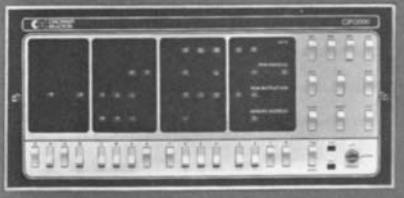
CIP/2200

pocket guide

NOVEMBER, 1971



CINCINNATI MILACRON

The following terms are used in this pocket guide:

NAME	Name field entry, or operand field symbol
SYMB	Conditional assembly symbol
ADDR	Operand expression used to calculate an address
I	Operand expression used to specify data which may be negative; must be absolute
N	Operand expression used to specify data which may not be negative, as well as counts, masks, etc.; must be absolute
L	Operand expression used to specify explicit length value; must be absolute
DATA	Data storage initialization specification.

INSTRUCTION SET LISTED NUMERICALLY BY OPCODE

Opcode	Instruction	Mneomic	Operands	Indicators
00	Halt	HLT		
01	Trap	TRP		
02	Enter Sense Switches	ESW		
03	Interchange A&B	IAB		
04	Disable Interrupt System	DIN		INT
05	Enable Interrupt System	EIN		INT
06	Disable Interval Timer	DIT		
07	Enable Interval Timer	EIT		
08	Reset OV, Set WL=1	RO1		OV,WL
09	Reset OV, Set WL=2	RO2		OV,WL
0A	Reset OV, Set WL=3	RO3		OV,WL
0B	Reset OV, Set WL=4	RO4		OV,WL
0C	Set OV, Set WL=1	SO1		OV,WL
0D	Set OV, Set WL=2	SO2		OV,WL
0E	Set OV, Set WL=3	SO3		OV,WL
0F	Set OV, Set WL=4	SO4		OV,WL
10	Skip if Overflow Set	SOV	ADDR	OV
11	Skip if A=0	SAZ	ADDR	
12	Skip if B=0	SBZ	ADDR	
13	Skip if X=0	SXZ	ADDR	
14	Skip if A Negative	SAN	ADDR	
15	Skip if X Negative	SXN	ADDR	
16	Skip if A=B	SAB	ADDR	
17	Skip if A=X	SAX	ADDR	

Opcode	Instruction	Mnemonic	Operands	Indicators
18	Skip if Overflow Not Set	NOV	ADDR	OV
19	Skip if A≠0	NAZ	ADDR	
1A	Skip if B≠0	NBZ	ADDR	
1B	Skip if X≠0	NXZ	ADDR	
1C	Skip if A Not Negative	NAN	ADDR	
1D	Skip if X Not Negative	NXN	ADDR	
1E	Skip if A≠B	NAB	ADDR	
1F	Skip if A≠X	NAX	ADDR	
20	Rotate Left A	RLA	N	
21	Rotate Left B	RLB	N	
22	Rotate Left Long	RLL	N	
23	Decrement A	DCA		OV
24	Logical Right A	LRA	N	
25	Logical Right B	LRB	N	
26	Logical Right Long	LRL	N	
27	Decrement B	DCB		OV
28	Arithmetic Left A	ALA	N	
29	ArithmeticLeft B	ALB	N	
2A	Arithmetic Left Long	ALL	N	
2B	Transfer A to B	TAB		
2C	Arithmetic Right A	ARA	N	
2D	Arithmetic Right B	ARB	N	
2E	Arithmetic Right Long	ARL	N	
2F	Transfer B to A	TBA		
30	Input Byte Serially	IBS		
31	Input Byte to A	IBA	DO,DEV	
32	Input Byte to B	IBB	DO,DEV	
33	Input Byte to Memory	IBM	DO,DEV, ADDR(X)	
34	No Operation	NOP		
35	Interchange A&X	IAX		
36	Interchange B&X	IBX		
38	Output Byte Serially	OBS		
39	Output Byte From A	OBA	DO,DEV	
3A	Output Byte From B	OBB	DO,DEV	
3B	Output Byte From Memory	OBM	DO,DEV, ADDR(X)	
40	OR B to A	ORA		
41	Exclusive OR B to A	XRA		
42	OR A to B	ORB		
43	Exclusive OR A to B	XRB		
44	Increment Index	INX		OV

