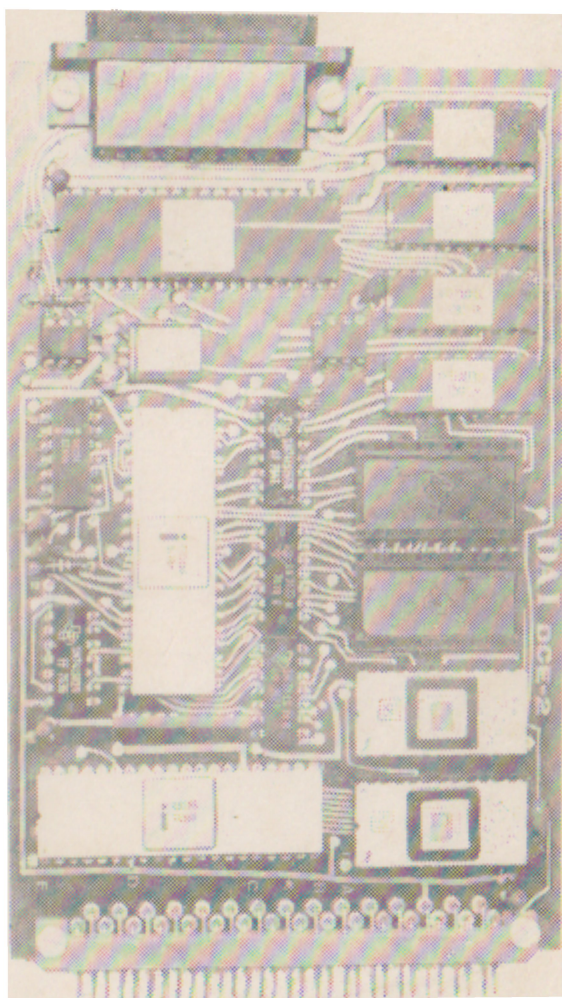


DAI

**DCE SINGLE EUROCARD
MICROCOMPUTERS**

REFERENCE CARD



**DATA APPLICATIONS
INTERNATIONAL**

00	NOP		3E	MVI	A,dd	7C	MOV	A,H
01	LXI	B,addr	3F	CMC		7D	MOV	A,L
02	STAX	B	40	MOV	B,B	7E	MOV	A,M
03	INX	B	41	MOV	B,C	7F	MOV	A,A
04	INR	B	42	MOV	B,D	80	ADD	B
05	DCR	B	43	MOV	B,E	81	ADD	C
06	MVI	B,dd	44	MOV	B,H	82	ADD	D
07	RLC		45	MOV	B,L	83	ADD	E
08	---		46	MOV	B,M	84	ADD	H
09	DAD	B	47	MOV	B,A	85	ADD	L
0A	LDAX	B	48	MOV	C,B	86	ADD	M
0B	DCX	B	49	MOV	C,C	87	ADD	A
0C	INR	C	4A	MOV	C,D	88	ADC	B
0D	DCR	C	4B	MOV	C,E	89	ADC	C
0E	MVI	C,dd	4C	MOV	C,H	8A	ADC	D
0F	RRC		4D	MOV	C,L	8B	ADC	E
10	---		4E	MOV	C,M	8C	ADC	H
11	LXI	D,addr	4F	MOV	C,A	8D	ADC	L
12	STAX	D	50	MOV	D,B	8E	ADC	M
13	INX	D	51	MOV	D,C	8F	ADC	A
14	INR	D	52	MOV	D,D	90	SUB	B
15	DCR	D	53	MOV	D,E	91	SUB	C
16	MVI	D,dd	54	MOV	D,H	92	SUB	D
17	RAL		55	MOV	D,L	93	SUB	E
18	---		56	MOV	D,M	94	SUB	H
19	DAD	D	57	MOV	D,A	95	SUB	L
1A	LDAX	D	58	MOV	E,B	96	SUB	M
1B	DCX	D	59	MOV	E,C	97	SUB	A
1C	INR	E	5A	MOV	E,D	98	SBB	B
1D	DCR	E	5B	MOV	E,E	99	SBB	C
1E	MVI	E,dd	5C	MOV	E,H	9A	SBB	D
1F	RAR		5D	MOV	E,L	9B	SBB	E
20	---		5E	MOV	E,M	9C	SBB	H
21	LXI	H,addr	5F	MOV	E,A	9D	SBB	L
22	SHLD	addr	60	MOV	H,B	9E	SBB	M
23	INX	H	61	MOV	H,C	9F	SBB	A
24	INR	H	62	MOV	H,D	A0	ANA	B
25	DCR	H	63	MOV	H,E	A1	ANA	C
26	MVI	H,dd	64	MOV	H,H	A2	ANA	D
27	DAA		65	MOV	H,L	A3	ANA	E
