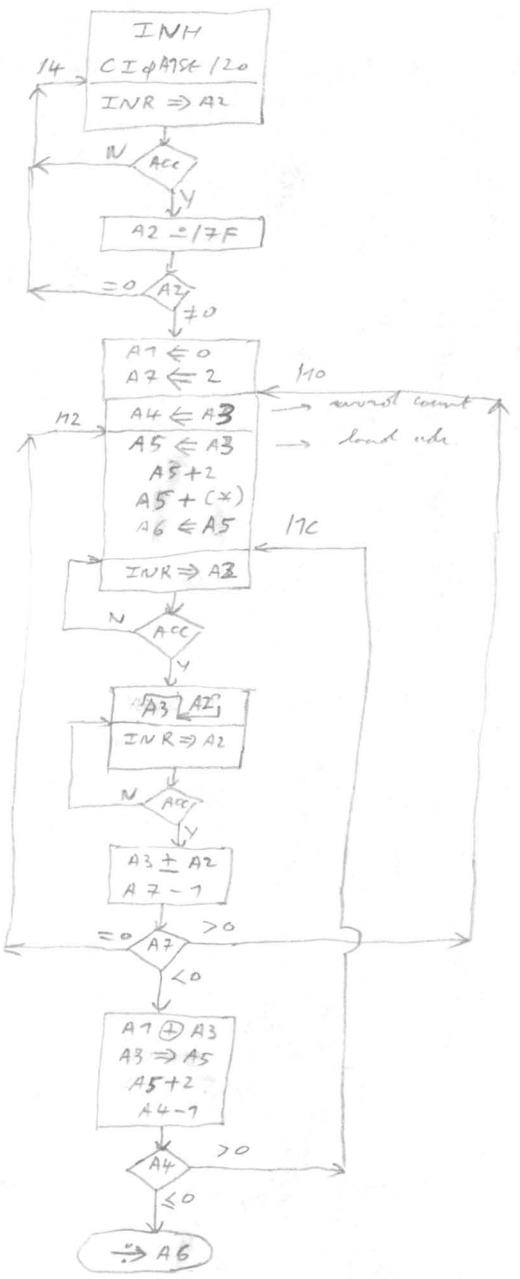
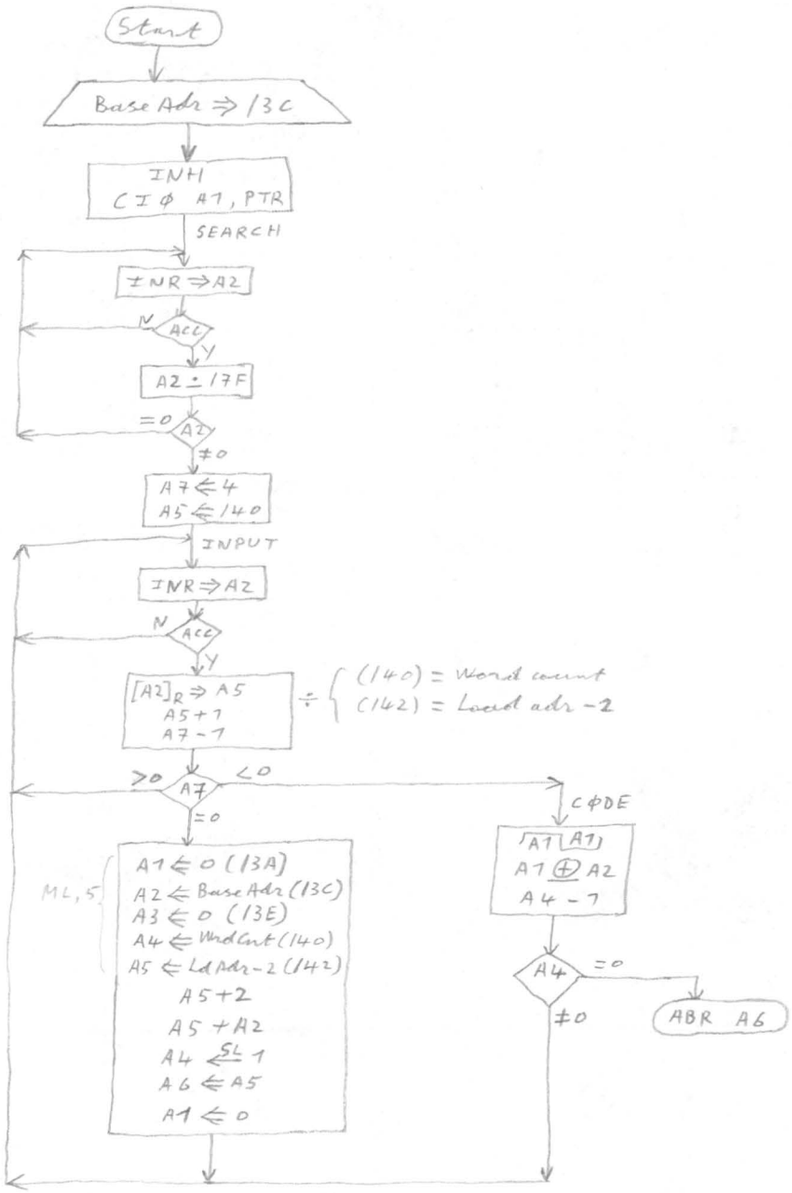
Bootstrap loader

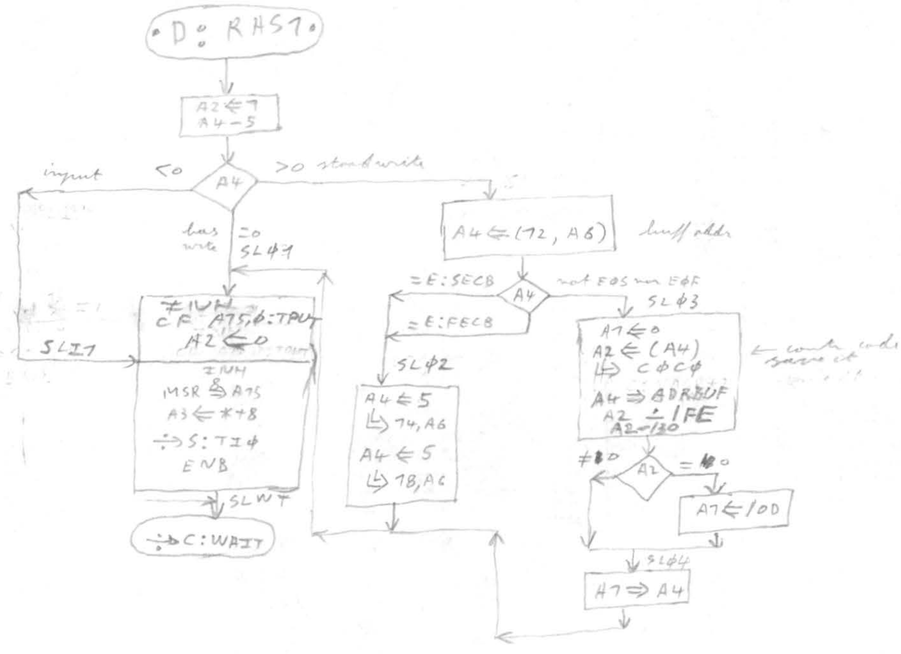
BØØT89



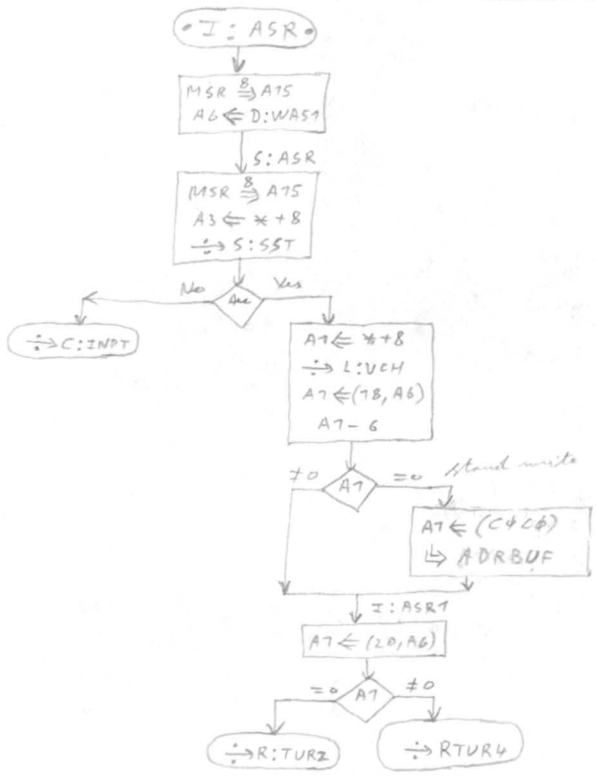
Boatstrap Loader BØØT88 R2V1



DRSL
GPBΦM



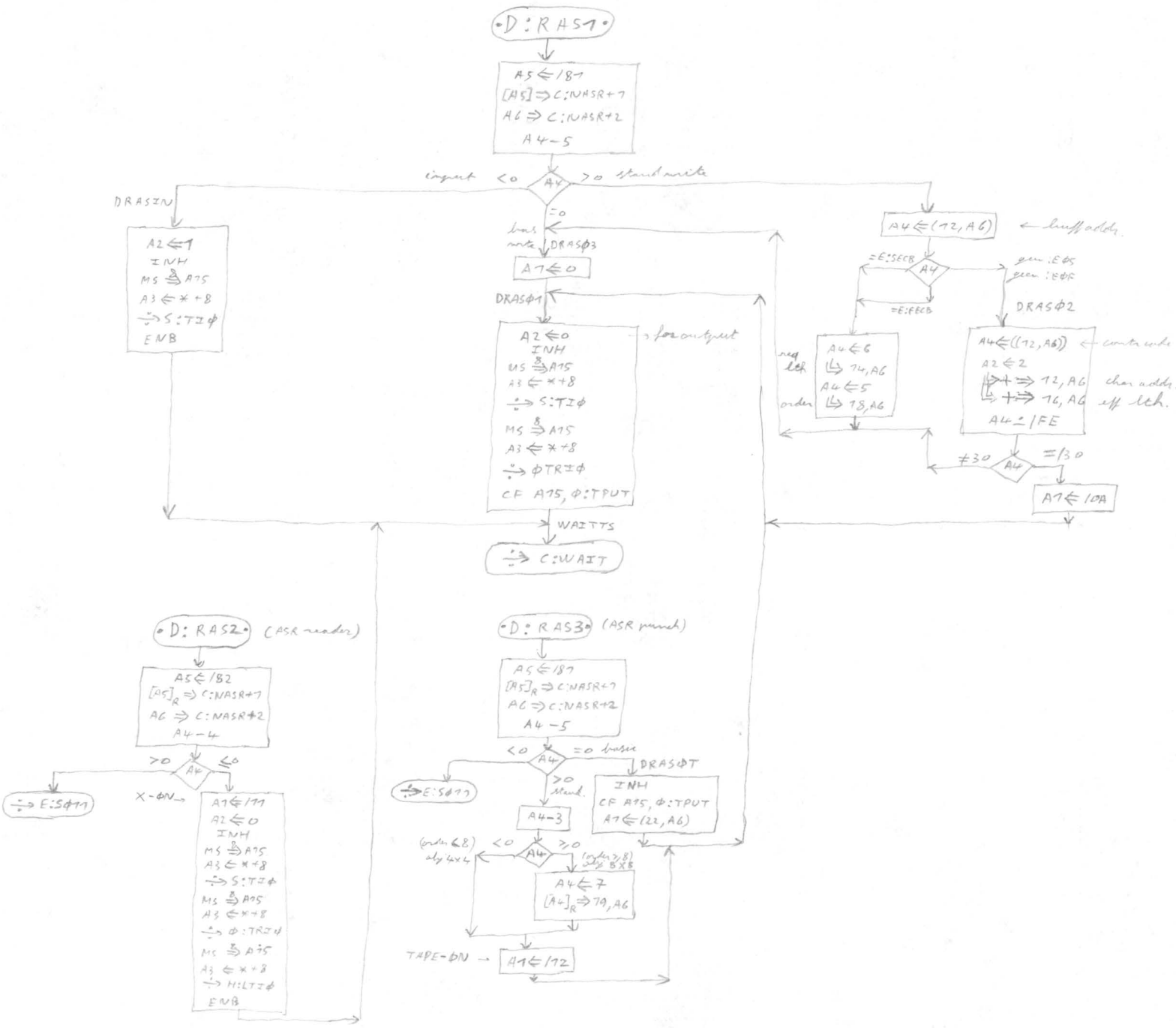
D: RAS1
A2 ⇐ 1
A4 − 5
A4
input
<0
>0 standwrite
=0
SL Φ1
A4 ⇐ (12, A6)
buff addr
≠IUH
CF A15,Φ:TPUT
A2 ⇐ 0
SLI1
IUH
MSR ⇒ A15
A3 ⇐ *+8
÷> S:TIΦ
ENB
SLWT
÷> C:WAIT
=E:SECB
=E:FECB
not EOS nor EΦF
SLΦ2
A4 ⇐ 5
↳ 14, A6
A4 ⇐ 5
↳ 18, AC
SLΦ3
A1 ⇐ 0
A2 ⇐ (A4)
↳ CΦCΦ
A4 ⇒ ADRBUF
A2 ÷ /FE
A2 − 120
A2
= 0
A1 ⇐ /0D
SLΦ4
A1 ⇒ A4

