

KEYKV

Driver Guide

Version: 1.00

Contents

| | |
|--|----|
| KEYKV Driver..... | 3 |
| Driver specifications | 3 |
| Driver files | 3 |
| Target Display Unit | 4 |
| Device specifications | 4 |
| Working with Communication Drivers | 5 |
| Adding a communication driver to the project..... | 5 |
| Configuring the driver communication settings..... | 6 |
| About driver worksheets..... | 8 |
| Adding and configuring a Standard Driver Sheet | 8 |
| Configuring the Main Driver Sheet..... | 11 |
| Checking the Communication Driver task..... | 14 |
| Supported Registers | 15 |
| Troubleshooting | 19 |
| Checking status codes | 19 |
| Monitoring device communications..... | 19 |
| Revision history | 20 |
| Contacting Technical Support..... | 20 |

CHAPTER 1

KEYKV Driver

The KEYKV Driver for KEYENCE KV-7000 - 8000 Series devices enables Ethernet communication between the Studio system and remote devices, as per the specifications discussed in this document.

This document assumes you have read the "Development Environment" section in the main Studio documentation.

We assume you are familiar with working in a Windows environment, and we do not attempt to explain Windows navigation, file management, and so forth. If you are unfamiliar with any of these procedures, we recommend using the Windows Help feature or consulting your Microsoft Windows documentation.

Driver specifications

This section identifies the software and hardware components required to implement communication between the KEYKV driver in Studio and remote devices.

Driver files

The KEYKV driver package comprises the following files, which are automatically installed in the Drv folder of the Studio application directory:

KEYKV.DLL: Compiled driver.

KEYKV.INI: Internal driver file. *You must not modify this file.*

KEYKV.MSG: Internal driver file defining error messages for error codes. (These error codes are described in detail in the Troubleshooting section.) *You must not modify this file.*

KEYKV.PDF: This document provides information about using the driver.

Note: You must use a compatible PDF reader to view the KEYKV.PDF file. You can install Acrobat Reader from the Studio installation CD, or download it from Adobe's website.

You can use the KEYKV driver on the following operating systems:

- Windows:
 - Windows 11
 - Windows 10, version 1909 or later (including LTSC/LTSB versions)
- Windows Server:
 - Windows Server 2022
 - Windows Server 2019
 - Windows Server 2016
- Windows Embedded:
 - Windows 11 IoT Enterprise
 - Windows 10 IoT Enterprise (LTSC/LTSB version only)

Target Display Unit

To establish communication, the target display unit must be one of the following series:

- IPC series
- PC/AT series

Device specifications

To establish communication, the external device must meet the following specifications:

| Series Name | CPU | Link Unit | Interface |
|----------------|--------------------|---------------------------|-----------|
| KV-7000 Series | KV-7300 | KV-LE20V | Ethernet |
| | | KV-LE21V | |
| | | KV-EP21V | |
| | | KV-XLE02 | |
| | KV-7500 | KV-LE20V | |
| | | KV-LE21V | |
| | | KV-EP21V | |
| | | Ethernet port on CPU Unit | |
| | | KV-XLE02 | |
| | KV-8000 Series | KV-8000 | |
| KV-LE20V | | | |
| KV-LE21V | | | |
| KV-EP21V | | | |
| KV-XLE02 | | | |
| KV-X Series | KV-X310 KV-X520 | Ethernet port on CPU Unit | |
| | KV-X530 KV-X550 | KV-XLE02 | |

