Z80-SIO™

Product Specification
Preliminary

MARCH 1978

The Zilog Z80 product line is a complete set of microcomputer components, development systems and support software. The Z80 microcomputer component set includes all of the circuits necessary to build high-performance microcomputer systems with virtually no other logic and a minimum number of low cost standard memory elements.

The Z80-SIO (Serial Input/Output) circuit is a programmable, dual-channel device which provides formatting of data for serial data communication. It is capable of handling asynchronous, synchronous and synchronous bit oriented protocols such as IBM BiSync, HDLC, SDLC and virtually any other serial protocol. It can generate CRC codes in any synchronous mode and can be programmed by the CPU for any traditional asynchronous format.

Structure

- N-channel Silicon Gate Depletion Load Technology
- Forty Pin DIP
- Single 5 volt power supply
- Single phase 5 volt clock
- Two Full Duplex channels

Features

- Two independent full duplex channels
- Data rates – 0 to 550K bits/second
- Receiver data registers quadruply buffered; transmitter doubly buffered.
- Asynchronous operation
  - 5, 6, 7 or 8 bits/character
  - 1, 1½ or 2 stop bits
  - Even, odd or no parity
  - x1, x16, x32 and x 64 clock modes
  - Break generation and detection
  - Parity, Overrun and Framing error detection
- Binary Synchronous operation
  - Internal or external character synchronozation
  - One or two Sync characters in separate registers
  - Automatic Sync Character Insertion
  - CRC generation and checking
- HDLC or IBM SDLC operation
  - Automatic Zero insertion and deletion
  - Automatic Flag insertion
  - Address field recognition
  - I-field residue handling
  - Valid receive messages protected from overrun
  - CRC generation and checking
- Eight modem control inputs and outputs
- Both CRC-16 and CRC-CCITT (-0 and -1) are implemented
- Daisy chain priority interrupt logic included to provide for automatic interrupt vectoring without external logic.
- All inputs and outputs fully TTL compatible.

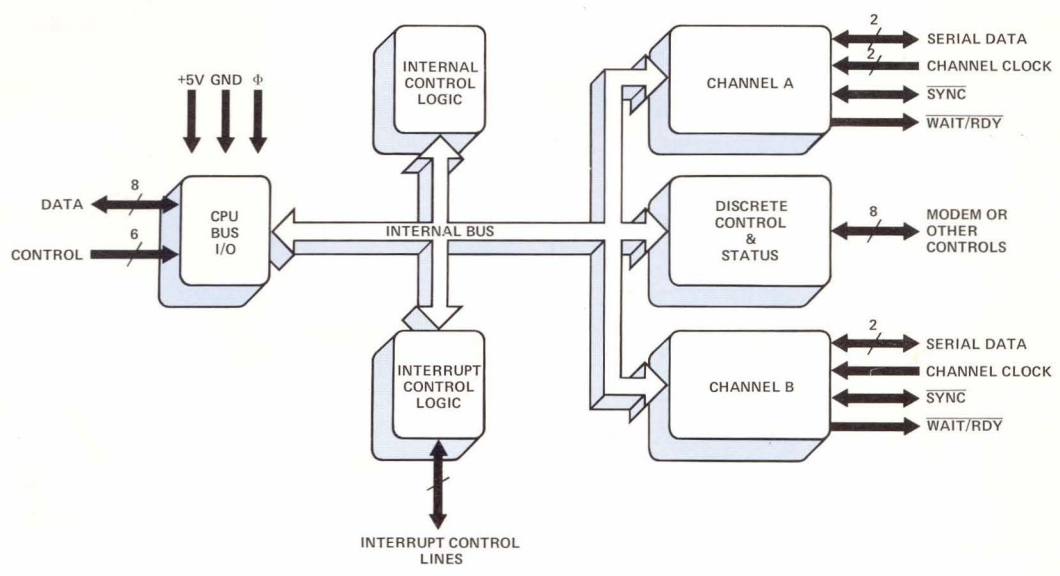


FIGURE 1
SIO BLOCK DIAGRAM

TM: Z80 is a trademark of Zilog, Inc.

SIO Architecture

A block diagram of the SIO is shown in Figure 1. The internal structure includes a Z80-CPU bus interface, internal control and interrupt logic and two full duplex channels. The interrupt control logic determines which channel and which device within the channel is the highest priority for purposes of the automatic interrupt vectoring. Priority is fixed with Channel A assigned higher priority than Channel B and the Receiver, Transmitter and External/Status assigned priority in that order within each channel.

The channel logic is shown in block form in Figure 2. Each channel has five 8-bit control registers, two 8-bit status registers and two 8-bit sync character registers. The interrupt vector is written into an additional 8-bit register in Channel B and may also be read thru that channel. The receiver has three 8-bit buffer registers in FIFO arrangement in addition to the 8-bit input shift register. The transmitter has one 8-bit buffer register in addition to the 8-bit output shift register. The CRC generator/checkers are 16-bit shift registers with appropriate internal feedback (programmable) for two different CRC codes.

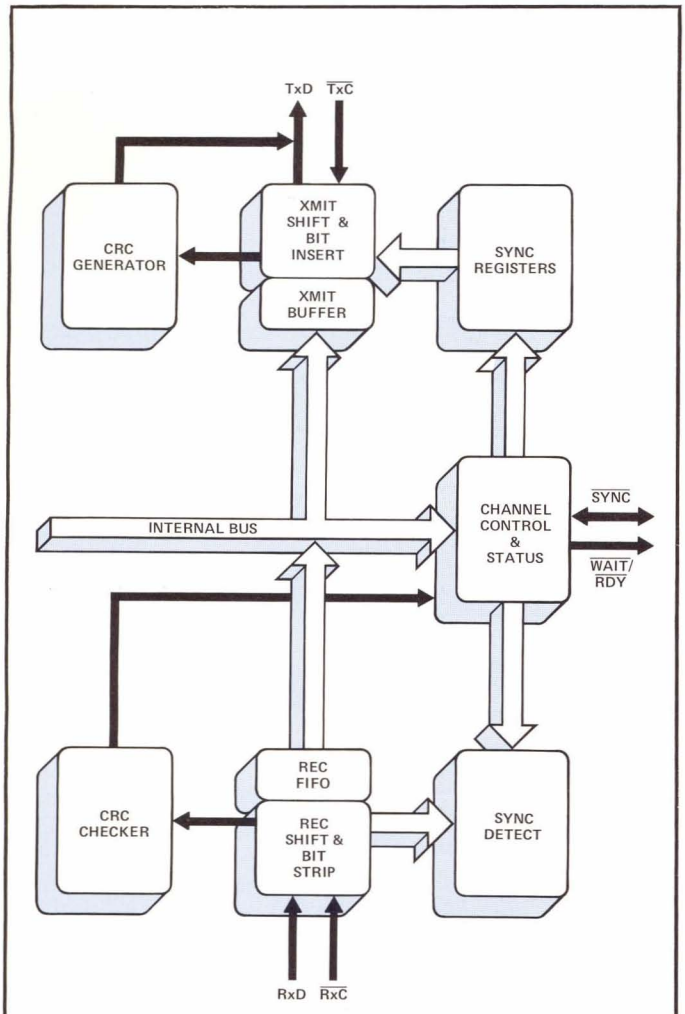
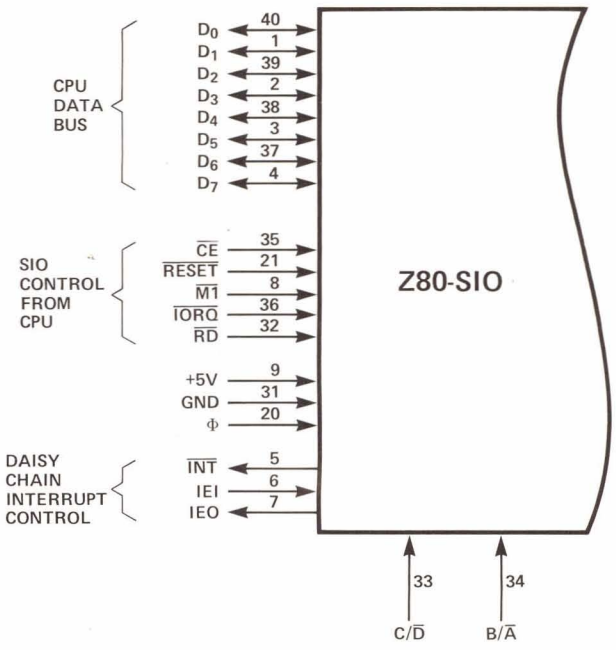


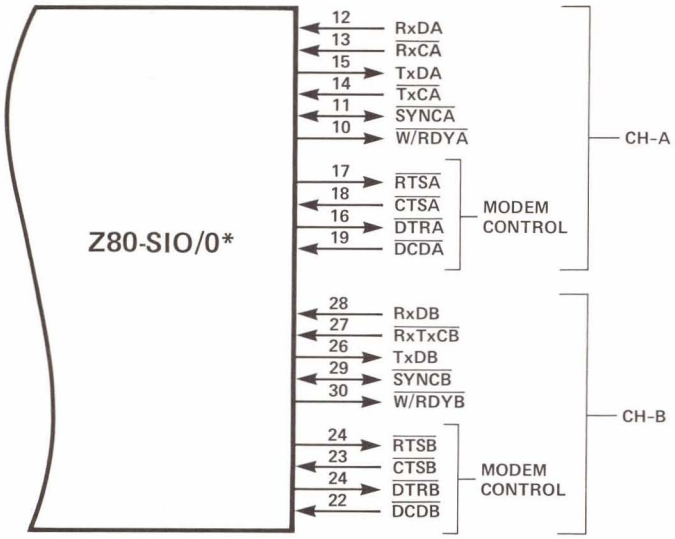
FIGURE 2
CHANNEL BLOCK DIAGRAM

Z80-SIO Pin Description



D_0–D_7	System Data Bus (bidirectional, tri-state)
B/$\overline{A}$	Channel B or A select (input high is Channel B)
C/$\overline{D}$	Control or Data select (input high is control)
$\overline{CE}$	Chip Enable (input, active low)
$\overline{M1}$	Machine Cycle One Signal from Z80-CPU (input, active low)
$\overline{IORQ}$	Input/Output request from Z80-CPU (input, active low)
$\overline{RD}$	Read Cycle Status from the Z80-CPU (input, active low)
Φ	System Clock (input)
$\overline{RESET}$	Reset (input, active low) disables both receivers and transmitters. TxDA and TxDB are forced marking. Modem controls are forced high. Control registers must be rewritten after SIO is reset and before any data is transmitted or received. All interrupts are disabled.
IEI	Interrupt Enable In (input, active high)
IEO	Interrupt Enable Out (output, active high) IEI and IEO form a daisy-chain connection for priority interrupt control.
$\overline{INT}$	Interrupt Request (output, open drain, active low).

Z80-SIO Pin Description



*See note below on bonding option.

$\overline{\text{WAIT/READY}}$ A
$\overline{\text{WAIT/READY}}$ B
Two pins, one for each channel. They may be programmed to serve as ready lines for use with a DMA Controller or they may serve as wait lines to synchronize the Z80-CPU to the SIO data rate.

$\overline{\text{CTSA}}$, $\overline{\text{CTSB}}$
Clear to Send (2 pins, inputs, active low). When programmed as "auto enables," these inputs inhibit the transmitters of their respective channels. If these pins are not programmed as transmitter enables, they may be programmed as general-purpose input pins. These inputs are Schmitt-trigger buffered to allow slow-risetime inputs.

$\overline{\text{DCDA}}$, $\overline{\text{DCDB}}$
Data Carrier Detect (2 pins, inputs, active low.) These pins are similar to the $\overline{\text{CTS}}$ inputs, except that they are usable as receiver inhibits rather than transmitter inhibits.

RxDA, RxDB
Receive Data. (2 pins, inputs, active high.)

TxDA, TxDB
Transmit Data. (2 pins, outputs, active high.)

*$\overline{\text{RxCA}}$, $\overline{\text{RxCB}}$
Receiver Clocks (inputs, active low.) (Two pads, one per channel. See note on Bonding Option.) Clock may be x1, x16, x32 or x64 the data rate in asynchronous modes.

*$\overline{\text{TxCA}}$, $\overline{\text{TxCB}}$
Transmitter Clocks (inputs, active high.) (Two pads, one per channel. See note on Bonding Option.) May be x1, x16, x32 or x64 baud rate, but same multiplier must be observed as for receiver. The TxC and RxC inputs are Schmitt-trigger buffered, for relaxed rise and fall time requirements.

*These clocks can be directly driven by the Z80-CTC (Counter Timer Circuit) for fully programmable baud rate generation.

$\overline{\text{RTSA}}$, $\overline{\text{RTSB}}$
Request to Send (2 pins, outputs, active low.) When the $\overline{\text{RTS}}$ bit is set, the $\overline{\text{RTS}}$ pin goes low. When the bit is reset in asynchronous mode, the pin goes high, but only after the transmitter is empty. In synchronous modes, $\overline{\text{RTS}}$ is a simple output which strictly follows the state of the $\overline{\text{RTS}}$ bit.

