



ORKUSTOFNUN  
Jarðhitadeild

**SPRUNGUMÆLIR VIÐ LEIRHNJÚK**

Hefti I: Texti og viðaukar A-C

Einar Hrafnkell Haraldsson  
Sverrir Hákonarson

OS-89012/JHD-06

Mars 1989

## **SPRUNGUMÆLIR VIÐ LEIRHNJÚK**

Hefti I: Texti og viðaukar A-C

Einar Hrafnkell Haraldsson  
Sverrir Hákonarson

OS-89012/JHD-06 - B      Mars 1989

## EFNISYFIRLIT

1. INNGANGUR	5
2. MÆLIBÚNAÐUR	5
3. NOTKUN KERFISINS	7
3.1 Tengingar PC tölvunnar	7
3.2 Forritið SAFNA	7
3.3 Mæligögnin	8
3.4 Raðtengi 2	9
3.5 Prentarinn	9
3.6 Aðvörun	9
3.7 Skilaboð	10
4. STILLINGAR SÖFNUNARTÆKIS	10
4.1 Skipanir	11
4.1.1 A -- Stilla dagatal og klukku	11
4.1.2 B -- Sjá dagsetningu og klukku	11
4.1.3 D -- Sjá dagnúmer	11
4.1.4 E -- Skoða innganga í söfnunartæki	12
4.1.5 H -- Hjálp	12
4.1.6 K -- Kveikja á sendi	12
4.1.7 O -- Núllstillta geymsluhólf	13
4.1.8 M -- Sjá ýmsar stærðir	13
4.1.9 Q -- Stoppa mælingu	13
4.1.10 S -- Slökkva á sendi	14
4.1.11 T -- Tæma úr geymsluhólfi	14
4.1.12 W -- Hefja mælingu	14
4.1.13 -- Gera ekkert merkilegt (skipunin er orðabil)	14
4.1.14 % -- Fara í aflsnautt ástand	14
4.2 Skilaboð	14
5. ÚRVINNSLA	15
5.1 Forritið LESA	15
5.2 Form gagnaskráa úr sprungumæli	15
5.3 Forritið BREYTA	16
5.4 Geymsla gagnaskráa	16
6. MÆLIHLUTI	17
6.1 Tölvubretti	17
6.2 A/D bretti	18
6.3 Minnisbretti	18
6.4 Samskiptabretti	18
6.5 Stýribretti	18
6.6 Sprungumælar	18
7. SENDIHLUTI	20
7.1 Mótaldið	20
7.2 Magnarar fyrir A/D breytu	20
7.3 Spennu- og hitamælingin	20
7.4 Afl söfnunartækisins	20

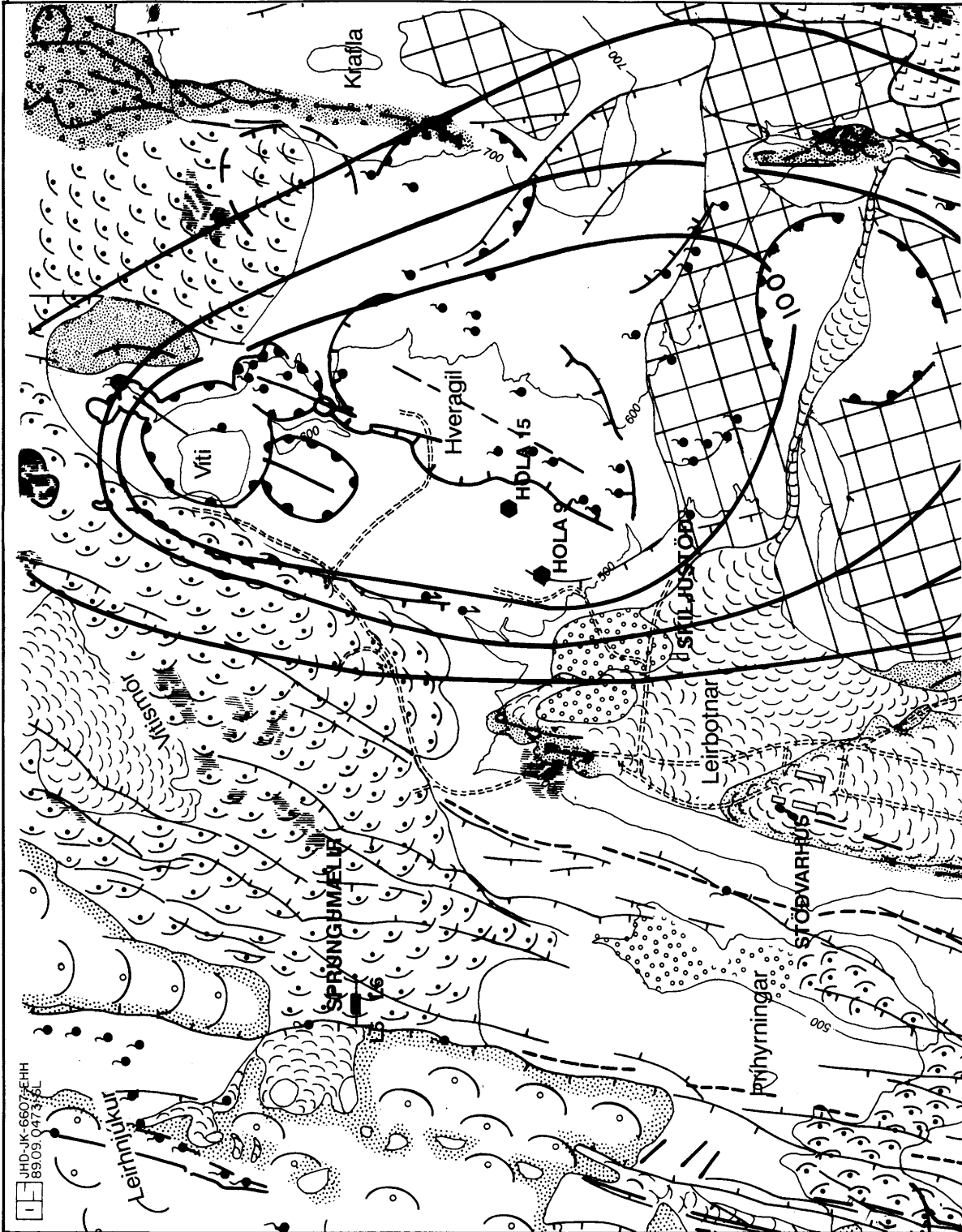
7.5 Sendirinn	21
8. MÓTTÖKUHLUTI	21
8.1 Móttakarinn	21
8.2 Mótaldið	22
8.3 A/D breytan	23

#### VIÐAUKAR

A. Teikningar	25
B. Listun á forritum í PC tölvu	37
C. Listun á forritum í söfnunartæki	53

#### Í HEFTI II

D. Örtölvubretti	
E. Samskiptabretti	
F. A/D bretti	
G. Minnisbretti	
H. Bæklingur um móttakara	
I. Bæklingur um sendi	
J. Bæklingur um loftnet	
K. Leyfisbréf frá Pósti og Síma	
L. Upplýsingar um afbrigðilegar rásir	
M. Diskettur með forritum	



Yfirlitskort vegna sprungumælis við Leirhnjúk, desember 1988. Hluti af Jarðfræðikorti af vinnslusvæðum Kröfluvirkjunar (JHD-JK-6607 KS).

## 1. INNGANGUR

Þessi skýrsla lýsir mælikerfi, sem skráir vídd tveggja sprungna við Leirhnjúk í Mývatnssveit í Suður-Þingeyjarsýslu. Mælitækin eru smíðuð hjá Orkustofnun í samvinnu við Almannavarnir ríkisins og Landsvirkjun. Við landris og landsig víkka og þrengjast sprungur við Leirhnjúk. Tækin skrá þessar hreyfingar og ef þær verða mjög hraðar, sendir tækið frá sér aðvörunarmerki, sem heyrir í stjórnsl Kröfluvirkjunar.

Mælikerfið er þannig að í sprungunum hjá Leirhnjúk er komið fyrir sprungumælum og við þá tengt söfnunartæki, sem mælir vídd þeirra og sendir upplýsingar um hana með radiósendi að stöðvarhúsi Kröfluvirkjunar. Þar er radiómóttakari sem tekur við gögnunum. Við hann er tengd PC tölvu, þar sem gögnin eru skráð á diskling, prentuð út á prentara og viðvörun gefin ef hreyfingar sprungnanna verða hraðar. Einnig er hallamælir frá Norrænu Eldfjallastöðinni tengdur við PC tölvuna, og gögnin frá honum prentuð út samhliða.

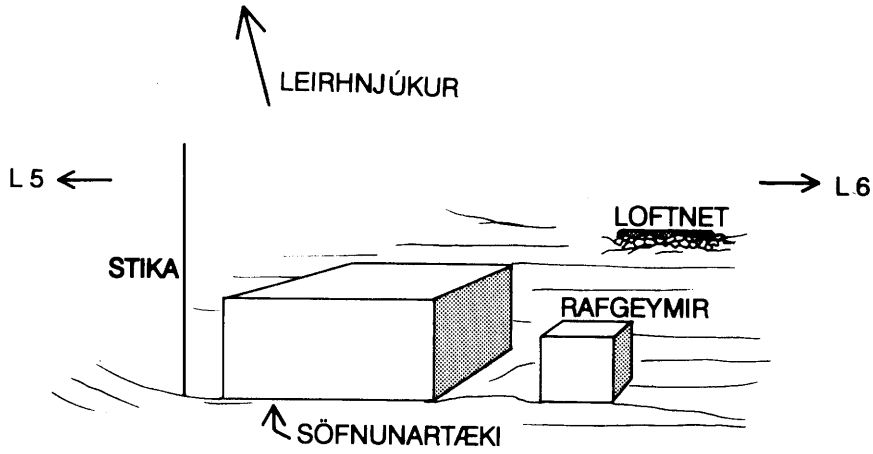
Sprungur á Kröflusvæðinu voru mældar með rennimáli á meðan goshrinur gengu yfir á árum áður. Frá því í september 1984 hafa tvær sprungur við Leirhnjúk verið mældar með tölraenu söfnunartæki. Þetta söfnunartæki verður látið ganga samhliða mælikerfinu, sem hér verður lýst.

## 2. MÆLIBÚNAÐUR

Á Kröflusvæðinu er eftirfarandi búnaður til sprungumælinga:

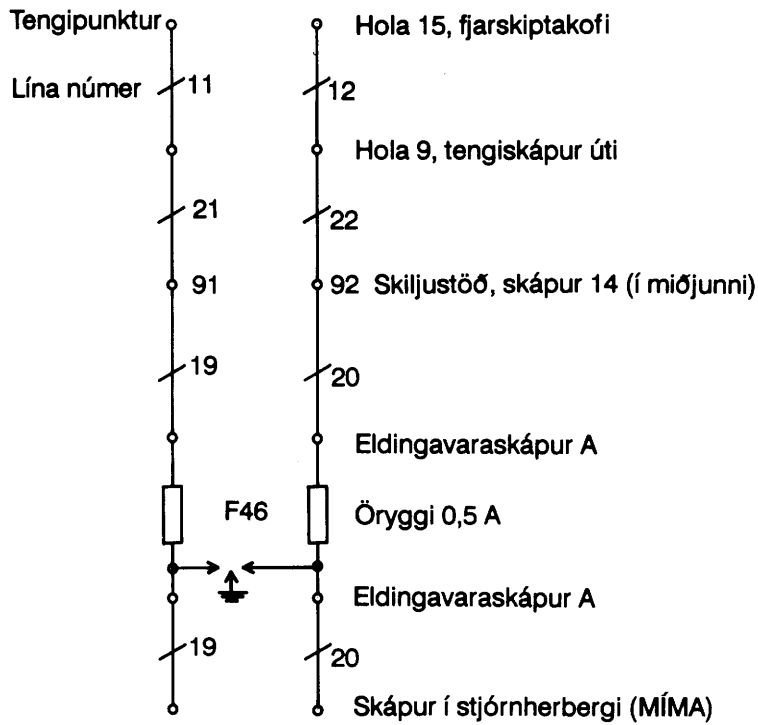
- 1) Skammt austan og sunnan Leirhnjúks eru sprungumælar í tveimur sprungum, sem eru kallaðar L5 og L6. Á milli þeirra eru um það bil 200 m. Þar mitt á milli er söfnunartæki og radiósendir í gráum kassa merkt með stíku. Rafgeymir er í litlum álkassa við hliðina á söfnunartækinu og loftnet liggur á jörðinni í svörtu plaströri, sem grjóti hefur verið hlaðið að (sjá mynd 2.1).
- 2) Í fjarskiptakofa við Holu 15 er radiómóttakari og loftnet. Loftnetið er inni í kofanum og hangir í loftinu. Móttakarinn er í plastkassa og fær afl frá netspennunni í kofanum.
- 3) Merkið frá radiómóttakaranum fer eftir fjöllínustreng frá holu 15 að holu 9 (tengiskápur úti) og þaðan í Skiljustöð (skápur 14), en þaðan í eldingavaraskáp A í Stöðvarhúsi og þaðan upp í skáp í stjórnherbergi, sem kallaður er MÍMA. Þangað liggur ljósgrár kapall (lagður undir gólfi) frá hallamælubúnaði, sem er á gólfinu í MSB-skápnum, þar sem stendur LINE 2 TO AKUREYRI. Úr MÍMA skápnum fara tveir ljósgráir kaplar eftir loftinu í norðausturhorn stjórnherbergisins.
- 4) Í norðausturhorni stjórnherbergis er tölvu af Lingó gerð og prentari af Citizen gerð.

JHD.JED.6607.EHH  
89.01.0052 0D



MYND 2.1. Afstöðumynd tækjabúnaðar

JHD.JED.6607.EHH  
89.01.0050 T



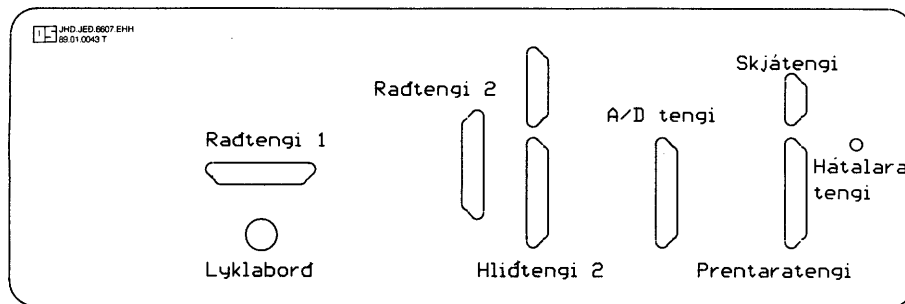
MYND 2.2 Lagnir að stöðvarhúsi

### 3. NOTKUN KERFISINS

Aðalhluti kerfisins í stöðvarhúsinu er PC tölvu, sem tekur á móti gögnum frá radiósendinum og vinnur úr þeim. Forritið sem stýrir tölvunni heitir SAFNA. Kerfið er að mestu sjálfvirkt en á mánaðarfresti þarf að skipta um diskling í tölvunni og huga að hvort nægur pappír sé í prentara. Einnig þarf að skipta um rafgeyminn í söfnunartækinu við Leirhnjúk, þegar spenna hans er kominn niður í 11,5 Volt en sú spenna sést á skjá PC tölvunnar.

#### 3.1 Tengingar PC tölvunnar

Ýmiss búnaður tengist PC tölvunni. Mynd 3.1 sýnir bakhluta tölvunnar og tengin sem á henni eru. Prentarinn tengist í prentaratengið. Í radtengi 1 tengist mótaldið frá söfnunartækinu og tengið á A/D kortinu tengist hallamælirinn. Í hátalaratengið má tengja aukahátalara fyrir aðvörunarmerkið.



MYND 3.1. Tenging á bakhlíð tölvu

#### 3.2 Forritið SAFNA

Forritið SAFNA hleðst sjálfkrafa inn þegar kveikt er á tölvunni. Til þess að það gangi þarf disklingur merktur SAFNA að vera í drifi A og forsniðinn gagnadisklingur í drifi B. Einnig þarf prentarinn að vera tengdur og tilbúinn (on line ljós kveikt). Bæði drifin þurfa að vera lokuð.

Í hvert sinn sem forritið er keyrt býr það til nýja gagnaskrá. Ef allt gengur eðlilega fyrir sig birtist nafn hennar efst á skjánum til vinstri. Nafn skrárinnar er dagsetning dagsins, ásamt aukastaf sem er til þess að greina sundur skrár, sem verða til á sama degi. Allar gagnaskrárnar enda á .KRA. Til dæmis er skráin 010888C.KRA þriðja skráin sem er búin til 1. ágúst 1988.

Gagnaskrárnar eru ekki textaskrár. Því er ekki hægt að skoða skrárnar með DOS skipuninni TYPE. Með forritinu LESA er hægt að lesa þær. Lýsing á forritinu LESA er í kafla 5.1.

Góð regla er að skipta um gagnadisklinga mánaðarlega, en þá eru venjulega búinn um 100 Kbæti af geymslurými hans. Birtist þá tilkynning á skjánum um að nauðsynlegt sé að skipta um hann. Óhætt er að skipta um disklinginn og setja tóman og forsniðinn í staðin, á meðan forritið er í gangi. Aðeins verður að athuga að opna ekki drifið á meðan forritið er að lesa eða skrifa á disklinginn en þá logar rauða ljósið á því.

Ef forritið reynir að skrifa gögn á gagnadisklinginn á meðan drifið er opið geta eftirfarandi skilaboð komið á skjáinn:



Not ready error reading drive B  
Abort, Retry, Ignore?

Ef þetta gerist er drifinu lokað og síðan svarað með því að styðja á R-takkann á lyklaborðinu (R stendur fyrir Retry).

Mynd 3.2 sýnir skjáinn þegar nokkrar mælingar eru komnar. Hver mæling tekur hluta af línu vinstra megin á skjánum. Í kafla nr. 3.3, Mæligögnin, eru tölurnar útskýrðar. Hægra megin birtist tíminn þegar síðustu gögnin bárust. Einnig sést spennan á rafgeymi söfnunartækisins og hitastig í söfnunartækinu. Þar fyrir neðan birtast ýmiss skilaboð. Þau eru talin upp og útskýrð í kaflanum 3.6, Skilaboð.

b:141188.KRA	Nafn gagnaskráar er b:141188.KRA
366 859 3241 47 2 28 126 59 3	Síðasta mæling kom 9:53 141188
366 905 3241 48 2 28 126 59 3	Staða hallamælis: 4
366 911 3241 49 2 28 126 59 3	Aflspenna: 12.6 V Hitastig: 5.9 °C
366 917 3241 50 2 28 126 59 3	
366 923 3241 51 2 28 126 59 3	Hljóðaödvörðun er á
366 829 3241 52 3 28 126 59 3	Skiptið um diskling í drifi b:
366 935 3241 53 3 28 126 59 3	
366 941 3241 54 3 28 126 59 4	
366 947 3241 55 3 28 126 59 4	
366 953 3241 56 3 28 126 59 4	

*MYND 3.2. Mælingar á tölvuskjá*

Hægt er að enda keyrslu á forritinu með því að ýta á !-takkann. Ef keyrslan er stöðvuð á einhvern annan veg t.d. með control-C, er hætt á að tölvan fari í baklás, því þá verður enn í gangi hluti af forritinu sem tekur við gögnum frá raðtengi 1. Annað hvort verður að slökkva á tölvunni eða reyna að keyra forritið aftur.

### 3.3 Mæligögnin

Þegar gögn berast frá sprungumælinum býr forritið til mælistreng úr þeim ásamt tíma og dagsetningu PC tölvunnar. Mælistrengurinn inniheldur eftirfarandi:

1. Dagsetningu samkvæmt klukku PC tölvunnar. Þetta er 16 bita heiltala. Fyrstu 6 bitarnir, að frátöldum þeim fyrsta sem er formerkisbiti, geyma árið talið frá og með 1988, næstu 4 bitarnir geyma mánuðina og síðustu 5 dagana.
2. Tíma samkvæmt klukku PC tölvunnar.
3. Dagnúmer, talið frá 1/1/1980.
4. Númer mælingar innan dagsins.
5. Stöðu sprungu L6 (rás 1).
6. Stöðu sprungu L5 (rás 2).
7. Spennu á rafgeyminum á sprungumælinum (rás 3). Deila þarf með 10 til að finna rétta spennu.
8. Útíhitastig við sprungumælinn (rás 4). Deila þarf með 10 til að finna réttan hita.

### 9. Stöðu hallamælisins.

Jafnóðum og gögnin berast, birtast þau á skjánum. Mælistrengurinn er einnig skráður á gagnadisklinginn á sama hátt.

