



Product:	SNMPManager
Product version:	V2.7
Document ID:	PM-SNMPManager
Doc revision:	A
Written/Appr.:	OH / SL
Date:	14. Nov. 2008

Industrial Control Design AS



SNMPManager V2.7

Programmers Manual

The content of this document is confidential information not to be published without the consent of Industrial Control Design AS.

Industrial Control Design AS, www.icd.no, support@icd.no, forum.icd.no

Contents

1. INTRODUCTION.....	3
1.1. About.....	3
1.2. Terms and Definitions.....	3
<hr/>	
2. INSTALLATION.....	5
2.1. Prerequisites.....	5
2.2. Quick setup of a project utilizing the SNMPManager library.....	5
2.3. Modify the project xml files.....	10
2.4. Install on a Controller.....	10

1. Introduction

1.1. About

This document describes how the SNMPManager CDP component works, and how to set it up and use it with the CDP system. The SNMPManager CDP component has the following features:

- The SNMPManager is able to send SNMP Set and Get requests to several agents (SNMP v. 1 and 2).
- Get requests can be sent regularly, specified by two different intervals.
- Responses from agents can be used to update CDP Signals.
- It can receive two external messages, SetSNMPvalue and GetSNMPvalue, and send back an external message GetSNMPvalueReply.
- It is able to receive SNMP Trap messages sent from several sources (SNMP v. 1 and 2).
- It is able to decode the received Trap messages, and generate and Set readable Alarm-messages with updated Description-field based on the received trap. These Alarm-messages can be read in a standard web-browser or in the CDP Browser. The messages can also be logged.
- The received traps can also be specified to either Set or Clear a unique Alarm. Such unique Alarms are auto-created based on specified Name, AgentNumbers and optional Values. They can also optional be routed to CDP Signals by SignalOutSet.
- Configuration of the component is done in the component's XML-file (SNMPManager.xml).

1.2. Terms and Definitions

Agent

Software that enables a device to respond to manager requests to view or update MIB data, and send traps reporting problems or significant events.

BER

Basic Encoding Rules - A set of rules for translating ASN.1 values into a stream of octets to be transmitted across a network.

CDP

Control Design Platform.

CDP Controller

Computer (Usually an industrial computer) running CDP application, usually on a true real- time operating system.

CDPUI

The CDP graphical user interface.

Component

Object with strict interface specification.

MIB

Management Information Base. A logical database made up of the configuration, status and statistical information stored at a device. MIB-files are readable text-files.

OID

Object Identifier, a string of numbers derived from a global naming tree, used to identify an object.
E.g. :”1.3.6.1.4.1.16177.1.1.9.1”

SNMP

Simple Network Management Protocol. A protocol using UDP that enables a management station to configure, monitor, and receive trap (alarm) messages from network devices

2. Installation

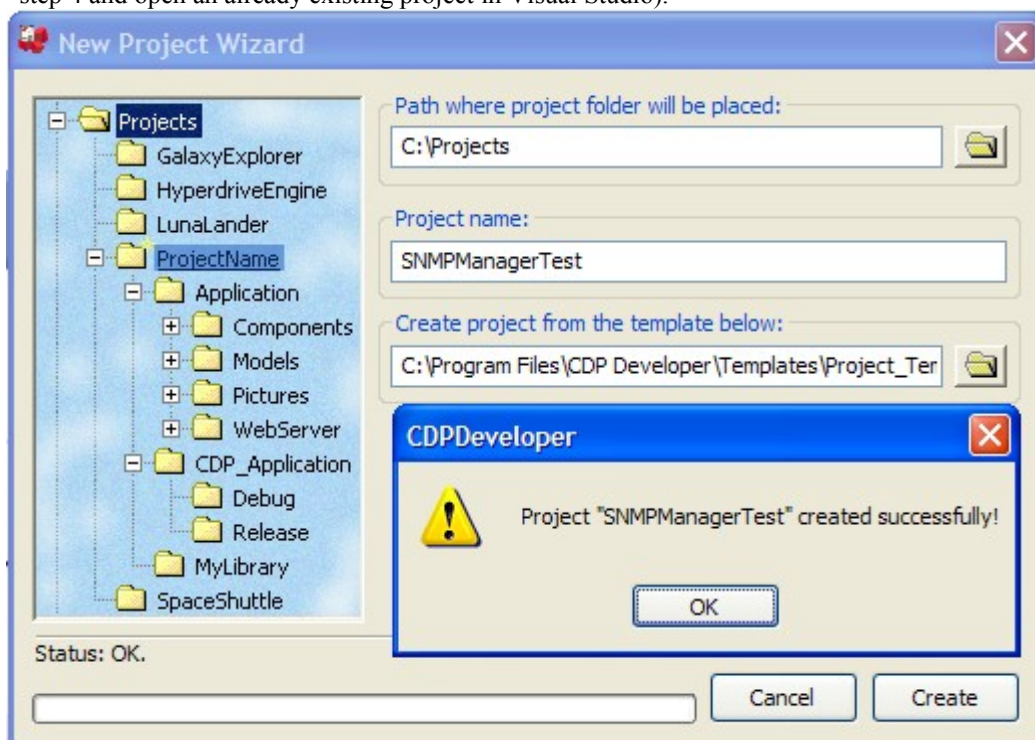
The SNMP Manager can be delivered both as source code and as a separate library which is linked into the application. If delivered as separate library, the SNMPManagerLib Setup will by default install all files in sub-folders of “CDP Developer”.

