IP-AD8100-D

Octal Differential Input 100 kHz Simultaneous
Burst A/D in 16 bits
IndustryPack Module

REFERENCE MANUAL

804-16-000-4000
Version 1.7

Oct. 2003

ALPHI TECHNOLOGY CORPORATION

6202 S. Maple Avenue #120
Tempe, AZ 85283 USA
Tel: (480) 838-2428
Fax: (480) 838-4477

NOTICE

The information in this document has been carefully checked and is believed to be entirely reliable. While all reasonable efforts to ensure accuracy have been taken in the preparation of this manual, ALPHI TECHNOLOGY assumes no responsibility resulting from omissions or errors in this manual, or from the use of information contain herein.

ALPHI TECHNOLOGY reserves the right to make any changes, without notice, to this or any of ALPHI TECHNOLOGY's products to improve reliability, performance, function or design.

ALPHI TECHNOLOGY does not assume any liability arising out of the application or use of any product or circuit described herein; nor does ALPHI TECHNOLOGY convey any license under its patent rights or the rights of others.

ALPHI TECHNOLOGY CORPORATION

All Rights Reserved

This document shall not be duplicated, nor its contents used
for any purpose, unless express permission has been granted in advance.

TABLE OF CONTENTS

ADENDUM TO REVISION 1.5 : REVISED I/O MAP AND PROGRAMMING PROCEDURES TO REFLECT ACCESS TO IP-AD8100 IN BYTE INSTEAD OF 32 BIT DSP ACCESS.

ADENDUM TO REVISION 1.6 : REVISED ALL PROCEDURES TO REFLECT THE INCREASED SIZE OF THE FIFO, FROM 16K(3FFF) TO 64K(FFFF).

Copyright ? 2000, ALPHI Technology Corporation

1. GENERAL DESCRIPTION

1.1 INTRODUCTION

The **ATC-AD8100K** is a 16 bit single width IP module designed for high speed burst A/D data acquisition in 16 bits. The primary features of the **ATC-AD8100K** are:

? Eight channels of simultaneous 16 bit A/D acquisition, operating at a maximum rate of 100 kHz. Inputs are +/- 10V Single Ended.
? Up to 16K of samples stored divided among active A/D converters in onboard FIFO memory.
? Full support for pre-trigger and post trigger acquisition.
? Sampling clock selected from one of the following sources: Internal divider (IPCLK / N), IPSTROBE, and external clock.
? Trigger event selected from one of the following sources: Write to IP register, IPSTROBE, and external trigger.
? Continuous streaming acquisition is also possible at lower throughput.
? IP bus operates at 32 MHz.

1.2 IDSPACE

Up to 32 bytes of registered data provide information about the module to the User. The lower address contains data related to the type of module, revision, etc. .. Only ODD address are valid.

ID space address	Description	Value
$01	ASCII “I”	$49
$03	ASCII “P”	$50
$05	ASCII “A”	$41
$07	ASCII “H”	$48
$09	Manufacturer identification	$11
$0B	Module type	$18
$0D	Revision module	$0A
$0F	Reserved	
$11	Driver ID,low byte	

$13	Driver ID,high byte	
$15	Number of bytes used	$0A
$17	CRC	
$19-$3F	User space	

Table 1-1 IDSEL0 SPACE byte content

1.3 FUNCTIONAL DESCRIPTION

A data flow block diagram of the **ATC-AD8100** is presented in Figure 1-1.

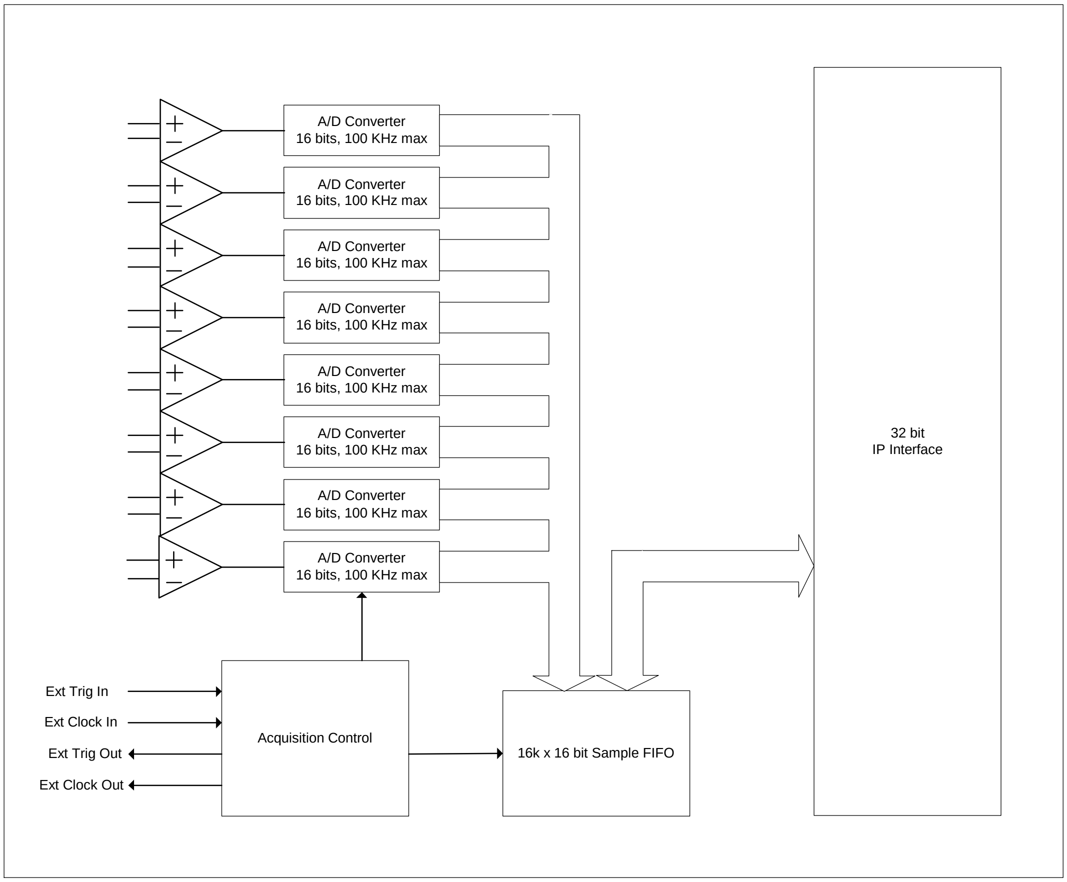


Figure 1.1: Data Flow Block Diagram

A/D samples are acquired and stored into the 16 bit FIFO when acquisition has been started. Once the pre-trigger data has been acquired (FIFO /PAE goes from low to high), triggering becomes active. Further acquisition before the trigger is seen will result in the earliest data being discarded at the same time new data is saved, thus maintaining the most recent data in the FIFO.

Once the trigger event is seen, no more data is discarded from the FIFO, and acquisition proceeds until the FIFO is full (FIFO /FF becomes low). At this point, the acquisition stops, and the HOST can read the data from the IP. Interrupts can be sent to the HOST at Event and at acquisition finished.

If desired, such as for a continuous acquisition, data can be read from the FIFO while acquisition is in progress, for continuous streaming. Unfortunately, it will not be possible to maintain a high sampling rate in this scenario due to the time constraints of the IP interface.

1.4 SOFTWARE SUPPORT

The **ATC-AD8100K** is supported under *Windows NT* / 2000 by two sample programs, which are supplied with the IP in the board support package. Both examples are designed to work with an IPM type carrier from ALPHI, such as the PCI-4IPM.

One sample program, called SnapShot, fully exercises the IP module in pre- and post-trigger modes, and displays the data to the screen. Data can be stored to a file and can be reloaded in the program at a later time.

The second program, called DrawIpAdc, operates the IP in continuous mode, and displays the data to the screen.

Full source to both the DSP code and the applications are provided.

2. THEORY OF OPERATION

2.1 ANALOG INPUTS

Eight Single Ended analog inputs are provided. By default, the inputs are high impedance +/- 10 volt.

A/D Converters

There are eight A/D converters. The A/D converters operate continuously at the selected sampling rate. Results are either stored in the FIFO or discarded when appropriate. Only selected channels are saved into the FIFO, so that depth per channel is proportional to number of channels saved.

2.1.1 BURST MODE

The customer may desire to think of the IP module as similar to a Digital Storage Oscilloscope (DSO). A DSO can store and display several waveforms and can record signals prior to the trigger. The trigger point is completely configurable by reprogramming the FIFO's programmable empty flag. This is easily done, and the software DSP examples demonstrate how.

Setting up and operating this type of acquisition is easy. First, the *Clock* and *Trigger Sources* need to be configured. If internal clocking is used, then the divisor needs to be programmed as well. In this mode, *Arming Source* should be set to PAE. The FIFO should be reset, and the desired PAE point programmed into the FIFO.

When the application is ready to capture data, it should write to the **Start Acquisition** address (value is not important). This action will cause data to be stored into the FIFO (previously conversions were dropped as they finished). Conversions will be saved in the FIFO until the programmed number of samples raises PAE. At this point, the trigger is armed, and further writes to the FIFO result in corresponding reads from the FIFO, discarding the earliest data.

Once the trigger event is seen (external signal, IPSTROBE, or software write to **Trigger Event**), then the earliest data is no longer discarded, and the FIFO fills to full.

Sampling continues until the FIFO FF flag is low, stopping the acquisition. If an early end to acquisition is desired, a write to **Stop Acquisition** will accomplish this.

Interrupts may be generated by PAE going high (indicating that triggering is now active) and by FF going low (indicating that the acquisition is now finished).

2.1.2 CONTINUOUS MODE

If alternatively, continuous acquisition is desired, then the following procedure is used. Remember that in a real world scenario, it is not possible to read the data at the maximum rate that the IP is capable of. There is no way to predict the exact performance as it depends on the carrier board and the application.

Setting up and operating this type of acquisition is also easy. First, the *Clock Source* need to be configured. If internal clocking is used, then the divisor needs to be programmed as well. For *Trigger Source*, select Software Strobe Only. Set *Arming*

Source to Never Discard. The FIFO should be reset, and the desired PAE point programmed into the FIFO.

Interrupts can be generated by PAE going high (indicating that the programmed number of samples is available to read from the FIFO).

When it is desired to start acquisition, write to **Start Acquisition** and then **Trigger Event**. When it is desired to stop acquisition, write to **Stop Acquisition**.

2.1.3 INTERRUPTS

Interrupts can be generated from two events:

? When the acquisition has stopped because FF has gone low.
? When the specified pretrigger is reached because PAE has gone high.
? When the specified number of samples are in the FIFO because PAE has gone high. (Continuous mode)

3. INTERFACE TO THE IP CARRIER

The IP carrier controls this IP via a set of registers in IP IOSPACE. There is no memory on this IP.

3.1 REGISTERS

The I/O map access is displayed in 8 bit wide accesses except for the FIFO which is 16 bits wide, so all registers are at ODD addresses except for the FIFO register. If using word access convert I/O map to even addresses.

Copyright ? 2000, ALPHI Technology Corporation

IP-8100 Address	Read/ Write	Access	Wait State @ 32MHz	Register
0x01	R/W	Byte	2	Internal Clock Divisor 0
0x03	R/W	Byte	2	Internal Clock Divisor 1
0x05	R/W	Byte	2	Internal Clock Divisor 2
0x07	R/W	Byte	2	Channel Mask
0x09	W/only	Byte	2	Trigger Event
0x0B				Unused
0x0D	W/only	Byte	2	Stop Acquisition
0x0F	R/W	Byte	2	Acquisition Control Register
0x11	R/W	Byte	2	Source Intreq1
0x13	W	Byte	2	Clear Intreq0 latch
0x15	R/W	Byte		Read/Write trigger register
0x17	R/W	Byte	2	FIFO Control / Status
0x19	W/only	Byte	2	Reset FIFO
0x1A	R/W	Word	2	Hardware FIFO
0x27	WS	Byte	2	Start Acquisition
0x29	R/W	Byte	2	Interrupt Vector register

Table 3.2 IO Registers

3.1.1 INTERNAL CLOCK DIVISOR [0, 1, 2]

These three 8 bit registers combine to form one 24 bit register which serves as a divisor on the IP clock when internal sampling clock is selected. Note this IP only runs at 32Mhz.

$$SamplingRate \; ? \; \frac{IPClockFreq}{N \; ? \; 1}$$

3.1.2 CHANNEL MASK

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CH7	CH6	CH5	CH5	CH3	CH2	CH1	CH0

This 8 bit register allows for enabling or disabling the appropriate channels from the FIFO. If the appropriate bit is set (1), then the channel is not stored into the FIFO. If the bit is cleared (0), then the channel is stored in the FIFO.

Channels are stored in order, starting with channel 0.

3.1.3 TRIGGER EVENT

A write to this register will cause the trigger to occur, regardless of the setting in the **Acquisition Control Register**.

3.1.4 STOP ACQUISITION

A write to this register will stop acquisition immediately. The FIFO will probably not have full capacity stored, but a full set of samples from all enabled channels will have been stored.

3.1.5 ACQUISITION CONTROL REGISTER

Bits 7-6	Bits 5-4	Bits 3-2	Bits 1-0
Sampling Clock Source	Trigger Source	Stop Acquisition Source	Trigger Enable Source

This register allows for the configuration of the acquisition state machine.

Sampling Clock Source determines where the A/D converters get their start conversion source from.

Sampling Clock Source	Meaning
00	Internal Sampling Clock (IP Clock divided by Divisor)
01	IPSTROBE
10	External Clock from connector

Trigger Source determines what event causes the acquisition to switch from pre-triggered to post-triggered. A write to **Software Strobe** will always cause a trigger, provided the pre-trigger data has been acquired.

Trigger Source	Meaning
00	**Software Strobe** only
01	IPSTROBE
10	External Trigger from connector

Stop Acquisition Source determines the cause of ending the acquisition. It should probably be set to FF.

Stop Acquisition Source	Meaning
00	FIFO FF Flag
01	FIFO PAF Flag

Copyright ? 2000, ALPHI Technology Corporation

Trigger Enable Source determines how the state machine decides that the trigger should be enabled. For burst acquisition, set to Enable After PAE Satisfied. For continuous acquisition, set to Always Enabled.

Trigger Enable Source	Meaning
00	Enable After PAE Satisfied
01	Always Enabled

3.1.6 SOURCE INTREQ1

This register allows to enable one signal source that will generate an interrupt Intereq1 .

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
Not Used	Not Used	END_ ACQ_ OUT	FIFO EF	FIFO PAE	FIFO HF	FIFO PAF	FIFO FF

FIFO FF

This bit, when low (0), indicates that there are 16384 samples stored in the hardware FIFO. It is usually an indication that the acquisition is completed.

FIFO PAF

This bit, when low (0), indicates that there is a programmable number of samples stored in the hardware FIFO. It has no meaning for the acquisition modes discussed in this manual.

FIFO HF

This bit, when low (0), indicates that there is a 16384/2 samples number of samples stored in the hardware FIFO. It has no meaning for the acquisition modes discussed in this manual.

FIFO PAE

This bit, when low (0), indicates that there is less than a programmable number of samples stored in the hardware FIFO. When this bit goes high, it indicates that the pre-trigger sampling requirement has been met, and that it is now possible to trigger the acquisition. In continuous modes, this bit being high indicates that there are N samples ready to be read, where N-1 is the number programmed in the FIFO.

FIFO EF

This bit, when low (0), indicates that there are no samples stored in the hardware FIFO.

END_ACQ_OUT

When the acquisition cycle is ended a raising signal will generate an interrupt.

Note:

The interrupt Intreq1 is latched. To reset the latch, a read of the FIFO Control/Status register is needed.

3.1.7 READ / WRITE TRIGGER

This register allows to control the trigger when an acquisition to the hardware FIFO is being made.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
INT1 RESET RDFIFO	Event Enable	Master	INT1 RESET EF	Disable/ Enable Readback	ACQ ONCE	IDLE Enable	CONV Source

CONV SOURCE

This bit, when high (1), enables the event input to be the source for start acquisition.

IDLE ENABLE

This bit is set to 1 and is internally used. Not configurable.

ACQ ONCE

This bit, when high (1), is to ReArm the acquisition upon FIFO empty (EF = 0).

DISABLE/ENABLE READBACK

This bit when high (1), disables the automatic readback of the FIFO thus allowing to store data that is written by hand. Once is written into the FIFO set bit back to (0) low to allow the readback of the data written.

INT1 RESET EF

This bit, when high (1), Is to remove INTREQ-1 when empty FIFO (EF = low).

MASTER

This bit, when high (1), will make the IP module the source for the sample clock, the signal will appear on IP-STROBE (P1 pin 46) as an output. The MASTER bit allows the synchronization of multiple IP’s

EVENT ENABLE

This bit, when high (1) enables the external event input on P2 pin 24. The event pin can be used to externally start conversion.

INT 1 RESET RDFIFO

This bit , when high (1) Is to remove INTREQ-1 when RDFIFO goes low.

Copyright ? 2000, ALPHI Technology Corporation

3.1.8 FIFO CONTROL / STATUS

This register allows for querying the current state of the hardware FIFO flags and a means to program the FIFO.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
FIFO FF	FIFO PAF	FIFO PAE	FIFO EF	N/A	N/A	N/A	FIFO REG

FIFO FF

This bit, when low (0), indicates that there are 16384 samples stored in the hardware FIFO. It is usually an indication that the acquisition is completed.

FIFO PAF

This bit, when low (0), indicates that there is a programmable number of samples stored in the hardware FIFO. It has no meaning for the acquisition modes discussed in this manual.

FIFO PAE

This bit, when low (0), indicates that there is less than a programmable number of samples stored in the hardware FIFO. When this bit goes high, it indicates that the pre-trigger sampling requirement has been met, and that it is now possible to trigger the acquisition. In continuous modes, this bit being high indicates that there are N samples ready to be read, where N-1 is the number programmed in the FIFO.

FIFO EF

This bit, when low (0), indicates that there are no samples stored in the hardware FIFO.

FIFO REG

This bit, when cleared to low (0), causes writes to **Hardware FIFO** to be placed into the FIFO for testing purposes. Reads of **Hardware FIFO** will access the contents of the FIFO. When the bit is set high (1), writes to and reads of the FIFO will access the FIFO programming registers.

3.1.9 RESET THE HARDWARE FIFO

A write to this location will reset the hardware FIFO to empty. It is required to initialize the FIFO before any access is made.

3.1.10 Hardware FIFO

If the **FIFO REG** bit is clear (0), then a read of this location will respond with the oldest pair of samples in the FIFO. If the **FIFO REG** bit is set (1), then a read of this location will read the internal configuration registers of the FIFO.

Data is stored as 16 bit WORDs in 2's complement format. Only enabled channels will be found in the FIFO. See **Channel Mask** for details.

If the **FIFO REG** bit is clear (0), then a write to this location will directly add the value to the FIFO for testing purposes. If the **FIFO REG** bit is set (1), then a write to this location will program the internal configuration registers of the FIFO.

The FIFO is actually an IDT 72265 or equivalent.

3.1.11 Start Acquisition

A write to this register will start acquisition immediately. Data will be stored until the pre-trigger condition is met (PAE goes high). At this time, old data will be discarded from the FIFO as new data is written into the FIFO.

If continuous acquisition is desired, then a write here must be followed with a write to **Trigger Acquisition**.

3.1.12 Interrupt Vector Register

An 8 bit Interrupt Vector register is available for future applications

4. HARDWARE DETAILS

4.1 RESET SIGNALS

The **ATC-AD8100K** is reset when the IP carrier issues a reset.

4.2 CONNECTORS AND JUMPERS

4.3 CONNECTORS

The connector placement is depicted below. There are no configuration jumpers.

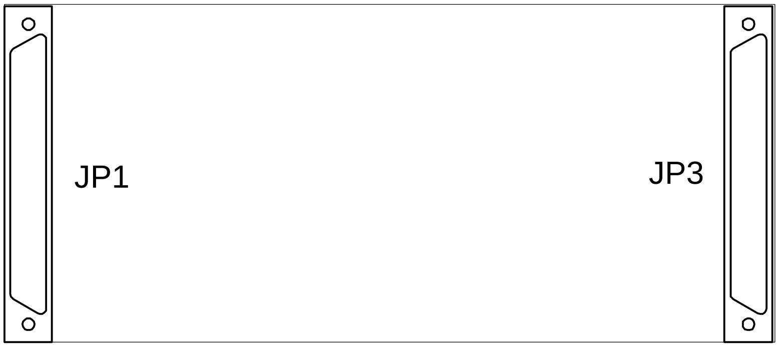


Figure 4.1: Connector Locations

4.3.1 CONNECTOR DESCRIPTIONS

IP External I/O Connector (JP3)

A 50 pin subminiature D shelled connector is used to route the analog signals to the IP. The IP carrier then takes these signals and presents them for customer use. See the documentation for the IP carrier for more details.

The signals are routed as follows.

Pin	Connection	Pin	Connection
1	IN01+	26	IN01-
2	AGND	27	AGND
3	IN02+	28	IN02-
4	AGND	29	AGND
5	IN03+	30	IN03-
6	AGND	31	AGND
7	IN04+	32	IN04-
8	AGND	33	AGND
9	IN05+	34	IN05-
10	AGND	35	AGND
11	IN06+	36	IN06-
12	AGND	37	AGND
13	IN07+	38	IN07-
14	AGND	39	AGND
15	IN08+	40	IN08-
16	AGND	41	AGND
17		42	
18		43	
19		44	
20		45	CLOCK OUT
21		46	DGND
22	DGND	47	EXT CLOCK IN
23	EVENT	48	DGND
24	DGND	49	EXT TRIG IN
25	EXT TRIG OUT	50	DGND

Table 4.3: IP External I/O Connector (JP3)

Copyright ? 2000, ALPHI Technology Corporation

5. QUICK REFERENCE FOR PROGRAMMING AD8100 FIFO

This procedure will explain how to program the internal divider clock, set the FIFO pointer and acquire the data injected on the I/O channels.

Each address starts from the base of the I/O space different carriers might vary.

Follow the procedure in the order it's written do not skip any steps.

Program the Internal sample clock

? At 8Mhz to set the Internal Clock at 200uS do the following.

? $00 - Internal clock divisor 0, write $3F.

? $03 - Internal clock divisor 1, write $6.

? At 32Mhz to set the Internal Clock at 200uS do the following.

? $01 - Internal clock divisor 0, write $FF.

? $03 - Internal clock divisor 1, write $18.

Clear Intreq0 latch

? $13 - Write $0 - This clears Intreq0. INTREQ-0 is used only to inform the beginning of the first acquisition cycle. It will stay low until reset. It has no effect on further behavior of the state machine.

Program PAE pointer.

? $17 - FIFO control/status, write $1 this sets FIFO_REG bit to "1". This enables and allows access of the FIFO programming registers.

? $19 - Hardware FIFO reset, write $0 a write to this location will reset the hardware FIFO.

? Program FIFO register. (example 64 locations)

? $1A - Hardware FIFO, should read $7F (PAE = 128 locations)

? $1A - Hardware FIFO, write $3F (PAE = 64 locations).

Close FIFO register

? $17 - FIFO control/status, write $0 FIFO_Register bit = "0". Now we are in FIFO data normal mode. Data are written inside the FIFO and data can be read from the FIFO.

Start FIFO acquisition.

? $27 - Start Acquisition, write $0. A write to this register will start acquisition immediately.

? After starting acquisition – you should see INT_REQ0 go low when PAE pointer has reached 64 locations and stay low.

End acquisition

FIFO is full, lets read the data from the FIFO

$17 - FIFO control/status, read of FIFO status register should be $30.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
FIFO FF	FIFO PAF	FIFO PAE	FIFO EF	FIFO HF	N/A	N/A	FIFO REG

? $1A - Hardware FIFO, read data in the FIFO = 1 time. This will clear the FIFO-FF.

? $17 - FIFO control/status, FIFO-FF should be back to a 1 – should read $B0.

? If read more than 64 times then the PAF will be cleared and should read $F0.

Once the data in the FIFO has been read, write to address $9 the value $0 this will clear Intreq0 and then start the next FIFO access.

Programming Read/Write of AD8100 FIFO

This procedure explains how set the FIFO pointer then write values into the FIFO by hand and readback.

Each address starts from the base of the I/O space different carriers might vary.

Program the Internal sample clock

? At 8Mhz to set the Internal Clock at 200uS do the following.

? $00 - Internal clock divisor 0, write $3F.

? $03 - Internal clock divisor 1, write $6.

? At 32Mhz to set the Internal Clock at 200uS do the following.

? $01 - Internal clock divisor 0, write $FF.

? $03 - Internal clock divisor 1, write $18.

Clear Intreq0 latch

? $13 - Write $0 - This clears Intreq0. INTREQ-0 is used only to inform the beginning of the first acquisition cycle. It will stay low until reset. It has no effect on further behavior of the state machine.

Program PAE pointer.

? $17 - FIFO control/status, write $1 this sets FIFO_REG bit to “1”. This enables and allows access of the FIFO programming registers.

? $19 - Hardware FIFO reset, write $0 a write to this location will reset the hardware FIFO.

? Program FIFO register. (example 64 locations)

? $1A - Hardware FIFO, should read $7F (PAE = 128 locations)

? $1A - Hardware FIFO, write $3F (PAE = 64 locations).

Close FIFO register

? $17 - FIFO control/status, write $0 FIFO_Register bit = “0”. Now we are in FIFO data normal mode. Data are written inside the FIFO and data can be read from the FIFO.

Disable Readback of FIFO

? $15 - Read/Write Trigger register, write $08 Setting this bit high will disable the automatic readback of the FIFO thus retaining the data for further readback.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
INT1 Reset RDFifo	Event Enable	Master	INT1 Reset EF	Disable/ Enable Readback	ACQ ONCE	IDLE Enable	CONV Source

Write Data to FIFO

? $1A – Hardware FIFO – we have set the pointer of the FIFO to allow up to 64 locations of writes before it is full, anything after 64 locations will be discarded.

? Proceed to make consecutive writes to Hardware FIFO $1A the values with wish in word when done proceed to next step.

Enable Readback of FIFO Data

? $15 – Read/Write Trigger register – write $00 this will clear the disable/enable readback bit and allow for readback of hardware FIFO.

Verify FIFO Data

? $1A – make consecutive reads of this address and verify data is correct.

6. BURST ACQUISITION

This example shows how to program the FIFO pointer (PAE and PAF) if needed, Start the acquisition, read the FIFO register pointer. Read data from FIFO and start over again

? Generate an interrupt at PAE pointer location (64 locations from beginning if not programmed)

? Fill the FIFO and stop when FF is activated (low)

? Acquisition will start upon receiving a software command

At the base address of the I/O access of your carrier board enter the following.

Program the sample clock

? At 8Mhz to set the Internal Clock at 200uS do the following.

? $00 - Internal clock divisor 0, write $3F.

? $03 - Internal clock divisor 1, write $6.

? At 32Mhz to set the Internal Clock at 200uS do the following.

? $01 - Internal clock divisor 0, write $FF.

? $03 - Internal clock divisor 1, write $18.

Clear Intreq0 latch

? $13 - write $0 Note: Intreq0 sole source is PAE active (high)

Program PAE and PAF pointer (optional)

? PAE is equal to 127 (7F hex) upon hardware FIFO reset or Power on reset. PAF is equal to FIFO full less 127 location

? $17 - FIFO control/status, write $1 this sets FIFO_REG bit to “1” (high) allowing access of the FIFO programming registers (PAE and PAF pointers).Note: Read or write at this same address provides access to the registers in a circular mode.

? $1A - Hardware FIFO reset, write $0 a write to this location will reset the hardware FIFO and enable access to the FIFO registers. When reset goes high, status of FIFO_REG bit defines access to FIFO registers or data FIFO.

? Program FIFO register. (example 64 locations)

? $1A - Hardware FIFO, should read $7F (PAE = 128 locations)

? $1A - Hardware FIFO, write $3F (PAE = 64 locations).

? $17 - FIFO control/status, write $0 FIFO_Register bit = “0”. Now we are in FIFO data normal mode (data are written inside the FIFO at end of conversion). Data can be read from the FIFO.

Start FIFO acquisition.

? $27 - Start Acquisition, write $0 A write to this register will start acquisition immediately.

? After starting acquisition - should see INT_REQ0 go low when PAE pointer is reached and stay low.

? A display of the signal WR_FIFO will show 8 access (each pulse is 8 FIFO writes) before INT-REQ0 goes low.

? Acquisition will continue until FIFO is FIFO FF active (default). PAF can be selected by programming register $7 Acquisition Control register

End acquisition

FIFO is full, let read data from FIFO

? $17 - FIFO control/status, read FIFO status should read $30.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
FIFO FF	FIFO PAF	FIFO PAE	FIFO EF	FIFO HF	N/A	N/A	FIFO REG

? $1A - Hardware FIFO, read data in the FIFO = 1 time. This will clear the FIFO-FF.

? $17 - FIFO control/status, FIFO-FF should be back to a 1 – should read $B0.

? If read more than 127 times then the PAF (127 location before the FIFO full as default) will be cleared – should read $F0.

? If sufficient read is performed (more than the PAE location pointer) INTREQ0 can be re-programmed by clearing INT latch at address $09

7. ACQUISITION IN A PRE-TRIGGER AND POST TRIGGER MODE.

Upon a start acquisition command, the FIFO will acquire data until PAE pointer is reached. Then the FIFO will stay IDLE waiting to receive a signal (End acquisition Idle) to resume storing data.

The acquisition is still storing the new data and discarding at the same time the oldest data.

After the acquisition has resume, the FIFO will be filled up to FF or PAF as example above.

Proceed as example 1 to initialize the PAE, PAF pointer if needed.

Programmation of the PAE pointer as the idle location

Write $1 to the Acquisition control register ($0F). PAE will now be the source for the idle point.

Acquisition will start as above, store data up to the Idle pointer, discard old data when new data are stored, and wait for a signal to resume storing data until end of acquisition. In this example we use a software command $4. Other source is described in the manual.

Start Acquisition

? FIFO control/status, should read $F8

? $27 write $0 at location $13

A write to this register will start acquisition immediately.

PAE interrupt is generated (INTREQ0).

Display of the WRFIFO and RDFIFO lines when INTREQ0 go low should show RDFIFO active WHEN WRFIFO is active.

RESUME acquisition by ending IDLE mode

? $04 - Trigger source, write $00 this command is used to resume acquisition.

? $0B - FIFO control/status, should read $30 Acquisition has ended with FF active. Other condition can be programmed (see manual).

4. Real-Time Acquisition (using Ext-CLK and reading of A/D data using DMA state machine).

This example shows how to program the IP-AD8100K using Ext-CLK as a sample clock. After starting initialization, the host arms the acquisition. The next Ext-CLK will start the acquisition on the 8 channels simultaneously. An INTREQ-0 is then asserted on the first acquisition. The 8 A/D channels results are stored inside the FIFO. Setting up the hardware FIFO with PAE pointer set to 0 and PAF pointer set to FFF8, INTREQ-1 will be activated when 8 samples are written, then the acquisition will stop.

The DMA state machine can now read the 8 results. At the end of the 8 reads the state machine is then ReArmed for the next Ext-CLK.

INTREQ1 can be programmed to be removed upon the first read of the FIFO, or FIFO being empty (EF = 0). In this case PAE and EF are changing at the sametime, due to the fact that PAE pointer is set to zero.

The following will explain how to program the IP-AD8100K. Each address is written starting from the base address of the I/O space on the carrier board.

PROGRAM REGISTER FOR EXT-CLK

? $0F -write $84 this sets up Ext-CLK and Stop acquisition source to be PAF.

? Ext-CLK's input is on the input side of the IP on pin 47.

Bits 7-6	Bits 5-4	Bits 3-2	Bits 1-0
Sampling Clock Source	Trigger Source	Stop Acquisition Source	Trigger Enable Source

Sampling Clock Source determines where the A/D converters get their start conversion source from.

Sampling Clock Source	Meaning
00	Internal Sampling Clock (IP Clock divided by Divisor)
01	IPSTROBE
10	External Clock from connector

Stop Acquisition Source determines the cause of ending the acquisition.

Stop Acquisition Source	Meaning
00	FIFO FF Flag
01	FIFO PAF Flag

Program PAF as source for INTREQ-1

? $11 - write 2 This will cause INTREQ-1 to go low when PAF goes low.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
Not Used	Not Used	END_ ACQ_ OUT	FIFO EF	FIFO PAE	FIFO HF	FIFO PAF	FIFO FF

Program Read / Write register

? $15 write – 14 enable (Acq Once bit 2) and (INTREQ-1 reset EF bit 4). Acq Once - the purpose is to Rearm the acquisition upon FIFO empty (EF = 0). INT1 reset EF - when this bit is enabled INTREQ1 is removed when EF goes low.

BIT 07	BIT 06	BIT 05	BIT 04	BIT 03	BIT 02	BIT 01	BIT 00
INT1 RESET RDFIFO	Event Enable	Master	INT1 RESET EF	8-32Mhz	ACQ ONCE	IDLE Enable	CONV Source

Clear INTREQ-0 latch

? $11 write $0 Note: INTREQ-0 is the sole source for PAE active (high). INTREQ-0 is used only to inform the beginning of the first acquisition cycle. It will stay low until reset. It has no effect on further behavior of the state machine.

Program PAE and PAF pointer

PAE is equal to 127 (7F hex) upon hardware FIFO reset or Power on reset. PAF is equal to FIFO full less 127 location

? $1A - Hardware FIFO reset, write $0 a write to this location will reset the hardware FIFO and enable access to the FIFO registers.

? $17 - FIFO control/status, Should read $C8 now write $1 This sets the FIFO_REG bit to “1” (high) which allows access of the FIFO programming registers (PAE and PAF pointers).

Program FIFO register

? Note: Read or write at this same address provides access to the registers in a circular mode.

First access programs PAE, after writing increment on the same address without making a read. The access programs PAF, after writing exit the register.

? $1A - Hardware FIFO, should read $7F (PAE = 128 locations)

? $1A - Hardware FIFO, write $0 for PAE then increment without a read of address.

? $1A - Hardware FIFO, write $FFF8 and exit register

? $17 - FIFO control/status, write $0 This sets the FIFO_REG bit = “0” (low). Now we are in FIFO data normal mode (data are written inside the FIFO at end of conversion). Data can be read from the FIFO.

Start FIFO Acquisition

? $27 - Start Acquisition, write $0 A write to this register will arm the state machine. The next Ext-CLK signal will start a conversion.

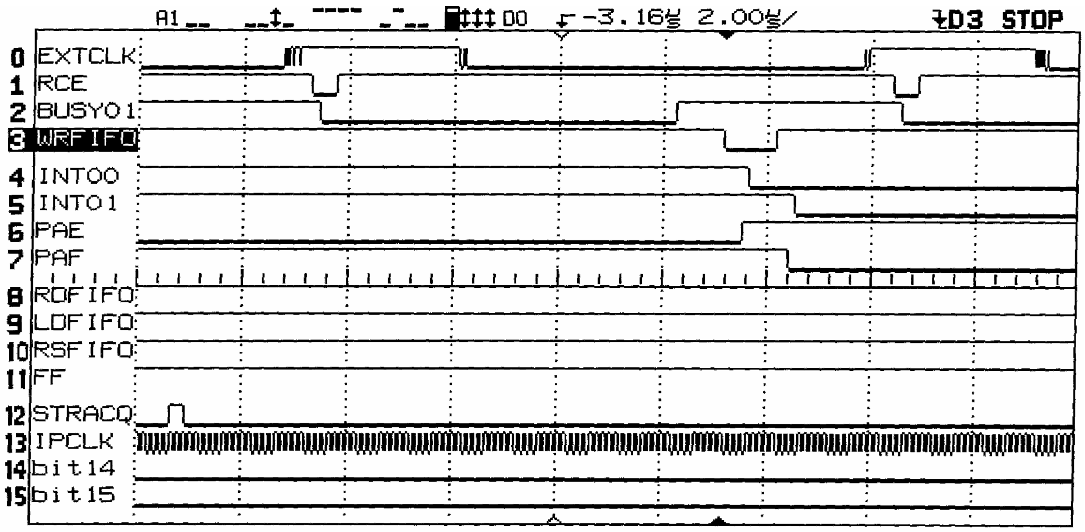
? After starting acquisition - should see INT_REQ0 go low when PAE pointer is reached and stay low.

? A display of the signal WR_FIFO will show 1 access (each pulse is 8 FIFO writes) before INT-REQ1 goes low.

? Acquisition will continue until PAF goes low and INTREQ-1 goes low.

? See timing for example of first acquisition.

Start Acquisition timing

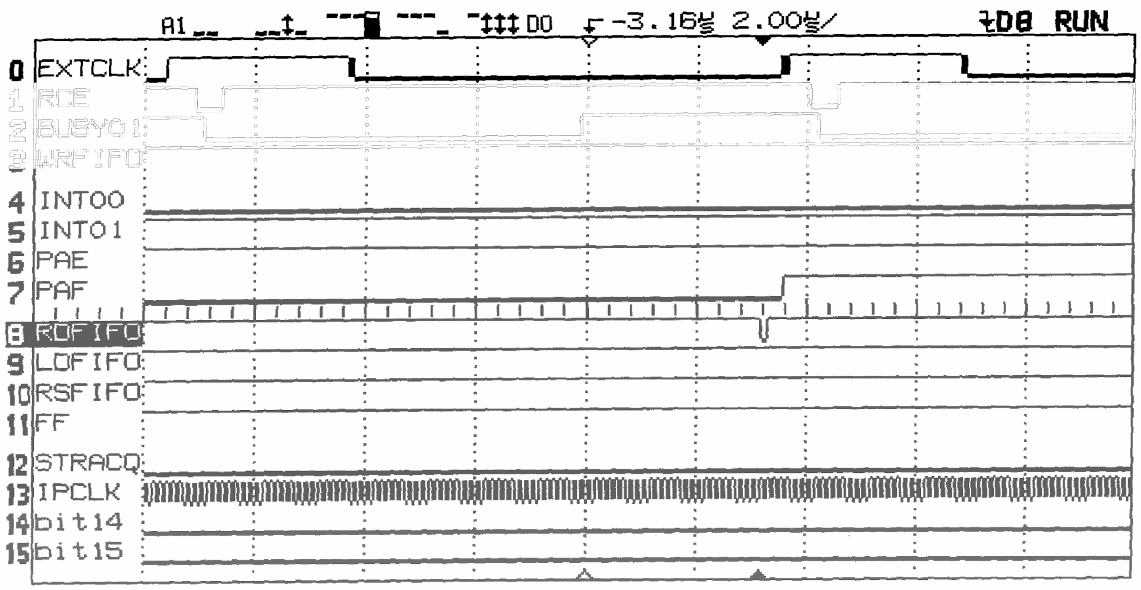


Read data from FIFO

? $1A – Hardware FIFO, Make first read of FIFO. Notice PAF goes high, meaning there is less than 8 data left in FIFO.

? See timing for example of first read.

First read of hardware FIFO

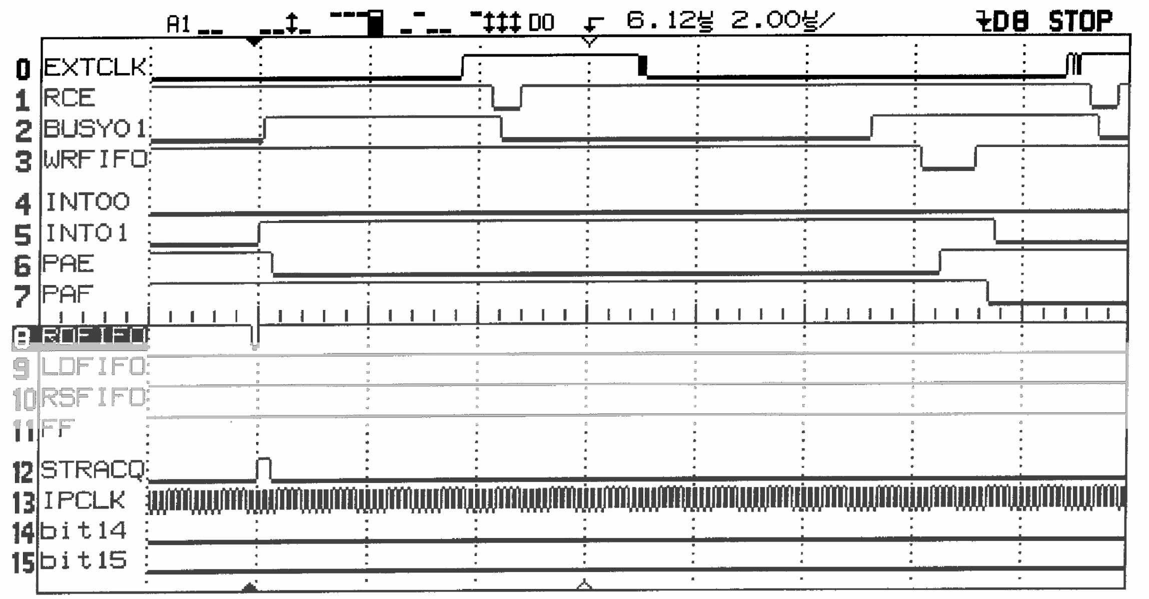


Continue reading FIFO data

? $1A – Hardware FIFO, Continue reading the FIFO. After the 8th read the PAE pointer goes low and INTREQ-1 goes high, Rearming itself for the next acquisition upon receiving Ext-CLK.

? See timing for example of the 8th (last read).

Last read of hardware FIFO



FIFO ACCESS

Read of the FIFO is located at the Base address I/O space +$1A.

To allow the use of a DMA machine that cannot read from a location without incrementing and writing to another location, we added the possibility to access the FIFO at 8 consecutive addresses so that the DMA *will look like* it's incrementing the address but for the AD8100 module it will be accessing the same address. The 8 addresses are located at BASE address I/O space + $28 to $3F.

Reading first at the address $28 will provide the data for channel #1, reading at the next address $29 will provide the data from channel #2 and so on.

Copyright ? 2000, ALPHI Technology Corporation

It is not possible to read channel #2 at address $29 and then read at address $28 for channel #1.

We also provide the possibility within the memory space to access the FIFO data.

Reading at the BASE address IP memory will provide the data from the channel #1, read at next location will be channel #2 and so on. The image of the FIFO will look like it has been mapped into the memory location from the DMA's point of view.

Note: Address specified in this manual are seen from an ALPHI DSP carrier where each address represents a 32bit word.

If using another carrier, the address will have to be shifted to the left once.

Example: address $28 will be $30.