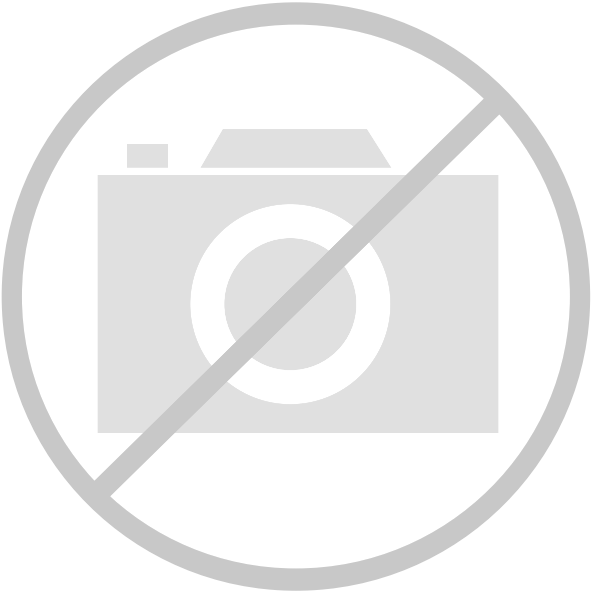
Zilog - Z8L38210FSG Datasheet



Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Obsolete
Core Processor	Z380C
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	10MHz
Co-Processors/DSP	-
RAM Controllers	-
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	-
SATA	-
USB	-
Voltage - I/O	3.3V
Operating Temperature	0°C ~ 70°C (TA)
Security Features	-
Package / Case	144-QFP
Supplier Device Package	144-QFP (28x28)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8l38210fsg

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

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 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{TXEN}}$1-0/$\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

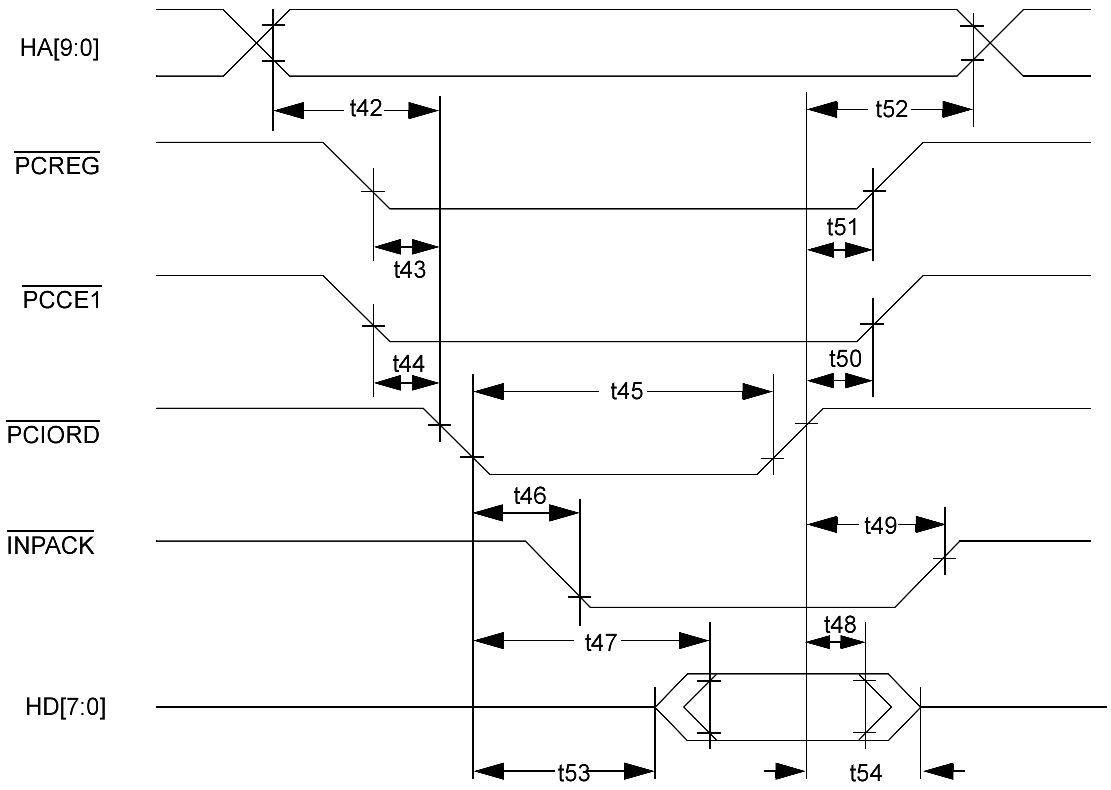


Figure 7. Host - PCMCIA I/O Read Timing Diagram

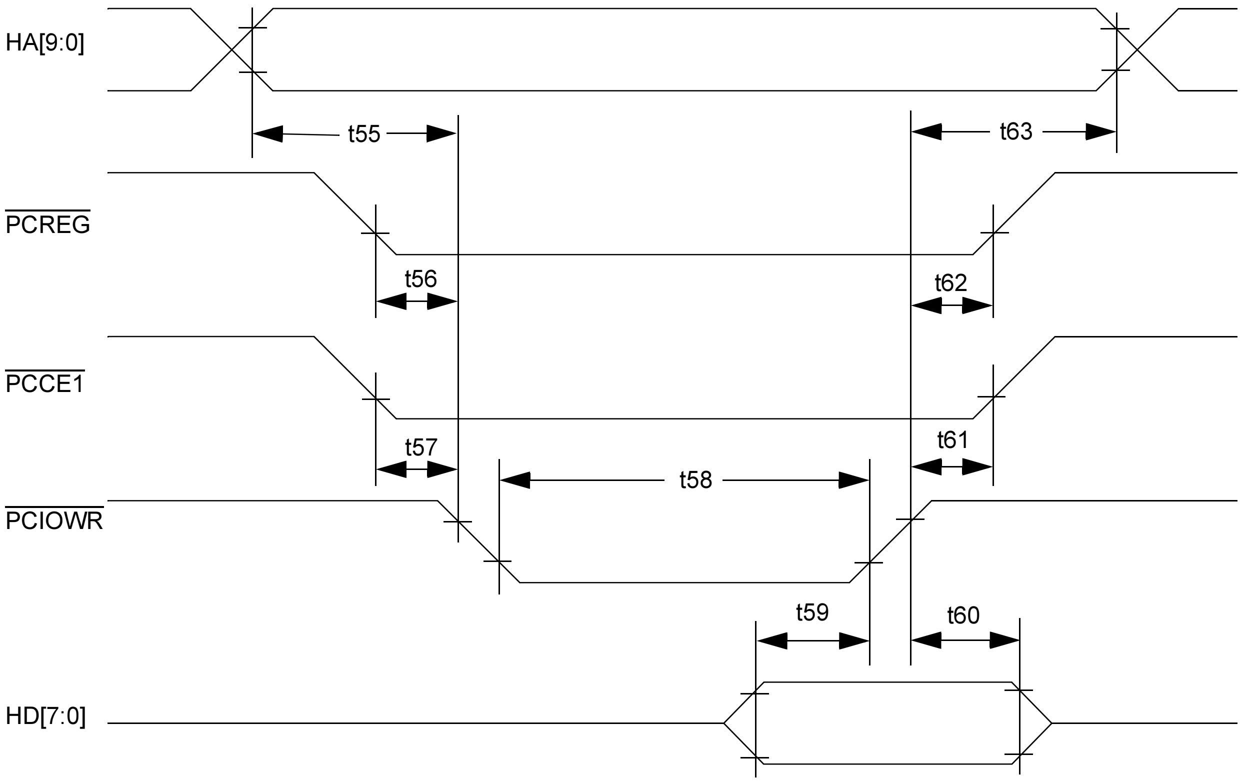

HA[9:0]
t55
t63
$\overline{\text{PCREG}}$
t56
t62
$\overline{\text{PCCE1}}$
t57
t61
t58
$\overline{\text{PCIOWR}}$
t59
t60
HD[7:0]

Figure 8. Host - PCMCIA I/O Write 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.

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

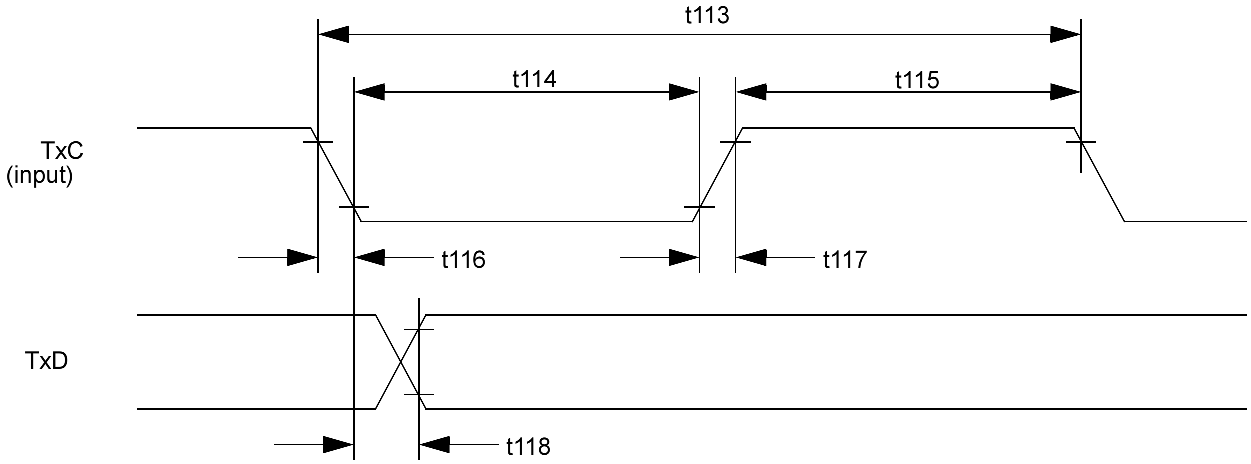


Figure 19. HDLC Transmit Timing Diagram (Full Time HDLC)

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

Pin Name	Pin Number(s)	Description
$\overline{\text{M1}}$	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).
$\overline{\text{MRD}}$	126	**Memory Read** (output, active Low, 3-state): This signal indicates that the addressed memory location places its data on the data bus. $\overline{\text{MRD}}$ is active from the end of T1 until the end of T4 during memory read transactions.
$\overline{\text{MSIZE}}$	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 $\overline{\text{BLEN}}$ pin description on page 38) $\overline{\text{MSIZE}}$ 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.
$\overline{\text{MWR}}$	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 $\overline{\text{BHEN}}$ and $\overline{\text{BLEN}}$. $\overline{\text{MWR}}$ is active from the end of T2 until the end of T4 during memory write transactions.
$\overline{\text{NMI}}$	135	**Nonmaskable Interrupt** (input, falling edge-triggered): This input has higher priority than the maskable interrupt inputs $\overline{\text{INT3-INT0}}$.
$\overline{\text{RESET}}$	134	**Reset** (input, active Low): This input must be active for a minimum of five BUSCLK periods to initialize the Z382. The effect of $\overline{\text{RESET}}$ is described in detail in the Reset section.
$\overline{\text{ROMCS}}$	42	**ROM Chip Select** (output, active Low): After Reset, the Z382 drives this output and $\overline{\text{MSIZE}}$ Low for all memory accesses with A23=0. Software can program the chip select logic to assert $\overline{\text{ROMCS}}$ 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 $\overline{\text{ROMCS}}$, and program the hardware not to force $\overline{\text{MSIZE}}$ Low in the first two instructions of the ROM code.

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.
$\overline{\text{HDAK0}}$ $\overline{\text{HDAK1}}$	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 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.

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.

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

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.

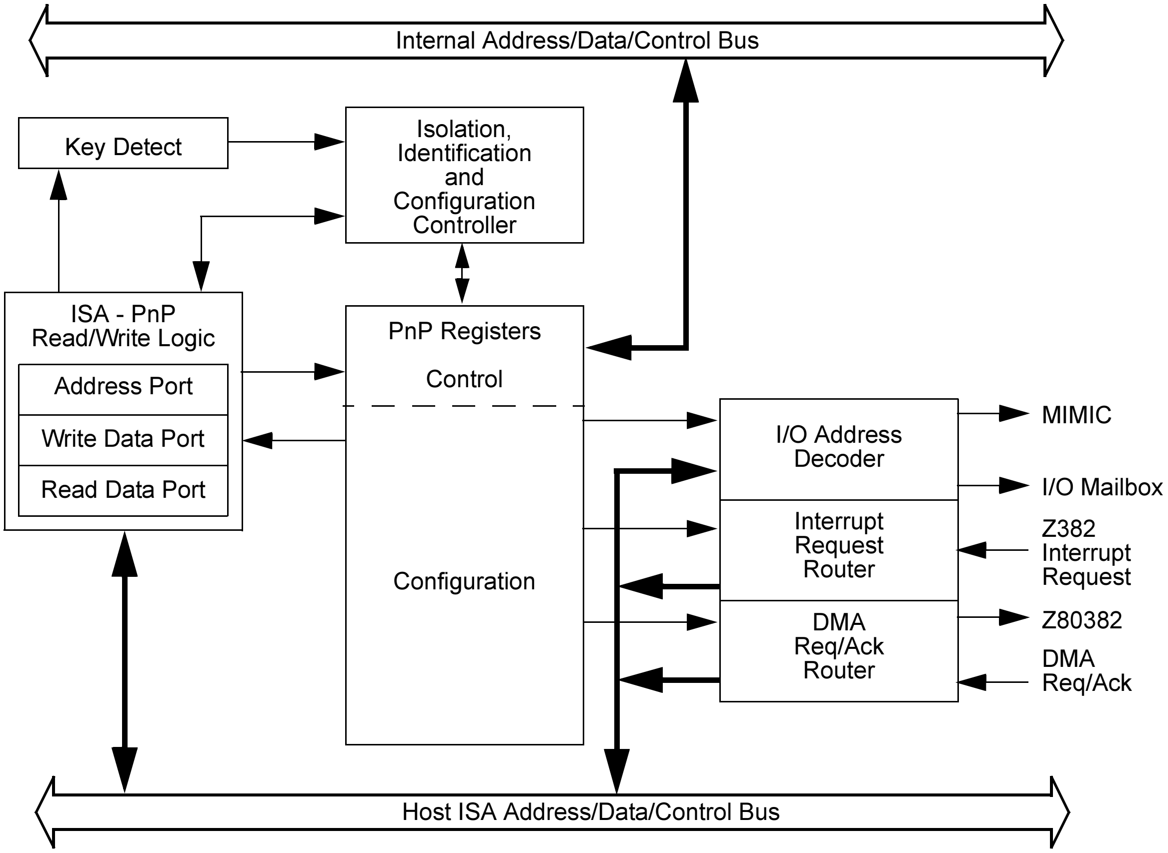


Figure 26. Plug-and-Play Interface Block Diagram

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.

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.

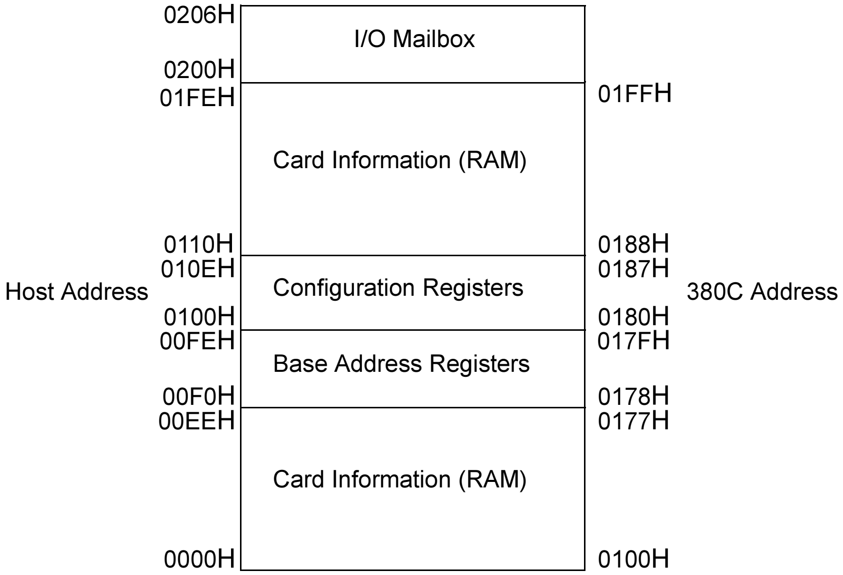


Figure 28. PCMCIA Attribute Memory Organization

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.

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

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

Figure 30 illustrates the major functional blocks within the ASCI.

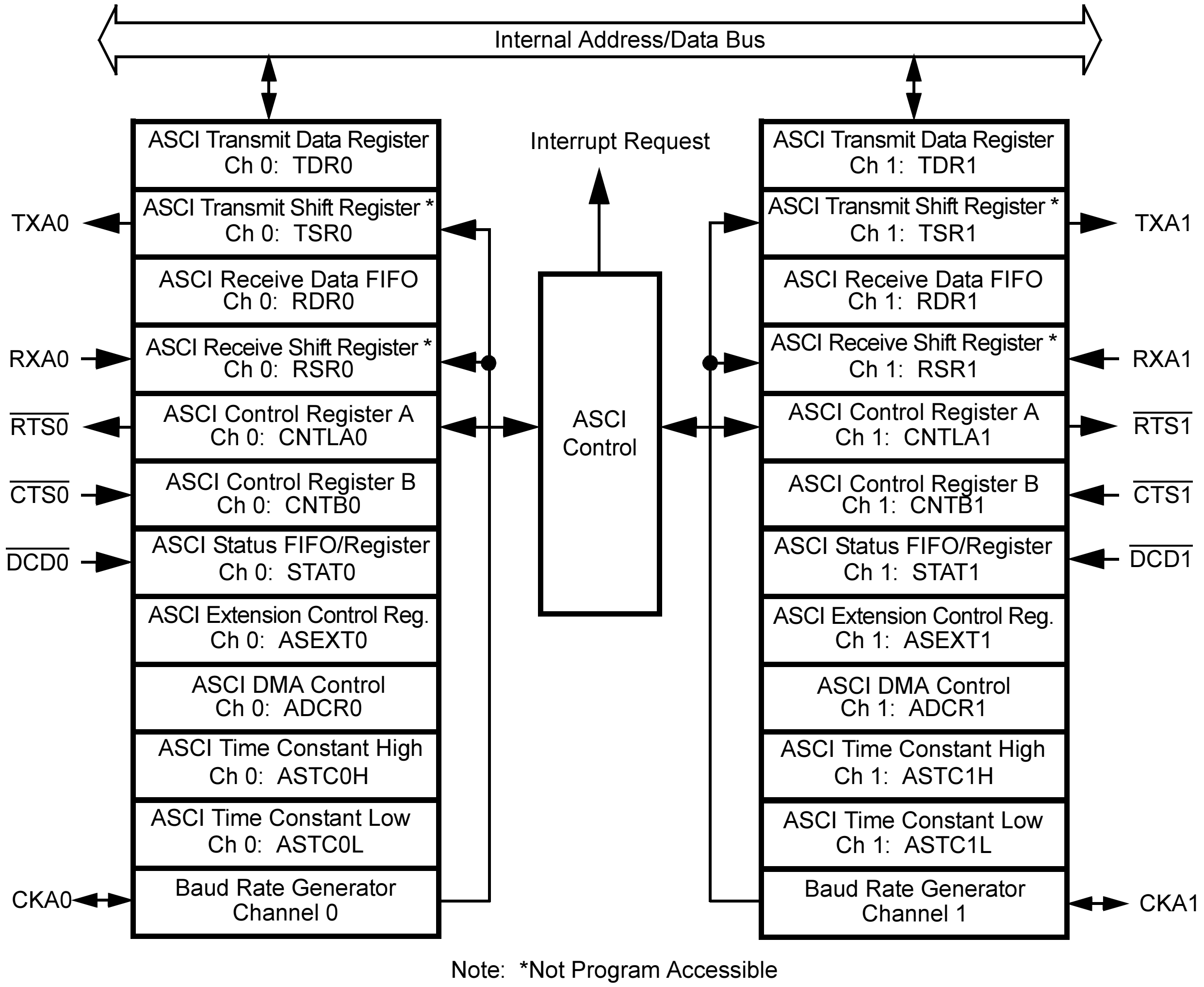

Internal Address/Data Bus
Interrupt Request
ASCI Control
ASCI Transmit Data Register Ch 0: TDR0
ASCI Transmit Shift Register * Ch 0: TSR0
ASCI Receive Data FIFO Ch 0: RDR0
ASCI Receive Shift Register * Ch 0: RSR0
ASCI Control Register A Ch 0: CNTLA0
ASCI Control Register B Ch 0: CNTB0
ASCI Status FIFO/Register Ch 0: STAT0
ASCI Extension Control Reg. Ch 0: ASEXT0
ASCI DMA Control Ch 0: ADCR0
ASCI Time Constant High Ch 0: ASTC0H
ASCI Time Constant Low Ch 0: ASTC0L
Baud Rate Generator Channel 0
TXA0
RXA0
RTS0
CTS0
DCD0
CKA0
ASCI Transmit Data Register Ch 1: TDR1
ASCI Transmit Shift Register * Ch 1: TSR1
ASCI Receive Data FIFO Ch 1: RDR1
ASCI Receive Shift Register * Ch 1: RSR1
ASCI Control Register A Ch 1: CNTLA1
ASCI Control Register B Ch 1: CNTB1
ASCI Status FIFO/Register Ch 1: STAT1
ASCI Extension Control Reg. Ch 1: ASEXT1
ASCI DMA Control Ch 1: ADCR1
ASCI Time Constant High Ch 1: ASTC1H
ASCI Time Constant Low Ch 1: ASTC1L
Baud Rate Generator Channel 1
TXA1
RXA1
RTS1
CTS1
DCD1
CKA1
Note: *Not Program Accessible

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.

- 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.

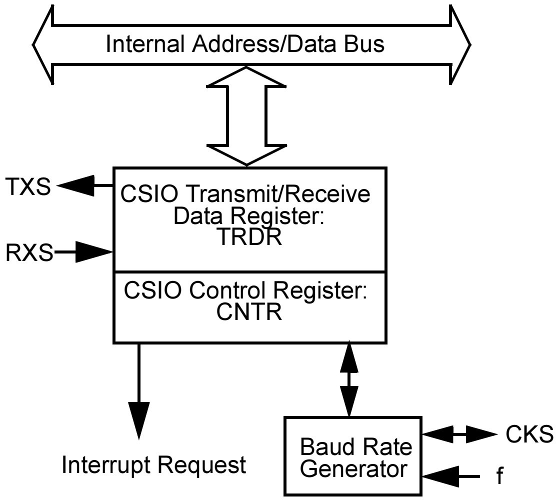
GCI Status Register 1. This register provides receive and transmit status conditions for Monitor 0 and 1 channels.

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.


Internal Address/Data Bus
TXS
RXS
CSIO Transmit/Receive Data Register: TRDR
CSIO Control Register: CNTR
Interrupt Request
Baud Rate Generator
CKS
f

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:

- 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