INTERNAL DOCUMENT No. 298

A user's guide to the MultiMet Sensor and Calibration Facility

R W Pascal  A L Williams & R Ahmed

1991
A user's guide to the MultiMet Sensor and Calibration Facility

R W Pascal  A L Williams & R Ahmed

1991
**Title**

A user's guide to the MultiMet Sensor Management and Calibration Facility.

**Reference**

Institute of Oceanographic Sciences Deacon Laboratory, Internal Document, No. 298, 114pp. (Unpublished manuscript)

**Abstract**

The report describes the operating instructions and procedures for the MultiMet Sensor Management and Calibration Facility. This includes a description of the Meteological database METDB, and the Sensor Management database which organises the large number of sensors required by the Multimet System. Calibration procedures and policies are also described for the various types of sensors used.
INDEX

Contents.  

Introduction  

Main Menu  

PART 1 SYSTEM FUNDAMENTALS  

1.1 METHB  

1.2 CALIBRATIONS

- Calibration Menu  
- Tank Calibration  
- Manual Input  
- Calibration Certificate  

1.3 MULTIMET SENSOR MANAGEMENT  

- Allocation Menu  
- Selecting Sensors for Deployment  
- Sensors Returning from Deployment  
- Sensors made available for Deployment  
- Dump to Printer Sensor List  
- New Sensors Entries  

PART 2 OPERATING PROCEDURES  

2.1 CHOOSING A SENSOR  

2.2 RETURNING A SENSOR  

- Psychrometers  
- SST  
- Wind Speed  
- Temperature Sensors
### 2.3 VISITING THE MET OFFICE

<table>
<thead>
<tr>
<th>Prior to visit</th>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>Items to take</td>
<td>17</td>
</tr>
<tr>
<td>On Return</td>
<td>18</td>
</tr>
</tbody>
</table>

### APPENDIX 1

- Multimet Sensor Database | 19
- Sensors ID's             | 19
- Sensor Type              | 19
- Manufactures Code        | 20
- Sensor Numbers           | 20
- File Structure           | 20
- Example List File        | 22

### APPENDIX 2

- Sensor Deployment        | 23
- Calibration Details      | 24

### APPENDIX 3

- 3.1 Calibration System   | 25
- 3.2 Multimet Sensor Management System | 26
- 3.3 METDB Database       | 27

### APPENDIX 4

- MAINMENU.BAS
- CALMENU.BAS
- TANK.BAS
- SEND.BAS
- SEN3.BAS
- WINDCAL.BAS
- TEMCAL.BAS
- CALCERT.BAS
- SENPRT.BAS
MULTIMET SENSOR MANAGEMENT
AND
CALIBRATION FACILITY

INTRODUCTION

The aim of this report is to describe and provide operating instructions for the MultiMet Sensor Management and Calibration Facility. Section 1 provides details of the system fundamentals and structure. Section 2 describes the operating procedures. An appendix provides information needed for maintaining the system including listings of the programs.

The Temperature Controlled Tank at present lives in the calibration room, where it has a good stable environment, enabling accurate calibrations. A TANDON 386 PC is used to control the tank with the addition of a STRAWBERRY TREE I/O card. The I/O card reads the data from the sensors and also controls the tank, the F25 precision thermometer is read via the RS423 port. The PC also has the prime database as it is generally updated from the tank software. It was decided that the various software programs should be integrated into one SENSOR MANAGEMENT package. This package called MultiMet Sensor Management, is a menu driven program, making it as user friendly as possible.

The Sensor Management package performs four functions :-

a) Access to METDB database
b) Controlling the Calibration Tank
c) Generation of Calibration Certificates
d) Sensor Management

All the above functions can be accessed by the use of menu driven software, which are interlinked to create one package. The first menu is automatically displayed when the PC is turned on, making sure all software is loaded correctly. It provides the following options.
Main Menu

a) Calibration Menu
This loads a second menu which gives the choice of calibration software ie tank or manual input.

b) Allocation Menu
This loads the first menu of the sensor management software.

c) Metdb
Metdb SHELLS out of GWBASIC and loads PIPEDREAM so that the database can be accessed.

d) Quit
Returns the user back to DOS.
PART 1 SYSTEM FUNDAMENTALS

1.1 METDB

Metdb is a database of meteorological sensors, it was compiled on PIPEDREAM, a word processing/spreadsheet/database program. Each sensor on the database is allocated a MultiMet number (xxx), and has the following files:

- List file - "SSMMnnnn.L"
- Details file - "Details.xxx"
- History file - "History.xxx"
- Calibration files - "SSIIYYI.xxx"

Where
- SS = Sensor type code
- MM = Manufacturer's code
- nnnn = Serial number
- xxx = MultiMet number
- II = Julian day number
- YY = Year number
- I = Repetition letter

The List file is a list of Details file, History file, and Calibration files, an example of for temperature sensor VII029 is:

List file name TTVI1029.L

Contents
\metdb\metdb_da\DETAILS.100
\metdb\metdb_da\HISTORY.100
\metdb\metdb_da\TW19590a.100
\metdb\metdb_da\TD19590a.100
\metdb\metdb_da\TW19590b.100
\metdb\metdb_da\TD19590b.100

Note that two calibrations performed on the same day for a sensor will generate Calibration files of the same name except for the Repetition letter.
PIPEDREAM reads and links the contents of a List file together, enabling the various files to be treated almost as one. The History file is updated every time a calibration is performed on the sensor, when the sensor is deployed, and when it returns from a cruise. Access to the METDB database on PIPEDREAM allows the user to examine the files for any sensor by shelling out of GWBASIC into PIPEDREAM. This is a little bit slow, so for prolonged investigation into the database it is best to enter PIPEDREAM from DOS.
1.2 CALIBRATIONS

The Calibration Tank is used to calibrate temperature sensors by immersing the sensor in water of a set temperature, measured by a precision thermometer. The tank calibration software controls the tank and steps it through various temperatures set by the user. During a calibration the PC takes simultaneous readings of both the sensors and the precision thermometer as the tank reaches a stable temperature. On completion of the calibration, both the History and List files are automatically updated. All data points and a polynomial least squares fit, generating up to four coefficients, are written to a new Calibrations file.

Calibration Menu

There are three basic functions accessed via the calibration menu, these are :-

a) Tank calibrations of any temperature sensors

b) Manual input for wind speed and temperature sensor data, generating calibration coefficients and updating the database.

c) Generation of calibration certificates.

Tank Calibrations

To run the tank calibration software select the Calibration Menu from the Main Menu, this then gives the option of "TANK", which loads the control software. All calibrations performed by this method are automatically entered into the database, with both List files and History files being updated. This does require however, that the sensors already exist on the database.

Manual Input

When calibrations are performed outside of I.O.S. they generally need processing to produce calibration coefficients, also the database will require updating. To perform this task there is a manual input option where the raw calibration data is input via a program editor. There are two programs to choose from, one for wind speed sensors and one for
temperature sensors, both of which generate a polynomial least squares fit. Again the database is fully updated, therefore all sensors must exist on the database, as the program will check and detect an error. Both programs are menu driven enabling the user to select the type of sensor to be calibrated.

**Calibration Certificates**

This software compares any two selected calibrations from one sensor, and performs a comparison between them. This is useful in testing the latest calibration with an earlier one, to see whether the sensor has a stable calibration. Calibration Certificates can be produced any time that they are required. To run this program the user can select either wind speed or temperature from the Calibration Menu, as both programs will give the certificate option. The type of sensor requiring a certificate is chosen when the Calibration Certificate program is executed.
1.3 MULTIMET SENSOR MANAGEMENT

Due to the large number of sensors, about 70-80, it is necessary to have some means by which to keep track of where they all are, and in what state. The MultiMet Sensor Management system has its own list of sensors and their MultiMet numbers, these are stored in a random access file called Sensors.dat. Each sensor in the file is tagged with a letter R, D or M, this give's an indication of the condition of sensor.

R = Sensor has current Calibration (READY)
D = Sensor is Deployed (DEPLOYED)
M = Sensor requires new Calibration (MAINTENANCE)

This effectively makes three lists, and when any sensor is moved from one list to another the sensors History file is updated. The entry consists of the date and a suitable comment, also the user can add further comments if required. The software also displays which sensors are in which list, along with the sensors History file, enabling the user to choose which is the best sensor for the task he has in mind.

Allocation Menu

This menu gives access to the sensor management software and allows new sensors to be entered into the database. The objective is to record any change in a sensor status, such as a sensor being returned from a deployment. There are three categories for the sensor management and one for new sensors, these are :-

a) Calibrated Sensors
Calibrated sensors refers to a list of sensors that have a current calibration and are ready for deployment.

b) Deployed Sensors
Deployed sensors is a list of sensors currently deployed.

c) Sensors Requiring Calibration
Sensors requiring calibration is a list of sensors recently returned from deployment and needing post calibrations, or for sensors that still require some form of maintenance.

d) New Sensor Entries
For creating a new database entry and allocating a MultiMet number to the new sensor.

e) Dump to Printer Sensor List
This option allows the user to print out all sensors of each type, including their status and MultiMet numbers.

Selecting Sensors For Deployment

From the Main Menu select the Allocation Menu, move the cursor to CALIBRATED SENSORS and press <RETURN>. Now a choice of sensor types is displayed, select the appropriate sensor type, such as air temperature. The program now displays a list of air temperature sensors that have valid calibrations and are available for deployment. The cursor keys enable the user to scroll through the list and select a particular sensor. When this is done the History file for that sensor is displayed. This enables the user to check the past history of the sensor, and make sure that it is suitable for the required use. The sensor can be de-selected if it is not suitable, or by entering < />, it will be removed from the calibrated sensor list and entered into the deployed sensor list. The History file is updated with the comment "sensor deployed", the user can also add further comments such as where the sensors is to be deployed etc. The program then returns the user back to the Select Sensor Type Menu, so that another sensor can be chosen, or return to the Allocation Menu.

Sensors Returning From Deployment

From the Main Menu select the Allocation Menu, move the cursor to DEPLOYED SENSORS and press <RETURN>. Now a choice of sensor types is displayed, select the appropriate sensor type, such as air temperature. The program now displays a list of air temperature sensors that are currently deployed. The cursor keys enable the user to scroll through the list and select the sensor that has just returned. When this is done the History file for that sensor is displayed. This enables the user to check where the sensor has come from and any other relevant comments. The sensor can be de-selected if it is the wrong one, or by entering < />, it will be removed from the deployed sensor list and entered into the sensors requiring calibration list. The History file is updated with the comment, "sensor returned to maintenance", the user can also add further comments such as what state the sensor is in, i.e. whether it is working or not, etc. The program then returns the user back to the Select Sensor Type Menu, so that another sensor can be chosen or return back to the Allocation Menu.
Sensors Made Available For Deployment

From the Main Menu select the Allocation Menu, move the cursor to SENSORS REQUIRING CALIBRATION and press <RETURN>. Now a choice of sensor types is displayed, select the appropriate sensor type, such as air temperature. The program now displays a list of air temperature sensors that are currently under maintenance. The cursor keys enable the user to scroll through the list and select the sensor that has just been refurbished and has a current calibration. When this is done the History file for that sensor is displayed. This enables the user to check whether the sensor has all reported problems corrected. The sensor can be de-selected if this is not so, or by entering </>, it will be removed from the sensor requiring calibration list and entered into the calibrated sensors list. The History file is updated with the comment "sensor now available", the user can also add further comments such as whether a fan will need changing soon, etc. The program then returns the user back to the Select Sensor Type Menu, so that another sensor can be chosen, or return back to the Allocation Menu.

Dump To Printer Sensor List

From the Main Menu select the Allocation Menu, move the cursor to DUMP TO PRINTER SENSOR LIST press <RETURN>. The program then displays a SENSOR TYPE menu, selecting a particular type will dump to the printer a list of every sensor in the database of that type. Included in the printed list is the MultiMet number of each sensor, and the sensors status.

New Sensors Entries

When there is a need for a new sensor to be added to the database and a MultiMet number allocated to it, select NEW SENSOR ENTRIES. The program creates a blank Details and History file and inserts them into a List file, later the full details must be added to the files. The Sensor Management file is also updated and the sensor filed under the chosen list. To run the software first select the Allocation Menu from the Main Menu, selecting NEW SENSOR ENTRIES produces a menu of sensor types. Selecting the correct SENSOR TYPE means that the new MultiMet number is chosen in the correct range. The display then shows the new number and asks for the sensor type code, examples of which are shown on the left of the screen, these are always two characters. The user is then prompted for the serial number of the sensor, this is a four digit code preceded by two characters representing the manufacturer's name (for further details refer to PKT's description in the database handbooks). Finally the user is asked under which list should the sensor be filed R/M/D,
then prompted to press <any key> when all files have been written to disc. The user is then left in the Select Sensor Type Menu, and can return back to the Allocation Menu.

Under the Select Sensor Type Menu, there is also an option to Change MultiMet Allocation, enabling the user to change the details of a sensor as recorded in the MultiMet Sensor Management file. This option has no effect on the current List / Details / History files allocated to that MultiMet number. Possible uses could be to re allocate a MultiMet number to a new sensor, or changing the allocation code without updating the History file.
PART 2 OPERATING PROCEDURES

2.1 CHOOSE A SENSOR

All sensor details and their status are held in the MultiMet Sensor Management files stored in the TANDON 386 PC used for controlling the calibration tank. Therefore by accessing this information the Sensor Management system will produce a list of available sensors of the type required. The system enables the user to scroll through this list, as well as examining the History file of any sensor, so checking its suitability for that particular deployment. When a suitable sensor is located it is selected and so removed from the "sensor available" list and put into the "deployed sensors" list. Comments can be added to the automated update History file, indicating where the sensor is to be deployed. The sensor can now be removed from the racks behind the lift and put into a transportation case, ready for deployment.

The procedure for selecting a sensor for deployment from the Sensor Management system is as follows.

A) Turn on the mains supply at the wall behind the PC and turn on the PC.

B) The PC should be displaying the Main Menu of the Sensor Management system, select the Allocation Menu and move the cursor to "Calibrated Sensors" and press <RETURN>.

C) A choice of sensor types is now displayed, select the appropriate sensor type, such as Air Temperature. The system now displays a list of air temperature sensors that are all ready for deployment.

D) Use the cursor keys to scroll up and down through the list, by pressing <RETURN> a sensor's History file is displayed on the screen. Pressing <1> will allow another sensor to be examined, or by pressing </> the sensor is finally selected for deployment.

E) The History file is updated with the comment "sensor now deployed", additional comments can be added, such as where the sensor is to be deployed.

F) From the Sensor Type Menu return back to the Allocation Menu, and from there to the Main Menu.
H) Go to sensor storage area and remove selected sensor from shelf, and store in transportation box ready for deployment.

2.2 RETURNING A SENSOR

Sensors should be returned to the area behind the lift, removed from their travel boxes and stored in their correct locations on the racks. If possible a wash in fresh water to remove any salt would be advantageous. The Sensor Management filing system should be updated by removing the Returned sensors from the Deployed list. Comments can be entered on the current state of the sensor and what repairs if any need doing. The sensor requires a post deployment calibration as soon as practically possible. When calibrating sensors it is possible to have a mix of sensor types in the tank at the same time, assuming the same temperature ranges are required by all sensors.

Psychrometers

Psychrometers are calibrated in our own calibration tank, therefore little delay should be necessary before a post calibration is completed. The following steps should be taken to complete a post calibration.

a) Dismantle the psychrometer by removing the lower half of the case and detaching the water bottle holder from the lid. This requires two screws to be undone from the lid, and the wick to be removed from the wet bulb element. If the wick is in a bad condition it should be replaced, or boiled to clean and remove all salt.

b) Turn on the mains supply at the wall behind the PC. Check that the Tank, Cooler, F25, Interface power supply, and the Tandon PC all have their supplies turned on. Also check that the tank control box is switched to PC.

c) Visually inspect the fan for any signs of corrosion and attach the fan power test lead to the psychrometer. With a 12 volt supply the current consumption should be about 0.2 A. If the fan fails to rotate or is particularly noisy, then it should be replaced after the calibration is completed.

d) Place the psychrometer in the tank, as near to the F25 probe as possible, using the support bars. The maximum number of psychrometers to be calibrated at one time is six.
e) The PC should be displaying the Main Menu of the Sensor Management software. Select the Calibration menu, followed by "TANK". This will run the tank control software.

f) Enter your name, then on the displayed menu, check that the arrow is pointing to psychrometers. Enter the number of sensors to be calibrated. Press the ESC key to move on.

g) Enter in the serial number of the sensor to be calibrated such as VI1029. This will display the details about that sensor, such as which electronics tube it belongs with. Now connect the psychrometer to the correct set of electronics. Press ESC when all sensors have been entered in this way.

h) The program now tells the user which sensor should be connected to which channel. Each cable coming from the interface box is labelled with its channel number, and this can be plugged into the electronics tube. As each sensor is connected, the channel number and frequency output from the sensor are displayed on the screen. Check that this is correct, i.e. in the range of 1500 - 2000 Hz for a psychrometer, for each channel. This updates about every 10 sec, therefore it will take a little while before the readings are stable.

i) Press <RETURN> to continue, this will move on to the temperature range and steps. <RETURN> will enter the default values of 0 - 35 DEG step 5, a calculated duration time for the calibration will now be displayed. If this is acceptable then continue by pressing <RETURN>.

The tank software will now perform the calibration without any further user input, storing the data in a temporary file. On completion, the relevant files for METDB database are created or updated, such as List file, History file and Calibration files.

**SST**

Again SST sensors are calibrated in our own tank and therefore can be calibrated without any delay.

a) Thoroughly wash both the SST rope and the electronics tube to remove all salt.

b) Remove all weights from the rope, check for corrosion on screws etc.
c) Turn on the mains supply at the wall behind the PC. Check that the Tank, Cooler, F25, Interface power supply, and the Tandon PC all have their supplies turned on. Also check that the tank control box is switched to PC.

d) Place the rope or up to four sets of sensors, in to the tank, as close to the F25 probe as possible.

e) The PC should be displaying the main menu of the Sensor Management software. Select the Calibration menu, followed by TANK. This will run the tank control software.

f) Enter your name, then on the displayed menu, check that the arrow is pointing to SST'S. Enter the number of sensors to be calibrated. Press the ESC key to move on.

g) Enter in the serial number of the sensor to be calibrated such as O10005. This will display the details about that sensor, such as which electronics tube it belongs with. Now connect the SST sensor to the correct set of electronics. Press ESC when all sensors have been entered in this way.

h) The program now tells the user which sensor should be connected to which channel. Each cable coming from the interface box is labelled with its channel number, and this can be plugged into the electronics tube. As each sensor is connected, the channel number and frequency output from the sensor are displayed on the screen. Check that this is correct ie in the range of 1500 - 2000 Hz for an SST sensor, for each channel. This updates about every 10 sec, therefore it will take a little while before the readings are stable.

i) Press <RETURN> to continue, this will move on to the temperature range and steps. <RETURN> will enter the default values of 0 - 35 DEG step 5, a calculated duration time for the calibration will now be displayed. If this is acceptable then continue by pressing <RETURN>.

The tank software will now perform the calibration without any further user input, storing the data in a temporary file. On completion, the relevant files for METDB database are created or updated, such as List file, History file and Calibration files.

Wind speed (manual entries)

All wind speed sensors are calibrated at the Met Office at Bracknell, therefore there may be some delay before calibrations are done. This makes it all the more essential that the sensors
are washed and cleaned before being stored in their correct slots on the racks. On return from the Met Office the data needs to be assimilated into the database and proper calibrations calculated. This is achieved by the following steps.

a) Turn on the mains supply at the wall behind the PC, then switch on the Tandon PC.

b) The PC should be displaying the main menu of the Sensor Management software. Select the Calibration menu, followed by WIND SPEED.

c) The wind speed calibration program displays major types of wind sensors such as Cup Anemometer, select the relevant type.

d) Input date of calibration ie date measurements were taken at the Met Office. If unknown use the default date.

e) Enter your name.

f) Enter the drive letter that database is held on, i.e. D:. This is also the default setting which is correct for the Tank PC.

g) Enter the calibration standard, this is usually the TESTOVENT when the measurements were taken at the Met Office.

h) Enter any comments.

i) Follow instructions for entering sensor details such as serial number, etc.

j) Details from database are now displayed, check that correct sensor has been chosen.

k) Calibration data is now input via an editor provided by the program.

l) On completion of data entry, the coefficients are calculated and the database is automatically updated.

m) Program returns the user back to the Main Menu.
Temperature Sensors (manual entry)

Generally temperature sensors are calibrated using the TANK system (see Psychrometers for details). When this is not the case and a set of calibration measurements need to be manually entered and the coefficients calculated, follow the steps below.

a) Turn on the mains supply at the wall behind the PC, then switch on the Tandon PC.

b) The PC should be displaying the main menu of the Sensor Management software. Select the Calibration menu, followed by TEMPERATURE.

c) The temperature calibration program displays major types of temperature sensors such as Psychrometer, select the relevant type.

d) Input date of calibration, i.e. date measurements were taken. If unknown use the default date.

e) Enter your name.

f) Enter the drive letter that database is held on ie D:. This is also the default setting which is correct for the Tank PC.

g) Enter the calibration standard, this is usually the TESTOVENT when the measurements were taken at the Met Office.

h) Enter any comments.

i) Follow instructions for entering sensor details such as serial number, etc.

j) Details from database are now displayed, check that correct sensor has been chosen.

k) Calibration data is now input via an editor provided by the program.

l) On completion of data entry, the coefficients are calculated and the database is automatically updated.

m) Program returns the user back to the Main Menu.
2.3 VISITING THE MET OFFICE

When calibrating Wind Speed sensors it is usual to use a wind tunnel, but I.O.S. does not have this facility. The nearest usable wind tunnel is at the Met Office at Bracknell, and is therefore used for any calibrations of this kind. The normal procedure is to take measurements of the frequency output of the sensors at set wind speed values from 5 - 35 m/s. This is measured on a counter or DVM depending on whether the output is frequency or analog, and the wind speed is measured on the Met Office TESTOVENT. These figures are then entered into the database, and coefficients calculated, by using the manual entry for wind speeds in the MultiMet Sensor Management system. (see 2.2 Returning a Sensor for more details)

The normal procedure is as follows:-

Prior to visit

A) Phone the Met Office to arrange a convenient time to visit.
   Phone No. 0344 420242
   Contact PAUL JOY

B) Check with transport that an I.O.S. car is available.

C) Select the sensors for calibration by using the MultiMet Management system.

D) Remove selected sensors from storage racks and pack into suitable travelling boxes.

Items to take

A) Each sensor type will require a stand to mount the sensor in the tunnel.

<table>
<thead>
<tr>
<th>Sensor Type</th>
<th>Stand Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vector</td>
<td>Met Office stand available</td>
</tr>
<tr>
<td>Young</td>
<td>Metal right angle stand for generator only</td>
</tr>
<tr>
<td>Young Monitor AQ</td>
<td>Spigot stand to mount whole sensor.</td>
</tr>
</tbody>
</table>

B) Counter and DVM to measure output from sensors.

C) Notebook for writing results in.
On Return

A) Put sensors back in their correct position on the storage racks.

B) Enter calibrations into database. Refer to "2.2 Returning a Sensor" for more explicit details on this procedure.

C) Enter into a sensor's History file any significant details, discovered during the calibration, such as sticky bearing etc.
**APPENDIX 1**

**MULTIMET SENSOR DATABASE - Specifications**

The database will be maintained using PIPEDREAM.

**ALL FILES WILL BE TEXT FILES SAVED FROM PIPEDREAM WITH THE "TAB" OPTION SELECTED.** Such files can be imported directly into dBase III or Microsoft Word.

Each sensor will be known by an eight character "Sensor ID" and will also have a three character MultiMet Number allocated sequentially as the sensors are purchased.

*Sensors ID's will be of form SSMmmn

where

SS = sensor type
MM = manufacturer
mmn = sensor number (derived from the manufacturers serial no.)

**Sensor Type (SS)**
The suggested sensor types are

(wind velocity)

WS Wind Speed (eg Cup anemometer)
WW Wind velocity (eg Propeller Vane instrument)
WD Wind direction (eg Vane)

(ships velocity)

VS Vessel's speed
VH Vessel's head

(air temperature and humidity)

TD Dry Bulb Temperature
TW Wet Bulb Temperature
TT Psychrometer (wet + dry bulb)
HU Relative humidity device (eg Humicap)

(sea surface temperature)

SF Sea surface temperature (Fish)
SP Sea surface temperature (Soap)
SR Sea surface temperature (Radiometer)
ST Sea surface temperature (eg Hull contact)

(others)
RS Short Wave Radiation (eg Kipp solarimeter)
RL Long Wave Radiation (eg Eppley Pyranometer)
AP Air Pressure
PD Precipitation detector
PG Precipitation gauge
AC Accelerometer

Manufacturers Code (MM)
The suggested manufacturers codes are

BG Brookes & Gatehouse
DC Digicourse
DE Delta-T Devices
DI Didcot Instruments
EP Eppley
IO IOS designed/built
KZ Kipp & Zonen
SU Sundstrand
VA Vaisala
VI Vector Instruments
YG R M Young

Sensor Numbers (nnnn)
The four digit numbers should be equal to the manufacturers serial number (eg for Vector Instruments) or to the last four digits of that serial number. Where no serial number is given (eg R M Young) the sensor number may reflect the order of purchase, or the MultiMet number may be used.

File Structure
In describing the file structure the example used will be a Vector Instruments Psychrometer Serial number 1029 having MultiMet number 123.
The structure described will hold for the Z88, IBM PC and Master 128, it can be converted for use under ADFS (eg on Master 128 or Archimedes) by using the MOVE.EXE on the Master 512.

Files will all be stored at a single directory level, extensions will be used to distinguish between files.

<table>
<thead>
<tr>
<th>File</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>*.HLP</td>
<td>&quot;readme&quot; text files describing the database structure.</td>
</tr>
<tr>
<td>*.L</td>
<td>pipedream list files</td>
</tr>
<tr>
<td>SSMMnnnn.L</td>
<td>pipedream list file for a given sensor</td>
</tr>
<tr>
<td>Details.L</td>
<td>(for example) list file allowing all sensors details to be printed</td>
</tr>
<tr>
<td>CUM33.L</td>
<td>(for example) list file allowing calibrations for a particular cruise to be printed</td>
</tr>
</tbody>
</table>

File

| Sensors.txt | Master file containing list of MultiMet Numbers, sensor ID's and sensor serial numbers. |

<table>
<thead>
<tr>
<th>MultiMet Number</th>
<th>Sensor ID</th>
<th>Serial Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>123</td>
<td>TTV11029</td>
<td>1029</td>
</tr>
</tbody>
</table>

Details.mmm (where mmm is the MultiMet number, eg details.123) contains the information which generally is constant for each type of sensor. For example, sensors type, cost, spare parts, etc.

History.mmm Information on calibration and usage of the sensors.
SSddmmyy.mmm (where SS is the sensor type and ddmmyy is the calibration date). These files contain information for a given calibration, thus for the psychrometer there will be files

TD011187.123, TW011187.123,
TD100689.123, TW100689.123 say.

EXAMPLE LIST FILE

FOR THE ABOVE EXAMPLE THE LINK FILE "TDVI1019.L" would contain ...

Details.123
History.123
TD011187.123
TW011187.123
TD100689.123
TW100689.123
APPENDIX 2

CALIBRATION POLICY FOR WIND SPEED AND TEMPERATURE SENSORS

SENSOR DEPLOYMENT

1. For Individual Cruise

Sensors should be calibrated within one month of cruise commencement or completion. Where the transportation overheads exceed this period, calibration should be as close as possible to this time scale to minimise the time between post and pre calibrations.

2. Multiple Cruise Campaign

Sensors should be calibrated within month prior to deployment, where the transportation allows.

Sensors should not be deployed for periods exceeding 4 months.

When sensors are removed there should be overlap in deployment time between sensors of identical types, to allow calibration checks to be made in-situ. Therefore not all of identical sensor types should be removed at one routine service during a campaign, but replacement should be rotated between sensor suites/deployment sites.

Returned sensors should be post calibrated within one month of removal from data collection, where transport allows.

3. Sensors removed because of calibration problems

These should be calibrated within one month of removal from data collection, transportation allowing.
CALIBRATION DETAILS

1. All sensors before deployment must be entered into the database with all associated files:
   - List file
   - History file
   - Calibration file

2. Pre-deployment calibrations should be supplied to scientist concerned after calibration as a print out of the calibration file.

3. Post-deployment calibrations should be supplied to scientist concerned after re-calibration as a CALIBRATION CERTIFICATE between the pre and post calibrations.

