CCS 200, 300 & 400
Z-80 Family

PROGRAMMING GUIDE

California Computer Systems

CCS SYSTEMS 200, 300, and 400

Z-80™ FAMILY

PROGRAMMING REFERENCE MANUAL

MANUAL 89000-23400 REV. A

CALIFORNIA COMPUTER SYSTEMS
250 CARIBBEAN DRIVE
SUNNYVALE CA 94086

The information contained in this manual is believed to be correct at the time of publication. However, CCS assumes no liability resulting from the use of this publication.

Publication history:

Revision A printed March 1981

INTRODUCTION

This manual is divided into sections corresponding to the six Z-80 devices used in the CCS 200, 300, and 400 systems: the CPU, the SIO, the DART, the PIO, the CTC, and the DMA. Each section provides a complete discussion of how to program the device in question, though the implementation of the devices in CCS systems may often limit the programming possibilities. Such limitations are noted in the Programming Information chapters of the individual board manuals.

In order to minimize the number of ports occupied by a device, the Z-80 peripherals have been designed so that a number of registers may occupy a single address. In some cases part of the base register for a certain address is used to point to the register to be accessed on the next read or write. In other cases certain bits determine the register being accessed by the current operation. The result is that programming of the Z-80 family can become somewhat complicated. It is the aim of this manual to simplify the procedure as much as possible. This manual is not a Z-80 programming tutorial, however. It is designed to be used as a reference by programmers experienced with similar kinds of programming, if not with programming the Z-80 itself. For those who desire a tutorial approach, a number of good books are available.

TABLE OF CONTENTS

CHAPTER 1

PROGRAMMING THE CPU

The Z-8Ø differs significantly from other microprocessors in its register configuration, instruction set, and interrupt response. Advantages of the Z-8Ø include a duplicate set of accumulator, flag, and general purpose registers, a large instructiion set including the 78 8Ø8Ø instructions, and a powerful daisy-chain interrupt mode supported by the Z-8Ø family peripherals. The essential information for CPU programming is provided in this section; for supplementary information see any of the numerous readily-available publications dealing with Z-8Ø programming.

1.1 THE CPU REGISTERS

The Z-8Ø CPU has four 16-bit and eighteen 8-bit program-accessible registers, as indicated in Figure 1.1. Descriptions of the registers follow.

1.1.1 Accumulator (A) and Flag (F) Registers

The two 8-bit accumulators A and A' hold the result of arithmetic and logical operations while their associated flag registers indicate the special results of such operations. A single exchange instruction allows the programmer to work with either pair of registers.

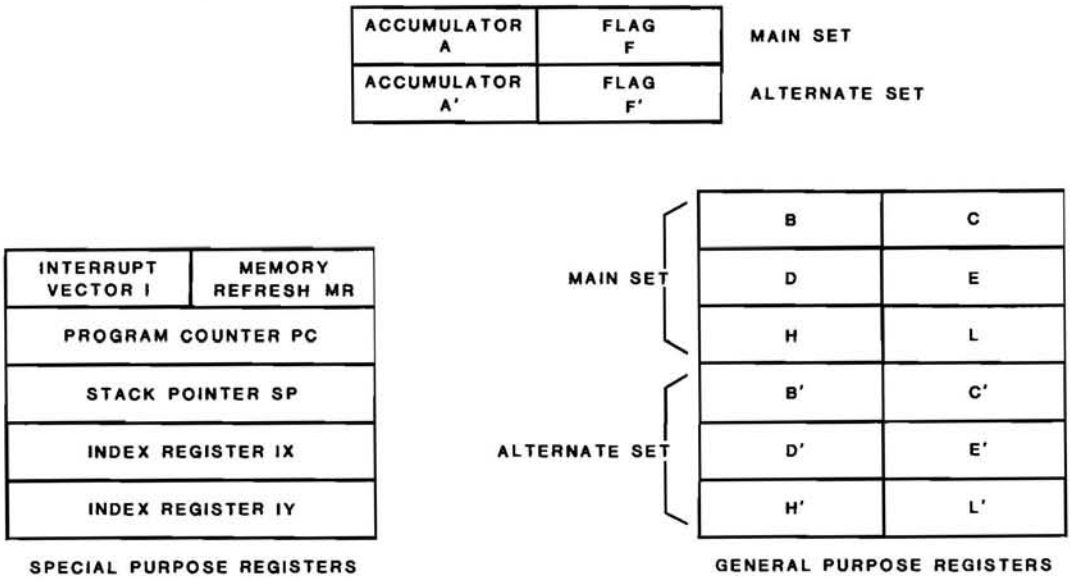


FIGURE 1.1. Z-80 REGISTERS

1.1.2 Special Purpose Registers

Program Counter (PC)--This 16-bit register holds the memory address of the current instruction. The PC is automatically incremented after its contents have been transferred to the address lines. A program jump overrides the incrementer and places a new value in the PC.

Stack Pointer (SP)--This 16-bit register holds the address of the current top of a stack located anywhere in external RAM memory. The PUSH and POP instructions push data from specific registers onto the stack or pop the data off the stack into specific registers.

Index Registers (IX and IY)--These two independent 16-bit registers hold a base address that is used in indexed addressing modes. This base address is used in conjunction with a displacement byte (a two's complement integer) in an indexed instruction to specify a location in memory.

Interrupt Page Address Register (I)--This register is used in interrupt response mode 2. The register stores the high order 8-bits of the indirect address; the interrupting device provides the lower 8-bits.

Memory Refresh Register (R)--This register is the counter register for dynamic memory refresh. It holds the refresh address placed on the address bus during the last two clock cycles of every M1 cycle. The address is automatically incremented. You would not normally access this register, but can load it for testing purposes.

1.1.3 General Purpose Registers

The general purpose registers consist of primary and alternate sets of six 8-bit registers. They can be used as individual 8-bit registers or as 16-bit register pairs. The primary pairs are BC, DE, and HL; the alternate pairs are BC', DE', and HL'. A single exchange command allows the programmer to switch from one set to the other.

1.2 CPU INTERRUPT MODES

The Z-80 CPU is capable of three modes of maskable interrupt response; the mode in which the CPU operates at a given time is determined by software. The three modes are as follows:

Mode 0--This mode is identical to the 8080A interrupt response mode. Data put onto the bus by the interrupting device is interpreted as an instruction by the CPU and immediately executed. The CPU will be in this mode after any system reset.

Mode 1--In this mode the CPU responds to an interrupt by executing a restart to location 0038h.

Mode 2--This powerful mode takes advantage of the Z-80 family interrupt daisy chain capabilities. If a Z-80 device has requested an interrupt (and

no higher-priority device has also requested an interrupt), it puts an 8-bit interrupt vector on the bus when the CPU acknowledges the interrupt. The CPU then reads the next instruction from the two contiguous locations specified by the I register (high byte) and the byte put on the data bus by the interrupting device (low byte; bit Ø = Ø). The I register is programmable, allowing the programmer to dedicate any page of memory for an interrupt service routine table. The interrupt vectors for the various devices are also programmable; in addition, some devices modify the interrupt vector according to the cause of the interrupt. In order to allow time for the interrupt daisy chain to implement its priority control, two wait states are inserted in all interrupt acknowledge cycles.

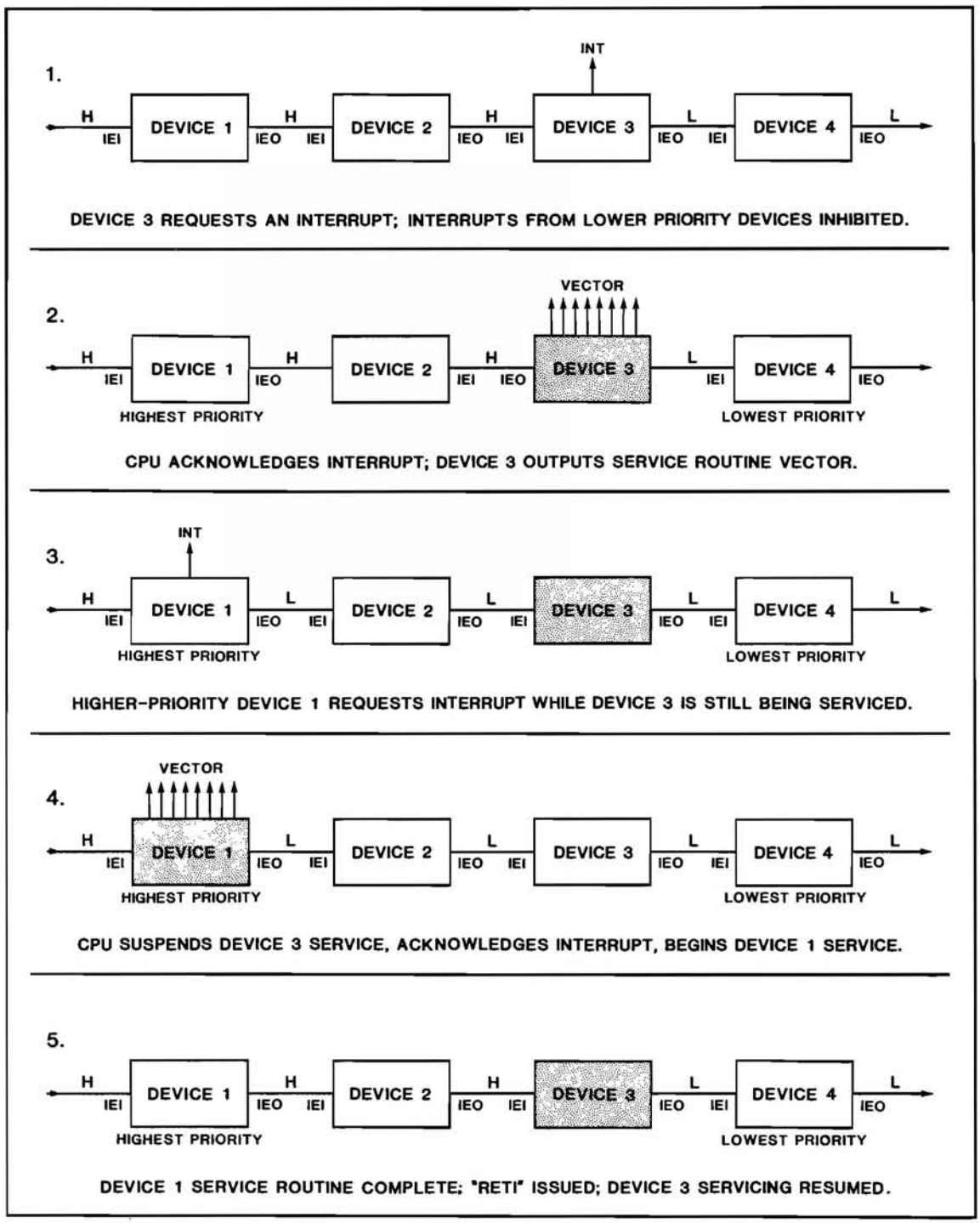
Figure 1.2 illustrates Mode 2 Interrupt operation.

The Z-8Ø responds to non-maskable interrupts by saving the current contents of the PC register on the stack, ignoring the data bus, and executing a call to location ØØ66h.

1.3 Z-8Ø INSTRUCTION SET

The Z-8Ø instruction set includes the 78 8Ø8Ø instructions plus 8Ø additional instructions. The 8Ø8Ø-compatible instructions use the same object codes, but different mnemonics. Programs written in 8Ø8Ø code can be duplicated in Z-8Ø code, but because of the 8Ø additional instructions Z-8Ø can often perform the same functions using fewer bytes than the 8Ø8Ø.

FIGURE 1.2. Z-8Ø MODE 2 INTERRUPT DAISY CHAIN OPERATION



1.3.1 ABBREVIATIONS USED IN THE INSTRUCTION SET TABLE

addr	A 16-bit memory address.
label	A 16-bit memory address represented by a mnemonic.
port	An 8-bit port address.
data	Single-byte data.
data 16	Double-byte data.
reg	One of the eight-bit registers A, B, C, D, E, H, L.
regd,regs	Destination register, source register.
pr	One of the register pairs AF, BC, DE, HL.
rp	One of the register pairs BC, DE, HL, SP.
rr	One of the register pairs specified in the operation description.
'	Denotes an alternate register or register pair: e.g., AF'.
A	The Accumulator.
F	The Program Status Register
R	The Refresh Register.
IV	The Interrupt Vector Register.
IX, IY	The 16-bit Index Registers.
PC	The Program Counter.
SP	The Stack Pointer.
B7, BØ, etc.	Individual bits of an eight-bit register.
()	The operand is the contents of the memory location addressed by the contents of the enclosed location.