Opcode	Instruction	Mnemonic	Operands	Indicators
45	Decrement Index	DCS		OV
46	Add Word Length to X	AWX		OV
47	Subtract Word Length from X	SWX		OV
48	Increment A	INA		OV
49	Increment B	INB		OV
4A	Ones Complement A	OCA		
4B	Ones Complement B	OCB		
4C	Transfer A to X	TAX		
4D	Transfer B to X	DCX		
4E	Transfer X to A	TXA		
4F	Transfer X to B	TXB		
50	Add to Word Immediate	AWI	I,ADDR(X)	OV,ALI
51	Branch On Condition	BOC	N,ADDR(X)	
52	Move Immediate	MVI	N,ADDR(X)	
53	Compare Logical Immediate	CLI	N,ADDR(X)	ALI
54	Test Under Mask Immediate	TMI	N,ADDR(X)	ALI
55	Set Bits Under Mask Immediate	SMI	N,ADDR(X)	ALI
56	Clear Bits Under Mask Immediate	CMI	N,ADDR(X)	ALI
57	Invert Bits Under Mask Immediate	IMI	N,ADDR(X)	ALI
58	Add Decimal	ADD	$ADDR_T$(L$_T$,X), $ADDR_S$(L$_S$,X)	OV,ALI
59	Subtract Decimal	SBD	$ADDR_T$(L$_T$,X), $ADDR_S$(L$_S$,X)	OV,ALI
5A	Multiply Step Decimal	MSD	$ADDR_T$(L$_T$,X), $ADDR_S$(L$_S$,X)	OV
5B	Divide Step Decimal	DSD	$ADDR_T$(L$_T$,X), $ADDR_S$(L$_S$,X)	OV
5C	Move Character String Left	MVL	$ADDR_T$(L,X) $ADDR_S$(X)	
5D	Move Character String Right	MVR	$ADDR_T$(L,X) $ADDR_S$(X)	
5E	Rom Exit	XIT		
5F00	Save Machine State	SAV		
5F01	Return	RET		all
5F02	Add to Index Immediate	AXI	I	OV
5F03	Return Displaced	RTN	I	all
5F04	Edit and Mark	EDT	$ADDR_T$(L,X), $ADDR_S$(X)	ALI
5F05	Compare Logical Character	CLC	$ADDR_T$(L,X) $ADDR_S$(X)	ALI
5F06	Translate Under Mask	TRM	N,$ADDR_T$(L,X) $ADDR_S$(X)	

Opcode	Instruction	Mnemonic	Operands	Indicators
5F07	Translate and Test Under Mask	TTM	N,ADDR$_T$(L,X), ADDR$_S$(X)	ALI
5F08-5FFF	Secondary Rom Exit	XT2	N	
60-67*	Jump	JMP	ADDR(X)	
68-6F*	Return Jump	RTJ	ADDR(X)	
70-77*	Increment Word in Memory	IWM	ADDR(X)	OV,ALI
78-7F*	Decrement Word in Memory	DWM	ADDR(X)	OV,ALI
80-87*	Load X	LDX	ADDR(X)	
88-8F*	Store X	STX	ADDR(X)	
90-97*	Multiply Step	MST	ADDR(X)	
98-9F*	Divide Step	DST	ADDR(X)	
A0-A7*	Add to A	ADA	ADDR(X)	OV
A8-AF*	Add Variable	ADV	ADDR(X)	OV
B0-B7*	Subtract From A	SBA	ADDR(X)	OV
B8-BF*	Subtract Variable	SBV	ADDR(X)	OV
C0-C7*	Load B	LDB	ADDR(X)	
C8-CF*	Store B	STB	ADDR(X)	
D0-D7*	AND Memory To A	ANA	ADDR(X)	
D8-DF*	AND Variable	ANV	ADDR(X)	
E0-E7*	Load A	LDA	ADDR(X)	
E8-EF*	Load Variable	LDV	ADDR(X)	
F0-F7*	Store A	STA	ADDR(X)	
F8-FF*	Store Variable	STV	ADDR(X)	

