

Accel32 Manual

*Accel32 Server Description
and Installation Guide*

Version 4.14

Microstar Laboratories, Inc.

This manual contains proprietary information which is protected by copyright. All rights are reserved. No part of this manual may be photocopied, reproduced, or translated to another language without prior written consent of Microstar Laboratories, Inc.

Copyright © 1996 - 2002

Microstar Laboratories, Inc.
2265 116th Avenue N.E.
Bellevue, WA 98004
Tel: (425) 453-2345
Fax: (425) 453-3199
[http:// www.mstarlabs.com](http://www.mstarlabs.com)

Microstar Laboratories, DAPcell, Data Acquisition Processor, DAP, DAPL, and DAPview are trademarks of Microstar Laboratories, Inc.

Microstar Laboratories requires express written approval from its President if any Microstar Laboratories products are to be used in or with systems, devices, or applications in which failure can be expected to endanger human life.

Microsoft, MS, and MS-DOS are registered trademarks of Microsoft Corporation. Windows is a trademark of Microsoft Corporation. IBM is a registered trademark of International Business Machines Corporation. Intel is a registered trademark of Intel Corporation. Novell and NetWare are registered trademarks of Novell, Inc. Other brand and product names are trademarks or registered trademarks of their respective holders.

Version 4.14

Contents

1. Accel32	1
Software Components.....	2
AcomK	3
AcomK.sys - the Kernel Driver for the Windows NT/2000/XP Operating System.....	3
AcomK.vxd - the Virtual Device for the Windows 9x/ME Operating System.....	3
AcomS – the Accel32 Service	4
Mslacom - the Accel32 Control Panel Extension Program	5
2. Installation	7
Installation with PCI Data Acquisition Processors	7
Under Windows XP/2000/98/ME	7
Under Windows 95	7
Under Windows NT 4.0	8
Installation with ISA Data Acquisition Processors.....	8
3. Using Accel32	11
Communication Pipes.....	12
UNC (Universal Naming Convention) Pipe Names	13
Mapping ACCEL Devices to UNC Pipe Names.....	15
Running 32-bit Data Acquisition Processor Applications	18
Running 16-bit DOS and Windows Applications.....	19
4. Developing 32-bit Data Acquisition Applications	21
C/C++	22
Delphi	23

1. Accel32

Accel32 is a software package that offers a communication interface between application programs and Data Acquisition Processors under the Windows XP, Windows 2000, Windows ME, Windows 98, Windows 95 or the Windows NT 4.0 operating system.

Accel32 employs a client/server model. The clients are data acquisition applications using a client interface provided by a dynamic link library; the server is a Win32 service.

Please note the following restrictions for Accel32:

- It is compatible with 80x86 PC platforms only.
- It supports only the e-series and a-series Data Acquisition Processors.
- It supports DAPL 4.10 or later and DAPL 2000 1.00 or later.

Accel32 offers seamless communication between PC programs and one or more Data Acquisition Processors on the PC. Up to 14 Data Acquisition Processors can be hosted on one machine with the support of Accel32.

Almost all existing 16-bit DOS and Windows Data Acquisition Processor applications run unmodified with Accel32 under the Windows NT/2000/XP operating system. Accel32 does not support DOS Data Acquisition Processor applications under the Windows 9x/ME operating system.

The Accel32 distribution includes support files for developing 32-bit Windows applications in Visual C++ (4.0 and later) and Borland C++ (5.0 and later) and Borland Delphi (2.0 and later). See the `DAPIO32.HLP` file for a complete description of the services provided by the DAPIO32 interface.

Software Components

Accel32 server software consists of a kernel-mode driver and a user-mode service. The kernel driver operates on the Data Acquisition Processor hardware; and the service offers the functionality that an application needs in order to communicate with the Data Acquisition Processor.

Accel32 client software provides a consistent interface for DOS, 16-bit Windows, and 32-bit Windows applications.

Following is a list of files that constitute Accel32:

ACOMS.EXE	—	server service program
ACOMK.SYS	—	server kernel driver for the Windows NT/2000/XP operating system
ACOMK.VXD	—	server virtual device for the Windows 9x/ME operating system
ACOMSVC.DLL	—	server dynamic link library
DAPIO32.DLL	—	client dynamic link library
WACCEL.DRV	—	client 16-bit Windows driver stub
ACOM32.DLL	—	client virtual device driver for the Windows NT/2000/XP operating system
ACOM.NT	—	client DOS driver stub for the Windows NT/2000/XP operating system
MSLACOM.CPL	—	Accel32 control panel extension program

Not all the files are required to run a Data Acquisition Processor application. WACCEL.DRV is only required to run a 16-bit Windows application. ACOM32.DLL and ACOM.NT are only required to run a DOS application.

