

Deployment of RobotSDK_v3.0

Mengwen He

2014-09-02

RobotSDK_v3.0 Package



Doc



Src



Install.bat



Kernel.bat



Kernel.sh



Tools.bat



Tools.sh

Doc	Documentation from Doxygen
Src	Source Code
Install	Batch to Install Tools and Kernel
Kernel	Batch/Shell to Build Kernel
Tools	Batch/Shell to Build Tools

Prerequisite

Windows

- VS 2005~2013
- Qt 5 *
- qt-vs-addin **
- Boost ***

Linux

- Qt Creator
- Qt 5
- Boost

* Qt's version should be consistent with VS.

* <http://qt-project.org/downloads> or you could compile Qt as [appendix A](#)

** qt-vs-addin should support Qt 5

** <http://qt-project.org/downloads>

*** Boost's version should be consistent with VS.

*** <http://sourceforge.net/projects/boost/files/boost/>

Deployment on Windows

Visual Studio Command Prompt

- A tool of Visual Studio
 - x64 version will use x64 compiler
 - x86 version will use x86 compiler
- Located in Visual Studio menu
- Used to compile Tools and Kernel
- Please start with administrator



1. Build Tools (1/2)

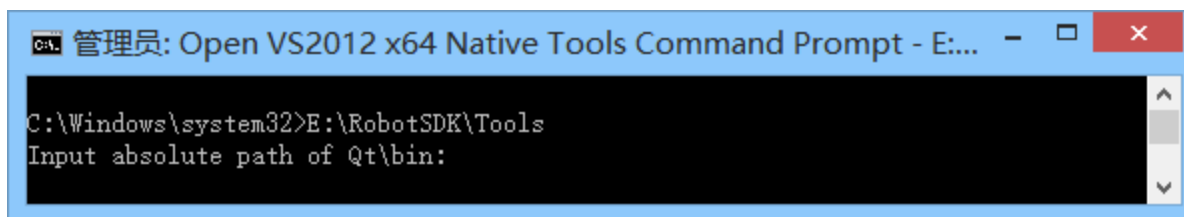
1. Start VS command prompt **with administrator**
2. Run following commands
 - (assume RobotSDK_v3.0 package is in E:\RobotSDK)
 - **cd /D E:\RobotSDK**
 - (assume Qt is installed in D:\SDK\Qt)
 - **Tools D:\SDK\Qt\bin**

Or directly run following commands

- **E:\RobotSDK\Tools D:\SDK\Qt\bin**

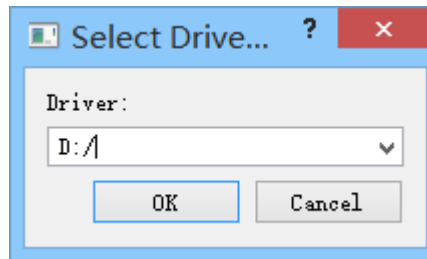
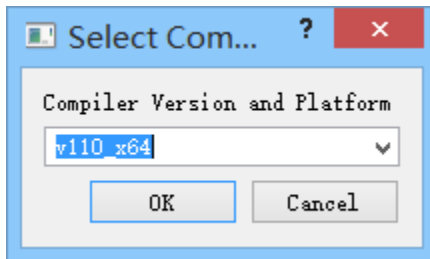
Or without parameter

- **E:\RobotSDK\Tools**
- Then it **may** require to specify the Qt\bin folder as below



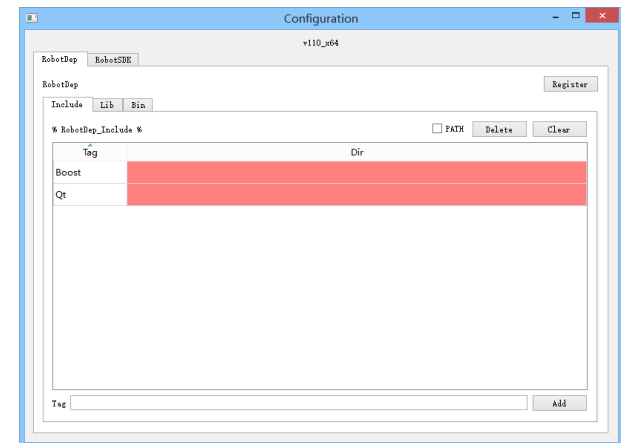
1. Build Tools (2/2)