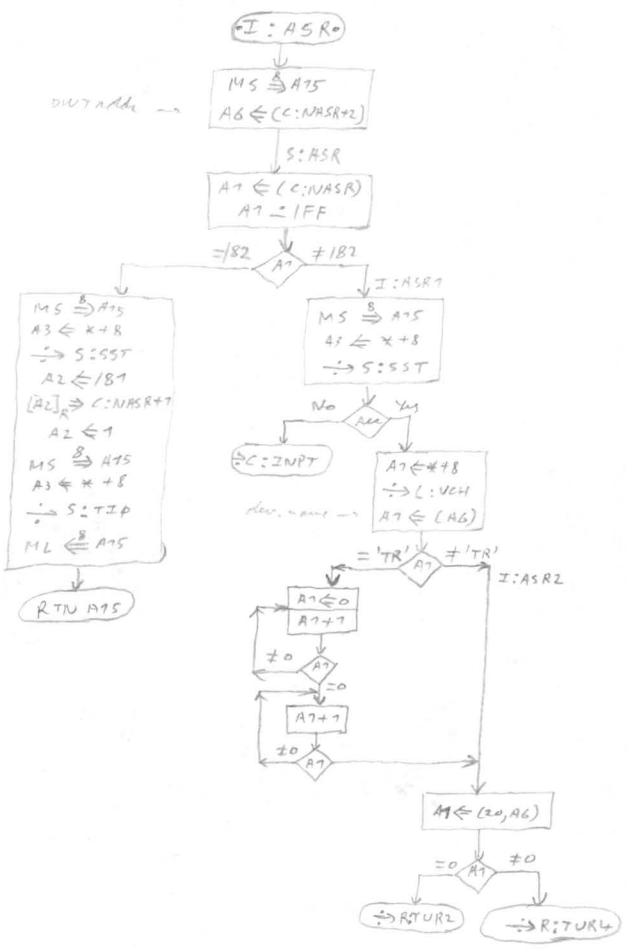


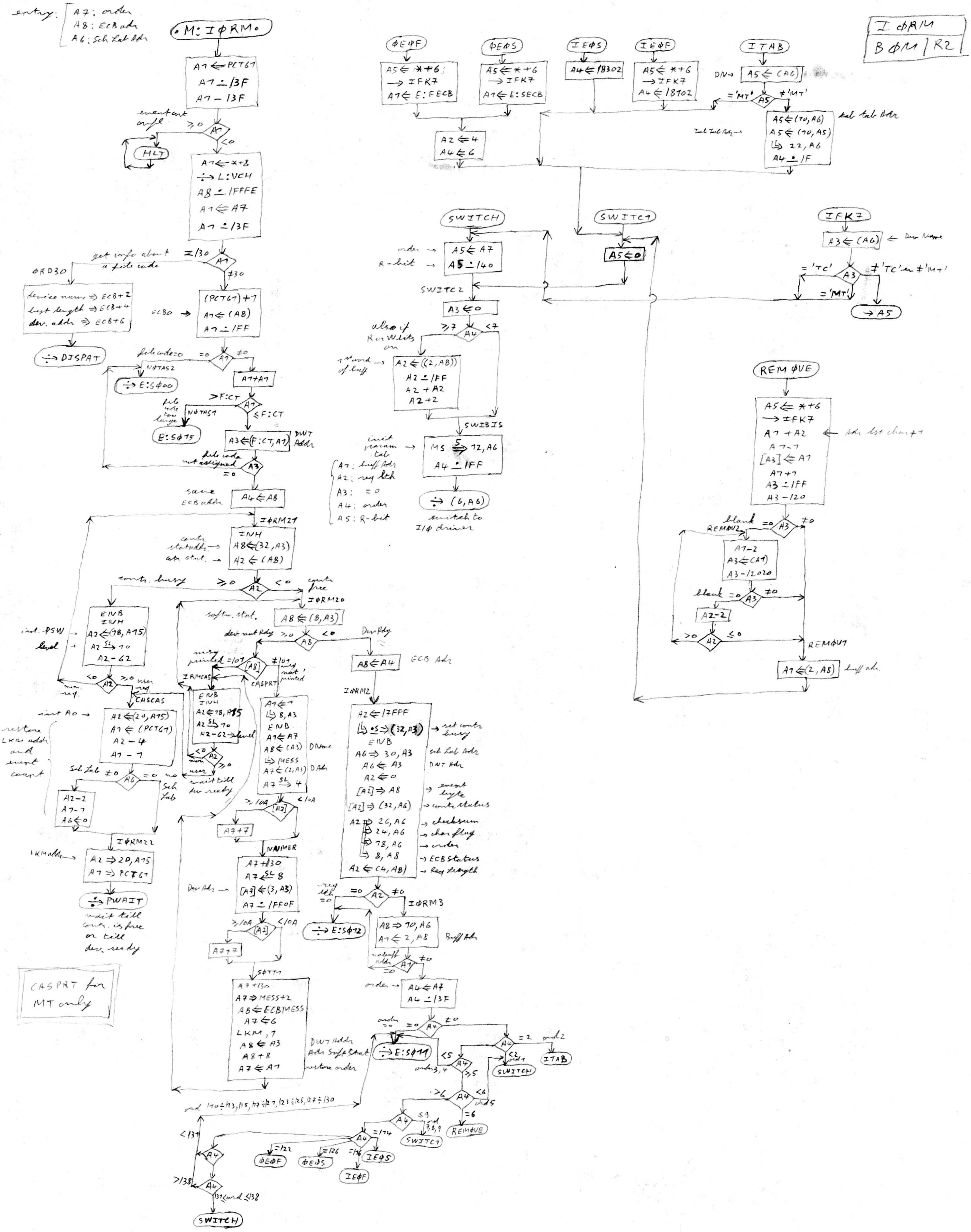
I: ASR
MSR ⇒ A15
A6 ⇐ D:WAS1
S: ASR
MSR ⇒ A15
A3 ⇐ *+8
÷> S:SST
No
Yes
÷> C:INPT
A1 ⇐ *+8
÷> L:VCH
A1 ⇐ (18, A6)
A1 − 6
≠0
A1
=0
stand write
A1 ⇐ (C4CΦ)
↳ ADRBUF
I: ASR1
A1 ⇐ (20, A6)
=0
A1
≠0
÷> R:TUR2
÷> RTUR4

DRIVER

BOOT | RZ



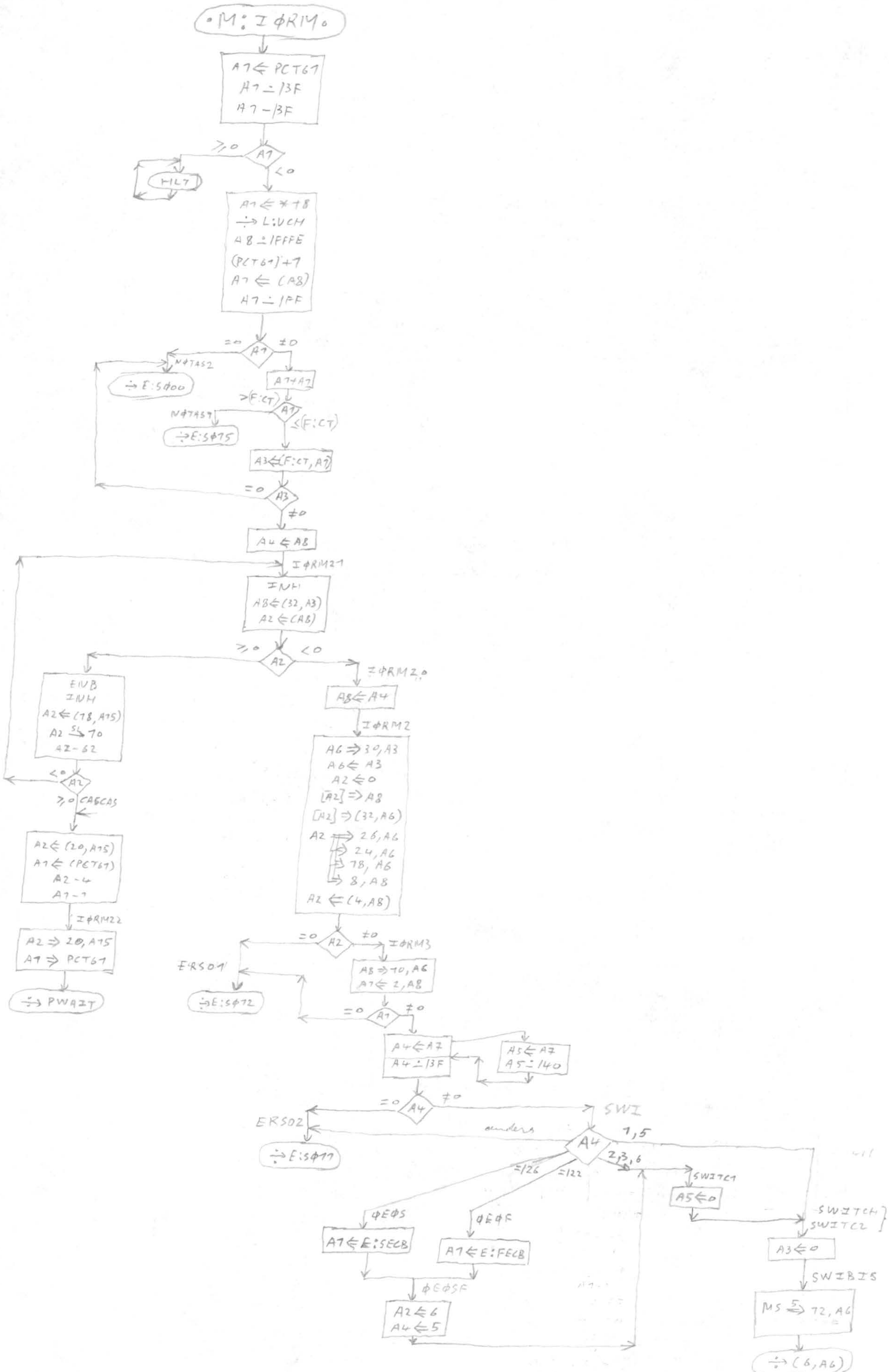




IØRM
BØM | R2
entry: A7: order
A8: ECB adr
A6: Sch Lab Adr
•M:IØRM•
A1 ⇐ PCT61
A1 ∸ /3F
A1 − 13F
event cnt ovfl
≥0
A1
<0
HLT
A1 ⇐ *+8
÷> L:VCH
A8 ∸ /FFFE
A1 ⇐ A7
A1 ∸ /3F
get info about a file code
=/30
≠30
ØRD30
device name ⇒ ECB+2
buf length ⇒ ECB+4
dev. addr ⇒ ECB+6
÷> DISPAT
(PCT61)+1
ECB0 →
A1 ⇐ (A8)
A1 ∸ /FF
filecode=0
=0
≠0
NØTAS2
÷> E:SØ00
A1+A1
>F:CT
≤F:CT
file code too large
NØTAS1
E:SØ15
A3 ⇐ (F:CT, A1)
DWT Addr
file code not assigned
A3
=0
save ECB addr
A4 ⇐ A8
IØRM21
INH
contr status addr →
A8 ⇐ (32, A3)
contr stat. →
A2 ⇐ (A8)
contr. busy
≥0
A2
<0
contr free
IØRM20
softw. stat.
A8 ⇐ (8, A3)
dev not Rdy
≥0
A8
<0
Dev Rdy
ENB
INH
int. PSW →
level →
A2 ⇐ (1B, A15)
A2 SL→ 10
A2 − 62
<0
>0 user req.
mon. req.
CASCAS
int A0 →
A2 ⇐ (20, A15)
A1 ⇐ (PCT61)
A2 − 4
A1 − 1
restore LKM addr and event count
Sch Lab ≠0
A6
=0 no Sch Lab
A2−2
A1−1
A6 ⇐ 0
IØRM22
LKM addr →
A2 ⇒ 20, A15
A1 ⇒ PCT61
÷> PWAIT
wait till contr. is free or till dev. ready
CASPRT for MT only
msg printed =101
[A8]
≠101 msg not printed
IRMCAS
CASPRT
ENB
INH
A2 ⇐ 1B, A15
A2 SL→ 10
A2−62 → level
<0 mon
A2
≥0 user
wait till dev ready
A1 ⇐ 1
↳ 8, A3
ENB
A1 ⇐ A7
A8 ⇐ (A3) DName
↳ MESS
A7 ⇐ (2, A3) DAdr
A7 SL→ 4
≥/0A
[A7]
</0A
A7+7
NAIMER
A7+/130
A7 SL⇐ 8
Dev Adr →
[A7] ⇐ (3, A3)
A7 ∸ /FF0F
≥/0A
</0A
A7+7
SØTT1
A7+/130
A7 ⇒ MESS+2
A8 ⇐ ECBIMESS
A7 ⇐ 6
LKM, 1
A8 ⇐ A3
A8+8
A7 ⇐ A1
DWT Addr
Adr Soft Stat
restore order
A8 ⇐ A4
ECB Adr
IØRM2
A2 ⇐ /7FFF
↳ •S ⇒ (32, A3) → set contr busy
ENB
A6 ⇒ 30, A3 Sch Lab Adr
A6 ⇐ A3 DWT Adr
A2 ⇐ 0
[A2] ⇒ A8 → event byte
[A2] ⇒ (32, A6) → contr status
A2 ⇒ 26, A6 → checksum
↳ 24, A6 → char flag
⇒ 18, A6 → order
↳ 8, A8 → ECB Status
A2 ⇐ (4, A8) → Req Length
req lth =0
=0
A2
≠0
IØRM3
÷> E:SØ12
A8 ⇒ 10, A6
A1 ⇐ 2, A8
Buff Adr
no buff addr
A1
=0
≠0
order →
A4 ⇐ A7
A4 ∸ /3F
order =0
=0
A4
≠0
÷> E:SØ11
A4
=2 ord2
ITAB
<2 ord1
SWITCH
<5
A4
order 3,4
≥5
>6
A4
<6
ord5
=6
REMØVE
A4
≤9 ord 7,8,9
SWITC1
ord /10÷/13, /15, /17÷/21, /23÷/25, /27÷/30
A4
=/14
=/22
ØEØF
=/26
ØEØS
=/16
IEØF
IEØS
</37
A4
>/38
A4
/37≤ord≤/38
SWITCH
ØEØF
A5 ⇐ *+6
→ IFK7
A1 ⇐ E:FECB
ØEØS
A5 ⇐ *+6
→ IFK7
A1 ⇐ E:SECB
IEØS
A4 ⇐ /8302
IEØF
A5 ⇐ *+6
→ IFK7
A4 ⇐ /8102
ITAB
DN →
A5 ⇐ (A6)
='MT'
A5
≠'MT'
A5 ⇐ (10, A6)
Lab tab Adr
A5 ⇐ (10, A5)
↳ 22, A6
A4 ∸ /F
A2 ⇐ 4
A4 ⇐ 6
SWITCH
order →
A5 ⇐ A7
R-bit →
A5 ∸ /40
SWITC2
A3 ⇐ 0
≥7
A4
<7
also if R or W bits on
1st word of buff →
A2 ⇐ ((2, A8))
A2 ∸ /FF
A2 + A2
A2+2
SWIBIS
init param tab →
MS 5⇒ 12, A6
A4 ∸ /FF
A1: buff Adr
A2: req lth
A3: =0
A4: order
A5: R-bit
÷> (6, A6)
switch to I/Ø driver
SWITC1
A5 ⇐ 0
IFK7
A3 ⇐ (A6) ← Dev Name
='TC'
A3
≠'TC' and ≠'MT'
='MT'
→ A5
REMØVE
A5 ⇐ *+6
→ IFK7
A1 + A2 ← Adr lst char+1
A1−1
[A3] ⇐ A1
A1+1
A3 ∸ /FF
A3 − /20
blank =0
A3
≠0
REMØV2
A1−2
A3 ⇐ (A1)
A3 − /2020
blank =0
A3
≠0
A2−2
>0
A2
≤0
REMØV1
A1 ⇐ (2, A8) buff adr

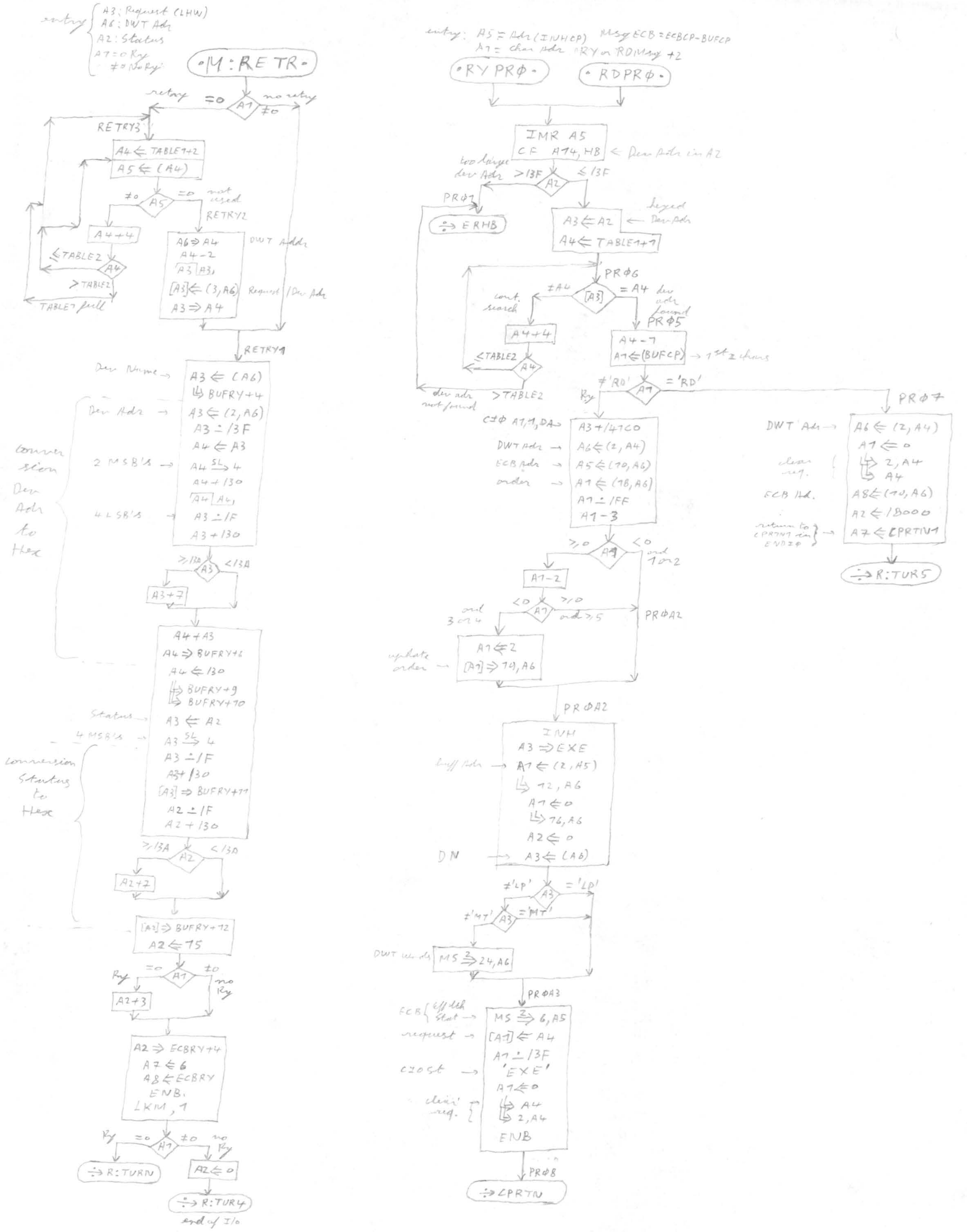
IΦRM

DIBΦM



M: IΦRM
A7 ⇐ PCT61
A7 ∸ 13F
A7 − 3F
HLT
A1 ⇐ * + 8
L:UCH
A8 ∸ 1FFFE
(PCT61) + 1
A7 ⇐ (A8)
A7 ∸ 1FF
A7
E:S000
E:S075
A3 ⇐ (F:CT, A7)
A3
A4 ⇐ A8
IΦRM21
INH
A8 ⇐ (32, A3)
A2 ⇐ (A8)
A2
IΦRM20
A8 ⇐ A4
IΦRM2
ENB
INH
A2 ⇐ (18, A15)
A2 − 62
A2 ⇐ (20, A15)
A1 ⇐ (PCT61)
IΦRM22
A2 ⇒ 20, A15
A1 ⇒ PCT61
PWAIT
A6 ⇒ 30, A3
A6 ⇐ A3
A2 ⇐ 0
A2 ⇐ (4, A8)
IΦRM3
ERS01
E:SΦ72
A5 ⇒ 10, A6
A1 ⇐ 2, A8
A4 ⇐ A7
A4 ∸ 13F
A5 ⇐ A7
A5 ∸ 140
A4
SWI
ERS02
E:SΦ77
1, 5
2, 3, 6
ΦEΦS
ΦEΦF
A1 ⇐ E:SECB
A1 ⇐ E:FECB
ΦEΦSF
A2 ⇐ 6
A4 ⇐ 5
SWITC1
A5 ⇐ 0
SWITCH
SWITC2
A3 ⇐ 0
SWIBIS
(6, A6)

