

CPCI-069-041104

コンパクト PCI CPU ボード

aPCI-8011

ソフトウェアマニュアル

株式会社 **アドテック システム サイエンス**

目次

1 . はじめに.....	2
2 . 動作環境.....	2
3 . ファイル一覧.....	3
4 . ドライバ.....	4
4 - 1 . Windowsへのインストール.....	4
4 - 1 - 1 . Windows2000 へのインストール.....	4
4 - 1 - 2 . WindowsXPへのインストール.....	12
4 - 2 . アンインストール.....	21
4 - 2 - 1 . Windows2000 からのアンインストール.....	21
4 - 2 - 2 . WindowsXPからのアンインストール.....	24
5 . サンプルソース.....	27
6 . API仕様.....	28
6 - 1 . 概要.....	28
6 - 2 . プログラム構成.....	28
6 - 3 . APIリファレンス.....	29
6 - 3 - 1 . Apci8011Create.....	29
6 - 3 - 2 . Apci8011Close.....	30
6 - 3 - 3 . Apci8011GetVersion.....	31
6 - 3 - 4 . Apci8011EnableWatchdog.....	32
6 - 3 - 5 . Apci8011RefreshWatchdog.....	33
6 - 3 - 6 . Apci8011DisableWatchdog.....	34
6 - 3 - 7 . Apci8011GetLastError.....	35
6 - 4 . 定義.....	36
6 - 4 - 1 . エラーコード.....	36
製品のお問い合わせについて.....	37
改訂履歴.....	38

1. はじめに

本マニュアルは、コンパクト PCI CPU ボード「aPCI-8011」を Windows2000/WindowsXP でご利用いただくためのドライバのインストール手順および API 仕様などについて記述しています。ハードウェアに関する詳細は、ハードウェアマニュアルを参照してください。

2. 動作環境

API は、Windows(R) 2000 および Windows(R) XP 上で、aPCI-8011 (以下「デバイス」)が有するウォッチドッグタイマ(アプリケーションの暴走などによる処理異常を検出するための機能) を制御します。

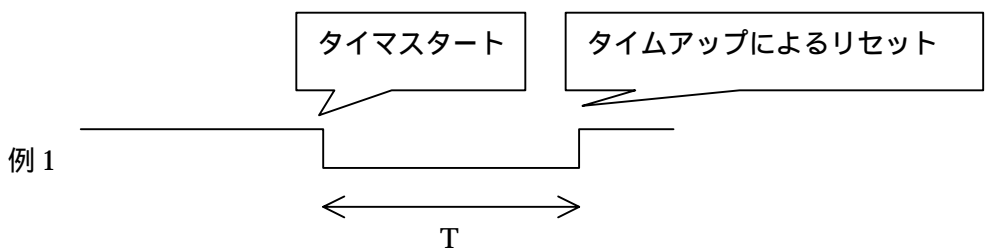
ウォッチドッグタイマを使用しない場合は、インストールの必要はありません。

ウォッチドッグタイマとは

このタイマの特徴は、タイマがタイムアップするとハードウェアにリセットがかかるということです。リセットがかかると当然コンピュータは初期状態からのスタートになり再スタートすることになります。

これを図で説明すると下図の様になります。

まず例 1 は単純にウォッチドッグタイマをスタートさせただけの時です。このときには指定した時間 (T) が経つとウォッチドッグタイマがタイムアップしコンピュータにリセット信号が出力され再スタートします。



例 2
Apci8011RefreshWatchdog()の実行

このタイマの使い方は上図例 2 のように、タイマを最初スタートさせたら、あとは常に一定時間以内に (タイマがタイムアップする前に) タイマを Apci8011RefreshWatchdog() でクリアしてタイマのカウントを再スタートさせます。

これが続いている限りはタイマがタイムアップすることが無いので、リセットがかかることは有りません。

3. ファイル一覧

¥		
├	Win2000_XP	Windows2000/XP 用 32bit ドライバ
├	├ apci8011.inf	インストールファイル
├	├ apci8011.sys	ドライバ
├	├ apci8011.dll	ドライバ制御 DLL
├	Sample	サンプルソース
├	├ VC	Visual C++ サンプルソース Ver.4/5/6 対応
├	├ ├ apci8011.h	DLL 定義ヘッダ
├	├ ├ apci8011w.h	ラッパー関数ヘッダ
├	├ ├ apci8011w.c	ラッパー関数
├	├ ├ apci8011s.c	ラッパー関数使用例
├	├ ├ buildvc.txt	サンプルソース構築例
├	├ ├ Lib	
├	├ ├ ├ Win2000_XP	
├	├ ├ ├ ├ apci8011.lib	インポートライブラリ
├	├ VB	Visual Basic サンプルソース Ver.4(32bit)/5/6 対応
├	├ ├ apci8011.bas	DLL 定義
├	├ ├ apci8011w.bas	ラッパー関数
├	├ ├ apci8011s.bas	ラッパー関数使用例
├	├ ├ buildvb.txt	サンプルソース構築例
├	├ Delphi	Delphi サンプルソース Ver.2/3/4/5/6/7 対応
├	├ ├ apci8011w.pas	DLL 定義 / ラッパー関数
├	├ ├ apci8011s.pas	ラッパー関数使用例
├	├ ├ apci8011s.dfm	フォームファイル
├	├ ├ builddp.txt	サンプルソース構築例
├	apci8011.hlp/cnt	ヘルプ
├	apci8011ap.txt	API リファレンス
├	readme.txt	リードミー

4 . ドライバ

4 - 1 . Windows へのインストール

4 - 1 - 1 . Windows2000 へのインストール

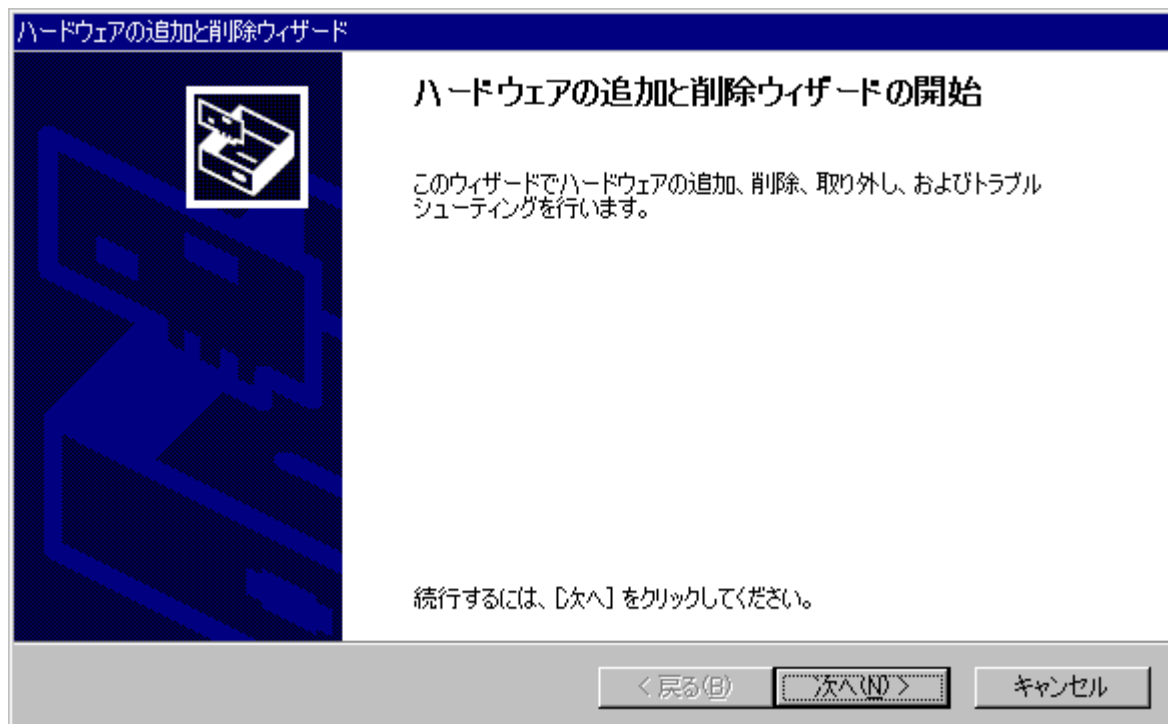
本製品を Windows2000 上で使用するためには、ハードウェア・ウィザードによるインストールが必要です。

以下の手順に従って登録を行ってください。

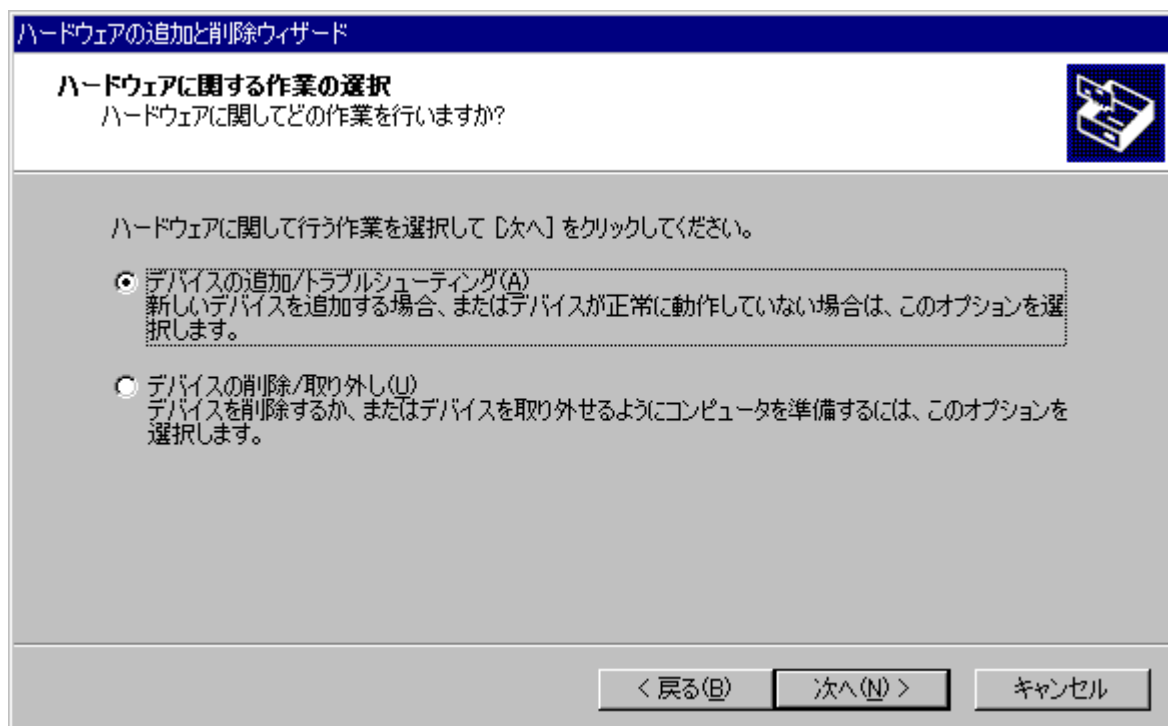
1. 「Administrator」でログオンし、[スタート - 設定 - コントロールパネル]を実行してコントロールパネルを開いてください。開いたら[ハードウェアの追加と削除]アイコンをダブルクリックします。



