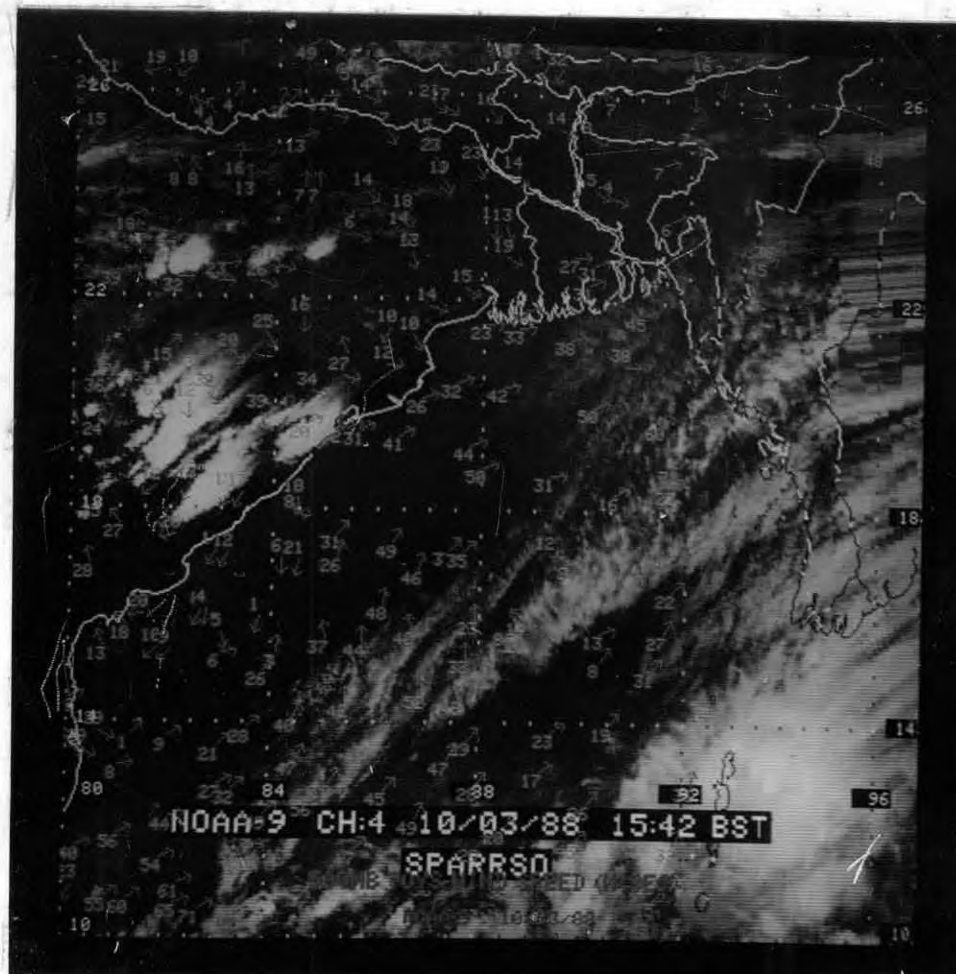
COMPUTER IMAGING OF METEOROLOGICAL PARAMETERS DERIVED FROM NOAA SATELLITE AND GROUND STATIONS

A dissertation submitted to Electrical and Electronic Engineering Department BUET for the degree of Master of Science in Electrical Engineering.

Abu Raihan Azimul Hoque

April 1988



Frontispiece: Display of wind direction and velocity with the help of the developed software on cloud picture received from NOAA satellite.

Declaration

This is to certify that this work has been done by me and this has not been submitted elsewhere for the award of any degree or diploma or for publication.

A. R. Azimul Hoque

(A. R. Azimul Hoque)

Accepted as satisfactory for partial fulfilment of the requirements for the degree of Master of Science in Engineering (Electrical and Electronic Engineering).

Board of Examiners:

1. Saiful Islam, M.Sc.Eng.,Ph.D.,FIEB,MIEE — Chairman and Supervisor — Saiful Islam 2.5.88
   Associate Professor,
   EEE Department,
   Bangladesh University of
   Engineering and Technology,
   Dhaka 1000.

2. Prof. Md. Mujibur Rahman, Ph.D.,FIEB — Member (Ex-officio)
   Head,
   EEE Department,
   Bangladesh University of
   Engineering and Technology,
   Dhaka 1000.

3. Kazi Mohiuddin Ahmed, M.Sc.Engg.,Ph.D.,MIEB — Member — 2/5/88
   Associate Professor,
   EEE Department,
   Bangladesh University of
   Engineering and Technology,
   Dhaka 1000

4. Md. Rezwan Khan, M.Sc., Ph.D. — Member — RKhan 2.5.88
   Assistant Professor,
   EEE Department,
   Bangladesh University of
   Engineering and Technology,
   Dhaka 1000

5. Abdul Musawwir Chowdhury Ph.D. — Member (External) — AMChowdhury 2/5/88
   Director (Research),
   Space Research and Remote
   Sensing Organization,
   Mohakash Biggyan Bhavan, Agargaon,
   Sher-e-Bangla Nagar,
   Dhaka 1207.

Acknowledgements

The author expresses his deep gratitude to his supervisor Dr. Saiful Islam for his constant guidance and valuable suggestions in this work.

The author acknowledges the support and encouragement received from Dr. Mujibur Rahman, Head of EEE Department BUET.

The author acknowledges his indebtness to Dr. A. M. Chowdhury of SPARRSO for providing helpful information related to this work and for his encouragement.

Thanks are given to Dr. F. A. Khan ex-Chairman and Dr. M. A. H. Pramanik, Director (T) of SPARRSO for kindly allowing to use the facilities at SPARRSO. The author acknowledges the suggestions, support and cooperation he received from his fellow colleagues at SPARRSO.

The author acknowledges the cooperation received from the Director of Bangladesh Meteorological Department. Thanks are due to Mr. Fakhrul Quyyum, Mr. Mehtabuddin and friends of BMD for their help in providing the necessary information and data.

Thanks are due to Gene Shaeffer and Robert Mahoney of SSAI, USA for providing valuable information and necessary data related to this work.

Contents

Page

Abstract

Computer has been found to be a convenient and valuable tool for receiving, storing, processing and displaying various meteorological information sent by the weather satellites. The meteorologists and the weather forecasters find the displays obtained using computer techniques very much helpful for easy understanding of weather conditions as well as for fast reading of required weather parameters. While significant development has been made in the area of computer softwares for processing and displaying of meteorological parameters throughout the world, very little work has so far been done in Bangladesh. In this work three new software packages have been developed which are expected to find application at Bangladesh Space Research and Remote Sensing Organization (SPARRSO).

The first software has been developed to decode, process and store Global Telecommunication System (GTS) meteorological data in the VAX 11/750 computers of SPARRSO. The decoding processes used by the GTS network are the basis of the developed computer programs. This software prepares GTS data for further processing in order to obtain displays and maps. The newly developed software replaces previously used manual methods of processing coded data.

The second software has been developed for processing the Geo-Potential Height (GPH) values and plotting GPH contour maps using the incoming coded form GPH values received from GTS and Tiros Operational Vertical Sounder (TOVS) (from National Oceanic and Atmospheric Administration (NOAA) satellite) sources. Producing GPH contour maps using computer techniques is new at SPARRSO and indeed in Bangladesh. In this computer method equispaced imaginary lines are drawn on the display area in both x and y direction to obtain grid points. The GPH values at the grid points are derived using a $1/r^2$ method and the available GPH values. The desired GPH values are thus available at equal intervals in both x and y directions which permits drawing of contour lines without any difficulty.

The third software has been developed to display the wind parameter by using TOVS and GTS data. This software utilizes the various hardware and software features of the IIS image processor of SPARRSO. In the display of the wind parameter produced by this software the wind data is displayed with directional arrows and numerical values of velocity magnitudes.

Computer programmes and standard operating procedures (SOPs) for the three software packages are presented.

Chapter 1

Introduction

1.1 Introduction

Weather forecasting is very much needed since weather influences our day to day activities significantly. Scientists and technologists from early stages of civilization have been trying to find ways out for early prediction of weather. As a result in the field of meteorology gradual advancements have been taking place for early prediction of weather. In recent years, with the advent of modern computer technology and the advancement of digital image processing, the previously used manual methods of mapping, plotting, and preparing weather charts are being replaced by new computer techniques. Due to the application of modern techniques weather prediction has improved to a large extent. In fact now-a-days in advanced countries, using satellites and appropriate weather installations, weather forecasting is done for nearly a week with good accuracy.

Bangladesh can receive weather and earth resources information from Land satellite (LANDSAT) [36], Geostationary Meteorological Satellite (GMS) [26], National Oceanic and Atmospheric Administration (NOAA) [25] [24] and Satellite Probatoire d'Observation de la Terre (SPOT) [30] [31] satellites. Of these NOAA and GMS are weather satellites whilst LANDSAT and SPOT are earth resources satellites. Bangladesh Space Research and Remote Sensing Organization (SPARRSO) receives from all these four satellites but can process data received directly from NOAA and GMS satellites only. Data received from LANDSAT and SPOT satellites can not be straightaway processed at SPARRSO because part of the preprocessing system for this is yet to be installed. However, Landsat and SPOT satellite data after certain interval are currently being imported from nearby countries in Computer Compatible Tapes (CCT) for processing at SPARRSO. Our present interest is mainly on the data sent by weather satellites. This work is specially concerned with the data received directly (on a real time basis) from NOAA satellite.

There is another source of weather data which comes through GTS (Global telecommunication system) channels at Bangladsh Meteorological Department (BMD) on teleprinters. These data received in the form of teleprinter printouts are brought occationally to SPARRSO and manually entered into the hard disk of the VAX-11/750 minicomputer in the form of data files. This work is also concerned with the processing of these data received from the GTS channels through the teletype printers.

1.2 Currently used display system of weather data at SPARRSO and BMD

SPARRSO receives signals and images from NOAA [25] and GMS [26] meteorological satellites everyday. The received signals and images provide significant information for Weather Prediction. The NOAA satellite receives 'Tiros Information Processor' (TIP) [13] data stream and 'Advanced Very High Resolution Radiometer' (AVHRR) [23] imagery. The TIP data stream contains 'Tiros Operational Vertical Sounder' (TOVS) [13] data, 'Data Collection System' (DCS) [13] data and 'spacecraft and instrument telemetry' [13] informaton. TIP data and AVHRR video data are then fed to 'Manipulated Information Rate Processor' (MIRP) [13]. Data from MIRP are then transmitted as 'High Resolution Picture Transmission' (HRPT) [23] format and 'Automatic Picture Transmission' (APT) [23] analog format. Using proper ground receiving system HRPT data are received in the frequency range of

1698.0 MHz. to 1707.0 MHz. [13]. The APT data are received in the frequency range of 137.50 MHz.to 137.62 MHz. [13]. This flow of data in the system discussed above is presented in figure 1.1.

The TOVS data contains vertical profiles of atmospheric temperatures and water vapour information. A number of weather parameters can be derived from these TOVS data by using appropriate computer software. At SPARRSO processing of TOVS data is being done using TOVS software package [14] obtained from 'Cooperative Institute for Meteorological Satellite Studies (CIMSS)' Wisconsin University Wisconsin USA. With the existing software packages SPARRSO can display data values of various weather parameters such as geopotential height (GPH), air temperature, dew point temperature, total precipitable water, stability index etc.

Bangladesh Meteorological Department (BMD) receives meteorological data transmitted from various land stations in and around Bangladesh through the Global Telecommunication System (GTS) channels [4]. At present these data are manually plotted on a previously prepared 'meteorological chart'. On the same chart pressure level GPH contour lines , wind (speed and direction) and temperature of different stations are indicated manually. These manual drawing work is time consuming as well as labourious.

1.3 Difficulties encountered with the existing display

The existing software facilities at SPARRSO do neither permit one to process nor to obtain a display of the GTS data. The wind data, an important parameter, also can not be displayed. Besides these, with the existing display facilities the GPH data contour map can not be readily made available out of the computer.

Display of GTS data, improvement on wind data display, and

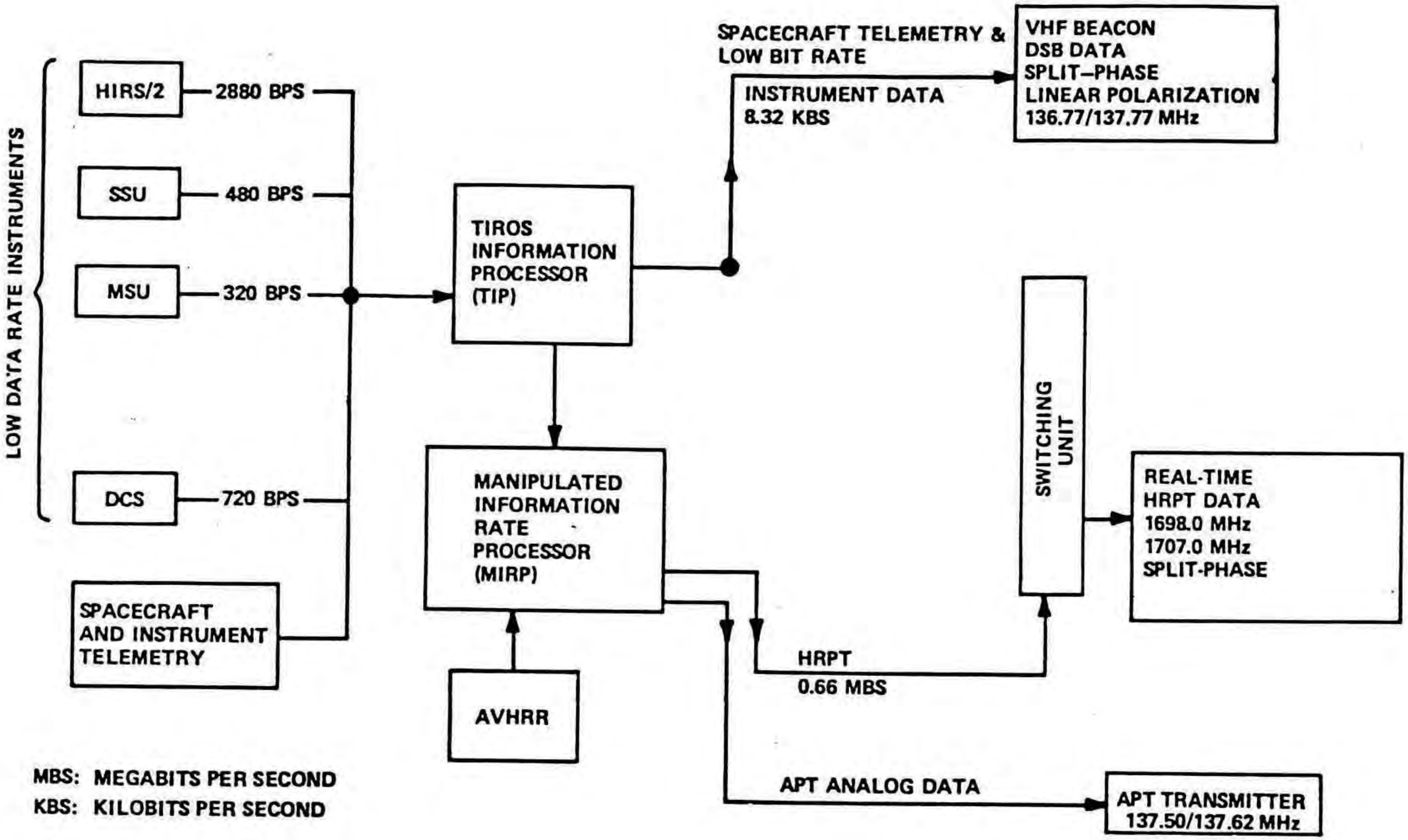


Figure 1.1 Real time system data flow for NOAA satellite.

contouring of the GPH data are the three jobs which need development at SPARRSO. It is thought that successful development on these jobs would make significant addition to the existing displays of SPARRSO and make the total display much more valuable to the weather forecasters.

1.4 Motivation of this work

The ACEMS (Agro-Climatic and Environmental Monitoring System) personnels at SPARRSO in their weekly meeting discuss developments as well as the difficulties and further needs for extension of the existing systems and softwares. As part of system management and software development job the author attends these meetings. The difficulties of the existing display systems of meteorological parameters were pointed out in some of the meetings during the year 1986. The motivation for this work to develop new softwares occurred from these discussions.

1.5 Objectives of the work

The aim of this work is to display some of the meteororlogical parameters in computer imaging and graphical formats. A number of weather parameters need display for easy interpretation. Of these weather parameters geopotential height and wind have been taken in this work for display, because these are very widely used parameters for weather prediction.

The first objective is to develop a technique of formatting the radiosonde data [4] obtained through GTS channels as well as computer processing and monitor display technique of the same data.The second objective is to display the geopotential height in contours. The third objective of this work is to display the wind data in magnitude as well as in direction format with proper geometric boundaries.

1.6 This dissertation

A brief description of the Computer Hardware and other peripheral devices used for the purpose, is presented in chapter two.

Chapter three describes the available sources of data, which comes from the TOVS instruments of NOAA satellite as well as from the Global Telecommunication Systems (GTS).

In chapter four, decoding of GTS data, its processing and storing into a binary file is described.

Contouring of geopotential height (GPH) data is described in chapter five. The description also includes the creation of grid data file using a computer program and the description of the computer programs used for contouring of the GPH data.

Chapter six describes the technique involved in displaying the wind data. The program used for the display of wind data is also described. Some IIS hardware features that deserves merit in display of the wind data are briefly described.

'Standard Operating Procedure (SOP)' for the jobs that can be done using the softwares created in this work are presented in chapter seven.

Discussions are presented in chapter eight. The discussion includes the use and application of the results, accuracy that is obtained with respect to the real data and the achievements that are made. The difficlties encountered during the work are also presented. Suggestions for future improvements are also incorporated in this chapter.

Appendices, list of abbreviations and references are enclosed at the end of this dissertation.

Chapter 2

Hardware environment

2.1 Introduction

The two satellites NOAA and GMS collect meteorological information with the help of their sensors. As part of its job SPARRSO receive the meteorological information sent by these two satellites. SPARRSO's Agro-Climatic and Environmental Monitoring System (ACEMS) has a computer system at Agargaon consisting of two VAX-11/750 minicomputers. This computer system receive meteorological information from NOAA and GMS satellites through respective ground receiving networks in real time on a daily basis. The incoming signal through the receiving systems comes to an interface called Laboratory Peripheral Accelerator (LPA) [10], which converts the received signals into digital data. After proper formatting these data are stored into the disk media for processing. A block diagram of this receiving and processing system is presented in figure 2.1

It may be mentioned here that besides the information sent by NOAA and GMS satellites, the computer system also handles, through computer compatible tapes, the information sent by Landsat and Spot satellites. However, this thesis will deal only with the information received from NOAA satellite.

In this chapter SPARRSO's receiving system for NOAA satellite data is described. SPARRSO's computer facilities and the IIS image processor used for this research work are also described.

Section 2.2 describes the NOAA satellite data reception system at SPARRSO that is used on regular basis to receive NOAA passes.

Section 2.3 of this chapter describes ACEMS computer system that is now in operation at SPARRSO.

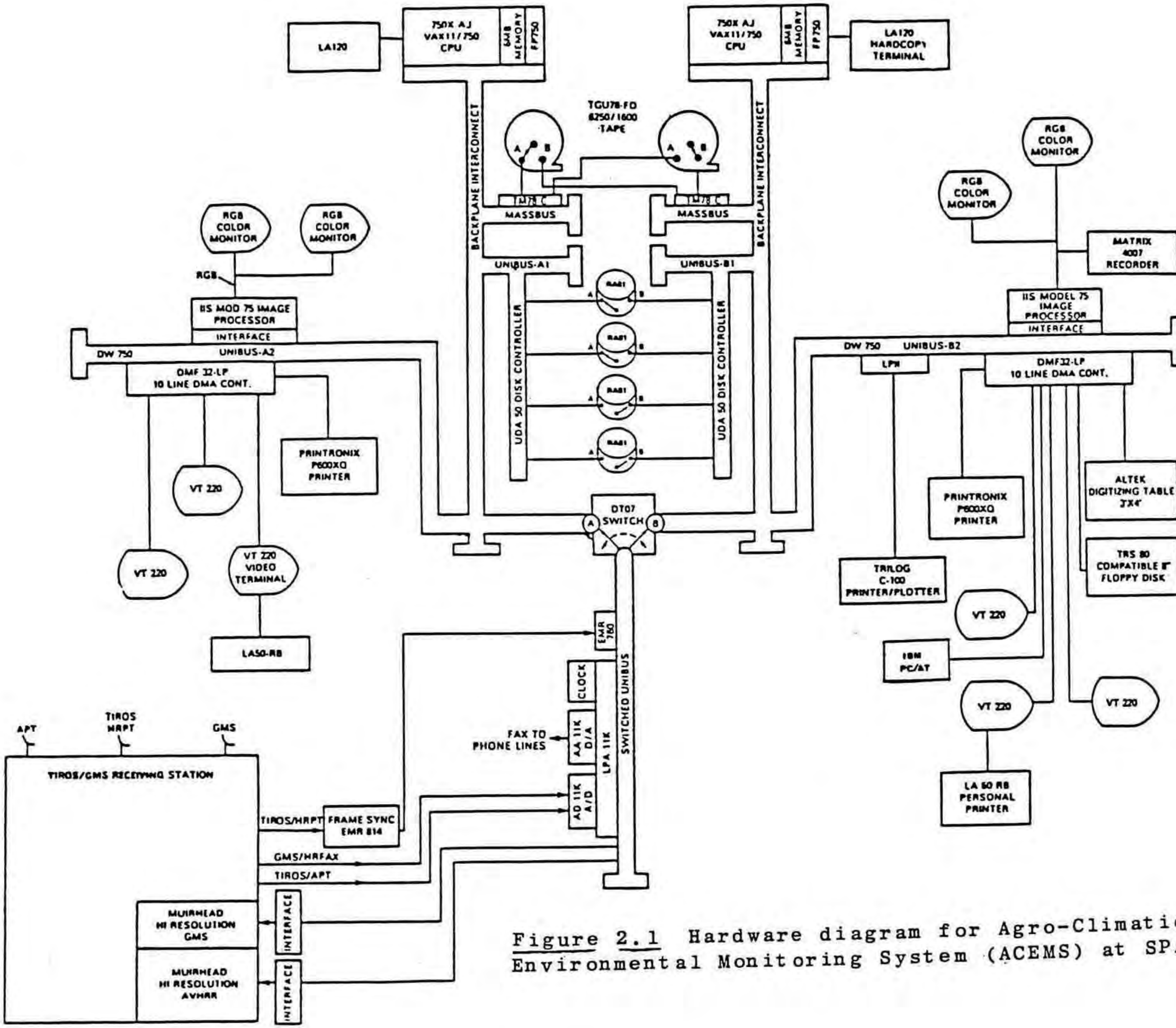


Figure 2.1 Hardware diagram for Agro-Climatic and Environmental Monitoring System (ACEMS) at SPARRSO.

Various IIS image processor features are described in section 2.4

2.2 NOAA satellite data reception system

The ACEMS ground station receives the S-band, digital, bi-phase encoded HRPT image, and TIPs data (containing TOVS and DCS data) using an 8 feet autotrack paraboloid reflector antenna [32].

The 1698 MHz carrier is converted to approximately 342 MHz,using a downconverter. This signal is then demodulated in a Microdyne 1100 AR radio receiver [33] and conditioned in an 'Aydin monitor Systems' bit-synchronizer

The resulting serial bit stream is converted to 10-bit parallel data words in an EMR-814 frame formatter[11]. The formatter detects pseudo-random bit pattern marking the beginning of each scan line and also maintains certain quality checks on the incoming bit stream.

The 10-bit parallel data streams are read by TACQ [8] program and put into the VAX 11/750 through the EMR-760 [12] buffered data channel. The block diagram is illustrated in figure 2.2.

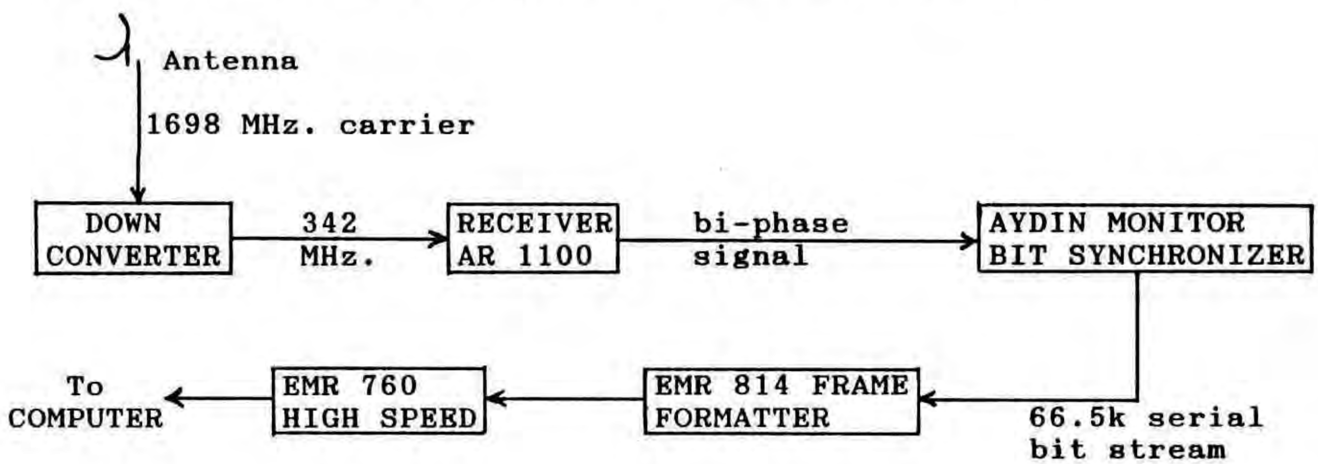


Figure 2.2. block diagram showing receiving and decoding system of NOAA signal at SPARRSO.

2.3 The ACEMS computer system at SPARRSO

Two identical VAX-11/750 mini computers are operating at SPARRSO in two systems. The systems are known in SPARRSO as system 'A' and system 'B'. Each system can independently receive real time satellite data through a swithcing device called DT07 switch. Each system has the following features in common,

(a) A memory capacity of 6 megabytes RAM. Extendable upto 8 megabytes.

(b) A Cache memory of 4 kbytes. This keeps the latest instructions for future reference. It helps in increasing the operating performance by 90 percent than by using the normal memory.

(c) A UDA50 disk controller which can connect upto four disk units with the central processing unit (CPU).

(d) Five Winchester dual-ported fixed disk drive (type RA81). Each unit has a capacity of 456 Megabytes.

(e) A high capacity massbus controller (data transfer rate 2.3 Mbytes/sec) which helps in accessing two dual-ported magnetic tape drives (capacity of 1600/6250 bpi) concurrently.

(f) An IIS image processor to do image processing in real time while displaying the data on an RGB color monitor.

(g) A DMF 32-LP [21] 10 line DMA controller which provides access to a parallel printer, one synchronous port and eight RS232-C [19] ports.

(h) A DZ-11 multiplexer to connect eight asynchronous RS232-C ports.

(i) A k560 Muirhead device for black and white hardcopy output of IIS processed imagery.

Apart from the above features, system A has an Altek digitiging tablet for digitigation of map and other data. A trilog C-100 color printer/plotter is also connected to system B to do color plotting and graphical representation of data. One matrix film processor and recorder is also connected to system B to take pictures from a RGB color monitor.

One IBM-PC/AT microcomputer with 30 Mbytes of hard disk and one Tandy TRS-80/16B microcomputer with 15 Mbytes of hard disk are also present. These two microcomputers work independently as well as communicate with VAX-11/750 using emulators and connencts through RS232-C serial cable.

2.4 The IIS Image Processing system

The IIS Model 575 image processor system is a state-of-the-art processing package that utilizes an advanced hardware architecture to perform most processing in real time. This general purpose image array processor, provides the architecture to implement a wide variety of complex image processing algorithms. Traditional point processing operations are accomplished in real-time (30 frames per second). Iterative or 'recursive spatial operation' such as convolution, filtering and classification of images are accomplished in near real-time (a few frames per second). Complex geometric operations such as rotation, rubber sheet warping and map projections are accomplished in seconds. A block diagram of the IIS hardware diagram is shown in figure 2.3.

Some of the features and capabilities of the Model 575 Image processor [3] can be summarized as :

(i) Up to sixteen 512x512x8-bit Refresh Memories are available on only eight boards through the use of 64k dynamic RAMs

(ii) Pipeline processing channels employed to handle arithmetic functions in parallel for each primary color (red, green, blue)

(iii) 32 Look-up tables in each pipeline (96 total at 256x12-bit) provide four levels of "local area" processing in each color.