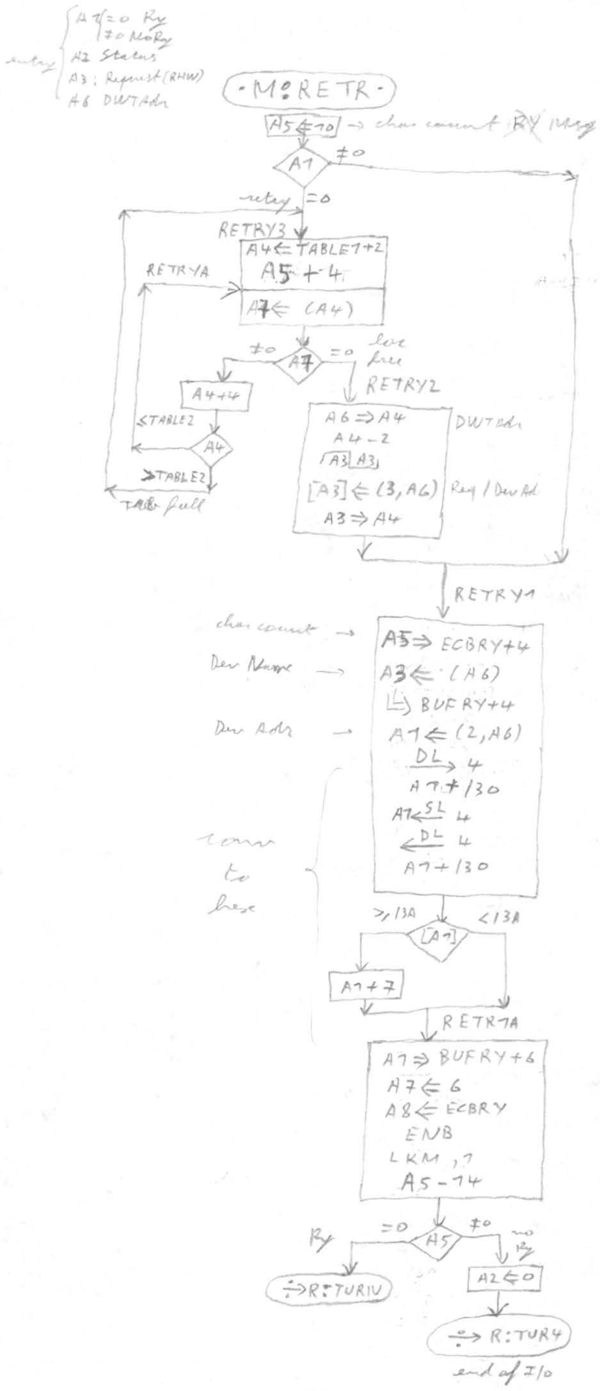
M:RETR | BØM R2



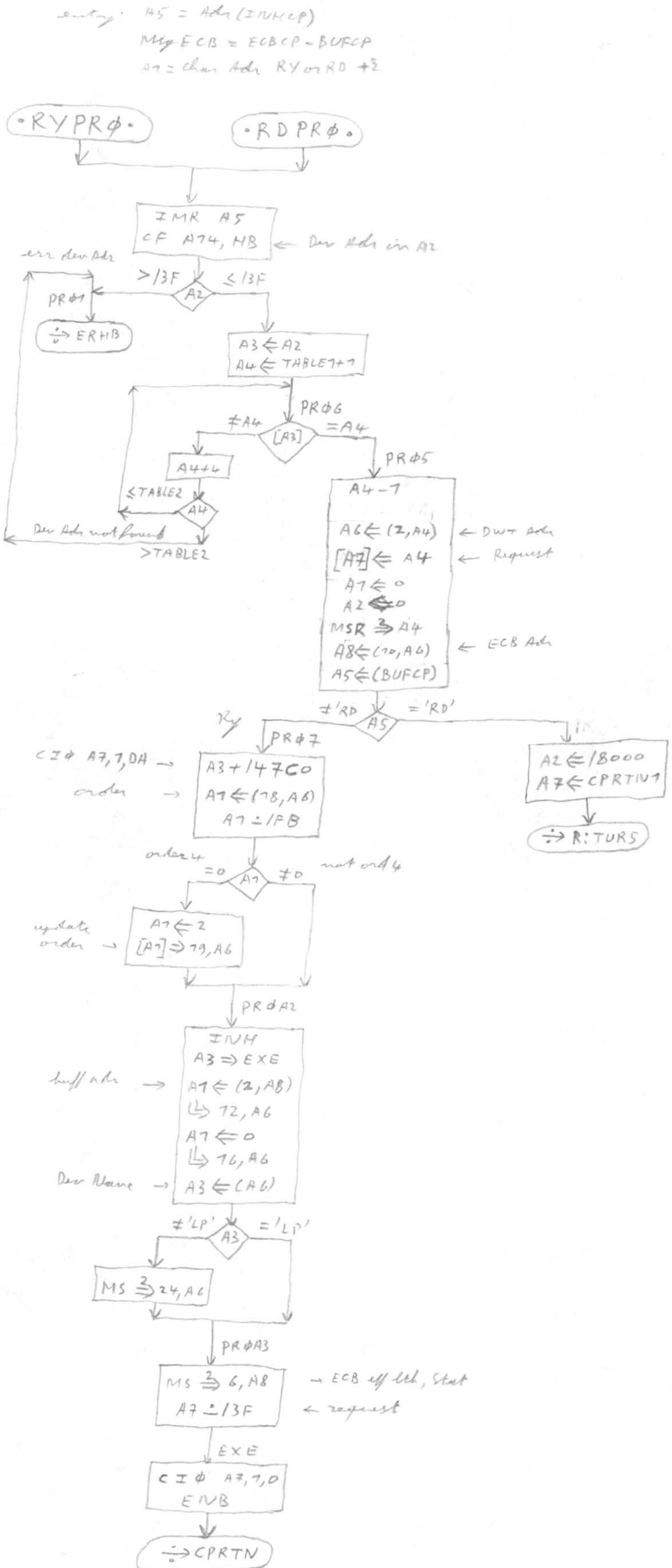
M:RETR
A3: Request (LHW)
A6: DWT Adr
A2: Status
RETRY3
A4 ⇐ TABLE1+2
A5 ⇐ (A4)
RETRY2
A6 ⇒ A4
A4−2
A3 ⇒ A4
TABLE1 full
RETRY1
A3 ⇐ (A6)
BUFRY+4
A3 ⇐ (2,A6)
A3 ∸ /3F
A4 ⇐ A3
A4 ∸ /F
A4 + /30
A3 ∸ /F
A3 + /30
A3+7
A4+A3
A4 ⇒ BUFRY+6
A4 ⇐ /30
BUFRY+9
BUFRY+10
A3 ⇐ A2
A3 ∸ /F
A3 + /30
BUFRY+11
A2 ∸ /F
A2 + /30
A2+7
BUFRY+12
A2 ⇐ 15
A2+3
A2 ⇒ ECBRY+4
A7 ⇐ 6
A8 ⇐ ECBRY
ENB.
LKM, 1
A2 ⇐ 0
R:TURN
R:TUR4
end of I/O
RYPRØ
RDPRØ
IMR A5
CF A14, HB
PRØ1
ERHB
A3 ⇐ A2
A4 ⇐ TABLE1+1
PRØ6
PRØ5
A4+4
A4−1
A1 ⇐ (BUFCP)
TABLE2
dev adr not found
PRØ7
A3 + /41C0
A6 ⇐ (2,A4)
A5 ⇐ (10,A6)
A1 ⇐ (18,A6)
A1 ∸ /FF
A1−3
A1−2
PRØA2
A1 ⇐ 2
INH
A3 ⇒ EXE
A1 ⇐ (2,A5)
12, A6
A1 ⇐ 0
16, A6
A2 ⇐ 0
A3 ⇐ (A6)
MS ⇒ 24, A6
PRØA3
MS ⇒ 6, A5
[A1] ⇐ A4
A1 ∸ /3F
'EXE'
A1 ⇐ 0
A4
2, A4
ENB
PRØ8
LPRTN
A6 ⇐ (2,A4)
A1 ⇐ 0
2, A4
A4
A8 ⇐ (10,A6)
A2 ⇐ /B000
A7 ⇐ LPRTIV1
R:TUR5

M:RETR

DIBØM

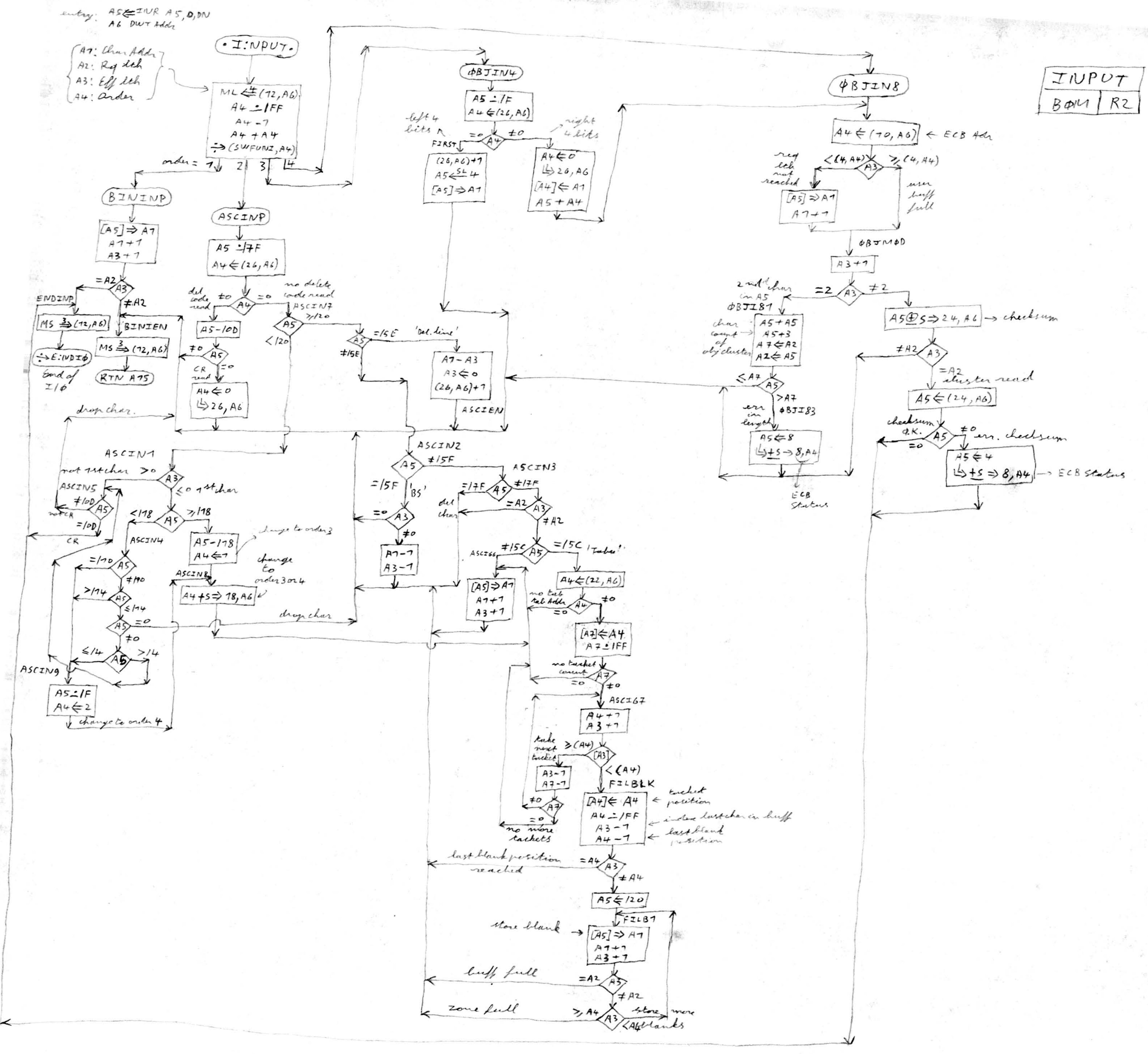


entry
A1 ≠0 Ry
=0 NoRy
A2 Status
A3 ; Request (RHW)
A6 DWTAdr
·M:RETR·
A5 ⇐ 10 → char count RY Msg
A1
≠0
=0
retry
RETRY3
A4 ⇐ TABLE1+2
A5+4
RETRY4
A7 ⇐ (A4)
A7
≠0
=0
free
A4+4
STABLE2
A4
>TABLE2
Table full
RETRY2
A6 ⇒ A4
A4-2
A3 A3
[A3] ⇐ (3, A6)
A3 ⇒ A4
DWTAdr
Req / Dev Ad
RETRY1
char count →
Dev Name →
Dev Adr →
A5 ⇒ ECBRY+4
A3 ⇐ (A6)
BUFRY+4
A1 ⇐ (2, A6)
DL 4
A1 ∸ /30
A1 SL 4
DL 4
A1 + /30
come to here
>/13A
</13A
[A1]
A1+7
RETRY1A
A1 ⇒ BUFRY+6
A7 ⇐ 6
A8 ⇐ ECBRY
ENB
LKM ,1
A5 - 14
Ry
=0
≠0
A5
A2 ⇐ 0
÷ R:TURN
÷ R:TUR4
end of I/O



entry: A5 = Adr (INHCP)
Msg ECB = ECBCP - BUFCP
A1 = char Adr RY or RD +2
·RYPRØ·
·RDPRØ·
IMR A5
CF A14, HB
← Dev Adr in A2
err dev Adr
>/13F
≤/3F
A2
PRØ1
÷ ERHB
A3 ⇐ A2
A4 ⇐ TABLE1+1
PRØ6
≠A4
=A4
[A3]
A4+4
STABLE2
A4
Dev Adr not found
>TABLE2
PRØ5
A4-1
A6 ⇐ (2, A4)
[A7] ⇐ A4
A1 ⇐ 0
A2 ⇐ 0
MSR ⇒ A4
A8 ⇐ (10, A6)
A5 ⇐ (BUFCP)
← DWT Adr
← Request
← ECB Adr
≠'RD
='RD'
A5
Ry
PRØ7
CIØ A7,1,0A →
order →
A3 + /47C0
A1 ⇐ (18, A6)
A1 ∸ /FB
A2 ⇐ /8000
A7 ⇐ CPRTN1
÷ R:TUR5
order 4
=0
≠0
not ord 4
A1
update order →
A1 ⇐ 2
[A1] ⇒ 19, A6
PRØA2
INH
A3 ⇒ EXE
buff adr →
A1 ⇐ (2, A8)
12, A6
A1 ⇐ 0
16, A6
Dev Name →
A3 ⇐ (A6)
≠'LP'
='LP'
A3
MS ⇒ 24, A6
PRØA3
MS ⇒ 6, A8
A7 ∸ /3F
→ ECB eff lth, Stat
← request
EXE
CIØ A7,1,0
ENB
÷ CPRTN

