



PRELIMINARY *

Product Specification

AHA3580

80 MBytes/sec ALDC Data Compression Coprocessor IC

2365 NE Hopkins Court
Pullman, WA 99163-5601
tel: 509.334.1000
fax: 509.334.9000
e-mail: sales@aha.com
www.aha.com



advancedhardwarearchitectures

* *This specification represents a product still in the design cycle, undergoing testing processes, any specifications are based on design goals only. Parameters may be subject to change pending completion of characterization.*

Table of Contents

1.0	Introduction	1
1.1	Conventions, Notations and Definitions	1
1.2	Features	1
1.3	Applications	1
1.4	Functional Description	2
1.4.1	Port A and Port B Interfaces	3
1.4.1.1	FAS466 DMA Mode	3
1.4.1.2	Initiator Synchronous DMA Mode	3
1.4.2	Data Expansion During Compression	4
1.4.3	Multiple Records	4
1.4.4	Byte Alignment	4
2.0	Compression Operation	5
2.1	Compression Pass Through	5
2.2	Compression	5
3.0	Decompression Operation	6
3.1	Decompression Pass Through	6
3.2	Decompression	6
3.3	Decompression Output Disabled Mode	6
4.0	Microprocessor Interface and Register Access	6
4.1	Microprocessor Interface	6
4.1.1	Interrupts	6
4.1.2	Resets	7
4.1.3	Port A Interface FIFO Access	7
4.2	Register Access	8
4.3	Pausing / Resume	8
5.0	Port A and Port B Configuration	8
6.0	Register Description	9
6.1	Status 0 (STAT0)	10
6.2	Status 1 (STAT1)	11
6.3	Port A Configuration 0 (ACNF0)	12
6.4	Port A Configuration 1 (ACNF1)	12
6.5	Port B Configuration 0 (BCNF0)	12
6.6	Port B Configuration 1 (BCNF1)	13
6.7	Identification (ID0, ID1)	13
6.8	Port A Polarity (APOL)	13
6.9	Port B Polarity (BPOL)	14
6.10	Port A Transfer Count (ATCL0, ATCL1, ATCH0, ATCH1)	14
6.11	Record Count (RCL0, RCL1, RCH0, RCH1)	15
6.12	Port B Compare Count (BCCL0, BCCL1, BCCH0, BCCH1)	15
6.13	Port B Transfer Count (BTCL0, BTCL1, BTCH0, BTCH1)	16
6.14	Port A FIFO Data Access (AFIF0, AFIF1)	16
6.15	Compressed Bytes Processed (CBPL0, CBPL1, CBPH0, CBPH1)	17
6.16	Port A FIFO Control (AFCT)	17
6.17	Error Status (ERRS)	18
6.18	Interrupt Status 0 (INTS0)	18
6.19	Interrupt Status 1 (INTS1)	19
6.20	Command (CMND)	20
6.21	Record Length (RLL0, RLL1, RLH0, RLH1)	21
6.22	Data Disabled Count (DDCL0, DDCL1, DDCH0, DDCH1)	22
6.23	Error Mask (EMSK)	22

6.24	<i>Interrupt Mask 0 (IMSK0)</i>	23
6.25	<i>Interrupt Mask 1 (IMSK1)</i>	23
7.0	<i>Signal Descriptions</i>	24
7.1	<i>Microprocessor Interface</i>	24
7.2	<i>Port A Interface</i>	25
7.3	<i>Port B Interface</i>	26
8.0	<i>Pinout</i>	27
9.0	<i>Electrical Specifications</i>	29
9.1	<i>Absolute Maximum Ratings</i>	29
9.2	<i>Recommended Operating Conditions</i>	29
9.3	<i>DC Specifications</i>	29
10.0	<i>Timing Specifications</i>	30
11.0	<i>Packaging</i>	41
12.0	<i>Ordering Information</i>	42
12.1	<i>Available Parts</i>	42
12.2	<i>Part Numbering</i>	42
13.0	<i>AHA Related Technical Publications</i>	42

Figures

Figure 1:	Functional Block Diagram	2
Figure 2:	Multiple Record Compression	4
Figure 3:	Port A Interface Input Padding	5
Figure 4:	TQFP Pinout	28
Figure 5:	Clock Timing	30
Figure 6:	Reset Timing	30
Figure 7:	Processor Read Timing, MMODE = 1	31
Figure 8:	Processor Write Timing, MMODE = 1	32
Figure 9:	Processor Read Timing, MMODE = 0	33
Figure 10:	Processor Write Timing, MMODE = 0	34
Figure 11:	Port A, FAS466 DMA Slave Mode Read	35
Figure 12:	Port A, FAS466 DMA Slave Mode Write	36
Figure 13:	Port B, FAS466 DMA Master Mode Read	37
Figure 14:	Port B, FAS466 DMA Master Mode Write	38
Figure 15:	Port A, Initiator Synchronous DMA Mode In Timing	39
Figure 16:	Port A, Initiator Synchronous DMA Mode Out Timing	39
Figure 17:	Port B, Initiator Synchronous DMA Mode Out Timing	40
Figure 18:	Port B, Initiator Synchronous DMA Mode In Timing	40
Figure 19:	AHA3580 TQFP Package Specifications	41

Tables

Table 1:	Microprocessor Interface Control Signals	6
Table 2:	Port A Interface Signals	8
Table 3:	Port B Interface Signals	8
Table 4:	Clock Timing	30
Table 5:	Reset Timing	30
Table 6:	Processor Read Timing, MMODE = 1	31
Table 7:	Processor Write Timing, MMODE = 1	32
Table 8:	Processor Read Timing, MMODE = 0	33
Table 9:	Processor Write Timing, MMODE = 0	34
Table 10:	Port A, FAS466 DMA Slave Mode Read	35
Table 11:	Port A, FAS466 DMA Slave Mode Write	36
Table 12:	Port B, FAS466 DMA Master Mode Read	37
Table 13:	Port B, FAS466 DMA Master Mode Write	38
Table 14:	Port A, Initiator Synchronous DMA Mode In Timing	39
Table 15:	Port A, Initiator Synchronous DMA Mode Out Timing	39
Table 16:	Port B, Initiator Synchronous DMA Mode Out Timing	40
Table 17:	Port B, Initiator Synchronous DMA Mode In Timing	40
Table 18:	TQFP (Thin Quad Flat Pack) 14 × 14 mm Package Dimensions	41

1.0 INTRODUCTION

AHA3580 is a single chip lossless compression and decompression integrated circuit implementing the industry standard lossless adaptive data compression algorithm, also known as ALDC. The device compresses, decompresses or passes through data unchanged depending on the operating mode selected. This device achieves an average compression ratio of 2:1 on typical computer files. The flexible hardware interface makes this part suitable for many applications.

Port A DMA interface connects directly to popular industry standard SCSI controllers from QLogic and ST Microelectronics (Adaptec designed) and a fibre channel controller from QLogic (FAS440).

Content Addressable Memory (CAM) within the compression/decompression engine eliminates the need for external SRAMS.

Included in this specification is a functional overview, operation modes, register descriptions, DC and AC Electrical characteristics, ordering information, and a listing of related technical publications. It is intended for hardware and software engineers designing a compression system using AHA3580.

AHA designs and develops lossless compression, forward error correction and data storage formatter/controller ICs. Technical publications are available upon request.

1.1 CONVENTIONS, NOTATIONS AND DEFINITIONS

- Active low signals have an “N” appended to the end of the signal name. For example, CSN and WRITEN.
- “Signal assertion” means the signal is logically true.
- Hex values are represented with a prefix of “0x”, such as Register “0x00”. Binary values do not contain a prefix, for example, MMODE = 1.
- A prefix or suffix of “x” indicates a letter missing in a register name or signal name. For example, xCNF0 refers to the ACNF0 or BCNF0 register.
- A range of signal names or register bits is denoted by a set of colons between the numbers. Most significant bit is always shown first, followed by least significant bit. For example, MDATA[7:0] indicates signal names MDATA7 through MDATA0.
- Mega Bytes per second is referred to as MBytes/sec or MB/sec.
- IBM is a registered trademark of IBM.

1.2 FEATURES

PERFORMANCE:

- 80 MB/s data compression, decompression or pass-through rate with a single 80 MHz clock
- 2:1 average compression ratio
- A four byte *Record Length* register allows record lengths up to 4 gigabytes
- Four byte *Record Count* register allows multiple record transfers
- Error checking in decompression mode reportable via an interrupt

FLEXIBILITY:

- Polled or interrupt driven I/O
- Port A/B DMA interfaces include FAS466, FAS440 and AIC-43C97C
- Programmable polarity for DMA control signals
- DMA FIFO access via microprocessor port at Port A Interface

SYSTEM INTERFACE:

- Single chip data compression solution
- Two selectable microprocessor interfaces
- Programmable Interrupts
- Interfaces directly with industry standard SCSI chips and FAS440 fibre channel controller

OTHERS:

- Open standard ALDC adaptive lossless compression algorithm
- Complies to QIC-154, ECMA 222, ANSI X3.280-1996 and ISO 15200 standard specifications
- Algorithm compatible to IBM ALDC1-20S-HA and IBM ALDC1-20S-LP, and AHA3520
- 100 pin package in 14 × 14 mm TQFP body
- Lower power 3.3 Volt device

1.3 APPLICATIONS

- Tape drives
- Network communications – wired and wireless

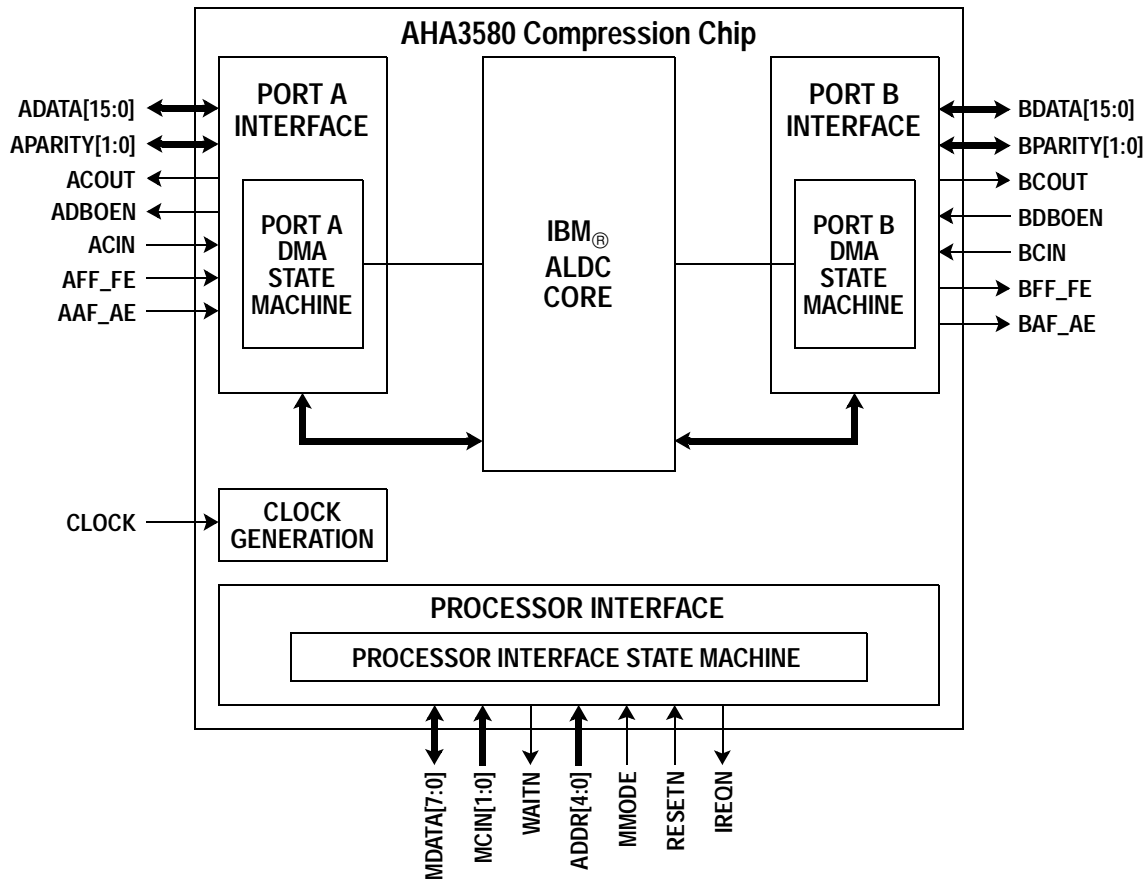
1.4 FUNCTIONAL DESCRIPTION

AHA3580 is a compression/decompression device residing between the host interface, usually SCSI, and the buffer manager ASIC. Major blocks in this device are the Microprocessor Interface, Port A Interface, Port B Interface, and the Compression/Decompression Engine. The Microprocessor Interface provides status and control information by register access. Port A and Port B Interfaces are configurable for polarity, handshaking modes, and other options. The operating mode establishes the direction of both the Port A and Port B Interfaces. Compression or Compression Pass Through sets the Port A Interface as an input and the Port B Interface as an output. Conversely Decompression or

Decompression Pass Through sets the Port A Interface as an output and the Port B Interface as an input. Decompression Output Disabled mode allows the device to decompress a user programmed number of records while dumping the uncompressed data, then automatically begin outputting the remaining uncompressed records.

A four byte *Record Length* register and a four byte *Record Count* register allow the user to partition the data into multiple records. Compression Pass Through mode and Decompression Pass Through modes allow data transfers through the device without changing the data. Both interfaces, Port A and Port B, have selectable transfer modes.

Figure 1: Functional Block Diagram



IBM is a registered trademark of IBM.

1.4.1 PORT A AND PORT B INTERFACES

Both Port A and Port B Interfaces are independently configurable via the *Port A Configuration* registers (ACNFx), the *Port A Polarity* register (APOL), the *Port B Configuration* registers (BCNFx), and the *Port B Polarity* register (BPOL). Both Ports may be configured to operate in FAS466 mode or Initiator Synchronous mode.

1.4.1.1 FAS466 DMA MODE

The FAS466 mode interface is capable of 160 MBytes/sec burst data transfers into or out of the Port A and Port B Interfaces.

A data transfer consists of DREQx followed by DACKx asserting. DACKx is only asserted when the FIFO status signals, xFF_FE and xAF_AE, permit a transfer. Slave Mode Read operation differs from Slave Mode Write such that data transfers are delayed one clock with Slave Mode Read. Transfers are also delayed one clock with Port B FAS466 Master Mode Write operation. Transfers are not delayed one clock after a valid DACKx with Port A FAS466 Slave Mode Write and Port B FAS466 Master Mode Read transfers.

FAS466 Port A Slave Mode Read operation transfers data from the external device to the AHA3580 Port A interface. During a read operation, ADBOEN must be asserted. A 16-bit transfer occurs on the second clock cycle after assertion of DACKA. Transfers continue always delayed to the second clock after a valid DACKA. Port B FAS466 Master Mode Write operation functions similarly since data transfers also are delayed to the second clock after a valid DACKB signal.

FAS466 Port A Slave Mode Write operation transfers data from the AHA3580 Port A interface to the external device. ADBOEN must be deasserted for a write transfer. When DACKA is asserted, each rising edge of clock transfers a 16-bit data word to the external device. Port B Master Mode Read operation functions similarly since data transfers occur every clock edge if DACKB is asserted.

The AHA3580 monitors the input FIFO status signals, AFF_FE and AAF_AE, at Port A and controls the transfer via the DACKA signal, thus avoiding data loss and/or FIFO corruption. When ADBOEN is asserted during a read transfer, the external FAS466 device drives the 16-bit data and parity onto the ADATA[15:0] and APARITY[1:0] for transfers into the AHA3580 device.

When AFF_FE is asserted, data can not be transferred (DACKx will not be asserted). When AAF_FE is asserted, data is transferred every other clock while sampling AFF_FE.

The AHA3580 Port B asserts BFF_FE when the FIFO is empty during a write transfer or full during a read transfer indicating that no more data may be transferred. Port B asserts BAF_AE when the FIFO is almost empty during a write transfer or almost full during a read transfer. When BAF_AE is asserted, transfers should be done every other clock while checking the status of the BFF_FE signal to determine if another transfer can be done. DREQB will remain asserted during the entire transfer.

1.4.1.2 INITIATOR SYNCHRONOUS DMA MODE

This mode is compatible with the SCSI DMA Initiator Synchronous mode of the AIC-43C97C device from ST Microelectronics. The SCSI controller should be programmed for two clock wide transfer cycles, 16-bit interface, and synchronous mode. The maximum transfer rate in this mode is 40 Mega transfers per second with an 80 MHz clock, or 80 Mbytes per second.

During a decompression or decompression pass through operation data is transferred from Port A to the external SCSI controller. The AHA device drives the data on the ADATA[15:0] and asserts DREQA. The external device accepts the data and responds by asserting a DACKA. Multiple DREQA pulses along with data may occur before the first DACKA gets asserted.

During a compression or compression pass through operation data is transferred from the external SCSI controller to Port A. In this mode the AHA device generates a DREQA pulse. The external device responds by driving the data onto the bus and asserting the DACKA signal. The external device may receive multiple DREQA pulses before responding with the first data word and a DACKA pulse.

The AHA device will stop generating DREQA pulses if the number generated is 16 greater than the number of DACKA pulses received. The total number of DACKA pulses must match to total number of DREQA pulses.

Port B Initiator Synchronous DMA Mode is similar except it operates as a Slave port and the DMA count is programmed in the external device. During a decompression or decompression pass through operation data is transferred from the external device to Port B. The external device drives the data on the BDATA[15:0] and asserts DREQB. The AHA3580 accepts the data and responds by asserting DACKB. Multiple DREQB pulses along with data may occur before the first DACKB gets asserted.

During a compression or compression pass through operation data is transferred from Port B to the external device. In this mode the external device generates a DREQB pulse. The AHA device responds by driving the data onto the bus and asserting the DACKB signal. The AHA device may receive multiple DREQA pulses, up to a maximum of 16, before responding with the first data word and a DACKB pulse.

The total number of DACKB pulses must match the total number of DREQB pulses. The external device may not generate a DREQB if the DREQB count is 16 greater than the DACKB count.

1.4.2 DATA EXPANSION DURING COMPRESSION

Data expansion occurs when the size of the data increases during a compression operation. This typically occurs when the data is compressed prior to input into the chip. The EXPAND status bit is set if the *Port B Transfer Count* is larger than the *Port A Transfer Count* register. If data expansion caused the *Port B Transfer Count* to exceed its maximum 4-byte value then the BTC Overflow Error status gets set. Worst case expansion allowable by the algorithm is 12.5% or (9/8 times the uncompressed Record Length).

1.4.3 MULTIPLE RECORDS

The AHA3580 device has two provisions to manage compressing a block of data into multiple records: automatic segmentation into multiple records at the Port A interface and the Reset history buffer command. During compression operation,

the Port A interface automatically partitions the uncompressed data into equal length records according to the *Record Count* and *Record Length* registers. The two sets of registers determine the number of records and length of each record in the data transfer operation. When compressing multiple records the device retains the contents of the history buffer between records. This usually improves compression ratio by allowing data from the current record to match against data from the previous record. During decompression, the previous record must be decompressed prior to the current record unless the history buffer is reset just before compressing the current record. For example, Figure 2 shows three records with a history buffer reset before record three. In this case, record three can be decompressed without previously decompressing records one and two. However, decompressing record two requires decompressing record one first.

When processing multiple records (*Record Count* is greater than one), the *Record Length* must be greater than 0x22.

1.4.4 BYTE ALIGNMENT

Both the Port A and Port B interfaces support the insertion and removal of padding bytes to align data transfers to any byte boundary within a two-byte or four-byte wide memory system. Figure 3 shows the four padding possibilities. In this figure, padding bytes are designated P_i, and normal data bytes are designated D_i. Four bits within the command register are used to specify the desired input and output padding for a given command.

Pad bytes are not counted by any of the counters.

Figure 2: Multiple Record Compression

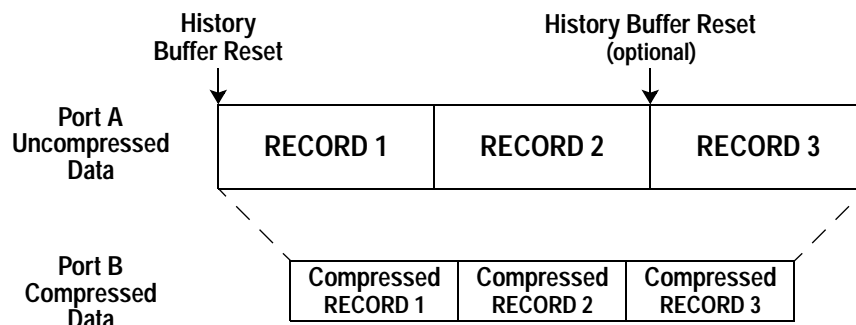
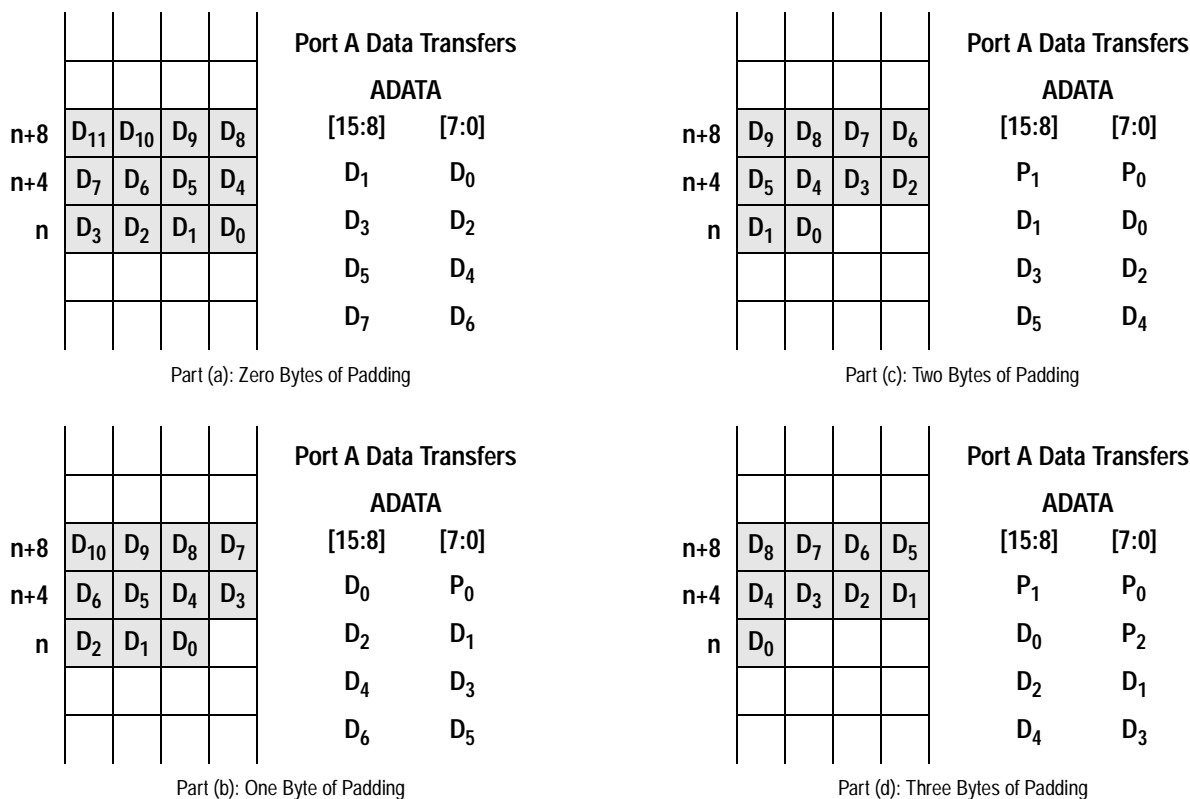


Figure 3: Port A Interface Input Padding



2.0 COMPRESSION OPERATION

2.1 COMPRESSION PASS THROUGH

Compression Pass Through mode allows data to enter the Port A Interface, transfer through the ALDC core and exit through the Port B Interface unchanged. Pass through mode uses the Port A Transfer counter, Port B Transfer counter and *Record Length* and *Record Count* registers. The DONE status bit and interrupt (if not masked) are set when the transfer completes.

2.2 COMPRESSION

During compression operation, uncompressed data flows into the Port A Interface, is compressed by the compression engine, and the compressed data transferred out of the Port B Interface.

The device contains a Content Addressable Memory (CAM). The CAM is the history buffer during compression operation. The compressor appends an end marker control code to the end of the compressed data. It also pads the end of a transfer to a byte boundary with zeroes.

The compression engine constantly monitors the performance of compression for expansion during compression operation. When the Port B Transfer Count is larger than the Port A Transfer Count the EXPAND bit in the *Status 0* register is set indicating data expansion during compression operation.

Port A Interface count increments with each byte received and when this count equals the transfer size, all bytes in this transfer have been received into Port A.

A compression operation is complete when the last byte transfers out of the Port B Interface and the *Record Length* is zero and the *Record Count* is one, thus setting the DONE status bit and generating a Done Interrupt if it is not masked.

3.0 DECOMPRESSION OPERATION

3.1 DECOMPRESSION PASS THROUGH

Decompression Pass Through mode allows data to enter the Port B Interface, transfer through the ALDC core and exit through the Port A Interface unchanged. Pass through mode uses the Port A Transfer counter, Port B Transfer counter, *Record Length* and *Record Count* registers. The DONE status bit and interrupt (if not masked) are set when the transfer completes.

3.2 DECOMPRESSION

During Decompression mode, compressed data flows into the Port B Interface and is decompressed. The resulting uncompressed data is transferred out of the Port A Interface.

A decompression operation is complete when the last byte transfers out of the Port A Interface, thus setting the DONE status bit and generating a Done Interrupt if it is not masked.

Decoder Control Code Errors are generated if invalid control codes are detected in the compressed data stream. This error is reported in the *Error Status* register.

Multiple records can be decompressed by programming the *Record Count* register. The *Record Count* register decrements every time an End of Record is decoded.

3.3 DECOMPRESSION OUTPUT DISABLED MODE

Decompression output disabled mode allows the user to program the number of records into the *Data Disable Count* register to decompress while discarding the output. The device then switches to normal decompression mode and continues to decompress the remaining records determined by the remaining number of records in the *Record Count* register, and transfers this data out of Port A.

4.0 MICROPROCESSOR INTERFACE AND REGISTER ACCESS

4.1 MICROPROCESSOR INTERFACE

Microprocessor Interface configuration is determined by the MMODE pin. If MMODE is tied high, transfers are controlled by a chip select signal (CSN) and a read/write signal (RWN), if MMODE is tied low, transfers are controlled by separate read (READN) and write (WRITEN) signals. Refer to Section 10.0 *Timing Specifications* for timing diagrams.

Table 1: Microprocessor Interface Control Signals

PIN NAME	MMODE TIED LOW	MMODE TIED HIGH
MCIN[0]	READN	CSN
MCIN[1]	WRITEN	RWN
WAITN	WAITN	WAITN
ADDR[0]	ADDR[0] = 0 selects register bits 7:0 ADDR[0] = 1 selects register bits 15:8	ADDR[0] = 0 selects register bits 15:8 ADDR[0] = 1 selects register bits 7:0

4.1.1 INTERRUPTS

IREQN is the hardware interrupt signal. IREQN is a standard TTL output. When active, it indicates an interrupt is set in the device. The microprocessor can determine the cause of the interrupt by reading the *Interrupt Status* register.

Masking individual interrupts with the *Interrupt Mask* register disables particular interrupts from causing the interrupt signal pin to assert (IREQN).

The interrupt signals are reset to their inactive state when either a hardware or software reset occurs, new compression operation begins, or by writing a zero to the *Interrupt Status* bit.

In general, the *Interrupt Status* and *Status* bits get set even if the *Interrupt Mask* bits are set. The exceptions are the One Byte at Port B, End of Record at Port B, One Byte at Port A, and End of Record at Port A. If these interrupts are masked, this status information can only be provided at the end of transfer, not at end of records because the ALDC core does not identify end of records in the data stream.

4.1.2 RESETS

There is a hardware reset signal and a software reset. When the RESETN signal is asserted all registers are reset, current operations are cancelled, and the history buffer is cleared. The software reset via the *Command* register does not affect the *Configuration* registers (ACNF or BCNF), *Identification* register (ID), the *Polarity* registers (APOL or BPOL), or the *Command* register (CMND). All other registers are reset, current operations cancelled and the history buffer cleared.

Section 6.0 *Register Description* lists the register values after a hardware reset, software reset command, and after a transfer command.

A new transfer command does not reset the data path; therefore, a hardware reset or software reset is generally required prior to issuing a new transfer command.

4.1.3 PORT A INTERFACE FIFO ACCESS

It is possible to access the Port A Interface FIFO from the microprocessor interface. This allows the uncompressed data stream to be altered from the microprocessor. This may be useful to properly handle exception conditions. Both read and write accesses are available. Only the Port A Interface FIFO is accessible from the microprocessor interface. In order to access the FIFO from the microprocessor interface, data transfers on the Port A interface must be suspended. The DMA device attached to the Port A interface must deactivate the DREQA line before attempting to access the FIFO from the microprocessor interface. Unpredictable results occur if DREQA is active during FIFO access from the microprocessor interface.

Two registers are used to control access to the FIFO: the *Port A FIFO Control* (AFCT) register and the *Port A FIFO Data* (AFIF) register. AFIF is a two-byte register used to hold data to be written to the Port A Interface FIFO during compression operations and to hold data read from the Port A Interface FIFO during decompression operations. Two bits within AFCT are defined: Access Port A FIFO (ACCF) and Request Port A FIFO (REQF). The Access Port A FIFO bit must be set for the entire duration of a read or write access to the Port A FIFO. This bit controls whether the Port A FIFO is accessed from the Port A interface or the microprocessor interface. The REQF bit is used as a semaphore to request a read or a write to the Port A Interface FIFO. Read or write is determined by the current command being executed. The FIFO can be read only during decompression commands and can be written only during compression commands.

Writing to the Port A Interface FIFO, assuming a compression or compression bypass operation is being executed, requires the following:

- 1) Suspend transfers on Port A Interface (DREQA input must be deasserted).
- 2) Write a Select Port A Command.
- 3) Set ACCF.
- 4) Place data to be written to the original data interface FIFO in AFIF.
- 5) Set REQF.
- 6) Read REQF until REQF returns to a zero.
- 7) Repeat steps 3 to 5 as necessary.
- 8) Clear ACCF and resume DMA operations.

Reading from the Port A Interface FIFO, assuming a decompression bypass, decompression or decompression output disabled operation is being executed, requires the following:

- 1) Suspend transfers on Port A Interface (DREQA input must be deasserted).
- 2) Write a Select Port A Command.
- 3) Set ACCF.
- 4) Set REQF.
- 5) Read REQF until REQF returns zero. REQF is reset when two bytes have been read from the Port A Interface FIFO and placed in AFIF.
- 6) Read data from AFIF.
- 7) Repeat steps 3 to 5 as necessary.
- 8) Clear ACCF and resume DMA operations.

All Port A interface status indicators are updated exactly as if the data is read from or written to the Port A interface data bus. For instance:

- The Port A Interface Transfer Count (ATC) will increment as bytes are transferred through the microprocessor interface.
- All Status bits (STAT0 and STAT1) and Interrupt Status bits (INTS) will operate when data is transferred through the microprocessor interface.
- Padding bytes are supported at command boundaries.
- Padding bytes may have to be inserted to ensure that the last transfer from the microprocessor ends on an even-byte boundary.

4.2 REGISTER ACCESS

MMODE determines whether ADDR[0] selects even or odd addressed registers. When MMODE = 1 and ADDR[0]=0, odd addressed registers are accessible. MMODE=1 causes ADDR[0] input signal to be inverted.

The registers may not be stable if PAUSED is not set. Registers should only be written when they are stable.

When writing to registers that are defined as 16-bit registers, both bytes must be written before the register is updated. When writing the 16-bit *Command* register, the command is executed when the most significant byte is written. ADDR[0] selects between the upper and lower bytes of 16-bit registers.

Registers in the ALDC core require longer to access than the external microprocessor interface permits. Therefore, if back to back writes to the same address ever occur, they must be separated by a minimum of 8 clocks.

4.3 PAUSING / RESUME

When a Pause command is issued or an unmasked data transfer interrupt occurs, the device pauses at the next break in the DMA handshaking. The following unmasked interrupts cause the device to pause: ODT (Output Disable Terminated), EORPA (End of Record at Port A), BPA (One Byte at Port A), EORPB (End of Record at Port B), BPB (One Byte at Port B), BCMP (Port B Interface Compare), and EORD (End of Record at Decoder). When a port is in slave mode, it pauses after xCOUT

(DACKx) deasserts. When a port is in master mode, the PAUSED status bit will get set even if xCOUT (DREQx) is asserted. However, in this case, several transfers could occur before the interface pauses and DREQx remains deasserted. Status updates and no more transfers will occur. Once paused and the last transfer is complete, the data busses are put in high impedance. Operation is continued by issuing a resume command

Registers in the ALDC core require longer to access than the external microprocessor interface permits. Therefore, these registers must be prefetched for external reads. To assure that the values read from these registers are current, it is recommended that a Pause command be issued and Paused Status read prior to reading these registers. When a pause command is received, it takes up to 40 clock cycles to update these registers. The PAUSED status bit is not set until the registers are updated. Additional microprocessor accesses during this time will delay the prefetched reads and paused status. Registers that must be prefetched include the *Compressed Bytes Processed*, *Error Status*, *Interrupt Status*, *Record Count* and *Data Disable Count* registers.

5.0 PORT A AND PORT B CONFIGURATION

Port A and Port B are both 16-bit bidirectional data ports with parity checking and generation. The ports are controlled by the configuration registers ACNF[15:0] and BCNF[15:0], and polarity registers APOL[7:0] and BPOL[7:0].

Table 2: Port A Interface Signals

SIGNAL NAME	AIC-43C97C	FAS466	APOL bit	DIRECTION
ACIN	DACKA	DREQA	7	I
ACOUT	DREQA	DACKA	5	O
ADBOEN	deasserted	ADBOEN	3	O
AFF_FE	not used	AFF_FE	1	I
AAF_AE	not used	AAF_AE	0	I

Table 3: Port B Interface Signals

SIGNAL NAME	FAS466	AIC-43C97C	BPOL bit	DIRECTION
BCIN	DACKB	DREQB	7	I
BCOUT	DREQB	DACKB	5	O
BDBOEN	BDBOEN	deasserted	3	I
BFF_FE	BFF_FE	not used	1	O
BAF_AE	BAF_AE	not used	0	O

6.0 REGISTER DESCRIPTION

ADDR[4:0]		MNEMONIC	REGISTER NAME	R/W	NOTES	REGISTER RESET VALUE			PAGE #
MMODE = 0	MMODE = 1					HARDWARE RESET	RESET COMMAND	NEW TRANSFER COMMAND	
0x00	0x01	STAT0	Status, Byte 0	R	1	0x00	0x00	0x80	10
0x01	0x00	STAT1	Status, Byte 1	R	1, 4	0x0C	0x0C	0000UU00	11
0x00	0x01	ACNF0	Port A Configuration, Byte 0	R/W	2	0x00	unchanged	unchanged	12
0x01	0x00	ACNF1	Port A Configuration, Byte 1	R/W	2	0x00	unchanged	unchanged	12
0x00	0x01	BCNF0	Port B Configuration, Byte 0	R/W	3	0x00	unchanged	unchanged	12
0x01	0x00	BCNF1	Port B Configuration, Byte 1	R/W	3	0x00	unchanged	unchanged	13
0x02	0x03	ID0	Identification 0	R	1	0x80	0x80	0x80	13
0x03	0x02	ID1	Identification 1	R	1	0x35	0x35	0x35	13
0x02	0x03	APOL	Port A Polarity	R/W	2	0xFF	unchanged	unchanged	13
0x03	0x02	<i>res</i>	<i>Reserved</i>						
0x02	0x03	BPOL	Port B Polarity	R/W	3	0xDF	unchanged	unchanged	14
0x03	0x02	<i>res</i>	<i>Reserved</i>						
0x04	0x05	ATCH0	Port A Transfer Count, Byte 2	R	1	0x00	0x00	0x00	14
0x05	0x04	ATCH1	Port A Transfer Count, Byte 3	R	1	0x00	0x00	0x00	14
0x04	0x05	RCH0	Record Count, Byte 2	R/W	2	0x00	0x00	0x00	15
0x05	0x04	RCH1	Record Count, Byte 3	R/W	2	0x00	0x00	0x00	15
0x04	0x05	BCCH0	Port B Compare Count, Byte 2	R/W	3	0x00	0x00	0x00	15
0x05	0x04	BCCH1	Port B Compare Count, Byte 3	R/W	3	0x00	0x00	0x00	15
0x06	0x07	ATCL0	Port A Transfer Count, Byte 0	R	1	0x00	0x00	0x00	14
0x07	0x06	ATCL1	Port A Transfer Count, Byte 1	R	1	0x00	0x00	0x00	14
0x06	0x07	RCL0	Record Count, Byte 0	R/W	2	0x00	0x00	0x00	15
0x07	0x06	RCL1	Record Count, Byte 1	R/W	2	0x00	0x00	0x00	15
0x06	0x07	BCCL0	Port B Compare Count, Byte 0	R/W	3	0x00	0x00	0x00	15
0x07	0x06	BCCL1	Port B Compare Count, Byte 1	R/W	3	0x00	0x00	0x00	15
0x08	0x09	BTCH0	Port B Transfer Count, Byte 2	R	1	0x00	0x00	0x00	16
0x09	0x08	BTCH1	Port B Transfer Count, Byte 3	R	1	0x00	0x00	0x00	16
0x08	0x09	AFIF0	Port A FIFO Data Access, Byte 0	R/W	2	0x00	0x00	0x00	16
0x09	0x08	AFIF1	Port A FIFO Data Access, Byte 1	R/W	2	0x00	0x00	0x00	16
0x08	0x09	CBPH0	Compressed Bytes Processed, Byte 2	R	3	0x00	0x00	0x00	17
0x09	0x08	CBPH1	Compressed Bytes Processed, Byte 3	R	3	0x00	0x00	0x00	17
0x0A	0x0B	BTCL0	Port B Transfer Count, Byte 0	R	1	0x00	0x00	0x00	16
0x0B	0x0A	BTCL1	Port B Transfer Count, Byte 1	R	1	0x00	0x00	0x00	16
0x0A	0x0B	AFCT	Port A FIFO Control	R/W	2	0x00	0x00	0x00	17
0x0B	0x0A	<i>res</i>	<i>Reserved</i>		2				
0x0A	0x0B	CBPL0	Compressed Bytes Processed, Byte 0	R	3	0x00	0x00	0x00	17
0x0B	0x0A	CBPL1	Compressed Bytes Processed, Byte 1	R	3	0x00	0x00	0x00	17
0x0C	0x0D	ERRS	Error Status	R	1	0x00	0x00	0x00	18
0x0D	0x0C	<i>res</i>	<i>Reserved</i>						
0x0E	0x0F	INTS0	Interrupt Status, Byte 0	R/W	1	0x00	0x00	0x00	18
0x0F	0x0E	INTS1	Interrupt Status, Byte 1	R/W	1	0x00	0x00	0x00	19
0x10	0x11	CMND0	Command 0	R/W		0x00	0x00	0x00	20
0x11	0x10	CMND1	Command 1	R/W		0x00	0xA0	0x00	20
0x12	0x13	<i>res</i>	<i>Reserved</i>						
0x13	0x12	<i>res</i>	<i>Reserved</i>						
0x14	0x15	RLH0	Record Length, Byte 2	R/W		0x00	0x00	unchanged	21
0x15	0x14	RLH1	Record Length, Byte 3	R/W		0x00	0x00	unchanged	21

ADDR[4:0]		MNEMONIC	REGISTER NAME	R/W	NOTES	REGISTER RESET VALUE			PAGE #
MMODE = 0	MMODE = 1					HARDWARE RESET	RESET COMMAND	NEW TRANSFER COMMAND	
0x16	0x17	RLL0	Record Length, Byte 0	R/W		0x00	0x00	unchanged	21
0x17	0x16	RLL1	Record Length, Byte 1	R/W		0x00	0x00	unchanged	21
0x18	0x19	DDCH0	Data Disabled Count, Byte 2	R/W		0x00	0x00	unchanged	22
0x19	0x18	DDCH1	Data Disabled Count, Byte 3	R/W		0x00	0x00	unchanged	22
0x1A	0x1B	DDCL0	Data Disabled Count, Byte 0	R/W		0x00	0x00	unchanged	22
0x1B	0x1A	DDCL1	Data Disabled Count, Byte 1	R/W		0x00	0x00	unchanged	22
0x1C	0x1D	EMSK	Error Mask	R/W		0x00	0x00	unchanged	22
0x1D	0x1C	<i>res</i>	<i>Reserved</i>						
0x1E	0x1F	IMSK0	Interrupt Mask 0	R/W		0x00	0x00	unchanged	23
0x1F	0x1E	IMSK1	Interrupt Mask 1	R/W		0x00	0x00	unchanged	23

Notes:

- 1) When CMND is not a Selection Command.
- 2) When CMND is a Select Port A Configuration Command.
- 3) When CMND is a Select Port B Configuration Command.
- 4) U identifies a bit that is unchanged.

6.1 STATUS 0 (STAT0)

Read Only

Hardware Reset Value = 0x00

Reset Command = 0x00

MMODE =		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	1								
0x00	0x01	BUSY	PAUSED	OUTDIS	BYPASS	EXPAND	ANYINT	ANYERR	DONE

Any status bit which is active when the device pauses, due to an interrupt or Pause Command, will remain active until there is a Resume Command.

BUSY - Busy. This bit is set when a data transfer operation begins. It is cleared when the data transfer operation completes successfully, when an unmasked error occurs, or when a reset occurs.

PAUSED - Paused. This bit is set when a data transfer operation is currently paused. It is cleared when a paused data transfer operation is resumed, when a reset occurs, or on a new transfer.

OUTDIS - Output Disabled. This bit is set when Port A Interface output is disabled. It is cleared when Port A Interface output is re-enabled, when a reset occurs, or on a new transfer.

BYPASS - Bypass. This bit is set after a Start Compression Bypass or a Start Decompression Bypass command is written to the *Command* register. It is cleared after a Start Compression, Start Decompression, Start Decompression Output Disable, when a reset occurs, when an unmasked error occurs, or when a transfer is complete.

EXPAND - Expansion. This bit is set when the *Port B Transfer Count* register is larger than the *Port A Transfer Count* register. It may toggle many times during a compression operation. It is cleared when another data transfer operation begins or when a reset occurs.

ANYINT - Any Interrupt. This bit is set while an unmasked interrupt is active. Cleared on a new transfer, and when all unmasked interrupts have been cleared.

ANYERR - Any Error. This bit is set when an unmasked error occurs. It is cleared when a data transfer operation begins or when a reset occurs.

DONE - Done. This bit is set when the current data transfer operation is complete. It is cleared when a data transfer operation begins or when a reset occurs.

6.2 STATUS 1 (STAT1)

Read Only

Hardware Reset Value = 0x0C

Reset Command = 0x0C

<i>MMODE</i> =		<i>bit15</i>	<i>bit14</i>	<i>bit13</i>	<i>bit12</i>	<i>bit11</i>	<i>bit10</i>	<i>bit9</i>	<i>bit8</i>
0	1	EORD	BCMP	BPB	EORPB	EMPB	EMPA	BPA	EORPA
0x01	0x00								

The *Status* bits **BPB**, **EORPB**, **BPA** and **EORPA** will only get set after the last word is transferred if the following *Interrupt Mask* bits are set: **BPBM**, **EORPBM**, **BPAM** and **EORPAM**. If these bits are set, the ALDC core provides end of transfer information, but no end of record information.

- EORD** - End of Record at Decoder. This bit is set when the ALDC decoder detects an End of Record control code in the compressed data stream or when an ALDC Decoder Control Code Error occurs. This bit is cleared after reset, when the decoder begins processing the first codeword of the next record, or when a new data transfer operation begins. It is valid for Decompression and Decompression Output Disable modes.
- BCMP** - Port B Interface Compare. This bit is set when Port B Transfer Count is greater than or equal to Port B Interface Compare Count. Otherwise, it is cleared. This bit is cleared after reset or when a new data transfer operation begins. This bit is valid for all modes of operation.
- BPB** - One Byte at Port B. During compression bypass and compression operations, this bit is set at the same time the End of Record at Port B (STAT1[4] and INTS1[4]) is set if only one byte at the Port B Interface is part of the current record. During decompression bypass operation, this bit is set during the last data transfer of the record at the Port B Interface if only one byte belongs to the current record. This bit is cleared after reset, when a new data transfer operation begins, or when the first byte of the next record is transferred. Not valid during Decompression and Decompression Output Disable modes.
- EORPB** - End of Record at Port B. During compression bypass and compression operations, this bit is set when the last byte of a compressed record is transferred out of the Port B interface. During decompression bypass operations, this bit is set when the last byte of a record is transferred into the Port B interface. This bit is cleared after reset, when a new data transfer operation begins, or when the first byte of the next record is transferred. Not valid during Decompression and Decompression Output Disable modes.
- EMPB** - Empty at Port B. This bit is set when there is no data in the Port B interface data path. This bit must be set when writing to the *Record Length* register during Decompression bypass operation and when writing to the *Record Count* register during Decompression and Decompression Output disabled operations. Set after reset.
- EMPA** - Empty at Port A. This bit is set when there is no data in the Port A interface data path. This bit must be set when writing to the *Record Length* or *Record Count* registers during Compression and Compression Bypass operations. Set after reset.
- BPA** - One Byte at Port A. During compression bypass and compression operations, this bit is set during the last data transfer of the record at the Port A interface if only one byte belongs to the current record. During decompression bypass, decompression, and decompression output disabled modes, this bit is set the same time the End of Record at Port A interface bit (STAT1[0] and INTS1[0]) is set if only one byte at the Port A interface is part of the current record. This bit is cleared after reset, when a new data transfer operation begins, or when the first byte of the next record is transferred.
- EORPA** - End of Record at Port A. During compression bypass and compression operations, this bit is set each time the Record Length (RL) is decremented to zero. During decompression bypass, decompression, and decompression output disabled operations, this bit is set when the last byte of a record is transferred out the Port A interface. This bit is cleared after reset, when a new data transfer operation begins, or when the first byte of the next record is transferred.

6.3 PORT A CONFIGURATION 0 (ACNF0)

Read/Write
Hardware Reset Value = 0x00
Reset Command = unchanged

MMODE =

0	1	<i>bit7</i>	<i>bit6</i>	<i>bit5</i>	<i>bit4</i>	<i>bit3</i>	<i>bit2</i>	<i>bit1</i>	<i>bit0</i>
0x00	0x01	reserved							

6.4 PORT A CONFIGURATION 1 (ACNF1)

Read/Write
Hardware Reset Value = 0x00
Reset Command = unchanged

MMODE =

0	1	<i>bit15</i>	<i>bit14</i>	<i>bit13</i>	<i>bit12</i>	<i>bit11</i>	<i>bit10</i>	<i>bit9</i>	<i>bit8</i>
0x01	0x00	PARITY	ODD	SLAVE	MODE[2:0]			reserved	

PARITY - Parity. When set, parity checking is enabled for the ADATA[15:0] data bus. When cleared, parity checking is disabled for the ADATA[15:0] bus.

ODD - Odd. Setting this bit along with PARITY enables odd parity checking and generation on the ADATA[15:0] data bus. When cleared with PARITY set even parity checking and generation is enabled on the ADATA[15:0] data bus.

SLAVE - Slave. Must always be written with a one.

MODE[2:0]-DMA Mode. These bits configure the interface DMA mode of the Port A Interface with values as defined below.

<i>MODE[2:0]</i>	<i>DMA TYPE</i>	<i>MASTER/SLAVE</i>
000	Reserved	—
001	Reserved	—
010	Reserved	—
011	Reserved	—
100	Reserved	—
101	FAS466	SLAVE
110	43C97C SCSI Initiator	MASTER
111	reserved	—

6.5 PORT B CONFIGURATION 0 (BCNF0)

Read/Write
Hardware Reset Value = 0x00
Reset Command = unchanged

MMODE =

0	1	<i>bit7</i>	<i>bit6</i>	<i>bit5</i>	<i>bit4</i>	<i>bit3</i>	<i>bit2</i>	<i>bit1</i>	<i>bit0</i>	
0x00	0x01	reserved					FIFOTH[3:0]			

FIFOTH[3:0]-FIFO Threshold. These bits configure the Port A FIFO threshold value. Values from 0001 through 1111 are valid. A value of 0000 results in the same operation as 0001. Valid for FAS466 mode.

6.6 PORT B CONFIGURATION 1 (BCNF1)

Read/Write
Hardware Reset Value = 0x00
Reset Command = unchanged

MMODE =

<i>0</i>	<i>1</i>	<i>bit15</i>	<i>bit14</i>	<i>bit13</i>	<i>bit12</i>	<i>bit11</i>	<i>bit10</i>	<i>bit9</i>	<i>bit8</i>
0x01	0x00	PARITY	ODD	reserved	MODE[2:0]			reserved	

PARITY - Parity. When set, parity checking is enabled for the BDATA[15:0] data bus. When cleared, parity checking is disabled for the BDATA[15:0] bus.

ODD - Odd. When set, odd parity checking and generation is used on the BDATA[15:0] data bus. When cleared, even parity checking and generation is used on the BDATA[15:0] data bus.

MODE[2:0]-DMA Mode. These bits configure the interface DMA mode of the Port B Interface with values as defined below.

MODE[2:0]	DMA TYPE	MASTER/SLAVE
000	Reserved	—
001	Reserved	—
010	Reserved	—
011	Reserved	—
100	Reserved	—
101	FAS466	MASTER
110	43C97C SCSI Initiator	SLAVE
111	reserved	—

6.7 IDENTIFICATION (ID0, ID1)

Read Only
Hardware Reset Value = 0x3580
Reset Command = 0x3580

MMODE =

<i>0</i>	<i>1</i>	
0x02	0x03	ID[7:0]
0x03	0x02	ID[15:8]

ID[15:0]- The bits of this register correspond to the identification code of the chip. This register is accessible when CMND is not a Selection Command.

6.8 PORT A POLARITY (APOL)

Read/Write
Hardware Reset Value = 0xFF
Reset Command = unchanged

MMODE =

<i>0</i>	<i>1</i>	<i>bit7</i>	<i>bit6</i>	<i>bit5</i>	<i>bit4</i>	<i>bit3</i>	<i>bit2</i>	<i>bit1</i>	<i>bit0</i>
0x02	0x03	ACIN	reserved	ACOUT	reserved	ADBOEN	reserved	AFF_FE	AAF_AE

The bits of this register correspond to Port A Interface signals. A set bit programs the corresponding signal to be active low. A cleared bit programs the corresponding signal to be active high. This register is only accessible when CMND is Select Port A Configuration.

6.9 PORT B POLARITY (BPOL)

Read/Write
 Hardware Reset Value = 0xDF
 Reset Command = unchanged

<i>MMODE</i> =		<i>bit7</i>	<i>bit6</i>	<i>bit5</i>	<i>bit4</i>	<i>bit3</i>	<i>bit2</i>	<i>bit1</i>	<i>bit0</i>
0	1	BCIN	reserved	BCOUT	reserved	BDBOEN	reserved	BFF_FE	BAF_AE
0x02	0x03								

The bits of this register correspond to Port B Interface signals. A set bit programs the corresponding signal to be active low. A cleared bit programs the corresponding signal to be active high. This register is only accessible when CMND is Select Port B Configuration.

6.10 PORT A TRANSFER COUNT (ATCL0, ATCL1, ATCH0, ATCH1)

Read Only
 Hardware Reset Value = 0x00000000
 Reset Command = 0x00000000

Port A Transfer Count Low

<i>MMODE</i> =		
0	1	
0x06	0x07	ATCL[7:0]
0x07	0x06	ATCL[15:8]

Port A Transfer Count High

<i>MMODE</i> =		
0	1	
0x04	0x05	ATCH[7:0]
0x05	0x04	ATCH[15:8]

ATC[31:0]- Port A Transfer Count. These registers provide status information on the number of bytes transferred for a current data transfer operation. During a compression operation, **ATC** is incremented as each original data byte is received by the Port A Interface. When **ATC** equals the product of the Record Count and Record Length during compression, all bytes in the compression operation have been received by the AHA3580. During a decompression operation, **ATC** is incremented as each decompressed data byte is sent by the Port A Interface. This register is only accessible when CMND is not a Selection Command.

In the case where only one byte is required to complete a transfer operation (i.e., an odd number of bytes in the transfer), the **ATC** is incremented by one after the byte transfers. **ATC** should not be used to determine the decompression operation is complete. Instead, use the **DONE** status bit and/or interrupt. Data blocks of *Record Count* times *Record Length* must be smaller the $(2^{32}-1)$ to prevent overflow of this 4-byte *Transfer Count* register. Reset on new transfer commands. Pad bytes are not counted.

6.11 RECORD COUNT (RCL0, RCL1, RCH0, RCH1)

Read/Write
 Hardware Reset Value = 0x00000000
 Reset Command = 0x00000000

Record Count Low

MMODE =

0	1	
0x06	0x07	RCL[7:0]
0x07	0x06	RCL[15:8]

Record Count High

MMODE =

0	1	
0x04	0x05	RCH[7:0]
0x05	0x04	RCH[15:8]

RC[31:0]- Record Count indicates the number of records to be compressed or decompressed. Record Count must be set to 0x00000001 during Decompression Bypass. If the Record Count must be written to during a compression operation, then the Empty at Port A (**EMPA**) *Status* bit must be set. If the Record Count must be written to during a decompression operation, then the Empty at Port B (**EMPB**) *Status* bit must be set.

6.12 PORT B COMPARE COUNT (BCCL0, BCCL1, BCCH0, BCCH1)

Read/Write
 Hardware Reset Value = 0x00000000
 Reset Command = 0x00000000

Port B Compare Count Low

MMODE =

0	1	
0x06	0x07	BCCL[7:0]
0x07	0x06	BCCL[15:8]

Port B Compare Count High

MMODE =

0	1	
0x04	0x05	BCCH[7:0]
0x05	0x04	BCCH[15:8]

BCC[31:0] Port B compare count register is used to pause the device after a specified number of bytes are transferred at the Port B interface. Port B Compare Count is a four byte register with the two most significant bytes contained in Port B Compare Count High (BCCH), and the two least significant bytes contained in the Port B Compare Count Low register (BCCL).

6.13 PORT B TRANSFER COUNT (BTCL0, BTCL1, BTCH0, BTCH1)

Read Only

Hardware Reset Value = 0x00000000

Reset Command = 0x00000000

Port B Transfer Count Low

*M*MODE =

0	1	
0x0A	0x0B	BTCL[7:0]
0x0B	0x0A	BTCL[15:8]

Port B Transfer Count High

*M*MODE =

0	1	
0x08	0x09	BTCH[7:0]
0x09	0x08	BTCH[15:8]

BTC[31:0] -Port B Transfer Count. These registers provide status information on the number of bytes transferred for a current data transfer operation. During a compression operation, BTC is incremented as each compressed data byte is sent by the Port B Interface. During a decompression operation, BTC is incremented as each compressed data byte is received by the Port B Interface. This register is only accessible when CMND is not a Selection Command.

In the special case where only one byte is required to complete a transfer operation (i.e., an odd number of bytes in the transfer), the BTC is incremented by one after the byte transfers. BTC should not be used to determine the decompression operation is complete. Instead, use the DONE status bit and/or interrupt. Data blocks of Record Count times Record Length must be smaller than $(2^{32} - 1)$ to prevent overflow of this 4-byte transfer count register.

Reset by a new compression mode transfer command, but not by a new decompression mode transfer. Pad bytes are not counted.

6.14 PORT A FIFO DATA ACCESS (AFIF0, AFIF1)

Read/Write

Hardware Reset Value = 0x0000

Reset Command = 0x0000

*M*MODE =

0	1	
0x08	0x09	FA[7:0]
0x09	0x08	FA[15:8]

FA[15:0]- Port A FIFO Data register is a temporary holding register for data to be written to or read from the Port A interface FIFO. During compression bypass and compression operations, the Port A FIFO indicates it has received the data by resetting REQF in the AFCT register. During decompression bypass, decompression, and decompression output disabled operations, data may be read from this register after the Port A FIFO resets REQF in the AFCT register. This register is only accessible when CMND is a Select Port A Configuration Command. This register is reset by a new transfer.

6.15 COMPRESSED BYTES PROCESSED (CBPL0, CBPL1, CBPH0, CBPH1)

Read/Write
 Hardware Reset Value = 0x00000000
 Reset Command = 0x00000000

Compressed Bytes Processed Low

MMODE =

0	1	
0x0A	0x0B	CBPL[7:0]
0x0B	0x0A	CBPL[15:8]

Compressed Bytes Processed High

MMODE =

0	1	
0x08	0x09	CBPH[7:0]
0x09	0x08	CBPH[15:8]

CBPL[31:0] -Compressed Bytes Processed counter. Counts the total number of bytes processed by the ALDC decoder during decompression and decompression output disabled operations. It can be used in conjunction with the Port B Transfer Count to determine the number of compressed bytes, if any, that reside in the Port B interface and ALDC core.

6.16 PORT A FIFO CONTROL (AFCT)

Read/Write
 Hardware Reset Value = 0x00
 Reset Command = 0x00

MMODE =

0	1	<i>bit7</i>	<i>bit6</i>	<i>bit5</i>	<i>bit4</i>	<i>bit3</i>	<i>bit2</i>	<i>bit1</i>	<i>bit0</i>
0x0A	0x0B	reserved						ACCF	REQF

ACCF - Access FIFO. When set, access to the Port A FIFO is redirected from the Port A interface to the microprocessor interface. This bit is cleared after reset or a new transfer.

REQF - Request to FIFO. During compression bypass and compression operations, this bit is set to one requesting a write to the Port A FIFO. During decompression bypass, decompression, and decompression output disabled operations, this bit is set to one requesting a read from the Port A interface FIFO. This bit is cleared when the Port A FIFO has completed the request or after a reset. This register is only accessible when CMND is a Select Port A configuration command. Reset by a new transfer.

6.17 ERROR STATUS (ERRS)

Read Only

Hardware Reset Value = 0x00

Reset Command = 0x00

<i>MMODE</i> =		<i>bit7</i>	<i>bit6</i>	<i>bit5</i>	<i>bit4</i>	<i>bit3</i>	<i>bit2</i>	<i>bit1</i>	<i>bit0</i>
0	1								
0x0C	0x0D	reserved	APERR	BPERR	reserved	BTCO	ATCO	ADCC	reserved

The *Error Status* register provides error status bits to the microprocessor. These bits are set regardless of the error mask settings. Reset by a new compression mode transfer.

APERR - Port A Interface Parity Error. This bit is set when a parity error is detected during a transfer into *ADATA*[15:0] and the Port A Interface Parity bit is set. It is cleared when a new compression mode transfer begins or when a reset occurs.

BPERR - Port B Interface Parity Error. This bit is set when a parity error is detected during a transfer into *BDATA*[15:0] and the Port B Interface Parity bit is set. It is cleared when a new compression mode transfer begins or when a reset occurs.

BTCO - Port B Transfer Count Overflow Error. This bit is set when a carry out is detected on the *Port B Transfer Count* register. It is cleared when a new compression mode transfer begins or when a reset occurs.

ATCO - Port A Transfer Count Overflow Error. This bit is set when a carry out is detected on the *Port A Transfer Count* register. It is cleared when a new compression mode transfer begins or when a reset occurs.

ADCC - ALDC Decoder Control Code Error. This bit is set during decompression when an invalid control code is detected in the compressed data stream. It is cleared when a new compression mode transfer begins or when a reset occurs.

6.18 INTERRUPT STATUS 0 (INTS0)

Read Only

Hardware Reset Value = 0x00

Reset Command = 0x00

<i>MMODE</i> =		<i>bit7</i>	<i>bit6</i>	<i>bit5</i>	<i>bit4</i>	<i>bit3</i>	<i>bit2</i>	<i>bit1</i>	<i>bit0</i>
0	1								
0x0E	0x0F	DONE	PAUSED	ODT	reserved			ERROR	

Interrupt Status bits are reset by writing a zero. This is referred to as an interrupt reset. Writing a one has no effect.

DONE - Done Interrupt. This bit is set when data transfer has completed on the Port B Interface during compression and when data transfer has completed on the Port A Interface during decompression. It is cleared when a new compression mode transfer begins, when a reset occurs, or by an interrupt reset.

PAUSED - Paused Interrupt. This bit is set by a Pause command, or an unmasked data transfer interrupt. It is cleared when a new compression mode transfer begins, when a reset occurs, or by an interrupt reset.

ODT - Output Disabled Terminated. This bit is set when the end of record of the last suppressed record is processed by the ALDC decoder. This bit is cleared after reset, after an interrupt reset is written, or when a new compression mode transfer begins.

ERROR - Error Interrupt. This bit is set when an unmasked error occurs. It is cleared when a new compression mode transfer begins or when a reset occurs. The *Error Status* register is used to determine the cause of the error.

6.19 INTERRUPT STATUS 1 (INTS1)

Read/Write
 Hardware Reset Value = 0x00
 Reset Command = 0x00

<i>MMODE</i> =		<i>bit15</i>	<i>bit14</i>	<i>bit13</i>	<i>bit12</i>	<i>bit11</i>	<i>bit10</i>	<i>bit9</i>	<i>bit8</i>
0	1	EORD	BCMP	BPB	EORPB	reserved		BPA	EORPA
0x0F	0x0E								

The *Interrupt Status* bits **BPB**, **EORPB**, **BPA** and **EORPA** will only get set after the last word is transferred if the following *Interrupt Mask* bits are set: **BPBM**, **EORPBM**, **BPAM** and **EORPAM**. If these mask bits are set, the ALDC core provides end of transfer information, but no end of record information.

- EORD** - End of Record at Decoder, This bit is set when the ALDC decoder detects an End of Record control code in the compressed data stream or when an ALDC Decoder Control Code Error occurs. This bit is cleared after reset, when an interrupt reset is written, or when a new compression mode transfer begins.
- BCMP** - Port B Interface Compare. This bit is set when Port B Transfer Count is greater than or equal to Port B Interface Compare Count. This bit is cleared after reset, when an interrupt reset is written, or when a new compression mode transfer begins.
- BPB** - One Byte at Port B. During compression bypass and compression operations, this bit is set at the same time the End of Record at Port B (STAT1[4] and INTS1[4]) is set if only one byte at the Port B Interface is part of the current record. During decompression bypass operation, this bit is set during the last data transfer of the record at the Port B Interface if only one byte belongs to the current record. This bit is cleared after reset, when an interrupt reset is written, or when a new compression mode transfer begins.
- EORPB** - End of Record at Port B. During compression bypass and compression operations, this bit is set when the last byte of a compressed record is transferred out of the Port B interface. During decompression bypass operations, this bit is set when the last byte of a record is transferred into the Port B interface. This bit is cleared after reset, when an interrupt reset is written, or when a new compression mode transfer begins.
- BPA** - One Byte at Port A. During compression bypass and compression operations, this bit is set during the last data transfer of the record at the Port A interface if only one byte belongs to the current record. During decompression bypass, decompression, and decompression output disabled modes, this bit is set the same time the End of Record at Port A interface bit (STAT1[0] and INTS1[0]) is set if only one byte at the Port A interface is part of the current record. This bit is cleared after reset, when an interrupt reset is written, or when a new compression mode transfer begins.
- EORPA** - End of Record at Port A. During compression bypass and compression operations, this bit is set each time the Record Length (RL) is decremented to zero. During decompression bypass, decompression, and decompression output disabled operations, this bit is set when the last byte of a record is transferred out the Port A interface. This bit is cleared after reset, when an interrupt reset is written, or when a new compression mode transfer begins.

6.20 COMMAND (CMND)

Read/Write
 Hardware Reset Value = 0x0000
 Reset Command = 0xA000

MMODE =

0	1	
0x10	0x11	CMND[7:0]
0x11	0x10	CMND[15:8]

Unspecified opcodes are reserved and may not be written.

CMND[15:0]-Command. This register provides for operation as described in the following table.

CMND[15:0]	ACTION
SELECTION COMMANDS	
0xC100	Select Port A Configuration. The <i>Port A Configuration</i> and <i>Port A Polarity</i> registers are enabled for reads and writes.
0xC200	Select Port B Configuration. The <i>Port B Configuration</i> and <i>Port B Polarity</i> registers are enabled for reads and writes.
TRANSFER COMMANDS (Described in Sections 2.0 and 3.0)	
0x5000-0x500F	Start Compression Bypass. – CMND[3:2] determines the number of pad bytes to expect at the Port A interface. – CMND[1:0] determines the number of pad bytes to insert at the Port B interface.
0x5800-0x580F	Start Compression. – CMND[3:2] determines the number of pad bytes to expect at the Port A interface. – CMND[1:0] determines the number of pad bytes to insert at the Port B interface.
0x6000-0x600F	Start Decompression Bypass. – CMND[3:2] determines the number of pad bytes to expect at the Port B interface. – CMND[1:0] determines the number of pad bytes to insert at the Port A interface.
0x6800-0x680F	Start Decompression. – CMND[3:2] determines the number of pad bytes to expect at the Port B interface. – CMND[1:0] determines the number of pad bytes to insert at the Port A interface.
0x6C00-0x6C0F	Start Decompression Output Disabled. – CMND[3:2] determines the number of pad bytes to expect at the Port B interface. – CMND[1:0] determines the number of pad bytes to insert at the Port A interface.

<i>CMND[15:0]</i>	<i>ACTION</i>
CONTROL COMMANDS	
0x4200	Pause. When a data transfer operation is in progress, any current operation steps are completed and the Port A Interface and Port B Interface data busses are placed into a high impedance state. The Paused Interrupt and Paused Status bits are then set. All data currently being processed by the data transfer operation is preserved.
0x4400-0x440F	Resume. A previously paused data transfer operation resumes processing. The Paused Interrupt and Paused status bits are cleared and the Busy status bit is set. – RESUME[3:2] determines the number of pad bytes at the Port B interface. – RESUME[1:0] determines the number of pad bytes at the Port A interface.
0xA000	Software Reset. The <i>Port A Configuration</i> , <i>Port B Configuration</i> , <i>Identification</i> , <i>Port A Polarity</i> , and <i>Port B Polarity</i> registers are not affected by this command. All other registers are reset, current operations are cancelled, and the history buffer is cleared. Twelve clocks are required to complete the reset operation. Suspend writing to any registers during this time.
0xA400	Reset the history buffer. Only use between compression operations.
MISCELLANEOUS COMMANDS	
0x0000	NOP, no operation is performed.

6.21 RECORD LENGTH (RLL0, RLL1, RLH0, RLH1)

Read/Write
 Hardware Reset Value = 0x00000000
 Reset Command = 0x00000000

Record Length Low

MMODE =

0	1	
0x16	0x17	RLL[7:0]
0x17	0x16	RLL[15:8]

Record Length High

MMODE =

0	1	
0x14	0x15	RLH[7:0]
0x15	0x14	RLH[15:8]

RL[31:0]- The Record Length register indicates the number of Bytes contained in each uncompressed data record for compression bypass and compression operations. This register decrements with each Byte transferred into Port A. When the Record Length reaches zero, the Port Interface bits STAT1[8] and INTS1[8] are set. During decompression bypass operations, the Record Length register indicates the total number of bytes to transfer. Record Length is not used for decompression and decompression output disabled operations.

When processing multiple records (Record Count is greater than one), the Record Length must be greater than 0x22. If Record Length = 0x00000000 when a new transfer or resume command are written, the counter rolls over to 0x10000000. When Record Count is greater than 1, then Record Length must be greater than 0x22. Pad bytes are not counted.

6.22 DATA DISABLED COUNT (DDCL0, DDCL1, DDCH0, DDCH1)

Read/Write
 Hardware Reset Value = 0x00000000
 Reset Command = 0x00000000

Data Disabled Count Low

MMODE =

0	1	
0x1A	0x1B	DDCL[7:0]
0x1B	0x1A	DDCL[15:8]

Data Disabled Count High

MMODE =

0	1	
0x18	0x19	DDCH[7:0]
0x19	0x18	DDCH[15:8]

DDC[31:0]- Data Disabled Count. The *Data Disabled Count* register provides the microprocessor control of the number of records skipped during a Start Decompression Output Disabled operation. If the Data Disabled Count is set to zero during a Start Decompression Output Disabled operation or the DDC is greater than the Record Count during a Start Decompression Output Disabled operation, then the Port A Interface output is disabled for the entire transfer.

6.23 ERROR MASK (EMSK)

Read/Write
 Hardware Reset Value = 0x00
 Reset Command = 0x00

MMODE =

0	1	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x1C	0x1D	reserved	APERRM	BPERRM	reserved	BTCOM	ATCOM	ADCCM	reserved

The *Error Mask* register provides error reporting configuration to the microprocessor. If an unmasked error status bit is active, ANYERR status and ERROR interrupts are set. Errors are masked by setting the appropriate mask bit to one.

APERRM -Port A Interface Parity Error Mask.

BPERRM -Port B Interface Parity Error Mask.

BTCOM - Port B Transfer Count Overflow Error Mask.

ATCOM - Port A Transfer Count Overflow Error Mask.

ADCCM - ALDC Decoder Control Code Error Mask.

6.24 INTERRUPT MASK 0 (IMSK0)

Read/Write
Hardware Reset Value = 0x00
Reset Command = 0x00

MMODE =

0	1	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x1E	0x1F	DONEM	PAUSEDM	ODTM	reserved				ERRORM

The *Interrupt Mask 0* register masks the individual interrupts allowing the user to control which ones may cause the Interrupt signal pin (IREQN) to assert. For example, if DONEM and PAUSEDM are set with ERRORM cleared, only an ERROR interrupt will cause the Interrupt signal pin to assert. Interrupts are masked by setting the appropriate mask bit to one.

DONEM - Done Interrupt Mask.

PAUSEDM - Paused Interrupt Mask.

ODTM - Output Disabled Terminated Mask.

ERRORM - Error Interrupt Mask.

6.25 INTERRUPT MASK 1 (IMSK1)

Read/Write
Hardware Reset Value = 0x00
Reset Command = 0x00

MMODE =

0	1	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
0x1F	0x1E	EORDM	BCMPM	BPBM	EORPBM	reserved		BPAM	EORPAM

The *Interrupt Mask 1* register masks the individual interrupts allowing the user to control which ones may cause the Interrupt signal pin (IREQN) to assert. Interrupts are masked by setting the appropriate mask bit to one.

EORDM - End of Record at Decoder Interrupt Mask.

BCMPM - Port B Interface Compare Interrupt Mask.

BPBM - One Byte at Port B Interrupt Mask.

EORPBM - End of Record at Port B Interrupt Mask.

BPAM - One Byte at Port A Interrupt Mask.

EORPAM - End of Record at Port A Interrupt Mask.

7.0 SIGNAL DESCRIPTIONS

This section contains descriptions for all the pins. Each signal has a type code associated with it. The type codes are described in the following table.

<i>TYPE CODE</i>	<i>DESCRIPTION</i>
I	Input only pin
O	Output only pin
I/O	Input/Output pin

7.1 MICROPROCESSOR INTERFACE

<i>MICROPROCESSOR INTERFACE</i>			
<i>SIGNAL</i>	<i>TYPE</i>	<i>DESCRIPTION</i>	<i>DEFAULT AFTER RESET</i>
MDATA[7:0]	I/O	Microprocessor data bus	Hi-Z
MCIN[0]	I	Microprocessor interface control pin [0]. If MMODE is high this pin is CSN. If MMODE is low this pin is READN.	Input
MCIN[1]	I	Microprocessor interface control pin [1]. If MMODE is high this pin is RWN. If MMODE is low this pin is WRITEN.	Input
WAITN	O	Microprocessor output signal. WAITN is driven during CSN and then goes to tristate with a resistive pullup.	High
ADDR[4:0]	I	Microprocessor Interface address bus, used to select internal registers.	Input
MMODE	I	Microprocessor Interface mode selector pin.	Input
RESETN	I	Hardware reset signal.	Input
IREQN	O	Interrupt request output signal.	High
CLOCK	I	Clock input	Input
+TIE	I	These pins must be tied high in the system.	Input
-TIE	I	These pins must be tied low in the system.	Input

7.2 PORT A INTERFACE

PORT A INTERFACE			
SIGNAL	TYPE	DESCRIPTION	DEFAULT AFTER RESET
ACIN	I	Port A Interface Control Input signal. This signal functions as DREQA. Polarity is programmed by APOL[7].	Input
ACOUT	O	Port A Interface Control Output signal. This signal functions as DACKA. Polarity is programmed by APOL[5].	High
ADBOEN	O	Port A Interface Control signal. Controls direction of Port A transfers. A low level indicates transfers into Port A, and a high level indicates transfers out of Port A.	High
APARITY[1:0]	I/O	When enabled, this pin checks parity on input and generates parity for output for the AD bus. APARITY[0] is used for AD[7:0], and APARITY[1] is used for AD[15:8]. Setting ACNF[15]=1 enables APARITY[0]. Setting ACNF[15]=1 and ACNF[10]=1 enables APARITY[1]. When disabled these pins may be tied high, tied low or not connected.	Hi-Z
ADATA[15:0]	I/O	Port A Interface Data bus. The upper eight bits [15:8] are enabled by setting ACNF[10]=1. When the upper eight bits are disabled they may be tied high, tied low, or not connected.	Hi-Z
AFF_FE	I	FIFO Full/FIFO Empty. In FAS466 mode this signal is asserted when the external devices FIFO reaches the full or empty state, depending on the data transfer direction.	Input
AAF_AE	I	FIFO Almost Full/FIFO Almost Empty. In FAS466 mode this signal is asserted when the external devices programmable FIFO threshold has been reached.	Input

Note: Refer to Section 5.0 Port A and Port B Configuration and Table 2 for configuration of Port A control signals.

7.3 PORT B INTERFACE

PORT B INTERFACE			
SIGNAL	TYPE	DESCRIPTION	DEFAULT AFTER RESET
BCIN	I	Port B Interface Control Input signal. This signal functions as DACKB. Polarity is programmed by BPOL[7].	Input
BCOUT	O	Port B Interface Control Output signal. This signal functions as DREQB. Polarity is programmed by BPOL[5].	High
BDBOEN	I	Port B Interface Control signal. Controls direction of transfers. A low level indicates transfers out of Port B, and a high level indicates transfers into Port B	High
BFF_FE	O	Port B Interface Output signal. Port B FIFO almost full signal. Polarity is programmed by BPOL[1]. Exactly when this flag gets set depends on the threshold bits in the <i>Port B Configuration 0</i> register. In FAS466 DMA mode this signal is FIFO Full or FIFO Empty depending on the direction of the transfer.	High
BAF_AE	O	Port B Interface Output signal. Port B almost empty signal. Polarity is programmed by BPOL[0]. Exactly when this flag gets set depends on the threshold bits in the <i>Port B Configuration 0</i> register. In FAS466 DMA mode this signal is Almost Full or Almost Empty depending on the direction of the transfer.	Low
BPARITY[1:0]	I/O	When enabled, this pin checks parity on input and generates parity for output for the BD bus. BPARITY[0] is used for BD[7:0], and BPARITY[1] is used for BD[15:8]. Setting BCNF[15]=1 enables BPARITY[0]. Setting BCNF[15]=1 and BCNF[10]=1 enables BPARITY[1]. When disabled these pins may be tied high, tied low or not connected.	Hi-Z
BDATA[15:0]	I/O	Port B Interface Data bus. The upper eight bits [15:8] are enabled by setting BCNF[10]=1. When the upper eight bits are disabled they may be tied high, tied low, or not connected.	Hi-Z

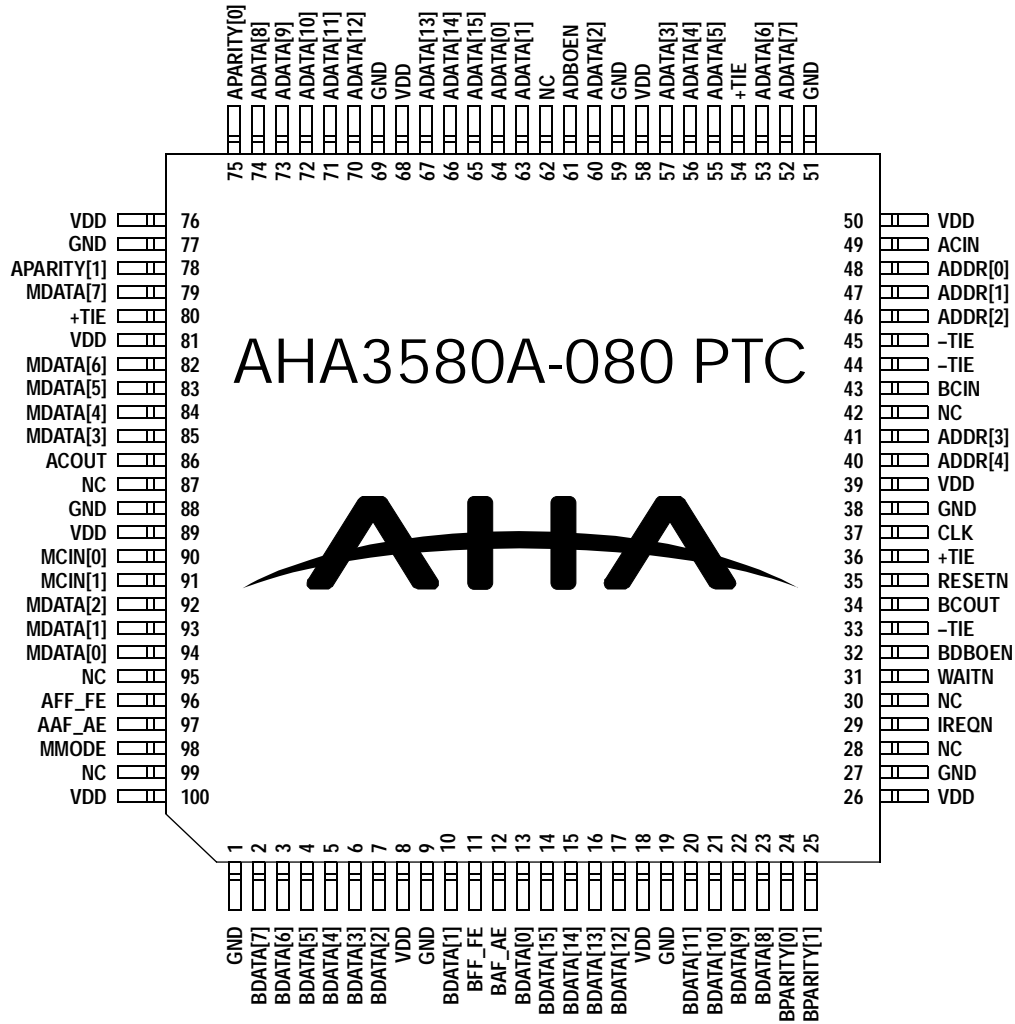
Note: Refer to Section 5.0 Port A and Port B Configuration and Table 3 for configuration of Port B control signals.

8.0 PINOUT

<i>PIN</i>	<i>TQFP</i>	<i>SIGNAL</i>
99		No Connect
100		VDD
1		GND
2		BDATA[7]
3		BDATA[6]
4		BDATA[5]
5		BDATA[4]
6		BDATA[3]
7		BDATA[2]
8		VDD
9		GND
10		BDATA[1]
11		BFF_FE
12		BAF_AE
13		BDATA[0]
14		BDATA[15]
15		BDATA[14]
16		BDATA[13]
17		BDATA[12]
18		VDD
19		GND
20		BDATA[11]
21		BDATA[10]
22		BDATA[9]
23		BDATA[8]
24		BPARITY[0]
25		BPARITY[1]
26		VDD
27		GND
28		No Connect
29		IREQN
30		No Connect
31		WAITN
32		BDBOEN
33		-TIE
34		BCOUT
35		RESETN
36		+TIE
37		CLK
38		GND
39		VDD
40		ADDR[4]
41		ADDR[3]
42		No Connect
43		BCIN
44		-TIE
45		-TIE
46		ADDR[2]
47		ADDR[1]
48		ADDR[0]

<i>PIN</i>	<i>TQFP</i>	<i>SIGNAL</i>
49		ACIN
50		VDD
51		GND
52		ADATA[7]
53		ADATA[6]
54		+TIE
55		ADATA[5]
56		ADATA[4]
57		ADATA[3]
58		VDD
59		GND
60		ADATA[2]
61		ADBOEN
62		No Connect
63		ADATA[1]
64		ADATA[0]
65		ADATA[15]
66		ADATA[14]
67		ADATA[13]
68		VDD
69		GND
70		ADATA[12]
71		ADATA[11]
72		ADATA[10]
73		ADATA[9]
74		ADATA[8]
75		APARITY[0]
76		VDD
77		GND
78		APARITY[1]
79		MDATA[7]
80		+TIE
81		VDD
82		MDATA[6]
83		MDATA[5]
84		MDATA[4]
85		MDATA[3]
86		ACOUT
87		No Connect
88		GND
89		VDD
90		MCIN[0]
91		MCIN[1]
92		MDATA[2]
93		MDATA[1]
94		MDATA[0]
95		No Connect
96		AFF_FE
97		AAF_AE
98		MMODE

Figure 4: TQFP Pinout



NC = No Connect

9.0 ELECTRICAL SPECIFICATIONS

9.1 ABSOLUTE MAXIMUM RATINGS

<i>ABSOLUTE MAXIMUM RATINGS</i>					
<i>SYMBOL</i>	<i>PARAMETER</i>	<i>MIN</i>	<i>MAX</i>	<i>UNITS</i>	<i>NOTES</i>
Vdd	Power supply voltage		3.6	Volts	
Vpin	Voltage applied to any signal pin	0	5.5	Volts	

9.2 RECOMMENDED OPERATING CONDITIONS

<i>RECOMMENDED OPERATING CONDITIONS</i>					
<i>SYMBOL</i>	<i>PARAMETER</i>	<i>MIN</i>	<i>MAX</i>	<i>UNITS</i>	<i>NOTES</i>
Vdd	Power supply voltage	3.0	3.6	Volts	
Ta	Ambient temperature	0	+70	°C	
Tc	Case temperature		+95	°C	

9.3 DC SPECIFICATIONS

<i>DC SPECIFICATIONS</i>							
<i>SYMBOL</i>	<i>PARAMETER</i>	<i>CONDITIONS</i>	<i>MIN</i>	<i>TYP</i>	<i>MAX</i>	<i>UNITS</i>	<i>NOTES</i>
Vil	Input low voltage			0	0.8	Volts	
Vih	Input high voltage		2.0	3.3	5.5	Volts	
Vol	Output low voltage	Iol = 4.0 mAmps	0	0	0.4	Volts	
Voh	Output high voltage	Ioh = -0.4 mAmps	2.4	3.3	Vdd	Volts	
Iil	Input low current	Vin = 0 Volts			5	μAmps	
Iih	Input high current	Vin = Vdd Volts			5	μAmps	
Iozl	Output tristate low current	Vout = 0 Volts			20	μAmps	
Iozh	Output tristate high current	Vout = Vdd Volts			20	μAmps	
IddA	Active Idd current	Vdd = 3.6 Volts			TBD	mAmps	1
Idd	Supply current (static)				TBD	mAmps	
Iol	Low level output current				TBD	mAmps	
Ioh	High level output current				TBD	mAmps	
Cin	Input capacitance				3	pF	
Cout	Output capacitance				6	pF	

Notes:

1) Test Conditions: worst case compression current; 0mA loads.

10.0 TIMING SPECIFICATIONS

Notes:

1) All AC timings are referenced to 1.4 Volts.

Figure 5: Clock Timing

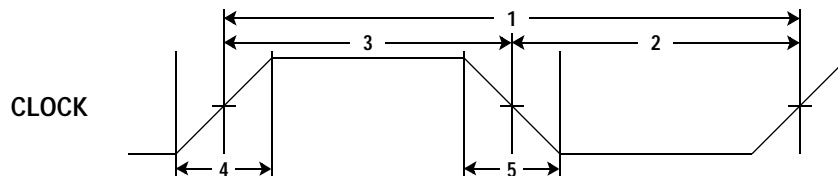


Table 4: Clock Timing

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	CLK period	12.5		ns	1
2	CLK low pulsewidth	5		ns	1
3	CLK high pulsewidth	5		ns	1
4	CLK rise time		3	ns	2
5	CLK fall time		3	ns	2

Notes:

1) All AC Timings are referenced to 1.4 Volts

2) Rise and fall times are between 0.1 V_{dd} and 0.9 V_{dd}.

Figure 6: Reset Timing

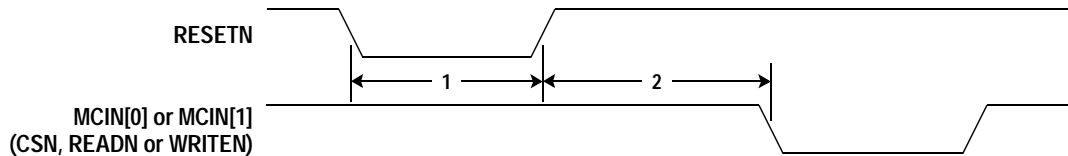


Table 5: Reset Timing

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	RESETN pulsewidth	5		clocks	
2	RESETN delay to CSN, READN or WRITEN	2		clocks	

Figure 7: Processor Read Timing, MMODE = 1

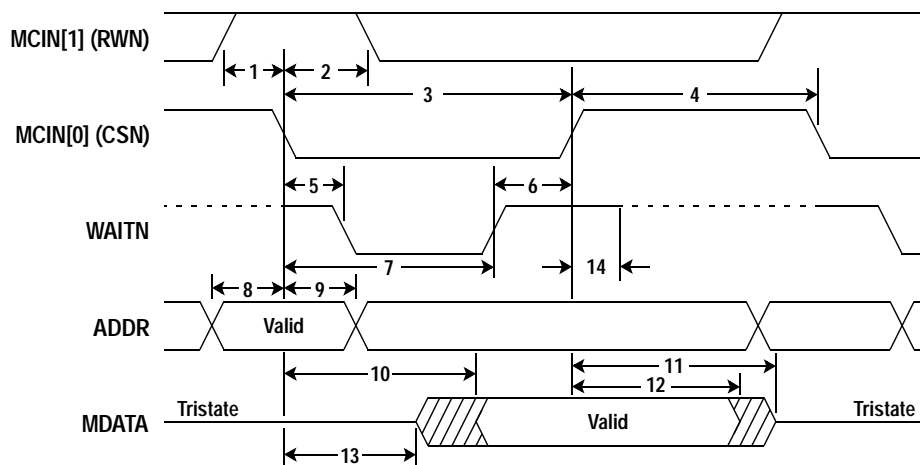


Table 6: Processor Read Timing, MMODE = 1

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	RWN setup to CSN asserted	4		ns	
2	RWN hold from CSN asserted	4		ns	
3	CSN pulsewidth	3		clocks	1
4	Delay from CSN deasserted until next CSN	1 clock+5 ns			
5	CSN asserted to WAITN asserted		18	ns	
6	CSN hold from WAITN deasserted	0		ns	1
7	WAITN deasserted from CSN asserted	2 clocks	3 clocks+18 ns		
8	ADDR setup to CSN asserted	2		ns	2
9	ADDR hold from CSN asserted	6		ns	2
10	MDATA valid from CSN asserted		2 clocks+15 ns		
11	MDATA tristate from CSN deasserted	3	20	ns	
12	MDATA hold from CSN deasserted	3	20	ns	
13	CSN asserted to MDATA driven	1 clock			
14	CSN deasserted to WAITN tristate		10	ns	

Note:

- 1) When WAITN causes CSN to deassert, ignore number 3, otherwise ignore number 6.
- 2) The device latches ADDR on the falling edge of CSN. The user should latch MDATA on the rising edge of CSN.

Figure 8: Processor Write Timing, MMODE = 1

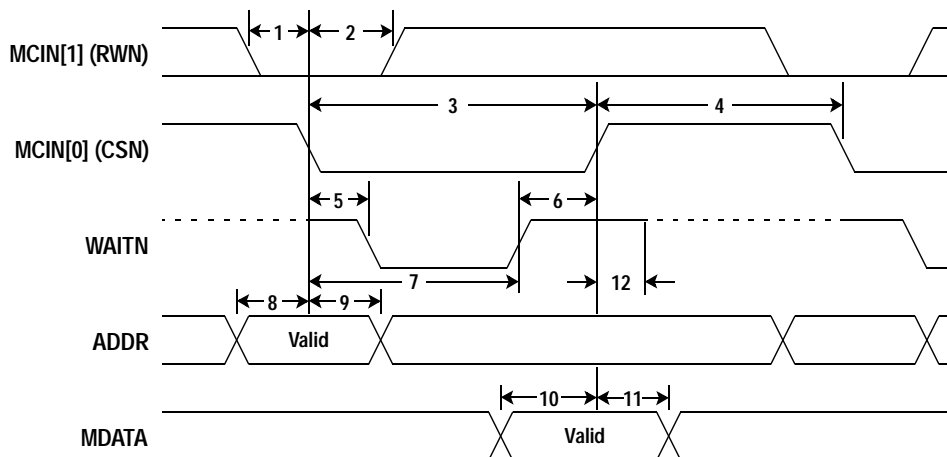


Table 7: Processor Write Timing, MMODE = 1

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	RWN setup to CSN asserted	4		ns	
2	RWN hold from CSN asserted	4		ns	
3	CSN pulsewidth	2		clocks	1
4	Delay from CSN deasserted until next CSN	1 clock+5 ns			2
5	CSN asserted to WAITN asserted		18	ns	
6	CSN hold from WAITN deasserted	0		ns	1
7	WAITN deasserted from CSN asserted	1 clock	2 clocks+18 ns		
8	ADDR setup to CSN asserted	2		ns	3
9	ADDR hold from CSN asserted	6		ns	3
10	MDATA valid before CSN deasserted	4		ns	
11	MDATA hold from CSN deasserted	4		ns	
12	CSN deasserted to WAITN tristate		10	ns	

Notes:

- 1) When WAITN causes CSN to deassert, ignore number 3, otherwise ignore number 6.
- 2) When a read to a register immediately follows a write to that same register or to the command register, CSN must deassert for a minimum of 3 clocks after the write.
- 3) The device latches ADDR on the falling edge of CSN.

Figure 9: Processor Read Timing, MMODE = 0

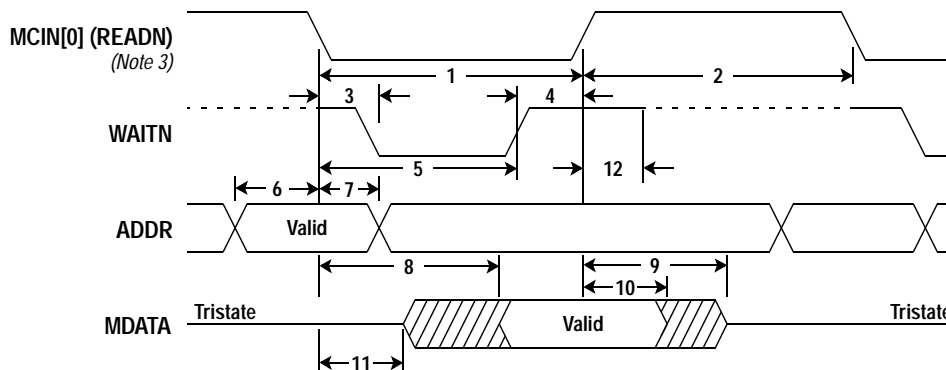


Table 8: Processor Read Timing, MMODE = 0

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	READN pulsewidth	3		clocks	1
2	Delay from READN deasserted until next READN	2		clocks	
3	READN asserted to WAITN asserted		18	ns	
4	READN hold from WAITN deasserted	0		ns	1
5	WAITN deasserted from READN asserted	2 clocks	3 clocks+18 ns		
6	ADDR setup to READN asserted	2		ns	2
7	ADDR hold from READN asserted	6		ns	2
8	MDATA valid from READN asserted		2 clocks+15 ns		
9	MDATA tristate from READN deasserted		20	ns	
10	MDATA hold from READN deasserted	3		ns	
11	MDATA asserted from READN asserted	1 clock			
12	READN deasserted to WAITN tristate		10	ns	

Notes:

- 1) When WAITN causes READN to deassert ignore number 1, otherwise ignore number 4.
- 2) The device latches ADDR on the falling edge of READN. The user should latch MDATA on the rising edge of READN.
- 3) WRITEN must be deasserted during register reads.

Figure 10: Processor Write Timing, MMODE = 0

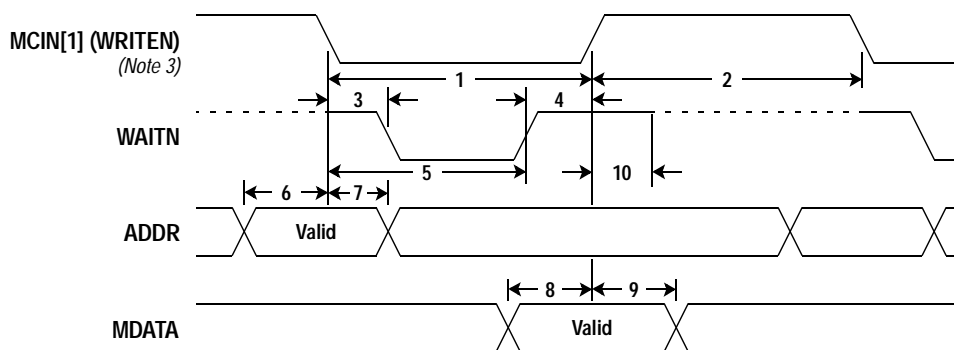


Table 9: Processor Write Timing, MMODE = 0

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	WRITEN pulsewidth	2		clocks	1
2	Delay from WRITEN deasserted until next WRITEN	3		clocks	
3	WRITEN asserted to WAITN asserted		18	ns	
4	WRITEN hold from WAITN deasserted	0		ns	1
5	WAITN deasserted from WRITEN asserted	1 clock	2 clocks+18 ns		
6	ADDR setup to WRITEN asserted	2		ns	2
7	ADDR hold from WRITEN asserted	6		ns	2
8	MDATA valid before WRITEN deasserted	4		ns	
9	MDATA hold from WRITEN deasserted	4		ns	
10	WRITEN deasserted to WAITN tristate		10	ns	

Notes:

- 1) When WAITN causes WRITEN to deassert ignore number 1, otherwise ignore number 4.
- 2) The device latches ADDR on the falling edge of WRITEN.
- 3) READN must be deasserted during register writes.

Figure 11: Port A, FAS466 DMA Slave Mode Read

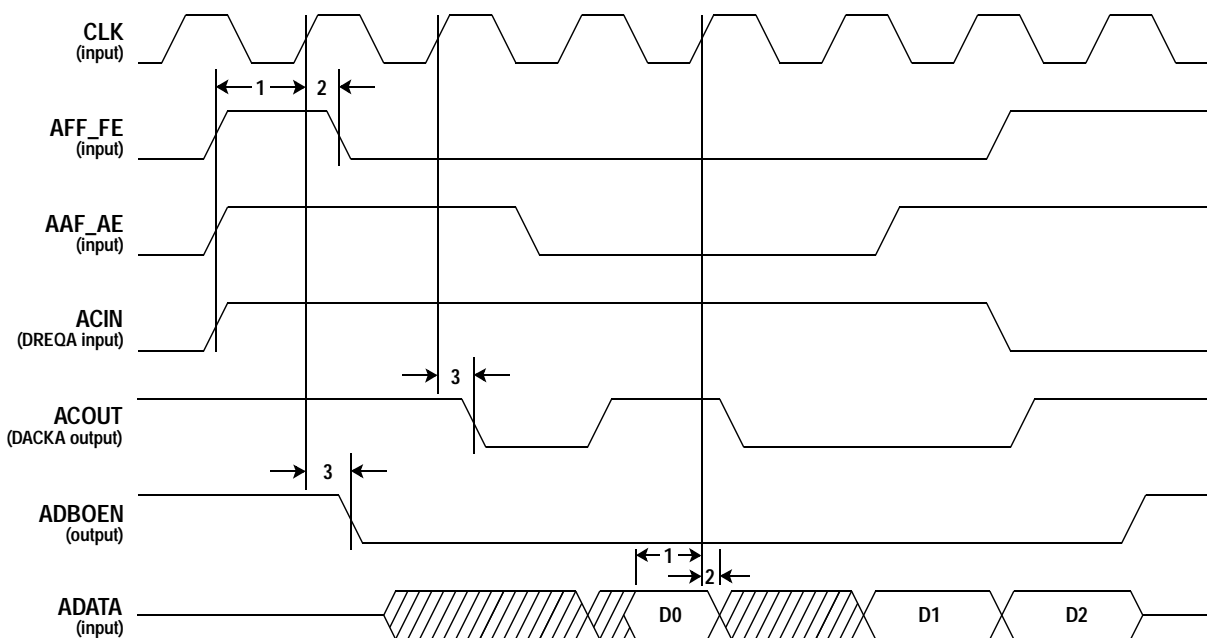


Table 10: Port A, FAS466 DMA Slave Mode Read

<i>NUMBER</i>	<i>PARAMETER</i>	<i>MIN</i>	<i>MAX</i>	<i>UNITS</i>	<i>NOTES</i>
1	Input setup to CLK rising edge	3		ns	
2	Input hold from CLK rising edge	2		ns	
3	Output delay from CLK rising	2	9	ns	

Figure 12: Port A, FAS466 DMA Slave Mode Write

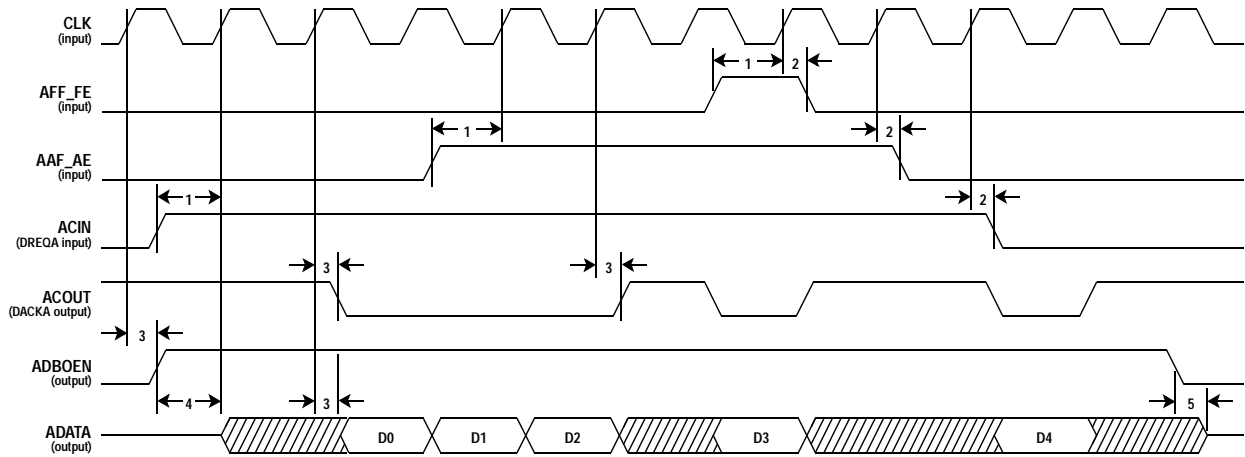


Table 11: Port A, FAS466 DMA Slave Mode Write

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	Input signals: AFF_FE, AAF_AE and DREQA setup to CLK rising edge	3		ns	
2	Input Signals: AFF_FE, AAF_AE and DREQA hold from CLK rising edge	2		ns	
3	Outputs valid from CLK rising	2	9	ns	
4	ADBOEN inactive to ADATA driven	2		ns	
5	ADATA tristate from ADBOEN active	0	5	ns	

Figure 13: Port B, FAS466 DMA Master Mode Read

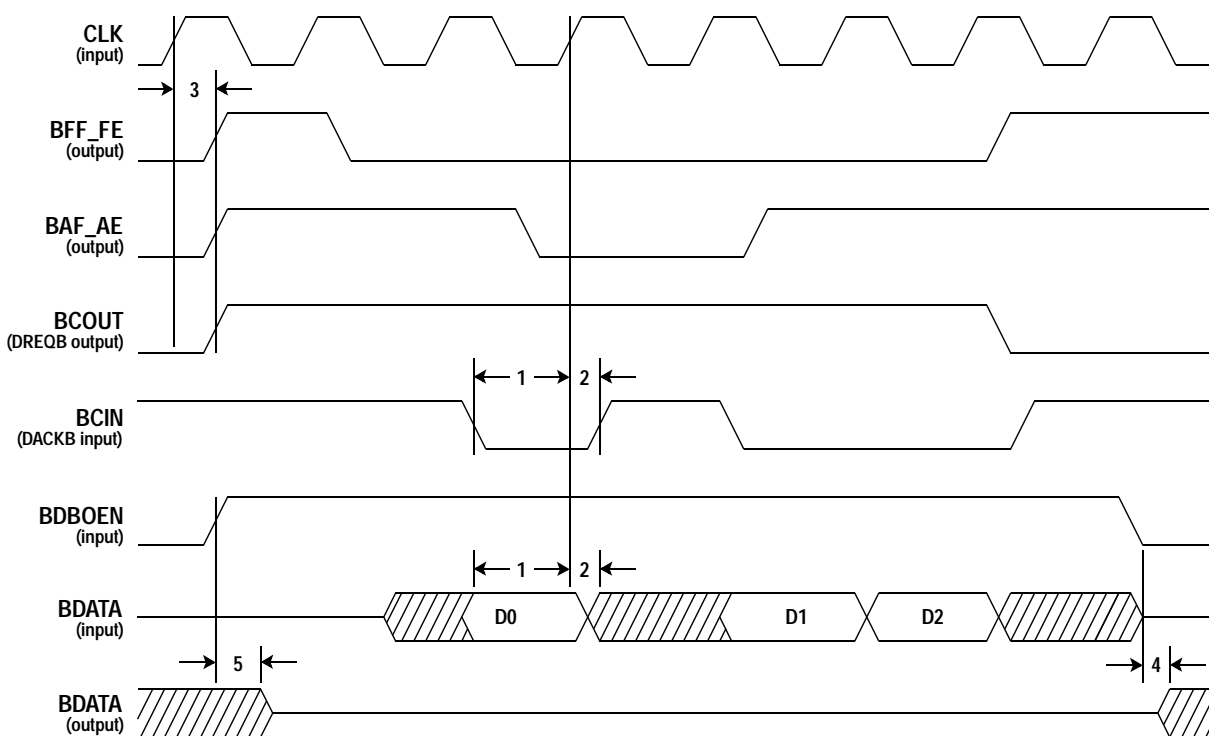


Table 12: Port B, FAS466 DMA Master Mode Read

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	DACKB and BDATA setup to CLK rising edge	3		ns	
2	DACKB and BDATA hold from CLK rising edge	2		ns	
3	Output signals delay from CLK rising	2	9	ns	
4	BDBOEN inactive to BDATA (output) tristated		5	ns	1
5	BDBOEN active to BDATA (output) driven	0		ns	1

Notes:

1) The device controlling BDBOEN must prevent floating and contention on BDATA.

Figure 14: Port B, FAS466 DMA Master Mode Write

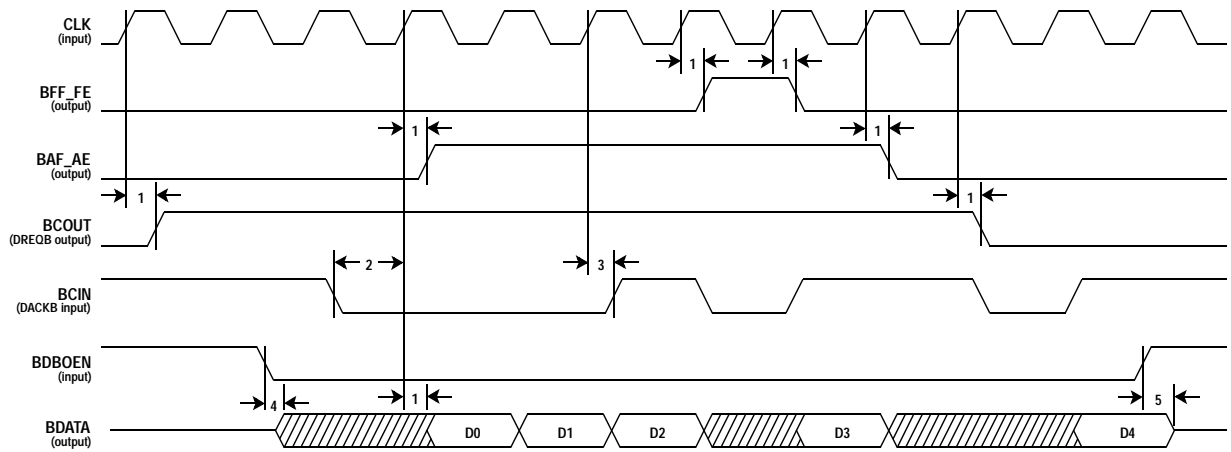


Table 13: Port B, FAS466 DMA Master Mode Write

<i>NUMBER</i>	<i>PARAMETER</i>	<i>MIN</i>	<i>MAX</i>	<i>UNITS</i>	<i>NOTES</i>
1	Output valid from CLK rising edge	2	9	ns	
2	Input setup to CLK rising edge	3		ns	
3	Input hold from CLK rising edge	2		ns	
4	BDATA driven from active edge of BDBOEN	0		ns	
5	BDATA tristate from inactive edge of BDBOEN		5	ns	

Figure 15: Port A, Initiator Synchronous DMA Mode In Timing

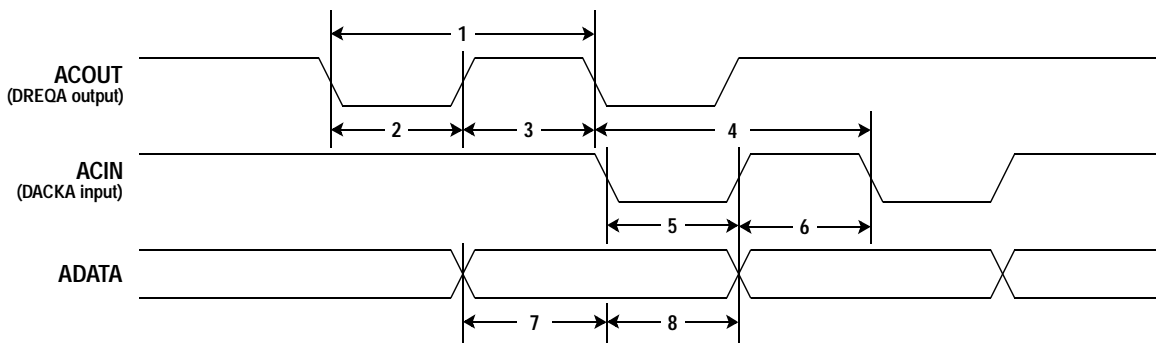


Table 14: Port A, Initiator Synchronous DMA Mode In Timing

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	DREQA cycle time	2		clocks	
2	DREQA asserted width	1		clocks	
3	DREQA negated width	1		clocks	
4	DACKA cycle time	2		clocks	
5	DACKA asserted width	1		clocks	
6	DACKA negated width	1		clocks	
7	ADATA valid to DACKA asserted	1		clocks	
8	DACKA asserted to data invalid	1		clocks	

Figure 16: Port A, Initiator Synchronous DMA Mode Out Timing

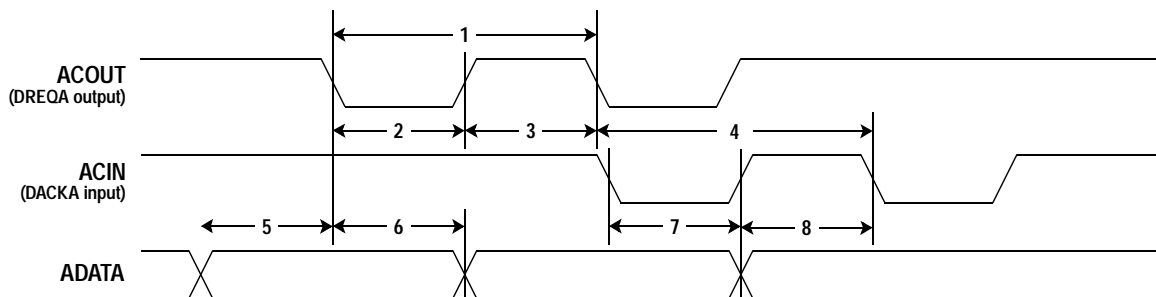


Table 15: Port A, Initiator Synchronous DMA Mode Out Timing

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	DREQA cycle time	2		clocks	
2	DREQA asserted width	1		clocks	
3	DREQA negated width	1		clocks	
4	DACKA cycle time	2		clocks	
5	ADATA valid to DREQA asserted	5		ns	
6	DREQA asserted to ADATA invalid	1		clocks	
7	DACKA asserted width	1		clocks	
8	DACKA negated width	1		clocks	

Figure 17: Port B, Initiator Synchronous DMA Mode Out Timing

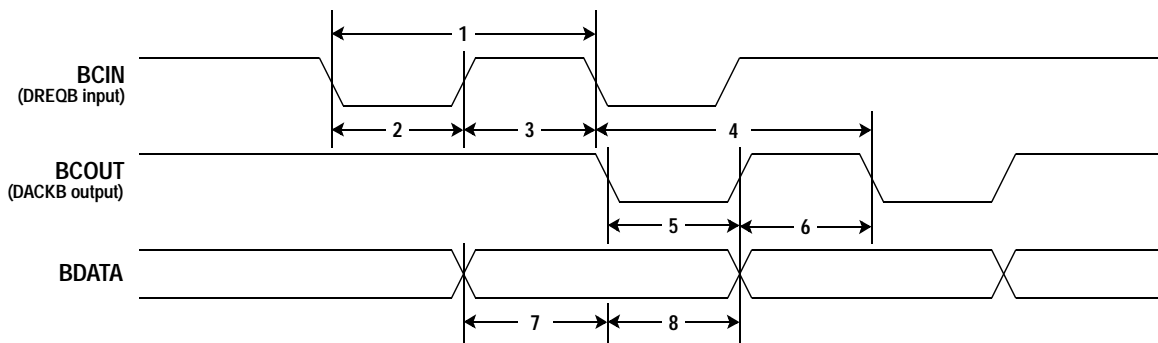


Table 16: Port B, Initiator Synchronous DMA Mode Out Timing

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	DREQB cycle time	2		clocks	
2	DREQB asserted width	1		clocks	
3	DREQB negated width	1		clocks	
4	DACKB cycle time	2		clocks	
5	DACKB asserted width	1		clocks	
6	DACKB negated width	1		clocks	
7	BDATA valid to DACKB asserted	1		clocks	
8	DACKB asserted to data invalid	1		clocks	

Figure 18: Port B, Initiator Synchronous DMA Mode In Timing

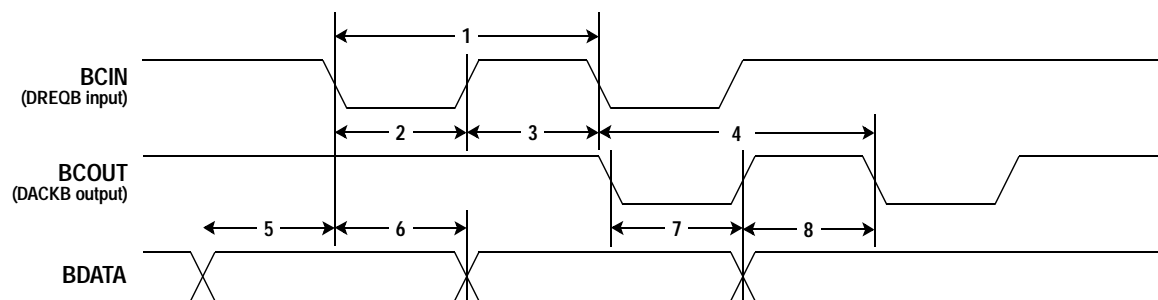


Table 17: Port B, Initiator Synchronous DMA Mode In Timing

NUMBER	PARAMETER	MIN	MAX	UNITS	NOTES
1	DREQB cycle time	2		clocks	
2	DREQB asserted width	1		clocks	
3	DREQB negated width	1		clocks	
4	DACKB cycle time	2		clocks	
5	BDATA valid to DREQB asserted	5		ns	
6	DREQB asserted to BDATA invalid	1		clocks	
7	DACKB asserted width	1		clocks	
8	DACKB negated width	1		clocks	

11.0 PACKAGING

Figure 19: AHA3580 TQFP Package Specifications

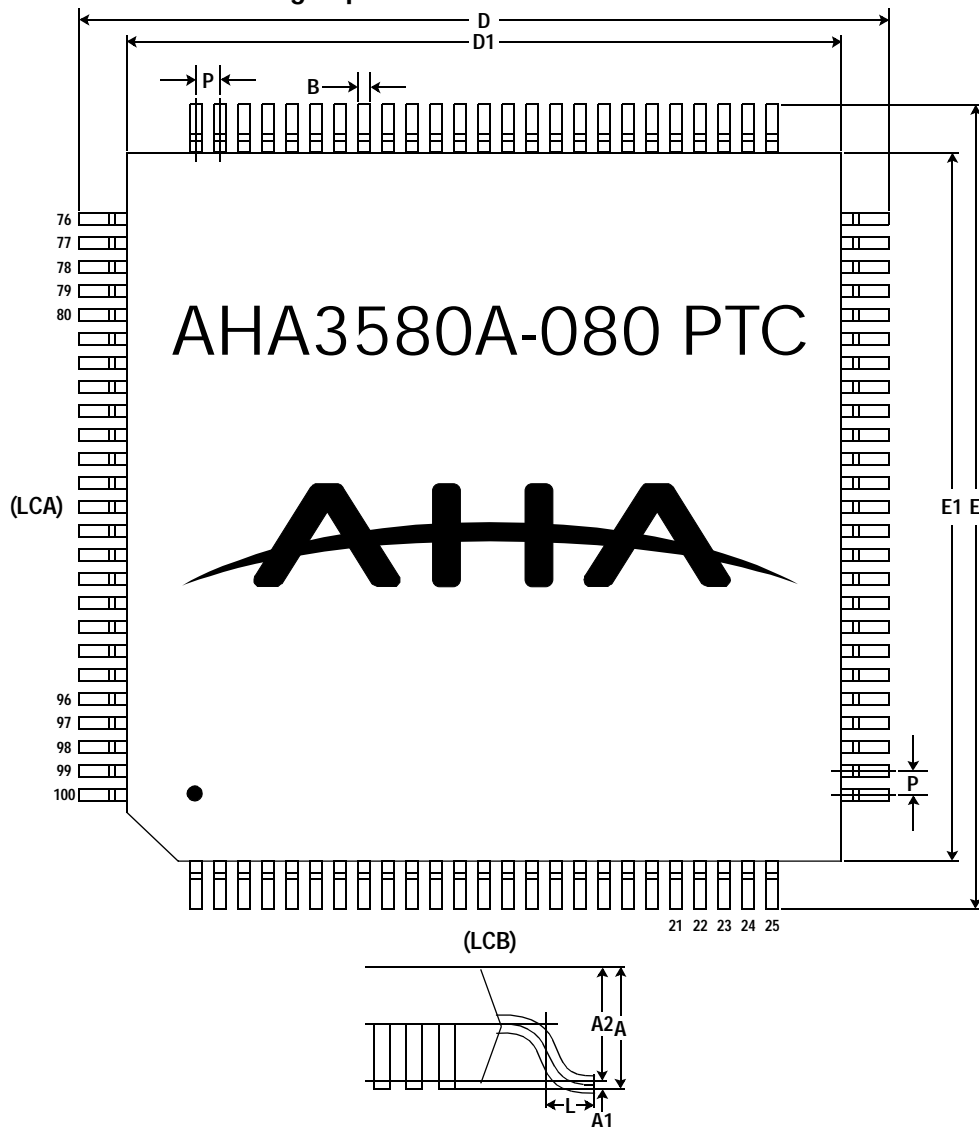


Table 18: TQFP (Thin Quad Flat Pack) 14 × 14 mm Package Dimensions
(All dimensions are in mm)

SYMBOL	NUMBER OF PIN AND SPECIFICATION DIMENSION		
	100		
	SB		
	MIN	NOM	MAX
(LCA)	25		
(LCB)	25		
A			1.7
A1	0.05		0.15
A2	1.35	1.4	1.45
D	15.80	16.0	16.20
D1	13.90	14.0	14.10
E	15.80	16.0	16.20
E1	13.90	14.0	14.10
L	0.50	0.60	0.75
P		0.50	
B	0.17	0.22	0.27

JEDEC Outline MO-136

12.0 ORDERING INFORMATION

12.1 AVAILABLE PARTS

<i>PART NUMBER</i>	<i>DESCRIPTION</i>
AHA3580A-080 PTC	80 MBytes/sec ALDC Data Compression Coprocessor IC, 3.3V I/O, 2.5V Core

12.2 PART NUMBERING

<i>AHA</i>	<i>3580</i>	<i>A-</i>	<i>080</i>	<i>P</i>	<i>T</i>	<i>C</i>
Manufacturer	Device Number	Revision Level	Speed Designation	Package Material	Package Type	Test Specification

Device Number:

3580

Revision Letter:

A

Package Material Codes:

P Plastic

Package Type Codes:

T T - Thin

Test Specifications:

C Commercial 0°C to +70°C

13.0 AHA RELATED TECHNICAL PUBLICATIONS

<i>DOCUMENT #</i>	<i>DESCRIPTION</i>
PB3580	AHA Product Brief – AHA3580 80 MBytes/sec ALDC Data Compression Coprocessor IC
PB3540	AHA Product Brief – AHA3540 40 MBytes/sec ALDC Data Compression Coprocessor IC
PS3540	AHA Product Specification – AHA3540 40 MBytes/sec ALDC Data Compression Coprocessor IC
PS3520	AHA Product Specification – AHA3520 20 MBytes/sec ALDC Data Compression Coprocessor IC
ABDC17	AHA Application Brief – Differences between AHA3540 and IBM ALDC1-20S-LP Devices
ANDC18	AHA Application Note – Differences between AHA and IBM Devices