

Porting from KINGSTAR 4.x to MaxRT wRTOS™

USER GUIDE

IntervalZero

MaxRT
wRTOS

Porting from KINGSTAR 4.x to MaxRT wRTOS

IZ-DOC-MaxRT-wRTOS-0033-R1

Copyright © 2026 by IntervalZero, Inc. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means, graphic, electronic, or mechanical, including photocopying, and recording or by any information storage or retrieval system without the prior written permission of IntervalZero, Inc. unless such copying is expressly permitted by federal copyright law.

While every effort has been made to ensure the accuracy and completeness of all information in this document, IntervalZero, Inc. assumes no liability to any party for any loss or damage caused by errors or omissions or by statements of any kind in this document, its updates, supplements, or special editions, whether such errors, omissions, or statements result from negligence, accident, or any other cause. IntervalZero, Inc. further assumes no liability arising out of the application or use of any product or system described herein; nor any liability for incidental or consequential damages arising from the use of this document. IntervalZero, Inc. disclaims all warranties regarding the information contained herein, whether expressed, implied or statutory, including implied warranties of merchantability or fitness for a particular purpose.

IntervalZero, Inc. reserves the right to change this document and the products described herein without notice.

MaxRT wRTOS™ is a trademark of IntervalZero, Inc. Throughout this guide, wRTOS refers to MaxRT wRTOS™.

Microsoft is a registered trademark, and Windows 11 and Windows 10 are trademarks of Microsoft Corporation.

EtherCAT® is a registered trademark and patented technology licensed by Beckhoff Automation GmbH, Germany.

All other companies and product names may be trademarks or registered trademarks of their respective holders.

IntervalZero

200 Fifth Avenue, FL 6, STE 6020

Waltham, MA 02451

Phone: 781-996-4481

www.intervalzero.com

Contents

Porting from KINGSTAR 4.x to MaxRT wRTOS	1
Product Comparison	2
Product Packages	2
KINGSTAR EtherCAT Fieldbus Integration	5
Licensing, Activation, and Core Configuration	6
wRTOS Product Code	6
Activating Product Packages and Configuration	7
Dongle Activation Utilities	8
License Types	8
Configuring and Controlling the Runtime	10
Configuring and Controlling KINGSTAR	10
Configuring and Controlling wRTOS	10
Tool Comparison	11
Project Settings	13
SDK Code Changes	14
New APIs	14
New E-CAT APIs	14
New Managed E-CAT APIs	16
Breaking API Changes	18
Breaking Changes to Header and Library Files	18
Breaking Changes to Namespaces	18
Breaking Changes to E-CAT APIs	19
Breaking Changes to Managed E-CAT APIs	130
Removed APIs	227
Support	234
Contacting Technical Support by Phone	234

Porting from KINGSTAR 4.x to MaxRT wRTOS

In this guide, we highlight the major differences between KINGSTAR 4.x and MaxRT wRTOS and describe the steps required to port an existing KINGSTAR application to MaxRT wRTOS.

TOPICS:

- [Product Comparison](#)
- [Licensing, Activation, and Core Configuration](#)
- [Configuring and Controlling the Runtime](#)
- [Tool Comparison](#)
- [Project Settings](#)
- [SDK Code Changes](#)

1

Product Comparison

This chapter compares product components and describes KINGSTAR EtherCAT Fieldbus Integration.

Product Packages

KINGSTAR and MaxRT wRTOS components are available in these product packages:

KINGSTAR

- KINGSTAR Runtime includes these components:
 - KINGSTAR Fieldbus Runtime
 - RTX64 Runtime

These product packages require KINGSTAR Runtime and can be installed with KINGSTAR Runtime but must be purchased separately:

- KINGSTAR Feature packages
- KINGSTAR Motion packages
- KINGSTAR Axis packages
- SDK

MaxRT wRTOS

- wRTOS Runtime (can be installed silently) includes this component:
 - Network Link Layer (NL2, replaces the NAL in RTX64)
 - Virtual Network
 - Network Relay

These product packages and components require wRTOS Runtime and can be installed with wRTOS Runtime but must be purchased separately:

- Basic Networking includes these components:
 - TCP/IP Stack
 - GigE Vision
- Fieldbuses includes this component:
 - E-CAT
- SDK (can be installed silently)
- Runtime Merge Modules

This comparison table shows the correspondence between KINGSTAR and MaxRT wRTOS in terms of components and packages.

KINGSTAR

MaxRT wRTOS

KINGSTAR Fieldbus Runtime component

wRTOS Fieldbuses package (requires wRTOS Runtime and must be purchased separately)

KINGSTAR Fieldbus Runtime component includes the KINGSTAR Subsystem and KINGSTAR tools:

The E-CAT MainDevice and E-CAT tools are included in the wRTOS Fieldbuses package or wRTOS Runtime.

- KINGSTAR Subsystem (renamed to E-CAT MainDevice in MaxRT wRTOS)
- Configuration Tool (renamed to E-CAT Configuration Tool in MaxRT wRTOS)
- ESI Import Tool (renamed to E-CAT ESI Import Tool in MaxRT wRTOS)
- Control Panel
- Analysis Console (renamed to Message Viewer in MaxRT wRTOS)
- Scope

- E-CAT MainDevice (included in the wRTOS Fieldbuses package)
- E-CAT Configuration Tool (included in the wRTOS Fieldbuses package)
- E-CAT ESI Import Tool (included in the wRTOS Fieldbuses package)
- Control Panel (included in wRTOS Runtime)
- Message Viewer (included in wRTOS Runtime)
- Scope (not available in wRTOS 1.0.1)

RTX64 Runtime component

wRTOS Fieldbuses package

- Entry (2 cores)
- Basic (3 cores)
- Professional (7 cores)

SDK

SDK

KINGSTAR add-on packages

MaxRT wRTOS

Note: Requires KINGSTAR Runtime and must be purchased separately

KINGSTAR Feature packages:

- Cable Redundancy
- High Speed Timer
- Hot Connect
- Multiple Master
- PLC
- Vision

- These features require the wRTOS Fieldbuses package and must be purchased separately:
 - E-CAT Cable Redundancy
 - E-CAT High Speed Timer
 - E-CAT Hot Connect
 - E-CAT Multiple MainDevices
- PLC (not supported in wRTOS 1.0.1)
- wRTOS GigE Vision (requires the wRTOS Basic Networking package and must be purchased separately)

KINGSTAR Motion Packages (each motion package includes a Level 1 Axis package):

- Point to Point (PTP)
- Synchronization (includes PTP)
- Blending (includes PTP and Sync)
- Kinematics (includes PTP, Sync, and Blending)

wRTOS 1.0.1 does not support the Motion feature.

KINGSTAR Axis packages:

- Level 1 Axis (8 axes)
- Level 2 Axis (16 axes)
- Level 3 Axis (32 axes)
- Level 4 Axis (unlimited axes)

This is tied to the Motion feature, which is not supported in wRTOS 1.0.1.

KINGSTAR EtherCAT Fieldbus Integration

MaxRT wRTOS incorporates the KINGSTAR EtherCAT Fieldbus functionality in a wRTOS E-CAT component in the wRTOS Fieldbuses product package (see above).

The E-CAT component and features require licenses. See [wRTOS Product Codes](#) in this guide for more details.

Note: wRTOS 1.0.1 does not support OPC UA, KINGSTAR Motion, and PLC.

MaxRT wRTOS SDK includes E-CAT Fieldbus (RTECAT) functions and data types related to the wRTOS E-CAT Fieldbus used for EtherCAT communication.

- To build RTSS applications, include the header **RtecatApi.h** in your project and link with the library **RtecatApi.lib**.
- To build Windows applications, include the header **RtecatApi.h** in your project and link with the library **RtecatApi_W64.lib**.

Note: wRTOS 1.0.1 does not support APIs for KINGSTAR .NET Class, KINGSTAR COM and KINGSTAR Motion.

2

Licensing, Activation, and Core Configuration

This chapter outlines wRTOS product codes, licensing concepts, and activation and configuration methods in KINGSTAR and MaxRT wRTOS.

wRTOS Product Code

MaxRT wRTOS uses different product codes to identify product components and features, as listed below:

Product code	Description
WBLD64	Allows you to build applications. This feature does not include debugging tools.
WECR64	wRTOS E-CAT Cable Redundancy allows for the EtherCAT MainDevice to support Cable Redundancy.
WEHC64	wRTOS E-CAT Hot Connect allows the EtherCAT MainDevice to support Hot Connect.
WEHST64	wRTOS E-CAT High Speed Timer allows the EtherCAT MainDevice to use a High-Speed Timer.
WEMM64	wRTOS E-CAT Multiple MainDevices supports running multiple EtherCAT MainDevices. These require the wRTOS Fieldbus (WFBS64) component package.
WFBS64	wRTOS Fieldbus supports the use of the EtherCAT MainDevice and future fieldbuses.

Product code	Description
WNET64	wRTOS Basic Networking supports the use of TCP/IP Stack applications above the NL2.
WRTOS64	wRTOS Runtime includes the Real-time Subsystem and Network Link Layer (NL2), Network Relay, and Virtual Network. See Runtime Editions for a list of the wRTOS Runtime editions and the maximum number of Real-Time Subsystem (RTSS) cores they support.
WSDK64	wRTOS Software Development Kit (SDK) is used to develop, build, and debug applications.
WVIS64	wRTOS GigE Vision allows the use of GigE and GeniCam drivers. This component also requires the wRTOS Basic Networking (WNET64) component package.

Activating Product Packages and Configuration

Product packages must be activated with a valid license before you can use them. KINGSTAR and MaxRT wRTOS provide different activation options, as described below.

KINGSTAR Activation and Configuration

KINGSTAR Runtime includes the KINGSTAR Fieldbus and RTX64 cores. After installing the Runtime, you must activate additional product packages and configure the system. KINGSTAR provides these options for activating product packages:

- Online Activation:
 - After installing KINGSTAR Runtime, use RTX64 Activation and Configuration GUI utility immediately to activate product packages in the **Enter your activation key** field and assign processors between Windows and RTSS in the **Set the RTSS boot configuration** field.
 - Use KINGSTAR Control Panel to configure KINGSTAR Subsystem settings.
- Offline Activation: Use a license file.

For more information, see the KINGSTAR Help or *KINGSTAR Runtime Installation Guide*.

wRTOS Activation and Configuration

MaxRT wRTOS provides these options for activating the E-CAT component and E-CAT configuration:

- wRTOS Settings:
 - On the Licensing and Activation page, which appears immediately after MaxRT wRTOS installation, you can activate the E-CAT component and features and view license information.
 - On the Core Configuration page, you can view the total system cores available, set or change the RTSS configuration, and view the core assignments for wRTOS processes and the E-CAT component. When wRTOS Runtime is installed, you can assign system cores to Windows or wRTOS.
 - On the E-CAT page, you can configure settings for the EtherCAT Fieldbus.
- You can use the MaxRTActivationUtil.exe command line utility to silently activate the E-CAT component and configure system processors.

For more information, see the MaxRT wRTOS Help, *MaxRT wRTOS Installation Guide*, or *MaxRT wRTOS Deployment Guide*.

Dongle Activation Utilities

IntervalZero provides standalone dongle activation utilities for KINGSTAR and MaxRT wRTOS that allow you to separate the network activation process. Instead of activating dongles on the production floor, you can activate them at another location and provide importable license files and dongles to your production environment.

- KINGSTAR: After purchasing the KINGSTAR Runtime license, you can purchase a dongle to include the license. Plug the dongle into your system and use the RTX64 Activation and Configuration GUI utility to assign processors between Windows and RTSS. For more information, see the KINGSTAR Help or *KINGSTAR Runtime Installation Guide*.
- MaxRT wRTOS: Use the MaxRT Dongle Activation Utility, available from the MaxRT wRTOS space on the IntervalZero Support Site.

License Types

A license string is generated from the Activation Key. This is the process by which one or more product components or packages are activated, depending on the components or packages you have purchased. Licenses are provided by the IntervalZero License Server and are node-locked, meaning they can only be used on a single machine or dongle.

KINGSTAR License Types

KINGSTAR offers these license types:

- An Evaluation license is a temporary license with an expiration date.
- A Retail license is perpetual and does not expire.

wRTOS License Types

MaxRT wRTOS offers these license types:

- An Evaluation license is a temporary license with an expiration date.
- Multiple Retail license types:
 - A perpetual Retail license does not expire.
 - A time-limited Retail license has an expiration date.

3

Configuring and Controlling the Runtime

KINGSTAR and MaxRT wRTOS provide tools for configuring and controlling the Runtime and its dependent components.

Configuring and Controlling KINGSTAR

KINGSTAR provides a Control Panel with shortcuts to launch system settings pages and KINGSTAR tools. You can also start or stop the KINGSTAR Subsystem, KINGSTAR PLC, and RTX64 Subsystem.

For more information on KINGSTAR Control Panel, see the [KINGSTAR Help](#).

Configuring and Controlling wRTOS

MaxRT wRTOS Control Panel allows you to control (Start, Stop, Restart) the wRTOS Subsystem and other Runtime components, including NL2, TCP/IP, and E-CAT. It also provides shortcuts to launch several other wRTOS tools. wRTOS Control Panel does not provide settings for configuring the Runtime and other components. wRTOS configuration settings are available in the wRTOS Settings tool.

wRTOS Settings allows you to activate product components, configure RTSS and component cores, and configure default behavior for the Real-Time Subsystem (RTSS), RTSS applications, networking, fieldbuses, and other features. You can also access other wRTOS tools and resources. If you've purchased and installed the wRTOS SDK, wRTOS Settings will display information on the Visual Studio versions installed on the system.

4

Tool Comparison

KINGSTAR and MaxRT wRTOS provide an extensive set of tools to assist with various tasks. This chapter compares the KINGSTAR tools to the wRTOS E-CAT tools.

KINGSTAR tools

MaxRT wRTOS E-CAT tools

Analysis Console

- Included in KINGSTAR Fieldbus Runtime

Message Viewer

- Included in wRTOS Runtime

Note: wRTOS 1.0.1 does not support Diagnostic.

Configuration Tool

- Included in KINGSTAR Fieldbus Runtime

E-CAT Configuration Tool

- Included in the wRTOS Fieldbus package

Control Panel

- Included in KINGSTAR Fieldbus Runtime

Control Panel

- Included in wRTOS Runtime

Diagnostic Tool

- Included in KINGSTAR Fieldbus Runtime

E-CAT Diagnostic Tool

- Included in the wRTOS Fieldbus package

ESI Import Tool

- Included in KINGSTAR Fieldbus Runtime

E-CAT ESI Import Tool

- Included in the wRTOS Fieldbus package
-

KINGSTAR tools

MaxRT wRTOS E-CAT tools

Scope

wRTOS 1.0.1 does not support this tool.

- Included in KINGSTAR Fieldbus Runtime
-

Commands

wRTOS 1.0.1 does not support this tool.

- Included in KINGSTAR Fieldbus Runtime

5

Project Settings

KINGSTAR and MaxRT wRTOS provide Visual Studio application templates that can be used to create a real-time application (RTSS) executable or real-time DLL (RTDLL). The KINGSTAR SDK and wRTOS SDK include different sets of headers and libraries. This chapter compares the headers and libraries between KINGSTAR SDK and wRTOS E-CAT SDK.

KINGSTAR API set

wRTOS E-CAT Fieldbus API set

KINGSTAR Fieldbus

E-CAT Fieldbus (RTECAT) APIs

- Header: ksapi.h
- Libraries: KsApi.lib (Windows), KsApi_Rtss.lib (RTSS)

- Header: RtecatApi.h
- Libraries: RtecatApi_W64.lib (Windows), RtecatApi.lib (RTSS)

KINGSTAR COM

wRTOS 1.0.1 does not support KINGSTAR COM.

- Header: kscom.h
- Libraries: KsApi.lib (Windows), KsApi_Rtss.lib (RTSS)

KINGSTAR Motion

wRTOS 1.0.1 does not support KINGSTAR Motion.

- Header: ksmotion.h
- Libraries: KsApi.lib (Windows), KsApi_Rtss.lib (RTSS)

6

SDK Code Changes

This topic outlines the APIs included in KINGSTAR 4.x SDKs that were added, changed, or removed in wRTOS 1.x SDKs.

IN THIS CHAPTER:

- [New APIs](#)
 - [Breaking API Changes](#)
 - [Removed APIs](#)
-

New APIs

New E-CAT APIs

This section lists the new E-CAT APIs that were not present in KINGSTAR 4.x SDK but are included in wRTOS 1.x SDKs.

Axis Control APIs

- Added `RtecatForceAxisActualCurrent`, which writes the actual current of a simulated axis.
- Added `RtecatForceAxisActualPosition`, which writes the actual position of a simulated axis.
- Added `RtecatForceAxisActualTorque`, which writes the actual torque of a simulated axis.
- Added `RtecatForceAxisActualVelocity`, which writes the actual velocity of a simulated axis.
- Added `RtecatForceAxisCustomInput1`, which writes the first custom input variable of a simulated axis.
- Added `RtecatForceAxisCustomInput2`, which writes the second custom input variable of a simulated axis.
- Added `RtecatForceAxisFollowingError`, which writes the following error of a simulated axis.
- Added `RtecatForceAxisMopDisplay`, which writes the modes of operation display of a simulated axis.
- Added `RtecatForceAxisSecondEncoder`, which writes the secondary encoder's position of a simulated axis.

- Added **RtecatForceAxisStatusWord**, which writes the status word of a simulated axis.
- Added **RtecatForceAxisTouchProbeStatus**, which writes the touch probe status of a simulated axis.
- Added **RtecatGetAxisStatus**, which gets the status of an axis.
- Added **RtecatOpenAxisByIndex**, which opens the axis object by its index.
- Added **RtecatOpenAxisByLocation**, which opens the axis object by its location.
- Added **RtecatReadAxisControlWord**, which reads the control word of an axis.
- Added **RtecatReadAxisCustomOutput1**, which reads the first custom output variable of an axis.
- Added **RtecatReadAxisCustomOutput2**, which reads the second custom output variable of an axis.
- Added **RtecatReadAxisMaxCurrent**, which reads the maximum current of an axis.
- Added **RtecatReadAxisMaxTorque**, which reads the maximum torque of an axis.
- Added **RtecatReadAxisMop**, which reads the modes of operation of an axis.
- Added **RtecatReadAxisProfileAcceleration**, which reads the profile acceleration of an axis.
- Added **RtecatReadAxisProfileDeceleration**, which reads the profile deceleration of an axis.
- Added **RtecatReadAxisProfileVelocity**, which reads the profile velocity of an axis.
- Added **RtecatReadAxisTargetPosition**, which reads the target position of an axis.
- Added **RtecatReadAxisTargetTorque**, which reads the target torque of an axis.
- Added **RtecatReadAxisTargetVelocity**, which reads the target velocity of an axis.
- Added **RtecatReadAxisTorqueOffset**, which reads the torque offset of an axis.
- Added **RtecatReadAxisTouchProbeControl**, which reads the touch probe control of an axis.
- Added **RtecatSetAxisCustomName**, which sets a custom name for an axis.

Diagnostics APIs

- Added **RtecatResetDiagnostics**, which resets MainDevice and SubDevice diagnostics information.

MainDevice APIs

- Added **RtecatCloseHandle**, which closes an E-CAT object handle.
- Added **RtecatOpenMainDevice**, which opens a MainDevice object.

Module Control APIs

- Added **RtecatGetModuleStatus**, which gets the status of a module.
- Added **RtecatOpenModuleByIndex**, which opens a module object by its index.

- Added `RtecatOpenModuleByLocation`, which opens a module object by its location.
- Added `RtecatSetModuleCustomName`, which sets a custom name for a module.

SubDevice APIs

- Added `RtecatGetSubDeviceStatus`, which gets the status of a SubDevice.
- Added `RtecatOpenSubDeviceByExplicitId`, which opens a SubDevice object by its explicit identification.
- Added `RtecatOpenSubDeviceByIndex`, which opens a SubDevice object by its index.
- Added `RtecatOpenSubDeviceByStationAddress`, which opens a SubDevice object by its station address.

E-CAT Data Types

- Added `RTECAT_HANDLE`, which is a handle to a MainDevice, SubDevice, axis, module or command object.
- Added `RTECAT_AXIS_STATUS`, which describes the status of an axis.
- Added `RTECAT_MODULE_STATUS`, which describes the status of a module.
- Added `RTECAT_COMMAND_STATE`, which identifies the state of a command.

New Managed E-CAT APIs

This section lists the new managed E-CAT APIs that were not present in KINGSTAR 4.x SDK but are included in wRTOS 1.x SDKs.

Axis Control APIs

- Added method `GetAxisStatus`, which gets the status of an axis.
- Added method `OpenAxisByIndex`, which opens an axis object by its index.
- Added method `OpenAxisByLocation`, which opens an axis object by its location.
- Added method `SetAxisCustomName`, which sets a custom name for an axis.
- Added method `ReadAxisControlWord`, which gets the status of an axis.
- Added method `ReadAxisTargetPosition`, which reads the target position of an axis.
- Added method `ReadAxisTargetTorque`, which reads the target torque of an axis.
- Added method `ReadAxisTargetVelocity`, which reads the target velocity of an axis.
- Added method `ReadAxisTorqueOffset`, which reads the target torque offset of an axis.
- Added method `ReadAxisMaxTorque`, which reads the maximum torque of an axis.
- Added method `ReadAxisMaxCurrent`, which reads the maximum current of an axis.

- Added method **ReadAxisTouchProbeControl**, which reads the touch probe control of an axis.
- Added method **ReadAxisMop**, which reads the modes of operation of an axis.
- Added method **ReadAxisProfileVelocity**, which reads the profile velocity of an axis.
- Added method **ReadAxisProfileAcceleration**, which reads the profile acceleration of an axis.
- Added method **ReadAxisProfileDeceleration**, which reads the profile deceleration of an axis.
- Added method **ReadAxisCustomOutput1**, which reads the first custom output variable of an axis.
- Added method **ReadAxisCustomOutput2**, which reads the second custom output variable of an axis.
- Added method **ForceAxisStatusWord**, which writes the status word of a simulated axis.
- Added method **ForceAxisActualCurrent**, which writes the actual current of a simulated axis.
- Added method **ForceAxisActualPosition**, which writes the actual position of a simulated axis.
- Added method **ForceAxisActualTorque**, which writes the actual torque of a simulated axis.
- Added method **ForceAxisActualVelocity**, which writes the actual velocity of a simulated axis.
- Added method **ForceAxisCustomInput1**, which writes the first custom input variable of a simulated axis.
- Added method **ForceAxisCustomInput2**, which writes the second custom input variable of a simulated axis.
- Added method **ForceAxisFollowingError**, which writes the following error of a simulated axis.
- Added method **ForceAxisMopDisplay**, which writes the modes of operation display of a simulated axis.
- Added method **ForceAxisSecondEncoder**, which writes the secondary encoder's position of a simulated axis.
- Added method **ForceAxisTouchProbeStatus**, which writes the touch probe status of a simulated axis.

Diagnostics APIs

- Added method **ResetDiagnostics**, which resets MainDevice and SubDevice diagnostics information.

MainDevice APIs

- Added class **Library**, which provides various operations for E-CAT.
- Added constructor **Library**, which is the default constructor.
- Added method **CloseHandle**, which closes an open E-CAT object handle.
- Added method **GetStatus (Status)**, which gets the status of a MainDevice.
- Added method **GetStatus (Diagnostics)**, which gets the diagnostics information of a MainDevice.
- Added method **OpenMainDevice**, which opens a MainDevice object.

Module Control APIs

- Added method **GetModuleStatus**, which gets the status of a module.
- Added method **OpenModuleByIndex**, which opens a module object by its index.
- Added method **OpenModuleByLocation**, which opens a module object by its location.
- Added method **SetModuleCustomName**, which sets a custom name for a module.

SubDevice APIs

- Added method **OpenSubDeviceByExplicitId**, which opens a SubDevice object by its explicit identification.
- Added method **OpenSubDeviceByIndex**, which opens a SubDevice object by its index.
- Added method **OpenSubDeviceByStationAddress**, which opens a SubDevice object by its station address.
- Added method **GetSubDeviceStatus**, which gets the status of a SubDevice.

Other E-CAT APIs

- Added class **EcatHandle**, which is a handle to a MainDevice, SubDevice, axis, module or command.
- Added class **AxisStatus**, which describes an axis status.
- Added class **ModuleStatus**, which describes an axis status.

Breaking API Changes

This section lists the APIs included in KINGSTAR 4.x SDK that underwent breaking changes in wRTOS 1.x SDKs.

Breaking Changes to Header and Library Files

- The following header and library files were renamed:
 - **ksapi.h** has been renamed **RtecatApi.h**
 - **KsApi_Rtss.lib** has been renamed **RtecatApi.lib** (RTSS)
 - **KsApi.lib** has been renamed **RtecatApi_W64.lib** (Windows)

Breaking Changes to Namespaces

- The following namespaces were renamed:
 - **IntervalZero.KINGSTAR.Local.Api** has been renamed **IntervalZero.MaxRT.Ecat.Api.NetApi**
 - **IntervalZero.KINGSTAR.Base** has been renamed **IntervalZero.MaxRT.Ecat.Api**

Breaking Changes to E-CAT APIs

Axis Control APIs

KINGSTAR 4.x API

wRTOS 1.0 and later

ForceAxisInputBit

```
KsError ForceAxisInputBit(  
    int Index,  
    int BitOffset,  
    BOOL Value  
);
```

RtecatForceAxisInputBit

```
RTECAT_ERROR RtecatForceAxisInputBit(  
    RTECAT_HANDLE Axis,  
    ULONG BitOffset,  
    BOOL Value  
);
```

- Renamed **ForceAxisInputBit** to **RtecatForceAxisInputBit**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *BitOffset* from **int** to **ULONG**.

ForceAxisInputByte

```
KsError ForceAxisInputByte(  
    int Index,  
    int ByteOffset,  
    BYTE Value  
);
```

RtecatForceAxisInputByte

```
RTECAT_ERROR RtecatForceAxisInputByte(  
    RTECAT_HANDLE Axis,  
    ULONG ByteOffset,  
    BYTE Value  
);
```

- Renamed **ForceAxisInputByte** to **RtecatForceAxisInputByte**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.

ForceAxisInputDWord

```
KsError ForceAxisInputDWord(  
    int Index,  
    int ByteOffset,  
    DWORD Value  
);
```

RtecatForceAxisInputDWord

```
RTECAT_ERROR RtecatForceAxisInputDWord(  
    RTECAT_HANDLE Axis,  
    ULONG ByteOffset,  
    DWORD Value  
);
```

- Renamed **ForceAxisInputDWord** to **RtecatForceAxisInputDWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.

ForceAxisInputs

```
KsError ForceAxisInputs(  
    int Index,  
    DWORD Value  
);
```

RtecatForceAxisDigitalInputs

```
RTECAT_ERROR RtecatForceAxisDigitalInputs(  
    RTECAT_HANDLE Axis,  
    DWORD Value  
);
```

- Renamed **ForceAxisInputs** to **RtecatForceAxisDigitalInputs**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
-

ForceAxisInputWord

```
KsError ForceAxisInputWord(
    int Index,
    int ByteOffset,
    WORD Value
);
```

RtecatForceAxisInputWord

```
RTECAT_ERROR RtecatForceAxisInputWord(
    RTECAT_HANDLE Axis,
    ULONG ByteOffset,
    WORD Value
);
```

- Renamed **ForceAxisInputWord** to **RtecatForceAxisInputWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.

GetAxisBuffers

```
KsError GetAxisBuffers(
    int Index,
    void** InputBuffer,
    void** OutputBuffer
);
```

RtecatGetAxisBuffers

```
RTECAT_ERROR RtecatGetAxisBuffers(
    RTECAT_HANDLE Axis,
    void** pInputBuffer,
    void** pOutputBuffer
);
```

- Renamed **GetAxisBuffers** to **RtecatGetAxisBuffers**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Renamed parameter *InputBuffer* to *pInputBuffer*.
- Renamed parameter *OutputBuffer* to *pOutputBuffer*.

ReadAxisActualCurrent

```
KsError ReadAxisActualCurrent(  
    int Index,  
    short* Value  
);
```

RtecatReadAxisActualCurrent

```
RTECAT_ERROR RtecatReadAxisActualCurrent(  
    RTECAT_HANDLE Axis,  
    short* pValue  
);
```

- Renamed **ReadAxisActualCurrent** to **RtecatReadAxisActualCurrent**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisActualPosition

```
KsError ReadAxisActualPosition(  
    int Index,  
    int* Value  
);
```

RtecatReadAxisActualPosition

```
RTECAT_ERROR RtecatReadAxisActualPosition(  
    RTECAT_HANDLE Axis,  
    int* pValue  
);
```

- Renamed **ReadAxisActualPosition** to **RtecatReadAxisActualPosition**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisActualTorque

```
KsError ReadAxisActualTorque(  
    int Index,  
    short* Value  
);
```

RtecatReadAxisActualTorque

```
RTECAT_ERROR RtecatReadAxisActualTorque(  
    RTECAT_HANDLE Axis,  
    short* pValue  
);
```

- Renamed **ReadAxisActualTorque** to **RtecatReadAxisActualTorque**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Renamed parameter *Value* to *pValue*.