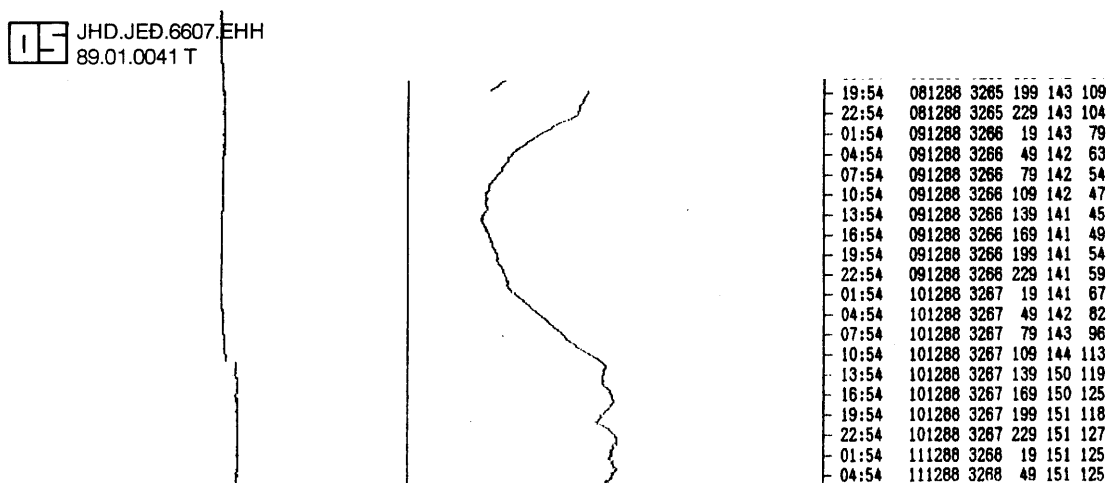
### 3.4 Raðtengi 2

Mælistrengurinn er einnig sendur á raðtengi 2. Það gefur möguleika á að tengja t.d. síma-mótald við tengið eða aðra tölvu og fá þannig aðgang að gögnunum. Sendihraða, parbita, stoppbita og orðlengd fyrir tengið má stilla í undirforritinu Stillacom2. Sjálfgefnu gildin eru 300 bitar/sec., engan pörunarbita, einn stoppbita og 8 bita orðlengd.

### 3.5 Prentarinn

Prentaraforritin eru samin fyrir Citizen LSP 10 prentara. Hann skráir feril sprungunnar L6 vinstra megin og feril hallamælisins hægra megin í hvert sinn sem gögn berast frá sprungu-mælinum. Á mynd 3.3 sést útprintunin.

Upplausn hvors ferils er 256 punktar þversum, en langsum er upplausnin 216 mælingar á tommu. Í 30. hvert skipti eru prentaðar tölulegar upplýsingar um ferlana hægra megin við þá. Fyrstu tvær tölurnar eru tími og dagsetning PC tölvunnar, næstu tvær eru dagnúmer og númer mælingar innan dagsins. Og síðast er prentuð staða ferlana. Á hverju blaði úr prentaranum sjást um 10 sólarhringar.



MYND 3.3. Útskrift á tölvuskjá

### 3.6 Aðvörun

Forritið SAFNA er hannað þannig að það geti gefið aðvörun ef hratt landris eða landsig verður. Við hverja mælingu skoðar forritið ákveðinn fjölda mælinga aftur í tímann t.d. 2 mælingar. Aðvörun er gefin ef breytingar verða á báðum sprungunum tvö mæligildi í röð og færslan verður í sömu átt. Aðvörunin er óháð því hversu mikil breytingin er.

Fjöldi mælinga, sem forritið skoðar, þarf ekki að vera sá sami fyrir báðar sprungurnar. Þar sem sprungan L6 er mun virkari en L5 er eðlilegra að skoða fleiri mælingar á henni aftur í tímann en á L5. Valdar voru 3 mælingar á L6 og 2 mælingar á L5. Hægt er að breyta þessum gildum í undirforritinu Init. Breytan FJ1 er fyrir rás 1 eða L6 en FJ2 er fyrir rás 2 eða L5.

Aðvörunin er gefin með því að birta skilaboð á skjánum um að miklar breytingar eigi sér stað. Einnig má láta tölvuna gera viðvart með hljóðaðvörun (sjá kafla 3.7 Skilaboð) .

### 3.7 Skilaboð

Hér á eftir fer upptalning og skýringar á skilaboðum sem birst geta á skjánum.

Afl á söfnunartækinu er í lágmarki !

Þegar spennan á rafgeyminum við söfnunartækið hjá Leirhnjúk er komin niður fyrir 11.5 V kemur þessi aðvörun. Þá verður að skipta um geyminn því aflíð má ekki fara niður fyrir 11.0 V. Ef það gerist hættir sendirinn að vinna.

Prentarinn er ekki virkur !

Ef prentarinn er ekki í lagi koma þessi skilaboð á skjáinn. Þá er hugsanlegt að blöðin séu búinn, hann sé ekki "on line", eða að það sé slökkt á honum. Það sést á ljósunum framan á honum.

Skriptið um diskling í drifi B !

Þessi aðvörun kemur þegar gagnadisklingurinn er að verða fullur eða orðinn fullur. Þá þarf að skipta um diskling og setja forsniðinn og tóman í staðinn.

Villa í samskiptum !

Ef lélegt samband er á milli sendisins og móttakarans og gögnin komast aðeins að hluta til skila eða ef mikið suð truflar sendinguna, þá koma þessi skilaboð.

Miklar breytingar eiga sér stað !

Ef óeðlilega miklar breytingar verða á sprungunum L5 og L6 birtist þessi aðvörun á skjánum. Forritið athugar gildi aftur í tímann og er gefin aðvörun ef breytingarnar fara út fyrir þau mörk, sem er lýst í kafla 3.6. Einnig gefur tölvan það til kynna með hljóðaðvör- un.

Hljóðaðvörun er (ekki) á !

Hægt er að ráða því hvort hljóðaðvörunin sé til staðar eða ekki, þ.e. að aðvörunin um miklar breytingar sé gefin með hljóðmerki eða ekki. Það er gert með því að styðja á a- takkann.

## 4. STILLINGAR SÖFNUNARTÆKIS

Söfnunartækið, sem er við Leirhnjúk, er hægt að stilla, sjá ástand þess og lesa gögn úr því. Það er gert með því að tengja við það PC tölvu og nota skjáhermi, sem getur tekið við gögnum svo sem forritinu H100. Hraðinn skal vera 300 bitar/sec. Snúra frá PC tölvunni er tengd við 25 pinna tengi, sem er á stuttum flatkapli, sem kemur frá samskiptabrettinu. Tengingar kapals eru eftirfarandi:

25 pinna tengi á Flatkapli		25 pinna tengi á PC tölvu
Pinni 7	-----	Pinni 7
Pinni 2	-----	Pinni 3
Pinni 3	-----	Pinni 2

Til þess að ná sambandi við söfnunartækið þarf að halda inni gráum takka, sem er á stýribrettinu. Takkann er hægt að ýta á í gegnum stóra opið, sem er á grænu grindinni, er hýsir brettin. Söfnunartækið er að öllu jöfnu í svokölluðu aflsparnaðar ástandi, en á mínútu fresti vaknar það upp og athugar hvort kominn sé tími til þess að mæla og senda gögn eða gráa takkanum sé haldið inni. Halda þarf takkanum inni allt að einni mínútu áður en tækið skynjar það. Síðan er hægt að gefa skipanir. Ef engin skipun er gefin í 25 sekúndur samfleitt, þá fer söfnunartækið í aflsparnaðar ástand og þarf að vekja það upp að nýju með gráa takkanum.

## 4.1 Skipanir

Stafróf tækisins er bundið við enska stafi og vantar því séríslensku stafina í spurningar og skilaboð frá tækinu. Í öllum tilvikum er textinn samt auðskilinn.

### 4.1.1 A -- Stilla dagatal og klukku

Þessi skipun stillir dagatal og klukku söfnunartækisins. Klukkan gengur alltaf meðan afl er á tækinu frá aðalrafhlöðu eða vararrafhlöðum. Klukkan endurstillist ekki þótt söfnunartækið sé endurstillt með RESET rofanum.

Þegar ýtt hefur verið á A, þá spyr tækið um dagsetningu og klukku. Svara þarf með einum eða tveimur tölustöfum. Ef villa er í innslættinum þarf að gefa skipunina aftur. Spurningarnar eru flestar á ensku. Að lokum sýnir tækið tímann, sem verið var að stilla.

Year: 88  
Mounth: 10  
Dagur: 19  
Hour: 11  
Minutes: 51  
Seconds: 45

1988-10-19 11:51:45

### 4.1.2 B -- Sjá dagsetningu og klukku

Þessi skipun sýnir dagsetningu og tíma eins og hann er í söfnunartækinu. Formið er:

ár-mánuður-dagur klukkustund:mínútur:sekúndur

1988-10-19 11:51:51

### 4.1.3 D -- Sjá dagnúmer

Fyrir hverja mælingu er skráð dagnúmer, sem er miðað við 1. janúar 1980. Þessi skipun reiknar út dagnúmer miðað við klukkuna í söfnunartækinu.

Dæmi:

1. janúar 1980 hefur dagnúmer 1

- 2. janúar 1980 hefur dagnúmer 2
- 1. janúar 1988 hefur dagnúmer 2923
- 1. janúar 1989 hefur dagnúmer 3289
- 1. janúar 1990 hefur dagnúmer 3654

Þegar skipunin D er gefin 19. október 1988 kemur:

Dagnumer er: 3215

#### 4.1.4 E -- Skoða innganga í söfnunartæki

Í söfnunartækinu eru 4 inngangar inn á analog/digital breytu, sem er 8 bita. Rás1 og Rás2 sýna stöðu nemanna í sprungunum. Inngangur 3, sem táknaður er með V er mælikvarði á spennu aðalrafgeymis söfnunartækisins. Margfalda þarf gildið með 0,0977 til að fá rétta spennu. Inngangur 4, sem táknaður er með t, er mælikvarði á hitastigið í °C. Hitastigið má reikna út samkvæmt formúlunni, sem gildir fyrir hitastigið -30°C til +30°C með +/- 1°C nákvæmni.

$$\text{Hiti } ^\circ\text{C} = -19,5 * \ln[(4,55 * t)/(512 - t)]$$

Dagsetning og tími er sýndur á eftir mæligildum.

Ras1 = 32, Ras2 = 35, V = 131, t = 31  
1988-10-19 11:52:2

Samkvæmt þessu er spennan  $131 * 0,0977 = 12,8$  Volt og hitastigið 23,9°C

#### 4.1.5 H -- Hjálp

Þessi skipun sýnir allar skipanir, sem hægt er að framkvæma í söfnunartækinu. Eftir að ýtt hefur verið á H kemur:

Skipanir:

- A -> Stilla dagatal og klukku
- B -> Sja klukku og dagatal
- D -> Sja dagnúmer frá 1/1/1980
- E -> Skoða innganga a a/d breytu
- H -> Hjálp
- K -> Kveikja á sendi
- O -> Nullstilli Geymsluhof
- M -> Ymsar staerdir
- Q -> Stoppa maelingu
- S -> Slokkva á sendi
- T -> Taema ur geymsluholfi
- W -> Starta maelingu
- > Gera ekkert merkilegt
- % -> Skila forritinu i aflsnaudu formi

#### 4.1.6 K -- Kveikja á sendi

Þessi skipun kveikir á sendi og er aðeins notuð við prófanir á kerfinu. Til að slökkva á sendinum er notuð skipunin S. Ef söfnunartækið er að safna, þá slekkur það á sendinum við lok næstu mælingu.

Eftir að skipunin hefur verið gefin kemur á skjáinn:

Kveikt a sendi

#### 4.1.7 O -- Núllstillta geymsluhólf

Þessi skipun er til þess að núllstillta geymsluhólf söfnunartækisins. Þessa skipun ætti aldrei að þurfa að nota. Ef hún er notuð er hætt á að mæligögn fari forgörðum.

Þegar skipunin hefur verið gefin er notandinn spurður hvort hann sé viss um að hann ætli að núllstillta geymsluhólfið. Ef svarað er með J eða j, þá er skipunin framkvæmd. Ef svarað er með einhverjum öðrum stöfum er skipunin hundsúð.

Eftir að skipunin hefur verið gefin kemur á skjáinn:

Ertu viss? j = ja [ svarað með N ]  
Hringholf var ekki nullstillt

eða

Ertu viss? j = ja [ svarað með J ]  
Hringholf var nullstillt

#### 4.1.8 M -- Sjá ýmsar stærðir

Með þessari skipun er hægt að skoða nokkrar stærðir í söfnunartækinu.

Eftir að ýtt hefur verið á M getur til dæmis komið:

-----  
Fjöldi inta er 3  
Stopp = 0  
Hringgeymsla = 4  
-----  
eða  
-----  
Fjöldi inta er 32  
Stopp = 255  
Hringgeymsla = 128  
-----

Fjöldi inta er tala, sem hækkar um 1 fyrir hverja mínútu, sem tækið er í gangi. Þessi tala er á bilinu -32768 til 32767. Stopp getur verið 0, sem merkir að tækið sé að safna eða 255, sem segir að skipunin S hafi verið framkvæmd og tækið sé ekki að mæla. Hringgeymsla sýnir hversu mörg gildi eru komin í geymsluhólf í minni söfnunartækisins. Á miðnætti er ein mæling sett í hringgeymsluna og hækkar teljarinn um 4 í hvert skipti.

#### 4.1.9 Q -- Stoppa mælingu

Með þessari skipun er hægt að stoppa mælingu. Hana ætti að öllu jöfnu ekki að nota nema þegar verið er að lesa út úr söfnunartækinu með skipuninni T. Nauðsynlegt er síðan að setja söfnun aftur í gang með skipuninni W.

Eftir að skipunin hefur verið gefin kemur á skjáinn:

Stopp - mælir ekki

#### 4.1.10 S -- Slökkva á sendi

Með þessari skipun er hægt að slökkva á sendi til dæmis eftir að kveikt hefur verið á honum með skipuninni K.

Eftir að skipunin hefur verið gefin kemur á skjáinn:

Slokkt a sendi

#### 4.1.11 T -- Tæma úr geymsluhólfi

Þessi skipun er notuð til að tæma úr geymsluhólfinu. Allt hólfið er sent út eða 2048 tölur.

Eftir að skipun hefur verið gefin kemur á skjáinn:

Utlestur ur Sprungumaeli vid Leirhnuk.

Utlesturinn hefst

1988-10-17 13:54:50

Byrjun

107 12 255 255 108 12 36 200 109 12 36 199 110 12 36 199

111 12 36 199 112 12 36 199 113 12 36 199 114 12 36 199

115 12 36 199 .....

.....

.....

116 12 36 199 117 12 36 198 118 12 36 198 119 12 36 198

120 12 36 198 121 12 36 198 122 12 36 198 123 12 36 198

124 12 36 198

Endir.

Utlestrinum lokid kl.

1988-10-17 13:58:21

#### 4.1.12 W -- Hefja mælingu

Með þessari skipun er söfnunartækið sett í mælingarástand aftur til dæmis eftir að það hefur verið stöðvað með skipuninni Q.

Eftir að skipunin hefur verið gefin kemur á skjáinn:

Mæling hefst

#### 4.1.13 -- Gera ekkert merkilegt (skipunin er orðabil)

Þessa skipun má til dæmis nota öðru hverju til þess að koma í veg fyrir að söfnunartækið fari í aflsnautt ástand.

#### 4.1.14 % -- Fara í aflsnautt ástand

Með þessari skipun er söfnunartækið sett í aflsnautt ástand og það svarar með:

\*\*\* Bless \*\*\*

## 4.2 Skilaboð

Ef gefin er skipun, sem ekki er þekkt, þá er svarað með

\*\*\* Villa Ekki rett adgerd \*\*\*

Þegar klukka söfnunartækisins er að breytst, er ekki hægt að lesa út úr henni og þessi skilaboð koma:

Reyndu aftur (Update in progress)

Söfnunartækið fer sjálft í aflsnautt ástand ef engin skipun hefur verið gefin í 25 sekúndur. Þá kemur á skjáinn:

\*\*\* Bless \*\*\*

Þegar söfnunartækið hefur verið vakið úr aflsnauðu ástandi kemur á þessi texti frá því:

Sprungumælir við Leirhnuk.  
Utgafa 1.00, 12/9/1988/EHH.  
Gefdu skipanir (H = Hjalp)

Þegar söfnunartækið hefur verið endurræst með RESET takkanum kemur þessi texti:

Orkustofnun JHD-JES  
Version 1.00, 12/9/1988/EHH

Þegar tækið er búið að mæla eina mælingu, kemur á skjáinn:

3215 1 32 35 131 31

Fremsta talan er dagnúmerið. Önnur talan er númer mælingar innan dagsins. Þriðja talan er rás 1, sú fjórða er rás 2. Fimmta talan er spennan á rafgeymi og sú sjötta er hitastigið.

## 5. ÚRVINNSLA

Mæligögnum er safnað á tveimur stöðum í þessu kerfi. Í stöðvarhúsinu er safnað gögnum sex mínútna fresti á diskling í PC tölvu. Sú skrá er ekki á venjulegu textaformi til að spara pláss. Hana þarf að lesa með forritinu LESA. Gögnum er einnig safnað í söfnunartækinu við Leirhnjúk. Þar er safnað einu sinni á sólarhring. Við útlesturinn þarf PC tölvu með útlestrarforriti.

### 5.1 Forritið LESA

Forritið þýðir gagnaskrá sem endar á .KRA yfir á textaform. Fyrst er gefið upp drifið sem skráin er á. Þá birtast allar .KRA skrárnar sem eru á því drifi. Skráin er valin með ÖR UPP, ÖR NIÐUR, og ENTER tökkunum og spyr þá forritið um nafn skrárinnar sem gögnin eiga að fara í. Sjálfgefið er nafn .KRA skrárinnar en í staðin fyrir .KRA er .DAT, en notandinn getur valið hvaða nafn sem er. Það skrifast yfir sjálfgefna nafnið.

Síðast eru valin þau gögn sem nota á úr .KRA skránni, en valmynd með þeim birtist á skjánum. Notaðir eru ENTER, ÖR NIÐUR og ÖR UPP takkarnir á sama hátt og áður. Þegar ákveðin gögn hafa verið valin birtist \* fyrir framan þau. Hægt er að afturkalla valið með því að velja þau aftur en við það hverfur \* merkið. Einnig er hægt að velja hlaupandi breytu sem byrjar á 1 og endar í fjölda mælinga. Að lokum er farið neðst í valmyndina og valið HEFJA VINNSLU GAGNA.

### 5.2 Form gagnaskráa úr sprungumæli

Söfnunartækið skráir á miðnætti gildi fyrir rás 1 og 2 ásamt dagnúmeri í geymsluhólf, sem er í innra minni þess. Gögnin eru lesin út með skipuninni T. Þau þarf að hirða upp með skjáhermi-forriti til dæmis H100, sem getur geymt allt sem kemur á skjáinn í skrá. Þessari skrá má svo



renna í gegnum forritið BREYTA, sem breytir formi gagnaskrárinnar úr söfnunartækinu. Síðan er hægt að teikna ferlana með GRAPHER eða öðru slíku forriti.

Fremst og aftast í skránni er texti með útlestrardegi og tíma. Hin eiginlegu gögn eru í streng, sem er með 2048 tölum. Hver mæling tekur 4 tölur og eru þannig 512 mælingar í skránni, sem ná yfir 512 daga eða 1 ár og 5 mánuði.

Sem dæmi um nokkur gildi eru:

108 12 36 200 109 12 37 199 110 12 38 198

Dagnúmerið fæst þannig:

$180 + 256 * 12 = 3252$

Gildið fyrir rás 1 er 36 og rás 2 er 109. Næsta dagnúmer er svo 3253 og svo framvegis. Dagur númer 1 er 1. janúar 1980.

### 5.3 Forritið BREYTA

Forritið BREYTA getur lesið frumgagnskrána úr söfnunartækinu og breytt henni í hentugt form fyrir teikniforrit svo sem GRAPHER.

Forritið BREYTA er ræst þannig:  
BREYTA innskrá útskrá

Síðan gefur forritið þrjá möguleika á framhaldi og er þriðja leiðin algengust.

Leið 1: Innskrá án texta fremst og aftast.  
Útskrá án texta fremst og aftast

Leið 2: Innskrá eins og hún kemur frá söfnunartæki.  
Útskrá með texta fremst og aftast

Leið 3: Innskrá eins og hún kemur frá söfnunartæki  
Útskrá án texta fremst og aftast

Svaraðu með 1, 2 eða 3 :

Eftir að gögnin hafa farið samkvæmt leið 3 í gegnum BREYTA, líta gildin að ofan út þannig:

1 3252 36 200  
2 3253 37 199  
3 3254 38 198

Fremst er hlaupandi tala frá 1 upp í 512 þá dagnúmer, rás 1 og rás 2.

### 5.4 Geymsla gagnaskráa

Gefa þarf gagnaskrárnafn, sem tengir það við dagsetningu. Ágætt form er til dæmis LE171088.FRU fyrir frumskrá, sem lesin er út úr söfnunartækinu þann 17. október 1988 en LE stendur fyrir Leirhnjúk. Síðan er rétt að halda sama fornaefni á skráum út úr forritinu BREYTA en bæta við öðru eftirnafni. Sem dæmi LE171088.DAT. Síðan er rétt að safna sam-  
an í eina skrá öllum .DAT skráum til frekari úrvinnslu.

## 6. MÆLIHLUTI

Söfnunartækið er örtölvukerfi byggt á C-44 tölvubrautinni. Það samanstendur af örtölvu, rauntímaklukku, minni, A/D breytu samskiptahluta og stýribretti. Kerfið fær afl frá rafgeymi eða vararafhloðum, ef hans nýtur ekki við.

### 6.1 Tölvubretti

Örtölvubrettið er af gerðinni CPU-801 frá fyrirtækinu ONSET eins og öll brettin að undanskildu stýribrettinu, sem er sett saman hjá Orkustofnun. Örtölvun sjálf er af gerðinni NSC800, sem hefur sama skipanasett og Z80 örtölvun. Þetta tölvukerfi er sérstaklega hannað með það í huga að það noti lítið afl. Hér er það látið vinna þannig að á milli mælinga er það í aflsparnaðarhætti og notar það þá innan við  $100\mu\text{A}$  og spennan á kerfinu er um 2,8 V. Í vinnslu er spennan 5,0 V og notar það um 25 mA og í sendingu notar það allt að 150 mA, þegar það hefur kveikt á radiósendinum.

Minnisrými tölvukerfisins:

0000h	-	1FFFh	Eprom með forritum á minnisbretti.
2000h	-	27FFh	Ram með vinnsluminni á minnisbretti.
2800h	-	2FFFh	Ram með hringgeymslu á minnisbretti.
3000h	-	DFFFh	Ónotað.
E000h	-	F7FFh	Eprom á tölvubretti.
F800h	-	FBFFh	Notað á tölvubretti.
FC00h	-	FC7Fh	NSC 810 ram á tölvubretti.
FC80h	-	FCFFh	Notað á tölvubretti.
FD00h	-	FD19h	NSC 810 port og timer á tölvubretti.
FD20h	-	FDFh	Notað á tölvubretti.
FE00h	-	FE3Fh	MC 146818 rauntímaklukkan á tölvubretti.
FE40h	-	FFFFh	Notað á tölvubretti.

Inn/út rými:

00h	-	AFh	Ónotað.
B0h	-	BFh	MOR-800 A/D breyta.
C0h	-	CFh	ACR-10 samskiptabretti.
D0h	-	FFh	ónotað.

Á brettinu er rauntímaklukka, 8 Kbyte EPROM, 178 byte RAM, 22 inn/út línur og aflgjafi fyrir tölvukerfið. Nokkrar inn/út línur eru notaðar fyrir rofana og til að kveikja á sendi og afli inn á magnara frá sprungumælunum. Í EPROMINU, sem hefur vistina E000hex eru einungis skipanirnar JP E003 og JP 0000 til að gera allt vistrýmið virkt samanber lýsingu á tölvubrettinu í viðauka D. RAM minnið er ekki notað hér.

Rauntímaklukkan, sem er af gerðinni MC146818, er á tölvubrettinu og vekur örtölvuna með ígripi á mínútu fresti en við sjöttu hverja mínútu er mælt. Þá er kveikt á radiósendi og gögnin send. Einnig er send spennan rafgeyma og hitastig á mælistað. Á miðnætti skráir söfnunartækið í innra minni dagnúmer og gildi á rás 1 og 2.

Aflgjafi tölvubrettisins sér örtölvukerfinu fyrir afli. Hann tekur spennu inn á bilinu 9 V til 16 V og breytir henni í 5,0 V í vinnsluhætti en 2,8 V í aflsparnaðarástandi. Hluti af afli stýribrettisins kemur beint frá rafgeyminum ásamt afli til sendisins.

Forritin fyrir örtölvukerfið er skrifað að hluta í vélarmáli örtölvunnar og að hluta í forritunarmálinu C. Þau fylgja bæði prentuð í viðauka og á disklingi. Vélarmálsforritin eru þýdd með M80 vélarmálsþýðandanum (assembler) frá Microsoft en C forritin eru þýdd með C þýðanda frá Supersoft. Þau eru tengd með L80 forritinu (linker) frá Microsoft. Forritin voru unnin á IBM PC samhæfðri tölvu og líkt eftir CP/M umhverfinu með ACCELERATE 8/16 forritinu frá Software Toolworks.

## 6.2 A/D bretti

A/D breytan er af gerðinni MOR-800 og er fjögurra rása með 12 bita upplausn. Hér eru aðeins notaðir 8 bitar vegna þess að merkið frá sprungumælunum sveiflast innan þekktra marka og 8 bita upplausn er alveg nóg og sparar pláss í geymslu mæligagna. Innmerkið er á bilinu 0 til 2,5 V og lestur hvarrárs rásar tekur nálægt 10 ms. Á brettinu er hægt að hafa rauntíma-klukku og minni en ekki er þörf á því hér. Örtölvan kveikir á brettinu rétt á meðan mælt er til að spara afl.

## 6.3 Minnisbretti

Minnisbrettið er af gerðinni MEM-88. Það er með 8 sökkla fyrir minniskubba af gerðinni 27C16 EPROM eða 6116 RAM, sem hver um sig er 2 kbyte. Vist brettisins byrjar í 0000. Á því er 6 kbyte af EPROM, sem geyma stýriforrit kerfisins og 4 kbyte af RAM, sem eru notuð fyrir breytur og hringgeymslu.

## 6.4 Samskiptabretti

Samskiptabrettið er af gerðinni ACR-10. Það hefur eina RS-232 rás, sem getur verið á hraðanum frá 50 til 19200 baud. Hraðinn, sem er valinn hér er 300 bitar/sec vegna þess að mótdaldið vinnur á þeim hraða. Hann er valinn með rofum á brettinu og eiga þeir að vera frá 1 til 4: OPINN OPINN LOKAÐUR OPINN. Tölvan kveikir og slekkur á þessu bretti eftir þörfum til að spara afl.

## 6.5 Stýribretti

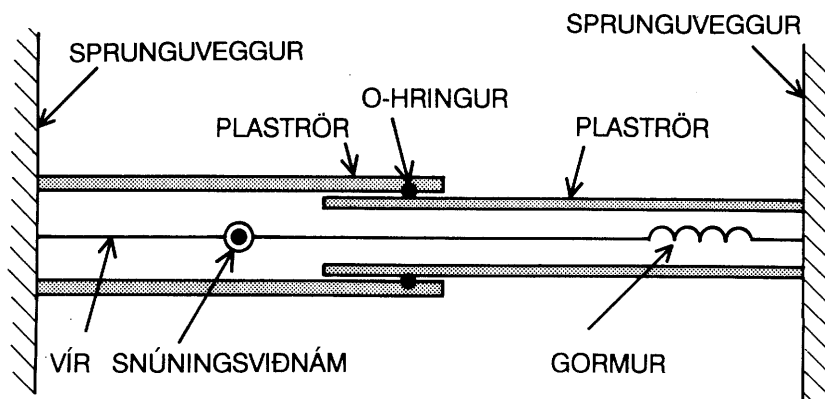
Á brettinu eru rofar og ljós sem tengjast stýringu á söfnunar-tækinu. Fimm línur liggja frá örtölvubrettinu inn á tengi TE101, og tengjast pinnum 1, 2, 3, 4 og 22. Við pinna 3 og 4 eru tengdir rofar SW104 og SW105. SW104 segir til um hvort mæling skuli gerð á 1 eða 6 mínútna fresti. Staða 1 á rofanum þýðir mælingu á 6 mínútna fresti en staða 2 þýðir mælingu á 1 mínútna fresti. Rofi SW105 er grár þrýstihnappur og er hann notaður til þess að komast í samband við söfnunartækið þegar það er í aflsnauðu ástandi.

Við merkin á pinnum 1 og 2 eru tengdar ljósdíóður. Merkið á pinna 1 sýnir þegar 1 mínútu ígrip verður. Merkið á pinna 2 kveikir á mótdaldi og sendi og merkið á pinna 22 kveikir á aðgerðarmögnunum, sem magna merkið frá sprungumælunum.

## 6.6 Sprungumælar

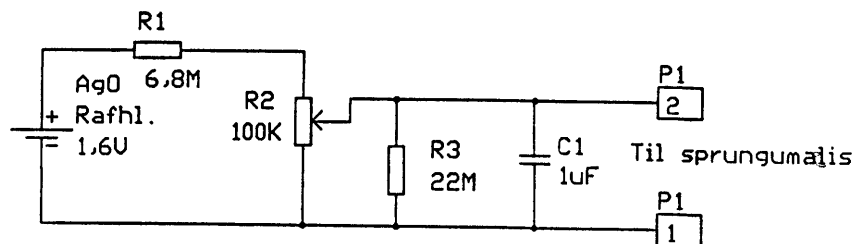
Sprungumælarnir eru festir í vegg sprunganna. Þegar þær hreyfast dragast þeir sundur eða ýtast saman. Sprungumælunum er nánar lýst á mynd 6.1. Neminn er vatnsheldur og rakadrægt þurrkefni, kísilgel, er haft inni í þeim. Vírinn er úr efni, sem hefur lítinn þenslustuðul sem fall af hitastigi. Ummálið á snúningsviðnáminu er 11,35 mm og ætla má að sprungan L6 falli saman um 0,1 mm á 6 mínútum við skyndilegt landsig.

JHD.JED.6607.EHH  
89.01.0051 0D



MYND 6.1 Nemi í sprungu

JHD.JED.6607.EHH  
89.01.0042 T



MYND 6.2 Tengingar í nema

Inni í nemanum er bretti, sem breytir viðnámsgildinu frá snúningsviðnáminu í spennu. Það er sýnt á teikningu 6.2

Rafhlaðan er af silfuroxíð, gerð til dæmis Union Carbide 355, sem er 210 mAh. Ástæðan til þess að nota silfuroxíð rafhlöðu er að spenna hennar er nánast fasti óháð því hvað hefur verið gengið á orkuforða hennar. Spenna hennar fellur hins vegar mjög hratt, þegar hún er byrjuð að falla á annað borð. Þessa rafhlöðu þarf að skipta um á hverju ári.

Viðnámið R2 er snúningsviðnám, sem snýst þegar sprungan hreyfist. Öxullinn í viðnáminu getur snúist hring eftir hring þannig að viðnámsgildið breytist úr 100 kohm í 0 ohm og öfugt, þegar öxullinn hefur snúist 360° miðað við núllstöðu viðnámsins. Þéttirinn C1 er til þess að halda spennunni til sprungumælis stífri, þegar mælt er. Einn hringur á snúningsviðnáminu svarar til færslu um 11,35 mm. Spennan út úr nemanum er á bilinu frá 0 V til 23,2 mV ef rafhlaðan er nákvæmlega 1,6 V.

## 7. SENDIHLUTI

Mótaldið, magnarar fyrir A/D breytuna og stýring fyrir söfnunartækið eru smíðuð á sama brettið. Lýsing á þessum hlutum fer hér á eftir ásamt lýsingu á sendinum. Teikning nr. 1 í viðauka, sem er á tveimur blöðum, er af brettinu.

### 7.1 Mótaldið

Mótaldið fyrir móttakarann er mjög svipað og fyrir sendinn. Það er á teikningu 1 í viðauka. Munurinn er aðeins sá að sendihlutinn TxD og TxA er tengdur en ekki móttökuhlutinn, RxD og RxA. TxD merkið fer á sama hátt inn á RS232 aðlögunarrásina LT1080 og þaðan inn á pinna 14 á tengi TE101.

Mótaldið þarf að vera í frumástandi (orginate mode), þar sem að móttökumótaldið er í svarástandi (answer mode). Rofarnir SW101 og SW102 eiga báðir að vera í stöðu 0, samkvæmt töflu 7.1. (sjá einnig teikningu 1 í viðauka)

Aflið fyrir sendinn og mótaldið kemur frá 12 V rafgeymi og tengist inn á tengi TE102, +12 V inn á pinna 2 og jörð inn á pinna 1. Til þess að söfnunartækið geti kveikt og slökkt þegar senda á gögn, er aflið tengt í gegn um segulrofa SW103. Stýrimerki á pinna 2 á tengi TE101 stjórna stöðu segulrofans.

Teikning nr. 5 í viðauka er af tengingu aflsins fyrir söfnunartækið.

### 7.2 Magnarar fyrir A/D breytu

Aðgerðarmagnararnir eru af tegundinni OP22. Þeir eru til þess að magna upp merkið frá sprungumælunum, þannig að heildarsvið þeirra, 25 mV fylli upp í heildarsvið A/D breytunnar sem er 2,5 V. Því þarf mögnunin að vera 100. Með hliðtengingu 1 MOhm og 100KOhm í afturverkunarrás aðgerðarmagnarans og 1KOhm viðnámi til jarðar, fæst 100,01 mögnun, er gefur óverulega skekkju. 10 KOhm stillanleg viðnám er tengd inn á pinna 1 og 5 á mögnurunum. Með viðnámunum má núllstilla þá. ICL7660 sér mögnurunum fyrir negatífri spennu. Hún býr til -4,5 V spennu úr +5 V. Hægt er að kveikja og slökkva á mögnurunum með stýrimerki inn á pinna 22 á tengi TE101.

### 7.3 Spennu- og hitamælingin

Söfnunartækið mælir spennuna á aflinu og útihitastig í hvert sinn sem það mælir sprungurnar. Viðnám R106, R107 og stillanlega viðnámið R108 deila aflspennunni þannig að út fæst einn tíundi spennunnar. Þessi spenna er tengd inn á A3 á A/D breytunni.

Hitaháð viðnám, R105, (thermistor) er notað til þess að mæla hitastig. Það er tengt í reglaða 5,0 V spennu ásamt 47 KOhm viðnámi, R104. Ólínulegt samband er milli hitastigs og viðnámsins. En í forritinu SAFNA er leiðrétt fyrir því. Þetta gildi er tengt inn á A4 á A/D breytunni.

### 7.4 Afl söfnunartækisins

Söfnunartækið notar 12 V spennu, sem tengist í tengi á kassanum. Teikning nr. 5 í viðauka sýnir tengingu aflsins. Díóða D501 tryggir að ekki fari rangt póluð spenna inn á söfnunartækið. Ef pólunum er víxlað leiðir díóðan og öryggið sem er 1 Amper fer.

Tvær 9 V hliðtengdar rafhlöður sjá söfnunartækinu fyrir varaafli. Ef spennan á ytri aflgjafa fellur niður fyrir 9 V taka rafhlöðurnar við. Þær halda þó söfnunartækinu einungis gangandi því sendirinn fær ekki afl frá þeim. Díóða D502 kemur í veg fyrir það. Ef það gerist skilar

söfnunartækið engum mælingum frá sér. Þess í stað safnar það gögnunum í sitt eigið minni. Því er lýst í kafla 3, STILLINGAR SÖFNUNARTÆKISINS og í kafla 4, ÚRVINNSLA.

## 7.5 Sendirinn

Sendirinn er af gerðinni Monitron, T45F-1 og vinnur á 469,5 MHz. RF aflíð er 150 mW. Á honum eru tvö tengi. Annað er RF tengi fyrir loftnetið og hitt er Bendix PT02E-8-4P og tengist það mótaldinu. Tenging kapalsins milli sendis og mótalds er samkvæmt töflu 7.1.

TAFLA 7.1 Tenging sendis og mótalds

TE102	Kapall	Bendix	Merki
1	gul/grænn	D	jörð
3	brúnn	A	+ 12 V
4	blár	B	útmerki

Toppgildið á merkinu frá mótaldinu inn á sendinn þarf að vera 1 Vpp. Hægt er að stilla styrkinn á merkinu með viðnámi á pinna 20 (TLA) á mótaldsrásinni MC145442 og sýnir tafla 3 í viðauka styrkinn fyrir mismunandi viðnámsgildi. Þar er miðað við 600 Ohm álag. En inngangsviðnámið í sendinn er um 100 KOhm og fæst 1 Vpp með óendanlegu TLA viðnámi. Því er pinninn hafður ótengdur.

Ýtalegri upplýsingar um sendinn er að finna í viðauka.

## 8. MÓTTÖKUHLUTI

### 8.1 Móttakarinn

Móttakarinn er af gerðinni Monitron model R45F og vinnur á 469,5 MHz. Á honum er hægt að stilla bæði hleypisvið (squelsch) og styrk (audio). Hæfilegur styrkur fyrir mótaldið er um 1,0 Vpp en hleypisvið er háð styrk á RF (radio frequency) merkinu og þar með móttökuskilyrðum.

Á móttakaranum eru tvö tengi. Annað er RF tengi fyrir loftnetið en hitt er Bendix PT02E-8-4P, en það tengist mótaldinu. Tengingin í kaplinum milli móttakarans og mótaldsins er samkvæmt töflu 8.1 ef móttakarinn fær afl frá mótaldinu.

*TAFLA 8.1 Tenging kapals milli móttakarans og mótaldsins*

BNC	Kapall	Bendix	Merki
karl	blár	B	útmerki
kerling	brúnn	A	12 V
skermur	gulur/grænn	D	jörð

## 8.2 Mótaldið

Mótaldið byggir á Motorola rás MC145442, rás númer U201 á teikningu 2 í viðauka. Hún er 300 baud og samhæfð við CCITT V.21 staðal ef við hana er notaður 3,579545 MHz kristall.

Þrjú pinnar á rásinni eru notaðir til þess að stýra ýmsum möguleikum, sjá töflu 8.2. Það er pinni 13 (mode) sem segir til um hvort rásin sé í frumástandi (originate mode) eða svarástandi (answer mode), pinni 2 (LP) en með honum má setja rásina í analog loopback ástand til þess að prófa hana og pinni 14 (SQT) en hann er fasttengdur í jörð. (Sjá einnig fylgirit um rásina í viðauka).

Rofarnir SW201 og SW202 eru tengdir við pinna 2 og 13 og eiga þeir að vera stilltir á svarástand. Þá gilda tíðnirnar samkvæmt töflu 8.3.

Þrjú merki eru tengd frá U201 inn á RS-232 tengið, TE201. Það eru CD, RxD, og TxD, en þau fara öll í gegnum LT 1080, U202, sem er tvöfaldur RS-232 móttakari og sendir. Merkin þrjú sjást á ljósdíóðum framan á kassanum ásamt aflinu.

Afl fær mótaldið frá 220VAC/12VDC aflgjafa. Hann getur einnig séð móttakaranum fyrir 12 V afli en 5 V reglari, 7805 sér rásunum fyrir spennu. 500 mA öryggi er á 220 V.

*TAFLA 8.2 Tengingar á mótaldsrás*

MODE pinni 13 SW202	LB pinni 2 SW201	Ástand
0	0	Svarástand
1	0	Frumástand
x	1	Analog loopback

TAFLA 8.3. CCITT V.21 staðall

Bítar	Frumástand		Svarástand	
	Sending	Móttaka	Sending	Móttaka
0	1180 Hz	1850 Hz	1850 Hz	1180 Hz
1	980 Hz	1650 Hz	1650 Hz	980 Hz

### 8.3 A/D breytan

Til þess að geta tekið við gögnum frá hallamæli var smíðuð A/D breyta á staðlað bretti fyrir PC braut. Teikning nr. 3 í viðauka er af henni. Notuð er 12 bita A/D breyta, 7109, og er henni stýrt með 8255A rás.

Hliðræna merkið frá hallamælinum tengist á tengi TE301, jörð á pinna 1 og merkið á pinna 14. Merkið fer síðan í gegnum magnara með mögnun 1, en með honum má núllstilla það. Heildarsvið A/D breytunnar er +/- 4,096V. Hægt er að breyta heildarsviðinu með því að breyta viðmiðunargildinu milli pinna 39 og 36 því heildarsviðið tvöföld viðmiðunarspennan:

$$V_{in} = 2 * V_{ref}$$

Stillanlega viðnámið R303 er þannig tengt að með því er hægt að stilla viðmiðunargildið,  $V_{ref}$  frá 0 til 2,8 V.

