

PRU <b>SEROP</b>	Title <b>Release Notes PC SDK 5.14.01</b>	Ref RP5.15	
Department DMRO/PRS	Date 2011-06-15	Filename Release Notes PC SDK 5.14.01.doc	Page 1/10
	Dealt with by, telephone		

## Table of Content

**GENERAL ..... 2**

**Release Information..... 2**

**Introduction..... 2**

**Installation..... 3**

**Hardware and Software requirements..... 4**

**Compatibility..... 5**

**Updates in PC SDK 5.14.01 vs. 5.14 ..... 6**

**Updates in PC SDK 5.14 vs. 5.13..... 6**

**Information 5.14.01 ..... 6**

**Information 5.14..... 6**

**Information 5.13.02 ..... 7**

**Information 5.13.01 ..... 7**

**Information 5.13 ..... 7**

**Corrected Issues 5.14.01 PDD..... 8**

**Corrected “Product Defect Document” 5.14 PDD ..... 8**

**Corrected “Product Defect Document” 5.13.02 PDD ..... 9**

**Corrected “Product Defect Document” 5.13.01 PDD ..... 9**

**Corrected “Product Defect Document” 5.13 PDD ..... 9**

**Known Limitations..... 9**

**Product Support ..... 10**

PRU <b>SEROP</b>	Title <b>Release Notes PC SDK 5.14.01</b>	Ref RP5.15	
Department DMRO/PRS	Date 2011-06-15	Filename Release Notes PC SDK 5.14.01.doc	Page 2/10
	Dealt with by, telephone		

## General

### Release Information

The information should be considered as last minutes information and most up-to-date. For more information please visit Robot Release Information (RRI) Homepage:

<http://roboticsportal.se.abb.com/prod/!display?action=page&id=351>

### Introduction

This file contains release notes for PC SDK 5.14.01.

Previously PC SDK and FlexPendant SDK were distributed as the product Robot Application Builder (RAB). Starting with RobotWare 5.13, PC SDK and FlexPendant SDK are distributed together with RobotStudio. The SDKs are not affected by this change.

PC SDK 5.14.01 is included in the RobotStudio installer, which is distributed on the RobotWare 5.14.01 DVD. It can be used for free by anyone who wants to develop a customized PC operator interface, for the IRC 5 controller. It can also be used to develop RobotStudio add-ins that interacts with the controller. With PC SDK it is possible to create an application that connects to one or several IRC5 controllers, real as well as virtual. No license is required to develop applications using PC SDK. The end user of a PC SDK application, needs to have the option PC Interface on the targeted controller.

The installation includes software, documentation and tools as specified below:

#### Software

PC SDK (5.14.01)

#### Documentation

Application manual – PC SDK (5.14.01), Html Help and Pdf).

PC SDK Reference Documentation (5.14.01), documentation of class libraries with method signatures in C# and Visual Basic (Html Help).

#### Tools

ABBControllerAPI.msm - merge module including the PS SDK dlls to be used when a PC SDK application is deployed to a customer's PC

ABB Industrial Robot Communication Runtime.msi - to be used when a PC SDK application is deployed to a PC without RobotStudio

After installation the documentation can be launched from Windows Start Menu\Programs\ABB Industrial IT\Robotics IT\Robot Studio 5.14.

In no event shall ABB be liable for incidental or consequential damages arising from use of this product, of the software and hardware described in relating product documentation.

Visit our web site at <http://www.robotstudio.com/community> for information and updates. Click the Download symbol to the right in order to download RobotStudio 5.14.01, including PC SDK 5.14.01 for free.

Click Developer Tools to learn more about the PC SDK and visit the User Forum, where developers discuss software problems and solutions online.

PRU <b>SEROP</b>	Title <b>Release Notes PC SDK 5.14.01</b>	Ref RP5.15	
Department DMRO/PRS	Date 2011-06-15	Filename Release Notes PC SDK 5.14.01.doc	Page 3/10
	Dealt with by, telephone		

## Installation

To install PC SDK click *RobotStudio* on the RobotWare & RobotStudio DVD. If you select the default installation option **Full**, PC SDK will be installed. If you want to install only PC SDK and not RobotStudio select the installation option **Custom**.

PC SDK 5.xx will be installed side by side with any previous major version of PC SDK 5.xx, while minor versions within a release will update to the latest one.

Before you start the installation you are recommended to read chapter 2, *Installation and development environment*, in the PC SDK Application Manual which is available on the Documentation DVD in PDF format.