28	---		66	MOV	H,M	A4	ANA	H
29	DAD	H	67	MOV	H,A	A5	ANA	L
2A	LHLD	addr	68	MOV	L,B	A6	ANA	M
2B	DCX	H	69	MOV	L,C	A7	ANA	A
2C	INR	L	6A	MOV	L,D	A8	XRA	B
2D	DCR	L	6B	MOV	L,E	A9	XRA	C
2E	MVI	L,dd	6C	MOV	L,H	AA	XRA	D
2F	CMA		6D	MOV	L,L	AB	XRA	E
30	---		6E	MOV	L,M	AC	XRA	H
31	LXI	SP,addr	6F	MOV	L,A	AD	XRA	L
32	STA	addr	70	MOV	M,B	AE	XRA	M
33	INX	SP	71	MOV	M,C	AF	XRA	A
34	INR	M	72	MOV	M,D	B0	ORA	B
35	DCR	M	73	MOV	M,E	B1	ORA	C
36	MVI	M,dd	74	MOV	M,H	B2	ORA	D
37	STC		75	MOV	M,L	B3	ORA	E
38	---		76	HLT		B4	ORA	H
39	DAD	SP	77	MOV	M,A	B5	ORA	L
3A	LDA	addr	78	MOV	A,B	B6	ORA	M
3B	DCX	SP	79	MOV	A,C	B7	ORA	A
3C	INR	A	7A	MOV	A,D	B8	CMP	B
3D	DCR	A	7B	MOV	A,E	B9	CMP	C

BA	CMP	D	D2	JNC	addr	EA	JPE	addr
BB	CMP	E	D3	OUT	dd	EB	XCHG	
BC	CMP	H	D4	CNC	addr	EC	CPE	addr
BD	CMP	L	D5	PUSH	D	ED	---	
BE	CMP	M	D6	SUI	dd	EE	XRI	dd
BF	CMP	A	D7	RST	2	EF	RST	5
CO	RNZ		D8	RC		FO	RP	
C1	POP	B	D9	---		F1	POP	PSW
C2	JNZ	addr	DA	JC	addr	F2	JP	addr
C3	JMP	addr	DB	IN	dd	F3	DI	
C4	CNZ	addr	DC	CC	addr	F4	CP	addr
C5	PUSH	B	DD	---		F5	PUSH	PSW
C6	ADI	dd	DE	SBI	dd	F6	ORI	dd
C7	RST	0	DF	RST	3	F7	RST	6
C8	RZ		EO	RPO		F8	RM	
C9	RET		E1	POP	H	F9	SPHL	
CA	JZ		E2	JPO	addr	FA	JM	addr
CB	---		E3	XTHL		FB	EI	
CC	CZ	addr	E4	CPO	addr	FC	CM	addr
CD	CALL	addr	E5	PUSH	H	FD	---	
CE	ACI	addr	E6	ANI	dd	FE	CPI	dd
CF	RST	1	E7	RST	4	FF	RST	7
DO	RNC		E8	RPE				
D1	POP	D	E9	PCHL				

