81/26

EISCAT
TECHNICAL
NOTE

Instruction Manual for EISCAT Digital Correlator (Revised)

by
Terrance Ho

KIRUNA
Sweden

INSTRUCTION MANUAL

FOR

EISCAT DIGITAL CORRELATOR (REVISED)

by

TERRANCE HO
MAX PLANCK INSTITUT
POSTFACH 20
D-3411 KATLENBURG-LINDAU 3
W. GERMANY

EISCAT Technical Note 81/26
Printed in Sweden
EISCAT Scientific Association
Kiruna, March 1981
ISSN 0349-2710

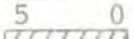
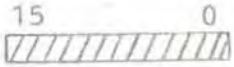
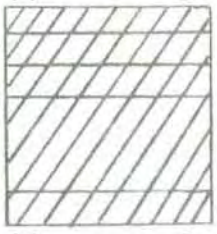
TABLE OF CONTENTS

I. PROGRAMMABLE REGISTERS

DATA-FIELD IN DIGITAL CORRELATOR

All numeric values are decimal.

ABBR.: APB: Address processor for buffer-memory, APM: Address processor for result-memory, C: Computer, F: Front-panel, P: Internal program.

REGISTER NAME		ADDRESS	NUMERIC-RANGE	LOAD-ACCESS
STATUSWORD (STAT)		1,	0-65535	C,F
START-ADDRESS-REGISTER (SAR)		4,	0-63	C,F,P
BASE-ADDRESS-REGISTER, APB (BAR)		5,	0-65535	C,F
DATA I-REGISTER, APB		6,	0-65535	C,F,P
REGISTER-STACK, APB* (APBRS)		16,0 16,1 16,2 . . . 16,15	0-65535 " " "	C,F " " "
REGISTER-STACK, APM (APMRS)		17,0, 17,1, 17,2, . . . 17,15,	0-4095 " " "	C,F " " "
LOAD-REGISTER FOR LOOP-COUNTER 1 (LCR1)		18,	0-4095	C,F,P
TEMPORARY STORAGE FOR VALUE OF LOOP-COUNTER 1 (LCR1A)		-	0-4095	P
LOAD-REGISTER FOR LOOP-COUNTER 2 (LCR2)		19,	0-4095	C,F,P
LOAD-REGISTER FOR LOOP-COUNTER 3 (LCR3)		20,	0-4095	C.F.P
"CORRELATOR-READY"-REGISTER (CRA)		63,	0-1	C,P

* APBRS must be loaded through the Data I-reg., APB.

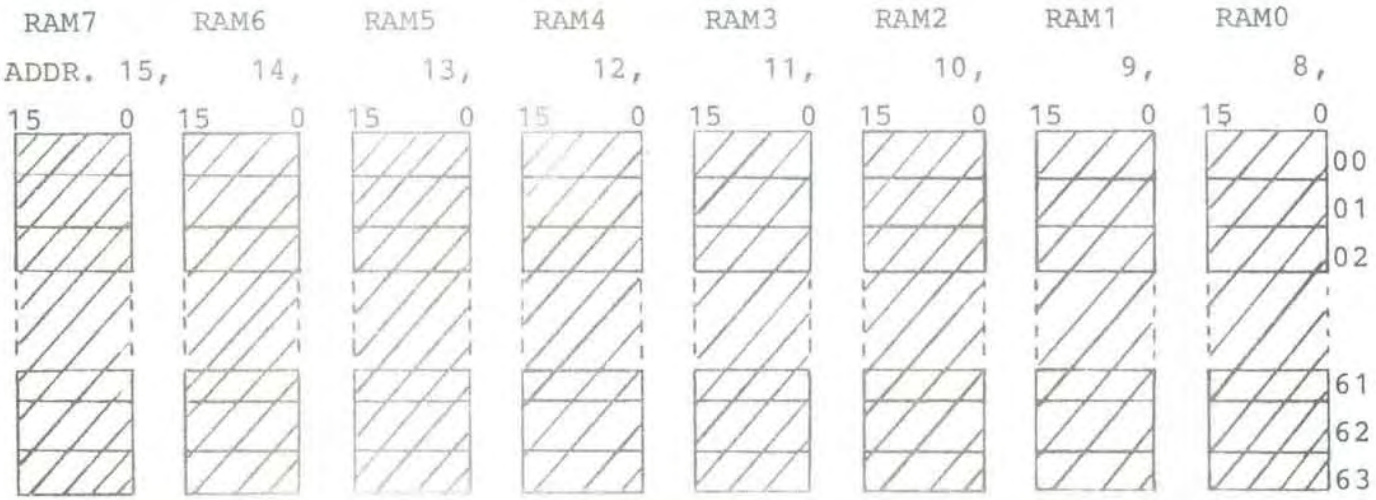
II. MEMORY-LOCATIONS

PROGRAM-FIELD IN DIGITAL CORRELATOR

All numeric values are decimal.

ABBR.: RAM: Random access memory.

PROGRAM-MEMORY STRUCTURE:



When accessed from external source (computer or front-panel) for program set-up, the memory is split into 8 separate pages (RAMO, RAM1,..) with separate address identification. Each location in page is given by the SUB-ADDRESS.

The correlator internal instruction word (128 bits) is defined by parallel read-out from the same SUB.ADDR. in all pages, i.e. Intruction word location is defined by the SUB.ADDR.

PROGRAM RESTRICTIONS: Location 00 is used for idle-running of the correlator.

Location 63 is used for storage of service-routine.

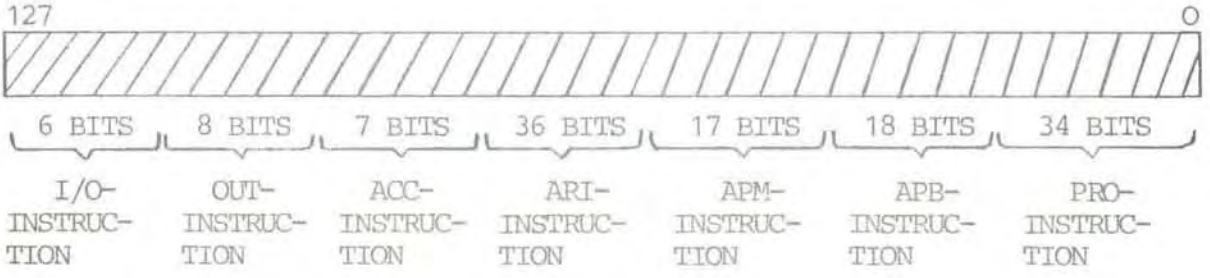
Pointer when handling external interrupts.

LOCATIONS FREE FOR PROGRAMMING: 01-62.

III. THE PROGRAM FIELD OF THE CORRELATOR

The Program Field of the correlator consists of 7 different functions which operate in parallel. The control is given by the 128-bits instruction word.

THE INSTRUCTION WORD SEPARATED INTO MAIN INTERNAL FUNCTIONS



PRO-INSTRUCTION: Controls the execution of the correlator program and reload of the programmable registers in the data field (SAR, BAR, LCR1, LCR2, LCR3).

APB-INSTRUCTION: Controls the addressing of the buffer memory (data input source) and generation of reload values to the registers in the data field (APB register stack).

APM-INSTRUCTION: Controls the addressing of the result memory (data storage) and generation of reload values to the registers in the data field (APM register stack).

ARI-INSTRUCTION: Controls arithmetical functions on data from the buffer memory.

ACC-INSTRUCTION: Controls arithmetical accumulation of processed data and data from the result memory.

OUT-INSTRUCTION: Controls direct memory access (DMA) from the result memory to the computer.

I/O-INSTRUCTION: Controls data/address communication in a multi-correlator system.

The actual bit-assignments for the functions are given on the next page. Due to hardware construction the different functions are scrambled.

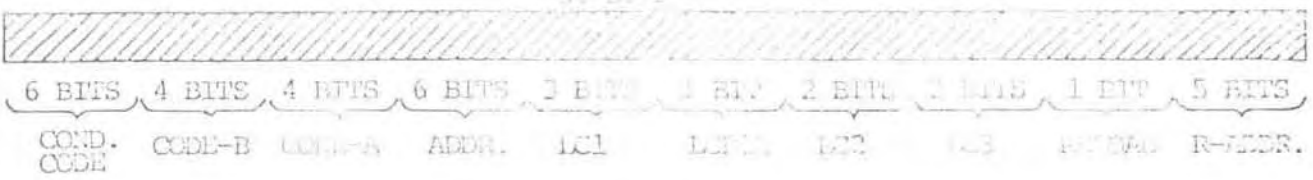
IV. BIT ASSIGNMENTS OF THE RAM MODULES

INSTRUCTION-WORD BIT ASSIGNMENTS:

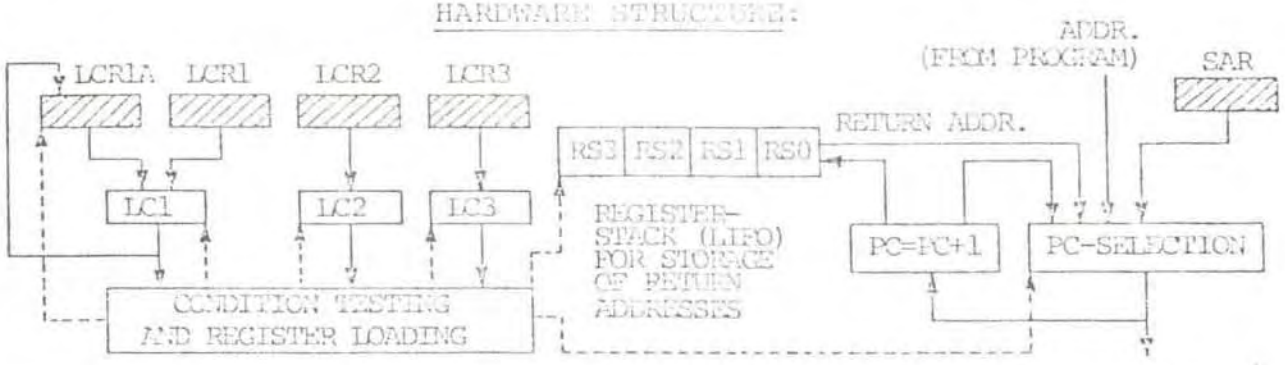
V. [illegible]

Abbreviations used: LC - Loop counter; LCR - Local register for loop counter; LIFO - Last in first out; PC - Program counter.

INSTRUCTION-WORD SEPARATION:



HARDWARE STRUCTURE:



▨ : REGISTERS IN DATA-FIELD

PC = SUB.ADDR. FOR NEXT INSTRUCTION READ-OUT FROM PROGRAM-MEMORY

All loop-counters have 12 bits with data-range: 0-4095

CONDITION CODE INSTRUCTIONS:

INTEGER CODING	TEST CONDITION
32,40,48,56	USE CODE A UNCONDITIONALLY
57	IF LC1=0 USE CODE B ELSE USE CODE A
58	IF LC2=0 USE CODE B ELSE USE CODE A
59	IF LC1 OR LC2=0 USE CODE B ELSE USE CODE A
60	IF LC3=0 USE CODE B ELSE USE CODE A
61	IF LC1 OR LC3=0 USE CODE B ELSE USE CODE A
62	IF LC2 OR LC3=0 USE CODE B ELSE USE CODE A
63	IF LC1 OR LC2 OR LC3=0 USE CODE B ELSE USE CODE A
50	IF LC2≠0 USE CODE B ELSE USE CODE A
51	IF LC1=0 OR LC2≠0 USE CODE B ELSE USE CODE A
54	IF LC2≠0 OR LC3=0 USE CODE B ELSE USE CODE A

INTEGER CODING	TEST CONDITION
55	IF LC1=0 OR LC2≠0 OR LC3=0 USE CODE B ELSE USE CODE A
44	IF LC3≠0 USE CODE B ELSE USE CODE A
45	IF LC1=0 OR LC3≠0 USE CODE B ELSE USE CODE A
46	IF LC2=0 OR LC3≠0 USE CODE B ELSE USE CODE A
47	IF LC1=0 OR LC2=0 OR LC3≠0 USE CODE B ELSE USE CODE A
38	IF LC2 OR LC3≠0 USE CODE B ELSE USE CODE A
39	IF LC1=0 OR LC2≠0 OR LC3≠0 USE CODE B ELSE USE CODE A
27	IF LC1=0 USE CODE B ELSE IF LC2=0 USE CODE A ELSE CONTINUE
30	IF LC3=0 USE CODE B ELSE IF LC2=0 USE CODE A ELSE CONTINUE
29	IF LC1 OR LC3=0 USE CODE B ELSE CONTINUE
31	IF LC1 OR LC3=0 USE CODE B ELSE IF LC2=0 USE CODE A ELSE CONTINUE
19	IF LC1=0 USE CODE B ELSE IF LC2≠0 USE CODE A ELSE CONTINUE
22	IF LC3=0 USE CODE B ELSE IF LC2≠0 USE CODE A ELSE CONTINUE
23	IF LC1 OR LC3=0 USE CODE B ELSE IF LC2≠0 USE CODE A ELSE CONTINUE
11	IF LC1≠0 USE CODE B ELSE IF LC2=0 USE CODE A ELSE CONTINUE
14	IF LC3≠0 USE CODE B ELSE IF LC2=0 USE CODE A ELSE CONTINUE
13	IF LC1 OR LC3≠0 USE CODE B ELSE CONTINUE
15	IF LC1 OR LC3≠0 USE CODE B ELSE IF LC2=0 USE CODE A ELSE CONTINUE
3	IF LC1≠0 USE CODE B ELSE IF LC2≠0 USE CODE A ELSE CONTINUE
5	IF LC3≠0 USE CODE B ELSE IF LC2≠0 USE CODE A ELSE CONTINUE
7	IF LC1 OR LC3≠0 USE CODE B ELSE IF LC2≠0 USE CODE A ELSE CONTINUE

CODE A/CODE B INSTRUCTIONS:

INTEGER CODING	FUNCTION	COMMENTS
0	PC=PC+1, POP STACK	CONTINUE AND DELETE LAST RETURN ADDRESS
1	PC=RETURN ADDRESS, POP STACK	RETURN AND DELETE LAST RETURN ADDRESS
2	PC=JUMP ADDRESS, POP STACK	JUMP AND DELETE LAST RETURN ADDRESS
3	PC=SAR, POP STACK	GOTO SAR AND DELETE LAST RETURN ADDRESS
4,12	PC=PC+1	CONTINUE
5,13	PC=RETURN ADDRESS	RETURN
6,14	PC=JUMP ADDRESS	JUMP
7,15	PC=SAR	GOTO SAR
8	PC=PC+1, PUSH STACK	CONTINUE AND SET RETURN ADDRESS
9	PC=RETURN ADDRESS, PUSH STACK	RETURN AND SET RETURN ADDRESS
10	PC=JUMP ADDRESS, PUSH STACK	JUMP AND SET RETURN ADDRESS
11	PC=SAR, PUSH STACK	GOTO SAR AND SET RETURN ADDRESS

LC1 INSTRUCTIONS:

INTEGER CODING	FUNCTION	COMMENTS
0	NOOP	NO OPERATION
1	LC1=LC1-1	DECREMENT LOOP COUNTER
2	LC1=LCR1	LOAD LOOP COUNTER
3	LC1=LCR1A	LOAD LOOP COUNTER
4	IF LC1=0 THEN LC1=LCR1, LC2=LC2-1 ELSE LC1=LC1-1	CONDITIONAL LOAD/ DECREMENT
5	IF LC1=0 OR LC3=0 THEN LC1=LCR1A ELSE LC1=LC1-1	CONDITIONAL LOAD/ DECREMENT
6	IF LC1=0 THEN LC1=LCR1 ELSE LC1=LC1-1	CONDITIONAL LOAD/ DECREMENT
7	IF LC1=0 THEN LC1=LCR1A ELSE LC1=LC1-1	CONDITIONAL LOAD/ DECREMENT

LC2 INSTRUCTIONS:

INTEGER CODING	FUNCTION	COMMENTS
0	NOOP	NO OPERATION
1	LC2=LC2-1	DECREMENT LOOP COUNTER
3	LC2=LCR2	LOAD LOOP COUNTER

LC3 INSTRUCTIONS:

INTEGER CODING	FUNCTION	COMMENTS
0	NOOP	NO OPERATION
1	LC3=LC3-1	DECREMENT LOOP COUNTER
2	LC3=LCR3	LOAD LOOP COUNTER
3	IF LC3=0 THEN LC3=LCR3 ELSE LC3=LC3-1	CONDITIONAL LOAD/ DECREMENT

LCR1A INSTRUCTIONS:

INTEGER CODING	FUNCTION	COMMENTS
0	NOOP	NO OPERATION
1	LCR1A=LC1	REGISTER LOAD

RELOAD INSTRUCTIONS:

INTEGER CODING	FUNCTION	COMMENTS
0	NOOP	NO OPERATION
1	REGISTER RELOAD	RELOAD VALUE FROM APB

RELOAD ADDRESSES:

INTEGER CODING	FUNCTION	COMMENTS
4	RELOAD SAR	IF RELOAD=1
5	" BAR, APB	" "
18	" LCR1	" "
19	" LCR2	" "
20	" LCR3	" "

PROGRAMMING NOTES

Only the given instruction values should be used.

Only a maximum of 4 return addresses can be stored.

When the push stack option is used (set up return address) the stored value is the present PC value+1.

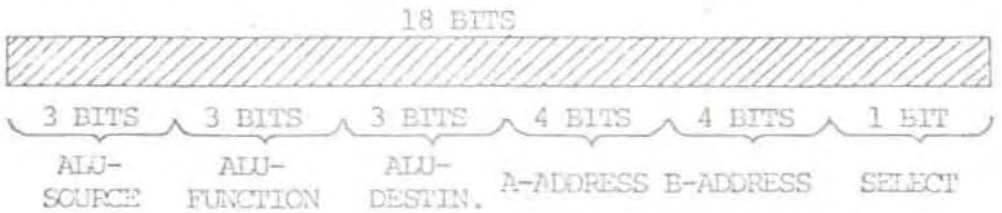
When the RELOAD option is used the next program instruction must not contain the RELOAD option or a load LOOP COUNTER instruction. The correlator needs 2 clock cycles for a RELOAD instruction.

Program Location 0 cannot be used for a register reload.

VI. APB/APM INSTRUCTION SET

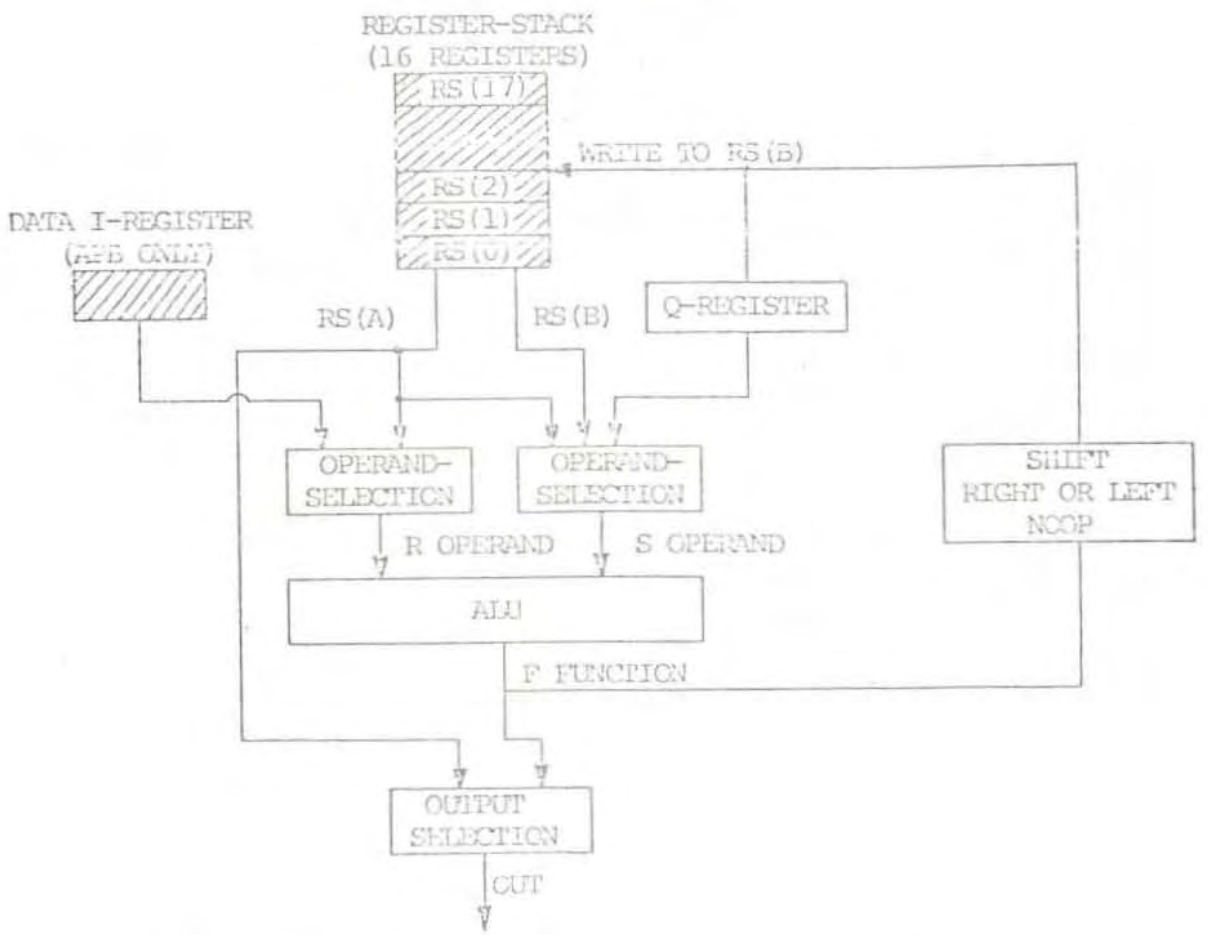
ABBREVIATIONS: APB - Address processor for Buffer memory;
APM - Address processor for result memory;
ALU - Arithmetical logical unit.

INSTRUCTION-WORD SEPARATION:



APM does not have the SELECT sub-instruction.