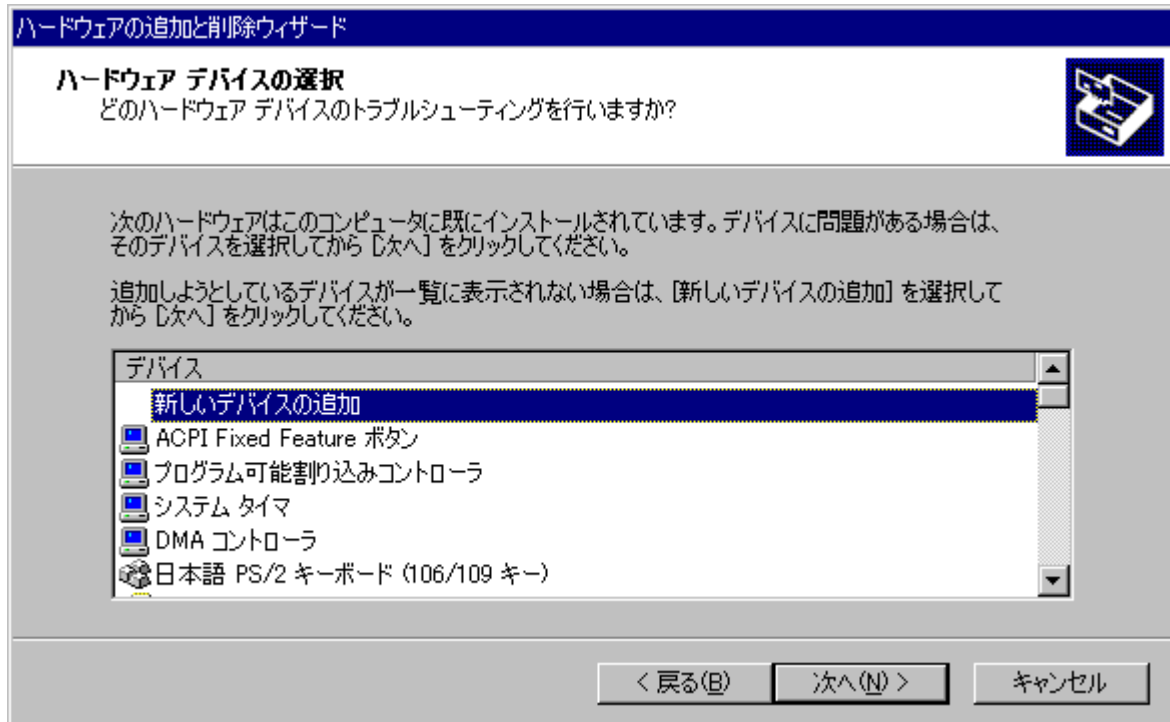
2. [次へ]ボタンをクリックします。



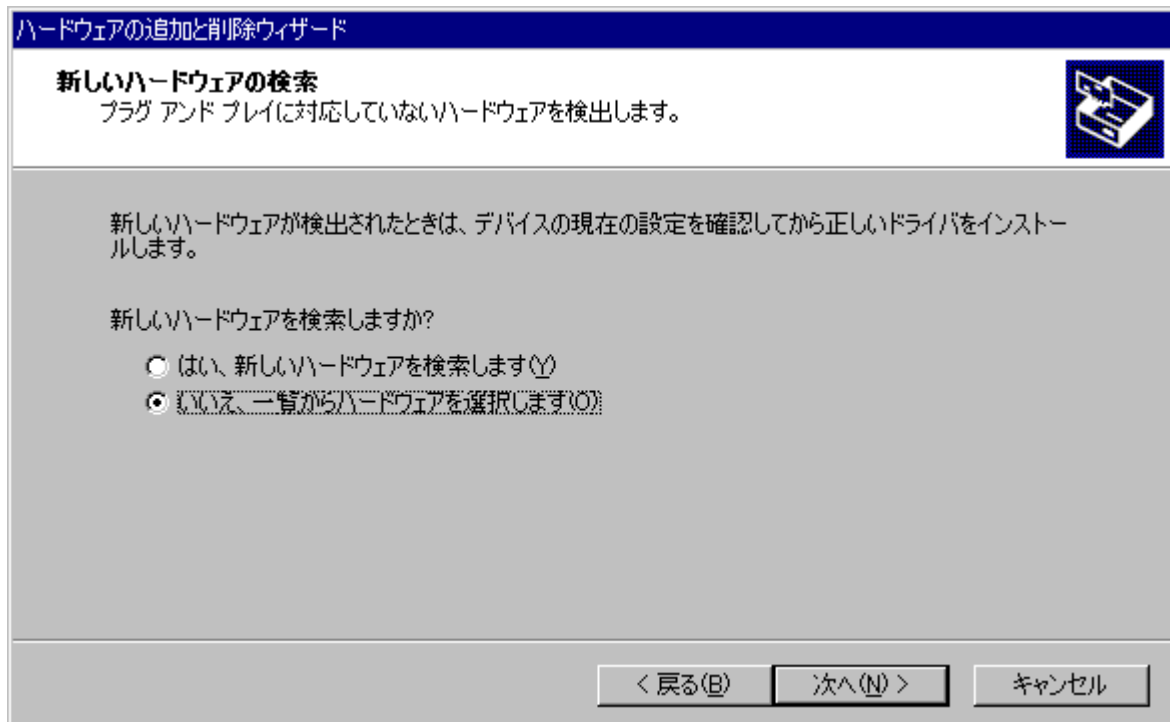
3. [デバイスの追加/トラブルシューティング]を選択し、[次へ]ボタンをクリックします。



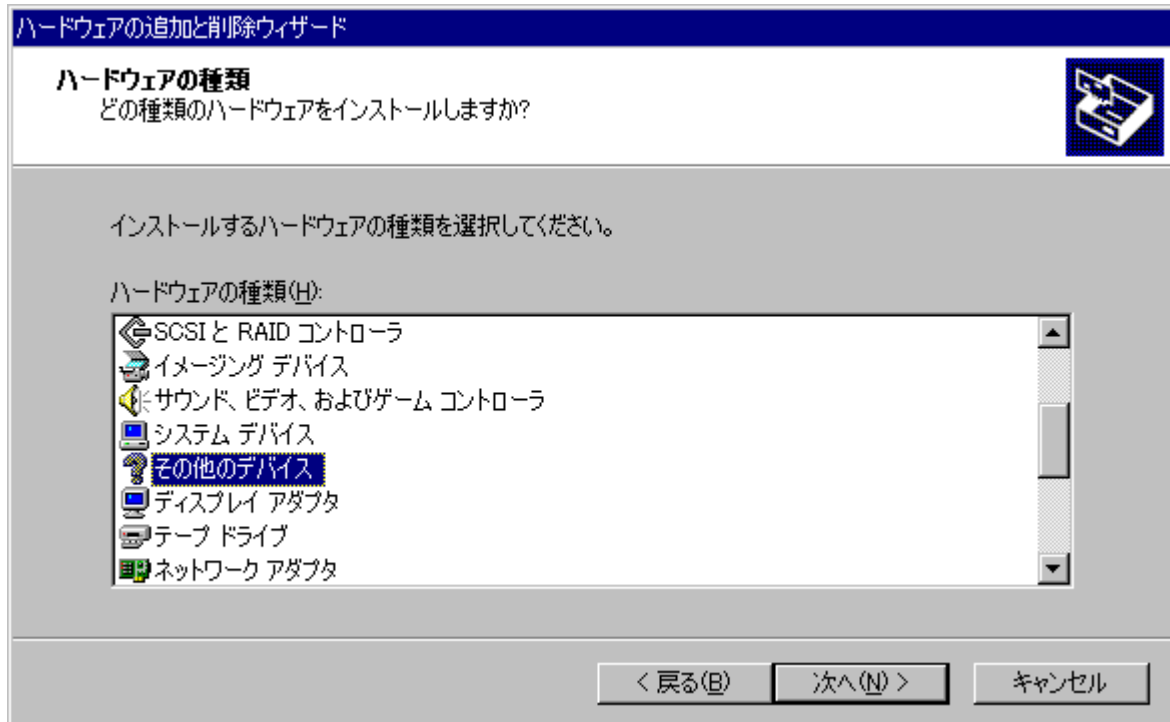
4. [新しいデバイスの追加]を選択し、[次へ]ボタンをクリックします。



5. [いいえ、一覧からハードウェアを選択します]を選択し、[次へ]ボタンをクリックします。



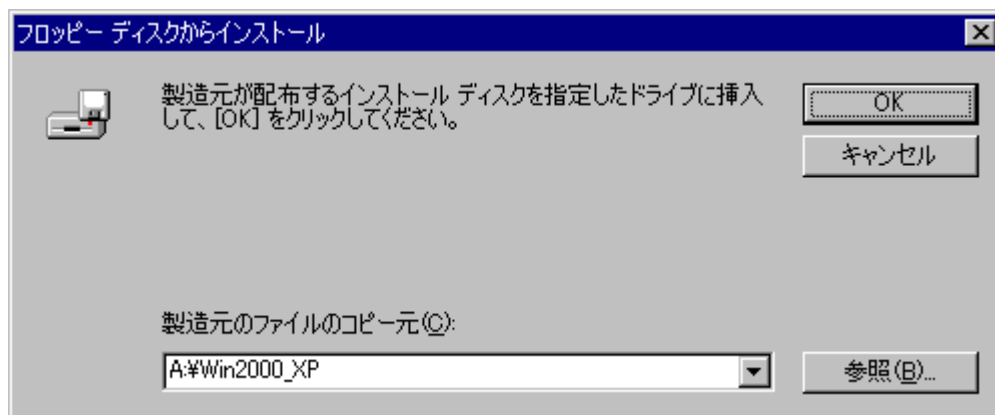
6. [その他のデバイス]を選択し、[次へ]ボタンをクリックします。



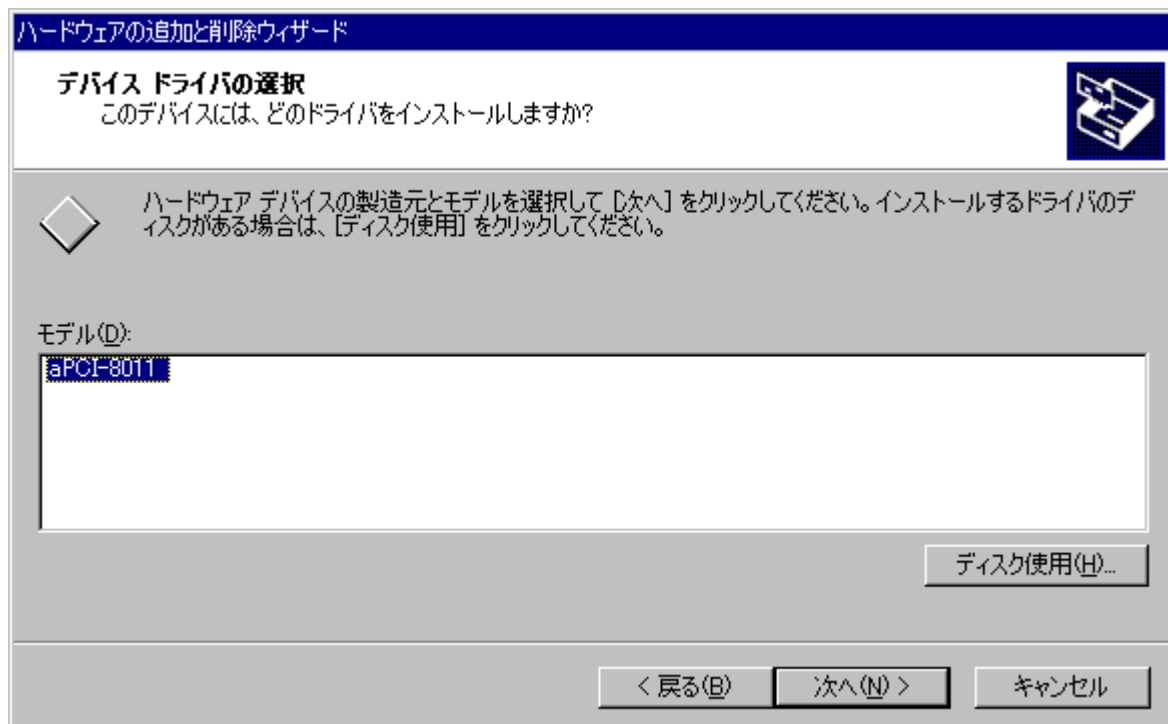
7. [ディスク使用]ボタンをクリックします。



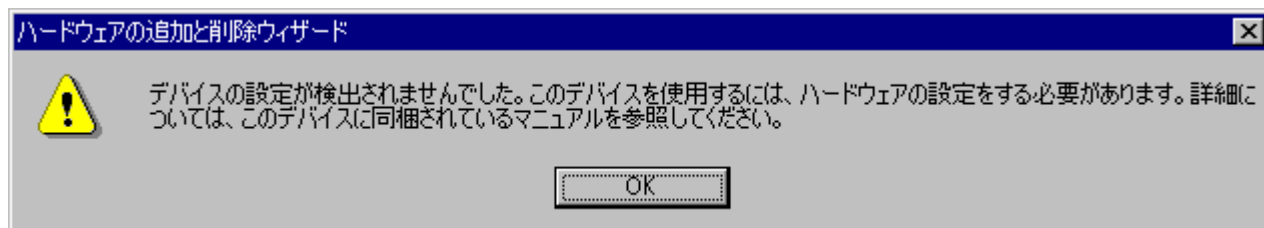
コピー元にドライバソフトの Win2000_XP ディレクトリを指定し、[OK]ボタンをクリックします。
ここでは「A:\Win2000_XP」を指定しています。



