

MINIDATOR
KONTROLLERAD
SMÄLTMATERIALHANTERING

LARS LÖFGREN

MINIDATORKONTROLLERAD SMÄLTMATERIALHANTERING

Lars Löfgren

Examensarbete vid
Institutionen för Reglerteknik, LTH, 1978

Handledare:
Jan Holst och Björn Wittenmark

INNEHALLSFÖRTECKNING

1.	Inledning	1
2.	Styrregler och systemkonstanter	4
3.	Programbeskrivningar	7
4.	Dummy reentrantlibrary	16
5.	Common-area	27
6.	Flödesscheman	30
	Björkschema	30
	ASC11	31
	BCD	32
	BL	33
	BLA	34
	FORD	35
	JVL	36
	KLAR	37
	KONTR	38
	LAMPA	39
	LAS	40
	MRED	42
	NOLL	43
	RA	44
	RAKNA	45
	RED	46
	SKRIV	47
	SP	48
	START	49
	STARUP	50
	STOPP	51
	UT	52
	VAG	53
	WRL	56
7.	Programlistningar	57
	ASC11	57
	BCD	59
	BL	60
	BLA	62
	FORD	64
	JVL	66
	KLAR	68
	KONTR	72

LAMPA	73
LAS	75
MRED	85
NOLL	87
RA	88
RAKNA	90
RED	94
SKRIV	95
SP	102
START	103
STARUP	106
STOPP	110
UT	111
VAG	113
WRL	116
8. Utskriftsprover	117
Appendix	
DIOMIN	A.1
IN	A.3
KNOCKA	A.4
LAES	A.5
REED	A.6
STDATA	A.9
UT2	A.11
WRITE	A.12

SAMMANFATTNING

Vid smältverk önskar man som regel att de material som hälls i smältugnen är så väl blandade som möjligt med varandra. I detta examensarbete beskrivs ett minidatorbaserat real-tidssystem som kontrollerar och styr ett sådant blandningsförfarande, samt den metod som Boliden AB använder vid blandningen av materialen. Systemet erhåller uppgifter ifrån operatören om vilka materialsorter som skall ingå, totala vikten för varje sort och blandningsnoggrannheten. Operatören kan när som helst under pågående blandning gå in och ändra materialsammansättningen. Dialogen mellan systemet och operatören sker dels med hjälp av en Teletype, dels med hjälp av tryckknappar som är belägna ute vid materiallossningsplatserna. Via en snabbskrivare lämnar systemet regelbundna uppgifter om 1) Hur långt pågående blandning har kommit. 2) Hur stor materialförbrukningen är för respektive material per månad och dygn. Systemet är i drift sedan 1977.

1 INLEDNING

I detta examensarbete skall vi studera ett datorsystem för reglering och uppbyggnad av en "bädd" med material för fyllande av en masugn. Reglering skall leda till att en så god blandning som möjligt av de ingående materialen sker och att så liten differens som möjligt uppnås från givna börvikter på de olika materialen.

En bädd är alltså en hög vari olika material ingår. Bädnen delas in i horisontella skikt, vilka kallas multiplar. Material som skall ingå i bädnen kan antingen lossas med truck från ett lager via en vågficka eller lossas direkt från järnvägsvagn. Material som lossas på det senare sättet kallas direktmaterial. Det finns två olika slags direktmaterial, de betecknas efter ursprungsarten. S för Stekenjokk och A för Aitik. Grundmaterial är det material som skall lossas först i varje multipel. Grundsats är den del av grundmaterialet som skall lossas innan den egentliga bäddningen börjar.

Systemet består av en Philips PM 852 dator med 32 K kärnminne, en Teletype för kommunikation med systemet, en PER-snabbskrivare för alla löpande utskrifter, våg med tillhörande viktindikator, två indikerings-tablåer, samt en manöverlåda med sifferdekader och tryckknappar för järnvägsslossningsplatsen. Systemet skall så väl som möjligt fördela och blanda de material, max tjugo stycken per bädd, som ingår i bädnen. Det skall även redovisa en dygnsförbrukning av varje material via en dygnsförbrukningsutskrift med efterföljande nollställning av dygnsregistret. Varje månad skall en månads förbrukningsutskrift ske med efterföljande nollställning av månadsregistret. Under en månad kan dock max 30 material registreras.

Kommunikation med systemet kan ske på tre olika sätt:

- 1 Genom att trycka på en knapp märkt "D" på Teletypen öppnar man en dialog med systemet. Dialogen är av fråge-svars typ, d v s datorn ställer en direkt fråga till operatören och väntar sedan på svar. Denna del finns närmare beskriven i avsnitt 3, under rubriken dialogprogram.

- 2 Tryckknappar vid vågficka. Det finns tre stycken tryckknappar och en omkopplare. Två tryckknappar används av truckföraren för att ändra förloppet av en bäddning. Den ena är märkt "NÄSTA MATERIAL" och användes således för att byta material utan att villkoren därtill är uppfyllda. Materialet kommer att beordras igen när alla övriga material i multipeln beordrats. Den andra är märkt "NÄSTA MULTIPEL" och används om man vill byta multipeln utan att villkoren därtill är uppfyllda. T ex om man använt knappen "NÄSTA MATERIAL" och man fortfarande inte önskar lossa materialet när det beordras igen, byter man alltså multipel genom att använda knappen "NÄSTA MULTIPEL".

Omkopplaren är till för att avbryta det automatiska vågförfarandet. Invägning (registrering) av lossad vikt kommer då att erhållas först när knappen märkt "DUNK" trycks in. Detta förfarande används om man kan befara att inte allt material i traktorskopan kan passera våggallret t ex på grund av frusna block av material. Truckföraren slår då om omkopplaren och först när han förvissat sig om att allt material har passerat ned i vågfickan tillåter han vägning genom att trycka in "DUNK" knappen.

- 3 Tryckknappar vid järnvägsslösningplatsen, en för vardera direktmaterialet. Dessa används då man vill meddela datorn att en vagn av materialet har lossats.

Systemet ger information till sin omgivning via fyra olika enheter:

- 1 Teletypen, se avsnitt tre under rubriken "Dialogprogram".
- 2 PER-snabbskrivare, där alla listor som behövs för dokumentering av en bädd och de av datorn beordrade förbrukningsutskriften presenteras.
- 3 Två indikeringstablåer. Dessa används för att tala om för den truckförare som lossar material till bädden i vågfickan:
 - a) Vilken multipel bädden har hunnit till.
 - b) Vilket material han skall lossa.

- c) Hur mycket han skall ha lossat totalt av beordrat material då pågående multipel är klar.
- d) Hur mycket han har lossat av beordrat material, a, b, c och d visar med hjälp av bladverk som stegas fram till önskat läge av datorn.
- e) Hur stor del av sin skopa han bör fylla nästa gång han hämtar beordrat material. Det finns tre alternativ: 1/1, 1/2 och 1/4 skopa.
- f) I vilket tillstånd vågen befinner sig:
 - röd lampa = våg upptagen
 - gul lampa = vänta
 - grön lampa = tömning tillåten.

- 4 Manöverlåda med tre sifferdekader och två tryckknappar. (Tryckknapparna se avsnitt 3 program JVL). På sifferdekaderna visas läget för direktmaterialen. För materialet från Stekenjokk visas två uppgifter, nämligen hur många vagnar som återstår att lossa och hur många som har lossats. För detta material kan dock max nio beordras för hela bädden. För det andra direktmaterialet Aitik visas endast hur många som bör lossas under pågående multipel. För detta material får mer än nio vagnar beordras för hela bäddar.

Systemet har plats för två bäddar samtidigt, dock kan bara en vara i drift åt gången. Då systemet arbetar i real-time kan emellertid operatören samtidigt som en bäddning sker, förbereda nästa bädd då all kommunikation via Teletype är utförbar. Man kan när som helst avbryta en bädd och starta en annan, bäddarnas status är nämligen lagrad i långa vektorer. Detta gör även strömavbrott ofarliga.

Fortsättningsvis kommer jag att behandla de styrvillkor som finns för systemet i avsnitt 2. I avsnitt 3 sker en beskrivning av de olika program som ingår i systemet med tonvikt lagd vid dialogprogrammet LAS. I systemet ingår även ett så kallat Dummy Reentrant Library (DRL). Dess mening och användning förklaras i avsnitt 4. Därefter kommer en förteckning över hur Common arean används, avsnitt 5. Slutligen kommer flödesscheman i avsnitt 6, programlistningar i avsnitt 7 och prov på utskrifter i avsnitt 8. Appendix innehåller programlistningar på program som ingår i systemet och är skrivna av andra personer.

2 STYRREGLER OCH SYSTEMKONSTANTER

Detta avsnitt kommer att behandla de variabler samt de styrregler som finns för systemet.

De variabler som finns är följande:

- a) Truckskopans storlek.
- b) Multiplarnas antal.
- c) Storleken av en järnvägsvagnslast.
- d) Vågens tareringsgräns.
- e) Första siffran i det tresiffriga bäddnumret.

I samband med de styrregler där dessa variabler ingår kommer deras funktion att närmare beskrivas.

De styrregler som finns gäller följande funktioner:

Grundsatsen och avbrutet material.

Material som lossas via truck.

Material som lossas ifrån järnvägsvagnar.

Vågfunktionen.

Val av bäddplats.

Blandningsnoggrannhet.

Grundsatsen och avbrutet material.

Generellt gäller för lossning av alla material att i den första multipeln får endast den del av grundmaterialet som kallas grundsatsen lossas. I de därefter följande multiplarna lossar man materialposterna i nummerordning. Har emellertid en materialpost blivit avbruten genom att tryckknappen "NÄSTA MATERIAL" tryckts in, beordras den åter efter den sista posten inom multipeln. Posterna 1, 4 --- 20 lossas via truck. Post 2 och 3 från järnvägsvagnar.

Material som lossas via truck.

För truckposterna har man tre lossningsmöjligheter, 1/1, 1/2 eller 1/4 skopa.

Hel skopa beordras så länge den resterande delen av beordrad mängd inom multipeln av materialet överstiger 87,5 procent av hel skopa. När resterande mängd ligger i intervallet 87,5 - 37,5 procent av hel skopa beordras halv skopa. Kvarts skopa beordras endast om man är i sista multipeln eller om resterande mängd av materialet inom bädden understiger hel skopa. Kvartsskopa beordras tills mindre än 12,5 procent av beordrad mängd av materialet återstår inom multipeln. Därefter tar man nästa materialpost. Ett material hoppas över om beordrad mängd inom multipeln understiger 12,5 procent av hel skopa. Av detta framgår att det största felet för ett material är 12,5 procent av hel skopa, d v s skopans storlek bestämmer noggrannheten i ton räknat. Skopornas storlek kan ändras och sättes vid programmets initiering till 15 ton. För att uppnå så jämn fördelning som möjligt av de trucklastade materialen över multiplarna dividerar man vid varje multipelbyte resterande mängd av materialen med resterande antal multiplar. Detta medför att ett fel som uppstått vid lossningen av ett material i föregående multipel fördelas jämnt över resterande antal multiplar och belastar inte enbart nästa.

Material som lossas ifrån järnvägsvagnar.

Vid start av bäddning så räknar man om börvikten i ton för materialposterna två och tre, de s k direktmaterialen till antal järnvägsvagnar. Detta sker genom att dividera börvikten med en variabel som anger ungefärlig vikt för ett järnvägsglass i ton, vid initiering av programmet sätts detta värde till 45 ton. För materialpost två gäller att det får vara max nio vagnsglass i hela bädden. Dessa får lossas helt godtyckligt under bäddningen. Materialpost tre lossas på lite annorlunda sätt och antalet vagnsglass får överstiga nio stycken. Dessa vagnsglass skall lossas så jämnt som möjligt över bädden och om det blir lass över från en multipel så adderas dessa istället till det antal som beordras i nästa multipel.

Vågfunktionen.

Max vikt på vågen är tjugo ton och invägning tillåts endast när resterande vikt i vågkärlet understiger tareringsgränsen, vid initiering sätts den till två ton. Vågprogrammet reglerar lossningen i vågen på så sätt att:

Om vikten i vågen överstiger två ton så tänds röd lampa på indikerings-tablå. Skulle trucken komma dit under denna period så upptäcker vågprogrammet detta genom en induktiv närvaroindikator och tänds gul "VÄNTA" lampa, förutom den röda lampan. När sedan vikten i vågkärlet understiger tareringsgränsen stoppas en bandtransportör och vågen tareras varefter grön lampa tänds.

Om trucken anländer efter det tareringsgränsen underskridits så erhåller den tömningstillstånd direkt. Skulle vikten i vågfickan ha kommit ned under fyrahundra kilo innan trucken anlant så stoppas bandtransportören och vågen tareras utan att närvaro av truck krävs. Grön lampa tänds också. Min vikt för registrering av vägning är ett ton och stillestånd anses råda, d v s allt material anses tömt i vågfickan om två avläsningar med tvåhundra ms mellanrum är lika. Vågens funktion enligt detta kan avbrytas med "DUNK" funktionen som beskrivits i inledningsavsnittet.

Val av bäddplats.

Den första siffran i bäddnumret anger vilken typ av bädd man skall bädda och vilken bäddplats som skall användas. 1 - 2 anger torkugnsbädd, 3 - 4 rostugnsbädd, 5 - 6 är reserver och har samma bäddplats. I bäddnumret ingår ytterligare två siffror som används som ett löpnummer under året. Av detta kan man se att 100 är minsta möjliga och 699 största möjliga bäddnummer.

Blandningsnoggrannhet.

En uppgift som direkt påverkar bäddens blandning är antalet multiplar, ju fler multiplar desto bättre blandning av materialen. Någon regel för hur man skall välja antalet multiplar finns inte. Man bestämmer antalet multiplar i varje enskilt fall.

Exempel:

Två material A och B om vardera två ton skall bilda en bädd bestående av ett antal multiplar. Först tar vi två multiplar. Materialen kommer då att lossas med ett ton A och ett ton B i första multipeln. Likadant kommer det att ske i den andra och materialen är blandade som ABAB. Väljer vi istället fyra multiplar kommer materialen istället att lossas med 0,5 ton vardera i varje multipel och blandningen blir ABABAB. Vi har fått en ökad blandning genom att öka antalet multiplar.

3 PROGRAMBESKRIVNINGAR

Detta avsnitt kommer att ge enkla beskrivningar av de program som ingår i systemet, med tyngdpunkten lagd vid dialogprogrammet. De program som ej återfinns i denna beskrivning eller endast är nämnda vid namn är antingen så små och enkla att de lättast förklaras med hjälp av flödes-scheman och programlistningar eller är skrivna av andra personer. Se avsnitt 6, 7 och Appendix.

Följande program kommer att behandlas:

Dialogprogram LAS.

Utskriftsprogram SKRIV.

Indikeringstablåer och lampor RA, BL, BLA, NOLL, UT och LAMPA.

Materialhantering RAKNA, KONTR, VAG, START, RED och FORD.

Järnvägslösning JVL och BCD.

Avslutande av bädd KLAR.

Bakgrundsprogram STARUP.

Dialogprogram LAS.

Dialogprogrammet är det program som operatören använder sig av vid all kommunikation med datorn via Teletype utom i ett fall, se beskrivning av KLAR-programmet. Själva programmet är uppbyggt som ett fråge-svars program där datorn hela tiden i möjligaste mån först kontrollerar operatörens svar och sedan vidtar åtgärder. Alla frågor ligger som datasatser för att man skall kunna utnyttja en speciell utskriftsrutin, vilken medger att svaret ges på samma rad. Programmet är uppdelat i arton olika delrutiner, varav många griper in i varandra. Dessa rutiner når operatören genom att trycka på en speciell knapp märkt "D" på Teletypen. Datorn frågar då vilket dialognummer som önskas och operatören ger ett tal mellan 1-30. Dessa trettio tal är indelade i tre grupper där 1-9 omfattar inläsning av materiallista, utstansning av materiallista och start eller stopp av bäddning. Nummer 10-19 omfattar olika utskrifter på Teletypen som operatören kan beordra. I 20-29 finns alla rutiner för de ändringar som är tillåtna. Som framgår av de tidigare uppgifterna är givetvis inte alla de trettio talen aktiva utan endast arton. Gruppindelningen är endast gjord för att

underlätta för operatören. De operationer som programmet tillåter operatören att utföra finns i nedanstående lista med respektive dialognummer:

- 1 Inmatning av materiallista via tangentbord.
- 2 Inmatning av materiallista via remsa.
- 3 Utstansning av materiallista.
- 4 Start av bäddning.
- 5 Stopp av pågående bäddning.
- 10 Utskrift av multipellista på Teletype.
- 11 Utskrift av materialförbrukning, samtliga material på Teletype.
- 12 Utskrift av materialförbrukning, enstaka material på Teletype.
- 13 Utskrift av bladverksindikering på Teletype.
- 14 Utskrift av gällande tareringsgräns, skoplastvikt och vagnslastvikt på Teletype.
- 20 Ändring av materialkod.
- 21 Ändring av börvikt.
- 22 Tillägg av materialpost.
- 23 Ändring av tareringsgräns.
- 24 Ändring av vagnslastvikt.
- 25 Ändring av skoplastvikt.
- 26 Ändring av datum och klockslag.
- 27 Nollställning av bäddningsregister.

För att få en liten uppfattning om hur programmet arbetar skall vi ta och titta litet på hur en dialog kan se ut och de kontroller som datorn gör, kursiv stil. Jag väljer dialog nummer ett som exempel.

DATOR

*Reservera teletypen.
Skriv fråga.*

DIALOGNUMMER?:

*Kontroll om giltigt nummer,
om ej hoppar programmet till
punkt nr 3. Utskrift av rubrik.*

OPERATÖR

Aktivera dialogprogrammet genom att trycka ned knappen märkt "D" på Teletype.

- 1 Efter varje svar ger operatören radframmatning, vagnretur, i fortsättningen förkortat LF/CR.

DATOROPERATÖRINMATNING AV MATERIALLISTA
VIA TANGENTBORD

*Kontroll om bäddregister ledigt,
om ej felutskrift: INGET BÄDD-
REGISTER LEDIGT. Sedan punkt
nr 3.*

BÄDDNUMMER?:

101 LF/CR

*Kontroll om heltal, annars
felutskrift: ER och vänta på
nytt värde. Kontroll om värde
99 x 699, annars punkt nr 3.
Om värde redan inläst fel-
utskrift: BÄDD REDAN INLÄST.
Sedan punkt nr 3.*

ANTAL MULTIPLAR?:

10 LF/CR

*Kontroll om heltal, annars
felutskrift: ER och vänta
på nytt värde.*

GRUNDSATSTON?:

50 LF/CR

*Kontroll om godkända tecken
för reeltals värde, annars
felutskrift: ER och vänta på
nytt värde.*

1 POST NR 1

GRUNDMATERIAL

MATERIALHÖG?:

31G LF/CR

*Kontroll om mindre än 4
tecken, annars felutskrift:
MAX 3 TECKEN och vänta på nytt
svar. Om operatör endast
svarade LF/CR gå till punkt 2
för då var föregående material
det sista. Kontroll så att inte
alla månads- och dygnsregistren
är upptagna. Det räcker att ett
är ledigt för att en bädd skall
kunna läsas in. Felutskrift:
MAX TRETTIO MATERIAL PER MÅNAD.
Sedan punkt nr 3.*

DATOROPERATÖR

BÖRVIKT TON?:

342.70 LF/CR

*Kontroll om godkända tecken
för reeltalsvärde, annars
felutskrift: ER och vänta på
nytt värde. Sortera in svaren
i det lediga bäddregistret,
kontrollera så att inte mer
än tjugo material läses in.
Vid tjugo avbryt inläsning
och gå till punkt nr 2,
annars hopp till punkt nr 1.*

POST NR 2

DIREKTMATERIAL

MATERIALHÖG?:

10S LF/CR

BÖRVIKT TON?:

0 LF/CR

POST NR 3

DIREKTMATERIAL

MATERIALHÖG?:

10A LF/CR

BÖRVIKT TON?:

649.21 LF/CR

POST NR 4

MATERIALHÖG?:

61C LF/CR

BÖRVIKT TON?:

30.78 LF/CR

POST NR 5

MATERIALHÖG?:

61E LF/CR

BÖRVIKT TON?:

363.13 LF/CR

POST NR 6

MATERIALHÖG?:

71N LF/CR

BÖRVIKT TON?:

298.87 LF/CR

DATOROPERATÖR

POST NR 7

MATERIALHÖG?:

73E LF/CR

BÖRVIKT TON?:

303.55 LF/CR

POST NR 8

MATERIALHÖG?:

761 LF/CR

BÖRVIKT TON?:

3.60 LF/CR

POST NR 9

MATERIALHÖG?:

LF/CR

*Post åtta sista material, skriv
materiallista punkt nr 2.*

2	DATUM	19761115	MATERIALLISTA
	KLOCKSLAG	12.23	
	BÄDDNUMMER	101	
	ANTAL MULTIPLAR	10	
	GRUNDSATS	50.00	

POSTNR	MTRLHÖG	HELA BÄDDEN BÖRVIKT TON
1	3EG	342.70
3	10A	649.21
4	61C	30.78
5	61E	363.13
6	71N	298.87
7	73E	303.55
8	761	3.60

ALLT OK?:

JA LF/CR

*Om svar = ja punkt nr 3,
annars kommer följande fråga:*

DATOROPERATÖR

ÄNDRING AV POST?:

*Om svar = ja genomförs nedan-
stående sekvens varefter man går
till punkt nr 2. Om svar = ja noll-
ställs det använda bäddregistret
och man går till punkt nr 3.*

POST NR?:

MATERIALHÖG?:

BÖRVIKT TON?:

3 DIALOG SLUT

Tag bort reservation av teletypen.

Varje gång man aktiverat dialogprogrammet så skrivs "DIALOG SLUT" när det lämnas. Alla svar kontrolleras i görligaste mån för att förhindra felaktiga svar och åtgärder från operatören. Programmet har därigenom blivit tämligen omfattande, vilket kan framgå av listningen. Dess programnamn är LAS.

Utskriftsprogram, SKRIV.

Programmet SKRIV sköter om alla de utskrifter som kommer på radskrivaren samt de utskrifter på Teletypen som rör bädden eller materialförbrukningsregisterna. Alla rubrikerna ligger i datasatser för att de lätt skall kunna paras ihop efter behov. Vissa utskrifter kan bara göras under pågående bäddning, programmet kontrollerar därför om anropet är utförbart, om ej så lämnas rutinen genast. Vid anrop av programmet reserverar man alltid snabbskrivaren och vid behov även Teletypen för att förhindra att operatören går in och förstör utskriften. I detta sammanhang kan nämnas att det så kallade DEAD-LOCK-problemet, som endast kan uppstå i samband med SKRIV har lösts med hjälp av en rutin ATTCHW ur Fortranbiblioteket. Denna rutin gör reservation av en enhet om den är ledig eller lägger anropande program i väntläge till så inträffar. Detta anrop sker inte i SKRIV utan i det anropande programmet vilket medför att

SKRIV alltid kommer att gå klart och återvända till det anropande programmet där frikoppling av reserverad enhet sker. Sannolikheten för DEAD-LOCK emellan programmen som så är så gott som obefintlig då systemet är väldigt rakt vid sin exekvering. SKRIV och ATTCHW ingår i DRL, se avsnitt 5.

Indikeringstablåer och lampor, RA, BL, BLA, NOLL, UT och LAMPA.

Programmet RA ställer upp bladverken och lastindikeringslamporna samt tändar röd "UPPTAGET" lampa på indikeringstablåerna vid vågen. Programmet används då en bädd startas och då man byter material eller multipel.

Programmet BL anropas då man vill ändra den information som finns på indikeringstablåernas bladverk. Programmet räknar ut hur många steg som bladverket skall stegas fram från synkroniseringsläget för att rätt information skall visas. Programmet har fyra olika delar beroende på vilken rad på indikeringstablåen som skall ändras.

Programmet BLA anropas av BL och lägger då ut det antal pulser (steg) som BL har bestämt. Före inläggning av information ställs raden till ett synkroniseringsläge. Detta för att minska felrisken. Alla bladverken i raden stegas parallellt med omväxlande positiva och negativa pulser. När ett bladverk fått de pulser det skall ha så förhindras ytterligare stegningspulser att påverka det. Programmet NOLL används för att nollställa godtycklig rad på bladverkstablåen eller hela tablåen.

Programmet UT lägger ut signalerna till bladverken och ser till att dessa har en tillräckligt lång pulslängd.

Programmet LAMPA styr lamporna på de stora indikeringstablåerna och anropas varje gång man vill tända eller släcka en lampa. Rutinen har sju olika nivåer, en för varje lampa, plus möjligheten att släcka alla lampor.

Materialhantering RAKNA, KONTR, VAG, START, RED och FORD.

Programmet RAKNA ställer upp bladverken och lastindikeringslamporna med hänsyn tagen till de värden som vägts in på vågen. Programmet kontrollerar om ett material är klart, om en multipel är klar, om alla multiplar är

klara och i så fall om alla järnvägsvagnar är lossade. Vid multipelbyte går programmet igenom alla material i bädden för att se om operatören vidtagit några ändringsåtgärder. Före multipelbyte anropas programmet KONTR som kontrollerar om något material har blivit avbrutet och skall beordras igen av RAKNA innan byte sker. Detta sker emellertid ej om nästa multipel beordrats via tryckknapp. Se inledning.

Vågprogrammet VAG:s funktion har behandlats i avsnittet om styrregler och tas ej upp här.

Programmet START används då en bädd startas och man särskiljer tre olika fall, nämligen nystart, omstart och start av bädd som är klar och där endast följesedlar till järnvägsvagnar skall läsas in. I fall ett kontrollerar startprogrammet om det finns tillräckligt med förbrukningsregister för de nya material som kan ingå i bädden. Varje material i bädden förbinds också med ett av månads- och dygnsregistren. Därefter räknar man om börvikten för materialen till järnvägsvagnar med hjälp av programmet RED, detta för att kunna beordra direktmaterialpost två och tre. Programmet START kontrollerar vidare om transportbandens beordrade inställning blivit gjord. Om så ej skett lägger sig programmet i en 30 sekunders slinga som larmar vid varje genomlöpning via utskrift på Teletypen. Innan detta sker har emellertid en bäddningslista skrivits ut på snabbskrivaren. I fall två tilldelar man inte materialsorterna något månads- eller dygnsregister. Man utför inte heller omräkning till järnvägsvagnar via RED. Start av bäddning sker också direkt på det material där bäddningen blev avbruten. Fall tre medför endast att KLAR-programmet aktiveras. Se avsnitt om avslutande av bädd. START anropas från LAS.

Programmet FORD aktiveras varje gång en bäddning startas första gången vid multipelbyte och vid materialbyte inom multipeln. Programmets uppgift är att räkna ut hur mycket av ett material som skall lossas totalt inom multipeln. Det kontrollerar också om materialet får lossas, d v s det finns tillräckligt av materialet, eller om materialet inte har stängts av via någon åtgärd från operatören. Om vi befinner oss i första multipeln kommer programmet emellertid endast att släppa fram den första materialposten vilken beordrats i enlighet med grundsatsens storlek.

Järnvägsslossning JVL, BCD

JVL är det program som anropas då man använder tryckknapparna vid järnvägsslossningsplatsen. Programmet har tre olika funktioner beroende på värdet av registret ISK i common-arean, se avsnitt 5, ISK = 0 Stekenjokk, ISK = 1 Aitik lossad, ISK = 4 lägg ut minnesinnehåll på indikerings-tablå, används vid omstart av bäddning och multipelbyte. Vid de två första fallen kontrolleras alltid om operationen är utförbar, d v s att det finns vagnar som kan och skall lossas. Därefter så räknar man ned de återstående och upp lossade vagnar. Efter att ha räknat fram de uppgifter som skall visas på siffertablån vid järnvägsslossningsplatsen anropas programmet BDC som omvandlar ASC II kodade tal till BCD kodade tal, varefter resultatet visas.

Avslutande av bädd, KLAR.

Programmet KLAR aktiveras från RAKNA då det upptäcker att bädden är klar. Rutinens uppgifter är att beordra operatören att skriva in de verkliga värdena på järnvägsvagnarnas vikt eller om detta inte är möjligt, lägga bädden i viloläge. När alla järnvägsvagnar lästs in eller om sådana inte fanns, görs en slututskrift av bädden på snabbskrivaren där man redovisar beordrad vikt, slutvikt och skillnaden mellan dessa i ton för varje material som ingår i bädden. Man gör även en summering och visar totalvikten för bädden, med avseende på bör- respektive ärvikterna. Programmet ser även till att bädden tas bort från bäddregistren så att dessa blir lediga och kan ta emot en ny bädd.

Bakgrundsprogram, STARUP.

Programmet STARUP ligger som bakgrundsprogram och genomlöps så fort ingenting annat utfästes. Programmet känner av alla de tryckknappar som finns, kollar klockan för att se om dygns- eller månadsregistren skall skrivas ut samt startar Vågprogrammet var 200 ms. Vid spänningsbortfall så startas STARUP automatiskt av monitorn vid spänningsåterkomst.

4 DUMMY REENTRANT LIBRARY

Dummy Reentrant Library, i fortsättningen DRL, vad är det för något och vad används det till? Genom att besvara andra frågan först hoppas jag även kunna ge ett svar på den första.

DRL används då man vill spara minnesutrymme. Då man länkar ihop sina huvudprogram till laddningsmoduler så följer det alltid med standard fortran subrutiner. I Realtidsfortran har man emellertid inte bara ett huvudprogram utan kanske flera. Vid ett vanligt länkningsförfarande kommer alltså varje huvudprogram att innehålla ett fortranbibliotek. D v s har man tre huvudprogram så har man också tre fortranbibliotek. Man inser snabbt att detta är slöseri med minne eftersom ett bibliotek hade varit nog. DRL kan då användas för att åstadkomma detta enda bibliotek.

För att ta reda på vilka subrutiner som programmen använder, så länkar man dessa utan att ge dem tillgång till fortranbibliotek. Man får då en felutskrift på de subrutiner som saknas och kan därav bestämma vilka som bör vara med i DRL. När man sedan skapar sitt DRL använder man två program. Det första programmet innehåller bara anrop av de subrutiner som man vill ha med (behöver) i sitt DRL. Då man tagit med alla rutiner som behövs, så länkar man det första programmet. Av den laddkarta man då erhåller så skapar man sitt andra program. Laddningsmodulen man har erhållit sparas också, för att senare kunna tagas ut på remsa. Den är ju det fortranbibliotek man vill ha (behöver). Det andra programmet innehåller namnen på de program som finns i det skapande biblioteket, men man sätter i detta program namnen lika med den startadress vilken man erhållit ur den laddkarta som skrevs ut vid länkningen av det första programmet. För att erhålla absolut adress måste man till adressen för varje rutin addera längden av den monitor man använder då denna ligger före själva biblioteket. Man är nu klar att länka sina program.

Vid länkning förfar man nu så att man inkorporerar först sitt huvudprogram, sedan det program som innehåller startadresserna till subrutinerna i DRL i länkningseditorns arbetsarea. Ingår en common-area i de rutiner som finns i DRL måste man meddela länkningseditorn detta

genom att tala om common-areans startadress (adress ur laddkarta plus monitorns längd). Common-arean skall nämligen ligga före programmen. Länkningseditorn fungerar sedan så att den tittar efter vilka subrutiner huvudprogrammet behöver. Sedan letar den efter dem i sin arbetsarea, först därefter i systembiblioteket. Det andra programmet vi inkorporerade innehåller precis vad länkningseditorn söker, nämligen programnamn plus startadresser, d v s om allt går som det skall kommer vi på länkingslistan endast att hitta de programnamn som inte finns med i biblioteket. När man sedan laddar sitt system så börjar man med monitorn, sedan biblioteket och därefter de huvudprogram som finns i prioritetsordning.

Förklarande exempel på hur DRL skapas och hur en länkning går till visas på efterföljande sidor.

S:ASM FBIBL
LABEL = PROV

DATE = 77-02-16

PACK NER = 000 LASSE

ASM 04

I detta program raknas alla de rutiner
upp som man vill skall ingå i DRL. Ru-
tinerna deklarerar som externa och nämns
i datasatser för att länkningseditorn
skall kunna utföra en korrekt länkning.