CHAPTER 2

Working with Communication Drivers

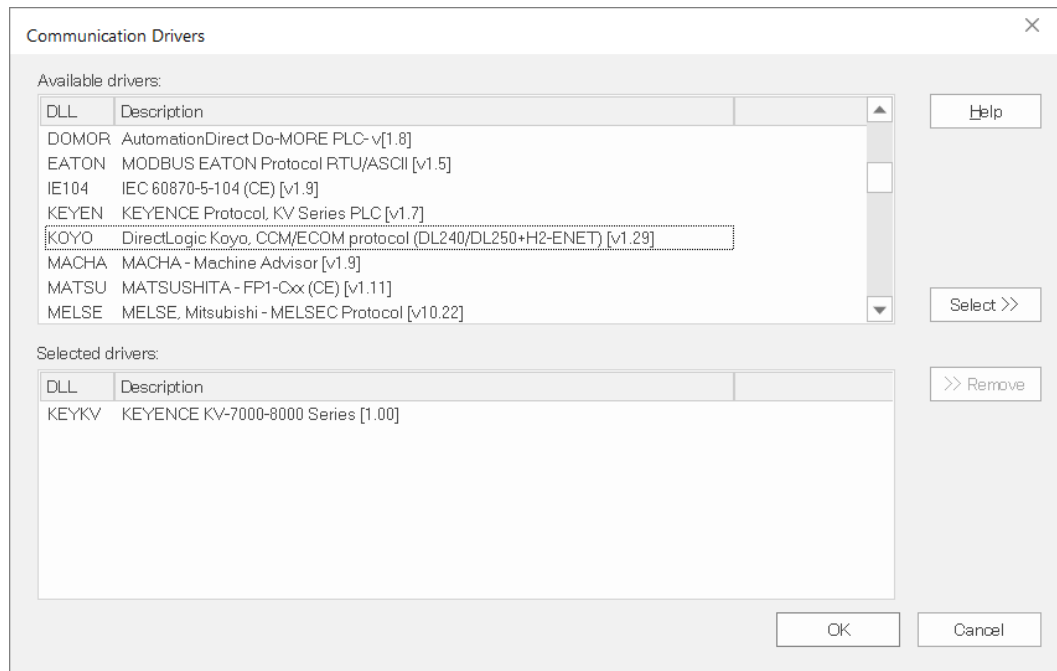
This section explains how to add a communication driver to the project and how to configure the communication settings for the driver.

Adding a communication driver to the project

This section explains how to add a communication driver to the project.

1. On the **Insert** tab of the ribbon, in the **Communication** group, click **Add/Remove Driver**.

The *Communication Drivers* dialog is displayed.



2. In the *Available drivers* list, click the communication driver to add.
3. Click **Select** to add the driver to the *Selected drivers* list.
4. Click **OK**.

The *Communication Drivers* dialog is closed and the selected driver is inserted in the **Drivers** folder in the Project Explorer.

Configuring the driver communication settings

This section explains how to configure the communication settings of the driver.

You must add the communication driver to the project before you can configure its settings. For more information, see *Adding a communication driver to the project* on Page 5.

The general procedure for configuring the driver communication settings is the same for all drivers. However, the specific settings are different for each driver, depending on the options and protocols used by the target device.

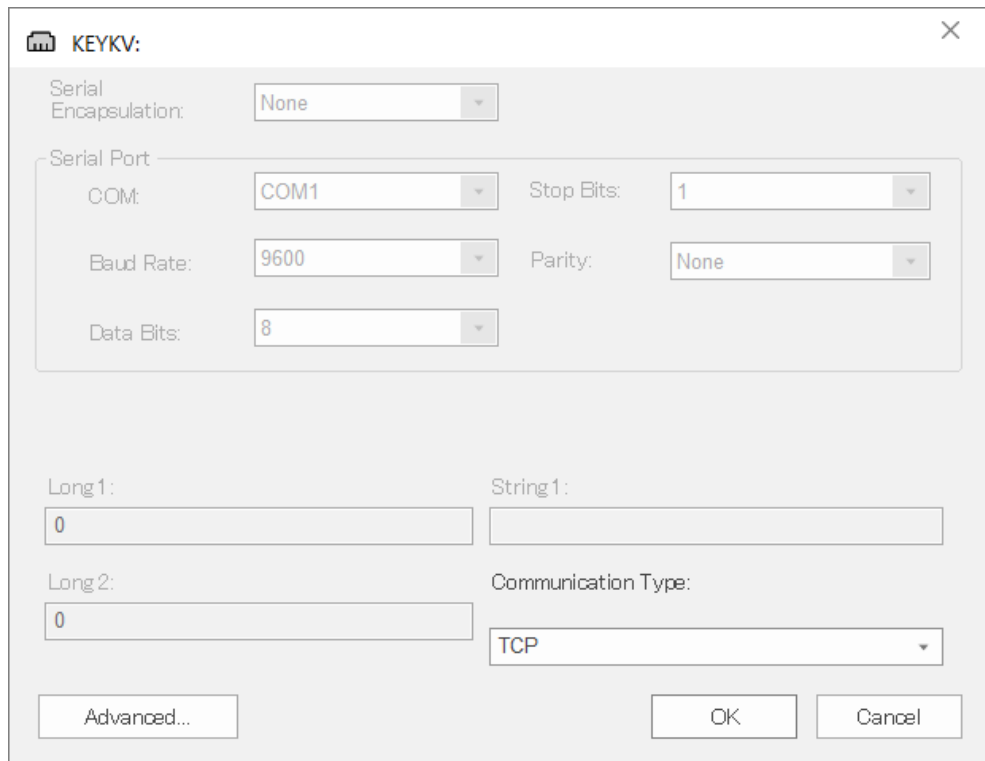
To configure the communication settings:

1. In the Project Explorer's **Comm** tab, expand the **Drivers** folder.

The folder contains the drivers that are currently enabled. If you do not see the driver that you want to configure, then it needs to be added.

