

1988
nr.18



hølp

havforskningsinstituttets
egg- og larveprogram

FORSKERKART.

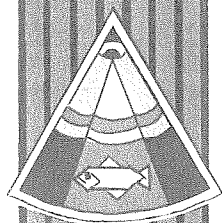
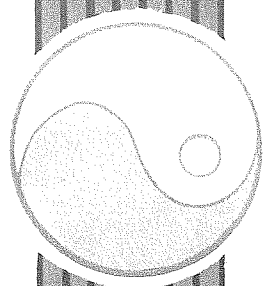
*EDB - Presentasjon
av marine data.*



FORSKERKART.

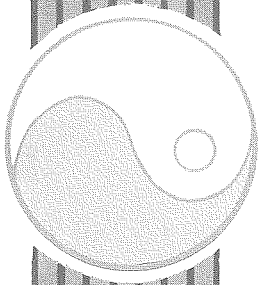
EDB-presentasjon av marine data
Brukerveiledning

1. Introduksjon
2. Les-Topografi
3. Koble-Linjer
4. Lag-Kartbase
5. Map-Library
6. ITAKS



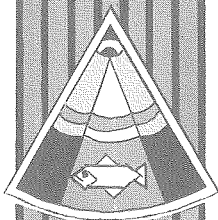
Fiskeridirektoratets
Havforskningsinstitutt

Institute of Marine Research
Directorate of Fisheries



FORSKERKART.
EDB-presentasjon av marine data.
Brukerveiledning del 1 av 6 :
Introduksjon, versjon 1.00 august 1988

Trond Westgård



Fiskeridirektoratets
Havforskningsinstitutt

Institute of Marine Research
Directorate of Fisheries

I N N H O L D S F O R T E G N E L S E

<u>Seksjon</u>	<u>Side</u>
1. Innledning	3
2. De enkelte programmoduler	5
2.1 Les-Topografi	5
2.2 Koble-Linjer	5
2.3 Lag-Kartbase	6
2.4 Map-Library	6
2.5 ITAKS	7
3. Referanser	8

1. Innledning

FORSKERKART er en samling EDB-program, skrevet i FORTRAN, som i løpet av de 7-8 siste årene er utviklet og samlet ved Havforskningsinstituttet (HI).

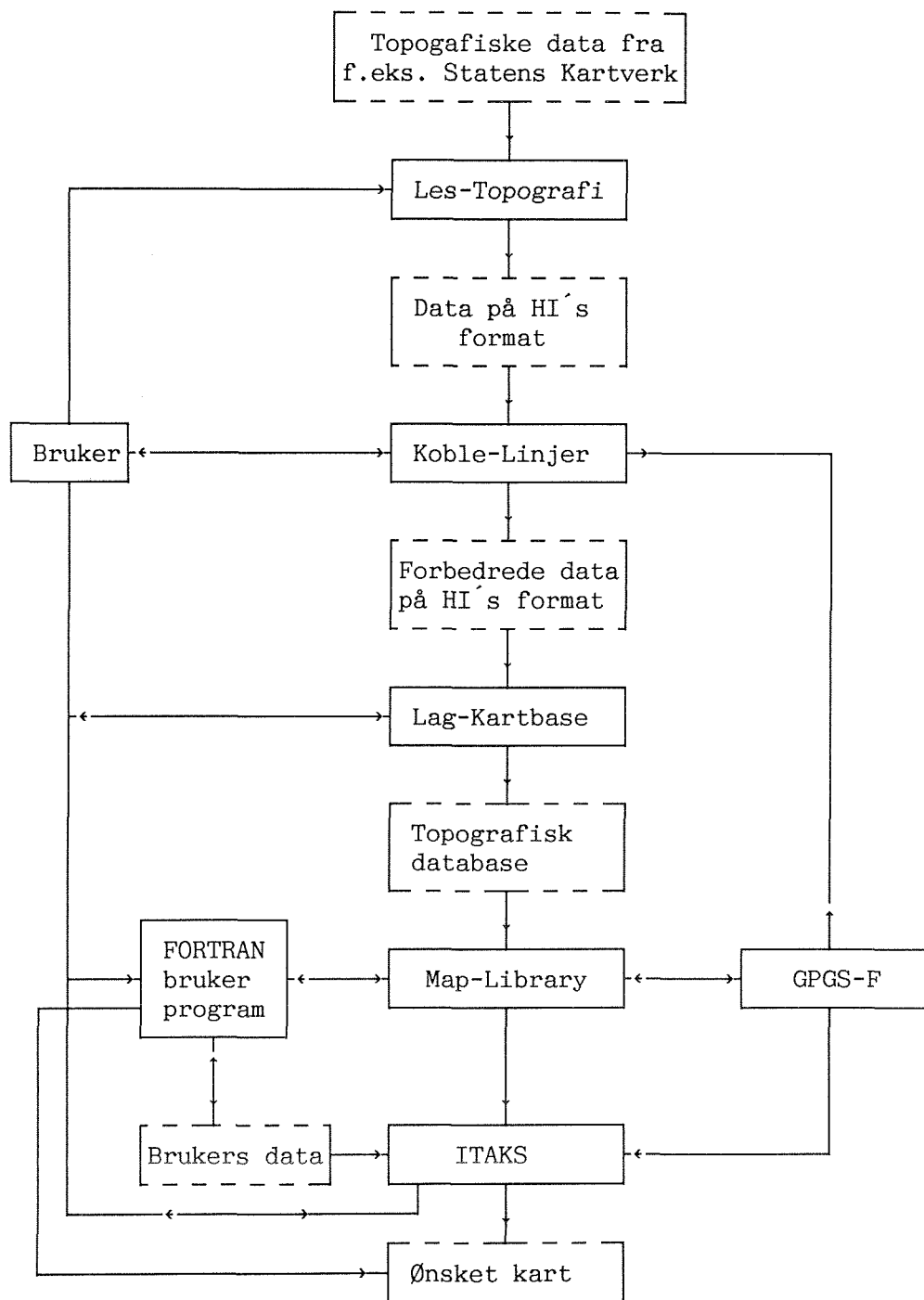
FORSKERKART er laget for å presentere marine forskningsdata på en effektiv måte. Brukeren kan gjøre dette på en rekke ulike måter avhengig av hans behov og kunnskaper. En bruker som bare er interessert i bunntopografi i et område og som behersker FORTRAN vil kunne skreddersy et program som gjør jobben, mens en sluttbruker som har svært begrenset kunnskap om EDB vil bruke det interaktive vindusorienterte programmet ITAKS for å få sine data presentert. Fig. 1 viser sammenhengen mellom modulene i FORSKERKART: Les-Topografi, KobleLinjer, Lag-Kartbase, Map-Library og ITAKS.

De ulike modulene i FORSKERKART kan brukes hver for seg. Vedlikehold og forbedring av de ulike programmodulene er også stort sett uavhengig av hverandre. Dette er grunnen til at denne manualen er bygget opp av selvstendige delrapporter i et løsbladsystem.

Mange personer har vært involvert i utviklingen av FORSKERKART som nå totalt består av flere tusen programlinjer. Flere av rutineene er skrevet av personer utenfor Havforskningsinstituttet. Jeg vil takke alle de som har gjort utviklingen av FORSKERKART mulig. Det er vanskelig å få med alle. Arvid Holm, tidligere ansatt ved UiB overlot oss kildekoden til mange nyttige program og ga oss støtte i starten. Gunnar Helle tidligere ansatt ved HI var med å skrev første versjon av Map-Library. Studentene Harald Engelstad, Sverre Vestbøstad, Veysel Soydan og Stein Holger Pettersen laget første versjon ITAKS. Tor Knutsen ved HI og Andreas Christiansen, Cap Gemini har senere entusiastisk fulgt opp videreutviklingen av FORSKERKART sammen med undertegnede.

HI er også takknemlig for det positive samarbeid vi har hatt med Nor-

ges kartverk som har overlevert oss både programvare og topografiske data, og Norsk Polarinstitutt som hadde algoritmer vi selv manglet og som passet meget godt inn med de rutiner vi selv hadde.



Figur 1. Dataflyt mellom moduler i FORSKERKART. Data er vist med stipulete bokser mens moduler som utfører logiske operasjoner på dataene er symbolisert med bokser med heltrukken linje.

Alle grafiske funksjoner i FORSKERKART bruker programpakken GPGS-F fra Norsigd, derfor kan programmene brukes på et stort antall maskintyper. Idag brukes f.eks. Map-Library både på DEC's VAX maskiner og på maskiner fra BULL og Norsk Data. Det er et ønske på sikt å skrive programmene om til den grafiske standarden 'Graphical Kernel System' (GKS), og samtidig tilpasse dem til operativsystemet UNIX. Da vil programmene på det nærmeste være uavhengig av maskintype.

2. De enkelte programmoduler

Vi skal her kort beskrive de ulike programmene i FORSKERKART. For en grundigere gjennomgang vises det til de ulike delrapportene.

2.1 Les-Topografi

Dette er et program som leser topografiske data i det format som er Norges Sjøkartverk's standard. HI mottar imidlertid dataene som geografiske koordinater og ikke som UTM-koordinater. Programmet oversetter dette formatet til et enklere format som Havforskningsinstituttet bruker internt. Hvert tema som f.eks. kan være data som beskriver et dybdenivå blir lagt ut på hver sin fil. Programmet kan enkelt modifiseres slik at andre format også kan leses. Dette gjør det mulig for en bruker å bygge opp egne tematiske data for spesielle formål.

2.2 Koble-Linjer

Nyere data som kommer fra kartinstitusjonene er ofte laget ved å bruke en scanning prosess. Resultatet av en slik prosess vil ofte være at f.eks. kystlinjer blir bygget opp av mange linjebiter som kan ligge på forskjellige steder i dataene. Det er først når alle linjebiter for kystlinjen er tegnet at en kan se hva resultatet er. En slik datastruktur betyr f.eks. at en øy ikke utgjør et lukket polygon som enkelt kan fylles med skravur eller farge på en figur laget av datama-

skinen.

I eldre data som er hånddigitalisert kan det være brudd i linjer, og holmer og skjær kan være digitalisert to ganger. Koble-Linjer sørger for at kystlinjer og dybdenivåer blir sammenhengende og at data som er dublisert blir fjernet. Brukeren kontrollerer denne prosessen interaktivt for å unngå feilkoblinger i dataene.

2.3 Lag-Kartbase

Datafilene som er resultatet av editeringsprosessen til Koble-Linjer fyller dårlig de krav som stilles til digitale kartdata. Sluttbrukere vil ofte bare være interessert i en mindre del av det geografiske området dataene dekker. Noen sluttbrukere vil bare være interessert i en grov oppløsning på sine data og greier seg med minste kartrammeutsnitt på $10^0 \cdot 10^0$ mens andre som arbeider i små geografiske områder kanskje ønsker at minste kartramme en kan hente ut skal være $1' \cdot 1'$. Lag-Kartbase gir brukeren mulighet til å velge størrelsen på den minste kartrammen det skal være mulig å hente ut og hvilket geografisk område som skal med i denne kartdatabasen. Dette betyr at de enkelte brukerne svært enkelt kan skreddersy kartdatabaser som dekkker deres behov i ulike områder. Kartdatabasen er lagret i geografiske koordinater som desimalgrader.

Tilgjengelige data ved HI pr. 1.8.1988 er kystkonturene i området (45^0N - 81^0N , 30^0V - 70^0Ø), kyst og bunnkonturer i Barentshavet syd ($69^030'\text{N}$ - $74^030'\text{N}$, 15^0V - 35^0V), og 3 kartblad i M711-serien for Masfjordområdet. Dessuten finnes kystlinjer for hele verden i relativt grov oppløsning.

2.4 Map-Library

Den topografiske kartdatabasen som ble laget med Lag-Kartbase kan leses med FORTRAN biblioteket Map-Library's rutiner. Kartene kan deretter tegnes ut på en rekke grafiske media, der grafisk skjerm og penn-plotter er det mest vanlige. Parametre som målestokk, projek-

sjonsmetode og kartutsnitt styres enkelt i rutinene. Når brukeren har laget en kartramme av ønsket område, og med ønskede topografiske nivå, kan et brukerprogram kalle opp rutiner for å tegne et skips kurslinjer, isolinjer for målte fysiske parametre, fiskeristatistikk osv. Fra egne FORTRAN-program er det mulig lese data fra mange kilder, f.eks. databaser eller vanlige filer. Det er også mulig å tegne snitt med rutiner i Map-Library. Spesialoppgaver kan utføres ved å bruke rutiner fra grafikkpakken GPGS-F direkte.

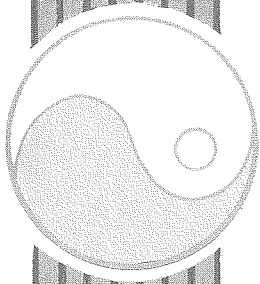
2.5 ITAKS

ITAKS er en forkortelse for Interaktiv Tegning Av Kart og Snitt. Dette er et rent sluttbrukerverktøy som det ikke forutsettes EDB-kunnskaper for å ta i bruk. Brukerne spesifiserer på en enkel måte ved å kommunisere med et sett av selvforklarende "vinduer" den jobben han vil ha gjort. Han inspiserer så resultatet på en grafisk skjerm og hvis han er fornøyd kan produktet sendes til plotter eller et annet tegnemedium. Han kan også lagre bilder i et bibliotek for senere bruk. ITAKS gir også mulighet for å lage et overlay på et kart på frihånd på den grafiske skjermen. Resultatet kan så lagres i et bibliotek eller sendes ut på plotter.

ITAKS virker i nåværende versjon bare på maskiner fra Norsk Data A/S p.g.a. skjermshåndteringssystemet FOCUS. I framtiden kan det være ønskelig å erstatte FOCUS med f.eks. X-windows som synes å bli en internasjonal standard for menysystemer, og dermed tilgjengelig på mange maskintyper.

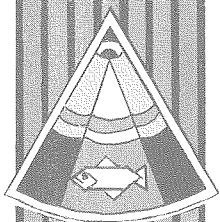
3. Referanser

- Anon. 1984. GPGS-F User's guide. 6th Edition. Tapir. Trondheim.
- Anon. 1985. FOCUS Screen Handling System. ND-60.137.5. Norsk Data a/s.
- Anon. 1986. ND FORTRAN Reference Manual. ND-60.145.7. Norsk Data a/s.
- Anon. 1986. Katalog over Norske Sjøkart og nautiske publikasjoner.
Norges Sjøkartverk, Stavanger.
- Anon. 1987. Digitale Kartverk for Barentshavet. Det Kongelige Olje-
og Energidepartement, Oslo.
- Anon. 1988. Satkart. NTNf's forskningsprogram for bruk av satellittdata
i kartlegging, måling og posisjonering. Rapport om
virksomheten i 1987. Norges Teknisk-Naturvitenskapelige
Forskningsråd, Oslo.
- Berge, T. 1987. Plotlib-1. Versjon F. Norsk Polarinstitutt.
- Engelsen, H. og Westgård, T. 1986. Brukerveiledning for ITAKS.
PS8603. Havforskningsinstituttet, Bergen.
- Taylor, J. 1976. CONMAP: A computer program for contouring of
oceanographic data. Marine Environmental Data Service.
Technical Note Nr. 12. Environment Canada. Fisheries
and Marine Service.
- Westgård, T. 1984. Map-Library. A user's guide to a subroutine
library for presentation of marine data. PS 8405.
Havforskningsinstituttet, Bergen.



FORSKERKART.
EDB-presentasjon av marine data.
Brukerveiledning del 2 av 6 :
Les-Topografi, versjon 1.00 august 1988

Andreas Christiansen
Trond Westgård



Fiskeridirektoratets
Havforskningsinstitutt

Institute of Marine Research
Directorate of Fisheries

ISBN 82-7461-004-0

I N N H O L D S F O R T E G N E L S E

<u>Seksjon</u>	<u>Side</u>
1. Innledning	3
2. Lesing av originalt magnetbånd	3
3. Dataenes lagringsformat	4
3.1 NSKV's lagringsformat	4
3.2 Havforskningsinstituttets midlertidige lagringsformat	6
4. Lesing av data fra NSKV	7

1. Innledning

De viktigste leverandører av digitale kartdata i norske havområder vil i fremtiden være Statens Kartverk og da især Norges Sjøkartverk (NSKV). For spesielle formål kan det også være aktuelt å kjøpe inn data fra kommersielle firma som Fjellanger-Widerø og andre. Havforskningsinstituttet (HI) har selv digitalisert kystlinjer for områder som er mye brukt til karttegning ved instituttet.

Programmet Les-Topografi som denne rapporten behandler, er ikke helt generelt utformet, men leser data i det format NSKV bruker og omformer dem til HI's midlertidige format. Brukerveiledningen i denne rapporten omhandler kun lesing av data fra NSKV. Rutinene er laget for ND-500 maskiner, men kan med små modifikasjoner flyttes til andre maskintyper.

Alle data forutsettes levert i grader, minutter og desimale sekunder. Disse koordinatene blir gjort om til desimale grader i HI's eget format.

2. Lesing av originalt magnetbånd

Før en kan bruke programmet Les-Topografi må programmet Les-Magnetband, skrevet av Trond Westgård, benyttes. En versjon av dette programmet finnes på bruker KART på ND-500 anlegget. Det forutsettes at brukeren vet hvor mange filer det er på båndet, hvor mange tegn hver blokk har og hvilken pakketetthet båndet har. Når brukeren har montert båndet i båndstasjonen, kan programmet startes ved å skrive:

@ND (KART)Les-Magnetband ↓

Følgende spørsmål må besvares:

- Gi antall filer på båndet : 1 ↓
- Gi filnavn dataene skal til: NSKV-ORIGINAL:DATA ↓
- Gi rekordlengde i tegn : 80 ↓

Svarene på spørsmålene vil gå fram av følgeskrivet som er med båndet fra NSKV. Dataene i filen NSKV-ORIGINAL:DATA legges ut binært for å spare plass. Filen kan med andre ord ikke inspiseres med NOTIS-WP eller en annen editor.

3. Dataenes lagringsformat

Før en går videre er det viktig å ha forståelse for hvordan dataene som mottas fra NSKV er lagret og hvordan de blir lagret ved HI.

3.1 NSKV's lagringsformat

Statens Kartverk bruker en temakode for å angi hva slags egenskap en geografisk posisjon har. Det finnes temakoder for riksgrenser, rørledninger, veinett, jernbaner, bregrenser, byer osv. Det sier seg selv at slik informasjon er unødvendig på kart ved HI. NSKV har derfor sendt oss dataene på format vist i Fig. 1 :

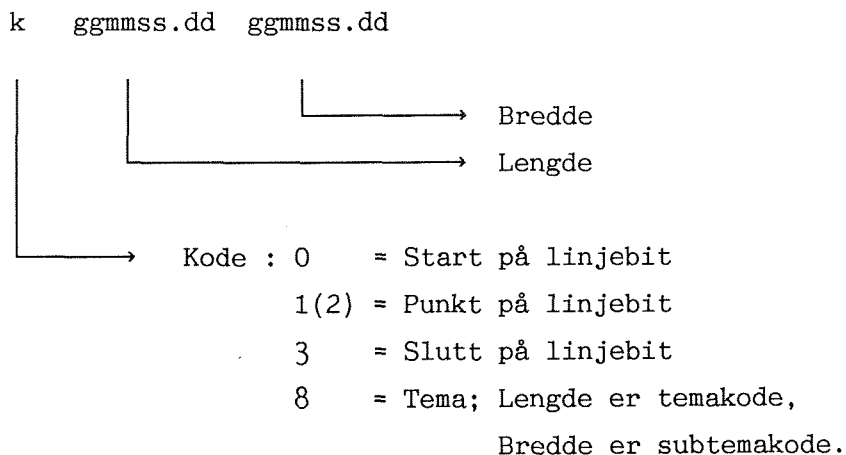


Fig. 1. Lagringsformat på magnetbånd fra Norges Sjøkartverk.

Hver record på filen NSKV-ORIGINAL:DATA består av 3 tall. Når koden, k, er 8 leses tall nr. 2 på recorden som temakode og tall nr. 3 som subtemakode, ellers er tall nr. 2 og 3 h.h.v. lengde og bredde i grader, minutter og desimale sekunder, koden k angir da start på linjebit (0), punkt på recorden (1 eller 2) eller slutt på linjebit (3).

Som eksempel vises de temakoder som er med i kartdatabasen for Barentshavet Syd (AKUP) :

Temakode	Subtemakode	Kommentar
191	1000	Kystkontur fra 1:350 000 kart (gammel serie)
195	1000	Kystkontur, Sovjet fra Norskegrense (ikke komplett)
100	2000	Kystkontur, Bjørnøya (ca. 1:600000)
200	100000	Dybdekurve, 100 meter.
200	200000	Dybdekurve, 200 meter.
.	.	.
.	.	. Dybdekurve for hver 100 meter.
.	.	.
200	2000000	Dybdekurve, 2000 meter.

3.2 Havforskningsinstituttets midlertidige lagringsformat

Kystlinjer og dybdekonturer er lagret i separate filer i HI's midlertidige format, der dataene i en fil har felles egenskap (f.eks. kystlinje eller 100-meters dybde). Posisjonene er lagret sekvensielt i desimale grader på binær form.

Hopp fra data som f.eks. beskriver en øy til data som beskriver neste øy, angis ved posisjon: 99.00, 99.00. Hver fil ender også opp med posisjonen 99.00,99.00. Sørlige bredder og vestlige lengder er negative tall.

Fil 1 <u>Kystkontur (0m)</u>	...	Fil n <u>Dybdekontur (2000m)</u>
Bredde Lengde		Bredde Lengde
Bredde Lengde		Bredde Lengde
. .		. .
. .		. .
Bredde Lengde		99.00 99.00 (hopp)
99.00 99.00 (hopp)		Bredde Lengde
Bredde Lengde		Bredde Lengde
. .		. .
. .		. .
Bredde Lengde		Bredde Lengde
99.00 99.00 (stopp)		99.00 99.00 (stopp)

Fig. 2. Havforskningsinstituttets midlertidige lagringsformat.

Det lages en fil for hver egenskap.

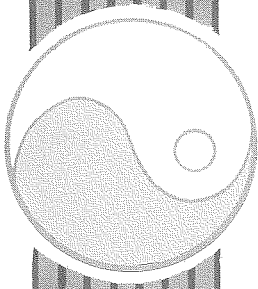
HI's endelige lagringsformat for topografiske data er beskrevet i FORSKERKART, Brukerveiledning, Del 4 : Lag-Kartbase. Før en kommer så langt må imidlertid Koble-Linjer kjøres. Dette er beskrevet i FORSKERKART, Brukerveiledning, Del 3 : Koble-Linjer.

4. Lesing av data fra NSKV

Programmet er i sin nåværende utgave skrevet for å ta imot data fra NSKV med kystlinjer og bunnkonturer i 100 m intervaller. Det kan derfor være nødvendig med noen endringer for at programmet skal virke for data fra andre kilder. For å starte programmet skriver du

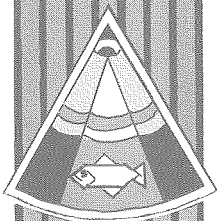
@ND (KART)Les-Topografi ↓

Programmet leser dataene i filen NSKV-ORIGINAL:DATA og legger disse ut på filene NSKV-0:DATA (kystlinjer), NSKV-100:DATA (100-m dybde) osv. Filene lagres binært for å spare plass blir .



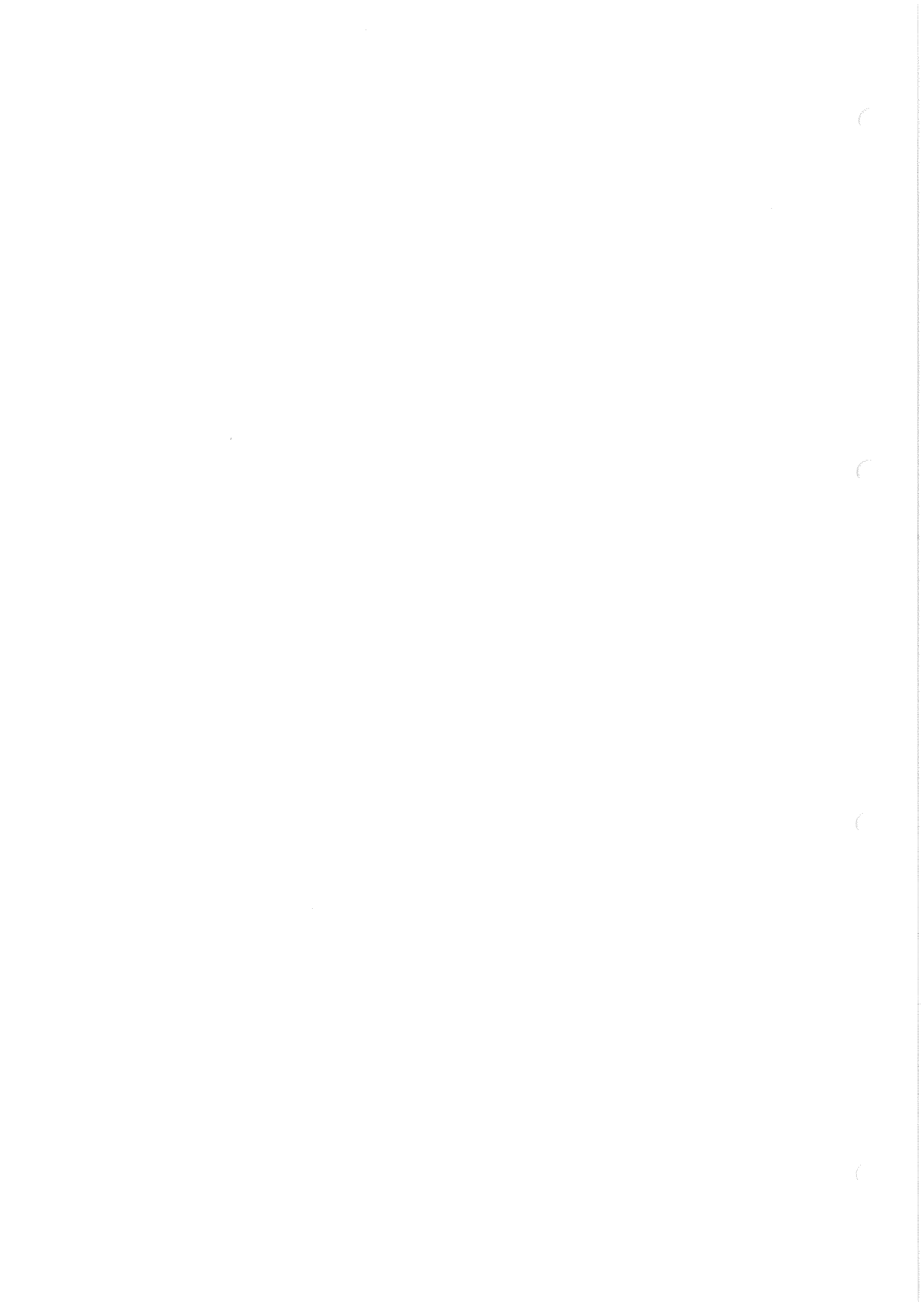
FORSKERKART.
EDB-presentasjon av marine data.
Brukerveiledning del 3 av 6 :
Koble-Linjer, versjon 1.00 august 1988

Trond Westgård
Andreas Christiansen



Fiskeridirektoratets
Havforskningsinstitutt

Institute of Marine Research
Directorate of Fisheries



I N N H O L D S F O R T E G N E L S E

<u>Seksjon</u>	<u>Side</u>
1. Innledning	3
2. Beskrivelse	4
2.1 Koble-Linjer	4
2.1.1 Justering av punkt	4
2.1.2 Kobling av linjer	5
2.2 Tegn-Raadata	7
3. System-dokumentasjon	10
3.1 Dataflyt-diagram	10
3.2 Rutine-beskrivelse	11
3.3 Data-beskrivelse	12

1. Innledning

Kartdata som er digitalisert manuelt, eller laget med en scanning/vektoreringsprosess blir ofte unøyaktige, og data for ulike segment i en kystlinje kan ofte ligge spredt i den datafilen som er resultatet av prosessen.

Dette gjør det vanskelig å strukturere de topografiske dataene i en effektiv datastruktur der linjene i de ulike topografiske nivå "vet om hverandre". En annen svakhet er når f.eks. kystlinjer ikke utgjør lukkede strukturer, så kan en ikke på en enkel måte fylle eller skravere øyer og fastland for visuelt å skille dem fra sjøen.

Koble-Linjer er et interaktivt program som utfører en slik sammenføyingsprosess. Dette skjer ved at programmet tar vare på posisjonene til start og stopp-punktene til alle linjesegmenter i originaldataene. Brukeren spesifiserer en toleranse-verdi i kilometer (ϵ), og to linjestykker kobles sammen hvis avstanden mellom endepunktene er mindre enn ϵ .

Dersom flere enn to endepunkter ligger innenfor toleransegrensen, har vi en flertydig kobling. Da vises de to linjestykkene i to "vinduer" på skjermens høyre halvdel, og brukeren får flere valgmuligheter for å rette opp feilen. Denne prosessen må ofte gjentaes med en stadig økende ϵ . Hvis en starter med å gi ϵ en stor verdi får en falske koblinger over f.eks. trange sund, tvers over øyer etc.

Programmet kan også sjekke om punkt innenfor ett og samme linjesegment ligger urimelig langt fra hverandre ("spikere"). Slike punkt kan justeres på lignende måte som flertydige koblinger.

2. Beskrivelse

2.1 Koble-Linjer

Programmet startes ved å skrive :

@ND (KART)Koble-Linjer↓

En dialog mellom bruker og program starter da opp som vist i det følgende :

Input kart fil : NSKV-100:DATA↓

Output kart fil : HI-100:DATA↓

Kommentar : Hvis "Input kart fil" ikke finnes fra før eller "Output kart fil" finnes fra før så gir programmet feilmelding. En må da sjekke filnavnene og starte på nytt.

2.1.1 Justering av punkt

En opsjon i Koble-Linjer er å justere punktene som ligger på et linjesegment slik at opplagte "spikere" på en kystlinje kan fjernes. Dette gjøres ved å lese gjennom hele "Input kart fil" og beregne middellavstand og standardavvik mellom påfølgende punkt innenfor linjesegmentene. Følgende dialog blir vist :

Vil du justere punktene på linjene (J/N) : J

Regner ut standardavvik.

XXXXXXXX records lest.

Standard avvik er YYYYYY.yy

Gi justeringsfaktor (... 3 4 5 ...) : 4

Kommentar : Linjene blir nå lest nok en gang for sjekk av de linjene der avstanden mellom to påfølgende punkt er mer enn 4 ganger standardavviket unna middelveidien. Slike punkt blir justert til midtpunktet mellom punktet før og etter i linjesegmentet.

Hvis slike linjesegment påtreffes så blir det tegnet på skjermen og brukeren får spørsmålet :

Justere punkt NNNN på linje MMMM (J/N) : J

Hvis brukeren svarer J så justeres punktet, svarer han N så kan han enten skrive ut linjesegmentet til filen KOBLE-ERROR:DATA, eller han kan gi en ny justeringsfaktor.

Når justeringsprosessen er fullført får brukeren meldingen :

Posisjonene er nå justert og skrevet til "Output filen".

Bruk denne som "Input fil" når linjene skal kobles.

Programmet stopper så, og må startes opp på nytt når linjene skal kobles.

2.1.2 Kobling av linjer

Hvis brukeren hopper over funksjonen for å justere punkt får han spørsmålet :

Akseptert toleranse i kilometer : xxx.xx

Akseptert toleranse i kilometer er variabelen € som er omtalt over. Neste spørsmål lyder :

Ignorerer tvetydige koblinger (J/N) : J

Hvis brukeren svarer J på dette spørsmålet ignoreres alle flertydige koblinger og ingen av spørsmålene angående disse vil bli stilt.

Brukeren får så kvittering på hvor mange linjesegment og geografiske posisjoner det var i "Input filen" :

NNNNN records lest.

MMMM linjesegmenter i filen

Programmet sjekker nå avstanden fra start og stopp av hvert linjesegment mot start og stopp-posisjonene til alle andre linjesegment. Brukeren får meldingene :

Sjekker linje MMMM

Hvis programmet påtreffer en ulovlig kobling kommer meldingen :

Tvetydig kobling N mellom linje MMM og linje NNN

Brukeren kan da velge å svare J på neste spørsmål d.v.s ignorere :

Ignorerer (J/N) : J

Hvis brukeren ikke ignorerer får han spørsmålene :

Fjern en linje (J/N) : J

Hvilken linje : NNN^d

Fjern siste punkter i en av linjene (J/N) : J

Hvilken linje : NNN^d

Hvor mange punkter : MM^d

Svarer brukeren N på noen av disse spørsmål, blir ikke underspørsmålene stilt. Hvis brukeren hverken vil fjerne linjer eller punkter, får han spørsmålet :

Omstart med ny toleranse (J/N) : N

Hvis brukeren ikke vil starte en ny prosess ved å minske ϵ får han spørsmålet :

Skriv til errorfil og stopp (J/N) : N

Svares det J på dette spørsmålet, vil koordinatene i de to linjesegmentene bli skrevet til filen KOBLE-ERROR:DATA på ASCII-form.

Hvis det finnes linjesegment som er koblet til flere andre tillates ikke brukeren å forsette.

2.2 Tegn-Raadata

Programmet Tegn-Raadata presenterer kartdataene før de blir "rutet opp" v.h.a. programmet Lag-Kartbase. En kan med andre ord inspisere kartdataene både før og etter at programmet Koble-Linjer har forbedret dataene. Brukeren bestemmer selv hvilket geografisk utsnitt som skal presenteres, og kan "blåse opp" et lite område for å sjekke dataenes kvalitet.

Tegn-Raadata kan også brukes til å plotte Koble-Error:Data ved problemer i Koble-Linjer.

Programmet startes ved å skrive

@ND Tegn-Raadata¹

Brukeren må svare på følgende spørsmål på skjermen :

1. Gi filnavn for kartdata : HI-100:DATA¹

2. Gi plottemedium : 2^d

3. Dataene kan ligge på filen i rekkefølgene :
1 = Breddegrad , Lengdegrad
2 = Lengdegrad , Breddegrad
Gi rekkefølge : 1^d

4. Dataene kan ligge på sekvensiell form som ASCII tegn. De kan da leses med et format eller fritt format. Dataene kan også lese binært.
Format (Fri=1,Format=2,Binær=3) : 3^d
(Gi format (Eks.: (2F9.4)) :)

5. Du kan oppgi hvor stort plottet skal være i cm.
Dataene gies i rekkefølgen (xmin,xmax,ymin,ymax)
Hvis du gir : 0.,0.,0.,0. brukes hele mediet
Gi ønsket størrelse : 0., 0., 0., 0.^d

6. Du kan oppgi hvor stort plottet skal være i grader
Dataene gies i rekkefølgen (xmin,xmax,ymin,ymax)
på formatet GGMM (f.eks 6000 eller 1230)
Hvis du gir : 0,0,0,0 brukes hele datafilen
Gi ønsket område : 1530, 2040, 4500, 4730^d

7. Skal endepunktene markeres (J/N) : J

8. Følgende kart-projeksjoner er tilgjengelige :
 - 1 - XY-koordinater
 - 2 - UTM-koordinater.
 - 3 - Mercator.
 - 4 - Kjegle.
 - 5 - Polarstereografisk.
 - 6 - Polarstereografisk m/pol i senter.Gi projeksjon : 2^d

9. Kartet skal ha et konstruksjonspunkt.
UTM bruker LONOR, ellers brukes LATOR
Hvis du gir 0, 0 beregnes et punkt.
Gi LATOR, LONOR : 0, 0

Etter at kartet er plottet stilles følgende spørsmål

10. Ny projeksjon (J/N) : N

Hvis svaret er J (ja) går en opp igjen til spørsmål 8.

11. Nytt plott (J/N) : N

Hvis svaret er J (ja) går en opp igjen til spørsmål 5., ellers (nei) avsluttes programmet.

3. System-dokumentasjon

3.1 Dataflyt-diagram

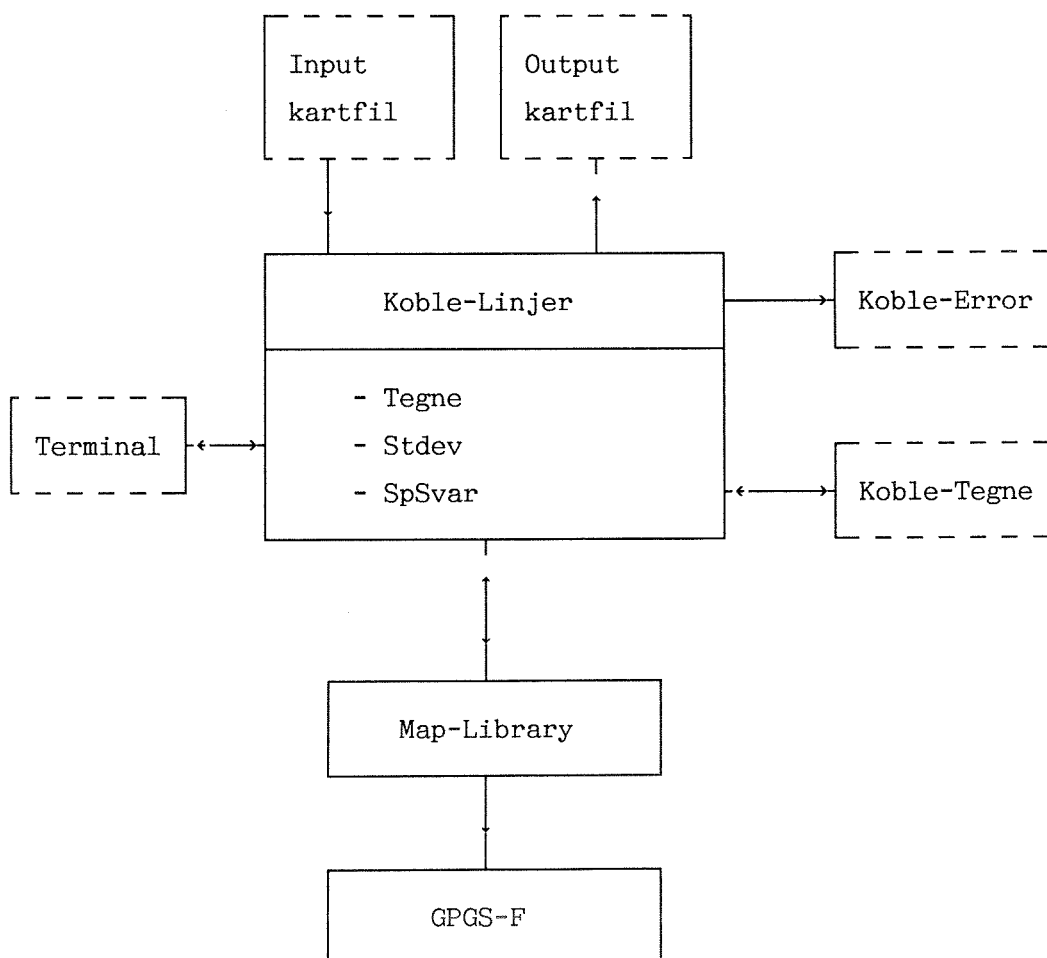


Fig. 1. Sammenheng mellom logiske rutiner og dataenheter (stiplete) i programmet Koble-Linjer.

3.2 Rutine-beskrivelse

Rutinenavn : Koble-Linjer
Rutinetype : Program
Lagret på fil : (Kart)Koble-Linjer:Symb
Beskrivelse : Hovedprogram som inneholder skjermdialog og logikk for kobling av linjeselementer.
Parametre : Ingen
Kallende rutiner : Ingen
Rutine-kall : Stdev, Tegne, Map-Library

Rutinenavn : Tegne
Rutinetype : Subrutine
Lagret på fil : (Kart)Koble-Linjer:Symb
Beskrivelse : Tegner en kartramme rundt et sett koordinater for å sjekke om punktene er korrekt plassert. Ved tegning av to linjestykker vil rammene bli plassert øverst og nederst på høyre halvdel av skjermen.
Parametre : Ist
Kallende rutiner : Koble-Linjer, StDev
Rutine-kall : Map-Library, GPGS-F

Rutinenavn : StDev
Rutinetype : Subrutine
Lagret på fil : (Kart)Koble-Linjer:Symb
Beskrivelse : Regner ut standard avvik og justerer avstanden mellom punktene i et sett koordinater. Inneholder skjermdialog.
Parametre : Ist
Kallende rutiner : Koble-Linjer
Rutine-kall : Map-Library, SpSvar, Tegne

Rutinenavn : SpSvar
Rutinetype : Subrutine
Lagret på fil : (Kart)Koble-Linjer:Symb
Beskrivelse : Stiller et spørsmål på skjerm, og returnerer svar-
et som en "upper-case character".
Parametre : Cstreng, Csvar
Kallende rutiner : Koble-Linjer, StDev
Rutine-kall : Ingen

3.3 Data-beskrivelse

Filnavn : <Input kartfil>
Lagringsformat : Binær
Beskrivelse : Sett med koordinater som beskriver linjestykker
slik en får dem fra NSKV, men på HI's format.

Filnavn : <Output kartfil>
Lagringsformat : Binær
Beskrivelse : Justerte og sammenkoplete linjestykker.

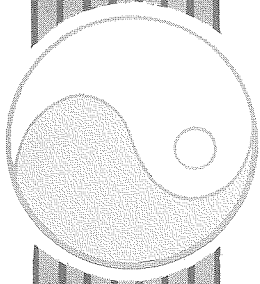
Filnavn : Koble-Error:data
Lagringsformat : Ascii
Beskrivelse : Linjestykker som hverken program eller bruker
finner en fornuftig kobling på.

Filnavn : Koble-Tegne:data
Lagringsformat : Binær
Beskrivelse : Mellomlagring av linjestykker som skal tegnes på
terminalen.

Del 3 av 6 : Koble-Linjer

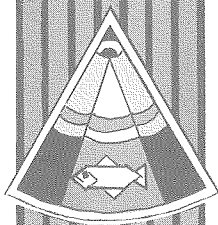
Recordformat

samtlige filer	:	Bredde	Lengde	
		Bredde	Lengde	
		.		
		.		
		.		
		Bredde	Lengde	
		Bredde	Lengde	
		99.00	99.00	(hopp)
		Bredde	Lengde	
		Bredde	Lengde	
		.		
		.		
		.		
		99.00	99.00	(slutt)



FORSKERKART.
EDB-presentasjon av marine data.
Brukerveiledning del 4 av 6 :
Lag-Kartbase, versjon 1.00 august 1988

Trond Westgård
Andreas Christiansen



Fiskeridirektoratets
Havforskningsinstitutt

Institute of Marine Research
Directorate of Fisheries

I N N H O L D S F O R T E G N E L S E

<u>Seksjon</u>	<u>Side</u>
1. Innledning.	3
2. Beskrivelse.	4
2.1 Lag-KartBase.	4
2.2 Lag-Corner.	6
2.3 Lag-Asciibase.	7
2.4 Tegn-Base.	7
3. System-dokumentasjon.	10
3.1 Dataflyt-diagram.	10
3.2 Rutine-beskrivelse.	11
3.3 Filstruktur.	14
3.3.1 Info-Fil.	14
3.3.2 Rute-Fil.	15
3.3.3 Kart-Fil.	16
4. Kartdatabaser ved Havforskningsinstituttet.	17

1. Innledning.

Topografiske data som HI har digitalisert selv eller mottatt fra andre institusjoner er lagret på sekvensielle filer, en fil for hvert topografisk nivå (egenskap). Disse filene blir behandlet først med programmene Les-Topografi og Koble-Linjer, se del 2 og 3 i denne brukerveiledningen.

Filene med topografiske data trenger ytterligere bearbeiding slik at en programmerer eller en sluttbruker raskt kan ta ut de ønskete kyst eller dybdekoter i et nærmere spesifisert geografisk område.

Programmet Lag-KartBase lagrer de topografiske data på en måte som er effektiv og kompakt. Den enkelte bruker kan med Lag-KartBase selv ta et utvalg av en større topografisk datamengde og lage en skreddersydd database for spesielle formål.

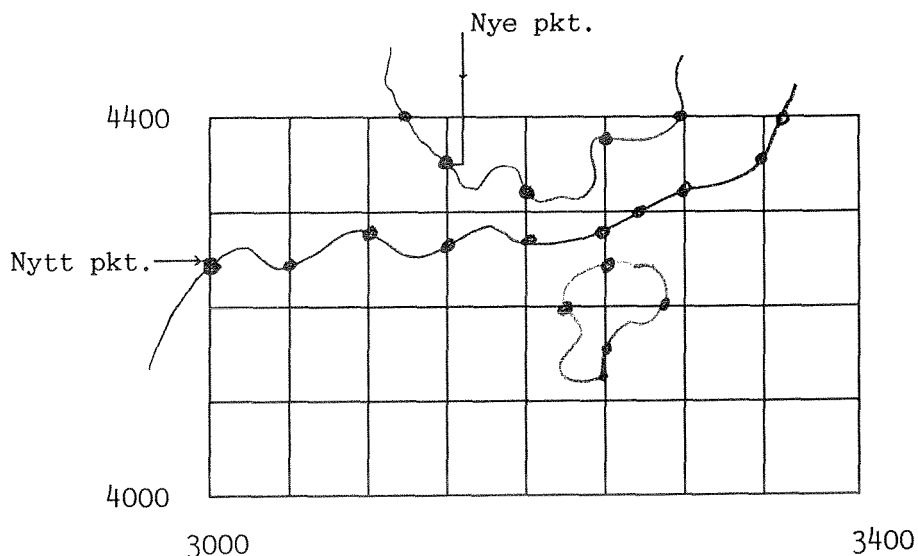


Fig. 1. Eksempel på en topografisk struktur med to nivå som Lag-KartBase kan behandle.

På Fig. 1 er hovedprinsippet i Lag-KartBase vist. Prinsippet er enkelt og går ut på at en logikk for interpolering legger inn nye geografiske

posisjoner der de topografiske dataene går ut av eller inn i det definerede "rutemønstret" eller krysser grenser mellom ruter. Den minste kartramme brukeren senere kan hente fram med den oppløsningen som er vist i Fig. 1 er 1⁰ i nord-syd retning og 30' i øst-vest retning. Den største kartrammen blir 30-34⁰ Ø og 40-44⁰ N selv om de originale dataene inneholdt informasjon utenfor dette området.

Gunnar Helle laget første versjon av Lag-KartBase. Torstein Pedersen har også bidratt sammen med forfatterne av denne brukerveiledning.

2. Beskrivelse.

2.1 Lag-KartBase.

Programmet startes ved å skrive :

@ND (Kart)Lag-KartBase^d

Dialogen mellom bruker og program er som vist under :

1. Kartbasen skal lagres på tre filer, en informasjonsfil, en pekerfil, og en kartfil som inneholder selve kartdataene (posisjonene). Basen skal gis et navn (max 10 tegn) og filene vil bli lagret som <basenavn>-INFO:MBAS
<basenavn>-RUTE:MBAS og <basenavn>-KART:MBAS
Gi navn til kartbasen (max 10 tegn) : KARTDATA^d

2. Input til Lag-Kartbase må være binære fil(er) med geografiske posisjoner i desimalgrader (y,x).

Gi minste breddegrad (GGMM (sør=neg.)) : 4500^d

Gi største breddegrad (GGMM (sør=neg.)) : 8100^d

Gi minste lengdegrad (GGMM (vest=neg.)) : -3000^d

Gi største lengdegrad (GGMM (vest=neg.)) : 7000^d

3. De endelige kartfilene skal "rutes opp" for å gi mulighet til å tegne utsnitt av totalfilen. Minste mulige "kartramme" som kan tegnes, spesifiseres ved å gi "rutens" lengde og bredde i hele antall grader og minutter.

Formatet er GGMM (f.eks. 0100 eller 0030).

Gi lengde pr. trinn i bredderetning : 0030^d

Gi lengde pr. trinn i lengderetning : 0100^d

4. Kartbasen kan når den er ferdig, inneholde flere nivåer egenskaper (f.eks. dybdekoter).

Beskriv de egenskapene du vil legge inn i denne basen (max 80 tegn, IKKE bruk kommategn) :

Kystlinjer for Barentshavet Norge EF Østersjøen Grønland og Island.^d

Gi antall egenskaper (max 21) : 1^d

Gi verdi for egenskap <1> : 0.^d

(Gi verdi for egenskap <2> :)

.

.

5. Nivå nummer nnn med verdi mmm

Gi antall input-kartfiler (max 20) : 1^d

Gi navn på kartfil <1> : KARTDATA:MAPB^d

(Gi navn på kartfil <2> :)

.

.

6. Rutefilen kan nå oppdateres med de forskjellige egenskapsnivåene.

Startverdi er nivået i kartbasenens NEDERSTE VENSTRE hjørne. Verdi -1 gir ingen oppdatering.

Er verdien ukjent eller feil verdi gis, kan rutefilen senere oppdateres med programmet Lag-Corner.

Gi verdi for kartbasens NEDERSTE VENSTRE hjørne : 100.[↓]

2.2 Lag-Corner.

Lag-Corner er et lite program som kun oppdaterer egenskapsdataene i hjørnene i rute-filen, m.a.o. det samme som siste del av programmet Lag-KartBase.

Programmet startes ved å skrive

@ND (Kart)Lag-Corner[↓]

og følgende spørsmål må besvares av brukeren :

1. Base-navn : KARTDATA[↓]
2. Bruker-navn : KART[↓]
3. Gi verdi for nederste venstre hjørne : 100.[↓]

Skal programmet kunne oppdatere rute-filen med topografisk egenskapsverdi til alle rutenes hjørner, må egenskapsverdien til kartets nederste venstre hjørne oppgis som startgverdi.

2.3 Lag-Asciibase.

Dette programmet lager ASCII-filer av de tre filene i en kartdatabase. Filene vil få samme navnet som før, men "fil-type" vil nå være :ASCII.

Programmet startes ved å skrive

@ND (Kart)Lag-Asciibase

og følgende spørsmål må besvares av brukeren :

1. Base-navn : KARTDATA
2. Bruker-navn : KART

Programmet kan være nyttig når kartdatabaser skal overføres til andre maskintyper enn Norsk Data. Et program som tilbakefører rute- og kart-filen til binær form må da finnes på mottakende maskin.

2.4 Tegn-Base.

Tegn-Base er et program som kan brukes til å teste ut kartdatabaser. Programmet tegner ut hele, eller utsnitt av kartdatabaser i valgfri projeksjon og på valgfritt plottemedium.