Microsoft Visual Studio development environment is used to develop PC SDK applications. Visual Studio 2005 Express or better, or Visual Studio 2008 Express or better is required.

The default installation path is C:\Program Files\ABB Industrial IT\Robotics IT\SDK\PC SDK 5.14.01

PRU <b>SEROP</b>	Title <b>Release Notes PC SDK 5.14.01</b>	Ref RP5.15	
Department DMRO/PRS	Date 2011-06-15	Filename Release Notes PC SDK 5.14.01.doc	Page 4/10
	Dealt with by, telephone		

## Hardware and Software requirements

### Software requirements

**Microsoft Windows XP SP3 – 32bit edition**

**Microsoft Windows 7 – 32bit edition**

**Microsoft Windows 7 – 64bit edition**

***Note!** PC SDK application that shall be executed on Windows 7 – 64 bit edition, must be built with target platform set to “x86”. The default setting in VisualStudio 2008 is “AnyCPU”*

### **Microsoft Visual Studio 2008**

PC SDK requires Express edition or better.

### **Microsoft Visual Studio 2010**

PC SDK requires Express edition or better.

***Note!** PC SDK applications requires Robot Communications Runtime to be installed on your PC. It comes with the RobotStudio installation, but can also be installed separately from C:\Program Files\ABB Industrial IT\Robotics IT\SDK\PC SDK 5.14.\redistributable\RobotCommunicationRuntime after you have installed PC SDK.*

### Recommended hardware

10 MB free disk-space on the installation hard disk

### IRC5 requirements

- RobotStudio 5.14.xx for building a test system and for debugging and testing in the virtual environment

- RobotWare option "PC Interface" for communication with a real IRC5 controller.

***Note!** PC applications developed with PC SDK 5.14.xx require RobotWare version 5.07 or higher on the IRC5 controller.*

PRU <b>SEROP</b>	Title <b>Release Notes PC SDK 5.14.01</b>	Ref RP5.15	
Department DMRO/PRS	Date 2011-06-15	Filename Release Notes PC SDK 5.14.01.doc	Page 5/10
	Dealt with by, telephone		

## Compatibility

### Compatibility

PC SDK 5.14 and 5.13 are compatible, there are no breaking changes.

PC SDK communicates with the robot controller using the Robot Communication Runtime, which is designed to be backwards compatible with earlier versions of RobotWare.

**Note!** Functionality introduced in later versions of RobotWare will not be available for a PC SDK application that is connected to a controller with an older version of RobotWare.

For example the Messaging functionality is only supported on RobotWare 5.10 and above. This means that a PC SDK application cannot use the `Ipc` class when communicating with a controller with RobotWare 5.09.

The code will compile, but an exception will be thrown at runtime. Application developers are responsible for handling this scenario in their applications. Please refer to the PC SDK Reference Documentation for details.

**Note!** Compatibility between RobotWare revisions is guaranteed (PC SDK 5.14 will be compatible with PC SDK/RW 5.14.01 etc).

### EventHandlers and events

For 5.09 the internal event architecture was completely redesigned.

1. All previous event handlers are [Obsolete] and existing events are changed to the `EventHandler<TEventArgs>` generic delegate.

**Ex:** This example shows how to change from the old to the new event handler type.

*// This old line will fail...*

```
myEventLog.MessageWritten += new MessageWrittenEventHandler( OnMessage );
```

*// and should be replaced like this...*

```
myEventLog.MessageWritten += new EventHandler<MessageWrittenEventArgs>(
    OnMessage );
```

*//or preferably...*

```
myEventLog.MessageWritten += OnMessage;
```

2. Previous versions of PC SDK used the Windows Thread Pool internally to raise events.

From 5.09 a single thread is used to dispatch all events. This reduces the risk of race conditions in client code. However, it makes it even more important to use the `Control.BeginInvoke( ... )/Control.EndInvoke( ... )` pattern to avoid event starvation. Additionally, we now raise all events internally prior to any external subscribers, as this will reduce the risk of race conditions between inner and external subscribers.

If your application is based on `Control.Invoke( ... )` and not on `Control.BeginInvoke( ... )` all events will be serialized, both internally and externally. However in previous PC SDK versions you may encounter "out-of-order" events and thread pool starvation through the use of `Control.Invoke( ... )`.

PRU <b>SEROP</b>	Title <b>Release Notes PC SDK 5.14.01</b>	Ref RP5.15	
Department DMRO/PRS	Date 2011-06-15	Filename Release Notes PC SDK 5.14.01.doc	Page 6/10
	Dealt with by, telephone		

Some public events raise an initial event immediately when the subscription is activated; however this is not consistent or by design and should therefore be avoided. Later versions of PC SDK will remove all initial events.

