

OCEAN SURFACE CURRENT SIMULATIONS
IN THE NORTH PACIFIC OCEAN AND BERING SEA
(OSCURS -- NUMERICAL MODEL)

by

W. James Ingraham, Jr.

and

Robert K. Miyahara

Resource Ecology and Fisheries Management Division
Northwest and Alaska Fisheries Center
National Marine Fisheries Service
National Oceanic and Atmospheric Administration
7600 Sand Point Way N.E.
Seattle, WA 98115

March 1988

This document is available to the public through:

National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

THIS PAGE INTENTIONALLY LEFT BLANK

CONTENTS

	Page
ABSTRACT	iv
INTRODUCTION	1
SET UP, INPUT, AND RUN INSTRUCTIONS	3
OSCURS Model(40X104) Grid	3
Input Coastline Boundary Conditions	5
Input Permanent Geostrophic Current File	8
Input of Sea Level Pressure Data Fields	9
Input Starting Locations for Progressive Vector Tracks	9
Input Control Parameters and Run Instructions	11
DAILY SEQUENCE LOOP AND OUTPUT FILES	15
Sea Level Pressure	15
Wind	16
Current Due to Wind	19
Tuning and Vector Summation of Currents	20
Total Currents With Boundary Effects	22
Transport (Progressive) Vectors	23
GRAPHICS OUTPUT PROGRAMS	25
Plots on 40X104 Grid	26
Geostrophic Currents	30
Wind Currents	33
Total Currents	35
Progressive Vector Tracks	39
Zoom Plot of Gulf of Alaska on 25X35 Grid.....	41
Geostrophic Currents	48
Wind Currents	50
Total Currents	56
Progressive Vector Tracks	65
REFERENCES	73
APPENDICES	75
1. Listing of Fortran Program "M/FINAL" OSCURS Model on 40X104 Grid	75
2. Listing of Fortran Program "M/PLOT/AUTO" Graphics Outputs on 40X104 Grid	113

ABSTRACT

A surface ocean current model, OSCURS (Ocean Surface CURrent Simulations), has been developed as a tool in ecosystem-fisheries- oceanographic research. Wind and ocean surface drift are computed from historic daily sea level pressure fields (1946 to 1987) on a 40X104 grid over the North Pacific Ocean and Bering Sea. Total flow is the vector sum of long-term baroclinic geostrophic flow (0/2000 db) and surface wind drift. Model outputs include daily vector current fields or progressive vector drift tracks from any selected location over any selected time period. Model tuning and calibration results will appear in subsequent publications.

INTRODUCTION

Both theoretical and empirical knowledge of ocean currents have increased gradually over the past several decades. Using satellite navigation to determine latitude-longitude coordinates has significantly improved our ability to accurately define the position of ships and instruments in the ocean. Recently, for the first time, drifting buoys were deployed in the North Pacific Ocean for a long enough time (1-2 years) to illustrate and generally verify the geostrophic circulation patterns which had been derived from the long-term averaged internal density distributions (calculated from historic measurements of temperature and salinity versus pressure or depth). This has provided a basis for computer simulation models of mean surface currents on an ocean-wide scale. In an effort to gain more information about details of flow for fisheries-oceanography studies, fisheries management research, and ecosystem model inputs, the following first order approach was adopted. A numerical simulation model "OSCURS" (Ocean Surface CURrent Simulations) was developed and has shown good promise in its ability to assess historic, daily ocean circulation in the North Pacific Ocean and Bering Sea between 1946 and 1987. This post-World War II time period was chosen because consistently better gridded sea level pressure data were available.

Although ocean currents are affected by many factors, only the two major factors which determine the driving forces--the permanent, internal, thermohaline density field and the

calculated local wind vectors--are used to generate surface currents in this model. Other factors such as tides, bathymetry, and atmospheric pressure effects, on sea level will be included in future model expansions as finer spatial resolution of details about flow over the shallow continental shelves is needed. Surface ocean currents considered here are composed from the vector sum of the local geostrophic flow and the local wind-induced flow. The model outputs are 1) ocean-wide, daily, vector fields of wind, wind current, geostrophic current, or total current or 2) progressive transport vectors from any selected spatial starting point or group of points over any time period from 1946 to 1987.

This report describes the detailed structure and mathematical basis for the OSCURS model including the (40X104) grid, setup, input, run instructions, the nature of computations, the daily run loop, and the output files created. The final section of the report contains the setup and run instructions for plotting model results on an ocean-wide scale or in zoom plot portions with graphics outputs on a Tektronics CRT or Calcomp Plotter. The Appendix contains listings of both Fortran programs, the model and the output graphics.

SET UP, INPUT, AND RUN INSTRUCTIONS

OSCURS Model (40X104) Grid

The (40X104) square computational grid of the model extends laterally across the North Pacific Ocean from the west coast of the U.S. (124°W) to southern Japan (130°E) and extends longitudinally southward from Bering Strait (67°N) to about latitude 30°N (Fig. 1). Note that the corner points of this grid are not necessarily the extremes of both latitude and longitude because of the nature of fitting a square, equal area grid to a portion of the nearly spherical Earth. Being a 1/4 mesh subset of a portion of the standard U.S. Navy Fleet Numerical Oceanography Center (FNOC) (63X63) Northern Hemisphere grid, our (40X104) grid has the properties of the FNOC grid but only on a finer scale. The base FNOC grid was derived by centering an equally spaced, square 380 km mesh at the North Pole of a Northern Hemisphere polar stereographic map projection true at 60°N with the columns aligned parallel to longitude 170°W. Our smaller, average mesh size of about 90 km was chosen to provide a better spatial resolution while still preserving cross-ocean continuity for computing long-term progressive transport vector tracks. Again, due to the spherical geometry, the mesh size varies slightly with latitude (from 95 km at 60°N to 83 km at 40°N). The formula used to compute the model grid length as a function of latitude in the pressure gradient calculation is

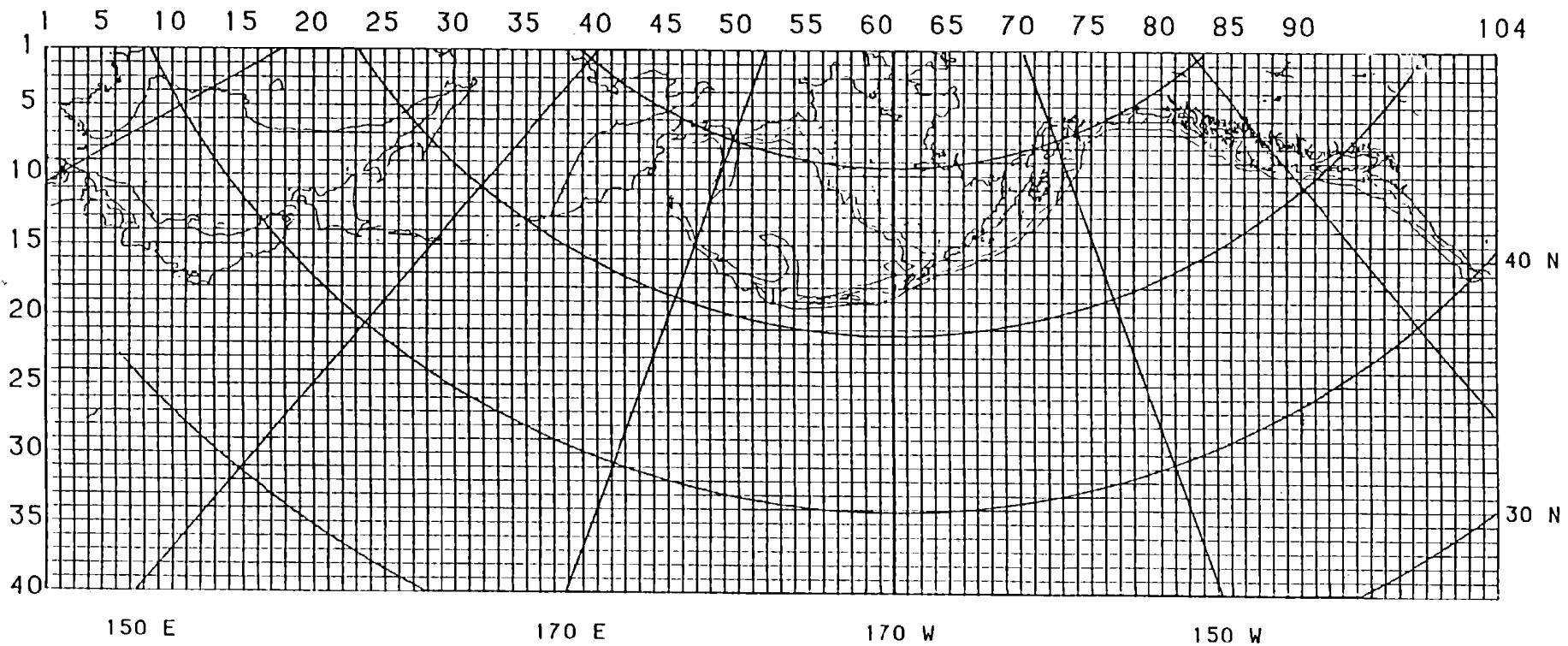


Figure 1.--Surface ocean current model (40x104) grid (95-83 km).

$$\text{Grid Length(km)} = 95.23 \left[\frac{1.0 + \sin(\text{latitude})}{1.0 + \sin(60)} \right]$$

A land-sea table (Fig. 2) is used to control the computations at different grid points related to their location over land or sea in the model. Values are assigned as follows: land=0, shallow ocean (<200m)=1, deep ocean (>200m)=2. Therefore, the land-sea table is used for tests to avoid zero points on the grid during both the computational and graphics output mode. Some land-sea values are set negative at points which are adjacent to the coastline for boundary condition calculations as described in the next section.

Input Coastline Boundary Conditions

To simulate boundary friction (zero slip at shoreline) and deflection of currents which impinge upon the coast at an angle all grid points within two grid lengths of the coastline were selected using the grid map (Fig. 1). Their grid coordinates (I,J) and the length (x-coordinate and y-coordinate) of the line normal to shore from each point were tabulated in grid space units and stored on disk Files 25 and 26. As these disk files are read into arrays, the corresponding (I,J) grid locations in the Land-Sea table are set negative to uniquely identify these particular points where rotation and deflection will be calculated in the subroutine CSTROT. The location of these points are indicated by the asterisks in Figure 3. The nature of the boundary modifications and calculations are discussed below under Total Currents with Boundary Effects.

Input Permanent Geostrophic Current File

A major component of total ocean flow is the geostrophic surface current computed from internal density gradients down to an assumed level of no motion. This is an estimate of long-term permanent flow. Ideally the model will be revised in the future to have seasonal or even monthly fields of permanent geostrophic currents, but due to the lack of year round data in oceanic areas and the considerable amount of time required to prepare fields with shallow water extrapolations from deep reference levels, a constant, long-term current field was used.

To obtain geostrophic currents, anomaly of Dynamic Height fields (0/2000 db) were computed from 1 X 1 degree historical means of temperature and salinity versus depth from the Bauer-Robinson Numerical Atlas of Ocean Basins (Bauer and Robinson 1985). Horizontal numerical interpolations were performed using a I-point Bessel central difference formula fit to the 1 X 1 degree square data to obtain data at the locations (latitude-longitude) of the model grid points.

Subroutine DELTAD accesses these data through File 33 and 34 (Anomaly of Dynamic Height data,) to compute the U (east-west) and V (north-south) components of geostrophic current (cm/sec) by standard methods (LaFonde 1951) and store these data in permanent File 38. DELTAD need not be called in subsequent runs of the model as the program then reads File 50 which is the equivalent of File 38.

Input of Sea Level Pressure Data Fields

The key input variable field which drives the model is daily sea level pressure data which was obtained from files of the U.S. Navy, FNOC Monterey, California, over the portion of the standard FNOC 380 km (63X63) grid between latitude 50N and 68°N and longitude 100°W and 120°E. Data processing of these data tapes into disk files containing daily sequences of sea level pressure data by month, season, or year for the Burroughs computer was documented by Ingraham et al. 1983. Because the OSCURS model grid is a 1/4-mesh, higher resolution subset, interpolations are performed using a 4-point spline central difference numerical surface fitting routine each time a new input daily field of sea level pressure is read. These data are the basis for the wind and wind current computations below.

Input Starting Locations for Progressive Vector Tracks

Before the main computation loop of the model is entered, it is necessary to choose the number of progressive vector tracks that you want to save on disk File 60 and their starting grid coordinates (I,.J). Two methods are provided, but in this version of the model the second method which saves every fourth grid point is skipped. Every grid point could be saved but storage space is economized here to show only the general capacities of the model. The first method is to put a few selected (I,J) starting grid points into File 28 (Table 1) which the model reads and counts. The counter IEND (line 412) must contain the exact number of these starting points. Although integer grid points

Table 1.--Selected starting grid points (I,J) for progressive transport vector computations on 40x104 grid (File 28, Title="P/I").

Record Number	I	J
1	10	73
2	11	73
3	11	74
4	12	74
5	12	75
6	13	75
7	14	76
8	15	77
9	16	78
10	17	79
11	18	80
12	19	81
13	20	82
14	21	83
15	22	84
16	23	85
17	24	86
18	20	12
19	28	16
20	24	20
21	20	24
22	16	28
23	32	28
24	20	36
25	28	36
26	36	36
27	20	44
28	28	52
29	36	52
30	28	64
31	36	64
32	16	73
33	28	76
34	36	76
35	8	80
36	12	84
37	24	88
38	32	88
39	16	96
40	24	96

are initially set up as starting locations, they may be reset to any decimal value or fraction of the way between grid points (see line 450-454 in the OSCURS model, Appendix 1).

Input Control Parameters and Run Instructions

Starting the model requires compiling the Fortran code and saving the object code interactively through CANDE (the standard Burroughs interactive language) followed by a "START M/FINAL/WFL" command. The Work Flow Language (WFL) program (M/FINAL/WFL) listed below (Program 1) shows the line numbers of the input and output file names and control parameters which must be edited to give the desired run conditions before the START command is given.

The output file names (lines 12-34 in M/FINAL/WFL) should include as much descriptive information about the runs they represent as possible. Although these names are long, the extra typing is quite important to clearly distinguish (now and at a later time) the many possible combinations of run parameters, especially when many runs of different time periods are to be compared.

Input File 2 is FNOC daily sea level pressure data for the desired time period. Retrieve these data from tape one year at a time in separate files of daily data fields from 1 January to 31 December. Edit into a smaller file if model starting month is greater than 1 to save reading time and disk space.

Input File 28 is a sequence of (I,J) grid point data specifying the nearest model grid point to the desired starting

Program 1. Listing of Work Flow Language (WFL) which starts the model run
(File Title="M/FINAL/WFL").

```

1  ?BEGIN JOB INGRAHAM/M/FINAL/W10DD00/S2X2/40X104;
2  QUEUE=10;
3  RUN OBJECT/M/FINAL;
4  FILE FILE2=FNWC/DSLPL/78/M7TOM12;
5  FILE FILE28=P/FIG/40X104;
6  FILE FILE50=CURRENTS/40X104/DYNCURR/S2X2;
7  FILE FILE51=CURRENTS/40X104/DSLPL/78/M7TOM12/D17START/W10DD00/S2X2
8      ON SCRATCH;
9  FILE FILE52=CURRENTS/40X104/WIND/78/M7TOM12/D17START/W10DD00/S2X2
10     ON SCRATCH;
11 FILE FILE53=CURRENTS/40X104/SFCOCN/78/M7TOM12/D17START/W10DD00/S2X
12     ON SCRATCH;
13 FILE FILE54=CURRENTS/40X104/W10DD00/78/M7TOM12/D17START/CSTROTATD/
14     ON SCRATCH;
15 FILE FILE55=CURRENTS/40X104/W10DD00/78/M7TOM12/D17START/S2X2
16     ON SCRATCH;
17 FILE FILE60=CURRENTS/40X104/PROGRESVECT/78/M7TOM12/D17START/W10DDO
18     ON SCRATCH;
19 FILE FILE5(KIND=READER);
20 ?DATA
21     MODEL CONTROL CARDS (CHANGE LEADING CONSTANTS IN CANDE)
22     78=====YEAR OF DSLPL FILE
23     7=====STARTING MONTH FOR RUN
24     12=====STOP AT END OF THIS MONTH
25     200=====NUMBER OF TOTAL DAYS IN RUN
26     1.0=====WIND SPEED MULTIPLIER (WFAC)
27     0.0=====GEOSTROPHIC CURRENT MULTIPLIER (DDFAC)
28     0=====FLAG TO CREATE NEW GEOSTROPHIC CURRENT FILE (0=NONE)
29     1=====FLAG TO STOP PRINTOUT OF FIELDS IN LOOP
30     1=====FLAG TO SUPRESS DISK STORAGE OF SLP,WIND,OC CUR PRE ROT
31     1=====FLAG TO SUPRESS DISK STORAGE OF EACH TOTAL CURR FIELD
32 ?
33 COPY CURRENTS/40X104/PROGRESVECT/78/M7TOM12/D17START/W10DD00/S2X2/
34     FROM SCRATCH(PACK) TO TYVO(PACK)
35 ?END JOB

```

position for each progressive vector to be stored on disk for the final plot.

Input File 5 contains input parameters which control the run time, the value of wind and current tuning factors, the amount of printing, and whether disk files are saved. Prior to run, edit this file and set the leading constant on each line according to your desired run conditions and the following instructions.

Line 21, YEAR-1900. This value must correspond to the same year as the data in File 2. Each run is limited to data from one calendar year only (at this time).

Line 24, STARTING MONTH. Program will read through the daily SLP data file and then start the computation loop when the starting month is encountered.

Line 23, STOP MONTH. Run will stop at the end of this month. To stop within a month see next line.

Line 24, RUN DAYS. Total number of days in run loop is used in stopping on a specific day. Check to be sure this number is greater than or equal to the time from start month to end month if a stop within a month is not desired.

Line 25, WIND TUNING FACTOR. In numerical experimenting with output vectors, different weighting factors may be applied to wind speed which will generate a corresponding linear change in surface ocean current due to wind.

Line 26, DELTAD TUNING FACTOR. This is a direct linear multiplier of each component of the permanent geostrophic current.

Line 27, NEW DELTAD FILE FLAG. For the initial run or in case the original anomaly of dynamic height field is changed (e.g., new reference level), this flag is set to "1" which results in the overwriting of file 38 with new U and V fields of permanent geostrophic current. In any subsequent run this flag is set to "0" and the program reads the previously stored version of the same name, File 50 with the title, "CURRENTS/40X104/DYNCURR/F".

Line 28, NO PRINT. Set this flag to "0" to receive a printout of each daily SLP (Sea Level Pressure), wind speed, U and V ocean currents due to wind before rotation of near

coastal vectors, and U and V total rotated current fields. For runs longer than a few weeks set the flag to "1" to avoid reams of printing.

Line 29, NO STORE. Set this flag to "1" to suppress storage of disk File numbers 51, 52, 53, and 55 leaving only the final total current fields stored in file 54. Set this flag to "0" to store all files on scratch disk.

Line 30, NO STORE FINAL. Set this flag to "1" to suppress storage of File 54, the final output daily current fields. Although these files require a lot of space these are the main results which should be archived on tape to avoid future reruns.

In output Files 51, 52, 53, 54, 55, and 60 the editing of file titles should include changes in year, months, start day, wind factor (W10), and delta-d factor (DD10) before the run is started. The "10" in "W10" and "DD10" indicates the tuning factors were both equal to "1.0" or neutral. Examples of other possible factors are "05"=0.5, "18"=1.8, and "00"=0.0. File 60 which contains the U and V components of all the transport vector starting and ending points in grid units is automatically stored on both SCRATCH disk and TYVO disk.

DAILY SEQUENCE LOOP AND OUTPUT FILES

The main function of this simulation model is to compute a surface drift current vector (cm/sec) at each grid point in the grid on a daily basis forming a daily series of velocity fields from which progressive vectors of drift tracks can be estimated. The following five steps are necessary to generate one field: 1) read in sea level pressure, 2) compute wind, 3) compute wind current, 4) tune and sum velocity components, and 5) add boundary effects. The sixth step is to compute the surface transport vectors (distance traveled per day) at each of the chosen starting points (the end points of these computed vectors become the starting points for the next day's vectors). Finally, vector velocity field components and transport vector start and end points in grid unit values are stored on separate disk files for plotting by the plot program. This procedure is repeated for the specified number of days (NDAYS) in the loop. Details of these steps and theoretical considerations are described below for one typical pass through the loop.

Sea Level Pressure

Subroutine READER (IMO, K, IYD, ITYP, SLP, SLPl) reads one 0000Z daily FNOC sea level pressure field from FILE 2 and returns the following variables, where IMO is the month, K is the year, IDY is the day and ITYP is the flag which normally equals zero but equals 999 at end of File 2. Also returned to the main program are the one-dimensional array SLP and the two-dimensional array SLPl which contain the data for processing and printing in the

program. A test is made to be sure the month read is the month desired. If not, another day of data is read in time sequence until the correct month is found. Units are converted to millibars from packed units (the difference from 1000.0 in tenths of millibars).

Conversion of data array, SLP, from the FNOC (20X44) grid to the model (40X104) grid array, SLP1, is performed by a I-point Bessel or Spline numerical curve fitting routine in two dimensions (a data surface fit). Comparisons between both interpolation routines indicate either one performs quite well. The interpolated field, array T1, is stored on disk File 51 by setting the flag NOSTOR=0.

Wind

In subroutine WIND(T1,LS1,ALAT2,WXX,WYY) three arrays are passed, where T1 is the SLP field which was just interpolated to the model grid, LS1 is a land-sea table with all values = 1 (computations are performed at each grid point even over land), and ALAT2 has the latitudes of each grid point. Two arrays (WXX and WYY) are returned which contain the computed U and V components of the wind vectors, respectively. Computations are done in two steps.

First, a pure geostrophic wind is calculated from the balance between the atmospheric pressure gradient force and the Coriolis force to the right of the wind looking in the direction of the wind. Since friction is absent in this theoretical approximation, wind vectors are parallel to the horizontal pressure

isobars. Solving for wind the formula is

$$W = \frac{1}{\rho f} \left[\frac{dP}{dL} \right]$$

where W is the wind vector (m/sec), ρ is the density of air, 1.22 kg/m³, f is the Coriolis parameter, $1.458 \times 10^{-4} \times \sin(\text{latitude})$, $\frac{dP}{dL}$ is the atmospheric pressure change per unit grid length which is computed using a 5-gridpoint central difference numerical curve fitting routine.

In the more natural case friction acts to slow the wind in the boundary layer near the ocean surface and causes the wind to be deflected to the left of the geostrophic wind with cross-isobar flow from higher toward lower pressure. 'Since the conditions affecting friction vary considerably in time and space, there have been many experiments to determine the effect of friction on wind flow structure in the boundary layer. Results in general show that at low wind speeds (<5 m/sec) in mid-latitude the deflection is about 20 degrees to the left and the magnitude reduction factor is about 85% of the pure geostrophic wind.

In the model variations of frictional effects with latitude and wind speed are also included. Values computed from the function used in the model shown in Table 2 indicate a further reduction in speed from 85% to about 80% away from mid-latitudes (25°-45°N), increases in the angle of deflection toward the equator, and decreases in the angle of deflection at higher speeds. Next, these resultant winds are used to calculate a commensurate surface ocean flow field.

Table 2. --Reduction factor and angle of deflection of the wind to the left of pure geostrophic wind.

Latitude	Reduction	Wind speed (m/sec)										
		0	2.8	5.7	8.5	11.3	14.1	17.0	19.8	22.6	25.5	28.3
		Angle of deflection (°)										
67.0	0.801	17.5	17.4	17.2	16.9	16.4	15.8	15.0	14.1	13.0	11.8	10.5
66.0	0.803	17.6	17.5	17.3	16.9	16.4	15.8	15.0	14.1	13.1	11.9	10.5
65.0	0.806	17.6	17.6	17.4	17.0	16.5	15.9	15.1	14.2	13.1	11.9	10.5
64.0	0.808	17.7	17.6	17.4	17.1	16.6	15.9	15.1	14.2	13.1	11.9	10.5
63.0	0.810	17.8	17.7	17.5	17.1	16.6	16.0	15.2	14.2	13.1	11.9	10.5
62.0	0.812	17.9	17.8	17.6	17.2	16.7	16.0	15.2	14.3	13.2	11.9	10.5
61.0	0.814	17.9	17.9	17.6	17.3	16.8	16.1	15.3	14.3	13.2	11.9	10.5
60.0	0.817	18.0	18.0	17.7	17.4	16.8	16.2	15.3	14.4	13.2	12.0	10.5
59.0	0.819	18.1	18.0	17.8	17.4	16.9	16.2	15.4	14.4	13.3	12.0	10.6
58.0	0.821	18.2	18.1	17.9	17.5	17.0	16.3	15.5	14.5	13.3	12.0	10.6
57.0	0.823	18.3	18.2	18.0	17.6	17.1	16.4	15.5	14.5	13.4	12.0	10.6
56.0	0.826	18.4	18.3	18.1	17.7	17.1	16.4	15.6	14.6	13.4	12.1	10.6
55.0	0.828	18.5	18.4	18.2	17.8	17.2	16.5	15.7	14.6	13.4	12.1	10.6
54.0	0.830	18.6	18.5	18.3	17.9	17.3	16.6	15.7	14.7	13.5	12.1	10.6
53.0	0.832	18.7	18.6	18.4	18.0	17.4	16.7	15.8	14.8	13.5	12.2	10.6
52.0	0.834	18.8	18.7	18.5	18.1	17.5	16.8	15.9	14.8	13.6	12.2	10.7
51.0	0.837	18.9	18.8	18.6	18.2	17.6	16.9	16.0	14.9	13.7	12.2	10.7
50.0	0.839	19.0	19.0	18.7	18.3	17.7	17.0	16.0	15.0	13.7	12.3	10.7
49.0	0.841	19.2	19.1	18.8	18.4	17.8	17.1	16.1	15.0	13.8	12.3	10.7
48.0	0.843	19.3	19.2	19.0	18.5	17.9	17.2	16.2	15.1	13.8	12.4	10.8
47.0	0.846	19.4	19.3	19.1	18.7	18.0	17.3	16.3	15.2	13.9	12.4	10.8
46.0	0.848	19.6	19.5	19.2	18.8	18.2	17.4	16.4	15.3	14.0	12.5	10.8
45.0	0.850	19.7	19.6	19.4	18.9	18.3	17.5	16.5	15.4	14.0	12.5	10.8
44.0	0.850	19.8	19.8	19.5	19.0	18.4	17.6	16.6	15.5	14.1	12.6	10.9
43.0	0.850	20.0	19.9	19.6	19.2	18.6	17.8	16.8	15.6	14.2	12.7	11.0
42.0	0.850	20.2	20.1	19.8	19.3	18.7	17.9	16.9	15.7	14.4	12.8	11.1
41.0	0.850	20.3	20.2	19.9	19.5	18.9	18.0	17.0	15.8	14.5	12.9	11.2
40.0	0.850	20.5	20.4	20.1	19.6	19.0	18.2	17.2	16.0	14.6	13.0	11.3
39.0	0.850	20.6	20.6	20.3	19.8	19.2	18.3	17.3	16.1	14.7	13.1	11.4
38.0	0.850	20.8	20.7	20.4	20.0	19.3	18.5	17.5	16.2	14.8	13.2	11.5
37.0	0.850	21.0	20.9	20.6	20.1	19.5	18.6	17.6	16.4	15.0	13.4	11.6
36.0	0.850	21.2	21.1	20.8	20.3	19.7	18.8	17.8	16.5	15.1	13.5	11.7
35.0	0.850	21.4	21.3	21.0	20.5	19.8	19.0	17.9	16.7	15.2	13.6	11.8
34.0	0.850	21.6	21.5	21.2	20.7	20.0	19.1	18.1	16.8	15.4	13.7	11.9
33.0	0.850	21.8	21.7	21.4	20.9	20.2	19.3	18.3	17.0	15.5	13.8	12.0
32.0	0.850	22.0	21.9	21.6	21.1	20.4	19.5	18.4	17.1	15.7	14.0	12.1
31.0	0.850	22.2	22.1	21.8	21.3	20.6	19.7	18.6	17.3	15.8	14.1	12.2
30.0	0.850	22.4	22.3	22.0	21.5	20.8	19.9	18.8	17.5	16.0	14.3	12.3
29.0	0.850	22.7	22.6	22.2	21.7	21.0	20.1	19.0	17.7	16.1	14.4	12.5
28.0	0.850	22.9	22.8	22.5	22.0	21.2	20.3	19.2	17.8	16.3	14.6	12.6
27.0	0.850	23.1	23.0	22.7	22.2	21.5	20.5	19.4	18.0	16.5	14.7	12.7
26.0	0.850	23.4	23.3	23.0	22.4	21.7	20.8	19.6	18.2	16.7	14.9	12.9
25.0	0.850	23.6	23.5	23.2	22.7	21.9	21.0	19.8	18.4	16.8	15.0	13.0
24.0	0.842	23.9	23.8	23.5	23.0	22.2	21.3	20.1	18.7	17.2	15.4	13.4
23.0	0.834	24.2	24.1	23.8	23.2	22.5	21.6	20.4	19.1	17.5	15.7	13.7
22.0	0.826	24.5	24.4	24.1	23.5	22.8	21.9	20.7	19.4	17.8	16.1	14.1
21.0	0.818	24.8	24.7	24.4	23.8	23.1	22.2	21.1	19.7	18.2	16.4	14.5
20.0	0.810	25.1	25.0	24.7	24.1	23.4	22.5	21.4	20.1	18.5	16.8	14.8

Current Due to Wind

Computation of the surface current due to wind, like the computation of wind, is done with empirical methods in two parts. First, the pure drift current vector without friction is calculated, then the vector is corrected for frictional effects (due to the drag of the underlying water) which causes an angle of deflection of the current to the right of the wind looking in the direction of flow. With very high wind speeds the angle of deflection becomes very small and it also becomes smaller with decreasing latitude.

Although the surface current in the upper 3 m of water has been found to be generally about 2% of the average climatological (usually the monthly mean) wind speed, what is desired here is a relationship between the surface current and the instantaneous wind. For part one of our computations the formula of Witting (1909) satisfies this purpose within the limitations previously stated

$$C = k \sqrt{W}$$

where C is the speed (cm/sec) of the surface current due to wind, k is the coefficient of proportionality which equals 4.8 when it includes both the speed of the mixed layer and mass transport by waves at the surface, and W is the wind speed (m/sec).

In part two (frictional effects on wind generated surface current) the angle of deflection (BETA) is computed from a newly designed three part function which starts with the Hela (1952) relationship:

$$\text{BETA} = \text{BETA1} - \text{K2} \sqrt{W}$$

where BETA1 is the initial constant angle of deflection of 34 degrees for wind speeds approaching zero and K2 is a constant of 7.5. In the second part of the function, BETA1 is modified to decrease linearly with latitude with a slope of 0.2933 which was determined from a graphic fit of data for a constant wind speed of 6 knots between 30°N and 70°N (Hubert and Laevastu 1967, p.11)

$$\text{BETA1} = 34.0 - (\text{DLAT} \times 0.2933)$$

where DLAT equals 67 -latitude (ie. the difference between the top latitude of the model (67°N) and the variable latitude. The final part of the computation used the constraint that the angle of deflection (BETA) approached zero at 20.59 m/sec (40 knots) at all latitudes. With BETA equal to zero and solving for K2 this permits an adjustment of K2 for the change in BETA1 with latitude with the relationship

$$\text{K2} = \frac{\text{BETA1}}{\sqrt{20.59}}$$

Thus, the range of values of the angle of deflection of the surface current to the right of the wind over the latitudes (20°-67°N) of the model for wind speeds 0 to 40 knots as computed by the combined function are shown in Table 3. Now that we have a permanent flow and a flow due to wind we proceed to tune and combine the components of flow.

Tuning and Vector Summation of Currents

In this experimental model, it is desirable to have tuning factors because of the remaining small uncertainties in the theoretical computations of ocean currents. Thus, any

Table 3.--Angle of deflection of surface current to the right of the wind as a function of latitude and wind speed.

Wind speed	0	5	10	15	20	25	30	35	40	(kts)
	0.00	2.57	5.15	7.72	10.30	12.87	15.44	18.02	20.59	(m/sec)
Latitude	Angle of deflection (°)									
67.0	34.00	21.98	17.00	13.18	9.96	7.12	4.56	2.20	0.00	
66.0	33.71	21.79	16.85	13.07	9.87	7.06	4.52	2.18	0.00	
65.0	33.41	21.60	16.71	12.95	9.79	7.00	4.48	2.16	0.00	
64.0	33.12	21.41	16.56	12.84	9.70	6.94	4.44	2.14	0.00	
63.0	32.83	21.22	16.41	12.72	9.61	6.87	4.40	2.12	0.00	
62.0	32.53	21.03	16.27	12.61	9.53	6.81	4.36	2.10	0.00	
61.0	32.24	20.84	16.12	12.50	9.44	6.75	4.32	2.08	0.00	
60.0	31.95	20.65	15.97	12.38	9.36	6.69	4.28	2.06	0.00	
59.0	31.65	20.46	15.83	12.27	9.27	6.63	4.24	2.04	0.00	
58.0	31.36	20.27	15.68	12.16	9.19	6.57	4.20	2.03	0.00	
57.0	31.07	20.08	15.53	12.04	9.10	6.51	4.16	2.01	0.00	
56.0	30.77	19.89	15.39	11.93	9.01	6.44	4.12	1.99	0.00	
55.0	30.48	19.70	15.24	11.82	8.93	6.38	4.08	1.97	0.00	
54.0	30.19	19.51	15.09	11.70	8.84	6.32	4.04	1.95	0.00	
53.0	29.89	19.32	14.95	11.59	8.76	6.26	4.01	1.93	0.00	
52.0	29.60	19.14	14.80	11.47	8.67	6.20	3.97	1.91	0.00	
51.0	29.31	18.95	14.65	11.36	8.58	6.14	3.93	1.89	0.00	
50.0	29.01	18.76	14.51	11.25	8.50	6.08	3.89	1.87	0.00	
49.0	28.72	18.57	14.36	11.13	8.41	6.01	3.85	1.85	0.00	
48.0	28.43	18.38	14.21	11.02	8.33	5.95	3.81	1.84	0.00	
47.0	28.13	18.19	14.07	10.91	8.24	5.89	3.77	1.82	0.00	
46.0	27.84	18.00	13.92	10.79	8.15	5.83	3.73	1.80	0.00	
45.0	27.55	17.81	13.77	10.68	8.07	5.77	3.69	1.78	0.00	
44.0	27.25	17.62	13.63	10.56	7.98	5.71	3.65	1.76	0.00	
43.0	26.96	17.43	13.48	10.45	7.90	5.65	3.61	1.74	0.00	
42.0	26.67	17.24	13.33	10.34	7.81	5.58	3.57	1.72	0.00	
41.0	26.37	17.05	13.19	10.22	7.72	5.52	3.53	1.70	0.00	
40.0	26.08	16.86	13.04	10.11	7.64	5.46	3.49	1.68	0.00	
39.0	25.79	16.67	12.89	10.00	7.55	5.40	3.45	1.67	0.00	
38.0	25.49	16.48	12.75	9.88	7.47	5.34	3.42	1.65	0.00	
37.0	25.20	16.29	12.60	9.77	7.38	5.28	3.38	1.63	0.00	
36.0	24.91	16.10	12.45	9.65	7.30	5.22	3.34	1.61	0.00	
35.0	24.61	15.91	12.31	9.54	7.21	5.16	3.30	1.59	0.00	
34.0	24.32	15.72	12.16	9.43	7.12	5.09	3.26	1.57	0.00	
33.0	24.03	15.53	12.01	9.31	7.04	5.03	3.22	1.55	0.00	
32.0	23.73	15.34	11.87	9.20	6.95	4.97	3.18	1.53	0.00	
31.0	23.44	15.15	11.72	9.09	6.87	4.91	3.14	1.51	0.00	
30.0	23.15	14.96	11.57	8.97	6.78	4.85	3.10	1.50	0.00	
29.0	22.85	14.77	11.43	8.86	6.69	4.79	3.06	1.48	0.00	
28.0	22.56	14.58	11.28	8.75	6.61	4.73	3.02	1.46	0.00	
27.0	22.27	14.40	11.13	8.63	6.52	4.66	2.98	1.44	0.00	
26.0	21.97	14.21	10.99	8.52	6.44	4.60	2.94	1.42	0.00	
25.0	21.68	14.02	10.84	8.40	6.35	4.54	2.90	1.40	0.00	
24.0	21.39	13.83	10.69	8.29	6.26	4.48	2.87	1.38	0.00	
23.0	21.09	13.64	10.55	8.18	6.18	4.42	2.83	1.36	0.00	
22.0	20.80	13.45	10.40	8.06	6.09	4.36	2.79	1.34	0.00	
21.0	20.51	13.26	10.25	7.95	6.01	4.30	2.75	1.32	0.00	
20.0	20.21	13.07	10.11	7.84	5.92	4.23	2.71	1.31	0.00	

combination of weighting factors may be selected here as a linear multiplier of either the permanent geostrophic current component or the wind current component. These adjustments will be particularly useful in computing longer term effects on progressive vector tracks for a variety of different conditions and comparing these results to obtain the best match to actual measurements from drifters that have been tracked in the ocean. Refer to Run Instructions section to set up tuning factors such as doubling the wind effect or setting wind equal zero to observe the pure permanent flow.

The total current vector field is now produced as a vector sum of the tuned geostrophic and wind current fields components of flow. This field is stored on Disk File 55 for later comparison to the field in which the vectors near shore have been adjusted for boundary conditions as described in the next section.

Total Currents With Boundary Effects

The final step in obtaining a vector field of total currents representative of this particular day in history is to call subroutine CSTROT(KYR, IMO, IDY, OCUS, OCVS, NOSTOF). This subroutine modifies the vectors within two grid lengths of the coast all along the entire coastline of the model for simulated boundary effects. These points have been preselected and read in as described above in the input section.

In the subroutine CSTROT current vectors at these points will be reduced in magnitude by the exponential function

$$\text{REDFAC (\%)} = 1.0 - (1/e^x)$$

where $x = \text{GDIST}(I,J)$, the normal distance from the grid point to the coastline in grid units.

Next the angle of the current vector is rotated depending upon its angle of incidence upon the coast relative to the line normal to shore. Looking toward shore, normal to the coastline trend, the current vectors which point to the left are rotated further toward the left and the vectors which point to the right are rotated farther toward the right. The maximum angle of rotation (45 degrees) is also diminished with distance from shore by the exponential factor, $1/e^X$, and a new function was designed to maximize the angle of rotation at an incident angle of 30 degrees. Away from 30 degrees the angle of rotation' is decreased proportionately to zero as the angle of incidence of zero degrees or 90 degrees are approached.

Transport (Progressive) Vectors

Now that a final velocity field is available a transport vector may be computed at any point on the grid, even between grid points. Transport vectors represent the distance traveled in one day proceeding at the velocity of the calculated surface current at that location. The starting point and end point of the vector are specified in grid units (I,J pairs), and the end point of this days vector is saved to be the starting point of tomorrows transport vector during the next pass through the loop.

The unique part of this routine is that the velocity at each starting point each time through the loop is interpolated spatially within the field using the spline surface fit. This

gives a much greater degree of precision to progressive vector tracks formed by connecting these successive transport vectors for NDAY5 compared to the tracks formed by the simpler method of connecting the transport vectors from a starting grid points which remains fixed in the field. Each day the transport vectors from each of the selected starting points are stored in the same order on Disk File 60. Thus, it is important to keep track of the number of starting points selected for later entry in the plot program. Note, that the plot program is organized to read and plot all the first day vectors then all the second day vectors and continue in this manner, not to plot one entire progressive vector track at a time followed by the next entire track, etc.

The loop is now complete and will be entered again for another day of computations provided another day of sea level pressure data was scheduled by the control cards of the WFL deck. Output files are added onto each time through the loop and are "CLOSED" only after NDAY5 when this program is stopped. Refer to the plot program instructions below for graphic output of results on a Tektronics 4014 terminal or CALCOMP Plot.

GRAPHICS OUTPUT PROGRAMS

Thanks to the availability of the "GRIDS" package of Fortran subroutines which was designed for the Burroughs 7811 computer to graphically display gridded data in geographical coordinates (latitude-longitude) on a Tektronix CRT Terminal or Calcomp Plotter (Swan 1983), the task of producing maps of our square grid data fields was greatly facilitated. The following describes our Fortran program which uses the CONTOUR and VECTOR graphics displays of GRIDS to show the model results first on an ocean-wide (40X104) grid then on a zoom (25X35) grid for the Gulf of Alaska.

Single day results of particular interest are the spatial relationships found among sea level pressure contours, wind vectors, and wind generated surface ocean current vectors. Choices of plots are 1) any combination of the five, single day vector fields or 2) one map of progressive transport vector tracks versus time. These are plotted for a specified number of days (NDAYS). A second version of the same program with more localized geographic coordinates is used for zoom plots (25X35) such as the Gulf of Alaska version presented here. Selections of which fields and the number to be plotted are made by editing your choices into the WFL code of the run job file each time the program is run. Now it is appropriate to examine the basic results of the model on an ocean-wide graphics scale.

Plots on 40X104 Grid

Ocean wide plots over the entire 40X104 grid are produced by the program "M/PLOT/AUTO" (listed in Appendix 2). If any modifications are made to this program the GRIDS subroutines must be bound again to the new object code by running the WFL job "M/PLOT/AUTO/BINDER" (Program 2). Plots are obtained by running the WFL job "M/PLOT/AUTO/WFL" (Program 3) after editing in your selections. The following describes the sequential flow of the plot program: the setup, the NDAYs plotting loop, and making your selections.

Setup within the Fortran code includes the standard GRIDS initialization with the reading of latitudes, longitudes (disk Files 30 and 31), and the land-sea table (disk Files 13 and 14) of the (40X104) grid and a call to STGRID to set up common arrays for all the GRIDS subroutines. Then latitude lines on the map are computed or read from previously stored disk file (set flag NOLATS=1 to read disk File 36) as is the coastline file (set flag IFAST=0 to read File 71) and the 50, 100, 200, and 2000 m bathymetry files (set flag NOCONV=1 to read Files 66, 67, 68, and 69). The alternative to reading the existing files is included here to allow for future versions which change the map size or area of interest and require recalculation of the map parameters. To create and store new disk files of latitude lines, coastline, and bathymetry just reverse the values set for the flags (NOLATS=0 IFAST=1, AND NOCONV=0). Next, the initial starting points for progressive vectors are read from File 28, and the permanent geostrophic ocean U and V current vector fields are read in from

Program 2. --Listing of Binder Job which binds and compiles the outputs
for the 40X104 grid (File Title="M/PLOT/AUTO/BINDER").

```
1  ?BEGIN JOB INGRAHAM/BINDER/M/PLOT/AUTO;
2  QUEUE=20;
3  COMPILE OBJECT/M/PLOT/AUTO FORTRAN LIBRARY;
4  COMPILER FILE TAPE(KIND=DISK,TITLE=M/PLOT/AUTO);
5  ?FORTRAN DATA
6  $SET MERGE
7  $RESET LIST
8  ?
9  BIND M/PLOT/AUTO/BOUND BINDER LIBRARY;
10 BINDER FILE HOST(TITLE=OBJECT/M/PLOT/AUTO);
11 ?BINDER EBCDIC
12 $RESET LIST
13 BIND=FROM GRIDS40104/=,(REFM0070)REFMOFF/= ON TYVO,
14      (REFM0070)PLOTCOMP87/= ON TYVO;
15 ?END JOB
```

Program 3.--listing of Word Flow Language (WFL) Program which runs the graphic outputs of the 40X104 grid (File Title="M/PLOT/AUTO/WFL").

```
1  ?BEGIN JOB INGRAHAM/M/PLOT/PGRV/78/BOUND;
2  QUEUE=20;
3  RUN M/PLOT/AUTO/BOUND;
4  FILE FILE10=DIR96202;
5  FILE FILE28=P/FIG/40X104;
6  FILE FILE51=CURRENTS/40X104/DSLPL/78/M7TOM12/D17START;
7  FILE FILE52=CURRENTS/40X104/WIND/78/M7TOM12/D17START;
8  FILE FILE53=CURRENTS/40X104/SFCOCN/78/M7TOM12/D17START;
9  FILE FILE54=CURRENTS/40X104/WOODD10/78/M7TOM12/D17START/CSTROTATD;
10 FILE FILE55=CURRENTS/40X104/WOODD10/78/M7TOM12/D17START;
11 FILE FILE60=CURRENTS/40X104/PROGRESVECT/78/M7TOM12/D17START/W10DD20
12     /FIG;
13 FILE FILE5(KIND=READER);
14 ?DATA
15     1.0=====EVERY GRID POINT PLOT SLP WITH CURRENTS
16     0.0=====1. PLOT WIND VECTORS (1.0=PLOT)
17     0.0=====2. PLOT WIND CURRENT VECTORS
18     0.0=====3. PLOT SLP CONTOURS WITH WIND AND CURRENT VECTORS
19     0.0=====4. PLOT GEOSTROPHIC CURRENT VECTORS
20     0.0=====5. PLOT WIND+THERMAL+GEOSTROPHIC VECTORS
21     0.0=====6. PLOT 5 WITH COASTAL BENDING
22     1.0=====7. PLOT PROGRESSIVE VECTORS
23     0.0=====PLOT OF 5 OVER 6 TO SEE DIFFERENCE IN BENDING (2.0=PLOT)
24 ?
25 ?END JOB
```

File 50. The final part of the set up is reading the card File 5 in the WFL run job (Program 3) to define the selected flags for the desired plots and the other run parameters. These flags and their respective settings to be edited into the WFL job are

EVGRPT -- On all vector plots chosen during this run, set EVGRPT =1 to plot arrows at every grid point on the grid; or else, set EVGRPT=0 to plot every other arrow on the grid which reduces the visual confusion from overlapping arrows.

PLWIND -- Set PLWIND=1 to plot the wind vector field in m/sec; or else set PLWIND=0.

IPLSLP -- Set IPLSLP=1 to plot sea level pressure contours on top of the wind and ocean current vector fields if they are chosen; or else set IPLSLP=0.

PLOCW -- Set PLOCW=1 to plot the ocean current vector field due to wind in cm/sec; or else set PLOCW=0.

PLOCDD -- Set PLOCDD=1 to plot the permanent ocean current vector field (Delta-D) in cm/sec; or else set PLOCDD=0.

PLOCS -- Set PLOCS=1 to plot the sum total surface ocean current vector field before adjustments for coastal boundary conditions; or else set PLOCS=0.

PLSOCR -- Set PLSOCR=1 to plot the total surface ocean current vector field; or else set PLSOCR=0. This series of fields are the final daily result of the model,

NOPLT -- Set NOPLT=AN INTEGER to skip plotting THIS NUMBER MINUS ONE vector fields; or else set NOPLT=1 to start the plotting at #1.

PROGV -- Set PROGV=1 to plot progressive transport vectors for NDAYS; or else set PROGV=0. Only one PROGV plot may be run per job, thus all other plot selections must be zero.

IJEND -- Set (IJEND=INTEGER) the number of progressive vector starting points stored in File 60 (the maximum # available to plot). This may be different from the number read in from File 28 which are the number (IIEND) to be actually plotted.

WFAC -- Set WFAC=DECIMAL NUMBER, the wind tuning factor that was used in this model run (usually = 1.0).

DDFAC -- Set DDFAC=DECIMAL NUMBER, the Delta-D tuning factor that was used in this model run (usually= 1.0). Both WFAC and DDFAC are used only in headings to indicate which run this plot displays.

NDAYS -- Set NDAYS=INTEGER, the number of days you decide to be plotted. This controls the amount of run time.

FILE -- Edit all file names to correspond to those existing disk files you desire to be plotted.

Each time through the NDAY5 (number of days to be plotted) loop there are two options. In the first option one or all of the five vector fields may be plotted in sequence on a Tektronics 4014 Terminal and a hard copy produced (names of flags to be set (1 or 0) in Program 3 are PLWIND, PLOCW, PLOCDD, PLOCS, or PLSOCR). The second option is to draw a single plot of one or more progressive transport vector tracks one arrow at a time for NDAY5 (flag PROG=1; all other plot flags=0). Examples of these model output plots on the ocean-wide (40X104) grid scale are shown below starting with the permanent flow field and then the daily variable flow fields for selected dates.

Geostrophic Currents

First, the permanent geostrophic current vector field over the whole grid is shown at every other grid point in Figure 4. On this large grid the major features of the surface ocean circulation in the North Pacific and Bering Sea are clearly evident. Notice the region of strong zonal flow of 20 to 40 cm/sec southwest of Japan in the Kuroshio Current. This flow gradually widens and diminishes to less than 5 cm/sec as it proceeds eastward generally along latitude lines in the Kuroshio Extension. The great Divergence occurs at the North American Continent (about 47°-49°N) with local acceleration to speeds of about 5 to 10 cm/sec in the two branches--southward in the coastal upwelling region of the California Current and northward in the coastal downwelling region of the Alaska Current. The next highest current speeds of 5 to 15 cm/sec are seen in the

PERMANENT

SURFACE OCEAN CURRENTS
(GEOSTROPHIC COMPONENT)

10 CM/SEC
→

BATHYMETRY

50 M
100 M ----
200 M ---
2000M ———

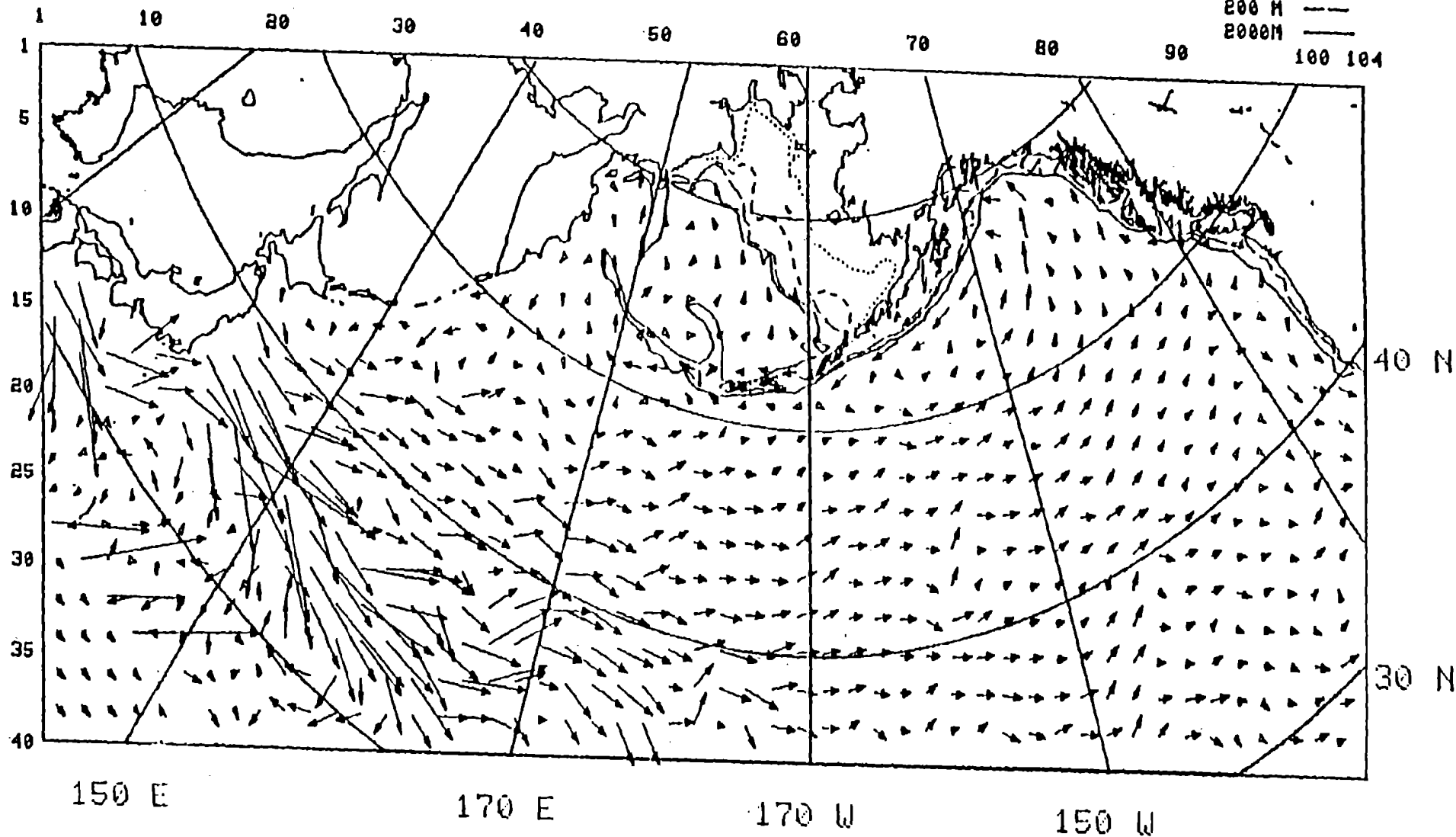


Figure 4.--Surface Geostrophic current vector field on 40x104 grid. Permanent flow component was calculated from anomaly of dynamic height data (0/2000 db).

westward return flow of the Alaskan Stream (another western boundary current) which accelerates due to the southward displacement of the westward flow along the Alaska Peninsula. The westward extension of the Alaskan Stream turns northward in the western Aleutian Islands and proceeds cyclonically (counterclockwise) around the Bering Sea Basin before joining the Oyashio Current, a third western boundary current flowing southward along the Kamchatka Peninsula and Kuril Island coasts. Thus, the appearance of a long closed loop in the mean circulation around the Subarctic North Pacific Region is shown completed as this flow turns eastward joining the Kuroshio Extension. The accuracy of these vectors in direction is quite acceptable to show major features, but speeds are suspect particularly in regions of concentrated flow such as the Alaskan Stream in which portions of its narrow high velocity core lie unresolvable between grid points of the model.

A tuning factor constant (DDFAC) is included as a linear multiplier to adjust the magnitude of the permanent flow field. This is helpful in model calibration exercises when model outputs are compared to actual drift current measurements (see below). Particularly north of 40°N the need for augmentation of the permanent flow is expected, because at the chosen reference level for zero motion in the geostrophic calculations of 2000 decibars (db) a deeper (about 2000 m), weak flow in the same direction as the surface current may still exist locally (Ingraham and Favorite 1968). South of 40°N , however, there is also some question as to the appropriate reference level which may even be

shallower than 2000 db due to the more complex water structure in the subtropical watermass. Despite the above considerations, this geostrophic current vector field is a good approximation to the basic permanent flow; but it is anticipated to improve the model by adding seasonal geostrophic flow in future versions. The other major component of flow is the wind-induced surface current which will be added to the permanent flow field to make up the final or total flow field.

Wind Currents

Surface ocean current vectors due to wind are expected to show a great diversity in daily flow patterns, generally different in both magnitude and direction from the permanent flow yet with some localized areas of strong enhancement or opposition on a short-term basis (generally 1-5 days). It takes two steps to generate a surface current field from a sea level pressure field as discussed above. Beginning with a sea level pressure field from an arbitrarily selected day in history, 17 July 1978, first a wind field is calculated (Fig. 5). Over such a wide expanse of ocean most of the common meteorologic features (cyclones, anticyclones, jets, strong shear zones, calms, etc.) will generally appear somewhere on the map. Conditions look about as expected for July with its typical pattern of summer low pressure areas over the Asian Continent which move eastward along the Aleutian Islands then northward across the Bering Sea and also a persistent Northeast Pacific High off the west coast of the United States. The key feature to notice here in

SEA LEVEL PRESSURE (MB) 78 - 7 - 17

SURFACE WIND VECTORS

10 M/SEC
→

BATHYMETRY

50 M
100 M - - - - -
200 M - - - - -
2000M - - - - -

100 104

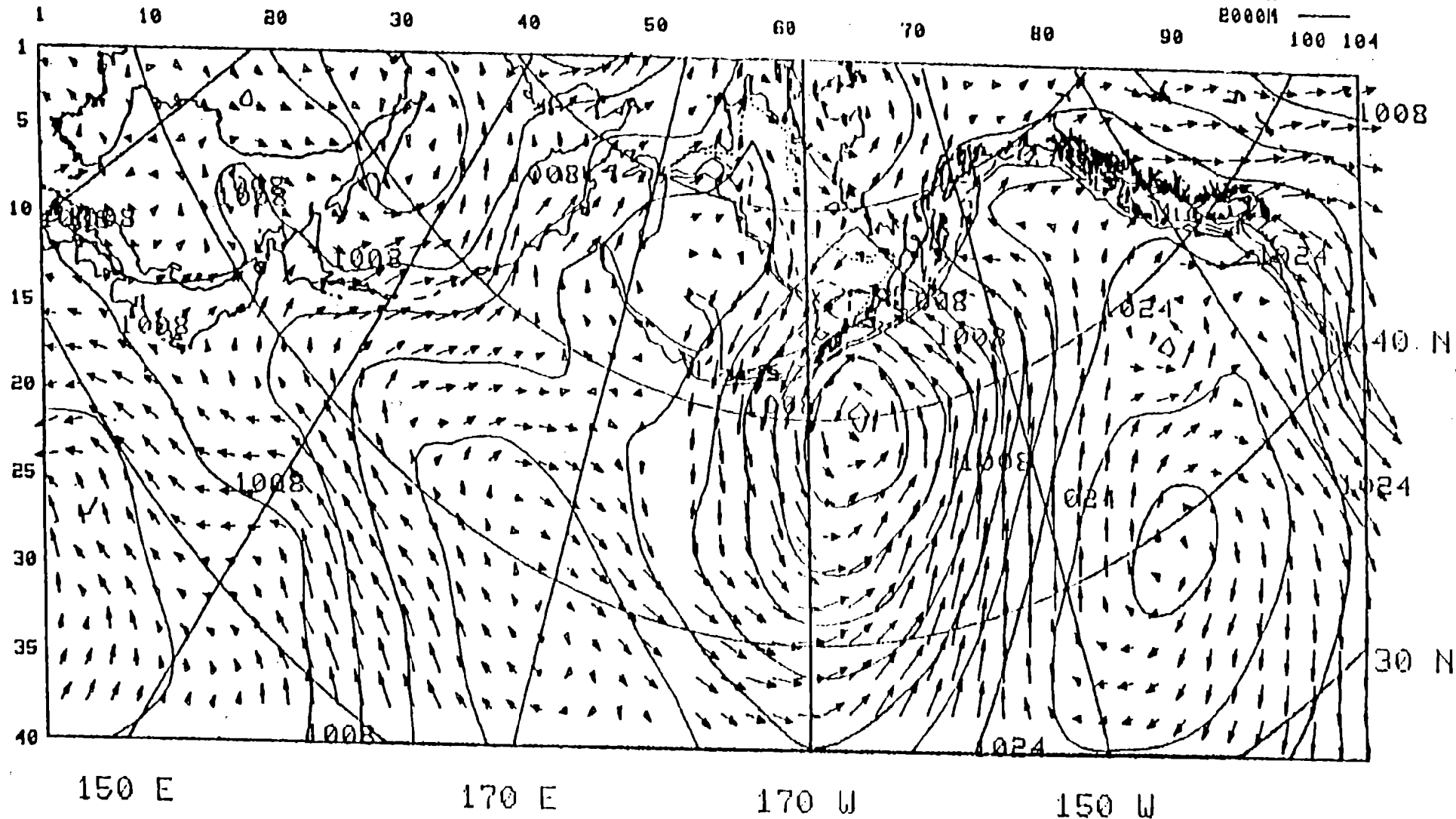


Figure 5.--Wind vector field (m/sec) calculated from arbitrarily selected sea level pressure field for 17 July 1978. Note cross-isobar flow toward lower pressure.

this one day chart is the cross-isobar wind flow. Facing with the direction of flow the angle of wind deflection to the left of the sea level pressure isobars varies in a complex manner with latitude and wind speed but is generally about 20 degrees.

By contrast in our second step the calculated surface ocean current has a deflection to the right of this wind which also varies with latitude and wind speed in a complex manner but with a slightly smaller magnitude of about 15 to 20 degrees. This gives the surface ocean current vectors (Fig. 6) an appearance of being more closely parallel to the sea level pressure isobars than the wind vectors although the speed of the ocean flow is only about 1 to 2% of the wind speed. The nature of this relationship and the bending of the ocean current vectors which impinge on the coast are more clearly seen below in zoom plots of this same data in the expanded area of the western Gulf of Alaska (see below for discussion of calculations).

Total Currents

Adding the permanent current velocity field and wind current velocity field produces the final resultant, the total current vector field for 17 July 1987 (Fig. 7). This is a one day example of the variety of complex local flow patterns that can occur in the Subarctic Region of the North Pacific Ocean. The average relatively smooth eastward flow along 40 N seen previously in the permanent component appears to be masked by very sinuous short-term variations. The effect of moving storms is seen in Figure 8 where plots of total currents for the six

SEA LEVEL PRESSURE (MB) 78 - 7 - 17

SURFACE OCEAN CURRENTS

(WIND COMPONENT)

10 CM/SEC



BATHYMETRY

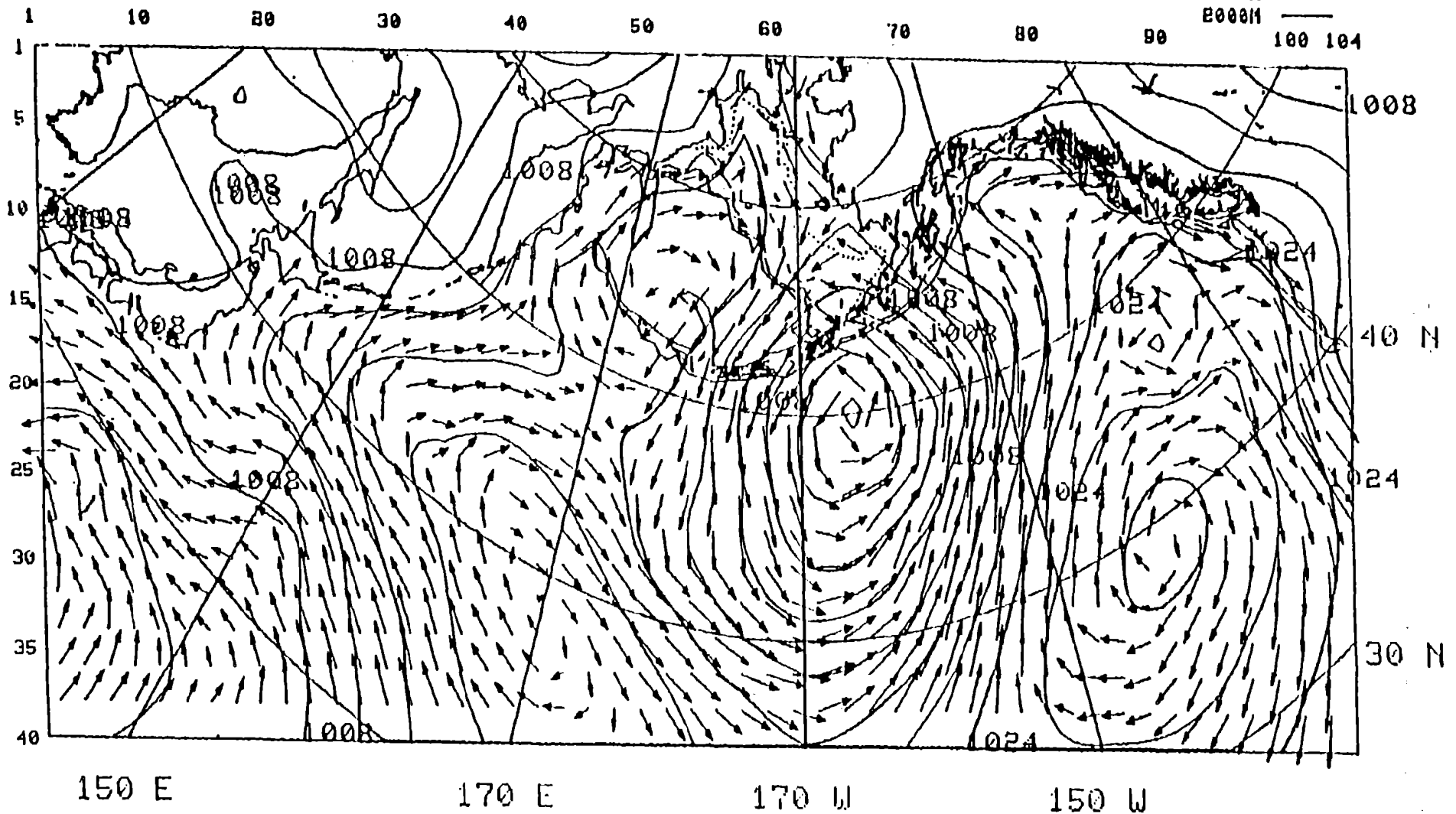
50 M

100 M - - - -

200 M - - - -

2000M ————

100 104



150 E

170 E

170 W

150 W

Figure 6.--Surface ocean current vector field (cm/sec) calculated from wind vector field (Fig. 5). Note: Deflection to right of wind alines currents closely with the isobars.

78 - 7 - 17

SURFACE OCEAN CURRENTS
(WIND + THERM + GEOS COMPONENT)
(X1.00, X 0.00, X 1.00)
10 CM/SEC
→

BATHYMETRY
50 M
100 M ----
200 M ---
2000M —

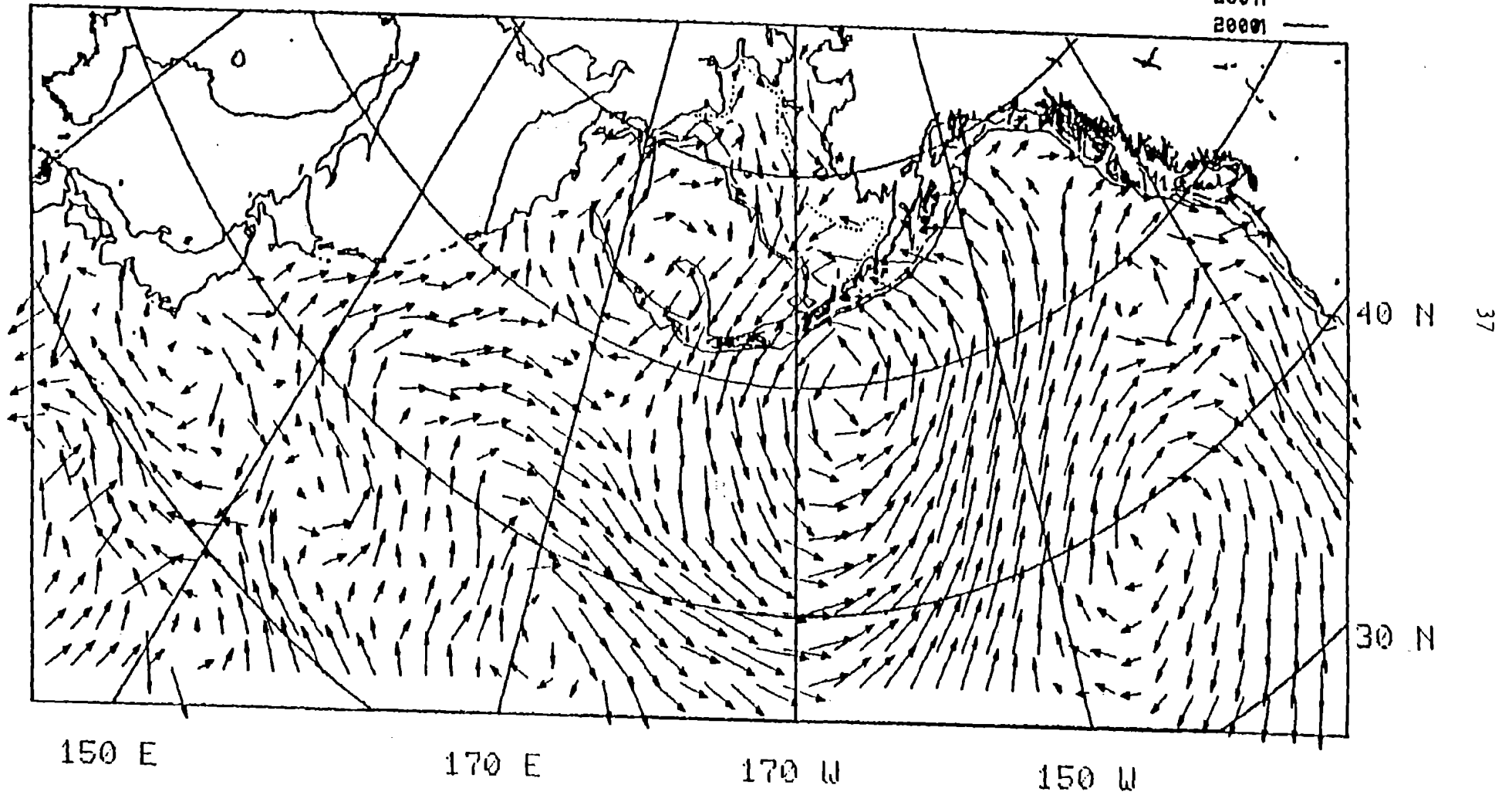


Figure 7.--Total surface ocean current vector field (sum of geostrophic plus wind current) for 17 July 1978. Final results of model computations.

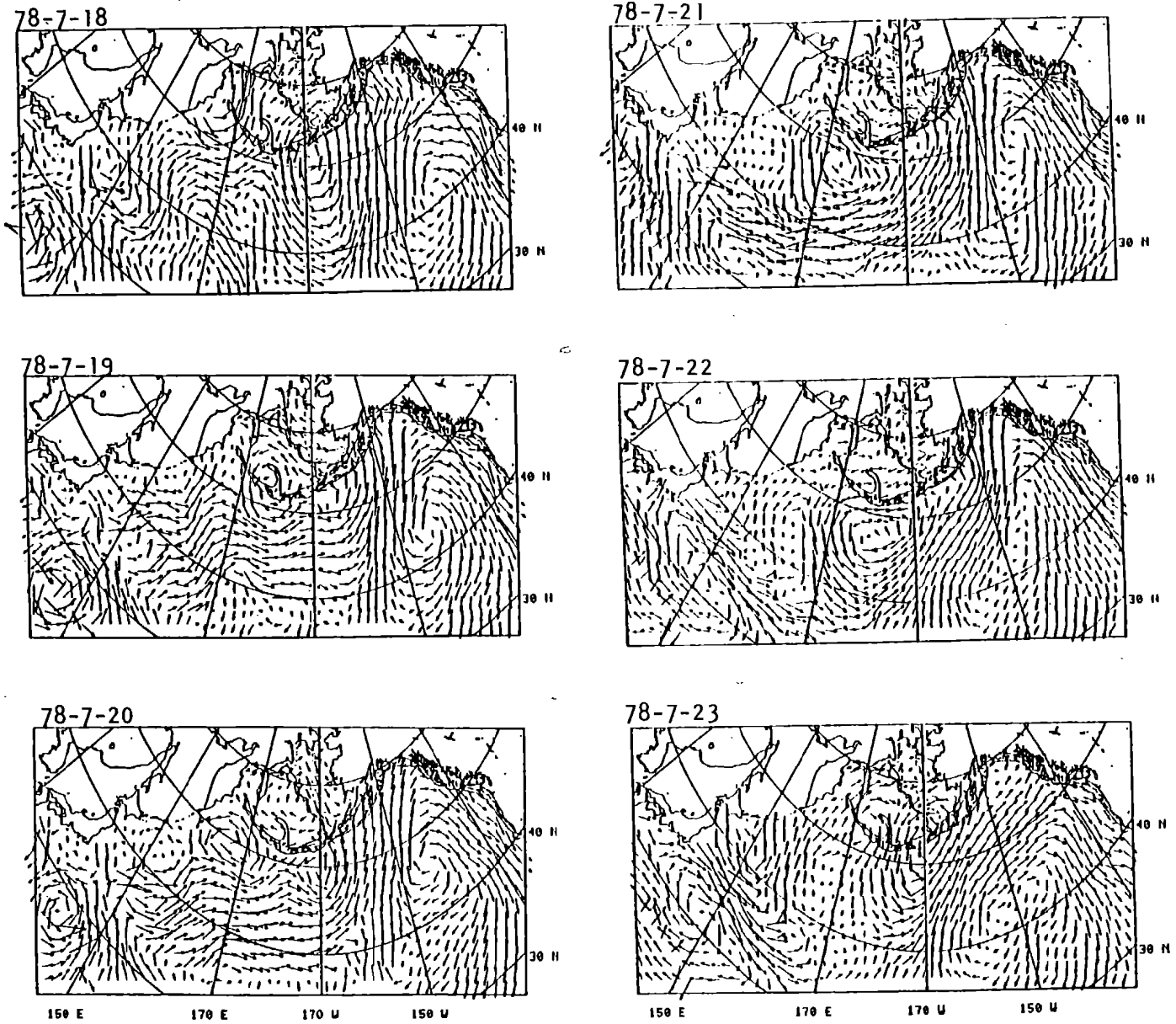


Figure 8.--Total surface currents, July 18 to 23, 1978. Model results for 6 subsequent days following 17 July 1978 (Fig. 7).

subsequent days show the nature of variability on a weekly basis. Some complete reversals of flow lead to changing conditions of convergence and divergence near shore along the coastline.

Considering the sheer number of possible model output fields (wind, wind current, and total current) the task of analysis seems a bit overwhelming. Our present data base of 41 years of daily data allows about 15,000 possible charts for each field. Let us recall that the reason for making the model was to trace the effects of all this short term variability over time. Therefore, the transformation from current vector fields to progressive transport (distance moved in one day) vector strings from any point in the model gives us a good simulation of actual ocean drift conditions for as many days as we choose and the ability to tune model results to actual drifter tracks.

Progressive Vector Tracks

Progressive transport vector tracks for July through December 1978 (Fig. 9(a)) starting from arbitrarily selected starting points give us a Lagrangian perspective of flow and an estimate of the daily trajectory of a parcel of water near the surface over a 5-month period under the influence of the permanent long-term mean geostrophic flow and daily winds. Of particular interest is the band of strong zonal flow between about 35°N and 45°N which shows a rather uniform eastward drift of surface water across most of the North Pacific Ocean with short-term (daily to a few weeks) fluctuations toward the north or south or reversals superimposed on the main eastward flow.

DAILY PROGRESSIVE VECTOR DISTANCE

FROM 78 - 7 - 17
TO 78 - 12 - 31

SURFACE OCEAN CURRENTS

(WIND + THERM + GEOS COMPONENT)
(X1.00, X 0.00, X 1.00)

BATHYMETRY

50 M
100 M ----
200 M ---
2000M ———

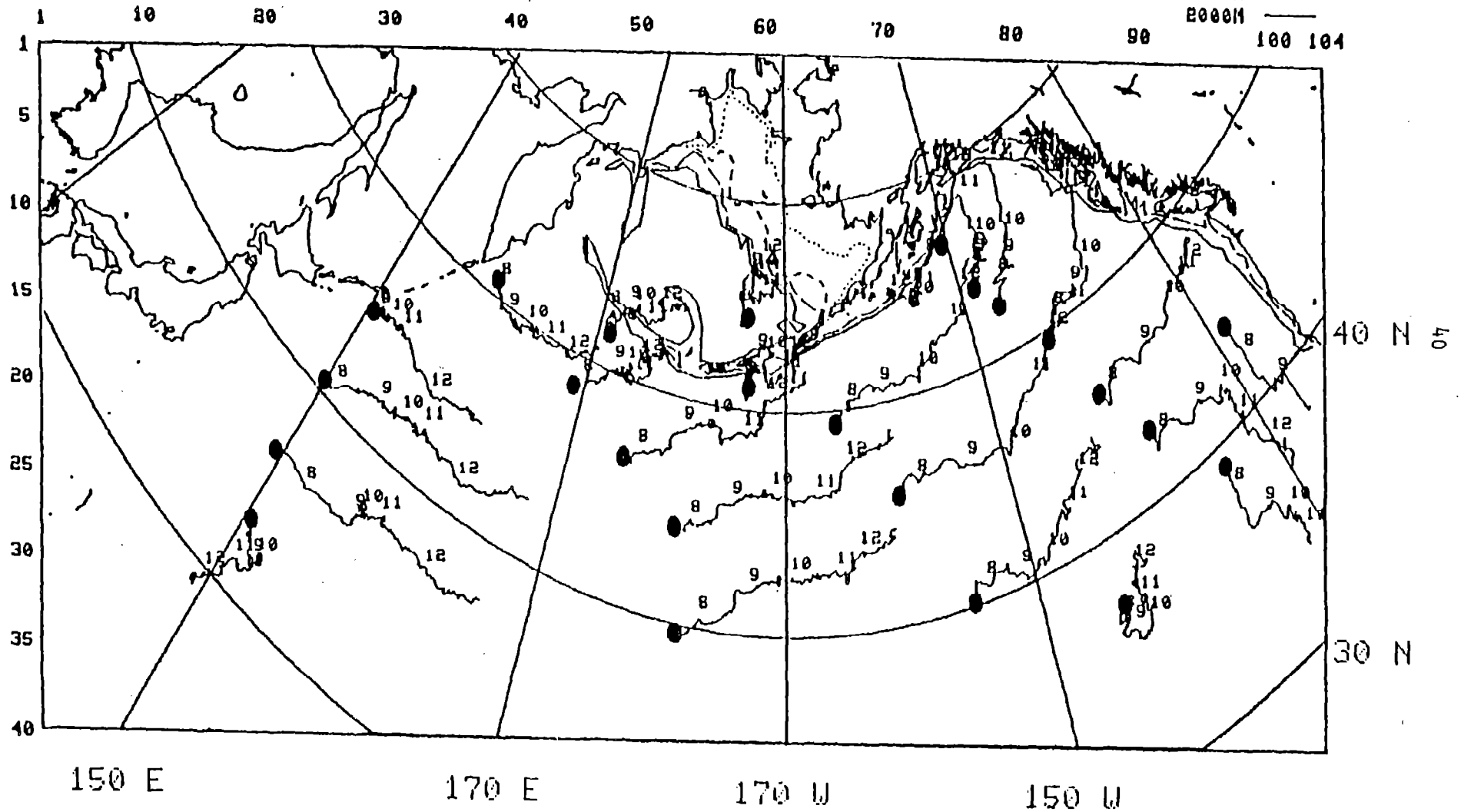


Figure 9(a). --Progressive daily transport vectors-Total surface ocean transport (17 July to 31 December 1978). Numbers along tracks correspond to the first day of the indicated month.

Many of these fluctuations seem to cancel each other in the north-south direction. Although the net surface drift in Figure 9(a) appears to be relatively uniform all across the ocean (speeds about 500-600 nautical miles per 165 days or about 6-8 cm/sec), the individual contribution from the permanent flow (Fig. 9(b)) and the wind current (Fig. 9(c)) changes considerably within this region. Off Japan at least during this summer the eastward flow is nearly all geostrophic with little wind contribution, but going eastward the geostrophic component gradually diminishes while the wind component increases. In the western Gulf of Alaska the geostrophic southwest flow of the Alaskan Stream (Fig. 9(b)) is in direct opposition to the northeastward flowing wind current (Fig. 9(c)). Some meandering and reversing appear in the overall cyclonic flow in the western Gulf of Alaska, while relatively stagnant conditions are present in the center of the oblong gyre between the westward and eastward flows. The time period for this sample model run was selected to coincide with the field data from satellite tracked drifting buoys released there in July and tracked through December 1978 (Reed 1980). These progressive transport vectors will be examined in greater detail in the next section on the zoom plot program.

Zoom Plot of Gulf of Alaska on 25X35 Grid

The purpose for drawing zoom plots, as the name indicates, is to zoom in on or expand a small area of interest and more clearly view model output. Zoom plots also assist in model tuning

DAILY PROGRESSIVE VECTOR DISTANCE

FROM 78 - 7 - 17
TO 78 - 12 - 31

SURFACE OCEAN CURRENTS

(WIND + THERM + GEOS COMPONENT)
(X 0.00, X 0.00, X 1.00)

BATHYMETRY

50 M
100 M - - - -
200 M - - - -
2000M - - - -

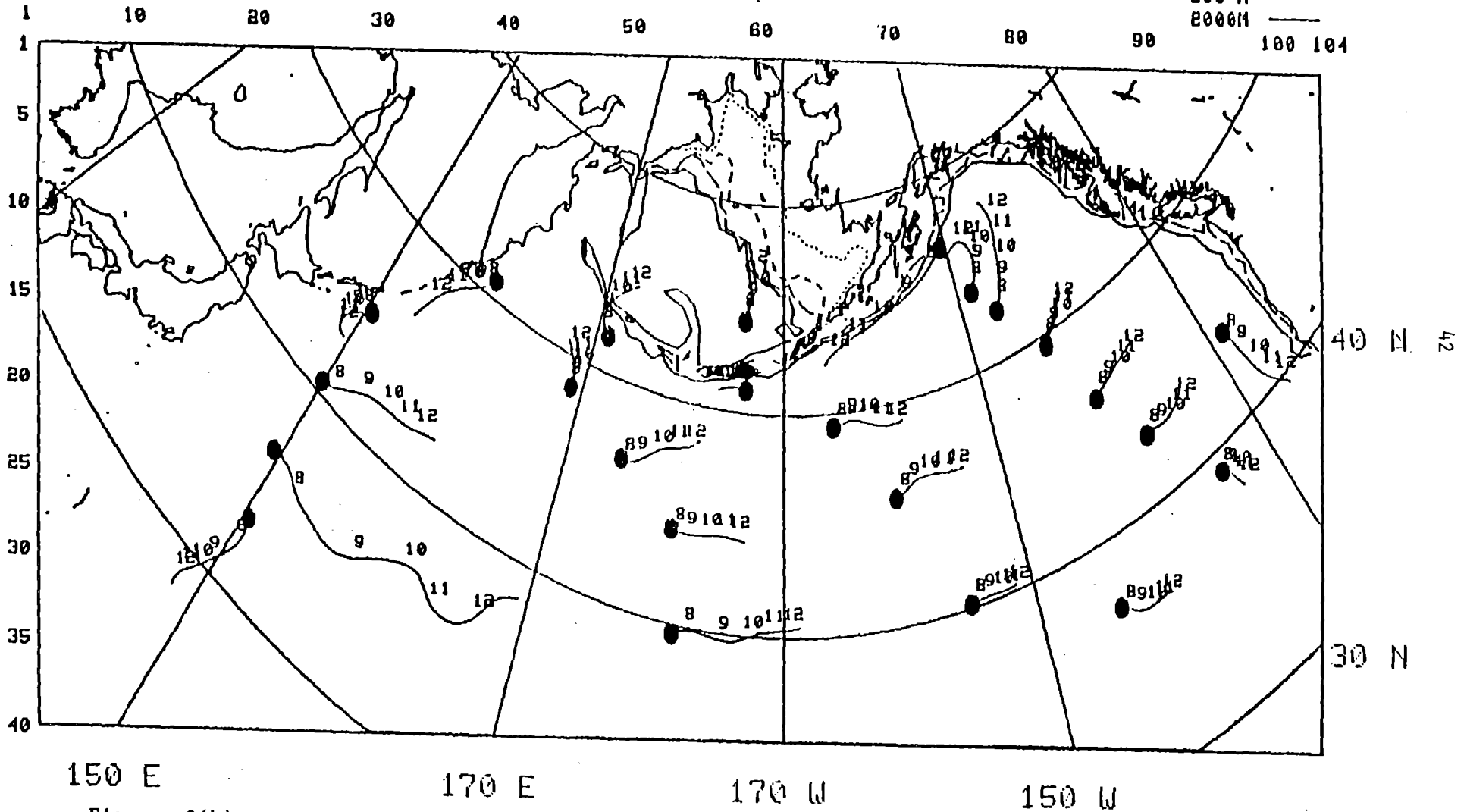


Figure 9(b).--Progressive daily transport vectors-Permanent geostrophic flow only (WFAC=0.0)(17 July to 31 December 1978).

DAILY PROGRESSIVE VECTOR DISTANCE

FROM 78 - 7 - 17

TO 78 - 12 - 31

SURFACE OCEAN CURRENTS

(WIND + THERM + GEOS COMPONENT)

(X1.00, X 0.00, X 0.00)

BATHYMETRY

50 M (dotted line)

100 M - - - - (dashed line)

200 M - - - - (dash-dot line)

2000M ——— (solid line)

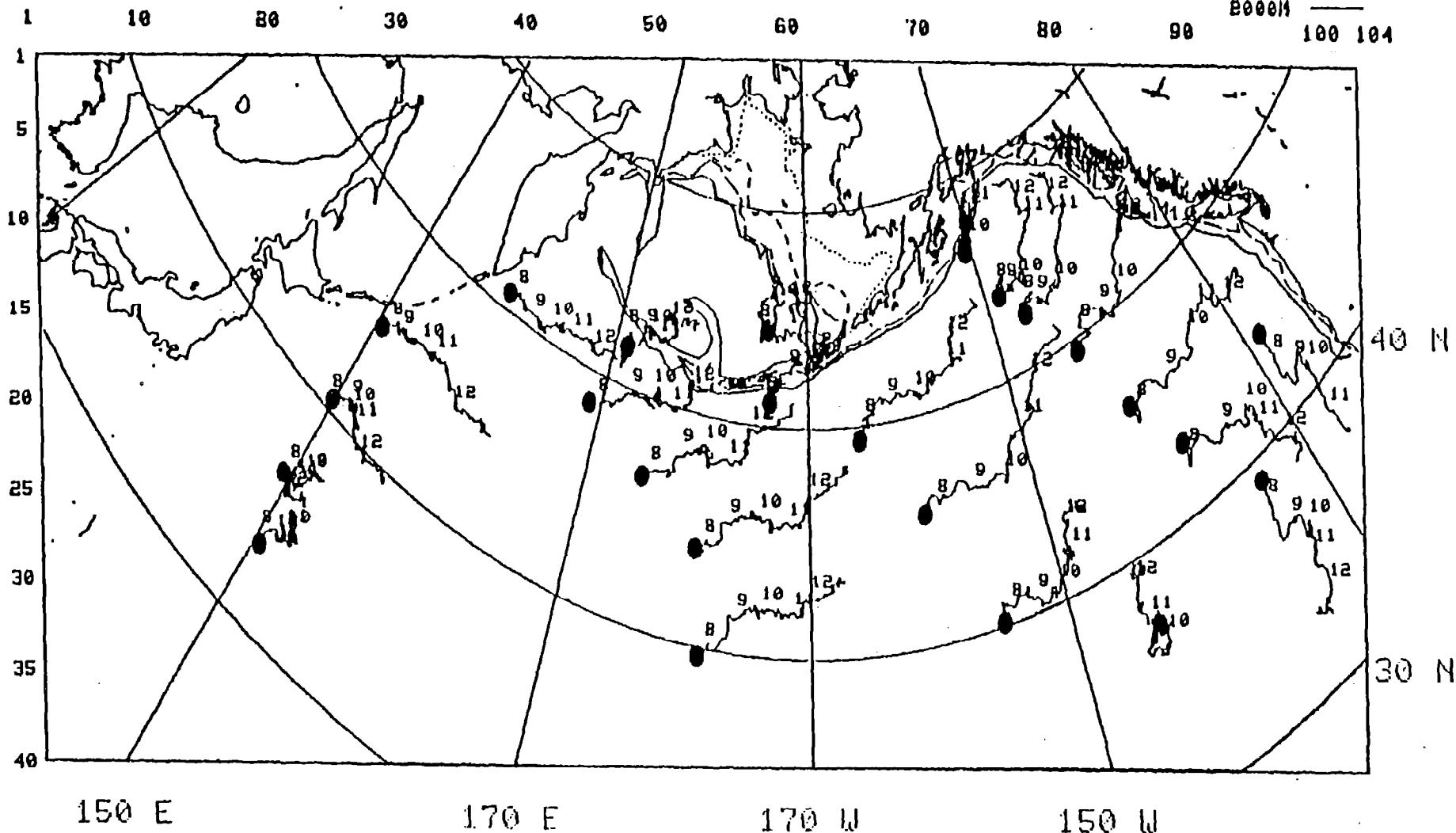


Figure 9(c). --Progressive daily vectors-Surface current due to wind only (DDFAC=0.0) (17 July to 31 December 1978).

and verification by displaying fine-scale details for comparisons between directly measured drift tracks and the progressive transport vector tracks computed by the model. Our first level of zoom is discussed here, an expansion of a 25X35 portion of the main grid (Fig. 10) in the Gulf of Alaska. The indices of the large grid, I=5-29 and J=56-90, correspond to zoom plot indices, I=1-25 and J=1-35, respectively.

Because the zoom plot program is essentially the same program as the ocean-wide plot program (already listed in Appendix 2), with some index changes and scale adjustments, the actual Fortran code is not listed here. The WFL code of the compiler-binder job (Program 4) must still be run before final execution of the model if any changes are made in the Fortran code. To run the zoom plot program, the same file name and parameter changes have to be edited into the WFL code of the run job (Program 5) as described above for the ocean-wide version.

A choice was made early in the model design to give a high priority to simulations of continuous drift tracks which could potentially cover ocean-wide space scales over annual to interannual time scales. One obvious consequence of this choice due to array size limitations was a sacrifice in resolution which could have been achieved by putting together several finer mesh submodels to make up the large grid. Avoided were the 'immense boundary problems of matching complex drift patterns back and forth across boundaries of several adjacent submodel size grids: and thus, we chose to rely on the graphics plot program to zoom in on results in small portions of the large grid. Although

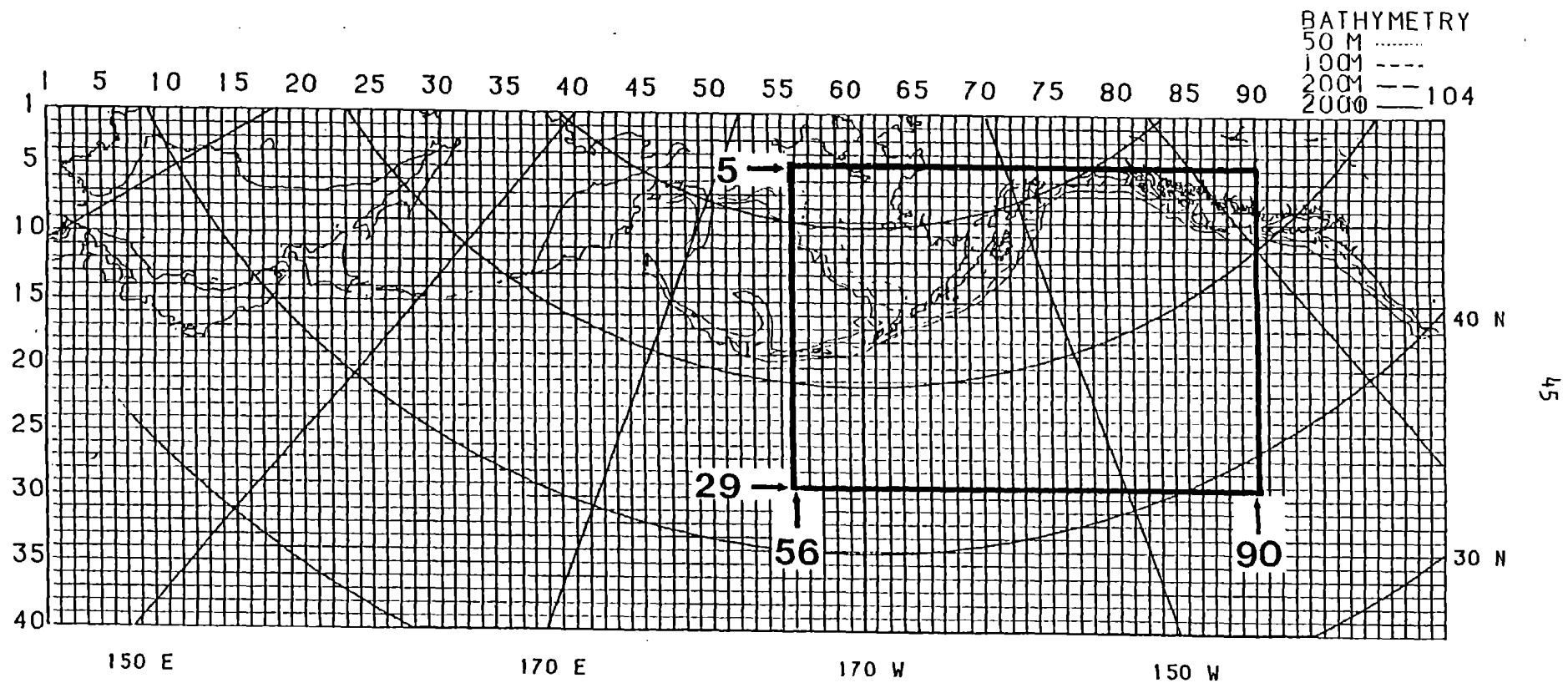


Figure 10.--Large area 40x104 grid and subset 25x35 grid for zoom graphics plots.

Program 4. --Listing of the Binder Job which binds and compiles the graphic outputs of the zoom model (File Title="ROB/M/PLOT/AUTO/ZOOM/WGULF/ARROW/BINDER").

```
1  ?BEGIN JOB MIYAHARA/BINDER/ROB/M/PLOT/AUTO/ZOOM/WGULF/ARROW;
2  QUEUE=20;
3  COMPILE OBJECT/ROB/M/PLOT/AUTO/ZOOM/WGULF/ARROW FORTRAN LIBRARY;
4  COMPILER FILE TAPE(KIND=DISK,TITLE=ROB/M/PLOT/AUTO/ZOOM/WGULF/ARROW
5  ?FORTRAN DATA
6  $SET MERGE
7  $RESET LIST
8  ?
9  BIND ROB/M/PLOT/AUTO/ZOOM/WGULF/ARROW/BOUND BINDER LIBRARY;
10 BINDER FILE HOST(TITLE=OBJECT/ROB/M/PLOT/AUTO/ZOOM/WGULF/ARROW);
11 ?BINDER EBCDIC
12 $RESET LIST
13 BIND=FROM GRIDS40104/=. ,(REFM0070)REFMOFF/= ON TYVO.
14      (REFM0070)PLOTCOMP87/= ON TYVO;
15 ?END JOB
```

Program 5. --Listing of Word Flow Language (WFL) Program which runs the graphic outputs of the zoom model (File Title="ROB/M/PLOT/AUTO/ZOOM/WGULF ARROW/WFL").

```

1  ?BEGIN JOB INGRAHAM/M/PLOT/PGRV/DDTEST/BOUND;
2  QUEUE=20;
3  RUN ROB/M/PLOT/AUTO/ZOOM/WGULF/ARROW/BOUND;
4  FILE FILE10=DIR96202;
5  FILE FILE28=P/3;
6  FILE FILE50=CURRENTS/40X104/DYNCURR/S2X2;
7  FILE FILE51=CURRENTS/40X104/DSLPL/DDTEST/FM7TOM12/D17START;
8  FILE FILE52=CURRENTS/40X104/WIND/DDTEST/FM7TOM12/D17START;
9  FILE FILE53=CURRENTS/40X104/SFCOCN/DDTEST/FM7TOM12/D17START;
10 FILE FILE54=CURRENTS/40X104/W10DD10/DDTEST/FM7TOM12/D17START/CSTROT
11 FILE FILE55=CURRENTS/40X104/W10DD10/DDTEST/FM7TOM12/D17START;
12 FILE FILE62=CURRENTS/40X104/PROGRESVECT/76/FM7TOM12/D17START/W075DD
13 FILE FILE61=CURRENTS/40X104/PROGRESVECT/77/FM7TOM12/D17START/W075DD
14 FILE FILE60=CURRENTS/40X104/PROGRESVECT/TEST/FM7TOM12/D17START/W10D
15 FILE FILE63=CURRENTS/40X104/PROGRESVECT/79/FM7TOM12/D17START/W075DD
16 FILE FILE64=CURRENTS/40X104/PROGRESVECT/78/FM7TOM12/D17START/W30DD
17 FILE FILE65=CURRENTS/40X104/PROGRESVECT/78/FM7TOM12/D17START/W20DD
18 FILE FILE80=CURRENTS/40X104/PROGRESVECT/80/FM7TOM12/D17START/W075DD
19 FILE FILE81=CURRENTS/40X104/PROGRESVECT/81/FM7TOM12/D17START/W075DD
20 FILE FILE82=CURRENTS/40X104/PROGRESVECT/82/FM7TOM12/D17START/W075DD
21 FILE FILE83=CURRENTS/40X104/PROGRESVECT/83/FM7TOM12/D17START/W075DD
22 ?DATA
23   1.0=====EVERY GRID POINT PLOT SLP WITH CURRENTS
24   5=====NUMBER OF DAYS TO PLOT
25   0=====PLOT SEA LEVEL PRESSURE CONTOURS ON PLOTS
26   0.0=====1. PLOT WIND VECTORS (1.0=PLOT)
27   0.0=====2. PLOT WIND CURRENT VECTORS
28   0.0=====3. PLOT THERMAL CURRENT VECTORS
29   0.0=====4. PLOT GEOSTROPHIC CURRENT VECTORS
30   0.0=====5. PLOT WIND+THERMAL+GEOSTROPHIC VECTORS
31   1.0=====6. PLOT 5 WITH COASTAL BENDING
32   0.0=====7. PLOT PROGRESSIVE VECTORS
33   0.0=====PLOT OF 5 OVER 6 TO SEE DIFFERENCE IN BENDING (2.0=PLOT)
34   1=====SET THE NUMBER TO PLOT OR NOPLT=1 TO START AT #1
35   0=====THE NUMBER OF PROGRESSIVE VECTOR STARTING POINTS
36   1.0=====THE WIND TUNING FACTOR
37   1.0=====THE DELTA-D TUNING FACTOR
38   ?
39 ?END JOB

```

other zooming is planned for future versions, the present scale is adequate to describe model outputs at this stage of development. Model results which were discussed above on the broad ocean-wide scale from plots of output values at every other grid point are now shown at every grid point in an enlarged portion of the Gulf of Alaska.

Geostrophic Currents

Geostrophic current vectors at zoom scale (Fig. 11(a)) show a relatively smooth, cyclonic flow around the Gulf of Alaska Gyre. But rather than being circular as the term "gyre" implies, this gyre has a closed asymmetrical form. This oblong shape is a result of the dynamic forces which cause the formation of the Alaskan Stream western boundary current where narrowing, southward flow, and friction along the land boundary of the Alaska Peninsula require intensification of flow and the high velocity axis of the Stream to be centered on the average just seaward of the edge of the continental shelf (water depth about 200 m). Therefore, the Alaskan Stream is essentially a permanent feature. The gyre is manifested by southward flow in the Alaskan Stream, by offshore flow where the stream weakens to the west, by eastward flow of the Subarctic Current to the south, and by northward then westward flow at the head of the gulf. A long, narrow, relatively stagnant zone appears in the central part of the gyre.