8 lægstu bitarnir eru tengdir við port A á 8255, en 4 hæstu eru tengdir við port B ásamt POL og OR. Þrjú stýrimerki eru síðan tengd við port C, en það eru STATUS, RUN/HOLD og CE/LOAD.

A/D breytunni er stýrt í "direct mode". Stuttur púls á RUN/HOLD kemur mælingu af stað og við það verður status bitinn hár. Þegar status bitinn er aftur orðinn lágur er mælingin búinn og með því að setja CE/LOAD niður er hægt að lesa gildið.

Rásin 8255A nær yfir 4 vistir, sem eru:

300	hex	port A
301	hex	port B
302	hex	port C
303	hex	control port

Með control porti má stýra einstökum bitum á porti C. Undirforritið Adbreyta í forritinu SAFNA sér um að lesa af breytunni. Sjá forritið í viðauka.

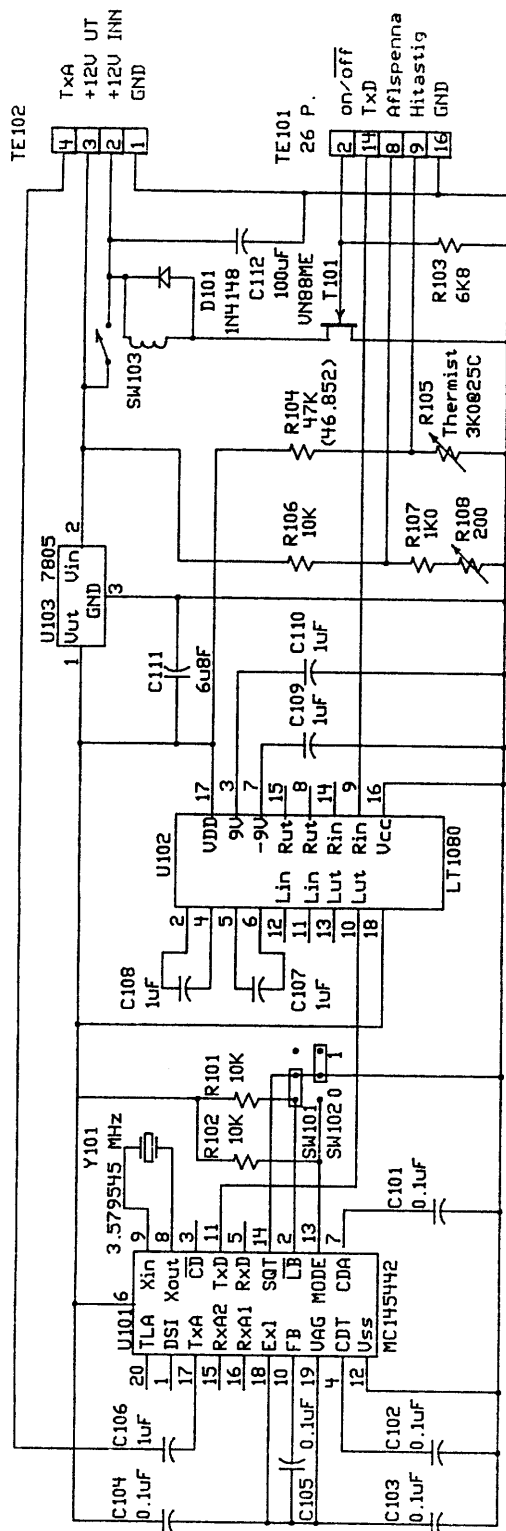




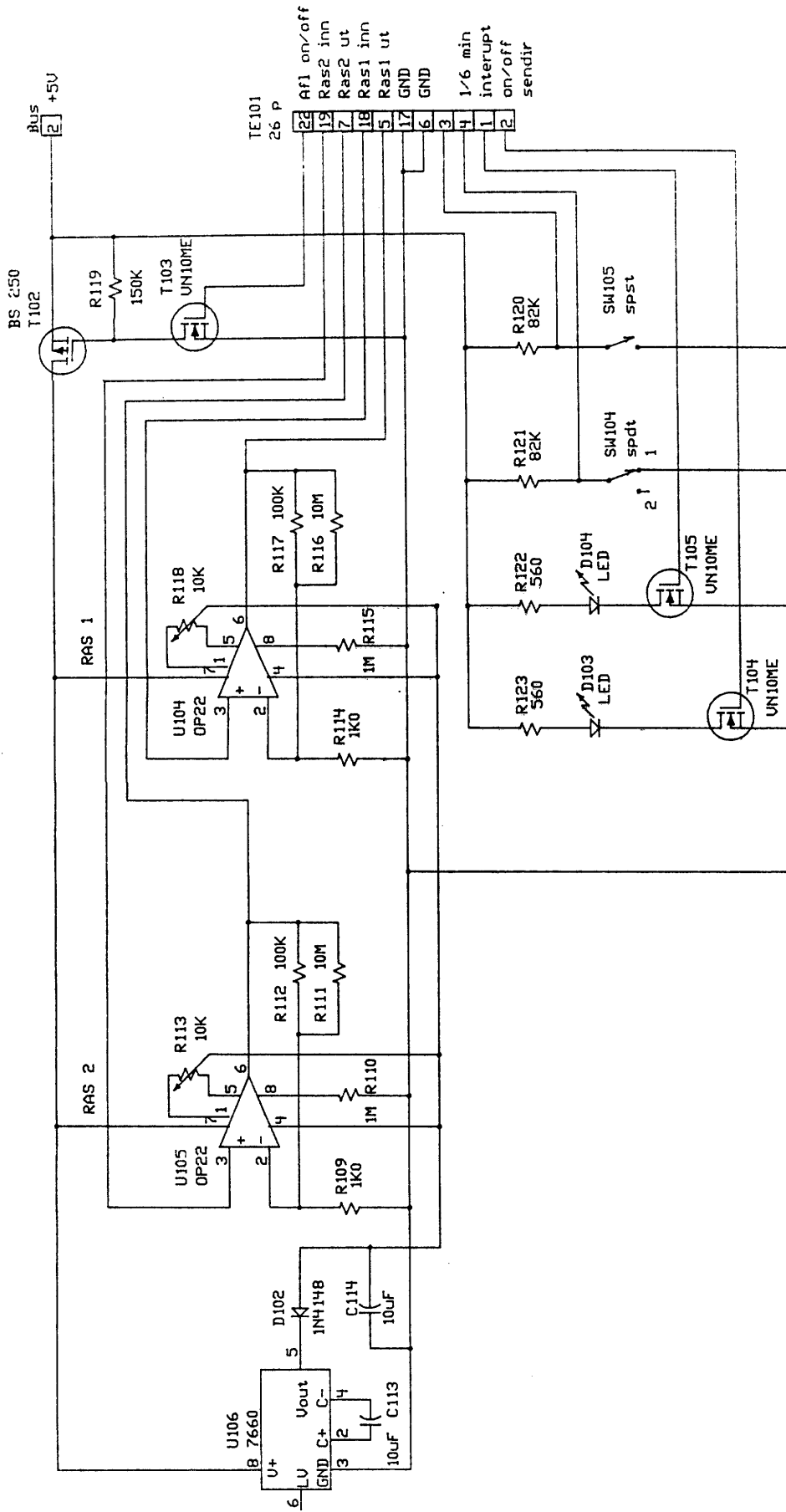
**VIÐAUKI A**

Teikningar

JHD.JED.6607.EHH  
89.01.0044.T

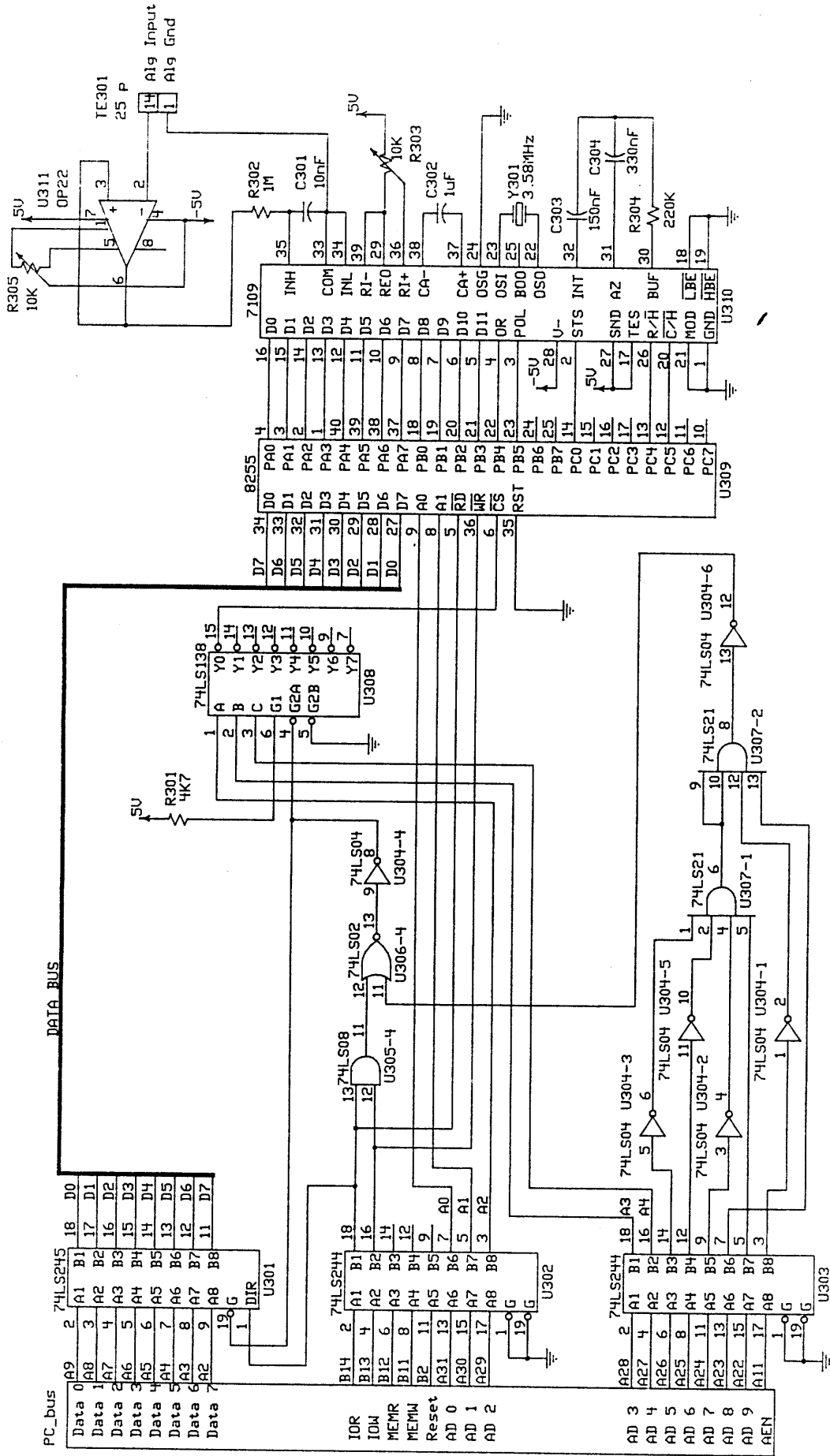


SVH 88 04 18	SPRUNGUMÆLIR
SVH 88 09 12	MODEM FYRIR
modem.dug	SENDI 300 baud
	Teikning nr.: 1
	Blad: 1
	Af: 2

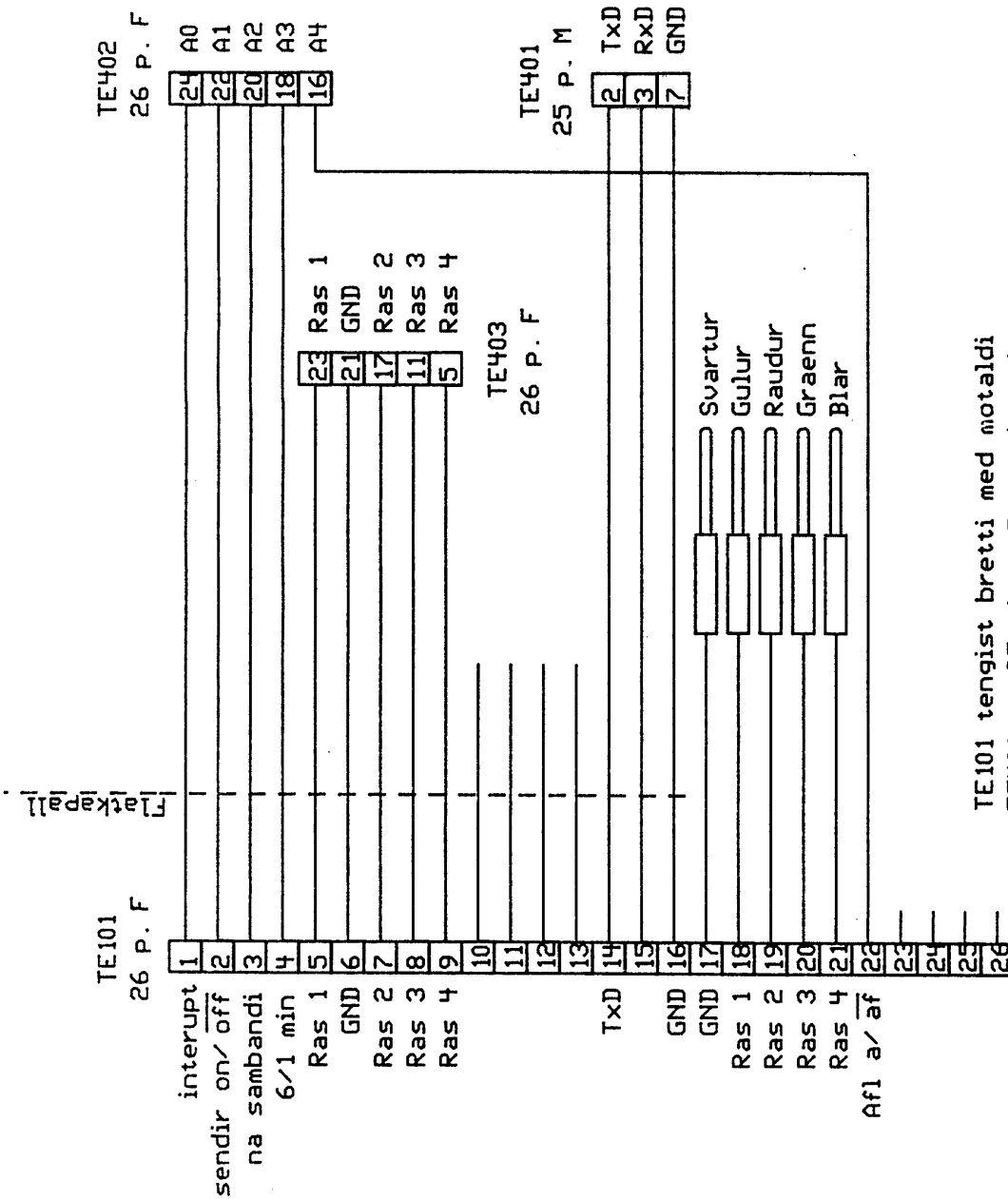


SVH 88 09 12	SPRUNGUMÆLIR	Blad: 2
	RIÐERDARMAGNARAR	Af: 2
buffer.dug	Teikning nr.: 1	

JHD:JED.6607.EHH  
89.01.0047 T



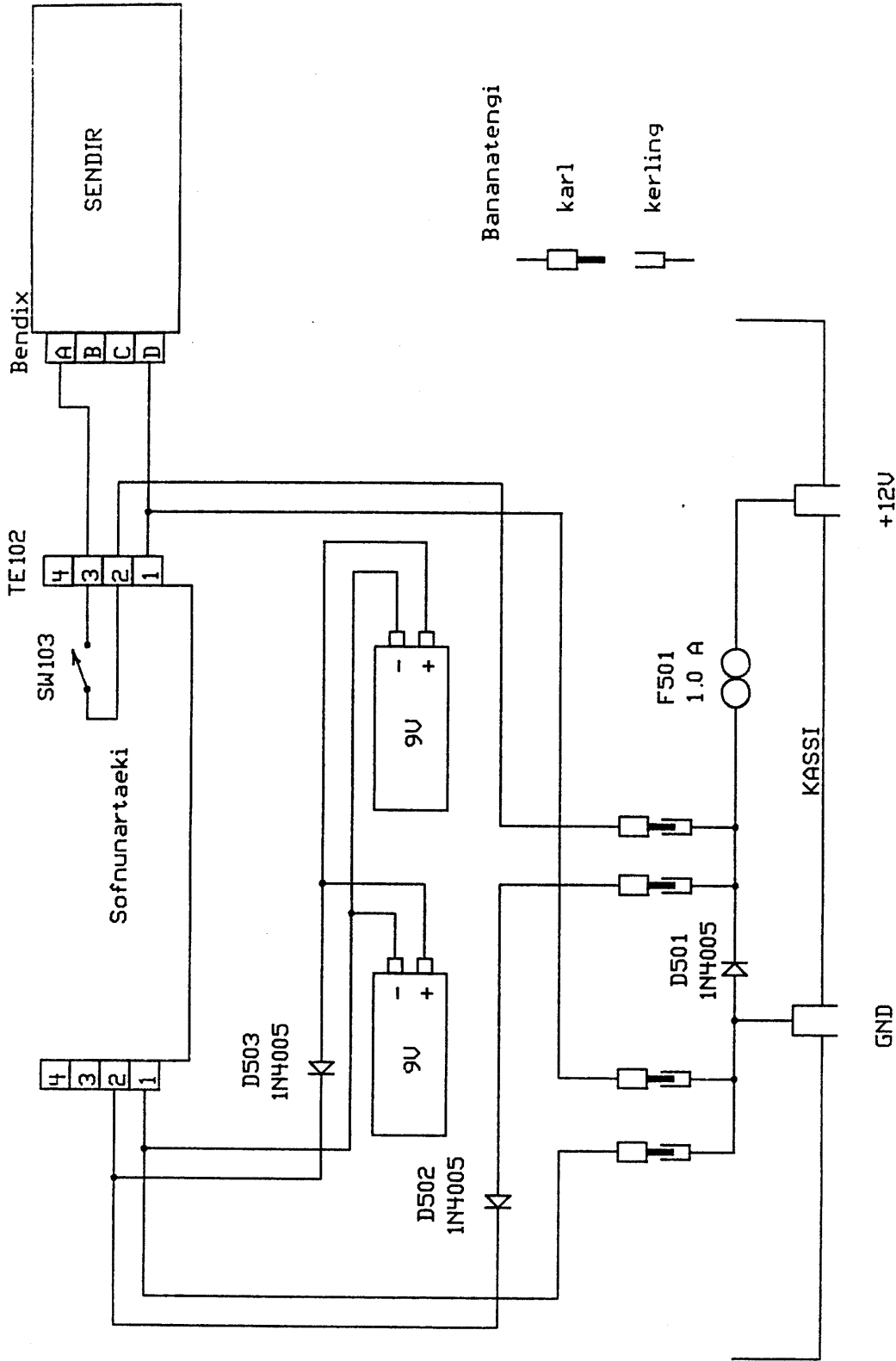
SvH 88 10 20	SPRUNGMAEL IR	Dlad:1
ad.dug	A/D BREYTA	Af:
	Teikning nr: 3	