ASCII - HEX - ASCII CONVERSION TABLE

	MSD	0	1	2	3	4	5	6	7
LSD		000	001	010	011	100	101	110	111
0	0000	NUL	DLE	SP	0	@	P	,	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	"	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	\$	4	D	T	d	t
5	0101	ENG	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	'	7	G	W	g	w
8	1000	BS	CAN	(8	H	X	h	x
9	1001	HT	EM)	9	I	Y	i	y
A	1010	LF	SUB	*	:	J	Z	j	z
B	1011	VT	ESC	+	;	K	[k	{
C	1100	FF	FS	,	<	L	\	l	
D	1101	CR	GS	-	=	M]	m	}
E	1110	SO	RS	.	>	N	↑	n	~
F	1111	SI	VS	/	?	O	←	o	DEL

INSTRUCTIONFUNCTIONHEXMOVE GROUP

		reg	A	B	C	D	E	H	L	M
MOV A, reg	$(A) \leftarrow (\text{reg})$		7F	78	79	7A	7B	7C	7D	7E
MOV B, reg	$(B) \leftarrow (\text{reg})$		47	40	41	42	43	44	45	46
MOV C, reg	$(C) \leftarrow (\text{reg})$		4F	48	49	4A	4B	4C	4D	4E
MOV D, reg	$(D) \leftarrow (\text{reg})$		57	50	51	52	53	54	55	56
MOV E, reg	$(E) \leftarrow (\text{reg})$		5F	58	59	5A	5B	5C	5D	5E
MOV H, reg	$(H) \leftarrow (\text{reg})$		67	60	61	62	63	64	65	66
MOV L, reg	$(L) \leftarrow (\text{reg})$		6F	68	69	6A	6B	6C	6D	6E
MOV M, reg	$(M) \leftarrow (\text{reg})$		77	70	71	72	73	74	75	--

ACCUMULATOR GROUP

ADD reg	$(A) \leftarrow (A) + (\text{reg})$	*	87	80	81	82	83	84	85	86
ADC reg	$(A) \leftarrow (A) + (\text{reg}) + (\text{CY})$	*	8F	88	89	8A	8B	8C	8D	8E
SUB reg	$(A) \leftarrow (A) - (\text{reg})$	*	97	90	91	92	93	94	95	96
SBB reg	$(A) \leftarrow (A) - (\text{reg}) - (\text{CY})$	*	9F	98	99	9A	9B	9C	9D	9E
ANA reg	$(A) \leftarrow (A) \wedge (\text{reg})$	*	A7	A0	A1	A2	A3	A4	A5	A6
XRA reg	$(A) \leftarrow (A) \vee (\text{reg})$	*	AF	A8	A9	AA	AB	AC	AD	AE
ORA reg	$(A) \leftarrow (A) \vee (\text{reg})$	*	B7	B0	B1	B2	B3	B4	B5	B6
CMP reg	$(A) - (\text{reg})$	*	BF	B8	B9	BA	BB	BC	BD	BE

INCREMENT/DECREMENT REGISTER

INR reg	$(\text{reg}) \leftarrow (\text{reg}) + 1$	**	3C	04	0C	14	1C	24	2C	34
DCR reg	$(\text{reg}) \leftarrow (\text{reg}) - 1$	**	3D	05	0D	15	1D	25	2D	35

REGISTER PAIR GROUP

		rp	B	D	H	SP	PSW
INX rp	$(\text{rp}) \leftarrow (\text{rp}) + 1$		03	13	23	33	--
DCX rp	$(\text{rp}) \leftarrow (\text{rp}) - 1$		0B	1B	2B	3B	--
LDAX rp	$(A) \leftarrow (\text{rp})$		0A	1A	--	--	--
STAX rp	$(\text{rp}) \leftarrow (A)$		02	12	--	--	--
DAD rp	$(H, L) \leftarrow (H, L) + (\text{rp})$ ***		09	19	29	39	--
PUSH rp	$((\text{SP}) - 1) \leftarrow (\text{rh}), ((\text{SP}) - 2) \leftarrow (\text{rl}),$ $(\text{SP}) \leftarrow (\text{SP}) - 2$		C5	D5	E5	--	F5
POP rp	$(\text{rl}) \leftarrow ((\text{SP})), (\text{rh}) \leftarrow ((\text{SP}) + 1),$ $(\text{SP}) \leftarrow (\text{SP}) + 2$		C1	D1	E1	--	F1 *

