

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.

Visit our website - *Click HERE*

PC-HELPER

High Resolution & Speed Analog I/O Board for PCI

ADA16-32/2(PCI)F

User's Guide

CONTEC CO., LTD.

Check Your Package

Thank you for purchasing the CONTEC product.

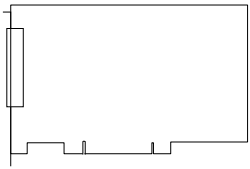
The product consists of the items listed below.

Check, with the following list, that your package is complete. If you discover damaged or missing items, contact your retailer.

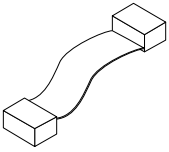
Product Configuration List

- Board [ADA16-32/2(PCI)F] …1
- First step guide … 1
- Disk *1 [API-PAC(W32)] …1
- Synchronization Control Cable (10cm) …1
- Warranty Certificate …1
- Serial Number Label …1

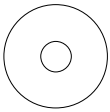
*1 The bundled disk contains the driver software and User's Guide (this guide)



ADA16-32/2(PCI)F



SC Cable



Disk
[API-PAC(W32)]



First step guide



Warranty Certificate



Serial Number Label

Copyright

Copyright 2003 CONTEC CO., LTD. ALL RIGHTS RESERVED

No part of this document may be copied or reproduced in any form by any means without prior written consent of CONTEC CO., LTD.

CONTEC CO., LTD. makes no commitment to update or keep current the information contained in this document. The information in this document is subject to change without notice.

All relevant issues have been considered in the preparation of this document. Should you notice an omission or any questionable item in this document, please feel free to notify CONTEC CO., LTD.

Regardless of the foregoing statement, CONTEC assumes no responsibility for any errors that may appear in this document or for results obtained by the user as a result of using this product.

Trademarks

MS, Microsoft, Windows and Windows NT are trademarks of Microsoft Corporation. Other brand and product names are trademarks of their respective holder.

Table of Contents

Check Your Package i

Copyright ii

Trademarks ii

Table of Contents iii

1. BEFORE USING THE PRODUCT 1

About the Board 1
- Features 1
- Support Software 3
- Cable & Connector (Option) 4
- Accessories (Option) 4

Customer Support 5
- Web Site 5

Limited Three-Years Warranty 5

How to Obtain Service 5

Liability 5

Safety Precautions 6
- Safety Information 6
- Handling Precautions 7
- Environment 8
- Inspection 8
- Storage 8
- Disposal 8

2. SETUP 9

What is Setup? 9
- Using the Board under Windows Using the Driver Library API-PAC(W32) 9
- Using the Board under Windows Using Software Other than the Driver Library API-PAC(W32) 9
- Using the Board under an OS Other than Windows 10

Step 1 Installing the Software 11
- Which Driver to Use 11
- Starting the Install Program 12

Step 2 Setting the Hardware 14
- Parts of the Board and Factory Defaults 14
- Setting the Board ID 15
- Plugging the Board 16

Step 3 Installing the Hardware 17
- Turning on the PC 17
- When Using API-AIO(WDM) 17

Step 4 Initializing the Software 19
When Using API-AIO(WDM) 19
Step 5 Checking Operations with the Diagnosis Program 21
What is the Diagnosis Program? 21
Check Method 21
Using the Diagnosis Program 22
Setup Troubleshooting 25
Symptoms and Actions 25
If your problem cannot be resolved 25

3. EXTERNAL CONNECTION 27

How to connect the connectors 27
Connector shape 27
Connector Pin Assignment 28
Analog Input Signal Connection 32
Single-ended Input 32
Differential Input 33
Analog Output Signal Connection 34
Digital I/O signals, Counter signals and Control signals Connection 35
Synchronization Control Connectors 36
SC Connectors 36
Connecting the SC Connectors (CN2 and CN3) 37

4. FUNCTIONS 39

Analog Input Function 39
1. Setting the Conversion Conditions 40
2. Starting/Stopping Operation 51
3. Monitoring the Status and Acquiring Data 51
4.Reset 55
Analog Output Function 56
1. Setting the Conversion Conditions 57
2. Starting/Stopping Operation 66
3. Monitoring the Status and Acquiring Data 67
4.Reset 67
Counter Function 69
1. Setting the Operating Conditions 69
2. Starting/Stopping Operation/Preset 71
3. Monitoring the Status and Acquiring Data 71
4.Reset 71
Digital Input Function 72
Digital Output Function 73
Event Controller Function 74

5. ABOUT SOFTWARE 79

Bundled disk Directory Structure 79

About Software for Windows 80
Accessing the Help File 80
Using Sample Programs 81
Usage of Utility Program 84
Uninstalling the Driver Libraries 89

About Software for Linux 90
Driver Software Install Procedure 90
Accessing the Help File 91
Using Sample Programs 91
Uninstalling the driver 91

6. ABOUT HARDWARE 93

For detailed technical information 93

Hardware specification 94

Block Diagram 96

Control Signal Timings 97
Control Signal Timings for Analog Input 97
Control Signal Timings for Analog Output 98
Control Signal Timings for Counter 99

About Calibration 100

1. Before Using the Product

About the Board

This product is a multi-function, PCI bus compliant interface board that incorporates high-precision analog inputs, high-precision analog outputs, digital inputs, digital outputs, and a counter function.

The board includes an event controller for integrated management of control signals by hardware and a bus master data transfer function for transferring large volumes of data at high speed. Together, these features provide all you need to build a high-performance PC-based measurement and control system.

You can use the driver library (API-PAC(W32)) supplied with the board to write Windows application programs in any programming language (such as Visual Basic, Visual C++, etc.) that supports the calling of Win32 API functions.

It can also collect data easily without a program when the data logger software [C-LOGGER] stored on the attached bundled disk is used. With plug-ins for the dedicated libraries, the board also supports MATLAB and LabVIEW.

Features

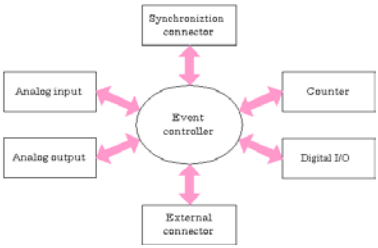
Multi-function

The board contains analog inputs (16-bit, 32ch), analog outputs (16-bit, 2ch), digital inputs (8ch), digital outputs (8ch), and counters (32-bit binary, 2ch). Combining all these features on one board allows complex systems to be implemented even on PCs with few spare expansion slots.

The event controller can be used to implement a wide range of different sampling control schemes

The board incorporates an event controller for integrated hardware control. The event controller can use the external control signals and the events generated by the board functions to start and stop analog input operation and perform clock control. This enables high-precision synchronization of the various board functions without requiring software. Also, each function can be operated separately.

Overview of event controller



The arrows in the figure indicate the flow of control signals. The main control signals included clock signals and the operation start and stop signals.

Example 1: Synchronize the timing of analog input and analog output based on an external clock signal.

Example 2: Start analog input operation each time the counter reaches a preset value.

Bus master transfer function and combined data I/O function

Bus master data transfer can be used for the analog inputs and outputs either separately or at the same time. This can be used to transfer large volumes of data between the board and PC without placing a load on the CPU.
When using bus master data transfer for analog input data, you can also transfer the analog output, digital input, digital output, and counter data at the same time synchronized with the analog input clock signal.
This function ensures reliable data synchronization in the systems you implement.

Buffer memory available for background processing independent of software

The analog inputs and outputs each have their own buffer memory which can be used when not using bus master transfer.
You can also perform analog input and output in the background, independent of software and the current status of the PC.

Software-based calibration

Calibration of analog input/output can be all performed by software. Apart from the adjustment information prepared before shipment, additional adjustment information can be stored according to the use environment.

Synchronization control connector provided for synchronizing operation

A synchronization control connector is provided for synchronized control of multiple boards. You can expand the number of channels simply by adding more boards.
This makes it easy to synchronize operation with other CONTEC boards that have a synchronization control connector.

Filter function for easy connection of external signals

The digital input signals, counter input signals, and the external control signals for analog I/O incorporate a digital filter to prevent problems such as chattering.

The same systems can be implemented on either desktop or notebook PCs

The "Analog F Series" boards (ADA16-32/2(PCI)F and ADA16-32/2(CB)F) have equivalent functionality.
Systems developed on a desktop PC can be ported directly to a notebook PC with minimal changes.

Supported to the data logger software [C-LOGGER]

Supporting the data logger software [C-LOGGER] that enables the graph display of recorded signal data, file saving, and dynamic transfer to the spreadsheet software program “Excel”

Plug-ins for the dedicated libraries, the board also supports MATLAB and LabVIEW.

We offer a dedicated library [ML-DAQ], which allows you to use this product on MATLAB by The MathWorks as well as another dedicated library [VI-DAQ], which allows you to use the product on LabVIEW.
These dedicated libraries are available, free of charge (downloadable), on our web site.

Support Software

You should use CONTEC support software according to your purpose and development environment.

Windows version of analog I/O driver **API-AIO(WDM)**
[Stored on the bundled disk driver library API-PAC(W32)]

The API-AIO(WDM) is the Windows version driver library software that provides products in the form of Win32 API functions (DLL). Various sample programs such as Visual Basic and Visual C++, etc and diagnostic program *1useful for checking operation is provided.

For more details on the supported OS, applicable language and new information, please visit the CONTEC's Web site.

Linux version of analog I/O driver **API-AIO(LNX)**
[Stored on the bundled disk driver library API-PAC(W32)]

The API-AIO(LNX) is the Linux version driver software which provides device drivers (modules) by shared library and kernel version. Various sample programs of gcc are provided.

For more details on the supported OS, applicable language and new information, please visit the CONTEC's Web site.

Data Logger Software **C-LOGGER**
[Stored on the bundled disk driver library API-PAC(W32)]

C-LOGGER is a data logger software program compatible with our analog I/O products. This program enables the graph display of recorded signal data, zoom observation, file saving, and dynamic transfer to the spreadsheet software "Excel". No troublesome programming is required.

For more details on the supported OS, applicable language and new information, please visit the CONTEC's Web site.

Data Acquisition library for MATLAB **ML-DAQ**
(Available for downloading (free of charge) from the CONTEC web site.)

This is the library software which allows you to use our analog I/O device products on MATLAB by the MathWorks. Each function is offered in accordance with the interface which is integrated in MATLAB's Data Acquisition Toolbox. For more details on the supported OS, applicable language and new information, please visit the CONTEC's Web site.

Data acquisition VI library for LabVIEW **VI-DAQ**
(Available for downloading (free of charge) from the CONTEC web site.)

This is a VI library to use in National Instruments LabVIEW.
VI-DAQ is created with a function form similar to that of LabVIEW's Data Acquisition VI, allowing you to use various devices without complicated settings.
For more details on the library and download of VI-DAQ, please visit the CONTEC's Web site.

Cable & Connector (Option)

Shielded Cable with One 96-Pin Half-Pitch Connector	:	PCA96PS-0.5P (0.5m)
	:	PCA96PS-1.5P (1.5m)
Shielded Cable with 96-Pin Half-Pitch Connectors at Both Ends	:	PCB96PS-0.5P (0.5m)
	:	PCB96PS-1.5P (1.5m)
Flat Cable with One 96-Pin Half-Pitch Connector	:	PCA96P-1.5 (1.5m)
Flat Cable with 96-Pin Half-Pitch Connectors at Both Ends	:	PCB96P-1.5 (1.5m)

Accessories (Option)

Buffer Amplifier Box for Analog Input Boards (32ch type)	:	ATBA-32F *1*2
Buffer Amplifier Box for Analog Input Boards (8ch type)	:	ATBA-8F *1*2*3
Terminal Unit for Cables (M2.5 x 96P)	:	DTP-64(PC) *1
Screw Terminal Unit (M3.5 x 96P)	:	EPD-96 *1
Screw Terminal Unit (M3 x 96P)	:	EPD-96A *1*4
BNC Terminal Unit (for analog input 32ch)	:	ATP-32F *1
BNC Terminal Unit (for analog input 8ch)	:	ATP-8 *1*3*5

1 PCB96PS- optional cable is required separately (0.5mm is recommended).
*2 An external power supply is necessary (optional AC adaptor POA200-20 prepared.)
*3 The analog input could have 8 channels to be used.
*4 “Spring-up” type terminal is used to prevent terminal screws from falling off.
*5 The digital input can be used up to four points, the digital output up to four points and the counter I/O up to 1 channel.

* Check the CONTEC’s Web site for more information on these options.

Customer Support

CONTEC provides the following support services for you to use CONTEC products more efficiently and comfortably.

Web Site

https://www.contec.com/

Latest product information

CONTEC provides up-to-date information on products.

CONTEC also provides product manuals and various technical documents in the PDF.

Free download

You can download updated driver software and differential files as well as sample programs available in several languages.

Note! For product information
Contact your retailer if you have any technical question about a CONTEC product or need its price, delivery time, or estimate information.

Limited Three-Years Warranty

CONTEC products are warranted by CONTEC CO., LTD. to be free from defects in material and workmanship for up to three years from the date of purchase by the original purchaser.

Repair will be free of charge only when this device is returned freight prepaid with a copy of the original invoice and a Return Merchandise Authorization to the distributor or the CONTEC group office, from which it was purchased.

This warranty is not applicable for scratches or normal wear, but only for the electronic circuitry and original products. The warranty is not applicable if the device has been tampered with or damaged through abuse, mistreatment, neglect, or unreasonable use, or if the original invoice is not included, in which case repairs will be considered beyond the warranty policy.

How to Obtain Service

For replacement or repair, return the device freight prepaid, with a copy of the original invoice. Please obtain a Return Merchandise Authorization number (RMA) from the CONTEC group office where you purchased before returning any product.

* No product will be accepted by CONTEC group without the RMA number.

Liability

The obligation of the warrantor is solely to repair or replace the product. In no event will the warrantor be liable for any incidental or consequential damages due to such defect or consequences that arise from inexperienced usage, misuse, or malfunction of this device.

Safety Precautions

Understand the following definitions and precautions to use the product safely.

Safety Information

This document provides safety information using the following symbols to prevent accidents resulting in injury or death and the destruction of equipment and resources. Understand the meanings of these labels to operate the equipment safely.

⚠ DANGER	DANGER indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.
⚠ WARNING	WARNING indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.
⚠ CAUTION	CAUTION indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury or in property damage.

Handling Precautions

⚠ DANGER

Do not use the product where it is exposed to flammable or corrosive gas. Doing so may result in an explosion, fire, electric shock, or failure.

⚠ CAUTION

- There are switches and jumpers on the board that need to be set in advance. Be sure to check these before installing the board.
- Only set the switches and jumpers on the board to the specified settings.
  Otherwise, the board may malfunction, overheat, or cause a failure.
- Do not strike or bend the board. Doing so could damage the board.
  Otherwise, the board may malfunction, overheat, cause a failure or breakage.
- Do not touch the board's metal plated terminals (edge connector) with your hands.
  Otherwise, the board may malfunction, overheat, or cause a failure.
  If the terminals are touched by someone's hands, clean the terminals with industrial alcohol.
- Do not plug or unplug the cables which are connected to the board while the PC or expansion unit is still turned on.
  Otherwise, the board may malfunction, overheat, or cause a failure.
  Be sure that the personal computer power is turned off.
- Make sure that your PC or expansion unit can supply ample power to all the boards installed.
  Insufficiently energized boards could malfunction, overheat, or cause a failure.
- The specifications of this product are subject to change without notice for enhancement and quality improvement.
  Even when using the product continuously, be sure to read the manual and understand the contents.
- Do not modify the product. CONTEC will bear no responsibility for any problems, etc., resulting from modifying this product.
- Regardless of the foregoing statements, CONTEC is not liable for any damages whatsoever (including damages for loss of business profits) arising out of the use or inability to use this CONTEC product or the information contained herein.
- Regarding “CE EMC Directive Notice”.
  Please connect the Interface Connector with a shielded cable to meet the mentioned standard above.

Environment

Use this product in the following environment. If used in an unauthorized environment, the board may overheat, malfunction, or cause a failure.

Operating temperature

0 - 50°C

Operating humidity

10 - 90%RH (No condensation)

Corrosive gases

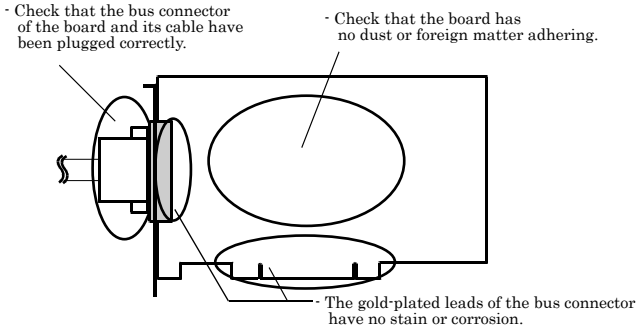
None

Floating dust particles

Not to be excessive

Inspection

Inspect the product periodically as follows to use it safely.



Storage

When storing this product, keep it in its original packing form.

(1) Put the board in the storage bag.
(2) Wrap it in the packing material, then put it in the box.
(3) Store the package at room temperature at a place free from direct sunlight, moisture, shock, vibration, magnetism, and static electricity.

Disposal

When disposing of the product, follow the disposal procedures stipulated under the relevant laws and municipal ordinances.

2. Setup

This chapter explains how to set up the board.

What is Setup?

Setup means a series of steps to take before the product can be used.
Different steps are required for software and hardware.
The setup procedure varies with the OS and applications used.

Using the Board under Windows

Using the Driver Library API-PAC(W32)

This section describes the setup procedure to be performed before you can start developing application programs for the board using the bundled disk "Driver Library API-PAC(W32)".

Taking the following steps sets up the software and hardware. You can use the diagnosis program later to check whether the software and hardware function normally.

Step 1 Installing the Software

Step 2 Setting the Hardware

Step 3 Installing the Hardware

Step 4 Initializing the Software

Step 5 Checking Operations with the Diagnosis Program

If Setup fails to be performed normally, see the "Setup Troubleshooting" section at the end of this chapter.

Using the Board under Windows

Using Software Other than the Driver Library API-PAC(W32)

For setting up software other than API-PAC(W32), refer to the manual for that software. See also the following parts of this manual as required.

This chapter Step 2 Setting the Hardware

This chapter Step 3 Installing the Hardware

Chapter 3 External Connection

Chapter 6 About Hardware

Using the Board under an OS Other than Windows

For using the board under Linux, see the following parts of this manual.

This chapter Step 2 Setting the Hardware

Chapter 3 External Connection

Chapter 5 About Software

Chapter 6 About Hardware

For using the board under an OS other than Windows and Linux, see the following parts of this manual.

This chapter Step 2 Setting the Hardware

Chapter 3 External Connection

Chapter 6 About Hardware

Step 1 Installing the Software

This explains how to install the driver library.

Before installing the hardware in a PC, install "Driver Library API-PAC(W32)" from the bundled disk or download and install the latest edition of this software from the CONTEC web site.

The following description assumes the operating system as Windows XP. Although some user interfaces are different depending on the OS used, the basic procedure is the same.

Which Driver to Use

CONTEC has two analog I/O drivers: API-AIO(WDM) and API-AIO(98/PC).

API-AIO(WDM) is a new driver for analog I/O under Windows.

This driver was developed to be easier to use and to provide additional functions above those provided by the previous API-AIO(98/PC) driver.

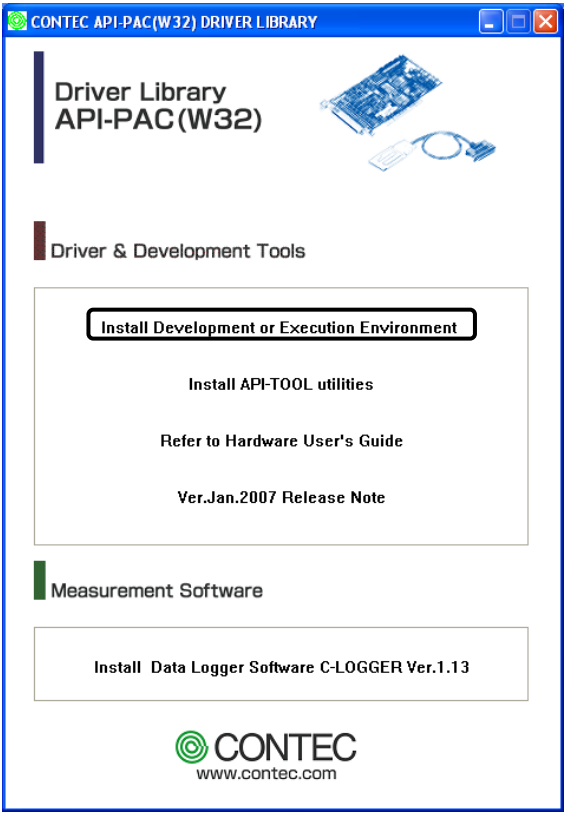
Please use the API-AIO(WDM) with this board. API-AIO(98/PC) is not supported.

Starting the Install Program

(1) **Load the bundled disk [API-PAC(W32)] on your PC.**

(2) **The API-PAC(W32) Installer window appears automatically.**
If the panel does not appear, run (drive letter):\AUTORUN.exe.

(3) **Click on the [Install the drivers] button.**



⚠ CAUTION

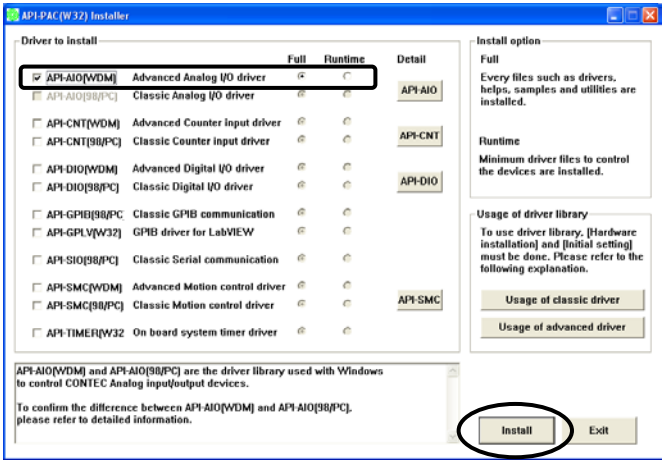
Before installing the software in Windows Vista, XP, Server 2003, 2000, log in as a user with administrator privileges.

Selecting API-AIO(WDM)

(1) **The following dialog box appears to select “Driver to install” and “Install option”, “Usage of driver library”.**

(2) **Select the "High Functionality WDM Analog I/O driver".**

(3) **Click on the [Install] button.**



* Clicking the [Details] button displays detailed information about API-AIO(WDM) and API-AIO(98/PC).

Run the installation

(1) **Complete the installation by following the instructions on the screen.**

(2) The Readme file appears when the installation is complete.

You have now finished installing the software.

Step 2 Setting the Hardware

This section describes how to set the board and plug it on your PC.

The board has some switches and jumper to be preset.
Check the on-board switches and jumpers before plugging the board into an expansion slot.

The board can be set up even with the factory defaults untouched. You can change board settings later.

Parts of the Board and Factory Defaults

Figure 2.1 shows the names of each part of the board.

The switch settings shown in the figure are the factory default settings.

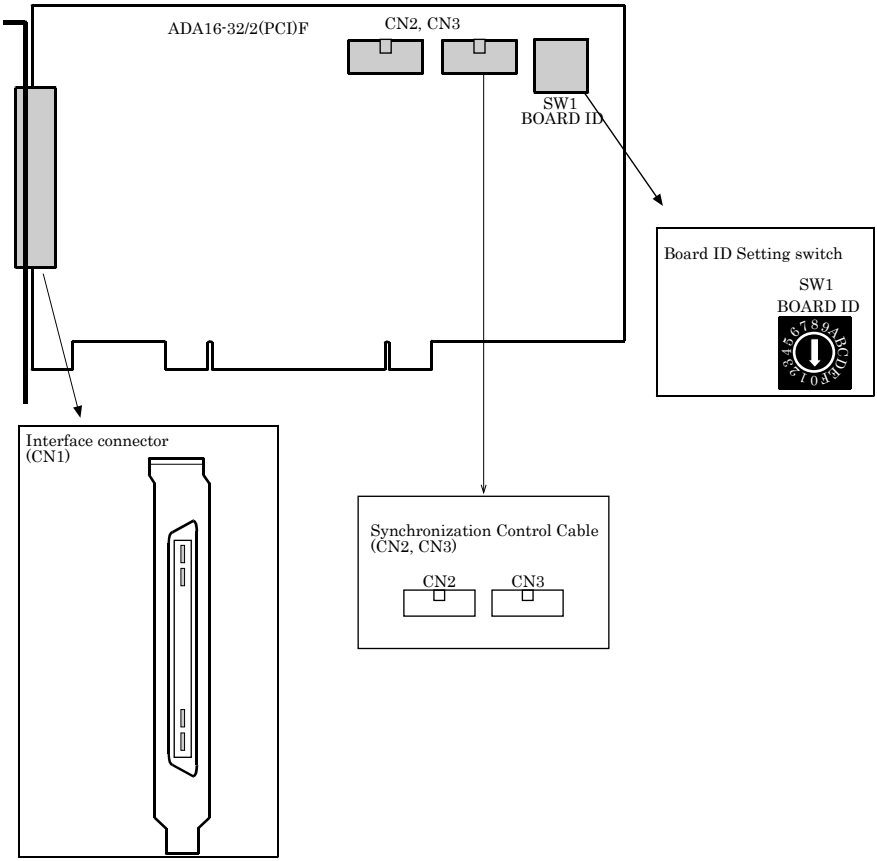


Figure 2.1. Part Names

Setting the Board ID

If you install two or more boards on one personal computer, assign a different ID value to each of the boards to distinguish them.

The board IDs can be set from 0 - Fh to identify up to sixteen boards.

If only one board is used, the original factory setting (Board ID = 0) should be used.

Setting Procedure

To set the board ID, use the rotary switch on the board. Turn the SW1 knob to set the board ID as shown below.

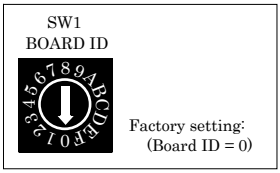


Figure 2.2. Board ID Settings (SW1)

Plugging the Board

(1) Before plugging the board, shut down the system, unplug the power code of your PC.

(2) Remove the cover from the PC so that the board can be mounted.

(3) Plug the board into an expansion slot.

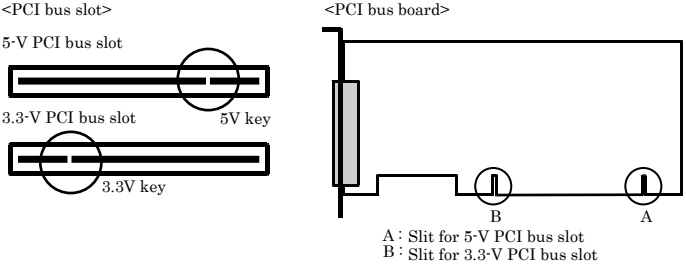
(4) Attach the board bracket to the PC with a screw.

(5) Put the cover back into place.



Applicable PCI bus slots

PCI bus slots used in PCs have keys to prevent 5V and 3.3V PCI bus boards from being accidentally plugged into wrong bus slots. This board can be plugged into both of the 5V and 3.3V PCI bus slots.



⚠ CAUTION

- Do not touch the board's metal plated terminals (edge connector) with your hands.
  Otherwise, the board may malfunction, overheat, or cause a failure.
  If the terminals are touched by someone's hands, clean the terminals with industrial alcohol.
- Do not install or remove the board to or from the slot while the computer's power is turned on.
  Otherwise, the board may malfunction, overheat, or cause a failure.
  Doing so could cause trouble. Be sure that the personal computer or the I/O expansion unit power is turned off.
- Make sure that your PC or expansion unit can supply ample power to all the boards installed.
  Insufficiently energized boards could malfunction, overheat, or cause a failure.
- Power supply from the PCI bus slot at +5V is required.

Step 3 Installing the Hardware

Windows needs to detect the I/O address and interrupt used by the board. This is called hardware installation.

When using more than one board, install the boards one at a time and do not install the next board until setup is complete for the previous board.

Turning on the PC

Turn on the power to your PC.

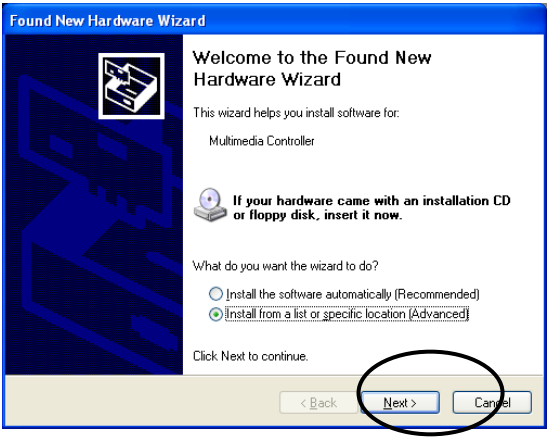
⚠ CAUTION

- The board cannot be properly installed unless the resources (I/O addresses and interrupt level) for the board can be allocated. Before attempting to install the board, first determine what PC resources are free to use.
- The resources used by each board do not depend on the location of the PCI bus slot or the board itself. If you remove two or more boards that have already been installed and then remount one of them on the computer, it is unknown that which one of the sets of resources previously assigned to the two boards is assigned to the remounted board. In this case, you must check the resource settings.

When Using API-AIO(WDM)

(1) When the " Found New Hardware Wizard" opens, select "Install the software automatically [Recommended]" and then click the "Next" button.

* The wizard may not appear for some OS versions and instead the installation will start automatically. In this case, proceed to the software initial setup step.



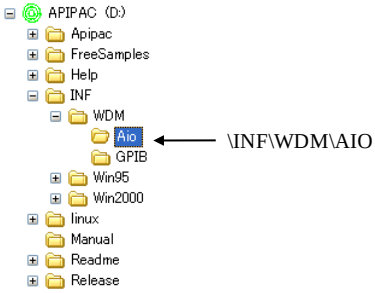
< If using Windows 98 or Windows Me >

If using Windows 98 or Windows Me, specify the folder that contains the setup information (INF) file from the **bundled disk**

Source folder

The setup information (INF) file is contained in the following folder on the bundled **bundled disk**.

\INF\WDM\AIO



You have now finished installing the hardware.

Step 4 Initializing the Software

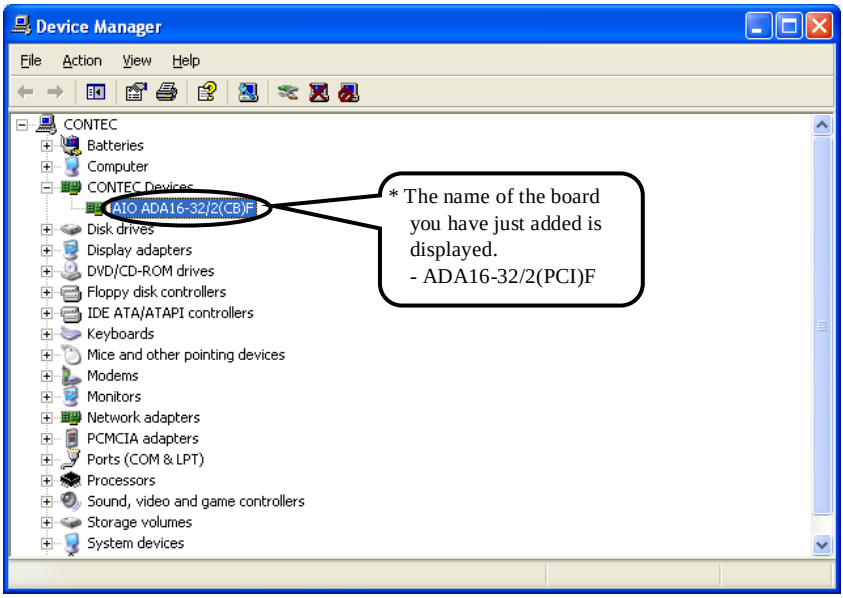
The driver library requires initial settings to determine the execution environment. This is called driver library initialization.

When Using API-AIO(WDM)

API-AIO(WDM) is initialized automatically during hardware installation. Therefore, if you want to use it with its initial settings, you can skip the setting procedure described in Step 4. To change the device name, follow the setting procedure shown below.

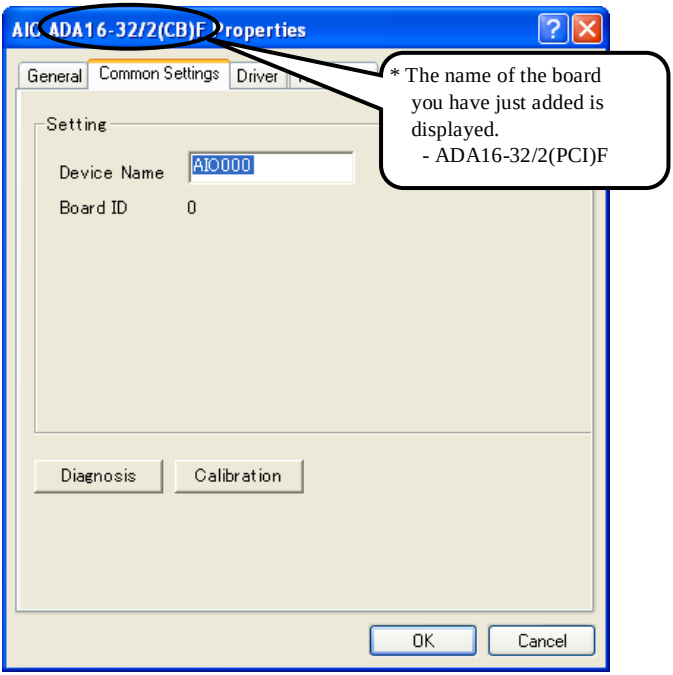
Setting the device name

(1) Run Device Manager. From [My Computer] - [Control Panel], select [System] and then select the [Device Manager] tab.
(You can also open Device Manager by right clicking on My Computer and selecting Properties.)



(2) The installed hardware appears under the CONTEC Devices node. Open the CONTEC Devices node and select the device you want to setup (the device name should appear highlighted). Click [Properties].

(3) The property page for the device opens.
Enter the device name in the common settings tab page and then click [OK].
The device name you set here is used later when programming.



* The initial device name that appears is a default value. You can use this default name if you wish.

* Make sure that you do not use the same name for more than one device.

You have now finished installing the initial setting of Software.

Step 5 Checking Operations with the Diagnosis Program

Use the diagnosis program to check that the board and driver software work normally, thereby you can confirm that they have been set up correctly.

What is the Diagnosis Program?

The diagnosis program diagnoses the states of the board and driver software.
It can also be used as a simple checker when an external device is actually connected.
Using the “Diagnosis Report” feature reports the driver settings, the presence or absence of the board, I/O status, and interrupt status.

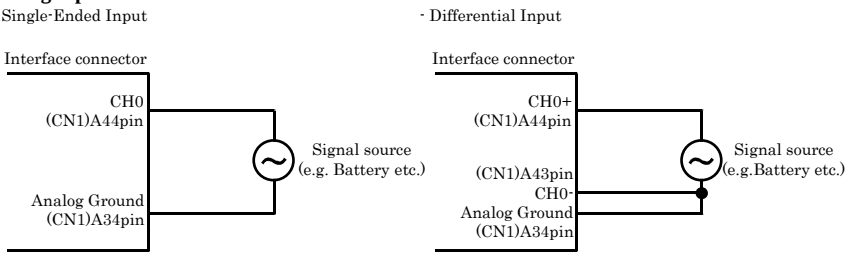
Check Method

To check the analog I/O data, use the loopback connection on the board to directly connect the analog output (channel 0) to an analog input, or connect to an external signal source.
No external connection is required if using loopback to check the I/O.

The figure below shows an example of checking by connecting to an external signal.
The example is for channel 0 on the ADA16-32/2(PCI)F. See "Chapter 3 External Connection" for details on how to make the connection.

Wiring Diagram

< Analog input >



⚠ CAUTION

- Input data remains indeterminate when no input pin is connected. The input pin for the channel not connected to the signal source must be connected to the analog ground.
  For details, see “Chapter 3 External Connection”.

< Analog output >

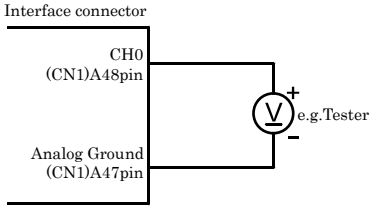
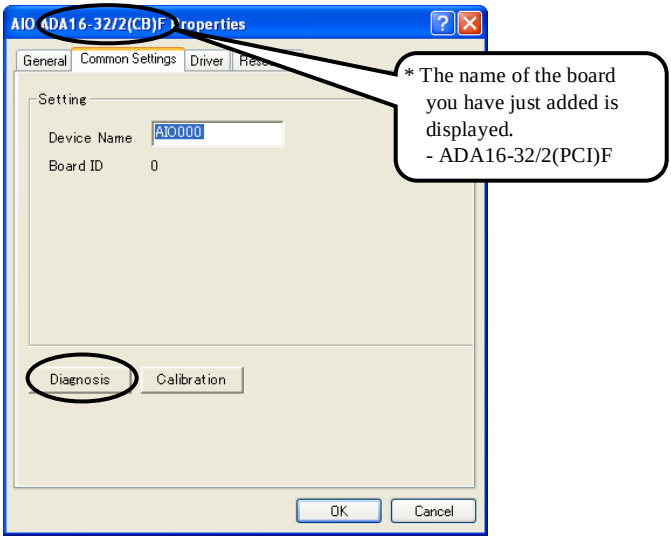


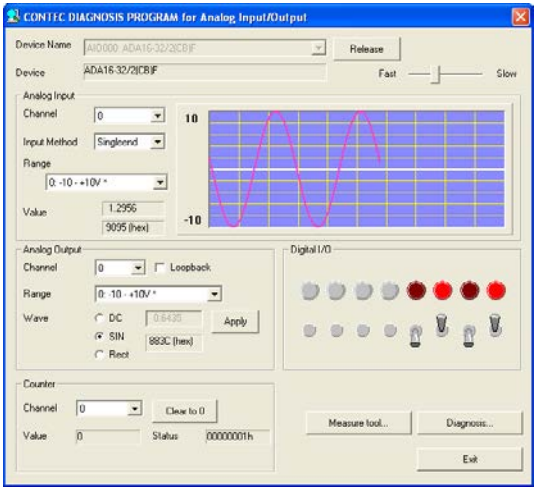
Figure 2.3. Wiring Diagram

Using the Diagnosis Program

Starting the Diagnosis Program

Click the [Diagnosis] button on the device property page to start the diagnosis program.





Analog input

Select the input channel, input type, and input range from the lists.

Input data is plotted on a graph.

Analog output

You can select the desired output channel and output range from the lists.

You can set the output data to DC (constant voltage), sine wave, or square wave.

By setting loopback ON, you can reconfigure the internal board hardware so that the channel 0 output signal is connected to all analog input channels.

Digital input / output

The upper row of circular lamps indicates the digital input states. Red indicates the bit is ON and brown indicates OFF.

Clicking the lower row of switches turns the digital output bits ON or OFF.

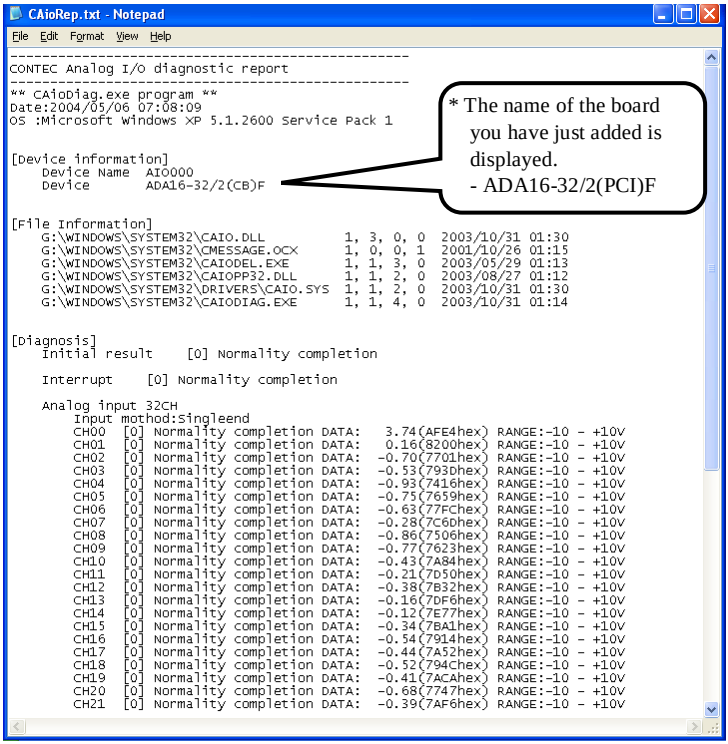
Counter input

Selecting a counter channel displays the count value and state of that counter channel.
Clicking the zero clear button resets the count to zero.

Diagnosis Report

(1) The diagnosis report saves detailed data, including the device settings and settings for each channel, to a text file and displays the file for you to view.

Clicking [Diagnosis Report] prompts you to specify where to save the report text file.




```
------------------------------------------------------
CONTEC Analog I/O diagnostic report
------------------------------------------------------
** CAioDiag.exe program **
Date:2004/05/06 07:08:09
OS :Microsoft Windows XP 5.1.2600 Service Pack 1

[Device information]
    Device Name  AIO000
    Device       ADA16-32/2(CB)F

[File Information]
    G:\WINDOWS\SYSTEM32\CAIO.DLL          1, 3, 0, 0  2003/10/31 01:30
    G:\WINDOWS\SYSTEM32\CMESSAGE.OCX      1, 0, 0, 1  2001/10/26 01:15
    G:\WINDOWS\SYSTEM32\CAIODEL.EXE       1, 1, 3, 0  2003/05/29 01:13
    G:\WINDOWS\SYSTEM32\CAIOPP32.DLL      1, 1, 2, 0  2003/08/27 01:12
    G:\WINDOWS\SYSTEM32\DRIVERS\CAIO.SYS  1, 1, 2, 0  2003/10/31 01:30
    G:\WINDOWS\SYSTEM32\CAIODIAG.EXE      1, 1, 4, 0  2003/10/31 01:14

[Diagnosis]
    Initial result    [0] Normality completion

    Interrupt     [0] Normality completion

    Analog input 32CH
        Input mothod:singleend
        CH00  [0] Normality completion DATA:   3.74(AFE4hex) RANGE:-10 - +10V
        CH01  [0] Normality completion DATA:   0.16(8200hex) RANGE:-10 - +10V
        CH02  [0] Normality completion DATA:  -0.70(7701hex) RANGE:-10 - +10V
        CH03  [0] Normality completion DATA:  -0.53(793Dhex) RANGE:-10 - +10V
        CH04  [0] Normality completion DATA:  -0.93(7416hex) RANGE:-10 - +10V
        CH05  [0] Normality completion DATA:  -0.75(7659hex) RANGE:-10 - +10V
        CH06  [0] Normality completion DATA:  -0.63(77FChex) RANGE:-10 - +10V
        CH07  [0] Normality completion DATA:  -0.28(7C6Dhex) RANGE:-10 - +10V
        CH08  [0] Normality completion DATA:  -0.86(7506hex) RANGE:-10 - +10V
        CH09  [0] Normality completion DATA:  -0.77(7623hex) RANGE:-10 - +10V
        CH10  [0] Normality completion DATA:  -0.43(7A84hex) RANGE:-10 - +10V
        CH11  [0] Normality completion DATA:  -0.21(7D50hex) RANGE:-10 - +10V
        CH12  [0] Normality completion DATA:  -0.38(7B32hex) RANGE:-10 - +10V
        CH13  [0] Normality completion DATA:  -0.16(7DF6hex) RANGE:-10 - +10V
        CH14  [0] Normality completion DATA:  -0.12(7E77hex) RANGE:-10 - +10V
        CH15  [0] Normality completion DATA:  -0.34(7BA1hex) RANGE:-10 - +10V
        CH16  [0] Normality completion DATA:  -0.54(7914hex) RANGE:-10 - +10V
        CH17  [0] Normality completion DATA:  -0.44(7A52hex) RANGE:-10 - +10V
        CH18  [0] Normality completion DATA:  -0.52(794Chex) RANGE:-10 - +10V
        CH19  [0] Normality completion DATA:  -0.41(7ACAhex) RANGE:-10 - +10V
        CH20  [0] Normality completion DATA:  -0.68(7747hex) RANGE:-10 - +10V
        CH21  [0] Normality completion DATA:  -0.39(7AF6hex) RANGE:-10 - +10V
```

(2) The diagnosis report contains the following data.

- Version of OS
- Device Information
- File Information
- Initialization, interrupts, current input or output state for each channel

Setup Troubleshooting

Symptoms and Actions

Data input or output does not operate correctly

- Run the diagnosis program to check that the device is registered and whether any initialization errors have occurred.
- Is there a problem with the device settings, wiring, or similar? Check the I/O range setting. Also, the input data will be undefined if the wiring terminals are not connected. Ensure that the channels you are using are correctly connected. Connect unused channels to analog ground.
- For voltage input, check by connecting a battery or similar if you do not have any other suitable signal source. Also check that connecting to analog ground reads correctly as 0V.

The diagnostic program works correctly but the application program does not.

The diagnostic program uses the API-TOOL functions. If the diagnostic program works correctly, other applications should work correctly also. If you have a problem, recheck your program taking note of the following points.

- Check the return values of the API functions.
- Refer to the source code for the sample programs.

The OS does not boot correctly or does not detect the device correctly.

Refer to the "Troubleshooting" section of API-AIO(WDM) HELP.

If your problem cannot be resolved

Contact your retailer.

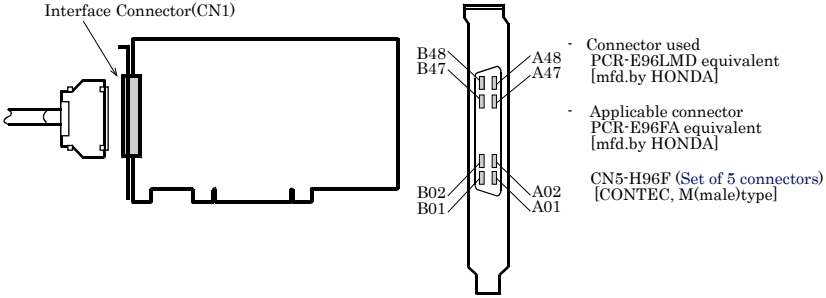
3. External Connection

This chapter describes the interface connectors on the board and the external I/O circuits.
Check the information available here when connecting an external device.

How to connect the connectors

Connector shape

To connect an external device to this board, plug the cable from the device into the interface connector (CN1) shown below.



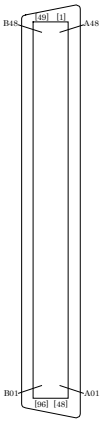
* Please refer to chapter 1 for more information on the supported cable and accessories.

Figure 3.1. Interface Connector Shape

Connector Pin Assignment

Pin Assignment of CN1 < Single-Ended Input >

N.C.	B48	A48	Analog Output 00
N.C.	B47	A47	Analog Ground (for AO)
N.C.	B46	A46	Analog Output 01
N.C.	B45	A45	Analog Ground (for AO)
Analog Input 08	B44	A44	Analog Input 00
Analog Input 24	B43	A43	Analog Input 16
Analog Input 09	B42	A42	Analog Input 01
Analog Input 25	B41	A41	Analog Input 17
Analog Ground (fro AI)	B40	A40	Analog Ground (for AI)
Analog Ground (for AI)	B39	A39	Analog Ground (for AI)
Analog Input 10	B38	A38	Analog Input 02
Analog Input 26	B37	A37	Analog Input 18
Analog Input 11	B36	A36	Analog Input 03
Analog Input 27	B35	A35	Analog Input 19
Analog Ground (for AI)	B34	A34	Analog Ground (for AI)
Analog Ground (for AI)	B33	A33	Analog Ground (for AI)
Analog Input 12	B32	A32	Analog Input 04
Analog Input 28	B31	A31	Analog Input 20
Analog Input 13	B30	A30	Analog Input 05
Analog Input 29	B29	A29	Analog Input 21
Analog Ground (for AI)	B28	A28	Analog Ground (for AI)
Analog Ground (for AI)	B27	A27	Analog Ground (for AI)
Analog Input 14	B26	A26	Analog Input 06
Analog Input 30	B25	A25	Analog Input 22
Analog Input 15	B24	A24	Analog Input 07
Analog Input 31	B23	A23	Analog Input 23
Analog Ground (for AI)	B22	A22	Analog Ground (for AI)
Analog Ground (for AI)	B21	A21	Analog Ground (for AI)
Digital Ground	B20	A20	Digital Ground
N.C.	B19	A19	N.C.
Digital Output 00	B18	A18	Digital Input 00
Digital Output 01	B17	A17	Digital Input 01
Digital Output 02	B16	A16	Digital Input 02
Digital Output 03	B15	A15	Digital Input 03
Digital Output 04	B14	A14	Digital Input 04
Digital Output 05	B13	A13	Digital Input 05
Digital Output 06	B12	A12	Digital Input 06
Digital Output 07	B11	A11	Digital Input 07
AO Control Signal Output 00	B10	A10	AI Control Signal Output 00
AO Control Signal Output 01	B09	A09	AI Control Signal Output 01
Digital Ground	B08	A08	Digital Ground
AO External Sampling Clock Input	B07	A07	AI External Sampling Clock Input
AO External Stop Trigger Input	B06	A06	AI External Stop Trigger Input
AO External Start Trigger Input	B05	A05	AI External Start Trigger Input
Counter UP Clock Input 01	B04	A04	Counter UP Clock Input 00
Reserved	B03	A03	Reserved
Counter Gate Control Input 01	B02	A02	Counter Gate Control Input 00
Control Output 01	B01	A01	Counter Output 00



- The numbers in square brackets [ ] are pin numbers designated by HONDA TSUSHIN KOGYO CO.,

Analog Input00 - Analog Input31	Analog input signal. The numbers correspond to channel numbers.
Analog Output00 - Analog Output01	Analog output signal. The numbers correspond to channel numbers.
Analog Ground	Common analog ground for analog I/O signals.
AI External Start Trigger Input	External trigger input for starting analog input sampling.
AI External Stop Trigger Input	External trigger input for stopping analog input sampling.
AI External Sampling Clock Input	External sampling clock input for analog input.
AI Control Signal Output 00	External sampling clock output signal for analog input.
AI Control Signal Output 01	External output signal for analog input status. Not currently connected.
AO External Start Trigger Input	External trigger input for starting analog output sampling.
AO External Stop Trigger Input	External trigger input for stopping analog output sampling.
AO External Sampling Clock Input	External sampling clock input for analog output.
AO Control Signal Output 00	External sampling clock output signal for analog output.
AO Control Signal Output 01	External output signal for analog output status. Not currently connected.
Digital Input00 - Digital Input07	Digital input signal.
Digital Output00 - Digital Output07	Digital output signal.
Counter Gate Control Input00 - Counter Gate Control Input01	Gate control input signal for counter.
Counter Up Clock Input00 - Counter Up Clock Input01	Count-up clock input signal for counter.
Counter Output00 - Counter Output01	Count match output signal for counter.
Digital Ground	Common digital ground for digital I/O signals, external trigger inputs, external sampling clock inputs, and counter I/O signals.
Reserved	Reserved pin
N.C.	No connection to this pin.

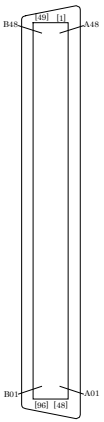
Figure 3.2. Pin Assignment of CN1 ***< 1 / 2 >***

⚠ CAUTION

- Do not connect any of the outputs and power outputs to the analog or digital ground. Neither connect outputs to each other. Doing either can result in a fault.
- If analog and digital ground are shorted together, noise on the digital signals may affect the analog signals. Accordingly, analog and digital ground should be separated.
- Leave "Reserved" pins unconnected. Connecting these pins may cause a fault in the board.

Pin Assignment of CN1 < Differential Input >

N.C.	B48	A48	Analog Output 00	
N.C.	B47	A47	Analog Ground (for AO)	
N.C.	B46	A46	Analog Output 01	
N.C.	B45	A45	Analog Ground (for AO)	
Analog Input 08[+]	B44	A44	Analog Input 00[+]	
Analog Input 08[-]	B43	A43	Analog Input 00[-]	
Analog Input 09[+]	B42	A42	Analog Input 01[+]	
Analog Input 09[-]	B41	A41	Analog Input 01[-]	
Analog Ground (for AI)	B40	A40	Analog Ground (for AI)	
Analog Ground (for AI)	B39	A39	Analog Ground (for AI)	
Analog Input 10[+]	B38	A38	Analog Input 02[+]	
Analog Input 10[-]	B37	A37	Analog Input 02[-]	
Analog Input 11[+]	B36	A36	Analog Input 03[+]	
Analog Input 11[-]	B35	A35	Analog Input 03[-]	
Analog Ground (for AI)	B34	A34	Analog Ground (for AI)	
Analog Ground (for AI)	B33	A33	Analog Ground (for AI)	
Analog Input 12[+]	B32	A32	Analog Input 04[+]	
Analog Input 12[-]	B31	A31	Analog Input 04[-]	
Analog Input 13[+]	B30	A30	Analog Input 05[+]	
Analog Input 13[-]	B29	A29	Analog Input 05[-]	
Analog Ground (for Ai)	B28	A28	Analog Ground (for AI)	
Analog Ground (for AI)	B27	A27	Analog Ground (for AI)	
Analog Input 14[+]	B26	A26	Analog Input 06[+]	
Analog Input 14[-]	B25	A25	Analog Input 06[-]	
Analog Input 15[+]	B24	A24	Analog Input 07[+]	
Analog Input 15[-]	B23	A23	Analog Input 07[-]	
Analog Ground (for AI)	B22	A22	Analog Ground (for AI)	
Analog Ground (for AI)	B21	A21	Analog Ground (for AI)	
Digital Ground	B20	A20	Digital Ground	
N.C.	B19	A19	N.C.	
Digital Output 00	B18	A18	Digital Input 00	
Digital Output 01	B17	A17	Digital Input 01	
Digital Output 02	B16	A16	Digital Input 02	
Digital Output 03	B15	A15	Digital Input 03	
Digital Output 04	B14	A14	Digital Input 04	
Digital Output 05	B13	A13	Digital Input 05	
Digital Output 06	B12	A12	Digital Input 06	
Digital Output 07	B11	A11	Digital Input 07	
AO Control Signal Output 00	B10	A10	AI Control Signal Output 00	
AO Control Signal Output 01	B09	A09	AI Control Signal Output 01	
Digital Ground	B08	A08	Digital Ground	
AO External Sampling Clock Input	B07	A07	AI External Sampling Clock Input	
AO External Stop Trigger Input	B06	A06	AI External Stop Trigger Input	
AO External Start Trigger Input	B05	A05	AI External Start Trigger Input	
Counter UP Clock Input 01	B04	A04	Counter UP Clock Input 00	
Reserved	B03	A03	Reserved	
Counter Gate Control Input 01	B02	A02	Counter Gate Control Input 00	
Counter Output 01	B01	A01	Counter Output 00	



- The numbers in square brackets [ ] are pin numbers designated by HONDA TSUSHIN KOGYO CO.,

Analog Input00 - Analog Input15	Analog input signal. The numbers correspond to channel numbers.
Analog Output00 - Analog Output01	Analog output signal. The numbers correspond to channel numbers.
Analog Ground	Common analog ground for analog I/O signals.
AI External Start Trigger Input	External trigger input for starting analog input sampling.
AI External Stop Trigger Input	External trigger input for stopping analog input sampling.
AI External Sampling Clock Input	External sampling clock input for analog input.
AI Control Signal Output 00	External sampling clock output signal for analog input.
AI Control Signal Output 01	External output signal for analog input status. Not currently connected.
AO External Start Trigger Input	External trigger input for starting analog output sampling.
AO External Stop Trigger Input	External trigger input for stopping analog output sampling.
AO External Sampling Clock Input	External sampling clock input for analog output.
AO Control Signal Output 00	External sampling clock output signal for analog output.
AO Control Signal Output 01	External output signal for analog output status. Not currently connected.
Digital Input00 - Digital Input07	Digital input signal.
Digital Output00 - Digital Output07	Digital output signal.
Counter Gate Control Input00 - Counter Gate Control Input01	Gate control input signal for counter.
Counter Up Clock Input00 - Counter Up Clock Input01	Count-up clock input signal for counter.
Counter Output00 - Counter Output01	Count match output signal for counter.
Digital Ground	Common digital ground for digital I/O signals, external trigger inputs, external sampling clock inputs, and counter I/O signals.
Reserved	Reserved pin
N.C.	No connection to this pin.

Figure 3.2. Pin Assignment of CN1 ***< 2 / 2 >***

⚠ CAUTION

- Do not connect any of the outputs and power outputs to the analog or digital ground. Neither connect outputs to each other. Doing either can result in a fault.
- If analog and digital ground are shorted together, noise on the digital signals may affect the analog signals. Accordingly, analog and digital ground should be separated.
- Leave "Reserved" pins unconnected. Connecting these pins may cause a fault in the board.

Analog Input Signal Connection

The procedure for connecting analog signals depends on whether the analog input signals are single-ended or differential. The sections below describe how to connect the signals using flat cable and shielded cable.

Single-ended Input

The following figure shows an example of flat cable connection.
Connect separate signal and ground wires for each analog input channel on CN1.

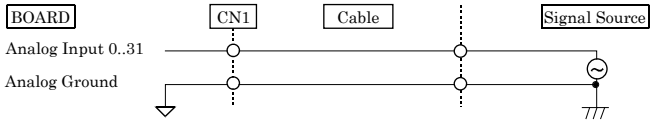


Figure 3.3. Single-ended Input Connection (Flat Cable)

The following figure shows an example of shield cable connection. Use shielded cable if the distance between the signal source and board is long or if you want to provide better protection from noise. For each analog input channel on CN1, connect the core wire to the signal line and connect the shielding to ground.

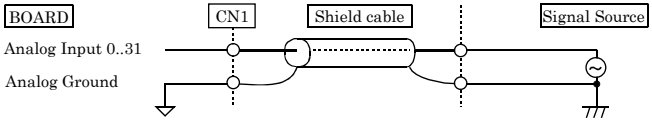


Figure 3.4. Single-ended Input Connection (Shield Cable)

⚠ CAUTION

- If the signal source contains over 1MHz signals, the signal may effect the cross-talk noise between channels.
- If the board and the signal source receive noise or the distance between the board and the signal source is too long, data may not be input properly.
- An input analog signal should not exceed the maximum input voltage (relate to the board analog ground). If it exceeds the maximum voltage, the board may be damaged.
- Connect all the unused analog input channels to analog ground.
- In the channel switching, the multiplexer does the electrical charge and discharge on the internal capacitor according to the signal voltage. Therefore, the voltage from the previous switching state may go into the next channel. It might cause the error of the signal source action. If this occurs, insert a high-speed amplifier as a buffer between the signal source and the analog input pin to reduce the fluctuation.
- An input pin may fail to obtain input data normally when the signal source connected to the pin has high impedance. If this is the case, change the signal source to one with lower output impedance or insert a high-speed amplifier buffer between the signal source and the analog input pin to reduce the effect.

Differential Input

The following figure shows an example of flat cable connection.
For each analog input channel on CN1, connect the "+" input to the signal and connect the "-" input to the signal source ground. Also connect the analog ground on the board to the signal source ground.

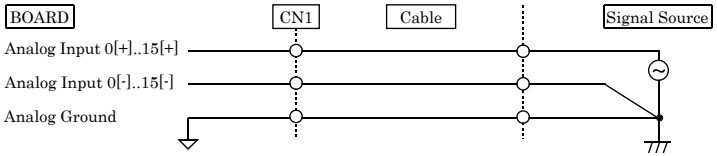


Figure 3.5. Differential Input Connection (Flat Cable)

The following figure shows an example of shielded cable connection. Use shielded cable if the distance between the signal source and board is long or if you want to provide better protection from noise. For each analog input channel on CN1, connect the "+" input to the signal and connect the "-" input to the signal source ground. Also connect the analog ground on the board and the signal source ground to the shielding.

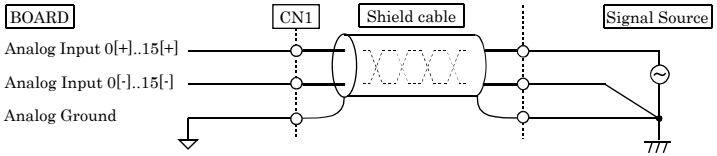


Figure 3.6. Differential Input Connection (Shield Cable)

⚠ CAUTION

- If the signal source contains over 1MHz signals, the signal may effect the cross-talk noise between channels.
- When the analog ground is not connected, the conversion data is not determined.
- If the board and the signal source receive noise or the distance between the board and the signal source is too long, data may not be input properly.
- An input analog signal should not exceed the maximum input voltage (relate to the board analog ground). If it exceeds the maximum voltage, the board may be damaged.
- Connect all the unused analog input channels to analog ground.
- In the channel switching, the multiplexer does the electrical charge and discharge on the internal capacitor according to the signal voltage. Therefore, the voltage from the previous switching state may go into the next channel. It might cause the error of the signal source action. If this occurs, insert a high-speed amplifier as a buffer between the signal source and the analog input pin to reduce the fluctuation.
- An input pin may fail to obtain input data normally when the signal source connected to the pin has high impedance. If this is the case, change the signal source to one with lower output impedance or insert a high-speed amplifier buffer between the signal source and the analog input pin to reduce the effect.

Analog Output Signal Connection

This section shows how to connect the analog output signal by using a flat cable or a shielded cable.

The following figure shows an example of flat cable connection.

Connect the signal source and ground to the CN1 analog output.

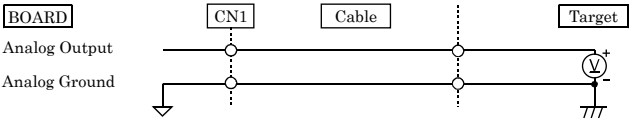


Figure 3.7. Analog Output Connection (Flat Cable)

The following figure shows an example of shielded cable connection. Use shielded cable if the distance between the signal source and board is long or if you want to provide better protection from noise. For the CN1 analog output, connect the core wire to the signal line and connect the shielding to ground.

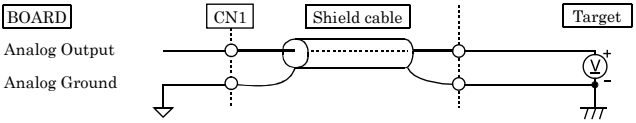


Figure 3.8. Analog Output Connection (Shield Cable)

⚠ CAUTION

- If the board or the connected wire receives noise, or the distance between the board and the target is long, data may not be outputted properly.
- For analog output signal, the current capacity is ±5mA (Max.). Check the specification of the connected device before connecting the board.
- Do not short the analog output signal to analog ground, digital ground, and/or power line. Doing so may damage the board.
- Do not connect an analog output signal to any other analog output, either on the board or on an external device, as this may cause a fault on the board.

Digital I/O signals, Counter signals and Control signals Connection

The following sections show examples of how to connect digital I/O signals, counter I/O signals, and other control I/O signals (external trigger input signals, sampling clock input signals, etc.).

All the digital I/O signals and control signals are TTL level signals.

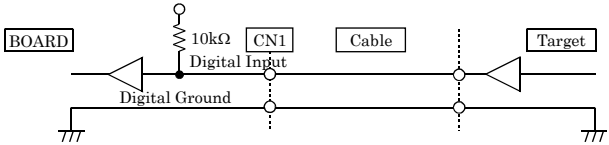


Figure 3.9. Digital Input Connection

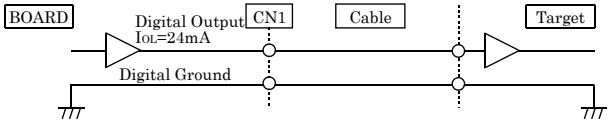


Figure 3.10. Digital Output Connection

About the counter input control signal

Counter Gate Control Input (refer to the chapter 3 Connector Pin Assignment) acts as an input that validate or invalidate the input of an external clock for the counter. This function enables the control of an external clock input for the counter. The external clock for the counter is effective when input is "High", and invalid when input is "Low". If unconnected, it is a pull-up in the board (card) and remains "High". Therefore the external clock for the counter is effective when the counter gate control input is not connected.

⚠ CAUTION

Do not short the output signals to analog ground, digital ground, and/or power line. Doing so may damage the board.

Reference

For the operation timings for control signal input, see “Control Signal Timings” in Chapter 6 “Hardware”.

Synchronization Control Connectors

SC Connectors

Controlling simultaneous operations between boards or controlling in sync with events is in part dependent on software performance. In order to enhance the reliability of the entire system and to solve these problems, the board is equipped with SC (Synchronization Control) connectors.

Connecting the SC connectors allows boards of the same or different models to operate in sync with one another.

From the boards connected with the SC cable, select one master board and use others as slaves. On the master board, set the signal to be supplied to the slave boards with the software. On the slave boards, the signal from the master board can be set to either the pacer clock operation start or stop factor.

All board operations can also be stopped with a stop request from the master in case of an error, for example, or when requested from a slave board. A maximum of 16 boards can be connected including the master.

For more information on the setup procedure, see the driver software online help. When the SC is not connected, use the board with stand-alone settings.

Example 1: When clock start and stop requirements are set the same for multiple boards

In order to synchronize master clock start and stop with slave boards you can build a synchronous system which does not depend on software processing capabilities.

If the board model is the same, data remains synchronized among boards even when channels are expanded. When board models are different, data still remains compatible since operating clock start and stop are dependent on the master.

(1) Connect the SC cable.

(2) Designate master/slave with the software.

(3) Assign to the connectors the clock start and stop signals to be output from the master.

(4) Set up slave boards so they can utilize all signals.

(5) Start in order of slave to master boards.

⚠ CAUTION

- When clock signals are assigned to the synchronization control connector, the maximum clock frequency is restricted to 5MHz.
- When signals are assigned to the synchronization control connector, a delay of approximately 100ns occurs at the slave board.

Example 2: When controlling slave operations with master's internal events

By outputting an internal event (interrupt) occurring on the master board, the slaves can start operating in sync with that signal.

(1) Connect the SC cable.

(2) Designate master/slave with the software.

(3) Assign to the connector the event signal to be output from the master.

(4) Set signals from the master to the start requirements on the slave boards.

(5) Start in order of slave to master boards.

Connecting the SC Connectors (CN2 and CN3)

This board is equipped with sync signal control connectors (CN2 and CN3) for connecting a sync signal cable. When the cable is connected, multiple boards can operate in sync with one another.

Connection Procedure

Connect the sync signal cable when two or more boards need to operate in sync with one another. Connect CN2 with a smaller ID number to CN3 with a greater ID number with the cable. You should only use the cable that came with the board.

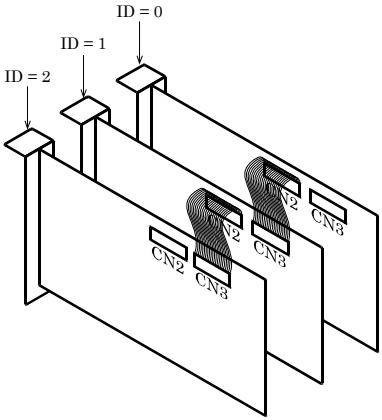


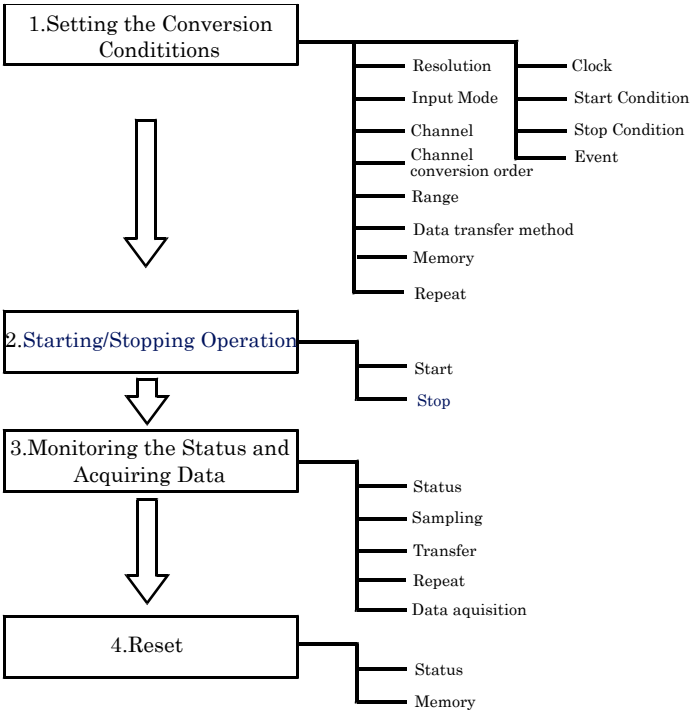
Figure 3.11. Connecting Cables

4. Functions

This chapter describes the different functions that can be implemented using the hardware and driver together. Unless stated otherwise, the driver is assumed to be API-AIO(WDM).

Analog Input Function

The board converts analog signals to digital data according to the resolution and stores it in memory. You can set a variety of conditions for analog input, including the input channel, sampling period, and sampling start/stop conditions.
Analog input processes are classified as follows:



1. Setting the Conversion Conditions

First, set the conditions for executing analog input.

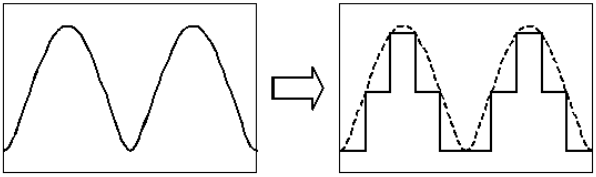
Resolution

"Resolution" signifies the number of bits used by an analog input device to represent analog signals. The higher the resolution, the more finely the voltage range is segmented, allowing the device to convert analog values to digital equivalents more precisely.

A device with a resolution of 12-bit divides the range width into 4096 segments.

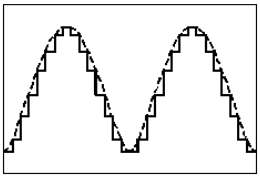
When the device covers the range of 0 - 10V, the minimum unit of converted voltages is 10÷4096 ≈ 2.44mV.

If the device has a resolution of 16-bit, it is 10÷65536 ≈ 0.153mV instead.



Analog Voltage before Conversion

Low Resolution Board



High Resolution Board

ADA16-32/2(PCI)F : The resolution is 16-bit.

Input Mode

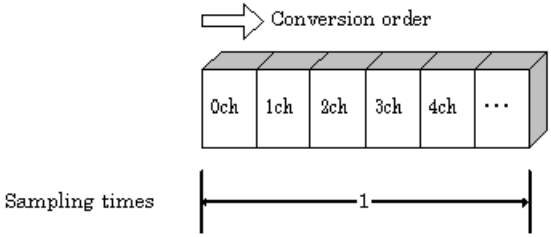
"Input Mode" indicates the method of connecting analog input signals.
The input modes available are single-ended input and differential input.
The single-ended input mode is suitable for the environment in which the potential difference between the signal source and ground and noise components can be ignored. For the environment in which they cannot be ignored, the differential input mode is suitable.
The number of channels available in differential input mode is half that in single-ended input mode.
This board uses on-board jumpers to set the input mode.

Channel

"Channel" represents each point of analog input.
For individual channel numbers, see "Using the On-board Connectors" to "Connector Pin Assignment" in Chapter 3 "External Connection".
You can specify an arbitrary number of points of analog input by setting the channels by means of software.

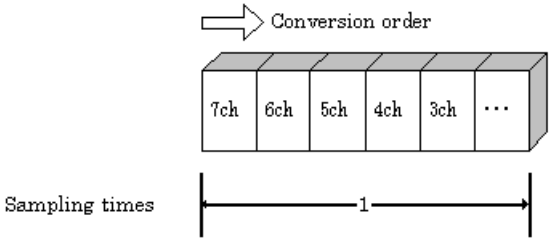
Channel conversion order

Normally, when performing conversion for more than one channel at each sampling, conversion is performed consecutively starting from channel 0.



If you wish, you can specify a different channel conversion order.

If the successive approximation input method is used, input is performed as specified by the channel conversion order setting.



Range

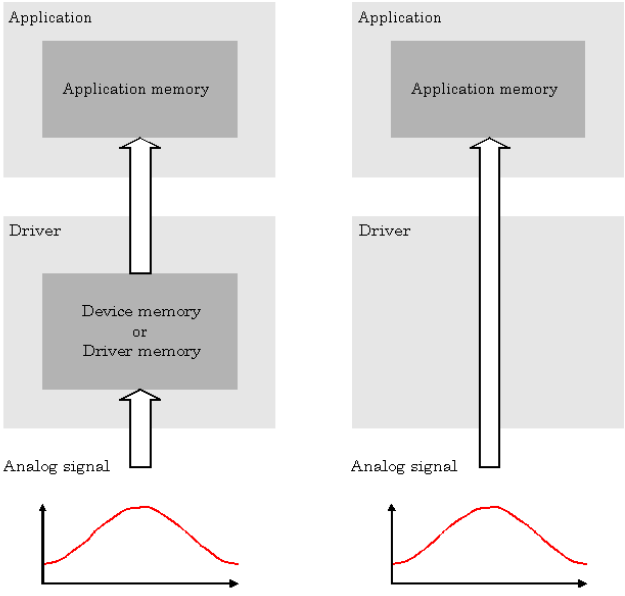
”Range” means the range of voltages at which analog input can be performed.

ADA16-32/2(PCI)F : The input range for the device is set by software.

Data transfer method

You can select between device buffer mode, which saves conversion data in a conversion data memory area either on the device or in the driver, or user buffer mode which saves the conversion data in application memory.

- **Device buffer mode**
  When conversion starts, data is saved in the device buffer (memory on the device itself or in the driver).
  The device buffer can operate as FIFO or ring memory.
  The application calls an API function at an appropriate timing and fetches the conversion data from the device buffer.
  Device buffer mode is simpler to use than user buffer mode as it handles data by sampling count and API functions are provided to retrieve conversion data directly as voltage values.
  As functions are provided to meet most practical requirements, device buffer mode is usually the best option.
- **User buffer mode**
  A region of application memory to store the conversion data is reserved before starting conversion and specified to the driver.
  When conversion starts, conversion data is transferred via the driver directly to the application memory.
  Whether or not to overwrite memory can be specified in user buffer mode.
  User buffer mode uses bus master transfer automatically on devices that support it.
  Select user buffer mode if you wish to use bus master transfer.



Device buffer mode **User buffer mode**

Memory format

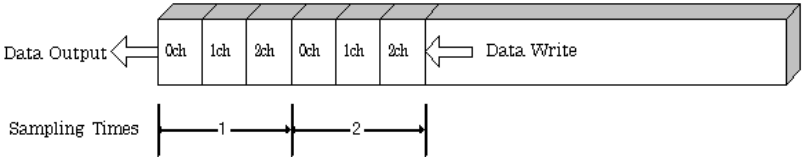
The memory format used by device buffer mode and user buffer mode can be specified by software.

Device buffer mode

- FIFO format

  In the FIFO (First In First Out) format, input data items are read from memory in the same order in which they were written to the memory. Input data items are fed out of the memory sequentially, where the oldest one is always read from the memory. The status monitor and application notification functions are provided, which check and report the state in which the memory has stored a fixed amount of data or in which the memory has become full.

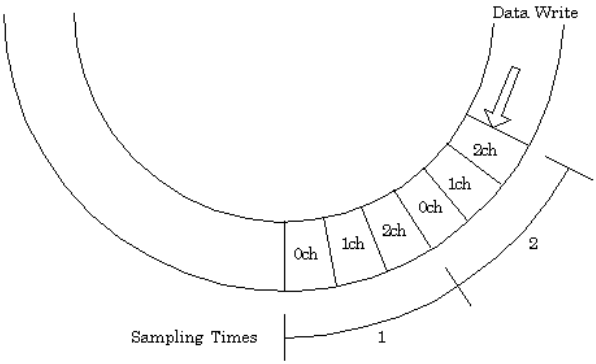
  The FIFO memory is used to obtain all input data from analog input in a short or infinite period of time.



- Ring format

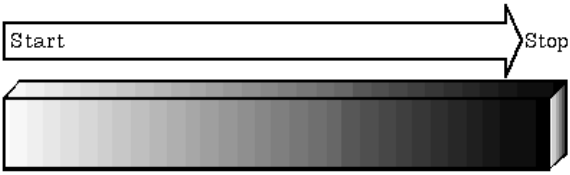
  In the ring format, the memory contains storage areas arranged in a ring. Input data items are written to the memory sequentially. When it stores data exceeding the limit, it overwrites the area storing the previous item of input data. The status monitor and application notification functions are provided, which check and report the state in which data has been written to certain areas of memory.

  The ring memory is used to obtain data where conversion has stopped due to some event, usually without obtaining data in the normal state.

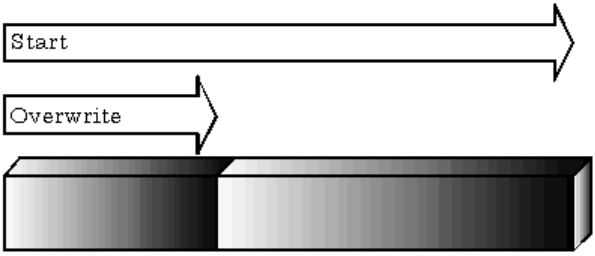


User buffer mode

- No overwriting
  Data transfer halts when conversion data transfer reaches the end of the user buffer.
  This memory format is useful if the number of samples is known in advance.

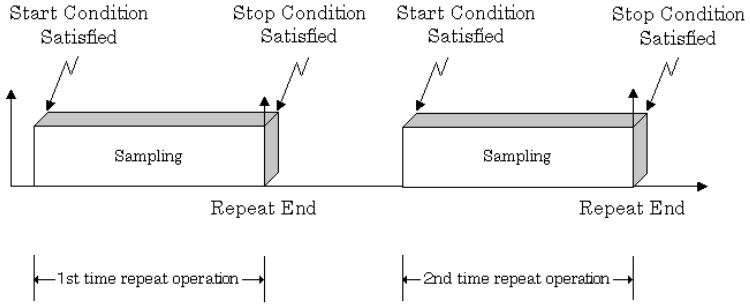


- Permit overwriting
  Data transfer does not halt when conversion data transfer reaches the end of the user buffer.
  Data transfer continues to overwrite memory until the conversion stop condition is established.
  This mode can be used for long term monitoring or for continuous sampling when all data must be captured.



Repeat

"Repeat" indicates the number of repetitions of sampling to be executed, from when the sampling start condition is satisfied until the end of sampling, including delayed sampling.
The number of repetitions is set by means of software, for which conversion is repeated. You can set an infinite number of repetitions, in which case the conversion is terminated by the software abort command.
Input data items are stored to the memory sequentially. The repetition state can be subject to status monitoring and application notification.



Clock

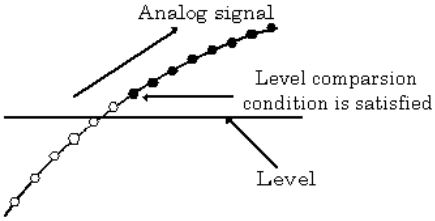
The sampling clock controls the sampling frequency. You can select either the internal sampling clock, external sampling clock, or the output of the event controller.
The sampling clock is selected by means of software.

- Internal sampling clock
  The clock signal from the on-board clock generator is used.
- External sampling clock
  The edge of the digital signal input from an external device is used for the sampling clock.
- Event controller output
  A specified output of the event controller is used as the sampling clock.
  Refer to the explanation of the event controller functions or to the driver help for details about the event controller.

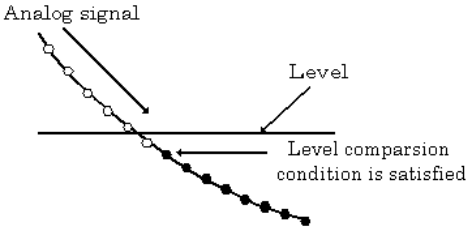
Start Condition

The condition for controlling the start of sampling can be selected from among software, input data comparison, an external trigger and an event controller. The conditions for controlling the start and stop of sampling are completely independent of each other; they can be set separately.

- Software
  The board starts sampling and storing input data to memory immediately after the operation start command is issued.
- Input data comparison
  When the operation start command is issued, the board compares the analog signal input through a specified channel to the value of the preset comparison level. If the analog signal satisfies the condition, the board starts storing input data.
  Level comparison conditions are set as two conditions: level and direction.



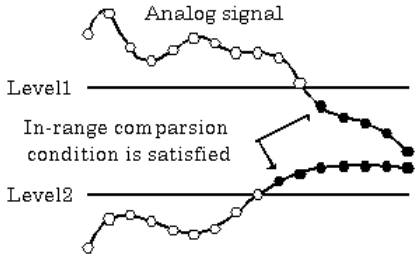
The above sketch shows that the level comparison condition is satisfied in the rising direction. The start condition is satisfied when the analog signal at the specified channel passes the comparison level in the rising direction. Input data items are stored to memory, starting with those at solid dots.



The above sketch shows that the level comparison condition is satisfied in the falling direction. The start condition is satisfied when the analog signal at the specified channel passes the comparison level in the falling direction. Input data items are stored to memory, starting with those at solid dots.

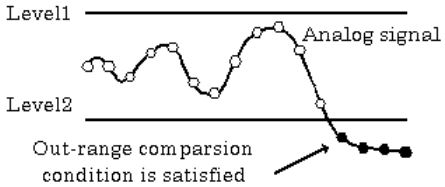
If you set the level comparison directions to both directions, the start condition is satisfied when the analog signal passes the level both in the rising and falling directions.

- Conversion data in-range comparison
  The in-range compare start condition is established when the level on the specified analog channel enters the range specified by level 1 and level 2. Saving of conversion data to memory is performed for the sample points indicated by the black dots.
  If the analog signal already exist within the in-range, the convention starts immediately.



  Start condition : Level 2 ≤ Analog signal ≤ Level 1

- Conversion data out-of-range comparison
  The out-of-range compare start condition is established when the level on the specified analog channel is outside the range specified by level 1 and level 2. Saving of conversion data to memory is performed for the sample points indicated by the black dots.
  If the analog signal already exist within the out-range, the convention starts immediately.



  Start condition : Analog signal ≤ Level 2 or Level 1 ≤ Analog signal

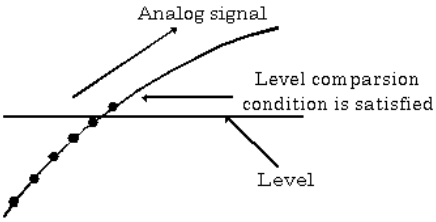
- External trigger
  The board starts waiting for an external control signal as soon as the operation start command is output.
  Sampling and data transfer to memory start when the specified edge (rising edge or falling edge) is input from the external control signal.

- Event controller output
  The board starts waiting for an external control signal as soon as the operation start command is output.
  Sampling and data transfer to memory start when the specified event controller output is received.
  Refer to the explanation of the event controller functions or to the driver help for details about the event controller.

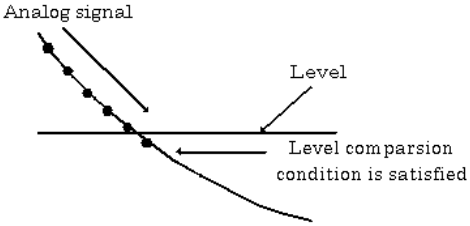
Stop Condition

The condition for controlling the stop of sampling can be selected from among the last sampling count, input data comparison, an external trigger, and software abort.
The board stops sampling whenever an error occurs irrespective of the stop condition setting.

- Last sampling count
  The board stops sampling after storing input data to memory for the specified number of times of sampling.
- Input data comparison
  Once the board has started sampling, it compares the analog signal input through a specified channel to the value of the preset comparison level. If the analog signal satisfies the condition, the board stops sampling.
  Level comparison conditions are set as two conditions: level and direction.

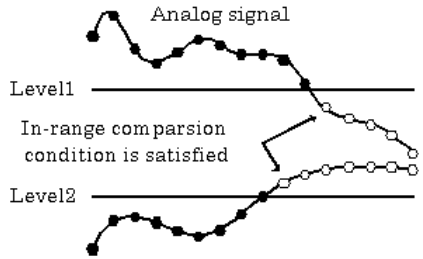


The above sketch shows that the level comparison condition is satisfied in the rising direction.
The stop condition is satisfied when the analog signal at the specified channel passes the comparison level in the rising direction. Input data items are stored to memory, ending until those at solid dots.



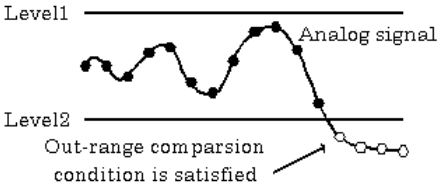
The above sketch shows that the level comparison condition is satisfied in the falling direction.
The stop condition is satisfied when the analog signal at the specified channel passes the comparison level in the falling direction. Input data items are stored to memory, ending until those at solid dots
If you set the level comparison directions to both directions, the start condition is satisfied when the analog signal passes the level both in the rising and falling directions.

- Conversion data in-range comparison
  The in-range compare stop condition is established when the level on the specified analog channel enters the range specified by level 1 and level 2. Saving of conversion data to memory is performed for the sample points indicated by the black dots.
  If the analog signal already exist within the in-range, the convention starts immediately.



  Start condition : Level 2 ≤ Analog signal ≤ Level 1

- Conversion data out-of-range comparison
  The out-of-range compare stop condition is established when the level on the specified analog channel is outside the range specified by level 1 and level 2. Saving of conversion data to memory is performed for the sample points indicated by the black dots.



  Start condition : Analog signal ≤ Level 2 or Level 1 ≤ Analog signal

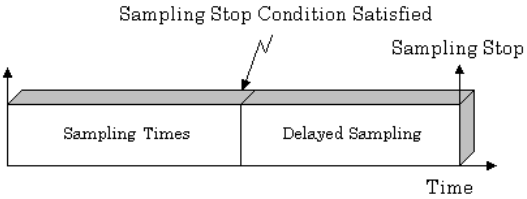
- External trigger
  The board starts waiting for an external control signal after the specified number of samples have been performed.
  Sampling stops when the specified edge (rising edge or falling edge) is input from the external control signal.

- Software
  Sampling continues indefinitely in this mode. Sampling only stops in response to a software command or an error.

- Event controller output
  Sampling stops when the specified event controller output is received.
  Refer to the explanation of the event controller functions or to the driver help for details about the event controller.

Delay

Delayed sampling is performed after the sampling stop condition is satisfied.
When a sampling stop condition other than the software abort command is satisfied, the board performs sampling for the specified number of times of delayed sampling to store input data to memory.
If you set the number of times of delayed sampling to 0, the board stops sampling the moment the sampling stop condition is satisfied.



Event

”Event” works as a function for reporting the occurrence of a certain board state to the application.

The following events can be used in combination depending on the specifications and purpose of the application.

- ”AD conversion start condition satisfied” event
  This event occurs when the AD conversion start condition is satisfied. The event is nullified when the conversion start condition is “software”.
- ”Repeat end” event
  This even occurs whenever a repetition is completed.
- ”End of device operation” event
  This event occurs when the entire operation including repetitions is completed.
- ”Stored specified sampling times” event
  This event occurs when sampling has been performed for the number of times set by software.
  This event can only be used in device buffer mode.
- "Specified number of transfers" event
  This event occurs each time a specified number of samples (set by software) has been completed.
  This event can only be used in user buffer mode.
- Overflow event
  This event occurs at an attempt to store input data with the memory full.
- Sampling clock error event
  This event occurs when conversion stops as an error occurs due to a sampling clock period that is too short.
- AD conversion error event
  This event occurs when conversion stops due to an AD conversion error.

2. Starting/Stopping Operation

Sampling is started by the software command.
Once started, sampling can be stopped by the software command at any timing.

3. Monitoring the Status and Acquiring Data

Software commands are used to monitor the operation status of the device and to acquire input data from memory. Status monitoring and data acquisition can be performed even during sampling.

Status

The current state of the device can be checked by obtaining the device status.
The following types of device status are available:

- Device operating
  The “device operating” status remains ON, after the execution of the sampling start command until the board completes conversion, aborts operation due to an error, or stops sampling in response to the command.
- Waiting for start trigger
  This status remains ON, after the board starts sampling until the start trigger is input, if the conversion start condition is an external trigger or level comparison. The status is set to OFF when the input trigger is input to start conversion.
  The status is set to ON whenever the board enters the conversion start wait status even when repeated operation has been set.
- Specified sampling data stored
  This status is set to ON when input data stored in memory has reached the amount corresponding to the preset number of times of sampling.
  If the memory format is FIFO, the status is set to OFF when the amount of input data in the memory falls below the value corresponding to the preset number of times of sampling as data is acquired.
  Once the status is set to ON when the memory format is ring, it remains ON until it is reset.
- Overflow
  An overflow error occurs when an attempt is made to store input data to memory while it has been full of input data.
  When the memory format is FIFO, the board stops conversion.
  When the memory format is ring, the board continues conversion while overwriting existing data with new one.
- Sampling clock error
  This error occurs when the sampling clock period is too short.
- AD conversion error
  If the “device operating” status remains ON (without terminating conversion) for an extended period of time, the driver regards that state as an operation error and sets this status to ON. This error stops sampling.

Sampling

The number of sampled items of input data stored in memory can be obtained by the software command. This command can only be used in device buffer mode.

Transfer

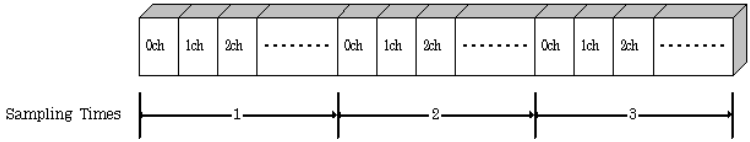
A software command is available to return the number of times data was transferred to user memory. This command can only be used in user buffer mode.

Repeat

The current repeat count can be obtained by the software command.

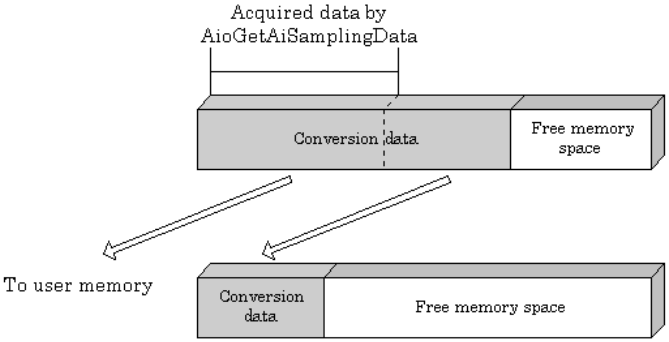
Data acquisition

When using the device buffer, the conversion data stored in memory can be retrieved using a software command.
The figure below shows the correspondence between the sampling count and the conversion channel for the conversion data stored in memory.

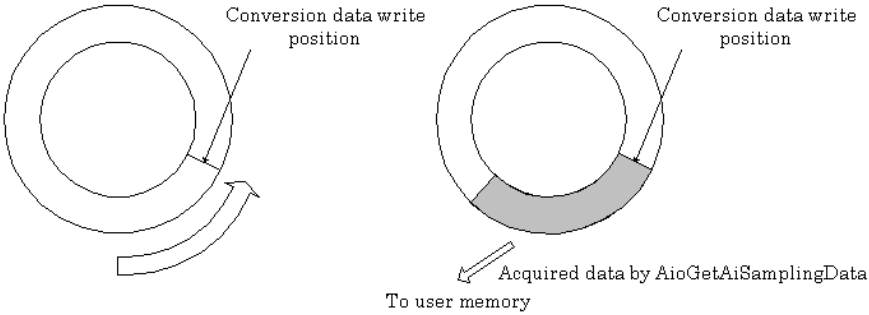


Input data is acquired differently depending on the memory format used.

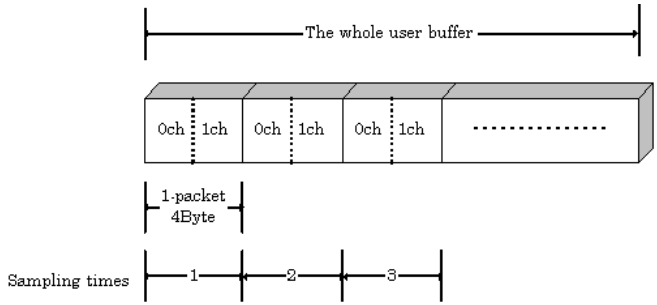
- Data acquisition in FIFO format
  When FIFO memory is used, the oldest data is always read first.
  The following sketch shows an image of data acquisition in FIFO format.
  When data is acquired from the memory, the free memory space increases by that data size. When data is acquired next, the oldest one of the existing data items is taken from the memory in the same way.
  The FIFO memory deletes data once that data is acquired.



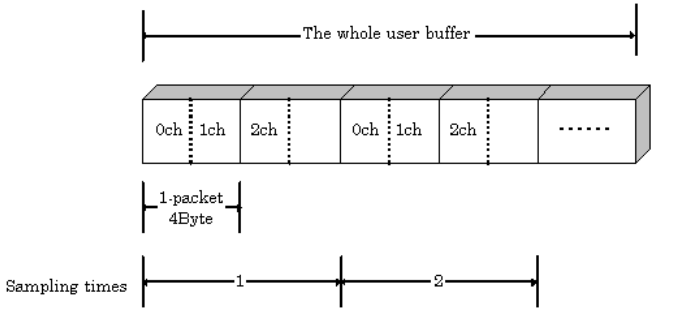
- Data acquisition in ring format
  When ring memory is used, data is read always with respect to the current input data write position.
  The following sketch shows an image of data acquisition in ring format.
  The sampling count obtained is always the number of times of sampling for up to the latest data (shaded portion below).
  The larger the number of samples taken, the older the data item acquired first.
  As the ring memory retains data even after that data is acquired, you can fetch the same data any number of times.



When using a user buffer, the conversion data is transferred directly to application memory by the driver.



The figure above shows how data is transferred to the user buffer for the case when two channels (0ch and 1ch) are used. Data transfer uses a single 4 byte packet which is the minimum data transfer size. Each packet contains two AD conversion data values in binary format.
In the above example, the lower two bytes of the packet contain the channel 0 data and the upper two bytes contain the channel 1 data.
When using two channels, each packet (each data transfer) contains the data for one sampling.



The figure above shows how data is transferred to the user buffer for the case when three channels (0ch, 1ch, and 2ch) are used.
As each packet contains two conversion data values, the uppermost two bytes are not used if an odd number of channels is being used, as in the example above.
When using three channels, two packets (two data transfer operations) are used for each sampling.

Conversion data

The following equation represents the relationship between input data and voltage.

Voltage = Input data x (Max. range value – Min. range value) / Resolution + Min. range value

The value of resolution for the 12-bit device is 4096; that for the 16-bit device is 65536.

The table below shows the relationship between input data and voltage in the ±10-V range.

Voltage	Conversion data (12-bit)
+9.995V	4095
:	:
0.005V	2049
0V	2048
-0.005V	2047
:	:
-10.000V	0

Voltage	Conversion data (16-bit)
+9.99970V	65535
:	:
0.00030V	32769
0V	32768
-0.00030V	32767
:	:
-10.000V	0

Ex.: When input data 3072 is input at a resolution of 12-bit in the ± 10-V range
Voltage = 3072 x (10 - (-10)) ÷ 4096 + (-10)
= 5.0

4.Reset

Various states can be reset by executing the following reset commands:

Status

This command resets the sampling clock error status and AD conversion error status.

Memory

This can only be used when the transfer mode is set to device buffer mode.

This command resets the following memory related states.

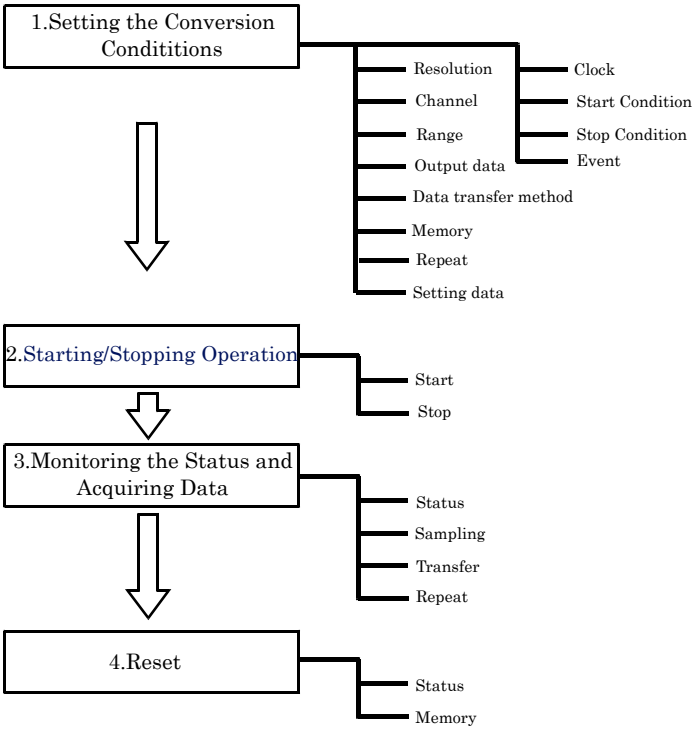
- Resets the conversion data in memory.
- Resets the repeat count to 0.
- Resets the sampling count to 0 when a stop trigger is input.
- Resets the buffer overflow status.
- Resets the status information for the specified data save count.

Analog Output Function

The board converts digital data to analog signals according to the resolution.

You can set a variety of conditions for analog output, including the output channel, sampling period, and sampling start/stop conditions.

Analog output processes are classified as follows:



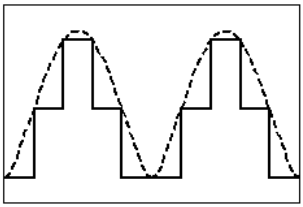
1. Setting the Conversion Conditions

First, set the conditions for executing analog input.

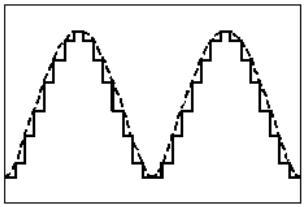
Resolution

"Resolution" signifies the number of bits used by an analog output device to represent analog signals. The higher the resolution, the more finely the voltage range is segmented, allowing the device to convert digital values to analog equivalents more precisely.
A device with a resolution of 12-bit divides the range width into 4096 segments.
When the device covers the range of 0 - 10V, the minimum unit of converted voltages is 10÷4096 ≈ 2.44mV.

If the device has a resolution of 16-bit, it is 10 ÷ 65536 ≈ 0.153mV instead.



Low Resolution Board



High Resolution Board

ADA16-32/2(PCI)F : The resolution is 16-bit.

Channel

"Channel" represents each point of analog output.
For individual channel numbers, see "Using the On-board Connectors" to "Connector Pin Assignment" in Chapter 3 "External Connection".
You can specify an arbitrary number of points of analog output by setting the channels by means of software.

Range

"Range" means the range of voltages at which analog output can be performed.

ADA16-32/2(PCI)F : The output range of the device is set by software.

Output data

Output data = ((Voltage – Min. range value) x Resolution) / (Max. range value – Min. range value)

The value of resolution for the 12-bit device is 4096; that for the 16-bit device is 65536.
The table below shows the relationship between output data and voltage in the ±10-V range.

Voltage	Output data(12-bit)
+9.995V	4095
:	:
0.005V	2049
0V	2048
-0.005V	2047
:	:
-10.000V	0

Voltage	Output data(16-bit)
+9.99970V	65535
:	:
0.00030V	32769
0V	32768
-0.00030V	32767
:	:
-10.000V	0

Ex.: When 3V is output at a resolution of 16-bit in the ±10-V range

Output data = (3 - (-10)) x 65536 ÷ (10 - (-10))
= 42598.4 *

* The value that can be set as output data at this time is an integer. Select "42598" or "42599" as the output data.
The analog signal corresponding to the output data contains an error as follows:
- Output data "42598" converted to: 2.9998 V
- Output data "42598" converted to: 3.0001 V

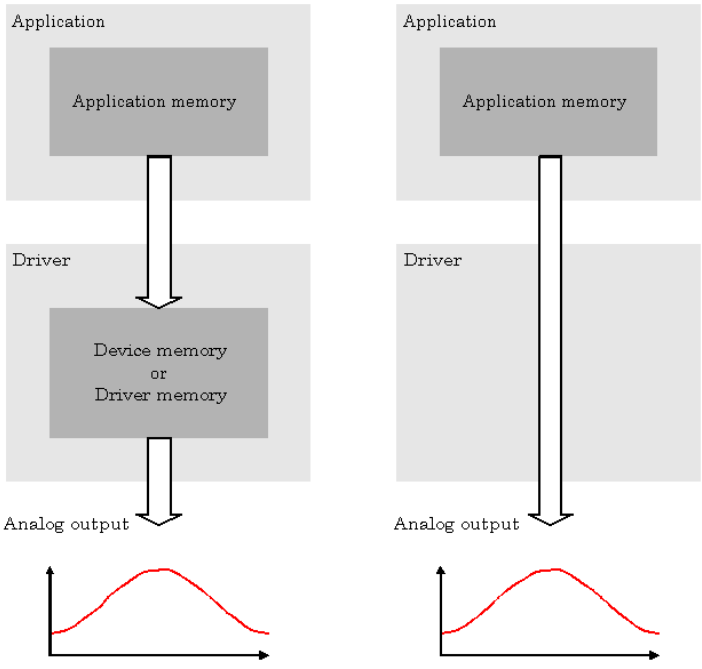
This error is a consequential error occurring when output data is obtained from an expected analog value.

Data transfer method

You can select between device buffer mode, which saves conversion data in a conversion data memory area either on the device or in the driver, or user buffer mode which saves the conversion data in application memory.

- Device buffer mode
  The application output data is first stored in the device buffer (memory on the device itself or in the driver).
  When conversion starts, the device starts outputting the output data.
  The device buffer can operate as FIFO or RING memory.
  Device buffer mode is simpler to use than user buffer mode as it handles conversion data by sampling count and API functions are provided to set conversion data directly as voltage values.
  As functions are provided to meet most practical requirements, device buffer mode is usually the best option.
- User buffer mode
  A region of application memory to store the output data is specified before starting conversion.
  When conversion starts, the driver outputs the conversion data directly to the device.
  Whether or not to overwrite memory can be specified in user buffer mode.

  User buffer mode uses bus master transfer automatically on devices that support it.
  Select user buffer mode if you wish to use bus master transfer.



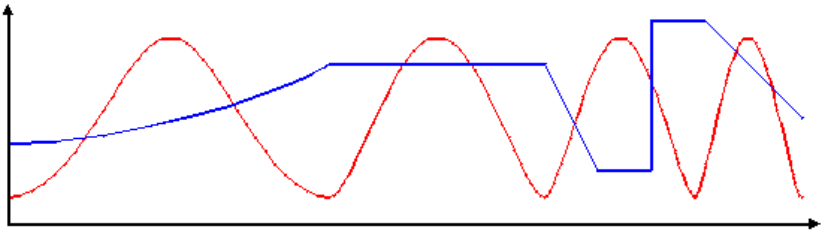
Device buffer mode **User buffer mode**

Memory format

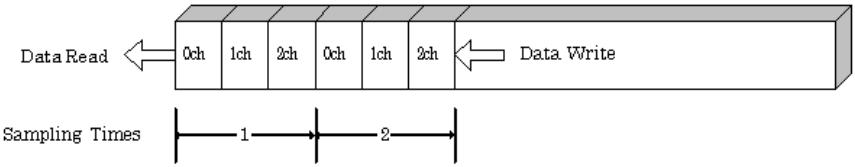
The memory format used by device buffer mode and user buffer mode can be specified by software.

Device buffer mode

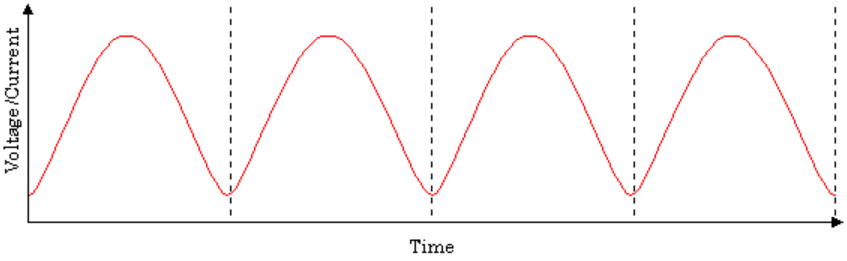
- FIFO format
  Use FIFO format if you wish to output a continuous arbitrary analog output like that shown below.



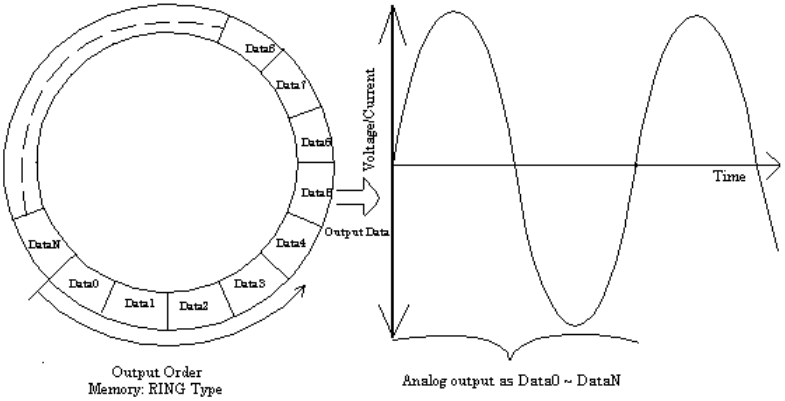
When using FIFO format, writing of conversion data to memory is always performed from after the most recent data and DA conversion is performed on the oldest data in memory.
You can write to memory during analog output operation.
An error occurs if the volume of data exceeds the memory size. However, this error does not stop analog output if it is in progress.



- Ring format
  Use ring format if you wish to output a repeated pattern like that shown below.



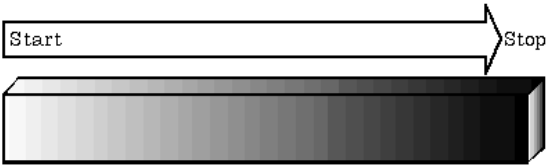
When using ring format, write the data for one complete cycle of the output waveform before starting operation.
You cannot write to the memory during its operation of analog output.
DA conversion data is output continuously in the sequence in which the ring format data is stored.



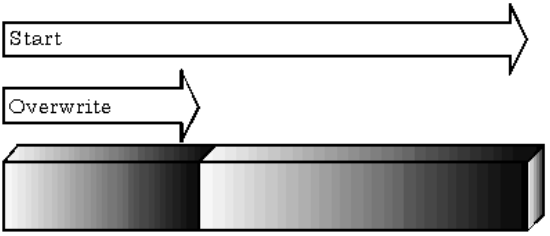
* Although the figure shows a single analog output channel, output from multiple channels is also possible.

User buffer mode

- No overwriting
  Data transfer stops after output reaches the end of the user buffer.
  This memory format is useful if the sampling count is known in advance.

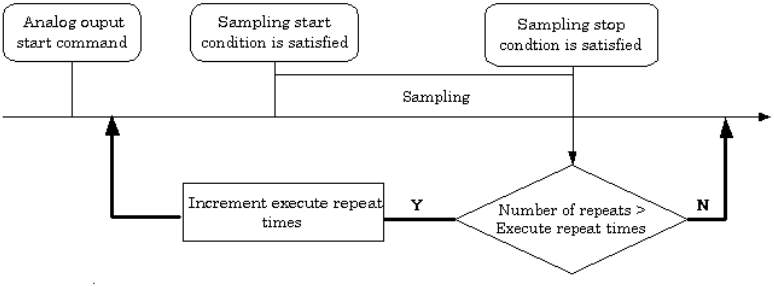


- Permit overwriting
  Data transfer does not stop when output reaches the end of the user buffer.
  Instead, transfer continues repeatedly until the conversion stop condition is established.
  In this mode, the analog output operates like a function generator and continuously generates a fixed waveform.



Repeat

You can specify a repeat count to perform sampling for a specified number of times.

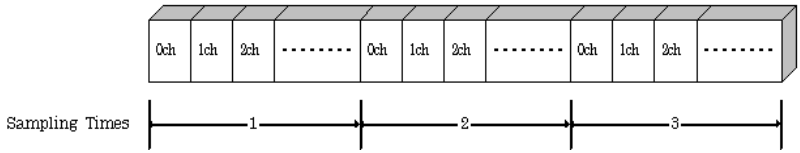


Memory must be set to ring format if a number of repetitions is to be specified.
(The number of repetitions cannot be specified for FIFO memory format.)
The number of repetitions is set by software and sampling is repeated for the specified number of times.
You can also specify that operation continue indefinitely. If set to repeat indefinitely, analog output operation is stopped by outputting a analog output stop command by software.

Setting data

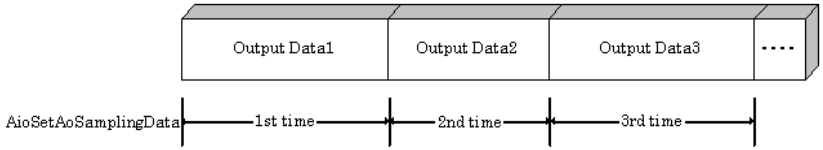
Device buffer mode

When using the device buffer, use a software command to save the data in memory.
The figure below shows the relationship between the conversion data stored in memory and the sampling count and conversion channels.

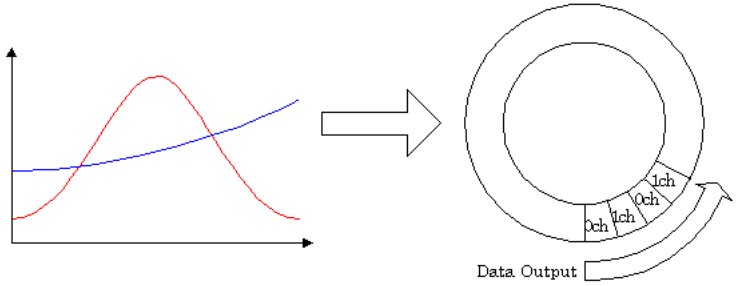


The procedure for setting the conversion data is different depending on the memory format being used.

- Procedure for FIFO format
  When using FIFO format, setting data to memory is always performed from the most recent data. New data can be added during conversion.

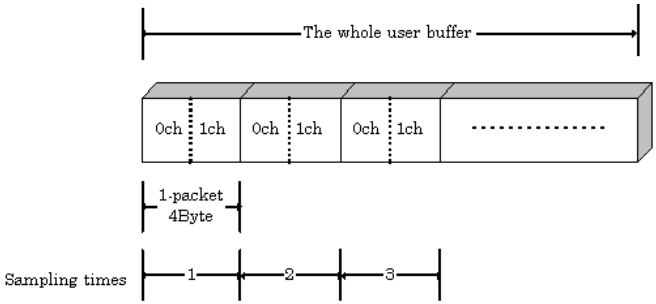


- Procedure for RING format
  When using ring format, a ring memory area large enough for the data to be set is reserved. Data cannot be modified during DA conversion.

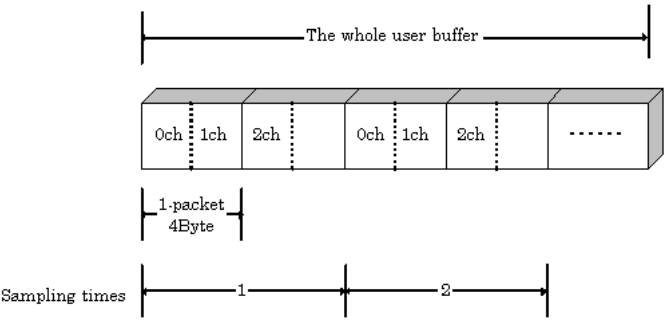


User buffer mode

Store the output data in the user buffer using the format shown below.



The figure above shows how to set the data in the user buffer for the case when two channels (0ch and 1ch) are used.
Data transfer is performed using a single 4 byte packet which is the minimum data transfer size.
Each packet contains two DA conversion data values in binary format.
In the above example, the lower two bytes of the packet contain the channel 0 data and the upper two bytes contain the channel 1 data.
When using two channels, each packet (each data transfer) contains the data for one sampling.



The figure above shows how to set the data in the user buffer for the case when three channels (0ch, 1ch, and 2ch) are used.
As each packet contains two conversion data values, the uppermost two bytes are not used if an odd number of channels is being used, as in the example above.
When using three channels, two packets (two data transfer operations) are used for each sampling.

As the ADA16-32/2(PCI)F supports a maximum of two channels, each packet always contains the data for a single sampling.

Clock

The sampling clock controls the sampling frequency. You can select either the internal sampling clock, or external sampling clock. The sampling clock is selected by means of software.

- Internal sampling clock
  The clock signal from the on-board clock generator is used.
- External sampling clock
  The edge of the digital signal input from an external device is used for the sampling clock.
- Event controller output
  A specified output of the event controller is used as the sampling clock.
  Refer to the explanation of the event controller functions or to the driver help for details about the event controller.

Start Condition

The condition for controlling the start of sampling can be selected from among software, external trigger and an event controller. The conditions for controlling the start and stop of sampling are completely independent of each other; they can be set separately.

- Software
  The board starts sampling and storing input data to memory immediately after the operation start command is issued.
- External trigger
  The board starts waiting for an external control signal as soon as the operation start command is output.
  Sampling and data transfer from memory start when the specified edge (rising edge or falling edge) is input from the external control signal.
- Event controller output
  The board starts waiting for an external control signal as soon as the operation start command is output.
  Sampling and data transfer from memory start when the specified event controller output is received.
  Refer to the explanation of the event controller functions or to the driver help for details about the event controller.

Stop Condition

The condition for controlling the stop of sampling can be selected from among the last sampling count, an external trigger, and software abort.

The board stops sampling whenever an error occurs irrespective of the stop condition setting.

- Last sampling count
  The board stops sampling after storing input data to memory for the specified number of times of sampling.
- External trigger
  The board starts waiting for an external control signal after the specified number of samples have been performed.
  Sampling stops when the specified edge (rising edge or falling edge) is input from the external control signal.

- Software
  Sampling continues indefinitely in this mode. Sampling only stops in response to a software command or an error.
- Event controller output
  Sampling stops when the specified event controller output is received.
  Refer to the explanation of the event controller functions or to the driver help for details about the event controller.

Event

”Event” works as a function for reporting the occurrence of a certain board state to the application.

The following events can be used in combination depending on the specifications and purpose of the application.

- ”DA conversion start condition satisfied” event
  This event occurs when the DA conversion start condition is satisfied. The event is nullified when the conversion start condition is “software”.
- ”Repeat end” event
  This even occurs whenever a repetition is completed.
- ”End of device operation” event
  This event occurs when the entire operation including repetitions is completed.
- "Specified number of output samples complete" event
  This event occurs when the number of output samples specified by software have been completed.
  This event is used when the device buffer mode is used for data transfer.
- Specified number of transfers event
  This event occurs each time a specified number of samples (set by software) has been completed.
  When FIFO format is used, the event occurs when the remaining number of samples falls below a specified level. The event occurs repeatedly as long as this condition is satisfied.
  When ring memory format is used, the event occurs when the number of samples output from memory reaches a specified value. The event occurs for the specified number of repetitions.
  This event is used when the user buffer mode is used for data transfer.
- Sampling clock error event
  This event occurs when conversion stops as an error occurs due to a sampling clock period that is too short.
- DA conversion error event
  This event occurs when conversion stops due to an DA conversion error.

2. Starting/Stopping Operation

Analog output operation is started by a software command (the analog output start command). Similarly, you can stop analog output at any time using a software command (the analog output stop command).

3. Monitoring the Status and Acquiring Data

You can use a software command to check the status of analog output operation and of the output data stored in memory.

Status

The current state of the device can be checked by obtaining the device status.

The following types of device status are available:

- Device operating
  The “device operating” status remains ON, after the execution of the sampling start command until the board completes conversion, aborts operation due to an error, or stops sampling in response to the command.
- Waiting for start trigger
  This status remains ON, after the board starts sampling until the start trigger is input, if the conversion start condition is an external trigger or ean vent controller output. The status is set to OFF when the input trigger is input to start conversion.
  The status is set to ON whenever the board enters the conversion start wait status even when repeated operation has been set.
- Specified number of data outputs
  This status turns ON when the output data set in memory has reached a predefined number of samples.
- Sampling clock error
  This error occurs when the sampling clock period is too short.
- DA conversion error
  If the “device operating” status remains ON (without terminating conversion) for an extended period of time, the driver regards that state as an operation error and sets this status to ON. This error stops sampling.

Sampling

The number of sampled items of output data transferred from in memory can be obtained by the software command.
This command can only be used in device buffer mode.

Transfer

You can use a software command to get the number of times output data has been transferred from user memory.
This command can only be used in user buffer mode.

Repeat

The current repeat count can be obtained by the software command.
This command can only be used in device buffer mode.

4.Reset

Various states can be reset by executing the following reset commands:

Status

This command resets the sampling clock error status and AD conversion error status.

Memory

This can only be used when the transfer mode is set to device buffer mode.

This command resets the following memory related states.

- Resets the conversion data in memory.
- Resets the repeat count to 0.
- Resets the sampling count to 0 when a stop trigger is input.
- Resets the status information for the specified data save count.

Counter Function

1. Setting the Operating Conditions

This specifies the conditions for counter operation.

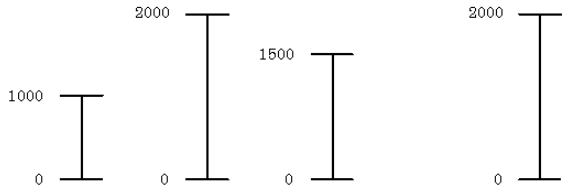
Operating conditions

The basic operation of the counter is to count an external input signal.
The counter includes a function to detect a count match and perform a specified operation when the current count value reaches a preset count value.

Compare count values

The compare count load function automatically loads the next compare count value when a count match occurs.

Time of count compare match	1st	2nd	3rd	. . . .	nth
Cnt[n]: Compare match value	Cnt[0] = 1000	Cnt[1] = 2000	Cnt[2] = 1500	. . . .	Cnt[n-1] = 2000



The figure above shows an example of using the compare count load function.

After the counter starts, the first count match occurs when the count reaches 1000.

When the count reaches 1000, the counter value at which the second compare count match is to occur (2000) is set.

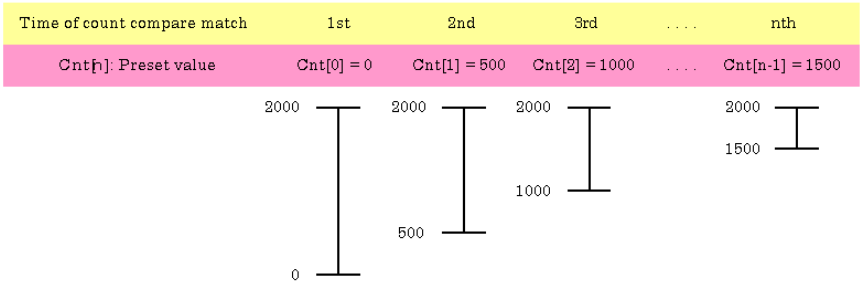
This continues with the next value from the array being set each time a count match occurs.

After the final value from the array is loaded, operation can start again from the beginning of the array.

Alternatively, loading can be halted (in which case, the compare count value remains at 2000).

Preset value

This function loads the next preset value automatically each time a compare count match occurs.



The figure above shows an example of using the preset value load function. In this example, the compare match value is set constant at 2000.

When the counter starts, zero is loaded as the initial preset count.

A compare count match occurs when the count reaches 2000. At this time, 500 is loaded as the next preset value and the count value jumps to 500. This continues with the next value from the array being set each time a count match occurs.

After the final value from the array is loaded, operation can start again from the beginning of the array.

Alternatively, loading can be halted (in which case, the preset value remains at 1500).

Input signal

The external clock or event controller output can be selected as the counter input signal.

Digital filter

A digital filter can be used on external input bits.
The filter time can be set to "don't use", 1μs, 128μs, or 16ms by software.

Event

The event function notifies the application when something occurs on the device.
The following events can be used as required.

- Compare count match event
  This event is triggered when a compare match occurs on the counter.
- Count overrun event
  This event is triggered when a counter overrun occurs.
- Counter operation error
  This event is triggered when a counter operation error causes the counter to stop.

2. Starting/Stopping Operation/Preset

Starting and stopping the counter and setting the preset values are performed using software commands. Once the counter has started, it can be stopped at any time by a software command.
The function for setting the preset values can also be called at any time regardless of whether the counter is running or not.

3. Monitoring the Status and Acquiring Data

Software commands can be used to monitor the device operating status and read counter data. Status monitoring and data acquisition can both be performed while the counter is running.

Status

The current state of the device can be checked by obtaining the device status.
The following types of device status are available:

- Counter operating
  The device operating status is ON from the time the operation start command is executed until operation stops due to a stop command or error.
- Compare count match
  The compare count match status turns ON when a count match occurs after the counter is started. The status is turned OFF by the status reset command.
- Overrun
  The overrun status turns ON if another count match occurs when the compare count match status is already ON. The status is turned OFF by the status reset command.
  Even if the overrun status turns ON, this does not stop the counter.
- Counter operation error
  Execution of driver processing may not be able to keep up if multiple count match events occur within a short time period.
  In this case, the counter operation error status turns ON and counter operation stops.

Data acquisition

The current count value can be read using a software command.

4.Reset

Various states can be reset by executing the following reset commands:

Counter reset

Resets the counter. This restores the counter to its state after power on.

Status

Resets the compare count match status and overrun status.

Digital Input Function

Input bit

Individual digital input points are called input bits.

When the number of input points of a device is 8, the bits are determined as bit 0 - bit 7.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Input in Bits

The state 1 (ON) or 0 (OFF) of each input bit can be obtained by specifying the bit.

Input in Bytes

Individual input bits can be input in byte units.

When the number of input points of the device is 4, the individual input bits are arranged as shown below and the byte data to be input is a value between 0 and 15 depending on the states of the bits.

EX. Input of bit 7 (OFF), bit 6 (ON), bit 5 (OFF), bit 4 (ON), bit 3 (OFF), bit 2 (ON), bit 1 (OFF) and bit 0 (ON)
Byte data = 85(55H)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Digital filter

A digital filter can be used on the input bits.
The filter time can be set to "don't use", 1µs, 128µs, or 16ms by software.

Digital Output Function

Output bit

Individual digital output points are called output bits.

When the number of output points of a device is 8, the bits are determined as bit 0 - bit 7.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Output in Bits

The state of each output bit can be changed to ON or OFF by specifying the bit and setting it to 1 or 0.

Output in Bytes

Individual output bits can be output in byte units.
When the number of output points of the device is 4, the individual output bits are arranged as shown below and byte data to be output is a value between 0 and 15.

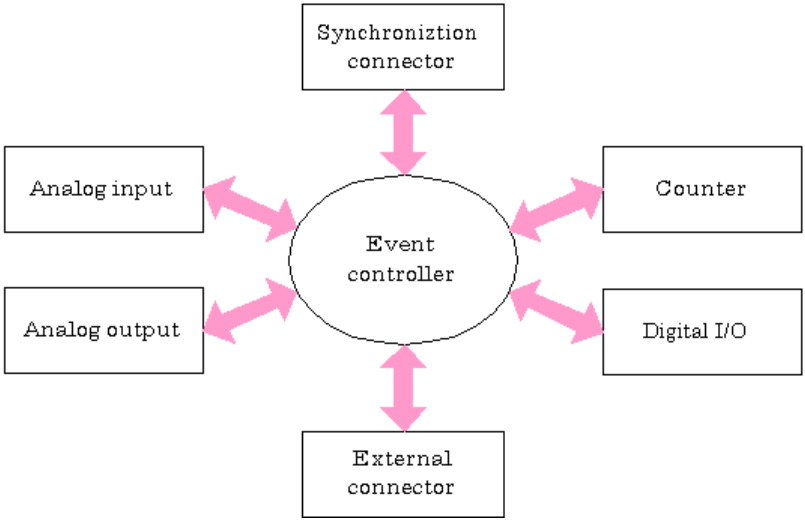
Ex. Output of bit 7 (ON), bit 6(OFF), bit 5 (ON), bit 4 (OFF), bit3 (ON), bit 2 (OFF), bit 1 (ON) and bit 0 (OFF)
Byte data = 170(AAH)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1(ON)	0(OFF)	1(ON)	0(OFF)	1(ON)	0(OFF)	1(ON)	0(OFF)

Event Controller Function

Overview of the event controller

The event controller is used to determine how the control signals between the various functions are interlinked.
Customizing the way in which the control signals are used allows more advanced operations to be implemented by synchronizing the different functions within a single device and by synchronizing operation between multiple devices.

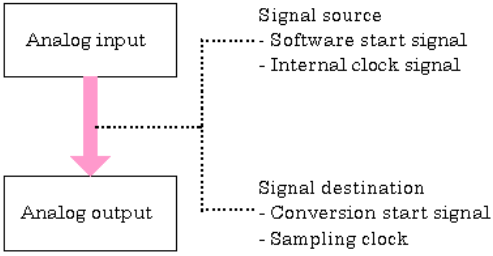


The arrows in the figure show the flow of control signals.

The main control signals include the operation start signals, operation stop signals, and clock signals.

Example of using the event controller (1)

The following example shows how to configure the event controller to perform analog output synchronized with analog input.



First, the conversion start signal for analog input can be configured to be used as the start signal for analog output so that analog input and output start simultaneously.

1. Set the analog input start condition
   This example specifies software activation.
2. Set the analog output start condition
   As the control signal from analog input is used as the start condition, set this as an event controller output.
3. Setting up the event controller
   Setup the event controller.
   The signal destination is the conversion start signal for analog output and the signal source is the analog input software start signal.

⚠ CAUTION

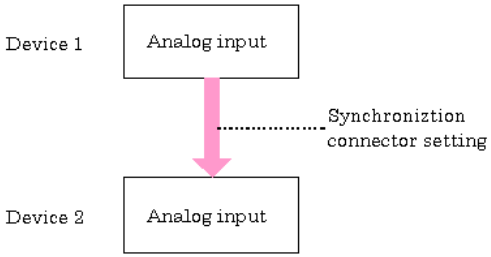
If the start condition for analog input is set to something other than software activation, the signal source for the event controller must also be specified.

Next, to perform both analog input and output with the same interval, configure so that the clock signal for analog input is used as the clock signal for analog output.

4. Set the clock to use for analog input
   This example specifies the internal clock.
5. Set the clock to use for analog output
   As the control signal from analog input will be used as the clock, set this as an event controller output.
6. Setting up the event controller
   Setup the event controller.
   The signal destination is the sampling clock for analog output and the signal source is the internal clock signal for analog input.
   In this example, conversion is started first for the analog output. Actual analog output does not start until analog input conversion starts.
7. Start conversion
   Start conversion.

Example of using the event controller (2)

The following example shows how to configure the event controller to synchronize analog input operation between two devices using the synchronization connector.



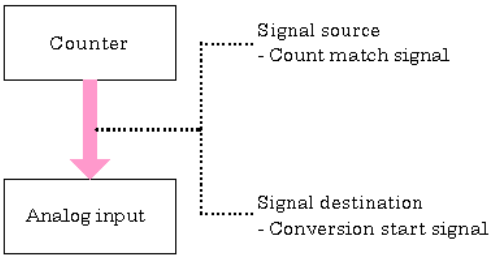
Synchronizing between two devices is performed by connecting control signals from device 1 to device 2.

1. Set the analog input start condition (device 1)
   This example specifies software activation.
2. Setting up the event controller (device 1)
   Setup the event controller to connect the software start signal to the synchronization connector for output to device 2.
   The signal destination is synchronization bus master signal 1 and the signal source is the analog input software start signal.
3. Set the clock to use for analog input (device 1)
   This example specifies the internal clock.
4. Setting up the event controller (device 1)
   Setup the event controller to connect the internal clock to the synchronization connector for output to device 2.
   The signal destination is synchronization bus master signal 2 and the signal source is the internal clock signal for analog input.
   Device 2 performs analog input in accordance with the control signals received from device 1.
5. Set the analog input start condition (device 2)
   As the control signal from device 1 is used as the start condition, set this as an output from the event controller.
6. Setting up the event controller (device 2)
   The software start signal from device 1 is connected to synchronization bus master signal 1 which in turn is connected to synchronization bus slave signal 1 in the synchronization connector. Accordingly, setup the event controller to use synchronization bus slave signal 1 as the conversion start signal for device 2.
   The signal destination is the conversion start signal for analog input and the signal source is synchronization bus slave signal 1.
7. Set the clock to use for analog input (device 2)
   As the control signal from device 1 is used as the clock, set this as an output from the event controller.

8. Setting up the event controller (device 2)
   The internal clock signal from device 1 is connected to synchronization bus master signal 2 which in turn is connected to synchronization bus slave signal 2 in the synchronization connector. Accordingly, setup the event controller to use synchronization bus slave signal 2 as the sampling clock for device 2.
   The signal destination is the sampling clock for analog input and the signal source is synchronization bus slave signal 2.
9. Start conversion
   In this example, conversion is started first on device 2. When analog input starts on device 1, analog input starts simultaneously on device 2.

Example of using the event controller (3)

The following example shows how to configure the event controller to start analog input when a count match occurs on the counter.

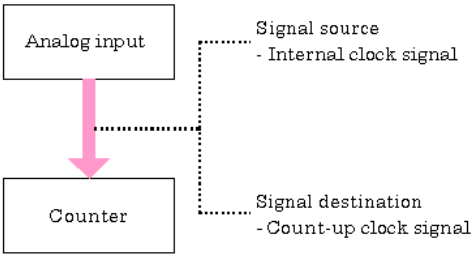


Connect the count match signal from the counter to the conversion start signal for analog input.

1. Set the analog input start condition
   As the control signal from the counter is used as the start condition, set this as an output from the event controller.
2. Setting up the event controller
   Setup the event controller.
   The signal destination is the conversion start signal for analog input and the signal source is the count match signal from the counter.
3. Start operation
   In this example, conversion is first started for analog input. Actual conversion does not start until a count match occurs on the counter.

Example of using the event controller (4)

The following example shows how to count the analog output clock and output the counter count match signal externally each time a specified number of clocks have been input.



This example does not require any setup of the event controller for analog output.
Event controller setup is only needed for the counter.

1. Set the counter input signal
   As this example uses the analog input clock signal, setup the event controller accordingly.
2. Setting up the event controller
   Setup the event controller.
   The signal destination is the count-up clock signal and the signal source is the internal clock signal for analog output.
3. Start conversion
   Start the counter first, then start analog output.

5. About Software

Bundled disk Directory Structure

```
\
├─ Autorun.exe          Installer Main Window
|  Readmej.html         Version information on each API-TOOL (Japanese)
|  Readmeu.html         Version information on each API-TOOL (English)
  .
  .
├──APIPAC               Each installer
|      ├─AIO
|      |    ├─DISK1
|      |    ├─DISK2
|      |    ├─……
|      |    ├─DISKN
|      ├─AioWdm
|      ├─CNT
|      ├─DIO
|      ├─……
  .
  .
|─HELP                  HELP file
|      ├─Aio
|      ├─Cnt
|      ├─……
  .
  .
|─INF                   Each INF file for OS
|      ├─WDM
|      ├─Win2000
|      ├─Win95
  .
  .
├─linux                 Linux driver file
|      ├─cnt
|      ├─dio
|      ├─……
  .
  .
|─Readme                Readme file for each driver
  .
  .
|─Release               Driver file on each API-TOOL
|      ├─API_NT         (For creation of a user-specific install program)
|      ├─API_W95
  .
  .
|─UsersGuide            Hardware User's Guide(PDF files)
```

About Software for Windows

The bundled disk “Driver library API-PAC(W32)” contains the functions that provide the following features:

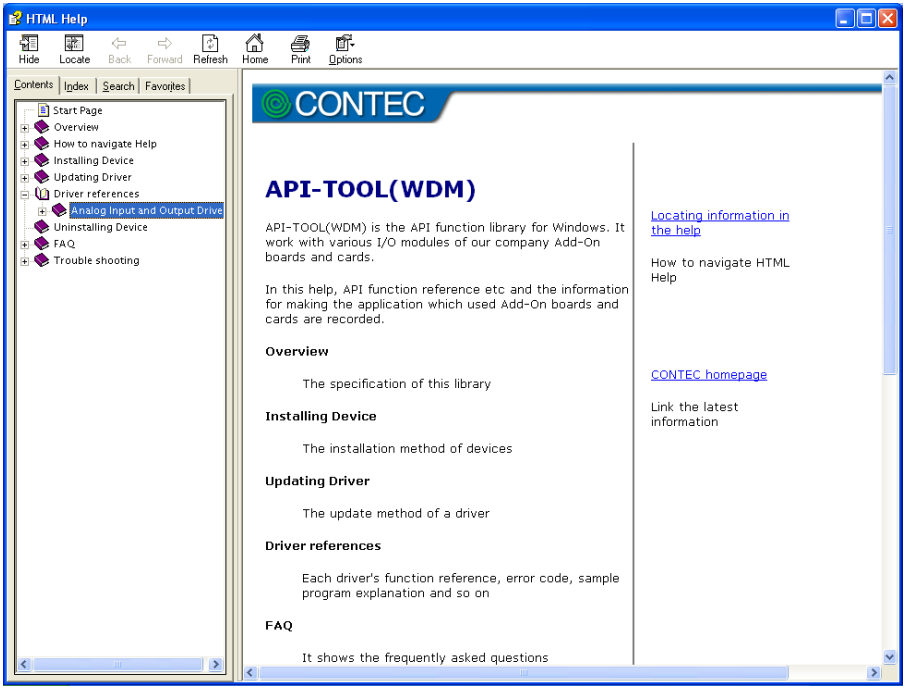
- Analog input or output through arbitrary channels
- Analog input at arbitrary intervals using the internal or external sampling clock
- Simultaneous monitoring of the termination of analog input sampling, buffer memory usage, and interrupt events such as occurrences of errors
- Driver option check using a demo driver even without the board installed

For details, refer to the help file. The help file provides various items of information such as “Function Reference”, “Sample Programs”, “Tutorial”, “FAQs”and “Troubleshooting”. Use them for program development and troubleshooting.

Accessing the Help File

(1) Click on the [Start] button on the Windows taskbar.

(2) From the Start Menu, select “Programs” – “CONTEC API-PAC(W32)” – “AIOWDM” – “API-AIO(WDM) HELP” to display help information.

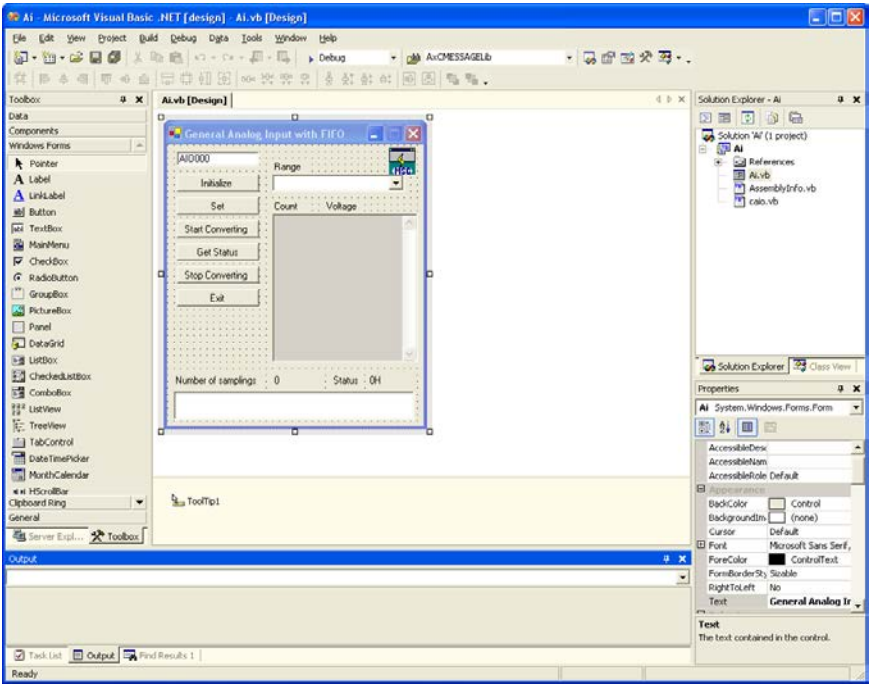


Using Sample Programs

Sample programs are provided for each of the basic operations. You can use these to check the operation of the board and as a reference when writing your own programs.

To use the sample programs, specify the device name in the property page for the program.

The sample programs are stored in \Program Files\CONTEC\API-PAC(W32)\AIOWDM\Samples.



Running a Sample Program

(1) Click on the [Start] button on the Windows taskbar.

(2) From the Start Menu, select “Programs” – “CONTEC API-PAC(W32)” – “AIOWDM” – “SAMPLE…”.

(3) A sample program is invoked.

Sample Programs - Examples

Analog input

Simple sample program

- SingleAi	Perform single analog input from specified channel
- MultiAi	Perform single analog input from multiple channels

Device buffer

- Ai	Perform standard analog input using a FIFO buffer
- AiPoll	Perform standard analog input by polling
- AiEx	Perform analog input for multiple channels using a FIFO buffer
- AiLong	Perform long-duration analog input using a FIFO buffer
- AiExt	Perform analog input using an external clock
- AiTrg	Perform analog input using an external trigger to start and stop operation
- AiLevel1	Use a level trigger to start analog input
- AiLevel2	Use a level trigger to stop analog input
- Ai2	Perform standard analog input using more than one device
- AiCall	Perform analog input using a callback routine
- AiSync1	Perform multi-channel analog input using two devices

User buffer

- AiUser1	Perform analog input for a specified duration using a user buffer
- AiUser2	Perform analog input indefinitely using a user buffer
- AiData	Perform analog input with attached data

Analog output

Simple sample program

- SingleAo	Perform single analog output from specified channel
- MultiAo	Perform single analog output from multiple channels

Device buffer

- Ao	Perform standard analog output using a FIFO buffer
- AoPoll	Perform standard analog output by polling
- AoEx	Perform analog output for multiple channels using a FIFO buffer
- AoLong	Perform long-duration analog output using a FIFO buffer
- AoExt	Perform analog output using an external clock
- AoRing	Perform continuous analog output using a ring buffer
- AoTrg	Perform analog output using an external trigger to start and stop operation
- Ao2	Perform standard analog output using more than one device
- AoCall	Perform analog output using a callback routine

User buffer

- AoUser1	Perform analog output for a specified duration using a user buffer
- AoUser2	Perform analog output indefinitely using a user buffer

Digital input/output

- DioBit	Perform digital I/O using bit values
- DioByte	Perform digital I/O using byte values

Counter/Timer

- Counter	General purpose counter
- Interval	Interval timer
- Watch	Stopwatch timer

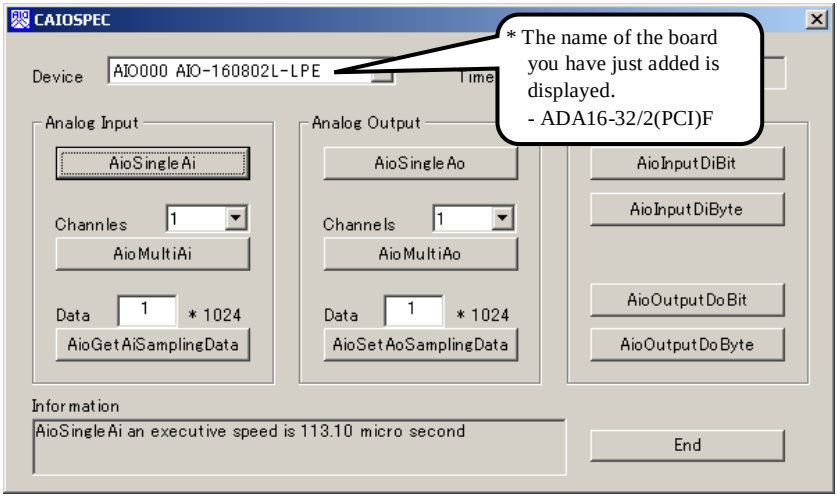
Others

- Convert	Data conversion
- Multi1	Synchronized analog I/O

Usage of Utility Program

Program for Measurement of Function Execution Speed

The execution time of some main functions can be measured in a function execution speed measurement program. To use a function execution speed measurement program, click the [execution time measurement] button in the diagnostic program.



Procedure

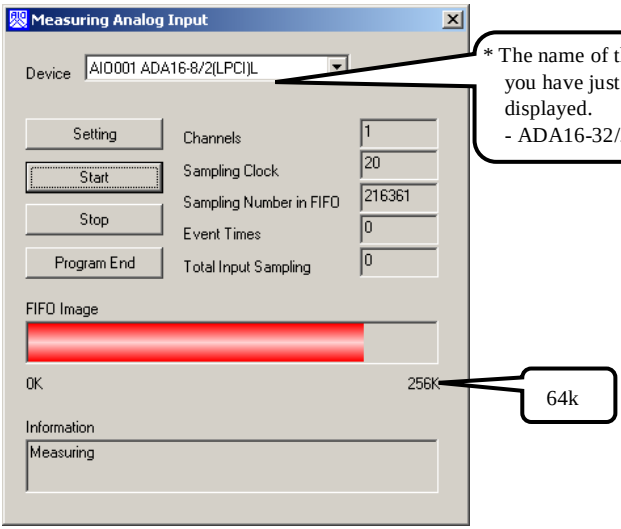
(1) Chose the measure device from device list.

(2) Click the button written with the function name to measure the execution speed of the function. Please choose from a list the number of channels used for conversion in function AioMultiAi and AioMultiAo. Input the transmission data size in function AioGetAiSamplingData and AioSetAoSamplingData. The transmission data is set by unit of kByte.

(3) End the application with an [end] button.

Analog Input Measurement Tool

It is an analog input measurement utility to carry out infinity sample in the FIFO memory. Once the conversion data of memory accumulates to a certain quantity, the event occurs and data of the memory is acquired. Data in the FIFO memory can be confirmed visually.

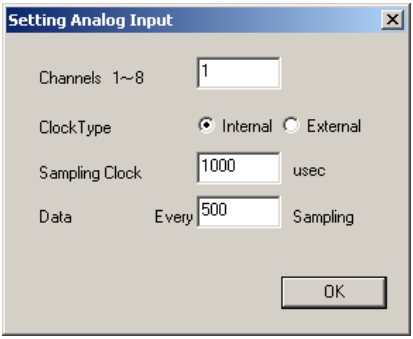


The number of channels used, the internal/external clock, the conversion speed, and the sampling frequency at which an event generates can be set. Since the notification of a sampling clock error event is sent, please make use of it for the conversion spec measurement under various conversion conditions.

Procedure

(1) Chose the device name of the device to be used from the upper left combo box, and click the setting button.

(2) The conversion conditions are set on the screen of the analog input setting.
Once an input is done at the sampling frequency specified as data taking-in sampling, an event occurs and data will be acquired. Click the OK button to finish setting the conditions, and returns to former screen.



(3) Start the measurement with measurement start button. The various states during the conversion are displayed.

The number of the samplings in FIFO is :
It is conversion data taken in the memory. This can be visually checked in a "memory image".

Event generation sampling frequency :
When the number of input sampling in FIFO reaches this frequency, the event generates.

Total input sampling frequency :
It is the total number of samplings for application in the memory.

Measurement may stop by the following errors.

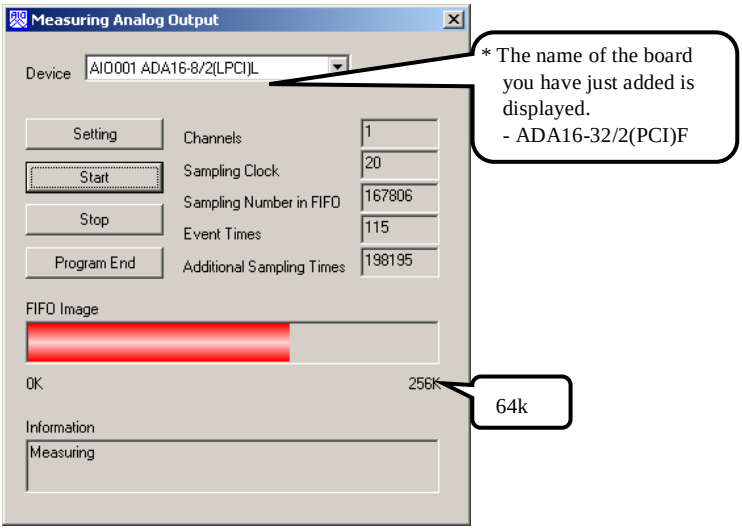
Sampling clock error :
It means that the conversion speed is too fast and the driver processing is not in time when converting at the internal clock.
The cycle of the clock is too fast when converting it at the external clock. Moreover, the cause by noise etc. is also concerned.

Buffer overflow :
The memory overflows since the conversion speed is too fast compared with the one at which data is inputted.

(4) Click the “stop” button, and measurement stops.

Analog Output Measurement Tool

It is an analog output measurement utility to carry out infinity sample in the FIFO memory. Once the conversion data of memory accumulates to a certain quantity, the event occurs and new output data is added. Data in the FIFO memory can be confirmed visually.



The number of channels used, the internal/external clock, the conversion speed, the sampling frequency at which an event generates and number of the samplings to be added can be set. Since the notification of a sampling clock error event is sent, please make use of it for the conversion spec measurement under various conversion conditions.

⚠ CAUTION

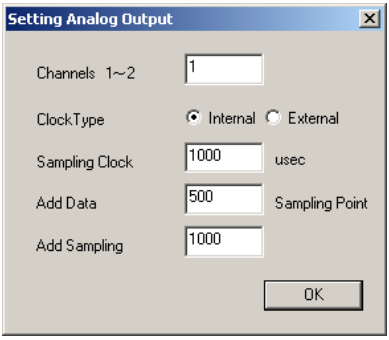
This program is made with Visual Basic. Therefore, it is not possible to execute it as it is in the environment in which Visual Basic is not installed. The program can be used by executing the following setup below the folder that installs the API-AIO(WDM).

AIOWDM\Utility\AoSpec\setup.exe

Procedure

(1) Chose the device name of the device to be used from the upper left combo box, and click the setting button.

(2) The conversion conditions are set on the screen of the setting analog output.



Once an output is done at the sampling frequency specified as data setting sampling frequency, an event occurs and data will be added. Click the OK button to finish setting the conditions, and returns to former screen.

(3) Start the measurement with measurement start button. The various states during the conversion are displayed.

The number of the samplings in FIFO is :
It is conversion data set in the memory. This can be visually checked in a "memory image".

Event generation sampling frequency :
When the number of unoutput sampling in FIFO reaches this frequency, the event generates.

Adding sampling frequency :
It is the number of samplings of output data to be added during the event.

Measurement may stop by the following errors.

Sampling clock error :
It means that the conversion speed is too fast and the driver processing is not in time when converting at the internal clock.
The cycle of the clock is too fast when converting it at the external clock. Moreover, the cause by noise etc. is also concerned.

(4) Click the “stop” button, and measurement stops.

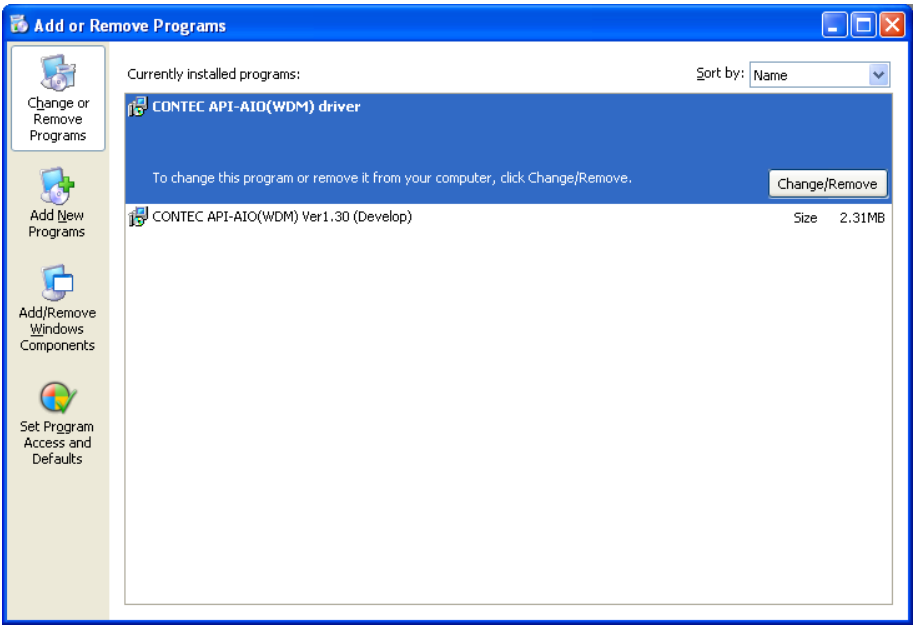
Uninstalling the Driver Libraries

To uninstall API-PAC(W32), follow the procedure below.

(1) Click on the [Start] button on the Windows taskbar. From the Start Menu, select “Settings” – “Control Panel”.

(2) Double-click on “Add/Remove Programs” in the Control Panel.

(3) Select "CONTEC API-AIO(WDM) driver" and "CONTEC API-AIO(WDM) VerX.XX (Development environment)" from the list of applications.
Click the [Change/Remove] button. Follow the on-screen instructions to uninstall the function libraries.



About Software for Linux

The Linux version of analog I/O function driver, API-AIO(LNX), provides functions that execute the following features:

- The analog input/output of a specified channel can be done.
- It is possible to operate as a set parameter to the analog input/output board is preserved by the default value, and the setting of the parameter doesn't exist.

For details, refer to the help file. The help file provides various items of information such as “Function Reference”, “Sample Programs”, and “FAQs”. Use them for program development and troubleshooting.

Driver Software Install Procedure

The Linux version for analog I/O driver, API-AIO(LNX), is supplied as a compressed file /linux/aio/caioXXX.tgz on the bundled disk. (Note: XXX represents the driver version.)

Mount the bundled disk as shown below, copy the file to an arbitrary directory, and decompress the file to install the driver.

For details on using the driver, refer to readme.txt and the help file in HTML format extracted by installation.

To install the driver, log in as a superuser.

Decompression and setup procedure

```
# cd
 # mount /dev/cdrom /mnt/cdrom                Mount the bundled disk.
 # cp /mnt/cdrom/linux/aio/caioXXX.tgz ./      Copy the compressed file.
 # tar xvfz caioXXX.tgz                       Decompress the compressed file.
 ................
 # cd contec/caio
 # make
                                              Compile the file.
 ................
 # make install                               Install.
 ................
 # cd config
 # ./config                                   Set up the board to be used.
 ..... Set as follows.........
 # ./contec_aio_start.sh                      Start the driver.
# cd
```

Accessing the Help File

(1) Invoke a web browser in your X-Window environment.

(2) In the browser, open diohelp.htm in the contec/caio/help directory.

Using Sample Programs

Sample programs have been prepared for specific basic applications.
Sample programs for each language are contained in the contec/caio/samples directory. For compiling them, refer to the manual for the desired language.

Uninstalling the driver

To uninstall the driver, use the uninstall shell script contained in the contec/caio directory. For details, check the contents of the script.

6. About Hardware

This chapter provides hardware specifications and hardware-related supplementary information.

For detailed technical information

For further detailed technical information (“Technical Reference” including the information such as an I/O map, configuration register, etc.), visit the Contec's web site (http://www.contec.com/support/) to call for it.

Hardware specification

Table 6.1. Specification (1/2)

Item		Specification
Analog input		
	Isolated specification	Un-Isolated
	Input type	Single-Ended Input or Differential Input
	Number of input channels	32ch (Single-Ended Input) 16ch (Differential Input)
	Input range	Bipolar ±10V, ±5V, ±2.5V or Unipolar 0 - +10V, 0 - +5V, 0 - +2.5V
	Absolute max. input voltage	±15V
	Input impedance	1MΩ or more
	Resolution	16bit
	Non-Linearity error *1	±5LSB
	Conversion speed	2μsec/ch (Max.)
	Buffer memory	64k Word FIFO or 64k Word RING
	Conversion start trigger	Software, conversion data compare, external trigger, and event controller output.
	Conversion stop trigger	Settings include data save complete, conversion data compare, external trigger, event controller output, and software.
	External start signal	TTL (Rising or falling edge can be selected by software)
	External stop signal	TTL (Rising or falling edge can be selected by software)
	External clock signal	TTL (Rising or falling edge can be selected by software)
	External status output signal	2 TTL Sampling clock output
Analog output		
	Isolated specification	Un-Isolated
	Number of output channels	2ch
	Output range	Bipolar ±10V, ±5V, ±2.5V, ±1.25V or Unipolar 0 - +10V, 0 - +5V, 0 - +2.5V
	Output current ability	±5mA
	Output impedance	1Ω or less
	Resolution	16bit
	Non-Linearity error *1	±3LSB
	Conversion speed	10μsec (Max.)
	Buffer memory	64k Word FIFO or 64k Word RING
	Conversion start trigger	Software, external trigger, and event controller output.
	Conversion stop trigger	Settings include data save complete, external trigger, event controller output, and software.
	External start signal	TTL (Rising or falling edge can be selected by software)
	External stop signal	TTL (Rising or falling edge can be selected by software)
	External clock signal	TTL (Rising or falling edge can be selected by software)
	External status output signal	2 TTL Sampling clock output
Digital I/O		
	Number of input channels	Un-Isolated input 8ch (TTL positive logic)
	Number of output channels	Un-Isolated output 8ch (TTL positive logic)

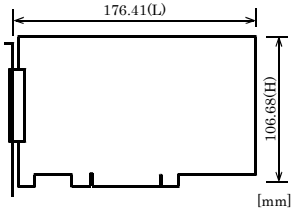
Table 6.1. Specification (2/2)

Item		Specification
Counter		
	Number of channels	2ch
	Counting system	Up count
	Max. count	FFFFFFFFh (Binary data,32bit)
	Number of external inputs	2 TTL (Gate/Up)/ch Gate (High level), Up (Rising edge)
	Number of external outputs	TTL 1 output/ch Count match output (positive logic, pulse output)
Bus master section		
	DMA channels	2ch (one each for input and output)
	Transfer bus width	32bit
	Transfer data length	8 PCI Words length (Max.)
	FIFO	1K-Word/ch
	Scatter/Gather function	64M-Byte/ch
Synchronization bus section		
	Control output signal	Selection of output signal with the software when specifying a sync master board.
	Control input signal	Selection of sync factor with the software when specifying sync slave boards.
	Max. board count for connection	16 boards including the master board
	Connector	PS-10PE-D4T1-B1 (JAE) or equivalent x 2
Common section		
	I/O address	64 ports x 1,256 ports x 1 region
	Interruption level	Errors and various factors, One interrupt request line as INTA
	Connector	96-pin half pitch connector [M(male)type] PCR-96LMD [HONDA TSUSHIN KOGYO CO., LTD.] or equivalent
	Power consumption	5VDC 1100mA (Max.)
	Operating condition	0 - 50°C, 10 - 90%RH (No condensation)
	PCI bus specification	32bit, 33MHz, Universal key shapes supported *2
	Dimension (mm)	176.41(L) x 105.68(H)
	Weight	130g

*1: The non-linearity error means an error of approximately 0.1% occurs over the maximum range at 0°C and 50°C ambient temperature.

*2: This board requires +5V power supply from expansion slots (it does not operate in the environment of only +3.3V power supply).

Board Dimensions



The standard outside dimension (L) is the distance from the end of the board to the outer surface of the slot cover.

Block Diagram

Figure 6.1 is a circuit block diagram of this board.

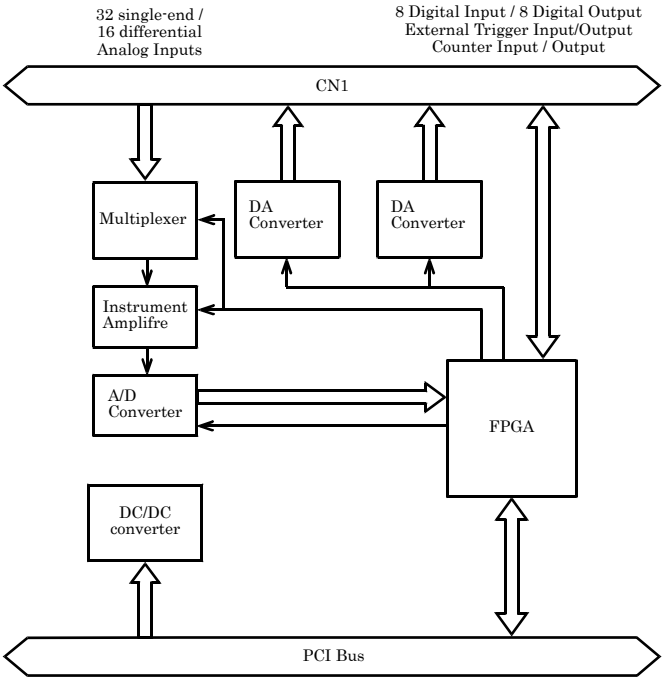


Figure 6.1. Block Diagram

Control Signal Timings

Control Signal Timings for Analog Input

Figures 6.2, 6.3, 6.4, and Table 6.2 show the control signal timings for the analog input function.

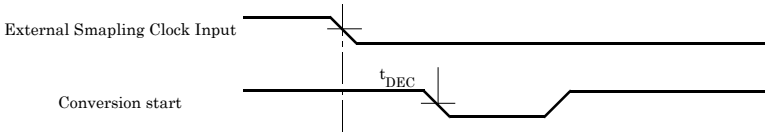


Figure 6.2. Timing Chart of External Sampling Clock

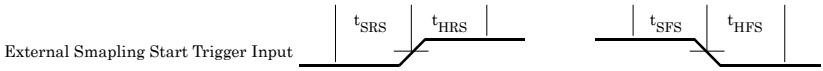


Figure 6.3. Timing Chart of Sampling Start Control Signal

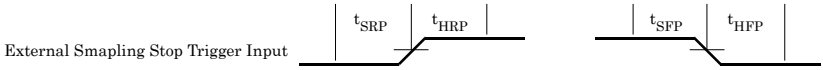


Figure 6.4. Timing Chart of Sampling Stop Control Signal

Table 6.2. Control Signal Timings

Parameter	Symbol	Time	Unit
Delay time from external sampling clock to first A/D start pulse	t_{DEC}	100	nsec
Set up time of sampling start (Rising edge)	t_{SRS}	100	nsec
Hold time of sampling start (Rising edge)	t_{HRS}	100	nsec
Set up time of sampling start (Falling edge)	t_{SFS}	100	nsec
Hold time of sampling start (Falling edge)	t_{HFS}	100	nsec
Set up time of sampling stop (Rising edge)	t_{SRP}	100	nsec
Hold time of sampling stop (Rising edge)	t_{HRP}	100	nsec
Set up time of sampling stop (Falling edge)	t_{SFP}	100	nsec
Hold time of sampling stop (Falling edge)	t_{HFP}	100	nsec

⚠ CAUTION

The times listed in Table 6.2 are for standard operating conditions.

Control Signal Timings for Analog Output

Figures 6.5, 6.6, 6.7 and Table 6.3 show the control signal timings for the analog input function.

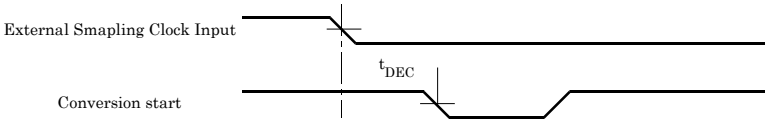


Figure 6.5. Timing Chart of External Sampling Clock (Analog output)

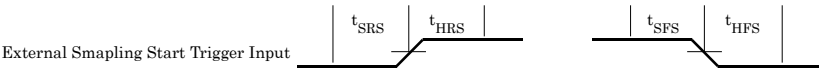


Figure 6.6. Timing Chart of Sampling Start Control Signal

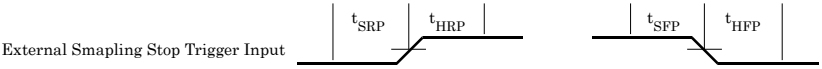


Figure 6.7. Timing Chart of Sampling Stop Control Signal

Table 6.3. Control Signal Timings

Parameter	Symbol	Time	Unit
Delay from external sampling clock to actual start	t_{DEC}	100	nsec
Settling time	t_{WS}	10000	nsec
Set up time of sampling start (Rising edge)	t_{SRS}	100	nsec
Hold time of sampling start (Rising edge)	t_{HRS}	100	nsec
Set up time of sampling start (Falling edge)	t_{SFS}	100	nsec
Hold time of sampling start (Falling edge)	t_{HFS}	100	nsec
Set up time of sampling stop (Rising edge)	t_{SRP}	100	nsec
Hold time of sampling stop (Rising edge)	t_{HRP}	100	nsec
Set up time of sampling stop (Falling edge)	t_{SFP}	100	nsec
Hold time of sampling stop (Falling edge)	t_{HFP}	100	nsec

⚠ CAUTION

The times listed in Table 6.3 are for standard operating conditions.

Control Signal Timings for Counter

Figures 6.8, 6.9, and Table 6.4 show the control signal timings for the analog input function.



Figure 6.8. Timing Chart of Counter Input Signal

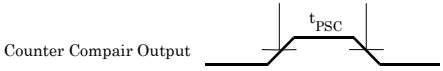


Figure 6.9. Timing Chart of Counter Output Signal (Pulse output)

Table 6.4. Control Signal Timings

Parameter	Symbol	Time	Unit
Set up time of counter input (Rising edge)	tSRC	100	nsec
Hold time of sampling start (Rising edge)	tHRC	100	nsec
Pulse width of counter output signal	tPSC	1000	nsec

⚠ CAUTION

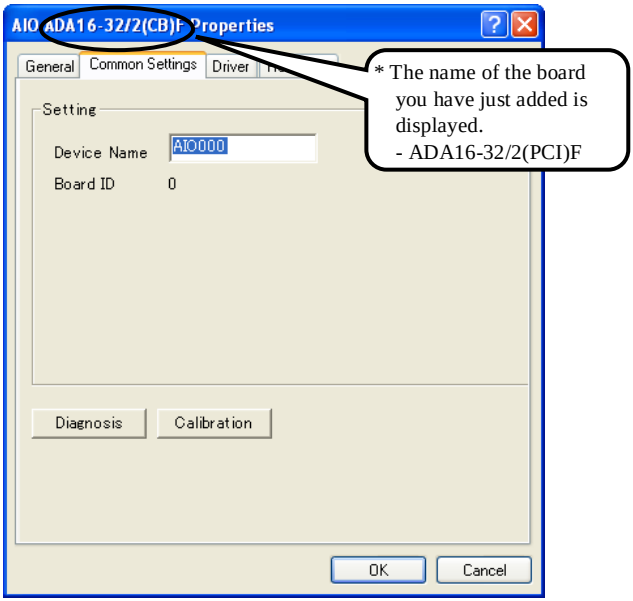
The times listed in Table 6.4 are for standard operating conditions.

About Calibration

Although this board is calibrated before shipping, you can use the calibration program to calibrate analog input and output yourself.

Starting the calibration program

Click the [Calibration] button on the property page for the device to start the calibration program.



Proceed with connecting the calibration equipment and performing the calibration in accordance with the instructions displayed by the calibration program.

Analog input calibration

Analog input calibration requires a reference voltage generator.
As the analog input has 16-bit resolution, use a reference voltage generator with a precision of at least 5 digits after the decimal point.
Calibrate one channel only for each range that you use.

Analog output calibration

Analog output calibration requires a digital multimeter.
As the analog output has 16-bit resolution, use a multimeter with a precision of at least 5 digits after the decimal point.
Calibrate each channel separately for each range that you use.

Factory setting

You can use the calibration program to restore the factory calibration settings.

If for some reason you are unable to achieve the rated accuracy, please contact the CONTEC information center.

ADA16-32/2(PCI)F

User's Guide

CONTEC CO., LTD.

December 2017 Edition

3-9-31, Himesato, Nishiyodogawa-ku, Osaka 555-0025, Japan
https://www.contec.com/

No part of this document may be copied or reproduced in any form by any means without prior written consent of CONTEC CO., LTD. [12252017]

[04102003]
[12252017_rev8]

Management No. A-46-739
Parts No. LYCP235

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