ReadAxisActualVelocity

```
KsError ReadAxisActualVelocity(  
    int Index,  
    int* Value  
);
```

RtecatReadAxisActualVelocity

```
RTECAT_ERROR RtecatReadAxisActualVelocity(  
    RTECAT_HANDLE Axis,  
    int* pValue  
);
```

- Renamed **ReadAxisActualVelocity** to **RtecatReadAxisActualVelocity**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisCustomInput1

```
KsError ReadAxisCustomInput1(  
    int Index,  
    DWORD* Value  
);
```

RtecatReadAxisCustomInput1

```
RTECAT_ERROR RtecatReadAxisCustomInput1(  
    RTECAT_HANDLE Axis,  
    DWORD* pValue  
);
```

- Renamed **ReadAxisCustomInput1** to **RtecatReadAxisCustomInput1**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisCustomInput2

```
KsError ReadAxisCustomInput2(  
    int Index,  
    DWORD* Value  
);
```

RtecatReadAxisCustomInput2

```
RTECAT_ERROR RtecatReadAxisCustomInput2(  
    RTECAT_HANDLE Axis,  
    DWORD* pValue  
);
```

- Renamed **ReadAxisCustomInput2** to **RtecatReadAxisCustomInput2**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisFollowingError

```
KsError ReadAxisFollowingError(
    int Index,
    int* Value
);
```

RtecatReadAxisFollowingError

```
RTECAT_ERROR RtecatReadAxisFollowingError(
    RTECAT_HANDLE Axis,
    int* pValue
);
```

- Renamed **ReadAxisFollowingError** to **RtecatReadAxisFollowingError**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Renamed parameter *Value* to *pValue*.

ReadAxisInputBit

```
KsError ReadAxisInputBit(
    int Index,
    int BitOffset,
    BOOL* Value
);
```

RtecatReadAxisInputBit

```
RTECAT_ERROR RtecatReadAxisInputBit(
    RTECAT_HANDLE Axis,
    ULONG BitOffset,
    BOOL* pValue
);
```

- Renamed **ReadAxisInputBit** to **RtecatReadAxisInputBit**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *BitOffset* from *int* to *ULONG*.
- Renamed parameter *Value* to *pValue*.

ReadAxisInputByte

```
KsError ReadAxisInputByte(  
    int Index,  
    int ByteOffset,  
    BYTE* Value  
);
```

RtecatReadAxisInputByte

```
RTECAT_ERROR RtecatReadAxisInputByte(  
    RTECAT_HANDLE Axis,  
    ULONG ByteOffset,  
    BYTE* pValue  
);
```

- Renamed **ReadAxisInputByte** to **RtecatReadAxisInputByte**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.
- Renamed parameter *Value* to *pValue*.

ReadAxisInputDWord

```
KsError ReadAxisInputDWord(  
    int Index,  
    int ByteOffset,  
    DWORD* Value  
);
```

RtecatReadAxisInputDWord

```
RTECAT_ERROR RtecatReadAxisInputDWord(  
    RTECAT_HANDLE Axis,  
    ULONG ByteOffset,  
    DWORD* pValue  
);
```

- Renamed **ReadAxisInputDWord** to **RtecatReadAxisInputDWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.
- Renamed parameter *Value* to *pValue*.

ReadAxisInputs

```
KsError ReadAxisInputs(  
    int Index,  
    DWORD* Value  
);
```

RtecatReadAxisDigitalInputs

```
RTECAT_ERROR RtecatReadAxisDigitalInputs(  
    RTECAT_HANDLE Axis,  
    DWORD* pValue  
);
```

- Renamed **ReadAxisInputs** to **RtecatReadAxisDigitalInputs**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisInputWord

```
KsError ReadAxisInputWord(  
    int Index,  
    int ByteOffset,  
    WORD* Value  
);
```

RtecatReadAxisInputWord

```
RTECAT_ERROR RtecatReadAxisInputWord(  
    RTECAT_HANDLE Axis,  
    ULONG ByteOffset,  
    WORD* pValue  
);
```

- Renamed **ReadAxisInputWord** to **RtecatReadAxisInputWord**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Changed the data type of *ByteOffset* from *int* to *ULONG*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisMopDisplay

```
KsError ReadAxisMopDisplay(  
    int Index,  
    BYTE* Value  
);
```

RtecatReadAxisMopDisplay

```
RTECAT_ERROR RtecatReadAxisMopDisplay(  
    RTECAT_HANDLE Axis,  
    BYTE* pValue  
);
```

- Renamed **ReadAxisMopDisplay** to **RtecatReadAxisMopDisplay**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Renamed parameter *Value* to *pValue*.

ReadAxisOutputBit

```
KsError ReadAxisOutputBit(  
    int Index,  
    int BitOffset,  
    BOOL* Value  
);
```

RtecatReadAxisOutputBit

```
RTECAT_ERROR RtecatReadAxisOutputBit(  
    RTECAT_HANDLE Axis,  
    ULONG BitOffset,  
    BOOL* pValue  
);
```

- Renamed **ReadAxisOutputBit** to **RtecatReadAxisOutputBit**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Changed the data type of *BitOffset* from *int* to *ULONG*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisOutputByte

```
KsError ReadAxisOutputByte(
    int Index,
    int ByteOffset,
    BYTE* Value
);
```

RtecatReadAxisOutputByte

```
RTECAT_ERROR RtecatReadAxisOutputByte(
    RTECAT_HANDLE Axis,
    ULONG ByteOffset,
    BYTE* pValue
);
```

- Renamed **ReadAxisOutputByte** to **RtecatReadAxisOutputByte**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.
- Renamed parameter *Value* to *pValue*.

ReadAxisOutputDWord

```
KsError ReadAxisOutputDWord(
    int Index,
    int ByteOffset,
    DWORD* Value
);
```

RtecatReadAxisOutputDWord

```
RTECAT_ERROR RtecatReadAxisOutputDWord(
    RTECAT_HANDLE Axis,
    ULONG ByteOffset,
    DWORD* pValue
);
```

- Renamed **ReadAxisOutputDWord** to **RtecatReadAxisOutputDWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.
- Renamed parameter *Value* to *pValue*.

ReadAxisOutputs

```
KsError ReadAxisOutputs(  
    int Index,  
    DWORD* Value  
);
```

RtecatReadAxisDigitalOutputs

```
RTECAT_ERROR RtecatReadAxisDigitalOutputs(  
    RTECAT_HANDLE Axis,  
    DWORD* pValue  
);
```

- Renamed **ReadAxisOutputs** to **RtecatReadAxisDigitalOutputs**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisOutputWord

```
KsError ReadAxisOutputWord(  
    int Index,  
    int ByteOffset,  
    WORD* Value  
);
```

RtecatReadAxisOutputWord

```
RTECAT_ERROR RtecatReadAxisOutputWord(  
    RTECAT_HANDLE Axis,  
    ULONG ByteOffset,  
    WORD* pValue  
);
```

- Renamed **ReadAxisOutputWord** to **RtecatReadAxisOutputWord**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Changed the data type of *ByteOffset* from *int* to *ULONG*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisSecondEncoder

```
KsError ReadAxisSecondEncoder(  
    int Index,  
    int* Value  
);
```

RtecatReadAxisSecondEncoder

```
RTECAT_ERROR RtecatReadAxisSecondEncoder(  
    RTECAT_HANDLE Axis,  
    int* pValue  
);
```

- Renamed **ReadAxisSecondEncoder** to **RtecatReadAxisSecondEncoder**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Renamed parameter *Value* to *pValue*.

ReadAxisStatusWord

```
KsError ReadAxisStatusWord(  
    int Index,  
    WORD* Value  
);
```

RtecatReadAxisStatusWord

```
RTECAT_ERROR RtecatReadAxisStatusWord(  
    RTECAT_HANDLE Axis,  
    WORD* pValue  
);
```

- Renamed **ReadAxisStatusWord** to **RtecatReadAxisStatusWord**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
 - Renamed parameter *Value* to *pValue*.
-

ReadAxisTouchProbeStatus

```
KsError ReadAxisTouchProbeStatus(
    int Index,
    WORD* Value
);
```

RtecatReadAxisTouchProbeStatus

```
RTECAT_ERROR RtecatReadAxisTouchProbeStatus(
    RTECAT_HANDLE Axis,
    WORD* pValue
);
```

- Renamed **ReadAxisTouchProbeStatus** to **RtecatReadAxisTouchProbeStatus**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Renamed parameter *Value* to *pValue*.

WriteAxisControlWord

```
KsError WriteAxisControlWord(
    int Index,
    WORD Value
);
```

RtecatWriteAxisControlWord

```
RTECAT_ERROR RtecatWriteAxisControlWord(
    RTECAT_HANDLE Axis,
    WORD Value
);
```

- Renamed **WriteAxisControlWord** to **RtecatWriteAxisControlWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.

WriteAxisCustomOutput1

```
KsError WriteAxisCustomOutput1(  
    int Index,  
    DWORD Value  
);
```

RtecatWriteAxisCustomOutput1

```
RTECAT_ERROR RtecatWriteAxisCustomOutput1(  
    RTECAT_HANDLE Axis,  
    DWORD Value  
);
```

- Renamed **WriteAxisCustomOutput1** to **RtecatWriteAxisCustomOutput1**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.

WriteAxisCustomOutput2

```
KsError WriteAxisCustomOutput2(  
    int Index,  
    DWORD Value  
);
```

RtecatWriteAxisCustomOutput2

```
RTECAT_ERROR RtecatWriteAxisCustomOutput2(  
    RTECAT_HANDLE Axis,  
    DWORD Value  
);
```

- Renamed **WriteAxisCustomOutput2** to **RtecatWriteAxisCustomOutput2**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
-

WriteAxisMaxCurrent

```
KsError WriteAxisMaxCurrent(  
    int Index,  
    short Value  
);
```

RtecatWriteAxisMaxCurrent

```
RTECAT_ERROR RtecatWriteAxisMaxCurrent(  
    RTECAT_HANDLE Axis,  
    short Value  
);
```

- Renamed **WriteAxisMaxCurrent** to **RtecatWriteAxisMaxCurrent**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
-

WriteAxisMaxTorque

```
KsError WriteAxisMaxTorque(  
    int Index,  
    short Value  
);
```

RtecatWriteAxisMaxTorque

```
RTECAT_ERROR RtecatWriteAxisMaxTorque(  
    RTECAT_HANDLE Axis,  
    short Value  
);
```

- Renamed **WriteAxisMaxTorque** to **RtecatWriteAxisMaxTorque**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
-

WriteAxisMop

```
KsError WriteAxisMop(  
    int Index,  
    BYTE Value  
);
```

RtecatWriteAxisMop

```
RTECAT_ERROR RtecatWriteAxisMop(  
    RTECAT_HANDLE Axis,  
    BYTE Value  
);
```

- Renamed **WriteAxisMop** to **RtecatWriteAxisMop**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.

WriteAxisOutputBit

```
KsError WriteAxisOutputBit(  
    int Index,  
    int BitOffset,  
    BOOL Value  
);
```

RtecatWriteAxisOutputBit

```
RTECAT_ERROR RtecatWriteAxisOutputBit(  
    RTECAT_HANDLE Axis,  
    ULONG BitOffset,  
    BOOL Value  
);
```

- Renamed **WriteAxisOutputBit** to **RtecatWriteAxisOutputBit**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *BitOffset* from *int* to *ULONG*.

WriteAxisOutputByte

```
KsError WriteAxisOutputByte(
    int Index,
    int ByteOffset,
    BYTE Value
);
```

RtecatWriteAxisOutputByte

```
RTECAT_ERROR RtecatWriteAxisOutputByte(
    RTECAT_HANDLE Axis,
    ULONG ByteOffset,
    BYTE Value
);
```

- Renamed **WriteAxisOutputByte** to **RtecatWriteAxisOutputByte**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.

WriteAxisOutputDWord

```
KsError WriteAxisOutputDWord(
    int Index,
    int ByteOffset,
    DWORD Value
);
```

RtecatWriteAxisOutputDWord

```
RTECAT_ERROR RtecatWriteAxisOutputDWord(
    RTECAT_HANDLE Axis,
    ULONG ByteOffset,
    DWORD Value
);
```

- Renamed **WriteAxisOutputDWord** to **RtecatWriteAxisOutputDWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.

WriteAxisOutputs

```
KsError WriteAxisOutputs(
    int Index,
    DWORD Value
);
```

RtecatWriteAxisDigitalOutputs

```
RTECAT_ERROR RtecatWriteAxisDigitalOutputs(
    RTECAT_HANDLE Axis,
    DWORD Value
);
```

- Renamed **WriteAxisOutputs** to **RtecatWriteAxisDigitalOutputs**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.

WriteAxisOutputWord

```
KsError WriteAxisOutputWord(
    int Index,
    int ByteOffset,
    WORD Value
);
```

RtecatWriteAxisOutputWord

```
RTECAT_ERROR RtecatWriteAxisOutputWord(
    RTECAT_HANDLE Axis,
    ULONG ByteOffset,
    WORD Value
);
```

- Renamed **WriteAxisOutputWord** to **RtecatWriteAxisOutputWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.

WriteAxisProfileAcceleration

```
KsError  
WriteAxisProfileAcceleration(  
    int Index,  
    int Value  
);
```

RtecatWriteAxisProfileAcceleration

```
RTECAT_ERROR  
RtecatWriteAxisProfileAcceleration(  
    RTECAT_HANDLE Axis,  
    int Value  
);
```

- Renamed **WriteAxisProfileAcceleration** to **RtecatWriteAxisProfileAcceleration**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.

WriteAxisProfileDeceleration

```
KsError  
WriteAxisProfileDeceleration(  
    int Index,  
    int Value  
);
```

RtecatWriteAxisProfileDeceleration

```
RTECAT_ERROR  
RtecatWriteAxisProfileDeceleration(  
    RTECAT_HANDLE Axis,  
    int Value  
);
```

- Renamed **WriteAxisProfileDeceleration** to **RtecatWriteAxisProfileDeceleration**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
-

WriteAxisProfileVelocity

```
KsError WriteAxisProfileVelocity(  
    int Index,  
    int Value  
);
```

RtecatWriteAxisProfileVelocity

```
RTECAT_ERROR  
RtecatWriteAxisProfileVelocity(  
    RTECAT_HANDLE Axis,  
    int Value  
);
```

- Renamed **WriteAxisProfileVelocity** to **RtecatWriteAxisProfileVelocity**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.

WriteAxisTargetPosition

```
KsError WriteAxisTargetPosition(  
    int Index,  
    int Value  
);
```

RtecatWriteAxisTargetPosition

```
RTECAT_ERROR RtecatWriteAxisTargetPosition  
(  
    RTECAT_HANDLE Axis,  
    int Value  
);
```

- Renamed **WriteAxisTargetPosition** to **RtecatWriteAxisTargetPosition**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
-

WriteAxisTargetTorque

```
KsError WriteAxisTargetTorque(  
    int Index,  
    short Value  
);
```

RtecatWriteAxisTargetTorque

```
RTECAT_ERROR RtecatWriteAxisTargetTorque(  
    RTECAT_HANDLE Axis,  
    short Value  
);
```

- Renamed **WriteAxisTargetTorque** to **RtecatWriteAxisTargetTorque**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.

WriteAxisTargetVelocity

```
KsError WriteAxisTargetVelocity(  
    int Index,  
    int Value  
);
```

RtecatWriteAxisTargetVelocity

```
RTECAT_ERROR RtecatWriteAxisTargetVelocity(  
    (  
    RTECAT_HANDLE Axis,  
    int Value  
);
```

- Renamed **WriteAxisTargetVelocity** to **RtecatWriteAxisTargetVelocity**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Axis*.
-

WriteAxisTorqueOffset

```
KsError WriteAxisTorqueOffset(  
    int Index,  
    short Value  
);
```

RtecatWriteAxisTorqueOffset

```
RTECAT_ERROR RtecatWriteAxisTorqueOffset(  
    RTECAT_HANDLE Axis,  
    short Value  
);
```

- Renamed **WriteAxisTorqueOffset** to **RtecatWriteAxisTorqueOffset**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.

WriteAxisTouchProbeControl

```
KsError  
WriteAxisTouchProbeControl(  
    int Index,  
    WORD Value  
);
```

RtecatWriteAxisTouchProbeControl

```
RTECAT_ERROR  
RtecatWriteAxisTouchProbeControl(  
    RTECAT_HANDLE Axis,  
    WORD Value  
);
```

- Renamed **WriteAxisTouchProbeControl** to **RtecatWriteAxisTouchProbeControl**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Axis*.

Command APIs

AbortCommand

```
KsCommandStatus AbortCommand(
    KsCommandStatus Status
);
```

RtecatAbortCommand

```
RTECAT_COMMAND_STATUS RtecatAbortCommand(
    RTECAT_COMMAND_STATUS Status
);
```

- Renamed **AbortCommand** to **RtecatAbortCommand**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Changed the data type of *Status* from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.

GetCommandsStatus

```
KsError GetCommandsStatus(
    int Length,
    KsCommandStatus* Statuses
);
```

RtecatGetCommandsStatus

```
RTECAT_ERROR RtecatGetCommandsStatus(
    ULONG Length,
    RTECAT_COMMAND_STATUS* pStatuses
);
```

- Renamed **GetCommandsStatus** to **RtecatGetCommandsStatus**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Changed the data type of *Length* from **int** to **ULONG**.
- Renamed parameter *Statuses* to *pStatuses* and changed its data type from **KsCommandStatus*** to **RTECAT_COMMAND_STATUS***.

GetCommandStatus

```
KsCommandStatus GetCommandStatus
(
    KsCommandStatus Status
);
```

RtecatGetCommandStatus

```
RTECAT_COMMAND_STATUS
RtecatGetCommandStatus(
    RTECAT_COMMAND_STATUS Status
);
```

- Renamed **GetCommandStatus** to **RtecatGetCommandStatus**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Changed the data type of *Status* from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.

WaitForCommand

```
KsCommandStatus WaitForCommand(
    double Timeout,
    BOOL AbortOnTimeout,
    KsCommandStatus Status
);
```

RtecatWaitForCommand

```
RTECAT_COMMAND_STATUS RtecatWaitForCommand(
    double Timeout,
    BOOL AbortOnTimeout,
    RTECAT_COMMAND_STATUS Status
);
```

- Renamed **WaitForCommand** to **RtecatWaitForCommand**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Changed the data type of *Status* from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.

Mailbox APIs

AoeReadCommand

```
KsCommandStatus AoeReadCommand(
    int SlaveId,
    int Port,
    int IndexGroup,
    int IndexOffset,
    BYTE* Value,
    int Length
);
```

RtecatAoeReadCommand

```
RTECAT_COMMAND_STATUS RtecatAoeReadCommand
(
    RTECAT_HANDLE SubDevice,
    WORD Port,
    ULONG IndexGroup,
    ULONG IndexOffset,
    BYTE* pValue,
    ULONG Length
);
```

- Renamed **AoeReadCommand** to **RtecatAoeReadCommand**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *Value* to *pValue*.
- Changed the data type of *Port* from int to WORD.
- Changed the data type of *IndexGroup*, *IndexOffset* and *Length* from int to ULONG.

AoeReadSdoObject

```
KsCommandStatus AoeReadSdoObject (
    int SlaveId,
    int Port,
    int ObIndex,
    int ObSubIndex,
    BOOL CompleteAccess,
    BYTE* Value,
    int Length
);
```

RtecatAoeReadSdoObject

```
RTECAT_COMMAND_STATUS
RtecatAoeReadSdoObject (
    RTECAT_HANDLE SubDevice,
    WORD Port,
    WORD ObIndex,
    BYTE ObSubIndex,
    BOOL CompleteAccess,
    BYTE* pValue,
    ULONG Length
);
```

- Renamed **AoeReadSdoObject** to **RtecatAoeReadSdoObject**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *Value* to *pValue*.
- Changed the data type of *Port* and *ObIndex* from int to WORD.
- Changed the data type of *ObSubIndex* from int to BYTE.
- Changed the data type of *Length* from int to ULONG.

AoeReadWriteCommand

```

KsCommandStatus
AoeReadWriteCommand(
    int SlaveId,
    int Port,
    int IndexGroup,
    int IndexOffset,
    BYTE* ReadValue,
    int ReadLength,
    BYTE* WriteValue,
    int WriteLength
);

```

RtecatAoeReadWriteCommand

```

RTECAT_COMMAND_STATUS
RtecatAoeReadWriteCommand(
    RTECAT_HANDLE SubDevice,
    WORD Port,
    ULONG IndexGroup,
    ULONG IndexOffset,
    BYTE* pReadValue,
    ULONG ReadLength,
    const BYTE* pWriteValue,
    ULONG WriteLength
);

```

- Renamed **AoeReadWriteCommand** to **RtecatAoeReadWriteCommand**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *ReadValue* to *pReadValue*.
- Renamed parameter *WriteValue* to *pWriteValue* and changed its data type from **BYTE*** to **const BYTE***.
- Changed the data type of *Port* from **int** to **WORD**.
- Changed the data type of *IndexGroup*, *IndexOffset*, *ReadLength* and *WriteLength* from **int** to **ULONG**.

AoeWriteCommand

```
KsCommandStatus AoeWriteCommand(  
    int SlaveId,  
    int Port,  
    int IndexGroup,  
    int IndexOffset,  
    BYTE* Value,  
    int Length  
);
```

RtecatAoeWriteCommand

```
RTECAT_COMMAND_STATUS  
RtecatAoeWriteCommand(  
    RTECAT_HANDLE SubDevice,  
    WORD Port,  
    ULONG IndexGroup,  
    ULONG IndexOffset,  
    const BYTE* pValue,  
    ULONG Length  
);
```

- Renamed **AoeWriteCommand** to **RtecatAoeWriteCommand**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *Value* to *pValue* and changed its data type from **BYTE*** to **const BYTE***.
- Changed the data type of *Port* from **int** to **WORD**.
- Changed the data type of *IndexGroup*, *IndexOffset* and *Length* from **int** to **ULONG**.

AoeWriteControlCommand

```

KsCommandStatus
AoeWriteControlCommand(
    int SlaveId,
    int Port,
    WORD Aoestate,
    WORD DeviceState,
    BYTE* Value,
    int Length
);

```

RtecatAoeWriteControlCommand

```

RTECAT_COMMAND_STATUS
RtecatAoeWriteControlCommand(
    RTECAT_HANDLE SubDevice,
    WORD Port,
    WORD Aoestate,
    WORD DeviceState,
    const BYTE* pValue,
    ULONG Length
);

```

- Renamed **AoeWriteControlCommand** to **RtecatAoeWriteControlCommand**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Changed the data type of *Port* from int to WORD.
- Renamed parameter *Value* to *pValue* and changed its data type from BYTE* to const BYTE*.
- Changed the data type of *Length* from int to ULONG.

AoeWriteSdoObject

```
KsCommandStatus AoeWriteSdoObject
(
    int SlaveId,
    int Port,
    int ObIndex,
    int ObSubIndex,
    BOOL CompleteAccess,
    BYTE* Value,
    int Length
);
```

RtecatAoeWriteSdoObject

```
RTECAT_COMMAND_STATUS
RtecatAoeWriteSdoObject(
    RTECAT_HANDLE SubDevice,
    WORD Port,
    WORD ObIndex,
    BYTE ObSubIndex,
    BOOL CompleteAccess,
    const BYTE* pValue,
    ULONG Length
);
```

- Renamed **AoeWriteSdoObject** to **RtecatAoeWriteSdoObject**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *Value* to *pValue* and changed its data type from **BYTE*** to **const BYTE***.
- Changed the data type of *Port* and *ObIndex* from **int** to **WORD**.
- Changed the data type of *ObSubIndex* from **int** to **BYTE**.
- Changed the data type of *Length* from **int** to **ULONG**.

CoeReadSdoEntryDescription

```
KsCommandStatus  
CoeReadSdoEntryDescription(  
    int SlaveId,  
    SdoEntryDescription* Data  
);
```

RtecatCoeReadSdoEntryDescription

```
RTECAT_COMMAND_STATUS  
RtecatCoeReadSdoEntryDescription(  
    RTECAT_HANDLE SubDevice,  
    RTECAT_SDO_ENTRY_DESCRIPTION* pData  
);
```

- Renamed **CoeReadSdoEntryDescription** to **RtecatCoeReadSdoEntryDescription**.
 - Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
 - Removed parameter *SlaveId* and added parameter *SubDevice*.
 - Renamed parameter *Data* to *pData* and changed its data type from **SdoEntryDescription*** to **RTECAT_SDO_ENTRY_DESCRIPTION***.
-

CoeReadSdoObject

```
KsCommandStatus CoeReadSdoObject(  
    int SlaveId,  
    int ObIndex,  
    int ObSubIndex,  
    BOOL CompleteAccess,  
    BYTE* Value,  
    int Length  
);
```

RtecatCoeReadSdoObject

```
RTECAT_COMMAND_STATUS  
RtecatCoeReadSdoObject(  
    RTECAT_HANDLE SubDevice,  
    WORD ObIndex,  
    BYTE ObSubIndex,  
    BOOL CompleteAccess,  
    BYTE* pValue,  
    ULONG Length  
);
```

- Renamed **CoeReadSdoObject** to **RtecatCoeReadSdoObject**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *Value* to *pValue*.
- Changed the data type of *ObIndex* from int to WORD.
- Changed the data type of *ObSubIndex* from int to BYTE.
- Changed the data type of *Length* from int to ULONG.

CoeReadSdoObjectDescription

```
KsCommandStatus  
CoeReadSdoObjectDescription(  
    int SlaveId,  
    SdoObjectDescription* Data  
);
```

RtecatCoeReadSdoObjectDescription

```
RTECAT_COMMAND_STATUS  
RtecatCoeReadSdoObjectDescription(  
    RTECAT_HANDLE SubDevice,  
    RTECAT_SDO_OBJECT_DESCRIPTION* pData  
);
```

- Renamed **CoeReadSdoObjectDescription** to **RtecatCoeReadSdoObjectDescription**.
 - Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
 - Removed parameter *SlaveId* and added parameter *SubDevice*.
 - Renamed parameter *Data* to *pData* and changed its data type from **SdoObjectDescription*** to **RTECAT_SDO_OBJECT_DESCRIPTION***.
-

CoeReadSdoODList

```
KsCommandStatus CoeReadSdoODList(  
    int SlaveId,  
    int ListType,  
    BYTE* Data,  
    int Length,  
    int* LengthRead  
);
```

RtecatCoeReadSdoODList

```
RTECAT_COMMAND_STATUS  
RtecatCoeReadSdoODList(  
    RTECAT_HANDLE SubDevice,  
    WORD ListType,  
    WORD* pData,  
    ULONG Length  
);
```

- Renamed **CoeReadSdoODList** to **RtecatCoeReadSdoODList**.
 - Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
 - Removed parameter *SlaveId* and added parameter *SubDevice*.
 - Renamed parameter *Data* to *pData* and changed its data type from **BYTE*** to **WORD***.
 - Changed the data type of *ListType* from **int** to **WORD**.
 - Changed the data type of *Length* from **int** to **ULONG**.
 - Removed parameter *LengthRead*.
-

CoeWriteSdoObject

```
KsCommandStatus CoeWriteSdoObject
(
    int SlaveId,
    int ObIndex,
    int ObSubIndex,
    BOOL CompleteAccess,
    BYTE* Value,
    int Length
);
```

RtecatCoeWriteSdoObject

```
RTECAT_COMMAND_STATUS
RtecatCoeWriteSdoObject(
    RTECAT_HANDLE SubDevice,
    WORD ObIndex,
    BYTE ObSubIndex,
    BOOL CompleteAccess,
    const BYTE* pValue,
    ULONG Length
);
```

- Renamed **CoeWriteSdoObject** to **RtecatCoeWriteSdoObject**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *Value* to *pValue* and changed its data type from **BYTE*** to **const BYTE***.
- Changed the data type of *ObIndex* from **int** to **WORD**.
- Changed the data type of *ObSubIndex* from **int** to **BYTE**.
- Changed the data type of *Length* from **int** to **ULONG**.

FoeReadFileToBuffer

```

KsCommandStatus
FoeReadFileToBuffer(
    int SlaveId,
    BOOL BootState,
    int NameLength,
    char* Name,
    DWORD Password,
    int BufferLength,
    BYTE* Buffer,
    int* ReadLength
);

```

RtecatFoeReadFileToBuffer

```

RTECAT_COMMAND_STATUS
RtecatFoeReadFileToBuffer(
    RTECAT_HANDLE SubDevice,
    BOOL BootState,
    const char* pName,
    DWORD Password,
    ULONG BufferLength,
    BYTE* pBuffer
);

```

- Renamed **FoeReadFileToBuffer** to **RtecatFoeReadFileToBuffer**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and *ReadLength*.
- Added parameter *SubDevice*.
- Removed parameter *NameLength*.
- Renamed parameter *Name* to *pName* and changed its data type from *char** to *const char**.
- Renamed parameter *Buffer* to *pBuffer*.
- Changed the data type of *BufferLength* from *int* to *ULONG*.