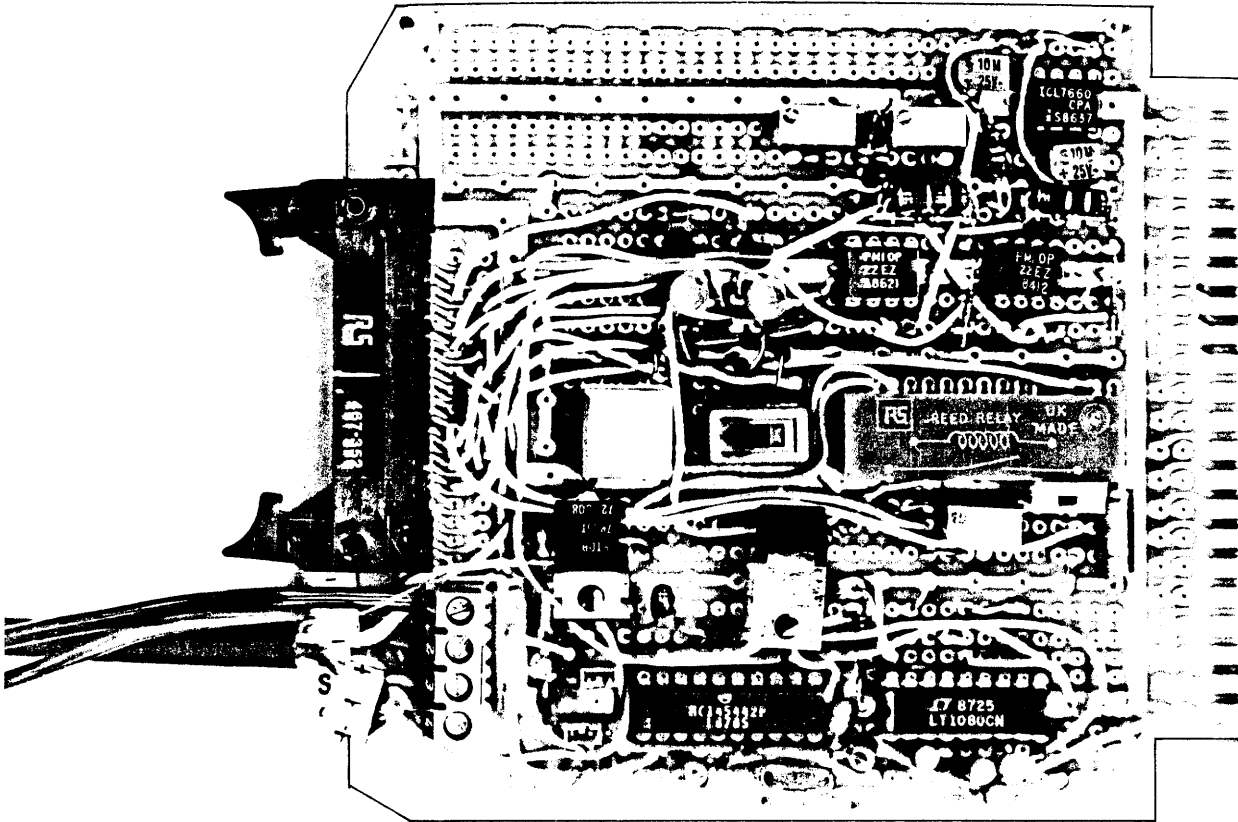
TE101 tengist bretti med motaldi  
 TE401 er 25 pinna D tengi samkvamt  
 RS232 stadli  
 TE402 tengist ortolvubretti  
 TE403 tengist A/D bretti

SVH 88 10 24	SPRUNGUMÆLIR	Blad: 1
kapall.dwg	Tengikapall	Af: 1
	Teikning nr: 4	

JHD..JED.6607.EHH  
189.01.0049 T



SvH 88 11 07	SPRUNGUMÆLIR	Blad: 1
	TENGING AFLS	Af: 1
af1.dwg	Teikning nr: 5	



*Stýřibrettí*



BOM - Wintek Bill-of-Materials V1.1r1

Teikning nr. 1

Sheet 1: buffer.dwg

Sheet 2: modem.dwg

site	part	value
=====		
	Bus	
	C101	0.1uF
	C102	0.1uF
	C103	0.1uF
	C104	0.1uF
	C105	0.1uF
	C106	1uF
	C107	1uF
	C108	1uF
	C109	1uF
	C110	1uF
	C111	6u8F
	C112	100uF
	C113	10uF
	C114	10uF
	D101	1N4148
	D102	1N4148
	D103	LED
	D104	LED
	R101	10K
	R102	10K
	R103	6K8
	R104	47K
	R105	3K0@25C (*)
	R106	10K
	R107	1K0
	R108	200
	R109	1K0
	R110	1M
	R111	10M
	R112	100K
	R113	10K
	R114	1K0
	R115	1M
	R116	10M
	R117	100K
	R118	10K
	R119	150K
	R120	82K
	R121	82K
	R122	560
	R123	560
	SW101	
	SW102	
	SW103	
	SW104	spdt
	SW105	spst
	T101	VN88ME

T102 BS 250  
T103 VN10ME  
T104 VN10ME  
T105 VN10ME  
TE101 26 P.  
TE102  
U101 MC145442 (\*)  
U102 LT1080 (\*)  
U103 7805  
U104 OP22  
U105 OP22  
U106 7660  
Y101 3.579545 MHz

BOM - Wintek Bill-of-Materials V1.1r1

Teikning 2

Sheet 1: modmot.dwg

site	part	value
=====		
	C201	0.1uF
	C202	0.1uF
	C203	0.1uF
	C204	0.1uF
	C205	0.1uF
	C206	1uF
	C207	1uF
	C208	1uF
	C209	1uF
	C210	1uF
	C211	6u8F
	C212	100nF
	D201	LED
	D202	LED
	D203	LED
	D204	LED
	F201	500mA
	R201	10K
	R202	10K
	R203	560
	R204	1K
	R205	1K
	R206	1K
	R207	1K
	SW201	
	SW202	
	TE201	9 p. D
	TE202	Twin BNC
	U201	MC145442 (*)
	U202	LT1080 (*)
	U203	7400
	U204	7805
	U205	220/12V
	Y201	3.579545MHz

BOM - Wintek Bill-of-Materials V1.1r1  
Teikning 3  
Sheet 1: dwgd.dwg

site	part	value
		=====
	C301	10nF
	C302	1uF
	C303	150nF
	C304	330nF
	PCbus	
	R301	4K7
	R302	1M
	R303	10K
	R304	220K
	R305	10K
	TE301	25 p
	U301	74LS245
	U302	74LS244
	U303	74LS244
	U304-1	74LS04
	U304-2	74LS04
	U304-3	74LS04
	U304-4	74LS04
	U304-5	74LS04
	U304-6	74LS04
	U305-4	74LS08
	U306-4	74LS02
	U307-1	74LS21
	U307-2	74LS21
	U308	74LS138
	U309	8255
	U310	7109
	U311	OP22
	Y301	3.58MHz

## VIÐAUKI B

Listun á forritum í PC tölvu

```

(*
  GAGNASÖFUNARTORRIT      FYRIR SJÁLFRIRKAN SPRUNGUMELI Í KRÖFLU
  V/1.2 88
  Skrifað í Turbo Pascal v/3.0

  Orkustofnun      12. Janúar 1989
  Sverrir Hákonarson *)

program SAFNA ;

function testl : boolean ;      external 'intr8250.bin' ;
function getchar : char ;      external testl [ 3 ] ;
procedure init(buffseg:integer) ; external testl [ 6 ] ;

type
  strengur = record
    dagsetn,timi,dagnr : integer ;
    nrinnandags,ras1,ras2,spenna : byte ;
    hiti : integer ;
    halli : byte ;
  end ;
  string80 = string[80] ;
  dagstrengur = string[8];
  regRecord = record
    ax,bx,cx,dx,bp,di,si,ds,es,flags:integer ;
  end;

const
  dport = $3F8 ;
  port2 = $2F8 ;

(* hitaledretting er strengur sem inniheldur hitastigsgildi sem
samsvarar ákvæðnu viðnámsgildi á hitaháðu viðnámi í söfnunatakínu
*)
hitaledretting : array[0..255] of integer =
(1000, 1000, 1000, 898, 807, 742, 685, 640, 601, 568,
539, 513, 490, 468, 449, 431, 414, 398, 384, 370,
357, 344, 333, 322, 311, 301, 291, 282, 273, 265,
256, 248, 241, 233, 226, 219, 212, 206, 199, 193,
187, 181, 176, 170, 165, 159, 154, 149, 144, 139,
134, 130, 125, 121, 116, 112, 108, 104, 99, 95,
91, 88, 84, 80, 76, 73, 69, 66, 62, 59,
55, 52, 49, 45, 42, 39, 36, 33, 30, 27,
24, 21, 18, 15, 12, 10, 7, 4, 1, -1,
-4, -7, -9, -12, -14, -17, -19, -22, -24, -27,
-29, -31, -34, -36, -38, -41, -43, -45, -47, -50,

```

```

-52, -54, -56, -58, -61, -63, -65, -67, -69, -71,
-73, -75, -77, -79, -81, -83, -85, -87, -89, -91,
-93, -95, -96, -98, -100, -102, -104, -106, -108, -109,
-111, -113, -115, -117, -118, -120, -122, -124, -125,
-129, -130, -132, -134, -135, -137, -139, -140, -142, -144,
-145, -147, -148, -150, -152, -153, -155, -156, -158, -160,
-176, -178, -179, -181, -182, -184, -185, -187, -188, -190,
-191, -193, -194, -195, -197, -198, -200, -201, -204,
-205, -207, -208, -210, -211, -212, -214, -215, -217, -218,
-219, -221, -222, -223, -225, -226, -228, -229, -230,
-233, -234, -236, -237, -238, -240, -241, -242, -244,
-246, -248, -249, -250, -251, -253, -254, -255, -257,
-259, -261, -262, -263, -264, -266, -267, -268, -270,
-272, -273, -275, -276, -277, -279) ;

```

```

(* stöðuv er 4*5 fylki til þess að aðgreina mælistreng frá suði
Línurnar gefa stöðuna 1 til 4 en dálkarnir 1 til 5 eru fyrir
mismunandi charatera.

```

```

char ASCII dálkur
LF 10 1
CR 13 2
space, 32,45 3
0..9 48..57 4
else 5

```

```

Upphafsstaðan er 1 (lína 1) og ef space eða - kemur,
í dálk 3 sem þýðir nýja stöðu, 2 (sjá fylki hér að neðan).
Ef einkver annar stafur kemur inn þýðir það dálkur 5 og þar
með upphafsstaðan aftur.
Ef loka staðan næst sem er 5, telst mælistrengurinn löglegur.
sjá procedure Lesagildi*)

```

```

stoduv : array[1..4,1..5] of integer =
((1,1,2,1,1),
(1,1,1,3,1),
(1,4,2,3,1),
(5,1,1,1,1)) ;

```

```

var
  streng : strengur ;
  prentari,modem : text ;
  skra : file of strengur ;
  skraarnafn : string[50] ;
  stafur : char ;
  stopp,drifok,lesturok,samskiptiok : boolean ;

```

```

alarmon,advonunon : boolean ;
fyrriStada : byte ;
fyrRabil,seinnabil,fyrriupp,seinniupp : boolean ;
sp,hiti : real ;
gildi,gagnadrif : string[29] ;
telja,prentun,bendill : integer ;
p : ^char ;
fj1,fj2 : integer ;
undan1,undan2,x,summa1,summa2 : integer ;
s1 : array [1..10] of integer ;
s2 : array [1..10] of integer ;

(* Initline sér um að taka frá minni fyrir bufferinn sem
heldur inniheldur gildin sem koma inn um sereal port 1. Einnig
er portíð stillt á rétt format *)

procedure Initline ;
var
  serealformat : integer ;
  i,serealbaudr,sereal8bits : integer ;
  serealparityeven,serealparityon : integer ;
  sereal2stoppbits,loopback : integer ;
  bufferseg,buffermark : integer ;
begin
  sereal8bits := 1 ; { fjöldi bita 8/7 }
  serealparityeven := 0 ; { parity even/odd }
  serealparityon := 0 ; { parity on/off }
  sereal2stoppbits := 0 ; { stoppbitar 2/1 }
  loopback := 0 ; { loopback on/off }
  serealbaudr := 384 ; { serealbaudr = 115200/baudr }

  getmem(p, 32*1024) ;
  bufferseg := seg(p) ;
  getmem(p, 32*1024) ;
  serealformat := 2 + ord(sereal8bits) + 16*ord(serealparityeven) +
  8*ord(serealparityon) + 4*ord(sereal2stoppbits) ;
  port[dport+4] := 15 + 16*ord(loopback) ;
  port[dport+3] := serealformat ;
  init(bufferseg) ;
  i := port[dport+3] ;
  port[dport+3] := 128 + i ;
  port[dport] := lo (serealbaudr) ;
  port[dport+1] := hi(serealbaudr) ;
  port[dport+3] := i ;
  assign(preantari,'prn') ;
  rewrite(preantari) ;
  gagnadrif := 'c:\' ;

alarmon := 3 ; {Fjöldi mæligilda á L6}
fj2 := 2 ; {Fjöldi mæligilda á L5}

end ; {initline}

(* Com2 stillir serealport 2 á rétt format *)

procedure StillaCom2 ;
var
  i,serealformat : integer ;
  serealbaudr,sereal8bits : integer ;
  serealparityeven,serealparityon : integer ;
  sereal2stoppbits : integer ;
begin
  sereal8bits := 1 ; { fjöldi bita 8/7 }
  serealparityeven := 0 ; { parity even/odd }
  serealparityon := 0 ; { parity on/off }
  sereal2stoppbits := 0 ; { stoppbitar 2/1 }
  serealbaudr := 384 ; { serealbaudr = 115200/baudr }

  serealformat := 2 + ord(sereal8bits) + 16*ord(serealparityeven) +
  8*ord(serealparityon) + 4*ord(sereal2stoppbits) ;
  port[port2 + 3] := serealformat ;
  i := port[port2 + 3] ;
  port[port2 + 3] := 128 + i ;
  port[port2] := lo (serealbaudr) ;
  port[port2 + 1] := hi(serealbaudr) ;
  port[port2 + 3] := i ;
end ; {StillaCom2}

(* IntrDisable gerir interrupt á RS232 portíð óvirk *)

procedure IntrDisable ;
begin
  port[dport+1] := 0 ;
end ; {IntrDisable}

(* IntraEnable gerir interrupt á RS232 portíð virk *)

procedure IntraEnable ;
var
  z : char ;

```

```

m : integer ;
begin
  while testl do z := getchar ;
  m := port[port] ;
  port[port + 1] := 1 ;
end; {IntrEnable}

{ * Hex breytir decimal tölu í hexadecimal tölu * }

function Hex(dec : integer) : integer ;
begin
  hex := (dec div 16) * 10 + dec mod 16 ;
end ;

{ * Getdate skilar 6 char. streng sem inniheldur
dagsetningu: ddimyy * }

function Getdate : dagstrengur ;
var
  status, year, month, day: integer ;
  ar, man, dag: string[2] ;
  error : integer ;
  dagss : string[8] ;
  regs: regrecord ;
begin
  repeat
    year := port[$249] ;
    status := port[$254] ;
  until status = 0 ;
  repeat
    month := port[$247] ;
    status := port[$254] ;
  until status = 0 ;
  repeat
    day := port[$246] ;
    status := port[$254] ;
  until status = 0 ;

  str(hex(year), ar) ;
  str(hex(month), man) ;
  str(hex(day), dag) ;

  m := port[port] ;
  port[port + 1] := 1 ;
end; {IntrEnable}

if length(dag) = 1 then insert('0', dag, 1);
if length(man) = 1 then insert('0', man, 1);
dagss := dag + man + ar;
Getdate := dagss ;

{ * streng.dagsetn inniheldur dagssetningu á forminu:
integer: msb xxxxxx xxxx xxxxx lsb * }

streng.dagsetn := (hex(year) - 88) * 512 + hex(month) * 32 + hex(day) ;

with regs do
begin
  ax := $2b shl 8 ;
  cx := (1900 + hex(year));
  dx := hex(day) + hex(month) shl 8 ;
end;
intr($21, regs);
end; {Getdate}

{ * Gettime er 4 char strengur sem skilar klukkuni hh:mm * }

function Gettime : dagstrengur;
var
  timi : string[8] ;
  stund, min, sek: string[2] ;
  regs: regrecord ;
  errorcode : integer ;
  hours, minutes, status : integer ;
begin
  repeat
    hours := port[$244] ;
    status := port[$254] ;
  until status = 0 ;
  repeat
    minutes := port[$243] ;
    status := port[$254] ;
  until status = 0 ;

  str(hex(hours), stund) ;
  str(hex(minutes), min) ;

  if length(stund) = 1 then insert('0', stund, 1) ;
  if length(min) = 1 then insert('0', min, 1) ;
  timi := stund + ':' + min ;

```



```

(Ef að breytan lesturok er true, þá eru gildin úr strengnum gildi
lesin í fox[i] }
if lesturok then
begin
  j := 0 ;
  b:= 2 ;
  errigildi := 0 ;
  for c := 2 to length(gildi) do
  if ((char(32)=copy(gildi,c,1))or(char(13)=copy(gildi,c,1)))
  and (errigildi=0) then
  begin
    j := j + 1 ;
    val(copy (gildi,b,c-b),fox[j],errigildi) ;
    b:= c + 1;
  end;
  samskiptok := (errigildi = 0) and (j = 6) ;
  lesturok := lesturok and (errigildi = 0)and (j = 6) ;
end ; (if lesturok)

if lesturok then
begin
  with streng do
  begin
    dagr := fox[1] ;
    nrinnandags := fox[2] ;
    ras1 := fox[3] ;
    ras2 := fox[4] ;
    sperma := trunc(fox[5] * 0.977) ;
    hiti := hitaleiðretting[fox[6]] ;
  end ; (with)
  Rest := Getdate ;
  Rest := Gettime ;
  end ;(if)
end ; (Lesagildi)

{ * ADgildi les gildi af a/d breytu og skilar gildinu sem
sem integer í breytuni streng.halli. *}

procedure Adbreyta ;
var
  i,portA,portB,portC,formerki : integer ;
begin
  port[$302] := $00 ;
  port[$303] := $93 ; {Les 10010011 í control port á 8255}
  port[$303] := $08 ; {00001011 setur 5 bit = 1, CE/LOAD }

```

```

Gettime := timei ;
val(stund + min,streng.timi,errorcode) ;
end ; (Gettime)

{ * Lesagildi les staf úr buffer þegar breytan testl verður true.
Lesagildi sér um að málstrengurinn uppfylli öll skilyrði um lengd og
að ekki sé ólöglegir characterar í honum *}

procedure Lesagildi;
var
  exit : boolean ;
  stada,inn,b,c,j,time,errigildi : integer;
  character : char ;
  fox : array [1..6] of integer ;
  rest : dagstrengur ;
begin
  exit := false;
  gildi := '';
  stada := 1 ;

  while not exit and testl and not (stada = 5) do
  begin
    character := getcharr ;
    case ord(character) of
      10 : inn := 1 ;
      13 : inn := 2 ;
      32,45 : inn := 3 ;
      48..57 : inn := 4 ;
      else inn := 5 ;
    end ;
    stada := stoduv[stada,inn] ;
    if stada in [2..5] then gildi := gildi + character
    else gildi := '' ;
  end ;
  (Bíður eftir næsta staf eða að
  breytan exit verður true eftir u.p.b. 600 ms)

  time := 0 ;
  repeat
    time := time + 1 ;
  until testl or exit or (stada = 5) ;
end ; (while)

lesturok := not exit and (stada =5);

```

```

port[$303] := $09 ; {00001001 setur 4 bita = 1, RUN/HOLD }
delay(5) ;
port[$303] := $08 ; {00001000 setur 4 bita = 0, RUN/HOLD }
repeat
  i := port[$302] ; {Bíður eftir að status verði 0 }
until i and 1 = 0 ;
port[$303] := $0A ; {00001010 setur 5 bit = 0, CE/LOAD }
portA := port[$300] ;
portB := port[$301] ;
if (32 and portB) = 32 then formerki := 1
else formerki := -1 ;
streng.halli := ((15 and portB) * 256 +) 128 + portA * formerki ;
port[$303] := $08 ; {00001011 setur 5 bit = 1, CE/LOAD }
end; {Abreyta}

{
  Bladsíðskipti sér um að
  telja mælingarnar og sendir formfeed *}

procedure Bladsíðskipti ;
begin
  prentun := prentun + 1 ;
  if prentun = 2400 then
  begin
    write(prentari,chr(12)) ;
    prentun := 0 ;
  end ;
end ; {bladsíðskipti}

{
  Prentadags verður true í 30. skipti sem prentað er *}

function Prentadags: boolean;
begin
  telja := telja + 1;
  prentadags := (telja = 30) ;
  if telja = 30 then telja := 0 ;
end; {Prentadags}

{
  Prentariok athugar prentarastatus, og Prentariok er true
  ef prentari er ok *}

function Prentariok : boolean;
var
  regs : regrecond;
  pok : boolean;
  begin
    regs.ax := $02 shl 8 ;
    regs.dx := $00 ;
    intr($17,regs);
    prentariok := (regs.ax = -28672) ;
end;{Prentariok}

{
  Prentagildi sendir eina mælingu á prentarann ásamt dagsetningu ef
  boolean breytan Prentadags er true *}

procedure Prentagildi(ferill1,ferill2:byte) ;
var
  dagsetn : boolean ;
  x : integer;
begin
  x := 1;
  { Bladsíðskipti;}
  dagsetn := Prentadags;
  if dagsetn then
    write(prentari,chr(27),**i,chr(6),chr(8),chr(2))
  else
    write(prentari,chr(27),**i,chr(6),chr(3),chr(2));
  repeat
    if (x=1) or (x=ferill1+2) or (x=ferill2+259) or (x=515) or (x=258)
    then write(prentari,chr(8))
    else write(prentari,chr(0));
    x := x + 1;
  until x > 515;
  if dagsetn then
  begin
    for x := 1 to 5 do write(prentari,chr(8));
    write(prentari,Gettime:6,Getdate:9
    streng.dagnr:5,streng.nriinnandags:4,ferill1:4,ferill2:4);
  end;
  write(prentari,chr(10));
end; {Prentagildi}

```

```

end ;

fjoldiskra := fjoldi ;
end;

(* Myskra byr til nýja skra sem heitir dagsetningu, ef önnur finnst ekki *)
procedure Myskra;
var
  fjold : integer ;
begin
  Bendill := 0 ;
  fjold := fjoldiskra(Getdate) ;
  skraarnafn := gagnadrif + Getdate + chr(65 + fjold) + '.KRA';
  writeIn(skraarnafn);
  assign(skra, skraarnafn);
  rewrite(skra);
  close(skra);
end; {Myskra}

(* OK athugar breytuma IOResult, ef hún er annað en 0 sem þýðir að villi hafi komið fram, ákveður undirforritið þá hvað skal gera eftir gildi breytunnar. *)
procedure OK(Wrerror:integer) ;
begin
  { writeIn('IOResult: ',Wrerror);}
  case Wrerror of
    0 : begin end;
    3,242,4 : begin
      Myskra;
      end;
    240 : Drifok := false ;
  else Drifok := false;
  end;
end;{OK}

(* Enoughspace athugar hvort að nóg plass sé á disknum enoughspace er true ef svo er *)

```

```

(* StillaPrentara sendir stýristafi á prentarann til þess að stilla hann *)
procedure StillaPrentara;
begin
  write(preantari,chr(27),'3',chr(1)); {Stilla prentara á 1/216" LF.}
  write(preantari,chr(27),'x1'); {Stilla prentara á gæðaletur.}
  write(preantari,chr(15)); {Stilla prentara á Compressed.}
  write(preantari,chr(27),'C',
  chr(0),chr(12)); {Stilla prentara á 12" blaðs.l}
end; {StillaPrentara}

(* Fjoldiskra athugar hversu margar skrár eru á gagnadrifi með dagsetningunni Getdate og skilar í fjoldiskra *)
function fjoldiskra( FileSpec: String80):integer;
var
  fjoldi : integer ;
  Reg : Regrecord;
  DTA : array[1..43] of byte;
begin
  FileSpec := gagnadrif + filespec + '? .kra' + chr(0) ;
  fjoldi := 0 ;
  Reg.DX := ofs(DTA);
  Reg.DS := seg(DTA);
  Reg.AX := $1A00;
  msdos(Reg);

  Reg.DX := ofs(FileSpec[1]);
  Reg.DS := seg(FileSpec[1]);
  Reg.CX := $10;
  Reg.AX := $4E00;
  msdos(Reg);

  while lo(reg.ax) = 0 do
  begin
    fjoldi := fjoldi + 1 ;
    Reg.DX := ofs(DTA) ;
    Reg.DS := seg(DTA) ;
    Reg.AX := $4F00 ;
    msdos(Reg) ;
  end;

```

```

function Enoughspace : boolean;
type
  regRecord = record
    ax,bx,cx,dx,bp,di,si,ds,es,flags:integer;
  end;
var
  regs:regRecord;
  freespace : integer;
  axr,bxr,cxr : real ;
begin
  {$I-}
  regs.ax := $36 shl 8;
  regs.dx := ord(upcase(copy(gagnadrif,1,1))) - 64 ;
  intr($21,regs);
  ok(ioresult) ;
  {$I+}
  axr := regs.ax ;
  bxr := regs.bx ;
  cxr := regs.cx ;
  freespace := round(bxr * axr * cxr / 1000) ;
  Enoughspace := (freespace > 260);

  {* freespace er fjöldi kbyte sem er laus á gagnadisknum}
  { writeIn(free space: ',freespace:5,regs.ax:5,regs.bx:5,
    regs.cx:5,regs.dx:5); }
end; {FreeSpace}

{* Skrifagildi skrifar mæligildin út á disk, á com 2 og skjá og
einnig dagsetningn *}
procedure Skrifagildi;
var
  karakter : array[1..9] of string[8] ;
  i,x,y,villanr : integer ;
  strengur : string[80] ;
  einstakur : char ;
  hallagildi : string[8] ;
  a,comstatus,error : integer ;
  comport : text ;
begin
  reset(skra) ;
  seek(skra,bendill) ;
  Bendill := bendill + 1 ;
  write(skra,streng);
  villanr := ioreult ;
  ok(villanr);
  close(skra);
  until villanr = 0 ;
  {$I+}

  with streng do
  begin
    str(dagsetn,karakter[1]) ;
    str(timi,karakter[2]) ;
    str(dagnr,karakter[3]) ;
    str(nrinnandags,karakter[4]) ;
    str(ras1,karakter[5]) ;
    str(ras2,karakter[6]) ;
    str(spenna,karakter[7]) ;
    str(hiti,karakter[8]) ;
    str(halli,karakter[9]) ;
  end ;

  strengur := '' ;
  for i := 1 to 9 do
  begin
    strengur := strengur + ' ' + karakter[i] ;
  end ;

  strengur := strengur + char(13) + char(10) ;
  for a := 1 to length(strengur) do
  begin
    einstakur :=copy(strengur,a,1) ;
    repeat
      comstatus := port[$2FD] ;
    until comstatus and $20 = $20 ;
  end ;
  write(strengur) ;

  x := wherey ;
  y := wherey ;
  sp := streng.spenna / 10 ;
  hiti := streng.hiti / 10 ;

```

```

gotoxy(42,1) ;
write('Nafn gagnaskráar er ',skraarnafn) ;
gotoxy(42,2) ;
write('Síðasta mæling kom ',Gettime,' ',Getdate,' ');
gotoxy(42,3) ;
write('Staða          hallamælis: ',streng.halli:5,' ');
gotoxy(42,4) ;
write('Afliþspenna: ',sp:3:1,' V Hitastig: ',hiti:4:1,' °C');
gotoxy(x,y) ;
end; {Skrifagildi}

{ * Aflok ákveður við hvaða spennu aðvörun skal gerð. * }
function Aflok : boolean ;
begin
  Aflok := (115 < streng.spenna) ;
end; {Aflok}

{ * Keyborad ákveður hvað gera skuli þegar slegið er á lýklaborðið
í þýðir að hætta á keyrslu á forritinu
A a ákvör hvort hljóðmerki skuli vera á eða ekki. * }
procedure Keyboard ;
begin
  read(kbd,staful) ;
  case staful of
    'i' : stopp := true ;
    'A','a' : alarmon := not alarmon ;
  end;
end; {Keyboard}

{ * Softalarmlar spilar noturnar G og D í 70 ms þar til í keypressed eða
testl verða ture * }
procedure Softalarmlar ;
const
  Gnota = 829 ;
  Dnota = 621 ;
var
  i,k : integer ;
begin
  k := 0 ;
  while alarmon and (k <= 10) do
  begin
    k := k + 1 ;
    if keypressed then Keyboard ;
    for i := 1 to 7 do
      begin
        sound(gnota) ;
        delay(70) ;
        nosound ;
        sound(dnota) ;
        delay(70) ;
        nosound ;
      end ;
      delay(1000) ;
    end ;
  end ; {Softalarmlar}

{ * Aðvörun athugar ákveðinn fjölda mæligilda rása1 og rása 2 og
ef átt hafa sér stað breytingar á þeim í aðra hvora áttina
gefur hún aðvörun með procedure Softalarmlar og skrifar
skilaboð á skerminn * }
procedure Advorum ;
{ * Athugun á Rás 1 * }
begin
  for x := fj1 downto 1 do s1[x+1] := s1[x] ;
  case streng.ras1 - undan1 of
    -32000..-1 : s1[1] := -1 ;
    0 : s1[1] := 0 ;
    1..32000 : s1[1] := 1 ;
  end ;
  undan1 := streng.ras1 ;
  summa1 := 0 ;
  for x := 1 to fj1 do summa1 := summa1 + s1[x] ;
end ;

{ * Athugun á Rás 2 * }
for x := fj2 downto 1 do s2[x+1] := s2[x] ;
case streng.ras2 - undan2 of
  -32000..-1 : s2[1] := -1 ;
  0 : s2[1] := 0 ;
  1..32000 : s2[1] := 1 ;
end ;
undan2 := streng.ras2 ;

```

```

summa2 := 0 ;
for x := 1 to fj2 do summa2 := summa2 + s2[x] ;
advorunon := (fj1 = abs(summa1)) and (fj2 = abs(summa2))
and (summa1 * summa2 >= 0) ;
end ; {advorun}

{ * Ath skrifar skilaboð ef eitthvað er að.* }
procedure Ath;
var
  xx,yy,i : integer ;
begin
  xx := wherex ;
  yy := wherey ;
  for i := 1 to 7 do
  begin
    gotoxy(42,4 + i) ;
    write('
  ');
  end ;
  i := 7 ;
  gotoxy(42,6) ;
  if alarmon then write('Hljóðádvörun er á (A-takki)')
  else write('Hljóðádvörun er ekki á (A-takki)') ;
  if advorunon then
  begin
    gotoxy(42,i) ;
    write('Miklar breytingar eiga sér stað !')
  end ;
  if not Aflok then
  begin
    gotoxy(42,i) ;
    write('Afl á söfnunartæki er að klárast !') ;
    i := i + 1 ;
  end ;
  if not prentariok then
  begin
    gotoxy(42,i) ;
    write('Prentarinn er ekki virkur !') ;
    i := i + 1 ;
  end ;
  if not enoughspace then
  begin
    gotoxy(42,i) ;
    write('Skriptið um diskling í drifi ',gagnadrif) ;
    i := i + 1 ;
  end ;
  if not samskiptiok then
  begin
    gotoxy(42,i) ;
    write('Villa í samskiptum !') ;
    i := i + 1 ;
  end ;
  gotoxy(xx,yy) ;
end; {Ath}

{ * Prentaritiib prentar streng á meðan
prentarinn er ekki tilbúinn * }
procedure Prentaritiib ;
begin
  while not prentariok do
  begin
    gotoxy(20,10) ;
    write('PRENTARINN ER EKKI TILBÚINN
  !!!') ;
    end ; {Prentaritiib}
  end ;
  (* ADALFORRIT *)
BEGIN
  Initline ;
  clrscr ;
  Intradisable ;
  Prentaritiib ;
  Stillaprentara ;
  StillaCom2 ;
  clrscr ;
  Nyskra ;
  stopp := false ;

```

```

lesturok := true ;
samskiptiok := true ;
alarmon := true ;
streng.spenna := 125 ;
telja := 0 ;
prentun := 0 ;
bendill := 0 ;

Intrenable ;

while not stopp do
begin
  Ath ;
  while alarmon and advorunon and not testl do Softalarm ;
  repeat until testl or keypressed ;

  if testl then
  begin
    LesaGildi ;
    if lesturok then
      begin
        Adbreyta ;
        if prentariok then
          begin
            Stillaprentara ;
            Prentagildi(streng.ras1,streng.halli) ;
          end ;(if prentariok)
          StillaCom2 ;
          Skrifagildi ;
          Advorun ;
          end ; (if lesturok)
        end ; (if testl)
      end ;
      if keypressed then Keyboard ;
    end; (while Stopp)
  end ;
  Intrinsic ;
  END.

```

```

(*   FORRIT SEM ÞYÐIR .KRA SKRÁR Á ASCII   FORMAT.
    v/1.1   Skrifað í Turbo   Pascal V/5.0
    Orkustofnun   10. Janúar 1989
    Sverrir Hákonarson
*)

program lesa ;
uses crt,dos ;

type
  strengur = record
    dagsetn,timi,dagmr : integer ;
    nrinnardags,ras1,ras2,spenna : byte ;
    hiti : integer ;
    halli : byte ;
  end;

var
  stjornur : array[0..9] of boolean ;
  skra : array[1..100] of string ;
  ch : char ;
  xorigin,yorigin,i,q,mrerror : integer ;
  innskra : file of strengur ;
  uts : text ;
  utskra,gagn : string ;
  maling : strengur ;
  drif : string ;

procedure directory ;
var
  s:string[80] ;
  Srec : SearchRec ;
  i,j : integer ;
begin
  i := 0 ;
  yorigin := 5 ;
  xorigin := 15 ;
  clrscr ;
  gotoxy(25,4) ;
  textcolor(7) ;
  textbackground(2) ;
  write('NAFN GAGNADRIFSI') ;
  textcolor(7) ;
  textbackground(0) ;
  write(' : ') ;
  readln(drif) ;

  clrscr ;
  FindFirst(drif+'*.kra',Anyfile,Srec) ;
  while doserror = 0 do
  begin
    i := i + 1 ;
    skra[i] := srec.name ;
    findnext(srec) ;
  end ;

  if i = 0 then
  begin
    gotoxy(20,15) ;
    textbackground(2) ;
    writeln('Engin gagnaskrá er á drifi ',drif,' !!!') ;
    textbackground(0) ;
    nrerror := 1 ;
    exit ;
  end ;
  textbackground(2) ;
  gotoxy(15,2) ;
  writeln('GAGNASKRÁR Á DRIFI ',drif) ;
  textbackground(0) ;

  for j := 1 to i do
  begin
    gotoxy(xorigin,yorigin + j - 1) ;
    write(skra[j]) ;
  end ;
  textbackground(2) ;
  textcolor(6) ;
  gotoxy(10,24) ;
  writeln('Notið ',#17,#196,#217,' ,#24,' #25,' til
  pess að velja skrána.' ) ;
  textbackground(2) ;
  textcolor(7) ;
  textbackground(0) ;

  gotoxy(13,yorigin) ;
  repeat
  ch := readkey ;
  if ch = #0 then
  case readkey of
    #80 :
      begin
        if wherey < i + yorigin - 1 then

```



```

gotoxy(wherex ,wherey + 1) ;
end;
#72 :
begin
  if wherey > yorigin then
    gotoxy(wherex , wherey -1) ;
  end ;
end ;
if ch = #13 then
  begin
    gagn := skra[wherey - yorigin + 1] ;
    assign(innskra,drif + gagn) ;
    reset(innskra) ;
    nrorror := ioresult ;
    if nrorror <> 0 then
      begin
        gotoxy(20,15) ;
        writeln('Skrána er ekki hægt að opna !!!!') ;
        exit ;
      end ;
    end ;
  end ;
  {
  if (ch = #13) and not (wherey = i) then
    begin
      if stjornur[wherey - 10] then
        write(' ')
      else write('*') ;
      gotoxy(wherex-1,wherey) ;
      stjornur[wherey - 10] := not stjornur[wherey - 10] ;
    end ;
  }
until ch = #13 ;
end;

procedure velja ;
begin
  clrscr ;
  textcolor(6) ;
  gotoxy(23,9) ;
  writeln('Hlupandi starð frá 1 til fjölda mælinga') ;
  gotoxy(23,10) ;
  writeln('Dagsetning samkvamt klukku PC tölvunar ') ;
  gotoxy(23,11) ;
  writeln('Tími samkvamt klukku PC tölvunar ') ;
end;

gotoxy(23,12) ;
writeln('Dagsenting samkvamt klukku söfnunartækisins') ;
gotoxy(23,13) ;
writeln('Númer innan dagsins') ;
gotoxy(23,14) ;
writeln('Sprunga L6') ;
gotoxy(23,15) ;
writeln('Sprunga L5') ;
gotoxy(23,16) ;
writeln('Spenna rafgeymis') ;
gotoxy(23,17) ;
writeln('Hitastigi') ;
gotoxy(23,18) ;
writeln('Hallamælin') ;
gotoxy(23,19) ;
textbackground(2) ;
textcolor(7) ;
writeln('HEFJA VINNSLU GAGNA') ;
gotoxy(10,24) ;
writeln('Notið ', #17,#196,#217,' ',#24,' #25,' til
textcolor(7) ;
textbackground(0) ;
for i := 0 to 9 do stjornur[i] := false;
gotoxy(20,9);
repeat
  ch := readkey ;
  if ch = #0 then
    case readkey of
      #80 :
        begin
          if wherey < 19 then
            gotoxy(wherex ,wherey + 1) ;
          end;
        #72 :
          begin
            if wherey > 9 then
              gotoxy(wherex ,wherey -1) ;
            end ;
          end ;
        if (ch = #13) and not (wherey = 19) then
          begin
            if stjornur[wherey - 9] then
              write(' ')
            else write('*') ;
            stjornur[wherey - 9] := not stjornur[wherey - 9] ;
            gotoxy(wherex - 1,wherey + 1) ;
          end ;
        end;

```

```

ch := #0 ;
end ;
until (wherey = 19) and (ch = #13) ;
end;

procedure veljautskra ;
begin
  clrscr ;
  gotoxy(30,5) ;
  textbackground(2);
  gagn := drif + copy(gagn,1,6) + '.DAT' ;
  write('Hafn útskraar : ' ;
  textbackground(0);
  write(gagn) ;
  gotoxy(46,5) ;
  readln(utskra) ;
  if utskra = '' then utskra := gagn ;

assign(uts,utskra) ;
{$I-}
rewrite(uts) ;
nresult := iresult ;
{$I+}
if nresult <> 0 then
  begin
    clrscr ;
    gotoxy(20,15) ;
    textbackground(2);
    writeln('Skráin er ólæsileg !!!') ;
    textbackground(0);
    exit ;
  end ;

with maling do
  begin
    q := q + 1 ;

    if stjornur[0] then
      write(uts,q,' ' ) ;
    if stjornur[1] then
      write(uts,leidretting(dagsetn),' ' ) ;
    if stjornur[2] then
      write(uts,timi,' ' ) ;
    if stjornur[3] then
      write(uts,dagnr,' ' ) ;
    if stjornur[4] then
      write(uts,nrinnandags,' ' ) ;
    if stjornur[5] then
      write(uts,rasi,' ' ) ;
    if stjornur[6] then
      write(uts,ras2,' ' ) ;
    if stjornur[7] then
      write(uts,spenna,' ' ) ;
    if stjornur[8] then
      write(uts,hiti,' ' ) ;
    if stjornur[9] then

```

lesa.pas Page 4

```
write(uts,halli, ' ');
end ; (with)
writeln(uts) ;
end ; (while)
end ; (lesa)

BEGIN
  directory ;

  if nerror = 0 then veljautskra ;
  if nerror = 0 then velja ;
  if nerror = 0 then
  begin
    lesaskra ;
    close(uts) ;
  end ;
END.
```

## VIÐAUKI C

Listun á forritum í söfnunartæki

```
# SPRU make file used with Microsoft make utility
# EHH/1988
# accel is an CP/M emulator on PC
# M80 is an CP/M macroassembler for Z80 (M80.CPM)
# L80 is an CP/M linker (L80.CPM)
# C2 is Supersoft second pass C Compiler (C2.CPM)
# CC is the first pass used outside this make file (CC.CPM)
# The syntax for CC is "accel cc forrit.c".
# Must be done for C programs before this makefile is used

bios.rel: bios.mac
    accel m80 =bios

post.rel: post.mac    # Used by the C compiler
    accel m80 =post

endi.rel: endi.mac
    accel m80 =endi

sprus.rel: sprus.cod  # C subroutines used by spru.c
    accel c2 sprus.cod
    accel m80 =sprus.asm

spru.hex: spru.cod sprus.rel bios.rel pre.rel post.rel endi.rel
    accel c2 spru.cod
    accel m80 =spru.asm
    accel l80 /p:0, /d:2000, bios, spru, sprus/s, libc/s, post, endi, spru/n/x/y/e

# Program starts at address 0. Data starts at address 2000 hex
# Libc is the Supersoft C library.
# Endi is to mark end of file
# The output is a file with the name spru.hex and is burned into
# eproms (3 * 27C16)
```

```

/* Spru.c
   Stýriforrit fyrir sprungumæli ritað í Supersoft C - 56 -
   Orkustofnun, Jarðhitadeild
   Einar Hrafnkell Haraldsson, 2. maí til 27. október 1988
*/

