,I2)
       READ 9,NWAST2
       WRITE (6,2)NWAST2
       WRITE (NSTR,315)
 315   FORMAT(10X)
       REWIND NSTR
       GO TO 400
 8     KEY=0
       J=1
 7     READ 9,DAY(J)
 9     FORMAT(8X,I2)
       IF(PRTDAY.EQ.1)WRITE (6,2)DAY(J)
       NOENP1(K)=J
       IF(DAY(J).EQ.99)GO TO 10
       J=J+1
       GO TO 7
C
C-----CHECK THAT DAY RANGE GIVEN IS WITHIN DATA FILE LIMITS.
 10    DAYBEG=DAY(1)
       NOENTR=NOENP1(K)-1
       DAYEND=DAY(NOENTR)
       DAYBOK=0
       DAYFOK=0
       DO 11 J=1,NOENTR
       IF(DAY(J).EQ.DAY1)DAYBOK=1
       IF(DAY(J).EQ.DAY2)DAYFOK=1
 11    CONTINUE
       IF(DAYBOK.EQ.1.AND.DAYFOK.EQ.1)GO TO 12
       RETURN
C
C-----FIND THE REQUIRED LINE NUMBERS INORDER TO SELECT DATA.
 12    DO 21 J=1,NOENTR
       IF(DAY(J).NE.DAY1)GO TO 6
       KEY=KEY+1
       IF(KEY.EQ.1)LINEST(K)=J
 6     IF(DAY(J).EQ.DYSTOP)GO TO 22
 21    CONTINUE
       IF(DYSTOP.EQ.NDMNP1)LINEFN(K)=NOENTR
       GO TO 4
 22    LINEFN(K)=J-1
C
C-----STREAM SELECTED DATA TO APPROPIATE ELECTRIC FILES.
 4     NSTREM=6
 83    LBEG=LINEST(K)
       LFIN=LINEFN(K)
       NA=NOENP1(K)
       READ 81,((ALPHAC(I,J),I=1,54),J=1,NA)
       DO 80 J=LBEG,LFIN
       WRITE (NSTR,81)(ALPHAC(I,J),I=1,54)
 81    FORMAT(54A1)
       IF(PRTDAY.EQ.1)WRITE (NSTREM,81)(ALPHAC(I,J),I=1,54)
 80    CONTINUE
```

```
 400  K=K+1
      CALL CLFILE(NSTR)
      KM1=K-1
      GO TO(401,402,403,404,405),KM1
C
C-----REMOVE UNWANTED STACK OF DATA(10 FILES) INCLUDING END OF FILE
C-----MARKERS (OCCURS BECAUSE MON2=MON1)
C-----THIS ALSO DEALS WITH THE EMPTY FILES.
C
 405  DO 31 K=1,10
      J=1
 332  READ 9,DAY(J)
      IF(PRTDAY.EQ.1)WRITE (6,2)DAY(J)
      IF(DAY(J).EQ.99)GO TO 31
      J=J+1
      GO TO 332
 31   CONTINUE
C
      K=1
      NSTR=21
 316  WRITE (NSTR,315)
      REWIND NSTR
      CALL CLFILE(NSTR)
      NSTR=NSTR+1
      K=K+1
      IF(K.LT.6)GO TO 316
      GO TO 30
C
C-----READ DAYS FROM ARPA-FILE FOR STATION.
C-----DEAL WITH EACH FILE FOR A STATION FOR THE FIRST MONTH.
 100  K=1
C-----REMOVE ANY END OF FILE MARKERS FOR STATIONS WHOSE
C-----DATA ARE NOT AVAILABLE.
      NSTR=11
      IF(NWO.EQ.0)GO TO 64
      GO TO 4100
 4010 NSTR=12
      IF(NEK.EQ.0)GO TO 64
      GO TO 4100
 4020 NSTR=13
      IF(NGB.EQ.0)GO TO 64
      GO TO 4100
 4030 NSTR=14
      IF(NWR.EQ.0)GO TO 64
      GO TO 4100
 4040 NSTR=15
      IF(NYK.EQ.0)GO TO 64
      GO TO 4100
 4100 READ 9,NWAST1
      WRITE (6,2)NWAST1
      READ 9,NWAST2
      WRITE (6,2)NWAST2
      WRITE (NSTR,315)
      REWIND NSTR
      GO TO 4000
 64   KEY=0
      J=1
 62   READ 9,DAY(J)
      IF(PRTDAY.EQ.1)WRITE (6,2)DAY(J)
      NOENP1(K)=J
      IF(DAY(J).EQ.99)GO TO 63
      J=J+1
      GO TO 62
C
C-----FIND IF DAY1 EXISTS IN THE FILE.
 63   DAYBEG=DAY(1)
      NOENTR=NOENP1(K)-1
      DAYEND=DAY(NOENTR)
      DAYBOK=0
      DO 32 J=1,NOENTR
      IF(DAY(J).EQ.DAY1)DAYBOK=1
 32   CONTINUE
      IF(DAYBOK.EQ.1)GO TO 33
      RETURN
C-----FIND LINE NUMBER CORRESPONDING TO GIVEN DAY FOR FIRST
C-----MONTHS DATA FOR EACH STATION.
 33   DO 25 J=1,NOENTR
      IF(DAY(J).NE.DAY1)GO TO 25
      KEY=KEY+1
      IF(KEY.EQ.1)LINEST(K)=J
 25   CONTINUE
C
C-----STREAM SELECTED DATA TO APPROPIATE ELECTRIC FILES.
      NSTREM=6
 88   LBEG=LINEST(K)
      LFIN=NOENP1(K)-1
      NA=NOENP1(K)
C-----NA IS UP TO AND INCLUDING 99 (END OF FILE MARKER).
      READ 81,((ALPHAC(I,J),I=1,54),J=1,NA)
      DO 89 J=LBEG,LFIN
      WRITE (NSTR,81)(ALPHAC(I,J),I=1,54)
 89   CONTINUE
 4000 K=K+1
      CALL CLFILE(NSTR)
      KM1=K-1
      GO TO(4010,4020,4030,4040,4050),KM1
C
C-----READ IN SECOND MONTH,S DATA TO FIND DAY2.
 4050 K=1
C-----REMOVE ANY END OF FILE MARKERS FOR STATIONS WHOSE
C-----DATA ARE NOT AVAILABLE.
      NSTR=21
```

```
      IF(NWO.EQ.0)GO TO 37
      GO TO 710
 701  NSTR=22
      IF(NEK.EQ.0)GO TO 37
      GO TO 710
 702  NSTR=23
      IF(NGB.EQ.0)GO TO 37
      GO TO 710
 703  NSTR=24
      IF(NWR.EQ.0)GO TO 37
      GO TO 710
 704  NSTR=25
      IF(NYK.EQ.0)GO TO 37
      GO TO 710
 710  READ 9,NWAST1
      WRITE (6,2)NWAST1
      READ 9,NWAST2
      WRITE (6,2)NWAST2
      WRITE (NSTR,315)
      REWIND NSTR
      GO TO 700
 37   J=1
 35   READ 9,DAY(J)
      IF(PRTDAY.EQ.1)WRITE (6,2)DAY(J)
      NUMEP1(K)=J
      IF(DAY(J).EQ.99)GO TO 101
      J=J+1
      GO TO 35
C
C-----CHECK THAT DAY2 IS WITHIN FILE LIMITS FOR EACH STATION.
 101  DAYBEG=DAY(1)
      NOENTR=NUMEP1(K)-1
      DAYEND=DAY(NOENTR)
      DAYFOK=0
      DO 38 J=1,NOENTR
      IF(DAY(J).EQ.DAY2)DAYFOK=1
 38   CONTINUE
      IF(DAYFOK.EQ.1)GO TO 39
      RETURN
C
C-----FIND LINE NUMBER CORRESPONDING TO DAY2.
 39   DO 90 J=1,NOENTR
      IF(DAY(J).EQ.DYSTOP)GO TO 91
 90   CONTINUE
      NODMP1=DAYEND+1
      IF(DYSTOP.EQ.NODMP1)LINEFN(K)=NOENTR
      GO TO 92
 91   LINEFN(K)=J-1
C
C-----STREAM SELECTED DATA TO APPROPIATE ELECTRIC FILES.
 92   NSTREM=6
 26   LBEG=1
      LFIN=LINEFN(K)
      NA=NUMEP1(K)
      READ 81,((ALPHAC(I,J),I=1,54),J=1,NA)
      DO 27 J=LBEG,LFIN
      WRITE (NSTR,81)(ALPHAC(I,J),I=1,54)
 27   CONTINUE
 700  K=K+1
      CALL CLFILE(NSTR)
      KM1=K-1
      GO TO(701,702,703,704,30),KM1
C
 30   RETURN
      END
```

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