FoeReadFileToFile

```
KsCommandStatus FoeReadFileToFile  
(  
    int SlaveId,  
    BOOL BootState,  
    int NameLength,  
    char* Name,  
    DWORD Password,  
    char* File  
);
```

RtecatFoeReadFileToFile

```
RTECAT_COMMAND_STATUS  
RtecatFoeReadFileToFile(  
    RTECAT_HANDLE SubDevice,  
    BOOL BootState,  
    const char* pName,  
    DWORD Password,  
    const wchar_t* pFile  
);
```

- Renamed **FoeReadFileToFile** to **RtecatFoeReadFileToFile**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameters *SlaveId* and *NameLength*.
- Added parameter *SubDevice*.
- Renamed parameter *Name* to *pName* and changed its data type from `char*` to `const char*`.
- Renamed parameter *File* to *pFile* and changed its data type from `char*` to `const wchar_t*`.

FoeWriteFileFromBuffer

```

KsCommandStatus
FoeWriteFileFromBuffer(
    int SlaveId,
    BOOL BootState,
    int NameLength,
    char* Name,
    DWORD Password,
    int BufferLength,
    BYTE* Buffer
);

```

RtecatFoeWriteFileFromBuffer

```

RTECAT_COMMAND_STATUS
RtecatFoeWriteFileFromBuffer(
    RTECAT_HANDLE SubDevice,
    BOOL BootState,
    const char* pName,
    DWORD Password,
    ULONG BufferLength,
    const BYTE* pBuffer
);

```

- Renamed **FoeWriteFileFromBuffer** to **RtecatFoeWriteFileFromBuffer**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameters *SlaveId* and *NameLength*.
- Added parameter *SubDevice*.
- Changed the data type of *BufferLength* from int to ULONG.
- Renamed parameter *Name* to *pName* and changed its data type from char* to const char*.
- Renamed parameter *Buffer* to *pBuffer* and changed its data type from BYTE* to const BYTE*.

FoeWriteFileFromFile

```

KsCommandStatus
FoeWriteFileFromFile(
    int SlaveId,
    BOOL BootState,
    int NameLength,
    char* Name,
    DWORD Password,
    char* Filer
);

```

RtecatFoeWriteFileFromFile

```

RTECAT_COMMAND_STATUS
RtecatFoeWriteFileFromFile(
    RTECAT_HANDLE SubDevice,
    BOOL BootState,
    const char* pName,
    DWORD Password,
    const wchar_t* pFile
);

```

- Renamed **FoeWriteFileFromFile** to **RtecatFoeWriteFileFromFile**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameters *SlaveId* and *NameLength*.
- Added parameter *SubDevice*.
- Renamed parameter *Name* to *pName* and changed its data type from `char*` to `const char*`.
- Renamed parameter *File* to *pFile* and changed its data type from `char*` to `const wchar_t*`.

ReadAxisSdoObject

```
KsCommandStatus ReadAxisSdoObject
(
    int Index,
    int ObIndex,
    int ObSubIndex,
    BOOL CompleteAccess,
    BYTE* Value,
    int Length
);
```

RtecatReadAxisSdoObject

```
RTECAT_COMMAND_STATUS
RtecatReadAxisSdoObject(
    RTECAT_HANDLE Axis,
    WORD ObIndex,
    BYTE ObSubIndex,
    BOOL CompleteAccess,
    BYTE* pValue,
    ULONG Length,
    BOOL Priority
);
```

- Renamed **ReadAxisSdoObject** to **RtecatReadAxisSdoObject**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *Index* and added parameters *Axis* and *Priority*.
- Renamed parameter *Value* to *pValue*.
- Changed the data type of *ObIndex* from int to WORD.
- Changed the data type of *ObSubIndex* from int to BYTE.
- Changed the data type of *Length* from int to ULONG.

ReadIOSdoObject

```
KsCommandStatus ReadIOSdoObject(
    int Index,
    int ObIndex,
    int ObSubIndex,
    BOOL CompleteAccess,
    BYTE* Value,
    int Length
);
```

RtecatReadModuleSdoObject

```
RTECAT_COMMAND_STATUS
RtecatReadModuleSdoObject(
    RTECAT_HANDLE Module,
    WORD ObIndex,
    BYTE ObSubIndex,
    BOOL CompleteAccess,
    BYTE* pValue,
    ULONG Length,
    BOOL Priority
);
```

- Renamed **ReadIOSdoObject** to **RtecatReadModuleSdoObject**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *Index* and added parameters *Module* and *Priority*.
- Renamed parameter *Value* to *pValue*.
- Changed the data type of *ObIndex* from int to WORD.
- Changed the data type of *ObSubIndex* from int to BYTE.
- Changed the data type of *Length* from int to ULONG.

RegisterMailboxCallback

```
KsError RegisterMailboxCallback(  
    MailboxCallback callback,  
    PVOID context  
);
```

RtecatRegisterMailboxCallback

```
RTECAT_ERROR RtecatRegisterMailboxCallback(  
    (  
        RTECAT_HANDLE MainDevice,  
        MAILBOX_CALLBACK UserCallback,  
        VOID* pContext  
    )  
);
```

- Renamed **RegisterMailboxCallback** to **RtecatRegisterMailboxCallback**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Added parameter *MainDevice*.
 - Renamed parameter *callback* to *UserCallback* and changed its data type from **MailboxCallback** to **MAILBOX_CALLBACK**.
 - Renamed parameter *context* to *pContext* and changed its data type from **PVOID** to **VOID***.
-

SoeReadSsc

```
KsCommandStatus SoeReadSsc(
    int SlaveId,
    BYTE DriveNo,
    BYTE ElementFlags,
    USHORT Idn,
    UCHAR* Value,
    ULONG Length
);
```

RtecatSoeReadSsc

```
RTECAT_COMMAND_STATUS RtecatSoeReadSsc(
    RTECAT_HANDLE SubDevice,
    BYTE DriveNo,
    BYTE ElementFlags,
    USHORT Idn,
    UCHAR* pValue,
    ULONG Length
);
```

- Renamed **SoeReadSsc** to **RtecatSoeReadSsc**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *Value* to *pValue*.

SoeWriteSsc

```
KsCommandStatus SoeWriteSsc(
    int SlaveId,
    BYTE DriveNo,
    BYTE ElementFlags,
    USHORT Idn,
    UCHAR* Value,
    ULONG Length
);
```

RtecatSoeWriteSsc

```
RTECAT_COMMAND_STATUS RtecatSoeWriteSsc(
    RTECAT_HANDLE SubDevice,
    BYTE DriveNo,
    BYTE ElementFlags,
    USHORT Idn,
    const UCHAR* pValue,
    ULONG Length
);
```

- Renamed **SoeWriteSsc** to **RtecatSoeWriteSsc**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *Value* to *pValue* and changed its data type from **UCHAR*** to **const UCHAR***.

VoeWrite

```
KsCommandStatus VoeWrite(  
    int SlaveId,  
    ULONG VendorId,  
    USHORT VendorType,  
    UCHAR* Data,  
    ULONG Length  
);
```

RtecatVoeWrite

```
RTECAT_COMMAND_STATUS RtecatVoeWrite(  
    RTECAT_HANDLE SubDevice,  
    DWORD VendorId,  
    USHORT VendorType,  
    const UCHAR* pData,  
    ULONG Length  
);
```

- Renamed **VoeWrite** to **RtecatVoeWrite**.
 - Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
 - Removed parameter *SlaveId* and added parameter *SubDevice*.
 - Changed the data type of *VendorId* from ULONG to DWORD.
 - Renamed parameter *Data* to *pData* and changed its data type from UCHAR* to const UCHAR*.
-

WriteAxisSdoObject

```

KsCommandStatus
WriteAxisSdoObject(
    int Index,
    int ObIndex,
    int ObSubIndex,
    BOOL CompleteAccess,
    BYTE* Value,
    int Length
);

```

RtecatWriteAxisSdoObject

```

RTECAT_COMMAND_STATUS
RtecatWriteAxisSdoObject(
    RTECAT_HANDLE Axis,
    WORD ObIndex,
    BYTE ObSubIndex,
    BOOL CompleteAccess,
    const BYTE* pValue,
    ULONG Length,
    BOOL Priority
);

```

- Renamed **WriteAxisSdoObject** to **RtecatWriteAxisSdoObject**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *Index* and added parameters *Axis* and *Priority*.
- Renamed parameter *Value* to *pValue* and changed its data type from **BYTE*** to **const BYTE***.
- Changed the data type of *ObIndex* from **int** to **WORD**.
- Changed the data type of *ObSubIndex* from **int** to **BYTE**.
- Changed the data type of *Length* from **int** to **ULONG**.

WriteIOSdoObject

```
KsCommandStatus WriteIOSdoObject(
    int Index,
    int ObIndex,
    int ObSubIndex,
    BOOL CompleteAccess,
    BYTE* Value,
    int Length
);
```

RtecatWriteModuleSdoObject

```
RTECAT_COMMAND_STATUS
RtecatWriteModuleSdoObject(
    RTECAT_HANDLE Module,
    WORD ObIndex,
    BYTE ObSubIndex,
    BOOL CompleteAccess,
    const BYTE* pValue,
    ULONG Length,
    BOOL Priority
);
```

- Renamed **WriteIOSdoObject** to **RtecatWriteModuleSdoObject**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *Index* and added parameters *Module* and *Priority*.
- Renamed parameter *Value* to *pValue* and changed its data type from **BYTE*** to **const BYTE***.
- Changed the data type of *ObIndex* from **int** to **WORD**.
- Changed the data type of *ObSubIndex* from **int** to **BYTE**.
- Changed the data type of *Length* from **int** to **ULONG**.

MainDevice APIs

Create

```
KsError Create(  
    int Instance,  
    int IdealProcessor  
);
```

RtecatCreateMainDevice

```
RTECAT_ERROR RtecatCreateMainDevice(  
    RTECAT_HANDLE* pMainDevice,  
    ULONG Instance  
);
```

- Renamed **Create** to **RtecatCreateMainDevice**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Added parameter *pMainDevice*.
 - Changed the data type of *Instance* from **int** to **ULONG**.
 - Removed parameter *IdealProcessor*.
-

Destroy

```
KsError Destroy(  
);
```

RtecatDestroyMainDevice

```
RTECAT_ERROR RtecatDestroyMainDevice(  
    RTECAT_HANDLE MainDevice  
);
```

- Renamed **Destroy** to **RtecatDestroyMainDevice**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Added parameter *MainDevice*.
-

GetDCSystemTime

```
KsError GetDCSystemTime(
    DcCycle Cycle,
    DWORD64* Time
);
```

RtecatGetDCSystemTime

```
RTECAT_ERROR RtecatGetDCSystemTime(
    RTECAT_HANDLE MainDevice,
    RTECAT_DC_CYCLE Cycle,
    DWORD64* pTime
);
```

- Renamed **GetDCSystemTime** to **RtecatGetDCSystemTime**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Added parameter *MainDevice*.
- Changed the data type of *Cycle* from **DcCycle** to **RTECAT_DC_CYCLE**.
- Renamed parameter *Time* to *pTime*.

GetFeatureStatus

```
KsError GetFeatureStatus(
    KsFeature Feature,
    double* Status
);
```

RtecatGetFeatureStatus

```
RTECAT_ERROR RtecatGetFeatureStatus(
    RTECAT_HANDLE MainDevice,
    RTECAT_FEATURE Feature,
    double* pStatus
);
```

- Renamed **GetFeatureStatus** to **RtecatGetFeatureStatus**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Added parameter *MainDevice*.
- Changed the data type of *Feature* from **KsFeature** to **RTECAT_FEATURE**.
- Renamed parameter *Status* to *pStatus*.

GetStatus

```
KsError GetStatus(  
    SubsystemStatus* Status,  
    SubsystemDiagnostics*  
Diagnostics  
);
```

RtecatGetStatus

```
RTECAT_ERROR RtecatGetStatus(  
    RTECAT_HANDLE MainDevice,  
    RTECAT_MAINDEVICE_STATUS* pStatus,  
    RTECAT_MAINDEVICE_DIAGNOSTICS*  
pDiagnostics  
);
```

- Renamed **GetStatus** to **RtecatGetStatus**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Added parameter *MainDevice*.
 - Renamed parameter *Status* to *pStatus* and changed its data type from **SubsystemStatus*** to **RTECAT_MAINDEVICE_STATUS***.
 - Renamed parameter *Diagnostics* to *pDiagnostics* and changed its data type from **SubsystemDiagnostics*** to **RTECAT_MAINDEVICE_DIAGNOSTICS***.
-

PulseHeartbeat

```
KsError PulseHeartbeat(  
);
```

RtecatPulseHeartbeat

```
RTECAT_ERROR RtecatPulseHeartbeat(  
    RTECAT_HANDLE MainDevice  
);
```

- Renamed **PulseHeartbeat** to **RtecatPulseHeartbeat**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Added parameter *MainDevice*.
-

RegisterCallback

```
KsError RegisterCallback(  
    AppCallback callback,  
    PVOID context  
);
```

RtecatRegisterCallbackOnReceive

```
RTECAT_ERROR  
RtecatRegisterCallbackOnReceive(  
    RTECAT_HANDLE MainDevice,  
    APP_CALLBACK UserCallback,  
    VOID* pContext,  
    ULONG IdealProcessor,  
    ULONG Priority  
);
```

- Renamed **RegisterCallback** to **RtecatRegisterCallbackOnReceive**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Added parameters *MainDevice*, *IdealProcessor*, and *Priority*.
 - Renamed parameter *callback* to *UserCallback* and changed its data type from **AppCallback** to **APP_CALLBACK**.
 - Renamed parameter *context* to *pContext* and changed its data type from **PVOID** to **VOID***.
-

RegisterCallbackOnTransmit

```
KsError  
RegisterCallbackOnTransmit(  
    AppCallback callback,  
    PVOID context,  
    int IdealProcessor,  
    int Priority  
);
```

RtecatRegisterCallbackOnTransmit

```
RTECAT_ERROR  
RtecatRegisterCallbackOnTransmit(  
    RTECAT_HANDLE MainDevice,  
    APP_CALLBACK UserCallback,  
    VOID* pContext,  
    ULONG IdealProcessor,  
    ULONG Priority  
);
```

- Renamed **RegisterCallbackOnTransmit** to **RtecatRegisterCallbackOnTransmit**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Added parameter *MainDevice*.
- Renamed parameter *callback* to *UserCallback* and changed its data type from **AppCallback** to **APP_CALLBACK**.
- Renamed parameter *context* to *pContext* and changed its data type from **PVOID** to **VOID***.
- Changed the data type of *IdealProcessor* and *Priority* from **int** to **ULONG**.

RequestState

```
KsCommandStatus RequestState(  
    EthercatState State  
);
```

RtecatRequestState

```
RTECAT_COMMAND_STATUS RtecatRequestState(  
    RTECAT_HANDLE MainDevice,  
    RTECAT_ETHERCAT_STATE State  
);
```

- Renamed **RequestState** to **RtecatRequestState**.
 - Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
 - Added parameter *MainDevice*.
 - Changed the data type of *State* from **EthercatState** to **RTECAT_ETHERCAT_STATE**.
-

Restart

```
KsCommandStatus Restart(  
);
```

RtecatRestart

```
RTECAT_COMMAND_STATUS RtecatRestart(  
    RTECAT_HANDLE MainDevice  
);
```

- Renamed **Restart** to **RtecatRestart**.
 - Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
 - Added parameter *MainDevice*.
-

SetFeatureStatus

```
KsError SetFeatureStatus(  
    KsFeature Feature,  
    double Status  
);
```

RtecatSetFeatureStatus

```
RTECAT_ERROR RtecatSetFeatureStatus(  
    RTECAT_HANDLE MainDevice,  
    RTECAT_FEATURE Feature,  
    double Status  
);
```

- Renamed **SetFeatureStatus** to **RtecatSetFeatureStatus**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Added parameter *MainDevice*.
- Changed the data type of *Feature* from **KsFeature** to **RTECAT_FEATURE**.

SetHeartbeat

```
KsError SetHeartbeat(  
    BOOL Enable,  
    double Timeout  
);
```

RtecatSetHeartbeat

```
RTECAT_ERROR RtecatSetHeartbeat(  
    RTECAT_HANDLE MainDevice,  
    BOOL Enable,  
    double Timeout  
);
```

- Renamed **SetHeartbeat** to **RtecatSetHeartbeat**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Added parameter *MainDevice*.
-

Start

```
KsCommandStatus Start(  
);
```

RtecatStart

```
RTECAT_COMMAND_STATUS RtecatStart(  
    RTECAT_HANDLE MainDevice  
);
```

- Renamed **SetHeartbeat** to **RtecatStart**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Added parameter *MainDevice*.

StartFromConfiguration

```
KsCommandStatus  
StartFromConfiguration(  
    char* Configuration  
);
```

RtecatStartFromConfiguration

```
RTECAT_COMMAND_STATUS  
RtecatStartFromConfiguration(  
    RTECAT_HANDLE MainDevice,  
    const wchar_t* pConfiguration  
);
```

- Renamed **StartFromConfiguration** to **RtecatStartFromConfiguration**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Added parameter *MainDevice*.
- Renamed parameter *Configuration* to *pConfiguration* and changed its data type from **char** to **const wchar_t**.

KINGSTAR 4.x API

wRTOS 1.0 and later

Stop

```
KsCommandStatus Stop(  
);
```

RtecatStop

```
RTECAT_COMMAND_STATUS RtecatStop(  
    RTECAT_HANDLE MainDevice  
);
```

- Renamed **Stop** to **RtecatStop**.
 - Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
 - Added parameter *MainDevice*.
-

Module Control APIs

KINGSTAR 4.x API

wRTOS 1.0 and later

ForceInputBit

```
KsError ForceInputBit(  
    int Index,  
    int BitOffset,  
    BOOL Value  
);
```

RtecatForceModuleInputBit

```
RTECAT_ERROR RtecatForceModuleInputBit(  
    RTECAT_HANDLE Module,  
    ULONG BitOffset,  
    BOOL Value  
);
```

- Renamed **ForceInputBit** to **RtecatForceModuleInputBit**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Removed parameter *Index* and added parameter *Module*.
 - Changed the data type of *BitOffset* from **int** to **ULONG**.
-

ForceInputByte

```
KsError ForceInputByte(
    int Index,
    int ByteOffset,
    BYTE Value
);
```

RtecatForceModuleInputByte

```
RTECAT_ERROR RtecatForceModuleInputByte(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    BYTE Value
);
```

- Renamed **ForceInputByte** to **RtecatForceModuleInputByte**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from int to ULONG.

ForceInputDWord

```
KsError ForceInputDWord(
    int Index,
    int ByteOffset,
    DWORD Value
);
```

RtecatForceModuleInputDWord

```
RTECAT_ERROR RtecatForceModuleInputDWord(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    DWORD Value
);
```

- Renamed **ForceInputDWord** to **RtecatForceModuleInputDWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from int to ULONG.

ForceInputWord

```
KsError ForceInputWord(
    int Index,
    int ByteOffset,
    WORD Value
);
```

RtecatForceModuleInputWord

```
RTECAT_ERROR RtecatForceModuleInputWord(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    WORD Value
);
```

- Renamed **ForceInputWord** to **RtecatForceModuleInputWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.

GetIOBuffers

```
KsError GetIOBuffers(
    int Index,
    void** InputBuffer,
    void** OutputBuffer
);
```

RtecatGetModuleBuffers

```
RTECAT_ERROR RtecatGetModuleBuffers(
    RTECAT_HANDLE Module,
    void** pInputBuffer,
    void** pOutputBuffer
);
```

- Renamed **GetIOBuffers** to **RtecatGetModuleBuffers**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Renamed parameters *InputBuffer* to *pInputBuffer* and *OutputBuffer* to *pOutputBuffer*.

ReadInputBit

```
KsError ReadInputBit(
    int Index,
    int BitOffset,
    BOOL* Value
);
```

RtecatReadModuleInputBit

```
RTECAT_ERROR RtecatReadModuleInputBit(
    RTECAT_HANDLE Module,
    ULONG BitOffset,
    BOOL* pValue
);
```

- Renamed **ReadInputBit** to **RtecatReadModuleInputBit**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *BitOffset* from **int** to **ULONG**.
- Renamed parameter *Value* to *pValue*.

ReadInputByte

```
KsError ReadInputByte(
    int Index,
    int ByteOffset,
    BYTE* Value
);
```

RtecatReadModuleInputByte

```
RTECAT_ERROR RtecatReadModuleInputByte(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    BYTE* pValue
);
```

- Renamed **ReadInputByte** to **RtecatReadModuleInputByte**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.
- Renamed parameter *Value* to *pValue*.

ReadInputDWord

```
KsError ReadInputDWord(
    int Index,
    int ByteOffset,
    DWORD* Value
);
```

RtecatReadModuleInputDWord

```
RTECAT_ERROR RtecatReadModuleInputDWord(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    DWORD* pValue
);
```

- Renamed **ReadInputDWord** to **RtecatReadModuleInputDWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from int to ULONG.
- Renamed parameter *Value* to *pValue*.

ReadInputWord

```
KsError ReadInputWord(
    int Index,
    int ByteOffset,
    WORD* Value
);
```

RtecatReadModuleInputWord

```
RTECAT_ERROR RtecatReadModuleInputWord(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    WORD* pValue
);
```

- Renamed **ReadInputWord** to **RtecatReadModuleInputWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from int to ULONG.
- Renamed parameter *Value* to *pValue*.

ReadOutputBit

```
KsError ReadOutputBit(
    int Index,
    int BitOffset,
    BOOL* Value
);
```

RtecatReadModuleOutputBit

```
RTECAT_ERROR RtecatReadModuleOutputBit(
    RTECAT_HANDLE Module,
    ULONG BitOffset,
    BOOL* pValue
);
```

- Renamed **ReadOutputBit** to **RtecatReadModuleOutputBit**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *BitOffset* from **int** to **ULONG**.
- Renamed parameter *Value* to *pValue*.

ReadOutputByte

```
KsError ReadOutputByte(
    int Index,
    int ByteOffset,
    BYTE* Value
);
```

RtecatReadModuleOutputByte

```
RTECAT_ERROR RtecatReadModuleOutputByte(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    BYTE* pValue
);
```

- Renamed **ReadOutputByte** to **RtecatReadModuleOutputByte**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.
- Renamed parameter *Value* to *pValue*.

ReadOutputDWord

```
KsError ReadOutputDWord(
    int Index,
    int ByteOffset,
    DWORD* Value
);
```

RtecatReadModuleOutputDWord

```
RTECAT_ERROR RtecatReadModuleOutputDWord(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    DWORD* pValue
);
```

- Renamed **ReadOutputDWord** to **RtecatReadModuleOutputDWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from int to ULONG.
- Renamed parameter *Value* to *pValue*.

ReadOutputWord

```
KsError ReadOutputWord(
    int Index,
    int ByteOffset,
    WORD* Value
);
```

RtecatReadModuleOutputWord

```
RTECAT_ERROR RtecatReadModuleOutputWord(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    WORD* pValue
);
```

- Renamed **ReadOutputWord** to **RtecatReadModuleOutputWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from int to ULONG.
- Renamed parameter *Value* to *pValue*.

WriteOutputBit

```
KsError WriteOutputBit(
    int Index,
    int BitOffset,
    BOOL Value
);
```

RtecatWriteModuleOutputBit

```
RTECAT_ERROR RtecatWriteModuleOutputBit(
    RTECAT_HANDLE Module,
    ULONG BitOffset,
    BOOL Value
);
```

- Renamed **WriteOutputBit** to **RtecatWriteModuleOutputBit**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *BitOffset* from **int** to **ULONG**.

WriteOutputByte

```
KsError WriteOutputByte(
    int Index,
    int ByteOffset,
    BYTE Value
);
```

RtecatWriteModuleOutputByte

```
RTECAT_ERROR RtecatWriteModuleOutputByte(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    BYTE Value
);
```

- Renamed **WriteOutputByte** to **RtecatWriteModuleOutputByte**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from **int** to **ULONG**.

WriteOutputDWord

```
KsError WriteOutputDWord(
    int Index,
    int ByteOffset,
    DWORD Value
);
```

RtecatWriteModuleOutputDWord

```
RTECAT_ERROR RtecatWriteModuleOutputDWord(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    DWORD Value
);
```

- Renamed **WriteOutputDWord** to **RtecatWriteModuleOutputDWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from int to ULONG.

WriteOutputWord

```
KsError WriteOutputWord(
    int Index,
    int ByteOffset,
    WORD Value
);
```

RtecatWriteModuleOutputWord

```
RTECAT_ERROR RtecatWriteModuleOutputWord(
    RTECAT_HANDLE Module,
    ULONG ByteOffset,
    WORD Value
);
```

- Renamed **WriteOutputWord** to **RtecatWriteModuleOutputWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *Index* and added parameter *Module*.
- Changed the data type of *ByteOffset* from int to ULONG.

Shared Memory APIs

GetMemoryPointer

```
KsError GetMemoryPointer(  
    void** ppMemory  
);
```

RtecatGetMemoryPointer

```
RTECAT_ERROR RtecatGetMemoryPointer(  
    void** pMemory  
);
```

- Renamed **GetMemoryPointer** to **RtecatGetMemoryPointer**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Renamed parameter *ppMemory* to *pMemory*.

ReadMemoryBit

```
KsError ReadMemoryBit(  
    int Offset,  
    int BitOffset,  
    BOOL* Value  
);
```

RtecatReadMemoryBit

```
RTECAT_ERROR RtecatReadMemoryBit(  
    ULONG Offset,  
    ULONG BitOffset,  
    BOOL* pValue  
);
```

- Renamed **ReadMemoryBit** to **RtecatReadMemoryBit**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Changed the data type of *Offset* and *BitOffset* from **int** to **ULONG**.
 - Renamed parameter *Value* to *pValue*.
-

ReadMemoryByte

```
KsError ReadMemoryByte(  
    int Offset,  
    BYTE* Value  
);
```

RtecatReadMemoryByte

```
RTECAT_ERROR RtecatReadMemoryByte(  
    ULONG Offset,  
    BYTE* pValue  
);
```

- Renamed **ReadMemoryByte** to **RtecatReadMemoryByte**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Changed the data type of *Offset* from **int** to **ULONG**.
 - Renamed parameter *Value* to *pValue*.
-

ReadMemoryDWord

```
KsError ReadMemoryDWord(  
    int Offset,  
    DWORD* Value  
);
```

RtecatReadMemoryDWord

```
RTECAT_ERROR RtecatReadMemoryDWord(  
    ULONG Offset,  
    DWORD* pValue  
);
```

- Renamed **ReadMemoryDWord** to **RtecatReadMemoryDWord**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Changed the data type of *Offset* from **int** to **ULONG**.
 - Renamed parameter *Value* to *pValue*.
-

ReadMemoryLWord

```
KsError ReadMemoryLWord(  
    int Offset,  
    nsigned long long* Value  
);
```

RtecatReadMemoryLWord

```
RTECAT_ERROR RtecatReadMemoryLWord(  
    ULONG Offset,  
    unsigned long long* pValue  
);
```

- Renamed **ReadMemoryLWord** to **RtecatReadMemoryLWord**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Changed the data type of *Offset* from int to ULONG.
 - Renamed parameter *Value* to *pValue*.
-

ReadMemoryWord

```
KsError ReadMemoryWord(  
    int Offset,  
    WORD* Value  
);
```

RtecatReadMemoryWord

```
RTECAT_ERROR RtecatReadMemoryWord(  
    ULONG Offset,  
    WORD* pValue  
);
```

- Renamed **ReadMemoryWord** to **RtecatReadMemoryWord**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Changed the data type of *Offset* from int to ULONG.
 - Renamed parameter *Value* to *pValue*.
-

WriteMemoryBit

```
KsError WriteMemoryBit(  
    int Offset,  
    int BitOffset,  
    BOOL Value  
);
```

RtecatWriteMemoryBit

```
RTECAT_ERROR RtecatWriteMemoryBit(  
    ULONG Offset,  
    ULONG BitOffset,  
    BOOL Value  
);
```

- Renamed **WriteMemoryBit** to **RtecatWriteMemoryBit**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Changed the data type of *Offset* and *BitOffset* from int to ULONG.

WriteMemoryByte

```
KsError WriteMemoryByte(  
    int Offset,  
    BYTE Value  
);
```

RtecatWriteMemoryByte

```
RTECAT_ERROR RtecatWriteMemoryByte(  
    ULONG Offset,  
    BYTE Value  
);
```

- Renamed **WriteMemoryByte** to **RtecatWriteMemoryByte**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Changed the data type of *Offset* from int to ULONG.

WriteMemoryDWord

```
KsError WriteMemoryDWord(  
    int Offset,  
    DWORD Value  
);
```

RtecatWriteMemoryDWord

```
RTECAT_ERROR RtecatWriteMemoryDWord(  
    ULONG Offset,  
    DWORD Value  
);
```

- Renamed **WriteMemoryDWord** to **RtecatWriteMemoryDWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Changed the data type of *Offset* from **int** to **ULONG**.

WriteMemoryLWord

```
KsError WriteMemoryLWord(  
    int Offset,  
    unsigned long long Value  
);
```

RtecatWriteMemoryLWord

```
RTECAT_ERROR RtecatWriteMemoryLWord(  
    ULONG Offset,  
    unsigned long long Value  
);
```

- Renamed **WriteMemoryLWord** to **RtecatWriteMemoryLWord**.
 - Changed the return type from **KsError** to **RTECAT_ERROR**.
 - Changed the data type of *Offset* from **int** to **ULONG**.
-

KINGSTAR 4.x API

wRTOS 1.0 and later

WriteMemoryWord

```
KsError WriteMemoryWord(  
    int Offset,  
    WORD Value  
);
```

RtecatWriteMemoryWord

```
RTECAT_ERROR RtecatWriteMemoryWord(  
    ULONG Offset,  
    WORD Value  
);
```

- Renamed **WriteMemoryWord** to **RtecatWriteMemoryWord**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Changed the data type of *Offset* from **int** to **ULONG**.