Flags:

C	Carry status
Z	Zero status
S	Sign status
P/O	Parity/Overflow status
AC	Auxiliary Carry status
N	Subtract status

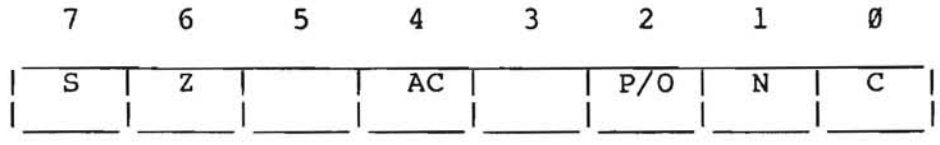
Symbols Used in Flag Columns:

(blank)	Flag not affected by operation
X	Flag affected by operation
1	Flag set by operation
Ø	Flag reset by operation
?	Flag unknown after operation
P	Flag shows parity status
O	Flag shows overflow status
I	Flag shows interrupt enable status

Conditions:

Z	Zero flag = 1
NZ	Zero flag = Ø
C	Carry flag = 1
NC	Carry flag = Ø
PO	Parity flag = Ø (Odd)
PE	Parity flag = 1 (Even)
P	Sign flag = Ø (Positive)
M	Sign flag = 1 (Minus)

The format of the Program Status Word is shown below. Blank bits are not used.



I/O INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
IN	A,port	2	DB yy							Input to A from I/O port addressed by A7-AØ.
IN	reg,(C)	2	ED		X	X	P	X	Ø	Input to register from port addressed by C.
INIR		2	ED B2		1	?	?	?	1	Block transfer from port addressed by C to memory addressed by incrementing HL. Stop when B, which decrements after each byte, reaches Ø.
INDR		2	ED BA		1	?	?	?	1	Block transfer from port addressed by C to memory addressed by decrementing HL. Stop when B, which decrements after each byte, reaches Ø.
INI		2	ED A2		X	?	?	?	1	Transfer byte from port addressed by C to memory addressed by HL. Increment HL, decrement B.
IND		2	ED AA		X	?	?	?	1	Transfer byte from port addressed by C to memory addressed by HL. Decrement HL, decrement B.
OUT	port,A	2	D3 yy							Output from A to I/O port addressed by A7-AØ.
OUT	(C),reg	2	ED Ø1sss ØØ1							Output from register to port addressed by C.
OTIR		2	ED B3		1	?	?	?	1	Block transfer to port addressed by C from memory addressed by incrementing HL. Stop when B, which decrements after each byte, reaches Ø.
OTDR		2	ED BB		1	?	?	?	1	Block transfer to port addressed by C from memory addressed by decrementing HL. Stop when B, which decrements after each byte, reaches Ø.
OUTI		2	ED A3		X	?	?	?	1	Transfer byte to port addressed by C from memory addressed by HL. Increment HL, decrement B.
OUTD		2	ED AB		X	?	?	?	1	Transfer byte to port addressed by C from memory addressed by HL. Decrement HL, decrement B.

IMMEDIATE LOAD INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
LD	reg,data	2	00ddd110 yy							Load data into specified register.
LD	rp,data 16	3	00xx0001 yyyy							Load two data bytes into specified register pair.
LD	IX,data 16	4	DD 21 yyyy							Load two data bytes into specified index.
LD	IY,data 16	4	FD 21 yyyy							
LD	(HL),data	2	36 yy							Store data in memory addressed by HL.
LD	(IX+disp),data	4	DD 36 yy yy							Store data in memory addressed relative to index.
	(IY+disp),data	4	FD 36 yy yy							

REGISTER/MEMORY LOAD INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
LD	A,(addr)	3	3A ppqq							Load A from directly addressed memory.
LD	BC,(addr)	4	ED 01xx1011 ppqq							Load specified pair of registers from directly addressed memory.
	DE,(addr)	4	ED 01xx1011 ppqq							
	HL,(addr)	3	2A ppqq							
	SP,(addr)	4	ED 01xx1011 ppqq							
	IX,(addr)	4	DD 2A ppqq							
	IY,(addr)	4	FD 2A ppqq							
LD	(addr),A	3	32 ppqq							Store A contents in directly addressed memory.
LD	(addr),BC	4	ED 43 ppqq							Store contents of specified register pair in directly addressed memory.
	(addr),DE	4	ED 53 ppqq							
	(addr),HL	3	22 ppqq							
	(addr),SP	4	ED 73 ppqq							
	(addr),IX	4	DD 22 ppqq							
	(addr),IY	4	FD 22 ppqq							
LD	A,(BC)	1	0A							Load A with contents of memory addressed by specified register pair.
	A,(DE)	1	1A							
LD	reg,(HL)	1	01ddd110							Load specified register with contents of memory addressed by HL.
LD	(BC),A	1	02							Store contents of A in memory addressed by specified register pair.
	(DE),A	1	12							
LD	(HL),reg	1	01110sss							Store contents of specified register in memory addressed by contents of HL.

LD	reg,(IX+disp)	3	DD Øldddl1Ø yy							Store contents of memory addressed relative to index in specified register.
	reg,(IY+disp)	3	FD Øldddl1Ø yy							
LD	(IX+disp),reg	3	DD Ø111Øsss							Store contents of specified register in memory addressed relative to index.
	(IY+disp),reg	3	FD Ø111Øsss							

BLOCK SEARCH/TRANSFER INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
LDIR		2	ED B0				Ø	Ø	Ø	Transfer block from memory addressed by HL to memory addressed by DE, incrementing both and decrementing BC with each byte. Stop when BC reaches Ø.
LDDR		2	ED B8				Ø	Ø	Ø	Transfer block from memory addressed by HL to memory addressed by DE, decrementing both and decrementing BC with each byte. Stop when BC reaches Ø.
LDI		2	ED A0				X	Ø	Ø	Transfer byte from memory addressed by HL to memory addressed by DE, incrementing both and decrementing BC.
LDD		2	ED A8				X	Ø	Ø	Transfer byte from memory addressed by HL to memory addressed by DE, decrementing both and decrementing BC.
CPIR		2	ED B1		X	X	X	X	1	Compare A contents with memory block addressed by incrementing HL. Decrement BC with each byte. Stop on match or BC contents Ø.
CPDR		2	ED B9		X	X	X	X	1	Compare A contents with memory block addressed by decrementing HL. Decrement BC with each byte. Stop on match or BC contents Ø.
CPI		2	ED A1		X	X	X	X	1	Compare A contents with memory byte addressed by HL. Increment HL and decrement BC.
CPD		2	ED A9		X	X	X	X	1	Compare A contents with memory byte addressed by HL. Decrement H1 and BC.

REGISTER/REGISTER LOAD INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
LD	regd,regs	1	Øldddsss							Move contents of source register to destination register.
LD	A,IV	2	ED 57		X	X	I	Ø	Ø	Move contents of Interrupt Vector register to A.
LD	A,R	2	ED 5F		X	X	I	Ø	Ø	Move contents of Refresh register to A.
LD	IV,A	2	ED 47							Load Interrupt Vector register with A contents.
LD	R,A	2	ED 4F							Load Refresh register with contents of A.
LD	SP,HL	1	F9							Move contents of HL to Stack Pointer.
LD	SP,IX SP,IY	2 2	DD F9 FD F9							Move contents of index register to Stack Pointer.
EX	DE,HL	1	EB							Move contents of HL to DE.
EX	AF,AF'	1	Ø8							Exchange primary and alternate program statuses.
EXX		1	D9							Exchange primary and alternate register pairs BC, DE, and HL.

STACK INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
PUSH	pr	1	11xxØ1Ø							Put contents of specified register pair on top of stack and decrement SP.
PUSH	IX IY	2 2	DD E5 FD E5							Put contents of index on top of stack and decrement SP.
POP	pr	1	11xxØØØ1							Put contents of top of stack in specified register pair and increment SP.
POP	IX IY	2 2	DD E1 DD E1							Put contents of top of stack in index and increment SP.
EX	(SP),HL	1	E3							Exchange contents of SP and HL.
EX	(SP),IX (SP),IY	2 2	DD E3 FD E3							Exchange contents of SP and index.

IMMEDIATE OPERATE INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERAND(S)	OPERATION
ADD	data	2	C6 yy	X	X	X	O	X	Ø	data	Add data to A. No Carry.
ADC	data	2	CE yy	X	X	X	O	X	Ø	data	Add data to A. Carry.
SUB	data	2	D6 yy	X	X	X	O	X	Ø	data	Subtract data from A. No Carry.
SBC	data	2	DE yy	X	X	X	O	X	Ø	data	Subtract data from A. Carry.
AND	data	2	E6 yy	Ø	X	X	P	1	Ø	data	And data with A.
OR	data	2	F6 yy	Ø	X	X	P	1	Ø	data	OR data with A.
XOR	data	2	EE yy	Ø	X	X	P	I	1	data	Exclusive-OR data with A.
CP	data	2	FE yy	X	X	X	O	X	1	data	Compare data with A. Only flags are affected.

REGISTER OPERATE INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
DAA		1	27	X	X	X	P	X		Decimal adjust A, assuming that A contents are the sum or difference of BCD operands.
CPL		1	2F					1	1	Complement Accumulator (ones complement).
NEG		2	ED 44	X	X	X	O	X	1	Negate Accumulator (twos complement).
INC	reg	1	00xxx100		X	X	O	X	Ø	Increment contents of specified register.
INC	rp	1	00xx0011							Increment contents of specified register pair.
INC	IX IY	1 1	DD 23 FD 23							Increment contents of index register.
DEC	reg	2	00xxx101		X	X	O	X	1	Decrement contents of specified register.
DEC	rp	1	00xx1011							Decrement contents of specified register pair.
DEC	IX IY	1 1	DD 2B FD 2B							Decrement contents of index register.

REGISTER/REGISTER OPERATE INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
ADD	reg	1	10000xxx	X	X	X	O	X	Ø	Add contents of specified register to A. No carry.
ADC	reg	1	10001xxx	X	X	X	O	X	Ø	Add contents of specified register to A. Carry.
SUB	reg	1	10010xxx	X	X	X	O	X	1	Subtract contents of specified register from A. No Carry.
SBC	reg	1	10011xxx	X	X	X	O	X	1	Subtract contents of specified register from A. Carry.
AND	reg	1	10100xxx	Ø	X	X	P	1	Ø	AND A with contents of specified register.
OR	reg	1	10110xxx	Ø	X	X	P	1	Ø	OR A with contents of specified register.
XOR	reg	1	10101xxx	Ø	X	X	P	1	Ø	Exclusive-OR A with contents of specified register.
CP	reg	1	10111xxx	X	X	X	O	X	1	Compare contents of A and of specified register. Only flags are affected.
ADD	HL,rp	1	00xx1001	X				?	Ø	Add contents of specified register pair to contents of HL. No Carry.
ADC	HL,rp	2	ED 01xx1010	X	X	X	Ø	?	Ø	Add contents of specified register pair to contents of HL. Carry.
SBC	HL,rp	2	ED 01xx0010	X	X	X	Ø	?	1	Subtract contents of specified register pair from contents of HL. Carry.
ADD	IX,rr IY,rr	2 2	DD 00xx1001 FD 00xx1001	X				?	Ø	Add contents of specified register pair (BC, DE, SP, other index) to contents of index. No Carry.

REGISTER/MEMORY OPERATE INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
ADD	(HL)	1	86	X	X	X	O	X	Ø	Add to A the contents of memory addressed by HL. No Carpy.
ADD	(IX+disp) (IY+disp)	3 3	DD 86 yy FD 86 yy	X	X	X	O	X	Ø	Add to A the contents of memory addressed relative to index. No Carry.
ADC	(HL)	1	8E	X	X	X	O	X	Ø	Add to A the contents of memory addressed by HL. Carry.

