CHALMERS
UNIVERSITY OF TECHNOLOGY

Avvikelsemätning av bergborrhål

En prototypimplementering med noggrannhetsanalys

Examensarbete inom civilingenjörsprogrammet

JONAS FORSSELL

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2015

Borehole deviation measurement
A prototype implementation with accuracy analysis
JONAS FORSSELL

© JONAS FORSSELL, 2015.

Department of Computer Science and Engineering
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone +46 (0)31 – 772 1000

Institutionen för data- och informationsteknik, Chalmers tekniska högskola

Göteborg 2015

Borehole deviation measurement
A prototype implementation with accuracy analysis
JONAS FORSSELL
Department of Computer Science and Engineering
Chalmers University of Technology

SUMMARY

The aim of my graduate project has been to evaluate the use of static accelerometers that measures static acceleration, and use of magnetometers for borehole deviation measurements. This has been done by evaluating the required accuracy for the measurement system, and building of a prototype that has been tested. The evaluation shows that it is hard to reach an accuracy that is better than ±0.1° for inclination and ±0.5° for azimuth, which were the requirements. The error in azimuth measurement is highest when the borehole is near vertical. This is explained by the offset errors in the sensors for the determination of the probe rotation. The evaluation shows that it is difficult to reach the accuracy requirements given the other constraints. In addition the probe dimension, environment, power supply and data transmission prove to be challenges.

The report is written in Swedish.

Keywords: borehole deviation, borehole logging, well logging.

Förord

Denna rapport utgör den skriftliga redovisningen av mitt examensarbete som ingår som en del i min utbildning på elektrotekniklinjen vid Chalmers tekniska högskola

Arbetet gjordes 1996 på uppdrag av Malå Geoscience, ett svenskt bolag som utvecklar och tillverkar instrument för att mäta geofysiska egenskaper i mark och berg.

För råd och hjälp vill jag tacka följande personer:

Dan Stenberg, Malå Geoscience
Bernth Johansson, Malå Geoscience

Stig-Göran Larsson, Chalmers Tekniska Högskola
Lars Svensson, Chalmers Tekniska Högskola

Dessutom vill jag tacka till mina vänner Roger Malmberg, Peter Vesmes och Björn Alving som granskat och korrekturläst.

Sammanfattning

Syftet med examensarbetet har varit att utvärdera möjligheten att använda accelerometrar som kan mäta statiska accelerationer och, använda magnetometrar för avvikelsemätning av bergborrhål. Detta har gjorts genom att analysera noggrannhetskraven för mätsystemet samt att bygga en prototyp som testats. Arbetet visar att det är svårt att nå en noggrannhet på ±0.1° för inklination och ±0.5° för azimut, vilka var kraven. Största felet i azimut uppstår när borrhålet är nästan vertikalt. Detta beror på att offsetfel i givarna för rotationsbestämningen av sonden får stor inverkan i detta fall. Utvärderingen visar att det är en utmaning att nå önskad noggrannhet givet övriga krav. Sedan kvarstår utmaningarna med sondens dimension, miljö, strömförsörjning och dataöverföring.

Innehåll

1 Bakgrund 8

1.1 Bergborrning 8

1.2 Riktningsavvikelse vid bergborrning 9

1.2.1 Begreppen azimut och inklination 9

1.2.2 Avvikelse hos bergborrhål 10

1.2.3 Vanliga metoder för avvikelsemätning 11

2 Inledning 13

2.1 Litteraturstudie 13

2.2 Beskrivning av företagets nuvarande mätare Boremac D2 14

2.3 Krav och förutsättningar för det nya instrumentet 15

2.4 Bestämning av inklination 16

2.5 Bestämning av azimut 16

2.6 Jordens magnetfält 17

2.7 Beräkning av inklination och azimut 18

2.8 Beräkning av borrhålets sträckning 20

3 Beskrivning av mätsystemet 22

3.1 Sonden 24

3.1.1 Accelerometrar 27

3.1.2 Magnetometrar 29

3.1.3 Mätkort 29

3.1.4 Mikroprocessorkort 31

3.1.5 Kraftkort 32

3.1.6 Sondprogram 33

3.2 Markenhet 34

3.2.1 Kommunikationselektronik 36

3.2.2 Strömförsörjning 36

3.2.3 Dator 36

3.2.4 PC program 36

4 Simulering av systemets mätnoggrannhet 37

4.1 Analys av mätfel i elektroniken 37

4.2 Analys av mätfel i givare 38

5 Test och utvärdering 43

5.1 Test i fixtur 43

5.1.1 Test av inklinationsmätning 43
5.2 Test i borrhål 44
5.2.1 Jämförelse mellan mätning upp och ner i borrhålet 44
5.2.2 Jämförande mätning med Boremac D2 44
6 Slutsats 45
7 Förbättringsförslag 46
8 Använda förkortningar och begrepp 47
8.1 Förkortningar 47
8.2 Ordlista 47
9 Referenser 48
10 Appendix A: Härledning av inklination och azimut 49
10.1 Härledning av Inklinationen φ 49
10.2 Härledning av azimut θ 50
11 Appendix B: Toleransanalys av elektronik 53
12 Appendix C: Toleransanalys av givare 56
12.1 Matlabscript för beräkning 56
12.2 Matlabscript för presentation 59
13 Appendix D: Ritningar på sondens mekanik 61
14 Appendix E: Ritningar på sondens elektronik 62
14.1 Inkopplingsschema sond 62
14.2 Schema datorkort 63
14.3 Komponentlista datorkort 65
14.4 Schema mätkort 66
14.5 Komponentlista mätkort 67
14.6 Schema DC/DC omvandlare 68
14.7 Komponentlista DC/DC omvandlare 69
15 Appendix F: Ritningar på markenhetens elektronik 70
15.1 Inkoppling av markenhet 70
15.2 Schema markenhet 71
15.3 Komponentlista markenhet 72
16 Appendix G: Sondprogram 73
16.1 MAIN.C 73
16.2 SIO0.C 75
16.3 AD77.C 77
16.4 COM.C 78

16.5 ADASM.S03 78
16.6 SIO0.H 80
16.7 AD77.H 81
16.8 DEFDEV.H 81
16.9 COM.H 82
16.10 IO552.H 83
16.11 ADASM.H 86
17 Appendix H: PC program 88
17.1 MAIN.C 88
17.2 SERIO.C 91
17.3 SERIO.H 93
18 Appendix I: Mätningar 94
18.1 Mätning av inklination i fixtur 94
18.2 Mätning av azimut i fixtur 95
18.3 Mätning av borrhål i utvecklingslabb 96
18.4 Jämförande mätning med en tidigare Boremac D2 mätning. 98

1 Bakgrund

Malå Geoscience utvecklar och tillverkar instrument för att mäta geofysiska egenskaper i mark och speciellt berg. Företaget gör instrument både för undersökning från markytan och i bergborrhål, bland annat ett instrument för avvikelsemätning av bergborrhål

Avvikelsemätning görs för att ta reda på hur ett bergborrhål har avvikit från den ansatta borriktningen, dvs. om borrhålet har ändrat inklination (lutning) eller har avvikit i azimut (kompassriktning). Det nuvarande instrumentet är uppbyggt med analog elektronik och har inget datorgränssnitt. Dessutom innehåller instrumentet komplicerad och känslig finmekanik.

I ett led att datorisera mätningarna i borrhålen vill Malå Geoscience utveckla ett nytt datoriserat instrument för avvikelsemätning. Dessutom vill företaget göra instrumentet mer robust och därför frångå lösningen med de finmekaniska komponenterna, vilka ersätts med accelerometrar och magnetometrar.

Syftet med examensarbetet har varit att utvärdera möjligheten att använda accelerometrar, som kan mäta statiska accelerationer, och magnetometrar, för avvikelsemätning av bergborrhål.

1.1 Bergborrning

För att öka förståelsen av denna rapport presenteras i detta avsnitt en kort redovisning av terminologin inom bergborrning.

Det finns en rad olika metoder för att borra bergborrhål. De metoder som används mest är:

– Hammarborrning: den vanligaste borrmetoden som används i de flesta förekommande bergarter. Principen går ut på att slå på borrsträngen och mellan slagen vrida en hårdmetallförsedd borrkrona som spräcker lös flisor ur berget.

– Roterande, krossande borrning: används vid oljeborrning och spränghålsborrning. Principen går ut på att använda en roterande borrkrona som består av tre tandade koniska rullar med hårdmetallstift. Eftersom borröret har en hög tyngd spräcks flisor av berget bort av hårdmetalltänderna som pressas in i berget.

– Roterande, skärande borrning: används främst i lösa bergarter. Principen är densamma som vid borrning i metall och trä. Här nyttjas en eggförsedd borrkrona som skjuvar bort materialet ur berget.

– Roterande, nötande borrning (eller diamantborrning): används främst vid prospekteringsborrning, då det är önskvärt att ta kärnprover ur berget. Borrmetoden går ut på att en diamantbeklädd ringborrkrona används. I denna metod utnyttjas ett högre varvtal på borrstången och lägre matartryck än vid de andra metoderna. Kärnprovet som tas ur berget pressas under borrningen in i borröret. Denna typ av bergborrhål borras vanligtvis med en diameter på 36 - 146 mm. Djupet på borrhålen varierar mellan ett tiotal meter upp till flera tusen meter.

Vid bergborrning är det ofta nödvändigt att borra genom ett täckande jordlager innan berget nås. För att borrhålet inte skall rasa samman är det nödvändigt att stabilisera borrhålets väggar. Oftast görs detta med ett rör av stål eller plast. När ett stålrör används kan detta påverka mätningen av azimut, om en metod används som bygger på jordens magnetfält.

För mer information om bergborrning hänvisas till [1].

1.2 Riktningsavvikelse vid bergborrning

1.2.1 Begreppen azimut och inklination

Med inklination menas i rapporten borrhålets eller en vektors lutning mot jordens horisontalplan (jordens tyngdkrafts normalplan), se bild 1-1. Med 0° menas horisontellt borrhål eller vektor och med 90° menas lodrät neråt mot jordens mitt. Följaktligen blir -90° alltså rakt upp. Inklination är definierad i intervallet -90° till 90°

Med azimut menas i rapporten borrhålets eller en vektors riktning eller kurs i geografisk mening i jordens horisontalplan (jordens tyngdkrafts normalplan), se bild 1-1. Där 0° är norrut, 90° är österut, 270° är västerut och 360° tillbaka norrut. Azimut är definierad i intervallet 0° till 360°

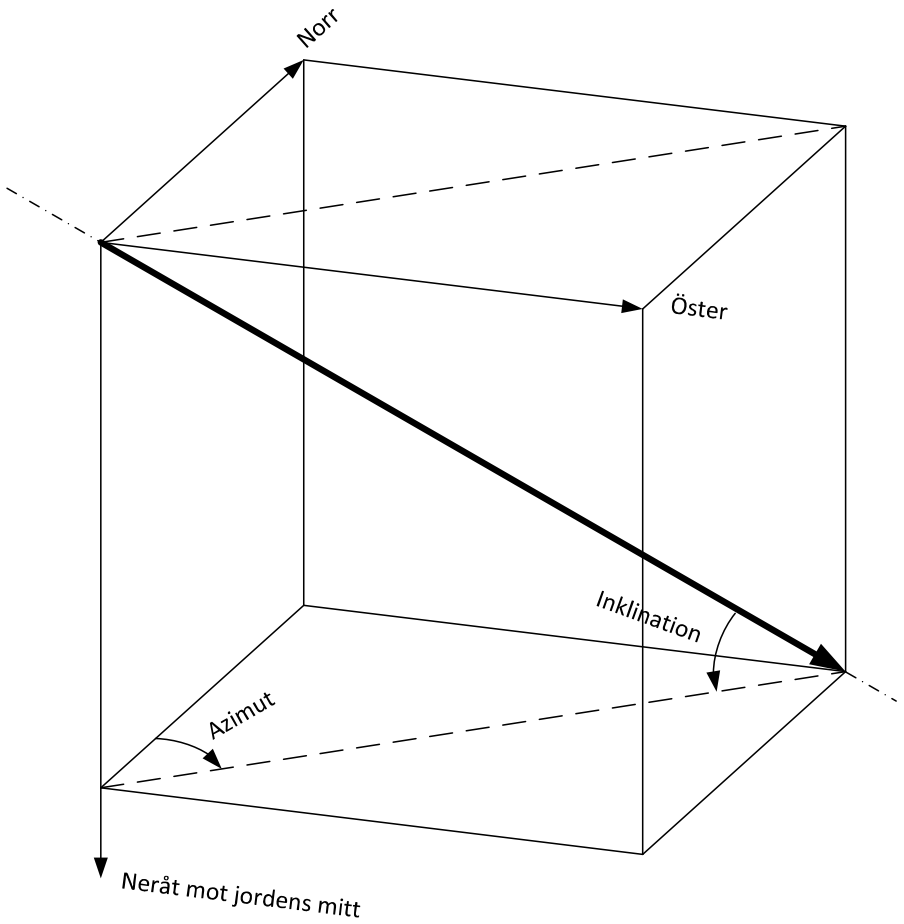


Bild 1-1. Azimut och inklination.

1.2.2 Avvikelse hos bergborrhål

Vid borrning är det vanligt att ha ett förutbestämt målområde som det är önskvärt att nå. Ett problem är att bergborrhål ofta avviker från den önskade riktningen. Genom att avvikelsemäta riktningen på bergborrhålet är det möjligt att avgöra huruvida borrningen skall påbörjas på nytt eller om ”styrkilar” skall sättas in i borrhålet för att på så sätt styra borren i rätt riktning. De viktigaste faktorerna som påverkar borrhålets riktningsavvikelse är:

- Bergets egenskaper.
- Hur stor spalten är mellan borrör och borrhålsvägg.
- Diametern på borrhålet.
- Borrkronans skärförmåga.
- Matningskraft på borrkronan.

Avvikelserna från den önskade riktningen kan bli väldigt stora. Kravet på noggrannhet är naturligtvis beroende på det aktuella uppdraget. Områden där avvikelsemätning är av intresse är t.ex. vid prospekteringar, markundersökningar, sprängarbeten och förstärkningsarbeten. Vid sprängarbeten är det möjligt att förutse sprängningsarbetets effekt, om spränghålens orienteringar är kända, och på så sätt är det möjligt att spara in sprängämne. Vid prospektering är det viktigt att med hög noggrannhet veta malmkroppars orientering och position före eventuell brytning påbörjas. Det finns exempel på misslyckade gruvoperationer där en påvisad fyndighet helt har missats. Det har också hänt att hål har avvikit in i järnvägstunnlar [4]. I de flesta fall är det lätt motivera kostnaden för en avvikelsemätning, och i vissa fall är det nödvändigt. Se bild 1-2 för en illustration av problemet.

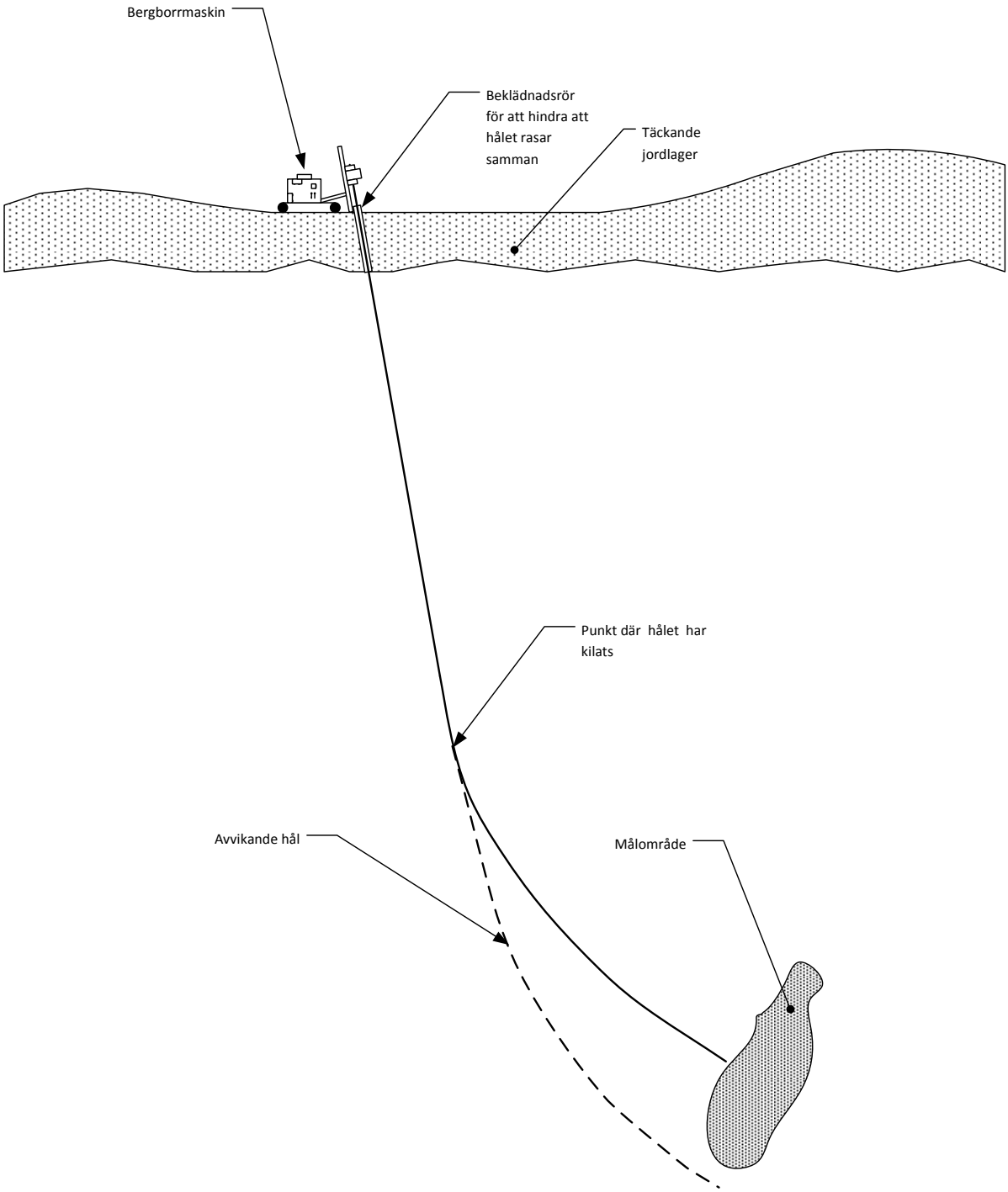


Bild 1-2. Illustration av ett avvikande bergborrhål, där kilning gjorts där streckade linjen startar.

1.2.3 Vanliga metoder för avvikelsemätning

Det finns olika metoder för att avvikelsemäta bergborrhål. Här kommer en kort orientering av tre olika metoder. Dessa bygger på att det i borrhålet mäts på ekvidistanta punkter med hjälp av en sond som förs ner i borrhålet. Se bild 1-3 för ett exempel på mätsystem.

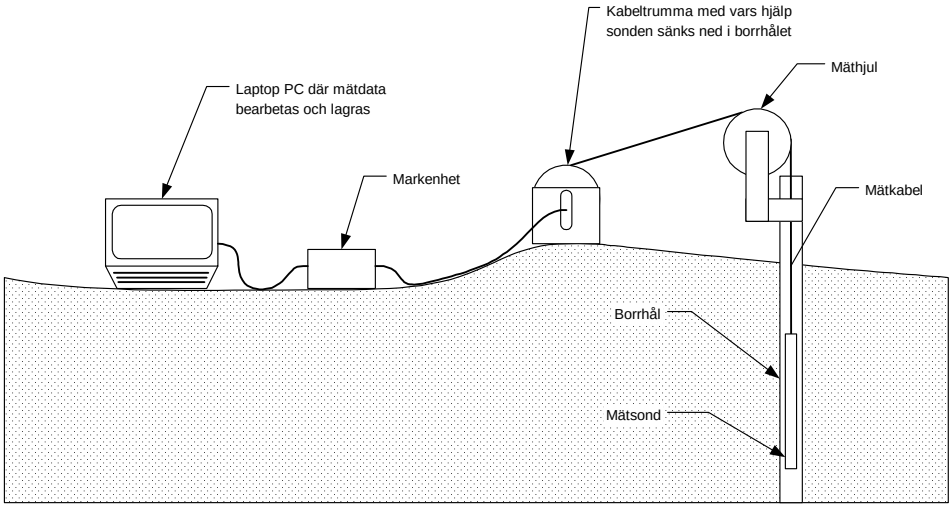


Bild 1-3. Mätsystem för avvikelsemätning.

Det finns relativa och absoluta metoder [4]. De relativa metoderna bygger på att borrhålets azimut och inklination i början på borrhålet är kända. Dessa kan mätas in med kompass och inklinometer innan mätningen av borrhålet påbörjas. Sedan mäts förändringen av borrhålets riktning mellan mätpunkterna. Absoluta metoder bygger på att borrhålets absoluta inklination och absoluta azimut mäts i varje punkt, utan att använda data från andra mätpunkter.

Den relativa metod som används mest är att mäta sondens böjning mellan punkter i borrhålet. En metod som används är att observera två ringar som är monterade inne i sonden med ett fast och känt avstånd. Ringarnas läge i förhållande till varandra mäts in med en kamera som sitter längst bak i sonden. När en punkt mätts in sänks sonden ned en sträcka som är lika lång som avståndet mellan ringarna. På så sätt hamnar den bakre ringen på den främre ringens gamla position. För att veta åt vilket håll sonden är böjd är det nödvändigt att veta hur sonden är roterad i borrhålet. Detta görs genom att samtidigt som en observation av ringarna sker så observeras också ett ringformat vattenpass, med t ex en kamera. Denna metod ger väldigt bra resultat om någon form av centreringsanordning av sonden nyttjas, så att den alltid ligger centrerad i borrhålet.

De absoluta metoderna går ut på att en mätning av borrhålets inklination (lutning) sker med någon typ av inklinometer. Men om endast borrhålets inklination är känd så är det dock inte möjligt att veta i vilken azimut (riktning) borrhålet avviker.

Den mest uppenbara metoden är att mäta azimut med hjälp av en kompass. Metoder som bygger på att mäta jordens magnetfält för att bestämma azimut är känslig för störningar i magnetfältet. Dessa störningar kan bero på förekomsten av magnetiska malmarter eller på att borrhålet har klätts med stålrör för att förhindra ras. Den andra metoden att bestämma azimut är med hjälp av gyron. Gyron har dock problem med drifter och med att de måste vara små gyron, som får plats i ett smalt borrhål, blir problemen större.

En annan absolut metod är att föra ner sonden med hjälp av stänger i borrhålet så att sonden inte kan rotera i borrhålet, och på så vis förs azimutinformationen ner till sonden. Problemet med stänger är att dessa måste skarvas med någon typ av bajonettlåsning eller gångjärn. Då dessa leder inte är helt glappfria kommer mätningen att försämras ju längre borrhålet är.

2 Inledning

2.1 Litteraturstudie

Handbok i prospekteringsborrning [1]:
En bok som beskriver hur prospekteringsborrning genomförs i detalj. I kapitel 14 ”Mätningar i bergborrhål”, sid 123 - 127, beskrivs olika metoder för avvikelsemätning av bergborrhål. Här finns Malå Geoscience Boremac D2 system beskrivet med noggrannhetsdata. Här nämns att Boremac D2 har en inklinationsmätning med noggrannhet som är bättre än 0.5°, och en azimutmätning som är bättre än ±1°, under förutsättning att borrhålet lutar mer än 2° mot vertikalplanet (nästan lodrätt hål). I övrigt har information från övriga delar av denna bok använts till kapitel 1.

Well logging for physical properties [2]:
En bok som beskriver olika mätmetoder för att mäta fysikaliska egenskaper i bergborrhål. I delkapitel 17-1 ”Directional surveying”, sid 503 – 513, beskrivs metoder för att räkna ut borrhålets sträckning med olika metoder. Här har den enklaste metoden, formeln 17-1, använts i detta arbete. Beräkning av borrhålets sträckning ingår inte i arbetet men använts för att jämföra mätdata med tidigare mätningar. Notera att inklinationen φ har en annan definition i denna rapport ($90 - \varphi$). Delkapitel 17-2 beskriver olika typer av instrument för avvikelsemätning. Här nämns användning av fluxgate magnetometrar och accelerometrar. Delkapitel 17-3 beskriver analys av mätfel och hur detta påverkar noggrannheten i bestämning av borrhålets slutposition. Denna typ av analyser ligger utanför detta arbete och lämnas till den intresserade läsaren.

Borehole surveying – A systematic approach to attaining quality data [3]:
Denna artikel förklarar nyttan av avvikelsemätning, vikten av hög kvalitet på data och när olika typer av mätmetoder kan användas. Här nämns användning av accelerometrar och magnetometrar, kallat DEMS (Down Hole Electronic Multi Shot). Noggrannhet på $\pm 0.1^{o}$ för inklination och azimut hävdas, dock nämns inte hur eller vilken tillverkare som uppnår detta.

Where in earth are we? [4]:
Även denna artikel går igenom olika mätmetoder för avvikelsemätning samt nyttan av att kartlägga avvikelsen. Här nämns vikten av att använda rätt mätmetod vid olika typer och hål. Problematiken runt magnetiska avvikelser som stör azimutmätningen med magnetometer tas upp. I denna artikel införs begreppet relativ och absolut mätmetodik som beskrivs i kapitel 1.2.3.

2.2 Beskrivning av företagets nuvarande mätare Boremac D2

Malå Geoscience nuvarande avvikelsemätare bygger på att inklinationen mäts i borrhålet med ett lod utformat som ett hjul. Se bild 2-1 för en principiell skiss av sonden. På detta hjul finns en resistiv bana som används för att bestämma inklinationen hos borrhålet elektriskt. Vid mätögonblicket låses lodhjulet med hjälp av en arm som samtidigt är en kontakt med vars hjälp resistansen mäts. Azimut bestäms av en kompass som är upphängd så att den alltid hänger horisontellt i en gimbal. På samma sätt som när inklinationen mäts så låses kompassen fast och på kompassen finns också en resistansbana som här används för att bestämma azimut. Inklinationsmätaren och kompassen måste hängas upp så att dessa kan rotera runt fritt runt sondens Z-axel, detta har lösts genom att inne i sonden ha en vagga som alltid är orienterad så att lodhjulet alltid är rätt orienterat, se bild 2-1.

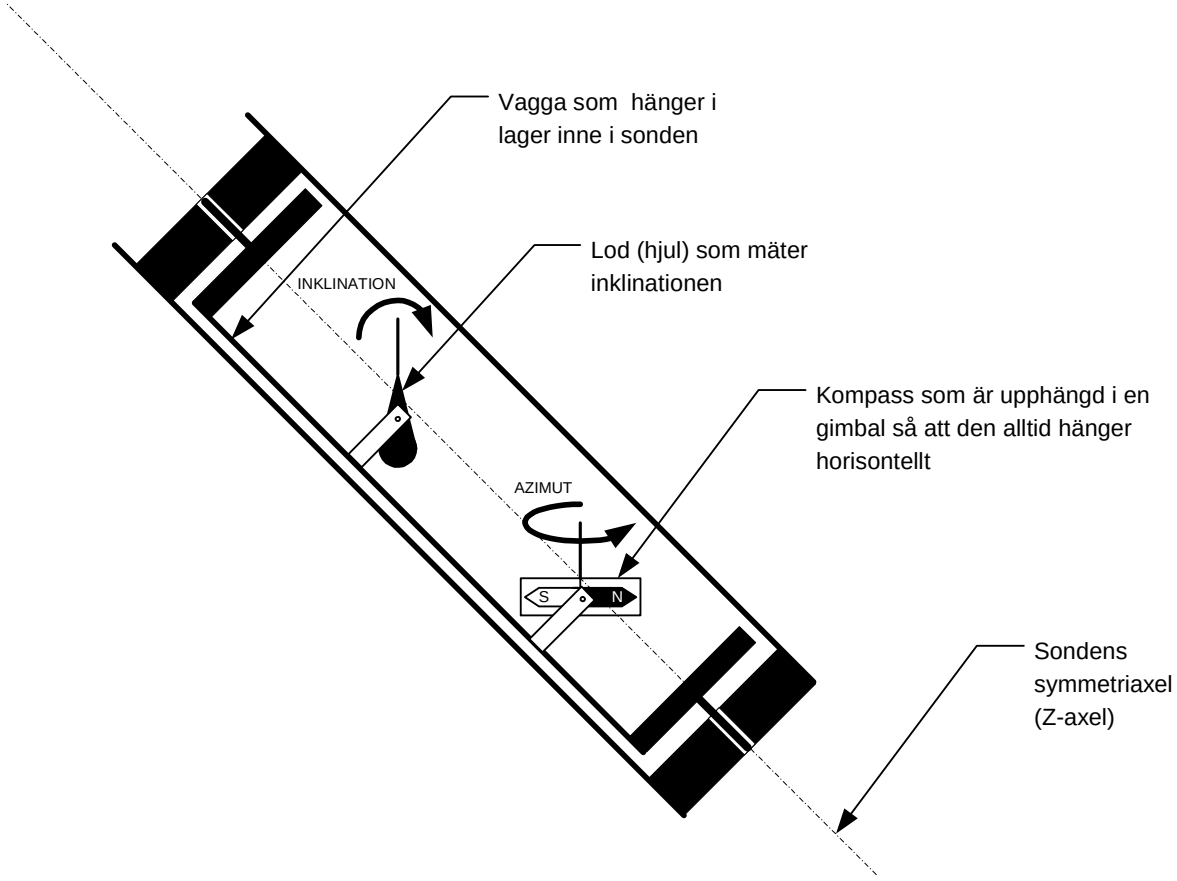


Bild 2-1. Genomskärning av en Boremac D2 sond.

2.3 Krav och förutsättningar för det nya instrumentet

Den här rapporten beskriver arbetet med att ta fram en efterföljare till Boremac D2. Mätmetoden för det nya instrumentet är samma som för det gamla, det vill säga en absolut metod som bygger på att bestämma azimut genom att mäta jordens magnetfält. Malå Geoscience har satt upp följande krav för hur det nya instrumentet skall byggas:

1. Instrumentet skall kunna mäta inklinationen med en noggrannhet bättre än ±0,1°
2. Instrumentet skall kunna mäta azimut med en noggrannhet bättre än ±0,5°.
3. För bestämning av inklinationen skall en treaxlig accelerometer som kan mäta statiska accelerationer användas.
4. För bestämning av azimut skall en treaxlig fluxgate magnetometer användas.
5. Instrumentet skall kunna användas i hål som är upp till 1500 meter djupa med en diameter som är minst 46 millimeter i diameter.
6. Det lilla datorkortet som Malå Geoscience har utvecklat för mätsonder skall användas.

Det arbete som skall utföras består av följande huvudpunkter:

1. Utvärdering och presentation av de matematiska sambanden för beräkning av inklination och azimut, se kapitel 2.7.
2. Noggrannhetsanalys av mätsystemet, se kapitel 4.
3. Systemdesign av mätsystemet, se kapitel 3.
4. Konstruktion av sondens mekanik, se kapitel 3.1.
5. Konstruktion av sondens elektronik, se kapitel 3.1.3, 3.1.4 och 3.1.5.
6. Programmering av mikroprocessorn i sonden, se kapitel 3.1.6.
7. Konstruktion av markenhetens elektronik, se kapitel 3.2.1 och 3.2.2.
8. Programmering av PC för mätvärdesinsamling och analys, se kapitel 3.2.4.
9. Test och utvärdering av hela mätsystemet, se kapitel 5.