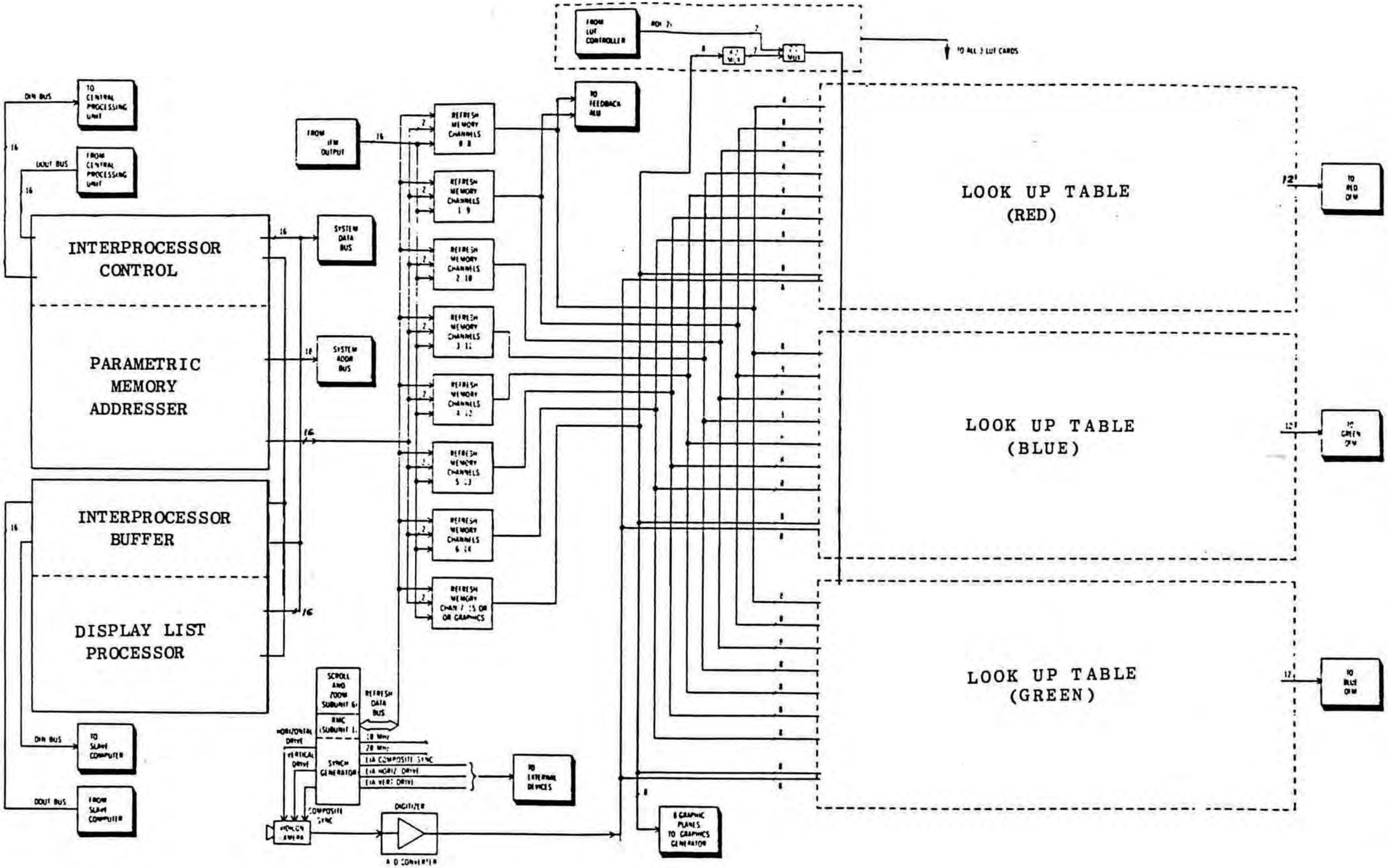


Figure 2.3 Simplified block diagram of IIS image processor.

(iv) 'Input function memory' provides scaling, negation, equalization, etc. of input data.

(v) The videometer hardware computes histograms on raw or processed images.

(vi) Hardware adder arrays allow real-time arithmetic operations (add, subtract, multiply, divide, Karhunen-loeve transforms, etc.) between image bands in each pipelines.

(vii) 'Parametric Memory Addressing' allows access of Refresh Memory horizontally, vertically or randomly, thus simplifying image transposition and geometric operations.

(viii) Display list processor provides for executing sequences of "macro" instructions.

(ix) 'Interactive roam about (real-time pan)all image memory' (up to a 2048x2048x8-bit image) with or without zoom.

(x) Independent X and Y hardware zoom for each refresh memory.

(xi) Independent X and Y hardware scroll for each refresh memory.

(xii) Split screen displays portion of upto four different images simultaneously.

(xiii) Upto sixteen 512x512x1-bit planes of 'graphics overlay memory'.

(xiv) Programmable-shaped cursor (64x64 array) displayed in up to 32k colors.

(xv) Trackball with fifteen function buttons for manual inputs.

2.5 Summary

The systems for receiving the information from the NOAA and GMS satellites are briefly described in block diagrams. An outline of the existing ACEMS computer system is presented. The IIS image processing system is also briefly described with the help of block diagrams.

Chapter 3

Data sources and parameter selection

3.1 Introduction

The meteorological information received in Bangladesh comes from three sources. The first source is the NOAA satellites which send images and meteorological parameters. The second one is the GMS satellite which sends images only. The signals from these two satellites are received at SPARRSO. The third source is the Global Telecommunication System (GTS) of 'World Meteorological Organization' (WMO) which receives the meteorological parameters from various observation stations in various places and countries. These parameters from the Global Telecommunication System are received by Bangladesh Meteorological Department.

A large volume of meteorological information is received at SPARRSO, which are stored in hard disks and tapes for various kinds of processing. Bangladesh Meteorological Department receives information through GTS channels in teletype machines which enables to obtain hardcopies for analyzing and storing.

In this chapter available sources for receiving TOVS and GTS data are described. Among a number of weather parameters available from these data, only two parameters have been selected for this work.

It was mentioned in chapter 2 that the data obtained from NOAA satellite contains TOVS and DCS data. This work is concerned inwith the processing of TOVS data only. In section 3.2 the processing of TOVS data as done at SPARRSO is described.

The arrangement of GTS data as received by BMD is described in section 3.3. Codes used to describe the weather parameters of the GTS data are also described in the same section.

Section 3.4 describes the available weather parameters as received from the TOVS data and the GTS data. Suitable parameters are selected from these weather parameters.

Section 3.5 describes the possible means to ingest the GTS data and the TOVS data into the computer.

3.2 Receiving and processing of TOVS data:

The SPARRSO ground station has the capability of receiving NOAA Satellite TOVS and data collection platforms (DCP) data in the digital form from the Tiros Information Processor [13] data stream. The TOVS data are obtained from 'High resolation Infra Red Sounder' (HIRS) [13] and 'Microwave Sounder Unit' (MSU) [13] sensors as placed in the NOAA Satellites. The TOVS data are extracted from TIP data stream, calibrated and processed by a software package designed and created by the 'Cooperative Institute for Meteorological Satellite Studies (CIMSS)', Wisconsinn University, Wisconsin, USA [9]. After performing all the necessary processing this package creates an output file named TOVRET.DAT which contains the required weather parameters ready for user access. A separate program computes 'geostrophic winds' for good data values of GPH fields by least-square objective analysis [9] and creates another file WINRET.DAT. This file contains wind data at different pressure levels.

3.3 GTS data

Bangladesh Meteorological Department receives upper and lower air radiosonde observation data and the surface data over a period ranging from three to twelve hours from various stations located inside Bangladesh through local communication systems. BMD also receives the same type of data for places outside Bangladesh through international satellite telecommunication channels, known as GTS channels. Data embedded in special codes are transmitted through these GTS channels which are received by the user countries in teletype machines. These data contain several meteorological parameters in different codes depending upon the type of parameters sent.

BMD receives four types of codes every day. The received codes are SYNOPS, SHIPS, TEMPS and PILOTS. Surface level reports (SYNOPS) and sea reports (SHIPS) are received every three hours and these reports contain GPH, air temperature, dew point temperature, weather reports and cloud information [35]. PILOT reports are received every six hours and the information received from PILOT reports are wind data. TEMP code is one of these four types of codes. In this work only the TEMP code is used.

TEMP is the name of the code for an upper level pressure, temperature, humidity and wind reports from land station as recommended by 'World Meteorological Organization' (WMO) [4]. The TEMP code itself has four different parts (i.e. parts A, B, C and D). Part A of TEMP code contains data for the surface and standard isobaric surfaces of 1000, 850, 700, 500, 400, 300, 250, 200, 150 and 100 mb. Part B contains the data upto and including 100 mb level. Part C reports the data for the standard isobaric surfaces of 70, 50, 30, 20 and 10 mb. Part D of the TEMP code reports data above 100 mb level. Only part B of the four parts of TEMP code mentioned above reports the cloud information. In Bangladesh only part A and part C are used. Part A is used in this work. The codes for part A are described below.

TEMP code: Part A

(1) MMNN YYGGD IIiii

where,

MM : the type of station transmitting data. If MM = TT , Land station and MM = UU , Sea station

NN : the Identification letters of the part of the report. There are usually four parts of the TEMP code namely parts A, B, C and d. In this work only the part A will be used. Hence NN is AA.

YY : Day of the month (GMT). When wind speeds are given in knots, 50 is added to YY.

D : Indicator of the last isobaric standard surface for which the wind group is included in part A or part C of TEMP.

for part A

D = 1 indicates upto 100mb or 150mb is used

D = 2 indicates upto 200mb or 250mb is used

D varies from 0 and 1 through 9 where 0 signifies the 1000mb level. For D = / no windgroup for any isobaric level is included in the code.

II : Block number. For Bangladesh it will always be reported as 41.

iii : International station number. For Dhaka it is 923.

(2)	$99P_0P_0P_0$	$T_0T_0T_{a0}D_0D_0$	$d_0d_0f_0f_0f_0$
	$P_1P_1h_1h_1h_1$	$T_1T_1T_{a1}D_1D_1$	$d_1d_1f_1f_1f_1$
	$P_2P_2h_2h_2h_2$	$T_2T_2T_{a2}D_2D_2$	$d_2d_2f_2f_2f_2$
	....	....	....
	$P_nP_nh_nh_nh_n$	$T_nT_nT_{an}D_nD_n$	$d_nd_nf_nf_nf_n$

where,

99 : Indicator for surface data.

P_1P_1 ... P_nP_n : Pressure in terms of millibars, at standard isobaric surfaces. 00 is 1000 mb, 85 is 850 mb and 10 is 100 mb

$P_0P_0P_0$: Pressure surface in whole millibars indicating a horizontal plane whose MSL is the same as that of floor of recording instrument.

$h_1h_1h_1$ ... $h_nh_nh_n$: Geopotential of the standard isobaric surfaces P_1P_1, P_2P_2, ... , P_nP_n in standard geopotential meters. Geopotential of pressure surfaces below 500mb are reported as whole meters. For 500mb and higher surfaces, tens of meters are to be multiplied.

T_0T_0 : Air temperature in whole degrees celsius, at surface.

T_1T_1 ... T_nT_n : Air temperature, in whole degrees celsius, at standard pressure levels.

T_{a0} : Approximate tenths value and sign (plus or minus) of the air temperature at the surface. For T_{a0} = even, value of temperature is in positive and for T_{a0} = odd, value of temperature is in negative.

T_{a1} ... T_{an} : Approximate tenths value and sign (plus or minus) of the air temperature at standard isobaric surfaces or at significant levels. For T_{a1} ... T_{an} = even, value of temperature is in +ve for T_{a1} ... T_{an} = odd, value of temperature is in -ve.

D_0D_0 : Dewpoint depression at the surface

D_1D_1 ... D_nD_n : Dewpoint depression at standard significant levels as specified by $h_1h_1h_1$ ... $h_nh_nh_n$. To obtain the dew point temperature, following rules are to be followed:

i) for DD<50, Dew point temperarute = TT.T - D.D,

ii) for 51<DD<55, Dew point temperature = Missing data and

iii) for DD>55, Dew point temperarute = TT.T - DD

d_0d_0,d_1d_1 ... d_nd_n : True direction (rounded off to the nearest 5 degree), in terms of degrees from which the wind is blowing at specified levels starting with the surface level. When on coding the figure of the wind direction after it has been rounded off to the nearest 5 degrees is multiplied by 100 and then added to the wind speed.e.g. 174 degrees/135 knots is encoded as 175*100 +135 = 17635, 57 degrees/38 knots is encoded as 5500 + 38 = 5538

$f_0f_0f_0, f_1f_1f_1 \ldots f_nf_nf_n$:

Wind speed in meters per second or knots at specified level starting with surface level. The following rules are to be followed:

i) if the date is from 1 to 31 , wind speed is in m/sec

ii) if the date is from 51 to 81 , wind speed is in knots

3.4 Parameter Selection

Several meteorological parameters are to be considered for a well prepared weather prediction. Usually air temperatures, dew point temperature, geopotential height, wind speed and direction, humidity, amount of precipitable water, instability index (total total index, TTI), geopotential thickness etc. are considered for the purpose.

In this work two most important parameters, wind (speed and direction) and geopotential height are considered as they are primarily used in daily weather forecasting.

3.5 Data ingestion into the computer

While satellite data are directly ingested into SPARRSO's computer using receiving ground station, it is yet to formulate a way so that GTS data as obtained by Bangladesh Meteorological Department can be sent down to the computer. In this work manual entry of the data is done. SPARRSO is planning to install a GTS channel, so that data can be ingested directly into the computer, thereby reducing the enormous manual labour in entering these data into the computer as has been faced by this work.

3.6 Summary

The receiving and processing of TOVS data at SPARRSO is described. GTS data as received by BMD is described explaining the codes used to define meteorological parameters. Briefly mentioning the various meteorological parmeters as received from the TOVS data and the GTS data, weather parmaeters are selected for this thesis work.

Chapter 4

Decoding of GTS data

4.1 Introduction

Bangladesh Meteorological Department receives coded GTS data day and night at every three hour interval. These data require decoding prior to use. In this chapter, the procedure of decoding the TEMP code of GTS data using a computer program is presented. The necessary computer program is prepared using Fortran-77 and the decoded data are stored in a file in binary form.

Section 4.2 of this chapter describes the data files used by the program MET.

A description of the decoding program is presented in section 4.3. The flow chart of the program is presented in this section whilst the programs are put in Appendix A.

Section 4.4. describes the operation of the program.

Error conditions encountered during execution of the computer program MET are described in section 4.5

Section 4.6 describes the program environment required to run the program MET.

4.2 Input and output data files for the decoding program

The decoding program takes two data files as input which are MET.ASC and LOC.ASC. The output of this program is stored in another file MET.DAT. These three data files are described below.

(a) MET.ASC : The filename MET indicates that the file contains meteorological parameters. The extension ASC is used to indicate that this file contains ASCII formatted data. The recordsize of the entered data in this file is 80 bytes. A data sample is shown in figure 4.1(a). The first record contains the date and time of data received [4]. The data file is then subdivided into a group of four records. The first record contains the country block and station number. Subsequent three records contain the parameters GPH, air temperature, DPD and wind (speed and direction) for i) the surface, ii) 1000 mb, iii) 850 mb, iv) 700 mb, v) 500 mb, vi) 400 mb, vii) 300 mb, viii) 250 mb, ix) 200 mb, x) 150 mb and xi) 100 mb respectively. The arrangement of these three records are shown in figure 4.1(b).

```
07:10:87 0000                ! DATE & TIME (IN GMT)
51034 60001                  ! COUNTRY BLOCK, STATION NO. & DATE
99002 04859 22503 00163 04859 22503 85471 04761 29511 70976 10969 28015
50549 24763 25530 40709 34564 24532 30904 47763 24534 25024 46363 25028
20173 45765 25531 15364 48566 25030 10631 50167 25016 88283 49563 25535
51076 10001                  ! COUNTRY BLOCK, STATION NO. & DATE
99036 06010 11503 00197 99999 99999 85517 02413 06001 70063 06519 26512
50563 18332 27031 40727 29133 26547 30926 45134 26552 25045 54133 27556
20185 99999 27543 15364 48566 25030 10631 50167 25016 88283 49563 25535
```

(a) Temp code received through GTS channels.

```
99036 06010 11503 00197 99999 99999 85517 02413 06001 70063 06519 26512
----------------- ----------------- ----------------- -----------------
     SURFACE           1000 mb            850 mb              750 mb

50563 18332 27031 40727 29133 26547 30926 45134 26552 25045 54133 27556
----------------- ----------------- ----------------- -----------------
     500 mb            400 mb             300 mb              250 mb

20185 99999 27543 15364 48566 25030 10631 50167 25016 88283 49563 25535
----------------- ----------------- ----------------- -----------------
     200 mb            150 mb             100 mb               88 mb
```

(b) Arrangement of data in different pressure level.

Figure 4.1 Coded meteorological data as obtained through GTS channels.

(b) LOC.ASC : The filename LOC indicates the location of land stations which receives the meteorological data. The extension ASC is used to indicate that this file contains ASCII formatted records. A data sample is shown in figure 4.2. The recordsize of the entered data in this file is 80 bytes in size. The fields of each record contain the country block number, station number, latitude, longitude and the city respcetively. This information is internationally accepted and is published in [5] each year.

Area code	Land station no.	Latitude	Longitude	City
41	923	23.77	90.38	Dhaka
42	809	22.65	88.45	Calcutta
42	871	21.47	80.20	Gondia
42	623	24.77	93.90	Imphal
43	003	19.12	72.85	Bombay
48	008	25.37	97.40	Myitkyina
55	591	39.67	91.13	Lhasa

Figure 4.2 Geographical location of land stations.

(c) MET.DAT : This is the output file which stores all the necessary file and header information and the processed data. It contains 112 integer words (each of 4 bytes in size) of data in each record. This file contains binary data which helps user in accessing the data easily and randomly. All the data are converted to integer format with suitable and multiple factors for ease of work.

The arrangement of data in this file is shown in Appendix B

4.3 Description of the decoding program

Figure 4.3 shows the flow chart of the decoding program. At the beginning of the program the related parameters are initialized. The date and time are then read from the MET.ASC file and saved for the MET.DAT file.

The computer reads the country block number and station number from the MET.ASC file. The computer then starts reading the LOC.ASC file. If the country block number and station number read from the LOC.ASC file are the same as read from the MET.ASC file then it reads the latitude and longitude values from the LOC.ASC file. The computer next reads the parameters from the MET.ASC file. After necessary decoding of the parameters according to the rules described in section 3.3 the computer then stores the data in the file MET.DAT and outputs to the user terminal (a sample is shown in figure 4.4).

AREA = 55 STATION = 591 CITY = Lhasa

GOE	AIR TEMP.	DEW TEMP.	WIND
650	-11	-231	29502
99999	99999	99999	99999
99999	99999	99999	99999
99999	99999	99999	99999
5720	-125	-345	25012
7370	-265	-435	27529
9390	-413	-553	27043
10610	-457	-587	27057
12080	-505	-645	25561
13920	-605	99999	25552
16370	-717	99999	26537

Figure 4.4 Processed GTS data as sent to the video terminal.

If the country block number and the station number read from the

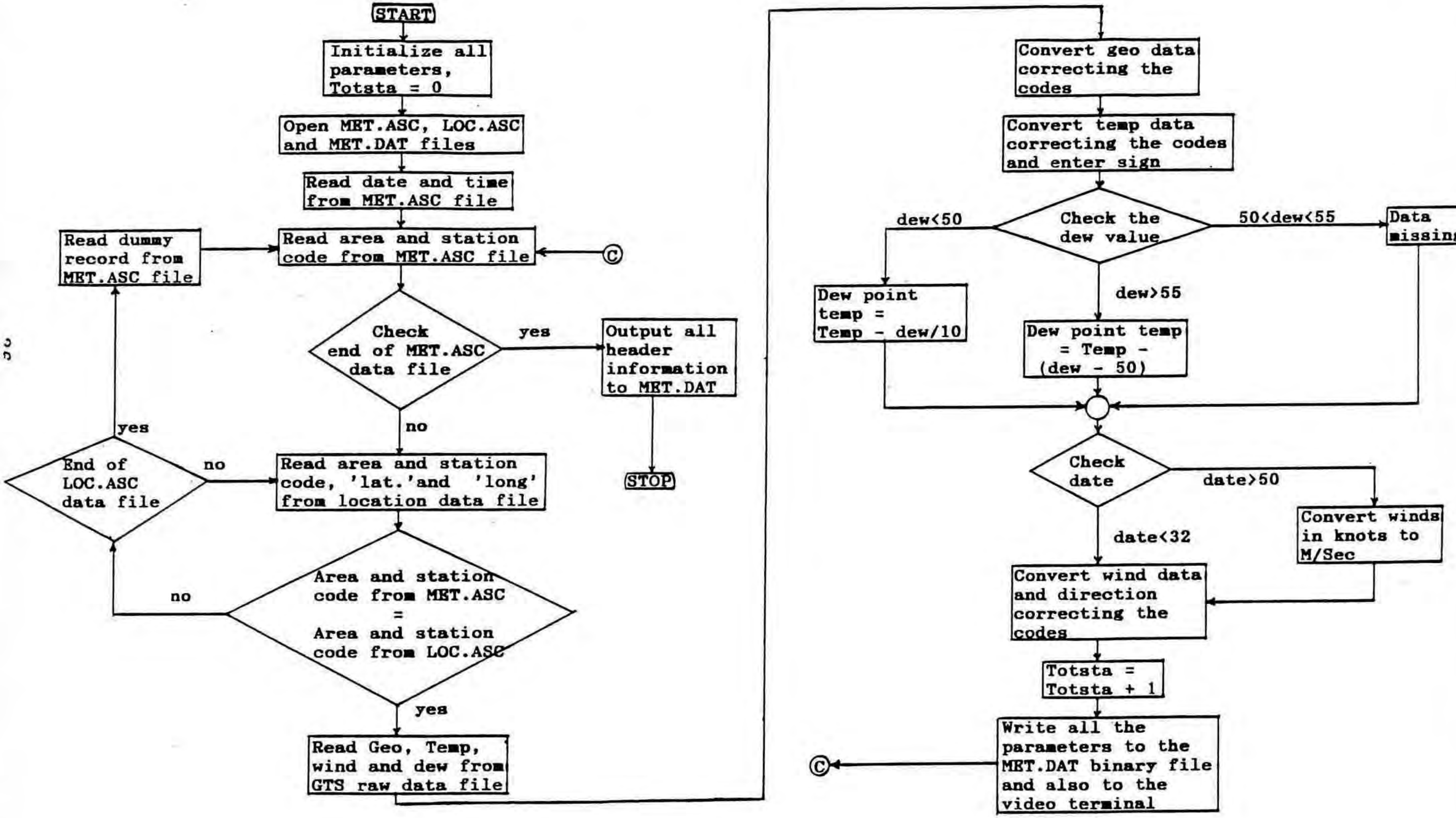


Figure 4.3 Flow chart of the decoding program MET.

MET.ASC file are found not to be same as in the LOC.ASC file the computer discards the parameters from the file MET.ASC. The computer then starts with the next record in MET.ASC. If the computer reaches the end of file of MET.ASC it abondones the processing of data, goes to the statement number 555, outputs the header informations in record number 1 of MET.DAT. It also outputs the data for total number of station received to the user terminal and terminates the execution of the program.

4.4 Program operation

This program named MET may be run from any user terminal (e.g. VT100) or hardcopy terminal (e.g. LA120 Decwriter). It can be run from any user account. The program is initiated by typing RUN MET

The program is terminated when :

(a) CNTRL-C or CNTRL-Y is pressed from the keyboard or

(b) the computer system or program fails.

When a station data is successfully detected the program processes and produces the output data. At the end of the work the total number of stations successfully located is displayed in the user terminal.

4.5 Error condition

Execution of the program may be terminated due to errors. This may happen if

(a) any of the files MET.ASC or LOC.ASC is not found;

(b) inconsistent data type is encountered, e.g. the program is reading an integer data while the data is actually real.

4.6 Program environment

This program can be executed in any mini or mainframe computer with a Fortran-77 compiler. The data files LOC.ASC and MET.ASC should be available in the computer at the time of running of this program.

4.7 Summary

Decoding of GTS data is necessary because it comes in coded form. It is decoded and processed to obtain various meteorological parameters. The input and output data files used in the decoding program are described. The decoding program is also briefly described. The program operation and error condition that arise when running the program are also described. The programming environment required to run the decoding program is briefly described.

Chapter 5

Contouring of geopotential height data

5.1 Introduction

Geopotential height is defined as the height of equal pressure level from the sea surface and is expressed as GPH = $h\rho g$ where ρ = 1 for air mass. This parameter can be obtained from both the GTS data and the TOVS data. With the existing facilities of SPARRSO the value of GPH parameter can be displayed on world data base map [34] using an IIS color monitor as shown in figure 5.1. Contouring of GPH is usually done on weather charts. In BMD the contouring of GPH data obtained from the GTS data is at present being done manually. A sample of manually prepared contour map at BMD is shown in figure 5.2.

In this chapter a computer technique for displaying contour map on world data base map using IIS monitor is presented. The computer technique involves computation and processing of the input data and graphical representation of the processed data on the IIS color monitor. In the developed program provisions have been made to accept both the GTS data and the TOVS data.

The total job of contouring starting from the GTS data or the TOVS data is done in two parts. These two parts are

(i) Creating grid data file.

(ii) Preparing contour map using the data from the grid file.

These two parts are done using the VAX-11/750 minicomputer of SPARRSO. Hence these two jobs could not be done in a single program due to the limitations of the existing system. For the first part the usual Fortran language is used while for the second part the graphical

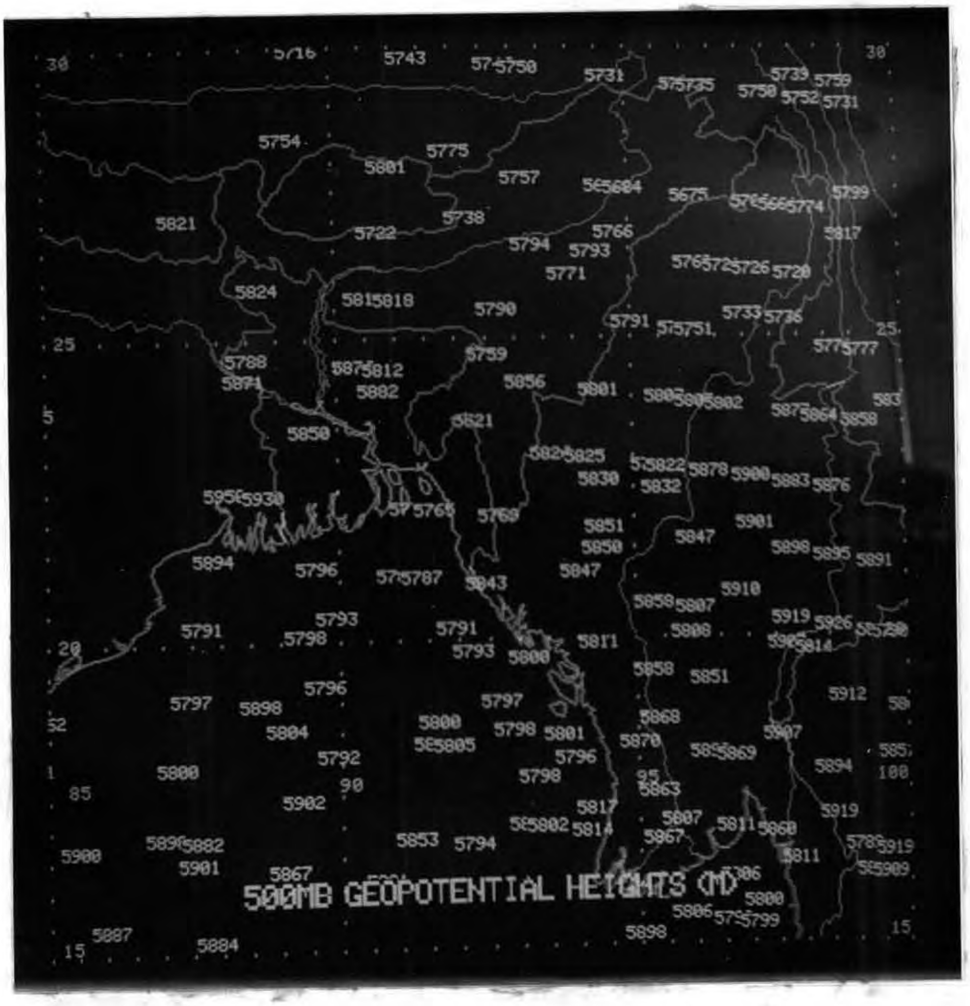


Figure 5.1 Existing display of geopotential height data at SPARRSO.