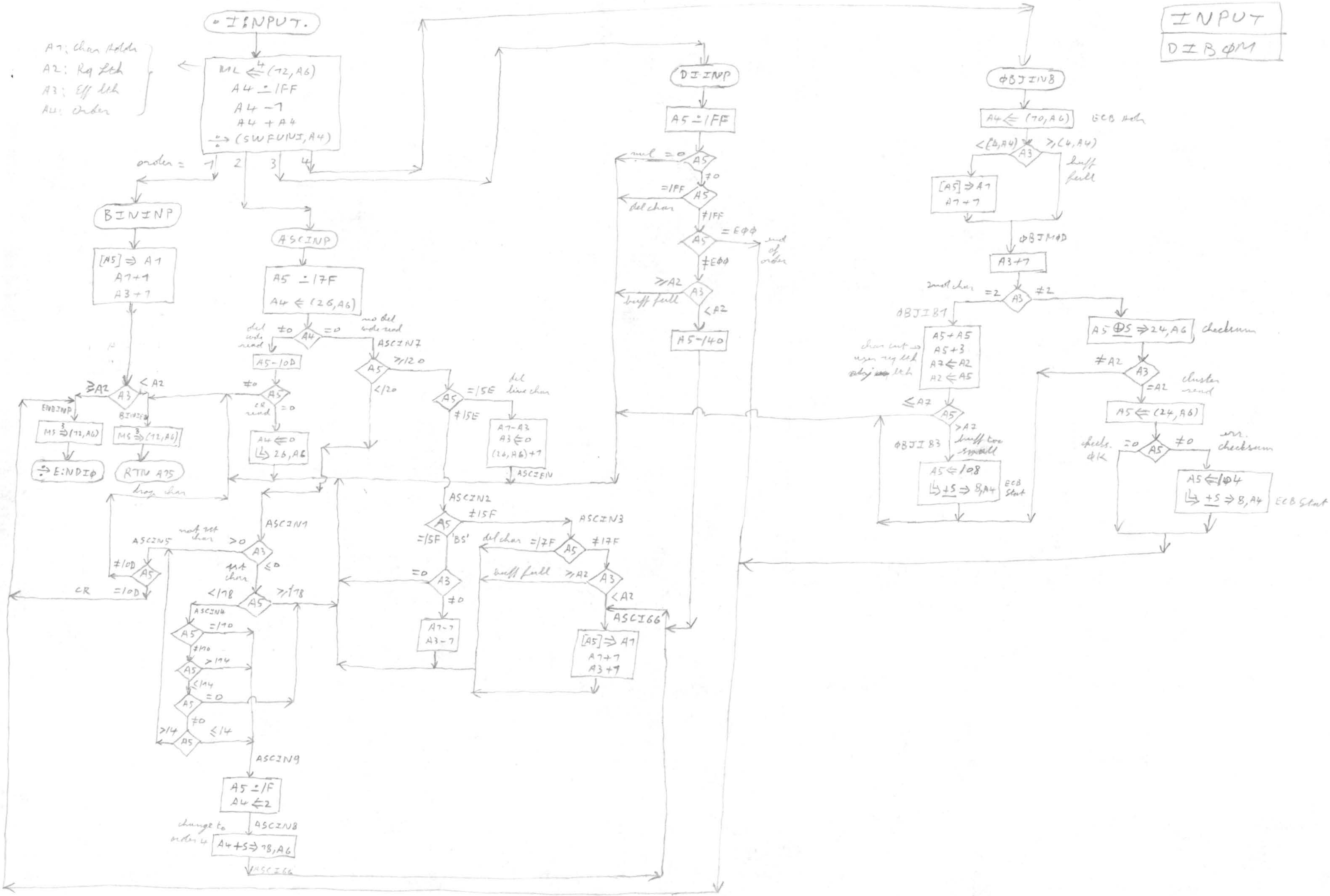
INPUT	
BΦM	R2



entry: A5 ⇐ INR A5,0,DN
A6 DWT Addr
A1: Char Addr
A2: Req lth
A3: Eff lth
A4: Order
•I:INPUT•
ML ⇐ (12,A6)
A4 ∸ /FF
A4 − 1
A4 + A4
÷> (SWFUNZ,A4)
order = 1 2 3 4
BINIMP
ASCINP
ΦBJIN4
ΦBJIN8
ECB Adr
ENDINP
BINIEN
RTN A15
End of I/Φ
ASCIN1
ASCIN2
ASCIN3
ASCIN4
ASCIN5
ASCIN7
ASCIN8
ASCIN9
ASCI66
ASCI67
ASCIEN
FILBLK
FILB1
ΦBJMΦD
ΦBJI81
ΦBJI83
drop char
change to order 3 or 4
change to order 4
ECB Status
checksum
err. checksum
cluster read
last blank position reached
buff full
zone full
store blank
store more blanks
no more tabs
no tab tab Addr

INPUT

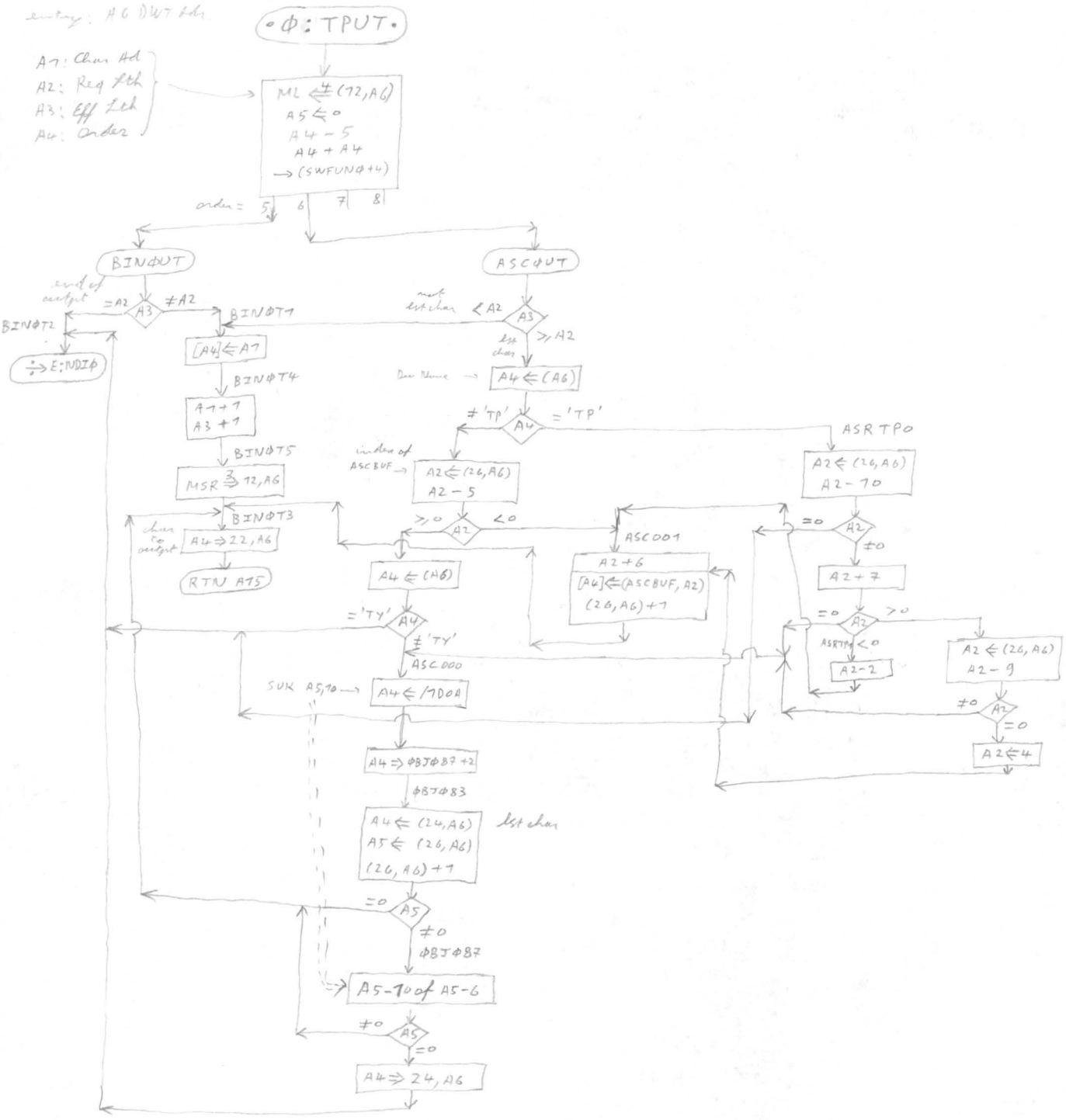
DIBØM



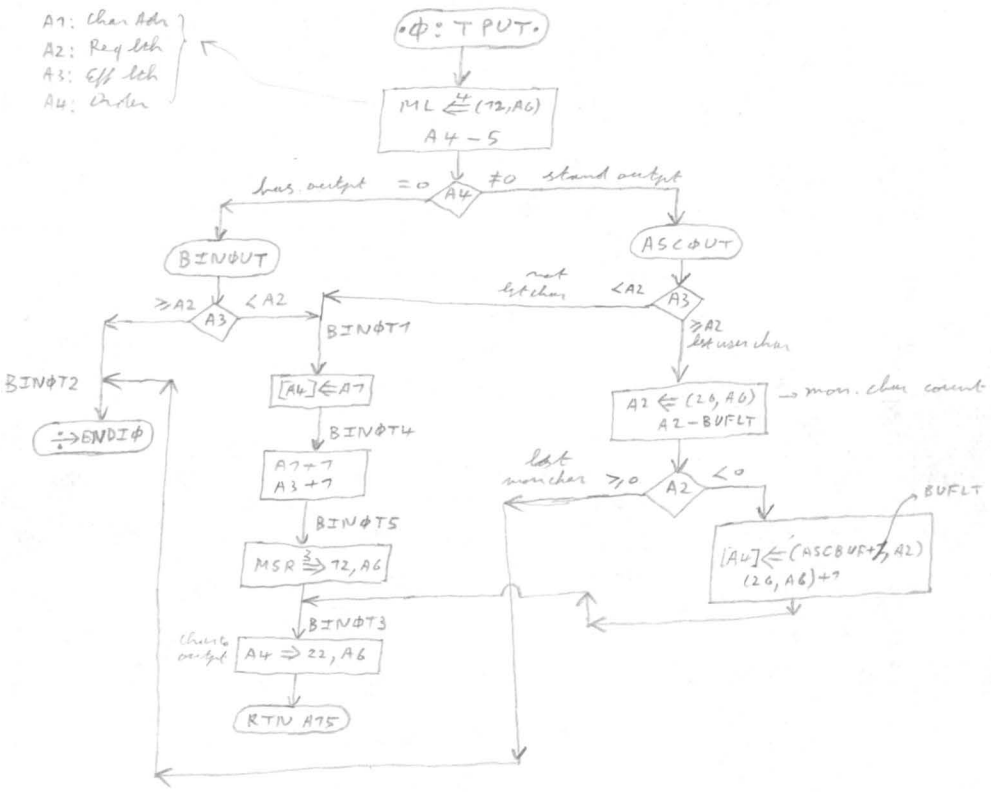
I:INPUT.
A1: Char holder
A2: Rq Lth
A3: Eff lth
A4: order
A4 ∸ IFF
A4 −1
A4 + A4
(SWFUNI, A4)
order = 1 2 3 4
BININP
[A5] ⇒ A1
A1+1
A3+1
ASCINP
A5 ∸ 17F
A4 ⇐ (26, A6)
ASCIN7
ASCIN1
ASCIN2
ASCIN3
ASCIN4
ASCIN5
ASCIN6
ASCIN8
ASCIN9
A5 ∸ 1F
A4 ⇐ 2
change to order 4
A4 +S ⇒ 18, A6
A1 − A2
A3 ⇐ 0
(26, A6) + 1
A1 − 1
A3 − 1
ASCIGG
[A5] ⇒ A1
A1 + 1
A3 + 1
buff full
del char
CR
E:NDIØ
DIINP
A5 ∸ IFF
A5 − 140
ØBJIN8
A4 ⇐ (10, A6)
ECB Adr
ØBJMØD
A3 + 1
ØBJI81
A5 + A5
A5 + 3
A1 ⇐ A2
A2 ⇐ A5
ØBJI83
buff too small
A5 ⇐ 108
ECB Stat
A5 ⊕S ⇒ 24, A6
checksum
cluster read
A5 ⇐ (24, A6)
err. checksum
A5 ⇐ Ø4
ECB Stat

entry: A6 DWT Adr

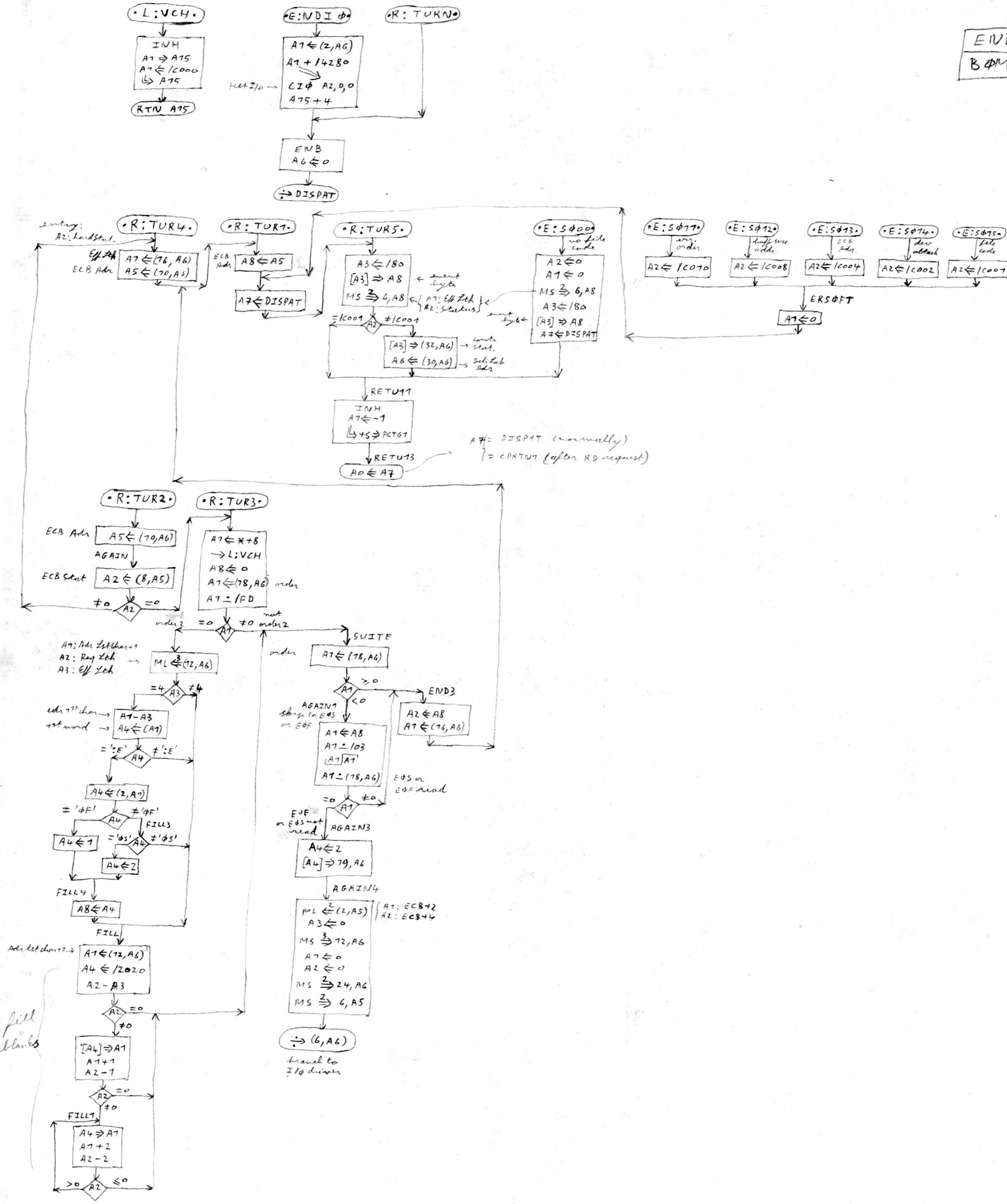
A1: Char Ad
A2: Req Lth
A3: Eff Lth
A4: Order