Programmet startes ved å skrive

@ND (Kart)Tegn-Base

Følgende spørsmål skal da besvares :

1. Gi basenavn : AKUP
2. Gi brukernavn for basen : KART

3. Denne kartbasen har følgende nivå-verdier :

.
. .
.

Hvor mange nivå ønskes : 3^d
Gi nummer på nivåene : 1,4,7^d
Gi linjetype for hvert nivå : 1,2,2^d
Gi farge for hvert nivå : 1,1,5^d

4. Gi plottemedium : 5^d

5. Du kan oppgi hvor stort plottet skal være i cm.
Dataene gies i rekkefølgen (xmin,xmax,ymin,ymax)
Hvis du gir : 0.,0.,0.,0. brukes hele mediet.
Gi ønsket størrelse : 0.,0.,0.,0.^d

6. Du kan oppgi hvor stort plottet skal være i grader
Dataene gies i rekkefølgen (xmin,xmax,ymin,ymax)
på formatet GGMM (f.eks 6000 eller 1230)
Hvis du gir : 0,0,0,0 brukes hele kartbasen.
Gi ønsket område : 0,0,0,0^d

7. Kartrammen kan markeres med grader og minutter
Dataene gies i rekkefølgen (x- og y-mellomrom)
på formatet GGMM (f.eks 0500 eller 0030)
Hvis du gir 0, 0 beregnes en markering.
Gi ønsket markering : 0500,0030^d

8. Kartrammen kan deles med streker i
lengde og bredderetning (x og y)
Hvis du gir 0, 0 blir det ingen streker.
Gi antall markeringer pr strek : 1,1^d

9. Skal kartrammen ha tekst (J/N) : J

10. Følgende kart-projeksjoner er tilgjengelige :

- 1 - XY-koordinater
- 2 - UTM-koordinater.
- 3 - Mercator.
- 4 - Kjegle.
- 5 - Polarstereografisk.
- 6 - Polarstereografisk m/pol i senter.

Gi projeksjon : 2^d

11. Kartet skal ha et konstruksjonspunkt.

UTM bruker LONOR, ellers brukes LATOR

Hvis du gir 0, 0 beregnes et punkt.

Gi LATOR, LONOR : 6900,2700^d

Etter at kartet er plottet stilles følgende spørsmål

12. Ny projeksjon (J/N) : N

Hvis svaret er J (ja) går en opp igjen til spørsmål 10

13. Nytt plott (J/N) : N

Hvis svaret er J (ja) går en opp igjen til spørsmål 1., ellers (nei) avsluttes programmet.

3. System-dokumentasjon.

3.1 Dataflyt-diagram.

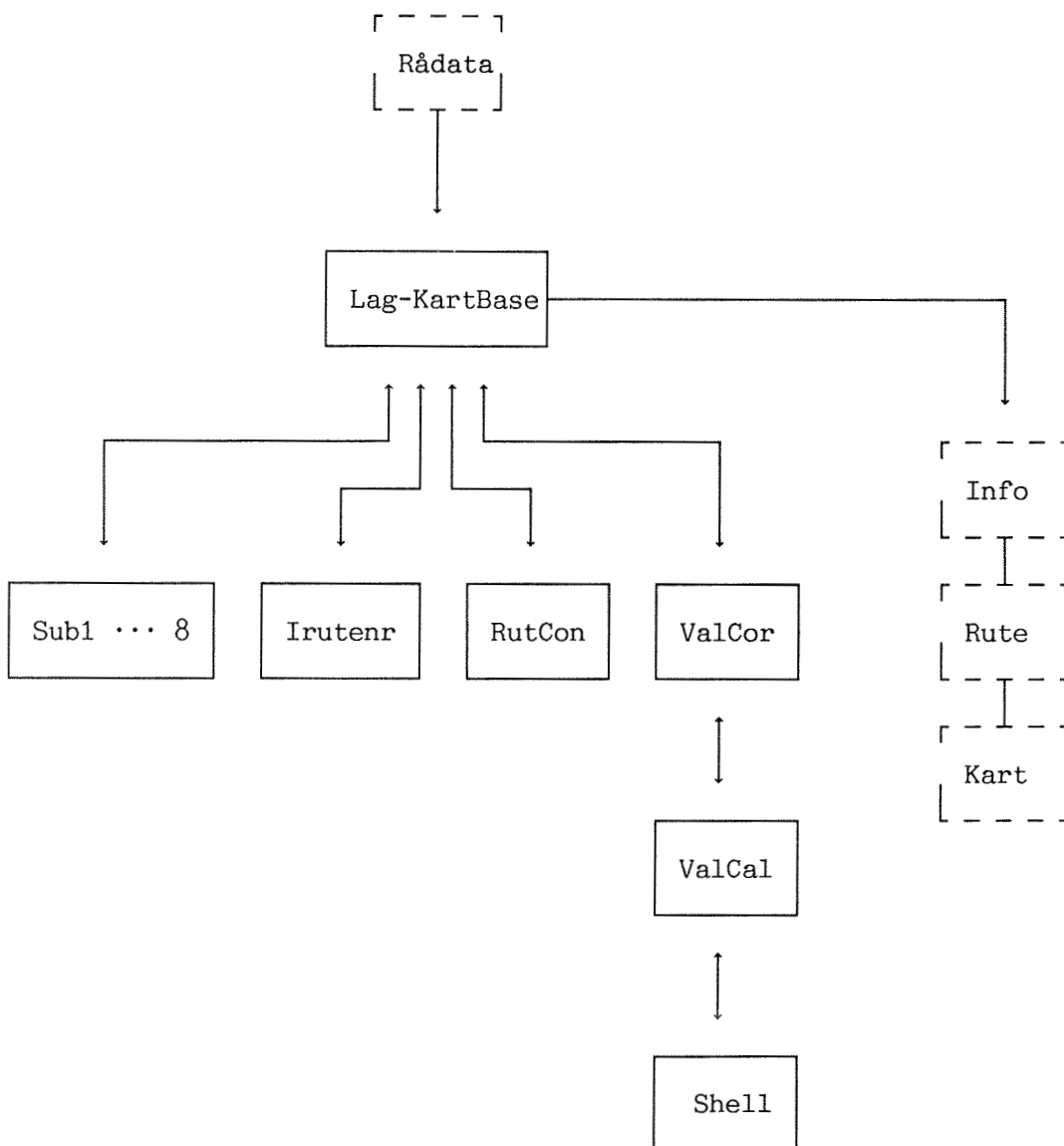


Fig. 2. Sammenheng mellom programmet Lag-KartBase, subrutiner og datafilene.

Del 4 av 6 : Lag-KartBase.

"Rådata" i Fig. 2 er filene med topografiske data, en fil for hvert nivå. Formatet på rådataene er beskrevet i dokumentasjonens del 2, Les-Topografi. Disse dataene leses av Lag-KartBase, det legges inn "kunstige" punkt på overgangen mellom rutene og dataene legges ut på tre nye filer, "Info", "Rute" og "Kart". Hvordan disse filene legges ut er nærmere beskrevet nedenfor.

3.2 Rutine-beskrivelse.

Rutinenavn : Lag-KartBase
 Rutinetype : Program
 Lagret på fil : (kart)Lag-KartBase:symb
 Beskrivelse : Hovedprogram som inneholder skjermdialogen og logikken for "oppruting" av kartgrunlaget. Programmet takler egenskapsdata og klipping i rådataene.
 Parametre : Ingen
 Kallende rutiner : Ingen
 Rutine-kall : Irutenr, Rutcon, Sub1, Sub2, ... , Sub8, ValCor

Rutinenavn : Sub1, Sub2, ... , Sub8
 Rutinetype : Subrutiner
 Lagret på fil : (Kart)Lag-KartBase:Symb
 Beskrivelse : Subrutinene genererer nye punkt i skjæringspunktene mellom to ruter. De åtte subrutinene fungerer etter følgende system

6 3 5
 2 0 1
 7 4 8

Vi har en geografisk posisjon som befinner seg i 0. Neste geografiske posisjon kan enten befinne seg i rute 0 eller i en av naborutene. Hvis neste posisjon er i en naborute aktiviseres en av rutinene Sub1..8.

Parametre : Y0, X0, Y3, X3, Y1, X1, Y2, X2,
(I0,I3,Icrut,Crossrut)
Y0,X0 og Y3,X3 er de to eksisterende punkt.
Y1,X1 og Y2,X2 lages på grensen.
Parametrene i parantes brukes kun av
Sub5 ... Sub8.

Kallende rutiner : Lag-KartBase

Rutine-kall : Ingen

Rutinenavn : Irutenr

Rutinetype : Funksjon

Lagret på fil : (Kart)Lag-KartBase:Symb

Beskrivelse : Finner rutenummer til en gitt posisjon.

Parametre : X, Y

Kallende rutiner : Lag-KartBase

Rutine-kall : Ingen

Rutinenavn : RutCon

Rutinetype : Subrutine

Lagret på fil : (Kart)Lag-KartBase:Symb

Beskrivelse : Rutinen konverterer mellom rutenummerne i en matr-
ise og rutenummerne i den samme matrisen med en
"rand" rundt.

Parametre : Ifunk, Irut0, IrutR, Istat

Kallende rutiner : Lag-KartBase

Rutine-kall : Ingen

Rutinenavn : ValCor

Rutinetype : Subrutine

Lagret på fil : (Kart)Lag-KartBase:Symb

Beskrivelse : Rutinen setter egenskapsverdi i hjørnene til hver
rute i kartdatabasen.

Parametre : Basenavn, Baseuser, Startverdi, Istat.

Kallende rutiner : Lag-KartBase

Rutine-kall : Basini, ValCal

Del 4 av 6 : Lag-KartBase.

Rutinenavn : ValCal
Rutinetype : Subrutine
Lagret på fil : (Kart)Lag-KartBase:Symb
Beskrivelse : Rutinen finner egenskaps-verdien til et hjørne i en rute.
Parametre : Rutenummer, Hjørnennummer, Startverdi, Hjørneverdi.
Kallende rutiner : ValCor
Rutine-kall : Shell

Rutinenavn : Shell
Rutinetype : Subrutine
Lagret på fil : (Kart)Lag-KartBase:Symb
Beskrivelse : Rutinen sorterer to integer-array etter stigende rekkefølge til det ene arrayet.
Parametre : Posisjon-array, verdiindex-array, antall element.
Kallende rutiner : ValCal
Rutine-kall : Ingen

3.3 Filstruktur.

Filstrukturen til en kartdatabase består av tre filer: en info-fil, en rute-fil og en kart-fil. Strukturen er hierarkisk oppbygd, og ble utviklet for å kunne lagre topografiske data mest mulig effektivt.

En kartdatabase identifiseres ved et basenavn på maksimum 10 tegn. Oppbyggingen av de forskjellige filnavnene forklares i delkapitlene om hver av filene. Disse navnene er imidlertid kun ment til internt bruk, programmerere og brukere benytter basenavnet overfor Map-Library og ITAKS.

3.3.1 Info-Fil.

I info-filen lagres nødvendig informasjon om kartdatabase. Filen er en ASCII-fil, og har navnet <basenavn>-INFO:MBAS. Denne filen kan leses med en editor, og inneholder følgende opplysninger.

1. Basenavn.
2. Egenskapsbeskrivelse (fritekst)
3. Geografisk område (N - S - V - Ø)
4. Rutelengde i grader og minutter
i bredde- og lengde-retning.
5. Antall egenskaper (NLEVELS)
6. Verdi for laveste egenskap (f.eks. 0 kyst)
7. Verdi for nest laveste egenskap (f.eks. 100 dybde)
-
-
-
- n. Verdi for høyeste egenskap (f.eks. 2000 dybde)

3.3.2 Rute-Fil.

Kartgrunnlaget i en kartdatabase er oppdelt i ruter av valgfri størrelse. I rute-filen lagres pekere fra rutene til koordinatene i kart-filen, og egenskapsverdier til hjørnene i alle rutene. Filen er en binær-fil med record-lengde $RECL = (4 + 2 \cdot NLEVELS) \cdot 4$ bytes, og har navnet <basenavn>-RUTE:MBAS.

Hver record som beskriver en rute har følgende layout :

E1, E2, E3, E4, S1, A1, S2, A2, ... , Sn, An

der :

E1 = Egenskap (topografisk nivå) i hjørne 1

E2 = Egenskap (topografisk nivå) i hjørne 2

E3 = Egenskap (topografisk nivå) i hjørne 3

E4 = Egenskap (topografisk nivå) i hjørne 4

S1 = Recordnummer hvor dataene til laveste nivå starter i "Kart-fil"

A1 = Antall posisjoner for laveste nivå i "Kart-fil"

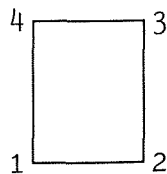
.

.

Sn = Recordnummer hvor datene til høyeste nivå starter i "Kart-fil"

An = Antall posisjoner for høyeste nivå i "Kart-fil"

De fire hjørnene i en rute er nummerert som følger :



3.3.3 Kart-Fil.

I kart-filen lagres de geografiske koordinatene til kartet. Filen er en binærfil med recordlengde RECL = 8 bytes, og har navnet <basenavn>-KART:MBAS.

Dataene lagres som bredde og lengde i desimale grader :

Bredde, Lengde

Bredde, Lengde

.

.

.

Bredde, Lengde

99.00, 99.00

(hopp)

Bredde, Lengde

Bredde, Lengde

.

.

.

4. Kartdatabaser ved Havforskningsinstituttet.

Ved HI er det pr. 1/9-88 tilgjengelig fire kartdatabaser, som alle ligger lagret på bruker KART på begge ND-500 maskinene.

- KARTDATA.

Kystkonturene i området $45^{\circ}\text{N} - 81^{\circ}\text{N}$, $30^{\circ}\text{V} - 70^{\circ}\text{Ø}$. Denne kartdatabasen har en rutestørrelse på 1° i lengde-, og $30'$ i bredde-retning. Dataene for norskekysten er digitalisert ved Jordskjelvstasjonen, UiB, mens resten av dataene er digitalisert ved HI. Kartet til KARTDATA er vist i fig. 2.

- VERDEN.

Kystkonturene for hele verden. Rutestørrelsen er 10° i lengde- og 5° i bredderetning. Verdenskartet er vist i fig. 3.

- AKUP.

Kyst- og dybdekonturer for den sørlige delen av Barentshavet ($69^{\circ}30'\text{N} - 74^{\circ}30'\text{N}$, $15^{\circ}\text{Ø} - 35^{\circ}\text{Ø}$). Ruteinndelingen er 1° i lengde- og $15'$ i bredderetning. Dypdekonturene er hver 100-meter ned til 1000 meter, samt 1500 og 2000 meter. HI har mottatt dataene fra Norges Sjøkartverk i forbindelse med gjennomføringen av AKUP-prosjektet. Kartdatabasen er vist i fig. 4.

- MASFJORDEN.

Kystkonturer for Masfjorden, $60^{\circ}30'\text{N} - 61^{\circ}03'\text{N}$, $4^{\circ}42'\text{Ø} - 5^{\circ}54'\text{Ø}$. Dette er kartdata med svært høy oppløsning, ruteinndelingen er $1'$ i både lengde- og bredderetning. Kartet er gradteigskart fra M711, serien mottatt fra Statens Kartverk. Kartdatabasen er vist i fig. 5.

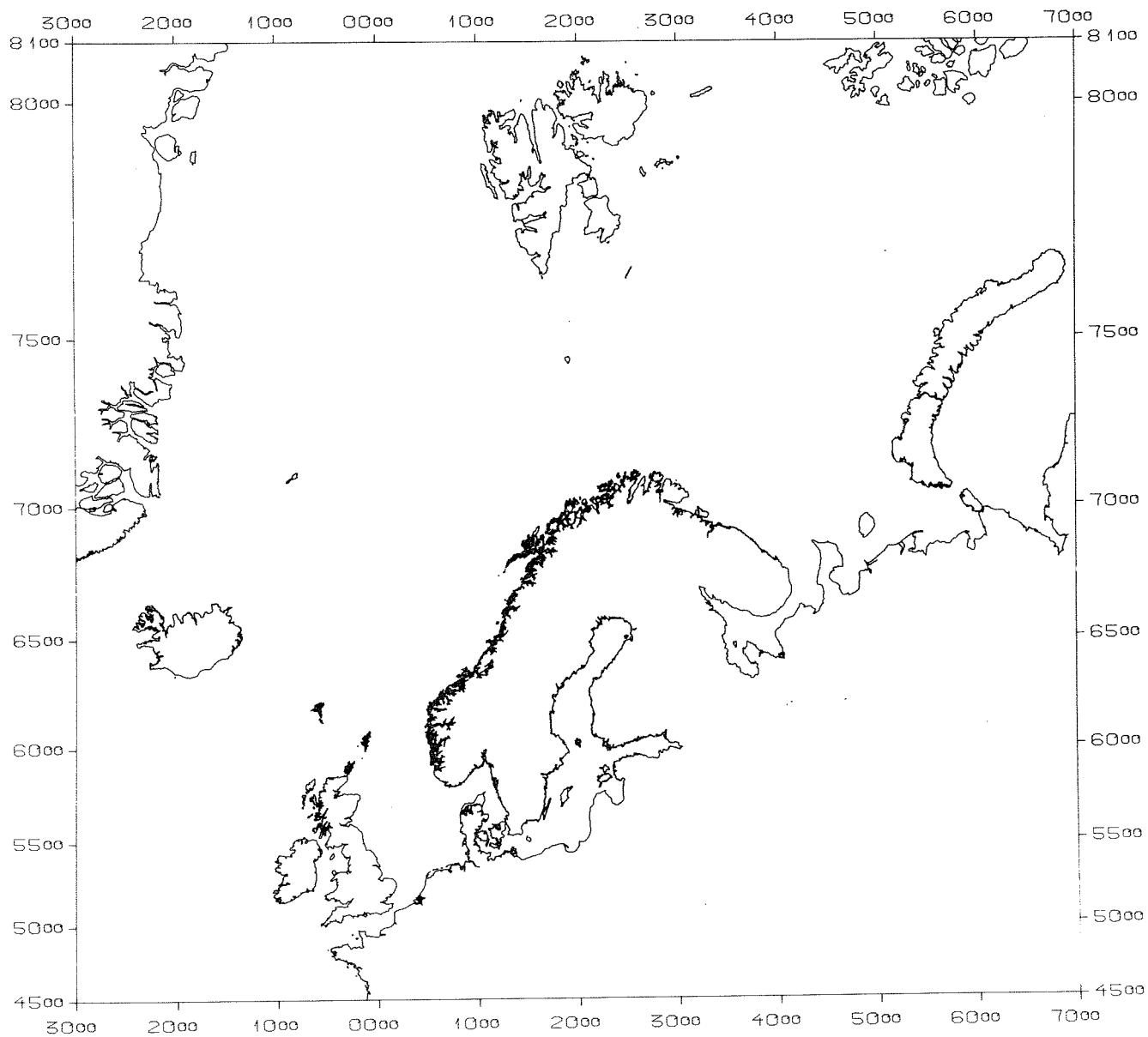


Fig. 2. KARTDATA.

Kystlinjer i området $45^{\circ}\text{N} - 81^{\circ}\text{N}$, $30^{\circ}\text{V} - 70^{\circ}\text{Ø}$.

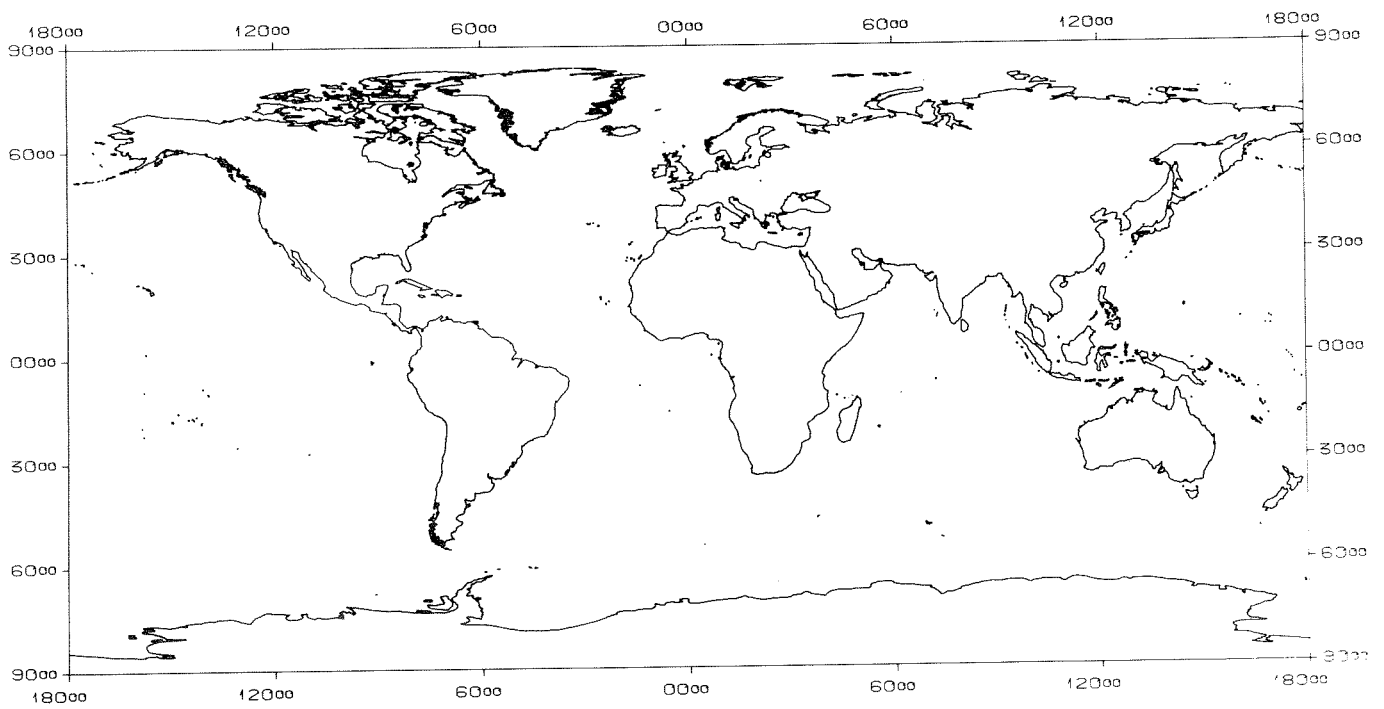


Fig. 3. VERDEN.
Kystkonturer for hele verden.

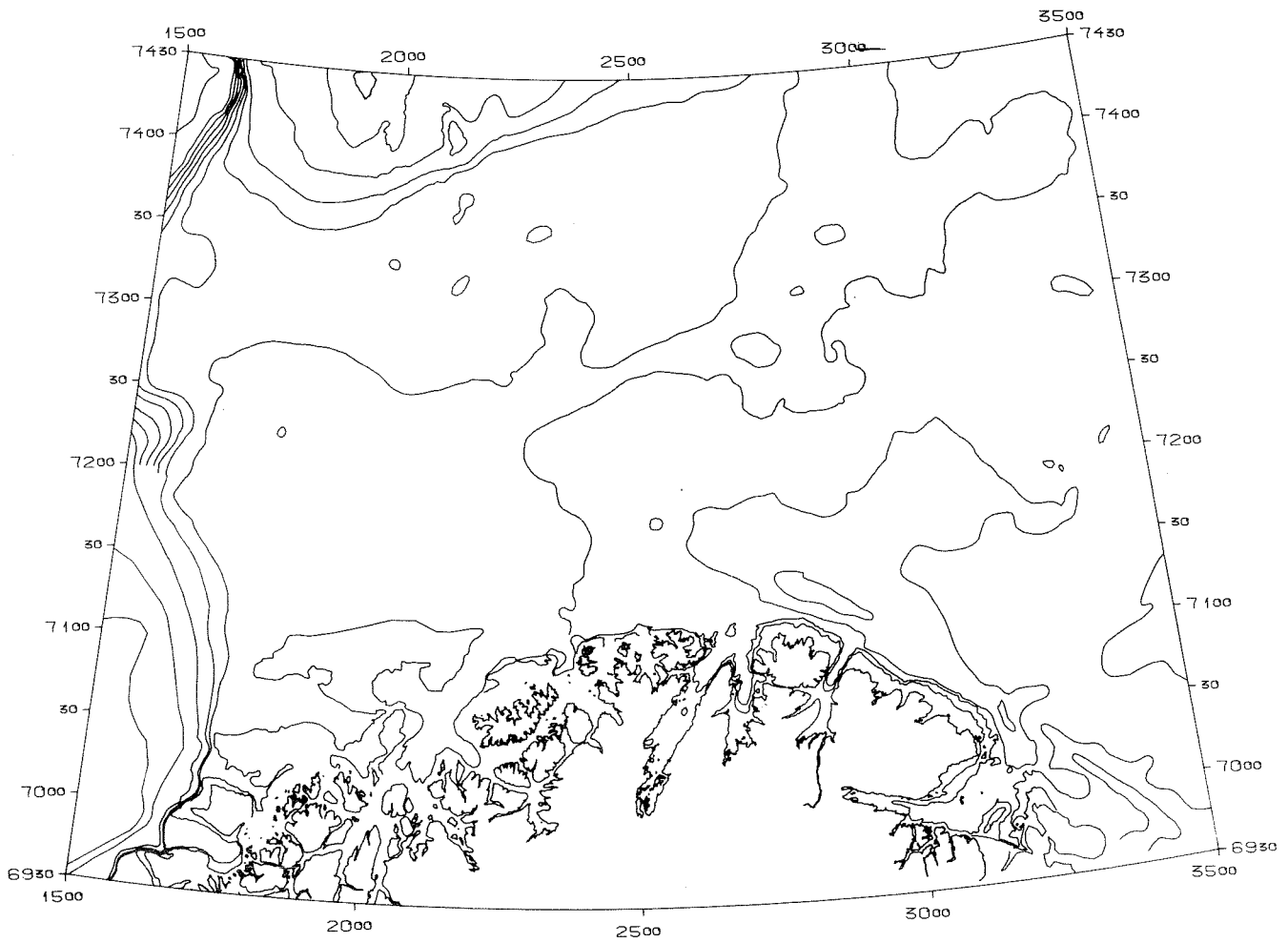


Fig. 4. AKUP.

Kyst- og dybdekonturer for sørlige deler av Barentshavet.
($69^{\circ}30'N - 74^{\circ}30'N$, $15^{\circ}E - 35^{\circ}E$)

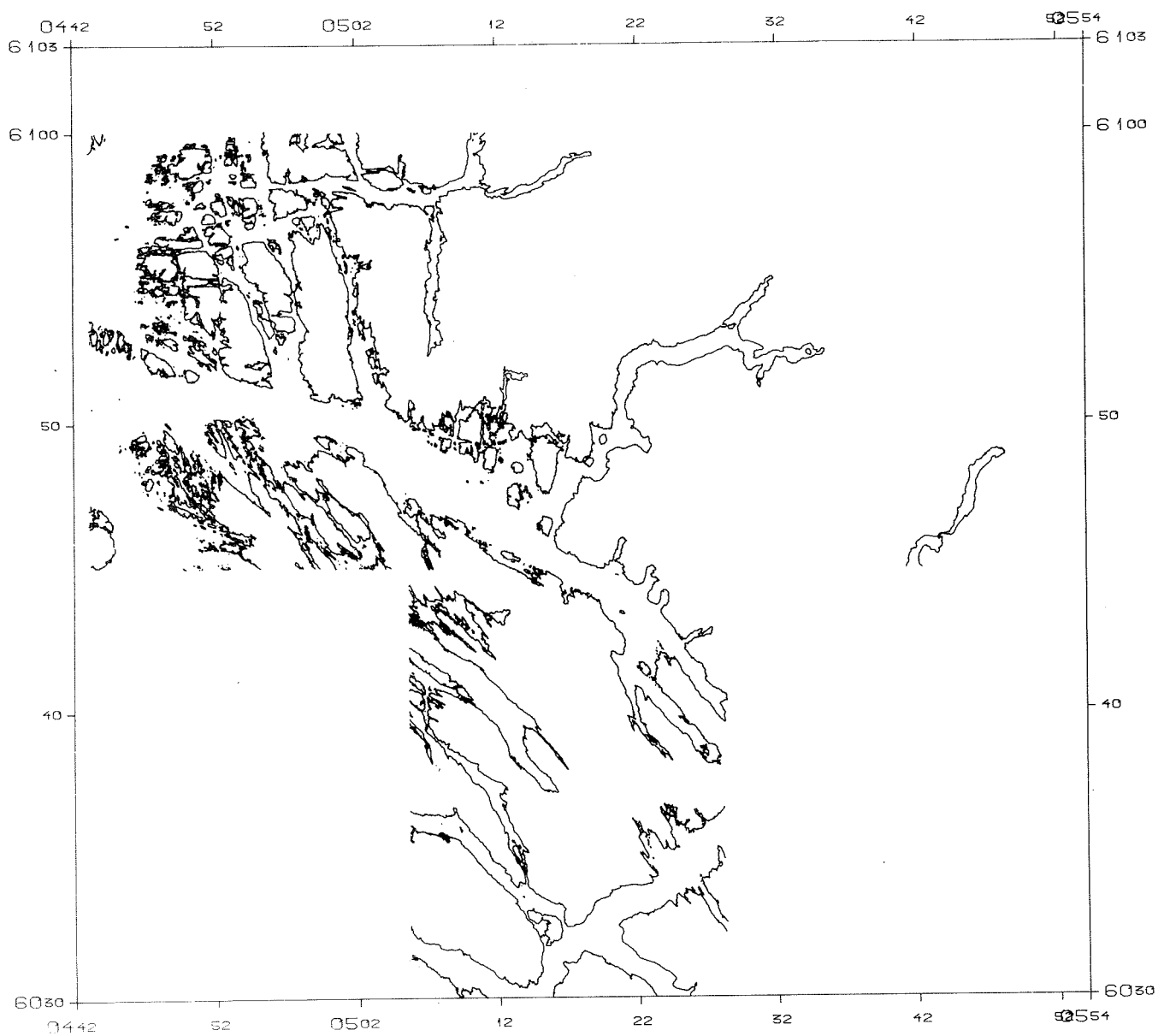
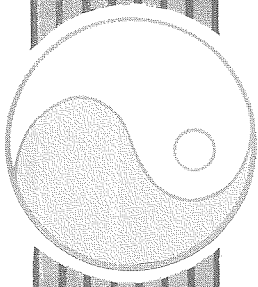


Fig. 5. MASFJORDEN.

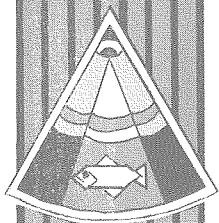
Kystkonturer for Masfjordområdet.

($60^{\circ}30'N - 61^{\circ}03'N$, $4^{\circ}42'Ø - 5^{\circ}54'Ø$)



FORSKERKART.
EDB-presentasjon av marine data
Users Guide, part 5 of 6 :
Map-Library, version 1.00 august 1988

Trond Westgård
Tor Knutsen
Andreas Christiansen



Fiskeridirektoratets
Havforskningsinstitutt

Institute of Marine Research
Directorate of Fisheries

C O N T E N T S

<u>Section</u>	<u>Page</u>
1 Introduction.	3
2 Map projections and topographical data.	5
3 Drawing of maps with Map-Library.	6
3.1 Drawing a chart frame with text.	6
3.2 Drawing of coastlines.	9
3.3 Drawing of a station net.	12
3.4 Interpolation and drawing of isolines.	15
3.5 Integration and drawing of an interpolated grid.	23
3.6 Text and symbols at specific positions.	26
3.7 Cross sections.	30
3.7.1 Horizontal presentation of stations.	30
3.7.2 Vertical presentation of stations.	35
3.8 Drawing of current-vectors.	40
3.9 Drawing of maps using colours.	43
4 Error handling in Map-Library.	44
4.1 Restrictions in use of external files.	46
5 Loading programs and plotting on different devices.	47

Section	Page
6	Geographical positions. Integer/Real degrees. 48
7	References. 48
8	Appendices. 49
8.1	Common blocks in Map-Library. 49
8.2	Error messages in Map-Library. 54
8.3	Quick reference to routines in Map-Library. 55
8.4	Reference to routines in Map-Library. 61
8.5	Filling of topographical levels 111
8.6	Graphical Kernel System (GKS). 119
8.6.1	Using Map-Library with GKS. 119
8.6.2	Deviations from standard GKS. 121
8.6.3	Compatibility with GPGS-version. 121
8.6.4	File-configuration on NORSK DATA 500. 122
8.6.5	Loading user programs. 123
8.6.6	Compiling Map-Library. 123
8.6.7	Transporting Map-Library to another system. 124
8.6.8	Program-examples. 126
8.6.9	GKS utility library. 128
8.6.9.1	GHISTA. 128
8.6.9.2	GHIEND. 128
8.6.9.3	GHIAKS. 129
8.6.9.4	GHILBL. 129
8.6.9.5	GHINB. 129
8.6.9.6	GRAFIN. 130

<u>Section</u>	<u>Page</u>
8.6.10 GKS include files.	130
8.6.11 GKS-references.	131
 Index	 133

1 Introduction.

Map-Library is a flexible tool designed for presentation of marine data. The routines can plot data on all graphical devices either as maps in one of several projections or as vertical sections. It is also possible to integrate a variable over an area to estimate the total amount present.

Map-Library uses the graphical subroutine package GPGS-F from NORSIGD (Anon., 1984) which contains all basic functions for graphical presentation of curves and alphanumeric data. To read this manual it is an advantage to have some knowledge of GPGS-F and FORTRAN.

The routines for 2-D interpolation and isoline drawings that are used here come from the Canadian plotting package CONMAP (Taylor, 1976). Some of these routines were modified by H. Røyset and A. Holm, University of Bergen, who also supplied us with the routines INTRPL and DISTR. T. Berge, The Norwegian Polar Research Institute, kindly provided routines for map projections (Berge, 1987). A. Giskehaug, University of Bergen, gave us digitized data for the Norwegian coast. We have also had an excellent cooperation with the Norwegian Hydrographic Service.

The routines in Map-Library are written in standard ANSI FORTRAN-77. Map-Library is at present working on ND-500 from Norsk Data A/S, DEC's VAX computer and on a BULL computer.

In GPGS-F there are two important concepts, window and viewport coordinates. The window is the user coordinate system, and the viewport is the normalized device coordinate system, whose values lies between 0 and 1. Let us clarify this by an example :

The range of the user coordinates is ± 50 in x-direction and -25 to 100 in y-direction and the specified viewport coordinates are 0.5 to 1 in the x-direction and 0.5 to 1 in the y-direction. A point with user coordinates (0., 37.5) will be in the position (0.75, 0.75) of the plotting medium.

It is possible in GPGS-F to transform a picture from one set of viewport coordinates to another, thereby allowing it to be shrunk or blown up quite easily. The disadvantage of this, however, is that alphanumeric information also will be transformed. Therefore Map-Library has a separate set of window and viewport coordinates, which ensures that whatever size the map has, the letters and figures will maintain their specified sizes.

In GPGS-F some routines must always be present, and they must be given in the correct order, as in the following :

```

C ** Users program **
C
C   DIMENSION WF(4), VF(4)
C
C WF = Window in cm. VF = Viewport on the plotting device
C
C IDEV is the GPGS-F ID-number of the plotting device. In this
C case a Tandberg screen.
C   IDEV = 63
C   CALL GPGS
C Initializes device :
C   CALL NITDEV (IDEV)
C Opens the window :
C   CALL WINDW (WF)
C Opens the viewport :
C   CALL VPORT (VF)
C Opens a picture segment :
C   CALL BGNPIC (1)
C
C
C The users generation of a picture
C
C
C Closes the picture segment :
C   CALL ENDPIC
C Closes the plotting device :
C   CALL RLSDEV (IDEV)
C   END

```

In most of the routines in Map-Library latitudes and longitudes are given as integers, e.g. $57^{\circ}30'N = 5730$. Latitudes south of the equator and longitudes west of Greenwich are given as negative numbers.

2 Map projections and topographical data.

Originally Map-Library was designed only to show maps in Mercator's projection with coastlines as the only topographical level. Later it has clearly been demonstrated that other projections and bottom topography is needed in marine science. At the Institute of Marine Research we have received data for the Barents Sea from the Norwegian Hydrographic Services which have been stored in the file structure that Map-Library demands.

This format, and a list of the available map-databases is described in part 4 of this Users Guide.

The following projections are available from Map-Library :

- 1 - XY-coordinates.
- 2 - UTM-coordinates.
- 3 - Mercator.
- 4 - Cone.
- 5 - Polarstereographic.
- 6 - Polarstereographic w/pole in center.

3 Drawing of maps with Map-Library.

3.1 Drawing a chart frame with text.

To draw a chart frame of a map on a plotting device two routines from Map-Library are used. The first is CNVCH. This routine must be called before the GPGS-F routines WINDW and VPORT. CNVCH has two options. When all elements in WF are put to zero before CNVCH is called the scale of the biggest map that it is possible to draw on the actual plotting device is computed. CNVCH finds the size of the window-coordinates, WF, in cm for the actual plotting device. The viewport, VF, used by Map-Library always has the range 0 to 1 both in x- and y-directions. CNVCH assures that space is left around the chart frame for labels and text.

The second option of CNVCH is the possibility to place the map on only a part of the plotting device. In this case WF must be given values in cm as to where the plot should be placed.

The user must also specify a projection for the chartframe.

The syntax of the call is

```
CNVCH (WF, VF, LATOR, LONOR, LAT1, LON1, LAT2, LON2,
>      SFRM, SCALE, NTXTLN, STXT, IMAP)
```

where

WF	= "Window" or "position"	Real array in/output
VF	= Viewport coordinates used by GPGS	Real array output
LATOR	= Construction latitude of map	integer input
LONOR	= Construction longitude of map	integer input
LAT1	= Lower latitude of map	integer input
LON1	= Lower longitude of map	integer input
LAT2	= Upper latitude of map	integer input
LON2	= Upper longitude of map	integer input

Part 5 of 6 : Map-Library.

SFRM	= Size of characters in chart frame in cm	real	input
SCALE	= Inverse of scale of map	integer	in/output
NTXTLN	= Number of textlines under plot (0,1,2,..)	integer	input
STXT	= Size of characters in text lines in cm	real	input
IMAP	= Map-projection	integer	input

If SCALE is given the value "0" on input, CNVCH computes the largest possible scale within the given window-coordinates (WF). The same also happens if the SCALE given on input will place the map outside WF.

After CNVCH is called, the chart frame is drawn by the call

```
CHFRM (LATOR, LONOR, LAT1, LON1, LAT2, LON2, LATIN, N1, LONIN, N2,
>      SFRM, SCALE, ILTPFRM, IMAP)
```

where LATOR, LONOR, LAT1, LAT2, LON1, LON2, SFRM, SCALE and IMAP must have the same value as they had in the call to CNVCH and

LATIN=	Distance between tick marks of latitude of map.	integer	input
	LATIN must be positive		
NLAT =	For every NLAT tick mark a line is drawn across the map. When NLAT is 0 no lines are drawn.	integer	input
LONIN=	Similar to LATIN	integer	input
NLON =	Similar to NLAT	integer	input
ILTPFRM=	Type of line to use in the frame	integer	input

Text under the chart frame is written by a call to

```
TEXT (TXT,NTXTLN,STXT,SFRM)
```

where

TXT	= Array of characters, max. 78 characters in each line.	character	input
-----	---	-----------	-------

NTXTLN, STXT and SFRM must have the same values as in the call to CNVCH.

The program below is a complete working example to draw a chart frame with a text.

The result is shown in fig. 1.

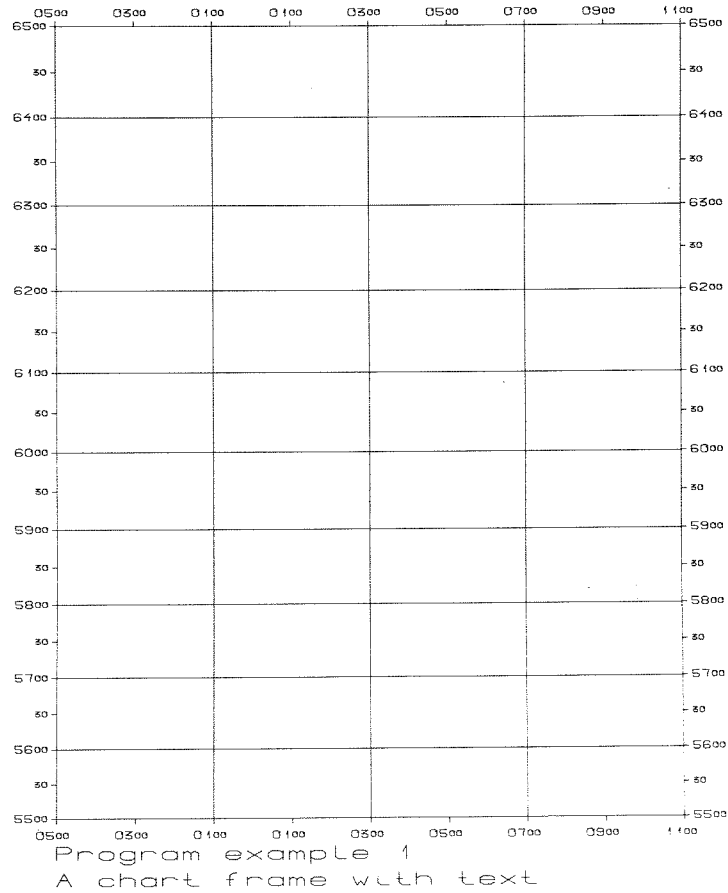


Fig. 1. A chart frame with text.

```

C ** PROG-EXAMPLE-1 **
C
C Draws a chart frame with text.
C-----
C
CHARACTER TXT(2)*50
DIMENSION WF(4), VF(4)
DATA LATOR, LONOR /6200, 0000/
DATA LAT1, LON1, LAT2, LON2 /5500, -0500, 6500, 1100/
DATA SFRM, NTXTLN, STXT /0.3, 2, .8/
DATA LATIN, NLAT /0030,2/
DATA LONIN, NLON /0200,2/
DATA ILTPFRM /1/
DATA WF / 4 * 0. /
DATA IMAP /3/

```

Part 5 of 6 : Map-Library.

```

C
  TXT(1)='Program example 1'
  TXT(2)='A chart frame with text'
C
  CALL GPGS
  IDEV = 63
  CALL NITDEV (IDEV)
C
  CALL CNVCH (WF,VF,LATOR,LONOR,LAT1,LON1,LAT2,LON2,
+           SFRM,SCALE,NTXTLN,STXT, IMAP)
  CALL WINDW (WF)
  CALL VPORT (VF)
C
  CALL BGNPIC (1)
    CALL CHFRM (LATOR, LONOR, LAT1, LON1, LAT2, LON2,
+             LATIN, NLAT, LONIN, NLON,
+             SFRM, SCALE, ILTPFRM, IMAP)
  CALL TEXT (TXT, NTXTLN, STXT, SFRM)
  CALL ENDPIC
C
  CALL RLSDEV (IDEV)
  END

```

3.2 Drawing of coastlines.

To draw coastlines and/or depthlines, Map-Library requires the geographical data stored in a map-database. The format of these map-databases is described in the documentation of Lag-KartBase (part 4 of this Users Guide). At present there are four databases available, all stored on user KART. Additional databases for special use may be made using the program Lag-KartBase.

Initiating a map-database and updating the necessary commonblocks is done by calling the subroutine BASINI.

To include a coastline in fig. 1, two statements are added to Prog-Example-1.

In the beginning of the program

```
CALL BASINI ( BASNVN , USRREF , ISTAT)
```

where

```

BASNVN = Name of map-database           character  input
USRREF = Reference to user where BASNVN is stored character  input
ISTAT  = Status (1=OK, 0=unknown base)  integer    output

```

and just after the call to CHFRM, the COAST-routine should be called

```

CALL COAST (LATOR, LONOR, LAT1, LON1, LAT2, LON2,
>          SCALE, IMAP, NLEVELS, LEVELS, LINTYP, LINCOL)

```

where LATOR, LONOR, LAT1, LON1, LAT2, LON2, SCALE and IMAP must have the same values as in the call to CNVCH and CHFRM and

```

NLEVELS = Number of levels (coast + depths)           integer input
LEVELS   = Array containing the levelnumbers          integer array input
LINTYP   = Array of linetypes to plot the levels      integer array input
LINCOL   = Array of colours to plot the levels        integer array input

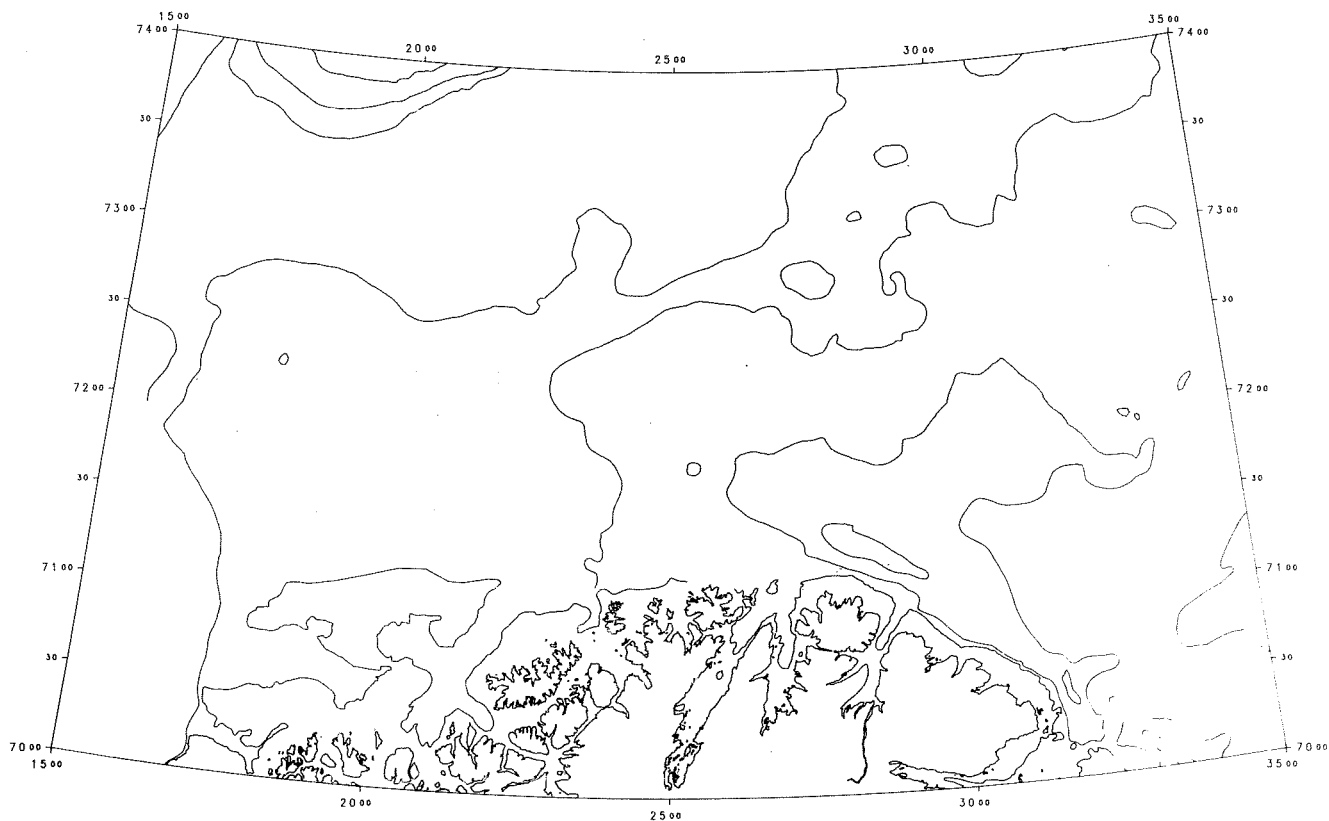
```

The result is shown in fig. 2. The values of NLAT and NLON are here set equal to 0. The calls to GPGS-F routine CFONT is not strictly necessary but is included to show how different text fonts easily could be included.

```

C ** PROG-EXAMPLE-2 **
C
C Draws a chart frame with text, and coast- and depth-lines.
C-----
C
C Includes common blocks describing topographical data :
C
$INCLUDE NEW-MAPLIB:INCL
C
CHARACTER TXT(2)*50
DIMENSION WF(4), VF(4)
DIMENSION LEVELS (5), LINTYP (5), LINCOL (5)
DATA LATOR, LONOR /7200, 2500/
DATA LAT1, LON1, LAT2, LON2 /7000, 1500, 7400, 3500/
DATA SFRM, NTXTLN, STXT /0.3, 2, .8/
DATA LATIN, NLAT /0030, 0/
DATA LONIN, NLON /0500, 0/
DATA ILTPFRM /1/
DATA NLEVELS, LEVELS / 5, 1, 3, 4, 5, 7 /
DATA LINTYP, LINCOL / 1, 1, 1, 1, 1, 1, 1, 1, 1, 1 /
DATA WF / 4 * 0. /

```



*Program example 2.
A chartframe with text, coastlines and depthlines.*

Fig. 2. A chart frame with text and coastlines.

```

C
C Initiates the wanted topographical database :
C
  CALL BASINI ( 'AKUP', 'KART', ISTAT)
C
  TXT(1)='Program example 2.'
  TXT(2)='A chartframe with text, coastlines and depthlines.'
C
  IMAP = 2
C
  CALL GPGS
  IDEV = 63
  CALL NITDEV (IDEV)
C
  CALL CNVCH (WF,VF,LATOR,LONOR,LAT1,LON1,LAT2,LON2,
+           SFRM,SCALE,NTXTLN,STXT, IMAP)
  CALL WINDW (WF)
  CALL VPORT (VF)
C
  CALL BGNPIC (1)
  CALL CFONT (1)
  CALL CHFRM (LATOR, LONOR, LAT1, LON1,
+           LAT2, LON2, LATIN, NLAT,

```

```

+          LONIN, NLON, SFRM, SCALE, ILTPFRM, IMAP)
CALL COAST (LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,
+          IMAP, NLEVELS, LEVELS, LINTYP, LINCOL)
CALL CFONT (3)
CALL TEXT (TXT,NTXTLN,STXT,SFRM)
CALL ENDPIC
c
CALL RLSDEV (IDEV)
END

```

3.3 Drawing of a station net.

Using Map-Library it is easy to plot a ship's course lines during a cruise. The positions given must be geographical coordinates. The courses are presented as straight lines between the stations.