The arrow lengths in Figure 11(a) indicate that velocities calculated from data with this large model grid spacing are

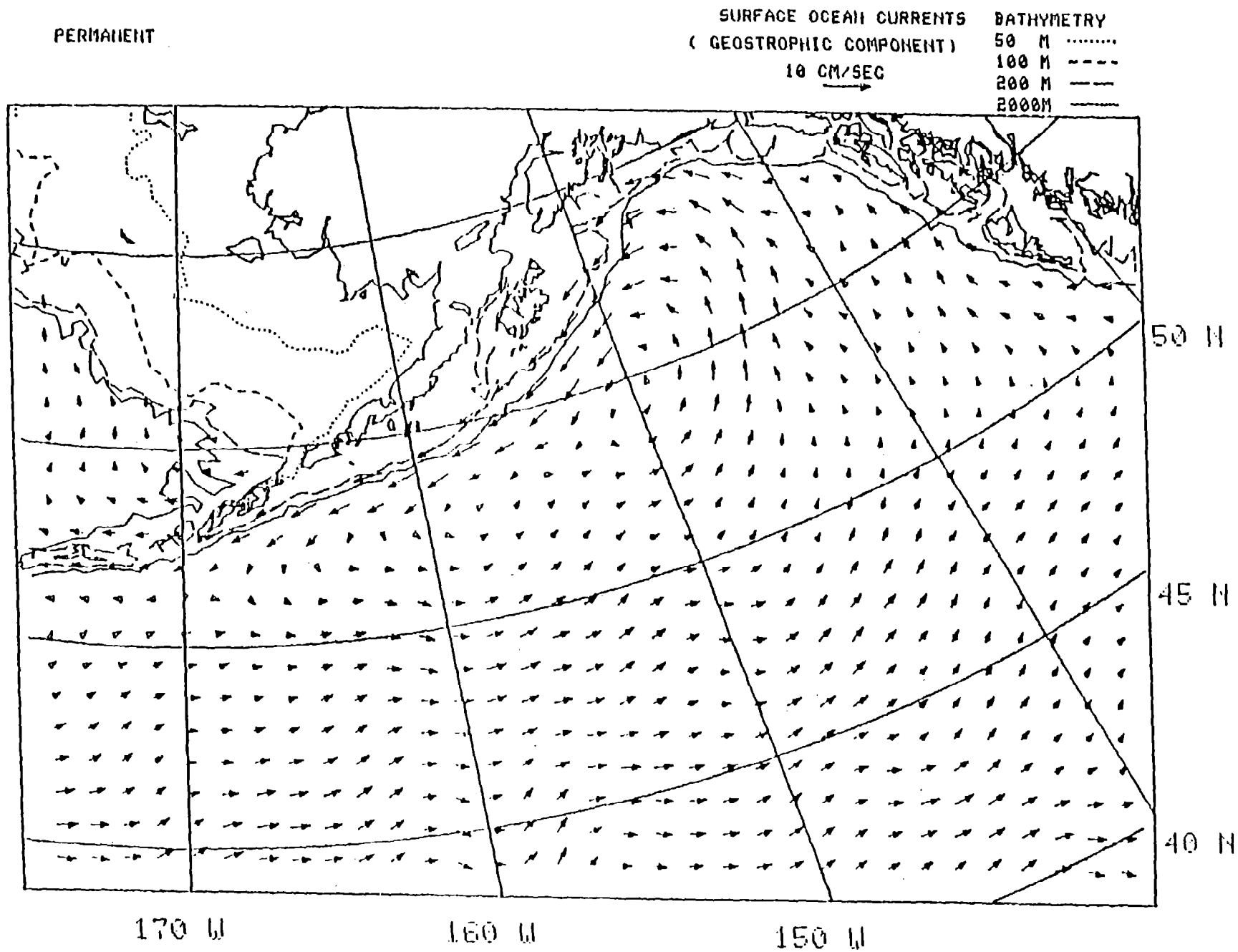


Figure 11(a). --Surface geostrophic current vector field on 25x35 grid (0/2000 db).
Permanent flow field calculated without modifiers in the Alaskan Stream.

slower than expected in the Alaskan Stream, generally less than 10 cm/sec. Closely spaced (5-15 km) conductivity, temperature and depth (CTD) station data have shown that the high velocity axis of the stream is narrow, about 15 to 40 km, much narrower than the 90 km grid of the model, and computed geostrophic speeds, may be as much as 60 cm/sec (Ingraham and Favorite 1968). To increase the reality of this simulation, velocity values calculated at certain grid points near the axis of the Alaskan Stream have been modified to reflect these higher known speeds. Modifiers used at specified grid points are shown in Table 4. Values of U and V components at these specified grid points are multiplied by 1.0 plus the corresponding modifier value. Figure 11(b) shows the geostrophic current vector field after modification for comparison with the initially calculated vectors in Figure 11(a). These modifications have greatly enhanced the simulated movement of drifters within the Alaskan Stream and will provide another mechanism for calibration and fine tuning of the model to actually measured drifter tracks from reported ocean experiments.

Wind Currents

Zoom plots of wind with sea level pressure contours superimposed (Fig. 12), currents due to wind with sea level pressure contours superimposed (Fig. 13), and total currents (Fig. 14a) for 17 July 1978, are shown to provide expanded details of the Gulf of Alaska portion of the ocean-wide plots which were discussed earlier in Figures 5, 6, and 7,

Table 4. --Modifier values for the calculation of the permanent geostrophic current field near the axis of the Alaskan Stream at specified (I,J) grid points. Model uses (1.0+modifier) times the value at the grid point to reflect higher known speeds.

I	J	Modifier
10	74	0.60
11	74	0.20
11	73	0.75
12	73	0.35
12	74	-0.15
13	73	-0.20
13	72	0.45
14	72	-0.20
14	71	0.45
15	70	0.30
15	69	0.50
16	69	-0.10
16	68	0.45
16	67	1.00
17	66	-0.10
17	65	0.95
18	64	-0.10
18	63	0.80
19	62	0.15
19	61	0.30
19	60	0.45
19	59	0.35

PERMANENT

SURFACE OCEAN CURRENTS
(GEOSTROPHIC COMPONENT)

10 CM/SEC
→

BATHYMETRY
50 M
100 M ----
200 M ---
2000M ———

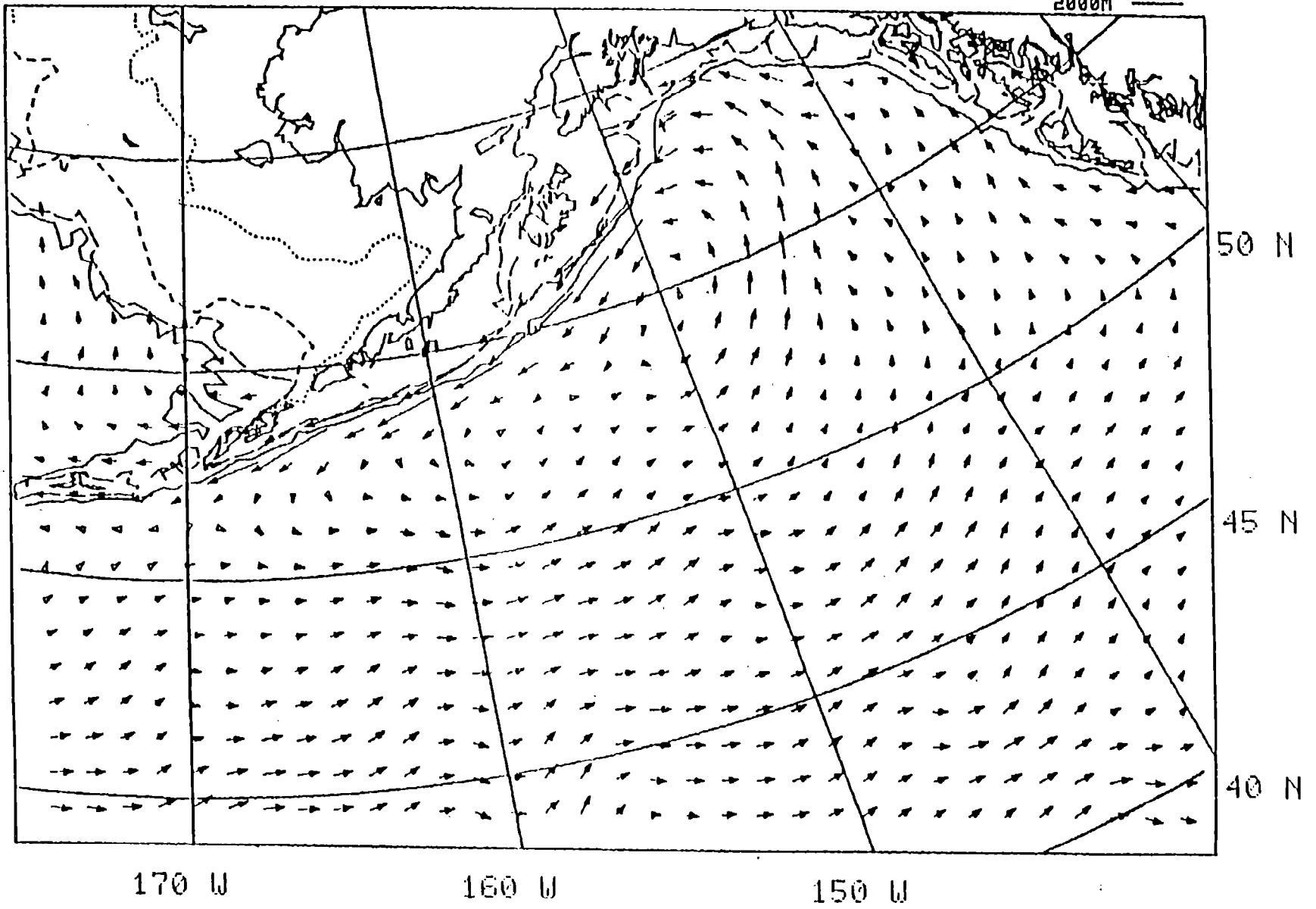


Figure 11(b). --Surface geostrophic current vector field on 25x35 grid (0/2000 db).
Permanent flow field with double modifiers in the Alaskan Stream.

SEA LEVEL PRESSURE (MB) 78 - 7 - 17

INTERVAL=8
NUMBERED=16

SURFACE WIND VECTORS

10 M/SEC

BATHYMETRY

50 M
100 M - - - -
200 M - - - -
2000M - - - -

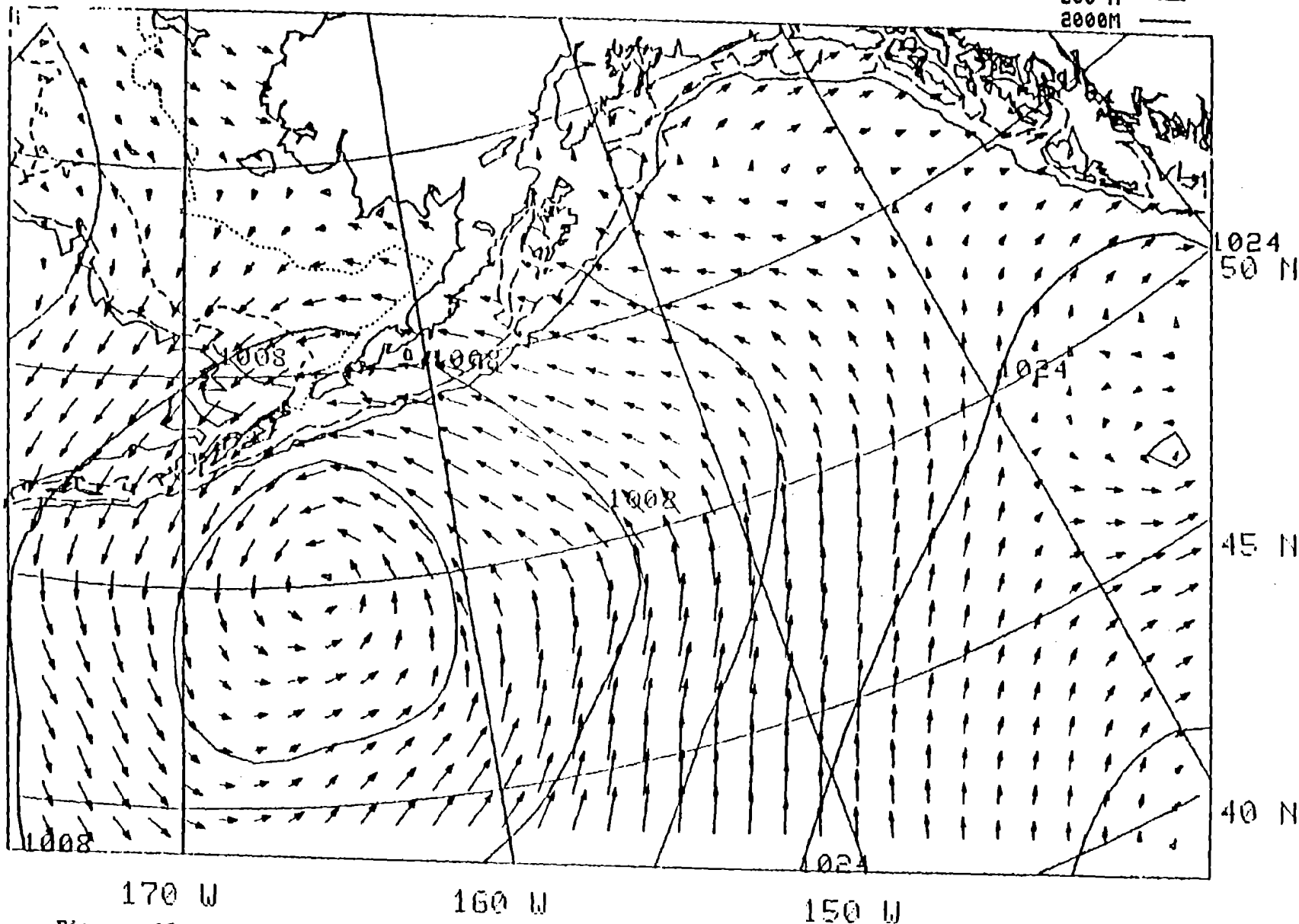


Figure 12.--Wind vector field (m/sec) calculated from sea level pressure fields for 17 July 1978. Zoom plot of Figure 5 showing cross-isobar flow into the low pressure centers.

SEA LEVEL PRESSURE (MB) 78 - 7 - 17

INTERVAL-8
NUMBERED-16

SURFACE OCEAN CURRENTS
(WIND COMPONENT)
10 CM/SEC

BATHYMETRY
50 M
100 M ----
200 M - - -
2000M _____

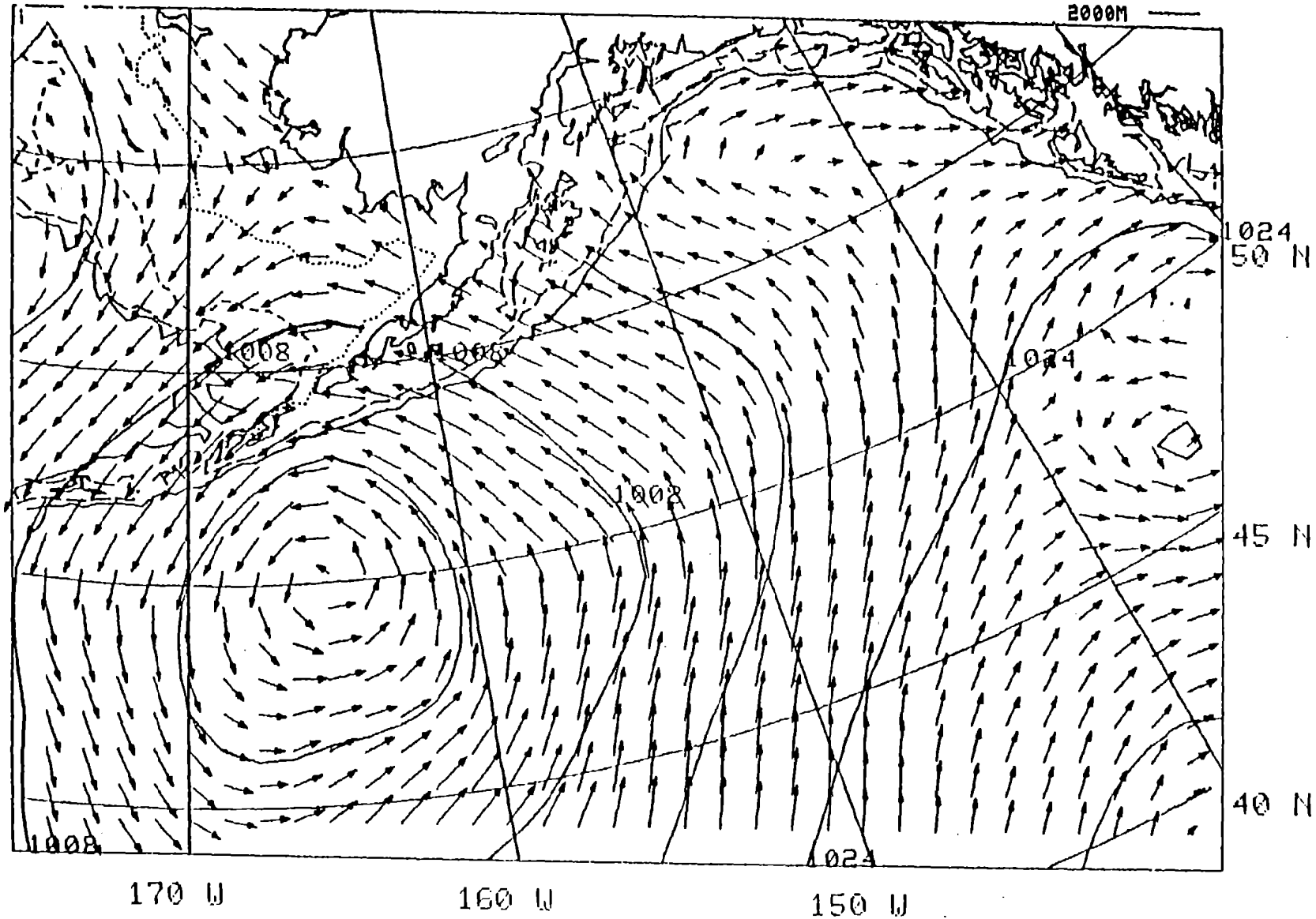


Figure 13.--Surface ocean current vector field (cm/sec) calculated from wind vector field in Figure 5 for 17 July 1978. Zoom plot of Figure 6 showing ocean currents nearly paralleling sea level pressure isobars.

78 - 7 - 17

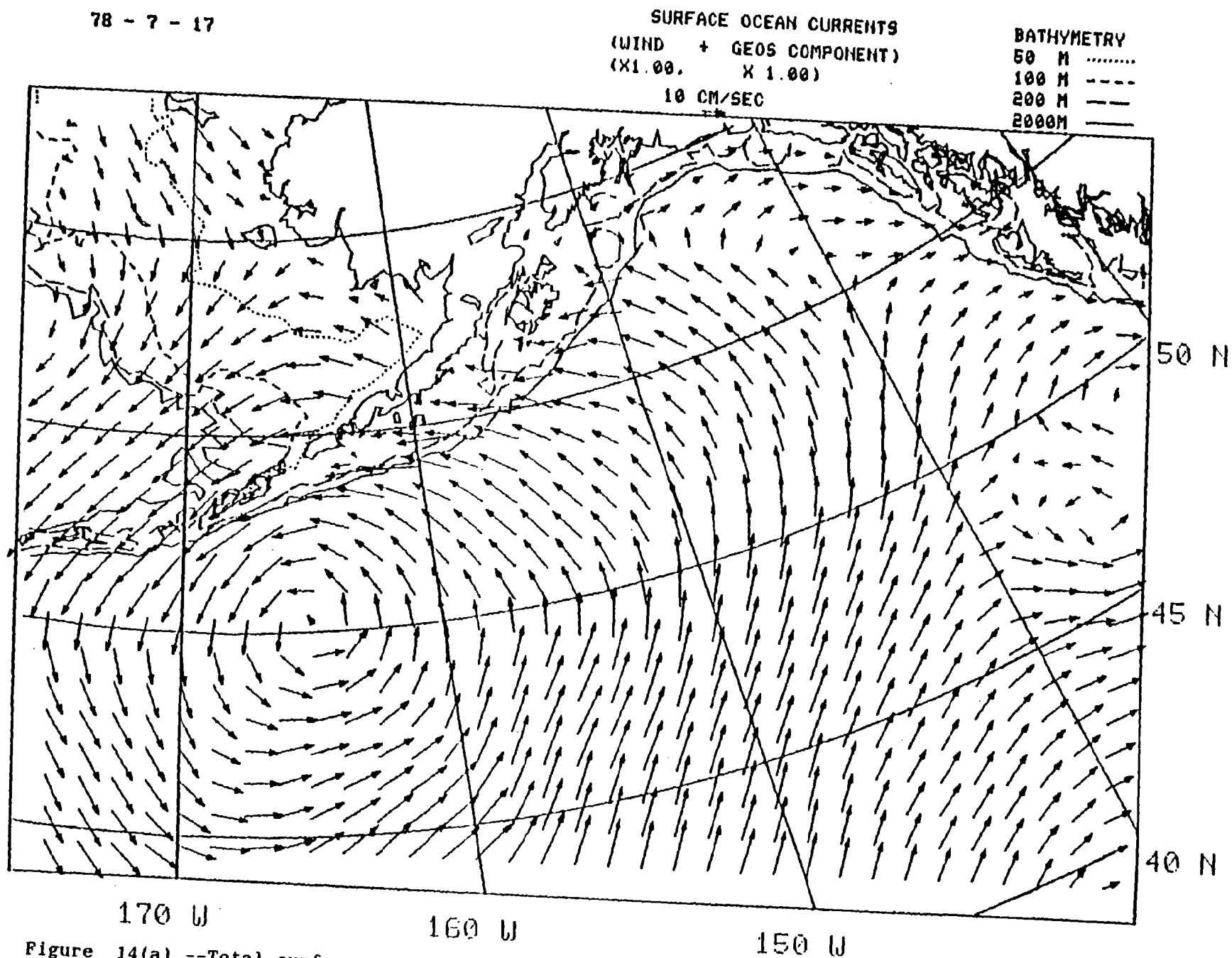


Figure 14(a).--Total surface ocean current vector field for 17 July 1978. Zoom plot of Figure 7.

respectively. Major output features of the model are more clearly seen here. Wind vectors (Fig. 12) show strong cross-isobar flow spiraling into the low pressure center at $50^{\circ}\text{N}-165^{\circ}\text{W}$ and out of the high pressure center at $40^{\circ}\text{N}-142^{\circ}\text{W}$. Ocean surface current vectors (Fig. 13) which are computed at an angle to the right of the wind direction at each point in the field are now nearly parallel to the sea level pressure isobars. Note that some cross-isobar flow still remains in some areas such as near $47^{\circ}\text{N}-153^{\circ}\text{W}$ where winds are stronger than about 15 m/sec, making the computed angle of deflection of the currents to the right of the wind smaller. As a general character trait, currents due to wind are rarely less than 5 cm/sec or greater than 20 cm/sec, but most speeds do fall between 10 and 20 cm/sec over wide areas. Thus, wind current speeds definitely dominate over the geostrophic speeds in the overall features of the total surface currents (Fig. 14(a)).

Total Currents

A final touch is added to this total surface current vector field at each grid point which is within two grid lengths of land to account for simulated boundary effects on onshore flow (see above discussion). Here, Figure 14(b) shows a one-example comparison of the two current vectors: 1) the initial vector before bending and reduction and 2) the final resultant current vector. Along the west coast of Canada these near shore vectors are deflected to the right in this example because their angle of incidence was to the right of the line normal to shore. Along the

78 - 7 - 17

SURFACE OCEAN CURRENTS
(WIND + THERM + GEOS COMPONENT)
(X1.00, X 1.00)
10 CM/SEC

BATHYMETRY
50 M
100 M ----
200 M ---
2000M ———

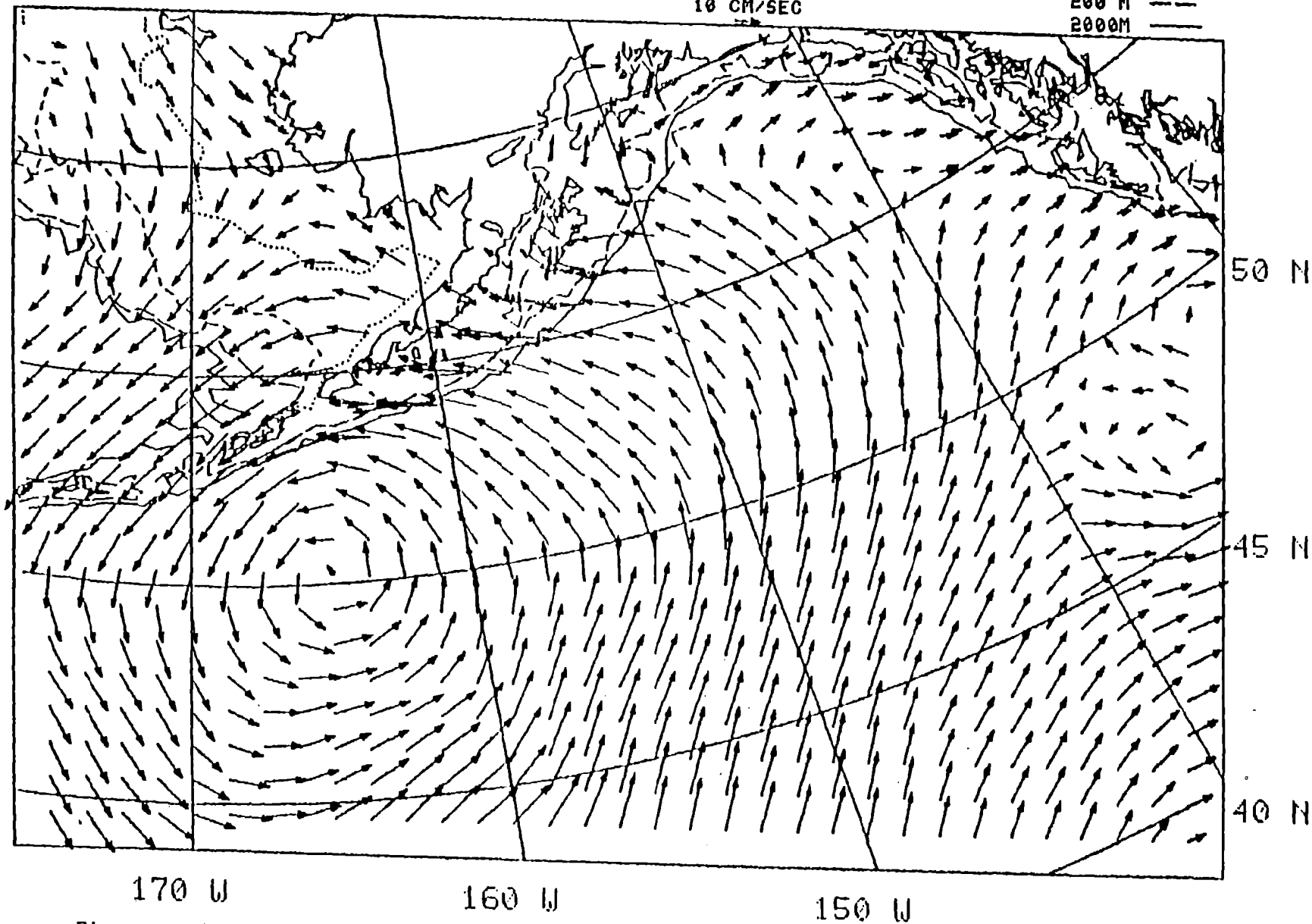


Figure 14(b).--Total surface ocean current vector field plus a second arrow near shore showing conditions before and after the costal bending routine was entered for 17 July 1978.

Alaska Peninsula the pattern shows the opposite deflection to the left. Between these two areas along the continental shelf near 150°W the flow tends to diverge at least for this one day in history. The lengths of the final total current vectors have also been diminished inversely proportional to their distance from shore. Vectors are forced mathematically to zero at shore, and no bending effects are computed for winds that are directed offshore. This completes the computation of the total current vector field for the first day of the model run.

As an example of the daily variability the next 6 days of total currents are shown in Figures 15(a-f) which are zoom plots of the six parts of Figure 8. In these six fields the overall flow pattern looks reasonably persistent with a strong 15 to 25 cm/sec current flowing northeastward towards the head of the Gulf of Alaska where this onshore flow diverges to the west and southeast. This coastal divergence area appears to shift westward from about 130°W to about 150°W and then shift back to 130°W over this 6-day period without substantial changes in the locations of the cyclonic gyre south of the Aleutian Islands at about 170°W and the anticyclonic gyre off the west coast of the United States at about 45°N - 45°W . The wind current component of flow appears to continue to dominate portions of even the strongest geostrophic areas like the Alaskan Stream, either enhancing or opposing the geostrophic flow at least on a short-term basis (1-3 days). These examples show that total currents of 10-20 cm/sec still appear to be consistently active over large areas of these fields even during summer. Velocities probably would have been even

78 - 7 - 18

SURFACE OCEAN CURRENTS
(WIND + GEOS COMPONENT)
(X1.00, X 1.00)
10 CM/SEC

BATHYMETRY
50 M
100 M ----
200 M ---
2000M ———

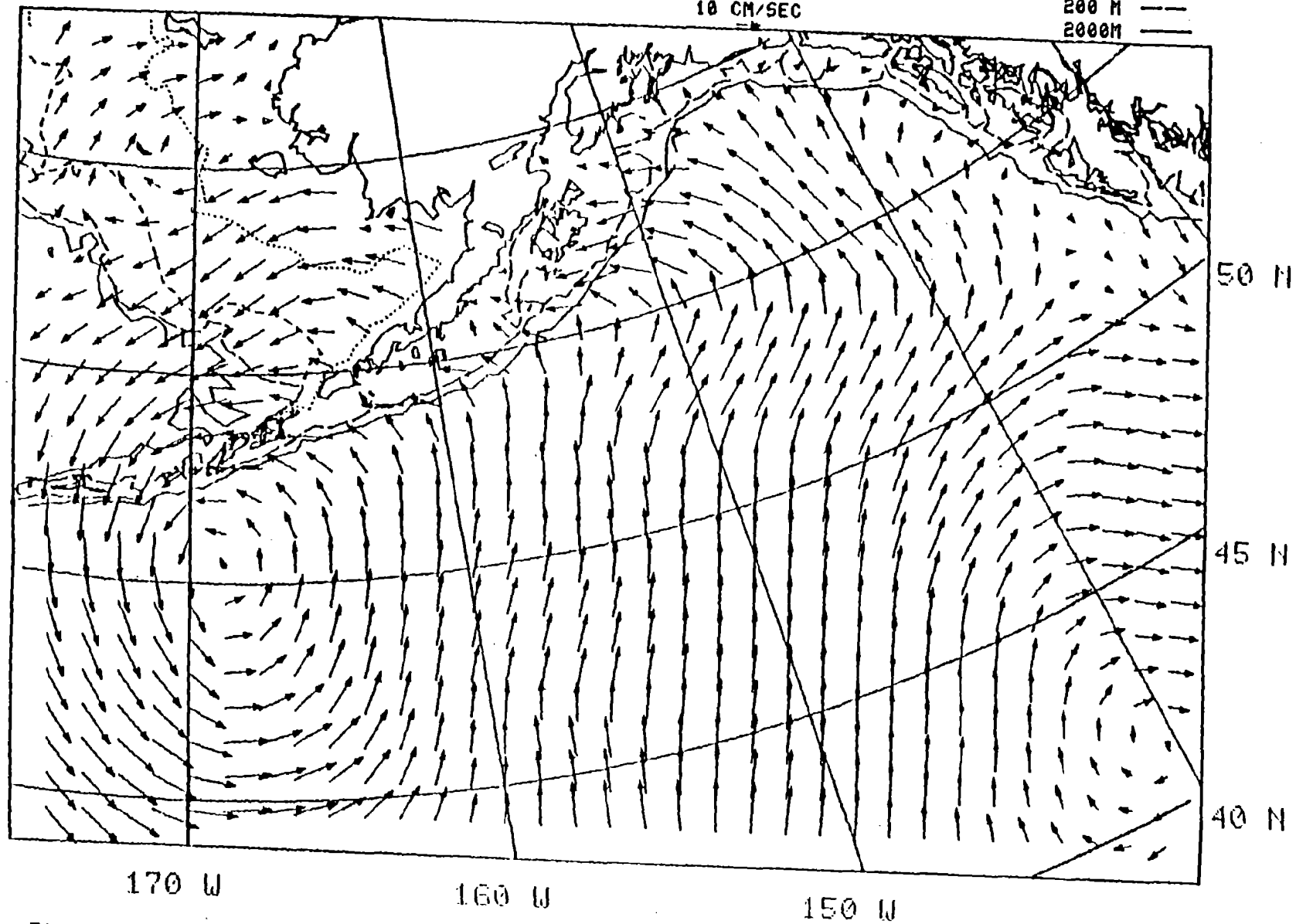


Figure 15(a).--Total surface ocean current vector field for 18 July 1978. Zoom plot of a portion of Figure 8.

78 - 7 - 19

SURFACE OCEAN CURRENTS
(WIND + GEOS COMPONENT)
(X1.00, X 1.00)
10 CM/SEC

BATHYMETRY
50 M
100 M ----
200 M ---
2000M —

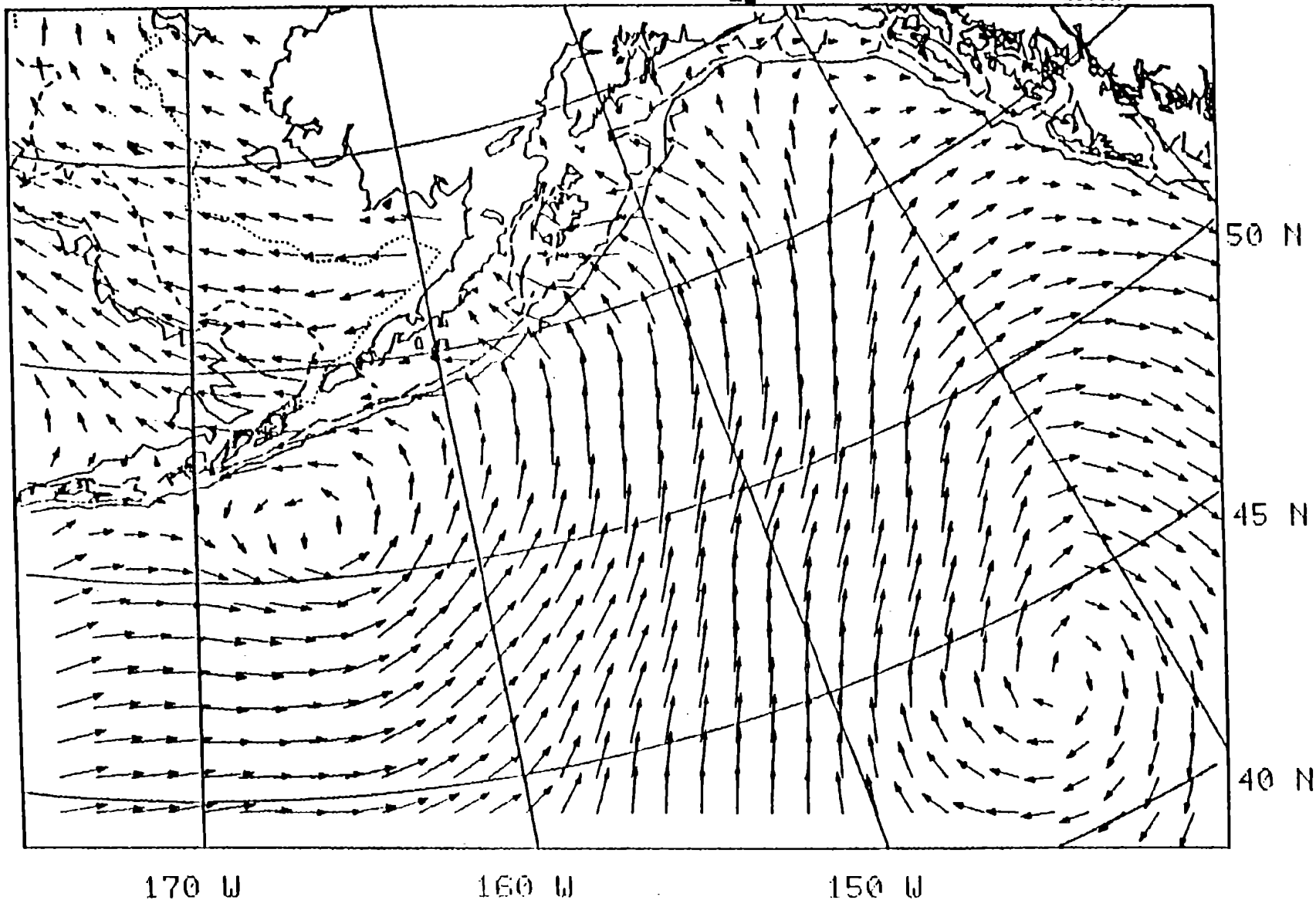


Figure 15(b).--Total surface ocean current vector field for 19 July 1978. Zoom plot of a portion of Figure 8.

78 - 7 - 80

SURFACE OCEAN CURRENTS
(WIND + GEOS COMPONENT)
(X1.00, X 1.00)
10 CM/SEC

BATHYMETRY
50 M
100 M ----
200 M ---
2000M —

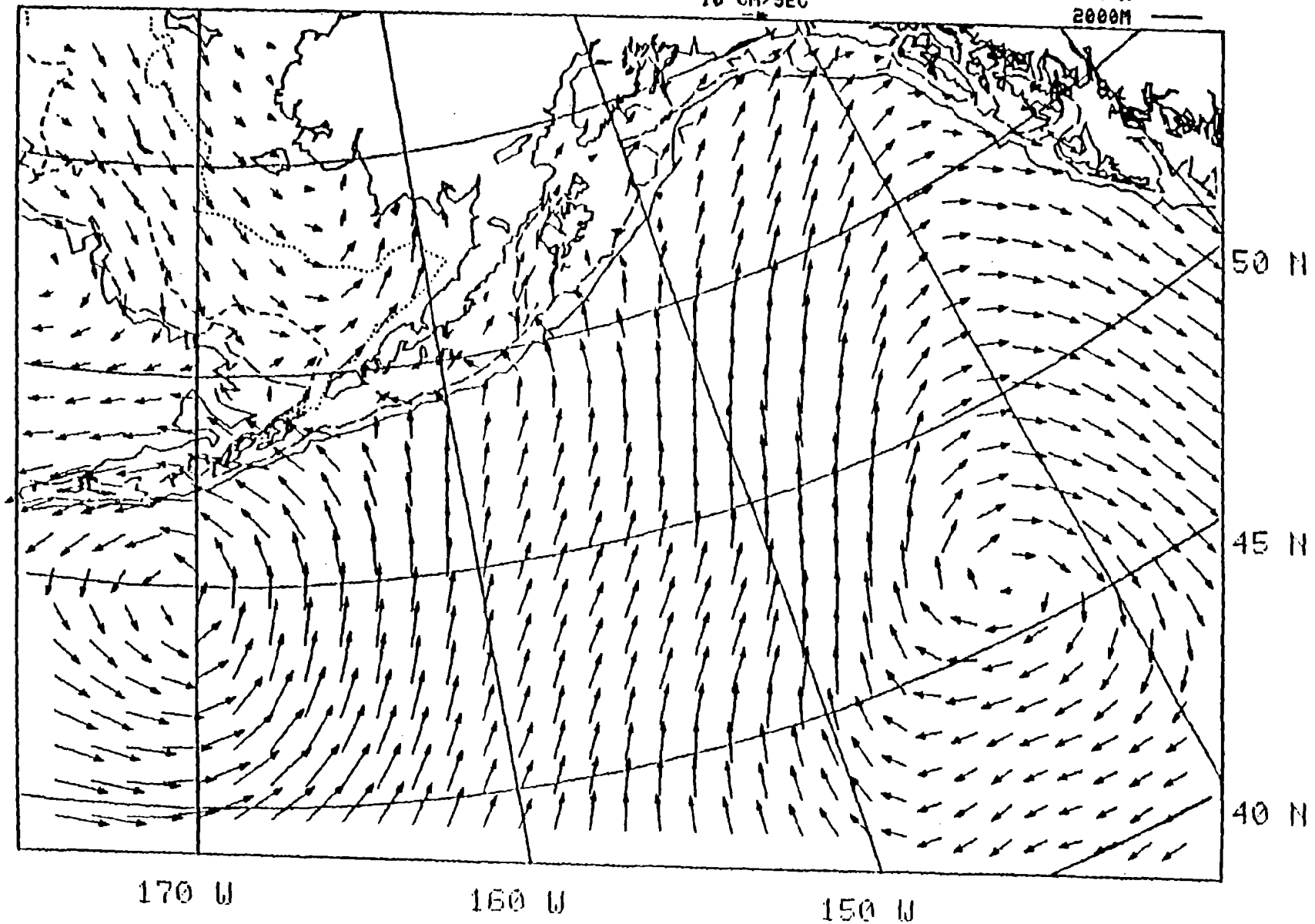
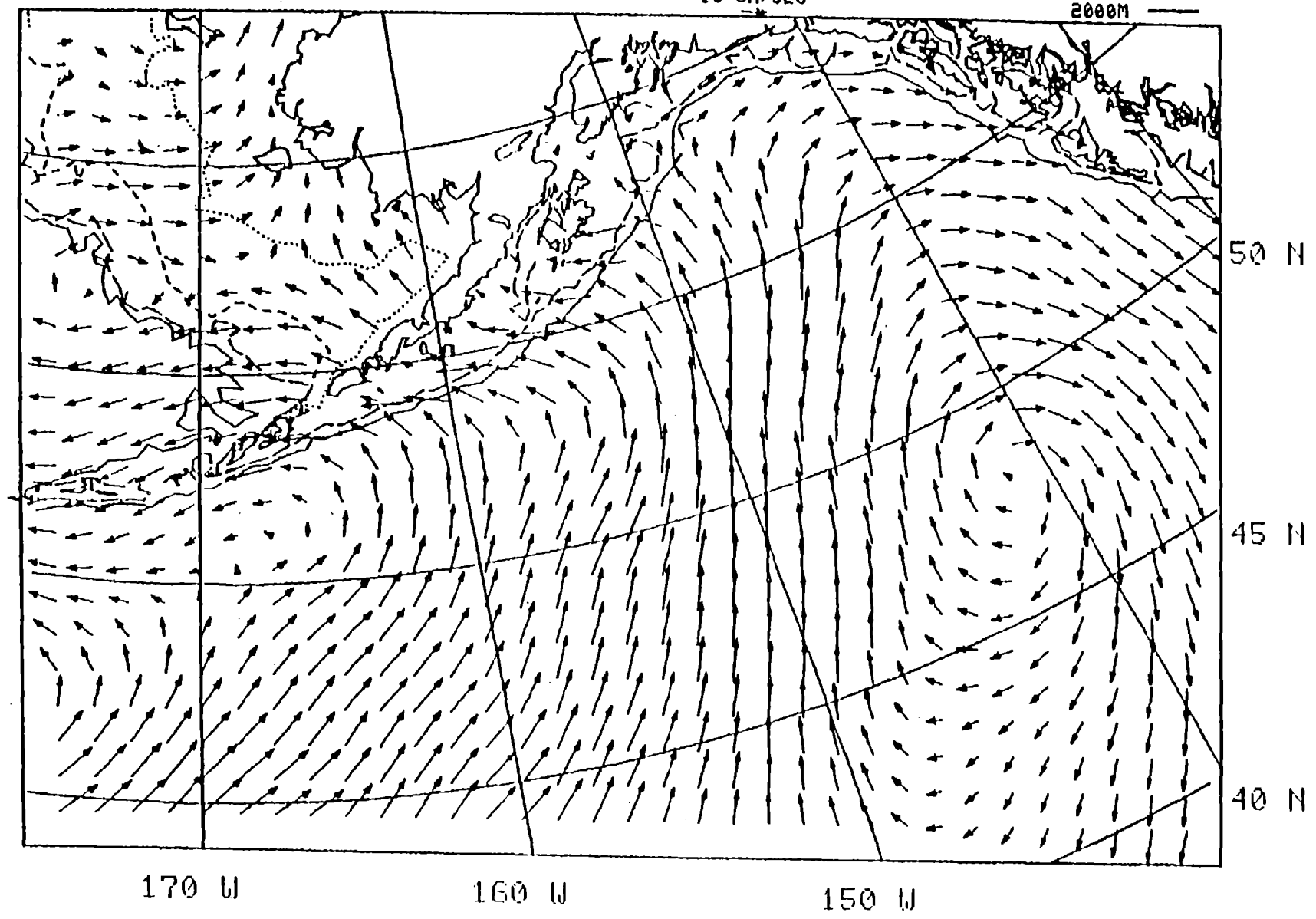


Figure 15(c).--Total surface current vector field for 20 July 1978. Zoom plot of a portion of Figure 8.

78 - 7 - 81

SURFACE OCEAN CURRENTS
(UIND + GEOS COMPONENT)
(X1.00, X 1.00)
10 CM/SEC

BATHYMETRY
50 M
100 M ----
200 M ---
2000M ———



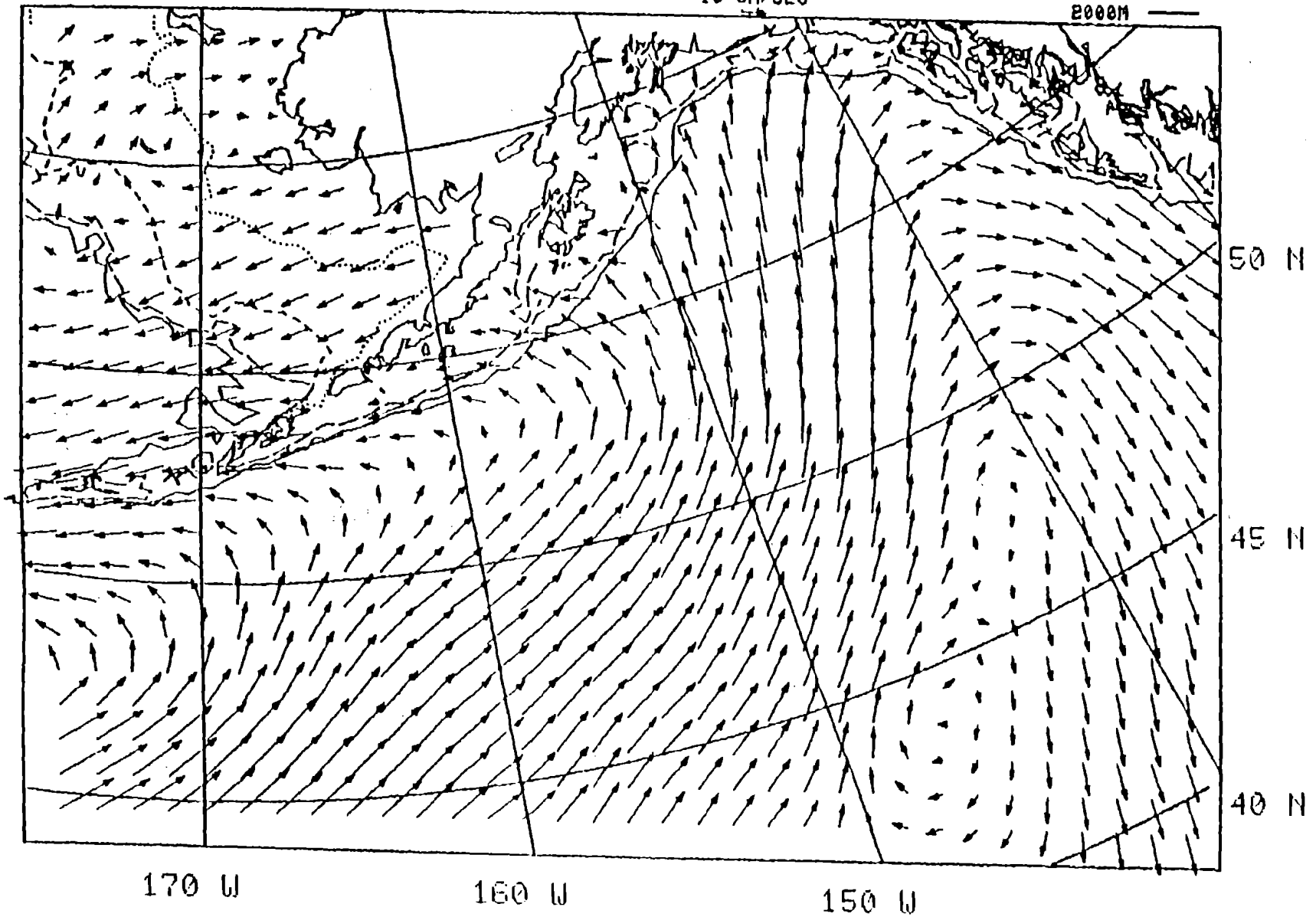
62

Figure 15(d).--Total surface current vector field for 21 July 1978. Zoom plot of a portion of Figure 8.

78 - 7 - 82

SURFACE OCEAN CURRENTS
(WIND + GEOS COMPONENT)
(X1.00, X 1.00)
10 CM/SEC

BATHYMETRY
50 M
100 M ----
200 M ---
2000M —



63

Figure 15(e).--Total surface current vector field for 22 July 1978. Zoom plot of a portion of Figure 8.