3. Start to compile Tools as right
4. Automatically launch ConfigSystem.exe
 - Select Compiler Version and Platform ([appendix B](#))
 - Select Driver to install RobotSDK
 - (assume select D, it will be in D:\SDK\RobotSDK)












```
管理员: Open VS2012 x64 Native Tools Command Prompt - T...
Copyright (C) Microsoft Corp. All rights reserved.
1> Build started: Project: ConfigSystem, Configuration: Release x64
2> Build started: Project: ConfigModule, Configuration: Release x64
3> Build started: Project: ConfigProject, Configuration: Release x64
1> MOC ..\..\..\Src\Tools\ConfigSystem\configuration.h
2> MOC ..\..\..\Src\Tools\ConfigProject\ConfigProject.h
3> MOC ..\..\..\Src\Tools\ConfigModule\scaninterfacefunction.h
1> MOC ..\..\..\Src\Tools\ConfigSystem\registerdirwidget.h
2> MOC ..\..\..\Src\Tools\ConfigProject\ConfigProject.ui
3> MOC ..\..\..\Src\Tools\ConfigModule\scaninterfacefunction.ui
1> MOC ..\..\..\Src\Tools\ConfigSystem\registerdirwidgetitem.h
2> configuration.cpp
3> moc_configproject.cpp
1> main.cpp
2> MOC ..\..\..\Src\Tools\ConfigSystem\configuration.ui
3> MOC ..\..\..\Src\Tools\ConfigSystem\mainwindow.ui
1> MOC ..\..\..\Src\Tools\ConfigSystem\projectsetting.ui
2> MOC ..\..\..\Src\Tools\ConfigSystem\registerdirwidget.ui
3> MOC ..\..\..\Src\Tools\ConfigSystem\registerdirwidgetitem.ui
1> configuration.cpp
2> ..\..\..\Src\Tools\ConfigProject\ConfigProject.cpp(35): warning C4996: 'getenv' : This function or variable may be unsafe. Consider using '_dupenv_s' instead. To disable deprecation, use '_CRT_SECURE_NO_WARNINGS'. See online help for details.
3> D:\Program Files (x86)\Microsoft Visual Studio 11.0\VC\include\stdlib.h(449) : see declaration of 'getenv'
2> ..\..\..\Src\Tools\ConfigProject\ConfigProject.cpp(39): warning C4996: 'getenv' : This function or variable may be unsafe. Consider using '_dupenv_s' instead. To disable deprecation, use '_CRT_SECURE_NO_WARNINGS'. See online help for details.
3> D:\Program Files (x86)\Microsoft Visual Studio 11.0\VC\include\stdlib.h(449) : see declaration of 'getenv'
2> main.cpp
3> moc_configproject.cpp
1> main.cpp
2> ..\..\..\Src\Tools\ConfigModule\scaninterfacefunction.cpp(192): warning C4996: 'getenv' : This function or variable may be unsafe. Consider using '_dupenv_s' instead. To disable deprecation, use '_CRT_SECURE_NO_WARNINGS'. See online help for details.
3> D:\Program Files (x86)\Microsoft Visual Studio 11.0\VC\include\stdlib.h(449) : see declaration of 'getenv'
2> ..\..\..\Src\Tools\ConfigModule\scaninterfacefunction.cpp(204): warning C4996: 'getenv' : This function or variable may be unsafe. Consider using '_dupenv_s' instead. To disable deprecation, use '_CRT_SECURE_NO_WARNINGS'. See online help for details.
3> D:\Program Files (x86)\Microsoft Visual Studio 11.0\VC\include\stdlib.h(449) : see declaration of 'getenv'
2> ..\..\..\Src\Tools\ConfigModule\scaninterfacefunction.cpp(1221): warning C4996: 'getenv' : This function or variable may be unsafe. Consider using '_dupenv_s' instead. To disable deprecation, use '_CRT_SECURE_NO_WARNINGS'. See online help for details.
3> D:\Program Files (x86)\Microsoft Visual Studio 11.0\VC\include\stdlib.h(449) : see declaration of 'getenv'
2> moc_scaninterfacefunction.cpp
3> Generating Code...
3> ConfigProject.vcxproj -> E:\RobotSDK\VS\Tools\ConfigProject\..\..\Build\Tools\ConfigProject.exe
2> Generating Code...
1> registerdirwidget.cpp
2> registerdirwidgetitem.cpp
3> ConfigModule.vcxproj -> E:\RobotSDK\VS\Tools\ConfigModule\..\..\Build\Tools\ConfigModule.exe
```