```
C
C*****************************************************************
C
C                        PROGRAM MAIN(2)
C                        ***************
C THIS IS A FORTRAN 4 PROGRAM WHICH ORGANISES THE PRINTING OF THE EVENT
C LIST . THIS CONSISTS OF THE FRONTPAGE AND PART 1 (I.E. DATA, ERRATA,
C ADDENDA) AND PART 2 (ANALYSIS OF SELECTED EVENTS).
C
C*****************************************************************
C
C
      DIMENSION JDAY(600,6),NHR(600,6),MIN(600,6),SECS(600,6)
      DIMENSION AMP(600,6),PERIOD(600,6),LDAY(2,600,6)
      DIMENSION NCHARN(10),SPARE(6),ALAMP(6,600,6),ALPER(5,600,6)
      DIMENSION ALPHA(24),DATA(10),TITLE(7),AINFOR(10),STNOUT(3)
C
      DATA NCHARN/4H0    ,4H1    ,4H2    ,4H3    ,4H4    ,4H5    ,4H6    ,
     14H7    ,4H8    ,4H9    /
      DATA ZERO/4H0    /
C
      INTEGER ALAMP,ALPER,SPARE,DAY1,DAY2,BULLNO,YEAR,ZERO,CH1,CH2,ALPHA
      INTEGER STREM6
C
      REAL*8  BLANK/8H        /,TITLE,STNOUT,AMP,AM,RV,AINFOR,DATA
      REAL*8  ERR/8HERRATA  /,ADD/8HADDENDA /,ENDFIL/8HFILE-END/
      REAL*8  STNWOL/8HNOWOL   /,STNEKA/8HNOEKA   /,STNGBA/8HNOGBA   /
      REAL*8  STNYKA/8HNOYKA   /,STNWRA/8HNOWRA   /
C
      REAL*16  SECS
C
      NDIMEN=600
```

```
C
C
C-----CAN OBTAIN PRINT-OUT (USES STREAM 6) OF INTENDED BULK OF DATA
C-----FOR BULLETIN. - FOR DEBUG PURPOSES - BY SETTING PARAMETER STREM6=1
      STREM6=0
C
C-----NUMBER OF STATIONS ORIGINALLY INTENDED WAS 6. ONLY 5 ARE
C-----OPERATIONAL NOW.
      NSTAT=6
C
      DO 15 I=1,7
      TITLE(I)=BLANK
 15   CONTINUE
C
      READ 200,BULLNO,YEAR
 200  FORMAT(1X,2A4)
      WRITE(6,201)BULLNO,YEAR
 201  FORMAT(1X,A4,1X,A4)
C
      READ 202,(ALPHA(I),I=1,24)
 202  FORMAT(1X,24A1)
      WRITE (6,206)(ALPHA(I),I=1,24)
 206  FORMAT(1X,24A1)
      CALL CHARAI(ALPHA,NCHARN,1,2,RV,DAY1)
      CALL CHARAI(ALPHA,NCHARN,4,2,RV,DAY2)
C
      READ 204,MON1,MON2
 204  FORMAT(1X,2A4)
      WRITE(6,205)MON1,MON2
 205  FORMAT(1X,2A4)
C
      READ 16,(TITLE(I),I=5,7),(STNOUT(I),I=1,3),LINE1,LINE2
 16   FORMAT(1X,A8,2X,A8,2X,A8,2X,A8,2X,A8,2X,A8,2X,I3,2X,I3)
      WRITE (6,101)(TITLE(I),I=5,7),(STNOUT(I),I=1,3),LINE1,LINE2
 101  FORMAT(2X,6(A8,2X),I3,2X,I3)
      READ 14,(TITLE(I),I=1,4)
 14   FORMAT(1X,4A8)
      WRITE (6,102)(TITLE(I),I=1,4)
 102  FORMAT(2X,4A8)
C
C     PRINT FRONTISPIECE
C
      CALL FRONTP(TITLE,NPAG)
C
C     PRINT DATA PAGES
C
      DO 5 M=1,NSTAT
      DO 4 I=1,NDIMEN
      JDAY(I,M)=0
      NHR(I,M)=0
      MIN(I,M)=0
      SECS(I,M)=0.0
      AMP(I,M)=0.0
      PERIOD(I,M)=0.0
 4    CONTINUE
 5    CONTINUE
C
      NWOL=0
      NEKA=0
      NGBA=0
      NYKA=0
      NWRA=0
      IF(STNOUT(1).NE.BLANK.OR.STNOUT(2).NE.BLANK.OR.STNOUT(3).NE.BLANK)
     1GO TO 60
      GO TO 80
 60   DO 61 I=1,3
      IF(STNOUT(I).EQ.STNWOL)NWOL=1
      IF(STNOUT(I).EQ.STNEKA)NEKA=1
      IF(STNOUT(I).EQ.STNGBA)NGBA=1
      IF(STNOUT(I).EQ.STNYKA)NYKA=1
      IF(STNOUT(I).EQ.STNWRA)NWRA=1
 61   CONTINUE
C
 80   DO 19 M=1,NSTAT
      IF(M.EQ.1.AND.NWOL.EQ.1)GO TO 18
      IF(M.EQ.2.AND.NEKA.EQ.1)GO TO 18
      IF(M.EQ.3.AND.NGBA.EQ.1)GO TO 18
      IF(M.EQ.4)GO TO 19
      IF(M.EQ.5.AND.NYKA.EQ.1)GO TO 18
      IF(M.EQ.6.AND.NWRA.EQ.1)GO TO 18
C
      I=1
 8    READ 1,(LDAY(N,I,M),N=1,2),NHR(I,M),MIN(I,M),SECS(I,M)
     1,(ALAMP(N,I,M),N=1,6),(ALPER(N,I,M),N=1,5)
 1    FORMAT(8X,2A1,1X,A2,A2,A16,4X,6A1,8X,5A1)
      IF(STREM6.EQ.1)WRITE (6,1)(LDAY(N,I,M),N=1,2),NHR(I,M),MIN(I,M),
     1SECS(I,M),(ALAMP(N,I,M),N=1,6),(ALPER(N,I,M),N=1,5)
C
C-----DECODE THE DAY.
      DO 760 N=1,2
      SPARE(N)=LDAY(N,I,M)
 760  CONTINUE
      CALL CHARAI(SPARE,NCHARN,1,2,RV,IV)
      JDAY(I,M)=IV
C
      CH1=LDAY(1,I,M)
      CH2=LDAY(2,I,M)
      IF(CH1.EQ.ZERO.AND.CH2.EQ.ZERO)GO TO 2
C-----DECODE AMPLITUDE AND PERIOD.
      DO 750 N=1,6
      SPARE(N)=ALAMP(N,I,M)
 750  CONTINUE
```

```
      CALL CHARAI(SPARE,NCHARN,1,6,AM,IAM)
      IF(IAM.NE.9999)GO TO 751
      AMP(I,M)=AM
      GO TO 752
 751  AMP(I,M)=FLOAT(IAM)
 752  DO 753 N=1,5
      SPARE(N)=ALPER(N,I,M)
 753  CONTINUE
      CALL CHARAI(SPARE,NCHARN,1,5,PER,IPER)
      IF(IPER.NE.9999)GO TO 754
      PERIOD(I,M)=PER
      GO TO 755
 754  PERIOD(I,M)=FLOAT(IPER)
 755  I=I+1
      GO TO 9
 2    NUMLIN=I-1
      CALL PAGES(M,TITLE,JDAY,NHR,MIN,SECS,AMP,PERIOD,NUMLIN
     1,NPAG,ALAMP,ALPER,NCHARN,LDAY,MCN1,MCN2)
      IF(JDAY(I,M).EQ.0.AND.M.EQ.NSTAT)GO TO 40
      GO TO 19
C-----REMOVE END OF DATA FILE MARKER FOR ABSENT STATION.
 18   READ 10,(DATA(I),I=1,9)
 10   FORMAT(9A8)
      WRITE (6,104)(DATA(I),I=1,9)
 104  FORMAT(2X,9A8)
 19   CONTINUE
C
C     TITLES FOR ERRATA AND/OR ADDENDA ALSO READ AND PRINT OF
C     CORRESPONDING BULLETIN NUMBERS , STATIONS AND DATA .
C
 40   IF(TITLE(6).EQ.ERR.OR.TITLE(7).EQ.ADD)GO TO 41
      GO TO 822
 41   WRITE (8,17)TITLE(5)
 17   FORMAT(/////,48X,16HBULLETIN NUMBER ,A8)
      WRITE (8,30)
 30   FORMAT(48X,21H---------------------)
      WRITE (8,11)(TITLE(I),I=1,4)
 11   FORMAT(1X,'PART 1  EVENT LISTS FOR PERIOD ',4A8)
      WRITE (8,32)
 32   FORMAT(1X,63H-------------------------------------------------------------
     1----------)
      WRITE (8,12)
 12   FORMAT(/3X,19HFORMAT OF DATA IS -)
      WRITE (8,96)
 96   FORMAT(7X,10HDAY   TIME,18X,9HAMPLITUDE,4X,6HPERIOD)
      WRITE (8,97)
 97   FORMAT(11X,9HHRMN SECS,14X,12H(NANOMETRES),1X,
     19H(SECONDS))
      LINERR=0
      IF(TITLE(6).EQ.BLANK)GO TO 422
      WRITE (8,44)
 44   FORMAT(//30X,6HERRATA)
      WRITE (8,9)
 9    FORMAT(30X,6H------)
 49   READ 43,(DATA(I),I=1,9)
 43   FORMAT(9A8)
      WRITE (6,105)(DATA(I),I=1,9)
 105  FORMAT(2X,9A8)
      IF(DATA(1).EQ.ENDFIL)GO TO 92
      WRITE (8,27)(DATA(I),I=1,9)
 27   FORMAT(/3X,9A8)
      LINERR=LINERR+1
      GO TO 49
C-----REMOVE END OF DATA FILE MARKER FOR EMPTY ERRATA FILE.
 422  READ 10,(DATA(I),I=1,9)
      WRITE (6,105)(DATA(I),I=1,9)
C
 92   LINADD=0
      IF(TITLE(7).EQ.BLANK)GO TO 501
      WRITE (8,45)
 45   FORMAT(//30X,7HADDENDA)
      WRITE (8,26)
 26   FORMAT(30X,7H-------)
 51   READ 43,(DATA(I),I=1,9)
      WRITE (6,105)(DATA(I),I=1,9)
      IF(DATA(1).EQ.ENDFIL)GO TO 94
      WRITE (8,27)(DATA(I),I=1,9)
      LINADD=LINADD+1
      GO TO 51
C-----REMOVE END OF DATA FILE MARKER FOR EMPTY ADDENDA FILE.
 501  READ 10,(DATA(I),I=1,9)
      WRITE (6,105)(DATA(I),I=1,9)
C
C     PRINT PAGE NUMBER.
C
 94   NBLANK=71-17-5-2*LINERR-2*LINADD
      DO 90 I=1,NBLANK
      WRITE (8,91)
 91   FORMAT(10X)
 90   CONTINUE
      NPAG=NPAG+1
      WRITE (8,6)NPAG
 6    FORMAT(25X,5HPAGE ,I2,//)
      GO TO 82
C
C-----REMOVE END OF DATA FILE MARKERS FOR EMPTY ERRATA
C-----AND ADDENDA FILES.
 822  READ 10,(DATA(I),I=1,9)
      WRITE (6,105)(DATA(I),I=1,9)
      READ 10,(DATA(I),I=1,9)
      WRITE (6,105)(DATA(I),I=1,9)
C
```