78 - 7 - 23

SURFACE OCEAN CURRENTS
(WIND + GEOS COMPONENT)
(X1.00, X 1.00)
10 CM/SEC

BATHYMETRY
50 M
100 M ----
200 M ---
2000M —

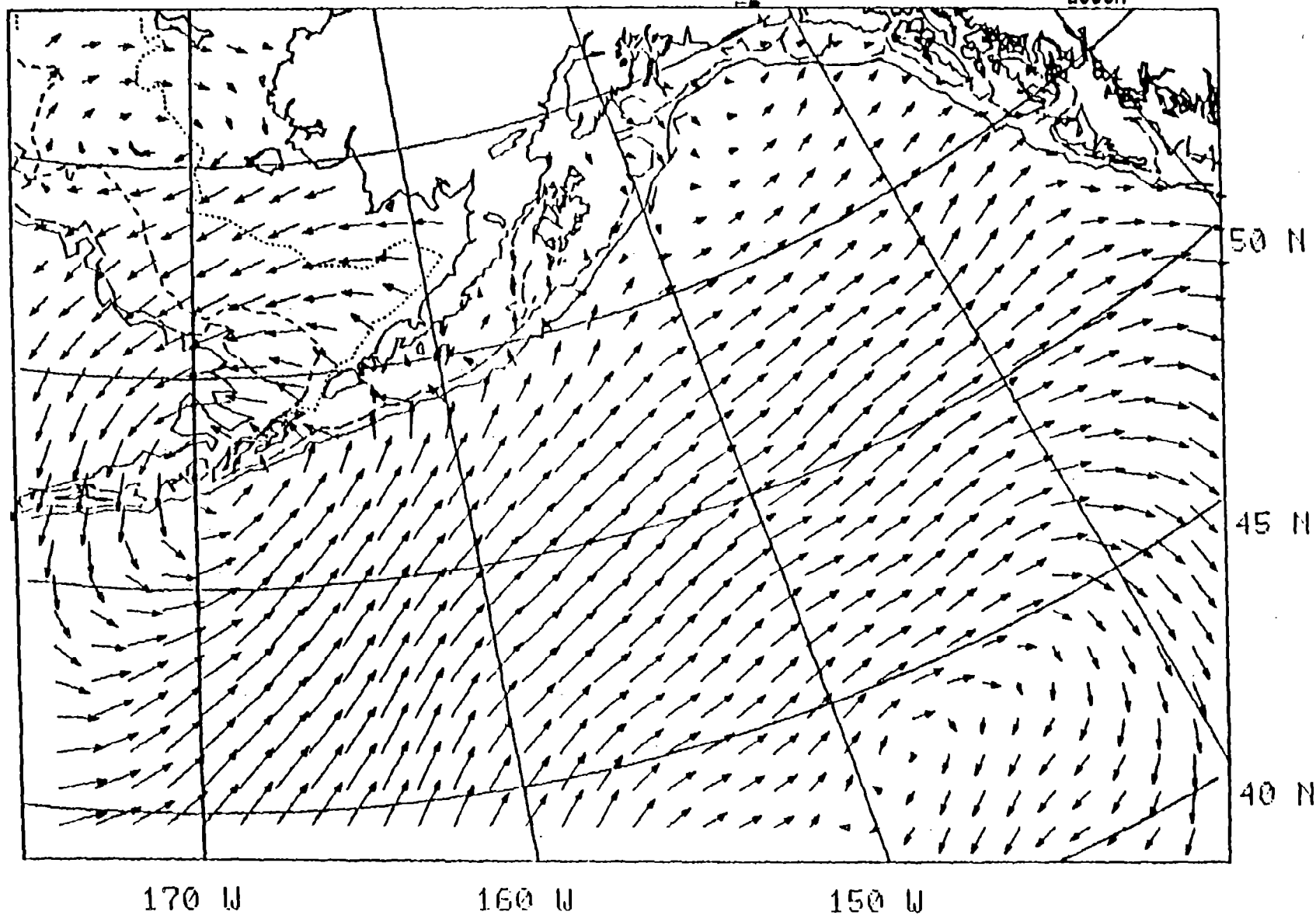


Figure 15(f). --Total surface current vector field for 23 July 1978. Zoom plot of a portion of Figure 8.

stronger if a winter series of days had been selected in this example.

Progressive Vector Tracks

Now that we have the complete total current vector field, recall that the last step in the model run loop was to compute a surface transport vector at a number of selected starting points to get the actual drift distance per day using velocities that were interpolated from this field at these points for the 24-hour average. Starting points for the next day are updated to be the end points of the previous day's transport vectors. These transport vectors have been saved on disk in a series of daily start and end point locations in grid units so that all the plot program has to do now is connect these start and end points day by day and do the map graphics to create the progressive vector tracks.

Zoom plots of progressive transport vectors are of particular interest because they summarize the model results over time from some desired historical starting date to a number of days hence. They also provide a mechanism for comparisons of model results with ocean drifter experiments to be used in calibration and tuning of the model. In Figure 16(a), a zoom plot of Figure 10(a), drifter tracks are simulated for about five and one-half months beginning in mid-July 1978, at nine starting locations which were selected to show the overall flow pattern. Indeed, near-surface ocean water drifted around the Gulf of Alaska Gyre as expected with all the tracks moving toward the east to

DAILY PROGRESSIVE VECTOR DISTANCE

FROM 78 - 7 - 17
 TO 78 - 12 - 31
 5 10 15 20 25 30

SURFACE OCEAN CURRENTS
 (WIND + THERM + GEOS COMPONENT)
 (X1.00, X 0.00, X 1.00)

BATHYMETRY
 50 M
 100 M ----
 200 M ---
 2000M ——— 35

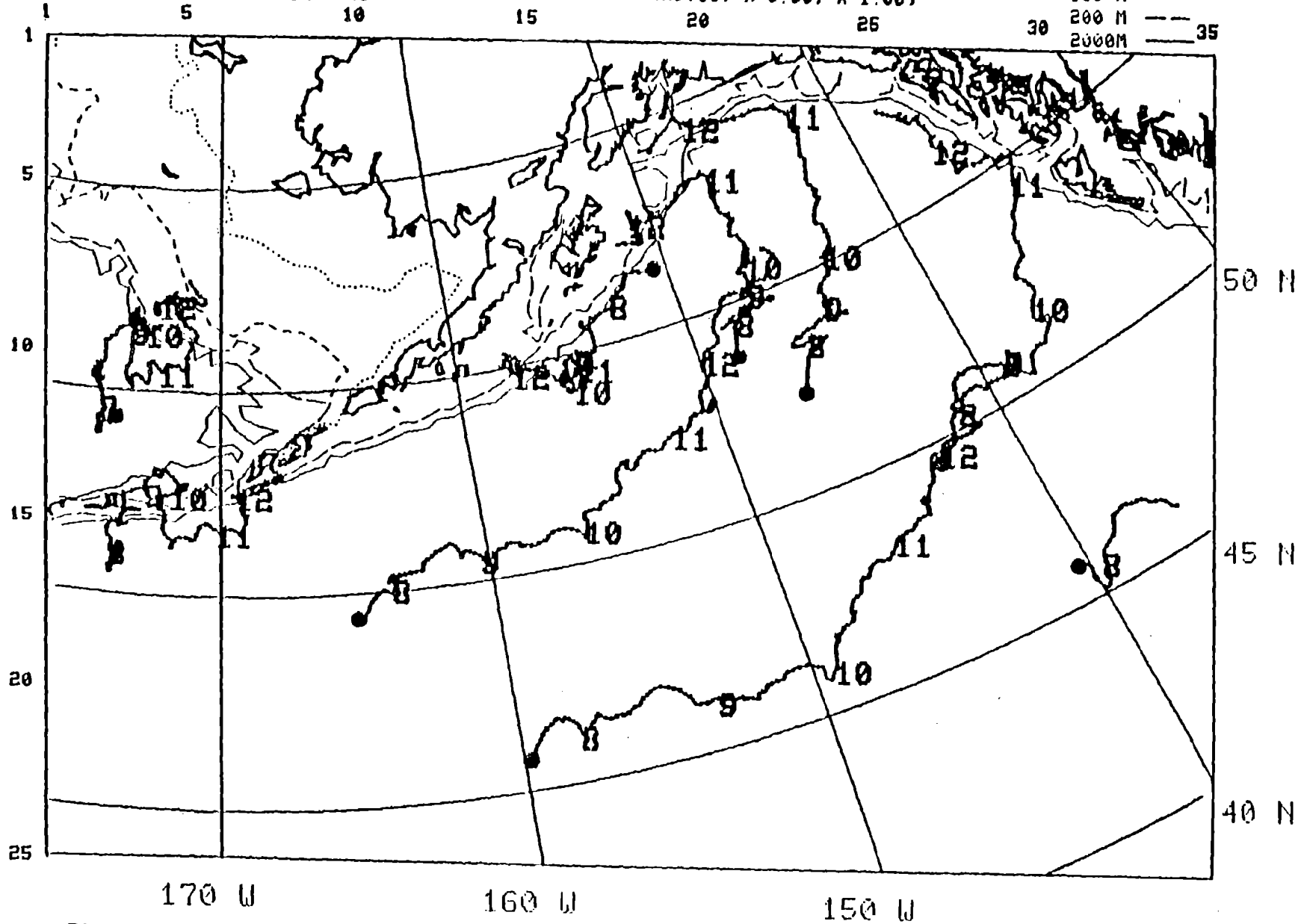


Figure 16(a).--Progressive daily transport vectors--Total surface ocean transport (17 July to 31 December 1978). Zoom plot of Figure 9(a).

northeast except those that enter the Alaskan Stream west of the top of the gulf. Speeds of drift, however, were slower than expected. The starting location of each track is marked by a large dot. Small connected arrows each represent one day of estimated distance drifted (transported), and the large numbers spaced along the tracks indicate the new month at the first day of the new month as it was encountered along the track. One feature evident in all the tracks is the acceleration in flow from the slow start in month 8 to a maximum in month 10; then relatively strong flow continued through month 12. Tracks in the southern branch of the gyre appear steadier than the others with a few small short-term loops, small meanders, and very few reversals. The longest track which started at 46°N - 160°W had an average drift of 11 km/day or a speed of about 10 cm/sec. Other tracks in the Alaskan Stream and Bering Sea are shorter and display more looping and random motions. The average drift of 3.5 km/day (3 cm/sec) for the track which started at 57°N - 150°W was surprisingly slow for the Alaskan Stream water. The other surprising feature was the absence of the southeastward recirculation of the gyre which has been shown to exist in the vicinity of 160°W (Reed 1980). These figures reflect the final model results at this stage of development. Some insight can be gained about the whys of the unusual behaviors by looking at tracks of individual flow components. In other words, what would drift tracks look like if the flow was purely geostrophic or purely local wind driven.

Pure geostrophic drift (WFAC=0.0) is shown in Figure 16(b), a zoom plot of Figure 10(b). This being a constant field, this flow has no time dependence, that is, tracks started at the same location may be started on any date and drift will follow the same track. Speeds around the gyre varied from 1 to 3 cm/sec in the southern part to about double, 6 cm/sec, in the Alaskan Stream.

Pure wind currents (DDFAC=0.0) are shown in Figure 16(c), a zoom plot of Figure 10(c). In contrast the flow everywhere is toward the northeast, even in the stream. These tracks give a better quantitative estimate of the dominance of wind-induced flow over geostrophic flow. The strongest wind drift track which started at 46°-160°W had an average speed of about 7 cm/sec supporting a geostrophic flow of 2-4 cm/sec in the same direction thus adding to the total flow of about 10 cm/sec. In the Alaskan Stream the wind-only drifter track (57°N-150°W) was the smallest in the whole area, only 1-3 cm/sec. Despite the small speed, this consistently opposing wind current effectively subtracted from the larger geostrophic component of 6 cm/sec to the slower drift of 3-4 cm/sec seen in Figure 16(a).

In all of the above discussion, only the pure untuned (DDFAC=1.0 and WFAC=1.0) basic model equations were used. These factors must be tuned as closely as possible to conditions in nature before indices reflecting changing conditions in nature can be computed by the model. A more complete quantitative discussion of statistical averages of model outputs (net drift in km/month, monthly displacement direction in

DAILY PROGRESSIVE VECTOR DISTANCE

FROM 78 - 7 - '17
 TO 78 - 12 - 31
 5 10 15

SURFACE OCEAN CURRENTS

(WIND + THERM + GEOS COMPONENT)
 (X 0.00, X 0.00, X 1.00)
 20 25 30

BATHYMETRY

50 M
 100 M - - - -
 200 M - - - -
 2000M - - - - 35

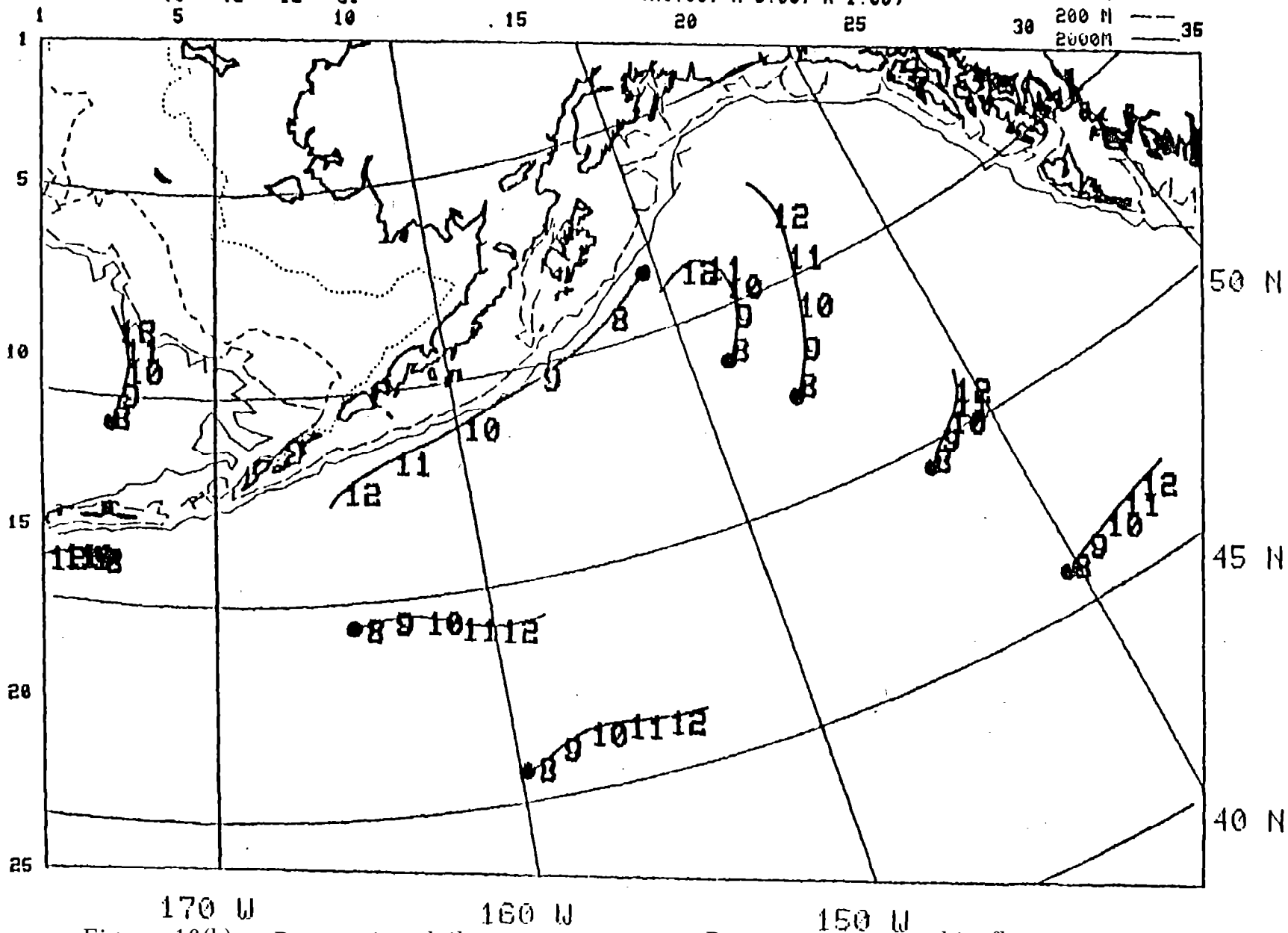


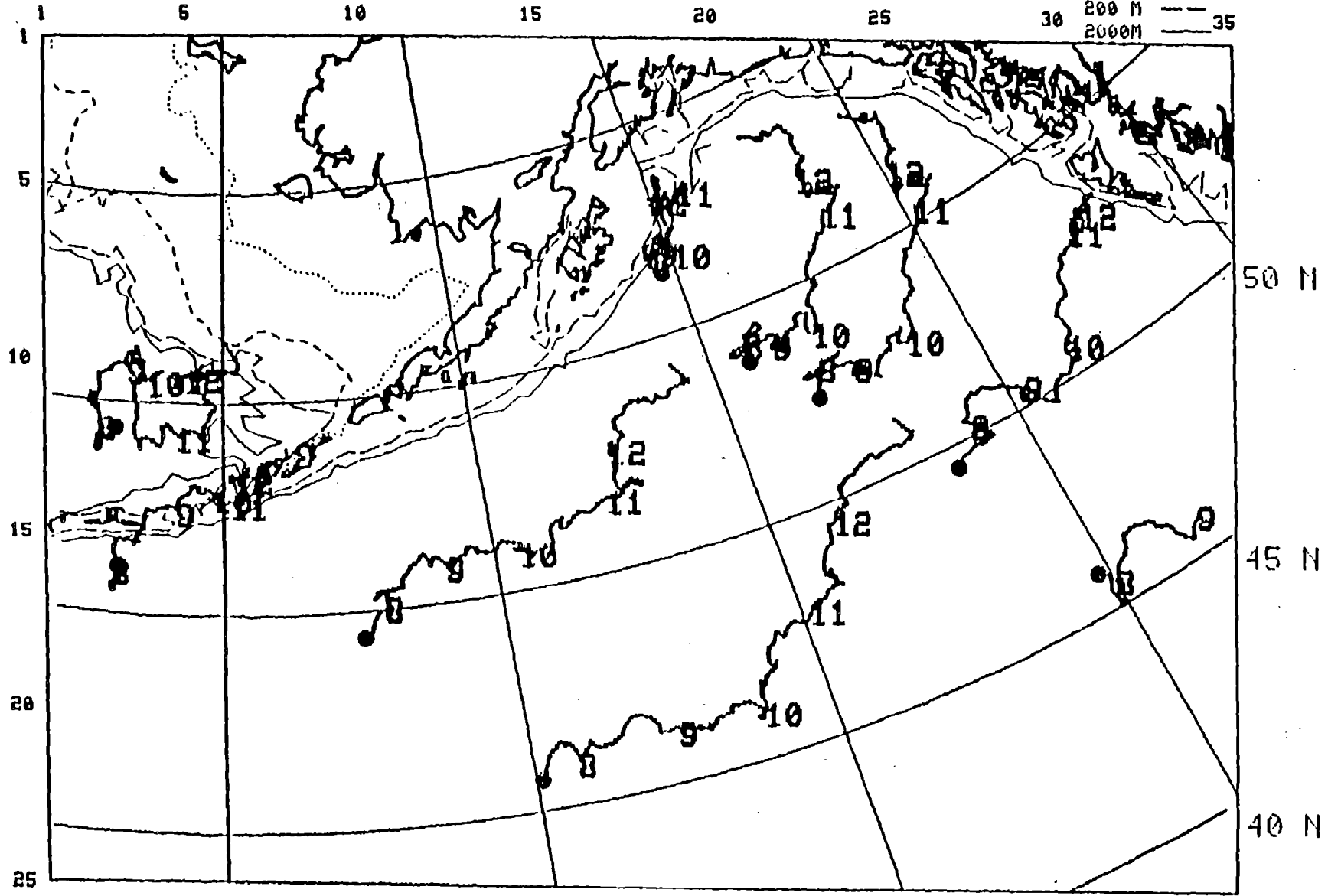
Figure 16(b). --Progressive daily transport vectors-Permanent geostrophic flow only (Wfac=0.0) (17 July to 31 December 1978). Zoom plot of Figure 9(b).

DAILY PROGRESSIVE VECTOR DISTANCE

FROM 78 - 7 - 17
 TO 78 - 12 - 31
 5 10 15 20 25 30

SURFACE OCEAN CURRENTS
 (WIND + THERM + GEOS COMPONENT)
 (X1.00, X 0.00, X 0.00)

BATHYMETRY
 50 M
 100 M ----
 200 M ---
 2000M _____ 35



170 W 160 W 150 W
 Figure 16(c). --Progressive daily transport vectors-Surface current due to wind only (DDFAC=0.0) (17 July to 31 December 1978). Zoom plot of Figure 9(c).

degrees true, etc.) will be reserved for a later date when tuning and calibration are completed. Refer to the next report in this series of model developments for a full discussion of tuning results and comparisons with actual ocean drifter track data. As a preview, the untuned model was run for one month starting at each drifter location on the first of each month, that is, three locations for months 8, 9, 10, 11, and 12 (Fig. 17). Note the lack of agreement in the Alaskan Stream area and the fair agreement in the rest of the Gulf of Alaska Gyre.

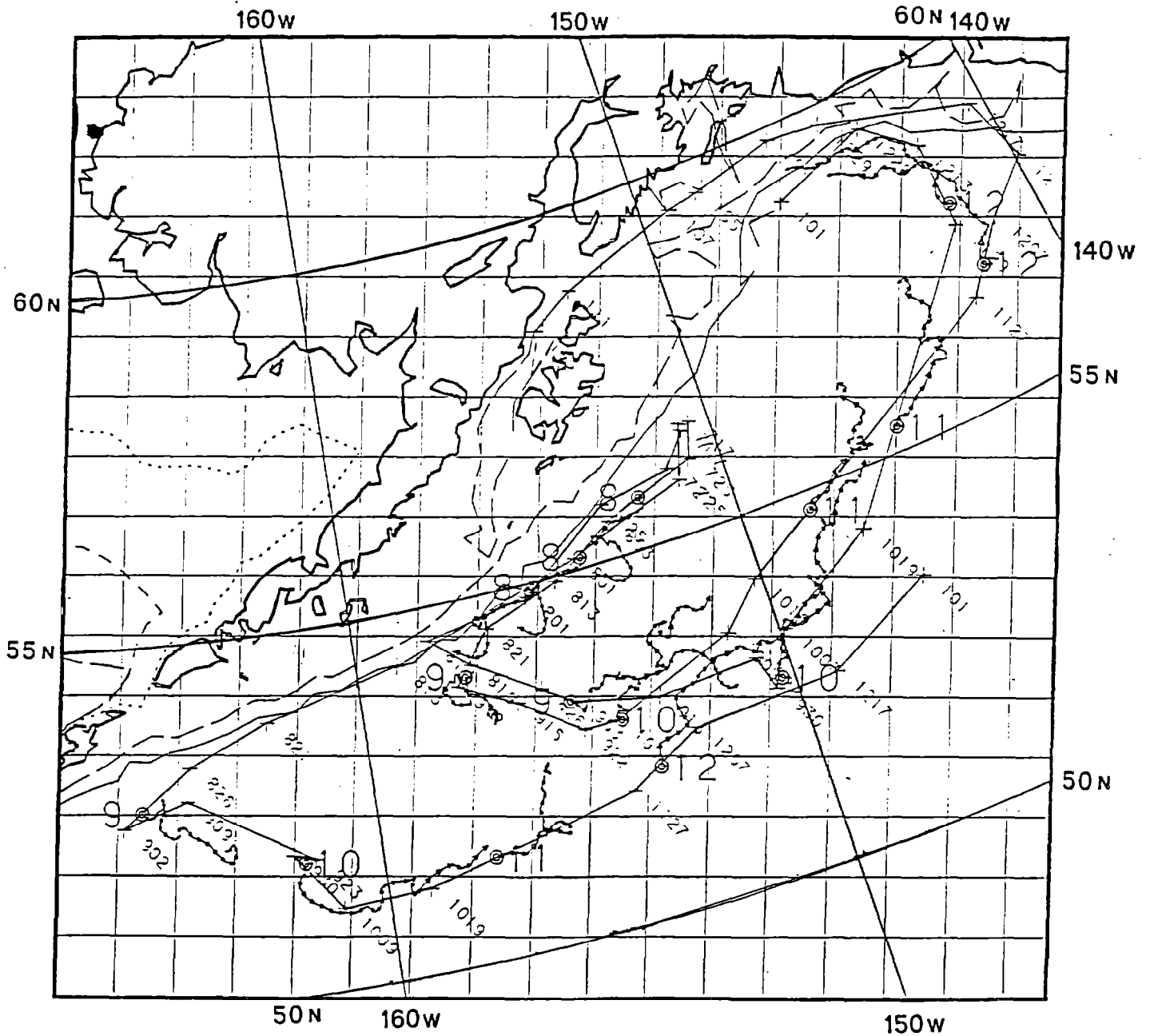


Figure 17. --Three drifter tracks from Reed (1980) and simulated drifter tracks (daily progressive transport vectors) from 30 day runs of the untuned OSCURS numerical model starting at the first day of Months 8, 9, 10, 11, and 12.

REFERENCES

- Bauer, R., and M. Robinson.
1985. Description of the Bauer-Robinson numerical atlas, version VIII, February 1985. Compass Systems, Inc., 4640 Jewell St., #204, San Diego, CA 92109, 13 p.
- Hela, I.
1952. Drift currents and permanent flow. Soc. Sci. Fennica, Comm. Phys. Math. 16(4):1-28.
- Hubert, W. E., and T. Laevastu.
1985. Synoptic analysis and forecasting of surface currents. U.S. Navy Fleet Numerical Weather Facility, Monterey, CA. Technical Note 9., 47 p.
- Ingraham, W. J., Jr., N. Pola Swan, Y. Lee, R. Miyahara, and M. Hayes.
1983. Processing of FNWC daily sea level pressure data on NWAFC Burroughs 7811. REFM Program Doc. 19, 101 p. Northwest and Alaska Fish. Cent., Natl. Mar. Fish. Serv., NOAA, 2725 Montlake Blvd. E., Seattle, WA 98112.
- Ingraham, W. James Jr., and F. Favorite.
1968. The Alaskan Stream south of Adak Island. Deep-Sea Res. 15(4):493-496.
- LaFonde, Eugene C.
1951. Processing oceanographic data. U.S. Navy Hydrographic Office, Washington D. C. Pub. 614, 114 p.

Reed, R. K.

1980. Direct Measurements of Recirculation in the Alaskan Stream. *J. Phys. Ocean.* 10(6): 976-978.

Swan, Nancy Pola.

1983. User's guide to plotting with the GRIDS Package. REFM Program Doc. No. 17, 52 p. Northwest and Alaska Fish. Cent., Natl. Mar. Fish. Serv., NOAA, 2725 Montlake Blvd. E., Seattle, WA 98112.

Witting, R.

1909. Zur Kenntnie den vom Winde Erzeigtem Oberflächenstromes. *Ann. Hydrogr. Marit. Met.* 73, 193 p.

APPENDIX 1

Listing of Fortran Program "M/FINAL" OSCURS Model on 40X104 Grid.

```

1  $RESET FREE
2  $ SET LINEINFO
3
4  FILE 1(KIND=DISK, TITLE="FNWC/LATLON/GRID", FILETYPE=7)
5  FILE 2(KIND=DISK, FILETYPE=7)
6  FILE 5(KIND=READER)
7  FILE 6(KIND=PRINTER)
8  FILE 12(KIND=DISK, TITLE="NUMBERS/GULF", FILETYPE=7)
9  FILE 13(KIND=DISK, TITLE="CURRENTS/40X64/LANDSEA", FILETYPE=7)
10 FILE 14(KIND=DISK, TITLE="CURRENTS/40X64W/LANDSEA", FILETYPE=7)
11 FILE 25(KIND=DISK, TITLE="CURRENTS/40X64W/COAST/DISTANCES/NORMALS"
12     ', FILETYPE=7)
13 FILE 26(KIND=DISK, TITLE="CURRENTS/40X64/COAST/DISTANCES/NORMALS"
14     ', FILETYPE=7)
15 FILE 28(KIND=DISK, TITLE="CURRENTS/40X104/PROGRESVECT/IJPLOTPPOINTS"
16     ' FILETYPE=7)
17 FILE 30(KIND=DISK, TITLE="CURRENTS/40X64/LATLON", FILETYPE=7)
18 FILE 31(KIND=DISK, TITLE="CURRENTS/40X64W/LATLON", FILETYPE=7)
19 FILE 33(KIND=DISK, TITLE="CURRENTS/40X64/DYNHTS/0/2000DB", FILETYPE=
20 FILE 34(KIND=DISK, TITLE="CURRENTS/40X64W/DYNHTS/0/2000DB", FILETYPE
21 FILE 38(KIND=DISK, TITLE="CURRENTS/40X104/DYNCURR/F", MAXRECSIZE=14,
22     'BLOCKSIZE=420, NEWFILE=. TRUE.)
23 FILE 95(KIND=DISK, TITLE=" RED", MAXRECSIZE=14, BLOCKSIZE=420,
24     'NEWFILE=. TRUE.)
25 FILE 96(KIND=DISK, TITLE=" ANG", MAXRECSIZE=14, BLOCKSIZE=420,
26     'NEWFILE=. TRUE.)
27 FILE 50(KIND=DISK, TITLE="CURRENTS/40X104/DYNCURR/S2", FILETYPE=7)
28 FILE 51(KIND=DISK,
29     'MAXRECSIZE=14, BLOCKSIZE=420, NEWFILE=. TRUE.)
30 FILE 52(KIND=DISK,
31     'MAXRECSIZE=14, BLOCKSIZE=420, NEWFILE=. TRUE.)
32 FILE 53(KIND=DISK,
33     'MAXRECSIZE=14, BLOCKSIZE=420, NEWFILE=. TRUE.)
34 FILE 54(KIND=DISK,
35     'MAXRECSIZE=14, BLOCKSIZE=420, NEWFILE=. TRUE.)
36 FILE 55(KIND=DISK,
37     'MAXRECSIZE=14, BLOCKSIZE=420, NEWFILE=. TRUE.)
38 FILE 60(KIND=DISK,
39     'MAXRECSIZE=14, BLOCKSIZE=420, NEWFILE=. TRUE.)
40     COMMON ANX(40, 104), ANY(40, 104), BB(40, 104), LS3(40, 104),
41     'GDIST(40, 104)
42     COMMON/GRDCOM/CFACT, M, N, NM, XI(105), YJ(41)
43     COMMON/LAT/ALAT2, LS2
44     COMMON/LON/ALON2
45     DIMENSION U(880), V(880), LS(40, 104), ALAT(40, 104), ALON(40, 104),
46     DIMENSION B(2)
47     DIMENSION C(1)
48     DIMENSION ALAT2(40, 104), ALON2(40, 104)
49     DIMENSION LS2(40, 104), LS1(40, 104)

```



```

50     DIMENSION OCUW(40,104),OCVW(40,104)
51     DIMENSION CSPEED(40,104)
52     DIMENSION WXX(40,104),WYY(40,104)
53     DIMENSION CSLP(2560)
54     DIMENSION T1(40,104),T2(40,104)
55     DIMENSION SLP1(40,104)
56     DIMENSION HWX(1),HWY(1)
57     DIMENSION HSLP(1),HSFOU(1),HSFOV(1)
58     DIMENSION DEPTH(4),NDATA(4),BATH(3,1000,4)
59     DIMENSION OCUS(40,104),OCVS(40,104)
60     DIMENSION OCUDD(40,104),OCVDD(40,104)
61     DIMENSION GU1(40,104),GV1(40,104),GU2(40,104),GV2(40,104)
62     DIMENSION IPV(500),JPV(500)
63     DIMENSION GIPV(500),GJPV(500)
64     DIMENSION ASP(10,100),AKTS(10),AMPS(10)
65     DIMENSION CC(72),ANGDEF(30),SPJ(30)
66     DIMENSION IAK(100),JAK(100),DDAK(100)
67     DATA A/' SLP YEAR= '/
68     DATA B/' , MONTH= '/
69     DATA C/' , DAY= '/
70     DATA HWX/' WX '/,HWY/' WY '/
71     DATA HSLP/'SLP '/
72     DATA HSFOU/'OUO '/,HSFOV/'OVO '/
73     NN=20;MM=44; NM=NN*MM
74     NN2=40
75     MM2=104
76     C
77     C
78     C SET CHOICES
79     C-----
80     C START MONTH AND NUMBER OF DAYS TO RUN
81         IMSTRT=7
82         IMEND=12
83         NDAYS=366
84     C     NDAYS=121
85     C     NDAYS=15
86         NFILE=51
87     C
88     C SET TUNING FACTORS
89         TFAC=0.0
90         WFAC=2.0
91         DDFAC=0.00
92         AKSTRM=1.0
93     C
94     C SET PRINT FACTOR (NO PRINT, >>>> NOPR=1)
95         NOPR=1
96         NOPR=0
97     C
98     C SET STORAGE OF FILES (NOT KEEP FILES, NOSTOR=1)
99         NOSTOR=1
100    C     NOSTOR=0
101    C SET STORAGE OF FINAL COMPUTED DAILY FIELDS (DO NOT STORE =1)
102    C THIS IS SUCH A LARGE FILE (58000 SECTORS FOR 4 MONTHS)
103        NOSTOF=1
104        NOSTOF=0

```

```

105 C CALL DELTAD ONLY TO STORE A NEW GEOSTROPHIC CURRENT FILE
106 C (SET NEWDD = 1 TO CREATE NEW FILE)
107 NEWDD=1
108 C-----
109 C
110 C READ IN CONTROL CARD PARAMETERS FROM WFL CARD DECK INPUT
111 WRITE (6,80)
112 80 FORMAT ('CONTROL CARD PARAMETERS FROM INPUT WFL CARD DECK'//)
113 84 FORMAT(I5,67A1)
114 85 FORMAT(I5,67A1/)
115 86 FORMAT(F5.1,67A1/)
116 87 FORMAT(F5.1,67A1)
117 READ (5,84)IYRSLP,(CC(I),I=1,67)
118 WRITE(6,85)IYRSLP,(CC(I),I=1,67)
119 READ (5,84)IYRSLP,(CC(I),I=1,67)
120 WRITE(6,85)IYRSLP,(CC(I),I=1,67)
121 WRITE(6,85)IMSTRT
122 READ (5,84)IMSTRT,(CC(I),I=1,67)
123 WRITE(6,85)IMSTRT,(CC(I),I=1,67)
124 WRITE(6,85)IMEND
125 READ (5,84)IMEND,(CC(I),I=1,67)
126 WRITE(6,85)IMEND,(CC(I),I=1,67)
127 WRITE(6,85)NDAYS
128 READ (5,84)NDAYS,(CC(I),I=1,67)
129 WRITE(6,85)NDAYS,(CC(I),I=1,67)
130 WRITE(6,86)WFAC
131 READ (5,87)WFAC,(CC(I),I=1,67)
132 WRITE(6,86)WFAC,(CC(I),I=1,67)
133 WRITE(6,86)DDFAC
134 READ (5,87)DDFAC,(CC(I),I=1,67)
135 WRITE(6,86)DDFAC,(CC(I),I=1,67)
136 WRITE(6,86)AKSTRM
137 READ (5,87)AKSTRM,(CC(I),I=1,67)
138 WRITE(6,86)AKSTRM,(CC(I),I=1,67)
139 WRITE(6,85)NEWDD
140 READ (5,84)NEWDD,(CC(I),I=1,67)
141 WRITE(6,85)NEWDD,(CC(I),I=1,67)
142 WRITE(6,85)NOPR
143 READ (5,84)NOPR,(CC(I),I=1,67)
144 WRITE(6,85)NOPR,(CC(I),I=1,67)
145 WRITE(6,85)NOSTOR
146 READ (5,84)NOSTOR,(CC(I),I=1,67)
147 WRITE(6,85)NOSTOR,(CC(I),I=1,67)
148 WRITE(6,85)NOSTOF
149 READ (5,84)NOSTOF,(CC(I),I=1,67)
150 WRITE(6,85)NOSTOF,(CC(I),I=1,67)
151 C SET UP FNOC GRID (LAT-LONG, LAND-SEA)
152 C
153 C READ IN LAT-LONG OF 20X44 GRID (FNMC GRID POINTS)
154 DO 300 I=1,20
155 DO 300 J=1,44
156 300 READ(1,1200,END=310)IDUM1, IDUM2, (ALAT(I,J), ALON(I,J))
157 1200 FORMAT(2I5,2F10.5)
158 C
159 C TEST WRITE LATS AND LONGS (20X44 GRID BY 2'S)

```

```

160 C WRITE(6,311)
161 311 FORMAT('1 FNWC LAT-LONGS'//)
162 C DO 301 I=1,20
163 C WRITE (6,304) (ALAT(I,J),J=1,44,2)
164 301 CONTINUE
165 304 FORMAT(23F6.1)
166 C DO 302 I=1,20
167 C WRITE (6,304) (ALON(I,J),J=1,44,2)
168 302 CONTINUE
169 C
170 C
171 C LOOK AT GRID DISTANCES NEAR 170 W (WHERE GRID IS ALONG A MERIDIA
172 C TEST GRID LENGTH FUNCTION (GRIDLN) VS. LATITUDE
173 WRITE(6,311)
174 NBR=0
175 SUM1=0.0
176 SUM2=0.0
177 SUM3=0.0
178 SUM4=0.0
179 DO 308 I=2,19
180 DAA=ALAT(I,22)-ALAT(I+1,22)
181 DAA2=ALAT(I-1,22)-ALAT(I+1,22)
182 DLAT=ALAT(I,22)-(DAA/2.0)
183 DLAT2=ALAT(I,22)
184 DAANM=DAA*60.0
185 DAANM2=DAA2*60.0
186 DAANM2=DAANM2/2.0
187 DAAKM=DAANM*1.852
188 DAAKM2=DAANM2*1.852
189 AKM=(1.0+(SIN(DLAT*3.14159/180.0)))/1.86603
190 AKM2=(1.0+(SIN(DLAT2*3.14159/180.0)))/1.86603
191 GD60=DAAKM/AKM
192 GD602=DAAKM2/AKM2
193 WRITE (6,299) (ALAT(I,J),J=15,25)
194 WRITE (6,307) I,ALAT(I,22),DLAT,DAA,DAANM,DAAKM,AKM,GD60
195 WRITE (6,307) I,ALAT(I,22),DLAT2,DAA2,DAANM2,DAAKM2,AKM2,GD60
196 '2
197 NBR=NBR+1
198 SUM1=SUM1+DAAKM
199 SUM2=SUM2+GD60
200 SUM3=SUM3+DAAKM2
201 SUM4=SUM4+GD602
202 308 CONTINUE
203 AV1=SUM1/FLOAT(NBR)
204 AV2=SUM2/FLOAT(NBR)
205 AV3=SUM3/FLOAT(NBR)
206 AV4=SUM4/FLOAT(NBR)
207 WRITE(6,307)NBR,AV1,AV3,AV2,AV4
208 DO 309 I=1,20
209 WRITE (6,299) (ALON(I,J),J=15,25)
210 309 CONTINUE
211 307 FORMAT(15,20X,7F11.6)
212 C
213 299 FORMAT(11F11.6)
214 C

```

```

215 C
216 WRITE (6,61)
217 DO 65 N=1,56
218 GLA=FLOAT(71-N)
219 GG=GDFNOC(GLA)
220 GGNM=GG/1.852
221 WRITE(6,62)N, GLA, GG, GGNM
222 65 CONTINUE
223 310 CONTINUE
224 C
225 C
226 C SET UP CURRENTS (40X104) GRID
227 C
228 C FILL LANDSEA TABLE FOR SLP COMPUTATIONS (ALL=1)
229 DO 211 I=1,40
230 DO 211 J=1,104
231 211 LS1(I,J)=1
232 C
233 C READ IN LAT-LONGS OF 40X64 + 40X64W GRIDS (CURRENTS MODEL)
234 C FILL RIGHT SIDE THEN LEFT SIDE TO CREATE WHOLE OCEAN ARRAY
235 C
236 DO 630 IFILE=1,2
237 DO 631 IK=1,4
238 DO 632 J=41,104
239 I2=IK*10
240 I1=I2-9
241 IF(IFILE.EQ.1)READ(30,104)(ALAT2(I,J),I=I1,I2)
242 IF(IFILE.EQ.2)READ(30,104)(ALON2(I,J),I=I1,I2)
243 632 CONTINUE
244 631 CONTINUE
245 630 CONTINUE
246 104 FORMAT (10F8.3,I4)
247 DO 700 IFILE=1,2
248 DO 701 IK=1,4
249 DO 702 J=1,64
250 I2=IK*10
251 I1=I2-9
252 IF(IFILE.EQ.1)READ(31,104)(ALAT2(I,J),I=I1,I2)
253 IF(IFILE.EQ.2)READ(31,104)(ALON2(I,J),I=I1,I2)
254 702 CONTINUE
255 701 CONTINUE
256 700 CONTINUE
257 C
258 C ROUGH PRINT LATS AND LONGS (EVERY 5TH COL.)
259 C WRITE(6,59)
260 59 FORMAT ('1 CURRENTS (40X104) LAT--LONGS (EVERY 5TH LONG)')
261 C DO 51 I=1,40
262 C WRITE(6,105) (ALAT2(I,J),J=1,104,5),ALAT2(I,104)
263 51 CONTINUE
264 DO 52 I=1,40
265 C WRITE(6,105) (ALON2(I,J),J=1,104,5),ALON2(I,104)
266 52 CONTINUE
267 105 FORMAT (22F6.1)
268 C
269 C LOOK AT GRID DISTANCES NEAR 170 W (WHERE GRID IS ALONG A MERIDIA

```

```

270 C TEST GRID LENGTH FUNCTION (GRIDLN) VS. LATITUDE
271 WRITE(6,59)
272 NBR=0
273 SUM1=0.0
274 SUM2=0.0
275 SUM3=0.0
276 SUM4=0.0
277 DO 56 I=2,39
278 DAA=ALAT2(I,61)-ALAT2(I+1,61)
279 DAA2=ALAT2(I-1,61)-ALAT2(I+1,61)
280 DLAT=ALAT2(I,61)-(DAA/2.0)
281 DLAT2=ALAT2(I,61)
282 DAANM=DAA*60.0
283 DAANM2=DAA2*60.0
284 DAANM2=DAANM2/2.0
285 DAAKM=DAANM*1.852
286 DAAKM2=DAANM2*1.852
287 AKM=(1.0+(SIN(DLAT*3.14159/180.0)))/1.86603
288 AKM2=(1.0+(SIN(DLAT2*3.14159/180.0)))/1.86603
289 GD60=DAAKM/AKM
290 GD602=DAAKM2/AKM2
291 WRITE (6,299) (ALAT2(I,J),J=55,65)
292 WRITE (6,307) I,ALAT2(I,61),DLAT,DAA,DAANM,DAAKM,AKM,GD60
293 WRITE (6,307) I,ALAT2(I,61),DLAT2,DAA2,DAANM2,DAAKM2,AKM2,GD6
294 '2
295 NBR=NBR+1
296 SUM1=SUM1+DAAKM
297 SUM2=SUM2+GD60
298 SUM3=SUM3+DAAKM2
299 SUM4=SUM4+GD602
300 56 CONTINUE
301 AV1=SUM1/FLOAT(NBR)
302 AV2=SUM2/FLOAT(NBR)
303 AV3=SUM3/FLOAT(NBR)
304 AV4=SUM4/FLOAT(NBR)
305 WRITE(6,307)NBR,AV1,AV3,AV2,AV4
306 DO 57 I=1,20
307 WRITE (6,299) (ALON2(I,J),J=55,65)
308 57 CONTINUE
309 C
310 WRITE (6,61)
311 61 FORMAT('1GRID LENGTH (KM) VS. LATITUDE'//)
312 DO 60 N=1,37
313 GLA=FLOAT(67-N)
314 GG=GRIDLN(GLA)
315 GGNM=GG/1.852
316 WRITE(6,62)N,GLA,GG,GGNM
317 60 CONTINUE
318 62 FORMAT(I4,'LATITUDE=',F6.2,' GRID LENGTH=',F11.6,' KM',5X,
319 'F11.6.' NM')
320 C
321 C READ LAND-SEA TABLE (LEFT AND RIGHT)
322 DO 1500 I=1,NN2
323 READ(14,1501,END=1503)(LS2(I,J),J=1,64)
324 C WRITE(6,1505) I,LS2(I,1)

```

```

325     1500 CONTINUE
326     1503 CONTINUE
327     1505 FORMAT (1H ,2I5)
328     1501 FORMAT(64I1)
329         DO 1600 I=1,NN2
330             READ(13,1501,END=1603)(LS2(I,K),K=41,104)
331     C     WRITE(6,1505) I,LS2(I,41)
332     1600 CONTINUE
333     1603 CONTINUE
334     C WRITE LAND-SEA
335         WRITE(6,1514)
336     1514 FORMAT('1 LAND-SEA 2, (40X104)'//)
337         DO 1515 I=1,NN2
338             WRITE(6,1516)(LS2(I,J),J=1,104)
339     1515 CONTINUE
340     1516 FORMAT(1H ,104I1)
341         WRITE(6,110)
342     110 FORMAT('1LANDSEA 3'//)
343         DO 220 I=1,40
344             DO 221 J=1,MM2
345                 LS3(I,J)=LS2(I,J)
346     221 CONTINUE
347         WRITE(6,1509) (LS3(I,JK),JK=1,MM2),I
348     220 CONTINUE
349     1509 FORMAT(1H ,104I1,I5)
350     C
351     C TO ROTATE CURRENT VECTORS IMPINGING ON THE COAST
352     C READ IN DISTANCES AND NORMALS TO THE COASTLINE
353     C FOR EACH GRID POINT WHICH IS WITHIN 2 GRID LENGTHS OF THE COAS
354     C DATA ARE USED IN COMPUTING CURRENT VECTOR ROTATION AND REDUCTION
355     C FOR VECTORS WHICH COME WITHIN 2 GRID POINTS OF SHORE
356     C AN EXPONENTIAL FUNCTION IS USED TO REDUCE THE MAGNITUDE
357     C DEPENDING UPON THE DISTANCE FROM SHORE
358     C AND ROTATION IS BASED ON BOTH THE DIRECTION AND MAGNITUDE
359     C OF THE ANGLE OF INCIDENCE
360     C AGAIN THE OCEAN WIDE ARRAY IS FILLED FROM THE TWO SMALLER
361     C FILES OF THE EAST AND WEST PREVIOUS VERSIONS OF M(MODEL)
362     C 324 READ(25,325,END=329)I,J,GDIST(I,J),ANY(I,J),ANX(I,J)
363     324 READ(25,325,END=329)I,J,V1,VV,V3
364         GDIST(I,J)=V1
365         ANY(I,J)=VV
366         ANX(I,J)=V3
367     C WRITE(6,327)I,J,GDIST(I,J),ANY(I,J),ANX(I,J)
368         LS3(I,J)=-1
369         GO TO 324
370     325 FORMAT(2I3,F5.2,F6.2,F6.2)
371     327 FORMAT(' WEST ',2I3,F5.2,F6.2,F6.2)
372     329 CONTINUE
373     C READ IN DISTANCES AND NORMALS FROM FILE 26
374     C 424 READ(26,425,END=429)I,J,GDIST(I,J),ANY(I,J),ANX(I,J)
375     424 READ(26,425,END=429)I,J,V1,VV,V3
376     C OFFSET FOR EASTERN GRID
377         J=J+40
378         GDIST(I,J)=V1
379         ANY(I,J)=VV

```

```

380     ANX(I,J)=V3
381 C   WRITE(6,427)I,J,GDIST(I,J),ANY(I,J),ANX(I,J)
382     LS3(I,J)=-1
383     GO TO 424
384     425 FORMAT(2I3,F5.2,F6.2,F6.2,)
385     427 FORMAT(' EAST ',2I3,F5.2,F6.2,F6.2,)
386     429 CONTINUE
387 C
388 C   TEMPORARY LAND-SEA TABLE = ALL 1'S
389     WRITE(6,110)
390     DO 1506 I=1,40
391     DO 1507 J=1,104
392     LS1(I,J)=1
393     1507 CONTINUE
394     WRITE(6,1509) (LS3(I,JK),JK=1,MM2),I
395     1506 CONTINUE
396 C
397 C
398 C -----
399 C   READ INITIAL I,J PLOINT TO PLOT IN PROGRESSIVE VECTOR PLOT
400 C -----
401     KK=1
402     1144 READ(28,1145,END=1146) IPV(KK),JPV(KK),GIPV(KK),GJPV(KK)
403     WRITE(6,1147) KK,IPV(KK),JPV(KK),GIPV(KK),GJPV(KK)
404     IF(IPV(KK).LE.0.OR.JPV(KK).LE.0) GO TO 1146
405     KK=KK+1
406     GO TO 1144
407     1145 FORMAT(2I5,2F6.1)
408     1147 FORMAT (3I5,2F6.1)
409     1146 CONTINUE
410     KK=KK-1
411     WRITE(6,1145) KK
412     IEND=KK
413 C
414     LSKIP=1
415     IF(LSKIP.EQ.1) GO TO 8888
416 C   SET STARTING GRID POINTS FOR PROGRESSIVE VECTORS
417 C   DO EVEN NUMBERED GRID POINTS BY 4'S
418 C   SAVE ONLY EVERY 4 GRID POINTS FOR PLOT AND TAPE FILES
419     LC=1
420 C   HORIZONTAL (12 TO 96 BY 4'S)
421     DO 40 L2=1,22
422     LJ=12+((L2-1)*4)
423 C   VERTICAL (8 TO 36 BY 4'S)
424     DO 41 L1=1,8
425     LI=8+((L1-1)*4)
426     IF (LS3(LI,LJ).EQ.0) GO TO 41
427     IPV(LC)=LI
428     JPV(LC)=LJ
429     LC=LC+1
430     41 CONTINUE
431     40 CONTINUE
432     KK=LC-1
433     WRITE(6,1145) KK
434     IEND=KK