ΦUTPUT
DIBΦM



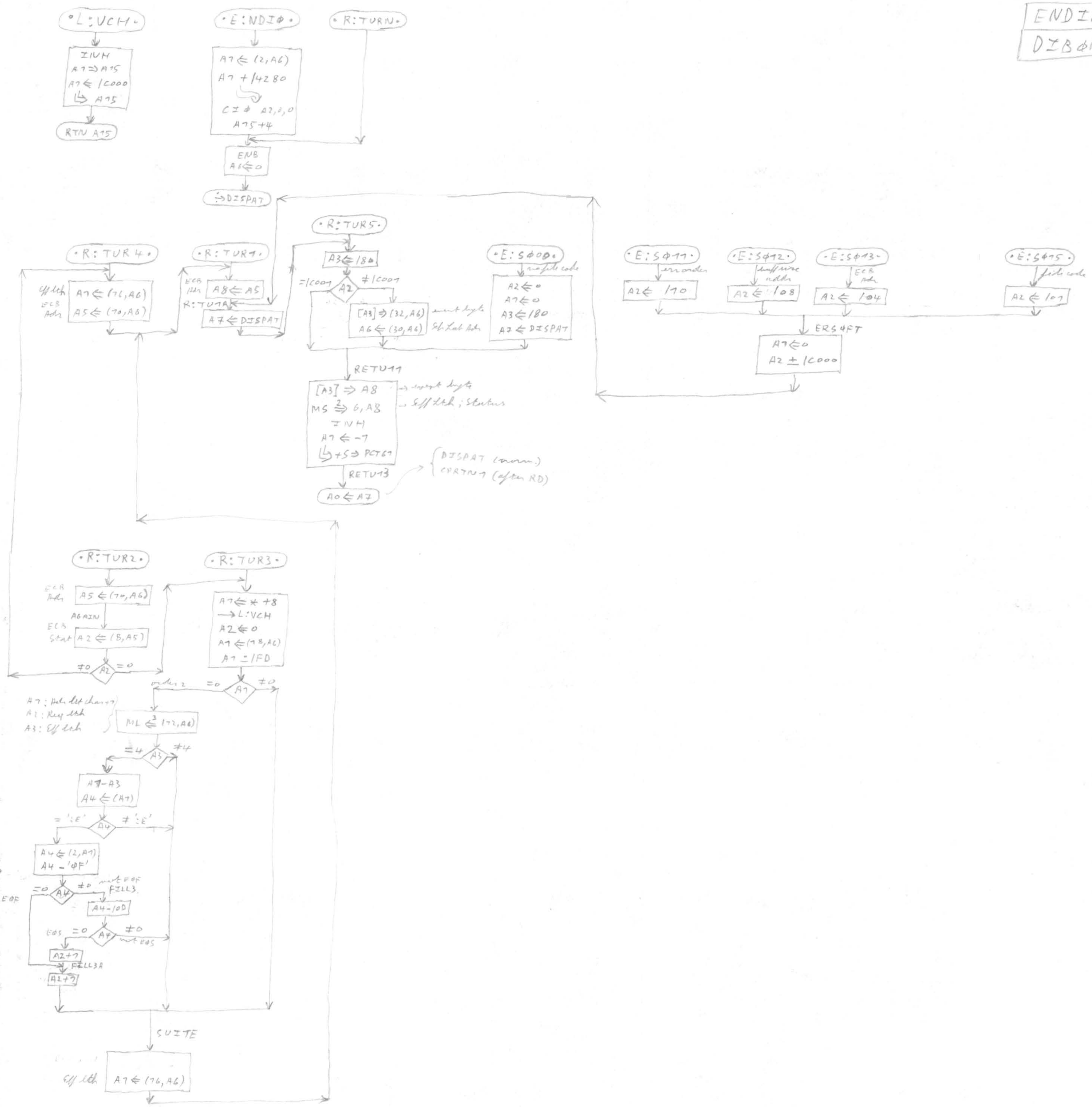
ENDIØ
BØM R2



L:VCH
INH
A1 ⇒ A15
A1 ⇐ /C000
A15
RTN A15
E:NDIØ
A1 ⇐ (2,A6)
A1 + /4280
CIØ A2,0,0
A15 + 4
R:TURN
ENB
A6 ⇐ 0
DISPAT
R:TUR4
A7 ⇐ (76,A6)
A5 ⇐ (70,A6)
ECB Adr
R:TUR1
A8 ⇐ A5
A7 ⇐ DISPAT
R:TUR5
A3 ⇐ 180
[A3] ⇒ A8
MS ⇒ 4,A8
=/C001
≠/C001
[A3] ⇒ (32,A6)
A6 ⇐ (30,A6)
E:SØ00
no file code
A2 ⇐ 0
A1 ⇐ 0
MS ⇒ 6,A8
A3 ⇐ 180
[A3] ⇒ A8
A7 ⇐ DISPAT
E:SØ11
E:SØ12
E:SØ13
E:SØ14
E:SØ15
A2 ⇐ /C010
A2 ⇐ /C008
A2 ⇐ /C004
A2 ⇐ /C002
A2 ⇐ /C001
ERSØFT
A1 ⇐ 0
RETU11
INH
A1 ⇐ -1
+5 ⇒ PCT61
RETU13
A0 ⇐ A7
A7 = DISPAT (normally)
= CPRTN1 (after RD request)
R:TUR2
ECB Adr
A5 ⇐ (70,A6)
AGAIN
ECB Stat
A2 ⇐ (8,A5)
≠0
=0
R:TUR3
A1 ⇐ *+8
→ L:VCH
A8 ⇐ 0
A1 ⇐ (78,A6) order
A1 ∸ /FD
=0
≠0
SUITE
A1 ⇐ (76,A6)
≥0
<0
END3
A2 ⇐ A8
A1 ⇐ (76,A6)
AGAIN1
A1 ⇐ A8
A1 ∸ /03
A1 ∸ (78,A6)
EØS or EØF read
AGAIN3
A4 ⇐ 2
[A4] ⇒ 79,A6
AGAIN4
ML ⇐ (2,A5)
A1: ECB+2
A2: ECB+4
A3 ⇐ 0
MS ⇒ 72,A6
A1 ⇐ 0
A2 ⇐ 0
MS ⇒ 24,A6
MS ⇒ 6,A5
÷ (6,A6)
branch to I/Ø driver
A1: Adr Lstchar+1
A2: Req Lth
A3: Eff Lth
ML ⇐ (72,A6)
=4
≠4
A1 - A3
A4 ⇐ (A1)
=':E'
≠':E'
A4 ⇐ (2,A1)
='ØF'
≠'ØF'
FILL3
A4 ⇐ 1
='ØS'
≠'ØS'
A4 ⇐ 2
FILL4
A8 ⇐ A4
FILL
A1 ⇐ (72,A6)
A4 ⇐ /2020
A2 - A3
fill blanks
[A4] ⇒ A1
A1 + 1
A2 - 1
FILL1
A4 ⇒ A1
A1 + 2
A2 - 2
>0
≤0

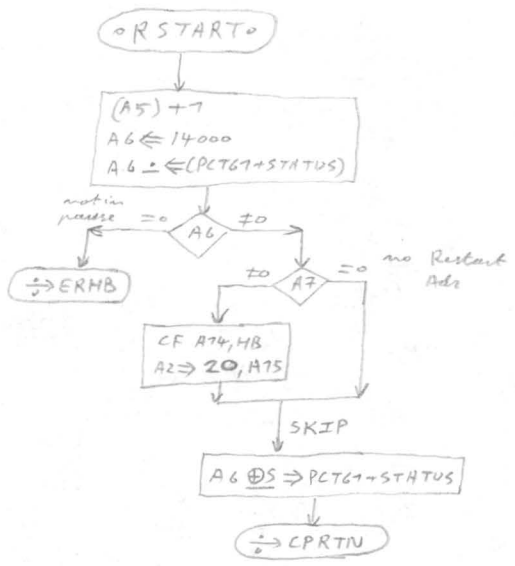
ENDIØ

DIBØM

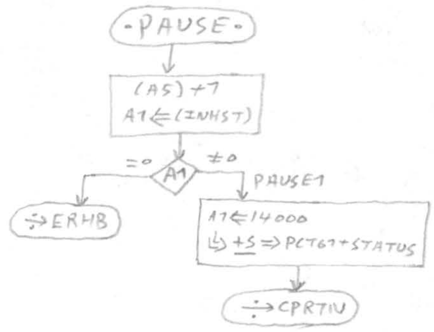


L:VCH
INH
A7 ⇒ A15
A7 ⇐ /C000
A15
RTN A15
E:NDIØ
A7 ⇐ (2,A6)
A7 + /4280
CIØ A2,0,0
A15+4
R:TURN
ENB
A1 ⇐ 0
DISPAT
R:TUR5
A3 ⇐ 180
A2
=/C001
≠/C001
[A3] ⇒ (32,A6)
A6 ⇐ (30,A6)
E:SØ00
A2 ⇐ 0
A7 ⇐ 0
A3 ⇐ 180
A7 ⇐ DISPAT
E:SØ11
A2 ⇐ /10
E:SØ12
A2 ⇐ /08
E:SØ13
A2 ⇐ /04
E:SØ15
A2 ⇐ /01
ERSØFT
A7 ⇐ 0
A2 ± /C000
R:TUR4
A7 ⇐ (76,A6)
A5 ⇐ (70,A6)
R:TUR1
A8 ⇐ A5
A7 ⇐ DISPAT
RETU11
[A3] ⇒ A8
MS ⇒ 6,A8
INH
A7 ⇐ -7
+5 ⇒ PCT67
RETU13
A0 ⇐ A7
DISPAT
R:TUR2
A5 ⇐ (70,A6)
AGAIN
A2 ⇐ (B,A5)
≠0
=0
R:TUR3
A7 ⇐ * +8
L:VCH
A2 ⇐ 0
A1 ⇐ (78,A6)
A1 ± /FD
=0
≠0
ML ⇐ (72,A6)
=4
≠4
A7-A3
A4 ⇐ (A7)
=':E'
≠':E'
A4 ⇐ (2,A7)
A4 - 'ØF'
EOF
FILL3
A4-/0D
EOS
A2+7
FILL32
SUITE
A7 ⇐ (76,A6)

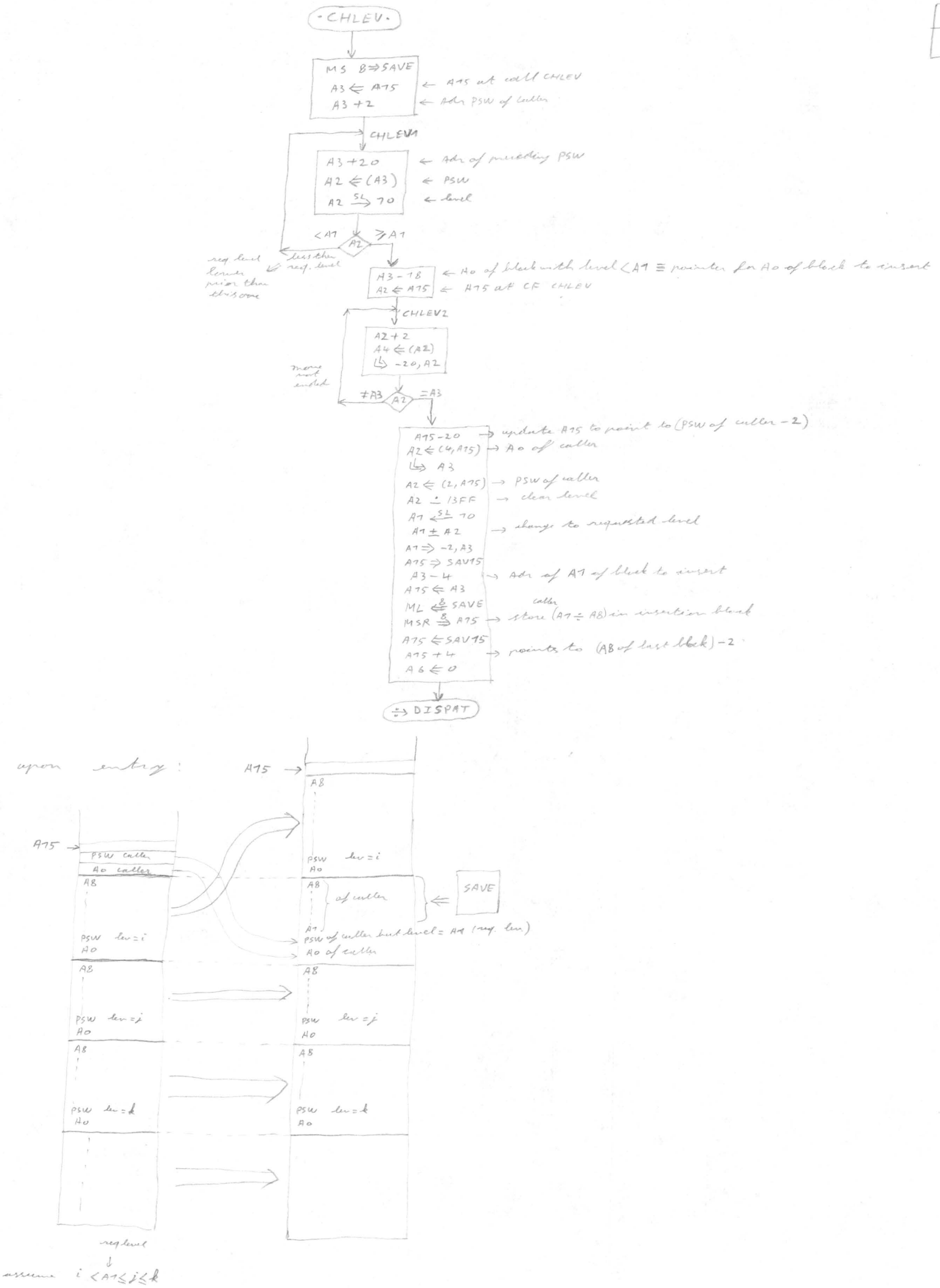
RSTART
DIBΦIM



PAUSE
DIBØIM

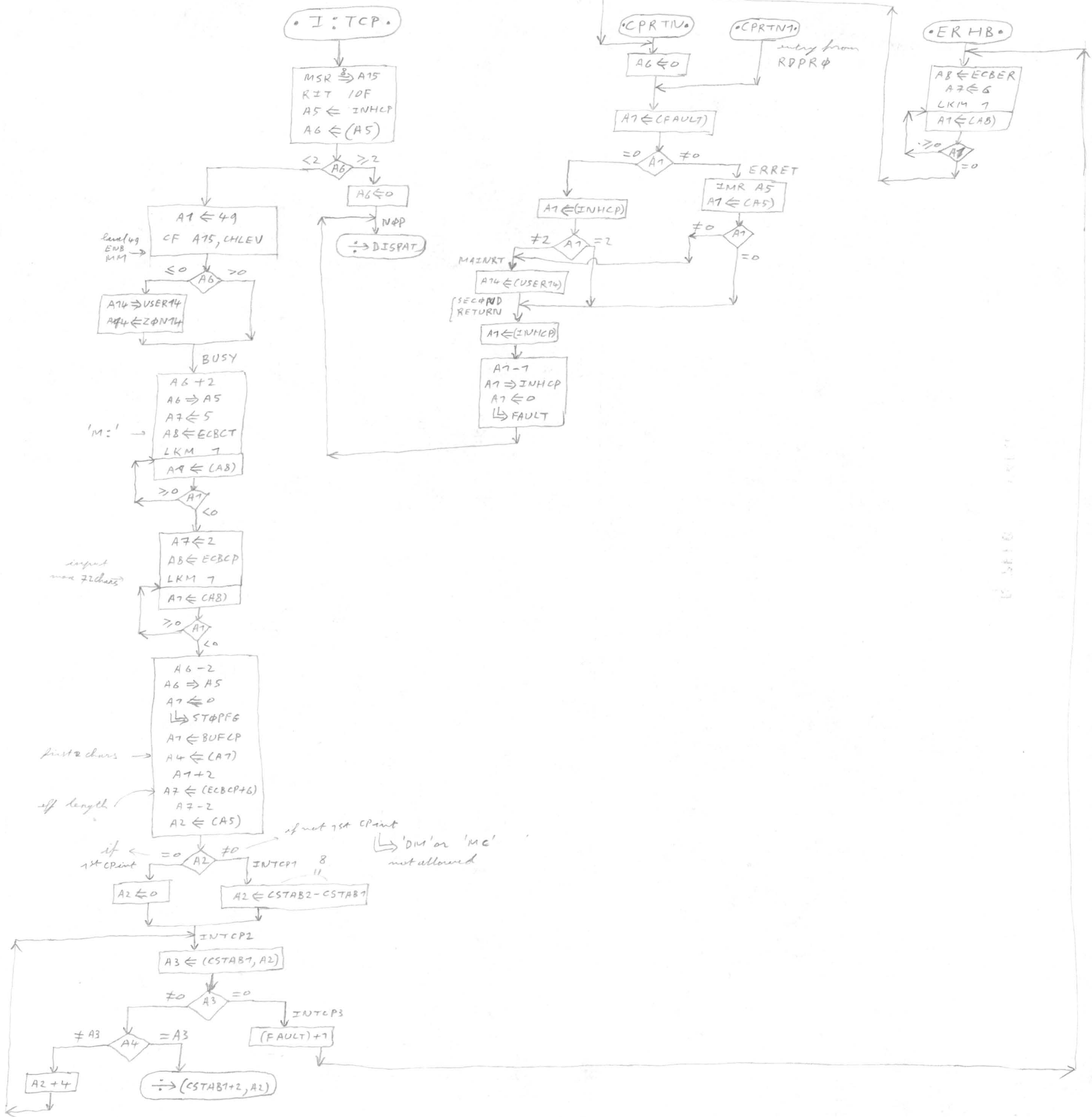


CHLEV	
BOM	R2


·CHLEV·
MS B⇒SAVE
A3 ⇐ A15
A3 +2
← A15 at call CHLEV
← Adr PSW of caller
CHLEV1
A3 +20
A2 ⇐ (A3)
A2 SL⇒ 10
← Adr of preceding PSW
← PSW
← level
<A1
A2
≥A1
req level lower than this one
less than req. level
A3 -18
A2 ⇐ A15
← A0 of block with level <A1 ≡ pointer for A0 of block to insert
← A15 at CF CHLEV
CHLEV2
A2 +2
A4 ⇐ (A2)
↳ -20, A2
more not ended
≠A3
A2
=A3
A15 -20
A2 ⇐ (4, A15)
↳ A3
A2 ⇐ (2, A15)
A2 ∴ 13FF
A1 SL⇐ 10
A1 ± A2
A1 ⇒ -2, A3
A15 ⇒ SAV15
A3 -4
A15 ⇐ A3
ML ⇐ SAVE
MSR ⇒ A15
A15 ⇐ SAV15
A15 +4
A6 ⇐ 0
→ update A15 to point to (PSW of caller -2)
→ A0 of caller
→ PSW of caller
→ clear level
→ change to requested level
→ Adr of A1 of block to insert
→ store (A1 ÷ A8) in insertion block
→ points to (A8 of last block) -2
÷→ DISPAT