HARDWARE STRUCTURE:



For APB: Address-bus to buffer memory or reload value to registers.
For APM: Read/write address to result memory.
Numeric range for APB: 0 - 65535 (16 BITS)
Numeric range for APM: 0 - 4095 (12 BITS)

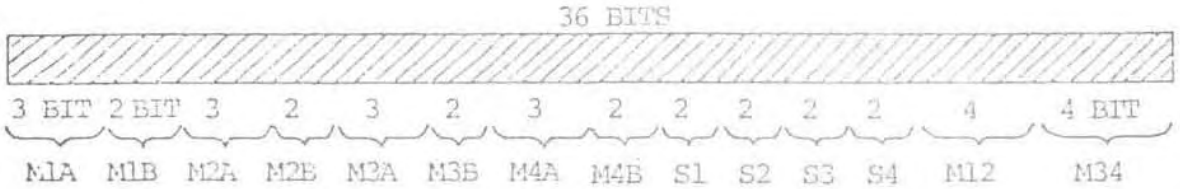
	INTEGER CODING	FUNCTION
ALU-SOURCE:	0	R=RS(A), S=Q
	1	R=RS(A), S=RS(B)
	2	R=0, S=Q
	3	R=0, S=RS(B)
	4	R=0, S=RS(A)
	5	R=DATAI, S=RS(A)
	6	R=DATAI, S=Q
	7	R=DATAI, S=0
ALU-FUNCTION:	0	F=R+S
	1	F=S-R
	2	F=R-S
	3	F=R OR S
	4	F=R AND S
	5	F=$\overline{R}$ AND S
	6	F=R EXOR S
	7	F=R EXNOR S
ALU-DESTINATION:	0	Q=F, OUT=F
	1	OUT=F
	2	RS(B)=F, OUT=RS(A)
	3	RS(B)=F, OUT=F
	4	RS(B)=F/2, Q=Q/2, OUT=F
	5	RS(B)=F/2, OUT=F
	6	RS(B)=2F, Q=2Q, OUT=F
	7	RS(B)=2F, OUT=F
A-ADDRESS/B-ADDRESS	0-15	A-ADDRESS IS ALWAYS A READ ADDRESS. B-ADDRESS IS A READ AND WRITE ADDRESS (THE SAME REGISTER STACK LOCATION CAN ALSO BE MODIFIED).
SELECT:	0	NOOP
	1	WHENEVER RS(B) IS REFERENCED, THE B-ADDRESS IS INHIBITED AND RS(LC1) IS USED INSTEAD. (LC1=LOOP COUNTER 1)

Note that the APM does not have the SELECT option.

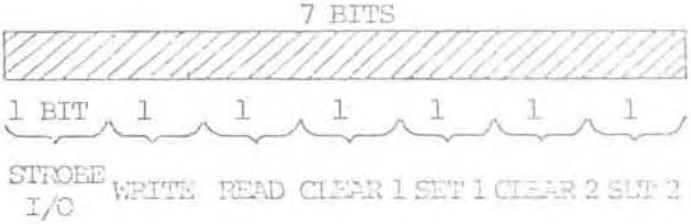
VII. ARI/ACC INSTRUCTION SET

ABBREVIATIONS: ARI - Arithmetical part of data processing; ACC - Data accumulators; ALU - Arithmetical logical unit; M1A - Operand A selection to multiplier 1; M1B - Operand B selection to multiplier 1; S1 - Strobe signals to multiplier 1; M12 - Arithmetical function on results from multiplier 1 and 2.

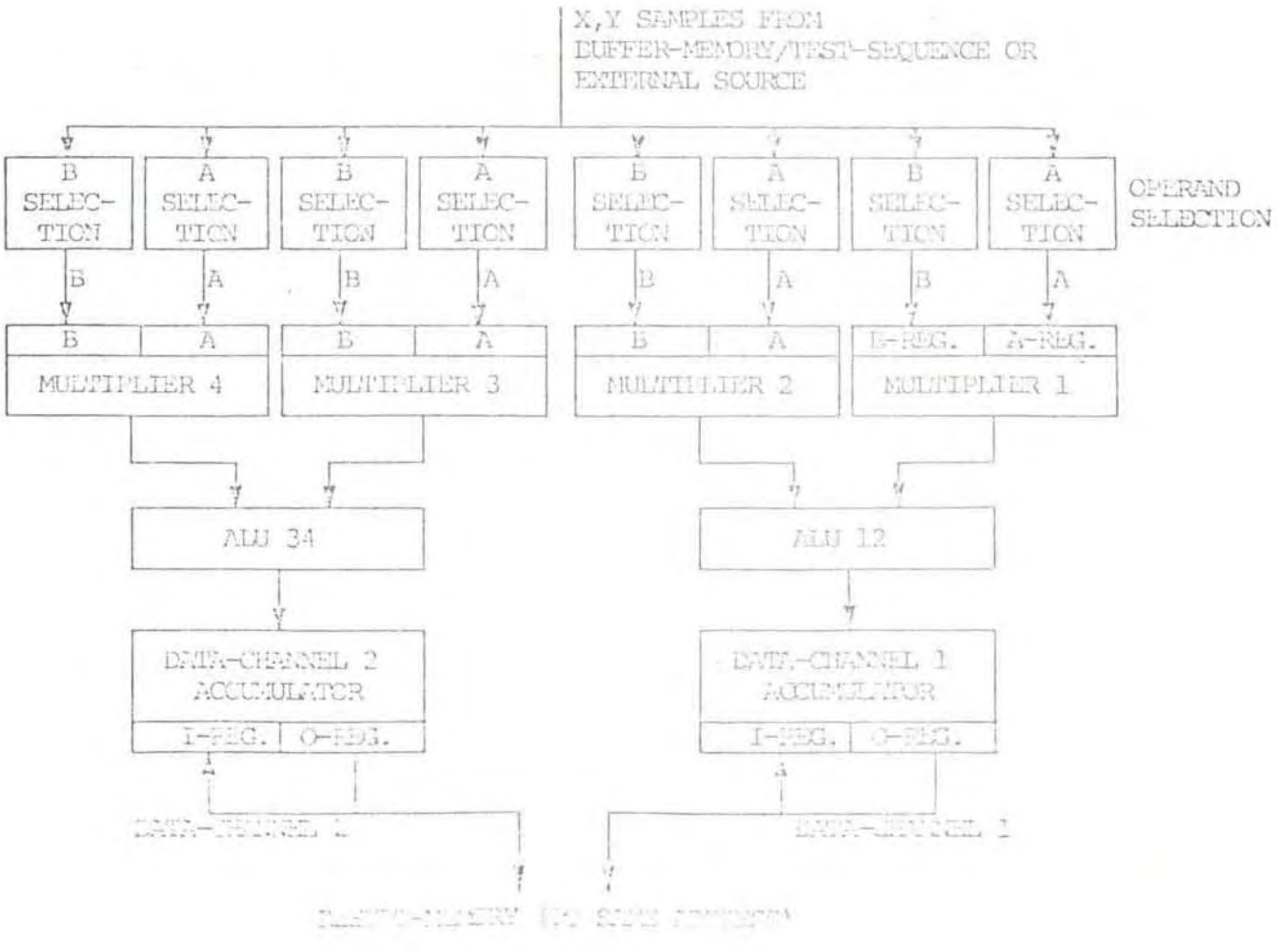
ARI INSTRUCTION-WORD SEPARATION:



ACC INSTRUCTION-WORD SEPARATION:



HARDWARE STRUCTURE:



	INTEGER CODING	FUNCTION	COMMENTS
M1A:	0	$A=X_{INTERNAL}$	SAME FOR ALL MULTIPLIERS
	1	$A=Y_{INTERNAL}$	
	2	$A=X_{EXTERNAL}$	
	3	$A=Y_{EXTERNAL}$	
	4-7	A=1	
M2B:	0	$B=X_{INTERNAL}$	SAME FOR ALL MULTIPLIERS
	1	$B=Y_{INTERNAL}$	
	2	$B=X_{EXTERNAL}$	
	3	$B=Y_{EXTERNAL}$	
S1:	0	NOOP	SAME FOR ALL MULTIPLIERS
	1	LOAD NEW OPERAND INTO A-REG.	
	2	LOAD NEW OPERAND INTO B-REG.	
	3	LOAD NEW OPERANDS INTO A- AND B-REGS.	
M12:	5	ALU 12=M2	SAME FOR ALU 34
	6	ALU 12=M1-M2	
	9	ALU 12=M1+M2	
	12	ALU 12=-1	
	15	ALU 12=M1	
STROBE I/O:	0	NOOP	DATA CHANNEL 1 AND 2 ACCUMULATORS ARE OPERATED IN PARALLEL
	1	STROBE NEW CONTENT INTO I/O REGS.	
WRITE:	0	NOOP	
	1	WRITE O-REG. TO RESULT MEMORY	
READ:	0	INTERNAL ACCUMULATION SELECTED	DEPENDENT ON VALUE OF SET 1 AND SET 2
	1	READ RESULT MEMORY TO I-REG. OR SET I-REG.=O	

	INTEGER CODING	FUNCTION	COMMENTS
SET 1:	0	IF READ=1 AND SET 2=0 THEN RESET I-REG (=0)	SET 1 IS ACTIVE UNTIL CLEAR 1=1
	1	IF READ=1 THEN GENERATE READ	
CLEAR 1:	0	NOOP	
	1	RESET SET 1=0	
SET 2:	0	ENABLE SET 1 CONTROL OF READ	SET 2 IS ACTIVE UNTIL CLEAR 2=1
	1	INHIBIT SET 1 CONTROL OF READ	SET 2 CAN ALSO BE SET FROM STATUSWORD
CLEAR 2:	0	NOOP	
	1	RESET SET 2=0	

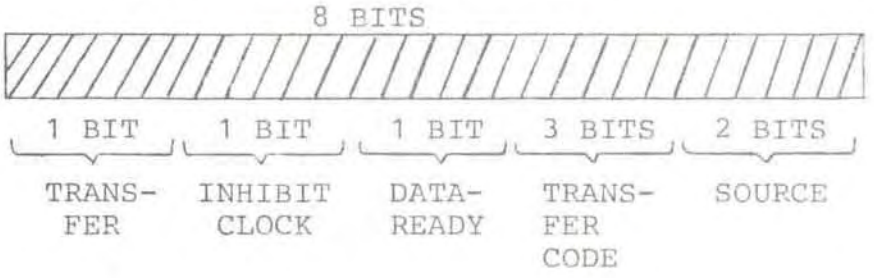
PROGRAMMING RESTRICTIONS:

It is allowed to select data read, accumulate and data write to the same result memory in one instruction cycle. However, it is not allowed to have a read and/or write operation to the same memory location in the next instruction cycle. This is due to the pipelining structure of the correlator.

VIII. OUT INSTRUCTION SET

ABBREVIATIONS: DMA - Direct memory access; LS - Least significant part; MS - Most significant part.

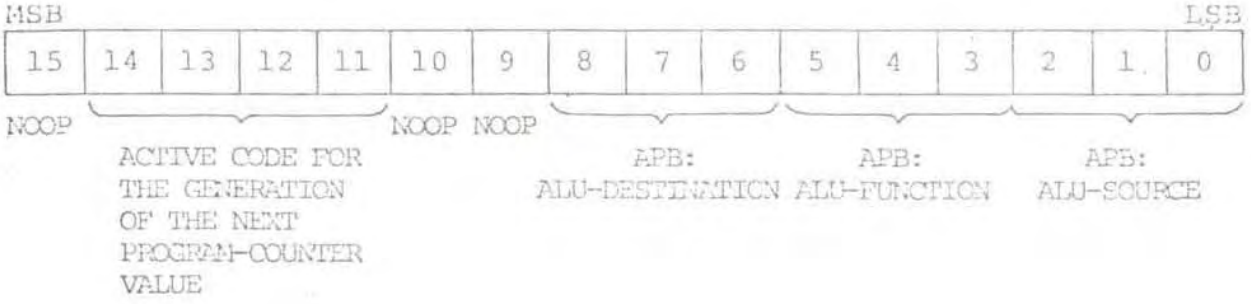
INSTRUCTION-WORD SEPERATION:



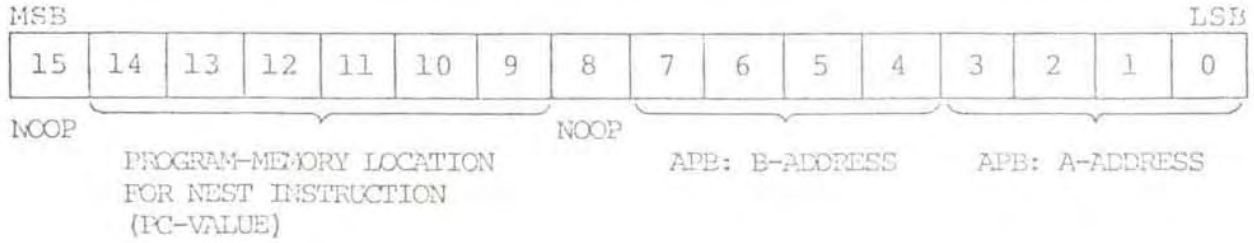
	INTEGER CODING	FUNCTION	COMMENTS
TRANSFER:	0	NOOP	TRANSFER NOT SELECTED
	1	ENABLE CORRELATOR FOR TRANSFER	MUST BE SET FOR TRANSFER
INHIBIT CLOCK:	0	ENABLE INTERNAL CLOCK OPERATION	
	1	INHIBIT CLOCK AND PROGRAM IS STOPPED	REQUIRES THAT DATA-READY IS ALWAYS SET
DATA-READY:	0	NOOP	
	1	SET "DATA-READY" TO CAMAC	"DATA-RECEIVED" FROM CAMAC. PROGRAM ADVANCES ONE INSTRUC-TION CYCLE
TRANSFER-CODE:	0	STATUS WORD	ACTIVE ON DMA-CHANNEL
	1	CONTROL WORD	AND FRONT PANEL
	2	RES. MEM, DATA-CHAN1:LS	GIVEN BY SOURCE INSTR.
	3	RES. MEM, DATA-CHAN1:MS	" " " "
	4	RES. MEM, DATA-CHAN2:LS	" " " "
	5	RES. MEM, DATA-CHAN2:MS	" " " "
	6	TEST-WORD 1	
	7	TEST-WORD 2	

	INTEGER CODING	FUNCTION	COMMENTS
SOURCE:	0	MEM. VALUES FROM MASTER MODULE	
	1	MEM. VALUES FROM SLAVE 1	MUST BE USED ONLY FOR TRANSFER CODES: 2-5
	2	MEM . VALUES FROM SLAVE 2	
	3	MEM. VALUES FROM SLAVE 3	

TEST-WORD 1 STRUCTURE:



TEST-WORD 2 STRUCTURE:



PROGRAMMING RESTRICTIONS:

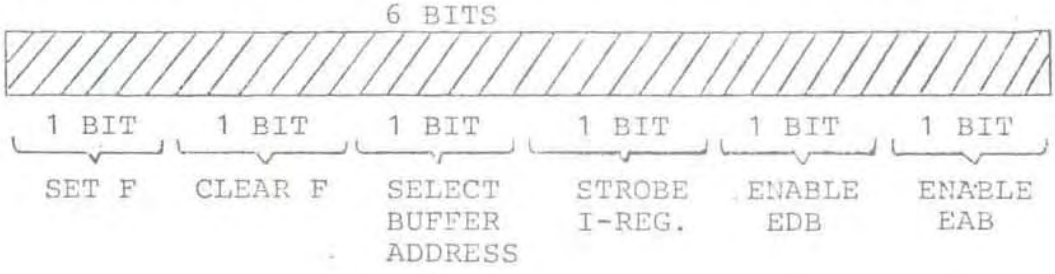
All Transfer programs should start at memory location 32. This value corresponds to the entry-point automatically loaded when the real-time command "data-transfer" is received from the radar-controller.