3. `Mastership.Request( ... )` throws an `InvalidOperationException` if the user is not authenticated against the controller, previous versions raised an `ArgumentException`.

## Updates in PC SDK 5.14.01 vs. 5.14

No changes

## Updates in PC SDK 5.14 vs. 5.13

No changes

## Information 5.14.01

### Eventlog.MessageWritten now works on Chinese system

There was a problem with listening to the event `Eventlog.MessageWritten` on a controller with Chinese language (culture code "zh"). This has now been corrected.

### Creating a single installer for a PC SDK application using InstallShield Limited Edition using Custom Action

Using InstallShield 2010 Limited Edition and Visual Studio 2010, you can create an installer that installs both your PC SDK application and the Robot Communication Runtime. This means you can deliver one installer to your customers instead of two.

This solution makes use of a Custom Action in InstallShield which launches the embedded Robot Communication Runtime redistributable installer. Please refer to this post in the Developer Tools User Forum for more details:

[http://www.robotstudio.com/forum/forum\\_posts.asp?TID=5047&SID=139z8358c3d313778a61366bzc384fc](http://www.robotstudio.com/forum/forum_posts.asp?TID=5047&SID=139z8358c3d313778a61366bzc384fc)

## Information 5.14

### Memory leakage in PC-SDK (DSE10353)

A memory leak in the PC SDK method `Controller.Rapid.GetRapidData()` has been corrected.

### No simple installation for PC SDK apps because msm file RCR (DSE10406)

The Robot Communication Runtime is at present not available as a merge module (.msm file), only as a separate installer. If you are creating an installer for an application built with PC SDK, it is not possible to let it install the Robot Communication Runtime.

PRU <b>SEROP</b>	Title <b>Release Notes PC SDK 5.14.01</b>	Ref RP5.15	
Department DMRO/PRS	Date 2011-06-15	Filename Release Notes PC SDK 5.14.01.doc	Page 7/10
	Dealt with by, telephone		

The Robot Communication Runtime must first be installed on the target computer, using the separate installer, which can be found in the Redistributable folder of the PC SDK installation.

A merge module containing the Robot Communication Runtime is planned for PC SDK 5.14.01.

## Information 5.13.02

### PC SDK exception in AWPP when running on Windows 7 - 64 bit (DSE9926)

PC SDK can now handle parenthesis in the system or release folders, e.g. "Program Files (x86)". Previous versions throw an exception.

### Parameter values for x- p- and i-start missing in Restart com (DSE10140)

From revision 5.13.02 is it possible to do a I-Start via PC SDK.

A new enum value, IStart, has been added to the enum ControllerStartMode.

E.g.

```
using (Mastership.Request(m_controller.Rapid))
using (Mastership.Request(m_controller.Configuration))
{
    m_controller.Restart(ControllerStartMode.IStart);
}
```

### PCSDK 5.13.255.0 Evenlog Class (DSE1069)

This is now corrected.

## Information 5.13.01

### Connect when controller is in SYS\_FAIL state (DSE9830)

A bug in PC SDK 5.13 makes it impossible to connect to a controller which is in sys-fail state. This has now been corrected.

## Information 5.13

**It is now possible to access RAPID data that are declared in hidden modules and in hidden tasks.**

E.g

```
RapidData data;
Controller controller;
```

*// Access data declared in a hidden task.*

*// A hidden task has the -hidden attribute defined in the Controller Configuration.*

```
data = controller.Rapid.GetRapidData("hidden_task", "module1", "num1");
```

*// Access data declared in a hidden module*

*// A hidden module has the -hidden attribute defined in the Controller Configuration.*

PRU <b>SEROP</b>	Title <b>Release Notes PC SDK 5.14.01</b>	Ref RP5.15	
Department DMRO/PRS	Date 2011-06-15	Filename Release Notes PC SDK 5.14.01.doc	Page 8/10
Dealt with by, telephone			

```
data = controller.Rapid.GetRapidData("T_ROB1", "hidden_module1", "num2");
```

```
// Access data declared in a shared hidden module
```

```
// A shared hidden module has the -hidden and the -shared attributes defined in the Controller Configuration.
```

```
data = controller.Rapid.GetRapidData("num3");
```

```
// Access data declared in an installed module
```

```
// An installed module is loaded from a install script using the -install option.
```

```
data = controller.Rapid.GetRapidData("T_ROB1", "#SYS", "num4");
```

```
// Access data declared in an installed hidden module
```

```
// An installed module is loaded from an install script using the -install and -hidden options.
```

```
data = controller.Rapid.GetRapidData("T_ROB1", "#SYS", "num5")
```

## Simulate physical setting of input signals on a Virtual Controller

It is now possible to simulate the physical setting of the value of an input signal.

