

Release Notes
RobotStudio SDK
5.60

Revision: A

The information in this manual is subject to change without notice and should not be construed as a commitment by ABB. ABB assumes no responsibility for any errors that may appear in this manual.

Except as may be expressly stated anywhere in this manual, nothing herein shall be construed as any kind of guarantee or warranty by ABB for losses, damages to persons or property, fitness for a specific purpose or the like.

In no event shall ABB be liable for incidental or consequential damages arising from use of this manual and products described herein.

This manual and parts thereof must not be reproduced or copied without ABB's written permission, and contents thereof must not be imparted to a third party nor be used for any unauthorized purpose. Contravention will be prosecuted.

Additional copies of this document may be obtained from ABB at its then current charge.

© Copyright 2013 ABB All rights reserved.
ABB AB
Robotics Products
SE-721 68 Västerås
Sweden

Overview	4
1 Release Information	5
1.1 General	5
1.2 Contents	5
1.3 Prerequisites	6
1.4 Software Requirements	6
2 Information to all users of Visual Studio Tools for Applications (VSTA)	7
2.1 VSTA is no longer installed	7
2.2 Switch to SmartComponents or Visual Studio 2012	7
3 What's New in 5.60?	8
3.1 All new types and methods	8
3.1.1 ABB.Robotics.RobotStudio.Environment	8
3.1.2 ABB.Robotics.RobotStudio	9
3.1.3 ABB.Robotics.RobotStudio.Stations	9
3.1.4 ABB.Robotics.RobotStudio.Stations.Forms	10
3.1.5 ABB.Robotics.RobotStudio.Math	13
4 Late-breaking information	14
4.1 Overview	14
5 Corrections	15
5.1 Corrections in RobotStudio 5.60	15
6 Known Limitations	16
6.1 Development Environment	16
7 Installation information	17
7.1 Installing RobotStudio SDK 5.60	17
8 Compatibility	18

Overview

Description

This document provides information about the new features, problems corrected, and installation of RobotStudio SDK.

Product Overview

RobotStudio SDK is available for download from the Robotics Developer Center.

It lets you develop applications on the RobotStudio platform. It could be utility Add-Ins or PowerPacs that provides new functionality customized user interfaces in RobotStudio.

You can also combine RobotStudio SDK with PC SDK to communicate with real or virtual IRC5 controllers.

Custom Smart Components with Code Behind can also be developed using the RobotStudio SDK.

To get started, or to learn more, visit our Developer Center:
developercenter.robotstudio.com.

Here you will also find the User Forum, where developers discuss software problems and solutions online.

1 Release Information

1.1 General

Release Name

The release name is RobotStudio SDK 5.60 and the build number is 5.60.5731.0084.

For information about RobotStudio, please refer to the document 'Release Notes RobotStudio 5.60.pdf'.

Release Date

The release date is November 29th, 2013.

1.2 Contents

Overview

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

Software

ABB.Robotics.Math.dll
ABB.Robotics.Environment.dll
ABB.Robotics.RobotStudio.dll
ABB.Robotics.RobotStudio.Controllers.dll
ABB.Robotics.RobotStudio.Documents.dll
ABB.Robotics.RobotStudio.Stations.dll
ABB.Robotics.RobotStudio.Stations.Forms.dll

Documentation

Reference Manual – RobotStudio SDK. Documentation of class libraries with method signatures in C#. Available at developercenter.robotstudio.com and the installation folder.

Documentation and samples are available developercenter.robotstudio.com.

- API reference
- Sample projects
- Walkthroughs

Reference Manual – RobotStudio SDK (Html Help).

After installation the help file can be launched from the RobotStudio Help Menu, and Windows Start Menu\Programs\ABB Industrial IT\Robotics IT\RobotStudio 5.60\SDK

Tools

LibraryCompiler.exe – A tool for batch creation for RobotStudio libraries such as SmartComponents and Mechanisms.

Visual Studio Project Templates (C#)

RobotStudio Add-In – template for a standard Add-In

RobotStudio Smart Component – template for a Smart Component with Code Behind and XML description.

1.3 Prerequisites

Before you install...

- You must have Administrator privileges.
 - Visual Studio 2012 needs to be installed, in order for the Add-In and SmartComponent Visual Studio project templates to be installed.
-

Licensing

The use of RobotStudio SDK free of charge.

In order to run an Add-In in RobotStudio, a RobotStudio Premium license is required.

1.4 Software Requirements

Operating system

Operating System	
Microsoft Windows 7 SP1	32-bit edition
Microsoft Windows 7 SP1 (recommended)	64-bit edition
Microsoft Windows 8.1 (recommended)	64-bit edition

Visual Studio 2012

Microsoft Visual Studio 2012 development environment is required.

The Visual Studio project templates is supported by the English version of Visual Studio only.

RobotStudio 5.60

RobotStudio 5.60 is required to debug and run Add-Ins and SmartComponents developed with RobotStudio SDK.

2 Information to all users of Visual Studio Tools for Applications (VSTA)

2.1 VSTA is no longer installed

From RobotStudio 5.60 onwards, VSTA will no longer be available in RobotStudio.

This has the following implications:

- The IDE is not available.
- Macros will not be executed.
- Stations and libraries with macros can be loaded, but the macros will not be executed.

2.2 Switch to SmartComponents or Visual Studio 2012

The C# or VB.NET code in a VSTA macro can be migrated to a Visual Studio 2012 Express C# project, and you can continue development using Visual Studio.

In case you have a macro that is used in together with the EventManager, you also have the option to implement the same functionality using RobotStudio SmartComponents, instead of using Visual Studio and C#.

If you want to migrate another macro, ABB recommends using Visual Studio 2012 Express.

3 What's New in 5.60?

Overview

This section contains information on the new features of RobotStudio SDK 5.60.

3.1 All new types and methods

Overview

This section contains information about all the new types in the API, and existing types which has been extended with new methods. Each updated namespace has a separate chapter. For extended types, the name of the type is in bold face, followed by its new methods.

3.1.1 ABB.Robotics.RobotStudio.Environment

Extended types

```
class CommandBarButton
```

```
Boolean DefaultChecked
```

```
Boolean DefaultEnabled
```

```
Boolean IsChecked
```

```
Boolean IsEnabled
```

```
class CommandBarGalleryPopup
```

```
Int32 NumberOfTextLines
```

```
class CommandBarPopup
```

```
CommandBarPopup: CommandBarPopup(String,  
String, CommandBarControl[])
```

```
class DisplayCommandGroupEventArgs
```

```
Void AddHeader(String)
```

```
class MainWindow
```

```
Boolean IsActive
```


3.1.2 ABB.Robotics.RobotStudio

Extended types

```
class LicenseValidator
static IDisposable AcquireLicenseToken(String)
```

```
class Options
Boolean RestartRequired
Void SetDefaultValue(String, String, Color)
Void SetValue(String, String, Color)
```

```
class ProjectDocument
Boolean IsActive
String Comment
```

```
class RobotStudioAPI
Void AssertMainThread()
Void SkipThreadAffinityChecks(Boolean)
```

3.1.3 ABB.Robotics.RobotStudio.Stations

New types

```
ClipPlane
ClipPlaneCollection
EdgeGraph
GraphicButton
GraphicButtonCollection
```

Extended types

```
class Camera
Void SetView(Vector3, Vector3, Vector3)
```

```
class Coedge
Coedge Next
Coedge Previous
```

```
class Edge
Boolean IsParameterValid(Double)
```

```
class Face
BoundingBox GetBoundingBox(Boolean)
Coedge Previous
```

3.1.4 ABB.Robotics.RobotStudio.Stations.Forms

New types

GraphicButton
GraphicButtonCollection
GraphicCapabilities
GraphicControlSettings
GraphicConverter
GraphicExportSettings
GraphicImportSettings
Light
LightCollection
enum LightType
ParticleSystem
ParticleSystemCollection
ParticleSystemData
PointCloud
PointCloudCollection
ProjectObjectTag
ProjectObjectTagCollection