First the geographical positions of the stations must be converted to the correct position on the map. This is done by calling

```

CNVDTI (IXP, IYP, XP, YP, NP, ITYPE, LATOR, LONOR, SCALE, IMAP )

```

LATOR, LONOR, SCALE and IMAP must have the same values as in the call to CNVCH and where

IXP	= Longitude of stations	integer array input
IYP	= Latitude of stations	integer array input
XP,YP	= Position of stations in user coordinates	real array output
NP	= Number of stations	integer input
ITYPE	= Type of coordinates	integer input
	0 = geographical position of stations given	
	1 = user coordinates of stations given	

The data points are then plotted on the map by a call to

```

DTPTS (XP, YP, ZP, NP, LABPT, ISYMB, SSYMB, SLABL, LTYPE, IREPOS)

```

where XP, YP and NP must have the same values as in CNVDTI and where

ZP	= Value of variable at positions XP,YP	real array input
LABPT	= Number of decimals in data point labels	int.arr input

0,1..= Number of decimals
-1 = Omit decimals
-2 = Omit labels on data points
-3 = Omit data points altogether

ISYMB = Type of symbol to be plotted integer array input

1 = +	6 = Diamond	11 = Dot
2 = X	7 = Z	
3 = Star	8 = Triangle	
4 = Square	9 = Triangle upside-down	
5 = Octagon	10 = Circle	

SSYMB = Size of symbol in cm real array input

SLABL = Size of characters of labels in cm real array input

LTYPE = Type of line between datapoints. integer array input

LTYPE < 0 means that the line will be drawn from the center of the actual datapoint to the center of the next datapoint.

0 = No line	4 = Dashed line
1 = Solid line	5 = Dash-dot line
2 = End point line	6..10 user defined linetypes
3 = Dotted line	

IREPOS= Position of symbol relative to geographic position.

integer array input

-1 = Centered
-2 = Right of the center
-3 = Above
-4 = Left of the center
-5 = Under

Note that IREPOS has negative values. When IREPOS is negative the variables LABPT, ISYMB, SSYMB, SLABL, LTYPE and IREPOS are taken to be one-dimensional arrays, with one dataelement, and all datapoints are plotted with the same appearance. When IREPOS(1) is positive LABPT, ISYMB, SSYMB, SLABL, LTYPE and IREPOS must be declared in your main program with dimension equal to NP. In this case each datapoint could

be plotted with different attributes.

The result of this program is shown in fig. 3.

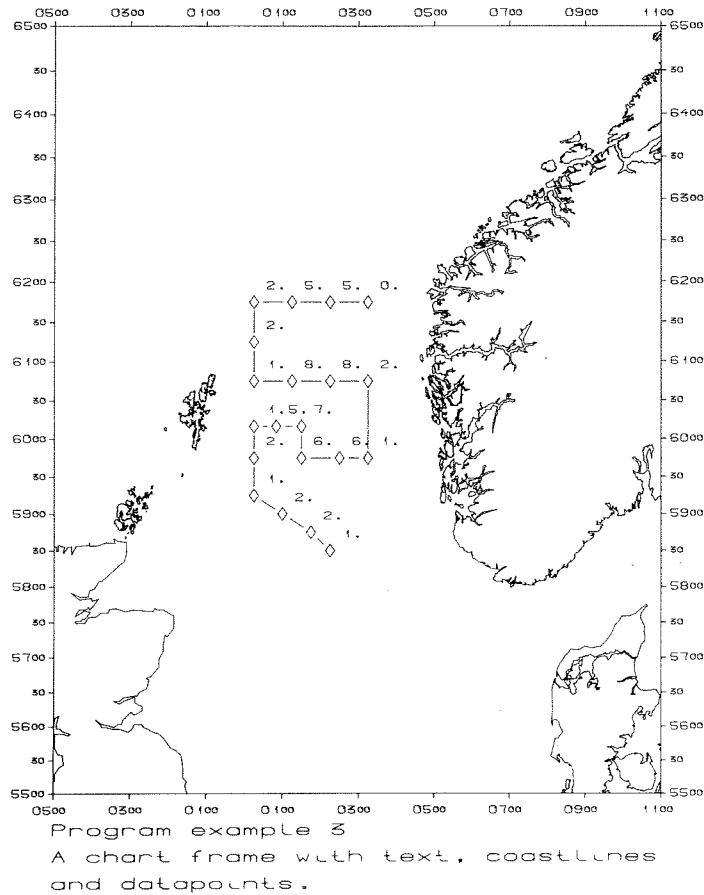


Fig. 3. A chart frame with text

```

C ** PROG-EXAMPLE-3 **
C
C Draws a chart frame with text, coastlines and datapoints.
C-----
C
C Includes common blocks describing topographical data :
C
$INCLUDE NEW-MAPLIB:INCL
C
CHARACTER TXT(3)*50
DIMENSION WF(4), VF(4), LATP(20), LONP(20)
DIMENSION XP(20), YP(20), ZP(20)
C
DATA LATOR, LONOR/6200, 0000/
DATA LAT1, LON1, LAT2, LON2/5500, -0500, 6500, 1100/
DATA SFRM, NTXTLN, STXT, ILTPFRM /0.3, 3, .8, 1/
DATA LATIN, NLAT/0030,0/
DATA LONIN, NLON/0200,0/

```

Part 5 of 6 : Map-Library.

```

C
  DATA NP, ITYPE/20,0/
  DATA LATP/4*6145,6115,4*6045,3*5945,3*6010,5945,
+       5915,5900,5845,5830/
  DATA LONP/0315,0215,0115,3*0015,0115,0215,2*0315,0230,
+       2*0130,0050,3*0015,0100,0145,0215/
  DATA ZP/0.,2*5.,2*2.,1.,2*8.,2.,1.,2*6.,7.,5.,1.,2.,1.,2*2.,1./

C
  DATA LABPT, ISYMB, SSYMB / 0, 6, .4 /
  DATA SLABL, LTYPE, IREPOS / .3, 1, -1 /
  DATA WF/ 4*0.0/
  DATA NLEVELS, LEVELS, LINTYP, LINCOL /1, 1, 1, 1/
  DATA IMAP / 3 /

C
  CALL BASINI ('KARTDATA', 'KART', ISTAT)

C
  TXT(1) = 'Program example 3'
  TXT(2) = 'A chart frame with text, coastlines'
  TXT(3) = 'and datapoints.'

C
  IDEV = 63

C
  CALL GPGS
  CALL NITDEV (IDEV)
  CALL CNVCH (WF, VF, LATOR, LONOR, LAT1, LON1, LAT2, LON2,
+           SFRM, SCALE, NTXTLN, STXT, IMAP)
  CALL WINDOW (WF)
  CALL VPORT (VF)

C
  CALL BGNPIC (1)
  CALL CHFRM (LATOR, LONOR, LAT1, LON1,
+           LAT2, LON2, LATIN, NLAT,
+           LONIN, NLON, SFRM, SCALE, ILTPFRM, IMAP)
  CALL COAST (LATOR, LONOR, LAT1, LON1, LAT2, LON2, SCALE,
+           IMAP, NLEVELS, LEVELS, LINTYP, LINCOL)
  CALL CNVDTI (LONP, LATP, XP, YP, NP, ITYPE, LATOR, LONOR,
+           SCALE, IMAP)
  CALL DTPTS (XP, YP, ZP, NP, LABPT, ISYMB, SSYMB, SLABL,
+           LTYPE, IREPOS)
  CALL TEXT (TXT, NTXTLN, STXT, SFRM)
  CALL ENDPIC

C
  CALL RLSDEV (IDEV)
  END

```

3.4 Interpolation and drawing of isolines.

An efficient way to visualize the density of a 2-D field is to draw isolines. Until recently this was a tedious manual procedure. Now several computer routines exist to perform such an interpolation. The routine ZGRID used here is described by Taylor (1976). The user has to

define a regular grid, Z, with NX lines in the x-direction and NY lines in the y-direction that masks the chart frame. The grid points are defined as the intersections of these lines. The input parameter CAYIN to ZGRID decides what type of interpolation to be used. When CAYIN is 0, a Laplace interpolation is performed. This is a 2-D linear interpolation. When CAYIN has a high value, an interpolation using spline functions is performed. A reasonable value of CAYIN is 5-10. Another important input parameter, NRNG, defines how many grid intervals a grid intersection might be apart from any data point. If the distance exceeds NRNG it is given a value of 10^{35} , i.e. set to undefined. We have made a small routine ZGRI which calls ZGRID. In ZGRI the user can decide whether or not he permits any point in the grid to be negative. Points in Z outside the region to be contoured should be initialized to 10^{35} .

ZGRI (Z, NX, NY, XP, YP, ZP, NP, CAYIN, NRNG, NGPS)

where

Z	= Array to be set up	real array in/output
NX,NY	= Dimension of Z	integer input
XP	= X-coordinates of data points	real array input
YP	= Y-coordinates of data points	real array input
ZP	= Values of the data points	real array input
NP	= Number of data points	integer input
CAYIN	= Degree of spline equation the user wants	real input
NRNG	= Number of grid spaces a grid point may be away from any data point before it is set to undefined.	integer input
NGPS	= Negative/positive values in Z	integer input
	-1 = negative values allowed	
	+1 = negative values not allowed	

Before an interpolation is begun the user must define what areas on the map that should be set to undefined. This can be done in two ways. The first is to use the coastlines in the chart frame by calling

BLKCH1 (Z, NX, NY, LATOR, LONOR, LAT1, LON1, LAT2, LON2, SCALE,

LATSEA, LONSEA, IST, IMAP)

where LATOR, LONOR, LAT1, LON1, LAT2, LON2, SCALE and IMAP must have the same values as in the call to CNVCH and

Z	= Array of grid points	real array output
NX	= Number of grid lines in x-direction	integer input
NY	= Number of grid lines in y-direction	integer input
LATSEA	= Latitude of a point at sea inside the chart frame	integer input
LONSEA	= Longitude of a point at sea inside the chart frame	integer input
IST	= Status variable	integer in/output

At input: IST = 0 ==> do the whole routine
 IST = 1 ==> checks if (LATSEA,LONSEA)
 is at sea

At output: IST = 0 ==> (LATSEA,LONSEA) is at sea
 IST = 1 ==> (LATSEA,LONSEA) is on shore

This routine must be given one point at sea within the chart frame. It then "senses" where the coastlines are and blanks grid points onshore and grid points closer to shore than half the grid step.

If the chartframe is non-square (UTM, polarstereographic, etc.), the routine BLKCH3 will be called at the end of BLKCH1 in order to set the area outside the chartframe to undefined.

In some cases it is not satisfactory to extrapolate data outside what may be defined as "the surveyed area". In these cases, the geographical coordinates of the vertices of a closed polygon defining the surveyed area are given by the user. The grid is then blanked by the routine

BLKCH2 (Z, NX, NY, IVERTX, IVERTY, NVERT, LATOR, LONOR, SCALE, IMAP)

where LATOR, LONOR, SCALE and IMAP must have the same values as in the

call to CNVCH and

Z	= Array to be blanked	real array output
NX	= Number of grid lines in x-direction	integer input
NY	= Number of grid lines in y-direction	integer input
IVERTX	= Longitudes of vertices in polygon	integer array input
IVERTY	= Latitudes of vertices in polygon	integer array input
NVERT	= Number of vertices in blanking polygon	integer input

The routine checks which grid points that are inside the polygon using the routine INSIDE (Taylor, 1976). All other grid points are set to undefined.

When the interpolation is done the user may smooth the data using

SMOOTH (Z,NX,NY,NSM)

Given Z(I,J) and NSM (0,1,2,3..) the routine applies Laplacian smoothing to Z, NSM times by means of the operation

$$Z(I,J)=Z(I,J)+.25*((Z(I,J+1)+Z(I,J-1),Z(I-1,J),Z(I+1,J))/4-Z(I,J))$$

The sweep is alternately SW to NE and NE to SW. Z is assumed to be zero at the edges. Unused points in Z should be 10^{35} (Taylor, 1976).

The routine CONTU (Taylor, 1976) is now used to present the isolines in the plot,

CONTU (Z,NX,NY,ZLEV,NDECL,LWGTL,NLEV,NARC,SLBLCO)

where

Z	= Grid points	real array input
NX	= Number of grid lines in x-direction	integer input
NY	= Number of grid lines in y-direction	integer input
ZLEV	= Values of contours to be drawn	real array input
NDECL	= Number of decimal places in the labels of the contours	integer array input

0,1..= number of decimals
-1 = no decimal fraction
-2 = no label

LWGTL = Type of the contour lines integer array input
0 = no line 3 = dotted line
1 = solid line 4 = dashed line
2 = end point 5 = dash-dot line

NLEV = Dimension of ZLEV, NDECL and LWGTL integer input
NARC = Number of subsegments each contour segment integer input
 are divided into

SLBLCO= Size of characters of labels on contours in cm real input

The parameter NARC decides how smoothly the isolines will be. It must have a value between 1 and 10. A value of NARC equal to 3 will often give a good result. If NARC has a too-high value, numerical peculiarities will occur, often resulting in "a double isoline". The program example below uses BLKCH1 i.e. coastlines for blanking and produces fig. 4.

```
C ** PROG-EXAMPLE-4 **
C
C Draws a chart frame with text, isolines and coastlines
C Grid is blanked using coastlines.
C-----
C
C Includes common blocks describing topographical data :
C
$INCLUDE NEW-MAPLIB:INCL
C
CHARACTER TXT(3)*50
DIMENSION WF(4),VF(4),LATP(20),LONP(20),Z(30,30)
DIMENSION XP(20),YP(20),ZP(20),ZLEV(5),LWGTL(5),NDECL(5)
DIMENSION ICLRL (5)
C
DATA LATOR, LONOR /6200, 0000/
DATA LAT1, LON1, LAT2, LON2 /5500, -0500, 6500, 1100/
DATA SFRM, NTXTLN, STXT, ILTPFRM /0.3, 3, .8, 1/
DATA LATIN, NLAT /0030, 0/
DATA LONIN, NLON /0200, 0/
DATA NP, ITYPE /20,0/
DATA LATP /4*6145,6115,4*6045,3*5945,3*6010,5945,
+ 5915,5900,5845,5830/
DATA LONP /0315,0215,0115,3*0015,0115,0215,2*0315,0230,
+ 2*0130,0050,3*0015,0100,0145,0215/
DATA ZP/0.,2*5.,2*2.,1.,2*8.,2.,1.,2*6.,7.,5.,1.,2.,1.,2*2.,1./
DATA NDECL/5*1/,NARC/3/,NLEV/5/,ZLEV/0.,1.,2.,5.,10./,LWGTL/5*1/
DATA NLEVELS, LEVELS, LINTYP, LINCOL /1, 1, 1, 1/
DATA ICLRL / 5*1 /
DATA SLBLCO / .3 /
```

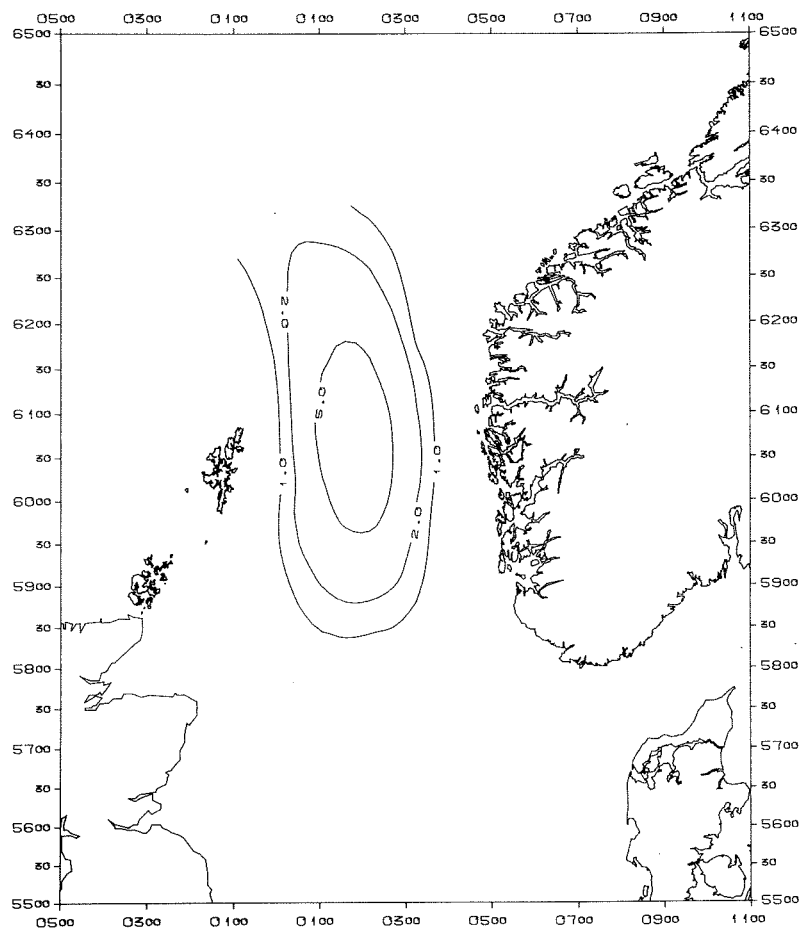


Fig. 4. A chart frame with text

```

C
DATA NX,NY,NGPS,NRNG,CAYIN,NSM /30,30,1,5,5..5/
DATA LATSEA,LONSEA /5800,0300/
DATA WF /4*0.0/
DATA IMAP / 3 /

C
CALL BASINI ('KARTDATA ', 'KART', ISTAT)

C
TXT(1) = 'Program example 4.'
TXT(2) = 'A chart frame with text, coastlines'
TXT(3) = 'and isolines. Grid is blanked using coastlines'
IDEV = 63

C
CALL GPGS
CALL NITDEV (IDEV)
CALL CNVCH (WF,VF,LATOR,LONOR,LAT1,LON1,LAT2,LON2,
+          SFRM,SCALE,NTXTLN,STXT,IMAP)
CALL WINDW (WF)
CALL VPORT (VF)
CALL BGNPIC (1)
CALL CHFRM (LATOR, LONOR, LAT1, LON1,
+          LAT2, LON2, LATIN, NLAT,
+          LONIN, NLON, SFRM, SCALE, ILTPFRM, IMAP)

```

Part 5 of 6 : Map-Library.

```

CALL COAST (LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,
+          IMAP, NLEVELS, LEVELS, LINTYP, LINCOL)
CALL CNVDTI (LONP,LATP,XP,YP,NP,ITYPE,LATOR,LONOR,
+          SCALE, IMAP)
CALL ZGRI (Z,NX,NY,XP,YP,ZP,NP,CAYIN,NRNG,NGPS)
CALL SMOOTH (Z,NX,NY,NSM)
CALL CONTU (Z,NX,NY,ZLEV,NDECL,LWGTL,ICLRL,
+          NLEV,NARC,SLBLCO)
CALL ENDPIC
CALL RLSDEV (IDEV)
END
    
```

In the next example a polygon defines the surveyed area and BLKCH2 is used to blank the grid. The result is shown in fig. 5.

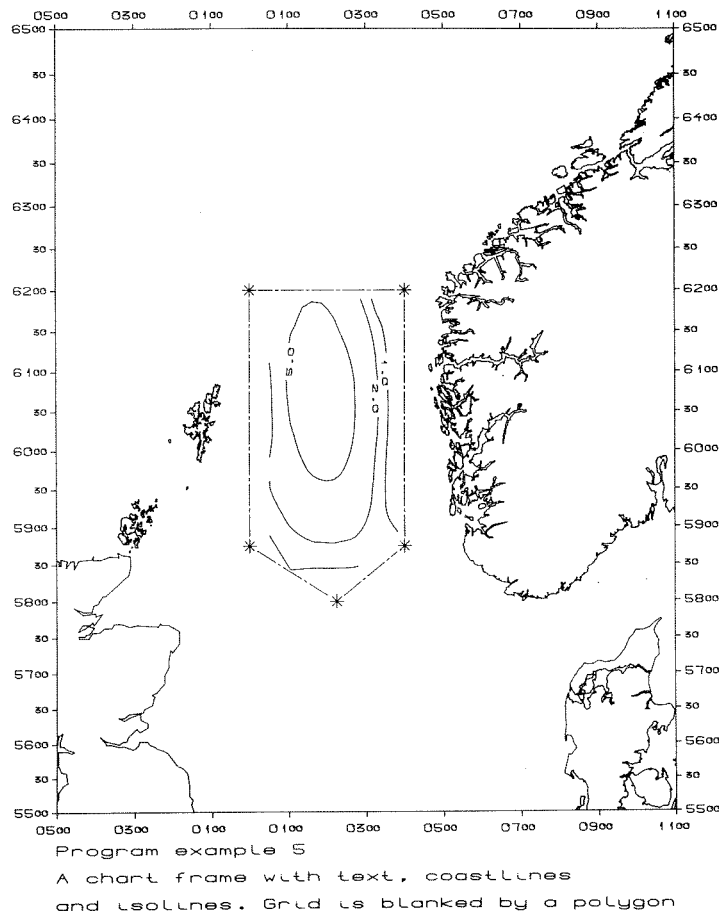


Fig. 5. Chart frame with coast- and isolines. Polygon blanks grid.

```

C ** PROG-EXAMPLE-5 **
C
C Draws a chart frame with text, isolines and coastlines
C Grid is blanked using a polygon.
C-----
C
C Includes common blocks describing topographical data :
    
```



```

C
$INCLUDE NEW-MAPLIB:INCL
C
CHARACTER TXT(3)*50
DIMENSION WF(4),VF(4),LATP(20),LONP(20),Z(30,30)
DIMENSION XP(20),YP(20),ZP(20),ZLEV(5),LWGT(5),NDECL(5)
DIMENSION IVERTX(5),IVERTY(5),VERTX(6),VERTY(6)
DIMENSION ICLRL (5)
C
DATA LATOR, LONOR/6200, 0000/
DATA LAT1, LON1, LAT2, LON2/5500, -0500, 6500, 1100/
DATA SFRM, NTXTLN, STXT, ILTPFRM /0.3, 3, .8,1/
DATA LATIN, NLAT/0030,0/
DATA LONIN, NLON/0200,0/
DATA NP,ITYPE/20,0/
DATA LATP/4*6145,6115,4*6045,3*5945,3*6010,5945,
+      5915,5900,5845,5830/
DATA LONP/0315,0215,0115,3*0015,0115,0215,2*0315,0230,
+      2*0130,0050,3*0015,0100,0145,0215/
DATA ZP/0.,2*5.,2*2.,1.,2*8.,2.,1.,2*6.,7.,5.,1.,2.,1.,2*2.,1./
DATA NDECL/5*1/,NARC/3/,NLEV/5/,ZLEV/0.,1.,2.,5.,10./,LWGT/5*1/
DATA ICLRL / 5*1 /
C
DATA LABPT,ISYMB,SSYMB / -2, 3, .4 /
DATA SLABL,LTYPE,IREPOS / .3, 5, -1 /
c
DATA SLBLCO / .3 /
DATA NX,NY,NGPS,NRNG,CAYIN,NSM/30,30,1,5,5.,5/
DATA NVERT/5/
DATA IVERTY/5845,5800,5845,6200,6200/
DATA IVERTX/0000,0215,0400,0400,0000/
DATA WF/4*0.0/, IMAP /3/
DATA NLEVELS, LEVELS, LINTYP, LINCOL /1, 1, 1, 1/
C
CALL BASINI ('KARTDATA','KART', ISTAT)
TXT(1) = 'Program example 5'
TXT(2) = 'A chart frame with text, coastlines'
TXT(3) = 'and isolines. Grid is blanked by a polygon'
IDEV=63
C
CALL GPGS
CALL NITDEV (IDEV)
CALL CNVCH (WF,VF,LATOR,LONOR,LAT1,LON1,LAT2,LON2,
+      SFRM,SCALE,NTXTLN,STXT,IMAP)
CALL WINDW (WF)
CALL VPORT (VF)
CALL BGNPIC (1)
CALL CHFRM (LATOR, LONOR, LAT1, LON1,
+      LAT2, LON2, LATIN, NLAT,
+      LONIN, NLON, SFRM, SCALE, ILTPFRM, IMAP)
CALL COAST (LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,
+      IMAP, NLEVELS, LEVELS, LINTYP, LINCOL)
CALL CNVDTI (LONP,LATP,XP,YP,NP,ITYPE,LATOR,LONOR,
+      SCALE,IMAP)

CALL TEXT (TXT,NTXTLN,STXT,SFRM)
CALL BLKCH2 (Z,NX,NY,IVERTX,IVERTY,NVERT,LATOR,LONOR,

```

Part 5 of 6 : Map-Library.

```

+           SCALE, IMAP)
CALL ZGRI   (Z, NX, NY, XP, YP, ZP, NP, CAYIN, NRRNG, NGPS)
CALL SMOOTH (Z, NX, NY, NSM)
CALL CONTU (Z, NX, NY, ZLEV, NDECL, LWGTL, ICLRL,
+           NLEV, NARC, SLBLCO)
C Draws the polygon :
CALL CNVDTI (IVERTX, IVERTY, VERTX, VERTY, NVERT, ITYPE,
+           LATOR, LONOR, SCALE, IMAP)
VERTX(NVERT+1) = VERTX(1)
VERTY(NVERT+1) = VERTY(1)
CALL DTPTS (VERTX, VERTY, ZP, NVERT+1,
+           LABPT, ISYMB, SSYMB, SLABL, LTYPE, IREPOS)
CALL ENDPIC
CALL RLSDEV (IDEV)
END

```

3.5 Integration and drawing of an interpolated grid.

In some cases it is not desirable to present the interpolated data by isolines since this process can obscure the real result of the interpolation, namely the values in each grid point. The routine

```
DRWGRD (Z, NX, NY, ILTPGRD, IPREGRD)
```

shows the result of the interpolation on the map. The value in each grid intersection is shown above and to the right of the point. When the value is represented by a blank, the grid point is undefined. At present, the linetype, ILTPGRD, and the presentation style, IPREGRD, are dummy parameters.

The routine INTGRT makes it possible to integrate a variable over a geographical area. One should be cautious using this routine since it works on the rectangular grid in cm. It is recommended to use the Mercator projection (IMAP = 3) with this routine.

The call syntax is

```
INTGRT (Z, NX, NY, LATOR, LONOR, LAT1, LON1, LAT2, LON2, SCALE, IUNIT, TOTAL)
```

where LATOR, LONOR, LAT1, LON1, LAT2, LON2 and SCALE must have the same values as in the call to CNVCH and

Z	= Array to be integrated	real array output
NX	= Number of grid lines in x-direction	integer input
NY	= Number of grid lines in y-direction	integer input
IUNIT	= The unit Z(I,J) is given in	integer input
	1 = units/square meter	
	2 = units/square nautical mile	
TOTAL	= Total number of units of the variable within the chart frame	real output

The measured data points must be given as number of units per square meter or square nautical mile. INTGRT uses the routine AREA, which computes the area of a section on the globe.

AREA (LAT1,LON1,LAT2,LON2,IUNIT,A)

where

LON1	= Lower longitude of sector	integer input
LON2	= Upper longitude of sector	integer input
LAT1	= Lower latitude of sector	integer input
LAT2	= Upper latitude of sector	integer input
IUNIT	= Wanted unit of A	integer input
	1 = square meters	
	2 = square nautical miles	
A	= Area of sector in m ² or nm ²	real output

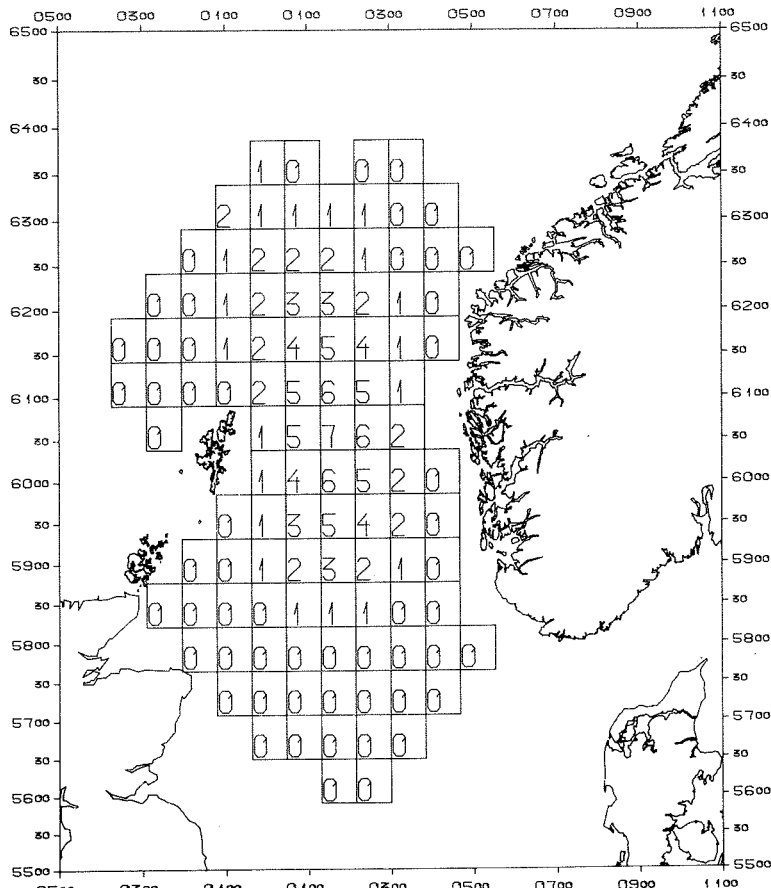
A complete program example is shown below and the result in fig. 6.

```

C ** PROG-EXAMPLE-6 **
C
C Draws a chart frame with text, coastlines and grid.
C One variable is integrated over the area.
C-----
C
C Includes common blocks describing topographical data :
C
$INCLUDE NEW-MAPLIB:INCL
C
CHARACTER TXT(4)*50
DIMENSION WF(4),VF(4),LATP(20),LONP(20),Z(20,20)
DIMENSION XP(20),YP(20),ZP(20)
DATA LATOR, LONOR/6200, 0000/
DATA LAT1, LON1, LAT2, LON2/5500, -0500, 6500, 1100/

```

Part 5 of 6 : Map-Library.



Program example 6
 A chart frame with text, grid and coastlines.
 Grid is blanked by coastlines
 TOTALSUM : .33E+08

Fig. 6. A chart frame with coast- and isolines and integration of a variable.

```

DATA SFRM, NTXTLN, STXT, ILTPFRM /0.3, 4, .8, 1/
DATA LATIN, NLAT/0030.0/
DATA LONIN, NLON/0200.0/
DATA LATSEA, LONSEA/5800,0300/
DATA NP, ITYPE/20,0/
DATA LATP/4*6145,6115,4*6045,3*5945,3*6010,5945,
+ 5915,5900,5845,5830/
DATA LONP/0315,0215,0115,3*0015,0115,0215,2*0315,0230,
+ 2*0130,0050,3*0015,0100,0145,0215/
DATA ZP/0.,2*5.,2*2.,1.,2*8.,2.,1.,2*6.,7.,5.,1.,2.,1.,2*2.,1./
DATA NX,NY,NGPS,NRNG,CAYIN,NSM,IUNIT/20,20,1,4,5.,5,1/
DATA NLEVELS, LEVELS, LINTYP, LINCOL /1,1,1,1/
DATA WF/0.,0.,0.,0./
DATA IMAP / 3 /

C
CALL BASINI ('KARTDATA','KART', ISTAT)

C
TXT(1) = 'Program example 6'
TXT(2) = 'A chart frame with text, grid and coastlines.'
TXT(3) = 'Grid is blanked by coastlines'
TXT(4) = ' '

C
    
```

```

IDEV = 63
CALL GPGS
CALL NITDEV (IDEV)
CALL CNVCH (WF,VF,LATOR,LONOR,LAT1,LON1,LAT2,LON2,
+          SFRM,SCALE,NTXTLN,STXT, IMAP)
CALL WINDW (WF)
CALL VPORT (VF)
CALL BGNPIC (1)
CALL CHFRM (LATOR, LONOR, LAT1, LON1,
+          LAT2, LON2, LATIN, NLAT,
+          LONIN, NLON, SFRM, SCALE, ILTPFRM, IMAP)
CALL COAST (LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,
+          IMAP, NLEVELS, LEVELS, LINTYP, LINCOL)
CALL CNVDTI (LONP,LATP,XP,YP,NP,ITYPE,LATOR,LONOR,
+          SCALE, IMAP)
CALL BLKCH1 (Z,NX,NY,LATOR,LONOR,LAT1,LON1,LAT2,LON2,
+          SCALE,LATSEA,LONSEA,IST, IMAP)
CALL ZGRI (Z,NX,NY,XP,YP,ZP,NP,CAYIN,NRNG,NGPS)
CALL SMOOTH (Z,NX,NY,NSM)
CALL DRWGRD (Z,NX,NY, 1, 0)
CALL INTGRT (Z,NX,NY,LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,
+          IUNIT,TOTAL, IMAP)
WRITE (TXT(4),'(10HTOTALSUM :,E10.2)') TOTAL
CALL TEXT (TXT,NTXTLN,STXT,SFRM)
CALL ENDPIC
CALL RLSDEV(IDEV)
END

```

3.6 Text and symbols at specific positions.

Often it is desired to write text or symbols at specific geographical positions on the map. By a call to

```
XYPSI (LATOR, LONOR, LATPT, LONPT, X, Y, SCALE, IMAP)
```

where LATOR, LONOR, SCALE and IMAP must have the same values as in the call to CNVCH and

LATPT = Latitude of point	integer input
LONPT = Longitude of point	integer input
X = X-coordinate of point on map in cm	real output
Y = Y-coordinate of point on map in cm	real output

the relative position in cm to the construction point of the map is found. To find the right position on the plotting medium, one has to scale these coordinates according to the window and viewport coordina-

tes computed by CNVCH. This is done by a call to

XYCONV (X, Y)

The right position is now found. To position the "pen" here a call to the GPGS-F routine LINE is necessary. Below, an example is shown in which we draw the name of a country and mark a station with a label on it. Symbols are drawn by the routine

PLTMRK (ISYMB, SSYMB)

where

ISYMB = Type of symbol to be drawn	integer input
1 = +	6 = Diamond
2 = X	7 = Z
3 = Star	8 = Triangle
4 = Square	9 = Triangle upside-down
5 = Octagon	10 = Circle
SSYMB = Size of symbol in cm	real input

In the program below, we position the pen "on shore" in Norway, write the name in skewed characters, move to a position in the North Sea, and plot a station with label. The result is shown in fig. 7.

```
C ** PROG-EXAMPLE-7**
C
C Draws a chart frame with text and special symbols.
C-----
C
C Includes common blocks describing topographical data :
C
$INCLUDE NEW-MAPLIB:INCL
C
CHARACTER TXT(2)*50
DIMENSION WF(4), VF(4)
DATA LATOR, LONOR/6200, 0000/
DATA LAT1, LON1, LAT2, LON2/5500, -0500, 6500, 1100/
DATA SFRM, NTXTLN, STXT, ILTPFRM /0.3, 2, .8, 1/
DATA LATIN, NLAT /0030,0/
DATA LONIN, NLON /0200,0/
DATA NLEVELS, LEVELS, LINTYP, LINCOL /1,1,1,1/
DATA WF /4*0.0/
DATA IMAP / 3 /
C
CALL BASINI ('KARTDATA','KART', ISTAT)
```

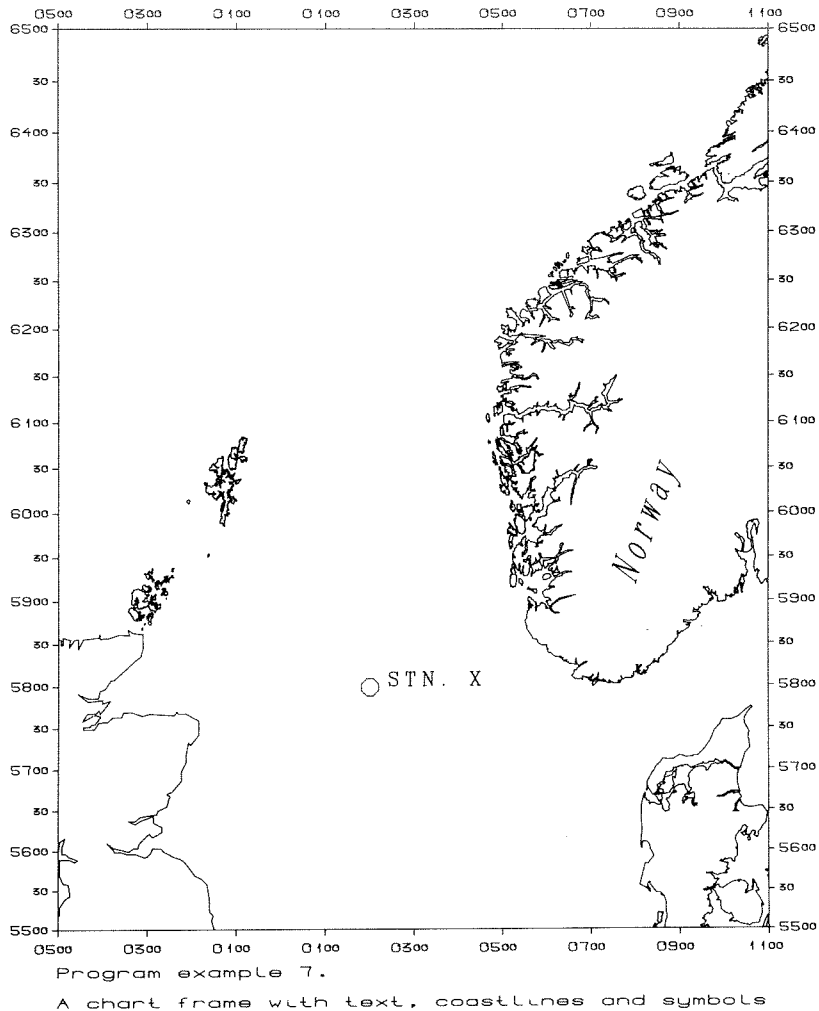


Fig. 7. A chart with symbols at geographical positions.

C

```
TXT(1)='Program example 7.'  
TXT(2)='A chart frame with text, coastlines and symbols'  
IDEV=63
```

C

```
CALL GPGS  
CALL NITDEV (IDEV)  
CALL CNVCH (WF,VF,LATOR,LONOR,LAT1,LON1,LAT2,LON2,  
+          SFRM,SCALE,NTXTLN,STXT,IMAP)  
CALL WINDW (WF)  
CALL VPORT (VF)  
CALL BGNPIC (1)  
CALL CHFRM (LATOR, LONOR, LAT1, LON1,  
+          LAT2, LON2, LATIN, NLAT,  
+          LONIN, NLON, SFRM, SCALE, ILTPFRM, IMAP)  
CALL COAST (LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,  
+          IMAP, NLEVELS, LEVELS, LINTYP, LINCOL)  
CALL TEXT (TXT,NTXTLN,STXT,SFRM)  
CALL XYPSI (LATOR,LONOR,5910,0800,X,Y,SCALE,IMAP)  
CALL XYCONV (X,Y)  
CALL LINE (X,Y,0)
```

C

```
CALL CSIZES (.6, 1.5)
```

Part 5 of 6 : Map-Library.

```

CALL CROTAD (70.)
CALL CSHEA (0.3)
CALL CFONT (2)
CALL CHARC ('Norway*.')
c
CALL CROTAD (0.)
CALL CSHEA (0.)
CALL XYPSI (LATOR,LONOR,5800,0200,X,Y,SCALE,IMAP)
CALL XYCONV (X,Y)
CALL LINE (X,Y,0)
CALL PLTMRK (5,.5)
CALL CSIZES (.5,.75)
CALL CHARC (' STN. X*.')
CALL ENDPIC
c
CALL RLSDEV (IDEV)
END

```

If a position on the plotting device is given in cm, it is possible to find its geographical coordinate by a call to

```
XYCONI (X,Y)
```

followed by a call to

```
GEOPSI (LATOR, LONOR, LATPT, LONPT, X, Y, SCALE, IMAP)
```

where LATOR, LONOR, SCALE and IMAP must have the same values as in the call to CNVCH, and

LATPT = The points latitude	integer ouput
LONPT = The points longitude	integer ouput
X,Y = Coordinates of th point in cm on the map	real input

This is a useful feature when one works on an interactive graphical device. The user may get the window and viewport coordinates used in Map-Library by checking the values in the following common block

```
Common /BHELP/ W(4), V(4), Size
```

in his program.

3.7 Cross sections.

Stations in a cross section can be positioned using geographical coordinates in several ways. In Map-Library only two possibilities are implemented. The zig-zag method places the first station at zero and computes the distance between all the stations. This can be thought of as a stretched zig-zag line. The method is not satisfactory when many stations are involved, e.g., data gathered in an area over a decade, or acoustic data to be presented each nautical mile. The second method is to project the position of the stations along great circles normal to the great circle passing through the two geographical positions the user has defined as the start and stop of the section. We will call this the projection method.

3.7.1 Horizontal presentation of stations.

The horizontal presentation of stations laying on a stretched zig-zag line is trivial and is not shown. To present stations included in a section using the projection method several problems must be solved. The mathematics are described in Westgård (1984). The routine

```
PRJCTN (LAT1,LON1,LAT2,LON2,LATX,LONX,XX,NX,DIST,YLTP,XLNP,NN)
```

where

LAT1 = Latitude of start of the section	integer input
LON1 = Longitude of start of the section	integer input
LAT2 = Latitude of stop of the section	integer input
LON2 = Longitude of stop of the section	integer input
LATX = Latitudes of the stations	integer array input
LONX = Longitudes of the stations	integer array input
XX = Distances in km that the stations should be placed away from (LAT1,LON1) along the section line. If XX is -1., for some stations, it is not possible to place the projection of it between (LAT1,LON1) and (LAT2,LON2).	real array output

NX = Number of stations to be examined integer input
DIST = Accepted distance in km that a station can be real input
apart from the great circle passing through
(LAT1,LON1) and (LAT2,LON2).
YLTP = Latitudes of the projections of the stations. real output
If it is not possible to project the station,
YLTP is returned with the value -200.
XLNP = Longitudes of the projections of the stations. real output
If it is not possible to project the station,
XLNP is returned with the value -200.
NN = Number of projected positions. NN should be integer input
equal to NX if projected positions of the
stations are wanted, otherwise it should be set
to zero.

computes the geographical coordinates of the intersections between the greatest circle, S1, passing through the start, P1, and stop, P2, of the section and the great circles passing through the stations and at the same time being normal to S1. If the projection of a station is outside the segment of S1 between P1 and P2, SEG1, it is rejected. In the projection method a station is also rejected if the distance from the projected position on S1 to the station is more than DIST km, measured along the great circle normal to S1 passing through the station.

The routine ZONE finds the four corners in an area where every point is less than DIST km from SEG1 measured normal to SEG1 (see fig. 8, page 32).

ZONE (YLAT1,XLON1,YLAT2,XLON2,DIST,YLATC,XLONC)

where

YLAT1 = Latitude of start of section in degrees real input
XLON1 = Longitude of start of section in degrees real input
YLAT2 = Latitude of stop of section in degrees real input
XLON2 = Longitude of stop of section in degrees real input
DIST = Half the width of the sector in km real input
YLATC = Latitudes of the four corners of the sector. real arr output

XLONC = Longitudes of the four corners of the sector. real arr output

A call to GRTCRL draws a great circle segment between two geographical coordinates on a Mercator map.

GRTCRL (LATOR,LONOR,YLAT1,XLON1,YLAT2,XLON2,SCALE)

where LATOR, LONOR and SCALE must have the same values as in the call to CNVCH and

YLAT1 = Latitude of point 1 in degrees	real,input
XLON1 = Longitude of point 1 in degrees	real,input
YLAT2 = Latitude of point 2 in degrees	real,input
XLON2 = Longitude of point 2 in degrees	real,input

Below is an example of how to use the routines. It is seen in fig. 8, that four stations are outside the sector and sixteen stations are inside it. Both real and projected positions are shown.

The distance between two geographical positions along a great circle is computed by the real function

DISTR (LAT1,LON1,LAT2,LON2)

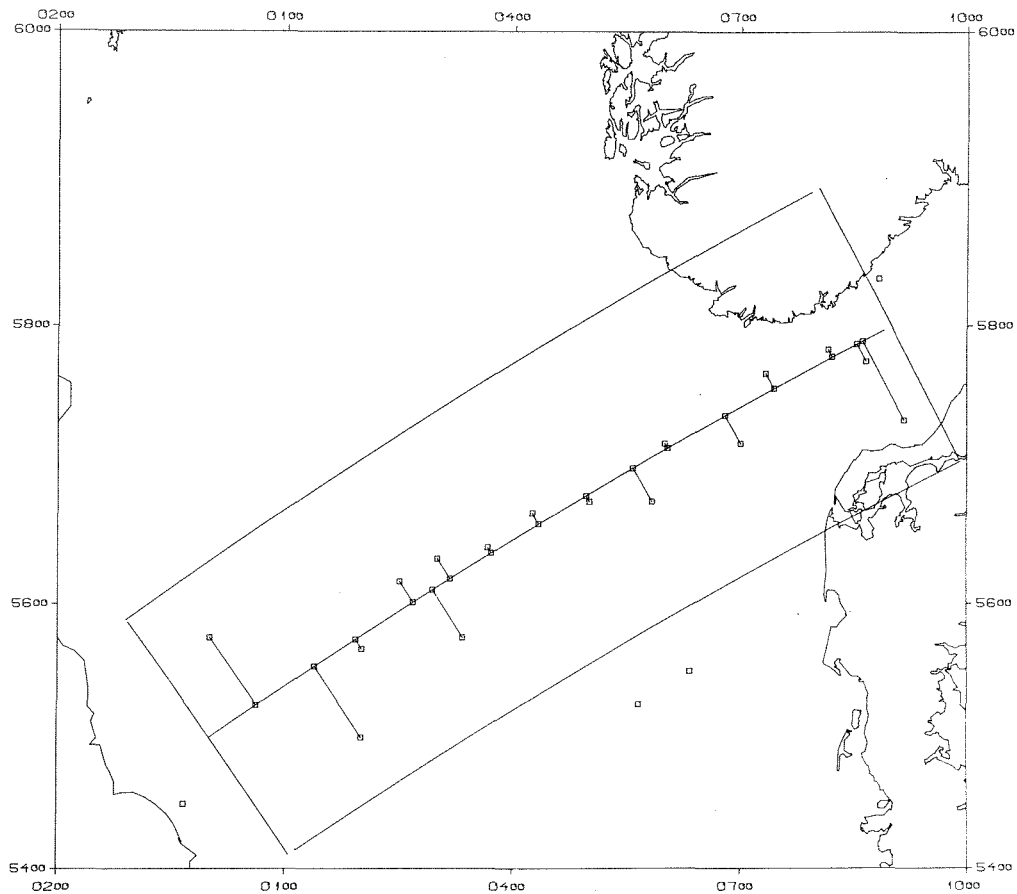
where

LAT1 = Latitude of first point	integer input
LON1 = Longitude of first point	integer input
LAT2 = Latitude of second point	integer input
LON2 = Longitude of second point	integer input

```

C ** PROG-EXAMPLE-8 **
C
C Draws a chartframe with costlines and an area delimited by four
C greatcircles. Stations within this area should be included in
C a section.
C-----
C
C Includes common blocks describing topographical data :
C
$INCLUDE NEW-MAPLIB:INCL
C
CHARACTER TXT(5)*60

```



Program example 8.
 Horizontal presentation of stations on a section.
 Number of stations : 20. The sector where every point
 is less than 120 km from the great circle between
 55 00 N, 00 00 W and 58 00 N, 09 00 E is shown

Fig. 8. Horizontal presentation of stations.

```

DIMENSION WF(4),VF(4),LATST(20),LONST(20),XST(20)
DIMENSION XLNP(20),YLTP(20),XLONC(4),YLATC(4)
DATA ISYMB/4/,SSYMB/.2/,NST/20/
DATA NTXTLN/5/,DIST/120./
DATA SFRM/.3/,STXT/.5/,ILTPFRM /1/,ILTPGCL /1/
DATA LATOR,LONOR/5700,0300/
DATA YLAT1,XLON1,YLAT2,XLON2/55.00,00.00,58.00,09.00/
DATA LAT1,LON1,LAT2,LON2/5500,0000,5800,0900/
DATA LATF1,LONF1,LATF2,LONF2/5400,-0200,6000,1000/
DATA LATIN,NLAT,LONIN,NLON /0200,0,0300,0/
DATA LATST/5430,5545,5500,5540,5610,5620,5625,5545,5515,5530,
+ 5640,5645,5645,5710,5710,5740,5720,5820,5750,5745/
DATA LONST/-0020,0000,0200,0200,0230,0300,0340,0320,0540,0620,
+ 0415,0500,0550,0600,0700,0720,0910,0850,0810,0840/
DATA WF/0.,0.,0.,0./
DATA NLEVELS, LEVELS, LINTYP, LINCOL /1,1,1,1/
    
```

C

```
CALL BASINI ('KARTDATA','KART', ISTAT)
```

C

```

TXT(1) = 'Program example 8.'
TXT(2) = 'Horizontal presentation of stations on a section.'
TXT(3) = 'Number of stations : 20. The sector where every point'
TXT(4) = 'is less than 120 km from the great circle between'
    
```

```

TXT(5) = '55 00 N, 00 00 W and 58 00 N, 09 00 E is shown'
IMAP = 3
IDEV = 63

C
CALL GPGS
CALL NITDEV (IDEV)
CALL CNVCH (WF,VF,LATOR,LONOR,LATF1,LONF1,LATF2,LONF2,
+          SFRM,SCALE,NTXTLN,STXT,IMAP)
CALL WINDW (WF)
CALL VPORT (VF)
CALL BGNPIC (1)
CALL CHFRM (LATOR,LONOR,LATF1,LONF1,
+          LATF2,LONF2,LATIN,NLAT,
+          LONIN,NLON,SFRM,SCALE,ILTPFRM,IMAP)
CALL COAST (LATOR,LONOR,LATF1,LONF1,LATF2,LONF2,SCALE,
+          IMAP,NLEVELS,LEVELS,LINTYP,LINCOL)
CALL TEXT (TXT,NTXTLN,STXT,SFRM)
CALL PRJCTN (LAT1,LON1,LAT2,LON2,LATST,LONST,XST,NST,DIST,
+          YLTP,XLNP,NST)
YLATOR = RDGR (LATOR)
XLONOR = RDGR (LONOR)
DO FOR I = 1,NST
  CALL XYPSI (LATOR,LONOR,LATST(I),LONST(I),XSTN,YSTN,
+          SCALE,IMAP)
  CALL XYCONV (XSTN,YSTN)
  CALL LINE (XSTN,YSTN,0)
  CALL PLTMRK (ISYMB,SSYMB)
  IF(XST(I).NE.-1) THEN
    CALL XYPSR (YLATOR,XLONOR,YLTP(I),XLNP(I),
+          XPSTN,YPSTN,SCALE,IMAP)
    CALL XYCONV (XPSTN,YPSTN)
    CALL LINE (XPSTN,YPSTN,1)
    CALL PLTMRK (ISYMB,SSYMB)
  endif
ENDDO

CALL GRTCRL (LATOR,LONOR,YLAT1,XLON1,YLAT2,XLON2,
+          SCALE,ILTPGCL,IMAP)
CALL ZONE (YLAT1,XLON1,YLAT2,XLON2,DIST,YLATC,XLONC)
CALL GRTCRL (LATOR,LONOR,YLATC(1),XLONC(1),YLATC(2),
+          XLONC(2),SCALE,ILTPGCL,IMAP)
CALL GRTCRL (LATOR,LONOR,YLATC(2),XLONC(2),YLATC(3),
+          XLONC(3),SCALE,ILTPGCL,IMAP)
CALL GRTCRL (LATOR,LONOR,YLATC(3),XLONC(3),YLATC(4),
+          XLONC(4),SCALE,ILTPGCL,IMAP)
CALL GRTCRL (LATOR,LONOR,YLATC(4),XLONC(4),YLATC(1),
+          XLONC(1),SCALE,ILTPGCL,IMAP)
CALL ENDPIC

C
CALL RLSDEV (IDEV)
END

```

3.7.2 Vertical presentation of stations.

To plot a vertical section, ZGRI, DTPTS, TEXT and CONTU are used in the same way as in the previous examples. Instead of CNVCH we use CNVCS1 to compute the right size of the window and viewport coordinates.