00000	IDENT FBIBL	F:IN
00001	EXTRN	F:IO
00002	EXTRN	F:CL
00003	EXTRN	F:L1
00004	EXTRN	F:L2
00005	EXTRN	F:IM
00006	EXTRN	ATTACH
00007	EXTRN	F:D1
00008	EXTRN	F:FL
00009	EXTRN	F:RC
00010	EXTRN	F:T1
00011	EXTRN	F:RP
00012	EXTRN	F:RS
00013	EXTRN	F:RM
00014	EXTRN	F:RD
00015	EXTRN	F:ER
00016	EXTRN	F:ST
00017	EXTRN	F:PA
00018	EXTRN	ACTIV
00019	EXTRN	IADDR
00020	EXTRN	ATTCHW
00021	EXTRN	IOR
00022	EXTRN	CTIM
00023	EXTRN	DATE
00024	EXTRN	DECB
00025	EXTRN	DETACH
00026	EXTRN	DTIM
00027	EXTRN	SIOW
00028	EXTRN	WAIT
00029	EXTRN	UT
00030	EXTRN	ASCII
00031	EXTRN	BLA
00032	EXTRN	BL
00033	EXTRN	NOLL
00034	EXTRN	SP
00035	EXTRN	BCD
00036	EXTRN	STOPP
00037	EXTRN	RA
00038	EXTRN	RED
00039	EXTRN	KONTR
00040	EXTRN	LAMPA
00041	EXTRN	

00042
00043
00044
00045
00046

EXTRN
EXTRN
EXTRN
EXTRN
DATA

RAKNA
SKRIV
F:IX
BTEST
F:IN,F:IO,F:CL,F:L1,F:L2,F:IM,ATTACH

0000	0000	X
0002	0000	X
0004	0000	X
0006	0000	X
0008	0000	X
000A	0000	X
000C	0000	X
000E	0000	X
0010	0000	X
0012	0000	X
0014	0000	X
0016	0000	X
0018	0000	X
001A	0000	X
001C	0000	X
001E	0000	X
0020	0000	X
0022	0000	X
0024	0000	X
0026	0000	X
0028	0000	X
002A	0000	X
002C	0000	X
002E	0000	X
0030	0000	X
0032	0000	X
0034	0000	X
0036	0000	X
0038	0000	X
003A	0000	X
003C	0000	X
003E	0000	X
0040	0000	X
0042	0000	X
0044	0000	X
0046	0000	X
0048	0000	X
004A	0000	X
004C	0000	X
004E	0000	X
0050	0000	X
0052	0000	X
0054	0000	X
0056	0000	X

00047

DATA

F:D1,F:FL,F:RC,F:T1,F:RP,F:RS,F:RM,F:RD,F:ER

00048

DATA

F:ST,F:PA,ACTIV,IADDR,IOR,BTEST,F:IX

00049

DATA

CTIM,DATE,DETACH,DTIM,SIOW,WAIT,UT

00050

DATA

ASCII,BLA,BL,NOLL,SP,BCD,STOPP,RA,RED,KONTR,LAMPA

00051

DATA

RAKNA,SKRIV,DECB

00052

END

SYMBOL TABLE

F:IN	X	F:IO	X	F:CL	X	F:L1	X
F:L2	X	F:IM	X	ATTACH	X	F:D1	X
F:FL	X	F:RC	X	F:T1	X	F:RP	X
F:RS	X	F:RM	X	F:RD	X	F:ER	X
F:ST	X	F:PA	X	ACTIV	X	IADDR	X
ATTCHW	X	IOR	X	CTIM	X	DATE	X
DECB	X	DETACH	X	DTIM	X	SIOW	X
WAIT	X	UT	X	ASCII	X	BLA	X
BL	X	NOLL	X	SP	X	BCD	X
STOPP	X	RA	X	RED	X	KONTR	X
LAMPA	X	RAKNA	X	SKRIV	X	F:IX	X
BTEST	X						

ASS.ERR. 00000

:EOF
PROG ELAPSED TIME: 00H-00M-00S-000MS-

S:

Länkning av det program där vi räknade upp alla de rutiner som vi ville ha med i DRL. Om man jämför listan till vänster med de program vi räknat upp så finner man flera program som har tillkommit eller försvunnit. Detta beror på att de program som vi har valt kan vara en del av ett block. Vid länkningen kommer emellertid hela blocket med och de enskilda programmen ersätts av blocknamnet. De enskilda programmen redovisas emellertid i laddkartan (symbol table). Ett exempel på detta är rutinerna F:RP och F:RS som har ersatts av F:RPRS.

OBS! Alla talvärden som finns i dessa tabeller är hexadecimala

F:BIBL	0008
F:IN	0060
F:IO	0448
F:EXCL	0636
F:VAR	0916
F:LW	099C
F:AW	09DA
F:INUM	0A14
F:OUTN	0E8D
F:COP	0D1C
F:ICI	0D3C
F:FLOC	0D64
F:FLIC	0F50
F:BIN	1026
F:IORB	10F2
F:IOWD	114D
F:IOCH	1162
F:IORC	11C2
F:IA	127C
F:DI	1334
F:DD	142C
F:DM	15DE
F:DP	1758
F:DN	1838
F:T3	1896
F:FL	18C2
F:RC	18D4
F:TLN	1988
F:TRA	19EC
F:FX	1A28
F:RNSA	1AC6
F:RPRS	1B3E
F:RMRD	1CB0
F:MS	1EAE
ACTIV	1F80
ADDR	1FA0
ATTACH	203C
BITOP	2056
CTIM	21D2
DATE	2154
DECB	2194
DETACH	21C4
DTIM	21D8
SIO	21FD
WAIT	2212

Har ersatt F:RP och F:RS

UT 225A
 ASCII 2296
 BLA 232D
 BL 253E
 NOLL 28E4
 SP 29FE
 BCD 2B14
 STOPP 2B3E
 RA 2C5A
 RED 2F64
 KONTR 3176
 LAMPA 32DC
 SKRIV 3452
 UT2 506C
 RAKNA 5088
 NO STRT.

Nedan följer den symboltabell som jag i texten har kallat laddkarta. Ur denna laddkarta hämtar vi startadresserna till common-arean och de rutiner som ingår i DRL.

*** SYMBOL TABLE ***

common-area

ATTCHW	2040 R	5B30 C	ATTACH	203C R
BLA	24E0 R	BCLR	BL	282C R
DATE	2154 R	BTEST	CTIM	2102 R
F:AB	1B36 R	DETACH	DTIM	21D8 R
F:CFW	110A R	F:BIN	F: BIRD	1106 R
F:D1	1312 R	F:CLOS	F:COP	0D1C R
F:DN	1838 R	F:DI	F:DM	15DE R
F:ER77	11BC R	F:DP	F:ER	1EFO R
F:FLOC	0D64 R	F:EXIT	F:FLIC	0F50 R
F:IM	127C R	F:FX	F:ID	128E R
F:IOAS	1182 R	F:IN	F:IO	0448 R
F:IOND	0986 R	F:IOB	F:IOIM	1156 R
F:IORQ	11F8 R	F:IOP	F:IORC	11C2 R
F:L1	071A R	F:LOWD	F:IX	12A4 R
F:PA	1F54 R	F:L2	F:OUTN	0BB0 R
F:RM	1CB6 R	F:RC	F:RLWS	0980 R
F:S2	12E2 R	F:RN	F:RS	1B56 R
F:ST	1F24 R	F:S4	F:SI	1B12 R
F:TEN	0E0C R	F:STAT	F:T3	1896 R
F:VAR	0916 R	F:TLN	F:TYFC	0005 A
IBYTE	1FA4 R	IADDR	IBTEST	20BA R
IWORD	1FC4 R	IEOR	ISHFT	207A R
LCMP	201E R	KONTR	LAMPA	3416 R
RA	2F06 R	LSUB	NOT	208A R
SIO	21F0 R	RAKNA	RVALUE	1FE4 R
STBYTE	1FA8 R	SIOW	SP	2AD4 R
UT	225A R	STOPP	STWORD	1FC8 R
		UT2		
		WAIT		
		2212 R		

Här redovisas åter F:RP och F:RS som enskilda rutiner.

I detta program räknas alla rutiner upp som ingår i DRL och till den adress som man fick ur laddkartan läggs monitorns längd.

IDENT DRL

00000
00001
00002
00003
00004
00005
00006
00007
00008
00009
00010
00011
00012
00013
00014
00015
00016
00017
00018
00019
00020
00021
00022
00023
00024
00025
00026
00027
00028
00029
00030
00031
00032
00033
00034
00035
00036
00037
00038
00039
00040
00041
00042
00043
00044
00045

IBYTE
CTIM
F:IM
F:L2
F:RP
F:ST
SIOW
ACTIV
DATE
DTIM
F:IN
F:RC
F:RS
F:T1
WAIT
ATTCHW
BSET
DECB
F:CL
F:IO
F:RD
F:S2
IADDR
ISHFT
BCLR
BIEST
DETACH
F:FL
F:L1
F:RM
F:S4
IAND
RA
RED
LAMP
SP
NOLL
BL
BLA
ASCII
RAKNA
BCD
KONTR
UT
SKRIV

00046	ENTRY	STOPP
00047	ENTRY	F:S6
00048	ENTRY	F:PA
00049	ENTRY	F:ER
00050	ENTRY	F:ID
00051	ENTRY	F:IX
00052	ENTRY	F:D1
00053	ENTRY	W:DAY
00054	ENTRY	W:MONTH
00055	ENTRY	W:YEAR
00056	ENTRY	H:TIME
00057	ENTRY	M:TIME
00058	ENTRY	S:TIME
00059	EQV	/O30A
00060	EQV	/O30C
00061	EQV	/O30E
00062	EQV	/O310
00063	EQV	/O312
00064	EQV	/O314
00065	EQV	/ZADD
00066	EQV	DRL+/0060
00067	EQV	DRL+/0448
00068	EQV	DRL+/0712
00069	EQV	DRL+/071A
00070	EQV	DRL+/0722
00071	EQV	DRL+/127C
00072	EQV	DRL+/128E
00073	EQV	DRL+/12A4
00074	EQV	DRL+/12E2
00075	EQV	DRL+/12E6
00076	EQV	DRL+/12EA
00077	EQV	DRL+/1312
00078	EQV	DRL+/18C2
00079	EQV	DRL+/18FE
00080	EQV	DRL+/1A2C
00081	EQV	DRL+/1B3E
00082	EQV	DRL+/1BE6
00083	EQV	DRL+/1CB6
00084	EQV	DRL+/1DE6
00085	EQV	DRL+/1EFO
00086	EQV	DRL+/1F24
00087	EQV	DRL+/1F54
00088	EQV	DRL+/1F80
00089	EQV	DRL+/1FA0
00090	EQV	DRL+/1FA4
00091	EQV	DRL+/2040
00092	EQV	DRL+/2080
00093	EQV	DRL+/209A

f

Adresser till register inne i monitorn
som används då man vill ställa realtids-
klockan.
Monitorns längd

W:DAY
W:MONTH
W:YEAR
H:TIME
M:TIME
S:TIME
DRL
F:IN
F:IO
F:CL
F:L1
F:L2
F:IM
F:ID
F:IX
F:S2
F:S4
F:S6
F:D1
F:FL
F:RC
F:T1
F:RP
F:RS
F:RM
F:RD
F:ER
F:ST
F:PA
ACTIV
IADDR
IBYTE
ATTCHW
IAND
ISHFT

```

00094 EQU DRL+/208A
00095 EQU DRL+/20D8
00096 EQU DRL+/20DC
00097 EQU DRL+/2102
00098 EQU DRL+/2154
00099 EQU DRL+/2194
00100 EQU DRL+/21C4
00101 EQU DRL+/21D8
00102 EQU DRL+/21F4
00103 EQU DRL+/2212
00104 EQU DRL+/225A
00105 EQU DRL+/229A
00106 EQU DRL+/24E0
00107 EQU DRL+/282C
00108 EQU DRL+/29AC
00109 EQU DRL+/2AD4
00110 EQU DRL+/2B16
00111 EQU DRL+/2C18
00112 EQU DRL+/2FD6
00113 EQU DRL+/311A
00114 EQU DRL+/3298
00115 EQU DRL+/3416
00116 EQU DRL+/5AC4
00117 EQU DRL+/4E50
00118 END

```

Addition av 2A00 och 229A hexadecimalt = 4C9A

SYMBOL TABLE

IBYTE	49A4	A	CTIM	4802	A	F:IM	3C7C	A	F:L2	3122	A
F:RP	453E	A	F:ST	4924	A	SIOW	4BF4	A	ACTIV	4980	A
DATE	4854	A	DTIM	48D8	A	F:IN	2A6D	A	F:RC	42FE	A
F:RS	45E6	A	F:T1	442C	A	WAIT	4C12	A	ATTCHM	4A4D	A
BSET	4AD8	A	DECB	4894	A	F:CL	3112	A	F:IO	2E48	A
F:RD	47E6	A	F:S2	3CE2	A	IADDR	49AD	A	ISHFT	4A9A	A
BCLR	4ADC	A	BTEST	4ABA	A	DETACH	4BC4	A	F:FL	42C2	A
F:L1	311A	A	F:RM	46B6	A	F:S4	3CE6	A	IAND	4A8D	A
RA	5906	A	RED	5B1A	A	LAMPA	5E16	A	SP	54D4	A
NOLL	53AC	A	BL	522C	A	BLA	4EED	A	ASCII	4C9A	A
RAKNA	84C4	A	BCD	5516	A	KONTR	5C98	A	VT	4C5A	A
SKRIV	7850	A	STOPP	5618	A	F:S6	3CEA	A	F:PA	4954	A
F:ER	48FD	A	F:ID	3C8E	A	F:IX	3CA4	A	F:D1	3D12	A
W:DAY	030A	A	W:MONT	030C	A	W:YEAR	030E	A	H:TIME	0310	A
M:TIME	0312	A	S:TIME	0314	A	DRL	2ADD	A			

Jfr med resultat ovan

ASS.ERR. 00000

:EOF
PROG ELAPSED TIME: 00H-00M-00S-000MS-

LÄNKNING AV RUTINEN JVL

S:SCR /0 Nollställning av arbetsarean
 S:INC JVL Inkoppling av huvudprogram
 S:INC DRL Inkoppling av program som innehåller startadresserna till rutinerna i DRL
 S:LKE U,M,/8531 Startadress för commonarean hexadecimalt
 LABEL = PROV DATE = 77-02-16 PACK NBR = 000 LASSE

JVL 0008
 DRL 02C8
 UT2 02C8
 Detta program ingick inte DRL

*** SYMBOL TABLE ***

common-area startadress

BCD	5516 A	ACTIV	4980 A	ASCII	4C9A A	ATTCHW	4A40 A
BSET	4AD8 A	BCLR	4ADC A	BL	522C A	BLA	4EEO A
DECB	4B94 A	BTEST	4ABA A	CTIM	4B02 A	DATE	4B54 A
F:D1	3D12 A	DETACH	4BC4 A	DTIM	4BD8 A	F:CL	3112 A
F:IM	3C7C A	F:ER	48FD A	F:FL	42C2 A	F:ID	3C8E A
F:L1	311A A	F:IN	2A60 A	F:IO	2E48 A	F:IX	3CA4 A
F:RD	47E6 A	F:L2	3122 A	F:PA	4954 A	F:RC	42FE A
F:S2	3CE2 A	F:RM	46B6 A	F:RP	453E A	F:RS	45E6 A
F:T1	442C A	F:S4	3CE6 A	F:S6	3CEA A	F:ST	4924 A
IBYTE	49A4 A	H:TIME	0310 A	IADDR	49A0 A	IAND	4A80 A
M:TIME	0312 A	ISHFT	4A9A A	KONTR	5C98 A	LAMPA	5E16 A
RED	5B1A A	NOLL	53AC A	RA	5906 A	RAKNA	84C4 A
SP	54D4 A	S:TIME	0314 A	SIOW	4BF4 A	SKRIV	7850 A
W:DAY	030A A	STOPP	5618 A	UT	4C5A A	UT2	02C8 R
		W:MONT	030C A	W:YEAR	030E A	WAIT	4C12 A

START = 0264 LENGTH = 02E4 REGION = 0187

:EOF
 PROG ELAPSED TIME: 00H-00M-00S-000MS-

S:KPF /L,JVL
 LABEL = PROV
 DATE = 77-02-16
 PACK NBR = 000
 LASSE

5 COMMON-AREA

I detta avsnitt skall vi se hur common-arean är disponerad och hur den används. Common-arean har följande utseende.

```
COMMON IBDNR(2), IA(2), MT(2), LP(2), GR(2), IBD(2, 20, 15)
x BD(2, 20, 5), NR, IS, ISK, BOR, KP(2), BUF(4), BUF1(4), IBDN
x MAN, TARA, VL, SKL, MANR, MANR1, IDEV1, IDEV2, IDEV3
x IDEV4, IDEV5, IDEV6, MREG(30, 3), REG(30, 2)
```

Registren IBDNR(2), IA(2), MT(2), LP(2) och GR(2) innehåller ett ord för var och ett av de två bäddregisterna i datorn. I vektorerna IBD(2, 20, 15) och BD(2, 20, 5) anger första index vilket bäddregister som adresseras. Indexnummer två anger vilken post man vill åt bland materialen och det tredje talar om vilket ord man arbetar med. Alla ovan angivna register används endast för att hålla reda på en bädds status och den information som operatören gett angående bäddens sammanställning.

Följande tabell innehåller en förklaring till hur de tidigare nämnda registren används, samt en beskrivning av det ord som index nummer tre i vektorerna IBD och BD anger.

Tabell 1 - disposition av register och vektorerna IBD och BD.

REGISTER	VEKTOR	ORDNR	ANVÄNDNING, INNEHÅLL
	IBDNR	-	Bäddnummer.
	IA	-	Totala antalet multiplar för en bädd.
	MT	-	Antal multiplar som avverkats.
	LP	-	Numret på den materialpost som bearbetas.
	GR	-	Grundsatsens storlek i ton.
	IBD	(.,.,1)	Postnummer två siffror.
	"	(.,.,2)	Ändringsmarkering och asterisk vid ändring av materialkod.
	"	(.,.,3)	Sort, två siffror.
	"	(.,.,4)	Sort-bokstav och asterisk, dock endast vid ändring av sort.
	"	(.,.,5)	Asterisk, dock endast vid börvikts ändring.

REGISTER	VEKTOR	ORDNR	ANVÄNDNING, INNEHÅLL
	IBD	(.,.,6)	Börvärdet för totala antalet järnvägsvagnar.
	"	(.,.,7)	Totala antalet återstående järnvägsvagnar.
	"	(.,.,8)	Totala antalet beordrade järnvägsvagnsclass fram till pågående multipel.
	"	(.,.,9)	Totala antalet lossade järnvägsvagnsclass.
	"	(.,.,10)	Här noteras om post blivit avbruten genom
	"	(.,.,11)	att lägga in AVBR.
	"	(.,.,12)	Reserv.
	"	(.,.,13)	Adress till dygns- och månadsregister.
	"	(.,.,14)	Markering om kvarts-skopa behövs.
	"	(.,.,15)	Talar om hur en materialpost skall behandlas. 0 = lossning tillåten, 1 = hoppa över, 2 = sätt ordet till ett vid nästa multipel- byte, 3 = hoppa över posten denna multipel, sätt ordet till noll vid multipelbyte.
	BD	(.,.,1)	Börvikt, hela bädden.
	"	(.,.,2)	Återstår, hela bädden.
	"	(.,.,3)	Börvikt fram till och med denna multipel.
	"	(.,.,4)	Ärvikt fram till och med denna multipel.
	"	(.,.,5)	Reserv.

Registren NR, IS, ISK, BOR, KP(2), BUF(4), BUF1(4), IBDN och MAN används endast för att transportera information mellan olika rutiner. Registret NR är det enda som har en specifik användning, nämligen att hålla reda på vilket bäddregister som är i drift, ett eller två. Noll uppfattas som att ingen bäddning pågår.

Storheterna TARA, VL och SKL är variabler och används i enlighet med tabell 2.

Tabell 2 - användning av TARA, VL och SKL.

VARIABLEL ANVÄNDNING

TARA	Tareringsgränd för väg.
VL	Vikten av ett järnvägsglass.
SKL	Vikten av en skoplast från trucken.

Registren MANR och MANR1 används för att lagra status på datorns utgångar, dock ej för bladverken. I MANR finns alla signalerna till lamporna på indikeringstablåerna, signalerna som väljer bäddplats samt start - stopp av vågbandet. Signalerna är lagrade som ettor eller nollor där etta är aktiv. Varje enhet har sin egen specifika bit, t ex motsvaras grön lampa av bit nummer 6 o s v. MANR1 innehåller BCD-koden för sju-segments displayen vid järnvägslossningsplatsen samt signalen för frysning av vågutslaget.

För alla in- resp utgångar till datorn gäller att dessa går över speciella kort som har sexton anslutningar, d v s ett ord.

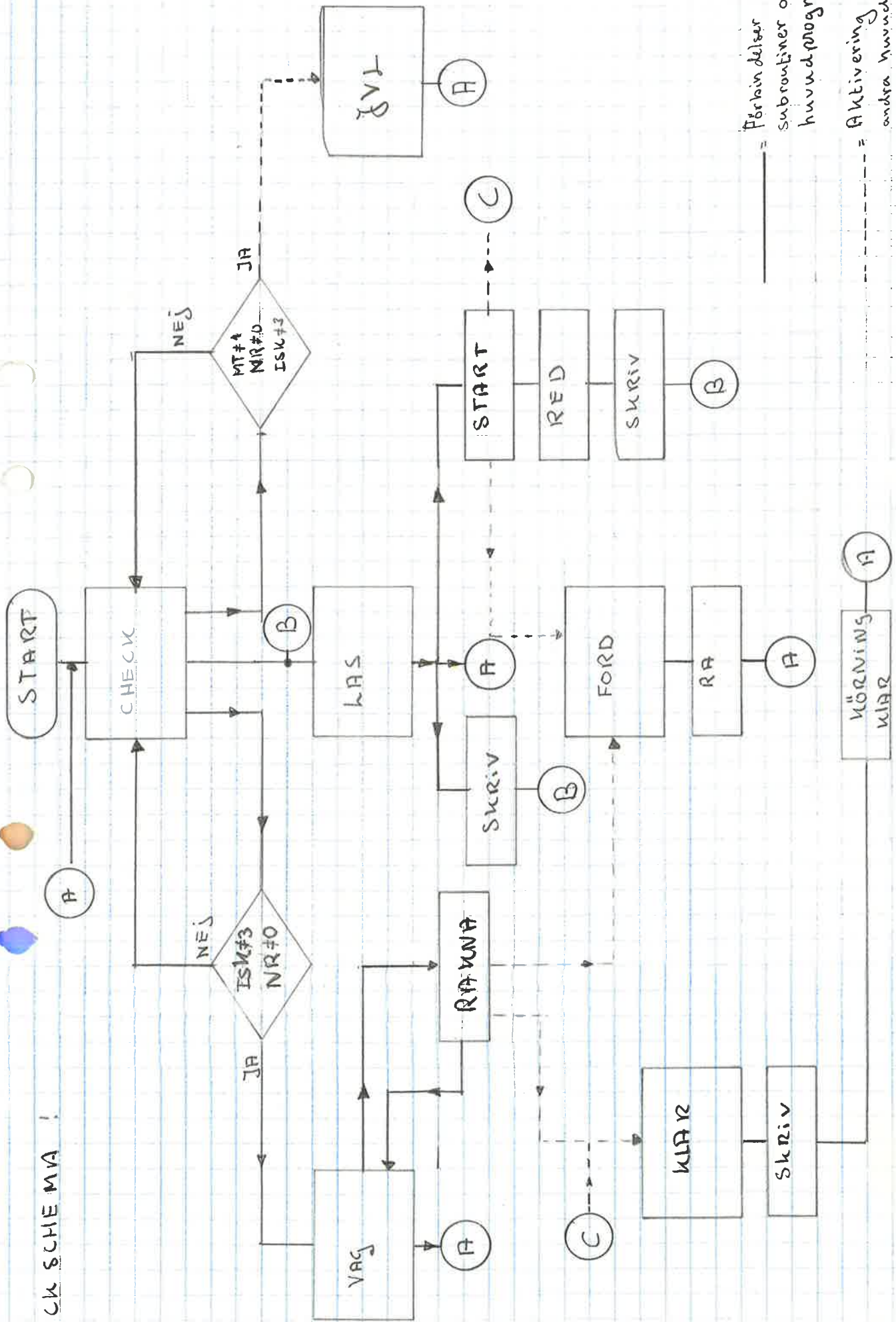
Registren IDEV1 - IDEV6 innehåller adresserna till de yttre enheter som finns anslutna till systemet. För närvarande används IDEV1 - IDEV4. IDEV5 - IDEV6 är reserver.

Vektorerna MREG(30, 3) och REG(30, 2) används för att lagra dygns- resp månadsförbrukningen för de material som används. Första index anger vilken av de trettio material som listan kan innehålla man vill åt. Användningen av index nummer två förklaras i tabellen nedan.

Tabell 3 - disposition av MREG och REG.

REGISTER	VEKTOR	ORDNR	ANVÄNDNING, INNEHÅLL
	MREG	(.,1)	Materialkod siffror.
	"	(.,2)	Materialkod bokstav.
	"	(.,3)	Reserv.
	REG	(.,1)	Dygnsförbrukning.
	"	(.,2)	Månadsförbrukning.

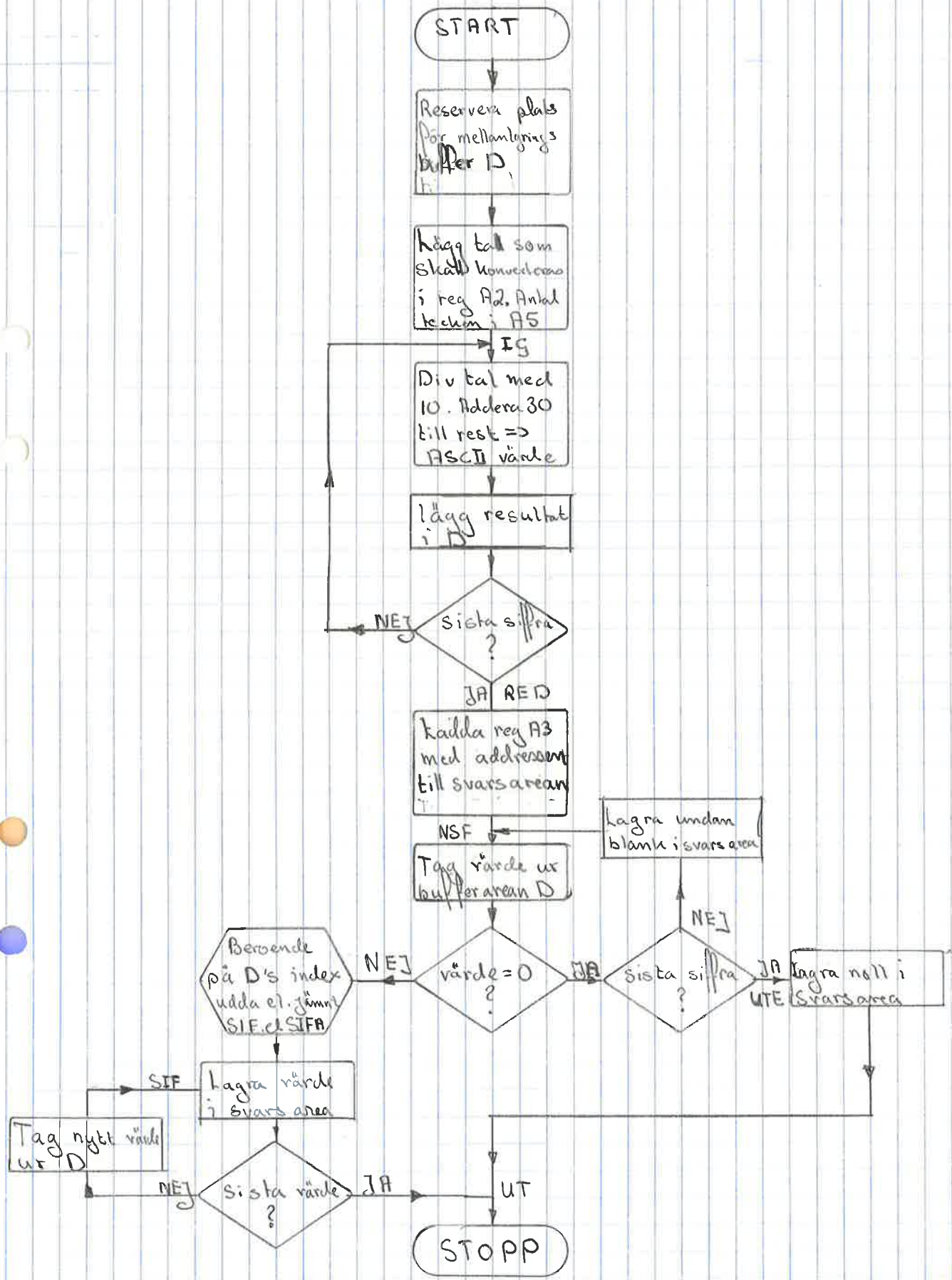
BLOCK SCHEMA !