2.4 Bestämning av inklination

För att bestämma inklinationen eller lutningen hos ett föremål går det att använda sig av någon form av lod eller vattenpass. Jordens tyngdkraft påverkar då lodet eller vätskan i libellen.

I det nya instrumentet har Malå Geoscience bestämt att detta skall göras med statiska accelerometrar. Dessa accelerometrar kan mäta accelerationer som är konstanta. De kommer att ge en utsignal på grund av jordens tyngdkraft vilket är nödvändigt för denna mätning. Eftersom mätmetoden bygger på att sonden är stilla när mätningen sker så är jordens tyngdkraft den enda kraften som kommer att mätas.

För detta används tre stycken enaxliga accelerometrar som endast är känsliga i en riktning och dessa monteras ortogonalt mot varandra i sonden. På så sätt är det möjligt att mäta både sondens inklination i borrhålet och hur sonden är roterad i borrhålet.

Jordens tyngdkraft är ganska konstant över jordytan och är ungefär 9,81 m/s^2 eller 1 g. Det spelar ingen större roll om det finns mindre variationer i jordens tyngdkraft eftersom den totala tyngdkraften mäts och alla mätningar normeras mot den totala tyngdkraftenen.

2.5 Bestämning av azimut

För att bestämma azimut kan jordens magnetfält utnyttjas. Den vanligaste tillämpningen är en vanlig kompass där nålen ställer sig parallell med horisontalkomposanten hos jordens magnetfält.

I det nya instrumentet har Malå Geoscience bestämt att detta skall göras med en treaxlig fluxgatemagnetometer. Denna typ av magnetometer kan mäta statiska magnetfält i tre ortogonala riktningar.

2.6 Jordens magnetfält

Det är viktigt att skilja på magnetisk nordpol och geografisk nordpol. Dessa är inte belägna på samma plats. Den skillnad i vinkel som fås i riktning mellan en avläsning på en kompass och riktningen mot den geografiska nordpolen kallas deklination (inte att förväxla med azimut). För Sverige är deklinationen idag mellan +5° och -5°. Andra egenskaper hos jordmagnetiska fältet är dess styrka som ligger någonstans runt 50 till 60 µT. Inklinationen hos magnetfältet (inte att förväxla med inklinationen hos borrhålet). Den ligger mellan 60° och 80°. Dessa siffror kan anses vara representativa för Sverige [5]. Se tabell 2-1 där informationen har sammanställts.

Parameter	Förhållanden för Sverige	Förhållanden för övriga världen
Jordmagnetiska fältets styrka	50 – 55 µT	24 – 60 µT
Jordmagnetiska fältets inklination	60 – 80°	0 – 90°
Jordmagnetiska fältets deklination	-5 – 5°	-60 – 60°

Tabell 2-1. Jordmagnetiska fältets egenskaper, data från 1995 [5].

I bild 2-2 visas det jordmagnetiska fältet som representeras av vektorn F. Den del av fältet som ligger i horisontalplanet kallas H. Vinkeln D i figuren är deklinationsvinkeln och I är inklinationsvinkeln. I figuren är också ett jordfixt koordinatsystem definierat där X-axeln representerar den geografiska norr riktningen, Y-axeln den geografiska öst riktningen samt Z-axeln mot jordens mittpunkt.

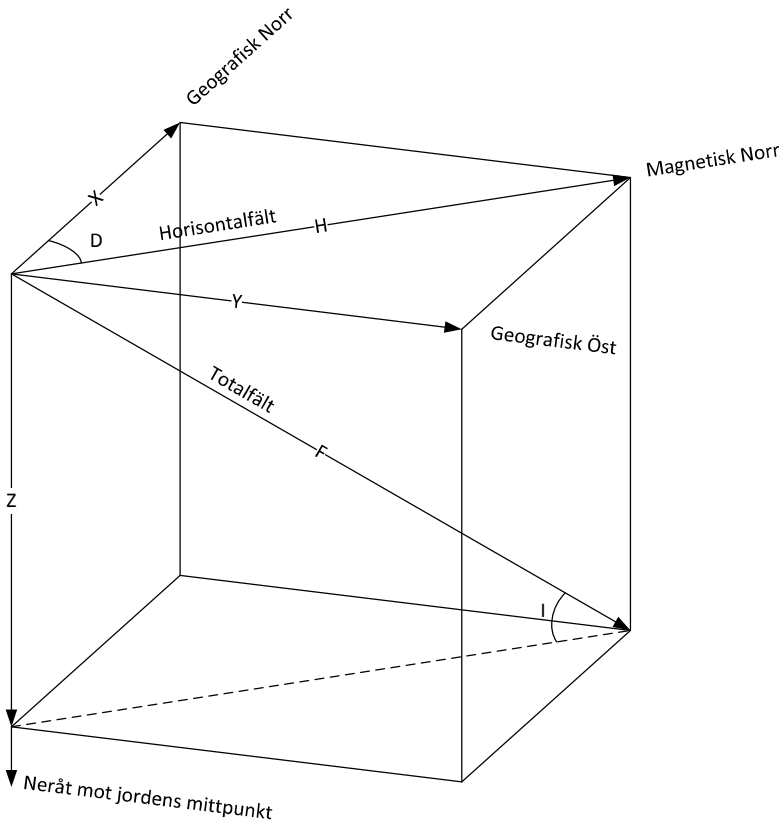


Bild 2-2. Illustration av jordens magnetfält. Inte att förväxla med borrhålets riktning.

2.7 Beräkning av inklination och azimut

För att beskriva hur sonden i det nya mätsystemet mäter jordens tyngdkraft och magnetfält definieras först ett koordinatsystem för sonden, se bild 2-3.

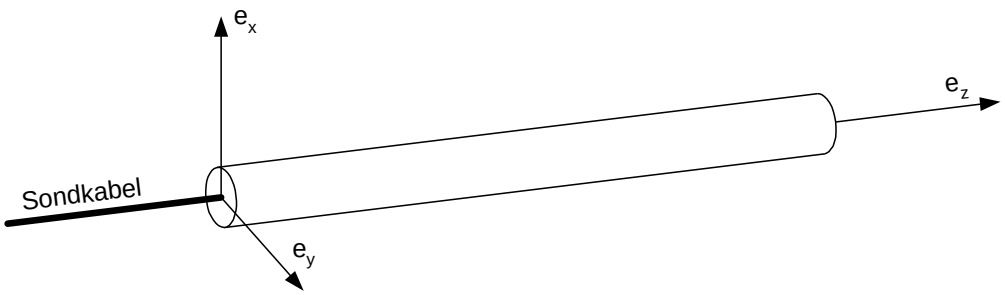


Bild 2-3. Sondens koordinatsystem.

Längs sonden finns Z-axeln, och positiv riktning är längs sonden neråt. Om sonden hängs i sin kabel så kommer sondens Z-axel att peka neråt. Sedan definieras X och Y axlarna så att axlarna bildar ett högerorienterat ortogonalt koordinatsystem. För enkelhetens skull bildas tre enhetsvektorer e_x, e_y och e_z som är orienterade i sondens koordinatsystem. Då fås:

B_x – Magnetfältet i sondens X-axel
B_y – Magnetfältet i sondens Y-axel
B_z – Magnetfältet i sondens Z-axel

Den sammansatta vektorn kallas för B-vektorn som definieras av:

$$B = \begin{bmatrix} B_x & B_y & B_z \end{bmatrix} \bullet \begin{bmatrix} e_x & e_y & e_z \end{bmatrix}$$

G_x – Tyngdkraften i sondens X-axel
G_y – Tyngdkraften i sondens Y-axel
G_z – Tyngdkraften i sondens Z-axel

Här definieras även en sammansatt vektor, G-vektorn som:

$$G = \begin{bmatrix} G_x & G_y & G_z \end{bmatrix} \bullet \begin{bmatrix} e_x & e_y & e_z \end{bmatrix}$$

Ur tyngdkraftens uppmätta komposanter beräknas sedan sondens inklination φ, se bild 2-4, som:

$$\varphi = \arctan\left[\frac{Gz}{\sqrt{Gx^2 + Gy^2}}\right]$$

Formel 2-1. Beräkning av inklination.

Resultatet ligger i intervallet -90° till 90° och om φ är negativ så lutar sonden uppåt. För härledning av formel 2-1 se appendix A.

Azimutvinkeln θ (se bild 2-4) beräknas på följande vis ur de uppmätta komposanterna för tyngdkraften och magnetfältet:

$$\theta = -\arctan 2\left[\frac{(B_x G_y - B_y G_x)\sqrt{G_x^2 + G_y^2 + G_z^2}}{B_z(G_x^2 + G_y^2) - G_z(B_x G_x + B_y G_y)}\right]$$

Formel 2-2. Beräkning av azimut.

När vinkeln θ bestäms så uppstår ett problem eftersom arctan funktionen endast är bestämd i intervallet -90° till +90°, därför används arctan2 funktionen (som finns i programspråket C) som är definierad i intervallet -180° till 180°. Eftersom azimutvinkeln ligger i intervallet 0 – 360° så får 360° läggas till när resultatet blir negativt. Notera här att både jordens magnetfält och tyngdkraft spelar in på sondens azimut. Detta kan förstås genom en tänkt projektion av den uppmätta B-vektorn i jordens koordinatsystem i det plan som spänns upp av X och Y-axeln, se bild 2-2. För att kunna göra detta måste kännedom om både sondens inklination och rotation i borrhålet finnas och dessa kan räknas ut med hjälp av information från accelerometrarna. För härledning av formel 2-2 se appendix A.

2.8 Beräkning av borrhålets sträckning

För att sedan kunna beräkna borrhålets sträckning i berget måste kännedom finnas om i vilka punkter längs borrhålet som inklination och azimut har bestämts och utgångspositionen. Här har utgångspositionen valts att anges med hjälp av koordinaterna X_0, Y_0 och Z_0, som motsvaras av borrhålets första mätpunkt i bild 2-4 (då n=0). För att kunna bestämma positionen i borrhålet mäts förflyttningen längs borrhålets sträckning. Denna information fås genom att kabelns längd mäts med ett mäthjul som i bild 1-3, där denna förflyttning mellan två punkter är definierad som dl. Sedan kan förflyttningen i lodrät, nordlig och östlig riktning beräknas. I bild 2-4 är förflyttningen i östlig riktning markerad med dy och förflyttningen i nordlig riktning med dx.

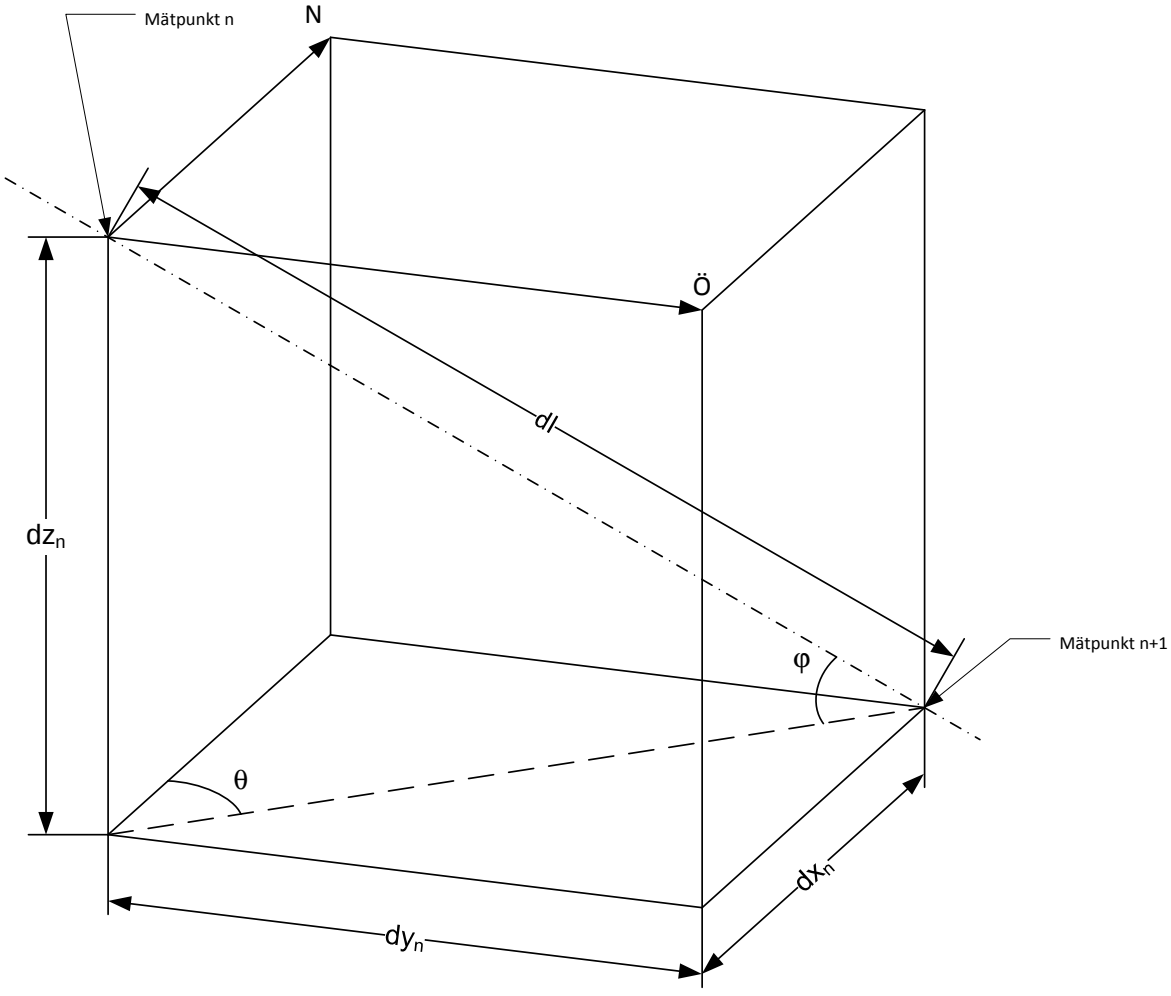


Bild 2-4. Koordinatsystem för borrhål.

Sedan beräknas med enkla trigonometriska beräkningar borrhålets sträckning.

$$dx_n = \cos(\varphi_n)\cos(\theta_n)dl$$

$$dy_n = \cos(\varphi_n)\sin(\theta_n)dl$$

$$dz_n = \sin(\varphi_n)dl$$

Borrhålet approximeras med en rät linje mellan mätpunkt n och mätpunkt n+1; nu kan borrhålets position i varje punkt beräknas som:

$$X_p = X_0 + \sum_{n=1}^{p-1} dx_n \, , p \geq 2$$

$$Y_p = Y_0 + \sum_{n=1}^{p-1} dy_n \, , p \geq 2$$

$$Z_p = Z_0 + \sum_{n=1}^{p-1} dz_n \, , p \geq 2$$

Där p är antalet mätpunkter fram till den önskade positionen. För mer information om ovanstående metod och andra metoder hänvisas till [2]. Observera att definitionen på inklinationen är 90 - φ i [2].

3 Beskrivning av mätsystemet

Det finns två väsentliga problem när det gäller att samla in mätdata från bergborrhål med elektroniska metoder. Dessa är strömförsörjning och mätdatainsamling. Sonden kan strömförsörjas antingen från markytan eller från batterier i sonden. Mätdatainsamlingen kan antingen göras i sonden eller överföras till markytan via en kabel.

Om varianten med batteriförsörjning i sonden väljs så lagras oftast data i sonden för att på så sätt slippa ifrån användning av kabel helt. I detta fall används en lina som sonden sänks ner med. Punkterna mäts då med jämna tidsintervall till exempel varje minut. Varje minut stoppas nedfirningen av sonden på de mätpunkter som skall mätas. När mätningen sedan är klar firas sonden upp och töms på mätdata. Nackdelarna med denna metod är användningen av batterier som är problematisk. Dessutom kan inte mätdata studeras direkt, något som är önskvärt att göra för att försäkra sig om att mätningen är riktig.

I det här instrumentet valdes att använda metoden med kommunikation till markytan och även strömförsörjning från markytan. Den kabel som befanns vara lämplig att använda var en femledarkabel som under en längre tid har använts av Malå Geoscience för borrhålsmätning. Kabeln är uppbyggd av fem ledare och fyra kevlartrådar som används för dragavlastning. Kabeln är klädd med polyuretan vilket gör att den har en densitet som är nära vatten. Detta gör att kabeln inte förlängs av sin egentyngd. Eftersom kabeln har en resistans på 90 Ω/km per ledare ger detta förluster när sonden skall strömförsörjas från markytan.

Det antogs att sonden kunde dra 10 W vilket är väl tilltaget. Försök har sedan tidigare gjorts med att använda denna kabel med en matningsspänning på 200 VDC. Detta gör att sonden kan dra en effekt på maximalt 37 watt med en kabellängd på 1500 meter. Detta ges av att man vid effektanpassning får effektmaximum vid sonden, se formel 3-1.

$$P_{\max} = \frac{(200V/2)^2}{90\Omega \cdot 2 \cdot 1{,}5} \approx 37W$$

Formel 3-1. Effektmaximum vid 200 VDC matning.

Tre ledare i kabeln återstår och på dessa måste kommunikationen med sonden ske. De metoder som står till buds vid så pass långa kablar är via modem eller via någon differentiell metod. En metod som Malå Geoscience tidigare framgångsrikt använt och som är enkel är RS-485. Problemet är att RS-485 behöver två ledningar i vardera kommunikationsriktningen och en jordledare om full duplex används. Genom att använda halv-duplex räcker de återstående tre ledarna eftersom inget behov av dubbelriktad kommunikation finns.

Pc-datorn kontrollerar kommunikationsriktningen på drivkretsen i markenheten och sonddatorn kontrollerar kommunikationsriktningen på drivkretsen i sonden. När ingen kommunikation sker lyssnar båda datorerna. När en mätning skall ske vänder markdatorn kommunikationsriktningen på sin kommunikationskrets och skickar ett kommando till sonden att den skall mäta. Därefter vänder markdatorn sin kommunikation till lyssning. När sonden har mätt färdigt vänder den sin kommunikationsriktning till sändning. Därefter skickar den data till markdatorn. När sändningen är klar vänder sonden sin kommunikationsriktning till mottagning och är då klar för nästa mätning. Se bild 3-1 för det kompletta systemet.

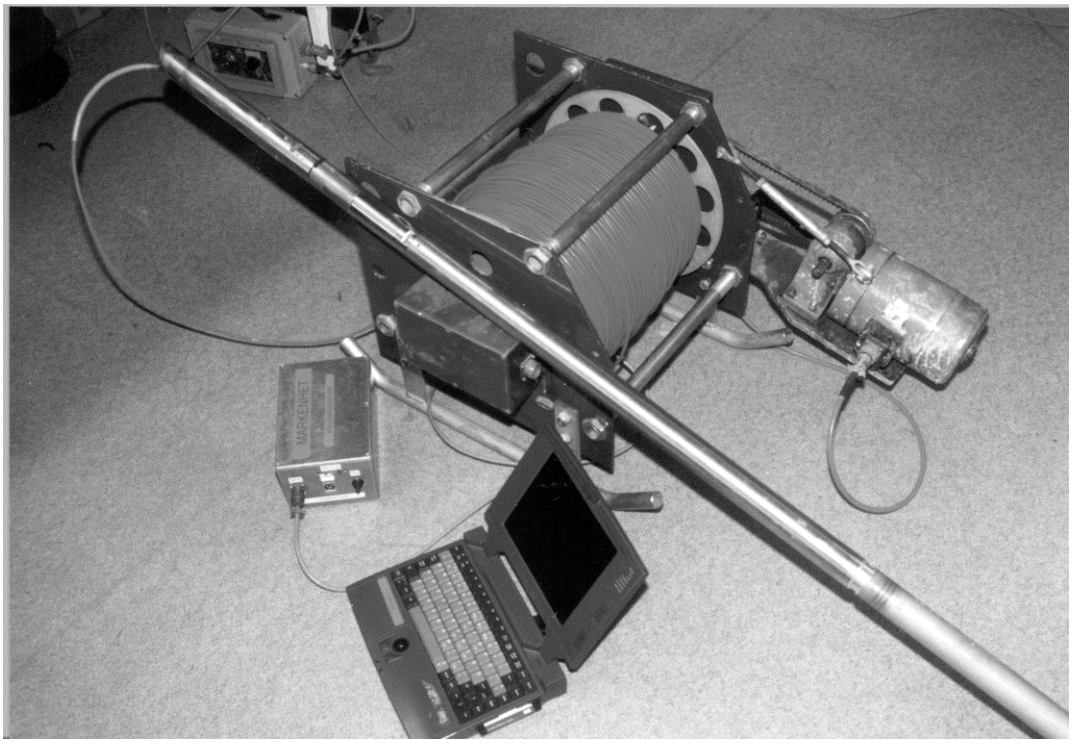


Bild 3-1. Det kompletta systemet.

3.1 Sonden

Sonden är den del av mätsystemet som sänks eller skjuts in i borrhålet och som har de största miljöpåkänningarna. De krav som ställs på denna är bland annat fysiska dimensioner. Borrhålet som kan borras har en minsta diameter (i det här fallet 46 mm) vilket gör att sonden måste ha en diameter mindre än detta. Malå Geoscience har tidigare utvecklat sonder för denna dimension på hål och använt sig av en ytterdiameter på 42 mm på sonderna. Denna dimension kommer även att användas för denna sond.

På grund av att sonden oftast sänks ner i grundvattenfyllda bergborrhål så kommer den att utsättas för extrema tryckpåkänningar. I detta fall är trycket ungefär 15 MPa vilket är det tryck som en vattenpelare på 1500 meter ger. Därför måste sonden vara tät och klara detta vattentryck. Detta gäller även kontakter och kablar. Eftersom Malå Geoscience har konstruerat och tillverkat sonder som klarar höga vattentryck så kommer inte det problemet att tas upp här. Grunderna för sonden finns som färdiga konstruktioner, dock är accelerometerhållaren och magnetometerhållaren unika för den nya sonden och konstruerades inom ramen för detta arbete.

Sonden måste byggas upp av ickemagnetiska material så långt som möjligt eftersom magnetiska material kan påverka magnetometern som bestämmer azimut. Detta gäller då speciellt den del av sonden som innehåller magnetometern. Därför har aluminium, plast, mässing och glasfiber valts till de nedre delarna av sonden, se bild 3-3. Till de övre delarna var det tvunget att använda rostfria rör för att få plats med accelerometrar och elektronik. Det mest logiska var då att montera magnetometern längst ner i sonden och där bygga sonden så omagnetisk som möjligt, se bild 3-4.

Erfarenheten visar att längden på sonden bör överstiga en meter för att få en sond som inte lätt fastnar i borrhålet. I detta fall valdes en sondlängd på 1700 mm för att inte få en alltför ohanterlig längd för transporter (bör kunna transporteras på taket av en bil).

Vidare krävs också att mekaniken är stabil och rak för att sondens mätprecision skall kunna garanteras.

Sonden skall kommunicera med och strömförsörjas från markytan. I sonden skall mätvärdesinsamling och strömförsörjning av givarna ske. För detta behövs kraftkort, mätkort och ett datorkort. Se bild 3-2 för ett blockschema över sonden.

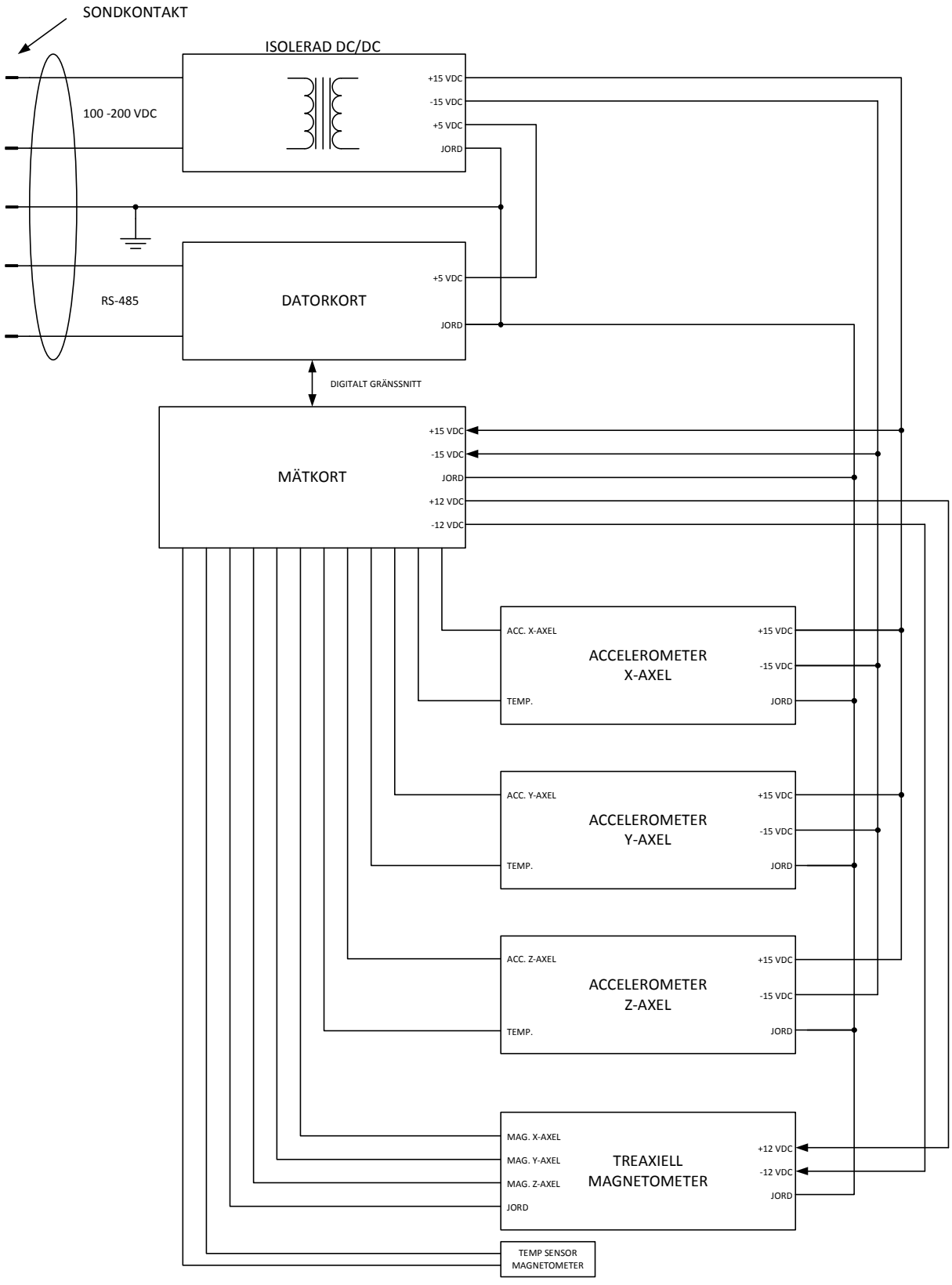


Bild 3-2. Blockschema över sonden.

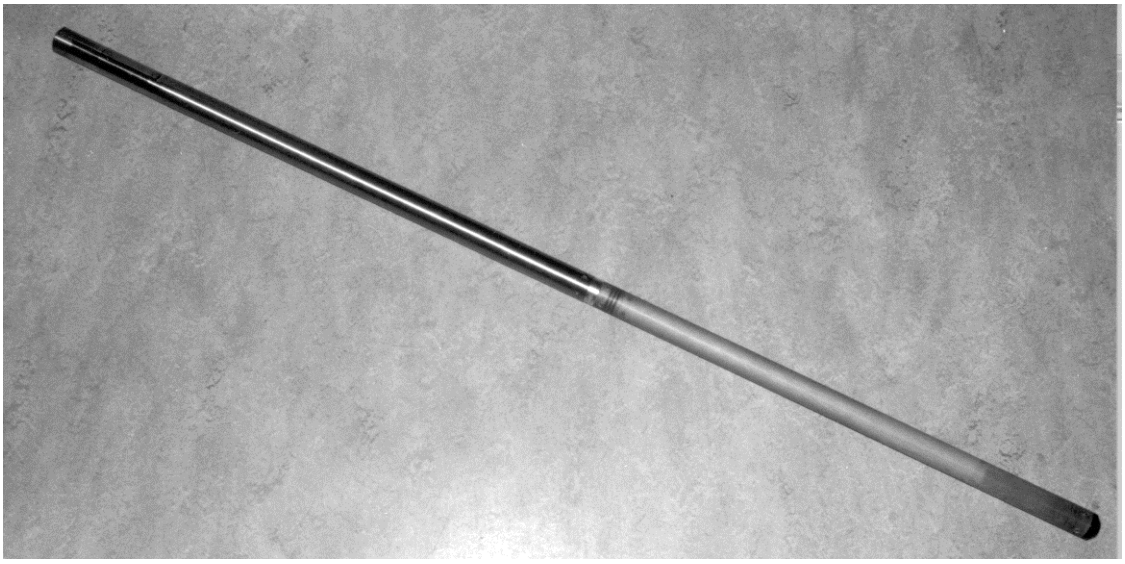


Bild 3-3. Komplett sond med ytterrör av glasfiber längst ner.

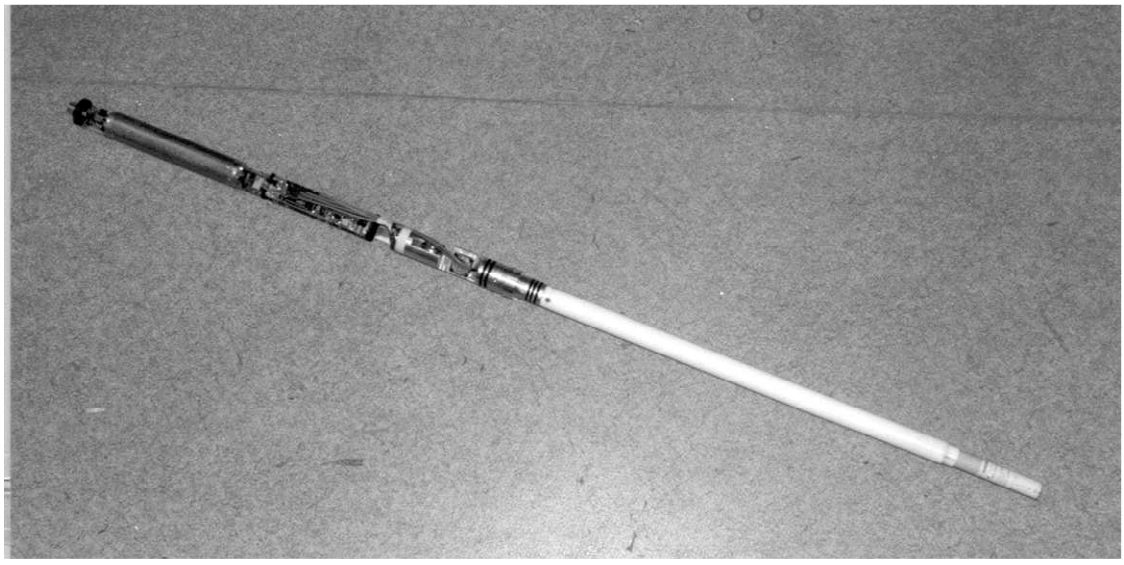


Bild 3-4. Komplett sond utan ytterrör med magnetometern längst ner.