8. 使用するデバイスを選択し、[次へ]ボタンをクリックします。



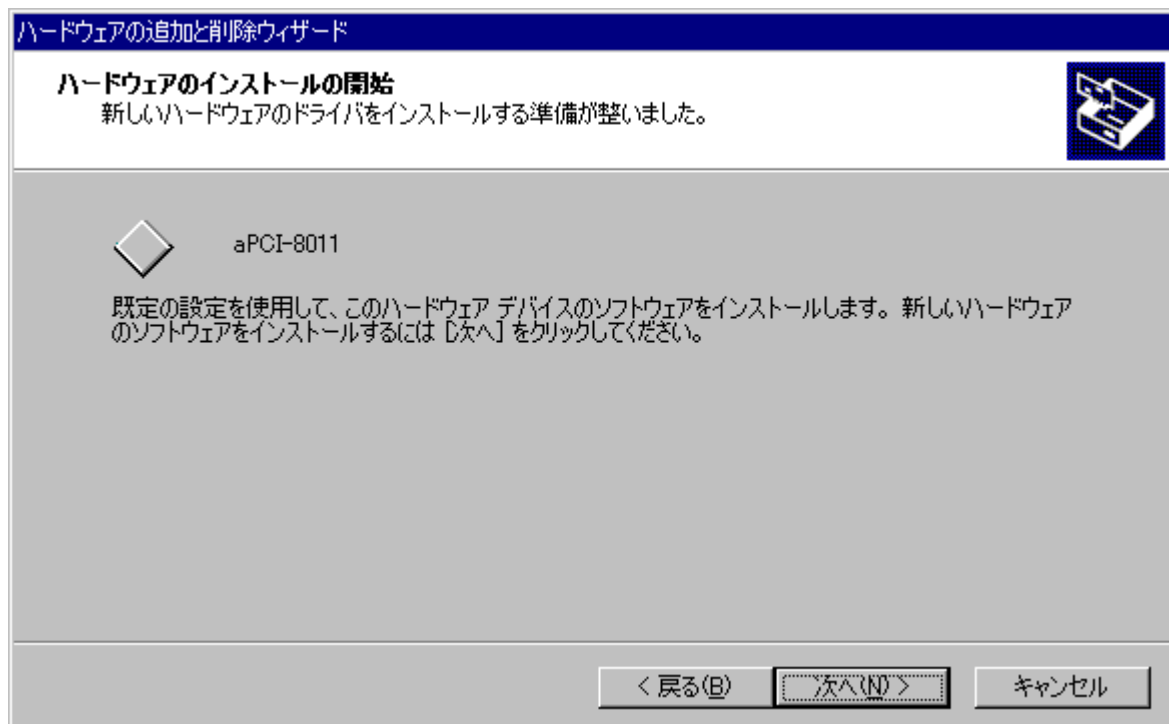
9. [OK]ボタンをクリックします。



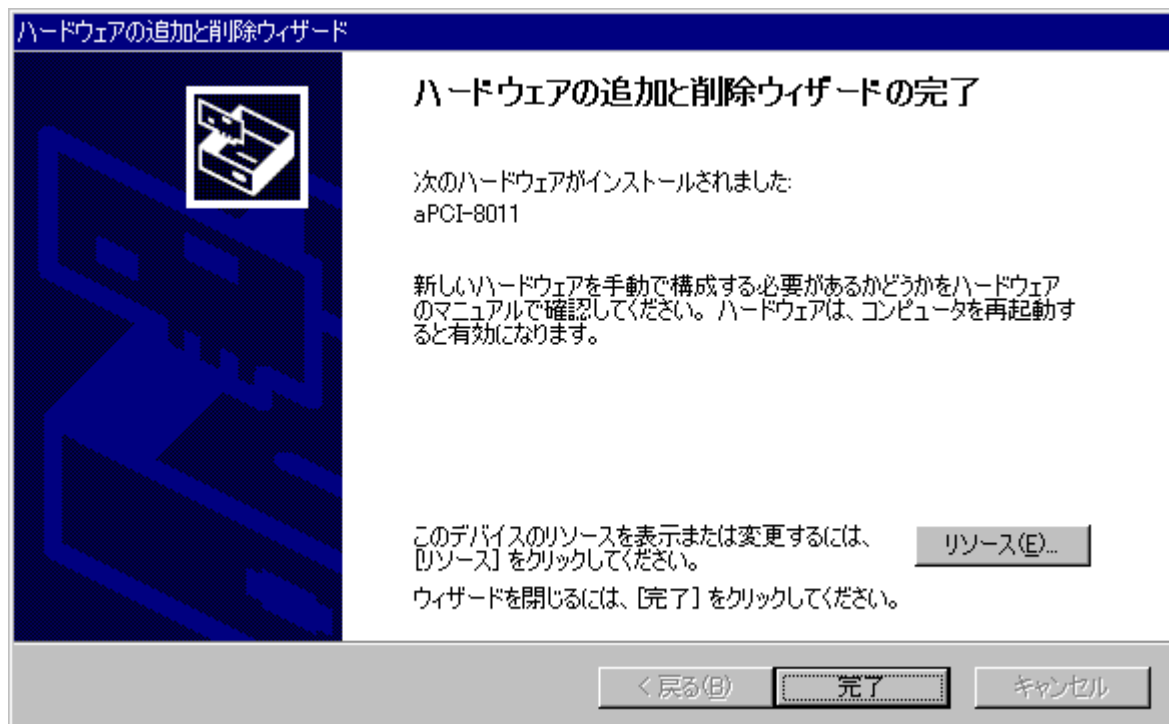
10. 他のデバイスとリソースが競合しないか確認し、[OK]ボタンをクリックします。



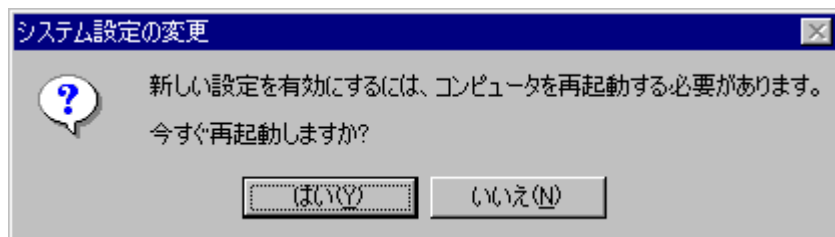
11. [次へ]ボタンをクリックします。



12. [完了]ボタンをクリックします。



13. 再起動を促すメッセージが表示されますので、Windows2000 を再起動します。



ドライバを制御する DLL ファイルは自動的にコピーされませんので、お使いの環境に合わせて、ファイルをコピーしてください。

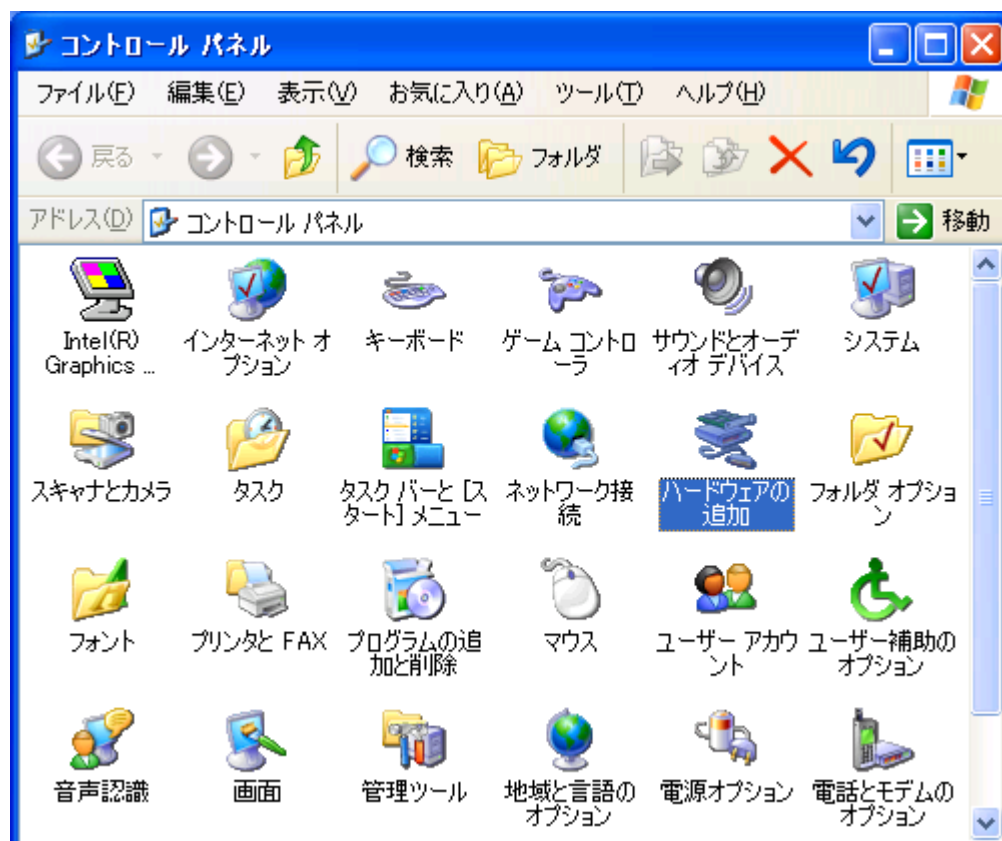
一般的には「アプリケーションと同じディレクトリ」か、Windows2000 では「%WinDir%\System32 (例 C:\Winnt\System32)」となります。