INSTRUCTION SET LISTED ALPHABETICALLY BY INSTRUCTION

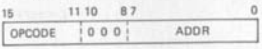
Instruction	Mnemonic	Operands	Indicators	Opcode
Add Decimal	ADD	$ADDR_T(L_T,X)$, $ADDR_S(L_S,X)$	OV,ALI	58
Add to A	ADA	ADDR(X)	OV	A0-A7*
Add to Index Immediate	AXI	I	OV	5F02
Add to Word Immediate	AWI	I,ADDR(X)	OV,ALI	50
Add Variable	ADV	ADDR(X)	OV	A8-AF*
Add Word Length to X	AWX		OV	46
AND Memory To A	ANA	ADDR(X)		D0-D7*
AND Variable	ANV	ADDR(X)		D8-DF*
Arithmetic Left A	ALA	N		28
Arithmetic Left B	ALB	N		29
Arithmetic Left Long	ALL	N		2A
Arithmetic Right A	ARA	N		2C
Arithmetic Right B	ARB	N		2D
Arithmetic Right Long	ARL	N		2E
Branch On Condition	BOC	N,ADDR(X)		51
Clear Bits Under Mask Immediate	CMI	N,ADDR(X)	ALI	56
Compare Logical Character	CLC	$ADDR_T(L,X)$ $ADDR_S(X)$	ALI	5F05
Compare Logical Immediate	CLI	N,ADDR(X)	ALI	53
Decrement A	DCA		OV	23
Decrement B	DCB		OV	27
Decrement Index	DCX		OV	45
Decrement Word in Memory	DWM	ADDR(X)	OV,ALI	78-7F*
Disable Interrupt System	DIN		INT	04
Disable Interval Timer	DIT			06
Divide Step Decimal	DSD	$ADDR_T(L_T,X)$, $ADDR_S(L_S,X)$	OV	5B
Divide Step	DST	ADDR(X)		98-9F*
Edit and Mark	EDT	$ADDR_T(L,X)$, $ADDR_S(X)$	ALI	5F04
Enable Interrupt System	EIN		INT	05
Enable Interval Timer	EIT			07
Enter Sense Switches	ESW			02
Exclusive OR A to B	XRB			43
Exclusive OR B to A	XRA			41
Halt	HLT			00
Increment A	INA		OV	48
Increment B	INB		OV	49
Increment Index	INX		OV	44

Instruction	Mnemonic	Operands	Indicators	Opcode
Increment Word in Memory	IWM	ADDR(X)	OV,ALI	70-77*
Input Byte to A	IBA	DO,DEV		31
Input Byte to B	IBB	DO,DEV		32
Input Byte to Memory	IBM	DO,DEV, ADDR(X)		33
Input Byte Serially	IBS			30
Interchange A&B	IAB			03
Interchange A&X	IAX			35
Interchange B&X	IBX			36
Invert Bits Under Mask Immediate	IMI	N,ADDR(X)	ALI	57
Jump	JMP	ADDR(X)		60-67
Load A	LDA	ADDR(X)		E0-E7*
Load B	LDB	ADDR(X)		C0-C7*
Load Variable	LDV	ADDR(X)		E8-EF*
Load X	LDX	ADDR(X)		80-87*
Logical Right A	LRA	N		24
Logical Right B	LRB	N		25
Logical Right Long	LRL	N		26
Move Character String Left	MVL	ADDR$_T$(L,X) ADDR$_S$(X)		5C
Move Character String Right	MVR	ADDR$_T$(L,X) ADDR$_S$(X)		5D
Move Immediate	MVI	N,ADDR(X)		52
Multiply Step	MST	ADDR(X)		90-97*
Multiply Step Decimal	MSD	ADDR$_T$(L$_T$,X), ADDR$_S$(L$_S$,X)		5A
No Operation	NOP			34
Ones Complement A	OCA			4A
Ones Complement B	OCB			4B
OR B to A	ORA			40
OR A to B	ORB			42
Output Byte From A	OBA	DO,DEV		39
Output Byte From B	OBB	DO,DEV		3A
Output Byte From Memory	OBM	DO,DEV, ADDR(X)		38
Output Byte Serially	OBS			38
Reset OV, Set WL=1	RO1		OV,WL	08
Reset OV, Set WL=2	RO2		OV,WL	09
Reset OV, Set WL=3	RO3		OV,WL	0A
Reset OV, Set WL=4	RO4		OV,WL	0B
Return	RET		all	5F01
Return Jump	RTJ	ADDR(X)		68-6F*
Return Displaced	RTN	I	all	5F03
Rom Exit	XIT			5E
Rotate Left A	RLA	N		20
Rotate Left B	RLB	N		21
Rotate Left Long	RLL	N		22