2. Right-click the driver to configure, and then from the shortcut menu, click **Settings**.

The *Communication Settings* dialog is displayed.



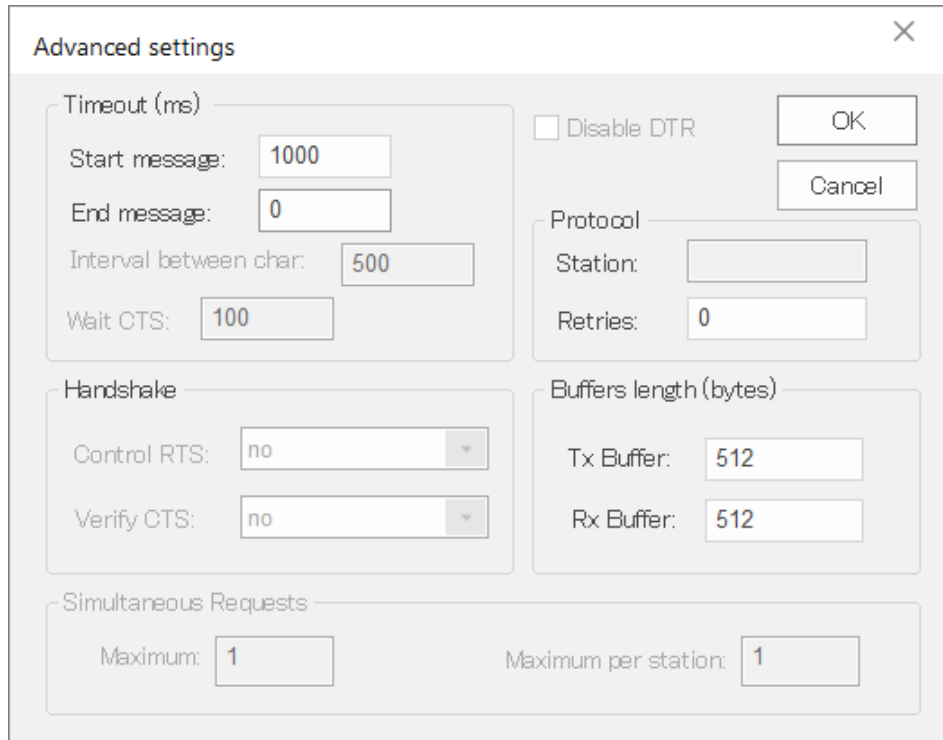
The image shows a dialog box titled "KEYKV:". It contains several settings for communication. At the top, "Serial Encapsulation:" is set to "None". Below this, a "Serial Port" section is expanded, showing "COM:" set to "COM1", "Baud Rate:" set to "9600", "Data Bits:" set to "8", "Stop Bits:" set to "1", and "Parity:" set to "None". Below the "Serial Port" section, there are two text input fields: "Long1:" with the value "0" and "String1:" which is empty. Below these, there are two more text input fields: "Long2:" with the value "0" and "Communication Type:" which is a dropdown menu set to "TCP". At the bottom of the dialog, there are three buttons: "Advanced...", "OK", and "Cancel".

3. Configure the remaining driver-specific settings as required.

Driver-specific communication settings

| Setting | Default Value | Valid Values |
|--------------------|---------------|---|
| Communication Type | UDP | <ul style="list-style-type: none">• UDP• TCP |

- Click **Advanced** to display the *Advanced settings* dialog.



The **Advanced settings** dialog box contains the following controls:

- Timeout (ms)**
 - Start message: 1000
 - End message: 0
 - Interval between char: 500
 - Wait CTS: 100
- Handshake**
 - Control RTS: no
 - Verify CTS: no
- Simultaneous Requests**
 - Maximum: 1
 - Maximum per station: 1
- ☐ Disable DTR
- Protocol**
 - Station:
 - Retries: 0
- Buffers length (bytes)**
 - Tx Buffer: 512
 - Rx Buffer: 512
- OK
- Cancel

Note: The settings in this dialog box are standard for all drivers. For more information about configuring these settings, see the "Communication" chapter of the *Help Manual*.

- Click **OK** to close the *Advanced settings* dialog.
- Click **OK** to save the settings and close the Communication Settings dialog.

CHAPTER 3

About driver worksheets

Like the other parts of the project, communication with remote devices is controlled with worksheets. This section explains how to add worksheets to the project and associate project tags with device registers.

Each selected driver includes a Main Driver Sheet (MDS) and one or more Standard Driver Sheets (SDS). The Main Driver Sheet defines tag/register associations and driver parameters that are in effect at all times, regardless of project behavior. In contrast, Standard Driver Sheets define tag/register associations that are triggered by specific project behavior.

The configuration of these worksheets is described in detail in the "Communication" chapter of the *Help Manual*, and the same general procedures are used for all drivers. Please review the procedures before continuing.

For the purposes of this document, only KEYKV driver-specific parameters and procedures are discussed in this document.

Adding and configuring a Standard Driver Sheet

By default, a communication driver does not include any Standard Driver Sheets. This section explains how to add a Standard Driver Sheet to the project and then how to configure the SDS.

The KEYKV driver must be added to the project before you can configure any of its worksheets. For more information, see *Adding a communication driver to the project* on Page 5.

Insert SDS to define additional tag/register associations triggered by specific project behavior.

Note: Most of the settings in the SDS are standard for all drivers. For more information about configuring these settings, see the "Communication" chapter of the *Help Manual*. The **Station** and **I/O Address** fields, however, use syntax that is specific to the KEYKV driver.

1. Do one of the following.
 - On the **Insert** tab of the ribbon, in the **Communication** group, click **Driver Sheet** and then select **KEYKV** from the list.
 - From the Project Explorer's **Comm** tab, right-click the **KEYKV** folder and from the shortcut menu, click **Insert**.

A new KEYKV driver worksheet is inserted into the **KEYKV** folder and opened for configuring.

Description

☐ Increase priority

Read Trigger: Enable Read when: Read Completed: Read Status:

Write Trigger: Enable Write on Tag: Write Completed: Write Status:

Station: Header: ☐ Min: ☐ Max:

| | Tag Name | Address | Div | Add |
|---|-------------|-------------|-------------|-------------|
| | Filter text | Filter text | Filter text | Filter text |
| * | | | | |
| * | | | | |

Note: Worksheets are numbered in order of creation, so the first worksheet is KEYKV001.drv.

2. Configure the Station and Header fields as described below.

For more information, see *Station* on Page 9 and *Header* on Page 9.

Note: Station field cannot be left empty.

3. For each tag/register association, insert a row in the worksheet body and then configure the row's fields as described below.

For more information, see *Tag Name* on Page 10 and *Address* on Page 10.

4. Save and close the worksheet.

Station

Specify the station in the driver sheet using the following syntax.

<IP Address>:<Port Number>

<IP Address> is the IP Address of the device on the Ethernet network.

<Port Number>: 1024...65535

If the Port Number is not set, uses the default port 8501.

During runtime, you can specify a tag in curly brackets to change the station (e.g. {Station}), but the tag that is referenced must follow the same syntax and contain a valid value.

Header

Specify the address of the first register of a block of registers on the target device.

The addresses declared in the body of the worksheet are simply offsets of this Header address.

When Read and Write actions are executed for the entire worksheet (using Read Trigger and Write Trigger, respectively), the entire block of registers/coils are scanned from the first to the last address.

The **Header** field uses the following syntax:

<Address Type>:(Address Number)

When a header is “blank” and the setting value is not supported device type, header value is set as default address.

Default address : DM:0

When the header is already defined, and the setting is for an unsupported device type or an unsupported range, the header is not updated.

This driver supports Curly Brackets in the Header, the same as other drivers. Set the header with a String variable, and if the string is not a valid address name, the driver sets an “Invalid header” error to the Read or Write Status.

Tag Name

Type the name of the project tag.

Address

Specify an offset from Header address.

The **Address** field uses the following syntax:

[Format]<Address Offset>.[Bit]

[Format]<Address Offset>.[Length] (String Format)

[Format] (Optional)

This parameter is the format of address values.

| Format | |
|--------|--------------------------------------|
| W | Signed 16 bits |
| UW | Unsigned 16 bits |
| DW | Signed 32 bits |
| UDW | Unsigned 32 bits |
| F | 32 bit float |
| S | String value (requires the “Length”) |

If [Format] is not set, all addresses are treated as follows.

- Bit Address: 1-bit value
- Word Address: Signed 16-bit value
- Double Word Address: Signed 32-bit value

<Address Offset>

This is a parameter added to the (Address Number) parameter configured in the **Header** field.

[Length] (Optional)

This value denotes the length of the string to read from a String type address.

The valid range: 1...256.

If the setting value is out of range, the value is not updated.

Note: The tag names are case sensitive. Tags with incorrect spelling or case will result in errors.

