

- DWT + 0 → 2 ascii char of device name
 + 2 → device addr.
 + 4 → best length
 + 6 → addr of destination driver
 + 8 → software status
 + 10 → ECB Addr.
 + 12 → { char Addr
 buff Addr at beginning
 + 14 → requested length
 + 16 → effective length
 + 18 → order (4 LSB)
 + 20 → retry bit with basic order (R bit of order in A7 with an LKM instr.)
 + 22 → { output → word to output
 input → tabulation table addr.
 + 24 → { checksum with object order / Disk: order
 line printer → same last char of buff.
 + 26 → { obj 4*4 → right or left
 line printer → same control code
 + 28 → A5
 + 30 → A6 (in I:PRM scheduled label addr.)
 + 32 → controller status addr (C:NASA, C:PTR etc)
 + 34 → attach
 + 36 → SST sequence addr.

DWT

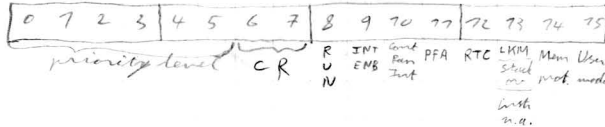
PCT 67

STADR DATA 0
 SAVADR DATA 0 = CVTBKA
 STATUS } DATA 0 = 0
 PCT 67 }
 ECBWT DATA 0 = 0
 ECB SCL DATA 0 = 0
 ↑
 at initialisation

ECB

- +0 → Event char | File code
 +2 → Buff Address
 +4 → Requested length
 +6 → Effective length
 +8 → Status word
 +10 → Tabulation table Addr.

PSW



CVT

CVT } 18000 memory size of 16K
 CVTMSZ }
 CVTSTB 57B addr of stack-base for A15
 CVTSBA 10 smallest buff area addr
 CVTBBA 10 biggest buff area addr
 CVTBKA 17800 background address

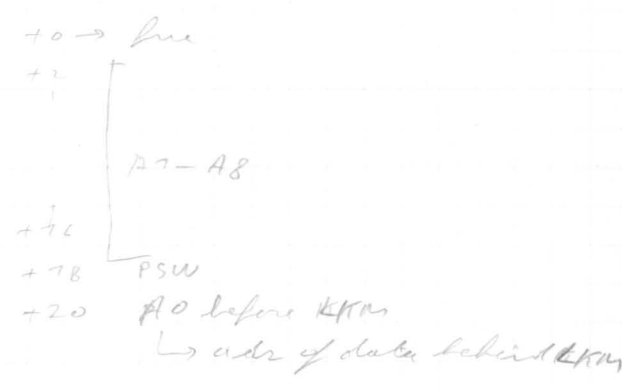
PSW



Stack base: 13C 25C
 40 112
 77C 380 → 790

CO
 40
 100
 80 → 12B

70 77 12 13 14
 1 1 1 1 1
 ↑



RJT 17D

LDR A3, -
 LDR A8, -
 LRM
 DATA

A7 = 0
 A3 = 2
 A6 = 0
 A7 = user data
 A0 = user prog + 2

input output: LRM, Data = 1 ⇒ M: IARM

A2 = adr M: IARM

LRM	instr	llz
1	M: IARM	77
2	WAIT	32
3	EXIT	29
4	GETBUF	38
5	FRBUFF	43
6	PSMAC	152
7	ABADR	148

M: IARM
 entry cond { A6 = Sch Lab Addr
 A7 = order
 A8 = ECB Addr
 A7 = Addr of ADDR7
 ↓
 L: VCH
 ADDR7 → (A75), A75-2
 1000 → (A75), A75-2
 RFN 75

A2 = 0
 A6 = DWT ADDR

→ ADDR7
 file code x2
 A7 = order
 A3 = DWT addr of dev.
 A4 = ECB Addr