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
ADC	(IX+disp) (IY+disp)	3 3	DD 8E yy FD 8E yy	X	X	X	O	X	Ø	Add to A the contents of memory addressed relative to index. Carry.
SUB	(HL)	1	96	X	X	X	O	X	1	Subtract from A the contents of memory addressed by HL. No Carry.
SUB	(IX+disp) (IY+disp)	3 3	DD 96 yy FD 96 yy	X	X	X	O	X	1	Subtract from A the contents of memory addressed relative to index. No Carry.
SBC	(HL)	1	9E	X	X	X	O	X	1	Subtract from A the contents of memory addressed by HL. Carry.
SBC	(IX+disp) (IY+disp)	3 3	DD 9E yy FD 9E yy	X	X	X	O	X	1	Subtract from A the contents of memory addressed relative to index. Carry.
AND	(HL)	1	A6	Ø	X	X	P	1	Ø	AND A with the contents of memory addressed by HL.
AND	(IX+disp) (IY+disp)	3 3	DD A6 yy FD A6 yy	Ø	X	X	P	1	Ø	AND A with the contents of memory addressed relative to index.
OR	(HL)	1	B6	Ø	X	X	P	1	Ø	OR A with the contents of memory addressed by HL.
OR	(IX+disp) (IY+disp)	3 3	DD B6 yy FD B6 yy	Ø	X	X	P	1	Ø	OR A with the contents of memory addressed relative to index.
XOR	(HL)	1	AE	Ø	X	X	P	1	Ø	Exclusive-OR A with memory addressed by HL.
XOR	(IX+disp) (IY+disp)	3 3	DD AE yy FD AE yy	Ø	X	X	P	1	Ø	Exclusive-OR A with the contents of memory addressed relative to index.
CP	(HL)	1	BE	X	X	X	O	X	1	Compare A with contents of memory addressed by HL. Only flags are affected.
CP	(IX+disp) (IY+disp)	3 3	DD BE yy FD BE yy	X	X	X	O	X	1	Compare A with contents of memory addressed relative to index. Only flags are affected.
INC	(HL)	1	34		X	X	O	X	Ø	Increment contents of memory addressed by HL.
INC	(IX+disp) (IY+disp)	3 3	DD 34 yy FD 34 yy		X	X	O	X	Ø	Increment contents of memory addressed relative to index.
DEC	(HL)	1	35		X	X	O	X	1	Decrement contents of memory addressed by HL.
DEC	(IX+disp) (IY+disp)	3 3	DD 35 yy FD 35 yy		X	X	O	X	1	Decrement contents of memory addressed relative to index.

REGISTER SHIFT AND ROTATE INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
RLCA		1	07	X				0	0	Rotate A left: B7 to B0 and Carry.
RLA		1	17	X				0	0	Rotate A left: B7 to Carry, Carry to B0.
RRCA		1	0F	X				0	0	Rotate A right: B0 to B7 and Carry.
RRA		1	1F	X				0	0	Rotate A right: B0 to Carry, Carry to B7.
RLC	reg	2	CB 00000xxx	X	X	X	P	0	0	Rotate contents of specified register left: B7 to B0 and Carry.
RL	reg	2	CB 00010xxx	X	X	X	P	0	0	Rotate contents of specified register left: B7 to Carry, Carry to B0.
RRC	reg	2	CB 00001xxx	X	X	X	P	0	0	Rotate contents of specified register right: B0 to B7 and Carry.
RR	reg	2	CB 00011xxx	X	X	X	P	0	0	Rotate contents of specified register right: B0 to Carry, Carry to B7.
SLA	reg	2	CB 00100xxx	X	X	X	P	0	0	Shift contents of specified register left and clear B0.
SRA	reg	2	CB 00101xxx	X	X	X	P	0	0	Shift contents of specified register right: B7 to B7 and B6.
SRL	reg	2	CB 00111xxx	X	X	X	P	0	0	Shift contents of specified register right and clear B7.
RLD		2	ED 6F	X	X	X	P	0	0	Rotate BCD digit left between A and memory addressed by HL: low nibble A to low nibble of memory, low nibble of memory to high nibble of memory, high nibble of memory to low nibble of A. High nibble of A unaffected.
RRD		2	ED 67	X	X	X	P	0	0	Rotate BCD digit right between A and memory addressed by HL: low nibble A to high nibble of memory, high nibble of memory to low nibble of memory, low nibble of memory to low nibble of A. High nibble of A unaffected.

MEMORY SHIFT AND ROTATE INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
RLC	(HL)	2	CB 06	X	X	X	P	0	0	Rotate left the contents of memory addressed by HL: B7 to B0 and Carry.
RLC	(IX+disp)	4	DD CB yy 06	X	X	X	P	0	0	Rotate left the contents of memory addressed relative to index: B7 to B0 and Carry.
	(IY+disp)	4	FD CB yy 06							
RL	(HL)	2	CB 16	X	X	X	P	0	0	Rotate left the contents of memory addressed by HL: B7 to Carry, Carry to B0.
RL	(IX+disp)	4	DD CB yy 16	X	X	X	P	0	0	Rotate left the contents of memory addressed relative to index: B7 to Carry, Carry to B0.
	(IY+disp)	4	FD CB yy 16							
RRC	(HL)	2	CB 0E	X	X	X	P	0	0	Rotate right the contents of memory addressed by HL: B0 to B7 and Carry.
RRC	(IX+disp)	4	DD CB yy 0E	X	X	X	P	0	0	Rotate right the contents of memory addressed relative to index: B0 to B7 and Carry.
	(IY+disp)	4	FD CB yy 0E							
RR	(HL)	2	CB 1E	X	X	X	P	0	0	Rotate right the contents of memory addressed by HL: B0 to Carry, Carry to B7.
RR	(IX+disp)	4	DD CB yy 1E	X	X	X	P	0	0	Rotate right the contents of memory addressed relative to index: B0 to Carry, Carry to B7.
	(IY+disp)	4	FD CB yy 1E							
SLA	(HL)	2	CB 26	X	X	X	P	0	0	Shift left the contents of memory addressed by HL: B7 to Carry, clear B0.
SLA	(IX+disp)	4	DD CB yy 26	X	X	X	P	0	0	Shift left the contents of memory addressed relative to index: B7 to Carry, clear B0.
	(IY+disp)	4	FD CB yy 26							
SRA	(HL)	2	CB 2E	X	X	X	P	0	0	Shift right the contents of memory addressed by HL: B7 to B6 and B7, B0 to Carry.
SRA	(IX+disp)	4	DD CB yy 2E	X	X	X	P	0	0	Shift right the contents of memory addressed relative to index: B7 to B6 and B7, B0 to Carry.
	(IY+disp)	4	DD CB yy 2E							
SRL	(HL)	2	CB 3E	X	X	X	P	0	0	Shift right the contents of memory addressed by HL: clear B7, B0 to Carry.
SRL	(IX+disp)	4	DD CB yy 3E	X	X	X	P	0	0	Shift right the contents of memory addressed relative to index: clear B7, B0 to Carry.
	(IY+disp)	4	FD CB yy 3E							

BIT MANIPULATION INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
BIT	b,reg	2	CB 01bbbxxx		X	?	?	1	0	Complement specified bit of specified register in Zero flag.
BIT	b,(HL)	2	CB 01bbb110		X	?	?	1	0	Complement of specified bit of contents of memory addressed by HL in Zero flag.
BIT	b,(IX+disp)	4	DD CB yy 01bbb110		X	?	?	1	0	Complement of specified bit of contents of memory addressed relative to index in Zero flag.
	b,(IY+disp)	4	FD CB yy 01bbb110							
SET	b,reg	2	CB 11bbbxxx							Set specified bit of specified register.
SET	b,(HL)	2	CB 11bbb110							Set specified bit of memory addressed by HL.
SET	b,(IX+disp)	4	DD CB yy 11bbb110							Set specified bit of memory addressed relative to index.
	b,(IY+disp)	4	FD CB yy 11bbb110							
RES	b,reg	2	CB 10bbbxxx							Reset specified bit of specified register.
RES	b,(HL)	2	CB 10bbb110							Reset specified bit of memory addressed by HL.
RES	b,(IX+disp)	4	DD CB yy 10bbb110							Reset specified bit of memory addressed relative to index.
	b,(IY+disp)	4	FD CB yy 10bbb110							

STATUS INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
SCF		1	37	1				0	0	Set Carry flag.
CCF		1	3F	?				?	0	Complement Carry flag.

JUMP INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
JP	label	3	C3 ppqq							Jump to address represented by label.
JR	disp	2	18 yy							Jump relative to present contents of PC.
JP	(HL)	1	E9							Jump to address contained in HL.

JP	(IX)	2	DD E9								Jump to address contained in index.
	(IY)	2	FD E9								

CONDITIONAL JUMP INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
JP	C,label	3	DA ppqq							Jump to address represented by label if condition is true; otherwise, continue in sequence.
	M,label	3	FA ppqq							
	NC,label	3	D2 ppqq							
	NZ,label	3	C2 ppqq							
	P,label	3	F2 ppqq							
	PE,label	3	EA ppqq							
	PO,label	3	E2 ppqq							
	Z,label	3	CA ppqq							
JR	C,disp	2	38 yy							Jump relative to PC contents if Carry is set.
JR	NC,disp	2	30 yy							Jump relative to PC contents if Carry is reset.
JR	Z,disp	2	28 yy							Jump relative to PC contents if Zero is set.
JR	NZ,disp	2	20 yy							Jump releative to PC contents if Zero is reset.
DJNZ	disp	2	10 yy							Decrement contents of B and jump relative to PC contents if result is not 0.

SUBROUTINE INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
CALL	label	3	CD ppqq							Jump to subroutine starting at address represented by label.
CALL	C,label	3	DC ppqq							Jump to subroutine starting at address represented by label if condition is satisfied; otherwise, continue in sequence.
	M,label	3	FC ppqq							
	NC,label	3	D4 ppqq							
	NZ,label	3	C4 ppqq							
	P,label	3	F4 ppqq							
	PE,label	3	EC ppqq							
	PO,label	3	E4 ppqq							
	Z,label	3	CC ppqq							
RET		1	C9							Return from subroutine to next instruction after last CALL.

RET	C,label	3	D8								Return from subroutine to next instruction after last CALL if condition is satisfied; otherwise, continue in sequence.
	M,label	3	F8								
	NC,label	3	D0								
	NZ,label	3	C0								
	P,label	3	F0								
	PE,label	3	E8								
	PO,label	3	E0								
	Z,label	3	C8								

INTERRUPT INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
DI		1	F3							Disable interrupts.
EI		1	FB							Enable interrupts.
RST	n	1	11xxx111							Save PC on stack and restart at location (8*n)h.
RETI		2	ED 4D							Return from interrupt.
RETN		0	ED 45							Return from nonmaskable interrupt.
IM	0	2	ED 46							Select interrupt mode 0, 1, or 2.
	1	2	ED 56							
	2	2	ED 5E							

NO OPERATION INSTRUCTIONS

MNEMONIC	OPERAND(S)	BYTES	OBJECT CODE	C	Z	S	P/O	AC	N	OPERATION
NOP		1	00							No operation; volatile memories refreshed.
HALT		1	76							CPU halts, executes NOP instructions.

CHAPTER 2

PROGRAMMING THE SIO

Each channel of the Serial I/O Controller is separately programmed. Channel A has seven command (write) registers and two status (read) registers; Channel B has eight command registers and three status registers. Address bits A1 and AØ select the channel and command or status port as shown in Table 2.1. The first register accessed after a reset is always a Ø register (either command or status); the level of the next register accessed is determined by bits 2-Ø of Command Register Ø; the next register accessed is a Ø register again.

A1	AØ	FUNCTION
Ø	Ø	Ch. A Data
Ø	1	Ch. A Command
1	Ø	Ch. B Data
1	1	Ch. B Command

TABLE 2.1

The programmer should keep in mind that the following programming instructions are based on the assumption that the SIO is a DTE device. (This is the standard assumption made for serial I/O devices.) If the SIO functions as DCE (as it does, for example, on the 282Ø), the definitions of transmission and reception assumed by the Programming Guide and by the RS-232-C specifications will be opposite each other. RS-232-C specifications define transmission as data transfer from a peripheral to the CPU. SIO data sheets define transmission as output from the SIO, i.e., transfer from the CPU to the peripheral when the SIO is used as DCE. Thus, for SIO's used as DCE, data received by the SIO comes in on interface line TxD, data transmitted by the SIO goes out on RxD, and the RTS, CTS, and DTR bits correspond to interface lines CTS, RTS, and DSR respectively.

2.1 COMMAND REGISTER Ø

This register is broken into three blocks: a three-bit register pointer, a three-bit command, and a two-bit CRC reset code.