5. Start to configure system environment
 - On RobotDep Tag, Boost and Qt must be set
 - On RobotSDK Tag, it could not be modified
 - Press “Register” button on both tags
 - Confirm everything is registered
 - Close ConfigSystem.exe



1. Tools Building Completed

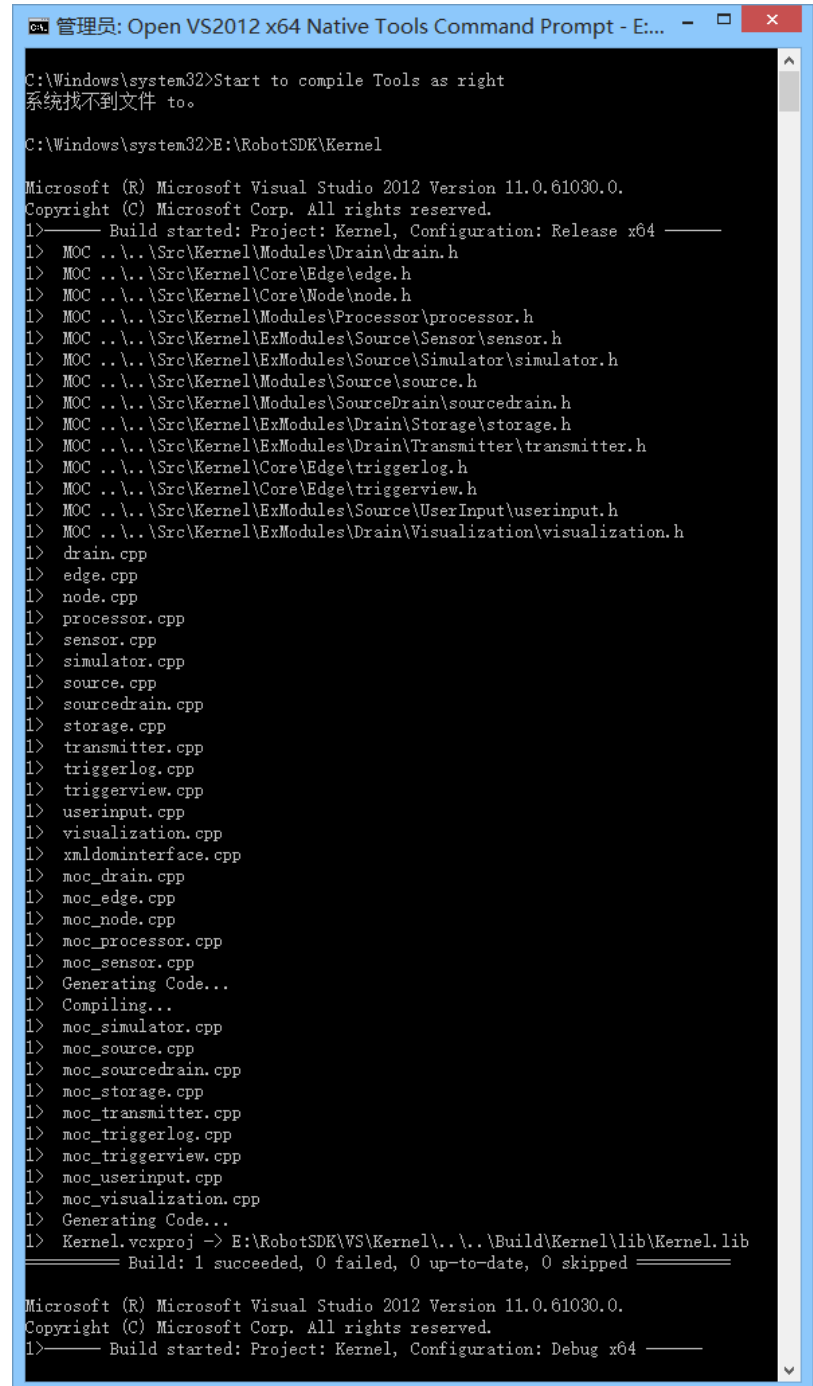
 Build
 Doc
 Src
 VS
 Install.bat
 Kernel.bat
 Kernel.sh
 Tools.bat
 Tools.sh

Build	Output Directory of Tools Applications: <ul style="list-style-type: none">• ConfigSystem.exe• ConfigModule.exe• ConfigProject.exe
VS	Output Directory of VS Project: <ul style="list-style-type: none">• Tools.sln<ul style="list-style-type: none">• ConfigSystem.vcxproj• ConfigModule.vcxproj• ConfigProject.vcxproj

- ConfigSystem.exe could finish configuration for **one time**
- After configuring system environment, **reboot is not necessary**
- Tools Building is the prerequisite of Kernel Building

2. Build Kernel

1. Start VS command prompt **with administrator**
 2. Run following commands
 - **cd /D E:\RobotSDK**
 - **Kernel**
- Or directly run following commands
- **E:\RobotSDK\Kernel**
3. Start to compile Tools as right












```
C:\Windows\system32>Start to compile Tools as right
系统找不到文件 to.

C:\Windows\system32>E:\RobotSDK\Kernel

Microsoft (R) Microsoft Visual Studio 2012 Version 11.0.61030.0.
Copyright (C) Microsoft Corp. All rights reserved.
1>----- Build started: Project: Kernel, Configuration: Release x64 -----
1> MOC ..\..\Src\Kernel\Modules\Drain\drain.h
1> MOC ..\..\Src\Kernel\Core\Edge\edge.h
1> MOC ..\..\Src\Kernel\Core\Node\node.h
1> MOC ..\..\Src\Kernel\Modules\Processor\processor.h
1> MOC ..\..\Src\Kernel\ExModules\Source\Sensor\sensor.h
1> MOC ..\..\Src\Kernel\ExModules\Source\Simulator\simulator.h
1> MOC ..\..\Src\Kernel\Modules\Source\source.h
1> MOC ..\..\Src\Kernel\Modules\SourceDrain\sourcedrain.h
1> MOC ..\..\Src\Kernel\ExModules\Drain\Storage\storage.h
1> MOC ..\..\Src\Kernel\ExModules\Drain\Transmitter\transmitter.h
1> MOC ..\..\Src\Kernel\Core\Edge\triggerlog.h
1> MOC ..\..\Src\Kernel\Core\Edge\triggerview.h
1> MOC ..\..\Src\Kernel\ExModules\Source\UserInput\userinput.h
1> MOC ..\..\Src\Kernel\ExModules\Drain\Visualization\visualization.h
1> drain.cpp
1> edge.cpp
1> node.cpp
1> processor.cpp
1> sensor.cpp
1> simulator.cpp
1> source.cpp
1> sourcedrain.cpp
1> storage.cpp
1> transmitter.cpp
1> triggerlog.cpp
1> triggerview.cpp
1> userinput.cpp
1> visualization.cpp
1> xmldominterface.cpp
1> moc_drain.cpp
1> moc_edge.cpp
1> moc_node.cpp
1> moc_processor.cpp
1> moc_sensor.cpp
1> Generating Code...
1> Compiling...
1> moc_simulator.cpp
1> moc_source.cpp
1> moc_sourcedrain.cpp
1> moc_storage.cpp
1> moc_transmitter.cpp
1> moc_triggerlog.cpp
1> moc_triggerview.cpp
1> moc_userinput.cpp
1> moc_visualization.cpp
1> Generating Code...
1> Kernel.vcxproj -> E:\RobotSDK\VS\Kernel\..\Build\Kernel\lib\Kernel.lib
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped =====

Microsoft (R) Microsoft Visual Studio 2012 Version 11.0.61030.0.
Copyright (C) Microsoft Corp. All rights reserved.
1>----- Build started: Project: Kernel, Configuration: Debug x64 -----
```

2. Kernel Building Completed

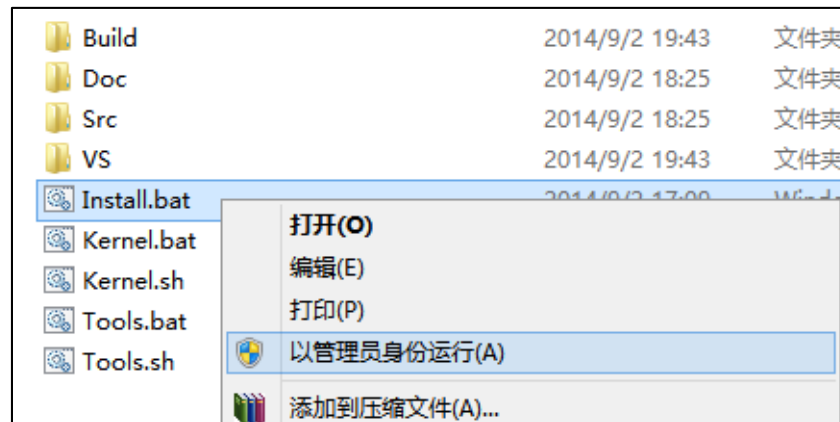
-  Build
-  Doc
-  Src
-  VS
-  Install.bat
-  Kernel.bat
-  Kernel.sh
-  Tools.bat
-  Tools.sh

Build	Output Directory of Kernel: <ul style="list-style-type: none">• include<ul style="list-style-type: none">• Kernel headers• CreateRule.xml• Interface_Functions.xml• RobotSDK_Global.h• lib<ul style="list-style-type: none">• Kernel.lib• Kernel_Debug.lib
VS	Output Directory of VS Project: <ul style="list-style-type: none">• Kernel.vcxproj

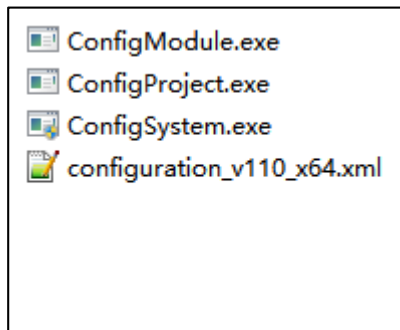
- Original Kernel's static link libraries are compressed into one Kernel.lib
- Support VS 2005 to 2013

3. Install

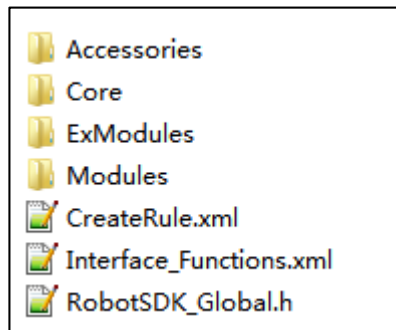
1. Start Install.bat **with administrator** as below



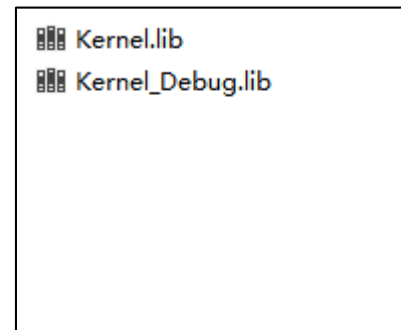
2. Everything will be installed to D:\SDK\RobotSDK



Tools



Kernel\include



Kernel\lib

Appendix

Appendix A

1. Install **ActivePerl** and **python**
2. Add python's directory to PATH
3. Start VS command prompt **with administrator**
4. Create **ConfigQt.bat*** in the folder of Qt's source code
5. Run ConfigQt.bat
6. Run nmake
7. Run nmake installs

* ConfigQt.bat see next page.

Appendix A (ConfigQt.bat)

@echo off

if "%1"==" " cd /D %~dp0

if not "%1"==" " cd /D %1

set /p driver=Set Disk Driver for Qt Installation (default: c):

if "%driver%"==" " (set driver=c)

set QTDIR=%driver%:\SDK\Qt

set PATH=%PATH%;%QTDIR%\bin

set /p vsversion=Set VS Version (default: 2012):

if "%vsversion%"==" " (set vsversion=2012)

set QMAKESPEC=win32-msvc%vsversion%

configure -prefix %QTDIR% -debug-and-release -opensource -opengl desktop -nomake examples -nomake tests

pause

@echo on

Appendix B

VS Version	Compiler Version	_MSC_VER
2013	v120	1800
2012	v110	1700
2010	v100	1600
2008	v90	1500
2005	v80	1400