DIRECT ADDRESS GROUP

LDA addr	$(A) \leftarrow (\text{addr})$		3A	al	ah		
STA addr	$(\text{addr}) \leftarrow (A)$		32	al	ah		
LHLD addr	$(L) \leftarrow (\text{addr}), (H) \leftarrow (\text{addr} + 1)$		2A	al	ah		
SHLD addr	$(\text{addr}) \leftarrow (L), (\text{addr} + 1) \leftarrow (H)$		22	al	ah		

IMMEDIATE GROUP

MVI A, data	$(A) \leftarrow \text{data}$		3E	dd			
MVI B, data	$(B) \leftarrow \text{data}$		06	dd			
MVI C, data	$(C) \leftarrow \text{data}$		0E	dd			
MVI D, data	$(D) \leftarrow \text{data}$		16	dd			
MVI E, data	$(E) \leftarrow \text{data}$		1E	dd			
MVI H, data	$(H) \leftarrow \text{data}$		26	dd			
MVI L, data	$(L) \leftarrow \text{data}$		2E	dd			
MVI M, data	$(M) \leftarrow \text{data}$		36	dd			
ADI data	$(A) \leftarrow (A) + \text{data}$	*	C6	dd			
ACI data	$(A) \leftarrow (A) + \text{data} + (\text{CY})$	*	CE	dd			
SUI data	$(A) \leftarrow (A) - \text{data}$	*	D6	dd			
SBI data	$(A) \leftarrow (A) - \text{data} - (\text{CY})$	*	DE	dd			
ANI data	$(A) \leftarrow (A) \wedge \text{data}$	*	E6	dd			
XRI data	$(A) \leftarrow (A) \vee \text{data}$	*	EE	dd			
ORI data	$(A) \leftarrow (A) \vee \text{data}$	*	F6	dd			
CPI data	$(A) - \text{data}$	*	FE	dd			
LXI B, addr	$(B) \leftarrow \text{ah}, (C) \leftarrow \text{al}$		01	al	ah		
LXI D, addr	$(D) \leftarrow \text{ah}, (E) \leftarrow \text{al}$		11	al	ah		
LXI H, addr	$(H) \leftarrow \text{ah}, (L) \leftarrow \text{al}$		21	al	ah		
LXI SP, addr	$(\text{SP}_H) \leftarrow \text{ah}, (\text{SP}_L) \leftarrow \text{al}$		31	al	ah		

INSTRUCTIONFUNCTIONHEXJUMP GROUP

JMP	addr	(PC) ← addr	C3	al	ah
JNZ	addr	If Z=0, (PC) ← addr	C2	al	ah
JZ	addr	If Z=1, (PC) ← addr	CA	al	ah
JNC	addr	If CY=0, (PC) ← addr	D2	al	ah
JC	addr	If CY=1, (PC) ← addr	DA	al	ah
JPO	addr	If P=0, (PC) ← addr	E2	al	ah
JPE	addr	If P=1, (PC) ← addr	EA	al	ah
JP	addr	If S=0, (PC) ← addr	F2	al	ah
JM	addr	If S=1, (PC) ← addr	FA	al	ah
PCHL		(PC _H) ← (H), (PC _L) ← (L)	E9		

CALL GROUP

CALL	addr	(TOS) ← (PC), (PC) ← addr	CD	al	ah
CNZ	addr	If Z=0, (TOS) ← (PC), (PC) ← addr	C4	al	ah
CZ	addr	If Z=1, (TOS) ← (PC), (PC) ← addr	CC	al	ah
CNC	addr	If CY=0, (TOS) ← (PC), (PC) ← addr	D4	al	ah
CC	addr	If CY=1, (TOS) ← (PC), (PC) ← addr	DC	al	ah
CPO	addr	If P=0, (TOS) ← (PC), (PC) ← addr	E4	al	ah
CPE	addr	If P=1, (TOS) ← (PC), (PC) ← addr	EC	al	ah
CP	addr	If S=0, (TOS) ← (PC), (PC) ← addr	F4	al	ah
CM	addr	If S=1, (TOS) ← (PC), (PC) ← addr	FC	al	ah