2.1. Prerequisites

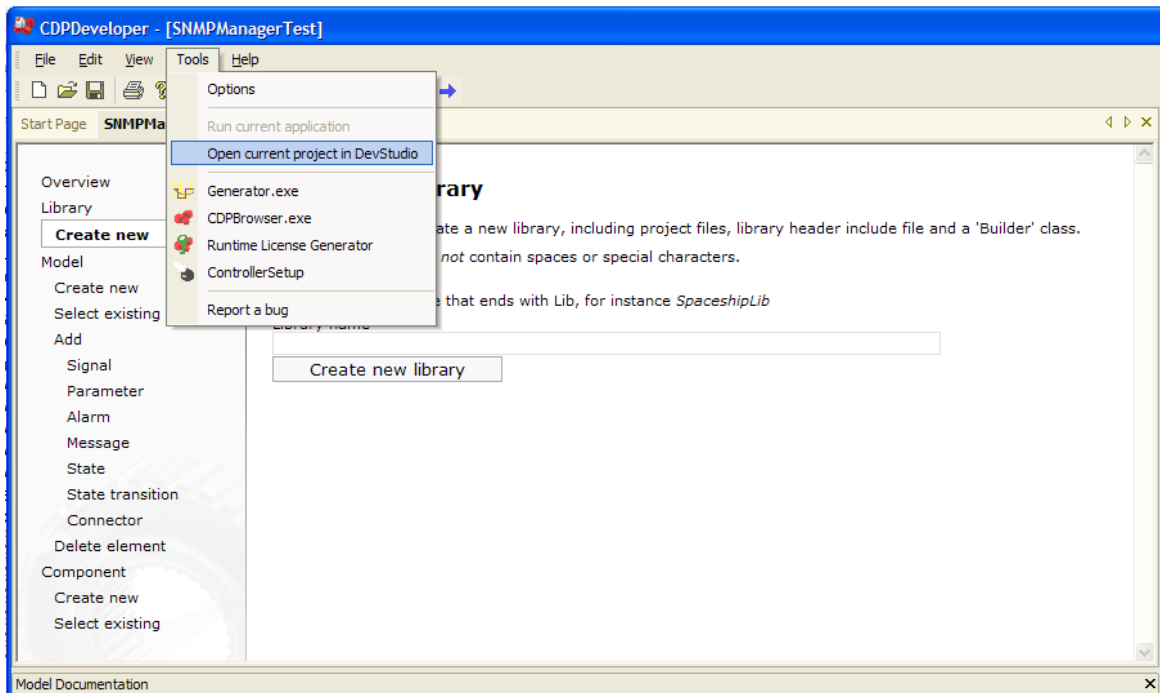
- A valid CDP license
- Familiar with CDP
- CDP version: 2.3.1.0
- OS: “Windows” or “RTOS version 5.12” or “Linux (based on gcc-4.1.1 and glibc-2.3.6 or glibc-2.6.1)”.

2.2. Quick setup of a project utilizing the SNMPManager library

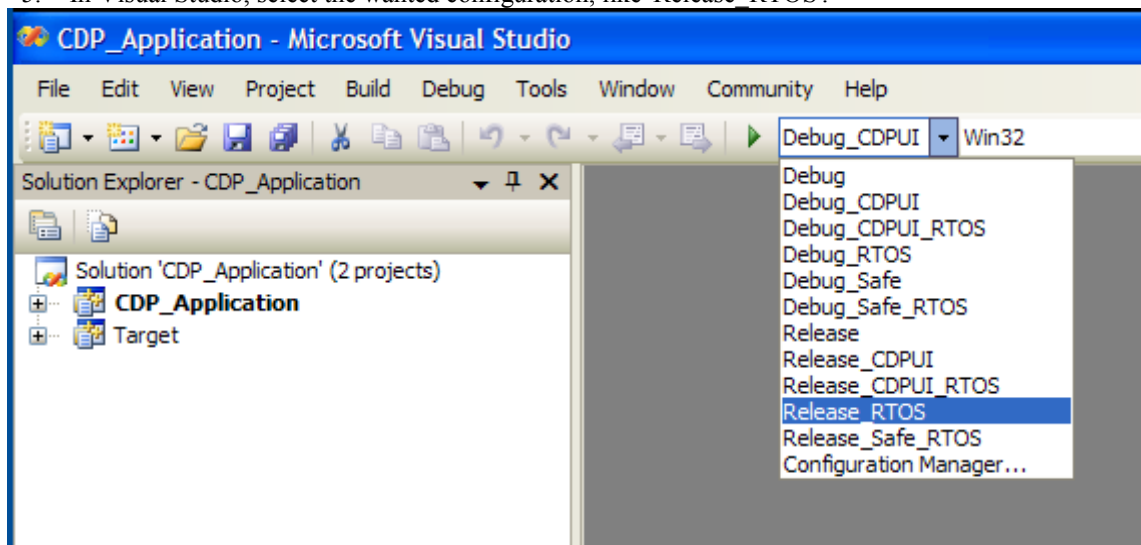
1. Start CDP Developer
2. Make a new project by selecting File->'New Project', type in project name and click Create (or skip to step 4 and open an already existing project in Visual Studio).



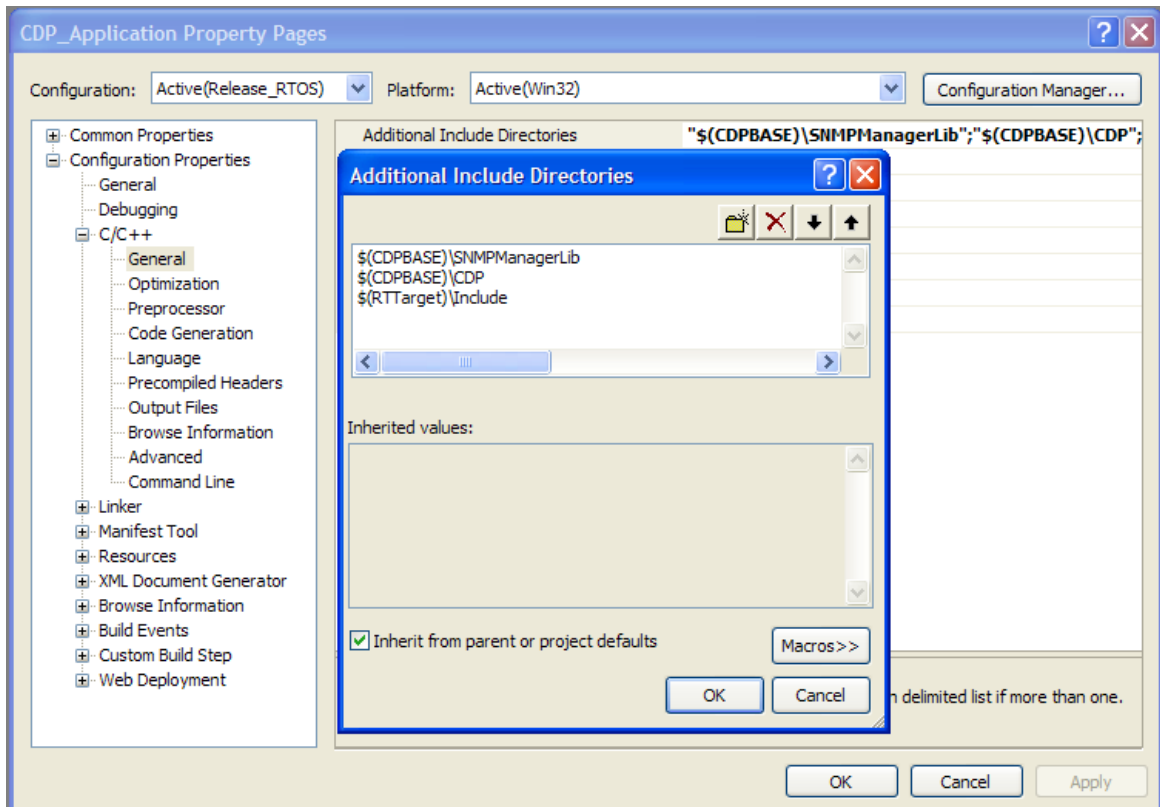
3. Choose Tools->'Open current project in DevStudio':



4. If Visual Studio asks to convert the project, accept this by selecting 'Next'/'Finish'/'Close' until done. Close the conversion report.
5. In Visual Studio, select the wanted configuration, like 'Release_RTOS'.

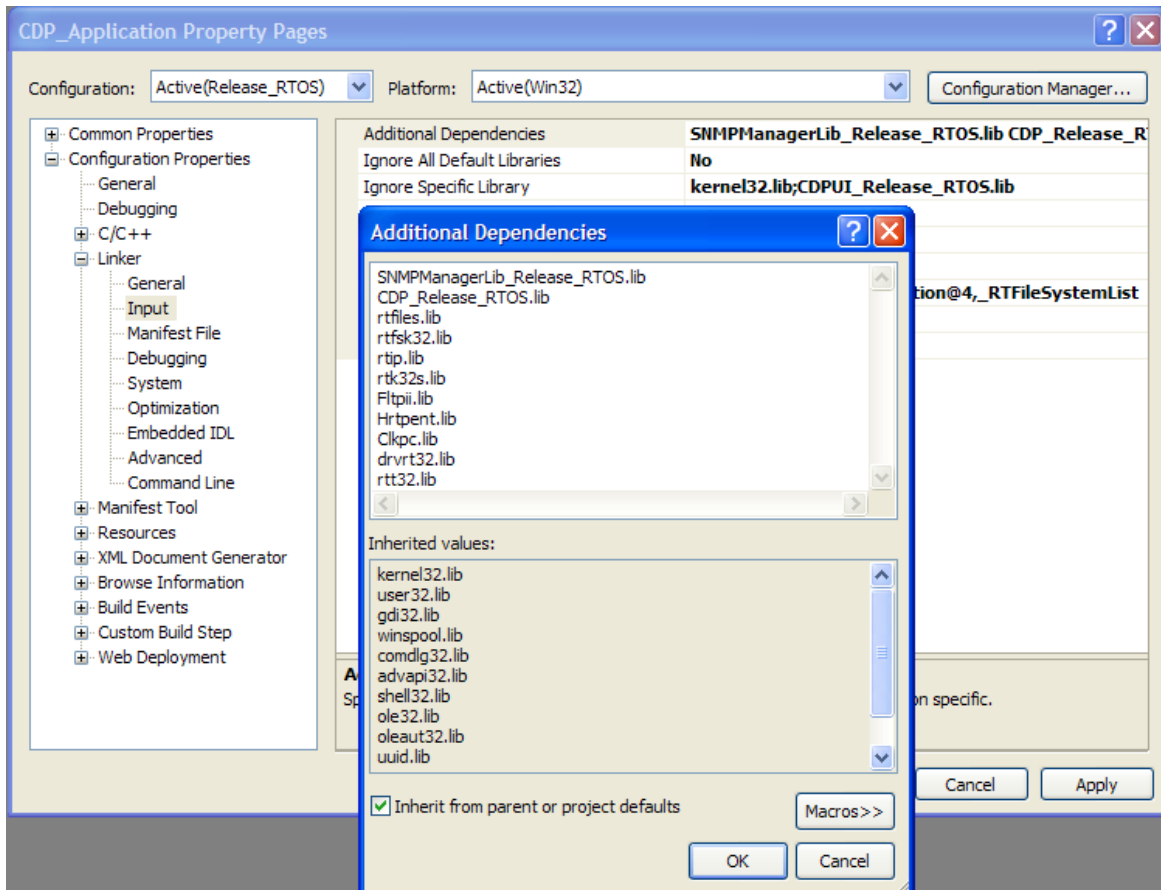


6. Select 'CDP_Application' from the 'Solution Explorer', right-click and select Properties.
7. In C++/Additional Include Directories, make sure it says:
"\$\$(CDPBASE)\SNMPManagerLib";"\$(CDPBASE)\CDP";"\$(RTTarget)\Include"

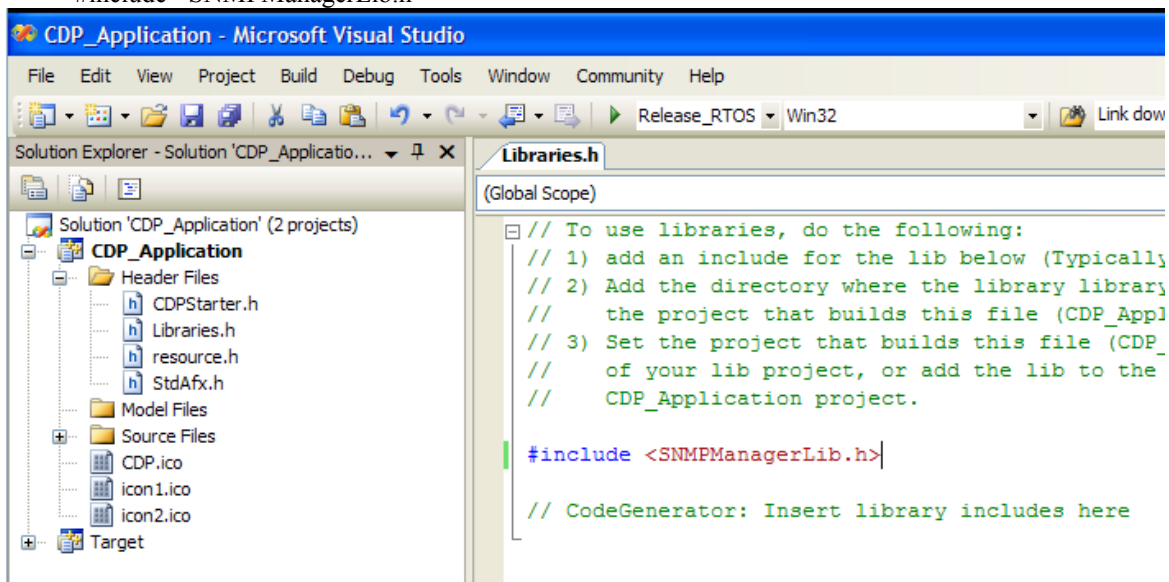


The compiler will look in the directories specified in 'Additional Include Directories' for files that you #include in your .cpp and .h files. If you get an error 'Can not open include file...', then it is most likely caused by a missing include directory, or that the file you #include does not exist.

8. In Linker->Input, make sure you list SNMPManagerLib_Release_RTOS.lib CDP_Release_RTOS.lib in addition to all the on-time rtos32 libraries you need. This will ensure that the linker finds all the functions that are referenced in the code.



9. In the 'Solution Explorer', inside the 'CDP_Application' project, locate the Header file 'Libraries.h', and add the line
`#include <SNMPManagerLib.h>`



10. If there is no 'Target' project in your 'Solution Explorer', select 'Solution'->'Add'->'Existing Project'. Choose the 'Target.vcproj' from the same folder in which your 'CDP_Application.vcproj' is at.
11. Right-Click Solution, select 'Configuration Manager...'. Make sure that all configurations have 'Release_RTOS' selected in 'Configuration':

2.3. Modify the project xml files

Make sure that the IPAddress and SubnetMask in the 'NetworkInterface'-section are configured to contain an unique IPAddress and correct subnet, e.g.:

```
<NetworkInterface Name="ETH0" MAC="" IPAddress="10.0.2.30" SubnetMask="255.255.255.0"></NetworkInterface>
```

Add the following to your project's Application.xml:

Inside the <Components> element, add an instance of an SNMPManager component, for instance:

```
<Component Name="SNMPManager" src="Components/SNMPManager.xml"></Component>
```

Or, inside the <Subcomponents> element, add:

```
<Subcomponent Name="SNMPManager" Model="SNMPManager" src="Components/SNMPManager.xml"></Subcomponent>
```

This will tell CDP to initialize a component named "SNMPManager" from a component file located at "Components\SNMPManager.xml". Make sure that your Models\ folder contains an SNMPManager.xml model file, or the component will not be initialized correctly.

The SNMPManager Model file can be found in \$(CDPBase)\Templates\Project_template\Application\Models\, and an example SNMPManager component file can be found in \$(CDPBase)\Templates\Project_template\Application\Components\.

2.4. Install on a Controller

Assuming you have a controller with PXE boot and a disk, you can use ControllerSetup.exe to download a bare-bone CDP application to the controller. See the document 'ControllerSetup.pdf' in the Doc folder where you installed CDP for more information on setting up a controller.

When the Controller has a CDP Application installed, you can use CDPFileManager.exe to upload files to the controller. If you have built for RTOS, upload these files to the root of the controller disk.

- CDP_Application\Release_RTOS\CDP.rtb.
- Everything inside the Application folder.

Make sure that you also upload rttboot.com from your "\$(RTTarget)\bin" folder to the root folder of the controller. This ensures that the RTOS32 loader is the most recent version