```

```

#include stdio.h
extern int mtelja, mtimi, dagnum, stopp, holf;

main()
{
  int i, temp, dag, man, ar;
  char c, tmp, s[20];

  /* Texti á skjá í hvert skipti, sem forritið er keyrt */
  puts("Sprungumaelir vid Leirhnuk.\n\r");
  puts("Utgafa 1.00, 12/9/1988/EHH.\n\r");
  puts("Gefdu skipanir (H = Hjalp)\n\r");

  /* Forritið sækir skipanir til lykklaborðs. Ef líða um það bil 25
   sec. án þess að skipanir koma fer forritið í aflsnautt ástand.
   Það gerist með því að undirforritið getdi skilar %, sem er
   skipun um aflsnautt ástand. */

  while ((c = getdi()) != '%') /* Ná í skipun frá lykklaborði */
    switch(tolower(c))
    {
      case ' ': /* Þessi skipun gerir ekkert merkilegt */
        break;

      case 'e': /* Skoða innganga frá A/D breytu */

        analkv(); /* Kveikja á analog mögnurum */
        mon(); /* Kveikja á a/d bretti */
        puts("Ras1 = "); putdec(mconv1());
        puts(", Ras2 = "); putdec(mconv2());
        senkv(); /* Kveikja á sendi */
        puts(", V = "); putdec(mconv3());
        puts(", t = "); putdec(mconv4());
        lesklu();
        sensl(); /* Slökkva á sendi */
        analsl(); /* Slökkva á analog mögnurum */
        moff(); /* Slökkva á a/d bretti */
        break;

      case 'a': /* Stilla rauntímaklukku */

        stiklu(); /* Stilla klukku */
        lesklu(); /* Lesa klukku */
        break;

      case 'b': /* Skoða dagasetningu og tíma */

        lesklu(); /* Lesa klukku */
        break;

      case 'd': /* Sjá dagnúmer
        Vistirnar fe?? eru í rauntímaklukkunni MC146818,
        sem er á örtölvubrettinu. Sjá nánar í upplýsingum
        með því. Þegar klukkan er að telja áfram, þá er
        hún upptekin og ekki er hægt að lesa úr henni.
        (Klókara hefði verið að bíða á meðan). */

        if ((peek(0xfe0a) & 0x80) == 0)
        {
          dag = peek(0xfe07);
          man = peek(0xfe08);
          ar = peek(0xfe09) + 1900;
          dagnum = tojul(dag, man, ar, 1980);
          puts("Dagnumer er: "); putdec(dagnum); puts("\n\r");
        }
        else
        {
          puts("Reyndu aftur (Update in progress)\n\r");
        }
        break;

      case 'k': /* Kveikja á sendi */

        analkv();
        senkv(); /* Kveikja á sendi */
        puts(" Kveikt á sendi\n\r");
        break;

      case 's': /* Slökkva á sendi */

        analsl();
        sensl(); /* Slökkva á sendi */
        puts(" Slökkt á sendi\n\r");
    }
}

```

```
break;
```

- 57 -

```
case 'q': /* Stoppa mælingu. Það er gert með því að
setja 255 í external breytuna stopp. Í hvert skipti,
sem verður í grip-A er athugað hvert gildið er
í stopp. 0 = er í mælingu. */
```

```
stopp = 255; /* Stoppa mælingu */
puts(" Stopp - mælir ekki\n\r");
break;
```

```
case 'o': /* Núllstillta bendi í hringhólfi. Það
er gert með því að setja external breytuna
holf = 0. Ekkert er hins vegar gert við gögnin, sem
eru í hólfinu. */
```

```
puts("Ertu viss? j = ja\n\r");
tmp = getdi();
if (tolower(tmp) != 'j')
{ puts("Hringholf var ekki nullstillt\n\r");
break;
}
holf = 0; /* Núllstillta geymsluhólf */
puts("Hringholf var nullstillt\n\r");
break;
```

```
case 'w': /* Starta mælingu */
```

```
stopp = 0; /* Starta mælingu */
puts(" Mæling hefst\n\r");
break;
```

```
case 'm': /* Ýmsar stærðir í forritinu */
```

```
puts("-----\n\r");
puts("Fjöldi inta er "); putdec(mtelja);
puts("\n\rStopp = "); putdec(stopp);
puts("\n\rHringgeymsla = "); putdec(holf);
puts("\n\r");
puts("-----\n\r");
break;
```

```
case 't': /* Útlestur á gögnum úr hringhólfi. Hólfið
hefst í vist 2800 hex og endar í vist 2fff hex. Alltaf
er lesið út úr öllu hólfinu. Bendirinn inn í hólfið,
breytan holf er ekki núllstillt þannig að næsta mæling
lendir aftanvið þá síðustu nema skipunin o hafi verið
framkvæmd. */
```

```
puts("Útlestur ur Sprungumæli vid Leirhnuk.\n\r");
puts("Útlesturinn hefst ");
lesklu();
puts("Byrjun\n\r");
for (i = 0x2800; i < 0x3000; i++)
{ putdec(peek(i));
puts(" ");
}
puts("\n\rEndir.");
puts("\n\rÚtlestrinum lokid kl. ");
lesklu();
break;
```

```
case 'h': /* Hjálp */
```

```
puts("Skipanir:\n\r");
puts("A -> Stilla dagatal og klukku\n\r");
puts("B -> Sja klukku og dagatal\n\r");
puts("D -> Sja dagnumer fra 1/1/1980\n\r");
puts("E -> Skoda innganga a a/d breytu\n\r");
puts("H -> Hjalp\n\r");
puts("K -> Kveikja a sendi\n\r");
puts("O -> Nullstillta Geymsluholf\n\r");
puts("M -> Ymsar staerdir\n\r");
puts("Q -> Stoppa mælingu\n\r");
puts("S -> Slokkva a sendi\n\r");
puts("T -> Taema ur geymsluholfi\n\r");
puts("W -> Starta mælingu\n\r");
puts(" -> Gera ekkert merkilegt\n\r");
puts("% -> Skila forritinu i aflsnaudu formi\n\r");
break;
```

```
default: /* Skipanir, sem ekki eru til */
```

```
puts("**** Villa Ekki rett adgerd ***\n\r");
break;
```

```
}
```

```
puts("**** Bless ***\n\r"); /* Fer í aflsnautt ástand */
```

```
/* Sprus.c
   Safn undirforrita fyrir sprungumæli ritað i
   Supersoft C.
   Orkustofnun, Jarðhitadeild
   Einar Hrafnkell Haraldsson 1988
*/
```

```
/* Getdi les frá uarti með bdos kerfiskalli nr. 6.
   Ef stafur hefur ekki komið eftir 65000 köll,
   þá er skilað til aðalforritsins '%', sem er
   skipunin um að fara í aflsnautt ástand. Þessi
   tími er um 25 sec.
*/
```

```
char getdi()
{
    char cc;
    unsigned int utr;

    for (utr = 1; utr < 65500; utr++)
    {
        cc = bdos(6,255);
        if (cc != 0)
            break;
        if (utr > 65000)
        {
            cc = '%';
            break;
        }
    }
    return cc;
}
```

```
/* Getchar sækir staf til uarts með kerfiskalli 1
*/
getchar()
{
    return bdos(1,0);
}
```

```
/* Puchar sendir staf til uarts með kerfiskalli 2.
*/
putchar(c)
char c;
{
    return bdos(2,c);
}
```

```
/* Getdir er svipað og getdi en biður endalaust eftir stafnum.
*/
char getdir()
{
    char c;
    while(( c = bdos(6,255)) == 0);
    return c;
}
```

```
/* Putdir sendir staf til uarts með kerfiskalli 6
*/
putdir(c)
char c;
{
    return bdos(6,c);
}
```

```
/* getline.c
   Forritið les inn línu í s af lengdinni lengd
   K&R bls. 67.
*/
```

```
#define EOFF 26          /* cdos EOF ctrl Z */
```

```
getline(s, lina)
char s[];
int lina;
{
    int c, i;
    i = 0;

    while (--lina > 0 && (c=getchar()) != EOFF && c != '\n' && c != '\r')
        s[i++] = c;
```



```
        s[i++] = c;
s[i] = '\0';
return(i);
}

/* Askint les inn integer tölu frá skjá
*/
int askint(s)
char s[];
{
    int i, num, form;

    getline(s, 15);
    i = 0;
    form = 1;

    if (s[i] == '-')
        form = -1;
    if (s[i] == '+')
        form = +1;
    for (num = 0; s[i] >= '0' && s[i] <= '9'; i++)
        num = 10 * num + s[i] - '0';

    return (num * form);
}
```

```
/* Stiklu stillir klukku á tölvubrettinu
*/
stiklu()
{
    char s[20];
    puts("Year: ");
    poke(0xfe09, askint(s));
    puts("\n\r");
    puts("Mounth: ");
    poke(0xfe08, askint(s));
    puts("\n\r");
    puts("Dagur: ");
    poke(0xfe07, askint(s));
    puts("\n\r");
    puts("Hour: ");
    poke(0xfe04, askint(s));
    puts("\n\r");
    puts("Minutes: ");
    poke(0xfe02, askint(s));
    puts("\n\r");
    puts("Seconds: ");
    poke(0xfe00, askint(s));
    puts("\n\r");
}
```

```
/* Lesklu les tíma og dagsetningu frá klukku á tölvubretti.
*/
lesklu()
{
    if ((peek(0xfe0a) & 0x80) == 0)
    {
        puts("\n\r19");
        putdec(peek(0xfe09));
        puts("-"); /* Year */
        putdec(peek(0xfe08));
        puts("-"); /* Mounth */
        putdec(peek(0xfe07));
        puts(" "); /* Day */
        putdec(peek(0xfe04));
        puts(":"); /* Hours */
        putdec(peek(0xfe02));
        puts(":"); /* Minutes */
        putdec(peek(0xfe00));
        puts("\n\r"); /* Secounds */
    }
    else
    {
        puts("Update in progress\n\r");
    }
}
```

/\* Fallið tojul breytir dagsetningu í fjölda daga talið frá grunnári.  
20. janúar 1988 Einar Hrafnkell Haraldsson.  
Stuðst er við forrit, úr bókinni Common C Functions eftir Brand  
Gallar: 1) Ekki er athugað hvort dagar eru leyfilegir né heldur  
hvort grunnár er lægra en árið, sem spurt er um.  
2) Ekki er tekið tillit til afbrigða í sögulegu tímatali  
yngra 10 daga, sem felldir voru úr á Íslandi frá 17.-26.

nóvember árið 1700 né heldur þess að árið 0 var ekki til.  
3) Forritið notar 16 bita integertölur, sem geta gefið  
65536 daga eða um 179 ár. Þannig má dagsetning ekki vera  
lengra frá grunnári en um 179 ár.

\*/

```
int tojul (day, month, year, base)
int day, month, year, base;
{
int n, retjul;
static int days[13];
days[0]=0; days[1]=31; days[2]=28; days[3]=31; days[4]=30; days[5]=31;
days[6]=30; days[7]=31; days[8]=31; days[9]=30; days[10]=31; days[11]=30;
days[12]=31;

if( (year % 4 == 0 && year % 100 != 0) || year % 400 == 0)
    days[2] = 29;
else
    days[2] = 28;

retjul = day;          /* Start with current month */
for (n = 1; n < month; n++)
    { retjul += days[n];
    }

for (n = base; n < year; n++)
    { if( (n % 4 == 0 && n% 100 != 0) || n % 400 == 0)
        retjul += 366;
      else
        retjul += 365;
    }

return (retjul);
}
```

```

;
; Forrit fyrir sprungumaeli.
; Orkustofnun, Jarðhitadeild
; Einar Hrafnkell Haraldsson 1988
;
; Forrit ritad fyrir M80 assembler fra Microsoft
; fyrir MSC800 (Z80) ortolvu.
; Skyringar i forritinu eru ritadar a bjagadri
; islensku vegna tess ad M80 assemblerinn litur
; a serislenska stafi sem villu i forritinu og
; er tvi ekki haegt ad nota ta. Sums stadar eru
; skyringar a ensku.
;
; Forritid skiptist i nokkra hluta:
;
; 1) Bios likir eftir nokkrum kerfiskollum ur
; CPM styrikerfinu.
; 2) Kaldstart.
; 3) Igripi-A.
; 4) Undirforritum.
;
;
; .XLIST
; .Z80
;
; =====
; BIOS
; =====
;
; Hluti af bios er tekinn ur forritum fyrir viðnámsmóttakara.
; Fyrst eru tad skilgreiningar.
;
CR EQU 0DH
LF EQU 0AH
BS EQU 08H
SPACE EQU 20H
DEL EQU 7FH
BEL EQU 07H
;
; CPM I/O-INTERFACE
;
; Her er likt eftir CPM-styrikerfi hvad vardar
; eftirtaldar "functionir"
;
; 0: System reset
;
;
; 1: Console input
; 2: Console output
; 5: List output
; 6: Direct console I/O
; 9: Print string
; 10: Read consle buffer
; 11: Get console status
;
; I tessari eftirlíkingu er ekki tekið tillit til
; I/O-ordsins i CPM
;
; -----
; public bios
;
; ORG 0
;
; bios: JP CSTART ; COLD-START
; DB 0 ; I/O-ORD
; DB 0 ; AUKA BYTE
; JP CPMIO ; CPM-INTERFACE
;
; org 02ch
; jp intc
;
; org 034h
; jp intb
;
; org 038h
; jp int
;
; org 03ch
; jp inta ; Inta service routine
;
; org 066h
; jp nmi
;
; ORG 80H
;
; CPMIO:
; Velja function. Numer functionar er i C-REG.
; Ef functionsnumer er staerra er 11 ta skal hoppad i
; "NOFUNC".
; LD A,C
; CP 12
; JP NC,NOFUNC
;
; -----

```

```

LD HL,FUNCTA ; HOPP-TAFLA
LD B,0
ADD HL,BC
ADD HL,BC ; HL BENDIR A VECTOR
PUSH DE
LD E,(HL)
INC HL
LD D,(HL)
EX DE,HL ; HOPP-VECTOR I HL-REG
POP DE
JP (HL)

FUNCTA:
; HOPPTAFLA. FYRIR OSKILGREINDAR FUNCTIONIR ER HOPPAD
; I "NOFUNC".
DW WSTART ; < 0> System reset
DW CONIN ; < 1> Console input
DW CONOUT ; < 2> Console output
DW NOFUNC ; < 3> Reader input
DW NOFUNC ; < 4> Punch output
DW LISTUT ; < 5> List output
DW DIRCON ; < 6> Direct console I/O
DW NOFUNC ; < 7> Get I/O byte
DW NOFUNC ; < 8> Set I/O byte
DW PRISTR ; < 9> Print string
DW RECOBU ; <10> Read console buffer
DW CONSTA ; <11> Get console status
; -----
NOFUNC:
; OSKILGREIND FUNCTION
SUB A
RET
; -----
WSTART:
; SYSTEM RESET
JP 0
; -----
CONIN:
; Console input
; FUNCTION 1
; Fyrst er bedid eftir ad takn se tilbuid
CALL CONSTA
JR Z,CONIN
; Data er tilbuid. Na i tad og senda tad
; jafnframt i output driver.
in a,(RXDATA)
AND 7FH
LD E,A
PUSH AF
CALL CONOUT
POP AF
RET
; -----
CONOUT:
; FUNCTION 2
; Bida eftir ad transmitter buffer verdi tomur
; Gert er rad fyrir takni i E-register.
in a,(UARTST)
AND TRBUEM
JR Z,CONOUT
LD A,E
out (TXDATA),a
RET
; -----
LISTUT:
; Sendir takn ut a prentarann.
JP NOFUNC
; FUNCTION 5
; -----
DIRCON:
; FUNCTION 6
; Direct console I/O. Ef E-reg inniheldur OFFH ta
; er um "input function" ad raeda, annars skal innhald

```

```

; E-reg sent ut a console.
LD A,E
INC A
JP NZ,CONOUT

; "Input function". Ef ekkert takn er tilbuid skal
; skila 0 i A-reg en amars takninu.
DIRIN: CALL CONSTA
RET Z
in a,(RXDATA)
AND 7FH
RET

; -----
PRISTR: ; FUNCTION 9
; Prentar ut strenginn sem DE-reg bendir a.
; Strengurinn haettir vid fyrsta "$".
EX DE,HL
PRI_HL: LD A,(HL)
CP '$'
RET Z
LD E,A
CALL CONOUT
INC HL
JR PRI_HL

; -----
RECOBU: ; FUNCTION 10
; "Read console buffer".
; Les takn fra console og hledur teim i (DE+2),
; (DE+3), o.s.fr. I (DE) er mesti fjoldi takna sem
; tekid skal a vid og i (DE+1) er fjoldinn sem
; kominn er a hverjum tima. Vid <CR> og <LF> er
; mottoku haett og eins ef fjoldi takna er kominn
; i hamark. Vid <DEL> og <BS> er seinasta takn
; turkkad ut ur buffernum og <BS><SP><BS> sent
; a terminalinn.
EX DE,HL
PUSH HL ; Bendir a mesta fjolda
INC HL
LD (HL),0 ; Nuverandi fjoldi

RELOOP:
; Na i takn fra console
CALL DIRIN
JR Z,RELOOP

; Ef takn er <LF> eda <CR> ta skal haetta
CP LF
JR Z,RCBEND
CP CR
JR Z,RCBEND

; Ef takn er <DEL> eda <BS> ta skal turkka
; ut seinasta takn, nema buffer se tomur
CP DEL
JR Z,DELETE
CP BS
JR Z,DELETE

; Takn er ekki <LF>,<CR>,<DEL>, eda <BS>. Se tad ekki
; "printable" skal senda <BEL> a console
CP ZOH
JR NC,PRINTA
LD E,BEL
CALL CONOUT
JR RELOOP

PRINTA:
; Baeta takni vid buffer og senda tad jafnframt a console
INC HL
LD (HL),A
LD E,A
CALL CONOUT

; Auka vid taknateljara og se hann jafn og
; heildarfjoldi ta skal haetta
EX (SP),HL
LD A,(HL) ; Hamarksfjoldi i A-reg
INC HL
INC (HL)
CP (HL)
JR Z,RCBEND

; Buffer er ekki fullur. Setja bendara a hamarksfjolda
RETREL: DEC HL
EX (SP),HL
JR RELOOP

DELETE:
; Turkka ut seinasta takn. Ef buffer er tomur
; er ekkert gert.