Any calibration differences outside sensor specification should be detailed and explained.
APPENDIX 3.2

MULTIMET SENSOR MANAGEMENT SYSTEM

MAINMENU.BAS

MAIN MENU
- Calibration Menu
- Allocation Menu
- Metdb
- Quit

SEND.BAS

ALLOCATION MENU
- Calibrated Sensors
- Deployed Sensors
- Sensors Requiring Calibration
- New Sensor Entries
- Dump Sensors to Printer
- Return to Main Menu

SEND.BAS

SELECT SENSOR TYPE
- Air Temperature
- Sea Temperature
- Wind Speed
- Wind Direction
- Radiometers
- Others
- Return to Allocation Menu

SEN3.BAS

SELECT SENSOR TYPE
- Air Temperature
- Sea Temperature
- Wind Speed
- Wind Direction
- Radiometers
- Others
- Change MultiMet Allocations
- Return to Allocation Menu

SENPR1.BAS

SELECT SENSOR TYPE
- Air Temperature
- Sea Temperature
- Wind Speed
- Wind Direction
- Radiometers
- Others
- Return to Allocation Menu
APPENDIX 3.3

METDB DATABASE

MAINMENU.BAS

MAIN MENU
Calibration Menu
Allocation Menu
Metdb
Quit

PIPEDREAM.EXE

PIPEDREAM MENU
F1 = Wind Speed
F2 = Wind Direction
F3 = Psychrometers
F4 = SST Sensors
F5 = Radiometers
F6 = General
F7 = Next Page of list file
F8 = Previous Page of list file
1000 REM **********************************************
1010 REM *******************  MAINMENU.BAS by RWP 8/11/90 ********************
1020 REM **********************************************************************
1040 REM
1041 REM  =====  Set error trap.....
1042 REM
1043 ON ERROR GOTO 40000
1044 REM
1045 REM  =====  Call  subroutine  to set up Ctrl Esc key trap.....
1046 REM
1047 GOSUB 14000
1048 KEY OFF
1049 SCREEN 9
1056 REM
1057 REM  =====  Set up user menu....
1058 REM
1059 COLOR 7,1: CLS
1060 REM
1061 REM  =====  Draw up menu box.....
1065 REM
1067 LINE (5,5)-(635,345),B
1090 LINE (180,83)-(480,266),2,BF
1100 LINE (185,88)-(475,261),64,B
1110 LOCATE 8,26:PRINT" MAIN MENU "
1120 LOCATE 10,26:PRINT"
1130 LOCATE 12,26:PRINT" Calibration Menu " ;CHR$(27);" "
1140 LOCATE 14,26:PRINT" Allocation Menu "
1142 LOCATE 16,26:PRINT" Metdb "
1145 LOCATE 18,26:PRINT" Quit "
1150 LOCATE 21,8:PRINT" Move arrow to required sensor type and press <RETURN> when finished"
1160 LOCATE 22,8:PRINT"Move around using cursor keys"
1161 COLOR 12,1:LOCATE 23,6:PRINT" To Return to previous Menu at ant point Press CTRL ESC and then RETURN"
1162 COLOR 3:LOCATE 4,29:PRINT "MULTIMET SENSOR MANAGEMENT"
1165 REM
1170 REM  =====  Define the cursor keys.....
1175 REM
1180 DN$=CHR$(0)+CHR$(80):DP$=CHR$(0)+CHR$(72)
1185 X=12:Y=59:KY$=""
1190 REM
1191 REM  =====  Set up pointer box in menu.....
1192 REM
1193 COLOR 7,1
1195 Y1 = 153
1198 GOSUB 20000
1200 LOCATE X,Y
1202 REM
1205 REM  =====  Reading input from keyboard....
1207 REM
1210 WHILE KY$<>CHR$(13)
1220  KY$=INKEY$
Subroutine to set up key trap for Ctrl Esc

Define keys to include all possible combinations of CAPS LOCK and NUM LOCK.

14016 REM ===== Define keys to include all possible combinations of CAPS LOCK and NUM LOCK.....

14020 KEY 15,CHR$(&H64)+CHR$(1)
14030 KEY 16,CHR$(&H44)+CHR$(1)
14040 KEY 17,CHR$(&H24)+CHR$(1)
14050 KEY 18,CHR$(&H4)+CHR$(1)

14060 REM
14064 REM ===== Direct all escape key combinations to line 65300 .....}
14066 REM
14070 ON KEY (15) GOSUB 65300
14080 ON KEY (16) GOSUB 65300
14090 ON KEY (17) GOSUB 65300
14100 ON KEY (18) GOSUB 65300

14115 REM ===== Turn all escape key combinations on.....
14117 REM
14120 KEY (15) ON
14130 KEY (16) ON
14140 KEY (17) ON
14150 KEY (18) ON

14200 RETURN

20000 REM
20010 REM ===== First overwrite old box.....
20020 REM
20030 REM
20040 REM
20050 REM ===== First overwrite old box.....
20060 REM
20070 LINE (199,Y1)-(464,Y1+15),2,B
20100 REM ===== Now draw box and flash at new position.....
20110 REM
20120 IF (KY$=DN$) THEN Y1 = Y1 + 28 ELSE IF (KY$=UP$) THEN Y1 = Y1 - 28
20140 FOR COL = 9 TO 15
20150 LINE (199,Y1)-(464,Y1+15),COL,B
20160 NEXT COL
20170 LINE (199,Y1)-(464,Y1+15),12,B
20500 RETURN
40000 REM
40002 REM --------------------------------------------------------
40005 REM ----- Error checking subroutines ----- 
40007 REM --------------------------------------------------------
40010 REM
40020 REM
40022 REM ===== Check for errors during file opening.....
40024 REM
40030 IF ERR = 53 THEN GOSUB 40100: RESUME 1040
40032 IF ERR = 52 THEN GOSUB 40200: RESUME 1480
40034 IF ERR = 71 THEN GOSUB 40300: RESUME 1520
40036 IF ERR = 72 THEN GOSUB 40400
40038 IF (ERR = 5 OR ERR = 11) THEN GOSUB 40500: RESUME 65300
40040 IF ERR = 76 THEN GOSUB 40600: RESUME 65300
40050 STOP
40100 REM
40105 REM ===== File not found.....
40110 REM
40115 LINE (50,196)-(330,250),3,BF
40120 COLOR 12,1: LOCATE 15,11: PRINT "Sensor not found in database"
40130 COLOR 15,1: LOCATE 16,17: PRINT "Please try again"
40140 BEEP
40190 RETURN
40200 REM
40205 REM ===== Invalid filename.....
40210 REM
40215 LINE (50,196)-(330,250),3,BF
40220 COLOR 12,1: LOCATE 15,14: PRINT "Invalid filename used"
40230 COLOR 15,1: LOCATE 16,17: PRINT "Please try again"
40240 BEEP
40290 RETURN
40300 REM
40305 REM ===== Drive not ready.....
40310 REM
40315 LINE (50,196)-(330,250),3,BF
40320 COLOR 12,1: LOCATE 15,15: PRINT " Drive ";'DRIVES;" \ not ready"
40330 COLOR 15,1: LOCATE 16,13: PRINT "Press return when ready"
40340 BEEP
40350 ANSWERS = ""
40360 WHILE (ANSWERS <> CHR$(13))
40370 ANSWERS = INKEYS
40380 WEND
40390 RETURN
40400 REM
40405 REM ===== Disk media error.....
40410 REM
40420 LINE (50,196)-(330,250),3,BF
40430 COLOR 12,1: LOCATE 15,11: PRINT "Error with hardware or disk"
40440 COLOR 15,1: LOCATE 16,17: PRINT "Program halted"
40490 RETURN
40500 REM
40502 REM ===== Illegal function call.....
40505 REM
40510 COLOR 12,8: LOCATE 17,33: PRINT "Insignificant data"
40520 LOCATE 18,26: PRINT "Could not calculate coefficients"
40530 COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
40540 WHILE (INKEYS = "")
40550 WEND
40555 CLS
40560 REM
40562 REM ===== Exit program and return to operating system.....
40565 REM
40570 RETURN
40580 REM
40590 REM ===== Path not found.....
40600 REM
40610 COLOR 12,1:L0CATE 17,24: PRINT "Could not find database on drive ";DRIVES
40620 COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
40630 WHILE (INKEYS = "")
40640 WEND
40650 WEND
40660 REM
40670 REM ===== Exit program and return to operating system.....
40680 REM
50000 REM
50100 REM DRAW GREEN BOX
50105 REM CLS
50110 COLOR 15,1
50120 FOR N=1 TO 150 STEP 2
50130 LINE (BB%-N,160-N)-(BB%+N,160+N),2,BF
50140 NEXT N
50150 RETURN
50160 REM
65300 REM ===== Subroutine for execution when Ctrl Esc has been pressed ----
65305 REM ---------------------------------------------------
65310 REM
65320 REM
65330 SCREEN 0,0,0
65340 LOCATE 14,9: PRINT "Program Execution Halted..... Returning to operating system"
65350 LOCATE 16,29: PRINT "---Press Any Key---"
65360 WHILE (INKEYS = "")
65370 WEND
65380 SYSTEM
1000 REM ******************************************************
1010 REM ****************************************************
1020 REM ******************************************************
1040 REM
1041 REM ===== Set error trap....
1042 REM
1043 ON ERROR GOTO 40000
1044 REM
1045 REM ===== Call subroutine to set up Ctrl Esc key trap....
1046 REM
1047 GOSUB 14000
1048 KEY OFF
1049 SCREEN 9
1056 REM
1057 REM ===== Set up user menu....
1058 REM
1059 COLOR 7,1:CLS
1060 REM
1061 REM ===== Draw up menu box....
1062 REM
1063 REM
1064 LINE (5,5)-(635,345),,,B
1065 LINE (180,83)-(480,266),2,BF
1066 LINE (185,88)-(475,261),64,B
1067 LOCATE 8,26:PRINT"CALIBRATION MENU"
1068 LOCATE 10,26:PRINT"
1069 LOCATE 12,26:PRINT"Tank;CHR$(27);"
1070 LOCATE 14,26:PRINT"Wind Speed"
1071 LOCATE 16,26:PRINT"Temperature"
1072 LOCATE 18,26:PRINT"Return to Main Menu"
1073 LOCATE 21,8:PRINT"Move arrow to required sensor type and press <RETDRN> when finished"
1074 LOCATE 22,26:PRINT"Move around using cursor keys"
1075 COLOR 3:LOCATE 4,29:PRINT"MULTIMET SENSOR MANAGEMENT"
1076 REM
1100 REM ===== Define the cursor keys....
1101 REM
1102 DN$=CHR$(0)+CHR$(80):UP$=CHR$(0)+CHR$(72)
1103 X=12:Y=59:KY$=""
1104 REM
1105 REM ===== Set up pointer box in menu....
1106 REM
1107 REM
1108 REM ="MULTIMET SENSOR MANAGEMENT"
1109 REM
1120 REM ===== Reading input from keyboard....
1121 REM
1122 WHILE KY$<>CHR$(13)
1123 KY$=INKEY$
1124 IF (KY$=DN$) AND ((X)>12)AND(X<=16) THEN LOCATE X,Y-3:PRINT";X=X+2:LOCATE X,Y-2:PRINT CHR$(27); : GOSUB 20000
1125 IF (KY$=UP$) AND ((X)>14)AND(X<=18) THEN LOCATE X,Y-3:PRINT";X=X-2:LOCATE X,Y-2:PRINT CHR$(27); : GOSUB 20000
1126 WEND
1292 IF X=16 THEN RUN "TEMPCAL"
1293 IF X=18 THEN RUN "MAINMENU"
1295 IF X=12 THEN RUN "Tank"
1296 IF X=14 THEN RUN "WINDCAL"
1297 GOTO 1059
1300 END
14000 REM
14002 REM
14005 REM ---- Subroutine to set up key trap for Ctrl Esc ----
14007 REM
14010 REM
14014 REM
14016 REM ----- Define keys to include all possible combinations of CAPS LOCK
14017 REM and NUM LOCK....
14018 REM
14020 KEY 15,CHR$(&H64)+CHR$(1)
14030 KEY 16,CHR$(&H44)+CHR$(1)
14040 KEY 17,CHR$(&H24)+CHR$(1)
14050 KEY 18,CHR$(&H4)+CHR$(1)
14060 REM
14064 REM ----- Direct all escape key combinations to line 65300 ..... 
14066 REM
14070 ON KEY (15) GOSUB 65300
14080 ON KEY (16) GOSUB 65300
14090 ON KEY (17) GOSUB 65300
14100 ON KEY (18) GOSUB 65300
14110 REM
14115 REM ----- Turn all escape key combinations on....
14117 REM
14120 KEY (15) ON
14130 KEY (16) ON
14140 KEY (17) ON
14150 KEY (18) ON
14200 RETURN
20000 REM
20010 REM
20020 REM ---- Subroutine for drawing & flashing pointer box in menu ----
20030 REM
20040 REM
20050 REM ---- First overwrite old box.....
20060 REM
20070 LINE (199,Y1)-(464,Y1+15),2,B
20100 REM ----- Now draw box and flash at new position.....
20110 REM
20120 IF (KY$ = DNS) THEN Y1 = Y1 + 28 ELSE IF (KY$ = UPS) THEN Y1 = Y1 - 28
20140 FOR COL = 9 TO 15
20150 LINE (199,Y1)-(464,Y1+15),COL,B
20160 NEXT COL
20170 LINE (199,Y1)-(464,Y1+15),12,B
20500 RETURN
40000 REM
40002 REM
40005 REM ---- Error checking subroutines ----
40007 REM

---End of Document---
40010 REM
40020 REM
40022 REM ======= Check for errors during file opening.....
40024 REM
40030 IF ERR = 53 THEN GOSUB 40100: RESUME 1040
40032 IF ERR = 52 THEN GOSUB 40200: RESUME 1480
40034 IF ERR = 71 THEN GOSUB 40300: RESUME 1520
40036 IF ERR = 72 THEN GOSUB 40400
40038 IF (ERR - 5 OR ERR = 11) THEN GOSUB 40500: RESUME 65300
40040 IF ERR = 76 THEN GOSUB 40600: RESUME 65300
40050 STOP
40100 REM
40105 REM ===== File not found.....
40110 REM
40115 LINE (50,196)-(330,250),3,BF
40120 COLOR 12,1: LOCATE 15,11: PRINT " Sensor not found in database "
40130 COLOR 15,1: LOCATE 16,17: PRINT " Please try again "
40140 BEEP
40190 RETURN
40200 REM
40205 REM ===== Invalid filename.....
40210 REM
40215 LINE (50,196)-(330,250),3,BF
40220 COLOR 12,1: LOCATE 15,14: PRINT " Invalid filename used "
40230 COLOR 15,1: LOCATE 16,17: PRINT " Please try again "
40240 BEEP
40290 RETURN
40300 REM
40305 REM ===== Drive not ready.....
40310 REM
40315 LINE (50,196)-(330,250),3,BF
40320 COLOR 12,1: LOCATE 15,15: PRINT " Drive ";DRIVE$;"\ not ready "
40330 COLOR 15,1: LOCATE 16,13: PRINT " Press return when ready "
40340 BEEP
40350 ANSWERS = ""
40360 WHILE (ANSWERS <> CHR$(13))
40370 ANSWERS = INKEYS
40380 WEND
40390 RETURN
40400 REM
40405 REM ===== Disk media error.....
40410 REM
40415 LINE (50,196)-(330,250),3,BF
40420 COLOR 12,1: LOCATE 15,11: PRINT " Error with hardware or disk "
40430 COLOR 15,1: LOCATE 16,17: PRINT " Program halted "
40440 RETURN
40450 REM
40452 REM ===== Illegal function call.....
40455 REM
40457 COLOR 12,8: LOCATE 17,33: PRINT "Insignificant data"
40460 LOCATE 18,26: PRINT "Could not calculate coefficients"
40470 COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
40480 WHILE (INKEY$ = "")
40490 WEND
CLS
REM ===== Exit program and return to operating system....
RETURN
REM ===== Path not found....
CLS
COLOR 12,1:LOCATE 17,24: PRINT "Could not find database on drive ":DRIVES
COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
WHILE (INKEY$ = "")
WEND
CLS
REM ===== Exit program and return to operating system....
RETURN
REM DRAW GREEN BOX
CLS
FOR N=1 TO 150 STEP 2
LINE (BB%-N,160-N)-(BB%+N,160+N),2,BF
NEXT N
RETURN
REM Subroutine for execution when Ctrl Esc has been pressed -----
CLS
SCREEN 0,0
LOCATE 14,9: PRINT "Program Execution Halted.... Returning to MAIN MENU"
LOCATE 16,29: PRINT "---Press Any Key---"
WHILE (INKEY$ = "")
WEND
RUN "MAINMENU"
10 REM ****************************************
20 REM ***  PROGRAM TANK.BAS  - May 1990 *******
25 REM *** By A. Williams *******************
30 REM ****************************************
35 REM ** Facility for detecting errors **
36 REM ON ERROR GOTO 7080
60 REM ** Dimension all the arrays for the coefficient calculations **
70 DIM X(40),Y(40),A$(15),B$(15),S$(15),G$(15),U$(15),Q$(40),P$(40),C$(40),STDERR(8),II(40)
80 REM ** Need to declare vars/arrays to allow for 160 channels & I/O's **
90 SN%=8:SM%=159:DIM B(159),A%(159)
100 REM ** First step is to get all the sensor details **
110 GOSUB 5230
120 REM ***  Input a temp cycle and display a profile ***
130 REM **  First set up screen to get the details **
140 CLS
150 LINE (15,15)-(625,335),3,BF
160 LINE (20,20)-(620,330),64,B
170 LOCATE 5,27:PRINT " Temperature cycle program "
180 LOCATE 6,23:PRINT " Please enter the following details "
190 LINE (100,110)-(540,290),8,BF
200 LINE (105,115)-(535,285),7,B
205 REM ** SIAT - start temperature **
206 REM ** STOT = stop temperature **
207 REM ** STPT = temperature step **
210 LOCATE 11,22:PRINT " Enter a start temperature "
220 LOCATE 12,22:PRINT " (Default <ENTER> = 0 degs) "
230 LOCATE 11,51:INPUT STAT
240 IF STAT-VAL(CHR$(13)) THEN STAT=0:LOCATE 11,52:PRINT STAT;
250 LOCATE 14,22:PRINT " Enter a stop temperature "
260 LOCATE 15,22:PRINT " (Default <ENTER> = 35 degs) "
270 LOCATE 14,51:INPUT STOT
280 IF STOT=VAL(CHR$(13)) THEN STOT=35:LOCATE 14,52:PRINT STOT;
290 LOCATE 17,22:PRINT " Enter the temperature step "
300 LOCATE 18,22:PRINT " (Default <ENTER> = 5 degs) "
310 LOCATE 17,51:INPUT STPT
320 IF STPT=VAL(CHR$(13)) THEN STPT=5:LOCATE 17,52:PRINT STPT;
330 REM ** now calculate and plot a temp profile **
340 REM * work out estimated time/date of completion *
350 REM * first get initial temp of tank *
355 REM ** Open RS232 communications port **
360 OPEN "COM:300,N,8,1" AS #3
370 REM ** Reset F25 to its original state **
375 REM * Set the units to degrees celcius *
380 WRITE#3,"C"+CHR$(10);FOR T=1 TO 20000:NEXT T
385 REM ** Disable controls on front panel **
390 WRITE#3,"L1"+CHR$(10);FOR T=1 TO 20000:NEXT T
395 REM ** Select the A probe **
400 WRITE#3,"P0"+CHR$(10);FOR T=1 TO 20000:NEXT T
405 REM ** Set the units to degrees celcius **
410 WRITE#3,"UO"+CHR$(10);FOR T=1 TO 20000:NEXT T
415 REM ** Set the resolution to 3 decimal places **
420 WRITE#3,"R1"+CHR$(10);FOR T=1 TO 20000:NEXT T
425 REM ** Get a temperature reading **
430 FOR T=0 TO 25000:NEXT T;WRITE#3,"T"+CHR$(10)
435 IF LOG(3)>-9 THEN INPUT #3,TEMP$ ELSE GO TO 430
440 REM ** set up second screen to display profile **
450 CLS
455 REM ** CD$ = current date, CT$ = current time **
460 CD$=DATE$;CT$=TIME$
470 MONTH=VAL(LEFT$(CD$,2)):DAY=VAL(MID$(CD$,4,2)):YEAR=VAL(RIGHT$(CD$,4))
480 HOUR=VAL(LEFT$(CT$,2)):MINUTE=VAL(MID$(CT$,4,2)):SECOND=VAL(RIGHT$(CT$,2))
485 REM ** CT = current temperature, ASLSTART = 1st temperature reading taken **
490 CT=VAL(MID$(TEMP$,2,7));ASLSTART=CT
500 LINE(5,5)-(635,345),7,B
510 LINE(40,20)-(600,200),8,BF
520 LINE(45,25)-(595,195),7,B
530 REM * draw the axes *
550 PSET(100,160):LINE-(570,160)
560 PSET(100,40):LINE-(100,170)
570 REM ** Label the axes **
575 TIKY=4;RCP=12
580 LABEL=STAT:INCR=:(STOT-STAT)/TIKY
590 FOR I=0 TO TIKY
600 PSET((I*12)/(STOT-STAT));LINE STEP(0,0)-STEP(5,0)
610 LOCATE RCP-(I*2),8:PRINT CINT(LABEL):LABEL LABEL+INCR
620 NEXT I
630 REM * work out time to reach start temp *
635 REM ** LENGTHX = length of the x-axis in minutes **
640 LENGTHX=0 : ARRDIM=((STOT-STAT)/STPT) : INDEX=0 : LASTLENX=0
645 REM ** REACHED() = time in minutes taken to reach each demand temp **
646 REM ** STABALISED() = time in minutes taken to reach demand and stabilise **
650 DIM REACHED(ARRDIM) : DIM STABALISED(ARRDIM)
660 PSET(100,160-((112/(STOT-STAT))*(CT-STAT)))
670 FOR TEMP = STAT TO STOT STEP STPT
680 IF CT > TEMP THEN GOSUB 1650 ELSE GOSUB 1790
690 LASTLENX=LENGTHX
700 INDEX=INDEX+1
710 CT = TEMP
720 NEXT TEMP
730 REM ** Plot the profile **
740 COLOR 14,1
750 FOR L=0 TO ((STOT-STAT)/STPT)
760 LINE-((REACHED(L)*465/LENGTHX)+100,160-((112/(STOT-STAT))*(L*STPT))),,,&HCCCC
770 LINE-((STABALISED(L)*465/LENGTHX)+100,160-((112/(STOT-STAT))*(L*STPT))),,,&HCCCC
780 NEXT L
790 COLOR 7,1
800 REM * These commands will plot the 'x' mark on the x-axis which shows the *
801 REM * time the calibration sequence will begin *
810 PSET((STABALISED(0)*465/LENGTHX)+97,157):LINE-((STABALISED(0)*465/LENGTHX)+103,163)
820 PSET((STABALISED(0)*465/LENGTHX)+97,163):LINE-((STABALISED(0)*465/LENGTHX)+103,157)
830 REM * print the information *
840 LOCATE 13,15:PRINT " ";CT$;" 
850 LOCATE 13,35:PRINT " ";START$;" 
860 PSET(278,172):LINE-(284,178)
870 PSET(278,178):LINE-(284,172)
880 LOCATE 13,60:PRINT STOPT$;" 
890 PSET(565,157):LINE-(565,163)
LOCATE 16,12:PRINT "X-axis : TIME"  Y-axis : TEMPERATURE"
910 LOCATE 16,9:PRINT "Current time - " ;CT$
920 LOCATE 20,9:PRINT "Approx start time - " ;START$
930 LOCATE 22,9:PRINT "Approx finish time - " ;STOPT$
940 LOCATE 18,48:PRINT "Current temp : ";ASLSTART;"C"
950 LOCATE 20,48:PRINT "Start temp : " ;STAT;"C"
960 LOCATE 22,48:PRINT "Stop temp : " ;STOP;"C"
970 VIEW PRINT 24 TO 24
975 REM ** Ask if the setup is suitable **
980 LOCATE 24,15:INPUT "Is this setup suitable (Y/N)" ;K$
990 VIEW PRINT
1000 IF ((K$="y") OR (K$="Y")) THEN CLS ELSE IF ((K$="n") OR (K$="N")) THEN CLEAR:GOTO 120 ELSE BEEP:GOTO 970
1010 REM *** If OK then proceed with calibration ***
1020 CLS:LOCATE 12,15:PRINT "Proceeding with calibration......please wait....";
1030 FOR T=1 TO 10000:NEXT T
1040 REM *** set up the screen ***
1050 GOSUB 1930
1060 REM ** set the start point on the graph **
1070 FOR T=0 TO 25000:NEXT T:WRITE#3,"T"+CHR$(10)
1075 IF LOC(3)>=9 THEN INPUT #3,TEMP$ ELSE GOTO 1070
1080 CT=VAL(MID$(TEMP$,2,7)):X=90
1090 PSET(90,274-((224/(STOPT-STAT))*(CT-STAT)))
1100 REM * Loop round all the demands *
1110 FOR DEMAND=STAT TO STOPT STEP STPT
1115 REM ** Convert temp demand to value acceptable by D/A **
1120 ENT%=(DEMAND+5)*80
1130 GOSUB 2400
1135 REM ** Set the first condition **
1140 CONDITION=ABS(CT-DEMAND)
1150 REM ** Decide whether heating or cooling and plot data while **
1151 REM ** first condition is met **
1160 IF CT>=DEMAND THEN BEEP:STATDS$="Cooling bath..":LOCATE 2,38:PRINT STATDS$;:WHILE CONDITION>.5:GOSUB 2240:CONDITION=ABS(CT-DEMAND):WEND
1170 IF CT<=DEMAND THEN BEEP:STATDS$="Heating bath..":LOCATE 2,38:PRINT STATDS$;:WHILE CONDITION>.5:GOSUB 2240:CONDITION=ABS(CT-DEMAND):WEND
1180 REM ** now check for second condition **
1190 GOSUB 2240:TEST2=CT : OK=0
1200 WHILE (OK>5)
1210 TEST1=TEST2
1220 GOSUB 2400
1230 TEST2=CT
1240 IF STATUS$="Cooling bath.." THEN IF (TEST2-TEST1)<-.003 THEN OK=OK+1
1250 IF STATUS$="Heating bath.." THEN IF (TEST2-TEST1)<.01 THEN OK=OK+1
1260 WEND
1270 REM ** now wait for bath to stabilize (5 mins) **
1280 BEEP:STATUS$="Stabilising...":LOCATE 2,38:PRINT STATUS$;
1290 CT$=TIME$ : MINUTE=VAL(MID$(CT$,4,2))
1295 REM ** STABMIN = minute unit when bath will be satalised **
1300 STABMIN=MINUTE+5
1310 IF STABMIN>59 THEN STABMIN=STABMIN-60
1320 WHILE MINUTE=>STABMIN
1330 GOSUB 2240
1340 CT$=TIME$ : MINUTE=VAL(MID$(CT$,4,2))
1350 WEND
1360 REM *** take readings from psychrometres ***
1370 GOSUB 2720
1380 REM ** write the readings and F25 temp to file **
1390 FOR FILNUM=1 TO TOTSEN
1400 OPEN FILARR$(FILNUM) FOR APPEND AS #1
1410 PRINT #1,USING "#####.#####";(30000-A%(CONNECTION(FILNUM)))/T1;:PRINT #1," "#:PRINT
1415 CLOSE #1
1420 REM ** need to write freq's to screen as well **
1425 IF FILNUM > 10 THEN XLOC=FILNUM-2 ELSE XLOC=FILNUM+8
1430 LOCATE XLOC,59:PRINT (CONNECTION(FILNUM))+1;:LOCATE XLOC,64 : PRINT
USING"#####.#####";(30000-A%(CONNECTION(FILNUM)))/T1;
1450 NEXT FILNUM
1460 NEXT DEMAND
1470 VIEW PRINT:CLS:CLOSE#3 : REM ** i.e. close communication port **
1480 REM ** Now calculate the coefficients using a routine from CALEDIT.BBC **
1490 LOCATE 12,18:PRINT "Please wait, calculating coefficients...":FOR TTIM=1 TO 8000:NEXT TTIM
1495 CLS
1500 FOR LOOP=1 TO TOTSEN
1510 FILE$=FILARR$(LOOP)
1515 GOSUB 3450
1520 NEXT LOOP
1521 DIM NDMBER$(30)
1522 REM ** Now need to copy over the temp files to the database **
1523 FOR LOOP=1 TO TOTSEN
1524 TEMPFILE$=FILARR$(LOOP) : LFILE$=LSTFILE$(LOOP)
1525 GOSUB 7170
1526 NEXT LOOP
1527 NEXT LOOP
1531 REM ** ask if output to printer required **
1532 LOCATE 10,15:PRINT "Do you wish to print out the calibration files ";
1533 LOCATE 10,63:INPUT K$ 
1534 IF ((K$="y")OR(K$="Y")) THEN LPTFLAG=1 ELSE IF ((K$="n")OR(K$="N")) THEN LPTFLAG=0 ELSE
1535 IF LPTFLAG=1 THEN FOR LOOP=1 TO TOTSEN : SHLCMD$="print "+:FILARR$(LOOP) : SHELL SHLCMD$: NEXT
1536 LOOP
1540 CLS : LOCATE 12,14:PRINT"Callibration finished...results written to database"
1550 END
1560 REM *** End of main program...all routines follow on from here ***
1570 REM********************************************************************************
1580 REM ** Routine to work out time for heating/cooling ***
1590 MINUTE-MINUTE+1
1600 IF MINUTE>59 THEN MINUTE=0 : HOUR-HOUR+1
1610 IF HOUR>23 THEN HOUR=0 : DAY=DAY+1
1620 LENGTHX-LENGTHX+1
1630 RETURN
1640 REM **---routine to cool---**
1650 REM **---routine to cool---**
1660 WHILE CT > TEMP
1670 CT=CT-.13
1680 GOSUB 1580
1690 WEND
1700 REACHED(INDEX)=LENGTHX
1710 STABTIME=LENGTHX+5
1720 WHILE LENGTHX < STABTIME
1730 GOSUB 1580
1740 WEND
1750 STABALISED(INDEX) = LENGTHX
1760 STR$ = STR$(HOUR) + " : " + STR$(MINUTE) + " : " + STR$(SECOND)
1770 RETURN
1780
1790 REM ---------------------------------------------------------------
1800 REM *** routine to heat ***
1810 WHILE CT < TEMP
1820 CT = CT + 0.6
1830 GOSUB 1580
1840 WEND
1850 REACHED(INDEX) = LENGTHX
1860 STABTIME = LENGTHX + 5
1870 WHILE LENGTHX < STABTIME
1880 GOSUB 1580
1890 WEND
1900 STABALISED(INDEX) = LENGTHX
1910 STOPT$ = STR$(HOUR) + " : " + STR$(MINUTE) + " : " + STR$(SECOND)
1920 RETURN
1930 REM ---------------------------------------------------------------
1940 REM *** Routine to set up the screen to ***
1950 REM *** display real-time temperature ***
1960 SCREEN 9:COLOR 7,1:CLS
1970 VIEW  PRINT 2 TO 23
1980 LOCATE 2,28:PRINT "Status : "; LOCATE 2,38:PRINT STATUS$;
1990 LINE(205,9)-(425,31),7,B
2000 LINE(5,5)-(635,345),7,B
2010 REM ** Draw the axes **
2020 PSET(90,280):LINE-(90,40)
2030 REM ** Draw dotted lines where demands are **
2040 FOR DOTLINE = STAT TO STOT STEP STPT
2050 LINE(90,274 - ((224/(STOT-STAT)) * (DOTLINE-STAT))) - (400,274 - ((224/(STOT-STAT)) * (DOTLINE-STAT))),&HCCCC
2060 NEXT DOTLINE
2070 REM ** Label y-axis **
2080 TIKY = no tick marks on y-axis, RCP = cursor position **
2090 TIKY = 8:RCP = 20
2100 FOR I = 0 TO TIKY
2110 PSET(87,274 - (2*(14*I))):LINE STEP(0,0)-STEP(5,0)
2120 LOCATE (RCP-(I*2)),5
2130 PRINT USING "*##.#"; LABEL
2140 LABEL = LABEL + INCR
2150 NEXT I
2160 LOCATE 23,16:PRINT "Demand temp : ";
2170 LOCATE 23,41:PRINT "Actual temp (F25) : ";
2180 REM ** now draw the box that shows freq readings **
2190 LINE (450,65)-(590,260),7,B
2200 LOCATE 6,62:PRINT TIMES$;
2210 LOCATE 7,59:PRINT "EFT";STOPT$
2220 LOCATE 8,59:PRINT "Ctr Freq";
2230 RETURN
2240 REM ---------------------------------------------------------------
2240 REM *** Routine to plot data and get another reading ***
2250 REM * need to check if x > 400 and if so refresh the graph *
2260 X=X+1
2270 IF X>400 THEN GOSUB 1930: X-90 :PSET(X,274-((224/(ST0T-STAT))*(CT-STAT)))
2280 IF STATUS$="Cooling bath.." THEN COLOR 9,1
2290 IF STATUS$="Heating bath.." THEN COLOR 12,1
2300 IF STATUS$="Stabilising..." THEN COLOR 10,1
2310 LINE-(X,274-((224/(ST0T-STAT))*(CT-STAT)))
2320 COLOR 7,1
2330 LOCATE 23,30:PRINT USING "##.##";DEMAND;:LOCATE 23,61:PRINT USING"#.#*#";CT;
2340 LOCATE 6,62 :  PRINT TIME$;
2345 REM  **
2350 FOR T=0 TO 25000:NEXT T:WRITE#3,"T''+CHR$(10)
2360 IF LOG(3)>-9  THEN INPUT *3,TEMP$  ELSE GOTO 2350
2370 CT='VAL(MID$(TEMP$,2,7))
2380 RETURN
2390 REM
2400 REM *** Routine to to issue a demand to the A/D converter ***
2410 REM ** Call initial setup routine **
2420 GOSUB 3270
2430 GOSOB:  REM Set channels 1-8 and 16-21 as outputs
2440 E$=HEXS(ENT%)
2450 HL%=LEN(E$)
2460 GOSUB 2570:  REM Convert number to binary in a%(array)
2470 C$=CHR$(79)  + CHR$(0):A*(20)-0:CALL  AM1(A*(0),B(0),C$):  REM Output to D/A converter
2480 A%(20)=1:CALL AM1(A%(0),B(0),C$):  REM Latch reading in D/A
2490 RETURN
2500 REM ** Routine to set channels 1-8 and 16-21 as outputs **
2510 CS=CHR$(79) + CHR$(0)
2520 FOR I%=0 TO 7:A%(I%)=1:NEXT
2530 FOR I%=15 TO 20:A%(I%)='1:NEXT
2540 CALL AM1(A%(0),B(0),C$)
2550 RETURN
2560 REM ** Routine to convert number to binary **
2570 FOR CH%=1 TO 3: REM hex byte number
2580 IF CH%>HL% THEN CH$="0"  ELSE CH$=MID$(E$,HL%-CH%+1,1)
2590 IF CH%<3 THEN 0F%=4*(CH%-1) ELSE 0F%-16:  REM 0F% is channel offset
2600 VCH%-ASC(CH$)-48
2610 IF VCH%>9 THEN VCH%=VCH%-7: REM decimal value of hex char.
2620 :
2630 FOR B%=3 TO 0 STEP -1:  REM convert hex char. to binary
2640 BV%=2**B% 
2650 BT%-INT(VCH%/((BV%))):VCH%=VCH%-%BT%*BV%
2660 A%(OF%+B%)=BT%
2670 NEXT B%
2680 :
2690 NEXT CH%
2700 RETURN
2710 REM ---------------------------------------------
2720 REM *** Routine to get frequency values from the sensors ***
2730 REM *** Call initial setup routine ***
2740 GOSUB 3270
2750 T1=10:REM ** Sets counter gate duration in seconds **
2760 REM ** Set up the counters **
2770 GOSUB 3190
2780 REM ** Set the initial counting values **
2790 GOSUB 3010
2800 REM ** Start counting **
2810 GOSUB 3090
2820 REM ** Test for end of freq measurement **
2830 FOR I=1 TO 10000
2840 CS="X"+CHR$(0):A%(0)=&H651D:B(0)-4
2850 CALL AM1(A%(0),B(0),CS):REM Test for count in progress
2860 IF (A%(0) AND &H1)=0 THEN I=10000
2870 NEXT
2880 REM ** Read the counter **
2890 CS="X"+CHR$(0):CALL AM1(A%(0),B(0),CS)
2970 RETURN
2980 REM
2990 REM *** Set the values at which the counters start counting ***
3000 REM ** set ctr3 div 50*T1, ctr4 div 65536, ctrs5-7 div 30000 **
3010 CS="X"+CHR$(0)
3020 A%(2)=50*T1:A%(3)=-25536
3030 FOR I=4 TO 23
3040 A%(I)=30000
3050 NEXT I
3060 CALL AM1(A%(0),B(0),CS)
3070 RETURN
3080 REM *** Routine to initiate the counting process ***
3090 CS="X"+CHR$(0)
3100 FOR I=4 TO 23
3110 A%(I)=&H6E1E
3120 NEXT I
3130 A%(3)=&H651E
3140 A%(2)=&H941E
3150 B(0)=0
3160 CALL AM1(A%(0),B(0),CS)
3170 RETURN
3180 REM *** Routine to set up counters/timers ***
3190 CS="X"+CHR$(0)
3200 A%(2)=&H6018:A%(3)=&H6018
3210 FOR I=4 TO 23
3220 A%(I)=&H6018
3230 NEXT I
3240 B(0)=0:CALL AM1(A%(0),B(0),CS)
3250 RETURN
3260 REM -----------------------------------------------
3270 REM *** INITIAL SETUP SUBROUTINE ***
3280 ' The following data for the machine code is improved over the example
3290 ' in the manual. This code checks for the presence of the driver
3300 ' before trying to use it.
3310 DATA &H50, &HE8, &H18, &H00, &H3D, &HFF, &HFF, &H74, &H0C, &H58, &HFA, &HB8, &H59, &H47, &HCD, &H60, &H90, &H90, &HCA, &H06
3320 &H00, &HB6, &HFF, &HFF, &H5D, &HCA, &H06, &H00, &H56, &H06, &HBB, &H00, &H00, &H8E, &HCD, &H26, &HA1, &H80, &H01, &H3D
3330 &H00, &H80, &H74, &H1A, &HB8, &HFF, &HFA, &H50, &H00, &H90, &H3C, &H74, &HFF, &HFF, &H07, &H5E, &H8B
3340 &H00, &H80, &H00, &H5C3, &H88, &HFF, &H74, &H8B, &H00, &H80, &H00, &H80, &H00, &H90, &H3D, &H8F, &H8C, &H26, &H8A, &H04, &H3C, &H3D, &H75, &H06, &H07, &H5E, &H8B
3350 &H00, &H00, &H00, &H90, &H00, &H00, &H80, &H00, &H80, &H00, &H00, &H80, &H00, &H00, &H00, &H80, &H00, &H00, &H80, &H00, &H00, &H00, &H00, &H00, &H00, &H00, &H00, &H00, &H00
3330 :  
3340 ADDR=$1703: REM Address of card + 3 
3350 REM APROGS and APROG1$ MUST be the first strings defined. 
3360 DEF SEG:APROG$=SPACE$(80):APROG1$=SPACE$(80):REM Strings will contain routines to CALL driver 
3370 A%=VARPTR(APROG$):AX%=VARPTR(APROG1$) 
3380 AM1=PEEK(A%+1)+PEEK(A%+2)*256:AM2=PEEK(AX%+1)+PEEK(AX%+2)*256 
3390 RESTORE 3310 
3400 FOR A=0 TO 76 
3410 READ A%:POKE A+AM1,A%:POKE A+AM2,A%:REM Install routines to CALL driver 
3420 NEXT:POKE AM2+19,16:POKE AM2+26,16 
3430 RETURN 
3440 REM ---------------------------------------------------------------
3450 REM *** This routine has to run under GW BASIC/d so that ***
3460 REM *** mathematical routines can use double precision numbers ***
3470 REM access data file and read in data 
3480 N=4: REM * Set 3 degrees of fit * 
3490 Z=0: O=1: K=12 
3500 G9#=-1E-18 : G8#=1E+18 : G7#-9.999999E+35 : G6#=-1E-36 
3510: 
3520: 
3530 REM zero data arrays 
3540 FOR I%=0 TO 40 : X(I%)=0:Y(I%)=0:II(I%)=0 : NEXT I% 
3550: 
3560 REM get the data file 
3580 OPEN FILES FOR INPUT AS #1 
3590: 
3600 REM ** Read in data **
3610 REM ** First skip the header **
3620 FOR SKIP=1 TO 12 : INPUT#1,SCRAP$ : NEXT SKIP 
3630 COUNT%=0 
3640 WHILE (NOT(E0F(1))) 
3650 COUNT%=COUNT%+1 
3660 INPUT#,X(COUNT%),Y(COUNT%) 
3670 II(COUNT%)=COUNT% 
3680 WEND 
3690 IF COUNT%<>M THEN M=COUNT% 
3700 MF=M 
3710: 
3720 CLOSE#1 
3725 REM ** Now open the same file for writing the results to **
3730 OPEN FILES FOR APPEND AS #2 
3740: 
3750 REM * Initialise standard error array to missing values * 
3760 FOR I%=1 TO 8 : STDERR(I%)=99999 : NEXT I% 
3770: 
3780 REM calculates the means for the data 
3790 T7#-Z:T7#-Z:T8#-Z:W7#=Z 
3800: 
3810 FOR I%=1 TO MF 
3820 W7#=W7#*X(II(I%)):T7#=T7#*Y(II(I%)):T8#=T8#*Y(II(I%))^2 
3830 NEXT 
3840 T9#=(MF*T8#-T7#^2)/(MF^2-MF) 
3850: 
3860 GOSUB 4210
3870 GOSUB 4810
3880 REM ** Print out all the results **
3950 :
3970 FOR L=1 TO MF
3980 Q#-Y(L)-C#(L)
4000 PRINT USING "##*.###_";Q# :
4010 IF C#(L)=0 THEN PRINT #2,"INFINITE"
4020 NEXT L
4030 :
4040 PRINT #2,"Std  error  of estimate for Y = "
4060 REM print out coefficients
4070 PRINT #2,"Term  Coefficient"
4110 :
4130 I2-J-1 : TERMS="C"+RIGHT$(STR$(I2),1)
4140 PRINT #2, TERMS;
4150 PRINT #2,"";U#$J
4160 NEXT
4170 :
4180 CLOSE #2
4190 RETURN