7	6	5	4	3	2	1	0
CRC RESET 1	CRC RESET Ø	BASIC COMMAND 2	BASIC COMMAND 1	BASIC COMMAND Ø	REGISTER POINTER 2	REGISTER POINTER 1	REGISTER POINTER Ø

COMMAND REGISTER 0

Bits 2-Ø These bits select the next register to be accessed (either command or status) as shown in Table 2.2.

BIT 2	BIT 1	BIT Ø	NEXT REGISTER
Ø	Ø	Ø	Register Ø
Ø	Ø	1	Register 1
Ø	1	Ø	Register 2
Ø	1	1	Register 3
1	Ø	Ø	Register 4
1	Ø	1	Register 5
1	1	Ø	Register 6
1	1	1	Register 7

TABLE 2.2

Bits 5-3 These three bits select one of eight commands as indicated in Table 2.3.

BIT 5	BIT 4	BIT 3	COMMAND
Ø	Ø	Ø	Ø: Null Code
Ø	Ø	1	1: Send Abort (SDLC Mode)
Ø	1	Ø	2: Reset Ext/Status Interrupts
Ø	1	1	3: Channel Reset
1	Ø	Ø	4: Reset Rx 1st Char Interrupt
1	Ø	1	5: Reset Tx Interrupt Pending
1	1	Ø	6: Reset Error Latches
1	1	1	7: Return From Interrupt (Ch. A)

TABLE 2.3

The NULL CODE, which has no effect, is used when Command Register Ø has been accessed to point to another register or for a CRC Reset.

The SEND ABORT command, used only in the SDLC

mode, causes a sequence of 8 to 13 ones to be transmitted.

The RESET EXTERNAL/STATUS INTERRUPTS command resets Status Register Ø Bits 7-3, which are all latched when a change occurs in one, and re-enables interrupts.

The CHANNEL RESET command has the same effect as a reset of the SIO except that it is limited to the channel addressed. When a channel is reset, four system clock cycles should be allowed before the channel is accessed again.

The RESET RECEIVER FIRST CHARACTER INTERRUPTS command enables interrupts after a first character interrupt has occurred. When a First-Character-Only Receiver Interrupt occurs (see Command register 1, Bits 4-3), it disables further interrupts except External/Status interrupts. This command is used to enable further interrupts.

If Transmitter Buffer Empty interrupts are enabled but there are no additional characters ready to be sent, issuing the RESET TRANSMITTER INTERRUPT PENDING command will prevent further transmitter interrupts until after the next character has been loaded into the transmitter.

The RESET ERROR LATCHES command resets the Status Register 1 bits indicating parity, overrun, framing, and CRC errors and re-enables interrupts on these error conditions.

The RETURN FROM INTERRUPT command, which must be issued only to Channel A, is not needed when using the SIO with a Z-8Ø CPU. To the SIO this command is identical to the RETI instruction on the data bus, allowing implementation of the internal SIO daisy chain in systems with no external daisy chain. The command resets the current interrupt, allowing the highest priority interrupt that the SIO has pending to be acknowledged (or, if the SIO has no interrupt pending, allowing the device's IEO to go high).

Bits 7-6 These bits are used to specify one of three CRC RESET modes as indicated in Table 2.4.

BIT 7	BIT 6	COMMAND
Ø	Ø	Null Code
Ø	1	Reset Rx CRC Checker
1	Ø	Reset Tx CRC Checker
1	1	Reset Tx UNDERRUN/EOM

TABLE 2.4

The NULL command, which has no effect, is used when Command Register Ø has been accessed for reasons other than a CRC Reset.

The RESET Rx CRC CHECKER command resets Command Register 3 Bit 3, disabling CRC checking by a receiver.

The RESET Tx CRC CHECKER command resets Command Register 5 Bit Ø, disabling CRC generation by a transmitter.

The RESET Tx UNDERRUN/EOM command resets Status Register Ø Bit 6.

2.2 COMMAND REGISTER 1

This register holds the control bits for various interrupt and Wait/Ready modes.

7	6	5	4	3	2	1	0
WAIT/ READY ENABLE	WAIT/ READY SELECT	WAIT/ READY Rx or Tx	Rx INT MODE 1	Rx INT INT Ø	STATUS AFFECTS VECTOR	Tx INT ENABLE	EXTERNAL INT ENABLE

COMMAND REGISTER 1

Bit Ø Setting this bit, EXTERNAL INTERRUPT ENABLE, allows interrupts to occur when any of Bits 7-3 of Status Register Ø changes state.

Bit 1 Setting this bit, TRANSMITTER INTERRUPT ENABLE, allows interrupts to occur when data is shifted from the Transmitter Buffer to the Transmitter Register.

Bit 2 Setting the STATUS AFFECTS VECTOR bit causes the interrupt vector to vary according to the interrupt condition as indicated in Table 2.5. If this bit is Ø, the vector programmed into the Interrupt Vector Register will be put on the bus no matter what the interrupt condition.

IV3	IV2	IV1	CONDITION
Ø	Ø	Ø	Ch. B Tx Buffer Empty
Ø	Ø	1	Ch. B Ext/Status Change
Ø	1	Ø	Ch. B Rx Char Available
Ø	1	1	Ch. B Special Rx Condition
1	Ø	Ø	Ch. A Tx Buffer Empty
1	Ø	1	Ch. A Ext/Status Change
1	1	Ø	Ch. A Rx Char Available
1	1	1	Ch. A Special Rx Condition

TABLE 2.5

Bits 4-3 These bits determine the RECEIVER INTERRUPT MODE according to Table 2.6. One of these bits must be 1 for framing, parity, overrun, and SDLC end-of-character interrupts to be enabled.

BIT 4	BIT 3	RECEIVER INTERRUPT MODE
Ø	Ø	Ø: Rx Interrupts Disabled
Ø	1	1: Rx Interrupt on 1st Character Only
1	Ø	2: Interrupt on All Rx Characters; Parity Affects Vector
1	1	3: Interrupt on All Rx Characters; Parity Does Not Affect Vector

TABLE 2.6

Bit 5 Setting this bit causes the corresponding WAIT/READY line, if enabled, to be active when the Receiver is empty; clearing it cause WAIT/READY to be active when the Transmitter Buffer is full.

Bit 6 This bit conditions the WAIT/READY line as a READY output to a DMAC when set or as a WAIT output to the CPU when cleared.

Bit 7 Setting this bit enables the WAIT/READY line; when this bit is cleared, WAIT/READY is high (Ready mode) or floats (Wait mode).

2.3 COMMAND REGISTER 2

This register may be written to at either the Port A or the Port B address, but may be read only at the Port B address. Bits 3-1 will be re-written according to the type of interrupt as shown in Table 2.5 if the Status Affects Vector bit (Command Register 1 Bit 2) is set. Bit Ø will always be read by the CPU as a Ø because the vector is used to point to a pair of addresses whose base is even.

7	6	5	4	3	2	1	0
INT VECTOR 7	INT VECTOR 6	INT VECTOR 5	INT VECTOR 4	INT VECTOR 3	INT VECTOR 2	INT VECTOR 1	INT VECTOR Ø

COMMAND REGISTER 2

2.4 COMMAND REGISTER 3

The bits of this register control receiver logic.

7	6	5	4	3	2	1	0
Rx BITS/ CHAR 1	Rx BITS/ CHAR Ø	AUTO ENABLES	ENTER HUNT	Rx CRC ENABLE	SDLC ADDRESS SEARCH	SYNC LOAD INHIBIT	Rx ENABLE

COMMAND REGISTER 3

Bit Ø — Setting this bit causes receiver operations to begin as programmed.

Bit 1 — If this bit is set, sync characters are stripped from incoming data before it is transferred to the CPU. Otherwise, the CPU receives the sync characters as data. CRC calculation is not affected. In Async and SDLC modes this bit should be cleared.

Bit 2 — This bit is used in the SDLC mode only. When set, it allows only messages preceded by the programmed address (the contents on

Command Register 6) or the global address (11111111) to be received. If this bit is cleared, all SDLC messages will be accepted. This bit should be cleared for Sync and Async modes.

Bit 3 Setting the RCVR CRC ENABLE bit causes CRC calculation to begin or restart with the last character transferred from the receiver register to the buffer stack. Clearing this bit turns off CRC calculation for all characters transferred to the buffer stack while the bit is Ø.

Bit 4 The ENTER HUNT MODE bit should be set if character sync is lost or if an SDLC message has been determined to be unneeded. When this bit is set, receiver logic begins looking for sync or flag characters.

Bit 5 Setting this bit causes the DCD* and CTS* inputs to function as receiver and transmitter AUTO ENABLES respectively; transmitting and receiving will not begin until the corresponding input is low. The status bits DCD and CTS are not affected by the state of this bit.

Bits 7-6 These two bits determine, according to Table 2.7, the number of incoming bits that will be assembled to form a character.

BIT 7	BIT 6	BITS/CHAR
Ø	Ø	5
Ø	1	7
1	Ø	6
1	1	8

TABLE 2.7

2.5 COMMAND REGISTER 4

The control bits held by this register affect basic receiver and transmitter parameters. This register should be the first register written to in any initialization sequence (the one possible exception being Command Register 2).

7	6	5	4	3	2	1	0
CLOCK RATE 1	CLOCK RATE Ø	SYNC MODE 1	SYNC MODE Ø	MODE/ # STOP BITS 1	MODE/ # STOP BITS Ø	EVEN/ ODD PARITY	PARITY ENABLE

COMMAND REGISTER 4

Bit Ø — Setting this bit causes a parity bit to be inserted in transmitted characters and expected in received characters. This bit should be disabled for SDLC mode.

Bit 1 — A 1 in this bit selects even parity, a Ø odd. The bit has no effect if parity is disabled.

Bits 3-2 — These bits select either SYNC or ASYNC mode, and in Async mode the number of stop bits per character, as shown in Table 2.8.

BIT 3	BIT 2	MODE / STOP BITS
Ø	Ø	Sync Modes
Ø	1	Async ; 1 Stop Bit
1	Ø	Async ; 1.5 Stop Bits
1	1	Async ; 2 Stop Bits

TABLE 2.8

Bits 5-4 — If Sync mode is selected by Bits 3-2, these bits select one of four Sync formats as indicated in Table 2.9.

BIT 5	BIT 4	SYNC MODE
Ø	Ø	8-Bit Programmed Sync
Ø	1	16-Bit Programmed Sync
1	Ø	SDLC Mode (Ø1111110 Sync)
1	1	External Sync Mode

TABLE 2.9

Bits 7-6 — These bits select the clock rate divisor which determines the data rate, as indicated in Table 2.1Ø. In Sync mode these bits must be set to ØØ (divisor of 1). In Async mode these bits determine whether data will be sampled every 64th, every 32nd, every 16th, or every clock cycle. The divisor is used for both receiver and transmitter.

BIT 7	BIT 6	DATA RATE
Ø	Ø	Clock Rate
Ø	1	Clock Rate / 16
1	Ø	Clock Rate / 32
1	1	Clock Rate / 64

TABLE 2.1Ø

2.6 COMMAND REGISTER 5

This register consists primarily of control bits which affect transmitter logic.

7	6	5	4	3	2	1	0
DTR	Tx BITS CHAR 1	Tx BITS CHAR Ø	SEND BREAK	Tx ENABLE	CRC TYPE SELECT	RTS	Tx CRC ENABLE

COMMAND REGISTER 5

Bit Ø — If this bit, TRANSMITTER CRC ENABLE, is set at the time a character is loaded from the transmitter buffer to the transmitter shift register, CRC will be calculated for that character. CRC characters will be sent when the transmitter becomes completely empty only if this bit is set. This bit should be set during Sync or SDLC transmission and cleared during Async transmission.

Bit 1 — The RTS bit controls the RTS* pin. In Sync and SDLC modes the RTS* pin is the complement of this bit. In Async mode, the RTS* pin goes high in response to the RTS bit being Ø only after the transmitter buffer has emptied.

Bit 2 — This bit selects the CRC code used by both receiver and transmitter. A 1 selects the CRC-16 polynomial, x16 + x15 + x2 + 1. A Ø selects the SDLC polynomial x16 + x12 + x5 + 1.

Bit 3 — Setting this bit enables data transmission. Resetting this bit causes the transmitter to be held in a marking state after the transmission in process is completed (with the exception of CRC characters, which will not be completed if this bit is reset during their transmission).