N.B. (TOS) ← (PC) designates the following:-
 ((SP)-1) ← (PC_H), ((SP)-2) ← (PC_L), (SP) ← (SP)-2

RETURN GROUP

RET		(PC) ← (TOS)	C9		
RNZ		If Z=0, (PC) ← (TOS)	C0		
RZ		If Z=1, (PC) ← (TOS)	C8		
RNC		If CY=0, (PC) ← (TOS)	D0		
RC		If CY=1, (PC) ← (TOS)	D8		
RPO		If P=0, (PC) ← (TOS)	E0		
RPE		If P=1, (PC) ← (TOS)	E8		
RP		If S=0, (PC) ← (TOS)	F0		
RM		If S=1, (PC) ← (TOS)	F8		

N.B. (PC) ← (TOS) designates the following:-
 (PC_L) ← ((SP)), (PC_H) ← ((SP)+1), (SP) ← (SP)+2

RESTART GROUP

RST	0	(TOS) ← (PC), (PC) ← 0 ₁₆	C7		
RST	1	(TOS) ← (PC), (PC) ← 8 ₁₆	CF		
RST	2	(TOS) ← (PC), (PC) ← 10 ₁₆	D7		
RST	3	(TOS) ← (PC), (PC) ← 18 ₁₆	DF		
RST	4	(TOS) ← (PC), (PC) ← 20 ₁₆	E7		
RST	5	(TOS) ← (PC), (PC) ← 28 ₁₆	EF		
RST	6	(TOS) ← (PC), (PC) ← 30 ₁₆	F7		
RST	7	(TOS) ← (PC), (PC) ← 38 ₁₆	FF		

ROTATE/CONTROL/SPECIAL GROUP

RLC		(A _{n+1}) ← (A _n), (A ₀) ← (A ₇), (CY) ← (A ₇)	***	07	
RRC		(A _n) ← (A _{n+1}), (A ₇) ← (A ₀), (CY) ← (A ₀)	***	0F	
RAL		(A _{n+1}) ← (A _n), (A ₀) ← (CY), (CY) ← (A ₇)	***	17	
RAR		(A _n) ← (A _{n+1}), (A ₇) ← (CY), (CY) ← (A ₀)	***	1F	
NOP		No operation		00	
HLT		Processor stopped until interrupt or reset		76	
DI		Interrupts disabled		F3	
EI		Interrupts enabled after next instruction		FB	
XTHL		(L) ← ((SP)), (H) ← ((SP)+1)		E3	
SPHL		(SP _H) ← (H), (SP _L) ← (L)		F9	
XCHG		(H) ← (D), (L) ← (E)		EB	
DAA		Decimal adjust accumulator	*	27	
CMA		(A) ← (A)		2F	
STC		(CY) ← 1	***	37	
CMC		(CY) ← (CY)	***	3F	
OUT	port	Not used in DCE Systems		D3	port
IN	port			DB	port

<u>INSTRUCTION</u>	<u>FUNCTION</u>	<u>HEX</u>	
<u>TICC GROUP</u>			
		<u>DCE 1/2</u>	<u>DCE-X</u>
TXMT	(Transmit buffer)←(A)	32 06 98	32 16 FF
LDRCV	(A)←(Receive buffer)	3A 00 98	3A 10 FF
STOUT	(output port)←(A)	32 07 98	32 17 FF
LDIN	(A)←(input port)	3A 01 98	3A 11 FF
STTIM 1	(Timer 1)←(A)	32 09 98	32 19 FF
STTIM 2	(Timer 2)←(A)	32 0A 98	32 1A FF
STTIM 3	(Timer 3)←(A)	32 0B 98	32 1B FF
STTIM 4	(Timer 4)←(A)	32 0C 98	32 1C FF
STTIM 5	(Timer 5)←(A)	32 0D 98	32 1D FF
LDSTA	(A)←(TICC status reg.)	3A 03 98	3A 13 FF
STTCM	(TICC Command reg.)←(A)	32 04 98	32 14 FF
STCRR	(Rate register)←(A)	32 05 98	32 15 FF
LDIPR	(A)←(Interrupt pending reg.)	3A 02 98	3A 12 FF
STIMR	(Interrupt Mask reg.)←(A)	32 08 98	32 18 FF