The previously existing method to set the value of a signal is similar to setting a signal from RAPID. Depending on the Access Level the controller safety system may or may not allow the signal value to be changed. When a physical signal is set by connecting it to +24V the controller cannot prevent it from happening, no matter what Access Level the signal has. By setting the property **-InputAsPhysical** on the **Signal** class, a subsequent write operation will simulate a physical write. This is important in a scenario where a PC SDK application shall set signal values in order to verify a RAPID program in a Virtual Controller

## Dnum variable not accessible from PC SDK (DSE9020)

It is now possible to access RAPID dnum variables via PC SDK.

A new class Dnum has been added.

## 32 bit group signals is now supported by the PC SDK. (DSE9386)

32 bit group signals is now supported by the PC SDK.

## Memory leak (DSE9493)

There was a memory leak in PCSDK when retrieving ipc messages. It leaked about 400 bytes per message it retrieves, this has now been corrected.

<i>Corrected Issues 5.14.01</i>	<i>PDD</i>
RobotStudio installation removes 5.13 SDK files	DSE10707
ControllerFileSystemInfo.Exists throws exception	Forum
PC SDK CapiDevTest Update	DSE10702
No simple installation for PC SDK apps because msm file RCR is missing	DSE10406
UIInstructionEvent does not support more than 4 lines	Forum
Comma in UIMessageBox text is interpreted as newline	Forum
<i>Corrected "Product Defect Document" 5.14</i>	<i>PDD</i>
Memory leakage in PC-SDK	DSE10353

PRU <b>SEROP</b>	Title <b>Release Notes PC SDK 5.14.01</b>	Ref RP5.15	
Department DMRO/PRS	Date 2011-06-15	Filename Release Notes PC SDK 5.14.01.doc	Page 9/10
	Dealt with by, telephone		

## Corrected "Product Defect Document" 5.13.02 PDD

PC SDK exception in AWPP when running on Windows 7 - 64 bit	DSE9926
Parameter values for x- p- and i-start missing in Restart com	DSE10140
PCSDK 5.13.255.0 Evenlog Class	DSE10169

## Corrected "Product Defect Document" 5.13.01 PDD

Connect when controller is in SYS_FAIL state	DSE9830
--	---------

## Corrected "Product Defect Document" 5.13 PDD

Dnum variable not accessible from PC SDK	DSE9020
32 bit group signals is now supported by the PC SDK.	DSE9386
Memory leak	DSE9493

## Known Limitations

- PC SDK UIInstructionEvent / RAPID UIListView - if you are using characters like double quote (") and brackets ([ ]) in the RAPID array of list item strings, the UIListViewEventArgs.ListItems may return the wrong number of list items. Avoid using double quote (") and brackets ([ ]).
- In manual mode, when a PC SDK application releases master and immediately requests master again, the FlexPendant is locked up when the user presses the prompt to grant write access to the PC SDK application. In automatic mode there is no problem.
- PC SDK IPC Messaging - the PC SDK erratically ALWAYS sends 444 byte. Data must therefore be null terminated. Also, there is a problem reading the data if the PC SDK thread is running as STA. Changing it to MTA will solve that problem.
- ArrayData.FillFromString is not correctly implemented in the PC SDK. Users are recommended to use RapidData.StringValue to read and write small arrays (e.g. 100 num variables) in one call.
- The property IsLocal on the class RapidData, returns true for shared data, even though such data is visible from all modules.

The declarations of default zonedata and speeddata has been moved from module BASE.SYS in each task, and are now installed as shared data.

If an application used to create a RapidData object representing for example "v1000", the IsLocal property returned false in RobotWare 5.12, but will return true in RobotWare 5.13."

PRU <b>SEROP</b>	Title <b>Release Notes PC SDK 5.14.01</b>	Ref RP5.15	
Department DMRO/PRS	Date 2011-06-15	Filename Release Notes PC SDK 5.14.01.doc	Page 10/10
	Dealt with by, telephone		

## *Product Support*

For technical support please turn to your local ABB office.

<http://www.abb.com/contacts>

For help, advice and the latest updates please visit the User Forum.

<http://www.robotstudio.com/forum>