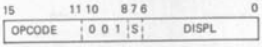
Instruction	Mnemonic	Operands	Indicators	Opcode
Save Machine State	SAV			5F00
Secondary Rom Exit	XT2	N		5F08-5FFF
Set Bits Under Mask Immediate	SMI	N,ADDR(X)	ALI	55
Set OV, Set WL=1	SO1		OV,WL	0C
Set OV, Set WL=2	SO2		OV,WL	0D
Set OV, Set WL=3	SO3		OV,WL	0E
Set OV, Set WL=4	SO4		OV,WL	0F
Skip if A=B	SAB	ADDR		16
Skip if A≠B	NAB	ADDR		1E
Skip if A Negative	SAN	ADDR		14
Skip if A Not Negative	NAN	ADDR		1C
Skip if A=0	SAZ	ADDR		11
Skip if A≠0	NAZ	ADDR		19
Skip if A=X	SAX	ADDR		17
Skip if A≠X	NAX	ADDR		1F
Skip if B=0	SBZ	ADDR		12
Skip if B≠0	NBZ	ADDR		1A
Skip if Overflow Set	SOV	ADDR	OV	10
Skip if Overflow Not Set	NOV	ADDR	OV	18
Skip if X=0	SXZ	ADDR		13
Skip if X≠0	NXZ	ADDR		1B
Skip if X Negative	SXN	ADDR		15
Skip if X not Negative	NXN	ADDR		1D
Store A	STA	ADDR(X)		F0-F7*
Store B	STB	ADDR(X)		C8-CF*
Store Variable	STV	ADDR(X)		F8-FF*
Store X	STX	ADDR(X)		88-8F*
Subtract Decimal	SBD	$ADDR_T(L_T,X)$, $ADDR_S(L_S,X)$	OV,ALI	59
Subtract From A	SBA	ADDR(X)	OV	B0-B7*
Subtract Variable	SBV	ADDR(X)	OV	B8-BF*
Subtract Word Length From X	SWX		OV	47
Test Under Mask Immediate	TMI	N,ADDR(X)	ALI	54
Transfer A to B	TAB			2B
Transfer A to X	TAX			4C
Transfer B to A	TBA			2F
Transfer B to X	TBX			4D
Transfer X to A	TXA			4E
Transfer X to B	TXB			4F
Translate Under Mask	TRM	N,$ADDR_T(L,X)$ $ADDR_S(X)$		5F06
Translate and Test Under Mask	TTM	N,$ADDR_T(L,X)$, $ADDR_S(X)$	ALI	5F07
Trap	TRP			01

ADDRESSING MODES

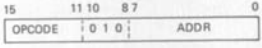
Direct page 0 (m=0) EA = ADDR



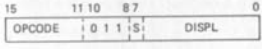
Direct relative (m=1) EA = (P) + DISPL



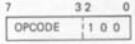
Indirect page 0 (m=2) EA = (ADDR)



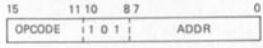
Indirect relative (m=3) EA = ((P) + DISPL)



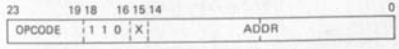
Base addressing (m=4) EA = (X)



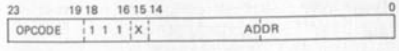
Base plus displacement addressing (m=5) EA = (X) + ADDR



Extended addressing (m=6) EA = ADDR or ADDR + (X)



Extended indirect addressing (m=7, jump and return jump only)



Literal addressing (m=7) EA = 2nd byte of instruction

Addressing Modes		(Assembly Language Coding)
OPC	ADDR	modes 0, 2 (direct page zero and direct relative)
OPC*	ADDR	modes 1, 3 (indirect page zero and indirect relative)
OPC-		mode 4 (base addressing)
OPC+	N	mode 5 (base plus displacement)
OPC/	ADDR(X)	mode 6 (extended addressing)
OPC=	DATA	mode 7 (literal addressing)

CONTROL CHARACTERS FOR EDIT AND MARK INSTRUCTION

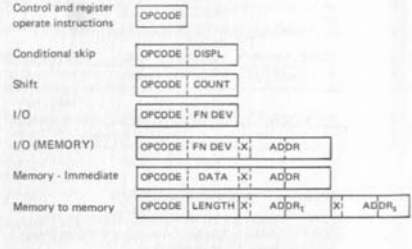
Digit select (ds) = X' 20'
Significance start (SS) = X' 21'
Field start (fs) = X' 22'

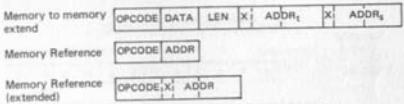
ASSEMBLER INSTRUCTIONS

	Mnemonic	Operands	Name
	ABS		set absolute mode
	CMN	N ($0 \leq N \leq 256$)	define common storage
%	CON		continue (name required)
	DC	DATA	define constant
	DS	N ($0 \leq N \leq 256$)	define storage
	EJECT		eject page
	END	ADDR	end
	ENT	$NAME_1, \ldots, NAME_n$	define entry symbol
	EQU	ADDR	equate (name required)
	EXT	$NAME_1, \ldots, NAME_n$	define external symbol
%	GTO	SYMB	go to
	IDENT	NAME	program identification
%	IF	I,SYMB	if
%	IFN	I,SYMB	if not
	ORG	ADDR	origin
	PRINT	I	print control
	REL		set relocatable mode
	SET	ADDR	set (name required)
	SPACE	N	space listing

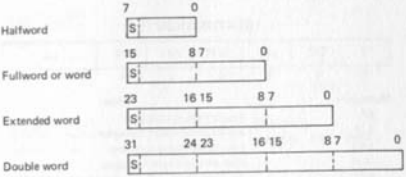
INSTRUCTION AND DATA FORMATS

Instruction Formats

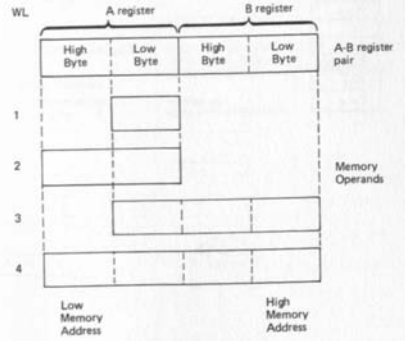




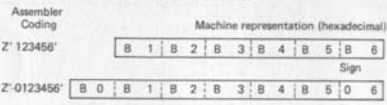
Variable Length Data Formats



Data Format	Assembly Coding	Size	Range
Halfword	H'n'	1 byte	$-128 \leq n \leq 255$
Fullword or word	F'n'	2 bytes	$-32768 \leq n \leq 65535$
Extended word	F'n'	3 bytes	$-8,388,608 \leq n \leq 16,777,215$
Double	F'n'	4 bytes	$-2,147,483,648 \leq n \leq 4,294,967,295$



Decimal Data Format

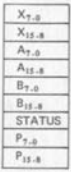


STATUS REGISTER

PF	E/Z	L/O	H/M	INT	OV	WL	
7	6	5	4	3	2	1	0

Mnemonic	Name
PF	Power Failure Indicator
E/Z	Equal/Zeros Result Indicator
L/O	Low/Ones Result Indicator
H/M	High/Mixed Result Indicator
INT	Interrupt System Disabled Indicator
OV	Overflow Indicator
WL	Word Length Indicator WL values of 00_2-11_2 correspond to word lengths of 1-4.

