

Anywire A40/20 シリーズ PCI インターフェース用  
Windows7(32bit) ドライバライブラリ  
使用説明書

2012/02/27

株式会社エニワイヤ

## 1. はじめに

本ソフトウェアは、AnyWire DB A40 シリーズの PCI インターフェースボードを Windows7(32bit) で使用するためのデバイスドライバです。

適応 OS	Windows7(32bit)
最大ボード数	16 枚
対象ボード	AP48-01 AP48-02

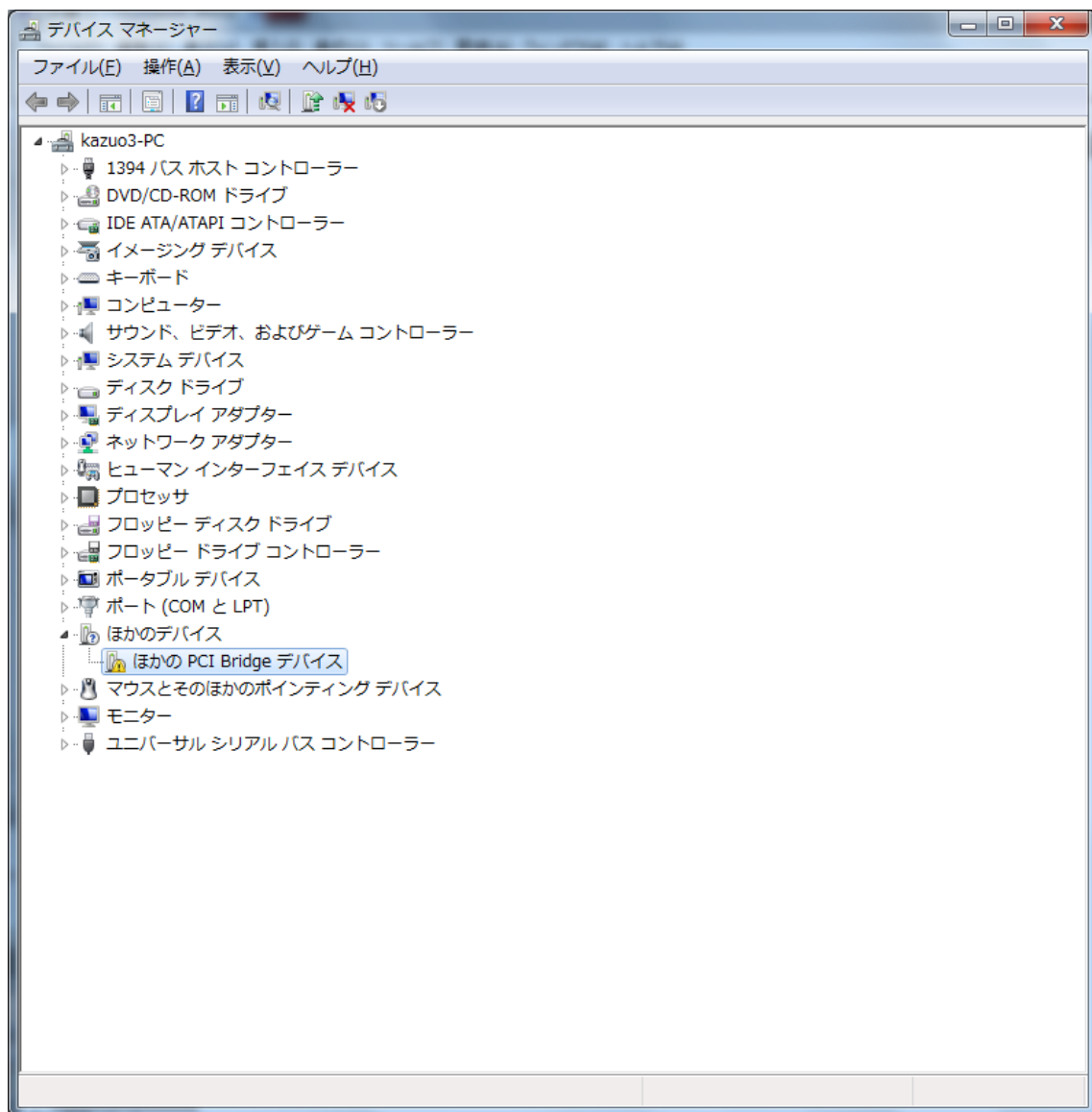
本ソフトウェアには以下のファイルが納められています。

AW_DB_A40_PCI_win7x86.sys	Window7(32bit) デバイスドライバ
AW_DB_A40_PCI_win7x86.inf	セットアップ情報ファイル
AW_DB_A40_PCI.dll	ドライバライブラリ DLL
AW_DB_A40_PCI.lib	ドライバライブラリ・リンク用ライブラリ
AW_Function.h	ドライバライブラリ ヘッダファイル
TestApp.exe	ドライバライブラリ使用 テストアプリケーション