_____ = Förbindelser mellan subrutiner och huvudprogram

- - - - - = Aktivering av andra huvudprogram

Flödesschema för rutinen ASCII



START

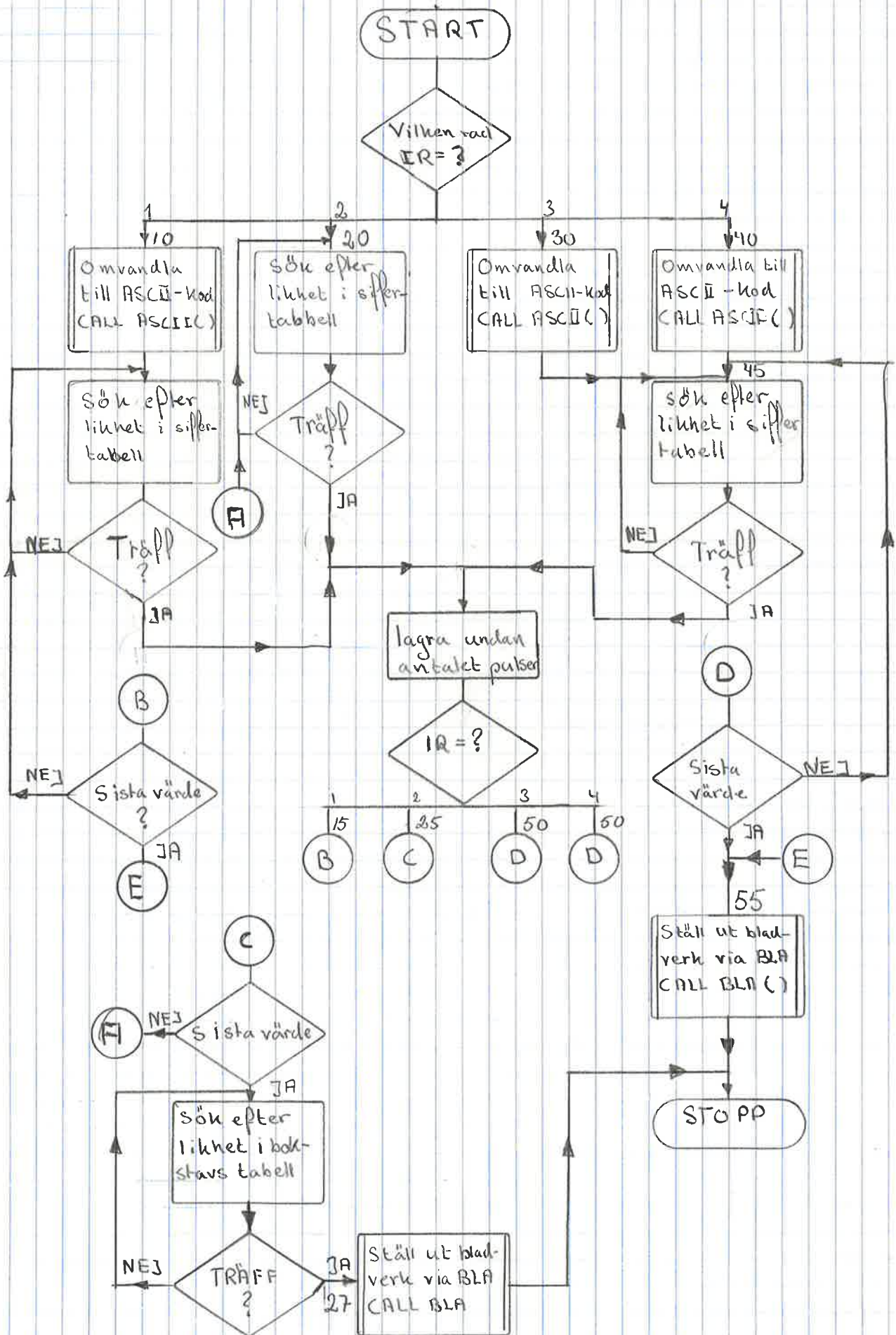
Ladda siffror
som skall om-
vandlas i reg
A1, A2 och A3

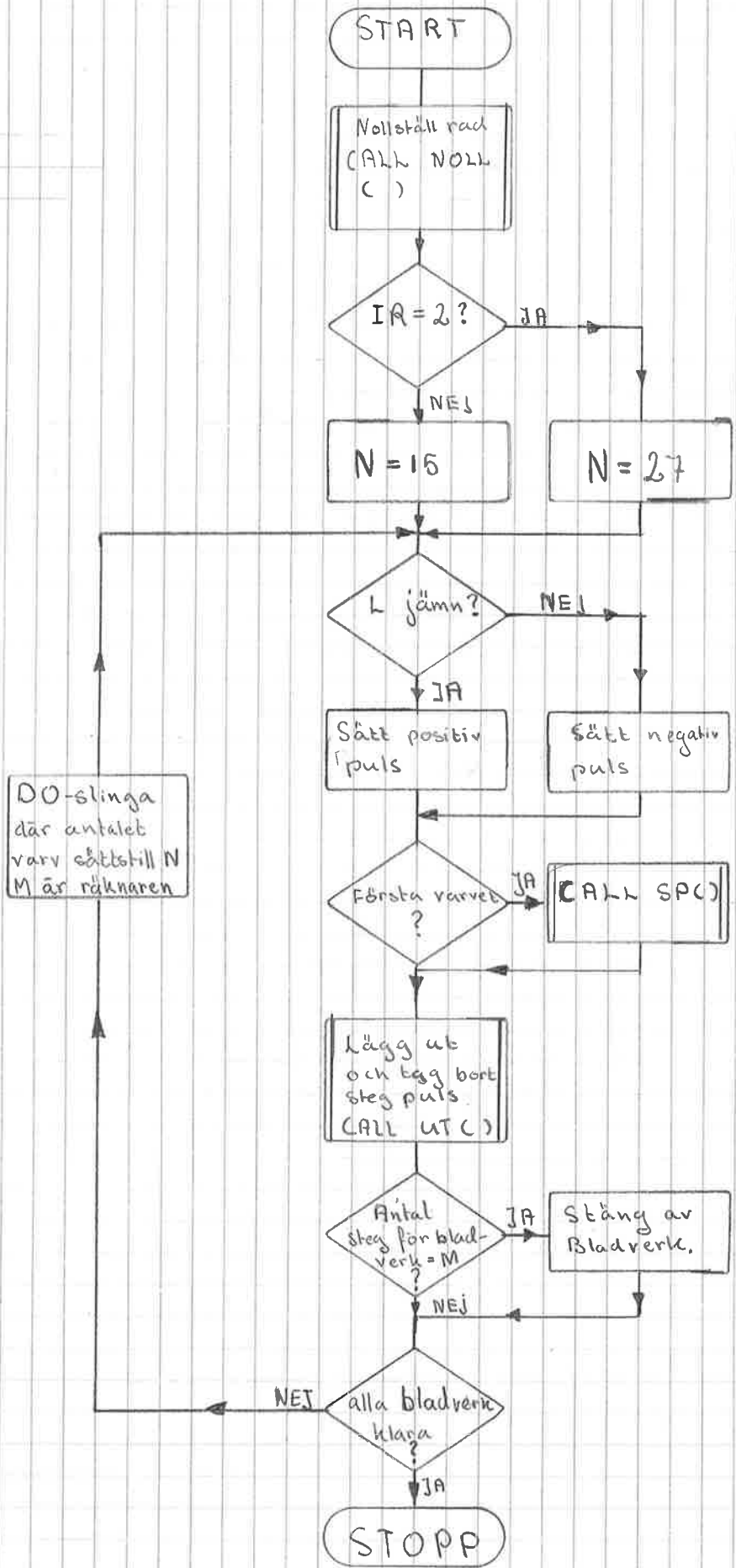
Ladda MAWR1
med addom till svar

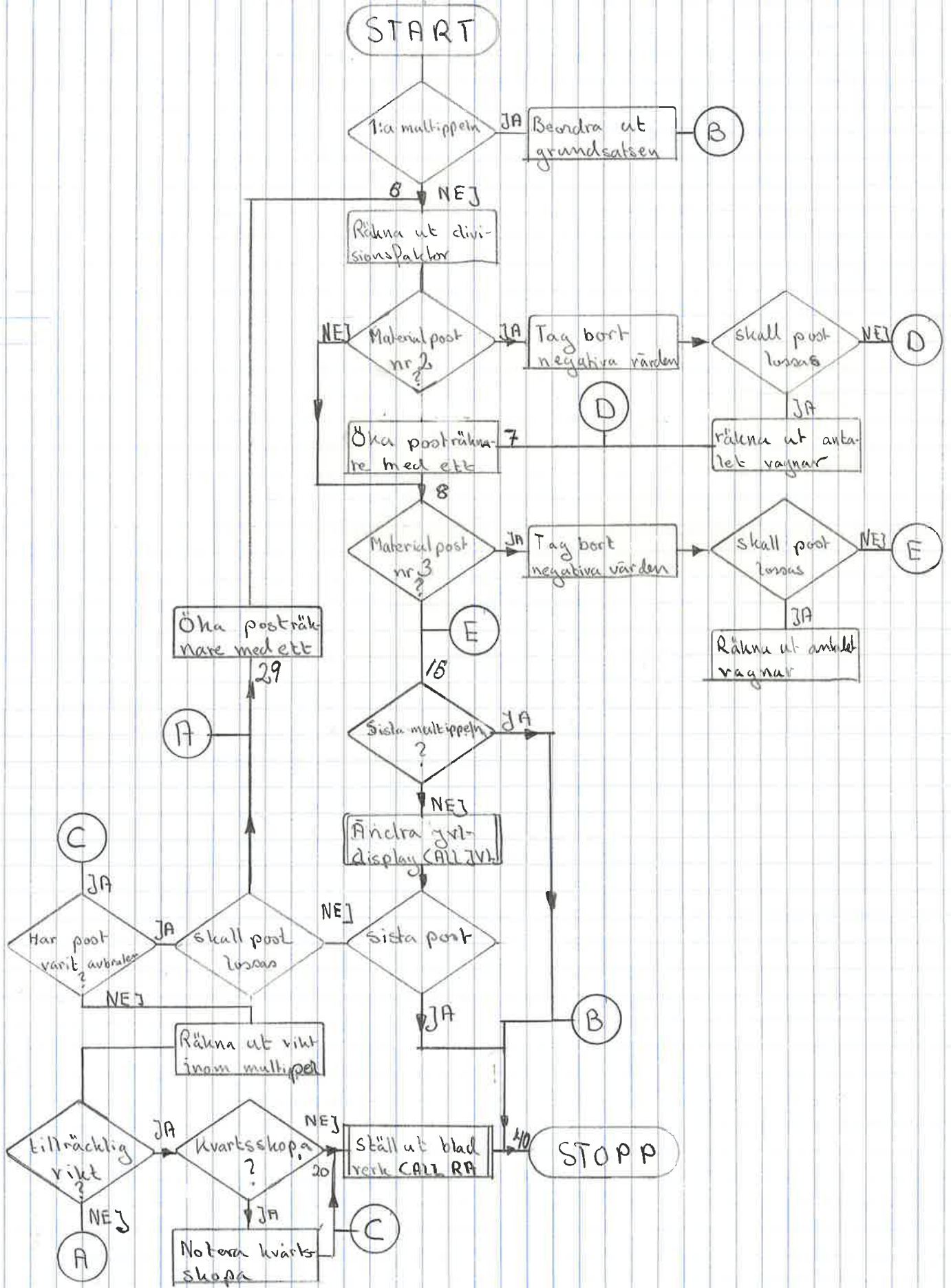
Ut för lämp-
ligt antal skift
så att reg A1 =
BCD kod

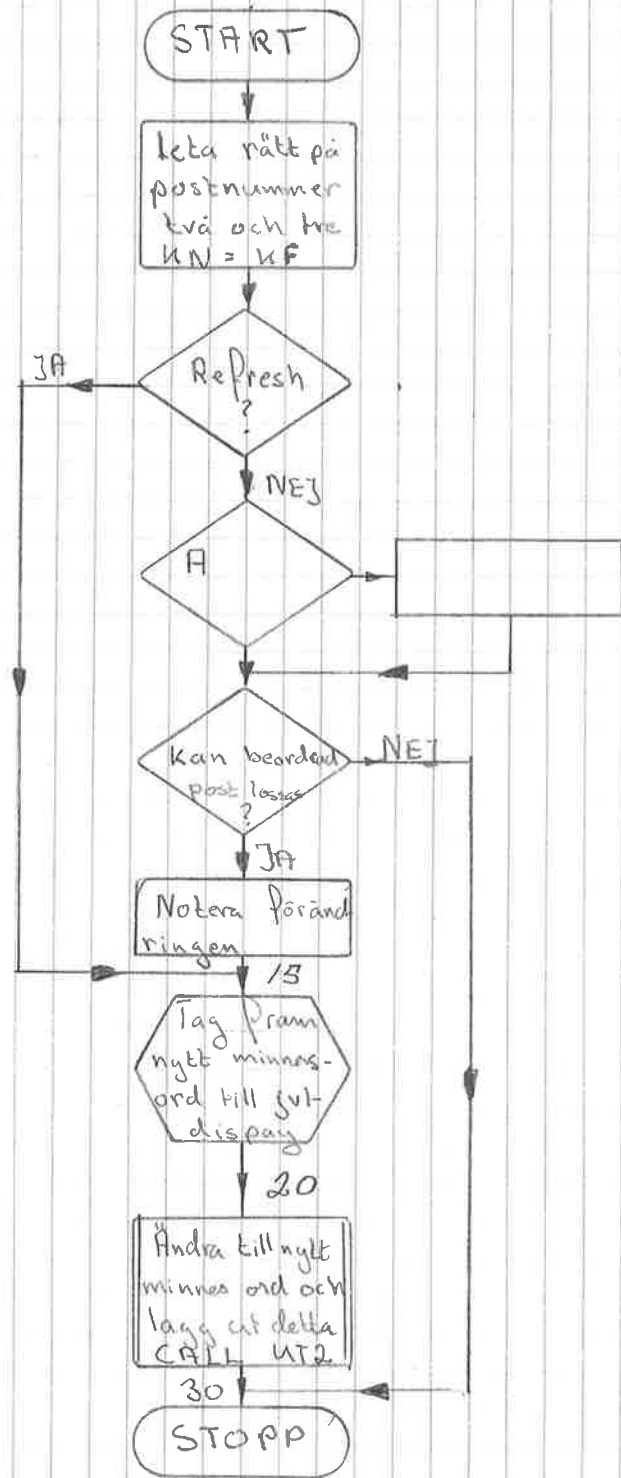
Lägg A1's
innehåll i svararen

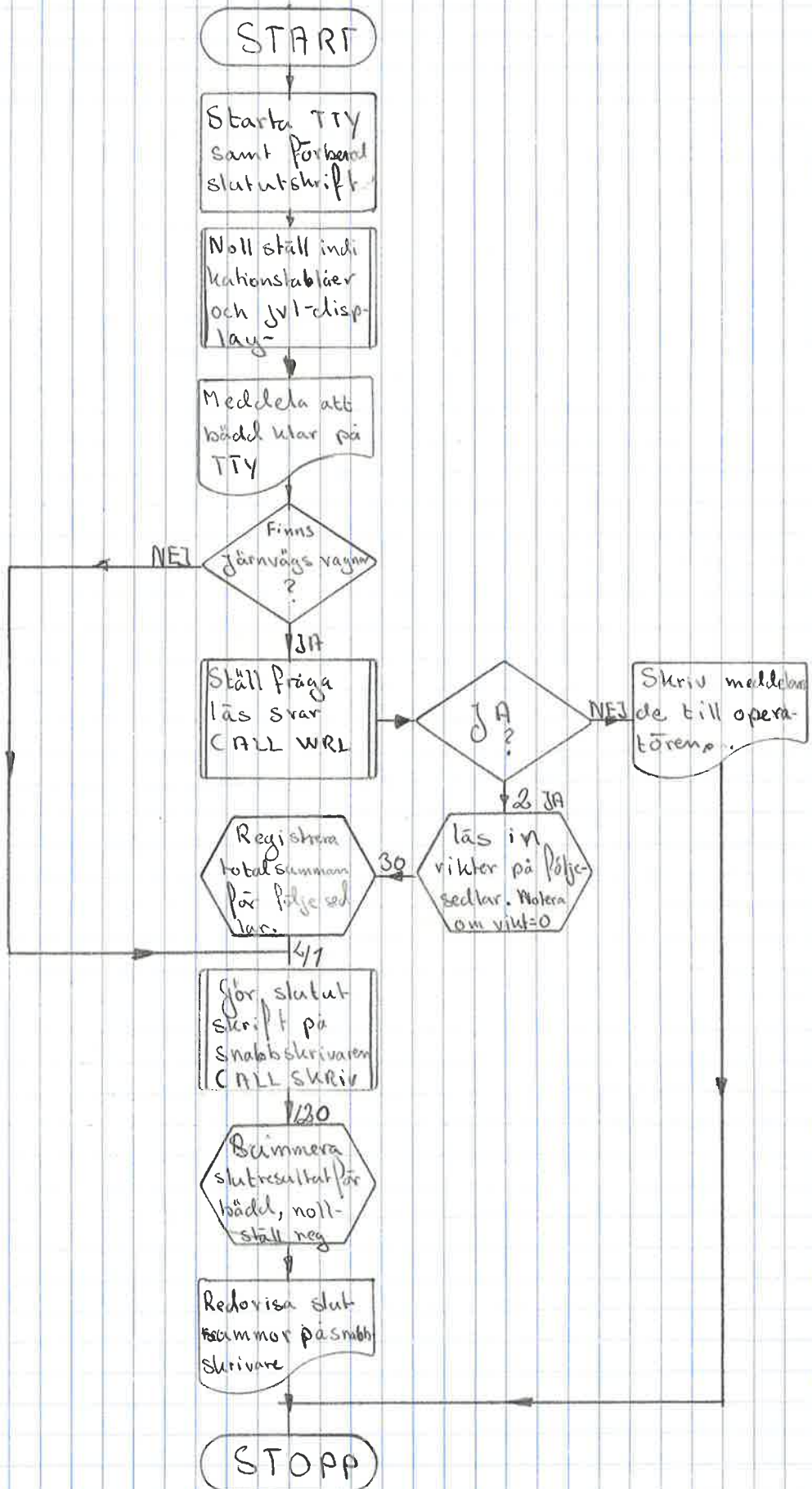
STOPP



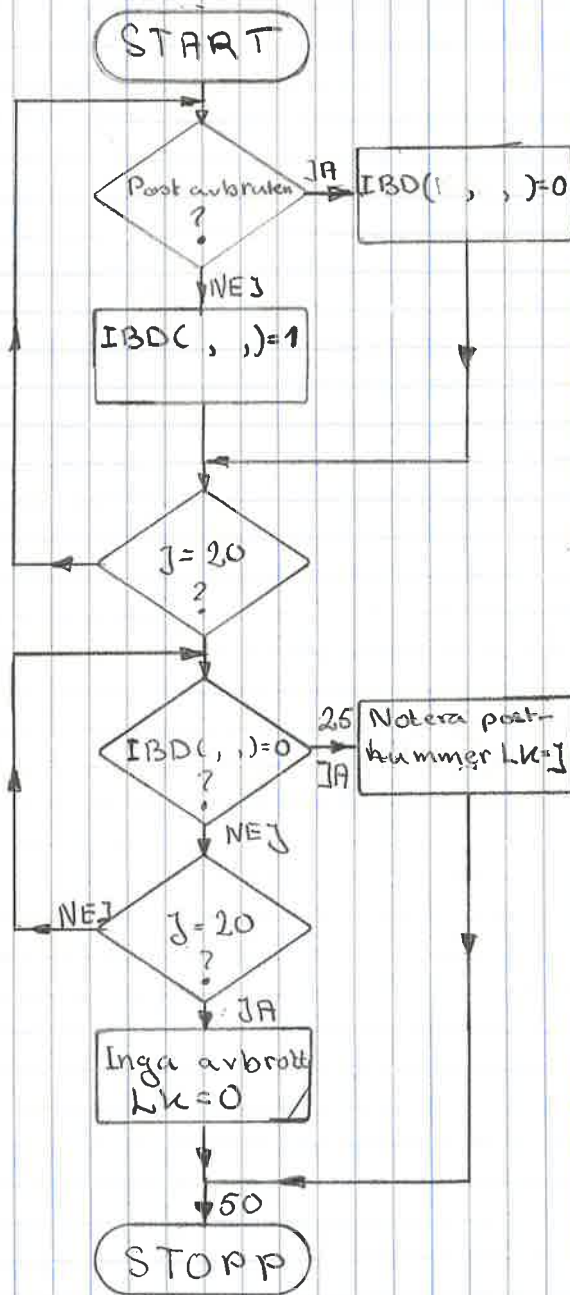




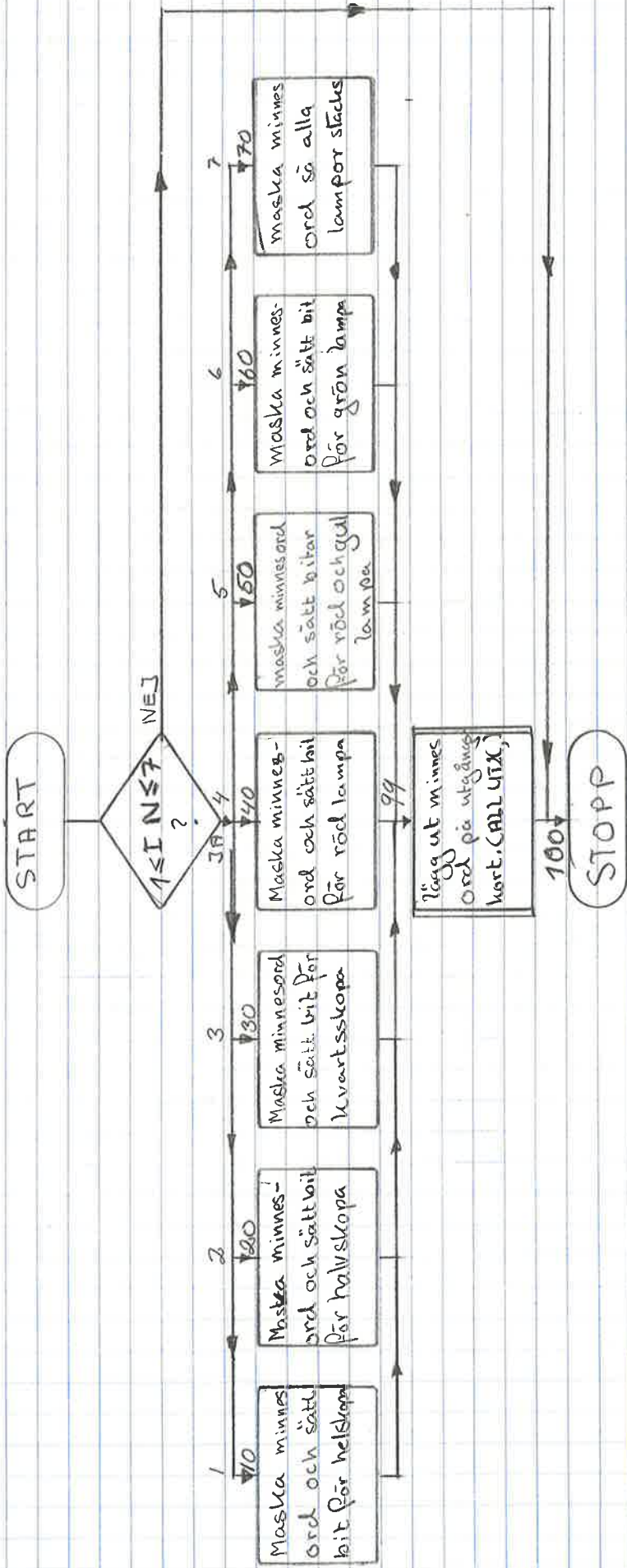


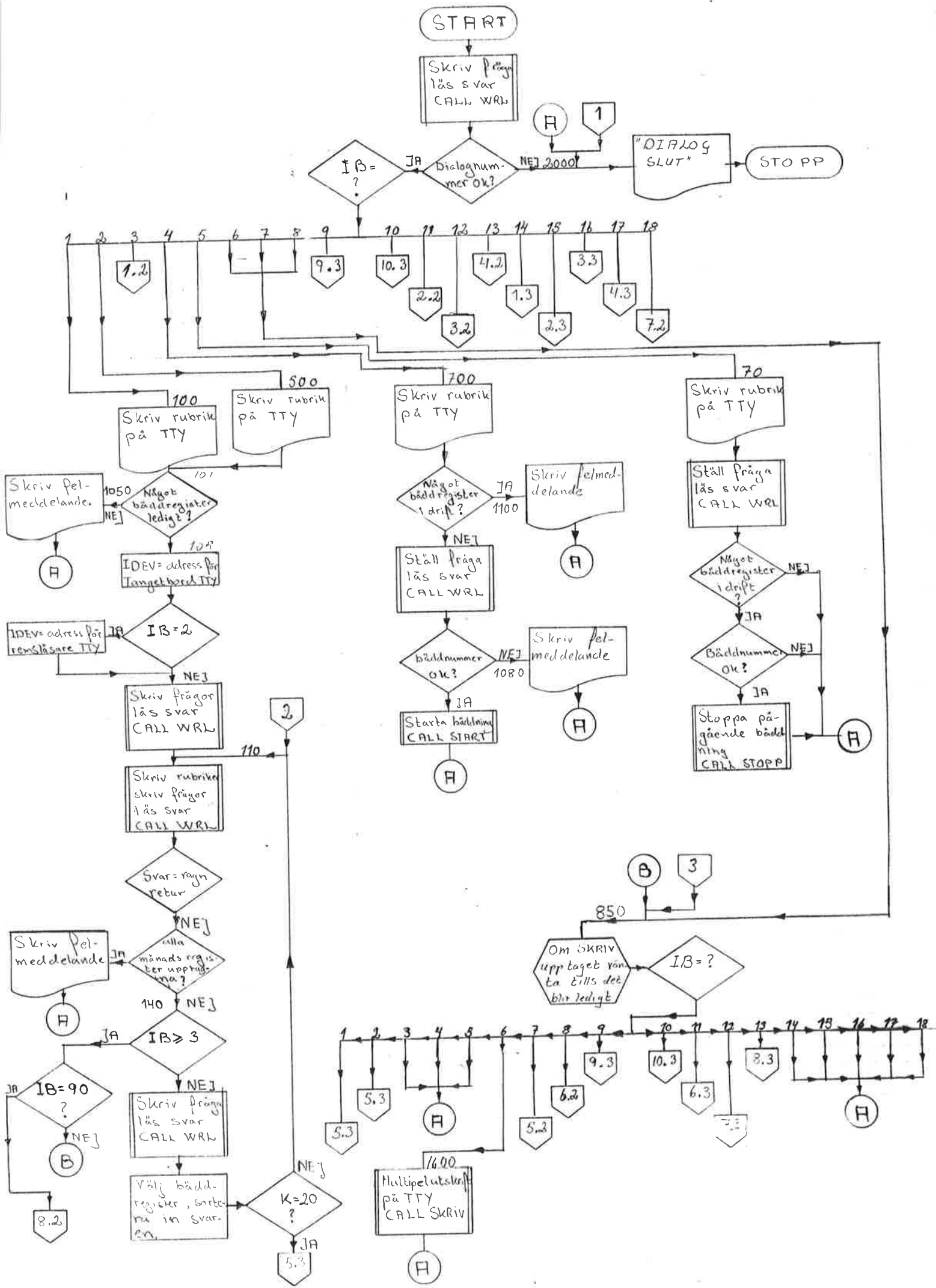


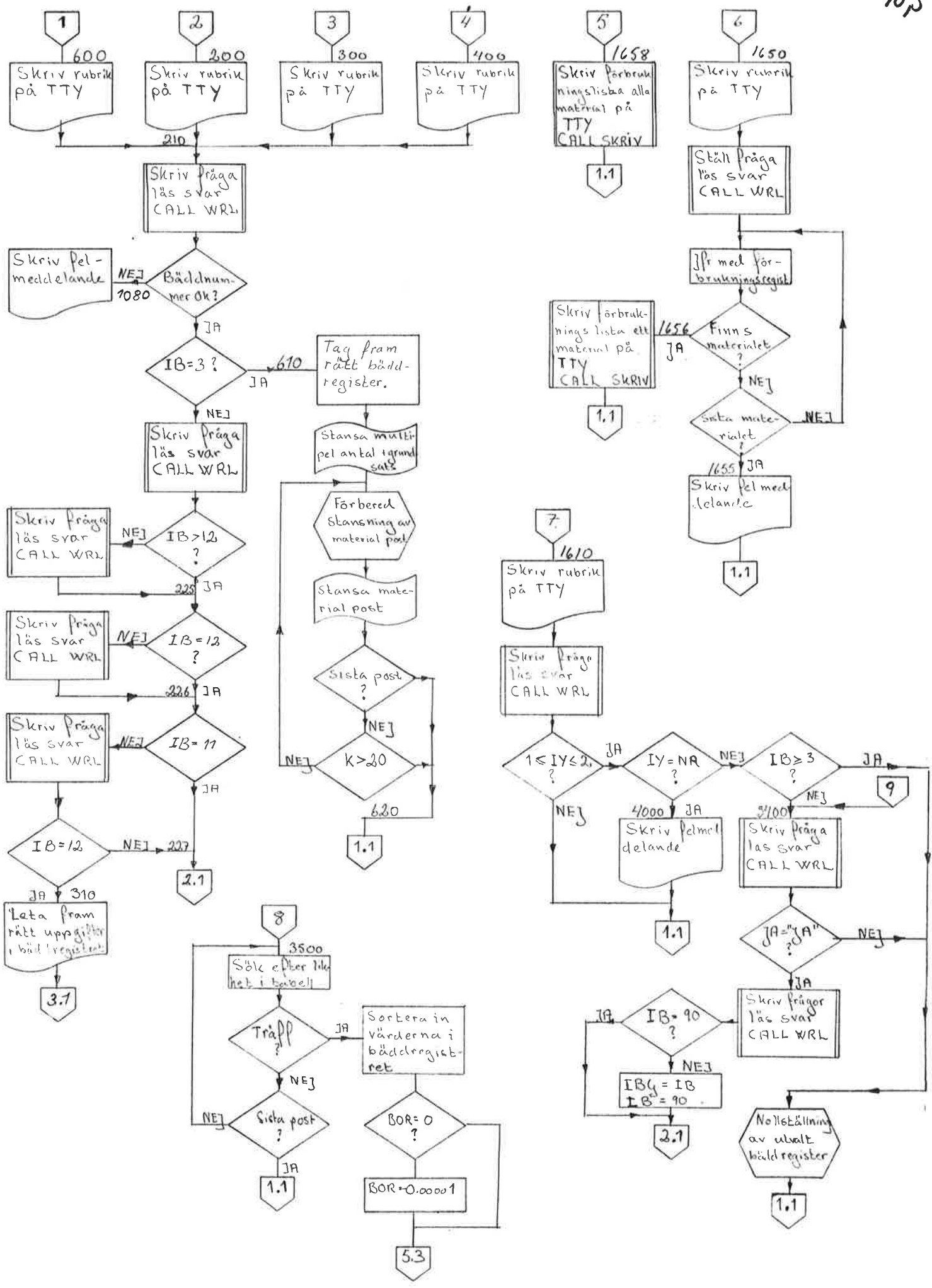
Flödesschema för rutinen KONTR.



Flödeschema för rutinen LAMPA







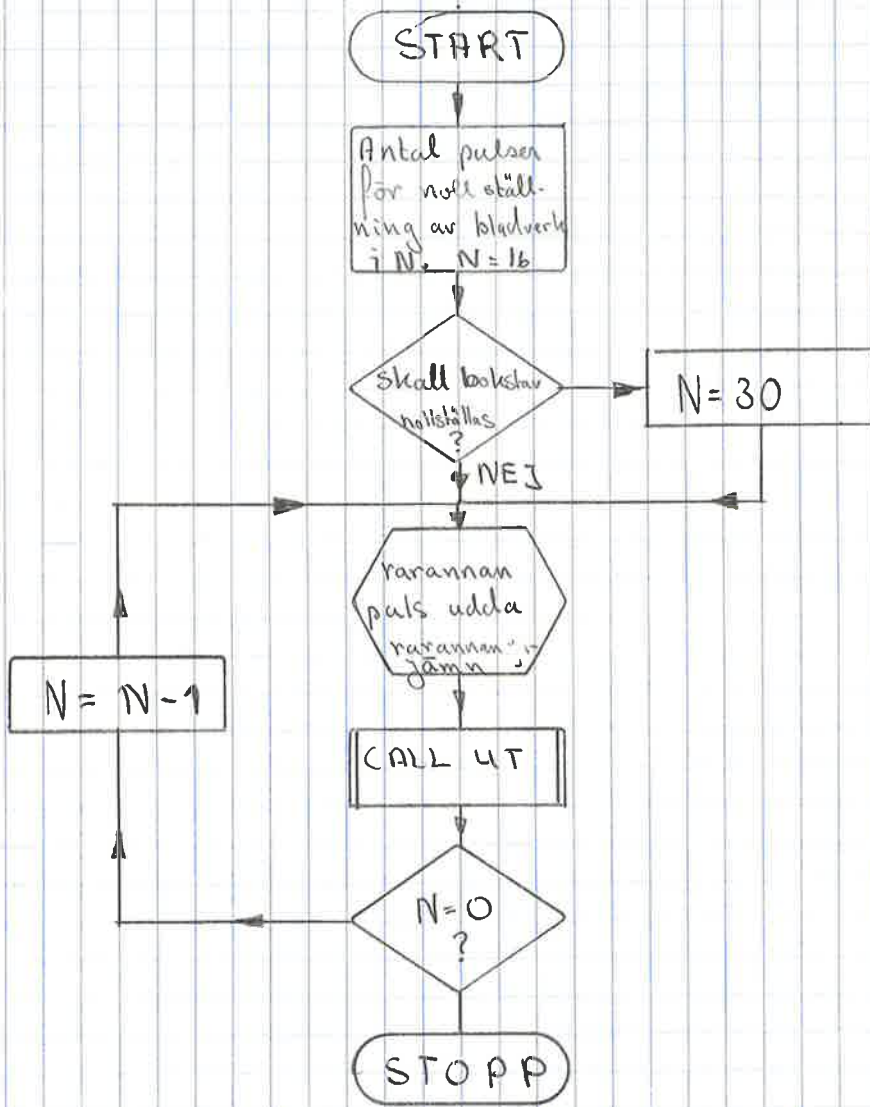
START

Nollställ minnes ord - Sätt adresser till periferantern

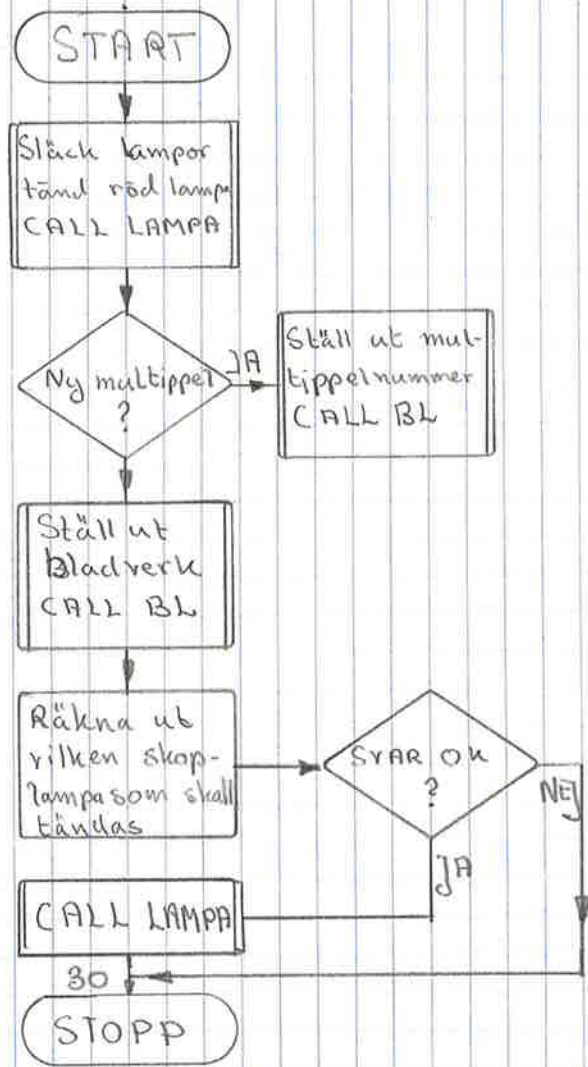
Sätt konstanter - Nollställ månads- och dygnsregistren

