PCRU01.DLL

Technical Guide

Introduction

General

All the communication routines of the PCRU01 are contained in a Dynamic Link Library PCRU01.DLL.

In this manual we will describe each of these functions provided by the DLL in detail. Calling the functions exported by the DLL, you can write custom Windows applications in Delphi, Visual Basic or any other 32-bit Windows application development tool that supports calls to a DLL.

A complete overview of the procedures and functions that are exported by the PCRU01.DLL follows.
Note that all the examples in the function description section are written in C++.
PCRU01 examples folder includes examples written in Visual Basic, Visual C# , Visual C++ and Delphi.

Readers should have an understanding of the basic data types as well as basic knowledge of the Microsoft Windows operating system.

Microsoft Visual Studio users please note: The PCRU01.DLL is a standard Windows DLL, you cannot reference it.

Calling convention

A calling convention is a scheme for how functions receive parameters from their caller and how they return a result. Different programming languages use different calling conventions, so it is important to know which calling convention is used by your programming language and which calling convention is used by the PCRU01.DLL.

The most common calling convention is the *stdcall* calling convention, and this is also the one we have used for our DLL.

If you are using .NET (VB.NET or C#) you do not need to worry about this since the calling convention in .NET is also *stdcall*. However if you are using C to import the functions provided by the DLL, you will need to pay special attention to this.

Overview of the Functions

General functions

`int OpenDevice();`	*Opens the communication link to the PCRU01 device*
`void CloseDevice();`	*Closes the link to the PCRU01 device*
`int Calibrate();`	*Eliminates the effect of the offset voltage of the PCRU01*

Input function

`void ReadData(int *Buffer);`	*Reads the timer counter status and the A/D data from the PCRU01 to a buffer in the application program*

Output function

```
void SetVoltageRange (int Channel, int Range);
```

Sets the input voltage range of the selected channel 1...4.

Revision History

Date	Version	Description
26/09/2016	-	Initial PCRU01 DLL Technical Guide release.

Function List

OpenDevice

Syntax

```
int OpenDevice();
```

Result

`Int:` If succeeded the return value will be 0.
Return value -1 indicates that PCRU01 was not found.

Description

Opens the communication link to the PCRU01. Loads the drivers needed to communicate via the USB port. This procedure must be performed before any attempts to communicate with the PCRU01.

Example

```
int h = OpenDevice();
switch (h)
{
    case  0:
        Label1->Text = "PCRU01 connected";
        break;
    case  -1 :
        Label1->Text = "PCRU01 not found";
        break;
}
```

CloseDevice

Syntax

```
void CloseDevice();
```

Description

Unloads the communication routines for PCRU01 and unloads the driver needed to communicate via the USB port. This is the last action of the application program before termination.

Example

```
        CloseDevice();
```

ReadData

Syntax

```
void ReadData(int *Buffer);
```

Parameter

`Buffer:` Pointer to an array of six 32-bit integers where the data will be read.

Description

Reads the timer counter status and the A/D data from the PCRU01 to a buffer in the application program. The timer counter is incremented every 1ms. The new data from A/D converter channels 1...4 is updated every time the timer counter is incremented.
Data structure:

Timer data		**A/D converter data**			
LSB	MSB	CH1	CH2	CH3	CH4

Example

```
        int Buffer[6];
        ReadData(Buffer);
```

SetVoltageRange

Syntax

```
void SetVoltageRange(int Channel, int Range);
```

Parameter

`Channel:` Value between 1 and 4 which corresponds to the input channel number whose voltage range is to be set.
`Range:` The index of the voltage range.

Index	**Input Voltage Range**
1	±20V
2	±8V
3	±4V
4	±2V
5	±0.8V
6	±0.4V

Description

Set the voltage range setting of the PCRU01.

Example

```
SetVoltageRange (1, 6);  // Channel 1 voltage range is set to be ±0.4V
```

Calibrate

Syntax

```
int Calibrate();
```

Result

`Int:` If succeeded the return value will be 0.
Return value -1 indicates that calibration was cancelled.
Return value -2 indicates that calibration was failed.

Description

Calibrates the PCRU01 to output as close as possible value 512 at zero input voltage.

Example

```
int h = Calibrate();
switch (h)
{
    case  0:
        Label1->Text = "PCRU01 calibrated";
        break;
    case  -1 :
        Label1->Text = "PCRU01 calibration cancelled";
        break;
    case  -2 :
        Label1->Text = "PCRU01 calibration failed";
        break;

}
```