```
 82   CALL PRELIM(TITLE,LINE1,LINE2,NPAG)
C
      RETURN
      END

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

C*****************************************************************
C
C             SUBROUTINE PROGRAM FRONTP
C             *************************
C THIS IS A FORTRAN 4 PROGRAM USED TO PRODUCE THE FRONT PAGE OF THE
C EVENT LIST.
C
C*****************************************************************
C
C
      SUBROUTINE FRONTP(TITLE,NPAG)
C
      DIMENSION TITLE(7)
C
      REAL*8  TITLE,TIME,DATE
C
      CALL CLOCK(TIME,DATE)
C
      WRITE(8,1)DATE,TITLE(5)
    1 FORMAT(/1X,'BLACKNEST SEISMIC EVENT SUMMARY',A8,26X,'NO. ',A8/
     11X,'----------------------------------------'26X,'---------')
      WRITE(8,2)
    2 FORMAT(//1X,'ISSUED BY   BLACKNEST DATA ANALYSIS CENTRE',7X,'PE-MC
     1D  BLACKNEST,')
      WRITE(8,3)
    3 FORMAT(50X,'BRIMPTON, NR READING  RG7 4RS'/
     150X,'BERKSHIRE, ENGLAND.')
      WRITE(8,4)(TITLE(I),I=1,4)
    4 FORMAT(/1X,'PART 1  EVENT LISTS FOR PERIOD   ',4A8//
     19X,'DATES,ARRIVAL TIMES,AMPLITUDES & PERIODS OF OBSERVED SEISMIC W
     1AVES*'/
     19X,'ARE GIVEN IN SEPARATE LISTS FOR EACH CONTRIBUTING STATION'/
     19X,'NOTE  *NORMALLY ONLY TELESEISMIC P-WAVES ARE REPORTED'/
     116X,'CURRENT DATA FORMAT OF LISTS IS BDAC 4')
      WRITE(8,5)(TITLE(I),I=1,4)
    5 FORMAT(/1X,'PART 2  ANALYSIS OF SELECTED EVENTS ',4A8//
     19X,'PROVISIONAL DETERMINATIONS OF EPICENTRES,ORIGIN TIMES & MAGNIT
     1UDES OF'/
     19X,'EVENTS LOCATED IN PROXIMITY OF NUCLEAR TEST SITES OR IN ASEISM
     1IC AREAS'/
     19X,'ARE GIVEN.'/
     19X,'NOTE   THESE DETERMINATIONS ARE MADE USING ONLY THE DATA FROM'
     1/16X,'THE STATIONS LISTED UNDER EACH ORIGIN TIME/LOCATION'/
     116X,'ESTIMATE.'/
     116X,'WHEN DATA FROM ONLY A SINGLE ARRAY STATION HAVE BEEN USED'/
     116X,'IN THE COMPUTATIONS,THE ARRIVAL TIME GIVEN RELATES TO'/
     116X,'THE ARRAY REFERENCE POINT.'
     1/16X,'JEFFREYS AND BULLEN(1958) TRAVEL TIMES USED,SURFACE FOCI ASS
     1UMED')
      WRITE(8,6)
    6 FORMAT(/1X,'EXPLANATION OF COLUMN HEADINGS & STATEMENTS IN EVENT L
     1ISTS'/
     19X,'AMP.',7X,'AMPLITUDE (NANOMETRES 1/2 PEAK TO PEAK)'/
     19X,'PER.',7X,'PERIOD (SECONDS)'/
     19X,'REGION',5X,'REGION NAME (FLINN & ENGDAHL)'/
     19X,'LAT.',7X,'LATITUDE OF EPICENTRE (DEGREES)'/
     19X,'LONG.',6X,'LONGITUDE OF EPICENTRE (DEGREES)'/
     19X,'MB',9X,'BODY WAVE MAGNITUDE (GUTENBERG & RICHTER 1956)'/
     19X,'MS',9X,'SURFACE WAVE MAGNITUDE (MARSHALL & BASHAM 1972)')
      WRITE(8,70)
   70 FORMAT(9X,'NO DATA',4X,'NO EVENTS REPORTED BY STATION ANALYST. THI
     1S NORMALLY REFERS'/
     120X,'TO THE 24 HR PERIOD COMMENCING AT 0000 HRS OF THE DATE'/
     120X,'GIVEN UNLESS OTHERWISE STATED IN THE LIST'/
     19X,'OUTAGE A   USABLE VISUAL RECORD NOT AVAILABLE TO ANALYST MAGNE
     1TIC TAPE'/
     120X,'RECORD NORMALLY AVAILABLE IN DUE COURSE.'/
     19X,'OUTAGE B   SYSTEM DOWNTIME,NO RECORDS(VISUAL OR MAG.TAPE)AVAIL
     1ABLE')
      WRITE(8,7)
    7 FORMAT(/1X,'STATION REFERENCE POINTS FOR'6X,'INSTRUMENT CONFIGURAT
     1IONS FOR ARRIVAL TIME'/1X,'REPORTED ARRIVAL TIME DATA',8X'DATA REP
     1ORTS')
      WRITE(8,8)
    8 FORMAT(/1X,'WOL  51 18 46N   01 13 22W',8X,'SINGLE SHORT PERIOD SE
     1ISMOMETER')
      WRITE(8,9)
    9 FORMAT(1X,'EKA  55 19 59N   03 09 33W',8X,'UNPHASED SUM OF 8 SHORT
     1 PERIOD SEISMOMETERS')
      WRITE(8,10)
   10 FORMAT(1X,'GBA  13 36 15N   77 26 10E',8X,'SINGLE SHORT PERIOD SEI
     1SMOMETER')
      WRITE(8,12)
   12 FORMAT(1X,'WRA  19 56 52S  134 21 03E',8X,'SINGLE SHORT PERIOD SEI
     1SMOMETER')
      WRITE(8,13)
   13 FORMAT(1X,'YKA  62 29 36N  114 36 19W',8X,'PHASED SUM OF 20 SHORT
     1PERIOD SEISMOMETERS',/)
      WRITE(8,18)
   18 FORMAT(/1X,'DATA ARE SUPPLIED BY COURTESY OF THE FOLLOWING ORGANIS
     1ATIONS ')
      WRITE(8,19)
   19 FORMAT(/2X,'GBA DATA  -  BHABHA ATOMIC RESEARCH CENTRE, SEISMOLOGY
```

```
     1 SECTION, MOD.LABS.,'/15X,'TROMBAY, BOMBAY 400 085, INDIA.')
      WRITE(8,20)
   20 FORMAT( 2X,'WRA DATA      THE AUSTRALIAN NATIONAL UNIVERSITY, RESEA
     1RCH SCHOOL OF EARTH'/15X,'SCIENCES,INSTITUTE OF ADVANCED STUDIES,P
     1O BOX4 CANBERRA,AUSTRALIA')
      WRITE(8,21)
   21 FORMAT( 2X,'YKA DATA      DEPARTMENT OF ENERGY MINES AND RESOURCES,
     1 EARTH PHYSICS BRANCH,'/15X,'DIVISION OF SEISMOLOGY & GEOTHERMAL S
     1TUDIES,OTTAWA,CANADA K1A 0Y3')
      WRITE(8,22)
   22 FORMAT(///30X,'PAGE  1'//)
      NPAG=1
C
      RETURN
      END

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


C
C****************************************************************
C
C              SUBROUTINE PROGRAM PAGES
C              ************************
C THIS IS A FORTRAN 4 PROGRAM USED TO SPACE THE DATA ON THE PRINTED
C PAGES.
C
C****************************************************************
C
C
      SUBROUTINE PAGES(M,TITLE,JDAY,NHR,MIN,SECS,AMP,PERIOD,NUMLIN
     1,NPAG,ALAMP,ALPER,NCHARN,LDAY,MON1,MON2)
C
      DIMENSION JDAY(600,6),NHR(600,6),MIN(600,6),SECS(600,6)
      DIMENSION AMP(600,6),PERIOD(600,6),LDAY(2,600,6),TITLE(7)
      DIMENSION NCHARN(10),ALAMP(6,600,6),ALPER(5,600,6)
C
      INTEGER ALAMP,ALPER
C
      REAL*8  TITLE,AMP
      REAL*16 SECS
C
C
C     (1)COUNT NUMBER OF CHANGES OF DAY.
C     (2)NOTE DAY WHICH OCCURS ON LINEND (IF ANY)
C     (3)TOTAL OF GAPS + EVENTS MUST NOT EXCEED LINEND.
C
      LINEND=76
      LINFIN=0
 2    LSTART=LINFIN+1
      LINFIN=LSTART+LINEND-1
C
      LSP1=LSTART+1
      NOGAPS=0
      NOEVTS=1
      NUMSKM=0
      LOOP=0
      DO 3 I=LSP1,LINFIN
      LOOP=LOOP+1
      IF(LOOP.EQ.NUMLIN)GO TO 100
      IF(JDAY(I-1,M).EQ.JDAY(I,M))GO TO 4
      NOGAPS=NOGAPS+1
      NOEVTS=NOEVTS+1
      IF((NOGAPS+NOEVTS).LE.LINEND)GO TO 70
      NOGAPS=NOGAPS-1
      GO TO 60
 70   IF((NOGAPS+NOEVTS).EQ.LINEND)GO TO 60
      GO TO 3
 4    NOEVTS=NOEVTS+1
 5    IF((NOGAPS+NOEVTS).EQ.LINEND)GO TO 6
 3    CONTINUE
      I=LINFIN
      GO TO 6
C
 60   NDAYRQ=JDAY(I,M)
      GO TO 61
C
 6    NDAYRQ=JDAY(I-1,M)
 61   NUTDRQ=0
      DO 7 I=LSTART,LINFIN
      IF(JDAY(I,M).EQ.NDAYRQ)GO TO 8
      NUTDRQ=NUTDRQ+1
 7    CONTINUE
C
 8    NUMSKM=NOEVTS-NUTDRQ
 30   LINFIN=LINFIN-NUMSKM-NOGAPS
      NUMLIN=NUMLIN-(LINFIN-LSTART)-1
C
      CALL HEAD(M,TITLE,JDAY,NHR,MIN,SECS,AMP,PERIOD,LSTART,LINFIN
     1,NOGAPS,NPAG,ALAMP,ALPER,NCHARN,LDAY,MON1,MON2)
C
      IF(NUMLIN.EQ.0)RETURN
      GO TO 2
C
 100  LINFIN=LSTART+NUMLIN-1
      IF((NUMLIN+NOGAPS).GT.LINEND)GO TO 18
C
      CALL HEAD(M,TITLE,JDAY,NHR,MIN,SECS,AMP,PERIOD,LSTART,LINFIN
     1,NOGAPS,NPAG,ALAMP,ALPER,NCHARN,LDAY,MON1,MON2)
      RETURN
C
 18   NSPACE=0
```

```
      NOEVTS=0
      I=LSP1
 40   IF(JDAY(I-1,M).EQ.JDAY(I,M))GO TO 9
      NSPACE=NSPACE+1
      NOEVTS=NOEVTS+1
      IF((NOEVTS+NSPACE).EQ.LINEND)GO TO 12
      GO TO 14
 9    NOEVTS=NOEVTS+1
      IF((NOEVTS+NSPACE).EQ.LINEND)GO TO 11
 14   I=I+1
      GO TO 40
C
 11   NDAYRQ=JDAY(I-2,M)
      NUTDRQ=0
      DO 13 I=LSTART,LINFIN
      IF(JDAY(I,M).EQ.NDAYRQ)GO TO 28
      NUTDRQ=NUTDRQ+1
 13   CONTINUE
C
 28   NUMSKM=NOEVTS-NUTDRQ
 12   LINFIN=LINFIN-NUMSKM-NSPACE-1
      NUMLIN=NUMLIN-(LINFIN-LSTART)-1
C
      CALL HEAD(M,TITLE,JDAY,NHR,MIN,SECS,AMP,PERIOD,LSTART,LINFIN
     1,NOGAPS,NPAG,ALAMP,ALPER,NCHARN,LDAY,MON1,MON2)
C
      IF(NUMLIN.EQ.0)GO TO 31
      LSTART=LINFIN+1
      LINFIN=LSTART+LINEND-1
      LSP1=LSTART+1
      NSPACE=0
      NOEVTS=0
      NUMSKM=1
      GO TO 100
C
 31   RETURN
      END

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


C
C*****************************************************************
C
C              SUBROUTINE PROGRAM HEAD
C              ***********************
C THIS IS A FORTRAN 4 PROGRAM USED TO PRINT THE DATA CONTAINED IN
C PART 1 OF THE EVENT LIST.
C
C*****************************************************************
C
C
      SUBROUTINE HEAD(M,TITLE,JDAY,NHR,MIN,SECS,AMP,PERIOD,LSTART,LINFIN
     1,NOGAPS,NPAG,ALAMP,ALPER,NCHARN,LDAY,MON1,MON2)
C
      DIMENSION JDAY(600,6),NHR(600,6),MIN(600,6),SECS(600,6)
      DIMENSION AMP(600,6),PERIOD(600,6),LDAY(2,600,6),TITLE(7)
      DIMENSION ALAMP(6,600,6),ALPER(5,600,6),NCHARN(10)
C
      INTEGER ALAMP,ALPER,SBLANK,AMON,BMON,C1,C2
C
C
      DATA SBLANK/4H    /
C
      REAL*8  TITLE,AMP
      REAL*16 SECS
C
      NPAG=NPAG+1
      WRITE (8,17)TITLE(5)
 17   FORMAT(///,48X,16HBULLETIN NUMBER ,A8)
      WRITE (8,30)
 30   FORMAT(48X,21H---------------------)
      WRITE (8,11)(TITLE(I),I=1,4)
 11   FORMAT(1X,'PART 1  EVENT LISTS FOR PERIOD ',4A8)
     1----------)
      WRITE (8,32)
 32   FORMAT(1X,63H-----------------------------------------------------
      GO TO(100,200,300,100,500,600),M
 100  WRITE (8,13)
 13   FORMAT(3X,11HSTATION WOL)
      GO TO 20
 200  WRITE (8,21)
 21   FORMAT(3X,11HSTATION EKA)
      GO TO 20
 300  WRITE (8,24)
 24   FORMAT(3X,11HSTATION GBA)
      GO TO 20
 500  WRITE (8,25)
 25   FORMAT(3X,11HSTATION YKA)
      GO TO 20
 600  WRITE (8,26)
 26   FORMAT(3X,11HSTATION WRA)
C
 20   WRITE (8,31)
 31   FORMAT(3X,11H-----------)
      WRITE (8,70)
 70   FORMAT(7X,10HDAY   TIME,21X,3HAMP,9X,3HPER)
      WRITE (8,71)
 71   FORMAT(11X,9HHRMN SECS)
      KEY1=0
      C1=NCHARN(1)
      C2=NCHARN(2)
```

```
      AMON=SBLANK
      BMON=SBLANK
      I=LSTART
      IF(AMP(I,M).EQ.0.0.AND.PERIOD(I,M).EQ.0.0)GO TO 14
      IF(LDAY(1,I,M).EQ.C1.AND.LDAY(2,I,M).EQ.C2)BMON=MON2
      WRITE (8,2)BMON,(LDAY(N,I,M),N=1,2),NHR(I,M),MIN(I,M),SECS(I,M)
     1,(ALAMP(N,I,M),N=1,6),(ALPER(N,I,M),N=1,5)
 2    FORMAT(/3X,A4,1X,2A1,1X,A2,A2,A16,4X,6A1,8X,5A1)
      GO TO 12
 14   WRITE (8,2)BMON,(LDAY(N,I,M),N=1,2),NHR(I,M),MIN(I,M),SECS(I,M)
C
 12   IF(LSTART.EQ.LINFIN)GO TO 35
      LSP1=LSTART+1
      DO 10 I=LSP1,LINFIN
C-----CHECK THAT NO DAYS HAVE BEEN MISSED OUT .
      NA=JDAY(I-1,M)
      NB=JDAY(I,M)
      IF((NA-NB).LT.2)GO TO 33
      IF((NA.EQ.28.OR.NA.EQ.29.OR.NA.EQ.30.OR.
     1NA.EQ.31).AND.NB.EQ.1)GO TO 33
      WRITE (8,40)
 40   FORMAT(2X,5H*****,1X,'AT LEAST A DAY HAS BEEN MISSED OUT ******')
C
 33   IF(BMON.EQ.MON2)KEY1=1
      BMON=SBLANK
      IF(LDAY(1,I,M).EQ.C1.AND.LDAY(2,I,M).EQ.C2.AND.KEY1.EQ.0)BMON=MON2
      IF(NA.EQ.NB)GO TO 3
      GO TO 5
 3    IF(AMP(I,M).EQ.0.0.AND.PERIOD(I,M).EQ.0.0)GO TO 1
      WRITE (8,7)BMON,(LDAY(N,I,M),N=1,2),NHR(I,M),MIN(I,M),SECS(I,M)
     1,(ALAMP(N,I,M),N=1,6),(ALPER(N,I,M),N=1,5)
 7    FORMAT(3X,A4,1X,2A1,1X,A2,A2,A16,4X,6A1,8X,5A1)
      GO TO 10
 1    WRITE (8,7)BMON,(LDAY(N,I,M),N=1,2),NHR(I,M),MIN(I,M),SECS(I,M)
      GO TO 10
 5    IF(AMP(I,M).EQ.0.0.AND.PERIOD(I,M).EQ.0.0)GO TO 9
      WRITE (8,6)BMON,(LDAY(N,I,M),N=1,2),NHR(I,M),MIN(I,M),SECS(I,M)
     1,(ALAMP(N,I,M),N=1,6),(ALPER(N,I,M),N=1,5)
 6    FORMAT(/3X,A4,1X,2A1,1X,A2,A2,A16,4X,6A1,8X,5A1)
      GO TO 10
 9    WRITE (8,6)BMON,(LDAY(N,I,M),N=1,2),NHR(I,M),MIN(I,M),SECS(I,M)
 10   CONTINUE
C
 35   NBLANK=89-14-(LINFIN-LSTART)-NOGAPS-1
      DO 18 I=1,NBLANK
      WRITE (8,19)
 19   FORMAT(10X)
 18   CONTINUE
      WRITE (8,22)NPAG
 22   FORMAT(25X,5HPAGE ,I2,//)
C
      RETURN
      END
```

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

```
C
C*****************************************************************
C
C                    SUBROUTINE PROGRAM PRELIM
C                    *************************
C THIS IS A FORTRAN 4 PROGRAM USED TO SELECT DATA FROM THE BLACKNEST
C DATABASE FILE CONTAINING LOCATED EVENTS AND PRINT IT AS PART 2 OF
C THE EVENT LIST.
C
C*****************************************************************
C
C
      SUBROUTINE PRELIM(TITLE,L1,L2,NPAG)
C
      DIMENSION ALPHAD(80,400),TITLE(7)
C
      INTEGER ALPHAD,ATEND,STREM6
C
      REAL*8 TITLE
C
C-----CAN OBTAIN PRINT-OUT (I.E. IN STREAM 6) - FOR DEBUG PURPOSES -
C-----BY SETTING PARAMETER STREM6=1 .
      STREM6=0
C
      NPAG=NPAG+1
C
C-----CHECK SUBMITTED LINE NUMBERS IF ANY
      IF(L1.EQ.0.AND.L2.EQ.0)GO TO 1
      GO TO 20
 1    WRITE (8,3)
 3    FORMAT(/9X,'NO EVENTS IN DEFINED CATEGORY')
C-----REMOVE UNWANTED DATA--I.E. CONTENTS OF PRELIDET FILE.
 999  J=1
 5    READ 4,(ALPHAD(I,J),I=1,80)
 4    FORMAT(80A1)
      IF(STREM6.EQ.1)WRITE (6,7)(ALPHAD(I,J),I=1,79)
      IF(ALPHAD(9,J).EQ.0.AND.ALPHAD(10,J).EQ.0)GO TO 23
      J=J+1
      GO TO 5
C
 20   IF(L1.GE.1.AND.L2.GT.L1)GO TO 6
      WRITE (6,40)L1,L2
 40   FORMAT(2X,'LINE COUNT ERROR',5X,I5,5X,I5)
      GO TO 999
C-----HOW MANY LINES OF DATA FROM PRELIDET ARE TO BE USED
```

```
C-----AT PRESENT PUT 5 EVENTS PER PAGE I.E. TOTAL OF 60 LINES
C-----OF DATA.
C
 6      IF(L2.GE.11)GO TO 43
        WRITE (6,40)L1,L2
        GO TO 999
C-----PUT TITLE ON FIRST PAGE OF PART 2 OF EVENT LIST.
 43     WRITE (8,17)TITLE(5)
 17     FORMAT(///,48X,16HBULLETIN NUMBER ,A8)
        WRITE (8,30)
 30     FORMAT(48X,21H---------------------)
        WRITE (8,11)(TITLE(I),I=1,4)
 11     FORMAT(1X,'PART 2  ANALYSIS OF SELECTED EVENTS ',4A8)
        WRITE (8,32)
 32     FORMAT(1X,68H-----------------------------------------------------
       1---------------)
C
        ATEND=0
C-----IF L2 LCONTR OR LESS THEN ONLY 1 PAGE WITH WHICH
C-----TO DEAL.
        IF(L1.EQ.1)GO TO 51
C-----IF L1 NOT EQUAL TO 1 THEN REMOVE INITIAL PORTION OF PRELIDET
C-----FILE NOT REQUIRED.
        J=1
 50     READ 4,(ALPHAD(I,J),I=1,80)
        IF(STREM6.EQ.1)WRITE (6,7)(ALPHAD(I,J),I=1,79)
        IF(ALPHAD(9,J).EQ.0.AND.ALPHAC(10,J).EQ.0)GO TO 23
        IF(J.EQ.(L1-1))GO TO 51
        J=J+1
        GO TO 50
C
 51     J=L1
        LCONTR=60
        LCNT=L2-L1+1
 15     KNT=0
C-----WRITE A PAGE OF PART 2 OF EVENT LIST.
 9      READ 4,(ALPHAD(I,J),I=1,80)
        IF(STREM6.EQ.1)WRITE (6,7)(ALPHAD(I,J),I=1,79)
        KNT=KNT+1
        IF(ALPHAD(9,J).EQ.0.AND.ALPHAC(10,J).EQ.0)GO TO 10
        WRITE (8,7)(ALPHAD(I,J),I=1,79)
 7      FORMAT(1X,79A1)
        LCNT=LCNT-1
        IF(KNT.EQ.LCONTR.AND.LCNT.NE.0)GO TO 8
        IF(LCNT.GT.0)GO TO 60
        GO TO 10
 60     J=J+1
        GO TO 9
C
C-----LESS THAN 1 PAGE SO FILL OUT WITH BLANK LINES.
 10     NOBLAN=LCONTR-KNT
        ATEND=1
        IF(NOBLAN.GE.LCONTR)GO TO 23
        IF(NOBLAN.LT.0)GO TO 23
        DO 70 I=1,NOBLAN
        WRITE (8,12)
 12     FORMAT(10X)
 70     CONTINUE
        GO TO 8
C
C-----PUT ON PAGE NUMBER--CHECK ANY MORE PAGES  YES-WRITE TITLES,
C-----NO-FINISH.
 8      WRITE (8,22)NPAG
 22     FORMAT(25X,5HPAGE ,I2,//)
        IF(ATEND.EQ.1)GO TO 23
        IF(L2.EQ.LCONTR)GO TO 23
        NPAG=NPAG+1
        IF(LCNT.LE.0)GO TO 23
        J=J+1
C-----PUT TITLES ON THE NEXT PAGE.
        WRITE (8,17)TITLE(5)
        WRITE (8,30)
        WRITE (8,11)(TITLE(I),I=1,4)
        WRITE (8,32)
        GO TO 15
C
 23     RETURN
        END

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

C
C*****************************************************************
C
C                 SUBROUTINE PROGRAM CHARAI
C                 *************************
C THIS IS A FORTRAN 4 PROGRAM USED TO INTERPRET SETS OF ALPHA CHARACTERS
C INTO NUMBERS.
C
C*****************************************************************
C
C
        SUBROUTINE CHARAI(ALPHAC,NCHARN,J,JTOT,RV,IV)
C
        DIMENSION ALPHAC(24),NCHARN(10),ICHAR(8)
C
        DATA COMMA/4H,   /,DECIPT/4H.   /,SBLANK/4H    /
C
        INTEGER ALPHAC,COMMA,DECIPT,SBLANK
C
        REAL*8 RV,FRACT
```

```
C
      RV=0.0
      IV=9999
C
C-----IDENTIFY CHARACTERS UP TO DECIMAL POINT.
      IPART=0
      DO 5 L=1,JTOT
      JPLM1=J+L-1
      IF(ALPHAC(JPLM1).EQ.DECIPT)GO TO 2
C-----IDENTIFY CHARACTER.
      DO 1 I=1,10
      IF(ALPHAC(JPLM1).EQ.NCHARN(I))ICHAR(L)=I-1
 1    CONTINUE
 5    CONTINUE
C
 2    LM1=L-1
      DO 6 I=1,LM1
      K=LM1-I
      IPART=IPART+10**K*ICHAR(I)
 6    CONTINUE
C
      IF(ALPHAC(JPLM1).EQ.DECIPT)GO TO 8
C
C-----INTEGER NUMBER.
      IV=IPART
      RETURN
C
C-----DECIMAL POINT BUT NOTHING AFTER IT.
 3    JPL=J+L
      IF(ALPHAC(JPL).NE.SBLANK.OR.ALPHAC(JPL).NE.COMMA)GO TO 3
      RV=FLOAT(IPART)
      RETURN
C
C-----REAL NUMBER
 8    ICHAR(L)=DECIPT
      LDEC=L
      FRACT=0.0
      JPLEND=J+JTOT-1
      DO 4 K=JPL,JPLEND
      L=L+1
C-----IDENTIFY REMAINING CHARACTERS.
      DO 7 I=1,10
      IF(ALPHAC(K).EQ.NCHARN(I))ICHAR(L)=I-1
 7    CONTINUE
 4    CONTINUE
C
      L=LDEC
      DO 9 I=JPL,JPLEND
      L=L+1
      LP=L-LDEC
      FRACT=FRACT+0.1**LP*FLOAT(ICHAR(L))
 9    CONTINUE
C
      RV=FLOAT(IPART)+FRACT
C
      RETURN
      END
```

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

## APPENDIX C

## HOUSEKEEPING PROGRAMS

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

THE HOUSEKEEPING PROGRAMS

| | SUB-DIRECTORY NAME | ELECTRIC FILE NAME |
|---|---|---|
| A.(JOB CONTROL + ELECTRIC) PROGRAMS. | | |
| (1)CLEAROUT | IN | CLEAROUT |
| (2)CLEAROUT.ED | IN | CLEAROUT.ED |
| (3)CLEAROUT* | MASTER | CLEARO50 |
| (4)CLEAROUT.ED* | MASTER | CLEARO50.ED |
| (5)UPDATE | IN | UPDATE |
| (6)UPDATE.ED | IN | UPDATE.ED |
| (7)UPDATE* | MASTER | UPDATE50 |
| (8)UPDATE.ED* | MASTER | UPDATE50.ED |
| (9)ORDERSUB | HOUSEKEP | ORDERSUB |
| (10)ORDERSUB.ED | HOUSEKEP | ORDERSUB.ED |
| (11)ORDERSUB* | MASTER | ORDERS50 |
| (12)ORDERSUB.ED* | MASTER | ORDERS50.ED |
| (13)RELOAD | HOUSEKEP | RELOAD |
| (14)RELOAD.ED | HOUSEKEP | RELOAD.ED |
| (15)REMOVE | HOUSEKEP | REMOVE |
| (16)TESTFORT | HOUSEKEP | TESTFORT |
| (17)TESTFORT.ED | HOUSEKEP | TESTFORT.ED |
| (18)MODMEMUL | HOUSEKEP | MODMEMUL |
| (19)MODMEMUL.ED | HOUSEKEP | MODMEMUL.ED |
| (20)EXPROGUL | HOUSEKEP | EXPROGUL |
| (21)EXPROGUL.ED | HOUSEKEP | EXPROGUL.ED |
| (22)LEDICMEM | HOUSEKEP | LEDICMEM |
| (23)LEDICMEM.ED | HOUSEKEP | LEDICMEM.ED |

NOTES * INDICATES THAT THESE PROGRAMS (ARCHIVED IN SUB-DIRECTORY MASTER) ARE DUPLICATE COPIES OF THE DATABASE PROGRAMS IN ROUTINE USE.

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

```
/*PRIORITY     12
//XA3WCLOT  JOB (ACCT,IC,0-C5),WARBURTON
/*ROUTE  PRINT ELECTRIC
//*
//* ***************************************************************
//*
//*                    PROGRAM CLEARCLT
//*                    ****************
//* (360 JOB CONTROL AND ELECTRIC LANGUAGES) PROGRAM USED TO PUT DATA
//* FILES INTO ARCHIVE STORAGE FROM THE BLACKNEST DATABASE.
//* NOTE THIS ELECTRIC FILE CLEAROUT HAS A CORRESPONDING EDIT FILE
//* CLEAROUT.ED.
//*
//* EXECUTION OF THE PROGRAM IS BY SUBMITTING THE COMMAND -
//*                 ---------------------------
//*                 EXEC FL=CLEAROUT,MONTH=NOV76
//*                 ---------------------------
//*      WHERE MONTH IS NAME OF FILES TO BE ARCHIVED FROM DATABASE.
//*
//* ***************************************************************
//*
//  EXEC  ELSEND
//G.SYSIN  DD  *
LOGIN ID=ID,ACCT=ACCT,KEY=XXXX
SETD C=A.WOL
CHANGE FL=MONTH,RETDT=01/01/79
ARCHIVE FL=MONTH
SETD C=A.EKA
CHANGE FL=MONTH,RETDT=01/01/79
ARCHIVE FL=MONTH
SETD C=A.GBA
CHANGE FL=MONTH,RETDT=01/01/79
ARCHIVE FL=MONTH
SETD C=A.WRA
CHANGE FL=MONTH,RETDT=01/01/79
ARCHIVE FL=MONTH
SETD C=A.YKA
CHANGE FL=MONTH,RETDT=01/01/79
ARCHIVE FL=MONTH
LOGOUT
/* END OF JOB XA3WCLOT
```

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

CLEAROUT.ED

```
&P LB= 1,LN=  26,C1= 11,C2= 15,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  27,C1= 12,C2= 16,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  29,C1= 11,C2= 15,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  30,C1= 12,C2= 16,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  32,C1= 11,C2= 15,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  33,C1= 12,C2= 16,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  35,C1= 11,C2= 15,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  36,C1= 12,C2= 16,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  38,C1= 11,C2= 15,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  39,C1= 12,C2= 16,CH=AN,DF=NO,NM=MONTH
```

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

```
/*PRIORITY     12
//XA3WUPDT  JOB (ACCT,ID,0-C5),WARBURTON
/*ROUTE  PRINT ELECTRIC
//*
//* ****************************************************************
//*
//*                          PROGRAM UPDATE
//*                          **************
//* (360 JOB CONTROL AND ELECTRIC LANGUAGES) PROGRAM USED TO PUT NEW
//* EMPTY FILES INTO THE BLACKNEST DATABASE READY TO RECEIVE DATA.
//* NOTE  THIS ELECTRIC FILE UPDATE HAS A CORRESPONDING EDIT FILE
//* UPDATE.ED.
//*
//* EXECUTION OF THE PROGRAM IS BY SUBMITTING THE COMMAND -
//*                 -------------------------
//*                 EXEC FL=UPDATE,MONTH=DEC76
//*                 -------------------------
//*   WHERE MONTH IS THE NAME OF THE FILES TO BE CREATED IN THE DATABASE
//*
//* ****************************************************************
//*
//  EXEC  ELSEND
//G.SYSIN  DD  *
LOGIN ID=ID,ACCT=ACCT,KEY=XXXX
SETD C=A.WOL
ENTER FL=MONTH,ACCESS=FRRR,ADDPTNR=F1 P2
AAA
**
CLEAR FL=MONTH
SETD C=A.EKA
ENTER FL=MONTH,ACCESS=FRRR,ADDPTNR=F1 P2
BBB
**
CLEAR FL=MONTH
SETD C=A.GBA
ENTER FL=MONTH,ACCESS=FRRR,ADDPTNR=F1 P2
CCC
**
CLEAR FL=MONTH
SETD C=A.WRA
ENTER FL=MONTH,ACCESS=FRRR,ADDPTNR=F1 P2
EEE
**
CLEAR FL=MONTH
SETD C=A.YKA
ENTER FL=MONTH,ACCESS=FRRR,ADDPTNR=F1 P2
FFF
**
CLEAR FL=MONTH
LOGOUT
/* END OF JOB XA3WUPDT
```

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 UPDATE.ED
 ---------

&P LB= 1,LN=  26,C1= 10,C2= 14,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  29,C1= 10,C2= 14,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  31,C1= 10,C2= 14,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  34,C1= 10,C2= 14,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  36,C1= 10,C2= 14,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  39,C1= 10,C2= 14,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  41,C1= 10,C2= 14,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  44,C1= 10,C2= 14,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  46,C1= 10,C2= 14,CH=AN,DF=NO,NM=MONTH
&P LB= 1,LN=  49,C1= 10,C2= 14,CH=AN,DF=NO,NM=MONTH

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
/*PRIORITY     12
//XA3WORDR JOB (ACCT,ID),BLAMEY
/*ROUTE  PRINT ELECTRIC
//*
//* ****************************************************************
//*
//*                    PROGRAM ORDERSUB
//*                    ****************
//* (360 JOB CONTROL AND ELECTRIC LANGUAGES) PROGRAM USED TO ORDER
//* FILES IN SUB-DIRECTORIES IN THE BLACKNEST DATABASE INTO
//* ALPHABETICAL ORDER AS AND WHEN REQUIRED.
//* NOTE THIS ELECTRIC FILE ORDERSUB HAS A CORRESPONDING EDIT FILE
//* ORDERSUB.ED.
//*
//* ---------------------------------------------------------------
//*
//* RESTRICTIONS ON THE USE OF THE PROGRAM.
//* ---------------------------------------
//*  (1)THE FILES IN THE SUB-DIRECTORY MUST NOT BE IN A MIXED
//* SITUATION I.E. SOME IMMEDIATELY ACCESSIBLE , SOME ARCHIVED
//* OR SOME IMMEDIATELY ACCESSIBLE , SOME BEING RESTORED
//* OTHERWISE WHEN THE PROGRAM IS RUN SOME OF THE FILES WILL
//* BE IN THE ORIGINAL AND SOME IN THE TEMPORARY DIRECTORIES.
//*  ALL FILES MUST EITHER BE ACCESSIBLE OR ALL FILES MUST BE
//* ARCHIVED.
//*
//* (2)THIS FACILITY CAN BE USED FOR RE-ORDERING A MAIN
//* DIRECTORY BUT ONLY BY THE SYSTEMS MANAGER .
//* A MEMBER OF THE USER SUPPORT GROUP SHOULD BE
//* CONTACTED IN THIS CASE.
//*
```

```
//* (3)IF FOR SOME REASON THE ORDER COMMAND IS NOT COMPLETED
//* THEN CONSULT THE ELECTRIC MANUAL PART IV 1.10
//*
//* ------------------------------------------------------------
//*
//* EXECUTION OF THE PROGRAM IS BY SUBMITTING THE COMMAND -
//*                 ---------------------------------
//*                 EXEC FL=ORDERSUB,DIRECTOR=HOUSEKEP
//*                 ---------------------------------
//*   WHERE DIRECTOR IS THE NAME OF THE SUB-DIRECTORY TO BE ORDERED.
//*
//* ************************************************************
//*
// EXEC ELCIRE
//G.SYSIN  DD  *
LOGIN ID=ID,ACCT=ACCT,KEY=XXXX
ORDER DR1=A.DIRECTOR,DR2=TEMPDIRO
LOGOUT
/* END OF JOB XA3WORDR
```

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 ORDERSUB.ED
 -----------

$P LB= 1,LN=  48,C1= 13,C2= 20,CH=AL,DF=NO,NM=DIRECTOR
```

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
/*PRIORITY     8
//XA3WRELL  JOB (ACCT,ID,0-40),BLAMEY
/*ROUTE  PRINT ELECTRIC
//*
//* ****************************************************************
//*
//*                         PROGRAM RELOAD
//*                         **************
//* (360 JOB CONTROL AND ELECTRIC LANGUAGES) PROGRAM WHICH WILL COMPILE
//* FORTRAN PROGRAMS INTO BINARY FORM AND PLACE THEM IN THE COMPUTER
//* LIBRARY. THIS PARTICULAR PROGRAM REFERS EXCLUSIVELY TO THE
//* DEVELOPEMENT OF THE LOADER PROGRAM. TO PUT A NEW BINARY
//* PROGRAM VERSION OF THE FORTRAN PROGRAM INTO THE COMPUTER
//* LIBRARY (WITHOUT OVERWRITING THE PREVIOUS PROGRAM) A DIFFERENT
//* MEMBER NAME IS USED.
//* NOTE  THIS ELECTRIC FILE RELOAD HAS A CORRESPONDING EDIT FILE
//* RELOAD.ED.
//*
//* EXECUTION OF THE PROGRAM IS BY SUBMITTING THE COMMAND -
//*                 ---------------------------------------
//*                 EXEC FL=RELOAD,F1=    ,F2=    ,F3=     .
//*                 ---------------------------------------
//*   WHERE F1,F2,F3 ARE THE NAMES OF THE FILES CONTAINING THE FORTRAN
//*   PROGRAMS.
//*
//* ****************************************************************
//*
// EXEC FHCL,LIBRARY='ULIB.ISC',MEMBER=XA3WUP50,REGION=300K
//C.SYSIN DD *
// EXEC FHCL,LIBRARY='ULIB.ISC',MEMBER=XA3WDC50
//C.SYSIN DD *
// EXEC FHCL,LIBRARY='ULIB.ISC',MEMBER=XA3WLI50
//C.SYSIN DD *
//L.SYSIN DD *
  INCLUDE LIBRARY(XA3WUP50,XA3WDC50,XA3WLI50)
  ENTRY MAIN
/* END OF JOB XA3WRELL
```

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 RELOAD.ED
 ---------

$S LN=  29.1,DF=NO,NM=F1
$S LN=  31.1,DF=NO,NM=F2
$S LN=  33.1,DF=NO,NM=F3
```

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
/*PRIORITY     12
//XA3WREMV JOB (ACCT,ID),BLAMEY
/*ROUTE  PRINT ELECTRIC
//*
//* ****************************************************************
//*
//*                         PROGRAM REMOVE
//*                         **************
//* THIS IS 360 JOB CONTROL PROGRAM WHICH SHOWS HOW TO DELETE
//* BINARY PROGRAMS HELD IN THE COMPUTER LIBRARY FROM THAT LIBRARY.
//*
//* EXECUTION OF THE PROGRAM IS BY SUBMITTING THE COMMAND -
//*                 --------------
//*                 EXEC FL=REMOVE
//*                 --------------
//*   NOTE - MODIFY THE NAMES OF THE MEMBERS TO THE PARTICULAR
//*          REQUIREMENTS NEEDED.
//*
//* ****************************************************************
```

```
//*
// EXEC  REMOVE,LIBRARY='ULIB.ISC',
// MEMBER='XA3WUP50,XA3WUC5C,XA3WLI50'
/* END OF JOB XA3WREMV
```

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

```
/*PRIORITY      12
//XA3WTFOR  JOB (ACCT,ID,0-10),BLAMEY
/*ROUTE  PRINT ELECTRIC
//*
//* ****************************************************************
//*                      PROGRAM TESTFORT
//*                      ================
//*  TO EXECUTE A PROGRAM FROM A USER LIBRARY SUPPLYING SOME MODIFIED
//*  FORTRAN 4 ROUTINES.
//*       THIS (360 JOB CONTROL AND ELECTRIC LANGUAGES) PROGRAM COMPILES
//*  (H LEVEL) THE MODIFIED ROUTINES FOR WHICH SOURCE CARDS (IMAGES)
//*  HAVE BEEN SUPPLIED. THE LINKAGE-EDITOR THEN OBTAINS THE REST OF
//*  THE PROGRAM FROM THE MEMBER 'MYPROG' ON 'ULIB.ISC'.
//*  ====================================
//*  NOTE  THAT 'MYPROG' REMAINS UNCHANGED.
//*  ====================================
//*
//*  THIS FILE HAS A CORRESPONDING EDIT FILE TO SUPPLY FORTRAN
//*  ROUTINES AFTER THE EXEC STATEMENT , A PLANT FOR THE NAME 'MYPROG'
//*  AND POSSIBLY DATA (ALTHOUGH THERE IS A DEFAULT VALUE ON THIS)
//*  AFTER THE //G.SYSIN CARD.
//*
//*  THE EXEC STATEMENT IS OF THE FORM -
//*      ----------------------------------------------------
//*      EXEC FL=TESTFORT,MYPROG=     ,F1=       ,JOBDATA=       .
//*      ----------------------------------------------------
//*               FL=JOBDATA MAY BE OMITTED.
//*
//*  NOTE  MORE THAN ONE FORTRAN PROGRAM CAN BE PUT IN FILE F1
//*        (E.G. A MAIN PROGRAM + SUBROUTINES)
//*
//* ****************************************************************
//*
// EXEC FHCLG,REGION.G=72K
//L.LIB DD DSN=ULIB.ISC,DISP=SHR
//L.SYSIN DD *
  INCLUDE LIB(MYPROG)
  ENTRY MAIN
//G.SYSIN DD *
/* END OF JOB XA3WTFOR
```

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

```
  TESTFORT.ED
  -----------

aS LB= 1,LN=  34,DF=NO,NM=F1
aP LB= 1,LN=  37,C1= 15,C2= 20,CH=NC,DF=NO,NM=MYPROG
aS LB= 1,LN=  39,DF=YS,NM=JOBDATA
```

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

```
/*PRIORITY      12
//XA3WMMUL  JOB (ACCT,ID,0-10),BLAMEY
/*ROUTE  PRINT ELECTRIC
//*
//* ****************************************************************
//*                      PROGRAM MODMEMUL
//*                      ================
//*  TO MODIFY A MEMBER OF A USER LIBRARY.
//*       THIS (360 JOB CONTROL AND ELECTRIC LANGUAGES) PROGRAM WILL
//*  COMPILE THE FORTRAN 4 ROUTINES PROVIDED (H LEVEL), AND ALSO INCLUDE
//*  OTHER ROUTINES FROM THE EXISTING VERSION OF 'OLDPROG' IN THE
//*  LIBRARY AND WRITE THE PROGRAM BACK TO THE LIBRARY WITH THE NAME
//*  'NEWPROG'.
//*
//*   IF THE NAMES 'OLDPROG' AND 'NEWPROG' ARE THE SAME THEN THE
//*  OLD VERSION OF THE PROGRAM IS OVERWRITTEN. IF THE NAMES ARE
//*  DIFFERENT THEN THE OLD VERSION WILL BE RETAINED.
//*
//*  THIS FILE HAS A CORRESPONDING EDIT FILE TO SUPPLY FORTRAN
//*  ROUTINES AFTER THE //C.SYSIN CARD, PLANTS FOR THE NAMES OLDPROG
//*  AND NEWPROG AND POSSIBLY SUPPLY OF DATA --ALTHOUGH THERE IS A
//*  A DEFAULT ON THE DATA--- AFTER THE //G.SYSIN CARD.
//*
//*  THE EXEC STATEMENT IS OF THE FORM -
//*  (NOTE  IF USER IS CREATING A NEW MEMBER TO PUT INTO LIBRARY THEN
//*   PARAMETER OLDPROG CAN BE SET TO WHATEVER NEWPROG IS BEING SET.
//*   ALSO IT IS NECESSARY TO CONSTRUCT THE NAME OF THE MEMBER
//*   TO A SPECIFIC FORMAT.)
//*
//*  ---------------------------------------------------------------
//*  EXEC FL=MODMEMUL,OLDPROG=     ,NEWPROG=    ,F1=     ,JOBDATA=       .
//*  ---------------------------------------------------------------
//*               FL=JOBDATA MAY BE OMITTED.
//*
//*  NOTE  MORE THAN ONE FORTRAN PROGRAM CAN BE PUT IN FILE F1
//*        (E.G. A MAIN PROGRAM + SUBROUTINES)
//*
//* ****************************************************************
//*
```

```
// EXEC FHCLG,LIBRARY='ULIB.ISC',MEMBER=NEWPROG,REGION.G=72K
//C.SYSIN DD *
//L.LIB DD DSN=ULIB.ISC,DISP=SHR
//L.SYSIN DD *
 INCLUDE LIBRARY(OLDPROG)
 ENTRY MAIN
//G.SYSIN DD *
/* END OF JOB XA3WMMUL

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  MODMEMUL.ED
  -----------


&P LB= 1,LN=  40,C1= 41,C2= 47,CH=NC,DF=NO,NM=NEWPROG
&S LB= 1,LN=  41,DF=NO,NM=F1
&P LB= 1,LN=  44,C1= 18,C2= 24,CH=NC,DF=NO,NM=CLOPROG
&S LB= 1,LN=  46,DF=YS,NM=JOBDATA

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


/*PRIORITY     12
//XA3WEPUL  JOB (ACCT,ID,0-10),BLAMEY
/*ROUTE  PRINT ELECTRIC
//* ***********************************************************
//*                        PROGRAM EXPROGUL
//*                        ================
//* TO EXECUTE A PROGRAM FROM A USER LIBRARY.
//*
//*  THIS IS A 360 JOB CONTROL AND ELECTRIC LANGUAGES PROGRAM
//*  THIS FILE HAS A CORRESPONDING EDIT FILE WHICH MUST PROVIDE
//*  A PLANT FOR THE NAME 'MYPROG' IN THE EXEC CARD AND ALSO
//*  DATA AFTER THE //G.SYSIN CARD.
//*
//*  THE EXEC STATEMENT IS OF THE FORM -
//*      ---------------------------------------------
//*      EXEC PL=EXPROGUL,MYPROG=       ,JOBDATA=      .
//*      ---------------------------------------------
//*
//* ***********************************************************
//*
// EXEC JOBLIB,LIBRARY='ULIB.ISC',MEMBER=MYPROG,REGION.G=72K
//G.SYSIN DD *
/* END OF JOB XA3WEPUL

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  EXPROGUL.ED
  -----------


&P LB= 1,LN=  21,C1= 42,C2= 47,CH=NC,DF=NO,NM=MYPROG
&S LB= 1,LN=  22,DF=NO,NM=JOBDATA

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


/*PRIORITY     12
//XA3WLEIM  JOB (ACCT,ID,0-10),BLAMEY
/*ROUTE  PRINT ELECTRIC
//*
//* ****************************************************************
//*                         PROGRAM LEDICMEM
//*                         ================
//*  THIS (360 JOB CONTROL AND ELECTRIC LANGUAGES) PROGRAM SHOWS THE
//*  METHOD OF LINK-EDITING TO-GETHER THREE  INCOMPLETE  MEMBERS
//*  MYPROG1 , MYPROG2 , MYPROG3 AND EXECUTING THE RESULTANT MODULE.
//*
//*  THIS FILE HAS A CORRESPONDING EDIT FILE TO PLANT NAMES
//*  FOR MYPROG1 ETC. AND TO SUPPLY DATA AFTER THE //G.SYSIN
//*  CARD.
//*
//*  THE EXEC STATEMENT IS OF THE FORM -
//*    ------------------------------------------------------------
//*    EXEC PL=LEDICMEM,MYPROG1=     ,MYPROG2=     ,MYPROG3=    ,JOBDATA= .
//*    ------------------------------------------------------------
//*
//* ****************************************************************
//*
// EXEC FHLG,REGION.G=72K
//L.LIB DD DSN=ULIB.ISC,DISP=SHR
//L.SYSIN DD *
  INCLUDE LIB(MYPROG1)
  INCLUDE LIB(MYPROG2)
  INCLUDE LIB(MYPROG3)
//G.SYSIN DD *
/* END OF JOB XA3WLEIM

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  LEDICMEM.ED
  -----------


&P LB= 1,LN=  26,C1= 15,C2= 21,CH=NC,DF=NO,NM=MYPROG1
&P LB= 1,LN=  27,C1= 15,C2= 21,CH=NC,DF=NO,NM=MYPROG2
&P LB= 1,LN=  28,C1= 15,C2= 21,CH=NC,DF=NO,NM=MYPROG3
&S LB= 1,LN=  29,DF=NO,NM=JOBDATA

 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

DOCUMENT CONTROL SHEET

Overall security classification of sheet .....Unclassified.....

(As far as possible this sheet should contain only unclassified information. If it is necessary to enter classified information, the box concerned must be marked to indicate the classification eg (R), (C) or (S)).

| 1. DRIC Reference (if known) | 2. Originator's Reference | 3. Agency Reference | 4. Report Security Classification |
|---|---|---|---|
| - | AWRE Report No. 037/76 | - | On Sale |

| 5. Originator's Code (if known) | 6. Originator (Corporate Author) Name and Location |
|---|---|
| - | Atomic Weapons Research Establishment, Aldermaston, Berkshire |
| **5a. Sponsoring Agency's Code (if known)** | **6a. Sponsoring Agency (Contract Authority) Name and Location** |
| - | - |

7. Title

The Establishment of the Blacknest Seismological Database on the Rutherford Laboratory System 360/195 Computer

7a. Title in Foreign Language (in the case of Translation)

-

7b. Presented at (for Conference Papers). Title, Place and Date of Conference

-

| 8. Author 1.Surname, Initials | 9a. Author 2 | 9b. Authors 3, 4 .... | 10. Date | pp | ref |
|---|---|---|---|---|---|
| Blamey C | - | - | February 1977 | 55 | 8 |

| 11. Contract Number | 12. Period | 13. Project | 14. Other References |
|---|---|---|---|
| - | - | - | - |

15. Distribution Statement

No Restriction

16. Descriptors (or Keywords) (TEST)

Seismological stations
Seismic arrays
Explosion effects
Seismic detection
Nuclear explosion detection
Data recorders
Data processing

Abstract

In order to assess the problems which might arise from monitoring a comprehensive test ban treaty by seismological methods, an experimental monitoring operation is being conducted. This work has involved the establishment of a database on the Rutherford Laboratory 360/195 system computer. The database can be accessed in the UK over the public telephone network and in the USA via ARPANET.