CNVCS1 (WF, VF, LAT1, LON1, LAT2, LON2, LATST, LONST, NST, IPROJ, SFRM, NTXTLN, STXT, XCM, YCM, DPTMAX, DPTMIN)

where

WF	= Window coordinates used by GPGS	real array output
VF	= Viewport coordinates used by GPGS	real array output
LAT1	= Latitude of start of section	integer input
LON1	= Longitude of stop of section	integer input
LAT2	= Latitude of start of section	integer input
LON2	= Longitude of stop of section	integer input
LATST	= Latitudes of stations	integer array input
LONST	= Longitudes of stations	integer array input
NST	= Number of stations	integer input
IPROJ	= Projection method (1=zig-zag, 2=great circle)	integer input
SFRM	= Size of characters of the frame in cm	real input
NTXTLN	= Number of text lines under plot (0,1,2,...)	integer input
STXT	= Size of characters in text lines in cm	real input
XCM	= Wanted size in cm of the in the x-direction	real in/output
YCM	= Wanted size in cm of the plot in the y-direction	real in/output
DPTMAX	= Maximum depth of the frame of the section	real input
DPTMIN	= Minimum depth of the frame of the section	real input

CNVCS2 projects stations and positions where bottom depths are measured to the right position on the section.

CNVCS2 (IPROJ, LAT1, LON1, LAT2, LON2, LATBTM, LONBTM, XBTM, YBTM, NBTM, LATST, LONST, XST, NDPTST, NST, XP, YP, ZP, NP, DIST)

Part of the grid will be under the sea bed. The corresponding grid points are blanked (given a value of 10^{35}) using the measured bottom depths as vertices in a blanking polygon. The blanking is done by the routine

BLKCS (Z,NX,NY,XBTM,YBTM,NBTM)

where

Z	= Grid to be blanked	real array output
NX	= Number of lines in grid in x-direction	integer input
NY	= Number of lines in grid in y-direction	integer input
XBTM	= X-coordinate of bottom depths in km.	real array input
YBTM	= Bottom depth in metric metres (positive)	real array input
NBTM	= Number of bottom depths measured, max 38.	real input

The frame of the section with labels is plotted by a call to

CRSFRM (XSTP,YSTP,SFRM)

where

XSTP	= Step size in user coordinates in x-direction	real input
YSTP	= Step size in user coordinates in y-direction	real input
SFRM	= Size of characters in frame in cm	real input

The bottom is drawn using the routine

BOTTOM (XBTM,YBTM,NBTM,NBTMPL)

where

XBTM	= X-coordinates of bottom depths in km.	real array input
YBTM	= Bottom depths measured in metric metres	real array input
NBTM	= Number of measured bottom depths	integer input
NBTMPL	= Number of bottom depths to be interpolated	integer input

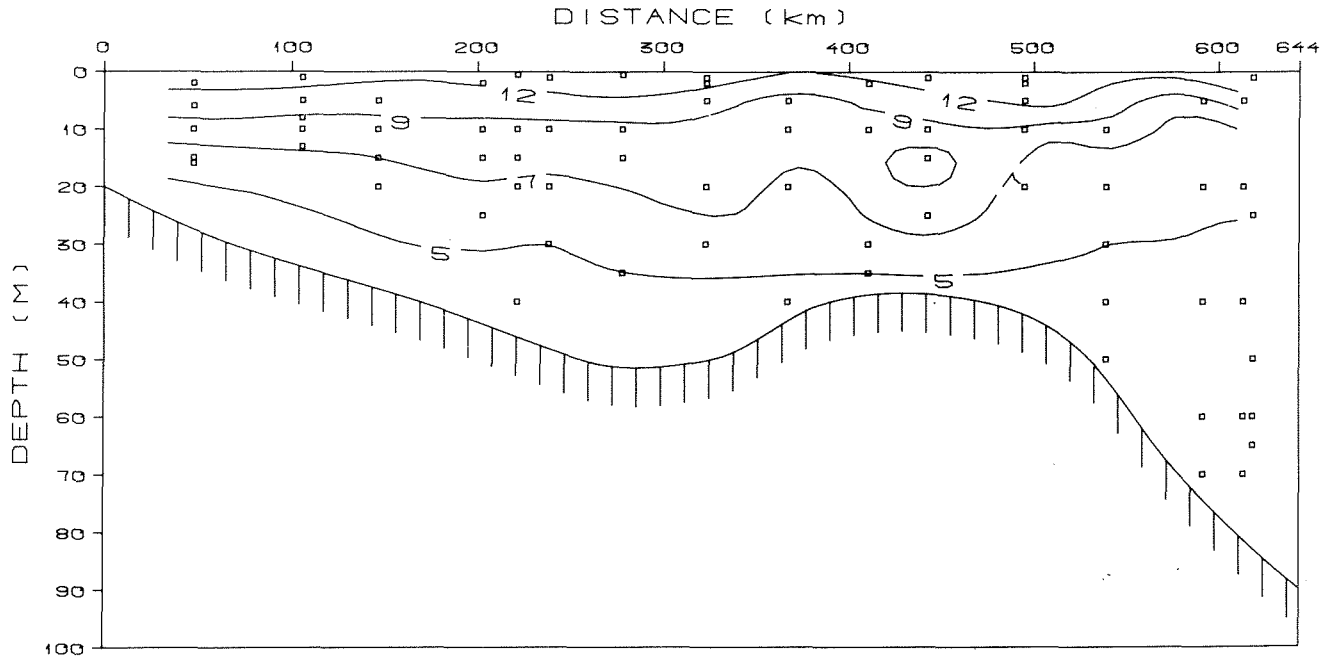
BOTTOM uses the routine INTRPL (Akima, 1972), which computes reasonable values of a singlevalued function $y=f(x)$ when some (x,y) values are given.

INTRPL (IU,L,X,Y,N,U,V)

where

IU = Number of the output unit, e.g., 1 = terminal integer input
 L = Number of input data points integer input
 X = Array of dimension L storing the X values real array input
 (abcissas) of the data points in ascending order
 Y = Array of dimension L storing the Y values real array input
 (ordinates) of the input data points
 N = Number of points at which interpolation of the
 Y values (ordinates) is desired integer input
 U = Array of dimension N storing the X values of real array input
 the desired points.
 V = Array of dimension N where the interpolated Y real array output
 values are stored.

```
C ** PROG-EXAMPLE-9 **
C
C Draws a section frame with text, isolines and datapoints
C Grid is blanked using bottom contour.
C-----
C
C Includes common blocks describing topographical data :
C
$INCLUDE NEW-MAPLIB:INCL
C
  CHARACTER TXT(3)*60
  DIMENSION WF(4),VF(4),LATST(20),LONST(20),LATBTM(10),LONBTM(10)
  DIMENSION XBTM(10),YBTM(10),NDPTST(20),XP(100),YP(100),ZP(100)
  DIMENSION Z(20,20),ZLEV(5),NDECL(5),LWGTL(5),XST(20)
  Dimension Iclrl (5),ISTNO(20)
  Real Xdist,Ydpth
c
DATA LABPT/-2/,ISYMB/4/,SSYMB,SLABL/.15..3/,LTYPE/0/
DATA XDIST/0.0/,YDPTH/0.0/,ILTPCRS /1/,ILTPBTM/1/
DATA NBTM,NST/10,20/,XSTP,YSTP/100..10./,IRED/0/,IKM/1/
DATA IPROJ/2/,NTXTLN/3/,DIST/120./,NBTMPL/100/
DATA SFRM/.4/,STXT/.5/,XCM,YCM/25..12./,DPTMAX,DPTMIN/100..0./
DATA LAT1,LON1,LAT2,LON2/5500,0000,5800,0900/
DATA LATBTM/5500,5515,5545,5610,5630,5650,5715,5730,5745,5800/
DATA LONBTM/0000,0100,0220,0330,0430,0520,0630,0720,0800,0900/
DATA CAYIN/5./,NRNG/5/,NGPS/1/,NX,NY/20,20/
DATA YBTM/20.,30.,40.,50.,50.,40.,40.,50.,70.,90./
```



PROG-EXAMPLE-9
 Number of stations 20. Number of projected stations 16.
 LAT1,LON1 = 5500,0000 LAT2,LON2 = 5800,0900

Fig. 9. Vertical presentations of stations.

```

DATA LATST/5430,5545,5500,5540,5610,5620,5625,5545,5515,5530,
* 5640,5645,5645,5710,5710,5740,5720,5820,5750,5745/
DATA LONST/-0020,0000,0200,0200,0230,0300,0340,0320,0540,0620,
* 0415,0500,0550,0600,0700,0720,0910,0850,0810,0840/
DATA NDPST/5,5,5,4,4,4,4,5,8,5,5,4,4,4,5,5,5,5,5,5/
DATA YP/2.,5.,10.,15.,18., 2.,6.,10.,15.,16., 1.,5.,8.,10.,13.,
* 5.,10.,15.,20., 2.,10.,15.,25., 1.,10.,20.,30., 5,10.,15.,35.,
* 5,10.,15.,20.,40., 1.,2.,10.,20.,25.,30.,31.,32.,
* 1.,5.,10.,15.,35., 1.,2.,5.,20.,30.,5.,10.,20.,40.,2.,10.,
* 30.,35.,1.,10.,15.,25.,1.,2.,5.,10.,20.,10.,20.,30.,40.,50.,
* 1.,25.,50.,60.,65.,10.,20.,30.,40.,50.,5.,20.,40.,60.,70.,
* 5.,20.,40.,60.,70.,4*0./
DATA ZP/12.,10.,8.,6.,5.,13.,10.,8.,6.,6.,14.,10.,8.,8.,8.,
* 10.,8.,7.,6.,13.,8.,8.,6.,14.,8.,6.,5.,16.,8.,8.,5.,14.,8.,8.,
* 6.,4.,13.,12.,8.,6.,6.,5.,5.,5.,13.,10.,8.,7.,5.,14.,12.,10.,8.,
* 6.,8.,8.,6.,4.,12.,8.,6.,5.,14.,8.,10.,8.,15.,14.,13.,8.,6.,
* 8.,6.,5.,4.,4.,16.,5.,4.,3.,2.,6.,5.,5.,4.,4.,8.,6.,5.,4.,
* 3.,10.,6.,5.,4.,3.,4*0./
DATA ISTNO /10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,
* 28,29 /
DATA ZLEV/2.,5.,7.,9.,12./,NLEV/5/,NARC/3/,SLBLCO/.5/
DATA ICLRL / 5*1 /
DATA NDECL/-1,-1,-1,-1,-1/,LWGTL/1,1,1,1,1/
    
```

```

DATA WF/0.,0.,0.,0./
C
TXT(1) = 'PROG-EXAMPLE-9'
TXT(2) = 'Number of stations 20. Number of projected stations 16.'
TXT(3) = 'LAT1,LON1 = 5500,0000 LAT2,LON2 = 5800,0900'
IDEV=63
C
CALL GPGS
CALL NITDEV (IDEV)
CALL CNVCS1 (WF,VF,LAT1,LON1,LAT2,LON2,LATST,LONST,NST,I PROJ,
*          SFRM,NTXTLN,STXT,XCM,YCM,DPTMAX,DPTMIN,XDIST,YDPTH)
CALL WINDW (WF)
CALL VPORT (VF)
CALL CNVCS2 (I PROJ,LAT1,LON1,LAT2,LON2,LATBTM,LONBTM,XBTM,YBTM,
*          NBTM,LATST,LONST,XST,NDPTST,NST,XP,YP,ZP,NP,
*          DIST,ISTNO)
CALL BLKCS (Z,NX,NY,XBTM,YBTM,NBTM)
CALL BGNPIC (1)
CALL CRSFRM (XSTP,YSTP,SFRM,ISTNO,ILTPCRS,NST,XST,IRED,IKM)
CALL TEXT (TXT,NTXTLN,STXT,SFRM)
CALL BOTTOM (XBTM,YBTM,NBTM,NBTMPL,ILTPBTM,SFRM)
CALL DTPTS (XP,YP,ZP,NP,LABPT,ISYMB,SSYMB,SLABT,LTYPE,-1)
CALL ZGRI (Z,NX,NY,XP,YP,ZP,NP,CAYIN,NRNG,NGPS)
CALL CONTU (Z,NX,NY,ZLEV,NDECL,LWGTL,ICLRL,NLEV,NARC,SLBLCO)
CALL ENDPIC
CALL RLSDEV (IDEV)
C
END

```

3.8 Drawing of current-vectors.

To draw current-vectors as arrows on a map, the subroutine ARROW has to be used.

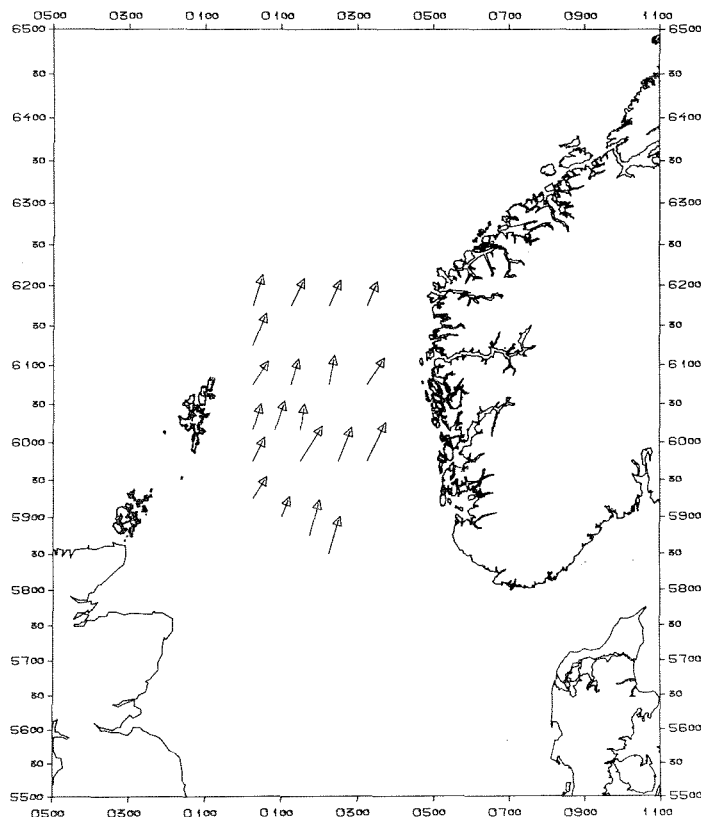
ARROW (XP,YP,NP,NORTH,WEST,CSCM,SSYMB,ICVCTR,LTYPE)

where

XP,YP	= Arrays of positions	Real arrays	Input
NORTH	= North/south current component in cm/s. North positive and south negative.	Real array	Input
WEST	= East/west current component in cm/s. East positive and west negative.	Real array	Input
CSCM	= Scale factor. 1 cm on the map equals a current speed of CSCM cm/s. CSCM = 10 cm/s, means that	Real	Input

a distance of 1 cm in the map is
 equivalent to a current speed of 10 cm/s.

NP	= Size of arrays XP, YP, NORTH, WEST, Icvctr and LTYPE.	Integer	Input
ISYMB	= 8 = Type of symbol is a triangle.	Integer	Input
SSYMB	= Size of symbol in cm	Real	Input
ICVCTR	= Colour index of vector.	Integer array	Input
LTYPE	= Line thickness.	Integer array	Input



Program example 10
 A chart frame with text, coastlines
 and current vectors.

Fig. 10. A map with coastlines and current-vectors.

```

C ** PROG-EXAMPLE-10 **
C
C Draws a chart frame with text, coastlines and current vectors.
C-----
C
C Includes common blocks describing topographical data :
C
$INCLUDE NEW-MAPLIB:INCL
C
CHARACTER TXT(3)*50
    
```

```

DIMENSION WF(4), VF(4), LATP(20), LONP(20)
DIMENSION XP(20), YP(20), ZP(20), ICLRVCTR(20), LTYPE(20)

Real North(20), West(20), CSCM, SSYMB

C
DATA LATOR, LONOR/6200, 0000/, CSCM/20./
DATA LAT1, LON1, LAT2, LON2/5500, -0500, 6500, 1100/
DATA SFRM, NTXTLN, STXT, ILTPFRM /0.3, 3, .8, 1/
DATA LATIN, NLAT/0030,0/
DATA LONIN, NLON/0200,0/

C
DATA NP, ITYPE/20,0/
DATA LATP/4*6145,6115,4*6045,3*5945,3*6010,5945,
+      5915,5900,5845,5830/
DATA LONP/0315,0215,0115,3*0015,0115,0215,2*0315,0230,
+      2*0130,0050,3*0015,0100,0145,0215/
DATA ZP/0.,2*5.,2*2.,1.,2*8.,2.,1.,2*6.,7.,5.,1.,2.,1.,2*2.,1./
DATA NORTH /12.,13.,14.,16.,17.,12.,13.,15.,14.,21.,18.,19.,13.,
+      15.,13.,12.,11.,10.,19.,20./
DATA WEST / 5.,6.,7.,5.,7.,8.,4.,3.,9.,10.,7.,12.,2.,5.,4.,6.,7.,
+      4.,5.,6./
DATA ICLRVCTR /20*1/, LTYPE /20*1/

C
DATA SSYMB / .4 /
DATA WF/ 4*0.0/
DATA NLEVELS, LEVELS, LINTYP, LINCOL /1, 1, 1, 1/
DATA IMAP / 3 /

C
CALL BASINI ('KARTDATA', 'KART', ISTAT)

C
TXT(1) = 'Program example 10'
TXT(2) = 'A chart frame with text, coastlines'
TXT(3) = 'and current-vectors.'

C
IDEV = 63

C
CALL GPGS
CALL NITDEV (IDEV)
CALL CNVCH (WF, VF, LATOR, LONOR, LAT1, LON1, LAT2, LON2,
+      SFRM, SCALE, NTXTLN, STXT, IMAP)
CALL WINDOW (WF)
CALL VPORT (VF)

C
CALL BGNPIC (1)
CALL CHFRM (LATOR, LONOR, LAT1, LON1,
+      LAT2, LON2, LATIN, NLAT,
+      LONIN, NLON, SFRM, SCALE, ILTPFRM, IMAP)
CALL COAST (LATOR, LONOR, LAT1, LON1, LAT2, LON2, SCALE,
+      IMAP, NLEVELS, LEVELS, LINTYP, LINCOL)
CALL CNVDTI (LONP, LATP, XP, YP, NP, ITYPE, LATOR, LONOR,
+      SCALE, IMAP)
CALL ARROW (XP, YP, NP, NORTH, WEST, CSCM, SSYMB, ICLRVCTR, LTYPE)
CALL TEXT (TXT, NTXTLN, STXT, SFRM)
CALL ENDPIC

C
CALL RLSDEV (IDEV)
END

```

3.9 Drawing of maps using colours.

If you want to plot a map with colours or hatching between the various depths levels, this can be specified in the parameter LINCOL in sub-routine COAST. The following program example shows how it can be done.



Prog-Example-12. Plots topographical levels.

A chartframe with text, coastlines and depthlines.

Fig. 11. A map with colours between levels.

```
C ** PROG-EXAMPLE-11 **  
C  
C Draws a chart frame with coast- and depth-lines,  
C and filling with colours between levels.  
C-----  
C  
C Includes common blocks describing topographical data :  
C  
$INCLUDE NEW-MAPLIB:INCL  
C
```

```

CHARACTER TXT(2)*50
DIMENSION WF(4), VF(4)
DIMENSION LEVELS (1), LINTYP (1), LINCOL (2)
DATA LATOR, LONOR /5800, 1100/
DATA LAT1, LON1, LAT2, LON2 /5700, 1000, 5800, 1200/
DATA SFRM, NTXTLN, STXT /0.3, 2, .8/
DATA LATIN, NLAT /0030, 0/
DATA LONIN, NLON /0200, 0/
DATA ILTPFRM /1/
DATA NLEVELS, LEVELS / -1 / , / 1 /
DATA LINTYP, LINCOL / 1 / , / -3, -4 /
DATA WF / 4 * 0. /

C
C Initiates the wanted topographical database :
C
CALL BASINI ( 'KARTDATA', 'P-O:KART', ISTAT)

C
TXT(1)='Program example 11.'
TXT(2)='A chartframe with colours between levels.'

C
IMAP = 2

C
CALL GPGS
IDEV = 62
CALL NITDEV (IDEV)

C
CALL CNVCH (WF,VF,LATOR,LONOR,LAT1,LON1,LAT2,LON2,
+          SFRM,SCALE,NTXTLN,STXT, IMAP)
CALL WINDW (WF)
CALL VPORT (VF)

C
CALL BGNPIC (1)
CALL CFONT (1)
CALL CHFRM (LATOR, LONOR, LAT1, LON1,
+          LAT2, LON2, LATIN, NLAT,
+          LONIN, NLON, SFRM, SCALE, ILTPFRM, IMAP)
CALL COAST (LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,
+          IMAP, NLEVELS, LEVELS, LINTYP, LINCOL)
CALL CFONT (3)
CALL TEXT (TXT,NTXTLN,STXT,SFRM)
CALL ENDPIC

C
CALL RLSDEV (IDEV)
END

```

4 Error handling in Map-Library.

To make the listed program examples in this manual as perspicuous as possible calls to error functions are not shown. The programmer is all the same encouraged to use them. In Map-Library there are three routines that takes care of error-messages. A routine in Map-Library may

have its own error messages, or call other routines in Map-Library that cause an error-message. The first type of error is called an explicit error, the second is an implicit error message. To trap the errors, Map-Library has two common blocks with labels /MAPER1/ and /MAPER2/. They are found in the includefile NEW-MAPERROR:INCL which the user must include in his program. For details, see the chapters "Common blocks in Map-Library" (page 49), and "Reference to routines in Map-Library" (page 61). In the appendix "Error messages in Map-Library" (page 54), you will find a list of the error numbers and the corresponding error-message. When Mlerr returns with the value "0", no error has appeared. The corresponding text to the error number is stored in ChMess is retrieved by a call to

MLERR1 (ICHOICE)

where

- ICHOICE = 0 Displays nothing, but updates ChMess
- 1 Displays ChMess on the screen
- 2 Displays ChMess on the screen and terminate.

The routine MLERR1 does not reset Mlerr to zero, to accomplish this and to be able to display all types of errors that occurred during a session you should call

MLERR2

This routine has no parameters. Its input and output is through the common blocks /MAPER1/ and /MAPER2/, and each time it is called it builds up a table of error messages. It also resets Mlerr to zero. To display all explicit and implicit errors that occurred during a session you should call

MLERR3

All error messages is then shown on the screen.

A skeleton of a Map-Library program may be as follows :


```

C ** Users program **
C
    DIMENSION WF(4), VF(4)
C
C WF = Window in cm. VF = Viewport on the plotting device
C
C
C Includes the common block with error variables :
C
$INCLUDE (Kart)Map-error:incl
    IMESSNO = 0           ;% Sets number of error messages to zero
C
    IDEV = 63            ;% GPGS-F ID for Tandberg screen.
C Initializes GPGS-F :
    CALL GPGS
C Initializes device :
    CALL NITDEV (IDEV)
C Opens the window :
    CALL WINDW (WF)
C Opens the viewport :
    CALL VPORT (VF)
C Reads the user's data :
    CALL READMYDATA      ;% The user's own routine to get data.
C Opens a picture segment :
    CALL BGNPIC (1)
        CALL CNVCH (.....)
        If (Mlerr .ne. 0) then           ;% This sequence should
            Call MLERR1 (1)             ;% be used after each
            Call MLERR2                 ;% call you make to
        Endif                           ;% Map-Library.
        .
        .
        .
C The users generation of a picture
        .
        .
C Closes the picture segment :
    CALL ENDPIC
C Displays all error messages that has occurred :
    If (IMESSNO .ne. 0) Call MLERR3
C Closes the plotting device :
    CALL RLSDEV (IDEV)
END

```

4.1 Restrictions in use of external files.

Map-Library is handling the opening and closing of the files with the map-databases. Therefore the following unit specifiers should be avoided:

6 Geographical positions. Integer/Real degrees.

In special applications it is necessary to describe geographical positions with a better resolution than one minute of arc. Some of the routines in Map-Library has their counterparts imbedded as entry-points in the same routine. The syntax to call these routines is the same except that latitudes and longitudes are given in real degrees, e.g., $57^{\circ}30'5'' = -57.501388$. The routines are :

XYPSR, similar to XYPSI
GEOPSR, similar to GEOPSI
DISTR, similar to DISTR
XYZR similar to XYZI

The function RDGR converts a position given as an integer to real position, e.g. -0538 to -5.6333, while IDGR converts a geographical position in real degrees to integer at the nearest minute.

7 References.

- Akima, H. 1972. Interpolation and Smooth Curve Fitting Based on Local Procedures (E2). Comm. ACM 15(10):914-918
- Anon. 1977. GPGS-F user's guide. Tapir, Trondheim.
- Anon. ND-500 FORTRAN reference manual. Norsk Data ND-60.145.8
- Anon. 1986b. Kart og kartbruk. Universitetsforlaget, 2nd edition (In Norwegian).
- Helle, G. 1984. Programs to digitize and store coastline data. Institute of Marine Research, Bergen no. PS8407
- Berge, T. 1987. Plotlib-1, versjon f. The Norwegian Polar Research Institute (In Norwegian).

Taylor, J. 1976. CONMAP : A computer Program for Contouring Oceanographic Data. Technical note no. 12. Marine Environmental Data Service. Canada.

Westgård, T. 1984. Geometrical calculations on the surface of a sphere. Institute of Marine Research, Bergen no. PS8404 (In Norwegian).

8 Appendices.

8.1 Common blocks in Map-Library.

The common blocks used are stored on two includefiles :

/KART1/, /KART2/, /KART3/, /KART4/, /KART5/ and /BHLP/

are stored in NEW-MAPLIB:INCL, and

/MAPER1/ and /MAPER1/

are stored in NEW-MAPERERROR:INCL

Common-blocks KART1-5 describes current mapdatabase, i.e. its name, on what files it is stored, number of depthlevels, area covered and the resolution. See also in the reference section at the routines BASINI (page 63), and MLERR2 (page 94).

/KART1/

Length In Bytes: 160

Variables		Datatype	Length In Bytes
KBASE	name of mapdatabase	CHARACTER*10	10
INFO	filename of infofile	CHARACTER*50	50
RUTE	filename of pointerfile	CHARACTER*50	50
KYST	filename of posi.file	CHARACTER*50	50

Assigned in routine : BASINI

Referenced in routine: CSTTRU, COAST, BLKCH1

/KART2/

Length In Bytes: 80

Variables		Datatype	Length In Bytes
KTEXT	describing text	CHARACTER*80	80

Assigned in routine : BASINI

/KART3/

Length In Bytes: 16

Variables		Datatype	Length In Bytes
YLATMX	maximum latitude	REAL*4	4
YLATMN	minimum latitude	REAL*4	4
XLONMN	maximum longitude	REAL*4	4
XLONMX	minimum longitude	REAL*4	4

Assigned in routine : BASINI

Referenced in routine: IHELP3

/KART4/

Length In Bytes: 28

Variables		Datatype	Length In Bytes
LATSTP	Latitude extension of one square	INTEGER*4	4
LONSTP	Longitude extension of one square	INTEGER*4	4
NHOR	number of squares in longitude direction	INTEGER*4	4
NVER	number of squares in latitude direction	INTEGER*4	4
NRUTE	number of squares in mapdatabase	INTEGER*4	4
NLATIN	number of squares pr. degree latitude	REAL*4	4

/MAPER1/

Length In Bytes: 144

Variables		Datatype	Length In Bytes
MLERR	error message number	INTEGER*4	4
IMESTB	message numbers occurred in one run	INTEGER*4 array	136
IMESNO	number of messages in one run	INTEGER*4	4

Assigned in routine : REDDIG, MLERR2, MLERR3, MLERR1, INTRPL, RDGR,
DISTR, NCHAR, DTPTS, INSIDE, CONTUR, ZGRID,
AREA, ZGRI, VALID, ARROW, BOTTOM, BLKCS,
BLKCH2, BLKCH1, ZONHLP, CNVC2A,
CNVCS2, CNVC1A, CNVCS1, CNVCH, BASINI

Referenced in routine: MLERR4, GEOPSI, XYPSI, DRWGRD,
CONTU, INTGRT, CSTTRU, CRSFRM
COAST, XYZI, GRTCRL, ZONE, PRJCTA,
PRJCTN, CNVDTR

/MAPER2/

Length In Bytes: 80

Variables		Datatype	Length In Bytes
CHMESS	error message text	CHARACTER*80	80

Assigned in routine : MLERR1

8.2 Error messages in Map-Library.

MLERR CHMESS

- 01 Illegal message number passed to routine MLERR1
- 02 WF has illegal values in CNVCH
- 03 WF has illegal input values to CNVCS1
- 04 Too many datapoints in routine CNVCS2
- 05 Sorry, N=0 in ZONHLP, 1
- 06 Sorry, N=1 in ZONHLP, 2
- 07 The fixed point is on shore, close to shore or outside
 chartframe
- 08 Error in routine BLKCS. NBTM too big
- 09 Illegal values of LAT1 or LON1, or LAT2 or LON2
- 10 All datapoints given to ZGRID are zero
- 11 Warning : Argument of NX > IDIM in routine CONTUR
- 12 Warning : Argument of NY > JDIM in routine CONTUR
- 13 Error in routine CONTUR: CONTUR buffer exceeded - increase
 IBUFF
- 14 One or several datapoints located outside the chart- or
 section frame
- 15 Invalid integer as latitude or longitude
- 16 Error in NCHAR
- 17 L=1 or less in INTRPL
- 18 N=0 or less in INTRPL
- 19 Identical x-values in INTRPL
- 20 X-values out of sequence in INTRPL
- 21 Error exit from BLKCH2. NVERT too big
- 22 Invalid latitude given to DISTR
- 23 Invalid longitude given to DISTR
- 24 Invalid input to AREA
- 25 Error in routine BOTTOM. NBTMPL too big
- 26 Illegal choice in routine MLERR1
- 27 Illegal choice and error message number in MLERR1
- 28 Error in ZGRI. NP must not exceed NPMAX

- 29 Two or several datapoints are located on same geographical position
- 30 Unknown map database
- 31 Error in routine INSIDE of Map-Library
- 32 Two following station numbers in the section are identical
- 33 Number of digits in station numbers exceed 4.
Stations renumbered.
- 34 Arrow head is outside chart frame. Arrow will not be plotted

8.3 Quick reference to routines in Map-Library.

- ARC fits line segments to make a smooth curve.
- AREA finds the area in a geographical sector in square nautical miles or metric meters.
- ARROW plots vectors as arrows in a geographical position on a map.
- BASINI initiates the map-database, and updates the commonblocks.
- BLKC2A same as BLKCH2 but positions in polygon is given and returned in the same arrays.
- BLKC3R same as BLKCH3 but borders of chartframe are given as real values.
- BLKCH1 blanks a grid on a map using the coast- and depth-lines.
- BLKCH2 blanks a grid on a map using a polygon.
- BLKCH3 blanks a grid on a map using the chartframe.
- BLKCS blanks a grid on a crosssection using the bottom contour.
- BOTTOM draws an interpolated bottom contour inside the section.

CHANGE swap two coordinates: [x1, y1] \leftrightarrow [x2, y2] .

CHFRM draws a chartframe using specified projection.

CNVC1A finds the right size of window and viewport coordinates for the actual plotting device for a crossection.

CNVC2A projects the stations to be included in a section to the right positon on a great circle between start and stop of the section.

CNVCH finds the right size of window and viewport coordinates or the actual plotting device for a map.

CNVCS1 finds the right size of window and viewport coordinates for the actual plotting device for a crossection.

CNVCS2 projects the stations to be included in a section to the right positon on a great circle between start and stop of the section.

CNVDTI rescales coordinates of datapoints to correct position in cm on the plot of a map.

CNVDTR same as CNVDTI but array of positions is given and returned in the same array.

COAST draws coast- and depth-lines within a chartframe.

CONSEG plots contours of a grid in parts beeing LSEG grid units per side.

CONTU prepare the drawing of isolines of the grid Z(I,J).

CONTUR draws the isolines of the grid Z(I,J) on the plot.

COSEG contour a grid Z (I, J) .

CRSFRM draws a crossection with labels on the axes.

CSTTRU checks if there is coast- or depth-lines within a given chartframe.

DISTR finds the distance in km from one position on the earth to another.

DISTRA same as distr but the positions are in decimal degrees.

DOUBLE expands a grid to twice its size using cubic interpolation

DRWGRD draws the gridlines with the values of each gridpoint.

DTPTS plots the datapoints on the map.

GEOPSI finds the geographical position of a point when its position on a map is given.

GEOPSR same as GEOPSI but input as real degrees.

GETLEV returns reasonable values of isolines.

GRAFIN sets a Tandberg terminal in alphanumerical or graphical mode.

GRTCRL plots a great circle on a map.

GTU transformation of geographicle coordinates to UTM-coordinates in a given sone.

HELP1 draws a linesegment on the chartframe, together with the small line outside the degreenumbers.

HELP2 write the degreenumbers on the chartframe.

HELP4 used by BLKCH1.

HLP computes constants used in CONTUR, INTGRT and DRWGRD.

IDGR converts a real position to an integer position to the nearest minuit.

IHELP3 find boxnumber of a position in a mapdatabase.

INSIDE decides if a point is outside or inside a polygon.

INTGRT area integrates $Z(I,J)$ to find total abundance in a geographical area. (e.g.: echo data or plankton data).

INTRPL interpolates a singlevalued function $Y=F(X)$.

LATLON given a points carteesian coordinates the routine finds the points latitude and longitude.

MLERR1, routine to gather all error messages. The user must specify options for printout of error messages. Common block NEW-MAPERERROR:INCL is used.

MLERR2, Updates the error table returned from Map-Library.

MLERR3, routine for writing error messages to the screen/terminal

MLERR4, a routine to check if the same error message is registered before.

NCHAR finds number of digits in a real number.

PLANE finds the coefficients of $Z=A*X + B*Y$.

PLTMRK plots one of ten possible symbols.

POLAR finds the area of a polygon.

PRJCTA same as PRJCTN but input are decimal degrees.

PRJCTN the logic used by CNVCS2.

RDGR real function converts an integer position to real degrees

REDDIG reduces number of digits in station number.

SMOOTH smooths the interpolated array.

STRVAL finds the first and last non-blank character in a string.

TEXT writes a text under the plot.

UPDTP updates a geographical position one stepsize.

UTG transforms a point given as UTM-coordinates to geographic-
al coordinates given as centesimal degrees.

VALID checks if latitudes and longitudes given to COAST, CSTTRU
and BLKCH1 is valid with the present mapdatabase.

XYCONI inverse of XYCONV.

XYCONV finds the right position in cm on the plotting device
using the window and viewport coordinates of CNVCH or
CNVCS1.

XYPSI finds the position of a point on a map using the project-
ion.

XYPSR same as XYPSI but input as real degrees.

XYZI given a point's latitude and longitude the routine finds
the point's carteesian coordinates.

XYZR same as XYZI but input as real degrees.

ZGRI interpolaion routine to find Z(I,J) from datapoints.

ZGRID sets up square grid for contouring, given arbitrarily placed data points. Laplace interpolation is used.

ZONE finds four positions DIST km from the great circle segment from 1 to 2.

ZONHLP a routine to do calculations needed in ZONE.

ZZHALF interpolation to get ZZ(2.5) .

8.4 Reference to routines in Map-Library.

ARC

Subroutine ARC (X,Y,NASK,EPS,XX,YY,NRET)

P u r p o s e :

Given four successive points X(K),Y(K),K=1,4 defining 3 segments, the points XX(K),YY(K),K=1,NRET are generated describing a curve between points 2 and 3 trying to match slope with similar curves from 1 to 2 and from 3 to 4. If the segment (2,3) is less than EPS in length, NRET is set to 2 and points 2 and 3 only are Returned in arrays XX,YY. Otherwise NRET are equal the input parameter NASK. The curve XX,YY is a cubic over segment (2,3). If segment (1,2) is less than EPS the second derivative at point 2 is set to zero. If segment (3,4) is less than EPS the second derivative at point 3 is set to zero. These two features may be used for starting and ending an open curve.

P a r a m e t e r s :

X	=	real array, input
Y	=	real array, input
NASK	=	integer , input
EPS	=	real , input
XX	=	real array, output
YY	=	real array, output
NRET	=	integer , output

R e f e r e n c e s / R e v i s i o n s :

Oceanography EMR December 1969

J.Taylor .1976. "CONMAP : A computer program for contouring oceanographih data." Marine Environmental Data Service. Technical Note Nr. 12.
Environment Canada. Fisheries and Marine Service.

R e m a r k s :

E r r o r C o n d i t i o n s :

AREA

Subroutine AREA (LAT1,LON1,LAT2,LON2,IUNIT,A)

P u r p o s e :

Computes the area of a sector of the globe in sq.meters or sq. nm

Parameters :

LON1	= Lower longitude of sector	integer	Input
LON2	= Upper longitude of sector	integer	Input
LAT1	= Lower latitude of sector	integer	Input
LAT2	= Upper latitude of sector	integer	Input
IUNIT	= Wanted unit, 1=sq.meters,2=sq.naut.mile	integer	Input
A	= Area of sector in m**2 or nm**2	real	Output

References / Revisions :

Trond Westgård, IMR, 11.83

Remarks :

Error Conditions :

24

ARROW

Subroutine ARROW (XP,YP,NP,NORTH,WEST,CSCM,SSYMB,ICVCTR,LTYPE)

Purpose :

Plots current vectors as arrows in a geographical position on a map.

Parameters :

XP,YP	= Arrays of positions	Real arrays	Input
NP	= Size of arrays XP, YP, NORTH, WEST, ICVCTR and LTYPE.	Integer	Input
NORTH	= North/south current component in cm/s. North positive and south negative.	Real array	Input
WEST	= East/west current component in cm/s. East positive and west negative.	Real array	Input
CSCM	= Scale factor. 1 cm on the map equals a current speed of CSCM cm/s. CSCM = 10 cm/s, means that a distance of 1 cm in the map is equivalent to a current speed of 10 cm/s.	Real	Input
SSYMB	= Size of symbol in cm	Real	Input
ICVCTR	= Colour index of vector.	Int. array	Input
LTYPE	= Line thickness.	Int. array	Input

References / Revisions :

Tor Knutsen, IMR, 03.88

Remarks :

Error Conditions :

29,14,34

BASINI

Subroutine BASINI (BASNVN, USEREF, ISTAT)

P u r p o s e :

Initiates the map-database, and updates the common blocks storing vital data about the active topographical database.

P a r a m e t e r s :

BASNVN : Name of map-database (max. 10 characters) character, input
 USEREF : Username for map-database character, input
 ISTAT : Status (1=OK, 0=unknown base) integer , output

R e f e r e n c e s / R e v i s i o n s :

Andreas Christiansen, Cap Gemini, December 1987

R e m a r k s :

Output from BASINI is through Common blocks. The topographical data is stored in three separate files.

Common /Kart1/ Kbase, Info, Rute, Kyst
 Common /Kart2/ Ktext
 Common /Kart3/ Ylatmx, Ylatmn, Xlonmn, Xlonmx
 Common /Kart4/ Latstp, Lonstp, Nhor, Nver, Nrute, Nlatin, Nlonin
 Common /Kart5/ Ndepth, Depths

Kbase = Name of map-database (first part of filenames) Character*10
 Info = Name of file with info-data. Character*50
 Rute = Name of file with pointers to Kyst. Character*50
 Kyst = Name of file with topographical data. Character*50
 Ktext = Text string describing the topographical data. Character*80

Ylatmn = Lowest latitude of the topographical data. Real*4
 Ylatmx = Highest latitude of the topographical data. Real*4
 Xlonmn = West longitude of the topographical data. Real*4
 Xlonmx = East longitude of the topographical data. Real*4
 Depths = Values of the depthlevels in the topo. data. Real(Ndepth)*4

Latstp = Smallest step in minutes in latitude. Integer*4
 Lonstp = Smallest step in minutes in longitude. Integer*4
 Nhor = Number of steps in longitude. Integer*4
 Nver = Number of steps in latitude. Integer*4
 Nrute = Number of small "charts" in Kbase (Nhor*Nver) Integer*4
 Nlatin = Number of steps in latitude in Kbase Integer*4
 Nlonin = Number of steps in longitude in Kbase Integer*4
 Ndepth = Number of depthlevels in Kbase (max. 21) Integer*4

Error Conditions :

30

BLKC2A

Subroutine BLKC2A (Z,NX,NY,VERTX,VERTY,NVERT,LATOR,LONOR,SCALE
&, IMAP)

Purpose :

Routine to blank part of grid using a polygon with NVERT vertices
on a map
The routine is 'working' in Mercator-projection.

Parameters :

Z(I,J)	= Array to be blanked	Real arr	Output
NX	= Number of lines in grid in x-direction	Integer	Input
NY	= Number of lines in grid in y-direction	Integer	Input
VERTX(I)	= Longitudes of vertices (west neg.)	Real arr	in/output
VERTY(I)	= Latitudes of vertices (south neg.)	Real arr	in/output
	On input VERTX and VERTY is given as Real numbers (e.g:56 deg 30 min = 5630.00)		
NVERT	= Number of vertices in blanking polygon	Integer	Input
	NVERT less than 3 means no blanking		
LATOR,LONOR	= Construction point of map	Integers	Input
SCALE	= Inverse of scale of map	Real	Input
IMAP	= Map-projection	Integer	Input
	1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical		
	5 - Polarstereographic 6 - Polarstereographic, pole in center		

References / Revisions :

Trond Westgård,IMR, 12.83

Remarks : None.

Error Conditions :

BLKC3R

Entry BLKC3R (Z, NX, NY
&, LZERO
&, LATOR, LONOR
&, YYLAT1, XXLON1
&, YYLAT2, XXLON2
&, SCALE, IMAP)

P u r p o s e :

Similar to BLKCH3,
but the chartframe-borders (LAT1/LAT2/LON1/LON2) are given as
real numbers in desimal degrees (YYLAT1/YYLAT2/XXLON1/XXLON2).

P a r a m e t e r s :

R e f e r e n c e s / R e v i s i o n s :

R e m a r k s :

BLKCH1

Subroutine BLKCH1 (Z,NX,NY,LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,
& LATSEA,LONSEA,IST, IMAP)

P u r p o s e :

Sets points in Z(NX,NY) on land to undefined
The routine is 'working' in Mercator-projection.

If the chartframe is non-rectangular, BLKCH3 is
called to blank outside chartframe.

P a r a m e t e r s :

Z	= Array of gridpoints	Real array, Output
NX	= Number of gridlines in x-direction	Integer , Input
NY	= Number of gridlines in y-direction	Integer , Input
LATOR	= Latitude of origo	Integer , Input
LONOR	= Longitude of origo	Integer , Input
LAT1	= Minimum latitude of CHFRM	Integer , Input
LON1	= Minimum longitude of CHFRM	Integer , Input
LAT2	= Maximum latitude of CHFRM	Integer , Input
LON2	= Maximum longitude of CHFRM	Integer , Input
SCALE	= Inverted scale of map	Real , Input
LATSEA	= Latitude of a point at sea	Integer , Input
LONSEA	= Longitude of a point at sea	Integer , Input
IST	= Status variable with two purposes	Integer , in/output
	Input: IST=0 ==> Do the whole routine	
	IST=1 ==> Checks if (LATSEA,LONSEA) is at sea	
	Output: IST=0 ==> (LATSEA,LONSEA) is at sea	
	IST=1 ==> (LATSEA,LONSEA) is on shore	
IMAP	= Map-projection	Integer Input
	1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical	
	5 - Polarstreographic 6 - Polarstereographic, pole in center	

R e f e r e n c e s / R e v i s i o n s :

Gunnar Helle, Institute of Marine Research, Bergen. 12.83
Andreas Christiansen, Cap Gemini, 12.87 / 4.88

Remarks :Error Conditions :

BLKCH2

Subroutine BLKCH2 (Z,NX,NY,IVERTX,IVERTY,NVERT,LATOR,LONOR,SCALE
&, IMAP)

Purpose :

Routine to blank part of grid using a polygon with NVERT vertices on a map.

Parameters :

Z(I,J)	= Array to be blanked	Real arr	Output
NX	= Number of lines in grid in x-direction	Integer	Input
NY	= Number of lines in grid in y-direction	Integer	Input
IVERTX(I)	= Longitudes of vertices (west negative)	Int. arr	Input
IVERTY(I)	= Latitudes of vertices (south negative)	Int. arr	Input
NVERT	= Number of vertices in blanking polygon, max 20. Nvert less than 3 means no blanking	Integer	Input
LATOR,LONOR	= Construction point of map	Integer	Input
SCALE	= Inverse of scale of map	Real	Input
IMAP	= Map-projection	Integer	Input
	1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical		
	5 - Polarstereographic 6 - Polarstereographic, pole in center		

References / Revisions :

Trond Westgård, IMR, 12.83

Remarks : None.Error Conditions :

BLKCH3

Subroutine BLKCH3 (Z, NX, NY
&, LZERO
&, LATOR, LONOR
&, LAT1, LON1
&, LAT2, LON2
&, SCALE, IMAP)

\$Include (Kart)New-MapError:Incl

Purpose :

Routine to blank the part of a grid which is outside a chartframe. The routine should be used when a map-projection giving a non-rectangular chartframe is used.

It is optional to zeroset the rest of the grid.

It is not necessary to call BLKCH3 when the routine BLKCH1 is used. BLKCH1 calls BLKCH3 when the chartframe is non-rectangular.

Parameters :

Z (I,J)	= Array to be blanked	Real arr	Output
NX	= Number of lines in grid in x-direction	Int.	Input
NY	= Number of lines in grid in y-direction	Int.	Input
LZERO	= Zeroset rest of grid if .TRUE.	Log.	Input
LATOR,LONOR	= Construction point of map	Int.	Input
LAT1	= Minimum latitude of frame (south negative)	Int.	Input
LON1	= Minimum longitude of frame (west negative)	Int.	Input
LAT2	= Maximum latitude of frame (south negative)	Int.	Input
LON2	= Maximum longitude of frame (west negative)	Int.	Input
SCALE	= Inverse of scale of map	Real	Input
IMAP	= Map-projection	Int.	Input
	1 - XY, 2 - UTM, 3 - Mercator, 4 - Cone		
	5 - Polarstreographic 6 - Polarstereographic, pole in center		

References / Revisions :

Andreas Christiansen, Cap Gemini, Bergen, April 1988.

Remarks :

BLKCS

Subroutine BLKCS (Z,NX,NY,XBTM,YBTM,NBTM)

Purpose :

Blanks grid to interpolate and contour a crosssection

Parameters :

Z(NX,NY)	= Array to be blanked	Real arr	Output
NX	= Number of columns in Z	Integer	Input
NY	= Number of rows in Zn Z	Integer	Input
XBTM	= X-coordinate of bottomdepth in kilometres	Real arr	Input
YBTM	= Bottomdepth in metric metres (positive)	Real arr	Input
NBTM	= Number of bottomdepths measured, max 38.	Real	Input

References / Revisions :

Trond Westgård, IMR, 12.83

Remarks : None.

Error Conditions :

8

----- BOTTOM -----

Subroutine BOTTOM (XBTM,YBTM,NBTM,NBTMPL,ILTPBT,SFRM)

Purpose :

Plots a thick bottomline smoothed by INTRPL

Parameters :

XBTM	= X-coordinates of bottomdepths measured in km	Real arr Input
YBTM	= Bottom depths measured in metric metres	Real arr Input
NBTM	= Number of bottom depths measured	Integer Input
NBTMPL	= Number of bottom depths one want to interpol.	Integer Input
ILTPBT	= Linetype to present the bottom with	Integer Input
SFRM	= Size of characters in section frame.	Real Input
	This parameter SFRM*2. gives the size of the bottom contour shading.	

References / Revisions :

Trond Westgård, IMR, 12.83

Tor Knutsen, IMR, 03.88

Remarks :

Error Conditions :

25

----- CHANGE -----

Subroutine CHANGE (X1,Y1,X2,Y2)

Purpose :

Switches two sets of numbers. X1 -> X2, X2 -> X1 ; Y1 -> Y2, Y2 -> Y1

Parameters :

X1,X2,Y1,Y2	= Four numbers	Real, Input, Output
-------------	----------------	---------------------

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 12.84

R e m a r k s : None.E r r o r C o n d i t i o n s : None.

CHFRM

Subroutine CHFRM (LATOR,LONOR,LAT1,LON1,LAT2,LON2,BIN,N1,LIN,N2,
& SFRM,SCALE,ILTFRM, IMAP)P u r p o s e :

The routine draws a chartframe using the specified projection.
All geographical positions is given as integers. South and west is
negative (e.g.: 56 degrees 35 minutes south = -5635).