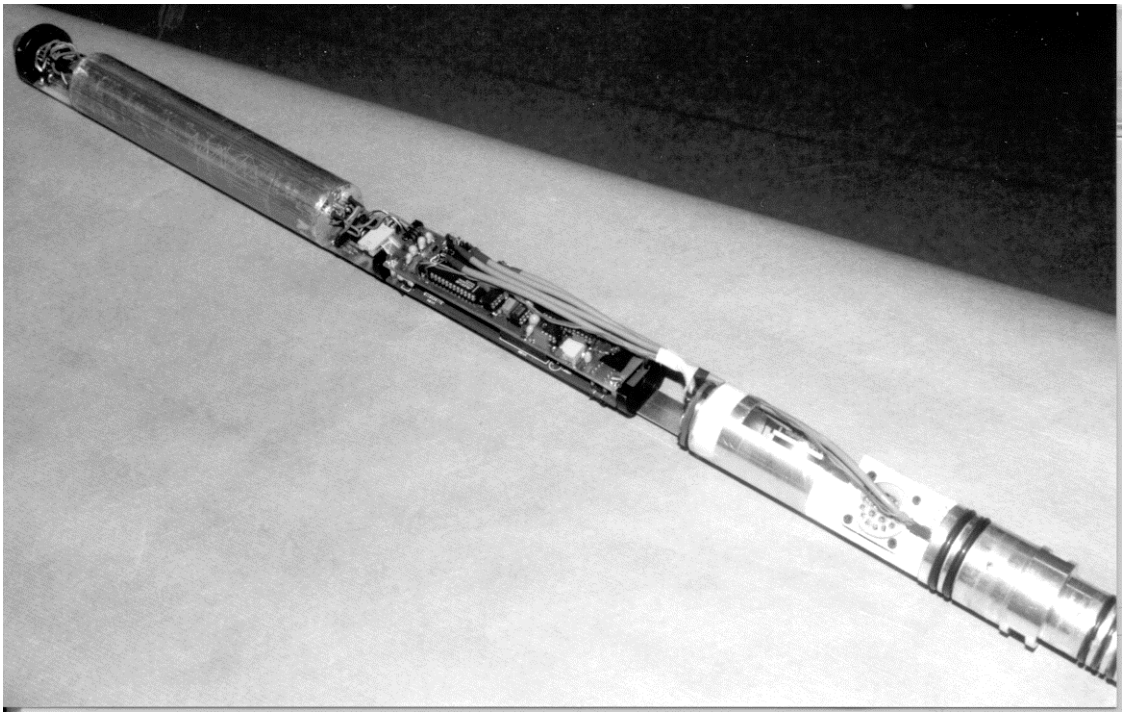


Bild 3-5. Närbild av sondelektronik och accelerometrar.

3.1.1 Accelerometrar

Det finns flera olika typer av accelerometrar. Huvudkriteriet för den här applikationen är att den skall kunna mäta statiska accelerationer eftersom jordens tyngdkraft är statisk. Detta medför att en mängd accelerometrar t.ex. av piezoelektrisk typ faller bort eftersom den bara kan mäta dynamiska accelerationer.

Förutom att accelerometrarna skall mäta statiskt ställs även andra krav som:

- Låg eller kalibrerbar offset
- Hög linjäritet
- Brett temperaturområde
- Små dimensioner
- Liten hysteres

Efter en diskussion med en distributör av givare framkom det att en servoaccelerometrar var den enda som klarade att uppfylla kraven. Valet föll på en accelerometer från Allied Signal: QA-700, se bild 3-6.

Servoaccelerometerns funktionssätt är att en massa hålls i ett jämviktsläge med hjälp av en spole i ett magnetfält. Sedan mäts den ström som krävs för att hålla massan i jämviktsläget, på så vis mäter den kraften som accelerometern utsätts för. De viktigaste data för QA-700 presenteras i tabell 3-1. QA-700 kan beställas med kalibreringsdata så att accelerometern kan linjäriseras och temperaturkompenseras med hjälp av mätmjukvaran. QA-700 innehåller en temperaturgivare som gör det lätt att temperaturkompensera givaren och uppnå en väldigt hög prestanda. Då de tre givarna skall monteras vinkelrätt mot varandra så utvecklades en hållare som en del av sondens mekanik, se bild 3-7.

Parameter	Värde	Kommentar
Matningsspänning	±13 till ±18 V	
Strömförbrukning	≤ 20 mA	Per spänning
Offsetfel	500 µg max	Temperaturmodellerat med data från tillverkare
Linjäritetsfel	500 ppm max	Temperaturmodellerat med data från tillverkare
Vinkelfel	≤ 2 mrad	≤ 0,41 mrad, individdata från tillverkare
Chocktålighet	250 g peak	
Temperaturområde	-55° till +95°	

Tabell 3-1. Data för QA-700

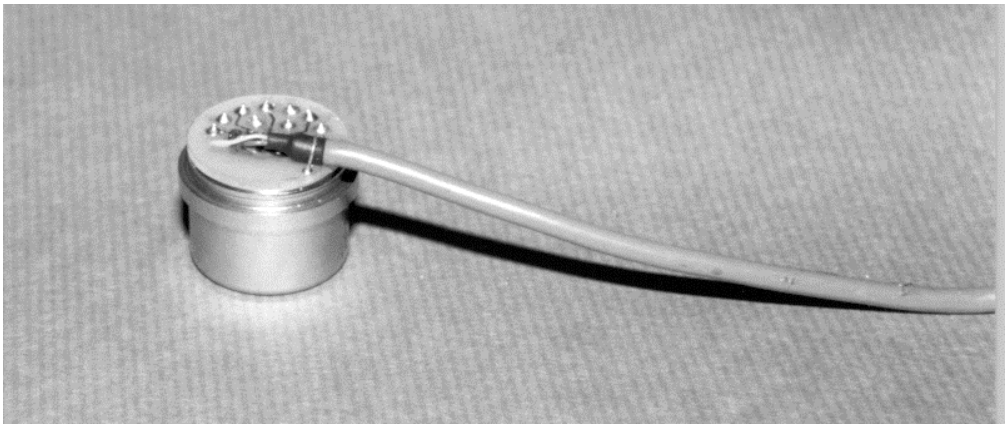


Bild 3-6. Accelerometern QA-700 med pålödd kabel.

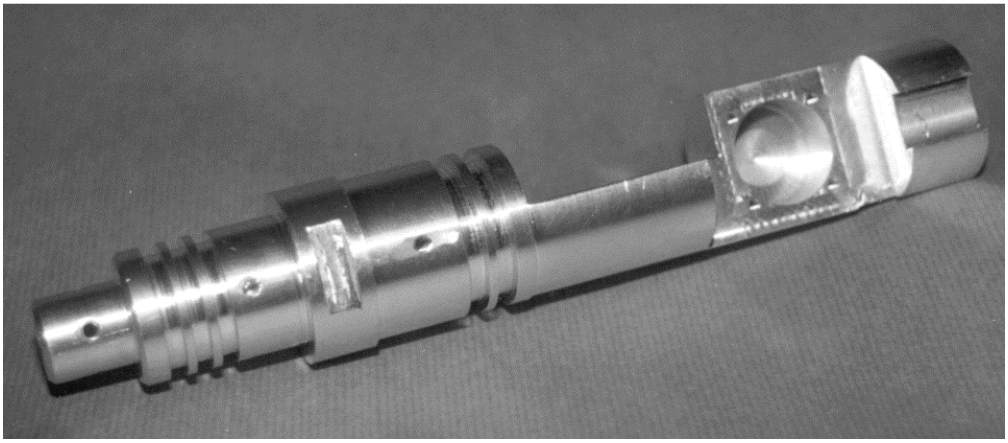


Bild 3-7. Hållare för de tre accelerometrarna.

3.1.2 Magnetometrar

Den magnetometer som ingick som förutsättning i arbetet är fluxgatemagnetometern. Den finns i treaxiellt utförande och har tillräckligt bra prestanda för applikationen, se bild 3-8. I detta fall valdes en fluxgatemagnetometer MAG-03MC från Bartington. Se tabell 3-2 för de viktigaste data.

Parameter	Värde	Kommentar
Matningsspänning	±12 till ±17 V.	
Strömförbrukning	≤ 20 mA	Per spänning
Offsetfel	±5 nT max	
Linjäritetsfel	< 15 ppm	
Kalibreringsfel	±0,5%	
Vinkelfel	±0,5° max	
Temperaturområde	-55° till +85°	

Tabell 3-2. Data för MAG-03MC

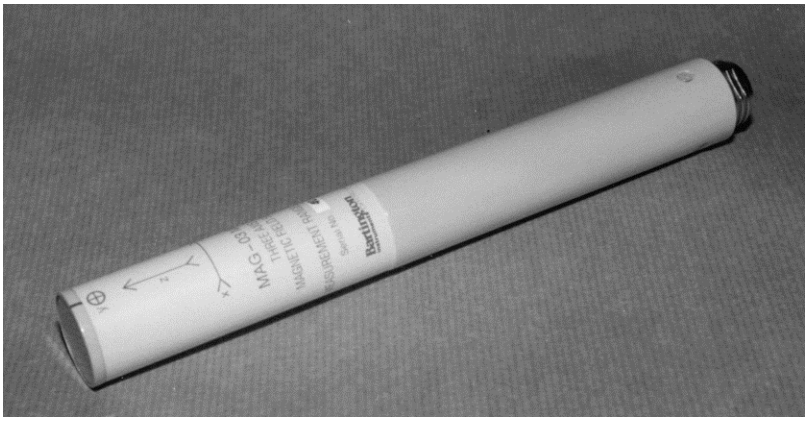


Bild 3-8. Magnetometern MAG-03MC.

3.1.3 Mätkort

För att kunna omvandla de uppmätta elektriska värdena från givarna för acceleration, magnetfält och temperatur, till respektive storheter, så måste de elektriska värdena omvandlas till digitala värden som sedan kan datorbehandlas. För att göra detta konstruerades ett mätinsamlingskort med ingångsväljare, filter och A/D-omvandlare, se bild 3-9.

Eftersom mätmetoden ställer låga krav på mäthastighet valdes en sigma-delta A/D-omvandlare AD7712 från Analog Devices. Den har valts för att den har hög precision kombinerat med ett lågt pris. Nackdelen är den låga bandbredden. Detta är dock inget problem eftersom det är statiska signaler som skall mätas. AD7712 kan ge maximalt 24-bitar om den ställs in så att den filtrerar signalen maximalt. Detta gör att det tar ca 10 sekunder att mäta alla kanaler på kortet.

För accelerometrarna finns det för varje axel två ingångar, en för acceleration och en för temperatur, båda med ingång för att mäta ström. För magnetometern finns det tre differentiella ingångar samt en icke differentiell ingång för att mäta ström (för temperaturgivare).

Ingångarna för magnetometern måste vara differentiella ty strömmen som går genom jordledaren är gemensam för strömförsörjning och alla axlar. Detta kommer att störa mätresultatet eftersom spänningsfallet i jordledaren annars kommer att bli en del av den uppmätta signalen. Därför mäts signalen differentiellt direkt i kontakten vid magnetometern mellan signalutgång och jord.

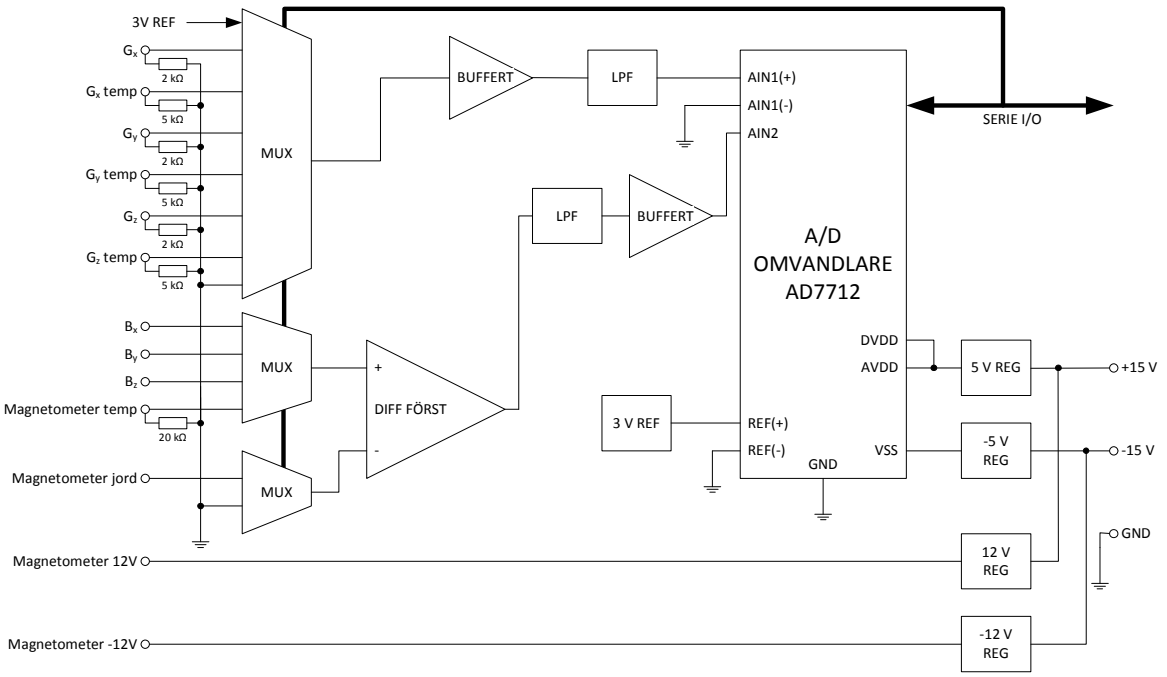


Bild 3-9. Blockschema över mätkortet.

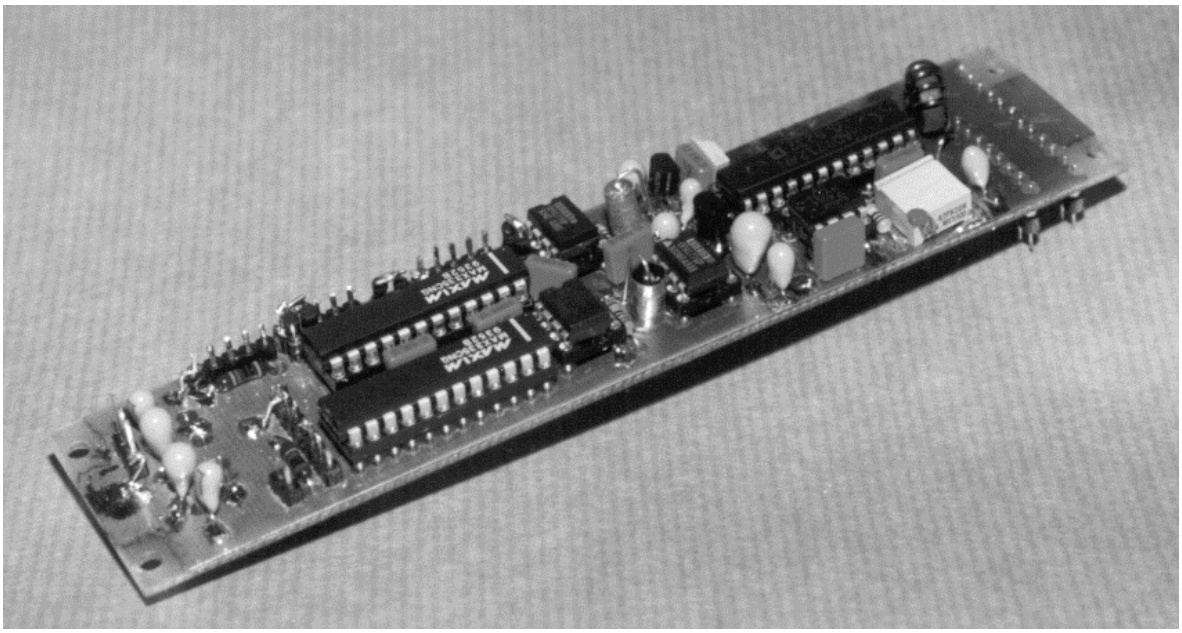


Bild 3-10. Mätkort till sonden.

3.1.4 Mikroprocessorkort

Mikroprocessorkortet, se bild 3-12, är uppbyggt runt en 80C552 processor från Philips som har en 8051 processorkärna, se bild 3-11. Kortet är en färdig design som använts i andra typer av sonder hos Malå Geoscience.

Grundläggande data för mikroprocessorkortet:

Minne:

- 64 kB SRAM minne
- 64 kB EPROM för lagring av program

Kommunikation:

- RS-485 interface
- Parallellportar för generella funktioner och kommunikation med periferikretsar bl.a. mätkortet.

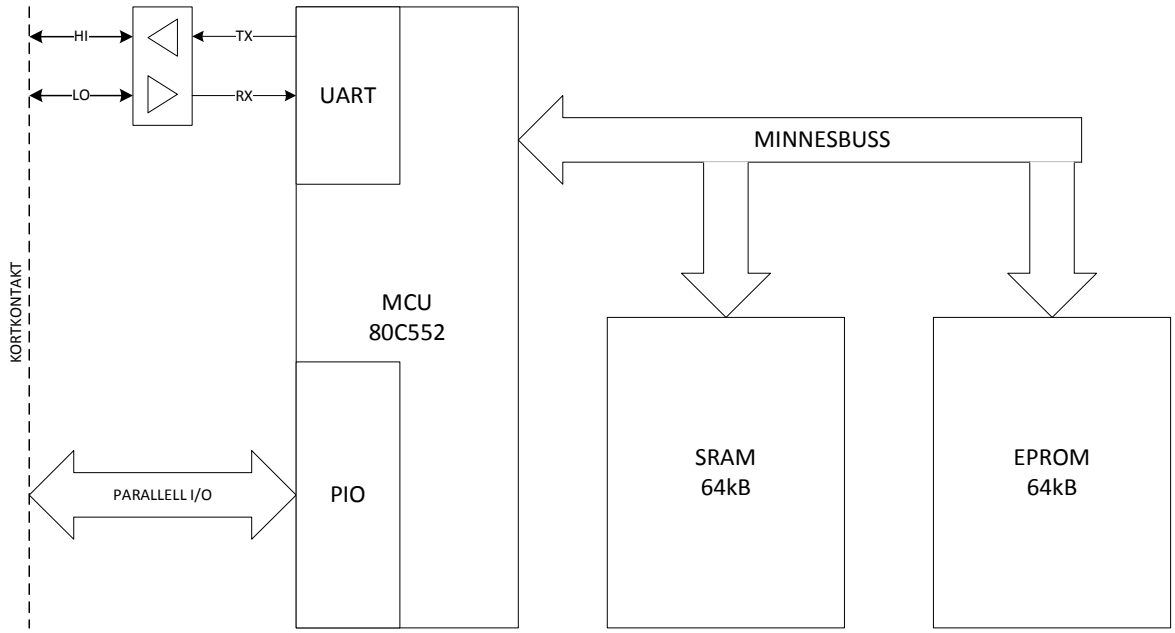


Bild 3-11. Blockschema över mikroprocessorkort.

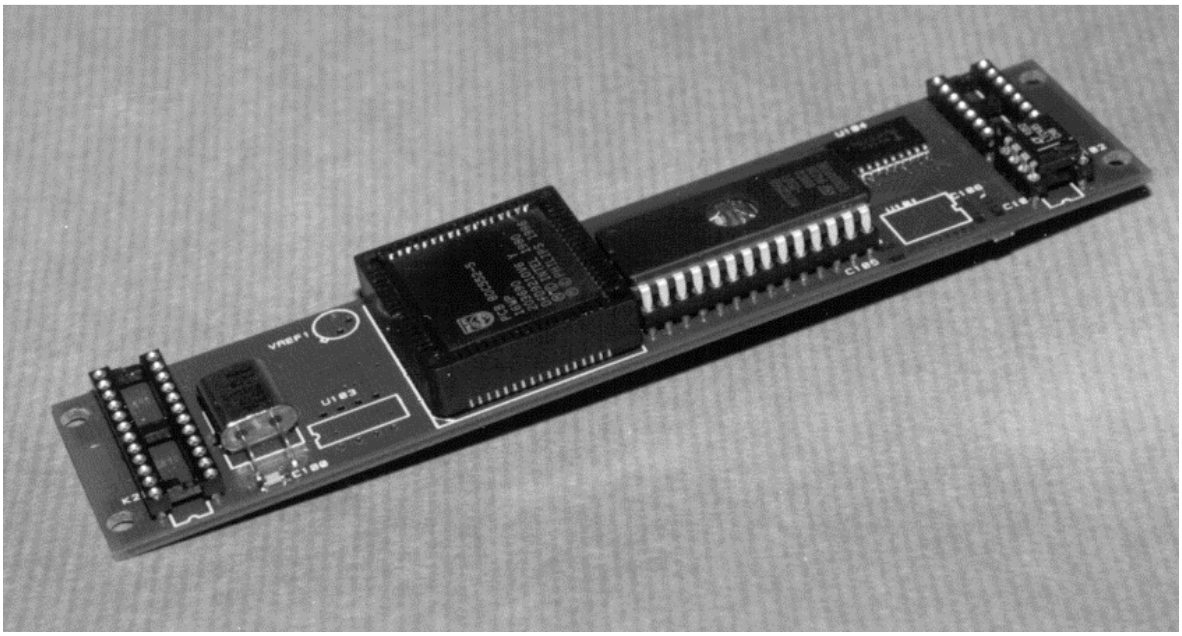


Bild 3-12. Mikroprocessorkort

3.1.5 Kraftkort

Kraftkortet till sonden konstruerades med en flybackomvandlare, som genererar +5V och ±15V för att försörja all elektronik i sonden. Omvandlaren konstruerades för att klara ett inspänningsområde mellan 100 och 200 VDC. Detta eftersom kabelförlusterna påverkar spänningen som finns tillgänglig vid sonden. En inspänning under 100 VDC är inte meningsfull då effektmaximum inte kan uppnås vid lägre inspänning då markaggregatet levererar 200 VDC. Då kretskortet, se bild 3-13, skall rymmas i sondröret krävdes en noggrann analys av måtten för att få plats. Komplett schema med komponentlista och finns i appendix E.

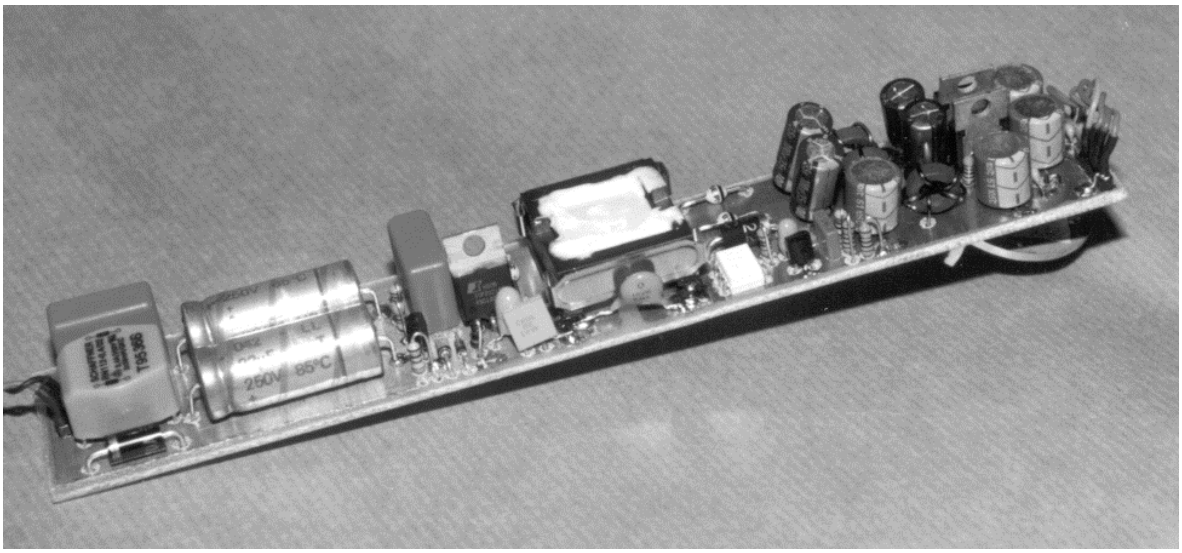


Bild 3-13. Kraftkort till sonden.

3.1.6 Sondprogram

Sondprogrammet som skrevs i C och assembler tar emot mätkommandon från marksystemet samt utför dessa. Mjukvaran består av en huvudloop som hela tiden genomlöps. Det finns två interruptrutiner som tar hand om kommunikation med marksystemet och mätkortet. Huvudloopen väntar tills kommandon finns i mottagningsbufferten. När ett kommando har mottagits avkodas detta och mätningen startas. När A/D-omvandlaren är klar läses data ut ur A/D-omvandlaren och skickas till marksystemet. Kommandot som skickas till sonden är ett 8 bitars ord, se tabell 3-3.

Kommando	Kommando ord [Hex]
Ställ in 10 Hz samplingsfrekvens	0x11
Ställ in 25 Hz samplingsfrekvens	0x12
Ställ in 30 Hz samplingsfrekvens	0x13
Ställ in 50 Hz samplingsfrekvens	0x14
Ställ in 60 Hz samplingsfrekvens	0x15
Mätning av Gx	0x21
Mätning av Gx sensor temperatur	0x22
Mätning av Gy	0x23
Mätning av Gy sensor temperatur	0x24
Mätning av Gz	0x25
Mätning av Gz sensor temperatur	0x26
Mätning av Bx	0x27
Mätning av By	0x28
Mätning av Bz	0x29
Mätning av magnetometer temperatur	0x2A
Mätning av samtliga sensorer	0x00
Kalibrering ingång ain1 på A/D	0x30
Kalibrering ingång ain2 på A/D	0x31

Tabell 3-3. Lista över kommandon till sonden.

När sonden har gjort mätningen skickas resultatet till markytan som tre stycken 8-bitars ord. Där det första ordet är mest signifikant. När kommandot mätning av samtliga sensorer har skickats svarar sonden med 30 stycken 8-bitars ord, i följden G_x, G_x temp, G_y, G_y temp, G_z, G_z temp, B_x, B_y, B_z och magnetometer temp.

Skalfaktorerna för de olika mätvärdena är listade i tabell 3-4. De digitala mätvärdena är representerade i intervallet 0xFFFFFF till 0x800000 för positiva tal och 0x7FFFFF till 0x000000 för negativa tal. Noll ligger alltså på mitt på skalan.

Kanal	Skalfaktor
Gx	1,37549E-07 g/LSB
Gx sensor temperatur	7,15256E-05 K/LSB
Gy	1,37549E-07 g/LSB
Gy sensor temperatur	7,15256E-05 K/LSB
Gz	1,37549E-07 g/LSB
Gz sensor temperatur	7,15256E-05 K/LSB
Bx	1,43051E-11 T/LSB
By	1,43051E-11 T/LSB
Bz	1,43051E-11 T/LSB
Magnetometer temperatur	1,78814E-05 K/LSB

Tabell 3-4. Skalfaktorer för mätvärden.

Mjukvaran kan ställas in för olika baudrate men i detta fall har 9600 baud valts eftersom det är en standardhastighet och inga större datamängder skall överföras. Övriga inställningar är 8-bitars ordlängd, en stoppbit och ingen paritet.

3.2 Markenhet

Markenheten är länken mellan Pc-datorn och sonden, och skall dessutom strömförsörja sonden. Då sonden har ett RS-485 och Pc-datorn ett RS-232 interface måste markenheten konvertera mellan RS-232 och RS-485, se bild 3-14. Då sonden behöver relativt hög spänning, 200 VDC och inkommande matning till systemet är 12 VDC så måste enheten omvandla 12 VDC till 200 VDC. På enheten finns kontakter för anslutning till 12 VDC källa, Pc-dator och sondkabel (kabeltrumma). Indikatorer för kommunikation (sändning och mottagning), 12 VDC och 200 VDC finns på markenheten för att underlätta funktionskontroll, se bild 3-15.

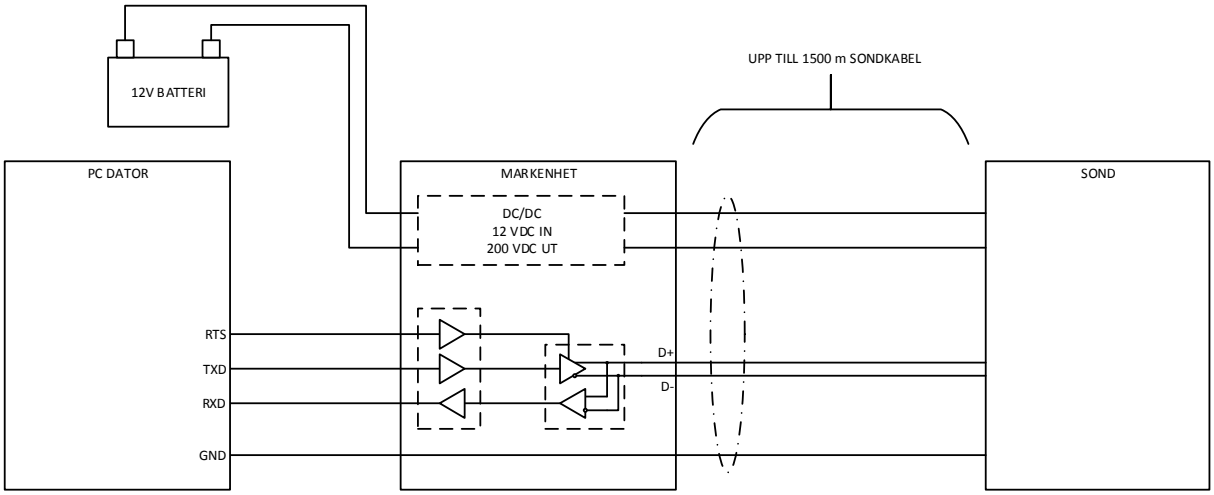


Bild 3-14. Blockschema över system med markenhet.

Bild 3-15. Markenhet.