```

```

EX      (SP),HL
INC     HL
LD      A,(HL)
AND     (HL)
JR      Z,RETREL

; Buffer ekki tomur. Deckrementa teljara og
; senda <BS><SP><BS> a console
DEC     (HL)
LD      DE,BSSPBS
PUSH   HL
CALL   PRISTR
POP    HL
JR     RETREL

RCBEND:
; Hreinsa stakkinn og snua heim
POP    DE
RET

; <BS><SP><BS> strengur
BSSPBS: DB BS,SPACE,BS,"$"

; -----
; -----
CONSTA: ; FUNCTION 11
; Console status. Ef <DA> flag i STATUS REGISTER er
; "1" ta skal skila A-reg med OFFH en annars med 0
in      a,(UARTST)
AND     DATARE
RET     Z
SUB     A ; ZF=0
DEC     A
RET

; -----
; -----
; UART definitions
; -----
qdart  equ 0 ; 0 = ekki notadur
acr10  equ 1 ; 1 = notadur

if qdart
; Skilgreiningar a hlidum a oDART-breitti.
UARTEN EQU OD4H ; BIT-0 HIGH
; -----
; -----
TXDATA EQU OD0H ; TRANSMIT DATA
RXDATA EQU OD0H ; RECEIVE DATA
UARTCO EQU OD1H ; UART CONTROL
UARTST EQU OD1H ; UART STATUS

TX2DAT EQU OD2H
RX2DAT EQU OD2H
UAR2CO EQU OD3H
UAR2ST EQU OD3H

DATARE EQU 2 ; DATA-READY-BITI
TRBUEM EQU 1 ; TRANSMITTER-BUFFER-EMPTY

endif
if acr10
UARTEN EQU 0c0H ; BIT-0 HIGH
TXDATA EQU 0c4H ; TRANSMIT DATA
RXDATA EQU 0c5H ; RECEIVE DATA
UARTCO EQU 0c6H ; UART CONTROL
UARTST EQU 0c7H ; UART STATUS

DATARE EQU 1 ; DATA-READY-BITI
TRBUEM EQU 080h ; TRANSMITTER-BUFFER-EMPTY

endif

; -----
; -----
; Set up UART
; -----
; -----
public usetup
; Kveikir a uart og frumstillir hann
usetup:
if qdart
; UART1 og 2 settur af stad. Fyrst er UART gerdur virkur.
; Tad er gert med tvi ad setja BIT0 i <UARTEN>
; i stodu 1
LD      A,1
out     (UARTEN),a
; NU TARF AD BIDA I CA. 2 SEK A MEDAN BRETTID
; JAFNAR SIG
LD      BC,0F000H
CALL   BIDABC
; Ta er tad kontrolregisterid
LD      A,11001110B
**
; 2 STOPPBITAR
; -----
; -----

```



```

call iosetu      ; Stilla io port
call analsl     ; Slokkva a analognurnum eftir reset
call sensl     ; Slokkva a sendi eftir reset
ld a,2         ; Biti 2 i porti a er ljós
ld (anidur),a  ; Slokkva a tvi

ld hl,0
ld (mtelja),hl
ld (stopp),hl

ld a,255
ld (gamdag),a

xor a
ld (numida),a
ld (mtimi),a

CALL USETUP
call her
call vers
ld a,6 - 1
ld (mtimi),a
call inta

call mon
call mconv1
call mconv2
call mconv3
call mconv4
call moff

public start
start: ld a,0fh
out (0bbh),a

;-----
;
;
;
;
Setja upp klukkuras.
;-----
ld a,20h
ld (0fe0ah),a
ld a,26h
ld (0fe0bh),a

ld a,0ffh
ld (0fe05h),a

```

```

; Minutes, ff = gera interrupt
; 0 = ekki interrupt vid sec.
; Seconds
ld (0fe03h),a
xor a
ld (0fe01h),a
ld a,(0fe0ch) ; Clear first int from RTC
ei
;-----
; ext ccstar ; C-Coldstart
; jp ccstar
; Pre.mac 15.12.1987/ehh
; This file is used with Supersoft C-compiler
ext MAIN ; I C forritinu
ext CCCPU ; I post
public CCXII, CCSTAR

CCSTAR: ld hl, CCCPU ; notad i Supersoft C forritinu
ld (hl),1 ; 1 for z80

public leti
; I leti er beðið a milli interrupta.
leti: nop
nop
nop
ld a,(porta)
bit 2,a
call z,fmain

ld a,70
call delay ; Bíða í 70 ms áður en slökkt er.
call uaroff ; Slökkva á uartinum

;
;
ld hl,0fa00h ; Fyrir wait
ld hl,0f800h ; Fyrir hypernate
ld (hl),a ; fara i power down mode
nop
nop
nop
nop

```





```

ld      a,1          ; velja ras 2
call   mconv        ;
ld      (hl),d      ; msb hirt ur
inc    hl

call   senkv        ; Kveikja a sendi
ld      b,4
ld      a,225       ; Bida i 4*225ms=900ms
call   delay
djnz  meira

ld      a,2          ; velja ras 3, sem er spenna a rafflodum.
call   mconv
ld      (hl),d
inc    hl

ld      a,3          ; Velja ras 4, sem er termistor.
call   mconv
ld      (hl),d
inc    hl

call   analst
call   moff

equ    lina
numr   lina+2
ras1   lina+3
ras2   lina+4
ras3   lina+5
ras4   lina+6

ld      hl,tmpbuf   ; Geymir linu a decimal formi
ld      bc,(dagnr)  ; Dagnumer
call   binds        ; Binary to decimal conversion

ld      bc,(numr)   ; Numer innan dags.
ld      b,0
call   binds        ; Binary to decimal conversion

ld      bc,(ras1)   ; Maeligildi rasar 1
ld      b,0
call   binds        ; Binary to decimal conversion

ld      bc,(ras2)   ; Maeligildi rasar nr 2
ld      b,0
call   binds        ; Binary to decimal conversion

ld      bc,(ras3)   ; Maeligildi rasar 3, (12 Volt)
ld      b,0
call   binds        ; Binary to decimal conversion

ld      bc,(ras4)   ; Maeligildi rasar nr 4, (termistor)
ld      b,0
call   binds        ; Binary to decimal conversion

ld      a,0dh       ; Cr
ld      (hl),a
inc    hl           ; Lf
ld      a,0ah
ld      (hl),a
inc    hl
ld      a,$!
ld      (hl),a

call   useup        ; Kveikja á uartinum
ld      de,tmpbuf
ld      c,9
call   0005         ; Prenta streng

ld      a,250       ; bida i 250 ms.
call   delay
call   sensl        ; slokkva a sendi

; Geyma gogn i geymsluholfi einu sinni a solarhring, sem er
; til daemis tegar numer maelingar er = 1
ld      a,(numida)
ld      b,1
cp     b
call   z,holgey     ; Geyma gogn i hringholfi ef
; Numer innan dags er 1

; -----
; extrn intac      ; Ritad i C
; push af
; push bc
; push de
; push hl
; call intac
; pop hl
; pop de
; pop bc
; pop af

```

```

xor      a          ; Nullstillta talningu inta a milli maelinga.
ld      (mtimi),a

;
;   extrn  lesklu    ; tmp skrifad i C
;   call  lesklu    ; ritar klukku a skja.

endint:  ex  af,af'
exx
ej

reti

; =====
;   UNDIRFORRIT
; =====
; -----
;   HER og VERS prenta strengi a skjainn.
; -----

public  her
equ    0dh
equ    0ah
beep   equ    7

her:   ld      de, string
       ld      c,9
       call   0005
       ret

string: db  'Orkustofnun JHD-JES',beep,cr,lf,$'

vers:  ld      de, utgafa
       ld      c,9
       call   0005
       ret

utgafa: db  'Version 1.00, 12/9/1988/EHH', cr,lf,$'

; -----
;   Undirforrit, sem finnur dagnumer ut fra dagsetningu i
;   MC146818 rauntimaklukku.
;   Kallar a undirforrit, sem er ritad i C.
; -----
grunar equ    1980

; -----
;   HOLGEY geyma gogn i hringgeymslu.
;   Hver faersla er:
;   Dagnumer = 2 byte
;   Ras1     = 1 byte
;   Ras2     = 1 byte
;   Samtals = 4 byte, sem er um 1,5 ar.
; -----

public  holgey
ld      bc,(dagnr)
ld      a,c
call   holtil
ld      a,b
call   holtil

ld      bc,(ras1)
; Reg c = gildi rasar 1

```

```

ld      a,c
call   holtil

ld      bc,(ras2)
ld      a,c
call   holtil

ret

;-----
; Geyma gogn i hringeymslu. Geymslan byrjar i holbyr (=2800)
; og er 2048 byte long.
; Ath. bendirinn holf er ekki nullstilltur eftir reset.
;-----
; Holttil tekur vid gognum i reg a og haekkar bendi a naesta
; lausa plass.
;-----
holti: ld      hl,(holf)      ; Bendir a naesta lausa plass
ld      de,holbyr
add     hl,de
ld      (hl),a
inc     hl
ld      a,7                ; hringholf med 2048 bytum
and     h
ld      h,a
ld      (holf),hl        ; Geyma bendinn a naesta lausa holf
ret

;-----
; Nokkur undirforrit, sem tengjast portum i NSC801
;-----
nsc801 equ      0fd00h      ; Base address for NSC801
porta  equ      nsc801      ; Port a
ddra   equ      nsc801+4    ; Data direction for port a
ddrc   equ      nsc801+6    ; Data direction for port c
mdr    equ      nsc801+7    ; Mode definition register for port a
anidur equ      nsc801+8    ; Port a bit clear
aupp   equ      nsc801+0ch  ; Port a bit set

; a0 = styra sendi utgangur
; a1 = ljós, sem kviknar vid hvert inta
; a2 = takki, sem er skodadur við hvert inta
; a3 = takki fyrir profunarmaelingu (maelt a 6 sec fresti).
;-----
; a4 = kveikir a analog mognumum.
; Stillir port i NSC 810
; setu: xor      a
ld      (mdr),a          ; Porta basic i/o
ld      a,00010011b     ; a7-a0, iiiiuiuu
ld      (ddra),a
ld      a,0ffh          ; Gera port c ad utgangi
ld      (ddrc),a        ; Eydir minnsta straum tannig.
ret

public senkv, sensl, analkv, analsl

; Kveikja a sendi
senkv: ld      a,1
ld      (aupp),a
ld      a,10            ; Bida i 10 ms
call   delay
ret

; Slokkva a sendi
sensl: ld      a,1
ld      (anidur),a
ret

; Kveikja a analog mognumum
analkv: ld     a,00010000b
ld      (aupp),a
ret

; Slokkva a analog mognumum
analsl: ld     a,00010000b
ld      (anidur),a
ret

;-----
; ad.mac 31.jan. 1988 Einar Hrafnkell Haraldsson
; Nokkur undirforrit fyrir MOR 800 A/D breytu
; fra ONSET.
;-----
mheima equ      080h      ; Address in i/o space
mcntrl equ      mheima    ; Read: A/D status
                                ; Write: d0,d1 = channel number
                                ; d3 = 0->8 bit, 1->12 bit
mormsb equ      mheima+1  ; Msb result
morlsb equ      mheima+2  ; Lsb result
meoc     equ      mheima+8 ; Bit 7 and 0 is zero if end of conversion

```

```

mpower equ mheima+8 ; Power control
mon:
public mon
ld a,80h ; Power up the a/d converter
out (mpower),a ; Power enable bit
ld a,5
call delay ; Wait for 5 ms to allow the voltage to settle
ld a,0ffh ; Enable control lines to a/d converter
out (mpower),a
ld a,0
call mconv ; Make one warm up conversion
ret

;
;
mconv:
public mconv
Entry: Reg A = channel number [0,1,2,3]
Exit: Reg D = Msb, Reg E = Lsb
or 8h ; 12 bit a/d conversion
out (mctrl),a ; Start a/d conversion
ld a,1
call delay ; wait 1 ms
in a,(meoc) ; Bit 7 is zero if eoc
rla
jr c,wteoc ; Wait for eoc (hafa max teljara)

in a,(morlsb)
ld e,a
in a,(mormsb)
ld d,a
ret

moff:
public moff
xor a
out (mpower),a
ret

; A/D breyta gera maelingu a ras 1-4. Skilar gildi i reg hl.
; Fyrir kall fra C-forriti
mconv1:
ld a,0
call mconv
ex de,hl ; Skilar gildi i reg d/e = msb/lb
ld l,h ; C forritid vill fa i reg hl
ld h,0 ; Skila efri 8 bitumnum i 0
ret

mconv2:
ld a,1
call mconv
ex de,hl
ld l,h
ld h,0
ret

mconv3:
ld a,2
call mconv
ex de,hl
ld l,h
ld h,0
ret

mconv4:
ld a,3
call mconv
ex de,hl
ld l,h
ld h,0
ret

;-----
; Delay.mac 31.Jan. 1988 Einar Hrafnkell Haraldsson
; Purpose: Delay from 1 to 256 milliseconds
; Entry: Reg A = number of milliseconds to delay
; ; A = 0 equals 256 ms.
; ; Returns to calling routine after the specified delay.
; ; Registers used: AF
; ; Time: (1 ms) * (Reg A)
; ; Size: Program 51 bytes.
; ; No data memory required.
;-----
cpms equ 3000 ; 3000 = 3 MHz clock (in NSC800 the crystal is 6 MHz)
; ; 2000 = 2 MHz clock.....
delay:
; Do all but the last milliseconds
push bc ; 17 cycles for the users call
call dly ; 11 cycles
; 32 cy. to return from dly

; Do 2 less than 1 ms for overhead
ld b,+(cpms/50)-2 ; 7cycles
;-----
; ;
ldlp: jp ldly1 ; 10 cycles
ldly1: jp ldly2 ; 10 cycles
ldly2: jp ldly3 ; 10 cycles
ldly3: add a,0 ; 7 cycles

```

```

; Exit in 33 cycles
pop bc ; 10 cycles
ld a,(delay) ; 13 cycles
ret ; 10 cycles
;=====
; 33 cycles

;-----
; Routine: dly
; Purpose: Delay all but last ms
; Entry: Reg A = total number of ms
; Exit: Delay all but the last ms
; Reg. used: AF,BC
;-----
dly: dec a ; 4 cycles
ret z ; 5 cycles (Ret when done 11 cycles)
ld b, +(cpms/50)-1 ; 7 cycles
;=====
; 16 cycles

dlp: jp dly1 ; 10 cycles
dly1: jp dly2 ; 10 cycles
dly2: jp dly3 ; 10 cycles
dly3: add a,0 ; 7 cycles
dly: djnz dlp ; 13 cycles
;=====
; 50 cycles

; Exit in 34 cycles
jp dly4 ; 10 cycles
dly4: jp dly5 ; 10 cycles
dly5: nop ; 4 cycles
jp dly ; 10 cycles
;=====
; 34 cycles

binhex2.z80
; Breytingar fra binhex1.z80 :
; 1) datasvaedi afmarkad med DATA skipun
; 2) global skipunum baett inn
; PROGRAMS TO CHANGE FROM AND TO BIN, HEX, DEC,
; FROM ASMLIB.REL

```

```

;+ global ahex,adec
;+ global binh1,binh2,binh4
;+ global binds,bindf,bindb,bind
;
; AHX CONVERTS HEX STRING TO BIN NUMBER
AHX: LD HL,00000H
A0126: LD A,(BC)
CALL A0137
CP 010H
RET NC
ADD HL,HL
ADD HL,HL
ADD HL,HL
ADD HL,HL
ADD HL,HL
LD L,A
LD A,A
INC BC
JP A0126
;
; A0137: SUB 030H
CP 00AH
RET C
SUB 007H
RET
;
; BINH4: PUSH BC
LD A,B
CALL BINH2
POP BC
LD A,C
LD B,A
;
; BINH2: LD A,B
RRR
RRR
RRR
RRR
CALL BINH1
LD A,B
AND 00FH
ADD A,030H
CP 03AH
LD (HL),A
INC HL
RET C
DEC HL
ADD A,007H
LD (HL),A

```







```
/* Forritð breyta.c
Orkustofnun, JHD október 1988
Einar Hrafnkell Haraldsson.
```

- 75 -

Forritð breytir formi skrár úr söfnunartæki. Skráin úr söfnunartækinu hefur 4 dálka. Tveir fyrstu eru dagnúmer en tveir síðari eru gildi fyrir rás 1 og 2. Út skilar forritið skrá með 4 dálkum. Fyrsti er hlaupandi tala frá 1 upp í 512, næsti er dagnúmer og tveir síðustu eru rás 1 og 2.

```
*/
```

```
#include <stdio.h>

FILE *inn, *ut;
main(argc, argv)
int argc;
char *argv[];
{
    unsigned int dn1, dn2, nd, r1, r2;
    int i, t = 1, tt = 1;
    char c, lina[81];

    if (argc < 3)
    {
        printf ("Rita skal: breyta skrá-inn skrá-út\n");
        exit(0);
    }
    if ((inn = fopen(argv[1],"r")) == NULL)
    {
        printf("Get ekki opnað inn skrána.\n");
        exit(0);
    }
    if ((ut = fopen(argv[2],"w")) == NULL)
    {
        printf("Get ekki opnað út skrána.\n");
        exit(0);
    }
    printf("Leið 1: Innskrá án texta fremst og aftast.\n");
    printf("        Útskrá án texta fremst og aftast\n");
    printf("Leið 2: Innskrá eins og hún kemur frá söfnunartæki.\n");
    printf("        Útskrá með texta fremst og aftast\n");
    printf("Leið 3: Innskrá eins og hún kemur frá söfnunartæki\n");
    printf("        Útskrá án texta fremst og aftast\n");
    printf("Svaraðu með 1 2 eða 3 : ");
    c = getchar ();
    switch(c) {

    case '1':

        while( (fscanf(inn, "%u %u %u %u", &dn1, &dn2, &r1, &r2)) != EOF)
        {
            printf("%u %u %u %u \n", t++, (dn1+(256*dn2)), r1, r2);
            fprintf(ut,"%u %u %u %u \n", tt++, (dn1+(256*dn2)), r1, r2);
        }
        break;

    case '2':

        for (i = 1 ; i < 5; i++)
        {
            fgets(lina, 79, inn);
            printf("%s", lina);
            fprintf(ut, "%s", lina);
        }

        printf("!!\n");
        fprintf(ut, "!!\n");

    case '3':

        for (i = 1; i < 513; i++)
        {
            fscanf(inn, "%u %u %u %u", &dn1, &dn2, &r1, &r2);
            printf("%u %u %u %u \n", t++, (dn1+(256*dn2)), r1, r2);
            fprintf(ut,"%u %u %u %u \n", tt++, (dn1+(256*dn2)), r1, r2);
        }

        for (i = 1 ; i < 6; i++)
        {
            fgets(lina, 79, inn);
            printf("%s", lina);
            fprintf(ut, "%s", lina);
        }
        break;

    case '3':

        for (i = 1 ; i < 5; i++)
            fgets(lina, 79, inn);
    }
}
```

```
for (i = 1; i < 513; i++)
{
    fscanf(inn, "%u %u %u %u", &dn1, &dn2, &r1, &r2);
    printf("%u %u %u %u \n", t++, (dn1+(256*dn2)), r1, r2);
    fprintf(ut, "%u %u %u %u \n", tt++, (dn1+(256*dn2)), r1, r2);
}
for (i = 1 ; i < 6; i++)
    fgets(lina, 79, inn);
break;

default:
    printf("Veldu rétt (1, 2 eða 3) \n");
    exit(0);
    break;
}

fclose(inn);
fclose(ut);
}
```

```

/*
Dagnumner.c
Forritið reiknar dagnumner miðað við dagsetningu og öfugt.
Dagar eru taldir frá og með 1. janúar 1980.
Einar Hrafnkell Haraldsson 1988
*/

#include <stdio.h>
#include <memory.h>
main()
{
char *svar;
char buff[101];
unsigned dag, man, ar;
int jul;
int base;
char *revjul();

/* Fallið tojul breytir dagsetningu í fjölda daga talið frá grunnári
20. janúar 1988 Einar Hrafnkell Haraldsson.
Stuðst er við forrit, úr bókinni Common C Functions eftir Brand
Gallar: 1) Ekki er athugað hvort dagar eru leyfilegir né heldur
hvort grunnár er lægra en árið, sem spurt er um.
*/
}

char *revjul(jul, year)
int jul;
int year;
{
static int days[]={0,31,28,31,30,31,30,31,31,30,31,30,31};
static char date[11];
int days_year;
int n = 1;
/* setmem (date, sizeof(date), 0); */
memset (date, 0, sizeof(date));
days[2] = 28;

do {
if (year % 4 == 0 && year % 100 != 0 || year % 400 == 0)
days_year = 366;
else
days_year = 365;
year++, jul -= days_year;
} while (jul > 0);

year--, jul += days_year;
if (days_year == 366)
days[2] = 29;

do
jul -= days[nt++];
while (jul > 0);

--n, jul += days[n];

return (date);
}

printf("-----\n");
printf("Dagnumner miðast við 1. janúar 1980\n");
printf("Val 1 : Breyta dagsetningu í dagnumner.\n");
printf("Val 2 : Breyta dagnumneri í dagsetningu.\n");
printf("Val 3 : Hætta\n");
printf("Svarðu með 1, 2 eða 3 : ");
scanf("%s", svar);
switch(svar[0])
{
case '1':
printf("\nGeftu dagsetningu (dd mm áááá) : ");
scanf ("%d %d %d", &dag, &man, &ar);
printf ("Dagnumner er : %u\n", tojul(dag, man, ar, 1980));
break;

case '2':
printf ("\nGeftu dagnumner : ");
scanf ("%d", &jul);
base = 1980;
printf("Dagurinn er (dd mm áááá) %s\n", revjul(jul, base));
break;

case '3':
exit(0);
break;

default:
printf ("*****VILLA**** Veldu rétt (1, 2, eða 3) \n");
break;
}

```

- 2) Ekki er tekib tillit til afbrigða í sögulegu tímatali vegna 10 daga, sem felldir voru úr á Íslandi frá 17.-26. nóvember árið 1700 né heldur þess að árið 0 var ekki til.
- 3) Forritið notar 16 bita integer tölur, sem geta gefið 65536 daga eða um 179 ár. Þannig má dagsetning ekki vera lengra frá grunnári en um 179 ár.

```

*/
int tojul (day, month, year, base)
int day, month, year, base;
{
  int n, retjul;
  static int days[13];
  days[0]=0; days[1]=31; days[2]=28; days[3]=31; days[4]=30; days[5]=31;
  days[6]=30; days[7]=31; days[8]=31; days[9]=30; days[10]=31; days[11]=30;
  days[12]=31;
  if( (year % 4 == 0 && year % 100 != 0) || year % 400 == 0)
    days[2] = 29;
  else
    days[2] = 28;
  retjul = day; /* Start with current month */
  { retjul += days[n];
    }
  for (n = 1; n < month; n++)
  { retjul += days[n];
    }
  for (n = base; n < year; n++)
  {
    if( (n % 4 == 0 && n % 100 != 0) || n % 400 == 0)
      retjul += 366;
    else
      retjul += 365;
  }
  return (retjul);
}

```