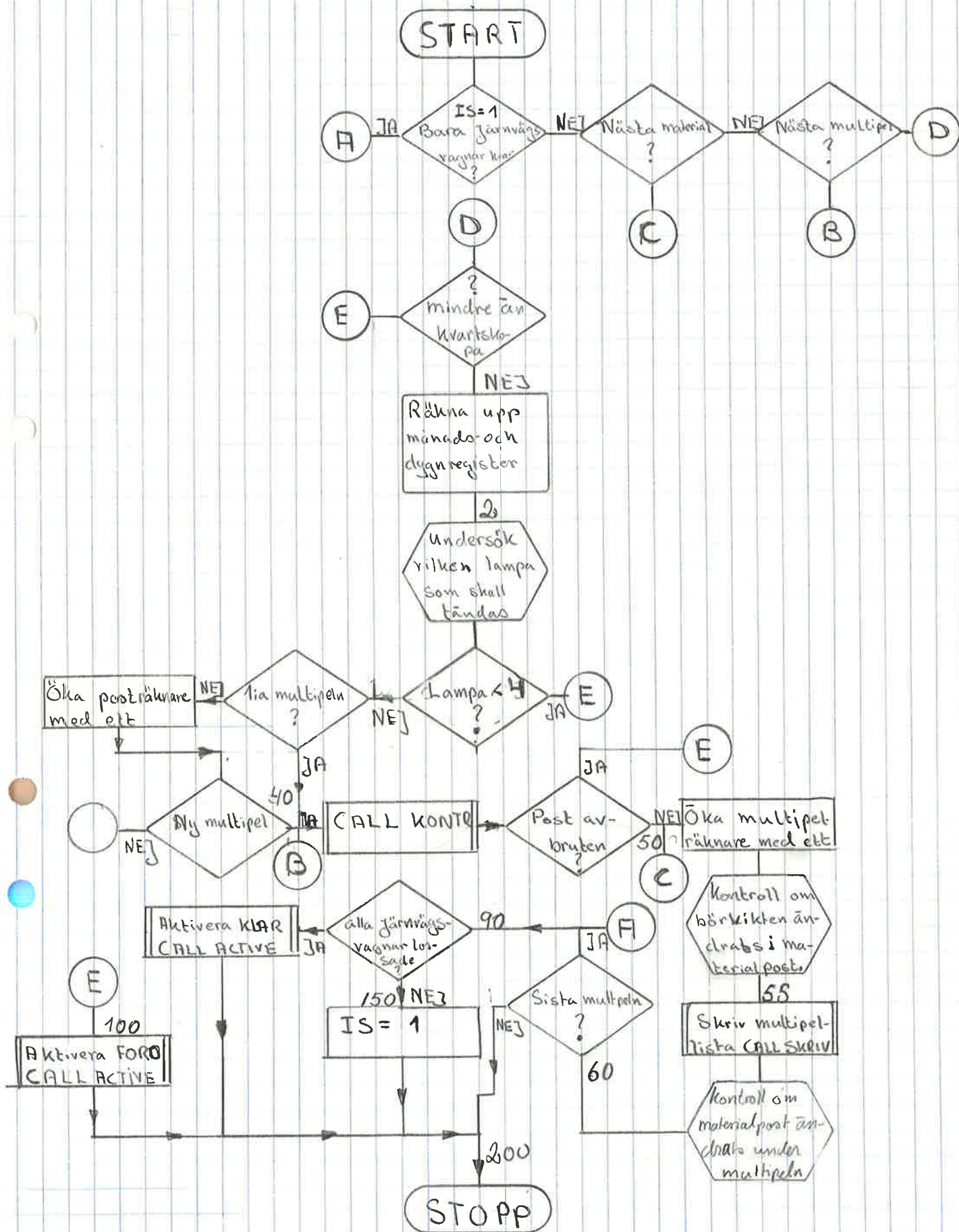
Aktivera STARUP CALL ACTIVE

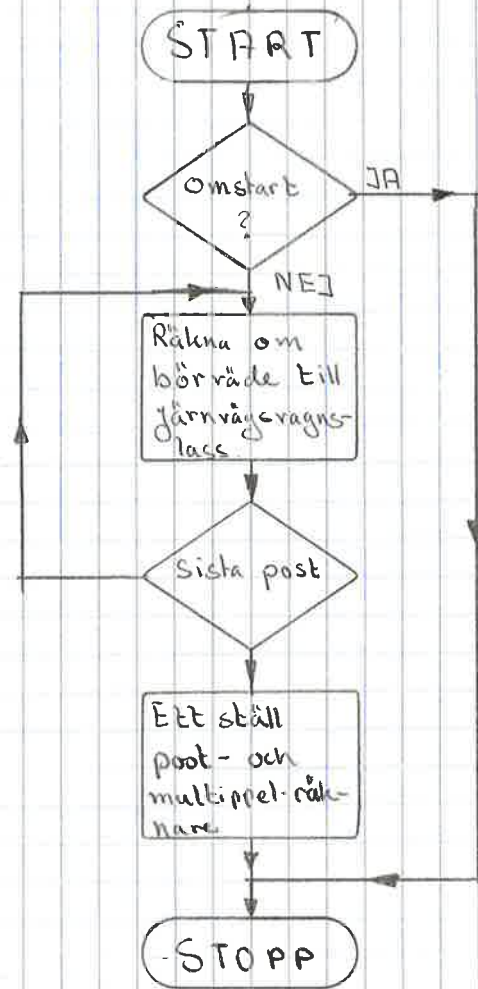
STOPPA

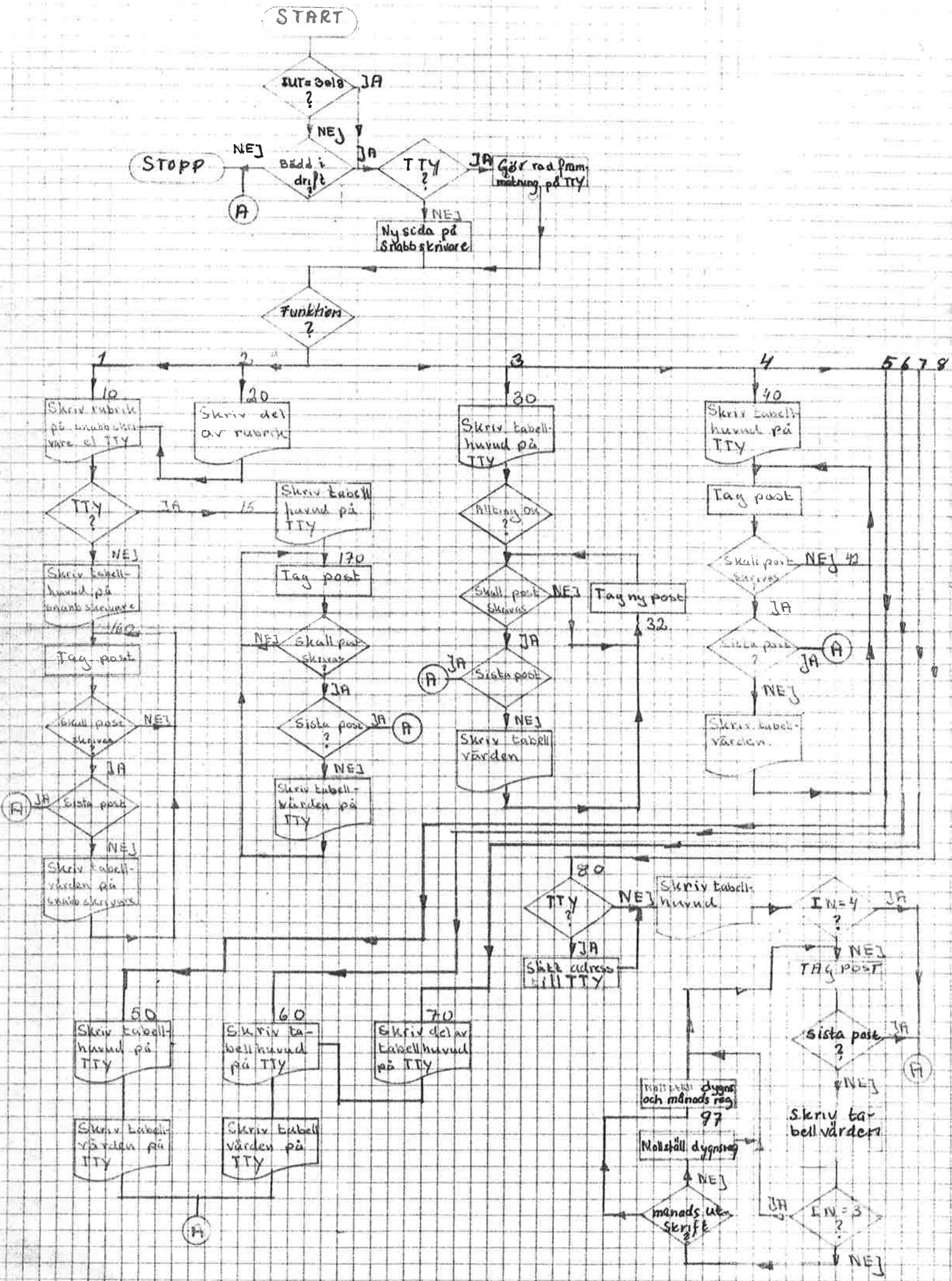


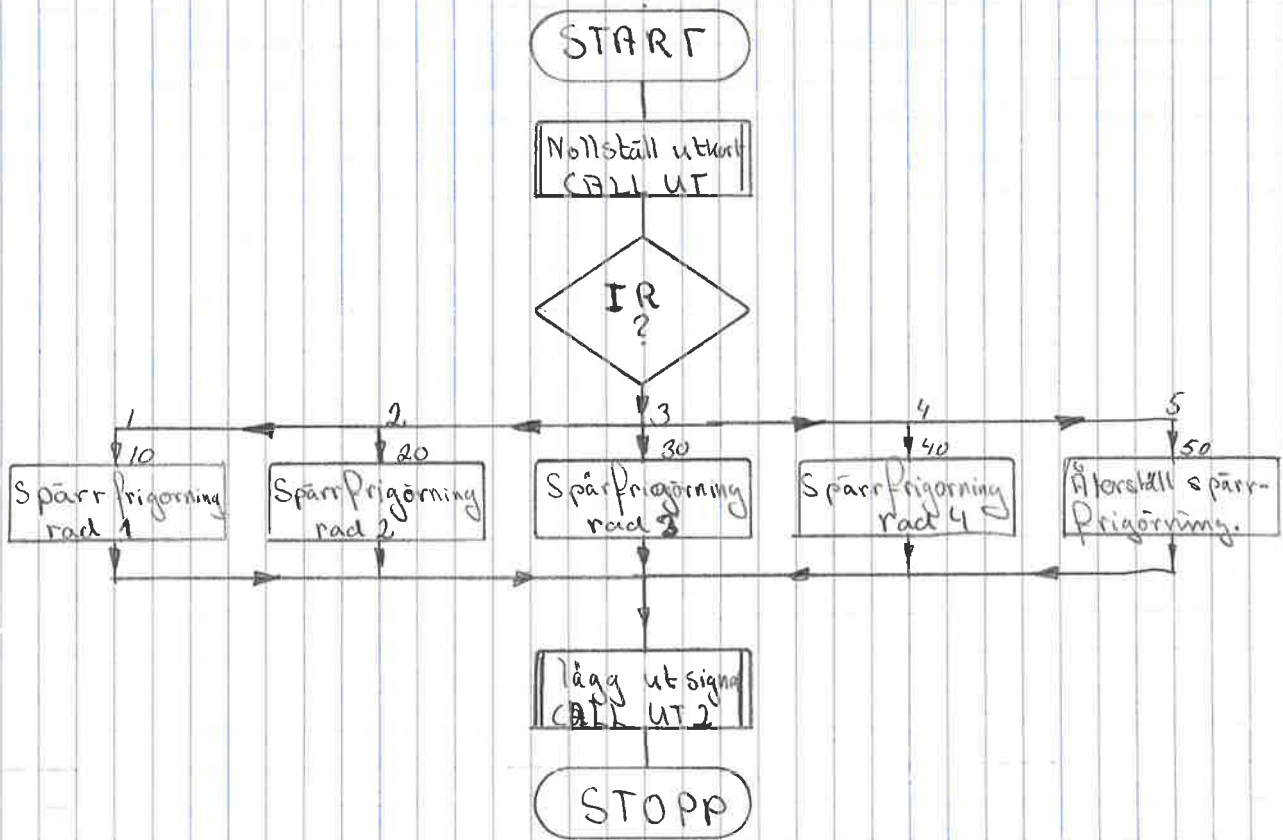
Flödesschema för rutinen RA



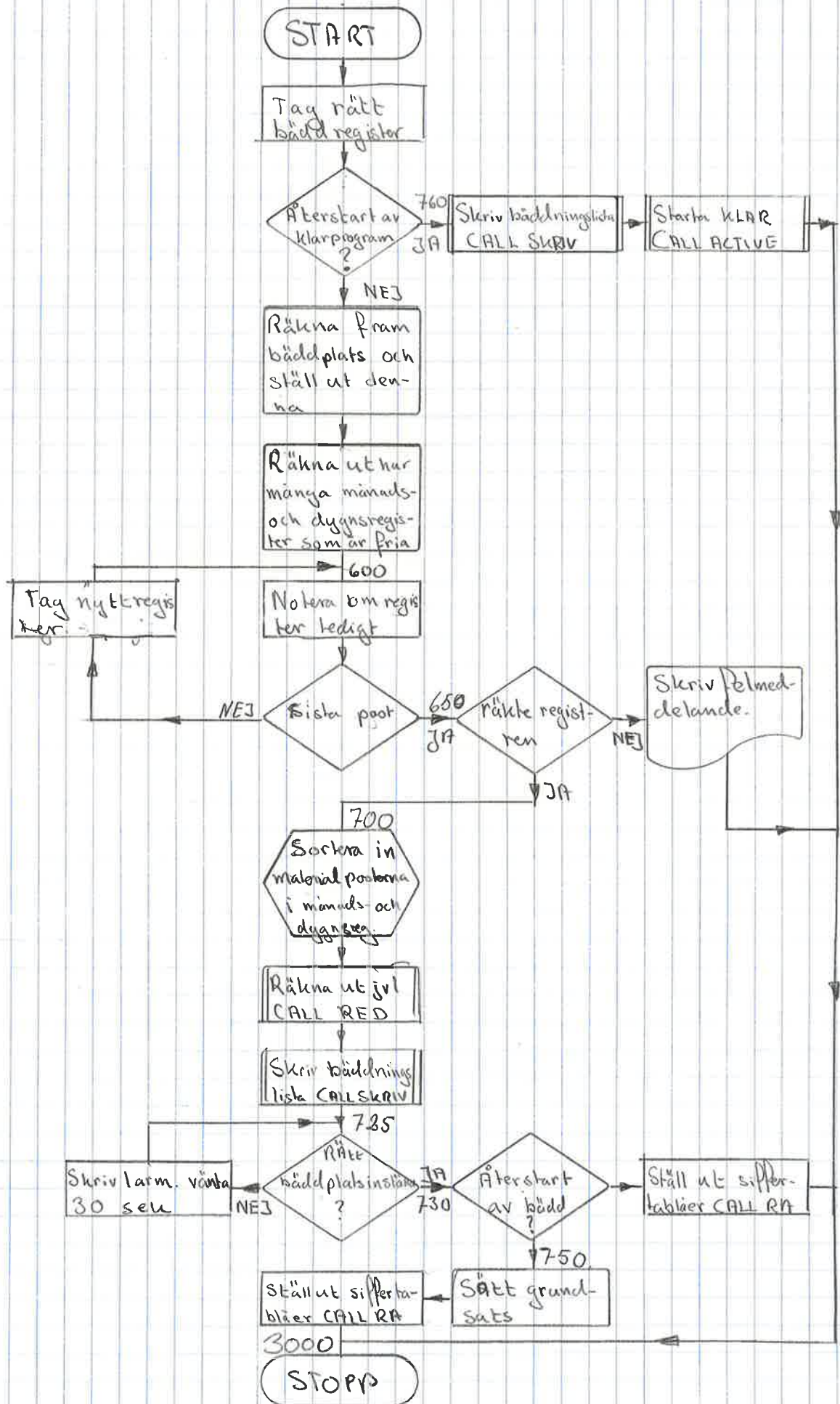


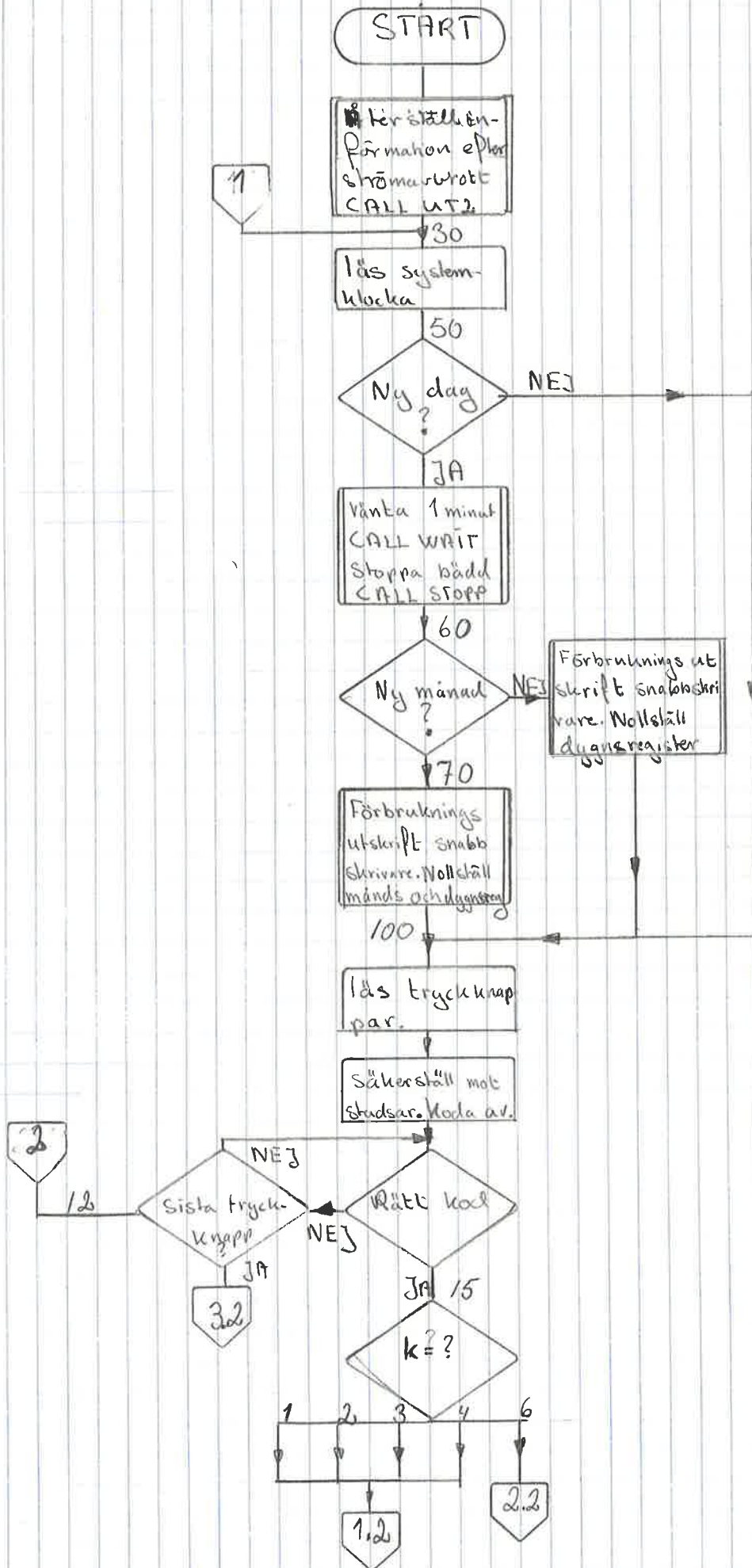


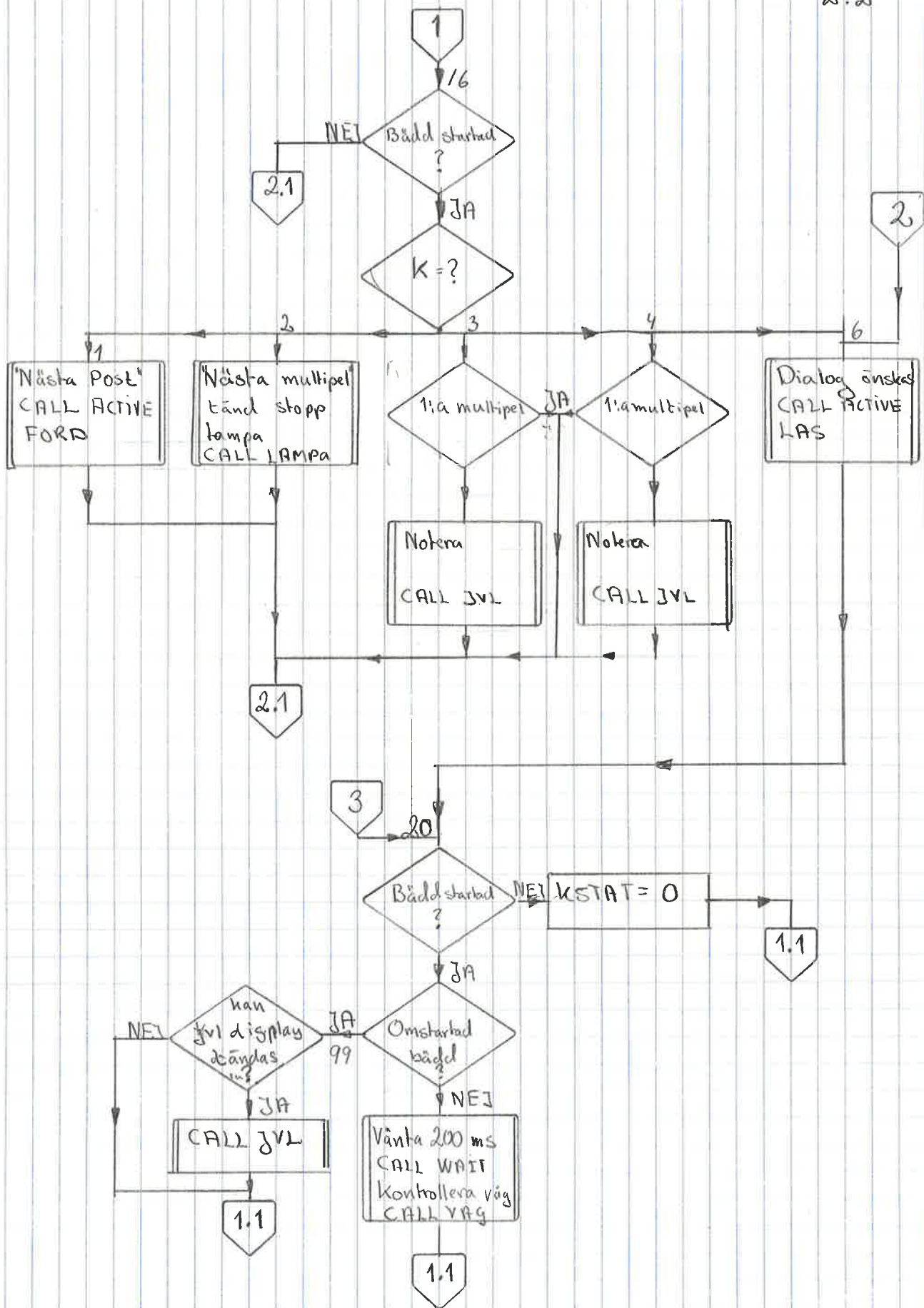




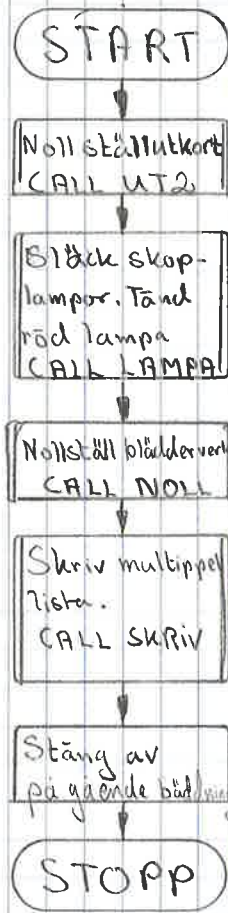
Flödesschema för rutinen START

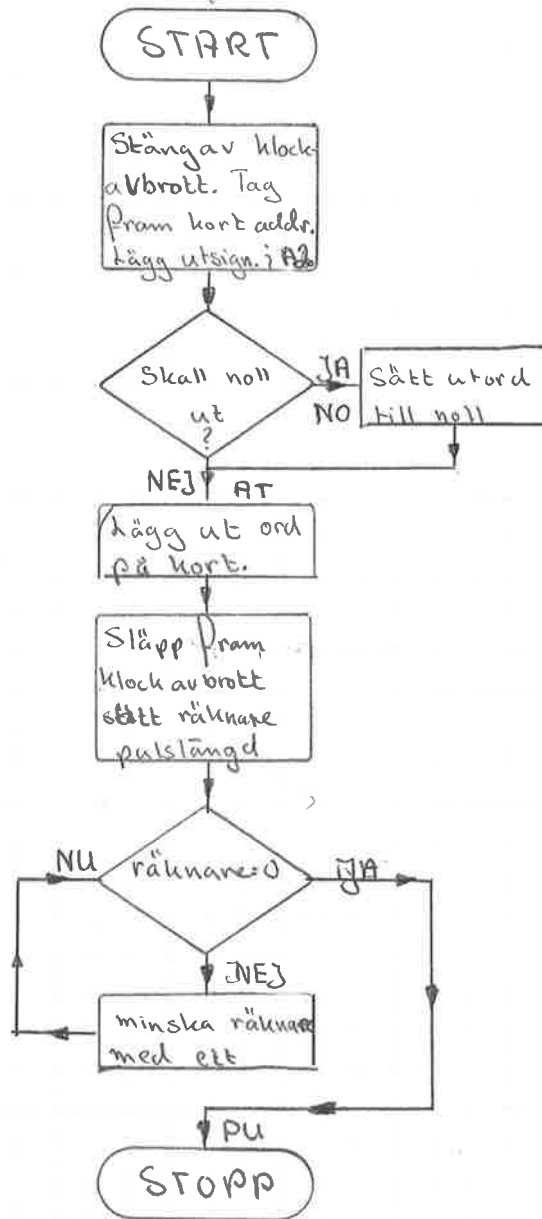


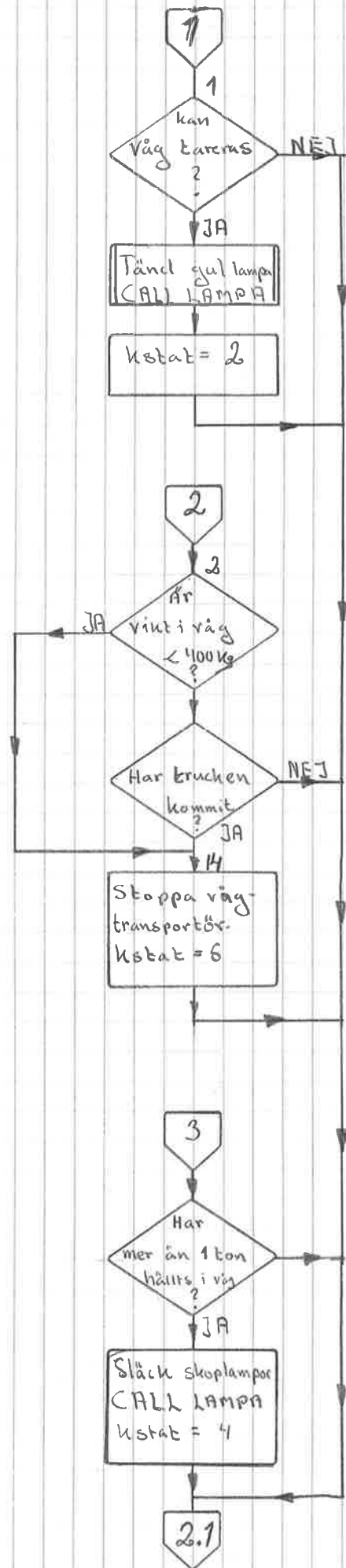


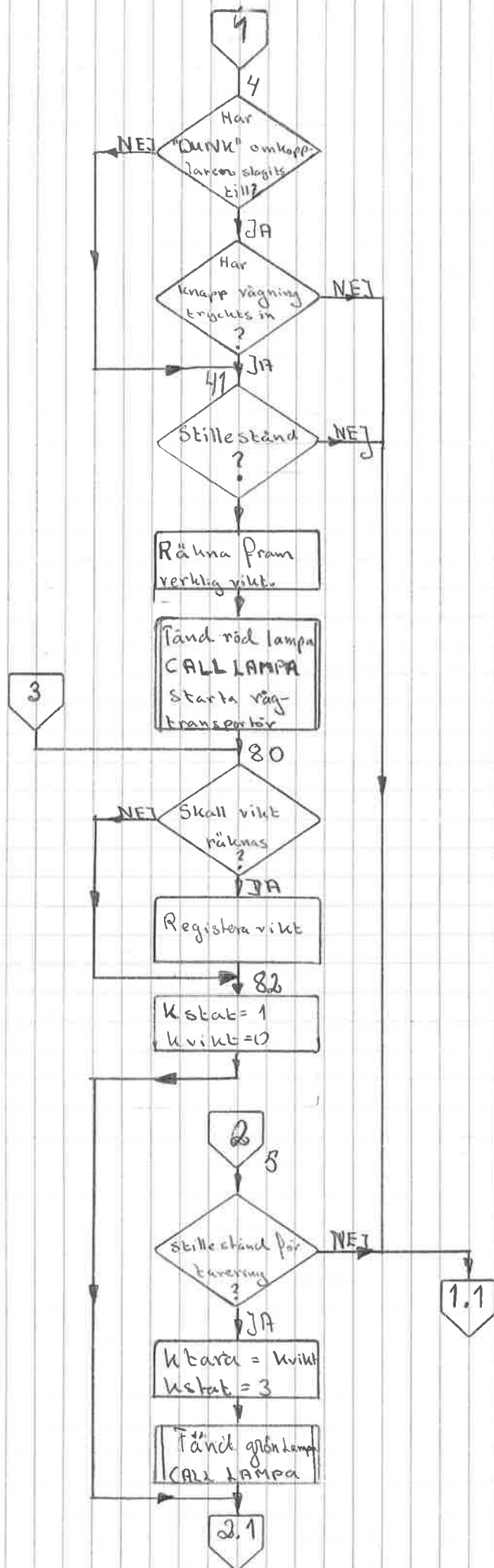


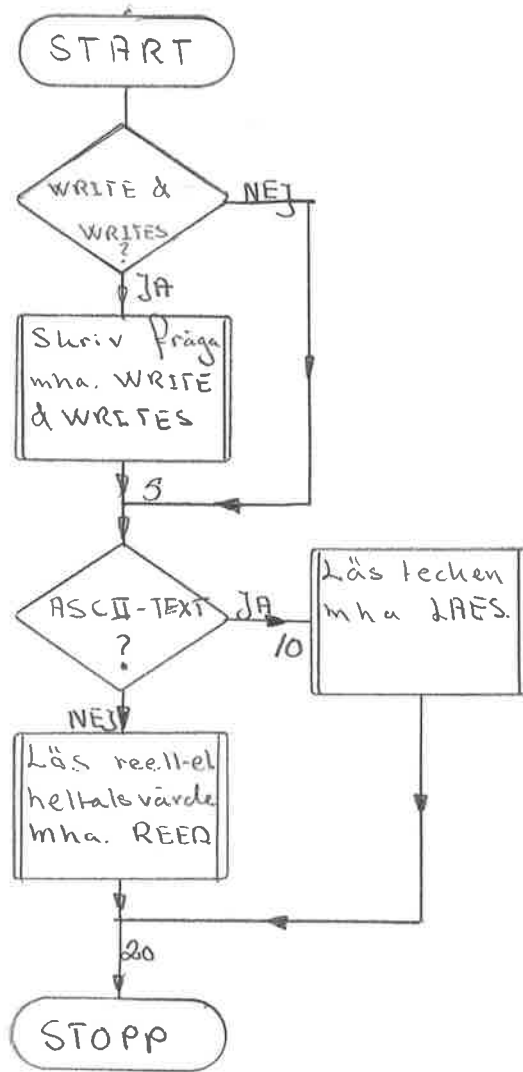
Flödesschema för rutinen STOPP












```

00036 004E 0000 F      SCR      A6,A3
00037 0050 E62D      ADK      A3,1
00038 0052 1301      CCK      A5:/0200
00039 0054 ED21
00040 0056 0200
00041 0058 8820      ABL(0)  UTE
00042 005A 0000 F      ADK      A5,2
00043 005C 1502      ABL      NSF
00044 005E 8F20
00045 0060 0034      SIF      A1,A3
00046 0062 E12D
00047 0064 1301      ADK      A3,1
00048 0066 39E8      SRC      A1,8
00049 0068 E12D      SIFA     A1,A3
00050 006A 1301      ADK      A3,1
00051 006C ED21      CCK      A5:/0200
00052 006E 0200
00053 0070 8820      ABL(0)  UT
00054 0072 0000 F      ADK      A5,2
00055 0074 1502      LD       A1,D,A5
00056 0076 8154
00057 0078 0000 R      SRC      A1,8
00058 007A 39E8      ABL      SIF
00059 007C 8F20
00060 007E 0062      LDKL     A6,/3030
00061 0080 8620
00062 0082 3030
00063 0084 1801      SUK      A3,1
00064 0086 E62D      SCR      A6,A3
00065 0088 F03A      RTN      A14
00066 0088 F03A      END