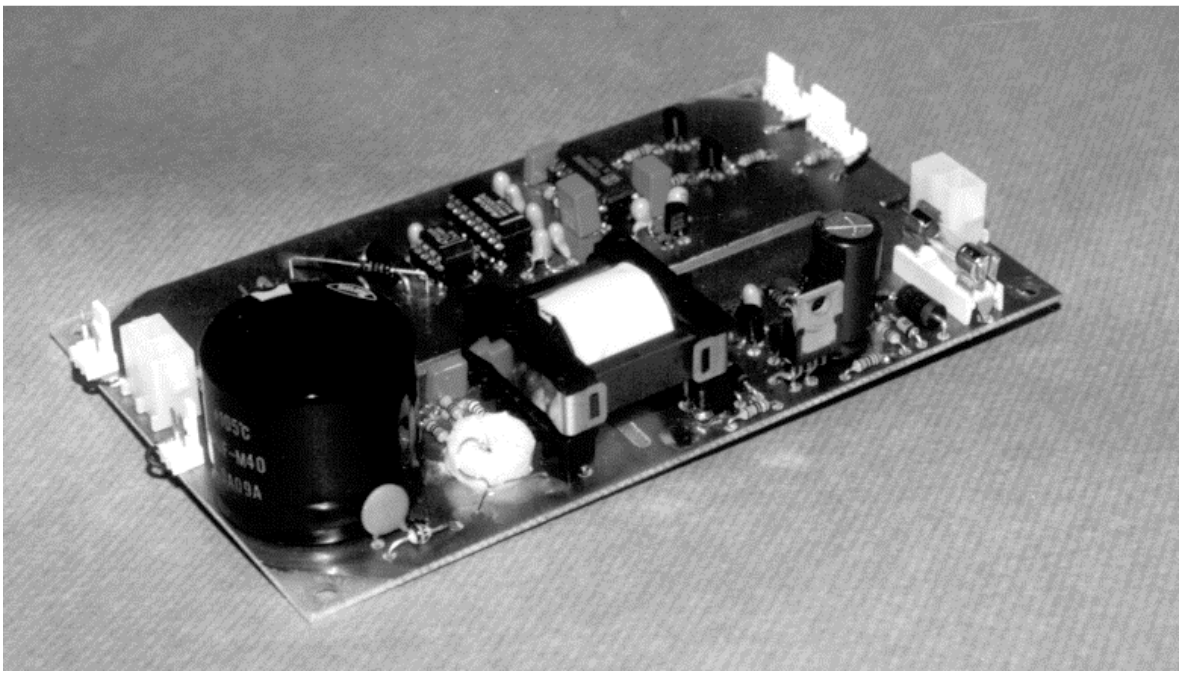


Bild 3-16. Kretskort till markenhet.

3.2.1 Kommunikationselektronik

Kommunikationselektroniken, se Bild 3-16, byggdes av en MAX232 krets som omvandlar RS-232 till CMOS nivåer och en LTC485 krets som omvandlar CMOS nivåer till RS-485 nivåer. Då RS-485 i detta fall används med halv duplex så styrs LTC485 transceiverns sändningsriktning av Pc-datorns RTS-signal som ingår i RS-232 interfacet. Dessutom finns det en enkel elektronik som tänder upp lysdioder när data skickas till och från sonden. För ritningar hänvisas till appendix F.

3.2.2 Strömförsörjning

Strömförsörjningen till sonden byggdes upp av en enkel icke isolerad flybackomvandlare på 20 W baserad på en LT1070 switchkrets. Omvandlaren är strömbegränsad för att skydda elektroniken från kortslutningar. En indikering av utspänning och inspänning finns för funktionskontroll. För ritningar se appendix F.

Parameter	Värde	Kommentar
Inspänning	9-14,5 VDC	Lämpligt område för ett 12 V batteri
Utspänning	200 VDC	
Maximal utström	100 mA	

Tabell 3-5. Data strömförsörjning markenhet.

3.2.3 Dator

Då programmet körs i DOS-fönster så ställs låga krav på operativsystemet. Dock krävs att operativsystemet stödjer direkt åtkomst av hårdvaran i COM porten. Pc-datorn skall ha en serieport med en 16650 kompatibel UART.

3.2.4 PC program

PC programmet utvecklades i C som en enkel MS-DOS applikation som går att köra i kommandofönstret under Windows. Programmet som kommunicerar med sonden via markenheten samlar in mätdata från sonden, beräknar kalibrerade mätvärden, azimut och inklination, därefter den geometriska förflyttningen i nord, öst och djupled. Data presenteras på skärmen och sparas i en fil, se appendix H.

PC programmet kräver ingen installation utan startas bara genom att köra LOGG.EXE. Programmet efterfrågar filnamn för mätresultatet, samplingsfrekvens, avstånd mellan mätpunkterna och startdjup. När inmatningen är klar startar första mätningen. När mätningen är klar så presenteras mätresultatet med djup, inklination, azimut, förflyttning och ackumulerad förflyttning i x, y och z-led. Nu kan man välja att stanna kvar och mäta om genom att trycka M, eller förflytta sonden till nästa mätpunkt med vinschen och trycka N för nästa mätning. När alla mätningar har genomförts så trycker man A, för att avsluta.

4 Simulering av systemets mätnoggrannhet

Kravet på avvikelsemätaren är att mätfelet på inklinationsmätningen skall vara mindre än ±0.1° och för azimutmätningen mindre än ±0.5°. Det visar sig dock att för azimutmätningen kan detta endast uppnås under vissa omständigheter. Vid mätning rakt ner kommer G_x och G_y komponenterna att bli väldigt små och ett litet offsetfel kommer att få stor inverkan på mätningen av sondens rotationsvinkel runt sin egen axel. Detta fel kommer att få direkt inverkan på azimut. Samma problem råder vid borrning vid nordpolen. Då kommer magnetkomponenten som bestämmer azimut att vara väldigt liten. För att få så bra mätning av azimut som möjligt så bör horisontalkomponenten av B vara så stor som möjligt och inklinationsvinkeln liten.

4.1 Analys av mätfel i elektroniken

De elektronikkomponenter som bidrar till mätfel ingår alla i mätkedjan.

För accelerometrarna är det:

- Resistanser för omvandling av ström till spänning.
- Multiplexer
- Mätförstärkare
- A/D-omvandlare
- Referens

För magnetometern är det:

- Multiplexer
- Mätförstärkare
- Buffertförstärkare
- A/D-omvandlare
- Referens

Eventuella förstärkningsfel som är gemensamma för kanalerna kommer att försvinna eftersom beräkningarna är normerade. Detta innebär också att fel i spänningsreferensen inte behöver beaktas. Vidare behöver inte heller eventuella fel i förstärkning i respektive mätkedja beaktas. En kanal i mätmultiplexern som är jordad används för att mäta upp offset i mätkedjan och på så sätt kompenseras offsetfelet bort.

De felbidrag som då är kvar är:

- Missmatch i resistanser för ström- till spänningsomvandling för accelerometrarna.
- Missmatch i resistans i mätmultiplexrar.
- Läckströmmar i multiplexer
- Linjäritetsfel i mätförstärkare
- Linjäritetsfel i A/D-omvandlare
- Upplösning i A/D-omvandlare

I appendix B visas hur stora dessa fel blir och är alla mindre än givarnas egna fel.

Offsetfel i mätkedjan för acceleration är ±5.4 μg jämfört med givarens offset på ±500 μg.

Olinjäritet i mätkedjan för acceleration är mindre än ±40 ppm vilket är mindre än accelerometrarnas maximala olinjäritet på ±500 ppm.

Offsetfel i mätkedjan för magnetometern är mindre än ±0,01 nT vilket är mindre än maximala offseten i magnetometern på ±5 nT.

Olinjäriteten på mindre än ±79 ppm är betydligt minder än det kalibreringsfel för magnetometern som är maximalt ±0.5%.

A/D-omvandlarens upplösning räcker till för att mäta rotationen med en upplösning på ±0.1° vid en inklination mellan -89,996° till 89,996°. Vilket är den mest kritiska mätningen när det gäller A/D-omvandlarens upplösning.

Sammanfattat så kommer givarna att dominera mätfelen.

4.2 Analys av mätfel i givare

För att undersöka hur bra systemet teoretiskt kan bli givet de givare som valts simulerades systemet i Matlab. Scriptet, se appendix C, räknar fram givarnas utsignaler givet magnetfält och tyngdkraft, i alla vinklar och rotationer som sonden kan ha (θ, φ och α). Positiva och negativa maximala offset- och skalfel läggs fel till i utsignalerna, i alla permutationer. Vinklarna räknas därefter ut, azimut θ och inklination φ. Den korrekta vinkeln dras från den framräknade vinkeln med mätfel, så endast mätfelet blir kvar. För varje mätpunkt tas det maximala absolutfelet ut ur mängden av felpermutationer.

I denna simulering tillämpas extremvärdesanalys vilket ger maximalt felbidrag som är en väldigt konservativ bedömning av mätfelet. Mekaniska toleranser har dock inte tagits med i denna analys.

För att hålla beräkningsbördan och datamängden på rimlig nivå valdes en vinkelupplösning på 2° för sondens rotation, azimut och inklination. Vilket ger 180 * 180 *90 = 2916000 analyspunkter och i varje punkt görs 4096 felpermutationer.

Jordens magnetfält kommer att påverka noggrannheten hos sonden och i denna simulering antogs en inklination på magnetfältet på 67° och en styrka på 55 μT, som representerar svenska förhållanden [5].

För magnetometern antogs ett skalfel mellan axlarna på ±0,5 % samt en offset på ±13 nT (en temperaturdrift inkluderad för 0 till 70°C). För accelerometrarna antogs ett skalfel på ±0,05 % och en offset på ±0,5 mg.

I bild 4-1 visas inklinationsfelet som funktion av inklination från simuleringen. Det maximala felet blir mindre än 0,08° vid ±48° inklination, vilket är mindre än ±0.1° som är kravet för systemet. Felet beror också på sondens rotation α. I diagrammet har maximalt värde för denna variation visats.

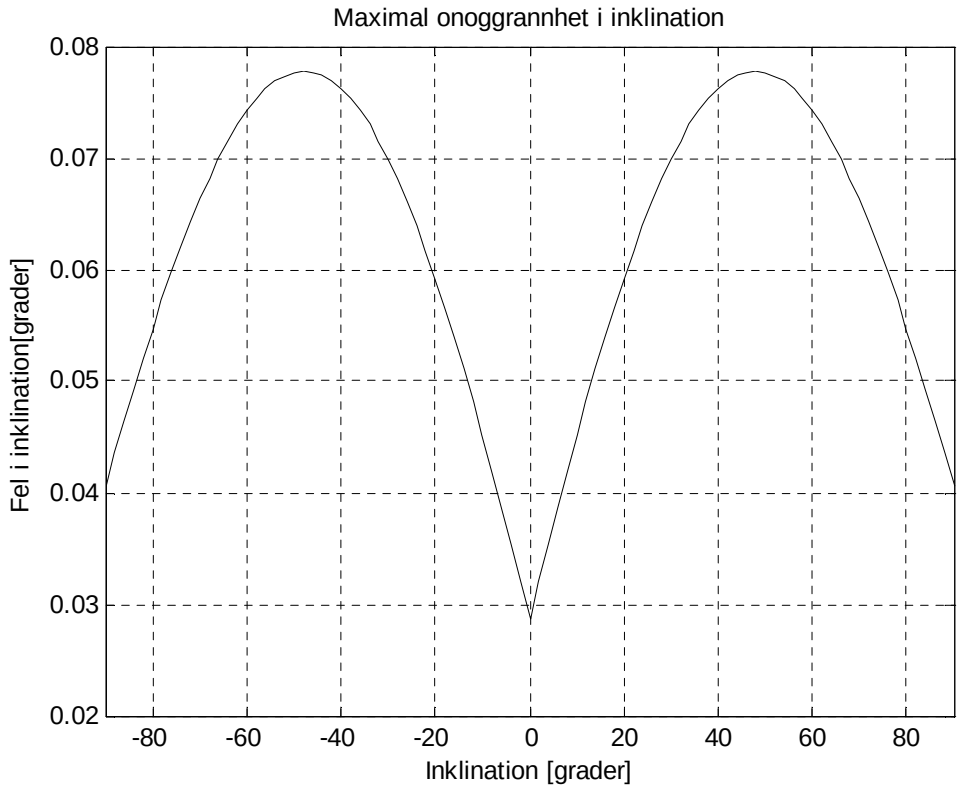


Bild 4-1. Inklinationsfel som funktion av inklinationen.

I bild 4-2 visas azimutfelet som funktion av azimut och inklination. Observera här att axeln för azimutfelet är logaritmisk, detta för att tydliggöra. I denna figur ser man tydligt hur azimutbestämningen får maximalt fel (±180°) när sonden är helt lodrät (inklination ±90°). Vid denna inklination är både G_x och G_y noll så att ett litet offsetfel i G_x eller G_y kommer att ge ett stort felbidrag.

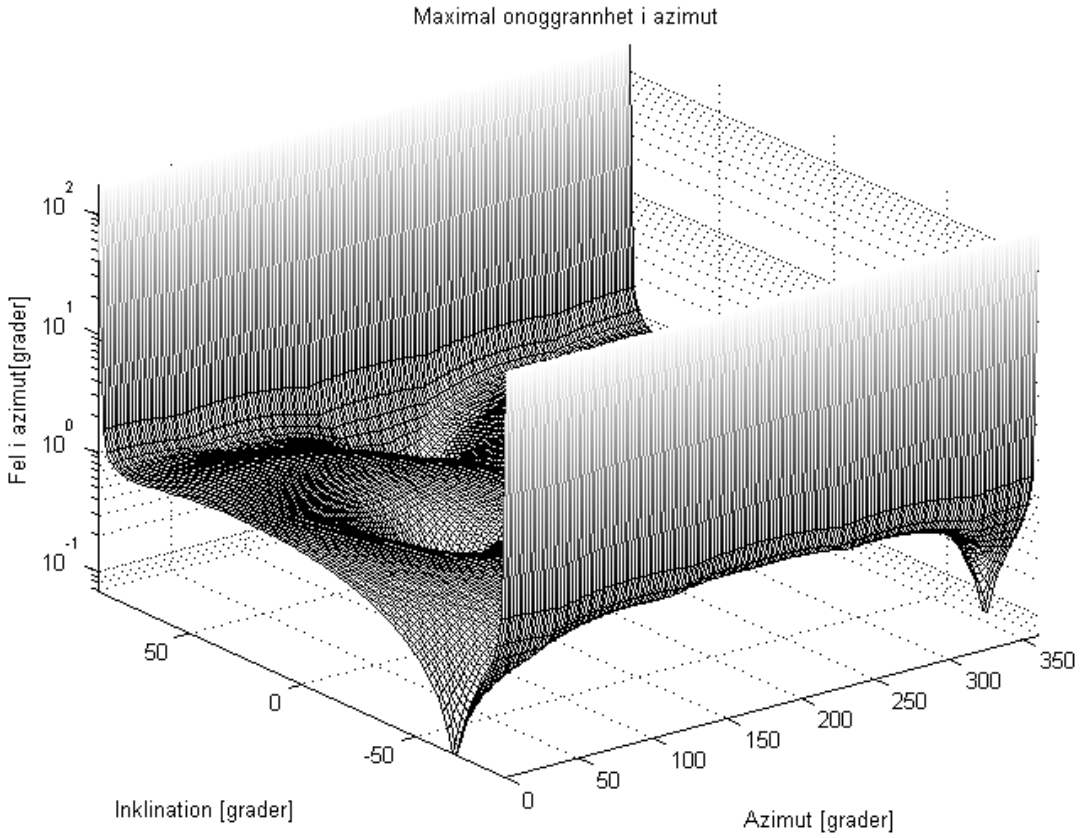


Bild 4-2. 3D-plot av azimutfelet som funktion av inklination och azimut.

För att närmare studera ytan mellan -80° och 80° i inklination har en konturbild skapats, se bild 4-3. I denna bild ser man att felet blir som lägst när sonden ligger helt längs magnetfältet, inklination 67° och rakt norrut, samt inklination -67° och rakt söderut. Detta beror på att felbidrag från G_x och G_y inte har någon inverkan eftersom B_x och B_y är väldigt små (teoretiskt noll). När sonden är horisontell (inklination 0°) får man två riktningar där felet är mindre, och det är rakt väster och österut (azimut 90° och 270°). Felbidragen i dessa riktningar blir lägre från Gz och Bz (teoretiskt noll). I bild 4-4 visas det maximala azimutfelet som funktion av inklination. Felvariationer som beror på rotation (α) och azimut (θ) har maximerats.

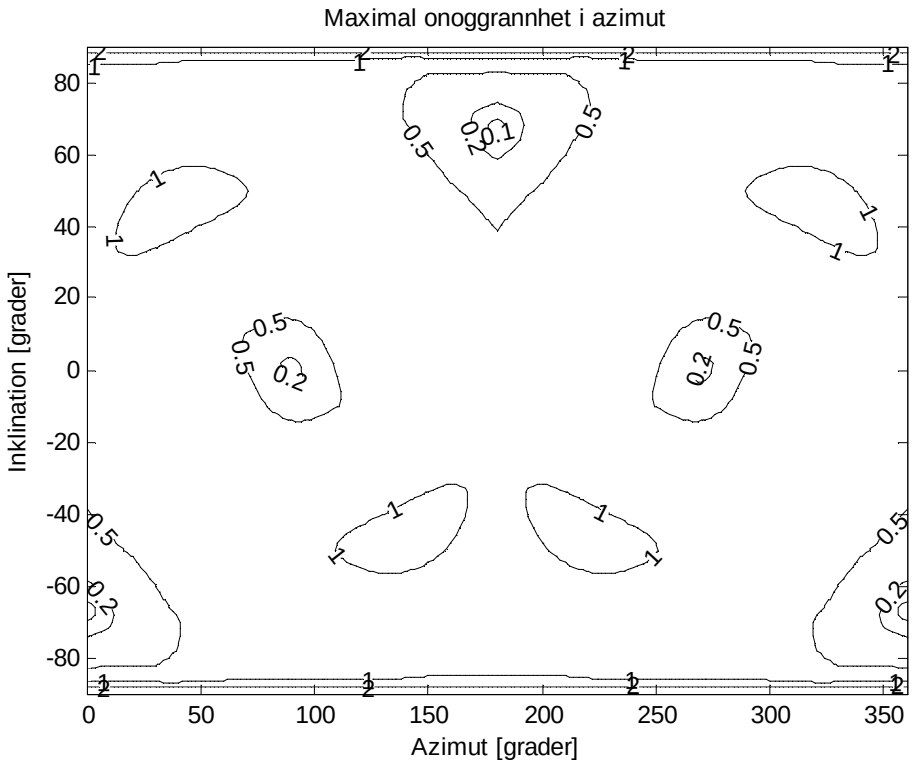


Bild 4-3. Kontur-bild av azimutfelet som funktion av inklination och azimut. Konturerna visar felet i grader.

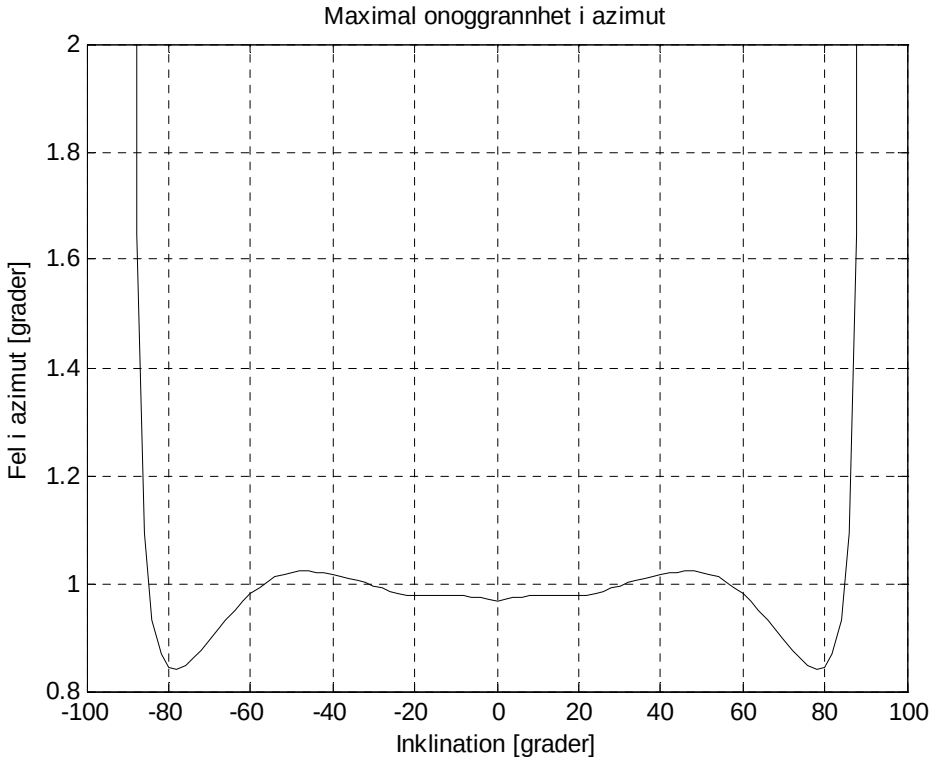


Bild 4-4. Maximalt azimutfel som funktion av inklination.

Sammanfattat är inklinationsmätningen inom kravet på ±0.1° med maximala felbidrag. Dessutom har accelerometrarna god ortogonalitet (0.03° enligt kalibreringsprotokoll från tillverkaren). Hållaren, se bild 3-7. har tillverkats med hög precision för att möta kravet på ortogonalitet (0.02°och ytjämnhet Ra 1,6).

För azimutmätningen nås inte kravet alls för vertikala borrhål, men om inklinationen ligger i intervallet -86° till 86° så har systemet ett fel som är mindre än ±1.10°, se tabell 4-1. Magnetometern som har en ortogonalitet ±0.5° kommer också att bidra till mätfelet. Kravet att nå en noggrannhet i azimut på ±0.5° kan bli svårt att nå.

En slutsats som kan dras är att kravet på ±0.5° för azimutmätningen måste kompletteras för att vara fullständigt, med begränsningar på borrhålets inklination, samt magnetfältets styrka och inklination.

Borrhål som är helt vertikala saknar riktning så om azimutbestämningen är fel i detta fall spelar ingen roll. Mätfel i azimut och inklination samt dess påverkan på borrhålets slutposition behandlas ingående i [2].

Inklination [°]	Maximalt fel i azimut [°]
90	± 180
88	± 1.65
86	± 1.10
84	± 0.94
46	± 1.02
-46	± 1.02
-84	± 0.94
-86	± 1.10
-88	± 1.65
-90	± 180

Tabell 4-1. Maximala azimutfel för några olika inklinationer.

5 Test och utvärdering

Testet gick ut på att dels utvärdera sondens mätnoggrannhet i en fixtur och sedan testa hela systemet i Malå Geoscience testborrhål och jämföra data med tidigare mätningar gjorda med Boremac D2 systemet.

5.1 Test i fixtur

Sonden testades genom att den monterades i en fixtur där sonden kunde roteras i dess frihetsgrader (θ, φ och α).

5.1.1 Test av inklinationsmätning

Sondens inklination ställdes in i 10° steg från 50°, 40° osv till -90° och tillbaka till 50° på andra sidan (dessutom mättes 45°), se appendix I. Tyvärr gick inte hela vinkelintervallet att täcka då sonden var för lång för fixturen. För att ställa in fixturen användes en inklinometer (INOGON) med en noggrannhet på 0.1°. Det hade varit önskvärt med en bättre inklinometer men det fanns inte att tillgå vid mättillfället. Mätningarna i appendix I visar att mätfelet är mindre än 0.1°. Eftersom instrumentet som användes för inställningen har lika hög noggrannhet som det som skall mätas upp så kan inte denna noggrannhet hävdas. En tumregel i industrin brukar vara att använda instrument som är minst 10 gånger bättre än det som skall kontrolleras eller kalibreras. En faktor 4 eller 3 gånger i mätnoggrannhet kan accepteras, men då måste marginaler läggas till med statistisk information i noggrannheten, för att med säkerhet säga att resultatet ligger inom toleranserna. I detta fall då den största avvikelsen är 0.09° så kan inte det dras några långtgående slutsatser utom att noggrannheten kan vara bättre än ±0.1°. Ett visst stöd finns för det då mätningarna sprider totalt 0,12° från minsta till största avvikelse.

4.1.2 Test av azimutmätning θ

För att testa azimut riktades sonden in med en handkompass med 1º gradering. Det hade varit önskvärt med en bättre kompass men det fanns inte att tillgå vid mättillfället. För att testa systemets noggrannhet gäller samma resonemang som i 5.1.1. Sonden testades i fixturen i fyra olika riktningar och med olika rotation på sonden (runt sondens Z-axel). Här kan slutsatsen dras att sonden inte är bättre än variationer som uppstod då sonden roterades i fixturen. I tabell 18-2 framgår att för azimut 180º är felet störst 2.508º. Därför kan det med säkerhet konstateras att sondens azimutmätning är sämre än ±1,254º. En övre gräns för felet är svårt att fastställa med aktuell testuppställning.

5.2 Test i borrhål

Sonden testades i Malå Geoscience testborrhål som finns under utvecklingslaboratoriets golv. Borrhålet som är 360,4 meter djupt med en diameter på 56 mm, har tidigare mätts med en Boremac D2 avvikelsemätare. Den mätningen används här som jämförelse. Mätningen med prototypen gjordes både på vägen ner och upp ur borrhålet. Mätningen påbörjades på ett djup av 25 meter för att undvika eventuella magnetisk påverkan från beklädnadsröret i stål som stabiliserar borrhålets översta del.

5.2.1 Jämförelse mellan mätning upp och ner i borrhålet

Denna mätning gjordes för att få en uppfattning om repeterbarheten hos mätsystemet, se appendix I. Borrhålet mättes två gånger med 5 m intervall mellan mätningarna. Först mättes på väg ner i borrhålet sedan på väg upp. Testet visar att inklinationsmätningen ger små fel utom i intervallet 155 - 190 meter. Två saker kan ha inträffat här. Det ena är att sonden roterat i borrhålet och visat på ett mätproblem och det andra är att sonden har ställt sig en aning snett i borrhålet. Största avvikelsen är 0.15° mellan upp och nerfärd i borrhålet och då största acceptabla skillnaden är 0.2° (±0.1°) så kan resultatet anses vara acceptabelt. För azimutmätningen är resultatet sämre, där största avvikelsen är 2.32° i samma djupintervall som inklinationsmätningen.

5.2.2 Jämförande mätning med Boremac D2

Denna jämförelse gjordes för att få en uppfattning om systemets noggrannhet jämfört med Boremac D2. Skillnaden mellan systemen i inklination ligger systematiskt på ungefär -0,5° och spridningen i fel mellan systemen ligger på 0,66°. Här kan diskuteras var felet ligger, i den nya sonden eller i Boremac D2, då Boremac D2 är specificerad att ha ett fel på max 0.5° i inklination [1]. Det kan dock aldrig bli mer än uppskattningar då borrhålet måste mätas med ett system som är mycket bättre än den nya sonden. Det har tidigare konstaterats att azimutmätningen för det nya systemet inte klarar kraven. Skillnaderna är ännu större här än vid testet i fixturen. I botten på borrhålet skiljer det 7,96° mellan systemen.

6 Slutsats

En fungerande implementation av en avvikelsemätare har byggts och testats. Noggrannheten för inklinationsmätningen för sonden ligger väldigt nära det specificerade på ±0,1°. Dock krävs mer noggranna instrument för att verifiera konstruktionen. För azimutmätningen nåddes inte kravet och mätningarna antyder att mätfelet ligger över 2.32 grader och kan till och med vara över 7.96 grader vilket i så fall måste förbättras i en framtida produkt. Orsaken till dessa mätfel kan ligga i olika faktorer:

– Magnetometern är specificerad till en ortogonalitet på ≤ ±0.5° och en inpassning mot kapslingen på ≤ ±0.5°. Bidraget från detta fel kan förklara en del av mätfelet men inte allt.

– Magnetometern är cirkulär och problematisk att passa in mot sonden då vinkelfel mellan sonden och magnetometern direkt påverkar noggrannheten.

– Sonden är skarvad vilket kan leda till att sonden inte blir tillräckligt rak för att klara kraven.

– Sonden innehåller magnetiska komponenter (accelerometrarna) som visserligen sitter en bit från magnetometern men som innehåller permanentmagneter. Dessa påverkar sannolikt magnetometern.

Simuleringen av mätnoggrannheten visar att det vid maximala felbidrag inte går att nå kravet för azimutnoggrannheten på ±0,5° utan snarare ±1.1, då inklinationen är i intervallet -86° till 86°. Dock är det osannolikt att alla givare ger maximala felbidrag. Sannolikt är att målet kan nås med en kompletterande kalibrering av sonden.

7 Förbättringsförslag

Först bör sonden göras än mer ren från magnetism genom att undersöka om accelerometrar med liknande prestanda finns tillgängliga utan magnetsystem. Samma typ av rör som användes för den nedre delen av sonden bör användas till hela sonden. En rakare sond som har mindre metall i sig blir då resultatet och detta bör förbättra azimutmätningen. Elektroniken måste då krympas och mindre accelerometrar användas då röret bara blir 32 mm invändigt.

Det bör finnas stöd i mjukvaran för kalibrering och kontroll av systemet. Kalibreringen kommer dock att bli svår att genomföra dels med de noggrannhetskrav som ställts och dels genom att hitta en strategi som minimerar antalet kalibreringspunkter.

Mjukvaran bör innehålla funktioner för plottning av hålets sträckning och innehålla funktion för export av data till andra mjukvaror för kartering.

8 Använda förkortningar och begrepp

8.1 Förkortningar

AC	Alternating Current
DC	Direct Current
DMM	Digital Multi Meter
EPROM	Erasable Programmable Read Only Memory
kB	Kilo Byte
LPF	Low Pass Filter
LSB	Least Significant Bit
MCU	Micro Controller Unit
PIO	Parallel Input/output
RS-232	Recommended Standard - 232
RS-485	Recommended Standard – 485
RTS	Request To Send
SRAM	Static Random Access Memory
UART	Universal Asynchronous Receiver Transmitter

8.2 Ordlista

Accelerometer är en sensor eller instrument som mäter acceleration. Enheten som instrumentet mäter i är i m/s2. Ibland använd även enheten g där 1 g motsvarar jordens tyngdkraft.

Azimut är riktningen i jordens horisontalplan mot ett objekt, eller en navigationsriktning. I denna rapport är riktningen angiven i grader i intervallet 0 – 360°, där 0° är rakt norrut (magnetiska norr). Azimut ökar positivt mot öster. Notera att i vissa beräkningar används även notationen med 0° för Nordlig, 90° för östlig, ±180° för sydlig och -90° västlig riktning. För att omvandla detta till 0 – 360° så läggs 360° till vid negativ azimut.

Gimbal är en lageranordning med låg friktion som används för att tillåta ett objekt att rotera runt en axel. En gimbal med två axlar används ofta på fartyg för att orientera kompassen parallell med horisonten.

Gyro är ett instrument som utnyttjar rörelsemängdens bevarande för att antingen skapa en gyrokompass för att göra sig oberoende av jordens magnetfält eller skapa ett instrument som kan mäta rotationshastigheten hos ett objekt ett så kallat rategyro.

Inklination är lutningen mellan ett objekt och horisontalplanet. I denna rapport är positiv lutning nedåt, negativ lutning uppåt och horisontell 0°. Lodrätt nedåt blir då +90°.

Inklinometer är ett instrument som mäter inklination.

Magnetometer är en sensor eller ett instrument som mäter magnetisk fältstyrka. Enheten som instrumentet mäter i är Tesla.

Tyngdkraft är i detta sammanhang summan av jordens gravitationskraft och centrifugalkraft.

9 Referenser

[1] Finkel Menachem, Thoring Torbjörn. *Handbok i prospekteringsborrning*. PBK-svenska gruvföreningen, 1989. ISBN 91-7970-826-9.

[2] Hearst Joseph R, Nelson Philip H. *Well logging for physical properties*. McGraw Hill, 1985, sid 503-513. ISBN 0-07-027698-6.

[3] Hayes D. *Borehole surveying – A systematic approach to attaining quality data*. Symposium on the application of borehole logging to mineral exploration and mining, Sunday 20th February 1994. Perth WA.

[4] Braithwaite Richard. *Where in earth are we?* geoDrilling International. December 1995, sid 10 – 15.

[5] Campbell Wallace H. *Introduction to Geomagnetic Field*s. Cambridge University press, 1997. ISBN 0-521-57193-6.

10 Appendix A: Härledning av inklination och azimut

Följande arbete har gjorts av Bernth Johansson, Malå Geoscience.

Här härleds formlerna för att beräkna sondens inklination φ och azimut θ För definitioner av vinklar och vektorer hänvisas till kapitel 2.7.

Sondens orientering i förhållande till omvärlden beskrivs med tre vinklar:

α - som anger sondens vridning kring sin Z-axel, sondens symmetriaxel.

φ - som anger sondens vinkel mellan Sx-Sy planet och G-vektorn, inklinationen.

θ - som anger sondens vinkel mellan sondens symmetriaxel och B-vektorn, azimut. Projicerat på normalplanet till G-vektorn dvs. kompassriktningen.

För att förtydliga resonemanget studerar vi sondens koordinatsystem, se bild 10-1.

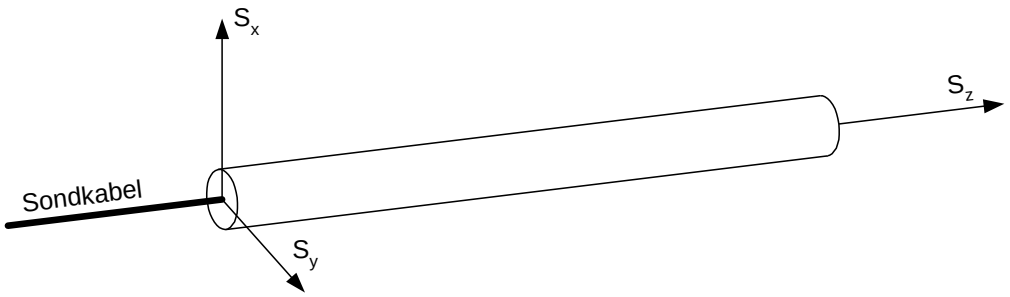


Bild 10-1. Sondens koordinatsystem.

10.1 Härledning av Inklinationen φ

Sonden har givare som känner av jordens tyngdkraft i tre komponenter Gx, Gy och Gz. Om dessa givare är orienterade som Sx, Sy och Sz enligt definitionen så kommer nu sonden med sina accelerometrar att kunna mäta jordens tyngdkraft i sondens koordinatsystem.

G:s projektion i Sx - Sy planet ger sondens vridning kring sin symmetriaxel Sz.

$$\tan(\alpha) = \frac{G_x}{G_y}$$

$$\sin(\alpha) = \frac{G_x}{\sqrt{G_x^2 + G_y^2}}$$

$$\cos(\alpha) = \frac{G_y}{\sqrt{G_x^2 + G_y^2}}$$

Genom att vrida G vinkeln -α runt Sz axeln fås G’ i Sz – Sy planet och då kan vinkeln mellan G’ och Sz axel beräknas som

$$G_x' = G_x \cos(\alpha) + G_y \sin(\alpha) = \frac{G_x G_y - G_x G_y}{\sqrt{G_x^2 + G_y^2}} = 0$$

$$G_y' = -G_x \sin(\alpha) + G_y \cos(\alpha) = \frac{G_x^2 + G_y^2}{\sqrt{G_x^2 + G_y^2}} = \sqrt{G_x^2 + G_y^2}$$

$$G_z' = G_z$$

G’ ligger nu i Sy – Sz planet.

Nu kan inklinationen φ bestämmas som.

$$\tan(\varphi) = \frac{G_z'}{G_y'} = \frac{G_z}{\sqrt{G_x^2 + G_y^2}}$$

$$\cos(\varphi) = \frac{G_y'}{\sqrt{G_y'^2 + G_z'^2}} = \frac{\sqrt{G_x^2 + G_y^2}}{\sqrt{G_x^2 + G_y^2 + G_z^2}}$$

$$\sin(\varphi) = \frac{G_z'}{\sqrt{G_y'^2 + G_z'^2}} = \frac{G_z}{\sqrt{G_x^2 + G_y^2 + G_z^2}}$$

10.2 Härledning av azimut θ

Sonden mäter upp magnetfältet i tre komponenter Bx, By och Bz. Givarna är orienterade som sondens koordinatsystem S_x, S_y och S_z. B är vektorn som beskriver jordens magnetism.

B:s projektion i Sx – Sy planet.

$$\tan(\delta) = \frac{B_x}{B_y}$$

$$\sin(\delta) = \frac{B_x}{\sqrt{B_x^2 + B_y^2}}$$

$$\cos(\delta) = \frac{B_y}{\sqrt{B_x^2 + B_y^2}}$$

Koordinatsystemet (S_x, S_y, S_z) vrids nu kring S_z-axeln så att S'_x-axeln hamnar i horisontalplanet dvs. -α.

$$B'_x = B_x \cos(\alpha) - B_y \sin(\alpha) = \frac{B_x G_y - B_y G_x}{\sqrt{G_x^2 + G_y^2}}$$

$$B'_y = B_x \sin(\alpha) + B_y \cos(\alpha) = \frac{B_x G_x + B_y G_y}{\sqrt{G_x^2 + G_y^2}}$$

$$B'_z = B_z$$

Nu vrids (S'_x, S'_y, S'_z) systemet kring S'_x-axeln så att även S'_z-axeln hamnar i horisontalplanet. Det går då ur B:s projektion i S''_x - S''_z planet att bestämma sondens azimut. Vinkeln som skall vridas är φ.

$$B''_x = B'_x$$

$$B''_y = B'_y \cos(\varphi) + B'_z \sin(\varphi)$$

$$B''_z = -B'_y \sin(\varphi) + B'_z \cos(\varphi)$$

Då följer

$$B''_x = \frac{B_x G_y - B_y G_x}{\sqrt{G_x^2 + G_y^2}}$$

$$B''_y = \frac{B_x G_x + B_y G_y + B_z G_z}{\sqrt{G_x^2 + G_y^2 + G_z^2}}$$

$$B''_z = \frac{B_z (G_x^2 + G_y^2) - G_z (B_x G_x + B_y G_y)}{\sqrt{G_x^2 + G_y^2}\sqrt{G_x^2 + G_y^2 + G_z^2}}$$

Nu kan β för sonden beräknas.

$$\tan(\beta) = \frac{B''_x}{B''_z} = \frac{(B_x G_y - B_y G_x)\sqrt{G_x^2 + G_y^2 + G_z^2}}{B_z (G_x^2 + G_y^2) - G_z (B_x G_x + B_y G_y)}$$

Arctangens ger endast β i intervallet -90° till 90° och vi behöver veta vinkeln i intervallet -180° till 180°. Genom att kontrolleras tecken på täljare och nämnare skaffas information om i vilken kvadrant β ligger. Denna funktion finns definierad som arctan2 i C (funktionen ligger i intervallet -180° till 180°). Vid navigation är vinkeln definierad i intervallet 0-360° där 0° är norrut och positiv vinkel ökar mot öster. Rotationsriktningen blir alltså tvärtom mot β. Rotationsriktningen skall alltså korrigeras och 360° skall läggas till om resultatet är negativt. Vi får alltså där θ ligger i intervallet -180° till 180° i formeln nedan. En korrektion måste alltså göras om resultatet blir negativt.

$$\theta = -\beta = -\arctan 2\left[\frac{(B_x G_y - B_y G_x)\sqrt{G_x^2 + G_y^2 + G_z^2}}{B_z(G_x^2 + G_y^2) - G_z(B_x G_x + B_y G_y)}\right]$$

11 Appendix B: Toleransanalys av elektronik

I mätkedjan för sensorerna ingår elektronik som bidrar till mätfel.

I signalkedjan för accelerometrarna ingår motstånd för ström till spänningsomvandling, multiplexer, buffertförstärkare, filter och A/D-omvandlare, se bild 11-1. Mätkedja för Accelerometer och magnetometeringångar.

Då A/D-omvandlaren innehåller stöd för offsetkompensering genom mätkedjan har detta utnyttjats. För att A/D-omvandlaren skall kunna mäta offseten i mätkedjan så slås switchen SW2 till och SW1 från, se bild 11-1. I detta läge mäts offset upp för alla ingående komponenter. När sedan SW1 slås till och SW2 slås från uppstår en skillnad och det är att R_{DSon} för en annan kanal i multiplexern används och motståndet som strömmen mäts i kopplas in. Variationen i R_{DSon} och resistansen i mätmotståndet ger en offset på grund av läckströmmarna i multiplexern och operationsförstärkarens biasström. Summan av strömmarna är ±6.4 nA detta i den kombinerade resistansen R_{DSon} + mätmotståndet (2 kΩ + 225 Ω) ger en offset på ±14 µV. Denna offset kompenseras alltså inte.

Offsetfelet hos accelerometrarna översatt till acceleration motsvarar ungefär ±14 µV / 2 kΩ / 1.3 mA/g) = ±5.4 µg. Offset i accelerometrarna som är mindre än ±500 µg är den dominerande komponenten därför kan offset i mätelektroniken försummas.

Olinjäriteter finns i motståndet som mäter strömmen, R_{DSon} i multiplexern, operationsförstärkaren och A/D-omvandlaren. Olinjäriteten som dominerar är A/D-omvandlaren, se tabell 11-1 för både offsetfel och olinjäriteter.

De gemensamma komponenternas förstärkningsfel tas bort eftersom de försvinner då mätningarna är normerade. Förstärkningsfelet på ±26 ppm från resistansernas missmatch kompenseras dock inte men är mycket mindre än accelerometrarnas olinjäritet på ±500 ppm.

Den summerade olinjäriteten är ±40 ppm. Olinjäriteten i accelerometrarna som är mindre än ±500 ppm är den dominerande komponenten därför kan olinjäriteten i mätelektroniken försummas.

Komponent	Typ	Offsetfel	Olinjäritet	Förstärkningsfel	Anmärkningar
Resistor	S102K	0	±0,3 ppm	±26 ppm	Motstånd uppmätt med DMM
Multiplexer	MAX335	±89 nV	±0,05 ppm	-	2225 Ω * 400 pA, Olinjäritet beror på variation i R_{DSon}.
Mätförstärkare	OPA177	±13,3 µV	±9 ppm	-	Offset på grund av biasström 2225 Ω * 6 nA, Övrig offset kalibreras bort. Olinjäritet beror på variation i råförstärkning och CMRR.
A/D-omvandlare	AD7712	0	±30 ppm	-	Offset kompenseras genom kalibrering-
Summa fel	-	±14 µV	±40 ppm	±26 ppm	-

Tabell 11-1. Felbidrag i accelerometerns mätkedja.

I signalkedjan för magnetometrarna ingår multiplexer, differentialförstärkare, filter, buffertförstärkare och A/D-omvandlare, se bild 11-1.

Då A/D-omvandlaren innehåller stöd för offsetkompensering genom mätkedjan har detta utnyttjats. För att A/D-omvandlaren skall kunna mäta offseten i mätkedjan så slås switcharna SW4och SW6 till och SW3och SW5 från, se bild 11-1. I detta läge mäts offset upp för alla ingående komponenter. När sedan SW3 och SW5 slås till och SW4 och SW6 slås från uppstår en skillnad och det är att R_{DSon} för en annan kanal i multiplexern används. Variationen i R_{DSon} ger en offset på grund av läckströmmarna i multiplexern och differentialförstärkarens biasström. Summan av strömmarna är ±4.4 nA och i R_{DSon} 225 Ω ger det en offset på ±1 µV. Denna offset kompenseras alltså inte.

Offseten på ±1 µV motsvarar ett magnetfält på ±0,01 nT vilket är försumbart jämfört med ±5 nT som är magnetometerns offset.

Olinjäriteten på 79 ppm är betydligt mindre än magnetometerns kalibreringsfel som är specificerat till 0,5%. Olinjäriteten kan då försummas för magnetometerns mätkedja.

Eftersom alla magnetometerkanaler går genom samma förstärkare så kan eventuella förstärkningsfel ignoreras eftersom mätningarna normeras.

Komponent	Typ	Offsetfel	Olinjäritet	Förstärkningsfel	Anmärkningar
Multiplexer	MAX335	±89 nV	±0,05 ppm	-	Offset och linjäritet på grund av variation av RDSon och läckström
Mätförstärkare	AD620	±0,9 µV	±40 ppm	-	Offset på grund av biasström genom RDSon
Mätförstärkare	OPA177	0	±9 ppm	-	Offset kompenseras genom kalibrering
A/D-omvandlare	AD7712	0	±30 ppm	-	Offset kompenseras genom kalibrering
Summa fel	-	±1 µV	±79 ppm	-	-

Tabell 11-2. Felbidrag i magnetometerns mätkedja.

A/D-omvandlarens har en upplösning är 24-bitar och detta kan vålla problem exempelvis när sonden används i hål som är nära lodräta. För att kunna bestämma rotation på sonden vid hög inklination måste ett tillräckligt antal bitar nyttjas för att få önskad noggrannhet. Antag att rotationen skall bestämmas med ±0,1° (±0.5° är kravet så en faktor 5 bättre är antagen för att ge utrymme för andra fel). För att få ett vinkelfel på mindre än ±0.1° krävs ca 9 bitar och tecken. Full utsignal från accelerometrarna ger en spänning på 2.6 V (2 kΩ * 1,3 mA). A/D omvandlaren har en upplösning på 23 bitar och tecken. 2^23 * 2.6V /3V = 7270126 (3V kommer från 3V referensen). Ur detta får vi inklinations max för en rotationsnoggrannhet på bättre än ±0.1° som arctan (7270126/512) vilket är 89,9960 det innebär att upplösningen i A/D omvandlaren är tillräcklig eftersom felet som accelerometrarna introducerar är betydligt större ±1,65° vid 88° inklination.

Följande data för komponenterna är inhämtat från tillverkarnas datablad:

- Tolerans hos resistanserna för accelerometeringångarna ±0,01%. Dessa motstånd mättes upp med en DMM med ±26 ppm noggrannhet och de värdena användes sedan i mjukvaran.
- Tolerans hos RDSon i mätmultiplexrar är inte specificerat men kan vara max 200 ohm, eftersom RDSon är max 200 ohm.
- Variation i RDSon hos multiplexerar 25 ohm pga. spänningsvariationer
- Läckström i multiplexer 400 pA (8 utgångar ihopkopplade och max 70° C)
- Biasström i OPA177 ±6 nA max.
- Offset i OPA177 ±100 µV max.
- CMRR i OPA177 140 dB
- Råförstärkning i OPA177 6000 V/mV
- Biasström i AD620 ±4 nA
- Offset i AD620 125 µV
- CMRR i AD620 90 dB
- Olinjäritet i AD620 40 ppm
- AD7712 INL ±0.003%

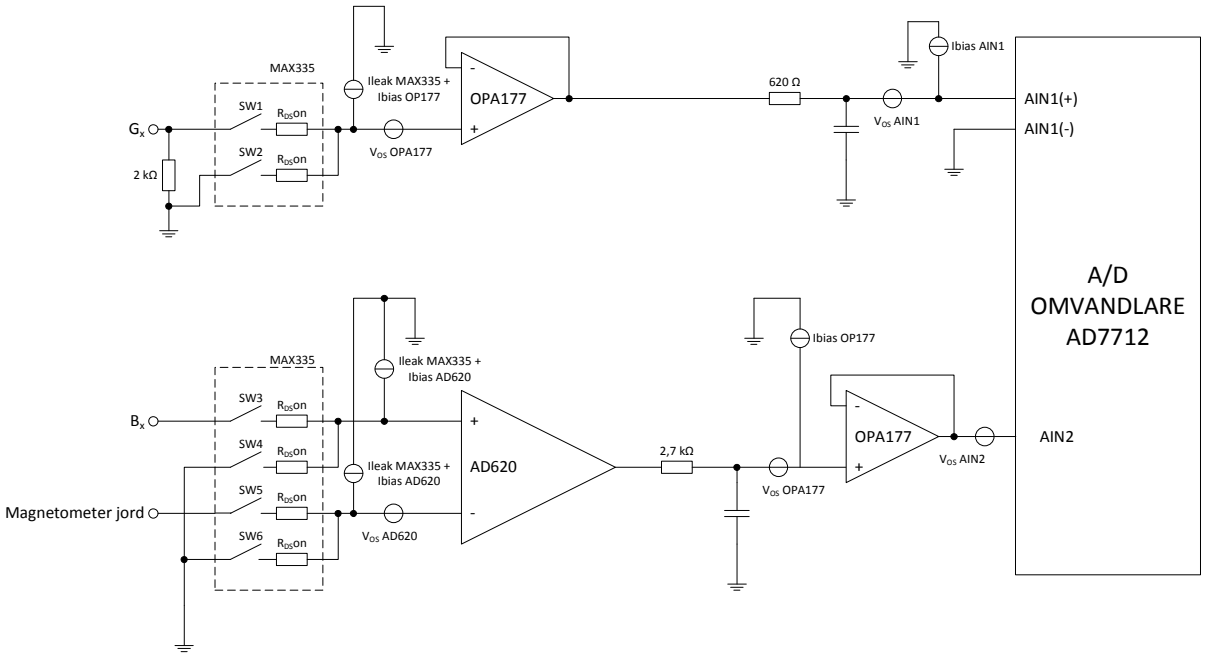


Bild 11-1. Mätkedja för Accelerometer och magnetometeringångar.

12 Appendix C: Toleransanalys av givare

12.1 Matlabscript för beräkning

```
function felanalys_calculation
% ========================================================================
%
% Funktion som beräknar avvikelsen hos avvikelsemätaren som funktion
% av azimuth och inklination
%
% ========================================================================
%
% ========================================================================
% Data hos jordmagnetiska fältet
% ========================================================================
%
% Inklination i grader hos magnetfältet och styrka i nT
%
maginkl = 67;
B_total = 55000;
%
% Uppdelat som komponeter i det jordfixa koordinatsystemet
%
Bx_jord = B_total * cos(maginkl * pi/180);
Bz_jord = B_total * sin(maginkl * pi/180);
%
% ========================================================================
% Fel hos givare
% ========================================================================
%
% Bias hos magnetometer 13 nT
%
Bias_mag = 13;
%
% Linearitetsfel hos magnetometer 0.5% mismatch mellan axlarna
%
Lin_mag = 0.005;
%
% Bias hos accelerometer 500 ug
%
Bias_acc = 500E-6;
%
% Lineäritetsfel hos accelerometer 500 ppm
%
Lin_acc = 500E-6;
%
% ========================================================================
% Övriga indata
% ========================================================================
%
% Bestäm hur finkornig analysen skall vara
%
% Antal grader mellan analyspunkterna
%
res = 2;
%
incr = 2*pi/(360/res);
%
% ========================================================================
% Beräkningsdel
% ========================================================================
%
% Gör vinkelmatriser, alfa = rotation, fi = dip, theta = azimuth
%
[alfa, fi ,theta] = meshgrid (-pi:incr:pi, -pi/2:incr:pi/2, -pi:incr:pi);
%
% Gör matriser för utdata
%
theta_max_error = zeros( size(theta));
```

```
alfa_max_error = zeros( size(alfa));
fi_max_error = zeros( size(fi));
%
% Beräkna Gx,Gy,Gz
%
Gx_ini = -cos(fi) .* cos(alfa);
Gy_ini = cos(fi) .* sin(alfa);
Gz_ini = sin(fi);
%
% Beräkna projektionen av jordens magnetvektor mot sondens koordinatsystem
%
Bx_ini = Bx_jord .* ( cos(alfa) .* sin(fi) .* cos(theta) - sin(alfa) .* sin(theta)) - Bz_jord
.* cos(alfa) .* cos(fi);
By_ini = Bx_jord .* ( -sin(alfa) .* sin(fi) .* cos(theta) - cos(alfa) .* sin(theta)) + Bz_jord
.* sin(alfa) .* cos(fi);
Bz_ini = Bx_jord .* cos(theta) .* cos(fi) + Bz_jord .* sin(fi);
%
% Löp igenom alla kombinationer av fel
%
tic;
% signcomb = 4095;
for signcomb = 0:4095
%
% Lägg till fel i Gx, Gy och Gz
%
Gx = (1 + esign(signcomb,1) * Lin_acc) .* Gx_ini + (esign(signcomb,4) * Bias_acc);
Gy = (1 + esign(signcomb,2) * Lin_acc) .* Gy_ini + (esign(signcomb,5) * Bias_acc);
Gz = (1 + esign(signcomb,3) * Lin_acc) .* Gz_ini + (esign(signcomb,6) * Bias_acc);
%
% Lägg till fel i Bx, By och Bz
%
Bx = (1 + esign(signcomb,7) * Lin_mag) .* Bx_ini + (esign(signcomb,10) * Bias_mag);
By = (1 + esign(signcomb,8) * Lin_mag) .* By_ini + (esign(signcomb,11) * Bias_mag);
Bz = (1 + esign(signcomb,9) * Lin_mag) .* Bz_ini + (esign(signcomb,12) * Bias_mag);
%
% Beräkna rotation runt sondens symmetriaxel (alfa)
%
alfa_ber = atan2 (Gy, -Gx);
%
% Kontrollera om det finns data som ligger i olika kvadranter dvs passerat +/- pi
%
corr_mtrx_minus_alfa = (alfa < -pi/2) & (alfa_ber > pi/2);
corr_mtrx_plus_alfa = (alfa > pi/2) & (alfa_ber < -pi/2);
%
% Lägg till eller dra ifrån 2*pi
%
alfa_ber = alfa_ber - (2*pi .* corr_mtrx_minus_alfa);
alfa_ber = alfa_ber + (2*pi .* corr_mtrx_plus_alfa);
%
% Beräkna vinkelfelet
%
alfa_error = abs((alfa - alfa_ber) .* (180/pi));
%
% Felet kan aldrig bli större än +/- 180 grader pga. moduloeffekten
%
corr_mtrx_180_alfa = alfa_error > 180;
alfa_error = -180 .* corr_mtrx_180_alfa + alfa_error;
%
% Lagra det största felet
%
alfa_max_error = max(alfa_error, alfa_max_error);
%
% Beräkna dipvinkel (fi)
%
fi_ber = atan(Gz./sqrt(Gx.^2 + Gy.^2));
%
% Beräkna felet i (fi)
%
fi_error = abs((fi - fi_ber) .* (180/pi));
%
% Felet kan aldrig bli större än +/- 90 grader pga. moduloeffekten
```

```
%
corr_mtrx_90_fi = fi_error > 90;
fi_error = -90 .* corr_mtrx_90_fi + fi_error;
%
% Lagra största felet hittills
%
fi_max_error = max(fi_error, fi_max_error);
%
% Beräkna azimut (theta)
%
theta_ber = atan2((Bx .* Gy - By .* Gx) .* sqrt (Gx.^2 + Gy.^2 + Gz.^2), Bz.*(Gx .^2 + Gy.^2)-
Gz.*(Bx.*Gx + By.*Gy));
%
% Ändra tecknet på theta_ber för att få rätt rotationsriktning
%
theta_ber = -theta_ber;
%
% Kontrollera om det finns data som har "wrappat över" dvs passerat +/- pi
%
corr_mtrx_minus = (theta < -pi/2) & (theta_ber > pi/2);
corr_mtrx_plus = (theta > pi/2) & (theta_ber < -pi/2);
%
% Lägg till eller dra ifrån 2*pi
%
theta_ber = theta_ber - (2*pi .* corr_mtrx_minus);
theta_ber = theta_ber + (2*pi .* corr_mtrx_plus);
%
% Beräkna vinkelfelet
%
theta_error = abs((theta - theta_ber) .* (180/pi));
%
% Felet kan aldrig bli större än +/- 180 grader
%
corr_mtrx_180 = theta_error > 180;
theta_error = -180 .* corr_mtrx_180 + theta_error;
%
% Lagra största felet hittills
%
theta_max_error = max(theta_error, theta_max_error);
%
% ========================================================================
% Spara data
% ========================================================================
%
save 'resultat';
end;
% ========================================================================
% En liten hjälpfunktion
% ========================================================================
function e = esign (binary,pos)
if bitget(binary,pos) == 1
    e = 1;
else
    e = -1;
end
```

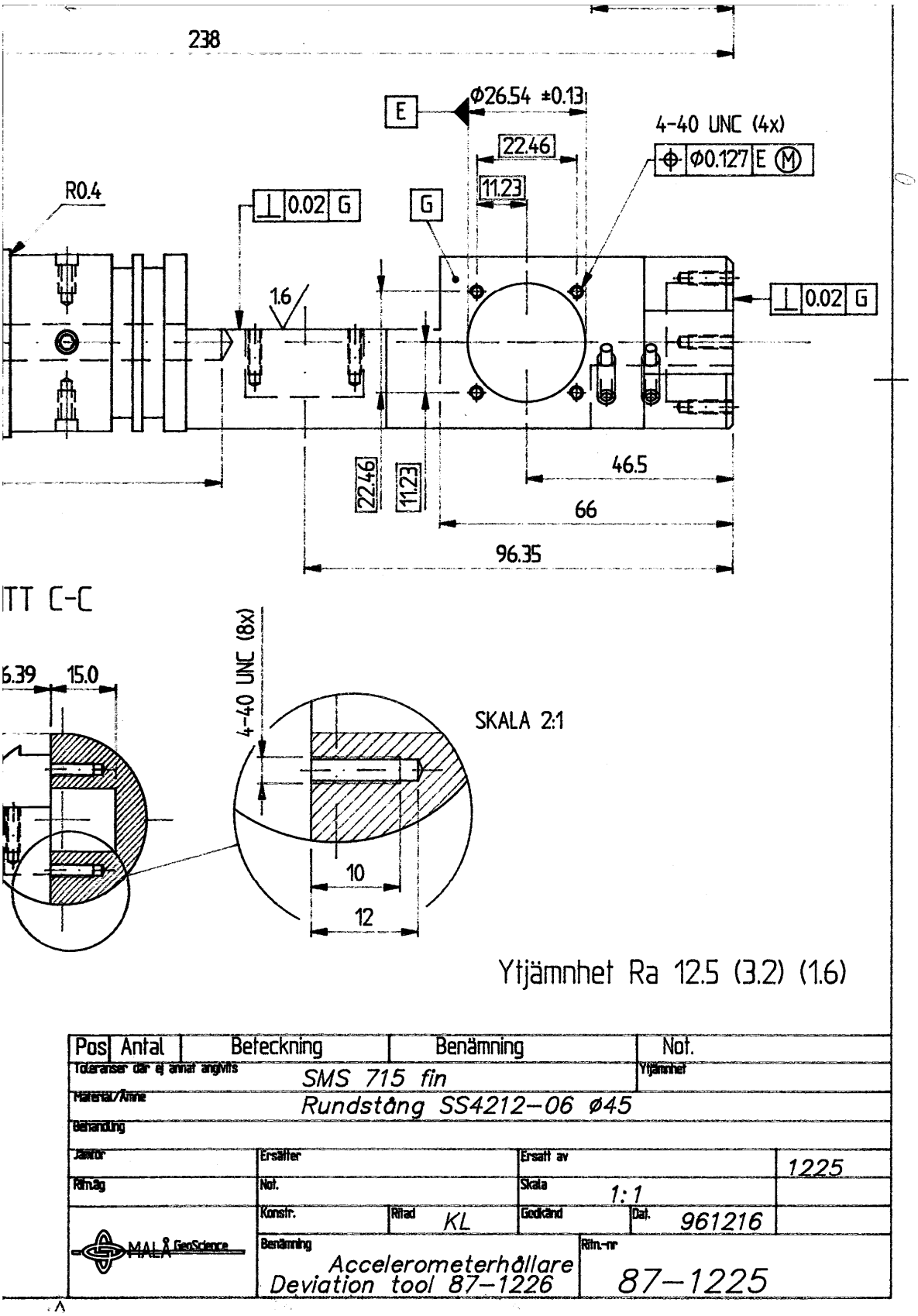
12.2 Matlabscript för presentation

```
function felanalys_presentation
% ========================================================================
%
% Funktion som plottar data framräknat i
% av azimuth och inklination
%
% ========================================================================
% Hämta data
% ========================================================================
%
load 'resultat';
%
% ========================================================================
% Presentera data
% ========================================================================
%
% Plotta inklinationsfelet
%
[x,y] = meshgrid(0:incr:2*pi, -pi/2:incr:pi/2);
z = squeeze(max(fi_max_error,[],3));
colormap(gray);
%
% Som en x-y plot
%
p = squeeze(max(z,[],2));
plot(-90:2:90,p(1:91),'k');
xlim([-90 90]);
title('Maximal onnoggrannhet i inklination');
ylabel('Fel i inklination[grader]');
xlabel('Inklination [grader]');
grid on;
%
% Plotta Rotationsfelet
%
figure;
[x,y] = meshgrid(0:incr:2*pi, -pi/2:incr:pi/2);
z = squeeze(max(alfa_max_error,[],3));
colormap(gray);
%
% Som en mesh
%
handle = meshc((180/pi) .* x,(180/pi) .* y,z);
title('Maximal onoggrannhet i rotation');
xlabel('Rotation [grader]');
ylabel('Inklination [grader]');
zlabel('Fel i rotation[grader]');
set(gca,'ZScale','log')
xlim([0 360]);
ylim([-90,90]);
zlim([0 180]);
%
% Som inzoomning av intressant område
%
figure;
p = squeeze(max(z,[],2));
plot(-90:2:90,p,'k');
title('Maximal onnoggrannhet i rotation');
xlabel('Inklination [grader]');
ylabel('Fel i rotation [grader]');
ylim([0 2]);
grid on;
%
% Plotta azimutfelet
%
figure;
[x,y] = meshgrid(0:incr:2*pi, -pi/2:incr:pi/2);
z = squeeze(max(theta_max_error,[],2));
colormap(gray);
```

```
%
% Som en mesh
%
handle = meshc((180/pi) .* x,(180/pi) .* y,z);
title('Maximal onoggrannhet i azimut');
xlabel('Azimut [grader]');
ylabel('Inklination [grader]');
zlabel('Fel i azimut[grader]');
set(gca,'ZScale','log')
xlim([0 360]);
ylim([-90,90]);
zlim([0 180]);
shading interp;
%
% Som en konturplot
%
figure;
colormap(gray);
[C,h] = contour((180/pi) .* x,(180/pi) .* y,z,[2, 1, 0.5, 0.2, 0.1]);
caxis ([1 6]);
clabel(C,h);
title('Maximal onnoggrannhet i Azimut');
xlabel('Azimut [grader]');
ylabel('Inklination [grader]');
%
% Som inzoomning av intressant område
%
figure;
p = squeeze(max(z,[],2));
plot(-90:2:90,p,'k');
title('Maximal onnoggrannhet i azimut');
xlabel('Inklination [grader]');
ylabel('Fel i azimut [grader]');
ylim([0.8 2]);
grid on;
end;
```

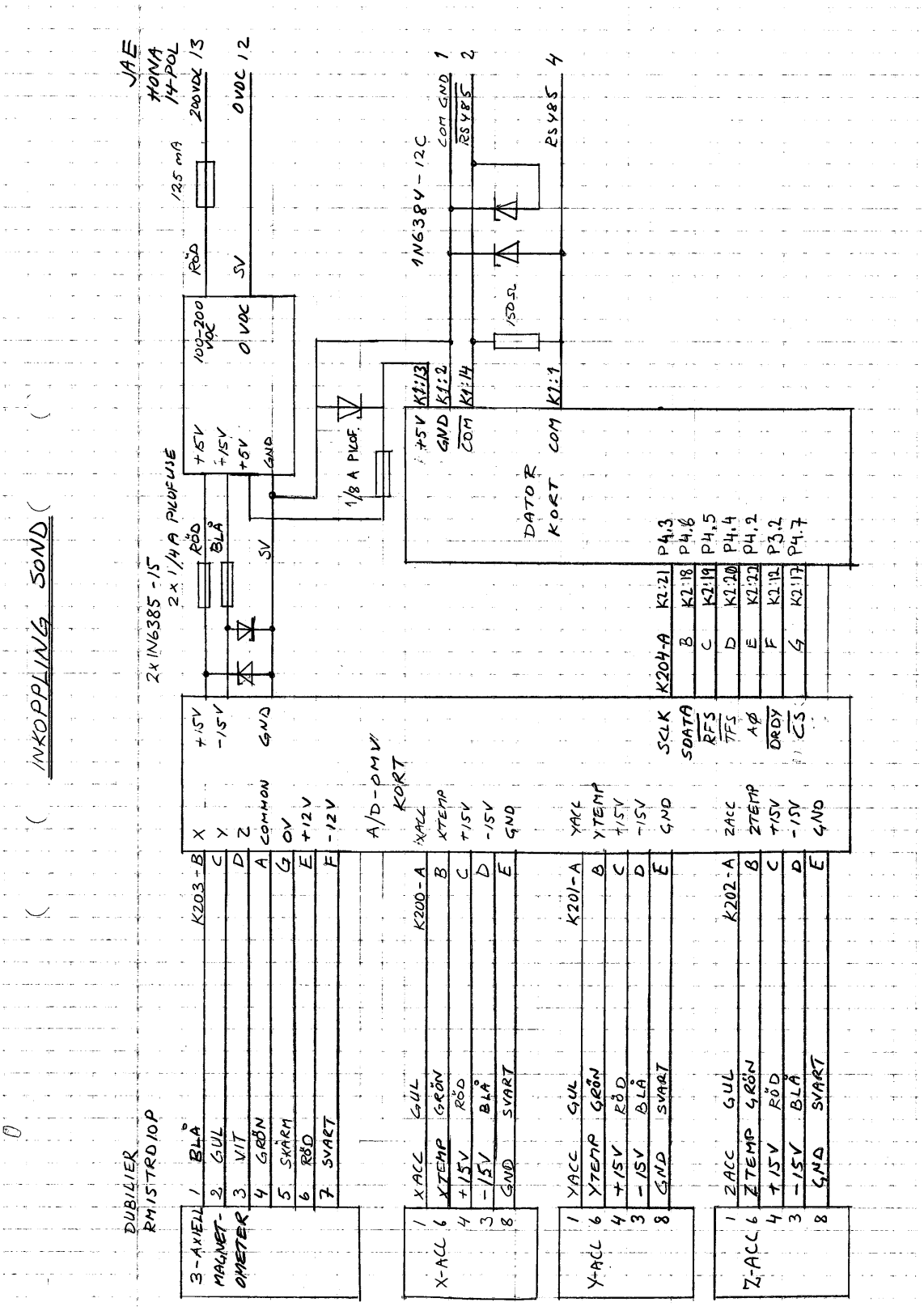
13 Appendix D: Ritningar på sondens mekanik

Ej inkluderade på grund av storlek.