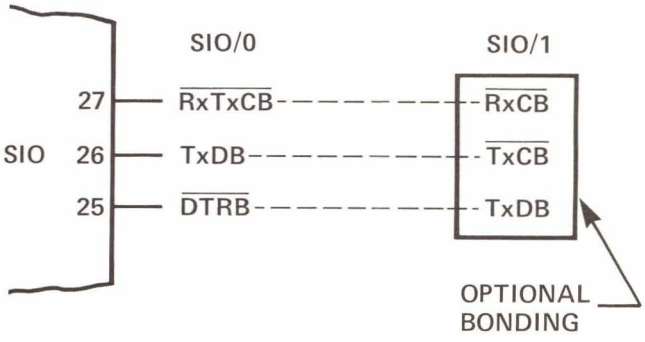
$\overline{\text{DTRA}}$, $\overline{\text{DTRB}}$
Data Terminal Ready (2 pins, output, active low.) Pin follows state programmed with DTR bit. (Two pads, one per channel. See note on Bonding Option.)

$\overline{\text{SYNCA}}$, $\overline{\text{SYNCB}}$
External Character Synchronization (2 pins, input/output, active low.) If the External Synchronization mode is selected, assembly of characters will begin on the next rising edge of $\overline{\text{RxC}}$. If internal character sync modes are selected, the pins are outputs that are active during part of the clock cycles that a sync character is recognized. The sync condition is not latched, so this pin will be active every time a sync pattern is recognized, regardless of character boundaries. In asynchronous modes, these pins are simple inputs to the Hunt/Sync bits in Status Register 0 and may be used for any input function desired.

NOTE: When used as an external synchronization pin, it must not become active for three system clock cycles after the previous rising edge of $\overline{\text{RxC}}$. This requirement normally can be met by allowing $\overline{\text{SYNC}}$ to change only on the falling edge of $\overline{\text{RxC}}$.

Note on Bonding Option:

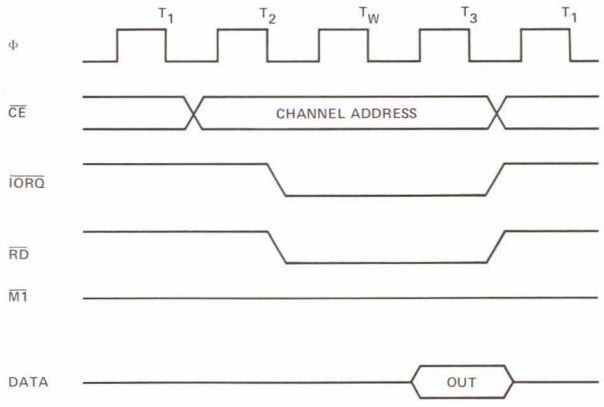
Due to package constraints, there are only two pins available for the three signals, $\overline{\text{TxCB}}$, $\overline{\text{RxCB}}$ and $\overline{\text{DTRB}}$. They are normally bonded so that $\overline{\text{TxCB}}$ and $\overline{\text{RxCB}}$ are one pin, and $\overline{\text{RxTxCB}}$ and $\overline{\text{DTRB}}$ is an available output. If there is a requirement for different clock rates or phases for $\overline{\text{RxCB}}$ and $\overline{\text{TxCB}}$, they may be bonded independently by sacrificing $\overline{\text{DTRB}}$.



SIO Timing Waveforms

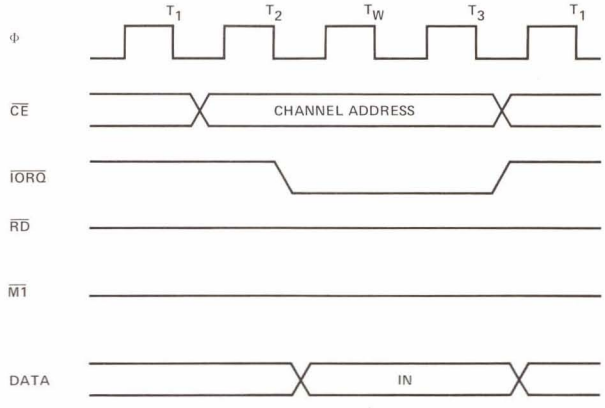
READ CYCLE

The timing associated with reading data or a status register within the SIO is illustrated here. Z80 Input instructions satisfy this timing.



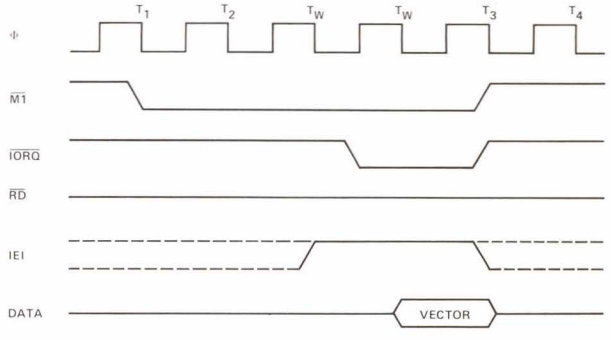
WRITE CYCLE

Illustrated here is the timing associated with a data or control byte being written into the SIO. Z80 Output Instructions satisfy this timing.



INTERRUPT ACKNOWLEDGE CYCLE

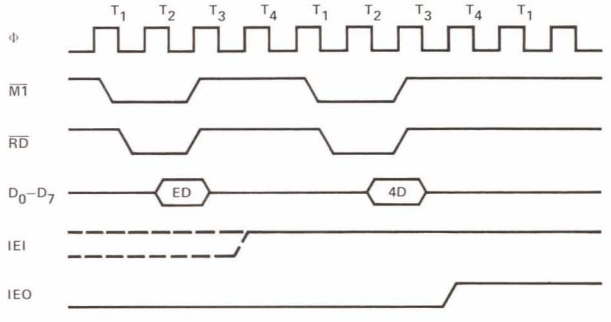
Some time after an interrupt is requested by the SIO, the CPU will send out an interrupt acknowledge ($\overline{M1}$ and $\overline{IORQ}$.) During this time, the interrupt logic of the SIO will determine the highest priority function which is requesting an interrupt. To insure that the daisy chain enable lines stabilize, channels are inhibited from changing their interrupt request status when $\overline{M1}$ is active (low). If the SIO is the highest priority device requesting an interrupt, the SIO will place the appropriate interrupt vector on the data bus when $\overline{IORQ}$ goes active.



RETURN FROM INTERRUPT CYCLE

If a Z80 peripheral device has no interrupt pending and is not under service, then its IEO = IEI. If it has an interrupt under service (i.e. it has already interrupted and received an interrupt acknowledge) then its IEO is always low, inhibiting lower priority chips from interrupting. If it has an interrupt pending which has not yet been acknowledged, IEO will be low unless an "ED" is decoded as the first byte of a two byte opcode. In this case, IEO will go high until the next opcode byte is decoded, whereupon it will again go low. If the second byte of the opcode was a "4D" then the opcode was an RETI instruction.

After an "ED" opcode is decoded, only the peripheral device which has interrupted and is currently under service will have its IEI high and its IEO low. This device is the highest priority device in the daisy chain which has received an interrupt acknowledge. All other peripherals have



IEI = IEO. If the next opcode byte decoded is "4D", this peripheral device will reset its "interrupt under service" condition.

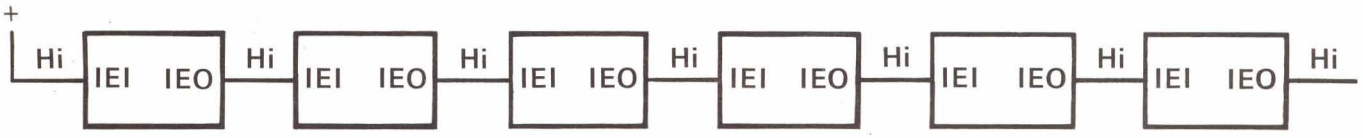
Wait cycles are allowed in the $\overline{M1}$ cycles, but cannot be used to extend high to low daisy chain ripple time. Wait cycles however, may be used for low to high daisy chain ripple during 4D.

Daisy Chain Interrupt Servicing

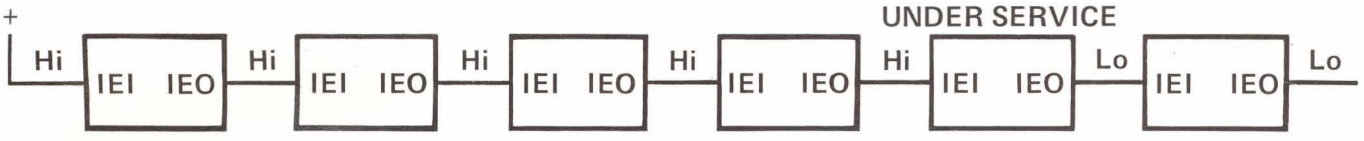
The following illustration is a typical nested interrupt sequence which may occur in the SIO. In a system with several peripheral chips, the other chips may be included in the daisy chain with either higher or lower priority than the SIO channels.

In this sequence, the transmitter of Channel B interrupts and is granted service. While it is being serviced, an external/status interrupt from Channel A occurs and is granted service. The service routine for the Channel A interrupt is completed and either the RETI instruction is executed or the RETI command is written into the SIO to indicate to Channel A that the external/status interrupt routine is complete. At this time, the service routine for the Channel B transmitter is resumed. When this routine is completed, another RETI instruction is executed to complete the service.

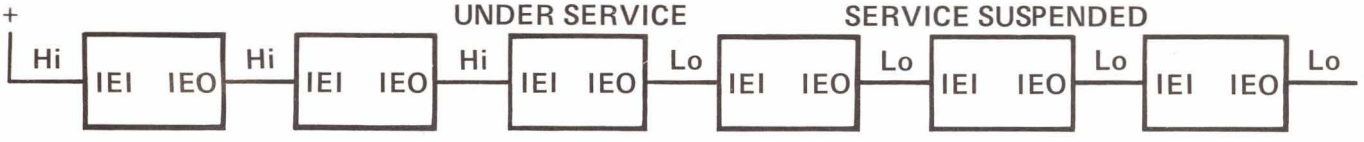
CHANNEL A RECEIVER	CHANNEL A TRANSMITTER	CHANNEL A EXTERNAL/ STATUS	CHANNEL B RECEIVER	CHANNEL B TRANSMITTER	CHANNEL B EXTERNAL/ STATUS



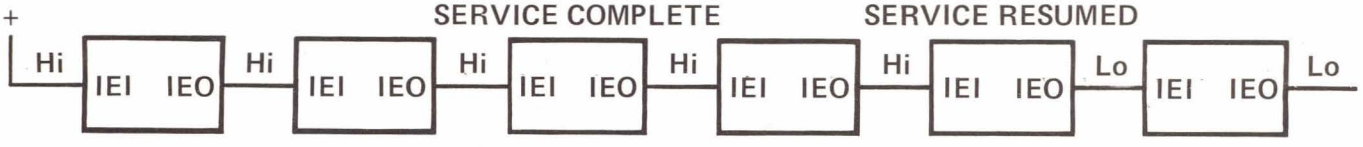
1. Priority Interrupt Daisy Chain before any interrupt occurs.



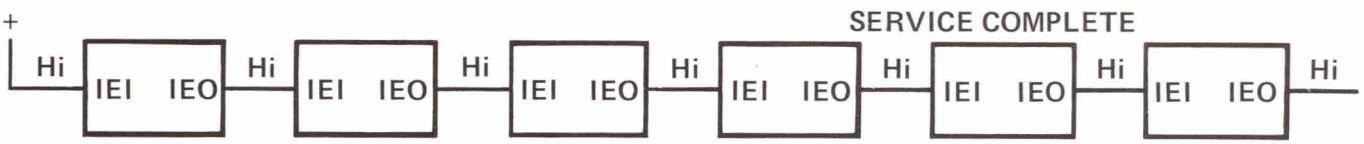
2. Channel B's transmitter interrupts and is acknowledged.



3. External/Status of Channel A interrupts suspending service of Channel B transmitter



4. Channel A External/Status routine complete. RETI issued, Channel B transmitter service resumed.



5. Channel B transmitter's service routine complete, second RETI issued.

Operation of the SIO is determined by the contents of the control registers. These must be programmed before any operations can be performed by the SIO. Some commands and modes may be changed during operation. The device status registers can be read at any time.

ASYNCHRONOUS MODES

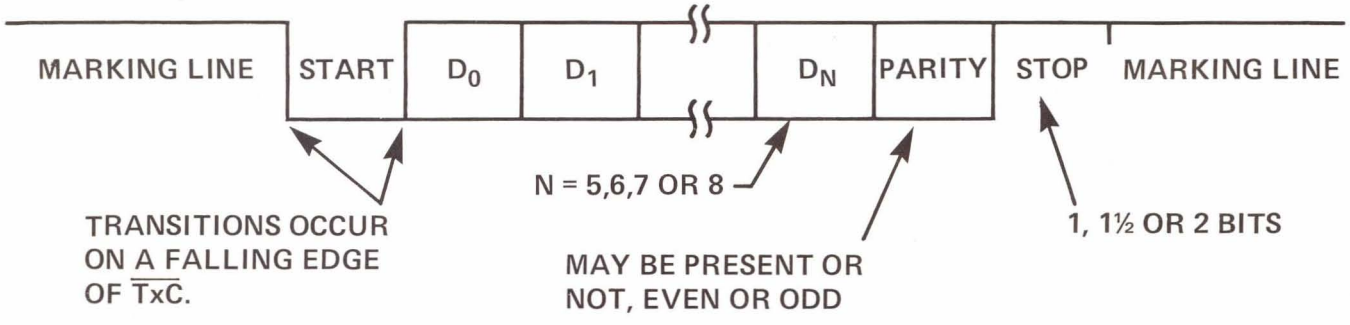
The receiver ports are quadruply buffered, i.e. there are three storage registers in addition to the input shift register. This allows additional time for the CPU to service an interrupt at the beginning of a block of high-speed data transfer. The error flags are also quadruply buffered and are loaded at the same time as the character. The Receiver Overrun and Parity Error flags are not reset unless an Error Reset, (latches) Command (Command 6) is issued. End of Frame and CRC/Framing error reflects the state of the character currently in the buffer unless reset by error reset. Thus, when the error status is read, it will reflect an error in the current word in the receive buffer in addition to any parity or overrun errors received since the last Error Reset, (latches) Command. In order to keep correspondence between the state of the error buffer and the contents of the receive registers, the status register should be read before the data (see exception). This is easily accomplished if the vectored interrupts are used since a special interrupt vector is generated for errors or end of frame.

If the status is read after the data is read, the error data for the next data word will also be included if it has been stacked in the buffer. If operations are being performed rapidly enough so that the next character will not yet be received, then the status register will remain valid. The exception occurs when the "Receive Interrupt on First Character Only" mode is selected. A special interrupt in this mode will hold error data and the character itself (even if read from the buffer) until the Error Reset, (latches) Command is issued. This prevents further data from becoming available in the receiver until the Reset is issued.

If the Interrupt on Every Character mode is selected, the interrupt vector will be different if error states exist in the status register. If receiver overrun should occur, despite the quadruple buffering, the most recent character received will be loaded. The character preceding it will be lost. When the character whch has been written over other characters is read, the Overflow bit will be set and the "Special Receive Condition" vector returned if "Status Affects Vector" is enabled.

It is possible to use the SIO in a polled environment. This requires monitoring of the "Receive Character Available" bit to know when to read a character. This bit is reset automatically when the receive buffers are all empty. The "Transmit Buffer Empty" bit is high whenever the transmit buffer is empty. In polled operation, it should be checked before writing data into the transmitter to prevent overwriting of data.

ASYNCHRONOUS FORMAT



TRANSMISSION

A data character sent by the SIO will be assembled as follows in asynchronous modes:

Idle state (no characters being sent) is a marking line (high) unless a break has been programmed in the control register, in which case, the line will remain spacing until the "send break" command has been removed or the chip is reset.

Transmission cannot begin unless the Transmit Enable bit is set. If the Auto Enables option is selected, then $\overline{CTS}$ must be low as well. If the 5 bits/character mode is selected, then unused bits (D_5, D_6 and D_7) must be zero in each data byte written into the SIO.

RECEIVING

Asynchronous reception will begin when the Receiver Enable bit is set. If the Auto Enables option is selected, the $\overline{DCD}$ must be low as well. A low (spacing) condition on RxD indicates a start bit. If the low persists for ½ bit time, the start bit is assumed to be valid and the data input is then sampled at mid-bit time until the entire character is assembled. This method of detecting a start bit improves error rejection when noise spikes exist on an otherwise marking line. If the X1 clock mode is selected, bit synchronization must be accomplished externally.

Synchronous Modes

The various synchronous modes all require a x1 clock for transmission and reception. Data is sampled on the rising edge of $\overline{RxC}$. Transmitter data transitions occur on the falling edge of $\overline{TxC}$.

In all cases, the receiver is in a hunt mode after a reset (internal or external). The hunt can begin only when the receiver is enabled. Only when character synchronization has been achieved can data transfer begin. If there is a loss of character synchronization, the hunt mode can be re-entered by writing a control word with the "Enter Hunt Mode" bit set.

The differences in operation of the monosync, bisync and external sync modes are only in the manner in which initial synchronization is achieved. Note: The mode of operation must be selected before the sync characters are loaded, since the registers are used differently in the various modes.

MONOSYNC; (8-BIT SYNC MODE)

Matching of a single sync character, programmed into Write register 7, implies character synchronization, which enables data transfer.

BISYNC: (16-BIT SYNC MODE)

Matching of two adjacent sync characters programmed in Write Registers 6 and 7 implies character synchronization. In both monosync and bisync modes, the $\overline{SYNC}$ pin will be active (low) any time the sync character sequence is detected and will remain low for the clock cycle in which it is detected.

EXTERNAL SYNC MODE

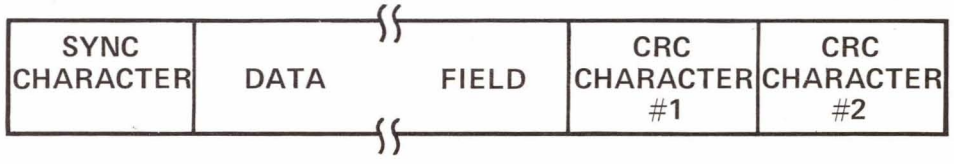
In this mode, character assembly *begins* on the first rising edge of $\overline{RxC}$ after the $\overline{SYNC}$ pin becomes active (low). It should be held active for at least three complete clock cycles.

In Monosync, Bisync and External sync modes, assembly will continue until the SIO is reset (either internally or with the Reset pin) or until the receiver is disabled (by command or with the $\overline{DCD}$ pin in the Auto Enables Mode) or until the CPU sets the "Enter Hunt Mode" bit.

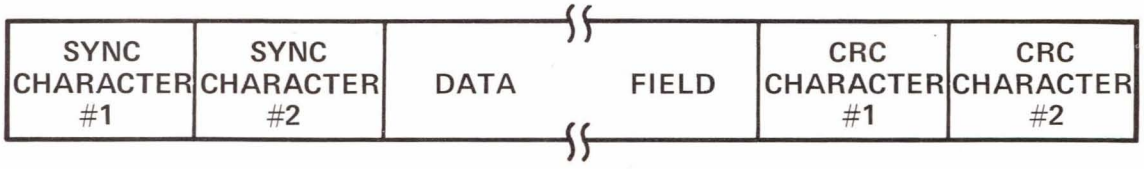
After initial synchronization has been achieved, the Monosync, Bisync, and External Sync modes are very similar. Any differences will be noted in the following, which is meant to apply to all three modes.

SYNCHRONOUS FORMATS

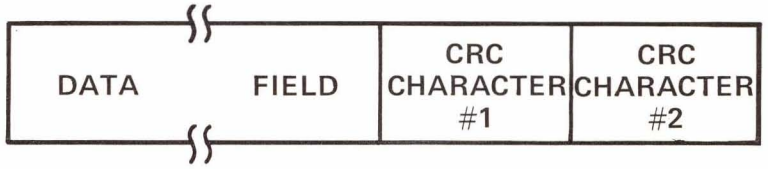
MONOSYNC MESSAGE FORMAT (Internal Sync Detect)



BISYNC MESSAGE FORMAT (Internal Sync Detect)



EXTERNAL SYNC DETECT FORMAT



Synchronous Modes (Except SDLC) Transmission:

A. Default state (after a Reset or transmitter not enabled) is a marking line. Break may be programmed to generate a spacing line, which begins as soon as programmed, regardless of the contents of the send register. With the transmitter enabled, and after modes have been selected, default is continuous transmission of the 8 or 16 bit sync character depending on which mode is selected.

B. Several Interrupt modes are possible:

1. Transmit interrupts enabled – every time that the transmit buffer becomes empty, an interrupt will be generated if the "Transmit Interrupt Enable" bit is set. The interrupt may be satisfied by either writing another character into the transmitter or by resetting the Transmitter Interrupt pending latch with the "Reset Transmitter Interrupt Pending" command (Command 5). If the interrupt is satisfied with this command and nothing more is written into the transmitter, there will be no further transmitter interrupts, as it is the buffer *becoming* empty that causes the interrupt. When another character is written, the transmitter can again become empty and interrupt again.
2. External/Status interrupts enabled – If the External/Status Interrupt Enable bit is set, Transmitter conditions such as starting to send CRC characters, starting to send Sync characters, and $\overline{\text{CTS}}$ changing state cause interrupts, which have a unique vector if "Status Affects Vector" is set.
3. All interrupts may be disabled for operation in a polled mode or to prevent interrupts at inappropriate times in a program's execution.

C. If CRC is not enabled, sync characters will automatically be inserted when the transmitter has no data to send. An interrupt is generated only after the first automatically inserted sync character has been loaded. If CRC is enabled, the first time the transmitter has no data to send, the 16-bit CRC is automatically sent, followed by sync characters. While sending CRC, the "Sending CRC/SYNCS" bit is set and the "Transmit Buffer Empty" bit indicates full. CRC is not calculated on the automatically inserted sync characters, but it will be calculated on any sync character sent as data unless the CRC generator is disabled when that character is loaded to the transmit shift register from the transmit buffer. When the CRC has been sent, the "Transmit Buffer Empty" bit goes high again, and an interrupt is generated to indicate that another message can begin. Control of the CRC generator may procede as follows:

The CRC generator should be reset by issuing the "RESET TRANSMIT CRC GENERATOR" Command, before any data is loaded. After CRC and the entire transmitter is enabled, data may be loaded. Before CRC is to be sent (but after the first data has been loaded), the CRC/SYNC SENT/SENDING flag must be reset with the "RESET CRC/SYNC SENT SENDING" Command.

Because sending of the CRC is inhibited when the CRC/SYNCS SENT/SENDING flag is set, the SIO can be used to automatically insert fill characters within messages instead of automatically sending the CRC. CRC is not calculated on syncs automatically inserted and when the end of the message is reached, the flag can be reset, thus allowing the CRC to be sent.

D. If the transmitter is disabled while a character is being sent, that character (whether Data or SYNC) will be sent as normal but will be followed by a marking line rather than CRC or sync characters. A character in the buffer when the transmitter is disabled will remain in the buffer. However, a programmed break will be effective as soon as it is written into the control register. Characters being transmitted, if any, will be lost.

E. In all modes, characters are sent low-order bits first, ie., D_0 before D_1, etc. for as many bits as are programmed. This requires right-hand justification of data to be transmitted if word length is less than 8 bits. If word length is 5 bits or less, the special technique described in the "Transmit Bits/Char" section must be used for the data format.

Synchronous Modes (Except SDLC) Reception:

A. After programming the mode and sync characters (in that order), the receiver may be enabled. It will then be in the Hunt Mode and will stay in that mode until:

1. A match is made with a single sync character (monosync mode) or
2. A match is made with a dual sync character (BiSync mode) or
3. The external $\overline{\text{SYNC}}$ pin is forced low. In cases (1) and (2) the external $\overline{\text{SYNC}}$ pin is an output which indicates that character synchronization has been achieved. In case (3) it is an input.

B. Character assembly begins after sync has been achieved. Four interrupt modes are possible.

1. No interrupts enabled – for a purely polled operation or for "off line" conditions.
2. Interrupt on first character only. This mode would normally be used to start a software polling loop or a block transfer instruction using the $\overline{\text{WAIT/READY}}$ output to synchronize the CPU to the incoming data rate. It could also be used with a DMA device. In this mode, the SIO will interrupt on the first character and thereafter will only interrupt if errors are detected. The mode is reset with the "Reset Receive Interrupt on First Character" command (Command 4).

   The first character received after this command is issued will also cause an interrupt. If External/Status interrupts are enabled, they may interrupt at any time. Parity errors do not cause interrupts in this mode, but End-of-Frame (SDLC Mode) and receiver overrun do cause interrupts.
3. Interrupt on every character – whenever the receiver buffer has a character an interrupt is generated. Error and special conditions generate a special vector if the "Status Affects Vector" mode is selected. A parity error may optionally not generate the special vector.

C. CRC checking generation may be used in the synchronous modes.

1. Calculation of the CRC on a particular character begins 8 bit times after the word has been transferred to the receive buffer. If CRC is enabled before the next character is transferred to the receive buffer, CRC will be calculated on the character. If CRC is disabled before the time of the next transfer, calculation will proceed on the word in progress, but the word just transferred to the buffer will not be included. This allows starting and stopping CRC checking on the various characters employed in BiSync.
2. The CRC may be enabled and disabled as many times as necessary for a given calculation.
3. CRC Codes are selected during the mode selection process. Either the CRC-16 polynomial $X^{16} + X^{15} + X^2 + 1$ or the SDLC polynomial $X^{16} + X^{12} + X^5 + 1$ may be used. In all except SDLC mode, the CRC calculator and checker are reset to all 0's. Transmitter and receiver must use the same polynomial.
4. In Monosync, Bisync and External Sync modes, the CRC/FRAMING ERROR bit contains the result of the comparison of the CRC checker to "all zeros" 16 bit times after the character has been loaded from the receive shift register to the buffer. The comparison is made with each load and is valid only as long as the character remains in the buffer. If time increases down the page, then the following holds:

   Character "A" loaded into the buffer

   Character "B" loaded into the buffer...
   If CRC is disabled before "C" is in the buffer it will not be calculated on "B".

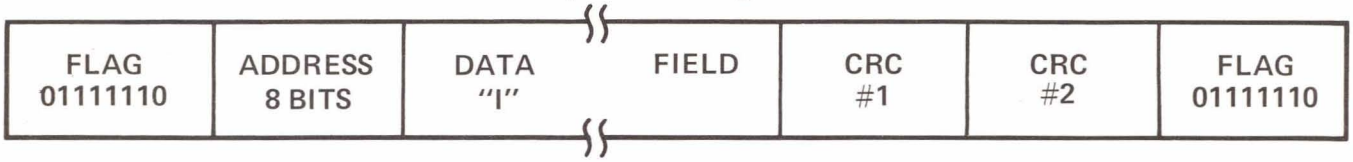
   Character "C" loaded into buffer...
   After "C" is loaded the "CRC FRAMING ERROR" bit shows the result of the comparison *thru* Character "A".

   Character "D" loaded into buffer...
   After "D" is in buffer, the CRC ERROR bit shows the result of the comparison thru Character "B".

   Because of the serial operation of the CRC calculation, the receiver clock ($\overline{\text{RxC}}$) must go through 16 cycles after the CRC character has been loaded into the receive buffer (20 cycles after the last bit is at the SIO RxD pin) before the CRC calculation is complete.

TRANSMISSION

SDLC/HDLC Message Format



SDLC MODE TRANSMISSION:

A. Normally, the CRC generator should be reset (with the "Reset Transmit CRC Generator" command) before a data block is transmitted. Reset may occur any time *after* the CRC of the previous message has been sent. During the time that CRC is being sent, the CRC/SYNC SENT/SENDING bit will be set, but the TRANS BUFFER EMPTY bit will not be set. After the CRC has been sent, the TRANS BUFFER EMPTY bit is set again, which will cause an interrupt signifying that the CRC has been sent, if transmit interrupts are enabled.

B. The idle device state (if the transmitter is enabled) is continuous flags being transmitted. If the transmitter is not enabled, a marking line is sent (idle line state).

C. An abort sequence may be sent by issuing the "Send Abort" command (Command 1). This causes at least 8 but less than 14 one's to be sent before the line reverts to continuous flags. Any data being transmitted and any data in the transmit buffer will be lost.

D. One to 8 bits per character may be sent. See the Register Description of Write Register 5, Transmit Bits Char. for an explanation of how this is accomplished. Since the number of bits/character may be changed "on the fly", this feature may be used to fill a data field with any number of bits. When used in conjunction with the Receiver Residue Codes, the SIO may receive a message of any number of bits length and retransmit it exactly as received with no previous information about the character structure of the I-field (if any). A change in the number of bits/character will not affect the character in the process of being shifted out. Characters will be sent with the number of bits programmed at the time that the character is loaded from the buffer to the transmitter.

E. As in other synchronous modes, the two byte CRC sequence will be sent automatically when the transmitter has no more data to send, i.e. when there is no character in the transmit buffer and the transmit shift register is empty. When the CRC sending begins, the CRC/SYNCS SENT/SENDING bit is set and a status change interrupt is generated if external/status interrupts are enabled. This may be used as a transmitter underrun indication. After the CRC has been sent, the line reverts to continuous flags, without shared zeros, i.e. ...
01111110011111110011111100 ....

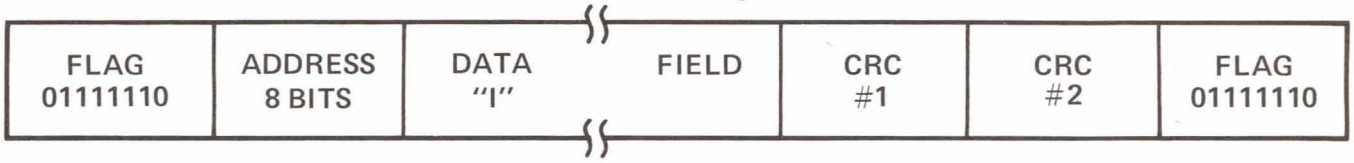
Control of the CRC generator may proceed as follows:

0. Set up necessary mode (only at initial power on), enable transmitter
1. Reset CRC generator
2. Write first byte of data (i.e. address)
3. Reset CRC/SYNCS SENT/SENDING bit
4. Write rest of data
5. After data is complete, CRC & flags will be sent automatically, and this sequence can repeat from 1.

F. Extra zeros are automatically inserted in the data stream where required to fulfill the requirement of 5 ones maximum in a row, except for flags or aborts.

G. When SDLC mode is selected, Reset of the CRC generator is actually a preset to all 1's. The SDLC CRC code must be selected.

RECEPTION
SDLC/HDLC Message Format



SDLC OPERATION, RECEIVER

A. Data transfer beings with the first non-flag character received after at least one flag (01111110) has been received if Address Search Mode has not been enabled. If Address Search Mode is enabled, then a flag rollowed by either the programmed address or the global address (11111111) is required before data transfer will begin.

1. If interrupts are disabled, the presence of characters in the receive buffer can be detected by observing the Receive Character Available bit in Read Register 0.
2. If the "Interrupt on First Character Only" mode has been selected, this would normally be used to initiate a block transfer. If the length of the message is unknown, the "special condition" (End of Frame) interrupt may be used to exit the instruction or software loop. The "Reset Interrupt on first character" command (Command 4) must be issued before an interrupt for a following block's first character can be generated.
3. Flags are not transferred. The extra zeros inserted in transmission are automatically deleted.
4. Aborts are detected as 7 or more one's and cause a status interrupt (if enabled) with the Break/Abort bit set in Read Register 0. After the "Reset External/ Status Interrupts" command (Command 2) has been issued, a second interrupt will occur when the continuous one's condition has been cleared.

B. In SCLC mode, control of the receive CRC checker is automatic. It is reset by the leading flag and CRC is calculated up to the final flag. The byte which has the "End-of-Frame" bit set is the byte which contains the result of the CRC check. If the CRC/Framing Error bit is not set, then the CRC indicates a valid message. A special check sequence is used for the SDLC check because of the preset to all one's. The final check must be 0001110100001111.

C. Character length may be changed "on the fly." If address and control bytes are processed as 8-bit characters, the receiver may be switched to a smaller character length during the time that the first information character is being assembled. This change must be made quickly enough so that it is effective before the number of bits specified have been assembled, i.e., if the change is to be from the 8-bit control to a 7-bit information field character length, the change must be made *before* the first 7 bits of the I-field have been assembled.

D. If address search mode is not used, or if messages have multi-byte addresses, an unwanted message need not be completely read by the CPU. Once the determination has been made that the message is not needed, writing the "Enter Hunt Mode" bit will suspend receiption until another message headed by a flag has been received.

E. When the trailing flag is received, an interrupt with a special vector is generated (if enabled). This signals that the byte with the "End of Frame" bit set has been received. In addition to the results of the CRC check, Read Register 1 has 3 bits of Residue Code valid at this time. For those cases in which the number of bits in the I-field is not an integral multiple of the character length used, these bits indicate the boundary between the CRC check bits and the I-field bits. For a detailed description of the meaning of these bits, see the description of the Residue Codes in Read Register 1.

F. Parity checking may be used on data in the information field only if 5-7 bit characters are used and only if a half-duplex protocol is being used. (There are no separate controls for parity on the receiver and transmitter so parity cannot, for example, be simultaneously disabled for transmitting an 8-bit address and enabled for receiving a 5-bit I-field character).

General

The Z80-SIO is a multi-function peripheral component specifically designed to satisfy a wide variety of serial data communications requirements in microcomputer systems. Its basic role is that of a serial to parallel, parallel to serial converter/controller but within that role it is configured by systems software programming so that its function or "personality" can be optimized for a given serial data communications application.

To program the Z80-SIO the systems software issues a series of commands that initialize the basic mode of operation desired and other commands to qualify conditions within the mode selected i.e. Stop Bits, Bits/Char, Sync Char etc. The command structure of the Z80-SIO is designed to take advantage of the powerful Z80 BLOCK I/O instructions to simplify programming, minimize overhead and optimize CPU interaction activities.

Each of the two channels of the Z80-SIO contain command registers that must be programmed via system software prior to functional operation. The channel select input (B/$\overline{A}$) and the control/data input (C/$\overline{D}$) are the command structure addressing controls, normally controlled by the address bus of the Z80 CPU.

C/$\overline{D}$	B/$\overline{A}$	Function
0	0	Channel A Data
0	1	Channel B Data
1	0	Channel A Commands/Status
1	1	Channel B Commands/Status

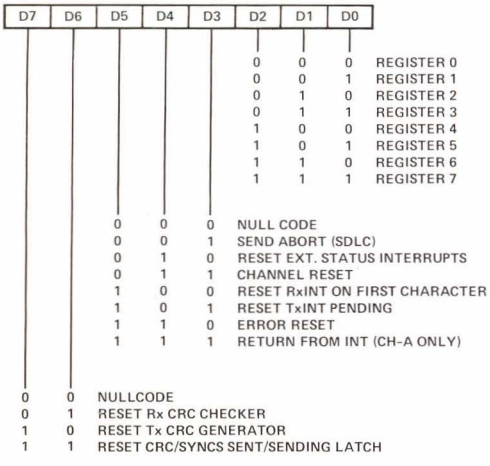
Write Registers

The Z80-SIO contains eight (8) registers in each channel that are programmed (written into) by the system software to configure the functional personality of each channel. All Write Registers, with the exception of Write Register 0, require two bytes to be properly programmed. The first byte contains 3 bits which point to the selected register (D0-D2) the second byte is the actual control word that is being written that register to configure the SIO.

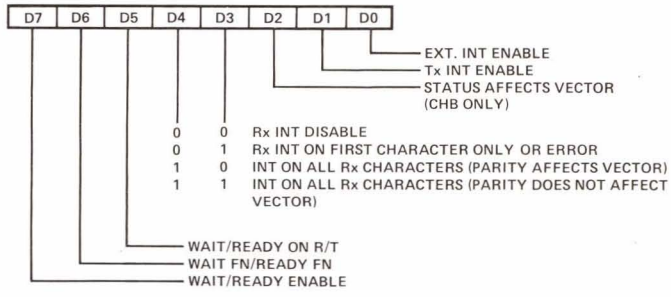
Write Register 0 is a special case. RESET (either internal command or external input) will initialize the SIO to Write Register 0. All basic commands (CMD2-CMD0) and CRC controls (CRC0, CRC1) can be accessed with a single byte using Write Register 0.

Contained in the first byte of any Write Register access are the basic commands (CMD2-CMD0) and the CRC controls (CRC0,CRC1) so that maximum system control and flexibility is maintained.

WRITE REGISTER 0

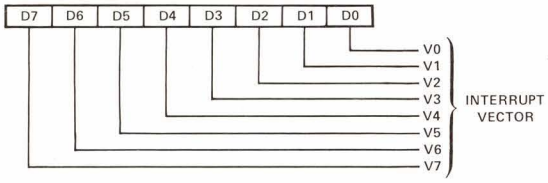


WRITE REGISTER 1

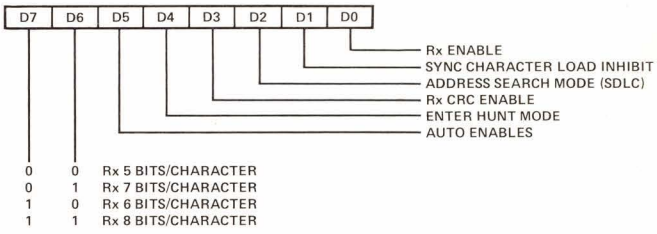


Write Registers (continued)

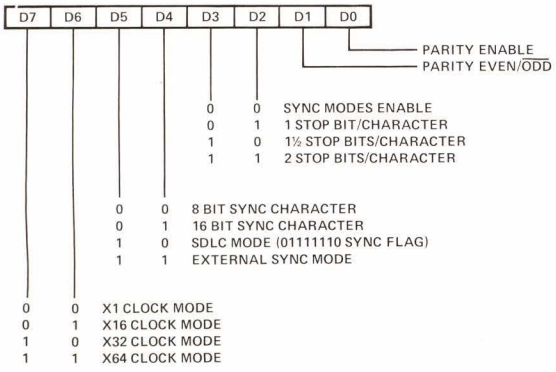
WRITE REGISTER 2



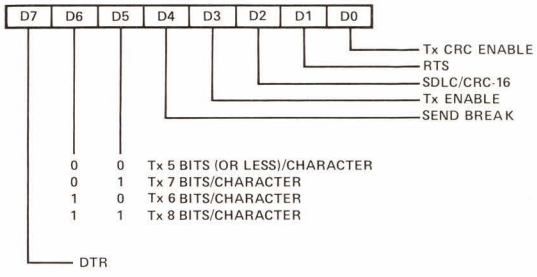
WRITE REGISTER 3



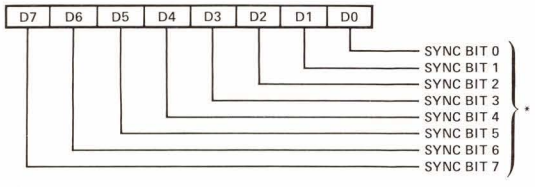
WRITE REGISTER 4



WRITE REGISTER 5

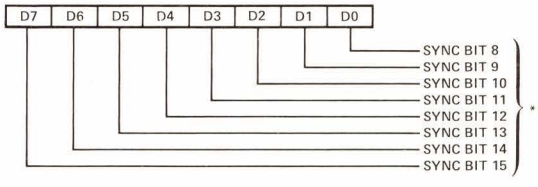


WRITE REGISTER 6



*ALSO SDLC ADDRESS FIELD

WRITE REGISTER 7



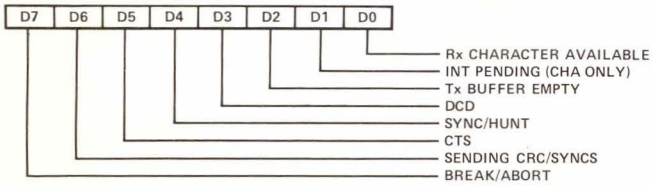
*FOR SDLC IT MUST BE PROGRAMMED TO "01111110" FOR FLAG RECOGNITION

Read Registers

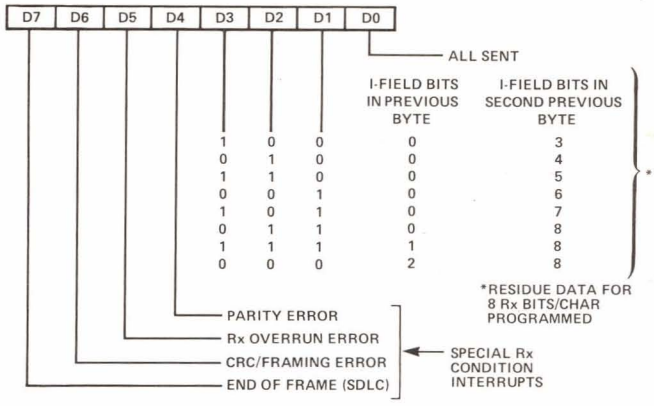
The Z80-SIO contains three (3) registers that can be read to obtain the status of each channel. Status information includes error conditions, interrupt vector, and standard communication interface protocol signals. To read the contents of a selected Read Register the system software must first write out to the SIO the byte containing pointer information (D0-D2) in exactly the same manner as a Write Register operation. Then by issuing a READ operation the contents of the addressed Read/Status Register can be read by the Z80-CPU.

The real power in this type of command structure is that the programmer has complete freedom after pointing to the selected register of either Reading or Writing to initialize or test that register. By designing software to initialize the Z80-SIO in a modular, structured fashion, the programmer can use the powerful Z80 BLOCK I/O instructions to significantly simplify and speed his software development and debug.

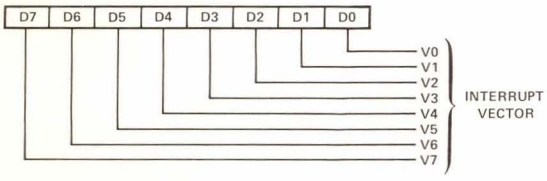
READ REGISTER 0



READ REGISTER 1



READ REGISTER 2 (Channel B Only)



Each channel contains the following control registers, addressed as commands (not data):

WRITE REGISTER 0, a command register:

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
CRC Reset Code 1	CRC Reset Code 0	CMD 2	CMD 1	CMD 0	PNT 2	PNT 1	PNT 0

$PNT_0 - PNT_2$ (D_0-D_2)

These are pointer bits which tell the SIO into which register the following byte is to be written. The first byte written into each channel after a reset (either by command or with the external reset pin) will go to write register 0. The byte following a read or write to any register (not register 0) will be to register 0.

CMD_0 to CMD_2 (D_3-D_5)

These are commands:

Command	CMD_2	CMD_1	CMD_0	
0	0	0	0	Null Command (no affect)
1	0	0	1	Send Abort (SDLC Mode)
2	0	1	0	Reset External/Status Interrupts
3	0	1	1	Channel Reset
4	1	0	0	Reset Receive Interrupt on First Character
5	1	0	1	Reset Transmitter Interrupt Pending
6	1	1	0	Error Reset (latches)
7	1	1	1	Return from Interrupt (Channel A only)

COMMAND 0 (The null command) has no affect. It's normal use is to do nothing while setting the pointers for a following byte.

COMMAND 1 (Send Abort) is used only with the SDLC mode to generate a sequence of 8 to 13 ones.

COMMAND 2 (Reset External/Status Interrupts). After an external or status interrupt (indicating a change on a modem line or a break condition, for example) the status bits of Read Register 0 are latched. This command re-enables them and and allows interrupts to occur. The latching allows capture of short pulses on the inputs until such time as the CPU can read the change.

COMMAND 3 (Channel Reset.) This command performs the same operation as an external reset, but only on a single channel. The Channel A Reset also resets the interrupt prioritization logic. All control registers must be rewritten after this command. After this command is written, four extra system (Φ) clock cycles should be allowed for the SIO reset time before any additional commands or controls are written into that channel of the SIO.

COMMAND 4 (Reset Receive Interrupt on First Receive Character.) If the "interrupt only on first receive character" mode of operation is programmed, it needs to be reactivated after each complete message is received, in preparation for the next message.

COMMAND 5 (Reset Transmitter Interrupt Pending.) The transmitter will interrupt when it becomes empty if the "interrupt every character" mode is selected. In those cases when there are no additional characters to be sent, issuing this command will prevent further transmitter interrupts (i.e. until after the next character has been loaded into the transmitter.)

COMMAND 6 (Error Reset, Latches.) Parity and overrun errors are latched in Read Register 1 until reset with this command. This allows errors occuring in block transfers to be examined only at the end of the block.

COMMAND 7 (Return from Interrupt.) This command (which must be issued in Channel A) is interpreted by the SIO in exactly the same way as it would interpret an RETI Command on the data bus, i.e. it would reset the Interrupt Under Service latch of the internal device (receiver, transmitter, etc.) under service and thus, by means of the daisy chain, allow lower priority devices to interrupt. The internal daisy chain may be used even in systems with no external daisy chain and no RETI Command by use of this command.

CRC RESET CODE 0 (D_6) and CRC RESET CODE 1 (D_7)

Together, these bits specify three reset modes.

CRC Reset Code 1	CRC Reset Code 0	
0	0	Null Code (no affect)
0	1	Reset Receive CRC Checker
1	0	Reset Transmit CRC Generator
1	1	Reset CRC/SYNCS Sent Sending latch

WRITE REGISTER 1 contains the control bits for the various interrupt and WAIT/READY modes.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Wait/ Ready Enable	ReadyFN/ WaitFN	W/Ready On $R/\overline{T}$	Receive interrupt Mode 1	Receive Interrupt Mode 0	Status Affects Vector	Trans Interrupt Enable	Ext Interrupts Enable

EXT INT ENABLE (D_0)

External Interrupt Enable, allows interrupts to occur as a result of transitions on the $\overline{DCD}$, $\overline{CTS}$ or $\overline{SYNC}$ lines or as a result of a Break Condition or the beginning of sending CRC or sync characters.

TRANS INT ENABLE (D_1)

Transmitter Interrupt Enable. If enabled, interrupts will occur whenever the transmitter buffer becomes empty.

STATUS AFFECTS VECTOR (D_2) (Channel B Only)

If this mode is selected, the vector returned from an interrupt acknowledge cycle will be variable according to the following:

	V_3	V_2	V_1	
Ch B	0	0	0	Ch B Transmit Buffer Empty
	0	0	1	Ch B External/Status Change
	0	1	0	Ch B Receive Character Available
	0	1	1	Ch B Special Receive Condition*
Ch A	1	0	0	Ch A Transmit Buffer Empty
	1	0	1	Ch A External/Status Change
	1	1	0	Ch A Receive Character Available
	1	1	1	Ch A Special Receive Condition*

*Special Receive Conditions → PARITY ERROR; Rx OVERRUN ERROR; CRC/FRAMING ERROR; END OF FRAME (SDLC)

If this bit is 0, the fixed vector programmed in the vector register is returned.

REC INT MODE 0 (D_3), REC INT MODE 1 (D_4)

Receive Interrupt Mode 0 and Receive Interrupt Mode 1 together specify the various character available conditions:

MODE	D_4 REC INT MODE 1	D_3 REC INT MODE 0	
0	0	0	Receiver interrupts disabled
1	0	1	Receive interrupt on first character only error
2	1	0	Interrupt on all Receive Characters-Parity/affects Vector
3	1	1	Interrupt on all Receive Characters-Parity error does not affect Vector.

W/READY on $R/\overline{T}$ (D_5)

When the $\overline{W/Ready}$ line is enabled, this bit selects whether it will be active when the receiver is empty (bit=1) or when the transmit buffer is full (bit=0).

READY FN/WAIT FN (D_6)

When used with the CPU as a Wait line, this bit should be programmed "0". When used with a DMA as a Ready line, it must be programmed "1". The Ready function can occur any time, regardless of whether the SIO is addressed or not. The Wait function is active only if the CPU attempts to read SIO data that has not yet been received, as would frequently occur if block transfer instructions are used with the SIO, or tries to write data while the transmit buffer is still full.

Also, as a Wait function, the output is open drain and occurs from the negative edge of Φ. As a Ready function, it is actively driven high and occurs from the positive edge of Φ.

WAIT/READY ENABL (D_7)

The Wait/Ready pin will remain high (Ready mode) or floating (Wait mode) until this bit is programmed to one.

WRITE REGISTER 2

Write Register 2 is the interrupt vector register and it exists only in Channel B. V_4-V_7 and V_0 are always returned exactly as written. V_1-V_3 are returned as written if the "Status Affects Vector", Control bit is "0".

WRITE REGISTER 3

Write register 3 contains control bits for some of the receiver logic.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
RCVR Bits/ Char 0	RCVR Bits/ Char 1	Auto Enables	Enter Hunt Mode	RECVR CRC Enabl	Address Search Mode	Sync Char Load Inhibit	Receiver Enabl

RECEIVER ENABLE (D_0)

A "1" programmed here allows receiver operations to begin.

SYNC CHAR LOAD INHIBIT (D_1)

Sync characters preceding a message will not be loaded into the receiver buffers if this option is selected. The CRC calculation is not stopped by the sync character being stripped.

ADDRESS SEARCH MODE (D_2)

If the SDLC mode is selected, this mode will cause messages with addresses not matching the programmed address or the global (11111111) address to be rejected, i.e., no interrupts occur unless an address match occurs if this mode is selected.

RECVR CRC ENABLE (D_3)

Receiver CRC Enable. If this bit is set, a calculation of CRC begins (or restarts) at the start of the last character transferred from the receive register to the buffer stack regardless of the number of characters in the stack.

ENTER HUNT MODE (D_4)

If character synchronization is lost for any reason, or if in SDLC mode, it is determined that the contents of an incoming message are not needed, Hunt mode may be reentered by writing a "1" to this bit.

AUTO ENABLES (D_5)

If this mode is selected, the $\overline{DCD}$ and $\overline{CTS}$ inputs are receiver and transmitter enables, respectively. If the mode is not selected, $\overline{DCD}$ and $\overline{CTS}$ are only inputs to their corresponding bits in Read Register 0.

RCVR BITS/CHAR 1 (D_6), RCVR BITS/CHAR 0 (D_7)

These bits together determine the number of serial receive bits that will be assembled to form a character.

These bits may be changed during the time that a character is being assembled, if it is done before the number of bits currently programmed is reached.

D_6 Receiver Bits/Character 1	D_7 Receiver Bits/character 0	Bits/Character
0	0	5
0	1	6
1	0	7
1	1	8

WRITE REGISTER 4

Write Register 4 contains control bits affecting both the receiver and transmitter.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Clock Rate 1	Clock Rate 0	Sync Modes 1	Sync Modes 0	Stop Bits 1	Stop Bits 0	Parity Even/ $\overline{Odd}$	Parity

PARITY (D_0)

If this bit is set, an additional bit position (in addition to those specified in the bits/character control) is added to transmitted data and is expected in receive data.

PARITY EVEN/$\overline{ODD}$ (D_1)

If parity is specified, this bit determines whether it is sent or checked as even or odd parity. (1=Even Parity)

STOP BITS 0 (D_2), STOP BITS 1 (D_3)

These bits determine the number of stop bits added to each asynchronous character sent. The receiver always checks for one stop bit.

The special (00) mode is used to signify that a synchronous mode is to be selected.

D_3 Stop Bits 1	D_2 Stop Bits 0	
0	0	Sync Modes
0	1	1 Stop Bit Per Character
1	0	1½ Stop Bits Per Character
1	1	2 Stop Bits Per Character

SYNC MODES 0 (D_4), SYNC MODES (D_5)

These select the various options for character synchronization:

Sync Mode 1	Sync Mode 0	
0	0	8-bit programmed sync
0	1	16-bit programmed sync
1	0	SDLC Mode (01111110 sync pattern)
1	1	External Sync Mode

CLOCK RATE 0 (D_6), CLOCK RATE 1 (D_7)

Specifies the multiplier between clock and data rates. For synchronous modes X1 must be specified. Any rate may be specified for the asynchronous modes. The same multiplier is used for both the receiver and transmitter.

In all modes, the system clock (Φ) must be at least 4.5 X the data rate. If the X1 clock rate is selected, bit synchronization must be accomplished externally.

Clock Rate 1	Clock Rate 0	
0	0	Data Rate X 1 = Clock Rate
0	1	Data Rate X16 = Clock Rate
1	0	Data Rate X32 = Clock Rate
1	1	Data Rate X64 = Clock Rate

WRITE REGISTER 5

Write Register 5 contains mostly control bits affecting the transmitter.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
DTR	Transmit Bits/ Char 0	Transmit Bits/ Char 1	Send Break	Transmit Enable	SDLC/ CRC16	RTS	Transmit CRC Enable

TRANSMIT CRC ENABLE (D_0)

This bit determines whether CRC is to be calculated on any particular send character. If set at the time of loading the character from the transmit buffer to the transmit shift register, CRC will be calculated on the character. CRC will not be automatically sent unless this bit is set when the transmitter is completely empty.

RTS (D_1)

Request to Send is the control bit for the $\overline{RTS}$ pin. When the RTS bit is set, the $\overline{RTS}$ goes active (low). When the bit is reset (to 0), the $\overline{RTS}$ pin will go inactive (high) only after the transmitter is empty.

SDLC/CRC/16 (D_2)

This bit selects the CRC code used by both the transmitter and the receiver. When reset, the SDLC polynomial $X^{16} + X^{12} + X^5 + 1$ is used. (In SDLC mode, the registers are preset to "all 1's" and a special check sequence is used.) When set, the CRC-16 polynomial $X^{16} + X^{15} + X^2 + 1$ is used.

TRANSMIT ENABLE (D_3)

Data will not be transmitted and the TxD pin will be held marking (high) until this bit is set. Data or Sync characters in the process of being transmitted will be completely sent if the transmit enable bit is reset after transmission has started. CRC characters will *not* be completely sent if the transmitter is disabled during the sending of a CRC character.

SEND BREAK (D_4)

When set, this bit directly forces the TxD pin spacing, regardless of any data being transmitted. When reset, the TxD pin is released.

TRANSMIT BITS/CHAR 0 (D_6), TRANSMIT BITS/ CHAR 1 (D_5),

These bits together control the number of bits that will be sent from each byte transferred to the transmit buffer.

D_5 Transmit Bits/Character 1	D_6 Transmit Bits/Character 0	Bits/Character
0	0	5 or less
0	1	6
1	0	7
1	1	8

Bits to be sent are assumed to be right justified. Low order bits (D_0) are sent first. The "5 or less" mode allows transmission of 1 to 5 bits in a character.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
1	1	1	1	0	0	0	D	Sends one bit
1	1	1	0	0	0	D	D	Sends two bits
1	1	0	0	0	D	D	D	Sends three bits
1	0	0	0	D	D	D	D	Sends four bits
0	0	0	D	D	D	D	D	Sends five bits

DTR (D_7)

Data Terminal Ready is the control bit for the $\overline{DTR}$ pin. When set, $\overline{DTR}$ is active (low). When reset (0) $\overline{DTR}$ is inactive (high).

WRITE REGISTER 6

This register contains the first 8 bits of a BiSync sequence. It must be programmed with the check address (if used) in SDLC mode, and must contain the sync character in the 8-bit sync mode. It contains the transmit sync character in the external sync mode.

WRITE REGISTER 7

This register contains the second byte of a 16-bit synchronization sequence, or the 8-bit sync character. For SDLC mode, it must be programmed to 01111110. It is not used in the external sync mode.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SYN15	SYN14	SYN13	SYN12	SYN11	SYN10	SYN9	SYN8

READ REGISTER 0

This is the register read if the register pointers are (000).

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Break/ Abort	Sending CRC/ Syncs	CTS	Sync/ Hunt	DCD	Transmit Buffer Empty	Interrupt Pending	Receive Character Available

RECEIVE CHARACTER AVAILABLE (D_0)

This bit is set when at least one character is available in the receive buffers.

INTERRUPT PENDING (D_1) (Channel A Only)

Any interrupt condition present in the entire SIO will cause this bit to be set, but it is present only in Channel A and is always 0 in Channel B.

TRANSMIT BUFFER EMPTY (D_2)

The Transmit Buffer Empty bit is set whenever the transmit buffer is empty, except when a CRC character is being sent in a synchronous mode.

DCD (D_3)

Shows the state of the $\overline{DCD}$ pin inverted at the time of the last change of any of the five "external/status" bits. (DCD, CTS, SYNC/HUNT, BREAK/ABORT or SENDING CRC/SYNCS.) To get the current state of the $\overline{DCD}$ pin, this bit must be read immediately following a "Reset External/Status Interrupts" command. (Command 2.)

$\overline{SYNC}$/HUNT (D_4)

In asynchronous modes, this bit is similar to the DCD and the CTS bits, except that it shows the state of the $\overline{SYNC}$ pin. In synchronous modes, this bit is reset when character synchronization is achieved and is set by writing the "Enter Hunt Mode" bit. Unlike the external pin, the bit remains reset until set by the "Enter Hunt Mode" bit. However, in the external sync mode, the bit gives the state of the external pin not hunt status.

CTS (D_5)

This bit is similar to the DCD bit, except that it shows the state of the $\overline{CTS}$ pin inverted.

SENDING CRC/SYNCS (D_6)

In synchronous modes, CRC is automatically sent when the transmitter is empty for the first time in a message. Interrupts are generated (if enabled) when this bit is set, but not when reset. If this bit is set and the TRANSMIT BUFFER EMPTY bit is not set, then the CRC character is being sent. TRANSMIT BUFFER EMPTY and SENDING CRC/SYNCS both set imply that SYNC characters are being sent.

BREAK/ABORT (D_7)

In asynchronous modes, this bit is set when a "break" is detected. After the inputs have been re-enabled (by the "Reset External/Status Interrupts" command, Command 2), the bit will be reset when the break stops. If "External/Status" interrupts are enabled, these changes of state cause interrupts. In SDLC mode, this bit is set by the detection of an abort sequence (7 or more 1's). It is not used in other synchronous modes.

READ REGISTER 1

This register is read when the register pointers are (001). The pointers automatically reset to (000) after a read from this register.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
End Of Frame (SDLC)	CRC/ Framing Error	Receiver Overrun Error	Parity Error	Residue Code 2	Residue Code 1	Residue Code 0	All Sent

ALL SENT (D_0)

In asynchronous modes, this bit is set when all characters have completely cleared the transmitter. Transitions of this bit do not cause interrupts. It is always set in synchronous modes.

RESIDUE CODE 0 (D_1)—RESIDUE CODE 2 (D_3)

These three bits indicate the length of the I-field in the SDLC mode in those cases where the I-field is not an integral multiple of the character length used. Only on the transfer on which the END OF FRAME (SDLC) bit is set do these codes have meaning.

For a receiver setting of eight bits per character, the codes signify the following:

Residue Code 2	Residue Code 1	Residue Code 0	I-Field Bits In Previous Byte	I-Field Bits In Second Previous Byte
1	0	0	0	3
0	1	0	0	4
1	1	0	0	5
0	0	1	0	6
1	0	1	0	7
0	1	1	0	8
1	1	1	1	8
0	0	0	2	8

I-Field bits are right-justified in all cases.

If a receive character length different from eight bits is used for the I-field, a table similar to the above may be constructed for each different character length. For no residue, i.e., the last character boundary coincides with the boundary of the I-Field and CRC Field, the Residue Code will always be:

Residue Code 2	Residue Code 1	Residue Code 0
1	0	1

PARITY ERROR (D_4)

When parity is enabled, this bit is set for those characters whose parity does not match the sense programmed. The bit is latched so that once an error occurs, the bit remains set until the Error Reset command, Command 6, is given.

RECEIVER OVERRUN ERROR (D_5)

This indicates that more than four characters have been received without a read from the CPU. Only the character that has been written over is flagged with this error, but when this character is read, the error condition is latched until reset by the Error Reset Command, Command 6. If "Status Affects Vector" is enabled, the character that has been overrun will interrupt with the "Special Receive Condition" vector.

CRC/FRAMING ERROR (D_6)

If a framing error occurs (in asynchronous modes), this bit is set (and not latched) only for the character on which it occurred. Detection of a framing error adds an additional ½ bit time to the character time so that the framing error will not also be interpreted as a new start bit. In synchronous modes, this bit indicates the result of comparing the CRC checker to the appropriate check value.

END OF FRAME (SDLC) (D_7)

In SDLC mode, this bit indicates that a valid ending flag has been received and that the CRC error and residue codes are valid.

READ REGISTER 2

This register contains the interrupt vector as written into Write Register 2 if the "Status Affects Vector" control bit is not set. If that control bit is set, it contains the interrupt vector as it would be returned were an interrupt from the SIO to be processed exactly at the time of the read. If no interrupts are pending, $V_3 = 0$, $V_2 = 1$, $V_1 = 1$ and other bits are as programmed. The register may be read only through Channel B.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
V_7	V_6	V_5	V_4	V_3	V_2	V_1	V_0

Variable if "Status Affects Vector" is enabled

Z80-SIO COMMAND STRUCTURE

Reg. #	Control C/D	Control RD	Control WR	DATA BITS D7	D6	D5	D4	D3	D2	D1	D0
0	1	1	0	CRC 1	CRC 0	CMD 2	CMD 1	CMD 0	0	0	0
	1	1	0	CRC 1	CRC 0	CMD 2	CMD 1	CMD 0	0	0	0
	1	0	1	Break/Abort	Sendg CRC/SYNC	CTS	SYNC/HUNT	DCD	TxBuffer EMPTY	INT Pending (CH-A Only)	RxChar Avail
1	1	1	0	CRC 1	CRC 0	CMD 2	CMD 1	CMD 0	0	0	1
	1	1	0	Wait/RDY EN	WaitFN/RDYFN	Wait/RDYon R/T	RxINT Mode 1	RxINT Mode 0	Status Affects V (CH-B Only)	TxINT EN	EXT INT EN
	1	0	1	Endo FrameSDLC	CRC FrameError	RxOVRN Error	Parity Error	Res. Code 2	Res. Code 1	Res. Code 0	All Sent
2	1	1	0	CRC 1	CRC 0	CMD 2	CMD 1	CMD 0	0	1	0
CH-B ONLY	1	1	0	V7	V6	V5	V4	V3	V2	V1	V0
	1	0	1	V7	V6	V5	V4	V3	V2	V1	V0
3	1	1	0	CRC 1	CRC 0	CMD 2	CMD 1	CMD 0	0	1	1
	1	1	0	RxBits/Char 0	RxBits/Char 1	Auto Enables	Enter Hunt Mode	RxCRC EN	AddrssSearch Md	SyncChar LD INH	RxEN
4	1	1	0	CRC 1	CRC 0	CMD 2	CMD 1	CMD 0	1	0	0
	1	1	0	Clock Rate 1	Clock Rate 0	Sync Mode 1	Sync Mode 0	Stop Bits 1	Stop Bits 0	Parity Even/Odd	Parity
5	1	1	0	CRC 1	CRC 0	CMD 2	CMD 1	CMD 0	1	0	1
	1	1	0	DTR	TxBits/Char 0	TxBits/Char 1	Send BREAK	TxEN	$\overline{\text{SDLC}}$/CRC-16	RTS	TxCRC EN
6	1	1	0	CRC 1	CRC 0	CMD 2	CMD 1	CMD 0	1	1	0
	1	1	0	SYNC/SDLC 7	SYNC/SDLC 6	SYNC/SDLC 5	SYNC/SDLC 4	SYNC/SDLC 3	SYNC/SDLC 2	SYNC/SDLC 1	SYNC/SDLC 0
7	1	1	0	CRC 1	CRC 0	CMD 2	CMD 1	CMD 0	1	1	1
	1	1	0	SYNC/SDLC 15	SYNC/SDLC 14	SYNC/SDLC 13	SYNC/SDLC 12	SYNC/SDLC 11	SYNC/SDLC 10	SYNC/SDLC 9	SYNC/SDLC 8

PROGRAMMING EXAMPLE

A typical start-up routine following an internal or external reset, would be as follows:

$B/\overline{A}$	$C/\overline{D}$	$\overline{RD}$	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	COMMENTS
1	1	1	0	0	0	0	0	0	1	0	Pointer set to Register 2B
1	1	1	V_7	V_6	V_5	V_4	V_3	V_2	V_1	V_0	Interrupt Vector loaded
1	1	1	0	0	0	0	0	1	0	0	Pointer set to Write Register 4B
1	1	1	0	1	X	X	0	1	1	1	Even parity, 1 stop bit, X16 clock, asynchronous mode selected
1	1	1	0	0	0	0	0	1	0	1	Pointer set to Write Register 5B
1	1	1	0	0	1	0	1	0	1	0	7 bits/transmit character, transmitter enabled
1	1	1	0	0	0	0	0	0	1	1	Pointer set to Write Register 3B
1	1	1	0	1	1	0	0	0	0	1	7 bits/receive character, DCD and CTS enable Receiver and Transmitter, Receiver enabled
1	1	1	0	0	0	0	0	0	0	1	Pointer set to Register 1B
1	1	1	0	0	0	1	0	1	1	1	Interrupt on every character, status affects Vector external/status interrupts enabled

Channel B is now setup to send and receive asynchronous data.

Setup for Channel A follows:

0	1	1	0	0	0	0	0	1	0	0	Pointer set to Write Register 4A
0	1	1	0	0	1	0	0	0	0	0	SDLC mode and X1 clock selected, no parity

Programming Example

$B/\overline{A}$	$C/\overline{D}$	$\overline{RD}$	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	COMMENTS
0	1	1	0	1	0	0	0	1	1	0	Pointer set to Write Register 6A, Reset Receive CRC Checker
0	1	1	AD_7	AD_6	AD_5	AD_4	AD_3	AD_2	AD_1	AD_0	SDLC message address entered
0	1	1	1	0	0	0	0	1	1	1	Pointer set to Write Register 7A, Reset Transmit CRC generator
0	1	1	0	1	1	1	1	1	1	0	SDLC Flag entered
0	1	1	0	0	0	0	0	0	0	1	Pointer set to Register 1A
0	1	1	0	0	0	1	0	1	1	1	Interrupt every character, status affects vector, external/status interrupts enabled
0	1	1	0	0	0	1	0	1	0	1	Pointer set to Write Register 5A, Reset External/Status Interrupts
0	1	1	1	1	1	0	1	0	0	0	SDLC CRC Code selected, 8 bits/transmit character, CRC and transmitter enabled
0	1	1	0	0	0	0	0	0	1	1	Pointer set to Write Register 3A
0	1	1	1	1	1	0	1	1	0	1	8 bits/receive character, DCD and CTS enable receiver and transmitter, receiver is enabled, SIO searches for programmed address

Channel A is now programmed for SDLC transfers.

0	0	1	D	D	D	D	D	D	D	D	Address byte to be sent by Ch. A
0	1	1	1	1	0	0	0	0	0	0	Reset CRC/SYNCS SENT/SENDING, pointer to register 0, so CRC can be automatically sent at end of message

Absolute Maximum Ratings

Temperature Under Bias	Specified operating range.
Storage Temperature	−65° C to +150° C
Voltage On Any Pin with Respect to Ground	−0.3V to +7V
Power Dissipation	1.5W

*Comment

Stresses above those listed under "Absolute Maximum Rating" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. Characteristics

T_A = 0°C to 70°C, V_{CC} = 5V ±5% unless otherwise specified

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	−0.3		.40	V	
V_{IHC}	Clock Input High Voltage	V_{CC}-.2[1]		V_{CC}	V	
V_{IL}	Input Low Voltage	−0.3		0.8	V	
V_{IH}	Input High Voltage	2.0		V_{CC}	V	
V_{OL}	Output Low Voltage			0.4	V	I_{OL} = 1.8 mA
V_{OH}	Output High Voltage	2.4			V	I_{OH} = −250μA
V_{CC}	Power Supply Current			100	mA	t_C = 400 nsec
I_{LI}	Input Leakage Current			10	μA	A_{IN} = 0 to V_{CC}
I_{LOH}	Tri-State Output Leakage Current in Float			10	μA	V_{OUT} = 2.4 to V_{CC}
I_{LOL}	Tri-State Output Leakage Current in Float			−10	μA	V_{OUT} = 0.4V
I_{LD}	Data Bus Leakage Current in Input Mode			±10	μA	0 ≤ V_{IN} ≤ V_{CC}

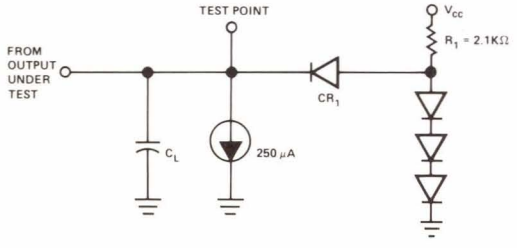
An external clock pull-up resistor of (330Ω) will meet both the AC and DC clock requirements.

Capacitance

T_A = 25°C, f = 1 MHz

Symbol	Parameter	Max.	Unit	Test Condition
C_Φ	Clock Capacitance	40	pF	Unmeasured Pins
C_{IN}	Input Capacitance	5	pF	Returned to Ground
C_{OUT}	Output Capacitance	10	pF	

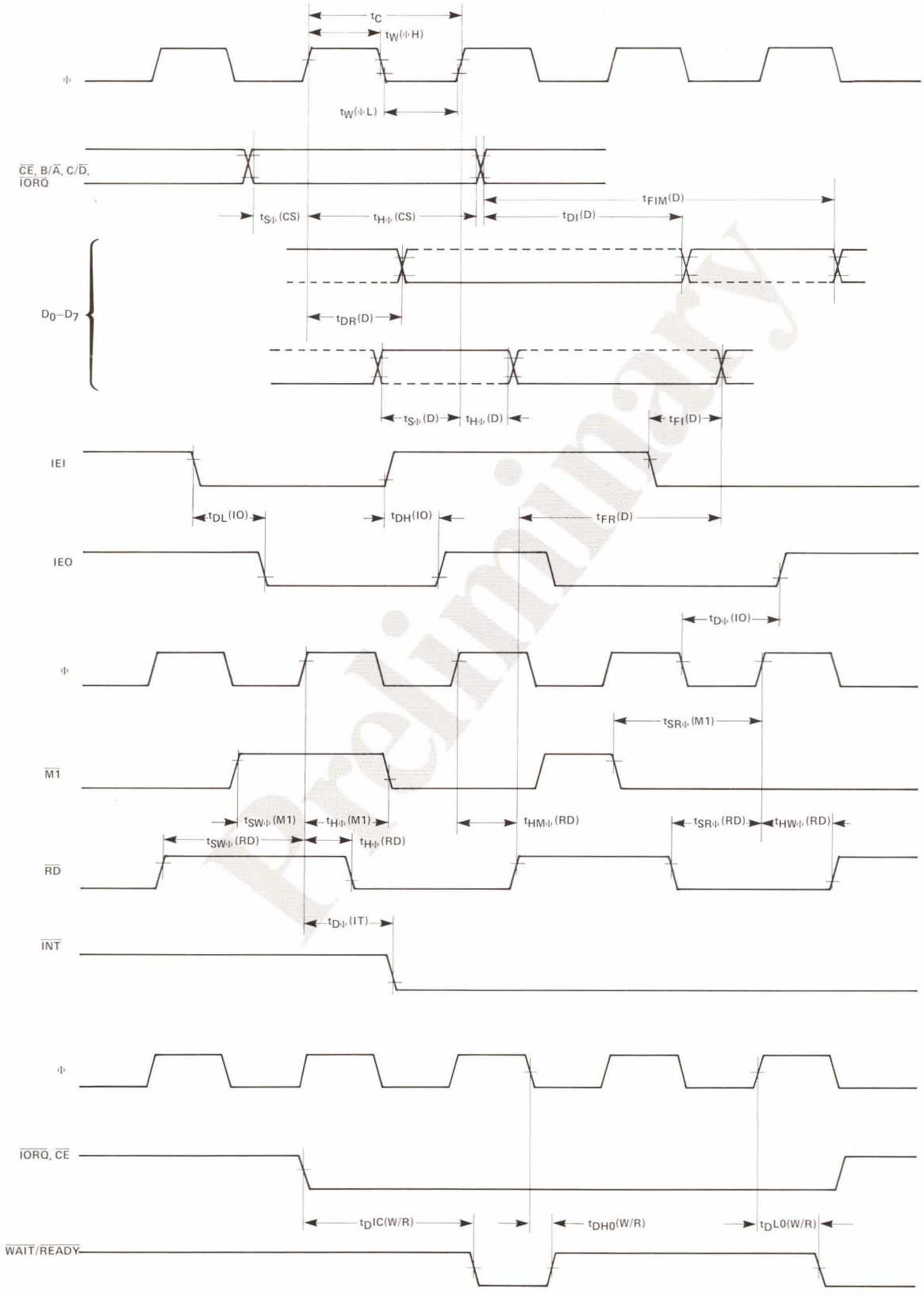
Load Circuit for Output



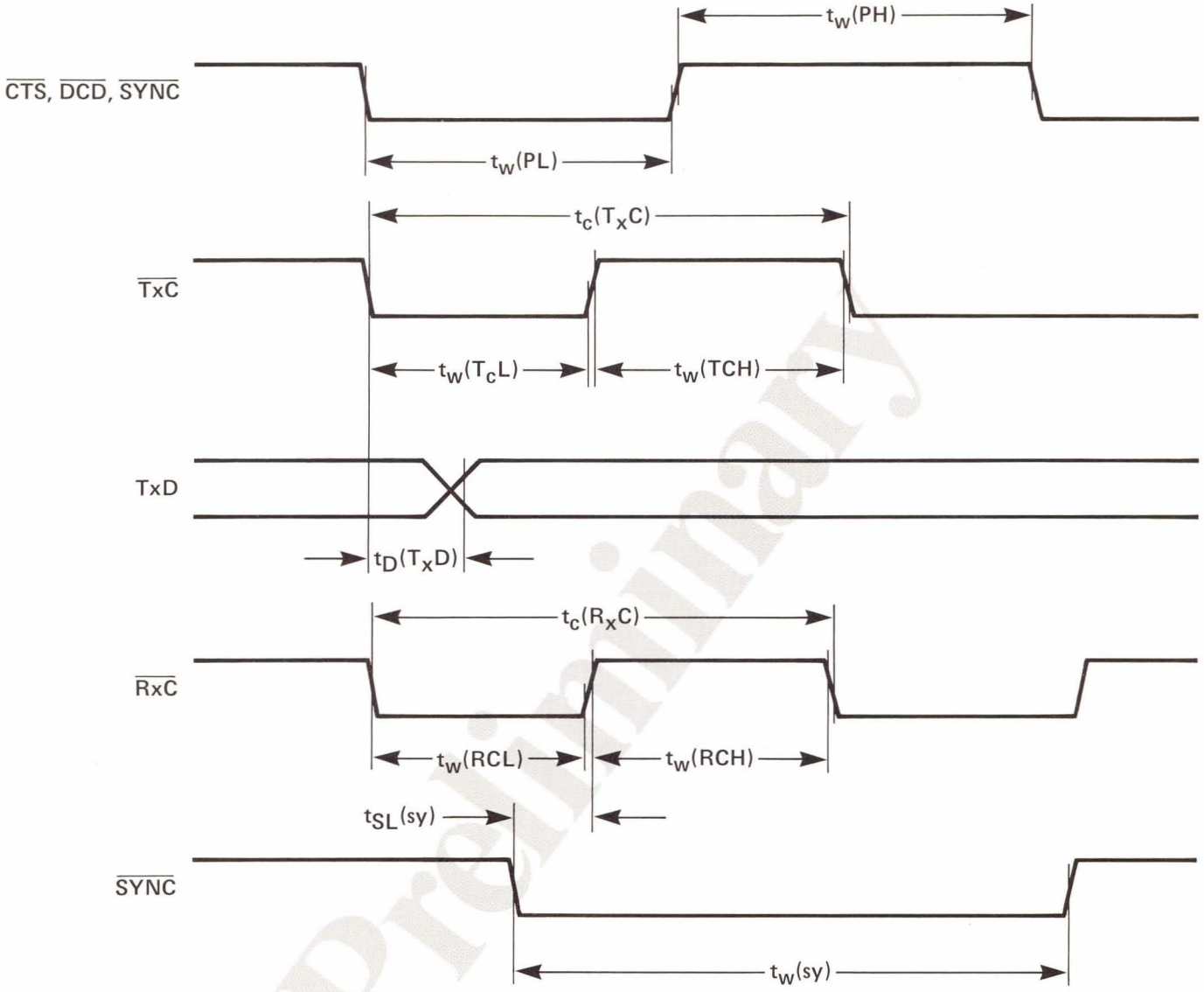
A.C. Timing Diagram

Timing measurements are made at the following voltages, unless otherwise specified:

	HIGH		LOW	
CLOCK	4.2V		.8V	Only for timing measurements
OUTPUT	2.0V		.8V	
INPUT	2.0V		.8V	
FLOAT	ΔV	=	±0.5V	



A.C. Timing Diagram (continued)



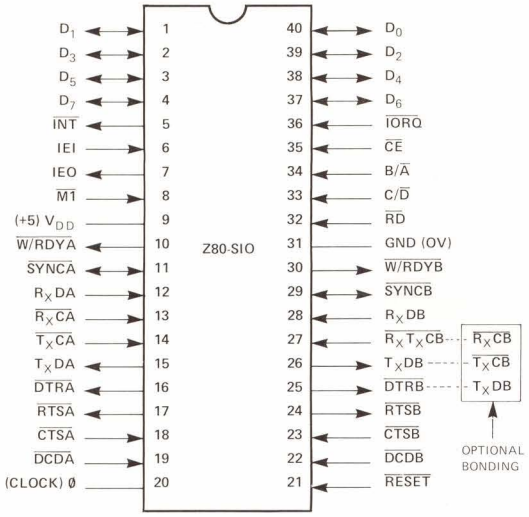
A.C. Characteristics

T_A = 0°C to 70°C, Vcc = +5V ±5%, unless otherwise noted

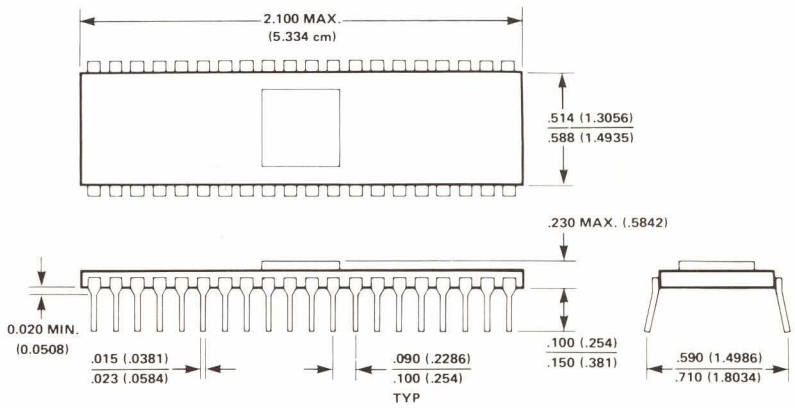
Signal	Symbol	Parameter	Min	Max	Unit	Comments
Φ	$t_C(\Phi)$	Clock Period	400	2000	nsec	
	$t_W(\Phi H)$	Clock Pulse Width, Clock High	170	2000	nsec	
	$t_W(\Phi L)$	Clock Pulse Width, Clock Low	170	2000	nsec	
	t_r, t_f	Clock Rise and Fall Times	-0-	30	nsec	
$\overline{CE}$,B/$\overline{A}$	t_H(CS)	Control Signal hold time from Rising Edge of Φ	-0-		nsec	NOTE 1
C/$\overline{D}$,$\overline{IORQ}$	t_S(CS)	Control Signal setup time from Rising Edge of Φ	160			
	t_{DR}(D)	Data Output Delay from Rising Edge of Φ during Read Cycle		480	nsec	
	$t_S\Phi$(D)	Data Setup Time to Rising Edge of Φ during Write Cycle or $\overline{M1}$ Cycle	50		nsec	
	$t_H\Phi$(D)	Data Hold Time from Rising Edge of Φ during Write Cycle or $\overline{M1}$ Cycle	-0-		nsec	
D_0-D_7	t_{DI}(D)	Data Output Delay from Falling Edge of $\overline{IORQ}$ during INTA Cycle		340	nsec	
	t_{FIM}(D)	Delay to Floating Bus from Rising Edge of $\overline{IORQ}$ during INTA Cycle		230	nsec	
	t_{FR}(D)	Delay to Floating Bus from Rising Edge of $\overline{RD}$ during Read Cycle		230	nsec	
	t_{FI}(D)	Delay to Floating Bus from Falling Edge of IEI during INTA Cycle		230	nsec	
IEO	t_{DL}(IO)	IEO Delay Time from Falling Edge of IEI		150	nsec	
	t_{DH}(IO)	IEO Delay Time from Rising Edge of IEI		250	nsec	
	$t_D\Phi$(IO)	IEO Delay Time from Falling Edge of $\overline{M1}$ (when interrupt occurs just prior to $\overline{M1}$)		300	nsec	
	$t_{SW}\Phi$(M1)	$\overline{M1}$ Setup Time to Rising Edge of Φ during Read or Write Cycle	210		nsec	
$\overline{M1}$	$t_{SR}\Phi$(M1)	$\overline{M1}$ Setup Time to Rising Edge of Φ during INTA or $\overline{M1}$ Cycle	210		nsec	
	$t_H\Phi$(M1)	$\overline{M1}$ Hold Time from Rising Edge of Φ	-0-		nsec	
$\overline{RD}$	$t_{SW}\Phi$(RD)	$\overline{RD}$ Setup Time to Rising Edge of Φ during Write or INTA Cycle	240		nsec	
	$t_H\Phi$(RD)	$\overline{RD}$ Hold Time from Rising Edge of Φ during INTA Cycle	-0-		nsec	
	$t_{SR}\Phi$(RD)	$\overline{RD}$ Setup Time to Rising Edge of Φ during Read or $\overline{M1}$ Cycle	240		nsec	
	$t_{HW}\Phi$(RD)	$\overline{RD}$ Hold Time from Rising Edge of Φ during Write Cycle	-0-		nsec	
	$t_{HM}\Phi$(RD)	$\overline{RD}$ Hold Time from Rising Edge of Φ during $\overline{M1}$ Cycle	-0-		nsec	
$\overline{INT}$	t_{DRx}(IT)	$\overline{INT}$ Delay Time from center of Receive Data Bit	10	13	ΦPeriods	
	t_{DTx}(IT)	$\overline{INT}$ Delay Time from center of Transmit Data Bit	5	9	ΦPeriods	
	$t_D\Phi$(IT)	$\overline{INT}$ Delay Time from Rising Edge of Φ		200	nsec	
$\overline{WAIT/READY}$	t_DIC(W/R)	$\overline{WAIT/READY}$ Delay Time from $\overline{IORQ}$ or $\overline{CE}$ in WAIT Mode		180	nsec	
	$t_DH\Phi$(W/R)	$\overline{WAIT/READY}$ Delay Time from Falling Edge of Φ, $\overline{WAIT/READY}$ HIGH, WAIT Mode		150	nsec	
	t_DRx(W/R)	$\overline{WAIT/READY}$ Delay Time from center of Receive Data Bit, Ready Mode	10	13	ΦPeriods	
	t_DTx(W/R)	$\overline{WAIT/READY}$ Delay Time from center of Transmit Data bit, Ready Mode	5	9	ΦPeriods	
	$t_DL\Phi$(W/R)	$\overline{WAIT/READY}$ Delay from Rising Edge of Φ, $\overline{WAIT/READY}$, Low, Ready Mode		120	nsec	
$\overline{CTSA}$,$\overline{CTSB}$ $\overline{DCDA}$,$\overline{DCDB}$, $\overline{SYNCA}$,$\overline{SYNCB}$	t_W(PH)	Minimum High Pulse Width for latching states into register and generating interrupt	200		nsec	
	t_W(PL)	Minimum Low Pulse Width for latching state into register and generating interrupt	200		nsec	
$\overline{SYNCA}$,$\overline{SYNCB}$	t_{DL}(SY)	Sync Pulse Delay Time from Center of Receive Data Bit, Output Modes	4	7	ΦPeriods	
	t_{SL}(SY)	Sync Pulse Setup Time to Rising Edge of RxC, External Sync Mode	100		nsec	
	t_W(SY)	Sync Pulse Width to Start Character Assembly	3		ΦPeriods	
	t_C(TxC)	Transmit Clock Period	400	∞	nsec	
$\overline{TxCA}$,$\overline{TxCB}$	t_W(TCH)	Transmit Clock Pulse Width, Clock High	180	∞	nsec	NOTE 2
	t_W(TCL)	Transmit Clock Pulse Width, Clock Low	180	∞		
TxDA,TxDB	t_D(TxD)	TxD Output Delay from Falling Edge of $\overline{TxC}$ (x1 Clock Mode)		400	nsec	
$\overline{RxCA}$,$\overline{RxCB}$	t_C(RxC)	Receive Clock Period	400	∞	nsec	
	t_W(RCH)	Receive Clock Pulse Width, Clock High	180	∞	nsec	NOTE 3
	t_W(RCL)	Receive Clock Pulse Width, Clock Low	180	∞	nsec	

NOTE 1: If WAIT is to be used, $\overline{CE}$, $\overline{IORQ}$, C/$\overline{D}$ and $\overline{M1}$ must be valid for as long as WAIT condition is to persist.
NOTE 2: In all modes, maximum data rate must be less than $\frac{1}{4.5}$ of system clock (Φ) rate.
NOTE 3: The RESET signal must be active a minimum of one complete Φ cycle.

Package Configuration



Package Outline



*Dimensions for metric system are in parentheses

Ordering Information

C – Ceramic
P – Plastic
S – Standard 5V±5%, 0° to 70°C
E – Extended 5V±5%, -40° to 85°C
M – Military 5V±10%, -55° to 125°C

Example:

Z80-SIO/1 CS (Ceramic–Standard Range)
Z80-SIO/0 PS (Plastic–Standard Range)

ZILOG Z80 MICROCOMPUTER SYSTEM COMPONENT FAMILY

• Z80, Z80A-CPU	CENTRAL PROCESSOR UNIT
• Z80, Z80A-PIO	PARALLEL I/O
• Z80, Z80A-CTC	COUNTER/TIMER CIRCUIT
• Z80, Z80A-DMA	DIRECT MEMORY ACCESS
• Z80, Z80A-SIO	SERIAL I/O
• Z6104	4K x 1 STATIC RAM
• Z6116	16K x 1 DYNAMIC RAM

ZILOG U.S. DISTRIBUTORS

EASTERN

Hallmark Electronics
4739 Commercial Drive
Huntsville, AL 35805
TEL 205 837 8700
TWX 810 726 2187

Hallmark Electronics
1302 West McNab Road
Fort Lauderdale, FL 33309
TEL 305 971 9280
TWX 510 956 9720

Hallmark Electronics
7233 Lake Ellenor Drive
Orlando, FL 32809
TEL 305 855 4020
TWX 810 850 0183

Hallmark Electronics
3355 Amberton Drive
Baltimore MD 21227
TEL 301 796 9300
TWX 710 862 1942

Hallmark Electronics
1208 Front Street
Building K
Raleigh, NC 27609
TEL 919 832 4465
TWX 510 928 1831

Hallmark Electronics
Pike Industrial Park
Huntington Valley, PA 19006
TEL 215 355 7300
TWX 510 667 1750

Summit
916 Main Street
Buffalo, NY 14202
TEL 716 884 3450

Wilshire Electronics
2554 State Street
Hamden, CT 06517
TEL 203 281 1166
TWX 800 922 1734

Wilshire Electronics
1855 New Highway
Farmingdale, LI, NY 11735
TEL 516 293 5775
TWX 212 895 8707

Wilshire Electronics
One Wilshire Road
Burlington, MA 01803
TEL 617 272 8200
TWX 710 332 6359

Wilshire Electronics
1111 Paulison Avenue
Clifton, NJ 07015
TEL 201 340 1900
TWX 710 989 7052

MIDWESTERN

Hallmark Electronics
180 Grossen Avenue
Elk Grove Village, IL 60076
TEL 312 437 8800
TWX 910 223 3645

Hallmark Electronics
11870 West 91st Street
Congleton Industrial Park
Shawnee Mission, KS 66214
TEL 913 888 4747
TWX 910 749 6620

Hallmark Electronics
9201 Penn Avenue South
Suite 10
Bloomington, MN 55435
TEL 612 884 9056
TWX 910 576 3187

Hallmark Electronics
13789 Rider Trail
Earth City, MO 63045
TEL 314 291 5350
TWX 910 760 0671

Hallmark Electronics
6969 Worthington Galena Road
Worthington, OH 43085
TEL 614 846 1882

Hallmark Electronics
4846 S. 83rd E. Avenue
Tulsa, OK 74145
TEL 918 835 8458
TWX 910 845 2290

Hallmark Electronics
3100-A Industrial Terrace
Austin, TX 78758
TEL 512 837 2841
TWX 910 874 2031

Hallmark Electronics
9333 Forest Lane
Dallas, TX 75231
TEL 214 234 7300
TWX 910 867 4721

Hallmark Electronics
8000 Westglen
Houston, TX 77063
TEL 713 781 6100
TWX 910 881 2711

Hallmark Electronics
237 South Curtis
West Allis, WI 53214
TEL 414 476 1270
TWX 910 262 3186

RM Electronics
4860 South Division
Kentwood, MI 49508
TEL 616 531 9300
TWX 810 273 8779

RM Electronics
47 Chestnut Lane
Westmont, Illinois 60559
TEL 312 323 9670

MOUNTAIN

Century Electronics
121 Elizabeth, NE
Albuquerque, NM 87123
TEL 505 292 2700
TWX 910 989 0625

WESTERN

Intermark Electronics
1802 E. Carnegie Avenue
Santa Ana, CA 92705
TEL 714 540 1322
TWX 910 595 1583

Intermark Electronics
4040 Sorrento Valley Blvd.
San Diego, CA 92121
TEL 714 279 5200
714 453 9005
TWX 910 335 1515

Intermark Electronics
1020 Stewart Drive
Sunnyvale, CA 94086
TEL 408 738 1111
TWX 910 339 9312

R.V. Weatherford Co.
6921 San Fernando Road
Glendale, CA 91201
TEL 213 849 3451
TWX 910 498 2223

R.V. Weatherford Co.
1550 Babbitt Avenue
Anaheim, CA 92805
TEL 714 634 9600
TWX 910 593 1334

R.V. Weatherford Co.
1095 East Third Street
Pomona, CA 91765
TEL 714 623 1261
TWX 910 581 3811

R.V. Weatherford Co.
3240 Hillview Avenue
Stanford Industrial Park
Palo Alto, CA 94304
TEL 415 493 5373

R.V. Weatherford Co.
3311 W. Earll Drive
Phoenix, AZ 85017
TEL 602 272 7144
TWX 910 951 0636

Sterling Electronics
5608 6th Avenue South
Seattle, WA 98108
TEL 206 762 9100
TLX 32-9652

Western Microtechnology
977 Benicia Avenue
Sunnyvale, CA 94086
TEL 408 737 1660

CANADA

Future Electronics
5647 Ferrier Street
Montreal, Quebec,
CANADA H4P 2K5
TEL 514 735 5775
TWX 610 421 3251

ZILOG REGIONAL SALES OFFICES

EASTERN REGION

Zilog, Inc.
76 Treble Cove Road
No. Billerica, MA 01862
TEL 617 667 2179
TWX 710 347 6660

MIDATLANTIC REGION

Zilog, Inc.
P.O. Box 92
Bergenfield, NJ 07625
TEL 201 385 9158
TWX 710 991 9771

MIDWESTERN REGION

Zilog, Inc.
1701 Woodfield Place
Suite 417
Schaumburg, IL 60195
TEL 312 885 8080
TWX 910 291 1064

SOUTHWESTERN

Zilog, Inc.
17982 Sky Park Ci
Suite C
Irvine, CA 92714
TEL 714 549 2891
TWX 910 595 280

10460 Bubb Road, Cupertino, California 95014 Telephone (408) 446-4666 TWX 910-338-7621

Zilog

03-0039-02

Printed in U.S.A.