4 - 1 - 2 . WindowsXP へのインストール

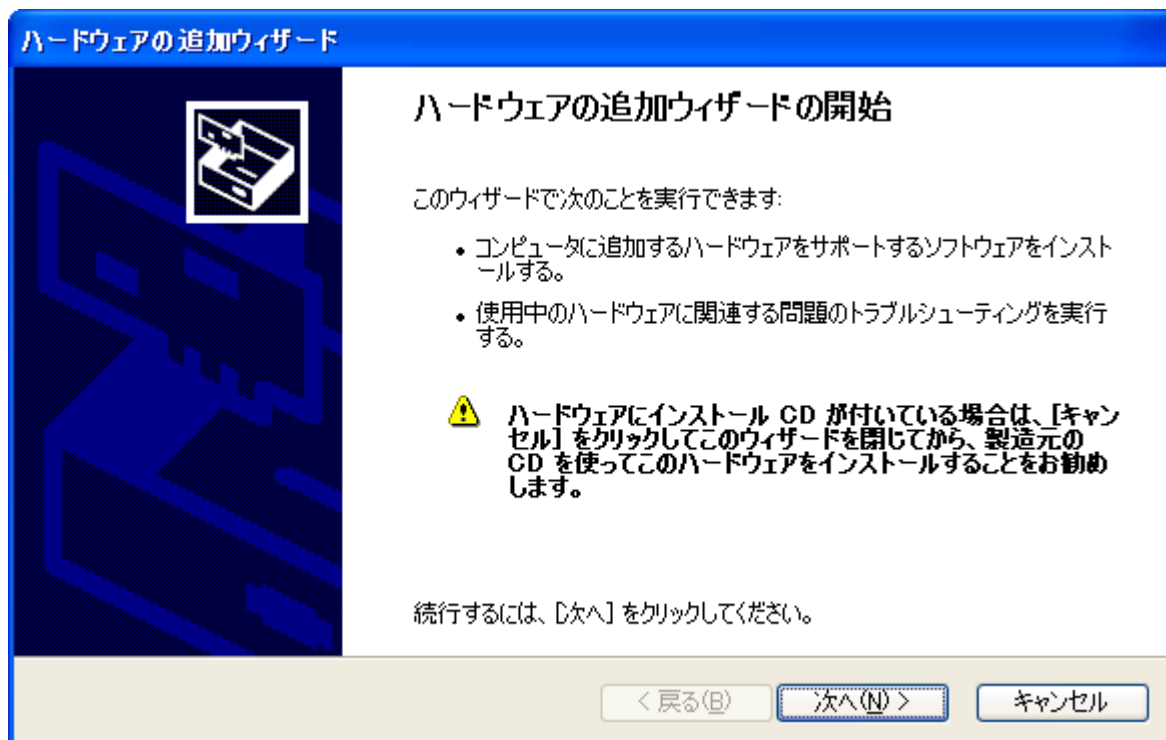
本製品を WindowsXP 上で使用するためには、ハードウェア・ウィザードによるインストールが必要です。

以下の手順に従って登録を行ってください。

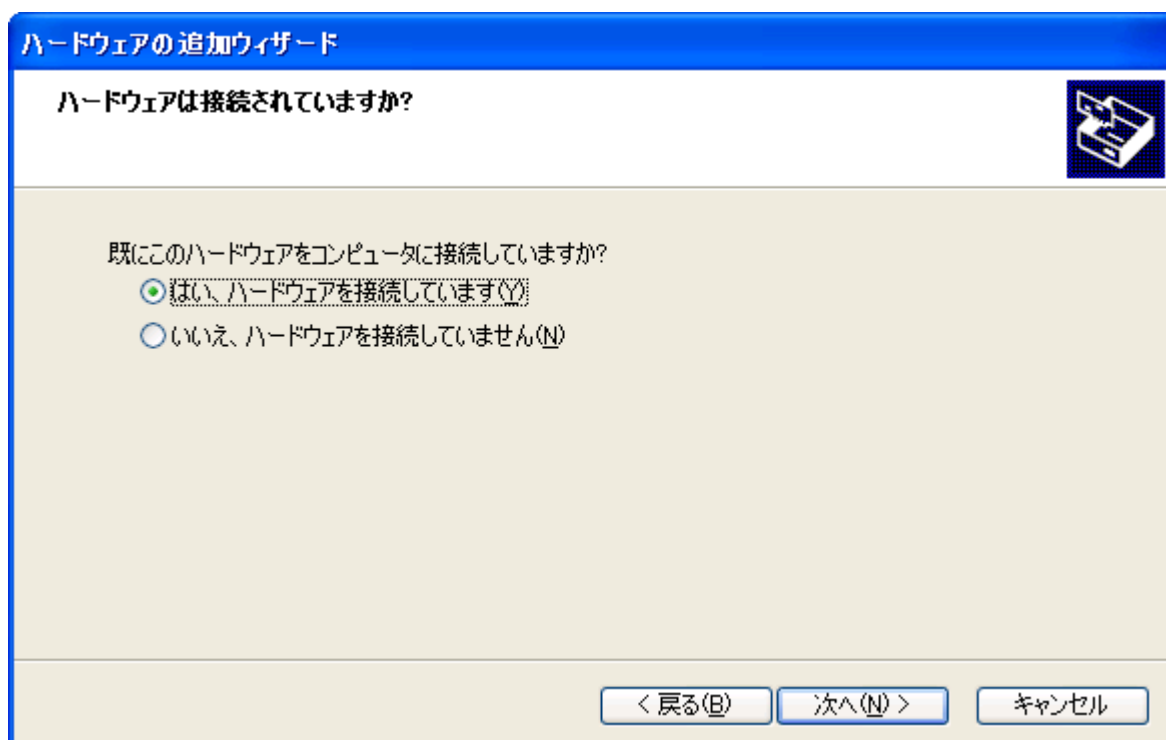
1. 「Administrator」でログオンし、[スタート - コントロールパネル]を実行してコントロールパネルを開いてください。開いたら[ハードウェアの追加]アイコンをダブルクリックします。



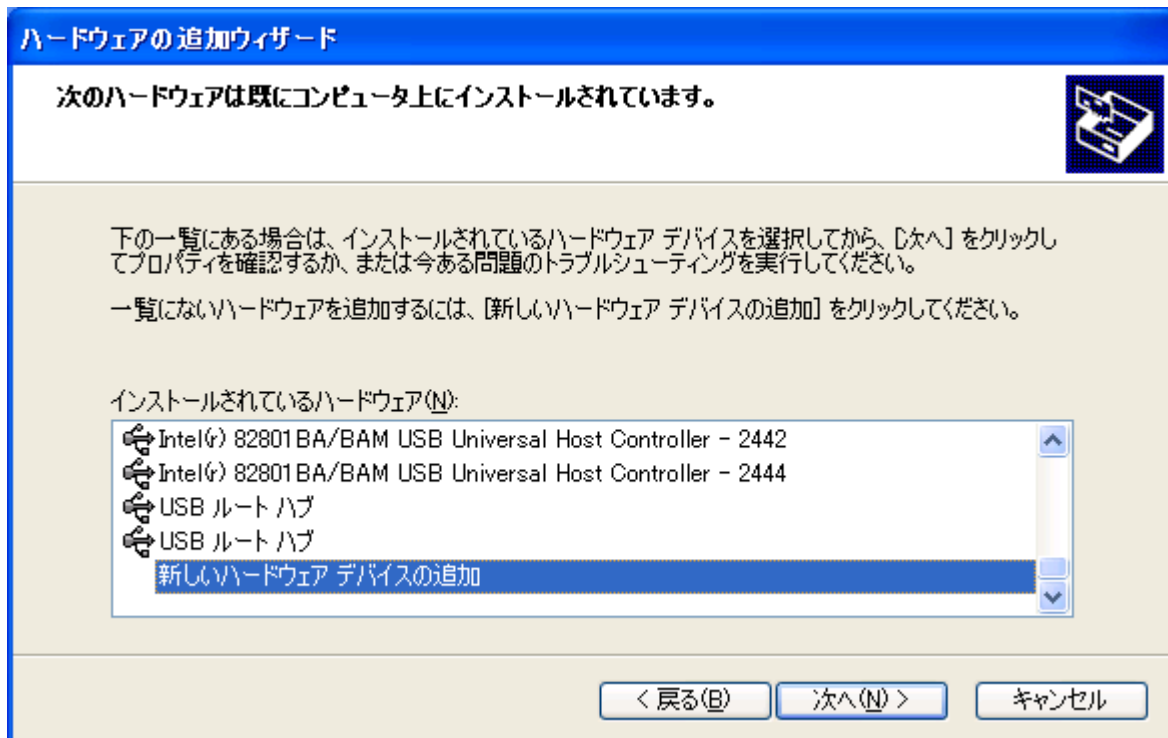
2. [次へ]ボタンをクリックします。



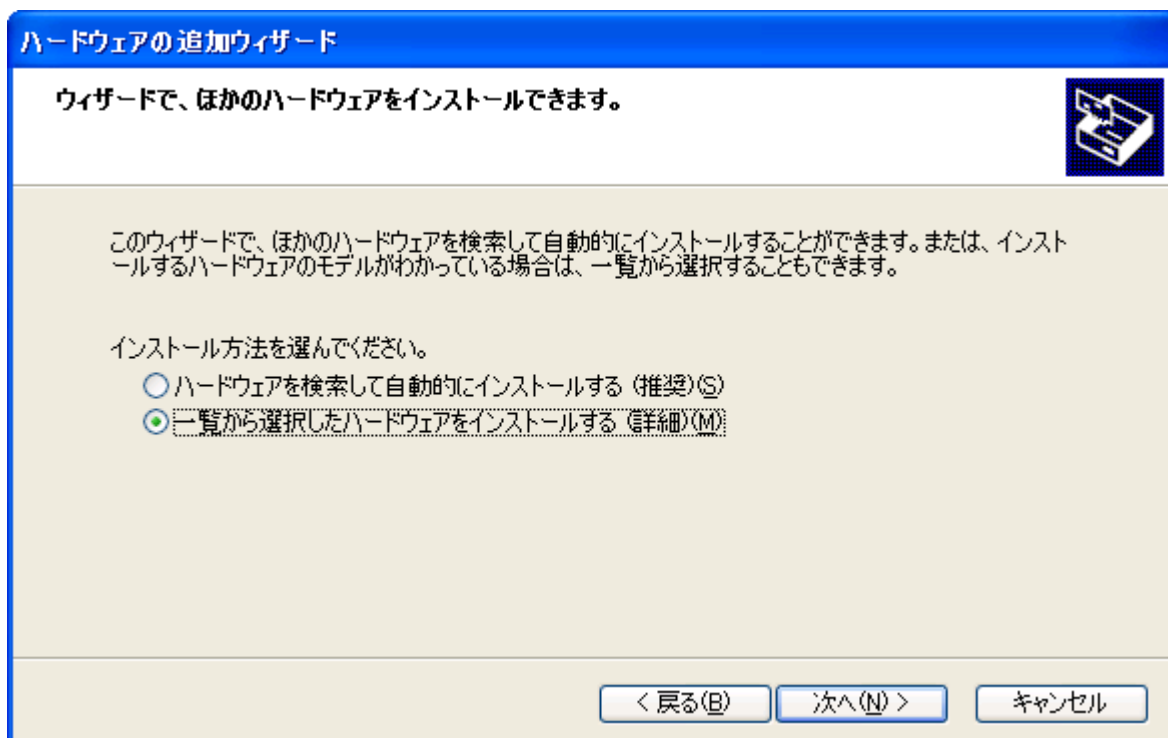
3. [はい、ハードウェアを接続しています]を選択し、[次へ]ボタンをクリックします。



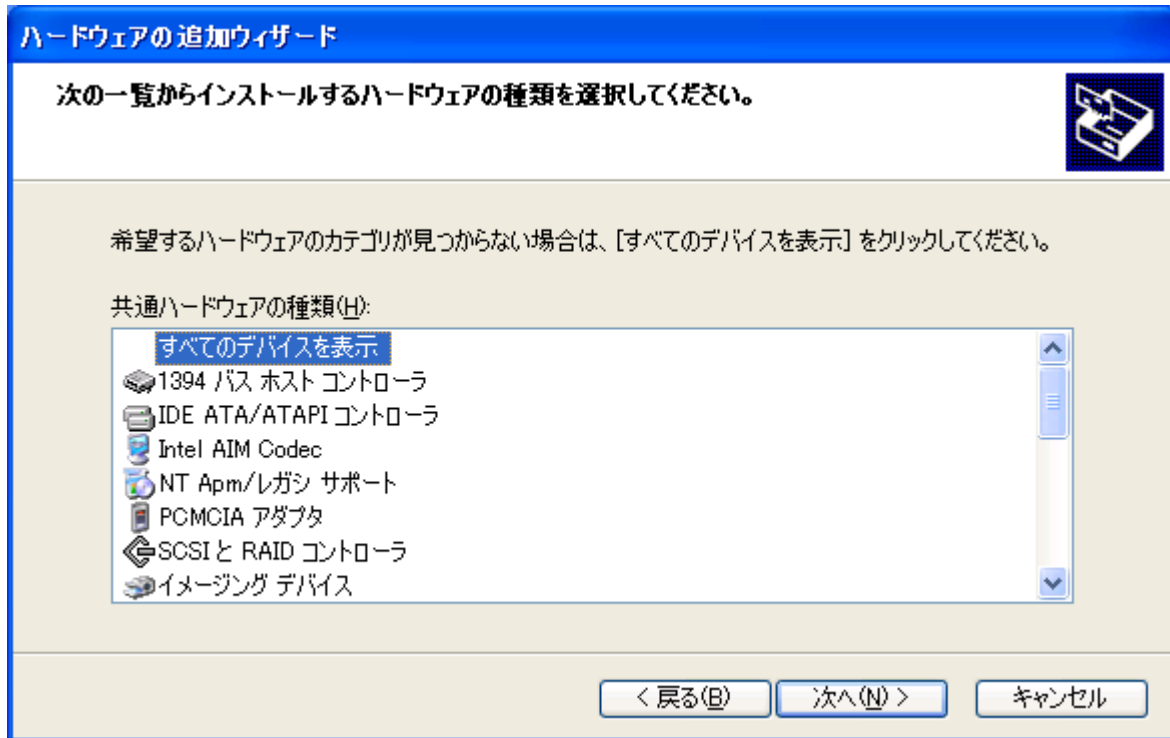
4. [新しいハードウェアデバイスの追加]を選択し、[次へ]ボタンをクリックします。



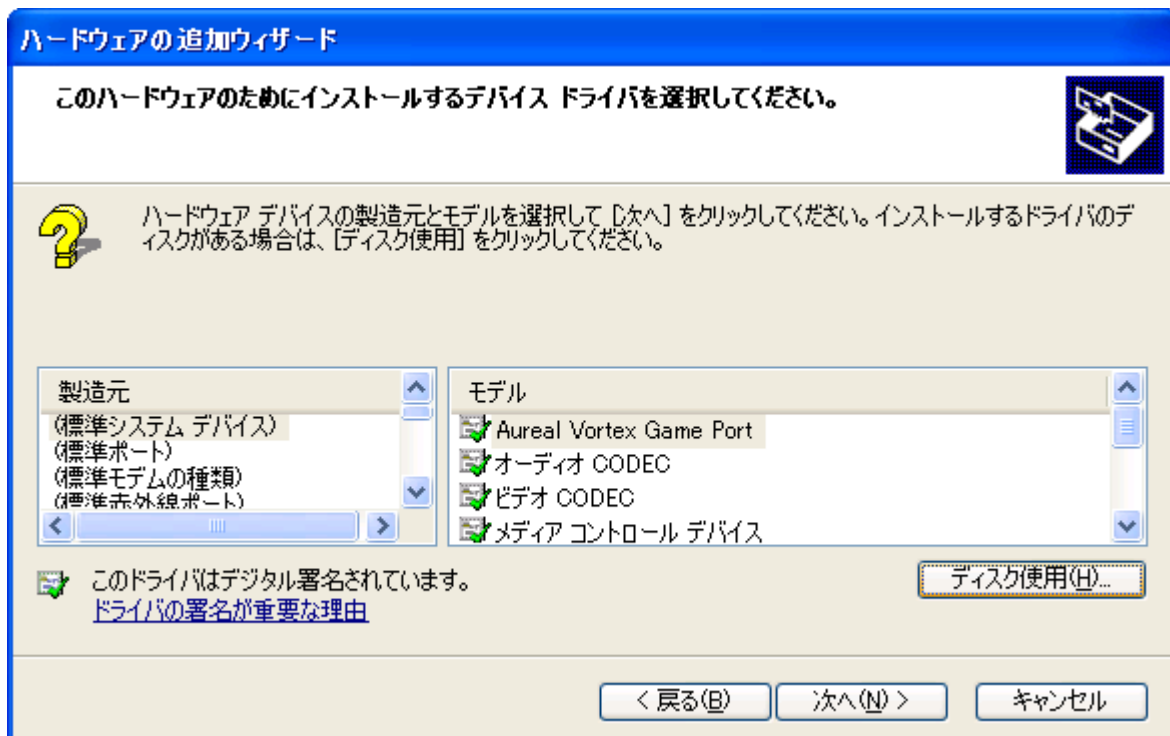
5. [一覧から選択したハードウェアをインストールする]を選択し、[次へ]ボタンをクリックします。



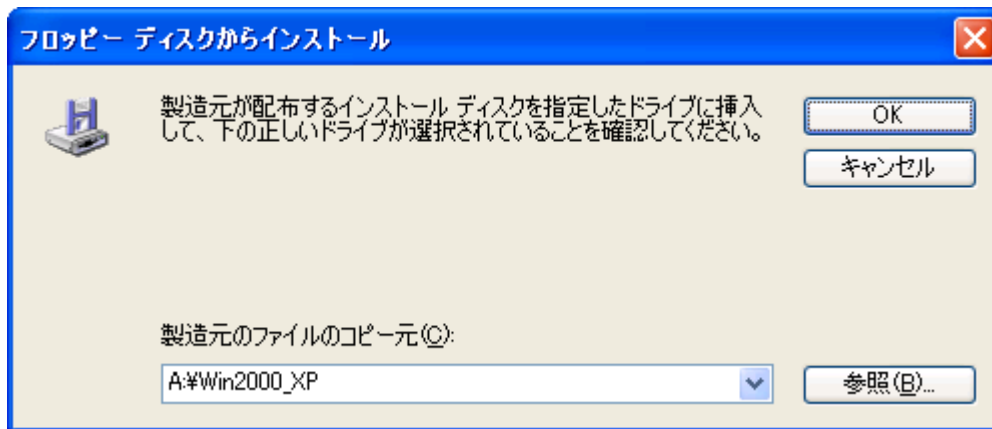
6. [すべてのデバイスを表示]を選択し、[次へ]ボタンをクリックします。



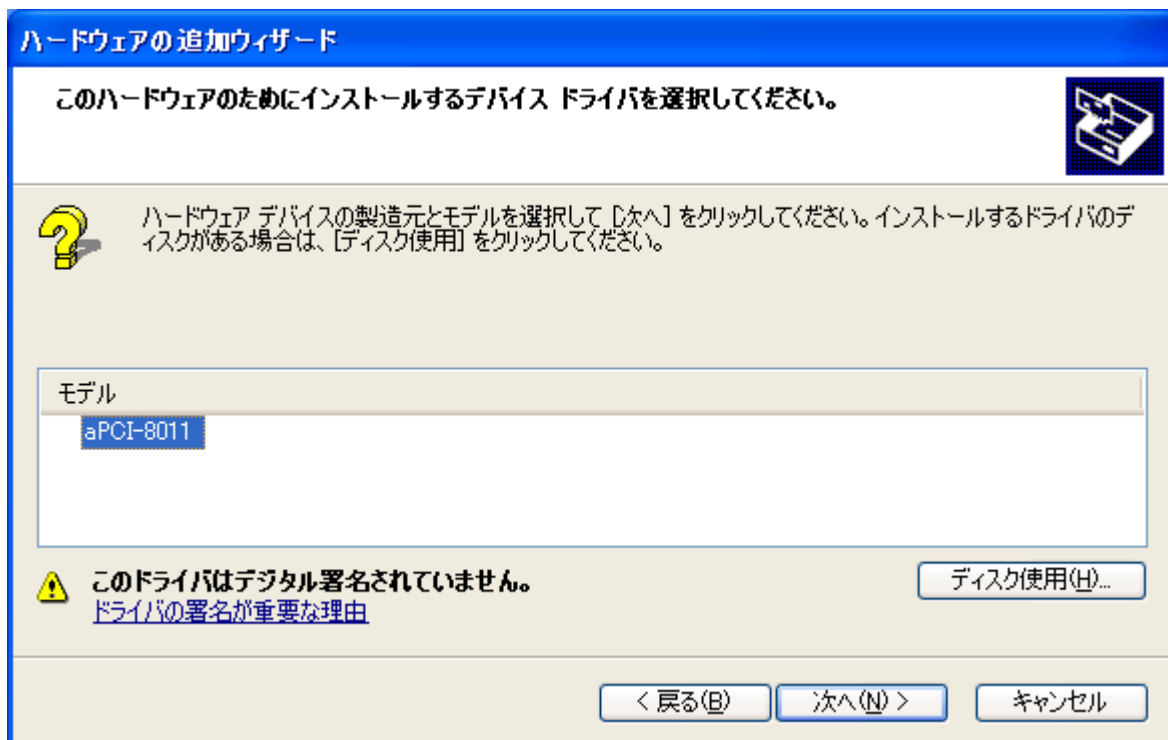
7. [ディスク使用]ボタンをクリックします。



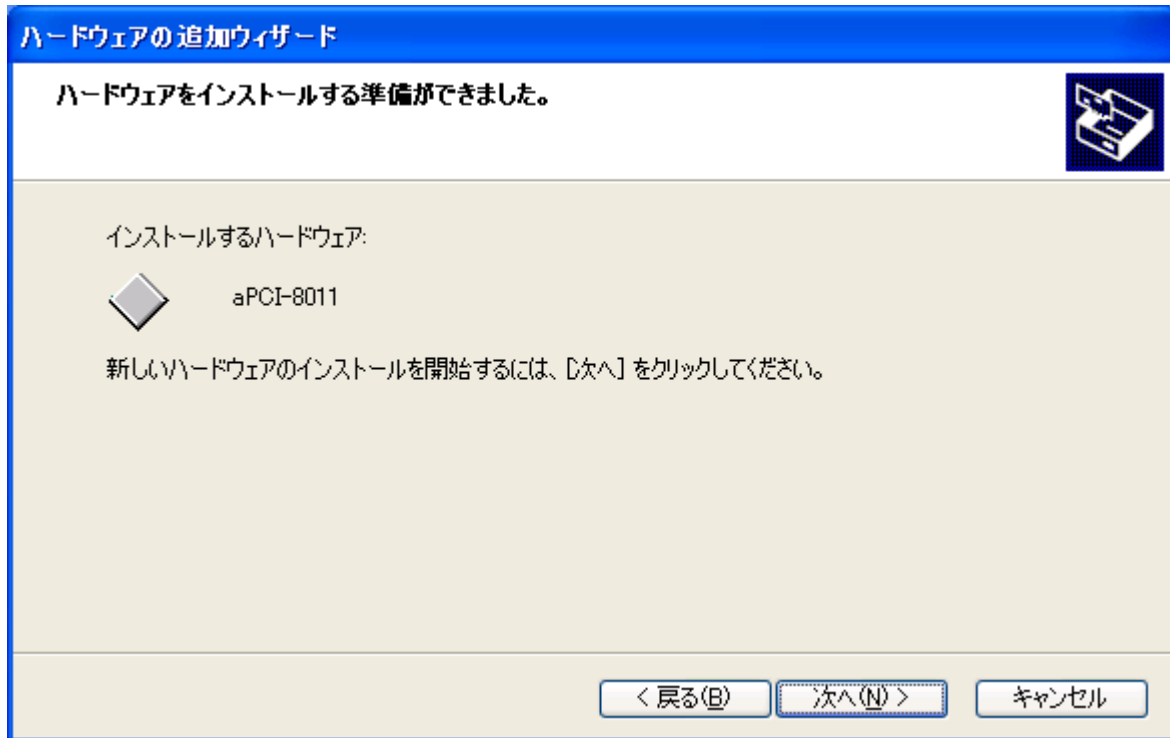
コピー元にドライバソフトの Win2000_XP ディレクトリを指定し、[OK]ボタンをクリックします。
ここでは「A:¥Win2000_XP」を指定しています。



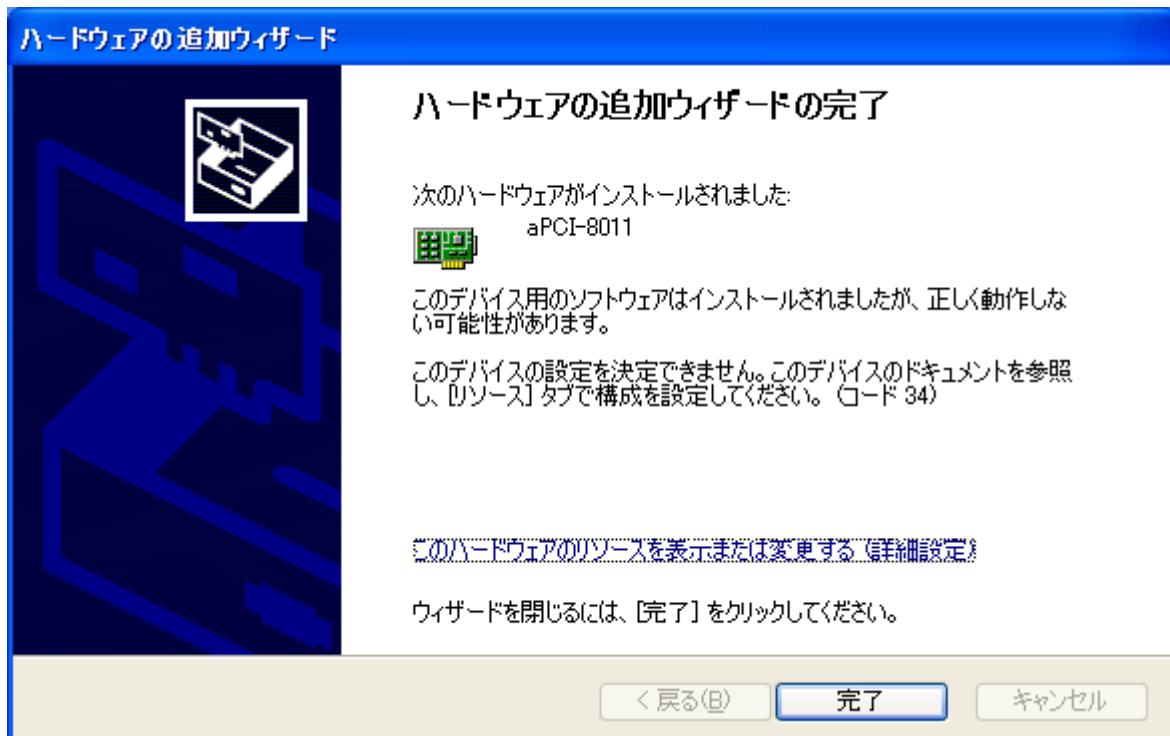
8. 使用するデバイスを選択し、[次へ]ボタンをクリックします。



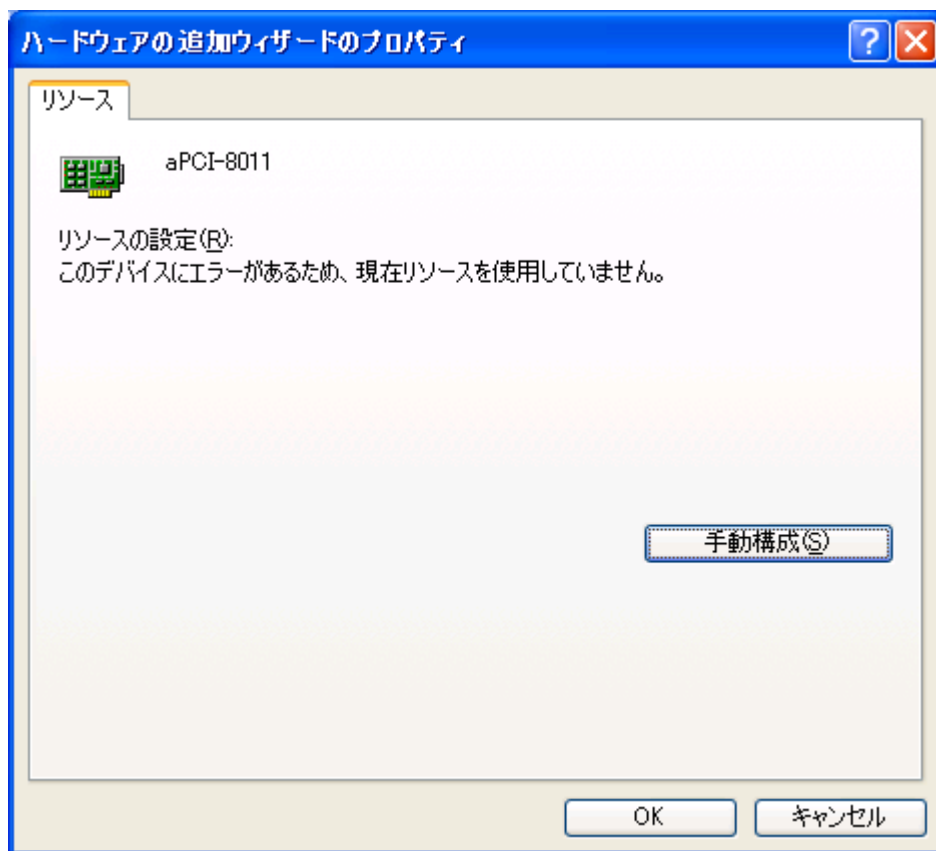
9. [次へ]ボタンをクリックします。



10. [このハードウェアのリソースを表示または変更する]をクリックします。



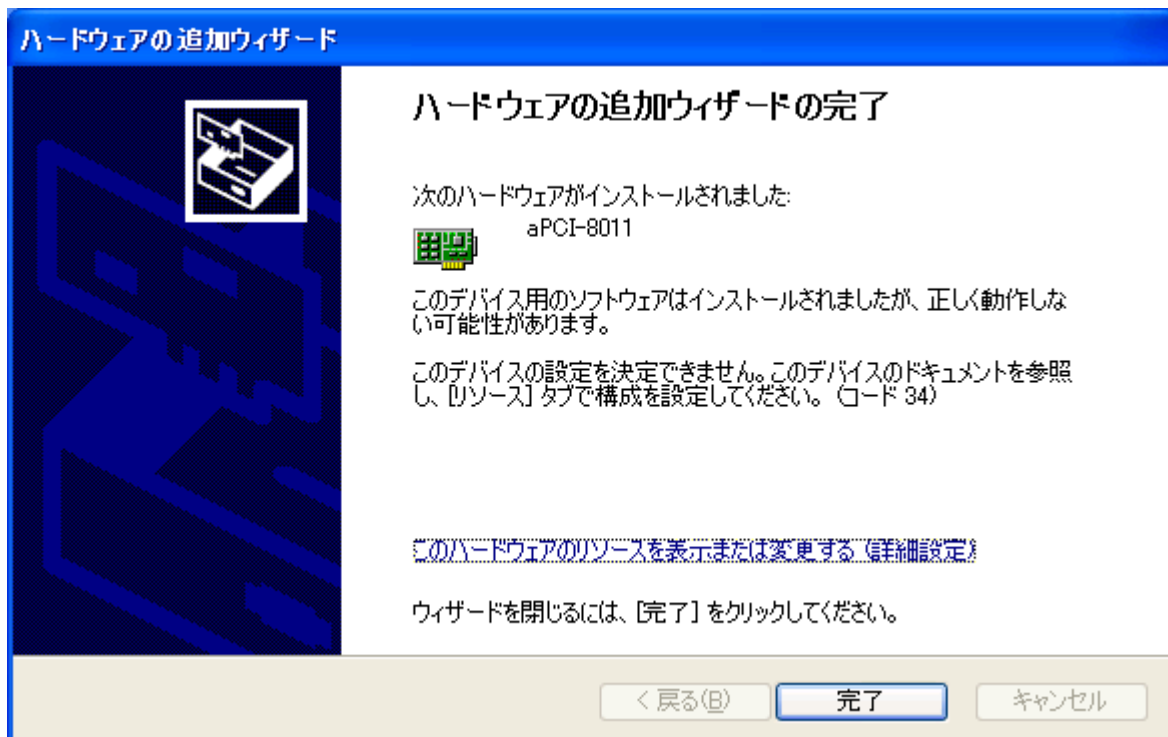
11. [手動構成]ボタンをクリックします。



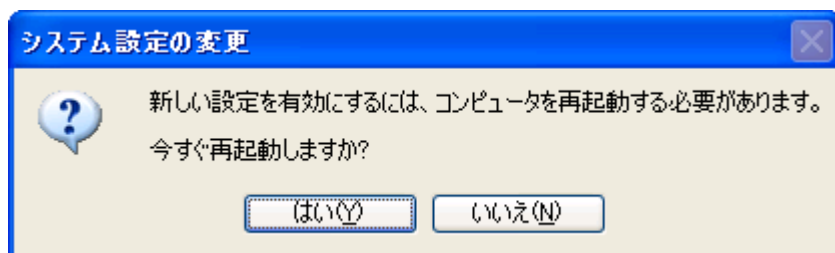
12. 他のデバイスとリソースが競合しないか確認し、[OK]ボタンをクリックします。



13. [完了]ボタンをクリックします。



14. 再起動を促すメッセージが表示されますので、WindowsXP をシャットダウンします。



ドライバを制御する DLL ファイルは自動的にコピーされませんので、お使いの環境に合わせて、ファイルをコピーしてください。

一般的には「アプリケーションと同じディレクトリ」か、WindowsXP では「%WinDir%\System32 (例 C:\Winnt\System32)」となります。

4 - 2 . アンインストール

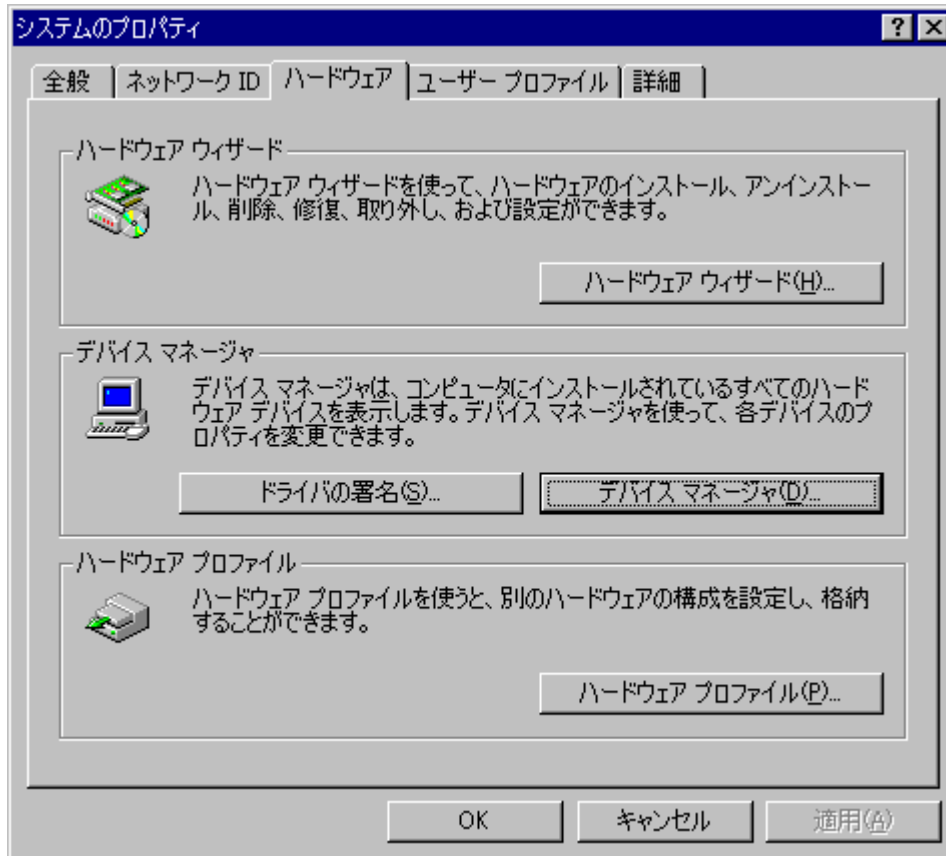
4 - 2 - 1 . Windows2000 からのアンインストール

インストールによって確保されたリソースを以下の手順で解放し、ドライバをアンインストールすることができます。

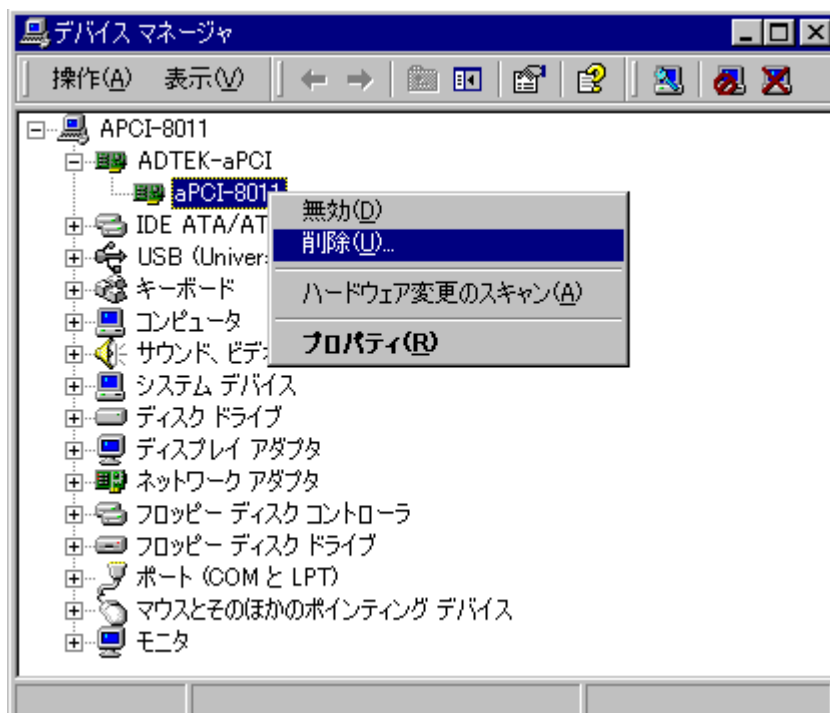
1. [スタート - 設定 - コントロールパネル]を実行してコントロールパネルを開いてください。開いたら[システム]アイコンをダブルクリックします。



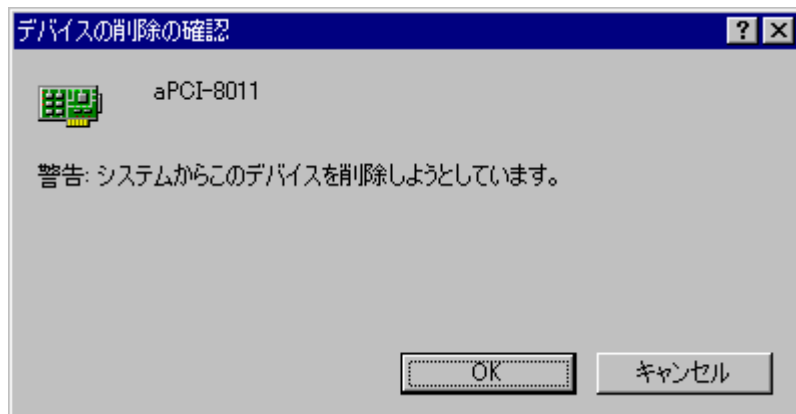
2. [ハードウェア]タブの[デバイスマネージャ]ボタンをクリックします。



3. [aPCI-8011]上で右クリックし、[削除]を選択します。



4. [OK]ボタンをクリックします。

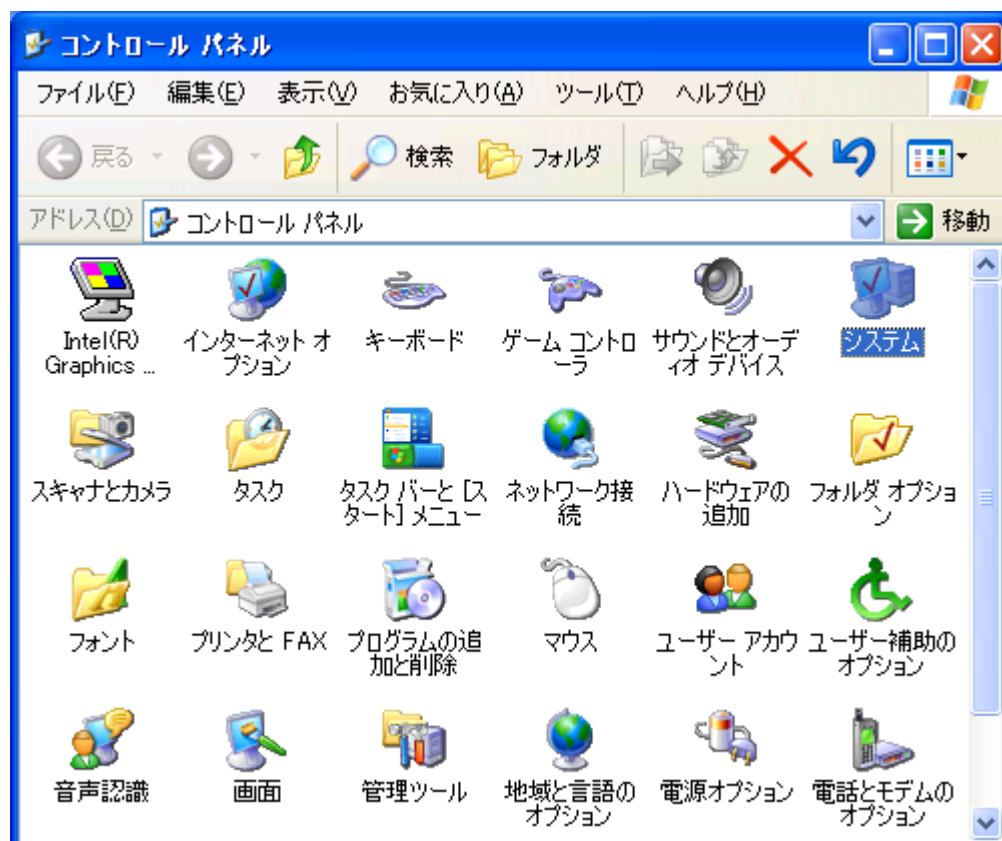


5. Windows2000 を終了します。

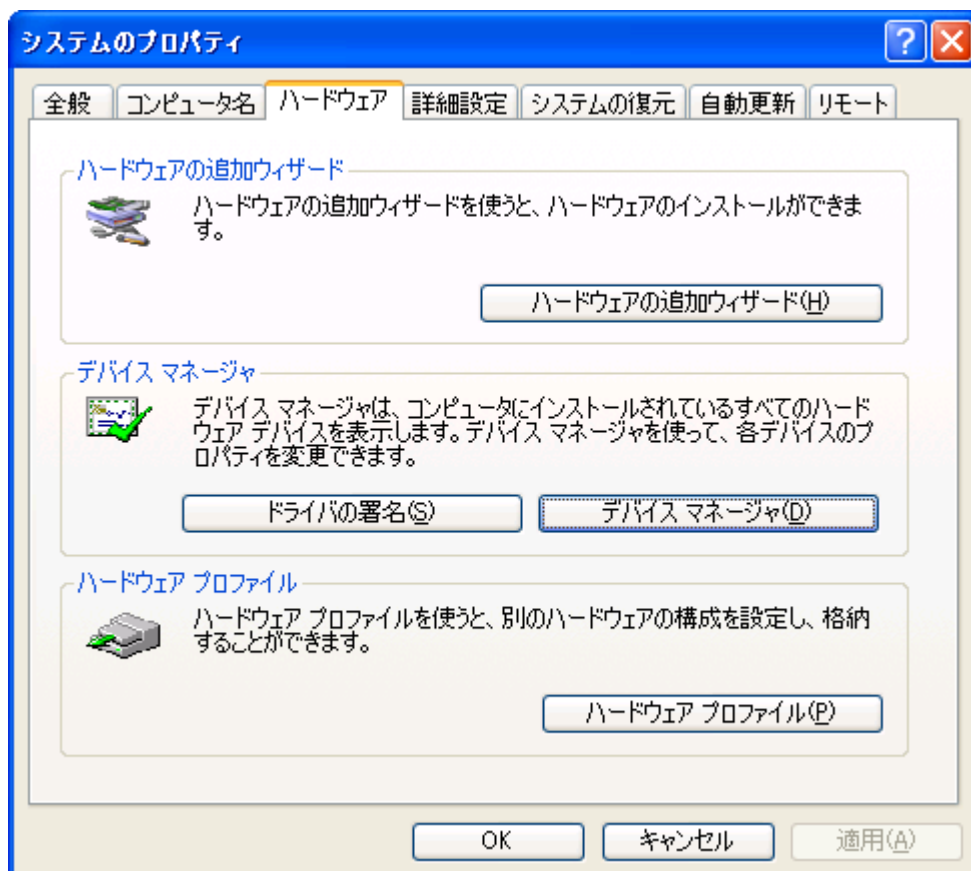
4 - 2 - 2 . WindowsXP からのアンインストール

インストールによって確保されたリソースを以下の手順で解放し、ドライバをアンインストールすることができます。

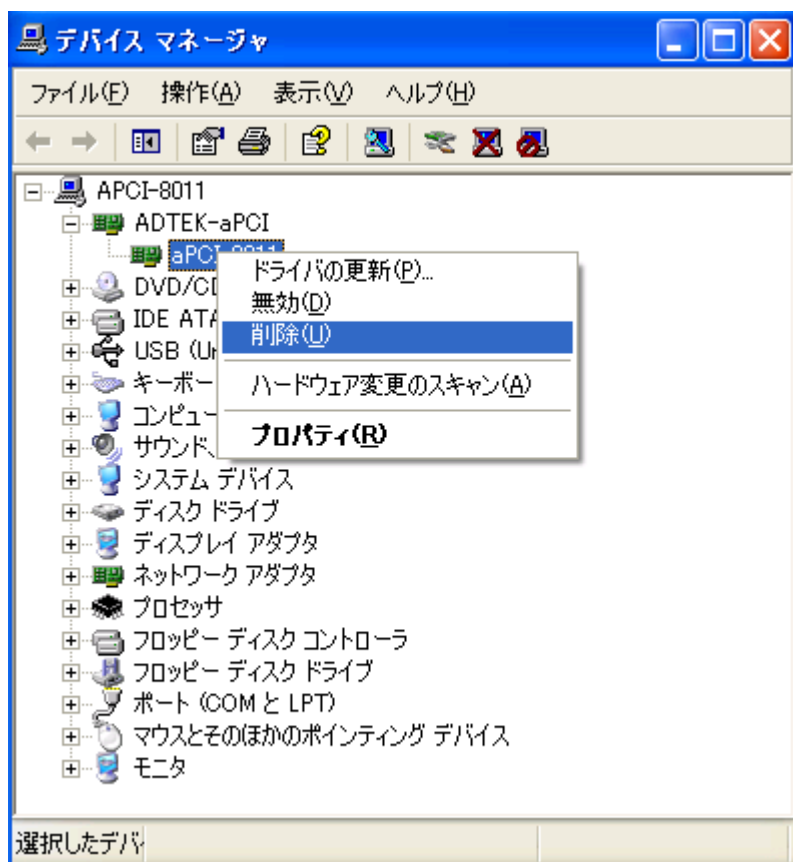
1. [スタート - コントロールパネル]を実行してコントロールパネルを開いてください。開いたら[システム]アイコンをダブルクリックします。



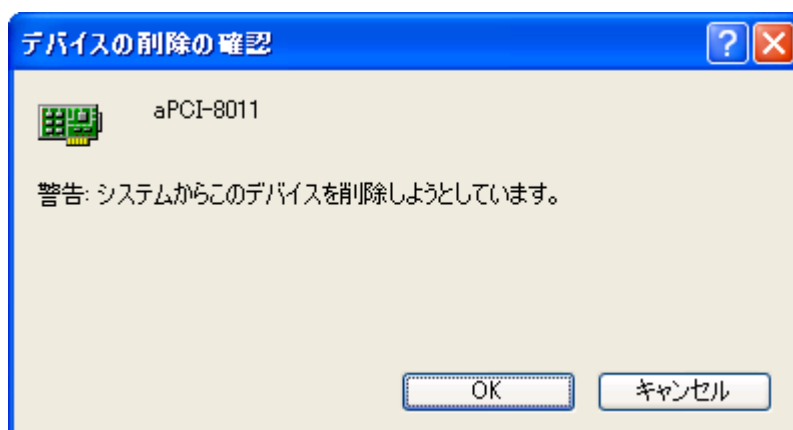
2. [ハードウェア]タブの[デバイスマネージャ]ボタンをクリックします。



3. [aPCI-8011]上で右クリックし、[削除]を選択します。



4. [OK]ボタンをクリックします。



5. WindowsXP を終了します。

5 . サンプルソース

サンプルソースには、Visual C++ (Ver.4/5/6) 版、Visual Basic (Ver.4(32bit)/5/6) 版、Delphi (Ver.2/3/4/5/6/7) 版がございます。

DLL 内の関数のインターフェースは、各関数のヘッダ、注釈および「6 . API 仕様」を参照してください。

<ラッパー関数>

apci8011w.* のように「w」が付いているファイルには、DLL 内の関数を簡単にコールするためのラッパー(Wrapper)関数が定義されています(DLL のロード/アンロード関数も含まれています)。デバイスの制御を開始する前には DLL ロード関数を、デバイスの制御を終了する際には DLL アンロード関数をコールしてください。

サンプルソースは各開発環境にて実行してお試しいただけます。
詳しくは各ディレクトリ内の buildxx.txt の例をご覧ください。

サンプルソースのご利用については、各開発環境および OS ・言語に対する十分な理解を前提としております。よって、これらそのものの使用方法に関するお問い合わせには一切お答えいたしかねますので、あらかじめご了承ください。

6 . API 仕様

6 - 1 . 概要

本章では、aPCI-8011 用ユーザ公開 API を定義します。
デバイスドライバの詳細については触れていません。

6 - 2 . プログラム構成

本章で定義される API は、apci8011.sys (デバイスドライバ) および apci8011.dll (ダイナミックリンクライブラリ) により実現されます。

6 - 3 . API リファレンス

6 - 3 - 1 . Apci8011Create

機 能 デバイスの使用を宣言

形 式 Visual C++

```
BOOL Apci8011Create(void);
```

形 式 Visual Basic

```
Function Apci8011Create() As Long
```

形 式 Delphi

```
function Apci8011Create(): BOOL;
```

戻り値 API が正常終了したか、失敗したかを返します。

FALSE 失敗。

TRUE 正常終了。

解 説 アプリケーションが使用することをドライバに通知します。

他の API を呼び出す前に必ずこの API を呼び出してください。

6 - 3 - 2 . Apci8011Close

機 能 デバイスの開放

形 式 Visual C++

```
BOOL Apci8011Close( void);
```

形 式 Visual Basic

```
Function Apci8011Close() As Long
```

形 式 Delphi

```
function Apci8011Close(): BOOL;
```

戻り値 API が正常終了したか、失敗したかを返します。

FALSE 失敗。

TRUE 正常終了。

解 説 デバイスを他のアプリケーションに開放することをドライバに通知します。
アプリケーションを終了する前に必ずこの API を呼び出してください。

6 - 3 - 3 . Apci8011GetVersion

機 能 バージョン情報取得

形 式 Visual C++
 BOOL Apci8011GetVersion
 (
 PDWORD pdwDllVer,
 PDWORD pdwDrvVer
);

形 式 Visual Basic
 Function Apci8011GetVersion
 (
 pdwDllVer As Long,
 pdwDrvVer As Long
) As Long

形 式 Delphi
 function Apci8011GetVersion
 (
 var pdwDllVer: DWORD;
 var pdwDrvVer: DWORD
): Boolean;

入 力 pdwDllVer
 DLL のバージョン情報を格納する領域へのポインタ。
 NULL 可。

pdwDrvVer
 DRIVER のバージョン情報を格納する領域へのポインタ。
 NULL 可。

出 力 *pdwDllVer
 DLL のバージョン情報。

*pdwDrvVer
 DRIVER のバージョン情報。

戻り値 API が正常終了したか、失敗したかを返します。
 FALSE 失敗。
 TRUE 正常終了。

解 説 DLL と DRIVER のバージョン情報を取得します。
 それぞれのバージョンは 4 桁で表現されます。
 初回リリースは 0.1.0.0 とし、0x00010000 となります。
 最大値は 255.255.255.255 となり、0xffffffff となります。

6 - 3 - 4 . Apci8011EnableWatchdog

機 能 ウォッチドックタイマを動作するようにします

形 式 Visual C++
BOOL Apci8011EnableWatchdog
(
 DWORD dwTime
);

形 式 Visual Basic
Function Apci8011EnableWatchdog
(
 ByVal dwTime as Long
) As Long

形 式 Delphi
function Apci8011EnableWatchdog
(
 dwTime: DWORD
): BOOL;

入 力 dwTime
 タイムアウト値を設定します。

0 : 1 秒に設定
1 : 2 秒に設定
2 : 4 秒に設定
3 : 8 秒に設定
4 : 16 秒に設定
5 : 32 秒に設定
6 : 64 秒に設定
7 : 128 秒に設定

戻り値 API が正常終了したか、失敗したかを返します。
FALSE 失敗。
TRUE 正常終了。

解 説 デバイスのウォッチドックタイマを動作するようにします。

6 - 3 - 5 . Apci8011RefreshWatchdog

機 能 ウォッチドックタイマを再設定します

形 式 Visual C++
 BOOL Apci8011RefreshWatchdog(void)

形 式 Visual Basic
 Function Apci8011RefreshWatchdog() As Long

形 式 Delphi
 function Apci8011RefreshWatchdog(): BOOL;

戻り値 API が正常終了したか、失敗したかを返します。
 FALSE 失敗。
 TRUE 正常終了。

解 説 デバイスのウォッチドックタイマを再設定します。

6 - 3 - 6 . Apci8011DisableWatchdog

機 能 サンプリングを開始します。

形 式 Visual C++
 BOOL Apci8011DisableWatchdog(void);

形 式 Visual Basic
 Function Apci8011DisableWatchdog() As Long

形 式 Delphi
 function Apci8011DisableWatchdog(): BOOL;

戻り値 API が正常終了したか、失敗したかを返します。
 FALSE 失敗。
 TRUE 正常終了。

解 説 デバイスのウォッチドックタイマを動作しないようにします。

6 - 3 - 7 . Apci8011GetLastError

機 能 エラーコード取得

形 式 Visual C++

```
DWORD Apci8011GetLastError(void);
```

形 式 Visual Basic

```
Function Apci8011GetLastError() As Long
```

形 式 Delphi

```
function Apci8011GetLastError(): DWORD;
```

戻り値 エラーコード

このデバイスのもっとも最近起こったエラーのコードを取得します。
デバイスに依存しないエラーを取得します。

6 - 4 . 定義

6 - 4 - 1 . エラーコード

#define APCI8011_SUCCESS	0	// 正常終了
#define APCI8011_ERR_NO_DEVICE	1	// No device or not an aPCI-8011
#define APCI8011_ERR_RESOURCE	2	// Resource exhaustion
#define APCI8011_ERR_INVALID_ARG	3	// Invalid argument(s)
#define APCI8011_ERR_SLOT	4	// 論理スロット番号が不正です (または見つかりません)
#define APCI8011_ERR_TIME	5	//ウォッチドックタイマのタイムアウト値が 不正です
#define APCI8011_ERR_NOT_CREATED	6	// デバイスがクリエイトされていません
#define APCI8011_ERR_INVALID_FUNC	7	//無効な関数呼び出しです
#define APCI8011_DISABLE_WDT	8	// ウォッチドックタイマが無効な状態です (無効な関数呼び出しです)
#define APCI8011_ERR_INVALID_SLOT	9	// Invalid slot(s)
#define APCI8011_ERR_IN_USE	10	// デバイス使用中です(NT)
#define APCI8011_ERR_INTERNAL	11	// システムエラー
#define APCI8011_ERR_SYSTEM	12	// システムエラー

APCI8011_SUCCESS は、初期時に初期化される値です。

以降エラーが起こった場合はエラー内容が更新されるまでそのエラー値を保持します。

製品のお問い合わせについて

お買い求めいただいた製品に対する次のようなお問い合わせは、お求めの販売店又は株式会社アドテックシステムサイエンスの各営業所にご連絡ください。

- ・ お求めの製品にご不審な点や万一欠品があったとき
- ・ 製品の修理
- ・ 製品の補充品や関連商品について
- ・ 本製品を使用した特注製品についてのご相談

技術サポート 技術的な内容のお問い合わせは、「ファックス」「郵送」「E-mail」のいずれかにて、下記までお問い合わせください。また、お問い合わせの際は、内容をできるだけ詳しく具体的にお書きくださるようお願いいたします。

技術的な内容のお問い合わせ先

株式会社 アドテック システム サイエンス テクニカルサポート
〒240-0005

神奈川県横浜市保土ヶ谷区神戸町 134 YBP ウエストタワー 8F

E-mail support@adtek.co.jp

Fax 045-331-7770

改訂履歴

発行年月日 2004年06月22日 第1版

発行年月日 2004年11月04日 改訂第2版
・所在地変更

aPCI-8011
ソフトウェアマニュアル

第2版発行 2004年11月04日
発行所 株式会社 アドテック システム サイエンス
〒240-0005 神奈川県横浜市保土ヶ谷区神戸町134
YBP ウエストタワー 8F
Tel 045-331-7575 (代) Fax 045-331-7770

不許複製

CPCI-069-041104
© 2004 ADTEK SYSTEM SCIENCE Co.,Ltd.