```

SYMBOL TABLE

ASCII	0004	R	D	0000	R	IG	0010	R	RED	002C	R
NSF	0034	R	SIF	0062	R	SIFA	0068	R	UTE	0080	R
UT	0088	R									

ASS.ERR. 00000

:EOF
PROG ELAPSED TIME: 00H-00M-00S-000MS-

```

00000 IDENT BCD
00001 * CALL BCD(SIF1,SIF2,SIF3,UTORD)
00002 * THIS ROUTINE TAKES SIF1-SIF3 AND PUTS THEM
00003 * IN BCD-CODE IN UTORD.SIF1-SIF3 SHOULD
00004 * BE BETWEEN 0-9 AND INTEGERS.THE FOUR
00005 * LAST BITS IN UTORD WILL BE ZEROS.
00006 *
00007 *
00008 *
00009 *
00010 *
00011 *
00012 *
00013 *
00014 *
00015 *
00016 *
00017 *
00018 *
00019 *
00020 *
00021 *
00022 *
00023 *
00024 *
00025 *
00026 *
00027 *
00028 *
00029 *
00030 *
00031 *
00032 *
00033 *
00034 *
00035 *
00036 *
00037 *
00038 *
00039 *

```

```

MANR1 ENTRY BCD
RES 1
LOAD REG A1-A3 WITH SIF1-SIF3 FINALLY
LOAD MANR1 WITH THE ADDRESS TO THE ANSWER-AREA.
BCD LDR* A7,A4 LOAD A1 WITH SIF1
LDR* A1,A7
ADK A4,2
LDR* A7,A4 LOAD A2 WITH SIF2
LDR* A2,A7
ADK A4,2
LDR* A7,A4 LOAD A3 WITH SIF3
LDR* A3,A7
ADK A4,2
LDR* A7,A4 STORE ADR TO UTORD IN MANR 1
ST A7,MANR1
R
00026
00027
00028
00029
00030
00031
00032
00033
00034
00035
00036
00037
00038
00039

```

```

BY SHIFTING THE DIFFERENT REGISTERS A1=-A3
DIFFERENT STEPS ITS POSSIBLE TO ADD THEM
ALL TOGETHER IN A1, THEN STORE THE RESULT IN
UTORD.
SLL A1,12
SLL A2,8
SLL A3,4
ADR A1,A2
ADR A1,A3
ST* A1,MANR1
RTN A14
END

```

```

SYMBOL TABLE
BCD 0002 R MANR1 0000 R

```

```

0000 IDENT BL
0001 SUBROUTINE BL(IR,IA1,IA2)
0002 C
0003 C
0004 C DENNA RUTIN BERÄKNAR DET ANTAL STEG-PULSER SOM
0005 C BEHÖVS FÖR ATT STÄLLA UPP EN SIFFRA ELLER EN
0006 C BOKSTAV PÅ BLADVEKET.
0007 C
0008 C RUTINEN ANROPAS CALL BL(IR,IA1,IA2)
0009 C IR=DEN RAD SOM SKALL ÄNDRAS PÅ BLADVERKET
0010 C IA1,IA2=DE ASCII-TECKEN SOM SKALL STÄLLAS UT
0011 C
0012 C
0013 C INTEGER BOKST(27),SIF(13),MA(4),KS(4),IB(2)
0014 C
0015 C DATA-AREA SOM INNEHÅLLER AV KODNINGS TABELLER FÖR
0016 C BOKSTÄVER OCH SIFFROR.
0017 C
0018 C DATA (BOKST(K),K=1,27)/$0045,$004E,$0041,$0052,
0019 C *$004F,$0043,$0048,$0055,$0046,$0056,$0057,
0020 C *$0059,$0058,$004B,$0020,$0049,$0053,$0054,$004C,
0021 C *$0044,$004D,$0050,$0047,$0042,$004A,$005A,$0051/
0022 C DATA (SIF(K),K=1,13)/$0031,$0032,$0033,$0034,$0035,
0023 C *$0036,$0037,$0038,$0039,$0030,$0001,$0001,$0020/
0024 C GO TO(10,20,30,40),IR
0025 C
0026 C RUTIN FÖR MULTIPELNUMRET
0027 C FÖRST SKER OMVANDLING TILL ASCII-KOD
0028 C
0029 10 CALL ASCII(IA1,IB(1))
0030 MA(1)=IBYTE(IADDR(IB(2)),0)
0031 MA(2)=IBYTE(IADDR(IB(2)),1)
0032 DO 15 K=1,2
0033 DO 13 N=1,13
0034 IF(MA(K)-EQ.SIF(N)) GO TO 51
0035 CONTINUE
0036 CONTINUE
0037 GO TO 55
0038 C
0039 C RUTIN FÖR HÖGNUMRET
0040 C
0041 20 MA(1)=IBYTE(IADDR(IA1),0)
0042 MA(2)=IBYTE(IADDR(IA1),1)
0043 MA(3)=IBYTE(IADDR(IA2),0)
0044 DO 25 K=1,2

```

```

0045 DO 23 N=1,13
0046 IF(MA(K).EQ.SIF(N)) GO TO 51
0047 CONTINUE
0048 CONTINUE
0049 DO 27 N=1,27
0050 IF(MA(3).EQ.BOKST(N)) CALL BLA(IR,KS(1),KS(2),N,KS(4))
0051 GO TO 56
0052
0053 RUTIN FÖR BÖRVIKT
0054
0055 CALL ASCII(IA1,IB(1))
0056 GO TO 45
0057
0058 RUTIN FÖR ÄRVIKT
0059
0060 CALL ASCII(IA1,IB(1))
0061 MA(1)=IBYTE(IADDR(IB(1)),0)
0062 MA(2)=IBYTE(IADDR(IB(1)),1)
0063 MA(3)=IBYTE(IADDR(IB(2)),0)
0064 MA(4)=IBYTE(IADDR(IB(2)),1)
0065 DO 50 K=1,4
0066 DO 53 N=1,13
0067 IF(MA(K).EQ.SIF(N)) GO TO 51
0068 CONTINUE
0069 CONTINUE
0070 GO TO 55
0071 KS(K)=N
0072 GO TO(15,25,50,50),IR
0073 CONTINUE
0074
0075 ANROP AV STEGNINGS-RUTIN
0076
0077 CALL BLA(IR,KS(1),KS(2),KS(3),KS(4))
0078 CONTINUE
0079 RETURN
0080 END
:EOS
:EOF

```

```

0000 IDENT BLA
0001 SUBROUTINE BLA(IR,KS1,KS2,KS3,KS4)
0002 C
0003 C
0004 C DENNA RUTIN STEGAR BLADVERKET MED DET ANTAL PULSER
0005 C SOM ANGES I ANROPET.
0006 C
0007 C RUTINEN ANROPAS CALL BLA(IR,KS(1),KS(2),KS(3),KS(4))
0008 C IR= DEN RAD PÅ BLADVERKET SOM SKALL STÄLLAS UT
0009 C KS(1)-KS(4)=ANTAL PULSER FÖR BLADVERK 1-4 I RADEN
0010 C
0011 C
0012 C INTEGER REL(4),BCL(4)
0013 C DATA (REL(I),I=1,4) /#C002,$3802,$0782,$007A/
0014 C DATA (BCL(I),I=1,4) /0,2,5,9/
0015 C CALL NOLL(IR)
0016 C N=15
0017 C
0018 C OM BOKSTAV I RAD 27 PULSER
0019 C
0020 C IF (IR.EQ.2) N=27
0021 C L=0
0022 C IU=REL(IR)
0023 C DO 10 M=1,N
0024 C
0025 C L=0 GER POSIV PULS L=1 GER NEGATIV PULS
0026 C
0027 C IF (L.EQ.2) L=0
0028 C IF (L.EQ.0) CALL BCLR(IU,0)
0029 C IF (L.EQ.0) CALL BSET(IU,1)
0030 C
0031 C OM FÖRSTA PULS DRAG SPÄRRFRIGÖRARE
0032 C
0033 C IF (M.EQ.1) CALL SP(IR)
0034 C CALL UT(IU)
0035 C
0036 C OM FÖRSTA PULS SLÄPP SPÄRRFRIGÖRARE
0037 C
0038 C IF (M.EQ.1) CALL SP(5)
0039 C CALL UT(5)
0040 C IB=15-BCL(IR)
0041 C
0042 C KONTROLL AV OM NGT BLADVERK STEGAT KLART
0043 C
0044 C IF (KS1.EQ.M) CALL BCLR(IU,IB)
0045 C IB=15-(BCL(IR)+1)
0046 C IF (KS2.EQ.M) CALL BCLR(IU,IB)
0047 C IB=15-(BCL(IR)+2)

```

```

0048 IF(KS3.EQ.M) CALL BCLR(IU,IB)
0049 IB=15-(BCL(IR)+3)
0050 IF(KS4.EQ.M) CALL BCLR(IU,IB)
0051 C
0052 C
0053 C
0054 IKL=IAND(IU,$FFF8)
0055 IF(IKL.EQ.0) GO TO 20
0056 CALL BCLR(IU,1)
0057 CALL BSET(IU,0)
0058 L=L+1
0059 CONTINUE
0060 RETURN
0061 END
:EOS
:EOF

```

S:

IDENT FORD

DENNA RUTIN BERÄKNAR HUR MYCKET AV VARJE MATERIAL
SOM SKALL LOSSAS INOM MULTIPLELN.

COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
*,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
*,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
*,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
INTEGER BUF,BUF1,ECB(2)
IECB=IADDR(ECB)

BÄDD STARTAD ?

IF(NR.EQ.0) GO TO 40

FÖRSTA MULTIPLELN ?

IF(MT(NR).NE.1) GOTO 5
BD(NR,1,3)=GR(NR)
GO TO 20

RÄKNA FRAM FAKTOR FÖR AMTAL RESTERANDE MULTIPLAR

RMT=IA(NR)-(MT(NR)-1)
IF(RMT.LE.1.) RMT=1.

BEORDRING AV AITIK VAGNAR

IF(IBD(NR,LP(NR),1).NE.2) GO TO 8
IF(IBD(NR,LP(NR),6).LE.0) IBD(NR,LP(NR),6)=0
IF(IBD(NR,LP(NR),7).LE.0) IBD(NR,LP(NR),7)=0
IF(IBD(NR,LP(NR),7).EQ.0) GO TO 7
IBD(NR,LP(NR),8)=IBD(NR,LP(NR),7)
LP(NR)=LP(NR)+1

BEORDRING AV STEKENJOKK

IF(IBD(NR,LP(NR),1).NE.3) GO TO 15
IF(IBD(NR,LP(NR),6).LE.0) IBD(NR,LP(NR),6)=0
IF(IBD(NR,LP(NR),7).LE.0) IBD(NR,LP(NR),7)=0
IF(IBD(NR,LP(NR),7).EQ.0) GO TO 15
IJV=((IBD(NR,LP(NR),6)-IBD(NR,LP(NR),8))/RMT)+0.5)
IBD(NR,LP(NR),8)=IBD(NR,LP(NR),8)+IJV
IJV=0

0000
0001 C
0002 C
0003 C
0004 C
0005 C
0006 C
0007
0008
0009
0010
0011
0012
0013 C
0014 C
0015 C
0016
0017 C
0018 C
0019 C
0020
0021
0022
0023 C
0024 C
0025 C
0026 S
0027
0028 C
0029 C
0030 C
0031
0032
0033
0034
0035
0036 7
0037 C
0038 C
0039 C
0040
0041
0042
0043
0044
0045
0046

```

0047 IF(MT(NR).EQ.IA(NR)) IBD(NR,LP(NR),8)=IBD(NR,LP(NR),6)
0048 IF(MT(NR).GE.(IA(NR)+1)) GO TO 40
0049 C
0050 OM MÖJLIGT REFRESH AV SJUSEGMENT-DISPLAY
0051 C
0052 IF(IBM(NR,LP(NR),1).LE.1) GO TO 16
0053 ISK=4
0054 CALL ACTIV('JVL ',IECB,D,M)
0055 C
0056 KAN POST LOSSAS ?
0057 C
0058 IF(BD(NR,LP(NR),1).EQ.0.) GO TO 40
0059 IF(BD(NR,LP(NR),1).LE.0.) GO TO 29
0060 IF(IBM(NR,LP(NR),15).EQ.1) GO TO 29
0061 IF(IBM(NR,LP(NR),15).GE.3) GO TO 29
0062 IF(IBM(NR,LP(NR),10).NE.0) GO TO 29
0063 IF((IBM(NR,LP(NR),1).EQ.2).OR.(IBM(NR,LP(NR),1).EQ.3)) GO
TO 29
0064 IF(IBM(NR,LP(NR),11).NE.0) GO TO 20
0065 C
0066 RÄKNA UT HUR MYCKET SOM SKALL LOSSAS I DENNA MULTIPEL
0067 C
0068 BD(NR,LP(NR),3)=BD(NR,LP(NR),4)+BD(NR,LP(NR),2)/RMT
0069 IF((BD(NR,LP(NR),3)-BD(NR,LP(NR),4)).LT.(0.125*SKL)) GO TO
29
0070 C
0071 MÅSTE KVARTSSKOPA BEORDRAS ?
0072 C
0073 IF(MT(NR).EQ.IA(NR)) BD(NR,LP(NR),3)=BD(NR,LP(NR),1)
0074 IF((BD(NR,LP(NR),2)/RMT).LE.SKL) IBD(NR,LP(NR),14)=1
0075 C
0076 ÄNDRA PÅ INDIKATIONSTABELÄRNA
0077 C
0078 CALL RA
0079 GO TO 40
0080 LP(NR)=LP(NR)+1
0081 GO TO 5
0082 CONTINUE
0083 STOP
0084 END
:EOS
:EOF

```


IDENT JVL

DENNA RUTIN SKÖTER KONTROLLEN
AV JÄRNVÄGSLOSTNINGS-PLATSEN.

COMMON IBDNR(2), IA(2), MT(2), LP(2), GR(2), IBD(2,20,15)
*, BD(2,20,5), NR, IS, ISK, BOR, KP(2), BUF(4), BUF1(4), IBDN
*, MAN, TARA, VL, SKL, MANR, MANR1, IDEV1, IDEV2, IDEV3
*, IDEV4, IDEV5, IDEV6, MREG(30,3), REG(30,2)
INTEGER BUF, BUF1, ECB(2)

LETA UPP POST NR 2 OCH 3

DO 1 K=1,20
IF(IBD(NR,K,1).EQ.3) KF=K
IF(IBD(NR,K,1).EQ.2) KS=K
CONTINUE
KN=KF

OM ISK=4 ENDAST REFRESH AV DISPLAY

IF(ISK.EQ.4) GO TO 15

ISK=1 STEKENJOKK ISK=0 AITIK

IF(ISK.EQ.1) KN=KS
IF((ISK.NE.1).AND.(IBD(NR,KF,8).EQ. IBD(NR,KF,9))) GO TO 30

FINNS DET VAGNAR ATT LOSSA ?

IF((IBD(NR,KN,8).EQ.0).OR.(IBD(NR,KN,7).EQ.0)) GO TO 30
IF((ISK.NE.1).AND.(ISK.NE.0)) GO TO 15

KVITTERA LOSSAD VAGN PÅ DISPLAY

ISK=0
IBD(NR,KN,7)=IBD(NR,KN,7)-1
IBD(NR,KN,9)=IBD(NR,KN,9)+1
IJ=IBD(NR,KF,8)-IBD(NR,KF,9)

MANR1=IAND(MANR1,#000F)
CALL BCD(IBD(NR,KS,7), IBD(NR,KS,9), IJ, IBC)
MANR1=MANR1+IBC
CALL UT2(?,MANR1)

0000
0001 C
0002 C
0003 C
0004 C
0005 C
0006 C
0007
0008
0009
0010
0011
0012 C
0013 C
0014 C
0015
0016
0017
0018 1
0019
0020 C
0021 C
0022 C
0023
0024 C
0025 C
0026 C
0027
0028

0029 C
0030 C
0031 C
0032
0033
0034 C
0035 C
0036 C
0037
0038
0039
0040 15
0041
0042
0043 20
0044

0045 30
0046
0047
:E09
:E0F

CONTINUE
STOP
END

S:

IDENT KLAR

DENNA RUTIN ANVÄNDS FÖR ATT LÄSA IN DE VERKLIGA
 VIKTERNA PÅ JÄRNVÄGSMANRARNAS SOM INGÅR I BÄDDEN.
 RUTINEN SER ÄVEN TILL ATT EN SLUTUTSKRIFT SKER SAMNT
 ATT BÄDD-REGISTREN FRIGÖRS OCH KAN TA EMOT NYA
 BÄDD DATA.

```

COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
*BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
*,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
*,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
INTEGER BUF,BUF1,ITE(2)
DIMENSION BV(2),IN(2)
ISK=3
IN(1)=2
IN(2)=3
BS=0.
BA=0.
    
```

RESERVERA TTY OCH PER. STARTA TTY.

```

CALL ATTCHW(IDEV1,M)
CALL ATTCHW(IDEV2,M)
WRITE(IDEV2,106)
    
```

LÄGG ASCII-TECKEN I REGISTER.

```

IBEL=#0707
ISP=#2020
IAST=#2020
ITE(1)=#2053
ITE(2)=#2041
    
```

NOLLSTÄLL INDIKATIONS-TABLÅER.

```

MANR1=IAND(MANR1,#000F)
CALL UT2(2,MANR1)
CALL BCLR(MANR,0)
CALL LAMPA(7)
CALL LAMPA(4)
CALL NOLL(5)
    
```

IFTA FRAM POST 2 OCH 3.

- 0000
- 0001 C
- 0002 C
- 0003 C
- 0004 C
- 0005 C
- 0006 C
- 0007 C
- 0008 C
- 0009 C
- 0010
- 0011
- 0012
- 0013
- 0014
- 0015
- 0016
- 0017
- 0018
- 0019
- 0020
- 0021 C
- 0022 C
- 0023 C
- 0024
- 0025
- 0026
- 0027 C
- 0028 C
- 0029 C
- 0030
- 0031
- 0032
- 0033
- 0034
- 0035 C
- 0036 C
- 0037 C
- 0038
- 0039
- 0040
- 0041
- 0042
- 0043
- 0044 C
- 0045 C

```

0046 C DO 5 K=1,20
0047 IF(IBD(NR,K,1).EQ.2) IN(1)=K
0048 IF(IBD(NR,K,1).EQ.3) IN(2)=K
0049 5
0050 C
0051 C MEDDELA BÄDD KLAR.
0052 C
0053 C WRITE(IDEV2,100)IBDNR(NR)
0054 C
0055 C FINNS VAGNSLASTER I BÄDDEN ?
0056 C
0057 C IF(IBD(NR,IN(1),9).NE.0) GO TO 4
0058 C IF(IBD(NR,IN(2),9).EQ.0) GO TO 41
0059 C
0060 C KAN JÄRNVÄGSVAGNARNA LÄSAS IN
0061 C
0062 C 4 WRITE(IDEV2,101)IBEL
0063 C JA=7
0064 CALL WRL(ISP,2,REAL,JA,2,IDEV2)
0065 IF(JA.EQ.$4A41) GO TO 2
0066 IBD(NR,1,8)=1
0067 WRITE(IDEV2,107)
0068 WRITE(IDEV1,109)
0069 WRITE(IDEV2,110)
0070 CALL DETACH(IDEV1,M)
0071 CALL DETACH(IDEV2,M)
0072 GO TO 144
0073 C
0074 C LÄS IN JÄRNVÄGSVAGNARNAS VERKLIGA VIKTER
0075 C
0076 C 2 CONTINUE
0077 DO 40 IP=1,2
0078 WRITE(IDEV2,102)ITE(IP)
0079 BV(IP)=0.
0080 DO 10 IH=1,100
0081 IF(IBD(NR,IN(IP),9).EQ.0) GO TO 30
0082 IBD(NR,IN(IP),9)=IBD(NR,IN(IP),9)-1
0083 WRITE(IDEV2,103)IH,IBEL
0084 BR=0.
0085 INT=1
0086 CALL WRL(ISP,2,BR,INT,0,IDEV2)
0087 IF(BR.LE.0.) IAST=#2A20
0088 10 BV(IP)=BV(IP)+BR
0089 30 BD(NR,IN(IP),4)=BV(IP)
0090 BD(NR,IN(IP),2)=BD(NR,IN(IP),1)-BV(IP)
0091 REG(1BD(NR,IN(IP),13),1)=REG(1BD(NR,IN(IP),13),1)+BV(IP)
0092 REG(1BD(NR,IN(IP),13),2)=REG(1BD(NR,IN(IP),13),2)+BV(IP)
0093 C 40 CONTINUE

```

```

0094 WRITE(IDEV2,110)
0095 CONTINUE
0096 CALL DETACH(IDEV2,M)
0097 C
0098 C GÖR SLUTUTSKRIFT SAMNT SUMMERA MATERIALEN
0099 C NOLLSTÄLL BÄDDREGISTRET
0100 C
0101 CALL ATTCHW(IDEV1,M)
0102 CALL SKRIV(2,0)
0103 DO 140 K=1,20
0104 BS=BS+BD(NR,K,1)
0105 BA=BA+BD(NR,K,4)
0106 DO 130 J=1,15
0107 IBD(NR,K,J)=0
0108 DO 135 J=1,4
0109 BD(NR,K,J)=0.
0110 CONTINUE
0111 IBDNR(NR)=0
0112 MT(NR)=0
0113 IA(NR)=0
0114 LP(NR)=0
0115 C
0116 C SKRIV SLUTSUMMAN PÅ BÄDDEN
0117 C
0118 IF(IDEV1.EQ.5) WRITE(IDEV2,104)BS,BA,IAST
0119 IF(IDEV1.EQ.5) WRITE(IDEV2,108)
0120 IF(IDEV1.EQ.5) GO TO 144
0121 WRITE(IDEV1,105)BS,BA,IAST
0122 WRITE(IDEV1,108)
0123 CALL DETACH(IDEV1,M)
0124 GO TO 144
0125 FORMAT(' BÄDD NUMMER',I4,' KLAR')
0126 FORMAT(' VILL DU LÄSA IN VAGNSLASTVIKTERNA?',A2)
0127 FORMAT(' DIREKT MATERIAL',A2)
0128 FORMAT(' VAGNSLAST',I3,' VIKT TON?',A2)
0129 FORMAT(1H '7X','TOTALT',F7.2,23X,'TOTALT',F7.2,A2)
0130 FORMAT(1H '8X','TOTALT',F7.2,49X,'TOTALT',F7.2,A2)
0131 FORMAT(' ',/)
0132 FORMAT(' BÄDDNINGEN AVBRUTEN')
0133 FORMAT(1H0,' *=VAGNSLASTER SAKNAS')
0134 FORMAT(1H0,' BÄDDNINGEN AVBRUTEN;DÅ VAGNSLASTER EJ INLÄSITS
)
0135 FORMAT('/',/, ' DIALOG SLUT')
0136 NR=0
0137 IS=0
0138 STOP
0139 END
:EOS

```

S:

:EOF

```

0000 IDENT KONTR
0001 SUBROUTINE KONTR(LK)
0002 C
0003 C
0004 C DENNA RUTIN KONTROLLERAR OM NÅGOT MATERIAL HAR BLIVIT
0005 C AVBRUTET UNDER MULTIPLELN. SKULLE SÅ VARA FALLET SÅ SER
0006 C RUTINEN TILL ATT MATERIALET BEORDRAS IGEN INNAN NÄSTA
0007 C MULTIPLEL BEORDRAS.
0008 C
0009 C RUTINEN ANROPAS CALL KONTR(LK)
0010 C LK=POSTNUMMRET PÅ EV. AVBRUTET MATERIAL
0011 C LK=0 INGEN POST AVBRUTEN
0012 C
0013 C
0014 C
0015 C COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0016 C *,IBD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0017 C *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0018 C *,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0019 C INTEGER BUF,BUF1
0020 C
0021 C OM POST AVBRUTEN IBD(, , 10)=0
0022 C
0023 C DO 20 J=1,20
0024 C IF(IBD(NR,J,10).EQ.0) IBD(NR,J,10)=1
0025 C IF(IBD(NR,J,10).EQ.#4156) IBD(NR,J,10)=0
0026 C CONTINUE
0027 C
0028 C HAR NÅGON POST BLIVIT AVBRUTEN ?
0029 C
0030 C DO 10 J=1,20
0031 C IF(IBD(NR,J,10).EQ.0) GO TO 25
0032 C CONTINUE
0033 C
0034 C LK=0 INGEN POST AVBRUTEN
0035 C
0036 C LK=0
0037 C GO TO 50
0038 C
0039 C LK=POSTNUMMER PÅ AVBRUTET MATERIAL
0040 C
0041 C LK=J
0042 C CONTINUE
0043 C RETURN
0044 C END
0045 C RETURN

```

```

0000 IDENT LAMPA
0001 SUBROUTINE LAMPA(IN)
0002 C
0003 C
0004 C DENNA RUTIN TÄNDER LAMPORNA PÅ INDIKATIONS-
0005 C TABLÄRNA BEROENDE PÅ IN:S VÄRDE
0006 C
0007 C RUTINEN ANROPAS CALL LAMPA(IN)
0008 C DÄR IN 1-7
0009 C
0010 C
0011 C COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0012 C *,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0013 C *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0014 C *,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0015 C INTEGER BUF,BUF1
0016 C IF(IN.GE.8) GO TO 100
0017 C IF(IN.LE.0) GO TO 100
0018 C GO TO(10,20,30,40,50,60,70),IN
0019 C
0020 C TÄND HEL SKOPA
0021 C
0022 C 10 MANR=MANR.AND.$F1FF
0023 C CALL BSET(MANR,11)
0024 C GO TO 99
0025 C
0026 C TÄND HALV SKOPA
0027 C
0028 C 20 MANR=MANR.AND.$F1FF
0029 C CALL BSET(MANR,10)
0030 C GO TO 99
0031 C
0032 C TÄND KVARTS SKOPA
0033 C
0034 C 30 MANR=MANR.AND.$F1FF
0035 C CALL BSET(MANR,9)
0036 C GO TO 99
0037 C
0038 C TÄND RÖD LAMPA
0039 C
0040 C 40 MANR=MANR.AND.$FESF
0041 C CALL BSET(MANR,8)
0042 C GO TO 99
0043 C
0044 C TÄND RÖD OCH GUL LAMPA
0045 C
0046 C 50 MANR=MANR.AND.$FESF

```



```

0047 CALL BSET(MANR,8)
0048 CALL BSET(MANR,7)
0049 GO TO 99
0050 C
0051 C TÄND GRÖN LAMPA
0052 C
0053 C MANR=MANR .AND. $FE3F
0054 CALL BSET(MANR,6)
0055 GO TO 99
0056 C
0057 C SLÄCK ALLA LAMPOR
0058 C
0059 C MANR=MANR .AND. $F1FF
0060 CALL UT2(1,MANR)
0061 CONTINUE
0062 RETURN
0063 END
:EOS
:EOF

```

S:

```

0000 IDENT LAS
0001 C
0002 C
0003 C
0004 C
0005 C
0006 C
0007 COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0008 *,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0009 *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3,IDEV4
0010 *,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0011 INTEGER BUF,BUF1,OE
0012 *,TEXT(8,6),TEX1(7),TEX2(9),TEX3(8),TEX5(9),TEX6(5)
0013 *,TEX7(9),TEX8(10),TEX11(4)
0014 DATA ((TEXT(J,I),J=1,6),I=1,6) /
0015 *2HBA,2HDD,2HNU,2HMM,2HER,2H?,2*2H,
0016 *2HAN,2HTA,2HL,2HMU,2HLT,2HIP,2HLA,2HR?,
0017 *2HMA,2HTE,2HRI,2HAL,$485C,2HG?,2*2H,
0018 *$425C,2HRV,2HIK,2HT,2HTO,2HN?,2*2H,
0019 *2HGR,2HUN,2HDS,2HAT,2HS,2HTO,2HN?,2H,
0020 *2HPO,2HST,2HNU,2HMM,2HER,2H?,2*2H /
0021 DATA (TEX1(I),I=1,7) /2HDI,2HAL,2HOG
0022 *,2HNU,2HMM,2HER,2H? /
0023 DATA (TEX2(I),I=1,9) /2HAN,2HDR,2HIN,2HGS,2HMA
0024 *,2HRK,2HER,2HIN,2HG? /
0025 DATA (TEX3(I),I=1,8) /2HAN,2HDR,2HIN,2HG,
0026 *2HAV,2HP,2HOS,2HT? /
0027 DATA (TEX5(I),I=1,9) /2HVA,2HGN,2HSL,2HAS,2HTV,
0028 *2HIK,2HT,2HTO,2HN? /
0029 DATA (TEX6(I),I=1,5) /2HRE,2HGI,2HST,2HER,2H? /
0030 DATA (TEX7(I),I=1,9) /2HSK,2HOP,2HLA,2HST,
0031 *2HVI,2HKT,2HT,2HON,2H? /
0032 DATA (TEX8(I),I=1,10) /2HTA,2HRE,2HRI,2HNG,
0033 *2HSG,2HRA,2HNS,2HT,2HON,2H? /
0034 DATA (TEX11(I),I=1,4) /2HAL,2HLT,2H0,2HK? /
0035 OE=$5C00
0036 CALL ATTCHW(IDEV2,M)
0037 WRITE(IDEV2,3019)
0038 IB=0
0039 C
0040 C
0041 C
0042 C
0043 C
0044 C
0045 C
INLÄSNING AV KOD FÖR DEN DIALOG SOM ÖNSKAS
CALL WRL(TEX1(1),14,REAL,IB,0,IDEV2)
KONTROLL OM INLÄST KOD LIGGER INOM GRÄNSERNA

```

```

0046 IF (IB.LE.0) GO TO 2000
0047 IF ((IB.GE.6).AND.(IB.LE.9)) GO TO 2000
0048 IF ((IB.GE.15).AND.(IB.LE.19)) GO TO 2000
0049 IF ((IB.GE.10).AND.(IB.LE.19)) IB=IB-4
0050 IF (IB.GE.20) IB=IB-9
0051 IF (IB.GE.19) GO TO 2000
0052 GO TO (100,500,600,700,70,850,850,850
0053 *,80,90,200,300,400,1620,1630,1640,1670,1610),IB
0054 C
0055 C   RUTIN FÖR INLÄSNING AV BÄDD VIA TANGETBORD PÅ TTY
0056 C   DELAR AV DENNA RUTIN ANVÄNDS ÄVEN AV ANDRA RUTINER
0057 C
0058 100 WRITE (IDEV2,1001)
0059 101 IF ((IBDNR(1).EQ.0).OR.(IBDNR(2).EQ.0)) GO TO 105
0060 102 GO TO 1050
0061 103 IBDN=0
0062 104 CALL WRL(TEXT(1,1),16,REAL,IBDN,0,IDEV2)
0063 105 IF ((IBDN.LE.99).OR.(IBDN.GE.700)) GO TO 2000
0064 106 IF ((IBDN.EQ.IBDNR(1)).OR.(IBDN.EQ.IBDNR(2))) GO TO 1060
0065 107 IDV=IDEV2
0066 108 IF (IB.EQ.2) IDV=IDEV3
0067 109 MAN=0
0068 110 CALL WRL(TEXT(1,2),16,REAL,MAN,0,IDV)
0069 111 INT=1
0070 112 CALL WRL(TEXT(1,5),16,BR,INT,0,IDV)
0071 113 DO 150 K=1,20
0072 114 IDV=IDEV2
0073 115 IF (IB.EQ.2) IDV=IDEV3
0074 116 WRITE(IDEV2,1002) K
0075 117 IF (K.EQ.1) WRITE(IDEV2,1018)
0076 118 IF ((K.EQ.2).OR.(K.EQ.3)) WRITE(IDEV2,1019)
0077 119 INT=6
0078 120 CALL WRL(TEXT(1,3),16,REAL,INT,3,IDV)
0079 121 IF (BUF1(1).EQ.#2020) GO TO 850
0080 122 DO 115 J=1,30
0081 123 IF ((BUF1(1).EQ.MREG(J,1)).AND.(BUF1(2).EQ.MREG(J,2))) GO T
0 140
0082 124 CONTINUE
0083 125 DO 120 J=1,30
0084 126 IF (MREG(J,1).EQ.0) GO TO 140
0085 127 CONTINUE
0086 128 WRITE(IDEV2,1004)
0087 129 GO TO 2000
0088 130 IMR=J
0089 131 IF (IB.EQ.90) GO TO 3500
0090 132 IF (IB.GE.3) GO TO 850
0091 133 INT=1
0092 134 BOR=0.