Each Standard Driver Sheet can have up to 4096 rows. However, the **Read Trigger**, **Enable Read When Idle**, and **Write Trigger** commands attempt to communicate the entire block of addresses that are configured in the sheet. So, if the block of addresses is larger than the maximum block size supported by the driver protocol, then it will result in a communication error (e.g., "invalid block size") during runtime. Therefore, the maximum block size imposes a practical limit on the number of rows in the sheet.

Examples of Header and Address

For examples of how register addresses are composed using the values in the I/O Address field, see the following table:

| Register Address in the Device | Representation | Header Field | Address Field |
|--------------------------------|--------------------|--------------|---------------|
| D10 | Decimal (Word) | D:0 | 10 |
| | | D:0 | DW10 |
| | | D:0 | S10.12 |
| W0C | Hexadecimal (Word) | W:0 | 0C |
| | | W:0 | DW0C |
| | | W:0 | S0C.A |

Configuring the Main Driver Sheet

When you add the KEYKV driver to the project, the Main Driver Sheet is automatically included in the **KEYKV** folder in the Project Explorer. This section describes how to configure the Main Driver Sheet.

The KEYKV driver must be added to the project before you can configure any of its worksheets. For more information, see *Adding a communication driver to the project* on Page 5.

The Main Driver Sheet defines tag/register associations and driver parameters that are in effect at all times, regardless of project behavior. The worksheet is continuously processed during runtime.

Note: Most of the settings in this worksheet are standard for all drivers. For more information about configuring these settings, see the "Communication" chapter of the *Help Manual*. The **Station** and **I/O Address** fields, however, use syntax that is specific to the KEYKV driver.

1. Do one of the following.

- ☐ On the **Insert** tab of the ribbon, in the **Communication** group, click **Main Driver Sheet** and then select **KEYKV** from the list.
- ☐ From the Project Explorer's **Comm** tab, expand the **KEYKV** folder and then double-click **MAIN DRIVER SHEET**.

The Main Driver Sheet is displayed.

- For each tag/register association to create, insert a row in the worksheet body and then configure the row's fields.

For more information, see *Tag Name* on Page 12, and *Station* on Page 12, and *I/O Address* on Page 12.

Note: The Main Driver Sheet can have up to 32767 rows. If you need to configure more than 32767 communication addresses, then either configure additional Standard Driver Sheets or create additional instances of the driver.

- Save and close the worksheet.

Tag Name

Type the name of the project tag.

Station

Please refer to information on the Station field for the Standard Driver Sheet. See *Station* on Page 9.

I/O Address

Specify the address of the associated device register. The valid range of the offset value varies depending on the type of device used.

The **I/O Address** field uses the following syntax:

<Address type>:[Format](Address Number).[Bit]

<Address type>:[Format](Address Number).[Length] (String Format)

String Encode: String data is treated as UTF-8. Multi-byte character is not supported.

<Address Type> / (Address Number)

For information about available register types and valid ranges of each register, see *Supported Registers* on Page 15.

[Format] / [Bit] / [Length]

Please refer to information on the Address field for the Standard Driver Sheet. See *[Format] (Optional)* / *[Bit] (Optional)* / *[Length] (Optional)* on Page 10

Examples

For examples of how register addresses are composed using the values in the I/O Address field, see the following table:

| Register Address in the Device | Representation | I/O Address |
|--------------------------------|--------------------|-------------|
| DM10 | Decimal (Word) | DM:10 |
| | | DM:DW10 |
| | | DM:S10.12 |
| W0C | Hexadecimal (Word) | W:0C |
| | | W:DW0C |
| | | W:S0C.A |

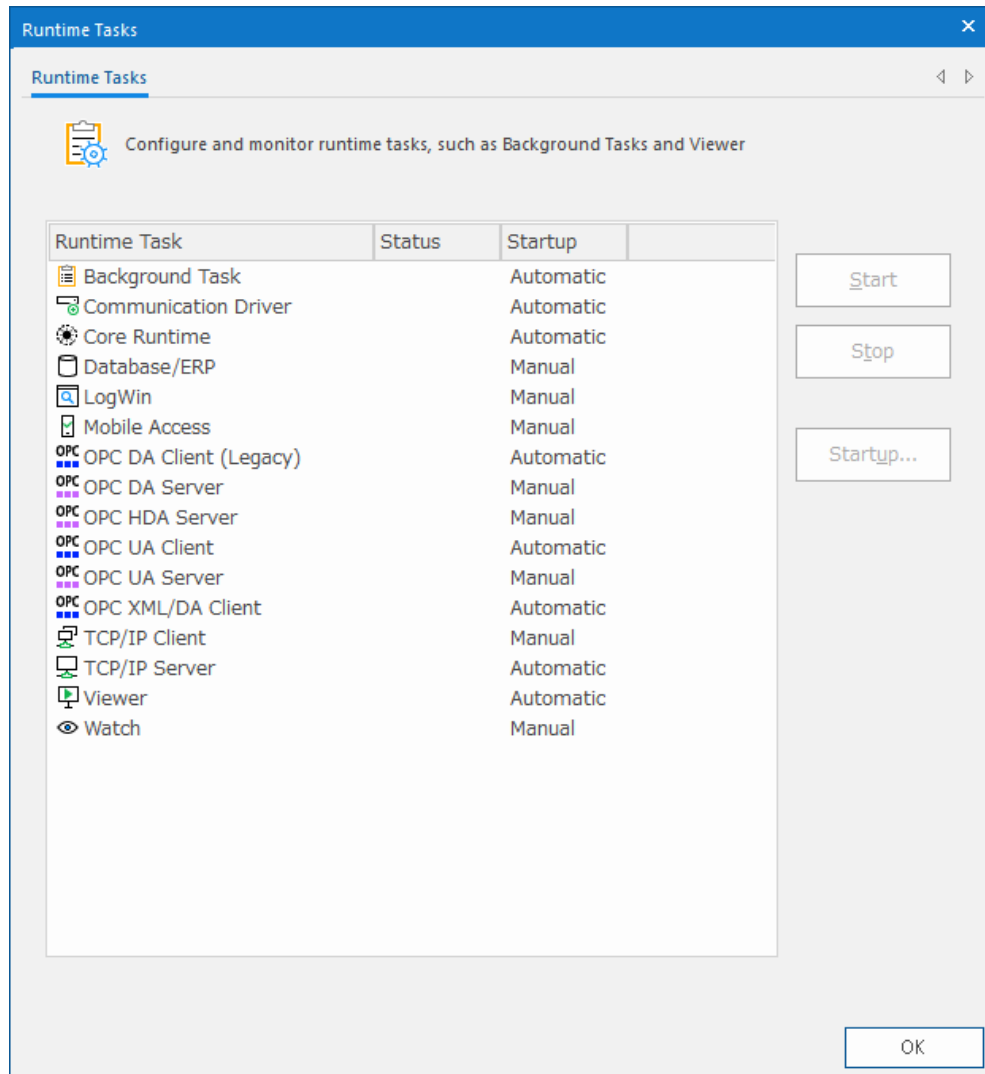
Checking the Communication Driver task

This section describes how to check the status of the Communication Driver task among the list of execution tasks.

The Communication Driver task handles communication with remote devices and the processing of the driver worksheets. By default, the task is configured to start up automatically when the project is run, but you can check it yourself.

1. On the **Home** tab of the ribbon, in either the **Local Management** or the **Remote Management** group (depending on where the project server is running), click **Runtime Tasks**.

The *Runtime Tasks* dialog is displayed.



2. Verify that the **Communication Driver** task is set to **Automatic**.
If the setting is correct, then proceed to the next step.
3. If the **Communication Driver** task is set to **Manual**, select the task and then click **Startup** to change the task to **Automatic**.
4. Click **OK** to close the *Project Status* dialog.

CHAPTER 4

Supported Registers

The ranges of supported register addresses are shown in the table below. Please note that the actual supported range of registers varies depending on the device that is used. Please check the actual range in the manual of your device.

[KV-7000 series]

| Register | | Bit Address | Word Address | 32bit | Note |
|---|----------------------|-------------|--|-------|---|
| Input Relay Output relay Internal Auxiliary Relay | | ---- | R:0-R:1999 (R:0-R:999* ²) | L/H | |
| Link Relay | | ---- | B:00-B:07FF | L/H | |
| Internal Auxiliary Relay | | ---- | MR:0-MR:3999* ¹ | L/H | |
| Latch Relay | | ---- | LR:0-LR:999 | L/H | |
| Control Relay | | ---- | CR:0-CR:79 | L/H | |
| Work Relay | | ---- | VB:00-VB:0F9F | L/H | |
| Timer (Contact) | | T:0-T:3999 | ---- | - | Bit Access Only |
| Counter (Contact) | | C:0-C:3999 | ---- | - | Bit Access Only |
| Timer (Setting value) | | ---- | TS:0-TS:3999 | L/H | Word Access Only* ³ |
| Counter (Setting value) | | ---- | CS:0-CS:3999 | L/H | Word Access Only* ³ |
| Timer (Current value) | | ---- | TC:0-TC:3999 | L/H | Word Access Only* ³ |
| Counter (Current value) | | ---- | CC:0-CC:3999 | L/H | Word Access Only* ³ |
| Data Memory | | ---- | DM:0-DM:65534 | L/H | |
| Extension Data Memory | | ---- | EM:0-EM:65534 | L/H | |
| File Register | Current bank | ---- | FM:0-FM:32767 | L/H | |
| | Serial number system | ---- | ZF:0-ZF:524287 | L/H | |
| Temporary Data Memory | | ---- | TM:0-TM:511 | L/H | |
| Control Memory | | ---- | CM:0-CM:5999 | L/H | |
| Link Register | | ---- | W:00-W:07FFF | L/H | |
| Work Memory | | ---- | VM:0-VM:50999 | L/H | |
| Index Register | | ---- | Z:1-Z:12 | L/H | Word Access Only * ³ * ⁴ |
| Digital Trimmer | | ---- | TRM:0-TRM:7 | L/H | Word Access Only Read Only* ³ |

*1 When connecting through KV-EP21V, KV-LE21V or KV-LE20V, the range is MR00000 ~ MR 99915.

*2 With CPU function version 2.2 or earlier, the R register totals 16000 relays, with a range of R:000 to R:999. The range of R that can be read/written via KV-EP21V/KV-LE21V/KV-LE20V is R:000 to R:999.

*3 32-bit device

*4 Do not write to Z11 or Z12 as they are used in the system of the External Device.

NOTE:

Read Only: Cannot write to the address. If it is used for writing parts, a runtime error occurs.

[KV-8000 series]

| Register | | Bit Address | Word Address | 32bit | Note |
|---|----------------------|-------------|------------------------------|-------|---|
| Input Relay Output relay Internal Auxiliary Relay | | ---- | R:0-R:1999* ¹ | L/H | |
| Link Relay | | ---- | B:00-B:07FF | L/H | |
| Internal Auxiliary Relay | | ---- | MR:0-MR:3999* ¹ | L/H | |
| Latch Relay | | ---- | LR:0-LR:999 | L/H | |
| Control Relay | | ---- | CR:0-CR:79 | L/H | |
| Work Relay | | ---- | VB:00-VB:0F9F | L/H | |
| Timer (Contact) | | T:0-T:3999 | ---- | - | Bit Access Only |
| Counter (Contact) | | C:0-C:3999 | ---- | - | Bit Access Only |
| Timer (Setting value) | | ---- | TS:0-TS:3999 | L/H | Word Access Only* ² |
| Counter (Setting value) | | ---- | CS:0-CS:3999 | L/H | Word Access Only* ² |
| Timer (Current value) | | ---- | TC:0-TC:3999 | L/H | Word Access Only* ² |
| Counter (Current value) | | ---- | CC:0-CC:3999 | L/H | Word Access Only* ² |
| Data Memory | | ---- | DM:0-DM:65534 | L/H | |
| Extension Data Memory | | ---- | EM:0-EM:65534 | L/H | |
| File Register | Current bank | ---- | FM:0-FM:32767 | L/H | |
| | Serial number system | ---- | ZF:0-ZF:524287 | L/H | |
| Temporary Data Memory | | ---- | TM:0-TM:511 | L/H | |
| Control Memory | | ---- | CM:0-CM:7599 | L/H | |
| Link Register | | ---- | W:00-W:07FFF | L/H | |
| Work Memory | | ---- | VM:0-VM:589823* ¹ | L/H | |
| Index Register | | ---- | Z:1-Z:12 | L/H | Word Access Only * ² * ³ |
| Digital Trimmer | | ---- | TRM:0-TRM:7 | L/H | Word Access Only Read Only* ² |

*¹ When connecting through KV-EP21V, KV-LE21V or KV-LE20V, the range is R:000 ~ R:999, MR:000 ~ MR999, VM:00000 ~ VM:63999.

*² 32-bit device

*³ Do not write to Z11 or Z12 as they are used in the system of the External Device.

NOTE:

Read Only: Cannot write to the address. If it is used for writing parts, a runtime error occurs.

[KV-X series]

| Register | | Bit Address | Word Address | 32bit | Note |
|---|----------------------|-------------|------------------------------|-------|---|
| Input Relay Output relay Internal Auxiliary Relay | | ---- | R:0-R:1999* ¹ | L/H | |
| Link Relay | | ---- | B:00-B:07FF | L/H | |
| Internal Auxiliary Relay | | ---- | MR:0-MR:3999* ¹ | L/H | |
| Latch Relay | | ---- | LR:0-LR:999 | L/H | |
| Control Relay | | ---- | CR:0-CR:79 | L/H | |
| Work Relay | | ---- | VB:00-VB:0F9F | L/H | |
| Timer (Contact) | | T:0-T:3999 | ---- | - | Bit Access Only |
| Counter (Contact) | | C:0-C:3999 | ---- | - | Bit Access Only |
| Timer (Setting value) | | ---- | TS:0-TS:3999 | L/H | Word Access Only* ² |
| Counter (Setting value) | | ---- | CS:0-CS:3999 | L/H | Word Access Only* ² |
| Timer (Current value) | | ---- | TC:0-TC:3999 | L/H | Word Access Only* ² |
| Counter (Current value) | | ---- | CC:0-CC:3999 | L/H | Word Access Only* ² |
| Data Memory | | ---- | DM:0-DM:65534 | L/H | |
| Extension Data Memory | | ---- | EM:0-EM:65534 | L/H | |
| File Register | Current bank | ---- | FM:0-FM:32767 | L/H | |
| | Serial number system | ---- | ZF:0-ZF:524287 | L/H | |
| Temporary Data Memory | | ---- | TM:0-TM:511 | L/H | |
| Control Memory | | ---- | CM:0-CM:7599 | L/H | |
| Link Register | | ---- | W:00-W:07FFF | L/H | |
| Work Memory | | ---- | VM:0-VM:589823* ¹ | L/H | |
| Index Register | | ---- | Z:1-Z:10 | L/H | Word Access Only * ² * ³ |
| Digital Trimmer | | ---- | TRM:0-TRM:7 | L/H | Word Access Only Read Only* ² |

*¹ When connecting through KV-EP21V, KV-LE21V or KV-LE20V, the range is R:000 ~ R:999, MR:000 ~ MR999, VM:00000 ~ VM:63999.

*² 32-bit device

*³ Do not write to Z11 or Z12 as they are used in the system of the External Device.

NOTE:

Read Only: Cannot write to the address. If it is used for writing parts, a runtime error occurs.

CHAPTER 5

Troubleshooting

This section lists the most common errors for this driver, their probable causes, and basic procedures to resolve them.

Checking status codes

If the KEYKV driver fails to communicate with the target device, then the database tags configured for the **Read Status** and **Write Status** fields of the driver sheets will receive a status code. Use this status code and the following tables to identify the issue that occurred and how it might be resolved.

| Error Code | Error Message | Possible Causes |
|------------|---------------------------|--|
| 0 | OK | Communicating without error. |
| 1 | Invalid Protocol | The received message is invalid. |
| 2 | Invalid Command | The Driver worksheet is configured incorrectly. |
| 3 | Error start communication | The PLC is OFF, cable is disconnected, Wrong address, and so on. |
| 4 | Error end communication | The PLC is OFF, cable is disconnected, Wrong address, and so on. |
| 9 | Error Sending data | Connection error or there is a problem with sending data. |
| 10 | Relay Number Error | Undefined relay, counter, timer, DM, CTH, or CTC number was specified. |

Errors returned from PLC as a response:

| Error Code | Error Message |
|------------|--|
| 1000 | E0 : Abnormal Device No. |
| 1001 | E2 : Program not Registered (Command M1) |
| 1004 | E4 : Write Disabled |
| 1006 | E6 : No Comments |

Monitoring device communications

You can monitor the communication status by establishing an event log in Studio's *Output* window (LogWin module). To establish a log for Field Read Commands, Field Write Commands and Serial Communication, right-click in the *Output* window and select the desired options from the pop-up menu.

You can also use the LogWin module to establish an event log on a remote unit that runs Windows Embedded. The log is saved on the unit in the celog.txt file, which can later be downloaded.

Revision history

If you are unable to establish communication between Studio and the target device, try to establish communication using the device's own programming software. Quite often, communication is interrupted by a hardware or cable problem or by a device configuration error. If you can successfully communicate using the programming software, then recheck the driver's communication settings in Studio.

Contacting Technical Support

If you must contact Technical Support, please have the following information ready:

- **Operating System** and **Project Information**: To find this information, click **Support** in the **Help** tab of the ribbon.
- **Driver Version** and **Communication Log**: Displays in the *Output* window (LogWin module) when the driver is enabled and the project is running.
- **Device Model** and **Boards**: Consult the hardware manufacturer's documentation for this information.

This section provides a log of all changes made to the driver.

| Driver Version | Revision Date | Description of Changes |
|----------------|---------------|------------------------|
| 1.00 | May, 2025 | First driver revision |