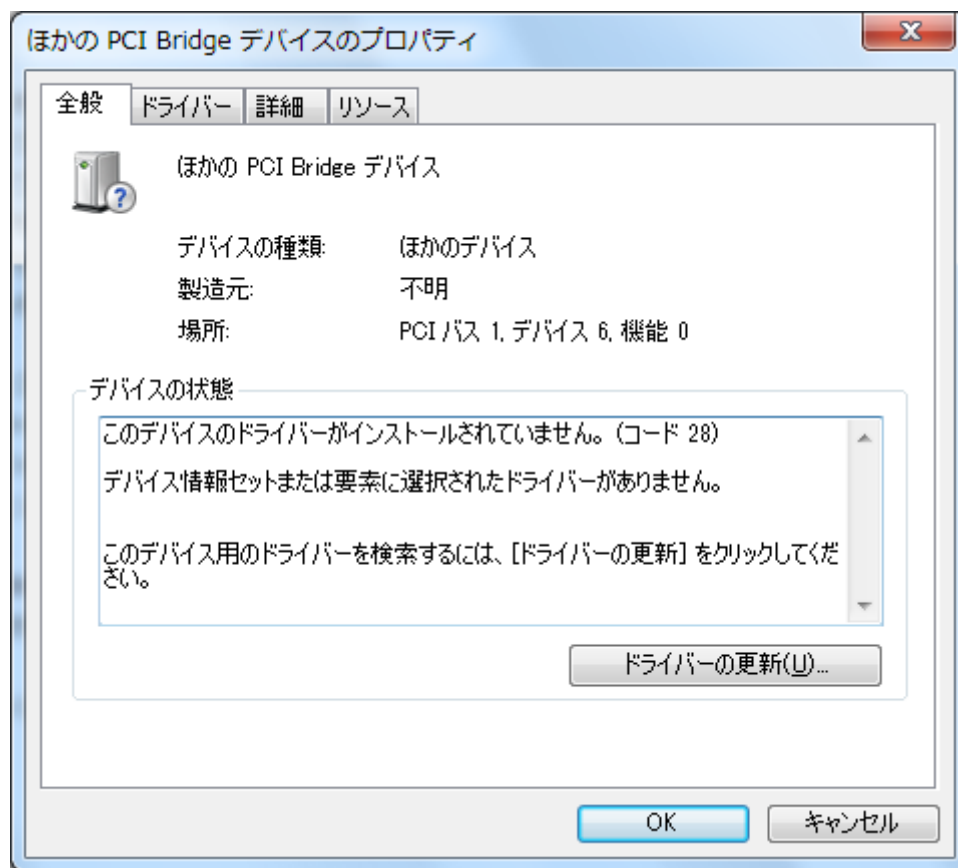
## 2. ドライバのインストール

本ボードをパソコンの P C I バスに挿入して、電源を入れると plug & play 機能が働き、ハードウェア追加ウィザードが起動されます。

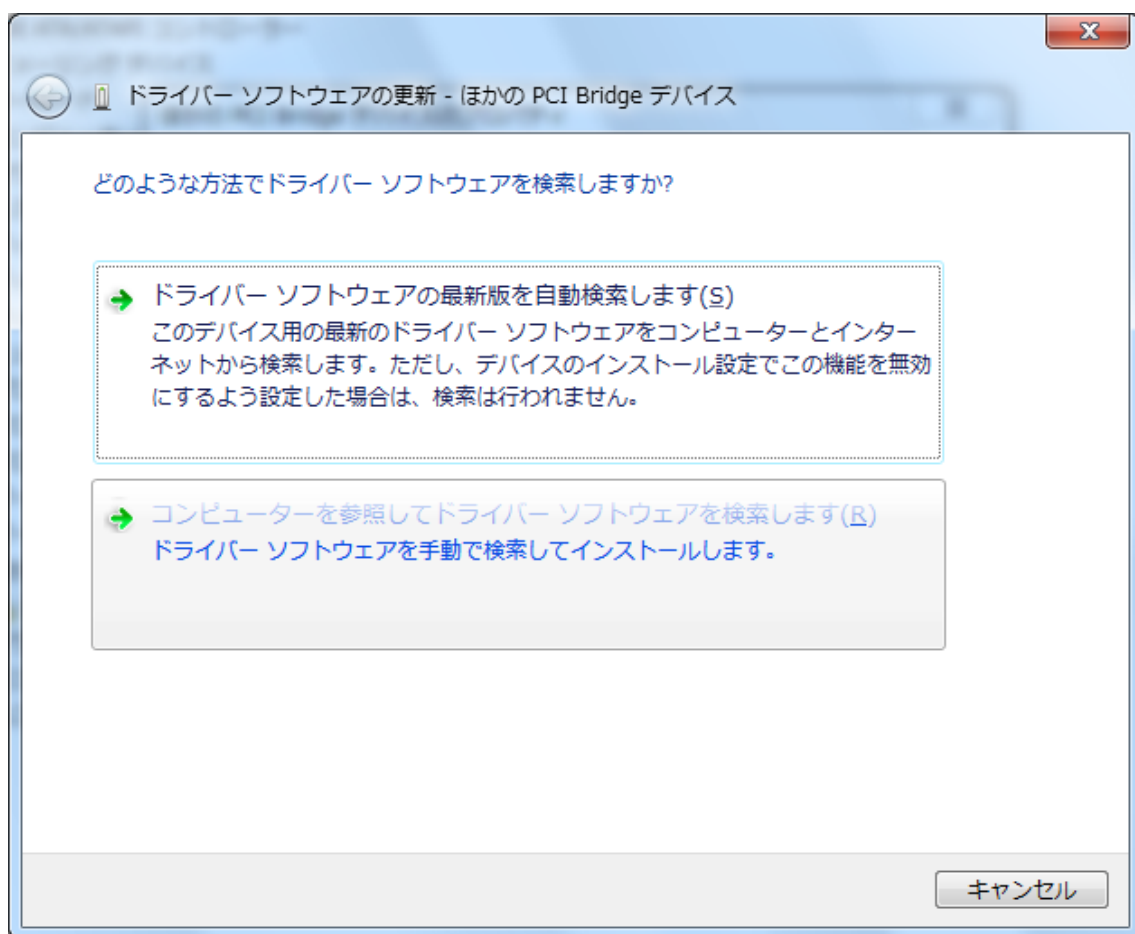
起動されない場合は、[設定] [コントロールパネル] [デバイスとプリンター][デバイスマネージャ]をダブルクリックして起動してください。



