Z80382/Z8L382

Z80382 Data Communications Processor

Preliminary Product Specification

PS006702-1201

ZiLOG Worldwide Headquarters • 910 E. Hamilton Avenue • Campbell, CA 95008
Telephone: 408.558.8500 • Fax: 408.558.8300 • www.ZiLOG.com

This publication is subject to replacement by a later edition. To determine whether a later edition exists, or to request copies of publications, contact:

ZiLOG Worldwide Headquarters
910 E. Hamilton Avenue
Campbell, CA 95008
Telephone: 408.558.8500
Fax: 408.558.8300
www.ZiLOG.com

Document Disclaimer

Manual Conventions

The following conventions have been adopted to provide clarity and ease of use:

- `Courier Regular 10-point` highlights the following items
  - Bit
  - Software code
  - File names and paths
  - Hexadecimal value

Table of Contents

List of Figures

List of Tables

Z80382, Z8L382 Data Communications Processor

Features

- Embedded Z380™ Microprocessor
  - Maintains Object Code Compatibility with Z80® and Z180™ Microprocessors
  - Enhanced Instruction Set for 16-Bit Operation
  - 16 MB Linear Addressing
  - Two Clock Cycle Instruction Execution Minimum
  - Four On-Chip Register Banks
  - BC/DE/HL/IX/IY Augmented to 32 Bits
  - Clock Divide-by-Two and Multiply-by-Two Options
  - Fully Static CMOS Design with Low-Power STANDBY Mode
  - 16-Bit Internal Bus
  - Dynamic Bus Sizing (8/16-Bit Inter-Operability)
- 16550 MIMIC with I/O Mailbox, DMA Mailbox, and 16 mA Bus Drive
- Three HDLC Synchronous Serial Channels
  - Serial Data Rate of up to 10 Mbps
- GCI/SCIT Bus Interface
- Eight Advanced DMA Channels with 24-Bit Addressing
- Plug-and-Play ISA Interface
- PCMCIA Interface
- Two Enhanced ASCIs (UARTs) with 16-Bit Baud Rate Generators (BRG)
- Clocked Serial I/O Channel (CSIO) for Use with Serial Memory
- Two 16-Bit Timers with Flexible Prescalers
- Three Memory Chip Selects with Wait-State Generators
- Watch-Dog Timer (WDT)

- Up to 32 General-Purpose I/O Pins
- DC to 20 MHz Operating Frequency @ 5.0V
- DC to 10 MHz Operating Frequency @ 3.3V
- 144-Pin QFP and VQFP Style Packages

General Description

The Z80382 (Z382) is designed to address high-end data communication applications such as digital modems (ISDN, GSM, Mobitex & Modacom), xDSL and analog modems (V.34 and beyond). The Z382 provides a performance upgrade to existing Z80- and Z18x-based designs by utilizing the increased bandwidth of the 380C processor. The Z8L382 is a low voltage version of the device.

Notes: In this document the notation 380C denotes the Z380-compatible CPU core which is embedded in the Z382.

An overline ($\overline{\ }$) above a signal name indicates that the signal is asserted in the Low state and negated in the High state.

The 380C microprocessor is a high-performance processor with fast and efficient throughput and increased memory addressing capabilities. The 380C offers a continuing growth path for present Z80- or Z18x-based designs, while maintaining Z80 and Z180 object code compatibility. Its enhancements include added instructions, expanded 16 MB address space and flexible bus interface timing.

In the 380C, the basic addressing modes of the Z80 microprocessor have been augmented to include Stack Pointer Relative loads and stores, 16-bit and 24-bit indexed offsets, and more flexible Indirect Register addressing. Internally, all of the addressing modes allow up to 32-bit linear addressing; however, the Z382 has 24 address pins, therefore it can address a maximum of 16 MB of memory.

Other additions to the instruction set include a full complement of 16-bit arithmetic and logical operations, 16-bit I/O operations, multiply and divide, and a complete set of register-to-register loads and exchanges.

The 380C register file includes alternate versions of the IX and IY registers. There are four banks of registers in the 380C, along with instructions for switching among them. All of the 16-bit register pairs and index registers in the basic Z80 microprocessor register file are expanded to 32 bits.

The Z382 includes dynamic bus sizing to allow any mix of 8- and 16-bit memory, and I/O devices in a system. One application for this capability would be to copy code from a low-cost, slow 8-bit ROM to 16-bit RAM, from which it can be exe-

cuted at much higher speeds. Memory bus sizes can be configured internally by software to eliminate the need for external logic to drive MSIZE.

Some features that have traditionally been handled by external peripherals have been incorporated in the Z382. These on-chip peripherals reduce system chip count and interconnections on the external bus. These peripherals, illustrated in the Z382 Block Diagram in Figure 1, are summarized below.

HDLC Synchronous Channels. Three HDLC channels operate at serial data rates of up to 10 Mbps and feature 8-byte receive and transmit FIFOs. These can be used for modems, general data communications, and ISDN. The ISDN can be handled separately or through the GCI/SCIT bus interface. HDLC Channels always transfer data through the DMA channels. A transparent mode is selectable. Two of the HDLC cells can be pin multiplexed with the ASCIs (UARTs) to provide dynamically switchable (async-sync) DTE interfaces.

DMA Channels. The eight DMA channels provide 24-bit memory addressing and can transfer memory block sizes of up to 64 KB (16-bits) word. These DMA channels can be dynamically assigned to serve the HDLC ports, MIMIC COM port, Host DMA Mailbox, or ASCIs in any mixture. Linked list operation allows all HDLC transmitters and receivers to operate at or above T1/E1 rates simultaneously without loading the bus bandwidth.

16550 MIMIC. Provides connection to a PC ISA bus and emulation of the 16550 UART register set. Improvements include 16 mA output drivers and internal COM port address decoding to reduce external PC interface components.

ASCI. Two flexible asynchronous serial channels with baud rate generators, modem control and status.

CSIO. A clocked serial I/O channel which can be used for serial memory interface.

Timers. Two 16-bit counter/timers with flexible prescalers for wide-range timing applications.

GCI/SCIT Bus Interface. A common interface to ISDN interface devices. Internal signals from this module can be connected to the HDLC channels to provide two B-channels and one D-channel for ISDN.

Plug-and-Play ISA Interface. Provides auto-configuration in ISA (AT bus) applications.

PCMCIA Interface. Provides connectivity to a PCMCIA bus.

32-Bit General-Purpose I/O. For non-PC add-in applications, four 8-bit ports are provided for general- purpose I/O. In ISA or PCMCIA applications, the pins from two of the ports are reallocated to host bus signals and are not available. Pins from the other two ports are selectively multiplexed with on-chip peripheral functions (ASCIs, CSIO, PRT). These pins are individually programmable for input/output mode.

I/O Chip Selects. Two I/O chip selects are provided to support I/O access of external peripherals. Each has a programmable base address and provides I/O decode sizes ranging from 8 to 512 bytes.

ROM/RAM Chip Selects with Wait-State Generators. Chip select outputs are provided to decode memory addresses and provide memory chip enables. Each chip select has its own Wait State Generator to allow use of memories with different speeds.

Watch-Dog Timer. A Watch-Dog Timer (WDT) with a wide range of time-constants prevents code runaway and possible resulting system damage. The RESET input can be forced as an output upon the terminal count of the WDT. This allows external peripherals to be reset along with the Z382

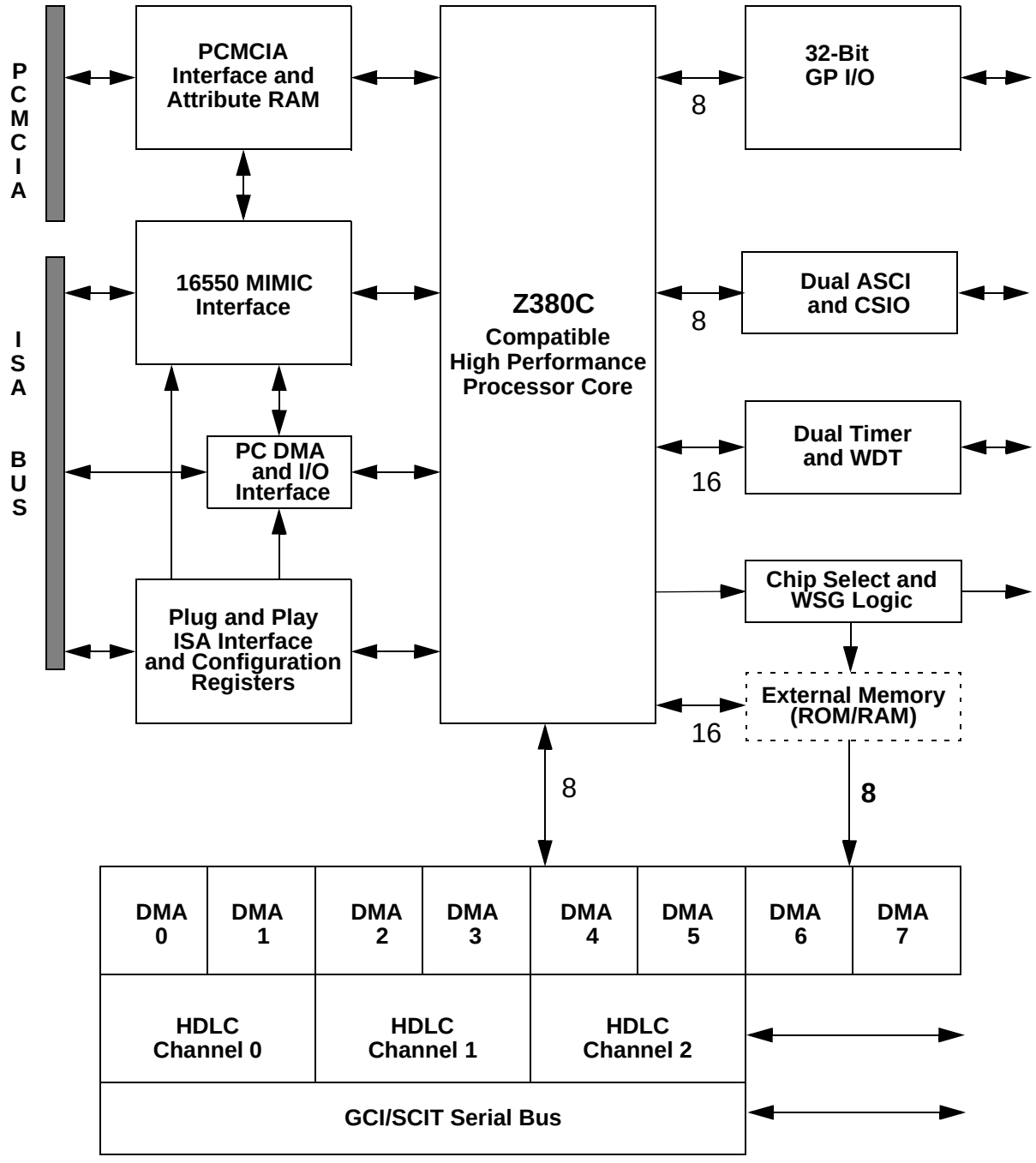


Figure 1. Z80382 Block Diagram

Z80382 Pin Description

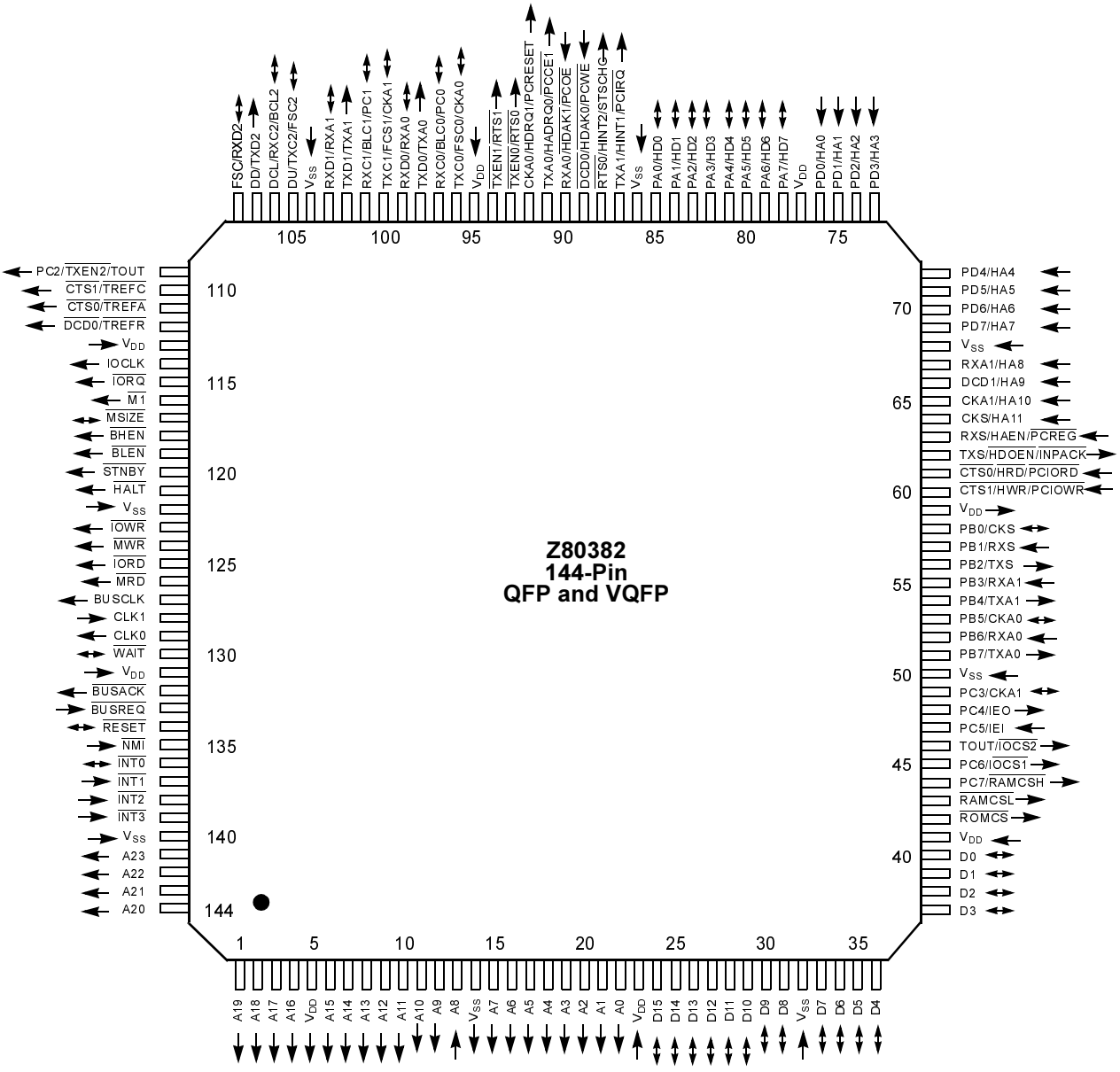


Figure 2. Z80382 144-Pin QFP and VQFP Pin Description

Absolute Maximum Ratings

Symbol	Description	Value	Unit
V_{DD}	Supply Voltage	-0.3 to +7.0	V
V_{IN}	Input Voltage	-0.3 to V_{DD}+0.3	V
T_{OPR}	Operating Temp	0 to +70	°C
T_{STG}	Storage Temp	-55 to +150	°C

Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This rating is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Standard Test Conditions

The DC Characteristics which follow apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND (0V). Positive current flows into the referenced pin (Figure 3, Test Load Diagram).

- Operating temperature range:
  - Standard: 0°C to 70°C
- Voltage Supply Range:
  - +4.5V ≤ V_{DD} ≤ +5.5V (Z80382 versions)
  - +3.0V ≤ V_{DD} ≤ +3.6V (Z8L382 versions)

All AC parameters assume a load capacitance of 50 pF. Add 10 ns delay for each 50 pF increase in load up to a maximum of 150 pF for the data bus and 100 pF for address and control lines. AC timing measurements are referenced to 1.5 volts (except for clock, which is referenced to the 10% and 90% points). Maximum capacitive load for PHI is 125 pF.

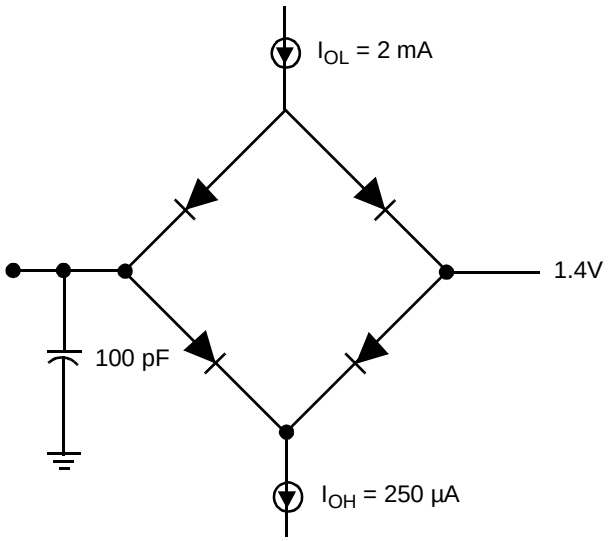


Figure 3. Test Load Diagram

DC Characteristics

Pin Numbers and Input/Output Classifications

Table 1. Pin Numbers and Input/Output Classifications

Pin	Input Class[1]	Output Class[1]	Pin Number(s)
$\overline{\text{BHEN}}$		O	118
$\overline{\text{BLEN}}$		O	119
$\overline{\text{BUSACK}}$		O	132
$\overline{\text{BUSREQ}}$	I		133
$\overline{\text{CTS0}}$/$\overline{\text{HRD}}$/$\overline{\text{PCIORD}}$	I		61
$\overline{\text{CTS0}}$/$\overline{\text{TREFA}}$	I	O	111
$\overline{\text{CTS1}}$/$\overline{\text{HWR}}$/$\overline{\text{PCIOWR}}$	I		60
$\overline{\text{CTS1}}$/$\overline{\text{TREFC}}$	I	O	110
$\overline{\text{DCD0}}$/$\overline{\text{HDAK0}}$/$\overline{\text{PCWE}}$	I		89
$\overline{\text{DCD0}}$/$\overline{\text{TREFR}}$	I	O	112
$\overline{\text{DCD1}}$/HA9	I		66
$\overline{\text{HALT}}$		O	121

Table 1. Pin Numbers and Input/Output Classifications (Continued)

Pin	Input Class[1]	Output Class[1]	Pin Number(s)
$\overline{\text{INT0-3}}$	R		136 - 139
$\overline{\text{IORD}}$	I	3	125
$\overline{\text{IORQ}}$	I	3	115
$\overline{\text{IOWR}}$	I	3	123
$\overline{\text{M1}}$	I	3	116
$\overline{\text{MRD}}$	I	3	126
$\overline{\text{MSIZE}}$	I	D	117
$\overline{\text{MWR}}$	I	3	124
$\overline{\text{NMI}}$	R		135
$\overline{\text{RAMCSL}}$		O	43
$\overline{\text{RESET}}$	R	D	134
$\overline{\text{ROMCS}}$	O		42
$\overline{\text{RTS0}}$/HINT2/STSCHG		H	88
$\overline{\text{STNBY}}$		O	120
$\overline{\text{TXEN1-0}}$/$\overline{\text{RTS1-0}}$		O	94 -93
$\overline{\text{WAIT}}$	I	D	130
A23-0	I	3	141 - 144, 1 - 4, 6 - 13, 15 - 22
BUSCLK		H	127
CKA0/HDRQ1/PCRESET	I	H	92
CKA1/HA10	I	3	65
CKS/HA11	I	3	64
CLKI	R		128
CLKO		O	129
D15-0	I	3	24 - 31, 33 - 40
DCL/RXC2/BCL2	I		106
DD/TXD2	I	D (DD) O (TXD2)	107
DU/TXC2/FSC2	I	D (DU) O (TXC2, FSC2)	105
FSC/RXD2	I		108

Table 1. Pin Numbers and Input/Output Classifications (Continued)

Pin	Input Class[1]	Output Class[1]	Pin Number(s)
IOCLK	I	O	114
PA7-0/HD7-0	I	H	78 -85
PB0/CKS	I	3	58
PB1/RXS	i	3	57
PB2/TXS	I	3	56
PB3/RXA1	I	3	55
PB4/TXA1	I	3	54
PB5/CKA0	I	3	53
PB6/RXA0	I	3	52
PB7/TXA0	I	3	51
PC2/$\overline{\text{TXEN2}}$/TOUT	I	3	109
PC3/CKA1	I	3	49
PC4/IEO	I	3	48
PC5/IEI	I	3	47
PC6/$\overline{\text{IOCS1}}$	I	3	45
PC7/$\overline{\text{RAMCSH}}$	I	3	44
PD7-0/HA7-0	I	3	69 - 76
RXA0/$\overline{\text{HDAK1}}$/$\overline{\text{PCOE}}$	I	D	90
RXA1/HA8	I	3	67
RXC1-0/BCL1-0/PC1-0	I	3	101, 97
RXD1-0/RXA1-0	I	O	103, 99
RXS/HAEN/$\overline{\text{PCREG}}$	I	D	63
TOUT/$\overline{\text{IOCS2}}$		3	46
TXA0/HDRQ0/$\overline{\text{PCCE1}}$	I	H	91
TXA1/HINT1/$\overline{\text{PCIRQ}}$		H	87
TXC1-0/FSC1-0/CKA1-0	I	3	100, 96
TXD1-0/TXA1-0		O	102, 98
TXS/$\overline{\text{HDOEN}}$/$\overline{\text{INPACK}}$		O	62

Table 1. Pin Numbers and Input/Output Classifications (Continued)

Pin	Input Class[1]	Output Class[1]	Pin Number(s)
V_{DD}			5, 23, 41, 59, 77, 95, 113, 131
V_{SS}			14, 32, 50, 68, 86, 104, 122, 140

Note: 1. Characteristics of each pin are listed in terms of the classifications in the DC Characteristics Table 2 and Table 3 which follow.

Specifications apply over Standard Operating Conditions unless otherwise noted.

Table 2. Output Class Characteristics

Output Class(1)	Type	V_{OL} Max.	V_{OH} Min.	C_{OUT} Max.(2)
O	Totem Pole	0.4V @ I_{OL} = 2.0 mA	V_{DD} - 1.2V @ I_{OH} = 200 µA	15 pF
		Slew Rate = 0.33 V/ns min @ C_{LOAD} = 50 pF		
3	3-State	0.4V @ I_{OL} = 2.0 ma	V_{DD} - 1.2V @ I_{OH} = 200 µA	
		Slew Rate = 0.33 V/ns min @ C_{LOAD} = 50 pF		
H	High Drive 3-State	0.4V max @ I_{OL} = 16 mA, V_{DD} = 5V	2.4V min @ I_{OH} = 5mA, V_{DD} = 5V	15 pF
		Slew Rate = 0.33 V/ns min @ C_{LOAD} = 50 pF		
D	Open-Drain	0.4V max @ I_{OL} = 16 mA	--	15 pF

Notes:
1. The Pin Numbers and Input/Output Classifications described in Table 1 identifies the specific output pins in each class.
2. Applies to Output only or I/O.

Table 3. Input Class Characteristics

Input Class[1]	V_{IL} Max. (Z80382)	V_{IL} Max. (Z8L382)	V_{IH} Min. (Z80382)	V_{IH} Min. (Z8L382)	Minimum Hysteresis
I	0.8V	0.6V	2.0V	2.0V	0.4V
	I_I = ±10 µA max, V_I = 0 to 5V (includes leakage if I/O) C_{IN} = 5 pF max (if input only, see output type if I/O)				
	Note: Inputs of this type include a weak-latch circuit, except that a register bit can disable those for pins PB7-0.				
R	0.4V	0.4V	V_{DD} - 0.6V	V_{DD} - 0.3V	0.4V
	I_I = ±10 µA max, V_I = 0 to 5V C_{IN} = 5 pF max				
	Note: Inputs of this type except CLKI include a weak-latch circuit.				

Note: The Pin Numbers and Input/Output Classifications described in Table 1 identifies the specific input pins in each class.

Table 4. DC Electrical Characteristics

Symbol	Item	Condition	Min.	Typ.	Max.	Unit
V_{IH}	Input High Voltage	See Table 3				V
V_{IL}	Input Low Voltage	See Table 3				V
V_{OH}	Output High Voltage	See Table 2				V
V_{OL}	Output Low Voltage	See Table 2				V
I_{IL}	Input Leakage Current, All Inputs Except CLKI, CLKO	V_{IN} = 0.5V to V_{DD} - 0.5V			1.0	µA
I_{TL}	Tristate Leakage Current	V_{IN} = 0.5 to V_{DD} - 0.5			1.0	µA
I_{DD}	V_{DD} Supply Current (1, 3) Normal Operation	BUSCLK = 10 MHz V_{DD} = 3.3V			75	mA
		BUSCLK = 10 MHz V_{DD} = 5V			90	mA
		BUSCLK = 20 MHz V_{DD} = 5V			150	mA
I_{DDS}	V_{DD} Supply Current Standby Mode (1, 2, 3)	BUSCLK = 10 MHz V_{DD} = 3.3V			50	µA
		BUSCLK = 10 MHz V_{DD} = 5V			50	µA
		BUSCLK = 20 MHz V_{DD} = 5V			50	µA

Notes:
1. V_{IH} min = V_{DD}-1.0V, V_{IL} max = 0.8V. All output terminals are at no load.
2. On-chip peripherals with independent clocks are inactive (not being clocked).
3. BUSCLK is the internal processor clock frequency.

AC Characteristics

380C Processor Timing (See Figure 4.)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 5. AC Characteristics

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Unit**
t1	Clock Cycle Time[1]	25	DC	50	DC	ns
t2	Clock High Time[1]	10		20		ns
t3	Clock Low Time[1]	10		20		ns
t4	Clock Rise Time[1]		3		5	ns
t5	Clock Fall Time[1]		3		5	ns
t6	CLKI Low to BUSCLK High Delay		25		35	ns
t7	CLKI High to BUSCLK Low Delay		25		35	ns
t8	BUSCLK High to Output Valid[2]		10		10	ns
t9	BUSCLK Low to Output Valid[3]		10		10	ns
t10	Input Setup to BUSCLK Rise[4]	10		15		ns
t11	Input Hold from BUSCLK High[4]	0		0		ns
t12	$\overline{\text{BUSREQ}}$ Setup to BUSCLK Fall[5]	10		15		ns
t13	$\overline{\text{BUSREQ}}$ Hold from BUSCLK Low[5]	0		0		ns
t14	$\overline{\text{WAIT}}$ Setup to BUSCLK Rise[6]	10		15		ns
t15	$\overline{\text{WAIT}}$ Hold from BUSCLK High[6]	0		0		ns
t16	$\overline{\text{WAIT}}$ Setup to BUSCLK Fall[6]	15		15		ns
t17	$\overline{\text{WAIT}}$ Hold from BUSCLK Low[6]	0		0		ns
t18	$\overline{\text{NMI}}$ Width Low	15		15		ns
t19	$\overline{\text{RESET}}$ Width Low	10		10		t1
t20	$\overline{\text{INT1}}$, $\overline{\text{INT2}}$, $\overline{\text{INT3}}$ Low Width[7]	15		15		ns
t21	$\overline{\text{INT1}}$, $\overline{\text{INT2}}$, $\overline{\text{INT3}}$ High Width[7]	15		15		ns

Table 5. AC Characteristics (Continued)

Symbol	Parameter	Z80382 Min.	Z80382 Max.	Z8L382 Min.	Z8L382 Max.	Unit

Notes:
1. Applies to the oscillator or external clock input. The maximum internal clock frequency (BUSCLK) is limited to 20 MHz for the Z80382 and 10 MHz for the Z8L382. Input clock frequencies greater than these values must use the CLKI/2 mode for creating BUSCLK. This is the default state after Reset.
2. Applies to A23-0, $\overline{BHEN}$, $\overline{BLEN}$, IOCLK, $\overline{IOCS1}$, $\overline{IOCS2}$, $\overline{ROMCS}$, $\overline{RAMCSL}$, $\overline{RAMCSH}$, $\overline{M1}$, $\overline{BUSACK}$, $\overline{MRD}$, $\overline{MWR}$, $\overline{TREFA}$, $\overline{TREFC}$, $\overline{TREFR}$
3. Applies to D15-0, $\overline{HALT}$, $\overline{STNBY}$, $\overline{IORQ}$, $\overline{IORD}$, $\overline{IOWR}$, $\overline{MSIZE}$, $\overline{BUSACK}$, $\overline{MRD}$, $\overline{MWR}$, $\overline{TREFC}$, $\overline{TREFR}$
4. Applicable for Data Bus and $\overline{MSIZE}$ inputs.
5. $\overline{BUSREQ}$ can also be asserted/deasserted asynchronously.
6. External waits asserted at $\overline{WAIT}$ input.
7. In edge-triggered mode.

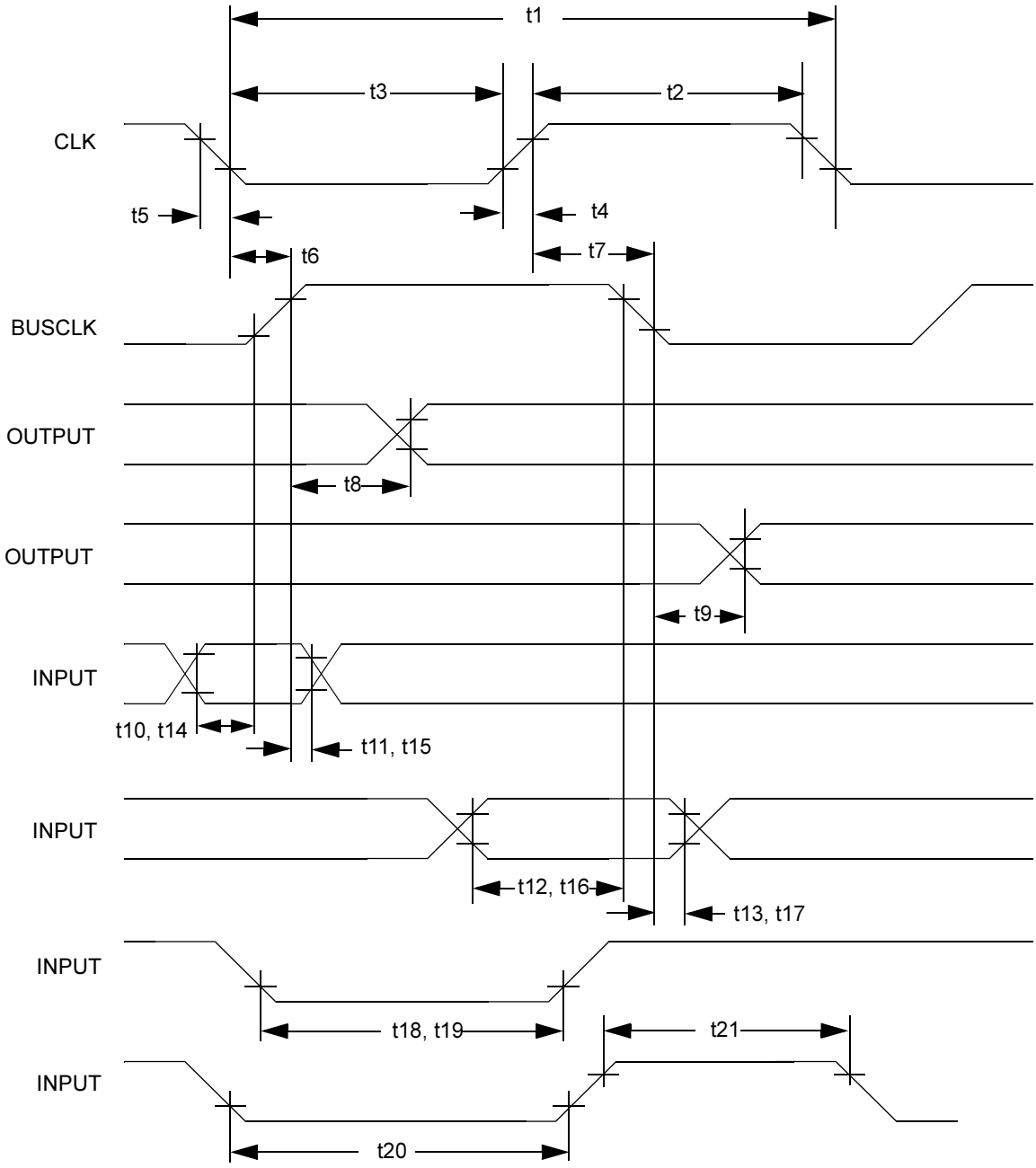


Figure 4. 380C Processor Timing Diagram

Host-PCMCIA Attribute Memory Read Timing (See Figure 5)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 6. Host-PCMCIA Attribute Memory Read Timing

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Unit**
t22	Address Setup to $\overline{\text{PCCE1}}$ Fall	15		20		ns
t23	Address Setup to $\overline{\text{PCOE}}$ Fall	15		20		ns
t24	Address Hold from $\overline{\text{PCCE1}}$ High	5		5		ns
t25	Address Hold from $\overline{\text{PCOE}}$ High	5		5		ns
t26	$\overline{\text{PCCE1}}$ Low to Data Active	0		0		ns
t27	$\overline{\text{PCOE}}$ Low to Data Active	0		0		ns
t28	$\overline{\text{PCCE1}}$ Low to Data Valid		60		60	ns
t29	$\overline{\text{PCOE}}$ Low to Data Valid		60		60	ns
t30	Data Invalid from $\overline{\text{PCCE1}}$ High	5		5		ns
t31	Data Invalid from $\overline{\text{PCOE}}$ High	5		5		ns
t32	Data 3-state from $\overline{\text{PCCE1}}$ High		20		20	ns
t33	Data 3-state from $\overline{\text{PCOE}}$ High		20		20	ns
Note: Timings also apply for reads from registers located in the attribute memory space.						

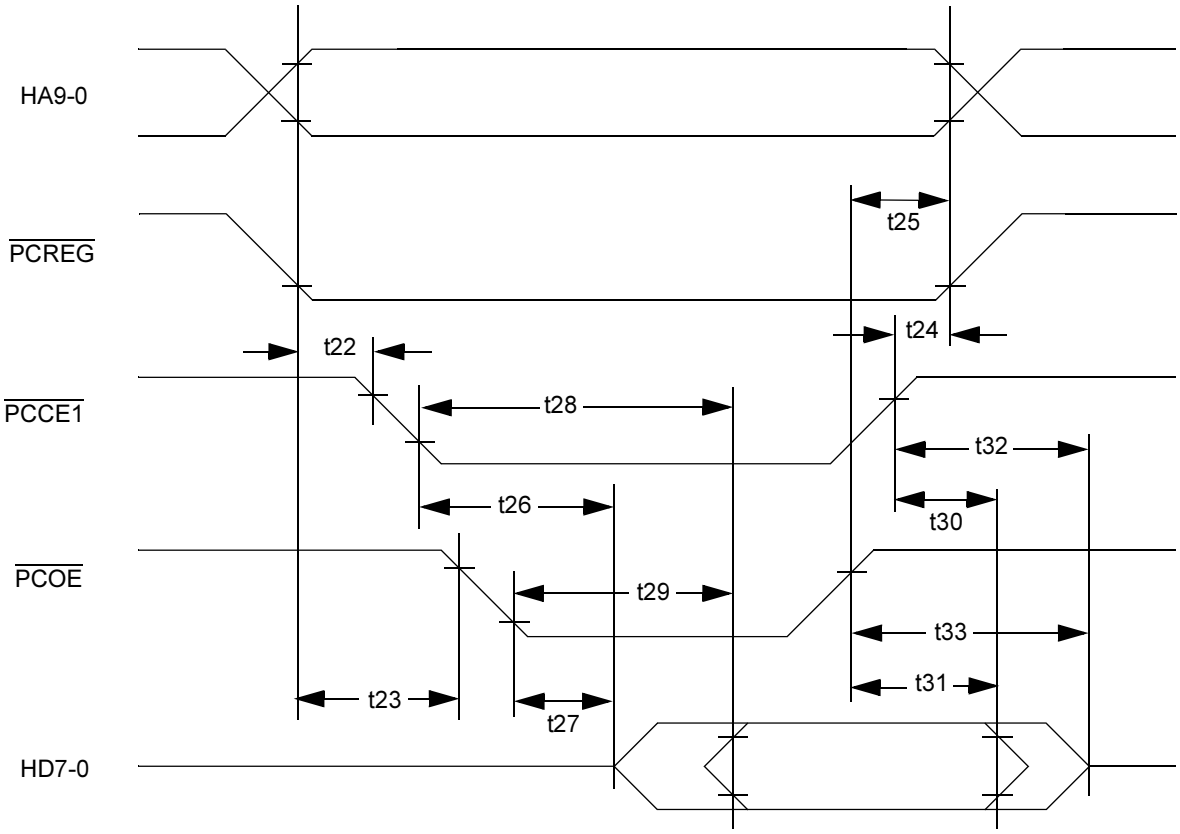


Figure 5. Host - PCMCIA Attribute Memory Read Timing Diagram

Host-PCMCIA Attribute Memory Write Timing (See Figure 6.)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 7. Host-PCMCIA Attribute Memory Write Timing

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Unit**
t34	Address Setup to $\overline{\text{PCCE1}}$ Fall	30		35		ns
t35	Address Setup to $\overline{\text{PCWE}}$ Fall	30		35		ns
t36	Address Hold from $\overline{\text{PCCE1}}$ High	10		10		ns
t37	Address Hold from $\overline{\text{PCWE}}$ High	10		10		ns
t38	Data Setup to $\overline{\text{PCCE1}}$ Rise	20		20		ns
t39	Data Setup to $\overline{\text{PCWE}}$ Rise	20		20		ns
t40	Data Hold from $\overline{\text{PCCE1}}$ High	10		10		ns
t41	Data Hold from $\overline{\text{PCWE}}$ High	10		10		ns
Note: Timings also apply for writes to registers located in the attribute memory space.						

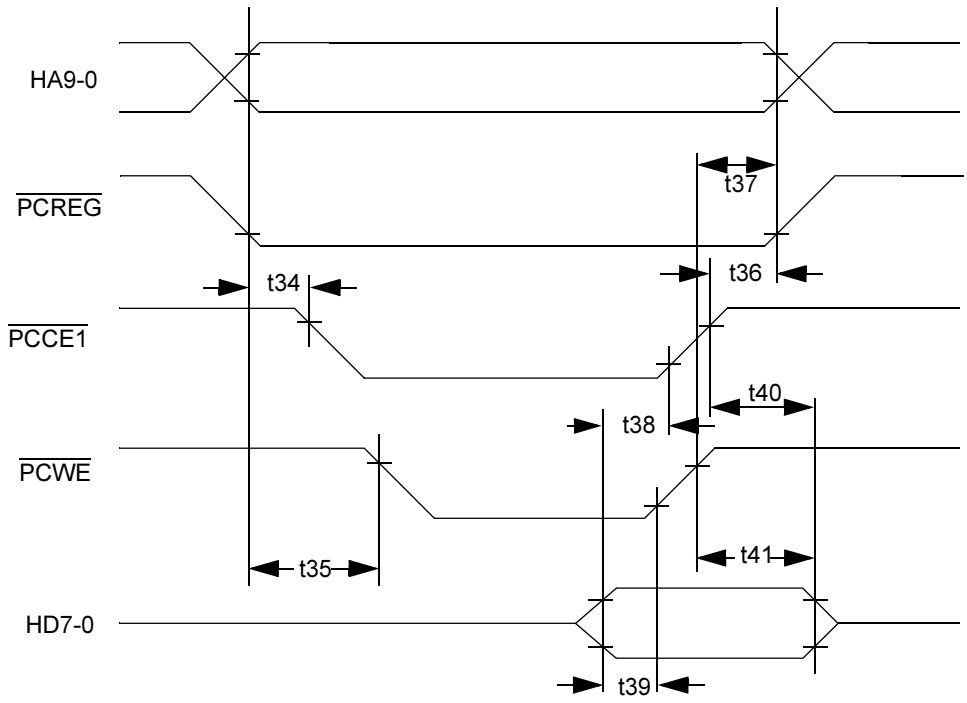


Figure 6. Host - PCMCIA Attribute Memory Write Timing Diagram

Host-PCMCIA I/O Read Timing (See Figure 7)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 8. Host-PCMCIA I/O Read

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t42	Address Setup to $\overline{\text{PCIORD}}$ Fall	50		50		ns
t43	$\overline{\text{PCREG}}$ Setup to $\overline{\text{PCIORD}}$ Fall	5		5		ns
t44	$\overline{\text{PCCE1}}$ Setup to $\overline{\text{PCIORD}}$ Fall	5		5		ns
t45	$\overline{\text{PCIORD}}$ Low Width	125		125		ns
t46	$\overline{\text{INPACK}}$ Low from $\overline{\text{PCIORD}}$ Low		10		10	ns
t47	Data Valid from $\overline{\text{PCIORD}}$ Low		50		50	ns
t48	Data Invalid from $\overline{\text{PCIORD}}$ High	5		5		ns
t49	$\overline{\text{INPACK}}$ High from $\overline{\text{PCIORD}}$ High		10		10	ns
t50	$\overline{\text{PCCE1}}$ Hold from $\overline{\text{PCIORD}}$ High	10		10		ns
t51	$\overline{\text{PCREG}}$ Hold from $\overline{\text{PCIORD}}$ High	10		10		ns
t52	Address Hold from $\overline{\text{PCIORD}}$ High	10		10		ns
t53	$\overline{\text{PCIORD}}$ Low to Data Active	0		0		ns
t54	Data 3-state from $\overline{\text{PCIORD}}$ High		20		20	ns

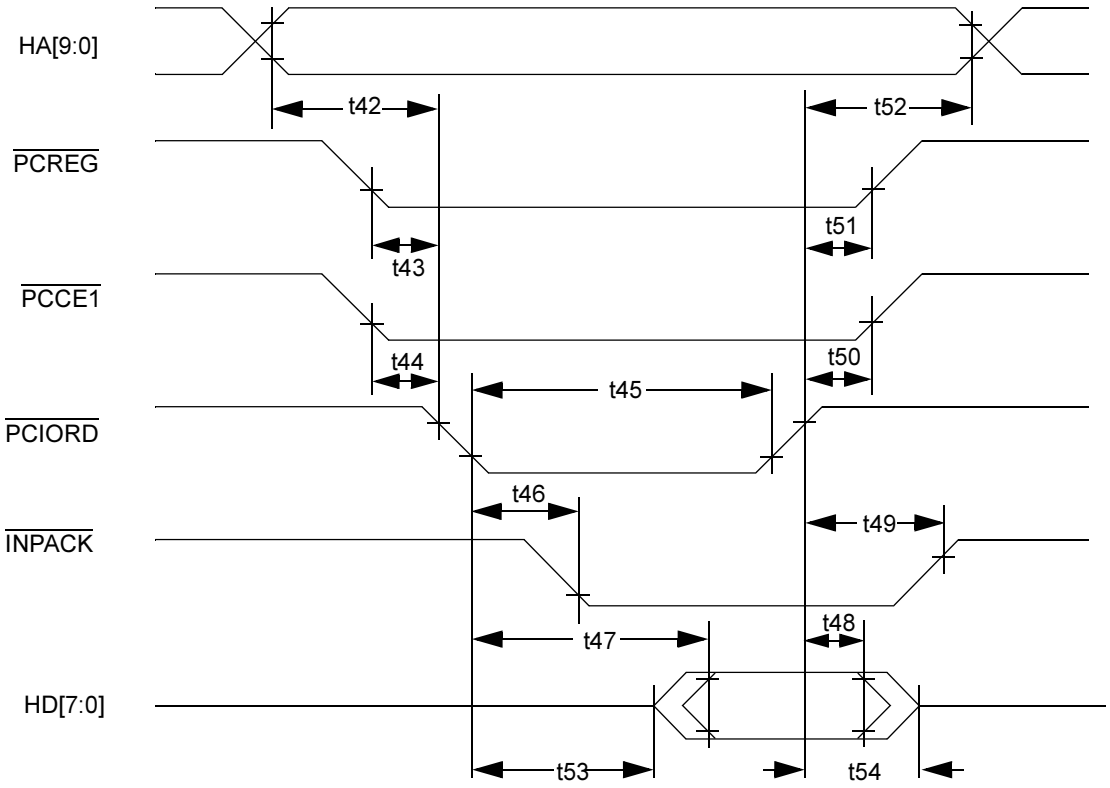


Figure 7. Host - PCMCIA I/O Read Timing Diagram

Host-PCMCIA I/O Write Timing (See Figure 8)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 9. Host-PCMCIA I/O Write Timing

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t55	Address Setup to $\overline{\text{PCIOWR}}$ Fall	50		50		ns
t56	$\overline{\text{PCREG}}$ Setup to $\overline{\text{PCIOWR}}$ Fall	5		5		ns
t57	$\overline{\text{PCCE1}}$ Setup to $\overline{\text{PCIOWR}}$ Fall	5		5		ns
t58	$\overline{\text{PCIOWR}}$ Low Width	125		125		ns
t59	Data Setup to $\overline{\text{PCIOWR}}$ Rise	35		35		ns
t60	Data Hold from $\overline{\text{PCIOWR}}$ High	20		20		ns
t61	$\overline{\text{PCCE1}}$ Hold from $\overline{\text{PCIOWR}}$ High	10		10		ns
t62	$\overline{\text{PCREG}}$ Hold from $\overline{\text{PCIOWR}}$ High	10		10		ns
t63	Address Hold from $\overline{\text{PCIOWR}}$ High	10		10		ns

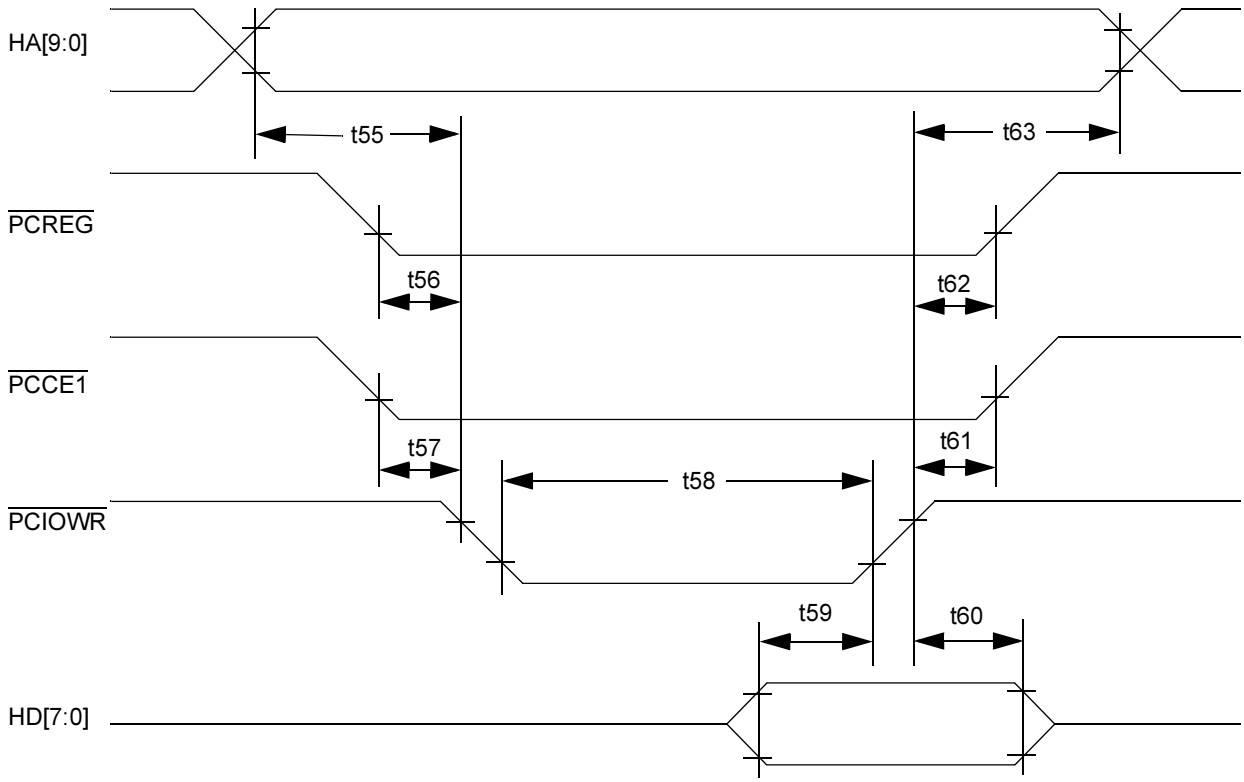


Figure 8. Host - PCMCIA I/O Write Timing Diagram

Timer Output Timing (See Figure 9)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 10. Timer Output Timing

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t64	BUSCLK Low to TOUT Valid		20		20	ns

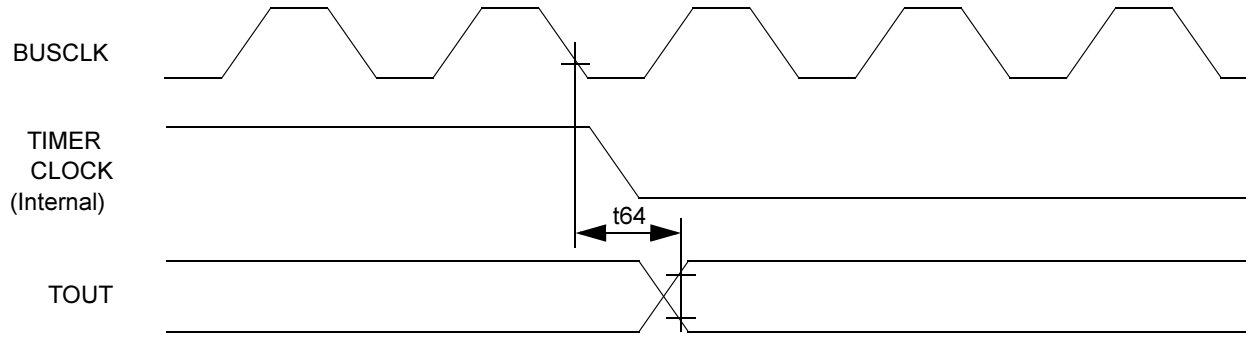


Figure 9. Timer Output Timing Diagram

CSIO Receive/Transmit Timing (See Figure 10)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 11. CSIO Receive/Transmit Timing

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t66	CKS Low to TXS Valid		40		40	ns
t67	RXS Setup to CKS Rise	20		20		ns
t68	RXS Hold from CKS High	5		5		ns
t71	CKS External Clock Period	50		50		ns

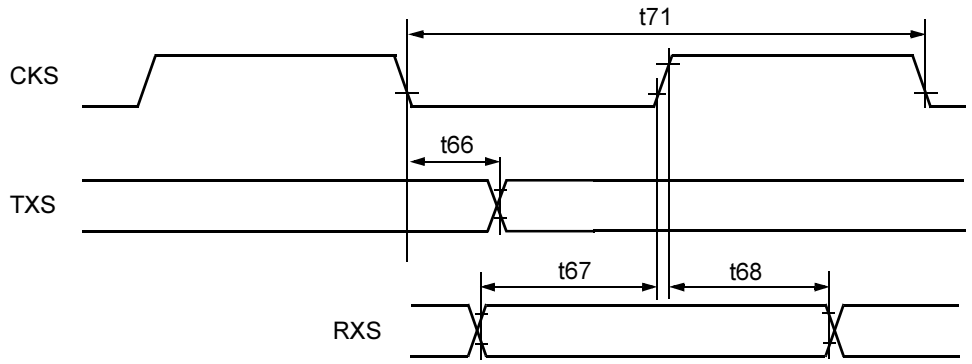


Figure 10. CSIO Receive/Transmit Timing Diagram

ASCI Transmitter Timing (See Figure 11)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 12. ASCI Transmitter Timing

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t72	External Transmit Clock Period	50		50		ns
t73	External Transmit Clock Low Width	30		30		ns
t74	External Transmit Clock High Width	30		30		ns
t75	External Transmit Clock Fall Time		5		10	ns
t76	External Transmit Clock Rise Time		5		10	ns
t77	CKA Low to TXA Data Valid		20		30	ns

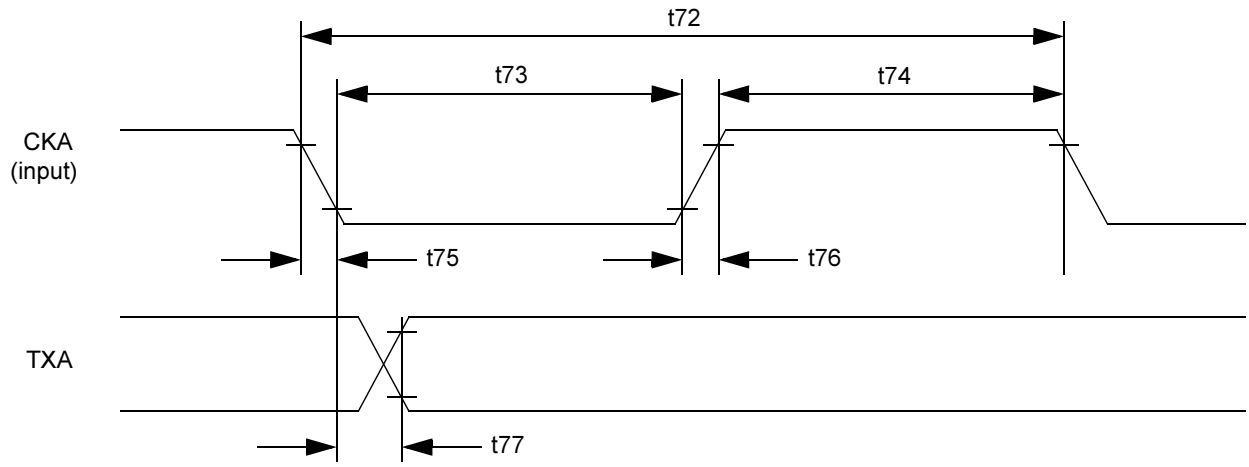


Figure 11. ASCI Transmitter Timing Diagram

ASCI Receiver Timing (See Figure 12)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 13. ASCI Receiver Timing

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t79	External Receive Clock Period	50		50		ns
t80	External Receive Clock Low Time	30		30		ns
t81	External Receive Clock High Time	30		30		ns
t82	External Receive Clock Fall Time		5		10	ns
t83	External Receive Clock Rise Time		5		10	ns
t84	RXA Setup to CKA Rise	20		25		ns
t85	RXA Hold from CKA High	5		5		ns

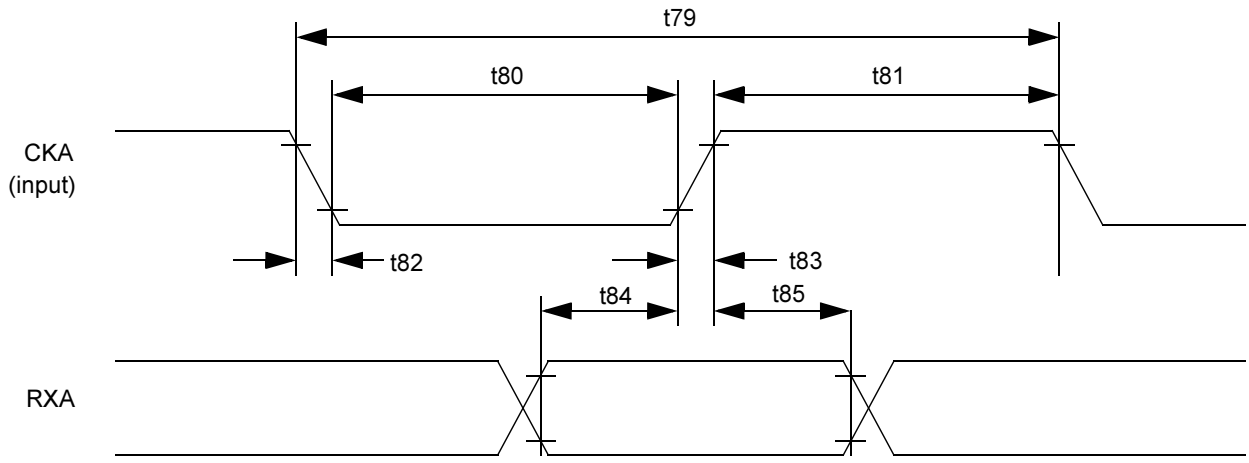


Figure 12. ASCI Receiver Timing Diagram

ASCI/CSIO Timing-Baud Rate Generator, $\overline{\text{CTSA}}$, $\overline{\text{DCDA}}$, and $\overline{\text{RTSA}}$ Timing (See Figures 13 through 15)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 14. ASCI/CSIO Timing

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t88	BUSCLK High to BRG Output Delay		30		40	ns
t89	BRG Output Fall Time		10		15	ns
t90	BRG Output Rise Time		10		15	ns
t91	$\overline{\text{CTS}}$ Low Time	100		100		ns
t92	$\overline{\text{CTS}}$ High Time	100		100		ns
t93	$\overline{\text{DCD}}$ Low Time	100		100		ns
t94	$\overline{\text{DCD}}$ High Time	100		100		ns
t95	BUSCLK Low to $\overline{\text{RTS}}$ Valid		20		20	ns

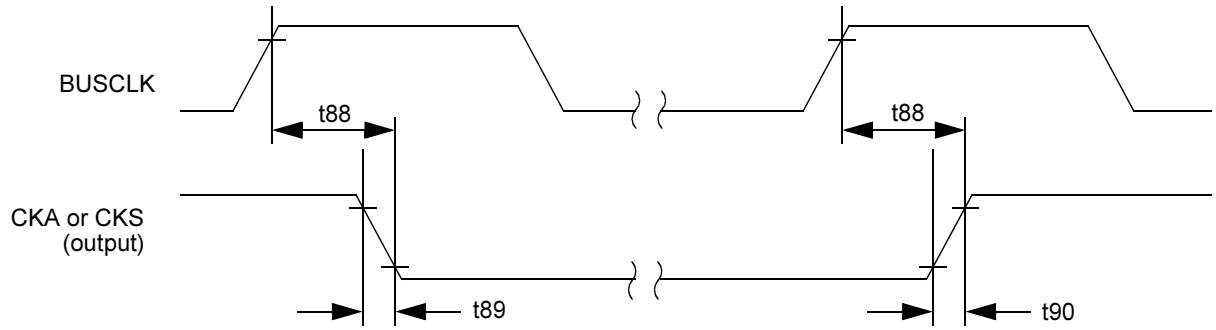


Figure 13. Baud Rate Generator Timing Diagram

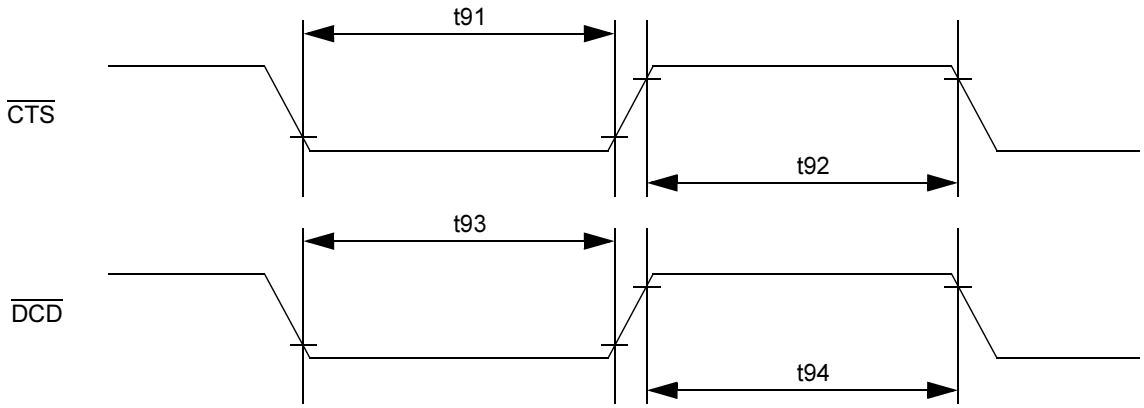


Figure 14. $\overline{\text{CTSA}}$ and $\overline{\text{DCDA}}$ Timing Diagram

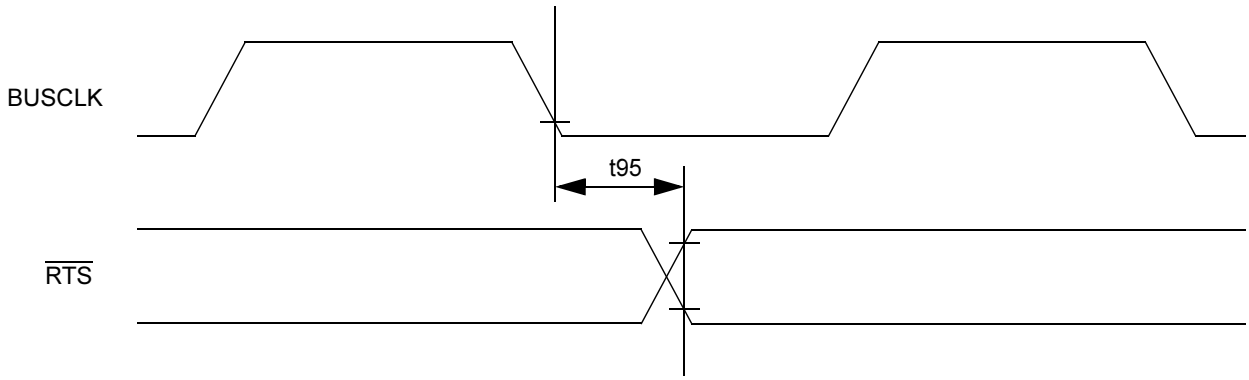


Figure 15. $\overline{\text{RTSA}}$ Timing Diagram

General-Purpose I/O Port Timing. (See Figure 16)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 15. General-Purpose I/O Port Timing

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t100	I/O Port Data Setup to $\overline{\text{IORD}}$ Fall	10		10		ns
t101	I/O Port Data Hold from $\overline{\text{IORD}}$ High	5		5		ns
t102	I/O Port Data Valid from $\overline{\text{IOWR}}$ High		20		20	ns

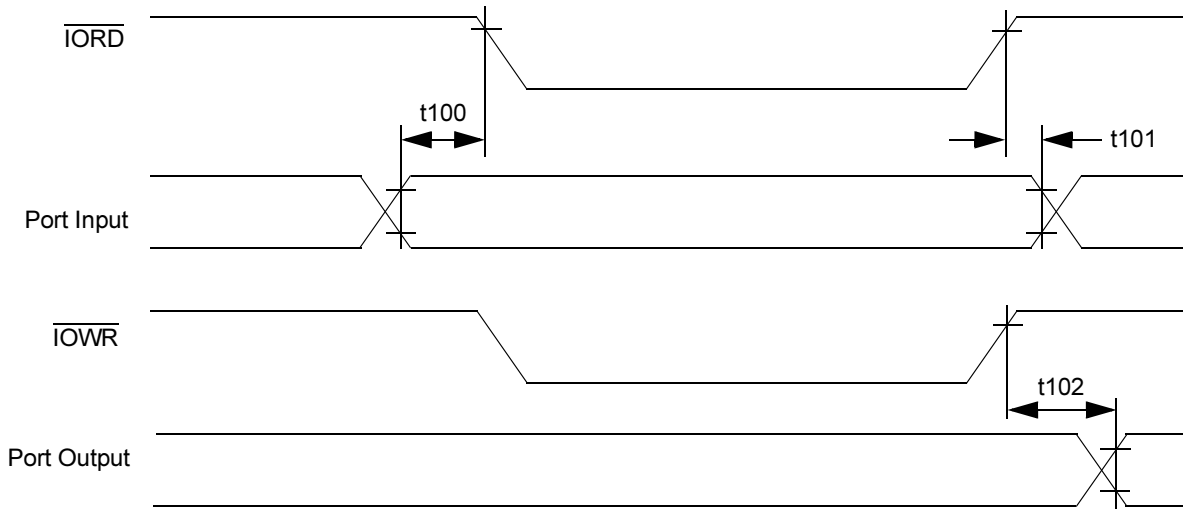


Figure 16. General-Purpose I/O Port Timing Diagram

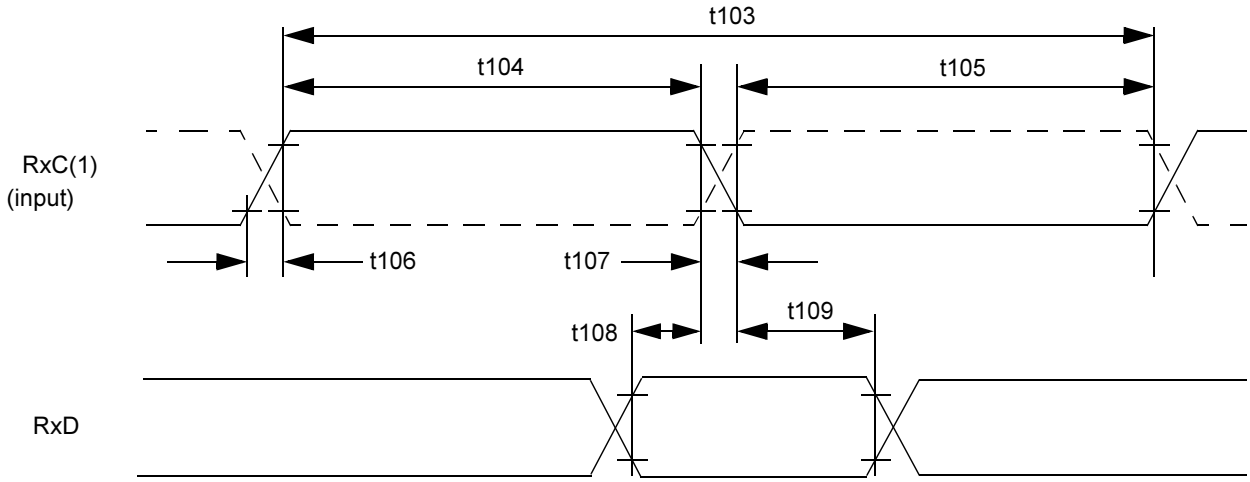
HDLC Receive Timing- Full Time HDLC Mode (See Figures 17 and 18)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 16. HDLC Receive Timing

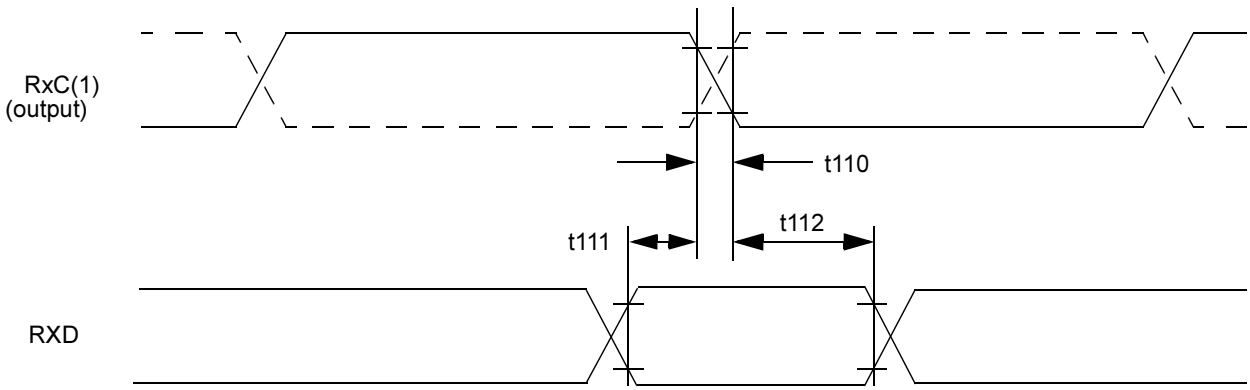
		Z80382		Z8L382		
No	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t103	External Receive Clock Period[1]	50		50		ns
t104	External Receive Clock Low Time	15		15		ns
t105	External Receive Clock High Time	15		15		ns
t106	External Receive Clock Rise Time		5		10	ns
t107	External Receive Clock Fall Time		5		10	ns
t108	RxD Setup to RxC Edge (External RxC)[1]	20		20		ns
t109	RxD Hold from RxC Low/High (External RxC)[1]	5		5		ns
t110	RxC rise/fall time (Internal RxC)[1]		5		10	ns
t111	RxD Setup to RxC Edge (Internal RxC)[1]	20		25		ns
t112	RxD Hold from RxC Low/High (Internal RxC)[1]	5		10		ns

Note: 1. Receive clock sampling edge is configurable by means of RIRn[6]. See Z80382 User Manual.



Note 1. HDLC clock triggering polarity is configurable by means of RIRn[6]. See Z80382 User Manual.

Figure 17. HDLC Receive Timing Diagram (Full Time HDLC, RxC Input)



Note 1. Receive clock sampling edge is configurable by means of RIRn[6]. See Z80382 User Manual.

Figure 18. HDLC Receive Timing (Full Time HDLC, RxC Output)

HDLC Transmit Timing- Full Time HDLC Mode (See Figure 19)

Table 17. HDLC Transmit Timing

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t113	External Transmit Clock Period[1]	50		50		ns
t114	External Transmit Clock High Time	15		15		ns
t115	External Transmit Clock Low Time	15		15		ns
t116	External Transmit Clock Fall Time		5		10	ns
t117	External Transmit Clock Rise Time		5		10	ns
t118	TxC Low to TxD Data Valid		20		25	ns

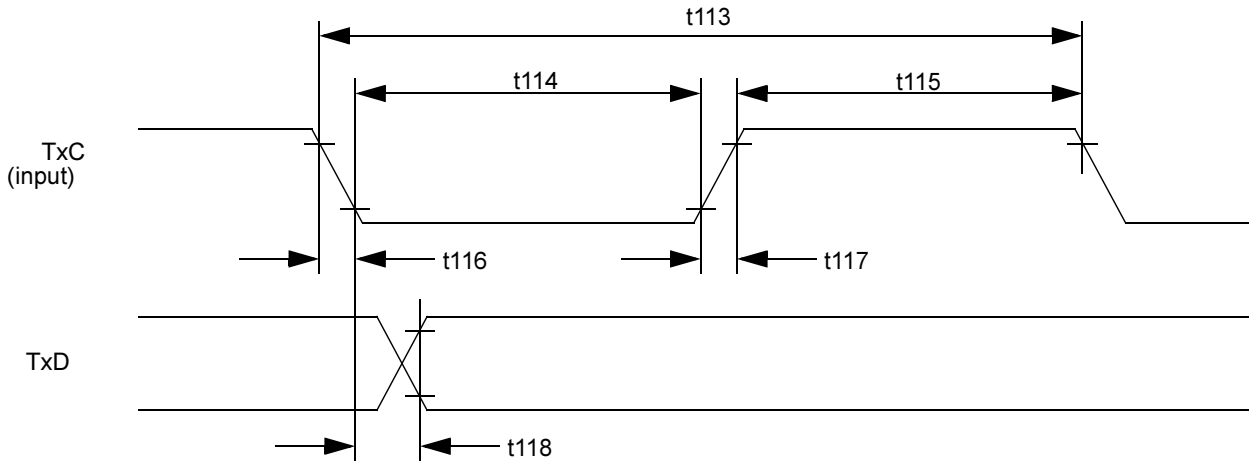


Figure 19. HDLC Transmit Timing Diagram (Full Time HDLC)

HDLC Timing - Non-GCI TDM mode (See Figure 20)

Specifications apply over Standard Operating Conditions unless otherwise noted.
C_L = 50 pF for outputs

Table 18. HDLC Timing - Non-GCI TDM Mode

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t120	FSC Setup to BCL Fall	30		50		ns
t121	FSC Hold from BCL Low	20		30		ns
t122	BCL Period	50		50		ns
t123	BCL High Time	15		15		ns
t124	BCL Low Time	15		15		ns
t125	BCL High to $\overline{\text{TxEN}}$ Low		15		20	ns
t126	BCL High to $\overline{\text{TxEN}}$ High		15		20	ns
t127	BCL High to TxD Valid		15		20	ns
t128	BCL High to TxD Invalid	15		20		ns
t129	RxD Setup to BCL Fall (Rise)[1]	15		20		ns
t130	RxD Hold from BCL Low (High)[1]	5		5		ns

Note: 1. Receive clock sampling edge is configurable by means of RIRn[6]. See Z80382 User Manual.

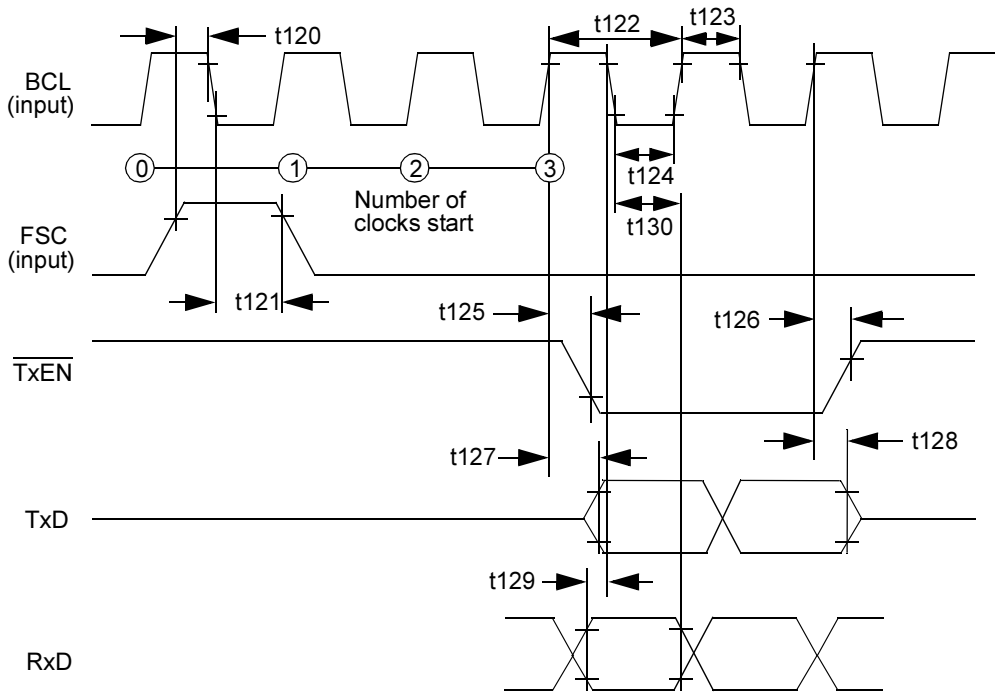


Figure 20. HDLC Timing Diagram - Non-GCI TDM Mode (for Start = 3, Length = 2, Negative Edge RxD Sampling

GCI/SCIT Timing-Slave Characteristics (See Figure 21)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 19. GCI/SCIT Timing - Slave Characteristics

		Z80382		Z8L382		
Symbol	**Parameter**	**Min.**	**Max.**	**Min.**	**Max.**	**Units**
t131	DCL Data Clock Rise/Fall Time		5		10	ns
t132	DCL Clock Period	50		50		ns
t133	DCL Pulse Width High	15		15		ns
t134	FSC Setup to DCL Fall	30		30		ns
t135	FSC Hold from DCL Low	5		10		ns
t136	DCL High to DU/DD Transmit Data Valid		15		20	ns

Table 19. GCI/SCIT Timing - Slave Characteristics (Continued)

Symbol	Parameter	Z80382		Z8L382		Units
		Min.	Max.	Min.	Max.	
t137	FSC High to DU/DD Transmit Data Valid		15		20	ns
t138	DU/DD Receive Data Setup to DCL Fall	15		20		ns
t139	DU/DD Receive Data Hold from DCL Low	0		0		ns

GCI/SCIT Timing - Master Characteristics (See Figure 21)

Specifications apply over Standard Operating Conditions unless otherwise noted. C_L = 50 pF for outputs.

Table 20. GCI/SCIT Timing - Master Characteristics

Symbol	Parameter	Z80382		Z8L382		Units
		Min.	Max.	Min.	Max.	
t131	DCL Data Clock Rise/Fall Time		5		10	ns
t132	DCL Clock Period	50		50		ns
t133	DCL Pulse Width High	15		15		ns
t134	FSC Setup to DCL Fall	30		30		ns
t135	FSC Hold from DCL Low	5		10		ns
t136	DCL High to DU/DD Transmit Data Valid		15		20	ns
t137	FSC High to DU/DD Transmit Data Valid		15		20	ns
t138	DU/DD Receive Data Setup to DCL Fall	15		20		ns
t139	DU/DD Receive Data Hold from DCL Low	0		0		ns
t140	FSC High from DCL High	0		0		ns

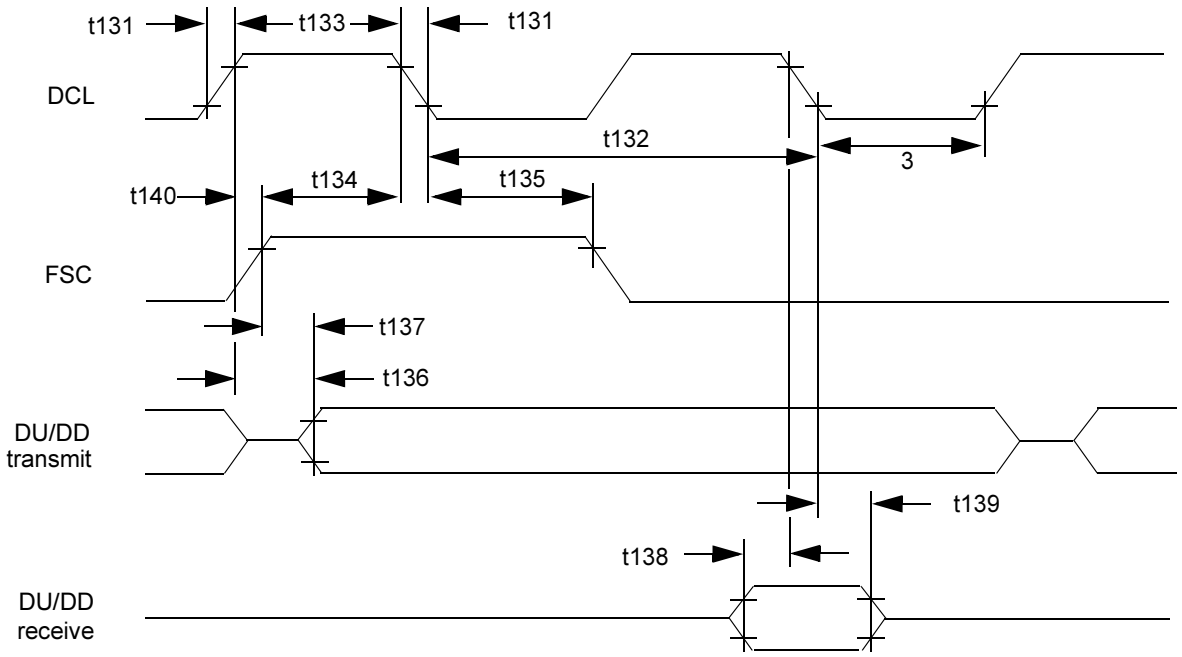


Figure 21. GCI/SCIT Slave and Master Timing Diagram

Pin Functions

Tables 21 through 11 describe the input and output signals of the Z382. Signals are normally asserted in the High state and negated in the Low state. An overline ( ) above the signal name indicates that the signal is asserted in the Low state and negated in the High state.

Many pins have multiple functions, and thus may appear more than once in the pin description tables. In each table, such pins are described using their function in that mode. Likewise, some signals may be output on alternate pins depending on the mode under which the Z382 is operating. The notation xx/yy in the Pin Number column indicates that the signal may be assigned to pin xx or pin yy.

Table 21. Multiprocessor Unit (MPU) Signals

Pin Name	Pin Number(s)	Description
A23 - 0	141 - 144, 1 - 4, 6 - 13, 15 - 22	**Address Bus** (outputs, active High, 3-state): These non-multiplexed address signals provide a linear memory address space of 16 MB. The address signals are also used to access I/O devices.
$\overline{\text{BUSACK}}$	132	**Bus Acknowledge** (output, active Low, 3-state): This signal, when asserted, indicates that the 380C has accepted an external bus request and has 3-stated its output drivers for the address bus, databus and the bus control signals TREFR, TREFA, TREFC, BHEN, BLEN, MRD, MWR, IORQ, IORD, and IOWR. The 380C cannot provide any DRAM refresh transactions while it is in the bus acknowledge state.
$\overline{\text{BHEN}}$	118	**Byte High Enable** (output, active Low, 3-state): This signal is asserted at the beginning of a memory or refresh transaction, to request an operation on D15-8. If software initiates a 16-bit memory operation, but MSIZE is asserted indicating a byte-wide memory, only the data on D7-0 is transferred in the current transaction, and another transaction is performed to transfer the other data byte, also on D7-0. See note in the next paragraph under BLEN pin description.
$\overline{\text{BLEN}}$	119	**Byte Low Enable** (output, active Low, 3-state): This signal is asserted at the beginning of a memory or refresh transaction, to request an operation on D7-0. If software initiates a 16-bit memory transaction, but MSIZE is asserted indicating a byte-wide memory, only the data on D7-0 is transferred in the current transaction, and another transaction is performed to transfer the other data byte, also on D7-0. Note: To align Z382 documentation and terminology with historical Z80 and industry practice, the names of the BHEN and BLEN pins, as well as the D15-8 and D7-0 pins, have been swapped on the Z382 compared to the Z380. This fact should be significant only for those using a Z380 Emulator in a Z382-based project.
$\overline{\text{BUSREQ}}$	133	**Bus Request** (input, active Low): When this signal is asserted, an external bus master is requesting control of the bus. BREQ has higher priority than all nonmaskable and maskable interrupt requests.
BUSCLK	127	**Bus Clock** (output, active High, 3-state): This signal is the reference edge for the majority of other signals generated by the 380C. Its frequency may be that of the CLKI pin, or CLKI divided by two or times two.

Table 21. Multiprocessor Unit (MPU) Signals (Continued)

Pin Name	Pin Number(s)	Description
D15-0	24 - 31 33 - 40	**D15-0 Data Bus** (input/output, active High, 3-state): This bidirectional 16-bit data bus is used for data transfer between the 380C and memory or I/O devices. In a memory word transfer, the even-addressed (A0=0) byte is transferred on D7-0, and the odd-addressed (A0=1) byte on D15-8. 8-bit memories should be connected to D7-0, while 8-bit I/O devices should be attached to D15-8 (this difference tends to equalize electrical loading). (See note under BLEN pin description on page 38.)
HALT STNBY	121 120	**HALT, STANDBY Status** (outputs, active Low): These two outputs indicate the status of the Z382 as follows: STNBY / HALT / Mode H / H / Normal instruction execution H / L / HALT instruction L / H / SLEEP Mode: clock runs but is blocked from most of the chip L / L / STANDBY Mode: oscillator is stopped
INT3 INT2 INT1	139 138 137	**Interrupt Requests** (inputs, active Low): Asynchronous maskable interrupt inputs. Can be selected as low- or high-level sensitive, or as falling- or rising-edge triggered.
INT0	136	**Interrupt Request** (input, active Low): INT0 is logically ORed (positive-logic ANDed) with the interrupt requests from the on-chip MIMIC, DMAs, and HDLC controllers, to create the processor's INT0 input.
IOCLK	114	**Input/Output Clock** (output, active High, 3-state): This signal is a program controlled divided-down version of BUSCLK. The division factor can be two, four, six or eight with I/O transactions and interrupt-acknowledge transactions occurring relative to IOCLK. IOCLK can be disabled, in which case BUSCLK is the timing reference for I/O transactions. Note: The INTACK output of the Z380 has been omitted on the Z382 for pinning reasons. A similar signal can be easily obtained by low-active-ANDing (positive-logic ORing) the M1 and IORQ outputs.
IORQ	115	**Input/Output Request** (output, active Low, 3-state): This signal is active during all I/O read and write transactions and interrupt acknowledge transactions.
IORD	125	**Input/Output Read Strobe** (output, active Low, 3-state): This signal is used to strobe data from the peripherals during I/O read transactions.
IOWR	123	**Input/Output Write Strobe** (output, active Low, 3-state): This signal is used to strobe data into the peripherals during I/O write transactions.
IOCS1 IOCS2	45 46	**Input/Output Chip Select** (output, active Low): These outputs may be used to access external I/O devices. The base I/O address and range are programmable.

Table 21. Multiprocessor Unit (MPU) Signals (Continued)

Pin Name	Pin Number(s)	Description
$\overline{M1}$	116	**Machine Cycle One** (output, active Low, 3-state): This signal is active during instruction fetch and interrupt acknowledge transactions. The Z382 does not support RETI decoding by Z80 peripherals (PIO, SIO, and CTC). It does support Z80-type interrupt daisy-chaining by devices that include explicit clearing of IUS (for example, SCC).
$\overline{MRD}$	126	**Memory Read** (output, active Low, 3-state): This signal indicates that the addressed memory location places its data on the data bus. $\overline{MRD}$ is active from the end of T1 until the end of T4 during memory read transactions.
$\overline{MSIZE}$	117	**Memory Size** (input/open-drain output, active Low): In 16-bit memory operations, this signal indicates whether the addressed memory location is word size (logic High) or byte size (logic Low). In the latter case, the 8-bit memory is connected to the D7-0 lines, and an additional memory transaction on D7-0 automatically is generated to transfer the other byte of the word. (See the note on pin name swapping after the $\overline{BLEN}$ pin description on page 38) $\overline{MSIZE}$ is driven as an open-drain output by the memory decoding modules, when they are enabled in 8-bit mode and the address falls within their range.
$\overline{MWR}$	124	**Memory Write** (output, active Low, 3-state): This signal indicates that the addressed memory location stores the data on the databus, as qualified by $\overline{BHEN}$ and $\overline{BLEN}$. $\overline{MWR}$ is active from the end of T2 until the end of T4 during memory write transactions.
$\overline{NMI}$	135	**Nonmaskable Interrupt** (input, falling edge-triggered): This input has higher priority than the maskable interrupt inputs INT3-INT0.
$\overline{RESET}$	134	**Reset** (input, active Low): This input must be active for a minimum of five BUSCLK periods to initialize the Z382. The effect of $\overline{RESET}$ is described in detail in the Reset section.
$\overline{ROMCS}$	42	**ROM Chip Select** (output, active Low): After Reset, the Z382 drives this output and $\overline{MSIZE}$ Low for all memory accesses with A23=0. Software can program the chip select logic to assert $\overline{ROMCS}$ for a different range of memory addresses. If ROM is 16 bits wide and composed of two 8-bit devices, connect the Chip Select inputs of both devices to $\overline{ROMCS}$, and program the hardware not to force $\overline{MSIZE}$ Low in the first two instructions of the ROM code.

Table 21. Multiprocessor Unit (MPU) Signals (Continued)

Pin Name	Pin Number(s)	Description
RAMCSL RAMCSH	43 44	**RAM Chip Select Low, High** (outputs, active Low): After Reset, the Z382 drives RAMCSL and MSIZE Low for memory cycles with A23=1, and puts the RAMCSH pin under control of its alternate use (a port pin). If RAM is only eight bits wide, connect its Chip Select input to RAMCSL. If RAM is 16 bits wide, connect one of these pins to the chip select of each 8-bit RAM, and reprogram the hardware not to force MSIZE Low, in which case RAMCSL is qualified with BLEN, and RAMCSH is qualified with BHEN. On the Z382 these signals have the same timing as address lines, so there is no timing penalty for this qualification.
TREFA	111	**Timing Reference A** (output, active Low, 3-state): This timing reference signal goes Low at the end of T2 and returns High at the end of T4 during a memory read, memory write or refresh transaction. It can be used to control the address multiplexer for a DRAM interface or as the RAS signal at higher processor clock rates.
TREFC	110	**Timing Reference C** (output, active Low, 3-state): This timing reference signal goes Low at the end of T3 and returns High at the end of T4 during a memory read, memory write or refresh transaction. It can be used as the CAS signal for DRAM accesses.
TREFR	112	**Timing Reference R** (output, active Low, 3-state): This timing reference signal goes Low at the end of T1 and returns High at the end of T4 during a memory read, memory write or refresh transaction. It can be used as the RAS signal for DRAM accesses.
WAIT	130	**Wait** (input, active Low): This input is sampled by BUSCLK or IOCLK, as appropriate, to insert Wait states into the current bus transaction.

.

Table 22. UART, Timer and CSIO Signals

Pin Name	Pin Number(s)	Description
CKA0 CKA1	53/92/96 49/65/100	**Asynchronous Clock 0, 1** (Bidirectional): Clock signals to or from the asynchronous channels (ASCIs).
$\overline{\text{CTS0}}$ $\overline{\text{CTS1}}$	61/111 60/110	**Clear to Send 0, 1** (Inputs, active Low): Transmit control signals for the ASCI channels.
$\overline{\text{DCD0}}$ $\overline{\text{DCD1}}$	89/112 66	**Data Carrier Detect 0, 1** (Inputs, active Low): Receive control signals for the ASCI channels. $\overline{\text{DCD1}}$ is not available in ISA applications.
$\overline{\text{RTS0}}$ $\overline{\text{RTS1}}$	88/93 94	**Request to Send 0, 1** (Outputs, active Low, 3-state): Software-controlled output from the ASCI channels.
RXA0 RXA1	52/90/99 55/67/103	**Receive Data 0, 1** (Inputs): ASCI Receive data.
TXA0 TXA1	51/91/98 54/87/102	**Transmit Data 0, 1** (Outputs): ASCI Transmit data.
CKS	58/64	**Serial Clock** (Bidirectional): The clock for the CSIO channel.
RXS	57/63	**Clocked Serial Receive Data** (Input): Receive data for the CSIO channel.
TXS	56/62	**Clocked Serial Transmit Data** (Output): Transmit data from the CSIO channel.
TOUT	46/109	**Timer Out** (Output, active High): Pulse output from PRT1.

Table 23. ISA Bus Signals

Pin Name	Pin Number(s)	Description
HD7-0	78 - 85	**Host Data Bus** (Input/Output, 3-state): ISA or PCMCIA data bus.
$\overline{\text{HDOEN}}$	62	**Host Data Output Enable** (Output, active Low): This signal goes Low when the Host reads data from the MIMIC, the I/O Mailbox, or the Plug and Play interface, and during Host DMA read cycles.
HA11-0	64 - 67 69 - 76	**Host Address** (Input): Part of the ISA or PCMCIA address bus. The MS bits can be decoded by the built-in address decoder; bits 2-0 determine which MIMIC register the Host accesses. Bits 11-10 are decoded only by the Plug and Play ISA module.
HAEN	63	**Host Address Enable** (Input): HAEN must be Low to qualify COM Port decoding, I/O Mailbox decoding, and Plug and Play decoding. To support 16-bit decoding of Host I/O addresses, provide an external decoder for HA15-12 and HAEN all Low and connect its Low-active output to this pin.
$\overline{\text{HWR}}$	60	**Host Write** (Input, active Low): The Host drives this input Low to signal the MIMIC that a write operation is taking place.
$\overline{\text{HRD}}$	61	**Host Read** (Input, active Low): This input is used by the Host to signal the MIMIC interface that a read operation is taking place.
HINT1 HINT2	87 88	**Host Interrupt** (Outputs, active High): One of these outputs is driven High by the Plug and Play module when the MIMIC requests an interrupt from the Host. The unused signal is 3-stated.
$\overline{\text{HDAK0}}$ HDAK1	89 90	**Host DMA Acknowledge** (Inputs, active Low): These inputs indicate that the Host DMA controller has acknowledged the request and is transferring data.
HDRQ0 HDRQ1	91 92	**Host DMA Request** (Outputs, active High, 3-state): These outputs request a DMA transfer operation from the Host.

Table 24. Parallel Ports

Pin Name	Pin Number(s)	Description
PA7-0 PB7-0 PC7-0 PD7-0	78 - 85 51 - 58 44 - 45, 47 - 49, 109, 101, 97 69 - 76	**Parallel Ports A, B, C, D** (Input/Outputs): These lines can be configured as inputs or outputs on a bit-by-bit basis. In an ISA or PCMCIA application, Ports A and D are not pinned out, the registers for Ports A and D are used by the MIMIC function, and Ports B and C are selectively multiplexed with the on-chip peripherals (ASCIs, CSIO, PRT). In other applications all four ports are available with minimal multiplexing.

Table 25. HDLC Serial Channel and GCI/SCIT Signals

Pin Name	Pin Number(s)	Description
TxD0 TxD1 TxD2	98 102 107	**HDLC Transmit** (outputs): These pins are used to transmit serial data from the HDLC controllers when they are not operating by means of the GCI/SCIT interface.
RxD0 RxD1 RxD2	99 103 108	**HDLC Receive** (inputs): These pins are used to receive serial data for the HDLC controllers when they are not operating by means of the GCI/SCIT interface.
TxC0/FSC0 TxC1/FSC1 TxC2/FSC2	96 100 105	**HDLC Transmit Clock/Frame Sync** (input/outputs): In non-TDM, non-GCI modes, these can be used as external bit clock inputs or can be programmed to output the Tx clock. In non-GCI TDM mode, these pins carry the Frame Sync pulse.
RxC0/BCL0 RxC1/BCL1 RxC2/BCL2	97 101 106	**HDLC Clock/Bit Clock** (inputs): Optional external bit clock inputs.
$\overline{\text{TxEN0}}$ $\overline{\text{TxEN1}}$ $\overline{\text{TxEN2}}$	93 94 109	**HDLC Transmit Enable** (outputs, active Low): In a non-GCI TDM mode, these outputs indicate when an HDLC Transmitter is enabled and is in its active time slot. In non-GCI, non-TDM mode, these outputs are Low when the Transmitter is enabled. They can be used to enable an external driver on the TxD line.
DU DD	105 107	**GCI/SCIT Data Upstream, Downstream** (input/outputs, open-drain): The two bidirectional data streams of the GCI/SCIT interface.
DCL	106	**GCI/SCIT Clock** (input): Bit clock for the GCI/SCIT interface.
FSC	108	**GCI/SCIT Frame Sync** (input): This pin is used to synchronize the GCI/SCIT serial frames. This pin is driven active by “the upstream device” (ISDN transceiver) at the start of each GCI/SCIT frame.

Table 26. PCMCIA Interface Signals

Pin Name	Pin Number(s)	Description
HA9-0	66 - 67 69 - 76	**PCMCIA Address Bus** (inputs): Provide Host PC addressing of attribute memory, configuration registers, and MIMIC. Decoded by the I/O address decoder.
HD7-0	78 - 85	**PCMCIA Data Bus** (input/outputs): Used for data transfers between the Host PC and the MIMIC, the attribute memory, and the configuration registers.
$\overline{\text{PCIORD}}$	61	**PCMCIA I/O Read** (input, active Low): Used to generate the $\overline{\text{INPACK}}$ signal when an I/O read cycle is within the configured range, and reads from the MIMIC.
$\overline{\text{PCIOWR}}$	60	**PCMCIA I/O Write** (input, active Low): This signal is used to write to the MIMIC.
$\overline{\text{PCWE}}$	89	**PCMCIA Write Enable** (input, active Low): Used to write to the attribute memory or to the configuration register which is addressed by means of HA9-1. Such an operation is recognized when $\overline{\text{PCWE}}$, $\overline{\text{PCCE1}}$, and $\overline{\text{PCREG}}$ are all Low, and either the interface is configured for I/O and memory operation, or $\overline{\text{PCIRQ}}$ is High, signifying ready, when configured as a memory only interface. The data applied while $\overline{\text{PCWE}}$ is Low are written to the attribute memory range on the positive edge of the $\overline{\text{PCWE}}$ or Card-enable ($\overline{\text{PCCE1}}$) signal.
$\overline{\text{PCOE}}$	90	**PCMCIA Output Enable** (input, active Low): $\overline{\text{PCOE}}$, $\overline{\text{PCCE1}}$, and $\overline{\text{PCREG}}$ all Low signify a read from attribute memory or a configuration register as selected by HA9-1.
$\overline{\text{PCCE1}}$	91	**PCMCIA Chip Enable 1** (input, active Low): $\overline{\text{PCCE1}}$ Low indicates a read or write access to: an even addressed byte in attribute memory, a configuration register, or the MIMIC.
$\overline{\text{PCREG}}$	63	**PCMCIA Register Select** (input, active Low): $\overline{\text{PCREG}}$ Low indicates a read or write access to the attribute memory range or to the I/O address range.
$\overline{\text{INPACK}}$	62	**PCMCIA Input Acknowledge** (Output, active Low): $\overline{\text{INPACK}}$ goes Low while an I/O read access is performed within the configured I/O address range. If the PCMCIA interface is configured such that it reacts independent of the address to all I/O read cycles, then $\overline{\text{INPACK}}$ is activated with $\overline{\text{PCIORD}}$.

Table 26. PCMCIA Interface Signals (Continued)

PCIRQ	87	**PCMCIA Interrupt Request** (Output, active Low): After the PCMCIA interface is reset it is in a MEMORY-ONLY mode, and this signal is driven Low to signify a Busy state until the 380C writes a register bit to indicate it is ready. After the card is then configured by the Host, PCIRQ goes Low to request a Host PC interrupt when the internal INT0 signal is asserted by the MIMIC. PCIRQ is monitored by the PCMCIA Host adapter and, dependent on the configuration, connected to one of the Host interrupts (for example, COM1 or COM2 interrupt). PCIRQ can be programmed to be a pulsed interrupt with a minimal pulse length of one microsecond, or a level-interrupt that is reset when the interrupt is processed by the Host. This choice is made by means of bit 6 of the Configuration Option Register.
PCRESET	92	**PCMCIA Reset** (input, active High): Setting PCRESET High resets the PCMCIA interface. The card configuration register is cleared and the PCMCIA interface operates in the MEMORY-ONLY mode until it is configured again. The attribute memory has to be initialized by the controller, and the Ready/Busy (PCIRQ) signal has to be deactivated.
STSCHG	88	**PCMCIA Status Change** (output): This output is controlled by a bit in the PCMCIA module's 380C Control Register.

Table 27. Other Signals

Pin Name	Pin Number(s)	Description
CLKI	128	**Clock/Crystal** (input, active High): An externally generated clock can be input at this pin. Alternatively, a crystal can be connected between CLKI and CLKO. In either case, the frequency at this pin can be used directly as the processor clock (BUSCLK), or divided by two or multiplied by two, under software control.
CLKO	129	**CLKO Crystal** (output, active High): Crystal oscillator connection.This pin must be left open if an externally generated clock is input at the CLKI pin. Feedback on this pin can be disabled by software to save power and noise when an external clock is used.
IEI	47	**Interrupt Enable In** (input, active High): If external devices are connected to INT0, and have higher interrupt priority than the on-chip MIMIC, DMAs, and HDLC controllers, this signal must be connected to the IEO output of the lowest-priority among such devices.
IEO	48	**Interrupt Enable Out** (output, active High): If external devices are connected to INT0, and have lower interrupt priority than the on-chip MIMIC, DMAs, and HDLC controllers, this signal must be connected to the IEI input of the highest-priority device.
V_{DD}	5, 23, 41, 59, 77, 95, 113, 131	**Power Supply**: These eight pins carry power to the device. They must be tied to the same voltage externally.
V_{SS}	14, 32, 50, 68, 86, 104, 122, 140	**Ground**: These eight pins are the ground references for the device. They must be tied to the same voltage externally.

Functional Description

The functional blocks within the Z382 can be broadly identified as central processing unit, host interface, serial communication channels, DMA control, timers and counters, and system interface logic. Each of these blocks are further described in the sections which follow.

For additional information, please refer to the Z382 User's Manual, available from your ZiLOG representative or distributor.

Central Processing Unit

The Central Processing Unit (CPU) core of the Z382 is the 380C (Z380), which is a binary-compatible extension of the Z80® and Z180™ CPU architectures. High

throughput rates for the 380C are achieved by a high clock rate, high bus bandwidth and instruction fetch/execute overlap. Communicating to the external world through an 8- or 16-bit data bus, the 380C is a full 32-bit machine internally, with a 32-bit ALU and 32-bit registers.

Modes of Operation

The 380C can operate in either NATIVE or EXTENDED mode, as controlled by a bit in the Select Register (SR). In NATIVE mode (the default configuration after Reset), all address manipulations are performed modulo 65536 (16 bits).

In this mode the Program Counter (PC) only increments across 16 bits, all address manipulation instructions (increment, decrement, add, subtract, indexed, stack relative, and PC relative) only operate on 16 bits, and the Stack Pointer (SP) only increments and decrements across 16 bits. The program counter high-order word is left at all zeros, as are the high-order words of the stack pointer and the I register. NATIVE mode is fully compatible with the Z80 CPU's 64 KB address space.

It is still possible to address memory outside of the 64 KB address space for data storage and retrieval in NATIVE mode, however, as direct addresses, indirect addresses, and the high-order word of the SP, I and the IX and IY registers may be loaded with non-zero values. Executed code and interrupt service routines must reside in the lowest 64 KB of the address space.

In EXTENDED mode, all address manipulation instructions operate on 32 bits, potentially allowing access to a 4 GB address space. In both NATIVE and EXTENDED modes, however, the Z382 outputs only 24 bits of the address onto the external address bus, limiting the actual usable address space to 16 MB. Only the width of manipulated addresses distinguish NATIVE from EXTENDED mode. The 380C implements one instruction to allow switching from NATIVE to EXTENDED mode, but once in EXTENDED mode, only Reset returns the CPU to NATIVE mode. This restriction applies because of the possibility of misplacing interrupt service routines or vector tables during the translation from EXTENDED mode back to NATIVE mode.

In addition to NATIVE and EXTENDED mode, which is specific to memory space addressing, the 380C can operate in either WORD or LONG WORD mode specific to data load and exchange operations. In WORD mode (the reset configuration), all word load and exchange operations manipulate 16-bit quantities. For example, only the low-order words of the source and destination are exchanged in an exchange operation, with the high-order words unaffected.

In LONG WORD mode, all 32 bits of the source and destination are exchanged. The 380C implements two instructions plus decoder directives to allow switching between WORD and LONG WORD modes. The two instructions perform a global

switch, while the decoder directives select a particular mode only for the instruction that they precede.

All word data arithmetic (as opposed to address manipulation arithmetic), rotate, shift and logical operations are always in 16-bit quantities. They are not controlled by either the NATIVE/EXTENDED or WORD/LONG WORD selections. The exceptions to the 16-bit quantities are, of course, those multiply and divide operations with 32-bit products or dividends.

Lastly, all word input/output operations are performed on 16-bit values.

CPU Address Spaces

The 380C architecture supports five distinct address spaces corresponding to the different types of locations that can be accessed by the CPU. These five address spaces are:

- CPU register space
- CPU control register space
- Memory address space
- I/O address space on-chip
- I/O address space external

CPU Register Space

The CPU register space, depicted in Figure 22, consists of all of the registers in the CPU register file. These CPU registers are used for data and address manipulation, and are an extension of the Z80 CPU register set. Four sets of this extended Z80 CPU register set are present in the 380C. Access to these registers is specified in the instruction, with the active register set selected by bits in the Select Register (SR) in the CPU control register space.

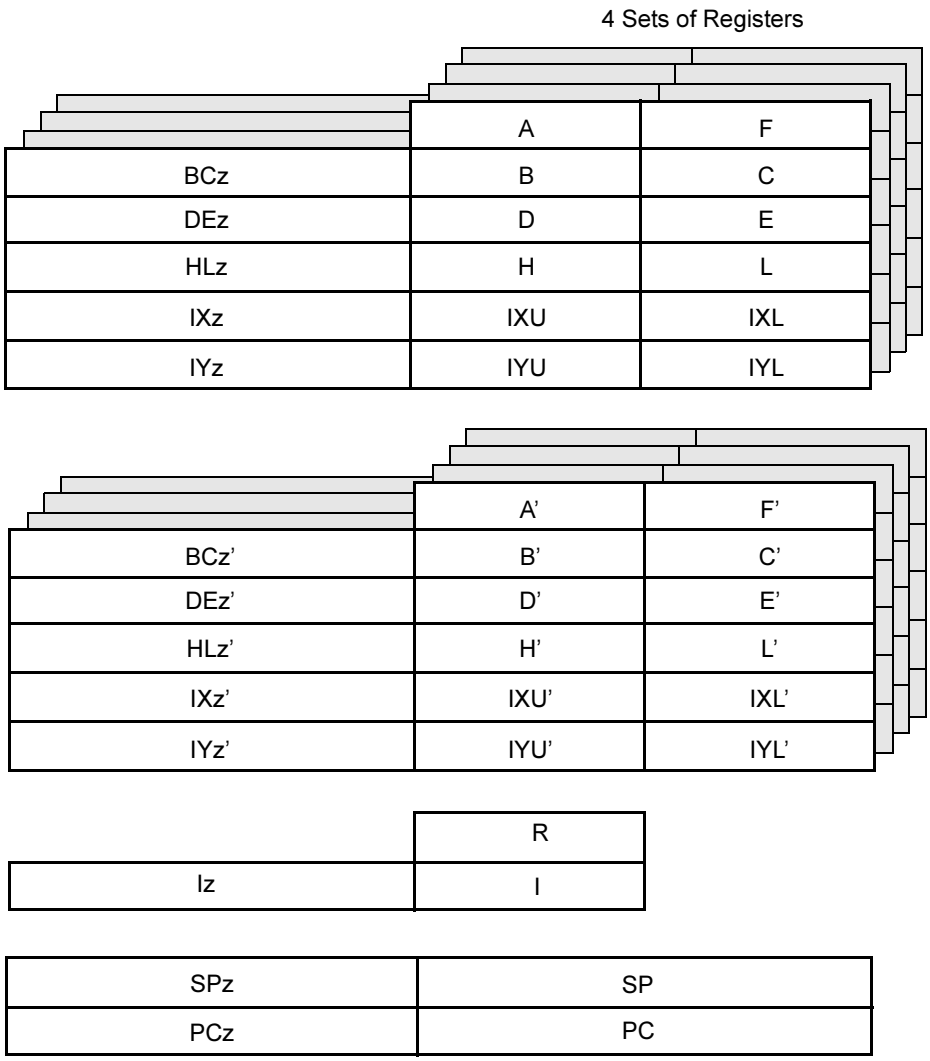


Figure 22. 380C Processor Core Register Set

Primary and Working Registers. The working register set is divided into the two register files; the primary file and the alternate file (designated by '). Each file contains an 8-bit Accumulator (A), a Flag register (F), and six general-purpose registers (B, C, D, E, H, and L). Only one file can be active at any given time, although data in the inactive file can still be accessed. Exchange instructions allow the programmer to exchange the active file with the inactive file.

The accumulator is the destination register for 8-bit arithmetic and logical operations. The six general-purpose registers can be paired (BC, DE, and HL), and are

extended to 32 bits by the z extension to the register to form three 32-bit general-purpose registers. The HL register serves as the 16-bit or 32-bit accumulator for word operations.

CPU Flag Register. The Flag register contains six flags that are set or reset by various CPU operations:

- Carry
- Add/Subtract
- Parity/Overflow
- Half Carry
- Zero
- Sign

Index Registers. The four Index registers, IX, IX', IY and IY', each hold a 32-bit base address that is used in the Indexed addressing mode. The Index registers can also function as general-purpose registers with the upper and lower bytes of the lower 16 bits accessed individually.

Interrupt Register. The Interrupt register (I) is used in interrupt modes 2 and 3 for INT0 to generate a 32-bit indirect address to an interrupt service routine. The I register supplies the upper 24 or 16 bits of the indirect address and the interrupting peripheral supplies the lower 8 or 16 bits. In the Assigned Vectors mode for INT1-3, the upper 16 bits of the vector are supplied by the I register; bits 15-9 are the assigned vector base and bits 8-0 are the assigned vector unique to each of INT1-3.

Program Counter. The Program Counter (PC) is used to sequence through instructions in the currently executing program and to generate relative addresses. The PC contains the 32-bit address of the current instruction being fetched from memory. In the NATIVE mode, the PC is effectively only 16 bits long, as carries from bit 15 to bit 16 are inhibited in this mode. In EXTENDED mode, the PC is allowed to increment across all 32 bits.

R Register. The R register can be used as a general-purpose 8-bit read/write register.

Stack Pointer. The Stack Pointer (SP) is used for saving information when an interrupt or trap occurs and for supporting subroutine calls and returns. Stack Pointer relative addressing allows parameter passing using the SP.

Select Register. The Select Register (SR) controls the register set selection and the operating modes of the 380C CPU.

CPU Control Register Space

The CPU control register space consists of the 32-bit Select Register (SR). The contents of SR determine the CPU operating mode, which register bank is used, the interrupt mode in effect, and other items of this type.

Memory Address Space

The actual usable memory space in the Z382 is 16 MB, because the lower 24 bits of the address are output on the external address bus. The 8-bit byte is the basic addressable element in the 380C memory address space. However, there are other addressable data elements; bits, 2-byte words, byte strings, and 4-byte words. The size of the data element being addressed depends on the instruction being executed as well as the WORD/LONG WORD mode.

When a word is stored in memory, the least significant byte precedes the more significant byte of the word, as in the Z80 CPU architecture. Also, the lower-addressed byte is present on the upper byte of the external data bus.

On-Chip and External I/O Address Space

The 380C CPU architecture distinguishes between the memory and I/O spaces and, therefore, requires specific I/O instructions. I/O instructions are used to access the Z382's internal peripherals as well as a number of control registers which deal with functions such as interrupts and traps. I/O instructions are also used to access external peripheral controllers connected to the Z382's external address, data and control busses.

Data Types

The Z380 CPU can operate on bits, Binary-Coded Decimal (BCD) digits (4 bits), bytes (8 bits), words (16 bits or 32 bits), byte strings, and word strings. Bits in registers can be set, cleared, and tested. BCD digits, packed two to a byte, can be manipulated with the Decimal Adjust Accumulator instruction (in conjunction with binary addition and subtraction) and the Rotate Digit instructions. Bytes are operated on by 8-bit load, arithmetic, logical, and shift and rotate instructions. Words are operated on in a similar manner by the word load, arithmetic, logical, and shift and rotate instructions. Block move and search operations can manipulate byte strings and word strings up to 64 KB or words long. Block I/O instructions have identical capabilities.

Addressing Modes

Addressing modes are used by the 380C to calculate the effective address of an operand needed for execution of an instruction. Seven addressing modes are supported by the CPU. Of these seven, one is an addition to the Z80 CPU

addressing modes (Stack Pointer Relative) and the remaining six modes are either existing or extensions to the Z80 CPU addressing modes.

Register Addressing

The operand is one of the 8-bit registers (A, B, C, D, E, H, L, IXU, IXL, IYU, IYL, A', B', C', D', E', H' or L'); or is one of the 16-bit or 32-bit registers (BC, DE, HL, IX, IY, BC', DE', HL', IX', IY' or SP) or one of the special registers (I or R).

Immediate Addressing

The operand is in the instruction itself and has no effective address. The DDIR IB and DDIR IW decoder directives allow specification of 24-bit and 32-bit immediate operands, respectively.

Indirect Register Addressing

The contents of a register specify the effective address of an operand. The HL register is the primary register used for memory accesses, but BC and DE can also be used. (For the JP instruction, IX and IY can also be used for indirection.) The BC register is used for I/O space accesses.

Direct Addressing

The effective address of the operand is the location whose address is contained in the instruction. Depending on the instruction, the operand is either in the I/O or memory address space. Sixteen bits of direct address is the norm, but the Decoder Directive Immediate Byte (DDIR IB) and Decoder Directive Immediate Word (DDIR IW) allow 24-bit and 32-bit direct addresses, respectively.

Indexed Addressing

The effective address of the operand is the location computed by adding the two's-complement signed displacement contained in the instruction to the contents of the IX or IY register. Eight bits of index is the norm, but the DDIR IB and DDIR IW decoder directives allow 16-bit and 24-bit indexes, respectively.

Program Counter Relative Addressing

An 8-, 16- or 24-bit displacement contained in the instruction is added to the Program Counter to generate the effective address. This mode is available only for Jump and Call instructions.

Stack Pointer Relative Addressing

The effective address of the operand is the location computed by adding the two's-complement signed displacement contained in the instruction to the con-

tents of the Stack Pointer. Eight bits of index is the norm, but the DDIR IB and DDIR IW decoder directives allow 16- and 24-bit indexes, respectively.

Instruction Set

The 380C instruction set is an expansion of the Z80 instruction set. The enhancements include support for additional addressing modes and a number of new instructions.

The 380C Op Codes are compatible with the Z80 CPU and Z180 MPU. A Z80/Z180 program can be executed on the 380C without modification.

The instruction set is divided into 12 groups by function; these are listed below. Consult the Z380 User's Manual for additional details on the instruction set.

- 8-bit Load/Exchange
- 16/32-bit Load, Exchange, Swap and Push/Pop
- Block Transfers and Search
- 8-bit Arithmetic and Logical Operations
- 16/32-bit Arithmetic Operations
- 8-bit Bit Manipulation, Rotate and Shift
- 16-bit Rotates and Shifts
- Program Control
- I/O Operations (Internal)
- I/O Operations (External)
- CPU Control
- Decoder Directives

Host Interface

The host interface block in the Z382 includes the 16550 MIMIC, the Host DMA Mailbox, the ISA PnP Interface and the PCMCIA Interface.

16550 MIMIC

The Z382 includes a 16x550 MIMIC interface that allows it to emulate the operation of a PC UART. The interface allows the Z382 to be connected directly to an ISA bus or PCMCIA bus without any external circuitry. The MIMIC contains the 16x550 register set and the same interrupt structure. The data path allows parallel transfer of data to and from the register set by the internal processor of the Z382.

Control of the register set is maintained by six priority encoded interrupts to the Z382. When the PC Host reads or writes to certain MIMIC registers, an interrupt to the Z382 is generated. Each interrupt can be individually masked off or all interrupts can be disabled by writing a single bit.

Two 8-bit timers are also available to control the data transfer rate of the MIMIC interface. Their input is tied to a Baud Rate Generator in the MIMIC, allowing a wide range of data rates to be emulated. Two additional 8-bit timers are available for programming the FIFO timeout feature (Four Character-Time Emulation) for both the Receiver and the Transmitter FIFOs.

The 16550 MIMIC supports the PC Host interrupt structure by means of the Plug and Play ISA or PCMCIA interface modules. COM Port decoding is also provided by the same modules.

A bit in the Z382 System Configuration Register controls whether the registers of the 16x550 MIMIC interface are accessible in any page of I/O space, as on the Z8018x family, because only the lowest eight address lines are decoded, or whether A15-8 must be zero to access the registers.

The MIMIC Interface can transfer both Transmit and Receive data under control of the Z382's DMA channels, thus minimizing processor overhead and maximizing throughput in high-speed applications.

Figure 23 illustrates the 16550 MIMIC Block Diagram.

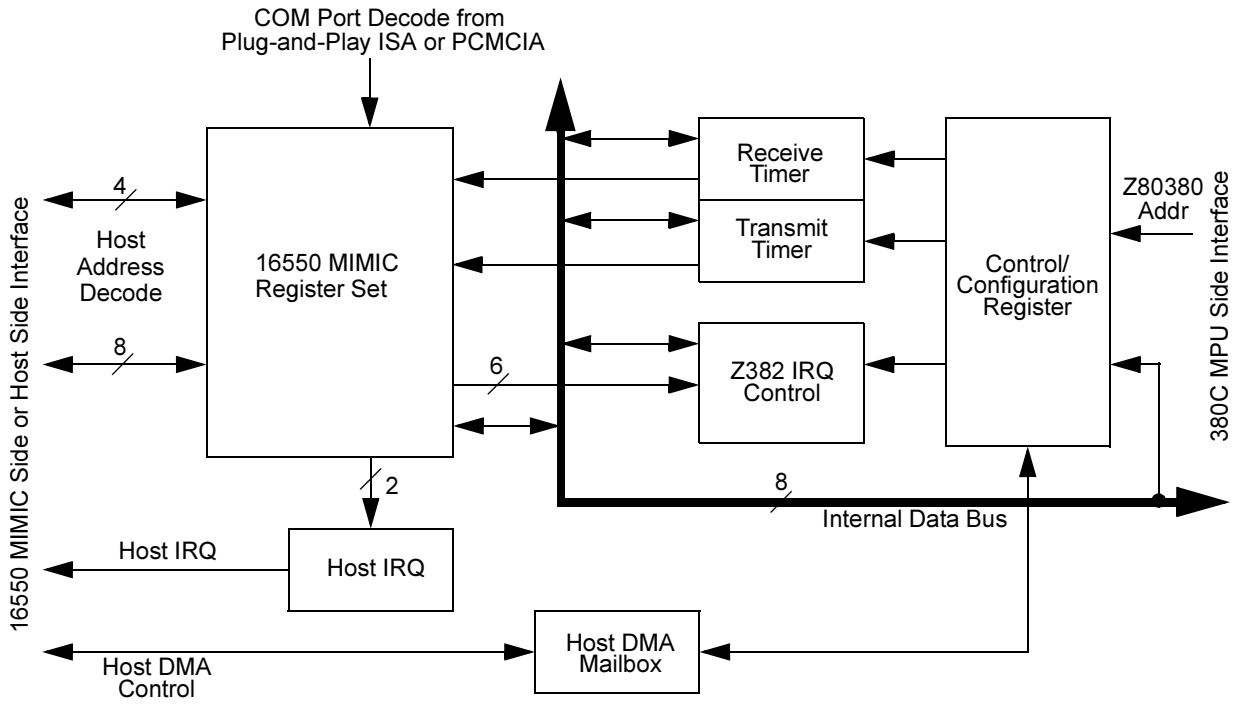


Figure 23. 16550 MIMIC Block Diagram

MIMIC Receiver FIFO

The receiver FIFO is 16-words deep and stores eight data bits and three error bits (Parity error, Framing error and Break detect) for each character received. The data and error bits move together in the FIFO. The error bits become available to the Host side of the interface when that particular location becomes the next address to read (top of FIFO). At that time these bits may either be read by the Host or they may cause an interrupt to the Host interface if so enabled. The error bits are set by the error status of the byte at the top of the FIFO but may only be cleared by reading the Line Status Register (LSR). If successive reads of the receiver FIFO are performed without reading the LSR, the status bits are set if any of the bytes read have the respective error bit set.

The Host interface may be interrupted when 1, 4, 8 or 14 bytes are available in the receiver FIFO. If the FIFO is not empty, but below the programmed trigger value, a timeout interrupt is available if the receiver FIFO is not written by the 380C or read by the Host by an interval determined by the Character Timeout Timer. This timer is an additional timer with 380C access-only which is used to emulate the 16550 four-character timeout delay. The timer receives the BRG as its input clock. Software determines the correct values to program into the Receiver Timeout Register

and the BRG to achieve the correct delay interval for timeout. These interrupts are cleared by the FIFO reaching the trigger point or by resetting the Timeout interval timer by a FIFO 380C write or Host read access.

With FIFO mode enabled, the 380C is interrupted when the receiver FIFO is empty. This bit corresponds to a Host read of the receiver buffer in non-FIFO (16450) mode. The interrupt source is cleared when the FIFO becomes non-empty or the 380C reads the IUS/IP register.

Figure 24 illustrates the 16550 MIMIC Receiver FIFO block diagram.

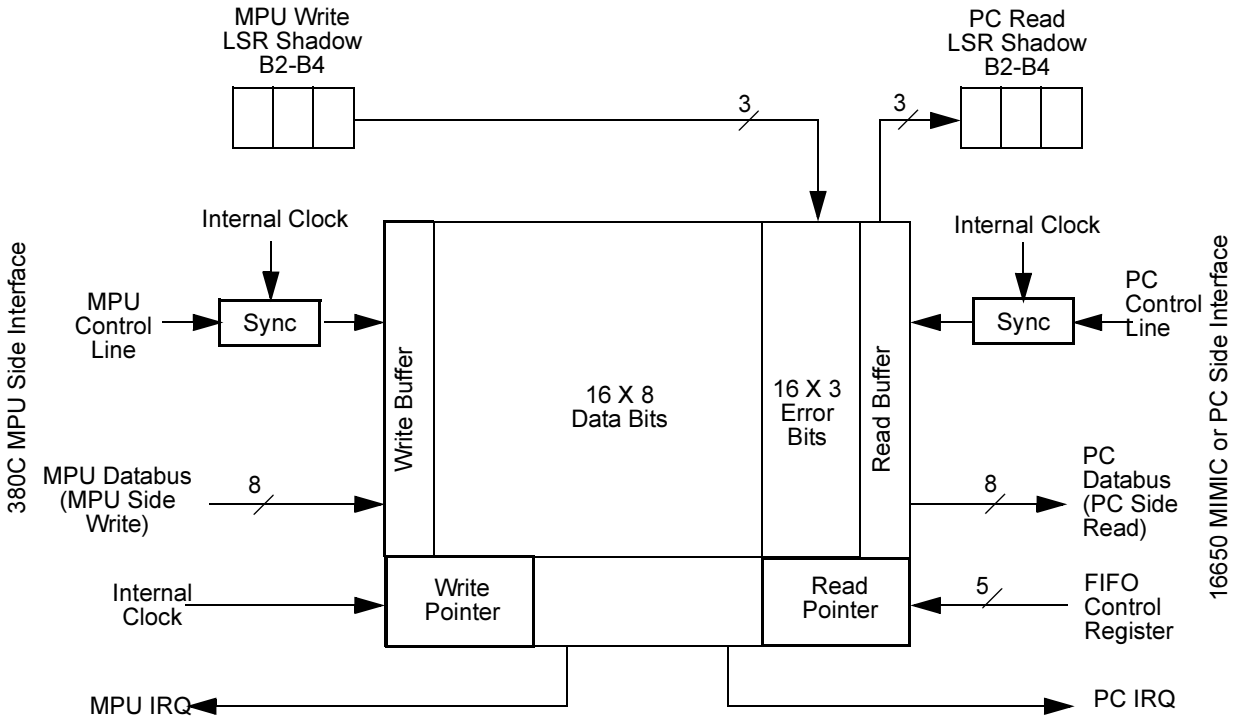


Figure 24. 16550 MIMIC Receiver FIFO Block Diagram

Transmitter FIFO

The transmitter FIFO is 16-bytes deep with Host write and 380C read access. In FIFO mode, the Host receives an interrupt when the transmitter FIFO becomes empty. The interrupt clears when the transmitter FIFO becomes non-empty or the IIR register is read by the Host. Figure 25 illustrates the MIMIC Transmitter FIFO block diagram.

On the 380C interface, the transmit FIFO can be programmed to interrupt the 380C on 1, 4, 8 or 14 bytes of available data. A timeout feature exists, the Transmitter Timeout Timer, which is an additional 8-bit timer using the BRG as the input

source. If the transmitter FIFO is non-empty and no Host write or 380C read of the FIFO has taken place within the timer interval, a timeout occurs, causing a corresponding interrupt to the 380C.

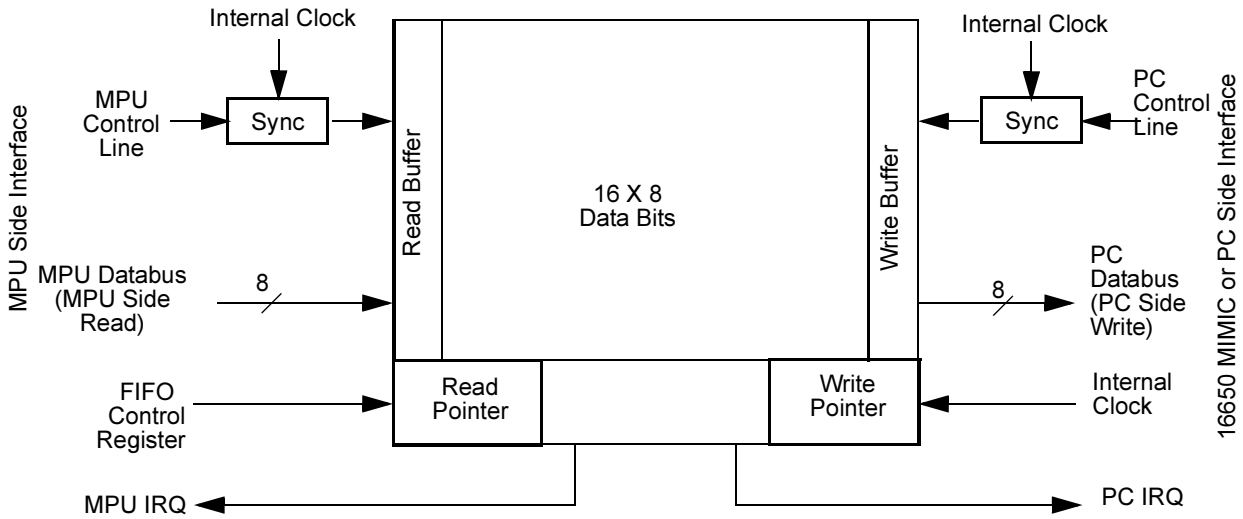


Figure 25. 16550 MIMIC Transmitter FIFO Block Diagram

Transmit And Receive Timers

Because of the speed at which data transfers can take place between the Z382 and the Host, there are two timers to alleviate any software problems that a high speed data transfer might cause. These timers allow the programmer to slow down the data transfer to simulate the MIMIC receiving and transmitting the data serially. The timers receive their input from the MIMIC BRG clock. This condition allows the programmer access to a 24-bit timer to slow the data transfers.

MIMIC Programming Registers

The MIMIC module contains a set of registers for programming various aspects of MIMIC operation. These are:

Table 28. MIMIC Programming Registers

Register	380C I/O Address
MIMIC Master Control Register	00FFH
MIMIC DMA Control Register	00EFH
MIMIC IUS/IP Register	00FEH
Interrupt Enable Register	00FDH

Table 28. MIMIC Programming Registers (Continued)

Interrupt Vector Register	00FCH
FIFO Status and Control Register	00ECH
Rx Timeout Time Constant Register	00EAH
Tx Timeout Time Constant Register	00EBH
Transmitter Time Constant Register	00FAH
Receiver Time Constant Register	00FBH

MIMIC-Host Interface Registers

In addition to the MIMIC programming registers, the Z382 contains a register set for interfacing with the Host by means of the MIMIC. These registers are used to emulate the 16550 UART so that the Host can access these registers in a manner similar to interfacing with the UART. This feature provides software compatibility with existing Host communication software. The registers are:

Table 29. MIMIC-Host Registers

Register	Host Address[1]	380C I/O Address
Receiver Buffer Register	00H[2]	00F0H
Transmit Holding Register	00H[2]	00F0H
Interrupt Enable Register	01H[2]	00F1H
Interrupt ID Register	02H	—
Line Control Register	03H	00F3H
Modem Control Register	04H	00F4H
Line Status Register	05H	00F5H
Modem Status Register	06H	00F6H
Scratch Register	07H	00F7H
Divisor Latch MS Byte	01H[3]	00F9H
Divisor Latch LS Byte	00H[3]	00F8H
FIFO Control Register	02H	00E9H
MIMIC Modification Register	—	00E9H

Note:
1. The host address is relative to the MIMIC base address decoded by the PnP ISA or PCMCIA modules in the Z382.
2. DLAB (LCR[7]) = 0.
3. DLAB (LCR[7]) = 1.

Baud Rate Generator

The Baud Rate Generator (BRG) provides emulation timing for the MIMIC. The BRG output clocks the MIMIC emulation counter, while the BRG itself is clocked by the BUSCLK output of the 380C. Two 8-bit registers are provided to program the BRG time constant. On-the-fly modification of the registers does not cause irregular BRG output.

Host DMA Mailbox

The Host DMA Mailbox facility provides a path for Host DMA data transfers separate from the MIMIC COM port. Commands and data flow over the COM port, while the DMA path can be used for other purposes. The Host DMA Mailbox feature includes control registers that allow Host DMA data transfer between Host memory and, for example, a modem speaker/microphone codec. Transfers are driven by the Host's DMA on one side; Z382 DMA channel(s) or programmed I/O

can be used on the other side. Thus, several modes of operation can be programmed:

- Host DMA Write, Z382 Polled Input
- Host DMA Read, Z382 Polled Output
- Host DMA Write with Z382 DMA
- Host DMA Read with Z382 DMA

On the ISA bus, the Z382 can use two independent DMA Mailbox facilities. When either of these facilities is enabled in the Plug and Play module, that module signals a DMA request by driving HDREQ0 or HDREQ1 High; if a facility is disabled, the corresponding HDREQ pin is 3-stated. A Low on one of the Acknowledge signals, HDACK0 or HDACK1, more or less simultaneously with HWR or HRD Low when the corresponding HDREQ line is being driven High, indicates a DMA cycle.

In a PCMCIA socket, only one DMA Mailbox can be used. When an option bit in one of the PCMCIA registers is 1, a DMA request is signalled by setting the INPACK output Low. A DMA cycle is signalled by having the PCREG line High while PCIORD or PCIOWR goes Low.

Plug-and-Play Interface

This module, with support from appropriate Z382-based firmware, complies with version 1.0a of the Microsoft™ /Intel™ “Plug-and-Play ISA” specification.

The Z382’s Plug-and-Play (PnP) module provides for I/O address decoding, interrupt channel selection and DMA channel selection. Pin limitations constrain the internal address decoding for I/O addresses to 12 bits. Because 16-bit decoding is preferred for full Plug-and-Play compliance, an additional input, HAEN, is provided which must be Low for a valid address decode. This input permits external decoding of HA15-12.

Figure 26 illustrates the Plug-and-Play Interface block diagram.

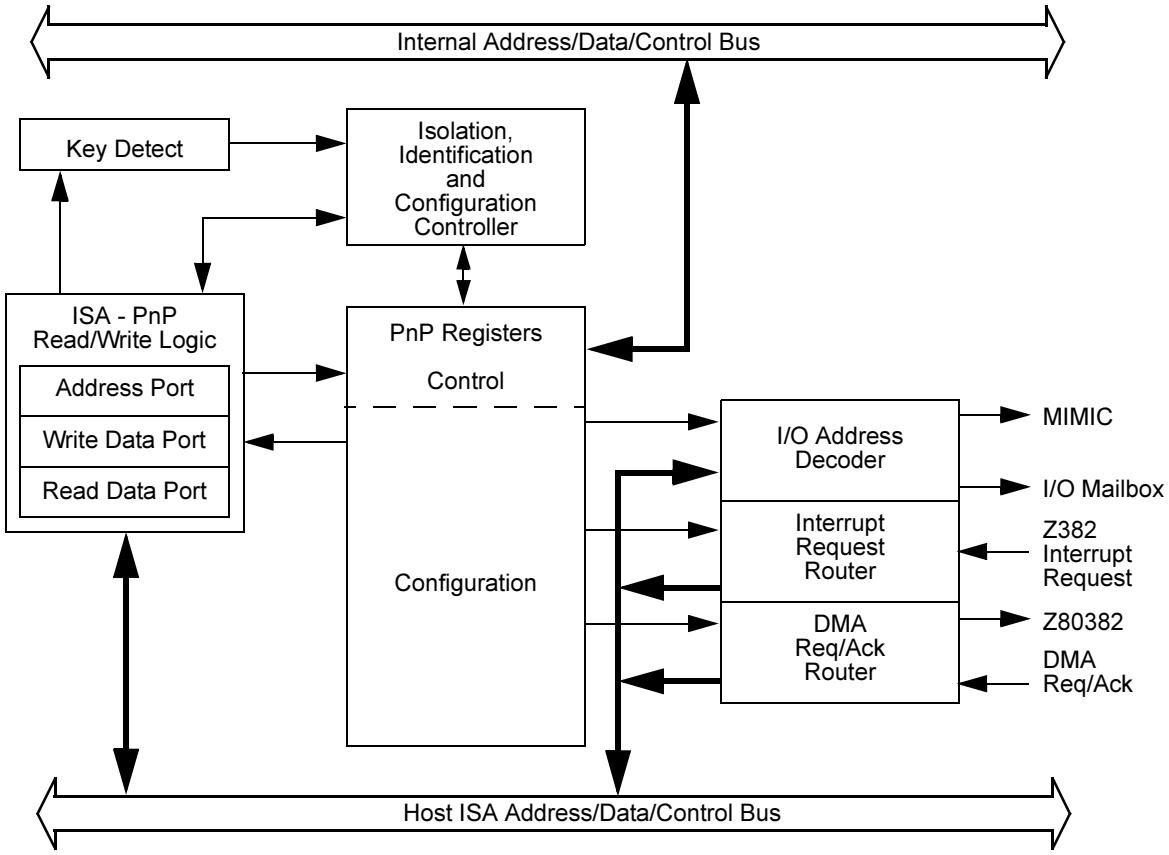


Figure 26. Plug-and-Play Interface Block Diagram

ISA Port

The PnP interface implements three 8-bit ports on the ISA bus. The Address port is a write-only port at the fixed address 0279H. The Write Data port is a write-only port at the fixed address 0A79H. The Read Data port is a read-only port at a programmable address among 0203H, 0207H, 020BH, ..., 03FFH.

The Host may write to the Address port for three reasons:

1. As part of sending an *Initiation key* to all the PnP cards in the system.
2. To select a register on one or all PnP cards as the destination of a subsequent write to the Write Data port.
3. To select a register on one card, or the *Isolation* facility on multiple cards, as the source of data in a subsequent read from the Read Data port.

Basic Operation

The space that the Host can access by writing to the Address port, and then writing to the Write Data port or reading from the Read Data port, is in large part also accessible to the 380C processor. Its 256 locations are sparsely populated with hardware registers.

After reset, and on command from host software, including in normal operation, the PnP interface is said to be in Wait for Key state. In this state, none of the PnP locations are accessible to the Host on the ISA bus. Before accessing any of these locations, the Host must first do a specified sequence of 34 write operations, called an Initiation key, to the Address port before it can access any of the registers of the PnP interface. The Initiation key is detected by the PnP interface hardware.

Each PnP card manufactured must have a non-zero 64-bit identity value that is divided into a 16-bit vendor ID, a 16-bit product ID including revision, and a 32-bit serial number. 380C firmware has complete control of this number; as no mechanism for storing or determining it is included in the PnP interface.

After sending an Initiation key, the Host can only access a few of these registers in a defined sequence. This sequence, called the Isolation protocol, selects the PnP card with the 64-bit value having the most low-order ones, among those in the system. The timing requirements of the Isolation protocol are quite slow compared to the speed of the 380C processor, and the 64-bit ID and an associated 8-bit checksum are sequenced to the PnP interface by the 380C, on a polled or interrupt-driven basis.

After isolating a card by means of the Isolation protocol, host software assigns the isolated card a Card Select Number (CSN), starting with address 01H and ascending for subsequent cards. Assigning a CSN eliminates the card from future repetitions of the protocol. Then, or later, host software reads the characteristics of the card, called the Resource Data, in a handshake manner with 380C firmware.

Host software repeats this process until it determines that it has scanned all of the PnP cards in the system. The host software allocates resources, including memory and I/O space addresses, interrupt levels, and DMA channels, and uses the various cards' CSNs to write these allocations to Configuration registers in the PnP register space.

Finally, host software places all the PnP interfaces in the system back in Wait for Key state, in which the interfaces perform address decoding and interface the interrupt and DMA requests and acknowledgments, but have no affect on other system operations. If the host software thereafter determines that the system requires reconfiguration, another Initiation key is sent. In this case, however, a specific card is addressed using the previously assigned CSN.

Configuration Registers

The following Configuration registers are implemented in the Z382 to provide for the resources required by the host to interface to the host-accessible functions within the chip:

- I/O Mailbox I/O Address
- MIMIC I/O Address
- Interrupt Request Level– This register can be selected to be output on either of the two available interrupt output lines. A unique Z382 feature allows these two pins to be configured to be any two of the ISA-bus interrupt lines.
- DMA Channel 0, DMA Channel 1– A unique Z382 feature allows the two DMA pin pairs to be configured to be any two of the seven ISA-bus DMA channels.

Host writes to the Configuration registers are effective immediately, in hardware, so there is no urgent need for the 380C processor to translate them into other register values. But the 380C processor can use the interrupt that occurs when the Host terminates Configuration state to examine what the Host has done to the Configuration registers, and operate accordingly in the future.

PCMCIA Interface

The PCMCIA Interface block integrates all the functions necessary for the operation of I/O interface cards in a PCMCIA 2.0 and 3.0 socket. These functions are:

- PCMCIA Interface Control
- Attribute Memory
- Configuration Registers
- I/O Interface
- Configurable Address Decoder
- Configurable Interrupt Logic
- Z380 Interface

Figure 27 illustrates the PCMCIA Interface block diagram

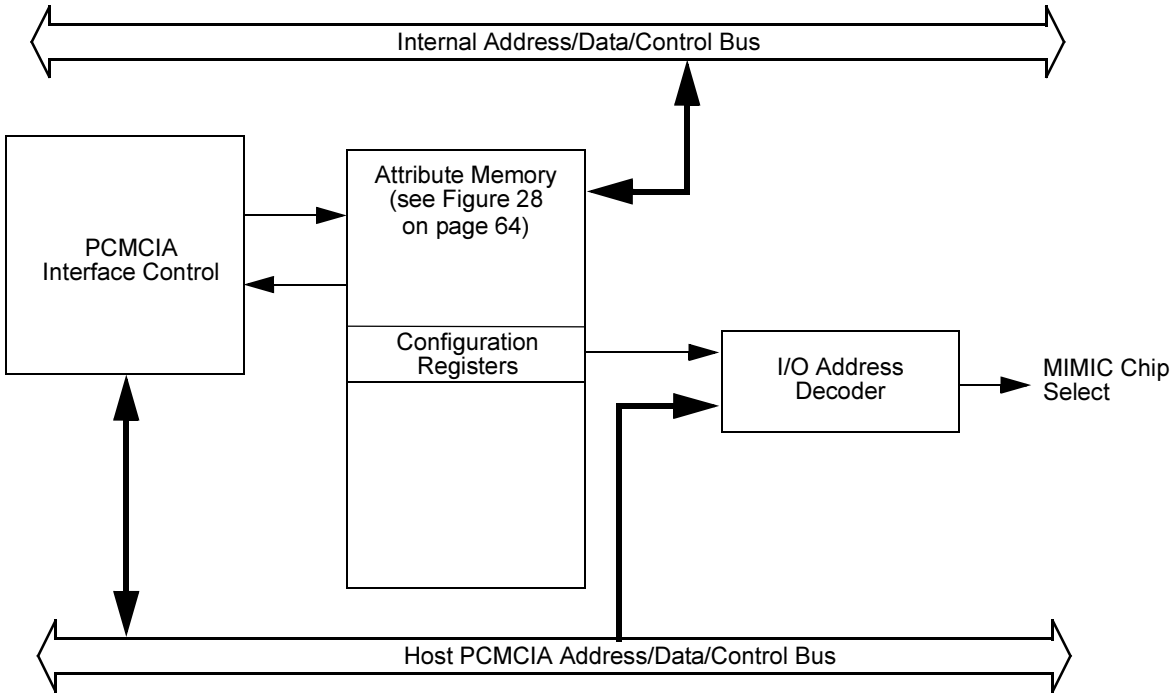


Figure 27. PCMCIA Interface Block Diagram

PCMCIA I/O Interface Control

The I/O interface contains the main functionality of the PCMCIA block. The interface decodes addesses for I/O accesses by the Host according to the PCMCIA standard. The Host writes to the Configuration Option Register an index to select the base address of the desired I/O address range. After configuration, I/O accesses to this address range are recognized, and the MIMIC chip select is asserted when a valid I/O access is performed and the address is in the configured address range.

Attribute Memory

The attribute memory is the primary mechanism for transfers of configuration data and status between the host system and the PCMCIA card. As depicted in Figure 28, the attribute memory is segmented into several sections. The Card Information section is 240 bytes of RAM which is loaded by the 380C with information describing the card and its resource requirements, data needed by the Host to configure the card. A portion of the attribute memory allows the host to access the I/O Mailbox registers. Lastly, sections in the attribute memory space are assigned to the Configuration Registers and the Base Address Registers.

On the Host side, attribute memory is accessible only on even byte addresses. On the 380C side attribute memory can be accessed as bytes or words.

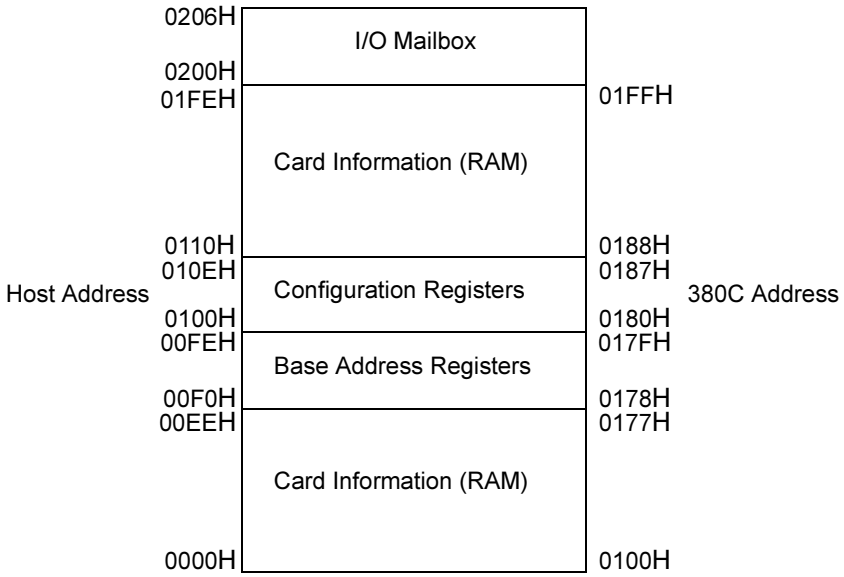


Figure 28. PCMCIA Attribute Memory Organization

Base Address Registers

These seven registers are written by the 380C with the base addresses of 8-byte windows in the host's I/O address space which the host can use to communicate with the host-accessible registers in the MIMIC.

Configuration Registers

There are five configuration registers of the PCMCIA 3.0 standard and in addition a version number register, two image base address registers, and the seven base address registers described in the previous paragraph. The Host accesses these registers to configure the interface and to retrieve status.

Configuration Option Register. This register is used on one side to configure the PCMCIA interface, controlling items such as type of interrupt, DMA enable, and selection of the Base Address Register. On the other side, a reset can be triggered by setting a certain bit.

Card Configuration and Status Register. This register contains information about the status of the interface, including whether certain signals have changed, interrupts, and power down.

Pin Replacement Register. This register is used to provide the status information which is otherwise provided on the PCIRQ pin (RDY/BSY).

Socket and Copy Register. This register is implemented for PCMCIA hosts expecting this optional register in a PCMCIA card. The register has no function in the Z382.

Extended Status Register. This register is used to enable and provide status information of external events.

Image Base Address Registers. These registers deliver a copy of the configured base address.

Interface Version Number Register. This register provides the version number of the PCMCIA interface. It also contains a bit which can be written to disable attribute memory write protection, allowing the host to write to the attribute memory.

Z380 Control Register (ZCR). This register controls the functions of the PCMCIA block by means of the Z380 controller. Accessible only to the 380C, this register controls access to the attribute memory by the 380C and allows the 380C to signal major status changes to the host.

Decoding and Routing Functions

The PCMCIA interface uses the values programmed in the Configuration Registers to decode a MIMIC chip select when the host I/O address signals match the programmed conditions.

Unlike the Plug-and-Play interface, the PCMCIA interface does not perform any routing functions on interrupt and DMA control signals. These functions are performed at the PCMCIA socket controller on the host side.

DMA Channels

The DMA channels of the Z382 build on ZiLOG's experience with the Z16C32 IUSC. They have only one mode of operation, which combines features of the IUSC's Array and Linked List modes. Each DMA channel has a pointer into a list structure, entries in which contain the addresses and lengths of data buffers.

Because the on-chip peripherals of the Z382 all operate with 8-bit data only, particularly the HDLC channels which the DMA channels are primarily intended to serve, the DMA channels also perform only 8-bit data transfers when operating with data buffers. However, because fetching a new list entry is an overhead operation that can compromise maximum data rates, list accesses use 16-bit transfers.

DMA Channel/Device Interface

The interface between the DMA channel and its client device includes six lines:

Table 30. DMA Channel/Device Interface

Data Request	Device to DMA
Terminate	Device to DMA
Type Fetch	DMA to device
Data Acknowledge	DMA to device
End of Buffer	DMA to device
Store Status	DMA to device

All of these lines are bussed, and are driven by the DMA channel and its client device that are currently selected by the DMA scanner.

DMA Operation

A DMA channel begins operation when software loads an address into its List Address Register (LAR). Writing the final (MS) byte of this register sets the channel's Run bit, which causes a request for bus access from the processor. When the processor grants bus access, the DMA channel proceeds to fetch the first List Entry from memory, beginning at the address in the LAR.

List entries (described in Figure 29) always begin at an 8-byte boundary, that is, at an address having its LS three bits 000. The general format of a list entry includes eight bytes:

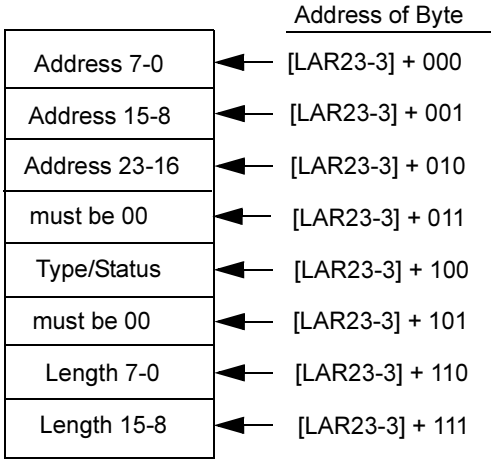


Figure 29. General Format of a DMA List Entry

The Type/Status byte defines various kinds of list entries, as follows:

Table 31. Type/Status Definitions

00H	End of List
01H	Transfer in List
02H	Ready Buffer, no Command, no End of Buffer notification
03H	Ready Buffer, no Command, notify device at End of Buffer
04H	Buffer in Progress
05H	Completed Buffer (no Status)
40H-7FH	Ready Buffer, with Command, no End of Buffer notification
80H-BFH	Ready Buffer, with Command, notify device at End of Buffer
C0H-FFH	Completed Buffer (with Status)

Upon fetching any Type/Status value except Transfer in List or Ready Buffer, the DMA channel clears its Run bit and requests an interrupt if its List Interrupt Enable bit is 1. This checking of the Type/Status byte helps prevent disorderly operation as well as buffer-ring wraparound.

Upon fetching a Transfer in List entry, the DMA channel fetches the Address portion of the entry, loads it into its LAR, and proceeds to fetch another list entry from that address. This is the mechanism by which buffer rings and linked lists are constructed.

If software requires knowledge of when a certain amount of data has been sent or received, such as an Address field in a received HDLC frame, a buffer of that length can be created with its own list entry. The DMA channel can provide an interrupt at the end of the buffer if desired.

When a DMA channel fetches a Type/Status byte from memory, it asserts the Type Fetch signal to its client device. This action prompts the client device to capture the Command if bits D7-6 of the Type/Status byte are `01` or `10`.

For example, the HDLC Transmitter uses the three LS bits of such a Type/Status byte to indicate how many bits to send from the last byte of the frame. The HDLC Receiver does not use any Command bits, so that Ready Buffer codes, with and without Command, are equivalent for HDLC reception.

Upon fetching any Ready Buffer entry, the DMA channel rewrites the Type/Status byte to the Buffer in Progress code, and then fetches the Address and Buffer Length fields and loads them into its Buffer Address and Length Registers (BAR and BLR) respectively. Thereafter the DMA channel transfers data into or out of the buffer, under control of the Data Request line, from its client device. If there is

no request at this point, as is typically the case when software initiates a Receive channel, the DMA channel relinquishes bus control to the processor or another DMA channel, and goes idle until the device asserts Data Request and/or Terminate. For Type/Status bytes requesting Notify device at end of buffer, the DMA channel asserts its client's End of Buffer line.

When a DMA channel has been initiated and has fetched its first list entry, no further action is taken unless and until its client device asserts Data Request and/or Terminate. When the client device does so, the DMA channel requests bus access from the processor. When access is granted, or when it is continuing operation after fetching a list entry, the DMA channel proceeds as follows:

If the device is asserting Data Request, with or without Terminate:

1. The DMA channel asserts Data Acknowledge to the device.
2. If its BLR indicates the buffer is ending, and the Status/Type byte for this buffer indicates Notify Device, the DMA channel also asserts the End of Buffer signal.
3. At the same time, the DMA channel places the address in its BAR on the address bus, and sets the control signals for a memory read or write per the I/O bit in its DMA Control/Status Register (DCSR).
4. Depending on the data direction, Data Acknowledge causes the device either to provide a byte of data on the data bus, or to capture a byte of data from the data bus.

How (and whether) a client device uses End of Buffer is device-dependent. The HDLC Transmitter passes this indication through its TxFIFO, and terminates the Tx frame after sending the data with which the DMA channel asserted End of Buffer. Because of this facility, the only time that an underrun occurs at the HDLC Transmitter is when the DMA does not provide data fast enough, inside a frame.

The HDLC Receiver does not do anything with End of Buffer, so Ready Buffer codes with or without EOB are equivalent for HDLC receiving.

At the end of each data transfer, the DMA channel increments the BAR by 1 and decrements the BLR by 1.

If the device signalled Data Request but not Terminate, and the Buffer Length Register has not been counted down to zero, and the Burst bit in the channel's DCSR is set, the DMA channel checks Data Request again. If Burst is 0, and/or if the device negates Data Request, the channel gives the bus back to the processor or another DMA channel, or it performs another data transfer.

If the device signalled Data Request, but not Terminate, and the Buffer Length Register has now been counted down to zero, the DMA channel proceeds as follows:

1. It puts the address of the Type/Status byte (from the LAR) on the address bus, and writes the code for Completed Buffer (no Status) into that byte.
2. If the DMA channel's Buffer IE field indicates `Interrupt for all buffers`, or `Interrupt for` Notify buffers and this was a Notify buffer, it sets its IP bit to request an interrupt.
3. It increments the LAR to the address following this list entry, and goes back to fetch a new list entry from that address, as described above.

Terminate

The HDLC receiver asserts this signal for an End of Frame, Abort, or Overrun condition. The HDLC Transmitter does so for an Underrun condition. After the DMA channel transfers a byte, if the device signals Data Request and Terminate, or if the device signals Terminate without Data Request, the DMA channel proceeds as follows:

Note: If the device encounters an error from which an operation can not continue without processor attention, after signalling, the device must refrain from asserting Data Request until software has done so. The HDLC Transmitter performs this function for Underrun.

1. The device places the address of the Length field on the address bus, and writes the current (16-bit) value in its BLR to memory at that address and the next higher address. This value enables software to determine how much data was actually written into, or read out of, this buffer.
2. The device places the address of the Type/Status byte on the address bus, sets the control signals for a memory write, signals Completed Buffer (with Status), and asserts the Store Status signal to the device.
3. The device, in response to Store Status, can place up to 6 bits of status on D5-0. For the HDLC receiver, this status includes Overrun, End of Frame, Abort, CRC Error, and the residual bit count. For the HDLC Transmitter, only Underrun will prompt a Terminate indication, so the specific status bits are unimportant.
4. After the Type/Status byte has been written, the DMA channel advances the LAR over this list entry, in other words, to the address of the next entry.
5. If the DMA channel's Buffer IE field indicates anything other than No Buffer Interrupts, it requests an interrupt.
6. The DMA channel then fetches another list entry from the address in the LAR, as described above.

Per-Channel Registers

There are eight DMA channels in the Z382. Each channel includes the following registers:

Table 32. Per-Channel Registers

List Address Register	(LAR, 21 bits)
Buffer Address Register	(BAR, 24 bits)
Buffer Length Register	(BLR, 16 bits)
DMA Control/Status Register	(DCSR, 8 bits)

The LAR and DCSR are read/write registers. Software tracks the progress of a DMA by monitoring its LAR. BARs and BLRs are accessible only by using special modes selected in the centralized DMA Control Register. The DMA channel stores ending BLR values in the list.

List Address Register

A three-byte register whose 21 most significant bits contain the base address of the current list. The DMA channel begins operation when the 380C writes the most significant byte of this register. The DMA controller updates this register as it processes new lists in response to links from previous lists. The three LS bits of the LAR are ignored on writing, and are read back as 100 (pointing at the current Type/Status byte in the list).

Buffer Address Register

The DMA controller loads the initial value of the current buffer address into this register from the address field of the current list. At the end of each data transfer, the DMA channel increments the BAR by one.

Buffer Length Register

The DMA controller loads the initial value of the current buffer length into this register from the buffer length field of the current list. At the end of each data transfer, the DMA channel decrements the BLR by one.

DMA Control/Status Register

Controls items such as I/O Direction, enabling/disabling BURST Mode, and enabling and disabling interrupts. Also provides certain per channel DMA and interrupt status conditions.

Centralized DMA Registers

Two registers listed below provide overall DMA subsystem control and status:

DMA Control Register	(DMACR)
DMA Vector Register	(DMAVR)

DMA Control Register (DMACR)

This register controls when bus control is returned to the 380C processor after a DMA channel has operated. It also provides modes whereby the Buffer Address and Buffer Length per-channel registers can be read and written.

DMA Vector Register (DMAVR)

This register contains the base interrupt vector for the DMA channels. It also identifies, during an interrupt acknowledge cycle, the interrupting DMA Channel.

Serial Communication Channels

The Z382 provides several means of serial data communications. These are the Asynchronous Serial Communication Interface (ASCI), the HDLC controllers, the GCI/SCIT interface and the Clocked Serial I/O Channel.

Asynchronous Serial Communications Interface (ASCI)

The Z382 provides two independently programmable ASCIs, each including a flexible baud rate generator. Key ASCI features include:

- Full-duplex operation
- Programmable data format
  - 7- or 8- data bits with optional ninth bit for multiprocessor communication
  - One or two stop bits
  - Odd, even or no parity
- Programmable baud rate generator
  - Divide-by-one, divide-by-16 and divide-by-64 modes
  - Up to three modem control signals per channel, depending on operating mode of the Z382
- Programmable interrupt conditions
- Four level data/status FIFOs for the receivers

- Receive parity, framing and overrun error detection
- Optional operation with on-chip DMA controllers

Figure 30 illustrates the major functional blocks within the ASCI.

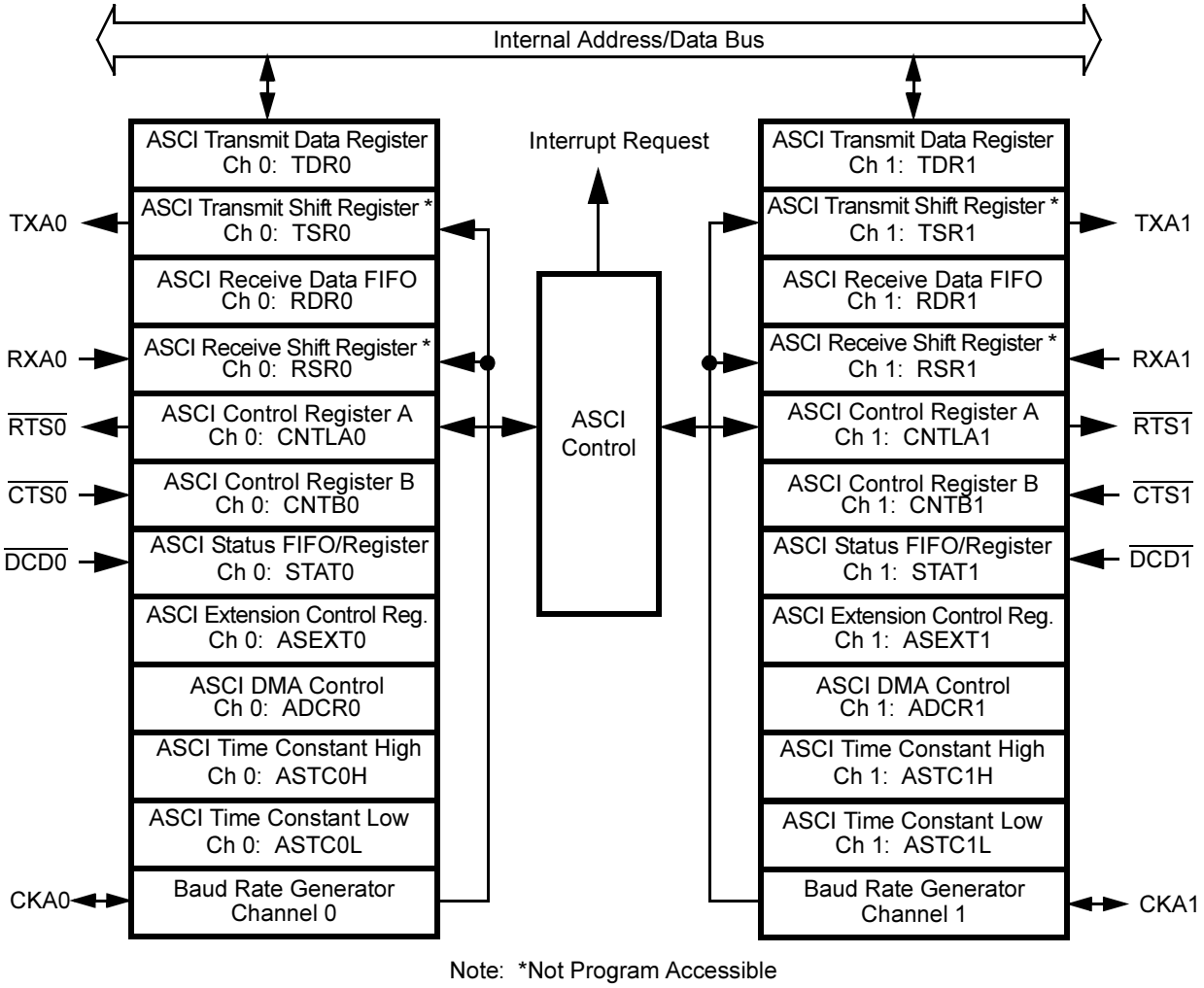


Figure 30. Asynchronous Serial Communications Interface (ASCI) Block Diagram

Transmit Data Register

Data written to the ASCI Transmit Data Register (TDR) is transferred to the Transmit Shift Register (TSR) as soon as the TSR is empty. Data can be written while the TSR is shifting out the previous byte of data, providing double buffering for the transmit data.

Data transfers into the TDR can be performed using I/O instructions or by using one of the DMA channels. This DMA process loads characters into the TDR as an associated status bit indicates that it has become available for data.

Transmit Shift Register

When the ASCI Transmit Shift Register receives data from the ASCI Transmit Data Register, the data is shifted out to the TxA pin. When transmission is complete, the next byte (if available) is automatically loaded from the TDR into the TSR and the next transmission starts. If no data is available for transmission, the TSR idles at a continuous High level.

Receive Shift Register

When the receiver is enabled, the RXA pin is monitored for a Low. One-half bit time after a Low is sensed at RXA, the ASCI samples RXA again. If RXA has returned to High, the ASCI ignores the previous Low and resumes looking for a new one. If RXA is still Low, the ASCI considers this bit a start bit and proceeds to clock in the data based upon the internal baud rate generator or the external clock at the CKA pin. The number of data bits, parity, multiprocessor and stop bits are selected by means of control bits in the CNTLA and CNTLB registers.

After the data is received, the appropriate MP, parity and one stop bit are checked. If there is an empty position available data and any errors are clocked into the receive data and status FIFOs during the stop bit. Interrupts, Receive Data Register Full Flag, and DMA requests also go active during this time. If there is no space in the FIFO at the time that the RSR attempts to transfer the received data into it, an overrun error occurs.

Receive Data FIFO

When a complete incoming data byte is assembled in the RSR, it is automatically transferred to the FIFO, which serves to reduce the incidence of overrun errors. The top (oldest) character in the FIFO (if any) can be read by means of the Receive Data Register (RDR).

An overrun occurs if the receive FIFO is still full when the receiver completes assembly of a character and is ready to transfer it to the FIFO. If this occurs, the overrun error bit associated with the previous byte in the FIFO is set. The latest data byte is not transferred from the shift register to the FIFO in this case, and is lost. After an overrun occurs, the receiver does not place any further data in the FIFO until the last good byte received has come to the top of the FIFO and sets the Overrun latch. Software then clears the Overrun latch.

When a break occurs (defined as a framing error with the data equal to all zeros), the all-zero byte with its associated error bits are transferred to the FIFO if it is not full. If the FIFO is full, an overrun is generated, but the break, framing error and

data are not transferred to the FIFO. Any time a break is detected, the receiver can not receive any more data until the RXA pin returns to a High state.

Data transfers from the receive FIFO can be performed using I/O instructions or by using one of the DMA channels. This DMA process reads characters from the RDR as an associated status bit indicates that data is available. The RxDMA request is disabled when any of the error flags (PE, FE or OVRN) is set, so that software can identify with which character a problem is associated.

ASCI Status FIFO/Register

This FIFO contains Parity Error, Framing Error, Rx Overrun, and Break status bits associated with each character in the receive data FIFO. The status of the oldest character (if any) can be read from the ASCI status register, which also provides several other status conditions which are non-FIFO.

The outputs of the error FIFO go to the set inputs of software-accessible error latches in the status register. Writing a 0 to the Error Flag Reset (EFR) bit in CNTLA is the only way to clear these latches. In other words, when an error bit reaches the top of the FIFO, the bit sets an error latch. If the FIFO has more data and the software reads the next byte out of the FIFO, the error latch remains set until the software writes a 0 to the EFR bit. The error bits are cumulative, so if additional errors are in the FIFO they set any unset error latches as they reach the top.

Baud Rate Generator

The baud rate generator (BRG) features two modes. The first mode provides a dual set of fixed clock divide ratios. In the second mode, the BRG is configured as a sixteen-bit down counter that divides the processor clock by the value in a software accessible, sixteen-bit, time constant register. This condition allows virtually any frequency to be created by appropriately selecting the main processor clock frequency. The BRG can also be disabled in favor of an external clock on the CKA pin.

The Receiver and Transmitter subsequently divide the output of the Baud Rate Generator (or the signal from the CKA pin) by 1, 16 or 64 under program control.

ASCI Register Set

Each ASCI contains a set of registers for programming various aspects of its operation. These registers are:

- Control Register A
- Control Register B
- Time Constant High Register

- Time Constant Low Register
- Extension Control Register
- Status Register
- Receive Data Register
- Transmit Data Register
- DMA Control Register
- Control Register A

HDLC Serial Channels

The Z382 features three high-speed serial channels, each comprised of a transmitter and a receiver, which can operate in HDLC or transparent (unframed) modes. All data transfers to and from the HDLC channels are carried out by the DMA channels. Thus, each HDLC channel requires an assigned DMA channel to perform its function. Facilities for interrupt-driven or polled transfer of HDLC data are not provided.

Software selects whether each channel's I/O is on device pins or on the internal TDM highway (the GCI/SCIT bus in the Z382). If device pins are used, they can be configured as either a classic synchronous serial interface, or as the interface to an external TDM highway or highways. The differences in pin use are as follows:

Table 33. Pin Use Differences in TDM/Full Time Operation

Pin	TDM Operation	Full Time Operation
TxD	3-stated outside the time slot.	Driven full time
RxD	Sampled within the time slot.	Sampled in every bit time
RxC/BCL	Common clock for Rx and Tx.	Rx Clock, optional Tx Clock
TxC/FSC	Frame Sync pulse for Rx and Tx.	Tx Clock in or out.
TxEN	Asserted within the time slot, optional enable for an external driver.	Asserted whenever Tx is enabled.

Eight-character FIFOs on both the transmit and receive side reduce the possibility of overrun and underrun conditions to a minimum, at data rates up to and beyond E1 (2.048 Mbps). Figure 31 illustrates the HDLC Channel block diagram.

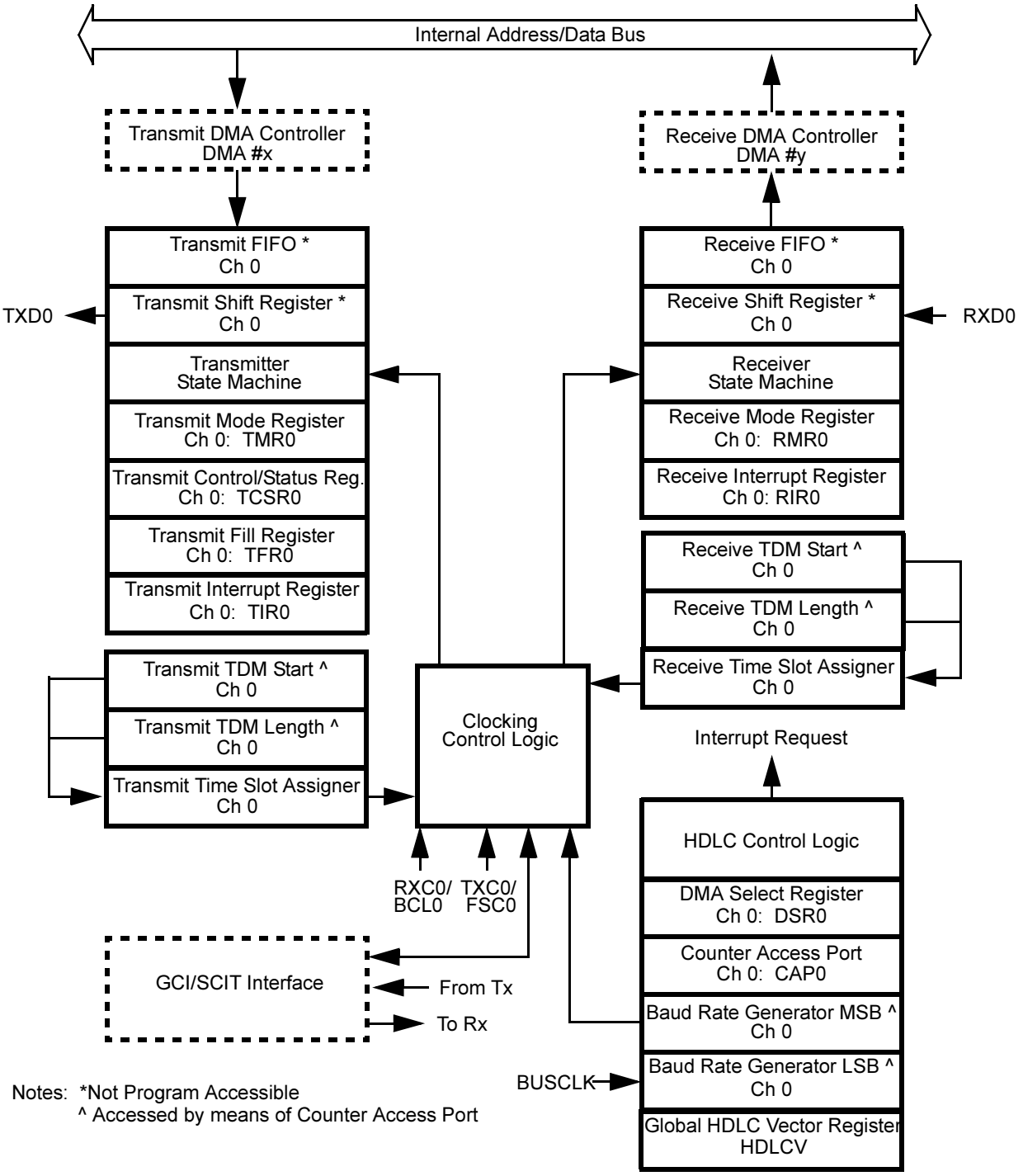


Figure 31. HDLC Channel Block Diagram (One of Three Channels Illustrated)

Interface with a Common TDM Module (for example, GCI/SCIT)

The interface between an HDLC channel and the GCI/SCIT module includes:

Table 34. HDLC Channel/GCI/SCIT Interface

TxD	A bussed line onto which HDLC Transmitters place data in their time slots, as directed by software programming.
RxD	A bussed line from which HDLC Receivers take data in their time slots, as directed by software programming.
BCL	A common bit clock for HDLC Transmitters and Receivers. Transmitters change data on TxD on falling edges of BCL, and Receivers sample data from RxD on rising edges of BCL.
FSC	Frame Sync, synchronous to BCL. Transmitters and Receivers measure their time slots independently from the rising edge of this signal. The duration of FSC can be one or more BCL cycles.
$\overline{\text{TxEN}}$	An output from each Transmitter to the common TDM module, indicating its time slot, that is, when it is placing data on TxD.

TDM Processing

When the Transmit (Receive) TDM Length register is non-zero, the Transmitter (Receiver) activates its Time Slot Assigner to clock Tx (Rx) data only within the time slot. If a TDM Start register is non-zero, after each pulse on Frame Sync, the Time Slot Assigner blocks clocking for the number of bits specified by the TDM Start register. Then, or immediately at Frame Sync if the Start value is zero, clocking is enabled for the number of bits specified by the TDM Length register. Thereafter, clocking is blocked until the next Frame Sync pulse. For example, the Start and Length values for the GCI subchannels are:

Table 35. GCI Subchannel Start and Length Values

Channel	Start	Length
B1 (64K bps)	0	8
B1 (56K)	0	7
B2 (64K)	8	8
B2 (56K)	8	7
D	24	2
IC1	32	8
IC2	40	8

Type/Status Bytes in DMA Lists

Note: Please refer to the description of Type/Status bytes in the section on the DMA channels (page 68) in conjunction with this topic.

Type/Status Bytes in Transmitter DMA Lists. In HDLC mode, a frame to be transmitted can be contained in one or more DMA buffers. The DMA list entry for the last (or only) buffer of a frame should have its Type/Status byte coded as `Ready Buffer, notify at End of Buffer`. The Transmitter sends the CRC (if enabled) and a closing Flag after the last byte of the buffer. Buffers that do not include the end of a frame should have their Type/Status bytes coded as Ready Buffer, no End of Buffer Notification.

Two control fields for the Transmitter do not reside in processor-accessible register bits, but can be controlled separately for each frame in Type/Status bytes in the DMA list:

- How many bits the Transmitter sends from the last byte of the frame.
- Whether the Transmitter sends its accumulated CRC at the end of the frame.

Either of these items can be changed automatically from one frame to the next if the Type/Status byte for the frame is coded as Ready Buffer, with Command and the control bits of that byte are set appropriately.

In HDLC modes or in TRANSPARENT mode with the Underrun Wait bit set to 1, completed Buffer codes in Type/Status bytes in Transmitter DMA lists are stored as `with Status` if the Transmitter encountered an Underrun while sending the data in the buffer. In all other cases, Type/Status bytes in Transmitter DMA lists are stored as no status.

Type/Status Bytes in Receiver DMA Lists. HDLC receivers do not use the Command nor End of Buffer notification features of the DMA channels. Thus all `Ready Buffer` codes in Type/Status bytes in Receiver DMA lists are equivalent.

A received frame can be contained in one DMA buffer, or can span two or more buffers. The end of a frame always makes the Receiver terminate its current DMA buffer and store frame status in its Type/Status byte.

When a buffer is filled with receive data, without the last character of the frame being stored in that buffer, that buffer's Type/Status byte is stored as Completed Buffer (no Status). Buffers that include the last character of a frame, and buffers that could not be completed because the Receiver encountered an Overrun condition, are stored as Completed Buffer (with Status). The least significant five bits of such a Type/Status byte indicate the status of the buffer.

Baud Rate Generator and DPLL

If an HDLC channel's Tx clock is taken from its Baud Rate Generator (BRG), and/or its Rx clock is taken from its DPLL, then the channel's BRG operates. A BRG counts down from the 16-bit value programmed into its Time Constant LS and MS registers, using the processor's BUSCLK. Each time the value is zero, the BRG toggles its output to the DPLL, and one clock later it reloads the value from the Time Constant registers.

If an HDLC channel's Rx clocking is taken from its DPLL, software programs the channel's Time Constant registers with a 16-bit value corresponding to 16 times the nominal data rate. When the DPLL detects a change on the raw Rx Data (before NRZI decoding), a counter is cleared that is incremented at 16X the nominal bit rate. Half a bit time later, an active edge is provided on its Rx clock output. In the absence of further data transitions the Rx clock is provided as the BRG output divided by 16.

Per-Channel Registers

Each HDLC channel includes the following I/O-mapped registers that can be read and written by the 380C processor:

- Transmit Mode Register
- Transmit Control/Status Register
- Transmit Interrupt Register
- Transmit Fill Register
- Receive Mode Register
- Receive Interrupt Register
- DMA Select Register
- Counter Access Port

Transmit Mode Register. This register selects the main operating mode of the Transmitter (TRANSPARENT, HDLC, NRZI HDLC), its I/O configuration (TDM, I/O by means of device pins, and so on), when DMA data transfers are requested, and action to be taken if an underrun occurs.

Tx Control/Status Register. This register controls the minimum number of bits sent between frames and the minimum number of bits sent after the Transmitter is enabled before the first data character of a frame is sent, what the Transmitter sends between frames, and the type of CRC used. Feedback is also provided on the current state of the transmitter.

Transmit Interrupt Register. This register controls and provides status of potential interrupting conditions in the transmitter. The mechanism is also provided for clearing conditions which are causing an interrupt.

Transmit Fill Register. This register holds a character that can be sent between frames in HDLC mode, or in case of an Underrun, in TRANSPARENT mode.

Receive Mode Register. This register selects the main operating mode of the Receiver (TRANSPARENT, HDLC, NRZI HDLC), its I/O configuration (TDM, I/O by means of device pins, and so on), when DMA data transfers are requested, when the receiver begins assembling characters, when it is switched from the Inactive state to TRANSPARENT mode, and the type of CRC used in HDLC modes.

Receive Interrupt Register. This register handles incoming interrupts. Most of the interrupt requirements for HDLC reception can be handled by enabling Status interrupts in the DMA channel associated with each Receiver. The only Receiver interrupt condition that is not handled by this means is the Idle condition. Idle interrupts are controlled by this register. This register also allows several commands which deal with interrupts and HUNT mode to be issued to the receiver.

DMA Select Register. This register selects the DMA channels to be used by the receiver and transmitter and enables their operation.

Counter Access Port. This port allows the 380C to write and read the starting values for various counters in the HDLC channel. These counters are the Baud Rate Generator time constant, the Transmitter TDM start and length values, and the Receiver TDM start and length values.

Global HDLC Vector Register. This register provides the base interrupt vector for the HDLC channels and identifies the HDLC device which is causing an interrupt to be issued.

GCI/SCIT Interface

GCI/SCIT Frame Structure (Terminal Mode)

GCI/SCIT includes three sub-frames called channels 0, 1, and 2, each containing 32 bits. This 12-byte frame is repeated at a rate of 8 KHz, giving an aggregate data rate of 768 Kbps. Figure 32 depicts the frame structure.

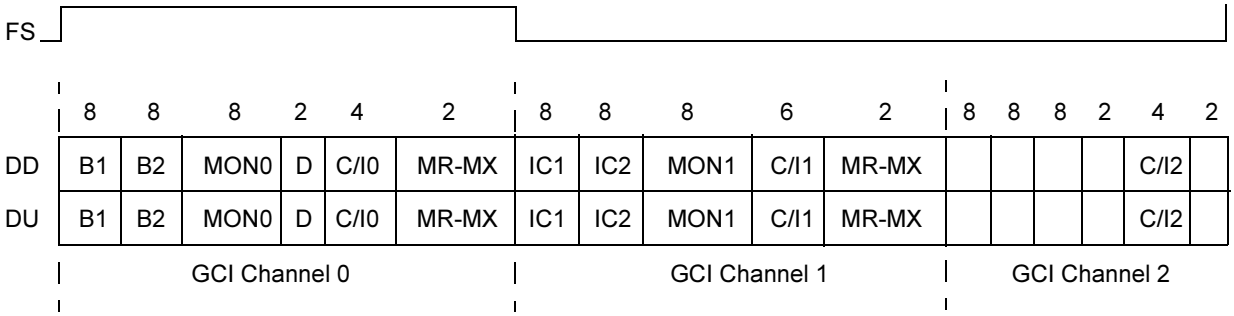


Figure 32. GCI/SCIT Frame Structure

B Channels. B1 and B2 are the first two 8-bit time slots after the frame sync pulse. Each B channel provides 64 Kbps of user data to/from the network.

Monitor Channels. There are two channels, monitor 0 and monitor 1. Each channel consists of eight bits of data and two associated handshake bits, MR and MX, that control data flow.

D Channel. The 16 Kbps D channel (2 bits per frame) provides a connection between the layer two and layer one components.

Command/Indicate Channels. Three command/indicate channels, C/I0, C/I1 and C/I2 are provided. Each sub-frame has one. (C/I2 is the same as TIC, as indicated below.) These channels provide real-time status between devices connected by means of the GCI/SCIT bus.

Intercommunication Channels. Two intercommunication subchannels are provided in GCI channel 1. These subchannels provide 64 Kbps data paths between user devices.

TIC Bus. The TIC bus is the same as C/I2 and is used for D channel access with some GCI/SCIT devices. It allows multiple layer-2 devices to individually gain access to the D and C/I channels located in the first sub-frame.

The data signals on the GCI/SCIT bus are called Data Upstream (DU) and Data Downstream (DD). Each of these is a bus that can be sensed as well as driven in an open-drain (open-collector) fashion by the Z382 and other devices. GCI practice defines certain fields on each line to flow in certain directions.

The Z382 always receives from DD and (when enabled) drives DU in the B2, MON0, D, C/I0, and MX0 fields. The Z382 always receives from DU and (when enabled) drives DD in the MR0 bit. Which line is driven and which is received is selected by software for the IC1, IC2, MON1, and C/I1 fields and the MX1 and MR1 bits, with MR1 always being in the opposite direction from MON1 and MX1.

Monitor Channel Operation

The monitor channels are full duplex and operate on a pseudo-asynchronous basis, in other words, data transfers take place synchronized to frame sync but the flow is controlled by a handshake procedure using the MX and MR bits. Figure 32 illustrates the handshake procedure (flow of events).

Idle. The MX and MR pair being held inactive (High) for two or more frames constitutes the channel being idle in that direction. The data received in the monitor channel is invalid and should be 11111111.

Start of Transmission. The first byte of data is placed on the bus and MX is activated (Low). MX remains active, and the data is repeated until an inactive-to-active transition of MR is received, indicating that the data has been captured by the receiver.

Subsequent Transmissions. The second and subsequent bytes are placed on the bus after the inactive to active transition of MR. At the time that the second byte is transmitted, MX is returned inactive for one frame time only; the data is valid in the same frame. In the following frame, MX returns active again and the same byte is transmitted. Data is repeated in subsequent frames and MX remains active until acknowledgment is detected (MR transition from inactive to active).

Maximum Speed Case. The transmitter is capable of minimizing the delay between bytes to achieve higher data throughput than is provided by the general case described previously. The first and second bytes are transmitted normally, However, starting with the third byte, the transmitter deactivates MX and transmits new data one frame time after MR is deactivated. In this way, the transmitter anticipates that MR is reactivated, which it accomplishes one frame time after it is deactivated, unless an abort is signalled by the receiver.

End of Message (EOM). The transmitter sends an EOM, normally after the last byte of data has been transmitted, by not reactivating MX after deactivating it in response to MR going inactive.

Reception. At the time the receiver detects the first byte, indicated by the inactive-to-active transition of MX, MR is inactive. In response to the activation of MX, the data is read off the bus and MR is activated. MR remains active until the next byte is received or an end of message is detected. Subsequent data is received from the bus on each falling edge of MX, and a monitor channel receive data available interrupt is generated. The data may be valid at the time that MX went inactive, one frame time prior to going active. MR is deactivated after the data is read and reactivated one frame time later. The transmitter detects MR going inactive and anticipates its reactivation one frame later. The reception of data is terminated by the reception of an end of message indication.

Abort. The abort is a signal from the receiver to the transmitter indicating that the data has been missed. It is not an abort in the classical sense, which is an indica-

tion that the current message should be ignored. The receiver indicates an abort by holding MR inactive for two or more frames in response to MX going inactive.

Flow Control. The receiver can hold off the transmitter by keeping MR active until the receiver is ready for the next byte. The transmitter does not start the next transmission cycle until MR goes inactive.

Monitor Channel Handling

Before transmitting data on a monitor channel, the processor looks at the Monitor 0 or 1 active status bit in GCI Status Register 2 to verify that the channel is inactive. The processor then writes the data to the Monitor Transmit Data Register. This condition enables the GCI hardware to proceed with the transmission of this data according to the monitor channel protocol. Upon receiving an acknowledge from the receiver, the transmit data request bit in GCI Status Register 1 is set, indicating that the monitor channel is ready to transmit another byte of data. When the last byte is acknowledged by the receiver, the processor sets the EOM request bit in the GCI Control Register and the monitor channel sends an end of message signal.

Upon receiving the monitor data, the receiver writes this data to the monitor receive register and sets the appropriate status bit. This action generates a Monitor Receive Data Available interrupt, instructing the processor to read this data.

Succeeding bytes of data are received in accordance to the monitor channel protocol and the processor is informed by means of the Monitor Receive Data Available interrupt. The processor forces the receiver to ask for an abort by setting the abort request bit. The receiver asks for an abort in transmission by sending an inactive MR for two consecutive frames. The abort transmission is indicated in the status bit by the transmitter.

Figure 33 illustrates Monitor Handshake Timing.

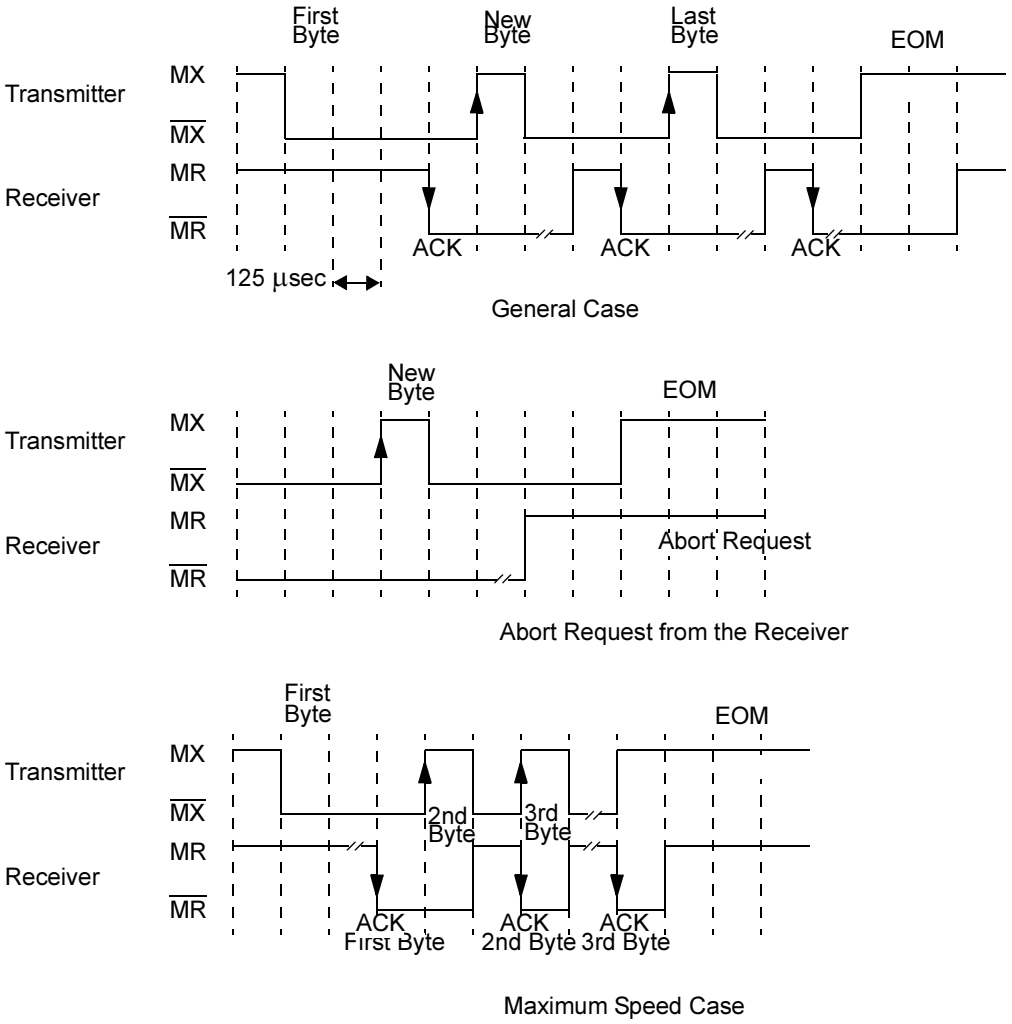


Figure 33. Monitor Handshake Timing Diagram

C/I Channel Operation

Data on C/I0 and C/I1 is transmitted continuously in each frame until new data is to be sent. A change in C/I channel data is considered valid if it has been received in two consecutive frames.

GCI/SCIT Bus Activation and Deactivation

Deactivation, Upstream to Downstream. The upstream (clock master) unit initiates deactivation by issuing a series of software handshakes by means of the C/I0 channel. The upstream unit waits for a deactivation indication from all down-

stream (clock slave) units. When this indication is received, a deactivation confirmation is issued, followed by stopping the clocks (forcing them Low) and placing the data pin in a high impedance state. After the clocks are stopped, the input pin is monitored for the presence of a timing request from the downstream unit (the pin being pulled Low).

Deactivation Request, Downstream to Upstream. Deactivation is normally initiated by the upstream device as described previously. When the downstream device receives the deactivation request over the C/I0 channel, it responds by sending the deactivation indication.

Activation Request, Downstream to Upstream. The downstream device can request that the clocks be started by pulling its data output line low. Once the clocks are started, the downstream unit requests activation by sending an activation request over the C/I0 channel.

Activation, Upstream to Downstream. The upstream unit activates the bus by starting the clocks and following the C/I0 channel-based activation procedure.

B1, B2, D, IC1, IC2 Channel Data. Rx data and the bit clock are supplied to the HDLC cells, and Tx data is taken from the HDLC cells. Each HDLC Transmitter and Receiver includes a Time Slot Assigner which can be programmed for any of the subchannels shown above.

Note: The HDLC Transmitters signal when they are sending data. These signals should not conflict with transmission by the GCI/SCIT module, but if they do, the HDLC modules have priority.

GCI/SCIT Registers

The GCI/SCIT interface includes the following I/O-mapped registers that can be read and written by the 380C processor:

- GCI Control Register
- Monitor 0 Transmit Data Register
- Monitor 0 Receive Data Register
- Monitor 1 Transmit Data Register
- Monitor 1 Receive Data Register
- C/I0–C/I2 Transmit Data Register
- C/I0–C/I2 Receive Data Register
- C/I1 Transmit Data Register
- C/I1 Receive Data Register
- GCI Status Register 1

- GCI Status Register 2
- GCI Interrupt Enable Register

GCI Control Register. This register controls the Monitor 1 and C/I1 Direction, the clock activation request to the master, enabling/disabling Monitors 1 and 0, and Monitors 1 and 0 EOM and Abort requests.

Monitor 0, Monitor 1, C/I0–C/I2 and C/I1 Transmit Data Registers. These registers transmit data on the respective channels in accordance with the GCI/SCIT protocol.

Monitor 0, Monitor 1, C/I0–C/I2 and C/I1 Receive Data Registers. These registers receive data from the respective channels in accordance with the GCI/SCIT protocol.

GCI Status Register 1. This register provides receive and transmit status conditions for Monitor 0 and 1 channels.

GCI Status Register 2. This register provides additional status conditions for the GCI/SCIT module.

GCI Interrupt Enable Register. This register provides control of interrupts from the various channels in the GCI/SCIT module.

Clocked Serial I/O (CSIO)

The Z382 includes a synchronous serial I/O port (CSIO) which provides half-duplex transmission/reception of fixed 8-bit data at a speed up to BUSCLK/20 bits/second. The CSIO is ideal for implementing a multiprocessor communication link between multiple Z80xxx family members. Figure 34 illustrates the CSIO block diagram.

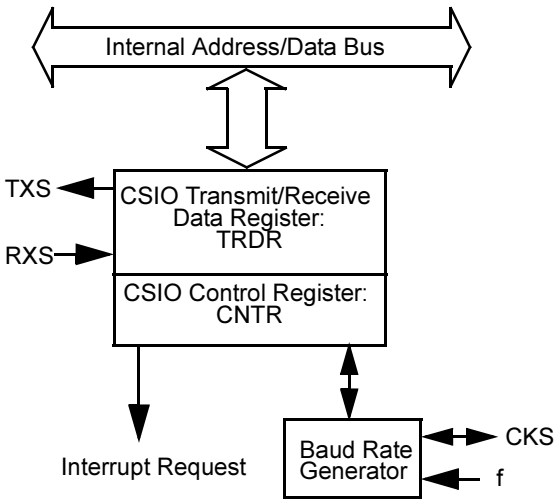


Figure 34. CSIO Block Diagram

Note: The three pins associated with the CSIO are multiplexed with other signals and must be configured for CSIO operation in order to use the CSIO as described in this section.

CSIO Registers

The CSIO channel includes the following I/O-mapped registers that can be read and written by the 380C processor:

- Transmit/Receive Data Register
- CSIO Control Register

CSIO Control Register. CNTR is used to monitor CSIO status, enable and disable the CSIO, enable and disable interrupt generation, and select the data clock speed and source.

CSIO Tx/Rx Data Register. TRDR is used for both CSIO transmission and reception in a half-duplex protocol. Thus, the system design must ensure that transmit and receive operations do not occur simultaneously. For example, if a CSIO transmission is attempted while the CSIO is receiving data, a CSIO does not work. Also, the TRDR is not buffered. Thus, attempting to perform a CSIO transmit while the previous transmit data is still being shifted out causes the shift data to be immediately updated, corrupting the transmit operation in progress. Similarly, reading TRDR while a transmit or receive is in progress must be avoided.

Counters, Timers and Other Miscellaneous Logic

Programmable Reload Timer

Figure 35 illustrates the Programmable Reload Timer block diagram.

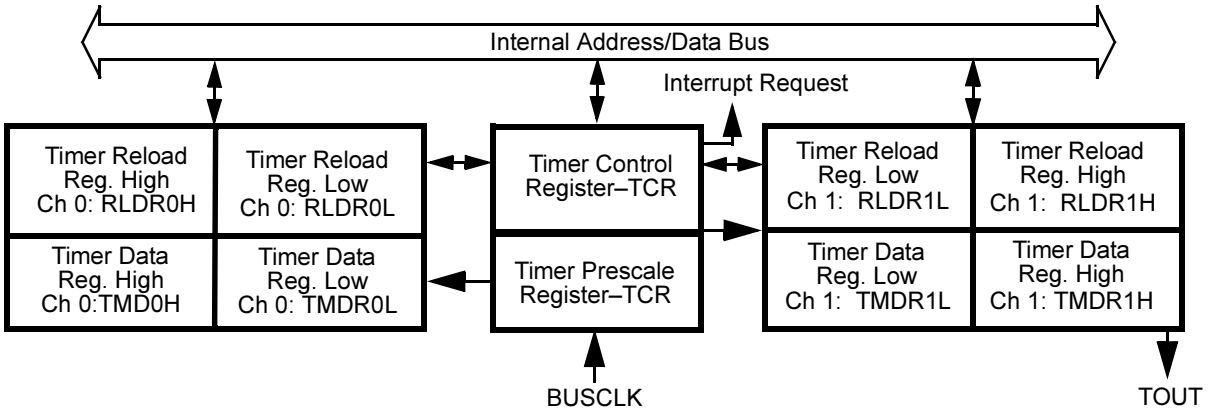


Figure 35. Programmable Reload Timer Block Diagram

The Z382 contains two separate 16-bit Programmable Reload Timers (PRT). Each PRT channel contains a 16-bit down counter and a 16-bit reload register. The down counter can be directly read and written and a down counter overflow interrupt can be enabled or disabled. Also, PRT1 can be programmed to set the T_{OUT} pin High or Low or to toggle it when the channel counts down to zero. PRT1 can perform programmable output waveform generation.

The two channels share a common status/control register and a Timer Prescale Register allowing the time base for each PRT to be programmed as the Z382 BUSCLK divided by a power of two.

PRT Common Registers

The PRTs share two I/O-mapped registers that can be read and written by the 380C processor:

- Timer Prescale Register
- Timer Control Register

Timer Prescale Register. Selects the rates at which each PRT is clocked, providing for BUSCLK divisors ranging from 1 to 32,768.

Timer Control Register. The TCR monitors the status of both PRT channels and controls enabling and disabling of down counting and interrupts. It also controls the effect of PRT1 on the T_{OUT} output pin.

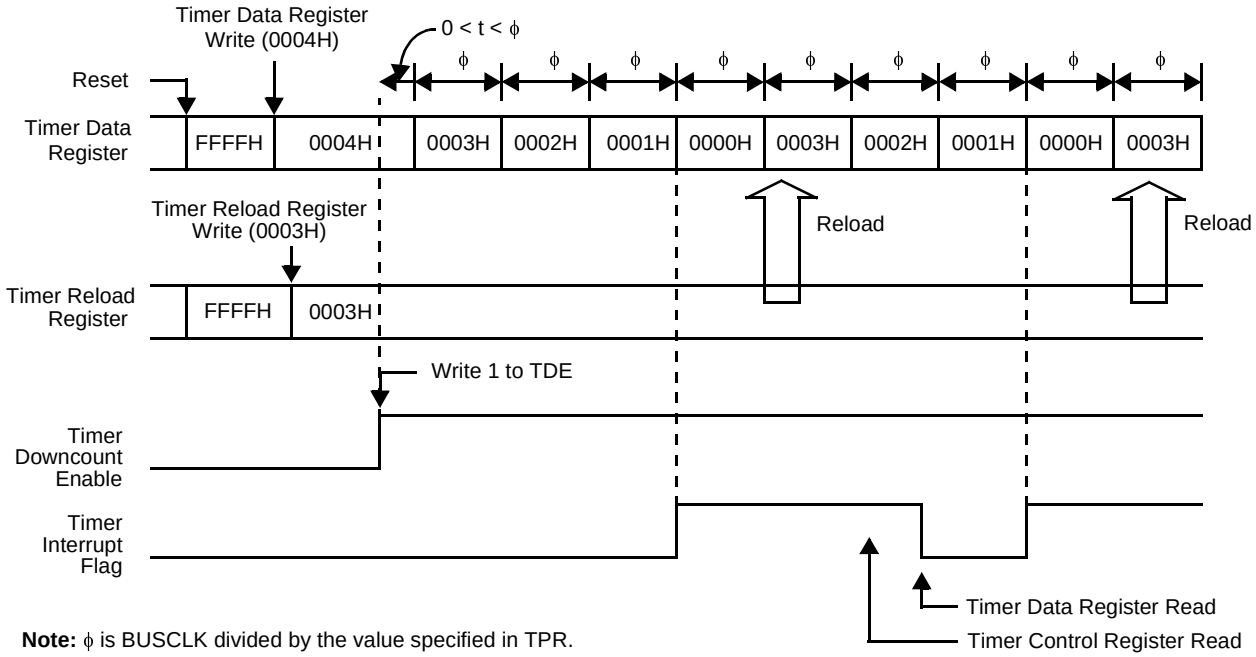
PRT Per-Channel Registers

The I/O-mapped per-channel registers in each PRT are:

- Timer Data Registers High/Low
- Timer Reload Registers High/Low

Timer Data Registers. Each PRT has a 16-bit Timer Data Register (TMDR). TMDR is decremented once every clock output from the timer prescaler, which divides the BUSCLK signal of the Z382 by a value which is specified, independently for PRT1 and PRT0, in the TPR. When TMDR counts down to 0, it is automatically reloaded with the value contained in its Timer Reload Register (RLDR).

Timer Reload Registers. Each PRT has a 16-bit Timer Reload Register (RLDR) When a PRT channel's TMDR counts down to 0, it is automatically reloaded with the contents of its RLDR. Figure 36 illustrates the operation of the PRT.



Note: ϕ is BUSCLK divided by the value specified in TPR.

Figure 36. PRT Operation Timing Diagram

Watch-Dog Timer

A Watch-Dog Timer (WDT) with programmable timeout intervals prevents code runaway and possible resulting system damage. The $\overline{\text{RESET}}$ input can be forced as an output upon the terminal count of the WDT, allowing external peripherals to be reset along with the Z382. Unlike other on-chip functions, the WDT is enabled

at Reset and must be disabled by software if its function is not desired. If software does not disable the WDT, it must periodically clear the WDT in order to avoid a hardware reset of the chip. The block diagram of the WDT is depicted in Figure 37.

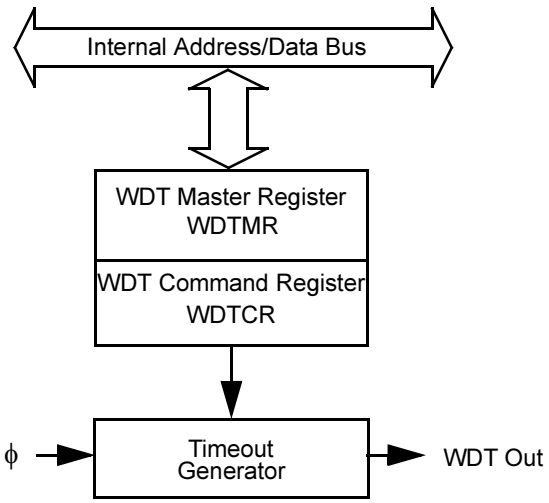


Figure 37. Watch-Dog Timer Block Diagram

WDT Registers

The CSIO channel includes the following I/O-mapped registers that can be read and written by the 380C processor:

- WDT Master Register
- WDT Command Register

Watch-Dog Timer Master Register. This register controls enabling/disabling of the WDT, its period, and whether the RESET pin is driven to reset external devices when the WDT times out.

WDT Command Register. The WDT decodes two values written to this register. One value is used to reset the WDT to a count of zero. The second value must be written to this register to disable the WDT.

Parallel Ports

The Z382 has four 8-bit bidirectional ports called ports A through D. A Direction Register associated with each port allows each bit of the port to be programmable as an input or an output.

Because of pin multiplexing, ports A and D are available only in non-Host applications, and the registers for Ports A and D are used by the MIMIC feature in Host

applications. Additional information on the multiplexing of the Port pins is provided in the Device Configuration section of this document.

Each port contains two registers accessible in the 380C's I/O address space:

- Port Direction Register
- Port Data Register

Bit 3 in the System Configuration Register controls whether only the lowest eight address lines are decoded, allowing the port data and direction registers to be accessed in any page of I/O space (as on the Z18x family), or whether access is limited to a single page (A[15-8] = 0).

Port Direction Register

The Direction Register determines which pins of the port are inputs and which are outputs.

In Host applications, the Port A and D Direction Registers are used to buffer data between the Host's HD7-0 lines and the Z382 for the Host DMA Mailbox and Host I/O Mailbox functions.

Port Data Register

When the 380C writes to the Data Register of an available port, the data is stored in this register. Any pins that are identified as output in the corresponding Port Direction Register are then driven with the new data. When the 380C reads the Data Register of an available port, the data on the external pins is returned.

In Host applications, the Port A and D Data Registers are used for implementation of the Host I/O Mailbox feature.

I/O Chip Selects

Two I/O chip selects, $\overline{\text{IOCS1}}$ and $\overline{\text{IOCS2}}$, are provided to support I/O access of external peripherals. These chip selects are asserted Low when some number of the 16 LSBs of the current 380C address match the values programmed in the IOCS registers. The number of bits actually compared is specified in one of the registers, providing I/O decode sizes ranging from 8 to 512 bytes.

Address comparisons take place during both memory and I/O cycles. The I/O Chip Selects are not asserted in INTACK cycles.

I/O Chip Select Registers

The following I/O-mapped registers are associated with the I/O chip selects and can be read and written by the 380C processor:

- I/OCS1 High and Low Address Registers
- I/OCS2 High and Low Address Registers

I/O Chip Select 1/2 High and Low Address Registers. Specify the base address and the I/O block size for I/O Chip Selects 1 and 2.

RAM and ROM Chip Selects

Three memory chip select outputs are provided: $\overline{\text{ROMCS}}$, $\overline{\text{RAMCSL}}$, and $\overline{\text{RAMCSH}}$. These outputs support both 8- and 16-bit memories, and are asserted for a selected address range (4 KB to 8 MB) during both memory and I/O cycles. Unlike Chip Select and $\overline{\text{MSIZE}}$ signalling, Wait State generation can be specified which occurs only during memory cycles.

For the selected ROM and/or RAM range, the $\overline{\text{MSIZE}}$ pin can be programmed to be forced Low in an open-drain fashion when the address is in the programmed range, forcing 8-bit accesses in one or both ranges. When $\overline{\text{MSIZE}}$ is forced for 8-bit RAM in this way, $\overline{\text{RAMCSL}}$ is asserted for all cycles in the selected address range, and the $\overline{\text{RAMCSH}}$ pin assumes its alternate use as port pin PC7. When $\overline{\text{MSIZE}}$ is not forced for 8-bit RAM, $\overline{\text{RAMCSL}}$ is qualified by $\overline{\text{BLEN}}$, and $\overline{\text{RAMCSH}}$ acts as a chip select output pin and is qualified by $\overline{\text{BHEN}}$.

RAM and ROM Chip Select Registers

The following I/O-mapped registers are associated with the RAM and ROM chip selects and can be read and written by the 380C processor:

- RAM Address High and Low Registers
- ROM Address High and Low Registers
- Memory Mode Register 1
- Memory Mode Register 2

RAM Address High and Low Registers. These registers specify which bits of the address bus are used in the address comparison and thus, the memory block size. The block size can range from 4 KB to 8 MB.

ROM Address High and Low Registers. These registers specify which bits of the address bus are used in the address comparison and thus, the memory block size. The block size can range from 4 KB to 8 MB.

Memory Mode Register 1. This register enables the ROM chip select, specifies the number of wait states for the ROM chip select, and specifies the number of T1 Wait states for the RAM chip select.

Memory Mode Register 2. This register enables the RAM chip select, specifies 8- or 16-bit memory accesses for the RAM and ROM chip selects independently, and specifies the number of T2 and T3 Wait states for the RAM chip select.

Interrupt Logic

The Z382's interrupt structure provides compatibility with the existing Z80 and Z180 with the following exception: the undefined Op Code trap's occurrence is with respect to the Z380 instruction set, and its response is improved (versus the Z180) to make trap handling easier. The Z380 offers additional features to enhance flexibility in system design.

Of the five external interrupt inputs provided, $\overline{\text{NMI}}$ is a nonmaskable interrupt. The remaining inputs, $\overline{\text{INT3-0}}$, are asynchronous maskable interrupt requests.

In an Interrupt Acknowledge transaction, address outputs A23-4 are driven to a logic High. One output among A3-0 is driven Low to indicate the maskable interrupt request being acknowledged. For example, when $\overline{\text{INT0}}$ is acknowledged, A3-1 are High and A0 is Low.

Interrupt modes 0 through 3 are supported for maskable interrupt request $\overline{\text{INT0}}$, which can be driven by external and on-chip sources. Modes 0, 1 and 2 have the same schemes as those in the Z80 and Z180. Mode 3 is similar to mode 2, except that 16-bit interrupt vectors are expected from the I/O devices. 8-bit and 16-bit I/O devices can be intermixed in this mode by having external pull up resistors at the data bus signals D15-8, for example.

The external maskable interrupt requests $\overline{\text{INT3-1}}$, as well as the less complex on-chip peripherals (PRTs, ASCIs, and CSIO) are handled in an assigned interrupt vectors mode. $\overline{\text{INT3-1}}$ can be used as Low or High active level-sensitive inputs, or as falling or rising edge-triggered inputs.

The Z382 can operate in either the NATIVE or EXTENDED mode. In NATIVE Mode, performing PUSH and POP instructions to the stack to save and retrieve interrupted PC values in interrupt handling use 16-bit sizes, and the stack pointer rolls over at the 64-KB boundary. In EXTENDED mode, the PC performs PUSH and POP instructions use 32-bit sizes, and the stack pointer rolls over at the 4-GB memory space boundary. The Z382 provides an Interrupt Register Extension, whose contents are always output as the address bus signals A23-16 when fetching the starting addresses of service routines from memory in interrupt modes 2 and 3 and the assigned vectors mode. In NATIVE Mode, such fetches are automatically done in 16-bit sizes and in EXTENDED Mode, in 32-bit sizes. These starting addresses should be even-aligned in memory locations. That is, their least significant bytes should have addresses with A0 = 0.

Interrupt Priority Ranking

The Z382 assigns a fixed priority ranking to handle its major categories of interrupt sources, as follows:

Table 36. Interrupt Source Priorities

Priority	Interrupt Sources
Highest	Trap (undefined Op Code)
	$\overline{\text{NMI}}$
	$\overline{\text{INT0}}$ (includes DMAs, MIMIC, HDLC controllers)
	$\overline{\text{INT1}}$
	$\overline{\text{INT2}}$
	GCI/SCIT
	PRT0
	PRT1
	CSIO
	ASCI0
	ASCI1
	Plug and Play ISA or PCMCIA
	I/O Mailbox
Lowest	$\overline{\text{INT3}}$

INT0 Peripherals

Those on-chip peripherals capable of generating their own interrupt vectors, including the MIMIC, DMAs, and HDLC controllers, have their interrupt requests logically OR'ed with the external $\overline{\text{INT0}}$ pin to produce the INT0 signal presented to the 380C processor. These interrupt sources are consecutive in the INT0 daisy-chain, but their relative priority can be programmed in the System Configuration Register. Their priority relative to external INT0 sources is controlled by how the Z382's IEI and IEO pins are connected.

Assigned Interrupt Vectors Mode ($\overline{\text{INT1-3}}$, PRTs, CSIO, ASCIs)

When the Z382 recognizes $\overline{\text{INT1-3}}$, or a request from an on-chip peripheral that cannot supply an interrupt vector (a PRT, CSIO, or ASCI), it generates an Interrupt Acknowledge transaction which is different from that for $\overline{\text{INT0}}$. This Interrupt Acknowledge transaction has $\overline{\text{IORQ}}$ active for external monitoring purposes, but

$\overline{M1}$, $\overline{IORD}$, and $\overline{IOWR}$ inactive so as not to stimulate external devices. The interrupted PC value is PUSHed onto the stack. IEF1 and IEF2 are cleared, disabling further maskable interrupt requests. The starting address of an interrupt service routine is fetched from a table entry and loaded into the PC to resume execution. The address of the table entry is composed of the I Extend contents as A31-16, the seven Vector Base bits of the Assigned Vectors Base Register as A15-9 and an assigned interrupt vector specific to the request being recognized as A8-0. The assigned vectors are as follows:

Table 37. Interrupt Sources and Assigned Vectors

Interrupt Source	Assigned Interrupt Vector
$\overline{INT1}$	00H
$\overline{INT2}$	04H
Reserved	08H
GCI/SCIT	0CH
PRT0	10H
PRT1	14H
CSIO	18H
ASCI0	1CH
ASCI1	20H
Plug-and-Play or PCMCIA	24H
I/O Mailbox	28H
$\overline{INT3}$	2CH

Trap Interrupt

The 380C generates a trap when an undefined Op Code is encountered. The trap is enabled immediately after reset, and it is not maskable. This feature can be used to increase software reliability or to implement extended instructions. An undefined Op Code can be fetched from the instruction stream, or it can be returned as a vector in an interrupt acknowledge transaction in interrupt mode 0.

Nonmaskable Interrupt

The nonmaskable interrupt input $\overline{NMI}$ is edge-sensitive, with the 380C internally latching the occurrence of its falling edge. When the latched version of $\overline{NMI}$ is recognized, the interrupted PC (Program Counter) value is pushed onto the stack, certain status flag manipulations are performed, and the 380C fetches and executes instructions from address 00000066H.

RETI Instruction

The original Z80 family I/O devices (PIO, SIO, CTC) are designed to monitor the Return from Interrupt Op Codes in the instruction stream, signifying the end of the current interrupt service routine. On the Z382, the M1 signal is active during all instruction fetch transactions. Because the Z382 may not execute an RETI that it fetches, and because it supports a 16-bit data bus, only half of which is visible to an 8-bit peripheral, the Z382 does not support RETI decoding by the PIO, SIO, and CTC.

Interrupt Registers

The following I/O-mapped registers are associated with interrupts and can be read and written by the 380C processor:

- Interrupt Enable Register
- Assigned Vectors Base Register
- INT3-1 Control Register
- Trap and Break Register

Interrupt Enable Register. This register provides the current status of the $\overline{\text{INT3-0}}$ pins and controls whether $\overline{\text{INT3}}$, $\overline{\text{INT2}}$, $\overline{\text{INT1}}$, and $\overline{\text{INT0}}$ are enabled or disabled. These flags are also affected by enable and disable interrupt instructions (DI (n) and EI (n)).

Assigned Vectors Base Register. The Interrupt Register Extension, Iz, together with the contents in bits 1-7 of this register, define the base address of the assigned interrupt vectors table in memory space.

INT3-1 Control Register. This register controls when and how the Z382 recognizes an interrupt on the corresponding pins (High or Low Level sensitive, Falling or Rising Edge Triggered) and provides the means for clearing edge triggered interrupt requests if such are specified for $\overline{\text{INT3-1}}$.

Trap and Break Register. Two bits of this register provide status on traps. One bit is set if an undefined opcode is fetched in the instruction stream. A second bit is set if an undefined opcode is returned as a vector in an interrupt acknowledge transaction in mode 0.

Z380-Compatible Peripheral Functions

The Z382 incorporates a number of Z80380 compatible functions. The Z382's I/O bus can be programmed to run at a slower rate than its memory bus. In addition, a heartbeat transaction can be generated on the I/O bus that emulates a Z80 instruction fetch cycle. Such cycles are needed for a particular Z80 family I/O

device to perform its interrupt functions. Finally, a DRAM refresh function is incorporated, with programmable refresh transaction burst size.

I/O Bus Control

The Z382 is designed to interface easily with external I/O devices that can be of the Z80 product family by supplying four I/O bus control signals: M1, IORQ, IORD, and IOWR. In addition, the Z382 supplies an IOCLK that is a divided down version of its BUSCLK. Programmable wait states can be inserted in the various I/O transactions.

DRAM Refresh

The Z382 is capable of providing refresh transactions to dynamic memories that have internal refresh address counters. A user can select how often refresh requests should be made to the Z80's External Interface Logic, as well as the burst size (number of refresh transactions) for each request iteration. The External Interface Logic grants these requests by performing refresh transactions with CAS-before-RAS timing on the TREFR, TREFA and TREFC bus control signals. In these transactions, BHEN, BLEN and the user specified chip select signal(s) are driven active to facilitate refreshing all the DRAM modules at the same time. A user can also specify the T1, T2 and T3 waits to be inserted.

Note: The Z382 cannot provide refresh transactions when it relinquishes the system bus, with its BREQ input active. In that situation, the number of missed refresh requests are accumulated in a counter, and when the Z382 regains the system bus, the missed refresh transactions are performed.

Low Power STANDBY Mode

The Z382 provides an optional STANDBY mode to minimize power consumption during system idle time. If this option is enabled, executing the Sleep instruction stops the Z382's oscillator if it is in use, and in any case stops clocking internal to the Z382 (except to PRT0 if it is enabled) and at the BUSCLK and IOCLK outputs. The STNBY and HALT signals go Low to indicate that the Z382 is entering the standby mode. All Z382 operations are suspended, the bus control signals are driven inactive and the address bus is driven High. STANDBY mode can be exited by asserting any of the RESET, NMI, INT3-INT0 (if enabled), or optionally, BREQ inputs.

If STANDBY mode is not enabled, the Sleep instruction does not stop the Z382's oscillator if it is in use, but blocks clocking from internal modules, except PRT0 if it is enabled. In this case, STNBY (but not HALT) goes Low to indicate the Z382's status.

Peripheral Function Control Registers

The functions described previously are controlled by a number of I/O mapped on-chip registers:

- Clock Control Register
- I/O Waits Register
- Refresh Registers 0, 1 and 2
- Refresh Wait Register
- Standby Mode Control Register

Clock Control Register. This register controls how BUSCLK is derived from the input clock (CLKI, CLKI/2 or CLKI x 2), provides a means of disabling CLKO to save power and reduce noise if an external clock is used, and controls the I/O Clock Rate (BUSCLK/8 to BUSCLK).

I/O Waits Register. This register allows for up to seven wait states to be inserted in external I/O read and write transactions, and at the latter portions of interrupt transactions to capture interrupt vectors. Also allows for up to seven wait states to be inserted at the early portions of interrupt acknowledge transactions, for the interrupt daisy chain through on-chip and possibly external I/O devices to settle.

Refresh Register 0. This register defines the interval between refresh requests to the Z382's External Interface Logic.

Refresh Register 1. This register provides the Missed Requests Count. This count increments by one when a refresh request is made and decrements by one when the Z382's External Interface Logic completes each burst of refresh transactions. A user can read the count status, and if necessary, take corrective actions such as adjusting the burst size.

Refresh Register 2. This register enables the refresh function and defines the number of refresh transactions per refresh request made to the Z382's External Interface Logic.

Refresh Wait Register. This register defines the number of T1, T2 and T3 wait states to be inserted in refresh transactions.

STANDBY Mode Control Register. This register enables the Z382 to go into low-power STANDBY mode when the Sleep instruction is executed, allows asserting BREQ to exit the mode, and specifies the approximate running duration of a warm-up counter that provides a delay before the Z382 resumes its clocking and operations, from the time an interrupt or bus request (if so enabled) is asserted to exit STANDBY mode.

Device Configuration

In addition to the configuration options provided in the registers associated with each of the major functional blocks in the Z382, there are two registers which control the overall device configuration:

- System Configuration Register
- Pin Multiplexing Register

System Configuration Register

The System Configuration Register controls the major modes of the Z382:

- How pins 60–92 are used:
  - Connected to the ISA bus of a host PC
  - Connected to the PCMCIA bus of a host PC
  - Used for the ASCIs, CSIO, and ports A and D, except that the full-time outputs among these signals (TXA0, TXA1, RTS0, TxS) are disabled
  - Used for the ASCIs, CSIO, and ports A and D, including the TXA0, TXA1, RTS0, and TxS outputs
- How pins 110–112 are used:
  - $\overline{\text{DCD0}}$, $\overline{\text{CTS0}}$, $\overline{\text{CTS1}}$ ASCI control signals
  - $\overline{\text{TREFA}}$, $\overline{\text{TREFC}}$, and $\overline{\text{TREFR}}$ DRAM control signals
- D15–0 use during reads from on-chip I/O devices:
  - The D15-0 pins are driven as outputs from the Z382
  - The pins are left 3-stated to reduce power consumption, noise, and EMI/RFI to some extent
- I/O address decoding of the MIMIC and Parallel Ports
  - A15-8 must be zero to access these features
  - The address decoding for these ports disregards address lines above A7, so that these devices are replicated in each 256-byte page of I/O space as on the Z80.
- The relative interrupt priority of the MIMIC, HDLC channels, and DMA channels on the INT0 daisy chain.

In addition to these controls, certain pins are multiplexed automatically based on the state of register bits in their associated functions.

Pin Multiplexing Register

The Pin Multiplexing Register controls smaller-scale pin multiplexing issues than those handled in the System Configuration Register.

- Whether the pins normally used for HDLC 0 are used for ASCI0 signals instead.
- Whether the pins normally used for HDLC 1 are used for ASCI1 signals instead.
- The functions of pins 47, 48, 49, 53, 56, 57, 58 and 109.

Programmable Low-Noise Drivers

To help reduce noise generated by the Z382 output switching, selected outputs can be placed in a reduced drive configuration. When a pin is placed in LOW NOISE mode, its drive is reduced to 1/3 of its normal output drive current. This decreases the slew rate of the driver, which reduces current spikes induced onto the power bussing of the Z382.

The Output Drive Control Register provides this function for a number of groups of Z382 output or I/O pins.

Z382 I/O Register Maps

Table 38. Z80380-Compatible Registers

Register Name	Z382 Address	Z380 Address	Access
Assigned Vectors Base Register	0018H	0018H	R/W
Trap and Break Register	0019H	0019H	R/W
I/O Waits Register	001EH	000EH	R/W
Refresh Waits Register	001FH	000FH	R/W
Clock Control Register	0021H	0011H	R/W
Refresh Register 0	0023H	0013H	R/W
Refresh Register 1	0024H	0014H	R/W
Refresh Register 2	0025H	0015H	R/W
Standby Mode Control Register	0026H	0016H	R/W
Interrupt Enable Register	0027H	0017H	R/W
Chip Version ID Register	0020H	00FFH	RO

Table 39. Z80382 ASCI, PRT, CSIO, WDT Registers

Register Name	I/O Address	Access
ASCI Control Register A Ch 0	0000H	R/W
ASCI Control Register A Ch 1	0001H	R/W
ASCI Control Register B Ch 0	0002H	R/W
ASCI Control Register B Ch 1	0003H	R/W
ASCI Status Register Ch 0	0004H	R/W
ASCI Status Register Ch 1	0005H	R/W
ASCI TX Data Register Ch 0	0006H	R/W
ASCI TX Data Register Ch 1	0007H	R/W
ASCI RX Data Register Ch 0	0008H	R/W
ASCI RX Data Register Ch 1	0009H	R/W
CSIO Control Register	000AH	R/W
CSIO Tx/Rx Data Register	000BH	R/W
Timer Data Register Ch OL	000CH	R/W
Timer Data Register Ch OH	000DH	R/W
Reload Register Ch OL	000EH	R/W
Reload Register Ch OH	000FH	R/W
Timer Control Register	0010H	R/W
Timer Prescale Register	0011H	R/W
ASCI0 Extension Control Register	0012H	R/W
ASCI1 Extension Control Register	0013H	R/W
Timer Data Register Ch 1L	0014H	R/W
Timer Data Register Ch 1H	0015H	R/W
Reload Register Ch 1L	0016H	R/W
Reload Register Ch 1H	0017H	R/W
ASCI0 Time Constant Low	001AH	R/W
ASCI0 Time Constant High	001BH	R/W
ASCI1 Time Constant Low	001CH	R/W

Table 39. Z80382 ASCI, PRT, CSIO, WDT Registers (Continued)

Register Name	I/O Address	Access
ASCI1 Time Constant High	001DH	R/W
WDT Master Register	0028H	R/W
WDT Command Register	0029H	WO

Table 40. Port and New Z80382 Registers

Register Name	I/O Address	Access
IOCS1 Low Register	002AH	R/W
IOCS1 High Register	002BH	R/W
IOCS2 Low Register	002CH	R/W
IOCS2 High Register	002DH	R/W
RAM Low Register	002EH	R/W
RAM High Register	002FH	R/W
ROM Low Register	0030H	R/W
ROM High Register	0031H	R/W
Memory Mode Register 1	0032H	R/W
Memory Mode Register 2	00D3H	R/W
System Configuration Register	0036H	R/W
Pin Multiplexing Register	0037H	R/W
ASCI0 DMA Control Register	0038H	R/W
ASCI1 DMA Control Register	0039H	R/W
Output Drive Control Register	003AH	R/W
INT3-1 Control Register	003BH	R/W
Port A Data Register	00EEH	R/W
Port A Data Direction Register	00EDH	R/W
Port B Data Register	00E5H	R/W
Port B Data Direction Register	00E4H	R/W
Port C Data Register	00DEH	R/W

Table 40. Port and New Z80382 Registers (Continued)

Register Name	I/O Address	Access
Port C Data Direction Register	00DDH	R/W
Port D Data Register	00E8H	R/W
Port D Data Direction Register	00E7H	R/W

Table 41. DMA Registers

Register Name	I/O Address	Access
DMA Control Register	003EH	R/W
DMA Vector Register	003FH	R/W
DMA0 List Address Register Low*	0040H	R/W
DMA0 List Address Register Middle*	0041H	R/W
DMA0 List Address Register High*	0042H	R/W
DMA0 Control/Status Register	0043H	R/W
DMA1 List Address Register Low*	0044H	R/W
DMA1 List Address Register Middle*	0045H	R/W
DMA1 List Address Register High*	0046H	R/W
DMA1 Control/Status Register	0047H	R/W
DMA2 List Address Register Low*	0048H	R/W
DMA2 List Address Register Middle*	0049H	R/W
DMA2 List Address Register High*	004AH	R/W
DMA2 Control/Status Register	004BH	R/W
DMA3 List Address Register Low*	004CH	R/W
DMA3 List Address Register Middle*	004DH	R/W
DMA3 List Address Register High*	004EH	R/W
DMA3 Control/Status Register	004FH	R/W
DMA4 List Address Register Low*	0050H	R/W
DMA4 List Address Register Middle*	0051H	R/W
DMA4 List Address Register High*	0052H	R/W
DMA4 Control/Status Register	0053H	R/W
DMA5 List Address Register Low*	0054H	R/W

Table 41. DMA Registers (Continued)

Register Name	I/O Address	Access
DMA5 List Address Register Middle*	0055H	R/W
DMA5 List Address Register High*	0056H	R/W
DMA5 Control/Status Register	0057H	R/W
DMA6 List Address Register Low*	0058H	R/W
DMA6 List Address Register Middle*	0059H	R/W
DMA6 List Address Register High*	005AH	R/W
DMA6 Control/Status Register	005BH	R/W
DMA7 List Address Register Low*	005CH	R/W
DMA7 List Address Register Middle*	005DH	R/W
DMA7 List Address Register High*	005EH	R/W
DMA7 Control/Status Register	005FH	R/W
Note: * These addresses can be selected to access the Buffer Address and Buffer Length register for testing.		

Table 42. HDLC Registers

Register Name	I/O Address	Access
HDLC Vector Register	003DH	R/W
HDLC0 Transmit Mode Register	0060H	R/W
HDLC0 Transmit Interrupt Register	0061H	R/W
HDLC0 Transmit Control/Status	0062H	R/W
HDLC0 Transmit Fill Register	0063H	R/W
HDLC0 Receive Mode Register	0064H	R/W
HDLC0 Receive Interrupt Register	0065H	R/W
HDLC0 Counter Access Port	0066H	R/W
HDLC0 DMA Select Register	0067H	R/W
HDLC1 Transmit Mode Register	0068H	R/W
HDLC1 Transmit Interrupt Register	0069H	R/W
HDLC1 Transmit Control/Status Register	006AH	R/W
HDLC1 Transmit Fill Register	006BH	R/W
HDLC1 Receive Mode Register	006CH	R/W

Table 42. HDLC Registers (Continued)

Register Name	I/O Address	Access
HDLC1 Receive Interrupt Register	006DH	R/W
HDLC1 Counter Access Port	006EH	R/W
HDLC1 DMA Select Register	006FH	R/W
HDLC2 Transmit Mode Register	0070H	R/W
HDLC2 Transmit Interrupt Register	0071H	R/W
HDLC2 Transmit Control/Status Register	0072H	R/W
HDLC2 Transmit Fill Register	0073H	R/W
HDLC2 Receive Mode Register	0074H	R/W
HDLC2 Receive Interrupt Register	0075H	R/W
HDLC2 Counter Access Port	0076H	R/W
HDLC2 DMA Select Register	0077H	R/W

Table 43. GCI/SCIT Registers

Register Name	I/O Address	Access
GCI Control Register	00C0H	R/W
GCI Status Register 1	00C1H	RO
GCI Status Register 2	00C2H	R/W
GCI Interrupt Enable Register	00C3H	R/W
MON0 Transmit Data Register	00C4H	WO
MON0 Receive Data Register	00C4H	RO
MON1 Transmit Data Register	00C5H	WO
MON1 Receive Data Register	00C5H	RO
C/I0 Transmit Data Register	00C6H	WO
C/I0 Receive Data Register	00C6H	RO
C/I1 Transmit Data Register	00C7H	WO
C/I1 Receive Data Register	00C7H	RO

Table 44. Z80382 MIMIC Registers

Register Name	I/O Address	Access	Host
MMC MIMIC Master Control Register	00FFH	R/W	None
IUS/IP Interrupt Pending	00FEH	R/Wb7	None
IE Interrupt Enable	00FDH	R/W	None
IVEC Interrupt Vector	00FCH	R/W	None
RTCR Receive Time Constant	00FBH	R/W	None
TTCR Transmit Time Constant	00FAH	R/W	None
DLM Divisor Latch (MSByte)	00F9H	RO	01H, DLAB=1, R/W
DLL Divisor Latch (LSByte)	00F8H	RO	00H, DLAB=1, R/W
SCR Scratch Register	00F7H	RO	07H, R/W
MSR Modem Status Register	00F6H	R/Wb7-4	0H6, RO
LSR Line Status Register	00F5H	R/Wb6432	0H5, RO
MCR Modem Control Register	00F4H	RO	04H, R/W
LCR Line Control Register	00F3H	RO	03H, R/W
IER Interrupt Enable Register	00F1H	RO	01H, DLAB=0, R/W
RBR Receiver Buffer Register	00F0H	WO	00H, DLAB=0, RO
THR Transmitter Holding Register	00F0H	RO	00H, DLAB=0, WO
MIMIC DMA Control Register	00EFH	R/W	None
FSCR FIFO Status and Control Register	00ECH	R/W7-4	None
TTTC Transmitter Timeout Time Constant Register	00EBH	R/W	None
RTTC Receiver Timeout Time Constant Register	00EAH	R/W	None
IIR Interrupt Identification Register	None	None	02H, RO
FCR FIFO Control Register	00E9H	RO	02H, RO
MIMIC Modification Register	00E9H	WO	None
Host DMA Control Register	00E6H	R/W	None
MIMIC BRG High Constant Register	00E1H	R/W	
MIMIC BRG Low Constant Register	00E0H	R/W	
IOBRG Register	00D6H	R/W	
Host I/O Status Register	00D5H	W bit1/R	Base + 10b R

Table 44. Z80382 MIMIC Registers (Continued)

Register Name	I/O Address	Access	Host
Host DMA Mailbox Control Register	00D2H	R/W	None
Host DMA Transmit Register 1	00D1H	RO	$\overline{\text{HDAK1}}$, $\overline{\text{HWR}}$ lo
Host DMA Receive Register 1	00D1H	WO	$\overline{\text{HDAK1}}$, $\overline{\text{HRD}}$ lo
Host DMA Transmit Register 0	00D0H	RO	$\overline{\text{HDAK0}}$, $\overline{\text{HWR}}$ lo
Host DMA Receive Register 0	00D0H	WO	$\overline{\text{HDAK0}}$, $\overline{\text{HRD}}$ lo

Table 45. PCNMCIA Memory and Registers

Register Name	I/O Address	Access	Host
Low Attribute Memory	0100H-177H	R/W	Attr 00-EEH even
Base Address Registers 0-6	017H8-17EH	R/W	Attr F0-FCH even
Z80 Control Register	017FH	R/W	None
Configuration Option Register	0180H	R/W	Attr 100H
Configuration Status Register	0181H	R/W	Attr 102H
Pin Replacement Register	0182H	R/W	Attr 104H
Socket Copy Register	0183H	R/W	Attr 106H
Extended Status Register	0184H	R/W	Attr 108H
Image Base Address Registers	0185,6H	R/W	Attr 10A, CH
Version Number Register	0187H	R/W	Attr 10EH
High Attribute Memory	0188H-1FFH	R/W	Attr 110-1FEH even

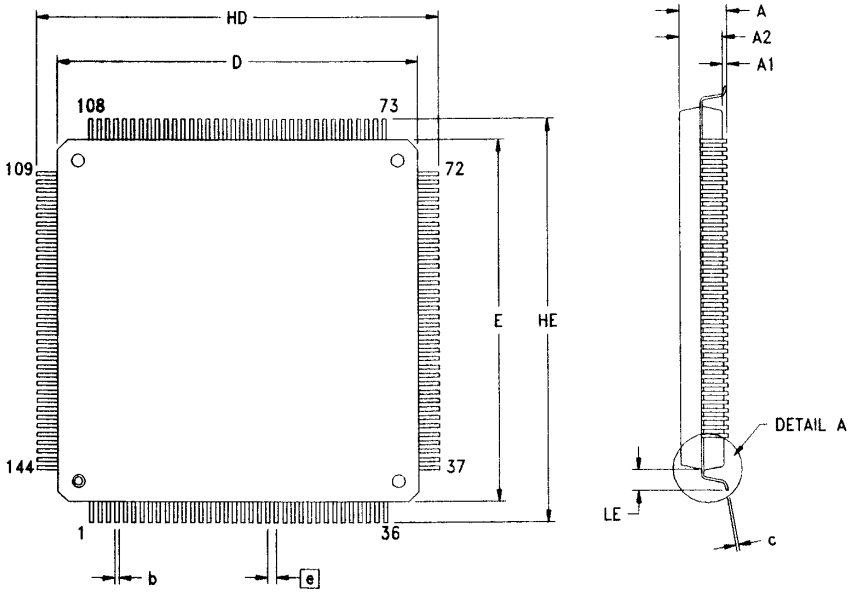
Table 46. Plug-and-Play ISA Registers

Register Name	I/O Address	Access	Host
PnP Address Register	None		I/O 0279H, WO
PnP Write Data Register	None		I/O 0A79H, WO
PnP Read Data Register	None		I/O 0203-3FFH, RO
Read Address Register	None		PnP 00H, WO
Isolation Register	None		PnP 01H, RO
Configuration Control Register	None		PnP 02H, WO

Table 46. Plug-and-Play ISA Registers (Continued)

Register Name	I/O Address	Access	Host
Wake Register	None		PnP 03H, WO
PnP Master Register	0102H	R/W	None
Resource Data Register	0104H	WO	PnP 04H, RO
PnP Status Register	0105H	RO	PnP 05H, RO
Card Select Number (CSN) Register	0106H	RO	PnP 06H, R/W
Logical Device Number Register	None		PnP 07H, RO
Activate Register	0130H	R/W	PnP 30H, R/W
I/O Range Check Register	None		PnP 31H, R/W
I/O Mailbox Base Address Registers	0160,1H	R/W	PnP 60,1H, R/W
MIMIC Base Address Registers	0162,3H	R/W	PnP 62,3H, R/W
Interrupt Request Level Register	0170H	R/W	PnP 70H, R/W
DMA Channel 0,1 Registers	0174,5H	R/W	PnP 74,5H, R/W

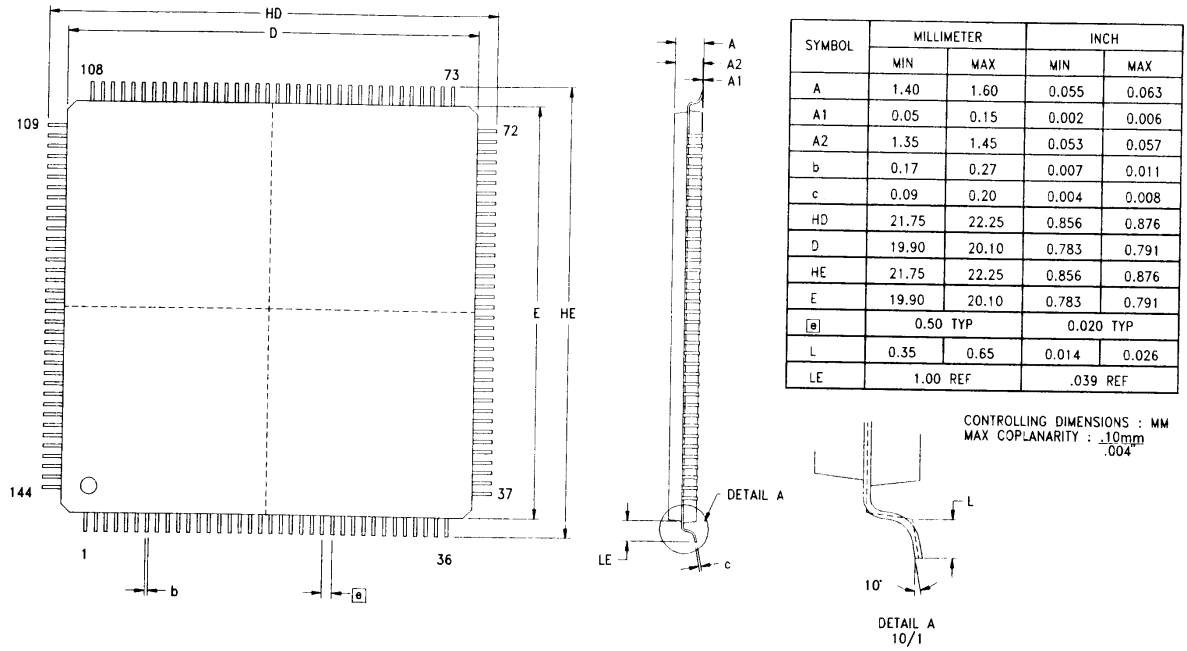
Package Information



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	3.42	4.07	.135	.160
A1	0.25	–	.010	–
A2	3.17	3.67	.125	.144
b	0.22	0.38	.009	.015
c	0.13	0.23	.004	.009
HD	30.95	31.45	1.219	1.238
D	27.90	28.10	1.098	1.106
HE	30.95	31.45	1.219	1.238
E	27.90	28.10	1.098	1.106
e	0.65 TYP		.026 TYP	
L	0.65	0.95	.026	.037
LE	1.60 REF		.063 REF	

CONTROLLING DIMENSIONS : MM
MAX COPLANARITY : .10mm (.004")

Figure 38. 144-Lead Plastic QFP Package Diagram



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	1.40	1.60	0.055	0.063
A1	0.05	0.15	0.002	0.006
A2	1.35	1.45	0.053	0.057
b	0.17	0.27	0.007	0.011
c	0.09	0.20	0.004	0.008
HD	21.75	22.25	0.856	0.876
D	19.90	20.10	0.783	0.791
HE	21.75	22.25	0.856	0.876
E	19.90	20.10	0.783	0.791
e	0.50 TYP		0.020 TYP	
L	0.35	0.65	0.014	0.026
LE	1.00 REF		.039 REF	

Figure 39. 144-Lead Plastic Low Profile VQFP Package Diagram

Ordering Information

Part Number	V_{DD}	Clock Speed(1)	Package Type	Package Code	Oper. Temp.
Z8038220ASC	5V ±10%	20 MHz	VQFP	MKT71C1173-00	0 to +70°C
Z8038220FSC	5V ±10%	20 MHz	QFP	MKT71C1163-00	0 to +70°C
Z8L38210ASC	3.3V ±10%	10 MHz	VQFP	MKT71C1173-00	0 to +70°C
Z8L38210FSC	3.3V ±10%	10 MHz	QFP	MKT71C1163-00	0 to +70°C

Note: 1. Refers to maximum internal bus clock frequency. See AC specifications for maximum external clock speed.

Package

F = QFP (Plastic Quad Flatpack)

A = VQFP (Very Small QFP)

Temperature

S = 0 to +70°C

Speeds

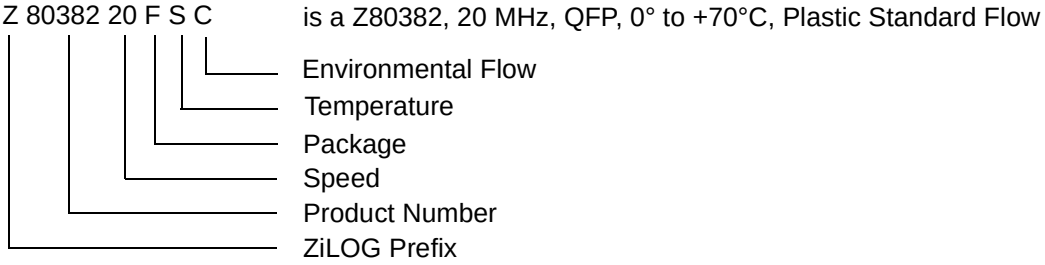
10 = 10 MHz

20 = 20 MHz

Environmental

C = Plastic Standard

Example:



Device Errata

The Z382 die revision that is currently shipping in production is the BB revision. This part contains the following errata:

1. The maximum clock speed for revision BB is 20 MHz (1x clock).
2. HAEN Low in address decoding is not included for the MIMIC and I/O Mailbox.
3. In the HDLC transmitter, Tx DMA Request when Tx FIFO is half empty: seems to work like “...when one slot available in Tx FIFO”. This condition is not a fatal error.
4. Bit seven of the System Configuration register was not taken into account in enabling pins 110, 111 and 112. Fixes for this situation include setting bit 5 to 1, or driving those lines to GND. When bit 5 is 0, these pins form an OR function with any other CTS, DCD pin which is enabled. Grounding pins 110, 111 and 112 and setting bit 5 of the System Configuration register to 0 allows the other enabled pin(s) to control the input(s).
5. There is a report that software cannot restart a DMA channel that has quit because it has reached the end of a list by setting the Run bit. This situation can be resolved by restarting the channel, loading the LAR.

Index

D

E

F

G

H

N

O

P

Q

R

S

T

U

V

W

Z

OCEAN CHIPS
Океан Электроники
Поставка электронных компонентов

Компания «Океан Электроники» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Поставка оригинальных импортных электронных компонентов напрямую с производств Америки, Европы и Азии, а так же с крупнейших складов мира;
- Широкая линейка поставок активных и пассивных импортных электронных компонентов (более 30 млн. наименований);
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Помощь Конструкторского Отдела и консультации квалифицированных инженеров;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Поставка электронных компонентов под контролем ВП;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- При необходимости вся продукция военного и аэрокосмического назначения проходит испытания и сертификацию в лаборатории (по согласованию с заказчиком);
- Поставка специализированных компонентов военного и аэрокосмического уровня качества (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Actel, Aeroflex, Peregrine, VPT, Syfer, Eurofarad, Texas Instruments, MS Kennedy, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Компания «Океан Электроники» является официальным дистрибьютором и эксклюзивным представителем в России одного из крупнейших производителей разъемов военного и аэрокосмического назначения «**JONHON**», а так же официальным дистрибьютором и эксклюзивным представителем в России производителя высокотехнологичных и надежных решений для передачи СВЧ сигналов «**FORSTAR**».

AVIC JONHON

«JONHON» (основан в 1970 г.)

Разъемы специального, военного и аэрокосмического назначения:

(Применяются в военной, авиационной, аэрокосмической, морской, железнодорожной, горно- и нефтедобывающей отраслях промышленности)

«FORSTAR» (основан в 1998 г.)

ВЧ соединители, коаксиальные кабели, кабельные сборки и микроволновые компоненты:

(Применяются в телекоммуникациях гражданского и специального назначения, в средствах связи, РЛС, а так же военной, авиационной и аэрокосмической отраслях промышленности).

FORSTAR®

Телефон: 8 (812) 309-75-97 (многоканальный)
Факс: 8 (812) 320-03-32
Электронная почта: ocean@oceanchips.ru
Web: http://oceanchips.ru/
Адрес: 198099, г. Санкт-Петербург, ул. Калинина, д. 2, корп. 4, лит. А