P a r a m e t e r s :

LATOR	= Construction latitude of map	Int. Input
LONOR	= Construction longitude of map	Int. Input
LAT1	= Minimum latitude of map	Int. Input
LON1	= Minimum longitude of map	Int. Input
LAT2	= Maximum latitude of map	Int. Input
LON2	= Maximum longitude of map	Int. Input
BIN	= Distance between tick marks of latitude of map BIN must be positive	Int. Input
N1	= For every N1 tick mark a line is drawn across the map. When N1 is 0 no lines is drawn.	Int. Input
LIN	= Similar to BIN	Int. Input
N2	= Similar to N1	Int. Input
SFRM	= Size of numbers in the chartframe in cm.	Real Input
SCALE	= Inverse of scale of map.	Real Input
Iltfm	= Type of line to use in the frame	Int. Input
IMAP	= Map-projection	Int. Input
	1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical	
	5 - Polarstreographic 6 - Polarstereographic, pole in center	

R e f e r e n c e s / R e v i s i o n s :

Gunnar Helle, IMR, 12.83

Trond Westgård, IMR, 1986, 1987

Andreas Christiansen, Cap Gemini, January 1988

R e m a r k s :E r r o r C o n d i t i o n s :

CNVC1A

Subroutine CNVC1A (WF, VF, LAT1, LON1, LAT2, LON2, YLATST, XLONST, NST,
& IPROJ, SFRM, NTXTLN, STXT, XCM, YCM, DPTMAX, DPTMIN, XDIST, YDPTH)

P u r p o s e :

Finds the right size of window and viewport coordinates for the actual plotting device to draw a crosssection. When the value of WF is zero WF is computed automatically, otherwise the values of WF gives the position of the plot on the plotting device. Geographical coordinates are input as real degrees.

P a r a m e t e r s :

WF	= "Window" or "position"	Real arr	in/output
VF	= Viewport coordinates actually used by gpgs	Real arr	Input
LAT1	= Latitude of start of section (south negative)	Int.	Input
LON1	= Longitude of stop of section (west negative)	Int.	Input
LAT2	= Latitude of start of section (south negative)	Int.	Input
LON2	= Longitude of stop of section (west negative)	Int.	Input
YLATST	= Latitudes of stations (deg.min.)(s neg.)	Real arr	Input
XLONST	= Longitudes of stations (deg.min.)(w neg.)	Real arr	Input
NST	= Number of stations	Int.	Input
IPROJ	= Projection method (1=zig-zag, 2=great circle)	Int.	Input
SFRM	= Size of characters of the frame in cm	Real	Input
NTXTLN	= Number of textlines under plot (0,1,2,...)	Int.	Input
STXT	= Size of characters in textlines in cm	Real	Input
XCM	= Wanted size in cm of your plot in x-direct.	Real	in/output
YCM	= Wanted size in cm of your plot in y-direct.	Real	in/output
DPTMAX	= Maximum depth on the frame of the section	Real	Input
DPTMIN	= Minimum depth on the frame of the section	Real	Input
XDIST	= XDIST km in the section gives the distance of 1 cm in x-direction of the plot.	Real	Input
YDPTH	= YDPTH meter in the section gives the depth of 1 cm in y-direction of the plot.	Real	Input

R e f e r e n c e s / R e v i s i o n s :

Tor Knutsen, IMR, 3.88

R e m a r k s :

E r r o r C o n d i t i o n s :

CNVC2A

Subroutine CNVC2A (IPROJ,LAT1,LON1,LAT2,LON2,YLATBT,XLONBT,XBTM,
& YBTM,NBTM,YLATST,XLONST,XST,NDPTST,NST,XP,YP,ZP,
& NP,DIST,ISTNO)

P u r p o s e :

Projects the geographical coordinates of the stations and bottomdepth measured as real degrees into the right position on the section the user has specified.

P a r a m e t e r s :

IPROJ = Projection method Int. Input

1 - The stations are placed relative to the start of the Section along a stretched zig zag line.

2 - The stations are projected onto the great circle passing through (LAT1,LON1) and (LAT2,LON2). If the station is more than DIST km away from this great circle it is omitted.

LAT1 = Latitude of start of section (south negative) Int., Input
 LON1 = Longitude of start of section (west negative) Int., Input
 LAT2 = Latitude of stop of section (south negative) Int., Input
 LON2 = Longitude of stop of section (west negative) Int., Input
 YLATBT = Lat. of bottom depths measured (south neg.) Real arr Input
 XLONBT = Long. of bottom depths measured(west neg.) Real arr Input
 XBTM = Distances in km from start of section Real arr Output
 YBTM = Bottomdepths in meters (positive !) Real arr in/output
 NBTM = Number of bottom depths measured Int. in/output
 YLATST = Latitudes of stations (south negative) Real arr Input
 XLONST = Longitudes of stations (west negative) Real arr Input
 XST = Position of stations away from (LAT1,LON1) Real arr Output
 The great circle to (LAT2,LON2) or
 On a stretched zig zag line

NDPTST= Number of depths measured at each station Int. array Input
 NST = Number of stations Int. in/output
 XP = X-coordinates of datapoints in km Real arr Output
 YP = Depths where measurements are done Real arr Output
 ZP = The measured values Real arr in/output
 NP = Number of datapoints Int. Output
 DIST = Distance in km a station is allowed to lay Real Input
 away from the section

ISTNO = Active station numbers nt. array, in/output

R e f e r e n c e s / R e v i s i o n s :

Tor Knutsen, IMR, 5.88

R e m a r k s : None

E r r o r C o n d i t i o n s :

4

----- CNVCH -----

Subroutine CNVCH (WF,VF,LATOR,LONOR,LAT1,LON1,LAT2,LON2,SFRM,
& SCALE,NTXTLN,STXT, IMAP)

P u r p o s e :

Finds the size of the plotting medium and then the right scale and window,viewport and size of the plotting device. When the user want the size of WF to be found automatically WF must be zero on Input, otherwise the values of WF are taken to be the area on the plotting medium in cm where the plot shall be. When SCALE is given a value of zero on Input the routine automatically scales your map using all four corners, LONOR and the greenwich meridian. If you give SCALE a value on Input the map will be of the wanted size if it is room for it on the medium.

P a r a m e t e r s :

WF	= "Window" or "position"	Real arr in/output
VF	= Viewport coordinates actually used by GPGS	Real arr Output
LATOR	= Construction latitude of map,south is neg.	Int. Input
LONOR	= Construction longitude of map,west is neg.	Int. Input
LAT1	= Lower latitude of map, south is neg.	Int. Input
LON1	= Lower longitude of map, west is neg.	Int. Input
LAT2	= Upper latitude of map, south is neg.	Int. Input
LON2	= Upper longitude of map, west is neg.	Int. Input
SFRM	= Size of Characters of the chartframe in cm	Real Input
SCALE	= Inverse of scale of map	Real in/output
NTXTLN	= Number of textlines under plot (0,1,2...)	Int. Input
STXT	= Size of characters in textlines in cm	Real Input
IMAP	= Map-projection	Int. Input
	1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical	
	5 - Polarstreographic 6 - Polarstereographic, pole in center	

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, Inst. of Marine Research, Bergen. 12.83,10.86,05.87
Tor Knutsen, Institute of Marine Research, Bergen, 04-05.87.
Andreas Christiansen, Cap Gemini, Bergen, 02.88.

R e m a r k s :

The routine gives output in common : Common /Bhlp/ W(4),V(4),SIZE

W	= "Window" coordinates	Real
V	= "Viewport" coordinates	Real
SIZE	= Size of the plotting medium in cm	Real

Error Conditions :

2

CNVCS1

Subroutine CNVCS1 (WF,VF,LAT1,LON1,LAT2,LON2,LATST,LONST,NST,I PROJ,
& SFRM,NTXTLN,STXT,XCM,YCM,DPTMAX,DPTMIN,XDIST,YDPTH)

P u r p o s e :

Finds the right size of window and viewport coordinates for the actual plotting device to draw a crosssection. When the value of WF is zero WF is computed automatically otherwise the values of WF gives the position of the plot on the plotting device.

P a r a m e t e r s :

WF	= "Window" or "position"	Real arr	in/output
VF	= Viewport coordinates actually used by gpgs	Real arr	Output
LAT1	= Latitude of start of section (south negative)	Int.	Input
LON1	= Longitude of stop of section (west negative)	Int.	Input
LAT2	= Latitude of start of section (south negative)	Int.	Input
LON2	= Longitude of stop of section (west negative)	Int.	Input
LATST	= Latitudes of stations (deg.min.)(s neg.)	Int. arr	Input
LONST	= Longitudes of stations (deg.min.)(w neg.)	Int. arr	Input
NST	= Number of stations	Int.	Input
I PROJ	= Projection method (1=zig-zag, 2=great circle)	Int.	Input
SFRM	= Size of characters of the frame in cm	Real	Input
NTXTLN	= Number of textlines under plot (0,1,2,....)	Int.	Input
STXT	= Size of characters in textlines in cm	Real	Input
XCM	= Wanted size in cm of your plot in x-direct.	Real in/o	utput
YCM	= Wanted size in cm of your plot in y-direct.	Real in/o	utput
DPTMAX	= Maximum depth on the frame of the section	Real	Input
DPTMIN	= Minimum depth on the frame of the section	Real	Input
XDIST	= XDIST km in the section gives the distance of 1 cm in x-direction of the plot.	Real	Input
YDPTH	= YDPTH meter in the section gives the depth of 1 cm in y-direction of the plot.	Real	Input

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, Institute of Marine Research, Bergen. 12.1983
Tor Knutsen, Institute of Marine Research, Bergen. 6.1988

E r r o r C o n d i t i o n s :

3

----- CNVCS2 -----

Subroutine CNVCS2 (IPROJ,LAT1,LON1,LAT2,LON2,LATBTM,LONBTM,XBTM,
& YBTM,NBTM,LATST,LONST,XST,NDPTST,NST,XP,YP,ZP,
& NP,DIST,ISTNO)

P u r p o s e :

Projects the stations and bottomdepths measured into the right position on the section the user has specified.

P a r a m e t e r s :

IPROJ = Projection method Int., Input

- 1 - The stations are placed relative to the start of the section along a stretched zig zag line.
- 2 - The stations are projected onto the great circle passing through (LAT1,LON1) and (LAT2,LON2). If the station is more than DIST km away from this great circle it is omitted.

LAT1 = Latitude of start of section (south negative)	Int.	Input
LON1 = Longitude of start of section (west negative)	Int.	Input
LAT2 = Latitude of stop of section (south negative)	Int.	Input
LON2 = Longitude of stop of section (west negative)	Int.	Input
LATBTM= Latitudes of bottom depths measured (south neg.)	Int. arr	Input
LONBTM= Longitudes of bottom depths measured(west neg.)	Int. arr	Input
XBTM = Distances in km from start of section	Real arr	Output
YBTM = Bottomdepths in meters (positive !!)	Real arr	in/output
NBTM = Number of bottom depths measured	Int.	in/output
LATST = Latitudes of stations (south negative)	Int. arr	Input
LONST = Longitudes of stations (west negative)	Int. arr	Input
XST = Position of stations away from (LAT1,LON1) The great circle to (LAT2,LON2) or On a stretched zig zag line	Real arr	Output
NDPTST= Number of depths measured at each station	Int. arr	Input
NST = Number of stations	Int.	in/output
XP = X-coordinates of datapoints in km	Real arr	Output
YP = Depths where measurements are done	Real arr	in/output
ZP = The measured values	Real arr	in/output
NP = Number of datapoints	Int.	Ou tput
DIST = Distance in km a station is allowed to lay Away from the section	Real	In put
ISTNO = Active station numbers	Int. arr	in/output

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 1.84
Tor Knutsen, IMR, 1.88

R e m a r k s : None

E r r o r C o n d i t i o n s :

4

CNVDTI

Subroutine CNVDTI (IXP,IYP,XP,YP,NP,ITYPE,LATOR,LONOR,SCALE, IMAP)

P u r p o s e :

Converts datapoints into correct plotting positions on a map

P a r a m e t e r s :

IXP	= Longitude of datapoints (west negative)	Int. arr Input
IYP	= Latitude of datapoints (south negative)	Int. arr Input
XP,YP	= Position of datapoints in user coordinates	Real arr Output
NP	= Number of datapoints	Int. arr Input
ITYPE	= Type of coordinates	Int.,Input
	0 = geographical position of dtpts given	
	1 = user coordinates of dtpts given	
LATOR	= Construction latitude of map	Int.,Input
LONOR	= Construction longitude of map	Int.,Input
SCALE	= Inverse of scale of map	Real ,Input
IMAP	= Map-projection	Int. Input
	1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical	
	5 - Polarstreographic 6 - Polarstereographic, pole in center	

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, Institute of Marine Research, Bergen. 12.83

R e m a r k s : None

E r r o r C o n d i t i o n s :

CNVDTR

Subroutine CNVDTR (XP,YP,NP,ITYPE,LATOR,LONOR,SCALE, IMAP)

P u r p o s e :

Converts datapoints into correct plotting positions on a map

P a r a m e t e r s :

XP,YP	= Position of datapoints (south and west negative)	Real in/output
-------	---	----------------

NP = Number of datapoints Int. array Input
 ITYPE = Indicator for the way the datapoints is given. Int. arr Input
 0=as realnumbers (e.g.:yp=56 deg 30 min=
 5630.00)
 1=as decimaldegrees (e.g.:yp=56 deg 30 min=
 56.50)
 2=user coordinates of dtpts given
 LATOR = Construction latitude of map Int.,Input
 LONOR = Construction longitude of map Int.,Input
 SCALE = Inverse of scale of map Real ,Input
 IMAP = Map-projection Int. Input
 1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical
 5 - Polarstreographic 6 - Polarstereographic, pole in center

References / Revisions :

Trond Westgård, Institute of Marine Research, Bergen 12.83

Remarks : None

Error Conditions :

----- COAST -----

Subroutine COAST (LATOR, LONOR, LAT1, LON1, LAT2, LON2
 &, SCALE, IMAP, NLEVEL, LEVELS, LINTYP, LINCOL)

Purpose :

Draws coastline and depthlines inside a chartframe

Parameters :

LATOR = Construction latitude of map (south negative) Int.,Input
 LONOR = Construction longitude of map (west negative) Int.,Input
 LAT1 = Minimum latitude of map (south negative) Int.,Input
 LON1 = Minimum longitude of map (west negative) Int.,Input
 LAT2 = Maximum latitude of map (south negative) Int.,Input
 LON2 = Maximum longitude of map (west negative) Int.,Input
 SCALE = Inverse of scale of map Real,Input
 IMAP = Map-projection Int. Input
 1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical
 5 - Polarstreographic 6 - Polarstereographic, pole in center
 NLEVEL = Number of levels (coast+depths) to be included. Int. Input
 LEVELS = Array containing the levelnumbers to be incl. Int.arr Input
 LINTYP = Linetype for plotting the various levels. Int.arr Input
 LINCOL = Colour for plotting the various levels. Int.arr Input

References / Revisions :

Gunnar Helle, IMR, 12.83
 Trond Westgård,IMR, 1986, 1987
 Tor Knutsen,IMR, 1987

Andreas Christiansen, Cap Gemini, December 1987

R e m a r k s : None.

E r r o r C o n d i t i o n s :

CONSEG

Subroutine CONSEG (Z,NX,NY,X1,Y1,XL,YL,ZLEV,NDECL,LWGTL,ICLRL,NLEV,
> HGT,NDIV,NARC,XOR,YOR)

P u r p o s e :

Plots contours of grid Z (I,J) in square segments of length LSEG grid units per side. Each grid square is subdivided into NDIV**2 subsquares using cubic polynomial interpolation. NDIV must be a power of 2.

P a r a m e t e r s :

See COSEG.

R e f e r e n c e s / R e v i s i o n s :

Oceanography EMR DEC/1969

J.Taylor .1976. "CONMAP : A computer program for contouring oceanographih data." Marine Environmental Data Service. Technical Note Nr. 12. Environment Canada. Fisheries and Marine Service.

R e m a r k s :

E r r o r C o n d i t i o n s :

CONTU

Subroutine CONTU (Z,NX,NY,ZLEV,NDECL,LWGTL,ICLRL,NLEV,NARC,SLBLCO)

P u r p o s e :

Draw isolines on a map. See also CONTUR.

P a r a m e t e r s :

Z(I,J)	= Array to be contoured	Real array, Input
NX,NY	= Dimension of Z(I,J)	Reals, Input
ZLEV(I)	= Values of contours to be drawn (I=1,NLEV)	Real arr, Input

NDECL(I) = Number of decimal places in the labels of contours (-1 :no decimal, -2 :no label) (I=1,NLEV) Int.array, Input

LWGTL(I) = Type of the contour line (I=1,NLEV) Int.array, Input
0 = no line 1 = solid line 2 = end point
3 = dotted 4 = dashed line 5 = dash-dot line
6..10 user defined linetypes

ICLRL(I) = Color of the contourlines (I=1,NLEV) Int.array, Input

NLEV = Number of contours Int.,Input

NARC = Number of subsegments each contour segment is divided into. NARC=1,2,.....,10. Care must be taken for too high values of NARC. Int.,Input

SLBLCO = Size of characters of labels on isolines in cm Real Input

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 1983, 1987
See also CONTUR.

R e m a r k s :

E r r o r C o n d i t i o n s :

----- CONTUR -----

Subroutine CONTUR (Z,IX,NX,NY,X1,Y1,XL,YL,ZLEV,NDECL,LWGTL,ICLRL,
> NLEV,HGT,NARC,XOR,YOR)

P u r p o s e :

Given a function Z(I,J) defined on a rectangular grid, contours are drawn at specified levels ZLEV with labels of NDECL decimals. Regions may be blanked by defining the function values equal to 10^{35} . Different line types may be selected and the contours can be made more smooth flowing (NARC).

P a r a m e t e r s :

Z(I,J) : (I=1,NX J=1,NY) Real array, Input
Function values to be contoured.
Z(1,1) defines the lower left origo.

IX = Dimension of first index of Z (must be \geq NX). Int. Input

NX,NY = Index limits in use. Int. Input

X1,Y1 = Lower left corner of the grid on the plotter. Real Input

XL,YL = Upper righth corner of the grid on the plotter. Real Input

ZLEV(I) : I=1,NLEV:
The values of the levels to be drawn. Real array Input

NDECL(I): I=1,NLEV:
Number of decimal places in labels of contours. Real array Input
-1: No decimal fraction.
-2: No label.

LWGTL(I): I=1,NLEV: Real array Input
Type of the contour line.

0 = no line 1 = solid line 2 = end point 3 = dotted
 4 = dashed line 5 = dash-dot line
 6..10 = user defined

ICLRL(I) = Color of the isolines (I=1,NLEV) Int. array, Input
 NLEV = Number of contour levels to be drawn. Integer, Input
 HGT = Height of characters in the label. Real, Input
 NARC = An arc of 'NARC' subsegments replaces each segment of the contour, Integer, Input
 NARC may have the values 1,2,3,...,10.
 In this way the contour can be made more smooth flowing, but care should be taken as overlapping of contours is possible for higher values of NARC. NARC=1 has no effect
 XOR,YOR = Position of origo on the plotter in cm. Real, Input

R e f e r e n c e s / R e v i s i o n s :

Oceanography EMR DEC/1969.

Modified by Halvor Råyset, Geophysical Inst., Univ. of Bergen 09.78

J.Taylor .1976. "CONMAP : A computer program for contouring oceanographih data." Marine Environmental Data Service. Technical Note Nr. 12.
 Environment Canada. Fisheries and Marine Service.

R e m a r k s :

E r r o r C o n d i t i o n s :

11,12,13

COSEG

Subroutine COSEG (Z,NX,NY,ZLEV,NDECL,LWGTL,ICLRL,NLEV,
 & NDIV,NARC,SLBLCO)

P u r p o s e :

See CONSEG.

P a r a m e t e r s :

Z(I,J) = Array to be contoured Real array in/output
 NX,NY = Dimension of Z(I,J) Int. Input
 ZLEV(I) = Values of contours to be drawn (I=1,NLEV) Real arr Input
 NDECL(I)= Number of decimal places in the labels of Int. arr Input
 contours (-1 : no decimals, -2 : no label)
 LWGTL(I)= Weight of the contour line (I=1,NLEV) Int. arr Input
 0 no line
 1 solid line
 2 end point
 3 dotted line
 4 dashed line

5 dash-dot line
6..10 user defined linetypes

ICLRL(I)	= Color of the contourlines (I=1,NLEV)	Int. array	Input
NLEV	= Number of contours	Int.	Input
NDIV	= Each grid square is divided into ndiv**2 sub-squares. Ndiv must take a value which is a power of 2. Ndiv=2 or 4 or 8 or ...	Int.	Input
NARC	= Number of subsegments each contour segment is divided into. narc=1,2, ... ,10. care must be taken for too high values of narc.	Int.	Input
SLBLCO	= Size of characters of labels on isolines in cm	Real	Input

References / Revisions :

Trond Westgård, IMR, 1983
See also CONSEG

Remarks :

Error Conditions :

CRSFMR

Subroutine CRSFMR (XSTP,YSTP,SFRM,ISTNO,ILTPCR,NST,XST,IRED,IKM)

Purpose :

Draws a frame of a crosssection with labels and values on axes

Parameters :

XSTP	= Stepsize in user coordinates in x-direction	Real,Input
YSTP	= Stepsize in user coordinates in y-direction	Real,Input
SFRM	= Size of characters in frame in cm	Real,input
ISTNO	= Array of active station numbers	Int. array in/output
ILTPCR	= Linetype to use in the frame	Int.,Input
NST	= Number of active stations	Int.,in/output
XST	= Position of stations away from (lat1,lon1) the great circle to (lat2,lon2) or on a stretched zig zag line	Real array,Input
IRED	= Ired = 1 choose reduced number of digits in station numbers.	Int.,Input
	Ired = 0 number of digits in as input.	
IKM	= Ikm =1 km markings along section.	Int.,Input
	Ikm =0 stations numbers along the section.	

References / Revisions :

Trond Westgård, IMR, 12.83
Tor Knutsen, IMR, 1.88

Remarks :

Error Conditions :

CSTTRU

Subroutine CSTTRU (LAT1,LON1,LAT2,LON2, NLEVEL, LEVELS, LOG)

P u r p o s e :

Checks if we have coast- or depth-lines within a chartframe or not

P a r a m e t e r s :

NLEVEL = Number of levels (coast+depth) to be included. Int.,Input
LEVELS = Array containing levelnumbers to be incl. Int.array,Input
LOG = Logical variable (.true. = lines) Logical,Output

For further explanations of variables see CHFRM

R e f e r e n c e s / R e v i s i o n s :

Gunnar Helle, IMR, 12.83
Andreas Christiansen, Cap Gemini, December 1987

R e m a r k s :

E r r o r C o n d i t i o n s :

DISTR

Function DISTR (LAT1,LON1,LAT2,LON2)

P u r p o s e :

Real function to compute the distance between two points on the earth along a great circle in kilometres.

P a r a m e t e r s :

LAT1 = Latitude of first point (south negative)	Integer,Input
LON1 = Longitude of first point (west negative)	Integer,Input
LAT2 = Latitude of second point	Integer,Input
LON2 = Longitude of second point	Integer,Input

R e f e r e n c e s / R e v i s i o n s :

Halvor Røyset, Institute of geophysics, University of Bergen 12.78
Trond Westgård, IMR, 12.83

R e m a r k s :

Error Conditions :-----
DISTRA

ENTRY DISTRA (XLAT1,XLON1,XLAT2,XLON2)

P u r p o s e :

Same as DISTR but now the positions is in decimal degrees.

P a r a m e t e r s :R e f e r e n c e s / R e v i s i o n s :R e m a r k s :E r r o r C o n d i t i o n s :

22,23

DOUBLE

Subroutine DOUBLE (Z,IX,NX,NY)

P u r p o s e :

The array Z(I,J) (I=1,NX J=1,NY) is expanded to twice its size.
That is : Z(I,J) (I=1,2*NX-1,2 J=1,2*NY-1,2.) The spaces are filled
using cubic polynomial interpolation. Undefined points of Z(I,J)
should be set to 10^{-35} .

P a r a m e t e r s :

Z(I,J) =	Array to be expanded.	Real arr in/output
IX =	Dimension of first index of Z (must be \geq NX).	Int.,Input
NX,NY =	Index limits in use.	Int.,Input

R e f e r e n c e s / R e v i s i o n s :

Oceanography EMR December 1969.

J.Taylor .1976. "CONMAP : A computer program for contouring
oceanographih data." Marine Environmental
Data Service. Technical Note Nr. 12.
Environment Canada. Fisheries and Marine Service.

R e m a r k s :E r r o r C o n d i t i o n s :

DRWGRD

Subroutine DRWGRD (Z,NX,NY,ILTPGR,IPREGR)

P u r p o s e :

Draws lines between the gridlines and writes the value of each gridpoint centered on the gridpoints.

For undefined gridpoints nothing are shown.

NOTE : Works in a rectangular grid only !!!

P a r a m e t e r s :

Z	= Grid to be drawn.	Real array, Input
NX, NY	= X-, and Y-dimension of Z	Integer , Input
ILTPGR	= Linetype to use in the grid	Integer , Input
IPREGR	= Presentation style	Integer , Input
	No effect at present !!!!	

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 12.83, 10.86

R e m a r k s :

E r r o r C o n d i t i o n s :

DTPTS

Subroutine DTPTS (XP,YP,ZP,NP,LABPT,ISYMB,SSYMB,SLABL,LTYPE,IREPOS)

PORPOSE :

Plots and labels data points.

The arrays LABPT,ISYMB,SSYMB,SLABL,LTYPE and IREPOS describing the "layout" of the datapoints may be of dimension 1 in the Calling program, meaning all datapoints will have the same layout. In this case IREPOS(1) must be negative. If IREPOS(1) is positive the arrays LABPT,ISYMB,SSYMB,SLABL, LTYPE and IREPOS must have dimension NP in the Calling program and all points will be uniquely drawn.

P a r a m e t e r s :

XP,YP	= Coordinates of datapoints.	Real arrays,Input
ZP	= Value in datapoint	Real array ,Input
NP	= Number of datapoints to be drawn.	Integer,Input
LABPT	= Number of decials in dt.point.lbl	Integer array,Input

-1 = omit decimals, -2 = omit labels, -3 = omit datapoint
 ISYMB = Type of symbol to be plotted on datapoints Int. array, Input
 0 = omit symbols, 1 = "plus", 2 = "X", 3 = "asterix"
 4 = "square" 5 = "octagon" 6 = "diamond" 7 = "Z"
 8 = "triangle" 9 = "triangle upside-down"
 10 = "circle" 11 = "dot"
 SSYMB = Size of symbol in cm Integer array, Input
 SLABL = Size of characters of labels in cm Real array, Input
 LTYPE = Type of line between datapoints. Integer array, Input
 0 = no line 1 = solid line 2 = end point line
 3 = dotted line 4 = dashed line 5 = dash-dot line
 6..10 = user defined linetypes
 NOTE : When LTYPE is -1,-2...-10 lines are drawn between
 centers of the datapoints. When LTYPE is positive
 the lines stops just before the symbols.
 IREPOS = Position of symbol relative to geographic pos. Int.arr, Input
 NOTE : Positive values means variable attributes for each
 datapoint. IREPOS(1) negative means that all
 datapoints will get the same attributes.
 +-1 = Centered
 +-2 = Right
 +-3 = Above
 +-4 = Left
 +-5 = Under

R e f e r e n c e s / R e v i s i o n s :

T.WESTGÅRD, INSTITUTE OF MARINE RESEARCH, BERGEN. DES. 1983
 Modified by Tor Knutsen, Institute of Marine Research, Bergen.
 February 1987. May/June 1987. August 1987.

R e m a r k s :

E r r o r C o n d i t i o n s :

14,29

GEOPSI

Subroutine GEOPSI (LATOR,LONOR,LATPT,LONPT,X,Y,SCALE, IMAP)

P u r p o s e :

Knowing a points position on a map in cm the points geo-
 graphical coordinates is found in degrees and minutes as integers.
 North and east positive, south and west negative.

P a r a m e t e r s :

LATOR = Latitude of construction	point of map	Integer, Input
LONOR = Longitude of construction	point of map	Integer, Input
LATPT = The points latitude		Integer, Output
LONPT = The points longitude		Integer, Output

X,Y = Coordinates of th point in cm on the map Real ,Input
 SCALE = Inverse of scale of the map Real ,Input
 IMAP = Map-projection Integer Input
 1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical
 5 - Polarstreographic 6 - Polarstereographic, pole in center

R e f e r e n c e s / R e v i s i o n s :

Gunnar Helle, IMR, 12.83^o
 Andreas Christiansen, Cap Gemini, January 1988

R e m a r k s :

E r r o r C o n d i t i o n s :

GEOPSR

ENTRY GEOPSR (A1,A2,A3,A4,X,Y,SCALE, IMAP)

P u r p o s e :

Similar to GEOPSI but now the geographical coordinates is given in decimal degrees.

P a r a m e t e r s :

R e f e r e n c e s / R e v i s i o n s :

R e m a r k s :

E r r o r C o n d i t i o n s :

GETLEV

Subroutine GETLEV (Z,NX,NY,DZ,ZLEV,NLEV)

P u r p o s e :

Given the array Z(NX,NY) and the contour interval DZ
 the array ZLEV(NLEV) of contour levels is Returned.

P a r a m e t e r s :

Z	=	Real array, Input
NX	=	Integer , Input
NY	=	Integer , Input
DZ	=	Real , Input

ZLEV = Real array, Output
NLEV = Real , Input

R e f e r e n c e s / R e v i s i o n s :

J.Taylor .1976. "CONMAP : A computer program for contouring
oceanographih data." Marine Environmental
Data Service. Technical Note Nr. 12.
Environment Canada. Fisheries and Marine Service.

R e m a r k s :

E r r o r C o n d i t i o n s :

GRAFIN

Subroutine GrafIn (Ifunc)

P u r p o s e :

Subroutine for initiating a Tandberg terminal in
alfanum/graphical modes.

P a r a m e t e r s :

Ifunc : 1 = turn alfanum modus on Integer Input
 2 = turn graphical modus on
 3 = clear screen
 4 = clear graphical screen

R e f e r e n c e s / R e v i s i o n s :

Andreas Christiansen, Cap Gemini, September 1987

R e m a r k s :

E r r o r C o n d i t i o n s :

GRTCRL

Subroutine GRTCRL (LATOR,LONOR,YLAT1,XLON1,YLAT2,XLON2,SCALE,
& ILTPGC, IMAP)

P u r p o s e :

Draws a great circle from one geographical position to another
on a map.

Parameters :

LATOR	= Construction latitude of map	Integer,Input
LONOR	= Construction longitude of map	Integer,Input
YLAT1	= Latitude of point 1 in degrees	Real,Input
XLON1	= Longitude of point 1 in degrees	Real,Input
YLAT2	= Latitude of point 2 in degrees	Real,Input
XLON2	= Longitude of point 2 in degrees	Real,Input
SCALE	= Inverse of scale of map	Real,Input
ILTPGC	= Linetype to present the great circle	Integer,Input
IMAP	= Map-projection	Integer Input
	1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical	
	5 - Polarstreographic 6 - Polarstereographic, pole in center	

References / Revisions :

Trond Westgård, Institute of Marine Research, Bergen 04.84

Remarks : None.

Error Conditions :

GTU

Subroutine GTU (XN,XE,SM)

Purpose :

Transforms geographical coordinates to UTM coordinates in a given zone
METHOD: Series expansion

Parameters :

On Input:	XN = Latitude in degrees.	Real*8
	XE = Longitude in degrees.	Real*8
	SM = Central meridian of UTM zone	Real*8
	in degrees.	
On Output:	XN = UTM north-coordinate.	
	XE = UTM east-coordinate.	

References / Revisions :

Torstein Berge 1983. Norwegian Polar Research Institute, Oslo.

Remarks :

Error Conditions :

HELP4

Subroutine HELP4 (Z,NX,NY,M,L,M1,L1)

P u r p o s e :

Routine necessary in BLKCH1

P a r a m e t e r s :

R e f e r e n c e s / R e v i s i o n s :

Gunnar Helle, IMR, 12.83

R e m a r k s :

E r r o r C o n d i t i o n s :

HLP

Subroutine HLP (XOR,YOR,X1,Y1,XL,YL,DX,DY,NX,NY)

P u r p o s e :

When size of plotting medium and viewport/window are computed by CNVCH the positions in cm on the plotting device to get the routines CONTUR, INTGRT and DRWGRD to work are computed.
Input from : Common /Bhlp/ W(4),V(4),SIZE

P a r a m e t e r s :

XOR,YOR	= Origo of plotting device in cm.	Real Output
X1,Y1	= Lower left of chartframe in cm	Real Output
XL,YL	= Upper right of chartframe in cm	Real Output
DX,DY	= Step size of grid in cm	Real Output
NX,NY	= Dimension of grid Z(I,J)	Integer Input

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 12.83

R e m a r k s :

E r r o r C o n d i t i o n s :

IDGR

Integer Function IDGR (Xlalo)

P u r p o s e :

Converts a real position to an integer position to the nearest minuit.

(E.g. : Xlalo = 56.50 : IDGR = 5630)

P a r a m e t e r s :

R e f e r e n c e s / R e v i s i o n s :

R e m a r k s :

E r r o r C o n d i t i o n s :

IHELP3

Integer Function IHELP3 (Y,X)

P u r p o s e :

Computes the index of the smallest chartframe Y,X belongs to.

P a r a m e t e r s :

Y = Latitude in decimal degrees

Real, Input

X = Longitude in decimal degrees

Real, Input

IHELP3 = Returned index value (-1 = not found)

Integer, Output

R e f e r e n c e s / R e v i s i o n s :

Gunnar Helle, IMR, 12.83

Trond Westgård, IMR, 11.86

Andreas Christiansen, Cap Gemini, 12.87

R e m a r k s :

E r r o r C o n d i t i o n s :

INSIDE

Subroutine INSIDE (X,Y,XB,YB,NB,IND)

P u r p o s e :

Given a point X,Y and the series XB(K),YB(K)(K=1..NB) defining vertices of a Closed polygon. IND is set to 1 if the point is in the polygon and 0 if outside. Each time a new set of bound points is introduced IND should be set to 999 on input. It is best to do a series of Y for a single fixed X.

Method : A count is made of the no. of times the boundary cuts the meridian thru (X,Y) south of (X,Y). An odd count indicates the point is inside, even indicates outside.

See : "A long way from Euclid" by Constance Reid p 174 .

P a r a m e t e r s :

X,Y	= Coordinates of point to test.	Real,Input
XB,YB	= Vertices of a Closed polygon.	Real arrays,Input
NB	= Number of vertices in polygon	Integer,Input
IND	= Indicator to tell if point is outside or inside.	Integer,in/output

R e f e r e n c e s / R e v i s i o n s :

Oceanography EMR OCT/69

J.Taylor .1976. "CONMAP : A computer program for contouring oceanographih data." Marine Environmental Data Service. Technical Note Nr. 12.
Environment Canada. Fisheries and Marine Service.

R e m a r k s :

E r r o r C o n d i t i o n s :

31

INTGRT

Subroutine INTGRT (Z,NX,NY,LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,
& IUNIT,TOTAL, IMAP)

P u r p o s e :

Routine that integrates interpolated data in a geographical area

Parameters :

Z(NX,NY)	= Array to be integrated	Real array, Input
LATOR,LONOR	= Construction point of the map	Integer, Input
LAT1,LON1	= Geographical pos. of lower left corner	Integer, Input
LAT2,LON2	= Geographical pos. of upper right corner	Integer, Input
SCALE	= Inverse of scale of the map	Real, Input
IUNIT	= The unit of Z (1=unit/sq.meters, 2=unit/sq.nauticalmile)	Integer, Input
TOTAL	= Result of the integration	Real, Output
IMAP	= Map-projection	Integer Input
	1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical	
	5 - Polarstreographic 6 - Polarstereographic, pole in center	

References / Revisions :

Trond Westgård, IMR, 11.83

Remarks :Error Conditions :

INTRPL

Subroutine INTRPL (IU,L,X,Y,N,U,V)

Purpose :

Interpolation of a single valued function.
This subroutine interpolates, from values of the function given a ordinates of input data points in the x-y plane and for a given set of x-values(abcissas), the values of a single valued function $y=y(x)$.

Parameters :

Input :

IU = Logical unit number of standard output unit
L = Number of input data points
X = Array of dimension L storing the x-values (abcissas) of the data points in ascending order
Y = Array of dimension L storing the y-values (ordinates) of the input data points
N = Number of points at which interpolation of the y-values (ordinates) is desired
U = Array of dimension N storing the x-values of the desired points.

Output :

V = Array of dimension N where the interpolated Y values are stored.

R e f e r e n c e s / R e v i s i o n s :

This algorithm was published in Comm. Acm. 15(10)oct 1972
by Hiroshi Akima, U.S.dept of commerce, Office of
Telecommunications,
Institute of telecommunications sciences, Boulder, Colorado.

R e m a r k s :

E r r o r C o n d i t i o n s :

17,18,19

LATLON

Subroutine LATLON (X,Y,Z,YLAT,XLON)

P u r p o s e :

Finds latitude (YLAT) and longitude (XLON) of a point in real degrees
when its cartesian coordinates (XYZI) on a sphere is known.

P a r a m e t e r s :

YLAT = Latitude of the position (south negative)	Real ,Output
YLON = Longitude of the position (west negative)	Real ,Output
X,Y,Z = Coordinates of the position	Real ,Input

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 05.85

R e m a r k s : None

E r r o r C o n d i t i o n s :

MLERR1

Subroutine MLERR1 (Ichoic)

P u r p o s e :

Prints or checks error message reported from a Map-Library routine.
The error message will be stored in variable Chmess in
Common block MAPER2. Mlerr is not reset to zero.

Parameters :

Ichoic = Option to display error message Integer, Input
 0 = No print out, but updates Chmess
 1 = Prints message on the screen
 2 = Prints message on the screen and terminates program

All other input/output through COMMON. See routine MLERR2.

References / Revisions :

Tor Knutsen, IMR, 03.87
 Trond Westgård, IMR, 04.88

Remarks :Error Conditions :

1,26,27

MLERR2

Subroutine MLERR2

Purpose :

Updates the error table returned from Map-Library in a way ascertains that each message will appear only once in the list of error messages in several calls to Map-Library routines.

Parameters :

All input/output are through COMMON BLOCK /MAPER1/ and /MAPER2/

Mlerr : Status variable containing error message number
 Chmess : Variable containing error message, max 80 characters
 Imestb : Table storing error message numbers.
 Imesno : Number of different error messages occurred.

NOTE : When the error message table in MLERR1 is increased Itab must also increase.

PARAMETER (Mlengt = 80, Itab = 32)
 Integer Mlerr, Ivalg, Imestb(Itab), Imesno
 CHARACTER Chmess*(Mlengt)

COMMON /MAPER1/ Mlerr, Imestb, Imesno
 COMMON /MAPER2/ Chmess

References / Revisions :

Trond Westgård, IMR, 04.88

Remarks :

Error Conditions :

None.

MLERR3

Subroutine MLERR3

Purpose :

A routine for writing error messages to the screen/terminal

Parameters :

Input/Output through common (See MLERR2)

References / Revisions :

Tor Knutsen, Institute of Marine Research, Bergen, Norway. 12.87

Remarks :Error Conditions :

MLERR4

Subroutine MLERR4 (IOK)

Purpose :

Checks if the same error message is registered before.
Input also through common (See MLERR2).

Parameters :

IOK = Status variable. Integer, Output
0 = Error message not registred before
1 = Error message registred before

References / Revisions :

Tor Knutsen, IMR, 12.87

Remarks :Error Conditions :

NCHAR

Subroutine NCHAR (X,NDIG)

P u r p o s e :

Determines the number of digits necessary to present a real number.

P a r a m e t e r s :

X = Real number to be presented. Real Input
NDIG = Number of characters necessary Integer Output

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 12.83

R e m a r k s :

E r r o r C o n d i t i o n s :

16

PLANE

Subroutine PLANE (X1,Y1,Z1,X2,Y2,Z2,A,B)

P u r p o s e :

Finds the coefficients in the equation of a plane $Z=A*X + B*Y$

P a r a m e t e r s :

X1,Y1,Z1 = Coordinates of point 1 in the plane. Reals, Input
X2,Y2,Z2 = Coordinates of point 2 in the plane. Reals, Input
A,B = Coefficients in the equation Reals, Output

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 01.84

R e m a r k s : None.

E r r o r C o n d i t i o n s :

PLTMRK

Subroutine PLTMRK (ISYMB,SSYMB)

P u r p o s e :

Plots a symbol of type ISYMB of size SSYMB cm centred around the current point of the plotting device.

P a r a m e t e r s :

ISYMB = Type of symbol to be plotted. Integer, Input
1 = "plus" 5 = "octagon" 9 = "triangle upside-down"
2 = "X" 6 = "diamond" 10 = "circle"
3 = "asterix" 7 = "Z" 11 = "dot"
4 = "square" 8 = "triangle"

SSYMB = Size of symbol in cm. Real, Input

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 12.83

R e m a r k s :

E r r o r C o n d i t i o n s :

POLAR

Subroutine POLAR (POLX, POLY, N, AREA, SM, IGRAD)

P u r p o s e :

Computes the area of on the surface of the globe. If the coordinates are in geographical degrees, they will be transformed into UTM-coordinates with central longitude SM. The area is found by using triangulation computation. First point in the polygon is fixed, whilst the other points are used sequentially two and two as the other corners in the triangles. The area is either positive or negative

P a r a m e t e r s :

POLX = X-coordinates of the polygon Real array Input
POLY = Y-coordinates of the polygon Real array Input
N = Number of coordinates Integer Input
AREA = Computed area Real Output
SM = Central longitude (when IGRAD > 0). Real Input
IGRAD = >0 coordinates in degrees, else in meters. Integer Input

R e f e r e n c e s / R e v i s i o n s :

Torstein Berge. (The Norwegian Polar Research Institute)

R e m a r k s :E r r o r C o n d i t i o n s :

----- PRJCTA -----

Subroutine PRJCTA (LAT1,LON1,LAT2,LON2,XLAT,XLON,XX,NX,DIST,
& YLTP,XLNP,NN)

P u r p o s e :

Projects the real positions of the stations onto the great circle passing through (LAT1,LON1) and (LAT2,LON2) and checks if the projection is between these two points .

LAT1 = Latitude of start of the section	Integer,Input
LON1 = Longitude of start of the section	Integer,Input
LAT2 = Latitude of stop of the section	Integer,Input
LON2 = Longitude of stop of the section	Integer,Input
XLAT = Latitudes of the stations	Real array,Input
XLON = Longitudes of the stations	Real array,Input
XX = Distances in km the stations should be placed away from (LAT1,LON1) along the section line.	Real array,Output
For some stations it is not possible to place the projection of it between (LAT1,LON1) and (LAT2,LON2).	
NX = Number of stations to be examined	Integer,Input
DIST = Accepted distance in km a station can be apart from the great circle passing (LAT1,LON1) and (LAT2,LON2)	Real,Input
YLTP = Latitudes of the projections of the stations if it is not possible to project the station yltp is returned with the value -200.	Real,Output
XLNP = Longitudes of the projections of the stations if it is not possible to project the station xlnp is returned with the value -200.	Real,Output
NN = Number of projections, equal to nx if projections are wanted, Else zero.	Integer, Input

R e f e r e n c e s / R e v i s i o n s :

Tor Knutsen, IMR, 3.88

R e m a r k s :E r r o r C o n d i t i o n s :

PRJCTN

Subroutine PRJCTN (LAT1,LON1,LAT2,LON2,LATX,LONX,XX,NX,DIST,
& YLTP,XLNP,NN)

P u r p o s e :

Projects the positions of the stations onto the great circle passing through (lat1,lon1) and (lat2,lon2) and checks if the projection is between these two points .

P a r a m e t e r s :

LAT1 = Latitude of start of the section	Integer, Input
LON1 = Longitude of start of the section	Integer, Input
LAT2 = Latitude of stop of the section	Integer, Input
LON2 = Longitude of stop of the section	Integer, Input
LATX = Latitudes of the stations	Integer array, Input
LONX = Longitudes of the stations	Integer array, Input
XX = Distances in km the stations should be placed away from (LAT1,LON1) along the section line.	Real array, Output
For some stations it is not possible to place the projection of it between (LAT1,LON1) and (LAT2,LON2).	
NX = Number of stations to be examined	Integer, Input
DIST = Accepted distance in km a station can be apart from the great circle passing (LAT1,LON1) and (LAT2,LON2)	Real ,Input
YLTP = Latitudes of the projections of the stations	Real arr, Output
If it is not possible to project the station YLTP is Returned with the value -200.	
XLNP = Longitudes of the projections of the stations	Real arr, Output
If it is not possible to project the station XLNP is Returned with the value -200.	
NN = Number of projections, equal to NX if projections are wanted Else zero.	Integer, Input

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, Institute of Marine Research, Bergen, Norway 01.84

R e m a r k s : None.

E r r o r C o n d i t i o n s :

RDGR

Real Function RDGR (LATLON)

P u r p o s e :

This function converts an "integer" position to a "real" position.
(E.g. : If LATLON = 5630 : RDGR = 56.50)

P a r a m e t e r s :

R e f e r e n c e s / R e v i s i o n s :

R e m a r k s :

E r r o r C o n d i t i o n s :

Error 15

REDDIG

Subroutine REDDIG (ISTNO,NST,INEW)

P u r p o s e :

Reduces number of digits in array of station numbers.
When numbering the stations along the section no overlap occur.

P a r a m e t e r s :

ISTNO = Array containing active station numbers.	Int. arr,Input
NST = Number of active stations	Int. Input
INEW = Array to store the modified station numbers.	Int. arr,Output

R e f e r e n c e s / R e v i s i o n s :

Tor Knutsen, IMR, 1988

R e m a r k s :

E r r o r C o n d i t i o n s :

32,33

SMOOTH

Subroutine SMOOTH (Z,NX,NY,NSM)

P u r p o s e :

Array Z(I,J) is smoothed using Laplacian smoothing NSM times.

I.e : $Z = Z + .25 * (\text{Average}(Z_N, Z_S, Z_E, Z_W) - Z)$

The sweep is alternately sw to ne and ne to sw.

ZXX and ZYY are assumed to be zero at edges.

Unused points in Z should be 10^{35} .

P a r a m e t e r s :

Z(I,J)	= Array to be smoothed.	Real array, in/output
NX,NY	= Dimension of Z(I,J)	Integer, Input
NSM	= Number of times to smooth Z(I,J)	Integer, Input

R e f e r e n c e s / R e v i s i o n s :

Oceanography EMR OCT/69 VERSION NO. 2 IMPROVED EDGING

J.Taylor .1976. "CONMAP : A computer program for contouring oceanographic data." Marine Environmental Data Service. Technical Note Nr. 12. Environment Canada. Fisheries and Marine Service.

R e m a r k s :

E r r o r C o n d i t i o n s :

STRVAL

Subroutine StrVal (Cstrng, Istart, Islutt)

P u r p o s e :

Finds the first and last non-blank character in a string.

P a r a m e t e r s :

Cstrng	: Input character string	Character, Input
Istart	: First non-blank character	Integer, Output
Islutt	: Last non-blank character	Integer, Output

R e f e r e n c e s / R e v i s i o n s :

Andreas Christiansen, Cap Gemini, July 1988

R e m a r k s :

This routine is meant to replace the ND Fortran extension (-1:-1) as character string subscript.

E r r o r C o n d i t i o n s :-----
TEXT

Subroutine TEXT (TXT,NTXTLN,STXT,SFRM)

P u r p o s e :

Writes a text under the plot. CNVCH or CNVCS1 must be Called.

P a r a m e t e r s :

TXT	= Array of characters,maximum 78 characters in each line. i.e. txt(ntxtln)*78	Character,Input
NTXTLN	= Number of textlines (1,2.....)	Integer ,Input
STXT	= Size of characters in text in cm	Real ,Input
SFRM	= Size of characters in frame in cm	Real ,Input

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, Institute of Marine Research, Bergen. 12.83

R e m a r k s : None

E r r o r C o n d i t i o n s :-----
UPDTP

Subroutine UPDTP (L,LIN,NN)

P u r p o s e :

Updates a geographical position one stepsize

P a r a m e t e r s :

L	= Latitude or longitude	Integer,in/output
LIN	= The stepsize (always positive)	Int.,Input
NN	= Stepdirection (north, east n>0 ; west, south, N<0)	Int.,Input

R e f e r e n c e s / R e v i s i o n s :

Gunnar Helle, IMR, 12.83

R e m a r k s :

E r r o r C o n d i t i o n s :

UTG

Subroutine UTG (XN,XE,SM,NS)

P u r p o s e :

Transforms a point in UTM-coordinates (XN,XE) into geographical coordinates in centesimal degrees using series expansion.

P a r a m e t e r s :

On Input :	XN = UTM north-coordinate.	Real*8
	XE = UTM east-coordinate	Real*8
	SM = Central meridian in the UTM-zone in degrees.	Real*8
	NS = Hemisphere	Integer
	+1 = North, -1 = South	

On Output: XN = Latitude in degrees.
XE = Longitude in degrees.

R e f e r e n c e s / R e v i s i o n s :

Torstein Berge. 09/1983. Norwegian Polar Research Institute, Oslo.

R e m a r k s :

E r r o r C o n d i t i o n s :

VALID

Subroutine VALID (LAT1,LON1,LAT2,LON2)

P u r p o s e :

Checks if latitudes and longitudes given to COAST,CSTTRU,
and BLKCH1 is valid with the present files with coastline data.

P a r a m e t e r s :

LAT1 = South border of chartframe	Integer, Input
LAT2 = North border of chartframe	Integer, Input
LON1 = West border of chartframe	Integer, Input

LON2 = East border of chartframe Integer, Input

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 11.86

R e m a r k s :

E r r o r C o n d i t i o n s :

XYCONI

Subroutine XYCONI (X,Y)

P u r p o s e :

Knowing a points position on the plotting device in cm and given the window and viewport coordinates computed by CNVCH or CNVCS1 the right position on a map relative to the construction point in cm is computed.

P a r a m e t e r s :

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 12.84

R e m a r k s :

E r r o r C o n d i t i o n s :

XYCONV

Subroutine XYCONV (X,Y)

P u r p o s e :

Transforms a position in cm on the map to the correct position on the plotting device using the window and viewport coordinates computed in CNVCH or CNVCS1.

P a r a m e t e r s :

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 12.83

R e m a r k s :

Error Conditions :-----
XYPSI

Subroutine XYPSI (LATOR,LONOR,LATPT,LONPT,X,Y,SCALE, IMAP)

P u r p o s e :

Finds the geographical point(latpt,lonpt)'s position on a map in cm.
North and east positive, south and west negative.