The hardware construction requires that before TRANSFER is set to 1 there must be at least 2 dummy instructions before it and that before TRANSFER is set to 0 (end of Transfer program) there must be at least 4 dummy instructions without INHIBIT CLOCK before it.

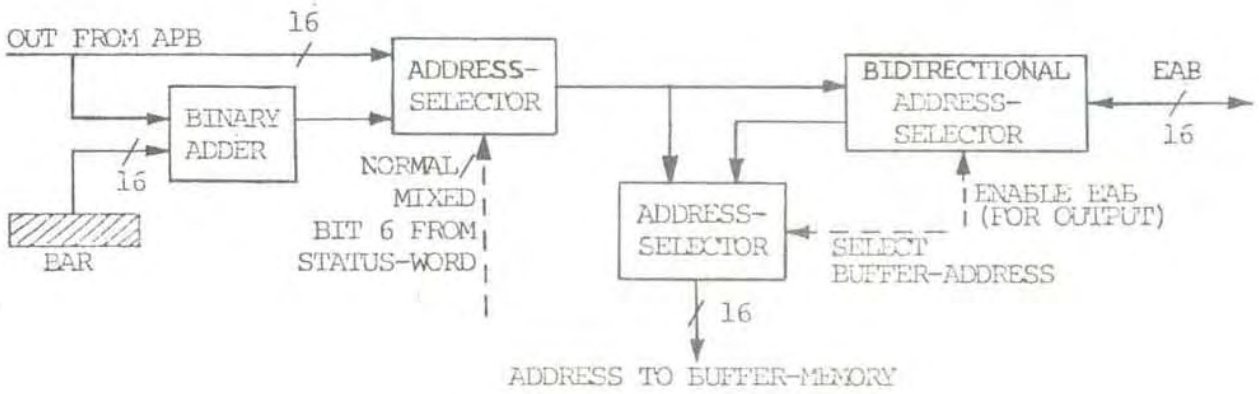
IX. I/O INSTRUCTION SET

ABBREVIATIONS: BAR - Base address register; EDB - External data bus; EAB - External address bus; APB - Address processor for buffer memory.

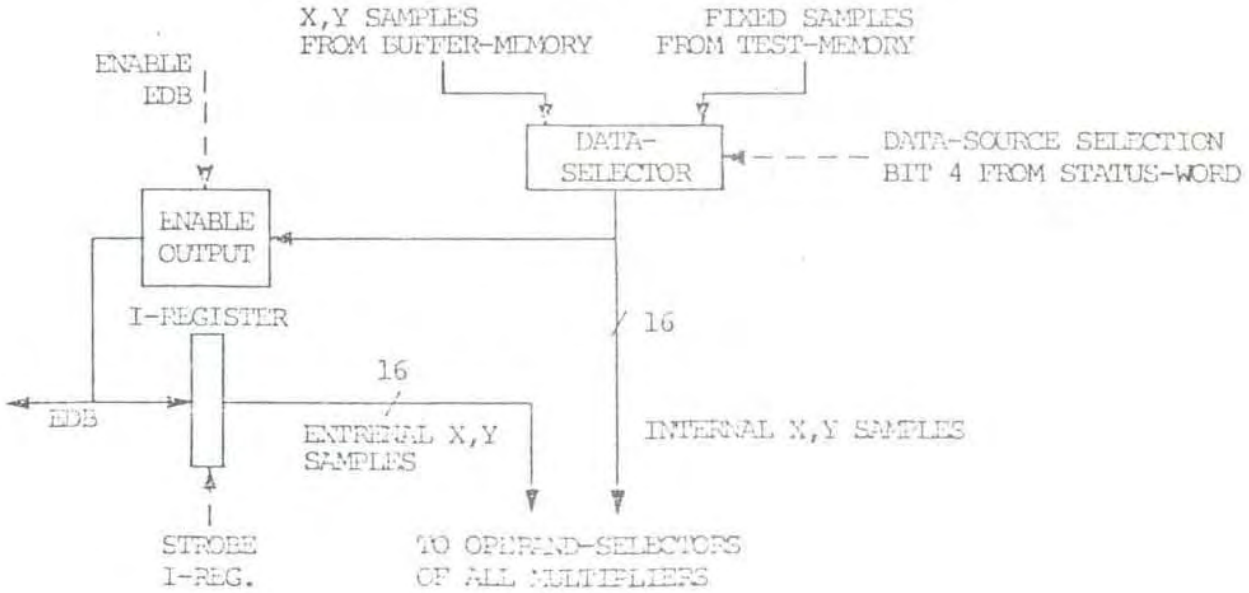
INSTRUCTION-WORD SEPARATION:



BUFFER-MEMORY ADDRESS GENERATION:



DATA-INPUT STRUCTURE:



	INTEGER CODING	FUNCTION
SET F:	0	NOOP
	1	SET FLAG
CLEAR F:	0	NOOP
	1	CLEAR FLAG
SELECT BUFFER ADDRESS:	0	BUFFER ADDRESS GENERATED BY INTERNAL HARDWARE
	1	BUFFER ADDRESS GERERATED BY EAB
STROBE I-REG.:	0	NOOP
	1	STROBE
ENABLE EDB:	0	NOOP
	1	INTERNAL X,Y SAMPLES ACTIVE ON EDB
ENABLE EAB:	0	NOOP
	1	INTERNAL ADDRESS ACTIVE ON EAB

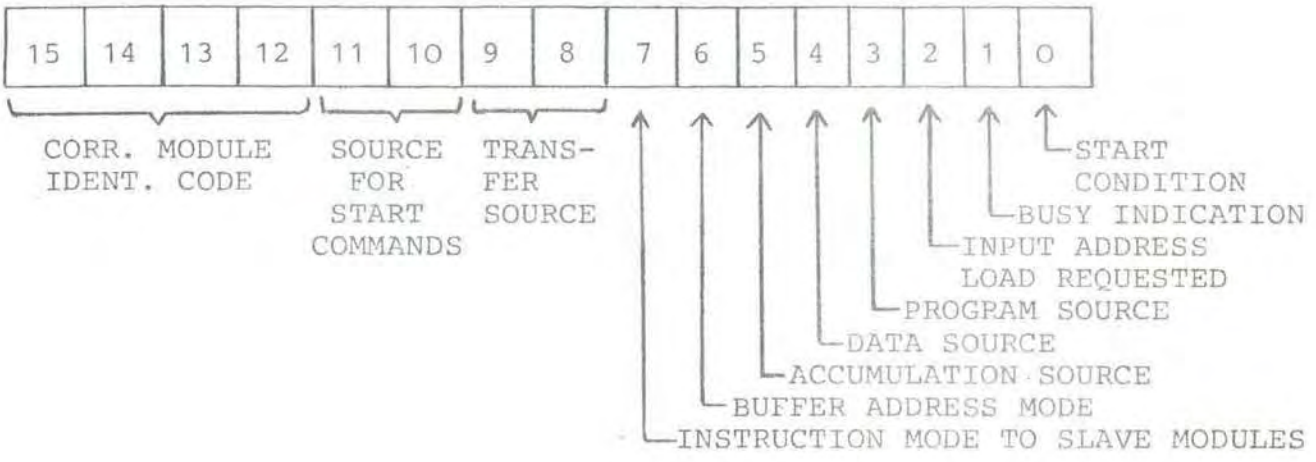
When the normal addressing mode is selected, set by bit 6 in the Statusword, the first internal address selector is always the output value from the APB. When the mixed addressing mode is selected the first internal address selector generates the output value from the APB plus the content of BAR in the first half of the instruction cycle, in the second half the output value from the APB is generated.