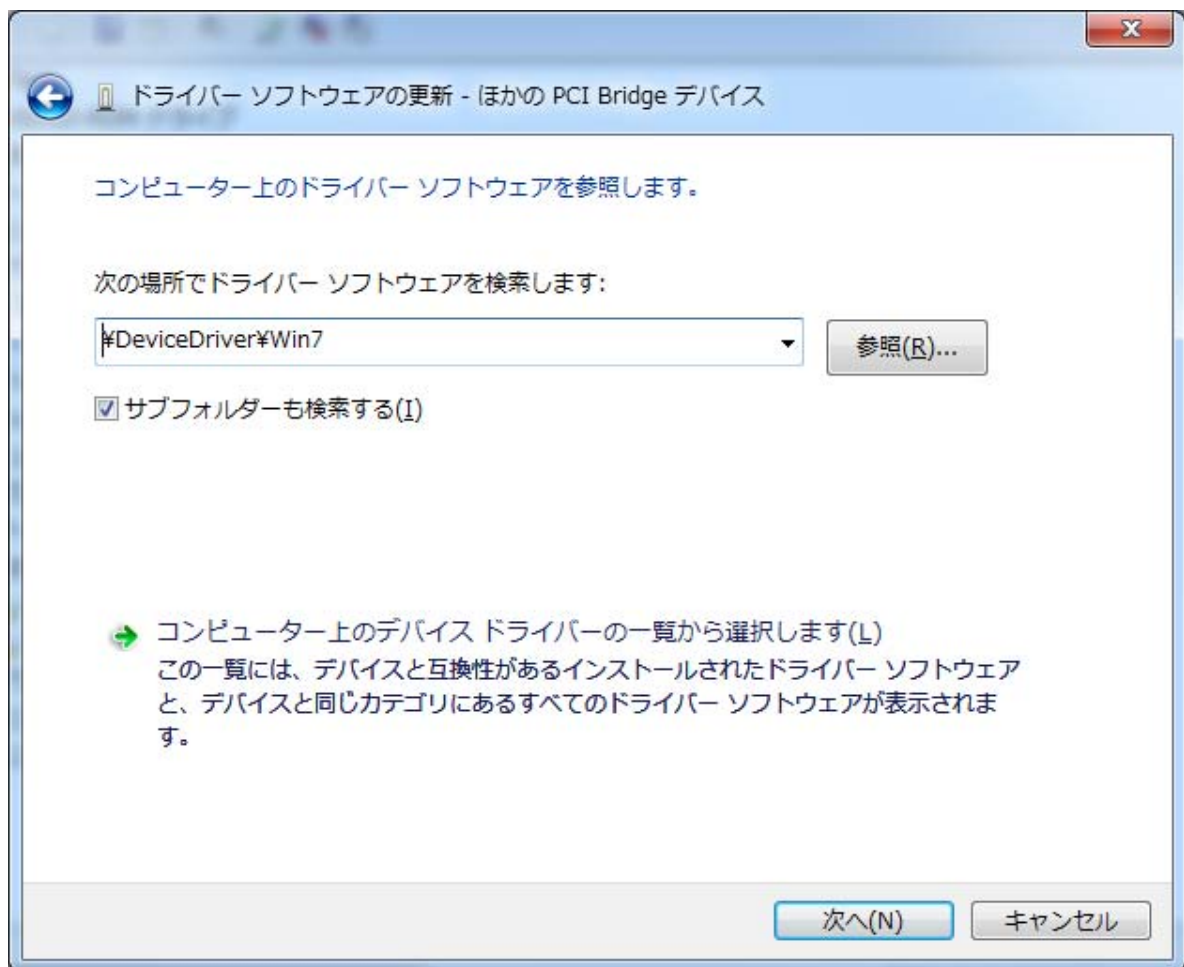
ほかの PCI Bridge デバイスをダブルクリックしてください



ドライバの更新を押してください

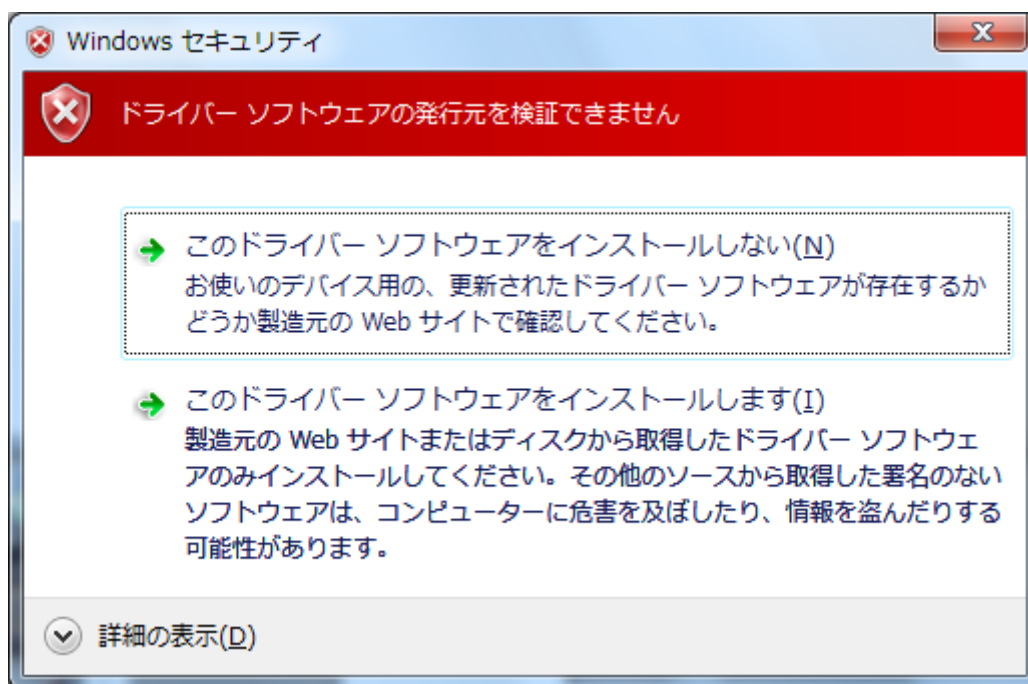


「ドライバーソフトウェアを手動で検索してインストールします」を選択してください。