4210 REM *** Routine arith1 (Copied from CALEDIT.BBC) ***
4230 FOR I%=1 TO MF: P#(I%)=Z: Q#(I%)=0: NEXT
4240 FOR I%=1 TO N: A#(I%)=Z: B#(I%)=Z: S#(I%)=Z: NEXT
4250 E1#=Z: F1#=Z: W1#=MF:N4=K: I=1: K1=2
4260 :
4270 IF N=0 THEN GOTO 4330
4280 K=1:N4
4290 :
4300 REM program loops to next line
4310 W#-Z
4320 :
4330 FOR L=1 TO MF: W#-W#+Y(L)*Q#(L): NEXT
4340 :
4350 S#(I%)=W#/W1#
4360 IF I-N4=0 THEN GOTO 4790
4370 IF I-MF=0 THEN GOTO 4790
4380 E1#-Z
4390 FOR L=1 TO MF
4400 A9#=ABS(Q#(L))
4410 IF A9#<O9# THEN GOTO 4550
4420 X9#=ABS(X(L))
4430 IF X9#<O6# THEN GOTO 4510
4440 L2#-.434295*LOG(X9#)+(.434295*LOG(A9#))
4450 IF L2#<36 THEN GOTO 4540
4460 REM SO X&Q TOO BIG FOR X*Q^2
4470 E1#=O7#
4480 L2#-36
4490 L=MF+1
4500 GOTO 4550
4510 IF A9#<1 THEN GOTO 4550
4520 IF A9#<08# THEN GOTO 4540
4530 REM SO Q TOO BIG FOR Q^2
4540 E1#=E1#+X(II(L))*A9^2
4550 NEXT
4560 IF L2#-(.434295*LOG(W1#))>-36 THEN GOTO 4590
4570 E1#=0
4580 GOTO 4600
4590 E1#=E1#/W1#
4600 A#(I+1)=E1#
4610 W#=-Z
4620 FOR L=1 TO MF
4630 V#=(X(II(L))-E1#)*Q#(L)-F1#*P#(L)
4640 P#(L)=Q#(L)
4650 Q#(L)=V#
4660 V9#=ABS(V#)
4670 IF V9#<09# THEN GOTO 4730
4680 IF V9#<08# THEN GOTO 4720
4690 W#=07#
4700 L=MF+1
4710 GOTO 4730
4720 W#=W#+V#*V#
4730 NEXT
4740 F1#=M#/W1#
4750 B#(I+2)=F1#
4760 W1#=W#
4770 I=I+1
4780 GOTO 4310
4790 RETURN
4800 REM
4810 REM *** Routine arith2 (Copied from CALEDIT.BBC) ***
4820 :
4830 FOR L=1 TO 13:G#(L)=Z:NEXT
4840 :
4850 G#(2)=0
4860 FOR J=1 TO N
4870 S1#=Z
4880 FOR L=2 TO N+1
4890 IF L=2 THEN GOTO 4910
4900 G#(L)=G#(L)-A#(L-1)*G#(L-1)-B#(L-1)*G#(L-2)
4910 S1#=S1#*S#(L-1)*G#(L)
4920 NEXT
4930 U#(J)=S1#
4940 L=N+1
4950 FOR J=2 TO N
4960 G#(L)=G#(L-1)
4970 L=L-1
4980 NEXT
4990 G#(2)=Z
5000 NEXT
5010 :
5020 PRINT " "
5030 T=Z
5040 FOR L=1 TO MF
5050 C#(L)=Z
J-N
FOR I2=1 TO N
C#(L)=(C#(L)*X(I2(L)))+O#(J)
J=J-1
NEXT
T3#=Y(I2(L))-C#(L)
T#-T#+T3#^2
NEXT
IF MF<>N THEN GOTO 5170
T5#=0
GOTO 5180
T5#=T#/(MF-N)
Q7#='1-T#/T9#*(MF-1))
IF MFON THEN GOTO 5170
T5#=0
GOTO 5180
DETERM = INT(Q7#*10000)/10000
STDERR(N-1)=SQR(T5#)
RETURN
REM "-------------------------------"
REM *** Routine to get all the sensor details ***
REM * Set up screen *
KEY OFF
SCREEN 9:COLOR 7,1:CLS
LINE (5,5)-(635,345),,B
LOCATE 4,25:PRINT"Psychrometer Calibration"
LOCATE 5,24:PRINT "Control and Logging program"
REM ** USERNAMES = name of present operator for filing purposes **
LOCATE 7,18 : INPDT"Please enter your user name ";USERNAME$
REM ** P = no. psychrometers, S = no. SST's **
REM ** A = no. air temp's, O = no. other sensors **
P=0:S=0:A=0:0=0
LOCATE 10,26:PRINT"Psychrometers ->";P
LOCATE 12,26:PRINT"SST's ";S
LOCATE 14,26:PRINT"Air Temps ";A
LOCATE 16,26:PRINT"Others ";O
LOCATE 20,10:PRINT"Enter number of sensors and press <ESC> when finished"
LOCATE 21,22  :  PRINT"Move around using cursor keys"
REM * Get all the details *
DN$=CHR$(0)+CHR$(B0):UP$=CHR$(0)+CHR$(72)
X=10:Y'=46:KY$=" 
WHILE KY$<>CHR$(27)
IF (X=10) AND ((KY$<>UPS)AND(KYS<>DNS)AND(KYS<>CHR$(27))) THEN LOCATE X,46:PRINT KYS;
IF (X=12) AND ((KY$<>UPS)AND(KYS<>DNS)AND(KYS<>CHR$(27))) THEN LOCATE X,46:PRINT KYS;
IF (X=14) AND ((KY$<>UPS)AND(KYS<>DNS)AND(KYS<>CHR$(27))) THEN LOCATE X,46:PRINT KYS;
IF (X=16) AND ((KY$<>UPS)AND(KYS<>DNS)AND(KYS<>CHR$(27))) THEN LOCATE X,46:PRINT KYS;
IF (KY$=DNS) AND ((X=>10)AND(X<=14)) THEN LOCATE X,Y-3:PRINT" ";X=X+2:LOCATE X,Y-3:PRINT" ";
IF (KY$=UPS) AND ((X=>12)AND(X<=16)) THEN LOCATE X,Y-3:PRINT" ";X=X-2:LOCATE X,Y-3:PRINT" ";
5540 WEND
5550 REM ** now scan the screen for the values **
5560 P=VAL(CHR$(SCREEN(10,46)))
5570 S=VAL(CHR$(SCREEN(12,46)))
5580 A=VAL(CHR$(SCREEN(14,46)))
5590 O=VAL(CHR$(SCREEN(16,46)))
5595 REM ** Check if no. of sensors within specified limits **
5600 IF (P>6) OR (S>4) OR (A>2) OR (O>2) THEN BEEP : LOCATE 24,6 : PRINT "Max no of sensors = 6,4,2,2 respectively...please re-enter" : FOR LOOP=1 TO 11000 : NEXT LOOP : CLEAR : GOTO 5280
5605 REM ** TOTSEN = total number of sensors to be calibrated **
5610 TOTSEN=(P*2)+S+A+O : REM * Wet and dry sensor for each psychrometer *
5620 REM *** deal with all the sensor group seperately ***
5625 REM ** CP() = array to hold cursor positions for screen setup purposes **
5626 REM ** ID$(P+S+A+O) = array to hold the id's of every sensor **
5627 REM ** CH2$(12) = array to hold data read in from the 'details.mmn' files **
5628 REM ** FILARR$(P+S+A+O) = array to file names of each new sensor cal. file **
5629 REM ** LSTFILE$(P+S+A+O) = array to hold filenames of each sensor list file **
5630 REM ** TOTSEN = total number of sensors to be calibrated **
5631 REM
5632 DIM CP(P+S+A+O):DIM ID$(P+S+A+O):DIM CH2$(12):DIM FILARR$(TOTSEN):II=1:N0MSEN=1
5640 REM ** Initialise the array that holds the id's **
5650 FOR INDEX=1 TO (P+S+A+O)
5660 ID$(INDEX)=""
5670 NEXT INDEX
5675 REM ** ZZZ = common variable holding the number of each sensor **
5680 REM *** first the psychrometers ***
5690 IF P<>0 THEN ZZZ=P : STLOOP=P : SPLOOP=P : SENSORS=" Psychrometer " : FILNM$="tt" : CLS :
LOCATE 12,15 : PRINT "Enter the psychrometer details...." : FOR LOOP=1 TO 10000 : NEXT : GOSUB 5980
5700 REM ** now deal with the sst's **
5710 IF S<>0 THEN ZZZ=S : STLOOP=P+1 : SPLOOP=P+S : SENSORS=" SST " : FILNM$="sp" : CLS :
LOCATE 12,15 : PRINT "Enter the SST details...." : FOR LOOP=1 TO 10000 : NEXT : GOSUB 5980
5720 REM ** thirdly, deal with the air temperatures **
5730 IF A<>0 THEN ZZZ=A : STLOOP=P+S+1 : SPLOOP=P+S+A : SENSORS=" Air temp " : FILNM$="td" : CLS :
LOCATE 12,15 : PRINT "Enter the air temp details...." : FOR LOOP=1 TO 10000 : NEXT : GOSUB 5980
5740 REM ** lastly, deal with any other sensors **
5750 IF O<>0 THEN ZZZ=O : STLOOP=P+S+A+1 : SPLOOP=P+S+A+O : SENSORS=" Others " : CLS :
LOCATE 12,15 : PRINT "Enter the other sensor details...." : FOR LOOP=1 TO 10000 : NEXT : GOSUB 5980
5760 REM *** now tell the user which connections to make ***
5770 CLS : NUMSEN=1
5780 LOCATE 2,10 ; PRINT"Please make the following connections ":
5790 FOR I=1 TO P
5800 LOCATE I+3,5;PRINT"Psychrometer ";I;" Sensor ";ID$(NUMSEN);" Connect to counters ";I*(I+2)+2;I*(I+2)+3+1;
5810 CONNECTION(CONNUM)=(I*2)+2;CONNECTION(CONNUM+1)=(I*2)+3+NUMSEN=NUMSEN+1;CONNUM=CONNUM+2
5820 NEXT I
5830 FOR I=1 TO S
5840 LOCATE I+P+4,5;PRINT"SST ";I;" Sensor ";ID$(NUMSEN);" Connect to counter ";I+15+1;
5850 CONNECTION(CONNUM)=I+15:NUMSEN=NUMSEN+1;CONNUM=CONNUM+1
5860 NEXT I
5870 FOR I=1 TO A
5880 LOCATE I+P+S+5,5;PRINT"Air temp ";I;" Sensor ";ID$(NUMSEN);" Connect to counter ";I+19+1;
5890 CONNECTION (CONNUM) = I+19: NUMSEN = NUMSEN + 1: CONNUM = CONNUM + 1
5900 NEXT I
5910 FOR I = 1 TO 0
5920 LOCATE I + P + A + 6, 5: PRINT "Others "; I; " Sensor "; ID$(NUMSEN); " Connect to counter "; I+21+1;
5930 CONNECTION (CONNUM) = I+21: NUMSEN = NUMSEN + 1: CONNUM = CONNUM + 1
5940 NEXT I
5941 AA = CSRLIN
5945 LOCATE AA + 2, 10: PRINT "Press <RETURN> at anytime to continue";
5946 LOCATE AA + 4, 5: PRINT "Current counter readings :";
5950 WHILE INKEY$ <> CHR$(13)
5951 GOSUB 2720
5952 FOR FILNUM = 1 TO TOTSEN
5953 IF FILNUM > 7 THEN YPOS = 35: SUBT = 7 ELSE YPOS = 0: SUBT = 0
5954 LOCATE AA + 5 + FILNUM - SUBT, YPOS: PRINT CONNECTION (FILNUM) + 1; " "; PRINT USING "#####.####"; (30000 - A%(CONNECTION (FILNUM))) / T1
5955 NEXT FILNUM
5956 MEND
5957 RETURN: REM ** i.e. return to the main program at line 60 **
5970 REM ------------------------------------------------------------------------
5980 REM ** Routine to get the details of each different sensor type **
6000 REM ** setup screen **
6010 J = STLOOP
6020 GOSUB 6170
6030 REM *** get all the sensor details ***
6040 WHILE J < SLOOP
6050 LOCATE CP(J), 4: PRINT " ";
6060 LOCATE CP(J), 37: INPUT ID$(NDMSEN)
6070 GOSUB 6350
6080 NDMSEN = NDMSEN + 1
6090 REM ** Now redraw the screen **
6100 GOSUB 6170
6110 J = J + 1
6120 WEND
6130 WHILE INKEY$ <> CHR$(27): WEND
6140 CLS
6150 RETURN
6160 REM ------------------------------------------------------------------------
6170 REM ** set up the screen **
6180 X1 = 10: Y1 = 10: X2 = 380: Y2 = 350: T2 = SLOOP
6190 COLOR 7, 1: CLS
6200 LOCATE 4, 51: PRINT " Enter ID for each sensor ";
6210 LOCATE 5, 51: PRINT " followed by <RETURN> "
6220 LOCATE 6, 53: PRINT " eg. VI1006 ";
6230 LOCATE 0, 51: PRINT "Press <ESC> when finished ";
6240 FOR I = 1 TO ZZ|
6250 LINE (X1, (Y2 / ZZ) * (I - 1)) + 5) - (X2, (Y2 / ZZ) * I) - 5), 3, BF
6260 LINE (X1 + 5, (Y2 / ZZ) * (I - 1)) + 5) - (X2 - 5, (Y2 / ZZ) * I) - 5), 64, B
6270 COP = ((Y2 / ZZ) * I) - ((Y2 / ZZ) * (I - 1))) / 2 + ((Y2 / ZZ) * (I - 1))
6280 CP(I2) = CINT((CPOS/14) + .5)
6290 COLOR 2, 1: LOCATE CP(I2), 8: PRINT SENSOR$[I];
6300 COLOR 7, 1: LOCATE CP(I2), 26: PRINT " Sensor ID : "; ID$(I2);
6310 I2 = I2 + 1
6320 NEXT I
6330 RETURN
6340 REM -----------------------------------------
6350 REM ** draw the window to show all details **
6360 GOSUB 6740
6370 LOCATE 6,17:PRINT SENSORS$;J;
6380 LOCATE 6,38:PRINT "Sensor ID : ";ID$(NUMSEN);"
6390 REM ** first get the multimet number (held in MMNS ) **
6400 MMNS=""
6405 REM ** Construct the filename **
6410 FILEIS="d:\ametdb\"+FILM$+ID$(NUMSEN)+".1"
6411 IF FILEIS="t" THEN LSTFILE$(IX)=FILEIS:LSTFILE$(IX+1)=FILEIS:IX=IX+2 ELSE
6420 OPEN FILEIS FOR INPUT AS #1
6425 REM ** TEMP$ = temporary var, CH$ = text taken from the file **
6430 WHILE (NOT(E0F(1)))
6440 INPUT#1,TEMP$:IF TEMP$<"" THEN CH$=TEMP$:MMNS=RIGHT$(CH$,3)
6450 WEND
6460 REM ** get today's JDAY for comparison with any existing ones in DB **
6470 DAY=VAL(MID$(DATE$,4,2)):MONTH=VAL(LEFT$(DATE$,2)):YEAR$=RIGHT$(DATE$,2)
6480 REM ** convert to jday **
6490 JDAY$=STR$(VAL(MID$("000031059090120151181212243273304334",((MONTH-1)*3)+1,3))+DAY)
6500 JDAY$=RIGHT$(JDAY$,3)
6510 REM ** TEMPS,TEMP2$ = temporary vars **
6515 REM ** VCODES$ = file version code, CURJDAYS$ = jday of most recent cal file
6520 TEMPS=RIGHT$(CH$,5): VCODES=LEFT$(TEMPS,1)
6530 TEMPS2=RIGHT$(CH$,10): CURJDAYS=LEFT$(TEMP2S,3)
6535 REM ** Test for a version code and if JDAY is same then increment it **
6536 REM ** ASVAL = ascil value of version code **
6540 IF ASC(VCODES$)<65 THEN ASVAL-65 ELSE IF CURJDAYS$=JDAY$ THEN ASVAL=ASC(VCODES$)+1 ELSE ASVAL=6
6550 VCODES$=CHR$(ASVAL)
6560 REM ** now get the remaining details **
6570 X=1
6580 FILE2S$="d:\ametdb\metdb_da\details."+MMNS
6590 OPEN FILE2S FOR INPUT AS #2
6600 WHILE (NOT(E0F(2)))
6610 INPUT#2,CH2$(X):X=X+1
6620 WEND
6630 LOCATE 8,17:PRINT " Multimet No : ";MMNS;"
6640 LOCATE 10,17:PRINT CH2$(10);
6650 LOCATE 12,17:PRINT CH2$(6);
6660 CLOSE #1,#2
6670 REM ** Check whether it the required sensor **
6680 REM ** If so then write the details to a file **
6690 LOCATE 16,17:PRINT " Is this the required sensor (Y/N) ";
6700 LOCATE 16,53:INPUT K$ 
6710 IF ((K$="y")OR(K$="Y")) THEN GOSUB 6820 : RETURN ELSE IF ((K$="n")OR(K$="N")) THEN J=J-1 : NUMSEN=NUMSEN-1 : RETURN ELSE BEEP : GOTO 6700
6720 RETURN
6730 REM -----------------------------------------
6740 REM ** Routine to draw the window **
6750 AA=320 : BB=175
6760 FOR CC=1 TO 125 STEP 2
6770 6770 LINE ((AA-CC)-95,BB-CC)-((AA+CC)+95,BB+CC),B,BF
6780 NEXT CC
GOSUB 6870 : STYPE$="tw"
GOSDB 6870
6790 LINE (105,55)-(535,295),7,B
6800 RETURN
6810 REM -----------------------------------------------
6820 REM ** Routine to check what to write to files **
6825 REM ** FILNM$ = instrument type for filename purposes **
6826 REM ** STYPE$ = sensor type **
6827 REM ** ELEC$ = sensor electronics number **
6830 IF FILNM$="tt" THEN STYPE$="td" ; ELEC$=RIGHT$(CH2$(10),3) ; GOSUB 6870 : STYPE$="tw" ;
6835 TEMP$=RIGHT$(CH2$(10),9) ; ELEC$=LEFT$(TEMP$,4) ; GOSUB 6870
6840 IF FILNM$="sp" THEN STYPE$="sp" ; ELEC$=RIGHT$(CH2$(10),3)
6850 RETURN
6860 REM -----------------------------------------------
6870 REM ** Routine to open a data file and write the relevant details **
6880 REM ** Open the new file **
6890 FILE3$="c:"+STYPE$+JDAY$+YEAR$+VCODE$+".'"+MMN$
6900 FILARR$(II)=FILE3$ : IF TOTSEN>II THEN 11=11+1
6910 OPEN FILE3$ FOR OUTPUT AS #3
6920 PRINT #3,"Multimet Sensor Calibration File"
6930 PRINT #3,""
6940 PRINT #3,"Sensor ID : ";STYPE$;ID$(J)
6950 PRINT #3,"Electronics : ";ELEC$
6960 PRINT #3,"Date : ";DATE$
6970 PRINT #3,"Operator : ";USERNAME$
6980 PRINT #3,"Standard : F25 S3422"
6990 PRINT #3,""
6991 PRINT #3,"Poly. deg : 3"
6992 PRINT #3,"Equation : hex=hex/6.25"
6993 PRINT #3,"Equation : val=C0+C1*hex+C2*hex^2+C3*hex^3+C4*hex^4"
6994 PRINT #3,""
7000 CLOSE #3
7060 RETURN
7070 REM -----------------------------------------------
7080 REM ** Error routine ***
7090 REM ** First, find type of error **
7100 REM * File not found *
7110 IF ERR=53 THEN LOCATE 8,17:PRINT " No such sensor found in database ";:BEEP:J=J-1:NUMSEN=NUMSEN+1:CLOSE#LOCATE 16,17:PRINT" Press <RETURN> to continue ";:WHILE INKEY$<CHR$(13):wend:RESE 6720
7120 REM * Disk not ready - Path not found *
7130 IF (ERR=71) OR (ERR=76) THEN LOCATE 8,17:PRINT" Please ensure that the database is ";:LOCATE 9,17:PRINT" ready for access ";:BEEP:J=J-1:NUMSEN=NUMSEN+1:CLOSE#LOCATE 16,17:PRINT" Press <RETURN> to continue ";:WHILE INKEY$<CHR$(13):wend:RESUME 6720
7140 REM * Device I/O error - Device timeout *
7150 IF (ERR=57) OR (ERR=24) THEN BEEP#CLS#LOCATE 12,21:PRINT"An error hasoccurred...press <RETURN> to re-initialise F25";CLOSE#RESUME 360
7160 REM -----------------------------------------------
7170 REM ** Routine to convert temporary files to multimet database ***
7180 REM ** files and update the list and history files ***
7210 OPEN TEMPFILE$ FOR INPUT AS #1
7215 MMFS$="d:\ametdb\ometdb_da\"+RIGHT$(TEMPFILE$,12)
7216 OPEN MMFS$ FOR OUTPUT AS #2
7217 PRINT #2,"\&O:A,12,72%"
7220 REM ** Get the first 11 lines as they stay the same **
7230 FOR I=1 TO 11
7240 INPUT #1,TEMP$
7250 PRINT #2,TEMP$  
7260 NEXT I  
7270 REM ** Now get the coefficients **  
7280 WHILE LEFT$(TEMP$,4)<>"Term"  
7290 INPUT #1,TEMP$  
7300 WEND  
7310 PRINT #2,TEMP$  
7320 WHILE NOT(EOF(1))  
7330 INPUT #1,TEMP$  
7340 PRINT #2,TEMP$  
7350 WEND  
7360 PRINT #2,"C4";PRINT #2,"C5";PRINT #2,"C6";PRINT #2,"C7"  
7370 PRINT #2,"Freq. Temp. Y-Cal Diff"  
7380 REM ** File pointer is at end therefore close file to reset it **  
7390 CLOSE #1  
7400 OPEN TEMPFILES FOR INPUT AS #1  
7410 REM ** Skip the first 12 lines **  
7420 FOR I=1 TO 12  
7430 INPUT #1,TEMP$  
7440 NEXT I  
7450 REM ** Now get the remaining values **  
7460 J=1:INPUT #1,NUMBERS(J)  
7470 WHILE LEFT$(NUMBERS(J),3)<>"Std"  
7480 J=J+1  
7490 INPUT #1,NUMBERS(J)  
7500 WEND  
7510 FOR I=1 TO (J-1)/2  
7520 PRINT #2,"Cal: ";:PRINT #2,NUMBERS(I);":NUMBERS(I+((J-1)/2))  
7530 NEXT I  
7540 PRINT #2,%B,12,60%:C,12,48%:D,12,36%:E,12,24%:F,12,12%  
7550 REM ** Append the list files in the database **  
7560 TEMPNAME$=RIGHT$(MMFS$,22)  
7570 OPEN LFILES FOR APPEND AS #1  
7580 PRINT #1,TEMPNAME$  
7590 CLOSE #1  
7600 REM ** Update the history file in the database **  
7610 FILE4$="d:\metdb\metdb_da\history."+RIGHT$(TEMPFILE$,3)  
7620 OPEN FILE4$ FOR APPEND AS #1  
7630 PRINT #1,""  
7640 PRINT #1,DATES;" Auto Callibrated"  
7650 CLOSE #1  
7660 RETURN
1000 REM ************************************************************
1005 REM **************************** SEND.BAS by RWP 8/11/90 ****************************
1010 REM **********************************************************************
1011 REM
1012 REM =====  Declare all arrays.....
1013 REM
1014 DIM FREQ$(20), WIND$(20), CALFILE$(100)
1015 DIM X(40), Y(40), II(40)
1016 DIM P#(40), Q#(40)
1017 DIM A#(40), B#(40), S#(40), G*(40)
1018 DRIVE$="D:"; PT%=1
1019 REM
1020 REM =====  Set error trap.....
1021 REM
1022 ON ERROR GOTO 40000
1023 REM
1024 REM =====  Call subroutine to set up Ctrl Esc key trap.....
1025 REM
1026 REM 14000: KEY OFF
1027 REM
1028 SCREEN 9
1029 REM
1030 REM =====  Draw up menu box.....
1031 REM
1032 COLOR 7,1: CLS
1033 REM
1034 REM =====  Draw up menu box.....
1035 REM
1036 LOCATE 6,26: PRINT "ALLOCATION MENU"
1037 LOCATE 8,26: PRINT "Calibrated Sensors" CHR$(27);" "
1038 LOCATE 10,26: PRINT "Deployed Sensors"
1039 LOCATE 12,26: PRINT "Sensors Requiring Calibration"
1040 LOCATE 14,26: PRINT "New Sensor Entries"
1041 LOCATE 16,26: PRINT "Dump to printer sensor list"
1042 LOCATE 18,26: PRINT "Return to Main Menu"
1043 LOCATE 21,8: PRINT "Move arrow to required sensor type and press <RETURN> when finished"
1044 LOCATE 22,26: PRINT "Move around using cursor keys"
1045 COLOR 3,1
1046 LOCATE 3,30: PRINT "MULTIMET SENSOR MANAGEMENT"
1047 REM
1048 REM =====  Define the cursor keys.....
1049 REM
1050 REM 80: UP$=CHR$(80)
1051 REM 72: DN$=CHR$(72)
1052 REM 10: XY$=CHR$(10)
1053 REM 11: M$=CHR$(11)
1054 REM 18: B$=CHR$(18)
1055 REM 35: A$=CHR$(35)
1056 REM
1057 REM =====  Set up pointer box in menu.....
1058 REM
1059 COLOR 7,1
1195 Y1 = 97
1198 GOSUB 20000
1200 LOCATE X,Y
1202 REM
1205 REM ===== Reading input from keyboard....
1207 REM
1210 WHILE KYS<>CHR$(13)
1220 KYS=INKEY$
1270 IF (KYS=DN$) AND ((X>6)AND(X<16)) THEN LOCATE X,Y-3:PRINT" ";X=X+2:LOCATE X,Y-2:PRINT CHR$(27); : GOSUB 20000
1280 IF (KYS=UP$) AND ((X>=8)AND(X<=18)) THEN LOCATE X,Y-3:PRINT" ";X=X-2:LOCATE X,Y-2:PRINT CHR$(27); : GOSUB 20000
1290 WEND
1291 IF X=8 THEN LT$="R"
1292 IF X=10 THEN LT$="D"
1293 IF X=12 THEN LT$="M"
1294 IF X=14 THEN RUN"SEN3"
1295 IF X=16 THEN RUN"senprt"
1296 IF X=18 THEN RUN"MAINMENU"
1298 GOTO 1500
1300 END
1500 REM -SETUP SELECT SENSOR TYPE MENU
1550 COLOR 7,1:CLS
1561 REM ===== Draw up menu box.....
1567 LINE (5,5)-(635,345),,8
1569 LINE (180,55)-(480,266),2,BF
1570 LINE (185,60)-(475,261),64,B
1575 LOCATE 6,26:PRINT" Air Temperature" "CHR$(27); "
1570 LOCATE 8,26:PRINT" Sea Temperature"
1572 LOCATE 10,26:PRINT" Wind Speed"
1573 LOCATE 12,26:PRINT" Wind Direction"
1574 LOCATE 14,26:PRINT" Radiometers"
1574 LOCATE 16,26:PRINT" Others"
1575 LOCATE 18,26:PRINT" Return to Allocation Menu"
1576 LOCATE 21,8:PRINT" Move arrow to required sensor type and press <RETURN> when finished"
1576 LOCATE 22,26 : PRINT"Move around using cursor keys"
1576 COLOR 3,1
1576 LOCATE 3,33:PRINT "SELECT SENSOR TYPE"
1577 REM ===== Define the cursor keys.....
1578 DNS=CHR$(0)+CHR$(80);UPS=CHR$(0)+CHR$(72)
1579 X=6;Y=59;KYS=""
1579 REM ===== Set up pointer box in menu.....
1579 COLOR 7,1
1579 Y1 = 69
1579 GOSUB 20000
1600 LOCATE X,Y
1605 REM ===== Reading input from keyboard....
1610 WHILE KYS<>CHR$(13)
1620 KYS=INKEY$
1670 IF (KYS=DNS) AND ((X>6)AND(X<16)) THEN LOCATE X,Y-3:PRINT" ";X=X+2:LOCATE X,Y-2:PRINT CHR$(27); : GOSUB 20000
1680 IF (KYS=UPS) AND ((X>=8)AND(X<=18)) THEN LOCATE X,Y-3:PRINT" ";X=X-2:LOCATE X,Y-2:PRINT CHR$(27); : GOSUB 20000
1685 WEND
1890 IF X=6 THEN CODE%=100:GOSUB 12000
1891 IF X=8 THEN CODE%=200:GOSUB 12000
1892 IF X=10 THEN CODE%=300:GOSUB 12000
1893 IF X=12 THEN CODE%=400:GOSUB 12000
1894 IF X=14 THEN CODE%=500:GOSUB 12000
1895 IF X=16 THEN CODE%=1:GOSUB 12000
1896 IF X=18 THEN GOTO 1059
1897 GOTO 1059
1200 LOCATE 16,26:PRINT"Enter New Sensor"
1201 REM
1202 REM Subroutine for menu to select two calibration files-----
1203 REM-----------------------------------------------
1204 TOTAL=0
1205 OPEN "R",#,"SENSORS.DAT",9
1206 FIELD #1,2 AS TS,6 AS SS,1 AS LS
1207 IF CODE%=1 THEN CT=98 ELSE CT=99
1208 FOR MULTI%=CODE% TO CODE%+CT
1210 GET #1,MULTI%
1212 IF LS<>LS THEN GOTO 1218
1214 TOTAL=TOTAL+1
1216 CALFILE$(TOTAL)=TS+SS+"."+STR$(MULTI%)
1218 NEXT MULTI%
1219 CLOSE#
1220 REM Draw menu.....
1221 REM
1222 COLOR 3,1
1223CLS
1224 LINE (1,10)-(270,30),15,B
1225 IF LS="R" THEN LOCATE 2,2:PRINT"SENSORS AVAILABLE FOR DEPLOYMENT"
1226 IF LS="D" THEN LOCATE 2,4:PRINT"SENSORS CURRENTLY DEPLOYED"
1227 IF LS="M" THEN LOCATE 2,2:PRINT"SENSORS CURRENTLY IN MAINTANANCE"
1228 COLOR 15,1
1229 LINE (20,40)-(250,340),15,B
1230 PAINT (100,200),2,15
1231 N = 3
1232 ROM = 3
1233 IF TOTAL < 10 THEN TOP = 1 ELSE TOP = TOTAL - 9
1234 FOR N = TOP TO TOTAL
1235 ROW = ROM + 2
1236 LOCATE ROW,10:PRINT"";CALFILE$(N)"
1237 NEXT N
1238 LINE (300,50)-(500,75),15,B
1239 PAINT (400,60),4,15
1240 REM
1241 REM Set pointers for cal files 1 and 2.....
1242 REM
1243 CI = TOTAL
1244 POINTER = TOTAL
1245 IF TOTAL > 10 THEN ROW = 23 ELSE ROW = POINTER * 2 + 3
1246 LOCATE 5,45:PRINT"";CALFILE$(CI)"
1247 COLOR 12,1:LOCATE 5,42:PRINT"1:"
12322 REM LINE (30,10)-(220,30),15,B
12330 REM
12332 REM IF LT$="R" THEN LOCATE 2,7:PRINT"SENSOR NOW DEPLOYED"
12334 REM IF LT$="D" THEN LOCATE 2,3:PRINT"SENSOR RETURNED TO MAINTANANCE"
12336 REM IF LT$="M" THEN LOCATE 2,7:PRINT"SENSOR NOW AVAILABLE"
12340 KY$ = ""
12350 IROW1 = 20
12360 WHILE KY$ <> "/" AND KY$<>CHR$(27)
12370 KY$ = INKEY$
12380 IF KY$ = "1" THEN GOSUB 12470: CI = POINTER
12400 WEND:C1=POINTER
12405 IF ASC(KY$)<>27 THEN GOTO 12440
12410 KY$ = ""
12420 OPEN "R",#1,"SENSORS.DAT",9
12424 FIELD #1,2 AS T$,6 AS S$,1 AS L$
12426 LSET T$=LEFT$(CALFILE$(C1),2)
12428 LSET S$=MID$(CALFILE$(C1),3,6)
12430 IF LT$="R" THEN LSET L$="D"
12432 IF LT$="D" THEN LSET L$="M"
12434 IF LT$="M" THEN LSET L$="R"
12436 CODE%=VAL(RIGHT$(CALFILE$(CI),3))
12438 PUT#1,CODE%
12440 CLOSE#1
12450 IF KY$>< CHR$(27) THEN GOSUB 65000
12460 RETURN
12470 REM
12480 REM -----------------------------------------------
12490 REM ---- Subroutine to allow user to choose a calibration file ----
12500 REM -----------------------------------------------
12510 REM
12520 REM
12530 PRINT (400,10),1,15:GOSUB 30000
12540 COLOR 10,1
12550 LOCATE 8,35: PRINT "Use UP DOWN keys to move cursor"
12560 COLOR 12,1
12570 LOCATE 10,35: PRINT "Press RETURN to make selection"
12580 REM ---- Define up down keys.....
12590 REM
12600 DN$ = CHR$(0) + CHR$(80)
12610 UP$  = CHR$(0) + CHR$(72)
12620 REM
12630 REM ---- Scan keyboard for Return and Up Down keys.....
12640 REM
12650 KY$ = ""
12660 COL = 12
12670 GOSUB 13100
12680 WHILE (KY$ <> CHR$(13) AND KY$ <> CHR$(27))
12690     KY$ = INKEY$
12700     IF (KY$ = UP$ AND POINTER > 1) THEN IF (ROW = 5) THEN GOSUB 12840 :COL = 12:GOSUB 13100
12710 ELSE COL = 15:GOSUB 13100: POINTER = POINTER - 1: ROW = ROW - 2: COL = 12:GOSUB 13100
12710     IF (KY$ = DN$ AND POINTER < TOTAL) THEN IF (ROW = 23) THEN GOSUB 12970: COL = 12: GOSUB
12710 ELSE COL = 15: GOSUB 13100: POINTER = POINTER + 1: ROW = ROW + 2: COL = 12: GOSUB 13100
12720 WEND
12725 IF KY$= CHR$(27) THEN GOTO 12839
LOCATE 1,35: PRINT "Enter / to select the required sensor"
COLOR 12,1
LOCATE 2,35: PRINT "Press 1 to change the sensor"
LOCATE 3,35; PRINT "to one of those shown on the left."
MULTIMET$=RIGHT$(CALFILE$(POINTER),3)
FOR I = 1 TO 3: X$=MID$(MULTIMET$,I,1):IF X$=" " THEN MULTIMET$=RIGHT$(MULTIMET$,3-I)
NEXT I
IF LEN(MULTIMET$)=1 THEN MULTIMET$="00"+MULTIMET$
IF LEN(MULTIMET$)=2 THEN MULTIMET$="0"+MULTIMET$
FF$=DRIVE$+"\ametdb\metdb_da\history."+MULTIMET$
OPEN FF$ FOR INPUT AS #3:XF=0
LINE (260,90)-(630,330),15,B
FOR I = 0 TO 5: INPUT#3,AF$:NEXT:COLOR 10,1
WHILE EOF(3)>0
INPUT #3,AF$
FOR I=1 TO LEN(AF$):TT$=MID$(AF$,I,1):IF TT$=CHR$(9) THEN TT$=" 
IF I MOD 45=0 THEN XF=XF+1
LOCATE 8+XF,34+I MOD 45:PRINT TT$;
XF=XF+1:IF XF>15 THEN XF=0: GOSUB 30000
WEND
CLOSE#3
REM
REM Subroutine to scroll screen up one place -----
LR = ROW
POINTER = POINTER - 1
COLOR 15,1
FOR LOOP = POINTER TO POINTER + 9
LOCATE LR,10: PRINT " ";CALFILE$(LOOP);" "
LR = LR + 2
NEXT LOOP
RETURN
REM
REM Subroutine to scroll down one place -----
LR = ROW
POINTER = POINTER + 1
COLOR 15,1
FOR LOOP = POINTER TO POINTER-9 STEP -1
LOCATE LR,10: PRINT " ";CALFILE$(LOOP);" "
LR = LR - 2
NEXT LOOP
RETURN
REM
13110 REM  Subroutine to write name of cal file in a specified colour ----
13120 REM  Write cal file in large menu box.....
13130 REM  Write cal file in small menu box.....
13140 REM
13150 REM
13160 REM
13170 REM
13180 REM
13190 LOCATE ROW,10: PRINT " ;CALFILE$(POINTER);" "
13200 REM
13210 REM
13220 REM
13230 LOCATE 5,45: PRINT " ;CALFILE$(POINTER);" "
13240 RETURN
14000 REM
14002 REM
14005 REM Subroutine to set up key trap for Ctrl Esc ----
14007 REM
14010 REM
14014 REM
14016 REM Define keys to include all possible combinations of CAPS LOCK
14017 REM and NUM LOCK.....
14018 REM
14020 KEY 15,CHR$(&H64)+CHR$(1)
14030 KEY 16,CHR$(&H44)+CHR$(1)
14040 KEY 17,CHR$(&H24)+CHR$(1)
14050 KEY 18,CHR$(&H)+CHR$(1)
14060 REM
14064 REM Direct all escape key combinations to line 65300.....
14066 REM
14070 ON KEY (15) GOSUB 65300
14080 ON KEY (16) GOSUB 65300
14090 ON KEY (17) GOSUB 65300
14100 ON KEY (18) GOSUB 65300
14110 REM
14115 REM Turn all escape key combinations on.....
14117 REM
14120 KEY (15) ON
14130 KEY (16) ON
14140 KEY (17) ON
14150 KEY (18) ON
14200 RETURN
20000 REM
20010 REM
20020 REM Subroutine for drawing & flashing pointer box in menu ----
20030 REM
20040 REM
20050 REM First overwrite old box.....
20060 REM
20070 LINE (199,Y1)-(464,Y1+15),2,B
20100 REM Now draw box and flash at new position.....
20110 REM
20120 IF (KY$ = DNS) THEN Y1 = Y1 + 28 ELSE IF (KY$ = UPS) THEN Y1 = Y1 - 28
20140 FOR COL = 9 TO 15
40400 REM
40405 REM ===== Disk media error.....
40410 REM
40420 LINE (50,196)-(330,250),1,BF
40430 COLOR 12,1: LOCATE 15,11: PRINT " Error with hardware or disk "
40440 COLOR 15,1: LOCATE 16,17: PRINT " Program halted "
40490 RETURN
40500 REM
40502 REM ===== Illegal function call.....
40505 REM
40510 COLOR 12,8: LOCATE 17,33: PRINT "Insignificant data"
40520 LOCATE 18,26: PRINT "Could not calculate coefficients"
40530 COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
40540 WHILE (INKEY$ = ""
40550 WEND
40555 CLS
40560 REM
40562 REM ===== Exit program and return to operating system.....
40565 REM
40569 RETURN
40600 REM
40602 REM ===== Path not found.....
40605 REM
40610 CLS
40620 COLOR 12,1:LOCATE 17,24: PRINT "Could not find database on drive ";DRIVE$
40630 COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
40640 WHILE (INKEY$ = ""
40650 WEND
40655 CLS
40660 REM
40662 REM ===== Exit program and return to operating system.....
40665 REM
40669 RETURN
50000 REM
50100 REM DRAW GREEN BOX
50105 REM CLS
50110 COLOR 15,1
50120 FOR N=1 TO 150 STEP 2
50130 LINE (BB%-N,160-N)-(BB%+N,160+N),2,BF
50140 NEXT N
50150 RETURN
65000 REM
65002 REM -------------------------------------------
65005 REM ===== Subroutine to update database ----
65006 REM -------------------------------------------
65010 REM
65015 REM ===== If Psychrometers has been chosen then change ID back to TT.....
65020 REM
65030 IF (IDS = "TW" OR ID$ = "TD") THEN ID$ = "TT"
65045 REM
65046 REM ===== Write name of new calibration file to ametdb directory.....
65047 REM
65050 REM FILEIS = DRIVES + "\AMETDB\" + IDS + SENSNUM$ + ".L"
65055 REM LOCATE 16,5: PRINT "Updating sensor file : ";FILEIS

REM OPEN FILE1$ FOR APPEND AS #1
REM PRINT #1,RIGHT$(FILE$,22)
REM CLOSE #1
REM
REM ===== Write to history file date of calibration....
REM
GOSUB 30000:MS=""
LOCATE 8,35:PRINT "ENTER COMMENTS OR CR"
LOCATE 10,35:INPUT;M$
FILE2$ = DRIVE$+"AMETDB\METDB_DA\HISTORY." + MULTIMET$
LOCATE 12,35: PRINT "Updating history file : 
LOCATE 14,35: PRINT FILE2$
OPEN FILE2$ FOR APPEND AS #2
DAYS = MID$(DATE$,4,2)
MONTH$ = LEFT$(DATE$,2)
YEAR$ = RIGHT$(DATE$,4)
IF LTS="R" THEN TEXT$="SENSOR DEPLOYED"
IF LTS="D" THEN TEXT$="SENSOR RETURNED TO MAINTANANCE"
IF LTS="M" THEN TEXT$="SENSOR NOW AVAILABLE"
PRINT #2, DAYS + "/" + MONTH$ + "/" + YEAR$+ "" +TEXT$
PRINT #2,M$
CLOSE #2
LOCATE 16,35: PRINT "---Press Any Key for Menu---"
WHILE (INKEY$ = "")
WEND
RETURN
REM
REM ----- Subroutine for execution when Ctrl Esc has been pressed -----
REM
CLS
SCREEN 0,0,0
LOCATE 14,9: PRINT "Program Execution Halted.... Returning to MAINMENU"
LOCATE 16,29: PRINT "---Press Any Key---"
WHILE (INKEY$ = "")
WEND
RUN "MAINMENU"
1000 REM **********************************************************************
1010 REM ******************  SEN3.BAS  by RWP 8/11/90  ***************************
1020 REM **********************************************************************
1040 REM
1041 REM ===== Set error trap.....
1042 REM
1043 ON ERROR GOTO 40000
1044 REM
1045 REM ===== Call subroutine to set up Ctrl Esc key trap.....
1046 REM
1047 GOSUB 14000
1048 KEY OFF
1049 SCREEN 9
1056 REM
1057 REM ===== Set up user menu....
1058 REM
1059 COLOR 7,1: CLS
1060 REM
1061 REM ===== Draw up menu box.....
1065 REM
1067 LINE (5,5)-(635,345),,B
1090 LINE (180,53)-(480,294),2,BF
1100 LINE (185,58)-(475,289),64,B
1110 LOCATE 6,26:PRINT" Air Temperature "
1120 LOCATE 8,26:PRINT" Sea Temperature "
1130 LOCATE 10,26:PRINT" Wind Speed "
1140 LOCATE 12,26:PRINT" Wind Direction "
1142 LOCATE 14,26:PRINT" Radiometers "
1143 LOCATE 16,26:PRINT" Others "
1144 LOCATE 18,26:PRINT" Change multimeter allocation "
1145 LOCATE 20,26:PRINT" Return to Allocation Menu "
1150 LOCATE 22,8:PRINT" Move arrow to required sensor type and press <RETURN> when finished"
1160 LOCATE 23,26:PRINT"Move around using cursor keys"
1162 COLOR 3:LOCATE 3,33:PRINT "SELECT SENSOR TYPE"
1165 REM
1170 REM ===== Define the cursor keys.....
1175 REM
1180 DNS=CHR$(0)+CHR$(80):UP$=CHR$(0)+CHR$(72)
1185 X=6;Y=59:KYS=""
1190 REM
1191 REM ===== Set up pointer box in menu.....
1192 REM
1193 COLOR 7,1
1195 Y1 = 69
1198 GOSUB 20000
1200 LOCATE X,Y
1202 REM
1205 REM ===== Reading input from keyboard....
1207 REM
1210 WHILE KYS<>CHR$(13)
1220 KYS=INKEY$
1270 IF (KYS=DNS) AND ((X>=6)AND(X<=18)) THEN LOCATE X,Y-3:PRINT" ";X=X+2:LOCATE X,Y-2:PRINT CHR$(27); : GOSUB 20000
IF (KY$=UP$) AND ((X>-8)AND(X<-20)) THEN LOCATE X,Y-3:PRINT" ;:X=X-2;LOCATE X,Y-2:PRINT CHR$(27);:GOSUB 20000
1290 WEND
1291 IF X=6 THEN METNO=100;GOSUB 30000
1292 IF X=8 THEN METNO=200;GOSUB 30000
1293 IF X=10 THEN METNO=300;GOSUB 30000
1294 IF X=12 THEN METNO=400;GOSUB 30000
1295 IF X=14 THEN METNO=500;GOSUB 30000
1296 IF X=16 THEN METNO=1;GOSUB 30000
1297 IF X=18 THEN METNO=9999;GOSUB 30000
1298 IF X=20 THEN RUN "SEND"
1299 GOTO 1059
1300 END
13000 REM
13002 REM -------------------------------
13005 REM ------ Subroutine to set up key trap for Ctrl Esc ------
13007 REM -------------------------------
13010 REM
13014 REM
13016 REM ------ Define keys to include all possible combinations of CAPS LOCK
13017 REM and NUM LOCK.....
13018 REM
13020 KEY 15,CHR$(&H64)+CHR$(1)
13030 KEY 16,CHR$(&H4)+CHR$(1)
13040 KEY 17,CHR$(&H24)+CHR$(1)
13050 KEY 18,CHR$(&H4)+CHR$(1)
13060 REM
13064 REM ------ Direct all escape key combinations to line 65300 ..... 
13066 REM
13070 ON KEY (15) GOSUB 65300
13080 ON KEY (16) GOSUB 65300
13090 ON KEY (17) GOSUB 65300
13100 ON KEY (18) GOSUB 65300
13110 REM
13115 REM ------ Turn all escape key combinations on.....
13117 REM
13120 KEY (15) ON
13130 KEY (16) ON
13140 KEY (17) ON
13150 KEY (18) ON
13160 RETURN
20000 REM
20010 REM -------------------------------
20020 REM ------ Subroutine for drawing & flashing pointer box in menu ------
20030 REM -------------------------------
20040 REM
20050 REM ------ First overwrite old box.....
20060 REM
20070 LINE (199,Y1)-(464,Y1+15),2,B
20100 REM ------ Now draw box and flash at new position.....
20110 REM
20120 IF (KY$=DN$) THEN Y1 = Y1 + 28 ELSE IF (KY$ = UP$) THEN Y1 = Y1 - 28
20140 FOR COL = 9 TO 15
20150 LINE (199,Y1)-(464,Y1+15),COL,B
20160 NEXT COL
20170 LINE (199,Y1)-(464,Y1+15),12,B
20500 RETURN
30000 SHELL"setkbd capslock=on":GOSUB 30150
30001 IF METNO=1 THEN GOSUB 30300
30002 IF METNO=100 THEN GOSUB 30400
30003 IF METNO=200 THEN GOSUB 30500
30004 IF METNO=300 THEN GOSUB 30600
30005 IF METNO=400 THEN GOSUB 30700
30006 IF METNO=500 THEN GOSUB 30800
30007 IF METNO=9999 THEN GOSUB 30900:GOTO 30045
30100 OPEN "SENSORS.DAT",9
30110 FIELD #1,2 AS T$,6 AS S$,L$ AS L$
30120 GET#1,METNO:IF L$<"R" AND L$<"M" AND L$<"D" GOTO 30030
30122 WHILE L$="R" OR L$="M" OR L$="D"
30126 METNO=METNO+1
30127 GETH#1,METNO
30128 WEND
30129 LOCATE 2,14:PRINT "Incorrect entry PRESS <ESC> and start again"
30130 LOCATE 5,40:PRINT "MultiMet NO.":METNO
30135 HIST$="Y"
30140 IF CODE%>999 GOTO 30130
30145 MMS="":MANU$="":SENS$="":LC$=""
30150 LOCATE 7,40:PRINT"SENSOR TYPE CODE ?";
30152 WHILE MMS<>CHR$(13)
30154 MMS=INPUT$(1)
30155 IF MMS=CHR$(13) THEN GOTO 30058
30156 IF MMS=CHR$(27) THEN HIST$="N":GOTO 30130
30157 PRINT MMS;:MANU$=MANU$+MMS
30158 WEND:MMS=""
30159 LOCATE 12,6:PRINT"Typical Serial no.VI1234"
30160 LOCATE 9,40:PRINT"SENSOR SERIAL NO. ?";
30162 WHILE MMS<>CHR$(13)
30164 MMS=INPUT$(1)
30165 IF MMS=CHR$(13) THEN GOTO 30068
30166 IF MMS=CHR$(27) THEN HIST$="N":GOTO 30130
30167 PRINT MMS;:SENS$=SENS$+MMS
30168 WEND
30170 LOCATE 11,40:PRINT "LOCATION CODE R/M/D ?";
30172 LCS=INPUT$(1):PRINT LCS
30173 IF LCS=CHR$(27) THEN HIST$="N":GOTO 30130
30175 IF LCS<="R" AND LCS<="M" AND LCS<="D" THEN LOCATE 11,62:PRINT"":GOTO 30070
30180 LSET T$=MANU
30189 LSET S$=SENS
30190 LSET L$=LCS
30195 PUT #1,METNO
30199 CLOS#1
30205 IF HIST$="Y" THEN GOSUB 60000
30140 GOTO 1059
30150 CLS
30160 LINE (30,10)-(600,340),7,BF
30170 LINE (35,15)-(595,335),8,B
30180 COLOR 15,1
30190 LINE (290,50)-(590,320),8,BF
LOCATE 5,6:PRINT " Other Sensors"
LOCATE 7,6:PRINT " AC - Accelerometers"
LOCATE 8,6:PRINT " AP - Air Pressure"
LOCATE 9,6:PRINT " HU - Relative Humidity"
LOCATE 10,6:PRINT " PD - Precipitation gauge"
RETURN
LOCATE 5,6:PRINT " Air Temperature Sensors"
LOCATE 7,6:PRINT " TT - Psychrometers"
LOCATE 8,6:PRINT " TD - Dry Bulb Temperatures"
LOCATE 9,6:PRINT " TW - Wet Bulb Temperatures"
RETURN
LOCATE 5,6:PRINT " Sea Temperature Sensors"
LOCATE 7,6:PRINT " SP - SOAP"
LOCATE 8,6:PRINT " SF - FISH"
RETURN
LOCATE 5,6:PRINT " Wind Speed Sensor"
LOCATE 7,6:PRINT " WS - Cup Anemometer"
LOCATE 8,6:PRINT " WW - Propellar Vane"
RETURN
LOCATE 5,6:PRINT " Wind Direction Sensors"
LOCATE 7,6:PRINT " WD - Wind Vane"
RETURN
LOCATE 5,40:INPUT "Enter MultiMet NO.";METNO
OPEN "R",#1,"SENSORS.DAT",9
FIELD #1,2 AS T$,6 AS S$,1 AS L$
GET#1,METNO
HIST$="N"
RETURN
REM ENTER MUTIMET NUMBER
REM
REM Error checking subroutines
REM
REM
REM
REM
40022 REM ===== Check for errors during file opening.....
40025 IF ERR = 53 THEN GOSUB 40100: RESUME 1040
40027 IF ERR = 52 THEN GOSUB 40200: RESUME 1299
40029 IF ERR = 71 THEN GOSUB 40300: RESUME 1520
40031 IF ERR = 72 THEN GOSUB 40400
40033 IF (ERR = 5 OR ERR = 11) THEN GOSUB 40500: RESUME 65300
40035 IF ERR = 76 THEN GOSUB 40600: RESUME 65300
40037 STOP
40040 REM
40100 REM ——— File not found.....
40110 LINE (50,196)-(330,250),3,BF
40115 COLOR 12,1: LOCATE 15,11: PRINT "Sensor not found in database"
40120 COLOR 15,1: LOCATE 16,17: PRINT "Please try again"
40125 BEEP
40130 RETURN
40140 REM
40210 REM ——— Invalid filename.....
40215 LINE (50,196)-(330,250),3,BF
40220 COLOR 12,1: LOCATE 15,14: PRINT "Invalid filename used"
40225 COLOR 15,1: LOCATE 16,17: PRINT "Please try again"
40230 BEEP
40235 RETURN
40240 REM
40310 REM ——— Drive not ready.....
40315 LINE (50,196)-(330,250),3,BF
40320 COLOR 12,1: LOCATE 15,15: PRINT "Drive \;DRIVE;\ not ready"
40325 COLOR 15,1: LOCATE 16,13: PRINT "Press return when ready"
40330 BEEP
40335 ANSWERS = ""
40340 WHILE (ANSWERS <> CHR$(13))
40345 ANSWERS = INKEY$
40350 WEND
40355 RETURN
40360 REM
40410 REM ——— Disk media error.....
40415 LINE (50,196)-(330,250),3,BF
40420 COLOR 12,1: LOCATE 15,11: PRINT "Error with hardware or disk"
40425 COLOR 15,1: LOCATE 16,17: PRINT "Program halted"
40430 RETURN
40440 REM
40500 REM ——— Illegal function call.....
40505 REM
40510 COLOR 12,8: LOCATE 17,33: PRINT "Insignificant data"
40520 LOCATE 18,26: PRINT "Could not calculate coefficients"
40530 COLOR 15,8: LOCATE 20,32: PRINT "——Press Any Key——"
40540 WHILE (INKEY$ = "")
40545 WEND
40550 CLS
40560 REM
40562 REM ===== Exit program and return to operating system.....
40565 REM
40590 RETURN
40600 REM
40602 REM ===== Path not found.....
40605 REM
40610 RETURN

40620 COLOR 12,1:LOCATE 17,24: PRINT "Could not find database on drive ":DRIVES$ 40630 COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
40640 WHILE (INKEY$ = "")
40650 WEND
40655 CLS
40660 REM
40662 REM ===== Exit program and return to operating system.....
40665 REM
40690 RETURN

50000 REM
50100 REM DRAW GREEN BOX
50105 REM CLS
50110 COLOR 15,1
50120 FOR N=1 TO 150 STEP 2
50130 LINE (BB%-N,160-N)-(BB%+N,160+N),2,BF
50140 NEXT N
50200 RETURN

60000 REM -----------------------------------------------
60010 REM ------ GENERATE LIST,HISTORY AND DETAILS FILES ------
60020 REM ------ FOR NEW SENSORS ------
60030 REM -----------------------------------------------
60040 METNO$=RIGHT$(STR$(METNO),LEN(STR$(METNO))-1)
60050 IF LEN(METNO$)=1 THEN METNO$="00"+METNO$
60060 IF LEN(METNO$)=2 THEN METNO$="0"+METNO$
60080 HIST$="COPY D:\AMETDB\METDB_DA\HISTORY D:\AMETDB\METDB_DA\HISTORY."+METNO$
60085 DETL$="COPY D:\AMETDB\METDB_DA\DETAILS D:\AMETDB\METDB_DA\DETAILS."+METNO$
60090 SHELL HIST$
60091 LOCATE 13,7:PRINT "HISTORY FILE GENERATED 
60095 SHELL DETL$
60096 LOCATE 15,7:PRINT "DETAILS FILE GENERATED 
60100 OPEN "D:\AMETDB\"+TS+SENS$+.L" FOR OUTPUT AS #1
60110 PRINT#1, ":METDB_DA\DETAILS."+METNO$
60120 PRINT#1, ":METDB_DA\HISTORY."+METNO$
60130 CLOSE#1
60136 LOCATE 17,7:PRINT"LIST FILE GENERATED"
60138 LOCATE 19,7:PRINT "PRESS ANY KEY TO CONTINUE"
60150 KY$=INKEY$
60155 IF KY$="" THEN GOTO 60150
60200 RETURN

65300 REM
65302 REM -----------------------------------------------
65305 REM ------ Subroutine for execution when Ctrl Esc has been pressed ------
65307 REM -----------------------------------------------
65310 REM
65320 CLS
65330 SCREEN 0,0,0
65340 LOCATE 14,9: PRINT "Program Execution Halted..... Returning to MAINMENU"
LOCATE 16,29: PRINT "---Press Any Key---"
WHILE (INKEY = "")
WEND
RUN "MAINMENU"
REM ***************************************************************
REM ************************  WINDCAL.BAS  by RAF  17/5/90  ********************
REM *************************************************************************
REM ——  Version  1.0  17/05/90
REM =====  This  program  was written  in G W BASIC by Refayet Ahmed.
REM =====  The program provides an easy and efficient way to obtain
REM =====  coefficients required to calibrate wind sensors.
REM =====  It also creates a data file which is written into the database.
REM =====  Details of the wind sensor data file are:
REM ——  Sensor  ID
REM =====  Current Julian Day of calibration
REM =====  Operator's name
REM =====  Standard calibrated against
REM =====  Coefficient terms and values
REM =====  Table of the data produced
REM ——=  Version  1.1  04/06/90
REM =====  New version allows drive to be specified. Stored as DRIVE$.
REM =====  Also date can be specified. All these specifications are asked for
REM =====  at the beginning of the program. The execution of this part of the
REM =====  program is in the subroutine between lines 15000 - 15450.
REM ——=  Version  1.2  14/06/90
REM =====  This version asks for all the specifications after the sensor
REM =====  has been chosen from the main menu.
REM =====  An addition to the main menu has been made. The user may now
REM =====  request a calibration certificate. This is accommodated in a
REM =====  subroutine between lines 12000 - 12490. This subroutine calls upon
REM =====  another GW Basic program called #######.BAS which makes a
REM =====  comparison between the two most recent calibration files and
REM =====  produces a printout. When the program #######.BAS has finished
REM =====  it returns the user back to the calling program (either
REM =====  WINDCAL.BAS or BATHCON.BAS).
REM ——=  Version  1.3  24/7/90
REM MAIN  MENU  NOW  ALLOWS  THE  USER  TO RETURN  TO  THE CALIBRATION MENU WHICH CALLED THIS PROGRAM ORIGINALLY
1000 REM ------------------------
1010 REM ------ Main program ------
1015 REM ------------------------
1017 REM
1020 REM ===== Declare all arrays.....
1021 REM
1024 DIM FREQ$(20) , WINDS$(20)
1026 DIM X(40), Y(40), LL(40)
1028 DIM P$(40), Q$(40)
1030 DIM A$(40), B$(40), S$(40), C$(40)
1040 REM
1041 REM ===== Set error trap.....
1042 REM
1043 ON ERROR GOTO 40000
1044 REM
1045 REM ===== Call subroutine to set up Ctrl Esc key trap.....
1046 REM
1047 GOSUB 14000
1048 KEY OFF
1049 SCREEN 9
1056 REM
1057 REM ===== Set up user menu....
1058 REM
1059 COLOR 7,1: CLS
1060 REM
1061 REM ===== Draw up menu box.....
1065 REM
1067 LINE (5,5)-(635,345),,B
1070 LOCATE 3,25:PRINT "Wind Sensor Calibration Program"
1090 LINE (180,83)-(480,266),2,BF
1100 LINE (185,88)-(475,261),64,B
1110 LOCATE 8,26:PRINT"Cup Anemometer" ;CHR$(27);""
1112 LOCATE 10,26:PRINT"Propeller Vane Instrument" "
1113 LOCATE 12,26:PRINT"Wind Direction" "
1114 LOCATE 14,26:PRINT"Others" "
1115 LOCATE 16,26:PRINT"Calibration Certificate" "
1116 LOCATE 18,26:PRINT"Return to Calibration Menu" "
1117 LOCATE 21,8:PRINT"Move arrow to required sensor type and press <RETURN> when finished"
1160 LOCATE 22,26 : PRINT"Move around using cursor keys"
1165 REM
1170 REM ===== Define the cursor keys.....
1175 REM
1180 DNS=CHR$(0)+CHR$(80);UPS=CHR$(0)+CHR$(72)
1185 X=8;Y=59;KY=""
1190 REM
1191 REM ===== Set up pointer box in menu.....
1192 REM
1193 COLOR 7,1
1195 Y1 = 97
1198 GOSUB 20000
1200 LOCATE X,Y
1202 REM
1205 REM ===== Reading input from keyboard....
1207 REM
1210 WHILE KY$<>CHR$(13)
1220   KY$=INKEY$
1230   IF (KY$=DN$) AND ((X>=8)AND(X<=16)) THEN LOCATE X,Y-3:PRINT"":";X=X+2:LOCATE X,Y-2:PRINT CHR$(27); GOSUB 20000
1240   IF (KY$=UP$) AND ((X>=10)AND(X<=18)) THEN LOCATE X,Y-3:PRINT"":";X=X-2:LOCATE X,Y-2:PRINT CHR$(27); GOSUB 20000
1250 WEND
1290 REM
1295 REM
1300 REM ===== Determine which sensor has been chosen from the menu and go to subroutine which assigns sensor name & ID code ..... 
1301 REM
1305 REM ===== Line 10000 Subroutine for Cup anemometer 
1306 REM Line 10500 Subroutine for Propeller vane instrument
1307 REM Line 11000 Subroutine for Vane
1308 REM Line 11500 Subroutine for Other sensors
1309 REM Line 65300 Subroutine to exit program
1310 REM
1320 IF X = 8 THEN GOSDB 10000 ELSE IF X=10 THEN GOSUB 10500 ELSE IF X = 12 THEN GOSUB 11000 ELSE I X = 14 THEN GOSUB 11500 ELSE IF X = 16 THEN GOSUB 12000 ELSE GOSUB 65300
1330 REM
1332 REM ===== Call subroutine to get user details..... 
1334 REM
1340 GOSDB 15000
1400 REM
1401 REM ===== Obtain sensor number from user..... 
1405 REM
1410 COLOR 12,1 : CLS
1420 LOCATE 4,51 : PRINT "Enter sensor number";
1425 COLOR 7,1
1430 LOCATE 6,53 : PRINT "e.g. VI1992";
1440 LOCATE 8,51 : PRINT "Press ";
1442 COLOR 12,1
1444 PRINT "<RETURN>";
1446 COLOR 7,1
1448 PRINT " when finished"
1450 LINE (10,5)-(380,345),3,BF
1460 LINE (15,10)-(375,340),64,B
1470 COLOR 2,1 : LOCATE 10,((51-LEN (SENS0R$))/2) : PRINT SENSOR$;
1480 COLOR 7,1 : LOCATE 13,15 : PRINT " Sensor ID : ";
1490 LOCATE 13,28 : INPUT SENSNUM$ 
1500 REM
1501 REM ===== Open file in database to obtain multimet number..... 
1502 REM
1510 FILE1$ = DRIVE$ + "AMETDB\" + ID$ + SENSNUM$ + ".L"
1520 OPEN FILE1$ FOR INPUT AS #1
1540 IF EOF(1) THEN CLS : COLOR 4,14:LOCATE 16,16 : PRINT "EMPTY FILE IN DATA BASE!!! PROGRAM CLOSED" : STOP
1542 WHILE (NOT EOF(1))
1545   INPUT #1,.BLANKS
1546   IF BLANKS <> "" THEN RECORDS = BLANKS
1547 WEND
1550 MULTIMETS = RIGHTS(RECORDS,3)
1560 REM
1561 REM ===== Obtain todays date from the internal clock and convert to Julian day.....
1563 REM
1570 DAY = VAL(DAYS$)
1580 MONTH = VAL(MONTHS$)
1590 JDAYS = STR$(VAL(MIDS$("0003105909012015118121243273304334",((MONTH-1)*3)+1,3))+DAY)
1610 JDAYS = RIGHTS(JDAYS$)
1620 REM
1630 REM ====== If there is no version code on the file then set one.....
1635 REM ====== If the Julian day on the file is the same as the present day
1636 REM then increment the version code.....
1640 REM
1650 VCODES = RIGHTS(RECORDS$)
1660 CURJDAYS = LEFT$(RIGHTS(RECORDS$,10),3)
1670 IF ASC(VCODES$) < 65 THEN ASVAL = 65 ELSE IF CURJDAYS=JDAYS THEN ASVAL = ASC(VCODES$)+1 ELSE ASVAL = 65
1680 VCODES = CHR$(ASVAL)
1700 REM
1705 REM ====== Obtain the details of the sensor from the data base and check
1706 REM with the user if the correct sensor has been chosen.....
1710 REM
1715 CLS
1820 FILE2$ = DRIVE$ + "\AMETDB\METDB_DA\DETAILS." + MULTIMET$.
1830 OPEN FILE2$ For INPUT AS #2
1835 N = 5
1840 WHILE NOT(EOF(2))
1850 INPDT #2,RECORDS$
1860 N = N + 1
1870 LOCATE N,18: PRINT RECORDS$
1880 WEND
1890 COLOR 12,1
1900 LOCATE 23,22: PRINT "Is this the required sensor? (y/n) :";
1905 ANSWERS = ""
1910 WHILE (ANSWERS <> "N" AND ANSWERS <> "Y")
1915 ANSWERS = INKEYS
1920 REM
1921 REM ====== If not required sensor then close the files and return to
1922 REM line 1400 where user inputs sensor number again.....
1925 REM
1930 IF (ANSWERS = "N" OR ANSWERS = "n") THEN CLOSE #1:CLOSE #2 :GOTO 1400
1940 REM
1942 REM ====== Accept upper or lower case "y"....
1945 REM
1950 IF ANSWERS = "y" THEN ANSWERS = "Y"
1955 WEND
1958 CLOSE #1,#2
1960 CLS
1965 REM
1966 REM ====== Build up filename to which to write.....
1967 REM
1970 FILE$ = DRIVES$ + "\AMETDB\METDB_DA\" + IDS$ + JDAYS$ + YEARS$ + VCODES$ + "." + MULTIMETS
2000 REM
2002 REM ====== Call subroutine for editor to enter data.....
2005 REM
2010 GOSUB 30000
2020 REM
2025 REM ===== Clear screen and print message.....
2030 REM
2040 CLS
2050 COLOR 12,8
2060 LOCATE 10,33: PRINT " PLEASE WAIT "
2065 COLOR 15,8
2070 LOCATE 12,27: PRINT " Calculating Coefficients "
3000 REM
3002 REM ===== Call subroutine to calculate coefficients.....
3005 REM
3010 GOSUB 60000
3100 REM
3105 REM ===== Call subroutine to update data base .....
3110 REM
3120 GOSUB 65000
3150 REM
3155 REM ===== Return to main menu again.....
3160 REM
3170 GOTO 1056
9999 END
10000 REM
10002 REM -------------------------------
10005 REM ===== Subroutine for Cup Anemometer =====
10010 REM -------------------------------
10020 ID$ = "WS"
10030 SENSORS = "Cup Anemometer"
10499 RETURN
10500 REM
10505 REM -------------------------------
10510 REM ===== Subroutine for Propeller Vane Instrument =====
10520 REM -------------------------------
10530 ID$ = "WW"
10540 SENSORS = "Propeller Vane Instrument"
10999 RETURN
11000 REM
11005 REM -------------------------------
11010 REM ===== Subroutine for Vane =====
11015 REM -------------------------------
11020 ID$ = "WD"
11030 SENSORS = "Vane"
11040 RETURN
11500 REM
11505 REM -------------------------------
11510 REM ===== Subroutine for other instruments =====
11515 REM -------------------------------
11517 CLS
11520 LOCATE 15,22
11530 INPUT "Enter sensor type code :-";ID$
11540 LOCATE 16,22
11550 INPUT "Enter sensor name :-";SENSORS
11560 RETURN
12000 REM
12002 REM -------------------------------
12005 REM ---- Subroutine for Calibration Certificate ----
12007 REM --------------------------------------------------
12010 REM
12015 COMMON PR0G$,FILE1$,DRIVE$,ID$
12020 PROG$  = "WINDCAL.BAS"
12025 IF FILE1$  = "" THEN CHAIN "CALCERT.BAS",1000
12030 CHAIN "CALCERT.BAS",1500
12031 RETURN
12032 REM
12033 REM Subroutine to set up key trap for Ctrl Esc ----
12034 REM
12035 REM ===== Define keys to include all possible combinations of CAPS LOCK and NUM LOCK,....
12036 REM
12037 KEY 15,CHR$(&H64)+CHR$(1)
12038 KEY 16,CHR$(&H44)+CHR$(1)
12039 KEY 17,CHR$(&H24)+CHR$(1)
12040 KEY 18,CHR$(&H4)+CHR$(1)
12041 REM
12042 REM ===== Direct all escape key combinations to line 65300 ....
12043 REM
12044 ON KEY (15)  GOSUB  65300
12045 ON KEY (16)  GOSUB  65300
12046 ON KEY (17)  GOSUB  65300
12047 ON KEY (18)  GOSUB  65300
12048 REM
12049 REM ===== Turn all escape key combinations on....
12050 REM
12051 KEY (15)  ON
12052 KEY (16)  ON
12053 KEY (17)  ON
12054 KEY (18)  ON
12055 REM
12056 REM ------ Subroutine to get user details ------
12057 REM
12058 CLS
12059 COLOR  15,8
12060 WHILE (USERDATES <> "/")
12061 REM
12062 REM  ===== Obtain current date from internal clock
12063 REM
12064 DAYS = MID$(DATE$,4,2)
12065 MONTH$ = LEFT$(DATE$,2)
12066 YEARS = RIGHT$(DATE$,2)
12067 WEND
12068 REM
12069 REM ------ Obtain date from user....
REM LOCATE 15,25: PRINT "Enter date dd/mm/yy "
LOCATE 16,25: PRINT "Or enter / for default of ";DAYS";/";MONTH$";/";YEAR$";
LOCATE 16,60: INPDTS OSERDATE$
IF (DSERDATE* - "/") THEN GOTO 15200
DAY$ = LEFT$(DSERDATE$,2)
MONTH$ = MID$(DSERDATE$,4,2)
YEAR$ = RIGHT$(0SERDATE$,2)
REM
REM  ===== Check if the date was entered in correct format.....
REM
REM  ===== Obtain username.....
CLS
DEFAULTS = OSERNAME$
IF DEFAULTS <> "" THEN LOCATE 16,25: PRINT "Enter / for ";DEFAULTS
LOCATE 15,25: INPUT "Please enter your username :";USERNAME$
IF OSERNAME$ <> "/" THEN USERNAME$ = DEFAULTS
CLS
REM  ===== Obtain drive on which database is held.....
CLS
WHILE (USERDRIVES <> "/")
LOCATE 15,20: PRINT "Enter name of drive on which database is held"
LOCATE 16,20: PRINT "Enter / for default drive D:
LOCATE 16,48: INPDTS USERDRIVES
IF USERDRIVES == "/" THEN DRIVES = "D": GOTO 15450
REM
REM ===== Check if drive has been specified correctly.....
REM
REM  ===== User enters standard to which sensor is calibrated against.....
CLS
LOCATE 15,10: PRINT "Enter the standard to which the sensor is calibrated against"
LOCATE 16,15: PRINT "Enter / for default standard of TESTOVENT"
LOCATE 16,57: INPUT STANDARDS
IF STANDARDS == THEN STANDARDS = "TESTOVENT"
CLS
REM  ===== User enters any comments.....
15555 REM
15560 CLS
15570 LOCATE 15,20: PRINT "Enter any comments (60 chars only)"
15580 LOCATE 16,20: PRINT "Enter / for blank"
15590 LOCATE 16,39: INPUT COMMENTS
15600 IF COMMENT$ = "/" THEN COMMENTS = ""
15610 RETURN
20000 REM
20010 REM ----- Subroutine for drawing & flashing pointer box in menu -----
30240 FREQ$(TOTAL) = ""
30250 TOTAL = TOTAL - 1
30500 REM
30502 REM ===== List data and show editor options.....
30505 REM
30510 CLS
30520 LINE (5,5)-(315,345),15,B
30530 LINE (321,0)-(635,345),15,B
30535 PAINT (400,200),2,30
30540 COLOR 4,1:LOCATE 2,3: PRINT "No Frequency Actual Wind Speed"
30545 COLOR 7,1
30550 FOR N = 1 TO TOTAL
30560 LOCATE N+3,2: PRINT N
30570 LOCATE N+3,B: PRINT "####.###": VAL (FREQ$(N))
30580 LOCATE N+3,25: PRINT USING "##*#.fff-f": VAL (MIND$(N))
30600 NEXT N
30610 FOR N = 1 TO 100
30620 LINE (450-N,150-N)-(450+N,150+N),8,BF
30650 NEXT N
30660 COLOR 15,1
30680 LOCATE 7,51: PRINT "Change 'C'"
30690 LOCATE 9,51: PRINT "Add 'A'"
30700 LOCATE 11,51: PRINT "Delete 'D'"
30710 LOCATE 13,51: PRINT "Re-enter 'R'"
30720 LOCATE 15,51: PRINT "Finish 'F'"
30730 LOCATE 20,45: PRINT "Press one of keys shown "
30740 LOCATE 21,45: PRINT "for required option "
30750 REM
30752 REM ===== Read users keystroke from and call appropriate subroutine.....
30755 REM
30760 REM ===== Line 32000 ===== Subroutine to change a data entry.
30762 REM Line 33000 ===== Subroutine to add to the data.
30764 REM Line 34000 ===== Subroutine to delete a data entry.
30766 REM Line 35000 ===== Subroutine to confirm re entry of all data.
30768 REM Line 30500 ===== Point where data is shown and editor options
30770 REM are listed.
30772 REM Line 30050 ===== Beginning of present subroutine where all the
30774 REM data is entered.
30776 REM Line 30120 ===== Point where data is entered.
30780 REM
30800 KY$ = ""
30810 WHILE (KY$<>"F" AND KY$<>"f")
30820  REM INKEY$ 30830  IF (KY$ = "C" OR KY$ = "c") THEN GOSUB 32000: GOTO 30500
30840  IF (KY$ = "A" OR KY$ = "a") THEN GOSUB 33000: GOTO 30120
30850  IF (KY$ = "D" OR KY$ = "d") THEN GOSUB 34000: GOTO 30500
30860  IF (KY$ = "R" OR KY$ = "r") THEN GOSUB 35000: IF ANSWERS = "Y" THEN GOTO 30050 ELSE GOTO
30870 WEND
31000 RETURN
32000 REM
32002 REM ----------------------------------------------
32005 REM ===== Subroutine to change data entry from editor menu -----
32007 REM ----------------------------------------------
32010 REM
32100 FOR N = 1 TO 50
32110 LINE (300,150-N)-(600,150+N),3,BF
32120 NEXT N
32150 COLOR 12,1: LOCATE 9,43: PRINT " Enter number ( 1 -";TOTAL;") which "
32160 LOCATE 10,47: PRINT " you wish to change: "
32170 COLOR 15,1: LOCATE 10,68: INPUT CHANGE$
32180 IF (VAL(CHANGE$) < 1 OR VAL(CHANGE$) > TOTAL) THEN BEEP: GOTO 32150
32190 CHANGE  = VAL (CHANGE$)
32195 LOCATE 12,47: PRINT " Enter the changes now "
32200 LOCATE CHANGE+3,2: PRINT CHANGE;
32210 COLOR 12,1: LOCATE 2,8: PRINT "Frequency"
32220 COLOR 15,1: LOCATE CHANGE+3,9: INPUT FREQ$(CHANGE)
32230 COLOR 4,1: LOCATE 2,8: PRINT "Frequency"
32240 COLOR 12,1:LOCATE 2,21: PRINT "Actual Wind Speed"
32250 COLOR 15,1:LOCATE CHANGE+3,24: INPUT WIND$(CHANGE)
32300 RETURN
33000 REM
33002 REM Subroutine to add to the data
33004 REM
33010 REM
33100 FOR N = 1 TO 50
33110 LINE (300,150-N)-(600,150+N),3,BF
33120 NEXT N
33130 COLOR 12,1: LOCATE 10,48: PRINT " Add to the data now "
33140 LOCATE 12,47:PRINT " Enter / when finished "
33150 TOTAL  = TOTAL + 1
33160 RETURN
34000 REM
34002 REM Subroutine to delete a data entry
34004 REM
34010 REM
34100 FOR N = 1 TO 50
34110 LINE (300,150-N)-(600,150+N),3,BF
34120 NEXT N
34130 COLOR 12,1: LOCATE 10,43: PRINT " Enter number of data pair "
34140 LOCATE 11,43: PRINT " to be deleted ( 1 -";TOTAL;") "
34150 COLOR 15,1: LOCATE 11,69: INPUT DEL$
34160 DEL  = VAL (DEL$)
34170 IF (DEL < 1 OR DEL > TOTAL) THEN BEEP: GOTO 34150
34180 COLOR 12,1:LOCATE 13,48: PRINT " Are you sure!!! (y/n) "
34190 ANSWERS  = ""
34200 WHILE (ANSWERS <> "y" AND ANSWERS <> "n")
34210 ANSWERS = INKEY
34220 IF (ANSWERS = "N" OR ANSWERS = "n") THEN RETURN
34230 WEND
34240 FOR N = DEL TO TOTAL-1
34250 FREQ$(N) = FREQ$(N+1)
34260 WIND$(N) = WIND$(N+1)
34270 NEXT N
34280 TOTAL  = TOTAL  - 1
34290 ROW  = ROW  - 1
34500 RETURN
35000 REM
35002 REM -----------------------------
35005 REM ------ Subroutine to confirm and allow re-entry of all data -----
35007 REM -----------------------------
35010 REM
35100 FOR N = 1 TO 65
35110 LINE (290,170-N)-(625,170+N),3,BF
35120 NEXT N
35130 COLOR 12,1: LOCATE 9,50: PRINT " PLEASE NOTE!!! "
35140 COLOR 15,1: LOCATE 11,39: PRINT " Proceeding further with this option "
35150 LOCATE 12,39: PRINT " will result in the deletion of all "
35160 LOCATE 13,39: PRINT " the data you have just entered. "
35170 COLOR 12,1: LOCATE 15,39: PRINT "Are you sure you wish to continue (y/n)"
35180 ANSWERS = ""
35190 WHILE (ANSWERS <> "Y" AND ANSWERS <> "y")
35200 ANSWERS = INKEYS
35210 IF (ANSWERS = "N" OR ANSWERS = "n") THEN RETURN
35220 WEND
35230 CLS
35240 ANSWERS = "Y"
35250 RETURN
40000 REM
40002 REM -----------------------------
40005 REM ------ Error checking subroutines -----
40007 REM -----------------------------
40100 REM
40200 REM
40202 REM ============ Check for errors during file opening.....
40204 REM
40230 IF ERR = 53 THEN GOSUB 40100: RESUME 1480
40232 IF ERR = 52 THEN GOSUB 40200: RESUME 1480
40234 IF ERR = 71 THEN GOSUB 40300: RESUME 1520
40236 IF ERR = 72 THEN GOSUB 40400
40238 IF (ERR = 5 OR ERR = 11) THEN GOSUB 40500: RESUME 65300
40240 IF ERR = 76 THEN GOSUB 40600: RESUME 65300
40250 STOP
40300 RETURN
40300 REM
40320 REM ============ File not found.....
40330 REM
40350 LINE (50,196)-(330,250),3,BF
40370 COLOR 12,1: LOCATE 15,11: PRINT " Sensor not found in database "
40390 COLOR 15,1: LOCATE 16,17: PRINT " Please try again "
40410 BEEP
40430 RETURN
40420 REM
40450 REM ============ Invalid filename.....
40460 REM
40480 LINE (50,196)-(330,250),3,BF
40500 COLOR 12,1: LOCATE 15,14: PRINT " Invalid filename used "
40520 COLOR 15,1: LOCATE 16,17: PRINT " Please try again "
40540 BEEP
40560 RETURN
40570 REM
40305 REM ===== Drive not ready.....
40310 REM
40315 LINE (50,196)-(330,250),3,BF
40320 COLOR 12,1: LOCATE 15,15: PRINT " Drive ";DRIVE$;" not ready 
40330 COLOR 15,1: LOCATE 16,13: PRINT " Press return when ready 
40340 BEEP
40350 ANSWERS = ""
40360 WHILE (ANSWERS <> CHR$(13))
40370 ANSWER$ = INKEY$
40380 WEND
40390 RETURN
40400 REM
40405 REM ===== Disk media error.....
40410 REM
40420 LINE (50,196)-(330,250),3,BF
40430 COLOR 12,1: LOCATE 15,11: PRINT " Error with hardware or disk 
40440 COLOR 15,1: LOCATE 16,17: PRINT " Program halted 
40490 RETURN
40500 REM
40502 REM ===== Illegal function call.....
40505 REM
40510 COLOR 12,8: LOCATE 17,33: PRINT "Insignificant data"
40520 LOCATE 18,26: PRINT "Could not calculate coefficients"
40530 COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
40540 WHILE (INKEY$ = "")
40550 WEND
40555 CLS
40560 REM
40562 REM ===== Exit program and return to operating system.....
40565 REM
40590 RETURN
40600 REM
40602 REM ===== Path not found.....
40605 REM
40610 CLS
40620 COLOR 12,1:LOCATE 17,24: PRINT "Could not find database on drive ";DRIVE$
40630 COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
40640 WHILE (INKEY$ = "")
40650 WEND
40655 CLS
40660 REM
40662 REM ===== Exit program and return to operating system.....
40665 REM
40690 RETURN
60000 REM
60001 REM -----------------------------------------------
60002 REM ===== Subroutine to calculate coefficients and write to file.....
60003 REM -----------------------------------------------
60005 REM
60009 REM *** This routine has to run under GWBASIC/d so that the ***
60010 REM *** mathematical routines can use double precision numbers ***
60020 REM access data file and read in data
60030 N=2 : REM * Set 3 degrees of fit *
60040 Z=0 : O=1 : K=12
REM zero data arrays
FOR I%=0 TO 40 : X(I%)=0:Y(I%)=0:II(I%)=0 : NEXT I%