GIC GROUP

GICC am, bm	(GICC Cmd reg.)←cd	3E cd +	32 03 1C or 32 03	FF
BCLR n	(P2Bn)←0	3E 'cc +	32 03 1C or 32 03	FF
BSET n	(P2Bn)←1	3E cs +	32 03 1C or 32 03	FF
LDGI m	(A)←(Port m)		3A 0m 1C or 3A 0m	FF
STGI m	(Port m)←(A)		32 0m 1C or 32 0m	FF
LDGIS 0	(A)←(Port 0)		3A 00 5C or 3A 08	FF
STGIS 0	(Port 0)←(A)		32 00 5C or 32 08	FF

N.B. cd = $8016 + (810 \times am)16 + bm.$
 eg. for am = 3, bm = 3; cd = $8016 + (810 \times 3)16 + 3$
 = $8016 + 1816 + 3 = 9B$

- cc = $2 \times n$
- cs = $(2 \times n) + 1$
- m = 0, 1 or 2
- (reg)/(rp) = contents of reg. or reg. pair.
- ((rp)) = contents of memory location whose address is held in reg. pair.
- M = memory location whose address is held in reg. pair HL.
- dd = 2 digit hex. data.
- addr = 4 digit hex. address or data.
- ah = high order address byte.
- al = low order address byte.
- * = All flags affected.
- ** = All flags except CY affected.
- *** = Only CY flags affected.

For modification of flags, and execution times for each instruction, refer to section 5.5 of DCE Systems Designers Handbook.

DCE UPT COMMANDS

- Z1, Z2, etc. Zero and initialize.
- X, XA, etc. Examine and change registers.
- V, VG, etc. Examine and change vectors.
- D Display memory.
- S Substitute into memory.
- F Fill block of memory.
- M Move blocks of memory.
- G Initiate user program.
- L Initiate program trace.
- R Read paper tape.
- W Punch paper tape.
- E Punch end of file record.
- N Punch blank tape.
- P Program EPROM.
- T Transfer EPROM to memory.
- C Compare EPROM to memory.
- K Call UAE Editor.
- A Call UAE Assembler.

DCE UAE EDITOR

Editor accepts command sequence produced as paper tape off-line or typed in directly. Commands can be separated by CR, LF. Reader starts on entry, to input command sequence, stopping on receipt of (cD) char. Tape to be edited is then placed in reader; receipt of any char. will restart reader and initiate edit function.

Commands and control characters

I'text'(cC) Insert 'text' at current pointer posn.
F'text'(cC) Find 'text'(leaves pointer at end of char. string).
Dn Delete n characters
Kn Kill n lines
Cn Copy n lines
NOTE: 'text' = character string to be inserted or found.
(cC) = end of text indicator (ETX or control C)
(cD) = end of command seq. indicator (EOT or control D)
n = decimal no. in range 1 - 255

DCE UAE ASSEMBLER

On entry, assembler requests type of DCE microcomputer on which program is to be executed, then requests the pass no. required. PASS 1 stores all symbolic labels and operands in memory. PASS 2 provides listing of program, with error indications and PASS 3 produces the object program on paper tape in standard HEX format.

Required Source Format and Directives

1. All numeric addresses and operands should be in HEX notation and must start with a numeric char; ie. FFFF should be typed as OFFF.
2. All symbols should be alpha chars. only, 5 chars. max., and followed by a colon (:) when specified.
3. Arithmetic functions are limited to + and - with number restrictions as in 1.
4. Directives are as follows:
 - ORG Defines start of program segment.
 - EQU Allows definition of symbols.
 - DB Allows definition of single byte.
 - DW Allows definition of double byte.
 - END Defines end of program. (MUST BE INCLUDED)
5. Comments must be preceded by a semi-colon (;) and can occupy a whole line.

All fields, (label, operator, operand and comments) must be separated by a space or optionally by the (cI) control I char. This allows for formatting of listing printout.

Error indications

These appear on the left hand margin during the listing pass, and have the following meaning:-

- S Syntax error ; indicates illegal statement.
- U Undefined label; indicates no value assigned to that label.
- M Multiple defn. ; indicates same symbol has been used twice, (also indicates use of opcode or reg. name as label)..

As the DCE assembler ignores delete codes (RUBOUT), corrections to the source tape can be carried out at the time of preparation by back spacing the tape, overpunching the incorrect character or characters with the delete code, and then punching the correct sequence required.

Full details of UPT Utility and UAE assembler/editor commands with examples can be found in DCE-DM User's Manual.

DATA APPLICATIONS INTERNATIONAL S.A.

Dreve des Renards 6

Boite aux Lettres 8

1180 BRUSSELS

BELGIUM

Tel: 375 11 14 Tlx: 61 067 DAIBRU

DATA APPLICATIONS (UK) LTD.

16b, Dyer Street

CIRENCESTER

Gloucestershire

ENGLAND

Tel: (0285) 61828

Tlx: 43605 becham

MICRONIC-DAI GMBH

Balanstrasse 138/1

8000 MUNCHEN 90

GERMANY

Tel: 686939

REPRESENTATIVES

BELGIUM

RODELCO SA/NV

Avenue Herbert Hoover 32

1200 BRUSSELS

Tel: (02) 734 67 46

Tlx: 61415 rodl b

SWITZERLAND

ELBATEX AG

Alberich-Zwyszigstrasse 28

5430 WEITINGEN

Tel: 265 641

Tlx: 55239 elba ch

INDIA

APPLIED ELECTRONICS LTD

Aplab House

A-5 Wagle Industrial Estate

THANA-400 604 MAHARASTRA

Tel: 59 18 61

Tlx: (081) 114479

SPAIN & PORTUGAL

MICROELECTRONICA Y CONTROL S.A.

Travesera de Gracia 15 2ª 4ª A

BARCELONA 6

Tel: 218 4497

Tlx: 59504 jumb e

GERMANY

Dipl. Ing. ERNST FEY Nachf

Horemansstrasse 28

8000 MUNCHEN 19

Tel: (089) 184 041

Tlx: 529259

SWEDEN

KOMPONENTBOLAGET NAXAB

Rox 4115

17104 SOLNA

Tel: 985 140

Tlx: 17912 kompab s

ITALY

EMESA S.P.A.

Via Ludovico da Viadana 9

20122 MILANO

Tel: 02 8690616

Tlx: 33 196 alifin

DENMARK

ANALOG DEVICES APS

Marielundvej 46D

2730 HERLEV

Tel: (02) 84 58 00

Tlx: 35280 analog dk

FINLAND

HAVULINNA OY

P.O. Box 68

SF-00811 HELSINKI 81

Tel: 755 4144

Tlx: 12426

FRANCE

TECHMATION

Le Doge

18/24 Quai de la Marne

75019 PARIS

Tel: 81 200 11 05

Tlx: 211541

HOLLAND

INELCO NEDERLAND BV

Joan Mayskenweg 22

AMSTERDAM 1006

Tel: (020) 93 48 24

Tlx: 14622

AUSTRIA

OTHMAR LACKNER

Singerstrasse 4/13

Postfach 253

1010 WIEN 1

Tel: 0222 529980

Tlx: 12886 laknr

GREECE

DIGICO LTD

ILLisou 28

ATHENS T.T. 403

Tel: 92 28 442/3

Tlx: 215877

NORWAY

SCANCOPTER A.S.

Josefinesgt. 26

OSLO 3

Tel: 02 694490

Tlx: 19112 scaco