The operand register of the multipliers are (when selected) strobed at the end of the second half of the instruction cycle. This gives that when the internal address is selected for the buffer memory the samples with the address reference output value from the APB are always strobed (independent of normal/mixed mode). The I-register connected to the EDB is strobed at the end of the first half of the instruction cycle and can be used for temporary storage of samples with the address reference output value from the APB plus the content of BAR (only when mixed addressing mode is selected).

The correlator can generate a flag signal (rear side connector) which can be set/cleared by the SET F/CLEAR F bits in the instruction word. Both the external data and address busses (EDB and EAB) are bidirectional and are intended for use in multi-correlator systems for possible data/address value transfers between modules.

X. THE STATUSWORD

STATUSWORD FUNCTIONS



BIT		FUNCTION	COMMENTS
15-12	(P1)	CORRELATOR MODULE IDENT. CODE (0-15)	MUST BE SET TO VALUE SHOWN ON FRONT PANEL
11-10	(P2)	0: FRONT PANEL	ENABLE THE GIVEN
		1: RADAR CONTROLLER	SOURCE FOR STARTING
		2: COMPUTER	THE CORRELATOR
9-8		0: MASTER MODULE	SET BY TRANSFER
		1: SLAVE 1	INSTRUCTION
		2: SLAVE 2	
		3: SLAVE 3	
7	(P3)	0: SAME ARI INSTRUCTIONS TO SLAVES	
		1: MODIFIED OPERAND SELECTIONS TO SLAVE MULTIPLIERS	SWITCHES INTERNAL DATA TO EXTERNAL DATA
6	(P4)	0: NORMAL ADDRESS MODE	SEE I/O INSTRUCTIONS
		1: MIXED ADDRESS MODE	
5	(P5)	0: START EXPERIMENT	IF SET 1=0, ACC INSTR.
		1: CONTINUE EXPERIMENT	GENERATE SET 2=1
4	(P6)	0: INTERNAL DATA FROM BUFFER MEMORY	SEE I/O INSTRUCTIONS
		1: INTERNAL DATA FROM TEST MEMORY	
3	(P7)	0: PROGRAMMABLE MEMORY	RAM MODULES
		1: MEMORY FOR FIXED INTERNAL PROGRAMS	USE OF PROM WITH SAR

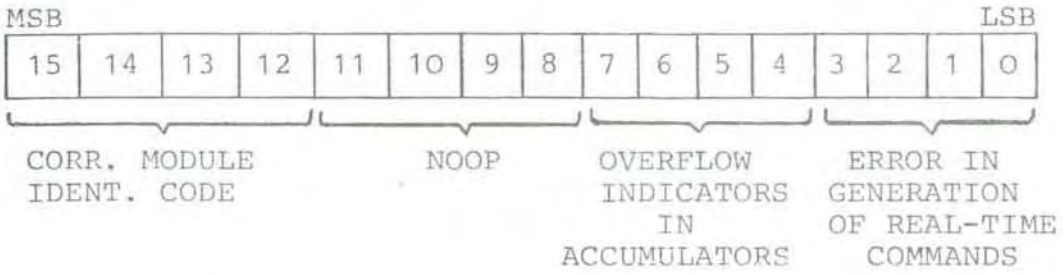
BIT	FUNCTION	COMMENTS
2	0: READY FOR LOADING REGS./ PROG. MEMORY LOCATIONS	
	1: ADDRESS TO DATA OR PROGRAM FIELD LOADED	
1	0: CORRELATOR NOT ACTIVE	SIGNAL ALSO AVAILABLE AT
	1: CORRELATOR BUSY	CONNECTOR ON REAR SIDE
0	0: NOT READY FOR ACCEPTING START COMMAND	
	1: CORRELATOR READY FOR START	SET BY DATA FIELD ADDRESS 63 (CRA)

P denotes which functions can be set in the Statusword.

In CORRSIM the functions must be entered in the following order: P1,P2,P3,P4,P5,P6,P7,. Note that each parameter must be separated with a comma.

XI. THE CONTROL WORD

CONTROL WORD FUNCTIONS



The given function is active when bit is set:

BIT	FUNCTION
0	INPUT PROGRAM REQUEST HAS BEEN ISSUED BUT CORRELATOR BUSY
1	START SIGNAL RECEIVED FROM COMPUTER OR RADAR CONTROLLER BUT CORRELATOR IN MANUAL OPERATION
2	START SIGNAL RECEIVED BUT CORRELATOR BUSY
3	START SIGNAL RECEIVED BUT CORRELATOR NOT READY
4	ACCUMULATOR OVERFLOW IN SLAVE 1
5	ACCUMULATOR OVERFLOW IN SLAVE 2
6	ACCUMULATOR OVERFLOW IN SLAVE 3
7	ACCUMULATOR OVERFLOW IN MASTER MODULE

When set the error bits remain active until the real time command correlator reset is given.

The Control word is shown on the front panel.

XII. REAL-TIME COMMANDS

All commands from computer or radar-controller are termed real-time commands. These commands have the following internal correlator response when the correlator is in remote-status (switch setting on front-panel).

REAL-TIME COMMAND	SOURCE	INTERNAL RESPONSE
START COMPUTE:	COMPUTER OR RADAR-CONTROLLER	TEST ON CORRELATOR READY AND BUSY AND ENABLING OF START-SOURCE. IF NOT GRANTED: SET ERROR BIT. IF GRANTED: START INTERNAL PROGRAM AT ENTRY-POINT: PC = SAR SET BIT 1 IN STATUSWORD.
START TRANSFER:	RADAR-CONTROLLER	SAME AS FOR START COMPUTE. ENTRY-POINT AT PC = 32
RESET:	COMPUTER	SET PC = 00, RESET ERROR BITS IN CONTROLWORD, RESET CORRELATOR READY AND BIT 1 IN STATUSWORD.
ADDRESS LOAD	COMPUTER	TEST ON CORRELATOR BUSY. IF NOT GRANTED: SET ERROR BIT. IF GRANTED: LOAD ADDRESS FOR DATA-LOAD TO DATA- OR PROGRAM-FIELD. SET BIT 2 IN STATUSWORD.
DATA LOAD:	COMPUTER	LOADS DATA WITH ADDRESS REFERENCE GIVEN ABOVE. RESET BIT 2 IN STATUSWORD:
INTERRUPT PROGRAM:	EXTERNAL SOURCE	WHEN FIRST CONTINUE-STATEMENT IS REACHED IN THE INTERNAL PROGRAM, THIS STATEMENT ADDRESS IS STORED, AND A JUMP TO PC = 63 IS PERFORMED. PROGRAM STOPS.
INTERRUPT RESET:	EXTERNAL SOURCE	INTERNAL PROGRAM IS RESTARTED AT BREAKPOINT GIVEN ABOVE.

XIII. REAL-TIME SIGNALS FROM THE CORRELATOR

SIGNAL	FUNCTION
CORRELATOR RUN:	BIT 1 FROM STATUSWORD, ACTIVE WHENEVER PC ≠ 00 AND PC ≠ 63
FLAG:	SET/RESET BY PROGRAM (I/O-INSTRUCTION)
DMA-REQUEST:	REQUEST OF COMPUTER DMA-CYCLE; SET BY DATA-READY
DATA-READY:	COMMUNICATION SIGNAL TO CAMAC WHEN DATA IS READY FOR TRANSFER. SET BY OUT-INSTRUCTION
ERROR-INTERRUPT:	ACTIVE WHEN ONE OF THE ERROR BITS IN CONTROL-WORD IS SET.