P a r a m e t e r s :

LATOR = Construction latitude of map	Integer,Input
LONOR = Construction longitude of map	Integer,Input
LATPT = Latitude of point	Integer,Input
LONPT = Longitude of point	Integer,Input
X = X-coordinate of point on map in cm	Real ,Output
Y = Y-coordinate of point on map in cm	Real ,Output
SCALE = Inverse of scale of map	Real ,Input
IMAP = Map-projection	Integer Input
1 - XY, 2 - UTM, 3 - Mercator, 4 - Conical	
5 - Polarstreographic 6 - Polarstereographic, pole in center	

R e f e r e n c e s / R e v i s i o n s :

Gunnar Helle, IMR, 12.83
Andreas Christiansen, Cap Gemini, January 1988 using routines
from : Torstein Berge. 1987. PLOTLIB-1, version F. Norwegian
Institute of Polar Research, Oslo, Norway.

R e m a r k s :E r r o r C o n d i t i o n s :-----
XYPSR

ENTRY XYPSR (A1, A2, A3, A4, X, Y, SCALE, IMAP)

P u r p o s e :

Similar to XYPSI but now geographical coordinates is given as real
numbers in decimal degrees

P a r a m e t e r s :R e f e r e n c e s / R e v i s i o n s :R e m a r k s :

Error Conditions :-----
XYZI

Subroutine XYZI (LAT,LON,X,Y,Z)

P u r p o s e :

Finds a points X,Y,Z coordinates when its latitude and longitude is known.

P a r a m e t e r s :

LAT = Latitude of the position (south negative)	Integer Input
LON = Longitude of the position (west negative)	Integer Input
X,Y,Z= Coordinates of the position	Real Output

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 01.84

R e m a r k s : NONE.E r r o r C o n d i t i o n s :-----
XYZR

ENTRY XYZR (XLAT,XLON,X,Y,Z)

P u r p o s e :

Same as XYZI but now input is given in decimal degrees

P a r a m e t e r s : See XYZIR e f e r e n c e s / R e v i s i o n s : See XYZIR e m a r k s : None.E r r o r C o n d i t i o n s :

ZGRI

Subroutine ZGRI (Z,NX,NY,XP,YP,ZP,NP,CAYIN,NRNG,NGPS)

P u r p o s e :

Interpolates datapoints into a 2-D grid. See also ZGRID.

P a r a m e t e r s :

Z(I,J)	= Array to be set up. points outside region to be contoured should be initialized to 10^{35} .	Real arr, in/output
NX,NY	= Max dimension of Z(I,J)	Integer,Input
XP(I)	= X-coordinat of ZP(I)	Real array,Input
YP(I)	= Y-coordinat of ZP(I)	Real array,Input
ZP(I)	= The datapoints that are given. (I=1,NP)	Real array,Input
NP	= The number of datapoints	Integer,Input
CAYIN	= Amount of spline equation the user want. (values from 0.0 to infinity is allowed) 0.0 gives a 2-D linear interpolation High values gives a spline interpolation	Real,Input
NRNG	= Points in Z(I,J) more than NRNG gridspace From nearest datapoint is set to undefined.	Integer,Input
NGPS	= If elements in Z(I,J) are allowed to have Negative values set ngps to -1 Else ngps = +1	Integer,Input

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, IMR, 12.83
See also ZGRID

R e m a r k s :

E r r o r C o n d i t i o n s :

ZGRID

Subroutine ZGRID (ZPIJ,KNXT,IMNEW,Z,NX,NY,X1,Y1,DX,DY,XP,YP,ZP,N,
> CAYIN,NRNG)

P u r p o s e :

Sets up square grid for contouring, given arbitrarily placed data points. laplace interpolation is used.

P a r a m e t e r s :

ZPIJ	=	Real array,
KNXT	=	Int. array,

IMNEW = Int. array,
 Z(I,J) = Array to be contoured. Real array, in/output
 Points in Z(I,J) outside region to be
 contoured should be initialized to 10^{35} .
 The rest should be 0.
 NX,NY = Max subscripts of Z(I,J). Integers, Input
 X1,Y1 = Coordinates of Z(1,1) Real, Input
 DX,DY = X and Y increments . Real, Input
 XP,YP,ZP = Arrays with pos. and value of data points. Real, Input
 N = Size of arrays XP,YP AND ZP . Integer,Input
 CAYIN = Amount of spline eqn (between 0 and inf.) Real, Input
 0 = pure laplace solution.
 inf. = pure spline solution
 NRRNG = Grid points more than NRRNG grid spaces from Integer Input
 nearest data point are set to undefined.

R e f e r e n c e s / R e v i s i o n s :

The method used here was lifted directly from notes left by
 Mr Ian Crain formerly with the Comp.Science.Div.
 info on relaxation soln of laplace eqn supplied by Dr T Murty.
 FORTRAN II Oceanography/EMR DEC/68 JDT
 Modification FEB/69 to get smoother results a portion of the
 beam eqn. was added to the Laplace eqn. giving :
 $\text{DELTA}2\text{X}(Z)+\text{DELTA}2\text{Y}(Z) - \text{K}(\text{DELTA}4\text{X}(Z)+\text{DELTA}4\text{Y}(Z))=0$.

DEC23/69 Data pts. no longer moved to grid pts.
 76 No printout.
 Modified to FORTRAN-77 04/1985. T.Westg rd IMR,Bergen.

J.Taylor .1976. "CONMAP : A computer program for contouring
 oceanographih data." Marine Environmental
 Data Service. Technical Note Nr. 12.
 Environment Canada. Fisheries and Marine Service.

R e m a r k s :

E r r o r C o n d i t i o n s :

10

----- ZONE -----

Subroutine ZONE (YLAT1,XLON1,YLAT2,XLON2,DIST,YLATC,XLONC)

P u r p o s e :

The routine finds the four corners of an area that is specified
 by a great circle from YLAT1,XLON1 to YLAT2,XLON2 and all points
 nearer than DIST km from this great circle.

P a r a m e t e r s :

YLAT1 = Latitude of start of section in degrees (s is neg.) Real,Input

XLON1 = Longitude of stop of section in degrees (w is neg.) Real,Input
 YLAT2 = Latitude of stop of section in degrees (s is neg.) Real,Input
 XLON2 = Longitude of stop of section in degrees (w is neg.) Real,Input
 DIST = Half the width of the sector in km Real,Input
 YLATC = Latitudes of the four corners of the sector Real arr,Output
 XLONC = Longitudes of the four corners of the sector Real arr,Output

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, Institute of Marine Research, Bergen, Norway 08.84

R e m a r k s : None.

E r r o r C o n d i t i o n s :

ZONHLP

Subroutine ZONHLP (A1,B1,CORD,X1,Y1,Z1,XN,YN,ZN,YLATC1,XLONC1,
 * YLATC2,XLONC2,NN)

P u r p o s e :

Does calculations needed in zone

P a r a m e t e r s :

No explanation given

R e f e r e n c e s / R e v i s i o n s :

Trond Westgård, Institute of Marine Research, Bergen, Norway, 08.84

R e m a r k s : None.

E r r o r C o n d i t i o n s :

5,6

ZZHALF

Function ZZHALF (ZZ,BIG)

P u r p o s e :

INTERPOLATION TO GET ZZ(2.5) .

P a r a m e t e r s :

R e f e r e n c e s / R e v i s i o n s :

Oceanography EMR 1969

R e m a r k s :

E r r o r C o n d i t i o n s :

8.5 Filling of topographical levels

In the present manual this section will overrule section 3.9 when describing filling of topographical levels with colours, and will replace section 3.9 in the next version of this Users Guide.

The filling of topographical levels with colours or hatching on a map is a difficult problem that is handled by Map-Library and the other programs of FORSKERKART. Suppose you have a rectangle with the level of the four corners and a set of curves describing the outlook of the contour lines for the different levels present. Assuming that each curve either starts and end on one of the four boundaries of the rectangle or is closed into itself. Then it is possible to build a set of closed polygons that has a level associated with it. The pipeline for geographical data included in FORSKERKART (see part 3 and 4 of this user guide) assures that these requirements are fulfilled. On this background The Chr. Michelsen Institute in Bergen developed an integer function Zcont, based on the specifications given by The Institute of Marine Research in Bergen, that builds polygons when the curves are given.

The call interface is,

```

status = Zcont(vLevel, vXc, vYc, nv
+           , cXc, cYc, nPoint, cCount, cLevel, nLines
+           , pXc, pYc, nOut, maxOut
+           , pLevel, pCount, nPoly, maxPol
+           , iWork, lthi, rWork, lthf)

```

where :

vLevel	= Level of corners	int.arr.	input
vXc	= x-coordinates of corners.	real.arr.	input
vYc	= y-coordinates of corners.	real.arr.	input
nv	= number of corners(=4)	int.	input
cXc	= x-coordinates for the points on the curves.	real arr.	input
cYc	= y-coordinates for the points on the curves.	real arr.	input
nPoint	= Total number of points on curves.	int.	input

cCount = Number of coordinates in each curve.	int. arr. input
cLevel = Level of each of the curves.	int. arr. input
nLines = Number of curves.	int. input
pXc = x-coordinates of points in polygons.	real arr.output
pYc = y-coordinates of points in polygons.	real arr.output
nOut = number of points in polygons.	int. output
maxOut = Maximum size of nOut.	int. input
pLevel = Level of each polygon.	int. arr.output
pCount = Number of vertices in each polygon.	int. arr.output
nPoly = Number of polygons build by Zcont.	int. output
maxPol = Maximum size of nPoly.	int. input
iWork = Integer data working array.	int. arr. in/output
lthi = Length of iWork.	int. input
rWork = Floating data working array.	real arr. in/output
lthf = Length of rWork.	real input

Note that the levels of curves are given as an index (cLevel) to the common declared array Depths containing the level values. If Zcont returns the value zero the polygons are build correctly, otherwise an error has occurred. The Map-Library error messages that has been added to the list in section 8.2 of this user guide are :

MLERR CHMESS

- 36 Too many lines in fill operation in COAST.Increase Mlines !
- 37 Too many points in fill operation in COAST.Increase Mpoint!
- 38 Fill error in COAST. Line not on border !
- 39 Fill error in COAST. Too many lines on border !
- 40 Fill error in COAST. Illegal order of points on border !
- 41 Fill error in COAST. Too few output polygons !
- 42 Fill error in COAST. Boundary lines not hor/vertical !
- 43 Fill error in COAST. Out of stack space !
- 44 Fill error in COAST. Too few output coordinates !
- 45 Fill error in COAST. Mismatch between nPoint and sum cCount(j)
- 46 Fill error in COAST. Impossible to detect level of contours
- 47 Fill error in COAST. Polygon with less than 3 points !

These error messages may occur when you use the filling option in

routine COAST (see below). When you use Zcont as a stand alone routine the return of non-zero values from 1 to 10 correspond to the Map-Library errors 38-47 in the table given above. Guide lines for dimensioning the arrays rWork and iWork are :

$$lthi = 3 * (nLines + 1) + 2 * maxCross$$

where :

nLines = number of curves in the rectangle.

maxCross = number of crossings between the curves and rim of rectangle
Each curve not closed in itself generates two crossings,
i.e. maxCross is always less than nLines*2.

If lthi is given a value of (5*nLines + 3) it is always large enough.

$$lthf = 3 * nLines + maxCross$$

If lthf is given a value of (6*nLines) it is always large enough.

The user can now call COAST in the same way as in section 3.2 to fill a geographical area with colour or hatching. Some of the variables has to be assigned other values than the usual.

- When NLEVEL is negative on input, hatch or solid fill of topographical levels are done.
- When hatch/fill is chosen (NLEVEL less than zero) be aware that LINCOL must be declared to the number of levels plus one (e.g. presence of coastlines implies presence of two levels sea and land).
- The colour of each topographical level is set by LINCOL(NLEVEL+1).
- When LINCOL(ilev) is negative the level will be filled with hatch/solid fill. Default is solid fill if the unit supplies it.
- The common block /HTC/ in file (kart)new-mapfill:incl lets the

user control the hatching angle and hatching distance.

```
Real HTCANG (nlvmax+1), HTCDST (nlvmax+1)
COMMON /HTC/ HTCANG(nlvmax+1),HTCDST(nlvmax+1)
```

- Parameter `nlvmax` is the maximum number of topographical levels that any map database opened in BASINI may contain. The parameter is declared in the file `(kart)new-maplib:incl`.
- Each element in the real array `HTCANG(nlvmax+1)` is assigned a default value of `-99.0`, i.e. solid fill are chosen. When the user includes the statement `$Include (kart)New-MapFill:incl` in his program and gives elements in `HTCANG(nlvmax+1)` values different from `-99.0` the corresponding levels are hatched with the given angle.
- The distance in cm between the hatched lines in each level are defined in the real array `HTCDST(nlvmax+1)`. This array is also stored in the common block `/HTC/`.

The program below is a complete working example showing how the user may specify a map using a Tektronix-4107 screen and the Tektronix-4693D colour copy unit.

```
C ** PROG-EXAMPLE-12 **
C
C Draws a chart frame with text, coastlines and
C filling between different topographical levels.
C-----
C
C Includes common blocks describing topographical database,
C error messages and common block to control hatching/filling
C between topographical levels.
C
    $INCLUDE (kart)New-MapLib:incl
    $INCLUDE (kart)New-MapError:incl
    $INCLUDE (kart)New-MapFill:incl
C
CHARACTER TXT(2)*50, MapBas*10, UseRef*32
DIMENSION WF(4), VF(4), SIZE(6), Iopt (3)
INTEGER LEVELS (nlvmax),LINTYP(nlvmax),LINCOL(nlvmax+1)
DATA SFRM, NTXTLN, STXT /0.3, 2, .8/
DATA LATIN, NLAT /0030, 0/
DATA LONIN, NLON /0500, 0/
DATA ILTPFM /1/
C
Write (*,'(1H$,A)') 'Give the plots position on medium in cm. : '
```

Part 5 of 6 : Map-Library.

```

Read (*,*) WF(1),WF(2),WF(3),WF(4)
Write (*,'(1H$,A)') 'Give NLEVEL (value < 0 means fill) : '
Read (*,*) NLEVEL
Do 10 i = 1,IABS(NLEVEL),1
  Write (*,'(1H$,A,I3,A)') 'Give colour for level',i,' : '
  Read (*,*) lincol(i)
  Write (*,'(1H$,A,I3,A)') 'Give hatchangle',i,' in degrees',
+      ' (-99.0 = solid) : '
  Read (*,*) htcang (i)
  Write (*,'(1H$,A,I3,A)') 'Give hatch distance ',i,' in cm : '
  Read (*,*) htcdst (i)
  Write (*,'(1H$,A,I3,A)') 'Give line type',i,' : '
  Read (*,*) lintyp(i)
  Write (*,'(1H$,A,I3,A)') 'Give index for level',i,' : '
  Read (*,*) levels(i)
10 Continue
  If (NLEVEL.lt.0) then
C The user want hatching/ solid fill :
  i = IABS(NLEVEL) + 1
  Write (*,'(1H$,A,I3,A)') 'Give colour for level',i,' : '
  Read (*,*) lincol(i)
  Write (*,'(1H$,A,I3,A)') 'Give hatchangle',i,' in degrees',
+      ' (-99.0 = solid) : '
  Read (*,*) htcang (i)
  Write (*,'(1H$,A,I3,A)') 'Give hatch distance ',i,' in cm : '
  Read (*,*) htcdst (i)
  Endif
C
C Initiates the wanted topographical database :
C
  Write (*,'(1H$,A)') 'Give name of map database : '
  Read (*,*) MapBas
  Write (*,'(1H$,A)') 'Give user reference of map database : '
  Read (*,*) UseRef
  CALL BASINI ( MapBas, UseRef, ISTAT)
C
  TXT(1)='Prog-Example-12. Plots topographical levels.'
  TXT(2)='A chartframe with text, coastlines and depthlines.'
C
  Write (*,'(1H$,A)') 'Give projection type : '
  Read (*,*)      Imap
C
  Write (*,'(1H$,A)') 'Give LAT1, LON1, LAT2 and LON2 : '
  Read (*,*) LAT1,LON1,LAT2,LON2
  CALL GPGS
  Write (*,'(1H$,A)') 'Give number of driver (4589 or 59) : '
  Read (*,*) IDEV
  If (IDEV.eq.4589) then
    IDEV = 89
    IOPT(1) = 45
    IOPT(2) = 0
    IOPT(3) = 1
    call nitopt(iopt(1),3)
  Elseif (idev .eq. 59) then
    call AsciiC (1,'%!0',3)
  Endif
  CALL NITDEV (IDEV)

```

```

CALL CNVCH (WF,VF,LATOR,LONOR,LAT1,LON1,LAT2,LON2,
+          SFRM,SCALE,NTXTLN,STXT, IMAP)
CALL WINDW (WF)
CALL VPORT (VF)
C
CALL BGNPIC (1)
CALL CFONT (1)
if (idev.eq.89) Call asciiC (45,'MW3;5)
CALL CHFRM (LATOR, LONOR, LAT1, LON1,
+          LAT2, LON2, LATIN, NLAT,
+          LONIN, NLON, SFRM, SCALE, ILTPFM, IMAP)
if (idev.eq.89) Call asciiC (45,'MW3;5)
CALL COAST (LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,
+          IMAP, NLEVEL, LEVELS, LINTYP, LINCOL)
CALL CFONT (3)
if (idev.eq.89) Call asciiC (45,'MW3;5)
CALL TEXT (TXT,NTXTLN,STXT,SFRM)
CALL ENDPIC
C
CALL RLSDEV (IDEV)
END
C
C ** AsciiC **
C
C Purpose :
C
C Send a command sequence to a graphical terminal (iunit=1) or a plotter
C in Teklike i.e. ANSI like style as ESC-sequences
C All parameters in string are printable ASCII characters.
C
C Trond Westgård, Institute of Marine Research, Bergen.
C 27.06.1988
C
C Subroutine AsciiC (iUnit,String,Lstrng)
C
C iUnit = The number of the output unit.
C String = Character string to send to Tek-4107 (without the ESC !!!)
C Lstrng = Number of characters in string.
C If Lstrng is negative on input no ESC is send.
C If Lstrng is equal to zero only an ESC is send.
C
C NOTE : Outch is machine dependent !!!!!
C End Head
C
C Character String * (*)
C Integer iUnit, Lstrng
C
C Sends an escape :
C If (Lstrng .ge. 0) Call Outch (iUnit,27)
C If (Lstrng .eq. 0) Return
C If (Lstrng .lt. 0) Lstrng = -Lstrng
C Do 10 i = 1,Lstrng,1
C Send an ascii-character :
C Call Outch (iUnit,ichar(String(i:i)))
10 Continue
End

```



Prog-Exemple-12. Plots topographical levels.

A char/frame with text, coastlines and depthlines.

Fig. 12. Filling of sea and coastlines

8.6 Graphical Kernel System (GKS).

The need for standardization, initiated the project of making a GKS-version of Map-Library. The GKS installed on the NORSK DATA 500 computers at the institute is MGKS from CERN (level MB), and it may be loaded from the file (GKS)MGKS-LIB:NRF

This chapter describes only what is specific to GKS. Problems concerning Map-Library in general is described elsewhere in this Users Guide.

8.6.1 Using Map-Library with GKS.

Some GKS-routines must always be present in programs using the GKS-version of Map-Library. The following two examples show the minimum number of statements to include when starting and ending a program.

It is important to remember that 'normalization transformation number' must be 1 when using Map-Library.

In order to keep the call syntax equal in the GKS- and GPGS-versions, the 'workstation identification number' must be 1 for the workstation used by Map-Library.

In the first example the library GKS-UTIL (see page 128) is not used.

```
C ** Users program **
C
C WF is size of map in cm,   VF is NDC viewport.
C
C     DIMENSION wf (4), vf (4)
C
C IWID is the workstation identification number. Always 1 for Map-Library.
C IWTP is the workstation type, in this case a Tandberg screen.
C IERR is the error file.
C
C     iwid  = 1
C     iwtyp = 401400
C     ierr  = 14
C
C
C Open error file, open GKS, open and activate workstation.
C
```

```

      OPEN (ierr, file = 'gksererror')
C
      Call GOPKS (ierr, idum)
      Call GOPWK (iwid, 1, iwtyp)
      Call GACWK (iwid)
      .
      .
C
C Select normalization transformation. Always number 1 for Map-Library.
C
      Call GSELNT (1)
      .
      .
C The users generation of a picture
      .
      .
C
C Deactivate and close workstation, close GKS and close errorfile.
C
      Call GDAWK (iwid)
      Call GCLWK (iwid)
      Call GCLKS
C
      CLOSE (ierr)
      END

```

In the second example the library GKS-UTIL (see page 128) is used.

```

C ** Users program **
C
C WF is size of map in cm, VF is NDC viewport.
C
      DIMENSION wf (4), vf (4)
C
C IWID is the workstation identification number. Always 1 for Map-Library.
C IWTP is the workstation type, in this case a Tandberg screen.
C
      iwid = 1
      iwtyp = 401400
C
C Open error file, open GKS, open and activate workstation.
C
      Call GHISTA (iwid, iwtyp)
      .
      .
C
C Select normalization transformation. Always number 1 for Map-Library.
C
      Call GSELNT (1)
      .
      .
C The users generation of a picture
      .
      .

```

```
C
C Deactivate and close workstation, close GKS and close errorfile.
C
    Call GHIEND
    END
```

8.6.2 Deviations from standard GKS.

MGKS contains one statement where the syntax is different from the standard GKS. This is not likely to happen, but should it occur in other versions of GKS, it is advised to follow the standard syntax in the user programs, and make a GKS-STD library. The MGKS-STD contains the following routine, and should be loaded together with MGKS-LIB.

```
Subroutine GQDSP (itype, ierr, iunit, rx, ry, lx, ly)
    Call GQMDS (itype, ierr, iunit, rx, ry, lx, ly)
return
end
```

8.6.3 Compatibility with GPGS-version.

Even if GKS has a different way of mapping from 'world' to 'workstation', the GKS-version of Map-Library uses the WF and VF arrays exactly the same way as the GPGS-version.

The call interface is the same as in the GPGS-version with one exception, the subroutine PLTMRK has two extra parameters describing the current position.

```
C
C ** PLTMRK **
C
    Subroutine PLTMRK (ISYMB, SSYMB, XSYMB, YSYMB)
C
C P u r p o s e :
C
C Plots a symbol of type ISYMB of size SSYMB cm centred around the
C point XSYMB/YSYMB on the plotting device.
C
C P a r a m e t e r s :
C
C ISYMB = Type of symbol to be plotted.           Integer, Input
C     1 = "plus",      2 = "X",      3 = "asterix"
C     4 = "square"    5 = "octagon"  6 = "diamond"
C     7 = "Z"         8 = "triangle" 9 = "triangle upside-down"
```

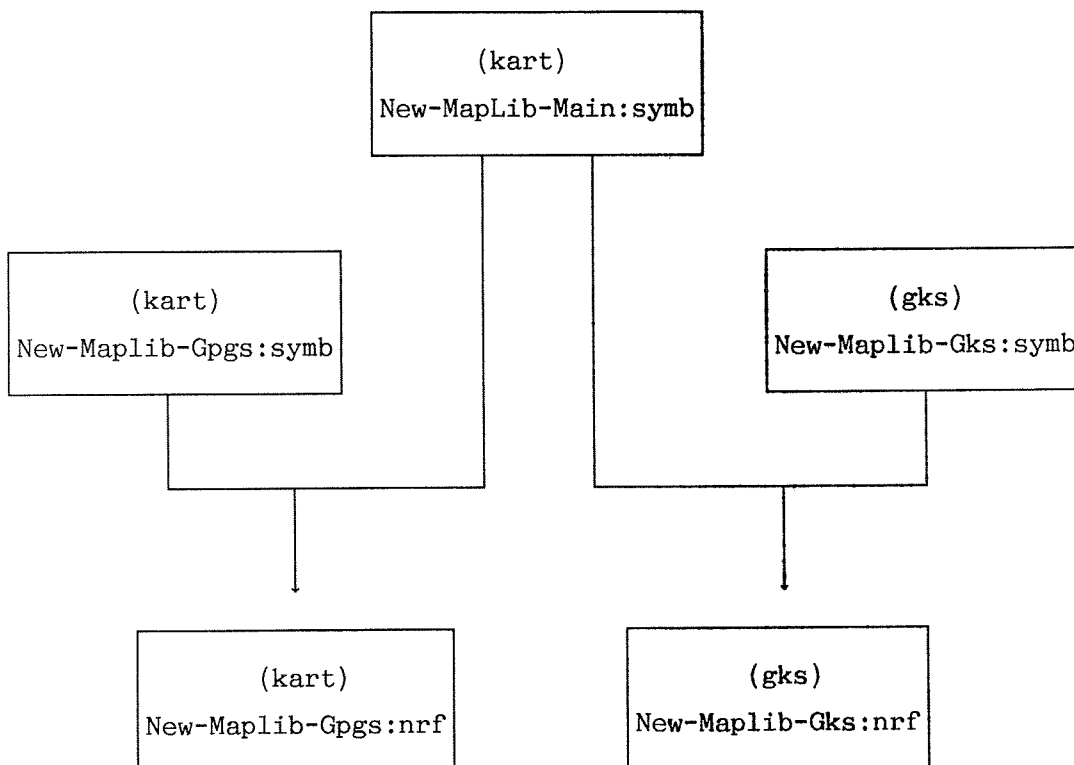
```

C      10 = "circle"    11 = "dot"
C SSYMB = Size of symbol in cm.                               Real,Input
C XSYMB = X-coordinate of center of symbol. World-coord.    Real,Input
C YSYMB = Y-coordinate of center of symbol. World-coord.    Real,Input
C
C R e f e r e n c e s / R e v i s i o n s :
C
C Trond Westgård, IMR, 12.83
C
C R e m a r k s :
C
C E r r o r C o n d i t i o n s :
C
C End Head

```

8.6.4 File-configuration on NORSK DATA 500.

To ensure compatibility between the two versions of Map-Library, the routines are stored in the following files.



8.6.5 Loading user programs.

To load a user program with MGKS and Map-Library on ND-500 the following entries must be present in a load sequence.

```
@nd-500 linkage-loader
  set-domain    <user-domain>
  load-segm     <user-program>
  .
  .
  load-segm     (gks)new-maplib
  load-segm     (gks)gks-util
  load-segm     (gks)mgks-std
  load-segm     (gks)mgks-lib
  end-domain
  exit
```

8.6.6 Compiling Map-Library.

Compiling the GKS-version of Map-Library requires the following job.

```
@fortran-500
  Library-Mode    ON
  Compile         (kart)New-Maplib-Main,,(gks)New-Maplib-Gks
  Compile         (gks)New-Maplib-Gks,.,.
  Exit
@Linkage-Loader
  Prepare         (gks)New-Maplib-Gks
  exit
```

whilst compiling the GPGS-version of Map-Library requires the following job.

```
@fortran-500
  Library-Mode    ON
  Compile         (kart)New-Maplib-Main,,(kart)New-Maplib-Gpgs
  Compile         (kart)New-Maplib-Gpgs,.,.
  Exit
@Linkage-Loader
  Prepare         (kart)New-Maplib-Gpgs
  exit
```

8.6.7 Transporting Map-Library to another system.

Some changes have to be made, when Map-Library shall be moved to another system.

- The syntax of the 'INCLUDE' statement throughout all the routines must be changed.
- All file-operations specific to one system are found in the subroutine FILUTL, which must be changed.

```

C
C ** FILUTL **
C
C       Subroutine FILUTL (IFUNC, BASNVN, USEREF, ISTAT)
C
C P u r p o s e :
C
C Performs the filehandling in Map-Library. All statements that might
C be altered when Map-Library is moved to another machine, should be
C in this routine.
C
C P a r a m e t e r s :
C
C IFUNC   : Input-function                Integer, Input
C          10 = Make the filenames of INFO, RUTE and KYST
C          21 = Open the file INFO
C          22 = Open the file RUTE
C          23 = Open the file KYST
C          31 = Close the file INFO
C          32 = Close the file RUTE
C          33 = Close the file KYST
C BASNVN  : Name of map-database          (only for IFUNC=10) Character, Input
C USEREF  : Username for map-database     (only for IFUNC=10) Character, Input
C ISTAT   : Returned status                Integer, Output
C
C R e f e r e n c e s / R e v i s i o n s :
C
C Andreas Christiansen, Cap Gemini, September 1988
C
C R e m a r k s :
C
C E r r o r C o n d i t i o n s :
C
C End Head
C

```

The GPGS-routine DATDEV does not exist in GKS, therefore the routine DATMED is introduced in the GKS-version of Map-Library.

If the GKS-routine GQDSP does not return size of medium in centimeters for a specific workstationtype, new entries should be included in the arrays IWSTRYP, WSSIZX and WSSIZY describing the workstationtype and size of medium in x- and y-direction in cm. Remember to increment the parameter NUMWS.

```

C
C ** DATMED **
C
C      Subroutine DATMED (iwrkid, xcm, ycm, xdc, ydc)
C
C P u r p o s e :
C
C Finds the size of the plotting medium in cm and in device coordinates.
C
C P a r a m e t e r s :
C
C IWRKID = Workstation identification           Integer  Input
C XCM    = Size of medium in cm coordinates. (x) Real    Output
C YCM    = Size of medium in cm coordinates. (y) Real    Output
C XDC    = Size of medium in device coordinates. (x) Real    Output
C YDC    = Size of medium in device coordinates. (y) Real    Output
C
C R e f e r e n c e s / R e v i s i o n s :
C
C Andreas Christiansen, Cap Gemini, Bergen, 09.88.
C
C R e m a r k s :
C
C E r r o r C o n d i t i o n s :
C
C End Head
C
C      Parameter (NumWs = 4)
C
C      Integer    iwstyp (numws)
C      Real       wssizx (numws)
C      Real       wssizy (numws)
C
C      Data       iwstyp / 1 , 4 , 401400 , 401404 /
C      Data       wssizx / 25.00 , 34.00 , 26.20062 , 26.20062 /
C      Data       wssizy / 18.00 , 26.00 , 19.00 , 19.00 /
    
```


8.6.8 Program-examples.

All the program-examples from the GPGS Users Guide have been rewritten into GKS. Two of them are included below. One which does not use GKS-UTIL (see page 128), and one which does.

```

C ** PROG-EXAMPLE-1 **
C
C Draws a chart frame with text.
C-----
C
C      Program PROEX1

C      CHARACTER TXT(2)*50
C      DIMENSION WF(4), VF(4)
C      DATA LATOR, LONOR /6200, 0000/
C      DATA LAT1, LON1, LAT2, LON2 /5500, -0500, 6500, 1100/
C      DATA SFRM, NTXTLN, STXT /0.2, 2, .6/
C      DATA LATIN, NLAT /0030,2/
C      DATA LONIN, NLON /0200,2/
C      DATA ILTPFRM /1/
C      DATA WF / 0., 0., 0., 0. /
C      DATA IMAP /3/, iwid / 1 /, iwtyp / 401400 /, IERR / 14 /

C
C      TXT(1)='Program example 1'
C      TXT(2)='A chart frame with text'

C
C      Call GRAFIN (3)
C      Call GRAFIN (2)

C      OPEN (ierr, file = 'gksexample')

C      Call GOPKS (ierr, idum)
C      Call GOPWK (iwid, 1, iwtyp)
C      Call GACWK (iwid)

C
C      Call CNVCH (WF, VF, LATOR, LONOR, LAT1, LON1, LAT2, LON2,
+              SFRM, SCALE, NTXTLN, STXT, IMAP)

C      Call GSELNT (1)

C      Call CHFRM (LATOR, LONOR, LAT1, LON1, LAT2, LON2,
+              LATIN, NLAT, LONIN, NLON,
+              SFRM, SCALE, ILTPFRM, IMAP)
C      Call TEXT (TXT, NTXTLN, STXT, SFRM)

C
C      Call GDAWK (iwid)
C      Call GCLWK (iwid)
C      Call GCLKS

C
C      CLOSE (ierr)
C      Call GRAFIN (1)
C      END

```

```
C ** PROG-EXAMPLE-2 **
C
C Draws a chart frame with text and coastlines.
C-----
C
C Includes common blocks describing topographical data :
C
$INCLUDE (kart)NEW-MAPLIB:INCL
C
CHARACTER TXT(2)*50
DIMENSION WF(4), VF(4)
DIMENSION LEVELS (5), LINTYP (5), LINCOL (5)
DATA LATOR, LONOR /7200, 2500/
DATA LAT1, LON1, LAT2, LON2 /7000, 1500, 7400, 3500/
DATA SFRM, NTXTLN, STXT /0.3, 2, .8/
DATA LATIN, NLAT /0030, 0/
DATA LONIN, NLON /0500, 0/
DATA ILTPFRM /1/
DATA NLEVELS, LEVELS / 5, 1, 3, 4, 5, 7 /
DATA LINTYP, LINCOL / 1, 1, 1, 1, 1, 1, 1, 1, 1, 1 /
DATA WF / 0., 0., 0., 0. /, iwid /1/, iwtyp / 401400 /
DATA SCALE / 0. /

C
C Initiates the wanted topographical database :
C
Call BASINI ( 'AKUP', 'P-T:KART', ISTAT)

C
TXT(1)='Program example 2.'
TXT(2)='A chartframe with text, coastlines and depthlines.'

C
IMAP = 2

C
Call GRAFIN (3)
Call GRAFIN (2)

Call GHISTA (iwid, iwtyp)

C
Call CNVCH (WF,VF,LATOR,LONOR,LAT1,LON1,LAT2,LON2,
+          SFRM,SCALE,NTXTLN,STXT, IMAP)

Call GSELNT (1)

Call GSTXFP (1, 0)
Call CHFRM (LATOR, LONOR, LAT1, LON1,
+          LAT2, LON2, LATIN, NLAT,
+          LONIN, NLON, SFRM, SCALE, ILTPFRM, IMAP)
Call COAST (LATOR,LONOR,LAT1,LON1,LAT2,LON2,SCALE,
+          IMAP, NLEVELS, LEVELS, LINTYP, LINCOL)
Call GSTXFP (501, 0)
Call TEXT (TXT,NTXTLN,STXT,SFRM)

C
Call GHIEND (iwid)
Call GRAFIN (1)
END
```

8.6.9 GKS utility library.

An additional subroutine library has been made to ease the initialization of GKS, and to perform some of the functions GPGS has, but not GKS (e.g. drawing of axes). The library is called GKS-UTIL and all the routines, except GRAFIN, have the prefix GHI.

The following routines are available at the moment.

8.6.9.1 GHISTA.

```

c=====c
c Subroutine for initiating the use of GKS in a FORTRAN c
c program. The routine opens error file, opens GKS, c
c opens and activates workstation. c
c c
c Parameters : Workid, workstation id. , Int., Inp. c
c Wktype, workstation type, Int., Inp. c
c c
c Written by : Andreas Christiansen, Cap Gemini c
c c
c Date written : 04 07 88 c
c c
c=====c
c
c Subroutine GHISTA (Workid, Wktype)
c -----

```

8.6.9.2 GHIEND.

```

c=====c
c Subroutine for cleaning up after the use of GKS in a c
c FORTRAN program. The routine deactivates and closes c
c workstation, closes GKS and closes errorfile. c
c c
c Parameters : Workid, workstation id. , Int., Inp. c
c c
c Written by : Andreas Christiansen, Cap Gemini c
c c
c Date written : 04 07 88 c
c c
c=====c
c
c Subroutine GHIEND (Workid)
c -----

```

8.6.9.3 GHIAKS.

SUBROUTINE GHIAKS (XMIN,XMAX,YMIN,YMAX,XTIC,YTIC)
c Draws simple axes under MGKS
c XMIN and XMAX are the beginning and end of the X-axis, in World Units.
c The axis is marked at intervals XTIC starting at XMIN,
c and the marks are labelled with their values.
c Similar for the Y-axis. The axes cross at XMIN,YMIN.
c
c Prerequicites:
c *****
c A normalization transformation must be established in world coordinates,
c and have transformation number 1. The window coordinates must be present
c in a common block called OMR. The window should be large enough
c to accomodate also labelling of the axes, i.e. there should some
c spare space to the left and below.
c The routine utilizes the routines GHILBL and GHINB.
c
c D.W.Skagen 7/9-88

8.6.9.4 GHILBL.

SUBROUTINE GHILBL(VERDI,X,Y,IB)
c Writes a number valued VERDI as graphical text in the position X, Y.
c VERDI must be a real number. IB is the number of digits behind the
c decimal point. If IB=0, the VERDI will be written in integerformat.
c Maximum number of digits is 10. X,Y are given in world coordinates.
c
c Prerequicites:
c *****
c Normalization transformation must be established, and have
c transformation number 1. The window coordinates must be present
c in a common block named OMR. Letter size (heigth) must be set,
c and must be given in standard coordinates (transformation 0).
c The subrutine uses the text attributes valid when it is called,
c in particular direction and alignment relative to the point X,Y.
c
c D.W.Skagen 7/9-88

8.6.9.5 GHINB.

FUNCTION GHINB(TALL)
c Determines the number of valid digits behind the decimal
c point in the number TALL (maximun is 8).
c
c D.W.Skagen 7/9-88

8.6.9.6 GRAFIN.

```
c
c ** GRAFIN **
c
c       Subroutine GrafIn (Ifunc)
c
c P u r p o s e :
c
c Subroutine for initiating a Tandberg terminal in alfanum/graphical modes.
c
c P a r a m e t e r s :
c
c       Ifunc :      1 = turn alfanum modus on           Integer, Input
c                   2 = turn graphical modus on
c                   3 = clear screen
c                   4 = clear graphical screen
c
c R e f e r e n c e s / R e v i s i o n s :
c
c Andreas Christiansen, Cap Gemini, September 1987
c
c R e m a r k s :
c
c E r r o r C o n d i t i o n s :
c
c End Head
```

8.6.10 GKS include files.

The three include files described in appendix B in the MGKS manual (Guide to the use of GKS at CERN) are stored in the following files on the NORSK DATA 500 computers, and they may be included in any user program.

```
(GKS)GKS-ENUM:INCL
(GKS)GKS-DEFTYP:INCL
(GKS)GKS-FONAMS:INCL
```

8.6.11 GKS-references.

Enderle G., Kansy K., Pfaff G. 1987. Computer Graphics Programming.
GKS - The Graphics Standard. Second and Enlarged Edition.
Springer Verlag, Berlin.

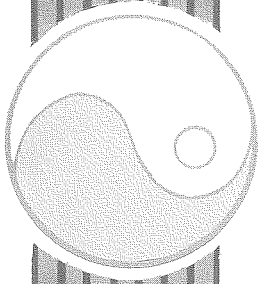
Anonymus 1986. Guide to the use of GKS at Cern.
Draft Manual for the use of GKS at CERN. (MGKS)
Version 1.5
DD Graphics Section.

I N D E X L I S T

Reference	Page
/BHLP/	29, 49, 73, 90
/KART1/	63
/KART2/	63
/KART3/	63
/KART4/	63
/KART5/	49, 63
/MAPER1/	45, 49, 95
/MAPER2/	95
ARC	55, 61, 79
AREA	3, 17, 21, 24, 30-32, 49, 54, 55, 58, 62, 72, 92, 98, 109, 113
ARROW	40, 42, 55, 62
BASINI	9, 11, 15, 20, 22, 25, 27, 33, 42, 44, 49-53, 55, 63, 114, 115, 127
BLKC2A	55, 64
BLKCH1	16, 17, 19, 26, 50, 52, 55, 59, 65, 67, 89, 104
BLKCH2	17, 21, 22, 55, 66
BLKCH3	17, 55, 65, 67
BLKCS	37, 40, 55, 68
BOTTOM	5, 35-38, 40, 55, 68, 71, 74, 89
CHANGE	56, 69
CHFRM	7, 9-11, 15, 20, 22, 26, 28, 34, 42, 44, 56, 65, 69, 81, 116, 126, 127
CNVC1A	56, 70
CNVC2A	56, 71
CNVCH	6, 7, 9-12, 15, 17, 18, 20, 22, 23, 26, 28, 32, 34, 35, 42, 44, 46, 52, 54, 56, 59, 72, 90, 102, 104, 105, 116, 126, 127
CNVCS1	35, 40, 54, 56, 73, 102, 104
CNVCS2	35, 40, 54, 56,

Reference	Page
CNVDTI	74
CNVDTI	12, 15, 21-23,
CNVDTI	26, 42, 56, 75
CNVDTI	53, 56, 76
CNVDTI	10, 12, 15, 21,
CNVDTI	22, 26, 28, 34,
CNVDTI	42, 44, 56, 76,
CNVDTI	113, 116, 127
CNVDTI	56, 77, 80
CNVDTI	18, 21, 23, 35,
CNVDTI	40, 56, 78
CNVDTI	54, 56, 78
CNVDTI	56, 79
CNVDTI	37, 40, 53, 57,
CNVDTI	80
CNVDTI	57, 59, 81
CNVDTI	32, 48, 54, 57,
CNVDTI	82
CNVDTI	19, 57, 83
CNVDTI	23, 26, 57, 83,
CNVDTI	90
CNVDTI	12, 15, 23, 40,
CNVDTI	57, 75, 76, 84
CNVDTI	29, 48, 57, 85,
CNVDTI	86
CNVDTI	57, 86
CNVDTI	119-121, 128,
CNVDTI	130, 131
CNVDTI	57, 87, 126,
CNVDTI	127, 130
CNVDTI	32, 34, 57, 87
CNVDTI	57, 88
CNVDTI	57, 88
CNVDTI	57, 89
CNVDTI	57, 89
CNVDTI	58, 90
CNVDTI	48, 58, 90
CNVDTI	51, 52, 58, 91
CNVDTI	17, 18, 32, 55,
CNVDTI	58, 76, 91
CNVDTI	23, 24, 26, 58,
CNVDTI	90, 92
CNVDTI	3, 38, 54, 58,
CNVDTI	68, 93
CNVDTI	58, 94, 100
CNVDTI	45, 46, 53, 54,
CNVDTI	94, 95
CNVDTI	45, 46, 49, 95
CNVDTI	45, 46, 95
CNVDTI	96
CNVDTI	54, 58, 96

Reference	Page
PLANE	58, 93, 97
PLTMRK	27, 29, 34, 58, 97, 121
POLAR	3, 48, 58, 88, 98, 104, 106
PRJCTA	58, 98
PRJCTN	30, 34, 58, 59, 99
RDGR	34, 48, 59, 100
REDDIG	59, 101
SMOOTH	18, 21, 23, 26, 48, 55, 59, 78, 79, 101
STRVAL	59, 102
TEXT	7, 9-12, 14, 15, 20, 22, 26-28, 34, 35, 40, 42, 44, 45, 50, 52, 53, 59, 63, 102, 116, 126, 127, 129
UPDTP	59, 103
UTG	59, 103
VALID	59, 104, 129
XYCONI	29, 59, 104
XYCONV	27-29, 34, 59, 105
XYPSI	26, 28, 29, 34, 48, 59, 105, 106
XYZI	48, 59, 106, 107
Zcont	112, 113
ZGRI	16, 21, 23, 26, 40, 59, 107
ZGRID	15, 16, 54, 60, 107, 108
ZONE	31, 34, 60, 88, 109
ZONHLP	60, 109
ZZHALF	60, 110



FORSKERKART.

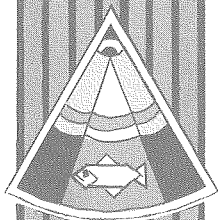
EDB-presentasjon av marine data

Brukerveiledning del 6 av 6 :

I T A K S, versjon 1.30 august 1988

Tor Knutsen

Trond Westgård



Fiskeridirektoratets
Havforskningsinstitutt

Institute of Marine Research
Directorate of Fisheries

I N N H O L D S F O R T E G N E L S E

<u>Seksjon</u>	<u>Side</u>
1 Innledning	3
1.1 Produktet	3
1.2 Hva du kan gjøre med ITAKS	4
1.3 Forhåndskunnskaper	6
2 Litt om dialogen mellom brukeren og ITAKS	7
2.1 Ulovlig verdi i datafelt	8
2.2 Navigering i Focus-sider	8
2.3 Meldinger til bruker	10
3 Hovedmenyen	11
3.1 Lage nye kart eller bilder	12
3.1.1 Bestem plote-medium, målestokk og plassering	12
3.1.2 Meny - nytt kart	16
3.1.2.1 Kartramme med kystlinjer	18
3.1.2.2 Stasjonsnett	19
3.1.2.3 Isolinjer	20
3.1.2.4 Integrering	20
3.1.2.5 Strøm- og vindvektorer	21
3.1.2.6 Horisontalt snitt	24
3.1.2.7 Vertikalt snitt	24
3.1.2.8 Tekst, symboler og linjer	29
3.1.2.9 Se bilde på skjermen	32
3.1.2.10 Plott bilde på nytt medium	32
3.1.2.11 Oversikt over bilder i bibliotek	32

Seksjon	Side
3.1.2.12 Hent bilde fra bibliotek	32
3.1.2.13 Lagre bilde på bibliotek	33
3.2 Valg av brukerprofil	34
3.3 Oversikt over lagrede bilder	36
3.4 Plott bilde fra bibliotek	36
3.5 Slett bilde i bibliotek	37
3.6 Pakke bibliotek	38
3.7 Sintran-kommandoer	38
3.8 Valg av kartdatabase	39
4 Datafiler og filgrensesnitt	40
5 Feilmeldinger ved lesing av datafil	45
6 Linjetyper	46
7 Farger	46
8 Skjermer og andre plottemedier på Havforskningsinstituttet .	48
9 Eksempler på bruk av ITAKS	51

Seksjon	Side
10 Tabell over tilgjengelige skrifttyper	55
11 Tabell over tilgjengelige fonttyper	58
12 REFERANSER	59
Stikkord	61

1 Innledning

En av målsetningene med Havforskningsinstituttets Egg og LarveProgram (HELP) er å utvikle og videreutvikle tjenlige EDB-verktøy for bl.a. framstilling av geografisk fordelte data. Dataene kan enten være samlet inn på havet (f.eks. fordeling av egg og larver) eller være modellresultater fra f.eks. driftmodeller for egg og larver benyttet ved konsekvensvurderinger. Ved Havforskningsinstituttet er det for dette formålet utviklet et fleksibelt og brukervennlig EDB-system kalt ITAKS (Interaktiv Tegning av Kart og Snitt) (Engelsen og Westgård, 1986; Knutsen og Westgård, 1988).

1.1 Produktet

Denne rapporten beskriver versjon 1.3 av ITAKS. Programmet er en overbygning til Map-Library (Westgård, 1984; Westgård, Knutsen og Christiansen, 1988). ITAKS gjør en bruker uten EDB-kunnskaper istand til å tegne kart med stasjonsnett, isolinjer o.l. på en svært enkel og rask måte. Brukeren kommuniserer mot et sett av skjermbilder, og flytter seg mellom og i bildene v.h.a. piltastene på tastaturet. Dialogen mellom program og bruker er laget med Focus skjermhåndterings-system (Anon., 1985). Bruken av Focus, gjør at selv om programmet er skrevet i Fortran-77 (Anon., 1986a), kan det bare brukes på maskiner fra Norsk Data A/S. Foreløpig er alle muligheter i programmet bare tilgjengelig på Tandberg skjerm med grafisk opsjon. En ny versjon er under arbeid der Tektronix 4107 fargeskjerm eller en personlig datamaskin (PD) med emulator for denne skjermen kan brukes.

ITAKS ble utviklet etter en ide av Trond Westgård, Havforskningsinstituttet (HI). Han har også vært prosjektansvarlig under utviklingen. Fire studenter fra Bergen Ingeniørhøgskole hadde våren 1986 sine prosjektoppgaver ved HI, Harald Engelsen og Sverre Vestbøstad arbeidet med Focus delen av ITAKS mens Stein Holger Pettersen og Veysel Soydan arbeidet med GPGS-F delen. GPGS-F er en forkortelse for General Purpose Graphics System in Fortran (Anon., 1984b). Harald Engelsen var sommeren 1986 engasjert ved HI for å kople de to delene sammen. Våren 1987 begynte Tor Knutsen ved HI og han har forbedret og utvidet den

første utgaven av programmet.

Forslag til forbedringer eller utvidelser av ITAKS mottas med takk. Produktet er svært nyttig for deg som arbeider med fremstilling av data samlet inn på havet eller for deg som ønsker å presentere resultater fra modeller.

1.2 Hva du kan gjøre med ITAKS

- Lage kartramme med kystlinjer og bunnkonturer i flere projeksjoner (Mercator, XY-projeksjon, UTM, Polarstereografisk, Polarstereografisk med pol i senter og kjegleprojeksjon).
- Lage vertikale snitt.
 - a) Plotte direkte med basis i egen datafil uten å gå veien om lage en kartramme.
 - b) Plotte kartramme og vise en horisontal utbredelse av stasjonene i en datafil. Deretter kan en plukke ut stasjoner som ligger nærmere enn X km fra en storsirkel som går gjennom et startpunkt A og sluttpunkt B. A og B er da de geografiske posisjoner for start og slutt i det vertikale snittet. X, A og B spesifiseres av brukeren.
- På grunnlag av valgt kartramme kan en tegne stasjonsnett, isolinjer, arealintegrere eller tegne en horisontal presentasjon av stasjoner som er valgt ut til å inngå i et snitt.
- Mulighet for å variere type og størrelse av symboler, linjetyper, merkelapper for stasjoner i et stasjonsnett. Plassering av symboler kan også varieres relativt til geografisk posisjon.
- Trajektorieplott av f.eks. drivbøyer med tidsmarkering.

- Strøm og vindvektorplott hvor farge og bredde på vektorer kan varieres.
- Legge inn tekst, symboler eller linjer på frihånd.
- Velge målestokk på kartet. Forutsetningen er at den oppgitte målestokk er større enn det ITAKS selv beregner mot plottemediet.
- Lagre bilder eller deler av det (segmenter) i et bibliotek for senere bruk.
- Plotte (kart)bilder med valgt plassering/størrelse i cm ut på ønsket medium (f.eks. Calcomp plotter)
- Slette enkelte gjøremål (segment), de interaktivt lagde segmenter og segmenter hentet fra bibliotek eller hele bildet (alle aktive segmenter).
- Velge/skifte brukerprofil mot ITAKS.
- Fleksibel bruk av biblioteker.
- Variable skrifttyper.
- Variable linjetyper i seksjonsramme, kartramme, kystlinjer, isolinjer, grid ved integrering.
- Variable linjetyper i storsirkler med tilhørende linjer ved horisontal presentasjon av data.
- Farger kan spesifiseres i seksjonsramme, kartramme, kystlinjer, isolinjer, grid ved arealintegrering.
- Farger kan spesifiseres i storsirkler med tilhørende linjer ved horisontal presentasjon av data.