REM FOR LOOP - 1 TO TOTAL
FOR LOOP - 1 TO TOTAL
    X(LOOP) = VAL (FREQ$(LOOP))
    Y(LOOP) = VAL (WIND$(LOOP))
    II(LOOP) = LOOP
    NEXT LOOP

MF = TOTAL
CLOSE#1
OPEN FILE$ FOR APPEND AS #2

REM * Initialise standard error array to missing values *
FOR I%=1 TO 8 : STDERR(I$)=9999 : NEXT I%

REM calculates the means for the data
T7#=T8#=W7#=Z:W7#=Z

FOR I%=1 TO MF
    H7#=W7#+X(II(1%)):T7#=T7*+Y(II(I%)):T8#=T8#+Y(II(I%))^2
    NEXT

T9*=(MF*W7#-T7'') / (MF^2-MF)

GOSUB 60870
GOSUB 61470
LOCATE 14,5: PRINT "Writing to calibration file : ";FILE$
REM ===== Write details to sensor calibration file =====
PRINT #2,"Multimet Sensor Calibration File"
PRINT #2, "Sensor ID: ", ID$ + SENSNUM$
PRINT #2, "Electronics: ", ELEC$
PRINT #2, "Date: ", DAY$; " / "; MONTH$; " / "; YEAR$
PRINT #2, "Sensor ID: ", ID$ + SENSNUM$
PRINT #2, "Operator: ", OPERNAME$
PRINT #2, "Standard: ", STANDARDS$
PRINT #2, "Comments: ", COMMENTS$
PRINT #2, "Poly.deg: N-1"
PRINT #2, "Equation val=C0+C1*hex+C2*hex^2+C3*hex^3+C4*hex^4"
PRINT #2, "Term Coefficient"
FOR J = 1 TO N
    PRINT #2, "C"; RIGHTS(STR$(J-1),1);,
    PRINT #2, TAB(8);U#(J)
NEXT J
FOR L = 1 TO MF
    Q8# = Y(II(L)) - C#(L)
    RMS# = RMS# + Q8#^2
    RMS# = RMS# + Q8#^2
60656 PRINT #2, "Call;";
60657 PRINT #2, TAB(13); USING "###.###"; VAL(FREQ$(L));
60660 PRINT #2, TAB(29); USING "###.###"; VAL(WIND$(L));
60665 PRINT #2, TAB(47); USING "###.###"; C#(L);
60667 PRINT #2, TAB(61); USING "###.###"; Q8#
60670 IF C#(L) = 0 THEN PRINT #2, "INFINITE"
60675 NEXT L
60680 PRINT #2, "Std error of estimate for Y="; STDERR(N-1)
60684 :
60685 CLOSE #2
60685 RETURN
60686 REM
60687 REM *** Routine arith1 (Copied from CALEDIT.BBC) ***
60688 :
60690 FOR I%=1 TO MF: P#(I%)=Z: Q#(I%)=0:NEXT
60690 FOR I%=1 TO 11: A#(I%)=Z: B#(I%)=2: S#(I%)=Z:NEXT
60690 E1#=Z: F1#=Z: W1#=MF:N4=K: I=1: K1=2
60690 :
60690 IF N=0 THEN GOTO 60990
60690 K1=N4
60695 :
60696 REM program loops to next line
60697 W#=Z
60698 :
60699 FOR L=1 TO MF: W#=W#+Y(II(L))*Q#(L): NEXT
60700 :
60701 S#(I)=W#/W1#
60700 IF I>N4>=0 THEN GOTO 61450
60700 IF I-MF>=0 THEN GOTO 61450
60700 IF I-N4>-0 THEN GOTO 61450
60700 IF I-MF>-0 THEN GOTO 61450
60700 E1#=E1#
60700 FOR L=1 TO MF
60700 A9# = ABS(Q#(L))
60700 IF A9#<09# THEN GOTO 61210
60700 X9# = ABS(X(II(L)))
60700 IF X9#<06# THEN GOTO 61170
60700 L2# = 0.434295*LOG(X9#)+2*.434295*LOG(A9#))
60700 IF L2#<36 THEN GOTO 61200
60700 IF L2#<36 THEN GOTO 61200
60700 L2#<36 THEN GOTO 61200
60700 REM SO X&Q TOO BIG FOR X*Q"2
60700 E1#=E1*+X(II(L)) *A9#''2
60700 NEXT
60700 IF L2#-(.434295*LOG(W1#))>-36 THEN GOTO 61250
60700 E1#=E1#/W1#
60700 A#(I+1)=E1#
60700 W#=Z
60700 FOR L=1 TO MF
61290  V#=(X(II(L))−E1#)*Q#(L)−F1#*P#(L)
61300  P#(L)=Q#(L)
61310  Q#(L)=V#
61320  V9#=ABS(V#)
61330  IF V9#<09* THEN GOTO 61390
61340  IF V9#<08# THEN GOTO 61380
61350  W#=O7#
61360  L=MF+1
61370  GOTO 61390
61380  W#=W#*V#*V#
61390  NEXT
61400  F1#=F#/W1#
61410  B#(I+2)=F1#
61420  W1#=W#
61430  I=I+1
61440  GOTO 60970
61450  RETURN
61460  REM
61470  REM *** Routine arith2 (Copied from CALEDIT.BBC)***
61480  FOR L=1 TO 13:G*(L)=Z:NEXT
61500  G#(2)=0
61520  FOR J=1 TO N
61530  S1#=Z
61540  FOR L=2 TO N+1
61550  IF L=2 THEN GOTO 61570
61560  G#(L)=G#(L)+A#(L-1)*G#(L-1)+B#(L-1)*G#(L-2)
61570  S1#=S1#+S*(L-1)*G#(L)
61580  NEXT
61590  U#(J)=S1#
61600  L=N+1
61610  FOR I2=2 TO N
61620  G#(L)=G#(L-1)
61630  L=L-1
61640  NEXT
61650  G#(2)=Z
61660  NEXT
61670  PRINT ""
61680  T=Z
61690  FOR L=1 TO MF
61700  C#(L)=Z
61710  J=N
61720  FOR I2=1 TO N
61730  C#(L)=(C#(L)*X(II(L)))+U#(J)
61740  J=J-1
61750  NEXT
61760  PRINT ""
61770  T3#=Y(II(L))−C#(L)
61780  T#=T#+T3#^2
61790  NEXT
61800  IF MF<<N THEN GOTO 61830
61810  T5#=0
61820  GOTO 61840
DETERM = INT(Q#*10000)/10000
STDERR(N-1)=SQR(T5#)
RETURN

REM

REM Subroutine to update database

REM Write name of new calibration file to ametdb directory.....
FILE1$ = DRIVES$ + AMETDB + ID$ + SENSNUM$ + ".L"
LOCATE 16,5: PRINT "Updating sensor file : ";FILE1$
OPEN FILE1$ FOR APPEND AS #1
PRINT #1, RIGnT$(FILE$, 22)
CLOSE #1

REM Write to history file date of calibration.....
FILE2$ = DRIVES$ + AMETDB\METDB_DA\HISTORY." + MOLTIMEI$
LOCATE 18,5: PRINT "Updating history file : ";FILE2$
OPEN FILE2$ FOR APPEND AS #2
PRINT #2, DAY$ + "/" + MONTH$ + "/" + YEARS + " Calibration."
CLOSE #2

LOCATE 20,22; PRINT "---Press Any Key for Main Menu---"
WHILE (INKEY$ = "")
RETURN

REM Subroutine for execution when Ctrl Esc has been pressed.....

REM MOD(24/7/90) BY R.P TO LOAD CALIBRATION MENU
RUN "CALMENU"
CLS
SCREEN 0,0,0
LOCATE 14,9: PRINT "Program Execution Halted..... Returning to operating system"
LOCATE 16,29: PRINT "---Press Any Key---"
WHILE (INKEYS = "")
RETURN

REM
REM *******************************************************************
REM ************************ TEMPCAL.BAS by RAF 17/5/90 *******************
REM *************************************************************************
REM =—.== Version 1.0 22/06/90
REM ===== This program was written in GW BASIC by Refayet Ahmed.
REM ===== This program is a copy of WINDCAL.BAS version 1.2 but has been
REM ===== converted to produce calibration files for temperature sensors
REM ===== rather than for wind sensors.
REM ===== All that has been changed is the names in the menu and the
REM ===== sensor type codes in the subroutine.
REM ===== The sensors which the program is able to calibrate are:
REM ===== Psychrometers TT (TW,TD)
REM ===== Air Temperatures AT
REM ===== SSTs SP
REM ===== Air Pressures AP
REM =====
REM ===== One additional difference in this program is that when
REM ===== Psychrometers are chosen the user must then also specify
REM ===== whether the calibration is wet or dry. Depending on this the
REM ===== calibration file will then be logged under TD or TS.
REM =====
REM R.W.P 24/7/90
REM modified to return user back to calibration menu

REM Main program
REM
REM ===== Declare all arrays
REM
DIMFREQ$(20), MIND$(20)
DIMX(40), Y(40), 11(40)
DIM (40), (40)
DIMA#(40), B#(40), S#(40), G#(40)

REM ===== Set error trap
REM
ON ERROR GOTO 40000

REM ===== Call subroutine to set up Ctrl Esc key trap
GOSUB 14000
KEY OFF
SCREEN 9

REM ===== Set up user menu
COLOR 7,1: CLS

REM ===== Draw up menu box
LINE (5,5)-(635,345),,B
LINE (180,83)-(480,266),2,BF
LINE (185,88)-(475,261),64,B
LOCATE 8,26:PRINT" Psychrometer ";CHR$(27);"
LOCATE 10,26:PRINT" Air Temperature "
LOCATE 12,26:PRINT" SST 
LOCATE 14,26:PRINT" Air pressure  
LOCATE 16,26:PRINT" Calibration Certificate 
LOCATE 18,26:PRINT" Return to Calibration Menu 
LOCATE 21,8:PRINT" Move arrow to required sensor type and press <RETURN> when finished 
LOCATE 22,26 : PRINT"Move around using cursor keys"
REM
REM ===== Define the cursor keys.....
REM
DN$=CHR$(0)+CHR$(80):UP$=CHR$(0)+CHR$(72)
X=-8:Y=-59:KY$="  
REM
Y1 = 97
GOSUB 20000
LOCATE X,Y
REM
REM ===== Reading input from keyboard....
REM
WHILE KY$<>CHR$(13)
    KY$=INKEY$
    IF (KY$=DN$) AND ((X>-8)AND(X<-16)) THEN LOCATE X,Y-3:PRINT"  ";X=X+2:LOCATE X,Y-2:PRINT CHR$(27);  :  GOSUB 20000
    IF (KY$=UP$) AND ((X>-10)AND(X<=18)) THEN LOCATE X,Y-3:PRINI"  ";X=X-2:LOCATE X,Y-2:PRINT CHR$(27);  :  GOSUB 20000
WEND
REM
REM ===== Determine which sensor has been chosen from the menu and go to subroutine which assigns sensor name & ID code ...
REM
REM ===== Call subroutine to get user details.....
REM
COLOR 12,1 : CLS
LOCATE 4,51  :  PRINT "Enter sensor number; e.g. VI1992";
COLOR 7,1
LOCATE 6,53 : PRINT "Press ";
COLOR 12,1
PRINT "<RETURN>";
COLOR 7,1
PRINT " when finished"
LINE (10,9)-(380,345),3,BF
COLOR 2,1 : LOCATE 10,(51-LEN (SENSORS))/2 : PRINT SENSORS;
COLOR 7,1 : LOCATE 13,15 : PRINT " Sensor ID : ";
LOCATE 13,28 : INPUT SENSNDMS
REM
REM ===== Open file in database to obtain multimet number.....
FILEIS - DRIVE$ + "XAMETDBX" + ID$ + SENSNDM$ + ".L"
OPEN FILE1$ FOR INPUT AS #1
IF EOF(1) THEN CLS : COLOR 4,14:L0CATE 16,16  : PRINT "EMPTY FILE IN DATA BASE!!! PROGRAM HALTED" : STOP
WEND
IF EOF(1))
INPUT #1,BLANK$
IF BLANK$ <> THEN RECORD$ = BLANK$
MULTIMET$ = RIGHT$(RECORD$,3)
REM
REM ===== If the sensor is a psychrometer then change the ID to either
REM TW (wet) or TD (dry).
REM
REM ===== Obtain todays date from the internal clock and convert to
REM Julian day....
DAY = VAL(DAY$)
MONTH = VAL(MONTH$)
JDAY$ = STR$(VAL(MID$("00003105909120151181212243273304334",((MONTH-1)*3)+1,3))+DAY)
JDAY$ = RIGHT$(JDAY$,3)
REM
REM ===== If there is no version code on the file then set one.....
REM
REM ===== If the Julian day on the file is the same as the present day
REM then increment the version code....
VCODE$ = RIGHT$(RECORD$,5)
CURJDAY$ = LEFT$(RIGHT$(RECORD$,10),3)
IF ASC(VCODE$) < 65 THEN ASVAL=65 ELSE IF CURJDAY$=JDAY$ THEN ASVAL = ASC(VCODE$)+1 ELSE ASVAL = 65
VCODE$ = CHR$(ASVAL)
REM
REM ===== Obtain the details of the sensor from the data base and check
REM with the user if the correct sensor has been chosen.....
CLS
FILE2$ = DRIVE$ + "\AMETDB\METDB_DA\DETAILS." + MULTIMET$
OPEN FILE2$ FOR INPUT AS #2
N = 5
WEND
1850 INPUT #2, RECORDS
1860 N = N + 1
1870 LOCATE N, 18: PRINT RECORDS
1880 WEND
1890 COLOR 12, 1
1900 LOCATE 23, 22: PRINT "Is this the required sensor? (y/n) :";
1905 ANSWERS = ""
1910 WHILE (ANSWERS <> "N" AND ANSWERS <> "Y")
1915 ANSWERS = INKEY$
1920 REM
1921 REM ===== If not required sensor then close the files and return to
1922 REM line 1400 where user inputs sensor number again.....
1925 REM
1930 IF (ANSWERS = "N" OR ANSWERS = "n") THEN CLOSE #1: CLOSE #2 : GOTO 1400
1940 REM
1942 REM ===== Accept upper or lower case "y"....
1945 REM
1950 IF ANSWERS = "y" THEN ANSWERS = "Y"
1955 WEND
1958 CLOSE #1, #2
1960 CLS
1967 REM
1968 REM ===== Build up filename to which to write.....
1969 REM
1970 FILES = DRIVES + "\AMETDB\METDB_DA\" + ID$ + JDAYS$ + YEARS + VCODES + "." + MULTIMETS
2000 REM
2002 REM ===== Call subroutine for editor to enter data.....
2005 REM
2010 GOSUB 30000
2020 REM
2025 REM ===== Clear screen and print message.....
2030 REM
2040 CLS
2050 COLOR 12, 8
2060 LOCATE 10, 33: PRINT " PLEASE WAIT ".
2065 COLOR 15, 8
2070 LOCATE 12, 27: PRINT " Calculating Coefficients ".
3000 REM
3002 REM ===== Call subroutine to calculate coefficients.....
3005 REM
3010 GOSUB 60000
3100 REM
3105 REM ===== Call subroutine to update data base.....
3110 REM
3120 GOSUB 65000
3150 REM
3155 REM ===== Return to main menu again.....
3160 REM
3170 GOTO 1056
9999 END
10000 REM
10002 REM ----------------------------------------
10005 REM ----- Subroutine for Psychrometer -----
10010 REM ----------------------------------------
10020 ID$ = "TT"
10030 SENSOR$ = "Psychrometer"
10040 HEADERS = "Temperature"
10050 REM
10055 REM ----- Ask user for type of Psychrometer (wet or dry)
10060 REM
10070 CLS
10080 LOCATE 12,10: PRINT "Enter type of Psychrometer Wet or Dry"
10090 COLOR 12,1
10100 LOCATE 14,10: PRINT "Press W or D"
10110 KY$ = ""
10120 WHILE (KY$ <> "w" AND KY$ <> "d" AND KY$ <> "W" AND KY$ <> "D")
10125 KY$ = INKEY$
10130 IF (KY$ = "w" OR KY$ = "W") THEN COND$ = "WET"
10140 IF (KY$ = "d" OR KY$ = "D") THEN COND$ = "DRY"
10150 WEND
10499 RETURN
10500 REM
10505 REM Subroutine for Air Temperature ----- Subroutine for SST ----- Subroutine for Air Pressure ----- Subroutine for Calibration Certificate -----
10550 HEADERS = "Temperature"
10999 RETURN
11000 REM
11005 REM Subroutine for Air Temperature ----- Subroutine for SST ----- Subroutine for Air Pressure ----- Subroutine for Calibration Certificate -----
11010 REM
11015 REM
11020 ID$ = "AT"
11030 SENSOR$ = "Air Temperature"
11040 HEADERS = "Temperature"
11490 RETURN
11500 REM
11505 REM Subroutine for Air Temperature ----- Subroutine for SST ----- Subroutine for Air Pressure ----- Subroutine for Calibration Certificate -----
11510 REM
11515 REM
11520 ID$ = "SP"
11530 SENSOR$ = "SST"
11540 HEADERS = "Temperature"
11560 RETURN
11500 REM
11505 REM Subroutine for Air Temperature ----- Subroutine for SST ----- Subroutine for Air Pressure ----- Subroutine for Calibration Certificate -----
11510 REM
11515 REM
11520 ID$ = "AP"
11530 SENSOR$ = "Air Pressure"
11540 HEADERS = "Pressure"
11560 RETURN
12000 REM
12002 REM Subroutine to set up key trap for Ctrl Esc -----
12000 REM
12002 REM Subroutine to set up key trap for Ctrl Esc -----
12005 COMMON PROGS,FILE1$,DRIVES
12015 COMMON PROGS,FILE1$,DRIVES
12020 PROGS = "TEMPCAL.BAS"
12030 IF FILE1$ = "" THEN CHAIN "CALCERT.BAS",1000 ELSE CHAIN "CALCERT.BAS",1500
12490 RETURN
14000 REM
14002 REM Subroutine to set up key trap for Ctrl Esc -----
14007 REM Subroutine to set up key trap for Ctrl Esc -----
Define keys to include all possible combinations of CAPS LOCK and NUM LOCK.....