AcomK

AcomK.sys - the Kernel Driver for the Windows NT/2000/XP Operating System

In the Windows NT/2000/XP operating system, a kernel driver is required to provide hardware access to the Data Acquisition Processor. The kernel driver of Accel32 is AcomK.sys. It is installed in the Devices applet of the Control Panel by the SETUP.EXE installation program.

AcomK.sys automatically starts when Windows starts and remains active across sessions until Windows shuts down. Any attempt to stop the kernel driver will result in disabling Accel32.

AcomK.vxd - the Virtual Device for the Windows 9x/ME Operating System

Windows 9x/ME uses virtual devices to support multi-tasking for Windows-based applications. The virtual device of Accel32 is AcomK.vxd. It works with Windows to process interrupts and carry out I/O operations for a given application without disrupting the execution of other applications.

AcomK.vxd automatically starts when the Accel32 service starts and stops when the service stops. Any attempt to stop the virtual device will result in disabling Accel32.

AcomS – the Accel32 Service

The Accel32 service is a user-mode program that implements the functionality that Accel32 offers. It is installed by the `SETUP.EXE` installation program as a Win32 service with the name Accel32 Service. The Accel32 service typically starts when the Windows system starts and remains active across sessions until the Windows system shuts down.

The Accel32 service initialization involves both the service software initialization and the Data Acquisition Processor initialization. Data Acquisition Processor communication is established as soon as the service initialization is complete.

The service software initialization detects Data Acquisition Processor hardware, allocates system resources, and sets up service control and data structures. The Accel32 service starts only if the entire initialization process is completed successfully. If an error occurs during the initialization, the service prints out a message and exits.

The Accel32 service initializes all of the Data Acquisition Processors on the system by downloading the correct DAPL operating system to each Data Acquisition Processor. A Data Acquisition Processor will not function without the DAPL operating system. If initialization of a Data Acquisition Processor fails, the service prints out a message and continues. However, no communication with that Data Acquisition Processor is possible even if the Accel32 service is started and remains active.

Running under the Windows NT/2000/XP operating system, Accel32 takes advantage of the system event log to provide more details about the error or warning.

After the Accel32 service is started, it can be stopped and re-started from the Control Panel. The service should only be stopped and re-started if re-initialization of the Data Acquisition Processor system is necessary. For example, if a Data Acquisition Processor fails to initialize because of a missing or bad DAPL file, the Data Acquisition Processor can be re-initialized by locating the correct DAPL file, stopping the service and then starting it again.

Each time the Accel32 service starts, it resets the Data Acquisition Processor by default. A Data Acquisition Processor that has already been functioning is re-initialized with a fresh DAPL operating system. Programs and data on that Data Acquisition Processor are permanently lost.

Mslacom - the Accel32 Control Panel Extension Program

An Accel32 control panel extension program `Mslacom.cpl` is installed in the Windows Control Panel with the name of “Data Acquisition Processors”. This program serves as the Accel32 service manager, providing user control over the service through five screens, the Control Screen, the Browser Screen, the Disk I/O Screen, the Mapping Screen, and the Modules Screen.

The Control Screen allows users to start, stop and configure the startup behavior of the Accel32 Service. It also allows user control over the execution priority of the service program. A configuration change does not take effect until the Service is re-started.

The Browser Screen allows users to view the configuration of the Data Acquisition Processors under the Service, including the hardware and the pipe information of the Data Acquisition Processors.

The Disk I/O screen is not available for Accel32. It provides additional functionality for the DAPcell or DAPcell Local Service.

The Mapping Screen provides easy user access to two utility programs that help create mappings between DOS ACCEL devices and Accel32 pipes and between pre-defined Accel32 DAP names and customized names.

The Modules Screen provides users with easy control over installation of DAPL downloadable command modules.

2. Installation

The installation procedure for Accel32 is slightly different depending on whether the Data Acquisition Processor is connected to an ISA bus or a PCI bus.

Installation with PCI Data Acquisition Processors

Under Windows XP/2000/98/ME

Installing Accel32 with one or more PCI Data Acquisition Processors requires interaction with the Windows Plug-and-Play Hardware Wizard. Following is a typical scenario for the first time installation:

- Power down the computer. Follow the installation procedures to install PCI Data Acquisition Processor hardware into available PCI slots.
- Power up and start Windows.
- Windows Plug-and-Play Hardware Wizard will start upon detecting a DAP as a PCI device.
- Select “Search for a suitable driver for my device (recommended)” and click “Next”.
- Insert the Accel32 Setup CD-ROM disk into the CD-ROM drive.
- Check “CD-ROM” and click “Next”.
- The next dialog will indicate that the Wizard has found a driver for this device. Click “Next”.
- The Accel32 SETUP . EXE program will start automatically. Follow its instructions to complete the installation.
- Click “Finish” to complete the installation with the Hardware Wizard.

Subsequent installation of PCI Data Acquisition Processors is similar to the first time installation except that SETUP . EXE has to be manually invoked after the completion of the installation with the Hardware Wizard.

Under Windows 95

Installing Accel32 with one or more PCI Data Acquisition Processors requires interaction with the Windows 95 Plug-and-Play Hardware Wizard. Following is a typical scenario for the first time installation:

- Power down the computer. Follow the installation procedures to install PCI Data Acquisition Processor hardware into available PCI slots.
- Power up and start Windows 95.
- Windows 95 Plug-and-Play Hardware Wizard will start upon detecting a DAP as a PCI device.
- Select “Driver from disk provided by hardware manufacturer” and click “OK”.
- Specify a location, insert the Accel32 Setup CD-ROM disk to that location and click “OK”.
- The “Select Device” dialog will appear with a highlighted “PCI Data Acquisition Processor”. Click “OK”.
- The Accel32 SETUP.EXE program will start automatically. Follow its instructions to complete the installation.

For a subsequent installation, the Hardware Wizard will find the device upon bootup and prompt with a dialog box. Just select “Windows default driver” and click “OK” or go through the same procedures as the first installation except in both cases SETUP.EXE has to be manually started for the rest of the installation.

Under Windows NT 4.0

The Installation procedure of the Accel32 software with PCI Data Acquisition Processors under Windows NT 4.0 is similar to the installation procedure with ISA Data Acquisition Processors.

Installation with ISA Data Acquisition Processors

Installing Accel32 with ISA Data Acquisition Processors only is a straightforward and self-explanatory procedure. Just insert the release CD-ROM disk into the CD-ROM drive and run SETUP.EXE. The program will automatically detect the operating system and install the software appropriate to the target system.

Before installing Accel32, make sure that all Data Acquisition Processors are properly installed and configured, and that there is no interrupt or I/O address conflict between the Data Acquisition Processors and the other devices on the system. If DOS is available on the machine, it is worthwhile to verify that all Data Acquisition Processors function correctly under DOS.

SETUP.EXE has an option to allow it to auto-detect the configuration of the Data Acquisition Processor system on the host machine. It is recommended to select this option so that SETUP.EXE can detect and report configuration errors or system resource conflicts if there are any.

SETUP.EXE also copies the required DAPL files to a directory that Accel32 knows to find. Have the Data Acquisition Processor release CD-ROM disk available before running the installation program.

SETUP.EXE installs most of the Accel32 components in the Windows system directory.

SETUP.EXE also creates registry keys under
HKEY_LOCAL_MACHINE\SOFTWARE\Microstar_Laboratories to provide the system with Data Acquisition Processor configuration information. Accel32 uses the information in the registry to initialize itself every time it starts.

To change an existing Data Acquisition Processor configuration, close all applications and run SETUP.EXE again.

To un-install Accel32, re-run SETUP.EXE and select the option to unregister Accel32 from the “Detect Dap Hardware” screen.

3. Using Accel32

Accel32 offers a consistent interface between applications and the Data Acquisition Processor. Applications communicate with the Data Acquisition Processor through a communication channel structure called the “communication pipe”. An application opens a handle to a pipe and then uses the handle to send or receive data through the pipe. A pipe can be opened for reading or writing only once; and once opened for that purpose, the pipe is reserved for access by the application exclusively until the application closes the open handle.

Communication Pipes

Like DOS and Windows DAP drivers from Microstar Laboratories, Accel32 buffers data from and to the Data Acquisition Processor. This buffering structure is called the “communication pipe”. There are communication pipes in the DAPL operating system running on the Data Acquisition Processor; there are also pipes in the PC. The communication pipes on both sides are logically connected on a one-to-one basis. The pipes in the PC can be viewed as extensions to the communication pipes on the Data Acquisition Processor. Each connected pair of pipes form a communication channel between PC applications and the Data Acquisition Processor. In Accel32, a communication channel is represented by the corresponding communication pipe on the Data Acquisition Processor.

There are four default communication pipes on each Data Acquisition Processor, \$SysIn, \$SysOut, \$BinIn and \$BinOut. They are uni-directional pipes for text and binary communication. PC applications typically interact with the Data Acquisition Processor by reading from or writing to one of the communication pipes. When Accel32 starts, it automatically creates the four default communication channels to each Data Acquisition Processor.

The four default communication channels provide most applications with adequate and complete communication with the Data Acquisition Processor. Additional communication channels can be created by using the DAPIO32 function `DapComPipeCreate`. See the DAPIO32 Reference Manual for information about how to use `DapComPipeCreate`.

UNC (Universal Naming Convention) Pipe Names

Accel32 addresses a pipe on the Data Acquisition Processor using the Universal Naming Convention (UNC). A UNC pipe name consists of three portions, a machine name, a Data Acquisition Processor name, and a pipe name. A UNC name is led by two backslashes with each component delimited by one backslash.

The machine is the local machine, denoted by a period.

The Data Acquisition Processor names are pre-defined as `Dap0`, `Dap1`, ..., and `Dap(N-1)` where N is the number of Data Acquisition Processors installed on the system.

For ISA Data Acquisition Processors, the ordering depends on the numeric ordering of the I/O addresses the Data Acquisition Processors use. For example, in a one Data Acquisition Processor system, the name of the Data Acquisition Processor is always `Dap0`. In a two Data Acquisition Processor system with addresses 0x220 and 0x230, the Data Acquisition Processor at 0x220 is named `DAP0` and the Data Acquisition Processor at 0x230 `Dap1`.

For PCI Data Acquisition Processors, the ordering depends on the PCI bus and slot numbers. A Data Acquisition Processor on a bus with a lower number is named before a Data Acquisition Processor on a bus with a high number. Within the same bus, a Data Acquisition Processor in a slot of a lower number is named before a Data Acquisition Processor in a slot of a higher number. For example, in a three Data Acquisition Processor system, the Data Acquisition Processor on bus 0 slot 10 is named `Dap0`, the one on bus 1 slot 0 is named `Dap1` and the one on bus 1 slot 2 is named `Dap2`.

The pipe names also are pre-defined. On the Data Acquisition Processor, two communication pipes are associated with one integer number but differ in transfer directions (either input or output). `$SysIn` and `$SysOut` are the default input and output pipes with the number zero while `$BinIn` and `$BinOut` are the default input and output pipes with the number one. Accel32 supports a maximum of 32 sets of input and output communication pipes on each Data Acquisition Processor; thus, the largest number that can be associated with a pipe is 31. Except for the two default sets, all pre-defined pipe names carry both the pipe number information and the pipe direction information. Following is a list of the 32 supported sets of communication pipes:

<code>\$SysIn</code>	<code>\$SysOut</code>
<code>\$BinIn</code>	<code>\$BinOut</code>

Cp2In	Cp20ut
Cp3In	Cp30ut
Cp4In	Cp40ut
...	
Cp31In	Cp310ut

Thus, the communication pipe \$SysIn of the Data Acquisition Processor Dap0 is represented by the UNC name \\.\Dap0\\$SysIn, and the pipe \$BinOut of the Data Acquisition Processor Dap1 \\.\Dap1\\$BinOut. For example,

```

\\.\DapX\$SysIn
\\.\DapX\$SysOut
\\.\DapX\$BinIn
\\.\DapX\$BinOut

```

are the four default communication pipes on the Data Acquisition Processor DapX.

Mapping ACCEL Devices to UNC Pipe Names

In the DOS and Windows DAP drivers, a communication channel to a Data Acquisition Processor is seen as an ACCEL device, either as a directed ACCEL device or as a numbered ACCEL device. (See the DAPL System Manual for information about ACCEL devices.) In order for Accel32 to support existing DOS and Windows applications, it is necessary to provide a mechanism to map the ACCEL device names to the corresponding UNC pipe names that Accel32 recognizes.

By default, Accel32 automatically maps ACCEL0 to \\.\Dap0\\$\SysIn and \\.\Dap0\\$\SysOut, and ACCEL1 to \\.\Dap0\\$\BinIn and \\.\Dap0\\$\BinOut. Then it maps ACCEL2 to \\.\Dap1\\$\SysIn and \\.\Dap1\\$\SysOut, and ACCEL3 to \\.\Dap1\\$\BinIn and \\.\Dap1\\$\BinOut. And so on, depending on how many Data Acquisition Processors are installed.

Custom mapping is fulfilled by a text file called ACOM.INI. This file specifies a complete set of mappings that an application requires. The Accel32 service reads in the file and makes the mapping according to the specification if such a mapping file exists.

ACOM.INI consists of at least two sections. The first section takes a pre-defined name ACCEL with a pre-defined key Configuration following it. The value of the key is a user-defined section name that includes the active mapping. More than one user-defined mapping section can exist, but only the one whose name is specified as the value of the key Configuration is considered active. Following is the section structure for an example ACOM.INI file. It has two mapping sections [oneDAP] and [twoDAPs]. Section [oneDAP] is active:

```
[ACCEL]
Configuration=oneDAP
;Configuration=twoDAPs
[oneDAP]
...
[twoDAPs]
...
```

Under a mapping section, the keys are the numbered ACCEL device names used by DOS and 16-bit Windows applications, such as ACCEL0, ACCEL1,..., followed by a direction symbol, > or <. The direction symbol distinguishes between input and output devices. The values of the keys are the complete target DAP pipe UNC addresses as described before.

```
ACCEL0 >= \\.\Dap0\$SysIn
```

The above key-value pair maps ACCEL0 to the \$SysIn of Dap0. When a 16-bit application opens ACCEL0, it opens a connection to the \$SysIn of Dap0. When the application uses the opened handle to write a block of data, the data will go to the \$SysIn of Dap0.

```
ACCEL1 <= \\.\Dap0$BinOut
```

The above key-value pair maps ACCEL1 to the \$BinOut of Dap0. When a 16-bit application opens ACCEL1, it opens a connection to the \$BinOut of Dap0. When the application uses the opened handle to read a block of data, the data are from the \$BinOut of Dap0.

Following are two complete example mapping sections:

- The [oneDAP] section maps ACCEL0 to Dap0 \$SysIn and \$SysOut, ACCEL1 to Dap0 \$BinIn and \$BinOut and ACCEL4 to Dap0 Cp4In and Cp4Out.
- The [twoDAPs] section maps ACCEL0 and ACCEL1 to Dap0 \$SysIn, \$SysOut, \$BinIn, and \$BinOut, and ACCEL8 and ACCEL9 to Dap1 \$SysIn, \$SysOut, \$BinIn, and \$BinOut for a two board system.

```
[oneDAP]
ACCEL0 >= \\.\Dap0\$SysIn
ACCEL0 <= \\.\Dap0\$SysOut
ACCEL1 >= \\.\Dap0$BinIn
ACCEL1 <= \\.\Dap0$BinOut
ACCEL4 >= \\.\Dap0\Cp4In
ACCEL4 <= \\.\Dap0\Cp4Out
```

```
[twoDAPs]
ACCEL0 >= \\.\Dap0\$SysIn
ACCEL0 <= \\.\Dap0\$SysOut
ACCEL1 >= \\.\Dap0$BinIn
ACCEL1 <= \\.\Dap0$BinOut
ACCEL8 >= \\.\Dap1\$SysIn
ACCEL8 <= \\.\Dap1\$SysOut
ACCEL9 >= \\.\Dap1$BinIn
ACCEL9 <= \\.\Dap1$BinOut
```

ACOM.INI usually goes to the Windows root directory. The complete set of pipes used in an application needs to be mapped whether a subset of the mapping matches the default assignment or not. If a DOS application opens a directed ACCEL device

(ACCEL without a number), it opens all of the mapped pipes specified in the active section.

Though such a mapping file can be created by hand-editing the file, it is recommended to create the mapping file through the Control Panel. The Accel32 control panel extension program “Data Acquisition Processors” allows you to create a custom mapping simply by mouse dragging and dropping.

Running 32-bit Data Acquisition Processor Applications

To run a 32-bit Data Acquisition Processor application with Accel32, simply start the program while Accel32 is active. Accel32 supports the Win32 multi-tasking and multi-threading features. More than one Data Acquisition Processor application and more than one thread of an application can run at the same time with the restriction that the sets of communication pipes they access do not overlap.

Accel32 prevents an application from acquiring access to a communication pipe that has been opened by another application. No two applications can share a communication pipe.

Accel32 allows an application to pass an opened handle from one thread to another. The application, however, usually should assign only one thread to perform direct I/O access with that handle. Simultaneous I/O access from multiple threads with the same handle may yield unexpected results.

Running 16-bit DOS and Windows Applications

16-bit DOS and Windows applications require additional software support from Accel32.

A 16-bit Windows application requires the 16-bit Windows device driver stub WACCEL.DRV as well as the DAPIO.DLL from the Microstar Laboratories Windows Toolkit. Multiple simultaneous 16-bit Windows Data Acquisition Processor applications are possible under the Windows NT/2000/XP operating system as long as they follow the restrictions mentioned above. Only a single 16-bit Windows Data Acquisition Processor application can run at a time under the Windows 9x/ME operating system.

DOS Data Acquisition Processor applications under the Windows NT/2000/XP operating system require a DOS driver stub ACOM.NT as well as a virtual device driver ACOM32.DLL. DOS Data Acquisition Processor applications are not supported under the Windows 9x/ME operating system.

To install the DOS driver stub under the Windows NT/2000/XP operating system, edit the file CONFIG.NT in the directory %SystemRoot%\System32 to add the following statement:

```
device = %SystemRoot%\System32\ACOM.NT
```

The file ACOM.NT and the virtual device driver ACOM32.DLL must reside in the proper directory. If this condition is met, ACOM.NT will be loaded whenever a virtual DOS machine is created. A DOS application running in that DOS virtual machine will be able to use the DOS driver stub and the virtual device driver as if it were using the DOS ACCEL driver under the DOS operating system. The DOS driver stub communicates with Accel32 through the virtual device driver ACOM32.DLL. Note that the DOS driver stub is not available in those virtual DOS machines that have been created before CONFIG.NT is edited. To load the driver stub, close those virtual DOS machines and open them again.

Almost all existing 16-bit applications run unmodified with Accel32. However, users should be aware of the following changes:

- The Accel32 initialization does not require running the programs ACOMINIT.EXE or DAPLINIT.EXE.
- For some applications that run on a multiple Data Acquisition Processor system or that use additional communication pipes, pipe mapping and/or manual pipe

creation may be necessary. Pipe mapping can be done by providing an `ACOM.INI` file; additional communication pipes can be created by using the example program `DAPPIPE.EXE` included in the Accel32 release CD-ROM disk.

- 16-bit Windows applications are no longer able to use the 16-bit Windows virtual ACCEL driver `VACCEL.386`. These applications communicate with Accel32 through the 16-bit Windows ACCEL driver stub `WACCEL.DRV`. The 16-bit `DAPIO.DLL` function `IsVxdRunning()` still returns `TRUE` even though `VACCEL.386` is not present. Note that the file of the stub `WACCEL.DRV` is not the same file as the 16-bit Windows installable driver `WACCEL.DRV` that is provided in the Microstar Laboratories Windows Toolkit.

4. Developing 32-bit Data Acquisition Applications

A 32-bit Data Acquisition Processor application can be created using any 32-bit application development tool that supports DLL calls.

Support files are provided with Accel32 for developing 32-bit applications with Microsoft Visual C++ (4.0 and later), Borland C++ (5.0 and later), and Borland Delphi (2.0 and later). Applications can be developed in other environments that support calling DLL functions by producing appropriate interface declarations for the functions provided in DAPI032.DLL in the environment. See the DAPI032 Reference Manual and the DAPI032.HLP file for the set of supported functions.

Developed by Microstar Laboratories, DAPtools OCX is available for use in Microsoft Visual Basic 4.0 and other environments that support ActiveX controls. Please contact your sales representative for more information on DAPtools OCX.

C/C++

To use the services provided by DAPIO32, a C/C++ application must include the header file `DAPIO32.H`, and must be linked with a version of `DAPIO32.LIB` that is compatible with the compiler used. `DAPIO32.H`, and both a Microsoft C++ compatible `DAPIO32.LIB` and a Borland C++ compatible `DAPIO32.LIB` are included with the installed software.

Delphi

To use the services provided by DAPIO32, a Delphi application must use the unit DAPIO32 contained in the file DAPIO32.PAS. DAPIO32.PAS is provided with the installed software.