Simulated Hardware APIs

ConfigureAxis

```
KsError ConfigureAxis(
    int Index,
    SlaveStatus Details,
    int Resolution
);
```

RtecatConfigureAxis

```
RTECAT_ERROR RtecatConfigureAxis(
    RTECAT_HANDLE MainDevice,
    ULONG Index,
    RTECAT_SUBDEVICE_STATUS Details,
    ULONG Resolution
);
```

- Renamed **ConfigureAxis** to **RtecatConfigureAxis**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Added parameter *MainDevice*.
- Changed the data type of *Index* and *Resolution* from **int** to **ULONG**.
- Changed the data type of *Details* from **SlaveStatus** to **RTECAT_SUBDEVICE_STATUS**.

ConfigureIo

```
KsError ConfigureIo(
    int Index,
    SlaveStatus Details
);
```

RtecatConfigureIo

```
RTECAT_ERROR RtecatConfigureIo(
    RTECAT_HANDLE MainDevice,
    ULONG Index,
    RTECAT_SUBDEVICE_STATUS Details
);
```

- Renamed **ConfigureIo** to **RtecatConfigureIo**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Added parameter *MainDevice*.
- Changed the data type of *Index* from **int** to **ULONG**.
- Changed the data type of *Details* from **SlaveStatus** to **RTECAT_SUBDEVICE_STATUS**.

GetConfiguredAxesCount

```
KsError GetConfiguredAxesCount(
    int* Count
);
```

RtecatGetConfiguredAxisCount

```
RTECAT_ERROR RtecatGetConfiguredAxisCount(
    RTECAT_HANDLE MainDevice,
    ULONG* pCount
);
```

- Renamed **GetConfiguredAxesCount** to **RtecatGetConfiguredAxisCount**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Added parameter *MainDevice*.
- Changed the parameter's name from *Count* to *pCount* and its data type from *int** to *ULONG**.

GetConfiguredIoCount

```
KsError GetConfiguredIoCount(
    int* Count
);
```

RtecatGetConfiguredIoCount

```
RTECAT_ERROR RtecatGetConfiguredIoCount(
    RTECAT_HANDLE MainDevice,
    ULONG* pCount
);
```

- Renamed **GetConfiguredIoCount** to **RtecatGetConfiguredIoCount**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Added parameter *MainDevice*.
- Changed the parameter's name from *Count* to *pCount* and its data type from *int** to *ULONG**.

SetConfiguredAxesCount

```
KsError SetConfiguredAxesCount(
    int Count
);
```

RtecatSetConfiguredAxisCount

```
RTECAT_ERROR RtecatSetConfiguredAxisCount(
    RTECAT_HANDLE MainDevice,
    ULONG Count
);
```

- Renamed **SetConfiguredAxesCount** to **RtecatSetConfiguredAxisCount**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Added parameter *MainDevice*.
- Changed the data type of parameter *Count* from int to ULONG.

SetConfiguredIoCount

```
KsError SetConfiguredIoCount(
    int Count
);
```

RtecatSetConfiguredIoCount

```
RTECAT_ERROR RtecatSetConfiguredIoCount(
    RTECAT_HANDLE MainDevice,
    ULONG Count
);
```

- Renamed **SetConfiguredIoCount** to **RtecatSetConfiguredIoCount**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Added parameter *MainDevice*.
- Changed the data type of parameter *Count* from int to ULONG.

SubDevice APIs

OverrideSlavePdoConfiguration

```
KsError
OverrideSlavePdoConfiguration(
    int SlaveId,
    int InputLength,
    int OutputLength
);
```

RtecatOverrideSubDevicePdoConfiguration

```
RTECAT_ERROR
RtecatOverrideSubDevicePdoConfiguration(
    RTECAT_HANDLE SubDevice,
    ULONG InputLength,
    ULONG OutputLength
);
```

- Renamed **OverrideSlavePdoConfiguration** to **RtecatOverrideSubDevicePdoConfiguration**.
- Changed the return type from **KsError** to **RTECAT_ERROR**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Changed the data type of *InputLength* and *OutputLength* from int to ULONG.

ReadAllSlaveEeprom

```
KsCommandStatus
ReadAllSlaveEeprom(
    int SlaveId,
    char* File
);
```

RtecatReadAllSubDeviceEeprom

```
RTECAT_COMMAND_STATUS
RtecatReadAllSubDeviceEeprom(
    RTECAT_HANDLE SubDevice,
    const wchar_t* pFile
);
```

- Renamed **ReadAllSlaveEeprom** to **RtecatReadAllSubDeviceEeprom**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *File* to *pFile* and changed its data type from char* to const wchar_t*.

ReadSlaveAlias

```
KsCommandStatus ReadSlaveAlias(
    int SlaveId,
    WORD* Alias
);
```

RtecatReadSubDeviceAlias

```
RTECAT_COMMAND_STATUS
RtecatReadSubDeviceAlias(
    RTECAT_HANDLE SubDevice,
    WORD* pAlias
);
```

- Renamed **ReadSlaveAlias** to **RtecatReadSubDeviceAlias**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *Alias* to *pAlias*.

ReadSlaveEeprom

```
KsCommandStatus ReadSlaveEeprom(
    int SlaveId,
    int Offset,
    DWORD* Data
);
```

RtecatReadSubDeviceEeprom

```
RTECAT_COMMAND_STATUS
RtecatReadSubDeviceEeprom(
    RTECAT_HANDLE SubDevice,
    ULONG Offset,
    DWORD* pData
);
```

- Renamed **ReadSlaveEeprom** to **RtecatReadSubDeviceEeprom**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Changed the data type of *Offset* from int to ULONG.
- Renamed parameter *Data* to *pData*.

ReadSlaveRegister

```
KsCommandStatus
ReadSlaveRegister(
    int SlaveId,
    int Offset,
    int Length,
    BYTE* Data
);
```

RtecatReadSubDeviceRegister

```
RTECAT_COMMAND_STATUS
RtecatReadSubDeviceRegister(
    RTECAT_HANDLE SubDevice,
    USHORT Offset,
    USHORT Length,
    BYTE* pData
);
```

- Renamed **ReadSlaveRegister** to **RtecatReadSubDeviceRegister**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Changed the data type of *Offset* and *Length* from int to USHORT.
- Renamed parameter *Data* to *pData*.

ReloadSlaveEeprom

```
KsCommandStatus
ReloadSlaveEeprom(
    int SlaveId
);
```

RtecatReloadSubDeviceEeprom

```
RTECAT_COMMAND_STATUS
RtecatReloadSubDeviceEeprom(
    RTECAT_HANDLE SubDevice
);
```

- Renamed **ReloadSlaveEeprom** to **RtecatReloadSubDeviceEeprom**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.

RequestSlaveState

```
KsCommandStatus
RequestSlaveState(
    int SlaveId,
    EthercatState State
);
```

RtecatRequestSubDeviceState

```
RTECAT_COMMAND_STATUS
RtecatRequestSubDeviceState(
    RTECAT_HANDLE SubDevice,
    RTECAT_ETHERCAT_STATE State
);
```

- Renamed **RequestSlaveState** to **RtecatRequestSubDeviceState**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Changed the data type of *State* from **EthercatState** to **RTECAT_ETHERCAT_STATE**.

ResetSlaveEsc

```
KsCommandStatus ResetSlaveEsc(
    int SlaveId
);
```

RtecatResetSubDeviceEsc

```
RTECAT_COMMAND_STATUS
RtecatResetSubDeviceEsc(
    RTECAT_HANDLE SubDevice
);
```

- Renamed **ResetSlaveEsc** to **RtecatResetSubDeviceEsc**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.

RestartSlave

```
KsCommandStatus RestartSlave(
    int SlaveId
);
```

RtecatRestartSubDevice

```
RTECAT_COMMAND_STATUS RtecatRestartSubDevice(
    RTECAT_HANDLE SubDevice
);
```

- Renamed **RestartSlave** to **RtecatRestartSubDevice**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.

WriteAllSlaveEEprom

```
KsCommandStatus
WriteAllSlaveEEprom(
    int SlaveId,
    char* File
);
```

RtecatWriteAllSubDeviceEEprom

```
RTECAT_COMMAND_STATUS
RtecatWriteAllSubDeviceEEprom(
    RTECAT_HANDLE SubDevice,
    const wchar_t* pFile
);
```

- Renamed **WriteAllSlaveEEprom** to **RtecatWriteAllSubDeviceEEprom**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Renamed parameter *File* to *pFile* and changed its data type from **char*** to **const wchar_t***.

WriteSlaveAlias

```
KsCommandStatus WriteSlaveAlias(
    int SlaveId,
    WORD Alias
);
```

RtecatWriteSubDeviceAlias

```
RTECAT_COMMAND_STATUS
RtecatWriteSubDeviceAlias(
    RTECAT_HANDLE SubDevice,
    WORD Alias
);
```

- Renamed **WriteSlaveAlias** to **RtecatWriteSubDeviceAlias**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.

WriteSlaveEeprom

```
KsCommandStatus WriteSlaveEeprom
(
    int SlaveId,
    int Offset,
    DWORD Data
);
```

RtecatWriteSubDeviceEeprom

```
RTECAT_COMMAND_STATUS
RtecatWriteSubDeviceEeprom(
    RTECAT_HANDLE SubDevice,
    ULONG Offset,
    DWORD Data
);
```

- Renamed **WriteSlaveEeprom** to **RtecatWriteSubDeviceEeprom**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Changed the data type of *Offset* from int to ULONG.

WriteSlaveRegister

```
KsCommandStatus
WriteSlaveRegister(
    int SlaveId,
    int Offset,
    int Length,
    BYTE* Data
);
```

RtecatWriteSubDeviceRegister

```
RTECAT_COMMAND_STATUS
RtecatWriteSubDeviceRegister(
    RTECAT_HANDLE SubDevice,
    USHORT Offset,
    USHORT Length,
    const BYTE* pData
);
```

- Renamed **WriteSlaveRegister** to **RtecatWriteSubDeviceRegister**.
- Changed the return type from **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Removed parameter *SlaveId* and added parameter *SubDevice*.
- Changed the data type of *Offset* and *Length* from int to USHORT.
- Renamed parameter *Data* to *pData* and changed its data type from BYTE* to const BYTE*.

E-CAT Data Types

CanOpenSetting

```
typedef struct _CanOpenSetting(
    INT RxPdoCount;
    CanPdo RxPdos[4];
    INT TxPdoCount;
    CanPdo TxPdos[4];
    INT SdoCommandCount;
    SdoCommand SdoCommands[16];
) CanOpenSetting;
```

RTECAT_CAN_SETTING

```
typedef struct _RTECAT_CAN_SETTING(
    ULONG RxPdoCount;
    RTECAT_CAN_PDO RxPdos[4];
    ULONG TxPdoCount;
    RTECAT_CAN_PDO TxPdos[4];
    ULONG SdoCommandCount;
    RTECAT_SDO_COMMAND SdoCommands[16];
) RTECAT_CAN_SETTING;
```

- Renamed **CanOpenSetting** to **RTECAT_CAN_SETTING**.
- Changed the data type of all fields.

CanPdo

```
typedef struct _CanPdo(
    BOOL Enable;
    INT Length;
    USHORT Index;
    BYTE Trigger;
) CanPdo;
```

RTECAT_CAN_PDO

```
typedef struct _RTECAT_CAN_PDO(
    BOOL Enable;
    ULONG Length;
    USHORT Index;
    BYTE Trigger;
) RTECAT_CAN_PDO;
```

- Renamed **CanPdo** to **RTECAT_CAN_PDO**.
- Changed the data type of *Length* from INT to ULONG.

EcatMdpSetting

```
typedef struct _EcatMdpSetting(
    INT InputLength;
    INT OutputLength;
    INT SdoCommandCount;
    SdoCommand SdoCommands[16];
) EcatMdpSetting;
```

RTECAT_ECATCHMP_SETTING

```
typedef struct _RTECAT_ECATCHMP_SETTING(
    ULONG InputLength;
    ULONG OutputLength;
    ULONG SdoCommandCount;
    RTECAT_SDO_COMMAND SdoCommands[16];
) RTECAT_ECATCHMP_SETTING;
```

- Renamed **EcatMdpSetting** to **RTECAT_ECATCHMP_SETTING**.
- Changed the data type of all fields.

KsCommandStatus

```
typedef struct _KsCommandStatus(
    HANDLE Handle;
    PVOID Value;
    INT ValueLength;
    BOOL Done;
    BOOL InSync;
    BOOL InVelocity;
    BOOL Busy;
    BOOL Active;
    BOOL CommandAborted;
    BOOL Error;
    KsError ErrorId;
) KsCommandStatus;
```

RTECAT_COMMAND_STATUS

```
typedef struct _RTECAT_COMMAND_STATUS(
    RTECAT_HANDLE Handle;
    PVOID Buffer;
    ULONG BufferType;
    ULONG BufferLength;
    ULONG DataLength;
    RTECAT_COMMAND_STATE State;
    RTECAT_ERROR ErrorId;
    ULONG ReturnCode;
) RTECAT_COMMAND_STATUS;
```

- Renamed **KsCommandStatus** to **RTECAT_COMMAND_STATUS**.
- Changed the data type of *Handle* from **HANDLE** to **RTECAT_HANDLE**.
- Changed the data type of *ErrorId* from **KsError** to **RTECAT_ERROR**.
- Added fields *Buffer*, *BufferType*, *BufferLength*, *DataLength*, *State*, and *ReturnCode*.
- Removed fields *Value*, *ValueLength*, *Done*, *InSync*, *InVelocity*, *Busy*, *Active*, *CommandAborted*, and *Error*.

MailboxCoeEmergency

```
typedef struct _MailboxCoeEmergency(
    USHORT ErrorCode;
    BYTE ErrorRegister;
    BYTE Data[5];
) MailboxCoeEmergency;
```

RTECAT_MAILBOX_COE_EMERGENCY

```
typedef struct _RTECAT_MAILBOX_COE_EMERGENCY(
    USHORT ErrorCode;
    BYTE ErrorRegister;
    BYTE Data[5];
) RTECAT_MAILBOX_COE_EMERGENCY;
```

- Renamed **MailboxCoeEmergency** to **RTECAT_MAILBOX_COE_EMERGENCY**.

MailboxData

```
typedef struct _MailboxData(
    MailboxCoeEmergency
CoeEmergency;
    MailboxSoeNotification
SoeNotification;
    MailboxSoeSlaveInfo
SoeSlaveInfo;
    MailboxVoe Voe;
) MailboxData;
```

RTECAT_MAILBOX_DATA

```
typedef struct _RTECAT_MAILBOX_DATA(
    RTECAT_MAILBOX_COE_EMERGENCY
CoeEmergency;
    RTECAT_MAILBOX_SOE_NOTIFICATION
SoeNotification;
    RTECAT_MAILBOX_SOE_SUBDEVICEINFO
SoeSubDeviceInfo;
    RTECAT_MAILBOX_VOE Voe;
) RTECAT_MAILBOX_DATA;
```

- Renamed **MailboxData** to **RTECAT_MAILBOX_DATA**.
- Renamed field *SoeSlaveInfo* to *SoeSubDeviceInfo*.
- Changed the data type of all fields.

MailboxSoeNotification

```
typedef struct _
MailboxSoeNotification(
    BYTE DriveNo;
    USHORT Idn;
    USHORT DataStatus;
) MailboxSoeNotification;
```

RTECAT_MAILBOX_SOE_NOTIFICATION

```
typedef struct _RTECAT_MAILBOX_SOE_
NOTIFICATION(
    BYTE DriveNo;
    USHORT Idn;
    USHORT DataStatus;
) RTECAT_MAILBOX_SOE_NOTIFICATION;
```

- Renamed **MailboxSoeNotification** to **RTECAT_MAILBOX_SOE_NOTIFICATION**.

MailboxSoeSlaveInfo

```
typedef struct _MailboxSoeSlaveInfo(
    BYTE DriveNo;
    BYTE ElementFlags;
    USHORT Idn;
    INT DataLength;
    UCHAR Data[2048];
) MailboxSoeSlaveInfo;
```

RTECAT_MAILBOX_SOE_SUBDEVICESINFO

```
typedef struct _RTECAT_MAILBOX_SOE_
SUBDEVICESINFO(
    BYTE DriveNo;
    BYTE ElementFlags;
    USHORT Idn;
    ULONG DataLength;
    UCHAR Data[2048];
) RTECAT_MAILBOX_SOE_SUBDEVICESINFO;
```

- Renamed **MailboxSoeSlaveInfo** to **RTECAT_MAILBOX_SOE_SUBDEVICESINFO**.
- Changed the data type of *DataLength* from INT to ULONG.

MailboxVoe

```
typedef struct _MailboxVoe(  
    UINT VendorId;  
    USHORT VendorType;  
    INT DataLength;  
    UCHAR Data[2048];  
) MailboxVoe;
```

RTECAT_MAILBOX_VOE

```
typedef struct _RTECAT_MAILBOX_VOE(  
    DWORD VendorId;  
    USHORT VendorType;  
    ULONG DataLength;  
    UCHAR Data[2048];  
) RTECAT_MAILBOX_VOE;
```

- Renamed **MailboxVoe** to **RTECAT_MAILBOX_VOE**.
- Changed the data type of *VendorId* from `UINT` to `DWORD`.
- Changed the data type of *DataLength* from `INT` to `ULONG`.

SdoCommand

```
typedef struct _SdoCommand(  
    USHORT Index;  
    BYTE SubIndex;  
    INT Length;  
    BYTE Data[8];  
) SdoCommand;
```

RTECAT_SDO_COMMAND

```
typedef struct _RTECAT_SDO_COMMAND(  
    WORD Index;  
    BYTE SubIndex;  
    ULONG Length;  
    BYTE Data[8];  
) RTECAT_SDO_COMMAND;
```

- Renamed **SdoCommand** to **RTECAT_SDO_COMMAND**.
 - Changed the data type of *Index* from `USHORT` to `WORD`.
 - Changed the data type of *Length* from `INT` to `ULONG`.
-

SdoEntryDescription

```
typedef struct _SdoEntryDescription(
    WORD Index;
    BYTE SubIndex;
    BYTE ValueInfo;
    WORD DataType;
    WORD BitLength;
    WORD ObjectAccess;
    BYTE Data[64];
) SdoEntryDescription;
```

RTECAT_SDO_ENTRY_DESCRIPTION

```
typedef struct _RTECAT_SDO_ENTRY_
DESCRIPTION(
    WORD Index;
    BYTE SubIndex;
    BYTE ValueInfo;
    WORD DataType;
    WORD BitLength;
    WORD ObjectAccess;
    BYTE Data[64];
) RTECAT_SDO_ENTRY_DESCRIPTION;
```

- Renamed **SdoEntryDescription** to **RTECAT_SDO_ENTRY_DESCRIPTION**.

SdoObjectDescription

```
typedef struct _SdoObjectDescription
(
    WORD Index;
    WORD DataType;
    BYTE MaxSubIndex;
    BYTE ObjectCode;
    CHAR Name[64];
) SdoObjectDescription;
```

RTECAT_SDO_OBJECT_DESCRIPTION

```
typedef struct _RTECAT_SDO_OBJECT_
DESCRIPTION(
    WORD Index;
    WORD DataType;
    BYTE MaxSubIndex;
    BYTE ObjectCode;
    CHAR Name[64];
) RTECAT_SDO_OBJECT_DESCRIPTION;
```

- Renamed **SdoObjectDescription** to **RTECAT_SDO_OBJECT_DESCRIPTION**.

SubsystemDiagnostics

```
typedef struct _SubsystemDiagnostics
(
    UINT VendorId;
    UINT ProductCode;
    UINT RevisionNo;
    UINT SerialNo;
    UINT CyclicPacketLost;
    UINT AcyclicPacketLost;
    int SlaveCount;
    SlaveDiagnostics Slaves[256];
) SubsystemDiagnostics;
```

RTECAT_MAINDEVICE_DIAGNOSTICS

```
typedef struct _RTECAT_MAINDEVICE_
DIAGNOSTICS (
    DWORD VendorId;
    DWORD ProductCode;
    DWORD RevisionNo;
    DWORD SerialNo;
    ULONG CyclicPacketLost;
    ULONG AcyclicPacketLost;
    ULONG SubDeviceCount;
    RTECAT_SUBDEVICE_DIAGNOSTICS
SubDevices[256];
) RTECAT_MAINDEVICE_DIAGNOSTICS;
```

- Renamed **SubsystemDiagnostics** to **RTECAT_MAINDEVICE_DIAGNOSTICS**.
- Renamed field *SlaveCount* to *SubDeviceCount*.
- Renamed field *Slaves* to *SubDevices*.
- Changed the data type of all fields.

SubsystemStatus

```
typedef struct _SubsystemStatus(
    EthercatState State;
    EthercatState RequestedState;
    int SlaveCount;
    int IOCount;
    int AxesCount;
    EthercatState SlaveStates[256];
    EthercatState IOStates[256];
    AxisState AxesStates[256];
) SubsystemStatus;
```

RTECAT_MAINDEVICE_STATUS

```
typedef struct _RTECAT_MAINDEVICE
STATUS(
    RTECAT_ETHERCAT_STATE State;
    RTECAT_ETHERCAT_STATE
RequestedState;
    ULONG SubDeviceCount;
    ULONG ModuleCount;
    ULONG AxisCount;
    LONGLONG
MainDeviceConfigurationStamp;
    RTECAT_ETHERCAT_STATE
SubDeviceStates[256];
    RTECAT_ETHERCAT_STATE ModuleStates
[256];
    RTECAT_ETHERCAT_STATE AxisStates
[256];
    LONGLONG
SubDeviceConfigurationStamps[256];
    LONGLONG ModuleConfigurationStamps
[256];
    LONGLONG AxisConfigurationStamps
[256];
) RTECAT_MAINDEVICE_STATUS;
```

- Renamed **SubsystemStatus** to **RTECAT_MAINDEVICE_STATUS**.
- Added fields *MainDeviceConfigurationStamp*, *SubDeviceConfigurationStamps*, *ModuleConfigurationStamps*, and *AxisConfigurationStamps*.
- Renamed field *SlaveCount* to *SubDeviceCount*.
- Renamed field *IOCount* to *ModuleCount*.
- Renamed field *AxesCount* to *AxisCount*.
- Renamed field *SlaveStates* to *SubDeviceStates*.
- Renamed field *IOStates* to *ModuleStates*.
- Renamed field *AxesStates* to *AxisStates*.

SlaveDiagnostics

```
typedef struct _SlaveDiagnostics{
    USHORT FixedAddress;
    CHAR Type[64];
    CHAR Name[64];
    UINT DeviceType;
    UINT VendorId;
    UINT ProductCode;
    UINT RevisionNo;
    UINT SerialNo;
    USHORT MailboxIn;
    USHORT MailboxOut;
    BYTE LinkStatus;
    BYTE LinkPreset;
    BYTE Flags;
    USHORT StationAddress;
    USHORT DlStatus;
    USHORT AlStatus;
    USHORT AlControl;
    USHORT AlStatusCode;
    BYTE LinkConnStatus;
    BYTE LinkConnControl;
    USHORT PortAddress[4];
    BYTE FrameCrc[4];
    BYTE PhyCrc[4];
    BYTE LostLink[4];
    UINT CyclicWc;
    UINT NotPresent;
    UINT AbnormalState;
} SlaveDiagnostics;
```

RTECAT_SUBDEVICE_DIAGNOSTICS

```
typedef struct _RTECAT_SUBDEVICE_
DIAGNOSTICS{
    USHORT FixedAddress;
    CHAR Type[64];
    CHAR Name[64];
    ULONG DeviceType;
    DWORD VendorId;
    DWORD ProductCode;
    DWORD RevisionNo;
    DWORD SerialNo;
    USHORT MailboxIn;
    USHORT MailboxOut;
    BYTE LinkStatus;
    BYTE LinkPreset;
    BYTE Flags;
    USHORT StationAddress;
    USHORT DlStatus;
    USHORT AlStatus;
    USHORT AlControl;
    USHORT AlStatusCode;
    BYTE LinkConnStatus;
    BYTE LinkConnControl;
    USHORT PortAddress[4];
    BYTE FrameCrc[4];
    BYTE PhyCrc[4];
    BYTE LostLink[4];
    ULONG CyclicWc;
    ULONG NotPresent;
    ULONG AbnormalState;
} RTECAT_SUBDEVICE_DIAGNOSTICS;
```

- Renamed **SlaveDiagnostics** to **RTECAT_SUBDEVICE_DIAGNOSTICS**.
- Changed the data type of *DeviceType*, *CyclicWc*, *NotPresent*, and *AbnormalState* from **UINT** to **ULONG**.
- Changed the data type of *VendorId*, *ProductCode*, *RevisionNo*, and *SerialNo* from **UINT** to **DWORD**.

SlaveStatus

```
typedef struct _SlaveStatus{
    CHAR Name[64];
    DWORD VendorId;
    DWORD ProductCode;
    DWORD RevisionNumber;
    DWORD SerialNumber;
    int SlaveId;
    int SlotId;
    USHORT PhysAddress;
    USHORT AliasAddress;
    USHORT ExplicitId;
    EthercatState State;
    EthercatState RequestedState;
    int InputLength;
    int OutputLength;
    int CycleTime;
    WORD VariableIndexOffset;
} SlaveStatus;
```

RTECAT_SUBDEVICE_STATUS

```
typedef struct _RTECAT_SUBDEVICE_STATUS
{
    CHAR Name[64];
    DWORD VendorId;
    DWORD ProductCode;
    DWORD RevisionNumber;
    DWORD SerialNumber;
    ULONG Index;
    ULONG PositionId;
    USHORT PhysAddress;
    USHORT AliasAddress;
    USHORT ExplicitId;
    RTECAT_ETHERCAT_STATE State;
    RTECAT_ETHERCAT_STATE
RequestedState;
    ULONG InputLength;
    ULONG OutputLength;
    ULONG CycleTime;
} RTECAT_SUBDEVICE_STATUS;
```

- Renamed **SlaveStatus** to **RTECAT_SUBDEVICE_STATUS**.
- Added fields *Index* and *PositionId*.
- Changed the data type of *State* and *RequestedState* from **EthercatState** to **RTECAT_ETHERCAT_STATE**.
- Changed the data type of *InputLength*, *OutputLength*, and *CycleTime* from int to ULONG.
- Removed fields *SlaveId*, *SlotId*, and *VariableIndexOffset*.