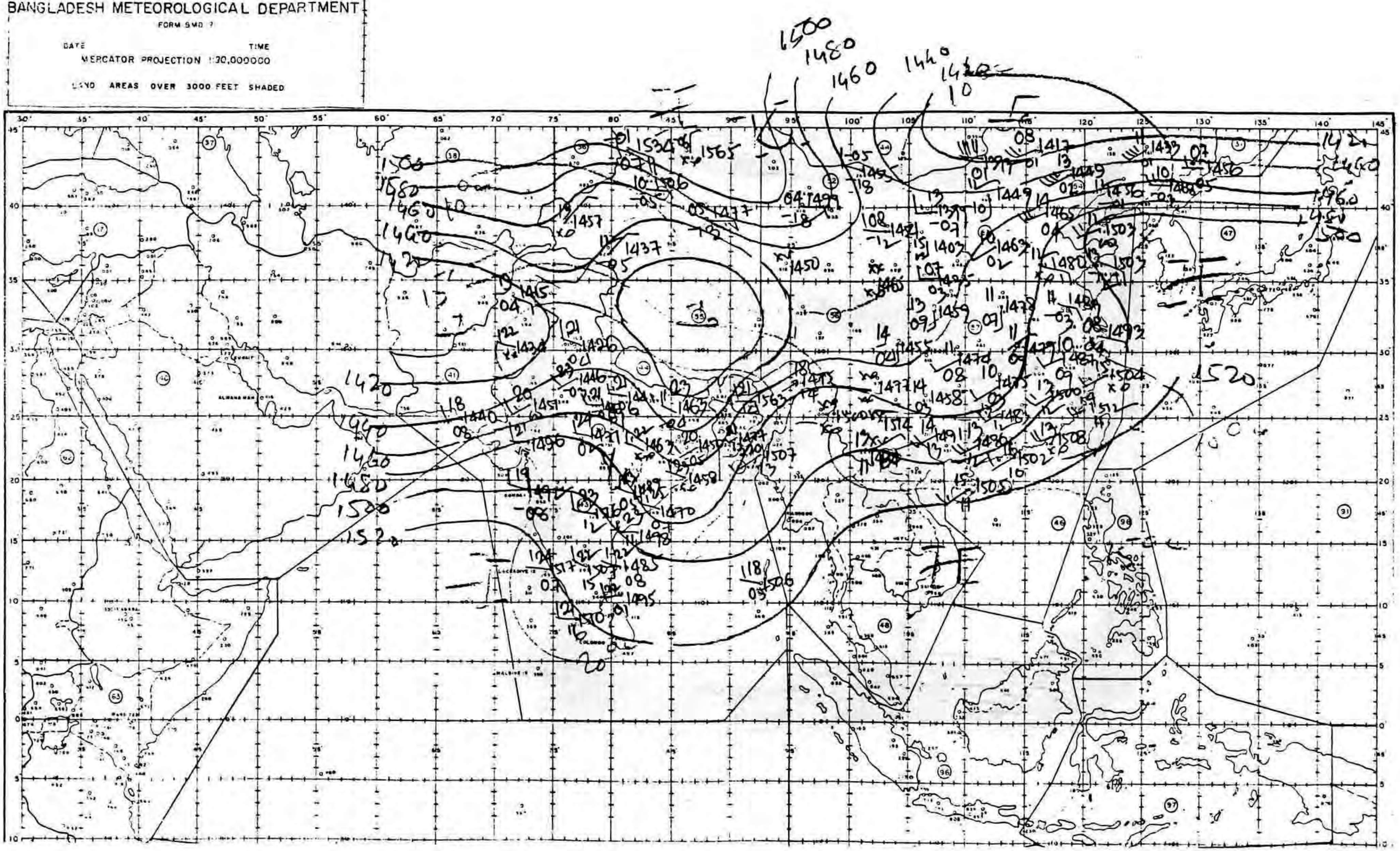


Figure 5.2 Manually prepared contour map (courtesy BMD).

package IDL is used. The graphical package IDL can not be used through fortran using the existing facilities at SPARRSO. The preparation of contour map is shown in block diagram in figure 5.3

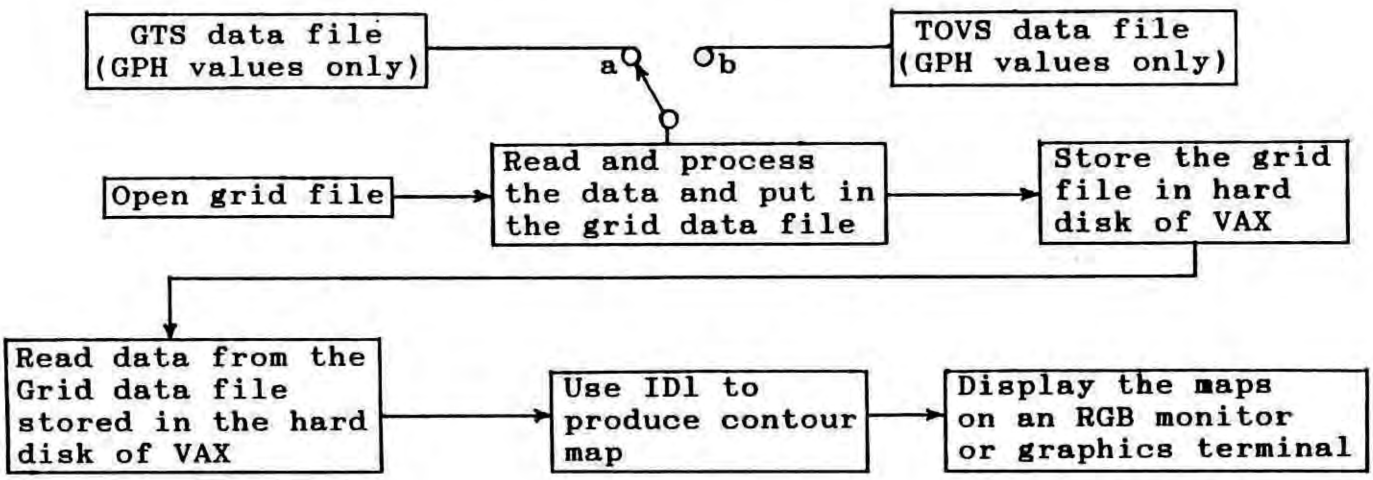


Figure 5.3 Block diagram showing the processes involved in obtaining a contour map from GPH values.

Section 5.2 describes the various data files used in the programs developed which are also described in the subsequent sections.

Section 5.3 describes the technique involved in producing a 'grid data file' from the GTS data or the TOVS data and storing the data in a two dimensional 64x64 array. The name of the program used for the purpose is GEO.FOR which is prepared in Fortran-77.

The technique of drawing contour lines of GPH data with proper geographic locations is described in section 5.4. The contour lines are drawn using the grid data files created by the program GEO. The program used for this work is GEO_CONTOUR.PRO which is a high level programming language (used extensively for plotting of data) known as IDL.

Flow charts of the computer programs are given in the relevent sections and the program listings are presented in Appendices C and D.

5.2 Input and output data files

The contouring program accepts either MET.DAT or TOVRETn.DAT data files as input depending upon whether it is GTS data or TOVS data. The two data files are describred below.

(i) MET.DAT : This file as described in section 4.2 is created by the program MET as described in section 4.3.

(ii) TOVRETn.DAT : These files are created by processing the TOVS parameters as obtained from NOAA satellites. The processing of TOVS data is done using a software package prepared by 'Cooperative Institute for Meteorological Satellite Studies' (CIMSS), Wisconsin university, Wisconsin, USA. These binary files contain all the meteorological parameters in binary number. Fast reading and writing of these files is possible because of the binary form. Here TOVRET indicates that the file contains TOVS retrieval parameters and 'n' signifies any letter A to Z. Fields 5 through 15 are used from these files which actually contain the geopotential height data.

A program named as GEO.FOR described in section 5.3 processes the data for MET.DAT or TOVRETn.DAT and puts in grid data files according to the data type selected i.e. GTS data or TOVS data. If TOVS data is selected the output processed data are put in 9 files named as GE1000.DAT, GEO850.DAT, GEO700.DAT, GEO500.DAT, GEO400.DAT, GEO300.DAT, GEO250.DAT, GEO200.DAT and GEO150.DAT. For GTS data type, the processed data are put in 10 data files named GTSSRF.DAT, GT1000.DAT, GTS850.DAT, GTS700.DAT, GTS500.DAT, GTS400.DAT, GTS300.DAT, GTS250.DAT, GTS200.DAT and GTS150.DAT. These grid data files are later used for contouring purposes.

5.3 Program to create the grid data files

To make a precise contour map one needs a good grid data file. In a good grid file, data are arranged in grid positions and the data values are uniform in nature. However, the data which are obtained either by GTS channels or through the TOVS instruments are scattered in a fixed latitude and longitude boundary. Preparation of a flawless contour map requires good skill of the person involved.

This computer program named GEO (written in Fortran-77) prompts an user to decide the latitude and longitude boundaries describing the area of interest. It also asks to decide the type of data to be used i.e. 'GTS data' or the TOVS data. The program then reads all the data values, latitude and longitude boundaries and date of data received from the required data file. After necessary processing of the geopotential height data by the subroutines it outputs the grid values to the binary data files GE1000.DAT and GEOhhh.DAT or GTSSRF.DAT, GT1000.DAT and GTShhh.DAT depending on the TOVS data or GTS data respectively. Where hhh stands for pressure levels from 850 MB TO 100 MB. The required subroutines are described briefly in the following subsections. The flow chart of the program GEO is presented in figure 5.4 and the subroutine program READ_GEO is presented in figure 5.5. The program GEO.FOR and the subroutine program READ_GEO.FOR are attached in Appendix C.

5.3.1 Subroutine for reading geopotential height data

This subroutine starts processing from the 1000 mb geopotential height data and processes upto 150 mb pressure level. It also processes GPH values of surface presurre level if data from GTS sources are used. The subroutine either opens the input data file MET.DAT for GTS data or TOVRETn.DAT for TOVS data depending on the

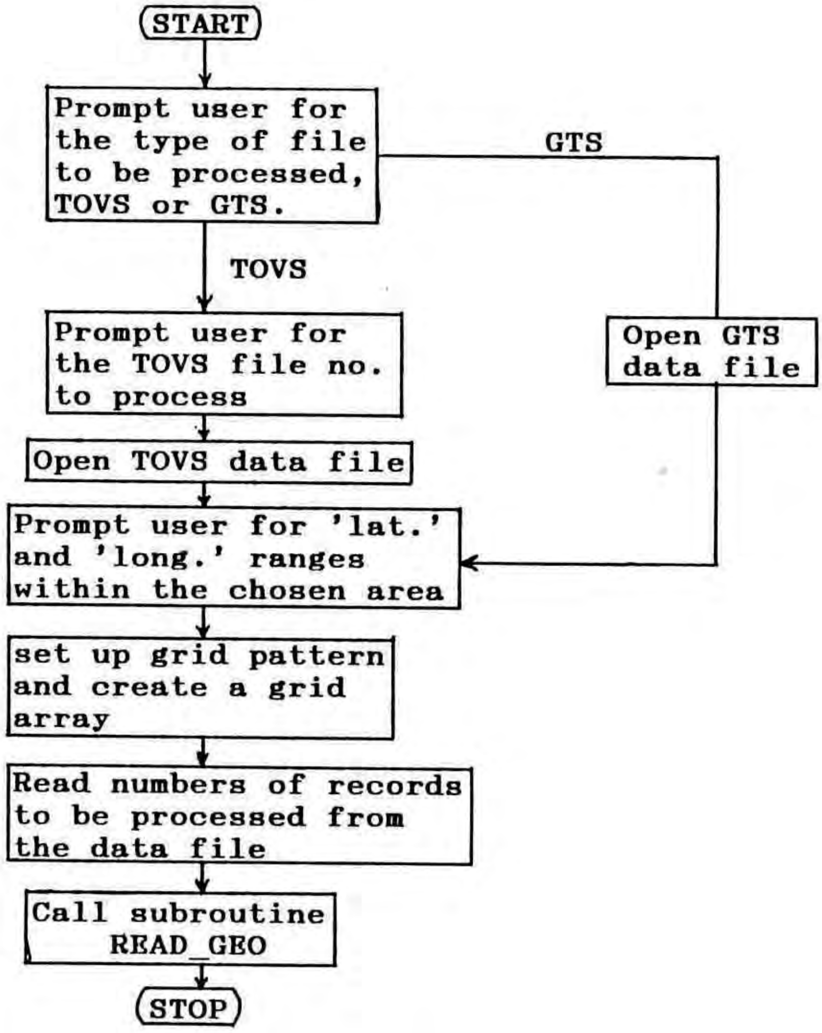


Figure 5.4 Flow chart of the program GEO. The flow chart for the subroutine READ_GEO is shown in figure 5.5. Arguments for the subroutine are not shown for simplicity.

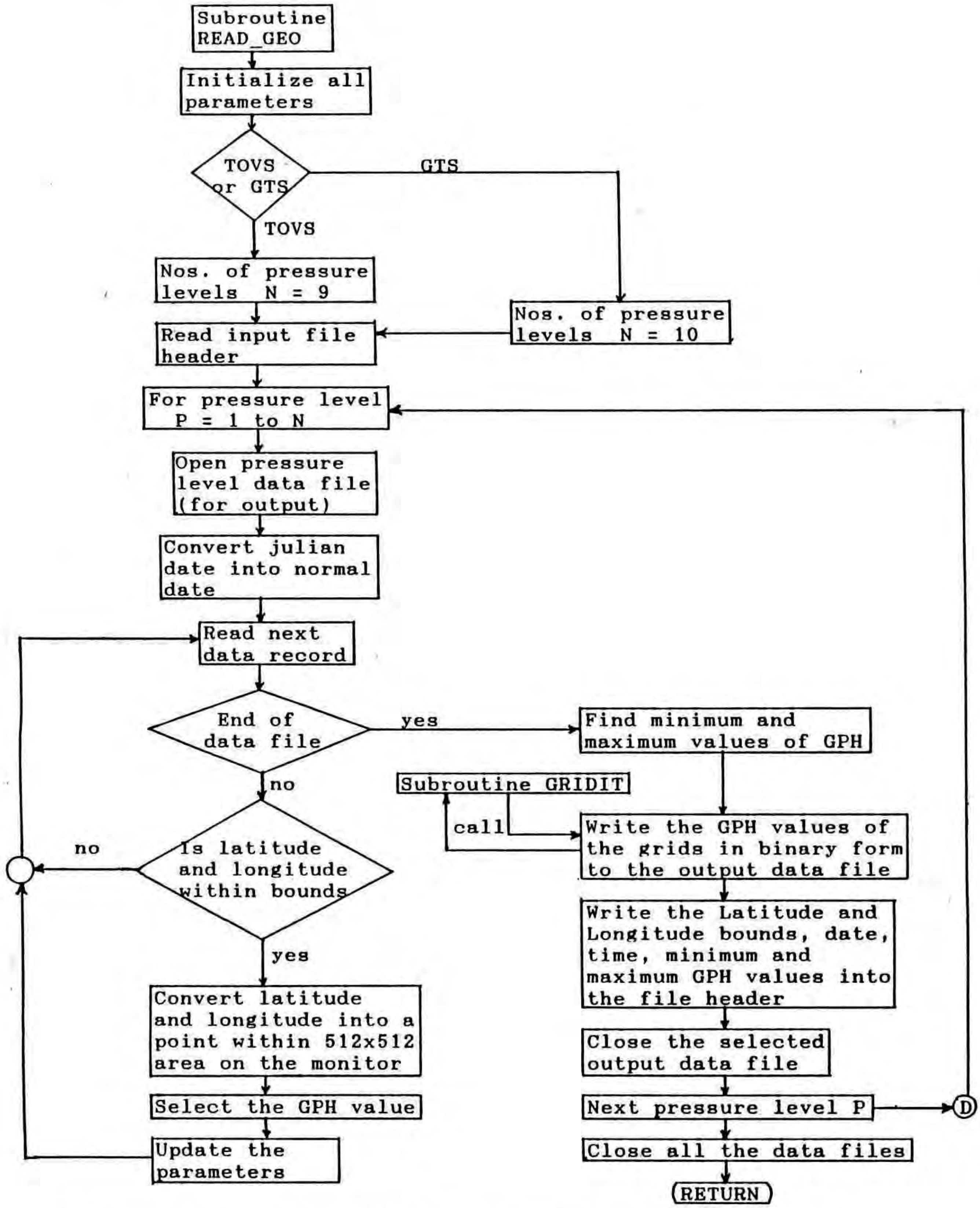


Figure 5.5 Flowchart of the subroutine program READ_GEO

user's request. The subroutine READ_GEO reads data using two subroutines RETIO or METIO and converts the julian date into a normal date using another subroutine NYYDDD. The program then checks whether the GPH values are within the predetermined latitude and longitude boundaries and separates those data falling within the boundaries. Another subroutine BILIN is then used to convert the latitude and longitude locations of the separated data coordinates in the form of (x,y). The selected latitude and longitude ranges are then scaled for displaying within the 512 pixels by 512 pixela area on the IIS monitor. Another subroutine FINDM is then called to find the minimum and maximum of the obatined GPH data. The program then creates a grid file in the form of a 64x64 array. The GPH values are then processed by the subroutine GRIDIT using a technique as described in section 5.3.3. The values are then stored in the the created grid file. The information of latitude and longitude boundaries, date of data received, maximum and minimum GPH values are stored in an additional space in the grid file.

5.3.2 Subroutines to manipulate data files

Two subroutines METIO and RETIO are used for manipulating the data files. The subroutine RETIO helps in opening, closing, reading and writing the file TOVRETn.DAT. The subroutine METIO helps in opening, closing, reading and writing of the data file MET.DAT. An argument IOPT is used in these programs which determines the type of operation to be done on a data file. IOPT = 0 indicates reading operation, IOPT = 1 indicates writing operation and IOPT = -1 indicates closing a data file.

5.3.3 Obtaining grid data files

GPH data obtained from GTS data or TOVS data are distributed in an area described by latitude and longitude limits. The distribution of these data is not uniform as can be seen in figure 5.6 and 5.7. These data are recalculated to constitute a 64x64 grid array. Each cell of this array contains a GPH data value. Figure 5.8 shows the grid array with the grid points. Figure 5.9 shows the input TOVS data as overlayed on the grid array. These data are useful to prepare contour maps. The conversion of the data is done by using the $1/r^2$ method which is popularly known as weighted average method. In this subroutine 'three point' weighted average value is calculated.

The distance calculated from a selected grid point to a received data point is the distance between two coordinates in a rectangular coordinate system as

$$d_i = (x_j - x_i)^2 + (y_j - y_i)^2 \quad \text{for } i = 1, n \qquad \ldots \quad (5.1)$$

where, (x_j and y_j) are the coordinates of the selected grid point location and (x_i and y_i) is the point of data location.

The GPH value at the jth grid point (x_j, y_j) is calculated by the mathmetical relationship

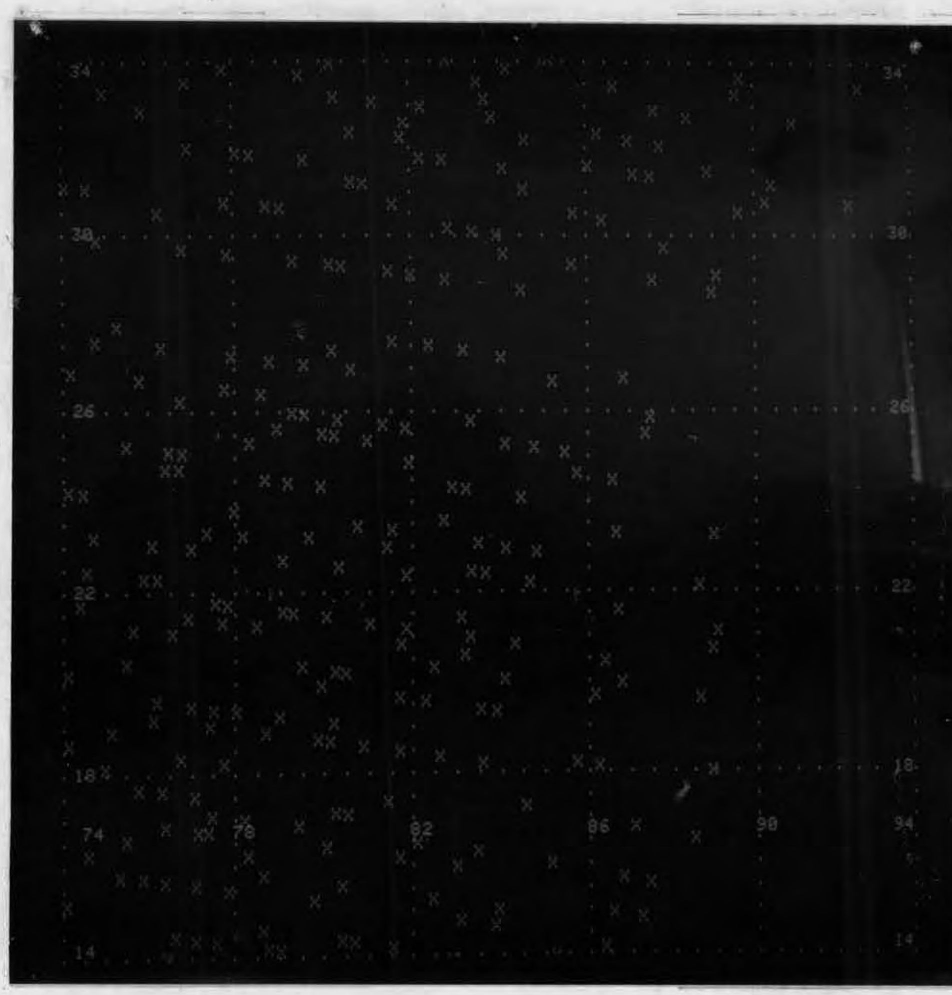
$$G(x_j, y_j) = 1/w^2 \sum_{k=1}^{3} H_k/d_k \qquad \ldots \quad (5.2)$$

where, H_k is the received GPH value at the point of observation nearest to the reference point (x_j, y_j) and $G(x_j,y_j)$ is the computed GPH value at the jth selected grid point (x_j, y_j).

w^2 is the weighted value which is computed using the relationship

$$w^2 = \sum_{k=1}^{3} \frac{1.0}{(x_j - x_k)^2 + (y_j - y_k)^2} \qquad \ldots \quad (5.3)$$

<u>Figure</u> <u>5.6</u> Latitude and longitude locations of the data sending land stations. The locations are shown with green 'x'. The data from these stations come through the GTS channels.



<u>Figure</u> <u>5.7</u> Latitude and longitude locations of the points at which the meteorological data are received by the TOVS sensors of the NOAA satellite. The points are shown with green 'x'.

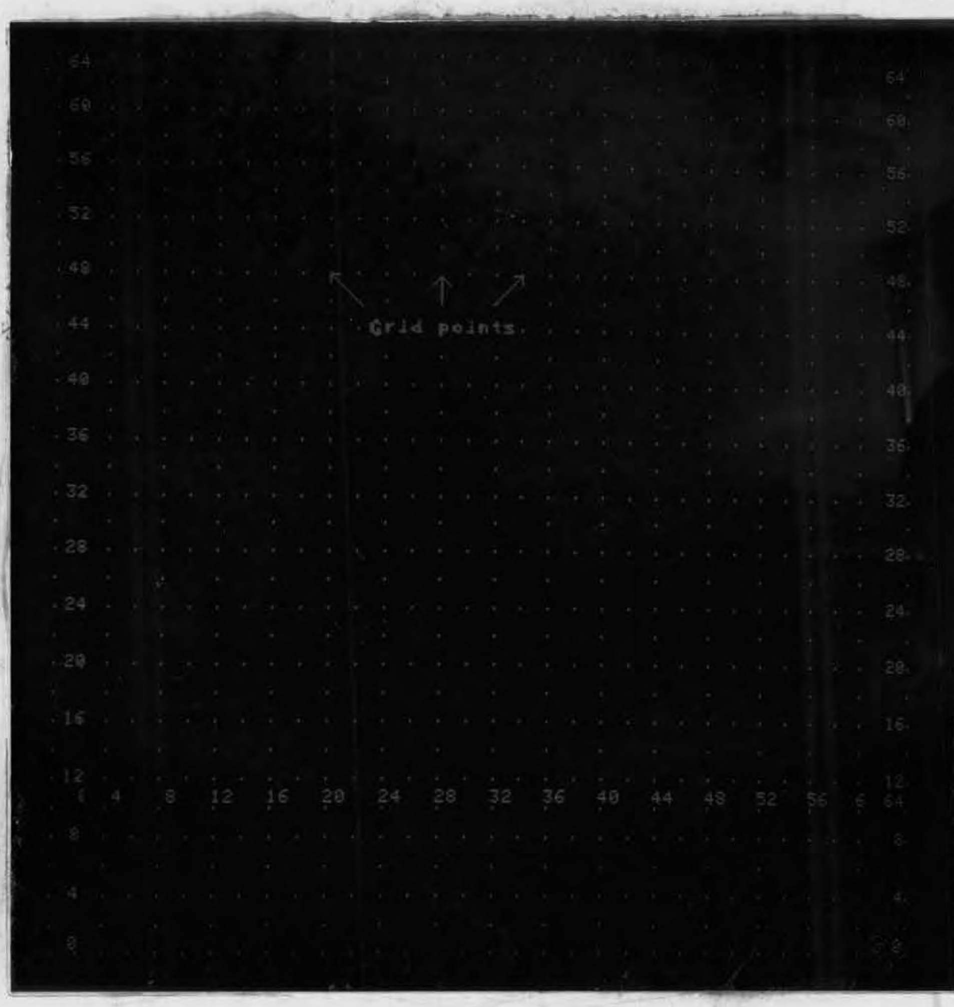


Figure 5.8 Latitude and longitude locations of the grid points at which the GPH values are computed. The grid points are shown with green dots.

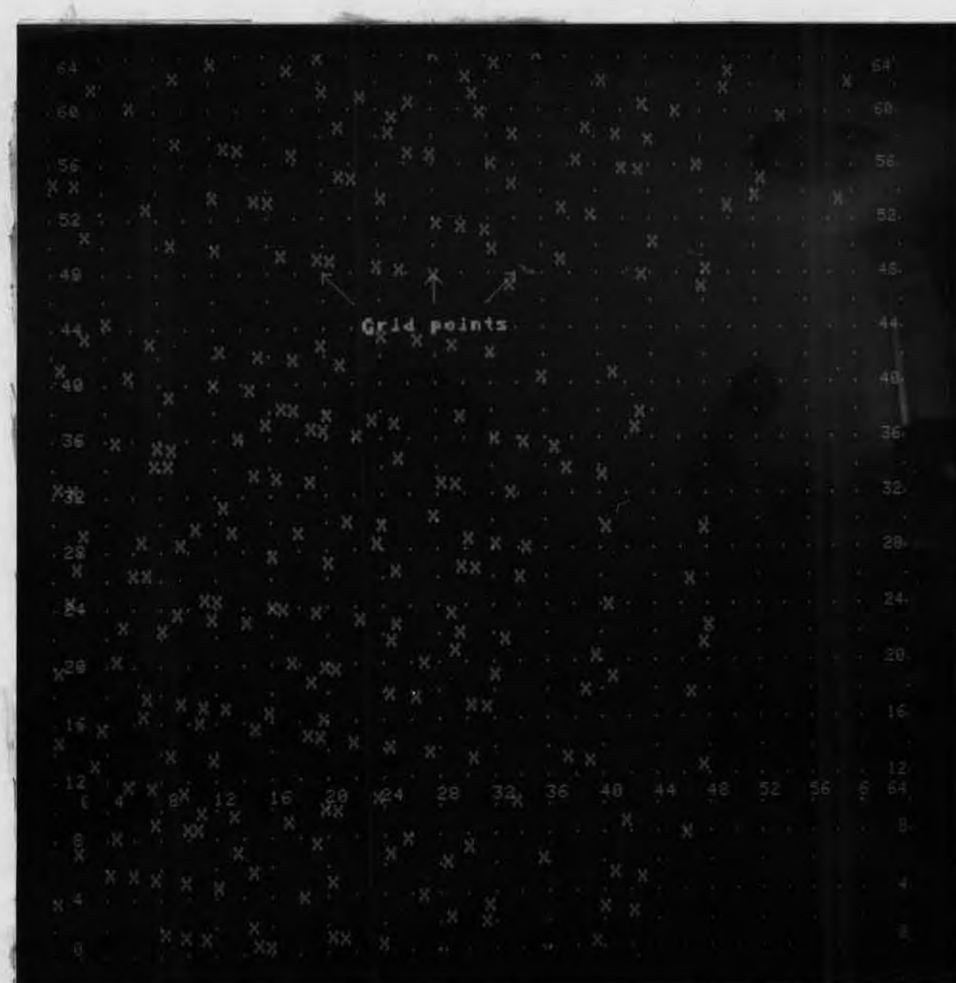


Figure 5.9 Superimposed display of actual data sending locations and the locations at which computed data are obtained. The latitude and longitude locations at which the TOVS data have been obtained are shown with orange 'x'. The latitude and longitude locations at grid points are shown with green dots. The two location maps are superimposed for comparison.

The location of the received GPH data points are suitably scaled prior to putting in the grid map. Since in this work a 64x64 grid array is used the data locations are scaled so that the values fall within the range of 1 and 64 location. The scaling is done using the relationship

$$x_j = [\ s/u_j * (\ x_j' - x_m)] + 1.0 \qquad \ldots (5.4)$$

$$y_j = [\ t/v_j * (\ y_j' - y_m)] + 1.0 \qquad \ldots (5.5)$$

where, x_j is the scaled longitude value for the jth received data y_j is the scaled latitude value for the jth received data, s is the difference between the maximum and the minimum x-axis (normalized longitude) location required for the scaling. In this work it is 64 - 1 = 63, t is the difference between the maximum and the minimum y-axis (normalized latitude) location required for the scaling. In this work it is 64 - 1 = 63. u_j is the difference between the maximum and the minimum x-axis (normalized longitude) locations which are to be scaled. v_j is the difference between the maximum and the minimum y-axis (normalized latitude) locations which are to be scaled. x_m is the minimum x-axis value. y_m is the minimum y-axis value.

After scaling the locations, the GPH data value at grid points are calculated using equation (5.1). The GPH data values are then smoothed using a subroutine SMOOTH. If smoothing is not done then large numbers of undesirable tiny contours will appear all over the display in addition to the desirable normal size contours. The subroutine SMOOTH uses a 9-point smoothing function. At any point j the smoothing of the GPH value $G(x_j,y_j)$ is done using the relationship

$$G'(x_j,y_j) = 1/f\ (\ G(x_j,y_j) + c * \sum_{i=j-1}^{j+1} \sum_{k=j-1}^{j+1} Z(i,k)) \qquad \ldots (5.6)$$

$$\text{for } i=k\neq j, \text{ and } 2 < j < 63$$

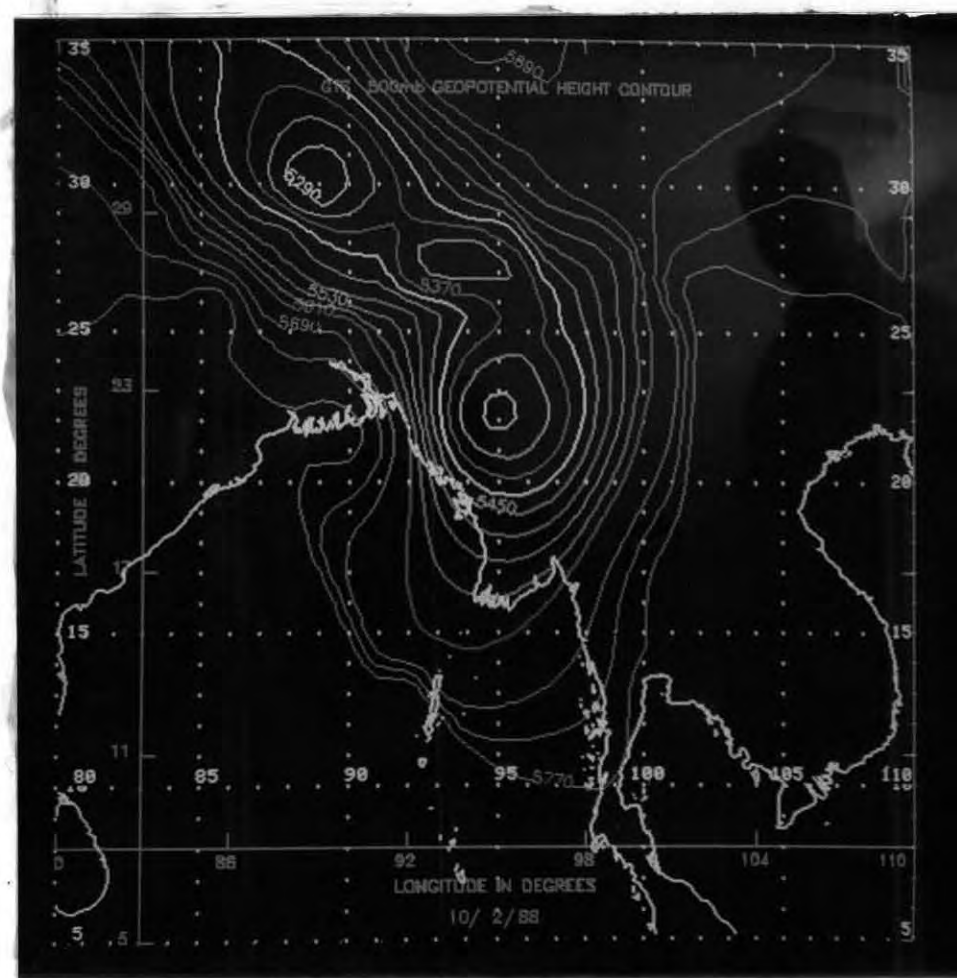
where, f = 1 + 8.0 * c, and the constant c = 0.5 for a 9-point smoother. The GRIDIT subroutine also uses another subroutine FINDM to calculate the minimum and the maximum values of an array.