```

```

435      8888 CONTINUE
436      C
437      C
438      C INITIALIZE GRID PLOT (UNITS ARE IN GRID VALUES)
439      C RETAIN ACTUAL GRID UNITS IN GU AND GV ARRAYS
440      C STARTING AT THE UPPER LEFT CORNER OF GRID (AS SEEN IN MAP VIEW)
441      C I INCREASES IN THE MINUS Y DIRECTION FROM 1 TO NN2(40)
442      C J INCREASES IN THE PLUS X DIRECTION FROM 1 TO MM2(104)
443          DO 3105 I=1,NN2
444          DO 3106 J=1,MM2
445              GU1(I,J)=J
446              GV1(I,J)=I
447      3106 CONTINUE
448      3105 CONTINUE
449      C
450      C ADD SPECIAL HALF GRID POINTS TO START COMPUTING
451          GU1(10,73)= 72.5
452          GV1(10,73)=10.5
453          GU1(11,74)=73.5
454          GV1(11,74)=11.5
455      C IF(1.EQ.1) GO TO 3190
456          DO 3189 K=1,KK
457              GU1(IPV(K),JPV(K))=GJPV(K)
458              GV1(IPV(K),JPV(K))=GIPV(K)
459              WRITE(6,1147)K,IPV(K),JPV(K),GU1(IPV(K),JPV(K)),GV1(IPV(K),JP
460      3189 CONTINUE
461      3190 CONTINUE
462      C
463      C
464      C -----
465      C COMPUTE GEOSTROPHIC OCEAN CURRENTS 0/2000 DB
466      C CALL DELTAD ONLY TO STORE A NEW GEOSTROPHIC CURRENT FILE
467      C (SET NEWDD = 1 TO CREATE NEW FILE)
468      C NEWDD=1
469      C -----
470          IF (NEWDD.EQ.1) CALL DELTAD
471      C
472      C -----
473      C READ PERMANENT FILE OF GEOSTROPHIC CURRENTS, U THEN V)
474      C -----
475          DO 38 J=1,104
476          DO 38 I=1,5
477              I2=I*8 ; I1=I2-7
478              READ(50,100,END=218)(OCUDD(IK,J),IK=I1,I2),KYR8,MO8,KDY8
479      38 CONTINUE
480      218 CONTINUE
481          DO 39 J=1,104
482          DO 39 I=1,5
483              I2=I*8 ; I1=I2-7
484              READ(50,100,END=219)(OCVDD(IK,J),IK=I1,I2),KYR9,MO9,KDY9
485      39 CONTINUE
486      219 CONTINUE
487      C
488      C OCEAN GEOSTROPHIC CURRENT VECTOR SPEED
489          DO 954 I=1,NN2

```



```

490      DO 955 J=1,MM2
491      CSPEED(I,J)=SQRT((OCUDD(I,J)*OCUDD(I,J))+(OCVDD(I,J)*OCVDD(I,
492      955 CONTINUE
493      954 CONTINUE
494      C   IF (NOPR.EQ.1) GO TO 957
495      WRITE(6,956) KYR,IMO,IDY
496      956 FORMAT('1 ORIGINAL ** GEOSTROPHIC SPEED (CM/SEC)',313//)
497      CALL PRINT2(CSPEED,1.0)
498      C
499      C READ IN ALASKAN STREAM AMPLIFICATION FACTOR
500      K=1
501      601 READ(12,602,END=609)IAK(K),JAK(K),DDAK(K)
502      DDAK(K)=DDAK(K)*AKSTRM
503      WRITE(6,604) K,IAK(K),JAK(K),DDAK(K)
504      K=K+1
505      GO TO 601
506      609 CONTINUE
507      602 FORMAT(2I4,F7.2)
508      604 FORMAT (3I4,F7.2)
509      DO 406 J=1,104
510      DO 407 I=1,40
511      DDAMP=1.0
512      DO 611 KK=1,K
513      IF(IAK(KK).EQ.I.AND.JAK(KK).EQ.J)GO TO 651
514      611 CONTINUE
515      GO TO 3333
516      651 DDAMP=1.0+DDAK(KK)
517      C MULTIPLY DD CURRENT BY DDAMP FACTOR HERE
518      OCUDD(I,J)=OCUDD(I,J)*DDAMP
519      OCVDD(I,J)=OCVDD(I,J)*DDAMP
520      3333 CONTINUE
521      C MULTIPLY DD CURRENT BY DD FACTOR HERE
522      OCUDD(I,J)=OCUDD(I,J)*DDFAC
523      OCVDD(I,J)=OCVDD(I,J)*DDFAC
524      IF (LS3(I,J).EQ.0) OCUDD(I,J)=0.0
525      IF (LS3(I,J).EQ.0) OCVDD(I,J)=0.0
526      407 CONTINUE
527      406 CONTINUE
528      C
529      C WRITE OUT U AND V OF GEOSTROPHIC CURRENT FOR FINE TUNING
530      WRITE(6,951)
531      951 FORMAT ('1 U -- COMPONENT OF GEOSTROPHIC CURRENT'//)
532      CALL PRINT2(OCUDD,1.0)
533      WRITE(6,952)
534      952 FORMAT ('1 V -- COMPONENT OF GEOSTROPHIC CURRENT'//)
535      CALL PRINT2(OCVDD,1.0)
536      C
537      C OCEAN GEOSTROPHIC CURRENT VECTOR SPEED
538      DO 944 I=1,NN2
539      DO 945 J=1,MM2
540      CSPEED(I,J)=SQRT((OCUDD(I,J)*OCUDD(I,J))+(OCVDD(I,J)*OCVDD(I,
541      945 CONTINUE
542      944 CONTINUE
543      C   IF (NOPR.EQ.1) GO TO 947
544      WRITE(6,946) KYR,IMO,IDY

```

```

545 946 FORMAT('1GEOSTROPHIC SPEED (CM/SEC)',313//)
546 CALL PRINT2(CSPEED,1.0)
547 947 CONTINUE
548 C
549 C
550 C DESCRIBE METHOD OF OBTAINING ANGLE OF DEFLECTION FOR SURFACE WIN
551 C DUE TO FRICTION AT SURFACE OF OCEAN (INTO CENTER OF LOW -- TO LE
552 C AS A FUNCTION OF WIND SPEED (M/SEC) AND LATITUDE
553 C
554 WRITE(6,543)
555 543 FORMAT('1TABLE 3.--FRICTIONAL EFFECTS ON COMPUTED WINDS, REDU
556 ' FACTOR AND ANGLE OF DEFLECTION')
557 WRITE(6,547)
558 547 FORMAT(11X,' OF SURFACE WIND TO THE LEFT OF THE PURE GEOSTROP
559 'IND'/)
560 WRITE(6,546)
561 WRITE(6,544)
562 546 FORMAT(44X,'WIND SPEED (M/SEC)')
563 544 FORMAT(' LAT REDUCTION 0 2 4 6 8 10
564 ' 14 16 18 20')
565 GLAT1=67.0
566 C DECREASE LATITUDE FROM 67N TO 20N BY 1 DEGREE INCREMENTS
567 KK=0
568 DO 540 I=1,48
569 GLAT=GLAT1-I+1
570 GLATR=GLAT/57.2958
571 SINLAT=SIN(GLATR)
572 C COMPUTE MANGITUDE REDUCTION (H U ROLL METHOD)
573 IF(GLAT.GT.45.0) RED=0.75+(0.1*((90.0-GLAT)/45.0))
574 IF(GLAT.LE.45.0.AND.GLAT.GE.25.0) RED=0.85
575 IF(GLAT.LT.25.0) RED=0.65+(0.2*(GLAT/25.0))
576 C COMPUTE DEFLECTION ANGLE AS A FUNCTION OF WIND SPEED
577 JJ=0
578 DO 541 J=1,21,2
579 JJ=JJ+1
580 COMP=J-1
581 XCOMP=RED*COMP
582 YCOMP=RED*COMP
583 SPEED=SQRT((XCOMP*XCOMP)+(YCOMP*YCOMP))
584 SPJ(JJ)=SPEED
585 ANG=(22.5-(0.0175*SPEED*SPEED))*(1.495/(SINLAT+1.0))
586 ANGDEF(JJ)=ANG
587 541 CONTINUE
588 C IF(KK.EQ.0)WRITE(6,545) (SPJ(JK),JK=1,11)
589 WRITE(95,542)GLAT,RED,(ANGDEF(JJJ),JJJ=1,11)
590 WRITE(6,542)GLAT,RED,(ANGDEF(JJJ),JJJ=1,11)
591 KK=KK+1
592 542 FORMAT(1X,F5.1,2X,F6.3,3X,12F6.1)
593 545 FORMAT (16X,11I6/)
594 540 CONTINUE
595 CLOSE(95,DISP=CRUNCH)
596 C
597 C
598 C
599 C DESCRIBE METHOD OF OBTAINING THE ROTATION ANGLE FOR

```

```

600 C SURFACE OCEAN CURRENTS DUE TO WIND
601 C ANGLE TO THE RIGTH OF THE WIND AS A FUNCTION OF WIND SPEED AND L
602 C REFERENCE FNWC TECHN. NOTE #9, 1967
603 C PAGE 10
604 C DEFLECTION ANGLE AS A FUNCTION OF SPEED ONLY IS:
605 C ANGLE=34(DEG)-7.5XSQRT V (M/SEC)
606 C A CONSTRAINT HERE = THE ANGLE APPROACHES ZERO AT 40 KTS
607 C PAGE 11
608 C ANGLE AS FN OF LAT ONLY BUT FOR 6 KT (3.088 M/SEC) WIND ONLY IS:
609 C GRAP DETERMINED A LINEAR RELATION BETWEEN 30-60 N (MODEL LAT R
610 C ANGLE=(0.2933)(LATITUDE) + 2.4
611 C
612 C BY MAKING INITIAL ANGLE OF DEFLECTION (34.0 IN ABOVE EQN)
613 C A LINEAR FN OF LATITUDE WITH THE ABOVE SLOPE (0.2933)
614 C CL=(34.0-((67.0-YLAT)*0.2933))
615 C AND HOLDING THE CONSTRAINT THAT THE ANGLE APPROACHES ZERO AT 4
616 C SP40=40.0*0.5148
617 C THE NEW COEFICIENT AS A FN OF LAT AT 40 KTS IS:
618 C CLV=CL/SQRT(SP40)
619 C THEREFORE:
620 C THE ANGLE OF DEFLECTION AS A FN OF LAT AND WIND SPEED IS:
621 C ASP(I,J)
622 C
623 YLAT1=67.0
624 DO 565 I=1,10
625 AI=5.0*(I-1)
626 SP=AI*0.5148
627 DO 566 J=1,48
628 YLAT=YLAT1-((J-1))
629 CL=(34.0-((YLAT1-YLAT)*0.2933))
630 CLV=CL/SQRT(SP40)
631 ASP(I,J)=CL-(CLV*SQRT(SP))
632 566 CONTINUE
633 565 CONTINUE
634 DO 567 I=1,10
635 AKTS(I)=(I-1)*5.0
636 AMPS(I)=AKTS(I)*0.5148
637 567 CONTINUE
638 WRITE (6,571)
639 571 FORMAT('1TABLE 4.--LE OF DEFLECTION OF SURFACE CURRENT TO THE
640 'T OF WIND')
641 WRITE(6,572)
642 572 FORMAT(' AS A FUNCTION OF LATITUDE AND WIND SPEED'/
643 WRITE(6,573) (AKTS(I),I=1,9)
644 573 FORMAT(' WIND SPEED',9I7,' (KTS)')
645 WRITE(6,574) (AMPS(I),I=1,9)
646 574 FORMAT(12X,9F7.2,' (M/SEC)'/)
647 WRITE(6,577)
648 577 FORMAT(' LATITUDE ANGLE OF DEFLECTION (DEGREES
649 DO 568 J=1,48
650 YLT=YLAT1-((J-1))
651 WRITE(96,575) YLT,(ASP(I,J),I=1,9)
652 568 CONTINUE
653 575 FORMAT (F6.1,6X,9F7.2)
654 WRITE(6,576)

```

```

655     576 FORMAT (1H1//)
656     CLOSE(96,DISP=CRUNCH)
657     C
658     C
659     C TEMPORARY STOP
660     C GO TO 901
661     C
662     C
663     C DO 30 DAYS OF PLOTS
664     C DO 10000 IDY9=1,NDAYS
665     C
666     C IF (IDY9.GT.2) NOPR=1
667     C WRITE(6,6002) IDY9
668     6002 FORMAT(' START PROCESSING DAY NUMBER ',I4)
669     C
670     11 CONTINUE
671     C
672     C READ IN SLP FOR LARGE SCALE FNOC GRID
673     C CALL READER(IMO,K,IDY,ITYP,SLP1)
674     C 999=EOF
675     C CLOSE FILES FOR STORED RECORDS
676     C IF(ITYP.EQ.999)GO TO 909
677     C IF (IMO.LT.IMSTRT.AND.IYRSLP.EQ.K) GO TO 11
678     C IF (IMO.GT.IMEND) GO TO 909
679     C KYR=K
680     C YEAR=K+1900
681     C DAY=IDY
682     C
683     107 FORMAT(13F9.2)
684     C
685     C IF(WFAC.LE.0.0) GO TO 6001
686     C CONVERT DSLP UNITS BACK TO MILLIBARS FROM PACKED STORAGE UNITS
687     C DO 25 I=1,NN
688     C DO 26 J=1,MM
689     C SLP1(I,J)=(SLP1(I,J)/10.0)+1000.0
690     26 CONTINUE
691     25 CONTINUE
692     C
693     C
694     226 CONTINUE
695     C
696     C IF(NOPR.EQ.1) GO TO 653
697     C ROUGH PRINT SLP
698     C DO 652 I=1,20
699     C WRITE(6,107) (SLP1(I,J),J=1,44,4),SLP1(I,44)
700     652 CONTINUE
701     C
702     653 CONTINUE
703     C
704     C
705     C
706     C BESSEL 4PT. INTERPOLATION FROM FNWC GRID TO CURRENTS 40X64 GRID
707     C FIND THE I,J GRID COORDINATES OF THE CURRENTS MODEL GRID
708     C RELATIVE TO THE FNWC GRID FOR USE IN BESSEL INTERPOLATION
709     C NOTE ; WE KNOW THE I,J'S OF THE CURRENTS GRID

```

```

710 C           THEY ARE 1/4 OF THE FNWC SUBSET
711           NC=1
712           IOK=99
713           IOK=0
714           DO 401 I=1,NN2
715           DO 402 J=1,MM2
716           XI1=ALON2(I,J)
717           XI2=XI1
718           XJ1=ALAT2(I,J)
719           XJ2=XJ1
720 C           CALL CONVIJ (XI2,XJ2,IERR,XLEN,YYLEN,MM,NN,ALAT,ALON)
721 C
722 C           COMPUTE I,J'S OF 40X64 GRID RELATIVE TO 20X44 GRID
723           XIXI=2.0+(FLOAT(I-1)*0.25)
724           XJ=7.0+(FLOAT(J-1)*0.25)
725 C           XI=XI2
726 C           XJ=XJ2
727 C
728 C           COMPARE BESSEL AND SPLINE INTERPOLATIONS
729 C           IF(I.EQ.20)CALL BESSEL(XJ,XIXI,IOK,VV,SLP1)
730           CALL SPLINE(XJ,XIXI,IOK,UU,SLP1)
731 C           IF(I.EQ.2) GO TO 901
732 C           CALL BESSEL(XI,XJ,IOK,VV,V2)
733           T1(I,J)=UU
734 C           T2(I,J)=VV
735           DUUVV=UU-VV
736 C           IF(I.EQ.20)WRITE(6,404)I,J,XI1,XJ1,XIXI,XJ,UU,VV,IERR,IOK,DUU
737           402 CONTINUE
738           401 CONTINUE
739           404 FORMAT(2I5,6F10.4,2I5,'   SPLINE-BESSEL',F15.11)
740 C
741           IF(NOPR.EQ.1) GO TO 709
742 C           PRINT 40X64 SLP FILE
743           DO 667 K=1,7
744           K2=16*K
745           K1=K2-15
746           IF(K2.GT.104) K2=104
747           WRITE(6,664) K1,K2
748           WRITE(6,671)
749           DO 662 I=1,NN2
750           WRITE(6,663) I,(T1(I,J),J=K1,K2)
751           662 CONTINUE
752           663 FORMAT(I3,18F7.1)
753           664 FORMAT('1SLP (40X104) 8Y 16S',5X,'FROM J=',I2,2X,'TO J=',I3)
754           671 FORMAT('   SPLINE' /)
755           669 FORMAT ('   BESSEL' /)
756           667 CONTINUE
757 C
758 C
759 C           SMOOTH BESSEL FUNCTION INTERPOLATED SLP FIELD
760 C           CALL SMTH(5,0.7,T1,T2)
761 C
762 C
763 C
764 C           PRINT SMOOTHED 40X64 SLP FILE

```

```

765 C WRITE(6,664)
766 C WRITE (6,669)
767 C DO 772 I=1,NN2
768 C WRITE(6,663) I,(T2(I,J),J=1,18)
769 772 CONTINUE
770 C
771 C WRITE(6,664)
772 C WRITE (6,669)
773 C DO 776 I=1,NN2
774 C WRITE(6,663) I,(T2(I,J),J=19,36)
775 776 CONTINUE
776 C
777 C WRITE(6,664)
778 C WRITE (6,669)
779 C DO 777 I=1,NN2
780 C WRITE(6,663) I,(T2(I,J),J=37,50)
781 777 CONTINUE
782 709 CONTINUE
783 C
784 C STORE FILE ON DISK OR NOT
785 IF(NOSTOR.LT.1)CALL OUTPT(T1,NFILE,KYR,IMO,IDY,HSLP)
786 IF(NOSTOR.LT.1)WRITE(6,992) KYR,IMO,IDY
787 992 FORMAT(' ',3I4,' WRITTEN TO SLP OUTPUT FILE')
788 C
789 C
790 C
791 C
792 C COMPUTE WIND VECTORS FROM SLP FIELD
793 CALL WIND(T1,LS1,ALAT2,WXX,WYY)
794 C
795 IF(NOSTOR.LT.1)CALL OUTPT(WXX,52,KYR,IMO,IDY,HWX)
796 IF(NOSTOR.LT.1)CALL OUTPT(WYY,52,KYR,IMO,IDY,HWY)
797 C
798 C IF (NOPR.EQ.1) GO TO 847
799 C PRESSURE CHANGE PER GRID LENGTH MAGNITUDE
800 DO 27 I=1,NN2
801 DO 28 J=1,MM2
802 IF((I+1).GE.NN2) GO TO 28
803 IF((J+1).GE.MM2) GO TO 28
804 TX=T1(I,J+1)-T1(I,J)
805 TY=T1(I+1,J)-T1(I,J)
806 CSPEED(I,J)=SQRT((TX*TX)+(TY*TY))
807 28 CONTINUE
808 27 CONTINUE
809 IF(IDY9.LT.2)WRITE(6,29) KYR,IMO,IDY
810 29 FORMAT('1ATMOS PRESS CHANGE X 10 (MAGNITUDE/GRID) (MB)',3I3//
811 IF(IDY9.LT.2) CALL PRINT2(CSPEED,10.0)
812 C
813 C WIND VECTOR SPEED
814 DO 844 I=1,NN2
815 DO 845 J=1,MM2
816 CSPEED(I,J)=SQRT((WXX(I,J)*WXX(I,J))+(WYY(I,J)*WYY(I,J)))
817 845 CONTINUE
818 844 CONTINUE
819 C IF (NOPR.EQ.1) GO TO 847

```

```

820         IF(IDY9.LT.2)WRITE(6,846) KYR,IMO,IDY
821     846 FORMAT(' WIND SPEED (M/SEC)',3I3//)
822         IF(IDY9.LT.2) CALL PRINT2(CSPEED,1.0)
823     C
824     847 CONTINUE
825     C
826     C COMPUTE SURFACE OCEAN CURRENT VECTOR FIELD
827     C SEE FUNCTION "WINDC" (SFC CURR DUE TO WIND)
828         DO 1844 I=1,NN2
829         DO 1845 J=1,MM2
830             OCUW(I,J)=WINDC(WXX(I,J),WYY(I,J),1,ALAT2(I,J))
831             OCVW(I,J)=WINDC(WYY(I,J),WXX(I,J),2,ALAT2(I,J))
832             CSPEED(I,J)=SQRT(OCUW(I,J)*OCUW(I,J)+OCVW(I,J)*OCVW(I,J))
833     1845 CONTINUE
834     1844 CONTINUE
835     C
836     C IF(NOPR.EQ.1) GO TO 1849
837         IF(IDY9.LT.2)WRITE(6,1848) KYR,IMO,IDY
838     1848 FORMAT(' SPEED OF SFC-OCN-CURRENT DUE TO WIND (CM/SEC)',3I3//)
839         IF(IDY9.LT.2) CALL PRINT2(CSPEED,1.0)
840         IF(NOPR.EQ.1) GO TO 1849
841         WRITE(6,1846) KYR,IMO,IDY
842     1846 FORMAT(' 1U--COMPONENT OF SURFACE OCEAN CURRENT DUE TO WIND (C
843         ')',3I3//)
844         CALL PRINT2(OCUW,1.0)
845         WRITE(6,1847) KYR,IMO,IDY
846     1847 FORMAT(' 1V--COMPONENT OF SURFACE OCEAN CURRENT DUE TO WIND (C
847         ')',3I3//)
848         CALL PRINT2(OCVW,1.0)
849     C
850     1849 CONTINUE
851     C
852     C
853     C STORE FILE 53 (SURFACE OCEAN CURRENTS DUE TO WIND)
854     C
855         IF(NOSTOR.LT.1)CALL OUTPT(OCUW,53,KYR,IMO,IDY,HSFOU)
856         IF(NOSTOR.LT.1)CALL OUTPT(OCVW,53,KYR,IMO,IDY,HSFOV)
857         IF(NOSTOR.LT.1)WRITE(6,298)KYR,IMO,IDY
858     298 FORMAT('/SFCOCN CURRENTS DUE TO WIND STORED FOR',3I4)
859     C
860     6001 CONTINUE
861     C
862     C
863     C
864     C -----
865     C TUNING AND SUMMATION OF CURRENT COMPONENTS
866     C -----
867     C
868     C MULTIPLY SFC CURRENT DUE TO WIND BY FACTOR HERE
869         DO 410 J=1,104
870         DO 411 I=1,40
871             OCUW(I,J)=OCUW(I,J)*WFAC
872             OCVW(I,J)=OCVW(I,J)*WFAC
873             OCUS(I,J)=OCUW(I,J)+OCUDD(I,J)
874             OCVS(I,J)=OCVW(I,J)+OCVDD(I,J)

```

```

875         IF (LS3(I,J).EQ.0) OCUS(I,J)=0.0
876         IF (LS3(I,J).EQ.0) OCVS(I,J)=0.0
877     C
878     411 CONTINUE
879     410 CONTINUE
880     C
881     C
882     C
883     C WRITE FILE OF TOTAL CURRENTS BEFORE ROTATION
884         IF(NOSTOR.EQ.1) GO TO 227
885         DO 94 J=1,104
886         DO 94 I=1,5
887             I2=I*8
888             I1=I2-7
889             WRITE(55,113,END=224)(OCUS(IK,J),IK=I1,I2),KYR,IMO,IDY
890     94 CONTINUE
891     224 DO 95 J=1,104
892         DO 95 I=1,5
893             I2=I*8
894             I1=I2-7
895             WRITE(55,113,END=225)(OCVS(IK,J),IK=I1,I2),KYR,IMO,IDY
896     95 CONTINUE
897     225 CONTINUE
898     113 FORMAT(8F8.2,3I3,A3)
899     C
900     227 CONTINUE
901     C
902     C
903     C
904     C -----
905     C           COASTAL BOUNDARY EFFECTS
906     C ROTATION AND REDUCTION OF VECTORS IMPINGING UPON THE COASTLINE
907     C -----
908     C RESULTS IN FILE 54
909     C
910     C
911     C
912         CALL CSTROT(KYR,IMO,IDY,OCUS,OCVS,NOSTOF)
913     C
914     C IF(NOPR.EQ.1) GO TO 2151
915     C
916         DO 2144 I=1,NN2
917         DO 2145 J=1,MM2
918             CSPEED(I,J)=SQRT(OCUS(I,J)*OCUS(I,J)+OCVS(I,J)*OCVS(I,J))
919     2145 CONTINUE
920     2144 CONTINUE
921         IF(IDY9.LT.2)WRITE(6,2148) KYR,IMO,IDY
922         IF(IDY9.LT.2)WRITE(6,2149) WFAC,TFAC,DDFAC
923     2148 FORMAT('1SPEED OF TOTAL SFC-OCN-CURRENT (CM/SEC)',3I3)
924         IF(IDY9.LT.2) CALL PRINT2(CSPEED,1.0)
925     C
926         IF(NOPR.EQ.1) GO TO 2151
927         WRITE(6,2146) KYR,IMO,IDY
928     2146 FORMAT('1U--COMPONENT OF TOTAL SURFACE OCEAN CURRENT (CM/SEC)
929         ',3I3)

```



```

930      WRITE(6,2149) WFAC,TFAC,DDFAC
931 2149 FORMAT(10X,'WIND FACTOR =',F5.2,5X,'THERMAL FACTOR =',F5.2,5X
932 'STROPHIC FACTOR =',F5.2/)
933      CALL PRINT2(OCUS,1.0)
934      WRITE(6,2147) KYR,IMO,IDY
935      WRITE(6,2149) WFAC,TFAC,DDFAC
936 2147 FORMAT('1V--COMPONENT OF TOTAL SURFACE OCEAN CURRENT (CM/SEC)
937 ',3I3)
938      CALL PRINT2(OCVS,1.0)
939 C
940      WRITE(6,296)KYR,IMO,IDY,WFAC,TFAC,DDFAC
941 296 FORMAT('/TOTAL SFCOCN CURRENTS (40X104) STORED FOR',3I4,5X,'W
942 ',F4.2,5X,'TFAC=',F4.2,5X,'DDFAC=',F4.2)
943 C
944 2151 CONTINUE
945 C
946 C
947 C
948 C -----
949 C          TRANSPORT (PROGRESSIVE) VECTORS
950 C -----
951 C
952 C
953      DO 2964 KK=1,IEND
954      I=IPV(KK)
955      J=JPV(KK)
956      IF(LS3(I,J).EQ.0)OCUS(I,J)=0.0
957      IF(LS3(I,J).EQ.0)OCVS(I,J)=0.0
958      U1=OCUS(I,J)
959      V1=OCVS(I,J)
960 C
961 C HANDLE OFF MAP POINTS
962      IF(GV1(I,J).GE.38.99)GV1(I,J)=38.99
963      IF(GV1(I,J).LE.1.0)GV1(I,J)=1.0
964      IF(GU1(I,J).GE.102.99)GU1(I,J)=102.99
965      IF(GU1(I,J).LE.1.0)GU1(I,J)=1.0
966 C
967      CALL SPLINE(GU1(I,J),GV1(I,J),IOK,UU,OCUS)
968      CALL SPLINE(GU1(I,J),GV1(I,J),IOK,VV,OCVS)
969 C
970 C COMPUTE NEW I,J GRID COORDINATES OF END POINT OF NET 24 HR. MOTI
971 C APPROXIMATE GRID LENGTH IS 90 KM
972 C (CM/SEC)(1KM/100000CM)(3600SEC/HR)(24HRS)(1GRDLEN/90KM)=====
973 C ===== 0.009600(GRDLEN)/(CM/SEC)
974 C TYPICAL FACTOR=0.009600
975 C THUS, IT TAKES A CURRENT SPEED OF 104 CM/SEC
976 C TO GO ONE GRID POINT OF 90 KM IN A DAY
977 C A MORE TYPICAL SPEED OF 15 CM/SEC YIELDS 0.144 GRID LENGTH
978 C THEREFORE ABOUT 1 % OF A GRID POINT PER CM/SEC
979 C
980 C COMPUTE FACTOR AT APPROPRIATE GRID LENGTH AT LATITUDE OF VECTOR
981 C GRID LENGTH VARIES WITH LATITUDE -- 97.6KM AT 66N TO 76.5KM AT 3
982      IGV=GV1(I,J)
983      JGU=GU1(I,J)
984      GLA=ALAT2(IGV,JGU)

```

```

985      GLN=GRIDLN(GLA)
986      FACTOR=0.864000/GLN
987      GU2(I,J)=GU1(I,J)+(UU*FACTOR)
988      GV2(I,J)=GV1(I,J)-(VV*FACTOR)
989      C
990      C HANDLE OFF MAP POINTS
991          IF(GV2(I,J).GE.38.99)GV2(I,J)=38.99
992          IF(GV2(I,J).LE.1.0)GV2(I,J)=1.0
993          IF(GU2(I,J).GE.102.99)GU2(I,J)=102.99
994          IF(GU2(I,J).LE.1.0)GU2(I,J)=1.0
995      C
996      C IF (I.EQ.30.AND.J.EQ.35)WRITE(6,3198)I,J,GU1(I,J),GV1(I,J),GU
997      C ' ),GV2(I,J),U1,V1,UU,VV
998      3198 FORMAT(2I4,8F15.4)
999      C WRITE(6,4104)IDY9,KK,I,J,GV1(I,J),GU1(I,J),GV2(I,J),GU2(I,J)
1000      CALL IJTOLL(GV1(I,J),GU1(I,J),GLA1,GLO1)
1001      CALL IJTOLL(GV2(I,J),GU2(I,J),GLA2,GLO2)
1002      C
1003      100 FORMAT(8F8.3,3I3,A3)
1004      C STORE PROGRESSIVE VECTOR FILES (60)
1005          WRITE(60,4104)IDY9,KK,KYR,IMO,IDY,I,J,GU1(I,J),GV1(I,J),GLA1,
1006          'GU2(I,J),GV2(I,J),GLA2,GLO2,WFAC,TFAC,DDFAC
1007      4104 FORMAT(2I3,3I2,2I3,8F7.3,3F3.1)
1008          GU1(I,J)=GU2(I,J)
1009          GV1(I,J)=GV2(I,J)
1010      2964 CONTINUE
1011      C
1012      C
1013      10000 CONTINUE
1014      909 CONTINUE
1015      C
1016          IF(NOSTOR.LT.1)CLOSE(NFILE,DISP=CRUNCH)
1017          IF(NOSTOR.LT.1)CLOSE(52,DISP=CRUNCH)
1018          IF(NOSTOR.LT.1)CLOSE(53,DISP=CRUNCH)
1019          IF(NOSTOF.LT.1)CLOSE(54,DISP=CRUNCH)
1020          IF(NOSTOR.LT.1)CLOSE(55,DISP=CRUNCH)
1021          CLOSE(60,DISP=CRUNCH)
1022      C
1023      C
1024      800 PRINT 30,IY1,LYR
1025      30 FORMAT(' YEARS NOT RIGHT. FIRST=',I2,', LAST=',I2)
1026      C
1027      901 CONTINUE
1028      END
1029      SUBROUTINE READER(IMO,IYR,IDY,ITYP,C)
1030          DIMENSION LS(40,104),A(40,104)
1031          DIMENSION C(40,104)
1032          NN=20;MM=44
1033          ITYP=0
1034      900 CONTINUE
1035          DO 100 J=1,44
1036              JJ=45-J
1037              READ(2,1100,END=800)IY,MM,NDY,(A(II,JJ),II=1,10)
1038              READ(2,1100,END=800)IY,MM,NDY,(A(II,JJ),II=11,20)
1039      100 CONTINUE

```

```

1040     1100 FORMAT(1X,3I2,8X,10F6.1)
1041     C
1042     WRITE(6,1110) IY,MM,NDY
1043     1110 FORMAT (10X,3I3,' FNWC (20X44) FILE READ')
1044     IYE=IY ; MME=MM ; NDYE=NDY
1045     DO 210 I=1,20
1046     DO 210 J=1,44
1047     II=21-I
1048     C(II,J)=A(I,J)
1049     210 CONTINUE
1050     IMO=MM
1051     IYR=IY
1052     IDY=NDY
1053     GO TO 700
1054     800 PRINT 10,IYE,MME,NDYE
1055     10 FORMAT(* END OF FILE. LAST RECORD FOUND FOR*3I5)
1056     ITYP=999
1057     RETURN
1058     600 ITYP=888
1059     700 RETURN
1060
1061     END
1062     SUBROUTINE BESSEL (XJ,XI,IOK,VAL,DATA)
1063     DIMENSION A(4),AIJ(4)
1064     DIMENSION DATA(40,104)
1065     C
1066     IF(IOK.NE.99) GO TO 300
1067     DO 310 I=1,40
1068     WRITE(6,305) (DATA(I,J),J=1,50,2)
1069     310 CONTINUE
1070     305 FORMAT(25F5.1)
1071     300 CONTINUE
1072     C
1073     IOK=0
1074     C LIMIT CALCULATIONS WITHIN 1 INTERIOR POINT OF LEFT AND BOTTOM
1075     C LIMIT CALCULATIONS WITHIN 2 INTERIOR PTS OF RIGHT AND TOP
1076     C DO 4 POINT BELLEL INTERPOLATION FOR VALUES BETWEEN GRID POINTS
1077     IA=XI
1078     JA=XJ
1079     C TEST IF INDEX WITHIN RANGE
1080     IF (IA.LT.2.OR.JA.LT.2) GO TO 998
1081     IF (IA.GT.18.OR.JA.GT.42) GO TO 998
1082     XINC=XJ-FLOAT(JA)
1083     YINC=XI-FLOAT(IA)
1084     DO 105 K=1,4
1085     JJ=JA+(K-2)
1086     DO 106 L=1,4
1087     II=IA+(L-2)
1088     AIJ(L)=DATA(II,JJ)
1089     IF(XJ.LT.1.9)WRITE(6,450)XI,XJ,L,II,JJ,AIJ(L)
1090     106 CONTINUE
1091     C VERTICAL INTERPOLATION
1092     A(K)=AIJ(2)+YINC*((AIJ(3)-AIJ(2))+((YINC-1.0)/4.0)*((AIJ(4)-A
1093     1)+(AIJ(1)-AIJ(2))))
1094     IF(XJ.LT.1.9)WRITE(6,451)XI,XJ,K,A(K)

```

```

1095     105 CONTINUE
1096     C HORIZONTAL INTERPOLATION
1097         VAL=A(2)+XINC*((A(3)-A(2))+((XINC-1.0)/4.0)*((A(4)-A(3))+
1098         1(A(1)-A(2))))
1099         IF (XJ.LT.1.9)WRITE(6,452)XI,XJ,VAL
1100     450 FORMAT(2F7.3,3I4,F7.3)
1101     451 FORMAT(2F7.3,I4,F7.3)
1102     452 FORMAT(3F7.3)
1103         GO TO 999
1104     998 IOK=1
1105         VAL=999999
1106     999 RETURN
1107         END
1108     SUBROUTINE CONVIJ (XX,YY,IERR,XLEN,YYLEN,M,N,ALAT,ALON)
1109     DIMENSION ALAT(40,104),ALON(40,104)
1110     DIMENSION SLON(40,104)
1111     IF (XX.LT.0.0) XX=-XX
1112     C
1113     C
1114     C GIVEN A RANDOM POINT ON THE GLOBE (LONGITUDE,LATITUDE) DEGREE-DE
1115     C AND A GRID MESH OF DEFINED (LONG,LAT) POINTS
1116     C FIND THE (I,J) COORDINATES OF THE RANDOM POINT (XX,YY)
1117     C RELATIVE TO THE GIVEN GRID MESH
1118     C REJECT POINTS OUTSIDE OF THE GRID
1119     C THIS IS A GENERALIZED SUBPROGRAM TO FIT ANY NEARLY SQUARE
1120     C EQUIDISTANT GRID IN GLOBE (SPHERICAL) SPACE
1121     C BUT
1122     C GRID LINES MAY HAVE A CONSTANT SLOPE OR MAY VARY WITHIN GRID
1123     C IN LONGITUDE-LATITUDE SPACE
1124     C FOR THE GIVEN (NXM) GRID ----- N=ROW#, M=COL# -----
1125     C THIS RESULTS IN THE ABILITY TO MAP SQUARE OR ODD SHAPED GRIDS
1126     C INTO A SQUARE OR RECTANGULAR PLOT TO MATCH THE PRINTED PAGE
1127     C XLEN IS THE DESIRED LENGTH OF THE X-AXIS IN INCHES
1128     C YLEN IS THE DESIRED LENGTH OF THE Y-AXIS IN INCHES
1129     C
1130     C
1131     C
1132     C
1133         N1=N-1
1134         M1=M-1
1135         DXX=XLEN/M1
1136         DYY=YYLEN/N1
1137     C
1138     C REJECT MOST POINTS OUT OF GRID EXTREMES
1139         IERR=0
1140     C
1141     C FIND MAX AND MIN LAT AND LONG OF GRID
1142         AMAXLA=0.0
1143         AMAXLO=0.0
1144         AMINLA=1000.0
1145         AMINLO=1000.0
1146         DO 1201 I=1,N
1147         DO 1202 J=1,M
1148         IF (ALAT(I,J).GT.AMAXLA) AMAXLA=ALAT(I,J)
1149         IF (ALON(I,J).GT.AMAXLO) AMAXLO=ALON(I,J)

```

```

1150         IF (ALAT(I,J).LT.AMINLA) AMINLA=ALAT(I,J)
1151         IF (ALON(I,J).LT.AMINLO) AMINLO=ALON(I,J)
1152     1202 CONTINUE
1153     1201 CONTINUE
1154     C
1155     C
1156     C REJECT POINT IF OUT OF GRID MAX AND MIN LAT OR LONG
1157         IF (YY.GT.AMAXLA) IERR=1
1158         IF (YY.LT.AMINLA) IERR=1
1159         IF (XX.GT.AMAXLO) IERR=1
1160         IF (XX.LT.AMINLO) IERR=1
1161     C JUMP TO ERROR IF POINT IS OUTSIDE OF SQUARE OF BOUNDARY EXTREMES
1162     C AND SET BOTH COORDINATES TO MAXIMUM PLOT INCHES BEFORE RETURN
1163     1240 IF (IERR.EQ.1.AND.XX.GT.XLEN) XX=XLEN
1164         IF (IERR.EQ.1.AND.YY.GT.YYLEN) YY=YYLEN
1165         IF (IERR.EQ.1) GO TO 9999
1166     C
1167     C
1168     C IN CASE THE GRID IS TILTED (ALL OR IN PART)
1169     C LOOK AROUND GRID BOUNDARY LINES TO REJECT EXTERIOR POINTS
1170     C COMPUTE EQUATION OF LINE BETWEEN 2 POINTS
1171     C IF XX IS BETWEEN THE POINTS TEST IF YY INSIDE OR OUTSIDE OF GRID
1172     C COVER THE CASES FOR EACH SIDE OF GRID SLOPING + OR - (LAT=HORIZ)
1173     C
1174     C LOOK AT TOP AND BOTTOM LINES OF GRID
1175         I=1
1176     8002 DO 8001 J=1,M1
1177         X1=ALON(I,J)
1178         X2=ALON(I,J+1)
1179         Y1=ALAT(I,J)
1180         Y2=ALAT(I,J+1)
1181         IF (XX.GT.X1.OR.XX.LT.X2) GO TO 8001
1182         IF (X2.EQ.X1) GO TO 8001
1183         Y0=Y1+((Y2-Y1)*(XX-X1)/(X2-X1))
1184         IF (YY.GT.Y0.AND.I.EQ.1) IERR=1
1185         IF (YY.GT.Y0.AND.I.EQ.1) GO TO 1240
1186         IF (YY.LT.Y0.AND.I.EQ.N) IERR=1
1187         IF (YY.LT.Y0.AND.I.EQ.N) GO TO 1240
1188     8001 CONTINUE
1189         IF (I.EQ.N) GO TO 8025
1190         I=N
1191         GO TO 8002
1192     8025 CONTINUE
1193     C
1194     C LOOK AT LEFT AND RIGHT BOUNDARY LINES
1195         J=1
1196     8007 DO 8006 I=1,N1
1197         X1=ALON(I,J)
1198         X2=ALON(I+1,J)
1199         Y1=ALAT(I,J)
1200         Y2=ALAT(I+1,J)
1201         IF (X2.EQ.X1) GO TO 8006
1202         IF ((X2-X1).LT.0.0) GO TO 8013
1203         IF (XX.LT.X1.OR.XX.GT.X2) GO TO 8006
1204         GO TO 8014

```

```

1205      8013 IF (XX.GT.X1.OR.XX.LT.X2) GO TO 8006
1206      8014 Y0=Y1+((Y2-Y1)*(XX-X1)/(X2-X1))
1207          IF (J.EQ.1.AND.(X2-X1).LT.0.0) GO TO 8011
1208          IF (J.EQ.M.AND.(X2-X1).GT.0.0) GO TO 8011
1209          IF (YY.GT.Y0) IERR=1
1210          IF (YY.GT.Y0) GO TO 1240
1211          GO TO 8006
1212      8011 CONTINUE
1213          IF (YY.LT.Y0) IERR=1
1214          IF (YY.LT.Y0) GO TO 1240
1215      8006 CONTINUE
1216          IF (J.EQ.M)GO TO 8035
1217          J=M
1218          GO TO 8007
1219      8035 CONTINUE
1220      C
1221      C NOW, WE DEFINITELY KNOW (XX,YY) IS INSIDE THE GRID OR ON BOUNDAR
1222      C
1223      C LOOK THROUGH ENTIRE GRID FOR CLOSEST GRID POINT TO POINT(XX,YY)
1224      C SEARCH USING LEAST SQUARE DISTANCE IN LINEAR 2D DEGREE SPACE
1225          DMIN=1000.0
1226          PI=3.141592
1227          XX=-XX
1228          DO 1207 I=1,N
1229          DO 1208 J=1,M
1230          SLON(I,J)=-ALON(I,J)
1231          X2=SLON(I,J)
1232          Y2=ALAT(I,J)
1233          DIST=SQRT(((X2-XX)*(X2-XX))+((Y2-YY)*(Y2-YY)))
1234          IF (DIST.LT.DMIN) NLAT=I
1235          IF (DIST.LT.DMIN) MLON=J
1236          IF (DIST.LT.DMIN) DMIN=DIST
1237      1208 CONTINUE
1238      1207 CONTINUE
1239      C
1240      C NLAT,MLON ARE THE I,J COORDINATES FOR THE NEAREST GRID POINT
1241      C
1242      C SET UP LAT AND LONG OF 4 SURROUNDING GRID POINTS
1243      C
1244      C
1245      C      P2
1246      C      .
1247      C      .
1248      C      .
1249      C P3...P0...P1
1250      C      .
1251      C      .
1252      C      .
1253      C      P4
1254      C
1255      C
1256      C
1257      C COMPUTATIONS ARE NOW DONE IN THIS 4-QUADRANT SYSTEM
1258      C ON BOUNDARY AVOID POINTS OUTSIDE OF GRID (SET THEM = CLOSEST GRI
1259          C=1000.0

```

```

1260      POX=SLON(NLAT,MLON)
1261      POY=ALAT(NLAT,MLON)
1262      P1X=POX; P2X=POX; P3X=POX; P4X=POX
1263      P1Y=POY; P2Y=POY; P3Y=POY; P4Y=POY
1264      IF (MLON.LT.M) P1X=SLON(NLAT,MLON+1)
1265      IF (MLON.LT.M) P1Y=ALAT(NLAT,MLON+1)
1266      IF (NLAT.GT.1) P2X=SLON(NLAT-1,MLON)
1267      IF (NLAT.GT.1) P2Y=ALAT(NLAT-1,MLON)
1268      IF (MLON.GT.1) P3X=SLON(NLAT,MLON-1)
1269      IF (MLON.GT.1) P3Y=ALAT(NLAT,MLON-1)
1270      IF (NLAT.LT.N) P4X=SLON(NLAT+1,MLON)
1271      IF (NLAT.LT.N) P4Y=ALAT(NLAT+1,MLON)
1272      C SPHERICAL GEOMETRY OF THE GLOBE
1273      C DICTATES THAT CONVERGING LONGITUDES MUST BE
1274      C MULTIPLIED BY COS(LAT) TO PRESERVE EQUIVALENT COORDINATE SCA
1275      XC=COS(ALAT(NLAT,MLON)*PI/180.0)
1276      DX0=(XX-POX)*XC
1277      DX1=(P1X-POX)*XC
1278      DX2=(P2X-POX)*XC
1279      DX3=(P3X-POX)*XC
1280      DX4=(P4X-POX)*XC
1281      DY0=YY-POY
1282      DY1=P1Y-POY
1283      DY2=P2Y-POY
1284      DY3=P3Y-POY
1285      DY4=P4Y-POY
1286      S0=DX0*DY0
1287      S1=DX1*DY1
1288      S2=DX2*DY2
1289      S3=DX3*DY3
1290      S4=DX4*DY4
1291      C
1292      C COMPUTE ANGLE (AP) BETWEEN LINE (XX,YY)-(MLON,NLAT) AND THE +X A
1293      C AND AVOID INFINITE TANGENTS
1294      AP=0.0; AE=0.0; AN=0.0; AW=0.0; AS=0.0
1295      IF (ABS(DX0).LE.(1.0/C)) AP=ATAN(SIGN(C,S0))
1296      IF (ABS(DX0).GT.(1.0/C)) AP=ATAN((DY0)/DX0)
1297      CALL RADCOR(DX0,DY0,ADD)
1298      AP=AP+ADD
1299      C
1300      C COMPUTE THE ANGLE EACH OF THE 4 COORDINATES MAKES WITH THE +X AX
1301      C AE IS THE ANGLE THE EAST LINE (P1-P0) MAKES WITH THE +X AXIS
1302      IF (ABS(DX1).LE.(1.0/C)) AE=ATAN(SIGN(C,S1))
1303      IF (ABS(DX1).GT.(1.0/C)) AE=ATAN((DY1)/DX1)
1304      CALL RADCOR(DX1,DY1,ADD)
1305      AE=AE+ADD
1306      C AN IS THE ANGLE THE NORTH LINE (P2-P0) MAKES WITH THE +X AXIS
1307      IF (ABS(DX2).LE.(1.0/C)) AN=ATAN(SIGN(C,S2))
1308      IF (ABS(DX2).GT.(1.0/C)) AN=ATAN((DY2)/DX2)
1309      CALL RADCOR(DX2,DY2,ADD)
1310      AN=AN+ADD
1311      C AW IS THE ANGLE THE WEST LINE (P3-P0) MAKES WITH THE +X AXIS
1312      IF (ABS(DX3).LE.(1.0/C)) AW=ATAN(SIGN(C,S3))
1313      IF (ABS(DX3).GT.(1.0/C)) AW=ATAN((DY3)/DX3)
1314      CALL RADCOR(DX3,DY3,ADD)

```

```

1315     AW=AW+ADD
1316   C AS IS THE ANGLE THE SOUTH LINE (P4-P0) MAKES WITH THE +X AXIS
1317     IF (ABS(DX4).LE.(1.0/C)) AS=ATAN(SIGN(C,S4))
1318     IF (ABS(DX4).GT.(1.0/C)) AS=ATAN((DY4)/DX4)
1319     CALL RADCOR(DX4,DY4,ADD)
1320     AS=AS+ADD
1321   C
1322   C COMPUTE LENGTH TO POINT(XX,YY)
1323     D0=SQRT((DX0*DX0)+(DY0*DY0))
1324     D1=SQRT((DX1*DX1)+(DY1*DY1))
1325     D2=SQRT((DX2*DX2)+(DY2*DY2))
1326     D3=SQRT((DX3*DX3)+(DY3*DY3))
1327     D4=SQRT((DX4*DX4)+(DY4*DY4))
1328   C
1329   C COMPUTE (ANGLE) THE ANGLE FROM +X AXIS OF ROTATED GRID
1330   C AE IS THE BASE LINE USED FOR COMPUTING + ANGLES BECAUSE
1331   C IT CORRESPONDS TO THE +X AXIS OF THE ROTATED GRIDS
1332   C WARNING: THE TRIGONOMETRY ASSUMES RIGHT ANGLES WITHIN THE GRID-
1333   C           BUT TESTS INDICATE THIS ROUTINE PERFORMES WELL
1334   C           FOR RELATIVELY SMALL DEVIATIONS (RT. ANGLE +- 10%)
1335     IF (MLON.EQ.M.AND.AW.LT.PI) AE=AW+PI
1336     IF (AP.LT.AE) ANGLE=(2.0*PI)+AP-AE
1337     IF (AP.GE.AE) ANGLE=AP-AE
1338   C
1339   C COMPUTE FRACTIONAL I,J'S AWAY FROM GRID POINT
1340     IF (D0.EQ.0.0) XFRACT=0.0
1341     IF (D0.EQ.0.0) YFRACT=0.0
1342     IF (D0.EQ.0.0) GO TO 4198
1343     DD=D1
1344     IF (D1.EQ.0.0) DD=D3
1345     IF (DD.EQ.0.0) XFRACT=0.0
1346     IF (DD.EQ.0.0) GO TO 603
1347     XFRACT=(D0/DD)*COS(ANGLE)
1348   603 DD=D2
1349     IF (D2.EQ.0.0) DD=D4
1350     IF (DD.EQ.0.0) YFRACT=0.0
1351     IF (DD.EQ.0.0) GO TO 4198
1352     YFRACT=-((D0/DD)*SIN(ANGLE))
1353   4198 XXX=XFRACT + FLOAT(MLON)
1354   4199 YYY=YFRACT + FLOAT(NLAT)
1355     IF (XXX.LT.1.0) XXX=1.0
1356     IF (YYY.LT.1.0) YYY=1.0
1357     IF (XXX.GT.M) XXX=M
1358     IF (YYY.GT.N) YYY=N
1359   C
1360   C CONVERT FROM I,J GRID VALUES TO I,J PLOT INCHES
1361   C   XX=XLEN*((XXX-1.0)/M1)
1362   C   YY=YYLEN-(YYLEN*((YYY-1.0)/N1))
1363     XX=XXX
1364     YY=YYY
1365   C
1366   9999 CONTINUE
1367     RETURN
1368   C
1369   C