AoeError

```

typedef enum _AoeError{
    errAoeNoError = 0,
    errAoeInternalError = 1,
    errAoeNoRTime = 2,
    errAoeLockedMemoryError = 3,
    errAoeMailBoxError = 4,
    errAoeWrongHMsg = 5,
    errAoeTargetPortNotFound = 6,
    errAoeTargetMachineNotFound = 7,
    errAoeUnknownCommandID = 8,
    errAoeBadTaskID = 9,
    errAoeNoIO = 10,
    errAoeUnknwonAmsCommand = 11,
    errAoeWin32Error = 12,
    errAoePortNotConnected = 13,
    errAoeInvalidAmsLength = 14,
    errAoeInvalidAmsNetID = 15,
    errAoeLowInstallLevel = 16,
    errAoeNoDebug = 17,
    errAoePortDisabled = 18,
    errAoePortConnected = 19,
    errAoeAmsSyncWin32Error = 20,
    errAoeSyncTimeOut = 21,
    errAoeAmsSyncAmsError = 22,
    errAoeAmsSyncNoIndexMap = 23,
    errAoeInvalidAmsPort = 24,
    errAoeNoMemory = 25,
    errAoeTCPSendError = 26,
    errAoeHostUnreachable = 27,
    errAoeNoLockedMemory = 1280,
    errAoeMailboxFull = 1282,
    errAoeDeviceError = 1792,
    errAoeDeviceServiceNotSupported
= 1793,
    errAoeDeviceInvalidGroup = 1794,
    errAoeDeviceInvalidOffset =
1795,

```

RTECAT_AOE_ERROR

```

typedef enum _RTECAT_AOE_ERROR{
    ecatErrAoeNoError = 0,
    ecatErrAoeInternalError = 1,
    ecatErrAoeNoRTime = 2,
    ecatErrAoeLockedMemoryError = 3,
    ecatErrAoeMailBoxError = 4,
    ecatErrAoeWrongHMsg = 5,
    ecatErrAoeTargetPortNotFound = 6,
    ecatErrAoeTargetMachineNotFound =
7,
    ecatErrAoeUnknownCommandID = 8,
    ecatErrAoeBadTaskID = 9,
    ecatErrAoeNoIO = 10,
    ecatErrAoeUnknwonAmsCommand = 11,
    ecatErrAoeWin32Error = 12,
    ecatErrAoePortNotConnected = 13,
    ecatErrAoeInvalidAmsLength = 14,
    ecatErrAoeInvalidAmsNetID = 15,
    ecatErrAoeLowInstallLevel = 16,
    ecatErrAoeNoDebug = 17,
    ecatErrAoePortDisabled = 18,
    ecatErrAoePortConnected = 19,
    ecatErrAoeAmsSyncWin32Error = 20,
    ecatErrAoeSyncTimeOut = 21,
    ecatErrAoeAmsSyncAmsError = 22,
    ecatErrAoeAmsSyncNoIndexMap = 23,
    ecatErrAoeInvalidAmsPort = 24,
    ecatErrAoeNoMemory = 25,
    ecatErrAoeTCPSendError = 26,
    ecatErrAoeHostUnreachable = 27,
    ecatErrAoeNoLockedMemory = 1280,
    ecatErrAoeMailboxFull = 1282,
    ecatErrAoeDeviceError = 1792,
    ecatErrAoeDeviceServiceNotSupported
= 1793,
    ecatErrAoeDeviceInvalidGroup =
1794,
    ecatErrAoeDeviceInvalidOffset =
1795,

```

KINGSTAR 4.x API

wRTOS 1.0 and later

```
errAoeDeviceInvalidAccess =  
1796,  
errAoeDeviceInvalidSize = 1797,  
errAoeDeviceInvalidData = 1798,  
errAoeDeviceNotReady = 1799,  
errAoeDeviceBusy = 1800,  
errAoeDeviceInvalidContext =  
1801,  
errAoeDeviceNoMemory = 1802,  
errAoeDeviceInavlidParam = 1803,  
errAoeDeviceNotFound = 1804,  
errAoeDeviceSyntaxError = 1805,  
errAoeDeviceIncompatible = 1806,  
errAoeDeviceSymbolNotFound =  
1808,  
errAoeDeviceSymbolVersionInvalid  
= 1809,  
errAoeDeviceInvalidState = 1810,  
  
errAoeDeviceTransModeNotSupported =  
1811,  
errAoeDeviceNotifyHandleInvalid  
= 1812,  
errAoeDeviceClientUnknown =  
1813,  
errAoeDeviceNoMoreHandles =  
1814,  
errAoeDeviceInvalidWatchsize =  
1815,  
errAoeDeviceNotInitialized =  
1816,  
errAoeDeviceTimeOut = 1817,  
errAoeDeviceNoInterface = 1818,  
errAoeDeviceInvalidInterface =  
1819,  
errAoeDeviceInvalidCLSID = 1820,  
errAoeDeviceInvalidObjectID =  
1821,
```

```
ecatErrAoeDeviceInvalidAccess =  
1796,  
ecatErrAoeDeviceInvalidSize = 1797,  
ecatErrAoeDeviceInvalidData = 1798,  
ecatErrAoeDeviceNotReady = 1799,  
ecatErrAoeDeviceBusy = 1800,  
ecatErrAoeDeviceInvalidContext =  
1801,  
ecatErrAoeDeviceNoMemory = 1802,  
ecatErrAoeDeviceInavlidParam =  
1803,  
ecatErrAoeDeviceNotFound = 1804,  
ecatErrAoeDeviceSyntaxError = 1805,  
ecatErrAoeDeviceIncompatible =  
1806,  
ecatErrAoeDeviceSymbolNotFound =  
1808,  
ecatErrAoeDeviceSymbolVersionInvalid =  
1809,  
ecatErrAoeDeviceInvalidState =  
1810,  
ecatErrAoeDeviceTransModeNotSupported =  
1811,  
ecatErrAoeDeviceNotifyHandleInvalid  
= 1812,  
ecatErrAoeDeviceClientUnknown =  
1813,  
ecatErrAoeDeviceNoMoreHandles =  
1814,  
ecatErrAoeDeviceInvalidWatchsize =  
1815,  
ecatErrAoeDeviceNotInitialized =  
1816,  
ecatErrAoeDeviceTimeOut = 1817,  
ecatErrAoeDeviceNoInterface = 1818,  
ecatErrAoeDeviceInvalidInterface =  
1819,
```

KINGSTAR 4.x API

wRTOS 1.0 and later

```
errAoeClientError = 1856,  
errAoeClientInvalidParameter =  
1857,  
errAoeClientListEmpty = 1858,  
errAoeClientVaraiableInUse =  
1859,  
errAoeClientDuplicateInvokeID =  
1860,  
errAoeClientSyncTimeOut = 1861,  
errAoeClientW32OR = 1862,  
errAoeClientTimeoutInvalid =  
1863,  
errAoeClientPortNotOpen = 1864,  
errAoeClientNoAmsAddr = 1865,  
errAoeClientSyncInternal = 1872,  
errAoeClientAddHash = 1873,  
errAoeClientRemoveHash = 1874,  
errAoeClientNoMoreSymbols =  
1875,  
errAoeClientSyncResInvalid =  
1876,  
errAoeClientSyncPortLocked =  
1877,  
errAoeClientQueueFull = 32768  
} AoeError;
```

```
ecatErrAoeDeviceInvalidCLSID =  
1820,  
ecatErrAoeDeviceInvalidObjectID =  
1821,  
ecatErrAoeClientError = 1856,  
ecatErrAoeClientInvalidParameter =  
1857,  
ecatErrAoeClientListEmpty = 1858,  
ecatErrAoeClientVaraiableInUse =  
1859,  
ecatErrAoeClientDuplicateInvokeID =  
1860,  
ecatErrAoeClientSyncTimeOut = 1861,  
ecatErrAoeClientW32OR = 1862,  
ecatErrAoeClientTimeoutInvalid =  
1863,  
ecatErrAoeClientPortNotOpen = 1864,  
ecatErrAoeClientNoAmsAddr = 1865,  
ecatErrAoeClientSyncInternal =  
1872,  
ecatErrAoeClientAddHash = 1873,  
ecatErrAoeClientRemoveHash = 1874,  
ecatErrAoeClientNoMoreSymbols =  
1875,  
ecatErrAoeClientSyncResInvalid =  
1876,  
ecatErrAoeClientSyncPortLocked =  
1877,  
ecatErrAoeClientQueueFull = 32768  
} RTECAT_AOE_ERROR;
```

- Renamed **AoeError** to **RTECAT_AOE_ERROR**.
- Renamed the prefix of all constants from *err* to *ecatErr*.

KINGSTAR 4.x API

wRTOS 1.0 and later

AppCallback

```
typedef int (*AppCallback) (PVOID  
Context);
```

APP_CALLBACK

```
typedef int (*APP_CALLBACK) (PVOID  
Context);
```

- Renamed **AppCallback** to **APP_CALLBACK**.
-

CoeError

```

typedef enum _CoeError{
    errCoeToggleBit = 0x05030000,
    errCoeTimeout = 0x05040000,
    errCoeCommandSpecifier =
0x05040001,
    errCoeOutOfMemory = 0x05040005,
    errCoeUnsupportedAccess =
0x06010000,
    errCoeWriteOnly = 0x06010001,
    errCoeReadOnly = 0x06010002,
    errCoeSubindexReadOnly =
0x06010003,
    errCoeNoCompleteAccess =
0x06010004,
    errCoeObjectTooLong =
0x06010005,
    errCoeObjectInPdo = 0x06010006,
    errCoeObjectNotExist =
0x06020000,
    errCoeNoPdoMapping = 0x06040041,
    errCoePdoLengthExceeded =
0x06040042,
    errCoeParameterIncompatible =
0x06040043,
    errCoeInternalIncompatible =
0x06040047,
    errCoeHardwareError =
0x06060000,
    errCoeLengthIncorrect =
0x06070010,
    errCoeLengthTooHigh =
0x06070012,
    errCoeLengthTooLow = 0x06070013,
    errCoeSubindexNotExist =
0x06090011,
    errCoeValueOutOfRange =
0x06090030,

```

RTECAT_COE_ERROR

```

typedef enum _RTECAT_COE_ERROR{
    ecatErrCoeToggleBit = 0x05030000,
    ecatErrCoeTimeout = 0x05040000,
    ecatErrCoeCommandSpecifier =
0x05040001,
    ecatErrCoeOutOfMemory = 0x05040005,
    ecatErrCoeUnsupportedAccess =
0x06010000,
    ecatErrCoeWriteOnly = 0x06010001,
    ecatErrCoeReadOnly = 0x06010002,
    ecatErrCoeSubindexReadOnly =
0x06010003,
    ecatErrCoeNoCompleteAccess =
0x06010004,
    ecatErrCoeObjectTooLong =
0x06010005,
    ecatErrCoeObjectInPdo = 0x06010006,
    ecatErrCoeObjectNotExist =
0x06020000,
    ecatErrCoeNoPdoMapping =
0x06040041,
    ecatErrCoePdoLengthExceeded =
0x06040042,
    ecatErrCoeParameterIncompatible =
0x06040043,
    ecatErrCoeInternalIncompatible =
0x06040047,
    ecatErrCoeHardwareError =
0x06060000,
    ecatErrCoeLengthIncorrect =
0x06070010,
    ecatErrCoeLengthTooHigh =
0x06070012,
    ecatErrCoeLengthTooLow =
0x06070013,
    ecatErrCoeSubindexNotExist =
0x06090011,

```

KINGSTAR 4.x API

wRTOS 1.0 and later

```
errCoeValueTooHigh = 0x06090031,  
errCoeValueTooLow = 0x06090032,  
errCoeConfiguredModuleListNotMatch =  
0x06090033,  
errCoeMaxBelowMin = 0x06090036,  
errCoeGeneralError = 0x08000000,  
errCoeCannotTransfer =  
0x08000020,  
errCoeCannotTransferLocal =  
0x08000021,  
errCoeWrongState = 0x08000022,  
errCoeDictionaryNotAvailable =  
0x08000023  
} CoeError;
```

```
ecatErrCoeValueOutOfRange =  
0x06090030,  
ecatErrCoeValueTooHigh =  
0x06090031,  
ecatErrCoeValueTooLow = 0x06090032,  
ecatErrCoeConfiguredModuleListNotMatch  
= 0x06090033,  
ecatErrCoeMaxBelowMin = 0x06090036,  
ecatErrCoeGeneralError =  
0x08000000,  
ecatErrCoeCannotTransfer =  
0x08000020,  
ecatErrCoeCannotTransferLocal =  
0x08000021,  
ecatErrCoeWrongState = 0x08000022,  
ecatErrCoeDictionaryNotAvailable =  
0x08000023  
} RTECAT_COE_ERROR;
```

- Renamed **CoeError** to **RTECAT_COE_ERROR**.
- Renamed the prefix of all constants from *err* to *ecatErr*.

DcCycle

```
typedef enum _DcCycle{
    dcInput = 0,
    dcCurrent,
    dcOutput,
    dcController
} DcCycle;
```

RTECAT_DC_CYCLE

```
typedef enum _RTECAT_DC_CYCLE{
    ecatDcInput = 0,
    ecatDcCurrent,
    ecatDcOutput
} RTECAT_DC_CYCLE;
```

- Renamed **DcCycle** to **RTECAT_DC_CYCLE**.
 - Renamed constant *dcInput* to *ecatDcInput*.
 - Renamed constant *dcCurrent* to *ecatDcCurrent*.
 - Renamed constant *dcOutput* to *ecatDcOutput*.
 - Removed constant *dcController*.
-

EscError

```

typedef enum _EscError{
    errEscNoError = 0x0000,
    errEscUnspecified = 0x0001,
    errEscNoMemory = 0x0002,
    errEscInvalidStateChange =
0x0011,
    errEscUnknownRequestedState =
0x0012,
    errEscBootstrapUnsupported =
0x0013,
    errEscNoValidFirmware = 0x0014,

errEscInvalidBootMailboxConfiguratio
n = 0x0015,

errEscInvalidMailboxConfiguration =
0x0016,

errEscInvalidSyncManagerConfiguratio
n = 0x0017,
    errEscNoValidInputs = 0x0018,
    errEscNoValidOutputs = 0x0019,
    errEscSynchronization = 0x001A,
    errEscSyncManagerWatchdog =
0x001B,
    errEscInvalidSyncManagerTypes =
0x001C,
    errEscInvalidOutputConfiguration
= 0x001D,
    errEscInvalidInputConfiguration
= 0x001E,

errEscInvalidWatchdogConfiguration =
0x001F,
    errEscSlaveNeedsColdRestart =
0x0020,
    errEscSlaveNeedsInit = 0x0021,

```

RTECAT_ESC_ERROR

```

typedef enum _RTECAT_ESC_ERROR{
    ecatErrEscNoError = 0x0000,
    ecatErrEscUnspecified = 0x0001,
    ecatErrEscNoMemory = 0x0002,
    ecatErrEscInvalidStateChange =
0x0011,
    ecatErrEscUnknownRequestedState =
0x0012,
    ecatErrEscBootstrapUnsupported =
0x0013,
    ecatErrEscNoValidFirmware = 0x0014,

ecatErr
EscInvalidBootMailboxConfiguration =
0x0015,

ecatErrEscInvalidMailboxConfiguration =
0x0016,

ecatErr
EscInvalidSyncManagerConfiguration =
0x0017,
    ecatErrEscNoValidInputs = 0x0018,
    ecatErrEscNoValidOutputs = 0x0019,
    ecatErrEscSynchronization = 0x001A,
    ecatErrEscSyncManagerWatchdog =
0x001B,
    ecatErrEscInvalidSyncManagerTypes =
0x001C,

ecatErrEscInvalidOutputConfiguration =
0x001D,
    ecatErrEscInvalidInputConfiguration
= 0x001E,

ecatErrEscInvalidWatchdogConfiguration
= 0x001F,

```

KINGSTAR 4.x API

```
errEscSlaveNeedsPreOP = 0x0022,  
errEscSlaveNeedsSafeOP = 0x0023,  
errEscInvalidInputMapping =  
0x0024,  
errEscInvalidOutputMapping =  
0x0025,  
errEscInconsistenSettings =  
0x0026,  
errEscFreeRunNotSupported =  
0x0027,  
errEscSynchronizationNotSupported =  
0x0028,  
errEscFreeRunNeeds3BufferMode =  
0x0029,  
errEscBackgroundWatchdog =  
0x002A,  
errEscNoValidInputsAndOutputs =  
0x002B,  
errEscFatalSync = 0x002C,  
errEscNoSync = 0x002D,  
errEscInvalidDcSynchConfiguration =  
0x0030,  
errEscInvalidDcLatchConfiguration =  
0x0031,  
errEscPll = 0x0032,  
errEscInvalidDcIo = 0x0033,  
errEscInvalidDcTimeout = 0x0034,  
errEscInvalidDcSyncCycleTime =  
0x0035,  
errEscDcSync0CycleTime = 0x0036,  
errEscDcSync1CycleTime = 0x0037,  
errEscMailboxAoE = 0x0041,  
errEscMailboxEoE = 0x0042,  
errEscMailboxCoE = 0x0043,  
errEscMailboxFoE = 0x0044,  
errEscMailboxSoE = 0x0045,  
errEscMailboxVoE = 0x004F,
```

wRTOS 1.0 and later

```
ecatErrEscSubDeviceNeedsColdRestart  
= 0x0020,  
ecatErrEscSubDeviceNeedsInit =  
0x0021,  
ecatErrEscSubDeviceNeedsPreOP =  
0x0022,  
ecatErrEscSubDeviceNeedsSafeOP =  
0x0023,  
ecatErrEscInvalidInputMapping =  
0x0024,  
ecatErrEscInvalidOutputMapping =  
0x0025,  
ecatErrEscInconsistentSettings =  
0x0026,  
ecatErrEscFreeRunNotSupported =  
0x0027,  
ecatErrEscSynchronizationNotSupported =  
0x0028,  
ecatErrEscFreeRunNeeds3BufferMode =  
0x0029,  
ecatErrEscBackgroundWatchdog =  
0x002A,  
ecatErrEscNoValidInputsAndOutputs =  
0x002B,  
ecatErrEscFatalSync = 0x002C,  
ecatErrEscNoSync = 0x002D,  
ecatErrEscInvalidDcSynchConfiguration =  
0x0030,  
ecatErrEscInvalidDcLatchConfiguration =  
0x0031,  
ecatErrEscPll = 0x0032,  
ecatErrEscInvalidDcIo = 0x0033,  
ecatErrEscInvalidDcTimeout =  
0x0034,  
ecatErrEscInvalidDcSyncCycleTime =  
0x0035,
```

```

errEscEepromNoAccess = 0x0050,
errEscEeprom = 0x0051,
errEscSlaveRestarted = 0x0060
} EscError;

```

```

ecatErrEscDcSync0CycleTime =
0x0036,
ecatErrEscDcSync1CycleTime =
0x0037,
ecatErrEscMailboxAoE = 0x0041,
ecatErrEscMailboxEoE = 0x0042,
ecatErrEscMailboxCoE = 0x0043,
ecatErrEscMailboxFoE = 0x0044,
ecatErrEscMailboxSoE = 0x0045,
ecatErrEscMailboxVoE = 0x004F,
ecatErrEscEepromNoAccess = 0x0050,
ecatErrEscEeprom = 0x0051,
ecatErrEscSubDeviceRestarted =
0x0060
} RTECAT_ESC_ERROR;

```

- Renamed **EscError** to **RTECAT_ESC_ERROR**.
- Renamed constant from *errEscSlaveNeedsColdRestart* to *ecatErrEscSubDeviceNeedsColdRestart*.
- Renamed constant from *errEscSlaveNeedsInit* to *ecatErrEscSubDeviceNeedsInit*.
- Renamed constant from *errEscSlaveNeedsPreOP* to *ecatErrEscSubDeviceNeedsPreOP*.
- Renamed constant from *errEscSlaveNeedsSafeOP* to *ecatErrEscSubDeviceNeedsSafeOP*.
- Renamed constant from *errEscSlaveRestarted* to *ecatErrEscSubDeviceRestarted*.
- Renamed the prefix of all constants from *err* to *ecatErr*.

EthercatState

```
typedef enum _EthercatState{
    ecatUnknown = -1,
    ecatOffline = 0,
    ecatInit = 1,
    ecatPreOP = 2,
    ecatBoot = 3,
    ecatSafeOP = 4,
    ecatOP = 8
} EthercatState;
```

RTECAT_ETHERCAT_STATE

```
typedef enum _RTECAT_ETHERCAT_STATE{
    ecatStateUnknown = -1,
    ecatStateOffline = 0,
    ecatStateInit = 1,
    ecatStatePreOP = 2,
    ecatStateBoot = 3,
    ecatStateSafeOP = 4,
    ecatStateOP = 8
} RTECAT_ETHERCAT_STATE;
```

- Renamed **EthercatState** to **RTECAT_ETHERCAT_STATE**.
 - Renamed the prefix of all constants from *ecat* to *ecatState*.
-

FoeError

```
typedef enum _FoeError{
    errFoeNotDefined = 0x0000,
    errFoeNotFound = 0x0001,
    errFoeAccessDenied = 0x0002,
    errFoeDiskFull = 0x0003,
    errFoeIllegal = 0x0004,
    errFoePacketNumberWrong =
0x0005,
    errFoeAlreadyExists = 0x0006,
    errFoeNoUser = 0x0007,
    errFoeBootstrapOnly = 0x0008,
    errFoeNotBootstrap = 0x0009,
    errFoeNoRights = 0x000A,
    errFoeProgramError = 0x000B,
    errFoeChecksumWrong = 0x000C,
    errFoeInvalidFirmware = 0x000D,
    errFoeReserved = 0x000E,
    errFoeNoFileToRead = 0x000F,
    errFoeNoFileHeader = 0x0010,
    errFoeFlashProblem = 0x0011,
    errFoeFileIncompatible = 0x0012
} FoeError;
```

RTECAT_FOE_ERROR

```
typedef enum _RTECAT_FOE_ERROR{
    ecatErrFoeNotDefined = 0x0000,
    ecatErrFoeNotFound = 0x0001,
    ecatErrFoeAccessDenied = 0x0002,
    ecatErrFoeDiskFull = 0x0003,
    ecatErrFoeIllegal = 0x0004,
    ecatErrFoePacketNumberWrong =
0x0005,
    ecatErrFoeAlreadyExists = 0x0006,
    ecatErrFoeNoUser = 0x0007,
    ecatErrFoeBootstrapOnly = 0x0008,
    ecatErrFoeNotBootstrap = 0x0009,
    ecatErrFoeNoRights = 0x000A,
    ecatErrFoeProgramError = 0x000B,
    ecatErrFoeChecksumWrong = 0x000C,
    ecatErrFoeInvalidFirmware = 0x000D,
    ecatErrFoeReserved = 0x000E,
    ecatErrFoeNoFileToRead = 0x000F,
    ecatErrFoeNoFileHeader = 0x0010,
    ecatErrFoeFlashProblem = 0x0011,
    ecatErrFoeFileIncompatible = 0x0012
} RTECAT_FOE_ERROR;
```

- Renamed **FoeError** to **RTECAT_FOE_ERROR**.
- Renamed the prefix of all constants from *err* to *ecatErr*.

KsAccessMode

```
typedef enum _KsAccessMode{
    accessPos = 0,
    accessVel,
    accessTor,
    accessPosVel,
    accessVelPos,
    accessPosVelTor,
    accessVelPosTor
} KsAccessMode;
```

RTECAT_ACCESS_MODE

```
typedef enum _RTECAT_ACCESS_MODE{
    ecatAccessPos = 0,
    ecatAccessVel,
    ecatAccessTor,
    ecatAccessPosVel,
    ecatAccessVelPos,
    ecatAccessPosVelTor,
    ecatAccessVelPosTor
} RTECAT_ACCESS_MODE;
```

- Renamed **KsAccessMode** to **RTECAT_ACCESS_MODE**.
- Renamed constant *accessPos* to *ecatAccessPos*.
- Renamed constant *accessVel* to *ecatAccessVel*.
- Renamed constant *accessTor* to *ecatAccessTor*.
- Renamed constant *accessPosVel* to *ecatAccessPosVel*.
- Renamed constant *accessVelPos* to *ecatAccessVelPos*.
- Renamed constant *accessPosVelTor* to *ecatAccessPosVelTor*.
- Renamed constant *accessVelPosTor* to *ecatAccessVelPosTor*.

KsError

```

typedef enum _KsError{
    // Normal Status codes 0x0000
    errNoError = 0,
    errBusy = 1,
    // API Errors 0x1000
    errNullParameter = 0x1000,
    errWrongParameter = 0x1001,
    errSubsystemNotRunning = 0x1002,
    errLinkBusy = 0x1003,
    errNoLicense = 0x1004,
    errWrongEnvironment = 0x1005,
    errVariableUnavailable = 0x1006,
    errVariableSizeIncoherent =
0x1007,
    errUserCancelled = 0x1008,
    errTimeout = 0x1009,
    errFeatureUnavailable = 0x100A,
    errFunctionInUse = 0x100B,
    errSlaveRejected = 0x100C,
    errNotDefined = 0x100D,
    errWrongSubsystemState = 0x100E,
    errWrongSlaveState = 0x100F,
    errLocked = 0x1010,
    // Startup Errors 0x2000
    errNoNicSelected = 0x2000,
    errNicNotFound = 0x2001,
    errLinkDisconnected = 0x2002,
    errNoSlave = 0x2003,
    errUnknownSlave = 0x2004,
    errNoMemory = 0x2005,
    errNoFile = 0x2006,
    errIncorrectFormat = 0x2007,
    errWrongDeviceCount = 0x2008,
    errInitFailed = 0x2009,
    errPreOpFailed = 0x200A,
    errSafeOpFailed = 0x200B,
    errOpFailed = 0x200C,
    errDcFailed = 0x200D,

```

RTECAT_ERROR

```

typedef enum _RTECAT_ERROR{
    // Normal Status codes 0x0000
    ecatErrNoError = 0,
    ecatErrBusy = 1,
    // API Errors 0x1000
    ecatErrNullParameter = 0x1000,
    ecatErrWrongParameter = 0x1001,
    ecatErrMainDeviceNotReady = 0x1002,
    ecatErrLinkBusy = 0x1003,
    ecatErrNoLicense = 0x1004,
    ecatErrWrongEnvironment = 0x1005,
    ecatErrVariableUnavailable =
0x1006,
    ecatErrVariableSizeIncoherent =
0x1007,
    ecatErrUserCancelled = 0x1008,
    ecatErrTimeout = 0x1009,
    ecatErrFeatureUnavailable = 0x100A,
    ecatErrFunctionInUse = 0x100B,
    ecatErrSubDeviceRejected = 0x100C,
    ecatErrNotDefined = 0x100D,
    ecatErrWrongMainDeviceStatus =
0x100E,
    ecatErrWrongSubDeviceState =
0x100F,
    ecatErrLocked = 0x1010,
    // Startup Errors 0x2000
    ecatErrNoNicSelected = 0x2000,
    ecatErrNicNotFound = 0x2001,
    ecatErrLinkDisconnected = 0x2002,
    ecatErrNoSubDevice = 0x2003,
    ecatErrUnknownSubDevice = 0x2004,
    ecatErrNoMemory = 0x2005,
    ecatErrNoFile = 0x2006,
    ecatErrIncorrectFormat = 0x2007,
    ecatErrWrongDeviceCount = 0x2008,
    ecatErrInitFailed = 0x2009,
    ecatErrPreOpFailed = 0x200A,

```

```

errMdpFailed = 0x200E,
errFailed = 0x200F,
errFileVersionNotSupported =
0x2010,
errThreadCreation = 0x2011,
errCabling = 0x2012,
errPacketLost = 0x2013,
errInvalidCycleTime = 0x2014,
errSystemOverload = 0x2015,
errWatchdog = 0x2016,
errUnexpectedData = 0x2017,
errSlaveState = 0x2018,
errConfigurationMismatch =
0x2019,
// Motion Errors 0x3000
errWrongControlMode = 0x3000,
errCommandBufferFull = 0x3001,
errInvalidHandle = 0x3002,
errWrongAxisState = 0x3003,
errNoSensorDefined = 0x3004,
errFollowingError = 0x3005,
errMinimumLimit = 0x3006,
errMaximumLimit = 0x3007,
errCommandJump = 0x3008,
errWrongGroupState = 0x3009,
errUnstableTuning = 0x300A,
errStaleTuning = 0x300B,
// SDO Errors 0x4000
errSdoToggleBit = 0x4000,
errSdoTimeout = 0x4001,
errSdoCommandSpecifier = 0x4002,
errSdoOutOfMemory = 0x4003,
errSdoUnsupportedAccess =
0x4004,
errSdoWriteOnly = 0x4005,
errSdoReadOnly = 0x4006,
errSdoSubindexReadOnly = 0x4007,
errSdoNoCompleteAccess = 0x4008,
errSdoObjectTooLong = 0x4009,
errSdoObjectInPdo = 0x400A,

```

```

ecatErrSafeOpFailed = 0x200B,
ecatErrOpFailed = 0x200C,
ecatErrDcFailed = 0x200D,
ecatErrMdpFailed = 0x200E,
ecatErrFailed = 0x200F,
ecatErrFileVersionNotSupported =
0x2010,
ecatErrThreadCreation = 0x2011,
ecatErrCabling = 0x2012,
ecatErrPacketLost = 0x2013,
ecatErrInvalidCycleTime = 0x2014,
ecatErrSystemOverload = 0x2015,
ecatErrWatchdog = 0x2016,
ecatErrUnexpectedData = 0x2017,
ecatErrSubDeviceState = 0x2018,
ecatErrConfigurationMismatch =
0x2019
} RTECAT_ERROR;

```

- Renamed **KsError** to **RTECAT_ERROR**.
- Renamed constant *errSlaveRejected* to *ecatErrSubDeviceRejected*.
- Renamed constant *errWrongSlaveState* to *ecatErrWrongSubDeviceState*.
- Renamed constant *errNoSlave* to *ecatErrNoSubDevice*.
- Renamed constant *errUnknownSlave* to *ecatErrUnknownSubDevice*.
- Renamed constant *errSlaveState* to *ecatErrSubDeviceState*.
- Renamed the prefix of all constants from *err* to *ecatErr*.
- Removed constants related to Motion and SDO errors.

```
    errSdoObjectNotExist = 0x400B,  
    errSdoNoPdoMapping = 0x400C,  
    errSdoPdoLengthExceeded =  
0x400D,  
    errSdoParameterIncompatible =  
0x400E,  
    errSdoInternalIncompatible =  
0x400F,  
    errSdoHardwareError = 0x4010,  
    errSdoLengthIncorrect = 0x4011,  
    errSdoLengthTooHigh = 0x4012,  
    errSdoLengthTooLow = 0x4013,  
    errSdoSubindexNotExist = 0x4014,  
    errSdoValueOutOfRange = 0x4015,  
    errSdoValueTooHigh = 0x4016,  
    errSdoValueTooLow = 0x4017,  
    errSdoMaxBelowMin = 0x4018,  
    errSdoGeneralError = 0x4019,  
    errSdoCannotTransfer = 0x401A,  
    errSdoCannotTransferLocal =  
0x401B,  
    errSdoWrongState = 0x401C,  
    errSdoDictionaryNotAvailable =  
0x401D  
} KsError;
```

KsFeature

```
typedef enum _KsFeature{
    featureCycleTime = 0,
    featureServerLog,
    featureAutoRestart,
    featureAutoRepair,
    featureAutoConfig,
    featureDc,
    featureDcCheck,
    featureDcMasterShift,
    featureRedundancy,
    featureHotConnect,
    featureAccessMode,
    featureAxisInput,
    featureAxisOutput,
    featureSecondEncoder,
    featureTorqueOffset,
    featureSyncControlMode,
    featureActualVelocity,
    featureActualTorque,
    featureActualCurrent,
    featureFollowingError,
    featureTouchProbe,
    featureMaxTorque,
    featureMaxCurrent,
    featureProfilePosition,
    featureShutdownHandler,
    featureBulkSynchronization,
    featureSyncMultiMaster,
    featureDcShift
} KsFeature;
```

RTECAT_FEATURE

```
typedef enum _RTECAT_FEATURE{
    ecatFeatureCycleTime = 0,
    ecatFeatureAutoRestart,
    ecatFeatureAutoRepair,
    ecatFeatureAutoConfig,
    ecatFeatureDc,
    ecatFeatureDcReferenceIndex,
    ecatFeatureDcCheck,
    ecatFeatureDcMainDeviceShift,
    ecatFeatureRedundancy,
    ecatFeatureHotConnect,
    ecatFeatureAccessMode,
    ecatFeatureAxisDigitalInput,
    ecatFeatureAxisDigitalOutput,
    ecatFeatureSecondEncoder,
    ecatFeatureTorqueOffset,
    ecatFeatureSyncControlMode,
    ecatFeatureActualVelocity,
    ecatFeatureActualTorque,
    ecatFeatureActualCurrent,
    ecatFeatureFollowingError,
    ecatFeatureTouchProbe,
    ecatFeatureMaxTorque,
    ecatFeatureMaxCurrent,
    ecatFeatureProfilePosition,
    ecatFeatureShutdownHandler,
    ecatFeatureBulkSynchronization,
    ecatFeatureSyncMultiMainDevice
} RTECAT_FEATURE;
```

- Renamed **KsFeature** to **RTECAT_FEATURE**.
- Added constant *ecatFeatureDcReferenceIndex*.
- Renamed all constants.
- Removed constants *featureServerLog* and *featureDcShift*.