Se Kapittel 12 for eksempler på bruk av ITAKS.

Hva ITAKS ikke kan gjøre :

- Flere kan ikke benytte ITAKS samtidig fra det samme brukerområdet.
- Forandre deler av en kartramme, isolinjer eller et stasjonsnett etter at det er tegnet.
- Forandre et tidligere laget "overlay" av tekst, symboler og linjer.
- Hente kartramme fra bibliotek og så utføre f.eks. isolinjer i området. Du må definere og utføre en kartramme først.
- Rotere/transformere kartbildene (segmentene) på skjerm eller plotter.

1.3 Forhåndskunnskaper

Det kreves liten kjennskap til EDB, men det er en fordel å kjenne til funksjonene i Map-Library (Westgård, 1984; Westgård et al., 1988). Map-Library er et sett med rutiner som en kan bruke fra et FORTRAN-program for å tegne kartrammer, kystlinjer, isolinjer osv.. Hvis du er i tvil om hva datafeltene inneholder er det lurt å slå opp i Map-Library (Westgård, Knutsen og Christiansen, 1988) for mer informasjon. De fleste variabelnavn til venstre i ledeteksten på hver FOCUS-side i ITAKS finner du igjen i Map-Library. For å hente data fra filer er det en fordel med kjennskap til SINTRAN, operativsystemet på ND-500 anlegget, og hvordan en i FORTRAN formattsatser kan definere i hvilke kolonner variablene ligger. For å sette opp datafiler som skal leses av ITAKS eller inspisere slike bør du kunne bruke en av teksteditorene på ND-500. Det er lettere å bruke systemet hvis du kjenner funksjonene på Notis-tatstaturet som f.eks. "ekspander", "angre" og "stryk". En viss kjennskap til begrepene billedsegment, vindu og synsport som brukes i

grafisk databehandling skader ikke.

Det kan også være nyttig å kjenne begrepene segment og buffer i GPGS-F (Anon., 1984a).

Definisjon av buffer og segment:

Et buffer er et kontinuerlig område i datamaskinens hukommelse eller masselager.

Et segment er en samling av grafiske og ikke-grafiske elementer lagret i et kontinuerlig bufferområde og identifisert av et spesifikt heltall mellom 0 - 4096. Grafiske elementer må være innenfor et segment.

Altså : Et segment er en samling av grafiske elementer (f.eks. linjer) som logisk hører sammen og danner en grafisk tegning (f.eks. et symbol, eller et stasjonsnett). Hver tegning eller del av en tegning får av ITAKS en entydig identifikator.

2 Litt om dialogen mellom brukeren og ITAKS

All dialog mellom brukeren og ITAKS skjer gjennom skjermbilder som er laget med skjermhåndteringsprogrammet Focus. En Focus-side består av ledetekst og datafelt. Du må alltid sørge for at datafeltene har en fornuftig verdi før du forlater siden. Spesielt gjelder dette definering av gjøremål. Et gjøremål kan f.eks. være : "Lag en kartramme med kystlinjer". For å definere et gjøremål fullstendig må ITAKS bruke data fra flere skjermbilder. Et gjøremål er i denne rapporten alltid angitt i hermetegn.

Første gang du bruker ITAKS vil alle sidene ha standard verdier. Du kan derfor bruke UTFØR -tasten direkte og få fram et plott uten å definere sidene først. Når du forlater ITAKS vil dine spesifikasjoner bli lagret på filen ITAKS:INIT. Neste gang du kommer inn i ITAKS vil denne filen leses og datafeltene i skjermbildene vil få samme verdi

som de du spesifiserte forrige gang du brukte ITAKS. Vi kaller derfor ITAKS:INIT filen for din brukerprofil mot ITAKS.

2.1 Ulovlig verdi i datafelt

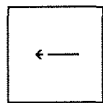
Hvis du oppgir ulovlige verdier i et felt gies en melding og du kan ikke forlate feltet før det har fått en lovlig verdi. Verdiområde kan endres, men dette må gjøres i FOCUS-DEF (Anon., 1985). Kontakt i dette tilfelle den som er programansvarlig for ITAKS.

2.2 Navigering i Focus-sider

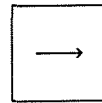
Når du har kommet inn i en Focus-side kan du bevege deg i datafeltene, hoppe til nye datafelt og til nye Focus-sider. Nedenfor finner du en kort beskrivelse av hvilke funksjonstaster du da må bruke.

Bevegelse i et datafelt

For å bevege seg i feltet nyttes venstrepil eller høyrepil.



Venstrepil

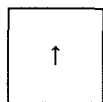


Høyrepil

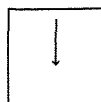
Hvis venstrepil eller høyrepil brukes i h.h.v. første eller siste posisjon i et datafelt foretaes et feltskifte dersom det eksisterer et nabofelt.

Fra felt til felt

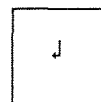
For å bevege seg mellom feltene nyttes opp-pil, ned-pil eller retur tast.



opp-pil



ned-pil

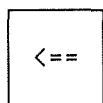
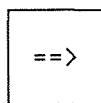


retur

Med opp-pil flytter du til feltet ovenfor, brukes ned-pil hopper du til feltet nedenfor mens bruk av retur gir hopp til logisk neste felt som f.eks. kan være feltet ved siden av.

Fra side til side

Når du vil ut av en Focus-side bruker du en av tastene som er vist på sidens nederste linje.

Venstre
side-pilHøyre
side-pil

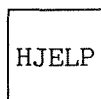
Hjem

Venstre side-pil = Utfør eller forrige side

Høyre side-pil = Definer eller valg eller neste side

Hjem = Tilbake eller avslutt gjøremål

Focus funksjonstaster



De Focus-sidene som har teksten <HJELP> INFORMASJON på nederste eller øverste linje har hjelpesider. Disse fås fram ved å trykke HJELP i ønsket datafelt. For å komme tilbake til datafeltet/siden trykkes en vilkårlig tast.

Slike hjelpesider kan legges inn på alle Focus-sider og for hvert datafelt. Dette må gjøres i FOCUS-DEF (Anon., 1985).

Kontakt i dette tilfelle programansvarlig for ITAKS.

STRYK

Trykkes STRYK når du er i et datafelt vil det som står der forsvinne. Du kan få det fram igjen ved å trykke :

ANGRE

Innenfor hvert datafelt kan du bruke ekspander funksjonen på samme måte som i NOTIS ved å trykke :

INNS
EKSP

2.3 Meldinger til bruker

Under kjøring av ITAKS vil det dukke opp en del meldinger på nederste linje av skjermen. Disse krever en kvittering i form av et tastetrykk. De fleste meldinger gir brukeren et valg mellom flere alternativ. Hvis brukeren får en slik melding kan han i enkelte tilfelle returnere til meny ved å trykke HJEM, dette gjelder spesielt når feil er oppstått. En melding starter alltid med '>'.

Eksempel på melding :

> Ikke registrert bilde, fortsette (J/N/HJEM) ?

Her er det tre muligheter : 1. 'J' = Fortsetter kjøring.
2. 'N' = Leser nytt filnavn.
3. Hjemtast = Returnerer til meny.

3 Hovedmenyen

For å starte opp ITAKS logger du deg inn på ND-500 anlegget på vanlig måte og gir kommandoen (brukerens svar er understreket) :

@ND (KART)ITAKS↓

Du vil da få følgende bilde på skjermen :

MAP/ITAKS	NIVÅ - 0

Versjon 1.3	H O V E D M E N Y

. ==>	Lage nye kart eller bilder
. ==>	Valg av brukerprofil
. ==>	Oversikt over bilder i bibliotek
. ==>	Plott bilde fra bibliotek
. ==>	Slett bilde i bibliotek
. ==>	Pakk bibliotek
. ==>	Sintran-kommandoer
. ==>	Valg av kartdatabase

↖	AVSLUTT ITAKS ==> VALG

For å gå opp og ned i menyen brukes opp-pil og ned-pil. Hjem-tasten avslutter ITAKS mens du med høyre side-pil velger gjøremål.

Kort om hvert valg :

Lage nye kart eller . . . : Bygger opp kartbilder etter eget bilde ønske. Se Kap. 3.1.

Valg av brukerprofil. . . : Du kan velge deg en annen brukerprofil enn ITAKS:INIT som er standard.
Se Kap. 3.2.

MAP/ITAKS	NIVÅ - 1

BESTEM PLOTTE-MEDIUM	

På Havforskningsinstituttet har vi følgende media :	
1. Textronix 4012 skjerm	6. Philips plotter
2. Calcomp plotter	7. Watanabe plotter
3. HP 7221 plotter	8. HP 7550A OSEAN
4. HP 7475 plotter	9. Tektronix 4105 Fargeskjerm
5. Tandberg skjerm	10. Tektronix 4107 Fargeskjerm
6. Phillips plotter	11. HP 7550A BLOKJEMOS
	12. Tektronix 4693D/4510 fargeplotter
IDEV	Oppgi ønsket plotte-medium (GPGS-devicenr kan gies direkte. Sett minustegn (-) foran tallet) >
SCALE	Målestokk på kart (def.0., automatisk skalering) > 0000000.
MWF(4)	Bestem plassering på medium i cm >/...../...../...../ X-min/X-max/Y-min/Y-max

<HJELP> INFORMASJON	==> NESTE SIDE

Dersom ditt valgte medium ikke er gitt i tabellen på siden over kan du gi inn det nummeret som GPGS benytter mot plottemediet. Dette innebærer at du må ha nøye kjennskap til GPGS. Hvis ikke, ta kontakt med den som er ansvarlig for installasjon av GPGS. Skal du gi GPGS-nr direkte gir du dette med et minustegn foran tallet.

Målestokk på kartet spesifiserer du kun i de tilfellene du ønsker å reproducere et sjøkart eller deler av dette i original størrelse. En annen mulighet er hvis du har to sjøkart og ønsker å sette sammen deler av disse til ett kart (Fig. 1).

Ved valg av målestokk er det den inverse verdien som oppgives. Hvis målestokken på et kart du ønsker å reproducere er 1 : 700000, så gir du variabelen SCALE verdien 700000. Verdien du gir må være større enn den verdi ITAKS beregner for målestokken, ut fra geografiske posisjoner og valgt størrelse på kartet i cm. Det er svært viktig at du også oppgir korrekt konstruksjonsbredde til Mercatorkartet du ønsker å reproducere. Konstruksjonsbredde er den breddegrad hvor målestokken er korrekt. Flytt i dette tilfelle til siste side som vises under "Karttramme med kystlinjer" (se Kap. 3.1.2.1). Hvis ikke vil konstruksjonsbredden bli beregnet til ca. midt i kartet ditt og det kan bli store

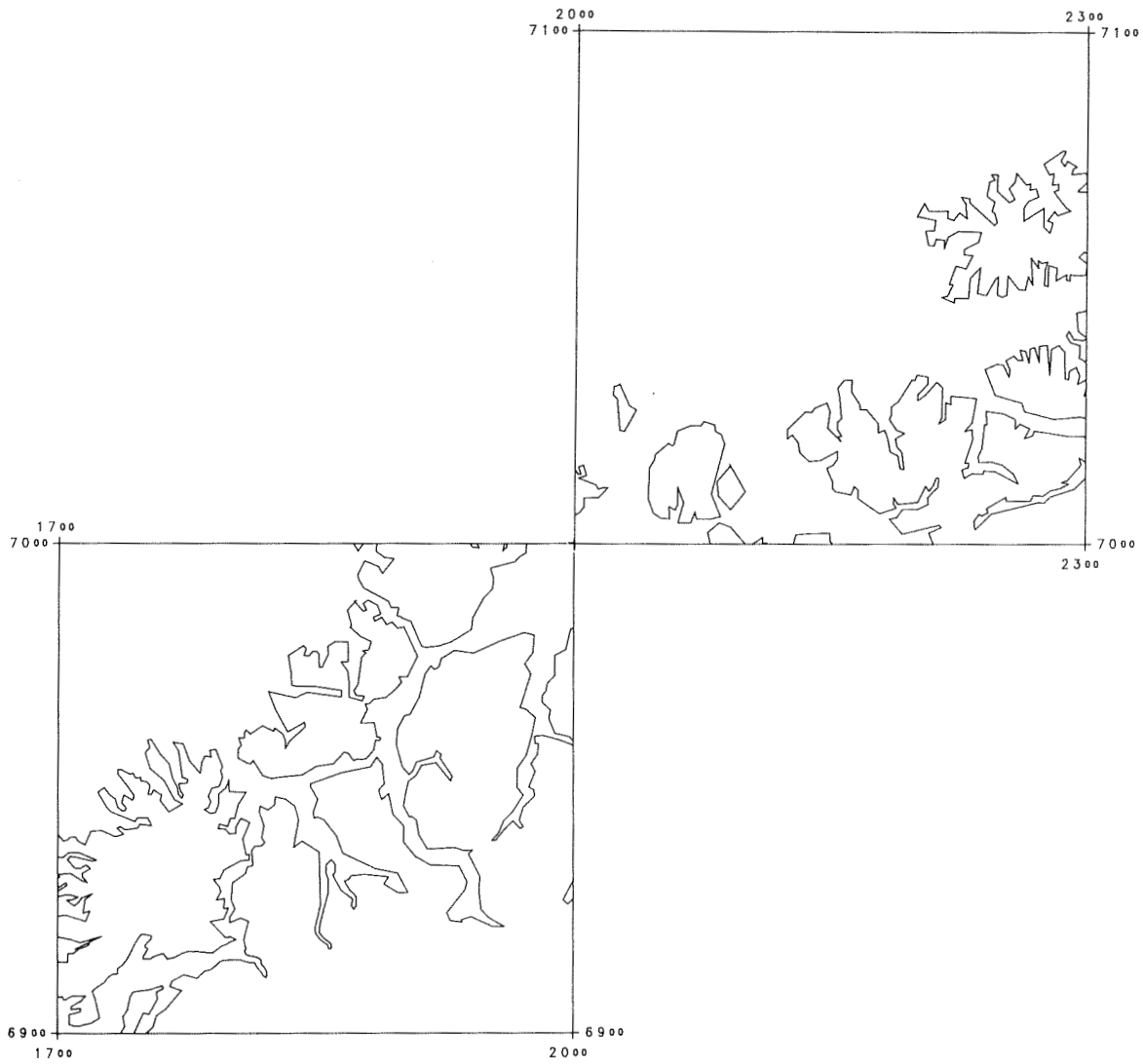


Fig. 1. Sammenkjedning av delkart fra originale sjøkart

avvik fra originalkartet. Dersom du vil ha målestokken skrevet ut i teksten under kartrammen er du fri til å velge dette i en av sidene under "Kartramme med kystlinjer".

Når du tegner på skjerm er det lurt å bruke hele skjermen. På Calcomp plotteren som har A0 format vil en nesten alltid ønske å tegne bare på deler av arealet. Husk at plasseringen også bestemmer størrelsen på plottet. Blir det oppgitt for store verdier vil systemet gi feilmelding. Maksimumverdier for utstrekning av x-akse og y-akse er gitt i Tabell 1 for noen medier. Hvis du ønsker tilsvarende verdier for andre medium må du se i GPGS-F manualen.

Dersom $x_{min}, x_{max}, y_{min}, y_{max}$ alle er null, vil ITAKS velge de størst mulige verdiene for hvert enkelt medium. Dersom du først velger å plotte på skjermen vil et identisk bilde bli lagt ut på et pseudo segment eller buffer (se definisjon foran). Velger du etterpå å plotte på nytt medium f.eks. HP-7550A, vil plottet hentes fra buffer og sendes til det valgte mediet. Vær oppmerksom på at størrelsen på dette plottet vil være noe mindre enn for et plott som sendes direkte ut på det samme mediet uten at det først er plottet på skjermen.

Tabell 1. Minimum og maksimumsverdier (cm) for x- og y-akse på ulike plottemedier.

	x-min	x-max	y-min	y-max
Tandberg skjerm	0.0	26.0	0.0	19.0
Calcomp	0.0	118.0	0.0	84.0
HP-7550A (A3)	0.0	39.9	0.0	27.0
HP-7550A (A4)	0.0	26.0	0.0	19.0

3.1.2 Meny - nytt kart

Når du har valgt plottemedium kommer dette bildet fram på skjermen :

MAP/ITAKS		NIVÅ - 2	

(STRYK)		M E N Y - N Y T T K A R T	

			UTFØRT
<== . ==>	Kartramme med kystlinjer	>	NEI
<== . ==>	Stasjonsnett	>	NEI
<== . ==>	Isolinjer	>	NEI
<== . ==>	Integrering	>	NEI
<== . ==>	Strøm / vind vektorer	>	NEI
<== . ==>	Horisontalt snitt	>	NEI
<== . ==>	Vertikalt snitt	>	NEI
<== .	Tekst, symboler og linjer	>	NEI
<== .	Se bilde på skjermen		
<== .	Plott bilde på nytt medium		
<== .	Oversikt bilder i bibliotek		
<== .	Hent bilde fra bibliotek		
<== .	Lagre bilde på bibliotek		

<== UTFØR	↖	TILBAKE TIL HOVEDMENY	==> DEFINER

For å gå opp og ned i menyen brukes opp-pil og ned-pil. Hjem-tasten resulterer i at en går tilbake til hovedmenyen, mens du med høyre side-pil velger gjøremål du vil spesifisere eller kontrollere parametrene til. Venstre side-pil resulterer i at det valgte gjøremål utføres direkte. Du bør imidlertid først gå inn på de aktuelle sidene ved hjelp av DEFINER-tasten (==>) og sjekke at de på forhånd definerte verdier er fornuftige, iallefall første gang du bruker ITAKS. Variablene i kolonnen merket UTFØRT i bildet over har to verdier : JA eller NEI. Disse viser om funksjonen har et aktivt billedsegment eller ikke.

Beskrivelse av valgene i menyen :

Kartramme med kystlinjer . . : Definerer eller tegner en kartramme i forskjellige projeksjoner samt kystlinjer, bunnkonturer og tekst under kartrammen (1 segment).

- Stasjonsnett : Definerer eller tegner et stasjonsnett (kurslinjer) (1 segment).
- Isolinjer : Definerer eller tegner isolinjer (1 segment).
- Integrering : Definerer eller tegner og beregner mengde av en målt variabel i et område.
- Strøm / vind vektorer . . . : Definerer eller tegner strøm-, vindvektorer eller andre variable som kan presenteres på denne måten (1 segment).
- Horisontalt snitt : Definerer eller tegner en horisontal presentasjon av stasjoner som skal inngå i et snitt (1 segment).
- Vertikalt snitt : Definerer eller tegner et vertikalt snitt også kalt seksjon (1 segment).
- Tekst, symboler og linjer. : Legger interaktivt på tekst, symboler og linjer og editerer inntil man er fornøyd (flere segment).
- Se bilde på skjermen : Viser det grafiske bildet som er lagret lokalt på skjermen.
- Plott bilde på nytt medium . : Tegne bildet (alle segment som er i buffer) på nytt medium.
- Oversikt over lagrede bilder : Liste navn på alle (kart)bilder du har lagret i biblioteket.
- Hent bilde fra bibliotek . . : Legge (kart)bilder fra biblioteket inn i buffer (ferdige segment).
- Lagre bilde på bibliotek. : Lagre (kart)bilde (d.v.s. alle segment

i buffer) på bibliotek.

STRYK

- tasten : Dersom du trykker denne tasten i feltene for "Kartramme med kystlinjer", "Stasjonsnett", "Isolinjer", "Integrering", "Strøm / vindvektorer", "Horisontalt snitt" og "Vertikalt snitt" vil du kunne slette tilsvarende aktive segmenter.

Trykker du stryk-tasten i feltet for "Tekst, symboler og linjer" vil du kunne slette alle bildesegmenter laget interaktivt samt alle segmenter som er hentet inn fra biblioteket.

3.1.2.1 Kartramme med kystlinjer

For å lage en kartramme med kystlinjer må du spesifisere et geografisk område, hvorledes kartrammen skal se ut ("layout") og kartprojeksjon. Dette gjøres ved å trykke høyre side-pil. Du vil da komme inn i en side med datafelt hvor du kan avgrense aktuelt geografisk område og gi ønsket kartprojeksjon. De mulige kartprojeksjoner er XY-koordinater, UTM-koordinater, Mercator projeksjon, kjegle projeksjon, polarstereografisk projeksjon og polarstereografisk projeksjon med pol i senter (se Anon., 1986b; Berge, 1987; Westgård et al., 1988)

Du kan ikke tegne en ny kartramme oppå en allerede eksisterende kartramme. Du får melding om dette i programmet.

"Kartramme med kystlinjer" har i alt 4 Focus-sider. Det er ikke nødvendig å definere alle variable på disse sidene før du tegner et kart. De variablene du ikke setter verdi på vil få en verdi fra brukerprofilen. Når kartrammen er tegnet kommer meldingen

> Plotting ferdig ! Trykk en tast.

Ved å trykke en vilkårlig tast kommer du tilbake til MENY-NYTT KART.

3.1.2.2 Stasjonsnett

For å lage et stasjonsnett (kurslinjer) må du ha en datafil med geografiske posisjoner og måledata. Hvordan disse datafilene må se ut er beskrevet Kapittel 4. Du kommer inn i sidene ved å trykke høyre sidepil i MENY-NYTT KART. "Stasjonsnett" har 2 sider. På første side (se Kap. 4) finnes variabelen ATTR. Ved hjelp av denne kan du velge om du vil lese inn attributter fra den samme datafilen som du har dine geografiske posisjoner og måledata. Attributter kan defineres som egenskaper (f.eks. form, størrelse og farge) til punkt, linjer og datapunkt merkelapper. Svarer du J på spørsmål om attributter vil neste side ikke bli vist på skjermen. I dette tilfelle må du i tillegg til geografiske posisjoner og måleverdi ha lagt inn seks kolonner med tall. Dette er verdir til de seks attributtene LABPT (antall desimaler etter komma i datapunkt-merkelapper), ISYMB (symboltype), SSYMB (størrelse av symbol i cm), SLABL (størrelse av karakterer i datapunkt-merkelapper i cm), LTYPE (linjetype mellom datapunkt) og IREPOS (posisjonering av symbol relativt til datapunkt. Med datapunkt menes her en geografisk posisjon.

Vil du at alle datapunktene skal ha samme type symboler, være sentrert i forhold til geografisk posisjon og linjene mellom dem ha samme linjetype, svarer du N(ei) på ATTR og setter variabelen IREPOS på neste FOCUS-side i "Stasjonsnett" til -1. Dersom du f.eks. ønsker enkle Argos drivbaneplokk kan du få tegnet heltrukne linjer fra senter av et datapunkt til senter av påfølgende datapunkt, ved å bruke negative verdier for linjetype. Når stasjonsnettet tegnes får du melding om hvor mange punkt som ligger innenfor din spesifiserte kartramme og kan velge om du vil fortsette eller ikke.

I ITAKS er det en sperre slik at du ikke kan utføre "Stasjonsnett" sammen med "Integrering", "Strøm- og vindvektorer", "Horisontalt snitt" eller "Vertikalt snitt". Plottet ville da bli lite oversiktlig. Du får melding om dette i programmet.

Når stasjonsnettet er plottet får du følgende melding opp på skjermen

> Plotting ferdig ! Trykk en tast.

Trykk en vilkårlig tast og du kommer tilbake til menyen.

3.1.2.3 Isolinjer

For å lage isolinjer må du ha en datafil med geografiske posisjoner og måledata (se Kap. 4). Datafilen kan være identisk med datafilen for "Stasjonsnett" og "Integrering". Det geografiske område der isolinjer skal tegnes kan avgrenses med kystlinjer eller med et polygon definert ved hjelp av geografiske posisjoner innlest fra en egen fil. "Isolinjer" har fem FOCUS-sider hvor en kan gi verdier til forskjellige variable som har betydning for plottets utseende.

Når du utfører "Isolinjer" får du melding om hvor mange punkt som ligger innenfor din spesifiserte kartramme og du kan velge om du vil fortsette eller ikke.

Også her er det en sperre i ITAKS slik at du ikke kan utføre "Isolinjer" sammen med "Integrering", "Strøm- og vindvektorer", "Horisontalt snitt" og "Vertikalt snitt". Plottet ville da bli svært uoversiktlig. Du får melding om dette i programmet.

Når isolinjene er plottet får du melding om at plotting er ferdig (jfr. Kap. 3.1.2.1), og du trykker en vilkårlig tast for å komme tilbake til menyen.

3.1.2.4 Integrering

"Integrering" med ITAKS betyr å bruke rutinene i Map-Library der data i et geografisk område blir interpolert i et rutenett i et Mercator kart eller en annen projeksjon. For hver rute i nettet blir det beregnet et areal i kvadratmeter eller kvadrat-nautisk mil som blir multiplisert med beregnet gjennomsnittlig verdi i ruten. Verdien i ruten er

interpolert ved å bruke dataene målt på brukerens stasjoner og rutinen ZGRI i Map-Library. Det anbefales å bruke Mercator kart for denne funksjonen.

Du må ha en datafil med geografiske posisjoner og måledata (se Kap. 4), eventuelt også en polygonfil for avgrensning av området som skal integreres. Polygonfila kan foreløpig bare inneholde heltallsgrader (LAT1, LON1). Du kommer inn i sidene ved å trykke høyre side-pil i MENY-NYTT KART. "Integrering" har fire FOCUS-sider til endring av variabelverdier.

NB! Ved integrering er siste linje i tekstutskriften under plottet reservert til den verdien en har integrert opp (= totalsum).

Det er også her lagt inn en sperre slik at du ikke kan utføre "Integrering" sammen med "Stasjonsnett", "Isolinjer", "Strøm- og vindvektorer", "Horisontalt snitt" og "Vertikalt snitt". Plottet ville da bli svært uoversiktlig. Du får melding om dette i programmet.

Når du utfører "Integrering" får du melding om hvor mange punkt som ligger innenfor din spesifiserte kartramme og kan da velge om du vil fortsette eller ikke.

Når "Integrering" er utført får du melding om at plotting er ferdig (jfr. Kap. 3.1.2.1), og du trykker en vilkårlig tast for å komme tilbake til menyen.

3.1.2.5 Strøm- og vindvektorer

I ITAKS versjon 1.3 er det nå mulig å fremstille strøm- eller vindvektorer i et horisontalt kart (Fig. 2).

Data som skal inn til denne delen av ITAKS er bredde, lengde, nord/sør og øst/vest komponenten av strøm, vind eller eventuelt en annen parameter som kan fremstilles på denne måten. I tillegg kan en lese inn en eller to kolonner med tall for farge og/eller bredde på vektorene som

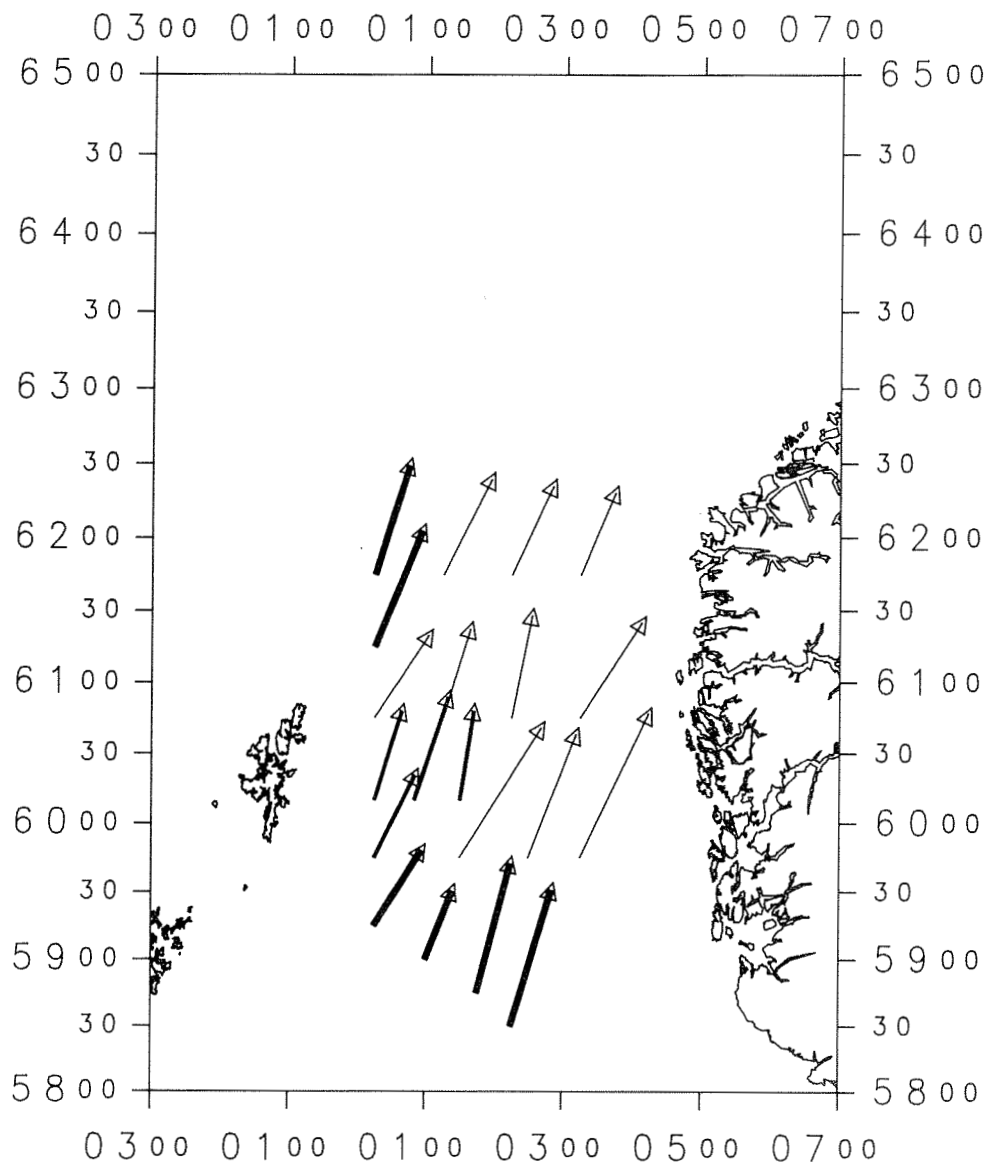


Fig. 2. Plotting av strømvektorer

skal plottes. Med variabelen SARH kan en bestemme størrelsen på pilhodet til vektoren. Ved å angi -1 for ICLRVTR (Farge) og/eller ITHICK (Bredde) betyr at verdien for disse variablene skal leses fra datafilen. Vektorene plottes da som angitt i kolonnene for Farge og/eller Bredde på fila (Tabell 2). Hvis en ønsker at alle vektorene skal ha samme farge og bredde gir en inn det tallet som svarer til farge- og breddekode på den aktuelle FOCUS-siden. "Strøm- og vindvektorer" har kun en FOCUS-side og filtype velges her ved hjelp av variabelen TYPE1.

Tabell 2. Eksempel på filtype når en ønsker å plote vektorer.

	Bredde	Lengde	Nord/sør	Øst/vest	Farge	Bredde
kolonne :1.....2.....3.....4.....5.....	
	5830	0810	4.5	2.9	2	9
	5835	0810	9.5	1.9	2	9
	5840	0810	14.5	-0.9	2	9
	5845	0810	3.5	-5.9	4	10
	5850	0810	2.5	2.9	4	10
	5855	0810	4.0	8.9	4	10

Tykkelsen av vektorene kan endres ved hjelp av variabelen ITHICK. ITHICK =1 gir en heltrukket linje som kun er avhengig av penntykkelse i plotteren. Utover dette er seks linjetykkelser tilgjengelig.

ITHICK=6 ==> Linjetykkele 0.4 mm

ITHICK=7 ==> Linjetykkele 0.8 mm

ITHICK=8 ==> Linjetykkele 1.0 mm

ITHICK=9 ==> Linjetykkele 1.4 mm

ITHICK=10 ==> Linjetykkele 1.8 mm

Tallkodene for verdiene til variablene ICLRVTR og ITHICK svarer ellers til tallkoder for farge og linjetype som en kan finne under "Isolinjer" eller "Vertikalt snitt". Både under "Strøm- og vindvektorer" og "Isolinjer" finnes det hjelpeside for nærmere informasjon om linjetyper. Begrensninger i GPGS-F gjør at det ikke er mulig å definere flere linjetykkelser enn dette. Hvis det er ønskelig kan de forhåndsdefinerte linjetykkelsene endres. Ta da kontakt med programansvarlig. Det arbeides med en versjon hvor en interaktivt kan endre linjetyper og tykkelse etter behov.

Nord- og øst komponentene er positive mens sør og vest komponentene av vektorene er negative. Nord/sør og øst/vest skal oppgies i cm/s. Med variabelen XCM kan du angi at 1 cm i kartet skal svare til en vektor lengde på XCM cm/s.

3.1.2.6 Horisontalt snitt

For å lage en horisontal presentasjon av data som skal inngå i et snitt må du ha datafil med geografiske posisjoner (se Kap. 4). Det er fullt mulig å benytte samme filtype her som under "Vertikalt snitt" (se Kap. 3.1.2.7). Du kommer inn i sidene ved å trykke høyre sidepil. "Horisontalt snitt" har tre FOCUS-sider til endring av variabelverdier.

Når du utfører "Horisontalt snitt" får du melding om hvor mange punkt som ligger innenfor din spesifiserte kartramme og du kan da velge om du vil fortsette eller ikke.

Også i denne delen er det lagt inn en sperre i programmet slik at du ikke kan utføre "Horisontalt snitt" sammen med "Stasjonsnett", "Iso-linjer", "Strøm og vindvektorer" eller "Integrering". Plottet ville da bli lite oversiktlig. Du får melding om dette i programmet.

Når "Horisontalt snitt" er utført får du melding om at plotting er ferdig (jfr. Kap. 3.1.2.1), og du trykker en vilkårlig tast for å komme tilbake til menyen.

3.1.2.7 Vertikalt snitt

Vertikale snitt har man muligheter for å fremstille på to forskjellige måter :

A. Plotte vertikale snitt hvor stasjonene først presenteres i et horisontalt kart (Fig. 3 og Kap. 3.1.2.6)

- et datasett større enn det som utgjør snittet kan være tilgjengelig for programmet. (F.eks. alle stasjoner i en dekning).
- angi at data er ekstrahert via horisontalt snitt.
- ekstrahere stasjoner innenfor et gitt område og dernest vise

disse i et vertikalt snitt.

B. Plotte vertikale snitt direkte

- kun de data som definerer snittet du skal plotte må være tilgjengelig for programmet.
- angi at data ikke er ekstrahert via horisontalt snitt.

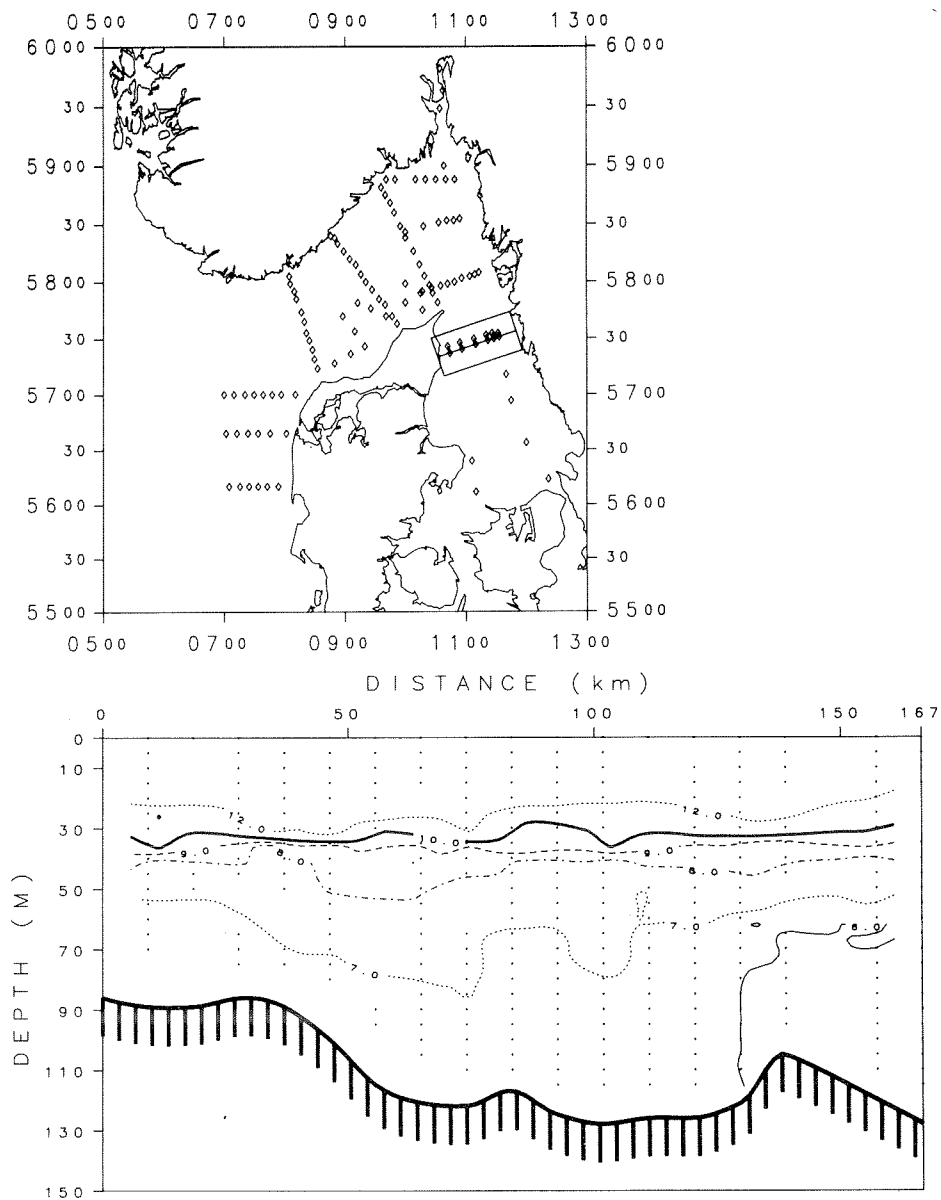


Fig. 3. Plotting av vertikale snitt

For at du skal kunne utføre sekvensen : "Kartramme med kystlinjer" - "Horisontalt snitt" - "Vertikalt snitt" er det viktig at du ved innlesning av data i "Horisontalt snitt" spesifiserer samme filtype som du gjør i "Vertikalt snitt". Ekskludering av stasjoner skjer nemlig her og data om dette overføres til "Vertikalt snitt".

Velges metoden i pkt. A over vil det avgrensede området defineres av en start og sluttposisjon LAT1,LON1 og LAT2,LON2 og en gitt avstand DIST km fra storsirkelen som går gjennom start og sluttposisjonene (se Fig. 3).

Svarer man J på spørsmål om stasjonene kommer fra "Horisontalt snitt" betyr dette at verdiene for LAT1,LON1, LAT2,LON2 og DIST leses fra FOCUS-sidene under "Horisontalt snitt". Disse verdiene overstyrer verdier som beregnes av ITAKS ut fra opprinnelig datasett. De utvalgte stasjonene projiseres inn på storsirkelen som går gjennom start og sluttposisjon for snittet. Stasjoner som faller utenfor vil resultere at følgende melding skrives til skjermen :

> One or several datapoints are located outside the chart- or section frame !

Når du har gitt filnavn for stasjonsdata på side 1 under "Vertikalt snitt" leses rådatafilen for å finne ut hvilke parametre som ligger der, f.eks. SALT og TEMP. Disse vises så til brukeren slik at denne kan velge den parameter som skal presenteres i et vertikalt snitt. Det er mulig å lese inn 10 variable utenom dyp. Ut fra de innleste data finner også programmet nordligste og sørligste bredde og tilhørende lengdegrader. Disse settes som standardverdier til snittet. I mange tilfeller vil disse imidlertid ikke passe som start- og sluttposisjonene i et "Vertikalt snitt". Du bør derfor fortsette til den siden under "Vertikalt snitt" hvor du kan angi start og sluttposisjoner og eventuelt modifisere disse. Når du nå går tilbake for å utføre et "Vertikalt snitt" vil det være disse verdiene som gjelder. Dersom du igjen går tilbake til FOCUS-siden hvor du finner start og slutt posisjonene til snittet, vil disse verdiene bli automatisk beregnet på nytt. Har du gitt inn dine egne verdier for start og slutt-

posisjoner, bør du derfor ikke gå tilbake til denne siden før du plotter snittet.

Også maksimumsdyp for de innleste data blir funnet. Denne verdien pluss 10m settes som maksimumsdyp i snittet og overføres til variabelen MAXDFRM som vises på side 3 i "Vertikalt snitt". Dersom du vil endre maksimumsdyp for snittet må det gjøres på denne siden.

På grunn av begrensninger i Map-Library må startlengde (LON1) alltid ligge vest for sluttlengde (LON2), (LON1 < LON2). Dersom LON1 og LON2 ikke tilfredstiller disse kravene vil programmet justere disse verdiene slik at LON1 < LON2. I de tilfellene startbredde ligger sør for sluttbredde (LAT1 < LAT2) vil snittet sees fra øst, og starten av snittet ligge til venstre på plottet. Dersom en vil se snittet fra vest kan en bytte om start- og sluttposisjoner for snittet. Også i dette tilfelle vil programmet sjekke om startlengde ligger vest for sluttlengde og automatisk korrigere verdiene hvis så ikke er tilfelle.

Format på de innleste data :

1>3	Den første linjen angir antall variable på
2>SALT,TEMP,SIGT	fila. Linje nr. 2 angir variabeltypen med
3>1024 5832 0213 145.5	en karaktervariabel på 4 karakterer f.eks.
4>5	SALT=Saltholdighet, TEMP=Temperatur. Du
5>0.0 32.345 1.23 22.456	kan bruke dine egne forkortelser, men
6>1.0 32.445 1.13 22.556	antall karakterer må være 4. Stasjonsdata
7>2.0 32.545 1.03 22.656	begynner i linje 3. Første nummer er
8>3.0 32.645 1.23 22.756	stasjonsnummer så kommer bredde og lengde
9>4.0 32.745 1.73 22.856	og til slutt er angitt bunndyp på
	stasjonen. I linje 4 er angitt antall
	serier på stasjonen.): Antall dyp med
	observasjoner.

Vær klar over at rekkefølgen av variablene i linje 5 - 9 må svare til rekkefølgen av variablene navngitt i linje 2. Dybde skal alltid være i kolonne 1.

Når du først har fremstilt data ved hjelp av gjøremålet "Horisontalt snitt" vil de to segmentene "Kartramme med kystlinjer" og "Horisontalt snitt" være definert. Disse vil imidlertid bli slettet før du tegner "Vertikalt snitt". Du får melding om dette på skjermen og de tilhørende statusvariablene UTFØRT settes til NEI igjen. Grunnen er at du ellers vil få plottet alle segmentene på skjerm eller plotter og derved få et lite oversiktlig bilde. Nå vil kun det vertikale snittet bli presentert hvis du sender bildet til nytt medium.

Skalering av "Vertikalt snitt"

Det er mulig å foreta den første skalering ved å angi Xmin, Xmax og Ymin, Ymax når en bestemmer hvilket medium det skal plottes på og hvor på mediet plottet skal komme.

I tillegg har du følgende muligheter for "skalering" av vertikale snitt.

- A. Ved bruk av variablene XCM og YCM. Her er XCM utstrekning i cm fra start til slutt av seksjonen. YCM er utstrekning i cm fra minimumsdyp til maksimumsdyp av seksjon.
- B. I tillegg til A) finnes det en mulighet for å gi verdi til to variable kalt XKM og YDP. Her kan en spesifisere hvor langt 1 cm i seksjonen skal være. F.eks. XKM=5 betyr at 1 cm i seksjonen svarer til en avstand på 5 km. På samme måte vil YDP=10 bety at 1 cm i seksjonen svarer til en dybde på 10 m. Denne funksjonen kan neglisjeres ved å sette variablene XKM=0.0 og YDP=0.0.

Bunnprofiler i "Vertikalt snitt"

Bunnprofiler kan genereres på to måter :

- A. Bunnprofil data kan leses inn fra en egen fil. En samlefil for bunnprofiler i fjordområder eksisterer og denne heter (KART) BUNNPROFILER:DATA. Denne filen inneholder posisjoner langs snittet i heltallsgrader, desimalgrader og grader og desimale minutter.

B. Bunnrypene leses fra rådatafila i likhet med de andre parametrene som skal plottes. Dette kan f.eks. være ekkodyp slik bunnrypene er registrert i Havforskningsinstituttets STD-database (Salt, Temperatur og Dyp) (se Tjora, 1984, 1987a, 1987b; Lygren, 1987) eller i næringsssaltdatabasen (se Seglem, 1987).

Både selve bunnprofilen og skyggelegging av bunnen tegnes ut med samme linjetype. Denne er linjetype =10 (se Kap. 6). Foreløpig er det ikke mulig å tegne disse med forskjellige typer, men dette kan endres hvis ønskelig. Dersom bunnkonturene ikke skal tegnes, setter du maksimumsdypet i seksjonen som en negativ verdi. F.eks. gir MAXDFRM=-150.0 et maksimumsdyp i seksjonen på 150m og ingen bunnkontur blir tegnet.

3.1.2.8 Tekst, symboler og linjer

I denne delen av ITAKS kan du tegne linjer, symboler og tekst på "frihånd". I Focus-siden TEKST/SYMBOLER/LINJER, som er vist nedenfor, kan du enkelt endre betingelsene for hva du tegner, f.eks. kan du ha mange skrifttyper, størrelser og vinkler i samme tegning. For å greie dette må du bruke en Tandberg grafisk skjerm. Resultatet kan tegnes på plotter eller lagres på fil.

MAP/ITAKS		NIVÅ - 2.0	

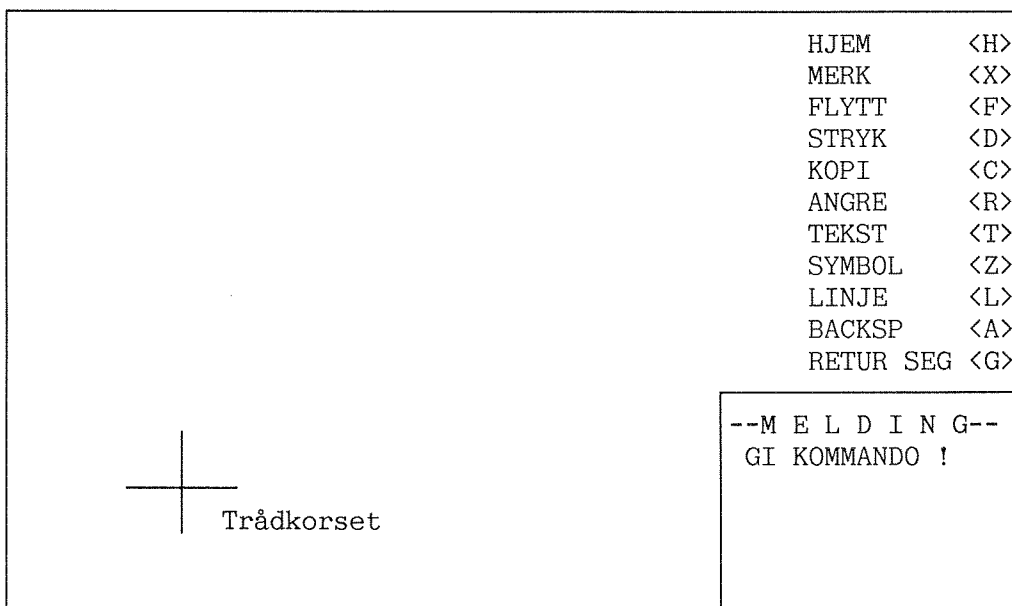
TEKST / SYMBOLER / LINJER			

- TEKST -			
IFONT	Bokstavtype	(Font 0..7)	> .
XT	Bokstavstørrelse i cm	(X-retning)	>
VINKEL	Rotasjonsvinkel	(+/- grader)	>
KURSIV	Kursiv skrift	(+/- grader)	>
- SYMBOL -			
ISYBNR	Symboltype	(1..11)	> ..
DIAM	Diameter i cm		>
- LINJE -			
LTYPE	Linjetype	(1..5)	> .

<== UTFØR	↑ TILBAKE TIL MENY	<HJELP>	INFORMASJON

Når du trykker venstre side-pil (UTFØR) skifter Tandberg skjermen fra vanlig alfanumerisk modus til et grafisk editeringsbilde.

Tilgjengelige kommandoer vises hele tiden til høyre i bildet mens du i en meldingsrute nede til høyre får veiledning om hvor du er og hvordan du skal komme videre. Alle steder der en bokstav står i vinkelhaker betyr dette at du skal trykke CTRL tasten samtidig med angitt bokstav (Eks.: <A> = CTRL + A). For å flytte trådkorset bruker du piltastene. Ut av editeringsdelen kommer du ved å trykke HJEM som i dette tilfellet er <H>.



Kort forklaring av kommandoene :

- HJEM <H> - Går tilbake til Focus-side TEKST/SYMBOLER/LINJER.
Her kan en gi nye verdier eller returnere til LAGE KART MENY.
- MERK <X> - Peker ut et tekst-, symbol- eller linje-segment for FLYTTing, SLETTing eller KOPIering.
NB ! Pek alltid i begynnelsen av et segment !
Det kan skje at GPGS ikke greier å finne et segment ved første forsøk. Prøv da igjen.
- FLYTT <F> - Flytter merket segment.
- STRYK <D> - Stryker merket segment.
- KOPI <C> - Kopierer merket segment.
- ANGRE <R> - Angrer forrige gjøremål på merket segment.
- TEKST <T> - Lager tekst.

For å se hvilke teksttyper som er tilgjengelig
se i appendix.

SYMBOL <Z> - Lager symboler.

LINJE <L> - Lager linjer/punkt.

BACKSP <A> - Sletter siste tegn, symbol eller linje
du laget. Flere anslag gjør at du "blar" deg bakover.

RETUR SEG <G> - for å fortsette tegning av tekst,
symbol eller linjer i ny posisjon (nytt segment
åpnes).

3.1.2.9 Se bilde på skjermen

Velger du dette gjøremålet blir Focus-siden fjernet fra skjermen og grafisk skjerm slått på slik at du kan inspisere det du har tegnet. Du får meldingen :

> Visning ferdig ! Trykk en tast.

For å komme tilbake til LAGE KART MENY trykker du en vilkårlig tast.