```



```

1370 C
1371 END
1372 SUBROUTINE SMTH(N,ALPHA,FL,FL2)
1373 DIMENSION TDEP(40,104),TML(40,104),FL(40,104),XF(40,104),
1374 1TCX(40,104),TCY(40,104),XTR2(40,104),XTR(40,104),TS(40,104),
1375 2SLP(40,104),WCX(40,104),WCY(40,104),LS2(40,104)
1376 DIMENSION FL2(40,104)
1377 BETA=1.0-ALPHA
1378 DO 2 I=1,40
1379 DO 2 J=1,104
1380 2 XF(I,J)=FL(I,J)
1381 DO 40 M=1,N
1382 10 DO 20 I=2,39
1383 DO 20 J=2,103
1384 XY=FL(I,J)
1385 A=FL(I,J+1)
1386 B=FL(I+1,J)
1387 C=FL(I,J-1)
1388 D=FL(I-1,J)
1389 IF (A.EQ.0.OR.B.EQ.0) GO TO 20
1390 IF (C.EQ.0.OR.D.EQ.0) GO TO 20
1391 XF(I,J) = (ALPHA*XY) + (BETA*((A+B+C+D)/4.0))
1392 20 CONTINUE
1393 DO 30 I=2,39
1394 DO 30 J=2,49
1395 FL(I,J)=XF(I,J)
1396 30 CONTINUE
1397 40 CONTINUE
1398 RETURN
1399 END
1400 SUBROUTINE SPLINE (X,Y,IOK,VAL,DATA)
1401 DIMENSION A(4),AIJ(4)
1402 DIMENSION DATA(40,104)
1403 C
1404 C
1405 IOK=0
1406 C DO 4 PT NATURAL SPLINE INTERPOLATION FOR VALUES BETWEEN GRID PTS
1407 IA=Y
1408 JA=X
1409 C LIMIT CALCULATIONS WITHIN 1 INTERIOR POINT OF LEFT AND BOTTOM
1410 C LIMIT CALCULATIONS WITHIN 2 INTERIOR PTS OF RIGHT AND TOP
1411 IF (IA.LT.2.OR.JA.LT.2) GO TO 998
1412 C IF (IA.GT.18.OR.JA.GT.42) GO TO 998
1413 IF (IA.GT.38.OR.JA.GT.102) GO TO 998
1414 DX=1.0
1415 X1=JA
1416 Y1=IA
1417 XP1=X1+1.0
1418 YP1=Y1+1.0
1419 DO 105 K=1,4
1420 JJ=JA+(K-2)
1421 DO 106 L=1,4
1422 II=IA+(L-2)
1423 AIJ(L)=DATA(II,JJ)
1424 C IF(XJ.LT.1.9)WRITE(6,450)XI,XJ,L,II,JJ,AIJ(L)

```

```

1425     106 CONTINUE
1426   C  VERTICAL INTERPOLATION
1427     GPP=(AIJ(3)-(2.0*AIJ(2))+AIJ(1))/(DX*DX)
1428     GPPP1=(AIJ(4)-(2.0*AIJ(3))+AIJ(2))/(DX*DX)
1429     G1=((GPP/6.0)*(((YP1-Y)**(3.0))/DX)-(DX*(YP1-Y))))
1430     G2=((GPPP1/6.0)*(((Y-Y1)**(3.0))/DX)-(DX*(Y-Y1))))
1431     G3=(AIJ(2)*((YP1-Y)/DX))
1432     G4=(AIJ(3)*((Y-Y1)/DX))
1433     G=G1+G2+G3+G4
1434   C  IF(G.LT.20.0)WRITE (66,9412) N,M,K,NK,GPP,GPPP1,G1,G2,G3,G4,G
1435     A(K)=G
1436     105 CONTINUE
1437   C  HORIZONTAL INTERPOLATION
1438     GPP=(A(3)-(2.0*A(2))+A(1))/(DX*DX)
1439     GPPP1=(A(4)-(2.0*A(3))+A(2))/(DX*DX)
1440     G1=((GPP/6.0)*(((XP1-X)**(3.0))/DX)-(DX*(XP1-X))))
1441     G2=((GPPP1/6.0)*(((X-X1)**(3.0))/DX)-(DX*(X-X1))))
1442     G3=(A(2)*((XP1-X)/DX))
1443     G4=(A(3)*((X-X1)/DX))
1444     G=G1+G2+G3+G4
1445     VAL=G
1446     450 FORMAT(2F7.3,3I4,F7.3)
1447     451 FORMAT(2F7.3,I4,F7.3)
1448     452 FORMAT(3F7.3)
1449     GO TO 999
1450     998 IOK=1
1451     VAL=999999
1452     999 RETURN
1453     END
1454     SUBROUTINE PRINT1(P,NOX)
1455     DIMENSION P(40,104),IQ(64),PQ(40,104)
1456     199 FORMAT(1H1,5X,'LAND-SEA-GRID')
1457     200 FORMAT(1H1,5X,'TEMPERATURE AT 200 M(CX10)')
1458     201 FORMAT(1H1,5X,'DYNAMIC TOPOGRAPHY (DYNMX100)')
1459     202 FORMAT(1H1,5X,'SURFACE PRESSURE(MBX10)')
1460     203 FORMAT(1H1,5X,'SURFACE TEMPERATURE(CX10)')
1461     204 FORMAT(1H1,5X,'MODEL OUTPUT (CM/SEC) ')
1462     205 FORMAT(1H1,5X,'WEIGHTED AVERAGE OF TEMP 0-200M (C) X10.0')
1463     206 FORMAT (1H1,5X,'SURFACE WIND SPEED (M/SEC)')
1464     209 FORMAT(1H1,5X,'U -- COMPONENT OF WIND SPEED (M/SEC)')
1465     210 FORMAT(1H1,5X,'V -- COMPONENT OF WIND SPEED (M/SEC)')
1466     211 FORMAT(1H1,5X,'SURFACE OCEAN CURRENT SPEED (CM/SEC)')
1467     220 FORMAT (I3,32I4)
1468     212 FORMAT(1H1,5X,'U--DYNAMIC OCEAN CURRENT (CM/SEC)')
1469     213 FORMAT(1H1,5X,'V--DYNAMIC OCEAN CURRENT (CM/SEC)')
1470     KR=6
1471   C  DO 500 IP=1,2
1472     GO TO (10,20,30,40,42,44,70,80,90,100,110,120,130)NOX
1473     10  WRITE(KR,200)
1474     DO 11 I=1,40      ; DO 11 J=1,104
1475     11  PQ(I,J)=P(I,J)*10.0
1476     GO TO 50
1477     20  WRITE(KR,201)
1478     DO 21 I=1,40      ; DO 21 J=1,104
1479     21  PQ(I,J)=P(I,J)*100.0

```

```
1480      GO TO 50
1481 30  WRITE(KR,202)
1482      DO 31 I=1,40      ; DO 31 J=1,104
1483 31  PQ(I,J)=(P(I,J)-1000)*10.0
1484      GO TO 50
1485 40  WRITE(KR,203)
1486      DO 41 I=1,40      ; DO 41 J=1,104
1487 41  PQ(I,J)=P(I,J)*10.0
1488      GO TO 50
1489 42  WRITE(KR,204)
1490      DO 43 I=1,40
1491      DO 43 J=1,104
1492 43  PQ(I,J)=P(I,J)
1493      GO TO 50
1494 44  WRITE(KR,205)
1495      DO 45 I=1,40
1496      DO 45 J=1,104
1497 45  PQ(I,J)=(P(I,J))*10.0
1498      GO TO 50
1499 70  WRITE (KR,199)
1500      DO 71 I=1,40
1501      DO 71 J=1,104
1502 71  PQ(I,J)=P(I,J)
1503      GO TO 50
1504 80  WRITE (KR,206)
1505      DO 81 I=1,40
1506      DO 81 J=1,104
1507 81  PQ(I,J)=P(I,J)
1508      GO TO 50
1509 90  WRITE(KR,209)
1510      DO 91 I=1,40
1511      DO 91 J=1,104
1512 91  PQ(I,J)=P(I,J)
1513      GO TO 50
1514 100 WRITE(KR,210)
1515      DO 101 I=1,40
1516      DO 101 J=1,104
1517 101 PQ(I,J)=P(I,J)
1518      GO TO 50
1519 110 WRITE(KR,211)
1520      DO 111 I=1,40
1521      DO 111 J=1,104
1522 111 PQ(I,J)=P(I,J)
1523      GO TO 50
1524 120 WRITE (KR,212)
1525      DO 121 I=1,40
1526      DO 121 J=1,104
1527 121 PQ(I,J)=P(I,J)
1528      GO TO 50
1529 130 WRITE (KR,213)
1530      DO 131 I=1,40
1531      DO 131 J=1,104
1532 131 PQ(I,J)=P(I,J)
1533      C
1534      C
```

```

1535     50 CONTINUE
1536     C
1537     C PRINT LEFT 32 COLS. THEN PRINT RIGHT 32 COLS.
1538         DO 58 KJ=1,2
1539             J2=KJ*32
1540             J1=J2-31
1541             IF(NOX.EQ.1.AND.KJ.EQ.2) WRITE(KR,200)
1542             IF(NOX.EQ.2.AND.KJ.EQ.2) WRITE(KR,201)
1543             IF(NOX.EQ.6.AND.KJ.EQ.2) WRITE(KR,205)
1544             IF(NOX.EQ.4.AND.KJ.EQ.2) WRITE(KR,203)
1545             IF(NOX.EQ.3.AND.KJ.EQ.2) WRITE(KR,202)
1546             IF(NOX.EQ.8.AND.KJ.EQ.2) WRITE(KR,206)
1547             IF(NOX.EQ.9.AND.KJ.EQ.2) WRITE(KR,209)
1548             IF(NOX.EQ.10.AND.KJ.EQ.2) WRITE(KR,210)
1549             IF(NOX.EQ.11.AND.KJ.EQ.2) WRITE(KR,211)
1550             IF(NOX.EQ.12.AND.KJ.EQ.2) WRITE(KR,212)
1551             IF(NOX.EQ.13.AND.KJ.EQ.2) WRITE(KR,213)
1552         DO 60 I=1,40
1553             DO 57 JJ=J1,J2
1554                 57 IQ(JJ)=PQ(I,JJ)
1555                 WRITE (KR,220) I,(IQ(J),J=J1,J2)
1556             60 CONTINUE
1557             58 CONTINUE
1558         500 CONTINUE
1559         RETURN
1560         END
1561         SUBROUTINE OUTPT(OUT,NFILE,KYR,MO,KDY,H)
1562         DIMENSION OUT(40,104),H(1)
1563         DO 60 J=1,104
1564             DO 60 I=1,5
1565                 I2=I*8
1566                 I1=I2-7
1567                 WRITE (NFILE,100) (OUT(IK,J),IK=I1,I2),KYR,MO,KDY,H
1568             60 CONTINUE
1569         100 FORMAT (8F8.2,3I3,A3)
1570         RETURN
1571         END
1572         SUBROUTINE WIND(SLP5,LS2,BLAT,WX,WY)
1573         DIMENSION SLP5(40,104),BLAT(40,104),WX(40,104),WY(40,104)
1574         DIMENSION LS2(40,104)
1575         100 FORMAT(1X,18F8.3)
1576         ALPHA=0.75 ; BETA=1.0
1577         OMEGAV=0.80
1578     C COMPUTE SURFACE GEOSTROPHIC WIND IN M/SEC
1579     C WIND(M./SEC) = (DP/DL)/(RHO)(CORIOLIS)
1580     C 1 BAR = 10(6TH) DYNES/CM**2
1581     C WEIGHT IN KG = MG = (KG)(9.8M/SEC**2)
1582     C 1 MBAR = 10.1971 KG/M**2 = 100.0 KG M-1 SEC-2
1583     C CW = (100)/(90000)(1.22)(10-4TH) = 9.11
1584     C 1XGRIDLENGTH IS APPROX. 90000 M
1585     C TYPICAL WIND SPEED COMPUTATION IS FOR 1MB GRADIENT PER GRID LENG
1586     C W(M/S)=(1 MB)(100 KG M-1 SEC-2 MB-1)/(90000 M)(1.22 KG M-3)(0.00
1587     C                                     58 S-1)(SIN LAT)
1588     C (1)(100)/(90000)(1.22)(0.0001458)(.707) = 8.84 M/S
1589     C

```

```

1590 C           INCLUDES REDUCTION AND ROTATION OF WIND
1591 C
1592           PI=3.1415962
1593           THETA=-15.0*PI/180.0 ; RR=0.7 ; A1=RR*COS(THETA)
1594           B2=A1 ; B1=RR*SIN(THETA) ; A2=-B1
1595           DO 20 I=2,39
1596           DO 20 J=2,103
1597 C COMPUTE PURE GEOSTROPHIC WIND (WITH NO FRICTION)
1598           GLAT=BLAT(I,J)
1599           GG=GRIDLN(GLAT)
1600           XR=SIN(BLAT(I,J)/57.296)
1601           CRLS=0.0001458*XR
1602 C KM TO METERS
1603           GG=GG*1000.0
1604           CW=100.0/(GG*1.22*CRLS)
1605 C           IF(LS2(I,J).EQ.0.OR.LS2(I+1,J).EQ.0.OR.LS2(I,J+1).EQ.0) GO T
1606 C           IF (LS2(I,J).EQ.0) GO TO 21
1607 C USE 5-POINT GRADIENT IN INTERIOR OF GRID
1608 C USE 3-POINT GRADIENT ONE GRID POINT FROM BOUNDARY
1609           GRADX=SLP5(I,J+1)-SLP5(I,J-1)
1610           GRADX=GRADX/2.0
1611           IF (J.EQ.2.OR.J.EQ.103) GO TO 41
1612           GRADX=SLP5(I,J-2)-(8.0*SLP5(I,J-1))+(8.0*SLP5(I,J+1))-SLP5(I,
1613           GRADX=GRADX/12.0
1614           41 CONTINUE
1615           GRADY=SLP5(I-1,J)-SLP5(I+1,J)
1616           GRADY=GRADY/2.0
1617           IF (I.EQ.2.OR.I.EQ.39) GO TO 42
1618           GRADY=SLP5(I-2,J)-(8.0*SLP5(I-1,J))+(8.0*SLP5(I+1,J))-SLP5(I+
1619           GRADY=-GRADY/12.0
1620           42 CONTINUE
1621           WU=-GRADY*CW
1622           WV=GRADX*CW
1623           GLATR=GLAT/57.2958
1624           SINLAT=SIN(GLATR)
1625 C COMPUTE MANGITUDE REDUCTION (H U ROLL METHOD)
1626           IF(GLAT.GT.45.0) RED=0.75+(0.1*((90.0-GLAT)/45.0))
1627           IF(GLAT.LE.45.0.AND.GLAT.GE.25.0) RED=0.85
1628           IF(GLAT.LT.25.0) RED=0.65+(0.2*(GLAT/25.0))
1629 C COMPUTE DEFLECTION ANGLE AS A FUNCTION OF WIND SPEED
1630 C USE REDUCED COMPONENTS IN FIGURING THE ANGLE OF DEFLECTION
1631           XCOMP=RED*WU
1632           YCOMP=RED*WV
1633           SPEED=SQRT((XCOMP*XCOMP)+(YCOMP*YCOMP))
1634           ANG=(22.5-(0.0175*SPEED*SPEED))*(1.495/(SINLAT+1.0))
1635           THETA=-ANG*PI/180.0
1636           A1=RED*COS(THETA)
1637           B2=A1
1638           B1=RED*SIN(THETA)
1639           A2=-B1
1640           WX(I,J)=WU*A1 + WV*B1
1641           WY(I,J)=WV*B2 + WU*A2
1642           GO TO 20
1643           21 WX(I,J)=0.0 ; WY(I,J)=0.0
1644           20 CONTINUE

```

```

1645      DO 22 M=1,40
1646      WX(M,104)=WX(M,103)
1647      WY(M,104)=WY(M,103)
1648      WX(M,1)=WX(M,2)
1649      WY(M,1)=WY(M,2)
1650      22 CONTINUE
1651      DO 24 N=1,104
1652      WX(40,N)=WX(39,N)
1653      WY(40,N)=WY(39,N)
1654      WX(1,N)=WX(2,N)
1655      WY(1,N)=WY(2,N)
1656      24 CONTINUE
1657      C WRITE WIND SPEED
1658      C CALL PRINT1(SLP5,3)
1659      C CALL PRINT1(WX,9)
1660      C CALL PRINT1(WY,10)
1661      395 CONTINUE
1662      RETURN
1663      END
1664      FUNCTION WINDC(W1,W2,M,YLAT)
1665      C
1666      C CURRENT (CM/SEC) = 4.8X(SQUARE ROOT WIND SPEED M/SEC)
1667      C
1668      C
1669      C FACTOR IS BETWEEN 4 AND 5 IF MASS TRANSPORT BY WAVES IS INCLUDED
1670      XK=4.8
1671      C
1672      C
1673      SPDSQD=W1**2 + W2**2
1674      IF(SPDSQD-5.0E-04)10,10,20
1675      10 WINDC=0.0
1676      RETURN
1677      20 SPEED=SQRT(SPDSQD)
1678      SQROOT=SQRT(SPEED)
1679      SP40=40.0*0.5148
1680      CL=(34.0-((67.0-YLAT)*0.2933))
1681      CLV=CL/SQRT(SP40)
1682      ANGLD=CL-(CLV*SQRT(SPEED))
1683      B1=COS(ANGLD/57.296)
1684      B2=SIN(ANGLD/57.296)
1685      W10=XK*(W1/SQROOT)
1686      W20=XK*(W2/SQROOT)
1687      IF(M.EQ.2) W20=-W20
1688      WINDC=W10*B1 + W20*B2
1689      C IF (M.EQ.5) WRITE(6,88)W1,W2,A2,W10,W20,D2,B1,B2,WINDC
1690      88 FORMAT (1H ,9F12.5)
1691      RETURN
1692      END
1693      SUBROUTINE OCEANC(TML,TCX,TCY,HH)
1694      COMMON/LAT/ALAT2,LS2
1695      DIMENSION TDEP(40,104),TML(40,104),FL(40,104),XF(40,104),
1696      1TCX(40,104),TCY(40,104),XTR2(40,104),XTR(40,104),TS(40,104),
1697      2SLP(40,104),WCX(40,104),WCY(40,104),LS2(40,104)
1698      DIMENSION HH(1),ALAT2(40,104)
1699      GAMMA=1.025 ; T=0.0

```

```

1700      DO 70 I=1,39
1701      DO 70 J=1,103
1702      C GRIDSIZE VARIES WITH LATITUDE 83-96 KM (SEE GRIDLN FUNCTION)
1703      XLEN=GRIDLN(ALAT2(I,J))
1704      C CHANGE KM TO M
1705      XLEN=XLEN*1000.0
1706      C
1707      XOR=SIN(ALAT2(I,J))/57.296)
1708      COR=0.0001458*XOR
1709      NP1=LS2(I,J) ; NP2=LS2(I,J+1) ; NP3=LS2(I+1,J)
1710      C ZERO VALUE IF ANY OF THE LAND-SEA GRID POINTS ARE ZERO
1711      IF (NP1.LT.2) GO TO 65
1712      DDI=TML(I,J+1)-TML(I,J)
1713      DDJ=TML(I+1,J)-TML(I,J)
1714      19 IF(HH(1).EQ.3HDYC) GO TO 20
1715      GO TO 10
1716      65 TCX(I,J)=0.0 ; TCY(I,J)=0.0 ; GO TO 70
1717      10 CONTINUE
1718      550 FORMAT (1H ,215,7F15.7)
1719      TCX(I,J)=THERM(DDI,DDJ,XLEN,COR,TML(I,J),T)
1720      TCY(I,J)=T
1721      C WRITE (6,550) I,J,TML(I,J),DDI,DDJ,COR,XLEN,TCX(I,J),TCY(I,J)
1722      GO TO 70
1723      20 TCX(I,J)=DYCUR(DDI,DDJ,XLEN,COR,T)
1724      TCY(I,J)=T
1725      70 CONTINUE
1726      DO 75 M=1,40
1727      TCX(M,104)=TCX(M,103)
1728      TCY(M,104)=TCY(M,103)
1729      75 CONTINUE
1730      DO 77 M=1,104
1731      TCX(40,M)=TCX(39,M)
1732      TCY(40,M)=TCY(39,M)
1733      77 CONTINUE
1734      RETURN
1735      END
1736      FUNCTION THERM(DDI,DDJ,XLN,COR,TML,T)
1737      C
1738      C THWIND = DELTAT(G)(DZ)/(F)(TBAR)
1739      C
1740      C TYPICAL THERMAL WIND CALCULATION
1741      C C1 = G(DZ)=(9.8M/SEC**2)(200M)=1961.2 M**2/SEC**2
1742      C C2 = (0.0001458)(10+273)(SIN 45)(82000) = 2392.45 M SEC-1 DEGK
1743      C ONE DEGREE GRADIENT GIVES TYPICAL VALUE OF CURRENT = 0.8197 M/SE
1744      C
1745      C
1746      C      TIMES 0.0544 = 4.46 CM/SEC
1747      C
1748      ALPHA=100.0
1749      C1=1961.2
1750      C2=TML*COR*XLN
1751      T=DDI*C1/C2
1752      THERM=DDJ*C1/C2
1753      T=T*ALPHA ; THERM=THERM*ALPHA
1754      C

```

```

1755 C BEST FIT CORRELATION
1756 C CONVERTS TO DYNAMIC HEIGHT CALCULATION WITH SALINITY CONSTANT
1757 T=T*0.0544
1758 THERM=THERM*0.0544
1759 C
1760 RETURN
1761 END
1762 FUNCTION DYCUR(DDI,DDJ,XLN,COR,T)
1763 C TYPICAL CALCULATION
1764 C  $10(0.1M^{**2}SEC^{**2})$  DIVIDED BY  $(0.0001458SEC^{**1})(SIN45)(16400$ 
1765 C  $= 0.0873$  M/SEC
1766 C ALPHA CONVERTS TO 8.73 CM/SEC
1767 ALPHA=100.0
1768 C2=XLN*COR
1769 T+=ALPHA*10.0*DDI/C2
1770 DYCUR+=ALPHA*10.0*DDJ/C2
1771 RETURN
1772 END
1773 FUNCTION GRIDLN(GLAT)
1774 C GRID LENGTH AS A FUNCTION OF GEOGRAPHIC LATITUDE FOR 40X104 GRID
1775 AKM=(1.0+(SIN(GLAT*3.14159/180.0)))/1.86603
1776 GRIDLN=AKM*95.23
1777 RETURN
1778 END
1779 FUNCTION GDFNOC(GLAT)
1780 C GRID LENGTH AS A FUNCTION OF GEOGRAPHIC LATITUDE FOR FNOC 20X44
1781 AKM=(1.0+(SIN(GLAT*3.14159/180.0)))/1.86603
1782 GDFNOC=AKM*380.8
1783 RETURN
1784 END
1785 SUBROUTINE DELTAD
1786 COMMON/LAT/ALAT2,LS2
1787 DIMENSION TDEP(40,104),TML(40,104),FL(40,104),XF(40,104),
1788 1TCX(40,104),TCY(40,104),XTR2(40,104),XTR(40,104),TS(40,104),
1789 2SLP(40,104),WCX(40,104),WCY(40,104),IGD(40,104)
1790 DIMENSION HUDD(1),HVDD(1)
1791 DATA HUDD/'UDD '/
1792 DATA HVDD/'VDD '/
1793 C*****INPUT THE DYNAMIC TOPOGRAPHY*****
1794 C LTM
1795 KYR=99
1796 MO=99
1797 KDY=99
1798 C READ IN WESTERN PART OF NORTH PACIFIC GRID
1799 DO 10 J=1,64
1800 DO 10 I=1,5
1801 I2=I*8 ; I1=I2-7
1802 READ(34,111) (TML(IK,J),IK=I1,I2),KYR,MO,KDY,H
1803 111 FORMAT (8F8.3,3I3,A3)
1804 10 CONTINUE
1805 C READ IN EASTERN PART OF NORTH PACIFIC GRID
1806 DO 12 J=41,104
1807 DO 12 I=1,5
1808 I2=I*8 ; I1=I2-7
1809 READ(33,111) (TML(IK,J),IK=I1,I2),KYR,MO,KDY,H

```



```

1810 12 CONTINUE
1811 C
1812 C
1813 C ADD MISSING S.W. CORNER TO MAP OF DD
1814 DO 20 I=27,40
1815 B=I-26
1816 FI=8/13.0
1817 DO 21 J=1,16
1818 C=16-J
1819 FJ=C/15.0
1820 IF (TML(I,J).LT.0.01) TML(I,J)=2.7+((FI+FJ)/10.0)
1821 21 CONTINUE
1822 20 CONTINUE
1823 C
1824 C PRINT DYNAMIC HEIGHTS
1825 WRITE(6,305)
1826 305 FORMAT('1 DYNAMIC HEIGHTS'//)
1827 CALL PRINT2(TML,10.0)
1828 C
1829 C
1830 C
1831 C PRINT DECIMAL DYN HTS FOR ALASKAN STREAM TUNING
1832 WRITE (6,90)
1833 90 FORMAT ('1DYNAMIC HEIGHTS'//)
1834 DO 91 J=1,104
1835 J2=105-J
1836 WRITE (6,92) J,J2,(TML(I,J2),I=1,20)
1837 91 CONTINUE
1838 92 FORMAT (2I4,20F6.3)
1839 C
1840 C
1841 C CALL PRINT1(TML,2)
1842 C*****
1843 C CALL SMTH(2,0.98,TML)
1844 C CALL PRINT1(TML,2)
1845 CALL OCEANC(TML,TCX,TCY,3HDYC)
1846 C CALL PRINT1(TCX,12)
1847 C CALL PRINT1(TCY,13)
1848 C CALL SMTH(3,0.88,TCY)
1849 C CALL SMTH(3,0.88,TCX)
1850 C CALL PRINT1(TCX,5)
1851 C CALL PRINT1(TCY,5)
1852 C
1853 C PRINT GEOSTROPHIC CURRENTS
1854 WRITE(6,300)
1855 300 FORMAT('1 U--COMPONENT OF GEOSTROPHIC SURFACE CURRENT (CM/SEC)
1856 WRITE(6,303)
1857 303 FORMAT (10X,'LONG--TERM MEAN'/)
1858 CALL PRINT2(TCX,1.0)
1859 WRITE(6,301)
1860 301 FORMAT('1V COMPONENT OF GEOSTROPHIC SURFACE CURRENT (CM/SEC)
1861 CALL PRINT2(TCY,1.0)
1862 C
1863 CALL OUTPT(TCX,38,KYR,MO,KDY,HUDD)
1864 CALL OUTPT(TCY,38,KYR,MO,KDY,HVDD)

```

```

1865 C   CLOSE(38,DISP=CRUNCH)
1866     RETURN
1867     END
1868     SUBROUTINE PRINT2(P,F)
1869     DIMENSION P(40,104),IP(40,104),I1(40)
1870     WRITE(6,302)
1871     DO 1 I=1,40
1872       I1(I)=I
1873     1 CONTINUE
1874     WRITE(6,301) (I1(K),K=1,40)
1875     WRITE(6,302)
1876     302 FORMAT(/)
1877     DO 200 I=1,40
1878     DO 201 JB=1,104
1879       J=105-JB
1880       IP(I,J)=P(I,JB)*F
1881     201 CONTINUE
1882     200 CONTINUE
1883     DO 300 J=1,104
1884     WRITE(6,301) (IP(K,J),K=1,40),J
1885     300 CONTINUE
1886     301 FORMAT (1X,40I3,I6)
1887     RETURN
1888     END
1889     SUBROUTINE CSTROT(KYRS,MO5,KDY5,U,V,NOSTOF)
1890     COMMON ANX(40,104),ANY(40,104),BB(40,104),LS3(40,104),
1891     'GDIST(40,104)
1892     DIMENSION GDIST1(40,104),GRDANG(40,104),U(40,104),V(40,104)
1893     DIMENSION U2S(40,104),V2S(40,104)
1894     C
1895     C TEST TANGENT AND ARCTANGENT COMPUTATIONS
1896     PI=3.141596
1897     RAD=PI/180.0
1898     ANG1=-2.0*PI
1899     DANG=10.0*RAD
1900     1 TAN1=TAN(ANG1)
1901     ATAN1=ATAN(TAN1)
1902     ANG2=ATAN1/RAD
1903     C WRITE(6,106) ANG1,DANG,ANG2,TAN1,ATAN1
1904     106 FORMAT(5F11.5)
1905     ANG1=ANG1+DANG
1906     IF (ANG1.LE.(2.0*PI)) GO TO 1
1907     C
1908     C
1909     C ONSHORE COMPONENT REDUCTION FACTOR
1910     C IS PROPORTIONAL TO 1 OVER E TO THE DISTANCE FROM COAST
1911     C NOTE: E**-0=1, E**-1=0.37, E**-2=0.14, E**-3=0.05
1912     C THE DISTANCE UNITS ARE IN GRID LENGTHS
1913     C
1914     C
1915     DO 10 I=2,39
1916     DO 11 J=2,103
1917     U2S(I,J)=U(I,J)
1918     V2S(I,J)=V(I,J)
1919     IF (LS3(I,J).GE.0) GO TO 11

```

```

1920     IF(U(I,J).EQ.0.0.AND.V(I,J).EQ.0.0) GO TO 11
1921     CALL ANGLD(U(I,J),V(I,J),CURANG)
1922     CALL ANGLD(ANX(I,J),ANY(I,J),CSTANG)
1923     DELANG=CURANG-CSTANG
1924     IF (DELANG.GE.270.0) DELANG=360.0-DELANG
1925     IF (DELANG.LE.-270.0) DELANG=360.0+DELANG
1926 C     WRITE(6,109)I,J,U(I,J),V(I,J),CURANG,CSTANG,DELANG
1927     109 FORMAT(2I5,5F15.6)
1928 C
1929 C
1930 C ROTATE FOR DELANG BETWEEN -90 AND +90 DEGREES FROM THE COAST
1931 C DELANG IS THE ANGLE THE CURRENT VECTOR MAKES WITH THE NORMAL TO
1932 C +90 IS PARALLEL TO THE COAST TO THE LEFT OF FACING THE COAST
1933 C -90 IS PARALLEL TO THE COAST TO THE RIGHT
1934 C 00 IS FACING THE COAST (NORMAL)
1935 C DISTANCES FROM THE COAST ARE TABULATED IN ARRAY GDIST
1936 C
1937     IF(DELANG.GE.90.0) GO TO 11
1938     IF(DELANG.LE.-90.0) GO TO 11
1939 C ROTATE GRID POINTS NEXT TO THE COAST IN THE DIRECTION OF INCIDEN
1940 C AND LOGARITHMICALLY INVERSELY PROPORTIONAL TO DISTANCE
1941 C REDUCING FACTOR
1942     REDFAC=(1.0-(1.0/EXP(GDIST(I,J))))
1943 C ROTATE
1944     ABD=ABS(DELANG)
1945 C MAXIMUM ROTATION IS 45 DEGREES
1946 C DIMINISHED BY E TO THE -DISTANCE FROM THE COAST (IN GRID LENGTHS
1947     DROT=45.0/EXP(GDIST(I,J))
1948 C SET ROTATION AT A MAXIMUM OF 30 DEGREES ANGLE OF INCIDENCE
1949 C DECREASING TO ZERO AT ZERO AND 90 DEGREES ANGLE OF INCIDENCE
1950     IF((ABD-30.0).LT.0.0) DIMFAC=SIN((ABD*90.0/30.0)/57.2958)
1951     IF((ABD-30.0).GE.0.0) DIMFAC=SIN(((90.0-ABD)*90.0/60.0)/57.29
1952     DROT=DROT*DIMFAC
1953     IF (DELANG.LT.0.0) DROT=-DROT
1954     CRANG2=CURANG+DROT
1955 C FIND NEW COMPONENTS OF CURRENT VECTOR AFTER ROTATION
1956     RLEN1=SQRT((U(I,J)*U(I,J))+V(I,J)*V(I,J))
1957     U2=RLEN1*COS(CRANG2/57.2958)
1958     V2=RLEN1*SIN(CRANG2/57.2958)
1959     IF(V(I,J).LT.0.0) V2=-ABS(V2)
1960     IF(U(I,J).LT.0.0) U2=-ABS(U2)
1961 C REDUCE MAGNITUDE OF CURRENT VECTOR
1962     UR=U2*REDFAC
1963     VR=V2*REDFAC
1964 C     WRITE(6,333)I,J,GDIST(I,J),REDFAC,DROT,CRANG2,RLEN1,U(I,J),V(
1965 C     'U2,V2,UR,VR
1966     333 FORMAT(2I4,11F10.4)
1967 C
1968     U2S(I,J)=UR
1969     V2S(I,J)=VR
1970     U(I,J)=UR
1971     V(I,J)=VR
1972 C
1973     11 CONTINUE
1974     10 CONTINUE

```

```

1975 C
1976 IF(NOSTOF.EQ.1) RETURN
1977 C
1978 C WRITE FILE OF CURRENTS ROTATED AT THE COAST
1979 213 DO 34 J=1,104
1980 DO 34 I=1,5
1981 I2=I*8
1982 I1=I2-7
1983 WRITE(54,100,END=214)(U2S(IK,J),IK=I1,I2),KYR5,MO5,KDYS
1984 34 CONTINUE
1985 214 DO 35 J=1,104
1986 DO 35 I=1,5
1987 I2=I*8
1988 I1=I2-7
1989 WRITE(54,100,END=215)(V2S(IK,J),IK=I1,I2),KYR5,MO5,KDYS
1990 35 CONTINUE
1991 215 CONTINUE
1992 100 FORMAT(8F8.2,3I3,A3)
1993 C
1994 C
1995 C
1996 RETURN
1997 END
1998 SUBROUTINE ANGLD(X,Y,AAA)
1999 IQ1=0
2000 IQ2=0
2001 IQ3=0
2002 IQ4=0
2003 IF (X.GE.0.0.AND.Y.GE.0.0) IQ1=1
2004 IF (X.LT.0.0.AND.Y.GE.0.0) IQ2=1
2005 IF (X.LT.0.0.AND.Y.LT.0.0) IQ3=1
2006 IF (X.GE.0.0.AND.Y.LT.0.0) IQ4=1
2007 PI=3.141596
2008 RAD=PI/180.0
2009 IF (X.GE.0.0.AND.X.LT.0.000001) X=0.000001
2010 IF (X.LT.0.0.AND.X.GT.-0.000001) X=-0.000001
2011 AA=Y/X
2012 IF (IQ1.EQ.1) ANG=ATAN(AA)
2013 IF (IQ1.EQ.1) GO TO 50
2014 IF (IQ2.EQ.1) ANG=PI + ATAN(AA)
2015 IF (IQ2.EQ.1) GO TO 50
2016 IF (IQ3.EQ.1) ANG=PI + ATAN(AA)
2017 IF (IQ3.EQ.1) GO TO 50
2018 IF (IQ4.EQ.1) ANG=(2.0*PI) + ATAN(AA)
2019 50 CONTINUE
2020 AAA=ANG/RAD
2021 C WRITE(6,101)IQ1,IQ2,IQ3,IQ4,Y,X,AA,ANG,AAA
2022 101 FORMAT(4I4,5F11.5)
2023 RETURN
2024 END
2025 SUBROUTINE IJTOLL(Y,X,Y2,X2)
2026 COMMON/LAT/ALAT2,LS2
2027 COMMON/LON/ALON2
2028 DIMENSION ALAT2(40,104),ALON2(40,104)
2029 C

```

```

2030 C
2031 C CONVERT (I,J) TO LAT-LONG
2032 C
2033 C
2034 C OR FIND LAT AND LONG OF RANDOM GRID POINTS (Y,X)
2035 C WITHIN SLOPING GRID SQUARE A----B
2036 C C----C
2037 C
2038 C
2039 IX=X
2040 DX=X-IX
2041 IY=Y
2042 DY=Y-IY
2043 C CORNER A = (IY,IX) INTEGERS
2044 C GO DOWN AC TO (YB,XB)
2045 YB=ALAT2(IY,IX)+(DY*(ALAT2(IY+1,IX)-ALAT2(IY,IX)))
2046 XB=ALON2(IY,IX)+(DY*(ALON2(IY+1,IX)-ALON2(IY,IX)))
2047 C COMPUTE WEIGHTED AVERAGE SLOPES FOR AB AND CD PER UNIT GRID DIST
2048 SLA=((1.0-DY)*(ALAT2(IY,IX+1)-ALAT2(IY,IX))) +
2049 ' (( DY)*(ALAT2(IY+1,IX+1)-ALAT2(IY+1,IX)))
2050 SLO=((1.0-DY)*(ALON2(IY,IX+1)-ALON2(IY,IX))) +
2051 ' (( DY)*(ALON2(IY+1,IX+1)-ALON2(IY+1,IX)))
2052 C FIND LAT-LONG OF X,Y
2053 Y2=YB+(DX*SLA)
2054 X2=XB+(DX*SLO)
2055 C WRITE(6,5011)ALAT2(IY,IX),ALAT2(IY,IX+1),ALAT2(IY+1,IX),ALAT2
2056 C ',IX+1),YB,SLA,Y,DY,Y2
2057 5011 FORMAT(' LAT CONV ',9F8.3)
2058 C WRITE(6,5012)ALON2(IY,IX),ALON2(IY,IX+1),ALON2(IY+1,IX),ALON2
2059 C ',IX+1),XB,SLO,X,DX,X2
2060 5012 FORMAT(' LON CONV ',9F8.3)
2061 RETURN
2062 END

```

APPENDIX 2

Listing of Fortran Program "M/PLOT/AUTO" Graphics Outputs on 40X104 Grid.

```

1  $RESET FREE
2  FILE 1(KIND=DISK,TITLE="FNWC/LATLON/GRID",FILETYPE=7)
3  FILE 2(KIND=DISK,FILETYPE=7)
4  FILE 6(KIND=PRINTER)
5  FILE 13(KIND=DISK,TITLE="CURRENTS/40X64/LANDSEA",FILETYPE=7)
6  FILE 14(KIND=DISK,TITLE="CURRENTS/40X64W/LANDSEA",FILETYPE=7)
7  FILE 25(KIND=DISK,TITLE="CURRENTS/40X64W/COAST/DISTANCES/NORMALS"
8     ',FILETYPE=7)
9  FILE 26(KIND=DISK,TITLE="CURRENTS/40X64/COAST/DISTANCES/NORMALS"
10     ',FILETYPE=7)
11 FILE 28(KIND=DISK,TITLE="CURRENTS/40X104/PROGRESVECT/IJPLOTPPOINTS"
12     'FILETYPE=7)
13 FILE 30(KIND=DISK,TITLE="CURRENTS/40X64/LATLON",FILETYPE=7)
14 FILE 31(KIND=DISK,TITLE="CURRENTS/40X64W/LATLON",FILETYPE=7)
15 FILE 35(KIND=DISK,TITLE="CURRENTS/40X104/FASTLATITUDES/CCSQ/FIG",
16     'MAXRECSIZE=14,BLOCKSIZE=420,NEWFILE=.TRUE.)
17 FILE 36(KIND=DISK,TITLE="CURRENTS/40X104/FASTLATITUDES/CCSQ/FIG",
18     'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
19 FILE 40(KIND=DISK,TITLE="AF",MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=
20 FILE 41(KIND=DISK,TITLE="PRINT/MORE/LAND",FILETYPE=7)
21 FILE 51(KIND=DISK,
22     'TITLE="CURRENTS/40X104/DSL/78/FM7",
23     'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
24 FILE 52(KIND=DISK,
25     'TITLE="CURRENTS/40X104/WIND/78/FM7/D17TOD31",
26     'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
27 FILE 49(KIND=DISK,TITLE="CURRENTS/40X104/TCURR/M1TOM12",
28     'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
29 FILE 50(KIND=DISK,TITLE="CURRENTS/40X104/DYNCURR/S",
30     'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
31 FILE 53(KIND=DISK,
32     'TITLE="CURRENTS/40X104/SFCOCN/78/FM7/D17TOD31",
33     'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
34 FILE 54(KIND=DISK,
35     'TITLE="CURRENTS/40X104/W10DD10/CURR/78/FM7/D17TOD31/CSTROTATD
36     'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
37 FILE 55(KIND=DISK,
38     'TITLE="CURRENTS/40X104/W10DD10/CURR/78/FM7/D17TOD31",
39     'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
40 FILE 60(KIND=DISK,
41     'TITLE="CURRENTS/40X104/PROGRESVECT/78/FM7/D17TOD31",
42     'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
43 FILE 16(KIND=DISK,TITLE="RACE/DEPTHCONTOURS/CHAR/50M",FILETYPE=7)
44 FILE 17(KIND=DISK,TITLE="RACE/DEPTHCONTOURS/CHAR/100M",FILETYPE=7)
45 FILE 18(KIND=DISK,TITLE="RACE/DEPTHCONTOURS/CHAR/200M",FILETYPE=7)
46 FILE 19(KIND=DISK,TITLE="RACE/DEPTHCONTOURS/CHAR/2000M",FILETYPE=7)
47 FILE 56(KIND=DISK,TITLE="CURRENTS/40X104/FASTBATH/CCSQ/FIG/50M",
48     'MAXRECSIZE=14,BLOCKSIZE=420,NEWFILE=.TRUE.)
49 FILE 57(KIND=DISK,TITLE="CURRENTS/40X104/FASTBATH/CCSQ/FIG/100M",
50     'MAXRECSIZE=14,BLOCKSIZE=420,NEWFILE=.TRUE.)

```

```

51 FILE 58(KIND=DISK,TITLE="CURRENTS/40X104/FASTBATH/CCSQ/FIG/200M",
52 'MAXRECSIZE=14,BLOCKSIZE=420,NEWFILE=.TRUE.)
53 FILE 59(KIND=DISK,TITLE="CURRENTS/40X104/FASTBATH/CCSQ/FIG/2000M",
54 'MAXRECSIZE=14,BLOCKSIZE=420,NEWFILE=.TRUE.)
55 FILE 66(KIND=DISK,TITLE="CURRENTS/40X104/FASTBATH/CCSQ/FIG/50M",
56 'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
57 FILE 67(KIND=DISK,TITLE="CURRENTS/40X104/FASTBATH/CCSQ/FIG/100M",
58 'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
59 FILE 68(KIND=DISK,TITLE="CURRENTS/40X104/FASTBATH/CCSQ/FIG/200M",
60 'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
61 FILE 69(KIND=DISK,TITLE="CURRENTS/40X104/FASTBATH/CCSQ/FIG/2000M",
62 'MAXRECSIZE=14,BLOCKSIZE=420,FILETYPE=7)
63 FILE 70(KIND=DISK,TITLE="RACE/COAST/CHAR/FNWC",FILETYPE=7)
64 FILE 71(KIND=DISK,TITLE="CURRENTS/40X104/FASTCOAST/CCSQ/FIG",
65 'MAXRECSIZE=14,BLOCKSIZE=420,NEWFILE=.TRUE.)
66 FILE 73(KIND=DISK,TITLE="CURRENTS/40X104/FASTCOAST/CCSQ/FIG",
67 'FILETYPE=7)
68 COMMON ANX(40,104),ANY(40,104),BB(40,104),LS3(40,104),
69 'GDIST(40,104)
70 COMMON/GROCOM/CFACT,M,N,NM,XI(105),YJ(41)
71 DIMENSION ALAT2(40,104),ALON2(40,104)
72 DIMENSION LS1(40,104),LS2(40,104)
73 DIMENSION U2(40,104),V2(40,104),OCUW(40,104),OCVW(40,104)
74 DIMENSION POCU(40,104),POCV(40,104),OCUU(4160),OCVV(4160),
75 'CSPEED(40,104)
76 DIMENSION OCUT(40,104),OCVT(40,104)
77 DIMENSION WXX(40,104),WYY(40,104),PWXX(40,104),PWYY(40,104)
78 DIMENSION CSLP(4160),BSV2(576)
79 DIMENSION T1(40,104),T2(40,104)
80 DIMENSION IA(80)
81 DIMENSION SLP(4160),SLP1(40,104),SLP2(40,104)
82 DIMENSION HWX(1),HWY(1),WXXU(4160),WYYV(4160),WSPEED(40,104)
83 DIMENSION HSLP(1),HSFOU(1),HSFOV(1)
84 DIMENSION DEPTH(4),NDATA(4),BATH(3,1000,4)
85 DIMENSION A(3),B(2),C(1)
86 DIMENSION C1(25000),C2(25000),C3(25000)
87 DIMENSION OCUS(40,104),OCVS(40,104)
88 DIMENSION OCUDD(40,104),OCVDD(40,104)
89 DIMENSION GU1(200),GV1(200),GU2(200),GV2(200)
90 DIMENSION A2(3)
91 DIMENSION IPV(200),JPV(200)
92 DIMENSION ALATSF(14)
93 DIMENSION PLO(2000),PLA(2000),IPEN(2000)
94 DIMENSION YMAX(100),IOFFMP(200)
95 C DATA ALATSF/31.9,46.5,22.3,60.7,32.7,65.9,35.8,67.4/
96 DATA ALATSF/47.8,46.2,60.5,33.9,65.7,32.9,
97 '67.3,35.8,65.9,32.8,60.5,22.5,46.3,32.0/
98 DATA A/" FROM YEAR= "/
99 DATA A2/" TO YEAR= "/
100 DATA B/" MONTH="/
101 DATA C/" DAY="/
102 DATA IAA/' '/
103 DATA IAB/'*****'/
104 DATA HWX/' WX '/,HWY/' WY '/
105 DATA HSLP/' SLP '/

```

```

106      DATA HSF0U/'OU0  '/,HSFOV/'OV0  '/
107      NN2=40
108      MM2=104
109      C
110      C
111      C-----
112      C PROGRAM TO PLOT CURRENT FIELDS ON 40X64W GRID
113      C 1. PLOT WIND VECTORS
114      C 2. PLOT WIND CURRENT VECTORS
115      C 3. PLOT THERMAL CURRENT VECTORS
116      C 4. PLOT GEOSTROPHIC CURRENT VECTORS
117      C 5. PLOT WIND + THERMAL + GEOSTROPHIC CURRENT VECTORS
118      C 6. ADD COASTAL BENDING OF CURRENTS TO 5. AND PLOT
119      C 7. PROGRESSIVE VECTORS (AS STORED IN FILE)
120      C
121      C
122      C
123      C-----
124      C
125      C
126      C#####
127      C#####
128      C LABEL LONG'S
129      C CHOOZE WHICH PLOTS TO DO
130          EVGRPT=0.0
131          PLWIND=0.0
132          PLOCHW=0.0
133          PLOCT=0.0
134          PLOCDD=0.0
135          PLOCS=0.0
136          PLSOCR=0.0
137      C PLOT SLP WITH CURRENTS (IPLSLP=1)
138          IPLSLP=0
139      C PROGRESIVE VECTORS
140          PROGV=1.0
141          IFCOP=1
142      C IF NPLOT=2 THE PLOT OF WIND+THERMAL+GEO CURRENT VECTORS
143      C (#5)WILL BE PLOTTED WITH THE COASTAL BENDING (#6) OVERLAYED
144      C ON THE PLOT.
145      C MAKE SURE THAT PLOCS AND PLSOCR IS BOTH EQUAL TO 1.0
146          NPLOT=0
147          WRITE(6,80)
148      80 FORMAT('1CONTROL CARD PARAMETERS FROM WFL CARD DECK'//)
149      81 FORMAT(F5.1,67A1)
150      82 FORMAT(F5.1,67A1/)
151          READ(5,81)EVGRPT,(CC(I),I=1,67)
152          WRITE(6,82)EVGRPT,(CC(I),I=1,67)
153          READ(5,81)PLWIND,(CC(I),I=1,67)
154          WRITE(6,82)PLWIND,(CC(I),I=1,67)
155          READ(5,81)PLOCHW,(CC(I),I=1,67)
156          WRITE(6,82)PLOCHW,(CC(I),I=1,67)
157          READ(5,81)IPLSLP,(CC(I),I=1,67)
158          WRITE(6,82)IPLSLP,(CC(I),I=1,67)
159          READ(5,81)PLOCDD,(CC(I),I=1,67)
160          WRITE(6,82)PLOCDD,(CC(I),I=1,67)

```



```

161     READ(5,81)PLOCS,(CC(I),I=1,67)
162     WRITE(6,82)PLOCS,(CC(I),I=1,67)
163     READ(5,81)PLSOGR,(CC(I),I=1,67)
164     WRITE(6,82)PLSOGR,(CC(I),I=1,67)
165     READ(5,81)PROGV,(CC(I),I=1,67)
166     WRITE(6,82)PROGV,(CC(I),I=1,67)
167     READ(5,81)NPLOT,(CC(I),I=1,67)
168     WRITE(6,82)NPLOT,(CC(I),I=1,67)
169     C SKIP PLOTS 1 UP TO 1
170     NOPLT=1
171     C DO NDAYS OF PLOTS
172     C   NDAYS=12
173     NDAYS=180
174     C   NDAYS=10
175     NDAYS=500
176     C#####
177     WFAC=1.0
178     TFAC=0.0
179     DDFAC=1.0
180     C
181     C
182     C
183     C READ IN LAT-LONGS OF 40X64 + 40X64W GRIDS (CURRENTS MODEL)
184     C
185     DO 630 IFILE=1,2
186     DO 631 IK=1,4
187     DO 632 J=41,104
188     I2=IK*10
189     I1=I2-9
190     IF(IFILE.EQ.1)READ(30,104)(ALAT2(I,J),I=I1,I2)
191     IF(IFILE.EQ.2)READ(30,104)(ALON2(I,J),I=I1,I2)
192     632 CONTINUE
193     631 CONTINUE
194     630 CONTINUE
195     104 FORMAT (10F8.3,I4)
196     DO 680 IFILE=1,2
197     DO 681 IK=1,4
198     DO 682 J=1,64
199     I2=IK*10
200     I1=I2-9
201     IF(IFILE.EQ.1)READ(31,104)(ALAT2(I,J),I=I1,I2)
202     IF(IFILE.EQ.2)READ(31,104)(ALON2(I,J),I=I1,I2)
203     682 CONTINUE
204     681 CONTINUE
205     680 CONTINUE
206     C
207     C ROUGH PRINT LATS AND LONGS
208     DO 51 I=1,40
209     C   WRITE(6,105) (ALAT2(I,J),J=1,104,5),ALAT2(I,104)
210     51 CONTINUE
211     DO 52 I=1,40
212     C   WRITE(6,105) (ALON2(I,J),J=1,104,5),ALON2(I,104)
213     52 CONTINUE
214     105 FORMAT (22F6.1)
215     C

```

```

216 C
217 C READ LAND-SEA TABLE
218 C
219 DO 1500 I=1,NN2
220 READ(14,1501,END=1503)(LS3(I,J),J=1,64)
221 C WRITE(6,1505) I,LS3(I,1)
222 1500 CONTINUE
223 1503 CONTINUE
224 1505 FORMAT (1H ,2I5)
225 1501 FORMAT(64I1)
226 DO 1600 I=1,NN2
227 READ(13,1501,END=1603)(LS3(I,K),K=41,104)
228 C WRITE(6,1505) I,LS3(I,41)
229 1600 CONTINUE
230 1603 CONTINUE
231 C
232 C WRITE LAND-SEA
233 WRITE(6,1514)
234 1514 FORMAT('1 LAND-SEA 2, (40X104)')//)
235 DO 1515 I=1,NN2
236 WRITE(6,1516)(LS3(I,J),J=1,104)
237 1515 CONTINUE
238 1516 FORMAT(1H ,104I1)
239 C
240 C
241 C
242 DO 8001 I=1,40
243 DO 8002 J=1,104
244 LS1(I,J)=1
245 8002 CONTINUE
246 8001 CONTINUE
247 C
248 C
249 C READ INITIAL I,J PLOINT TO PLOT IN PROGRESSIVE VECTOR PLOT
250 KK=1
251 1144 READ(28,1145,END=1146) IPV(KK),JPV(KK)
252 WRITE(6,1145) KK,IPV(KK),JPV(KK)
253 IF(IPV(KK).LE.0.OR.JPV(KK).LE.0) GO TO 1146
254 KK=KK+1
255 GO TO 1144
256 1145 FORMAT(3I5)
257 1146 CONTINUE
258 KK=KK-1
259 WRITE(6,1145) KK
260 IIEND=KK
261 C
262 C
263 C
264 XLN=12.
265 YLN=9.0
266 YYLEN=8.75
267 YYLEN=9.25
268 CALL RTIME(101)
269 C FOR SQUARE GRID --- XLN=12.0, YLN SHOULD BE =4.54369 PLUS 1 INCH
270 C FOR SQUARE GRID --- XLN=15.0, YLN SHOULD BE =5.67961 PLUS 1 INCH

```

```

271      XLN=15.0
272      YLN=6.67961
273      YYLEN=YLN+0.5
274      CALL PLOTS(960,3,4096,0.,17.0,0.,9.0)
275 C     CALL PLOTS(960,3,4096,0.,14.5,0.,10.9)
276 C     CALL PLOTS(480,0,1024,0.,14.5,0.,10.9)
277      CALL RTIME(102)
278      CALL STGRID(NN2,MM2,ALAT2,ALON2,LS1,XLN,YLN)
279 C
280 C
281 C     SKIP COMPUTATION OF PLOT LATITUDE LINES
282      NOLATS=1
283 C     NOLATS=0
284      IF (NOLATS.EQ.1) GO TO 2222
285 C     STORE 30, 40, 50, AND 60 DEGREE LATITUDE LINES
286      NCC=1
287      DLON=0.5
288      DO 1131 NY=3,6
289      CALL RTIME(1)
290      YY=FLOAT(NY)*10.0
291      ALOMAX=250.0
292      ALOMIN=90.0
293      XX=ALOMAX
294      NC=1
295      YY1=YY
296 1132 CONTINUE
297      XX1=XX
298      CALL CONVRT(XX,YY,NEND)
299      IF (NEND.EQ.1) XX=XX1-DLON
300      IF (NEND.EQ.1) YY=YY1
301 C     IF(NEND.EQ.1)WRITE (6,1151) XX1,XX,XLN,YY1,YY,YLN,NEND,NC,NCC
302      IF (NEND.EQ.1.AND.XX1.LT.110.0) GO TO 1131
303      IF (NEND.EQ.1.) GO TO 1132
304 1151 FORMAT (6F10.4,3I6)
305      NPEN=2
306      IF (NC.EQ.1) NPEN=3
307      PLO(NCC)=XX
308      PLA(NCC)=YY
309      IPEN(NCC)=NPEN
310      WRITE(35,1159) XX,YY,NPEN,NC,NCC
311 1159 FORMAT(2F10.4,3I10)
312      XX=XX1-DLON
313      YY=YY1
314      IF (XX.LT.ALOMIN) GO TO 1131
315      NC=NC+1
316      NCC=NCC+1
317      GO TO 1132
318 1131 CONTINUE
319      CLOSE(35,DISP=CRUNCH)
320 C     TEST WRITE
321      DO 1133 I=1,104
322      II=I
323      IF (II.GT.41) II=41
324      WRITE (6,1134)CFAC, M,N,NM,XI(I),YJ(II),I
325 1133 CONTINUE

```

```

326     1134 FORMAT (4I8,2F8.4,I8)
327     C
328     C
329     NAROW=1
330     IF (NAROW.EQ.1) GO TO 2222
331     C TEST ARROWS
332     XX1=0.0
333     YY1=0.0
334     DXX1=0.25
335     DYY1=0.1
336     DDXX1=0.05
337     DDYY1=0.1
338     NNN=1
339     1156 X1=XX1*(XLN/103.0)
340     Y1=(YLN-1.0)-(YY1*((YLN-1.0)/39.0))
341     X2=XX1+DXX1
342     Y2=YY1+DYY1
343     X2=X2*(XLN/103.0)
344     Y2=(YLN-1.0)-(Y2*((YLN-1.0)/39.0))
345     X1=X1+1.0
346     Y1=Y1+1.0
347     X2=X2+1.0
348     Y2=Y2+1.0
349     CALL PVECTR(X1,Y1,X2,Y2,10.0,10.0)
350     WRITE(6,1155)XX1,YY1,NNN,X1,Y1,X2,Y2
351     1155 FORMAT (1H ,2F10.3,15,4F10.3)
352     XX1=XX1+DXX1
353     DYY1=DYY1+DDYY1
354     YY1=YY1+DYY1
355     IF (YY1.GT.15.0.AND.DXX1.GE.0.0) DXX1=-DXX1
356     NNN=NNN+1
357     IF (YY1.LT.40.0) GO TO 1156
358     C
359     C STOP HERE
360     GO TO 11111
361     2222 CONTINUE
362     C
363     C READ FASTLATS
364     IF(NOLATS.LT.1) GO TO 2223
365     NCC=1
366     2225 READ(36,1159,END=2223)PLO(NCC),PLA(NCC),IPEN(NCC)
367     NCC=NCC+1
368     GO TO 2225
369     2223 CONTINUE
370     C
371     NCC=NCC-1
372     IPEN(NCC)=3
373     C
374     C
375     CALL RTIME(2)
376     C
377     C MAKE FASTCOASTLINE FILE
378     ISKIP=1
379     IFAST=1
380     IFAST=0

```

```

381     IF (IFAST.EQ.1) CALL FASTCO(IFAST,ISKIP,IC)
382     IF (IFAST.LT.20) CALL CSET(73,IC,C1,C2,C3)
383     C   CALL COAST(0,1,IC,C1,C2,C3)
384     CALL RTIME(3)
385     C*
386     C*
387     C* READ BATHYMETRY FILE
388     C*
389     DEPTH(1)=50.
390     DEPTH(2)=100.
391     DEPTH(3)=200.0
392     DEPTH(4)=2000.0
393     NDATA(1)=1000
394     NDATA(2)=1000
395     NDATA(3)=1000
396     NDATA(4)=1000
397     NDATA(2)=0
398     NDATA(4)=0
399     C
400     C
401     C SKIP CONVERSION OF BATHYMETRY IF NOCONV=1
402     NOCONV=1
403     C   NOCONV=0
404     IF(NOCONV.EQ.1) GO TO 951
405     C
406     IB1=0
407     IB2=0
408     IEND1=0
409     IEND2=0
410     IEND3=0
411     IEND4=0
412     C
413     910 CONTINUE
414     C
415     NDATA(1)=1000
416     DO 911 II1=1,1000
417     IF (IEND1.EQ.1) NDATA(1)=0
418     IF (IEND1.EQ.1) GO TO 913
419     READ(16,9900,END=912) (BATH(J,II1,1),J=1,3),NRECS1
420     IF (II1.EQ.1) BATH(3,1,1)=3
421     911 CONTINUE
422     GO TO 913
423     912 IEND1=1
424     NDATA(1)=II1-2
425     WRITE(6,9911) II1,NRECS1
426     9911 FORMAT(' END OF FILE 16, LAST READ =',2I6)
427     913 CONTINUE
428     C
429     NDATA(2)=0
430     DO 921 II2=1,1000
431     IF (IEND2.EQ.1) NDATA(2)=0
432     IF (IEND2.EQ.1) GO TO 923
433     READ(17,9900,END=922) (BATH(J,II2,2),J=1,3),NRECS2
434     IF (II2.EQ.1) BATH(3,1,2)=3
435     921 CONTINUE

```

```

436      GO TO 923
437      922 IEND2=1
438      NDATA(2)=II2-2
439      WRITE(6,9912) II2,NRECS2
440      9912 FORMAT(' END OF FILE 17, LAST READ =',2I6)
441      923 CONTINUE
442      C
443      NDATA(3)=1000
444      DO 931 II3=1,1000
445      IF (IEND3.EQ.1) NDATA(3)=0
446      IF (IEND3.EQ.1) GO TO 933
447      READ(18,9900,END=932) (BATH(J,II3,3),J=1,3),NRECS3
448      IF (II3.EQ.1) BATH(3,1,3)=3
449      931 CONTINUE
450      GO TO 933
451      932 IEND3=1
452      NDATA(3)=II3-2
453      WRITE(6,9913) II3,NRECS3
454      9913 FORMAT(' END OF FILE 18, LAST READ =',2I6)
455      933 CONTINUE
456      C
457      NDATA(4)=0
458      DO 941 II4=1,1000
459      IF (IEND4.EQ.1) NDATA(4)=0
460      IF (IEND4.EQ.1) GO TO 943
461      READ(19,9900,END=942) (BATH(J,II4,4),J=1,3),NRECS4
462      IF (II4.EQ.1) BATH(3,1,4)=3
463      941 CONTINUE
464      GO TO 943
465      942 IEND4=1
466      WRITE(6,9914) II4,NRECS4
467      9914 FORMAT(' END OF FILE 19, LAST READ =',2I6)
468      NDATA(4)=II4-2
469      943 CONTINUE
470      WRITE(6,9901)NRECS1,NRECS2,NRECS3,NRECS4,(NDATA(NN),NN=1,4)
471      9900 FORMAT(2F10.4,2I10)
472      C
473      C CONVERT BATHYMETRY
474      CALL BATHY(1,0,0,BATH,NDATA,DEPTH)
475      C
476      C
477      C STORE FILES OF FAST BATHYMETRY CONVERTED
478      DO 915 NB=1,4
479      IFILE=55+NB
480      IF (NDATA(NB).EQ.0) GO TO 915
481      DO 916 NK=1,NDATA(NB)
482      WRITE(IFILE,9900) (BATH(J,NK,NB),J=1,3)
483      916 CONTINUE
484      915 CONTINUE
485      CLOSE(56,DISP=CRUNCH)
486      CLOSE(57,DISP=CRUNCH)
487      CLOSE(58,DISP=CRUNCH)
488      CLOSE(59,DISP=CRUNCH)
489      C
490      C

```

```

491     951 CONTINUE
492     C
493         CALL RTIME(4)
494     C READ CONVERTED FAST BATHYMETRY FILES
495         IF (NOCONV.NE.1) GO TO 960
496         DO 952 NK=1,1000
497             READ(66,9900,END=953) (BATH(J,NK,1),J=1,3)
498         952 CONTINUE
499         953 CONTINUE
500             NDATA(1)=NK-1
501             WRITE(6,9926) NDATA(1)
502         9926 FORMAT ('0END OF FILE 66 ',I5,' RECORDS READ'/)
503     C
504         DO 954 NK=1,1000
505             READ(67,9900,END=955) (BATH(J,NK,2),J=1,3)
506         954 CONTINUE
507         955 CONTINUE
508             NDATA(2)=NK-1
509             WRITE(6,9927) NDATA(2)
510         9927 FORMAT ('0END OF FILE 67 ',I5,' RECORDS READ'/)
511     C
512         DO 956 NK=1,1000
513             READ(68,9900,END=957) (BATH(J,NK,3),J=1,3)
514         956 CONTINUE
515         957 CONTINUE
516             NDATA(3)=NK-1
517             WRITE(6,9928) NDATA(3)
518         9928 FORMAT ('0END OF FILE 68 ',I5,' RECORDS READ'/)
519     C
520         DO 958 NK=1,1000
521             READ(69,9900,END=959) (BATH(J,NK,4),J=1,3)
522         958 CONTINUE
523         959 CONTINUE
524             NDATA(4)=NK-1
525             WRITE(6,9929) NDATA(4)
526         9929 FORMAT ('0END OF FILE 69 ',I5,' RECORDS READ'/)
527         960 CONTINUE
528             CALL RTIME(5)
529     C
530     C* DRAW BATHYMETRY
531     C
532         WRITE(6,9901)NRECS1,NRECS2,NRECS3,NRECS4,(NDATA(NN),NN=1,4)
533         9901 FORMAT(8I10)
534             IFLBLR=1
535             CALL RTIME(6)
536     C
537     C
538     C GET MORE BATHYMETRY DATA
539     C     IF(IEND1.EQ.1.AND.IEND2.EQ.1) IB1=1
540     C     IF(IEND3.EQ.1.AND.IEND4.EQ.1) IB2=1
541     C     IF(IB1.NE.1.OR.IB2.NE.1) GO TO 910
542     C
543     C     DO 36 J=1,104
544     C     DO 36 I=1,5
545         I2=I*8

```

```

546      I1=I2-7
547    C  READ (49,100,END=216)(OCUT(IK,J),IK=I1,I2),KYR6,MO6,KDY6
548      36 CONTINUE
549    C 216 DO 37 J=1,104
550      C  DO 37 I=1,5
551        I2=I*8
552        I1=I2-7
553    C  READ(49,100,END=217)(OCVT(IK,J),IK=I1,I2),KYR7,MO7,KDY7
554      37 CONTINUE
555    217 CONTINUE
556      DO 38 J=1,104
557      DO 38 I=1,5
558      I2=I*8 ; I1=I2-7
559      READ(50,100,END=218)(OCUDD(IK,J),IK=I1,I2),KYR8,MO8,KDY8
560    38 CONTINUE
561    218 CONTINUE
562      DO 39 J=1,104
563      DO 39 I=1,5
564      I2=I*8 ; I1=I2-7
565      READ(50,100,END=219)(OCVDD(IK,J),IK=I1,I2),KYR9,MO9,KDY9
566    39 CONTINUE
567    219 CONTINUE
568    C
569    C MULTIPLY BY TEMP AND DD FACTORS HERE
570      DO 406 J=1,104
571      DO 407 I=1,40
572      OCUT(I,J)=OCUT(I,J)*TFAC
573      OCVT(I,J)=OCVT(I,J)*TFAC
574      OCUDD(I,J)=OCUDD(I,J)*DDFAC
575      OCVDD(I,J)=OCVDD(I,J)*DDFAC
576    407 CONTINUE
577    406 CONTINUE
578    C
579    C
580    C DO 1 SETS OF PLOTS
581      DO 9000 KPLT=1,NDAYS
582    C
583    C READ IN SLP, WIND, AND OCNC FILES
584    C
585      IF(IPLSLP.LT.1.0.AND.PLOCW.LT.1.0.AND.PLWIND.LT.1.0) GO TO 21
586      DO 31 J=1,104
587      DO 31 I=1,5
588      I2=I*8
589      I1=I2-7
590      READ(51,100,END=211)(SLP1(IK,J),IK=I1,I2),KYR1,MO1,KDY1
591    31 CONTINUE
592    211 DO 32 J=1,104
593      DO 32 I=1,5
594      I2=I*8
595      I1=I2-7
596      READ(52,100,END=212)(WXX(IK,J),IK=I1,I2),KYR2,MO2,KDY2
597    32 CONTINUE
598    212 DO 33 J=1,104
599      DO 33 I=1,5
600      I2=I*8

```



```

601         I1=I2-7
602         READ(52,100,END=213)(WYY(IK,J),IK=I1,I2),KYR3,MO3,KDY3
603         33 CONTINUE
604         213 DO 34 J=1,104
605             DO 34 I=1,5
606                 I2=I*8
607                 I1=I2-7
608             READ (53,100,END=214)(OCUW(IK,J),IK=I1,I2),KYR4,MO4,KDY4
609         34 CONTINUE
610         214 DO 35 J=1,104
611             DO 35 I=1,5
612                 I2=I*8
613                 I1=I2-7
614             READ(53,100,END=9000)(OCVW(IK,J),IK=I1,I2),KYR5,MO5,KDY5
615         35 CONTINUE
616         CALL DATFIX(SLP1,CSLP)
617         215 CONTINUE
618         100 FORMAT(8F8.3,3I3,A3)
619         C
620         C WRITE(6,231) KYR1,MO1,KDY1
621         C WRITE(6,231) KYR2,MO2,KDY2
622         C WRITE(6,231) KYR3,MO3,KDY3
623         C WRITE(6,231) KYR4,MO4,KDY4
624         C WRITE(6,231) KYR5,MO5,KDY5
625         C WRITE(6,231) KYR6,MO6,KDY6
626         C WRITE(6,231) KYR7,MO7,KDY7
627         C WRITE(6,231) KYR8,MO8,KDY8
628         C WRITE(6,231) KYR9,MO9,KDY9
629         231 FORMAT(3I10)
630         YEAR=KYR1
631         AMO=MO1
632         DAY=KDY1
633         C CALL RTIME(7)
634         FYR=KYR1
635         FMO=MO1
636         FDY=KDY1
637         C
638         C
639         C SKIP PLOTS FROM 1 TO NOPLT
640             IF (KPLT.LT.NOPLT) GO TO 965
641         C
642         C-----
643         C-----
644         C
645         C DO PLOT NUMBER 1 (SLP CONTOURS + WIND VECTORS)
646         C
647         C-----
648         C-----
649             IF (PLWIND.LT.1.0) GO TO 961
650         C
651         C PRINT SLP (40X64W) CONTOUR ARRAY
652         C WRITE (6,664)
653         664 FORMAT('1 SLP CONTOURS'/)
654         C DO 175 KJ=1,416
655         C I2=KJ*10

```

```

656 C I1=I2-9
657 C WRITE (6,176) (CSLP(IK),IK=I1,I2),I1,I2,KJ
658 175 CONTINUE
659 176 FORMAT (10F10.2,3I5)
660 CALL PLOT (1.0,1.0,-3)
661 CALL NEWPEN(2)
662 CALL GRIDER(102,38)
663 C CALL GRIDER(0,0)
664 CALL NUMBER
665 C
666 C PLOT LATITUDE LINES
667 DO 9141 IPK=1,NCC
668 CALL PLOT (PLO(IPK),PLA(IPK),IPEN(IPK))
669 9141 CONTINUE
670 C
671 CALL NEWPEN(2)
672 C
673 C PLOT LONGITUDES/CCSQ/FIG FROM 910 (+20) TO 230 DEGREES
674 CHSZ=0.307
675 DO 9161 LO=1,7
676 IF(LO.EQ.1) GO TO 9161
677 XX=110.0+((LO-1)*20.0)
678 XXL=XX
679 NLA=LO*2
680 YY=ALATSF(NLA-1)
681 XX1=XX
682 YY1=YY
683 CALL CONVRT(XX,YY,IEND)
684 CALL PLOT (XX,YY,3)
685 WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
686 YY=ALATSF(NLA)
687 XX=XX1
688 YY1=YY
689 CALL CONVRT(XX,YY,IEND)
690 CALL PLOT (XX,YY,2)
691 WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
692 IF(LO.EQ.2) GO TO 9161
693 IF (LO.EQ.7) GO TO 9161
694 XA=XX-(2.0*CHSZ)
695 YA=YY - 0.5
696 IF (LO.EQ.5) XXL=170.0
697 IF (LO.EQ.6) XXL=150.0
698 CALL NUMBER(XA,YA,CHSZ,XXL,0.0,-1)
699 XAW=999.0
700 YAW=999.0
701 IF(LO.EQ.3.OR.LO.EQ.4)CALL SYMBOL(XAW,YAW,CHSZ,' W',0.0,2)
702 IF(LO.GE.5.OR.LO.EQ.6)CALL SYMBOL(XAW,YAW,CHSZ,' E',0.0,2)
703 9161 CONTINUE
704 C LABEL LAT'S
705 ALO=170.0
706 DO 9176 NL=1,2
707 ALA=30.0 + ((NL-1)*10)
708 ALA1=ALA
709 IF(NL.EQ.1) CLO=133.5
710 IF(NL.EQ.2) CLO=123.0

```

```

711     CALL CONVRT(CLO,ALA,IEND)
712     WRITE(6,6072)CLO,CLA,ALA1
713     CLA=ALA-(0.5*CHSZ)
714     CLO=XLN+0.1
715     CALL NUMBER(CLO,CLA,CHSZ,ALA1,0.0,-1)
716     WRITE(6,6072)CLO,CLA,ALA1
717     IF (NL.LT.3)CALL SYMBOL(999.0,999.0,CHSZ,' N',0.0,2)
718 9176 CONTINUE
719 C
720     XX=0.5
721     YY=YYLEN-0.3
722     HH=0.2
723     CALL SYMBOL(XX,YY,HH,'SEA LEVEL PRESSURE (MB) ',0.0,26)
724     CALL NUMBER(999.0,999.0,HH,FYR,0.0,-1)
725     CALL SYMBOL(999.0,999.0,HH,' - ',0.0,3)
726     CALL NUMBER(999.0,999.0,HH,FMO,0.0,-1)
727     CALL SYMBOL(999.0,999.0,HH,' - ',0.0,3)
728     CALL NUMBER(999.0,999.0,HH,FDY,0.0,-1)
729 C
730 C
731     CALL NEWPEN(1)
732     CALL CONTOR(0,1,900.,1600.,4.,8.,16.,0,CSLP)
733 C
734 C SCALE DOWN ARROWS IN WIND VECTOR PLOT
735     FACTOR=0.25
736     DO 844 I=1,NN2
737     DO 845 J=1,MM2
738     WSPEED(I,J)=SQRT((WXX(I,J)*WXX(I,J))+(WYY(I,J)*WYY(I,J)))
739     PWXX(I,J)=WXX(I,J)*FACTOR
740     PWYY(I,J)=WYY(I,J)*FACTOR
741 845 CONTINUE
742 844 CONTINUE
743 C
744     CALL PRINT1(WSPEED,8)
745 C
746 C
747 C ZERO OUT EVERY OTHER ROW IND COL.
748 C TO REDUCE THE NUMBER OF ARROWS IN PLOT
749     IF (EVGRPT.EQ.1.0) GO TO 971
750     DO 765 JR=1,104,2
751     DO 765 IR=1,40
752     PWXX(IR,JR)=0.0
753     PWYY(IR,JR)=0.0
754 765 CONTINUE
755     DO 766 IR=1,40,2
756     DO 766 JR=1,104
757     PWXX(IR,JR)=0.0
758     PWYY(IR,JR)=0.0
759 766 CONTINUE
760 971 CONTINUE
761     DO 767 IR=1,40
762     PWXX(IR,50)=0.0
763     PWYY(IR,50)=0.0
764 767 CONTINUE
765     DO 768 JR=1,104

```

```

766         PWXX(40,JR)=0.0
767         PWYY(40,JR)=0.0
768     768 CONTINUE
769     C
770     C
771         CALL DATFIX(PWXX,WXXU)
772         CALL DATFIX(PWYY,WYYV)
773         CALL VECTOR(WXXU,WYYV)
774     C
775     C
776     C
777     C PLOT SCALE ARROW (1 INCH = 10CM/SEC)
778         XP=XLN*0.72
779         YP=YYLEN-0.4
780         UVAL=10.0*FACTOR
781         CALL AVECTR(XP,YP,UVAL,0.0)
782         XP=XP-0.4
783         YP=YYLEN-0.3
784         CALL SYMBOL(XP,YP,0.2,9H 10 M/SEC ,0.,9)
785         XP=XLN*0.63
786         YP=YYLEN
787         CALL SYMBOL(XP,YP,0.2,20HSURFACE WIND VECTORS ,0.0,20)
788     C
789         CALL COAST(0,1,IC,C1,C2,C3)
790         CALL BATHY(0,1,IFLBLR,BATH,NDATA,DEPTH)
791         CALL FIXER(IFCOP)
792     961 CONTINUE
793     C     CALL RTIME(8)
794         IF (PLOCW.LT.1.0) GO TO 962
795     C
796     C-----
797     C-----
798     C
799     C DO PLOT NUMBER 2 (SLP CONTOURS + WIND COMP. OCEAN SURFACE CURREN
800     C
801     C-----
802     C-----
803     C
804     C COMPUTE AND SCALE DOWN ARROWS IN OCEAN CURRENT VECTOR PLOT
805     C SEE FUNCTION "WINDC" (SFC CURR DUE TO WIND)
806         FACTOR=0.25
807         DO 1844 I=1,NN2
808         DO 1845 J=1,MM2
809             CSPEED(I,J)=SQRT(OCUW(I,J)*OCUW(I,J)+OCVW(I,J)*OCVW(I,J))
810             IF(LS3(I,J).EQ.0)OCUW(I,J)=0.0
811             IF(LS3(I,J).EQ.0)OCVW(I,J)=0.0
812             POCU(I,J)=OCUW(I,J)*FACTOR
813             POCV(I,J)=OCVW(I,J)*FACTOR
814     1845 CONTINUE
815     1844 CONTINUE
816     C
817         CALL PRINT1(CSPEED,11)
818     C
819     C ZERO OUT EVERY OTHER ROW IND COL.
820     C TO REDUCE THE NUMBER OF ARROWS IN PLOT

```

```

821     IF (EVGRPT.EQ.1.0) GO TO 972
822     DO 865 JR=1,104,2
823     DO 865 IR=1,40
824     POCU(IR,JR)=0.0
825     POCV(IR,JR)=0.0
826     865 CONTINUE
827     DO 866 IR=1,40,2
828     DO 866 JR=1,104
829     POCU(IR,JR)=0.0
830     POCV(IR,JR)=0.0
831     866 CONTINUE
832     972 CONTINUE
833     DO 867 IR=1,40
834     POCU(IR,104)=0.0
835     POCV(IR,104)=0.0
836     867 CONTINUE
837     DO 868 JR=1,104
838     POCU(40,JR)=0.0
839     POCV(40,JR)=0.0
840     868 CONTINUE
841     C
842     C
843     C
844     CALL PLOT(1.,1.,-3)
845     CALL DATFIX(POCU,OCUU)
846     CALL DATFIX(POCV,OCVV)
847     CALL NEWPEN(1)
848     CALL VECTOR(OCUU,OCVV)
849     CALL NEWPEN(2)
850     C
851     C CALL CONTOR(0,1,900.,1600.,8.,16.,16.,0,CSLP)
852     CALL CONTOR(0,1,900.,1600.,4.,8.,16.,0,CSLP)
853     CALL GRIDER(102,38)
854     C CALL GRIDER(0,0)
855     CALL NUMMER
856     C
857     C PLOT LATITUDE LINES
858     DO 8141 IPK=1,NCC
859     CALL PLOT (PLO(IPK),PLA(IPK),IPEN(IPK))
860     8141 CONTINUE
861     C
862     CALL NEWPEN(2)
863     C
864     C PLOT LONGITUDES/CCSQ/FIG FROM 810 (+20) TO 230 DEGREES
865     CHSZ=0.307
866     DO 8161 LO=1,7
867     IF(LO.EQ.1) GO TO 8161
868     XX=110.0+((LO-1)*20.0)
869     XXL=XX
870     NLA=LO*2
871     YY=ALATSF(NLA-1)
872     XX1=XX
873     YY1=YY
874     CALL CONVRT(XX,YY,IEND)
875     CALL PLOT (XX,YY,3)

```

```

876      WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
877      YY=ALATSF(NLA)
878      XX=XX1
879      YY1=YY
880      CALL CONVRT(XX,YY,IEND)
881      CALL PLOT (XX,YY,2)
882      WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
883      IF(LO.EQ.2) GO TO 8161
884      IF (LO.EQ.7) GO TO 8161
885      XA=XX-(2.0*CHSZ)
886      YA=YY - 0.5
887      IF (LO.EQ.5) XXL=170.0
888      IF (LO.EQ.6) XXL=150.0
889      CALL NUMBER(XA,YA,CHSZ,XXL,0.0,-1)
890      XAW=999.0
891      YAW=999.0
892      IF(LO.EQ.3.OR.LO.EQ.4)CALL SYMBOL(XAW,YAW,CHSZ,' W',0.0,2)
893      IF(LO.GE.5.OR.LO.EQ.6)CALL SYMBOL(XAW,YAW,CHSZ,' E',0.0,2)
894      8161 CONTINUE
895      C LABEL LAT'S
896      ALO=170.0
897      DO 8176 NL=1,2
898      ALA=30.0 + ((NL-1)*10)
899      ALA1=ALA
900      IF(NL.EQ.1) CLO=133.5
901      IF(NL.EQ.2) CLO=123.0
902      CALL CONVRT(CLO,ALA,IEND)
903      WRITE(6,6072)CLO,CLA,ALA1
904      CLA=ALA-(0.5*CHSZ)
905      CLO=XLN+0.1
906      CALL NUMBER(CLO,CLA,CHSZ,ALA1,0.0,-1)
907      WRITE(6,6072)CLO,CLA,ALA1
908      IF (NL.LT.3)CALL SYMBOL(999.0,999.0,CHSZ,' N',0.0,2)
909      8176 CONTINUE
910      C
911      XX=0.5
912      YY=YYLEN-0.3
913      HH=0.2
914      CALL SYMBOL(XX,YY,HH,'SEA LEVEL PRESSURE (MB) ',0.0,26)
915      CALL NUMBER(999.0,999.0,HH,FYR,0.0,-1)
916      CALL SYMBOL(999.0,999.0,HH,' - ',0.0,3)
917      CALL NUMBER(999.0,999.0,HH,FMO,0.0,-1)
918      CALL SYMBOL(999.0,999.0,HH,' - ',0.0,3)
919      CALL NUMBER(999.0,999.0,HH,FDY,0.0,-1)
920      C
921      C
922      C
923      C* DRAW BATHYMETRY
924      C
925      WRITE(6,9901)NRECS1,NRECS2,NRECS3,NRECS4,(NDATA(NN),NN=1,4)
926      IFLBLR=1
927      CALL BATHY(0,1,IFLBLR,BATH,NDATA,DEPTH)
928      C
929      C
930      C PLOT SCALE ARROW (1 INCH = 10CM/SEC)

```

```

931      XP=XLN*0.72
932      YP=YYLEN-0.7
933      UVAL=10.0*FACTOR
934      CALL AVECTR(XP,YP,UVAL,0.0)
935      XP=XP-0.4
936      YP=YP+0.1
937      CALL SYMBOL(XP,YP,0.2,10H10 CM/SEC ,0.,10)
938      XP=XLN*0.63
939      YP=YYLEN
940      CALL SYMBOL(XP,YP,0.2,22HSURFACE OCEAN CURRENTS ,0.0,22)
941      YP=YP-0.3
942      CALL SYMBOL(XP,YP,0.2,19H (WIND COMPONENT) ,0.0,19)
943      C
944      CALL COAST(0,1,IC,C1,C2,C3)
945      CALL FIXER(IFCOP)
946      962 CONTINUE
947      C CALL RTIME(9)
948      IF (PLOCT.LT.1.0) GO TO 963
949      C
950      C-----
951      C-----
952      C
953      C DO PLOT NUMBER 3 (THERMAL COMPONENT OF OCEAN SURFACE CURRENTS-0/
954      C
955      C-----
956      C-----
957      C
958      C COMPUTE AND SCALE DOWN ARROWS IN OCEAN CURRENT VECTOR PLOT
959      FACTOR=1.25
960      DO 2844 I=1,NN2
961      DO 2845 J=1,MM2
962      CSPEED(I,J)=SQRT(OCUT(I,J)*OCUT(I,J)+OCVT(I,J)*OCVT(I,J))
963      IF(LS3(I,J).EQ.0)OCUT(I,J)=0.0
964      IF(LS3(I,J).EQ.0)OCVT(I,J)=0.0
965      POCU(I,J)=OCUT(I,J)*FACTOR
966      POCV(I,J)=OCVT(I,J)*FACTOR
967      2845 CONTINUE
968      2844 CONTINUE
969      C
970      CALL PRINT1(CSPEED,11)
971      C
972      C ZERO OUT EVERY OTHER ROW IND COL.
973      C TO REDUCE THE NUMBER OF ARROWS IN PLOT
974      IF (EVGRPT.EQ.1.0) GO TO 973
975      DO 875 JR=1,104,2
976      DO 875 IR=1,40
977      POCU(IR,JR)=0.0
978      POCV(IR,JR)=0.0
979      875 CONTINUE
980      DO 876 IR=1,40,2
981      DO 876 JR=1,104
982      POCU(IR,JR)=0.0
983      POCV(IR,JR)=0.0
984      876 CONTINUE
985      DO 877 IR=1,40

```

```

986      POCU(IR,104)=0.0
987      POCV(IR,104)=0.0
988      877 CONTINUE
989      DO 878 JR=1,104
990      POCU(40,JR)=0.0
991      POCV(40,JR)=0.0
992      878 CONTINUE
993      973 CONTINUE
994      C
995      C
996      C
997      CALL PLOT(1.,1.,-3)
998      CALL DATFIX(POCU,OCUU)
999      CALL DATFIX(POCV,OCVV)
1000     CALL NEWPEN(1)
1001     CALL VECTOR(OCUU,OCVV)
1002     CALL NEWPEN(2)
1003     C
1004     C   CALL CONTOR(0,1,900.,1600.,8.,16.,16.,1,CSLP)
1005     CALL GRIDER(102,38)
1006     C   CALL GRIDER(0,0)
1007     CALL NUMMER
1008     CALL SYMBOL(0.,YYLEN,.2,A,0.,16)
1009     XX=2.0
1010     CALL NUMBER(XX,YYLEN,.2,YEAR,0.,-1)
1011     XX=XX+.6
1012     CALL SYMBOL(XX,YYLEN,.2,B,0.,9)
1013     XX=XX+1.0
1014     CALL NUMBER(XX,YYLEN,.2,AMO,0.,-1)
1015     XX=XX+0.4
1016     CALL SYMBOL(XX,YYLEN,0.2,C,0.0,6)
1017     XX=XX+0.75
1018     CALL NUMBER(XX,YYLEN,0.2,DAY,0.0,-1)
1019     C
1020     C*  DRAW BATHYMETRY
1021     C
1022     WRITE(6,9901)NRECS1,NRECS2,NRECS3,NRECS4,(NDATA(NN),NN=1,4)
1023     IFLBLR=1
1024     CALL BATHY(0,1,IFLBLR,BATH,NDATA,DEPTH)
1025     C
1026     C
1027     C   PLOT SCALE ARROW (1 INCH = 10CM/SEC)
1028     XP=XLN*.72
1029     YP=YYLEN-0.7
1030     UVAL=10.0*FACTOR
1031     CALL AVECTR(XP,YP,UVAL,0.0)
1032     XP=XP-0.4
1033     YP=YP+0.1
1034     CALL SYMBOL(XP,YP,0.2,10H10 CM/SEC ,0.,10)
1035     XP=XLN*.63
1036     YP=YYLEN
1037     CALL SYMBOL(XP,YP,0.2,22HSURFACE OCEAN CURRENTS ,0.0,22)
1038     YP=YP-0.3
1039     CALL SYMBOL(XP,YP,0.2,22H (THERMAL COMPONENT) ,0.0,22)
1040     C

```



```

1041     CALL COAST(0,1,IC,C1,C2,C3)
1042     CALL FIXER(IFCOP)
1043     963 CONTINUE
1044     C     CALL RTIME(10)
1045     IF (PLOCDD.LT.1.0) GO TO 964
1046     C
1047     C-----
1048     C-----
1049     C
1050     C DO PLOT NUMBER 4 (GEOSTROPHIC OCEAN SURFACE CURRENTS-0/2000 DB)
1051     C
1052     C-----
1053     C-----
1054     C COMPUTE AND SCALE DOWN ARROWS IN OCEAN CURRENT VECTOR PLOT
1055     FACTOR=.25
1056     DO 2944 I=1,NN2
1057     DO 2945 J=1,MM2
1058     OCUS(I,J)=OCUDD(I,J)
1059     OCVS(I,J)=OCVDD(I,J)
1060     CSPEED(I,J)=SQRT(OCUS(I,J)*OCUS(I,J)+OCVS(I,J)*OCVS(I,J))
1061     IF(LS3(I,J).EQ.0)OCUS(I,J)=0.0
1062     IF(LS3(I,J).EQ.0)OCVS(I,J)=0.0
1063     POCU(I,J)=OCUS(I,J)*FACTOR
1064     POCV(I,J)=OCVS(I,J)*FACTOR
1065     2945 CONTINUE
1066     2944 CONTINUE
1067     C
1068     CALL PRINT1(CSPEED,11)
1069     C
1070     C ZERO OUT EVERY OTHER ROW IND COL.
1071     C TO REDUCE THE NUMBER OF ARROWS IN PLOT
1072     IF (EVGRPT.EQ.1.0) GO TO 974
1073     DO 885 JR=1,104,2
1074     DO 885 IR=1,40
1075     POCU(IR,JR)=0.0
1076     POCV(IR,JR)=0.0
1077     885 CONTINUE
1078     DO 886 IR=1,40,2
1079     DO 886 JR=1,104
1080     POCU(IR,JR)=0.0
1081     POCV(IR,JR)=0.0
1082     886 CONTINUE
1083     DO 887 IR=1,40
1084     POCU(IR,104)=0.0
1085     POCV(IR,104)=0.0
1086     887 CONTINUE
1087     DO 888 JR=1,104
1088     POCU(40,JR)=0.0
1089     POCV(40,JR)=0.0
1090     888 CONTINUE
1091     974 CONTINUE
1092     C
1093     C
1094     C
1095     CALL PLOT(1.,1.,-3)

```

```

1096     CALL DATFIX(POCU,OCUU)
1097     CALL DATFIX(POCV,OCVV)
1098     CALL NEWPEN(1)
1099     CALL VECTOR(OCUU,OCVV)
1100     CALL NEWPEN(2)
1101     C
1102     C   CALL CONTOR(0,1,900.,1600.,8.,16.,16.,1,CSLP)
1103     CALL GRIDER(102,38)
1104     C   CALL GRIDER(0,0)
1105     CALL NUMBER
1106     C
1107     C PLOT LATITUDE LINES
1108     DO 7141 IPK=1,NCC
1109     CALL PLOT (PLO(IPK),PLA(IPK),IPEN(IPK))
1110     7141 CONTINUE
1111     C
1112     CALL NEWPEN(2)
1113     C
1114     C PLOT LONGITUDES/CCSQ/FIG FROM 710 (+20) TO 230 DEGREES
1115     CHSZ=0.307
1116     DO 7161 LO=1,7
1117     IF(LO.EQ.1) GO TO 7161
1118     XX=110.0+((LO-1)*20.0)
1119     XXL=XX
1120     NLA=LO*2
1121     YY=ALATSF(NLA-1)
1122     XX1=XX
1123     YY1=YY
1124     CALL CONVRT(XX,YY,IEND)
1125     CALL PLOT (XX,YY,3)
1126     WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
1127     YY=ALATSF(NLA)
1128     XX=XX1
1129     YY1=YY
1130     CALL CONVRT(XX,YY,IEND)
1131     CALL PLOT (XX,YY,2)
1132     WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
1133     IF(LO.EQ.2) GO TO 7161
1134     IF (LO.EQ.7) GO TO 7161
1135     XA=XX-(2.0*CHSZ)
1136     YA=YY - 0.5
1137     IF (LO.EQ.5) XXL=170.0
1138     IF (LO.EQ.6) XXL=150.0
1139     CALL NUMBER(XA,YA,CHSZ,XXL,0.0,-1)
1140     XAW=999.0
1141     YAW=999.0
1142     IF(LO.EQ.3.OR.LO.EQ.4)CALL SYMBOL(XAW,YAW,CHSZ,' W',0.0,2)
1143     IF(LO.GE.5.OR.LO.EQ.6)CALL SYMBOL(XAW,YAW,CHSZ,' E',0.0,2)
1144     7161 CONTINUE
1145     C LABEL LAT'S
1146     ALO=170.0
1147     DO 7176 NL=1,2
1148     ALA=30.0 + ((NL-1)*10)
1149     ALA1=ALA
1150     IF(NL.EQ.1) CLO=133.5

```

```

1151     IF(NL.EQ.2) CLO=123.0
1152     CALL CONVRT(CLO,ALA,IEND)
1153     WRITE(6,6072)CLO,CLA,ALA1
1154     CLA=ALA-(0.5*CHSZ)
1155     CLO=XLN+0.1
1156     CALL NUMBER(CLO,CLA,CHSZ,ALA1,0.0,-1)
1157     WRITE(6,6072)CLO,CLA,ALA1
1158     IF (NL.LT.3)CALL SYMBOL(999.0,999.0,CHSZ,' N',0.0,2)
1159     7176 CONTINUE
1160     C
1161     XX=0.5
1162     YY=YYLEN-0.3
1163     HH=0.2
1164     CALL SYMBOL(XX,YY,HH,'PERMANENT ',0.0,10)
1165     C
1166     C
1167     C* DRAW BATHYMETRY
1168     C
1169     WRITE(6,9901)NRECS1,NRECS2,NRECS3,NRECS4,(NDATA(NN),NN=1,4)
1170     IFLBLR=1
1171     CALL BATHY(0,1,IFLBLR,BATH,NDATA,DEPTH)
1172     C
1173     C
1174     C PLOT SCALE ARROW (1 INCH = 10CM/SEC)
1175     XP=XLN*0.72
1176     YP=YYLEN-0.7
1177     UVAL=10.0*FACTOR
1178     CALL AVECTR(XP,YP,UVAL,0.0)
1179     XP=XP-0.4
1180     YP=YP+0.1
1181     CALL SYMBOL(XP,YP,0.2,10H10 CM/SEC ,0.,10)
1182     XP=XLN*0.63
1183     YP=YYLEN
1184     CALL SYMBOL(XP,YP,0.2,22HSURFACE OCEAN CURRENTS ,0.0,22)
1185     YP=YP-0.3
1186     XP=XP-0.35
1187     CALL SYMBOL(XP,YP,0.2,27H( GEOSTROPHIC COMPONENT) ,0.0,27)
1188     C
1189     CALL COAST(0,1,IC,C1,C2,C3)
1190     CALL FIXER(IFCOP)
1191     964 CONTINUE
1192     C CALL RTIME(11)
1193     C
1194     C
1195     C-----
1196     C-----
1197     C
1198     C DO PLOT NUMBER 5 (WIND+THERM+GEOS OCEAN SURFACE CURRENTS-0/200M
1199     C
1200     C-----
1201     C-----
1202     C
1203     IF(PLOCS.LT.1.0) GO TO 965
1204     C
1205     C COMPUTE AND SCALE DOWN ARROWS IN OCEAN CURRENT VECTOR PLOT

```

```

1206     FACTOR=.25
1207     DO 2954 I=1,MM2
1208     DO 2955 J=1,MM2
1209     CSPEED(I,J)=SQRT(OCUS(I,J)*OCUS(I,J)+OCVS(I,J)*OCVS(I,J))
1210     IF(LS3(I,J).EQ.0)OCUS(I,J)=0.0
1211     IF(LS3(I,J).EQ.0)OCVS(I,J)=0.0
1212     POCU(I,J)=OCUS(I,J)*FACTOR
1213     POCV(I,J)=OCVS(I,J)*FACTOR
1214     2955 CONTINUE
1215     2954 CONTINUE
1216     C
1217     CALL PRINT1(CSPEED,11)
1218     C
1219     C ZERO OUT EVERY OTHER ROW IND COL.
1220     C TO REDUCE THE NUMBER OF ARROWS IN PLOT
1221     IF (EVGRPT.EQ.1.0) GO TO 975
1222     DO 995 JR=1,104,2
1223     DO 995 IR=1,40
1224     POCU(IR,JR)=0.0
1225     POCV(IR,JR)=0.0
1226     995 CONTINUE
1227     DO 996 IR=1,40,2
1228     DO 996 JR=1,104
1229     POCU(IR,JR)=0.0
1230     POCV(IR,JR)=0.0
1231     996 CONTINUE
1232     DO 997 IR=1,40
1233     POCU(IR,104)=0.0
1234     POCV(IR,104)=0.0
1235     997 CONTINUE
1236     DO 998 JR=1,104
1237     POCU(40,JR)=0.0
1238     POCV(40,JR)=0.0
1239     998 CONTINUE
1240     975 CONTINUE
1241     C
1242     CALL PLOT(1.,1.,-3)
1243     CALL DATFIX(POCU,OCUU)
1244     CALL DATFIX(POCV,OCVV)
1245     CALL NEWPEN(1)
1246     CALL VECTOR(OCUU,OCVV)
1247     CALL NEWPEN(2)
1248     C
1249     C CALL CONTOR(0,1,900.,1600.,8.,16.,16.,1,CSLP)
1250     CALL GRIDER(102,38)
1251     C CALL GRIDER(0,0)
1252     CALL NUMMER
1253     C
1254     C PLOT LATITUDE LINES
1255     DO 6141 IPK=1,NCC
1256     CALL PLOT (PLO(IPK),PLA(IPK),IPEN(IPK))
1257     6141 CONTINUE
1258     C
1259     CALL NEWPEN(2)
1260     C

```

```

1261 C PLOT LONGITUDES/CCSQ/FIG FROM 610 (+20) TO 230 DEGREES
1262 CHSZ=0.307
1263 DO 6161 LO=1,7
1264 IF(LO.EQ.1) GO TO 6161
1265 XX=110.0+((LO-1)*20.0)
1266 XXL=XX
1267 NLA=LO*2
1268 YY=ALATSF(NLA-1)
1269 XX1=XX
1270 YY1=YY
1271 CALL CONVRT(XX,YY,IEND)
1272 CALL PLOT (XX,YY,3)
1273 WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
1274 YY=ALATSF(NLA)
1275 XX=XX1
1276 YY1=YY
1277 CALL CONVRT(XX,YY,IEND)
1278 CALL PLOT (XX,YY,2)
1279 WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
1280 IF(LO.EQ.2) GO TO 6161
1281 IF (LO.EQ.7) GO TO 6161
1282 XA=XX-(2.0*CHSZ)
1283 YA=YY - 0.5
1284 IF (LO.EQ.5) XXL=170.0
1285 IF (LO.EQ.6) XXL=150.0
1286 CALL NUMBER(XA,YA,CHSZ,XXL,0.0,-1)
1287 XAW=999.0
1288 YAW=999.0
1289 IF(LO.EQ.3.OR.LO.EQ.4)CALL SYMBOL(XAW,YAW,CHSZ,' W',0.0,2)
1290 IF(LO.GE.5.OR.LO.EQ.6)CALL SYMBOL(XAW,YAW,CHSZ,' E',0.0,2)
1291 6161 CONTINUE
1292 C LABEL LAT'S
1293 ALO=170.0
1294 DO 6176 NL=1,2
1295 ALA=30.0 + ((NL-1)*10)
1296 ALA1=ALA
1297 IF(NL.EQ.1) CLO=133.5
1298 IF(NL.EQ.2) CLO=123.0
1299 CALL CONVRT(CLO,ALA,IEND)
1300 WRITE(6,6072)CLO,CLA,ALA1
1301 CLA=ALA-(0.5*CHSZ)
1302 CLO=XLN+0.1
1303 CALL NUMBER(CLO,CLA,CHSZ,ALA1,0.0,-1)
1304 WRITE(6,6072)CLO,CLA,ALA1
1305 IF (NL.LT.3)CALL SYMBOL(999.0,999.0,CHSZ,' N',0.0,2)
1306 6176 CONTINUE
1307 C
1308 XX=0.5
1309 YY=YYLEN-0.3
1310 HH=0.2
1311 CALL SYMBOL(XX,YY,HH,'SEA LEVEL PRESSURE (MB) ',0.0,26)
1312 CALL NUMBER(999.0,999.0,HH,FYR,0.0,-1)
1313 CALL SYMBOL(999.0,999.0,HH,' - ',0.0,3)
1314 CALL NUMBER(999.0,999.0,HH,FMO,0.0,-1)
1315 CALL SYMBOL(999.0,999.0,HH,' - ',0.0,3)

```

```

1316     CALL NUMBER(999.0,999.0,HH,FDY,0.0,-1)
1317     C
1318     C
1319     C
1320     C* DRAW BATHYMETRY
1321     C
1322     WRITE(6,9901)NRECS1,NRECS2,NRECS3,NRECS4,(NDATA(NN),NN=1,4)
1323     IFLBLR=1
1324     CALL BATHY(0,1,IFLBLR,BATH,NDATA,DEPTH)
1325     C
1326     C
1327     C PLOT SCALE ARROW (1 INCH = 10CM/SEC)
1328     XP=XLN*0.72
1329     YP=YYLEN-0.7
1330     UVAL=10.0*FACTOR
1331     CALL AVECTR(XP,YP,UVAL,0.0)
1332     XP=XP-0.4
1333     YP=YP+0.1
1334     CALL SYMBOL(XP,YP,0.2,10H10 CM/SEC ,0.,10)
1335     XP=XLN*0.63
1336     YP=YYLEN
1337     CALL SYMBOL(XP,YP,0.2,22HSURFACE OCEAN CURRENTS ,0.0,22)
1338     YP=YP-0.3
1339     XP=XP-0.35
1340     CALL SYMBOL(XP,YP,0.2,27H(WIND+THERM+GEOS COMPONENT) ,0.0,2
1341     C
1342     CALL COAST(0,1,IC,C1,C2,C3)
1343     CALL FIXER(IFCOP)
1344     C
1345     C
1346     965 CONTINUE
1347     IF(PLSOCR.LT.1.0) GO TO 9000
1348     C
1349     C READ FILE OF TOTAL CURRENTS AFTER ROTATION
1350     DO 94 J=1,104
1351     DO 94 I=1,5
1352     I2=I*8
1353     I1=I2-7
1354     READ(54,113,END=3110)(OCUS(IK,J),IK=I1,I2),KYR5,MO5,KDY5
1355     94 CONTINUE
1356     224 DO 95 J=1,104
1357     DO 95 I=1,5
1358     I2=I*8
1359     I1=I2-7
1360     READ(54,113,END=3110)(OCVS(IK,J),IK=I1,I2),KYR5,MO5,KDY5
1361     95 CONTINUE
1362     225 CONTINUE
1363     IF(NDYPL.EQ.1) DY1=KDY5
1364     IF(NDYPL.EQ.1) AMO1=MO5
1365     IF(NDYPL.EQ.1) YR1=KYR5
1366     FDY=KDY5
1367     FMO=MO5
1368     FYR=KYR5
1369     CALL RTIME(12)
1370     113 FORMAT(8F8.2,3I3,A3)

```

```

1371 C
1372 C
1373 IF (KPLT.LT.NOPLT) GO TO 9001
1374 C
1375 C
1376 C-----
1377 C-----
1378 C
1379 C DO PLOT NUMBER 6 (WIND+THERM+GEOS OCEAN SURFACE CURRENTS-0/200M
1380 C WITH NEAR COAST EFFECT VECTOR ROTATION
1381 C
1382 C-----
1383 C-----
1384 C
1385 3110 CONTINUE
1386 C
1387 C
1388 C COMPUTE AND SCALE DOWN ARROWS IN OCEAN CURRENT VECTOR PLOT
1389 FACTOR=.25
1390 DO 3954 I=1,NN2
1391 DO 3955 J=1,MM2
1392 CSPEED(I,J)=SQRT(OCUS(I,J)*OCUS(I,J)+OCVS(I,J)*OCVS(I,J))
1393 IF(LS3(I,J).EQ.0)OCUS(I,J)=0.0
1394 IF(LS3(I,J).EQ.0)OCVS(I,J)=0.0
1395 POCU(I,J)=OCUS(I,J)*FACTOR
1396 POCV(I,J)=OCVS(I,J)*FACTOR
1397 3955 CONTINUE
1398 3954 CONTINUE
1399 C
1400 C
1401 C
1402 CALL PRINT1(CSPEED,11)
1403 C
1404 C ZERO OUT EVERY OTHER ROW IND COL.
1405 C TO REDUCE THE NUMBER OF ARROWS IN PLOT
1406 IF (EVGRPT.EQ.1.0) GO TO 976
1407 DO 994 JR=1,104,2
1408 DO 994 IR=1,40
1409 POCU(IR,JR)=0.0
1410 POCV(IR,JR)=0.0
1411 994 CONTINUE
1412 DO 993 IR=1,40,2
1413 DO 993 JR=1,104
1414 POCU(IR,JR)=0.0
1415 POCV(IR,JR)=0.0
1416 993 CONTINUE
1417 976 CONTINUE
1418 DO 992 IR=1,40
1419 POCU(IR,104)=0.0
1420 POCV(IR,104)=0.0
1421 992 CONTINUE
1422 DO 991 JR=1,104
1423 POCU(40,JR)=0.0
1424 POCV(40,JR)=0.0
1425 991 CONTINUE

```

```

1426 C
1427 C
1428 C
1429 CALL PLOT(1.,1.,-3)
1430 CALL DATFIX(POCU,OCUU)
1431 CALL DATFIX(POCV,OCVV)
1432 CALL NEWPEN(1)
1433 CALL VECTOR(OCUU,OCVV)
1434 CALL NEWPEN(2)
1435 C
1436 IF(IPLSLP.EQ.1)CALL CONTOR(0,1,900.,1600.,8.,16.,16.,0,CSLP)
1437 CALL GRIDER(102,38)
1438 C CALL GRIDER(0,0)
1439 C
1440 C
1441 C* DRAW BATHYMETRY
1442 C
1443 WRITE(6,9901)NRECS1,NRECS2,NRECS3,NRECS4,(NDATA(NN),NN=1,4)
1444 IFLBLR=1
1445 CALL BATHY(0,1,IFLBLR,BATH,NDATA,DEPTH)
1446 C
1447 C
1448 C PLOT LATITUDE LINES
1449 DO 5141 IPK=1,NCC
1450 CALL PLOT (PLO(IPK),PLA(IPK),IPEN(IPK))
1451 5141 CONTINUE
1452 C
1453 CALL NEWPEN(2)
1454 C
1455 C PLOT LONGITUDES/CCSQ/FIG FROM 510 (+20) TO 230 DEGREES
1456 CHSZ=0.307
1457 DO 5151 LO=1,7
1458 IF(LO.EQ.1) GO TO 5151
1459 XX=110.0+((LO-1)*20.0)
1460 XXL=XX
1461 NLA=LO*2
1462 YY=ALATSF(NLA-1)
1463 XX1=XX
1464 YY1=YY
1465 CALL CONVRT(XX,YY,IEND)
1466 CALL PLOT (XX,YY,3)
1467 WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
1468 YY=ALATSF(NLA)
1469 XX=XX1
1470 YY1=YY
1471 CALL CONVRT(XX,YY,IEND)
1472 CALL PLOT (XX,YY,2)
1473 WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
1474 IF(LO.EQ.2) GO TO 5151
1475 IF (LO.EQ.7) GO TO 5151
1476 XA=XX-(2.0*CHSZ)
1477 YA=YY - 0.5
1478 IF (LO.EQ.5) XXL=170.0
1479 IF (LO.EQ.6) XXL=150.0
1480 CALL NUMBER(XA,YA,CHSZ,XXL,0.0,-1)

```



```

1481     XAW=999.0
1482     YAW=999.0
1483     IF(LO.EQ.3.OR.LO.EQ.4)CALL SYMBOL(XAW,YAW,CHSZ,' W',0.0,2)
1484     IF(LO.GE.5.OR.LO.EQ.6)CALL SYMBOL(XAW,YAW,CHSZ,' E',0.0,2)
1485     5151 CONTINUE
1486     C LABEL LAT'S
1487     ALO=170.0
1488     DO 5176 NL=1,2
1489     ALA=30.0 + ((NL-1)*10)
1490     ALA1=ALA
1491     IF(NL.EQ.1) CLO=133.5
1492     IF(NL.EQ.2) CLO=123.0
1493     CALL CONVRT(CLO,ALA,IEND)
1494     WRITE(6,6072)CLO,CLA,ALA1
1495     CLA=ALA-(0.5*CHSZ)
1496     CLO=XLN+0.1
1497     CALL NUMBER(CLO,CLA,CHSZ,ALA1,0.0,-1)
1498     WRITE(6,6072)CLO,CLA,ALA1
1499     IF (NL.LT.3)CALL SYMBOL(999.0,999.0,CHSZ,' N',0.0,2)
1500     5176 CONTINUE
1501     C
1502     XX=0.5
1503     YY=YYLEN-0.3
1504     HH=0.2
1505     CALL SYMBOL(XX,YY,HH,' ',0.0,1)
1506     IF(IPLSLP.EQ.1)CALL SYMBOL(XX,YY,HH,'SEA LEVEL PRESSURE (MB)
1507     '0.0,26)
1508     CALL NUMBER(999.0,999.0,HH,FYR,0.0,-1)
1509     CALL SYMBOL(999.0,999.0,HH,' - ',0.0,3)
1510     CALL NUMBER(999.0,999.0,HH,FMO,0.0,-1)
1511     CALL SYMBOL(999.0,999.0,HH,' - ',0.0,3)
1512     CALL NUMBER(999.0,999.0,HH,FDY,0.0,-1)
1513     C
1514     C
1515     C PLOT SCALE ARROW (1 INCH = 10CM/SEC)
1516     XP=XLN*0.55
1517     YP=YYLEN
1518     CALL SYMBOL(XP,YP,0.2,22HSURFACE OCEAN CURRENTS ,0.0,22)
1519     YP=YP-0.3
1520     XP=XP-0.45
1521     CALL SYMBOL(XP,YP,0.2,31H(WIND + THERM + GEOS COMPONENT),0.0,
1522     YP=YP-0.25
1523     CALL SYMBOL(XP,YP,0.2,'(X ',0.0,2)
1524     CALL NUMBER(999.0,999.0,0.2,WFAC,0.0,2)
1525     CALL SYMBOL(999.0,999.0,0.2,' , X ',0.0,4)
1526     CALL NUMBER(999.0,999.0,0.2,TFAC,0.0,2)
1527     CALL SYMBOL(999.0,999.0,0.2,' , X ',0.0,4)
1528     CALL NUMBER(999.0,999.0,0.2,ODFAC,0.0,2)
1529     CALL SYMBOL(999.0,999.0,0.2,' )',0.0,1)
1530     XP=XP+1.0
1531     YP=YP-0.4
1532     UVAL=10.0*FACTOR
1533     CALL AVECTR(XP,YP,UVAL,0.0)
1534     XP=XP-0.4
1535     YP=YP+0.1

```

```

1536     CALL SYMBOL(XP,YP,0.2,10H10 CM/SEC ,0.,10)
1537     C
1538     CALL COAST(0,1,IC,C1,C2,C3)
1539     CALL FIXER(IFCOP)
1540     CALL RTIME(14)
1541     9000 CONTINUE
1542     C
1543     C
1544     C
1545     C
1546     9001 CONTINUE
1547     C-----
1548     C-----
1549     C
1550     C DO PLOT NUMBER 7 (PROGRESSIVE VECTORS)
1551     C
1552     C-----
1553     C-----
1554     C
1555         IF (PROGV.LT.1.0) GO TO 11111
1556     C DO PLOT OF PROGRESSIVE VECTORS
1557     C
1558         NDYPL=0
1559     966 CONTINUE
1560         NDYPL=NDYPL+1
1561     C
1562     C READ FILES OF PROGRESSIVE VECTORS
1563         IJEND=152
1564         IJEND=24
1565         IJEND=66
1566         IJEND=33
1567         IJEND=1
1568         DO 3115 L=1,IJEND
1569             READ(60,4104,END=967)IDY9,KK,KYR,IMO,IDY,I,J,GU1(L),GV1(L),GL
1570             '01,GU2(L),GV2(L),GLA2,GLO2,WFAC,TFAC,DDFAC
1571     4104 FORMAT(2I3,3I2,2I3,8F7.3,3F3.1)
1572             IF (IDY9.GT.2.AND.IDY9.LT.180) GO TO 4118
1573             WRITE(6,4105)IDY9,KK,KYR,IMO,IDY,I,J,GU1(L),GV1(L),GLA1,GLO1,
1574             'GU2(L),GV2(L),GLA2,GLO2,WFAC,TFAC,DDFAC
1575     4118 CONTINUE
1576     C
1577     C
1578     C HANDLE OFF MAP POINTS
1579         IF(GV1(L).GE.38.99)IOFFMP(L)=1
1580         IF(GV1(L).LE.1.01)IOFFMP(L)=1
1581         IF(GU1(L).GE.102.99)IOFFMP(L)=1
1582         IF(GU1(L).LE.1.01)IOFFMP(L)=1
1583         IF(GV2(L).GE.38.99)IOFFMP(L)=1
1584         IF(GV2(L).LE.1.01)IOFFMP(L)=1
1585         IF(GU2(L).GE.102.99)IOFFMP(L)=1
1586         IF(GU2(L).LE.1.01)IOFFMP(L)=1
1587     C
1588         IF(IOFFMP(L).EQ.1) GO TO 3115
1589         IPLALL=1
1590         IPLALL=0

```

```

1591     IF(IDY9.GT.NDAYS) GO TO 967
1592 C    OO 3127 NXK=16,96,8
1593 C    IF(J.EQ.NXK) GO TO 3115
1594     3127 CONTINUE
1595     IF(IPLALL.EQ.1) GO TO 4129
1596     DO 4120 LK=1,IIEND
1597     IF (IDY9.GT.0.AND.IDY9.LT.180) GO TO 4119
1598 C    WRITE(6,4108)LK,IIENO,I,J,IPV(LK),JPV(LK)
1599     4119 CONTINUE
1600     IF(I.EQ.IPV(LK).AND.J.EQ.JPV(LK)) GO TO 4129
1601     4108 FORMAT(' TEST I J ',6I5)
1602     4120 CONTINUE
1603     GO TO 3115
1604     3125 IOTHER=0
1605     GO TO 3115
1606     4129 CONTINUE
1607 C
1608     IF (IDY9.GT.2.AND.IDY9.LT.180) GO TO 4106
1609     WRITE(6,4105)IDY9,KK,KYR,IMO,IDY,I,J,GU1(L),GV1(L),GLA1,GLO1,
1610     'GU2(L),GV2(L),GLA2,GLO2,WFAC,TFAC,DDFAC
1611     4105 FORMAT (2I4,3I2,2I4,8F10.3,3F5.1)
1612     4106 CONTINUE
1613     X1=(GU1(L)-1.0)*(XLN/103.0)
1614     IF (IDY9.EQ.1) YR1=KYR
1615     IF (IDY9.EQ.1) MO1=IMO
1616     IF (IDY9.EQ.1) DY1=IDY
1617     FYR=KYR
1618     FMO=IMO
1619     FDY=IDY
1620     Y1=(YLN-1.0)-((GV1(L)-1.0)*((YLN-1.0)/39.0))
1621     X2=(GU2(L)-1.0)*(XLN/103.0)
1622     Y2=(YLN-1.0)-((GV2(L)-1.0)*((YLN-1.0)/39.0))
1623     X1=X1+1.0
1624     Y1=Y1+1.0
1625     X2=X2+1.0
1626     Y2=Y2+1.0
1627 C    IF(LS3(I,J).EQ.0) GO TO 3119
1628 C    IF(IDY9.EQ.1)CALL SPCSYM(X1,Y1,0.307,42,0.0,-1)
1629     IF(IDY9.EQ.1)CALL DOT(X1,Y1,0.050)
1630     3119 CONTINUE
1631 C    IF(IDY9.LE.2)WRITE(6,6062)IDY9,L,X1,Y1,X2,Y2
1632 C    WRITE(6,6062)IDY9,L,X1,Y1,X2,Y2
1633     6062 FORMAT(2I5,4F10.3)
1634 C    CALL PVECTR(X1,Y1,X2,Y2,OCUS(I,J),OCVS(I,J))
1635     CALL PVECTR(X1,Y1,X2,Y2,2.0,2.0)
1636     IF(FDY.EQ.1.0)CALL NUMBER(X2,(Y2+0.1),CHSZ,FMO,0.0,-1)
1637     3115 CONTINUE
1638 C
1639     IF (IDY9.GT.2.AND.IDY9.LT.175) GO TO 4117
1640 C    WRITE(6,6032)NDYPL, IDY9
1641     6032 FORMAT(' PROGRESSIVE VECTOR PLOT DAY NO. ',2I5)
1642     4117 CONTINUE
1643     GO TO 966
1644     967 CONTINUE
1645 C

```

```

1646 C
1647 C
1648 C   CALL PRINT1(CSPEED,11)
1649 C
1650 C   ZERO OUT EVERY OTHER ROW IND COL.
1651 C   TO REDUCE THE NUMBER OF ARROWS IN PLOT
1652     IF (EVGRPT.EQ.1.0) GO TO 776
1653     DO 884 JR=1,104,2
1654     DO 884 IR=1,40
1655     POCU(IR,JR)=0.0
1656     POCV(IR,JR)=0.0
1657   884 CONTINUE
1658     DO 883 IR=1,40,2
1659     DO 883 JR=1,104
1660     POCU(IR,JR)=0.0
1661     POCV(IR,JR)=0.0
1662   883 CONTINUE
1663     DO 882 IR=1,40
1664     POCU(IR,104)=0.0
1665     POCV(IR,104)=0.0
1666   882 CONTINUE
1667     DO 881 JR=1,104
1668     POCU(40,JR)=0.0
1669     POCV(40,JR)=0.0
1670   881 CONTINUE
1671   776 CONTINUE
1672 C
1673 C
1674 C
1675     CALL PLOT(1.,1.,-3)
1676 C   CALL DATFIX(POCU,OCUU)
1677 C   CALL DATFIX(POCV,OCVV)
1678     CALL NEWPEN(1)
1679 C   CALL VECTOR(OCUU,OCVV)
1680     CALL NEWPEN(2)
1681 C
1682 C   CALL CONTOR(0,1,900.,1600.,8.,16.,16.,1,CSLP)
1683     CALL GRIDER(102,38)
1684 C   CALL GRIDER(0,0)
1685 C
1686 C
1687 C   PLOT LATITUDE LINES
1688     DO 1141 IPK=1,NCC
1689     CALL PLOT (PLO(IPK),PLA(IPK),IPEN(IPK))
1690   1141 CONTINUE
1691 C
1692     CALL NEWPEN(2)
1693 C
1694 C   PLOT LONGITUDES/CCSQ/FIG FROM 110 (+20) TO 230 DEGREES
1695     CHSZ=0.307
1696     DO 1161 LO=1,7
1697     IF(LO.EQ.1) GO TO 1161
1698     XX=110.0+((LO-1)*20.0)
1699     XXL=XX
1700     NLA=LO*2

```

```

1701     YY=ALATSF(NLA-1)
1702     XX1=XX
1703     YY1=YY
1704     CALL CONVRT(XX,YY,IEND)
1705     CALL PLOT (XX,YY,3)
1706     WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
1707     YY=ALATSF(NLA)
1708     XX=XX1
1709     YY1=YY
1710     CALL CONVRT(XX,YY,IEND)
1711     CALL PLOT (XX,YY,2)
1712     WRITE(6,4007)LO,NLA,IEND,XX1,YY1,XX,YY
1713 4007  FORMAT(3I5,4F10.4)
1714     IF(LO.EQ.2) GO TO 1161
1715     IF (LO.EQ.7) GO TO 1161
1716     XA=XX-(2.0*CHSZ)
1717     YA=YY - 0.5
1718     IF (LO.EQ.5) XXL=170.0
1719     IF (LO.EQ.6) XXL=150.0
1720     CALL NUMBER(XA,YA,CHSZ,XXL,0.0,-1)
1721     XAW=999.0
1722     YAW=999.0
1723     IF(LO.EQ.3.OR.LO.EQ.4)CALL SYMBOL(XAW,YAW,CHSZ,' W',0.0,2)
1724     IF(LO.GE.5.OR.LO.EQ.6)CALL SYMBOL(XAW,YAW,CHSZ,' E',0.0,2)
1725     1161 CONTINUE
1726 C LABEL LAT'S
1727     ALO=170.0
1728     DO 1176 NL=1,2
1729     ALA=30.0 + ((NL-1)*10)
1730     ALA1=ALA
1731     IF(NL.EQ.1) CLO=133.5
1732     IF(NL.EQ.2) CLO=123.0
1733     CALL CONVRT(CLO,ALA,IEND)
1734     WRITE(6,6072)CLO,CLA,ALA1
1735     CLA=ALA-(0.5*CHSZ)
1736     CLO=XLN+0.1
1737     CALL NUMBER(CLO,CLA,CHSZ,ALA1,0.0,-1)
1738     WRITE(6,6072)CLO,CLA,ALA1
1739 6072  FORMAT (' LABEL LATS ',3F10.4)
1740     IF (NL.LT.3)CALL SYMBOL(999.0,999.0,CHSZ,' N',0.0,2)
1741     1176 CONTINUE
1742 C
1743     CALL NUMBER
1744     CALL SYMBOL(0.,YYLEN,.2,'DAILY PROGRESSIVE VECTOR DISTANCE',0
1745     XX=2.0
1746     YY=YYLEN-0.3
1747     HH=0.2
1748     CALL SYMBOL(XX,YY,HH,'FROM ',0.0,7)
1749     CALL NUMBER(999.0,999.0,HH,YR1,0.0,-1)
1750     CALL SYMBOL(999.0,999.0,HH,' - ',0.0,3)
1751     CALL NUMBER(999.0,999.0,HH,MO1,0.0,-1)
1752     CALL SYMBOL(999.0,999.0,HH,' - ',0.0,3)
1753     CALL NUMBER(999.0,999.0,HH,DY1,0.0,-1)
1754     YY=YY-0.3
1755     CALL SYMBOL(XX,YY,HH,' TO ',0.0,7)

```

```

1756      CALL NUMBER(999.0,999.0,HH,FYR,0.0,-1)
1757      CALL SYMBOL(999.0,999.0,HH,'-',0.0,3)
1758      CALL NUMBER(999.0,999.0,HH,FMO,0.0,-1)
1759      CALL SYMBOL(999.0,999.0,HH,'-',0.0,3)
1760      CALL NUMBER(999.0,999.0,HH,FDY,0.0,-1)
1761      C
1762      C* DRAW BATHYMETRY
1763      C
1764      WRITE(6,9901)NRECS1,NRECS2,NRECS3,NRECS4,(NDATA(NN),NN=1,4)
1765      IFLBLR=1
1766      CALL BATHY(0,1,IFLBLR,BATH,NDATA,DEPTH)
1767      C
1768      C
1769      C PLOT SCALE ARROW (1 INCH = 10CM/SEC)
1770      XP=XLN*0.72
1771      YP=YYLEN-0.7
1772      UVAL=10.0*FACTOR
1773      C CALL AVECTR(XP,YP,UVAL,0.0)
1774      XP=XP-0.4
1775      YP=YP+0.1
1776      C CALL SYMBOL(XP,YP,0.2,10H10 CM/SEC ,0.,10)
1777      XP=XLN*0.55
1778      YP=YYLEN
1779      CALL SYMBOL(XP,YP,0.2,22HSURFACE OCEAN CURRENTS ,0:0,22)
1780      YP=YP-0.3
1781      XP=XP-0.45
1782      CALL SYMBOL(XP,YP,0.2,31H(WIND + THERM + GEOS COMPONENT),0.0,
1783      YP=YP-0.25
1784      CALL SYMBOL(XP,YP,0.2,'(X ',0.0,2)
1785      CALL NUMBER(999.0,999.0,0.2,WFAC,0.0,2)
1786      CALL SYMBOL(999.0,999.0,0.2,' X ',0.0,4)
1787      CALL NUMBER(999.0,999.0,0.2,TFAC,0.0,2)
1788      CALL SYMBOL(999.0,999.0,0.2,' X ',0.0,4)
1789      CALL NUMBER(999.0,999.0,0.2,DOFAC,0.0,2)
1790      CALL SYMBOL(999.0,999.0,0.2,')',0.0,1)
1791      C
1792      CALL COAST(0,1,IC,C1,C2,C3)
1793      CALL FIXER(IFCOP)
1794      C
1795      C
1796      C
1797      11111 CONTINUE
1798      C
1799      C
1800      C
1801      C
1802      C
1803      CALL PLOT(0.,0.,999)
1804      END
1805      SUBROUTINE AVECTR(XI,YJ,U,V)
1806      DATA VFAC/10.0/
1807      CALL PLOT(XI,YJ,3)
1808      IF(U.EQ.0. .AND. V.EQ.0.)GO TO 200
1809      X=XI+U/VFAC
1810      Y=YJ+V/VFAC

```

```

1811      CALL PLOT(X,Y,2)
1812      Z=.09/SQRT(U*U+V*V)
1813      SI=V*Z
1814      CO=U*Z
1815      XX=SI*.3
1816      YY=CO*.3
1817      CO=X-CO
1818      SI=Y-SI
1819      CALL PLOT(CO-XX,SI+YY,2)
1820      CALL PLOT(CO+XX,SI-YY,2)
1821      CALL PLOT(X,Y,2)
1822  200  CONTINUE
1823      RETURN
1824      END
1825      SUBROUTINE PVECTR(X1,Y1,X2,Y2,U,V)
1826      DATA VFAC/10.0/
1827      CALL PLOT(X1,Y1,3)
1828  C    IF(U.EQ.0. .AND. V.EQ.0.)GO TO 200
1829  C    X=XI+U/VFAC
1830  C    Y=YJ+V/VFAC
1831      CALL PLOT(X2,Y2,2)
1832      DX=X2-X1
1833      DY=Y1-Y2
1834      DL=SQRT((DX*DX)+(DY*DY))
1835  C    DL2=DL*.3
1836      Z=6.0*DL
1837      Z=9.0*DL
1838  C    Z=4.0*DL
1839      IF (Z.GT.1.0) Z=1.0
1840      IF (Z.GT.0.2) Z=0.2
1841      SPEED=SQRT((U*U)+(V*V))
1842  C    IF (SPEED.LT.2.0) GO TO 200
1843      SI=-DY*Z
1844      CO=DX*Z
1845      SI2=SI
1846      CO2=CO
1847  C    IF (SI.GT.DL2) SI=DL2
1848  C    IF (CO.GT.DL2) CO=DL2
1849  C    IF (SI.LT.-DL2) SI=-DL2
1850  C    IF (CO.LT.-DL2) CO=-DL2
1851  C    XX=SI*.3
1852  C    YY=CO*.3
1853      XX=SI*.5
1854      YY=CO*.5
1855      CO=X2-CO
1856      SI=Y2-SI
1857  C    IF(U.EQ.10.0.AND.V.EQ.10.0)WRITE(6,222)X1,Y1,X2,Y2,DL,Z,SI,CO
1858  C    'Y,CO2,SI2
1859  222  FORMAT(12F10.3/)
1860      CALL PLOT(CO-XX,SI+YY,2)
1861      CALL PLOT(CO+XX,SI-YY,2)
1862      CALL PLOT(X2,Y2,2)
1863  200  CONTINUE
1864      RETURN
1865      END

```

```

1866      SUBROUTINE PRIAFP(FLD,KM)
1867      COMMON/BLAND/AL(24,22)
1868      COMMON/LN0SEA/SEAA(40,104)
1869      DIMENSION LAB(6),LISARY(14),NAME1(3),NAME2(3),NAME3(3)
1870      DIMENSION BLON(11,11),BLAT(11,11),XII(11,11),YJJ(11,11)
1871      DIMENSION CLON(24,24),CLAT(24,24)
1872      DIMENSION XXXX(11,11),YYYY(11,11)
1873      DIMENSION IFISH(18)
1874      DIMENSION T1(24,24),SST(24,24,12),BOTT(24,24,12)
1875      DIMENSION IA(80)
1876      DIMENSION FLD(24,24),IFLD(24,24),PRF(24,24),IAF(24,24)
1877      DATA IAP/' . '/
1878      DATA IAZP/' 0.00'/
1879      C
1880      C
1881      C KM=0, FULL UNITS, NO DECIMAL PRINTED
1882      C KM=1, DIVIDE BY 100
1883      C KM=2, DIVIDE BY 1000
1884      C KM=3, DIVIDE BY 10
1885      C KM=4, DIVIDE BY 100, PRINT 1 DECIMAL PLACE
1886      C KM=5, FULL UNITS, PRINT 1 DECIMAL PLACE
1887      C KM=6, FULL UNITS, PRINT 2 DECIMAL PLACES
1888      C KM=10, F12.2 F-ORMAT ON GRIDS2 DISK
1889      C
1890      IF (IFIRST.EQ.88) GO TO 9998
1891      C
1892      C
1893      NOUT=24
1894      MOUT=24
1895      C
1896      C GET RID OF ZERO IN PRINT (SET UP A-FORMAT ARRAY)
1897      REWIND 40
1898      DO 200 I=1,NOUT
1899      DO 200 J=1,MOUT
1900      WRITE (40,201) I,J,FLD(I,J)
1901      200 CONTINUE
1902      201 FORMAT (2I5,F5.2)
1903      REWIND 40
1904      DO 205 I=1,NOUT
1905      DO 205 J=1,MOUT
1906      READ (40,202) I,J,IAF(I,J)
1907      IF (IAF(I,J).EQ.IAZP) IAF(I,J)=IAP
1908      205 CONTINUE
1909      202 FORMAT (2I5,A5)
1910      C
1911      C
1912      C
1913      C
1914      9998 CONTINUE
1915      IF (KM-1) 1,2,3
1916      3 IF (KM-3) 2,2,6
1917      6 IF (KM-5) 7,8,999
1918      1 PRINT 10
1919      C 10 FORMAT (/,5X,10HORIG.UNITS)
1920      10 FORMAT (1H )

```



```

1921     PRINT 500,(N,N=1,24)
1922     500 FORMAT (/ ,5X,25I5)
1923     PRINT 511,(N,(FLD(N,M),M=1,24),N=1,24)
1924     511 FORMAT (/I3,2X,24F5.0)
1925     GO TO 100
1926     7 PRINT 11
1927     11 FORMAT (/ ,5X,17HUNITS DIV. BY 100)
1928     PRINT 500,(N,N=1,24)
1929     DO 12 N=1,24
1930     DO 12 M=1,24
1931     PRF(N,M)=FLD(N,M)/100.
1932     12 CONTINUE
1933     PRINT 512,(N,(PRF(N,M),M=1,24),N=1,24)
1934     512 FORMAT (/I3,2X,24F5.1)
1935     GO TO 100
1936     8 PRINT 10
1937     PRINT 500,(N,N=1,24)
1938     PRINT 512,(N,(FLD(N,M),M=1,24),N=1,24)
1939     GO TO 100
1940     999 IF (KM.NE.10) GO TO 9
1941     IDSK=71
1942     CALL RDGRID(FLD)
1943     GO TO 100
1944     9 PRINT 10
1945     PRINT 500,(N,N=1,24)
1946     WRITE (6,7083)
1947     7083 FORMAT (4X,124H*****
1948     '*****
1949     '***** )
1950     DO 591 N=1,24
1951     WRITE (6,1265) (AL(N,LL),LL=1,22)
1952     1265 FORMAT (22A6)
1953     PRINT 513,(N,(IAF(N,M),M=1,24))
1954     PRINT 525,(SEAA(N,M),M=1,24)
1955     525 FORMAT (1H+,5X,24A5)
1956     591 CONTINUE
1957     WRITE (6,7083)
1958     513 FORMAT(1H ,I2,3H **,24A5,2H**)
1959     GO TO 100
1960     2 IF (KM-2) 20,21,22
1961     20 DIL=100.
1962     PRINT 11
1963     GO TO 30
1964     21 DIL=1000.
1965     PRINT 24
1966     24 FORMAT(/ ,5X,18HUNITS DIV. BY 1000)
1967     GO TO 30
1968     22 DIL=10.
1969     PRINT 25
1970     25 FORMAT(/ ,5X,16HUNITS DIV. BY 10)
1971     30 DO 40 N=1,24
1972     DO 40 M=1,24
1973     IFLD(N,M)=FLD(N,M)/DIL
1974     40 CONTINUE
1975     PRINT 500,(N,N=1,24)

```

```

1976      PRINT 514, (N, (IFLD(N,M), M=1, 24), N=1, 24)
1977      514 FORMAT (/I3, 2X, 24I5)
1978      100 RETURN
1979      END
1980      SUBROUTINE SMTH(N, ALPHA, FL, FL2)
1981      DIMENSION TDEP(40, 104), TML(40, 104), FL(40, 104), XF(40, 104),
1982      1TCX(40, 104), TCY(40, 104), XTR2(40, 104), XTR(40, 104), TS(40, 104),
1983      2SLP(40, 104), WCX(40, 104), WCY(40, 104), IGD(40, 104)
1984      DIMENSION FL2(40, 104)
1985      BETA=1.0-ALPHA
1986      DO 2 I=1, 40
1987      DO 2 J=1, 50
1988      2 XF(I, J)=FL(I, J)
1989      DO 40 M=1, N
1990      10 DO 20 I=2, 39
1991      DO 20 J=2, 49
1992      XY=FL(I, J)
1993      A=FL(I, J+1)
1994      B=FL(I+1, J)
1995      C=FL(I, J-1)
1996      D=FL(I-1, J)
1997      IF (A.EQ.0.OR.B.EQ.0) GO TO 20
1998      IF (C.EQ.0.OR.D.EQ.0) GO TO 20
1999      XF(I, J) = (ALPHA*XY) + (BETA*((A+B+C+D)/4.0))
2000      20 CONTINUE
2001      DO 30 I=2, 39
2002      DO 30 J=2, 49
2003      FL(I, J)=XF(I, J)
2004      30 CONTINUE
2005      40 CONTINUE
2006      RETURN
2007      END
2008      SUBROUTINE PRINT1(P, NOX)
2009      DIMENSION P(40, 104), IQ(50), PQ(40, 104)
2010      199 FORMAT(1H1, 5X, 'LAND-SEA-GRID')
2011      200 FORMAT(1H1, 5X, 'TEMPERATURE AT DEPTH(CX10)')
2012      201 FORMAT(1H1, 5X, 'GEOSTROPHIC TOPOGRAPHY (DYNMX100)')
2013      202 FORMAT(1H1, 5X, 'SURFACE PRESSURE(MBX10)')
2014      203 FORMAT(1H1, 5X, 'SURFACE TEMPERATURE(CX10)')
2015      204 FORMAT(1H1, 5X, 'MODEL OUTPUT (CM/SEC) ')
2016      205 FORMAT(1H1, 5X, 'WEIGHTED AVERAGE OF TEMP 0-200M (C) X10.0')
2017      206 FORMAT (1H1, 5X, 'SURFACE WIND SPEED (M/SEC)')
2018      209 FORMAT(1H1, 5X, 'U -- COMPONENT OF WIND SPEED (M/SEC)')
2019      210 FORMAT(1H1, 5X, 'V -- COMPONENT OF WIND SPEED (M/SEC)')
2020      211 FORMAT(1H1, 5X, 'SURFACE OCEAN CURRENT SPEED (CM/SEC)')
2021      220 FORMAT (1X, 26I5)
2022      212 FORMAT(4X, 18(1X, F6.1))
2023      KR=6
2024      C DO 500 IP=1, 2
2025      GO TO (10, 20, 30, 40, 42, 44, 70, 80, 90, 100, 110)NOX
2026      10 WRITE(KR, 200)
2027      DO 11 I=1, 40 ; DO 11 J=1, 50
2028      11 PQ(I, J)=P(I, J)*10.0
2029      GO TO 50
2030      20 WRITE(KR, 201)

```

```

2031      DO 21 I=1,40      ; DO 21 J=1,50
2032      21 PQ(I,J)=P(I,J)*100.0
2033      GO TO 50
2034      30 WRITE(KR,202)
2035      DO 31 I=1,40      ; DO 31 J=1,50
2036      31 PQ(I,J)=(P(I,J)-1000)*10.0
2037      GO TO 50
2038      40 WRITE(KR,203)
2039      DO 41 I=1,40      ; DO 41 J=1,50
2040      41 PQ(I,J)=P(I,J)*10.0
2041      GO TO 50
2042      42 WRITE(KR,204)
2043      DO 43 I=1,40
2044      DO 43 J=1,50
2045      43 PQ(I,J)=P(I,J)
2046      GO TO 50
2047      44 WRITE(KR,205)
2048      DO 45 I=1,40
2049      DO 45 J=1,50
2050      45 PQ(I,J)=(P(I,J))*10.0
2051      GO TO 50
2052      70 WRITE (KR,199)
2053      DO 71 I=1,40
2054      DO 71 J=1,50
2055      71 PQ(I,J)=P(I,J)
2056      GO TO 50
2057      80 WRITE (KR,206)
2058      DO 81 I=1,40
2059      DO 81 J=1,50
2060      81 PQ(I,J)=P(I,J)
2061      GO TO 50
2062      90 WRITE(KR,209)
2063      DO 91 I=1,40
2064      DO 91 J=1,50
2065      91 PQ(I,J)=P(I,J)
2066      GO TO 50
2067      100 WRITE(KR,210)
2068      DO 101 I=1,40
2069      DO 101 J=1,50
2070      101 PQ(I,J)=P(I,J)
2071      GO TO 50
2072      110 WRITE(KR,211)
2073      DO 111 I=1,40
2074      DO 111 J=1,50
2075      111 PQ(I,J)=P(I,J)
2076      C
2077      C
2078      50 CONTINUE
2079      C
2080      C PRINT LEFT 25 COLS. THEN PRINT RIGHT 25 COLS.
2081      DO 58 KJ=1,2
2082      J2=KJ*25
2083      J1=J2-24
2084      IF(NOX.EQ.3.AND.KJ.EQ.2) WRITE(KR,202)
2085      IF(NOX.EQ.8.AND.KJ.EQ.2) WRITE(KR,206)

```

```

2086      IF(NOX.EQ.9.AND.KJ.EQ.2) WRITE(KR,209)
2087      IF(NOX.EQ.10.AND.KJ.EQ.2) WRITE(KR,210)
2088      IF(NOX.EQ.11.AND.KJ.EQ.2) WRITE(KR,211)
2089      DO 60 I=1,40
2090      DO 57 JJ=J1,J2
2091      57 IQ(JJ)=PQ(I,JJ)
2092      WRITE (KR,220) I,(IQ(J),J=J1,J2)
2093      60 CONTINUE
2094      58 CONTINUE
2095      500 CONTINUE
2096      RETURN
2097      END
2098      C*****
2099      C*          SUBROUTINE FASTCOAST
2100      C*
2101      C*****
2102      SUBROUTINE FASTCO(IFAST,ISKIP,NREC)
2103      NREC=0
2104      NIE=0
2105      NSKIP=0
2106      1 READ(70,107,END=49) XC,YC,IPEN
2107      NSKIP=NSKIP+1
2108      IF (IPEN.EQ.3.OR.NSKIP.EQ.ISKIP) GO TO 2
2109      GO TO 1
2110      2 CONTINUE
2111      NSKIP=0
2112      107 FORMAT (2F10.4,I10)
2113      C      XC=-XC
2114      CALL CONVRT(XC,YC,IE)
2115      IF (IE.GT.0) NIE=NIE+1
2116      IF (NIE.GT.0) IPEN=3
2117      IF (IE.EQ.0) NIE=0
2118      IF (NIE.GT.2) GO TO 1
2119      IF (IE.GT.0) IPEN=3
2120      IF (NREC.EQ.0) IPEN=3
2121      CALL PLOT(XC,YC,IPEN)
2122      NREC=NREC+1
2123      WRITE (71,108) XC,YC,IPEN
2124      108 FORMAT (2F10.4,I3)
2125      GO TO 1
2126      49 CLOSE(71,DISP=CRUNCH)
2127      WRITE(6,110) NREC
2128      RETURN
2129      110 FORMAT (1H1,'NUMBER OF FASTCOAST RECORDS =',I10)
2130      END
2131      C*****
2132      C*
2133      C*          SUBROUTINE CSET
2134      C*
2135      C*****
2136      SUBROUTINE CSET(NFILE,IC,C1,C2,C3)
2137      DIMENSION C1(25000),C2(25000),C3(25000)
2138      N1=1
2139      N2=0
2140      1 READ(NFILE,100,END=999)C1(N1),C2(N1),C3(N1)

```

```

2141     100 FORMAT(2F10.4,I3)
2142     N1=N1+1
2143     N2=N2+1
2144     GO TO 1
2145     999 CONTINUE
2146     IC=N2
2147     RETURN
2148     END
2149     SUBROUTINE ANGLD(X,Y,AAA)
2150     IQ1=0
2151     IQ2=0
2152     IQ3=0
2153     IQ4=0
2154     IF (X.GE.0.0.AND.Y.GE.0.0) IQ1=1
2155     IF (X.LT.0.0.AND.Y.GE.0.0) IQ2=1
2156     IF (X.LT.0.0.AND.Y.LT.0.0) IQ3=1
2157     IF (X.GE.0.0.AND.Y.LT.0.0) IQ4=1
2158     PI=3.141596
2159     RAD=PI/180.0
2160     IF (X.GE.0.0.AND.X.LT.0.000001) X=0.000001
2161     IF (X.LT.0.0.AND.X.GT.-0.000001) X=-0.000001
2162     AA=Y/X
2163     IF (IQ1.EQ.1) ANG=ATAN(AA)
2164     IF (IQ1.EQ.1) GO TO 50
2165     IF (IQ2.EQ.1) ANG=PI + ATAN(AA)
2166     IF (IQ2.EQ.1) GO TO 50
2167     IF (IQ3.EQ.1) ANG=PI + ATAN(AA)
2168     IF (IQ3.EQ.1) GO TO 50
2169     IF (IQ4.EQ.1) ANG=(2.0*PI) + ATAN(AA)
2170     50 CONTINUE
2171     AAA=ANG/RAD
2172     C WRITE(6,101)IQ1,IQ2,IQ3,IQ4,Y,X,AA,ANG,AAA
2173     101 FORMAT(4I4,SF11.5)
2174     RETURN
2175     END
2176     SUBROUTINE OUTPT(OUT,NFILE,KYR,MO,KDY,H)
2177     DIMENSION OUT(40,104),H(1)
2178     DO 50 J=1,104
2179     DO 60 I=1,5
2180     I2=I*8
2181     I1=I2-7
2182     WRITE (NFILE,100) (OUT(IK,J),IK=I1,I2),KYR,MO,KDY,H
2183     60 CONTINUE
2184     100 FORMAT (8F8.2,3I3,A3)
2185     ENDFILE 17
2186     RETURN
2187     END
2188     SUBROUTINE RTIME(N)
2189     IPC=TIME(2)
2190     IO=TIME(3)
2191     WRITE(6,120) N,IPC,IO
2192     120 FORMAT (' TIME NO.',I4,5X,'CPU TIME =',I10,5X,'I/O TIME =',I1
2193     RETURN
2194     END
2195     C*****

```

```

2196 C*      SUBROUTINE STGRID      *
2197 C*      *
2198 C* SETS UP COMMON BLOCKS TO BE USED BY ROUTINES*
2199 C* NN,MM ARE NUMBER ROWS, NUMBER COLS IN GRID *
2200 C* ALA,ALO ARE LON,LAT ARRAYS OF GRID POINTS *
2201 C* ILS IS THE LAND-SEA ARRAY *
2202 C* XL,YL ARE THE WIDTH AND HEIGHT OF THE PLOT *
2203 C*****
2204      SUBROUTINE STGRID(NN,MM,ALA,ALO,ILS,XL,YL)
2205      DIMENSION ALA(40,104),ALO(40,104),ILS(40,104)
2206      DIMENSION SLAT(40,104),SLON(40,104)
2207      COMMON/GRDCOM/CFACT,M,N,NM,XI(105),YJ(41)
2208      COMMON/SIZES/XLEN,YLEN,YYLEN,SIZ1H,SIZ1V,SIZ2H,SIZ2V,
2209 &SIZ3H,SIZ3V,SIZ4H,SIZ4V
2210      COMMON/LSEA/LS(40,104)
2211      COMMON/LATLON/ALAT(40,104),ALON(40,104)
2212      COMMON/SSYM/ISYM(5)
2213      M=MM
2214      N=NN
2215      NM=N*M
2216      XLEN=XL
2217      YLEN=YL
2218      YYLEN=YLEN-1.0
2219      SIZ1H=0.198
2220      SIZ1V=0.307
2221      SIZ2H=0.181
2222      SIZ2V=0.290
2223      SIZ3H=0.120
2224      SIZ3V=0.185
2225      SIZ4H=0.110
2226      SIZ4V=0.168
2227      DO 10 I=1,N
2228      DO 10 J=1,M
2229      ALAT(I,J)=ALA(I,J)
2230      ALON(I,J)=ALO(I,J)
2231      10 LS(I,J)=ILS(I,J)
2232      930 FORMAT (I5,15F8.3)
2233      DO 60 I=1,N
2234      60 CONTINUE
2235      ISYM(1)=40
2236      ISYM(2)=44
2237      ISYM(3)=41
2238      ISYM(4)=42
2239      ISYM(5)=32
2240      DO 20 I=1,N
2241      DO 21 J=1,M
2242      IM1=(N+1)-I
2243      IF (J.GT.1.AND.I.LT.N) GO TO 21
2244      SLON(I,J)=ALON(I,J)
2245      SLAT(I,J)=ALAT(I,J)
2246      944 FORMAT (4I5,2F10.4,I10)
2247      CALL CONVRT(SLON(I,J),SLAT(I,J),IE)
2248      YJ(IM1)=SLAT(I,J)
2249      XI(J)=SLON(I,J)
2250      21 CONTINUE

```

```

2251      20 CONTINUE
2252      RETURN
2253      END
2254      SUBROUTINE SPLINE (X,Y,IOK,VAL,DATA)
2255      DIMENSION A(4),AIJ(4)
2256      DIMENSION DATA(40,104)
2257      C
2258      C
2259      IOK=0
2260      C DO 4 PT NATURAL SPLINE INTERPOLATION FOR VALUES BETWEEN GRID PTS
2261      IA=Y
2262      JA=X
2263      C LIMIT CALCULATIONS WITHIN 1 INTERIOR POINT OF LEFT AND BOTTOM
2264      C LIMIT CALCULATIONS WITHIN 2 INTERIOR PTS OF RIGHT AND TOP
2265      IF (IA.LT.2.OR.JA.LT.2) GO TO 998
2266      IF (IA.GT.38.OR.JA.GT.62) GO TO 998
2267      DX=1.0
2268      X1=JA
2269      Y1=IA
2270      XP1=X1+1.0
2271      YP1=Y1+1.0
2272      DO 105 K=1,4
2273      JJ=JA+(K-2)
2274      DO 106 L=1,4
2275      II=IA+(L-2)
2276      AIJ(L)=DATA(II,JJ)
2277      C IF(XJ.LT.1.9)WRITE(6,450)XI,XJ,L,II,JJ,AIJ(L)
2278      106 CONTINUE
2279      C VERTICAL INTERPOLATION
2280      GPP=(AIJ(3)-(2.0*AIJ(2))+AIJ(1))/(DX*DX)
2281      GPPP1=(AIJ(4)-(2.0*AIJ(3))+AIJ(2))/(DX*DX)
2282      G1=((GPP/6.0)*(((YP1-Y)**(3.0))/DX)-(DX*(YP1-Y)))
2283      G2=((GPPP1/6.0)*(((Y-Y1)**(3.0))/DX)-(DX*(Y-Y1)))
2284      G3=(AIJ(2)*((YP1-Y)/DX))
2285      G4=(AIJ(3)*((Y-Y1)/DX))
2286      G=G1+G2+G3+G4
2287      C IF(G.LT.20.0)WRITE (66,9412) N,M,K,NK,GPP,GPPP1,G1,G2,G3,G4,G
2288      A(K)=G
2289      105 CONTINUE
2290      C HORIZONTAL INTERPOLATION
2291      GPP=(A(3)-(2.0*A(2))+A(1))/(DX*DX)
2292      GPPP1=(A(4)-(2.0*A(3))+A(2))/(DX*DX)
2293      G1=((GPP/6.0)*(((XP1-X)**(3.0))/DX)-(DX*(XP1-X)))
2294      G2=((GPPP1/6.0)*(((X-X1)**(3.0))/DX)-(DX*(X-X1)))
2295      G3=(A(2)*((XP1-X)/DX))
2296      G4=(A(3)*((X-X1)/DX))
2297      G=G1+G2+G3+G4
2298      VAL=G
2299      450 FORMAT(2F7.3,3I4,F7.3)
2300      451 FORMAT(2F7.3,I4,F7.3)
2301      452 FORMAT(3F7.3)
2302      GO TO 999
2303      998 IOK=1
2304      VAL=0.0
2305      999 RETURN

```

```
2306      END
2307      SUBROUTINE DOT(X0,Y0,SIZE)
2308      DR=0.02
2309      R=DR
2310      T=0.0
2311      DT=30.0*(3.1416/180.0)
2312      T2=360.0*(3.1416/180.0)
2313      CALL PLOT(X0,Y0,3)
2314      1 CONTINUE
2315      X=R*COS(T)
2316      Y=R*SIN(T)
2317      CALL PLOT((X0+X),(Y0+Y),2)
2318      T=T+DT
2319      IF (T.LE.T2)GO TO 1
2320      T=0.0
2321      R=R+DR
2322      IF (R.LT.SIZE) GO TO 1
2323      RETURN
2324      END
```