MailboxCallback

```
typedef int(*MailboxCallback)(
    PVOID Context,
    INT SlaveId,
    MailboxType Type,
    MailboxData Data
);
```

MAILBOX_CALLBACK

```
typedef int(*MAILBOX_CALLBACK)(
    PVOID Context,
    ULONG SubDeviceIndex,
    RTECAT_MAILBOX_TYPE Type,
    RTECAT_MAILBOX_DATA Data
);
```

- Renamed **MailboxCallback** to **MAILBOX_CALLBACK**.
- Renamed parameter *SlaveId* to **SubDeviceIndex** and changed its data type from INT to ULONG.
- Changed the data type of parameter *Type* from **MailboxType** to **RTECAT_MAILBOX_TYPE**.
- Changed the data type of parameter *Data* from **MailboxData** to **RTECAT_MAILBOX_DATA**.

MailboxType

```
typedef enum _MailboxType{
    mbCoeEmergency = 0,
    mbSoeNotification,
    mbSoeSlaveInfo,
    mbVoe
} MailboxType;
```

RTECAT_MAILBOX_TYPE

```
typedef enum _RTECAT_MAILBOX_TYPE{
    ecatMbCoeEmergency = 0,
    ecatMbSoeNotification,
    ecatMbSoeSubDeviceInfo,
    ecatMbVoe
} RTECAT_MAILBOX_TYPE;
```

- Renamed **MailboxType** to **RTECAT_MAILBOX_TYPE**.
- Renamed the prefix of all constants from *mb* to *ecatMb*.

SoeError

```

typedef enum _SoeError{
    // General errors
    errSoeNoError = 0x0000,
    errSoeNotOpened = 0x0001,
    errSoeInvalidAccessClosing =
0x0009,
    // Element 1 (IDN) errors
    errSoeNoIdn = 0x1001,
    errSoeIdnInvalidAccess = 0x1009,
    // Element 2 (Name) errors
    errSoeNoName = 0x2001,
    errSoeNameTooShort = 0x2002,
    errSoeNameTooLong = 0x2003,
    errSoeNameReadOnly = 0x2004,
    errSoeNameWriteProtected =
0x2005,
    // Element 3 (Attribute) errors
    errSoeAttributeTooShort =
0x3002,
    errSoeAttributeTooLong = 0x3003,
    errSoeAttributeReadOnly =
0x3004,
    errSoeAttributeWriteProtected =
0x3005,
    // Element 4 (Unit) errors
    errSoeNoUnits = 0x4001,
    errSoeUnitTooShort = 0x4002,
    errSoeUnitTooLong = 0x4003,
    errSoeUnitReadOnly = 0x4004,
    errSoeUnitWriteProtected =
0x4005,
    // Element 5 (Minimum input)
errors
    errSoeNoMinInput = 0x5001,
    errSoeMinInputTooShort = 0x5002,
    errSoeMinInputTooLong = 0x5003,
    errSoeMinInputReadOnly = 0x5004,

```

RTECAT_SOE_ERROR

```

typedef enum _RTECAT_SOE_ERROR{
    // General errors
    ecatErrSoeNoError = 0x0000,
    ecatErrSoeNotOpened = 0x0001,
    ecatErrSoeInvalidAccessClosing =
0x0009,
    // Element 1 (IDN) errors
    ecatErrSoeNoIdn = 0x1001,
    ecatErrSoeIdnInvalidAccess =
0x1009,
    // Element 2 (Name) errors
    ecatErrSoeNoName = 0x2001,
    ecatErrSoeNameTooShort = 0x2002,
    ecatErrSoeNameTooLong = 0x2003,
    ecatErrSoeNameReadOnly = 0x2004,
    ecatErrSoeNameWriteProtected =
0x2005,
    // Element 3 (Attribute) errors
    ecatErrSoeAttributeTooShort =
0x3002,
    ecatErrSoeAttributeTooLong =
0x3003,
    ecatErrSoeAttributeReadOnly =
0x3004,
    ecatErrSoeAttributeWriteProtected =
0x3005,
    // Element 4 (Unit) errors
    ecatErrSoeNoUnits = 0x4001,
    ecatErrSoeUnitTooShort = 0x4002,
    ecatErrSoeUnitTooLong = 0x4003,
    ecatErrSoeUnitReadOnly = 0x4004,
    ecatErrSoeUnitWriteProtected =
0x4005,
    // Element 5 (Minimum input) errors
    ecatErrSoeNoMinInput = 0x5001,
    ecatErrSoeMinInputTooShort =
0x5002,
    ecatErrSoeMinInputTooLong = 0x5003,

```

KINGSTAR 4.x API

```
errSoeMinInputWriteProtected =
0x5005,
// Element 6 (Maximum input)
errors
errSoeNoMaxInput = 0x6001,
errSoeMaxInputTooShort = 0x6002,
errSoeMaxInputTooLong = 0x6003,
errSoeMaxInputReadOnly = 0x6004,
errSoeMaxInputWriteProtected =
0x6005,
// Element 7 (Operation data)
errors
errSoeDataTooShort = 0x7002,
errSoeDataTooLong = 0x7003,
errSoeDataReadOnly = 0x7004,
errSoeDataWriteProtected =
0x7005,
errSoeDataBellowMin = 0x7006,
errSoeDataOverMax = 0x7007,
errSoeDataInvalid = 0x7008,
errSoeDataPasswordProtected =
0x7009,
errSoeDataInPdo = 0x700A,
errSoeDataInvalidAddress =
0x700B,
errSoeDataWriteDisabled =
0x700C,
errSoeDataInvalidFloat = 0x700D,
errSoeDataParamProtected =
0x700E,
errSoeDataOpProtected = 0x700F,
errSoeCommandAlreadyActive =
0x7010,
errSoeCommandNotInterruptible =
0x7011,
errSoeCommandNotExecutableNow =
0x7012,
errSoeCommandNotExecutable =
0x7013,
```

wRTOS 1.0 and later

```
ecatErrSoeMinInputReadOnly =
0x5004,
ecatErrSoeMinInputWriteProtected =
0x5005,
// Element 6 (Maximum input) errors
ecatErrSoeNoMaxInput = 0x6001,
ecatErrSoeMaxInputTooShort =
0x6002,
ecatErrSoeMaxInputTooLong = 0x6003,
ecatErrSoeMaxInputReadOnly =
0x6004,
ecatErrSoeMaxInputWriteProtected =
0x6005,
// Element 7 (Operation data)
errors
ecatErrSoeDataTooShort = 0x7002,
ecatErrSoeDataTooLong = 0x7003,
ecatErrSoeDataReadOnly = 0x7004,
ecatErrSoeDataWriteProtected =
0x7005,
ecatErrSoeDataBellowMin = 0x7006,
ecatErrSoeDataOverMax = 0x7007,
ecatErrSoeDataInvalid = 0x7008,
ecatErrSoeDataPasswordProtected =
0x7009,
ecatErrSoeDataInPdo = 0x700A,
ecatErrSoeDataInvalidAddress =
0x700B,
ecatErrSoeDataWriteDisabled =
0x700C,
ecatErrSoeDataInvalidFloat =
0x700D,
ecatErrSoeDataParamProtected =
0x700E,
ecatErrSoeDataOpProtected = 0x700F,
ecatErrSoeCommandAlreadyActive =
0x7010,
ecatErrSoeCommandNotInterruptible =
0x7011,
```

```

    errSoeCommandInvalidLength =
0x7014,
    // Segment access errors
    errSoeIdnInvalid = 0x7101,
    errSoeEmptyList = 0x7102,
    errSoeListTooLong = 0x7103,
    errSoeReadTooLong = 0x7104,
    errSoeIdnWriteProtected =
0x7105,
    errSoeSegmentDataBellowMin =
0x7106,
    errSoeSegmentDataAboveMax =
0x7107,
    errSoeListInvalidIndex = 0x7108,
    errSoeParameterNotSegment =
0x7109,
    errSoeIdnNotPermitted = 0x710A,
    // Service channel
    errSoeChannelNoError = 0xD000,
    errSoeChannelNotAvailable =
0xD001,
    errSoeChannelEngaged = 0xD002,
    errSoeChannelBusy = 0xD003,
    errSoeSlaveNotReachable =
0xD004,
    errSoeChannelAborted = 0xD005,
    errSoeChannelReadOnly = 0xD006
} SoeError;

```

```

    ecatErrSoeCommandNotExecutableNow =
0x7012,
    ecatErrSoeCommandNotExecutable =
0x7013,
    ecatErrSoeCommandInvalidLength =
0x7014,
    // Segment access errors
    ecatErrSoeIdnInvalid = 0x7101,
    ecatErrSoeEmptyList = 0x7102,
    ecatErrSoeListTooLong = 0x7103,
    ecatErrSoeReadTooLong = 0x7104,
    ecatErrSoeIdnWriteProtected =
0x7105,
    ecatErrSoeSegmentDataBellowMin =
0x7106,
    ecatErrSoeSegmentDataAboveMax =
0x7107,
    ecatErrSoeListInvalidIndex =
0x7108,
    ecatErrSoeParameterNotSegment =
0x7109,
    ecatErrSoeIdnNotPermitted = 0x710A,
    // Service channel
    ecatErrSoeChannelNoError = 0xD000,
    ecatErrSoeChannelNotAvailable =
0xD001,
    ecatErrSoeChannelEngaged = 0xD002,
    ecatErrSoeChannelBusy = 0xD003,
    ecatErrSoeSubDeviceNotReachable =
0xD004,
    ecatErrSoeChannelAborted = 0xD005,
    ecatErrSoeChannelReadOnly = 0xD006
} RTECAT_SOE_ERROR;

```

- Renamed **SoeError** to **RTECAT_SOE_ERROR**.
- Renamed constant *errSoeSlaveNotReachable* to *ecatErrSoeSubDeviceNotReachable*.
- Renamed the prefix of all constants from *err* to *ecatErr*.

Breaking Changes to Managed E-CAT APIs

Managed Axis Control APIs

KINGSTAR 4.x API

ForceAxisInputBit

```
public KsError ForceAxisInputBit(  
    int Index,  
    int BitOffset,  
    bool Value  
)
```

wRTOS 1.0 and later

ForceAxisInputBit

```
public EcatErrorCode ForceAxisInputBit  
(  
    EcatHandle axis,  
    uint bitOffset,  
    bool value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *BitOffset* to *bitOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

ForceAxisInputByte

```
public KsError ForceAxisInputByte(
    int Index,
    int ByteOffset,
    byte Value
)
```

ForceAxisInputByte

```
public EcatErrorCode
ForceAxisInputByte(
    EcatHandle axis,
    uint byteOffset,
    byte value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

ForceAxisInputDWord

```
public KsError ForceAxisInputDWord(
    int Index,
    int ByteOffset,
    uint Value
)
```

ForceAxisInputDWord

```
public EcatErrorCode
ForceAxisInputDWord(
    EcatHandle axis,
    uint byteOffset,
    uint value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

ForceAxisInputWord

```
public KsError ForceAxisInputWord(
    int Index,
    int ByteOffset,
    ushort Value
)
```

ForceAxisInputWord

```
public EcatErrorCode
ForceAxisInputWord(
    EcatHandle axis,
    uint byteOffset,
    ushort value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

ReadAxisInputBit

```
public KsReturn<bool>
ReadAxisInputBit(
    int Index,
    int BitOffset
)
```

ReadAxisInputBit

```
public EcatErrorCode ReadAxisInputBit(
    EcatHandle axis,
    uint bitOffset,
    ref bool value
)
```

- Changed the return type from **KsReturn<bool>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Renamed parameter *BitOffset* to *bitOffset* and changed its data type from int to uint.
- Removed parameter *Index*.

ReadAxisInputByte

```
public KsReturn<byte>
ReadAxisInputByte(
    int Index,
    int ByteOffset
)
```

ReadAxisInputByte

```
public EcatErrorCode ReadAxisInputByte
(
    EcatHandle axis,
    uint byteOffset,
    ref byte value
)
```

- Changed the return type from **KsReturn<byte>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Removed parameter *Index*.

ReadAxisInputWord

```
public KsReturn<ushort>
ReadAxisInputWord(
    int Index,
    int ByteOffset
)
```

ReadAxisInputWord

```
public EcatErrorCode ReadAxisInputWord
(
    EcatHandle axis,
    uint byteOffset,
    ref ushort value
)
```

- Changed the return type from **KsReturn<ushort>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Removed parameter *Index*.

ReadAxisInputDWord

```
public KsReturn<uint>
ReadAxisInputDWord(
    int Index,
    int ByteOffset
)
```

ReadAxisInputDWord

```
public EcatErrorCode
ReadAxisInputDWord(
    EcatHandle axis,
    uint byteOffset,
    ref uint value
)
```

- Changed the return type from **KsReturn<uint>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Removed parameter *Index*.

ReadAxisOutputBit

```
public KsReturn<bool>
ReadAxisOutputBit(
    int Index,
    int BitOffset
)
```

ReadAxisOutputBit

```
public EcatErrorCode ReadAxisOutputBit
(
    EcatHandle axis,
    uint bitOffset,
    ref bool value
)
```

- Changed the return type from **KsReturn<bool>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Renamed parameter *BitOffset* to *bitOffset* and changed its data type from int to uint.
- Removed parameter *Index*.

ReadAxisOutputByte

```
public KsReturn<byte>
ReadAxisOutputByte(
    int Index,
    int ByteOffset
)
```

ReadAxisOutputByte

```
public EcatErrorCode
ReadAxisOutputByte(
    EcatHandle axis,
    uint byteOffset,
    ref byte value
)
```

- Changed the return type from **KsReturn<byte>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Removed parameter *Index*.

ReadAxisOutputWord

```
public KsReturn<ushort>
ReadAxisOutputWord(
    int Index,
    int ByteOffset
)
```

ReadAxisOutputWord

```
public EcatErrorCode
ReadAxisOutputWord(
    EcatHandle axis,
    uint byteOffset,
    ref ushort value
)
```

- Changed the return type from **KsReturn<ushort>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Removed parameter *Index*.

ReadAxisOutputDWord

```
public KsReturn<uint>
ReadAxisOutputDWord(
    int Index,
    int ByteOffset
)
```

ReadAxisOutputDWord

```
public EcatErrorCode
ReadAxisOutputDWord(
    EcatHandle axis,
    uint byteOffset,
    ref uint value
)
```

- Changed the return type from **KsReturn<uint>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Removed parameter *Index*.

WriteAxisOutputBit

```
public KsError WriteAxisOutputBit(
    int Index,
    int BitOffset,
    bool Value
)
```

WriteAxisOutputBit

```
public EcatErrorCode
WriteAxisOutputBit(
    EcatHandle axis,
    uint bitOffset,
    bool value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *BitOffset* to *bitOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

WriteAxisOutputByte

```
public KsError WriteAxisOutputByte(  
    int Index,  
    int ByteOffset,  
    byte Value  
)
```

WriteAxisOutputByte

```
public EcatErrorCode  
WriteAxisOutputByte(  
    EcatHandle axis,  
    uint byteOffset,  
    byte value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

WriteAxisOutputDWord

```
public KsError WriteAxisOutputDWord(  
    int Index,  
    int ByteOffset,  
    uint Value  
)
```

WriteAxisOutputDWord

```
public EcatErrorCode  
WriteAxisOutputDWord(  
    EcatHandle axis,  
    uint byteOffset,  
    uint value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisOutputWord

```
public KsError WriteAxisOutputWord(
    int Index,
    int ByteOffset,
    ushort Value
)
```

WriteAxisOutputWord

```
public EcatErrorCode
WriteAxisOutputWord(
    EcatHandle axis,
    uint byteOffset,
    ushort value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

ReadAxisInputs

```
public KsReturn<uint> ReadAxisInputs(
    int Index
)
```

ReadAxisDigitalInputs

```
public EcatErrorCode
ReadAxisDigitalInputs(
    EcatHandle axis,
    ref uint value
)
```

- Renamed **ReadAxisInputs** to **ReadAxisDigitalInputs**.
- Changed the return type from **KsReturn<uint>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Removed parameter *Index*.

ReadAxisOutputs

```
public KsReturn<uint> ReadAxisOutputs
(
    int Index
)
```

ReadAxisDigitalOutputs

```
public EcatErrorCode
ReadAxisDigitalOutputs(
    EcatHandle axis,
    ref uint value
)
```

- Renamed **ReadAxisOutputs** to **ReadAxisDigitalOutputs**.
- Changed the return type from **KsReturn<uint>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Removed parameter *Index*.

ReadAxisStatusWord

```
public KsReturn<ushort>
ReadAxisStatusWord(
    int Index
)
```

ReadAxisStatusWord

```
public EcatErrorCode
ReadAxisStatusWord(
    EcatHandle axis,
    ref ushort value
)
```

- Changed the return type from **KsReturn<ushort>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Removed parameter *Index*.

ReadAxisActualPosition

```
public KsReturn<int>
ReadAxisActualPosition(
    int Index
)
```

ReadAxisActualPosition

```
public EcatErrorCode
ReadAxisActualPosition(
    EcatHandle axis,
    ref int value
)
```

- Changed the return type from **KsReturn<int>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Removed parameter *Index*.

ReadAxisActualTorque

```
public KsReturn<short>
ReadAxisActualTorque(
    int Index
)
```

ReadAxisActualTorque

```
public EcatErrorCode
ReadAxisActualTorque(
    EcatHandle axis,
    ref short value
)
```

- Changed the return type from **KsReturn<short>** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Added parameter *value*.
 - Removed parameter *Index*.
-

ReadAxisActualVelocity

```
public KsReturn<int>
ReadAxisActualVelocity(
    int Index
)
```

ReadAxisActualVelocity

```
public EcatErrorCode
ReadAxisActualVelocity(
    EcatHandle axis,
    ref int value
)
```

- Changed the return type from **KsReturn<int>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Removed parameter *Index*.

ReadAxisActualCurrent

```
public KsReturn<short>
ReadAxisActualCurrent(
    int Index
)
```

ReadAxisActualCurrent

```
public EcatErrorCode
ReadAxisActualCurrent(
    EcatHandle axis,
    ref short value
)
```

- Changed the return type from **KsReturn<short>** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Added parameter *value*.
 - Removed parameter *Index*.
-

ReadAxisSecondEncoder

```
public KsReturn<int>
ReadAxisSecondEncoder(
    int Index
)
```

ReadAxisSecondEncoder

```
public EcatErrorCode
ReadAxisSecondEncoder(
    EcatHandle axis,
    ref int value
)
```

- Changed the return type from **KsReturn<int>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Removed parameter *Index*.

ReadAxisFollowingError

```
public KsReturn<int>
ReadAxisFollowingError(
    int Index
)
```

ReadAxisFollowingError

```
public EcatErrorCode
ReadAxisFollowingError(
    EcatHandle axis,
    ref int value
)
```

- Changed the return type from **KsReturn<int>** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Added parameter *value*.
 - Removed parameter *Index*.
-

ReadAxisTouchProbeStatus

```
public KsReturn<ushort>
ReadAxisTouchProbeStatus(
    int Index
)
```

ReadAxisTouchProbeStatus

```
public EcatErrorCode
ReadAxisTouchProbeStatus(
    EcatHandle axis,
    ref ushort value
)
```

- Changed the return type from **KsReturn<ushort>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Removed parameter *Index*.

ReadAxisMopDisplay

```
public KsReturn<byte>
ReadAxisMopDisplay(
    int Index
)
```

ReadAxisMopDisplay

```
public EcatErrorCode
ReadAxisMopDisplay(
    EcatHandle axis,
    ref byte value
)
```

- Changed the return type from **KsReturn<byte>** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Added parameter *value*.
 - Removed parameter *Index*.
-

ReadAxisCustomInput1

```
public KsReturn<uint>
ReadAxisCustomInput1(
    int Index
)
```

ReadAxisCustomInput1

```
public EcatErrorCode
ReadAxisCustomInput1(
    EcatHandle axis,
    ref uint value
)
```

- Changed the return type from **KsReturn<uint>** to **EcatErrorCode**.
- Added parameter *axis*.
- Added parameter *value*.
- Removed parameter *Index*.

ReadAxisCustomInput2

```
public KsReturn<uint>
ReadAxisCustomInput2(
    int Index
)
```

ReadAxisCustomInput2

```
public EcatErrorCode
ReadAxisCustomInput2(
    EcatHandle axis,
    ref uint value
)
```

- Changed the return type from **KsReturn<uint>** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Added parameter *value*.
 - Removed parameter *Index*.
-

ForceAxisInputs

```
public KsError ForceAxisInputs(  
    int Index,  
    uint Value  
)
```

ForceAxisDigitalInputs

```
public EcatErrorCode  
ForceAxisDigitalInputs(  
    EcatHandle axis,  
    uint value  
)
```

- Renamed **ForceAxisInputs** to **ForceAxisDigitalInputs**.
- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

WriteAxisOutputs

```
public KsError WriteAxisOutputs(  
    int Index,  
    uint Value  
)
```

WriteAxisDigitalOutputs

```
public EcatErrorCode  
WriteAxisDigitalOutputs(  
    EcatHandle axis,  
    uint value  
)
```

- Renamed **WriteAxisOutputs** to **WriteAxisDigitalOutputs**.
 - Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisControlWord

```
public KsError WriteAxisControlWord(  
    int Index,  
    ushort Value  
)
```

WriteAxisControlWord

```
public EcatErrorCode  
WriteAxisControlWord(  
    EcatHandle axis,  
    ushort value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

WriteAxisCustomOutput1

```
public KsError WriteAxisCustomOutput1  
(  
    int Index,  
    uint Value  
)
```

WriteAxisCustomOutput1

```
public EcatErrorCode  
WriteAxisCustomOutput1(  
    EcatHandle axis,  
    uint value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisCustomOutput2

```
public KsError WriteAxisCustomOutput2
(
    int Index,
    uint Value
)
```

WriteAxisCustomOutput2

```
public EcatErrorCode
WriteAxisCustomOutput2(
    EcatHandle axis,
    uint value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

WriteAxisMaxCurrent

```
public KsError WriteAxisMaxCurrent(
    int Index,
    short Value
)
```

WriteAxisMaxCurrent

```
public EcatErrorCode
WriteAxisMaxCurrent(
    EcatHandle axis,
    short value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisMaxTorque

```
public KsError WriteAxisMaxTorque(  
    int Index,  
    short Value  
)
```

WriteAxisMaxTorque

```
public EcatErrorCode  
WriteAxisMaxTorque(  
    EcatHandle axis,  
    short value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisMop

```
public KsError WriteAxisMop(  
    int Index,  
    byte Value  
)
```

WriteAxisMop

```
public EcatErrorCode WriteAxisMop(  
    EcatHandle axis,  
    byte value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisProfileAcceleration

```
public KsError
WriteAxisProfileAcceleration(
    int Index,
    int Value
)
```

WriteAxisProfileAcceleration

```
public EcatErrorCode
WriteAxisProfileAcceleration(
    EcatHandle axis,
    int value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisProfileDeceleration

```
public KsError
WriteAxisProfileDeceleration(
    int Index,
    int Value
)
```

WriteAxisProfileDeceleration

```
public EcatErrorCode
WriteAxisProfileDeceleration(
    EcatHandle axis,
    int value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisProfileVelocity

```
public KsError
WriteAxisProfileVelocity(
    int Index,
    int Value
)
```

WriteAxisProfileVelocity

```
public EcatErrorCode
WriteAxisProfileVelocity(
    EcatHandle axis,
    int value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

WriteAxisTargetPosition

```
public KsError
WriteAxisTargetPosition(
    int Index,
    int Value
)
```

WriteAxisTargetPosition

```
public EcatErrorCode
WriteAxisTargetPosition(
    EcatHandle axis,
    int value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisTargetTorque

```
public KsError WriteAxisTargetTorque(  
    int Index,  
    short Value  
)
```

WriteAxisTargetTorque

```
public EcatErrorCode  
WriteAxisTargetTorque(  
    EcatHandle axis,  
    short value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisTargetVelocity

```
public KsError  
WriteAxisTargetVelocity(  
    int Index,  
    int Value  
)
```

WriteAxisTargetVelocity

```
public EcatErrorCode  
WriteAxisTargetVelocity(  
    EcatHandle axis,  
    int value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *axis*.
 - Renamed parameter *Value* to *value*.
 - Removed parameter *Index*.
-

WriteAxisTorqueOffset

```
public KsError WriteAxisTorqueOffset(
    int Index,
    short Value
)
```

WriteAxisTorqueOffset

```
public EcatErrorCode
WriteAxisTorqueOffset(
    EcatHandle axis,
    short value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

WriteAxisTouchProbeControl

```
public KsError
WriteAxisTouchProbeControl(
    int Index,
    ushort Value
)
```

WriteAxisTouchProbeControl

```
public EcatErrorCode
WriteAxisTouchProbeControl(
    EcatHandle axis,
    ushort value
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *axis*.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

Managed Command APIs

AbortCommand

```
public KsCommandStatus AbortCommand(
    KsCommandStatus Status
)
```

AbortCommand

```
public CommandStatus AbortCommand(
    CommandStatus status
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus** to **CommandStatus**.

AbortCommand

```
public KsCommandStatus<byte[]>
AbortCommand(
    KsCommandStatus<byte[]> Status
)
```

AbortCommand

```
public CommandStatus<byte[]>
AbortCommand(
    CommandStatus<byte[]> status
)
```

- Changed the return type from **KsCommandStatus<byte[]>** to **CommandStatus<byte[]>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<byte[]>** to **CommandStatus<byte[]>**.

AbortCommand

```
public KsCommandStatus<uint>
AbortCommand(
    KsCommandStatus<uint> Status
)
```

AbortCommand

```
public CommandStatus<uint>
AbortCommand(
    CommandStatus<uint> status
)
```

- Changed the return type from **KsCommandStatus<uint>** to **CommandStatus<uint>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<uint>** to **CommandStatus<uint>**.

AbortCommand

```
public KsCommandStatus<ushort>
AbortCommand(
    KsCommandStatus<ushort> Status
)
```

AbortCommand

```
public CommandStatus<ushort>
AbortCommand(
    CommandStatus<ushort> status
)
```

- Changed the return type from **KsCommandStatus<ushort>** to **CommandStatus<ushort>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<ushort>** to **CommandStatus<ushort>**.

AbortCommand

```
public KsCommandStatus<ushort[]>
AbortCommand(
    KsCommandStatus<ushort[]> Status
)
```

AbortCommand

```
public CommandStatus<ushort[]>
AbortCommand(
    CommandStatus<ushort[]> status
)
```

- Changed the return type from **KsCommandStatus<ushort[]>** to **CommandStatus<ushort[]>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<ushort[]>** to **CommandStatus<ushort[]>**.

AbortCommand

```
public
KsCommandStatus<SdoEntryDescription>
AbortCommand(

KsCommandStatus<SdoEntryDescription>
Status
)
```

AbortCommand

```
public
CommandStatus<SdoEntryDescription>
AbortCommand(

CommandStatus<SdoEntryDescription>
status
)
```

- Changed the return type from **KsCommandStatus<SdoEntryDescription>** to **CommandStatus<SdoEntryDescription>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<SdoEntryDescription>** to **CommandStatus<SdoEntryDescription>**.

AbortCommand

```
public
KsCommandStatus<SdoObjectDescription>
AbortCommand (

KsCommandStatus<SdoObjectDescription>
Status
)
```

GetCommandsStatus

```
public KsReturn<KsCommandStatus<byte
[]>[]> GetCommandsStatus (
    KsCommandStatus<byte[]>[] Statuses
)
```

AbortCommand

```
public
CommandStatus<SdoObjectDescription>
AbortCommand (

CommandStatus<SdoObjectDescription>
status
)
```

- Changed the return type from **KsCommandStatus<SdoObjectDescription>** to **CommandStatus<SdoObjectDescription>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<SdoObjectDescription>** to **CommandStatus<SdoObjectDescription>**.

GetCommandsStatus

```
public EcatErrorCode
GetCommandsStatus (
    ref CommandStatus<byte[]>[]
statuses
)
```

- Changed the return type from **KsReturn<KsCommandStatus<byte[]>[]>** to **EcatErrorCode**.
- Renamed parameter *Statuses* to *statuses* and changed its data type from **KsCommandStatus<byte[]>[]** to **ref CommandStatus<byte[]>[]**.

GetCommandStatus

```
public KsCommandStatus
GetCommandStatus(
    KsCommandStatus Status
)
```

GetCommandStatus

```
public CommandStatus GetCommandStatus
(
    CommandStatus status
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus** to **CommandStatus**.