XIV. FRONT-PANEL COMMANDS

All real-time commands can be generated on the front-panel when the correlator is in manual status (front-panel switch).

POWER-ON PROCEDURE:
Set power on and push master reset button. Select status: Manual or remote.

DATA/PROGRAM-FIELD LOADING FROM FRONT-PANEL:
Manual status must be selected. Set addr., sub-addr., on chip-switches:

	11	10	9	8	7	6	5	4	3	2	1	0	SWITCH NO.
ADDRESS FORMAT	x	x	x	x	x	x	x	x	x	x	x	x	

ADDR. SUB-ADDR.

Push address-load (correlator must not be busy) and led-indicator becomes active. Set data value on chip-switches and push data-load. Load-sequence is finished. Repeate for new loading.

DATA-DISPLAY FUNCTIONS ON FRONT-PANEL (MANUAL STATUS):
Set display switch to "Front-panel". Led-indicators are showing:

DISPLAY CODE (DECIMAL)

0:	STATUSWORD	
1:	CONTROLWORD	
2:	DATA CHANNEL 1, LS-PART	
3:	DATA CHANNEL 1, MS-PART	RES-MEMORY ADDRESS
4:	DATA CHANNEL 2, LS-PART	GIVEN BY APM
5:	DATA CHANNEL 2, MS-PART	
6:	TESTWORD 1	
7:	TESTWORD 2	SEE OUT-INSTRUCTIONS

When display switch is set to program, internal program-instruction out is active. (For PC = 0 display-code is always 0: Statusword).

DISPLAY OF RESULT-MEMORY CONTENT (MANUAL STATUS)
Set display switch to "Front-panel" and load APM-instruction word 7, 0, 1, ØØ, ØØ into program-location 00. By display-code 2, 3, 4 or 5 and memory-address set on chip-switches, memory-content can be read on the led-display.

This function can be used only when PC = 00.

TEST OPTIONS:

SINGLE-CYCLE PROGRAM-EXECUTION (MANUAL STATUS):

Set front-panel switch to "Clock-Single". In this mode single instruction-cycle is generated when "Clock-Advance" is pushed. When display-switch is set to "Front-Panel" all display selections are available. When a front-panel function is required, push function-switch and then push "Clock-Advance".

DMA TRANSFER-PROGRAM TESTING (MANUAL STATUS):

Set switch to "Transfer Inhibit". In this mode data-ready to CAMAC is inhibited, and program will stop when transfer-instructions are active. Simulation of CAMAC-response can be made by pushing "Data-Received".

EISCAT publications

F. du Castel, O. Holt, B. Hultqvist, H. Kohl and M. Tiuri:
A European Incoherent Scatter Facility in the Auroral Zone (EISCAT).
A Feasibility Study ("The Green Report") June 1971. (Out of print).

O. Bratteng and A. Haug:
Model Ionosphere at High Latitude, EISCAT Feasibility Study, Report No. 9.
The Auroral Observatory, Tromsö July 1971. (Out of print).

A European Incoherent Scatter Facility in the Auroral Zone, UHF System and Organization ("The Yellow Report"), June 1974.

EISCAT Annual Report 1976.(Out of print).

P.S. Kildal and T. Hagfors:
Balance between investment in reflector and feed in the VHF cylindrical antenna.
EISCAT Technical Notes No. 77/1, 1977.

T. Hagfors:
Least mean square fitting of data to physical models.
EISCAT Technical Notes No. 78/2, 1978.

T. Hagfors:
The effect of ice on an antenna reflector.
EISCAT Technical Notes No. 78/3, 1978.

T. Hagfors:
The bandwidth of a linear phased array with stepped delay corrections.
EISCAT Technical Notes No. 78/4, 1978.

Data Group meeting in Kiruna, Sweden, 18-20 Jan. 1978
EISCAT Meetings No. 78/1, 1978

EISCAT Annual Report 1977

H-J. Alker:
Measurement principles in the EISCAT system
EISCAT Technical Notes No. 78/5, 1978

EISCAT Data Group meeting in Tromsö, Norway 30-31 May, 1978
EISCAT Meetings No. 78/2, 1978.

P-S. Kildal:
Discrete phase steering by permuting precut phase cables.
EISCAT Technical Notes No. 78/6, 1978

EISCAT UHF antenna acceptance test.
EISCAT Technical Notes No. 78/7, 1978.

P-S. Kildal:
Feeder elements for the EISCAT VHF parabolic cylinder antenna.
EISCAT Technical Notes No. 78/8, 1978.

H-J. Alker:
Program CORRSIM: System for program development and software simulation of EISCAT digital correlator, User's Manual.
EISCAT Technical Notes No. 79/9, 1979.

H-J. Alker:
Instruction manual for EISCAT digital correlator.
EISCAT Technical Notes No. 79/10, 1979

H-J. Alker:
A programmable correlator module for the EISCAT radar system.
EISCAT Technical Notes No. 79/11, 1979.

T. Ho and H-J. Alker:
Scientific programming of the EISCAT digital correlator.
EISCAT Technical Notes No. 79/12, 1979.

S. Westerlund (editor):
Proceedings EISCAT Annual Review Meeting 1969. Part I and II,
Abisko, Sweden, 12-16 March 1979.
EISCAT Meetings No. 79/3, 1979.

J. Murdin:
EISCAT UHF Geometry.
EISCAT Technical Notes No. 79/13, 1979.

T. Hagfors:
Transmitter Polarization Control in the EISCAT UHF System.
EISCAT Technical Notes No. 79/14, 1979.

B. Törustad:
A description of the assembly language for the EISCAT digital correlator.
EISCAT Technical Notes No. 79/15, 1979.

J. Murdin:
Errors in incoherent scatter radar measurements.
EISCAT Technical Notes No. 79/16, 1979.

EISCAT Digital Correlator. TEST MANUAL.
EISCAT Technical Notes No. 79/17, 1979.

G. Lejeune:
A program library for incoherent scatter calculation.
EISCAT Technical Notes No. 79/18, 1979.

K. Folkestad:
Lectures for EISCAT Personnel, Volume I
EISCAT Technical Notes No. 79/19, 1979.

Svein A. Kvalvik:
Correlator Buffer-Memory for the EISCAT Radar system
EISCAT Technical Notes. No. 80/20.

P-S. Kildal:
EISCAT VHF Antenna Tests
EISCAT Technical Notes No. 80/21

J. Armstrong:
EISCAT Experiment Preparation Manual
EISCAT Technical Notes No. 80/22

A. Farmer:
EISCAT Data Gathering and Dissemination
EISCAT Technical Note 80/23

Terrance Ho and Hans-Jørgen Alker:
Scientific Programming of the EISCAT Digital Correlator (Revised)
EISCAT Technical Note 81/24

Terrance Ho:
Programs Corrsim, Corrtest: System for Program Development and
Software Simulation of EISCAT Digital Correlator, User's manual.
EISCAT Technical Note 81/25

Terrance Ho:
Instruction Manual for EISCAT Digital Correlator (Revised).
EISCAT Technical Note 81/26

Terrance Ho:
Standard Subroutines and Programs for EISCAT Digital Correlator.
EISCAT Technical Note 81/27

Terrance Ho:
Pocket Manual for Programming the EISCAT Digital Correlator.
EISCAT Technical Note 81/28

K. Folkestad:
Lectures for EISCAT Personnel, Volume II.
EISCAT Technical Note 81/29

M. Lehtinen och Anna-Liisa Turunen:
EISCAT UHF antenna direction calibration
EISCAT Technical Note 81/30

K. Folkestad:
Use of the EISCAT Radar as a supplement to rocket measurements.
EISCAT Technical Note 81/31