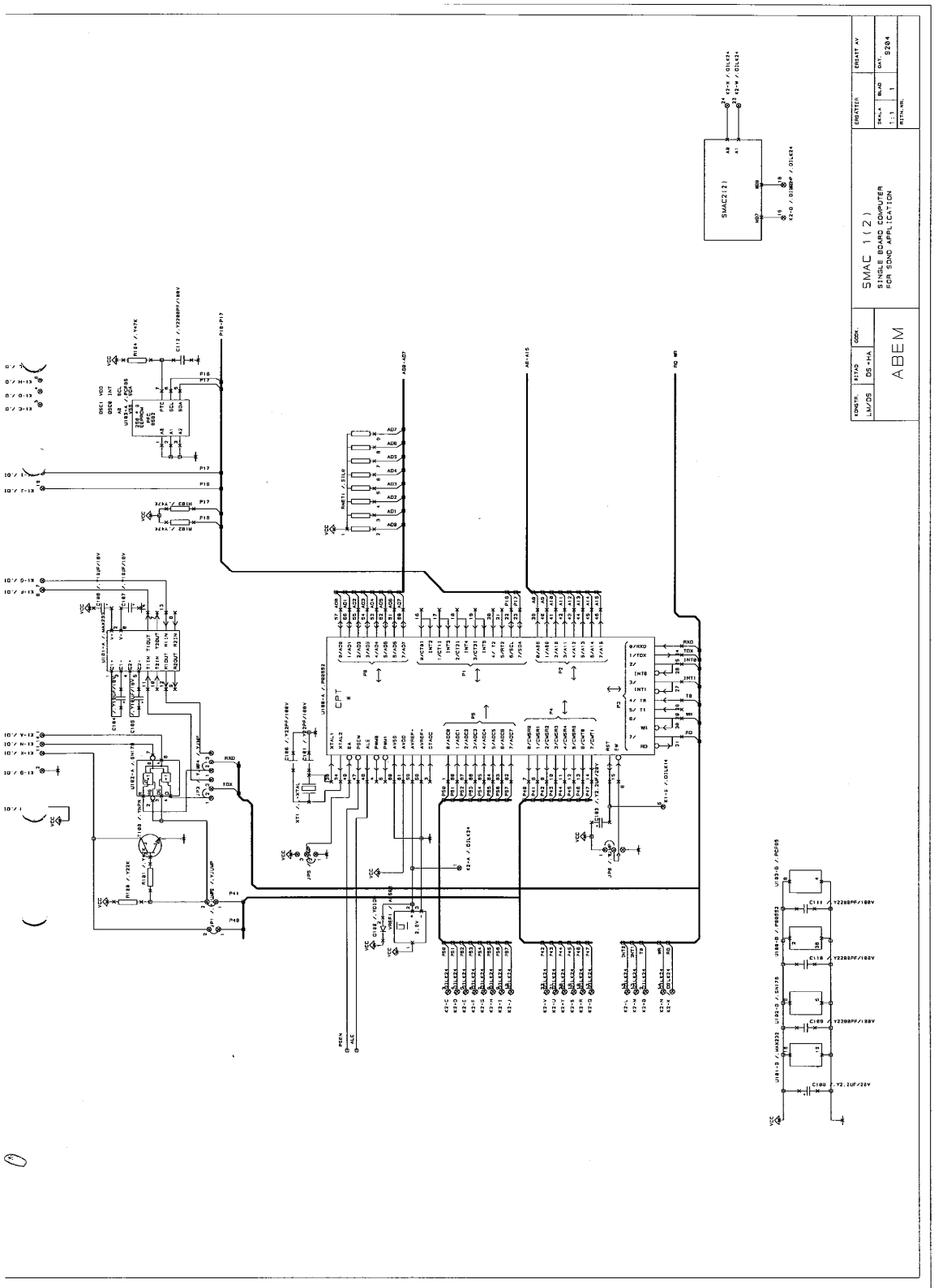


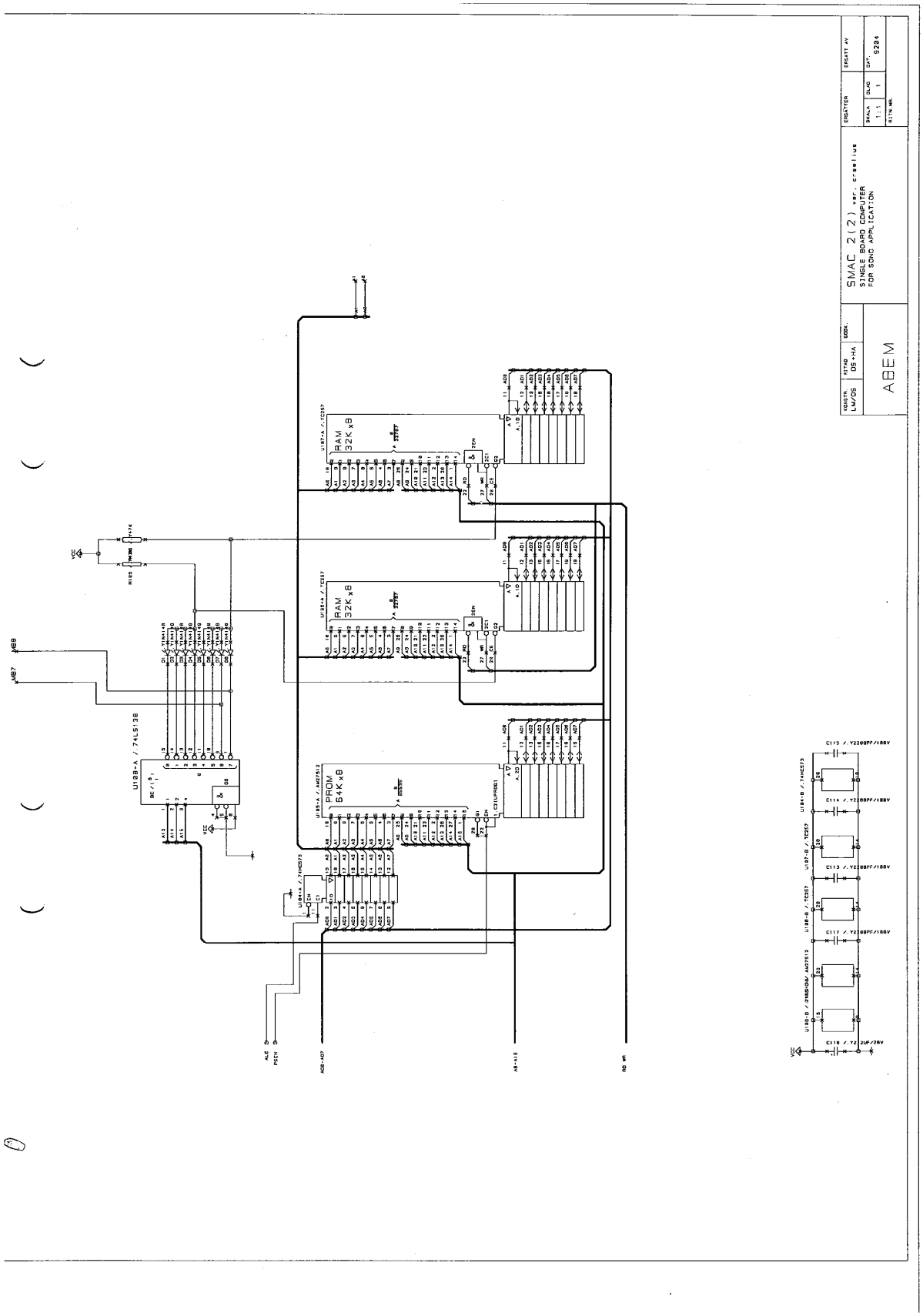
14 Appendix E: Ritningar på sondens elektronik

14.1 Inkopplingsschema sond



14.2 Schema datorkort



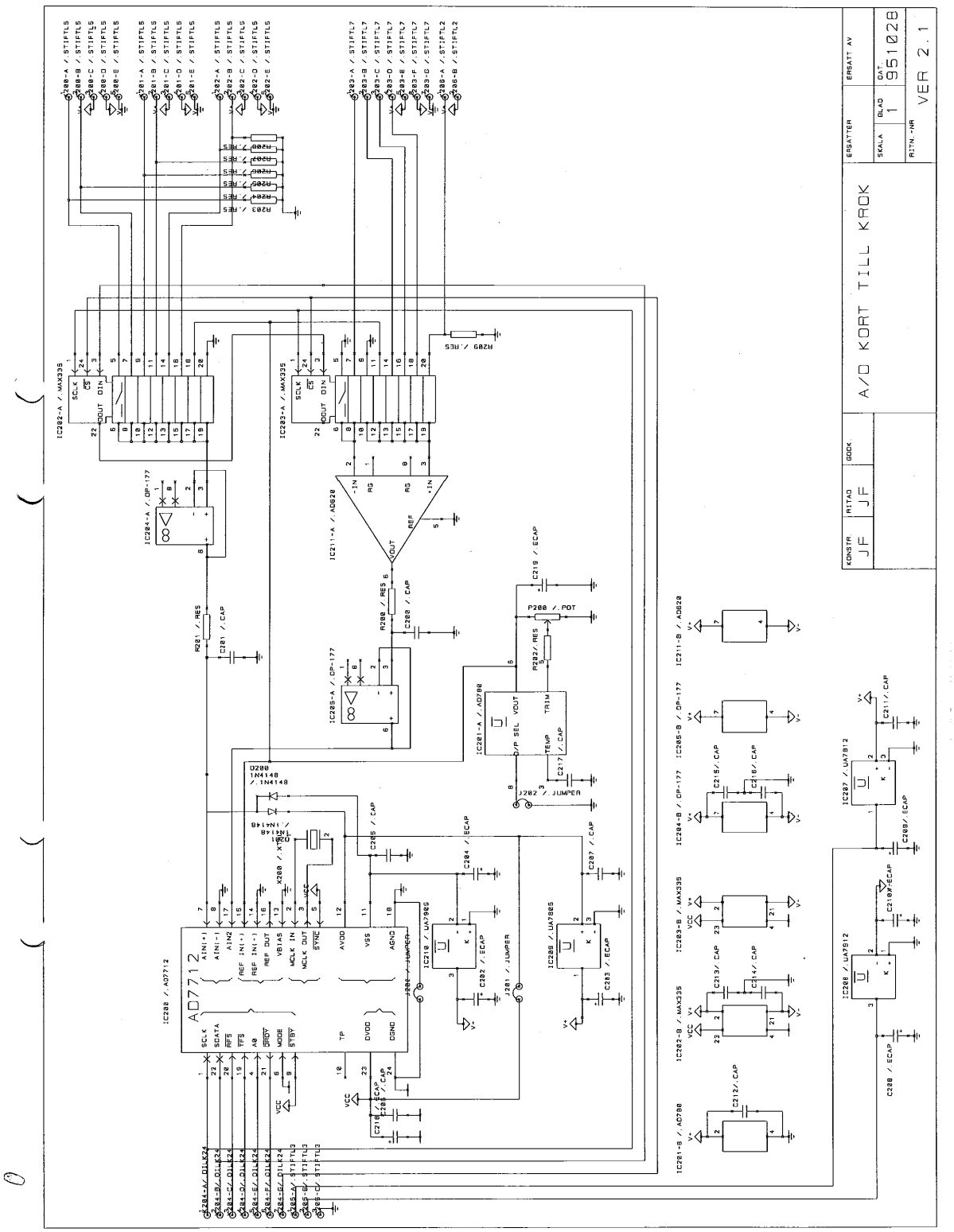

SMAC 2(2)
SINGLE BOARD COMPUTER
FOR SOND APPLICATION
ABEM
RAM 32K x8
PROM 64K x8
U108-A / 74LS138

14.3 Komponentlista datorkort

15. DATORKORT CALIPER

Beteckning	Antal	Komponent	Firma
U100	1	PCD80C552-4-16WP	MULTI. A270 372
U102	1	LTC485	ELFA 73-485-01
U104	1	74HC573AP	MULTI. A272 072
U105	1	27C256-15	ELFA 73-732-93
U106	1	TC55257BFL	GLSi Elektronik
VREF1	1	MX580LH	IE KOMPONENTER
XT1	1	11,0590 MHz	ELFA 74-505-62
T100	1	BC847B	ELFA
D100	1	BAS32	ELFA 70-300-67
C100,101,109,110, 113,114,115,117	8	22pF	ELFA
C103	1	2,2/20V Ytm.	ELFA
C108	1	10/25V	ELFA
C116	1	3,3/20V Ytm.	ELFA
R100	1	5,6 K Ytm.	ELFA
R101	1	22 K Ytm.	ELFA
RNET	1	10 K	ELFA 60-385-33
K1	1	Sockel,vir	ELFA 48-124-59
K2	1	Sockel,vir	MULTI.

14.4 Schema mätkort

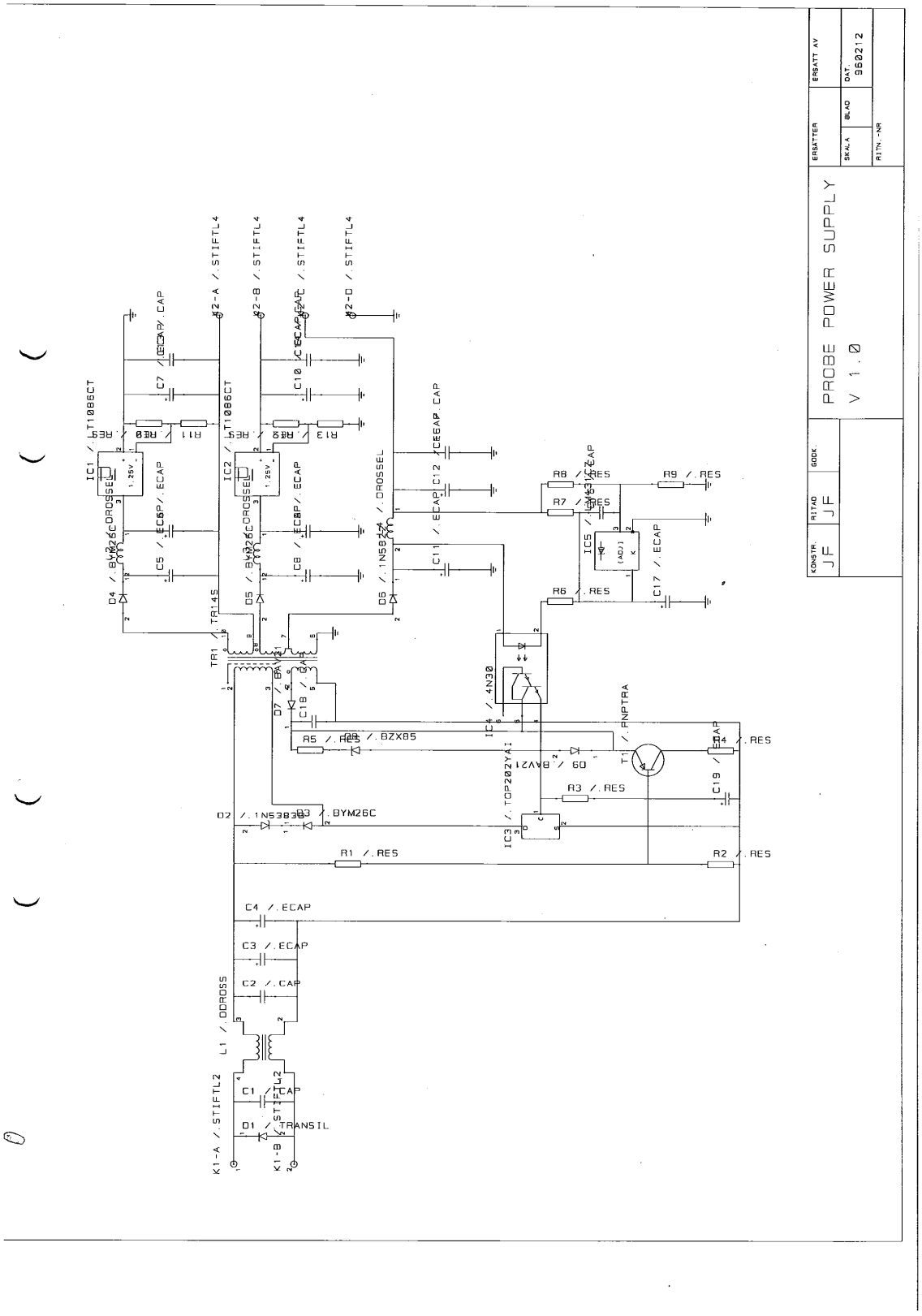


14.5 Komponentlista mätkort

A/D-OMVANDLARE

PART NO.	DESCRIPTION	QTY	NAME
AD7712	21-BIT SIGMA DELTA A/D CONVERT	1	
IC200			
OP-177	AMPLIFIER	2	
IC204-205			
XTAL	CHRYSTALL	1	X200
-	RESISTOR	10	
R200-209			
-	CAPACITOR	12	
C200-201			
C205-207			
C211-217			
-	POTENTIOMETER	1	P200
UA7812	VOLTAGE REGULATOR	1	
IC207			
UA7912	VOLTAGE REGULATOR	1	
IC208			
UA7805	VOLTAGE REGULATOR	1	
IC209			
UA7905	VOLTAGE REGULATOR	1	
IC210			
STIFTL3		1	K205
-	ELECTROLYTIC CAP	8	
C202-204			
C208-210			
C218-219			
-	JUMPER 2-PIN	3	
J200-202			
DILK24	DILKONT. 24-PIN	1	K204
1N4148	DIODE	2	
D200-201			
AD780		1	
IC201			
AD620	LOW COST INSTRUMENT AMPLIFIER	1	
IC211			
MAX335		2	
IC202-203			
STIFTL5		3	
K200-202			
STIFTL7		1	K203
STIFTL2		1	K206

14.6 Schema DC/DC omvandlare



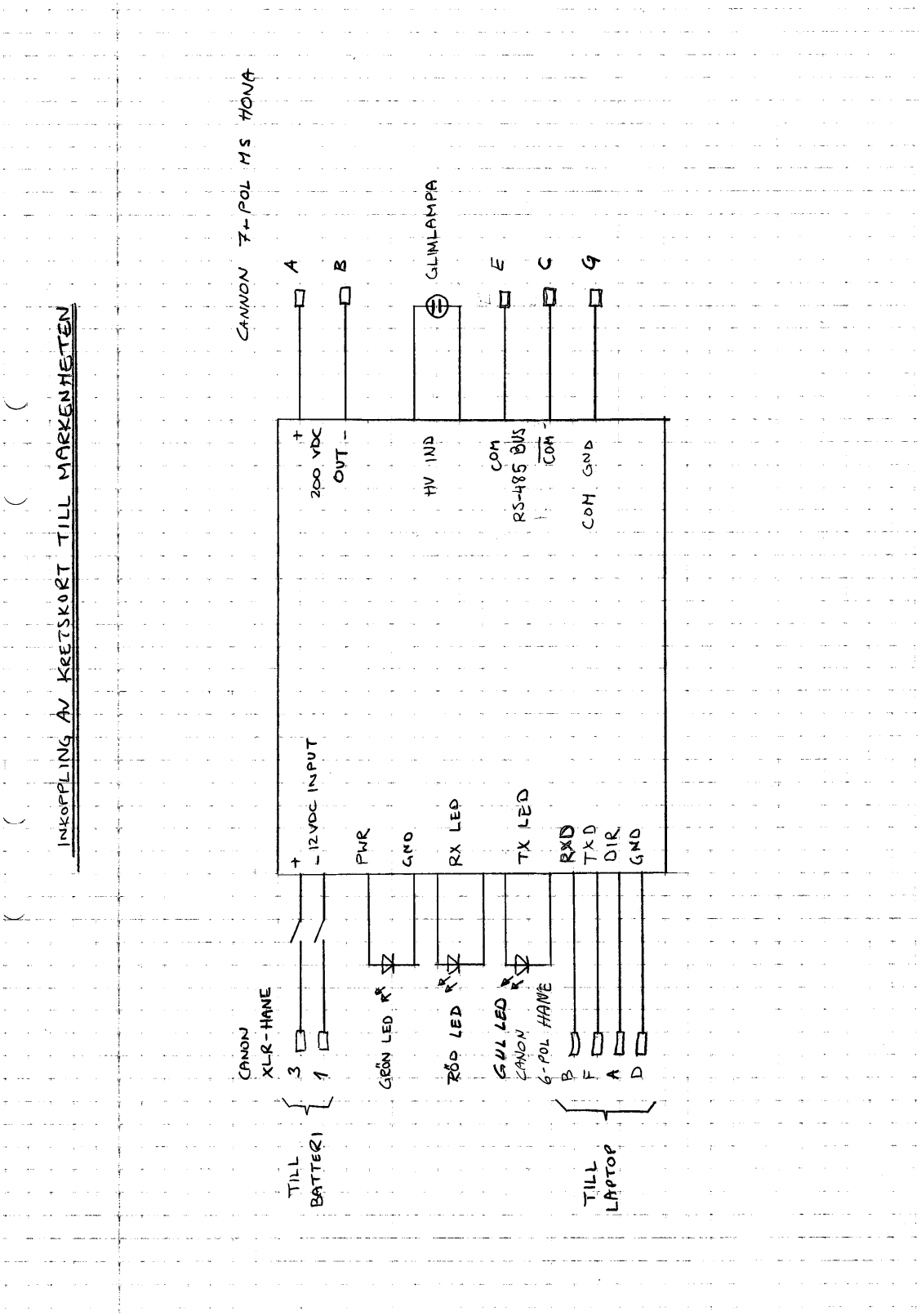
14.7 Komponentlista DC/DC omvandlare

SOND DC/DC OMVANDLARE

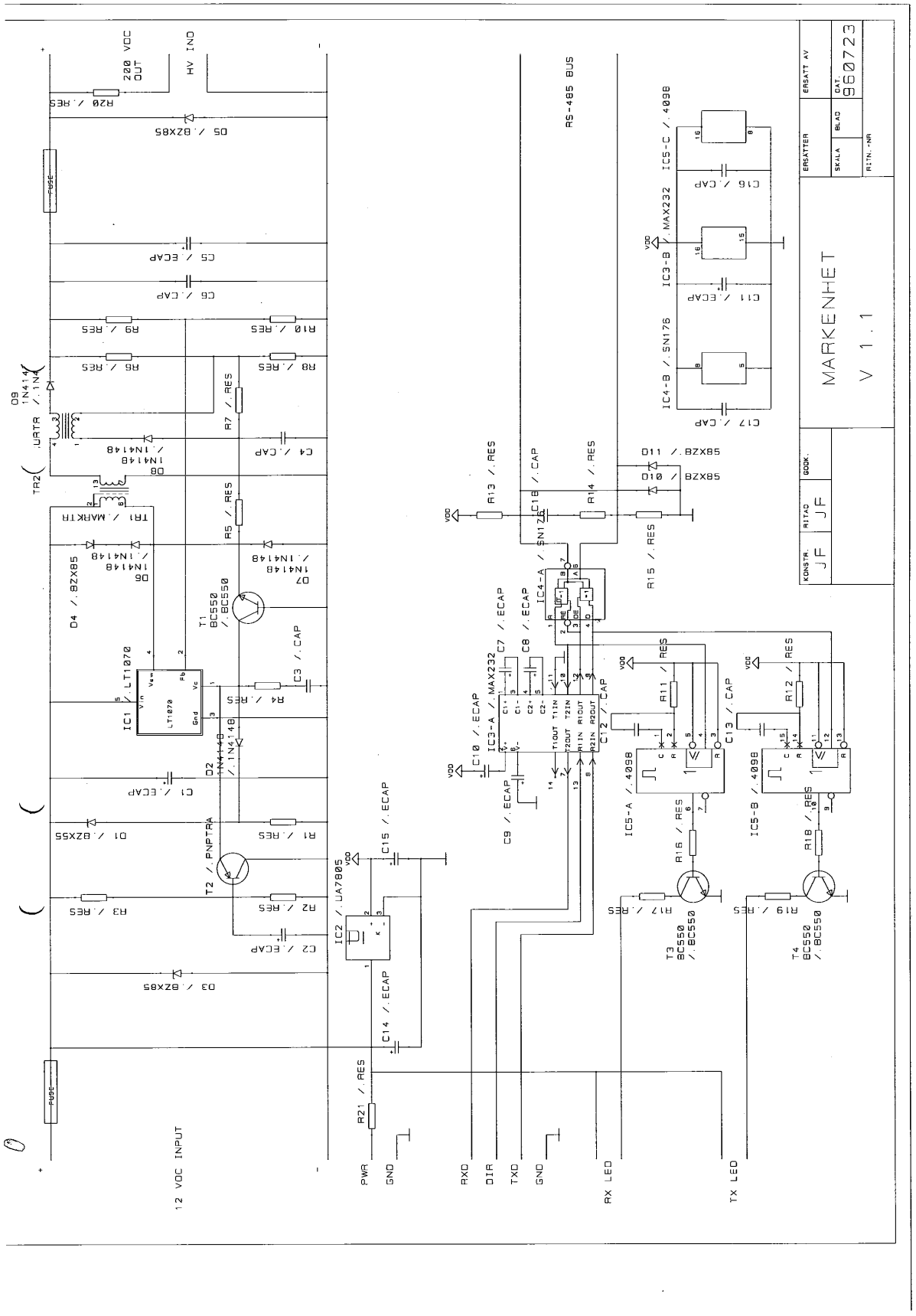
PART NO.	DESCRIPTION	QTY	NAME
TRANSIL		1	D1
-	CAPACITOR	7	C1-2
16			C13-C18
DDROSS		1	L1
-	ELECTROLYTIC CAP	12	C3-C17
12			C19
STIFTL2		1	K1
TR14S		1	TR1
1N5383B		1	D2
BYM26C		3	D3-5
1N5822		1	D6
BAV21		2	D7 D9
DROSSEL		3	L2-4
LT1086CT		2	IC1-2
TOP202YAI		1	IC3
-	RESISTOR	13	R1-13
STIFTL4		1	K2
4N30		1	IC4
LM431CZ		1	IC5
BZX85	ZENERDIODE	1	D8
-	TRANSISTOR	1	T1