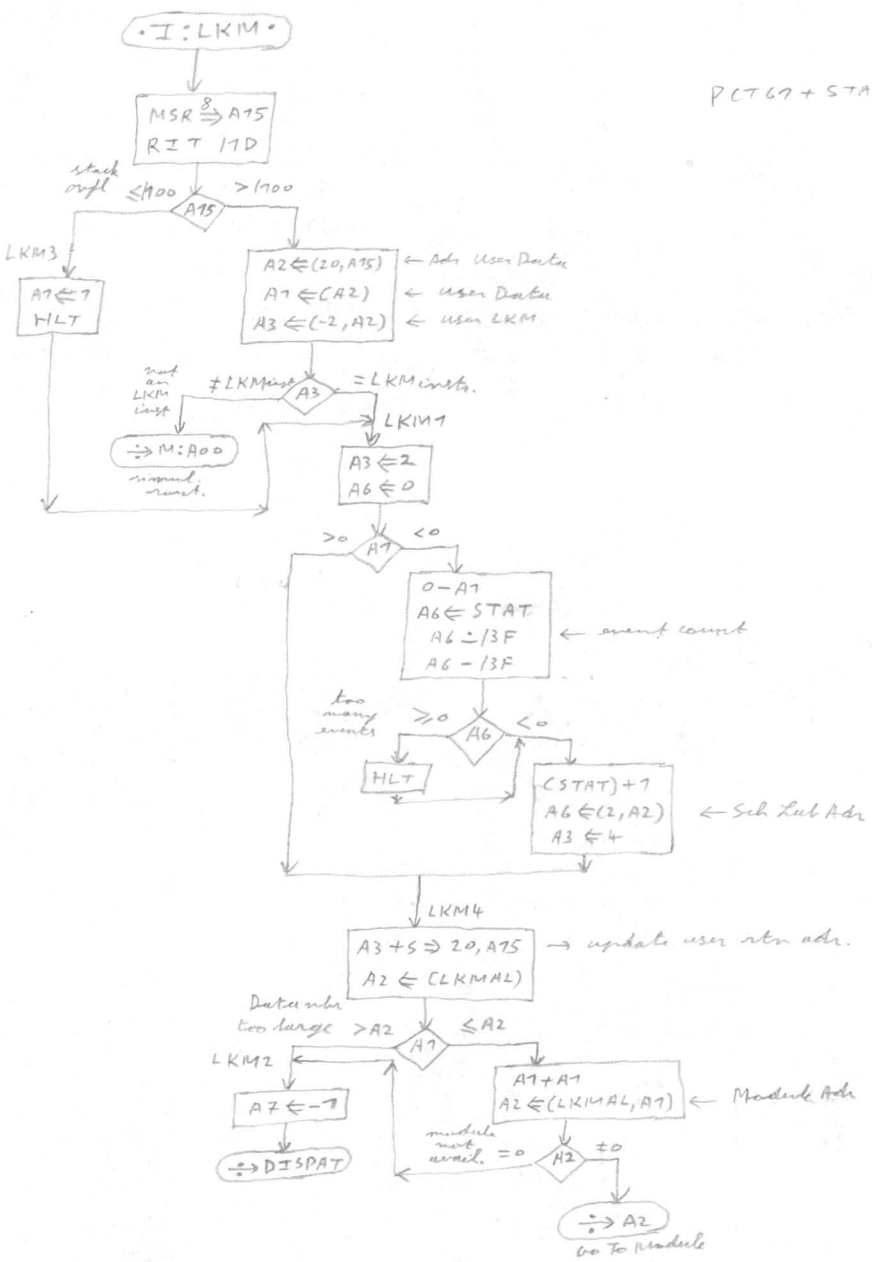
upon entry :
A15
PSW caller
A0 caller
A8
PSW lev = i
A0
PSW lev = j
A0
PSW lev = k
A0
A15
A8
PSW lev = i
A0
A8
of caller
SAVE
A1
PSW of caller but level = A4 (req. lev.)
A0 of caller
A8
PSW lev = j
A0
A8
PSW lev = k
A0
req level
assume i <A1≤ j ≤ k

INTCP
BØ14 | R2

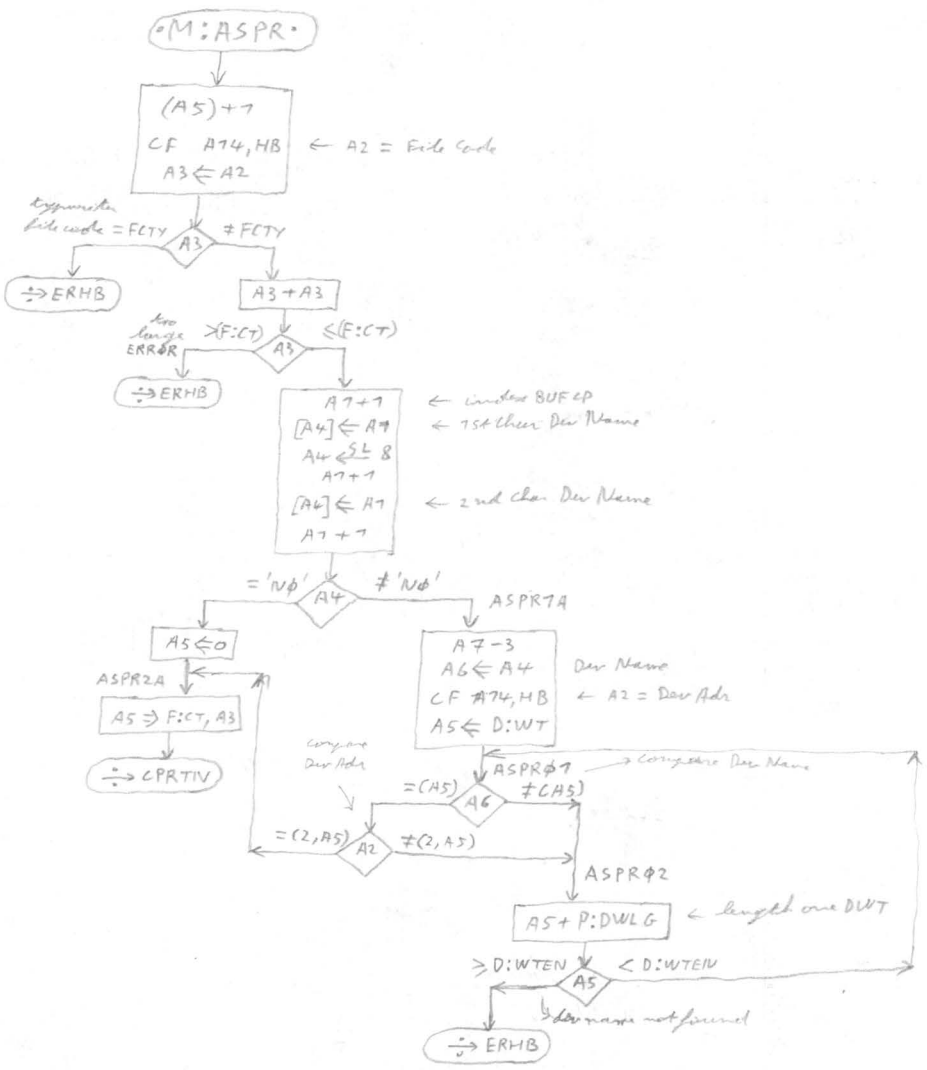


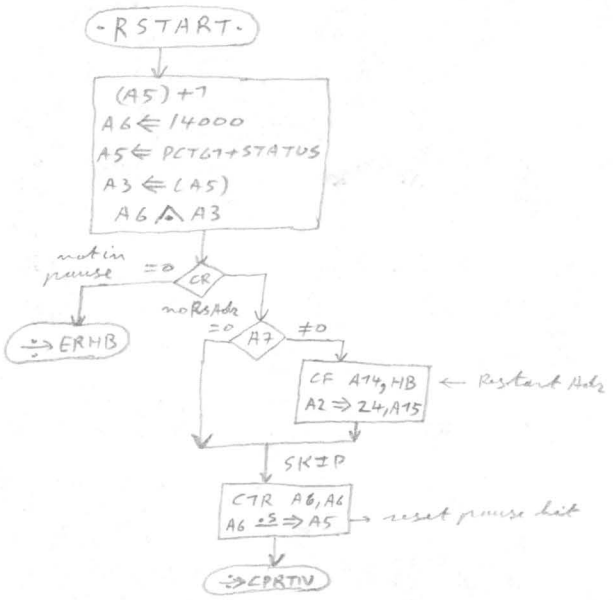
I:LKM | BØM | R2

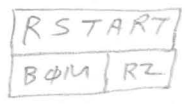
PCT67 + STATUS ≡ STAT

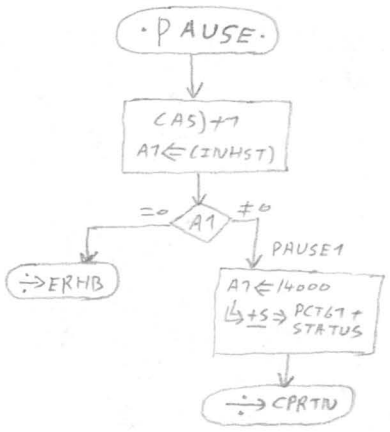


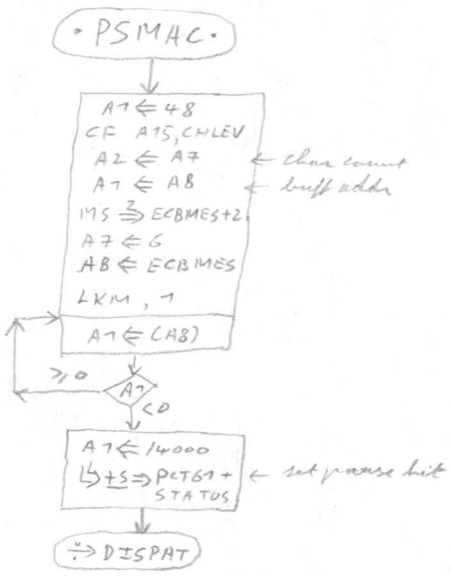
M:ASPR | BØIM | R2




-RSTART-
(A5)+7
A6 ⇐ 14000
A5 ⇐ PCTGT+STATUS
A3 ⇐ (A5)
A6 ∧ A3
notin pause =0
CR
noRsAdr =0
A7
≠0
→ ERHB
CF A14, HB
A2 ⇒ 24, A15
← Restart Hdr
SKIP
CTR A6, A6
A6 .S ⇒ A5
→ reset pause bit
→ CPRTIV


RSTART
BOM
R2

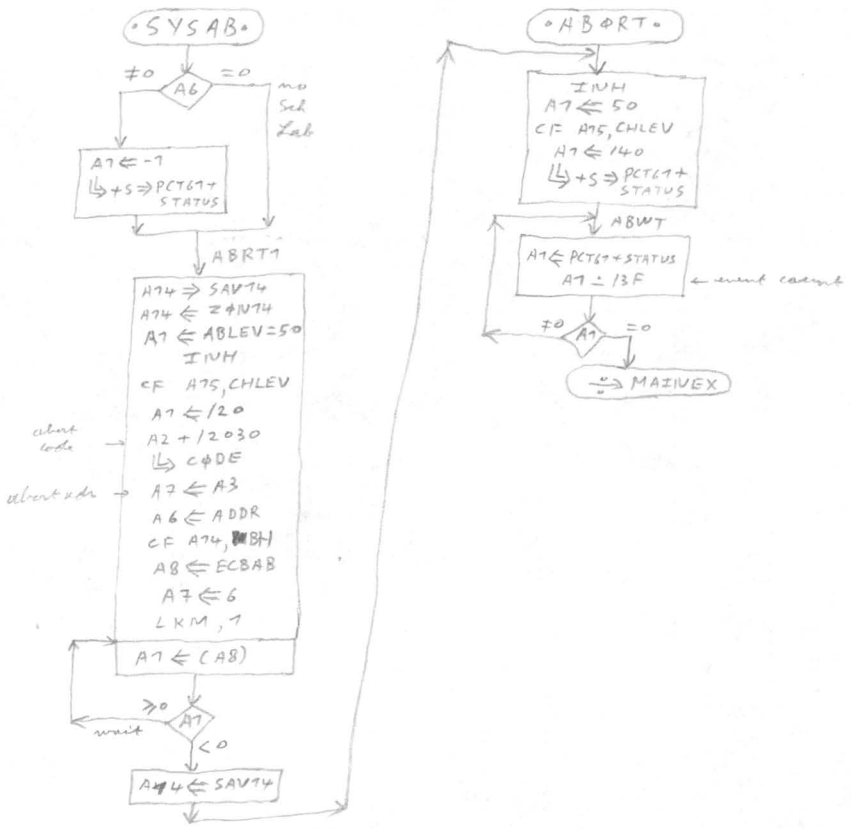

·PAUSE·
CAS)+1
A1⇐(INHST)
=0
A1
≠0
÷>ERHB
PAUSE1
A1⇐14000
↳±S⇒ PCTBT + STATUS
÷>CPRTN


·PSMAC·
A1⇐48
CF A15,CHLEV
A2⇐A7 ← chan count
A1⇐AB ← buff addr
IMS 3⇒ ECBMES+2
A7⇐6
AB⇐ECBMES
LKM ,1
A1⇐(AB)
≥0
A7
<0
A1⇐14000
↳±S⇒PCTBT + STATUS ← set pause bit
÷>DISPAT