KEY 15, CHRS($64)+CHR$(1)
KEY 16, CHRS($44)+CHR$(1)
KEY 17, CHRS($24)+CHR$(1)
KEY 18, CHRS($4)+CHR$(1)

Direct all escape key combinations to line 65300 ..... 

ON KEY (15) GOSUB 65300
ON KEY (16) GOSUB 65300
ON KEY (17) GOSUB 65300
ON KEY (18) GOSUB 65300

Turn all escape key combinations on.....

KEY (15) ON
KEY (16) ON
KEY (17) ON
KEY (18) ON

Subroutine to get user details

CLS
COLOR 15,8
USERDAYS$ ="
WHILE (USERDAYS$ <> "/")

Obtain date from user.
LOCATE 15,25: PRINT "Enter date dd/mm/yy 
LOCATE 16,25: PRINT "Or enter / for default of ";DAYS;"/";MONTHS;"/";YEARS;": "

Check if the date was entered in correct format.....
IF (VAL(DAYS) < 1 OR VAL(DAYS) > 31) OR (VAL(MONTHS) < 1 OR VAL(MONTHS) > 12) OR (VAL(YEARS) < 80 OR VAL(YEARS) > 99) THEN USERDATES = "": BEEP ELSE USERDATES = "/

REM

===== Obtain username....

CLS

DEFADLT$ = USERNAME$

IF DEFAULTS <> "" THEN LOCATE 16,25: PRINT "Enter / for ": DEFAULTS

LOCATE 15,25: INPUT "Please enter your username :"; OSERNAME$

IF OSERNAME$ = "/" THEN OSERNAME$ = DEFAULTS

CLS

REM

===== Obtain drive on which database is held.....

CLS

WHILE (USERDRIVES <> "/")

LOCATE 16,20: PRINT "Enter name of drive on which database is held"

LOCATE 16,20: PRINT "Enter / for default drive D: "

LOCATE 16,48: INPUT USERDRIVES$

IF USERDRIVES$ = "/" THEN DRIVE$ = "D:": GOTO 15450

REM

===== Check if drive has been specified correctly

IF USERDRIVES$ = "" THEN USERDRIVES$ = ": BEEP

IF (((LEN(USERDRIVES$) > 1) AND (MID$(USERDRIVES$,2,1) <> ":")) THEN USERDRIVES$ = ": BEEP

IF (((LEN(USERDRIVES$) > 2) AND (MID$(USERDRIVES$,3,1) <> "\") THEN USERDRIVES$ = ": BEEP

IF LEN(USERDRIVES$) > 3 THEN USERDRIVES$ = "": BEEP

DRIVECODE = ASC(LEFT$(USERDRIVES$,1))

IF (DRIVECODE >= 65 AND DRIVECODE <= 69) OR (DRIVECODE >= 97 AND DRIVECODE <= 101) THEN DRIVES$ = CHR$(DRIVECODE) + ":": USERDRIVES$ = "/" ELSE BEEP

WEND

REM

===== User enters standard to which sensor is calibrated against.....

CLS

LOCATE 16,10: PRINT "Enter the standard to which the sensor is calibrated against"

LOCATE 16,15: PRINT "Enter / for default standard of F25 S3633"

LOCATE 16,57: INPUT STANDARDS$

IF STANDARDS$ = "/" THEN STANDARDS$ = "F25 S3633"

CLS

REM

===== User enters any comments.....

CLS

LOCATE 15,20: PRINT "Enter any comments (60 chars only)"

LOCATE 16,20: PRINT "Enter / for blank"

LOCATE 16,39: INPUT COMMENTS$

IF COMMENTS$ = "/" THEN COMMENTS$ = ""

RETURN

REM

REM Subroutine for drawing & flashing pointer box in menu -----

REM
20050 REM ===== First overwrite old box.....
20060 REM
20070 LINE (199,Y1)-(464,Y1+15),2,B
20100 REM ===== Now draw box and flash at new position.....
20110 REM
20120 IF (KY$ = DN$) THEN Y1 = Y1 + 28 ELSE IF (KY$ = UP$) THEN Y1 = Y1 - 28
20140 FOR COL = 9 TO 15
20150 LINE (199,Y1)-(464,Y1+15),COL,B
20160 NEXT COL
20170 LINE (199,Y1)-(464,Y1+15),12,B
20500 RETURN
30000 REM
30005 REM
30010 REM Subroutine for entering and editing data ----
30015 REM ........................................................................
30020 REM
30030 REM Set up screen.....
30035 REM
30050 LINE (5,5)-(315,345),15,B
30060 LINE (321,0)-(640,350),3,BF
30070 COLOR 15,1: LOCATE 2,45: PRINT " Enter the calibration data 
" 30080 LOCATE 4,45: PRINT " Enter / when finished 
" 30085 COLOR 12,1: LOCATE 6,45: PRINT " NOTE: 
" 30090 COLOR 15,1: PRINT "Max No of data pairs = 20 
" 30110 TOTAL = 1
30115 ROW = 4
30120 REM
30130 REM Enter frequency and actual wind speed data pairs.....
30140 REM
30145 COLOR 4,1:LOCATE 2,3: PRINT "No Frequency Actual ";HEADER$
30147 WHILE (FREQ$(TOTAL) <> ")"
30150 COLOR 12,1: LOCATE 2,8: PRINT "Frequency"
30155 COLOR 7,1:LOCATE ROW,2:PRINT TOTAL
30160 LOCATE ROW,9: INPUT FREQ$(TOTAL)
30170 COLOR 4,1: LOCATE 2,8: PRINT "Frequency"
30180 IF FREQ$(TOTAL) = "/" THEN GOTO 30230
30190 COLOR 12,1: LOCATE 2,21: PRINT "Actual Wind Speed"
30200 COLOR 7,1: LOCATE ROW,24: INPUT WIND$(TOTAL)
30210 COLOR 4,1: LOCATE 2,21: PRINT "Actual Wind Speed"
30220 TOTAL = TOTAL + 1: ROW = ROW + 1
30225 IF TOTAL = 21 THEN FREQ$(TOTAL) = "/"
30230 WEND
30240 FREQ$(TOTAL) = ""
30250 TOTAL = TOTAL - 1
30500 REM
30502 REM List data and show editor options.....
30503 REM
30510 CLS
30520 LINE (5,5)-(315,345),15,B
30530 LINE (321,0)-(635,345),15,B
30535 PAINT (400,200),2,30
30540 COLOR 4,1:LOCATE 2,3: PRINT "No Frequency Actual ";HEADER$
30545 COLOR 7,1
30550 FOR N = 1 TO TOTAL
30560 LOCATE N+3,2: PRINT N
30570 LOCATE N+3,8: PRINT USING "####.###"; VAL (FREQ$(N))
30580 LOCATE N+3,25: PRINT USING "####.###"; VAL (WIND$(N))
30600 NEXT N
30610 FOR N = 1 TO 100
30620 LINE (450-N,150-N)-(450+N,150+N),8,BF
30650 NEXT N
30660 COLOR 15,1
30680 LOCATE 7,51: PRINT "Add"
30690 LOCATE 9,51: PRINT "Delete"
30700 LOCATE 11,51: PRINT "Re-enter 'R'"
30710 LOCATE 13,51: PRINT "Finish 'F'"
30730 LOCATE 20,45: PRINT "Press one of keys shown "
30740 LOCATE 21,45: PRINT "for required option "
30750 REM
30760 REM ----- Read users keystroke from and call appropriate subroutine.....
30770 REM
30780 REM ----- Line 32000 ----- Subroutine to change a data entry.
30790 REM ----- Line 33000 ----- Subroutine to add to the data.
30800 REM ----- Line 34000 ----- Subroutine to delete a data entry.
30810 REM ----- Line 35000 ----- Subroutine to confirm re entry of all data.
30820 REM ----- Line 30500 ----- Point where data is shown and editor options
30830 REM are listed.
30840 REM ----- Line 30050 ----- Beginning of present subroutine where all the
30850 REM data is entered.
30860 REM ----- Line 30120 ----- Point where data is entered.
30870 REM
30880 REM
30890 KY$ = ""
30900 WHILE (KY$<>"F" AND KY$<>"f")
30910 KYS = INKEYS
30920 IF (KYS = "C" OR KYS = "c") THEN GOSUB 32000: GOTO 30500
30930 IF (KYS = "A" OR KYS = "a") THEN GOSUB 33000: GOTO 30120
30940 IF (KYS = "D" OR KYS = "d") THEN GOSUB 34000: GOTO 30500
30950 IF (KYS = "R" OR KYS = "r") THEN GOSUB 35000: IF ANSWER = "Y" THEN GOTO 30050 ELSE GOTO 30500
30970 WEND
31000 RETURN
31002 REM
31003 REM ----- Subroutine to change data entry from editor menu -----
31007 REM
31010 REM
31015 FOR N = 1 TO 50
31020 LINE (300,150-N)-(600,150+N),3,BF
31025 NEXT N
31050 COLOR 12,1: LOCATE 9,43: PRINT " Enter number ( 1 -";TOTAL;") which "
31060 LOCATE 10,47: PRINT " you wish to change:- "
31070 COLOR 15,1: LOCATE 10,68: INPUT CHANGES
31080 IF (VAL(CHANGES) < 1 OR VAL(CHANGES) > TOTAL) THEN BEEP: GOTO 32150
31090 CHANGES = VAL (CHANGES)
31100 LOCATE 12,47: PRINT " Enter the changes now "
31110 LOCATE CHANGE+3,2: PRINT CHANGES;
31120 LOCATE CHANGE+3,2: PRINT "Frequency"
32220 COLOR 15,1: LOCATE CHANGE+3,9: INPUT FREQ$(CHANGE)
32230 COLOR 4,1: LOCATE 2,8: PRINT "Frequency"
32240 COLOR 12,1:LOCATE 2,21: PRINT "Actual Wind Speed"
32250 COLOR 15,1: LOCATE CHANGE+3,24: INPUT WIND$(CHANGE)
32300 RETURN
33000 REM
33002 REM --------------------------------------------------
33005 REM ----- Subroutine to add to the data -----  
33010 REM
33100 FOR N = 1 TO 50
33110 LINE (300,150-N)-(600,150+N),3,BF
33120 NEXT N
33130 COLOR 12,1: LOCATE 10,48: PRINT " Add to the data now "
33140 LOCATE 12,47:PRINT " Enter / when finished "
33145 TOTAL = TOTAL + 1
33150 RETURN
34000 REM
34002 REM --------------------------------------------------
34005 REM ----- Subroutine to delete a data entry -----  
34010 REM
34100 FOR N = 1 TO 50
34110 LINE (300,150-N)-(600,150+N),3,BF
34120 NEXT N
34130 COLOR 12,1: LOCATE 10,43: PRINT " Enter number of data pair "
34140 LOCATE 11,43: PRINT " to be deleted ( 1 -";TOTAL;") "
34150 COLOR 15,1: LOCATE 11,69: INPUT DEL$
34160 DEL = VAL (DEL$)
34170 IF (DEL < 1 OR DEL > TOTAL) THEN BEEP: GOTO 34150
34180 COLOR 12,1: LOCATE 13,48: PRINT " Are you sure!!! (y/n) "
34190 ANSWER$ = ""
34200 WHILE (ANSWER$ <> "Y" AND ANSWER$ <> "y")
34210 ANSWER$ = INKEYS
34220 IF (ANSWER$ = "N" OR ANSWER$ = "n") THEN RETURN
34230 WEND
34240 FOR N = DEL TO TOTAL-1
34250 FREQ$(N) = FREQ$(N+1)
34260 WIND$(N) = WIND$(N+1)
34270 NEXT N
34280 TOTAL = TOTAL - 1
34290 ROW = ROW - 1
34500 RETURN
35000 REM
35002 REM --------------------------------------------------
35005 REM ----- Subroutine to confirm and allow re-entry of all data -----  
35007 REM --------------------------------------------------
35010 REM
35100 FOR N = 1 TO 65
35110 LINE (290,170-N)-(625,170+N),3,BF
35120 NEXT N
35130 COLOR 12,1: LOCATE 9,50: PRINT " PLEASE NOTE!!! "
35140 COLOR 15,1: LOCATE 11,39: PRINT " Proceeding further with this option "
35150 LOCATE 12,39: PRINT " will result in the deletion of all "
35160 LOCATE 13,39: PRINT " the data you have just entered. "
35170 COLOR 12,1: LOCATE 15,39: PRINT "Are you sure you wish to continue (y/n)"
35180 ANSWER$ = ""
35190 WHILE (ANSWER$ <> "Y" AND ANSWER$ <> "y")
35200 ANSWER$ = INKEY$
35210 IF (ANSWER$ = "N" OR ANSWER$ = "n") THEN RETURN
35220 WEND
35230 CLS
35240 ANSWER$ = "Y"
35250 RETURN
40000 REM
40002 REM
40005 REM ===== Check for errors during file opening.....
40006 REM
40007 IF ERR = 53 THEN GOSUB 40100: RESUME 1480
40008 IF ERR = 52 THEN GOSUB 40200: RESUME 1480
40009 IF ERR = 71 THEN GOSUB 40300: RESUME 1520
40010 IF ERR = 72 THEN GOSUB 40400
40011 IF (ERR = 5 OR ERR = 11) THEN GOSUB 40500: RESUME 65300
40012 IF ERR = 76 THEN GOSUB 40600: RESUME 65300
40013 STOP
40014 REM
40015 REM ===== File not found.....
40016 REM
40017 LINE (50,196)-(330,250),3,BF
40018 COLOR 12,1: LOCATE 15,11: PRINT " Sensor not found in database "
40019 COLOR 15,1: LOCATE 16,17: PRINT " Please try again "
40020 BEEP
40021 RETURN
40022 REM
40023 REM ===== Invalid filename.....
40024 REM
40025 LINE (50,196)-(330,250),3,BF
40026 COLOR 12,1: LOCATE 15,14: PRINT " Invalid filename used "
40027 COLOR 15,1: LOCATE 16,17: PRINT " Please try again "
40028 BEEP
40029 RETURN
40030 REM
40031 REM ===== Drive not ready.....
40032 REM
40033 LINE (50,196)-(330,250),3,BF
40034 COLOR 12,1: LOCATE 15,15: PRINT " Drive ":DRIVES:\ not ready "
40035 COLOR 15,1: LOCATE 16,13: PRINT " Press return when ready "
40036 BEEP
40037 REM
40038 REM
40039 REM
40040 REM
40405 REM ===== Disk media error....
40410 REM
40420 LINE (50,196)-(330,250),3,BF
40430 COLOR 12,1:LOCATE 15,11:PRINT " Error with hardware or disk "
40440 COLOR 15,1:LOCATE 16,17:PRINT " Program halted "
40450 RETURN
40455 REM
40460 REM ===== Illegal function call.....
40465 REM
40470 COLOR 12,8:LOCATE 17,33:PRINT "Insignificant data"
40480 LOCATE 18,26:PRINT "Could not calculate coefficients"
40490 COLOR 15,8:LOCATE 20,32:PRINT " Press Any Key ---"
40500 WHILE (INKEY$ = "")
40505 WEND
40510 CLS
40520 REM
40525 REM ===== Exit program and return to operating system.....
40530 REM
40535 REM
40540 REM ===== Path not found.....
40545 REM
40550 REM
40555 REM
40560 REM
40565 REM
40570 REM
40575 REM
40580 REM
40585 REM
40590 RETURN
60000 REM
60001 REM
60002 REM Subroutine to calculate coefficients and write to file.....
60003 REM
60004 REM
60005 REM
60009 REM *** This routine has to run under GWBASIC/d so that the ***
60010 REM *** mathematical routines can use double precision numbers ***
60020 REM access data file and read in data
60030 N=4 : REM * Set 3 degrees of fit *
60040 K=0 : O=1 : E=12
60050 O6#=-1E-18 : O8#=-1E+18 : O7#=-9.999999E+35 : O6#=-1E-36
60060 :
60065 :
60070 REM zero data arrays
60080 FOR I%=0 TO 40 : X(I%)=0:Y(I%)=0:II(I%)=0 : NEXT I%
60090 :
60100 FOR LOOP = 1 TO TOTAL
60110 X(LOOP) = VAL (FREQS(LOOP))
60120 Y(LOOP) = VAL (WINDS(LOOP))
60130 II(LOOP) = LOOP
60140 NEXT LOOP
60150 MF = TOTAL
60290:  60300 CLOSE#1  60310 OPEN FILES FOR APPEND AS #2  60320:  60330 REM * Initialise standard error array to missing values *  60340 FOR I%-1 TO 8 : STDERR(I%)=9999 : NEXT 1%  60350:  60360 REM calculates the means for the data  60370 T7#-Z:T8#=Z:H1#=Z  60380:  60390 FOR I%=1 TO MF  60400 W7#-W7#+X(II(I%)):T7f-T7#+Y(II(1%)):T8*=T8*+Y(II(I%))“2  60410 NEXT  60420 T9#-(MF*T8#-T7*“2)/(MF“2-MF)  60430:  60520 GOSUB 60870  60530 GOSUB 61470  60534 LOCATE 14,5: PRINT "Writing to calibration file : ";FILES  60540 REM  60542  REM ===== Write details to sensor calibration file  60545 REM  60547 PRINT *2, "%CO:A,12,72%"  60550 PRINT *2,"Multimet Sensor Calibration File"  60555 PRINT *2,  60560 PRINT *2, "Sensor ID:", IDS + SENSNUM$  60565 PRINT *2, "Electronics:", ELECS  60570 PRINT *2, "Date:",DAYS;"/";MONTH$;"/";YEARS  60575 PRINT *2, "Operator:", USERNAMES  60580 PRINT *2, "Standard:", STANDARDS  60585 PRINT *2, "Comments:", COMMENTS  60590 PRINT *2,  60595 PRINT *2, "Poly.deg:N-1"  60600 PRINT *2, "Equation val=C0+C1*hex+C2*hex“2+C3*hex“3+C4*hex“4"  60605 PRINT *2, "Term Coefficient"  60610 FOR J - 1 TO N  60615 PRINT *2,"C";RIGHT$(STR$(J-1),1),1;  60625 PRINT *2, TAB(8);U#(J)  60630 NEXT J  60632 PRINT *2, ""  60635 PRINT *2, TAB(13),"Frequ."; TAB(29),HEADERS;TAB(49);"Y-cal";TAB(64);"Diff"  60640 RMS$ - 0  60645 FOR L = 1 TO MF  60650 Q8* = Y(II(L)) - C$(L)  60655 RMS$ = RMS$ + Q8*“2  60656 PRINT *2, "Cal:";  60657 PRINT *2, TAB(13); USING "#####"; VAL(FREQ$(L));  60660 PRINT *2, TAB(29); USING "#####"; VAL(WIND$(L));  60665 PRINT *2, TAB(47); USING "#####";C$(L);  60667 PRINT *2, TAB(61); USING "#####";Q8#  60670 IF C$(L) = 0 THEN PRINT *2, "INFINITE"  60675 NEXT L  60680 PRINT *2, "Std error of estimate for Y=";STDERR(N-1)  60690 PRINT *2, "%PO%"  60695 PRINT *2, "%CO:B,12,60%CO:C,12,48%CO:D,12,36%CO:E,12,24%CO:F,12,12%"
60840:
60845 CLOSE #2
60850 RETURN
60860 REM
60870 REM *** Routine arithl ( Copied from CALEDIT.BBC ) ***
60880:
60890 FOR I%=1 TO MF: P%(I%) = Z; Q%(I%) = 0: NEXT
60900 FOR I%=1 TO 11: A%(I%) = Z; B%(I%) = 2: S%(I%) = Z; NEXT
60910 E1# = Z; F1# = Z; W1# = MF; N4 = K; I-1; K1 = 2
60920:
60930 IF N = 0 THEN GOTO 60990
60940 K1 = N4
60950:
60960 REM program loops to next line
60970 M# = Z
60980:
60990 FOR L = 1 TO WF: M* = W* + Y(II(L)) * Q#(L): NEXT
61000:
61010 IF I- N4 > -0 THEN GOTO 61450
61020 IF I- MF > -0 THEN GOTO 61450
61030 E1# = Z
61040 FOR L = 1 TO MF
61050 A9# = ABS(Q#(L))
61060 IF A9# < 09# THEN GOTO 61210
61070 X9# = ABS(X(II(L)))
61080 IF X9# < 06# THEN GOTO 61170
61090 L2# = 434295 * LOG(X9#) + (2 * .434295 * LOG(A9#))
61100 IF L2# < 36 THEN GOTO 61200
61110 REM SO X&Q TOO BIG FOR X*Q^2
61120 E1# = E1# + X(II(L)) * A9# * 2
61130 L2# = 36
61140 E1# = 5#
61150 L = MF + 1
61160 GOTO 61210
61170 IF A9# < 1 THEN GOTO 61210
61180 IF A9# < 08# THEN GOTO 61200
61190 REM SO Q TOO BIG FOR Q^2
61200 E1# = E1# + X(II(L)) * A9# * 2
61210 NEXT
61220 IF L2# = (.434295 * LOG(W1#)) > -36 THEN GOTO 61250
61230 E1# = 0
61240 GOTO 61260
61250 E1# = E1# / W1#
61260 A%(I+1) = E1#
61270 W# = Z
61280 FOR L = 1 TO MF
61290 V# = (X(II(L)) - E1#) * Q#(L) - F1# * P#(L)
61300 P#(L) = Q#(L)
61310 Q#(L) = V#
61320 V9# = ABS(V#)
61330 IF V9# < 09# THEN GOTO 61390
61340 IF V9# < 08# THEN GOTO 61380
61350 W# = 07#
61360 L = MF + 1
61370 GOTO 61390  
61380 W#-W#+V#*V#  
61390 NEXT  
61400 F1#-W#/W1#  
61410 B#(I+2)=F1#  
61420 W1#=W#  
61430 I=I+1  
61440 GOTO 60970  
61450 RETURN  
61460 REM  
61470 REM *** Routine arith2 (Copied from CALEDIT.BBC)***  
61480 :  
61490 FOR L=1 TO 13:G#(L)=Z:NEXT  
61500 :  
61510 G#(2)=0  
61520 FOR J=1 TO N  
61530 S1#=Z  
61540 FOR L=2 TO N+1  
61550 IF L=2 THEN GOTO 61570  
61560 G*(L)(L)-G#(L)*A#(L-1)*G#(L-1)-B#(L-1)*G#(L-2)  
61570 S1#=S1#+S#(L-1)*G#(L)  
61580 NEXT  
61590 U#(J)=S1#  
61600 L=N+1  
61610 FOR I2=2 TO N  
61620 G#(L)=G#(L-1)  
61630 L=L-1  
61640 NEXT  
61650 G#(2)=Z  
61660 NEXT  
61670 :  
61680 PRINT " "  
61690 T=Z  
61700 FOR L=1 TO MF  
61710 C#(L)=Z  
61720 J=N  
61730 FOR I2=1 TO N  
61740 C#(L)=(C#(L)*X(I2(L)))+U#(J)  
61750 J=J-1  
61760 NEXT  
61770 T3#=Y(I2(L))-C#(L)  
61780 T#=T#+T3#^2  
61790 NEXT  
61800 IF MF<>N THEN GOTO 61830  
61810 T5#=0  
61820 GOTO 61840  
61830 T5#=T#/((MF-N))  
61840 Q7#=-1-T#/((T9#*(MF-1)))  
61850 :  
61860 DETERM = INT(Q7#*10000)/10000  
61870 STDERR(N-1)=SQR(T5#)  
61880 RETURN  
65000 REM  
65002 REM ----------------------------------------
65005 REM ----- Subroutine to update database ----- 
65007 REM -------------------------------------------------------------------------- 
65010 REM 65015 REM ----- If Psychrometers has been chosen then change ID back to TT..... 
65020 REM 65030 IF (ID$ = "TW" OR ID$ = "TD") THEN ID$ = "TT" 
65045 REM 65046 REM ----- Write name of new calibration file to ametdb directory..... 
65047 REM 65050 FILE1$ = DRIVE$ + "XAMEIDBX" + ID$ + SENSNUM$ + ".L" 
65055 LOCATE 16,5: PRINT "Updating sensor file : ";FILE1$ 
65060 OPEN FILE1$ FOR APPEND AS #1 
65070 PRINT #1,RIGHT$(FILE$,22) 
65080 CLOSE #1 
65090 REM 65095 REM ----- Write to history file date of calibration..... 
65100 REM 65110 FILE2$ = DRIVE$ + "\AMETDB\MEIDB_DA\HISTORY." + MULTIMET$ 
65115 LOCATE 18,5: PRINT "Updating history file : ";FILE2$ 
65120 OPEN FILE2$ FOR APPEND AS #2 
65130 PRINT #2, DAY$ + "/" + MONTH$ + "/" + YEAR$ + " Calibration." 
65140 CLOSE #2 
65150 LOCATE 20,22: PRINT "----Press Any Key for Main Menu----" 
65160 WHILE (INKEY$ = ")") 
65170 WEND 
65180 RETURN 
65190 REM 65200 REM 65210 REM ------------------------------------------------------------------------- 
65215 REM ----- Subroutine for execution when Ctrl Esc has been pressed ----- 
65220 REM -------------------------------------------------------------------------- 
65230 REM 65235 RUN"calmenu" 
65240 CLS 
65250 SCREEN 0,0,0 
65260 LOCATE 14,9: PRINT "Program Execution Halted..... Returning to operating system" 
65270 LOCATE 16,29: PRINT "----Press Any Key----" 
65280 WHILE (INKEY$ = ")") 
65290 WEND 
65300 SYSTEM
**CALCERT.BAS by RAF 18/6/90**

This program is executed when a calibration certificate is asked for by one of two programs, either WINDCAL.BAS or BATHCON.BAS

This program produces a sensor calibration certificate which is a comparison between two sets of calibration coefficients from different calibrations.

The certificate which is to be produced is by default that of the sensor most recently calibrated. However the user may choose a different sensor if required.

The names of the two most recent calibration files are taken from the AMETDB directory. (It need not be the two most recent; the user may specify the calibration file if required).

From each file the coefficients are read. Then for both sets of coefficients Y-Cals are calculated using test data. The differences between the two sets of Y-Cals are calculated and then mean and standard deviation of these differences are calculated.

Finally the output is sent to the printer in the form of a certificate and execution is returned to the calling program.

If program is not executed via a chained program then get specifications from user.....

Obtain from the user the drive on which the database is held.....

Get from N = 1 TO 50

Locate 17,44: Print "Enter letter of drive on"

Locate 18,45: Print " which database is held"

Color 15,1: Locate 20,44: Print " Enter / for default (D)"

While (Asc(Drive$) < 65 OR Asc(Drive$) > 69) AND (Asc(Drive$) < 97 OR Asc(Drive$) > 101) AND (Drive$ <> "/")
DRIVES = INKEY$
IF DRIVES = "" THEN DRIVES = ""
WEND
IF DRIVES = "/" THEN DRIVES = "D"
DRIVES = DRIVES + ":"
CLS
REM
REM ===== Set up screen in high resolution for graphics.....
REM
KEY OFF
SCREEN 9
CLS
COLOR 15,1: CLS
REM
REM ===== Declare all arrays.....
DIM CALFILES(20)
DIM COEF1(8), COEF2(8), C(8)
DIM YCAL1(10), YCAL2(10), DIFF(10)
DIM JDAYFILE(20)
DIM HEX(IO)
REM
REM ===== Obtain from user sensor name and show default.....
DIMFILE$ = DRIVES + "XametdbX" + IDS + SERIALS + ".1"
REM ===== Open the sensor file in the AMETDB directory then read and store
OPEN FILE$ FOR INPUT AS #1
TOTAL = 1
WHILE NOT(EOF(1))
INPOT #1,RECORDS
CHECKS = MIDS(RECORDS,11,8)
IF RECORDS <> "" AND CHECKS <> "HISTORY." AND CHECKS <> "DETAILS." AND CHECKS <> "history." AND CHECKS <> "details." THEN CALFILES(TOTAL) = RECORDS: TOTAL = TOTAL + 1
WEND
TOTAL = TOTAL - 1
CLOSE #1
REM
REM ===== Check that at least two calibration files exist.....
REM
REM ===== Call subroutine to sort calfiles in accending order of date.....
REM
REM ===== Call subroutine for menu to choose two calibration files.....
REM
GOSUB 10000
CLS
REM ===== Open the two calibration files....
CALFILE1$ = DRIVE$ + "\AMETDB" + CALFILE$(C1)
CALFILE2$ = DRIVE$ + "\AMETDB" + CALFILE$(C2)
OPEN CALFILE1$ FOR INPUT AS #1
OPEN CALFILE2$ FOR INPUT AS #2
REM ===== Determine from the sensor id code the number of coefficients....
IF (LEFT$(ID$,1) = "W" OR LEFT$(ID$,1) = "w") THEN RESTORE 9000: COEFTOT = 2 ELSE RESTORE 9100
COEFTOT = 4
REM ===== Call subroutine to pick out the coefficients from the Cal file...
FILENUM = 1
GOSUB 20000
CLOSE #1
FOR N = 0 TO 8
  COEFl(N) = C(N)
NEXT N
FILENUM = 2
GOSUB 20000
CLOSE #2
FOR N = 0 TO 8
  COEF2(N) = C(N)
NEXT N
REM ===== Call subroutine to calculate Y-Cals....
FILENUM = 1
GOSUB 21000
REM ===== Calculate the difference and sum the difference of the Y-cals....
SUMDIFF = 0
FOR N = 1 TO 10
  DIFF(N) = YCAL1(N) - YCAL2(N)
  SUMDIFF = SUMDIFF + DIFF(N)
NEXT N
REM ===== Calculate the mean and standard deviation....
MEAN = SUMDIFF/10
REM ===== Show user results....
COLOR 15,1
CLS
COLOR 15,1
REM ===== Search data for name of sensor....
RESTORE 9220
2750 WHILE (IDS <> CH1$ AND IDS <> CH2$)
2760 READ CH1$, CH2$, SENSORS
2770 WEND
2780 PRINT TAB(5); SENSORS, SERIALS
2790 PRINT TAB(5); "Cal "; MID$(CALFILE$(C1), 11, 8)
2800 FOR N = 0 TO COEFTOT - 1
2810 PRINT TAB(20); "C"; N; COEF1(N)
2820 NEXT N
2830 PRINT
2840 FOR N = 0 TO COEFTOT - 1
2850 PRINT TAB(20); "C"; N; COEF2(N)
2860 NEXT N
2870 PRINT
2880 PRINT TAB(5); "Freq"; TAB(17); "Temp A"; TAB(28); " Temp B"; " Diff"
2890 FOR N = 1 TO 10
2900 PRINT TAB(5); HEX(N), YCAL1(N), YCAL2(N), DIFF(N)
2910 NEXT N
2912 PRINT TAB(5); "Mean = "; SUMDIFF/10
2920 COLOR 12, 1
2930 LOCATE 24, 5: PRINT "Press P to print or RETURN to get back to menu"
2940 KY$ = ""
2950 WHILE KY$ <> CHR$(13)
2960 KY$ = INKEY$
2970 WHILE KY$ = "p" OR KY$ = "P"
2971 LPRINT CHR$(27); CHR$(69);
2972 LPRINT CHR$(27); CHR$(104); CHR$(1);
2974 LPRINT CHR$(27); CHR$(97); CHR$(49);
2975 LPRINT "Calibration Certificate"
2980 LPRINT TAB(17); SENSORS, SERIALS$: LPRINT
2985 LPRINT: LPRINT CHR$(27); CHR$(119); CHR$(48);
2987 LPRINT CHR$(27); CHR$(97); CHR$(48);
2989 LPRINT CHR$(27); CHR$(14);
2990 LPRINT "Date Produced : "; DATE$: LPRINT
2995 LPRINT TAB(17); "Cal "; MID$(CALFILE$(C1), 11, 8)
3000 FOR N = 0 TO COEFTOT - 1
3010 LPRINT TAB(32); "C"; N; COEF1(N)
3020 NEXT N
3030 LPRINT
3040 LPRINT
3050 LPRINT TAB(17); "Cal "; MID$(CALFILE$(C2), 11, 8)
3060 FOR N = 0 TO COEFTOT - 1
3070 LPRINT TAB(32); "C"; N; COEF2(N)
3080 NEXT N
3090 LPRINT
3100 LPRINT TAB(17); "Freq"; TAB(29); "Temp A"; TAB(40); " Temp B"; " Diff"
3110 FOR N = 1 TO 10
3120 LPRINT TAB(17); HEX(N), YCAL1(N), YCAL2(N), DIFF(N)
3130 NEXT N
3140 LPRINT CHR$(12);
3150 KY$ = ""
3160 WEND
Sample data emulating sensor frequencies:

DATA 1,4,7,10,13,16,19,22,25,28
DATA 1400,1500,1600,1700,1800,1900,2000,2100,2200,2300

Data connecting type of sensor with the id code:

DATA "WS","ws","Wind Speed"
DATA "WW","ww","Wind Velocity"
DATA "WD","wd","Wind Direction"
DATA "TT","tt","Psychrometer"
DATA "SP","sp","SST"
DATA "AT","at","Air Temperature"
DATA "AP","ap","Air Pressure"
10215 KY$ = ""
10220 WHILE KY$ <> "/"
10230   KY$ = INKEY$
10240   IF KY$ = "1" THEN ROW2 = 3: GOSUB 12000: C1 = POINTER
10250   IF KY$ = "2" THEN ROW2 = 5: GOSUB 12000: C2 = POINTER
10260 WEND
10265 KY$ = ""
10270 REM
10275 REM ----- Check that the two chosen Cal files are not the same.....
10280 REM
10290 IF CALFILE$(C1) = CALFILE$(C2) THEN BEEP: GOTO 10190
11000 RETURN
12000 REM
12005 REM
12010 REM ---- Subroutine to allow user to choose a calibration file ----
12015 REM
12020 REM
12030 PAINT (400,300),0,15
12032 COLOR 10,1
12033 LOCATE 8,35: PRINT "Use UP DOWN keys to move cursor"
12035 COLOR 12,1
12036 LOCATE 10,35: PRINT "Press RETURN to make selection"
12040 REM ----- Define up down keys.....
12050 REM
12060 DN$ = CHR$(0) + CHR$(80)
12070 UP$ = CHR$(0) + CHR$(72)
12080 REM
12090 REM ----- Scan keyboard for Return and Up Down keys.....
12100 REM
12110 KY$ = ""
12113 COL = 12
12115 GOSUB 13000
12120 WHILE KY$ <> CHR$(13)
12130   KY$ = INKEY$
12140   IF (KY$ = UP$ AND POINTER > 1) THEN IF (ROW = 4) THEN GOSUB 12400 : COL = 12:GOSUB 13000 ELSE COL = 15:GOSUB 13000: POINTER = POINTER - 1: ROW = ROW - 2: COL = 12 :GOSUB 13000
12145   IF (KY$ = DN$ AND POINTER < TOTAL) THEN IF (ROW = 22) THEN GOSUB 12500: COL = 12: GOSUB 13000 ELSE COL = 15: GOSUB 13000: POINTER = POINTER + 1: ROW = ROW + 2: COL = 12: GOSUB 13000
12160 WEND
12165 COL = 15: GOSUB 13000
12170 KY$ = ""
12180 PAINT (400,300),1,15
12181 COLOR 10,1
12182 LOCATE 8,35: PRINT "Enter / to produce a calibration certificate"
12183 LOCATE 9,35: PRINT "of the files shown above"
12184 COLOR 12,1
12185 LOCATE 11,35: PRINT "Press 1 or 2 to change the corresponding"
12186 LOCATE 12,35: PRINT "calibration file to one of those shown on"
12187 LOCATE 13,35: PRINT "the left."
12300 RETURN
12400 REM
12401 REM -----------------------------------------
12402 REM ---- Subroutine to scroll screen up one place -----
12403 REM - ------------------------------------------
12405 REM
12410 LR = ROW
12415 POINTER = POINTER - 1
12417 COLOR 15,1
12420 FOR LOOP = POINTER TO POINTER + 9
12425 LOCATE LR,12: PRINT " ";MID$(CALFILE$(LOOP),11,8);" "
12430 LR = LR + 2
12435 NEXT LOOP
12440 RETURN
12500 REM
12501 REM ----- Subroutine to scroll down one place ----- 
12503 REM - ------------------------------------------
12505 REM
12510 LR = ROW
12520 POINTER = POINTER + 1
12525 COLOR 15,1
12530 FOR LOOP = POINTER TO POINTER-9 STEP -1
12535 LOCATE LR,12: PRINT " ";MID$(CALFILE$(LOOP),11,8);"
12540 LR = LR - 2
12545 NEXT LOOP
12550 RETURN
13000 REM
13005 REM ----- Subroutine to write name of calfile in a specified colour ----- 
13015 REM - ------------------------------------------
13020 REM
13030 REM
13040 REM ----- Write cal file in large menu box.....
13050 REM
13060 COLOR COL,1
13070 LOCATE ROW,12: PRINT " ";MID$(CALFILE$(POINTER),11,8);" 
13080 REM
13090 REM ----- Write cal file in small menu box.....
13100 REM
13110 LOCATE ROW2,45: PRINT " ";MID$(CALFILE$(POINTER),11,8);" 
13120 RETURN
14000 REM
14005 REM ----- Subroutine to write initial menu option in specified colour ----- 
14015 REM - ------------------------------------------
14020 REM
14040 COLOR COL,1
14060 LOCATE ROW,30: PRINT OPTN$(POINTER)
14070 RETURN
20000 REM
20005 REM ----- Subroutine to pick out coefficients from the Cal file ----- 
20015 REM - ------------------------------------------
20020 REM
20030 REM
20035 REM ----- Ignore all records until coefficient record is encountered.....
20040 REM
20050 While (Left$(RECORDS, 4)) <> "Term"
20060 Input #FILENUM, RECORDS
20070 Wend
20080 Rem
20085 Rem ===== Put coefficients into array.....
20090 Rem
20100 For N = 0 To COEFTOT - 1
20110 Input #FILENUM, RECORDS
20120 C(N) = Val(Right$(RECORD$, Len(RECORD$) - 2))
20130 Next N
20140 Return
21000 Rem
21005 Rem ===== Subroutine to calculate Y-Cals =====
21010 Rem
21020 Rem
21030 For N = 1 To 10
21040 Read HEX(N)
21050 YCAL1(N) = COEF1(0) + COEF1(1) * HEX(N) + COEF1(2) * HEX(N)^2 + COEF1(3) * HEX(N)^3 + COEF1(4) * HEX(N)^4
21060 YCAL2(N) = COEF2(0) + COEF2(1) * HEX(N) + COEF2(2) * HEX(N)^2 + COEF2(3) * HEX(N)^3 + COEF2(4) * HEX(N)^4
21070 Next N
21080 Return
22000 Rem
22002 Rem ===== Subroutine to draw menu and allow user to select options =====
22005 Rem
22010 Rem
22020 Rem ===== Draw menu screen.....
22030 Rem
22040 Key Off
22050 Screen 9
22055 Cls
22057 Locate 5, 29: Print "CALIBRATION CERTIFICATE"
22060 Color 12, 1
22070 Line (210, 80)-(415, 268), 15, B
22080 Paint (230, 100), 2, 15
22090 Locate 23, 16: Print "Choose the sensor type required for calibration"
22095 Color 15, 1
22097 Restore 22150
22100 For ROM = 8 To 18 Step 2
22102 Pointer = ROM/2 - 3
22105 Read OPTN$(POINTER)
22110 Locate ROM, 30: Print OPTN$(POINTER)
22120 Next ROM
22130 Color 12, 1: Locate 8, 30: Print OPTNS(1)
22140 Color 12, 1: Locate 8, 30: Print OPTNS(1)
22150 Data " Wind Speed "
22152 Data " Psychrometer "
22154 Data " SST "
22156 Data " Air Pressure "
22158 Data " Air Temperature "
22160 Data " Quit "
22170 Key$ = ""
REM Define up/down keys.....
DN$  =  CHR$(0)  +  CHR$(80)
UP$  =  CHR$(0)  +  CHR$(72)

REM  =====  Scan keyboard for RETURN and UP DOWN keys.....
ROM - 8:  POINTER  =  1
WHILE  KY$  <>  CHR$(13)
   KY$  =  INKEY$
   IF  (KY$  =  DNS AND  ROW < 18)  THEN  COL  =  15:  GOSUB  14000:  ROW  =  ROW  +  2:  POINTER  =  POINTER  +  1:  COL  =  12:  GOSUB  14000
   IF  (KY$  =  UP$ AND  ROW > 8)  THEN  COL  =  15:  GOSUB  14000:  ROW  =  ROW  -  2:  POINTER  =  POINTER  -  1:  COL  =  12:  GOSUB  14000
WEND

REM  =====  Determine which sensor has been chosen.....
IF  POINTER  =  1  THEN  ID$  =  "ws"  ELSE  IF  POINTER  =  2  THEN  ID$  =  "tt"  ELSE  IF  POINTER  =  3  THEN  ID$  =  "sp"  ELSE  IF  POINTER  =  4  THEN  ID$  =  "ap"  ELSE  IF  POINTER  =  5  THEN  ID$  =  "at"  ELSE  GOSUB  36000

REM  =====  Obtain serial number from user.....
FOR  N  =  1  TO  50
   LINE  (150,200-N)-(485,200+N),6,BF
NEXT  N
COLOR  12,1
LOCATE  13,22:  PRINT  "Enter the serial number of the sensor"
COLOR  15,1:  LOCATE  15,28:  PRINT  "Enter the serial number of the sensor"
INPUT  SERIAL$  RETURN

REM  Subroutine to sort Cal files in ascending order of date
FOR  N  =  1  TO  TOTAL
   VCODE  =  ASC(MID$(CALFILE$(N),18,1))
   IF  NOT  (VCODE  >  97  AND  VCODE  <  122)  OR  (VCODE  >=  65  AND  VCODE  <=  90)  THEN  GOSUB  3500
   ELSE  JDAY$  =  MID$(CALFILE$(N),13,5):  YEAR$  =  MID$(CALFILE$(N),16,2)
   JDAY$  =  YEAR$  +  LEFT$(JDAY$,3)
   JDAYFILE(N)  =  VAL(JDAY$)
NEXT  N

REM  =====  Set up loop to keep sorting until no more swaps have been made.
ISSWAP  =  1

30140 WHILE ISSWAP <> 0
30150      ISSWAP = 0
30160    FOR POSN = 1 TO TOTAL-1
30170      IF JDAYFILE(POSN+1) < JDAYFILE(POSN) THEN GOSUB 35200
30180    NEXT POSN
30190  WEND
30999  RETURN

35000  REM
35005  REM --------------------------------------------------
35010  REM ----- Subroutine to convert date to Julian day ----- 
35015  REM --------------------------------------------------
35020  REM
35030  DAY = VAL(MID$(CALFILE$(N),13,2))
35040  MONTH = VAL(MID$(CALFILE$(N),15,2))
35050  YEARS = MID$(CALFILE$(N),17,2)
35060  JDAY$ = STR$(VAL(MID$("000031059090120151181212243273304334",((MONTH-1)*3)+1,3))+DAY)
35070  JDAY$ = YEARS + RIGHTS(JDAY$,3)
35100  RETURN
35200  REM
35205  REM --------------------------------------------------
35210  REM ----- Subroutine to swap the position of the calfile names ----- 
35215  REM --------------------------------------------------
35220  REM
35230  REM
35235  REM ----- Swap the JDys.....
35240  REM
35250  SWAP JDAYFILE(POSN),JDAYFILE(POSN+1)
35260  SWAP CALFILE$(POSN),CALFILE$(POSN+1)
35270  ISSWAP = 1
35280  RETURN
36000  REM
36002  REM --------------------------------------------------
36005  REM ----- Subroutine to quit program ----- 
36007  REM --------------------------------------------------
36010  REM
36020  IF PROG$ <> "" THEN CHAIN PROG$
36030  SYSTEM
36040  END
1000 REM *****************************************************************
1010 REM ******************  SENPRT.BAS  by RWP 19/11/90  ************************
1020 REM **********************************************************************
1040 REM
1041 REM  =====  Set error trap....
1042 REM
1043 ON ERROR GOTO 40000
1044 REM
1045 REM =====  Call subroutine to set up Ctrl Esc key trap....
1046 REM
1047 GOSUB 14000
1048 KEY OFF
1049 SCREEN 9
1056 REM
1057 REM =====  Set up user menu....
1058 REM
1059 COLOR 7,1: CLS
1060 REM
1061 REM ===== Draw up menu box....
1065 REM
1067 LINE (5,5)-(635,345),B
1069 LINE (180,53)-(480,266),2,BF
1070 LINE (185,58)-(475,261),64,B
1071 LOCATE 6,26:PRINT" Air Temperature  " ;CHR$(27);" "
1072 LOCATE 8,26:PRINT" Sea Temperature  "
1073 LOCATE 10,26:PRINT" Wind Speed  "
1074 LOCATE 12,26:PRINT" Wind Direction  "
1075 LOCATE 14,26:PRINT" Radiometers  "
1076 LOCATE 16,26:PRINT" Others  "
1077 LOCATE 18,26:PRINT" Return to Allocation Menu  "
1078 LOCATE 22,8:PRINT" Move arrow to required sensor type and press <RETURN> when finished"
1079 LOCATE 23,26 : PRINT"Move around using cursor keys"
1080 COLOR 3:LOCATE 3,27:PRINT "SELECT SENSOR TYPE for PRINTER"
1085 REM
1086 REM =====  Define the cursor keys.....
1087 REM
1088 DN$'=CHR$(0) +CHR$(80):UP$=CHR$(0) +CHR$(72)
1089 X=6:Y=59:KY$=" "
1090 REM
1091 REM ===== Set up pointer box in menu.....
1092 REM
1093 COLOR 7,1
1095 Y1 = 69
1096 GOSUB 20000
1100 LOCATE X,Y
1102 REM
1105 REM ===== Reading input from keyboard....
1107 REM
1108 WHILE KY$<>CHR$(13)
1110 KY$=INKEYS
1112 IF (KY$=DN$) AND ((X>=6)AND(X<=16)) THEN LOCATE X,Y-3:PRINT" ";X=X+2:LOCATE X,Y-2:PRINT CHR$(27); : GOSUB 20000
1113 IF (KY$=UP$) AND ((X>=8)AND(X<=18)) THEN LOCATE X,Y-3:PRINT" ";X=X-2:LOCATE X,Y-2:PRINT CHR$(27); : GOSUB 20000
1290 WEND
1291 IF X=6 THEN METNO=100:GOSUB 30000
1292 IF X=8 THEN METNO=200:GOSUB 30000
1293 IF X=10 THEN METNO=300:GOSUB 30000
1294 IF X=12 THEN METNO=400:GOSUB 30000
1295 IF X=14 THEN METNO=500:GOSUB 30000
1296 IF X=16 THEN METNO=1:GOSUB 30000
1297 IF X=18 THEN RUN "SEND"
1299 GOTO 1059
1300 END
14000 REM
14002 REM -------------------------------
14005 REM ----- Subroutine to set up key trap for Ctrl Esc ----- 
14007 REM -------------------------------
14010 REM 
14014 REM 
14016 REM ----- Define keys to include all possible combinations of CAPS LOCK
14017 REM  and NUM LOCK.....
14018 REM 
14020 KEY 15,CHR$(&H64)+CHR$(1)
14030 KEY 16,CHR$(&H44)+CHR$(1)
14040 KEY 17,CHR$(&H24)+CHR$(1)
14050 KEY 18,CHR$(&H4)+CHR$(1)
14060 REM 
14064 REM ==== Direct all escape key combinations to line 65300 .....
14066 REM 
14070 ON KEY (15) GOSUB 65300
14080 ON KEY (16) GOSUB 65300
14090 ON KEY (17) GOSUB 65300
14100 ON KEY (18) GOSUB 65300
14110 REM 
14115 REM ==== Turn all escape key combinations on.....
14117 REM 
14120 KEY (15) ON
14130 KEY (16) ON
14140 KEY (17) ON
14150 KEY (18) ON
14200 RETURN
20000 REM
20010 REM -------------------------------
20020 REM ----- Subroutine for drawing & flashing pointer box in menu ----- 
20030 REM -------------------------------
20040 REM 
20050 REM ==== First overwrite old box.....
20060 REM 
20070 LINE (199,Y1)-(464,Y1+15),2,B
20100 REM ==== Now draw box and flash at new position.....
20110 REM 
20120 IF (KY$ = DNS) THEN Y1 = Y1 + 28 ELSE IF (KY$ = UP$) THEN Y1 = Y1 - 28
20140 FOR COL = 9 TO 15
20150 LINE (199,Y1)-(464,Y1+15),COL,B
20160 NEXT COL
20170 LINE (199,Y1)-(464,Y1+15),12,B
20500 RETURN
0000 REM
0010 OPEN "R", #1, "SENSORS.DAT", 9
0020 FIELD #1, 2 AS T$, 6 AS S$, 1 AS L$
0030 IF METNO=1 THEN LPRINT: LPRINT 
0040 IF METNO=100 THEN LPRINT: LPRINT 
0050 IF METNO=200 THEN LPRINT: LPRINT 
0060 IF METNO=300 THEN LPRINT: LPRINT 
0070 IF METNO=400 THEN LPRINT: LPRINT 
0080 IF METNO=500 THEN LPRINT: LPRINT 
0090 IF METNO=1 THEN CT=98 ELSE CT=99 
0100 LPRINT: LPRINT 
0105 FOR I=0 TO CT 
0106 GET#1, METNO+I 
0107 CL0SE#1 
0108 GOTO 1059 
0110 REM 
0120 REM Error checking subroutines 
0121 IF ERR = 5 THEN GOSUB 40100 
0122 IF ERR = 52 THEN GOSUB 40200 
0123 IF ERR = 71 THEN GOSUB 40300 
0124 IF ERR = 72 THEN GOSUB 40400 
0125 IF (ERR = 5 OR ERR = 71) THEN GOSUB 40500: RESUME 65300 
0126 IF ERR = 76 THEN GOSUB 40600: RESUME 65300 
0127 STOP 
0128 REM 
0129 DATE "$DATE$" 
0130 status" 
0131 Please try again 
0132 Invalid filename used 
0133 Please try again 
0134 Please try again 
0135 Please try again 
0136 Please try again 
0137 Please try again 
0138 Please try again 
0139 Please try again 
0140 Please try again 
0141 Please try again 
0142 Please try again 
0143 Please try again 
0144 Please try again 
0145 Please try again 
0146 Please try again 
0147 Please try again 
0148 Please try again 
0149 Please try again 
0150 Please try again 
0151 Please try again 
0152 Please try again 
0153 Please try again 
0154 Please try again 
0155 Please try again 
0156 Please try again 
0157 Please try again 
0158 Please try again 
0159 Please try again 
0160 Please try again 
0161 Please try again 
0162 Please try again 
0163 Please try again 
0164 Please try again 
0165 Please try again 
0166 Please try again 
0167 Please try again 
0168 Please try again 
0169 Please try again 
0170 Please try again 
0171 Please try again 
0172 Please try again 
0173 Please try again 
0174 Please try again 
0175 Please try again 
0176 Please try again 
0177 Please try again 
0178 Please try again 
0179 Please try again 
0180 Please try again 
0181 Please try again 
0182 Please try again 
0183 Please try again 
0184 Please try again 
0185 Please try again 
0186 Please try again 
0187 Please try again 
0188 Please try again 
0189 Please try again 
0190 Please try again 
0191 Please try again 
0192 Please try again 
0193 Please try again 
0194 Please try again 
0195 Please try again 
0196 Please try again 
0197 Please try again 
0198 Please try again 
0199 Please try again 
0200 Please try again 
0201 Please try again 
0202 Please try again 
0203 Please try again 
0204 Please try again 
0205 Please try again 
0206 Please try again 
0207 Please try again 
0208 Please try again 
0209 Please try again 
0210 Please try again 
0211 Please try again 
0212 Please try again 
0213 Please try again 
0214 Please try again 
0215 Please try again 
0216 Please try again 
0217 Please try again 
0218 Please try again 
0219 Please try again 
0220 Please try again 
0221 Please try again 
0222 Please try again 
0223 Please try again 
0224 Please try again 
0225 Please try again 
0226 Please try again 
0227 Please try again 
0228 Please try again 
0229 Please try again 
0230 Please try again 
0231 Please try again 
0232 Please try again 
0233 Please try again 
0234 Please try again 
0235 Please try again 
0236 Please try again 
0237 Please try again 
0238 Please try again 
0239 Please try again 
0240 Please try again 
0241 Please try again 
0242 Please try again 
0243 Please try again 
0244 Please try again 
0245 Please try again 
0246 Please try again 
0247 Please try again 
0248 Please try again 
0249 Please try again 
0250 Please try again 
0251 Please try again 
0252 Please try again 
0253 Please try again 
0254 Please try again 
0255 Please try again 
0256 Please try again 
0257 Please try again 
0258 Please try again 
0259 Please try again 
0260 Please try again 
0261 Please try again 
0262 Please try again 
0263 Please try again 
0264 Please try again 
0265 Please try again 
0266 Please try again 
0267 Please try again 
0268 Please try again 
0269 Please try again 
0270 Please try again 
0271 Please try again 
0272 Please try again 
0273 Please try again 
0274 Please try again 
0275 Please try again 
0276 Please try again 
0277 Please try again 
0278 Please try again 
0279 Please try again 
0280 Please try again 
0281 Please try again 
0282 Please try again 
0283 Please try again 
0284 Please try again 
0285 Please try again 
0286 Please try again 
0287 Please try again 
0288 Please try again 
0289 Please try again 
0290 Please try again 
0291 Please try again 
0292 Please try again 
0293 Please try again 
0294 Please try again 
0295 Please try again 
0296 Please try again 
0297 Please try again 
0298 Please try again 
0299 Please try again 
0300 Please try again 
0301 Please try again 
0302 Please try again 
0303 Please try again 
0304 Please try again 
0305 Please try again 
0306 Please try again 
0307 Please try again 
0308 Please try again 
0309 Please try again 
0310 Please try again 
0311 Please try again 
0312 Please try again 
0313 Please try again 
0314 Please try again 
0315 Please try again
40320 COLOR 12,1: LOCATE 15,15: PRINT " Drive "; DRIVES; " not ready "
40330 COLOR 15,1: LOCATE 16,13: PRINT " Press return when ready "
40340 BEEP
40350 ANSWER$ = ""
40360 WHILE (ANSWERS <> CHR$(13))
40370 ANSWERS = INKEY$  
40380 WEND
40390 RETURN
40400 REM
40405 REM ===== Disk media error.....
40410 REM
40420 LINE (50,196)-(330,250),3,BF
40430 COLOR 12,1: LOCATE 15,11: PRINT " Error with hardware or disk "
40440 COLOR 15,1: LOCATE 16,17: PRINT " Program halted "
40450 RETURN
40500 REM
40502 REM ===== Illegal function call.....
40505 REM
40510 COLOR 12,8: LOCATE 17,33: PRINT " Insignificant data"
40520 LOCATE 18,26: PRINT "Could not calculate coefficients"
40530 COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
40540 WHILE (INKEY$ = "")
40550 WEND
40555 CLS
40560 REM
40562 REM ===== Exit program and return to operating system.....
40565 REM
40590 RETURN
40600 REM
40602 REM ===== Path not found.....
40605 REM
40610 CLS
40620 COLOR 12,1:LOCATE 17,24: PRINT "Could not find database on drive "; DRIVES
40630 COLOR 15,8: LOCATE 20,32: PRINT "---Press Any Key---"
40640 WHILE (INKEY$ = "")
40650 WEND
40655 CLS
40660 REM
40662 REM ===== Exit program and return to operating system.....
40665 REM
40690 RETURN
50000 REM
50100 REM DRAW GREEN BOX
50105 REM CLS
50110 COLOR 15,1
50120 FOR N=1 TO 150 STEP 2
50130 LINE (BB%N,160-N)-(BB%N,160+N),2,BF
50140 NEXT N
50200 RETURN
65300 REM
65302 REM --------------------------------------------
65305 REM ===== Subroutine for execution when Ctrl Esc has been pressed -----
65307 REM --------------------------------------------
65310 REM
65320 CLS
65330 SCREEN 0,0,0
65340 LOCATE 14,9: PRINT "Program Execution Halted.... Returning to MAINMENU"
65345 LOCATE 16,29: PRINT "---Press Any Key---"
65350 WHILE (INKEYS = ")
65360 WEND
65370 RUN "MAINMENU"