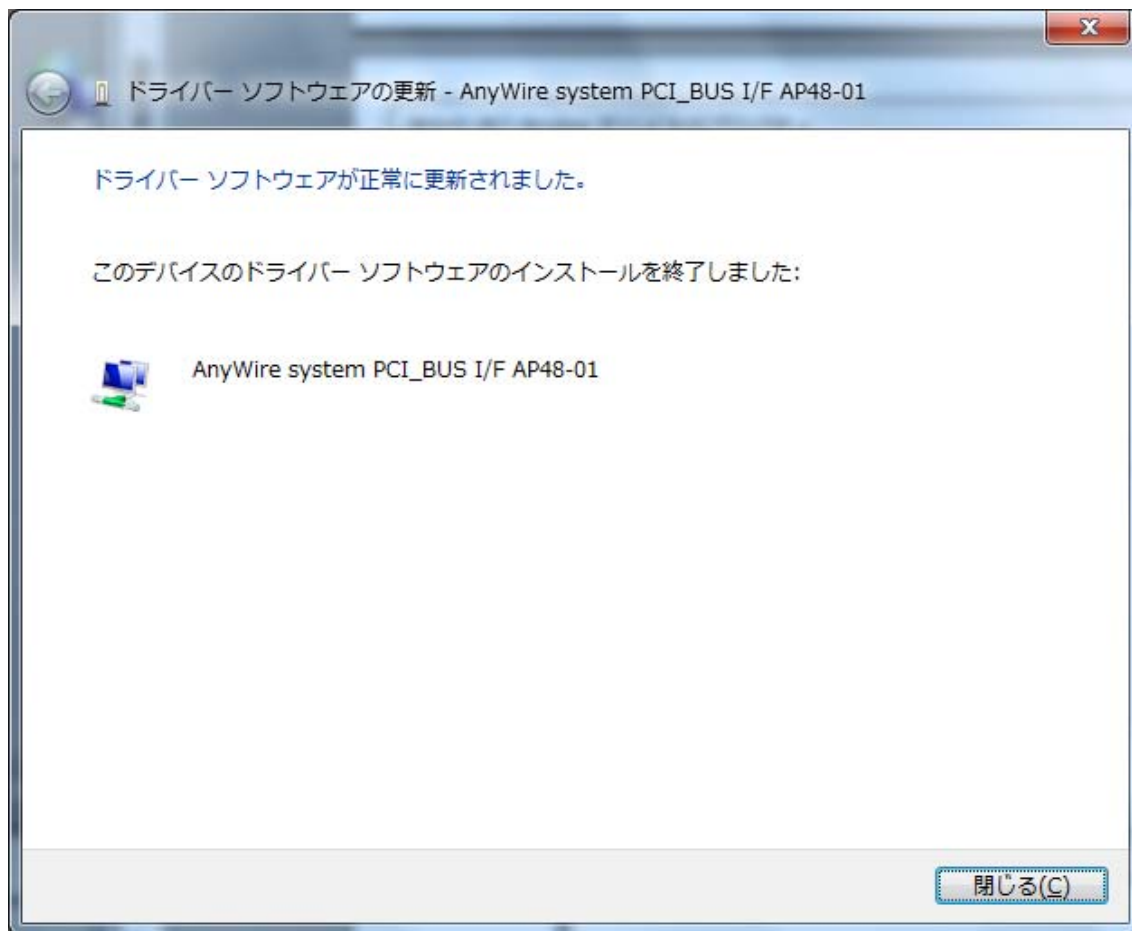


参照ボタンを押して、デバイスドライバが納められているフォルダを選択し、次へを押してください。

途中で、以下の表示が出ますが、「このドライバーソフトウェアをインストールします」を選択してください。

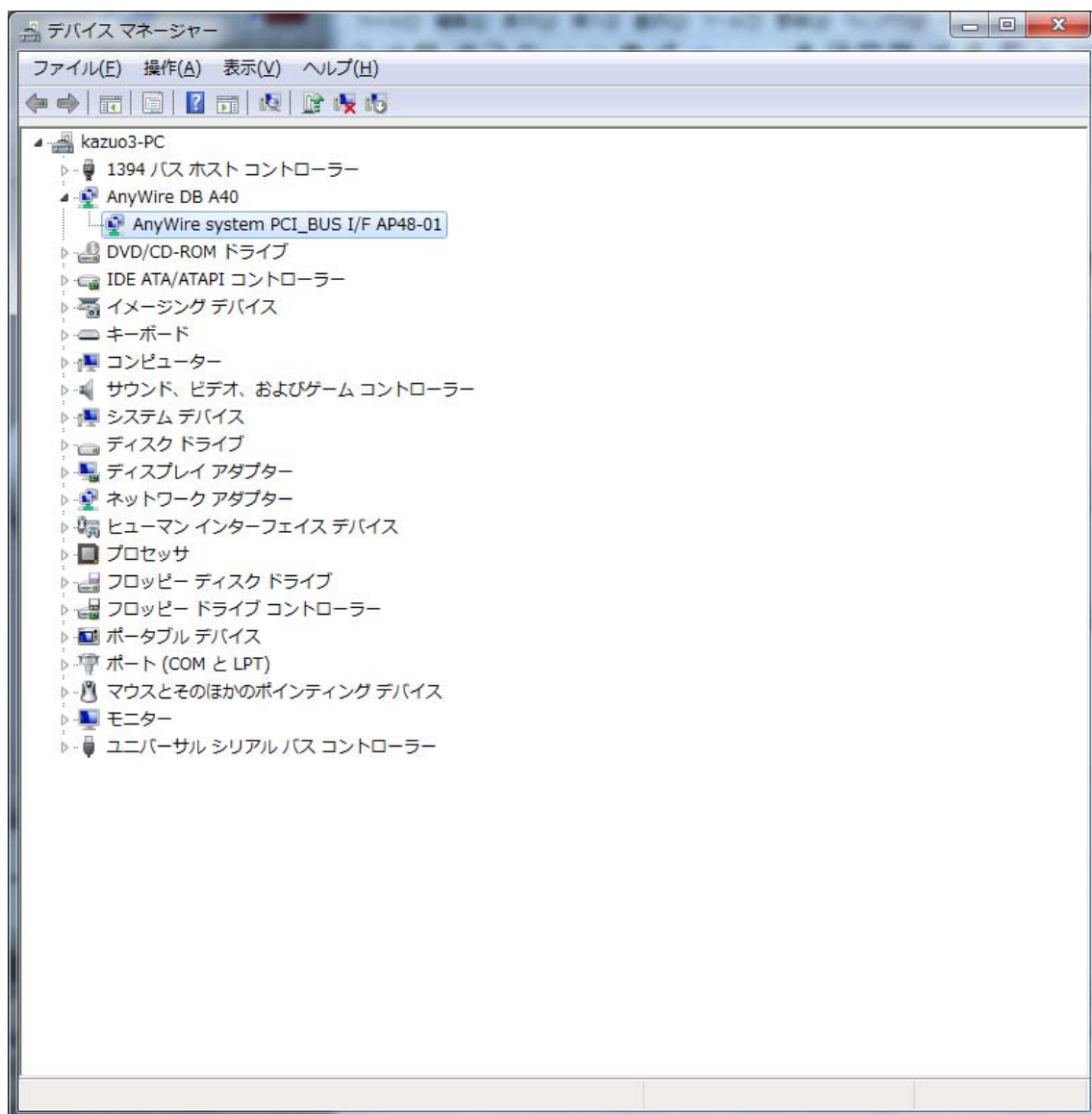


以下の表示になれば、インストールの完了です。





デバイスマネージャで、デバイスが認識されていることを確認してください。



### 3. ドライバライブラリ

ドライバライブラリは、以下の3つのファイルで構成されます

ファイル名	内容
AW_DB_A40_PCI.dll	ドライバライブラリ DLL
AW_DB_A40_PCI.lib	ドライバライブラリ・リンク用ライブラリ
AW_Function.h	ドライバライブラリ ヘッダファイル

### 3－1．インターフェース関数

関数名	内容
AW_GetDeviceInfo	ボードの情報を取得する
AW_ReadByte	バイトデータを入力する
AW_WriteByte	バイトデータを出力する
AW_ReadWord	ワードデータを入力する
AW_WriteWord	ワードデータを出力する

### 3 - 1 - 1 . ボードの情報を取得する

int AW_GetDeviceInfo(AW_DEVICE_LIST *AwDeviceList)	
引数	内容
AW_DEVICE_LIST *AwDeviceList	ボード情報を格納する構造体へのポインタ
戻り値	
int	認識されているボード数

使用方法：

```
int ReturnValue;
```

```
AW_DEVICE_LIST AwDeviceList;
```

```
ReturnValue = AW_GetDeviceInfo(&AwDeviceList);
```

AW_DEVICE_LIST 構造体	デバイス情報リスト構造体
unsigned int BoardCount;	認識ボード数
AW_DEVICE_INFO DeviceInfo[16];	デバイス情報

AW_DEVICE_INFO 構造体	デバイス情報構造体
unsigned short BoardNo;	ボード番号
unsigned short BoardType;	ボードタイプ 0x0000: デバイス無し 0x0020: AP48-01 0x0021: AP48-02
unsigned long MemSize;	メモリサイズ(アドレス範囲)

### 3-1-2. バイトデータを入力する

int AW_ReadByte(unsigned int BoardNo,ULONG Address,UCHAR *ByteData)	
引数	内容
unsigned int BoardNo	対象ボード番号
ULONG Address	入力アドレス
UCHAR *ByteData	読み出したデータを格納する変数へのポインタ
戻り値	
int	0 : 正常終了 -1 : ボード番号が範囲外 -2 : 存在しないボード番号が指定された -3 : パラメータエラー(アドレス範囲外等)

使用方法：

```
int ReturnValue;
```

```
unsigned int BoardNo;
```

```
ULONG Address;
```

```
UCHAR ByteData;
```

```
BoardNo = 1;
```

```
Address = 0x0000;
```

```
ReturnValue = AW_ReadByte(BoardNo,Address,&ByteData);
```

### 3 - 1 - 3. バイトデータを出力する

int AW_WriteByte(unsigned int BoardNo,ULONG Address,UCHAR ByteData)	
引数	内容
unsigned int BoardNo	対象ボード番号
ULONG Address	出力アドレス
UCHAR ByteData	出力データ
戻り値	
int	0 : 正常終了 -1 : ボード番号が範囲外 -2 : 存在しないボード番号が指定された -3 : パラメータエラー(アドレス範囲外等)

使用方法 :

```
int ReturnValue;
```

```
unsigned int BoardNo;
```

```
ULONG Address;
```

```
UCHAR ByteData;
```

```
BoardNo = 1;
```

```
Address = 0x0000;
```

```
ByteData = 0xFF;
```

```
ReturnValue = AW_WriteByte(BoardNo,Address,ByteData);
```

### 3 - 1 - 4. ワードデータを入力する

int AW_ReadWord(unsigned int BoardNo,ULONG Address,USHORT *WordData)	
引数	内容
unsigned int BoardNo	対象ボード番号
ULONG Address	入力アドレス
USHORT *WordData	読み出したデータを格納する変数へのポインタ
戻り値	
int	0 : 正常終了 -1 : ボード番号が範囲外 -2 : 存在しないボード番号が指定された -3 : パラメータエラー(アドレス範囲外等)

使用方法：

```
int ReturnValue;  
unsigned int BoardNo;  
ULONG Address;  
USHORT WordData;
```

```
BoardNo = 1;  
Address = 0x0000;
```

```
ReturnValue = AW_ReadWord(BoardNo,Address,&WordData);
```

### 3 - 1 - 5 . ワードデータを出力する

int AW_WriteWord(unsigned int BoardNo,ULONG Address,USHORT WordData)	
引数	内容
unsigned int BoardNo	対象ボード番号
ULONG Address	出力アドレス
USHORT WordData	出力データ
戻り値	
int	0 : 正常終了 -1 : ボード番号が範囲外 -2 : 存在しないボード番号が指定された -3 : パラメータエラー(アドレス範囲外等)

使用方法 :

```
int ReturnValue;
```

```
unsigned int BoardNo;
```

```
ULONG Address;
```

```
USHORT WordData;
```

```
BoardNo = 1;
```

```
Address = 0x0000;
```

```
WordData = 0xFFFF;
```

```
ReturnValue = AW_WriteWord(BoardNo,Address,WordData);
```



### 3 - 1 - 6 . C#でのプログラミング

DLL の関数インポート情報は、以下のようにして下さい。

```
[DllImport("AW_DB_A40_PCI.dll")]
extern static int AW_GetDeviceInfo(IntPtr DeviceList);
[DllImport("AW_DB_A40_PCI.dll")]
extern static int AW_ReadByte(uint BoardNo, uint Address, ref byte ByteData);
[DllImport("AW_DB_A40_PCI.dll")]
extern static int AW_WriteByte(uint BoardNo, uint Address, byte ByteData);
[DllImport("AW_DB_A40_PCI.dll")]
extern static int AW_ReadWord(uint BoardNo, uint Address, ref ushort WordData);
[DllImport("AW_DB_A40_PCI.dll")]
extern static int AW_WriteWord(uint BoardNo, uint Address, ushort WordData);
```

C#のサンプルコードです。（ボード情報の取得）

---

```
public partial class Form1 : Form {

    [StructLayout(LayoutKind.Sequential)]
    public struct _AW_DEVICE_INFO {
        public ushort BoardNo;
        public ushort BoardType;
        public uint MemSize;
    }

    [StructLayout(LayoutKind.Sequential)]
    public struct _AW_DEVICE_LIST {
        public uint BordCount;
        public _AW_DEVICE_INFO AwDeviceInfo0;
        public _AW_DEVICE_INFO AwDeviceInfo1;
        public _AW_DEVICE_INFO AwDeviceInfo2;
        public _AW_DEVICE_INFO AwDeviceInfo3;
        public _AW_DEVICE_INFO AwDeviceInfo4;
        public _AW_DEVICE_INFO AwDeviceInfo5;
        public _AW_DEVICE_INFO AwDeviceInfo6;
        public _AW_DEVICE_INFO AwDeviceInfo7;
        public _AW_DEVICE_INFO AwDeviceInfo8;
        public _AW_DEVICE_INFO AwDeviceInfo9;
        public _AW_DEVICE_INFO AwDeviceInfo10;
        public _AW_DEVICE_INFO AwDeviceInfo11;
        public _AW_DEVICE_INFO AwDeviceInfo12;
        public _AW_DEVICE_INFO AwDeviceInfo13;
        public _AW_DEVICE_INFO AwDeviceInfo14;
        public _AW_DEVICE_INFO AwDeviceInfo15;
    }
}
```

```

public static _AW_DEVICE_LIST AwDeviceList;

[DllImport("AW_DB_A40_PCI.dll")]
extern static int AW_GetDeviceInfo(IntPtr DeviceList);
[DllImport("AW_DB_A40_PCI.dll")]
extern static int AW_ReadByte(uint BoardNo, uint Address, ref byte ByteData);
[DllImport("AW_DB_A40_PCI.dll")]
extern static int AW_WriteByte(uint BoardNo, uint Address, byte ByteData);
[DllImport("AW_DB_A40_PCI.dll")]
extern static int AW_ReadWord(uint BoardNo, uint Address, ref ushort WordData);
[DllImport("AW_DB_A40_PCI.dll")]
extern static int AW_WriteWord(uint BoardNo, uint Address, ushort WordData);

private void button1_Click(object sender, EventArgs e) {
    AwDeviceList = new _AW_DEVICE_LIST();

    IntPtr AwDeviceListPtr=Marshal.AllocCoTaskMem(Marshal.SizeOf(AwDeviceList));
    Marshal.StructureToPtr(AwDeviceList, AwDeviceListPtr, false);

    Int32 int32Count = AW_GetDeviceInfo(AwDeviceListPtr);

    AwDeviceList
    =(_AW_DEVICE_LIST)Marshal.PtrToStructure(AwDeviceListPtr, AwDeviceList.GetType());

    Marshal.FreeCoTaskMem(AwDeviceListPtr);
}

```

---