15 Appendix F: Ritningar på markenhetens elektronik

15.1 Inkoppling av markenhet



15.2 Schema markenhet



15.3 Komponentlista markenhet

KOMPONENTLISTA TILL MARKENHET V 1.0

Bet.	Ant.	Namn	Lev.	Best.
IC1	1	LT1070CT	ELFA	73-099-90
IC2	1	78L05AWS	ELFA	73-091-15
IC3	1	MAX232CPE	ELFA	73-023-26
IC4	1	LTC485CN	ELFA	73-485-01
IC5	1	4098	ELFA	73-697-21
T1,T3-4	3	BC550C	ELFA	71-056-12
T2	1	BC560C	ELFA	71-056-61
D1	1	BZX55/C8V2	ELFA	70-054-32
D2	1	BAT42	ELFA	70-084-36
D3	1	1.5KE18	ELFA	70-121-64
D4	1	1.5KE33	ELFA	70-122-22
D5	1	1.5KE220	ELFA	70-123-05
D6-7	2	11DQ06	ESD	074730G
D8	1	BAV21	ELFA	70-052-18
D9	1	BYV26C	ELFA	70-028-43
D10-11	2	1.5KE12C	ELFA	70-121-15
C1	1	Lågimp. ellyt 1000 uF/25 V	ELFA	67-190-82
C2	1	Tantalkond. 1,5 uF/25 V	ELFA	67-770-07
C3	1	~~MKS2~~ Tabell 2,2µF 0,22 uF/63 V	~~ELFA~~	~~65-227-26~~
C4,16-18	3	MKS2 0,1 uF/63 V	ELFA	65-226-84
C5	1	Lågimp. ellyt 470 uF/250 V	MULTI	P110 322
C6	1	Ker. kond. 10 nF/500 V	ELFA	65-695-11
C7	1	Tantalkond. 4,7 uF/6,3 V	ELFA	67-748-06
C8	1	Tantalkond. 4,7 uF/16 V	ELFA	67-765-04
C9,14	2	Tantalkond. 10 uF/16 V	ELFA	67-767-02
C10-11,15	3	Tantalkond. 10 uF/6.3 V	ELFA	67-750-01
C12-13	2	~~Tantalkond. 1,5 uF/25 V~~ 10 ms 68µF/63V	~~ELFA~~	~~67-770-07~~
R1,13,15	3	1,5 kohm 1/4 W	ELFA	60-724-66
R2	1	22,1 kohm 1/4 W	ELFA	60-737-95
R3	1	78,7 kohm 1/4 W	ELFA	60-744-47
R4	1	~~2,74 kohm~~ 1/4 W 8,2 kohm	~~ELFA~~	~~60-727-63~~
R5	1	301 ohm 1/4 W	ELFA	60-716-74
R6	1	100 kohm 1/2 W	ELFA	60-127-36
R7	1	100 ohm 1/4 w	ELFA	60-711-46
R8	1	499 ohm 1/4 W	ELFA	60-719-30
R9	1	200 kohm 1/4 W	ELFA	60-748-84
R10	1	1,24 kohm 1/4 W	ELFA	60-723-83
R11-12	2	1 Mohm 1/4 W	ELFA	60-757-09
R14	1	150 ohm 1/4 W	ELFA	60-713-28
R16,18	2	75 kohm 1/4 W	ELFA	60-744-21
R17,19,21	3	470 ohm 1/4 W	ELFA	60-719-06
R20	1	100 kohm 1/2 W	ELFA	60-127-36
TR1	1	Transformator ETD 29 lindad enl beskrivning. består av:		
	1	Bobin ETD29	ELFA	58-760-16
	2	Clip	ELFA	58-760-24
	1	Kärnhalva 3F3	ELFA	58-760-32
	1	Kärnhalva 3F3 gap 0,35 mm	ELFA	58-760-40
TR2	1	Transformator 3F3 lindad enl beskrivning. består av:		
	1	Toroid 3F3 12,5x7,5x5 mm	ELFA	58-755-21

16 Appendix G: Sondprogram

16.1 MAIN.C

```
#include "c:\xmac\sio0.h"
#include "c:\xmac\ad77.h"
#include "c:\xmac\defdev.h"
#include "c:\xmac\com.h"
#include "c:\xmac\io552.h"

void main(void){
  unsigned long int meas,speed;
  unsigned char command;
  sio0_init(1,9600);
  ad77_init();
  EA = 1;
    while(true){                          /* programmets huvud loop */
    while(!sio0_getchar(&command));
    switch(command){

                          case update_speed_10:
        speed = S10;
                          break;

                          case update_speed_25:
        speed = S25;
                          break;

                          case update_speed_30:
        speed = S30;
                          break;

                          case update_speed_50:
        speed = S50;
                          break;

                          case update_speed_60:
        speed = S60;
                          break;

                          case calib_chan_1:
        while(!ad77_setup(speed,NORM,G1,WL,0x0105));
                          while(!ad77_get_meas(&meas));
                          com_putmeas(meas,3);
                          break;

                          case calib_chan_2:
        while(!ad77_setup(speed,NORM,G1,WL,0x0205));
                          while(!ad77_get_meas(&meas));
                          com_putmeas(meas,3);
                          break;

                          case calib_chan_3:
        while(!ad77_setup(speed,NORM,G1,WL,0x0405));
                          while(!ad77_get_meas(&meas));
                          com_putmeas(meas,3);
                          break;

        case calib_chan_4:
        while(!ad77_setup(speed,NORM,G1,WL,0x0805));
                          while(!ad77_get_meas(&meas));
                          com_putmeas(meas,3);
                          break;

        case calib_chan_5:
        while(!ad77_setup(speed,NORM,G1,WL,0x1005));
                          while(!ad77_get_meas(&meas));
                          com_putmeas(meas,3);
                          break;
```

```
case calib_chan_6:
while(!ad77_setup(speed,NORM,G1,WL,0x2005));
                  while(!ad77_get_meas(&meas));
                  com_putmeas(meas,3);
                  break;

                  case calib_chan_7:
while(!ad77_setup(speed,NORM,G1,(WL | AIN2),0x8012));
while(!ad77_get_meas(&meas));
                  com_putmeas(meas,3);
                  break;

                  case calib_chan_8:
while(!ad77_setup(speed,NORM,G1,(WL | AIN2),0x8022));
while(!ad77_get_meas(&meas));
                  com_putmeas(meas,3);
                  break;

                  case calib_chan_9:
while(!ad77_setup(speed,NORM,G1,(WL | AIN2),0x8042));
while(!ad77_get_meas(&meas));
                  com_putmeas(meas,3);
                  break;

case calib_chan_10:
while(!ad77_setup(speed,NORM,G1,(WL | AIN2),0x8081));
while(!ad77_get_meas(&meas));
                  com_putmeas(meas,3);
                  break;

case system_cal_ain1:
while(!ad77_setup(speed,SYS_CAL_ZERO,G1,WL,0x8005));
while(!ad77_setup(speed,SYS_CAL_MAX,G1,WL,0x4005));
                  break;

case system_cal_ain2:
while(!ad77_setup(speed,SYS_INT_MAX,G1,(WL | AIN2),0x8005));
                  break;

case send_meas:
while(!ad77_setup(speed,SYS_CAL_ZERO,G1,WL,0x8005));
while(!ad77_setup(speed,SYS_CAL_MAX,G1,WL,0x4005));
while(!ad77_setup(speed,NORM,G1,WL,0x0105));
while(!ad77_get_meas(&meas));
com_putmeas(meas,3);
while(!ad77_setup(speed,NORM,G1,WL,0x0205));
while(!ad77_get_meas(&meas));
com_putmeas(meas,3);
while(!ad77_setup(speed,NORM,G1,WL,0x0405));
while(!ad77_get_meas(&meas));
com_putmeas(meas,3);
while(!ad77_setup(speed,NORM,G1,WL,0x0805));
while(!ad77_get_meas(&meas));
com_putmeas(meas,3);
while(!ad77_setup(speed,NORM,G1,WL,0x1005));
while(!ad77_get_meas(&meas));
com_putmeas(meas,3);
while(!ad77_setup(speed,NORM,G1,WL,0x2005));
while(!ad77_get_meas(&meas));
com_putmeas(meas,3);
while(!ad77_setup(speed,SYS_INT_MAX,G1,(WL | AIN2),0x8005));
while(!ad77_setup(speed,NORM,G1,(WL | AIN2),0x8012));
while(!ad77_get_meas(&meas));
com_putmeas(meas,3);
while(!ad77_setup(speed,NORM,G1,(WL | AIN2),0x8022));
while(!ad77_get_meas(&meas));
com_putmeas(meas,3);
while(!ad77_setup(speed,NORM,G1,(WL | AIN2),0x8042));
while(!ad77_get_meas(&meas));
com_putmeas(meas,3);
```

```
        while(!ad77_setup(speed,NORM,G1,(WL | AIN2),0x8081));
        while(!ad77_get_meas(&meas));
        com_putmeas(meas,3);
        }
    }
}
```

16.2 SIO0.C

```
/* sio0.c detta är modulen som hanterar serieporten */


#include "c:/xmac/io552.h"
#include "c:/xmac/defdev.h"

static unsigned char rx_buf[sio0_buf_size];  /* Utrymme för mottagna tecken */
static unsigned char tx_buf[sio0_buf_size];      /* Tecken som skall sändas */
static unsigned char *rx_in = rx_buf,*rx_ut = rx_buf;    /* Mottagar pekare */
static unsigned char *tx_in = tx_buf,*tx_ut = tx_buf;        /* Sänd pekare */
static int tx_in_progress;            /* Flagga som anger om sändning pågår */
static int occupy_com_int;          /* Flagga som är en kopia av occupy_com */

/* sio0_init Initierar serieporten enligt argumenten se proc. manual.       */
/* baud anger baudrate giltiga värden är 1200, 2400, 4800, 9600, 19200      */
/* mode anger i vilken mode sio0 går giltiga värden 0, 1, 2, 3              */
/* funktionen påverkar TH1, PCON, TMOD, TCON, S0CON, TI, RI, ES0, REN, PS0. */


void sio0_init(unsigned char mode, int baud){
             switch (baud){
        case 1200 : TH1 = sio0_S1200; break;          /* Ladda TH1 med rätt */
        case 2400 : TH1 = sio0_S2400; break;            /* omladdningsvärde */
        case 4800 : TH1 = sio0_S4800; break;
        case 9600 : TH1 = sio0_S9600; break;
        case 19200 : TH1 = sio0_S19200; PCON=PCON | 0x0080; break; /* Sätt SMOD */
        default : ;
             }
    TMOD = TMOD | 0x20;               /* Starta timer1 som baudrate generator */
             TCON = TCON | 0xC0;
             switch (mode){
        case 0 : S0CON = 0x0000; break;            /* mode 0 8-bit shift reg. */
        case 1 : S0CON = 0x0040; break;                 /* mode 1 8-bit uart */
        case 2 : S0CON = 0x0080; break;      /* mode 2 parity eller multiproc */
        case 3 : S0CON = 0x00C0; break;  /* mode 3 = mode 2 med var. baudrate */
        default : ;
             }
    tx_in_progress = false;                              /* Sändning pågår ej */
             sio0_occupy_com = false;
             occupy_com_int = false;
    TI = false;                         /* Nolla avbrottsflaggan för sändning */
    RI = false;                        /* Nolla avbrottsflaggan för mottagning */
    PS0 = true;                                  /* Ge sio0 hög prioritet */
    ES0 = true;                             /* Möjligör avbrott från serieport */
    REN = true;                                  /* Möjligör mottagning */
}

/* sio0_putchar Lägger ett tecken i sänd buffert och startar sändprocedur.    */
/* Funktionen tar ett tecken av typen unsigned char som parameter.            */
/* Om tecken buffert var full returnerar funktionen false annars true.        */
/* Funktionen påverkar ej några variabler utanför modulen sio0.               */


int sio0_putchar (unsigned char byte){
             unsigned char *tmp;
    tmp = tx_in;                            /* ställ pekare på tx_in position */
    *tx_in = byte;                              /* lägg tecken i sänd buffert */
    tmp++;                                 /* flytta pekare framåt en position */
    if (tmp > &tx_buf[sio0_buf_size-1]){    /* om tx_in pekar utanför buffert */
        tmp = tx_buf;                              /* börja då om från början */
```

```
            }
    if (tmp == tx_ut){           /* Om buffert är full avbryt och returnera 0 */
             return (false);
             }
    tx_in = tmp;               /* Om tecken fick plats uppdatera tx_in pekare */
    if (!tx_in_progress){                    /* Om ej sändning starta denna. */
        tx_in_progress = true;                   /* Generera avbrott för att */
            TI = true;                        /* påbörja sändning av tecken */
        }
        return (true);
}


/* sio0_getchar läser in ett tecken från mottagningsbuffert.                  */
/* Funktionen tar en pekare av typen unsigned char som pekar på variabeln     */
/* där tecknet skall läggas.                                                  */
/* Funktionen returnerar false om mottagningsbuffer tom annars true.          */
/* Funktionen påverkar ej några variabler utanför modulen sio0.               */

int sio0_getchar(unsigned char *byte){
    if (rx_in == rx_ut){        /* Om buffert är tom avsluta och returnera 0. */
             return(false);
             }
    *byte = *rx_ut;                    /* Om buffert ej tom returnera tecken. */
    rx_ut++;                                    /* ™ka rx_in pekare en pos */
    if (rx_ut > &rx_buf[sio0_buf_size-1]){/* Om pekaren pekar utanför buffert */
        rx_ut = rx_buf;                         /* börja då om från början. */
             }
             return(true);
}

/* Avbrottsrutin för serieporten. Rutinen lägger vid mottagning tecken i      */
/* mottagningsbuffert. Vid sändning hämtas tecken från sändnings buffert.     */
/* Om motagningsbufferten har blivit full så skrivs sista tecknet över av det */
/* nya tecknet. Vid sändning läggs signalen sio0_occupy_com hög. Om signalen  */
/* sio0_com_not_occupied är låg sänds tecknet ut på porten men signalen       */
/* sio0_occupy_com hålls låg.                                                 */
/* Funktionen påverkar TI, RI, S0BUF samt sio0_occupy_com.                    */


interrupt [0x23] void sio0_int (){
             unsigned char *tmp;
    if (TI) {                               /* Behandla avbrott från sändning */
      if (tx_in_progress){                        /* Om sändning pågår starta */
            if (tx_in == tx_ut){            /* Om buffert tom stoppa sändning */
                     sio0_occupy_com = false;
                     occupy_com_int =false;
                     tx_in_progress = false;
                    }
                                      else{
                    sio0_occupy_com = true; /* starta kommunikation */
                    occupy_com_int = true;
                    S0BUF = *tx_ut;
                    tx_ut++;                        /* Öka sändpekare ett steg */
                if (tx_ut > &tx_buf[sio0_buf_size-1]){ /* Om tx_ut pekar utanför */
                        tx_ut = tx_buf;         /* buffer börja om från början. */
                                                                        }
                                                    }
                           }
        TI = false;                               /* Nollställ sändnings avbrott */
             }
             if (RI) {
        *rx_in = S0BUF;                         /* Läs in ett tecken till mott. */
        tmp = rx_in;                                                /* buffert  */
        rx_in++;                                          /* ™ka sändningspekare */
        if (rx_in > &rx_buf[sio0_buf_size-1]){ /* Om rx_in pekar utanför buffert */
            rx_in = rx_buf;                          /* börja då om från början. */
                           }
        if (rx_in == rx_ut){                     /* Om buffert full stå kvar på */
            rx_in = tmp;                                        /* samma ställe. */
                           }
```

```
        RI = false;                             /* Nollställ mottagningsavbrott */
             }
}
```

16.3 AD77.C

```
/* AD77.c program som innehåller drivrutiner för AD7712 A/D-omvandlare */

#include "c:\xmac\io552.h"
#include "c:\xmac\defdev.h"
#include "c:\xmac\adasm.h"

unsigned long slask;
static unsigned long command,measure;
static unsigned int wait = 0,newmeas = false;
static unsigned int chan,prev_command = 0x0000;

void ad77_init (void){                  /* Initiera port mot AD7712 */
  init();                               /* Assembler rutin */
  IT0 = false;                          /* Nivåkänsligt avbrott */
  EX0 = true;                           /* Möjligör avbrott från ad7712 */
    }

int ad77_setup(unsigned long int speed, unsigned char mode,unsigned char gain, unsigned char
control, unsigned int channel){
  if ( wait != 0 ) return (false);
  command = mode;                       /* Räkna ut kommando till A/D-omvandlare */
  command = (command << 3) + gain;
  command = (command << 6) + control;
  command = (command << 12) + speed;
  chan = channel;
  wait = muxdelay;
  return(true);
}

int ad77_get_kalib(unsigned long *zero,unsigned long *fullrange){
  if  (wait != 0) return(false);                /* Kolla om A/D är klar avsluta annars */
  command = (prev_command & 0x1FFFFF) | 0xC00000; /* Fixa kommando för att hämta data från
A/D-omv. */
  EX0 = false;                                  /* Stoppa avbrott från A/D-omv. */
  tx(0,command);                                /* Skicka kommando till A/D-omv */
  *zero = rx (1,3);                             /* Läs in offsetvärde */
  command = (prev_command & 0x1FFFFF) | 0xE00000; /* Fixa kommando för att hämta data från
A/D-omv. */
  tx(0,command);                                /* Skicka kommando till A/D-omv. */
  *fullrange =rx(1,3);                          /* Läs in fullskalevärde */
  EX0 = true;                                   /* Tillåt avbrott från A/D-omv. */
  return(true);
}

int ad77_put_kalib(unsigned long zero,unsigned long fullrange){
  if (wait != 0) return(false);                 /* Kolla om A/D är klar avsluta annars */
  command = (prev_command & 0x1FFFFF) | 0xC00000; /* Fixa kommando för att skicka data till
A/D-omv. */
  EX0 = false;                                  /* Stoppa avbrott från A/D-omv */
  tx(0,command);                                /* Skicka kommando till A/D-omv */
  tx(1,zero);                                   /* Skicka offsetvärde */
  command = (prev_command & 0x1FFFFF) | 0xE00000; /* Fixa kommando för att skicka data till
A/D-omv. */
  tx(0,command);                                /* Skicka kommando till A/D-omv. */
  tx(1,fullrange);                              /* Skicka fullskalevärde */
  EX0 = true;                                   /* Tillåt avbrott från A/D-omv. */
  return(true);
}

int ad77_get_meas (unsigned long *meas){
  if ((wait != 0) || (newmeas == false)) return (false);      /* Avsluta om nytt mätvärde ej
finns */
  *meas = measure;                      /* plocka upp mätvärde som avbrottsrutin hämtat */
  newmeas = false;                      /* Markera aktuellt mätvärde som gammalt*/
```

```
  return (true);
}

interrupt [ 0x03 ] void int0( void ){
  if (wait != 0) wait--;                    /* Om filtret ej hunnit svänga in räkna ner */
  if (wait == 0) newmeas = true;            /* Om filtret hunnit svänga in flagga för nytt
värde */
  if (prev_command & 0x8000){               /* Kolla om ordlängd är 16 eller 24 bitar */
    measure = rx(1,3);                      /* Hämta mätvärden från A/D omvandlare */
            }
    else {
    measure = rx(1,2);
    }
  if (wait == ( muxdelay - 1 )){
  mux(chan);                                /* Ställ in extern mux */
  tx(0,command);                            /* Ställ in A/D-omvandlare */
  prev_command = command;
 }
}
```

16.4 COM.C

```
/* com.c modul som innehåller funktioner som har hand om kommunikations     */
/* protokollet                                                               */


#include "c:/xmac/io552.h"
#include "c:/xmac/defdev.h"
#include "c:/xmac/sio0.h"

/* com_putmeas tar ett tal av typen long int och sänder ur detta ett antal  */
/* bytes så att LSB sänds sist och att dom bytes som sänds före räknas så   */
/* att LSB alltid sänds. Funktinen tar en parameter av typen long int som är*/
/* mätvärdet och ett unsigned char som anger hur många bytes räknat från    */
/* LSB som skall sändas. Funktionen påverkar modulen sio0 med funktionen    */
/* sio0_putchar                                                              */

void com_putmeas(unsigned long meas, unsigned int no_of_bytes){
             int shift,slask;
             unsigned long tmp,test;
             for (shift = (no_of_bytes-1); shift >= 0; shift--){
                          tmp = meas;
                          slask = shift * 8;
        test = tmp >> slask;                    /* skifta in rätt byte i LSB */
        while (!sio0_putchar(test));                            /* sänd LSB */
             }
}
```

16.5 ADASM.S03

```
        NAME    adasm(18)
        RSEG    CODE(0)
        PUBLIC  init
        $DEFFN  init(0,0,0,0,0,0,0,0)
        PUBLIC  mux
        $DEFFN  mux(0,0,0,0,0,0,2,0)
        PUBLIC  rx
        $DEFFN  rx(0,0,0,0,0,0,1,1)
        PUBLIC  tx
        $DEFFN  tx(0,0,0,0,0,0,4,1)
        EXTERN  ?CL8051L_4_10_L17
        RSEG    CODE

SCLK    EQU     0C3H                    ;Bitadress till P4.3
RFS     EQU     0C5H                    ;Bitadress till P4.5
SDATA   EQU     0C6H                    ;Bitadress till P4.6
TFS     EQU     0C4H                    ;Bitadress till P4.4
A0      EQU     0C2H                    ;Bitadress till P4.2
```

```
MUX     EQU     0C7H                    ;Bitadress till P4.7
IEN0    EQU     0A8H                    ;Direktadress till IEN0
RTE     EQU     0EFH                    ;Direktadress till RTE
STE     EQU     0EEH                    ;Direktadress till STE
DIS     EQU     003H                    ;Mask som används för att T2 ej skall
                                        ;använda port pinnarna

; Rutin som skickar ett ord till A/D-omvandlaren
; void tx ( bit a0, unsigned long command )
; a0 anger hur signalen A0 skall sättas
; command är det 24-bitars ord som skall överföras till A/D omvndlaren
; Rutinen sparar EA ,och nollställer den under sändning. När sändningen är
; klar återställs EA till ingångsvärdet. Rutinen påverkar signalerna A0,TFS,
; SDATA och SCLK

tx:     MOV     A0,C                    ;Sätt signalen A0 till A/D omvandlaren
        MOV     DPTR,#$LOCBX tx + 1     ;Ladda DPTR med adress till första byten
                                        ;som skall överföras
        MOV     R0,#3                   ;Ladda R0 med antal bytes som skall
                                        ;överföras till A/D omvandlaren
        MOV     R2,IEN0                 ;Hindra avbrott och spara avbrotts
        CLR     EA                      ;flaggan
        CLR     TFS                     ;Gör klart för överföring genom att
                                        ;nollställa signalen TFS
NYTT:   MOVX    A,@DPTR                 ;Lägg in första byten i ACC
        INC     DPTR                    ;Öka DPTR till nästa byte
        LCALL   SKICKA                  ;Anropa skicka för att skicka en byte
        DJNZ    R0,NYTT                 ;Nästa varv för nästa byte
        SETB    TFS                     ;Lägg TFS hög för att avsluta överföring
        MOV     IEN0,R2                 ; terställ avbrotts flaggan
        RET                             ;Avsluta funktion tx

SKICKA: MOV     R1,#8                   ;Ladda R1 med antal bitar som skall
                                        ;sändas
NYBIT:  RLC     A                       ;Rotera ACC så att C innehåller biten
                                        ;som skall sändas
        MOV     SDATA,C                 ;Lägg ut CY på SDATA signalen
        SETB    SCLK                    ;Generera en klockpuls
        CLR     SCLK                    ;
        DJNZ    R1,NYBIT                ;Nästa varv för nästa bit
        RET                             ;Avsluta subrutinen

; Rutin som tar emot ett ord från A/D omvandlaren
; unsigned long rx ( bit a0, unsigned char length )
; a0 anger hur signalen A0 till A/D omvandlaren skall sättas
; length anger hur många bytes utdataordet skall innehålla dvs 2 eller 3
; utdata som levereras innhåller 16 eller 24 bitar
; Rutinen sparar EA ,och nollställer den under mottagning. När mottagningen är
; klar återställs EA till ingångsvärdet. Rutinen påverkar signalerna A0,RFS
; SDATA och SCLK.

rx:     MOV     A0,C                    ;Sätt signalen A0 till A/D omvandlaren
        MOV     DPTR,#$LOCBX rx         ;Ladda in antal bytes som skall tas
        MOVX    A,@DPTR                 ;emot i R0
        MOV     R0,A
        MOV     R4,#0                   ;Nollställ MSB utdata
        MOV     R5,#0
        MOV     R2,IEN0                 ;Hindra avbrott och spara
        CLR     EA                      ;avbrotts flaggan
        SETB    SDATA                   ;Sätt SDATA till inport
        CLR     RFS                     ;Gör klart för mottaging genom att
                                        ;nollställa RFS
        CJNE    R0,#03,BIT16            ;Om endast 16 bitar hoppa till bit16
        LCALL   IN                      ;Hoppa till subrutin in för första byte
        MOV     R5,A                    ;Ladda in MSB om 24 bitar
BIT16:  LCALL   IN
        MOV     R6,A
        LCALL   IN                      ;Hoppa till subrutin för sista byte
        MOV     R7,A                    ;Ladda in LSB på returadress
        SETB    RFS                     ;Ettställ RFS
        MOV     IEN0,R2                 ; terställ avbrotts flaggan
```

```
        RET                             ;Avsluta funktion rx

IN:     MOV     R1,#8                   ;Ladda för att ta emot 8 bitar
NEXTB:  MOV     C,SDATA                 ;Ta in mottagen bit i C
        RLC     A                       ;Skifta in mottagen bit i ACC
        SETB    SCLK                    ;Generera klockpuls
        CLR     SCLK
        DJNZ    R1,NEXTB                ;Nästa varv för nästa bit
        RET                             ;Avsluta subrutin

; Rutin som ställer in extern mux på önskad kanal
; void mux(unsigned char chan)
; chan ställer in muxen på önskad kanal 0 till 7
; Rutinen påverkar signalerna SDATA och MUX

mux:    MOV     R2,IEN0                 ;Hindra avbrott och spara status
        CLR     EA
        CLR     SCLK                    ;Nollställ klocka till muxen
        CLR     MUX                     ;Aktivera mux
        MOV     A,R7                    ;Ladda ACC med muxinställning
        MOV     R0,#8                   ;Ladda antal bitar som skall skiftas
NY1:    RLC     A                       ;Skifta in bit i C
        MOV     SDATA,C                 ;Lägg ut bit på sdata
        SETB    SCLK                    ;Generera klockpuls
        CLR     SCLK
        DJNZ    R0,NY1                  ;Nästa varv för nästa bit
        MOV     A,R6                    ;Ladda ACC med muxinställning
        MOV     R0,#8                   ;Ladda antal bitar som skall skiftas
NY2:    RLC     A                       ;Skifta in bit i C
        MOV     SDATA,C                 ;Lägg ut bit på sdata
        SETB    SCLK                    ;Generera klockpuls
        CLR     SCLK
        DJNZ    R0,NY2                  ;Nästa varv för nästa bit
        SETB    MUX                     ;Stäng av mux
        MOV     IEN0,R2                 ;Återställ avbrott
        RET                             ;Avsluta funktion mux

; Rutin som ställer porten rätt mot A/D omvandlaren
; void init(void)
; Rutinen påverkar SCLK,RFS,TFS,MUX,RTE,STE och A0

init:   ANL     RTE,#DIS                ;Konfigurera portpinnar
        ANL     STE,#DIS                ;
        CLR     SCLK                    ;Nollställ SCLK
        SETB    RFS                     ;Ettställ RFS
        SETB    TFS                     ;Ettställ TFS
        SETB    MUX                     ;Ettställ MUX
        CLR     A0                      ;Nollställ A0
        RET                             ;Avsluta funktion init
        END
```

16.6 SIO0.H

```
/* sio0.h includefil som definierar seriekommunikationsfunktionerna i sio0.c */


/* sio0_init Initierar serieporten enligt argumenten se proc. manual.        */
/* baud anger baudrate giltiga värden är 1200, 2400, 4800, 9600, 19200       */
/* mode anger i vilken mode sio0 går giltiga värden 0, 1, 2, 3               */
/* funktionen påverkar TH1, PCON, TMOD, TCON, S0CON, TI, RI, ES0, REN, PS0.  */


extern void sio0_init(unsigned char mode,int baud);

/* sio0_putchar Lägger ett tecken i sänd buffert och startar sändprocedur.   */
/* Funktionen tar ett tecken av typen unsigned char som parameter.           */
/* Om tecken buffert var full returnerar funktionen false annars true.       */
/* Funktionen påverkar ej några variabler utanför modulen sio0.              */
```

```
extern int sio0_putchar(unsigned char byte);

/* sio0_getchar läser in ett tecken från mottagningsbuffert.                 */
/* Funktionen tar en pekare av typen unsigned char som pekar på variabeln    */
/* där tecknet skall läggas. Funktionen returnerar false om                  */
/* mottagningsbuffer tom annars true. Funktionen påverkar ej några variabler */
/* utanför modulen sio0.                                                     */


extern int sio0_getchar(unsigned char *byte);
```

16.7 AD77.H

```
/* ad77.h includefil som definerar funktionerna i ad77.c */

/* ad77_init initierar kommunikationsporten mot ad7712 */
extern void ad77_init(void);

/* ad77_setup programmerar AD7712 kontroll register */
extern int ad77_setup(unsigned long int speed, unsigned char mode,
unsigned char gain, unsigned char control, unsigned int channel);

/* ad77_get_kalib hämtar kalibreingsdata från AD7712 */
extern int ad77_get_kalib(unsigned long *zero,unsigned long *fullrange);

/* ad77_put_kalib skickar kalibreringsdata till AD7712 */
extern int ad77_put_kalib(unsigned long zero,unsigned long fullrange);

/* ad7712_get_meas hämtar kalibreringsdata till AD7712 */
extern int ad77_get_meas (unsigned long *meas);
```

16.8 DEFDEV.H

```
/* defdev.h includefil som inneh†ller definitioner f”r hela systemet
*/

#define comspeed 9600
#define commode  1

/* definitioner för bit manipulering
*/

#define true 1
#define false 0


/* definitioner för com modulen som hanterar protokollet
*/

#define com_address "DEVI0001"
#define com_adr_size 8
#define com_adr_local 0x0f


/* definitoner för sio0 gäller för kristall 11.059 MHz
*/

#define sio0_buf_size 50

#define sio0_S1200  0x00E8
#define sio0_S2400  0x00F4
#define sio0_S4800  0x00FA
#define sio0_S9600  0x00FD
#define sio0_S19200 0x00FD

#define sio0_occupy_com P4.1
```

```
/* definitioner för AD7712 drivrutinen
*/

#define ad21_sclk     P4.3
#define ad21_sdata    P4.8
#define ad21_rfs_inv  P4.5
#define ad21_tfs_inv  P4.4
#define ad21_a0       P4.2
#define ad21_mux      P4.7
#define ad21_drdy_inv P3.2
#define muxdelay 7

#define NORM          0x00
#define SELF_CAL      0x01
#define SYS_CAL_ZERO  0x02
#define SYS_CAL_MAX   0x03
#define SYS_INT_MAX   0x04
#define BKG_CAL       0x05

#define G1            0x00
#define G2            0x01
#define G4            0x02
#define G8            0x03
#define G16           0x04
#define G32           0x05
#define G64           0x06
#define G128          0x07

#define AIN2          0x20
#define PDN           0x10
#define WL            0x08
#define BO            0x02
#define UNI           0x01

#define S10           0x07d0
#define S25           0x030d
#define S30           0x028b
#define S50           0x0187
#define S60           0x0146

/* definitioner för kommandosekvensen
*/

#define update_speed_10 0x11
#define update_speed_25 0x12
#define update_speed_30 0x13
#define update_speed_50 0x14
#define update_speed_60 0x15

#define calib_chan_1    0x21
#define calib_chan_2    0x22
#define calib_chan_3    0x23
#define calib_chan_4    0x24
#define calib_chan_5    0x25
#define calib_chan_6    0x26
#define calib_chan_7    0x27
#define calib_chan_8    0x28
#define calib_chan_9    0x29
#define calib_chan_10   0x2A
#define send_meas       0x00

#define system_cal_ain1 0x30
#define system_cal_ain2 0x31
```

16.9 COM.H

```
/* com.h headerfil till com.c som definierar funktionerna till              */
/* kommunikationsprotokollet                                                */
```

```
/* com_putmeas tar ett tal av typen long int och sänder ur detta ett antal   */
/* bytes s† att LSB sänds sist och att dom bytes som sänds före räknas så    */
/* att LSB alltid sänds. Funktinen tar en parameter av typen long int som är */
/* mätvärdet och ett unsigned char som anger hur många bytes räknat från LSB */
/* som skall sändas. Funktionen p†verkar modulen sio0 med funktionen         */
/* sio0_putchar                                                              */

extern void com_putmeas(unsigned long meas, unsigned int no_of_bytes );
```

16.10 IO552.H

```
/*                      - IO552.H -

   Special header for the 80C552 processor.

   Version 4.00 [IANR]

*/

#pragma language=extended

/*==============================*/
/* Predefined SFR Byte Addresses */
/*==============================*/

sfr   P0        = 0x80;
sfr   P1        = 0x90;
sfr   P2        = 0xA0;
sfr   P3        = 0xB0;
sfr   P4        = 0xC0;
sfr   P5        = 0xC4;
sfr   PSW       = 0xD0;
sfr   ACC       = 0xE0;
sfr   B         = 0xF0;
sfr   SP        = 0x81;
sfr   DPL       = 0x82;
sfr   DPH       = 0x83;
sfr   PCON      = 0x87;
sfr   TCON      = 0x88;
sfr   TM2CON    = 0xEA;
sfr   TMOD      = 0x89;
sfr   TM2IR     = 0xCB;
sfr   TL0       = 0x8A;
sfr   TL1       = 0x8B;
sfr   TH0       = 0x8C;
sfr   TH1       = 0x8D;
sfr   TML2      = 0xEC;
sfr   TMH2      = 0xED;
sfr   CTCON     = 0xEB;
sfr   CTH3      = 0xCF;
sfr   CTH2      = 0xCE;
sfr   CTH1      = 0xCD;
sfr   CTH0      = 0xCC;
sfr   CMH2      = 0xCB;
sfr   CMH1      = 0xCA;
sfr   CMH0      = 0xC9;
sfr   CTL3      = 0xAF;
sfr   CTL2      = 0xAE;
sfr   CTL1      = 0xAD;
sfr   CTL0      = 0xAC;
sfr   CML2      = 0xAB;
sfr   CML1      = 0xAA;
sfr   CML0      = 0xA9;
sfr   IEN0      = 0xA8;
sfr   IEN1      = 0xE8;
sfr   IP0       = 0xB8;
sfr   IP1       = 0xF8;
sfr   S0CON     = 0x98;
sfr   S1CON     = 0xD8;
```

```
sfr   S0BUF     = 0x99;
sfr   S1ADR     = 0xDB;
sfr   S1DAT     = 0xDA;
sfr   S1STAT    = 0xD9;
sfr   RCAP2L    = 0xCA;
sfr   RCAP2H    = 0xCB;
sfr   TL2       = 0xCC;
sfr   TH2       = 0xCD;
sfr   T3        = 0xFF;
sfr   PWMP      = 0xFE;
sfr   PWM1      = 0xFD;
sfr   PWM0      = 0xFC;
sfr   RTE       = 0xEF;
sfr   STE       = 0xEE;
sfr   ADCON     = 0xC5;
sfr   ADCH      = 0xC6;

/*==============================*/
/* Predefined SFR Bit Addresses */
/*==============================*/

/*========PSW========*/

bit   CY        = 0xD7;
bit   AC        = 0xD6;
bit   F0        = 0xD5;
bit   RS1       = 0xD4;
bit   RS0       = 0xD3;
bit   OV        = 0xD2;
bit   F1        = 0xD1;
bit   P         = 0xD0;

/*========TCON=======*/

bit   TF1       = 0x8F;
bit   TR1       = 0x8E;
bit   TF0       = 0x8D;
bit   TR0       = 0x8C;
bit   IE1       = 0x8B;
bit   IT1       = 0x8A;
bit   IE0       = 0x89;
bit   IT0       = 0x88;

/*=======IEN0========*/

bit   EA        = 0xAF;
bit   EAD       = 0xAE;
bit   ES1       = 0xAD;
bit   ES0       = 0xAC;
bit   ET1       = 0xAB;
bit   EX1       = 0xAA;
bit   ET0       = 0xA9;
bit   EX0       = 0xA8;

/*=======IEN1========*/

bit   ET2       = 0xEF;
bit   ECM2      = 0xEE;
bit   ECM1      = 0xED;
bit   ECM0      = 0xEC;
bit   ECT3      = 0xEB;
bit   ECT2      = 0xEA;
bit   ECT1      = 0xE9;
bit   ECT0      = 0xE8;

/*========IP0========*/

bit   PAD       = 0xBE;
bit   PS1       = 0xBD;
bit   PS0       = 0xBC;
bit   PT1       = 0xBB;
```

```
bit   PX1       = 0xBA;
bit   PT0       = 0xB9;
bit   PX0       = 0xB8;

/*========IP1========*/

bit   PT2       = 0xFF;
bit   PCM2      = 0xFE;
bit   PCM1      = 0xFD;
bit   PCM0      = 0xFC;
bit   PCT3      = 0xFB;
bit   PCT2      = 0xFA;
bit   PCT1      = 0xF9;
bit   PCT0      = 0xF8;


/*========P1=========*/

bit   SDA       = 0x97;
bit   SCL       = 0x96;
bit   RT2       = 0x95;
bit   T2        = 0x94;
bit   CT3I      = 0x93;
bit   CT2I      = 0x92;
bit   CT1I      = 0x91;
bit   CT01      = 0x90;


/*========P3=========*/

bit   RD        = 0xB7;
bit   WR        = 0xB6;
bit   T1        = 0xB5;
bit   T0        = 0xB4;
bit   INT1      = 0xB3;
bit   INT0      = 0xB2;
bit   TXD       = 0xB1;
bit   RXD       = 0xB0;

/*========P4=========*/

bit   CMT1      = 0xC7;
bit   CMT0      = 0xC6;
bit   CMSR5     = 0xC5;
bit   CMSR4     = 0xC4;
bit   CMSR3     = 0xC3;
bit   CMSR2     = 0xC2;
bit   CMSR1     = 0xC1;
bit   CMSR0     = 0xC0;

/*========S0CON=======*/

bit   SM0       = 0x9F;
bit   SM1       = 0x9E;
bit   SM2       = 0x9D;
bit   REN       = 0x9C;
bit   TB8       = 0x9B;
bit   RB8       = 0x9A;
bit   TI        = 0x99;
bit   RI        = 0x98;

/*========S1CON=======*/

bit   ENS_1     = 0xDE;
bit   STA       = 0xDD;
bit   STO       = 0xDC;
bit   SI        = 0xDB;
bit   AA        = 0xDA;
bit   CR1       = 0xD9;
bit   CR0       = 0xD8;
```

```
/*=======TM2IR========*/

bit   T2OV      = 0xCF;
bit   CMI2      = 0xCE;
bit   CMI1      = 0xCD;
bit   CMI0      = 0xCC;
bit   CT3       = 0xCB;
bit   CT2       = 0xCA;
bit   CT1       = 0xC9;
bit   CT0       = 0xC8;


/*==============================*/
/* Interrupt Vector Definitions */
/*==============================*/

interrupt [0x03] void EX0_int (void);    /* External Interrupt 0 */

interrupt [0x0B] void T0_int (void);      /* Timer 0 Overflow */

interrupt [0x13] void EX1_int (void);     /* External Interrupt 1 */

interrupt [0x1B] void T1_int (void);      /* Timer 1 Overflow */

interrupt [0x23] void SIO0_int (void);    /* Serial Port UART */

interrupt [0x2B] void SIO1_int (void);    /* Serial I2C */

interrupt [0x33] void CT0_int (void);     /* T2 Capture 0 */

interrupt [0x3B] void CT1_int (void);     /* T2 Capture 1 */

interrupt [0x43] void CT2_int (void);     /* T2 Capture 2 */

interrupt [0x4B] void CT3_int (void);     /* T2 Capture 3 */

interrupt [0x53] void ADC_int (void);     /* AD Ready */

interrupt [0x5B] void CM0_int (void);     /* T2 Compare 0 */

interrupt [0x63] void CM1_int (void);     /* T2 Compare 1 */

interrupt [0x6B] void CM2_int (void);     /* T2 Compare 2 */

interrupt [0x73] void T2_int (void);      /* T2 Overflow */
```

16.11 ADASM.H

```
/* adasm.h includefil som definierar funktionerna i maskinspråksmodulen
adasm.s03 som är drivrutiner till A/D omvandlaren AD7712. Definitioner för
portpinnar och signaler ligger i källkodsfilen adasm.s03 */

/* init initierar porten mot A/D omvandlaren. Rutinen påverkar signalerna
SCLK, RFS, TFS, MUX och A0
*/

extern void init( void );

/* tx sänder ett ord om 24-bitar till A/D omvandlaren. a0 anger hur signalen A0
skall sättas dvs om man skall skriva kommando eller kalibreringsdata till
omvandlaren. command är kommandoordet som skall sändas. command är ett
32-bitars ord men det är bara dom 24 lägsta bitarna som överförs. Rutinen
påverkar signalerna A0,TFS,SDATA,SCLK,RTE,STE. Dessutom nollställs EA under
överföringen. EA återställs till ingångsvärdet när överföringen är klar
*/

extern void tx( bit a0, unsigned long command);

/* rx tar emot ett ord från A/D omvandlaren. a0 anger hur signalen A0 skall
```

```
sättas dvs om man skall läsa ett kommandoord eller om man skall läsa mätdata.
length anger hur många bytes man skall läsa dvs 2 eller 3. Rutinen returnerar
ett 32 bitars ord som antingen innehåller 2 eller 3 bytes. Rutinen påverkar
signalerna A0,RFS,SDATA,SCLK. Dessutom nollställs EA under överföringen. EA
återställs till ingångsvärdet när överföringen är klar
*/

extern unsigned long rx( bit a0, unsigned char chan);

/* mux ställer in den externa muxen på valt kanalnummer. chan anger vilken
kanal som skall väljas dvs 0 till 7. Rutinen påverkar signalerna SDATA och MUX
*/

extern void mux(unsigned int chan);
```

17 Appendix H: PC program

17.1 MAIN.C

```
#include <stdio.h>
#include <math.h>
#include <conio.h>
#include <dos.h>
#include <ctype.h>
#include "serio.h"
#define pi 3.14159265359
#define NAMN 15
main (){
int slask,error,speed,mode;
unsigned char byte1,byte2,byte3,command;
double xacc,yacc,zacc,xtemp,ytemp,ztemp,xbias,ybias,zbias,xsf,ysf,zsf,rot,dip;
double hx,hy,hz,magtemp,azi,tot,atot,depth,dX,dY,dZ,AccX,AccY,AccZ;
double depthinc;
char filnamn[NAMN];
FILE *utfil;
clrscr();
printf("Ange filnamn:");
scanf("%s",filnamn);
utfil = fopen(filnamn,"w");
fcloseall();
if (utfil == NULL){
printf("det sket sig med filen\n");
}
command = 0x00;
sercom(1,1,&error,&command);
printf("\nange samplingshastighet:");
scanf("%i",&speed);
printf("\nange djupstegning:");
scanf("%lf",&depthinc);
printf("\nange startdjup:");
scanf("%lf",&depth);
switch(speed){
case 10:
command = 0x11;
break;
case 25:
command = 0x12;
break;
case 30:
command = 0x13;
break;
case 50:
command = 0x14;
break;
case 60:
command = 0x15;
break;
default:
command = 0x11;
break;
}
sercom(1,2,&error,&command);
clrscr();
window(1,1,80,25);
dZ = 0;
dX = 0;
dY = 0;
AccZ = 0;
AccX = 0;
AccY = 0;
mode = 'N';
cprintf("Depth   Incl.   Azi.    dZ      dX      dY      AccZ    AccX    AccY");
window(1,2,80,24);
do{
```

```
window(1,25,80,25);
gotoxy(1,1);
cprintf("Kommando sänt till sonden");
clreol();
command = 0x00;
sercom(1,2,&error,&command);
sercom(1,3,&error,&byte1);
sercom(1,3,&error,&byte2);
sercom(1,3,&error,&byte3);
gotoxy(1,1);
cprintf("*--------->");
clreol();
xacc = 3.90625023283E-3*byte1;      /* räkna om till volt */
xacc += 1.5258789972E-5*byte2;
xacc += 5.96046483281E-8*byte3;
xacc = ( xacc - 0.5 ) * 6;
xacc = xacc/2.00003;                /* räkna om till mA */
sercom(1,3,&error,&byte1);
sercom(1,3,&error,&byte2);
sercom(1,3,&error,&byte3);
gotoxy(1,1);
cprintf("**-------->");
clreol();
xtemp = 3.90625023283E-3*byte1;    /* räkna om till volt */
xtemp += 1.5258789972E-5*byte2;
xtemp += 5.96046483281E-8*byte3;
xtemp = ( xtemp - 0.5 ) * 6;
xtemp = (xtemp/4.99997E-3)- 293;   /* parametrar för #8048 */
xsf=1.289853+0.013985E-2*xtemp+0.004909E-4*pow(xtemp,2)+0.000197E-6*pow(xtemp,3);
xbias=2291.3-1493E-2*xtemp-145E-4*pow(xtemp,2)+484E-6*pow(xtemp,3);
xacc=( xacc/xsf )-xbias*1E-6;
sercom(1,3,&error,&byte1);
sercom(1,3,&error,&byte2);
sercom(1,3,&error,&byte3);
gotoxy(1,1);
cprintf("***------->");
clreol();
yacc = 3.90625023283E-3*byte1;
yacc += 1.5258789972E-5*byte2;
yacc += 5.96046483281E-8*byte3;
yacc = ( yacc - 0.5 ) * 6;
yacc = yacc/1.99993;
sercom(1,3,&error,&byte1);
sercom(1,3,&error,&byte2);
sercom(1,3,&error,&byte3);
gotoxy(1,1);
cprintf("****------>");
clreol();
ytemp = 3.90625023283E-3*byte1;
ytemp += 1.5258789972E-5*byte2;
ytemp += 5.96046483281E-8*byte3;
ytemp = ( ytemp - 0.5 ) * 6;
ytemp = (ytemp/4.99912E-3)-293;   /*parametrar för #8049 */
ysf=1.295667+0.014534E-2*ytemp+0.005159E-4*pow(ytemp,2)+0.000426E-6*pow(ytemp,3);
ybias=1018.9-274E-2*ytemp-149E-4*pow(ytemp,2)+71E-6*pow(ytemp,3);
yacc=yacc/ysf-ybias*1E-6;
sercom(1,3,&error,&byte1);
sercom(1,3,&error,&byte2);
sercom(1,3,&error,&byte3);
gotoxy(1,1);
cprintf("*****----->");
clreol();
zacc = 3.90625023283E-3*byte1;
zacc += 1.5258789972E-5*byte2;
zacc += 5.96046483281E-8*byte3;
zacc = ( zacc - 0.5 ) * 6;
zacc = zacc/1.99998;
sercom(1,3,&error,&byte1);
sercom(1,3,&error,&byte2);
sercom(1,3,&error,&byte3);
gotoxy(1,1);
```

```
cprintf("******---->");
clreol();
ztemp = 3.90625023283E-3*byte1;
ztemp += 1.5258789972E-5*byte2;
ztemp += 5.96046483281E-8*byte3;
ztemp = ( ztemp - 0.5 ) * 6;
ztemp = (ztemp/5.00390E-3)-293;  /*parametrar för #8087 */
zsf=1.295048+0.012589E-2*ztemp+0.004959E-4*pow(ztemp,2)+0.001245E-6*pow(ztemp,3);
zbias=542.3+879E-2*ztemp-21E-4*pow(ztemp,2)-16E-6*pow(ztemp,3);
zacc=zacc/zsf-zbias*1E-6;
rot=57.2957795131*atan2(xacc,yacc);
dip=57.2957795131*atan(zacc/sqrt(pow(xacc,2)+pow(yacc,2)));
sercom(1,3,&error,&byte1);
sercom(1,3,&error,&byte2);
sercom(1,3,&error,&byte3);
gotoxy(1,1);
cprintf("*******--->");
clreol();
hx = 3.90625023283E-3*byte1;
hx += 1.5258789972E-5*byte2;
hx += 5.96046483281E-8*byte3;
hx = ( hx - 0.5 ) * 24;
sercom(1,3,&error,&byte1);
sercom(1,3,&error,&byte2);
sercom(1,3,&error,&byte3);
gotoxy(1,1);
cprintf("********-->");
clreol();
hy = 3.90625023283E-3*byte1;
hy += 1.5258789972E-5*byte2;
hy += 5.96046483281E-8*byte3;
hy = ( hy - 0.5 ) * 24;
sercom(1,3,&error,&byte1);
sercom(1,3,&error,&byte2);
sercom(1,3,&error,&byte3);
gotoxy(1,1);
cprintf("*********->");
clreol();
hz = 3.90625023283E-3*byte1;
hz += 1.5258789972E-5*byte2;
hz += 5.96046483281E-8*byte3;
hz = ( hz - 0.5 ) * 24;
sercom(1,3,&error,&byte1);
sercom(1,3,&error,&byte2);
sercom(1,3,&error,&byte3);
gotoxy(1,1);
cprintf("**********>");
clreol();
magtemp = 3.90625023283E-3*byte1;
magtemp += 1.5258789972E-5*byte2;
magtemp += 5.96046483281E-8*byte3;
magtemp = ( magtemp - 0.5 ) * 24;
tot = 7000*sqrt(hx*hx+hy*hy+hz*hz);
atot = sqrt(xacc*xacc+yacc*yacc+zacc*zacc);
azi = 57.2957795131*atan2(((hx*yacc-hy*xacc)*sqrt(xacc*xacc+yacc*yacc+zacc*zacc))
,(hz*(xacc*xacc+yacc*yacc)-(zacc*(hx*xacc+hy*yacc))));
azi = -azi;
if (azi < 0) azi += 360;
window(1,2,80,24);
if (mode != 'M'){
gotoxy(1,1);
delline();
}
gotoxy(1,23);
cprintf("%7.2f",depth);
gotoxy(9,23);
cprintf("%7.3f",dip);
gotoxy(17,23);
cprintf("%7.3f",azi);
gotoxy(25,23);
cprintf("%6.2f",dZ);
```

```
gotoxy(33,23);
cprintf("%6.2f",dX);
gotoxy(41,23);
cprintf("%6.2f",dY);
gotoxy(49,23);
cprintf("%6.2f",AccZ);
gotoxy(57,23);
cprintf("%6.2f",AccX);
gotoxy(65,23);
cprintf("%6.2f",AccY);
window(1,25,80,25);
gotoxy(1,1);
highvideo();
cprintf("M");
normvideo();
cprintf("ät igen ");
highvideo();
cprintf("N");
normvideo();
cprintf("ästa mätning ");
highvideo();
cprintf("A");
normvideo();
cprintf("vsluta");
clreol();
do{
mode = getch();
mode = toascii(mode);
mode = toupper(mode);
}while(!((mode == 'M') || (mode == 'N') || (mode == 'A')));
if (mode != 'M'){
utfil = fopen(filnamn,"a");
slask = fprintf(utfil,"%f\t%f\t%f\t%f\t%f\t%f\t%f\t%f\t%f\n"
,depth,dip,azi,dZ,dX,dY,AccZ,AccX,AccY);
fcloseall();
dZ = depthinc * sin(dip/57.2957795131);
dY = depthinc * cos(dip/57.2957795131) * sin(azi/57.2957795131);
dX = depthinc * cos(dip/57.2957795131) * cos(azi/57.2957795131);
AccZ += dZ;
AccY += dY;
AccX += dX;
depth += depthinc;
if (slask == -1 ){
printf("det sket sig med att skriva på filen \n");
}
}
}while ((error == 0) && (mode != 'A'));
fcloseall();
return (0);
}
```

17.2 SERIO.C

```
#include <stdio.h>
#include <conio.h>
#include <dos.h>
#include "serio.h"



#define ocw1 0x21
#define ocw2 0x20
#define com1 0x3F8
#define com2 0x2F8

#define init 1
#define tx   2
#define rx   3
```

```
#define d0 1
#define d1 2
#define d2 4
#define d3 8
#define d4 16
#define d5 32
#define d6 64
#define d7 128

void sercom( int port, int action, int *err,unsigned char *serbyte ){
             int base;
             int ret = 0;
             int rbr,thr,ier,iir,lcr,mcr,lsr,msr,scr,dll,dlh;
             if (port == 2){
                          base = com2;
                          }
             else {
                          base = com1;
             }
             rbr = base+0;
             thr = base+0;
             ier = base+1;
             iir = base+2;
             lcr = base+3;
             mcr = base+4;
             lsr = base+5;
             msr = base+6;
             scr = base+7;
             dll = base+0;
             dlh = base+1;
             switch (action){

                          case init:
                                       outp(ocw1,inp(ocw1) | d4);
                                       outp(ier,0);
                                       inp(rbr);
                                       outp(lcr,d0+d1+d7);
                                       outp(dll,12);
                                       outp(dlh,0);
                                       outp(lcr,inp(lcr) & (255-128) );
                                       outp(ier,0);
                                       outp(mcr,d3+d1);
                                       inp(rbr);
                                       outp( mcr,(inp(mcr) | d1 ) );
                          break;

                          case tx:
                                       while( ( inp(lsr) & d5 ) == 0);
                                       outp( mcr,(inp(mcr)&( 255 - d1 )));
                                       outp(thr,*serbyte);
                                       while( ( inp(lsr) & d6 ) == 1);
                                       while( ( inp(lsr) & d6 ) == 0);
                                       outp( mcr,(inp(mcr) | d1 ) );
                          break;

                          case rx:
                                       outp(mcr,d3+d1);
                                       do {
                                                    ret = inp(lsr);
                                                    if(ret & (d1+d2+d3)) {
                                                    *err = ret & 14;
                                                    }
                                                     else{
                                                    *err = 0;
                                                    }
                                                    if (ret & d0){
                                                    *serbyte = inp(rbr);
                                                    }
                                       }
                                       while(((ret & (d0+d1+d2+d3)) || kbhit()) == 0);
                          break;
```

```
                        }
                        }
```

17.3 SERIO.H

```
extern void sercom(int port,int action,int *err,unsigned char *serbyte);
```

18 Appendix I: Mätningar

18.1 Mätning av inklination i fixtur

Inställd inklination [°]	Uppmätt inklination [°]	Fel [°]
45	45,028	0,028
40	40,092	0,092
30	30,071	0,071
20	20,091	0,091
10	10,094	0,094
0	0,082	0,082
-10	-9,952	0,048
-20	-19,929	0,071
-30	-29,957	0,043
-40	-39,978	0,022
-50	-50,007	-0,007
-60	-59,987	0,013
-70	-69,982	0,018
-80	-79,985	0,015
-90	-89,937	0,063
-80	-80,002	-0,002
-70	-69,975	0,025
-60	-59,988	0,012
-50	-49,957	0,043
-40	-40,017	-0,017
-30	-29,944	0,056
-20	-19,985	0,015
-10	-9,951	0,049
0	0,045	0,045
10	10,064	0,064
20	20,029	0,029
30	30,027	0,027
40	39,982	-0,018
45	45,088	0,088
Avvikelse max		0,094
Avvikelse min		-0,018
Max- min		0,113
Medelvärde		0,040
Median		0,043
Standard avvikelse		0,033

Tabell 18-1. Mätning av inklination i fixtur.

18.2 Mätning av azimut i fixtur

Sonden monterades horisontellt (inklination 0º)

Inställd azimut [°]	0		90	
Rotation	Uppmätt azimut	Fel i azimut	Uppmätt azimut	Fel i azimut
0	0,906	0,906	90,238	0,238
90	0,112	0,112	89,348	-0,652
180	1,544	1,544	88,37	-1,63
270	-0,079	-0,079	89,158	-0,842
Maximal positiv avvikelse		1,544		0,238
Maximal negativ avvikelse		-0,079		-1,63
Max - Min		1,623		1,868

Inställd azimut [°]	180		270	
Rotation	Uppmätt azimut	Fel i azimut	Uppmätt azimut	Fel i azimut
0	178,162	-1,838	270,601	0,601
90	180,67	0,67	271,712	1,712
180	179,38	-0,62	271,526	1,526
270	180,425	0,425	270,32	0,32
Maximal positiv avvikelse		0,67		1,712
Maximal negativ avvikelse		-1,838		0,32
Max - Min		2,508		1,392

Tabell 18-2. Mätning av azimut i fixtur.

18.3 Mätning av borrhål i utvecklingslabb

Djup [m]	NER		UPP		Avvikelse	Avvikelse
	Inklination [°]	Azimut [°]	Inklination [°]	Azimut [°]	Inklination [°]	Azimut [°]
25	72,81	223,47	72,84	223,16	-0,03	0,31
30	72,76	223,19	72,79	222,87	-0,03	0,33
35	72,79	223,13	72,82	222,69	-0,03	0,44
40	72,79	222,98	72,75	221,46	0,04	1,52
45	72,83	223,10	72,77	221,81	0,06	1,29
50	72,80	223,07	72,82	223,58	-0,02	-0,51
55	72,75	223,30	72,76	223,76	-0,02	-0,46
60	72,72	223,56	72,68	222,65	0,04	0,91
65	72,67	223,76	72,71	224,43	-0,04	-0,67
70	72,68	223,72	72,71	223,16	-0,03	0,57
75	72,70	223,76	72,73	224,59	-0,03	-0,83
80	72,71	223,93	72,74	223,55	-0,03	0,38
85	72,67	223,90	72,69	223,34	-0,02	0,56
90	72,71	223,96	72,74	223,50	-0,03	0,46
95	72,75	224,20	72,76	223,54	-0,01	0,67
100	72,75	224,34	72,74	223,64	0,01	0,70
105	72,71	224,38	72,72	223,77	-0,01	0,61
110	72,68	224,45	72,68	223,95	0,00	0,50
115	72,70	224,63	72,68	224,05	0,03	0,58
120	72,72	224,61	72,68	224,22	0,04	0,38
125	72,75	224,80	72,70	224,22	0,05	0,58
130	72,67	224,68	72,63	224,19	0,04	0,49
135	72,67	224,71	72,63	224,22	0,04	0,49
140	72,75	224,72	72,70	224,15	0,05	0,57
145	72,70	224,71	72,65	224,14	0,05	0,57
150	72,64	224,76	72,59	224,06	0,05	0,70
155	72,63	225,37	72,51	223,35	0,13	2,02
160	72,64	225,38	72,52	223,47	0,12	1,91
165	72,71	225,76	72,58	223,62	0,14	2,15
170	72,67	225,66	72,55	223,60	0,13	2,06
175	72,60	226,04	72,45	223,72	0,15	2,32
180	72,60	226,06	72,46	223,83	0,13	2,23
185	72,52	226,32	72,38	224,01	0,14	2,31
190	72,35	226,17	72,24	224,12	0,11	2,05
195	72,42	225,85	72,40	225,48	0,02	0,37
200	72,44	226,13	72,39	225,53	0,04	0,60
205	72,54	226,32	72,49	225,60	0,05	0,73
210	72,59	226,50	72,54	225,88	0,05	0,62
215	72,56	226,54	72,52	225,94	0,04	0,60
220	72,61	226,22	72,59	225,96	0,01	0,26
225	72,74	226,49	72,70	225,96	0,04	0,53

230	72,74	226,57	72,70	226,01	0,04	0,55
235	72,74	226,63	72,70	226,12	0,04	0,51
240	72,73	226,75	72,68	226,17	0,05	0,58
245	72,73	226,76	72,68	226,17	0,05	0,59
250	72,71	226,65	72,66	226,10	0,05	0,56
255	72,69	226,71	72,65	226,23	0,04	0,48
260	72,65	226,79	72,61	226,31	0,04	0,48
265	72,61	226,87	72,57	226,30	0,05	0,56
270	72,56	226,85	72,55	226,37	0,01	0,48
275	72,56	226,86	72,53	226,54	0,03	0,31
280	72,56	226,95	72,54	226,65	0,03	0,30
285	72,57	227,05	72,54	226,64	0,03	0,41
290	72,57	227,19	72,55	226,71	0,02	0,48
295	72,59	227,15	72,57	226,81	0,02	0,34
300	72,60	227,16	72,57	226,79	0,02	0,37
305	72,59	227,35	72,56	226,89	0,03	0,46
310	72,53	227,08	72,51	226,80	0,02	0,29
315	72,50	227,21	72,48	226,84	0,01	0,37
320	72,46	227,21	72,45	226,75	0,01	0,46
325	72,41	227,27	72,39	226,99	0,02	0,28
330	72,40	227,48	72,39	227,19	0,01	0,29
335	72,37	227,75	72,37	227,38	0,01	0,37
340	72,33	227,59	72,33	227,51	0,00	0,08
345	72,35	227,92	72,35	227,67	0,01	0,25
350	72,34	227,95	72,34	227,96	0,00	-0,01
		Maximal positiv avvikelse			0,15	2,32
		Maximal negativ avvikelse			-0,04	-0,83
		Max - min			0,19	3,15
		Medelvärde			0,03	0,63
		Median			0,03	0,50
		Standardavvikelse			0,05	0,66

Tabell 18-3. Mätning i borrhål.

18.4 Jämförande mätning med en tidigare Boremac D2 mätning.

D2 mätning			Avvikelse Inklination NER	Avvikelse Azimut NER	Avvikelse Inklination UPP	Avvikelse Azimut UPP
Djup [m]	Inklination [°]	Azimut [°]				
30	72,4	222	-0,36	-1,19	-0,39	-0,87
40	72,4	223	-0,39	0,02	-0,35	1,54
50	72,4	223	-0,40	-0,07	-0,42	-0,58
60	72,4	223	-0,32	-0,56	-0,28	0,35
70	72,6	222	-0,08	-1,72	-0,11	-1,16
80	72,6	222	-0,11	-1,93	-0,14	-1,55
90	72,3	223	-0,41	-0,96	-0,44	-0,50
100	72,3	222	-0,45	-2,34	-0,44	-1,64
110	72,3	222	-0,38	-2,45	-0,38	-1,95
120	72,3	224	-0,42	-0,61	-0,38	-0,22
130	72,1	226	-0,57	1,32	-0,53	1,81
140	72,1	226	-0,65	1,28	-0,60	1,85
150	71,9	228	-0,74	3,24	-0,69	3,94
160	71,9	228	-0,74	2,62	-0,62	4,53
170	72	229	-0,67	3,34	-0,55	5,40
180	72	227	-0,60	0,94	-0,46	3,17
190	72	228	-0,35	1,83	-0,24	3,88
200	72	228	-0,44	1,87	-0,39	2,47
210	72	227	-0,59	0,50	-0,54	1,12
220	72,1	226	-0,51	-0,22	-0,49	0,04
230	72,1	227	-0,64	0,43	-0,60	0,99
240	72	228	-0,73	1,25	-0,68	1,83
250	72	228	-0,71	1,35	-0,66	1,90
260	72	227	-0,65	0,21	-0,61	0,69
270	72	225	-0,56	-1,85	-0,55	-1,37
280	72	225	-0,56	-1,95	-0,54	-1,65
290	72	226	-0,57	-1,19	-0,55	-0,71
300	71,9	225	-0,70	-2,16	-0,67	-1,79
310	71,9	224	-0,63	-3,08	-0,61	-2,80
320	71,9	222	-0,56	-5,21	-0,55	-4,75
330	71,9	220	-0,50	-7,48	-0,49	-7,19
340	71,9	220	-0,43	-7,59	-0,43	-7,51
350	71,8	220	-0,54	-7,95	-0,54	-7,96
	Maximal positiv avvikelse		-0,08	3,34	-0,11	5,40
	Maximal negativ avvikelse		-0,74	-7,95	-0,69	-7,96
	Max - min		0,66	11,29	0,57	13,36
	Medelvärde		-0,51	-0,92	-0,48	-0,26
	Median		-0,56	-0,56	-0,53	-0,22
	Standard avvikelse		0,16	2,87	0,15	3,23

Tabell 18-4. Jämförande mätning Boremac D2.