5.4 Program to draw the contour lines

After creating the grid data files as described in section 5.3, the program GEO_CONTOUR.PRO does the necessary contouring of the data. figure 5.10 shows a contour map using GTS data and in figure 5.11 another contour map is presented using TOVS data.

The program, written in a high level programming language and used extensively for plotting purposes, is run inside the IDL environment by typing XIDL. A flow chart of the program GEO_CONTOUR.PRO is shown in figure 5.12 and the program listing is attached in Appendix D. When the IDL> prompt appears the program is started by typing .RUN GEO_CONTOUR.PRO. The program is compiled, linked and is ready for use.

The pressure level for which the contour lines are to be drawn is first chosen. The program then notifies about the area bounded by the latitude and the longitude limits, date, type of data (GTS data or TOVS data) stored in the 'grid data file' and the minimum and the maximum GPH value. Total numbers of contour levels, mimimum and the maximum levels that are to be contoured are then selected. The program then reads the data from the 'grid data file' and makes necessary contour plotting on the IIS color monitor or the user graphics terminal.



<u>Figure</u> <u>5.10</u> Contour map for 500 mb pressure level using GTS data.

Figure 5.11 Contour map for 500 mb pressure level using TOVS data.

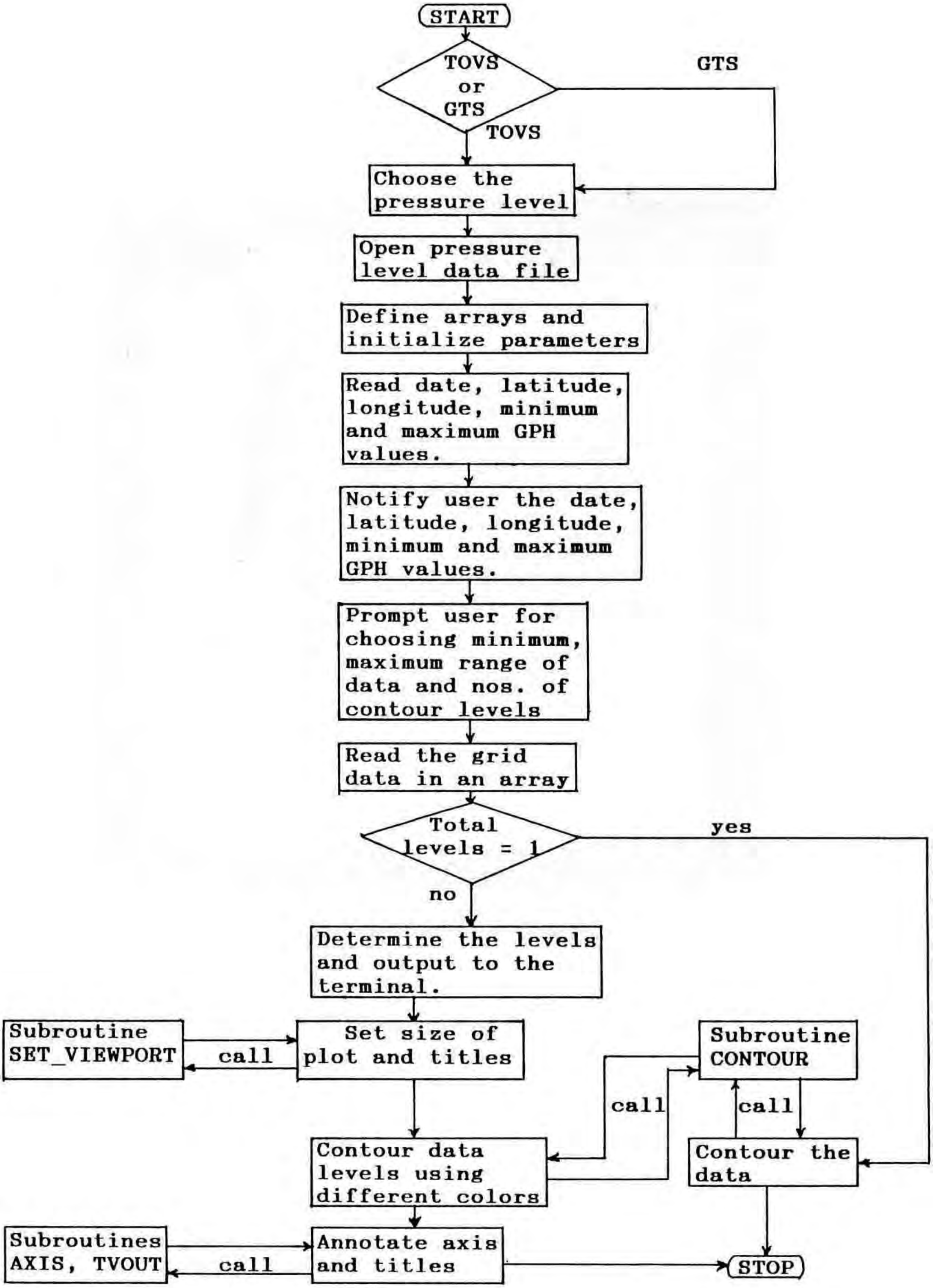


Figure 5.9 Flow chart of the program GEO_CONTOUR.PRO

Summary

Various data files used to prepare contour maps are briefly described. The technique involved in creating grid data files is explained. The computer program used to draw contour maps are described.

Chapter 6

Display of wind velocity and direction

6.1 Introduction

Wind is one of the much used parameters in weather forecasting. Direction and velocity are required for specifying the wind parameter. The direction is expressed in terms of geometrical angles with the reference line coinciding with the direction of true north. The angle is calculated by the amount of rotation in the clockwise direction. The usual units for the wind velocity is either meters per second or nautical miles. The wind blows towards the reference point. Some examples are shown in figure 6.1. The amount of severity with which the storms, cyclones, tornados, norwesters etc. moves is determined by the wind data.

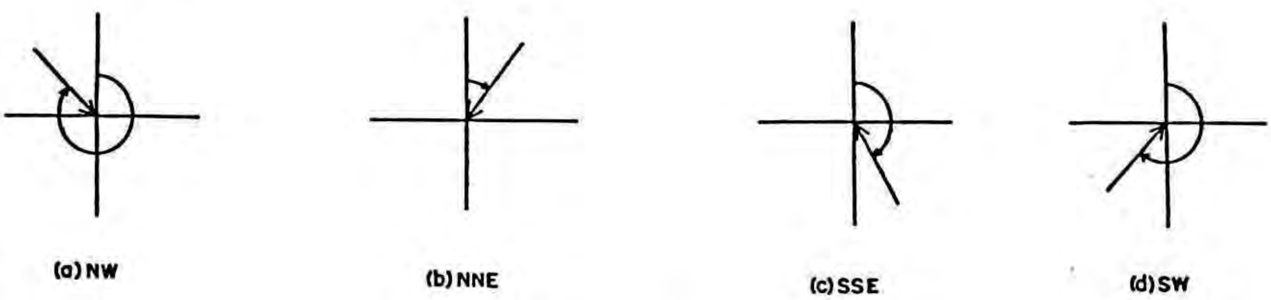


Figure 6.1 Directional conventions used in meteorology for displaying wind in weather map. (a) Wind is from north-west (NW), (b) Wind is from north-north-east (NNE), (c) Wind is from south-south-east (SSE) and (d) Wind is from south-east (SW).

Display of the wind data varies depending on the amount of data available, forecasters' need, users' acceptability and the available features. Conventional display of wind data in symbolic form is shown in figure 6.2. Another form of displaying wind data used by ECMWF and a number of other users is shown in figure 6.3.

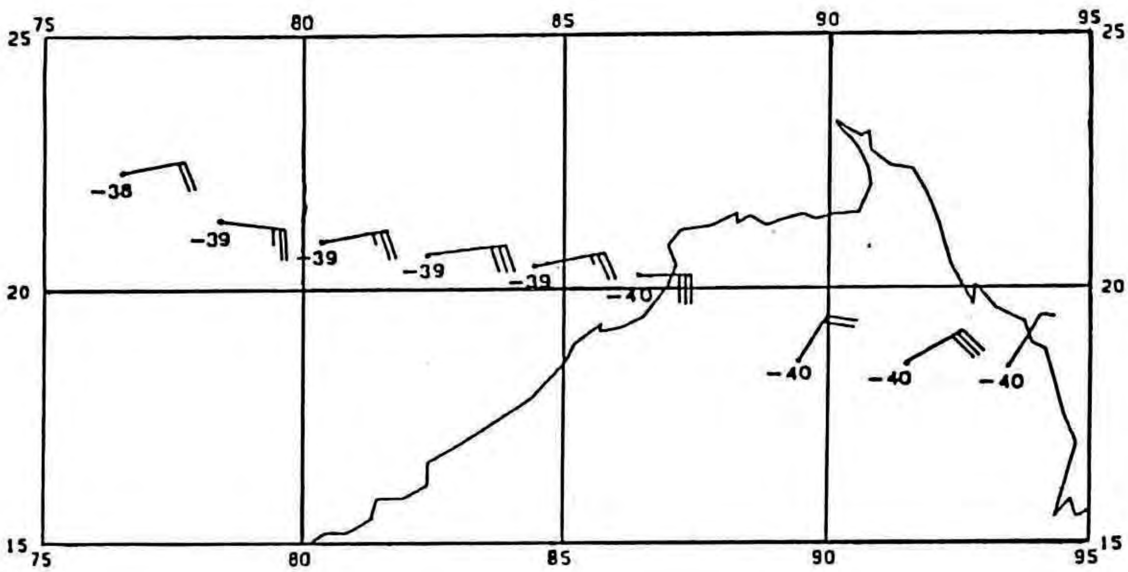


Figure 6.2 Conventional display of wind data with the help of symbols. This figure has been taken from reference [29].

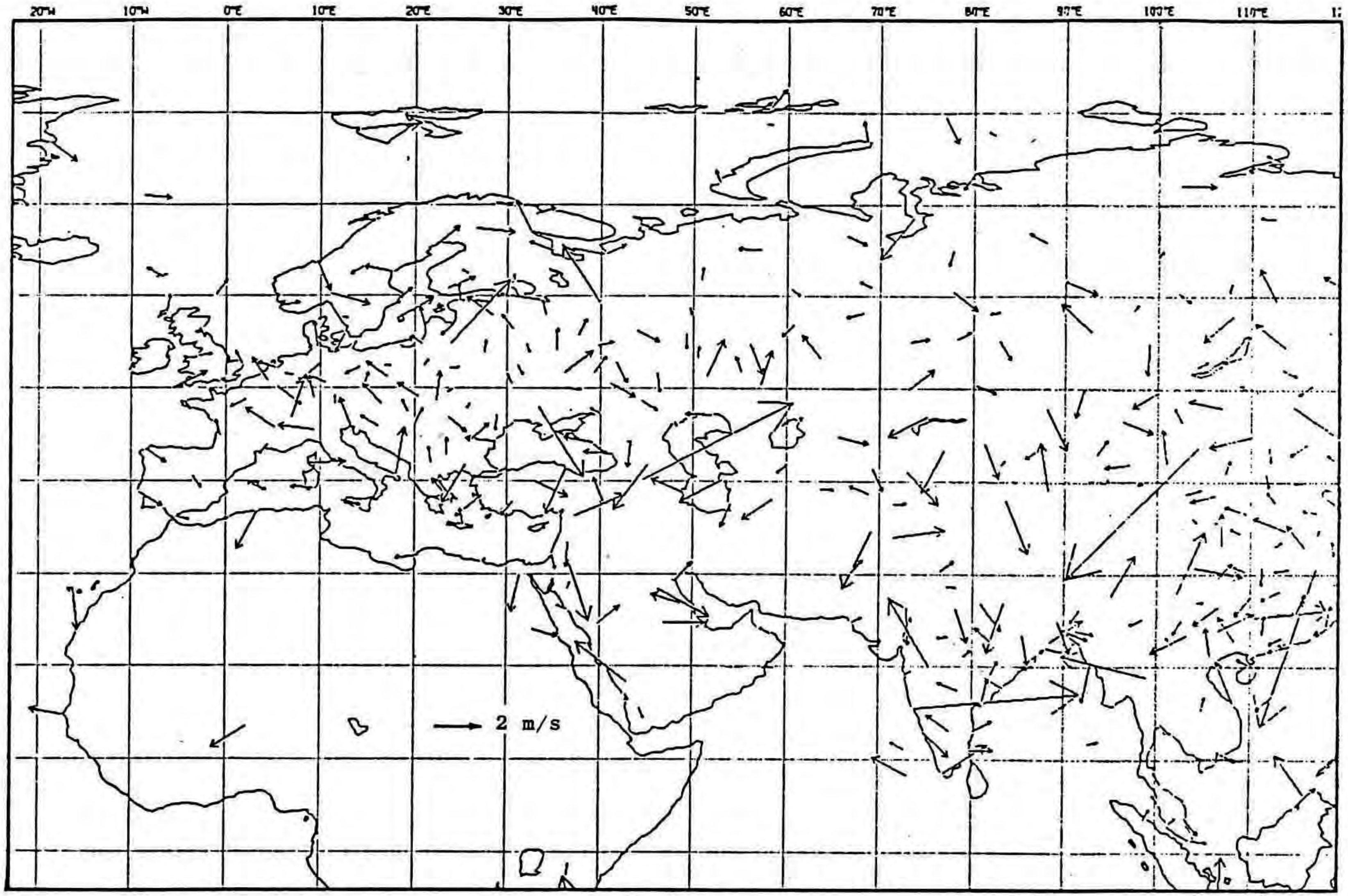


Figure 6.3 Wind data display by ECMWF. This technique is also used by a number of other users. This figure has been taken from reference [28]

In this work another form of display of the wind data is attempted. In this method wind speed of a selected pressure level is displayed on one of the graphics planes [2] of the IIS color monitor in one color. The directional arrows are displayed in another graphics plane using a different color. The two colored graphics planes are then superimposed. The two colors are chosen so that images can be seen in good contrast. The versatility of the graphics planes of the IIS image processor has allowed to achieve this sort of display. The program THESD.DRI is used for this work which utilizes the hardware features of IIS image processor by using a number of 'S575 image processing routines' [3]. The code used is written in Fortran-77 programming language. The program uses both the GTS data and the TOVS data. Flow chart of the computer program THESD is shown in figure 6.6. Flow charts of subroutines THESPLT and VALGET used by the main program THESD are shown in figure 6.7 and 6.8 respectively. The programs are attached in Appendix E.

Section 6.2 of this chapter describes the data files used in this work.

Some of the hardware facilities of IIS image processor that have been utilized in this work are described in section 6.3.

In section 6.4 the methodology and programming concepts that are used in this work are described.

6.2 Input data files for display of wind data

The display program takes two binary data files WINRETn.DAT and MET.DAT as input. The program reads the necessary data from any of these two data files and displays on the IIS color monitor. The data files are described below.

(i) MET.DAT : This file, as described in section 4.3 and Appendix B, is created by the program MET as described in section 4.4. This data file contains latitude and longitude values in field 1 and field 2 of each record. The wind data are stored in fields 81 to 91 of each record as shown below.

Field 81 = wind data for surface
field 82 = wind data for 1000 mb pressure level
Field 83 = wind data for 850 mb pressure level
Field 84 = wind data for 700 mb pressure level
Field 85 = wind data for 500 mb pressure level
Field 86 = wind data for 400 mb pressure level
Field 87 = wind data for 300 mb pressure level
Field 88 = wind data for 250 mb pressure level
Field 89 = wind data for 200 mb pressure level
Field 90 = wind data for 150 mb pressure level
Field 91 = wind data for 100 mb pressure level

Upon request, data of a particular pressure level is displayed on the RGB monitor.

(ii) WINRETn.DAT : This binary data file is created by the use of TOVS software package [9], [14] as created by CIMSS, Wisconsin university, Madison, USA. TOVS data is ingested from the HIRS and MSU instruments of NOAA satellite. These data are then processed by the TOVS software package to retrieve meteorological parameters. The output data are stored in the file TOVRETn.DAT, where n is any letter from A through Z. A modified program WINRET is then executed to calculate geostrphic wind [15] from the geopotential height data and the output data are stored in the file WINRETn.DAT, where n is any letter from A to Z. Wind data in this file is stored as follows

Field 5 = Latitude in degrees.

field 6 = Longitude in degrees.

Field 9 = wind data for 850 mb pressure level

Field 12 = wind data for 700 mb pressure level

Field 15 = wind data for 500 mb pressure level

Field 18 = wind data for 400 mb pressure level

Field 21 = wind data for 300 mb pressure level

Field 24 = wind data for 250 mb pressure level

Field 27 = wind data for 200 mb pressure level

Field 30 = wind data for 150 mb pressure level

Field 33 = wind data for 100 mb pressure level

Upon request, wind data of a particular pressure level and a particular latitude-longitude location is displayed on the IIS color monitor.

6.3 Some IIS hardware features

The IIS model 75 image processing software package utilizes an advanced hardware architecture to perform most processing in real time. The following subsections describe some of the hardware features which are utilized in this program

6.3.1 Graphics memory

The IIS image processor has 14 accessible 512x512x8-bit refresh memories. Images can be loaded into these memories for processing purposes. In addition there are eight 512x512x1-bit graphics planes. The graphics planes are accessed randomly at any point for read or write operation. Of the eight graphics planes one plane is used for displaying instructions to do a certain job. This plane known as the status plane is displayed by pressing the foot-pedal connected with the trackball. A sample status plane is shown in figure 6.4.

	A	B	C	D	F
1	Enter draw mode, any button to exit mode	Change planes/ color	Erase (using blotch)		Quit
2	Delete last vertex	Increment zoom factor	Opaque/ translucent toggle		
3	Pick new vertex	decrement zoom factor	Rubber band toggle	Close region	

Figure 6.4. A menu written on one of 8 graphics planes known as the status plane. This menu has been supplied by IIS.

The eight graphics planes and one bit cursor data stream form a nine-bit data stream which is applied to the graphics color assignment function memory. One or more graphics planes can be simultaneously used for plotting, displaying or other operations. Each graphics plane is color coded prior to use. Hence it is possible to use two or more graphics planes at the same time as has been done in this work.

The graphics multiplexer provides for programming selection of one of the eight graphic planes to drive the status digital to analog converter. Thus one graphics plane can drive a status monitor to display alphanumeric data, menus, histograms, plots etc.

6.3.2 Programmable cursor

The cursor is used to locate specific point or area in an 512x512 area on the color monitor. It can also be used to draw lines, curves and also boxes. The cursor provided with the model 75 image processor can be considered as a "pseudo graphics plane". The cursor RAM, which is 64x64x1-bit memory, can be treated just like the other graphics planes by the graphics color assignment function memory.

The cursor can be shaped as plus, cross, box, circle, arrow or any other user selectable form. The position of the cursor on the 512x512 space of the "pseudo graphics plane" can be controlled by the host computer or by the trackball. The host computer can command the cursor position or read back the cursor position at any time.

6.3.3 Trackball and the foot-pedal

A trackball unit along with a foot-pedal comes up with the IIS model 75 image processor. The trackball is used to manually position the cursor in the X-Y graphics planes. The cursor position is used for many IIS functions such as reading pixel values and X-Y locations, drawing polygons, zooming a specific area of an image, scrolling through the image etc. The cursor position can be moved by rolling the trackball or pressing four specific buttons on the trackball assembly. These four buttons move the cursor up, down, left and right direction by one pixel increment.

the trackball housing also provides fifteen function buttons. Programmers assign specific functions attached with these buttons. Upon pressing one of these buttons, predetermined fuctions associated with that particular button is executed. The functions associated with these buttons can be programmed on graphics plane number 7 (the eigth plane) known as the status plane. This status plane can be displayed on the IIS monitor by pressing the foot-pedal.

6.3.4 Graphics color assignment function memory (GCAFM)

The nine-bit data stream from the graphics mux is applied to the "graphics color assignment function memory (GCAFM)" [3]. The purpose of this GCAFM is to assign one of the possible 32,768 colors ($2 \times 2^5 \times 2^5 \times 2^5$) to each graphics plane. This aids in assigning colors to the graphics planes when the planes overlay each other so that all the graphics planes and the cursor can be seen at the same time.

The GCAFM has 512 locations of 16-bits each. The most significant bit of each location is defined as the overlay/insert or replace/add mode. The next fifteen locations is divided into three groups. Each group contains five bits and is assigned by an intensity value of a color component. The color components are red, green and blue respectively. This is shown in figure 6.5.

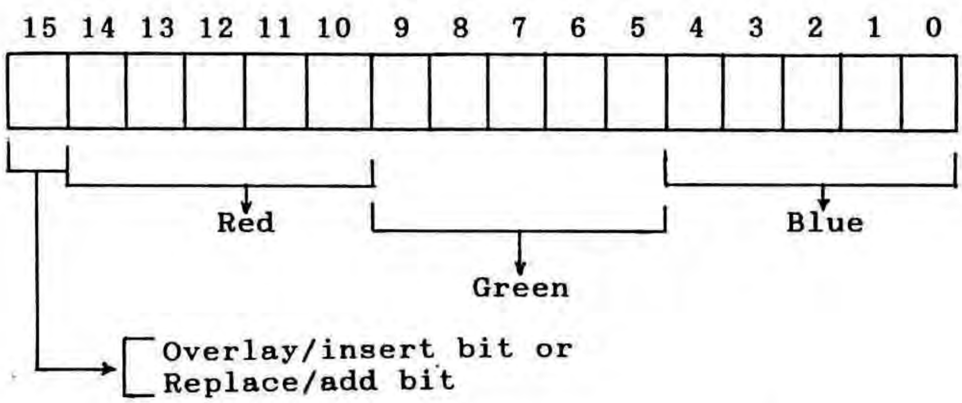


Figure 6.5 One of the Locations of Graphics Color Assignment Function Memories (GCAFM) showing the arrangements of the color components information

6.4 Program to display wind data

Wind data is displayed on the IIS monitor using a IIS function. This function is created by assigning the name of the function to the computer program THESD as DIR>ADD WIND THESD [17]. The computer program THESD is written in Fortran-77 and uses a number of IIS routines [2]. The function is executed inside the IIS image processing environment. The environment is invoked by typing CI (stands for command interpreter). When an CI> prompt is obtained, the IIS image processor is initialized by typing CI> ACQUIRE [16]. After acquiring the refresh memories and the graphics planes of the image processor,

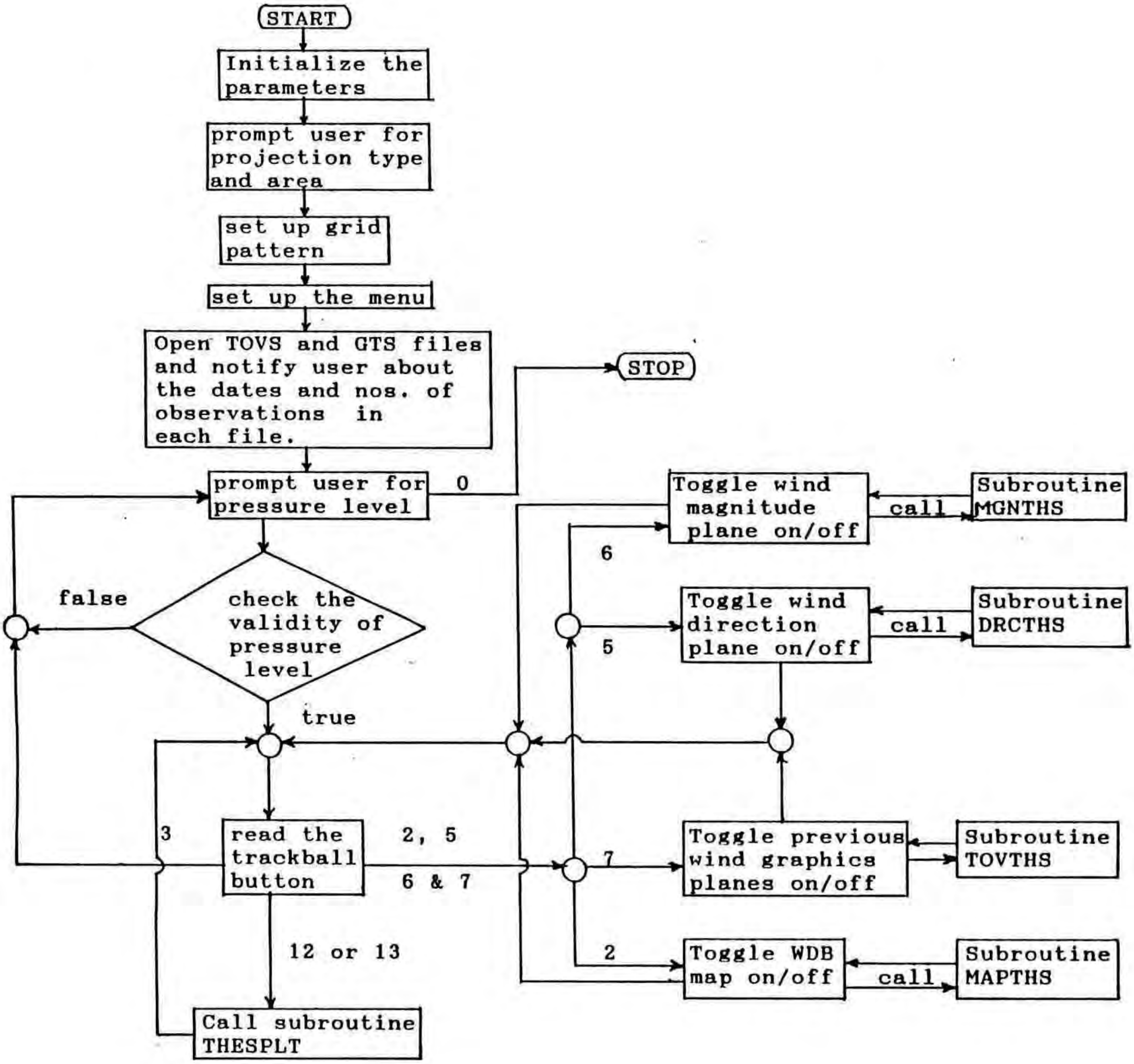


Figure 6.6 Flow chart of the main computer program THESD. Subroutines THESPLT and VELGET are shown in figures 6.7 and 6.8.the following pages. Arguments for the subroutines are not shown for simplicity.

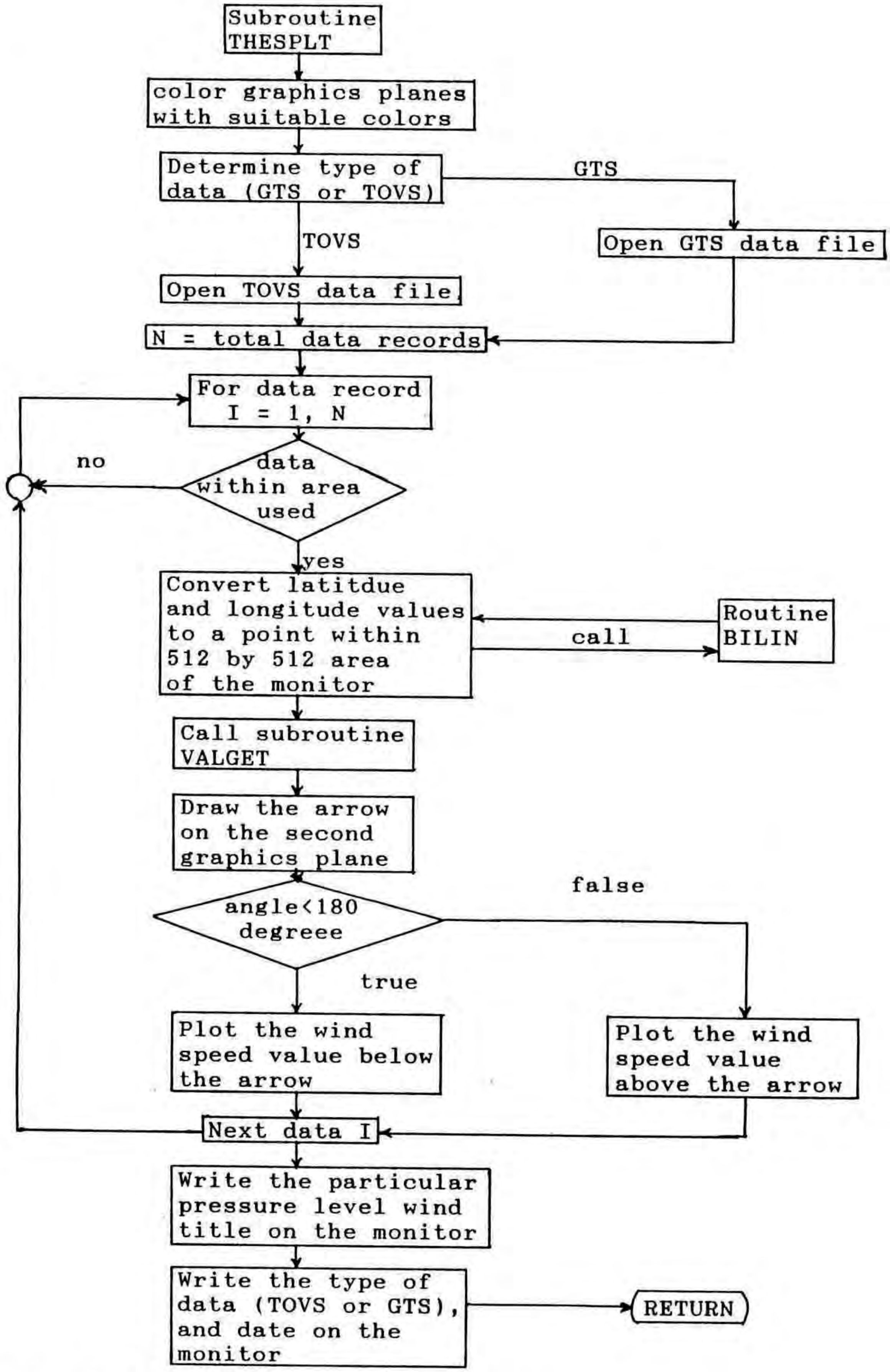


Figure 6.7 Flow chart of the subroutine program THESPLT

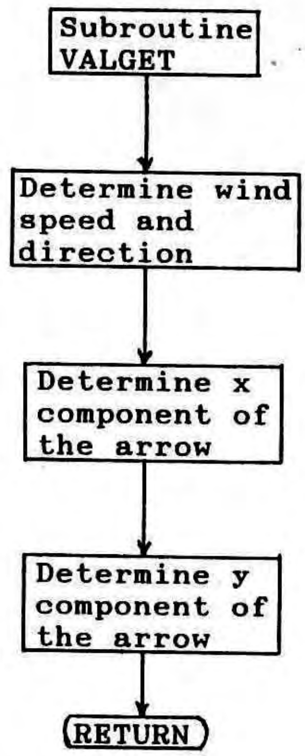


Figure 6.8 Flow chart of the subroutine program VALGET

the function WIND is called by typing CI> WIND ?. The function WIND then executes the program THESD which does the display of wind data on the IIS color monitor. The following subsections describe the operations that are done by the program THESD to display the wind data on the IIS color monitor.

6.4.1 Initiation of the display area

The program THESD does a number of steps as described below

(a) The program initializes various parameters and arrays with proper values.

(b) This program prompts for the type of geometric projection and latitude-longitude area to be selected. Obtaining proper values the program sets up grid patterns on the IIS color monitor by calling any of the subroutines EQGRD, MCGRD, TMGRD, NOVGRD or STGRD as in [7].

(C) A menu is setup on the status plane to assign some of the trackball function buttons with specific functions. This menu is used for the selection of wind data that are to be displayed. The menu is shown in figure 6.9.

(d) TOVS and GTS data files are open and made available to read in. The program also notifies to the video terminal about the date of creation of TOVS and GTS data files and the total number of data present in each file.

(e) The program notifies about the available pressure levels and waits for a particular pressure level to be selected. When a pressure level is selected, the program checks the validity of that pressure level. If the pressure level is '0' the program stops execution and goes to the statement number 999. Otherwise the program cycles through a loop waiting for one of the function buttons to be pressed.

	A	B	C	D	F
1	TOVS press. level wind data	GTS press. level wind data			Change press level (EXIT)
2					toggle map
3		toggle directional plane	Toggle magnitude plane	Toggle graphics plane	

Figure 6.9 Trackball function menu as written on the status plane to display the wind data using the developed program.

(f) As shown in the function menu of figure 6.9, the program, upon pressing of one of the function buttons, executes a particular subroutine except the button 3. When button 3 (F3) is pressed, the program goes back to step (e).

(g) If button 2 (or F2) is pressed, the program calls the subroutine MAPTHS to toggle geographical boundaries to be turned off and on. The geographical boundary may be called by typing CI> W'MAP ? which is a modified IIS image processing function.

(h) If button 7 (or D1) is pressed the graphic planes of previously displayed wind data are turned on by using a subroutine TOVTHS.

(i) If button 12 (A3) or 13 (B3) is pressed the subroutine THESPLT is used to display the wind parameter for TOVS data or GTS data respectively.

(j) When the program reaches statement number 999, it closes all the data files that are opened earlier and stops execution of the program THESD. IIS image processing environment is then returned by displaying the CI> prompt.

6.4.2 Display of wind data

The IIS image processor has eight graphics planes. Of these eight planes, one plane is reserved for the menu to be displayed on the status plane (bit plane number 7). Bit plane number '0' is left to optionally display the WDB map on it. The rest six graphics planes are used by the function WIND (the program THESD) to display the wind data. Bit plane numbers 1 to 6 (graphics planes second to seventh) are used for this display. The program THESD requires two graphics planes at a time to display the wind data. When button number 12 or 13 of the trackball is pushed, the subroutine THESPLT is called by the main program THESD to display the wind data. The subroutine THESPLT, when first called, selects graphics planes 1 and 2. Plane number 1 is color coded in green to display the directional arrows and the plane number 2 is color coded in cyan to display the wind velocity data value. The subroutine then selects TOVS data or the GTS data according to the button numbers 12 or 13 respectively that is pressed.

The subroutine selects the wind data and the latitude and longitude location from the data file and decides the validity of the data location. If the data is within the selected area bounded by the latitudes and longitudes limits decided earlier it is selected. The subroutine THESPLT then calls another subroutine BILIN [18] (available as library subroutine in the computer at SPARRSO) to convert the latitude-longitude location of the wind data to the X-Y location on the graphics plane. The wind data is then displayed on that X-Y location by the use of the subroutine THESPLT and VALGET using the follwing method.

The wind data has two parts, the wind velocity and the wind direction. It is understood, in meteorology, that the wind comes from the direction calculated clockwise from the true north as shown earlier in figure 6.1. To display the wind direction on a graphics plane directional arrows are used. In doing so, the wind direction is at first converted into trigonometrical angle by using the following formula

$$\theta_d = 450^\circ - \theta_s \qquad \ldots . \quad (6.1)$$

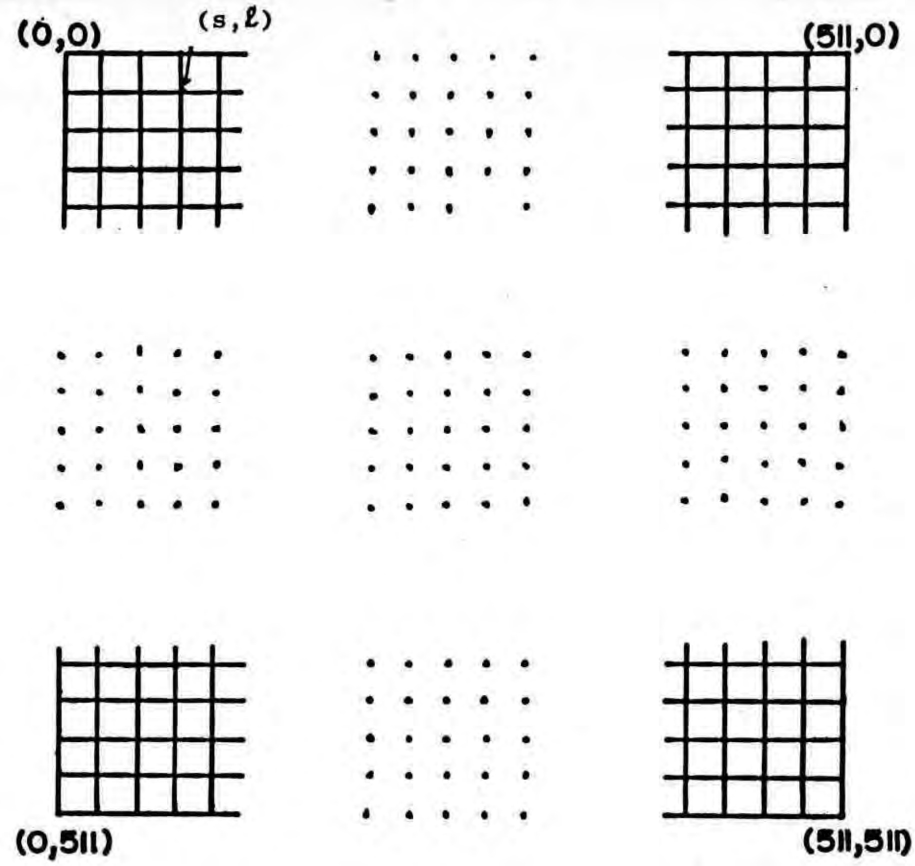
where, θ_s is the angle in degrees obtained to represent the wind direction and θ_d is the angle in degrees calculated to display the wind direction on the IIS monitor. If θ_d becomes greater than 360 degrees than 360° should be subtracted from it to get the smaller angle. This means $\theta_d = \theta_d' - 360$ where θ_d' is the angle which is greater than 360°.

The IIS color monitor has its origin (0,0) at upper left hand corner with samples and lines increasing towards right and below respectively on the screen. This is shown in figure 6.10. Each pixel on the IIS color monitor is considered to have coordinate points denoted by a sample number and a line number (s,l). This is shown in figure 6.11.

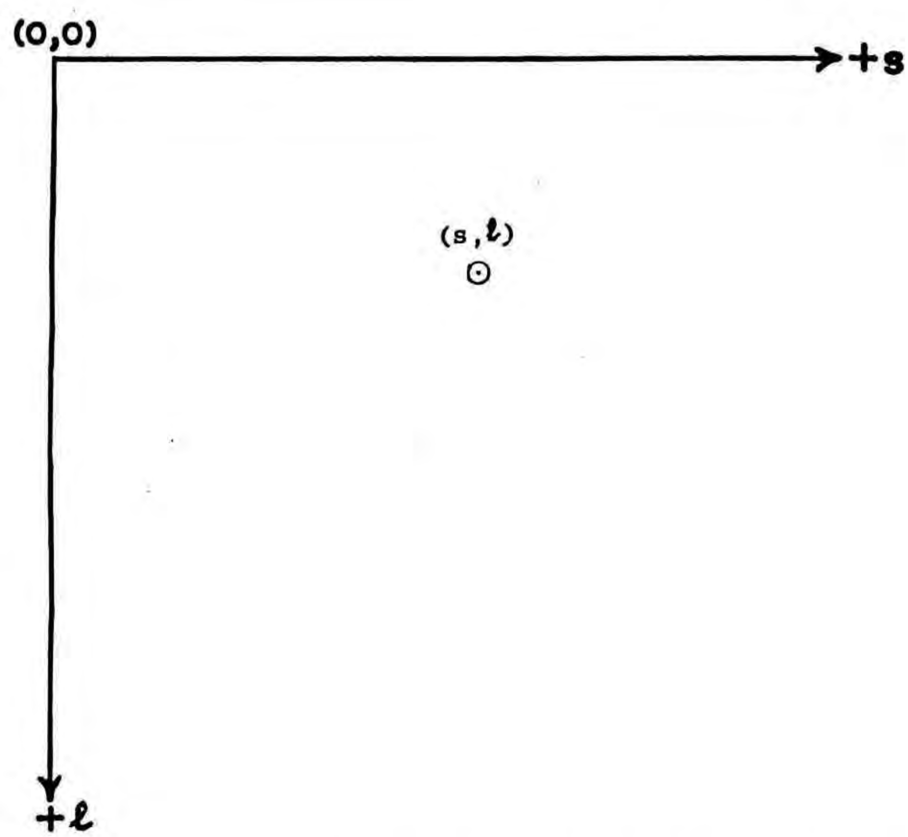
In this work ten pixels are used to draw the body of the directional arrow and five pixels are used to draw each line of the arrow head. The arrow head is drawn with the arrow-tip coinciding with the point of data location calculated from the latitude and the longitude values. The arrow head is formed using two lines drawn at 45° from the body line as shown in the figure 6.12b. The direction of wind data is calculated on the IIS monitor using the relation

$$s_1 = s_0 + A \cos\theta_d \qquad \ldots . \quad (6.2)$$

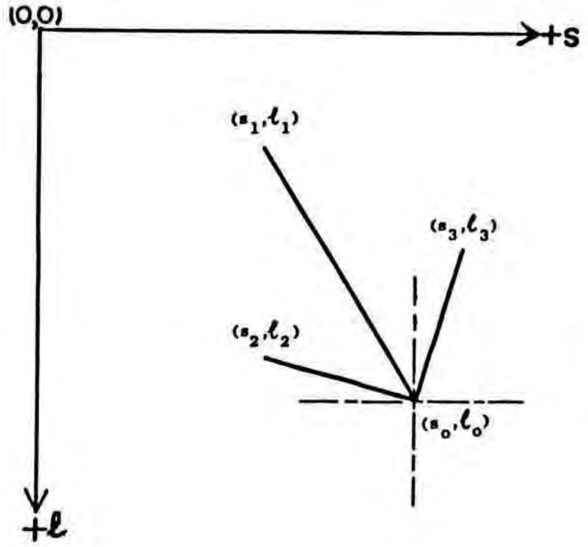
$$l_1 = l_0 - A \sin\theta_d \qquad \ldots . \quad (6.3)$$



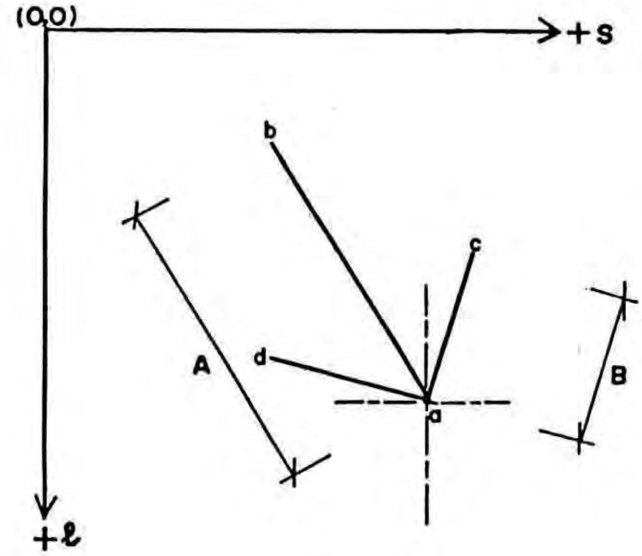
<u>Figure</u> <u>6.10</u> Coordinate system as used on the IIS color monitor.



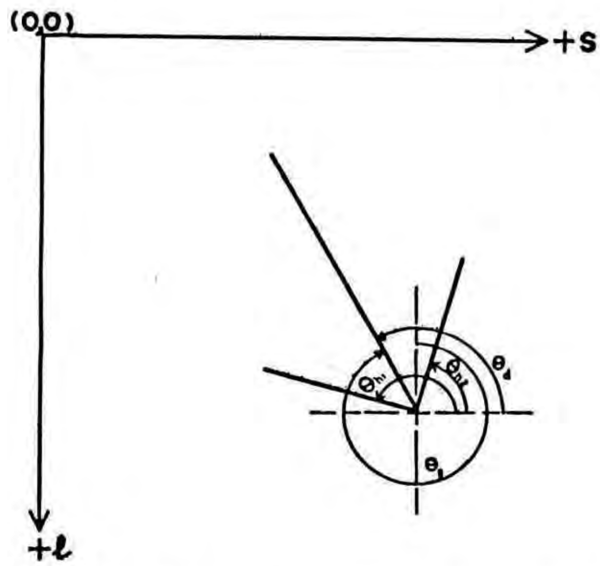
<u>Figure</u> <u>6.11</u> s, l location to represent the pixels on a IIS color monitor.



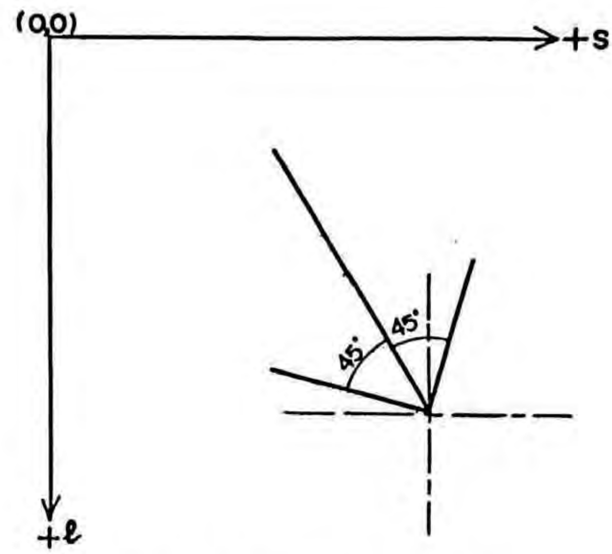
(a) Coordinates of the arrow.



(b) Length of the arrow body = A
Each side of the arrow head = B



(c) Angle conventions. θ_d, θ_{h1}, θ_{h2} +ve angle used in this technique. θ_s +ve angle used by meteorologists.



(d) Each side of the arrow head subtends an angle of 45^o with the body of the arrow.

<u>Figure</u> <u>6.12</u> The technique of displaying wind directional arrows on the IIS color monitor. The arrow body from (s_0,l_0) to (s_1,l_1) has 10 pixels. Each side of the arrow head extends from (s_0,l_0) to (s_2,l_2) (or from (s_1,l_1) to (s_3,l_3)) has 5 pixels.

where, (s_0, l_0) and (s_1, l_1) are the s, l locations, (i.e. the coordinates) on the IIS monitor. (s_0, l_0) represent the coordinates of the arrow tip whilst (s_1, l_1) represent the coordinates of the tail point of the arrow. This means that the body of the arrow extends from (s_1, l_1) to (s_0, l_0). The value for s_0 and l_0 are obtained from the latitude and the longitude location of the wind data after necessary scaling by using the subroutine BILIN. The variable 'A' denotes the required number of pixels for the body of the arrow (here A = 10 pixels).

The arrow heads are calculated from the point of data location by using the formula

$$s_2 = s_0 + B \cos\theta_{h1} \quad \ldots. \quad (6.4)$$

$$l_2 = l_0 - B \sin\theta_{h1} \quad \ldots. \quad (6.5)$$

$$s_3 = s_0 + B \cos\theta_{h2} \quad \ldots. \quad (6.6)$$

$$l_3 = l_0 - B \sin\theta_{h2} \quad \ldots. \quad (6.7)$$

where, s_2, s_3 are the 's' coordinates of the arrow head on the IIS color monitor and is calculated from the point of reference (i.e. s_0). l_2, l_3 are the 'l' coordinates of the arrow head on the IIS color monitor and is calculated from the point of reference(i.e. l_0).

θ_{h1}, θ_{h2} are the angles calculated for the two lines to be drawn to represent the arrow head, where

$$\theta_{h1} = \theta_d - 45 \quad \ldots. \quad (6.8)$$

$$\theta_{h2} = \theta_d + 45 \quad \ldots. \quad (6.9)$$

where, θ_d is the angle obtained from equation (6.1) to represent the wind direction.

The directional arrow is formed by drawing lines from the point of data location to the points represented by the coordinates as obtained by the equations (6.2) - (6.7). An example is shown in figure 6.12.

After the arrow has been displayed on the graphics plane number '1', the wind data value is displayed on the graphics plane number '2' by using the relationships,

$$s_4 = s_0 - 3 \quad \text{for } \theta_d < 180^\circ \quad \ldots\ldots \quad (6.10)$$

$$l_4 = l_0 + 6 \quad \text{for } \theta_d < 180^\circ \quad \ldots\ldots \quad (6.11)$$

$$s_5 = s_0 - 2 \quad \text{for } \theta_d > 180^\circ \quad \ldots\ldots \quad (6.12)$$

$$l_5 = l_0 - 6 \quad \text{for } \theta_d > 180^\circ \quad \ldots\ldots \quad (6.13)$$

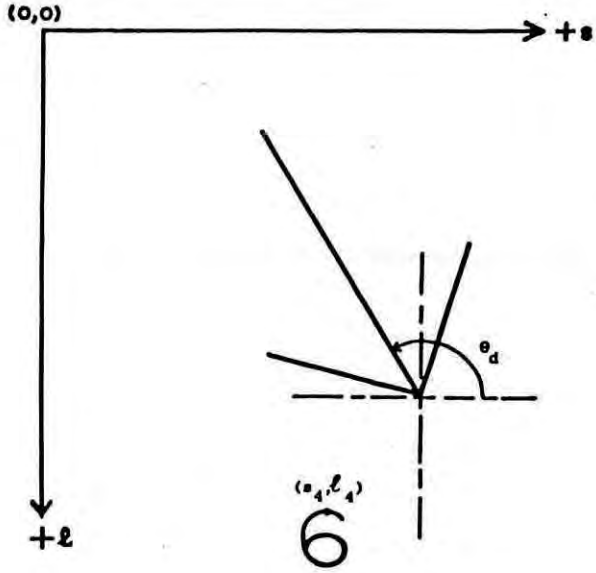
The data values are displayed such that the overlapping of data values and directional arrows do not occur. This is shown in figure 6.13.

After the display of the wind data on the graphics planes 1 and 2, the graphics planes are annotated with the information of type of data (i.e. GIS data or TOVS data), date of data received. The satellite identification code (i.e. NOAA-9 or NOAA-10) is also displayed if TOVS data are used. Figure 6.14 shows the wind data as displayed on the IIS color monitor and obtained from the GTS sources. Figure 6.15 shows the wind data as displayed on the IIS color monitor and obtained from the TOVS sources. After completion of display, the subroutine THESPLT returns the control to the main program THESD.

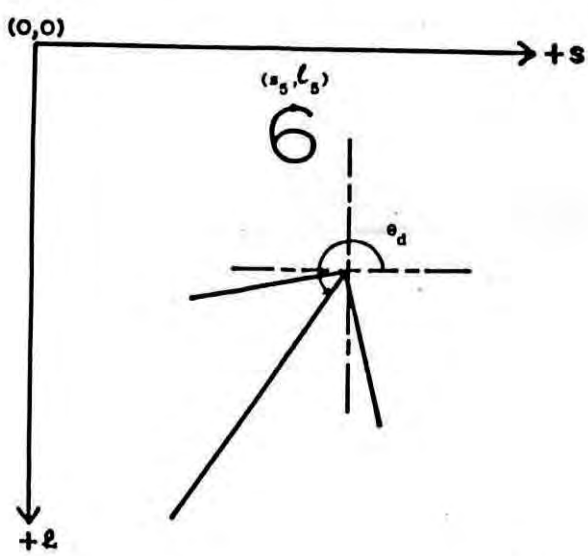
When a second call to THESPLT is made from the program THESD, the subroutine display the wind data and direction on graphics places 3 and 4. Subsequent calls will display the data on graphics planes 5 and 6. If another display is requested, the program THESPLT will display the data on graphics planes 1 and 2 after erasing the previously displayed data.

The following SS75 routines [2] are called by the subroutine THESPLT and VALGET to display and annotate the wind data.

(a) OFGRF To selectively turn off the requested graphics plane.



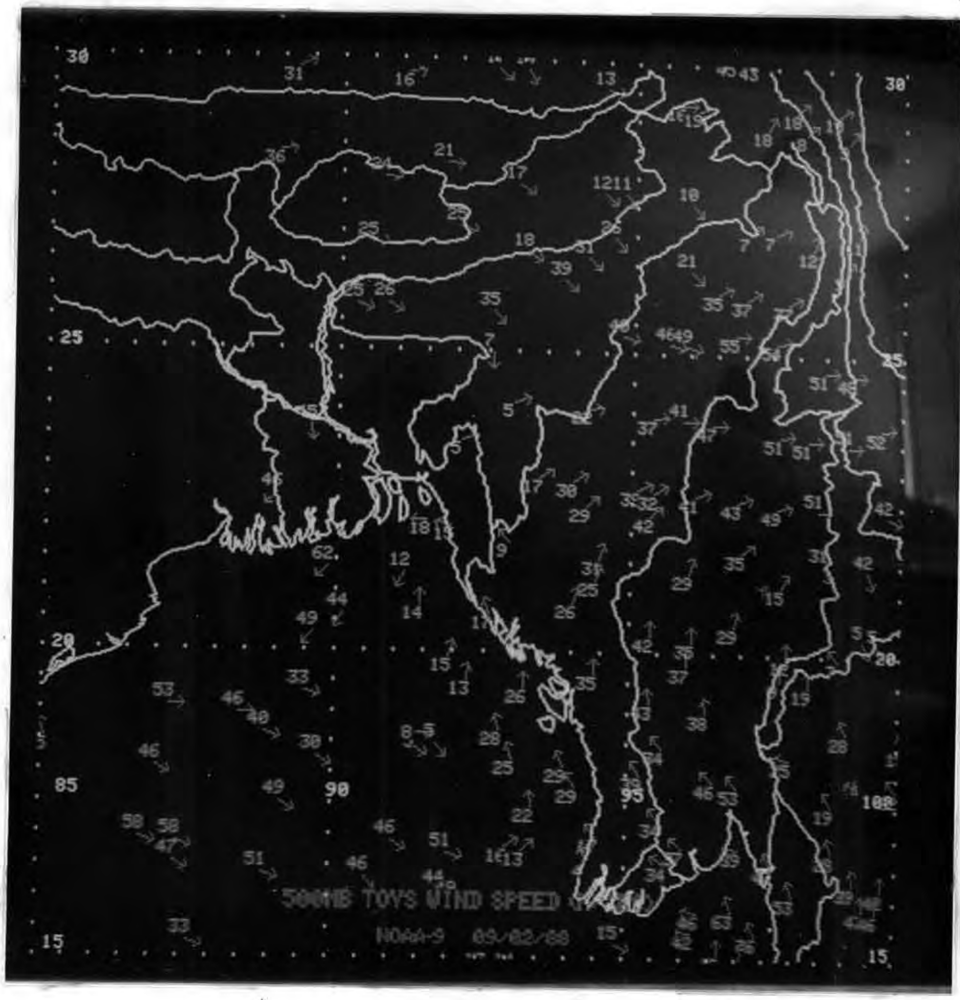
(a) For $\theta_d < 180^\circ$



(b) For $\theta_d > 180^\circ$

Figure 6.13 The technique of displaying wind and direction used in the new method. The magnitude of wind velocity in m/s can be seen near the arrow tip. The arrow direction shows the direction of wind.

Figure 6.14 Display of wind data on the IIS color monitor using the developed software. This display has been obtained using GTS data.



<u>Figure</u> <u>6.15</u> Display of wind data on the IIS color monitor using the developed software. This display has been obtained using TOVS data.

(b) BCHAN To selectively erase or blank the requested graphics plane

(c) CLRPLN To selectively color code the requested graphics plane.

(d) MOVCH To move a string of bytes from one location to another.

(e) BNASC To covert an integer value from one location into an ASCII character and store in another location.

(f) DVECT To connect two x,y coordinate pairs with a vector.

(g) DCHAR To write characters or symbols in a user selected channel or bitplane.

6.4.3. Toggling of the graphics planes

Provisions have been made in the program THESD to toggle certain graphics planes on or off whenever required. Toggling of graphics planes aids viewers to get a better display of wind data. By pressing the trackball function-button numbers 2, 5, 6, and 7 four types of toggling of the graphics planes can be achieved. The effects of pressing the four function-buttons are briefly discussed below.

i) Trackball function-button no. 2: A WDB map can be overlayed on the first graphics plane (i.e., bit plane number '0') of the IIS color monitor. This overlayed map is produced by the IIS funtion W'MAP. The function W'MAP is infact almost same as the funtion MAP of Shaffer [22] except for some modifications in total number of graphics planes and color. This W'MAP function uses only the graphics plane number '1'. The function WIND (program THESD) uses this map to overlay wind data on it to give user a clear idea about the area on which he is working. The map shows the latitude and longitude lines of the area of interest, coastal and land boundaries of the countries that falls within that latitude and longitude limits. Program THESD can selectively turn this map residing in the graphics plane number '1' on or off by the use of trackball function button number 2. A subroutine MAPTHS is used to selectively toggle this map on or off according to the user's demand. This toggling helps one in a better viewing of the wind data.

ii) Trackball function-button nos. 5 and 6: When the trackball function button number '6' is pressed, toggling of the graphics plane containing wind velocity values takes place. The user simply turns on or off the graphics plane containing wind velocity values to take a better view of the wind direction as displayed by the directional arrows. This is achieved by pressing the trackball function-button no. 6. For a better view of the wind velocity values only, one can toggle the graphics plane containing the directional arrows on or off by pressing the functon button number '5'. To toggle the graphics plane containing the wind velocity data value on or off, a subroutine named MGNTHS is called. Another subroutine DRCTHS is called to toggle the graphics plane containing the directional arrows on or off. Figure 6.15 shows the display of wind velocity and directional arrows as overlayed on the WDB map. Figure 6.16 shows the wind data, the WDB map being toggled off. Figure 6.17 shows the display of wind velocity on the WDB map, the directional arrwos being toggled off. Figure 6.18 shows the display of wind directional arrows on the WDB map, the wind velocity being toggled off.

iii) Trackball function-button no. 7: As described in section 6.4.2., two graphics planes are required to display the wind velocity and directional arrows. Provisions are made to use graphics plane numbers two to seven for display of wind data. For the display of first set of wind data graphics planes 2 and 3 are used. Graphics plane numbers 4 and 5 are used to diplay the second set of wind data

Figure 6.16 Display of wind data using the developed technique. The WDB map has been ommitted with the help of trackball function button no. 2.

Figure 6.17 Display of wind velocity on WDB map using the developed technique. Directional arrows have been toggled off with the help of trackball function button no. 5.

<u>Figure</u> <u>6.18</u> Display of wind direction on WDB map using the developed technique. Wind velocity values have been toggled off with the help of trackball function button no. 6.

and for the third set of data graphics plane numbers 6 and 7 are used. If a previously displayed data is required to display agian on the IIS color monitor, a user can simply toggle off the graphics planes showing current wind data and toggles on the last set of data. This is done simply by pressing the trackball function-button number '7'. Toggling of such graphics planes helps in minimising the time required to display the last set of wind data which can also be done by the combined use of function-button numbers '3' and '12' or '13'. Three set of wind data can be stored in 6 graphics plane by the program THESD as described earlier. Hence, an user with the help of function button number 7 can toggle three sets of wind data on or off.

6.5 Summary

Wind velocity along with the directional arrows are displayed on the IIS color monitor. The technique of displaying wind data along with the directional arrows is presented. The structure and arrangement of the previously processed data files which contains the final wind data are described. Some IIS hardware features, which are required for the purpose of displaying the wind data are described. The computer program required to display the wind data is also described.

Chapter 7

Standard operating procedure

7.1 Introduction

"Standard operating procedure (SOP) " of any software package is usually written to facilitate the use of the package without going through the details of the programs of the software package. The SOPs, like [20], aids scientists, researchers and normal users to get the output of a software by simply knowing the purpose of the software and the methods of using it. In the following sections, SOPs are presented so that an user can use the softwares developed in this work and produce the required outputs without any difficulty.

In section 7.2 some of the prompts which appear during running the programs in interactive mode are introduced.

Section 7.3 describes the SOP for decoding of GTS data.

Section 7.4 describes the SOP for preparing a contour map of GPH using GTS or TOVS data.

The SOP of the softwares for displaying wind data value with directional arrows are described in section 7.5

7.2 Prompts for interactive operation

In this work programs are prepared to run in interactive mode. This means, during execution of the programs, two-way exchange of information between the computer and an user terminal takes place. The user is required to answer certain prompts produced by the computer.

In the follwing sections, the characters and words written in bold are prompted by the computer. The answers given following prompts (in bold) are examples according to which the user should answer. The '**$**' character that is used is prompted by the VAX/VMS operating system showing that it is ready to accept any command. The 'CI>' prompt is made by the IIS image processing environment. 'DUA2:IDL>' prompt indicates that access to the IDL account has been made but it is still working under VAX/VMS operating system. 'IDL>' prompt shows access has been made into the IDL environment and it is ready to accept any valid IDL commands.

7.3 SOP for decoding of GTS data

The program MET decodes GTS data as obtained from GTS channels. The program reads the ASCII formatted GTS data from the file MET.ASC and co-relates with the latitude and longitude values of the respective stations by looking at the file LOC.ASC. The program MET converts the coded data into meteorological parameters and stores into the binary file MET.DAT for future use, such as in programs described in chapters 5 and 6.

i) Program syntax

To run the GTS data decoding program MET, type in

$ RUN MET

ii) Check data files

Edit file MET.ASC by typing

$ EDIT MET.ASC

Do necessary editing of the file by entering GTS data according to the program format.

iii) Program operation

The program named MET can be run from any user terminal (e.g. VT100) or any hardcopy terminal (e.g. LA120 Decwriter). It can also be run from any user account. The program is terminated when :

(a) CNTRL-C or CNTRL-Y is pressed from the keyboard.

(b) The computer system or program fails.

iv) Error condition

Generally the errors for the program may arise

(a) due to software errors preventing program execution.

(b) if any of the files MET.ASC or LOC.ASC is not found.

(c) due to inconsistent data type encountered. e.g. the program is reading an integer data but the data is real.

v) Program environment

This program can be executed in any computer system with a CRT terminal and a Fortran-77 compiler. The data files LOC.ASC and MET.ASC should be present in auxiliary memory (e.g. disk devices) at the time of running of this program.

7.4 SOP for contouring geopotential height data

Contouring of GPH data is done on an RGB color monitor. The program GEO creates grid data files with GPH data. The GPH data are read from a preselected latitude and longitude boundary. The output from the program GEO is the grid data files. Program GEO_CONTOUR.PRO reads data from the grid data files and prepares a contour map on the RGB color monitor. The contouring of GPH data is done for an user selected pressure level.

7.4.1 SOP for creating grid data files

The following procedures are to be followed to create grid data files.

i) Logging in to the user account TOVS

The enter key is to be pressed repeatedly to obtain the prompt 'username'. After typing in the user name the password is to be typed in as shown below

USERNAME: TOVS

PASSWORD: SPARRSO ! the password will not be displayed

The prompt '$' will then appear on the screen to indicate that proper access to the account TOVS has been made.

<u>ii) Input data files</u>

Following data files are read from the directory DUA3:[TOVS] from the VAX-11/750 computer at SPARRSO

(i) MET.DAT containing GTS data

(ii) TOVRETn.DAT containing TOVS data, where n is any letter from 'A' to 'Z'.

<u>iii) Program syntax</u>

To run the grid file creation program, type in the following

$ RUN GEO

<u>iv) Prompts to be answered</u>

The following prompts are to be answered accordingly

```
Enter type of data
1 = TOVS, 2 = GTS
1                                   ! here TOVS data has been selected.
Enter TOVS file number to process
2                                   ! 2 defines the character 'B'
Total TOVS records = 574            ! indicating 574 data values can be
                                      accessed for the file TOVRETB.DAT

Enter southern lat., northern lat.,
western long. and eastern long.
10,30,80,100                        ! user selected latitude and
                                      longitude bounderies
```

The program starts processing GPH data for each pressure level and after completion of each pressure level, displays the folowing message to the terminal

Number of data points for index(1) = 213 where index(1) indicates pressure level 1000 mb, index(2) is for pressure level 850 mb and so on. The value 213 is the number of data records that is available within the latitude and longitude boundaries as mentioned earlier by the user. The program goes on processing the GPH data for rest of the pressure levels that are availabe and store the data into the grid data files. The program terminates with the VAX/VMS operating system prompt **$**.

v) Data files created

The program GEO creates the following grid data files in the account DUA3:[TOVS]

(a) GE1000.DAT, GEO850.DAT, GEO700.DAT, GEO500.DAT, GEO400.DAT, GEO300.DAT, GEO250.DAT, GEO200.DAT, AND GEO150.DAT if one wants to create grid data files using TOVS data.

(b) GT1000.DAT, GTS850.DAT, GTS700.DAT, GTS500.DAT, GTS400.DAT, GTS300.DAT, GTS250.DAT, GTS200.DAT, GTS150.DAT, and GTSSRF.DAT if one wants to create grid data files using GTS data.

7.4.2 SOP for creating a contour map

The program GEO_CONTOUR.PRO uses IDL software package to draw isobar lines on a contour map. This map can be displayed on the RGB color monitor by using the IIS hardware features. Graphics video terminals can also be used to display this contour map. With proper setting a plotter can also be used to get a hard copy output.

i) Logging in to the useraccount IDL

The enter key is to be pressed repeatedly to obtain the prompt 'username'. After typing in the user name the password is to be typed in as shown below.

USERNAME: IDL

PASSWORD: SPARRSO ! the password will not be displayed

The prompt 'DUA2:IDL>' will then appear on the screen to indicate that proper access to the account IDL has been made.

ii) Input data files

The data files mentioned in serial number (ii) of subsection 7.4.1 are used by the program GEO_CONTOUR.PRO to create a contour map.

iii) Steps to be followed

The following prompts are to be answered accordingly

DUA2:IDL> XIDL ! to enter into the IDL environment

IDL> RUN GEO_CONTOUR.PRO ! the IDL environment will find the file GEO_CONTOUR.PRO and start compiling and linking the program before executing it. The user should answer to the prompts accordingly similar to the one as shown below.

```
Which type of data file to contour
     1 = TOVS, 2 = GTS
2                                    ! here GTS data is selected
Enter pressure level for GTS data file
1 = 1000 mb,  2 = 850 mb,  3 = 700 mb,  4 = 500 mb
5 =  400 mb,  6 = 300 mb,  7 = 250 mb,  8 = 200 mb
9 =  150 mb, 10 = surface
7                                    ! pressure level 250 mb has been
                                       selected.
Date of data processed
10/1/88                              ! Date is obtained from the first record
                                       of the data file
The latitude range is from  10.00 to 30.00        !  as  mentioned  in
The longitude range is from 80.00 to 100.00       ! the program GEO.
Minimum Geopotential height for this file is = 5280
Maximum Geopotential height for this file is = 5760
```

The program then prompts for the following parameters,

How many isobar lines to be drawn

15 ! maximum 40 lines can be drawn.

Insert minimum isobar line

5300

Insert maximum isobar line

5700 ! the minimum and maximum lines should
! be in the range of 5280 and 5760 as
! mentioned by the program earlier.

The program will calculate the intermediate isobar lines calculating from the minimum and maximum values and total number of isobars to be drawn. It will then draw the contour lines on the RGB color monitor.

7.4.3 Program environment

The program GEO and GEO_CONTOUR are executed to draw a contour map. Program GEO runs in any computer system with at least 2000 records of disk spaces to store the output grid data files. Whereas, the program GEO_CONTOUR.PRO needs an environment which supports IDL programs to be executed. IDL programs vary from systems to systems and can not be copied from one computer site to another. The program also needs an RGB color monitor which can be accessed using an image processor such as IIS. With proper system setup GEO_COTROUR.PRO can also draw contour maps on a graphics terminal or on a plotter.

7.5 SOP for displaying wind data

The IIS function WIND (the computer program THESD) plots wind data value and the directional arrows on the IIS color monitor. The function runs in an IIS image processing environment . Wind data are read from the files MET.DAT (GTS data) and WINRETn.DAT (for TOVS data) where n is any letter 'A' to 'Z'.

i) Logging in to the user account S575

The enter key is to be pressed repeatedly to obtain the prompt 'username'. After typing in the username the password is to be typed as shown below.

USERNAME: S575

PASSWORD: SPARRSO   ! the password will not be displayed

The prompt '$' will then appear on the screen to indicate that proper access to the account S575 has been made.

ii) Input data files

Following data files are read from the directory DUA3:[TOVS] from the VAX-11/750 computer at SPARRSO

(i) MET.DAT containing GTS data

(ii) WINRETn.DAT containing TOVS wind data, where n is any letter from 'A' to 'Z'.

iii) Steps to be followed

After successful log into the user account S575, a '$' prompt will appear. Then the following prompts are to be answered accordingly.

```
$ s tov                          ! to enter into the subdirectory
                                 ! [S575.TOV]
$ edit assign.com                ! edit logical assignmet of files
*c                               ! set screen to full screen mode
ASSIGN DUA3:[TOVS]WINRETB.DAT WIND_DAT    ! logical name assignments.
ASSIGN DUA3:[TOVS]MET.DAT MET_DAT
Press CNTL and 'Z' keys simultaneously and answer to the prompts
*exit
$ @assign.com                    ! execute the assignments of the logical
                                 ! names
$ ci
CI> acquire                      ! initialize and make IIS image
```

```
                                ! processor available
CI> w'map ?                     ! Producing world data base map
int PTYPE = 1 ?                 ! select equirectangular projection.
int ILATS = 15 ? 10             ! south latitude as 10 degrees
int ILATN = 30 ? 30             ! north latitude as 30 degrees
int ILNGE = 95 ? 97             ! east longitude as 97 degrees
int ILNGW = 80 ? 77             ! west longitude as 77 degrees
int GRID SPACING = 5 ? 4        ! draw grid lines at 4 degree intervals
mndt chr *006 CONTINENTS = ? ASIA        ! draw map using ASIA.DAT file

CI> wind ?
int PTYPE = 1 ?                 ! select equirectangular projection.
int ILATS = 15 ? 10             ! south latitude as 10 degrees
int ILATN = 30 ? 30             ! north latitude as 30 degrees
int ILNGE = 95 ? 97             ! east longitude as 97 degrees
int ILNGW = 80 ? 77             ! west longitude as 77 degrees
                                ! same area as the w'map
     ------------------------------
     TOVS data taken on 13/3/88
     Number of TOVS retrievals = 589
     ------------------------------
     GTS taken on 13/3/88
     Number of GTS retrievals  =  59
     Pressure levels (MB):   850, 700, 500, 400,
     ( type 0 to exit )      300, 250, 200, 150
                             100
int  Pressure_level = 500 ? 700    ! prompts for pressure level to be
                                   ! selected. Here 700 mb is
                                   ! selected.
```

The setup is ready for the user for display of wind data. User selects the appropriate function by pressing the trackball function button key from the menu shown in figure 6.9.

7.5.1 Program environment

The function WIND requires an IIS image processing environment to execute the program THESD. The function also needs an IIS image processor and an RGB color monitor to display the wind data.

7.6 Output of data

Section 7.3 to 7.5 describes the SOPs which enable one to (i) decode, process and store GTS data, (ii) prepare a contour map of GPH data and (iii) display of wind data. The output results of these data are made available on the following devices.

a) **RGB color monitor:** Contour maps of GPH values are drawn on the RGB color monitor using different colors. Wind data are also displayed on the monitor. Output to the monitor is very quick and requires not more than 2 minutes for displaying wind data or preparing contour maps of GPH values. The data are stored in the hard disk and can be seen whenever required.

b) **CRT graphics terminal:** With proper setup GPH contour maps can be shown on a CRT graphics terminal. For monochromic terminals at SPARRSO, contour maps can be seen in only one color (in amber color only). A color monitor and a video terminal with keyboard are shown in figure 7.1.

c) **Muirhead:** Display images are saved from the RGB color monitor and stored in the hard disk of VAX-11/750 minicomputer at SPARRSO. This images are printed at the Muirhead output device using the existing software facilities at SPARRSO. The images are printed in black and white and requires approximately 20 minutes to get a hardcopy printout. A sample printout is shown in figure 7.2

d) **Trilog color plotter:** Display images are saved from the graphics planes of the IIS image processor and stored in the hard disk. By using the existing software facilities at SPARRSO these images can be plotted on the trilog color plotter. The time required to form a printable file of an image is approximately 70 minutes. However each colored hardcopy output from this printable file takes 3 minutes. A sample output from the trilog color plotter is shown in figure 7.3.

e) **Photograph:** Color or black and white photographs can be taken directly from the RGB color monitor. The color film processing facility at SPARRSO makes it easier to have a color print within two hours. The color photographs shown in this thesis work are taken from RGB color monitor at ACEMS computer centre of SPARRSO and processed by using the existing color photo processing facilities at the photo laboratory of SPARRSO.

7.7 Summary

SOP for decoding of GTS data is presented. SOPs to prepare a contour map of GPH data and display of wind data along with directional arrows are also presented. The computer programs produces output in a number of devices. The output devices are briefly described.

Figure 7.1 Photograph showing the RGB color monitor, trackball unit, the keyboard and the video terminal used for this work.

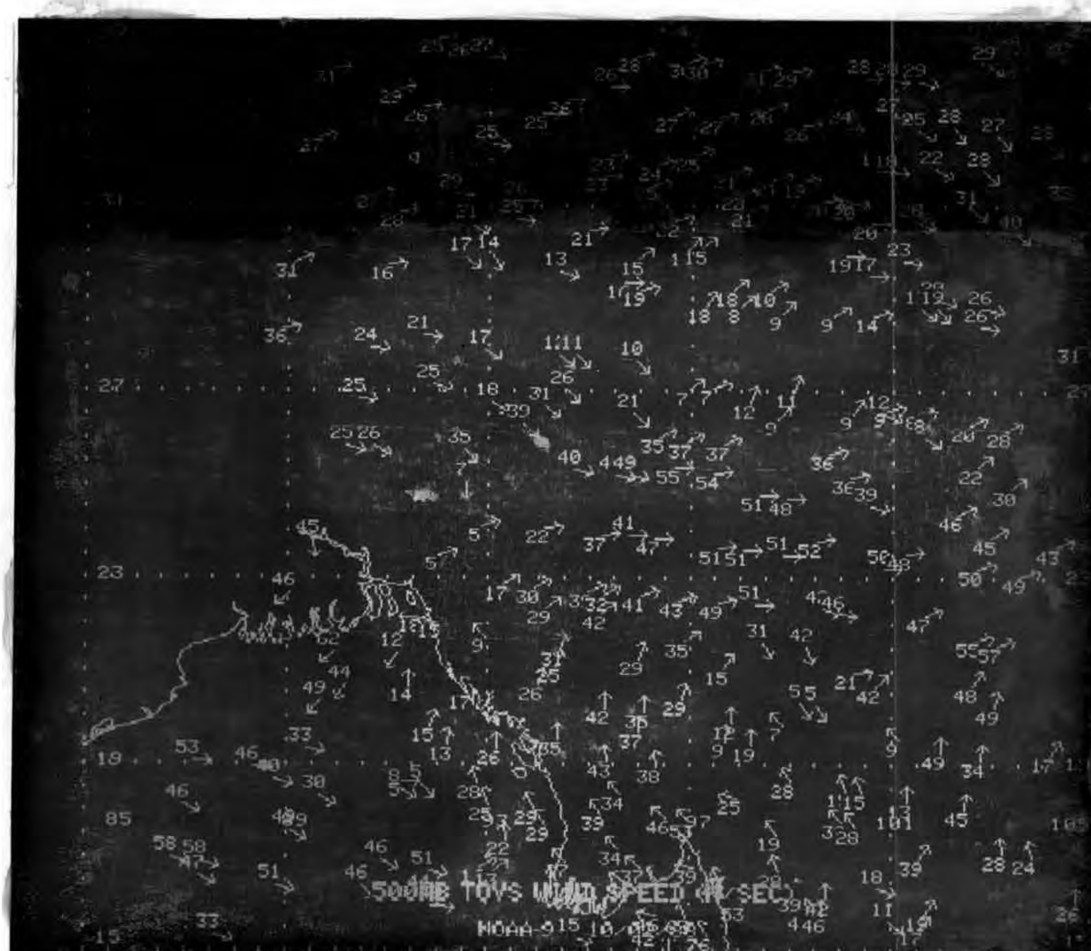


Figure 7.2 A sample of black and white hardcopy output of display obtained from the Muirhead device. The display obtained in a monitor is first saved in an image file in the VAX-11/750 hard disk and then sent to the Muirhead device.

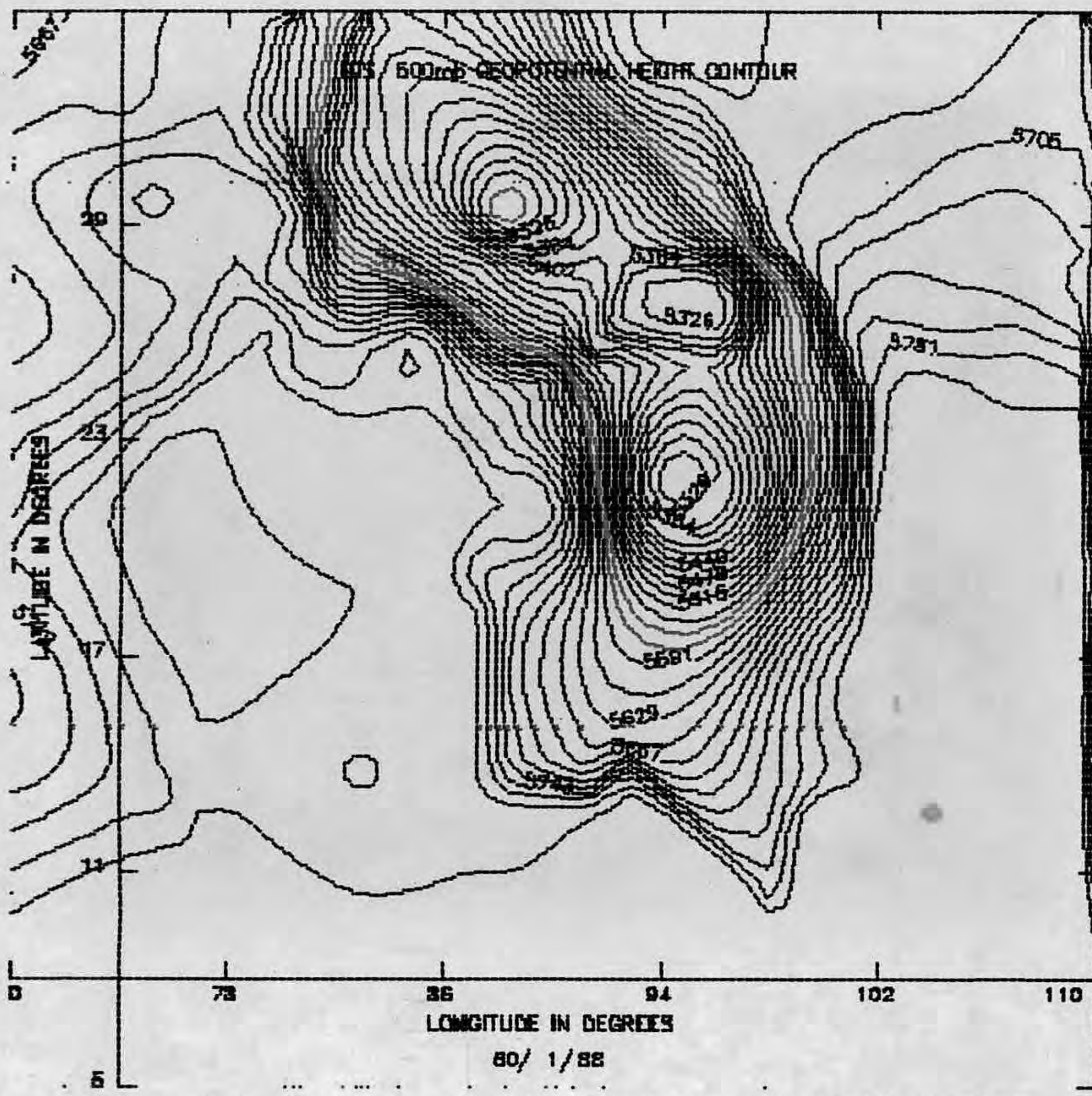


Figure 7.3 A sample of color hardcopy output of display obtained from the Trilog color plotter. The display obtained in a monitor is first saved in an image file in the VAX-11/750 hard disk and then sent to the color plotter.

Chapter 8

Discussions and suggestions for further work

8.1 Discussions

Display of images using information taken from the meteorological satellites and the presentation of meteorological parameters upon these images help weather predictors to save a lot of time and labour in forecasting weather. In this work computer methods have been presented for obtaining useful and informative displays of data received from NOAA satellites and GTS channels.

Developing computer softwares for digital image processing of meteorological parameters is new in Bangladesh. To the best knowledge of the author no significant work has so far been done on this in this country. At present non availability of sufficient literatures and information in Bangladesh imposes difficulty in doing good work in developing new softwares for displaying meteorological parameters. The few literatures available at SPARRSO have been found to be useful in getting ideas related to this type of digital image processing.

It has already been mentioned in chapter 1 that data ingested from NOAA and GMS satellites through ground receiving station at SPARRSO are fed directly into the computer system of SPARRSO. The satellite data are then processed by various software packages. One set of computer programs process and display the AVHRR video data obtained from NOAA satellites and GMS data obtained from GMS satellites. Another set of computer programs process the TOVS data to display the values of air temperature, dew-point temperature, total precipitable water, stability index, geopotential thickness as well as display of the profiles of air temperatures and dew-point temperature.

The existing facilities permit display of these parameters on a color monitor.

However the existing system can not display wind data and prepare contour map of GPH parameter. Besides these it also can not process radiosonde data received in the form of teleprinter printouts at BMD through the GTS channels. The end results of the work presented in this thesis are the computer softwares for obtaining the above mentioned displays.

The first part of this work is for decoding the coded GTS data, for processing and storing in a binary file. Coded data obtained through GTS channels at BMD comes in hardcopy form through teletype machine. No hardware facility is available to fed this data directly into the computer. GTS data can only be entered manually into the computer. The coded information are to be properly understood before entering into the computer. This is needed because missing data on some points or overwritten ambiguous printouts often require to be neglected or extrapolated before manually entering the data into the computer. Enormous manual labour is involved in entering these data into the computer. It has been observed that an operator with good typing speed and proper knowledge of errors in the hardcopy outputs of GTS data needs 8 to 9 hours to enter one set of GTS data manually into the computer.

Using the program developed in this work now one can decode and process the GTS data and can store these data in a binary file. The program has been used in trial basis and found to work satisfactorily. It has been observed that working with binary files is about four times faster than working with ASCII files. For this reason arrangements have been made for storing the processed GTS data in binary form.

Developing computer softwares to obtain contour maps of geopotential height from TOVS and GTS data was the second part of this work. Grid data files were created to rearrange and uniformly distribute the incoming GPH data from TOVS and the GTS sources at all grid points. This distribution is necessary since data obtained from both the sources remains distributed haphazardly without following any definite pattern. The incoming GTS data are distributed sparsely over the area of interest determined by the latitude and longitude limits. One or two data are available in the entire region of the Bay of Bengal. This makes it difficult to create grid data files using GPH data. The program GEO for creating the grid data files uses three closest point to evaluate a grid point GPH data. The closest data for the grid points at the Bay of Bengal region are at the lands. The calculated GPH data values at the grid points still contain some discrepancy.

After discussions with the experts from SSAI USA who visited and worked at SPARRSO it was learned that $1/r^2$ distance method is being used in a number of places outside Bangladesh. However, the same method is also used in this work successfuly. In this method GPH values at each grid point are calculated by using the distance provided by three closest data points from that particular grid point. This technique is described in chapter 6.

In the incoming GPH data sometimes erroneous data are found embedded within the good data. This creates difficulty in the work. However, good results can be obtained by marking the erroneous records of TOVRETn.DAT, and MET.DAT files using flags. These flagging can be done by putting suitable values in the unused fields of the erroneous records. Data of these flagged records will not be taken by the display programs and the grid file creation program by checking the flagged field.

The third part of this work was concerned with development of a computer program called THESD to display wind data on a color monitor. The speciality of this display is that it comprises of a display of longitude and latitude map, a display of wind data and a display of wind direction using arrows. The three part displays are actually superimposed to form a single image on the same screen. Wind parameters and the corresponding latitudes and longitudes obtained from this display have been compared with the corresponding values in the source data file MET.ASC. It has been observed that the program THESD works without any error.

The three softwares described above as well as in this thesis have been installed in the ACEMS computer system at SPARRSO. On some trial operations of these softwares the time required by different programs were recorded. It was observed that the decoding program MET takes around 10 minutes to decode, process and store data for 50 stations obtained through GTS channels. The program GEO requires 15 minutes to create 9 grid data files using TOVS data. The 9 grid data files are created for 9 pressure levels within a boundary of $25^{o}x25^{o}$ latitude and longitude limits. 10 minutes are required to process 10 pressure levels of GTS data for the same area. The program GEO thus requires more time to process TOVS data than the GTS data. This is so because the number of data received through TOVS is five to six times the number of data received through GTS for the same area. Display of contour map using the program GEO_CONTOUR.PRO for a particular pressure level on the RGB color monitor can be done within 2 minutes. For displaying wind data for a particular pressure level on the color monitor using the program THESD and IIS image processing facilities it takes approximately 2 minutes.

8.2 Contribution made in this work

i) A new software has been developed for decoding, processing and storing of GTS meteorological data in the VAX-11/750 computers of SPARRSO.

ii) A new software has been developed for processing the GPH fields using VAX-11/750 computer. The GPH values at chosen grid points are computed which permit plotting a contour map of the GPH values.

iii) Utilising various hardware and software features of the IIS image processor another software has been developed to display the wind parameter by using TOVS and 'GTS data'. The wind data is displayed with directional arrows and numerical values of velocity magnitudes on the same screen.

Using the existing facilities at SPARRSO hard copies of both the displays can be obtained in color as well as in black and white. 'Standard Operating Procedure (SOP)' of three developed softwares have also been prepared.

Using the developed system, contouring of the GPH values is faster and easier as contrasted with the existing system (in Bangladesh) of manually prepared contour maps and also the new software produces accurate contour maps. The wind data is displayed along with the directional arrows. This is a new method of displaying wind data as contrasted with the currently used method of showing the wind direction with rounded off data values (approximate) using symbols.

8.3 Advantages of the new methods

By looking at the new wind data display and the geopotential height contour lines, it is now easier as well as faster to read the weather parameters. This in turn enables the weather forecasters to achieve significant saving in time in weather prediction.

8.4 Suggestions for further work

A number of useful extensions of the work described in this thesis are possible. These are described below.

8.4.1 Display and processing of digitized data

SPARRSO is planning to procure a software called 'Geographic Information System' (GIS) from USA for its existing computer system to create data-base for various geographical information. This data-base is expected to be created using the existing computer facilities and image processing capabilities. The operation of the proposed software in the near future will require a digitizing table consisting of an illuminated table for fixing maps and a moveable electronic head connected to a computer terminal through an interface unit. This digitizing table and the associated hardwares have already been procured and put into operation at the SPARRSO ACEMS computer centre. Using this table, information from maps, charts and drawings can be entered into computer as digital data.

The existing system allows storing the entered digital data in the hard disk of VAX 11/750 and also allows display of these values on a map format on the monochrome graphics video terminals. Unfortunately it is not possible to display these values in the map format on a color monitor screen. Secondly it is required to display the latitude and longitude lines on the same screen similar to the wind data display of the work presented in this thesis. The digitized polygons (used in this method) of the maps, charts and drawings need to be displayed as colored and shaded polygons or line drawings. Last of all, no hardcopy output (i.e. outputs on plotters or printers) can be brought out using the existing softwares.

It is envisaged that these difficulties can be overcome by developing appropriate softwares following the work presented in this thesis. This projection is based on some sample work done in this line. However, further work needs to be done for the development of these type of softwares for display and hardcopy outputs.

8.4.2 New softwares to match the developed software for interfacing computers with the GTS channels

As mentioned earlier, BMD through its teletype machines receives the GTS data through the GTS channels from various stations located in and around Bangladesh. For better and immediate display it is necessary that to bring these data directly into computers instead of manually entering the data from the teletype printouts. It is hoped that soon such an interface will be installed. In fact SPARRSO in its development plan is trying to install one GTS channel in near future. Appropriate software packages will then be required to prepare the data for the softwares developed in this work. Future work in this line will thus produce useful softwares.

8.4.3 Software for other GTS codes

This work deals with only part A of the TEMP code data. However, development of softwares to deal with other parts of TEMP codes and also the SHIPS, PILOTS and SYNOPS codes which are concerned with the surface data, cloud data etc. will be required. A good amount of work can thus be done in future using the developed software as guideline.

8.4.4 Further development of the contouring software

In this work $1/r^2$ distance method is used to prepare contour maps. Other relationships and equations should be tried for improvement. Also efforts should be given to develope softwares for preparing contour maps of wind velocity, air temperatures, dew point temperatures etc. These contour maps would definitely help in producing a better and a complete picture of the above mentioned parameters. Overlaying of meteorological parameters on the contour maps can also be done by developing suitable softwares.

Appendix A

Printout of the program MET

The program 'MET' is used to decode and process the GTS data. The program then outputs the processed data to the binary file MET.DAT. This program was written in Fortran-77. A printout of the program is presented below.

```
c
c ***********************************************************
c *                                                         *
C *    READS GTS DATA & ADJUST THE WEATHER PARAMETER TO THE *
C *    TRUE VALUES &  OUTPUTS TO A BINARY FILE  MET.DAT     *
c *                                                         *
c ********************************************************** *
c
         PROGRAM MET
c
         CHARACTER*7 NAME,NAMEC
         CHARACTER*1 BUF(80)
          INTEGER GEO(11), WIND(11), TEMP(11),  dpd(11)
          INTEGER TIME, DATE(3), STATION,
        . dir(11),buffer(112),dew(11)
          REAL Rlat, Rlong
C
         DATA NU/15/,LENREC/112/,MISG/99999/
         DATA NAME,NAMEC/'MET.DAT','MET.ASC'/
C
          OPEN(NU,FILE=NAME,ACCESS='DIRECT',RECL=LENREC,
        1 STATUS='UNKNOWN')
          open(21, file=NAMEC, form='formatted', status='old')
          read(21,901) date(1),DATE(2),DATE(3), time
901       format(3(i2,1x), i4)
907       FORMAT(I5,1X,I2)
905       FORMAT(80A1,//)
          lloc = 0
802       OPEN(22, FILE='LOC.ASC', FORM='FORMATTED',
        1 STATUS='OLD')
          read(21,907,END=555) station, NDATE
          icnt = station/1000
          ist = station - icnt*1000
          open(22, file='loc.asc', form='formatted',
        1 status='old')
          read(22, 908)buf
908       format(80a1)
801       CONTINUE
          read (22, 903,END=666) ncnt, nst, Rlat, Rlong
          IF(ICNT.EQ.NCNT.AND.IST.EQ.NST) GO TO 701
          GO TO 801
666       READ(21,905) BUF
          CLOSE(22)
          GO TO 802
903       format(i2,1x,i3,2(f6.2))
701       PRINT*,'AREA = ',NCNT, '    STATION = ',NST
          do 552 j=1,11
          GEO(J) = MISG
```

```
              WIND(J) = MISG
              TEMP(J) = MISG
      552     DEW(J) = MISG
              DO 551 J=1,112
      551     BUFFER(J) = MISG
              read(21,902)((geo(II), TEMP(II),IDIR(II), DPD(II),
              1               wind(II)), II=1,11)
      902     format(3(4(2x,i3,1x,I2,I1,I2,1X,i5, 1x)/))
              IF(GEO(1).GT.900) GEO(1) = GEO(1) - 900
              geo(3) = geo(3) + 1000
              geo(5) = geo(5)*10
              geo(6) = geo(6)*10
              geo(7) = geo(7)*10
              geo(8) = geo(8)*10 + 10000
              geo(9) = geo(9)*10 + 10000
              geo(10) = geo(10)*10 + 10000
              geo(11) = geo(11)*10 + 10000
              if(geo(4).ge.900) then
                   geo(4) = geo(4) + 2000
              else
                   geo(4) = geo(4) + 3000
              endif
              DO 121 JK=1,11
              TEMP(JK) = TEMP(JK)*10 + idir(jk)
              IF(TEMP(JK).ge.990) TEMP(JK) = MISG
              DEW(JK) = 99999
              IF(TEMP(JK).EQ.99999) GO TO 121
              IF(MOD(IDIR(JK),2).EQ.1) THEN
                   TEMP(JK) = 0 - TEMP(JK)
              ENDIF
              IF(DPD(JK).GT.55) THEN
                  DEW(JK) = (TEMP(JK) - (FLOAT(DPD(JK))-50)*10)
              ELSEIF (DPD(JK).GE.51.AND.DPD(JK).LE.55) THEN
                  DEW(JK) = MISG
              ELSE
                   DEW(JK) = (TEMP(JK) - FLOAT(DPD(JK)))
              ENDIF
      C
      C  FOR DATE 1-31 , WIND VALUE IS EXPRESSED IN Meters/sec
      c  for date 51-81, WIND VALUE IS EXPRESSED IN Knots
      c  (Nautical miles) change knots into meters/sec
      c
              if(wind(jk).eq.99999) go to 121
              if (Ndate.gt.50) then
              m2wind = mod(wind(jk),1000)
              mwind = mod(m2wind,500)/2.0
              ndir =  (wind(jk) - mod(m2wind,500))/100
              wind(jk) = mwind + ndir*1000
              else
              m2wind = mod(wind(jk),1000)
              mwind = mod(m2wind,500)
              ndir =  (wind(jk) - mod(m2wind,500))/100
              wind(jk) = mwind + ndir*1000
              endif
      121     CONTINUE
              write(6,922)
      922     format(1x,///,1xt5,' Geo Poten. Hght', t25,
```

```
          .  'Temperature', t40,
          1  'Dew Point Temp.' , t60, 'Wind parameter'/)
924         format(1x, t8, i8, t25, f11.3, t42, f10.3, t62, i8)
            DO 122 I=1,11
            write(6,924)(geo(I), float(TEMP(I))/10.0,
            1       float(DEW(I))/10.0, wind(I))
122         CONTINUE
            lloc = lloc + 1
            BUFFER(1) = RLAT*100          BUFFER(2) = RLONG*100
            BUFFER(3) = ICNT
            BUFFER(4) = IST
            DO 554 JJ=1,11
            BUFFER(JJ+4) = GEO(JJ)
            BUFFER(JJ+19) = TEMP(JJ)
            BUFFER(JJ+34) = DEW(JJ)
            BUFFER(JJ+80) = WIND(JJ)
554         CONTINUE
            NREC = LLOC + 1
            WRITE(NU,REC=NREC)BUFFER
            CLOSE(22)
            go to 802
555         print*,'station located = ',lloc
            DO 557 K=1,112
557         BUFFER(K) =999
            BUFFER(1) = DATE(3)*10000 + DATE(2)*100 + DATE(1)
            BUFFER(2) = TIME
            BUFFER(112) = LLOC
            WRITE(NU,REC=1) BUFFER
            stop
            end
```

Appendix B

Arrangement of data in the data file MET.DAT

The arrangement of data in the data file MET.DAT as taken from an example is presented below. Record number 1 contains header information. Record number 1 as well as record number 2 contains 112 fields.

Record number 1

880110	0	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	999	999	999
999	999	999	39		

Note 1:
Field(1) = Date as YYMMDD
Field(2) = Time of the day in GMT.
Field(112) = Data received for the total number of stations.
999 = Kept for future use.

Record number 2

3148	9207	55	299	583	999
1999	2999	5700	7350	9360	10570
12030	13890	16490	99999	99999	99999
99999	-93	99999	99999	99999	-115
-275	-427	-503	-495	-567	-695
99999	99999	99999	99999	-263	99999
99999	99999	-285	-415	-557	-613
-615	-687	-1185	99999	99999	99999
99999	99999	99999	99999	99999	99999
99999	99999	99999	99999	99999	99999
99999	99999	99999	99999	99999	99999
99999	99999	99999	99999	99999	99999
99999	99999	99999	99999	99999	99999
99999	99999	0	99999	99999	99999
260030	260029	270039	265046	265065	265055
260036	99999	99999	99999	99999	99999
99999	99999	99999	99999	99999	99999
99999	99999	99999	99999	99999	99999
99999	99999	99999	99999		

Note 2:

Field(1) = Latitude of the station times 100.

Field(2) = Longitude of the station times 100.

Field(3) = Area code.

Field(4) = Station code.

Field(5) to field(15) = Geopotential height in meters for different pressure levels.

Field(20) to field(30) = Air temperature in degrees celcius times 10 for different pressure levels.

Field(35) to field(45) = Dew point temperature in degrees celcius times 10 for different pressure levels.

Field(81) to field(91) = Wind velocity + wind direction*1000 for different pressure levels .

99999 = Data missing or is kept for future use.

APPENDIX C

Printout of the program GEO

The Computer program 'GEO' is used to create grid data files using GPH parameter of GTS data or the TOVS data as input. This program is written in Fortran-77. A printout of the program is presented below.

```
c|-------------------------------------------------------------|
        PROGRAM GEO
c|-------------------------------------------------------------|
c
        integer    ibuf (112), ptype, mrec
        real       lats, latn, lnge, lngw, lat, lng
c
        common / grid / gridx(181,181), gridy(181,181)
        common / bound/ lats, latn, lnge, lngw
        common / POINT/ NFILE
C
C    prompt user for type of data TOVS or Radiosonde(GTS)
C
40        print*, ' '
          print*,'enter type of data '
          print*,' '
          print*,'1 = TOVS ;  2 = GTS'
          ACCEPT*,NDATA
c
c
c    prompt user for projection and area
c
          if(NDATA.EQ.1) THEN
               PRINT*,' '
               PRINT*,'ENTER TOVS FILE NO. TO PROCESS'
               PRINT*,' '
               ACCEPT*,NFILE
c
c    open TOVS file
c
               call retio (ibuf, 0, 0)
               mrec  = ibuf(112)
          PRINT*, 'Total TOVS records = ',mrec
          ELSEIF(NDATA.EQ.2) THEN
          CALL METIO ( IBUF, 0, 0)
          MREC = IBUF(112)
          PRINT*, 'Total GTS records = ',mrec
          ELSE
          print*,' You should enter 1 or 2 '
          GO TO 40
          ENDIF
50         ptype = 1
             print *,' '
          print *,'Enter southern lat., northern lat.,'
          print *,'      western lon., eastern lon.:'
             read  *,  lats, latn, lngw, lnge
             print *,' '
c
```

```
c   set up grid pattern, eqrectangular
c
      call eqgrd (lats, latn, lngw, lnge)
      call read_GEO (ibuf, mrec, ndata)
c
c   close GEO file and terminate
c
  999 continue
      call retio (ibuf, 0, -1)
      call metio (ibuf, 0, -1)
      return
      end

c|---------------------------------------------------------------|
      subroutine read_GEO (ibuf, mrec, ndata)
c|---------------------------------------------------------------|
c
      integer         ibuf(1), mrec
c
      integer         val, ix, iy
      real        lats, latn, lnge, lngw, lat, lng
      real        MONTH
c
      integer         npts, idim, jdim, iopt, wop(3)
      real        xdat(2000), ydat(2000), zdat(2000)
      real        z(64,64), sfac, xs(2000), ys(2000)
      real        data(2000,3), r2(2000)
       CHARACTER*6 FILE(9),GTSFILE(11)
       DATA FILE/'GE1000','GEO850','GEO700','GEO500',
       1         'GEO400','GEO300','GEO250','GEO200',
       2         'GEO150'/
       DATA GTSFILE/'GTSSRF','GT1000','GTS850','GTS700',
       1            'GTS500','GTS400','GTS300','GTS250',
       2            'GTS200','GTS150','GTS100'/
c
      common / bound/ lats, latn, lnge, lngw
c
c
       IF(NDATA.EQ.1) THEN
       MN = 9
       ELSE
       MN = 11
       ENDIF
       DO 77 INDEX=1,MN
       IF(NDATA.EQ.1) THEN
         OPEN (UNIT=22, NAME=FILE(INDEX), recordtype='fixed',
     .        organization='sequential',access='direct',
     .        FORM='UNFORMATTED', RECL=128, STATUS='UNKNOWN')
       ELSE
      OPEN (UNIT=22, NAME=GTSFILE(INDEX), recordtype='fixed',
     .        organization='sequential',access='direct',
     .        FORM='UNFORMATTED', RECL=128, STATUS='UNKNOWN')
       ENDIF
      npts = 0
       KINDEX = INDEX
       if (ndata.eq.1) then
           call retio(ibuf, 0, 0)
```

```
            else
                call metio(ibuf, 0, 0)
            endif
                  call nyyddd (ibuf(1), date, month, year)
c
           do 40  irec = 1, mrec
            if (ndata.eq.1) then
                call retio(ibuf, irec, 0)
            else
                call metio(ibuf, irec, 0)
            endif
              lat     = floatj(ibuf(1))   / 100.0
              lng     = floatj(ibuf(2))   / 100.0
              if (lat .ge. lats  .and.  lat .le. latn  .and.
     *           lng .ge. lngw  .and.  lng .le. lnge) then
            call bilin (latn, lats, lnge, lngw, lat, lng, ix, iy)
                 if (ibuf(102) .ne. 777) then
            if (ndata.eq.1) then
                    val = ibuf(index+4)
            else
                       val = ibuf(index+4)
            endif
            IF(VAL.EQ.99999) GO TO 40
c
                     npts = npts + 1
                     xdat(npts) = ix
                     ydat(npts) = 512 - iy
                     zdat(npts) = val
                 endif
              endif
 40         continue
           print *,' '
           print *,'number of raw data points FOR INDEX(',
            1    NDEX,')', npts
           print *,' '
           idim   = 64
           jdim   = 64
           iopt   = 2
           wop(1) = 0
           wop(2) = 0
           wop(3) = 0
           sfac   = 0.5
C
            CALL FINDM(ZDAT,WMIN,WMAX,NPTS)
           call gridit (npts, xdat, ydat, zdat, idim,
     .                 jdim, z, iopt,
     .                 wop, sfac, xs, ys, data, r2)
C
           do  i = 2, 65
              write (22,REC=I) (z(i,j), j=1,64)
           end do
            WRITE(22,REC=1) LATS,LATN,LNGW,LNGE,WMIN,WMAX,
            1              date, month, year, float(ndata)
           close (unit=22)
 77         CONTINUE
           return
           end
```

Appendix D

Printout of the program GEO CONTOUR.PRO

The program "GEO_CONTOUR.PRO" is used to prepare a contour map using data from a grid data file. The grid data files are created by the program GEO as shown in Appendix C. The program GEO_CONTOUR.PRO was written in IDL computer language. A printout of the program is presented below.

```
;   CONTOUR Geo Potential Height data USING IDL
;   ======================================
    !ORDER = 1
    set_plot,1
    A=FLTARR(64,64)
    B=FLTARR(128)
    noeras=-1
    level = fltarr(60) ;
    GOTO,TRY
BAD: PRINT,' '
     PRINT,'DON''T ENTER GARBAGE VALUE' TRY:    PRINT,' '
     PRINT,' Which type of data file to contour'
     print,'     1 = TOVS,   2 = GTS'
    read,ntype
     case ntype of
     1 : goto,strtov
     2 : goto,strgts
     else : goto,bad
        endcase
badgts:  print,'Enter proper value '
strgts:  print,'Enter pressure level for GTS data file '
      print,' '
      print,'  1 = 1000 mb , 2 = 850 mb , 3 = 700 mb , 4 = 500 mb'
      print,'  5 =  400 mb , 6 = 300 mb , 7 = 250 mb , 8 = 200 mb'
      print,'  9 =  150 mb , 11 = SURFACE '
     read, ipr
;
;  open gts data file ;
    case ipr of
     1 : OPENR,2,'DUA3:[TOVS]gt1000.DAT/UNFORMATTED/sequential/fixed'
     2 : OPENR,2,'DUA3:[TOVS]gts850.DAT/UNFORMATTED/sequential/fixed'
     3 : OPENR,2,'DUA3:[TOVS]gts700.DAT/UNFORMATTED/sequential/fixed'
     4 : OPENR,2,'DUA3:[TOVS]gts500.DAT/UNFORMATTED/sequential/fixed'
     5 : OPENR,2,'DUA3:[TOVS]gts400.DAT/UNFORMATTED/sequential/fixed'
     6 : OPENR,2,'DUA3:[TOVS]gts300.DAT/UNFORMATTED/sequential/fixed'
     7 : OPENR,2,'DUA3:[TOVS]gts250.DAT/UNFORMATTED/sequential/fixed'
     8 : OPENR,2,'DUA3:[TOVS]gts200.DAT/UNFORMATTED/sequential/fixed'
     9 : OPENR,2,'DUA3:[TOVS]gts150.DAT/UNFORMATTED/sequential/fixed'
;    10 : OPENR,2,'DUA3:[TOVS]gts100.DAT/UNFORMATTED/sequential/fixed'
    11 : OPENR,2,'DUA3:[TOVS]gtssrf.DAT/UNFORMATTED/sequential/fixed'
    else : goto,badgts
   endcase
    goto,start
badtov:  print,'Enter proper value '
strtov:  print,'Enter pressure level for TOVS data file '
      print,' '
      print,'  1 = 1000 mb , 2 = 850 mb , 3 = 700 mb , 4 = 500 mb'
      print,'  5 =  400 mb , 6 = 300 mb , 7 = 250 mb , 8 = 200 mb'
```

```
   print,'  9 =  150 mb , 10 = 100 mb                    '
  read, ipr
;  open TOVS data file
   case ipr of
    1 : OPENR,2,'DUA3:[TOVS]ge1000.DAT/UNFORMATTED/sequential/fixed'
    2 : OPENR,2,'DUA3:[TOVS]geo850.DAT/UNFORMATTED/sequential/fixed'
    3 : OPENR,2,'DUA3:[TOVS]geo700.DAT/UNFORMATTED/sequential/fixed'
    4 : OPENR,2,'DUA3:[TOVS]geo500.DAT/UNFORMATTED/sequential/fixed'
    5 : OPENR,2,'DUA3:[TOVS]geo400.DAT/UNFORMATTED/sequential/fixed'
    6 : OPENR,2,'DUA3:[TOVS]geo300.DAT/UNFORMATTED/sequential/fixed'
    7 : OPENR,2,'DUA3:[TOVS]geo250.DAT/UNFORMATTED/sequential/fixed'
    8 : OPENR,2,'DUA3:[TOVS]geo200.DAT/UNFORMATTED/sequential/fixed'
    9 : OPENR,2,'DUA3:[TOVS]geo150.DAT/UNFORMATTED/sequential/fixed'
   else : goto,badtov
  endcase start:
   I = 1
   forrd,2,b
   M1 = FIX(B(6))
   M2 = FIX(B(7))
   M3 = FIX(B(8))
   DTYPE = fix(B(9))
   PRINT,' '
   PRINT,"$(1X,'DATE OF DATA PROCESSED')"
   PRINT,"$(1X, I2,'/',I2,'/',I2)",M1, M2, M3
   PRINT,' '
   PRINT,'the latitude range is from',b(0),'to ',b(1)
   xmin = b(0)
   xmax = b(1)
   PRINT,' '
   PRINT,'the longitude range is from',b(2),'to ',b(3)
   ymin = b(2)
   ymax = b(3)
   PRINT,' '
   print,'minimum GEOPETENTIAL HEIGHT for this file is = ',b(4)
   print,'maximum GEOPETENTIAL HEIGHT for this file is = ',b(5)
   PRINT,' '
   PRINT,'HOW MANY LEVELS'
   READ,N
   PRINT,'INSERT MINIMUM LEVEL'
   READ,X
   LEVEL(0) = X
   PRINT,'INSERT MAXIMUM LEVEL'
   READ,Y
   LEVEL(N-1) = Y
   if (n eq 1) then BEGIN
     DELTA = LEVEL(0)
     goto,skip
   ENDIF
    DELTA = (LEVEL(N-1)-LEVEL(0))/(N-1)
   FOR I=1,N-2 DO LEVEL(I) = LEVEL(I-1)+ DELTA
   FOR I=1,N DO PRINT,'THE LEVELS ARE = ',LEVEL(I-1)
    PRINT,' DEFINED ARRAY AND LEVELS'
;
;  READ N, LON, LAT, PRESSURE
;  -------------------------
     FOR II=0,63 DO BEGIN
       FORRD,2,B
```

```
        for jj=0,63 do a(ii,jj) = b(jj)
;     PRINT,B ; PRINT,A(*,ii)
     ENDFOR
   CLOSE,2
;
;    define titles
;    -------------
      k = 0
      !fancy=1
      !type=108
   set_viewport,0.0,1.0,0.0,1.0
      !bcolor=8
      Contour,a,level(0) ,DELTA ,N
      GOTO,FINI SKIP:  CONTOUR,A FINI:  wait,3
   set_viewport,0.1,1.0,0.0,1.0
      !COLOR=5
      !type=28
      axis,Xmin,Xmax,0,'!5LATITUDE IN DEGREES'
   set_viewport,0.0,1.0,0.1,1.0
      axis,Ymin,Ymax,4,'!5LONGITUDE IN DEGREES' ;
   set_viewport,0.0,1.0,0.0,1.0
   !fancy = 1
      case DTYPE of
      1 : goto,tovant
      2 : goto,gtsant
      endcase gtsant: case ipr of
   1 : tvOUT,150,480,'!3 GTS 1000mb GEOPOTENTIAL HEIGHT CONTOUR',1
   2 : tvOUT,150,480,'!3 GTS  850mb GEOPOTENTIAL HEIGHT CONTOUR',1
   3 : tvOUT,150,480,'!3 GTS  700mb GEOPOTENTIAL HEIGHT CONTOUR',1
   4 : tvOUT,150,480,'!3 GTS  500mb GEOPOTENTIAL HEIGHT CONTOUR',1
   5 : tvOUT,150,480,'!3 GTS  400mb GEOPOTENTIAL HEIGHT CONTOUR',1
   6 : tvOUT,150,480,'!3 GTS  300mb GEOPOTENTIAL HEIGHT CONTOUR',1
   7 : tvOUT,150,480,'!3 GTS  250mb GEOPOTENTIAL HEIGHT CONTOUR',1
   8 : tvOUT,150,480,'!3 GTS  200mb GEOPOTENTIAL HEIGHT CONTOUR',1
   9 : tvOUT,150,480,'!3 GTS  150mb GEOPOTENTIAL HEIGHT CONTOUR',1
   11 : tvOUT,150,480,'!3 GTS  SURFACE GEOPOTENTIAL HEIGHT CONTOUR',1
   ENDCASE
   goto,date tovant: case ipr of
   1 : tvOUT,150,480,'!3 TOVS 1000mb GEOPOTENTIAL HEIGHT CONTOUR',1
   2 : tvOUT,150,480,'!3 TOVS  850mb GEOPOTENTIAL HEIGHT CONTOUR',1
   3 : tvOUT,150,480,'!3 TOVS  700mb GEOPOTENTIAL HEIGHT CONTOUR',1
   4 : tvOUT,150,480,'!3 TOVS  500mb GEOPOTENTIAL HEIGHT CONTOUR',1
   5 : tvOUT,150,480,'!3 TOVS  400mb GEOPOTENTIAL HEIGHT CONTOUR',1
   6 : tvOUT,150,480,'!3 TOVS  300mb GEOPOTENTIAL HEIGHT CONTOUR',1
   7 : tvOUT,150,480,'!3 TOVS  250mb GEOPOTENTIAL HEIGHT CONTOUR',1
   8 : tvOUT,150,480,'!3 TOVS  200mb GEOPOTENTIAL HEIGHT CONTOUR',1
   9 : tvOUT,150,480,'!3 TOVS  150mb GEOPOTENTIAL HEIGHT CONTOUR',1
   ENDCASE DATE:   tvOUT,200,10,M1,1
   M5 = '/'
   tvOUT,240,10,M5,1
   tvOUT,220,10,M2,1
   tvOUT,260,10,M5,1
   tvOUT,240,10,M3,1 ;
      END
```

APPENDIX E

Printout of the program THESD

The program "THESD" is used to display of wind data on the IIS color monitor using the GTS data or the TOVS data. The program is written in Fortran-77. A printout of the program is presented below.

```
c*..............................................................*
    SUBROUTINE thesd (FCB, NIDS, NODS, BUFFER, REALBF, BUFSIZ)
c*..............................................................*
c*                                                              *
c*  There are no inputs or outputs for the THES program.        *
c*                                                              *
c*                   \ wind ? \                                 *
c*                                                              *
c*     Available pressure levels:                               *
c*                850, 700, 500, 400, 300, 250, 200,            *
c*                150, 100 (mb)                                 *
c*                                                              *
c*..............................................................*
c*                                                              *
c*   program author:   A.R.AZIMUL HOQUE, S.E.,SPARRSO,DHAKA     *
c*                                                              *
c*..............................................................*
c*                                                              *
c*..............................................................*
c
      INTEGER  FCB(1), BUFFER(1)
      INTEGER  NIDS, NODS, BUFSIZ
      REAL     REALBF(1)
c
      integer    ptype, ilats, ilatn, ilnge, ilngw, parms(75)
      integer    pres, val(3)
      integer    limg, lmap, ltov, lbut1, lbut2,lbut3
      integer    font(2048), char(10), ipr(9), mrec, index
      integer*4  ibuf(112)
      real       lats, latn, lnge, lngw, lat, lng
c
      common / grid / gridx(181,181), gridy(181,181)
      common / bound/ lats, latn, lnge, lngw
      common / tog  / limg, lmap, lmgn, ldrc, ltov, lbut1,
     1       lbut2, lbut3
      common / disp / font, char, ipr, mrec, mrecgts, index
      common / mdate / ndate, nmonth, nyear, itime, idsat
c
c   initialize variables:  set up 9 pressure levels
c                          set to toggle switches
c
      data       ipr / 850, 700, 500, 400, 300, 250, 200, 150,
     *                 100/
c
      lmap = 1
      lMGN = 1
      lDRC = 1
      ltov = 1
      lbut1 = 0
```

```
c
c   check inputs and outputs
c
      if (nids .ne. 0  .and.  nods .ne. 0) then
      write (6,*) '*** WINDS EXPECTS NO INPUTS OR OUTPUTS ***'
         go to 999
      endif
c
c   blank graphic planes #1 to #6
c
      call bchan (fcb, buffer, -32768, 126)
c
c   promt user for projection and area
c
      call movch (16hPTYPE            , buffer, 0, 0, 16)
      call parmi (fcb, parms, ptype, buffer, 0, 1, 1, 1)
      ptype = 1
c
      call movch (16hILATS            , buffer, 0, 0, 16)
      call parmi (fcb, parms, ilats, buffer, 0, 1, 1, 2)
      ilats = 15
c
      call movch (16hILATN            , buffer, 0, 0, 16)
      call parmi (fcb, parms, ilatn, buffer, 0, 1, 1, 3)
      ilatn = 30
c
      call movch (16hILNGE            , buffer, 0, 0, 16)
      call parmi (fcb, parms, ilnge, buffer, 0, 1, 1, 4)
      ilnge = 95
c
      call movch (16hILNGW            , buffer, 0, 0, 16)
      call parmi (fcb, parms, ilngw, buffer, 0, 1, 1, 5)
      ilngw = 80
c
      call define (fcb, parms, 5, 0)
      write (6,*)
c
c   convert boundaries to real*4
c
      lats = floati(ilats)
      latn = floati(ilatn)
      lnge = floati(ilnge)
      lngw = floati(ilngw)
c
c   set up grid pattern (eqrectangular, mercator, transverse
c                 mercator, gnmonic, or polar steriographic)
c
      if (ptype .eq. 1) call eqgrd (lats, latn, lngw, lnge)
c
c   set up main menu
c
      call thmnu (fcb, buffer)
c
c   open TOVS WINDS data file
c
      call thsio (ibuf, 0, 0)
      call nyyddd (ibuf(1),ndate,nmonth,nyear)
```

```
          IDSAT = IBUF(9)
          ITIME = IBUF(2)
         PRINT*,' '
         PRINT*,' '
         WRITE(6,92576)
         WRITE(6,92567)NDATE,NMONTH,NYEAR
         WRITE(6,92576)
92576 FORMAT(' ',T19,'-----------')
92567 FORMAT(' TOVS DATA TAKEN ON   ',I2,
     .        '/',I2,'/',I2)
         mrec  = ibuf(111)
         print*, 'number of wind retrievals ',mrec
c
c   open GTS WINDS data file
c
         call metio (ibuf, 0, 0)
         call nyyddd (ibuf(1),ndate,nmonth,nyear)
         PRINT*,' '
         WRITE(6,92576)
         WRITE(6,92569)NDATE,NMONTH,NYEAR
         WRITE(6,92576)
92569 FORMAT(' GTS DATA TAKEN ON   ',I2,
     .        '/',I2,'/',I2)
         mrecgts  = ibuf(112)
   print*, 'number of GTS retrievals ',mrecgts
c
c   set up character font
c
         call chrin (fcb, 0, font)
c
c   display pressure levels
c
         write (6,*)
         write (6,*) 'PRESSURE LEVELS (MB): 850, 700, 500, 400,'
         write (6,*) '  (TYPE 0 TO EXIT)     300, 250, 200, 150,'
         write (6,*) '                        100'
         write (6,*)
c
c   promt user for pressure level
c
      5 continue
         call movch (16hPRESSURE_LEVEL  , buffer, 0, 0, 16)
         call parmi (fcb, parms, pres, buffer, 0, 1, 1, 1)
         pres = 500
c
         call define (fcb, parms, 1, 0)
c
c   check validity of selected pressure level (0-exit)
c
         if (pres .eq. 0) go to 999
         if (pres .ne. 850 .and. pres .ne. 700 .and.
        *    pres .ne.  500 .and. pres .ne. 400 .and. pres .ne.
        *    300 .and.pres .ne.  250 .and. pres .ne. 200 .and.
        *    pres .ne. 150 .and. pres .ne.  100) go to 5
c
c   compute index (buffer offset) for selected pressure level
c
```

```
          do 10  index = 1, 9
             if (pres .eq. ipr(index)) go to 20
       10 continue
       20 continue
c
c   read button (main menu mode):
c
          call rbutn (fcb, ibut, ix, iy)
c   button = 3(F3): select new pressure level or exit
c
          if (ibut .eq. 3) go to 5
c
c   button = 2(F2): toggle map graphics planes
c   button = 7(D1): toggle TOVS overlay
c
          if (ibut .eq. 2) call mapths (fcb, buffer)
          if (ibut .eq. 7) call tovths (fcb, buffer)
          if (ibut .eq. 5) call MGNths (fcb, buffer)
          if (ibut .eq. 6) call DRCths (fcb, buffer)
c
c   for all other buttons create TOVS overlay
c
          if (ibut .eq. 12 .or. ibut .eq.13 )
         *    call thesplt (fcb, buffer, ibuf, ibut)
          go to 20
c
c   clean up
c
      999 continue
          call metio (ibuf, 0, -1)
          call thsio (ibuf, 0, -1)
          CALL DEXEC (FCB)
          RETURN
          END

c|----------------------------------------------------------------|
          subroutine valget (ibuf, index, index1, index2, ibut,
     1                 ix, iy, ix1, ix2, iy1, iy2, angle, imgnd)
c|----------------------------------------------------------------|
c|                                                                |
c|                                                                |
c|   Calling sequence:                                            |
c|       call valget (ibuf, index, index1, index2, ibut,          |
c|                    ix, iy, ix1, ix2, iy1, iy2, angle, imgnd)   |
c|                                                                |
c|----------------------------------------------------------------|
c
          integer   index, index1, index2, ibut, val(3), values(1)
          integer*4 ibuf(1), iangle, imgnd
c
c   select parameter based on button pushed and  pressure
c   level (index)
c
          if (ibut .eq. 12)  then
c          print*, ' TOVS plot '
           mindex = (index-1)*3 + 1
           val(1) = ibuf(mindex+6)
```

```
          if(ibuf(mindex+6).eq.999999) return
          iangle = ibuf(mindex+8)/1000
          imgnd = ibuf(mindex+8) - (iangle*1000)
         else
          if(ibuf(index+82).eq.99999) return
          iangle = ibuf(index+80)/1000
          imgnd = ibuf(index+80) - (iangle*1000)
         endif
          angle = 450 - iangle
          IF(ANGLE.GT.360) ANGLE = ANGLE -360
          ismple = 10*cosd(angle)
          iline = 10*sind(angle)
          ix1 = ix
          ix2 = ix1 + ismple
          iy1 = iy
c
c    iline should be minus as          |
c                                      |     s   , l
c                                 -  - |     +     -
c                               ------|-----------> x
c                                      |
c                                 -  + |     +     +
c                                      v  y
          iy2 = iy1 - iline
          ival = val(1)
         return
         end
```

```
c|--------------------------------------------------------------|
         subroutine thhead (buffer, ibut)
c|--------------------------------------------------------------|
c|                                                              |
c|  Description:                                                |
c|     Writes the TOVS data description header to the           |
c|     graphics plane.                                          |
c|                                                              |
c|                                                              |
c|  Calling sequence:                                           |
c|      call thhead (buffer, ibut)                              |
c|                                                              |
c|--------------------------------------------------------------|
c
         integer   buffer(1), ibut
         integer   font(2048), char(10), ipr(9), mrec, index
c
         common / disp / font, char, ipr, mrec, mrecgts, index
c
c   Depending on the number of digits needed:
c          1 - convert binary pressure to ascii
c          2 - set number of offset bytes
c
         if (ipr(index) .gt. 999) then
            call bnasc (ipr(index), buffer, 0, 4, 10)
            ioff = 4
            go to 10
```

```
         endif
         if (ipr(index) .gt. 99) then
            call bnasc (ipr(index), buffer, 0, 3, 10)
            ioff = 3
            go to 10
         endif
            call bnasc (ipr(index), buffer, 0, 2, 10)
            ioff = 2
   10    continue
c
c   Branch to appropriate heading depending on user
c   selected button value.
c
         go to (999, 999, 999, 999, 999, 999, 999, 999, 999, 999,
     .        999, 120, 130, 999, 999) ibut
  120    call movch (30hMB TOVS WIND SPEED (M/SEC)           ,
     .                buffer,0,ioff,30)
         go to 999
  130    call movch (30hMB GTS WIND SPEED  (M/SEC)           ,
                     buffer,0,ioff,30)
         go to 999
c
  999    continue
         return
         end
```

```
c|----------------------------------------------------------|
         subroutine mapths (fcb, buffer)
c|----------------------------------------------------------|
c|                                                          |
c|   Description:                                           |
c| Toggles the WDB map and lat/lng grid overlay on and off  |
c|                                                          |
c|----------------------------------------------------------|
c|                                                          |
c|   Calling sequence:                                      |
c|       call mapths (fcb, buffer)                          |
c|                                                          |
c|----------------------------------------------------------|
c|                                                          |
c
         integer   fcb(1), buffer(1)
         common / tog  / limg, lmap, lmgn, ldrc, ltov, lbut1,
     .                  lbut2, lbut3
c   if map graphics are on then turn off, if off then turn on
c
         if (lmap .eq. 1) then
            call ofgrf (fcb, buffer, 0)
            lmap = 0
         else
            call clrpln (fcb, buffer, 0, 0.00, 1.0, 1.00, 1, 3)
            lmap = 1
         endif
  999    continue
         return
         end
```

```
c|-----------------------------------------------------------|
      subroutine MGNths (fcb, buffer)
c|-----------------------------------------------------------|
c|                                                           |
c|   Description:                                            |
c|      Toggles the magnitude plane on and off.              |
c|                                                           |
c|-----------------------------------------------------------|
c|                                                           |
c|   Calling sequence:                                       |
c|       call mgnths (fcb, buffer)                           |
c|                                                           |
c|-----------------------------------------------------------|
c|
      integer   fcb(1), buffer(1)
      common / tog  / limg, lmap, LMGN, LDRC, ltov, lbut1,
     .                lbut2, lbut3
c
c   if graphics is on then turn off, if off then turn on
c
      if (lmGN .eq. 1) then
        IF(LTOV.EQ.1) THEN
            call ofgrf (fcb, buffer, 5)
        ELSEIF(LTOV.EQ.2) THEN
       CALL OFGRF (FCB, BUFFER, 1)
       ELSE
       CALL OFGRF (FCB, BUFFER, 3)
       ENDIF
         lmGN = 0
      else
        IF(LTOV.EQ.1) THEN
          call clrpln (fcb, buffer, 5, 1.00, 0.00, 0.0, 1, 3)
        ELSEIF(LTOV.EQ.2) THEN
          call clrpln (fcb, buffer, 1, 1.00, 0.00, 0.0, 1, 3)
       ELSE
          call clrpln (fcb, buffer, 3, 1.00, 0.00, 0.0, 1, 3)
       ENDIF
         lmGN = 1
      endif
  999 continue
      return
      end
c|-------------------------------------------------------------|
      subroutine DRCths (fcb, buffer)
c|-------------------------------------------------------------|
c|                                                             |
c|   Description:                                              |
c|      Toggles the directional arrow plane on and off.        |
c|                                                             |
c|-------------------------------------------------------------|
c|                                                             |
c|   Calling sequence:                                         |
c|       call drcths (fcb, buffer)                             |
c|                                                             |
c|-------------------------------------------------------------|
c
      integer   fcb(1), buffer(1)
```

```
      common / tog  / limg, lmap, lmgn, ldrc, ltov, lbut1,
     .                lbut2, lbut3
c   if graphics is on then turn off, if off then turn on
c
      if (lDRC .eq. 1) then
        IF(LTOV.EQ.1) THEN
            call ofgrf (fcb, buffer, 6)
        ELSEIF(LTOV.EQ.2) THEN
       CALL OFGRF (FCB, BUFFER, 2)
       ELSE
       ENDIF
     .   lDRC = 0
      else
        IF(LTOV.EQ.1) THEN
          call clrpln (fcb, buffer, 6, 0.75, 0.75, 0.00, 1, 3)
        ELSEIF(LTOV.EQ.2) THEN
          call clrpln (fcb, buffer, 2, 0.75, 0.75, 0.00, 1, 3)
       ELSE
          call clrpln (fcb, buffer, 4, 0.75, 0.75, 0.00, 1, 3)
       ENDIF
        lDRC = 1
      endif
  999 continue
      return
      end
c|-------------------------------------------------------------|
      subroutine tovths (fcb, buffer)
c|-------------------------------------------------------------|
c|                                                             |
c|   Description:                                              |
c|      Toggles the TOVS planes from 1 to 2 to 3 to 1 .....    |
c|                                                             |
c|-------------------------------------------------------------|
c|                                                             |
c|   Calling sequence:                                         |
c|       call tovths (fcb, buffer)                             |
c|                                                             |
c|-------------------------------------------------------------|
c
      integer   fcb(1), buffer(1)
      common / tog  / limg, lmap, lmgn, ldrc, ltov, lbut1,
     .                lbut2, lbut3
c   if TOVS plane #1 and #2 are on then turn on planes #3 & #4
c   else turn on planes #5 and #6
c   else turn on planes #1 and #2
c
      go to (456,564,645),ltov
 456     call ofgrf (fcb, buffer, 3)
         call ofgrf (fcb, buffer, 4)
         call ofgrf (fcb, buffer, 5)
         call ofgrf (fcb, buffer, 6)
         call clrpln (fcb, buffer, 2, 0.75, 0.75, 0.00, 1, 3)
         call clrpln (fcb, buffer, 1, 1.00, 0.00, 0.00, 1, 3)
         ltov = 2
         go to 9456
 564     call ofgrf (fcb, buffer, 5)
         call ofgrf (fcb, buffer, 6)
```

```
                call ofgrf (fcb, buffer, 1)
                call ofgrf (fcb, buffer, 2)
                call clrpln (fcb, buffer, 4, 0.75, 0.75, 0.00, 1, 3)
                call clrpln (fcb, buffer, 3, 1.00, 0.00, 0.00, 1, 3)
                ltov = 3
                go to 9456
 645            call ofgrf (fcb, buffer, 1)
                call ofgrf (fcb, buffer, 2)
                call ofgrf (fcb, buffer, 3)
                call ofgrf (fcb, buffer, 4)
                call clrpln (fcb, buffer, 6, 0.75, 0.75, 0.00, 1, 3)
                call clrpln (fcb, buffer, 5, 1.00, 0.00, 0.00, 1, 3)
                ltov = 1
 9456           continue
              if(lmgn.eq.0) then
              call mgnths (fcb, buffer)
              lmgn = 1
              endif
              if(ldrc.eq.0) then
              call drcths (fcb, buffer)
              ldrc = 1
              endif
             return
             end

c|------------------------------------------------------------|
      SUBROUTINE thsio(IOUT,NSEC,IOPT,*)
c|------------------------------------------------------------|
c
C THE FIRST RECORD CONTAINS HEADER INFORMATION.
C THE NUMBER OF FOLLOWING RCORDS IS CONTAINED IN THE LAST
C (112) ELEMENT OF THE HEADER RECORD.  THE FILE IS DIRECT
C ACCESS.
C
      LOGICAL HERE
      BYTE NAME(12),LETTER(26)
      COMMON/POINT/NFILE
      integer*4 IOUT(112)
      DATA IU/24/,LENR/112/,IOPEN/1/
      DATA NAME/'T','O','V','R','E','T',' ','.','D','A','T',"0/
      DATA LETTER/'A','B','C','D','E','F','G','H','I','J',
     * 'K','L','M', 'N','O','P','Q','R','S','T','U',
     * 'V','W','X','Y','Z'/
      IF(IOPEN.EQ.0) GO TO 110
      IF(IOPT.GT.0) GO TO 100
  100 OPEN(IU,FILE='WIND_DAT',ACCESS='DIRECT',RECL=LENR,
    1     STATUS='UNKNOWN')
      IOPEN=0
  110 CONTINUE
      IF(IOPT.LT.0) GO TO 130
      IREC=NSEC+1
      IF(IOPT.GT.0) GO TO 120
      READ(IU,REC=IREC) (IOUT(I),I=1,LENR)
      GO TO 140
  120 WRITE(IU,REC=IREC) (IOUT(I),I=1,LENR)
      GO TO 140
  130 CLOSE(IU)
```

```
        IOPEN=1
  140 RETURN
      END
```

```
c|-------------------------------------------------------------|
      SUBROUTINE metio(IOUT,NSEC,IOPT,*)
c|-------------------------------------------------------------|
C
C THE FIRST RECORD CONTAINS HEADER INFORMATION.
C THE NUMBER OF FOLLOWING RECORDS IS CONTAINED IN THE LAST
C (112) ELEMENT OF THE HEADER RECORD.  THE FILE IS DIRECT
C ACCESS.
c
      LOGICAL HERE
      BYTE NAME(12),LETTER(26)
      COMMON/POINT/NFILE
      integer*4 IOUT(112)
      DATA IU/25/,LENR/112/,IOPEN/1/
      IF(IOPEN.EQ.0) GO TO 110
      IF(IOPT.GT.0) GO TO 100
  100 OPEN(IU,FILE='met.dat',ACCESS='DIRECT',RECL=LENR,
     1  STATUS='UNKNOWN')
      IOPEN=0
  110 CONTINUE
      IF(IOPT.LT.0) GO TO 130
      IREC=NSEC+1
      IF(IOPT.GT.0) GO TO 120
      READ(IU,REC=IREC) (IOUT(I),I=1,LENR)
      GO TO 140
  120 WRITE(IU,REC=IREC) (IOUT(I),I=1,LENR)
      GO TO 140
  130 CLOSE(IU)
     IOPEN=1
  140 RETURN
      END
```

```
c|-------------------------------------------------------------|
      SUBROUTINE NYYDDD(NDATE,nd,nm,ny)
c|-------------------------------------------------------------|
      INTEGER*4 NDATE
      IF(NDATE.LT.100000) RETURN
      NY=NDATE/10000
      NMND=MOD(NDATE,10000)
      NM=NMND/100
      ND=MOD(NMND,100)
      RETURN
      END
```

```
c|-------------------------------------------------------------|
      subroutine thesplt (fcb, buffer, ibuf, ibut)
c|-------------------------------------------------------------|
c|                                                             |
c|   Calling sequence:                                         |
c|       call thesplt (fcb, buffer, ibuf, ibut)                |
c|                                                             |
```

```
c|--------------------------------------------------------------------|
        integer   fcb(1), buffer(1), ibut,val(3)
        integer   limg, lmap, ltov, lbut1, lbut2, lbut3
        integer   font(2048), char(10), ipr(9), mrec, index
        integer   ipln, irec, values(20)
c
        integer*4 ibuf(1)
        real      lats, latn, lnge, lngw, lat, lng
c
        common / bound/ lats, latn, lnge, lngw
        common / tog  / limg, lmap, lmgn, ldrc, ltov, lbut1,
                        lbut2, lbut3
        common / disp / font, char, ipr, mrec, mrecgts, index
        common / MDATE / NDATE, NMONTH, NYEAR, ITIME, IDSAT
c
c
        data   ipln / 2 /
         data values/20*-1/
c
c   set up graphics:  if first time then set up plane #1 & #2
c          else #3 & #4, then #5 & #6
c
         if (ipln .eq. 2) then
            ltov = 1
            call ofgrf (fcb, buffer, 3)
            call ofgrf (fcb, buffer, 4)
            call ofgrf (fcb, buffer, 5)
            call ofgrf (fcb, buffer, 6)
            call bchan (fcb, buffer, -32768, ipln)
            call bchan (fcb, buffer, -32768, ipln*2)
            call clrpln (fcb, buffer, 1, 1.00, 0.00, 0.0, 1, 3)
            call clrpln (fcb, buffer, 2, 0.75, 0.75, 0.0, 1, 3)
         elseif (ipln.eq.8) then
            ltov = 2
            call ofgrf (fcb, buffer, 5)
            call ofgrf (fcb, buffer, 6)
            call ofgrf (fcb, buffer, 1)
            call ofgrf (fcb, buffer, 2)
            call bchan (fcb, buffer, -32768, ipln)
            call bchan (fcb, buffer, -32768, ipln*2)
            call clrpln (fcb, buffer, 3, 1.00, 0.00, 0.0, 1, 3)
            call clrpln (fcb, buffer, 4, 0.75, 0.75, 0.0, 1, 3)
         else
            ltov = 3
            call ofgrf (fcb, buffer, 1)
            call ofgrf (fcb, buffer, 2)
            call ofgrf (fcb, buffer, 3)
            call ofgrf (fcb, buffer, 4)
            call bchan (fcb, buffer, -32768, ipln)
            call bchan (fcb, buffer, -32768, ipln*2)
            call clrpln (fcb, buffer, 5, 1.00, 0.00, 0.0, 1, 3)
            call clrpln (fcb, buffer, 6, 0.75, 0.75, 0.0, 1, 3)
         endif
c
c   loop thru retrievals:
c      - read retrieval
c      - calculate latitude and longitude
```

```
c       - check to see if within boundaries
c       - interpolate to find (x,y)
c
         if(ibut.eq.12) lrec = mrec
         if(ibut.eq.13) lrec = mrecgts
        do 40  irec = 1, lrec
         if(ibut.eq.12) then
           call thsio (ibuf, irec, 0)
           lat     = floatj(ibuf(5))  / 100.0
           lng     = floatj(ibuf(6))  / 100.0
         else
           call metio (ibuf, irec, 0)
           lat     = floatj(ibuf(1))  / 100.0
           lng     = floatj(ibuf(2))  / 100.0
        endif
           if (lat .ge. lats  .and.  lat .le. latn  .and.
     *         lng .ge. lngw  .and.  lng .le. lnge) then
         call bilin (latn, lats, lnge, lngw, lat, lng, ix, iy)
            if (ibuf(102) .ne. 777) then
              call valget (ibuf, index, index1, index2, ibut,
     1                ix, iy, ix1, ix2, iy1, iy2, angle, imgnd)
        if(imgnd.eq.99) go to 40
              if (ltov .eq. 1) lbut1 = ibut
              if (ltov .eq. 2) lbut2 = ibut
              if (ltov .eq. 3) lbut3 = ibut
         call dvect(fcb, buffer, values, ix1,iy1,ix2,iy2,
     *                          200, -32768, ipln)
c
c  this portion is for arrow head
c
         tangle = angle - 45
         if(tangle.gt.360) tangle = tangle - 360
              iss = 5*cosd(tangle)
              isl = 5*sind(tangle)
                   ixx1 = ix1 + iss
                   iyy1 = iy1 - isl
         call dvect(fcb, buffer, values, ix1,iy1,ixx1,iyy1,
     *                          200, -32768, ipln)
         tangle = angle + 45
         if(tangle.gt.360) tangle = tangle - 360
              iss = 5*cosd(tangle)
              isl = 5*sind(tangle)
                   ixx1 = ix1 + iss
                   iyy1 = iy1 - isl
         call dvect(fcb, buffer, values, ix1,iy1,ixx1,iyy1,
     *                          200, -32768, ipln)
c
         if(angle.gt.180.0) then
         ixx1 = ix2 - 3
         ixx2 = iy2 + 6
         ixx1 = ix2 - 2
         ixx2 = iy2 - 6
         endif
         IF(IMGND.GE.100) THEN
                call bnasc (IMGND, char, 0, 3, 10)
                call dchar (fcb, buffer, font, 9, 1., CHAR, 3,
     *   ixx1, ixx2, 0., 0., 1, 1, 1, 1, -32768, ipln*2,
```

```
     *                    255,0, 0)
        ELSEIF(IMGND.LT.10) THEN
               call bnasc (IMGND, char, 0, 1, 10)
               call dchar (fcb, buffer, font, 9, 1., CHAR, 1,
     *   ixx1, ixx2, 0., 0., 1, 1, 1, 1, -32768, ipln*2,
     *                    255,0, 0)
        ELSE
               call bnasc (IMGND, char, 0, 2, 10)
               call dchar (fcb, buffer, font, 9, 1., CHAR, 2,
     *  ixx1, ixx2, 0., 0., 1, 1, 1, 1, -32768, ipln*2,
     *                    255,0, 0)
        ENDIF
           endif
          endif
   40 continue
c
c   write heading to graphics
c
       call thhead (buffer, ibut)
       call dchar (fcb, buffer(100), font, 12, 1., buffer, 30,
     *  256, 480, 0., 0., 1, 1, 1, 1, -32768, ipln, 255, 0, 0)
       call dchar (fcb, buffer(100), font, 12, 1., buffer, 30,
     *  256, 480, 0., 0., 1, 1, 1, 1, -32768, ipln*2,255,0, 0)
        if(ibut.eq.12) then
        IF(IDSAT.EQ.2) THEN
             CALL MOVCH (10HNOAA-10   , BUFFER,0,0,10)
        ELSE
             CALL MOVCH (10HNOAA-9    , BUFFER,0,0,10)
        ENDIF
        elseif(ibut.eq.13) then
             CALL MOVCH (10H          , BUFFER,0,0,10)
        ENDIF
        CALL MOVCH(1H/,BUFFER,0,12,1)
        CALL MOVCH(1H/,BUFFER,0,15,1)
        if (ndate.gt.50) then
        nndate = ndate - 50
        else
        nndate = ndate
        endif
        call bnasc(nndate, buffer, 10, 2, 10)
        call bnasc(nmonth, buffer, 13, 2, 10)
        call bnasc(nyear, buffer, 16, 2, 10)
       call dchar (fcb, buffer(100), font, 10, 1., buffer, 18,
     * 256, 500, 0., 0., 1, 1, 1, 1, -32768, ipln, 255, 0, 0)
       call dchar (fcb, buffer(100), font, 10, 1., buffer, 18,
     * 256, 500, 0., 0., 1, 1, 1, 1, -32768, ipln*2,255,0, 0)
C
c   setup graphics for next call to thesplt
c
       if (ipln .eq. 2) then
         ipln = 8
       elseif (ipln .eq. 8) then
         ipln = 32
       else
         ipln = 2
       endif
       return
       end
```

References

[1] "User's Guide, VAX Interactive data language (IDL) ", Research Systems Inc., 2021 Albion street, Denver, CO 80207, USA, 1987, pp. 1.1-1.4.

[2] "System 575 Intrinsics and Primitives, Version 3.0, Reference Manual", International Imaging Systems Inc., 1500 Buckeye Drive, Milpitas, CA 95035, USA, 1983, pp. 5.1-5.9, 5.39-5.44, 1.1-1.2.

[3] "Model 575 Image Processor, Product description", International Imaging Systems Inc., 1500 Buckeye Drive, Milpitas, CA 95035, USA, 1983, pp. 1.1-1.3, 2.1-2.22.

[4] "TEMP code 1978 for reporting Radiosonde and Rawinsonde observations", Bangladesh Meteorological Department, 1979.

[5] "Weather Reporting, volume A, Observing stations", WMO/OMM-No. 9, Secretariat of the World Meteorological organization, Geneva, Switzerland, June 1986.

[6] R.E. Bruce, L.D. Duncan and J.H. Pierluissi, "1977 : Experimental study of the relationship between Radiosonde temperatures and Satellite-derived temperatures", Monthly Weather Rev., 1977, pp. 493-496.

[7] G. Shaffer, "Standard operating procedure of Subroutine GRID", Science Systems and Application Inc., Aerospace building, 10210 Greenbelt road, Greenbelt , MD 20706, USA, 1984.

[8] "Tiros satellite HRPT data acquisition computer program", Bangladesh Agro-Climatic and Environmental Monitoring Project (ACEMP), Science Systems and Application Inc., Aerospace building, 10210 Greenbelt road, Greenbelt , MD 20706, USA, 1984, p-2.

[9] W.L. Smith, H.M. Woolf, C.M. Hayden, A.J. Scheriner, and J.F. Le Marshall, "The physical retrieval TOVS export package", Technical proceeding, First international TOVS study conference, Igls, Austria, 1983, pp. 111-131.

[10] "LPA11-K laboratory Peripheral Accelerator user's guide", Digital Equipment Corporation, Educational services, Digital Equipment Corporation, South Lawrence, Ma., USA.

[11] "Instruction manual, EMR Model 814-02 TIROS-N synchronizer (Volume I and II)", Fairchild Weston, P.O. Box 3041, Sarasota, Florida 33578, USA.

[12] "Instruction manual, Model 760 Buffered Data Channel", Fairchild Weston, P.O. Box 3041, Sarasota, Florida 33578, USA.

[13] "Atmospheric Sounding User's guide", NOAA, Technical report, National Environmental Satellite Studies (NESS 83), National technical information service, Department of Commerce, Sills building, 5285 Port Royal Road, Springfield, VA 22161, USA, April, 1981, pp. 11-64.

[14] James C. Nelson, "Documentation of the TOVS Bangladesh data processing software studies", CIMSS, Space Science and Engineering centre, University of Wisconsin, Madison, Wisconsin 53706, USA, Feb. 1985, pp. 1-12.

[15] Herbert J. Spiegel and Arnold Gruber, "From the weather vanes to satellite", John Wiley, New York, 1984, pp-176.

[16] "System 575 software, digital image processing version 3.1, User's manual", International Imaging Systems Inc., 1500 Buckeye Drive, Milpitas, CA 95035, USA, 1983, pp. 1.1-1.17, 2.2.

[17] "System 575 programmer's reference manual, version 3.1, User's manual", International Imaging Systems Inc., 1500 Buckeye Drive, Milpitas, CA 95035, USA, 1983, pp. 6.1-6.3.

[18] Gene Shaffer, "Subroutine BILIN", S575 software package, SPARRSO, Dhaka.

[19] V. Carl Hamacher, Zvonco G. Vranesic, Safwat G. Zaky, " Standard communication interfaces; Computer organiztion", McGraw-Hill International Book Company,. 1984, pp. 460-463.

[20] G. Shaffer, "SPARRSO SOP for program HRPT", Science Systems Applications Inc. (SSAI, Aerospace building, 10210 Greenbelt road, Greenbelt , MD 20706, USA, 1984.

[21] "DMF32 user's guide", Educational Services, Digital Equipment Corporation, South Lawrence, Ma., USA.

[22] G. Shaffer, "SPARRSO SOP for program MAP", Science Systems Applications Inc. (SSAI, Aerospace building, 10210 Greenbelt Road, Greenbelt , MD 20706, USA, 1984.

[23] Levin Lauritson, Gary J. Nelson and Frank W. Porto, "Data extraction and calibration of TIROS-N/NOAA radiometers." NOAA, Technical Memorandum, National Environmental Satellite Studies (NESS 107), National Technical Information Service, Department of Commerce, Sills building, 5285 Port Royal Road, Springfield, VA 22161, USA, November, 1979, pp. 38-40, 10-36.

[24] Arthur Scwab, "The TIROS-N/NOAA A-G satellite series", NOAA, Technical memorandum, National Environmental Satellite Studies (NESS 95), National Technical Information Service, Department of Commerce, Sills building, 5285 Port Royal Road, Springfield, VA 22161, USA, March, 1978.

[25] Arthur Scwab, "Modified version of the TIROS-N/NOAA A-G satellite series (NOAA E-J) advanced TIROS N (ATN)", NOAA, Technical Memorandum, National Environmental Satellite Studies (NESS 116), National Technical Information Service, Department of Commerce, Sills building, 5285 Port Royal Road, Springfield, VA 22161, USA, February, 1982, pp. 1-7.

[26] "GMS user's guide, Meteorological Satellite Centre, 3-235 Nakakiyoto, Kiyose, Tokyo, 180-04 Japan, March, 1980.

[27] "Technical manual K560-MS6 satellite image recorder", Muirhead subsystems Ltd., USA.

[28] H. M. De Tong, "Description of the quality control procedures utilized by the special aircraft data centre", Koninklinjk Nederlands Meteorological Instituut, De Bilt, the Nederlands, p-215.

[29] F. Delsol, "Monitoring the availability and quality of GTS observation at ECMWF", European centre for medium range weather forecasts (ECMWF), workshop on the use and quality control of meteorological observations, 1984, p-76.

[30] "SPOT Imagery simulation, prospective user", Centre National d'Etudes Spatials (CNES), Centre Spatial de Toulouse, 18 Avenue Edouardo-Belin, 31055 Toulouse Cedex, France, p-F2.

[31] "SPOT Satellite based remote sensing system", Centre National d'Etudes Spatials (CNES), Centre Spatial de Toulouse, 18 Avenue Edouardo-Belin, 31055 Toulouse Cedex, France, pp. 5-14.

[32] "Operation and maintenances manual for Metrack-8 antenna system", Datron systems Inc., 20700 Plummer street, Chatsworth, California 91371, pp. 1.1-1.16.

[33] "R. F. Telemetry and meteorological satellite receivers", 1100-AR receiver, Microdyne corporation, PO Box 1527, Rockville, MD 20850, USA, pp. 2-5.

[34] "World data base maps", S575 software package, SPARRSO, Dhaka.

[35] Dr. William A. Heckley, "Data analysis at ECMWF, a lecture note", Workshop on numerical weather prediction, SPARRSO, Dhaka, Bangladesh, October - November, 1986, pp. 35-37.

[36] Nichabar M. Short, Paul D. Lowman and Stanley C. Freden of Goddard Space Flight Centre (GSFC), NASA and William A. Finch of San Diego State University, "The Landsat system in Appendix A in 'Mission to earth': Landsat views the world.", US government printing office, Washington, DC 20402, USA, 1976, pp. 437-441.

List of abbreviations

GTS	Global Telecommunication Systems.
TOVS	Tiros Operational Vertical Sounder.
DCP	Data Collection Platform.
DCS	Data Collection System.
RAM	Random Access Memory.
RGB	Red, Green and blue color.
IIS	International Imaging system.
SOP	Standard Operating procedure.
ACEMS	Agro-Climatic and Environmental Monitoring System, SPARRSO, Dhaka, Bangladesh.
SPARRSO	Space Research and Remote Sensing Organisation, Dhaka, Bangladesh.
BMD	Bangladesh Meteorological Department, Dhaka, Bangladesh.
NOAA	National Oceanic and Atmospheric Administration, Maryland, U.S.A.
WMO	World Meteorological Organization.
SPOT	Satellite Probatoire d'Observation de la Terre, Toulouse, France.
GMS	Geostationary Meteorological Satellite System.
GPH	Geo-Potential Height.
DPD	Dew Point Depression.
AVHRR	Advanced Very High Resolution Radiometer.
HRPT	High Resolution Picture Transmission.
APT	Automatic Picture Transmission.
MIRP	Manipulated Information Rate Processor.
TIP	TIROS Information Processor.
IDL	Interactive Data Language
ASCII	American Standard Code for Informaion Interchange.
GTS data	Meteorological data obtained through Global Tele-communication Systems.
CIMSS	Cooperative Institute for Meteorological Satellite Studies.
TACQ	Tiros Acquisition.
PIXEL	Picture Element.

ECMWF	European Centre for Medium Range Weather Forecasting.
CCT	Cumputer Compatible Tape.
NESS	National Environmental Satellite Service, Washinton, USA.