NB ! Virker kun når du bruker en skjerm med grafisk hukommelse. Dette fungerer på en Tandberg grafisk skjerm og en Tektronix 4107 skjerm.

3.1.2.10 Plott bilde på nytt medium

Se Kap. 3.1.1 for beskrivelse av Focus-side.

Alle segmenter som er i buffer blir plottet på valgt medium med ønsket plassering og størrelse. Når plottet er ferdig får du melding om dette og du trykker en vilkårlig tast for å komme tilbake til MENY-NYTT KART.

3.1.2.11 Oversikt over bilder i bibliotek

Lister navn på alle lagrede bilder i ditt bibliotek. Samme gjøremål som beskrevet i Kap. 3.3.

3.1.2.12 Hent bilde fra bibliotek

Først må du oppgi bildenavn, hvis dette ikke finnes eller feil oppstår blir det gitt melding (se Kap. 2.3). Du må oppgi hele bildenavnet. Ingen forkortelser godtaes.

Kartbildet (ett eller flere segmenter) blir kopiert fra biblioteket til bufferet og du får spørsmål om du vil se bildet på ditt valgte

medium (normalt Tandberg skjerm).

Husk at et segment (f.eks. en kartramme) har en entydig identifikator (heltall), d.v.s. at hvis du har en kartramme på skjerm og i tillegg henter inn en fra biblioteket så vil du kunne legge disse oppå hverandre. Segmenter hentet fra biblioteker har identifikatorer som er forskjellig fra alle andre segment. Du kan derfor på ethvert tidspunkt hente inn segmenter fra biblioteket og sette disse sammen med "Stasjonsnett", "Isolinjer", "Integrering", "Strøm- og vindvektorer", "Horisontalt snitt" og "Tekst, symboler og linje" segmenter.

Pass på at du ikke tar inn kartrammer fra biblioteket som har forskjellig utstrekning i cm og geografisk område fra det du operer med på skjermen. Vær oppmerksom på at bilder (segmenter) hentet fra biblioteket ikke blir skalert i forhold til den kartramme du har på skjermen, hvis denne er forskjellig fra den du har i biblioteket.

3.1.2.13 Lagre bilde på bibliotek

Først må du oppgi bildenavn. Hvis oppgitt navn er i bruk gies det melding om dette og du kan enten oppgi nytt navn eller fortsette (d.v.s. slette det som var lagret i biblioteket under dette bildenavnet).

NB ! Alle bilder (segmenter) som er i bufferet vil bli lagret under det oppgitte bildenavnet. De blir kopiert inn på den ITAKS genererte biblioteksfilen ITAKS-BIB1:KART. Bildenavnet du oppgir blir registrert på den sekvensielle informasjonsfilen ITAKS:INDX for at ITAKS lett skal kunne finne igjen bildet i biblioteksfilen.

HUSK ! Kartbilder tar stor plass, slett derfor alltid bilder du ikke lenger har bruk for. (Se Kap. 3.4).

Det er lurt å kontrollere at man har tilstrekkelig lagringsplass før man begynner å lagre nye kart.

3.2 Valg av brukerprofil

Dersom du ønsker å bruke en annen brukerprofil enn den som er lagret på ITAKS:INIT går du inn på "Valg av brukerprofil" i hovedmenyen. Nederst på skjermen kommer da følgende spørsmål:

> Vil du virkelig endre INIT-fil navn (J/N/HJEM) ? J

Etter å ha besvart det første med 'J' får du spørsmålet :

> Navn på INIT-fil : SCALE:INIT

Her må du huske på å bruke typen :INIT som siste del av filnavnet. Understreking viser hva du skriver.

En ny brukerprofil kan lages ved å kopiere fila ITAKS:INIT over på en annen fil ved hjelp av følgende Sintran-kommando (Husk doble apostrofer !):

```
@COPY-FILE↓
DESTINATION FILE : "NYPROFIL:INIT"↓
SOURCE FILE : ITAKS:INIT↓
```

Dersom du valgte en annen brukerprofil enn ITAKS:INIT når du startet opp ITAKS (f.eks. SCALE:INIT) vil du bli gjort oppmerksom på dette når du forlater ITAKS. Følgende spørsmål vil da dukke opp på skjermen :

> Brukerprofil er forskjellig fra ITAKS:INIT ! Vil du endre den (J/N)?

Svarer du 'J' på dette spørsmålet betyr det at du nok en gang ønsker å endre navnet på brukerprofilen. Følgende spørsmål kommer da fram på skjermen :

> Gi nytt INIT-fil navn <CR = samme> :

Dersom du ved en feiltagelse svarte 'J' på første spørsmål vil du her kunne angre ved bare å gi RETUR ([↓]). Nå vil du få lagt brukerprofilen tilbake der du opprinnelig hentet den, nemlig på SCALE:INIT.

Du kan også hente INIT-filer (brukerprofiler) fra andre enn ditt eget brukerområde dersom du har lesetilgang på de aktuelle filer. Forsøker

du senere å gå ut av ITAKS vil følgende melding komme opp på skjermen:

> INIT-fila kan ikke skrives tilbake på annen bruker. Trykk en tast !

Dette skjer fordi en vanligvis ikke har skrivetilgang på annen brukers område. Etter å ha trykket en tast fåes derfor følgende melding :

> Gi nytt INIT-fil navn <CR gir ITAKS:INIT> :

Her kan du velge å legge brukerprofilen ut på en egen fil eller legge den tilbake på ITAKS:INIT som er standard brukerprofil for deg. Gir du mangelfull spesifisering av filnavnet får du meldingen :

> Ikke tilstrekkelig spesifisert INIT-fil navn ! Trykk en tast.

Du får da anledning til å rette opp filnavnet.

3.3 Oversikt over lagrede bilder

Følgende bilde kommer da fram på skjermen :

MAP/ITAKS	NIVÅ - 1
LISTE OVER EKSISTERENDE BILDENAVN	
FILANT	Antall bilder i biblioteket > 02
LINJER	ATRIBUTT
TRYKK ↓ I FELTET	

Når du kommer inn i denne siden må du trykke RETUR for å få listet alle bildenavnene. En melding blir gitt og du kan trykke en vilkårlig tast for å retunere til meny.

Navnene som blir listet her er dine egne navn på bilder du har lagret. All referanse til bildet går via disse navnene.

NB ! Skriv bildenavnet fullt ut hver gang du refererer til kartbildet slik at ITAKS kan sammenligne med sin interne bildekatalog.

3.4 Plott bilde fra bibliotek

Oppgi bildenavn, hvis dette ikke finnes eller feil oppstår blir det gitt melding (se Kap. 2.3).

Videre må du bestemme plote-medium og plassering (se Kap. 3.1.1).

Alle segmenter som er lagret under oppgitt bildenavn blir så plottet på valgt medium med ønsket plassering/størrelse. Kartbildet blir i dette tilfelle plottet direkte fra biblioteket og blir altså ikke lagt

i buffer !

Når plottet er ferdig får du en melding og må trykke en tast for å komme tilbake til hovedmeny.

3.5 Slett bilde i bibliotek

MAP/ITAKS	NIVÅ - 1

BESTEM BILDENAVN	

NB! Bruk helst store bokstaver hele tiden !	
BILDENAVN	Oppgi bildenavn (Max 16 tegn) >

AVSLUTT ↓ I FELTET	

Oppgi et bildenavn, hvis dette ikke finnes eller feil oppstår blir det gitt melding og du kommer tilbake til hovedmeny.

Du kan bruke rutinen på to måter :

1. Bildenavn = oppgitt navn blir slettet og bildet (Ett eller flere segment) slettes fra biblioteket. (Frigjør ikke lagringsplass).
2. Bildenavn = '*' sletter alle bildenavn i bildekatalogen og renser (sletter alle segmenter) i biblioteket. (Frigjør ikke lagringsplass).

Før du sletter vil du få valget mellom å fortsette eller avbryte.

3.6 Pakke bibliotek

Biblioteksfila ITAKS-BIB1:KART blir ekspandert hver gang du lagrer et nytt bilde. Slettes et bilde fra biblioteket som nevnt i forrige avsnitt, frigjøres ikke plass på fila ITAKS-BIB1:KART. Kun bildekatalogen som du finner på ITAKS:INDX oppdateres og denne oppdatering gjøres gyldig i ITAKS. Ved å utføre gjøremålet "Pakk bibliotek" vil du fysisk kunne fjerne bildesegmenter på ITAKS-BIB1:KART som du tidligere bare har fjernet rent logisk. Kun de bildene som er registrert på ITAKS:INDX vil være lagret på ITAKS-BIB1:KART etter at denne funksjonen er benyttet. På denne måten oppnår du også bedre utnyttelse av plassen på ditt brukerområde.

3.7 Sintran-kommandoer

Du kan oppgi en Sintran-kommando, hvis den ikke er gyldig blir den ignorert. Det blir gitt melding når kommandoen er utført og du må trykke en tast for å komme tilbake til hovedmeny. Vanlig Sintran syntaks gjelder.

Eksempel : FILE-STATISTICS ITAKS-BIB1:KART kan gies som :

> Oppgi Sintran-kommando: FI-STAT ITAKS-BIB1:KART^d

NB ! Det er ikke alle Sintran-kommandoer som kan utføres i programmet. (Begrensninger i monitor kall COMND (Command)).

Noen sintran-kommandoer som kan utføres :

FILE-STATISTICS
LIST-FILES
DELETE-FILE
CREATE-FILE
RENAME-FILE
EXPAND-FILE
COPY-FILE
SET-FILE-ACCESS

LIST-OPEN-FILES

OPEN-FILE

CLOSE-FILE

3.8 Valg av kartdatabase

Her kan du skifte kartdatabase for tegning av kystlinjer i et nytt geografisk område eller for bedre oppløsning i spesielle områder som f.eks. Masfjorden. Ved dette valget i hovedmenyen får du fram FOCUS-siden som vist under.

MAP/ITAKS		NIVÅ - 1	

FORANDRE KARTDATABASE			

Brukernavn >	P-O:KART	Base (Max 10 tegn) >	KARTDATA
YLATMN	Sørligste breddegrad	>
YLATMX	Nordligste breddegrad	>
XLONMN	Vestligste lengdegrad	>
XLONMX	Østligste lengdegrad	>
LATSTP	Breddegradssteg	>
LONSTP	Lengdegradssteg	>
NHOR	Antall horisontal	>	...
NVER	Antall vertikal	>	...
NRUTE	Antall ruter	>
NLATIN	Antall intervall pr. breddegrad	>	..
NLONIN	Antall intervall pr. lengdegrad	>	..
NDEPTH	Antall dybdenivåer	>	..
DEPTHS	Dybdenivåer (m)	>	0000.0>-0009.0>-0009.0>-0009.0>-0009.0
		>	>-0009.0>-0009.0>-0009.0>-0009.0>-0009.0>-0009.0>-0009.0>-0009.0
		>	>-0009.0>-0009.0>-0009.0>-0009.0>-0009.0>-0009.0>-0009.0>-0009.0

<HJELP> INFORMASJON		TILBAKE TIL HOVEDMENY	

På denne siden er det kun de to første feltene du har lov å endre. De andre feltene er informasjon om kartdatabase som kan være nyttig når du skal tegne kart. Variabelen DEPTHS angir hvor mange "egenskaper" basen inneholder. I vårt tilfelle er disse egenskapene dybdekoter. I prinsippet kunne de ha vært en hvilken som helst annen kartinformasjon f.eks. veitraseer, ledningsnett osv. Dersom -0009.0 er vist i feltet betyr dette at det ikke er gitt noen egenskap for dette nivået. NB ! Unøyaktige opplysninger her kan føre til feil ved tegning av kart eller lesing av oppgitte filer.

Det finnes også et program for å lage nye kartfiler på det formatet Map-Library og ITAKS krever (Christiansen & Westgård, 1988). Brukeren må ha ett eller flere sett med geografiske koordinater av kystlinjer og/eller dybdekoter. Disse kalles med en fellesbetegnelse for kartdatabaser.

På Havforskningsinstituttet finnes to ND-500 maskiner kalt henholdsvis Havfor1 og Havfor2. Kartdatabasene på disse maskinene er vist i Tabell 3.

Tabell 3. Oversikt over kartbaser ved Havforskningsinstituttet.

Maskin	Brukernavn	Basenavn
Havfor1	Pack-One:Kart Pack-Two:Kart " " " "	KARTDATA PORSANGEN AKUP MASFJORDEN GIJ VERDEN
Havfor2	Kart	PORSANGEN AKUP KARTDATA MASFJORDEN GIJ VERDEN

På siste FOCUS-side under "Kartramme med kystlinjer" kan du angi hvor mange og hvilke dybdenivåer du er interessert i å plote. Nivå 1 er kystlinjer. Det er pr. idag kun mulig å plote dybdekoter for kartdatabasen AKUP (se Tabell 3). Det geografiske området denne basen dekker er: $69^{\circ}30'N-74^{\circ}30'N$; $15^{\circ}\emptyset-35^{\circ}\emptyset$.

4 Datafiler og filgrensesnitt

For at en skal kunne produsere kart og tegne f.eks. isolinjer og stasjonsnett må de geografiske posisjonene og de tilhørende målte variablene, ligge på en fil som kan leses av ITAKS. På FOCUS-siden vist under, kan en innefor visse rammer angi spesifikasjonene for de data

som skal leses inn.

MAP/ITAKS		NIVÅ - 2.0	

STASJONSPOSISJONER - GRENSESNIITT MOT DATAFIL			

FILNAVN1	Navn på datafil	>
FTYPE1	Variabel rekkefølge	>	.
	1 = Lengde,Bredde,Målt verdi		
	2 = Bredde,Lengde,Målt verdi		
PTYPE1	Posisjonstype	>	.
	1 = heltallsgrader		
	2 = desimalgrader		
TYPE1	Variabelsett		
	1 = x,y,z		
	2 = x,y		
	3 = x,y,attr1,....,attr6	>	.
FORMAT1	Format (Organisering av data)	>
ATTR	Attributter fra fil ? (J/N)	>	.

↖ TILBAKE TIL MENY		==> NESTE SIDE	

Ved Havforskningsinstituttet er det lagt opp et system som gjør at man fra databaser for ulike typer miljødata, f.eks. næringssalter (Seglem, 1987), STD-data (se Tjora, 1984, 1987a, 1987b; Lygren, 1987), egg-, larve- og dyreplankton data (Bakkeplass, 1988) kan lage filer som ITAKS kan lese.

For at datafilen skal kunne lese av ITAKS må følgende forutsetninger være innfridd :

- Filnavnet kan maksimalt ha 16 tegn.

Filnavnet er tilstrekkelig definert ved minste entydige forkortelse. Hvis Sintran (se Kap. 1.3) ikke finner filen gies det melding om dette når du prøver å utføre gjøremålet (se Kap. 5). Datafilen må være sekvensiell med data i alle linjer. Har du laget datafilen i NOTIS-WP må du sørge for at filen er i 7-bits lagringsformat (se på statuslinjen i NOTIS-WP).

- Rekkefølge. Posisjoner og eventuelle måledata må ligge i en av rekkefølgene :

- Lengdegrad, Breddegrad, Måledata
- Breddegrad, Lengdegrad, Måledata
- Breddegrad, Lengdegrad, Måledata, Attr1,Attr6
- Lengdegrad, Breddegrad, Måledata, Attr1,Attr6

For å tegne et nett med stasjoner og stasjonsnummere settes måledatokolonnen lik stasjonsnummeret.

- Posisjonstype. Posisjoner oppgis i heltalls- eller desimalgrader.

- Heltallsgrader (f.eks. 0415,6030 = 4⁰15',60⁰30')
- Desimalgrader (f.eks. 04.25,60.50 = 4⁰15',60⁰30')

Vestlige lengder og sørlige bredder oppgis med negativt fortegn.

- Variabelsett

- x,y,z betyr at du har lengde, bredde og måledata eller bredde, lengde og måledata liggende på datafilen.
- x,y betyr at du har bredde og lengde eller lengde og bredde på filen din.
- x,y,z,attr1,....attr6 betyr at du har bredde, lengde, måledata og attributter eller lengde, bredde, måledata og attributter på din datafil.

- Formatet (dataorganisering) beskrives i Fortran notasjon.

Beskrivelsen av formatet kan være på maksimalt 30 tegn, alle bokstaver må være store. Du kan omslutte formatet med parenteser (..) eller skrive formattsatsen direkte. Hvis variablene ligger etter hverandre og det ikke finnes andre data på filen kan du bruke fritt format som angis med en *.

Eksempler på format :

- a) I4,2X,I4,2X,F5.2
- b) (F5.2,1X,F6.2,1X,F7.2)
- c) *
- d) (2I5,F5.1,2I5,2F7.1,2I5)

Hvordan det da må se ut hvis du leser filen med Notis :

kolonne :1.....2.....3.....4.....5.....				
	0330	5645	13.55						
a)	0330	5700	14.53						
	0330	5715	14.56						
	56.33	- 6.28	-133.23						
b)	56.33	- 5.28	-123.13						
	56.33	- 4.28	-113.04						
	5545	0300	12.						
c)	+5600	0300	13.0						
	+5630	0300	14.00						
	0315	6145	0.0	2	2	0.4	0.3	1	1
d)	0215	6145	5.0	-1	3	0.4	0.3	0	1
	0115	6145	5.0	-1	6	0.5	0.4	1	1

Rekkefølgen av de seks attributt variablene er : LABPT (antall desimaler etter komma i datapunkt-merkelapper), ISYMB (symboltype), SSYMB (størrelse av symbol i cm), SLABL (størrelse av karakterer i datapunkt-merkelapper i cm), LTYPE (linjetype mellom datapunkt) og IREPOS (posisjonering av symbol relativt til datapunkt).

Når du har utført et gjøremål med ITAKS f.eks. "Kartramme med kystlinjer", "Isolinjer" eller "Vertikalt snitt" registreres dette som frittstående enheter eller segmenter som har egne nummer. I Tabell 4 er angitt hvilke nummer som er reservert for de forskjellige gjøremålene. Disse tallene kan være nyttig å kjenne dersom det oppstår GPGS-F feil under plottingen.

Tabell 4. Segmentnumre for de ulike gjøremålene.

Gjøremål	Antall (max)	Segment nummer
Kartramme	1	1
Stasjonsnett	1	2
Isolinjer	1	3
Integrering	1	4
Strøm / vind	1	5
Horisontal	1	6
Vertikal	1	7
	Ledige	8 --> 100
Tekst	299	Start 101 --> 400
Symboler	299	" 401 --> 700
Linjer	299	" 701 --> 1000
Kopisegment	299	" 1001 --> 1300
Bibliotek	299	" 1301 --> 1600
	Ledige	" 1601 --> 4096

Filer ITAKS lager i ditt brukerområde :

Når du kjører ITAKS for første gang vil programmet lage fire filer på ditt brukerområde for å kunne lagre bilder. Disse er:

ITAKS-BIB1:KART - Bibliotek. En direktefil for lagring av hele segmenter identifisert av et heltall (0-4096). Lagring/henting/sletting gjøres av GPGS-rutiner.

ITAKS:INDX - Inneholder totalt antall segment lagret (d.v.s. bilde- elementer), antall registrerte bilder og tilhørende bildenavn lagret på ITAKS-BIB1:KART. For hvert bilde er også registrert startsegmentnr i bibliotek og antall segmenter (bilde-elementer) i hvert bilde.

Eksempel på ITAKS:INDX

kolonne :1.....2.....3.....4.....5.....
	3 2
LINJER	1301 1
ATRIBUTT	1302 2

ITAKS-BIB2:KART - Brukes når tekst, symbol og linjer tegnes for å kunne redigere bildet inntil man er fornøyd. Denne filen slettes etter kjøring, og har ingen betydning for de lagrede bildene.

ITAKS:INIT - Denne filen inneholder de data du fylte inn i datafeltene sist du brukte ITAKS. Hvis du ikke har en ITAKS:INIT fil liggende på ditt brukerområde kopieres INIT-fila automatisk fra bruker KART med "fornuftige" startverdier satt. Også første gang ITAKS brukes får du systemets standard verdier på denne måten.

NB ! Hvis du vil slette ITAKS-BIB1:KART må du alltid slette tilhørende indexfil ITAKS:INDX fordi ITAKS bruker indexfilen som utgangspunkt for den interne filkatalogen og videre lagring.

5 Feilmeldinger ved lesing av datafil

Når feil oppstår ved lesing av datafil gir ITAKS melding om dette på skjermen. Feilsituasjonene ITAKS håndterer :

- Sintran finner ikke filnavnet du har oppgitt og du får meldingen :

> Kan ikke finne datafilen : nnnn Sjekk filnavn, trykk en tast !

Du vil bli sendt tilbake til menyen og bør sjekke om du har filen, eller om filnavnet var feil skrevet.

- Feil oppstår under lesing av datafilen og du får meldingen :

> Feil oppstått under lesing av datafilen : nnnn Trykk en tast !

Du vil også nå bli sendt tilbake til menyen og bør sjekke at oppgitt format er i samsvar datafilens organisering.

- Datafilen inneholder flere datapunkt enn det interne bufferet er dimensjonert til (NPMAX=5000) og du får meldingen :

> Datafilen : nnnn har flere data enn NPMAX=xxx, fortsette (J/N) ?

Her kan du velge om du vil fortsette kjøring med de data som allerede er lest inn, eller om du vil avbryte gjøremålet.

6 Linjetyper

I GPGS-F er det kun mulig å anvende 11 forskjellige linjetyper (Anon., 1984a). Linjetyperne 0 -5 er såkalt "hardware" genererte. Dette betyr at den samme linjetypen vil variere noe, avhengig av plottemedium. Nå er det midlertid mulig å definere sine egne linjetyper i GPGS-F. Disse linjetyperne kalles "software" genererte og skal være identisk fra ett plottemedium til et annet. Denne muligheten er tatt i bruk i ITAKS. Linjetype 6 - 10 er nå, med unntak av gjøremålet "Strøm- og vindvektorer", avsatt til følgende linjetyper

- 6 : Prikket linje med 0.3 mm avstand mellom prikkene
- 7 : Prikket - stiplet linje
- 8 : Stiplet linje
- 9 : Heltrukken linje med tykkelse 0.4 mm
- 10 : Heltrukken linje med tykkelse 0.8 mm (jfr. bunnkontur Kap. 3.1.2.7)

Figur 4 gir en oversikt over linjetyperne slik de fremkommer på en HP-7550A.

7 Farger

Bruk av farger blir mer og mer vanlig ved fremstilling av kart eller annen grafikk. ITAKS er forberedt for dette. Havforskningsinstituttet har forskjellige plottemedier hvor en kan framstille fargegrafikk fra ITAKS. Plotterne HP-7550A, HP-7475A og Calcomp benytter alle fargepenner. Tektronix 4693D fargeplotter er imidlertid basert på et annet prinsipp: smelting av voks på papir. Sammen med en Tektronix 4510 rasteringsenhet gir Tektronix 4693D bilder med "laser"-kvalitet. Med

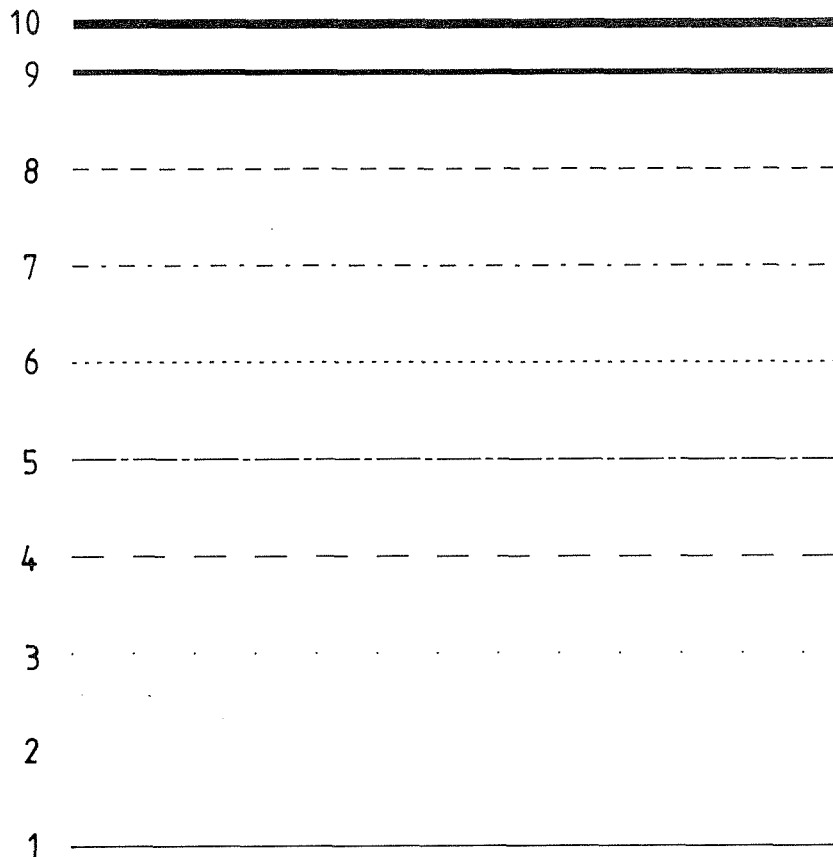


Fig. 4. Oversikt over linjetyper på HP-7550A plotter

denne fargeplotteren kan en lage papirkopier såvel som transparenter. Du har til rådighet en palett på 16.7 millioner farger eller 256 ulike gråtoner. Oppløsningen er 300 punkter pr. tomme. Denne fargeplotteren kan pr. idag bare benyttes mot plottepakken GPGS-F (Anon., 1984a). Imidlertid vil en også på sikt kunne koble den mot en MacIntosh II, IBM MS-DOS eller SUN Unix maskiner, og utnytte fargemulighetene som finnes på disse maskinene. Fargekoder brukt av GPGS-F er vist i Tabell 5.

Tabell 5. Fargekoder. Modifisert etter GPGS-F manualen (se Anon., 1984a).

Kode	Farge	
0	Sort	Bakgrunnsfarge
1	Hvit	
2	Rød	
3	Grønn	
4	Blå	
5	Blågrønn	
6	Fiolett	

NB ! Vær oppmerksom på at sort er bakgrunnsfarge på terminalen, mens noen plottere kan bruke hvit som bakgrunnsfarge.

8 Skjermer og andre plottemedier på Havforskningsinstituttet

Ethvert utskriftsmedium på Havforskningsinstituttets to ND-500 anlegg har en såkalt "peripheral file" som den er koblet mot. Denne fila har et navn og et nummer kalt logisk "device" nummer. Når du sender data til ett plottemedium brukes perifer-fila som en adresse slik at tegningen kommer på riktig plottemedium. I programmene som håndterer utskriftsmediene, kalt drivere, åpnes fila som dataene sendes til. Dette foregår ved hjelp av en OPEN-sats i driverne. I tillegg har GPGS-F (Anon., 1984a) et eget nummer for de enkelte utskriftsenhetene.

En OPEN-sats ser ut som følger :

```
OPEN (IUNIT,File='HP-7550A',STATUS='Old')
```

IUNIT er her et SINTRAN filnummer. Dette kan være det samme som logisk "device" nummer, men trenger ikke være det. Filnummer i SINTRAN kan imidlertid bare ha verdier fra 2 - 127, mens logisk "device" nummer kan være høyere enn dette. Fordelen med å la IUNIT og logisk "device" nummer være identiske er at en da kan ha kontroll med plottemediet fra sitt hovedprogram. F.eks. er det da mulig å sjekke om andre har

reservert det plottemediet en tenker å benytte. På denne måten kan en vente med å sende plott til mediet inntil det er ledig for å motta utskrift. Denne muligheten er tatt i bruk i ITAKS versjon 1.3. I Tabell 6 er vist hvilke plottemedier Havforskningsinstituttet benytter og koblingen mellom nummerering i SINTRAN og GPGS-F.

Tabell 6. Plottemedier på Havforskningsinstituttet. Alle tall er desimaltall.

Medium	HAVFOR1			HAVFOR2		
	Sintran Filnr.	Log. dev.nr.	GPGSnr.	Sintran Filnr.	Log. dev.nr.	GPGSnr.
Textronix 4012	1	1	20	1	1	20
Calcomp	-	-	81	61	61	81
HP 7221	60	60	11	-	60	11
HP 7475	61	61	04	-	-	04
Tandberg TDGO	1	1	63	1	1	63
Philips-IFB *		36	306		38	306
Philips-AKVA *		-	-		48	306
Philips-OSEAN *		38	306		-	-
Philips-2 *		-	-		36	306
Watanabe			14			14
HP 7550 OSEAN	58	58	80	-	58	80
HP 7550 BLOKJEM	44	1024	80	-	-	-
Textronix 4105	1	1	62	1	1	62
Textronix 4107	1	1	59	1	1	59
Tektronix 4693D	45	1029	89	-	-	-

* Sintran filnummer er uinteressant i denne forbindelse da det er generert "spooling" (utskriftskø) på disse mediene.

Når det gjelder driveren til HP-7550A plotteren er denne spesiell. Fordi det benyttes to plottere av denne typen på HAVFOR1 medfører det at driveren må kunne håndtere to ulike periferfiler. Dette er løst ved å innføre en COMMON-blokk i driveren som også må brukes i ditt hovedprogram hvis du selv programmerer mot GPGS-F og skal bruke ett av disse plottemediene. Denne COMMON-blokken overfører navnet og filnummeret til periferfila fra hovedprogram til driver. Du får tak i denne COMMON-blokken ved å innføre følgende linje i ditt hovedprogram :

```
$INCLUDE (GPGS-500)DVCHP7550:INCL
```

Common-blokken DVCHP7550:INCL inneholder følgende linjer :


```
C
C   Inkluder COMMON blokk som overfører unit nr. og filnavn på
C   utenhet i det tilfelle hvor det skal kjøres to HP-7550 plottere
C   mot HP70-driveren.
```

```
C
C           Character Utfil*16
C           Integer  Utnr
C
C           COMMON / Utdev / Utfil, Utnr
```

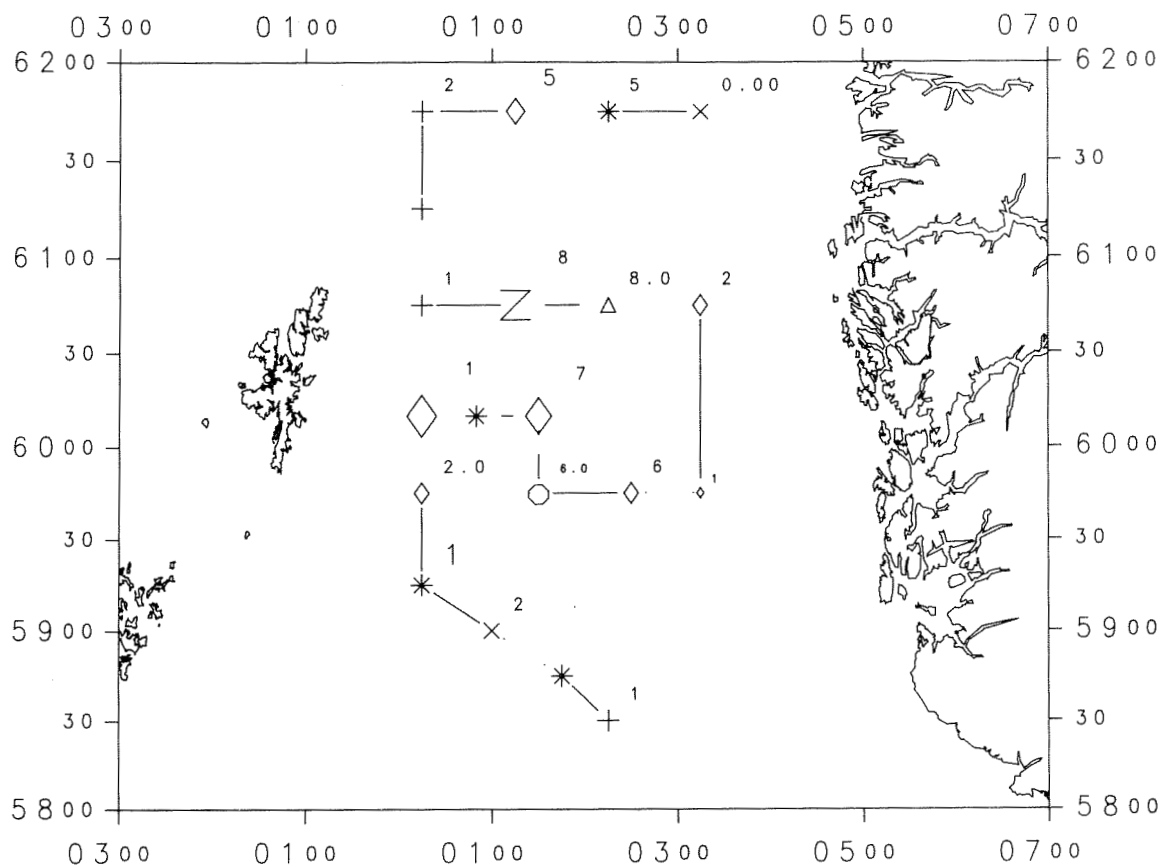
```
C
```

De to periferfilene som benyttes til de to HP-7550A plotterne på HAVFOR1, ligger på bruker SYSTEM og heter HP-7550A og HP-BIOKJEM. Hvis du ønsker å vite navnet på andre periferfiler kontakt systemansvarlig på HAVFOR1 og/eller HAVFOR2.

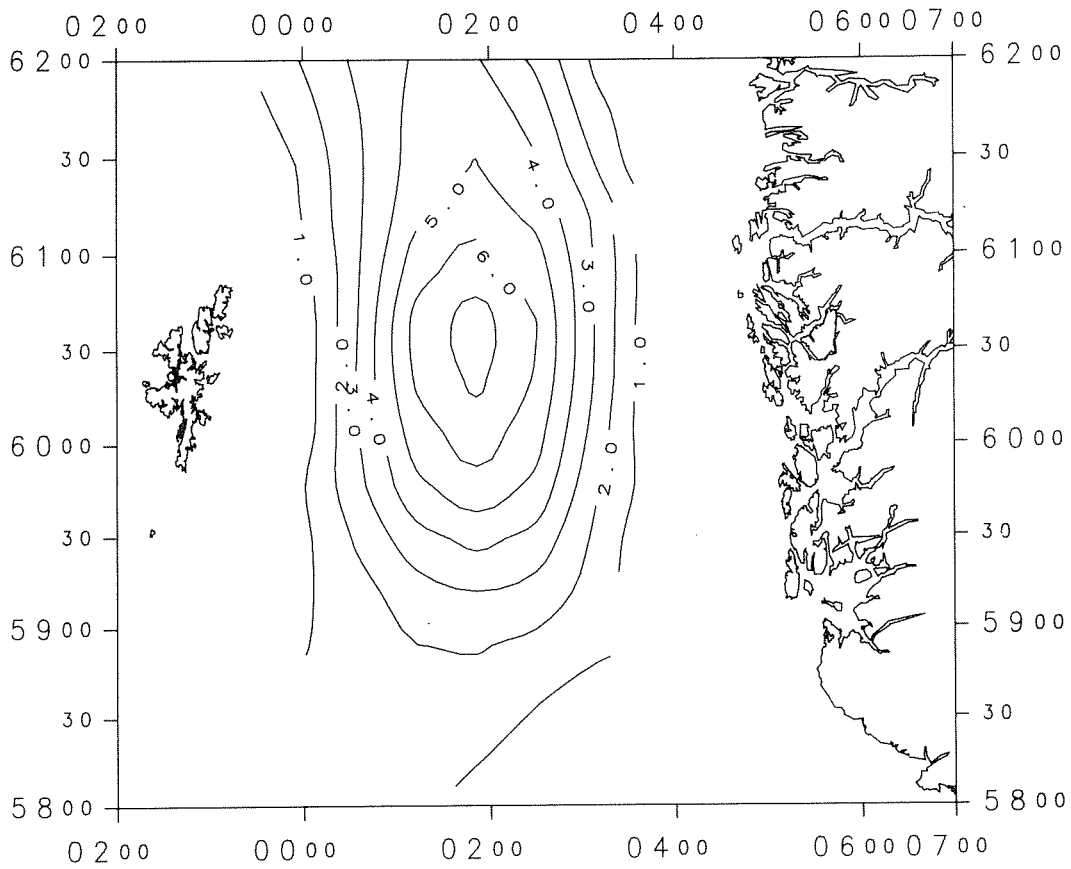
Når det gjelder skjermtyper kan ITAKS versjon 1.3 benyttes på Tandberg terminaler, IBM-kompatible MS-DOS maskiner knyttet til NORD-500 anlegget ved bruk av emulatoren TGRAF-07 fra Grafpoint Inc. (Tektronix 4107 fargeemulator), MacIntosh SE og MacIntosh II (Tektronix 4014, monokrom og Tektronix 4105 fargeemulator) ved bruk av emulatoren VersaTerm-PRO (Abelbeck Software), samt TGRAF-07 fra Grafpoint Inc. som også leveres til MacIntosh II og SE. En ny versjon av VersaTerm-PRO vil muligens inkludere deler av en Tektronix 4107 emulator. Forskjellen på Tektronix 4105 og 4107 er at Tektronix 4105 emulatoren ikke støtter lokal billedhåndtering på skjermen. Dette er en opsjon i ITAKS, men er ikke nødvendig dersom en bare skal fremstille plott på skjermen og deretter sende det til en plotter. Tektronix 4105/4107 emulatorene krever et DEC-VT100 alfanumerisk grensesnitt mot Sintran operativsystem på Nord-500 ved Havforskningsinstituttet. Denne skjermtypen må derfor defineres mot Sintran før en starter ITAKS. Dette gjøres ved følgende kommandoer til Sintran operativsystem:

```
@SET-TERMINAL-TYPE↓
  TERMINAL NUMBER:↓
  TERMINAL TYPE:6↓
@
```

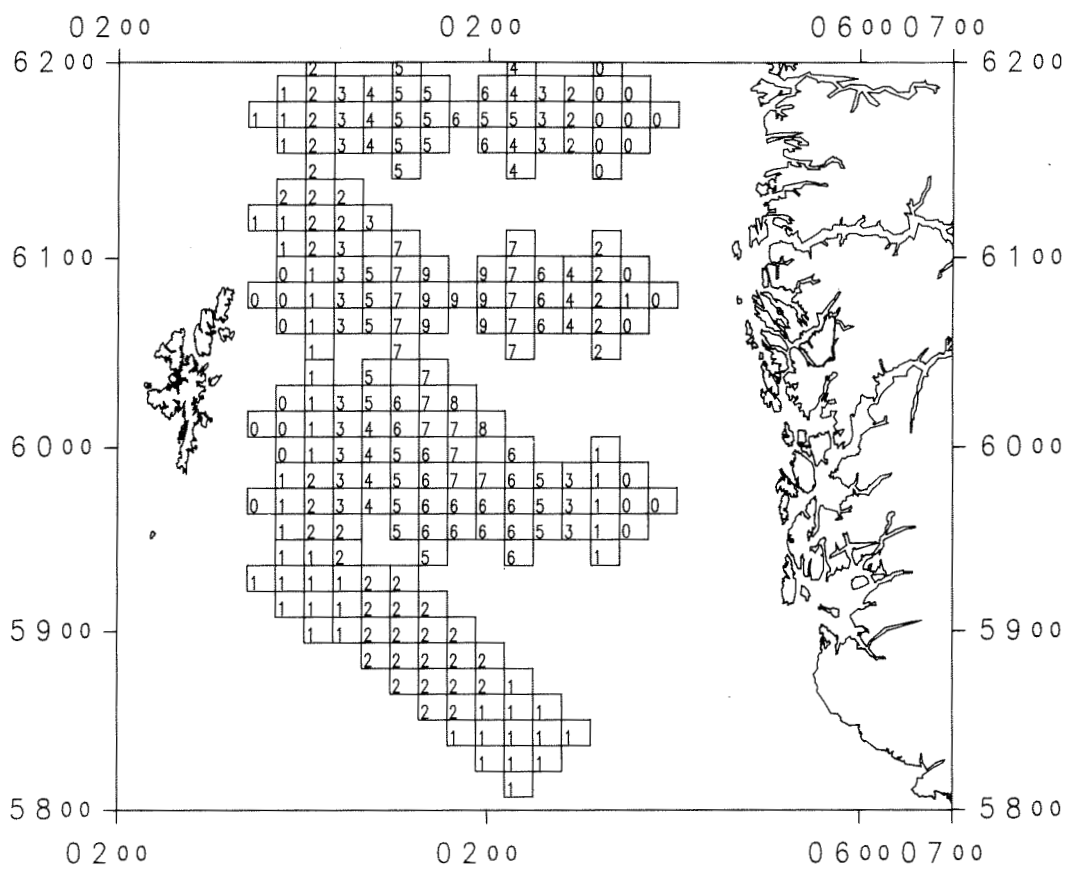
9 Eksempler på bruk av ITAKS



Eksempel 1. Kurslinjer og ulike stasjonstyper symbolisert ved hjelp av forskjellige symboler og linjetyper.



Eksempel 2. Isolinjer av en målt variabel i et geografisk område.

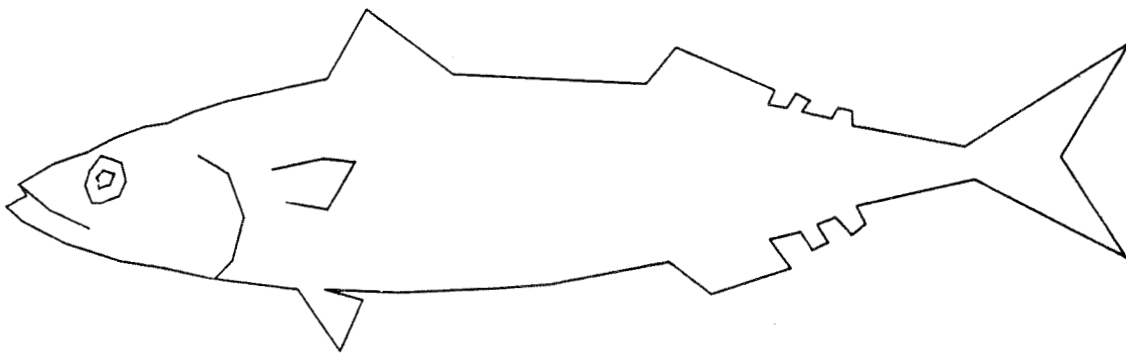


Eksempel 3. Integrering av en målt variabel i et geografisk område.

TOTALSUM : .18E+12

Formler

$$F(t) = \int_a^b t^{\frac{n}{2}} dt$$



Strekfigurer

Tekst i alle typer og størrelser

Eksempel 4. Bruk av den interaktive delen i ITAKS.

10 Tabell over tilgjengelige skrifttyper

Tabellen er hentet fra GPGS-F manualen (se Anon., 1984a)

ASCII-nummer	96 97 98 99 100101 102 103 104 105 106107108 109110 111 112113 114 115116 117 118 119120 121122 123124125 126
ASCII-tegn	` a b c d e f g h i j k l m n o p q r s t u v w x y z { } ~
GPGS standard	` a b c d e f g h i j k l m n o p q r s t u v w x y z { } ~
SIMPLEX ROMAN	` a b c d e f g h i j k l m n o p q r s t u v w x y z { } ~
COMPLEX ROMAN	` a b c d e f g h i j k l m n o p q r s t u v w x y z { } ~
COMPLEX ITALIC	` a b c d e f g h i j k l m n o p q r s t u v w x y z { } ~
DUPLEX ROMAN	` a b c d e f g h i j k l m n o p q r s t u v w x y z { } ~
SIMPLEX GREEK	α β γ δ ε φ χ η ι ψ κ λ μ ν ο π ϑ ρ σ τ υ ν ω ξ υ ζ
COMPLEX GREEK	α β γ δ ε φ χ η ι ψ κ λ μ ν ο π ϑ ρ σ τ υ ν ω ξ υ ζ
MATHEMATICAL	α § θ ε ∫ ϕ { [] ° Π θ } Σ ~ √ × { } ~

ASCII-nummer	64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95
ASCII-tegn	▽ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\] ^ _
GPGS standard	@ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\] ^ _
SIMPLEX ROMAN	▽ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\] ^ _
COMPLEX ROMAN	▽ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\] ^ _
COMPLEX ITALIC	▽ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\] ^ _
DUPLEX ROMAN	▽ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\] ^ _
SIMPLEX GREEK	A B Γ Δ E Φ X H I Ψ K Λ M N O Π Θ P Σ T τ N Ω Ξ T Z
COMPLEX GREEK	A B Γ Δ E Φ X H I Ψ K Λ M N O Π Θ P Σ T τ N Ω Ξ T Z
MATHEMATICAL	@ ∞ § θ ε ∫ ϕ { [] ° ∏ θ } ∑ ~ √ × []

ASCII-nummer	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
ASCII-tegn	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	
GPGS standard	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	
SIMPLEX ROMAN	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	
COMPLEX ROMAN	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	
COMPLEX ITALIC	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	
DUPLEX ROMAN	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?	
SIMPLEX GREEK																0	1	2	3	4	5	6	7	8	9							
COMPLEX GREEK																0	1	2	3	4	5	6	7	8	9							
MATHEMATICAL			×	∅	<	>	×	±	∓	·	÷	ϕ	†	‡											∞			≤	→	≅		

11 Tabell over tilgjengelige fonttyper

Tabellen er modifisert etter Appendix H i GPGS-F manualen (se Anon., 1984a)

0	: GPGS-F Standard
1	: Simplex Roman
2	: Complex Roman
3	: Complex Italic
4	: Duplex Roman
5	: Simplex Greek
6	: Complex Greek
7	: Mathematical

Fontene 5 - 7 er ikke gjort operative i ITAKS. Ta kontakt med programansvarlig hvis dette er ønskelig.

12 REFERANSER

- Anonymous, 1985. FOCUS : Screen Handling System Reference Manual.
ND-60.137.5 EN. Norsk Data A/S.
- Anonymous, 1984a. GPGS-F User's guide. 6th edition. NORSIGD.
Tapir forlag ISBN 82-519-0631.
- Anonymous, 1984b. TDV 2200 S : Reference Manual for Graphics.
Publ. no. 5458. Tandberg Data A/S.
- Anonymous, 1986a. ND FORTRAN : Reference Manual ND-60.145.7A.
Norsk Data A/S.
- Anonymous, 1986b. Kart og kartbruk. Universitetsforlaget A/S,
2. utgave. ISBN 82-00-18150-2.
- Bakkeplass, K., 1988. Beskrivelse av Larve- og Egg databasen. Internt
notat, Avdeling for Biologisk og Kjemisk Oseanografi. Hav-
forskningsinstituttet. 3 s.
- Berge, T., 1987. Plotlib-1. Versjon F. Norsk Polarinstitutt.
- Christiansen, A. og T. Westgård 1988. FORSKERKART.
Brukerveiledning, del 2 av 6 : Les-Topografi, versjon 1.00
august 1988. Rapportnr. 18. Havforskningsinstituttets Egg- og
Larveprogram.
- Engelsen, H. og T. Westgård 1986. Brukerveiledning for ITAKS,
Interaktiv Tegning Av Kart og Snitt. Rapportnr. PS8603.
Havforskningsinstituttet.
- Lygren, S., 1987. Kort beskrivelse av STD-DB. Internt notat, Avdeling
for Fysisk Oseanografi, Havforskningsinstituttet. 10 s.
- Seglem, K., 1988. Hvordan finne fram i avdelingens database ? Internt
notat, Avdeling for Biologisk og Kjemisk Oseanografi,
Havforskningsinstituttet. 6 s.

Tjora, P., 1984. Behandling av CTD-data. Rapp. Havforskningsinstituttet. FO 8408: 20 s.

Tjora, P., 1987a. Primærbehandling og presentasjon av Fysisk-Oseanografiske data. Hovedfagsoppgave i Informatikk, Studieretning databehandling. Institutt for Informatikk, Universitetet i Bergen. 89 s.

Tjora, P., 1987b. Primærbehandling og presentasjon av Fysisk-Oseanografiske data. Programlisting. Hovedfagsoppgave i Informatikk, Studieretning databehandling. Institutt for Informatikk, Universitetet i Bergen. 184 s.

Westgård, T. og A. Christiansen 1988. FORSKERKART. Brukerveiledning, del 3 av 6 : Koble-Linjer, versjon 1.00 august 1988. Rapportnr. 18. Havforskningsinstituttets Egg- og Larveprogram.

Westgård, T. og A. Christiansen 1988. FORSKERKART. Brukerveiledning, del 4 av 6 : Lag-KartBase, versjon 1.00 august 1988. Rapportnr. 18. Havforskningsinstituttets Egg- og Larveprogram.

Westgård, T. 1984. MAP-LIBRARY : A user's guide to a subroutine library for presentation of marine data. Rapportnr. PS8405. Havforskningsinstituttet.

Westgård, T., Knutsen, T. og A. Christiansen 1988. FORSKERKART. Users Guide, part 5 of 6 : Map-Library, version 1.00 August 1988. Report no. 18. Havforskningsinstituttets Egg- og Larveprogram.

STIKKORD LISTE

<u>Referanse</u>	<u>Side</u>
Attributter	19, 41, 42
Biblioteker	33
Brukerprofil	5, 8, 11, 34, 35
Buffer	7, 15, 17, 32, 37
Bunnprofil	28
Datafiler	6, 40
Farger	5, 46, 47
Filgrensesnitt	40
FOCUS	3, 9, 59
Format	15, 27, 41-43, 45
Funksjonstaster	8, 9
Gjøremål	5, 7, 9, 16, 30, 32, 43, 44
Hjelpesider	9
INIT-fil	34, 35
Integrering	16, 17, 20, 21, 44, 53
Isolinjer	3, 6, 16, 17, 20, 40, 44, 52
ITAKS:INIT	8, 11, 34, 35, 45
ITAKS-BIB1:KART	12, 38, 44, 45
Kartbase	12
Kartdata	39, 40
Kartdatabase	11, 12, 39
Kartramme	4, 6, 7, 16, 18-21, 24, 33, 44
Konstruksjonsbredde	13
Kystlinjer	4, 16, 18, 20, 39, 40

<u>Referanse</u>	<u>Side</u>
Linjer	5, 16, 17, 19, 29, 31, 36, 44, 49
Linjetyper	5, 23, 46, 47
Map-Library	3, 6, 20, 27, 40, 60
Meldinger	10
Målestokk	5, 12, 13
Segment	7, 15, 17, 30, 31, 33, 44
Sintran	38, 41, 45, 48- 50
Sintran-kommando	34
Skjermbilder	7
Stasjonsnett	6, 16, 17, 19, 40
Symboler	4-6, 16-18, 29, 33, 44, 51
Tekst	16, 29, 30, 44