GetCommandStatus

```
public KsCommandStatus<byte[]>
GetCommandStatus(
    KsCommandStatus<byte[]> Status
)
```

GetCommandStatus

```
public CommandStatus<byte[]>
GetCommandStatus(
    CommandStatus<byte[]> status
)
```

- Changed the return type from **KsCommandStatus<byte[]>** to **CommandStatus<byte[]>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<byte[]>** to **CommandStatus<byte[]>**.

GetCommandStatus

```
public KsCommandStatus<uint>
GetCommandStatus (
    KsCommandStatus<uint> Status
)
```

GetCommandStatus

```
public CommandStatus<uint>
GetCommandStatus (
    CommandStatus<uint> status
)
```

- Changed the return type from **KsCommandStatus<uint>** to **CommandStatus<uint>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<uint>** to **CommandStatus<uint>**.

GetCommandStatus

```
public KsCommandStatus<ushort>
GetCommandStatus (
    KsCommandStatus<ushort> Status
)
```

GetCommandStatus

```
public CommandStatus<ushort>
GetCommandStatus (
    CommandStatus<ushort> status
)
```

- Changed the return type from **KsCommandStatus<ushort>** to **CommandStatus<ushort>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<ushort>** to **CommandStatus<ushort>**.

GetCommandStatus

```
public KsCommandStatus<ushort[]>
GetCommandStatus (
    KsCommandStatus<ushort[]> Status
)
```

GetCommandStatus

```
public CommandStatus<ushort[]>
GetCommandStatus (
    CommandStatus<ushort[]> status
)
```

- Changed the return type from **KsCommandStatus<ushort[]>** to **CommandStatus<ushort[]>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<ushort[]>** to **CommandStatus<ushort[]>**.

GetCommandStatus

```
public
KsCommandStatus<SdoEntryDescription>
GetCommandStatus (

KsCommandStatus<SdoEntryDescription>
Status
)
```

GetCommandStatus

```
public
CommandStatus<SdoEntryDescription>
GetCommandStatus (

CommandStatus<SdoEntryDescription>
status
)
```

- Changed the return type from **KsCommandStatus<SdoEntryDescription>** to **CommandStatus<SdoEntryDescription>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<SdoEntryDescription>** to **CommandStatus<SdoEntryDescription>**.

GetCommandStatus

```
public
KsCommandStatus<SdoObjectDescription>
GetCommandStatus (

KsCommandStatus<SdoObjectDescription>
Status
)
```

WaitForCommand

```
public KsCommandStatus WaitForCommand(
    double Timeout,
    bool AbortOnTimeout,
    KsCommandStatus Status
)
```

GetCommandStatus

```
public
CommandStatus<SdoObjectDescription>
GetCommandStatus (

CommandStatus<SdoObjectDescription>
status
)
```

- Changed the return type from **KsCommandStatus<SdoObjectDescription>** to **CommandStatus<SdoObjectDescription>**.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<SdoObjectDescription>** to **CommandStatus<SdoObjectDescription>**.

WaitForCommand

```
public CommandStatus WaitForCommand(
    double timeout,
    bool abortOnTimeout,
    CommandStatus status
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Renamed parameter *Timeout* to *timeout*.
- Renamed parameter *AbortOnTimeout* to *abortOnTimeout*.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus** to **CommandStatus**.

WaitForCommand

```
public KsCommandStatus<byte[]>
WaitForCommand(
    double Timeout,
    bool AbortOnTimeout,
    KsCommandStatus<byte[]> Status
)
```

WaitForCommand

```
public CommandStatus<byte[]>
WaitForCommand(
    double timeout,
    bool abortOnTimeout,
    CommandStatus<byte[]> status
)
```

- Changed the return type from **KsCommandStatus<byte[]>** to **CommandStatus<byte[]>**.
- Renamed parameter *Timeout* to *timeout*.
- Renamed parameter *AbortOnTimeout* to *abortOnTimeout*.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<byte[]>** to **CommandStatus<byte[]>**.

WaitForCommand

```
public KsCommandStatus<uint>
WaitForCommand(
    double Timeout,
    bool AbortOnTimeout,
    KsCommandStatus<uint> Status
)
```

WaitForCommand

```
public CommandStatus<uint>
WaitForCommand(
    double timeout,
    bool abortOnTimeout,
    CommandStatus<uint> status
)
```

- Changed the return type from **KsCommandStatus<uint>** to **CommandStatus<uint>**.
- Renamed parameter *Timeout* to *timeout*.
- Renamed parameter *AbortOnTimeout* to *abortOnTimeout*.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<uint>** to **CommandStatus<uint>**.

WaitForCommand

```
public KsCommandStatus<ushort>
WaitForCommand(
    double Timeout,
    bool AbortOnTimeout,
    KsCommandStatus<ushort> Status
)
```

WaitForCommand

```
public CommandStatus<ushort>
WaitForCommand(
    double timeout,
    bool abortOnTimeout,
    CommandStatus<ushort> status
)
```

- Changed the return type from **KsCommandStatus<ushort>** to **CommandStatus<ushort>**.
- Renamed parameter *Timeout* to *timeout*.
- Renamed parameter *AbortOnTimeout* to *abortOnTimeout*.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<ushort>** to **CommandStatus<ushort>**.

WaitForCommand

```
public KsCommandStatus<ushort[]>
WaitForCommand(
    double Timeout,
    bool AbortOnTimeout,
    KsCommandStatus<ushort[]> Status
)
```

WaitForCommand

```
public CommandStatus<ushort[]>
WaitForCommand(
    double timeout,
    bool abortOnTimeout,
    CommandStatus<ushort[]> status
)
```

- Changed the return type from **KsCommandStatus<ushort[]>** to **CommandStatus<ushort[]>**.
- Renamed parameter *Timeout* to *timeout*.
- Renamed parameter *AbortOnTimeout* to *abortOnTimeout*.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<ushort[]>** to **CommandStatus<ushort[]>**.

WaitForCommand

```
public
KsCommandStatus<SdoEntryDescription>
WaitForCommand(
    double Timeout,
    bool AbortOnTimeout,

KsCommandStatus<SdoEntryDescription>
Status
)
```

WaitForCommand

```
public
CommandStatus<SdoEntryDescription>
WaitForCommand(
    double timeout,
    bool abortOnTimeout,

CommandStatus<SdoEntryDescription>
status
)
```

- Changed the return type from **KsCommandStatus<SdoEntryDescription>** to **CommandStatus<SdoEntryDescription>**.
 - Renamed parameter *Timeout* to *timeout*.
 - Renamed parameter *AbortOnTimeout* to *abortOnTimeout*.
 - Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<SdoEntryDescription>** to **CommandStatus<SdoEntryDescription>**.
-

WaitForCommand

```
public
KsCommandStatus<SdoObjectDescription>
WaitForCommand(
    double Timeout,
    bool AbortOnTimeout,

KsCommandStatus<SdoObjectDescription>
Status
)
```

WaitForCommand

```
public
CommandStatus<SdoObjectDescription>
WaitForCommand(
    double timeout,
    bool abortOnTimeout,

CommandStatus<SdoObjectDescription>
status
)
```

- Changed the return type from **KsCommandStatus<SdoObjectDescription>** to **CommandStatus<SdoObjectDescription>**.
- Renamed parameter *Timeout* to *timeout*.
- Renamed parameter *AbortOnTimeout* to *abortOnTimeout*.
- Renamed parameter *Status* to *status* and changed its data type from **KsCommandStatus<SdoObjectDescription>** to **CommandStatus<SdoObjectDescription>**.

Managed Mailbox APIs

AoeReadCommand

```
public KsCommandStatus<byte[]>
AoeReadCommand (
    int SlaveId,
    int Port,
    int IndexGroup,
    int IndexOffset,
    int Length
)
```

AoeReadCommand

```
public CommandStatus<byte[]>
AoeReadCommand (
    EcatHandle subDevice,
    ushort port,
    uint indexGroup,
    uint indexOffset,
    uint length
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Port* to *port* and changed its data type from int to ushort.
- Renamed parameter *IndexGroup* to *indexGroup* and changed its data type from int to uint.
- Renamed parameter *IndexOffset* to *indexOffset* and changed its data type from int to uint.
- Renamed parameter *Length* to *length* and changed its data type from int to uint.
- Removed parameter *SlaveId*.

AoeReadSdoObject

```
public KsCommandStatus<byte[]>
AoeReadSdoObject(
    int SlaveId,
    int Port,
    int ObIndex,
    int ObSubIndex,
    bool CompleteAccess,
    int Length
)
```

AoeReadSdoObject

```
public CommandStatus<byte[]>
AoeReadSdoObject(
    EcatHandle subDevice,
    ushort port,
    ushort obIndex,
    byte obSubIndex,
    bool completeAccess,
    uint length
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Port* to *port* and changed its data type from int to ushort.
- Renamed parameter *ObIndex* to *obIndex* and changed its data type from int to ushort.
- Renamed parameter *ObSubIndex* to *obSubIndex* and changed its data type from int to byte.
- Renamed parameter *CompleteAccess* to *completeAccess*.
- Renamed parameter *Length* to *length* and changed its data type from int to uint.
- Removed parameter *SlaveId*.

AoeReadWriteCommand

```
public KsCommandStatus<byte[]>
AoeReadWriteCommand(
    int SlaveId,
    int Port,
    int IndexGroup,
    int IndexOffset,
    int ReadLength,
    byte[] WriteValue
)
```

AoeReadWriteCommand

```
public CommandStatus<byte[]>
AoeReadWriteCommand(
    EcatHandle subDevice,
    ushort port,
    uint indexGroup,
    uint indexOffset,
    uint readLength,
    byte[] writeValue
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Port* to *port* and changed its data type from int to ushort.
- Renamed parameter *IndexGroup* to *indexGroup* and changed its data type from int to uint.
- Renamed parameter *IndexOffset* to *indexOffset* and changed its data type from int to uint.
- Renamed parameter *ReadLength* to *readLength* and changed its data type from int to uint.
- Renamed parameter *WriteValue* to *writeValue*.
- Removed parameter *SlaveId*.

AoeWriteCommand

```
public KsCommandStatus AoeWriteCommand(
    int SlaveId,
    int Port,
    int IndexGroup,
    int IndexOffset,
    byte[] Value
)
```

AoeWriteCommand

```
public CommandStatus AoeWriteCommand(
    EcatHandle subDevice,
    ushort port,
    uint indexGroup,
    uint indexOffset,
    byte[] value
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Port* to *port* and changed its data type from int to ushort.
- Renamed parameter *IndexGroup* to *indexGroup* and changed its data type from int to uint.
- Renamed parameter *IndexOffset* to *indexOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *SlaveId*.

AoeWriteControlCommand

```
public KsCommandStatus
AoeWriteControlCommand(
    int SlaveId,
    int Port,
    ushort AoeState,
    ushort DeviceState,
    byte[] Value
)
```

AoeWriteControlCommand

```
public CommandStatus
AoeWriteControlCommand(
    EcatHandle subDevice,
    ushort port,
    ushort aoeState,
    ushort deviceState,
    byte[] value
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Port* to *port* and changed its data type from int to ushort.
- Renamed parameter *AoeState* to *aoeState*.
- Renamed parameter *DeviceState* to *deviceState*.
- Renamed parameter *Value* to *value*.
- Removed parameter *SlaveId*.

AoeWriteSdoObject

```
public KsCommandStatus
AoeWriteSdoObject(
    int SlaveId,
    int Port,
    int ObIndex,
    int ObSubIndex,
    bool CompleteAccess,
    byte[] Value
)
```

AoeWriteSdoObject

```
public CommandStatus
AoeWriteSdoObject(
    EcatHandle subDevice,
    ushort port,
    ushort obIndex,
    byte obSubIndex,
    bool completeAccess,
    byte[] value
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Port* to *port* and changed its data type from int to ushort.
- Renamed parameter *ObIndex* to *obIndex* and changed its data type from int to ushort.
- Renamed parameter *ObSubIndex* to *obSubIndex* and changed its data type from int to byte.
- Renamed parameter *CompleteAccess* to *completeAccess*.
- Renamed parameter *Value* to *value*.
- Removed parameter *SlaveId*.

CoeReadSdoEntryDescription

```
public
KsCommandStatus<SdoEntryDescription>
CoeReadSdoEntryDescription(
    int SlaveId,
    int ObIndex,
    int ObSubIndex
)
```

CoeReadSdoEntryDescription

```
public
CommandStatus<SdoEntryDescription>
CoeReadSdoEntryDescription(
    EcatHandle subDevice,
    ushort obIndex,
    byte obSubIndex
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *ObIndex* to *obIndex* and changed its data type from int to ushort.
- Renamed parameter *ObSubIndex* to *obSubIndex* and changed its data type from int to byte.
- Removed parameter *SlaveId*.

CoeReadSdoObject

```
public KsCommandStatus<byte[]>
CoeReadSdoObject(
    int SlaveId,
    int ObIndex,
    int ObSubIndex,
    bool CompleteAccess,
    int Length
)
```

CoeReadSdoObject

```
public CommandStatus<byte[]>
CoeReadSdoObject(
    EcatHandle subDevice,
    ushort obIndex,
    byte obSubIndex,
    bool completeAccess,
    uint length
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *ObIndex* to *obIndex* and changed its data type from int to ushort.
- Renamed parameter *ObSubIndex* to *obSubIndex* and changed its data type from int to byte.
- Renamed parameter *CompleteAccess* to *completeAccess*.
- Renamed parameter *Length* to *length* and changed its data type from int to uint.
- Removed parameter *SlaveId*.

CoeReadSdoObjectDescription

```
public
KsCommandStatus<SdoObjectDescription>
CoeReadSdoObjectDescription(
    int SlaveId,
    int ObIndex
)
```

CoeReadSdoObjectDescription

```
public
CommandStatus<SdoObjectDescription>
CoeReadSdoObjectDescription(
    EcatHandle subDevice,
    ushort obIndex
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *ObIndex* to *obIndex* and changed its data type from `int` to `ushort`.
- Removed parameter *SlaveId*.

CoeReadSdoODList

```
public KsCommandStatus<ushort[]>
CoeReadSdoODList(
    int SlaveId,
    int ListType
)
```

CoeReadSdoODList

```
public CommandStatus<ushort[]>
CoeReadSdoODList(
    EcatHandle subDevice,
    ushort listType,
    uint length
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *ListType* to *listType* and changed its data type from `int` to `ushort`.
- Added parameter *length*.
- Removed parameter *SlaveId*.

CoeWriteSdoObject

```
public KsCommandStatus  
CoeWriteSdoObject(  
    int SlaveId,  
    int ObIndex,  
    int ObSubIndex,  
    bool CompleteAccess,  
    byte[] Value  
)
```

CoeWriteSdoObject

```
public CommandStatus  
CoeWriteSdoObject(  
    EcatHandle subDevice,  
    ushort obIndex,  
    byte obSubIndex,  
    bool completeAccess,  
    byte[] value  
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *ObIndex* to *obIndex* and changed its data type from int to ushort.
- Renamed parameter *ObSubIndex* to *obSubIndex* and changed its data type from int to byte.
- Renamed parameter *CompleteAccess* to *completeAccess*.
- Renamed parameter *Value* to *value*.
- Removed parameter *SlaveId*.

FoeReadFileToFile

```
public KsCommandStatus
FoeReadFileToFile(
    int SlaveId,
    bool BootState,
    string Name,
    uint Password,
    string File
)
```

FoeReadFileToFile

```
public CommandStatus
FoeReadFileToFile(
    EcatHandle subDevice,
    bool bootState,
    string name,
    uint password,
    string file
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *BootState* to *bootState*.
- Renamed parameter *Name* to *name*.
- Renamed parameter *Password* to *password*.
- Renamed parameter *File* to *file*.
- Removed parameter *SlaveId*.

FoeWriteFileFromFile

```
public KsCommandStatus
FoeWriteFileFromFile(
    int SlaveId,
    bool BootState,
    string Name,
    uint Password,
    string File
)
```

FoeWriteFileFromFile

```
public CommandStatus
FoeWriteFileFromFile(
    EcatHandle subDevice,
    bool bootState,
    string name,
    uint password,
    string file
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *BootState* to *bootState*.
- Renamed parameter *Name* to *name*.
- Renamed parameter *Password* to *password*.
- Renamed parameter *File* to *file*.
- Removed parameter *SlaveId*.

ReadAxisSdoObject

```
public KsCommandStatus<byte[]>
ReadAxisSdoObject (
    int Index,
    int ObIndex,
    int ObSubIndex,
    bool CompleteAccess,
    int Length
)
```

ReadAxisSdoObject

```
public CommandStatus<byte[]>
ReadAxisSdoObject (
    EcatHandle axis,
    ushort obIndex,
    byte obSubIndex,
    bool completeAccess,
    uint length,
    bool priority
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *axis*.
- Renamed parameter *ObIndex* to *obIndex* and changed its data type from int to ushort.
- Renamed parameter *ObSubIndex* to *obSubIndex* and changed its data type from int to byte.
- Renamed parameter *CompleteAccess* to *completeAccess*.
- Renamed parameter *Length* to *length* and changed its data type from int to uint.
- Added parameter *priority*.
- Removed parameter *Index*.

ReadIOSdoObject

```
public KsCommandStatus<byte[]>
ReadIOSdoObject(
    int Index,
    int ObIndex,
    int ObSubIndex,
    bool CompleteAccess,
    int Length
)
```

ReadModuleSdoObject

```
public CommandStatus<byte[]>
ReadModuleSdoObject(
    EcatHandle module,
    ushort obIndex,
    byte obSubIndex,
    bool completeAccess,
    uint length,
    bool priority
)
```

- Renamed **ReadIOSdoObject** to **ReadModuleSdoObject**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *module*.
- Renamed parameter *ObIndex* to *obIndex* and changed its data type from int to ushort.
- Renamed parameter *ObSubIndex* to *obSubIndex* and changed its data type from int to byte.
- Renamed parameter *CompleteAccess* to *completeAccess*.
- Renamed parameter *Length* to *length* and changed its data type from int to uint.
- Added parameter *priority*.
- Removed parameter *Index*.

SoeReadSsc

```
public KsCommandStatus<byte[]>
SoeReadSsc (
    int SlaveId,
    byte DriveNo,
    byte ElementFlags,
    ushort Idn,
    int Length
)
```

SoeReadSsc

```
public CommandStatus<byte[]>
SoeReadSsc (
    EcatHandle subDevice,
    byte driveNo,
    byte elementFlags,
    ushort idn,
    uint length
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *DriveNo* to *driveNo*.
- Renamed parameter *ElementFlags* to *elementFlags*.
- Renamed parameter *Idn* to *idn*.
- Renamed parameter *Length* to *length* and changed its data type from int to uint.
- Removed parameter *SlaveId*.

SoeWriteSsc

```
public KsCommandStatus SoeWriteSsc(  
    int SlaveId,  
    byte DriveNo,  
    byte ElementFlags,  
    ushort Idn,  
    byte[] Value  
)
```

SoeWriteSsc

```
public CommandStatus SoeWriteSsc(  
    EcatHandle subDevice,  
    byte driveNo,  
    byte elementFlags,  
    ushort idn,  
    byte[] value  
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *DriveNo* to *driveNo*.
- Renamed parameter *ElementFlags* to *elementFlags*.
- Renamed parameter *Idn* to *idn*.
- Renamed parameter *Value* to *value*.
- Removed parameter *SlaveId*.

VoeWrite

```
public KsCommandStatus VoeWrite(  
    int SlaveId,  
    uint VendorId,  
    ushort VendorType,  
    byte[] Data  
)
```

VoeWrite

```
public CommandStatus VoeWrite(  
    EcatHandle subDevice,  
    uint vendorId,  
    ushort vendorType,  
    byte[] data  
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *VendorId* to *vendorId*.
- Renamed parameter *VendorType* to *vendorType*.
- Renamed parameter *Data* to *data*.
- Removed parameter *SlaveId*.

WriteAxisSdoObject

```
public KsCommandStatus
WriteAxisSdoObject(
    int Index,
    int ObIndex,
    int ObSubIndex,
    bool CompleteAccess,
    byte[] Value
)
```

WriteAxisSdoObject

```
public CommandStatus
WriteAxisSdoObject(
    EcatHandle axis,
    ushort obIndex,
    byte obSubIndex,
    bool completeAccess,
    byte[] value,
    bool priority
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *axis*.
- Renamed parameter *ObIndex* to *obIndex* and changed its data type from int to ushort.
- Renamed parameter *ObSubIndex* to *obSubIndex* and changed its data type from int to byte.
- Renamed parameter *CompleteAccess* to *completeAccess*.
- Renamed parameter *Value* to *value*.
- Added parameter *priority*.
- Removed parameter *Index*.

WriteIOSdoObject

```
public KsCommandStatus
WriteIOSdoObject(
    int Index,
    int ObIndex,
    int ObSubIndex,
    bool CompleteAccess,
    byte[] Value
)
```

WriteModuleSdoObject

```
public CommandStatus
WriteModuleSdoObject(
    EcatHandle module,
    ushort obIndex,
    byte obSubIndex,
    bool completeAccess,
    byte[] value,
    bool priority
)
```

- Renamed **WriteIOSdoObject** to **WriteModuleSdoObject**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *module*.
- Renamed parameter *ObIndex* to *obIndex* and changed its data type from int to ushort.
- Renamed parameter *ObSubIndex* to *obSubIndex* and changed its data type from int to byte.
- Renamed parameter *CompleteAccess* to *completeAccess*.
- Renamed parameter *Value* to *value*.
- Added parameter *priority*.
- Removed parameter *Index*.

Managed MainDevice APIs

Create

```
public KsError Create(  
    int Instance,  
    int IdealProcessor  
)
```

CreateMainDevice

```
public EcatErrorCode CreateMainDevice  
(  
    ref EcatHandle mainDevice,  
    uint instance  
)
```

- Renamed **Create** to **CreateMainDevice**.
- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Renamed parameter *Instance* to *instance* and changed its data type from `int` to `uint`.
- Removed parameter *IdealProcessor*.

Destroy

```
public KsError Destroy()
```

DestroyMainDevice

```
public EcatErrorCode  
DestroyMainDevice(  
    ref EcatHandle mainDevice  
)
```

- Renamed **Destroy** to **DestroyMainDevice**.
 - Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *mainDevice*.
-

GetDCSystemTime

```
public KsReturn<ulong> GetDCSystemTime
(
    DcCycle Cycle
)
```

GetDCSystemTime

```
public EcatErrorCode GetDCSystemTime (
    EcatHandle mainDevice,
    DcCycle cycle,
    ref ulong time
)
```

- Changed the return type from **KsError<ulong>** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Renamed parameter *Cycle* to *cycle*.
- Added parameter *time*.

GetStatus

```
public KsError GetStatus(
    SubsystemStatus Status,
    SubsystemDiagnostics Diagnostics
)
```

GetStatus (Status, Diagnostics)

```
public EcatErrorCode GetStatus(
    EcatHandle mainDevice,
    ref MainDeviceStatus status,
    ref MainDeviceDiagnostics
    diagnostics
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Renamed parameter *Status* to *status* and changed its data type from **SubsystemStatus** to **ref MainDeviceStatus**.
- Renamed parameter *Diagnostics* to *diagnostics* and changed its data type from **SubsystemDiagnostics** to **ref MainDeviceDiagnostics**.

RequestState

```
public KsCommandStatus RequestState(  
    EthercatState State  
)
```

RequestState

```
public CommandStatus RequestState(  
    EcatHandle mainDevice,  
    EcatState state  
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *mainDevice*.
- Renamed parameter *State* to *state* and changed its data type from **EthercatState** to **EcatState**.

Restart

```
public KsCommandStatus Restart()
```

Restart

```
public CommandStatus Restart(  
    EcatHandle mainDevice  
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *mainDevice*.

Start

```
public KsCommandStatus Start()
```

Start

```
public CommandStatus Start(  
    EcatHandle mainDevice  
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
 - Added parameter *mainDevice*.
-

StartFromConfiguration

```
public KsCommandStatus  
StartFromConfiguration(  
    string Configuration  
)
```

StartFromConfiguration

```
public CommandStatus  
StartFromConfiguration(  
    EcatHandle mainDevice,  
    string configuration  
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
 - Added parameter *mainDevice*.
 - Renamed parameter *Configuration* to *configuration*.
-

Stop

```
public KsCommandStatus Stop()
```

Stop

```
public CommandStatus Stop(  
    EcatHandle mainDevice  
)
```

- Changed the return type from **KsCommandStatus** to **CommandStatus**.
 - Added parameter *mainDevice*.
-

SubsystemIsRunning

```
public KsReturn<bool>
SubsystemIsRunning()
```

IsMainDeviceRunning

```
public EcatErrorCode
IsMainDeviceRunning(
    EcatHandle mainDevice,
    ref bool created
)
```

- Renamed **SubsystemIsRunning** to **IsMainDeviceRunning**.
- Changed the return type from **KsReturn<bool>** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Added parameter *created*.

GetFeatureStatus

```
public KsReturn<double>
GetFeatureStatus(
    KsFeature Feature
)
```

GetFeatureStatus

```
public EcatErrorCode GetFeatureStatus
(
    EcatHandle mainDevice,
    EcatFeature feature,
    ref double status
)
```

- Changed the return type from **KsReturn<double>** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Renamed parameter *Feature* to *feature* and changed its data type from **KsFeature** to **EcatFeature**.
- Added parameter *status*.

PulseHeartbeat

```
public KsError PulseHeartbeat()
```

PulseHeartbeat

```
public EcatErrorCode PulseHeartbeat(  
    EcatHandle mainDevice  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *mainDevice*.

SetFeatureStatus

```
public KsError SetFeatureStatus(  
    KsFeature Feature,  
    double Value  
)
```

SetFeatureStatus

```
public EcatErrorCode SetFeatureStatus  
(  
    EcatHandle mainDevice,  
    EcatFeature feature,  
    double status  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Renamed parameter *Feature* to *feature* and changed its data type from **KsFeature** to **EcatFeature**.
- Added parameter *status*.
- Removed parameter *Value*.