Extended types

class GraphicControl
GraphicControl(GraphicControlSettings)
Boolean AdvancedLighting
GraphicButtonCollection Buttons
GraphicCapabilities Capabilities
ClipPlane ClipPlane
Boolean EnableTouchNavigation
Action<GraphicButton> GraphicButtonClicked
Action<GraphicButton> GraphicButtonTracked
GraphicPicker Picker {set}
Ray RayFromPoint(Int32, Int32)
Boolean SuspendRedraw {get}

class GraphicPickerEventArgs
Cursor Cursors
Material PickedMaterial

class NumericTextBox
TextBox InnerTextBox

class TemporaryGraphic
BoundingBox GetBoundingBox(Boolean)
Void UpdateParticleSystem(ParticleSystemData)

class TemporaryGraphicCollection

TemporaryGraphic DrawParticleSystem(Matrix4, ParticleSystemData)

TemporaryGraphic DrawPointCloud(Matrix4, Vector3[], Color, Double)

class GraphicComponent

CanImport(String)

BoundingBox GetBoundingBox(Boolean)

static Task<GraphicComponent>
ImportAsync(String)

static Task<GraphicComponent>
ImportAsync(String, GraphicImportSettings, IProgressCallback)

class Material

MaterialEffect Effect

SizeF TextureSize

class MeshFace

List<Vector3> EndPoints

enum PackAndGoFailureReason

CompatibleMediaNotFound

InvalidPassword

class RsIrc5Controller

IApiControllerConnection ControllerConnection

class ScreenshotOptions

AdvancedLighting

HideMarkups

NoShadows

SimpleLighting

TransparentBackground

class Simulator

static Int32 SimulationStepTimeout

static event CancelEventHandler Starting

class Station

Station ActiveStation

ClipPlaneCollection ClipPlanes

LightCollection Lights

ParticleSystemCollection ParticleSystems

PointCloudCollection PointClouds

ProjectObjectTagCollection Tags

class Texture

static Texture CreateCubeMap(Bitmap[])

IntPtr NativeTexture

Size Size

class UndoContext

Boolean ExecuteUndoStep(String, Action)

Boolean ExecuteUndoStep(String, Action,
Action<Exception>)

3.1.5 ABB.Robotics.RobotStudio.Math

Extended types

```
class BoundingBox
```

```
BoundingBox Intersect(BoundingBox)
```

```
Boolean Intersects(Ray)
```

```
Boolean IsValid()
```

4 Late-breaking information

4.1 Overview

This section contains late-breaking information that will be included in the appropriate documents in the subsequent releases.

5 Corrections

Overview

This section describes the defects solved in RobotStudio SDK.

5.1 Corrections in RobotStudio 5.60

6 Known Limitations

6.1 Development Environment

The RobotStudio API is not thread safe

Access to the RobotStudio API is not inherently thread safe. Only access the API from the thread that your Add-In was called from by RobotStudio. If multiple threads manipulate the object model it can be left in an inconsistent state. [Installation Information](#)

7 Installation information

7.1 Installing RobotStudio SDK 5.60

Installation instructions

RobotStudio SDK 5.60 will be installed side-by-side with any previous major version of RobotStudio SDK 5.xx, while minor versions within a release will update to the latest one.

The Visual Studio templates are not installed side by side. Only the latest installed version will be available.

How to install RobotStudio SDK on a PC

	Action
1	Browse to http://developercenter.robotstudio.com/ and select RobotStudio and then Download.
2	Download the file RobotStudioSDK5.60.exe to a folder on your computer.
3	Double click RobotStudioSDK5.60.exe to extract the installation files.
4	Double click setup.exe in the extracted folder to start the installation procedure.

8 Compatibility

The APIs in the RobotStudio SDK 5.60 are backwards compatible.