Bit 4 — When the SEND BREAK bit is set, the TxD pin is forced to the spacing state, remaining in that state until the SEND BREAK bit is reset.

Bits 6-5 These bits control the number of bits transmitted for each byte loaded into the transmitter buffer, as shown in Table 2.11. The bits are assumed to be right-justified. If five or fewer bits from a byte are desired, write the code for 5 bits/character and format the byte so that the desired bits are right-justified, the next three rightmost bits are Ø's, and all additional high-order bits are 1's. For example, a byte in which only three bits are to be interpreted by the transmitter as data for transmission would be formatted 11ØØØddd.

BIT 6	BIT 5	BITS/CHAR
Ø	Ø	5
Ø	1	7
1	Ø	6
1	1	8

TABLE 2.11

Bit 7 The DTR bit controls the DTR* pin. Setting this bit forces DTR* active; resetting it forces DTR* inactive.

2.7 COMMAND REGISTER 6

This register is used for all Sync modes. In MONOSYNC it holds the 8-bit sync character for transmitted messages. In BISYNC it holds the low-order byte of the 16-bit sync character. In EXTERNAL SYNC it contains the Sync character inserted in transmission underruns and matched during receiving. In SDLC it holds the check address which must precede an incoming message unless Command Register 3 Bit 2 is reset. The register has no effect on other modes.

7	6	5	4	3	2	1	0
SYNC/ ADDRESS 7	SYNC/ ADDRESS 6	SYNC/ ADDRESS 5	SYNC/ ADDRESS 4	SYNC/ ADDRESS 3	SYNC/ ADDRESS 2	SYNC/ ADDRESS 1	SYNC/ ADDRESS Ø

COMMAND REGISTER 6

2.8 COMMAND REGISTER 7

This register is used only for MONOSYNC, BISYNC and SDLC modes. In MONOSYNC it holds the match for Sync characters received. In BISYNC it holds the high-order byte of the 16-bit sync character. In SDLC it must contain the flag Ø1111110. The register has no effect on other modes.

7	6	5	4	3	2	1	0
SYNC/ FLAG 15	SYNC/ FLAG 14	SYNC/ FLAG 13	SYNC/ FLAG 12	SYNC/ FLAG 11	SYNC/ FLAG 10	SYNC/ FLAG 9	SYNC/ FLAG 8

COMMAND REGISTER 7

2.9 STATUS REGISTER Ø

This register holds various status bits for use during a data transfer operation. Bits 7-3, the External/Status Interrupt bits, are all latched when any one bit changes state and are reset by the RESET EXTERNAL/STATUS INTERRUPTS command (Command Register Ø Bits 5-3 = Ø1Ø). An External/Status bit thus indicates current state only if no change has occurred in any of the five bits since the last reset.

Received characters are transmitted from the Receiver Register to the bottom of a three-buffer FIFO stack. Each buffer has its own Error Buffer. When a receiver character is read, the contents of the topmost buffer are put on the data bus and the stack pointer decrements. When this occurs, the error status information for the character just read is lost. Therefore, the error status of a received character may be read only BEFORE the character is read.

7	6	5	4	3	2	1	0
BREAK/ ABORT	EOM/ UNDERRUN	CTS	SYNC/ HUNT	DCD	Tx BUFFER EMPTY	INT PENDING	Rx CHAR AVAIL

STATUS REGISTER 0

Bit Ø — This bit is 1 when a character is available in the receiver buffers. Interrupts on this condition are enabled by Command Register 1 Bits 4-3.

Bit 1 — Though this bit is available from Channel A only, it indicates an INTERRUPT PENDING in either channel. In Channel B this bit is always Ø.

Bit 2 — This bit is set whenever the Transmitter Buffer is empty (except when a CRC character is being sent).

Bit 3 — This bit reflects the state of the DCD* pin until latched as described above.

Bit 4 — In Async mode the SYNC/HUNT bit reflects the state of the SYNC* input. In external Sync mode, the SYNC* pin is also an input, indicating that synchronization has been achieved by the peripheral. In the internal Sync mode, SYNC* is an output active when the receiver recognizes the sync character. In either Sync mode the SYNC/HUNT bit is set by the SYNC* input being asserted low and reset by Command Register 3 Bit 4 (ENTER HUNT).

Bit 5 — This bit reflects the state of the CTS* pin until latched as described above.

Bit 6 — The END-OF-MESSAGE/UNDERRUN bit is set following a reset of the SIO and by transmitter underrun and end-of-message interrupts; it is reset by 11 written to Bits 7-6 of Command Register Ø. Underruns occur only in Sync and SDLC modes. Resetting this bit after transmission has begun causes a transmitter underrun to be interpreted as an end-of-message indicator; the CRC character is then transmitted, followed by the appropriate end-of-message character(s). Leaving this bit set is not allowed in SDLC mode; in Sync mode it causes sync characters to be inserted and Status Register Ø Bit 2 to be set whenever an underrun is detected.

Bit 7 — In Async mode this bit is set when a break is detected. The bit will be reset when the

break condition is removed only if a RESET EXTERNAL/STATUS INTERRUPTS command has been issued. In SDLC this bit is set when an abort sequence (seven or more 1's) is detected. The bit is not used in other Sync modes.

2.10 STATUS REGISTER 1

This register holds status bits (chiefly error indicators) and the residue code for SDLC mode. These bits are used at the end of an operation.

7	6	5	4	3	2	1	0
END OF FRAME (SDLC)	CRC/ FRAMING ERROR	Rx OVERRUN ERROR	PARITY ERROR	RESIDUE CODE 2	RESIDUE CODE 1	RESIDUE CODE Ø	ALL SENT

STATUS REGISTER 1

Bit Ø The ALL SENT bit indicates, in Async mode, that all characters have completely cleared the transmitter. It does not cause interrupts and is always set in Sync mode.

Bits 3-1 The RESIDUE CODE is used only in SDLC mode and indicates the residue if the I-Field is not an integral multiple of the character length. When the ending flag has been received (see Bit 7 of this register), these bits indicate how many valid bits occurred in the last two bytes, as shown in Table 2.12. If there is no residue, the residue code should always be Ø11.

BIT 3	BIT 2	BIT 1	I-FIELD PREV. BYTE	I-FIELD 2ND PREV. BYTE
1	Ø	Ø	Ø	3
Ø	1	Ø	Ø	4
1	1	Ø	Ø	5
Ø	Ø	1	Ø	6
1	Ø	1	Ø	7
Ø	1	1	Ø	8
1	1	1	1	8
Ø	Ø	Ø	2	8

TABLE 2.12

Bit 4 The latched PARITY ERROR bit is set whenever parity is enabled and a parity error is detected. The bit is reset only by the ERROR RESET command (see Command Register Ø Bits 5-3).

Bit 5 The latched RECEIVER OVERRUN ERROR bit indicates that more than four characters have been received without being read by the CPU. The bit is set when the overwritten character is read and reset by the ERROR RESET command. If the STATUS AFFECTS VECTOR bit (Command Register 1 Bit 2) is set, this bit set will generate a SPECIAL RECEIVE CONDITION interrupt.

Bit 6 When a framing error is detected during Async mode, this bit is set but not latched. In Sync mode this bit set indicates that the CRC calculation and check value do not match.

Bit 7 The END OF FRAME bit is used for SDLC mode only. Set, it indicates that a valid ending flag has been received and that the CRC error and residue bits in this register are valid.

2.11 STATUS REGISTER 2

This register is available in Channel B only and contains the interrupt vector. If the STATUS AFFECTS VECTOR bit is set, bits 3-1 will reflect any pending interrupt; if no interrupts are pending, they will hold Ø11. If the STATUS AFFECTS VECTOR bit is cleared, bits 3-1 will appear as written to Command Register 2.

7	6	5	4	3	2	1	0
INT VECTOR 7	INT VECTOR 6	INT VECTOR 5	INT VECTOR 4	INT VECTOR 3	INT VECTOR 2	INT VECTOR 1	INT VECTOR Ø

STATUS REGISTER 2

CHAPTER 3

PROGRAMMING THE DART

The Z-80 Dual Asynchronous Receiver/Transmitter provides two asynchronous serial I/O channels. Essentially, the DART functions as an SIO without synchronous capabilities; programming is the same except that Command Registers 6 and 7 do not exist and bits of other registers applicable only in synchronous mode are not used. Channel and command or status port selection is controlled by address bits A1 and A0 as indicated in Table 3.1. The first The first register accessed is always a 0 register (either command or status); the next register accessed is pointed to by Command Register 0; the third register accessed is a 0 register again.

A1	A0	FUNCTION
0	0	Ch. A Data
0	1	Ch. A Command
1	0	Ch. B Data
1	1	Ch. B Command

TABLE 3.1

3.1 COMMAND REGISTER 0

This register includes a three-bit register pointer and a three-bit command code. The register format is as follows:

7	6	5	4	3	2	1	0
NOT USED	NOT USED	BASIC COMMAND 2	BASIC COMMAND 1	BASIC COMMAND 0	REGISTER POINTER 2	REGISTER POINTER 1	REGISTER POINTER 0

COMMAND REGISTER 0

Bits 2-Ø These three bits select the register to be accessed on the next operation according to Table 3.2.

BIT 2	BIT 1	BIT Ø	NEXT REGISTER
Ø	Ø	Ø	Register Ø
Ø	Ø	1	Register 1
Ø	1	Ø	Register 2
Ø	1	1	Register 3
1	Ø	Ø	Register 4
1	Ø	1	Register 5
1	1	Ø	Register 6
1	1	1	Register 7

TABLE 3.2

Bit 5-3 These three bits are used for basic channel control commands as indicated in Table 3.3.

BIT 5	BIT 4	BIT 3	COMMAND
Ø	Ø	Ø	Ø: Null Code
Ø	Ø	1	1: Not Used
Ø	1	Ø	2: Reset Ext/Status Interrupts
Ø	1	1	3: Channel Reset
1	Ø	Ø	4: Reset Rx 1st Char Interrupt
1	Ø	1	5: Reset Tx Interrupt Pending
1	1	Ø	6: Reset Error Latches
1	1	1	7: Return From Interrupt (Ch. A)

TABLE 3.3

The NULL CODE, which has no effect, is used when Command Register Ø has been accessed to point to another register.

The RESET EXTERNAL/STATUS INTERRUPTS command resets Status Register Ø Bits 7-3, which are all latched when a change occurs in one, and re-enables interrupts.

The CHANNEL RESET command has the same effect as a reset of the DART except that it is limited to the channel addressed. When a channel is reset, four clock cycles should be allowed before that channel is accessed again.

The RESET RECEIVER FIRST CHARACTER INTERRUPTS command enables interrupts after a first character interrupt has occurred. When a First-Character-Only Receiver Interrupt occurs (see Command register 1, Bits 4-3), it disables further interrupts except External/Status

interrupts. This command is used to enable further interrupts.

If Transmitter Buffer Empty interrupts are enabled but there are no additional characters ready to be sent, issuing the SUPPRESS TRANSMITTER EMPTY INTERRUPT command will prevent further transmitter interrupts until after the next character has been loaded into the transmitter.

The ERROR RESET command resets Status Register 1 bits indicating parity, overrun, and framing errors and re-enables interrupts on these error conditions.

The RETURN FROM INTERRUPT command, which must be issued only to Channel A, is not needed when the DART is used with a Z-8Ø CPU. To the DART this command is identical to the RETI instruction on the data bus, allowing implementation of the internal DART daisy chain in systems with no external daisy chain. The command resets the current interrupt, allowing the highest priority interrupt that the DART has pending to be acknowledged (or, if the DART has no interrupt pending, allowing the device's IEO to go high).

Bits 7-6 These bits are not used.

3.2 COMMAND REGISTER 1

This register holds the control bits for various interrupt and Wait/Ready modes.

7	6	5	4	3	2	1	0
WAIT/ READY ENABLE	WAIT/ READY SELECT	WAIT/ READY Rx or Tx	Rx INT MODE 1	Rx INT INT Ø	STATUS AFFECTS VECTOR	Tx INT ENABLE	EXTERNAL INT ENABLE

COMMAND REGISTER 1

Bit Ø — Setting this bit, EXTERNAL INTERRUPT ENABLE, allows interrupts to occur when any of Bits 7-3 of Status Register Ø changes state.

Bit 1 — Setting this bit, TRANSMITTER INTERRUPT ENABLE, allows interrupts to occur when data is shifted from the Transmitter Buffer to the Transmitter Register.

Bit 2 — Setting the STATUS AFFECTS VECTOR bit causes the interrupt vector to vary according to the interrupt condition as indicated in the Table 3.4. If this bit is Ø, the vector programmed into the Interrupt Vector Register will be put on the bus no matter what the interrupt condition.

IV3	IV2	IV1	CONDITION
Ø	Ø	Ø	Ch. B Tx Buffer Empty
Ø	Ø	1	Ch. B Ext/Status Change
Ø	1	Ø	Ch. B Rx Char Available
Ø	1	1	Ch. B Special Rx Condition
1	Ø	Ø	Ch. A Tx Buffer Empty
1	Ø	1	Ch. A Ext/Status Change
1	1	Ø	Ch. A Rx Char Available
1	1	1	Ch. A Special Rx Condition

TABLE 3.4

Bits 4-3 — These bits determine the RECEIVER INTERRUPT MODE according to Table 3.5. One of these bits must be 1 for framing, parity, and overrun interrupts to be enabled.

BIT 4	BIT 3	RECEIVER INTERRUPT MODE
Ø	Ø	Ø: Rx Interrupts Disabled
Ø	1	1: Rx Interrupt on 1st Character Only
1	Ø	2: Interrupt on All Rx Characters; Parity Affects Vector
1	1	3: Interrupt on All Rx Characters; Parity Does Not Affect Vector

TABLE 3.5

Bit 5 — Setting this bit causes the corresponding WAIT/READY line, if enabled, to be active when the Receiver is empty; clearing it cause WAIT/READY to be active when the Transmitter Buffer is full.

Bit 6 — This bit conditions the WAIT/READY line as a READY output to a DMAC when a 1 or as a WAIT output to the CPU when a Ø.

Bit 7 Setting this bit enables the WAIT/READY line; when this bit is cleared, WAIT/READY is high (Ready mode) or floats (Wait mode).

3.3 COMMAND REGISTER 2

This register may be written to at either the Port A or the Port B address, but may be read at the Port B address only. Bits 3-1 will be re-written according to the type of interrupt as indicated in Table 3.4 if the Status Affects Vector bit (Command Register 1 Bit 2) is set. Bit Ø will always be read by the CPU as a Ø because the vector is used to point to a pair of addresses whose base is even.

7	6	5	4	3	2	1	0
INT VECTOR 7	INT VECTOR 6	INT VECTOR 5	INT VECTOR 4	INT VECTOR 3	INT VECTOR 2	INT VECTOR 1	INT VECTOR Ø

COMMAND REGISTER 2

3.4 COMMAND REGISTER 3

The bits of this register control receiver logic.

7	6	5	4	3	2	1	0
Rx BITS/ CHAR 1	Rx BITS/ CHAR Ø	AUTO ENABLES	NOT USED	NOT USED	NOT USED	NOT USED	Rx ENABLE

COMMAND REGISTER 3

Bit Ø Setting this bit causes receiver operations to begin as programmed.

Bits 4-1 These bits are not used.

Bit 5 Setting this bit causes the DCD* and CTS*

inputs to function as receiver and transmitter AUTO ENABLES respectively; transmitting and receiving will not begin until the corresponding input is low. The status bits DCD and CTS are not affected by the state of this bit.

Bits 7-6 These two bits determine, according to Table 3.6, the number of incoming bits that will be assembled to form a character.

BIT 7	BIT 6	BITS/CHAR
Ø	Ø	5
Ø	1	7
1	Ø	6
1	1	8

TABLE 3.6

3.5 COMMAND REGISTER 4

The control bits held by this register affect basic receiver and transmitter parameters. This register should be the first register written to in an initialization sequence (the one possible exception being Command Register 2).

7	6	5	4	3	2	1	0
CLOCK RATE 1	CLOCK RATE Ø	NOT USED 1	NOT USED Ø	# STOP BITS 1	#STOP BITS Ø	EVEN/ ODD PARITY	PARITY ENABLE

COMMAND REGISTER 4

Bit Ø Setting this bit causes a parity bit to be inserted in transmitted characters and expected in received characters.

Bit 1 A 1 in this bit selects even parity, a Ø odd. The bit has no effect if parity is disabled.

Bits 3-2 These bits select, according to Table 3.7, the number of stop bits per character.

BIT 3	BIT 2	STOP BITS
Ø	Ø	Not Used
Ø	1	1
1	Ø	1.5
1	1	2

TABLE 3.7

Bits 5-4 These bits are not used.

Bits 7-6 These bits select the clock rate divisor which determines the data rate, as indicated in Table 3.8.

BIT 7	BIT 6	DATA RATE
Ø	Ø	Clock Rate
Ø	1	Clock Rate / 16
1	Ø	Clock Rate / 32
1	1	Clock Rate / 64

TABLE 3.8

3.6 COMMAND REGISTER 5

This register consists primarily of control bits which affect transmitter logic.

7	6	5	4	3	2	1	0
DTR	Tx BITS CHAR 1	Tx BITS CHAR Ø	SEND BREAK	Tx ENABLE	NOT USED SELECT	RTS	NOT USED ENABLE

COMMAND REGISTER 5

Bit Ø This bit is not used.

Bit 1 The RTS bit controls the RTS* pin. The RST* pin goes high in response to the RTS bit being Ø only after the transmitter buffer has emptied.

Bit 2 This bit is not used.

Bit 3 Setting this bit enables data transmission. Clearing it causes the transmitter to be held in a marking state after the transmission in process is completed.

Bit 4 When the SEND BREAK bit is set, the TxD pin is forced to the spacing state, remaining in that state until the SEND BREAK bit is reset.

Bits 6-5 These bits control the number of bits transmitted from each byte loaded into the transmitter buffer, as shown in Table 3.9. The bits are assumed to be right-justified. If five or fewer bits from a byte are desired, write the code for 5 bits/character and format the byte so that the desired bits are right-justified, the next three rightmost bits are Ø's, and any additional high-order bits are 1's. For example, a byte in which only three bits are to be interpreted by the transmitter as data for transmission would be formatted 11ØØØddd.

BIT 6	BIT 5	BITS/CHAR
Ø	Ø	5
Ø	1	7
1	Ø	6
1	1	8

TABLE 3.9

Bit 7 The DTR bit controls the DTR* pin. Setting this bit forces DTR* active; resetting it forces DTR* inactive.

3.7 STATUS REGISTER Ø

This register holds various status bits for use during a data transfer operation. Bits 7-3, the External/Status Interrupt bits, are all latched when any one bit changes state and are reset by the RESET EXTERNAL/STATUS INTERRUPTS command (Command Register Ø Bits 5-3 = ØlØ). An External/Status bit thus indicates current state only if no change has occurred in any of the five bits since the last reset.

Received characters are transmitted from the Receiver Register to the bottom of a three-buffer FIFO stack. Each buffer has its own Error Buffer. When a receiver character is read, the contents of the topmost buffer are put on the data bus and the stack pointer decrements. When this occurs, the error status information for the character just read is lost. Therefore, the error status of a received character may be read only BEFORE the character is read.

7	6	5	4	3	2	1	0
BREAK	NOT USED	CTS	RI	DCD	Tx BUFFER EMPTY	INT PENDING	Rx CHAR AVAIL

STATUS REGISTER 0

Bit Ø — This bit is 1 when a character is available in the receiver buffers. Interrupts on this condition are enabled by Command Register 1 Bits 4-3.

Bit 1 — This bit set indicates an INTERRUPT PENDING. It is available in Channel A only; in Channel B this bit is always Ø.

Bit 2 — This bit is set whenever the Transmitter Buffer is empty.

Bit 3 — This bit reflects the state of the DCD* pin until latched as described above.

Bit 4 — This bit reflects the state of the RI* pin until latched as described above.

Bit 5 — This bit reflects the state of the CTS* pin until latched as described above.

Bit 6 — This bit is not used.

Bit 7 — This bit is set when a break is detected. The bit will be reset when the break condition is removed only if a RESET EXTERNAL/ STATUS INTERRUPTS command has been issued.

3.8 STATUS REGISTER 1

This register holds the error and all-sent status bits.

7	6	5	4	3	2	1	0
NOT USED	FRAMING ERROR	Rx OVERRUN ERROR	PARITY ERROR	NOT USED	NOT USED	NOT USED	ALL SENT

STATUS REGISTER 1

Bit Ø — The ALL SENT bit indicates that all characters have completely cleared the transmitter. It does not cause interrupts.

Bits 3-1 These bits are not used.

Bit 4 The latched PARITY ERROR bit is set whenever parity is enabled and a parity error is detected. The bit is reset only by the ERROR RESET command (see Command Register Ø Bits 5-3).

Bit 5 The latched RECEIVER OVERRUN ERROR bit indicates when more than four characters have been received without being read by the CPU. The bit is set when the overwritten character is read and reset by the ERROR RESET command. If the STATUS AFFECTS VECTOR bit (Command Register 1 Bit 2) is set, this bit set will generate a SPECIAL RECEIVE CONDITION interrupt.

Bit 6 When a FRAMING ERROR is detected, this bit is set but not latched.

Bit 7 This bit is not used.

3.9 STATUS REGISTER 2

This register is available in Channel B only and contains the interrupt vector. If the STATUS AFFECTS VECTOR bit is set, bits 3-1 will reflect any pending interrupt; if no interrupts are pending, they will hold Ø11. If the STATUS AFFECTS VECTOR bit is cleared, bits 3-1 will appear as programmed.

7	6	5	4	3	2	1	0
INT VECTOR 7	INT VECTOR 6	INT VECTOR 5	INT VECTOR 4	INT VECTOR 3	INT VECTOR 2	INT VECTOR 1	INT VECTOR Ø

STATUS REGISTER 2

CHAPTER 4

PROGRAMMING THE DMA

The DMA is the most complex of the Z-8Ø peripherals from the programmer's point of view. There are 18 writable registers, which are addressed in one of two ways, as follows:

1. There are six command registers directly accessible at the DMA's single port address. The register that is accessed by a command is determined by the states of bits 7, 1, and Ø of the command; bits 6-2 are used for the various commands themselves.

2. Several registers can be accessed only after a specific bit in one of the six first-level registers is set. These registers are thus in effect "nested" in the primary registers. In addition, some registers are double-nested; i.e., they are accessed next if a bit in a nested register is set. In all there are twelve nested write registers.

   If two or more nested-register-access bits are set in a single register, the nested registers will be accessed by consecutive writes, with the register indicated by the lowest-order bit being accessed first.

There are seven readable registers: a status register and the upper-lower pairs for Port A and Port B addresses and block length. These are accessed sequentially without intervening commands. A Read Mask Register allows the programmer to exclude any register(s) from the read sequence.

The DMA Controller can be programmed for operation in four modes. The four modes are:

Byte Mode: Control is returned to the CPU after each one-byte cycle.

Continuous Mode: Control is returned to the CPU only when the entire block operation is completed.

Burst Mode: Control is returned to the CPU when RDY goes inactive, at end of block, or on match.

Transparent Mode: DMA operation occurs during memory refresh times and therefore does not affect normal CPU operation.

4.1 COMMAND REGISTER Ø

This register is accessed when Bit 7 = Ø and Bits 1-Ø = ØØ.

7	6	5	4	3	2	1	0
0	TIMING BYTE FOLLOWS	FIXED ADDRESS	INC/ DEC ADDRESS	I/O OR MEMORY	PROGRAMS PORT A OR B	0	0

COMMAND REGISTER 0

Bit 2 This command programs Port A if this bit is a 1 and Port B if this bit is a Ø.

Bit 3 The port addressed by this command is an I/O device if this bit is 1 and main memory if this bit is Ø.

Bit 4 If Bit 5 is Ø, the address of the port indicated by Bit 2 will decrement if this bit is Ø and increment if this bit is 1. If Bit 5 is 1, this bit has no effect.

Bit 5 If this bit is 1, the port selected by Bit 2 has a fixed address; if this bit is Ø, the selected port's address increments or decrements according to Bit 4.

Bit 6 If this bit is 1, the next byte is written to the Timing Register. This register need not be programmed if standard Z-8Ø timing is to be followed.

7	6	5	4	3	2	1	0
WR* END EARLY	RD* END EARLY	NOT USED	NOT USED	MREQ* END EARLY	IORQ* END EARLY	TIMING CYCLE 1	TIMING CYCLE Ø

TIMING REGISTER

Bits 1-Ø These bits determine the basic cycle length of a DMA operation according to Table 4.1.

BIT 1	BIT Ø	CYCLES
Ø	Ø	4
Ø	1	3
1	Ø	2
1	1	1

TABLE 4.1

Bits 3-2, Bits 7-6 A Ø in any of these bits will cause the corresponding control signal to end one half clock cycle before the actual operation cycle ends. However, the total operation (Read and Write in Transfer or Read only in Search) must be at least two cycles long.

Bits 5-4 These bits are not used; their states are irrelevant.

4.2 COMMAND REGISTER 1

This register is accessed when Bit 7 = Ø and Bits 1-Ø = Ø1, 1Ø or 11.

7	6	5	4	3	2	1	0
0	BLK LGTH HIGH FOLLOWS	BLK LGTH LOW FOLLOWS	PORT A ADD HIGH FOLLOWS	PORT A ADD LOW FOLLOWS	PORT A OR B SOURCE	DMA OP SELECT 1	DMA OP SELECT Ø

COMMAND REGISTER 1

Bits 1-Ø These bits determine the type of operation to take place, as indicated in Table 4.2.

BIT 1	BIT Ø	OPERATION
Ø	Ø	Selects CRØ
Ø	1	Transfer Only
1	Ø	Search Only
1	1	Search and Transfer

TABLE 4.2

Bit 2 If this bit is 1, Port A is the source and Port B is the destination. If this bit is Ø, Port B is the source and Port A is the destination.

Bit 3 If this bit is 1, the next byte is the low byte of the Port A starting address.

7	6	5	4	3	2	1	0
PORT A ADDRESS 7	PORT A ADDRESS 6	PORT A ADDRESS 5	PORT A ADDRESS 4	PORT A ADDRESS 3	PORT A ADDRESS 2	PORT A ADDRESS 1	PORT A ADDRESS Ø

PORT A ADDRESS REGISTER (LOW)

Bit 4 If this bit is 1, the next byte is the high byte of the Port A starting address.

7	6	5	4	3	2	1	0
PORT A ADDRESS 15	PORT A ADDRESS 14	PORT A ADDRESS 13	PORT A ADDRESS 12	PORT A ADDRESS 11	PORT A ADDRESS 1Ø	PORT A ADDRESS 9	PORT A ADDRESS 8

PORT A ADDRESS REGISTER (HIGH)

Bit 5 If this bit is 1, the next byte is the low byte of the block length. The value in the Block Length Registers is a sixteen-digit binary number ONE LESS than the desired block length (i.e., if the block length is N, the Block Length Registers should hold the value N-1).

7	6	5	4	3	2	1	0
BLOCK LENGTH 7	BLOCK LENGTH 6	BLOCK LENGTH 5	BLOCK LENGTH 4	BLOCK LENGTH 3	BLOCK LENGTH 2	BLOCK LENGTH 1	BLOCK LENGTH Ø

BLOCK LENGTH REGISTER (LOW)

Bit 6 If this bit is 1, the next byte is the high byte of the block length.

7	6	5	4	3	2	1	0
BLOCK LENGTH 15	BLOCK LENGTH 14	BLOCK LENGTH 13	BLOCK LENGTH 12	BLOCK LENGTH 11	BLOCK LENGTH 10	BLOCK LENGTH 9	BLOCK LENGTH 8

BLOCK LENGTH REGISTER (HIGH)

4.3 COMMAND REGISTER 2

This register is accessed when Bit 7 = 1 and Bits 1-Ø = ØØ.

7	6	5	4	3	2	1	0
1	ENABLE CHIP	ENABLE INT	MATCH BYTE FOLLOWS	MASK BYTE FOLLOWS	STOP ON MATCH	0	0

COMMAND REGISTER 2

Bit 2 Setting this bit forces the DMA to stop when in search mode it finds a byte that matches the programmed Match Byte.

Bit 3 Setting this bit indicates that the next byte will be written to the Mask Register. The Mask Byte affects which bits are looked at and which are not in a search operation. If a Mask Register bit contains a Ø, that bit position is compared; if a bit contains a 1, that bit position is ignored.

7	6	5	4	3	2	1	0
MASK BYTE 7	MASK BYTE 6	MASK BYTE 5	MASK BYTE 4	MASK BYTE 3	MASK BYTE 2	MASK BYTE 1	MASK BYTE Ø

MASK REGISTER

Bit 4 Setting this bit indicates that the next byte will be written to the Match Register. The Match Byte is the byte to be compared with read data during a search operation.

7	6	5	4	3	2	1	0
MATCH BYTE 7	MATCH BYTE 6	MATCH BYTE 5	MATCH BYTE 4	MATCH BYTE 3	MATCH BYTE 2	MATCH BYTE 1	MATCH BYTE Ø

MATCH REGISTER

Bit 5 Setting this bit enables interrupts by the DMA.

Bit 6 Setting this bit enables the DMA. Any further command will disable the DMA, so the command in which this bit is set must be the last command preceding a DMA operation.

4.4 COMMAND REGISTER 3

This register is accessed when Bit 7 = 1 and Bits 1-Ø = Ø1.

7	6	5	4	3	2	1	0
1	MODE SELECT 1	MODE SELECT Ø	INT CONTROL FOLLOWS	PORT B ADDR HI FOLLOWS	PORT B ADDR LOW FOLLOWS	0	1

COMMAND REGISTER 3

Bit 2 Setting this bit indicates that the next byte will be written to the low half of the Port B Address Register.

7	6	5	4	3	2	1	0
PORT B ADDR 7	PORT B ADDR 6	PORT B ADDR 5	PORT B ADDR 4	PORT B ADDR 3	PORT B ADDR 2	PORT B ADDR 1	PORT B ADDR Ø

PORT B ADDRESS REGISTER (LOW)

Bit 3 Setting this bit indicates that the next byte will be written to the high half of the Port B Address Register.

7	6	5	4	3	2	1	0
PORT B ADDR 15	PORT B ADDR 14	PORT B ADDR 13	PORT B ADDR 12	PORT B ADDR 11	PORT B ADDR 10	PORT B ADDR 9	PORT B ADDR 8

PORT B ADDRESS REGISTER (HIGH)

Bit 4 Setting this bit indicates that the next byte will be written to the INTERRUPT CONTROL REGISTER. The format of the Interrupt Control Byte is as follows.

7	6	5	4	3	2	1	0
NOT USED	INT BEFORE BUS RQ	STATUS AFFECTS VECTOR	INT VECTOR FOLLOWS	PULSE OFFSET FOLLOWS	PULSE ENABLE	INT ON END OF BLOCK	INT ON MATCH FOUND

INTERRUPT CONTROL REGISTER

Bit 0 Setting this bit causes an interrupt at the end of a block transfer.

Bit 1 Setting this bit causes an interrupt when a byte is found which matches the Match Byte as modified by the Mask Byte. This interrupt will occur only during search operations.

Bit 2 Setting this bit causes a pulse to be generated on the INT*/PULSE* pin every 256 bytes, allowing the peripheral to keep track of how many bytes have been transferred.

Bit 3 Setting this bit indicates that the next byte will be written to the Pulse Count Register. This register is loaded with a binary number which is used to offset the beginning of the pulse generation from 1 to 255 bytes.

7	6	5	4	3	2	1	0
PULSE OFFSET 7	PULSE OFFSET 6	PULSE OFFSET 5	PULSE OFFSET 4	PULSE OFFSET 3	PULSE OFFSET 2	PULSE OFFSET 1	PULSE OFFSET 0

PULSE OFFSET REGISTER

Bit 4 Setting this byte indicates that the next byte will be written to the Interrupt Vector Register. The interrupt vector is released to the data bus when the interrupting DMA receives an interrupt acknowledge. Bit Ø is always returned as a Ø; Bits 2-1 vary if the STATUS AFFECTS VECTOR bit is 1.

7	6	5	4	3	2	1	0
INT VECTOR 7	INT VECTOR 6	INT VECTOR 5	INT VECTOR 4	INT VECTOR 3	INT VECTOR 2	INT VECTOR 1	INT VECTOR Ø

INTERRUPT VECTOR REGISTER

Bit 5 This is the STATUS AFFECTS VECTOR bit. When set, the Interrupt Vector Bits 2-1 are modified by the interrupt condition, as indicated by Table 4.3.

IV2	IV1	CONDITION
Ø	Ø	Interrupt on RDY
Ø	1	Match
1	Ø	End of Block
1	1	Match, End of Block

TABLE 4.3

Bit 6 Setting this bit causes the DMA to interrupt when its RDY input is asserted before it requests control of the bus.

Bit 7 This bit has no effect.

Bits 6-5 These bits select the mode of DMA operation as indicated by Table 4.4.

BIT 6	BIT 5	MODE
Ø	Ø	Byte
Ø	1	Continuous
1	Ø	Burst
1	1	Transparent

TABLE 4.4

4.5 COMMAND REGISTER 4

This register is accessed when Bit 7 = 1 and Bits 1-Ø = 1Ø.

7	6	5	4	3	2	1	0
1	NOT USED	AUTO REPEAT	WAIT ENABLE	RDY HIGH OR LOW	NOT USED	1	0

COMMAND REGISTER 4

Bit 2 This bit has no effect.

Bit 3 Setting this bit makes RDY active high; clearing it makes RDY active low.

Bit 4 Setting this bit allows the CE* pin to be used as a WAIT* input controlled by memory or I/O devices. This option is not implemented on the 282Ø

Bit 5 Setting this bit causes an operation to automatically repeat after the end of the block.

Bit 6 This bit has no effect.

4.6 COMMAND REGISTER 5

This register is accessed when Bit 7 = 1 and Bits 1-Ø = 11. The five variable bits of this register select one of 16 commands, as indicated in Table 4.5. Unused codes have no effect. Descriptions of the commands follow the table.

7	6	5	4	3	2	1	0
1	COMMAND BIT 4	COMMAND BIT 3	COMMAND BIT 2	COMMAND BIT 1	COMMAND BIT Ø	1	1

COMMAND REGISTER 5

HEX	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	COMMAND
83	Ø	Ø	Ø	Ø	Ø	Disable DMA
87	Ø	Ø	Ø	Ø	1	Enable DMA
8B	Ø	Ø	Ø	1	Ø	Reset Status
A3	Ø	1	Ø	Ø	Ø	Reset Interrupt
A7	Ø	1	Ø	Ø	1	Reset Read Sequence
AB	Ø	1	Ø	1	Ø	Enable Interrupts
AF	Ø	1	Ø	1	1	Disable Interrupts
B3	Ø	1	1	Ø	Ø	Force RDY
B7	Ø	1	1	Ø	1	Enable After RETI
BB	Ø	1	1	1	Ø	Read Byte Follows
BF	Ø	1	1	1	1	Read Status
C3	1	Ø	Ø	Ø	Ø	Reset
C7	1	Ø	Ø	Ø	1	Reset Port A Timing
CB	1	Ø	Ø	1	Ø	Reset Port B Timing
CF	1	Ø	Ø	1	1	Load
D3	1	Ø	1	Ø	Ø	Continue

TABLE 4.5

The ENABLE and DISABLE DMA commands enable and disable DMA operations but do not reset any functions.

The RESET STATUS command resets the END OF BLOCK and MATCH bits of the Status Register.

The RESET INTERRUPT command resets and disbales all interrupt circuitry.

The RESET READ command causes the next read to be of the first register whose corresponding bit position in the Read Mask Register holds a 1 (usually the Status Register).

The ENABLE INTERRUPT command allows interrupts to occur on programmed conditions.

The DISABLE INTERRUPT command prevents interrupt requests from being generated.

The FORCE RDY command allows the RDY input to be simulated under software control. This is necessary in memory-to-memory transfers where the RDY signal is not needed and may be used for other kinds of transfers as deemed appropriate by the programmer.

The ENABLE AFTER RETI command forces the DMA to wait until it has received a RETI to request control of the bus. This command must be followed by an ENABLE DMA command.

The READ BYTE FOLLOWS command causes the next byte to be written to the READ MASK REGISTER. The seven readable DMA registers are read sequentially from low-order to high-order as illustrated below, but a Ø in any position of the Read Mask Register causes the corresponding register to be skipped.

7	6	5	4	3	2	1	0
0	PORT B ADDRESS HIGH	PORT B ADDRESS LOW	PORT A ADDRESS HIGH	PORT A ADDRESS LOW	BLOCK LENGTH HIGH	BLOCK LENGTH LOW	STATUS

READ MASK REGISTER

The READ STATUS command causes the next read to be of the Status Register.

The RESET command resets all interrupt circuitry and disables interrupt and bus request logic.

The RESET TIMING commands reset the Timing Control Registers for Ports A and B to the standard Z-8Ø CPU timing.

The LOAD command resets the byte counter and loads the starting addresses for both ports.

The CONTINUE command resets the byte counter but does not affect the addresses.

4.7 STATUS REGISTER

This is the highest-priority readable register and will be the first register read unless it is masked by Bit Ø of the Read Mask Register. Its format is as follows:

7	6	5	4	3	2	1	0
NOT USED	NOT USED	END OF BLOCK	MATCH FOUND	INT PENDING	NOT USED	READY	DMA OP SINCE LOAD

STATUS REGISTER

Bit Ø — When this bit is 1, the programmed DMA operation has occurred.

Bit 1 — When this bit is Ø, the RDY input is active. This bit is also Ø after a FORCE RDY command.

Bit 2 — This bit is not used.

Bit 3 — When this bit is Ø the DMA has an interrupt request pending.

Bit 4 — This bit is Ø when in search mode a match has been found.

Bit 5 — This bit is Ø when the end of a programmed block is reached.

Bits 7-6 — These bits are not used.

4.8 OTHER READABLE REGISTERS

The other six DMA registers which may be read by the CPU include the PORT A and PORT B ADDRESS REGISTER pairs and the BLOCK LENGTH pair. These are read exactly as written in the order specified by the Read Mask Register.

CHAPTER 5

PROGRAMMING THE CTC

The CTC has nine accessible registers: Channel Control and Time Constant Registers for each of the four channels and an Interrupt Vector Register. The Channel Control Registers are write-only; the Time Constant and Interrupt Vector Registers can be written to or read. Address bits A1 and AØ select the channel, as shown in Table 5.1.

A1	AØ	SELECTS
Ø	Ø	Channel Ø
Ø	1	Channel 1
1	Ø	Channel 2
1	1	Channel 3

TABLE 5.1

The CTC command registers will normally be accessed in the following order: Interrupt Vector Register, Channel Control Register, Time Constant Register. The Interrupt Vector, identified by Command Bit 1 = Ø, must be written to Channel Ø only; the result of a write to one of Channels 1-3 with Bit 3 = Ø is undefined. When Command Bit 1 = 1, the Channel Control Register is accessed first; the next register accessed by a write to any CTC channel (if Command Bit Ø is Ø) is determined by Command Bit 1. If Bit 1 = Ø, the Channel Command Register is accessed again; if Bit 1 = 1, the Time Constant Register is accessed. Operation begins when the Time Constant Register is loaded.

Each channel of the CTC is separately programmed to operate in one of two modes: Timer Mode and Counter Mode. In the Timer Mode, the CLK/TRG input may or may not be used; if used, it functions solely to trigger operation, and has no affect once the timer is operating. The downcounter, loaded with the value in the Time Constant Register, decrements with every 16 or 256 cycles of the system clock,

depending on the prescaler selected. In Counter Mode, the CLK/TRG input is controlled by a clocking signal related to the events to be counted, the downcounter decrementing with every pulse of the CLK/TRG input. In either mode, each time the downcounter reaches zero the ZC/TO output (Channels Ø-2 only) pulses high, an interrupt is generated if enabled, and the downcounter is reloaded with the value in the Time Constant Register. Contents of the downcounter can be read at the channel address.

5.1 INTERRUPT VECTOR REGISTER

Bits 7-3 of this register are programmed by a write to the register. Bit Ø is always returned as a Ø. Bits 2-1 depend on the channel generating the interrupt request, as indicated in Table 5.2.

7	6	5	4	3	2	1	0
INT VECTOR 7	INT VECTOR 6	INT VECTOR 5	INT VECTOR 4	INT VECTOR 3	INT VECTOR 2	INT VECTOR 1	0

INTERRUPT VECTOR REGISTER

IV2	IV1	CHANNEL
Ø	Ø	Ø: 1st Priority
Ø	1	1: 2nd Priority
1	Ø	2: 3rd Priority
1	1	3: 4th Priority

TABLE 5.2

5.2 CHANNEL CONTROL REGISTERS

These registers are separately programmed for each channel. They are formatted as shown below.

7	6	5	4	3	2	1	0
ENABLE INT	TIMER/ COUNTER MODE	PRE-SCALER 16 OR 256	CLK/TRG EDGE SELECT	TIMER TRIGGER SELECT	TIME CONSTANT FOLLOWS	CHANNEL RESET	1

COMMAND REGISTER

Bit Ø — This bit must be a 1 to indicate that a Channel Control Register, and not the Interrupt Vector Register, is being accessed.

Bit 1 — Setting this bit resets the addressed channel, causing it to stop counting/timing. The channel begins counting/timing again when a new Time Constant is reloaded.

Bit 2 — Setting this bit causes the next write to the addressed channel to be to the Time Constant Register.

Bit 3 — This bit affects the Timer Mode only. If it is set, an external trigger is needed at the channel's CLK/TGR input to start the timing operation; if it is cleared, timing begins on the rising edge of T2 of the cycle after the loading of the Time Constant Register.

Bit 4 — This bit selects the edge to which the channel's CLK/TGR input is sensitive: 1 = positive edge, Ø = negative edge.

Bit 5 — This bit affects the Timer Mode only. When it is set, the prescaler is 256; when it is cleared, the prescaler is 16.

Bit 6 — This bit selects the mode in which the addressed channel wil operate: 1 = Counter Mode, Ø = Timer Mode.

Bit 7 — Setting this bit enables interrupts by the channel every time its downcounter reaches zero. If this bit is cleared, interrupts are not generated and pending interrupts are cleared.

5.3 TIME CONSTANT REGISTERS

A separate time constant is programmed for each channel. The Time Constant may be any value from 1 to 256 (ØØH is interpreted as 256). If a channel is not running when a Time Constant is written into it, it will begin timing or counting when the Time Constant Register is loaded.

7	6	5	4	3	2	1	0
TIME CONSTANT 7	TIME CONSTANT 6	TIME CONSTANT 5	TIME CONSTANT 4	TIME CONSTANT 3	TIME CONSTANT 2	TIME CONSTANT 1	TIME CONSTANT Ø

TIME CONSTANT REGISTER

5.4 PROGRAMMING BAUD RATES

The CTC is often used to select the baud rates for serial data transfers; this is true of the CCS 2820 and 2830 boards. Table 5.3 shows how to program the Prescaler and Time Constant, as well as the SIO/DART Data Rate, for common baud rates. Calculations are based on a 4 MHz system clock.

TABLE 5.3. PROGRAMMING CTC AND SIO/DART FOR SELECTED BAUD RATES

	TIME CONSTANTS FOR VARIOUS PRESCALERS AND DATA RATES							
BAUD RATE	P=16 DR=1	P=16 DR=16*	P=16 DR=32*	P=16 DR=64*	P=256 DR=1	P=256 DR=16*	P=256 DR=32*	P=256 DR=64*
50			156	78				
75		208	104	52	208	13		
110		142	71		142			
134.5		116	58	29	116			
150		104	52	26	104			
300		52	26	13	52			
600		26	13		26			
1200	208	13			13			
1800	139							
2000	125							
2400	104							
3600	69							
4800	52							
7200	35							
9600	26							
19200	13							

P Prescaler programmed for CTC.
DR Data Rate programmed for SIO, DART.
* Available for SIO in Async Mode only.

Errors are less than 1% for all combinations shown.

CHAPTER 6

PROGRAMMING THE PIO

The PIO has two separately programmable data channels occupying four port addresses. Each data channel consists of eigth bits of parallel data and two handshaking signals, Ready and Strobe. Address bits A1 and AØ determine the channel addressed and whether a command/status or data buffer will be accessed as indicated in Table 6.1.

A1	AØ	FUNCTION
Ø	Ø	Ch. A: Data
Ø	1	Ch. A: Command
1	Ø	Ch. B: Data
1	1	Ch. B: Command

TABLE 6.1

The PIO is capable of operating in four different modes: Output (Mode Ø), Input (Mode 1), Bi-directional (Mode 2), and Control (Mode 3). Either channel may be programmed for the input or output mode, in which data is passed in one direction or the other and both handshake lines are used. In the bi-directional mode, available for Port A only, the data lines of Port A are bi-directional and the handshake lines for both channels are used; Port B must therefore be programmed for Mode 3. In Mode 3 the handshaking lines are not used and each data bit is individually programmed for input or output. Input bits, which are generally used for status purposes, can be programmed to interrupt either when one or when all become active.

Each channel has six Command Registers, two of which are nested in a first-level register. Unlike the SIO channels, the PIO channels have independent Interrupt Vector Registers. An Interrupt Vector Register is accessed whenever Bit Ø of a command is Ø. Command Register Ø, 1, or

2 is accessed, according to the settings of Bits 3-1, when Bit Ø is 1. The Mode 3 Control Register is nested in Command Register Ø, and the Mask Register is nested in Command Register 1.

6.1 INTERRUPT VECTOR REGISTER

If bit Ø is a Ø, a command is identified as the interrupt vector. No vector bits are affected by status; the Interrupt Vector is always put on the bus exactly as programmed.

7	6	5	4	3	2	1	0
INT VECTOR 7	INT VECTOR 6	INT VECTOR 5	INT VECTOR 4	INT VECTOR 3	INT VECTOR 2	INT VECTOR 1	0

INTERRUPT VECTOR REGISTER

6.2 COMMAND REGISTER Ø

This register is accessed when command bits 3-Ø = 1 and is used to select the mode in which the addressed channel will operate.

7	6	5	4	3	2	1	0
MODE SELECT 1	MODE SELECT Ø	NOT USED	NOT USED	1	1	1	1

COMMAND REGISTER 0

Bits 5-4 These bits are not used.

Bits 7-6 These bits are used to select one of the four modes, as indicated in Table 6.2.

BIT 7	BIT 6	MODE
Ø	Ø	Ø: Output
Ø	1	1: Input
1	Ø	2: Bi-directional
1	1	3: Control

TABLE 6.2

If Mode 3 is selected by Command Register Ø Bits 7-6, the next byte written selects which bits are inputs and which are outputs. Writing a 1 to a bit of the Mode 3 Control Register programs the corresponding data bit as input; writing a Ø to a bit programs the corresponding data bit as output.

7	6	5	4	3	2	1	0
BIT 7 IN OR OUT	BIT 6 IN OR OUT	BIT 5 IN OR OUT	BIT 4 IN OR OUT	BIT 3 IN OR OUT	BIT 2 IN OR OUT	BIT 1 IN OR OUT	BIT Ø IN OR OUT

MODE 3 CONTROL REGISTER

6.3 COMMAND REGISTER 1

This register enables/disables interrupts and, in mode 3, determines the interrupt conditions. It is accessed when command bits Ø-2 = 1 and bit 3 = Ø.

7	6	5	4	3	2	1	0
ENABLE INT	MODE 3 AND/ OR	MODE 3 HIGH/ LOW	MODE 3 MASK FOLLOWS	0	1	1	1

COMMAND REGISTER 1

Bit 4 If this bit is set, the next write will be to the Mask Register. Setting a Mask Register bit to 1 disables interrupts by that bit in Mode 3.

7	6	5	4	3	2	1	0
MASK BYTE 7	MASK BYTE 6	MASK BYTE 5	MASK BYTE 4	MASK BYTE 3	MASK BYTE 2	MASK BYTE 1	MASK BYTE Ø

MASK REGISTER

Bit 5 In Mode 3, this bit selects whether bits will be considered active high (Bit 5 = 1) or active low (Bit 5 = Ø).

Bit 6 In Mode 3, if this bit is a 1, all unmasked bits must be active (AND condition) for an interrupt to be generated. If this bit is a Ø, an interrupt will occur when any one bit goes active (OR condition).

Bit 7 Setting this bit to 1 enables interrupts.

6.4 COMMAND REGISTER 2

This register, accessed when command bits 1-Ø = 1 and bits 3-2 = Ø, provides for interrupt enabling without modification of the Command Register 2 Mode 3 control bits.

7	6	5	4	3	2	1	0
ENABLE INT	NOT USED	NOT USED	NOT USED	0	0	1	1

COMMAND REGISTER 2

Bits 6-4 These bits are not used.

Bit 7 Setting this bit to 1 enables interrupts; writing a Ø to this bit disables interrupts.