```

```

0093 CALL WRL(TEXT(1,4),16,BOR,INT,0,IDV)
0094 IY=2
0095 IF(IBDN.EQ.IBDNR(1)) IY=1
0096 IF(IBDNR(1).EQ.0) IY=1
0097 IF(BOR.EQ.0.) BOR=-0.00001
0098 IBD(IY,K,1)=K
0099 IBD(IY,K,2)=$2020
0100 IBD(IY,K,3)=BUF1(1)
0101 IBD(IY,K,4)=BUF1(2)
0102 IBD(IY,K,5)=$2020
0103 GR(IY)=BR
0104 BD(IY,K,1)=BOR
0105 IBDNR(IY)=IBDN
0106 BUF1(2)=$2020
0107 IA(IY)=MAN
0108 CONTINUE
0109 GO TO 850
0110
0111 C
0112 C
0113 C
0114 200 WRITE(IDEV2,1005)
0115 210 IBDN=0
0116 CALL WRL(TEXT(1,1),16,REAL,IBDN,0,IDEV2)
0117 IF((IBDN.EQ.IBDNR(1)).OR.(IBDN.EQ.IBDNR(2))) GO TO 220
0118 GO TO 1080
0119 KP(1)=0
0120 IF(IB.EQ.3) GO TO 610
0121 CALL WRL(TEXT(1,6),16,REAL,KP(1),0,IDEV2)
0122 IF(IB.GE.12) GO TO 225
0123 INT=5
0124 CALL WRL(TEX2(1),18,REAL,INT,1,IDEV2)
0125 IF(IB.EQ.12) GO TO 226
0126 INT=6
0127 CALL WRL(TEXT(1,3),16,REAL,INT,3,IDEV2)
0128 IF(IB.EQ.11) GO TO 227
0129 INT=1
0130 CALL WRL(TEXT(1,4),16,BOR,INT,0,IDEV2)
0131 BUF(4)=$2020
0132 IF(IB.EQ.12) GO TO 310
0133 GO TO 110
0134 IY=1
0135 IF(IBDN.EQ.IBDNR(2)) IY=2
0136 DO 250 K=1,20
0137 IK=21-K
0138 IF(K.EQ.1) GO TO 247
0139 DO 245 J=1,15
0140 IRD(IY,IK+1,J)=IRD(IY,IK,J)

```

150
C
C
C
200
210
220
225
226
227
240
245

RUTIN FÖR ÄNDRING AV MATERIAL-KOD DEL AV DENNA
RUTIN UTNYTTJAS AV ANDRA RUTINER

GO TO 220

```

0141 DO 246 J=1,4
0142 BD(IY,IK+1,J)=BD(IY,IK,J)
0143 IF((IBD(IY,IK,1).NE.0).AND.(K.EQ.1)) GO TO 1070
0144 IF(IBD(IY,IK,1).EQ.KP(1)) GO TO 260
0145 CONTINUE
0146 MREG(IMR,1)=BUF1(1)
0147 MREG(IMR,2)=BUF1(2)
0148 DO 265 L=1,4
0149 BD(IY,IK+1,L)=0.
0150 IBD(IY,IK+1,1)=KP(1)
0151 IBD(IY,IK+1,2)=BUF(1)
0152 IBD(IY,IK+1,3)=BUF1(1)
0153 IBD(IY,IK+1,4)=BUF1(2)+#000A
0154 IBD(IY,IK+1,13)=IMR
0155 IBD(IY,IK+1,15)=3
0156 IBD(IY,IK,15)=2
0157 IF((IY.EQ.NR).AND.(LP(IY).GE.KP(1))) LP(IY)=LP(IY)+1
0158 BD(IY,IK+1,1)=BD(IY,IK,1)
0159 BD(IY,IK+1,2)=BD(IY,IK,2)
0160 GO TO 2000
0161 C
0162 C
0163 C
0164 300 WRITE(IDEV2,1006)OE
0165 GO TO 210
0166 310 IY=1
0167 IF(IBDN.EQ.IBDNR(2)) IY=2
0168 DO 320 L=1,20
0169 K=21-L
0170 IF(IBD(IY,K,1).EQ.KP(1)) GO TO 330
0171 CONTINUE
0172 330 KP(2)=K
0173 GO TO 850
0174 IF(BOR.EQ.0.) BOR=-0.00001
0175 BD(IY,K,1)=BOR
0176 IBD(IY,K,5)=$2A20
0177 GO TO 2000
0178 C
0179 C
0180 C
0181 400 WRITE(IDEV2,1007)
0182 GO TO 210
0183 410 IY=1
0184 IF(IBDN.EQ.IBDNR(2)) IY=2
0185 DO 415 IK=1,20
0186 IF(BD(IY,IK,1).EQ.0.) GO TO 420
0187 CONTINUE
0188 GO TO 1070

```

```

0189 420 IBD(IY,IK,1)=KP(1)
0190 IBD(IY,IK,2)=$2A20
0191 IBD(IY,IK,3)=BUF1(1)
0192 IBD(IY,IK,4)=BUF1(2)
0193 IBD(IY,IK,5)=$2020
0194 IBD(IY,IK,13)=IMR
0195 IBD(IY,IK,15)=3
0196 BD(IY,IK,1)=BOR
0197 BD(IY,IK,2)=BOR
0198 MREG(IMR,1)=BUF1(1)
0199 MREG(IMR,2)=BUF1(2)
0200 GO TO 2000
0201 C
0202 C
0203 C
0204 500 WRITE(IDEV2,1008)
0205 GO TO 101
0206 C
0207 C
0208 C
0209 600 WRITE(IDEV2,1009)
0210 GO TO 210
0211 610 IY=1
0212 IF(IBDN.EQ.IBDNR(2)) IY=2
0213 WRITE(IDEV4,3000)IA(IY)
0214 WRITE(IDEV4,3002)GR(IY)
0215 DO 620 K=1,20
0216 IF(IBD(IY,K,15).EQ.1) GO TO 620
0217 BAR=BD(IY,K,1)
0218 IF(BD(IY,K,1).LT.0.) BAR=0.
0219 ISORT=ISHFT(IBD(IY,K,3),-8)
0220 ISI=ISHFT(IBD(IY,K,3),8)
0221 ISOR=IBD(IY,K,4).AND.$FFF0
0222 ISOR=ISHFT(ISOR,-8)
0223 ISI=ISI+ISOR
0224 WRITE(IDEV4,2999)ISORT,ISI
0225 WRITE(IDEV4,3002)BAR
0226 IF(BD(IY,K,1).EQ.0.) GO TO 2000
0227 CONTINUE
0228 GO TO 2000
0229 C
0230 C
0231 C
0232 700 WRITE(IDEV2,1010)
0233 710 IF(NR.GE.1) GO TO 1100
0234 IBDN=0
0235 CALL WRL(TEXT(1,1),16,REAL,IBDN,0,IDEV2)
0236 IF(IBDN.EQ.IBDNR(1)) GO TO 730

```

RUTIN FÖR INLÄSNING AV MATERIALLISTA VIA REMSA

RUTIN FÖR UTSTANSNING AV MATERIALLISTA PÅ REMSA

RUTIN FÖR START AV BÄDDNING

```

0237 IF(IBDN.NE.IBDNR(2)) GO TO 1080
0238 CALL START
0239 GO TO 2000
0240 C
0241 C
0242 C
0243 70
0244 WRITE(IDEV2,1011)
0245 IBDN=0
0246 CALL WRL(TEXT(1,1),16,REAL,IBDN,0,IDEV2)
0247 IF(NR.EQ.0) GO TO 2000
0248 IF(IBDN.NE.IBDNR(NR)) GO TO 2000
0249 ISK=3
0250 CALL BCLR(MANR,0)
0251 CALL STOPP
0252 GO TO 2000
0253 950
0254 CALL ATTCHW(IDEV1,M)
0255 GO TO(1500,1500,2000,2000,2000,1600,1658,1650
*,80,90,1510,1520,1530,2000,2000,2000,2000),IB
0256 C
0257 C
0258 C
0259 C
0260 C
0261 C
0262 C
0263 1500
0264 CALL SKRIV(3,1)
0265 GO TO 1700
0266 1505
0267 CALL SKRIV(3,2)
0268 GO TO 1700
0269 C
0270 C
0271 1510
0272 CALL SKRIV(5,0)
0273 GO TO 1700
0274 C
0275 C
0276 1520
0277 CALL SKRIV(6,1)
0278 GO TO 1700
0279 C
0280 C
0281 1530
0282 CALL SKRIV(7,0)
0283 GO TO 1700
0284 C
0285 C
0286 MIJ T. UTSKR. ASS

```

```

0285 C
0286 1600 CALL SKRIV(1,1)
0287 GO TO 2000
0288 C
0289 C   Rutin för utskrift av bladverkets läge
0290 C
0291 80 IF(NR.EQ.0) GO TO 2000
0292 IBL0=BD(NR,LP(NR),3)
0293 IBLA=BD(NR,LP(NR),4)
0294 WRITE(IDEV2,3012)MT(NR)
0295 WRITE(IDEV2,3013)IBD(NR,LP(NR),3),IBD(NR,LP(NR),4)
0296 WRITE(IDEV2,3014)OE,IBLO
0297 WRITE(IDEV2,3015)IBLA
0298 GO TO 2000
0299 C
0300 C   Rutin för utskrift av de konstanter vi kan ändra
0301 C   på i systemet
0302 C
0303 90 WRITE(IDEV2,3016)TARA
0304 WRITE(IDEV2,3017)SKL
0305 WRITE(IDEV2,3018)VL
0306 GO TO 2000
0307 C
0308 C   Rutin för att skriva ut vilka bäddar som är inlästa
0309 C   i datorn, samt möjlighet att nollställa dessa.
0310 C   Nollställningen av valt bäddregister börjar på
0311 C   label 1614
0312 C
0313 1610 WRITE(IDEV2,3010)IBDNR(1),IBDNR(2)
0314 WRITE(IDEV2,3020)
0315 IY=0
0316 CALL WRL(TEX6(1),10,REAL,IY,0,IDEV2)
0317 IF(IY.EQ.0) GO TO 2000
0318 IF(IY.GE.3) GO TO 2000
0319 IF(IY.EQ.NR) GO TO 4000
0320 IF(IB.GE.3) GO TO 1614
0321 C
0322 C   DENNA BIT FRAM TILL LABEL 1614 ANVÄDS FÖR ATT TIL-
0323 C   LÄTA ÄNDRINGAR I MATRIALLISTAN DÅ DENNA LÄSTS IN VIA TTY
0324 C   ELLER REMSA
0325 C
0326 3400 JA=7
0327 CALL WRL(TEX3(1),16,REAL,JA,2,IDEV2)
0328 IF(JA.NE.$4A41) GO TO 1614
0329 KP(1)=0
0330 CALL WRL(TEXT(1,6),16,REAL,KP(1),0,IDEV2)
0331 INT=6
0332 CALL WRL(TEXT(1,3),16,REAL,INT,3,IDEV2)
0333

```



```

0333 INT=1
0334 CALL WRL(TEXT(1,4),16,BOR,INT,D,IDEV2)
0335 IF(IB.NE.90) IRG=IB
0336 IB=90
0337 GO TO 110
0338 DO 3501 I=1,20
0339 IF(KP(1).EQ.I) GO TO 3502
0340 CONTINUE
0341 GO TO 2000
0342 IRD(IY,I,3)=BUF1(1)
0343 IRD(IY,I,4)=BUF1(2)
0344 IF(BOR.EQ.0.) BOR=-0.00001
0345 RD(IY,I,1)=BOR
0346 GO TO 1500
0347 C
0348 C NOLLSTÄLLNING AV BÄDDREGISTER
0349 C
0350 1614 IBDNR(IY)=0
0351 MT(IY)=0
0352 LP(IY)=0
0353 IA(IY)=0
0354 DO 1613 I=1,20
0355 DO 1611 K=1,15
0356 IBD(IY,I,K)=0
0357 DO 1612 J=1,4
0358 RD(IY,I,J)=0.
0359 CONTINUE
0360 IF(IB.EQ.1) GO TO 100
0361 IF(IB.EQ.5) GO TO 500
0362 GO TO 2000
0363 C
0364 C ÄNDRING AV TARERINGSGRÄNS FÖR VÅG
0365 C
0366 1620 INT=1
0367 CALL WRL(TEX8(1),20,TARA,INT,D,IDEV2)
0368 GO TO 2000
0369 C
0370 C ÄNDRING AV VIKT SOM FRONTLASTAREN KAN TA I SIN SKOPA
0371 C
0372 1630 INT=1
0373 CALL WRL(TEX7(1),18,SKL,INT,D,IDEV2)
0374 GO TO 2000
0375 C
0376 C ÄNDRING AV VIKT FÖR JÄRNVÄGSSVAGNAR
0377 C
0378 1640 INT=1
0379 CALL WRL(TEX5(1),18,VL,INT,D,IDEV2)
0380 GO TO 2000

```

```

0381 C
0382 C   RUTIN FÖR UTSKRIFT AV FÖRBRUKNINGEN AV ETT ELLER
0383 C   ALLA MATERIAL
0384 C
0385 1650 WRITE(IDEV2,3001)OE
0386   INT=5
0387   CALL WRL(TEXT(1,3),16,REAL,INT,3,IDEV2)
0388   DO 1655 L=1,30
0389   IF((BUF(1).EQ.MREG(L,1)).AND.(BUF(2).EQ.MREG(L,2))) GO TO
1656
0390   CONTINUE
0391   WRITE(IDEV2,1003)
0392   GO TO 2000
0393   CALL SKRIV(8,4)
0394   WRITE(IDEV2,1016)MREG(L,1),MREG(L,2),REG(L,1),REG(L,2)
0395   GO TO 2000
0396   CALL SKRIV(8,3)
0397   GO TO 2000
0398 C
0399 C   RUTIN FÖR ATT STÄLLA DATORNS KLOCKA
0400 C
0401 1670 CALL KLOCKA
0402   GO TO 2000
0403 C
0404 C   RUTIN FÖR FRÅGAN "ALLT OK ?"
0405 C
0406 1700 WRITE(IDEV2,1705)
0407   FORMAT(/)
0408   JA=7
0409   CALL WRL(TEX11,8,REAL,JA,2,IDEV2)
0410   IF(JA.EQ.#4A41) GO TO 1720
0411   CALL DETACH(IDEV1,M)
0412   IF(IB.EQ.90) GO TO 3400
0413   GO TO(3400,3400,2000,2000,2000,2000,2000,2000,2000,2000)
0414   *,80,90,200,300,400,2000,2000,2000,2000,2000),IB
0415   IF(IB.EQ.90) IB=IBG
0416   GO TO(2000,2000,2000,2000,2000,2000,2000,2000,2000,2000)
0417   *,80,90,240,335,410,2000,2000,2000,2000,2000),IB
0418   FORMAT(' INMÄTNING AV MATERIALLISTA VIA TANGENTBORD')
0419   FORMAT(' POST NR',IS)
0420   FORMAT(' FINNS EJ I MATERIALLISTAN')
0421   FORMAT(' MAX 30 MATERIAL PER MÅNAD')
0422   FORMAT(' ÄNDRING AV MATERIALKOD')
0423   FORMAT(' ÄNDRING AV B',A1,'RVIKT')
0424   FORMAT(' TILLFOGANDE AV POST')
0425   FORMAT(' INMÄTNING AV MATERIALLISTA VIA REMSA')
0426   FORMAT(' UTSTANSNING AV MATERIALLISTA')
0427   FORMAT(' START AV RÄDDNING')

```

```

0428 1011 FORMAT(' STOPP AV PÅGÅENDE BÄDDNING')
0429 1012 FORMAT(' INGET BÄDDPROGRAM LEDIGT')
0430 1013 FORMAT(' BÄDD REDAN INLÄST')
0431 1014 FORMAT(' MAX 20 POSTER I EN BÄDD')
0432 1015 FORMAT(' BÄDD EJ INLÄST')
0433 1016 FORMAT('1H ,2X,2A2,7X,F7.2,11X,F7.2)
0434 1017 FORMAT(' START GÅR EJ BÄDD NUMMER',I4,' AKTIV')
0435 1018 FORMAT(' GRUNDMATERIAL')
0436 1019 FORMAT(' DIREKTMATERIAL')
0437 2999 FORMAT(2A2)
0438 3000 FORMAT(I3)
0439 3001 FORMAT(' F',A1,'RERKNINGSUTSKRIFT ETT MATERIAL')
0440 3002 FORMAT(F7.2)
0441 3010 FORMAT(' REG1 = BÄDDNR',I4,' , REG2 = BÄDDNR',I4)
0442 3011 FORMAT(' DIALOG SLUT')
0443 3012 FORMAT(' MULTIPEL NR',I3)
0444 3013 FORMAT(' SORT = ',Z,A2)
0445 3014 FORMAT(' ,A1,'NSKAS = ',I4,' TON')
0446 3015 FORMAT(' ,I4,' TON')
0447 3016 FORMAT(' TÄRERINGSGRÄNS VÄG = ',F4.2,' TON')
0448 3017 FORMAT(' SKOPLASTVIKT = ',F5.2,' TON')
0449 3018 FORMAT(' VAGNSLASTVIKT = ',F5.2,' TON')
0450 3019 FORMAT(' ,/')
0451 3020 FORMAT(' NOLLSTÄLLNING AV BÄDDREGISTER')
0452 3050 FORMAT(' GÅR EJ, REGISTER I DRIFT')
0453 C
0454 C
0455 C
0456 C
0457 1050 WRITE(IDEV2,1012)
0458 GO TO 2000
0459 1060 WRITE(IDEV2,1013)
0460 GO TO 2000
0461 1070 WRITE(IDEV2,1014)
0462 GO TO 2000
0463 1080 WRITE(IDEV2,1015)
0464 GO TO 2000
0465 1100 IF((NR.EQ.1).OR.(NR.EQ.2)) WRITE(IDEV2,1017)IBDNR(NR)
0466 GO TO 2000
0467 4000 WRITE(IDEV2,3050)
0468 2000 CONTINUE
0469 CALL DETACH(IDEV2,M)
0470 CALL DETACH(IDEV1,M)
0471 WRITE(IDEV2,3011)
0472 STOP
0473 END
:E08
:F0F

```

```

0000 IDENT MRED
0001 C
0002 C
0003 C DENNA RUTIN ANVÄNDS BARA DÅ SYSTEMET STARTAS EFTER
0004 C ATT HA LADATS IN VIA REMSLÄSAREN. RUTINENS UPPGIFT
0005 C ÄR ATT SÄTTA IN DE KONSTANTER SOM FINNS SAMT
0006 C ADRESSERNA TILL DE YTTRE ENHETER SOM SYSTEMET BEHÖVER.
0007 C VISSA REGISTER I COMMON-AREAN NOLLSTÄLLS ÄVEN.
0008 C
0009 C
0010 COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0011 * ,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0012 * ,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0013 * ,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0014 INTEGER BUF,BUF1
0015 INTEGER ECB(2)
0016 IECB=IADDR(ECB)
0017 C
0018 C NOLLSTÄLLNING AV REG
0019 C
0020 MANR=0
0021 MANR1=0
0022 NR=0
0023 ISK=0
0024 IS=0
0025 C
0026 C ADRESSER TILL YTTRE ENHETER
0027 C
0028 IDEV1=6
0029 IDEV2=5
0030 IDEV3=10
0031 IDEV4=9
0032 C
0033 C KONSTANTER
0034 C
0035 SKL=15
0036 VL=45
0037 TARA=2
0038 C
0039 C NOLLSTÄLLNING AV MÅNADS- OCH DYGNREGISTEREN
0040 C
0041 DO 10 L=1,30
0042 MREG(L,1)=0
0043 MREG(L,2)=0
0044 MREG(L,3)=0
0045 RFG(L,1)=0.

```

0046
0047
0048
0049
:E09
:E0F

10

REG(L,2)=0
CALL ACTIV('STARUP',IECB,D,M)
STOP
END

S:

```

0000 IDENT NOLL
0001 SUBROUTINE NOLL(IR)
0002 C
0003 C
0004 C
0005 C DENNA RUTIN NOLLSTÄLLER BEORDRAD RAD PÅ
0006 C INDIKATIONSTABLÄRNAS BLADVERK SÅ ATT
0007 C DESSA HAMNAR I ETT SYNKRONISERINGSLÄGE.
0008 C
0009 C RUTINEN ANROPAS CALL NOLL(IR)
0010 C IR=1-4 ANGER VILKEN RAD SOM SKALL
0011 C NOLLSTÄLLAS
0012 C IR=5 ANGER ATT ALLA RADER SKALL
0013 C NOLLSTÄLLAS
0014 C
0015 C INTEGER NO(5)
0016 C DATA (NO(I),I=1,5) /#C002,#3802,#0782,#007A,#FFFA/
0017 C
0018 C L=0
0019 C M=0
0020 C N=16
0021 C
0022 C FINNS BOKSTAV I RAD ?
0023 C
0024 C IF(IR.EQ.5) N=30
0025 C IF(IR.EQ.2) N=30
0026 C
0027 C GE UT STEGPULSER L=0 POSIV L=1 NEGATIV
0028 C
0029 C DO 10 K=1,N
0030 C IF(L.EQ.2) L=0
0031 C IF(L.EQ.0) IU=NO(IR)
0032 C CALL UT(IU)
0033 C CALL UT(5)
0034 C L=L+1
0035 C CALL BCLR(IU,1)
0036 C CALL BSET(IU,D)
0037 C RETURN
0038 C END
:EOS
:EOF

```

```

0000 IDENT RA
0001 SUBROUTINE RA
0002 C
0003 C
0004 C DENNA RUTIN ANVÄNDS FÖR ATT STÄLLA UT INDIKATIONSTAB-
0005 C LÅERNA VID START AV BÄDD, MULTIPELBYTE OCH MATERIALBYTE.
0006 C
0007 C
0008 C COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0009 C *,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IRDN
0010 C *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0011 C *,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0012 C INTEGER BUF,BUF1
0013 C
0014 C SLÄCK ALLA LAMPOR OCH TÄND RÖD LAMPA.
0015 C
0016 C CALL LAMPA(7)
0017 C CALL LAMPA(4)
0018 C
0019 C SKALL MULTIPELNUMRET ÄNDRAS ?
0020 C
0021 C IF((IS.EQ.3) .AND. MTGM=0)
0022 C IS=0
0023 C IF(MTGM.NE.MT(NR)) CALL BL(1,MT(NR),1)
0024 C MTGM=MT(NR)
0025 C
0026 C STÄLL UT HÖGNUMRET.
0027 C
0028 C CALL BL(2,IBD(NR,LP(NR),3),IBD(NR,LP(NR),4))
0029 C
0030 C STÄLL UT BÖR- OCH ÄRVIKT
0031 C
0032 C IB=BD(NR,LP(NR),3)
0033 C CALL BL(3,IB,1)
0034 C IB=BD(NR,LP(NR),4)
0035 C CALL BL(4,IB,1)
0036 C
0037 C TÄND RÄTT SKOPLAMPA I RELATION TILL MATERIALBEHOVET
0038 C
0039 C IF((BD(NR,LP(NR),3)-BD(NR,LP(NR),4)).GE.(0.875*SKL)) GO TO 10
0040 C
0041 C IF((BD(NR,LP(NR),3)-BD(NR,LP(NR),4)).GE.(0.375*SKL)) GO TO 15
0042 C
0043 C IF((IBD(NR,LP(NR),14).EQ.1).OR.(MT(NR).EQ.IA(NR))) GO TO 2
0044 C GO TO 30

```

```

0043 10 CALL LAMPA(1)
0044 GO TO 30
0045 15 CALL LAMPA(2)
0046 GO TO 30
0047 20 IF((BD(NR,LP(NR),3)-BD(NR,LP(NR),4)).LT.(0.125*SKL)) GO TO
      30
0049 CALL LAMPA(3)
0049 GO TO 30
0050 CONTINUE
0051 RETURN
0052 END
:EOS
:EOF

```

S:


```

0000 IDENT RAKNA
0001 SUBROUTINE RAKNA(RN)
0002 C
0003 C
0004 C DENNA RUTIN REGISTRERAR ALLA INKOMMANDE VIKTER OCH
0005 C UPDATERAR MÅNADS- OCH DYGNREGISTREN. NÄR ETT MATE-
0006 C RIAL ÄR KLART INOM EN MULTIPEL BEORDRAS NÄSTA MATE-
0007 C RIAL DIREKT. NÄR EN MULTIPEL ÄR KLAR SÅ KONTROLERAS
0008 C DEN OCH ALLA FÖRÄNDRINGAR REGISTRERAS INOM NÄSTA
0009 C BEORDRAS. EFTER VARJE MULTIPELBYTE BEORDRAS ÄVEN EN
0010 C MULTIPEL-UTSKRIFT
0011 C
0012 C RUTINEN ANROPAS CALL RAKNA(RN)
0013 C RN=REGISTRERAD VIKT AV VÄGEN.
0014 C
0015 C
0016 C COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0017 C *,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0018 C *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0019 C *,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0020 C INTEGER BUF,BUF1,ECB(2)
0021 C IECB=IADDR(ECB)
0022 C
0023 C BARA DIREKTMATERIAL KVAR ?
0024 C
0025 C IF (IS.EQ.1) GO TO 90
0026 C
0027 C TRYCKKNAPP NÄSTA MULTIPEL ANVÄND ?
0028 C
0029 C IF (LP(NR).EQ.30) GO TO 50
0030 C
0031 C MER ÄN 20 MATERIAL NY MULTIPEL
0032 C
0033 C IF (LP(NR).GE.21) GO TO 40
0034 C
0035 C KUNDE MATERIAL LOSSAS
0036 C
0037 C IF (BD(NR,LP(NR),1).LT.(0.125*SKL)) GO TO 30
0038 C IF (BD(NR,LP(NR),1).LE.0.) GO TO 2
0039 C
0040 C NOTERA VIKTFÖRÄNDRINGEN I BÄDDREGISTRET ÄNDRA
0041 C ÄRVIKT PÅ INDIKATIONSTABLÅN.
0042 C
0043 C BD(NR,LP(NR),4)=BD(NR,LP(NR),4)+RN
0044 C BD(NR,LP(NR),2)=BD(NR,LP(NR),2)-RN
0045 C ID=BD(NR,LP(NR),4)

```

0046 CALL BL(4, ID, D)
0047 C
0048 C RÄKNA UPP MÅNADS- OCH DYGNREGISTREN.
0049 C
0050 REG(IBD(NR, LP(NR), 13), 1) = REG(IBD(NR, LP(NR), 13), 1) + RN
0051 REG(IBD(NR, LP(NR), 13), 2) = REG(IBD(NR, LP(NR), 13), 2) + RN
0052 C
0053 C TÄND RÄTT SKOPLAMPA I RELATION TILL MATERIALBEHOVET
0054 C
0055 2 IF((BD(NR, LP(NR), 3) - BD(NR, LP(NR), 4)) .GE. (D.875*SKL)) GO TO 10
0056 15 IF((BD(NR, LP(NR), 3) - BD(NR, LP(NR), 4)) .GE. (D.375*SKL)) GO TO 15
0057 0 IF((IBD(NR, LP(NR), 14) .EQ. 1) .OR. (MT(NR) .EQ. IA(NR))) GO TO 2
0058 10 GO TO 30
0059 10 CALL LAMPA(1)
0060 GO TO 200
0061 15 CALL LAMPA(2)
0062 GO TO 200
0063 20 IF((BD(NR, LP(NR), 3) - BD(NR, LP(NR), 4)) .LT. (D.125*SKL)) GO TO 30
0064 30
0064 CALL LAMPA(3)
0065 GO TO 200
0066 C
0067 C NÄSTA MATERIAL
0068 C
0069 30 IF(MT(NR) .EQ. 1) LP(NR) = 20
0070 LP(NR) = LP(NR) + 1
0071 C
0072 C NY MULTIPLEL ?
0073 C
0074 IF(LP(NR) .GE. 21) GO TO 40
0075 IF(BD(NR, LP(NR), 1) .EQ. 0.) GO TO 40
0076 GO TO 100
0077 C
0078 C HAR NGT MATERIAL BLIVIT AVBRUTET IP=0 NEJ
0079 C
0080 40 CALL KONTR(IP)
0081 IF(IP .EQ. 0) GO TO 50
0082 LP(NR) = IP
0083 GO TO 100
0084 C
0085 C MULTIPLELBYTE
0086 C
0087 50 IF(MT(NR) .GE. (IA(NR) + 1)) MT(NR) = IA(NR)
0088 MT(NR) = MT(NR) + 1
0089 C

HAR ÄNDRING AV HÖGNUMMER GJORTS ?

0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C
0111 C
0112 C
0113 C
0114 C
0115 C
0116 C
0117 C
0118 C
0119 C
0120 C
0121 C
0122 C
0123 C
0124 C
0125 C
0126 C
0127 C
0128 C
0129 C
0130 C
0131 C
0132 C
0133 C
0134 C
0135 C
0136 C
0137 C

```

DO 55 K=1,20
IF (IBD(NR,K,15).NE.2) GO TO 54
BD(NR,K,1)=BD(NR,K,4)
BD(NR,K,3)=BD(NR,K,4)
BD(NR,(K+1),1)=BD(NR,K,2)
BD(NR,(K+1),2)=BD(NR,K,2)
BD(NR,K,2)=0.
IF (BD(NR,K,1).LE.0.) BD(NR,K,1)=-0.00001
CONTINUE
54
55 IF (IBD(NR,K,11).EQ.#4252) IBD(NR,K,10)=$4156

```

SKRIV MULTIPEL-LISTA.

```

CALL ATTCHW(IDEV1,M)
CALL SKRIV(1,D)
CALL DETAGH(IDEV1,M)
LP(NR)=1

```

HAR ÄNDRING AV BÖRVIKT GJORTS ?

```

DO 60 K=1,20
IF (IBD(NR,K,2).EQ.#2A20) IBD(NR,K,2)=$2020
IF (IBD(NR,K,15).EQ.3) IBD(NR,K,15)=0
IF (IBD(NR,K,15).EQ.2) IBD(NR,K,15)=1
IF (IBD(NR,K,5).EQ.#2020) GO TO 59
BD(NR,K,2)=BD(NR,K,1)-BD(NR,K,4)
IBD(NR,K,6)=(BD(NR,K,1)/VL)+0.5)
IBD(NR,K,7)=IBD(NR,K,6)-IBD(NR,K,9)
IF (IBD(NR,K,8).GE.IBD(NR,K,6)) IBD(NR,K,8)=IBD(NR,K,9)
CONTINUE
59
IF (IBD(NR,K,7).LE.0) IBD(NR,K,7)=0
IBD(NR,K,2)=IAND(IBD(NR,K,2),#FFFF)

```

ÅTERSTÄLL MEDDELANDE REGISTREN

```

IBD(NR,K,4)=IAND(IBD(NR,K,4),#FFFF)
IBD(NR,K,10)=0
IBD(NR,K,11)=0
IBD(NR,K,14)=0
IBD(NR,K,5)=$2020

```

SISTA MULTIPEL ?

```

IF (MT(NR).GE.(IA(NR)+1)) GO TO 90
GO TO 100

```

```

0138      C
0139      C
0140      90
0141      DO 92 KL=1,20
0142      IF (IBD(NR, KL, 1).EQ.2) GO TO 95
0143      CONTINUE
0144      C
0145      C
0146      95
0147      IF (IBD(NR, KL, 7).NE.0) GO TO 150
0148      IF (IBD(NR, (KL+1), 7).NE.0) GO TO 150
0149      MT(NR)=IA(NR)+1
0150      CALL ACTIV('KLAR ', IECB, D, M)
0151      GO TO 200
0152      CALL ACTIV('FORD ', IECB, D, M)
0153      GO TO 200
0154      IS=1
0155      MT(NR)=IA(NR)+1
0156      CONTINUE
0157      RETURN
          :E05
          :EOF

```

S:

```

0000 IDENT RED
0001 SUBROUTINE RED
0002 C
0003 C
0004 C
0005 C
0006 C
0007 C
0008 C
0009 C
0010 C
0011 C
0012 C
0013 C
0014 C
0015 C
0016 C
0017 C
0018 C
0019 C
0020 C
0021 C
0022 C
0023 C
0024 C
0025 C
0026 C
0027 10
0028 5
0029 C
0030 C
0031 C
0032 C
:EOS
:EOF

```

DENNA RUTIN ANVÄNDS FÖR ATT RÄKNA OM BÖRVÄRDET FÖR DIREKTMATERIALEN TILL JÄRNVÄGSMATERIALER ENLIGT KONSTANTEN, REGISRET SOM INNEHÅLLER ÅTERSTÅENDE VIKT AV ETT MATERIAL SÄTTIS TILL BÖRVÄRDET. DETTA GÖRS BARA FÖRSTA GÅNGEN EN BÄDD STARTAS.

```

COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
*,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
*,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
*,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
INTEGER BUF,BUF1

BÄDD STARTAD TIDIGARE ?

IF(MT(NR).NE.0) GO TO 5

RÄKNA FRAM JÄRNVÄGSMATERIALER SAMT SÄTT REG ÅTERSTÅR.

DO 10 K=1,20
BD(NR,K,2)=BD(NR,K,1)
IBD(NR,K,6)=((BD(NR,K,1)/VL)+0.5)
BD(NR,K,2)=BD(NR,K,1)-BD(NR,K,3)
IBD(NR,K,7)=IBD(NR,K,6)-IBD(NR,K,8)
CONTINUE
IF(MT(NR).EQ.0) LP(NR)=1
IF(MT(NR).EQ.0) MT(NR)=1
RETURN
END

```

S:

```

0000 IDENT SKRIV
0001 SUBROUTINE SKRIV(IUT,IN)
0002
0003
0004 DENNA RUTIN SKÖTER OM ALLA DE UTSKRIFTER SOM ÄR
0005 FÖRKIPPADE MED EN BÄDD. RUTINEN GÖR ÄVEN FÖRBRUK-
0006 NINGSUTSKRIFTERNA SAMNT NOLLSTÄLLNINGEN AV DESSA
0007 REGISTER.
0008
0009 RUTINEN ANROPAS CALL SKRIV(IUT,IN)
0010 IUT=VILKEN UTSKRIFT SOM ÖNSKAS
0011 IN=VILKEN ENHET SOM UTSKRIFTEN SKALL
0012 GÖRAS PÅ OCH EV. TILLHÖRANDE SPECIALFUNK.
0013
0014
0015 COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0016 *,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0017 *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0018 *,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0019 INTEGER RU1(12),RU2(14),RU3(14),RU4(14),RU5(14),
0020 *RU6(14),RU7(3),RU8(4),RU9(6),RU10(7),RU11(4),RU12(6),
0021 *RU13(4),RU14(2),RU15(1),RU16(7),RU17(7),RU18(7),
0022 *RU19(6),RU20(11),RU21(9),RU22(14),RU23(10),RU24(6),
0023 *RU25(7),RU26(2),RU27(3),FF(1),BUF,BUF1,IDA(6)
0024
0025 ALLA TABELLHUVUDEN I DATASATSER.
0026
0027 DATA (RU1(I),I=1,12) /2HDA,2HTU,2HM ,9*2H /
0028 DATA (RU2(I),I=1,14) /2HKL,2HOC,2HKS,2HLA,2HG ,9*2H /
0029 DATA (RU3(I),I=1,14) /2HBÄ,2HDD,2HMV,2HMM,2HER,9*2H /
0030 DATA (RU4(I),I=1,14) /2HAN,2HTA,2HL ,2HMV,2HLT,2HIP
0031 *,2HLA,2HR ,6*2H /
0032 DATA (RU5(I),I=1,14) /2HSE,2HNA,2HST,2H F,2HÄR,2HDI,2HGA
0033 *,2H M,2HVL,2HTI,2HPE,2HL ,2HNR,2H /
0034 DATA (RU6(I),I=1,14) /2HAN,2HTA,2HL ,2HÄT,2HER,2HST,2HÄE
0035 *,2HND,2HE ,2HMV,2HLT,2HIP,2HLA,2HR /
0036 DATA (RU7(I),I=1,3) /2HPO,2HST,2HNR/
0037 DATA (RU8(I),I=1,4) /2HMT,2HRL,$425C,2HG /
0038 DATA (RU9(I),I=1,6) /2HHE,2HLA,2H B,2HÄD,2HDE,2HN /
0039 DATA (RU10(I),I=1,7) /2HDE,2HNN,2HA ,2HMV,2HLT,2HIP,2HEL/
0040
0041 DATA (RU11(I),I=1,4) /$425C,2HRV,2HIK,2HT /
0042 DATA (RU12(I),I=1,6) /2HVA,2HGN,2HSL,2HAS,2HTE,2HR /
0043 DATA (RU13(I),I=1,4) /2HÄT,2HER,2HST,2HÄR/
0044 DATA (RU14(I),I=1,2) /$425C,2HR /
0045 DATA (RU15(I),I=1,1) /2HÄR/

```

```

0045 DATA (RV16(I),I=1,7) /2HMA,2HTE,2HRI,2HAL,2HLI,2HST,2HA /
0046 DATA (RV17(I),I=1,7) /2HBÄ,2HDD,2HNI,2HNG,2HSL,2HIS,2HTA /
0047 DATA (RV18(I),I=1,7) /2HMU,2HLI,2HIP,2HEL,2HLI,2HST,2HA /
0048 DATA (RV19(I),I=1,6) /2HSL,2HUT,2HUT,2HUT,2HUT,2HUT,2HUT,2HUT,2HUT /
0049 DATA (RV20(I),I=1,11) /2HÄN,2HDR,2HIN,2HG,2HAV
0050 *;2H M,2HAT,2HER,2HIA,2HLK,2HOD /
0051 DATA (RV21(I),I=1,9) /2HÄN,2HDR,2HIN,2HG,2HAV
0052 *;2H B,#5C52,2HVI,2HKT /
0053 DATA (RV22(I),I=1,14) /2HTI,2HLL,2HFO,2HGA,2HND
0054 *;2HE,2HAV,2H M,2HAT,2HER,2HIA,2HLP,2HOS,2HT /
0055 DATA (RV23(I),I=1,10) /2HMÄ,2HTE,2HRI,2HAL,#465C,2HRB
0056 *;2HRV,2HKN,2HIN,2HG /
0057 DATA (RV24(I),I=1,6) /2HUN,2HDE,2HR,2HDY,2HGN,2HET /
0058 DATA (RV25(I),I=1,7) /2HVUN,2HDE,2HR,2HMÄ,2HNA,2HDE,2HN /
0059 DATA (RV26(I),I=1,2) /2HTO,2HN /
0060 DATA (RV27(I),I=1,3) /2HVA,2HGN,2HSL /
0061 DATA (FF(I),I=1,1) /#200C /
0062 C
0063 C
0064 C
0065 C
0066 C
0067 C
0068 C
0069 C
0070 C
0071 C
0072 C
0073 C
0074 C
0075 C
0076 C
0077 C
0078 C
0079 C
0080 C
0081 C
0082 C
0083 C
0084 C
0085 C
0086 C
0087 C
0088 C

```

HÄMTA AKTUELLT KLOCKSLAG OCH DATUM

CALL DATE(IDA,M)

KAN UTSKRIFT UTFÖRAS ?

IF(IUT.EQ.3) GO TO 5

IF(IUT.EQ.8) GO TO 5

IF(NR.EQ.0) GO TO 200

TAG FRAM FÖR TABELLHUVUD NÖDVÄNDIGA UPPGIFTER.

IBNR=IBDNR(NR)

MNR=IA(NR)

MSNR=MT(NR)-1

MUR=MNR-MSNR

BEROENDE PÅ UTSKRIFT OCH VAL AV YTTRE ENHET
SÅ BEORDRAS FORM FEED ELLER RADFRAMMÄTNING.

IF((IUT.EQ.1).AND.(IN.EQ.1)) WRITE(IDEV2,1000)

IF((IUT.EQ.1).AND.(IN.EQ.1)) GO TO 5

IF(IUT.LE.4) WRITE(IDEV1,1000)FF

IF(IUT.GE.5) WRITE(IDEV2,1000)

IF(IUT.EQ.3) WRITE(IDEV2,1000)

IF((IUT.EQ.8).AND.(IN.GE.3)) WRITE(IDEV2,1000)

```

0089 IF((IUT.EQ.8).AND.(IN.LE.2)) WRITE(IDEV1,1000)FF
0090 FORMAT(/,A2)
0091 GO TO(10,20,30,40,50,60,70,80),IUT
0092 C
0093 C   TABELLHUVUD FÖR MULTIPELUTSKRIFT
0094 C
0095 10   IDV=IDEV1
0096 IF(IN.EQ.1) IDV=IDEV2
0097 WRITE(IDV,100)RU1,IDA(3),IDA(4),IDA(2),RU18
0098 WRITE(IDV,106)RU2,IDA(4),IDA(5),RU3,IBNR,RU4,MNR,RV5,
0099 *MSNR,RV6,MUR
0100 IF((IDEV1.EQ.5).OR.(IN.EQ.1)) GO TO 15
0101 WRITE(IDEV1,107)RU7,RU8,RU9,RU10
0102 WRITE(IDEV1,108)RU11,RU13,RU14,RU15
0103 WRITE(IDEV1,109)RU26,RU12,RU26,RU12,RU26,RU12
0104 GO TO 160
0105 15   WRITE(IDEV2,110)RU7,RU8,RU9,RU10
0106 WRITE(IDEV2,111)RU11,RU13,RU14,RU15
0107 WRITE(IDEV2,112)RU26,RU27,RU26,RU27,RU26,RU27
0108 GO TO 170
0109 C
0110 C   TABELLHUVUD FÖR SLUTUTSKRIFT
0111 C
0112 20   WRITE(IDEV1,101)RU1,IDA(3),IDA(1),IDA(2),RU19
0113 IDV=IDEV1
0114 GO TO 12
0115 C
0116 C   MATERIALLISTA VID INLÄSNING AV BÄDD
0117 C   MATERIAL VARS BÖRVIKT=0 SKRIVS EJ UT
0118 C
0119 30   WRITE(IDEV2,100)RU1,IDA(3),IDA(1),IDA(2),RU15
0120 WRITE(IDEV2,106)RU2,IDA(4),IDA(5),RU3,IBDN,RU4,MAN
0121 WRITE(IDEV2,125)GR(IN)
0122 WRITE(IDEV2,113)RU7,RU8,RU9
0123 WRITE(IDEV2,114)RU11
0124 WRITE(IDEV2,115)RU26
0125 IF((IN.NE.1).AND.(IN.NE.2)) GO TO 200
0126 DO 32 J=1,20
0127 IF(BD(IN,J,1).LT.0.) GO TO 32
0128 IF(BD(IN,J,1).EQ.0.) GO TO 33
0129 WRITE(IDEV2,154)IBD(IN,J,1),IBD(IN,J,2),IBD(IN,J,3)
0130 *,IBD(IN,J,4),BD(IN,J,1)
0131 CONTINUE
0132 32   GO TO 200
0133 C
0134 C   BÄDDNINGSLISTA VID START AV BÄDDNING
0135 C   MATERIAL VARS BÖRVIKT=0 SKRIVS EJ UT
0136 C

```



```

0137 40 WRITE(IDEV1,100)RU1,IDA(3),IDA(1),IDA(2),RU17
0138 WRITE(IDEV1,106)RU2,IDA(4),IDA(5),RV3,IBDN,RV4,MNR,RU5,
0139 *MSNR,RV6,MUR
0140 WRITE(IDEV1,113)RV7,RV8,RV9
0141 WRITE(IDEV1,116)RV11,RV13
0142 WRITE(IDEV1,117)RV26,RV12,RV26,RV12
0143 DO 42 J=1,20
0144 IF(BD(NR,J,1).LT.0.) GO TO 42
0145 IF(BD(NR,J,1).EQ.0.) GO TO 200
0146 IF((IBD(NR,J,1).EQ.2).OR.(IBD(NR,J,1).EQ.3)) WRITE(IDEV1
0147 *,156)IBD(NR,J,1),IBD(NR,J,2),IBD(NR,J,3),IBD(NR,J,4),BD(NR
,J,1)
0148 *,IBD(NR,J,5),IBD(NR,J,6),IBD(NR,J,7)
0149 IF((IBD(NR,J,1).EQ.2).OR.(IBD(NR,J,1).EQ.3)) GO TO 42
0150 WRITE(IDEV1,155)IBD(NR,J,1),IBD(NR,J,2),IBD(NR,J,3),IBD(NR
,J,4)
0151 *,BD(NR,J,1),IBD(NR,J,5),BD(NR,J,2)
0152 42 CONTINUE
0153 GO TO 200
0154 C
0155 C UTSKRIFT VID ÄNDRING AV MATERIALKOD
0156 C
0157 50 WRITE(IDEV2,102)RU1,IDA(3),IDA(1),IDA(2),RU20
0158 WRITE(IDEV2,106)RU2,IDA(4),IDA(5),RV3,IBDN
0159 WRITE(IDEV2,118)RV7,RV8
0160 WRITE(IDEV2,152)KP(1),BUF(1),BUF(1),BUF(1),BUF(1)(2)
0161 GO TO 200
0162 C
0163 C UTSKRIFT VID ÄNDRING AV BÖRVIKT
0164 C
0165 60 WRITE(IDEV2,103)RU1,IDA(3),IDA(1),IDA(2),RU21
0166 62 WRITE(IDEV2,106)RU2,IDA(4),IDA(5),RV3,IBDN
0167 WRITE(IDEV2,113)RV7,RV8,RV9
0168 WRITE(IDEV2,114)RV11
0169 WRITE(IDEV2,115)RV26
0170 IF(IN.EQ.1) GO TO 65
0171 WRITE(IDEV2,154)KP(1),BUF(4),BUF(1),BUF(1)(2),BOR
0172 GO TO 200
0173 65 IY=1
0174 IF(IBDN(2).EQ.IBDN) IY=2
0175 WRITE(IDEV2,154)KP(1),BUF(4),IBD(IY,KP(2),3)
0176 *,IBD(IY,KP(2),4),BOR
0177 GO TO 200
0178 C
0179 C UTSKRIFT VID TILLFÖGANDE AV MATERIALPOST
0180 C
0181 70 WRITE(IDEV2,104)RU1,IDA(3),IDA(1),IDA(2),RU22
0182 GO TO 62

```

UTSKRIFT AV DYGNS- OCH MÅNADSREGISTREN

IN=1 UTSKRIFT PÅ PER NOLLSTÄLLNING
 AV BÅDE DYGNS- OCH MÅNADSREGISTREN.
 IN=2 UTSKRIFT PÅ PER NOLLSTÄLLNING
 AV DYGNSREGISTREN
 IN=3 UTSKRIFT PÅ TTY INGEN NOLLSTÄLLNING
 IN=4 ENDAST TABELLHUVUD PÅ TTY

IDEV=IDEV1
 IF(IN.EQ.3) IDEV=IDEV2
 WRITE(IDEV,105)RU1,IDA(3),IDA(1),IDA(2),RU23
 WRITE(IDEV,128)RU2,IDA(4),IDA(5)
 WRITE(IDEV,119)RU6,RU24,RU25
 IF(IN.EQ.4) GO TO 200
 DO 99 J=1,30
 IF(MREG(J,1).EQ.0) GO TO 200
 WRITE(IDEV,153)MREG(J,1),MREG(J,2),REG(J,1),REG(J,2)
 IF(IN.EQ.3) GO TO 99
 REG(J,1)=0.
 IF(IN.EQ.1) GO TO 97
 GO TO 99

96

97

99

UTSKRIFT AV TABELL FÖR MULTIPEL- OCH
 SLUTUTSKRIFT PÅ PER. MATERIAL MED BÖR-
 VIKT=0 ÖVERHOPPAS.

DO 162 L=1,20
 IF(BD(NR,L,1).LT.0.) GO TO 162
 IF(BD(NR,L,1).EQ.0.) GO TO 200
 IF(IUT.EQ.2) WRITE(IDEV1,120)IBD(NR,L,1),IBD(NR,L,2)
 *,IBD(NR,L,3),IBD(NR,L,4),BD(NR,L,1),BD(NR,L,2),BD(NR,L,4)

160

IF(IUT.EQ.2) GO TO 162
 IF((IBD(NR,L,1).EQ.2).OR.(IBD(NR,L,1).EQ.3)) WRITE(
 *IDEV1,157)IBD(NR,L,1),IBD(NR,L,2),IBD(NR,L,3),IBD(NR
 *,L,4),BD(NR,L,1),IBD(NR,L,5),IBD(NR,L,6)
 *,IBD(NR,L,7),IBD(NR,L,8),IBD(NR,L,9)
 IF((IBD(NR,L,1).EQ.2).OR.(IBD(NR,L,1).EQ.3)) GO TO 162
 WRITE(IDEV1,151)IBD(NR,L,1),IBD(NR,L,2),IBD(NR,L,3),IBD(NR

0221

0222

0223

0224

0225

0226

0227

0228

0163

0164

0165

0166

0167

0168

0169

0190

0191

0192

0193

0194

0195

0196

0197

0198

0199

0200

0201

0202

0203

0204

0205

0206

0207

0208

0209

0210

0211

0212

0213

0214

0215

0216

0217

0218

0219

0220

```

0239 *,L,4),IBD(NR,L,10),IBD(NR,L,11)
0230 CONTINUE
0231 GO TO 200
0232 C
0233 UTSKRIFT AV TABELL FÖR MULTIPEL-OCH
0234 SLUTUTSKRIFT PÅ TTY. MATERIAL MED BÖR-
0235 VIKT=0 ÖVERHOPPAS.
0236 C
0237 DO 172 L=1,20
0238 IF(RD(NR,L,1),LT,0.) GO TO 172
0239 IF(RD(NR,L,1),EQ,0.) GO TO 200
0240 IF(IUT,EQ,2) WRITE(IDEV2,121)IBD(NR,L,1)
0241 *,IBD(NR,L,2),IBD(NR,L,3),IBD(NR,L,4),BD(NR,L,1)
0242 *,BD(NR,L,2),BD(NR,L,4)
0243 IF(IUT,EQ,2) GO TO 172
0244 IF((IBD(NR,L,1),EQ,2),OR,(IBD(NR,L,1),EQ,3)) WRITE(
0245 *IDEV2,158)IBD(NR,L,1),IBD(NR,L,2),IBD(NR,L,3),IBD(NR
0246 *,L,4),BD(NR,L,1),IBD(NR,L,5),IBD(NR,L,6),IBD(NR,L,7)
0247 *,IBD(NR,L,8),IBD(NR,L,9)
0248 IF((IBD(NR,L,1),EQ,2),OR,(IBD(NR,L,1),EQ,3)) GO TO 172
0249 WRITE(IDEV2,150)IBD(NR,L,1),IBD(NR,L,2),IBD(NR,L,3),IBD(NR
0250 *,L,4),BD(NR,L,1),IBD(NR,L,5),BD(NR,L,2),BD(NR,L,3),BD(NR
0251 *,L,4),IBD(NR,L,10),IBD(NR,L,11)
0252 CONTINUE
0253 GO TO 200
0254 FORMAT(1H ,12A2,1X,15,2I2,4X,7A2)
0255 FORMAT(1H ,12A2,1X,15,2I2,4X,6A2)
0256 FORMAT(1H ,12A2,1X,15,2I2,4X,11A2)
0257 FORMAT(1H ,12A2,1X,15,2I2,4X,9A2)
0258 FORMAT(1H ,12A2,1X,15,2I2,4X,14A2)
0259 FORMAT(1H ,12A2,1X,15,2I2,4X,10A2)
0260 FORMAT(1H ,14A2,13,1H,12/(1H ,14A2,I6))
0261 FORMAT(1HD,3A2,2H ,4A2,1X,6A2,12X,7A2)
0262 FORMAT(1H ,17X,4A2,13X,4A2,15X,2A2,17X,A2)
0263 FORMAT(1H ,17X,4(2A2,3X,6A2,2X))
0264 FORMAT(1HD,3A2,1X,4A2,6A2,12X,7A2)
0265 FORMAT(1H ,15X,4A2,4X,4A2,4X,2A2,8X,A2)
0266 FORMAT(1H ,14X,4(1X,2A2,1X,3A2))
0267 FORMAT(1HD,3A2,2X,4A2,1X,6A2)
0268 FORMAT(1H ,17X,4A2)
0269 FORMAT(1H ,17X,2A2)
0270 FORMAT(1H ,17X,4A2,13X,4A2)
0271 FORMAT(1H ,17X,2(2A2,3X,6A2,2X))
0272 FORMAT(1HD,3A2,2X,4A2)
0273 FORMAT(1HD,4A2,4X,6A2,5X,7A2)
0274 FORMAT(1H ,2X,12,A2,2X,2A2,1X,F7.2,5X
0275 *,F7.2,17X,F7.2)

```

```

0276 FORMAT(1H ,2X,I2,A2,3X,I2,A2,3X,2A2,2X,F7.2,14X
0277 * ,F7.2,35X,F7.2)
0278 FORMAT(' GRUNDSATS',18X,F7.2)
0279 FORMAT(1H ,14A2,I3,1H.,I2)
0280 FORMAT(1H ,2X,I2,A2,2X,2A2,1X,F7.2,A2,
0281 *3(3X,F7.2,2X),2A2)
0282 FORMAT(1H ,2X,I2,A2,3X,2A2,2X,F7.2,A2,
0283 *3(12X,F7.2,2X),2A2)
0284 FORMAT(1H ,2X,I2,A2,5X,2A2)
0285 FORMAT(1H ,2X,2A2,7X,F7.2,11X,F7.2)
0286 FORMAT(1H ,2X,I2,A2,3X,2A2,2X,F7.2)
0287 FORMAT(1H ,2X,I2,A2,3X,2A2,2X,F7.2,A2,12X,
0288 *F7.2)
0289 FORMAT(1H ,2X,I2,A2,3X,2A2,2X,F7.2,A2,4X,I2,19X,I2)
0290 FORMAT(1H ,2X,I2,A2,3X,2A2,2X,F7.2,A2,4X,I2,7X,
0291 *3(12X,I2,7X))
0292 FORMAT(1H ,2X,I2,A2,2X,2A2,1X,F7.2,A2,I2,1X,
0293 *3(9X,I2,1X))
0294 CONTINUE
0295 RETURN
0296 END
:EOS
:EOF

```

S:

LABEL = PROV DATE = 77-02-16 PACK NBR = 000 LASSE

```

0000 IDENT SP
0001 SUBROUTINE SP(IR)
0002 C
0003 C
0004 C DENNA RUTIN ANVÄNDS FÖR ATT DRA SPÄRRFRIGÖRAREN
0005 C TILL BLADVERKEN SÅ ATT DESSA KAN LÄMNA SINA
0006 C SYNKRONISERINGSLÄGEN.
0007 C
0008 C RUTINEN ANROPAS CALL SP(IR)
0009 C IR= DEN RAD PÅ INDIKATIONSTABLÄERNA VARS
0010 C BLADVERK SKALL FRIGÖRAS 1<IR<<4
0011 C IR=5 SLÄPP SPÄRRFRIGÖRAREN
0012 C
0013 C
0014 C COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0015 C *,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0016 C *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0017 C *,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0018 C INTEGER BUF,BUF1
0019 C
0020 C NOLLSTÄLL SPÄRRFRIGÖRARSIGNAL
0021 C
0022 C MANR=IAND(MANR,#OFFF)
0023 C CALL UT($0002)
0024 C
0025 C VÄLJ FUNKTION
0026 C
0027 C GO TO(10,20,30,40,50),IR
0028 C CALL BSET(MANR,15)
0029 C CALL UT2(1,MANR)
0030 C GO TO 100
0031 C CALL BSET(MANR,14)
0032 C CALL UT2(1,MANR)
0033 C GO TO 100
0034 C CALL BSET(MANR,13)
0035 C CALL UT2(1,MANR)
0036 C GO TO 100
0037 C CALL BSET(MANR,12)
0038 C CALL UT2(1,MANR)
0039 C GO TO 100
0040 C CALL UT2(1,MANR)
0041 C RETURN
0042 C END
:EOS
:ECF

```

```

0000 IDENT START
0001 SUBROUTINE START
0002 C
0003 C
0004 C DENNA RUTIN RESERVERAR ETT MÅNADS- OCH ETT DYGN-
0005 C REGISTER FÖR VARJE MATERIAL SÅ ATT DESS FÖRBRUKNING
0006 C KAN REGISTRERAS. RUTINEN SER OCKSÅ TILL ATT DE NÖD
0007 C VÄNDIGA ÅTGÄRDER SOM BEHÖVS BEROENDE PÅ BÄDDENS
0008 C STATUS VID START VIDTAS.
0009 C
0010 C
0011 C COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0012 C *,IBD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0013 C *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0014 C *,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0015 C INTEGER BUF,BUF1,ECB(2),INB(5),UTB(5)
0016 C LOGICAL BTEST
0017 C DATA (INB(I),I=1,5) /8,7,6,5,4/
0018 C DATA (UTB(I),I=1,5) /5,4,3,2,1/
0019 C IECB=IADDR(ECB)
0020 C IS=3
0021 C ISK=3
0022 C IY=1
0023 C DO 10 L=1,4
0024 C BUF(L)=0.
0025 C BUF1(L)=0.
0026 C BOR=0.
0027 C IF (IBDN.EQ. IBDNR(2)) IY=2
0028 C
0029 C SKALL ENDAST FÖLJESEDLAR LÄSAS IN ?
0030 C JA=HOPP 760
0031 C
0032 C IF (IBD(IY,1,8).EQ.1) GO TO 760
0033 C
0034 C KONTROLLERA BÄDDPLATSVÄL >0. STÄLL UT SIGNAL
0035 C FÖR VALD BÄDDPLATS.
0036 C
0037 C IBPL=IBDNR(IY)/100
0038 C IF (IBPL.LE.0) GO TO 3000
0039 C IF (IBPL.EQ.6) IBPL=5
0040 C MANR=0
0041 C MANR1=0
0042 C CALL BSET(MANR,UTB(IBPL))
0043 C CALL UT2(1,MANR)
0044 C
0045 C KONTROLLERA ATT MAN EJ ÖVERSTIGER 30 MATERIAL I MÅNADEN.

```

```

0046 C
0047 IVS=0
0048 IDS=0
0049 DO 600 IQ=1,30
0050 IF(MREG(IQ,1).EQ.0) IVS=IVS+1
0051 CONTINUE
0052 DO 650 L=1,20
0053 IF(BD(IY,L,1).EQ.0.) GO TO 630
0054 DO 625 IQ=1,30
0055 IV=IAND(IBD(IY,L,4),$FF20)
0056 IF((IBD(IY,L,3).EQ.MREG(IQ,1)).AND.(IV
0057 *.EQ.MREG(IQ,2))) GO TO 630
0058 CONTINUE
0059 IDS=IDS+1
0060 CONTINUE
0061 CONTINUE
0062 IF(IDS.LE.IVS) GO TO 700
0063 WRITE(IDEV2,1004)
0064 GO TO 3000
0065 C
0066 C TILDELA VARJE MATERIAL ETT MÅNADS- OCH DYGNREGISTER
0067 C OM MATERIALET REDAN ANVÄNT SE TILL ATT DET KOPPLAS
0068 C IHOP MED DET REGISTER SOM REDAN ÄR ANVÄNT.
0069 C
0070 700 CONTINUE
0071 DO 720 L=1,20
0072 IF(BD(IY,L,1).EQ.0.) GO TO 710
0073 DO 713 J=1,30
0074 IV=IAND(IBD(IY,L,4),$FF20)
0075 IF((IBD(IY,L,3).EQ.MREG(J,1)).AND.(IV
0076 *.EQ.MREG(J,2))) GO TO 715
0077 713 CONTINUE
0078 DO 714 J=1,30
0079 IF(MREG(J,1).EQ.0) GO TO 715
0080 CONTINUE
0081 714 MREG(J,1)=IBD(IY,L,3)
0082 715 MREG(J,2)=IV
0083 IBD(IY,L,13)=J
0084 CONTINUE
0085 720 CONTINUE
0086 710 NR=IV
0087 C
0088 C RÄKNA OM TILL JÄRNVÄGSVAGNAR SKRIV BÄDDNINGSLISTA.
0089 C
0090 CALL RED
0091 CALL ATTCHW(IDEV1,M)
0092 CALL SKRIV(4,0)
0093 CALL DETACH(IDEV1,M)

```

```

0094 C HAR BÄDDPLATSVÄL KVITTERATS ?
0095 C JA=HOPP TILL 730
0096 C NEJ=LARMUTSKRIFT VÄNTA 30 SEK NY AVLÄSNING
0097 C
0098 C
0099 C 725
0100 CALL IN(5,INO)
0101 INO=INO.XOR.#FFFF
0102 IF(BTEST(INO,INB(IBPL))) GO TO 730
0103 WRITE(IDEV2,1002)
0104 CALL WAIT(30,2,M)
0105 GO TO 725
0106 CONTINUE
0107 C 730
0108 C
0109 C
0110 C BÄDD STARTAD TIDIGARE ?
0111 C NEJ=HOPP 750
0112 C
0113 C IF(BD(NR,1,4).EQ.0.) GO TO 750
0114 C
0115 C STÄLL UPP INDIKATIONSTABLÄRNA
0116 C
0117 C CALL RA
0118 C IS=200
0119 C GO TO 3000
0120 C
0121 C GÖR KLART ATT LOSSA GRUNDSATSEN
0122 C STÄLL UPP INDIKATIONSTABLÄRNA
0123 C
0124 C BD(NR,1,3)=GR(NR)
0125 C CALL RA
0126 C GO TO 3000
0127 C
0128 C AKTIVERA KLAR FÖR ATT KUNNA LÄSA IN
0129 C FÖLJESEDLARNA TILL JÄRNVÄGSVÄGNARNA.
0130 C
0131 C NR=IY
0132 C CALL SKRIV(4,0)
0133 C CALL ACTIV('KLAR ',IECB,0,M)
0134 C IS=1
0135 C GO TO 3001
0136 C 1002
0137 C FORMAT(' FEL BÄDDPLATS INSTÄLLD PÅ BANDEN')
0138 C 1004
0139 C FORMAT(' MAX 30 MATERIAL PER MÅNAD')
0140 C 3000
0141 C ISK=0
0142 C 3001
0143 C CONTINUE
0144 C RETURN
0145 C END
:EOS
:EOF

```



```

0000 IDENT STARUP
0001 C
0002 C
0003 C DENNA RUTIN KONTROLLERAR DE TRYCKKNAPPAR SOM FINNS
0004 C OCH VIDTAR NÖDVÄNDIG ÅTGÄRDER EFTER AKTIVERING AV
0005 C DESSA. RUTINEN SER OCKSÅ TILL ATT MÅNADS- OCH
0006 C DYGNSREGISTREN SKRIVS UT VID RÄTT TIDPUNKTER.
0007 C VID SPÄNNINGSBORTFALL AKTIVERAS DENNA RUTIN AUTOMATISKT
0008 C AV MONITORN DÅ SPÄNNINGEN ÅTERKOMMER. VAR 200 MS SER
0009 C RUTINEN TILL ATT VÄGEN AVLÄSES.
0010 C
0011 C
0012 C COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0013 C *,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0014 C *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0015 C *,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0016 C INTEGER BUF,BUF1,IDA(6),ECB(2)
0017 C LOGICAL BTEST
0018 C INTEGER ITE(6),K
0019 C DATA(ITE(I),I=1,5) /14,13,12,11,9/
0020 C IECB=IADDR(ECB)
0021 C
0022 C ÅTERSTÄLLNING AV UTSIGNALER EFTER SPÄNNINGSBORTFALL.
0023 C
0024 C
0025 C CALL UT2(1,MANR)
0026 C CALL UT2(2,MANR1)
0027 C
0028 C LÄS KLOCKAN OCH DATUM FÖR ATT SE OM
0029 C FÖRBRUKNINGUTSKRIFT SKALL GÖRAS.
0030 C
0031 C CALL DATE(IDA,M)
0032 C IF((IDA(4).EQ.0).AND.(IDA(5).EQ.1)) GO TO 50
0033 C GO TO 100
0034 C
0035 C VÄNTA 1 MIN. OM BÄDD ÄR STARTAD STOPPAS DENNA
0036 C
0037 C CALL WAIT(1,3,M)
0038 C IF(NR.EQ.0) GO TO 60
0039 C CALL BCLR(MANR,D)
0040 C CALL STOPP
0041 C IF(IDA(2).EQ.1) GO TO 70
0042 C
0043 C DYGNSUTSKRIFT OCH NOLLSTÄLLNING AV DYGNSREGISTER.
0044 C
0045 C CALL ATTCHW(IDEV1,M)
0046 C CALL SKRTV(A,?)

```

```

0046 WRITE(IDEV1,1000)
0047 CALL DETACH(IDEV1,M)
0048 GO TO 100
0049 C
0050 C MÅNADS- OCH DYGNUTSKRIFT SAMT NOLLSTÄLLNING
0051 C AV MÅNADS- OCH DYGNREGISTEREN.
0052 C
0053 70 CALL ATTCHW(IDEV1,M)
0054 CALL SKRIV(8,1)
0055 WRITE(IDEV1,1001)
0056 CALL DETACH(IDEV1,M)
0057 C
0058 C LÄS TRYCKKNAPPAR
0059 C
0060 100 CALL IN(5,IL)
0061 IL=IL.XOR.#FFFF
0062 IBL=IL.XOR.IDO
0063 ITO=IL.AND.IBL
0064 IDO=IL
0065 C
0066 C ÄR TRYCKKNAPP AKTIVERAD ?
0067 C
0068 25 DO 10 K=1,5
0069 IF(BTEST(ITO,ITE(K))) GO TO 15
0070 CONTINUE
0071 10 CONTINUE
0072 C
0073 C INGEN TRYCKKNAPP AKTIVERAD
0074 C
0075 GO TO 20
0076 C
0077 C VÄLJ FUNKTION BERÖENDE PÅ TRYCKKNAPP.
0078 C OM BÄDD EJ STARTAD KAN ENDAST DIALOG AKTIVERAS.
0079 C
0080 15 GO TO (16,16,16,16,6),K
0081 16 IF((NR.EQ.D).OR.(ISK.EQ.3)) GO TO 12
0082 GO TO (1,2,3,4,6),K
0083 C
0084 C NÄSTA POST
0085 C
0086 1 IECB=IADDR(ECB)
0087 IBD(NR,LP(NR),10)=$4156
0088 IBD(NR,LP(NR),11)=$4252
0089 LP(NR)=LP(NR)+1
0090 CALL ACTIV('FORD ',IECB,D,M)
0091 KSTAT=1
0092 GO TO 12
0093 C

```

```

0094 C NÄSTA MATERIAL
0095 C
0096 C LP(NR)=30
0097 C CALL LAMPA(7)
0098 C CALL LAMPA(4)
0099 C KSTAT=1
0100 C GO TO 12
0101 C
0102 C STEKENJOKK LOSSAD
0103 C
0104 C 3 IF(MT(NR).EQ.1) GO TO 12
0105 C ISK=1
0106 C CALL ACTIV('JVL ',IECB,D,M)
0107 C GO TO 12
0108 C
0109 C AITIK LOSSAD
0110 C
0111 C 4 IF(MT(NR).EQ.1) GO TO 12
0112 C ISK=0
0113 C CALL ACTIV('JVL ',IECB,D,M)
0114 C GO TO 12
0115 C
0116 C AKTIVERA DIALOGPROGRAMMET.
0117 C
0118 C 6 CALL ACTIV('LAS ',IECB,D,M)
0119 C 20 CONTINUE
0120 C
0121 C OM INGEN BÄDD STARTAD SÄTTES VÅGPROGRAMMETS STATUS=1
0122 C
0123 C IF(NR.EQ.0) KSTAT=1
0124 C
0125 C INGEN BÄDD STARTAD = HOPP TILL 30
0126 C
0127 C IF((NR.EQ.0).OR.(ISK.EQ.3)) GO TO 30
0128 C
0129 C OM OMSTART AV BÄDD HOPP TILL 99 FÖR ATT EV
0130 C STÄLLA SJUSEGMENTSDISPLAY VID JÄRNVÄGSLÖSTNINGSPLATS.
0131 C
0132 C IF(IS.EQ.200) GO TO 99
0133 C
0134 C NOTERA OM ENDAST JÄRNVÄGSSVAGNAR ÅTERSTÅR
0135 C
0136 C IF(IS.EQ.1) IBD(NR,1,8)=2
0137 C IF(IBD(NR,1,8).EQ.2) IS=1
0138 C
0139 C VÄNTA 200 MS LÄS VÅG
0140 C
0141 C CALL WAIT(2,1,M)

```

```

0142 CALL VAG(KSTAT)
0143 GO TO 30
0144 ISK=4
99 IF(MT(NR).LE.1) GO TO 98
0145 IF((MT(NR).LE.2).AND.(IBD(NR,LP(NR),1).LE.1)) GO TO 98
0146 CALL ACTIV('JVL',IECB,O,M)
0147 IS=0
98 GO TO 30
0148 GO TO 30
0149 FORMAT(1H0,'DYGNSUTSKRIFT')
0150 FORMAT(1H0,'DYGNS- OCH MÅNADSUTSKRIFT')
0151 STOP
0152 END
0153 :E05
:EOF

```

S:

```

0000 IDENT STOPP
0001 SUBROUTINE STOPP
0002 C
0003 C
0004 C DENNA RUTIN ANVÄNDS DÅ OPERATÖREN VILL
0005 C AVBRYTA PÅGÅENDE BÄDDNING.
0006 C
0007 C
0008 C COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0009 C *,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0010 C *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0011 C *,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0012 C INTEGER BUF,BUF1
0013 C ISK=3
0014 C
0015 C NOLLSTÄLL BITAR I MANR1 OCH LÄGG
0016 C UT ORDET PÅ UTGÅNGSKORTET.
0017 C
0018 C MANR1=MANR1.AND.#000F
0019 C CALL UT2(2,MANR1)
0020 C
0021 C SLÄCK ALL LAMPOR TÄND SEDAN RÖD LAMPA.
0022 C
0023 C CALL LAMPA(7)
0024 C CALL LAMPA(4)
0025 C
0026 C NOLLSTÄLL BLADVERKEN.
0027 C
0028 C CALL NOLL(5)
0029 C
0030 C GÖR MULTIPLELUTSKRIFT
0031 C
0032 C CALL ATTCHW(IDEV1,M)
0033 C CALL SKRIV(1,0)
0034 C WRITE(IDEV1,10)
0035 C CALL DETACH(IDEV1,M)
0036 C NR=0
0037 C 10 FORMAT(' BÄDDNINGEN AVBRUTEN' )
0038 C RETURN
0039 C END
:EOS
:EOF

```



```

00039 0032 8820
0034 0000
00040 0036 8F20
0038 002C R
00041 003A F03A
00042

```

SYMBOL TABLE

```

VT • 0000 R M10S1 003A A FUNK 0080 A NO 001A R
AT 001E R NU 002C R PU 003A R

```

ASS.ERR. 00000

```

:EOF
PROG ELAPSED TIME: 00H-00M-00S-000MS-

```

S:

IDENT VAG
SUBROUTINE VAG(KSTAT)

DENNA RUTIN ANVÄDS FÖR ATT LÄSA AV VÄGEN OCH ÖVERFÖRA VÄGVÄRDET TILL RUTINEN RAKNA FÖR REGISTRERING. RUTINEN TÄNDER OCH SLÄCKER ÄVEN DEN RÖDA, GULA OCH GRÖNA LAMPAN FÖR ATT HÅLLA TRUCKFÖRAREN UPPLYST OM VAD VÄGENS STATUS ÄR FÖR NÄRVARANDE

RUTINEN ANROPAS CALL VAG(KSTAT)

KSTAT=1 VIKT I VÄGKÄRL ÖVERSKRIDER TARERINGSGRÄNS
KSTAT=2 OM VIKT < 400 KG ELLER TRUCK NÄRVARANDE
SÅ KOMMER VÅG ATT TARERAS.
KSTAT=3 VÄNTA PÅ ATT LOSSAD VIKT > 1000 KG.
KSTAT=4 SÖK STILLESTÅND OCH ANROPA DÄREFTER RAKNA FÖR REGISTRERING AV VIKT
KSTAT=5 TARERA VÄGEN.

COMMON IBDNR(2), IA(2), MT(2), LP(2), GR(2), IBD(2,20,15)
*, BD(2,20,5), NR, IS, ISK, BOR, KP(2), BUF(4), BUF1(4), IBDN
*, MAN, TARA, VL, SKL, MANR, MANR1, IDEV1, IDEV2, IDEV3
*, IDEV4, IDEV5, IDEV6, MREG(30,3), REG(30,2)
INTEGER BUF, BUF1
LOGICAL BTEST
SKOPA=0.

SKALL NÅGON VIKT REGISTRERAS ? NEJ=HOPP 80

IF((LP(NR).GE.21).OR.(BD(NR,LP(NR),1).EQ.0.
*)) GO TO 80
IF(IS.EQ.1) GO TO 80

AVLÄS VÅGINSTRUMENT
OM FEL PÅ VÅG HOPP TILL 85

CALL BSET(MANR1,D)
CALL UT2(2,MANR1)
CALL IN(5,INO)
IF (BTEST(INO,15)) GOTO 85
CALL DIOMIN(4,KD,KVIKT)
CALL BCLR(MANR1,D)
CALL UT2(2,MANR1)

0000
0001
0002 C
0003 C
0004 C
0005 C
0006 C
0007 C
0008 C
0009 C
0010 C
0011 C
0012 C
0013 C
0014 C
0015 C
0016 C
0017 C
0018 C
0019 C
0020 C
0021 C
0022
0023
0024
0025
0026
0027
0028
0029 C
0030 C
0031 C
0032
0033
0034
0035 C
0036 C
0037 C
0038 C
0039 100
0040
0041
0042
0043
0044
0045


```

0046 VIKT=KVIKT*0.01
0047 C
0048 C VILKEN STATUS HAR UPPNÅTS ?
0049 C
0050 C GOTO (1,2,3,4,5),KSTAT
0051 C
0052 C VIKT I VÄGKÄRL < TARERINGSGRÄNS ?
0053 C
0054 C 1 IF (VIKT.GT.TARA) GOTO 10
0055 C CALL LAMPA(5)
0056 C KSTAT=2
0057 C GO TO 10
0058 C
0059 C SKALL TARERING UTFÖRAS ?
0060 C
0061 C 2 IF (VIKT.LT.=4) GOTO 14
0062 C CALL IN(5,K)
0063 C IF (.NOT.BTEST(K,10)) GOTO 14
0064 C GOTO 10
0065 C 14 CALL BSET(MANR,D)
0066 C CALL UT2(1,MANR)
0067 C KSTAT=5
0068 C GOTO 10
0069 C
0070 C UTFÖR OM MÖJLIGT TARERING
0071 C
0072 C 5 IF (KVIKT.NE.KVIKT1) GOTO 13
0073 C KTARA=KVIKT
0074 C CALL LAMPA(6)
0075 C KSTAT=3
0076 C GOTO 10
0077 C
0078 C KAN VIKT REGISTRERAS > 1000 KG
0079 C
0080 C 3 IF ((KVIKT-KTARA).LT.100) GOTO 10
0081 C CALL LAMPA(7)
0082 C KSTAT=4
0083 C GOTO 10
0084 C
0085 C OM DUNK-FÖRFARANDE ÖNSKAS VÄNTA PÅ KLARSIGNAL
0086 C ANNARS HOPP TILL 41
0087 C
0088 C 4 IF(BTEST(INO,D)) GO TO 41
0089 C IF (.NOT.BTEST(INO,1)) GOTO 41
0090 C GO TO 13
0091 C
0092 C STILLESTÄND ? NEJ=HOPP TILL 13.
0093 C

```

```

0094 41 IF (KVIKT1.NE.KVIKT) GOTO 13
0095 C
0096 C FÖR ÖVER KORREKT VIKT TILL RAKNA
0097 C
0098 SKOPA=(KVIKT-KTARA)*D.01
0099 CALL LAMPA(4)
0100 CALL BCLR(MANR,D)
0101 CALL UT2(1,MANR)
0102 80 IF(NR.EQ.D) GO TO 82
0103 CALL RAKNA(SKOPA)
0104 C
0105 C FÖRBERED NY INVÄGNING.
0106 C
0107 82 KSTAT=1
0108 KVIKT=D
0109 GOTO 10
0110 C
0111 C LARM VID FEL PÅ VÅG. VÄNTA TILLS FRÅGA
0112 C OM VÅG LAGAD BESVATS MED JA.
0113 C
0114 85 WRITE(IDEV2,90)
0115 90 FORMAT(' FEL PÅ VÅG')
0116 CALL LAMPA(4)
0117 CALL BCLR(MANR1,0)
0118 CALL UT2(2,MANR1)
0119 92 IBEL=#0707
0120 WRITE(IDEV2,95)IBEL
0121 95 FORMAT(' ÄR VÅGEN LAGAD?',A2)
0122 JA=7
0123 CALL LAES(2,JA,IDEV2)
0124 IF(JA.NE.#4A41) GO TO 92
0125 IF (KSTAT.EQ.1) CALL LAMPA(4)
0126 IF (KSTAT.EQ.2) CALL LAMPA(5)
0127 IF (KSTAT.EQ.3) CALL LAMPA(6)
0128 IF (KSTAT.EQ.4) CALL LAMPA(6)
0129 IF (KSTAT.EQ.5) CALL LAMPA(5)
0130 GO TO 100
0131 C
0132 C
0133 13 KVIKT1=KVIKT
0134 10 CONTINUE
0135 RETURN
0136 END
:EOS
:EOF

```

```

0000 IDENT WRL
0001 SUBROUTINE WRL(ITE,KL,REAL,INT,LD,IDEV)
0002
0003
0004 DENNA RUTIN ANVÄNDS FÖR ATT SAMMANFOGA
0005 ANROPEN TILL VISSA LÄS- OCH SKRIVRUTINER
0006 OCH PÅ SÅ SÄTT SPARA PROGRAMUTRYMME.
0007
0008 RUTINEN ANROPAS CALL WRL(ITE,KL,REAL,INT,LD,IDEV)
0009
0010 ITE=STARTCELL TILL DATA-AREA MED TEXTSTRÄNG
0011 KL=ANTAL TECKEN SOM INGÅR I DATA AREAN
0012 REAL=ADRESS TILL BUFFER FÖR REELLTAL VID INLÄSNING
0013 INT=0 HELTAL SKALL LÄSAS OCH STOPPAS I BUF2
0014 INT=1 REELLTAL SKALL LÄSAS IN TILL REAL
0015 INT>1 TEXT SKALL LÄSAS IN
0016 LD=ANTAL MAX TECKEN SOM FÅR FINNAS I INLÄST TEXT
0017 IDEV=ADRESS TILL YTRE ENHET VARIFRÅN INLÄSNING SKER
0018
0019
0020 COMMON IBDNR(2),IA(2),MT(2),LP(2),GR(2),IBD(2,20,15)
0021 *,BD(2,20,5),NR,IS,ISK,BOR,KP(2),BUF(4),BUF1(4),IBDN
0022 *,MAN,TARA,VL,SKL,MANR,MANR1,IDEV1,IDEV2,IDEV3
0023 *,IDEV4,IDEV5,IDEV6,MREG(30,3),REG(30,2)
0024 INTEGER BUF,BUF1,BUF2(4),ECB(6),ECB1(6)
0025
0026 SKALL TEXT SKIVAS UT ? NEJ=HOPP TILL 5
0027
0028 IF((KL.EQ.2).AND.(INT.EQ.1)) GO TO 5
0029 IF((KL.EQ.2).AND.(INT.EQ.7)) GO TO 5
0030 CALL WRITE(ECB1,ITE,IDEV2,KL)
0031 CALL WRITES(IDEV2)
0032
0033 CONTINUE
0034
0035 OM INT <2 SÅ SKALL TALVÄRDEN LÄSAS IN
0036
0037 IF(INT.GE.2) GO TO 10
0038 CALL REED(ECB,BUF2,REAL,INT,IDEV)
0039 GO TO 20
0040
0041 BEROENDE PÅ INT:S VÄRDE SÅ LÄS IN EN ASCII TEXT
0042 OCH LÄGG DEN I SIN SVARSBUFFER.
0043
0044 IF(INT.EQ.5) CALL LAES(LD,BUF,IDEV)
0045 IF(INT.EQ.6) CALL LAES(LD,BUF1,IDEV)
0046 IF(INT.EQ.7) CALL LAES(LD,INT,IDEV)
0047 RETURN
0048
0049 END

```

0000
0001
0002
0003
0004
0005
0006
0007
0008
0009
0010
0011
0012
0013
0014
0015
0016
0017
0018
0019
0020
0021
0022
0023
0024
0025
0026
0027
0028
0029
0030
0031
0032
0033
0034
0035
0036
0037
0038
0039
0040
0041
0042
0043
0044
0045
0046
0047

2

4

6

8

10

12

DATUM 1978 224 BÄDDNINGSLISTA
 KLOCKSLAG 13.22
 BÄDDNUMMER 310
 ANTAL MULTIPLAR 20
 SENAST FÄRDIGA MULTIPEL NR 0
 ANTAL ÅTERSTÄENDE MULTIPLAR 20

POSTNR MTRLHÖG HELA BÄDDEN
 BÖRVIKT

	TON	VAGNSLASTER	ÅTERSTÅR	TON	VAGNSLASTER
1	721.17	7	721.17	7	
2	315.79	4		4	
3	160.00				
4	107.34				
5	240.00				
6	575.60				
7	108.17				
8	107.34				
				107.34	
				240.00	
				575.60	
				108.17	
				107.34	

32

34

36

38

40

42

44

46

MULTIPELLISTA

DATUM 1978 224
 KLOCKSLAG 13.23
 BÄDDNUMMER 310
 ANTAL MULTIPLAR 20
 SENAST FÄRDIGA MULTIPEL NR 1
 ANTAL ÅTERSTÄENDE MULTIPLAR 19

POSTNR	MTRLHÖG	HELA BÄDDEN	BÖRVIKT	ÅTERSTÅR	DENNA MULTIPEL	ÅR	VAGNSLASTER	VAGNSLASTER
			TON	TON	TON	TON	100.00	100.00
1	34A	721.17	721.17	621.17	100.00	100.00	0	0
2	00B	315.79	7	7	0	0	0	0
3	00R	160.00	4	4	0	0	0	0
4	75A	107.34		107.34	.00	.00		.00
5	91B	240.00		240.00	.00	.00		.00
6	32R	575.60		575.60	.00	.00		.00
7	35E	108.17		108.17	.00	.00		.00
8	75A	107.34		107.34	.00	.00		.00

MULTIPELLISTA

1978 224
 13.38
 310
 20
 2
 18

HELA BÄDDEN

1 34A 721.17
 2 00B 315.79
 3 00R 160.00
 4 75A 107.34
 5 91B 240.00
 6 32R 575.60
 7 35E 108.17
 8 75A 107.34

DENNA MULTIPEL

BÖR TON 160.95
 13.52
 26.55
 60.31
 5.69
 5.65

ÄR TON 158.00
 12.00
 28.00
 30.00
 8.00
 8.00

ÅTERSTÅR TON 563.17
 95.34
 212.00
 545.60
 100.17
 99.34

VAGNSLASTER 7
 4
 7
 4
 7
 4

VAGNSLASTER 0
 0

BÄDDNINGEN AVBRUTEN

MULTIPELLISTA

DATUM 1978 224
 KLOCKSLAG 13.33
 BÄDDNUMMER 310
 ANTAL MULTIPLAR 20
 SENAST FÄRDIGA MULTIPEL NR 2
 ANTAL ÅTERSTÄENDE MULTIPLAR 18

POSTNR MTRLHÖG HELA BÄDDEN

	POSTNR	MTRLHÖG	BÖRVIKT		ÅTERSTÅR		DENNA MULTIPEL		ÄR	
			TON	VAGNSLASTER	TON	VAGNSLASTER	TON	VAGNSLASTER	TON	VAGNSLASTER
1	34A	721.17	593.17	7	132.69	128.00	0	0	0	0
2	00B	315.79	99.34	7	5.65	8.00	0	0	8.00	0
3	00R	160.00	226.00	4	12.63	14.00	0	0	14.00	0
4	75A	107.34	545.60		30.29	30.00			30.00	
5	91B	240.00	100.17		5.67	8.00			8.00	
6	32R	575.60	99.34		5.65	8.00			8.00	
7	35E	108.17								
8	75A	107.34								

2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46

4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46

DATUM 1978 224 SLUTUTSKRIFT

KLOCKSLAG 15.0

BÄDDNUMMER 310

ANTAL MULTIPLAR 20

SENAST FÄRDIGA MULTIPLEL NR 20

ANTAL ÅTERSTÄENDE MULTIPLAR 0

POSTNR MTRLHÖG HELA BÄDDEN

BÖRVIKT

TON

34A 967.66

00B 69.30

00R 68.53

75A 107.34

91B 240.00

32R 575.60

35E 108.17

75A 107.34

31F 91.47

TOTALT 2335.41

DENNA MULTIPLEL

BÖR

TON

ÅTERSTÅR

TON

VAGNSLASTER

TON

VAGNSLASTER

TON

ÄR

TON

VAGNSLASTER

TON

TOTALT 2346.00

*=VAGNSLASTER SAKNAS

2
 4
 6
 8
 10 DATUM 19781220 MATERIALFÖRBRUKNING
 12 KLOCKSLAG 0. 2

MTRLHÖG	UNDER DYGNET	UNDER MÅNADEN
32E	1592.00	1592.00
00B	.00	.00
00R	675.00	675.00
91B	142.00	142.00
32X	96.00	96.00
32D	54.00	54.00
91H	48.00	48.00
75A	158.00	158.00

22
 24 DYGN SVTSKRIFT

26
 28
 30
 32
 34
 36
 38
 40
 42
 44
 46


```

00042 0038 230F ANK A3,X'F'
00043 003A 920C ADR A2,A3
00044 003C 1D01 SUK A5,1
00045 003E 590E RB(1)
00046 0040 8261 ST* A2,POINT
          TRANSFER BINARY
          0042 0000 R
          0044 F03A RTN A14
00047
00048 END

```

SYMBOL TABLE

```

DIOMIN 0002 R M10S1 003A A FUNK 0180 A POINT 0000 R
BCDBIN 002A R LOOP2 0034 R LOOP1 0032 R

```

ASS.ERR. 00000

:EOF PROG ELAPSED TIME: 00H-00M-00S-000MS-

S:

```

00000 IDENT IN
00001
00002
00003
00004 * * * CALL DISM(DISMNR,BCD-VALUE)
00005 *
00006 ENTRY IN
00007
00008 * *
00009 MIOS2 EQU /3A
00010 *
00011 FUNK EQU /180
00012 *
00013 POINT RES 1
00014 *
00015 IN
00016 8330 LDR* A3,A4
00017 812C LDR* A1,A3
00018 9120 ADKL A1,FUNK
00019 0180 INH
00020 20BF OTR A1,0,MIOS2
00021 413A RB(4) *-2
00022 5004
00023
00024 *READ IN
00025 *
00026 INR A2,0,MIOS2
00027 RB(4) *-2
00028 ENB A4,2
00029 ADK A3,A4
00030 LDR* A3,POINT
00031 ST A3,POINT
00032 ST* A2,POINT
00033 RTN A14
00034 END

```

SYMBOL TABLE

```

IN 0002 R MIOS2 003A A FUNK 0180 A POINT 0000 R

```

ASS.ERR. 00000

:EOF
PROG ELAPSED TIME: 00H-00M-00S-000MS-

```

0000 IDENT KLOCKA
0001 SUBROUTINE KLOCKA
0002 INTEGER ECB(2)
0003 INTEGER KTID(3),CLOC(6)
0004 TID
0005 C
0006 IECB=IADDR(ECB)
0007 KBILD=5
0008 WRITE(KBILD,1111)
0009 READ(KBILD,4321) KTID(1),KTID(2),KTID(3)
0010 CALL STDATA(KTID,1)
0011 C
0012 C
0013 C
0014 C
0015 C
0016 WRITE(KBILD,2222)
0017 10 READ(KBILD,1234) KTID(1),KTID(2),KTID(3)
0018 C
0019 CALL STDATA(KTID,D)
0020 C
0021 C
0022 SKRIV DATUM OCH TID
0023 CALL DATE(CLOC,M)
0024 WRITE(KBILD,2100) CLOC(3),CLOC(1),CLOC(2),
0025 *CLOC(4),CLOC(5),CLOC(6)
0026 C
0027 C
0028 1111 FORMAT(' STÄLL KLOCKAN',/
0029 *, ' SEX SIFFROR EX. 103045 (TIM,MIN,SEK)')
0030 2222 FORMAT(' MATA IN DATUM',/
0031 *, ' EX.300876 (DAG,MÅN,ÅR)')
0032 1234 FORMAT(3A2)
0033 4321 FORMAT(3I2)
0034 2100 FORMAT(' DATUM',I5,2I3,' TID',3I3)
0035 RETURN
0036 END
:EOS
:EOF

```

S:

LABEL = PROV DATE = 77-02-16 PACK NBR = 000 LASSE

```

0000 IDENT LAES
0001 SUBROUTINE LAES(NR,LUFFER,KBILD)
0002 INTEGER BUFFER(15),ECB(6)
0003 INTEGER LUFFER(NR)
0004 CONTINUE
0005 K=NR/2
0006 DO 15 L=1,K
0007 LUFFER(L)=0
0008 CALL DECB(KBILD,IADDR(ECB),IADDR(BUFFER),30)
0009 CALL SIOW(2,IADDR(ECB),K)
0010 IF (ECB(4).GT.NR) GOTO 11
0011 K=ECB(4)/2
0012 IF ((ECB(4)-2*K).EQ.0) GOTO 16
0013 K=K+1
0014 DO 10 N=1,K
0015 LUFFER(N)=BUFFER(N)
0016 CONTINUE
0017 GOTO 13
0018 WRITE(KBILD,12) NR
0019 FORMAT (' MAX',I3,' TECKEN')
0020 GOTO 14
0021 CONTINUE
0022 RETURN
0023 END
:EOS
:EOF

```

S:

PRT /S,WRITE DATE = 77-02-16 PACK NBR = 000 LASSE
I AREI = PROV

```

0000 IDENT REED
0001 C
0002 C THIS ROUTINE READS AN ASCII CHARACTER STRING
0003 C AND CONVERTS IT DEPENDING ON THE VALUE OF
0004 C "INT" ON ENTRY. THE MAX NUMBER OF CHARACTERS
0005 C DEPENDS ON THE LENGTH OF "INPB"
0006 C
0007 C INT = 0 : CONVERT STRING TO SINGLE INTEGER
0008 C INT = 1 : " " FLOATING
0009 C INT = 2 : NO CONVERSION; ASCII STRING
0010 C
0011 C A RECORD CAN BE GIVEN AGAIN AFTER VERTICAL
0012 C ARROW LINE FEED/CARRIAGE RETURN; THE PRE-
0013 C CEEDING CHARACTER IS IGNORED BY GIVING A
0014 C LEFT ARROW
0015 C
0016 C A RUB-OUT OR CARRIAGE RETURN TERMINATES
0017 C THE RECORD
0018 C
0019 C CALLING SEQUENCE IS :
0020 C
0021 C CALL REED (ECB1,INPB,REAL,INT,IDEV)
0022 C
0023 C IN WHICH :
0024 C ECB1 = EVENT CONTROL BLOCK (6 WORDS)
0025 C INPB = INPUT BUFFER (ASCII-STRING)
0026 C REAL = LOCATION OF CONVERTED FLOATING VALUE
0027 C INT = " " INTEGER " " (ENTRY CODE)
0028 C IDEV = FILE CODE TO DEVICE FROM WHICH THE STRING IS READ
0029 C
0030 C IF THE CHARACTER STRING FOR A NUMBER CONTAINS
0031 C A WRONG CHARACTER (NOT A +,-,FIGURE OR .)
0032 C "ER" IS TYPED AND THE RECORD MUST BE GIVEN AGAIN
0033 C
0034 C SUBROUTINE REED (ECB,ASST,REAL,INT,IDEV)
0035 C INTEGER ASST(10),CHAR,ECB(6),ECBADD,INPBAD,SIGN,DECC
0036 C
0037 C 10 ECBADD = IADDR(ECB)
0038 C INPBAD = IADDR(ASST)
0039 C 11 CALL DECB (IDEV,ECBADD,INPBAD,20)
0040 C 12 CALL SIOW (2,ECBADD,K)
0041 C IF (K.NE.D) GOTO 10
0042 C
0043 C DO CONVERSION IF REQUESTED
0044 C
0045 C IS = INT
0046 C IF (INT.EQ.2) GOTO 80 ...

```

```

0047 K = ECB(4)
0048 IF (K.EQ.0) GOTO 80
0049 SIGN = 0
0050 M = 0
0051 DECC = 0
0052 IF (INT.EQ.1) M = 1
0053 INT = 0
0054 REAL = 0.
0055 30 DO 60 I = 1,K
0056 CHAR = IBYTE (INPBAD,I-1)
0057 IF (CHAR.EQ.$20) GOTO 60
0058 IF (CHAR.NE.$2D) GOTO 35
0059 SIGN = 1
0060 GOTO 60
0061 35 IF (CHAR.EQ.$2B) GOTO 60
0062 IF (CHAR.NE.$2E) GOTO 40
0063 IF (M.EQ.0) GOTO 70
0064 DECC = 1
0065 GOTO 60
0066 40 IF ( DECC .NE. 0 ) DECC = DECC + 1
0067 CHAR = CHAR - #30
0068 IF (CHAR.LT.0) GOTO 70
0069 IF (CHAR.GT.9) GOTO 70
0070 IF (M.EQ.0) GOTO 50
0071 REAL = REAL * 10. + CHAR
0072 GOTO 60
0073 C
0074 50 IF ( INT .GT. 3276 ) GOTO 70
0075 INT = INT * 10 + CHAR
0076 60 CONTINUE
0077 C
0078 IF ( DECC .LT. 2 ) GOTO 62
0079 DECC = DECC - 1
0080 DO 61 I = 1,DECC
0081 REAL = REAL / 10.
0082 C
0083 62 IF ( SIGN .EQ. 0 ) GOTO 80
0084 IF (M.EQ.0) GOTO 65
0085 REAL = 0. - REAL
0086 GOTO 80
0087 65 INT = 0 - INT
0088 GOTO 80
0089 C
0090 70 IF ( IBYTE(INPBAD,0) .NE. #2F) GOTO 75
0091 ASST(1) = $FFFF
0092 GOTO 80
0093 75 WRITE (IDEV,1000)
0094 INT = IS

```



```

00000
00001
00002
00003
00004
00005
00006
00007
00008
00009
00010
00011
00012
00013
00014
00015
00016
00017
00018
00019
00020

IDENT UT2
CALL UT2(NR,UTORD)

MIOS1
FUNK
UT2

ENTRY
EQU
EQU
INH
LDR*
LDR*
ADKL

ADK
LDR*
LDR*
OTR
RB(4)
OTR
RB(4)
ENB
RTN
END

UT2      20BF
          8330
          862C
          9620
          0080
          000A 1402
          000C 8330
          000E 872C
          0010 463A
          0012 5004
          0014 477A
          0016 5004
          0018 2840
          001A F03A

          A4,2
          A3,A4
          A7,A3
          A6,0,MIOS1
          *-2
          A7,1,MIOS1
          *-2
          A14

```

SYMBOL TABLE

```

UT2      0000 R MIOS1 003A A FUNK 0080 A

```

ASS.ERR. 00000

```

:EOF
PROG ELAPSED TIME: 00H-00M-00S-000MS-

```

S:

```

0000 IDENT WRITE
0001 SUBROUTINE WRITE( ECB, BUFFER, KBUILD, LENGTH)
0002 C
0003 INTEGER ECB(6)
0004 C
0005 ECBADD=IADDR( ECB )
0006 BUFADD=IADDR( BUFFER )
0007 CALL DECB( KBUILD, ECBADD, BUFADD, LENGTH )
0008 CALL SIOW( 5, ECBADD, M )
0009 IF ( M.EQ.0 ) GOTO 100
0010 WRITE ( KBUILD, 1000 )
0011 100 CONTINUE
0012 C
0013 RETURN
0014 1000 FORMAT( ' FEL 1' )
0015 C
0016 END
:EOS
:EDF

```

S: