ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *Datel PCI-416-G2A* at our website: ***Click HERE***

ORDERING INFORMATION

PCI-416 ___ 2 ___

Input Channels, A/D Resolution

B = 4SE chans., 14 bits
D = 1D chan., 12 bits
E = 16SE/8D chans., 12 bits
F = 2SE simul. chans., 12 bits
G = 2SE simul. chans., 14 bits
H = 1D chan., 12 bits
J = 8SE simul. chans., 12 bits
K = 2SE simul. chans., 12 bits
L = 16SE simul. chans., 12 bits
M = 4D simul. chans., 16 bits
N = 2SE simul. chans., 14 bits

FIFO Memory Size

2 = 8k A/D samples

Analog Input Polarity

A = Bipolar
B = Unipolar
Use this designator *only* for G, J, K and L models requiring polarity selection. **Caution:** Read the input range specifications carefully.

Example: PCI-416G2A 2 simul. channels, 14-bit A/D's, 8k FIFO, ±5V input

61-7342340	SMA male to BNC male coaxial cable, 1 meter length. One cable required per channel.
PC-490B	DB-25 screw termination adapter (cable not included).
PC-30259	PCI-416 less A/D module. Accepts 16-bit high-speed digital input and strobe at the module sockets.
UM-PCI-416	Spare user manual. One is included with board.

Each board is power-cycle burned-in, tested and calibrated. All models include a user's manual. An example program disk is available on request at no charge.

Software:

PCI-416SET	Setup/configuration data collection program. Executables only.
PCI-416SRC	Source code for setup and configuration program on MS-DOS disks. Includes "C" and assembly source code and executables. Documentation is on disk.
PCI-416NT, NTS	Windows NT version of PC-416SET/SRC. The full source code (PCI-416NTS) includes the execuatables (PCI-416NT). Consists of a user interface, DLL, and device driver.

Note: Software is *not* included with the board. Please add to your order

CE

DATEL® INNOVATION and EXCELLENCE

ISO 9001 REGISTERED

DS-0343A 8/96

DATEL, Inc. 11 Cabot Boulevard, Mansfield, MA 02048-1151
Tel: (508) 339-3000 / Fax: (508) 339-6356
For immediate assistance: (800) 233-2765

DATEL (UK) LTD. Tadley, England Tel: (01256)-880444
DATEL S.A.R.L. Montigny Le Bretonneux, France Tel: 1-34-60-01-01
DATEL GmbH Munchen, Germany Tel: 89-544334-0
DATEL KK Tokyo, Japan Tel: 3-3779-1031, Osaka Tel: 6-354-2025

DATEL makes no representation that the use of its products in the circuits described herein, or the use of other technical information contained herein, will not infringe upon existing or future patent rights. The descriptions contained herein do not imply the granting of licenses to make, use, or sell equipment constructed in accordance therewith. Specifications are subject to change without notice. The DATEL logo is a registered DATEL, Inc. trademark.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Start and Exit Mode

For applications which need the data streaming continuously to external software, the extensive library functions can easily be interleaved with user-written code. A special "start and exit" mode in PCI-416SET simply configures the board via menus, starts A/D conversion then exits to the operating system without saving any data. Next, the user's following program (usually in a batch file) retrieves data directly from the on-board registers or via 32-bit bus master mode. This is simple to program and very high speed. This data collection program may loop back repeatedly to get fast blocks of data and the PCI-416 continues filling the FIFO while the user program runs. This is true concurrent coprocessing with no lost data.

Start and exit mode offers a high degree of control over the board while avoiding time-comsuming menu design and coding. The system will even wait for an external trigger, giving time to set up the data collection program. Start and exit mode can be saved to automatically run, like all other PCI-416SET applications. A typical program flow for start and exit is shown:

1. SETUP PCI-416 board.
2. Save configuration to disk.
3. Start A/D and exit to MS-DOS (the A/D stays running).
   <start of user's program>
   LOOP:
4. If the FIFO overflowed, process the error.
5. Move block of A/D data from FIFO to host memory.
6. Do other processing on A/D data block (math, disk, display, etc.).
7. If more data is needed, GO TO LOOP.
8. Else, stop the A/D and quit.

The "Do other processing ..." step is a program written by the user. Notice that the PCI-416 continues with non-stop A/D sampling during this step.

Besides being a fast data recorder, PCI-416SET also performs register tests to verify proper board operation. In addition, PCI-416SET does calibration using an external dc voltage source, loads data into the D/A channel and exercises the digital I/O port.

The source code may be modified by the user (or by DATEL under special order) to adapt to any conceivable function.

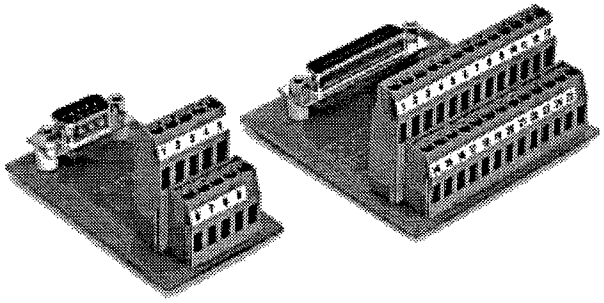


Figure 7. PC-490A and PC-490B D-Connector to Screw Terminal Adapters

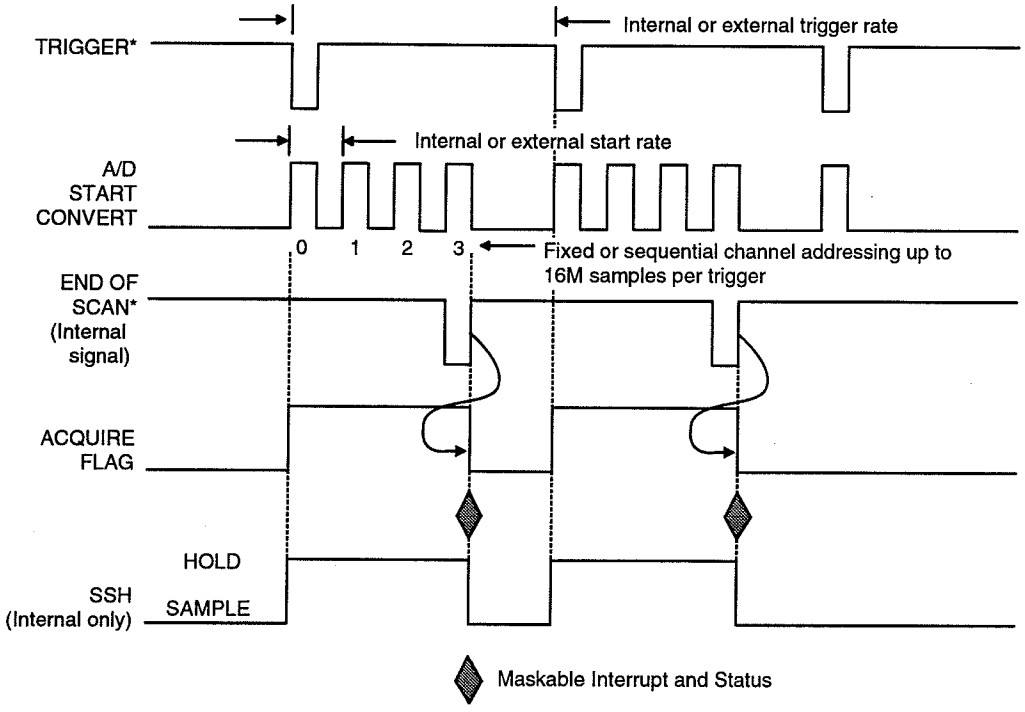


Figure 8. PCI-416 Timing Diagram

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Highly Parallel Array Sampling

Although the F, G, J,K, L, and N models of the PCI-416 offer unique high-performance simultaneous-sampling capabilities, this can be extended by connecting several PCI-416's in parallel. A master PCI-416 can distribute its internal trigger signal to several other 416's for concurrent sampling with practically no phase lag. Applications include sensor arrays, astrophysics, biomedical sampling, engine cells, multichannel audio, and aerospace structure testing.

Two interconnection methods accommodate either externally or internally triggered applications. External triggers simply connect in parallel to all PCI-416's. Each board is armed with its sample count and clocking systems to perform parallel sampling. Only one board sends interrupts. When data is ready, the host computer rapidly retrieves data blocks from each board in sequence.

For internal triggers, one board is the master generating triggers for itself and for all other boards. Slave 416's attach to the master 416's trigger output connection.

Trigger and Sample Count Systems

The PCI-416 accepts one of three triggers - external analog, internal or external digital. All three initiate identical internal actions. For the internally generated trigger, either a single trigger can be accepted ("single trigger mode") or the trigger can repeat ("continuous trigger mode") with programmable delays between each trigger.

Internal trigger rates are independent from but synchronous to the internal programmable A/D clock rate. The trigger starts a frame of samples. Each frame can be from one to 16,777,216 samples using the 24-bit counter. The system will collect the number of samples in the sample counter then stop and wait for the next trigger. Meanwhile, the sample counter will automatically reload in anticipation of the next trigger. Data flows into the FIFO memory which will notify the host that it has data to be saved. The FIFO size is independent of the frame size, therefore FIFO flags will occur separately from the sample counter.

A PCI bus interrupt can be generated after each frame completes (the "ACQuire" interrupt), or at each FIFO half full signal. Interrupt is also available after each bus master block transfer.

Three basic trigger modes are offered. A single trigger will start one frame then stop. The continuous trigger mode will generate repeating frames. In the last "forever" mode, the sample counter is disabled. A single trigger will start sampling indefinitely. The host computer must then externally decide when to stop sampling.

The PCI-416 will automatically control its own channel addressing such that the address advances immediately as a sample is sent to the FIFO. In single-channel mode, each frame will consist of data from only the selected channel. In automatic sequential addressing ("autoincrement"), the frame will contain one or more scans of channels, with addresses automatically wrapping around according to the channel capacity of the analog module.

The combination of programmable sample count, frame rate, A/D rate, and channel addressing mean that practically all conceivable applications can be done. The basic system timing is shown in Figure 8.

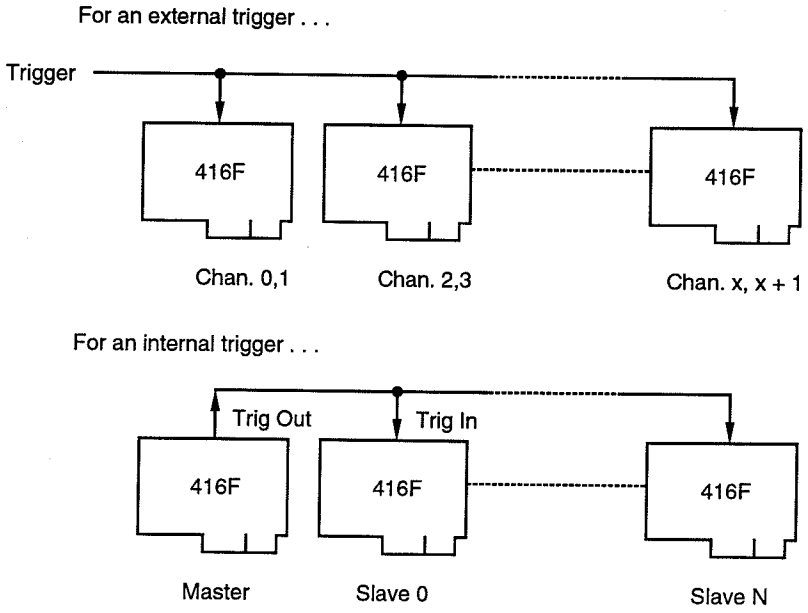


Figure 6. Highly Parallel Simultaneous Sampling

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Pre/Post Trigger Transient Capture Applications

A certain class of applications requires data sampled relative to one or more external events. Data before and after the event need to be analyzed. If the exact time of those external events cannot be predicted accurately but the event can be identified with a trigger, data must be recorded continuously then processed after the event occurred. At higher sample rates, the user must use all memory storage which has limited capacity but is still large enough to capture the event. A ring buffer circular storage method is used where new samples continually overwrite the oldest samples.

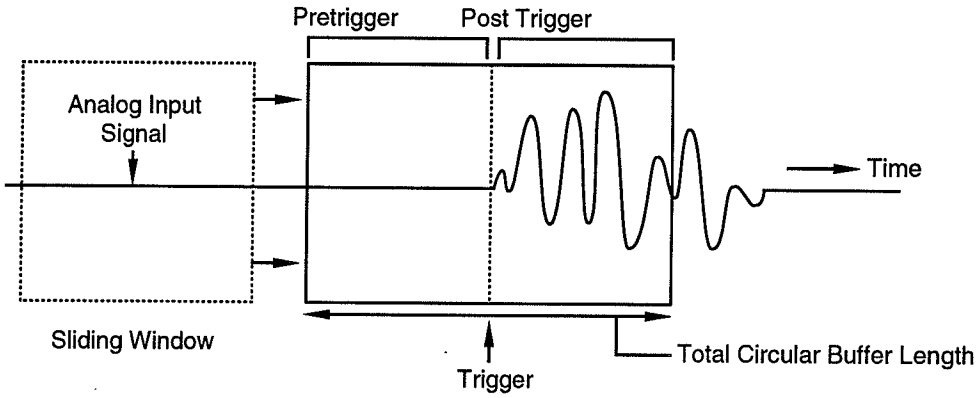


Figure 4. Transient Signal Capture

The PCI-416 accepts either a digital or analog (threshold trip) event trigger. An on-board D/A converter sets the comparator voltage level for the analog trigger. The system stores data before and after the trigger. A post trigger sample counter selects the number of offset samples after the trigger. The number of pretrigger samples equals the total circular storage minus the post trigger size. Note that pretrigger samples in Figure 5 are skewed over the buffer tail.

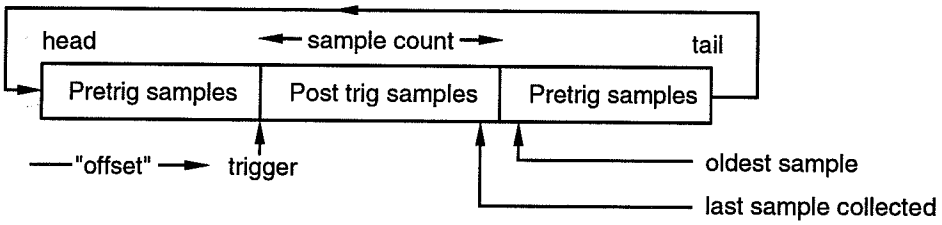


Figure 5. A/D Data Ring Buffering

A/D collection continues after the trigger until the system has stored the number of samples specified in the sample counter. The trigger sample can be found using backwards circular offset from the last sample saved. Multiple external events can be identified using a combination of the post trigger method and the marker inputs.

Special software available for the PCI-416 can access huge PCI memory. This requires the 80486 or Pentium CPU to enter protected mode and transfer the data. Collected A/D samples can then be saved to disk or tape.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A/D Data Format

A/D data is delivered as a stream from the FIFO memory. For multichannel inputs, this means that data is multiplexed by the channel address with a modulo address wrap-around at the top channel. For example, with 4-channel inputs, the output channel sequence is 0, 1, 2, 3, 0, 1, . . . One additional factor is that the 32-bit wide dual FIFO contains two A/D samples. Therefore the longword sequence is 0,1 . . . 2,3 . . . 0,1 . . .

The FIFO output can take two formats depending on which analog module is used and whether single-channel or autosequential (autoincrement) channel addressing is selected. For single-channel mode, data appears as follows:

31 16	15 0
Sample N+1	Sample N

If the addressing is selected for autoincrement, data appears this way:

31 16	15 0
Channel N+1	Channel N

Note that all A/D data is right-justified within the 16-bit data word with sign extension to bit 15 or 31. Also be aware that the PCI-416 uses "Intel" or little-endian addressing where lower (or earlier) data is lower in word memory.

PCI-416SET/SRC SETUP/CONFIGURATION SOFTWARE

PCI-416SET/SRC is an MS-DOS example program and data recorder written mostly in "C". The program fully exercises and self-tests the board. It also serves as an excellent software library if you wish to make modifications. The executable PCI-416SET is included with PCI-416SRC.

- Performs PCI BIOS verification and setup
- Automatically configures to the display adapter, CPU and memory
- Initializes the interrupt and bus master systems and D/A output
- Allocates base or extended memory
- Performs self-test and A/D-D/A calibration
- Configures A/D sample rate, frame rate and sample counter
- Selects trigger mode and bus master or I/O block transfer
- Selects disk file output format to integer binary
- Saves data to base memory, extended memory or disk
- Full source code in "C" and assembly is available

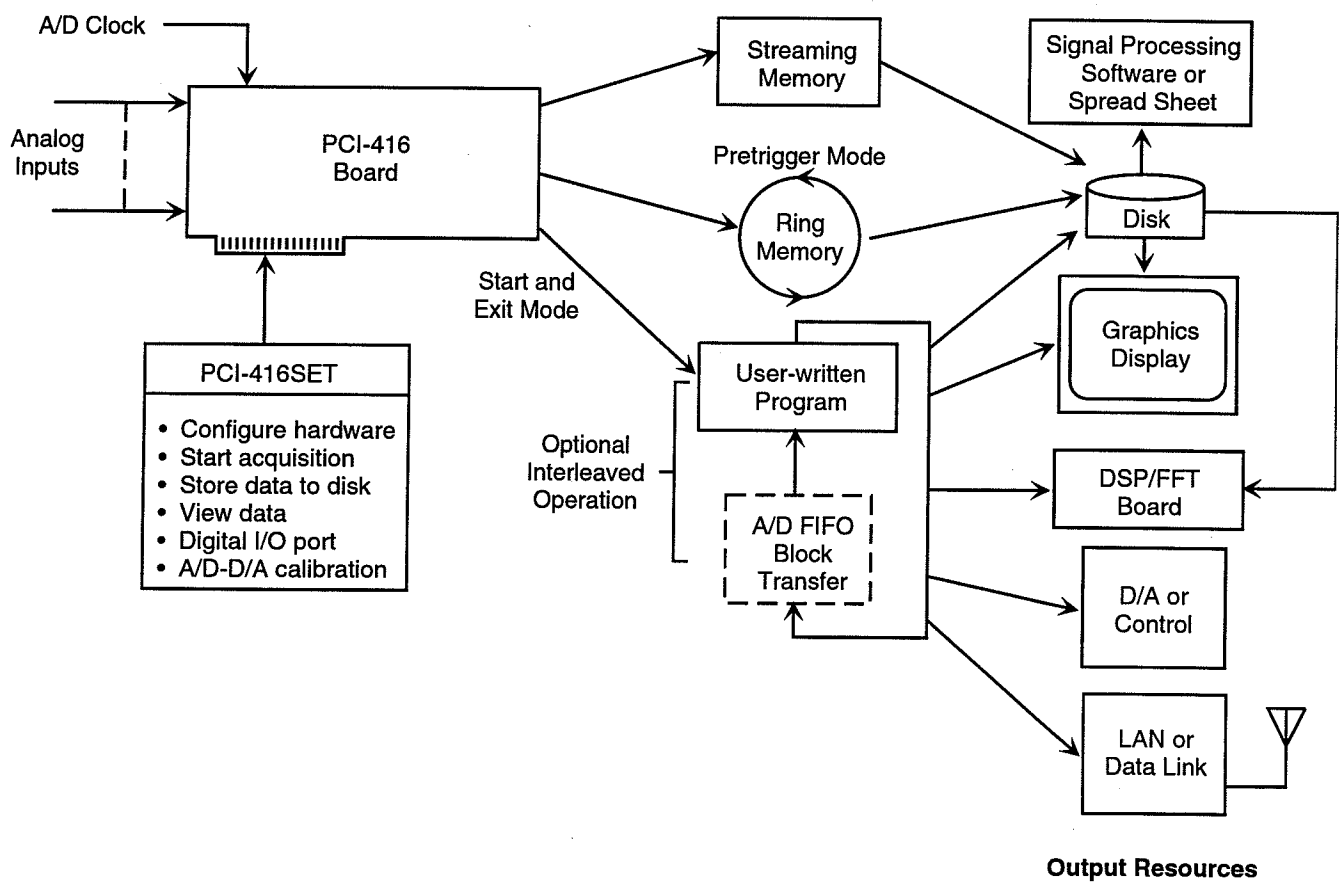


Figure 3. PCI-416SET Data Flow

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

CONNECTORS	
PCI bus	120-pin (dual 60) PCI edgeboard connector
Analog Input	DB-25 25-pin connector mounted on rear plate
External Trigger [Footnote 9]	On DB-25 analog connector
External A/D Clock In	BNC coaxial on rear plate
Digital I/O Port	Internal header connector. External clock, D/A mounted on board interior, suitable for flat cables.
DIGITAL I/O PORT	
Configuration	24 lines, programmable as input or output with latches and handshakes
Controller	82C55
Levels	TTL logic, 1 TTL load in or out (direct from 82C55)
Outport Settling Time	50ns, after write operation

FOOTNOTES

1. Resistor-programmed gain (user supplied) from x1 to x100 is available on PCI-416E with increased settling delay at higher gains. Requires precision gain resistor.

2. Total throughput includes MUX settling time after changing the channel address, S/H acquisition time to rated specifications, A/D conversion, and FIFO transfer. Total throughput is not delayed by host software whenever the FIFO is not full.

3. THD test conditions are:
   1. Input freq. 500kHz (416F) 200kHz (416B,E,G) 50kHz (416J,L,M) 1MHz (416D) 2MHz (416H)
   2. Generator/filter THD is -90dB minimum.
   3. THD computed by FFT to 5th harmonic.

$$THD = 20\left(\log 10\ \frac{(V2^2 + V3^2 + V4^2 + V5^2)^{0.5}}{Vin}\right)$$

   4. Inputs are full scale less 0.5dB. No channel advance.
   5. A/D sample rate = 500kHz (416B,E,G), 5MHz (416D), 2MHz (416F), 10MHz (416H), 250kHz (416J), 190kHz (416L,M)
   6. Crystal oscillator is used.

4. The rates shown for sequential sampling are the maximum A/D converter start rates and include MUX sequencing and settling. For example, if four channels of the PCI-416E were scanned, the maximum sample rate on any one channel would be 2µs x 4 channels = 8µs (125kHz per channel).

5. For fastest response on the analog comparator trigger, keep the reference voltage near the trip input voltage. To avoid overload recovery delays, do not let the trip input (or any other analog input) exceed ±10V.

6. Allow 20 minutes warmup time to rated specifications for models PCI-416B,G,M,N.

7. Use the crystal oscillator for best harmonic performance.

8. Avoid mixing external triggers which are a close submultiple of the internal A/D start clock to prevent sample jitter.

9. The BNC connector may be rewired to either external trigger or external A/D clock.

10. PCI-416J and 416L bipolar input is user-selectable ±5V or ±10V (default) per channel. Total gain error over temperature range is ±4 LSB maximum. Total zero/offset error over temperature range is ±4 LSB maximum. Monotonicity: no missing codes over temperature range.

11. Input polarity. Some models are fixed as bipolar only whereas others are user-selectable unipolar or bipolar. Still others require separate model numbers.

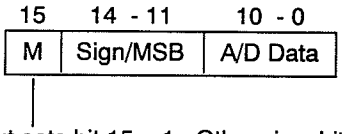
12. PCI-416D, H, K, and M inputs are jumpered as single-ended. Special, user-configured wiring allows differential operation. Contact DATEL.

13. Models F, G, J, K, L, M, and N use one A/D converter per channel.

14. The customer must use shielded cables to insure EMC compliance.

Marker Input

When selected in the Command Register, the marker input is for tagging A/D samples to an external event such as a clock timebase. The marker sets bit 15 of the A/D word to logic "1" for one sample clock cycle. Lower A/D bits still retain sign extension polarity. This tag is now stored in the FIFO along with the A/D sample. The user may do this as often as needed, and the marker can be asynchronous with the A/D sample clock. Post processing software then searches through the saved data to find each marked sample. The marker bit is not available for 16-bit A/D's.



Marker input sets bit 15 = 1. Otherwise, bit 15 = 0.
(12-bit A/D shown)

Figure 2. Marker Sample Tagging

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SPECIFICATIONS, CONTINUED

(Typical @ +25°C, dynamic conditions, unless noted)

A/D SAMPLE CLOCK	
Sample Clock Sources [Footnote 7]	Selectable from among: 1. Frequency synthesizer 2. 10MHz crystal oscillator 3. 20MHz crystal oscillator 4. 16-stage binary divider to either 1 or 2, maximum input: 10MHz 5. External digital input
Frequency Synthesizer	Output 5-10MHz in 625Hz steps, further divisible by 16-stage binary divider, all software programmable. Up to 40MHz is available in 2500Hz steps.
Total Sample Range	76.3Hz to 10MHz (40MHz available)
Oscillator Frequency Accuracy	±50ppm (+20 to +30°C)
Crystal Aging	±5ppm/year

TRIGGER CONTROL	
Trigger Sources [Footnote 8]	1. Analog threshold comparator using internal D/A to set trip level 2. Internal trigger derived from10MHz timebase, divided by 32-stage divider (82C54). Range: 20ns to 429.5 seconds. 3. External digital trigger
Trigger Response	Selectable from among: 1. Starts one frame ("single trigger mode") 2. Collects repeating frames, each started by a trigger ("continuous trigger mode"). 3. Runs the A/D "forever" (sample counter disabled)
A/D Samples per Frame	1 to 16,777,216 samples (24-bit counter) or "forever"
Analog Trigger Input Range	±10V
Analog Trigger Response	2µs [Footnote 5]
Analog Trigger Hysteresis	±40mV
Marker Input	Digital input which sets A/D bit 15 to logic "1" for one A/D clock cycle. Used to tag samples to external events if enabled.
Pretrigger Mode	The sample down-counter is delayed until an external trigger. Pretrigger samples are stored in a host ring buffer for transient capture, if enabled.

ANALOG OUTPUT	
Number of Channels	One channel
Function	Selectable from among: 1. General-purpose analog output 2. Threshold comparator for A/D trigger
Resolution	12 bits
Output Voltage Range	0 to +10V, ±5V, ±10V at 5mA max. (user selectable)
Linearity	±0.05% of FSR
Settling Time	5 microseconds (10V step)
Input Coding	Straight binary

PCI Bus	
Data Bus Size	32 bits
Address Bus Size	32 bits
PCI Controller Type	AMCC S5933 bus master or slave mode
Data Transfer	32-bit I/O or memory (selectable)
Bus Transfer Mode	Up to 2^{24} longwords, per PCI spec.
Interrupt	One interrupt, INTA# only per PCI spec.
Interrupt Sources	FIFO half full, sample count reached (ACQuire flag), bus master transfer done.

MISCELLANEOUS	
Board Identification Switch	4-bit DIP switch is factory preset to identify A/D module type. May be changed if another module is used.
Analog Section Modularity	The MUX-S/H-A/D module is socketed for function interchange.
Analog Section Adjustments	Offset and gain per channel for SSH on PCI-416F,G. A single offset and gain pot is provided on PCI-416B,D,E. Recommended recalibration interval is 90 days in stable conditions.
Operating Temp. Range	0 to +60°C, thermal shock ±1°C max per minute.
Storage Temp. Range	-25 to +85°C
Humidity	10% to 90%, non-condensing
Altitude	0 to 10,000 feet, forced cooling is required
Power Required	+5Vdc @ 3.0A max. from PCI bus. Compatible to +3.3V systems but makes no connection.
Outline Dimensions	4.2 x 12.28 x 0.5 inches, compatible to PCI bus

A/D MEMORY	
Architecture	First-In, First-Out (FIFO)
Memory Capacity	8192 A/D samples

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

FUNCTIONAL SPECIFICATIONS

(Typical at +25°C, dynamic conditions, gain = 1, unless noted)

ANALOG INPUTS	PCI-416L	PCI-416M*	PCI-416N*
Number of Channels	16 Simultaneous	4 Simultaneous	2 Simultaneous
Input Configuration (non-isolated)	Single Ended	Differential	Single Ended
Full Scale Input Ranges (user-selectable) (gain = 1)	±5V, ±10V, (user selectable) [Footnote 10]	±5V, 0 to +10V, (user selectable prgmble gain = x1 to x10)	±1.5V
Input Bias Current	N.A.	±20nA	N.A.
Input Overvoltage (no damage, power on)	±15V	±15V	±15V
Overvoltage Recovery Time, maximum	—	—	—
Common Mode Voltage Range, maximum	—	—	—
Input Impedance	8kΩ	10^10Ω	300Ω
SAMPLE/HOLD			
Acquisition Time	400ns	—	35ns
Aperture Delay	—	40ns	±10ns
Aperture Delay Uncertainty	—	—	5ps
A/D CONVERTER			
Resolution	12 bits	16 bits	14 bits
Conversion Period	2µs (all chans. in simul. sampling)	5µs (all chans. in simul. sampling)	200ns (all chans. in simul. sampling)
SYSTEM DC CHARACTERISTICS [Footnote 6]			
Integral Non-linearity (LSB of FSR)	±2	±4	±1
Differential Non-linearity (LSB of FSR)	±1	±3	±1
Full Scale Temperature Coefficient (LSB per °C)	[Footnote 10]	±1	±0.5
Zero or Offset Temperature Coefficient (LSB per °C)	[Footnote 10]	±1	±0.5
SYSTEM DYNAMIC PERFORMANCE [Footnote 2]			
Sample Rate (single channel only)	400kHz	200kHz	8MHz
Sample Rate per Channel (simul. or sequential chans.) [Footnote 4]	190kHz/chan.	200kHz/chan.	5MHz/chan.
Total Harmonic Distortion [Footnote 3]	-75dB	-80dB	-75dB

*Preliminary models, specifications subject to change. Call DATEL for latest information.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

FUNCTIONAL SPECIFICATIONS

(Typical at +25°C, dynamic conditions, gain = 1, unless noted)

ANALOG INPUTS	PCI-416G	PCI-416H	PCI-416J	PCI-416K
Number of Channels	2 Simultaneous	1	8 Simultaneous	2 Simultaneous
Input Configuration (non-isolated)	Single Ended	Differential	Single Ended	Single Ended
Full Scale Input Ranges (user-selectable) (gain = 1)	±5V or 0 to +10V (separate models)	±5V (other ranges special order)	±5V, ±10V [Footnote 10]	0 to +10V, ±5V (separate models)
Input Bias Current	±1nA	N.A.	±50nA	N.A.
Input Overvoltage (no damage, power on)	±15V	±15V	±15V	±15V
Overvoltage Recovery Time, maximum	2µs	1µs	—	—
Common Mode Voltage Range, maximum	—	±1V	—	—
Input Impedance	>1MΩ	2kΩ	8kΩ (bipolar)	1kΩ
SAMPLE/HOLD				
Acquisition Time	350ns	35ns	400ns	50ns
Aperture Delay	20ns	±10ns	—	10ns
Aperture Delay Uncertainty	±70ps	3ps rms	—	±7ps
A/D CONVERTER				
Resolution	14 bits	12 bits	12 bits	12 bits
Conversion Period	500ns*	100ns	2µs (all chans. in simul. sampling)	200ns
SYSTEM DC CHARACTERISTICS [Footnote 6]				
Integral Non-linearity (LSB of FSR)	±1.5	±1.5	±1	±2
Differential Non-linearity (LSB of FSR)	±1	±1	±1	±1
Full Scale Temperature Coefficient (LSB per °C)	±0.3	±1	[Footnote 10]	±0.1
Zero or Offset Temperature Coefficient (LSB per °C)	±0.3	±1	[Footnote 10]	±0.3
SYSTEM DYNAMIC PERFORMANCE [Footnote 2]				
Sample Rate (single channel only)	1MHz*	10MHz	400kHz	5MHz
Sample Rate per Channel (simul. or sequential chans.) [Footnote 4]	1MHz/chan.*	—	(250kHz/chan.)**	5MHz 10MHz (5MHz/chan.)
Total Harmonic Distortion [Footnote 3]	–80dB	–65dB	–75dB	–68dB

*Dual 2MHz 14-bit sampling is available on special order.
**A 380kHz/channel option is available on special order.

Artisan Technology Group - Quality Instrumentation... Guaranteed | (888) 88-SOURCE | www.artisantg.com

FUNCTIONAL SPECIFICATIONS

(Typical at +25°C, dynamic conditions, gain = 1, unless noted)

ANALOG INPUTS	PCI-416B	PCI-416D	PCI-416E	PCI-416F
Number of Channels	4	1	16SE/8D	2 Simultaneous
Input Configuration (non-isolated)	Single Ended	Differential	SE or Diff.	Single Ended
Full Scale Input Ranges (user-selectable)	0 to +10V ±10V	±5V (0 to +10V, special order)	0 to +10V ±10V	0 to +10V ±5V
(gain = 1)	±5V		±5V [Footnote 1]	
Input Bias Current	±1nA	—	±1nA	±1nA
Input Overvoltage (no damage, power on)	±15V	±15V	±15V	±15V
Overvoltage Recovery Time, maximum	2µs	2µs	2µs	2µs
Common Mode Voltage Range, maximum	—	±1V	±10V	—
Input Impedance	10MΩ	2kΩ	100MΩ	>1MΩ
SAMPLE/HOLD				
Acquisition Time	750ns	50ns	750ns	165ns
Aperture Delay	20ns	10ns	20ns	20ns
Aperture Delay Uncertainty	±100ps	±7ps	±40ps	±40ps
A/D CONVERTER				
Resolution	14 bits	12 bits	12 bits	12 bits
Conversion Period	1.6µs	200ns	500ns	400ns
SYSTEM DC CHARACTERISTICS [Footnote 6]				
Integral Non-linearity (LSB of FSR)	±1.5	±2	±1	±1
Differential Non-linearity (LSB of FSR)	±1	±1	±0.75	±1
Full Scale Temperature Coefficient (LSB per °C)	±0.3	±0.1	±0.1	±0.1
Zero or Offset Temperature Coefficient (LSB per °C)	±0.3	±0.3	±0.1	±0.1
SYSTEM DYNAMIC PERFORMANCE [Footnote 2]				
Sample Rate (single channel only)	500kHz	5MHz	2MHz	2MHz
Sample Rate per Channel (simul. or sequential chans.) [Footnote 4]	82kHz/chan.	—	125kHz/chan.	2MHz/chan.
Total Harmonic Distortion [Footnote 3]	–75dB	–68dB	–72dB	–70dB

Note: Model PCI-416J in short-cycled addressing is recommended in place of the previously announced PCI-416A. Model PCI-416E can substitute for the previously announced PCI-416C.

ANALOG INPUTS	
Programmable Gains	See Footnote 1
Common Mode Rejection (DC - 60Hz)	-80dB (g = 100) (416E)
Addressing Modes (short cycle channel addressing is software-selectable on PCI-416J,L)	1. Single channel 2. Simultaneous sampling 3. Sequential with autosequenced addressing 4. Random addressing by host software

A/D CONVERTER	
Output Coding	Positive-true, right justified, straight bin. (unipolar) or right-justified 2's comp. (bipolar) with sign extension thru bit 15
Warm-Up Period	10 minutes until rated specifications. [Footnote 6]

Please read all footnotes carefully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

sonar or acoustic sensor arrays, cross-channel computation, multiple carrier demodulation, interferometry, multichannel spectrometers, and highly concurrent system testing. High-quality, wide-bandwidth, low- noise A/D's and analog components are used. The design is ideal as an array processor "front end" or for DSP/FFT (Digital Signal Processing/Fast Fourier Transform) usage.

Two on-board software-programmable timebases run the A/D sample clock. A 40MHz frequency synthesizer provides high resolution whereas the 10MHz 16-stage programmable divider offers very low clock jitter. If preferred, external clocks may be accepted for both the A/D start clock and the trigger. And several PCI-416's may be connected in master-slave clocking for many simultaneous channels. A programmable 24-bit sample counter will collect long blocks up to 16 million samples. The trigger system can collect a single fixed length frame, N repeating frames separated by programmable delays, or it can run "forever". Interrupts to the PCI bus are programmable from the FIFO flags, bus master block transfer done, or the sample counter.

System features optimize gapless sampling without data loss. A pretrigger system can collect data continuously to host circuar memory (ring buffer) of several megabytes or more. When an external trigger is received, the PCI-416 will count down the number of preloaded post-trigger samples then automatically stop when all samples are collected. The trigger sample may then be found using a negative circular offset into the ring buffer, knowing the post-trigger sample count.

In addition, a digital marker input will tag data samples on the fly as often as needed. This provides later identification of external events without stopping sample collection. A D/A analog output channel is included to set the trip level for external analog triggers. Or the D/A can be used for analog output.

A general-purpose, 24-bit (82C55) digital I/O port which is separate from the fast A/D parallel port can control external circuits. The parallel port uses an internal header connector.

A menu-driven "no-programming" fast data recording system to memory or disk is available in the PCI-416SET software. This "SET" software can also be obtained in source code format to modify the system. The PCI-416 may be programmed in any language which can access a 32-bit PCI bus. For highest speed, the optimized "C" and assembly code is used under MS-DOS.

REGISTER I/O or MEMORY MAPPING

All of the PCI-416's registers require 32-bit instructions. DATEL software provides highly portable examples which can be used with any language. All registers are fully described in the User Manual included with the board.

Base Address Register	Function
BADR0	S5933 PCI controller operation registers
BADR1	Pass-Thru Address Register Latch (ARL)
BADR2	Read/Write general registers
	If ARL=0, Command register (write only)
	If ARL=4, Sample counter (write only)
	If ARL=8, Channel address register (write)
	Clear A/D FIFO memory (read)
	If ARL=12, A/D convert enable (write only)
	If ARL=16, PLL register (write only)
BADR3	High-speed read of status register or A/D FIFO data
	If ARL=0, Status register (read only)
	If ARL=4, A/D FIFO data (read only)
BADR4	Low-speed devices, 82C54 and 82C55, and analog output
	82C54 Programmable counter-timer (read/write)
	If ARL=0, Counter 0
	If ARL=4, Counter 1
	If ARL=8, Counter 2
	If ARL=12, Control register
	82C55 Programmable parallel port (read/write)
	If ARL=16, Port A
	If ARL=20, Port B
	If ARL=24, Port C
	If ARL=28, Control register
	Analog output (D/A channel)
	If ARL=32, D/A converter register (write only)

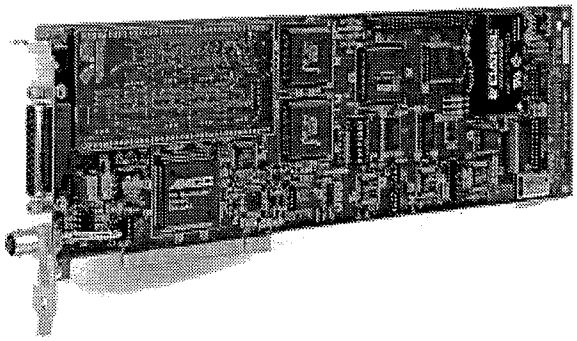
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DATEL®
INNOVATION and EXCELLENCE

PCI-416
Ultra-Performance, Analog Input Boards for PCI Computers

FEATURES

- **The ideal array-processor "front end"**
- **Up to 10MHz A/D sampling rates**
- **Choice of 12, 14 or 16-bit A/D resolutions**
- **Wideband inputs with low harmonic distortion**
- **Quick, 32-bit, PCI block transfers**
- **2 to 16-channel simultaneous sampling eliminates phase skew**
- **On-board A/D FIFO memory holds 8k samples**
- **32 megasamples or greater data streaming**
- **Pre/post-trigger, gap-free, ring buffering**
- **Great for DSP, FFT's, digital filtering, etc.**
- **Pentium® compatible**



GENERAL DESCRIPTION

The PCI-416 Family consists of several advanced-performance, data acquisition boards based on the 32-bit PCI bus architecture. With an emphasis on continuous, non-stop, high-speed streaming of A/D samples to host memory or disk, the system has been optimized for a wide range of signal-processing and data-recording applications. In very long "baseline" studies or high-speed transient analysis, the PCI-416 can collect more than 64 megabytes of "seamless" digitized data to host memory.

Exploiting a unique "banked" FIFO architecture, the PCI-416 moves two A/D words in each 32-bit PCI transfer. The FIFO memory (8k samples deep) serves to decouple the precise timing of the A/D converter from the block bursts of the PCI bus.

The PCI-416's optional analog front-ends utilize DATEL's low-noise, wide-bandwidth sampling A/D converters. All models exhibit excellent harmonic distortion and perform well in DSP/FFT applications. The PCI-416SET software readily implements a menu-driven, "no-programming", fast data recording system to memory and disk.

Several different "pluggable" analog options offer up to 16 input channels, multiple input ranges, single-ended/differential, sample rates to 10MHz, 12, 14, or 16-bit A/D resolution configurations and various simultaneous sampling choices up to 16 channels. The simultaneous feature is intended for parallel sampling applications that cannot tolerate phase skew introduced by the A/D system. These include

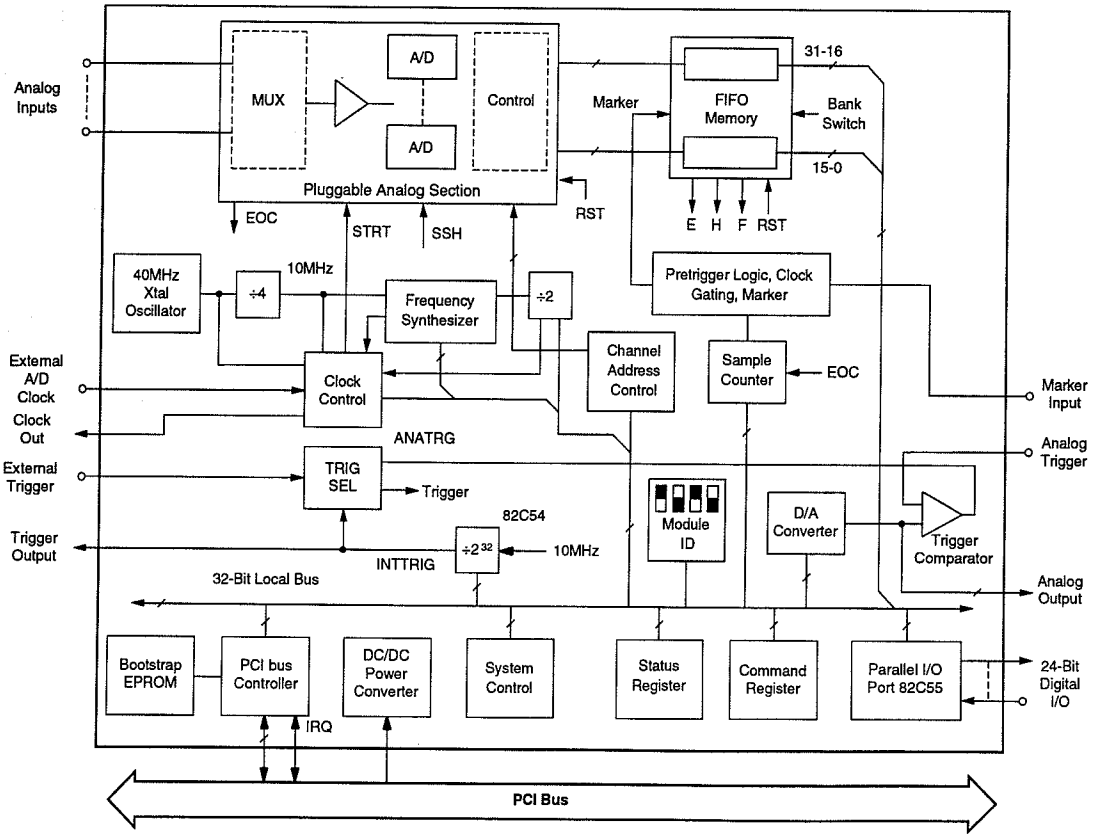


Figure 1. Functional Block Diagram

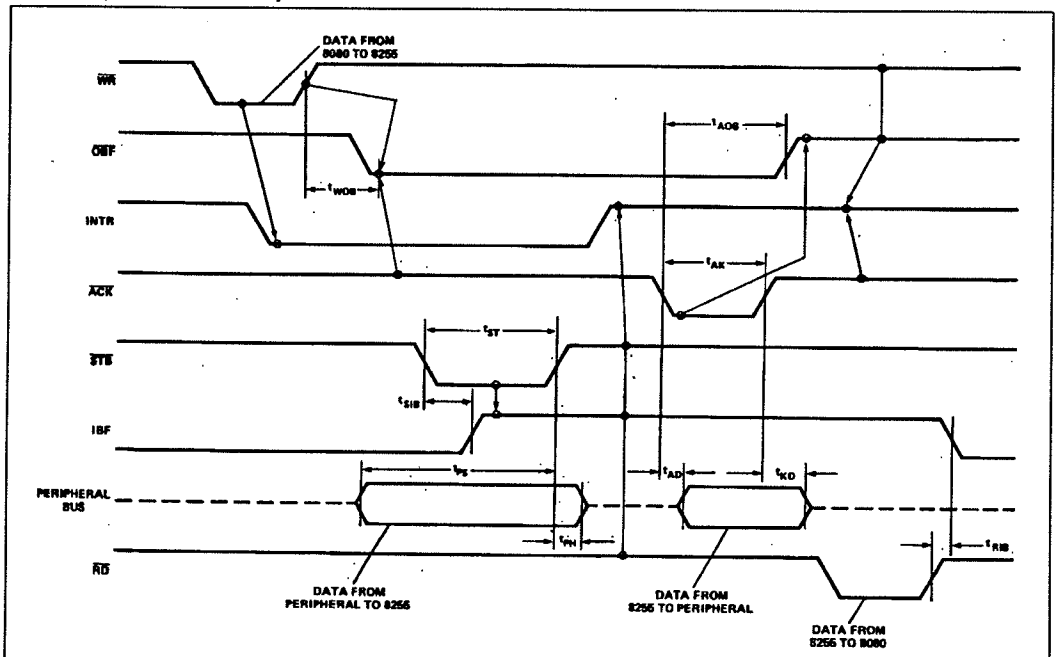
DATEL, Inc., 11 Cabot Boulevard, Mansfield, MA 02048-1151 (U.S.A.) • Tel: (508) 339-3000 Fax: (508) 339-6356 • For immediate assistance (800) 233-2765

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

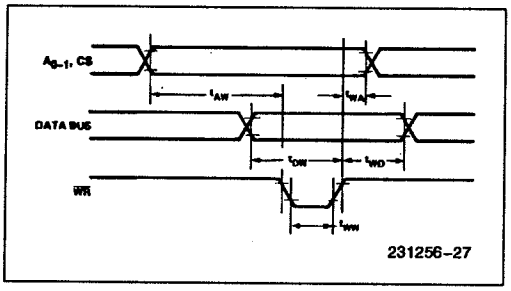
WAVEFORMS (Continued)

MODE 2 (BIDIRECTIONAL)

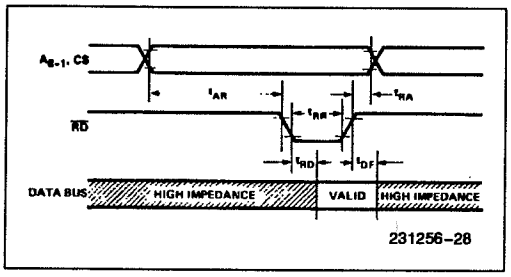


Note:
Any sequence where $\overline{WR}$ occurs before $\overline{ACK}$ AND $\overline{STB}$ occurs before $\overline{RD}$ is permissible.
(INTR = IBF • MASK • $\overline{STB}$ • $\overline{RD}$ + OBF • MASK • $\overline{ACK}$ • $\overline{WR}$)

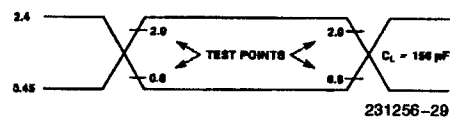
WRITE TIMING



READ TIMING

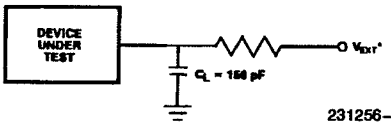


A.C. TESTING INPUT, OUTPUT WAVEFORM



A.C. Testing Inputs Are Driven At 2.4V For A Logic 1 And 0.45V For A Logic 0 Timing Measurements Are Made At 2.0V For A Logic 1 And 0.8 For A Logic 0.

A.C. TESTING LOAD CIRCUIT



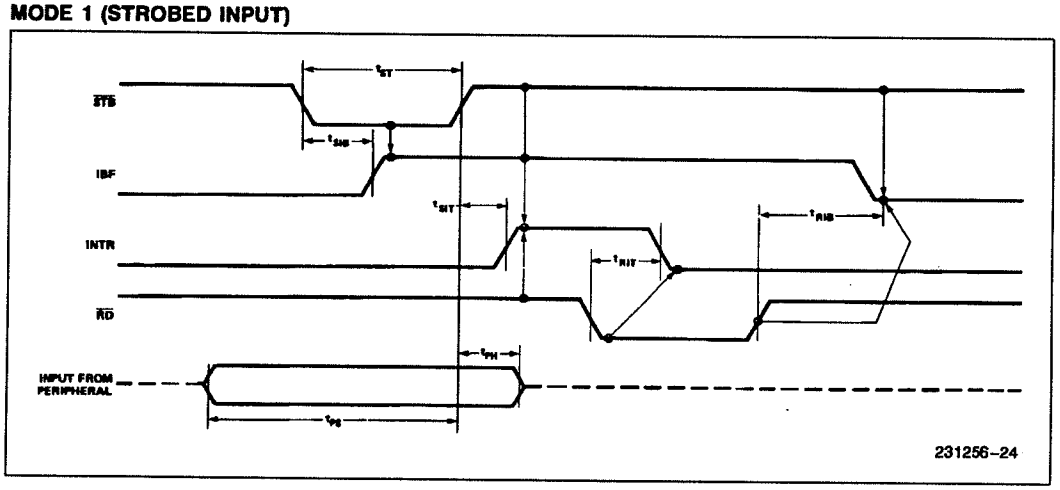
*V_{EXT} Is Set At Various Voltages During Testing To Guarantee The Specification. C_L Includes Jig Capacitance.

© INTEL CORPORATION, 1993

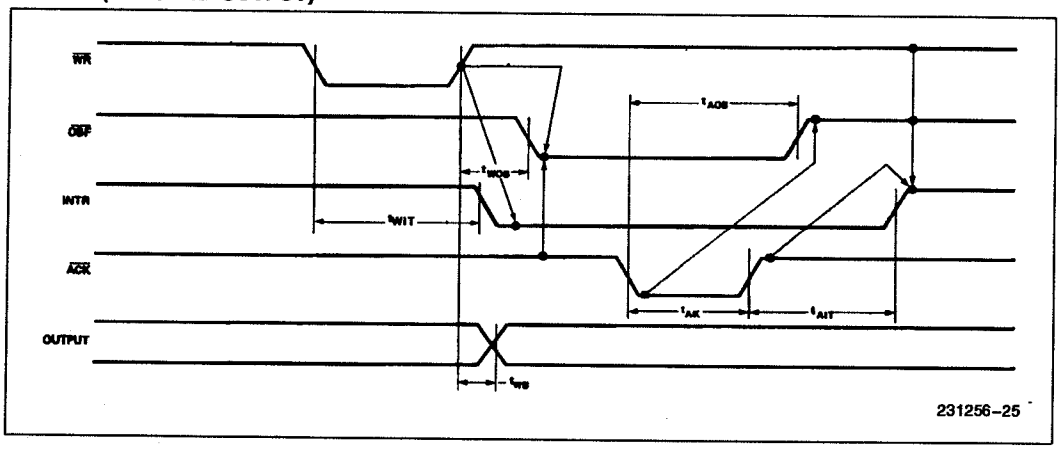
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

WAVEFORMS (Continued)

MODE 1 (STROBED INPUT)



MODE 1 (STROBED OUTPUT)

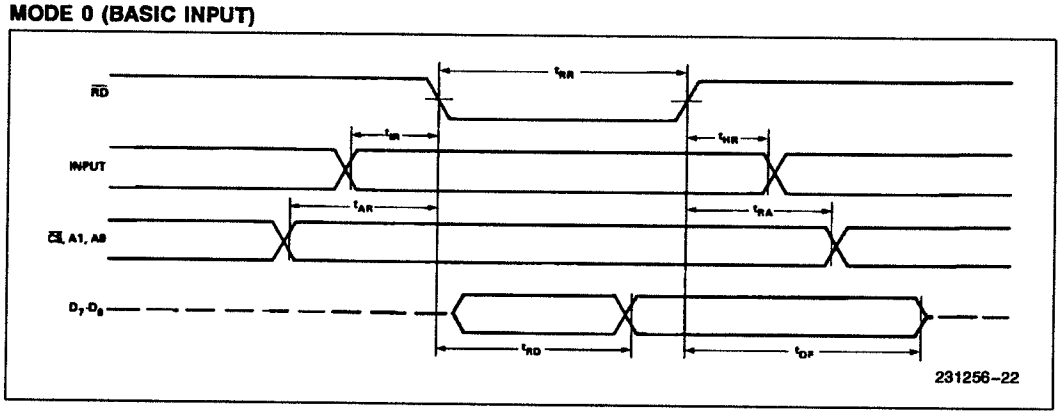


© INTEL CORPORATION, 1993

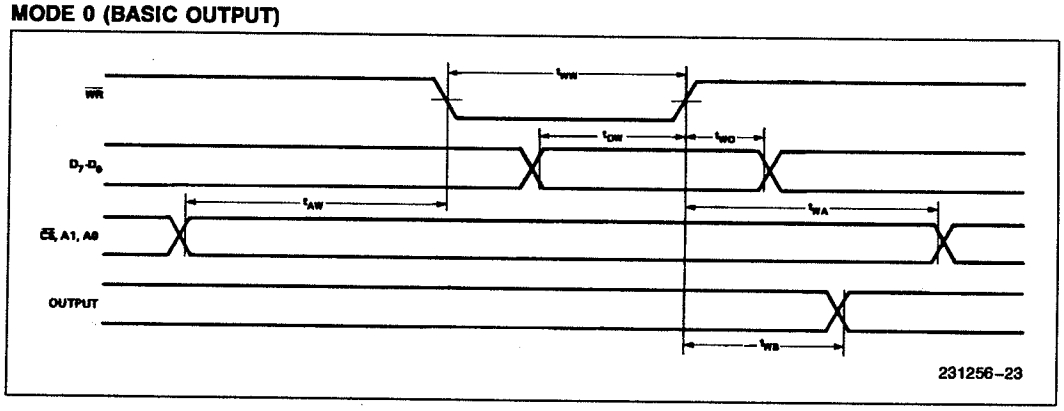
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

WAVEFORMS

MODE 0 (BASIC INPUT)



MODE 0 (BASIC OUTPUT)



© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

OTHER TIMINGS

Symbol	Parameter	82C55A-2		Units Conditions	Test
		Min	Max		
t_{WB}	$\overline{WR}$ = 1 to Output		350	ns	
t_{IR}	Peripheral Data Before $\overline{RD}$	0		ns	
t_{HR}	Peripheral Data After $\overline{RD}$	0		ns	
t_{AK}	$\overline{ACK}$ Pulse Width	200		ns	
t_{ST}	$\overline{STB}$ Pulse Width	100		ns	
t_{PS}	Per. Data Before $\overline{STB}$ High	20		ns	
t_{PH}	Per. Data After $\overline{STB}$ High	50		ns	
t_{AD}	$\overline{ACK}$ = 0 to Output		175	ns	
t_{KD}	$\overline{ACK}$ = 1 to Output Float	20	250	ns	
t_{WOB}	$\overline{WR}$ = 1 to $\overline{OBF}$ = 0		150	ns	
t_{AOB}	$\overline{ACK}$ = 0 to $\overline{OBF}$ = 1		150	ns	
t_{SIB}	$\overline{STB}$ = 0 to IBF = 1		150	ns	
t_{RIB}	$\overline{RD}$ = 1 to IBF = 0		150	ns	
t_{RIT}	$\overline{RD}$ = 0 to INTR = 0		200	ns	
t_{SIT}	$\overline{STB}$ = 1 to INTR = 1		150	ns	
t_{AIT}	$\overline{ACK}$ = 1 to INTR = 1		150	ns	
t_{WIT}	$\overline{WR}$ = 0 to INTR = 0		200	ns	see note 1
t_{RES}	Reset Pulse Width	500		ns	see note 2

NOTE:
1. INTR ↑ may occur as early as $\overline{WR}$ ↓.
2. Pulse width of initial Reset pulse after power on must be at least 50 μSec. Subsequent Reset pulses may be 500 ns minimum.

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

CAPACITANCE

T_A = 25°C, V_{CC} = GND = 0V

Symbol	Parameter	Min	Max	Units	Test Conditions
C_{IN}	Input Capacitance		10	pF	Unmeasured pins returned to GND f_c = 1 MHz(5)
$C_{I/O}$	I/O Capacitance		20	pF	

NOTE:
5. Sampled not 100% tested.

A.C. CHARACTERISTICS

T_A = 0° to 70°C, V_{CC} = +5V ±10%, GND = 0V

T_A = −40°C to +85°C for Extended Temperature

BUS PARAMETERS

READ CYCLE

Symbol	Parameter	82C55A-2		Units	Test Conditions
		Min	Max		
t_{AR}	Address Stable Before $\overline{RD}\downarrow$	0		ns	
t_{RA}	Address Hold Time After $\overline{RD}\uparrow$	0		ns	
t_{RR}	$\overline{RD}$ Pulse Width	150		ns	
t_{RD}	Data Delay from $\overline{RD}\downarrow$		120	ns	
t_{DF}	$\overline{RD}\uparrow$ to Data Floating	10	75	ns	
t_{RV}	Recovery Time between $\overline{RD}/\overline{WR}$	200		ns	

WRITE CYCLE

Symbol	Parameter	82C55A-2		Units	Test Conditions
		Min	Max		
t_{AW}	Address Stable Before $\overline{WR}\downarrow$	0		ns	
t_{WA}	Address Hold Time After $\overline{WR}\uparrow$	20		ns	Ports A & B
		20		ns	Port C
t_{WW}	$\overline{WR}$ Pulse Width	100		ns	
t_{DW}	Data Setup Time Before $\overline{WR}\uparrow$	100		ns	
t_{WD}	Data Hold Time After $\overline{WR}\uparrow$	30		ns	Ports A & B
		30		ns	Port C

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias....0°C to + 70°C
Storage Temperature .....− 65°C to + 150°C
Supply Voltage .....− 0.5 to + 8.0V
Operating Voltage .....+ 4V to + 7V
Voltage on any Input.....GND−2V to + 6.5V
Voltage on any Output ..GND−0.5V to V_{CC} + 0.5V
Power Dissipation .....1 Watt

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

D.C. CHARACTERISTICS

T_A = 0°C to 70°C, V_{CC} = +5V ±10%, GND = 0V (T_A = −40°C to +85°C for Extended Temperture)

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	−0.5	0.8	V	
V_{IH}	Input High Voltage	2.0	V_{CC}	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 2.5 mA
V_{OH}	Output High Voltage	3.0 V_{CC} − 0.4		V V	I_{OH} = −2.5 mA I_{OH} = −100 µA
I_{IL}	Input Leakage Current		±1	µA	V_{IN} = V_{CC} to 0V (Note 1)
I_{OFL}	Output Float Leakage Current		±10	µA	V_{IN} = V_{CC} to 0V (Note 2)
I_{DAR}	Darlington Drive Current	±2.5	(Note 4)	mA	Ports A, B, C R_{ext} = 500Ω V_{ext} = 1.7V
I_{PHL}	Port Hold Low Leakage Current	+50	+300	µA	V_{OUT} = 1.0V Port A only
I_{PHH}	Port Hold High Leakage Current	−50	−300	µA	V_{OUT} = 3.0V Ports A, B, C
I_{PHLO}	Port Hold Low Overdrive Current	−350		µA	V_{OUT} = 0.8V
I_{PHHO}	Port Hold High Overdrive Current	+350		µA	V_{OUT} = 3.0V
I_{CC}	V_{CC} Supply Current		10	mA	(Note 3)
I_{CCSB}	V_{CC} Supply Current-Standby		10	µA	V_{CC} = 5.5V V_{IN} = V_{CC} or GND Port Conditions If I/P = Open/High O/P = Open Only With Data Bus = High/Low $\overline{CS}$ = High Reset = Low Pure Inputs = Low/High

NOTES:
1. Pins A_1, A_0, $\overline{CS}$, $\overline{WR}$, $\overline{RD}$, Reset.
2. Data Bus; Ports B, C.
3. Outputs open.
4. Limit output current to 4.0 mA.

© INTEL CORPORATION, 1983

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Reading Port C Status

In Mode 0, Port C transfers data to or from the peripheral device. When the 82C55A is programmed to function in Modes 1 or 2, Port C generates or accepts "hand-shaking" signals with the peripheral device. Reading the contents of Port C allows the programmer to test or verify the "status" of each peripheral device and change the program flow accordingly.

There is no special instruction to read the status information from Port C. A normal read operation of Port C is executed to perform this function.

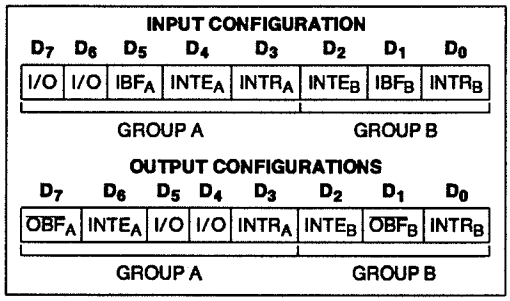


Figure 17a. MODE 1 Status Word Format

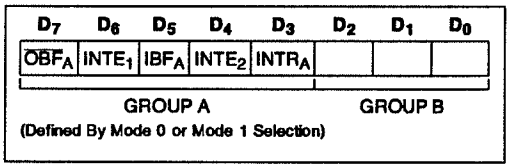


Figure 17b. MODE 2 Status Word Format

Interrupt Enable Flag	Position	Alternate Port C Pin Signal (Mode)
INTE B	PC2	$\overline{ACK}_B$ (Output Mode 1) or $\overline{STB}_B$ (Input Mode 1)
INTE A2	PC4	$\overline{STB}_A$ (Input Mode 1 or Mode 2)
INTE A1	PC6	$\overline{ACK}_A$ (Output Mode 1 or Mode 2

Figure 18. Interrupt Enable Flags in Modes 1 and 2

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Mode Definition Summary

	MODE 0	
	IN	OUT
PA_0	IN	OUT
PA_1	IN	OUT
PA_2	IN	OUT
PA_3	IN	OUT
PA_4	IN	OUT
PA_5	IN	OUT
PA_6	IN	OUT
PA_7	IN	OUT
PB_0	IN	OUT
PB_1	IN	OUT
PB_2	IN	OUT
PB_3	IN	OUT
PB_4	IN	OUT
PB_5	IN	OUT
PB_6	IN	OUT
PB_7	IN	OUT
PC_0	IN	OUT
PC_1	IN	OUT
PC_2	IN	OUT
PC_3	IN	OUT
PC_4	IN	OUT
PC_5	IN	OUT
PC_6	IN	OUT
PC_7	IN	OUT

MODE 1	
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
IN	OUT
$INTR_B$	$INTR_B$
IBF_B	$\overline{OBF}_B$
$\overline{STB}_B$	$\overline{ACK}_B$
$INTR_A$	$INTR_A$
$\overline{STB}_A$	I/O
IBF_A	I/O
I/O	$\overline{ACK}_A$
I/O	$\overline{OBF}_A$

MODE 2
GROUP A ONLY
↔
↔
↔
↔
↔
↔
↔
↔
—
—
—
—
—
—
—
—
I/O
I/O
I/O
$INTR_A$
$\overline{STB}_A$
IBF_A
$\overline{ACK}_A$
$\overline{OBF}_A$

MODE 0 OR MODE 1 ONLY

Special Mode Combination Considerations

There are several combinations of modes possible. For any combination, some or all of the Port C lines are used for control or status. The remaining bits are either inputs or outputs as defined by a "Set Mode" command.

During a read of Port C, the state of all the Port C lines, except the $\overline{ACK}$ and $\overline{STB}$ lines, will be placed on the data bus. In place of the $\overline{ACK}$ and $\overline{STB}$ line states, flag status will appear on the data bus in the PC2, PC4, and PC6 bit positions as illustrated by Figure 18.

Through a "Write Port C" command, only the Port C pins programmed as outputs in a Mode 0 group can be written. No other pins can be affected by a "Write Port C" command, nor can the interrupt enable flags be accessed. To write to any Port C output programmed as an output in a Mode 1 group or to change an interrupt enable flag, the "Set/Reset Port C Bit" command must be used.

With a "Set/Reset Port C Bit" command, any Port C line programmed as an output (including INTR, IBF and $\overline{OBF}$) can be written, or an interrupt enable flag can be either set or reset. Port C lines programmed as inputs, including $\overline{ACK}$ and $\overline{STB}$ lines, associated with Port C are not affected by a "Set/Reset Port C Bit" command. Writing to the corresponding Port C bit positions of the $\overline{ACK}$ and $\overline{STB}$ lines with the "Set/Reset Port C Bit" command will affect the Group A and Group B interrupt enable flags, as illustrated in Figure 18.

Current Drive Capability

Any output on Port A, B or C can sink or source 2.5 mA. This feature allows the 82C55A to directly drive Darlington type drivers and high-voltage displays that require such sink or source current.

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

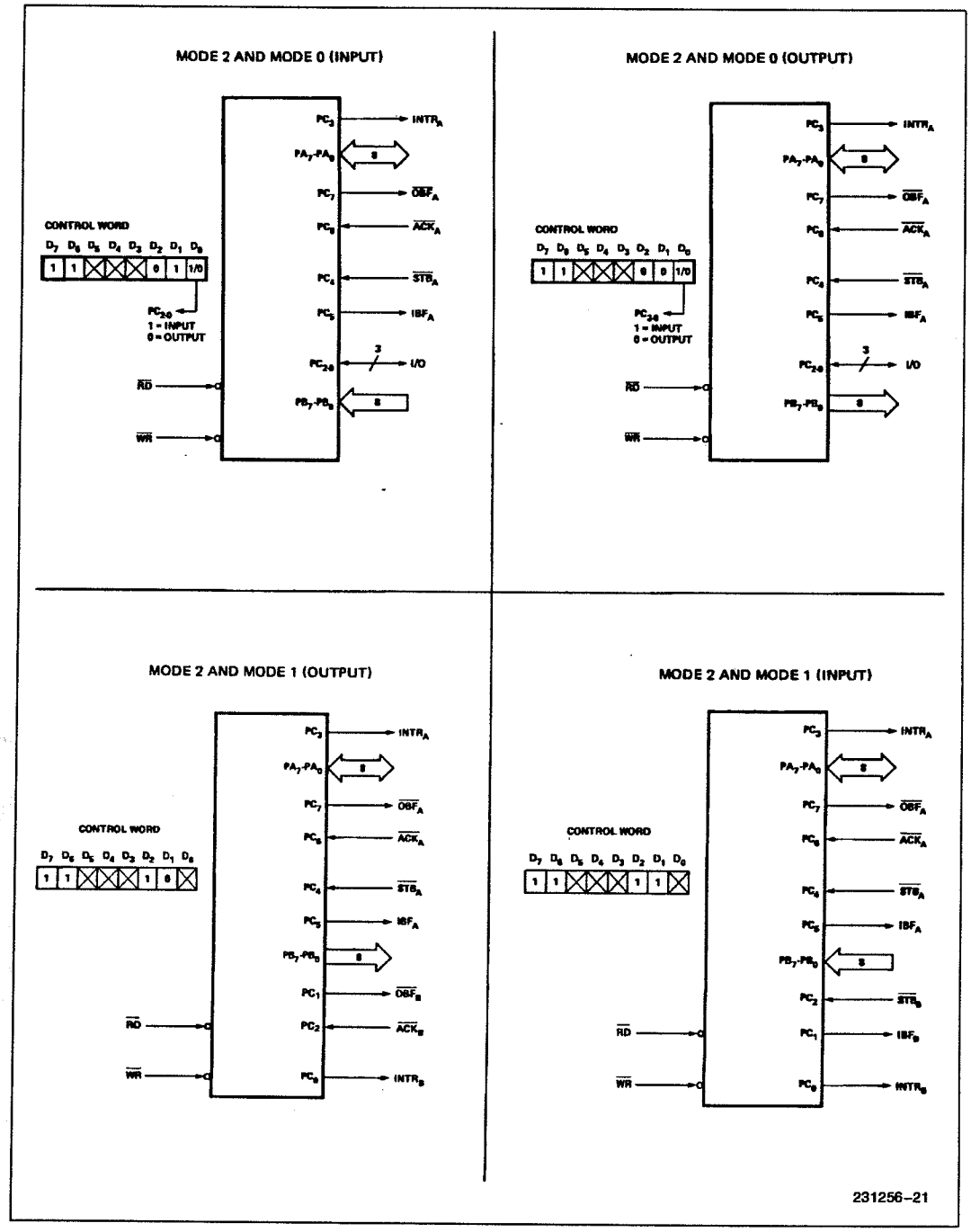


Figure 16. MODE 1/4 Combinations

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

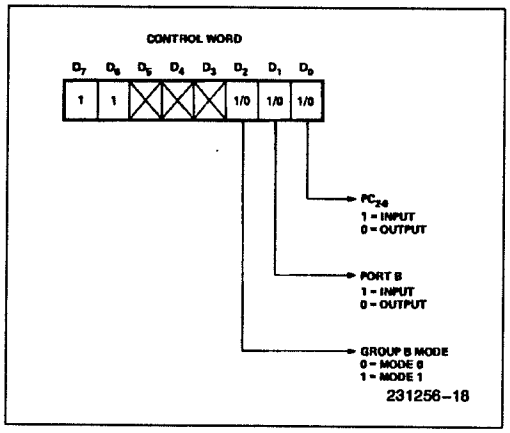


Figure 13. MODE Control Word

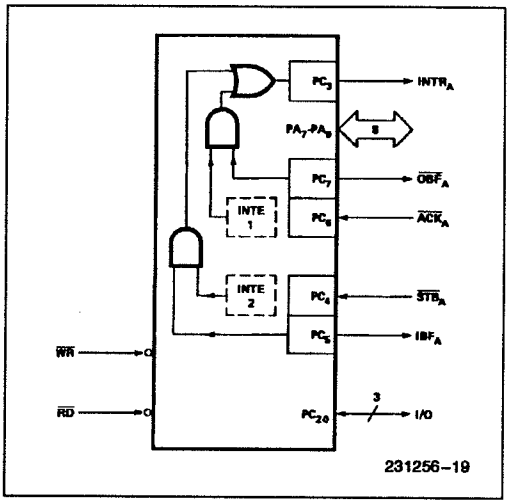


Figure 14. MODE 2

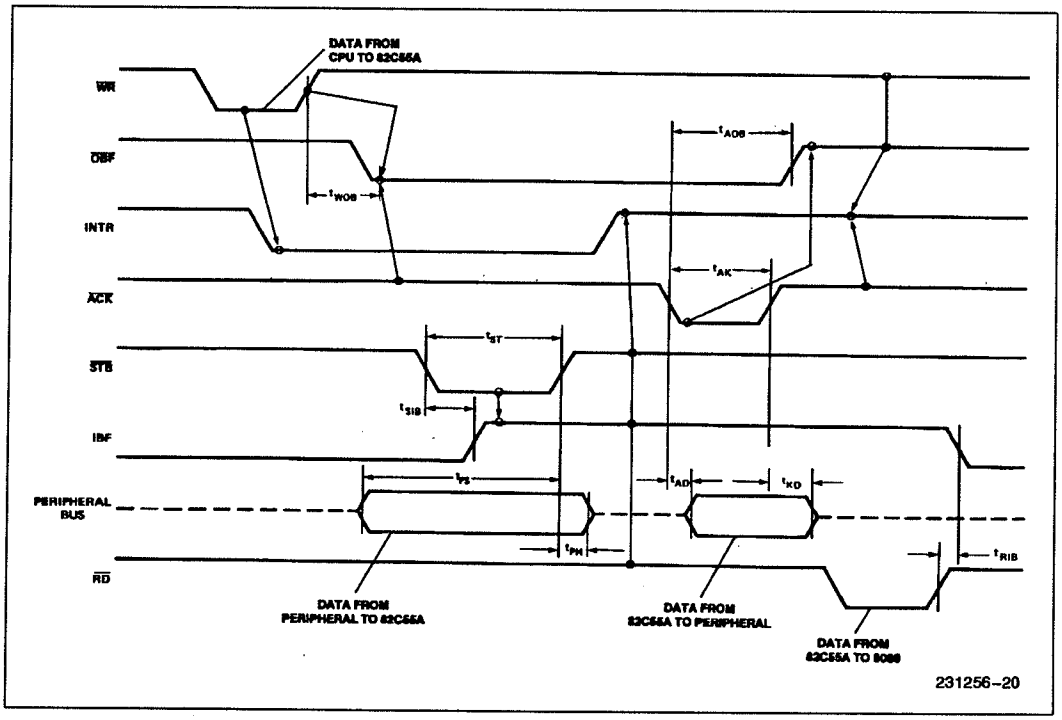


Figure 15. MODE 2 (Bidirectional)

NOTE:
Any sequence where $\overline{WR}$ occurs before $\overline{ACK}$, and $\overline{STB}$ occurs before $\overline{RD}$ is permissible.
($INTR = IBF \bullet \overline{MASK} \bullet \overline{STB} \bullet \overline{RD} + \overline{OBF} \bullet \overline{MASK} \bullet \overline{ACK} \bullet \overline{WR}$)

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Combinations of MODE 1

Port A and Port B can be individually defined as input or output in Mode 1 to support a wide variety of strobed I/O applications.

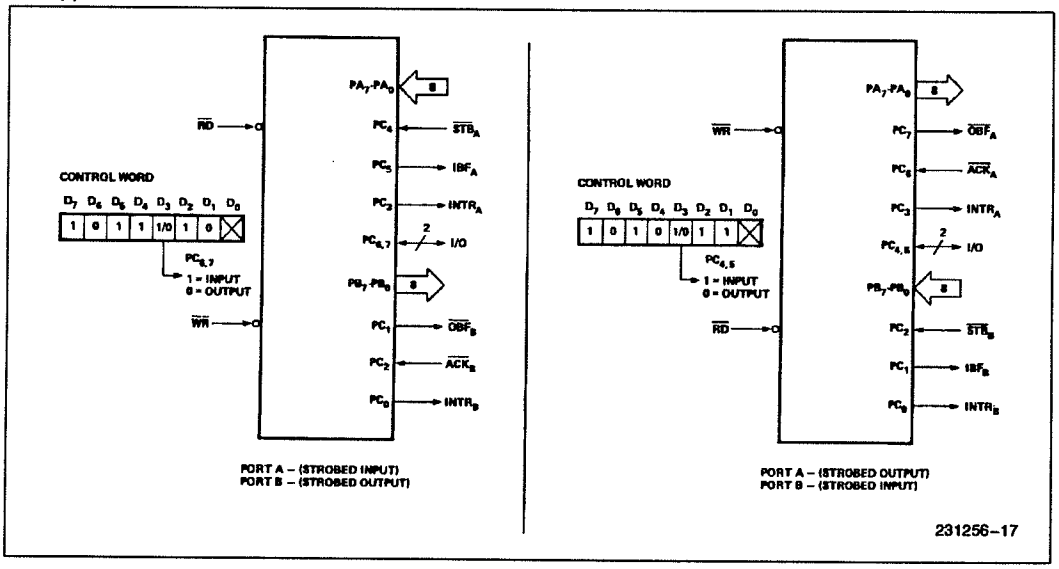


Figure 12. Combinations of MODE 1

Operating Modes

MODE 2 (Strobed Bidirectional Bus I/O). This functional configuration provides a means for communicating with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bidirectional bus I/O). "Handshaking" signals are provided to maintain proper bus flow discipline in a similar manner to MODE 1. Interrupt generation and enable/disable functions are also available.

MODE 2 Basic Functional Definitions:

- Used in Group A **only**.
- One 8-bit, bi-directional bus port (Port A) and a 5-bit control port (Port C).
- Both inputs and outputs are latched.
- The 5-bit control port (Port C) is used for control and status for the 8-bit, bi-directional bus port (Port A).

Bidirectional Bus I/O Control Signal Definition

INTR (Interrupt Request). A high on this output can be used to interrupt the CPU for input or output operations.

Output Operations

$\overline{OBF}$ (Output Buffer Full). The $\overline{OBF}$ output will go "low" to indicate that the CPU has written data out to port A.

$\overline{ACK}$ (Acknowledge). A "low" on this input enables the tri-state output buffer of Port A to send out the data. Otherwise, the output buffer will be in the high impedance state.

INTE 1 (The INTE Flip-Flop Associated with $\overline{OBF}$). Controlled by bit set/reset of PC_6.

Input Operations

$\overline{STB}$ (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F). A "high" on this output indicates that data has been loaded into the input latch.

INTE 2 (The INTE Flip-Flop Associated with IBF). Controlled by bit set/reset of PC_4.

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Output Control Signal Definition

$\overline{OBF}$ (Output Buffer Full F/F). The $\overline{OBF}$ output will go "low" to indicate that the CPU has written data out to the specified port. The $\overline{OBF}$ F/F will be set by the rising edge of the $\overline{WR}$ input and reset by $\overline{ACK}$ input being low.

$\overline{ACK}$ (Acknowledge Input). A "low" on this input informs the 82C55A that the data from Port A or Port B has been accepted. In essence, a response from the peripheral device indicating that it has received the data output by the CPU.

INTR (Interrupt Request). A "high" on this output can be used to interrupt the CPU when an output device has accepted data transmitted by the CPU. INTR is set when $\overline{ACK}$ is a "one", $\overline{OBF}$ is a "one" and INTE is a "one". It is reset by the falling edge of $\overline{WR}$.

INTE A

Controlled by bit set/reset of PC_6.

INTE B

Controlled by bit set/reset of PC_2.

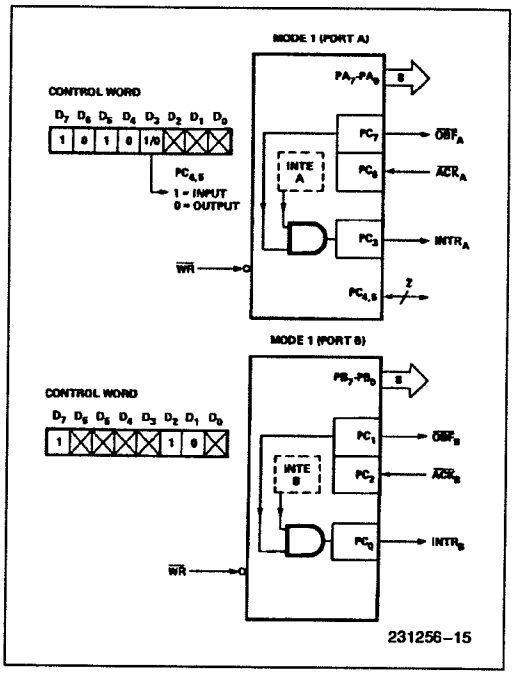


Figure 10. MODE 1 Output

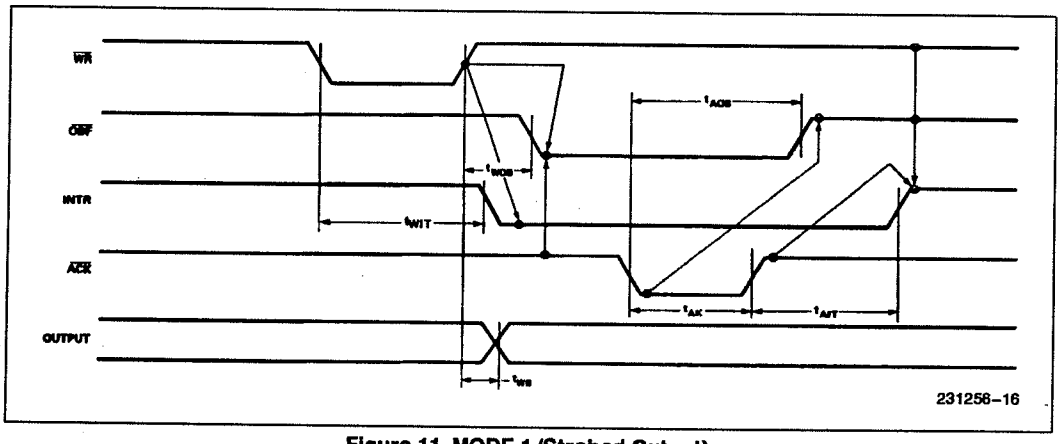


Figure 11. MODE 1 (Strobed Output)

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Input Control Signal Definition

$\overline{STB}$ (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F)

A "high" on this output indicates that the data has been loaded into the input latch; in essence, an acknowledgement. IBF is set by $\overline{STB}$ input being low and is reset by the rising edge of the $\overline{RD}$ input.

INTR (Interrupt Request)

A "high" on this output can be used to interrupt the CPU when an input device is requesting service. INTR is set by the $\overline{STB}$ is a "one", IBF is a "one" and INTE is a "one". It is reset by the falling edge of $\overline{RD}$. This procedure allows an input device to request service from the CPU by simply strobing its data into the port.

INTE A

Controlled by bit set/reset of PC_4.

INTE B

Controlled by bit set/reset of PC_2.

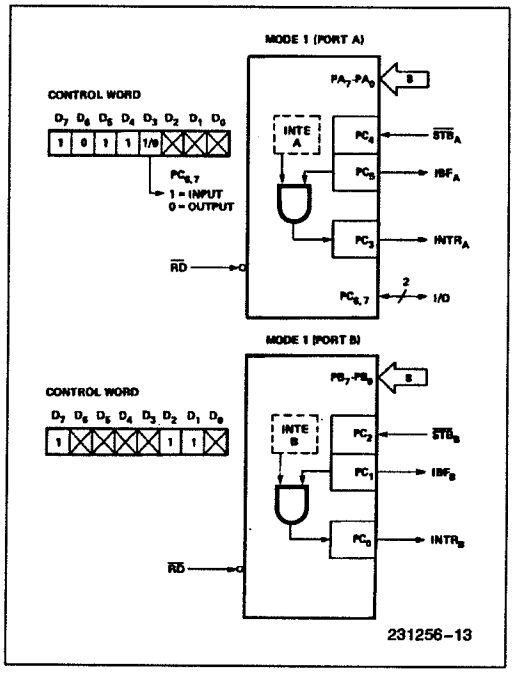


Figure 8. MODE 1 Input

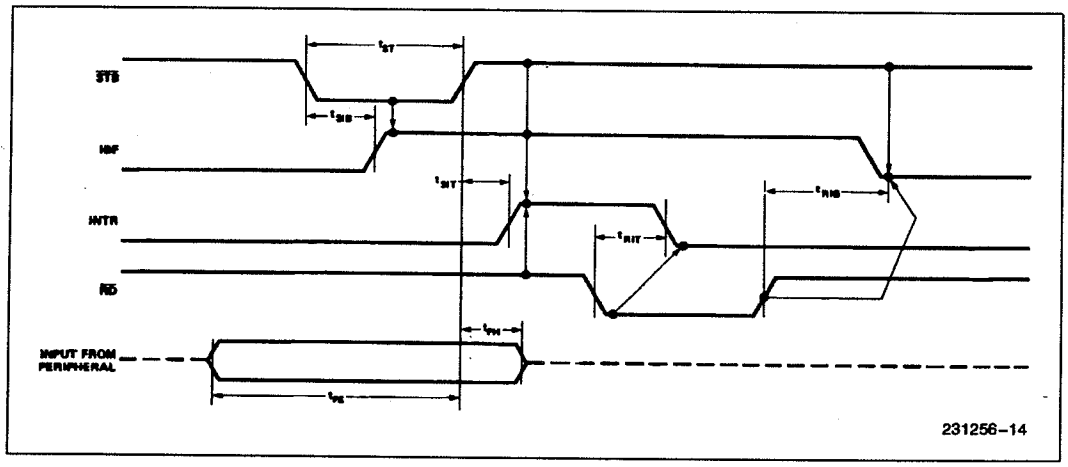
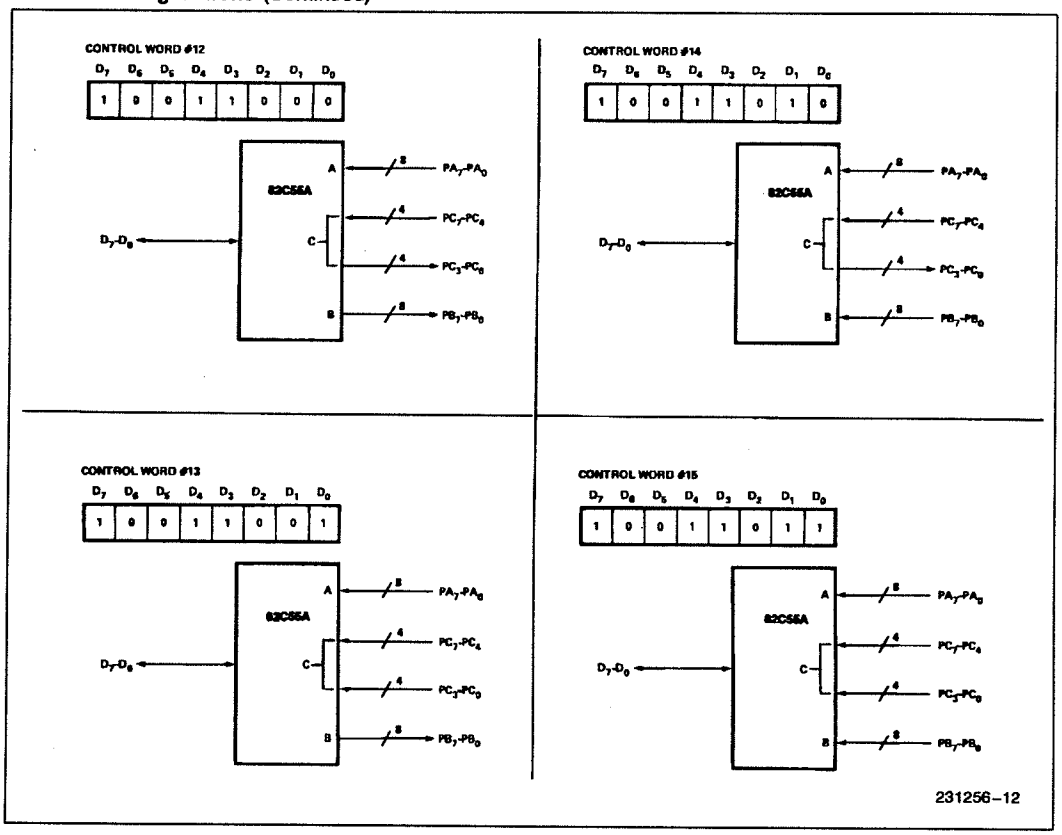


Figure 9. MODE 1 (Strobed Input)

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

MODE 0 Configurations (Continued)



Operating Modes

MODE 1 (Strobed Input/Output). This functional configuration provides a means for transferring I/O data to or from a specified port in conjunction with strobes or "handshaking" signals. In mode 1, Port A and Port B use the lines on Port C to generate or accept these "handshaking" signals.

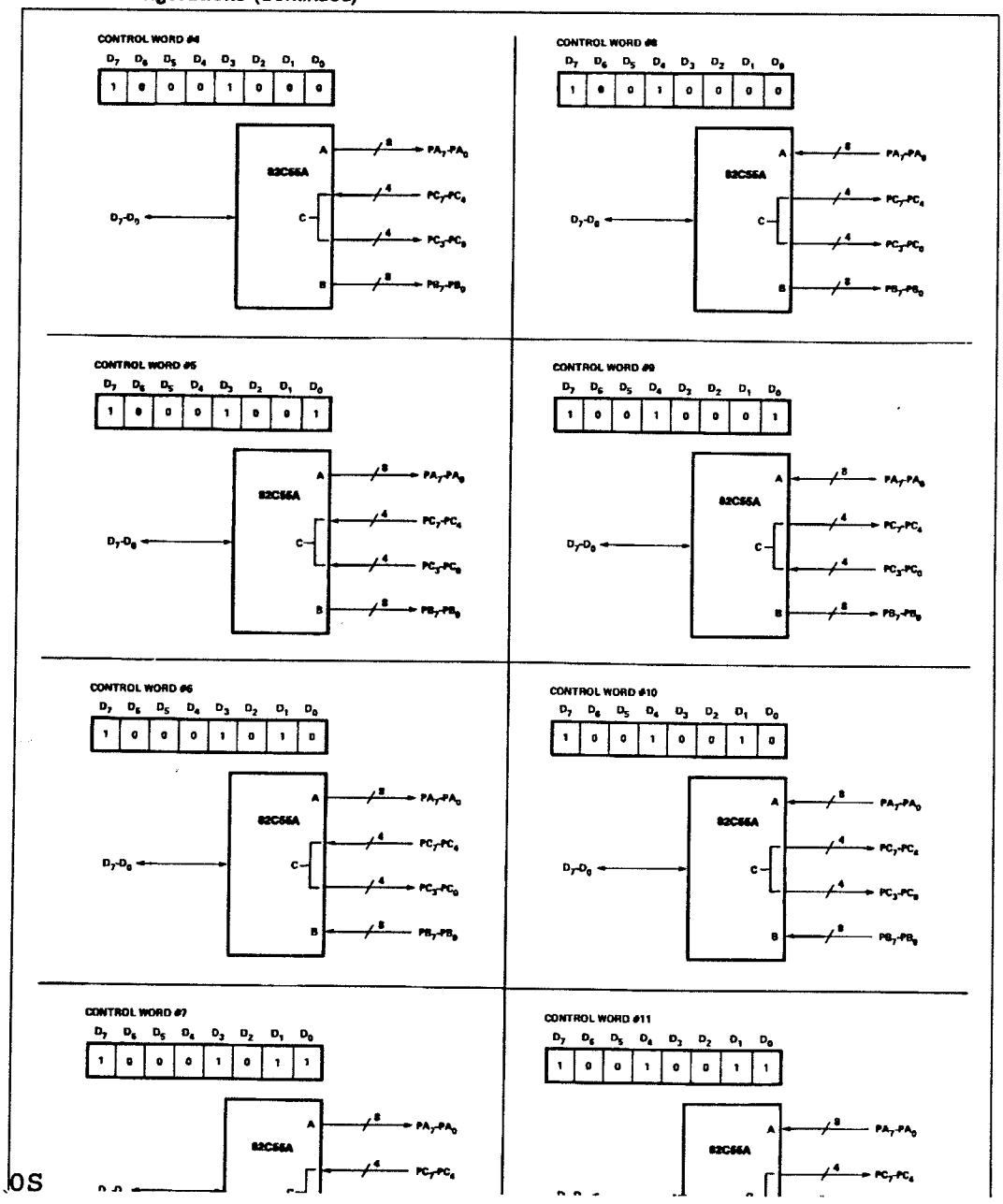
Mode 1 Basic functional Definitions:

- Two Groups (Group A and Group B).
- Each group contains one 8-bit data port and one 4-bit control/data port.
- The 8-bit data port can be either input or output Both inputs and outputs are latched.
- The 4-bit port is used for control and status of the 8-bit data port.

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

MODE 0 Configurations (Continued)

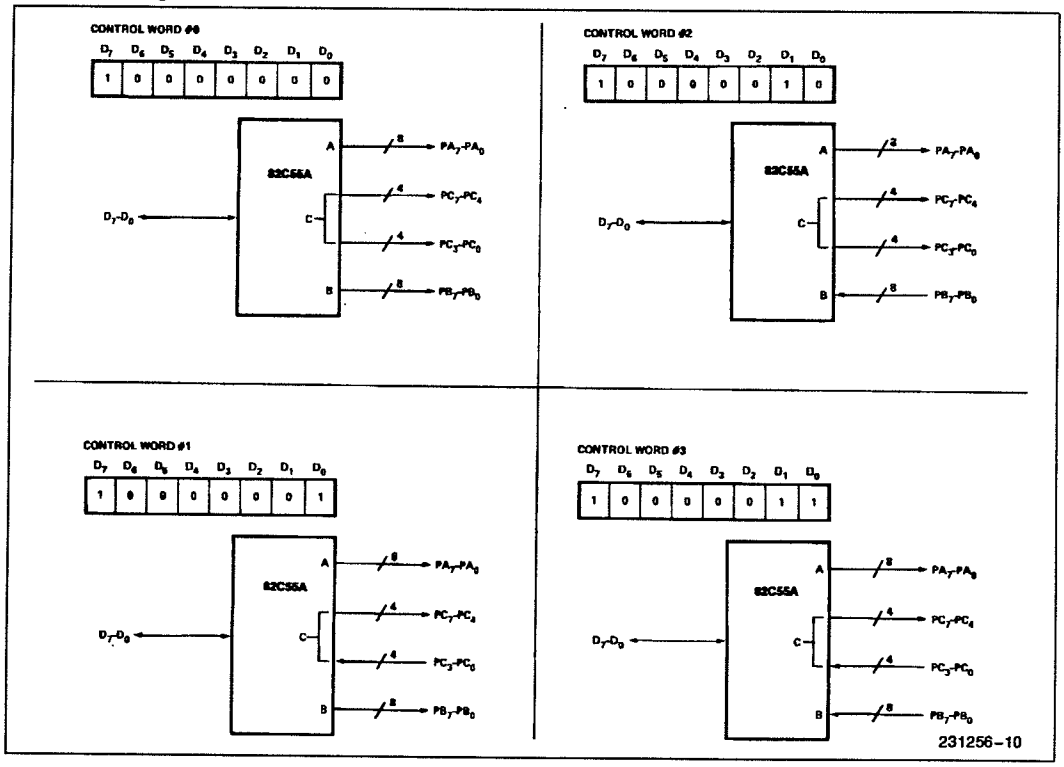


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

MODE 0 Port Definition

A		B		GROUP A			GROUP B	
D_4	D_3	D_1	D_0	PORT A	PORT C (UPPER)	#	PORT B	PORT C (LOWER)
0	0	0	0	OUTPUT	OUTPUT	0	OUTPUT	OUTPUT
0	0	0	1	OUTPUT	OUTPUT	1	OUTPUT	INPUT
0	0	1	0	OUTPUT	OUTPUT	2	INPUT	OUTPUT
0	0	1	1	OUTPUT	OUTPUT	3	INPUT	INPUT
0	1	0	0	OUTPUT	INPUT	4	OUTPUT	OUTPUT
0	1	0	1	OUTPUT	INPUT	5	OUTPUT	INPUT
0	1	1	0	OUTPUT	INPUT	6	INPUT	OUTPUT
0	1	1	1	OUTPUT	INPUT	7	INPUT	INPUT
1	0	0	0	INPUT	OUTPUT	8	OUTPUT	OUTPUT
1	0	0	1	INPUT	OUTPUT	9	OUTPUT	INPUT
1	0	1	0	INPUT	OUTPUT	10	INPUT	OUTPUT
1	0	1	1	INPUT	OUTPUT	11	INPUT	INPUT
1	1	0	0	INPUT	INPUT	12	OUTPUT	OUTPUT
1	1	0	1	INPUT	INPUT	13	OUTPUT	INPUT
1	1	1	0	INPUT	INPUT	14	INPUT	OUTPUT
1	1	1	1	INPUT	INPUT	15	INPUT	INPUT

MODE 0 Configurations



© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

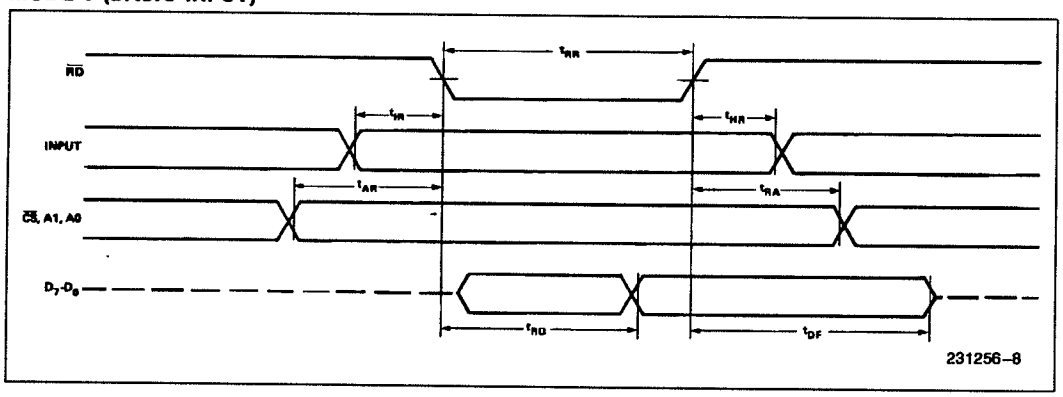
Operating Modes

Mode 0 (Basic Input/Output). This functional configuration provides simple input and output operations for each of the three ports. No "handshaking" is required, data is simply written to or read from a specified port.

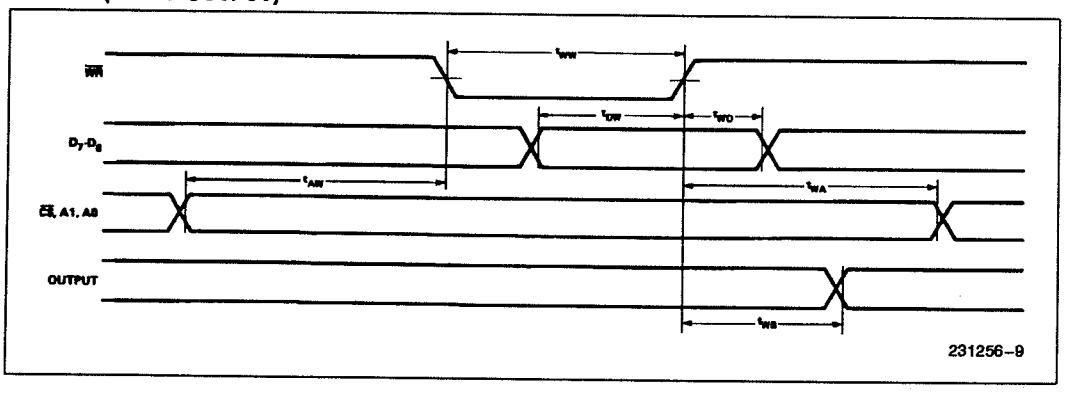
Mode 0 Basic Functional Definitions:

- Two 8-bit ports and two 4-bit ports.
- Any port can be input or output.
- Outputs are latched.
- Inputs are not latched.
- 16 different Input/Output configurations are possible in this Mode.

MODE 0 (BASIC INPUT)



MODE 0 (BASIC OUTPUT)



© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

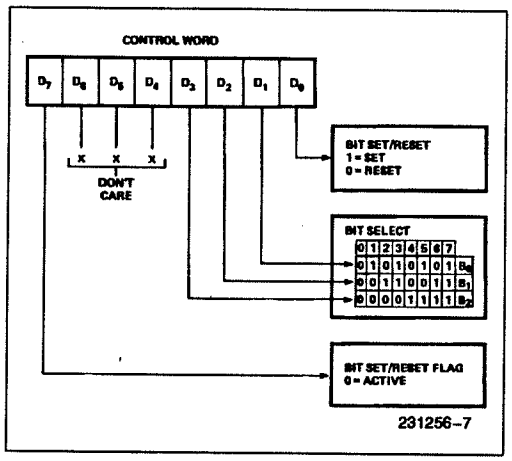


Figure 7. Bit Set/Reset Format

Interrupt Control Functions

When the 82C55A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop, using the bit set/reset function of port C.

This function allows the Programmer to disallow or allow a specific I/O device to interrupt the CPU without affecting any other device in the interrupt structure.

INTE flip-flop definition:

(BIT-SET)—INTE is SET—Interrupt enable
(BIT-RESET)—INTE is RESET—Interrupt disable

Note:
All Mask flip-flops are automatically reset during mode selection and device Reset.

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

82C55A OPERATIONAL DESCRIPTION

Mode Selection

There are three basic modes of operation that can be selected by the system software:

Mode 0 — Basic input/output
Mode 1 — Strobed Input/output
Mode 2 — Bi-directional Bus

When the reset input goes "high" all ports will be set to the input mode with all 24 port lines held at a logic "one" level by the internal bus hold devices (see Figure 4 Note). After the reset is removed the 82C55A can remain in the input mode with no additional initialization required. This eliminates the need for pullup or pulldown devices in "all CMOS" designs. During the execution of the system program, any of the other modes may be selected by using a single output instruction. This allows a single 82C55A to service a variety of peripheral devices with a simple software maintenance routine.

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance; Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

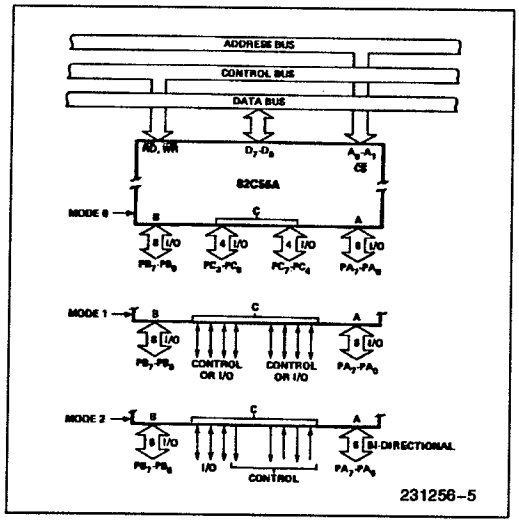


Figure 5. Basic Mode Definitions and Bus Interface

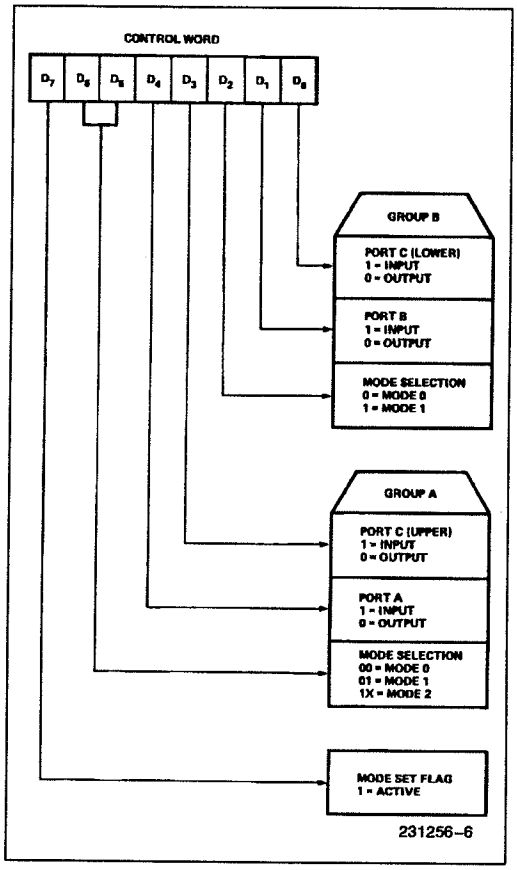


Figure 6. Mode Definition Format

The mode definitions and possible mode combinations may seem confusing at first but after a cursory review of the complete device operation a simple, logical I/O approach will surface. The design of the 82C55A has taken into account things such as efficient PC board layout, control signal definition vs PC layout and complete functional flexibility to support almost any peripheral device with no external logic. Such design represents the maximum use of the available pins.

Single Bit Set/Reset Feature

Any of the eight bits of Port C can be Set or Reset using a single OUTput instruction. This feature reduces software requirements in Control-based applications.

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were data output ports.

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

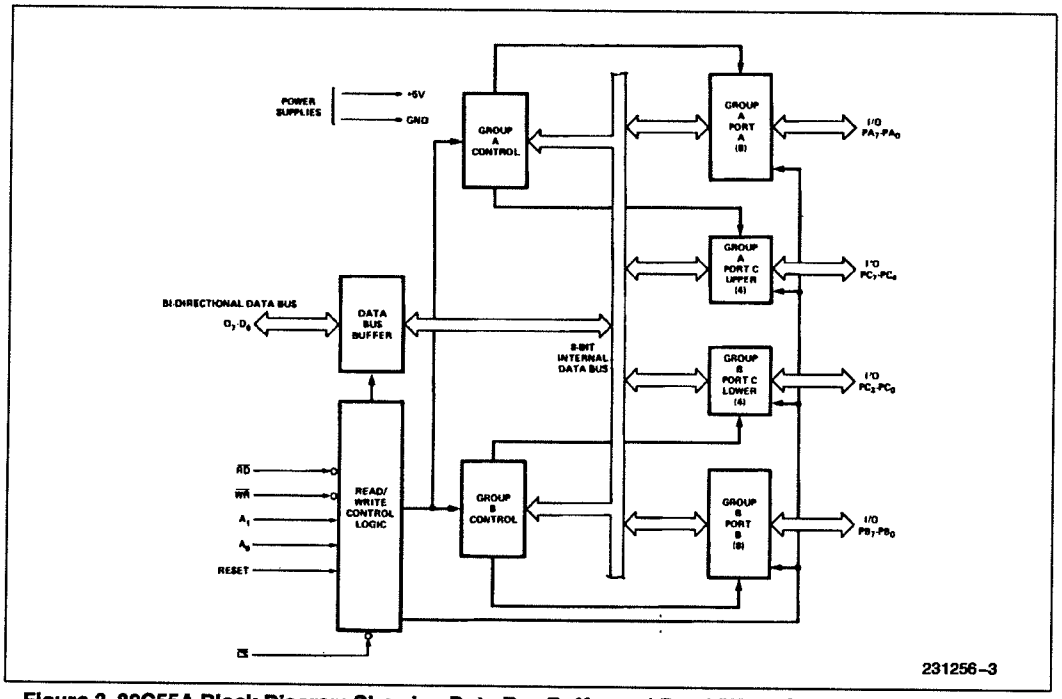
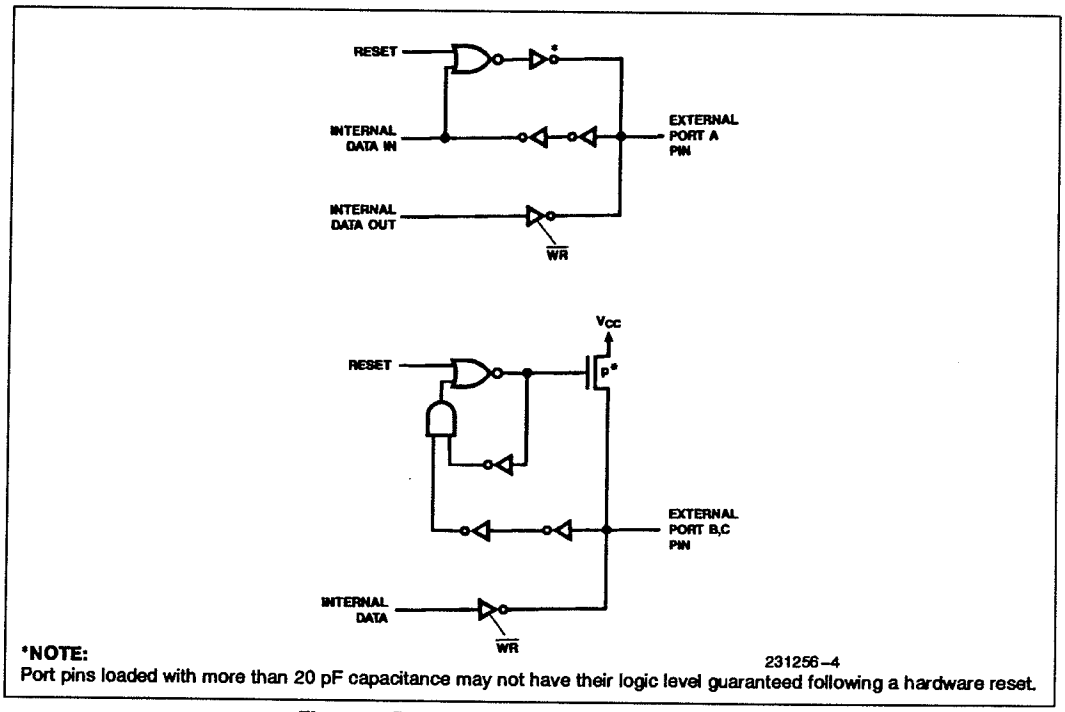


Figure 3. 82C55A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions



***NOTE:**
Port pins loaded with more than 20 pF capacitance may not have their logic level guaranteed following a hardware reset.

Figure 4. Port A, B, C, Bus-hold Configuration

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

82C55A FUNCTIONAL DESCRIPTION

General

The 82C55A is a programmable peripheral interface device designed for use in Intel microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 82C55A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interface the 82C55A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 82C55A. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 82C55A.

Each of the Control blocks (Group A and Group B) accepts "commands" from the Read/Write Control Logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

Control Group A - Port A and Port C upper (C7–C4)
Control Group B - Port B and Port C lower (C3–C0)

The control word register can be both written and read as shown in the address decode table in the pin descriptions. Figure 6 shows the control word format for both Read and Write operations. When the control word is read, bit D7 will always be a logic "1", as this implies control word mode information.

Ports A, B, and C

The 82C55A contains three 8-bit ports (A, B, and C). All can be configured in a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 82C55A.

Port A. One 8-bit data output latch/buffer and one 8-bit input latch buffer. Both "pull-up" and "pull-down" bus hold devices are present on Port A.

Port B. One 8-bit data input/output latch/buffer. Only "pull-up" bus hold devices are present on Port B.

Port C. One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B. Only "pull-up" bus hold devices are present on Port C.

See Figure 4 for the bus-hold circuit configuration for Port A, B, and C.

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 1. Pin Description

Symbol	Pin Number Dip	Pin Number PLCC	Type	Name and Function
PA_{3-0}	1–4	2–5	I/O	**PORT A, PINS 0–3:** Lower nibble of an 8-bit data output latch/ buffer and an 8-bit data input latch.
$\overline{RD}$	5	6	I	**READ CONTROL:** This input is low during CPU read operations.
$\overline{CS}$	6	7	I	**CHIP SELECT:** A low on this input enables the 82C55A to respond to $\overline{RD}$ and $\overline{WR}$ signals. $\overline{RD}$ and WR are ignored otherwise.
GND	7	8		**System Ground**
A_{1-0}	8–9	9–10	I	**ADDRESS:** These input signals, in conjunction $\overline{RD}$ and $\overline{WR}$, control the selection of one of the three ports or the control word registers.

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	Input Operation (Read)
0	0	0	1	0	Port A - Data Bus
0	1	0	1	0	Port B - Data Bus
1	0	0	1	0	Port C - Data Bus
1	1	0	1	0	Control Word - Data Bus
					Output Operation (Write)
0	0	1	0	0	Data Bus - Port A
0	1	1	0	0	Data Bus - Port B
1	0	1	0	0	Data Bus - Port C
1	1	1	0	0	Data Bus - Control
					Disable Function
X	X	X	X	1	Data Bus - 3 - State
X	X	1	1	0	Data Bus - 3 - State

PC_{7-4}	10–13	11,13–15	I/O	**PORT C, PINS 4–7:** Upper nibble of an 8-bit data output latch/ buffer and an 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B.
PC_{0-3}	14–17	16–19	I/O	**PORT C, PINS 0–3:** Lower nibble of Port C.
PB_{0-7}	18–25	20–22, 24–28	I/O	**PORT B, PINS 0–7:** An 8-bit data output latch/buffer and an 8-bit data input buffer.
V_{CC}	26	29		**SYSTEM POWER:** + 5V Power Supply.
D_{7-0}	27–34	30–33, 35–38	I/O	**DATA BUS:** Bi-directional, tri-state data bus lines, connected to system data bus.
RESET	35	39	I	**RESET:** A high on this input clears the control register and all ports are set to the input mode.
$\overline{WR}$	36	40	I	**WRITE CONTROL:** This input is low during CPU write operations.
PA_{7-4}	37–40	41–44	I/O	**PORT A, PINS 4–7:** Upper nibble of an 8-bit data output latch/ buffer and an 8-bit data input latch.
NC		1, 12, 23, 34		No Connect

© INTEL CORPORATION, 1993

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

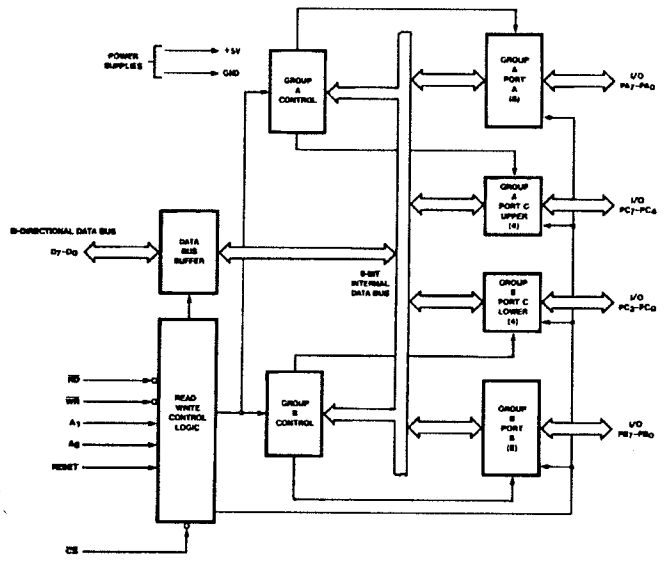
82C55A
CHMOS PROGRAMMABLE PERIPHERAL INTERFACE

- **Compatible with all Intel and Most Other Microprocessors**
- **High Speed, "Zero Wait State" Operation with 8 MHz 8086/88 and 80186/188**
- **24 Programmable I/O Pins**
- **Low Power CHMOS**
- **Completely TTL Compatible**
- **Control Word Read-Back Capability**
- **Direct Bit Set/Reset Capability**
- **2.5 mA DC Drive Capability on all I/O Port Outputs**
- **Available in 40-Pin DIP and 44-Pin PLCC**
- **Available in EXPRESS**
  - **—Standard Temperature Range**
  - **—Extended Temperature Range**

The Intel 82C55A is a high-performance, CHMOS version of the industry standard 8255A general purpose programmable I/O device which is designed for use with all Intel and most other microprocessors. It provides 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. The 82C55A is pin compatible with the NMOS 8255A and 8255A-5.

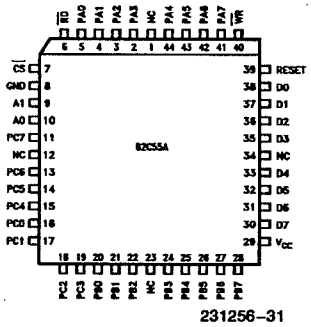
In MODE 0, each group of 12 I/O pins may be programmed in sets of 4 and 8 to be inputs or outputs. In MODE 1, each group may be programmed to have 8 lines of input or output. 3 of the remaining 4 pins are used for handshaking and interrupt control signals. MODE 2 is a strobed bi-directional bus configuration.

The 82C55A is fabricated on Intel's advanced CHMOS III technology which provides low power consumption with performance equal to or greater than the equivalent NMOS product. The 82C55A is available in 40-pin DIP and 44-pin plastic leaded chip carrier (PLCC) packages.

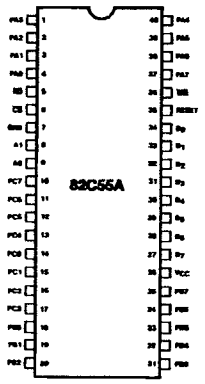


231256-1

Figure 1. 82C55A Block Diagram



231256-31



231256-2

Figure 2. 82C55A Pinout

Diagrams are for pin reference only. Package sizes are not to scale.

Intel Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in an Intel product. No other circuit patent licenses are implied. Information contained herein supersedes previously published specifications on these devices from Intel.
© INTEL CORPORATION, 1993
September 1987 Order Number: 231256-004

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

APPENDIX A

Intel 82C55A Data Sheet

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The term "scan" is sometimes confusing because it refers to only a single pass once through all the channels. A frame can have one or more multichannel scans.

8. Under "(T) Set Trigger Source", the trigger rate must be entered properly. It may default to zero and should be set to a correct value.

9. In future hardware revisions of the PCI-416, Bus Mastering is performed in a slightly different way so as to allow control of A/D transfers if a bus master cycle has not completed. Contact DATEL for details.

10. PCI-416SET does many things but some operations require software written by the user. DATEL occasionally encounters difficulties with programmers who are unfamiliar with *hardware registers*. While the PCI-416SRC source code encapsulates many register functions, there are situations not anticipated by PCI-416SRC when the raw registers must be directly manipulated. These include bit twiddling, byte or word swapping, shifting, sign testing, field masking and bit toggling among others.

Many registers are read or write only or have dead bits. A few registers disappear (change value) after you read or write them and you only have one chance. Still others cannot be read until other status bits are valid. Write only registers may require a memory "shadow" register to remember what you wrote. Some registers must be accessed slowly. And some registers require a defined sequence of reads or writes for valid information.

A difficult concept for non-hardware programmers is the precise *timing* of events. Unlike typical data processing operations, the software cannot directly control some aspects of board timing and the code must keep up with complex timing events occurring on the board. There are flag bits in registers to indicate this timing "progress". Much of this can be done in "C" but sometimes assembly language is better. Get expert help if you do understand how hardware registers work. They are not exactly like memory!

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3. When the (A)quire data function is started, the following message,

"OVERFLOW PT FIFO IS FULL"

indicates that the Pass-Through FIFO memory in the S5933 PCI controller (not the A/D FIFO) lost A/D data, probably because the A/D was too fast for the host to read. This could be a slow PCI bus (and very fast A/D) or the host software is taking too long.

4. The (D)Mux data file function asks how many channels are requested (it does not automatically know how many). It then creates N files, one for each channel such that all data for a single channel is in only one file. This function operates purely on the data file and takes place after all A/D samples have been saved to the single primary file.

5. Use the (D)isplay data file function on PC416SET to view the stored A/D samples. A typical hexadecimal display line might be:

0002 0FFF 0001 0800

This is the *display* corresponding to four sequential 12-bit A/D samples, each 16 bits wide. For 12-bit data, the bits are right-justified. If the data is unipolar 12-bits (as shown here), the top nybble will be zero. For bipolar 12 or 14-bit A/D's, the top nybble contains the sign bit extended from the A/D most significant bit. The second sample listed above shows all 12 A/D bits at logic "1" (i.e., full scale). The third sample is +1 count above minus full scale. The fourth sample shows the A/D at exactly half scale.

Note that if you use the raw *file* output or read data from memory loaded by the PCI-416, data is stored in Intel "little endian" format. The least significant byte is LOWEST in memory addressing or FIRST in the file. For example, the first few samples shown above would be stored on disk as:

02 00 FF 0F 01 00 00 08 (low byte, high byte, low byte, ... toward ascending memory)

The data in the saved file is always the raw binary data straight from the A/D FIFO (or directly from the memory without further manipulation). If it is multichannel, each sample will advance to the next channel, modulo to the total channel capacity.

The display capabilities in PC-416SET should be considered a "quick look" capability just to verify the data. You should investigate programs such as DADiSP or any good spreadsheet for far superior display, graphing and data math manipulation.

6. The data acquisition (P)arameters screen asks 3 questions. The number of frames sets a software counter on the host side which stops the system after a set number of frames. Within each frame, the hardware sample counter on the PCI-416 board must be set by the Samples per Trigger parameter.

7. Get the concept of a *frame* very clearly in mind by reading the User Manual thoroughly. A frame contains all the A/D samples specified in the local hardware sample counter. If the sample count does not align as an exact integer multiple of the channel capacity (number of channels in the A/D module), a frame will end part way through a scan of all channels.

One *trigger* starts each new frame and this frame rate is set by the trigger clock. If the sample counter has not finished counting down and a frame is still in progress, any new triggers will be ignored. Therefore the trigger rate should be set less than the A/D clock rate times the number of samples per frame.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.8 Start and Exit Mode

Start and Exit mode sets up the board from the configuration file, starts the A/D ***then exits immediately to MS-DOS*** *(or to the calling program).* Start and Exit leaves the A/D filling the FIFO, if it has already been triggered or it will wait for the first trigger. A user-written program then immediately runs after the exit back to MS-DOS. This user program collects data directly from the PCI-416 registers.

At low sample rates, the A/D FIFO may still be filling by the time the user's following data collection program runs. At high A/D clock speed, the FIFO may have already overflowed. Simply halt the A/D (using the convert enable bit), reset the FIFO, restart the A/D and wait for the next half full signal. If the trigger has not occurred, the user program can wait for the start of the ACQuire signal and will not lose any data. Start and Exit avoids having to write the user interface to run the board and provides maximum flexibility for the user program. Clearly, the user must be a competent PCI programmer however, much code can be taken from PCI-416SRC.

Note that there are many ways to operate using Start and Exit. Besides the batch file route, the user's "C" program could invoke PCI-416SET as a shelled function using the system( ) call. This makes PCI-416SET return very quickly and avoids missing the first FIFO's worth of data due to a slow file load.

8.9 Porting PCI-416SRC to Other Operating Systems

PCI-416SRC was designed for portability to other environments. In particular, the PCI-416 board can run in an Apple Power Mac, and other hosts. Here are some of the changes to consider:

- If the target environment will not use ANSI cursor controls ("VT-100" compatible), make the necessary changes. Some systems can accept the "curses" VT-100 emulation. PCI-416SET uses some color text controls unique to the PC environment.
- Most of PCI-416SRC is ANSI "C" however, the FIFO block transfers (and some other code) uses high speed assembly instructions. These must be changed for other systems.
- PCI-416SRC uses extended memory allocation BIOS functions that are unique to the PC. These must be changed for other systems.
- Other fully-compliant PCI systems should recognize the PCI BIOS calls and PCI-416SET uses the "C" INT functions for these BIOS calls. However, I/O ports, interrupts and DMA operations will be substantially different.
- PCI-416SET uses a simple graphics file display. This must be deleted or modified for other graphics environments.

8.10 Additional Notes on PCI-416SET Operation for Version 1.2: (June 1996)

1. Under the "DAQ Configuration (L)oad (S)ave (D)isplay", this information under (D)isplay is meaningless unless already initialized by a previously-saved, valid Configuration file. You may look at this information at any time and it will show the last saved Configuration.

2. Under the Configuration information, the ADM Clock is shown with a frequency and additional qualifiers after the slash as follows:

ADM Internal Clock: 50000 (Hz) / Freq. Syn. 20-40 MHz

The "/ Freq. Syn. 20-40 MHz" information describes the source of the clock and the clock selector digital MUX setting. You may also see the 82C54 Programmable Counter-Timer listed here. See the register descriptions in the User Manual.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Most options are straightforward but here is additional detail:

The DAQ Configuration is an *optional* file which stores parameters you have entered to set up PCI-416SET. You may reload a previously-saved file with (L)oad to save time configuring all screens. The (C)-Display option lists what the system thinks is the current configuration. Verify this before saving. The Configuration file system is *mandatory* if you use the command line autorun mode or start-and-exit.

The file to save A/D *data* (**not** the Configuration file) is set under Select Data File Name.

A/D data passes to either a linear buffer in host memory or a circular ring buffer using *Pretrigger* mode. These modes are selected under Set DAQ Buffering Mode.

Set Data Transfer Mode selects either bus mastering, program transfer by polling the FIFO half full bit or direct to disk. Both bus mastering and program transfer sends *all* A/D data *first* to host memory then saves it to disk. The "direct-to-disk" mode (which transfers by polling the HF) sends A/D data to disk, *as it is received.* Bus mastering offers the highest speed but may not work on all systems.

The Trigger Source can be either internal or external. Under external, make sure your external source is truly sending triggers. Otherwise operation may stall.

The MSBit Mode enables marker inputs to set the A/D Most Significant Bit. Note that the marker will have no effect on 16-bit A/D's.

The A/D can accept either external or internal start clock inputs using Set ADM Clock Source. Here again, the system may stop responding if you do not send valid external clocks.

A/D sample rates, number of samples per frame and trigger rates are all selected under Set DAQ Parameters.

After all selection items are made, start A/D sampling using Acquire Data.

After A/D data is collected, use the DATA FILE UTILITIES to view the data. Almost any spreadsheet, data base or signal processing math package will work too. The D(M)ux Data File function will separate a file of sequential multichannel A/D data into individual channel files.

The CALIBRATION section automatically programs the A/D system and presents analog inputs using a hexadecimal display. To adjust gain and zero potentiometers the system will continuously display A/D samples until stopped.

The UTILITIES section covers miscellaneous functions such as the D/A converter and parallel port. The Oscilloscope/Spectrum Analyzer presents a fundamental waveform or FFT (Fast Fourier Transform) graphics display. The arithmetic unit in the host CPU computes the FFT.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

For PCI-416SET version 1.2 or later, the **"/G"** switch invokes PCI-416SET in a "demo" mode. Most of the parameter screens are available (but cannot be modified) and A/D files can be played back through the FFT or waveform display modes, *including A/D files from boards other than the PCI-416.* Mode "/G" does not require the PCI-416 board to be installed. Except for the "G" mode the system may lock up if you attempt to take data with no board installed.

8.7 PCI-416 Top Menu

Version 1.2 of PCI-416SET incorporates more extensive screen graphics and an FFT display for A/D data than earlier versions. *These should be considered examples only.* They are not intended to replace the far more comprehensive graphics and data manipulation available in commercial systems. In particular, the graphics in V1.2 cannot be modified (axes labeling, scaling, offsets, etc.) without rewriting the source code.

Press the single letter in (parentheses) to invoke a function. Most keyboard entry items can be left as is with **ESCape**. Most items should be entered roughly in the order shown in the menu. If the system does not seem to respond in the time expected, try entering an **"X"** from the keyboard. In the interest of highest possible speed, the system does not extensively poll the keyboard or KB interrupt flags. Therefore, incorrect hardware settings may lock up the system. If you have difficulty with data transfers, try both status program mode and bus mastering. Sometimes changing host CPU speeds will show slower speeds that work and faster ones that lose data or cause data errors.

The A/D system may be fully tested under "CALIBRATION". Use this part of the menu *first* to verify operation of the board. The screen is divided into four major areas, data acquisition, data file utilities, A/D calibration and other utilities. PCI-416SET has the following screen format in version 1.2:

PCI-416SET Main Screen

```
PCI-416SET   V1.2   DATEL, Inc. © 1996

DATA ACQUISITION
DAQ Configuration  (L)oad  (S)ave  (C)-Display
(F) Select Data File Name
(B) Set DAQ Buffering Mode
(X) Set Data Transfer Mode
(T) Set Trigger Source
(R) Set MSBit Mode
(K) Set ADM Clock Source
(P) Set DAQ Parameters
(A) Acquire Data

DATA FILE UTILITIES
(D)isplay Data File  (G)raph Data File D (M)ux Data File

CALIBRATION
(1) Single Channel  (2) Multi-Channel ADM Calibration

UTILITIES
(3) Set DAC Output Voltage
(4) Access 8255 Parallel Ports
(5) Display PCI Configuration Registers
(6) Oscilloscope/Spectrum Analyzer
(Q) QUIT
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.4 Program Operation

> *Note: Your CONFIG.SYS file must include the line "DEVICE = ANSI.SYS" to use PCI-416SET*

The SET program is largely self-explanatory. The program (PC416SET.EXE) is invoked with the command line:

C:>PC416SET

Where C: is the logged on drive containing the PC416SET program.

(MS-DOS only allows 8 filename characters so the "I" is missing).

The initial screen verifies the presence of a valid PCI BIOS, finds the PCI-416 slot and determines its configuration space. It also initializes the Interrupt systems for the PCI-416. The program then reports the A/D module type and you should verify this information. If needed, PC416SET can be aborted at any time with "**x**" or **ESC**ape.

8.5 Configuration Files

For repeated experiments, after all parameters are entered on the screen, they may be saved out to a **configuration file**. This simplifies running the board again since any previous register values are automatically reloaded with a command line switch, listed below or can be loaded with a menu item within the program. To change one or more items from the previously loaded configuration, press Enter to step through each selection item not requiring a change then edit only those items needed.

Several configuration files can be saved, all with different output data files and different operating parameters if desired. Thus a library of operations can be conveniently built.

8.6 Command Line Parameters

The command line also accepts several optional parameters. By adding the switch, **"/F filename",** PCI-416SET will try to load a previously saved configuration file which automatically initializes all registers but does not start A/D sampling:

C:>PC416SET /F FILE.CFG

Where "FILE.CFG" is the user's choice of configuration file name. Adding an **"/R"** switch automatically starts A/D conversion and saves data to disk:

C:>PC416SET /F FILE.CFG /R

Using the "/R" switch, no menus are displayed. This command line can be embedded in an MS-DOS BATCH file for automated running. Always test the configuration before committing it to automated "/R" operation.

By adding the **"/X"** switch to the /F and /R, the PCI-416 will autoinitialize all registers and attempt to enter Start and Exit mode. The user's data collection program should immediately follow the Start and Exit Mode to recover data. This user program can be conveniently added to the batch file right after PC416SET. The A/D *data* file, if specified in the *configuration* file, will be ignored since the user will now directly control the PCI-416 registers and save data somewhere in his own program.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

If needed, Start and Exit mode can be set up in advance, the configuration saved to disk, then the entire operation runs automatically from a batch file.

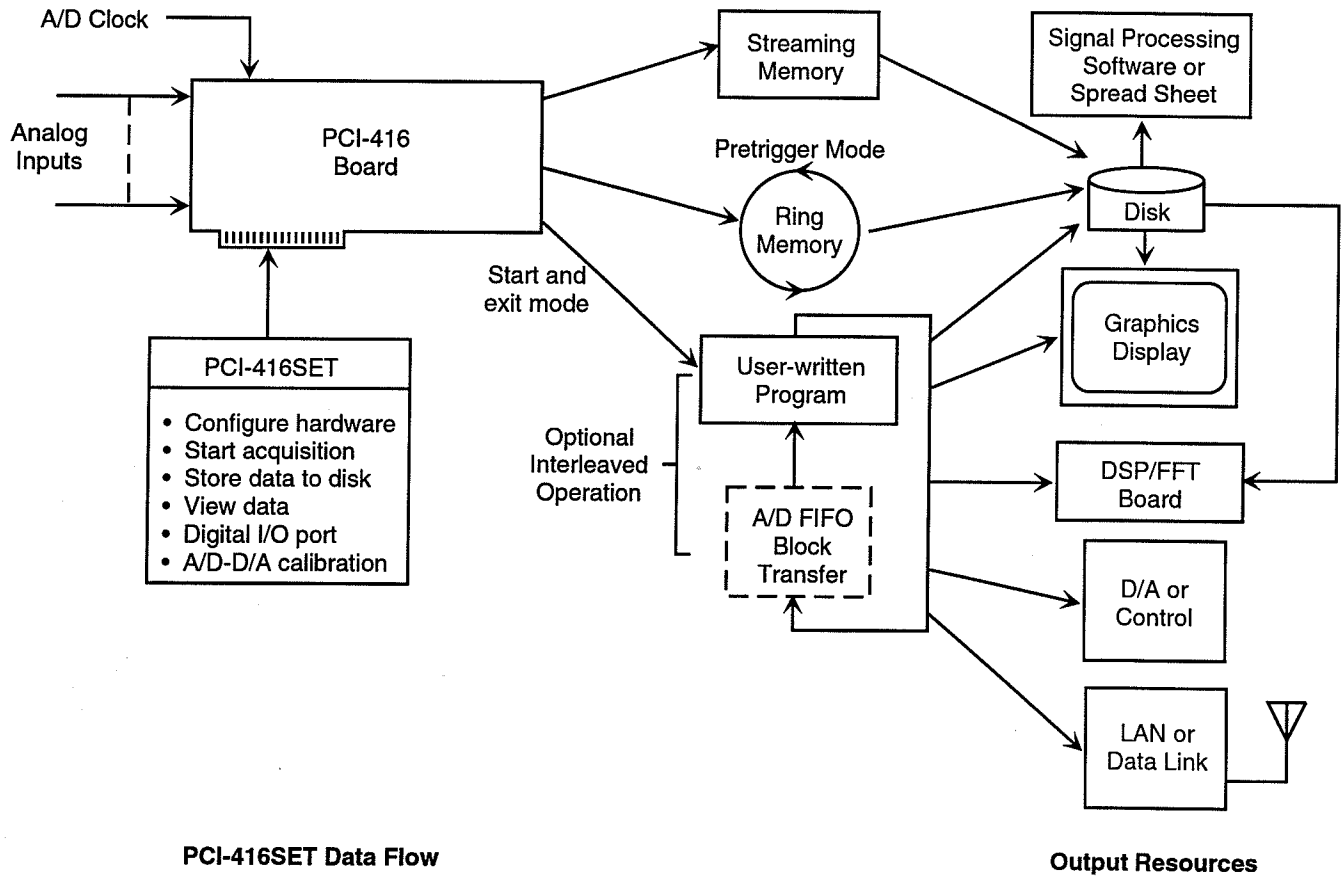


Figure 8.0 System Diagram

8.3 PCI-416 Program Design Philosophy

PCI-416SRC is professionally written and well organized in modular style for straightforward adaptation to many applications. PCI-416SRC, the full set of source files to PCI-416SET, was made as small and simple as possible as a teaching tool while achieving the full range of operating functions and all the speed available. This is done by collecting data to memory first then saving to disk *after* the A/D stops. This of course is limited by the amount of conventional or extended memory available in the host computer (but refer also to the ring buffer mode).

To keep the code as simple and fast as possible, no complex, slow windowed interface environment is employed with PCI-416SRC. Most of these windowed menu systems for MS-DOS are not portable and are not widely used or supported. Instead, PCI-416SRC uses simple "ANSI" cursor positioning which is supported under MS-DOS's ANSI.SYS device driver. Another benefit of this interface approach is that ANSI screen controls are widely used in UNIX and other operating systems in case the user needs to port PCI-416SET/SRC elsewhere. The easy user interface keeps the code and files at reasonable length and loads quickly.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 8

PCI-416 SOFTWARE

8.0 Introduction

PCI-416SET is an MS-DOS program which fully exercises the PCI-416 and collects data at the highest possible speed. The program has four major purposes:

- To verify proper operation of the board and as a preliminary diagnostic.
- To set up the board and collect A/D data to memory then disk.
- To provide an automated, batchable "front end" in "start and exit mode" to an optional user-written program.
- To serve as a comprehensive programming example.

Except for a small hexadecimal data display and simple signal display, PCI-416SET explicitly does *not* display data graphically. No attempt was made to duplicate the already powerful, external programs available to do A/D data graphics. To do so would have made PCI-416SET huge, slow and very difficult to understand. The philosophy was that users already are familiar with their favorite tools and do not wish to learn a new one.

PCI-416SET can run the board in numerous modes. However, it will not do all conceivable applications. Tasks which combine other operations between PCI-416 data block transfers will need user programming.

DATEL's PCI-416NT software for Windows NT operates very similarly to PCI-416SET but at lower speed.

8.1 "Front End" to Signal Processing Software

PCI-416SET will successfully "front end" all popular spreadsheets, data base programs, scientific/graphics analysis programs and signal processing packages in file transfer mode. This also includes all versions of these programs in Windows. Simply collect the data while in DOS mode, switch to Windows then process the output data files from PCI-416SET. Data is saved as a raw binary integer file and if needed, a simple external program available from DATEL converts PCI-416SET files to universal IEEE single precision floating point binary. If necessary, conversion is possible to floating point ASCII but at considerably more disk space and much lower speed. PCI-416SET output files do not include a file header and if needed, many signal processing programs can add the necessary header. Or it can be built with a simple "C" or BASIC program.

NOTICE: Do not operate PCI-416SET under the Windows 3.1 "MS-DOS" icon. The program may run slowly, can lock up, show limited memory or have other undefined operation. You must exit Windows 3.1 out to MS-DOS to run PCI-416SET. Windows 95 will accept PCI-416SET under the MS-DOS icon but speed may be reduced. Close all other unneeded applications.

8.2 Interleaved Operations Using "Start and Exit Mode"

While PCI-416SET does not directly interleave operation simultaneously with other programs, the source code can also be adapted to mix user functions between each A/D FIFO block transfer. The fundamental idea is to collect an A/D burst to system memory from the FIFO, manage the host buffer which is receiving FIFO data, and then, while the FIFO is filling the next A/D block (and there's a little spare time), do some other application briefly. User code must return quickly enough to prevent the A/D FIFO from overflowing. Such user programs might include buffer transfers to a DSP board, FFT graphics display or very high speed communications link. The user of course must verify that the total system can keep up with the PCI-416 without losing data.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Example #1:

Assume you want an internal trigger rate of 16kHz.

Divisor32 = 10^7Hz/(1.6 x 10^4) or,
Divisor32 = 625

This single 32-bit divisor must now be divided between the two counters. Remember that Divisor32 = Counter #0 x Counter #1. Both counters must be set to 2 or greater. Therefore, set one counter to "2" and the other counter will do further division. That leaves the Counter #0 divisor as:

Counter #0 divisor16 = 625/2 = 312 (integer division, truncated)

The closest we will come to a 16kHz trigger is 10MHz/(312 x 2) = 16.025kHz.

Now convert both the "2" divisor16 and the "312" divisor16 to 8-bit byte values using the same strategy and load into the 82C54 registers. Don't forget the propagation delays. The "2" divisor consists of MSB = 0 and the LSB = 2. The "312" divisor consists of a "1" MSB byte (312/256) and a "56" remainder for the LSB byte (312%256).

Note that while the minimum value for a whole counter is *two*, either byte in the counter can be zero if needed. In fact, if the count value is less than 256 (such as the "2" divisor example), you *must* load a zero in the MSB as a place holder. If you do not load the zero in the MSB, the counter loading sequence will get out of phase and undefined timing will result.

Example #2

Let's see what happens with a 3Hz (3 per second) trigger rate:

The divisor32 is 10MHz/3 = 3,333,333 with a remainder

Since this factor is larger than 65535, we can set one of the counters to some large divisor such as 1,000 then the other counter will be 3,333,333/1000 = 3333 (integer). The actual trigger rate will be 10MHz/(1000 x 3333) = 3.003Hz. Convert these two words into bytes and program the counters as before.

Note that you can manipulate combinations of higher or lower divisors and perhaps get even closer to our desired rate. For example with divisors of 100 and 33,333 the rate is 3.00003Hz. In general, if one divisor is near 65535, you will have better resolution.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5. Each 16-bit divisor must now be further subdivided into 8-bit values to accomodate the byte-loading nature of the 82C54 as follows:

Most Significant Byte #1 = Upper Divisor 16/256 (integer division)
Least Significant Byte #1 = Upper Divisor 16%256 (modulo division remainder)
Most Significant Byte #0 = Lower Divisor 16/256
Least Significant Byte #0 = Lower Divisor 16%256

Remember to do this *four* times for the four bytes in both counters.

We have used the "C" language convention indicating modulo division with the percent sign ("%"). This shows the remainder left from an integer division. For other languages, use whatever equivalent notation is available. Some BASIC's use "MOD". The results of *both* these operations will be whole numbers since they are about to be loaded into registers. Notice again that we do an integer division and there will be some remainder which appears in the modulo division.

If you only have an integer function (as you would in assembly language), this can also be done the hard way by computing the result back into the original:

$$\text{Remainder} = \text{Divisor16} - ((\text{Divisor16}/256) \times 256)$$

where all operations are *integer* math and are performed in the nested sequence shown. Do not simply cancel out the two 256 factors since this would avoid the remainder truncation. Hint: division by 256 is an 8-bit right shift within a 16-bit register.

Unlike the two 16-bit counters, the 8-bit counter halves within the 82C54 always operate in 256-count mode.

6. Write the resulting values into *both* the counters, in byte loading fashion. Remember the propagation delays required by the 82C54. *Remember - 32-bit DWORDs for all operations!*

How it Works

Both 16-bit values are loaded into the two counters. Since both gates are always enabled, the two counters will start counting down immediately after programming them.

For **continuous** internal trigger applications, triggers will be generated but no A/D samples will reach the FIFO memory until the A/D clock is initialized, the trigger source is selected and the A/D is enabled.

For **single** triggers, only one trigger is issued therefore, *setup all other registers before writing to the lower Counter #0*. You could also switch the system to external trigger (which is not being used), program the 82C54, then switch back to internal triggers (using the Command Register) in time to catch the first trigger pulse. If you program the system incorrectly, it is possible that no trigger will be generated and A/D samples will not load the FIFO. For initial testing, always use continuous triggers to be sure you eventually get a trigger.

In Mode 2 ("continuous trigger"), both counters automatically reload their initial values when they reach zero. So if you miss a trigger, you will catch it the next time. In Mode 4, the countdown occurs only *once* and does not reload.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Loading the 32-bit Trigger Divisor

Now that you know how to load the bytes into each counter, what values do you write? The internal trigger is derived from a 10MHz internal oscillator divided down by *two* cascaded 16-bit dividers (Counters #0 and #1) which are truncated by software. *Both counters must be programmed*, otherwise no triggers will be generated. Since each counter must be loaded in *byte* format only, *four* byte values must be developed and the computations can get interesting. Here goes:

To determine the proper codes to load,

1. Find the 32-bit trigger rate divisor you need by dividing 10MHz by the rate, or:

Divisor32 = 10MHz/TrigRate (in Hz)

where the "Divisor32" = (Timer #0 divisor 16) x (Timer #1 divisor 16)

Remember that this is the *trigger*, not the A/D sample rate. Therefore, trigger rates will be lower than the A/D rate. Do not use the trigger as an A/D clock.

2. Note that this must be an *integer* division. Therefore, you can only generate internal triggers which are a whole number divisor of 10MHz. If you must have an exact trigger rate which is not an integer divisor, use the External Trigger.

3. Assuming this divisor is acceptable, now convert it to upper and lower 16-bit words.

Although you can write any value between 2 and 65,535 to either counter, you will reach the limit (65,535) on one of the counters at very low trigger rates. To make things simple, set one of the 16-bit divisors to 2 and crank up the other divisor to what you need. If you reach the limit, start increasing the first divisor word. The net divisor value product is what counts.

The two chained 82C54 counters ***do not*** in effect form the upper and lower 16-bit words of a 32-bit divisor with the last counter (#1) acting as the most significant word and the first counter (#0) the least significant word. The reason is that these counters *are truncated by the count value*. They are only 16-bit counters when loaded with 65535. So the following analysis will ***not*** work when trying to figure the two 16-bit counter words:

Upper word (for Counter #1) = Divisor32/65536 /* don't use this */
Lower word (for Counter #0) = Divisor32 % 65536 /* don't use this */

4. The *minimum* divisor word for each counter is *two*. Therefore, the highest internal trigger frequency is 10MHz/(2 x 2) = 2.5MHz. The "^" up-arrow means exponentiation.

At high internal trigger rates, you will have less resolution. For example, the next higher trigger rate above 1MHz (10MHz/10) is 1.111MHz (10MHz/9). But you cannot generate a "9" divisor with a factor of 2 (*one* of the divisor words *must* be 2 or greater). So the next allowable trigger rate is really 1.25MHz (10MHz/(2x4)).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88[illegible] | www.artisantg.com

Single Internal Trigger:

Counters #0 and #1 specify the trigger rate. If only one trigger is required then Counter #1 must be configured for Mode 4.

Load the following hexadecimal numbers into the Control Word Register:

Load 34 hex in control register — Timer #0, LSB then MSB, Mode 2, Binary Count
Load xxxx at Counter Register #0 data

Load 78 hex in control register — Timer #1, LSB then MSB, Mode 4, Binary Count
Load 1 hex at Counter Register #1 data

Load 64 hex in control register — Timer #2, LSB then MSB, Mode 2, Binary Count
Load yyyy hex at Counter Register #2 data

xxxx = The number of conversions per Trigger - 1
yyyy = CLOCK1/(Required conversion rate in Hz)

Remember that the trigger rate = 10MHz/[timer 0 divisor x timer 1 divisor]

Continuous Internal Trigger:

In this mode, Counters #0 and #1 (the trigger rate specifier) are both set up for Mode 2. When the counter decrements from the initial count value to 1 the output goes low for one clock pulse, the Counter Element reloads the initial value and the sequence restarts. Hence, internal triggering continues until data acquisition is terminated by resetting bit 1 in the Command Register. The number of samples per trigger, i.e. scan count, is determined by Counter 0.

Load 34 hex in control register — Timer #0, R/L LSB then MSB, Mode 2, Binary Count
Load xxxx at Counter Register #0

Load 74 hex in control register — Timer #1, R/L LSB then MSB, Mode 2, Binary Count
Load yyyy at Counter Register #1

Load 64 hex in control register — Timer #2, R/L LSB then MSB, Mode 2, Binary Count
Load zzzz at Counter Register #2

xxxx = The number of conversions per Trigger 1
yyyy = CLOCK1/(Conversion rate in Hz)
zzzz = CLOCK2/(Trigger rate in Hz)

(See Example #1 in Section 5.4).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

7.2 82C54 Operation in PCI-416

Counter #2 (A/D Sample Rate):

This specifies the conversion start clock rate. As with Counter #0, the gate is controlled by the ACQuire flag and it is clocked by a 10MHz oscillator clock or by an external clock input.

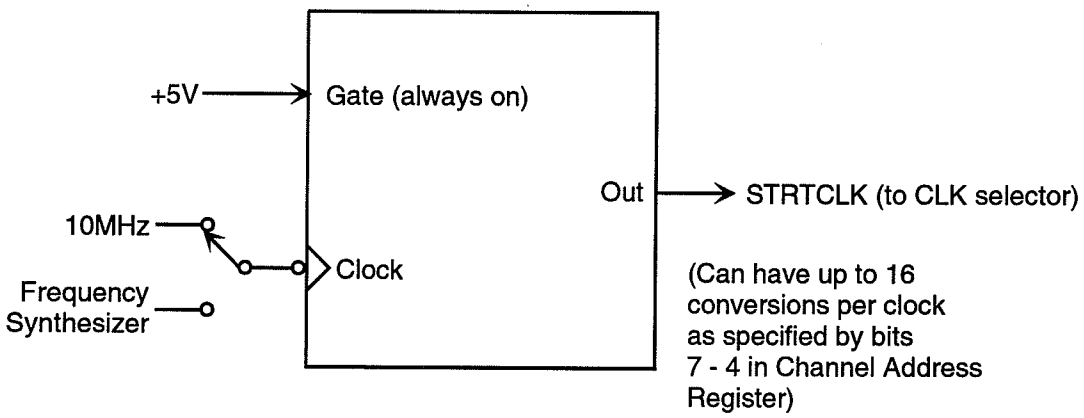


Figure 7.2b Counter #2

In SS&H mode (PCI-416A) Counter 2 specifies the per channel sample rate. Multiple A/D conversions (up to 4 selected by channel registers) occur on each sample/hold cycle.

Counter #0 and #1 in cascade (internal trigger rate):

This generates the internal trigger. It is clocked by 10MHz standard clock.

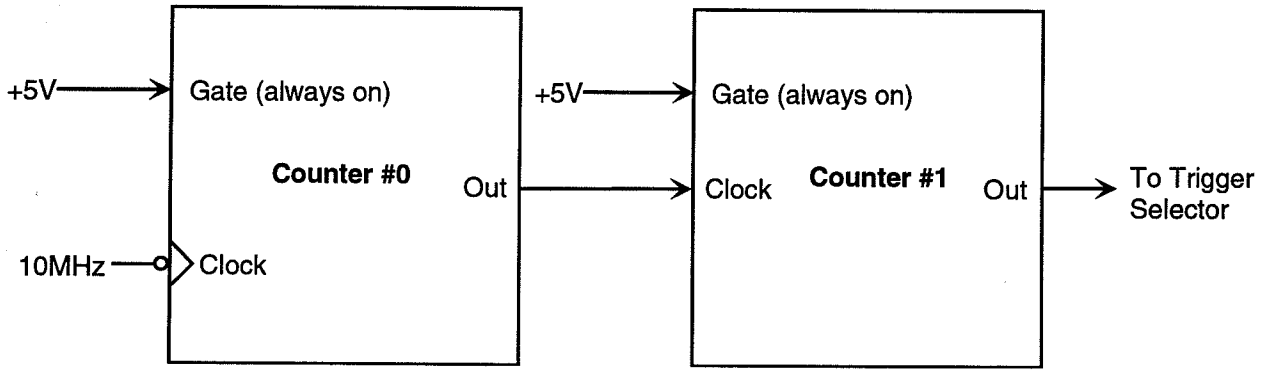


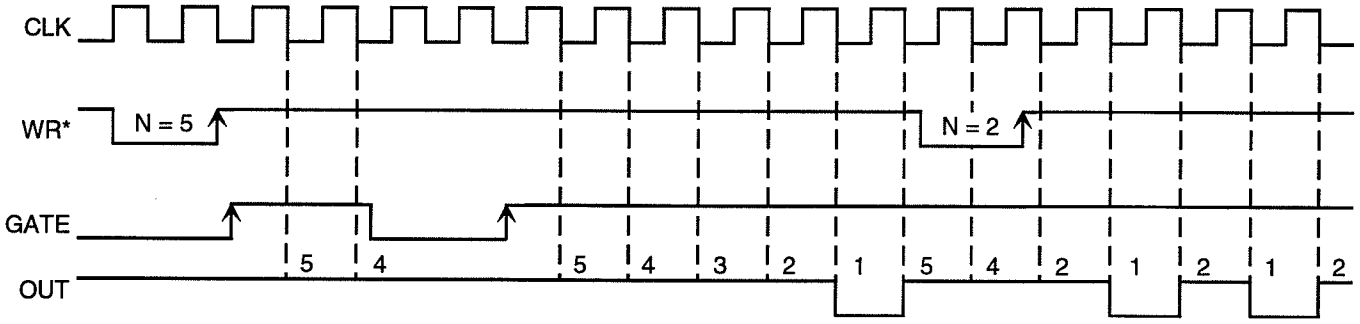
Figure 7.2c Counters #0 and #1

7.3 82C54 Programming Guide for the PCI-416

Note 1: The timer value registers are eight bits wide. The counter value must be divided into an eight bit LSB and an eight bit MSB if the initial value is wider than 8 bits i.e. greater than 255. The MSB may be determined by doing an integer division by 256. The LSB is obtained using the modulo operator i.e. LSB = count value modulo 256. The timer counter registers expect the LSB to be loaded first, followed by the MSB.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A high level on the gate validates counting while a low invalidates counting. On the falling edge of the gate signal the output goes high and counting terminates. The rising edge of the gate retriggers the counter i.e. the old initial value is reloaded, or if a new value has been written by the host, the new initial count value is loaded on the next falling clock edge. If a new value is written to the Counter Register during a count sequence it is loaded into the Counter Element either after gate retriggering or when the present count sequence terminates.



Note: In Mode 2, a count value of 1 is prohibited

Mode 2 Timing Diagram

Important: On the PCI-416, all 82C54 gate inputs are always enabled on. The 82C54 gates enable the outputs if the counters have reached zero.

Mode 4 - Software Triggered Strobe

Mode 4 creates a single delayed output pulse for one clock period and does not automatically repeat, unlike Mode 2. Another pulse is generated if the ocunter is reloaded by software. The PCI-416 uses Mode 4 in Counter #1 for a single internal trigger. The Counter #1 output is gated by bit 15 in the A/D Convert Enable Register. Since the countdown starts immediately after writing counter data, make sure the A/D is enabled first.

To select Mode 4 write the Control Word to select the Counter and configure the mode (e.g. for Counter 0 this is 38 hex).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

BCD
(Bit 0) 0 Straight binary count
1 BCD count

Counter Register (see Section 4 for addresses)

Counter 0 (internal trigger rate)
Counter 1 (internal trigger rate)
Counter 2 (A/D sample rate)

After writing the Control Word Register, load the Counter Register with the initial counter data. The Counter Register bit format is as follows:

31 - 8	7	6	5	4	3	2	1	0
Not used	C7	C6	C5	C4	C3	C2	C1	C0

Figure 7.2 Counter Register

(C7 . . . C0) Initial counter value data bits. Note that this is only an 8-bit register. 16-bit data is
(Bits 7 . . . 0) written as two bytes - controlled by bits 5 and 4 in the Control Word Register. Setting both of these to "1" and writing two bytes to the Counter Register will load the Register with the Least Significant Byte first, then the Most Significant Byte. The counter counts down from this initial value. On reaching 0 (Mode 5) or 1 (Mode 2) the output goes low for one clock period and returns high again.

Important: On high speed host computers, two contiguous accesses to the 82C54 registers should be separated by delays. The internal counter logic may be too slow (setup/hold time validation) for the higher speed computers.

7.1 82C54 Modes of Operation

Counter modes are specified by bits D3, D2, and D1 in the Control Word Register - as illustrated in Section 5.1. On the PCI-416, Mode 2 is used for continuous internal A/D triggering. Mode 5 generates a single internal trigger.

Mode 2 - Rate Generator

Control Word setting asserts the output high. When the count decrements to 1 the output goes low for the duration of one input clock period and then returns high again. The initial count value is then reloaded into the counter and the sequence restarts. If the initial count value is N, then the sequence repeats at intervals of N clock pulses.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88 SOURCE | www.artisantg.com

Functions are selected using 8-bit control words written to the Control Word Register. This register in turn configures the operation of all three counters. Each counter data register is then loaded with a 16-bit initial value by the host (two 8-bit writes). On the next input clock edge (falling) this initial data is loaded into the counters and counting commences. The output pulse duty cycle and frequency depends on a particular counter's mode of operation.

Important: Control words must always be written before counter data. Because of internal 82C54 register propagation times, two contiguous register accesses on the timer should be separated by a delay.

All PCI accesses must be 32-bit wide "DWORDs".

Control Word Register (I/O address BADR4, ARL = 12)

Each counter's operating mode is configured by control word programming. The format of the control word is shown in Figure 7.1.

31 - 8	7	6	5	4	3	2	1	0
X - X	SC1	SC0	RL1	RL0	M2	M1	M0	BCD

"X" indicates "don't care" bits

Figure 7.1 Control Word Register

Select Counter (Bits 7, 6)	SC1	SC0	
	0	0	Select counter #0
	0	1	Select counter #1
	1	0	Select counter #2
	1	1	Read Back command

The Read Back Command may be used to monitor the counter status at any time during operation - see later for details.

Read/Load (Bits 5, 4)	RL1	RL0	
	0	0	Counter latch operation - stores a given counter's output value for host reading. This value is held until read by the host.
	0	1	Read/Load a counter with its 8-bit LSB initial data value.
	1	0	Read/Load a counter with its 8-bit MSB initial data value.
	1	1	Read/Load a counter with its 8-bit LSB followed by its 8-bit MSB initial data value.

Note: A bit setting of "1 1" means loading the initial count value in two data bytes - one 8-bit write followed by another 8-bit write. It does NOT mean the host can write a 16-bit data word to the 82C54 - simply because the Counter Data Registers are physically only 8 bits wide.

Mode Select (Bits 3 . . . 1)	M2	M1	M0	
	X	1	0	Mode 2, rate generator
	1	0	0	Mode 4, software triggered strobe
	1	0	1	Mode 5, hardware triggered strobe

Note: "X" means don't care, i.e. can be either 0 or 1.

These bits select the mode of operation of each counter on the 82C54 timer. Each mode is dealt with in detail in Section 7.1.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 7

82C54 PROGRAMMABLE INTERVAL TIMER

7.0 General Description

This section describes the 82C54 programmable timer/counter on the PCI-416. It is used for two functions:

1. Internal Trigger Rate Clock *and*
2. Internal A/D Sample Rate Clock

The PCI-416 also accepts *external* trigger and A/D clocks - see the other sections of this manual.

The 82C54 can be programmed many ways. However, only a few modes are useful for the PCI-416 (see Section 5, "Using the PCI-416"). The 82C54 is made by numerous suppliers (Intel, NEC, Oki, etc.). Additional information on the 82C54 is available by consulting the data books of these manufacturers but it will probably not be necessary for programming the PCI-416.

The 82C54 is a CMOS programmable interval timer containing three individual 16 bit down counters that can count at speeds up to 10MHz. The timer features six different counter modes. All timer functions are fully programmable. 82C54 functions are pre-selected by a Control Word register and counter data is written to the Counter Registers - one for each counter.

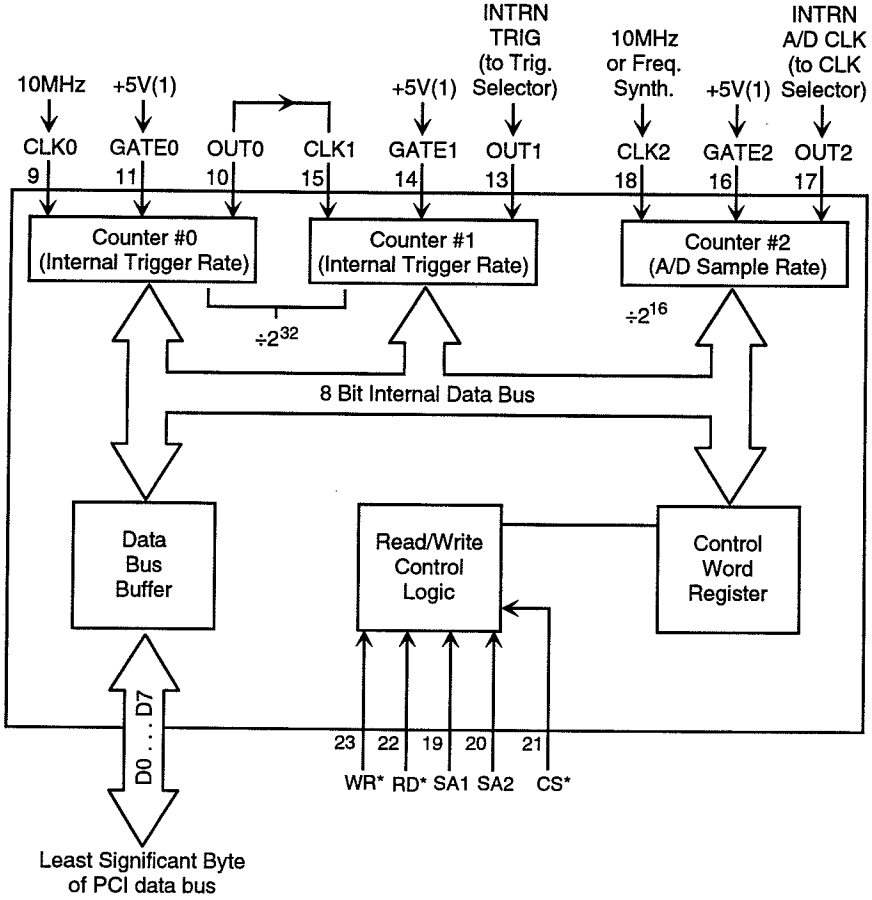


Figure 7.0 82C54 Programmable Timer

Note (1): +5V continuously enables all 3 gates "on".

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

However, if you enter "ver" (for version), the system responds with a message something like, "Windows 95 [version 4.00.950]". On further investigation, if you press "Ctrl-Alt-Del" for warm boot, Windows 95 optionally returns!! Therefore it never really left. A small Windows 95 control stub remains in memory at all times intercepting memory, disk and I/O accesses on a priority basis. The CPU must switch back and forth in protected mode to service these tasks. The extraordinarily high speed CPU is being hobbled by operating system priorities and protection. And the hundreds of overhead operations being performed in the background somewhat increase the risk of data errors. These effects are the heart of why Windows 95 slows down the PCI-416. This mode too cannot use PCI-416 bus mastering.

Windows 95 also gives you a third option at power up of installing *only* DOS if you press the F4 key immediately after power turn on. This mode *will* accept PCI-416SET bus mastering and does not leave any part of Windows 95 resident in memory. Read the Windows 95 installation instructions carefully to make sure you have this mode available.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

7. In bus master mode and very high A/D sample rates combined with other high duty cycle PCI peripherals, internal A/D FIFO overflow (and loss of data) can only be tested in the A/D status register after the block transfer completes. This and other speed effects may be probabilistic. That is, since several complex subsystems (disk and video controllers and the PCI-416) may all be contending for the bus simultaneously, overrun errors do not happen at exactly predictable times. Instead, you must make numerous measurements of long periods and look for how frequently lost data occurs.

This is a statistical situation and will need multiple repeated tests. Do not test with very short data sequences since you may never get an error. Use at least several megabytes of streaming data. Also, since the FIFO overflow flag does not tell you exactly which sample caused the overflow, consider testing with ever-increasing block sizes until the errors are more numerous.

6.12 Analog Signal Recommendations

1. Do not let the analog signal exceed the input voltage range. You may experience settling time delays or possible temporary crosstalk to other channels. Higher input voltages exceeding the maximum input range (typically ±15 Volts), may damage the board, even if they are very short duration.

2. Terminate unused input channels to ground on single-A/D systems which use an analog input multiplexer. Or use an input load resistor to keep inputs within the signal range. If the channels are unconnected, tiny input bias currents will cause the inputs to saturate, causing settling delays (but no damage) when an active channel is connected. If such termination is inconvenient, rearrange the channel sequence so that unused channels are higher in the channel order.

3. Many high speed A/D options have low input impedance. Make sure you understand concepts of source resistance if your input source does not have very low driving impedance. If necessary, make level measurements with the A/D connected to determine the actual level being presented.

4. For differential inputs, make sure *both* inputs remain within the Common Mode Voltage Range at all times during signal measurement. This rule is normally accomplished with a typical bridge type input. If you use an external shield system on the balanced inputs, make sure *no current* flows in the shield. If these signal concepts are not clear to you, get expert assistance.

6.13 Windows 95® Compatibility, March 1996

The following discussion does *not* apply to Windows 3.1. Many new computers are now supplied with Windows 95. DATEL is developing optimized 32-bit code which will get the best possible performance from Windows (but MS-DOS is still faster). Contact DATEL for availability on this system. In the meanwhile, PCI-416SET can be used in DOS mode with Windows 95 under certain conditions.

Windows 95 has several ways to access DOS. The first is the "DOS compatibility box" using the MS-DOS icon. This mode runs PCI-416SET successfully as long as the PCI-416 *does not use bus mastering.* Ironically, bus mastering is *the* feature you would want to reduce Windows' speed penalties. You can also minimize this box and run PCI-416SET in the background. The PCI-416 will interleave with other tasks. Close unnecessary tasks to minimize how much the speed degrades.

The second method is somewhat deceptive. Windows 95 can accept a "shut down to DOS" mode where Windows 95 disappears and only the DOS "C>" prompt remains. This concept is similar to exiting out to DOS from Windows 3.1 from the old Program Manager. From the DOS prompt, if you enter "exit" as if you were trying to return to Windows 95, the DOS prompt remains, leading you to believe that DOS truly is in control and Windows 95 is not in the system.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3. Note that the A/D sample count register and the 5933 block transfer count (MWTC) are separate, independent registers. Do not confuse the MWTC and the A/D sample counter. They interact as follows:

If the A/D sample count reaches zero (implying the end of a "frame") and the A/D FIFO reaches empty but the 5933 has not completed its block transfer count, data movement will restart with the next A/D trigger loading the FIFO. If there is a bus grant timeout, the PCI-416 will reacquire bus mastership shortly after new A/D samples enter the FIFO. Typically, losing and reacquiring the PCI bus is a very short period (assuming no other high duty cycle bus contention) so it will have negligible effect on total throughput speed. Users should carefully test everything with high bandwidth applications such as graphics.

If the A/D clock stops before it is expected (such as an external A/D clock which halts too soon), the FIFO will empty but the block transfer will not finish. The PCI-416 will then lose its mastership because it failed to make a transfer in the prescribed time (see the 5933 manual). An external supervisory program can determine that the block count is not done. Control may be reaquired by terminating the block transfer and reprogramming the 5933 and A/D registers. Bit 24 (the "add-on" bus reset) in the MCSR will reset all local PCI-416 A/D logic and restore all local register bits to ZERO.

If the A/D sample count is greater than the MWTC block transfer count, A/D samples will not be lost as long as the 5933 FIFO and the A/D FIFO never overflow and the system stays in bus master mode. As a practical matter, the 8-longword 5933 FIFO reaches capacity very quickly at high speeds but will not overflow because the A/D FIFO is inhibited from writing to it. Typically, this means that another block transfer must be set up quickly (i.e., before the FIFO overflows) if more data is being collected. Since the PCI bus is usually faster than any A/D rate, the system will "catch up" when a new bus master cycle starts. Since it is not possible to tell how much data is in the A/D FIFO, do not try to transfer more than half the FIFO when the half full flag appears.

4. A/D Command Bit 16 terminates writes from the A/D FIFO to the 5933 FIFO. And the same internal 5933 bus is used both for the A/D registers and A/D data flowing into the 5933 FIFO. Therefore access to the A/D registers after a bus master block terminates may lose any A/D samples which were not transferred from the 5933 FIFO after the bus master transfer terminates.

All of the above characteristics lead to a simple rule: Do not reset the Bus Master bit 16 to ZERO while the A/D is running. You risk losing data. If the Bus Master block has finished, set up another block and start it.

5. Higher priority devices may suspend PCI-416 bus mastering or produce low duty cycle, causing possible loss of A/D data if the A/D FIFO overflows. Refer to the PCI-416 references for details. This is rarely a problem because of the fast response of the PCI bus and the arbitration mechanisms.

6. If the A/D trigger is delayed and the A/D FIFO does not start filling immediately, the block transfer will delay starting and the 5933 will go through a short bus request/grant cycle before transferring data. Once the block transfer starts, if the trigger is periodic with counted blocks loading the A/D FIFO, the PCI-416 will toggle back and forth, taking the bus whenever it needs it. This operation is transparent to the user until the block finishes.

One other method for delayed triggers is to start the board in program mode (i.e., not bus mastering yet), polling for the ACQuire bit to detect when sampling has started. Confirm this with the A/D FIFO empty bit. After the FIFO starts filling, transition the system to bus master mode while the A/D continues running. A possible reason to use this method is in case the board does not start within the expected time.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Each transfer continually tests these two conditions and if either condition blocks transfer, the system will continuously loop until the conditions are satisfied again but will not exit the block transfer cycle if the block count is not complete. If there is momentarily no A/D data, the PCI-416 will relinquish the bus temporarily but returns immediately when more A/D samples appear. This operation continues until the whole block is done. If the host detects a locked up transfer, it can reset the 5933 and PCI-416 to reacquire control.

This operation proceeds transparently until the 5933 transfer byte count is complete. All normal PCI bus operation applies here such as a receiver which cannot accept the transfers, higher priority devices requesting the PCI bus, etc. Remember that only one PCI initiator can have bus mastership at any one time. However, review the PCI priority and "fairness" rules. Also study the effects of the Latency Timer. And be aware that the PCI priority strategy (round robin rotated, fixed priority, custom, etc.) is unique to your host PC and is explicitly not defined by the PCI standard. You must determine this priority scheme for your own PC (or replace it).

A very important point is that the A/D registers can no longer be accessed from the PCI bus during bus master operations. Therefore all internal A/D registers must be programmed before starting bus master mode. This includes the A/D Command Register bus master mode bit 16. If the host needs to reacquire control of the A/D registers after starting the block, terminate the bus mastering by command to the 5933. Then reset the A/D logic using bit 24 in the MCSR.

Ring Buffer Bus Mastership

An anomaly is seen when using this system for ring buffering. First, the start address and length of the host ring buffer is set to the MWAR and MWTC. As A/D data streams to memory, the end of the ring buffer is reached, the block transfer is done then the 5933 interrupts the host. This interrupt causes the host side software to reload the original starting memory address and the data now overwrites old samples in circular fashion.

Recall that the A/D sample counter may not be an exact integer multiple of this block transfer size. Since A/D sampling terminates with this sample count, the PCI block transfer may not complete because the A/D FIFO is now starved for data. And since the host system can no longer access the A/D status register while the 5933 is in bus master mode, the host cannot directly find out when the final A/D sample countdown is complete. This is handled with a timer in the host which determines that the 5933 block transfer interrupt did not occur when expected, therefore the final A/D sample countdown (but not the bus master block transfer) is now complete.

Use the address register in the 5933 to determine where the last sample was written in memory. Then count backwards circularly using the original A/D sample count to determine where the "trigger" sample occurred which started the final countdown.

Note that you cannot simply set the MWTC to some block count larger than the ring buffer because there is no automatic mechanism in the host to force the circular addressing.

Please observe the following conditions when using PCI-416 Bus Mastering:

1. There is no mechanism to prevent overwriting valid memory by the 5933 controller. The user must be careful to preallocate blocks of sufficient size. Some memory systems may trap an illegal (i.e., non-existent) memory address and terminate the block transfer.

2. The single interrupt from the 5933 PCI controller is reassigned during bus master operation. It is no longer available from the A/D FIFO half full or ACQuire signals but instead may optionally occur when the 5933 block transfer count is done.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To set up bus mastering, first do all normal PCI-416 initialization necessary to control the board in status mode. This includes testing for the presence of the PCI BIOS, determining the base addresses, slot number, vendor and device ID's, I/O or memory, space allocation, etc. There are numerous registers here so follow DATEL's source code. We recommend that you run the board first in status mode (i.e., non bus master mode) to verify proper operation. Then,

1. Since the A/D registers will not be visible during bus mastering, initialize all internal A/D registers except the A/D Command Register.

2. Load the AMCC 5933 PCI controller with the count and 32-bit physical address of the start of previously allocated destination memory which will accept A/D data. This count is the number of *bytes* (not longwords!) transferred during the bus master operation and can be a large number up to 64 million (2^26) bytes. Since the PCI-416 transfers are always longwords, this is 16 million longwords (2^24) or 32 million A/D samples but use the byte count. Load the count into the Transfer Count Register (MWTC) and the starting destination address into the Write Address Register (MWAR). Both registers are part of Base Address Register 0 (BADR0).

3. Set up any other required 5933 registers not previously written to (interrupts, latency timer, etc.) except the PCI Command Register (PCICMD). Initialize the Control/Status Register (MCSR) in BADR0. Do not confuse the MCSR with the A/D Command and Status Registers. See the 5933 and PCI-416 Manuals for details.

4. Now write to the A/D Command Register, setting bit 16 to ONE. After writing to this register, you can no longer access these registers until bus mastering is terminated.

5. Finally, enable bus mastering by writing to the PCICMD at Configuration Space offset 04h. Bit 2 controls bus mastering.

Destination Memory Allocation

Note that the PCI destination memory in MWAR must be previously allocated from the Operating System. Then its logical starting address may need to be locked then converted to an absolute physical address. And some systems may not give a contiguous block that equals the desired count size in the 5933. Therefore a long block may need to be broken up into several smaller blocks, all of which must be preallocated before the PCI-416 bus master operation and all needing their own separate block transfer phases. Do this with a table of allocation handles and block sizes. To achieve longer contiguous blocks, deallocate (free) all unneeded memory before allocating memory for the PCI-416. There is no inherent hardware in the 5933 to warn of writing to unallocated memory - so be careful!

The interrupt request from the AMCC 5933 can be optionally set up to indicate that this longword count is complete although this can also be determined by polling the 5933.

Operation

When A/D Command bit 16 is set to ONE, a control PAL (Programmable Array Logic) on the PCI-416 transfers data from the A/D FIFO memory to a small 8-longword FIFO inside the 5933. (Do not confuse these two FIFO's). The control PAL allows this data flow as long as:

1. The A/D FIFO is not empty and,
2. The internal 5933 FIFO is not full.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The following is a list of commonly used IRQ's for PC's:

Interrupt Line	Interrupt No.	Common Use	Interrupt Vector
IRQ0	Int 08h	Time of day Interrupt	0000:0020
IRQ1	Int 09h	Keyboard Interrupt	0000:0024
IRQ2	Int 0Ah	Cascade for IRQ 8 - IRQ 15	0000:0028
IRQ3	Int 0Bh	COM2 Interrupt	0000:002C
IRQ4	Int 0Ch	COM1 Interrupt	0000:0030
IRQ5	Int 0Dh	LPT2 Printer Port Interrupt	0000:0034
IRQ6	Int 0Eh	Diskette drive Interrupt	0000:0038
IRQ7	Int 0Fh	LPT1 Printer Port Interrupt	0000:003C
IRQ8	Int 70h	Real Time Clock	0000:01C0
IRQ9	Int 71h	Reserved. Redirected IRQ2	0000:01C4
IRQ10	Int 72h	Reserved	0000:01C8
IRQ11	Int 73h	Reserved	0000:01CC
IRQ12	Int 74h	Mouse (PS2)	0000:01D0
IRQ13	Int 75h	Coprocessor Error	0000:01D4
IRQ14	Int 76h	Fixed Drive Interrupt	0000:01D8
IRQ15	Int 77h	Reserved	0000:01DC

Note that if a network adapter is installed, it will use one of the reserved interrupt lines.

6.10 Policy on Design Information and Modifications

The design of the PCI-416 is proprietary and DATEL does not distribute schematic diagrams, internal timing charts and logic programming information. Specifically, the product is not intended or supported for internal modification or enhancement by the user *except* as covered in this User Manual. And customers should not attempt repair of the board except as described here. The purpose of this product manual is to give the experienced programmer and user complete application information from an external point of view, not a discussion of the internal design.

To help users understand these policies, the product is complex and any internal changes require thorough knowledge of high performance data conversion circuits and large, fast programmable devices. Also, a thorough understanding of the internal design requires a detailed review of internal engineering changes and documentation, not just schematics.

In a similar way, although you own your automobile, without being an expert, the detailed design information on its engine and systems is not distributed and probably is not very useful to you. Even if it were, the car manufacturer is not organized to handle requests for such information. For larger special orders with scheduled deliveries, DATEL will review customer requests to modify the product. This is always done from a set of clear, written specifications supplied by the customer. Thank you for your understanding.

6.11 PCI-416 Bus Master Operation

PCI bus mastering offers the highest possible speed available on the PCI-416. The PCI-416 A/D Command Register bit 16 enables PCI bus master operation. This is conceptually similar to DMA (Direct Memory Access) transfers in a PC but is really PCI bus mastering. It does not use an 8237-style DMA controller in the host computer and therefore isn't blocked in 64K max. groups. PCI-416 bus mastering works as follows:

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2. DATEL offers a free PCI-416 test program for additional diagnosis. If this Test program shows Interrupt Line = 1 (or any other incorrect setting), then it is possible that there is a conflict between the PCI-416 and the IDE PCI controller. Most PCI IDE Controllers use edge triggered interrupts. The PCI-416 (and every other expansion PCI board) uses level triggered interrupts.

3. ***Warning:*** *The following procedures use the BIOS Setup screen. Incorrect setup may temporarily disable your disk system. Your computer hardware manual usually lists a way to reset the system if you inadvertently write incorrect settings to the CMOS memory. If in doubt,* ***get expert assistance.*** *Your CMOS memory can also be erased by a bad CMOS battery. It's an excellent idea to go through the entire CMOS Setup system and write down all settings. Then keep this written record in a safe place in case you have to restore the CMOS memory.*

PCI devices all map their "INT" interrupt bus lines to a corresponding IRQ line to the system interrupt controller. Normally, each PCI slot has a unique INT line, starting at INT A up to INT D. The INT setting of any board *must* match the INT line specified for that slot. And normally, only one board can use a given INT. Each slot can often be reassigned to other INT's by mother board jumpering. And some boards can reassign their INT's. The mapping from INT's to IRQ's normally also uses jumpers or DIP switches on the motherboard. The PCI-416 card is set permanently for INTA (as recommended by the PCI spec for single-function boards).

You should also check your system manual for edge versus level triggering settings for each slot. Most PCI expansion boards (including the PCI-416) use level-triggered interrupts. All current PCI IDE disk controller boards use edge triggering. Note that if your system has *both* EISA and PCI buses, individual IRQ's must be assigned to either EISA or PCI but not *both.*

Try changing the jumper on the PCI IDE Hard Drive Controller to use INTB rather than INTA. In the BIOS Setup screen (available during bootup), set the INT Line to INTB for whatever slot the IDE controller is installed in. On some motherboards you must set a jumper on the motherboard to select edge triggered interrupts. Install the jumper for INTB IRQ14 (typical) and in the BIOS setup, select the interrupt source from the jumpers (not Auto Routed). Make sure that there is no conflict in the IRQ line for the PCI-416 and IDE card.

For example, if the IDE card is in PCI slot 2 and the PCI-416 is in PCI slot 3, in the BIOS setup, set Slot 2 to use INTB and IRQ14 and Slot 3 to use INTA and IRQ15 or other unused IRQ line. Likely candidates for the PCI-416 IRQ line are 5,7,9,10,11,12, and 15. Consult the motherboard manual.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3. Determine the output voltage range.

4. Using PCI-416SET, write the hexadecimal values, from Table 6.2 to the D/A Register. Start with the near MINUS Full Scale initial setting for trimming the Offset, (line marked OFFSET/ZERO Adj.), and set to NEAR POSITIVE Full Scale, initially, for adjusting the Gain (line marked GAIN Adj.). The other DAC Register data setting in the table may be used to verify the output accuracy.

DAC Data (Hex)	Analog Output Voltage for each D/A Full Scale Input Range		
	±10V	±5V	
0000	-10.000V	-5.0000V	
0001	-9.9951V	-4.9975V	OFFSET/ZERO Adj.
0002	-9.9902V	-4.9951V	
0004	-9.9805V	-4.9902V	
0008	-9.9609V	-4.9805V	
0010	-9.9219V	-4.9609V	
0020	-9.8438V	-4.9219V	
0040	-9.6875V	-4.8438V	
0080	-9.3750V	-4.6875V	
0100	-8.7500V	-4.3750V	
0200	-7.5000V	-3.7500V	
0400	-5.0000V	-2.5000V	
0800	0.0000V	0.0000V	
0C00	5.0000V	2.5000V	
0E00	7.5000V	3.7500V	
0F00	8.7500V	4.3750V	
0F80	9.3750V	4.6875V	
0FC0	9.6875V	4.8438V	
0FE0	9.8438V	4.9219V	
0FF0	9.9219V	4.9609V	
0FF8	9.9609V	4.9805V	
0FFC	9.9805V	4.9902V	
0FFE	9.9902V	4.9951V	GAIN Adj.
0FFF	9.9951V	4.9976V	

Table 6.2 D/A Calibration Table

6.9 Hard Drive Interrupt Conflict Solutions for the PCI-416

The PCI-416 can operate with both IDE and SCSI hard disk controllers which connect to PCI bus. The PCI-416 can also work with older, slower ISA bus disk controllers. If you encounter any problems using interrupts and/or Bus Mastering in PCI bus computers with IDE Hard Drive controllers connecting to PCI bus (such as Aptronix), try the following:

1. *Before* installing the PCI-416, make note of which PCI slot it will use. The PCI-416 *must* be a **bus master** to enable high speed block transfer. Consult your computer hardware manual to find a bus master PCI slot. On some older PCI systems, it is possible that not all PCI slots allow bus mastering. Most newer PCI systems accept bus mastering on all slots. If the PCI-416 runs correctly in Status or Interrupt transfer mode, but fails bus mastering, this could be because of the slot not accepting bus mastering. Bus master failure could also occur because the higher speed is too fast for the installed memory.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6.7 Universal Coding Chart

This chart shows the hexadecimal output codes to be expected for the analog input shown. "FS" means full scale. All hexadecimal data has the A/D Most Significant Bit sign-extended up to bit 15, according to the resolution of the A/D converter.

The chart uses the convention that the MSB major carry occurs at the half scale transition voltage. The MSB bit is active (either a 0 or 1 depending on the coding) and all other bits are inactive. At minus full scale all bits are off.

The "zero" or half scale analog input value can be set as a transition code of 2047 to 2048 for a 12-bit converter or it can be set at the midpoint of the 2048 code. Using the transition mode gives an equal number of codes spanning the midscale voltage. It also means that there is no code that designates the exact center. The midpoint of the "0000" two's complement output code is actually +1/2 LSB input voltage and not zero volts for a bipolar converter. The adjustment potentiometers allow a small correction range if other values are desired.

If the midpoint of the half scale code (2048) is set to zero volts or half the input voltage scale, then full scale will have one less code. This gives a code which designates the exact center. The calibration chart shows this unbalance.

Input Voltage Bipolar (Unipolar)	12-bit Offset Binary	12-bit 2's Compl.	14-bit Offset Binary	14-bit 2's Compl.	16-bit Offset Binary	16-bit 2's Comp.
+FS	FFFFh	07FFh	FFFFh	1FFFh	FFFFh	7FFFh
+FS -1 LSB	FFFEh	07FEh	FFFEh	1FFEh	FFFEh	7FFEh
Zero +1 LSB (1/2 FS +1 LSB)	F801h	0001h	E001h	0001h	8001h	0001h
Zero (1/2 FS)	F800h	0000h	E000h	0000h	8000h	0000h
Zero -1 LSB (1/2 FS -1 LSB)	07FFh	FFFFh	1FFFh	FFFFh	7FFFh	FFFFh
-FS +1 LSB (Zero +1 LSB)	0001h	F801h	0001h	E001h	0001h	8001h
-FS (Zero)	0000h	F800h	0000h	E000h	0000h	8000h

Ouput codes are in hexadecimal.

Table 6.1 Universal Coding Chart

6.8 D/A Calibration

1. Locate the offset and gain adjustment potentiometers on the PCI-416 board - these are on the motherboard.

2. Connect a precision Digital Voltage Meter (DVM) to the analog output connector. Connect the negative probe of the DVM to an analog ground point, e.g. connector case.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6.5 Zero/Offset Adjust

With the A/D converting on Channel 0, apply minus Full Scale + 1/2 LSB to the Channel 0 input. The inputs to be used are shown in Table 6.0. For 12-bit converters, adjust the Zero/Offset pot in the A/D module so that the output data is fluctuating between 0000 and 0001 hex for unipolar coding or F800 and F801 hex for 2's complement. For the 14-bit converter, the codes are 0000/0001 and E000/E001. Note that "binary" may mean straight or offset binary depending on the A/D converter input range. See Table 6.0.

	12-bit A/D data (hex)		Calibrator Input Voltages for A/D Full Scale Ranges				14-bit A/D data (hex)		
	2's Comp.	Offset Bin.	±10V	±5V	0 - 10V	±1.25V	2's Comp.	Offset Bin.	
	F800	0000	-10.000V	-5.0000V	0.0000V	-1.2500V	E000	0000	
OFFSET Adj.	F801	0001	-9.9951V	-4.9975V	0.0024V	-1.2494V	E004	0004	
	F802	0002	-9.9902V	-4.9951V	0.0048V	-1.2488V	E008	0008	
	F804	0004	-9.9805V	-4.9902V	0.0098V	-1.2476V	E010	0010	
	F808	0008	-9.9609V	-4.9805V	0.0195V	-1.2451V	E020	0020	
	F810	0010	-9.9219V	-4.9609V	0.0391V	-1.2423V	E040	0040	
	F820	0020	-9.8438V	-4.9219V	0.0781V	-1.2305V	E080	0080	
	F840	0040	-9.6875V	-4.8438V	0.1563V	-1.2107V	E100	0100	
	F880	0080	-9.3750V	-4.6875V	0.3125V	1.1719V	E200	0200	
	F900	0100	-8.7500V	-4.3750V	0.6250V	-1.0938V	E400	0400	
	FA00	0200	-7.5000V	-3.7500V	1.2500V	-0.9375V	E800	0800	
	FC00	0400	-5.0000V	-2.5000V	2.5000V	-0.6250V	F000	1000	
	0000	0800	0.0000V	0.0000V	5.0000V	0.0000V	0000	2000	
	0400	0C00	5.0000V	2.5000V	7.5000V	0.6250V	1000	3000	
	0600	0E00	7.5000V	3.7500V	8.7500V	0.9375V	1800	3800	
	0700	0F00	8.7500V	4.3750V	9.3750V	1.0938V	1C00	3C00	
	0780	0F80	9.3750V	4.6875V	9.6875V	1.1719V	1E00	3E00	
	07C0	0FC0	9.6875V	4.8438V	9.8438V	1.2109V	1F00	3F00	
	07E0	0FE0	9.8438V	4.9219V	9.9219V	1.2305V	1F80	3F80	
	07F0	0FF0	9.9219V	4.9609V	9.9609V	1.2423V	1FC0	3FC0	
	07F8	0FF8	9.9609V	4.9805V	9.9805V	1.2451V	1FE0	3FE0	
	07FC	0FFC	9.9805V	4.9902V	9.9902V	1.2476V	1FF0	3FF0	
GAIN Adj.	07FE	0FFE	9.9902V	4.9951V	9.9951V	1.2488V	1FF8	3FF8	
	07FF	0FFF	9.9951V	4.9976V	9.9976V	1.2494V	1FFC	3FFC	
	Not Applicable		9.9976V	4.9988V	9.9997V	1.2500V	1FFE	3FFE	GAIN Adj. (416E

Table 6.0 A/D Calibration Table

6.6 A/D Gain Adjustment

With the A/D converting on Channel 0, apply +Full Scale -1.5 LSB to the Channel 0 input as shown in Table 6.0. For the 12-bit A/D converter, adjust the full scale Gain pot in the A/D module to observe A/D readings fluctuating between FFFE hex and FFFF for unipolar coding or 07FE and 07FF hex for 2's complement. For the 14-bit converter, the codes are FFFE/FFFF or 1FFE/1FFF.

Important: Once the gain adjustment is done, repeat the zero/offset test again. Since these two adjustments interact, repeat them until no further improvement can be obtained.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Equipment Required

- Precision low-noise DC voltage calibrator such as DATEL's DVC-8500
- Miniature jeweler's screwdriver set and clipheads
- DATEL PC-490B screw terminal adapter or other means to connect generator signals to the board

6.4 Procedure

Consult the drawings in this manual to locate the gain and offset potentiometers. All the modules are designed so that these pots face upwards with the PCI-416 installed in the computer. The calibration environment should be electrically quiet. Avoid large industrial switching noise or excessive RF fields. Use short shielded leads to the signal input connections and do not route them near noisy sources. Avoid placing the PCI-416 near other digital boards and the switching power supply.

Allow several minutes warm-up before starting adjustments. The PCI-416SET includes a calibration mode which continually displays successive A/D values.

It is only necessary to calibrate channel zero for models PCI-416B through E and H since they all share a single A/D converter. The PCI-416A requires individual offset and gain calibration on each SSH channel. PCI-416J and PCI-416L are pre-calibrated.

The calibration procedure for all converters is essentially the same. Since the A/D converter is very linear, it only needs to be calibrated near the endpoints of its DC transfer function. Do not use the actual endpoints since we wish to observe some adjustment range.

An anomaly of a perfect A/D converter is that the midpoints of its code width cannot be directly observed. Only the transition voltage between codes can be detected. The calibration voltages are chosen to be at the decision points midway between code states. It is normal for the hexadecimal display to flicker slightly with each new conversion, especially for fast, high resolution converters. Adjust the pots so that the flicker spans the code states equally.

The zero/offset pot adjusts either the zero for unipolar converters or the minus full scale (offset) for bipolar converters. The full scale pot is always for positive full scale.

If you wish to select a fixed higher gain, recalibration is always required after adding the gain resistors. The procedures are the same but you will have to supply smaller input voltages. Data output flicker may increase with higher gain.

The bit weight ("LSB") is referenced in the discussions following. Bit weight is the amount of voltage difference necessary to cause a transition between code states in an ideal A/D converter. It is the total input voltage span (both positive and negative portions) divided by the number of code states or:

Bit weight = Input span (in Volts)/((2^(no. of resolution bits)) -1)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

In **Mode 0**, the 24 I/O lines are configured as three bytes assigned as either inputs or outputs. Any combination may be programmed, including all 24 out, all 24 in, 16out/8 in, etc. The third port C can be split as two 4-bit nybbles slaved to respective ports A and B, each separately controlled as In or Out. In this "basic I/O mode", there are no handshake or strobe lines to indicate when an external device has read a port or has new data ready. In addition, there are no latches to store data until the CPU or external device uses it. Mode 0 should be used for low speed bit control and status of external devices.

Mode 1 allows fully strobed and latched operation on two ports A and B. Port C is now dedicated as a strobe/handshake port and only 16 bits are available for data. These however may be 16 out, 16 in or 8 out/ 8 in. And the direction can be programmed In or Out as needed for either port A or B.

Mode 2 uses only port A and is intended as a single 8-bit fully bidirectional latched bus shared with other external byte-wide devices. This mode 3-states the bus to a high impedance value when no transfers are in progress so that external devices can use the bus exclusive of the 82C55. In or Out direction is set by the control register but handshake lines from the external device indicate when data is ready and when to change direction. Independent Input/Output 1-byte buffers hold data until it is transferred.

Mode 2 can also be combined with either mode 0 or 1 for port B. When this is done, port B can be selected as Input or Output. All four combinations for port B (Mode 0/1, In/Out) use some port C pins as handshakes.

For both Modes 1 and 2, one-byte data buffers prevent loss of data samples.

Programming details for the 82C55 are included in the manufacturer's data sheets, see Appendix A.

6.2 Programming the 82C55

Full programming details are listed in the attached data sheets. But first, design a strategy for controlling the parallel port. This amounts to an overall timing sequence for the entire application. If you wish to interleave high speed A/D transfers with 82C55 activity, the best time to update the parallel port is immediately *after* an A/D FIFO block has been burst to host memory. At that instant, the FIFO is mostly empty and there is a little time available for 82C55 processing. Since the host PC is *not* available during an A/D FIFO block transfer, external I/O devices accessing the 82C55 must be either buffered (with something like a FIFO) or they must be low speed I/O which is not affected by FIFO block transfer delays.

6.3 A/D Calibration

CAUTION: Calibration must be done with the computer operating while its top cover is removed. Most voltages inside the PC are not dangerous. However, you should avoid the power supply and AC line cord area during calibration. Be particularly careful not to drop any tools into the computer while it is operating. Do not move the computer or touch the hard disk while it is running. Do not accidentally touch any components except the trim pots.

The PCI-416 is supplied tested and calibrated by DATEL. The board should be recalibrated at 90 day intervals depending on conditions. To avoid damage, users performing calibration should be thoroughly familiar with miniature electronic assemblies. Handle the board with anti-static protection.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 6

OPERATION NOTES

6.0 82C55 Parallel Port

The PCI-416 includes a general purpose bidirectional 24-bit parallel port. It is mapped into the PCI controller configuration space and is capable of transfers to about 2.5 megabytes per second. The port is *not* directly linked to the A/D FIFO and cannot be used as a stand-alone output port for A/D samples. Refer to DATEL's PC-415 and PC-414 boards for non-bus high speed parallel ports dedicated exclusively to A/D FIFO data.

The 82C55 is a byte-wide device and will only accept separate 8-bit data even though it must be accessed from PCI bus with 32-bit DWORD's. This last point is somewhat confusing but here is the explanation. The 82C55 can accept 8, 16 or 24-bit input/output modes but these various-size "data objects" must be built with one or more 8-bit byte transfers. To develop a 24-bit output, you must write three 32-bit wide DWORD's with each DWORD containing an 8-bit byte in its least significant bits. All upper bits (31-8) are not used but a full 32-bit DWORD must be sent to satisfy the PCI S5933 controller.

The 82C55 is intended for medium speed status and control or data transfer to external devices using a dedicated program. The I/O bits appear on an internal 50-pin flat cable header connector which is pin compatible with several other popular 82C55 parallel port boards. The outputs will generally drive at least one TTL load. However each port has different pullup/pulldown characteristics to match external busses.

Interrupts are not implemented on the PCI-416's 82C55 for several reasons. The S5933 PCI controller has only a single interrupt request and this is switched between the A/D-FIFO half full signal or the sample counter ACQuire signal. 82C55 interrupts would normally be generated with each byte transferred since the 82C55 only has a single byte buffer for ports A and B. Therefore, sending an interrupt with each byte would significantly slow down *both* 82C55 transfers, the A/D FIFO and the host computer. For this reason, the operation strategy should be to collect a block of data from the 82C55 while the A/D is filling its FIFO then wait while the A/D FIFO is block transferred. Or use the 82C55 as a lower speed I/O controller. The 82C55 can connect to faster external devices which have a FIFO-like storage to collect data while the PCI-416 is busy reading the A/D FIFO (and briefly cannot service the 82C55).

6.1 Modes

The 82C55 consists of three data registers and a single control register. All four registers are mapped in the PCI-416's Base Address Register 4 (BADR4). The 82C55 is controlled by first writing a mode byte to the control register then access the data registers. After this initialization, the 82C55 operates in **three major modes.** If the mode is not changed, no further programming of the control register is required. The three modes have a large number of options, all selected by bit programming the byte in the control register.

The three modes are:

Basic Input/Output (Mode 0)
Strobed Input/Output (Mode 1)
Strobed Bidirectional Bus I/O (Mode 2)

The 82C55 has three features which enable these modes. Besides being highly programmable, the 82C55 can reassign its data I/O pins on port C to be status and handshake lines to sequence data transfers. Finally, the 82C55 contains single-byte storage registers so that data will not be lost and the host computer can keep track of data transfers by an external device.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Since you cannot actually make the bytes write backwards in the disk sectors, the entire buffer pointer is simply rotated circularly *by blocks* in a reverse direction for the odd blocks only. The even blocks are filled completely then the odd blocks are filled backwards *by blocks* (but forwards by bytes). This same method can be extended to serpentine tape drives with many of the same design issues. This offers event-driven storage in the gigabytes.

Delayed Trigger

A certain class of applications needs a fixed delay after an external event before sampling starts. The PCI-416 does not directly include such a delay timer. However the pre/post trigger system discussed above can easily be adapted. Simply compute how many samples the external delay results in then set the sample counter for this number of samples plus a sufficient number of samples after the trigger so that old samples are not overwritten. The PCI-416 does not have a double trigger however the external "gate" trigger may be adapted to a single logic input which remains asserted during an external event. The PCI-416's delayed count down system can also be combined with the user's external logic gate which shuts off A/D clocks from an outside event. If possible, route this gate over to a spare input bit on the digital I/O port and poll this bit with each FIFO block transfer. Empty the FIFO when software detects that the external A/D clock has stopped.

One other method is to route the external event to a digital I/O port input bit. Start the application but do not enable the A/D yet. Poll this bit at high speed waiting for the start. When the external bit gets set, start A/D sampling then revert to the pretrigger sample counter system. A second external input on the trigger input then starts the down counter. This method may miss a few high speed samples at the beginning.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Another difficulty is the transition during the *last* group of samples. Although we know when the ACQuire flag occurs (and therefore when the A/D stops sampling), this does not directly tell us *which sample* in the FIFO is the trigger sample. We need a method to determine the address of this trigger sample in the ring buffer. This can only be done by *counting backwards circularly from the last sample.* So the key is to find the last sample.

For highest speed, use the half full FIFO signal and perhaps, the DMA mode for fast transfers. But consider that as we near the end of the sample count, we will probably only get a partially filled FIFO when the A/D clock shuts off. Any data still left in the FIFO will interfere with the way we compute that last sample address. So our software must be "on the lookout" for this end of count situation *probably by polling the ACQuire flag with each block transfer.* Once the ACQuire flag sets, the host software should transition to a mode which collects all samples until the FIFO is empty *and* continues the circular addressing.

In fact, when sampling is all done, we need **four pointers**: the head and tail pointers, the last sample pointer and the relative circular address of the "trigger" sample. Once we have all four pointers, we can derive any information we need. Note that either the pretrigger or the post trigger samples may be "skewed" (split) in circular fashion over the head or tail pointers.

Some applications need the final results "**normalized**" so that the *first* pretrigger sample is *lowest* in the ring buffer at relative offset zero. This avoids ever having older samples before newer ones and simplifies access to samples without being skewed. Normalization can be done by physically rotating the entire buffer until the first pretrigger sample is located at the head pointer. Or special functions can be written which extract data from the ring buffer using trigger-relative offsets but do not require physically moving the buffer contents. Refer to DATEL's PC-430 and DVME-630 software for more details on ring buffering.

The ring buffering should make the circular memory *an integer multiple of the half-FIFO size.* Always transfer one half the FIFO *without using circular* addressing during the half block transfer. This will prevent "skewing" a block over the head and tail and having to compute whether each sample should be rotated around to the bottom of the ring. Both the DMA system and the "string block move" assembly instruction used to transfer half blocks, do not have provision for ring buffering. So the circular addressing must all be precomputed *before* starting the block move. A "near" string block move automatically does circular addressing within a 64K block.

Huge Circular "Disk File" Storage

For truly immense circular storage requirements which still wish to use the pre/post trigger circular techniques described above, consider creating a large "ring buffer" on a **hard disk** file. Obviously this will be much slower than the all-memory version but can accommodate hundreds of millions of samples. Be aware of file size limits on some systems. Most systems accept "fseek/lseek" 32-bit pointers. First be sure the disk has been defragmented and the sector interleave has been optimized. This forces the sectors to be physically adjacent. Applications include long term data loggers and crash test recorders which wait for some external event then only need the last few hours or days of data.

If a single disk file is used and the speed is near the maximum available for the disk drive, there will be an interval at the "tail pointer" when the system must mechanically seek backwards many cylinders to find the "head pointer". Unlike semiconductor memory which can switch pointers instantly, this single extra delay may be sufficient to lose data. Consider a redesign of the system which uses a file with numbered odd and even blocks. Even blocks are accessed forward then odd blocks are accessed backward. The effect is that the mechanical read/write head sweeps forward then backward without ever moving more than a cylinder at a time. This sets the "seek" delay to some smaller average which never exceeds a fixed amount. Since you can't see the read/write head on the disk drive, you can verify that the system is working properly at high A/D speed by listening to the disk drive issuing seek commands. They should run indefinitely without noticeable variation in rate. (Use a stethoscope if needed). You can also write software timing loops to test the average delay to read the ACQuire flag.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Most systems however, do not have enough memory to collect data in a large linear buffer as described above *or* the time when a trigger occurs is not known with precision. To keep memory requirements within bounds, we constantly reuse old memory by overwriting the earliest samples in a ***ring buffer*** also called ***circular memory.*** This data structure is formed in a large but finite section of memory and is bounded by head and tail pointer addresses. Refer to Figure 5.8.

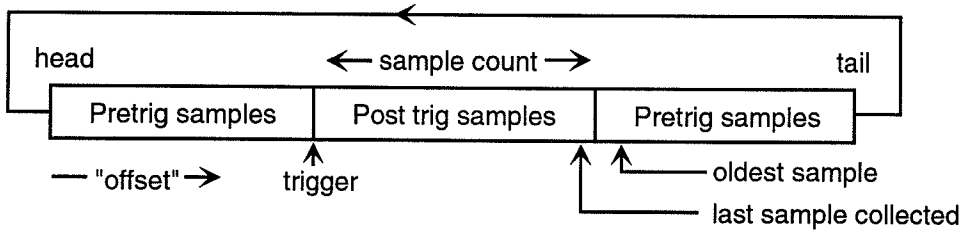


Figure 5.8 A/D Data Ring Buffering

A/D samples are continuously written to this memory in a rotating fashion such that at any one time, the buffer contains N number of samples. A "current sample" pointer keeps track of where the most recent sample was written to memory. The size of the ring buffer is computed by the sum of pretrigger plus post trigger samples. We assume that the total capacity in this buffer (N samples) is sufficient for the application.

This process is started and continues indefinitely until an external trigger occurs. The sample counter now counts down and when a count of zero is reached, the system stops. There are several practical details to make this system feasible. Some applications only care about the pretrigger samples and need a small number of postrigger samples while other applications need the reverse. Since we can only directly control the post-trigger sample counter, the key is to make sure the whole ring buffer collects the ***sum*** of the pretrigger and post trigger samples. You may also use the Marker system to flag the data.

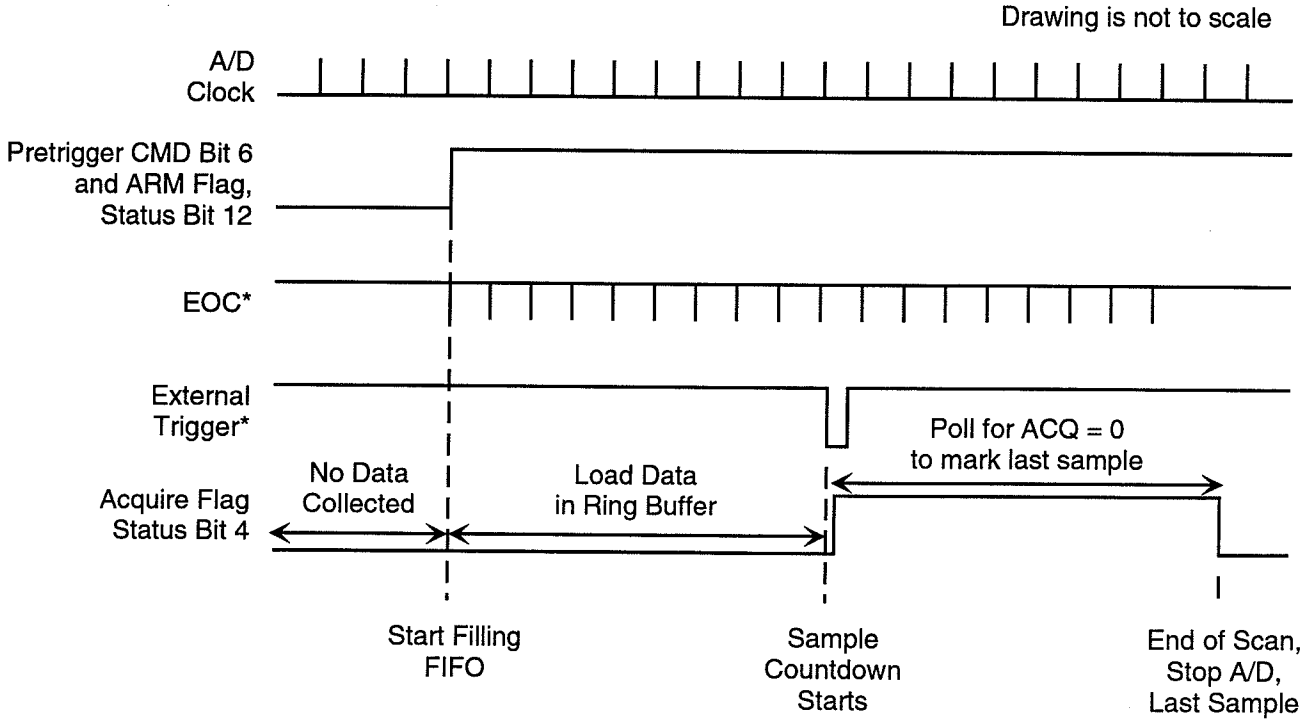


Figure 5.9 Pre-Trigger Timing Diagram

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Non-stop Sampling

The sample count system may also be *disabled.* This is used for streaming "gapless" applications where the A/D samples continuously, non-stop "forever". Examples include long-term high speed recording systems, communications receivers and satellite links, etc. Sampling is then stopped by external software. Such non-stop sampling is done as follows:

1. Select the A/D clock rate if the internal sample clock is used. For external clocking, start the A/D clock ticking. Leave the A/D disabled in register BADR2, ARL = 12. Clear the FIFO and prepare the host to receive data. Set up the interrupt and DMA systems if desired.
2. Enable the A/D to start conversions.
3. Set the pretrigger mode by writing a logic ONE into Command bit 6. The FIFO will begin filling with the next A/D clock. The host should transfer the FIFO when the half full flag appears. You may optionally reverse the sequence of these two operations, that is, set the pretrigger mode then enable A/D conversions. Note that *both* these registers must be enabled to collect A/D data.
4. *Do not send a trigger.* If you send a trigger, the sample count will now start down and A/D conversions will stop when the count is zero.
5. If desired, you may count samples externally by counting the number of FIFO blocks read.
6. When all samples have been taken, disable the A/D.

Pretrigger Mode

On the PCI-416 side, pretrigger mode is basically a variation of non-stop sampling described above *except* that a count is loaded into the sample counter before starting A/D conversions *and* a trigger is used. Pretrigger mode collects data **before and after an external event**. The PCI-416's sample count system is delayed as described above. When the external event occurs, a single trigger input is sent and the sample counter now begins to count down. A/D conversion ceases when the sample count reaches zero.

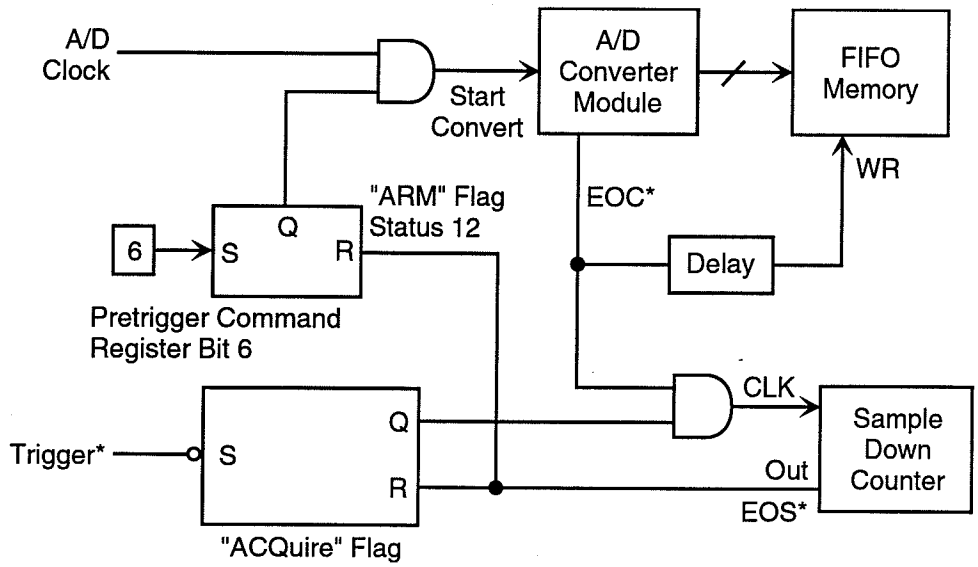


Figure 5.7 Pre-Trigger Mode Gating and Sample Counting

Pretrigger mode depends on cooperation between the PCI-416 and host software. If you know that the whole sampling session will not exceed some total period of time *and* you have sufficient memory capacity to store all possible samples, use a long linear buffer in the host. The application will now consist of these milestones: Sampling starts and at some undetermined (but maximum) time later into the sampling, an external event trigger occurs. This then initiates the sample down count and A/D collection terminates when this count reaches zero. The total memory capacity for A/D samples must be equal to both the time before and after the trigger.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The 24-bit sample count is loaded once into the Sample Count Register. ***Load one less than the desired count into the Sample Counter and do not load less than 2.*** For 256 samples, load 255. After a very short delay, the trigger sets the ACQuire flag to indicate that the frame has started. When ACQuire resets to zero, the frame is complete. This "End of Scan" (EOS) condition can also be directed to a PCI bus interrupt. See the system timing diagram:

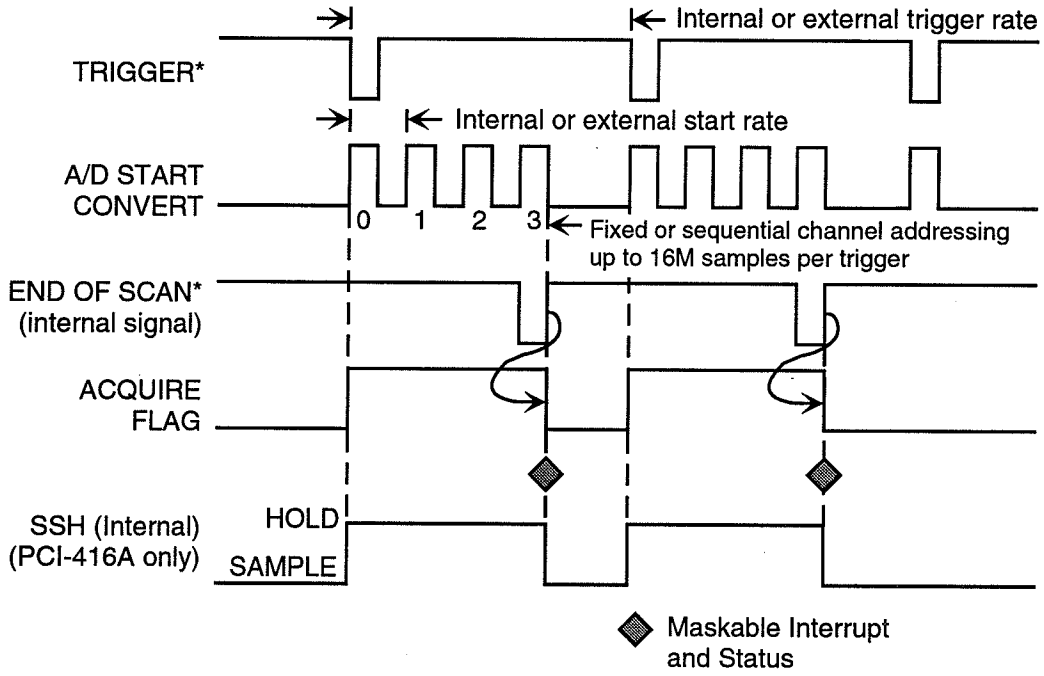


Figure 5.6 System Timing Diagram

You must compute the time period for each frame then avoid sending new triggers while a frame is in progress. Compute the time as follows:

Frame_period = A/D_sample_interval X samples_per_frame

The trigger period must exceed this frame period. ***Do not send new triggers before the previous frame has finished.*** Undefined operation may result. If the next trigger is too close to the end of the last frame, you may truncate sampling or cause other problems. Allow at least 25 nanoseconds from the last ACQuire reset to the next trigger. Remember that the channel address resets to zero at the end of each frame.

Since there is no channel address or frame number tagging of the data, you cannot determine the start of a new frame in the FIFO if you get out of sequence.

IMPORTANT: To positively identify the start of sampling, test the FIFO empty flag. If you are periodically polling the ACQuire flag to determine when a frame is done, there is a short delay between the actual start of sampling and when the ACQuire flag first gets set. Therefore, with extremely fast host software, it is possible to test ACQuire so quickly that it has not been set yet. This can be misinterpreted as a "frame done" condition. So the proper algorithm is as follows:

1. Set up everything then start A/D sampling.
2. Test the FIFO empty flag to verify that sampling has started. Consider using a large software counter to exit in case sampling did not start for some reason.
3. Transfer FIFO blocks and test the ACQuire flag with each block.
4. Empty the FIFO when ACQuire occurs.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP