CodeMeter Tutorial

CODESYS Control – Tutorial

Version 1.0

Content

1 Introduction

The product CodeMeter from WIBU Systems can be used for the following usages:

- Licensing: For single licensing of features
- Security:
  - For protecting software (encryption)
  - Encrypting/decrypting data
  - Storing secret data (like private keys of X.509 certificates)

CodeMeter is available in two different solutions, which can be used in parallel:

1. CodeMeter software solution
2. CodeMeter hardware solution

The support for both solutions is fully integrated in CODESYS and the Runtime System. In the Runtime System, this is implemented in the component CmpCodeMeter.

The hardware and software CodeMeter support can be realized in two different ways dependening on the used operating system:

1. CodeMeter Runtime:

   For Windows, Linux or OS X based systems, the CodeMeter Runtime is available as a separate installer and is running as a separate process. The Runtime System supports a CodeMeter Runtime by default.

2. CodeMeter Embedded:

   Alternatively you can use the CodeMeter solution on any system, on which the Runtime System is ported. This is called CodeMeter Embedded. To support CodeMeter Embedded, you have to set the following compiler switch when compiling the Runtime System:

   ```
   RTS_CODEMETER_EMBEDDED
   ```

   The CodeMeter hardware solution on CodeMeter Embedded is only supported for the following operating systems:

   - Windows
   - Windows CE
   - Linux
   - VxWorks

   See Mass Storage Communication for restrictions using the CodeMeter hardware solution on these operatins systems!

2 Accessing the CodeMeter features

Access to all CodeMeter features in the Runtime System you get via the CmpCodeMeter Interface (see RuntimeSystemDocumentation/Reference/CmpCodeMeterItf).

Access to all CodeMeter features in the IEC-Application you get via the CmpCodeMeter.library.

3 CodeMeter software solution

The SoftLicensing solution CmActLicense is based on encrypted files which are bound on controller specific attributes (like serialnumber, processorID or similar). This is used to store all secret information and to store the license entries. It is fully intergrated in the CODESYS Runtime System.

To activate this feature, you have to set the compiler switch compiling the Runtime System:

```
RTS_CODEMETER_SOFTLICENSING
```

As mentioned above, the software solution must comply 2 requirements:

REQ 1. Unique hardware attribute:

The solution must be bounded on a secure hardware attribute. This hardware attribute must be robust against modifiying and so it must be derived from the hardware (like a build in serial number, unique processorID, etc.).

This attribute is requested in side the Runtime System from the following SysTargetItf function (see RuntimeSystemDocumentation/Reference/SysTargetItf):

```
RTS_RESULT CDECL SysTargetGetSerialNumber_Secure(char **ppszSerialNumber, RTS_I32 *pnMaxLen);
```

REQ 2. Integrity of the Runtime System:

The Runtime System must be protected against compromizing the CodeMeter component/interface and the external library functions that provides the access to these security functions!

Both requirements must be proved in a FMEA analysis process in cooperation with 3S-Smart Software Solutions GmbH! So please contact the sales team for details, if you intend to integrate the CodeMeter software solution into your controller!

3.1 Creating an empty SoftLicense container

To create an empty SoftLicense license container, you need a so called wbb file. This wbb file contains the firmcode of the license vendor (see Licensing).

3S-Smart Software Solutions GmbH provides a wbb file to install an empty license container for 3S licenses.

The empty container will be bound on the secure serial number (see above Unique hardware attribute) at the first license installation operation! After this, the container is bould on this target resp the Unique hardware attribute.

4 CodeMeter hardware solution

There are several devices named CmDongle provided by WIBU, in which the CodeMeter solution is integrated. The devices are based on a smart card chip technology. All of these devices can be used to store secret data, licenses and to encrypt/decrypt data.

The CmDongle is available in the following different designs:

- USB-Dongle
- Memory-cards
- ASIC

4.1 Supported Features

1. Each CmDongle offers 328 kByte of memory for your licenses, which can accommodate up to 2,000 licenses with different license models. Each of these licenses may be a multi-user license with thousands of concurrent seats.
2. The CodeMeter devices supports the following crypto algorithms:
   - 128 bit AES
   - SHA-256
   - 2048 bit RSA
   - 224 bit ECC
3. The used SmartCard chip contains a random number generator according to FIPS140-1 and is certified according to EAL 5+ including a certified crypto library.

4.2 USB-Dongle

WIBU CodeMeter USB-dongles are available without or with flash memory (can be used to store files on it).

4.2.1 Variants accessing USB-dongles

1. Accessing via Mass Storage Communication (MSC)

   The USB-dongles are typically accessed from the Runtime System via the filesystem of the operating system (Mass Storage

Communication). This has a problem at operating system that uses a file chache for accessing files!

Reason:

Under these operating systems (Linux, VxWorks), you have to switch off globally the file cache to get the dongle to operate correctly, but this will reduce the file access times dramatically!! This is because you write an encrypted files on the USB-dongle and get back unencrypted files decrypted by the dongle. But this doen't work if you have a file cache. In this case you get back the last written encrypted file and not the unencrypted file!

Under Windows the file cache can be disabled per file, so here we have no problem. But for example under Linux and VxWorks the USB-dongles cannot be used this way! For this you have to use the Human Interface Device.

2. Accessing via Human Interface Device (HID)

   Newer CodeMeter dongles can be used by default as a HID (Human Interface Device). This is supported on the following operating systems within the Runtime System:

   - Windows
   - Windows CE
   - Linux
   - VxWorks

4.3 Memory-cards

There are different types of WIBU CodeMeter memory cards available:

- SD/microSD cards
- CF/CFast cards

4.4 ASIC

WIBU CodeMeter ASICs can be used to integrate directly on the controller board design. All features are available.

5 Operating system specific issues

On some operating systems there are special issues or jobs that must be done by the OEM customer to support CodeMeter correctly.

5.1 VxWorks

Requirements for the VxWorks Kernel Image (–> VIP Kernel Configuration):

- INCLUDE_USB –> USB Common Stack
- INCLUDE_USB_INIT –> USB Common Stack Initialization
- USB Host Controller:
  - INCLUDE_EHCI
  - INCLUDE_EHCI_INIT
- INCLUDE_HCD_BUS –> USB Host Controller Start
- FOLDER_USB_GEN2_DEVICES –> USB Generation 2 Class Device Drivers

  MANDATORY!!!: Does not work for VxWorks USB Generation 1 (some API functions missing)
- FOLDER_USB_GEN2_DEVICES_INIT
- INCLUDE_USB_HOST_CLASS_INIT –> USB Host Class Driver Initialization
- INCLUDE_POSIX_PTHREADS –> Support for POSIX Threads in the kernel

- INCLUDE_POSIX_SEM: Support for POSIX Semaphores in the kernel

5.2 Windows CE

Under Windows CE you need for the Human Interface Device (HID) dongle support the WibuCmHID.dll, which is part of the Runtime System delivery since v3.5.9.0. This Dll must be available in the Windows directory on the controller to recognize a HID-Dongles during startup of the Runtime System.

Additionally you need the following entries in the registry:

```
[HKEY_LOCAL_MACHINE\Drivers\USB\LoadClients\1615\Default\3\WibuCmHID]
"Dll"="WibuCmHID.DLL"

[HKEY_LOCAL_MACHINE\Drivers\USB\ClientDrivers\WibuCmHID]
"Prefix"="WCH"
"Dll"="WibuCmHID.DLL"
```

6 Licensing

WIBU CodeMeter can be used for single licensing of features via CodeMeter License Models. The license can be stored on CodeMeter software solution and CodeMeter hardware solution. A License is bound per license vendor to a so called Firmcode. And each feature has its own ProductCode to check the corresponding license.

In CODESYS you can get access to the installed CodeMeter licenses via:

- the CmpCodeMeter interface in the Runtime System (see RuntimeSystemDocumentation/Reference/CmpCodeMeterItf)
- the CmpCodeMeter.library out of the IEC-Application

6.1 3S-Smart Software Solutions GmbH licenses

3S-Smart Software Solutions GmbH uses its own Firmcode to protect their licensed features.

In the CODESYS you can use the following build in operator to check 3S license features:

UDINT __CHECKLICENSE (DWORD productID, BOOL xDoChallenge);

productID is the ID of the feature. xDoChallenge enables/disables an integrity check of the underlying library. A check can consume some time on low performant embedded controllers.

The return value is -1 for a Demo license and 0 if the license is invalid. If the license is valid, the license value is returned.

6.2 Other vendor licenses

If another vendor (like OEM customer) would like to license own features, they have to contact WIBU Systems to get their own licensing infrastructure.

Bibliography

[1] CodeMeter http://www.wibu.com/codemeter.html
[2] CodeMeter Runtime http://www.wibu.com/codemeter/codemeter-runtime.html
[3] CmDongle http://www.wibu.com/codemeter/cmdongle.html
[4] CmActLicense http://www.wibu.com/codemeter/cmactlicense.html
[5] WIBU http://www.wibu.com

Version History

Version	Description	Author	Date
0.1	Creation	AH	24.11.2016
0.2	Review	JT	06.12.2016
1.0	Release	AH	06.12.2016