CONTROL STACK ENTRY



SUBROUTINE STRUCTURES

```
Calling Sequence                    Subroutine
      .                                 SUBR  DC    **
      .                                       .
      .                                       .
RTJ/  SUBR                                    .
                                        JMP*  SUBR
```

```
Calling Sequence                    Subroutine
      .                                 SUBR  DC    **
      .                                       SAV
RTJ/  SUBR                                    .
                                              .
                                              .
                                              .
                                              .
                                        RET
```

```
Calling Sequence                    Subroutine
      .                                       .  OPC  OPND
      .                                       .
      .                                       .
JMP   *+4                                     .
DC    A'*+7'  RETURN ADDRESS                  .
SAV                                           .
JMP/  SUBR                              RET
```

INTERRUPT SERVICE ROUTINE STRUCTURE

```
INTSUB DC  **          INTERRUPT SUBROUTINE
*                      RETURN ADDRESS
       SAV             SAV MACHINE STATE
         .
         .             SUBROUTINE BODY
         .
       OBA  ARM,DEV    ARM DEVICE CONTROLLER
*                      FOR NEXT INTERRUPT
       RET             RETURN TO INTERRUPTED PROGRAM
*
*
ARM    EQU  2          ARM DEVICE ORDER
```

DEDICATED MEMORY LOCATIONS

DMC Address Assignments:

000-001	Device 0 Current Address
002-003	Device 0 Ending Address
004-005	Device 1 Current Address
006-007	Device 1 Ending Address
–	– – –
07C-07D	Device 31 Current Address
07E-07F	Device 31 Ending Address

DMA Address Assignments:

58	DMA1 Status
5C	DMA2 Status
60-61	DMA1 Buffer 1 Start Address
62-63	DMA1 Buffer 1 End Address
64-65	DMA1 Buffer 2 Start Address
–	– – –
6E-6F	DMA1 Buffer 4 End Address
70-71	DMA2 Buffer 1 Start Address
–	– – –
7E-7F	DMA2 Buffer 4 End Address

Interrupt Transfer Locations & Reserved System Areas:

080-081	Console Interrupt
082-083	DMA Interrupt
084-085	Interval Timer Counter
086-087	Interval Timer Interrupt
088-089	Reserved For Future Use
08A-08B	Memory Parity Interrupt
08C-08D	Stack Overflow/Underflow Interrupt
08E-08F	Power Fail Interrupt
090-091	Power Restart Interrupt
092-093	Control Stack Pointer
094-0FF	Free

External Interrupts:

100-101	External Interrupt 0
102-103	External Interrupt 1
–	– – –
17E-17F	External Interrupt 63

System Save Area:

180-18F

STANDARD CHARACTER CODES

SYMBOL	ASCII (HEX)	EBCDIC (HEX)	EBCDIC (PUNCH CODE)	SYMBOL	ASCII (HEX)	EBCDIC (HEX)	EBCDIC (PUNCH CODE)
blank	A0	40	blank	@	C0	7C	8-4
!	A1	5A	11-8-2	A	C1	C1	12-1
"	A2	7F	8-7	B	C2	C2	12-2
#	A3	7B	8-3	C	C3	C3	12-3
$	A4	5B	11-8-3	D	C4	C4	12-4
%	A5	6C	0-8-4	E	C5	C5	12-5
&	A6	50	12	F	C6	C6	12-6
'	A7	7D	8-5	G	C7	C7	12-7
(	A8	4D	12-8-5	H	C8	C8	12-8
)	A9	5D	11-8-5	I	C9	C9	12-9
*	AA	5C	11-8-4	J	CA	D1	11-1
+	AB	4E	12-8-6	K	CB	D2	11-2
,	AC	6B	0-8-3	L	CC	D3	11-3
-	AD	60	11	M	CD	D4	11-4
.	AE	4B	12-8-3	N	CE	D5	11-5
/	AF	61	0-1	O	CF	D6	11-6
0	B0	F0	0	P	D0	D7	11-7
1	B1	F1	1	Q	D1	D8	11-8
2	B2	F2	2	R	D2	D9	11-9
3	B3	F3	3	S	D3	E2	0-2
4	B4	F4	4	T	D4	E3	0-3
5	B5	F5	5	U	D5	E4	0-4
6	B6	F6	6	V	D6	E5	0-5
7	B7	F7	7	W	D7	E6	0-6
8	B8	F8	8	X	D8	E7	0-7
9	B9	F9	9	Y	D9	E8	0-8
:	BA	7A	8-2	Z	DA	E9	0-9
;	BB	5E	11-8-6	[	DB	4F	12-8-7
<	BC	4C	12-8-4	\	DC	4A	12-8-2
=	BD	7E	8-6	]	DD	5F	11-8-7
>	BE	6E	0-8-6	↑	DE	6D	0-8-5
?	BF	6F	0-8-7	←	DF	6A	0-8-2

TELETYPE CONTROL AND TRANSMISSION CODES

Function	ASCII
NULL	80
SOH	81
STX	82
ETX	83
EOT	84
ENQ	85
ACK	86
BELL	87
BS	88
H TAB	89
LINE FEED	8A
V TAB	8B
FORM	8C
CARRIAGE RETURN	8D
SO	8E
SI	8F
DLE	90

Function	ASCII
DC1 (Reader on)	91
DC2 (Punch on)	92
DC3 (Reader off)	93
DC4 (Punch off)	94
NAK	95
SYNC	96
ETB	97
CAN	98
EM	99
SUB	9A
ESC	9B
FS	9C
GS	9D
RS	9E
US	9F
DEL (Rubout)	FF

STANDARD I/O STATUS BIT ASSIGNMENTS

Bit Number	Condition	Description
0	Ready	0 = Device unavailable 1 = Device available
1	Input flag	0 = Byte not ready 1 = Byte ready
2	Output flag	0 = Device not ready to accept data 1 = Device ready for data
3	Error	0 = No error occurred 1 = Last byte was in error
4-7	Device dependent	

STANDARD I/O DEVICE ORDERS

Device Order	Operation	Description
0	Data	Causes data byte to be transferred.
1	Status/Function	Causes function or status byte to be transferred.
2	Arm	Sets external interrupt sequencer into an armed state.
3	Disarm	Sets external interrupt sequencer into a disarmed state.
4	Disconnect	Causes DMC input or output operation to be stopped.
5	Block Input	Starts DMC concurrent block input into memroy.
6	Block Output	Starts DMC concurrent block output from memory.
7	Device Dependent Order	Device dependent.

CIP/2200 STANDARD DEVICE ADDRESS ASSIGNMENTS

00	Parallel Teletype
01	Modem or Teletype
02	Tape Reader
03	Tape Punch
04	Card Reader
05	Line Printer
06	Drum/Disc
08	4 Byte Multiplexer
09 – 0C	Cassettes
11	Modem
16	DMA1
17	DMA2
18 – 1F	Priority Interrupt Group

STANDARD I/O DEVICE ORDERS

DEVICE ORDER		2nd BYTE OF I/O INSTRUCTION (BINARY)
0	Data	000ddddd
1	Status/Function	001ddddd
2	Arm	010ddddd
3	Disarm	011ddddd
4	Disconnect	100ddddd
5	Block Start Input	101ddddd
6	Block Start Output	110ddddd
7	Device Dependent	111ddddd

ddddd = 5 bit device address

DMA DEVICE ORDERS

DEVICE ORDER		2nd BYTE OF I/O INSTRUCTION
0	Input Status	000ddddd
1	Start Channel Cyclic Mode	001ddddd
2	Stop Channel at Block End	010ddddd
3	Start Channel and Stop at End of Block	011ddddd
4	Disconnect Channel Immediately	100ddddd

ddddd = 10110 for DMA1
ddddd = 10111 for DMA2

I/O DEVICE ORDER CODES

(Machine and source language equivalents)

Because the order code and device address share the third hex character of an instruction, the source and machine codes differ. Machine is twice source for device 0-F and twice +1 for device 10-1F.

INPUT/OUTPUT STATUS BITS

PARALLEL TELETYPE

(Device address: 00)

Bit	Value	Condition
7		Not Used
6		Not Used
5		Not Used
4		Not Used
3	0	No Error Detected
	1	Lost Data Detected
2	0	Write Buffer Not Ready
	1	Write Buffer Ready for Transfer from Computer
1	0	Read Buffer Not Ready
	1	Read Buffer Ready for Transfer to Computer
0	0	Concurrent Mode
	1	Not in Concurrent Mode

300 LPS PAPER TAPE READER

(Device address: 02)

Bit	Value	Condition
7		Not Used
6		Not Used
5		Not Used
4		Not Used
3	0	No Error
	1	Data Input Transfer Error
2		Not Used
1	0	Data Not Ready for Input Transfer
	1	Data Is Ready for Input Transfer
0	0	Device Not in Concurrent Mode
	1	Device In Concurrent Mode

DRPE PAPER TAPE PUNCH

(Device address: 03)

Bit	Value	Condition
7		Not Used
6	0	Tape Supply Greater Than 250 Feet
	1	Tape Supply Low – Less Than 250 Feet
5	0	Punch Not in Concurrent Mode
	1	Punch Operating in Concurrent Mode
4	0	Tape Supply OK
	1	Tape Supply Exhausted
3	0	No Device Errors
	1	Punch or Feed Error Detected
2	0	Punch Busy – Not Ready
	1	Punch Ready – Not Busy
1		Not Used
0	0	Test Switch Set for Punching
	1	Test Switch Not in Off Position

400 CPM CARD READER

(Device address: 04)

Bit	Value	Condition
7		Not Used
6		Not Used
5		Not Used
4	0	No Electro – Mechanical Failure
	1	Electro – Mechanical Failure Detected
3	0	No Ebcdic Error
	1	Ebcdic Error Detected
2	0	Hopper and Stacker at Satisfactory Levels
	1	Hopper Empty or Stacker Full
1	0	Character Not Ready for Input Transfer
	1	Character Is Ready for Input Transfer
0	0	Card Reader Busy – Not Ready for Operation
	1	Card Reader Ready for Operation

300 LPM LINE PRINTER

(Device address: 05)

Bit	Value	Condition
7	0	Printer Not at Top of Form
	1	Printer Is at Top of Form
6		Not Used
5	0	Device Not in Concurrent Mode
	1	Device In Concurrent Mode
4		Not Used
3	0	No Device Error
	1	Device Error Detected – Print, Form Check, Alarm
2	0	Operation Not Complete – Not Ready to Receive
	1	Operation Is Complete – Ready to Receive Another Character
1		Not Used
0	0	Device Not Ready
	1	Device Ready for Operation

CASSETTE TAPE TRANSPORT

(Device address: 09, 0A, 0B, 0C)

Bit	Value	Condition
7	1	Track protect violated
	0	Not violated
6	1	No change of tape address
	0	Tape address change no change
5	1	Transport stalled
	0	Transport not stalled
4	1	Beginning or end of tape
	0	Not Beginning or end of tape
3	1	Character in error
	0	Character not in error
2	1	Concurrent output motion
	0	Not concurrent output no motion
1	1	Concurrent mode input
	0	No concurrent mode input
0	1	Data ready } "Service" on prints
	0	Data not ready }

Process Controls Division | Cincinnati Milacron Company
Lebanon, Ohio 45036 | Tel. (513) 494-1200

Machine Tools
Process Controls
Chemicals
Plastics
Plastics Processing Machinery
Abrasives

CTL-364