PAUSE

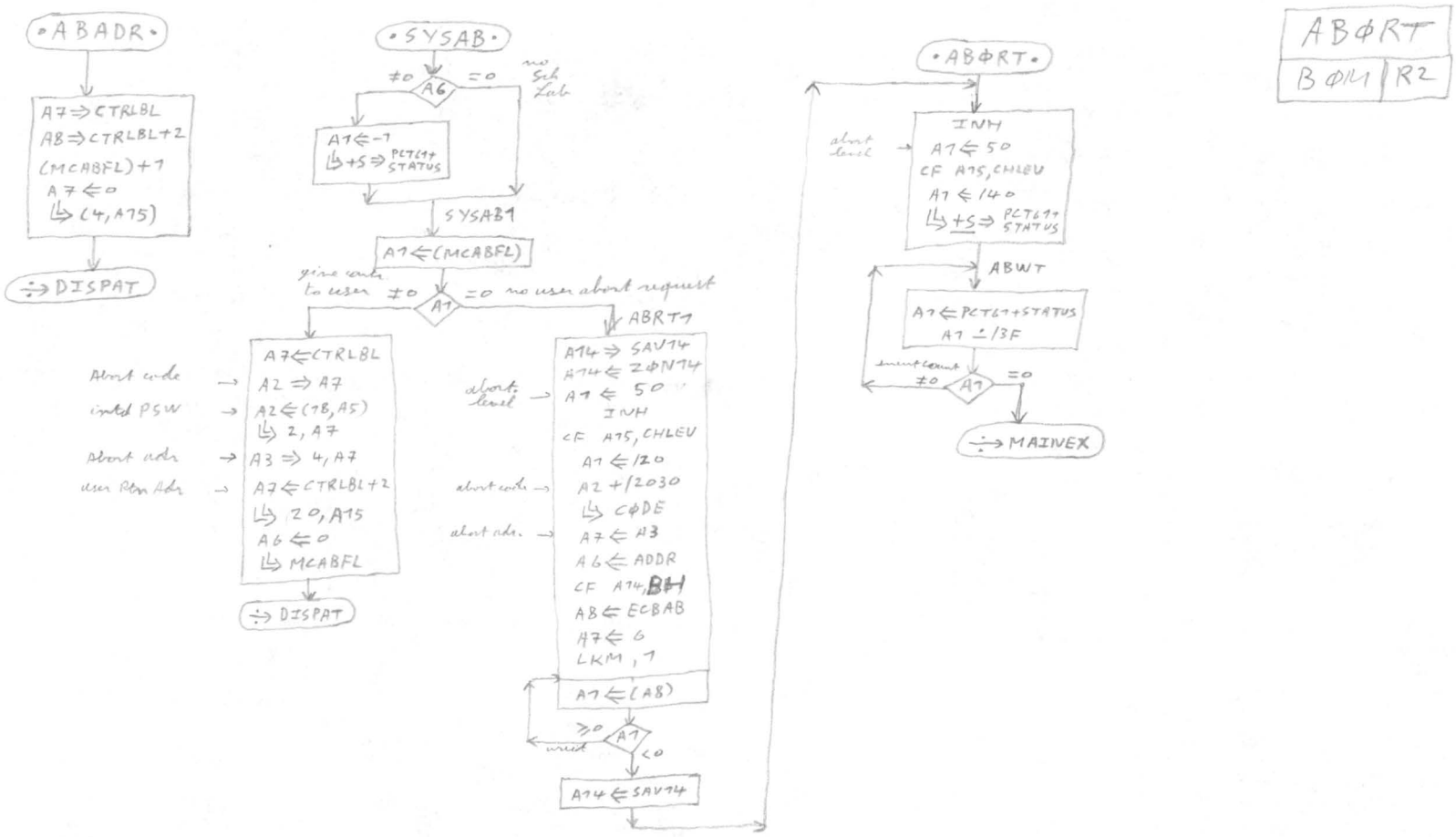
B⌀M | R2

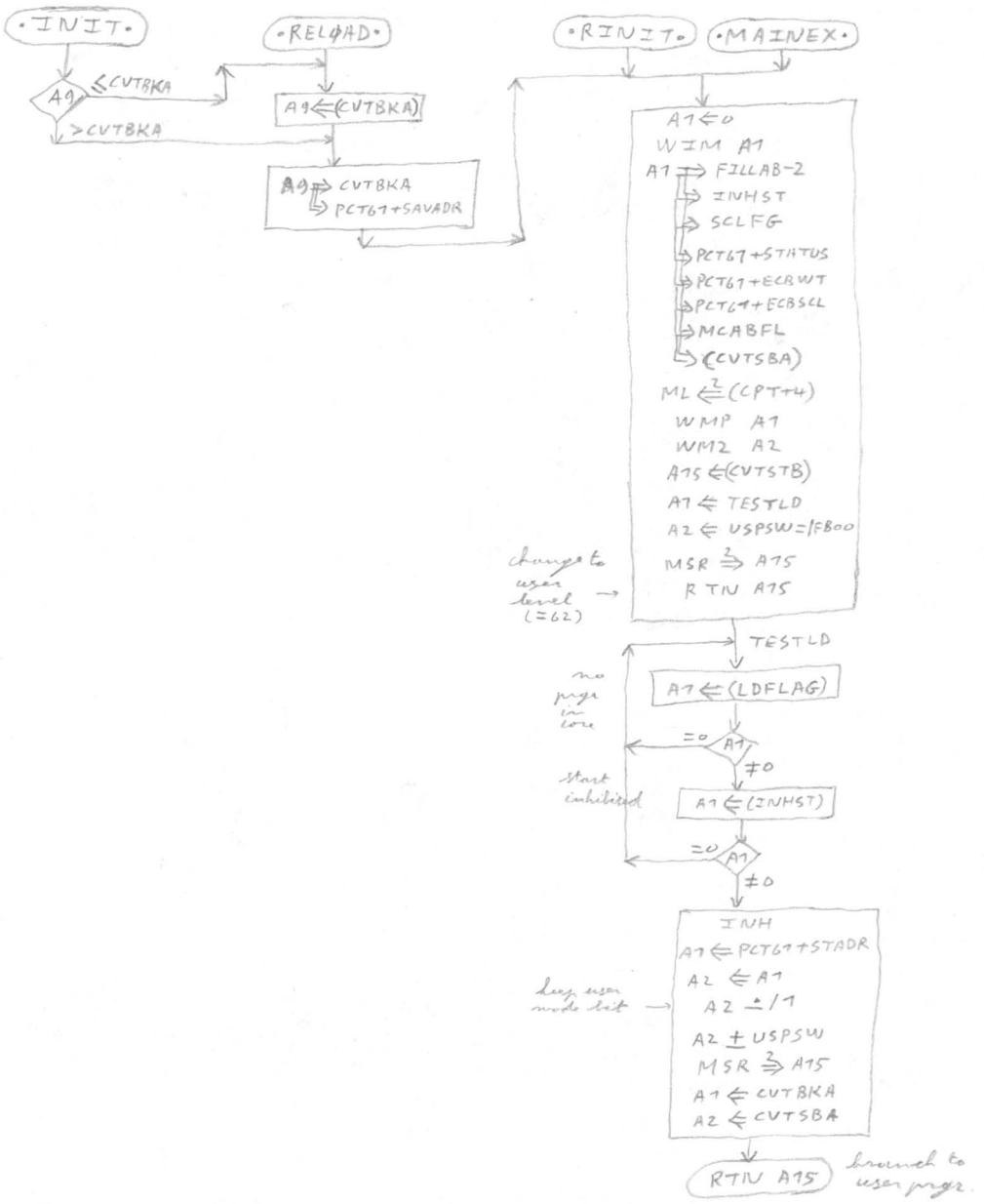
ABØRT
DIBØM

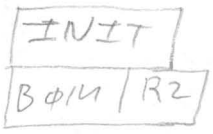

•SYSAB•
≠0
=0
AB
no Sel Lab
A7 ⇐ -1
↳+S⇒PCT67+ STATUS
ABRT1
A14⇒SAV14
A14⇐ZØIV14
A7⇐ABLEV=50
IVH
CF A15,CHLEV
A7⇐120
client code
A2+12030
↳CØDE
client xdr
A7⇐A3
A6⇐ADDR
CF A14,BH
A8⇐ECBAB
A7⇐6
LKM,7
A7⇐(A8)
≥0
A7
wait
<0
A14⇐SAV14

•ABØRT•
IVH
A7⇐50
CF A15,CHLEV
A7⇐140
↳+S⇒PCT67+ STATUS
ABWT
A7⇐PCT67+STATUS
A7÷13F
← event code
≠0
A7
=0
⇒MAIVEX

ABØRT	
BØM1	R2

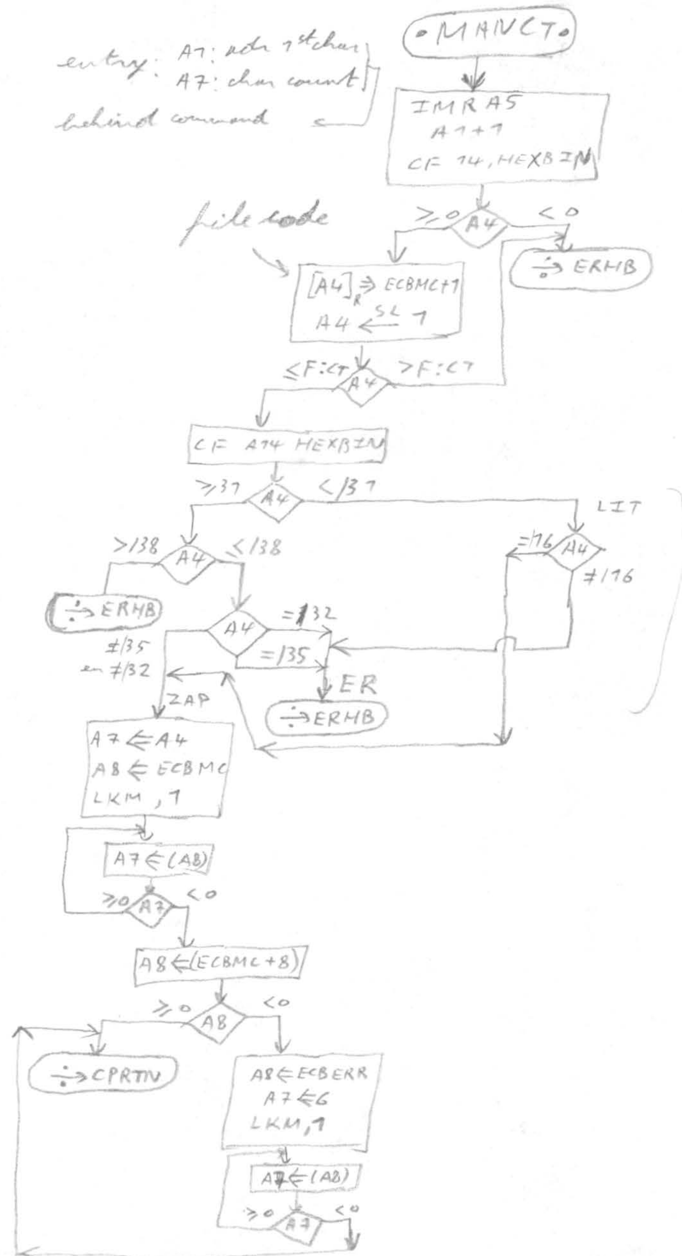



•INIT•
•RELOAD•
•RINIT•
•MAINEX•
A0
≤CVTBKA
>CVTBKA
A0⇐(CVTBKA)
A0⇒CVTBKA
PCT67+SAVADR
A1⇐0
WIM A1
A1⇒FILLAB-2
INHST
SCLFG
PCT67+STATUS
PCT67+ECBWT
PCT67+ECBSCL
MCABFL
(CVTSBA)
ML⇐2(CPT+4)
WMP A1
WM2 A2
A15⇐(CVTSTB)
A1⇐TESTLD
A2⇐USPSW=FB00
MSR⇒2 A15
RTN A15
change to user level (=62)
TESTLD
no pages in core
A1⇐(LDFLAG)
=0 A1
≠0
start inhibited
A1⇐(INHST)
=0 A1
≠0
INH
A1⇐PCT67+STADR
A2⇐A1
A2≐/1
leave user mode bit
A2±USPSW
MSR⇒2 A15
A1⇐CVTBKA
A2⇐CVTSBA
RTN A15
branch to user pages


INIT
Вфим
R2

MANCT

GPBØM | R2



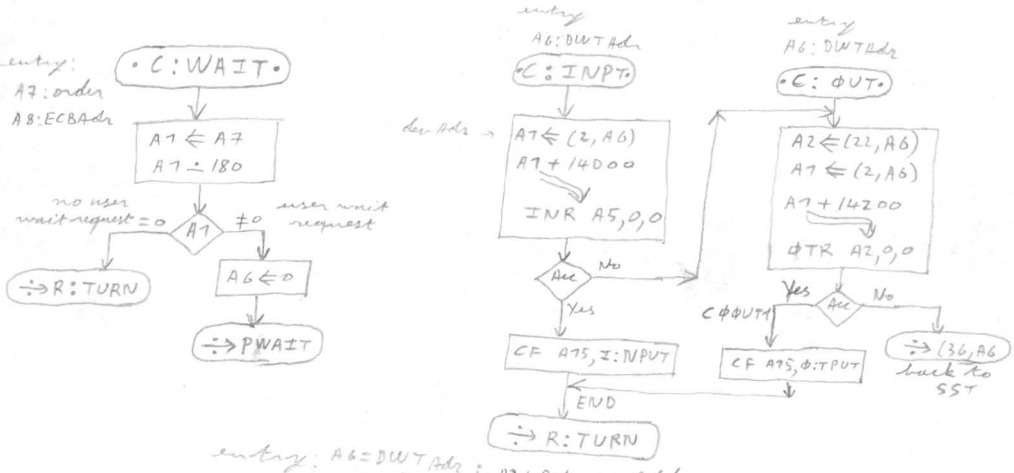
MC: filled order

allowed orders: /16, /31, /33, /34, /36, /37, /38

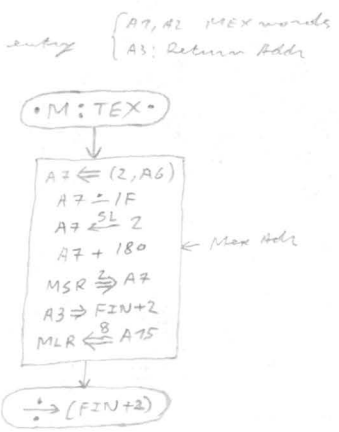
search filemark, AVEAS, Read

for MT

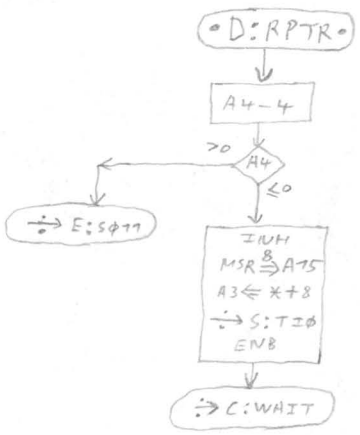
COMIO
BOM	R2

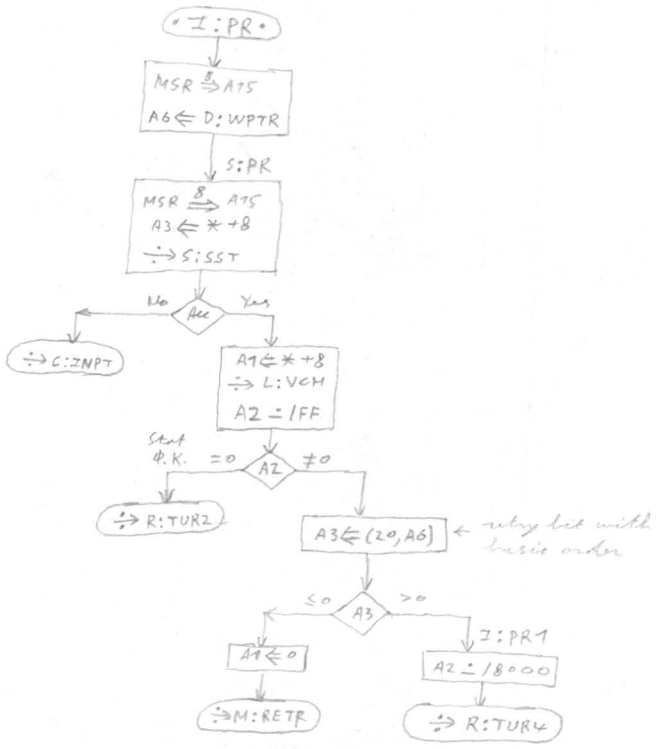

entry:
A7: order
A8: ECBAdr
•C:WAIT•
A7 ⇐ A7
A7 ÷ 180
no user wait request = 0
A7
≠0
user wait request
÷→R:TURN
A6⇐0
÷→PWAIT
entry
A6: DWTAdr
•C:INPT•
dev Adr →
A7⇐(2,A6)
A7+14000
INR A5,0,0
Acc
No
Yes
CF A7S,I:NPUT
END
÷→R:TURN
entry
A6: DWTAdr
•C:OUT•
A2⇐(22,A6)
A7⇐(2,A6)
A7+14200
OTR A2,0,0
Yes
Acc
No
COUT1
CF A7S,O:UTPUT
÷→(36,A6
back to SST


entry: A6=DWTAdr ; A3: Return Addr
•S:TIO•
•H:LTIO•
•O:TRIO•
•I:NRIO•
•S:SST•
A4⇐14200
A4⇐14280
A4⇐14700
A4⇐14900
A4⇐14AC0
A5⇐(2,A6)
A5÷13F
CIO A2,1,0
CIO A2,0,0
OTR A1,0,0
INR M,0p
SST A2,0
C:NSIO
INH
A5⇐(2,A6)
A5÷13F
SONO
A5±A4
A5⇒I:NS
A3⇒RETURN+2
MLR⇐B A15
'I:NS'
Il contenuto →
÷→(RETURN+2)


entry
A7, A2 MEX words
A3: Return Addr
•M:TEX•
A7⇐(2,A6)
A7÷1F
A7 SL 2
A7+180
MSR⇒A7
A3⇒FIN+2
MLR⇐B A15
← Mex Addr
÷→(FIN+2)

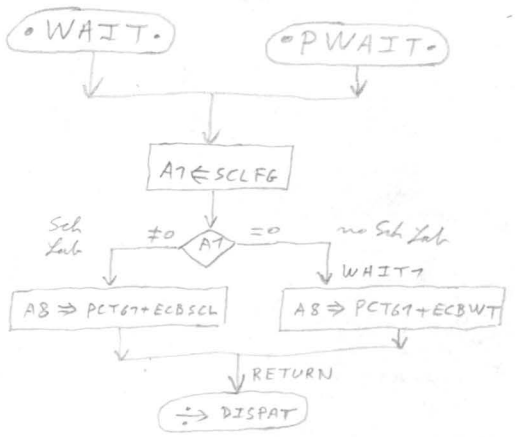
entry { A6: DWT Ad
A4: Aden

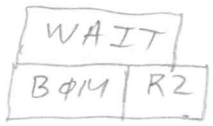




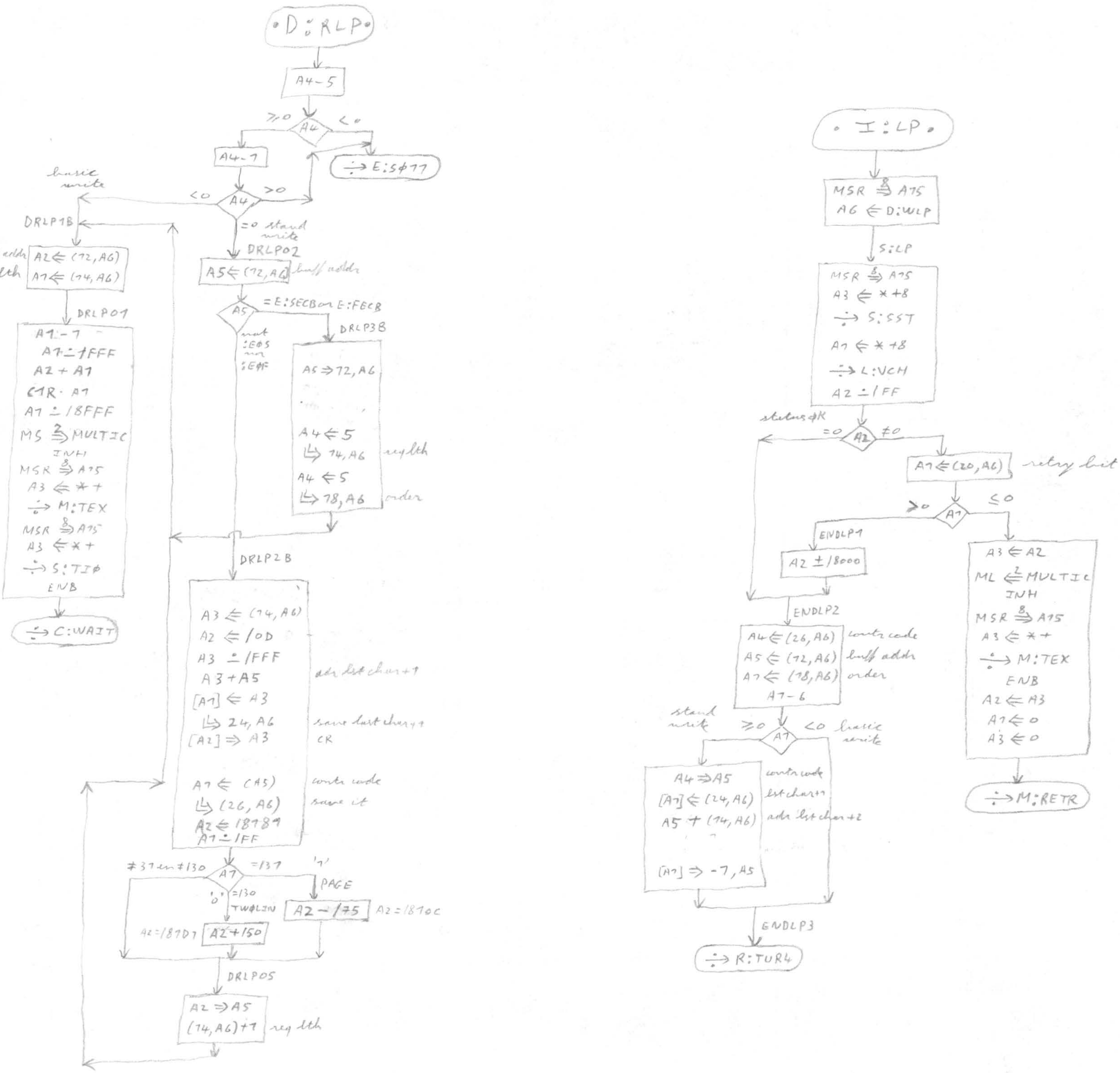
DRPR38
BØTT/R2

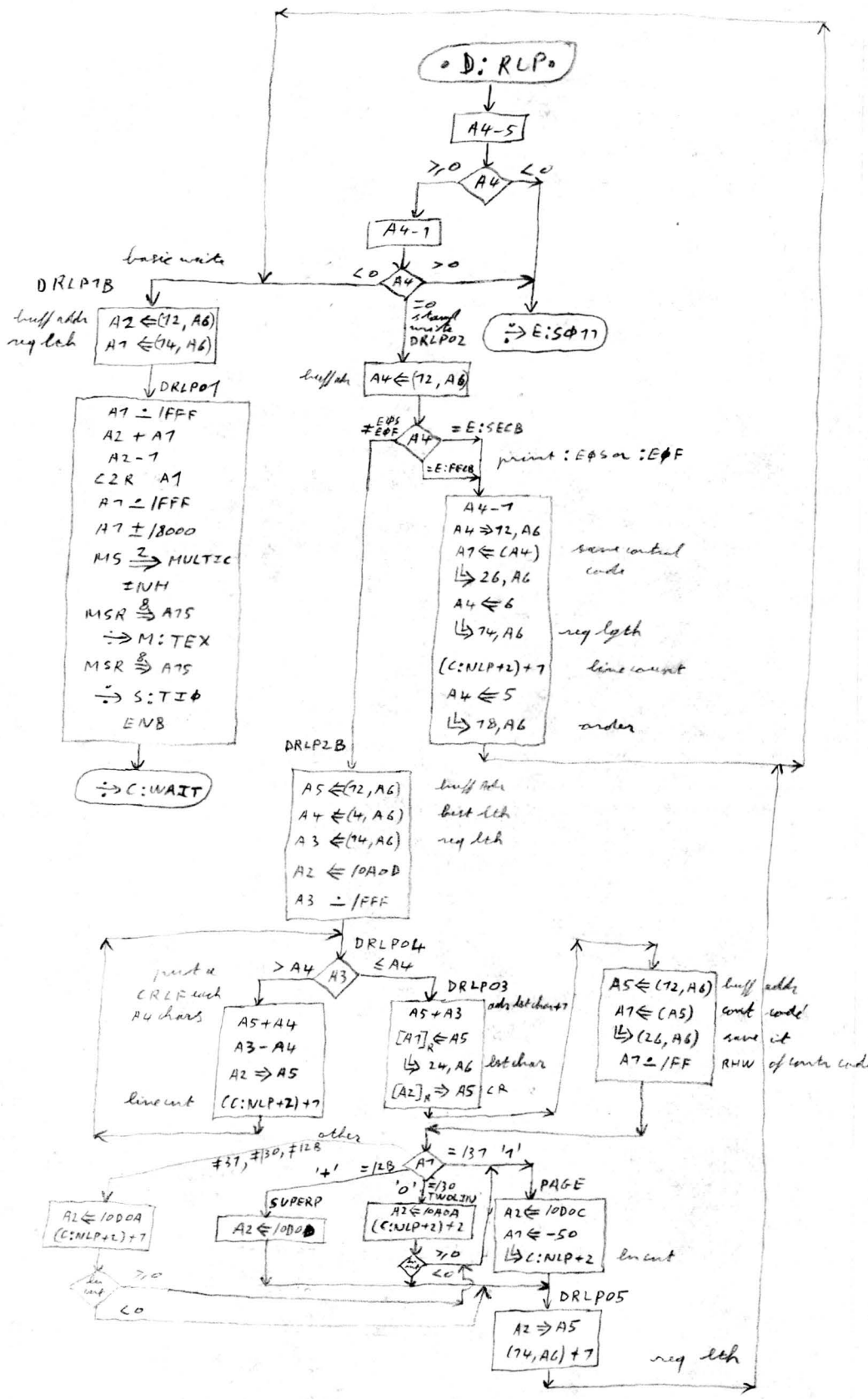
entry: A8 contains
Adr of word, wait
till that word is <0

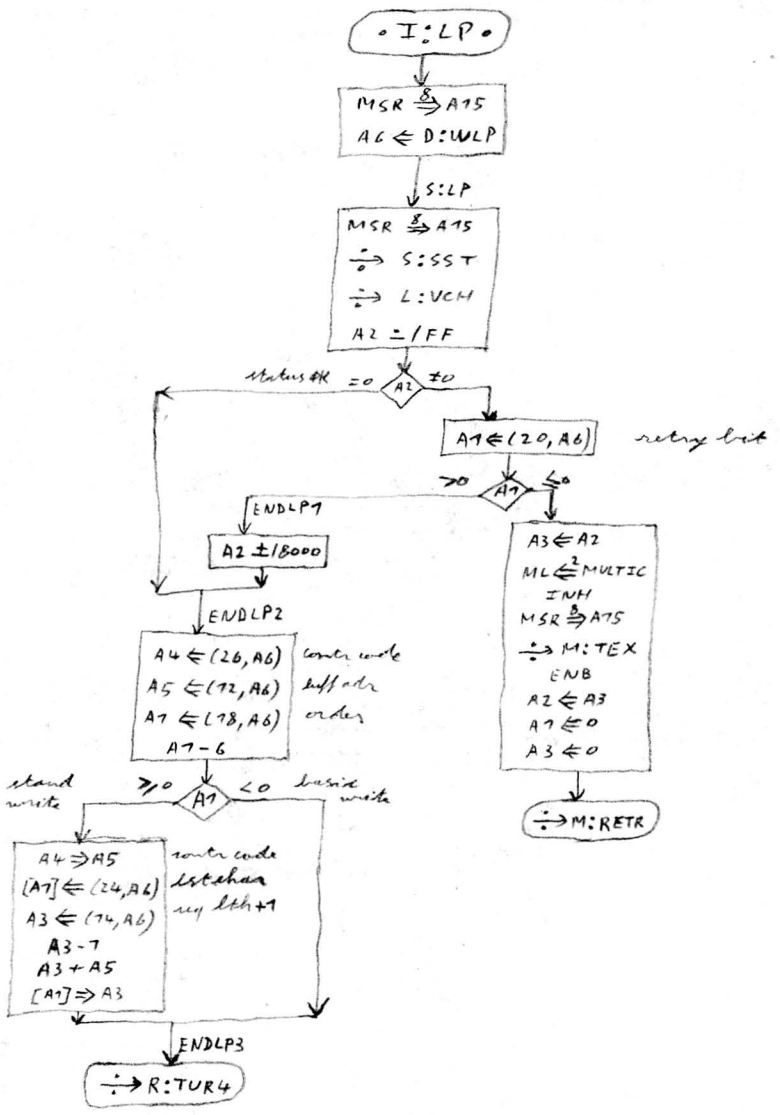




DRLP
GPBØM


D:RLP
A4-5
A4
E:SØ11
A4-1
basic write
stand write
DRLP1B
A2 ⇐ (12,A6)
A1 ⇐ (14,A6)
buff addr
req lth
DRLP01
DRLP02
A5 ⇐ (12,A6)
A5
=E:SECB or E:FECB
DRLP3B
A5 ⇒ 12,A6
A4 ⇐ 5
14,A6
A4 ⇐ 5
18,A6
order
DRLP2B
A3 ⇐ (14,A6)
A2 ⇐ /0D
A3 ∸ /FFF
A3 + A5
[A1] ⇐ A3
24,A6
save last char+1
CR
A1 ⇐ (A5)
contr code
save it
A2 ⇐ 18184
A1 ∸ /FF
A1
PAGE
A2 - /75
A2 = 1810C
TWØLIN
A2 + 150
A2 = 181D1
DRLP05
A2 ⇒ A5
(14,A6)+1
C:WAIT
I:LP
MSR ⇒ A15
A6 ⇐ D:WLP
S:LP
A3 ⇐ * + 8
S:SST
A1 ⇐ * + 8
L:VCH
A2 ∸ /FF
status OK
A2
A1 ⇐ (20,A6)
retry bit
A1
ENDLP1
A2 ± /8000
ENDLP2
A4 ⇐ (26,A6)
A5 ⇐ (12,A6)
A1 ⇐ (18,A6)
A1 - 6
contr code
buff addr
order
A1
basic write
A4 ⇒ A5
[A1] ⇐ (24,A6)
A5 + (14,A6)
[A1] ⇒ -1,A5
ENDLP3
R:TUR4
A3 ⇐ A2
ML ⇐ MULTIC
INH
MSR ⇒ A15
A3 ⇐ * +
M:TEX
ENB
A2 ⇐ A3
A1 ⇐ 0
A3 ⇐ 0
M:RETR


D:RLP
A4-5
A4
>0
<0
A4-1
basic write
DRLP1B
A2 ⇐ (72, A6)
A7 ⇐ (74, A6)
buff addr
req lgth
DRLP01
A7 ± /FFF
A2 + A7
A2-7
C2R A7
A7 ± /FFF
A7 ± /8000
MS 2⇒ MULTIC
INH
MSR 8⇒ A75
÷⇒ M:TEX
MSR 8⇒ A75
÷⇒ S:TIØ
ENB
÷⇒ C:WAIT
=0
stand write
DRLP02
÷⇒ E:SØ77
buff adr
A4 ⇐ (72, A6)
EØS ≠EØF
=E:SECB
=E:FFCB
print: EØS or :EØF
A4-7
A4 ⇒ 72, A6
A7 ⇐ (A4)
↳ 26, A6
A4 ⇐ 6
↳ 74, A6
(C:NLP+2)+7
A4 ⇐ 5
↳ 78, A6
save control code
req lgth
line count
order
DRLP2B
A5 ⇐ (72, A6)
A4 ⇐ (4, A6)
A3 ⇐ (74, A6)
A2 ⇐ 10A0B
A3 ± /FFF
buff adr
last lth
req lth
DRLP04
>A4
A3
≤A4
DRLP03
print a CR LF each A4 chars
A5+A4
A3-A4
A2 ⇒ A5
(C:NLP+2)+7
line cnt
A5+A3
[A7]R ⇐ A5
↳ 24, A6
[A2]R ⇒ A5
addr last char+7
last char
CR
A5 ⇐ (72, A6)
A7 ⇐ (A5)
↳ (26, A6)
A7 ± /FF
buff addr
cont code
save it
RHW of cont code
other
≠37, ≠30, ≠128
A7
=137 '7'
'+' =128
'0' ≠130 TWOLINE
PAGE
SUPERP
A1 ⇐ 10D0A
(C:NLP+2)+7
A2 ⇐ 10D0D
A2 ⇐ 10A0A
(C:NLP+2)+2
A2 ⇐ 10D0C
A7 ⇐ -50
↳ C:NLP+2
line cnt
>0
<0
DRLP05
A2 ⇒ A5
(74, A6)+7
req lth


I:LP
MSR 8⇒ A75
A6 ⇐ D:WLP
S:LP
MSR 8⇒ A75
÷⇒ S:SST
÷⇒ L:VCH
A2 ± /FF
status ØR =0
≠0
A7 ⇐ (20, A6)
retry lmt
>0
<0
ENDLP1
A2 ± /8000
A3 ⇐ A2
ML 2⇐ MULTIC
INH
MSR 8⇒ A75
÷⇒ M:TEX
ENB
A2 ⇐ A3
A7 ⇐ 0
A3 ⇐ 0
÷⇒ M:RETR
ENDLP2
A4 ⇐ (26, A6)
A5 ⇐ (72, A6)
A7 ⇐ (78, A6)
A7-6
cont code
buff adr
order
stand write
>0
A7
<0
basic write
A4 ⇒ A5
[A7] ⇐ (24, A6)
A3 ⇐ (74, A6)
A3-7
A3+A5
[A7] ⇒ A3
cont code
last char
req lth+1
ENDLP3
÷⇒ R:TURN

HDB CA3

modif
A7 4 ⇒ A6

•HEBICA•
IMR A5
MS u8 ⇒ SAV
START
A14 ⇐ STACK
A8 ⇐ ECB
CF A14 SLIN
[A7]R ⇐ BUF+4
A7 = 1FF
A7 − 120
'blank' =0 A7 ≠0
A7−1B
'+' =0 A7 ≠0
A7−2
'−' =0 A7 ≠0
SU
A7 ⇐ 18
A7−178
'E' =0 A7 ≠0
FOUT
EINDE
ML ⇐ SAV
÷ CPRTN
AD
A7+195
[A7]R ⇒ AS1
CF A14, HEBI
CF A14, SLIN
DEBI
A4 ⇐ −5
A2 ⇐ 0
DEBI2
[A3]R ⇐ BUF+5, A4
A3 − 130
A3 ≥0 <0 FOUT
A3 ≤9 >9
A2 + A3
OVFL N Y
A4 + 7
A4 <0 ≥0 AS1
A2 * 10
A7 =0
AS1
A5 +4/−A2
BIHE
A3 ⇐ 4
BIHE2
A7 ⇐ A5
A5 SS 4
A7 = 1F
A7 − 1A
A7 <0
A7 + 7
BIHE7
A7 + 13A
[A7]R ⇒ BUF+2, A3
A3 − 7
A3 >0 ≤0
A4 ⇐ 120
A5 ⇐ 7
RSLT1
[A4]R ⇒ BUF+2
RSLT2
A7 ⇐ 6
CF SL ΦUT

HID
CF A14, HEBI
BIDE
A4 ⇐ 120
A2 <0 ≥0
'−' →
0 − A2
A4 + 1D
BIDE7
A3 ⇐ 5
BIDE2
A7 ⇐ 0
(A7, A2) ÷ 10
A7 + 130
[A7]R ⇒ BUF+2, A3
A3 − 7
A3 >0 ≤0
A5 ⇐ 8

HEBI
A3 ⇐ 4
HEBI2
[A7]R ⇐ BUF −7, A3
A7 − 130
F A7 <0 ≥0
A7 ≥10 <10
A7 − 111
F A7 <0 ≥0
A7 − 5
F A7 >0 ≤0
A7 + 1F
HEBI7
DRL, 4
A3 − 7
A3 >0 ≤0
RTN A14

SLIN
A5 ⇐ 5
A7 ⇐ 1
SLΦUT
A5 ⇒ 4, A8
LKM, 7
A7 ⇐ (A8)
A7 ≥0 >0
RTN A14