KINGSTAR 4.x API

wRTOS 1.0 and later

SetHeartbeat

```
public KsError SetHeartbeat(  
    bool Enable,  
    double Timeout  
)
```

SetHeartbeat

```
public EcatErrorCode SetHeartbeat(  
    EcatHandle mainDevice,  
    bool enable,  
    double timeout  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Renamed parameter *Enable* to *enable*.
- Renamed parameter *Timeout* to *timeout*.

Managed Module Control APIs

ForceInputBit

```
public KsError ForceInputBit(  
    int Index,  
    int BitOffset,  
    bool Value  
)
```

ForceModuleInputBit

```
public EcatErrorCode ForceModuleInputBit  
(  
    EcatHandle module,  
    uint bitOffset,  
    bool value  
)
```

- Renamed **ForceInputBit** to **ForceModuleInputBit**.
- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *BitOffset* to *bitOffset* and changed its data type from `int` to `uint`.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

ForceInputByte

```
public KsError ForceInputByte(  
    int Index,  
    int ByteOffset,  
    byte Value  
)
```

ForceModuleInputByte

```
public EcatErrorCode  
ForceModuleInputByte(  
    EcatHandle module,  
    uint byteOffset,  
    byte value  
)
```

- Renamed **ForceInputByte** to **ForceModuleInputByte**.
- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

ForceInputDWord

```
public KsError ForceInputDWord(  
    int Index,  
    int ByteOffset,  
    uint Value  
)
```

ForceModuleInputDWord

```
public EcatErrorCode  
ForceModuleInputDWord(  
    EcatHandle module,  
    uint byteOffset,  
    uint value  
)
```

- Renamed **ForceInputDWord** to **ForceModuleInputDWord**.
- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

ForceInputWord

```
public KsError ForceInputWord(  
    int Index,  
    int ByteOffset,  
    ushort Value  
)
```

ForceModuleInputWord

```
public EcatErrorCode  
ForceModuleInputWord(  
    EcatHandle module,  
    uint byteOffset,  
    ushort value  
)
```

- Renamed **ForceInputWord** to **ForceModuleInputWord**.
- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

ReadInputBit

```
public KsReturn<bool> ReadInputBit(  
    int Index,  
    int BitOffset  
)
```

ReadModuleInputBit

```
public EcatErrorCode ReadModuleInputBit(  
    EcatHandle module,  
    uint bitOffset,  
    ref bool value  
)
```

- Renamed **ReadInputBit** to **ReadModuleInputBit**.
- Changed the return type from **KsReturn<bool>** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *BitOffset* to *bitOffset* and changed its data type from **int** to **uint**.
- Added parameter *value*.
- Removed parameter *Index*.

ReadInputByte

```
public KsReturn<byte> ReadInputByte
(
    int Index,
    int ByteOffset
)
```

ReadModuleInputByte

```
public EcatErrorCode ReadModuleInputByte
(
    EcatHandle module,
    uint byteOffset,
    ref byte value
)
```

- Renamed **ReadInputByte** to **ReadModuleInputByte**.
- Changed the return type from **KsReturn<byte>** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from **int** to **uint**.
- Added parameter *value*.
- Removed parameter *Index*.

ReadInputDWord

```
public KsReturn<uint>
ReadInputDWord(
    int Index,
    int ByteOffset
)
```

ReadModuleInputDWord

```
public EcatErrorCode
ReadModuleInputDWord(
    EcatHandle module,
    uint byteOffset,
    ref byte value
)
```

- Renamed **ReadInputDWord** to **ReadModuleInputDWord**.
- Changed the return type from **KsReturn<uint>** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from **int** to **uint**.
- Added parameter *value*.
- Removed parameter *Index*.

ReadInputWord

```
public KsReturn<ushort>
ReadInputWord(
    int Index,
    int ByteOffset
)
```

ReadModuleInputWord

```
public EcatErrorCode ReadModuleInputWord
(
    EcatHandle module,
    uint byteOffset,
    ref ushort value
)
```

- Renamed **ReadInputWord** to **ReadModuleInputWord**.
 - Changed the return type from **KsReturn<ushort>** to **EcatErrorCode**.
 - Added parameter *module*.
 - Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from **int** to **uint**.
 - Added parameter *value*.
 - Removed parameter *Index*.
-

ReadOutputBit

```
public KsReturn<bool> ReadOutputBit
(
    int Index,
    int BitOffset
)
```

ReadModuleOutputBit

```
public EcatErrorCode ReadModuleOutputBit
(
    EcatHandle module,
    uint bitOffset,
    ref bool value
)
```

- Renamed **ReadOutputBit** to **ReadModuleOutputBit**.
- Changed the return type from **KsReturn<bool>** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *BitOffset* to *bitOffset* and changed its data type from **int** to **uint**.
- Added parameter *value*.
- Removed parameter *Index*.

ReadOutputByte

```
public KsReturn<byte>
ReadOutputByte(
    int Index,
    int ByteOffset
)
```

ReadModuleOutputByte

```
public EcatErrorCode
ReadModuleOutputByte(
    EcatHandle module,
    uint byteOffset,
    ref byte value
)
```

- Renamed **ReadOutputByte** to **ReadModuleOutputByte**.
- Changed the return type from **KsReturn<byte>** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from **int** to **uint**.
- Added parameter *value*.
- Removed parameter *Index*.

ReadOutputDWord

```
public KsReturn<uint>
ReadOutputDWord(
    int Index,
    int ByteOffset
)
```

ReadModuleOutputDWord

```
public EcatErrorCode
ReadModuleOutputDWord(
    EcatHandle module,
    uint byteOffset,
    ref byte value
)
```

- Renamed **ReadOutputDWord** to **ReadModuleOutputDWord**.
- Changed the return type from **KsReturn<uint>** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from **int** to **uint**.
- Added parameter *value*.
- Removed parameter *Index*.

ReadOutputWord

```
public KsReturn<ushort>
ReadOutputWord(
    int Index,
    int ByteOffset
)
```

ReadModuleOutputWord

```
public EcatErrorCode
ReadModuleOutputWord(
    EcatHandle module,
    uint byteOffset,
    ref ushort value
)
```

- Renamed **ReadOutputWord** to **ReadModuleOutputWord**.
- Changed the return type from **KsReturn<ushort>** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from **int** to **uint**.
- Added parameter *value*.
- Removed parameter *Index*.

WriteOutputBit

```
public KsError WriteOutputBit(  
    int Index,  
    int BitOffset,  
    bool Value  
)
```

WriteModuleOutputBit

```
public EcatErrorCode  
WriteModuleOutputBit(  
    EcatHandle module,  
    uint bitOffset,  
    bool value  
)
```

- Renamed **WriteOutputBit** to **WriteModuleOutputBit**.
- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *BitOffset* to *bitOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

WriteOutputByte

```
public KsError WriteOutputByte(  
    int Index,  
    int ByteOffset,  
    byte Value  
)
```

WriteModuleOutputByte

```
public EcatErrorCode  
WriteModuleOutputByte(  
    EcatHandle module,  
    uint byteOffset,  
    byte value  
)
```

- Renamed **WriteOutputByte** to **WriteModuleOutputByte**.
- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

WriteOutputDWord

```
public KsError WriteOutputDWord(  
    int Index,  
    int ByteOffset,  
    uint Value  
)
```

WriteModuleOutputDWord

```
public EcatErrorCode  
WriteModuleOutputDWord(  
    EcatHandle module,  
    uint byteOffset,  
    uint value  
)
```

- Renamed **WriteOutputDWord** to **WriteModuleOutputDWord**.
- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from **int** to **uint**.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

WriteOutputWord

```
public KsError WriteOutputWord(  
    int Index,  
    int ByteOffset,  
    ushort Value  
)
```

WriteModuleOutputWord

```
public EcatErrorCode  
WriteModuleOutputWord(  
    EcatHandle module,  
    uint byteOffset,  
    ushort value  
)
```

- Renamed **ForcelInputWord** to **WriteModuleOutputWord**.
- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *module*.
- Renamed parameter *ByteOffset* to *byteOffset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.
- Removed parameter *Index*.

Managed Shared Memory APIs

ReadMemoryBit

```
public KsReturn<bool> ReadMemoryBit(
    int Offset,
    int BitOffset
)
```

ReadMemoryByte

```
public KsReturn<byte> ReadMemoryByte(
    int Offset
)
```

ReadMemoryBit

```
public EcatErrorCode ReadMemoryBit(
    uint offset,
    uint bitOffset,
    ref bool value
)
```

- Changed the return type from **KsReturn<bool>** to **EcatErrorCode**.
- Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
- Renamed parameter *BitOffset* to *bitOffset* and changed its data type from int to uint.
- Added parameter *value*.

ReadMemoryByte

```
public EcatErrorCode ReadMemoryByte(
    uint offset,
    ref byte value
)
```

- Changed the return type from **KsReturn<byte>** to **EcatErrorCode**.
- Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
- Added parameter *value*.

ReadMemoryDWord

```
public KsReturn<uint> ReadMemoryDWord(  
    int Offset  
)
```

ReadMemoryDWord

```
public EcatErrorCode ReadMemoryDWord(  
    uint offset,  
    ref uint value  
)
```

- Changed the return type from **KsReturn<uint>** to **EcatErrorCode**.
- Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
- Added parameter *value*.

ReadMemoryLWord

```
public KsReturn<ulong> ReadMemoryLWord(  
    int Offset  
)
```

ReadMemoryLWord

```
public EcatErrorCode ReadMemoryLWord(  
    uint offset,  
    ref ulong value  
)
```

- Changed the return type from **KsReturn<ulong>** to **EcatErrorCode**.
 - Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
 - Added parameter *value*.
-

ReadMemoryWord

```
public KsReturn<ushort> ReadMemoryWord  
(  
    int Offset  
)
```

ReadMemoryWord

```
public EcatErrorCode ReadMemoryWord(  
    uint offset,  
    ref ushort value  
)
```

- Changed the return type from **KsReturn<ushort>** to **EcatErrorCode**.
- Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
- Added parameter *value*.

WriteMemoryBit

```
public KsError WriteMemoryBit(  
    int Offset,  
    int BitOffset,  
    bool Value  
)
```

WriteMemoryBit

```
public EcatErrorCode WriteMemoryBit(  
    uint offset,  
    uint bitOffset,  
    bool value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
 - Renamed parameter *BitOffset* to *bitOffset* and changed its data type from int to uint.
 - Renamed parameter *Value* to *value*.
-

WriteMemoryByte

```
public KsError WriteMemoryByte(  
    int Offset,  
    byte Value  
)
```

WriteMemoryByte

```
public EcatErrorCode WriteMemoryByte(  
    uint offset,  
    byte value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.

WriteMemoryDWord

```
public KsError WriteMemoryDWord(  
    int Offset,  
    uint Value  
)
```

WriteMemoryDWord

```
public EcatErrorCode WriteMemoryDWord  
(  
    uint offset,  
    uint value  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
 - Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
 - Renamed parameter *Value* to *value*.
-

KINGSTAR 4.x API

WriteMemoryLWord

```
public KsError WriteMemoryLWord(  
    int Offset,  
    ulong Value  
)
```

wRTOS 1.0 and later

WriteMemoryLWord

```
public EcatErrorCode WriteMemoryLWord(  
    (   
        uint offset,  
        ulong value  
    )
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.

WriteMemoryWord

```
public KsError WriteMemoryWord(  
    int Offset,  
    ushort Value  
)
```

WriteMemoryWord

```
public EcatErrorCode WriteMemoryWord(  
    (   
        uint offset,  
        ushort value  
    )
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
- Renamed parameter *Value* to *value*.

Managed Simulated Hardware APIs

ConfigureAxis

```
public KsError ConfigureAxis(
    int Index,
    AxisDetails Details
)
```

ConfigureIo

```
public KsError ConfigureIo(
    int Index,
    SlaveStatus Details
)
```

ConfigureAxis

```
public EcatErrorCode ConfigureAxis(
    EcatHandle mainDevice,
    uint index,
    SubDeviceStatus details,
    uint resolution
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Renamed parameter *Index* to *index* and changed its data type from `int` to `uint`.
- Renamed parameter *Details* to *details* and changed its data type from **AxisDetails** to **SubDeviceStatus**.
- Added parameter *resolution*.

ConfigureIo

```
public EcatErrorCode ConfigureIo(
    EcatHandle mainDevice,
    uint index,
    SubDeviceStatus details
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Renamed parameter *Index* to *index* and changed its data type from `int` to `uint`.
- Renamed parameter *Details* to *details* and changed its data type from **SlaveStatus** to **SubDeviceStatus**.

GetConfiguredAxisCount

```
public KsReturn<int>  
GetConfiguredAxisCount()
```

GetConfiguredAxisCount

```
public EcatErrorCode  
GetConfiguredAxisCount(  
    EcatHandle mainDevice,  
    ref uint count  
)
```

- Changed the return type from **KsReturn<int>** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Added parameter *count*.

GetConfiguredIoCount

```
public KsReturn<int>  
GetConfiguredIoCount()
```

GetConfiguredIoCount

```
public EcatErrorCode  
GetConfiguredIoCount(  
    EcatHandle mainDevice,  
    ref uint count  
)
```

- Changed the return type from **KsReturn<int>** to **EcatErrorCode**.
 - Added parameter *mainDevice*.
 - Added parameter *count*.
-

SetConfiguredAxesCount

```
public KsError SetConfiguredAxesCount(  
    int Count  
)
```

SetConfiguredAxesCount

```
public EcatErrorCode  
SetConfiguredAxesCount(  
    EcatHandle mainDevice,  
    uint count  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Renamed parameter *Count* to *count* and changed its data type from int to uint.

SetConfiguredIoCount

```
public KsError SetConfiguredIoCount(  
    int Count  
)
```

SetConfiguredIoCount

```
public EcatErrorCode  
SetConfiguredIoCount(  
    EcatHandle mainDevice,  
    uint count  
)
```

- Changed the return type from **KsError** to **EcatErrorCode**.
- Added parameter *mainDevice*.
- Renamed parameter *Count* to *count* and changed its data type from int to uint.

Managed SubDevice APIs

OverrideSlavePdoConfiguration

```
public KsError  
OverrideSlavePdoConfiguration(  
    int SlaveId,  
    int InputLength,  
    int OutputLength  
)
```

OverrideSubDevicePdoConfiguration

```
public EcatErrorCode  
OverrideSubDevicePdoConfiguration(  
    EcatHandle subDevice,  
    uint inputLength,  
    uint outputLength  
)
```

- Renamed **OverrideSlavePdoConfiguration** to **OverrideSubDevicePdoConfiguration**.
 - Changed the return type from **KsError** to **EcatErrorCode**.
 - Added parameter *subDevice*.
 - Renamed parameter *InputLength* to *inputLength* and changed its data type from int to uint.
 - Renamed parameter *OutputLength* to *outputLength* and changed its data type from int to uint.
 - Removed parameter *SlaveId*.
-

ReadAllSlaveEeprom

```
public KsCommandStatus
ReadAllSlaveEeprom(
    int SlaveId,
    string File
)
```

ReadAllSubDeviceEeprom

```
public CommandStatus
ReadAllSubDeviceEeprom(
    EcatHandle subDevice,
    string file
)
```

- Renamed **ReadAllSlaveEeprom** to **ReadAllSubDeviceEeprom**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *File* to *file*.
- Removed parameter *SlaveId*.

ReadSlaveAlias

```
public KsCommandStatus<ushort>
ReadSlaveAlias(
    int SlaveId
)
```

ReadSubDeviceAlias

```
public CommandStatus<ushort>
ReadSubDeviceAlias(
    EcatHandle subDevice
)
```

- Renamed **ReadSlaveAlias** to **ReadSubDeviceAlias**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Removed parameter *SlaveId*.

ReadSlaveEeprom

```
public KsCommandStatus<uint>
ReadSlaveEeprom(
    int SlaveId,
    int Offset
)
```

ReadSubDeviceEeprom

```
public CommandStatus<uint>
ReadSubDeviceEeprom(
    EcatHandle subDevice,
    uint offset
)
```

- Renamed **ReadSlaveEeprom** to **ReadSubDeviceEeprom**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
- Removed parameter *SlaveId*.

ReadSlaveRegister

```
public KsCommandStatus<byte[]>
ReadSlaveRegister(
    int SlaveId,
    int Offset,
    int Length
)
```

ReadSubDeviceRegister

```
public CommandStatus<byte[]>
ReadSubDeviceRegister(
    EcatHandle subDevice,
    ushort offset,
    uint length
)
```

- Renamed **ReadSlaveRegister** to **ReadSubDeviceRegister**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Offset* to *offset* and changed its data type from int to ushort.
- Renamed parameter *Length* to *length* and changed its data type from int to uint.
- Removed parameter *SlaveId*.

ReloadSlaveEeprom

```
public KsCommandStatus
ReloadSlaveEeprom(
    int SlaveId
)
```

ReloadSubDeviceEeprom

```
public CommandStatus
ReloadSubDeviceEeprom(
    EcatHandle subDevice
)
```

- Renamed **ReloadSlaveEeprom** to **ReloadSubDeviceEeprom**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Removed parameter *SlaveId*.

RequestSlaveState

```
public KsCommandStatus  
RequestSlaveState(  
    int SlaveId,  
    EthercatState State  
)
```

RequestSubDeviceState

```
public CommandStatus  
RequestSubDeviceState(  
    EcatHandle subDevice,  
    EcatState state  
)
```

- Renamed **RequestSlaveState** to **RequestSubDeviceState**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *State* to *state* and changed its data type from **EthercatState** to **EcatState**.
- Removed parameter *SlaveId*.

ResetSlaveEsc

```
public KsCommandStatus ResetSlaveEsc  
(  
    int SlaveId  
)
```

ResetSubDeviceEsc

```
public CommandStatus ResetSubDeviceEsc(  
    EcatHandle subDevice  
)
```

- Renamed **ResetSlaveEsc** to **ResetSubDeviceEsc**.
 - Changed the return type from **KsCommandStatus** to **CommandStatus**.
 - Added parameter *subDevice*.
 - Removed parameter *SlaveId*.
-

RestartSlave

```
public KsCommandStatus RestartSlave(  
    int SlaveId  
)
```

RestartSubDevice

```
public CommandStatus RestartSubDevice(  
    EcatHandle subDevice  
)
```

- Renamed **RestartSlave** to **RestartSubDevice**.
 - Changed the return type from **KsCommandStatus** to **CommandStatus**.
 - Added parameter *subDevice*.
 - Removed parameter *SlaveId*.
-

WriteAllSlaveEEProm

```
public KsCommandStatus  
WriteAllSlaveEEProm(  
    int SlaveId,  
    string File  
)
```

WriteAllSubDeviceEEProm

```
public CommandStatus  
WriteAllSubDeviceEEProm(  
    EcatHandle subDevice,  
    string file  
)
```

- Renamed **WriteAllSlaveEEProm** to **WriteAllSubDeviceEEProm**.
 - Changed the return type from **KsCommandStatus** to **CommandStatus**.
 - Added parameter *subDevice*.
 - Renamed parameter *File* to *file*.
 - Removed parameter *SlaveId*.
-

WriteSlaveAlias

```
public KsCommandStatus
WriteSlaveAlias(
    int SlaveId,
    ushort Alias
)
```

WriteSubDeviceAlias

```
public CommandStatus WriteSubDeviceAlias
(
    EcatHandle subDevice,
    ushort alias
)
```

- Renamed **WriteSlaveAlias** to **WriteSubDeviceAlias**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Alias* to *alias*.
- Removed parameter *SlaveId*.

WriteSlaveEeprom

```
public KsCommandStatus
WriteSlaveEeprom(
    int SlaveId,
    int Offset,
    uint Data
)
```

WriteSubDeviceEeprom

```
public CommandStatus
WriteSubDeviceEeprom(
    EcatHandle subDevice,
    uint offset,
    uint data
)
```

- Renamed **WriteSlaveEeprom** to **WriteSubDeviceEeprom**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Offset* to *offset* and changed its data type from int to uint.
- Renamed parameter *Data* to *data*.
- Removed parameter *SlaveId*.

WriteSlaveRegister

```
public KsCommandStatus
WriteSlaveRegister(
    int SlaveId,
    int Offset,
    byte[] Data
)
```

WriteSubDeviceRegister

```
public CommandStatus
WriteSubDeviceRegister(
    EcatHandle subDevice,
    ushort offset,
    byte[] data
)
```

- Renamed **WriteSlaveRegister** to **WriteSubDeviceRegister**.
- Changed the return type from **KsCommandStatus** to **CommandStatus**.
- Added parameter *subDevice*.
- Renamed parameter *Offset* to *offset* and changed its data type from int to ushort.
- Renamed parameter *Data* to *data*.
- Removed parameter *SlaveId*.

Other Managed E-CAT APIs

KsCommandStatus Class

CommandStatus Class

- Renamed **KsCommandStatus** to **CommandStatus**.
- Added properties *Buffer*, *BufferLength*, *BufferType*, *DataLength*, *ReturnCode*, and *State*.
- Made property *ErrorId* writable and changed its data type from **KsError** to **EcatErrorCode**.
- Made property *Handle* writable and changed its data type from **ulong** to **EcatHandle**.
- Removed properties *Active*, *Busy*, *CommandAborted*, *Done*, *Error*, *InSync*, *InSyncLatch*, *InVelocity*, and *ValueLength*.

KsCommandStatus<T> Class

CommandStatus<T> Class

- Renamed **KsCommandStatus<T>** to **CommandStatus<T>**.

SubsystemDiagnostics Class

MainDeviceDiagnostics Class

- Renamed **SubsystemDiagnostics** to **MainDeviceDiagnostics**.
- Made all properties writable.
- Added properties *EcatHandle* and *SubDeviceCount*.
- Renamed property *Slaves* to *SubDevices* and changed its data type from **SlaveDiagnostics[]** to **SubDeviceDiagnostics[]**.

SubsystemStatus Class**MainDeviceStatus Class**

- Renamed **SubsystemStatus** to **MainDeviceStatus**.
- Converted all class members from fields to properties.
- Added properties *AxisConfigurationStamps*, *MainDeviceConfigurationStamp*, *ModuleConfigurationStamps*, and *SubDeviceConfigurationStamps*.
- Renamed *IOCount* to *ModuleCount*.
- Renamed *IOStates* to *ModuleStates*.
- Renamed *SlaveCount* to *SubDeviceCount*.
- Renamed *SlaveStates* to *SubDeviceStates*.
- Changed the data type of *AxesCount*, *ModuleCount*, and *SubDeviceCount* from int to uint.
- Changed the data type of *AxesStates* from **AxisState**[] to **EcatState**[].
- Changed the data type of *ModuleStates* and *SubDeviceStates* from **EthercatState** [] to **EcatState**[].
- Changed the data type of *RequestedState* and *State* from **EthercatState** to **EcatState**.

SlaveDiagnostics Class**SubDeviceDiagnostics Class**

- Renamed **SlaveDiagnostics** to **SubDeviceDiagnostics**.
 - Made all properties writable.
-

KINGSTAR 4.x API

wRTOS 1.0 and later

SlaveStatus Class

SubDeviceStatus Class

- Renamed **SlaveStatus** to **SubDeviceStatus**.
- Converted all class members from fields to properties.

Removed APIs

This section lists KINGSTAR 4.x APIs that were removed from wRTOS 1.x SDKs.

Removed E-CAT APIs

KINGSTAR 4.x API

wRTOS 1.0 and later

GetAxisByIndex

Removed. Use **RtecatOpenAxisByIndex** or **RtecatOpenAxisByLocation** to open an Axis and then use **RtecatGetAxisStatus** to get its status.

ReadAxisAlias

Removed.

WriteAxisAlias

Removed.

GetSlaveById

Removed. Use **RtecatOpenSubDeviceByExplicitId**, **RtecatOpenSubDeviceByIndex**, or **RtecatOpenSubDeviceByStationAddress** to open a SubDevice and then use **RtecatGetSubDeviceStatus** to get its status.

KINGSTAR 4.x API**wRTOS 1.0 and later**

GetIOByIndex	Removed. Use RtecatOpenModuleByIndex or RtecatOpenModuleByLocation to open a module and then use RtecatGetModuleStatus to get its status.
ReadIOAlias	Removed.
WriteIOAlias	Removed.
ConfigureDc	Removed. Use RtecatSetFeatureStatus .
EnableAliases	Removed.
EnableAutoConfig	Removed. Use RtecatSetFeatureStatus .
EnableAutoRepair	Removed. Use RtecatSetFeatureStatus .
EnableAutoRestart	Removed. Use RtecatSetFeatureStatus .
EnableHotConnect	Removed. Use RtecatSetFeatureStatus .
EnableRedundancy	Removed. Use RtecatSetFeatureStatus .
EnableServerLog	Removed.
EnableShutdownHandler	Removed. Use RtecatSetFeatureStatus .
GetAliasesStatus	Removed.
GetStatusEx	Removed. Use RtecatGetStatus .
Log	Removed. Use Variable Database Library (RTVDB APIs).

KINGSTAR 4.x API**wRTOS 1.0 and later**

RegisterCallbackExRemoved. Use **RtecatRegisterCallbackOnReceive**.

SetCycleTimeRemoved. Use **RtecatSetFeatureStatus**.

StopLog

Removed. Use Variable Database Library (RTVDB APIs).

UnlinkRemoved. Use **RtecatCloseHandle**.

AddModuleConfiguration

Removed.

RemoveModuleConfiguration

Removed.

KsLogChannel

Removed. Use Variable Database Library (RTVDB APIs).

McTriggerSource

Removed. Use Variable Database Library (RTVDB APIs).

SafetySetting

Removed.

SubsystemStatusExRemoved. Use **RTECAT_MAINDEVICE_STATUS**.

UserVariable

Removed. Use Variable Database Library (RTVDB APIs).

AxisState

Removed.

KsConfigurationType

Removed.

KsLogDataType

Removed. Use Variable Database Library (RTVDB APIs).

KINGSTAR 4.x API**wRTOS 1.0 and later**

KsLogSeverity	Removed. Use Variable Database Library (RTVDB APIs).
KsLogSource	Removed. Use Variable Database Library (RTVDB APIs).
KsLogTriggerType	Removed. Use Variable Database Library (RTVDB APIs).
KsLogVariable	Removed. Use Variable Database Library (RTVDB APIs).
McTriggerSourceType	Removed. Use Variable Database Library (RTVDB APIs).
McTriggerType	Removed. Use Variable Database Library (RTVDB APIs).
McTriggerVariableType	Removed. Use Variable Database Library (RTVDB APIs).

Removed Managed E-CAT APIs**KINGSTAR 4.x API****wRTOS 1.0 and later**

Api.CreateAsync	Removed.
Api.DestroyAsync	Removed.
Api.Dispose	Removed.
Api.EnableAliases	Removed.
Api.GetAliasesStatus	Removed.

KINGSTAR 4.x API**wRTOS 1.0 and later**

Api.GetStatusEx	Removed. Use Library.GetStatus .
Api.Log	Removed. Use Variable Database APIs .
Api.RegisterCallback	Removed.
Api.StopLog	Removed. Use Variable Database APIs .
Api.Unlink	Removed. Use Library.CloseHandle .
Api.AbortCommand (byte)	Removed.
Api.AbortCommand (double)	Removed.
Api.AbortCommand (integer)	Removed.
Api.AbortCommand (long)	Removed.
Api.AbortCommand (short)	Removed.
Api.AbortCommand (ulong)	Removed.
Api.GetCommandStatus (byte)	Removed.
Api.GetCommandStatus (double)	Removed.
Api.GetCommandStatus (integer)	Removed.
Api.GetCommandStatus (long)	Removed.
Api.GetCommandStatus (short)	Removed.
Api.GetCommandStatus (ulong)	Removed.

KINGSTAR 4.x API**wRTOS 1.0 and later**

Api.WaitForCommandAsync	Removed.
Api.WaitForCommandAsync (byte[])	Removed.
Api.WaitForCommandAsync (byte)	Removed.
Api.WaitForCommandAsync (double)	Removed.
Api.WaitForCommandAsync (integer)	Removed.
Api.WaitForCommandAsync (long)	Removed.
Api.WaitForCommandAsync (SdoEntryDescription)	Removed.
Api.WaitForCommandAsync (SdoObjectDescription)	Removed.
Api.WaitForCommandAsync (short)	Removed.
Api.WaitForCommandAsync (uint)	Removed.
Api.WaitForCommandAsync (ulong)	Removed.
Api.WaitForCommandAsync (ushort[])	Removed.
Api.WaitForCommandAsync (ushort)	Removed.
Api.WaitForCommand (byte)	Removed.
Api.WaitForCommand (double)	Removed.
Api.WaitForCommand (integer)	Removed.
Api.WaitForCommand (long)	Removed.

KINGSTAR 4.x API**wRTOS 1.0 and later**

Api.WaitForCommand (short)

Removed.

Api.WaitForCommand (ulong)

Removed.

Api.GetMemoryPointer

Removed.

Api.GetSlaveById

Removed. Use **OpenSubDeviceByExplicitId**, **OpenSubDeviceByIndex**, or **OpenSubDeviceByStationAddress** to open a SubDevice and then use **GetSubDeviceStatus** to get its status.

Api.GetIOBuffers

Removed.

Api.GetIOByIndex

Removed.

Api.ReadIOAlias

Removed.

Api.WriteIOAlias

Removed.

Api.EoeSetSlaveEoelp

Removed.

Api.FoeReadFileToBuffer

Removed.

Api.FoeWriteFileFromBuffer

Removed.

Api.RegisterMailboxCallback

Removed.

Api.GetAxisByIndex

Removed. Use **OpenAxisByIndex** or **OpenAxisByLocation** to open an axis and then use **GetAxisStatus** to get its status.

Api.GetAxisBuffers

Removed.

SlaveStatus Class

Removed. Use **AxisStatus Class** and **ModuleStatus Class**.

Support

For help with <Add Product Name variable>, contact IntervalZero Technical Support by phone or visit <https://www.intervalzero.com/en-support/en-customer-service/> to access the Customer Support Portal (requires valid credentials) and view online product Help, Support and Lifecycle policies, and Product Release Notices.

Contacting Technical Support by Phone

Note: If you purchased an IntervalZero product through a third-party reseller, please contact the reseller for support.

Location	Number	Hours
United States	1-781-996-4481 At the prompt, press 3 for Support.	Monday - Friday, 8:30 a.m. – 5:30 p.m. US Eastern Time (GMT-500), excluding holidays.
R.O.C. Taiwan	+ 886-2-2556-8117	Monday - Friday, 9:00 a.m. – 5:00 p.m. Taipei Standard Time (GMT+8), excluding holidays.

Before Calling Technical Support

Please have this information ready when you contact IntervalZero Technical Support:

- Your Support ID

Customers who purchase direct support receive an e-mail address and password for accessing the IntervalZero Customer Support Portal.
- Your <Add Product Name variable> version number

Note: You must have a valid maintenance contract to receive product support.

Index

A

- activating 7
- activation 6
 - dongles 8
 - license types 8

C

- configuring
 - E-CAT 7
 - Runtime 10
- controlling
 - components 10
- cores
 - configuring 6

E

- E-CAT APIs
 - Breaking Changes to APIs 19
 - Breaking Changes to Header and Library Files 18
 - Breaking Changes to Managed APIs 130
 - Breaking Changes to Namespaces 18
 - New APIs 14
 - New Managed APIs 16
 - Removed APIs 227
 - Removed Managed APIs 230

L

- licenses 8
- licensing 6

P

- product
 - activating 7
 - comparison 2

- packages 2
- project
 - configurations 13
 - settings 13

R

- Runtime
 - configuring 10
 - controlling 10

S

- SDK
 - code changes 14

T

- tools 11

W

- wRTOS product packages 6