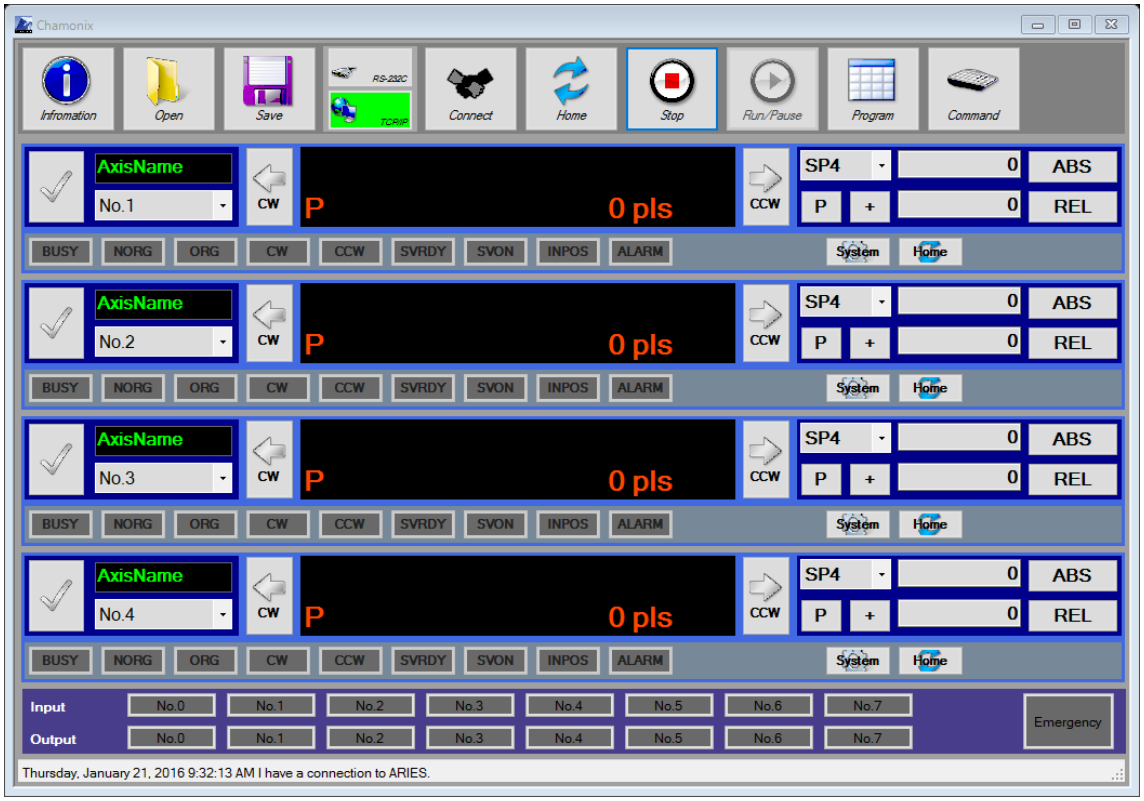
Chamonix

User's Guide

Rev. 1.2.3



KOHZU Precision Co., Ltd

Table of Contents

Table of Contents 2

1. Introduction 7

1.1. About Chamonix 7
1.2. About This User's Guide 8
1.3. Precautions for Use 9
1.4. Terms of License Agreement 10
1.5. Supported OS and System Requirements 11
1.6. Languages 12

2. Installation 13

2.1. Installation Files 13
2.2. Executing Setup 14
2.3. Deleting an Old Version 15
2.4. Confirming and Accepting License Agreement 16
2.5. Entering User Information 17
2.6. Starting Installation 18
2.7. Completing Installation 19
2.8. Checking Installation 20

3. Uninstallation 21

3.1. Control Panel 21
3.2. Uninstalling the Program 22
3.3. Confirmation Dialog 23

4. Wiring the Controller 24

4.1. RS-232C 24
4.2. TCP/IP 25
4.3. Connection to CRUX 27
4.3.1. Communications Settings of CRUX 27
Rotary switch setting for CRUX communication 27
Positions of rotary switch for CRUX communication setting 28
4.3.2. How to Connect Cables on CRUX 29

4.4. Connection to ARIES 30
4.4.1. Communication Settings of ARIES....30
Position of Rotary Switch for ARIES Communication setting....30
Setting of Rotary Switch for ARIES Communication 31
4.4.2. How to Connect Cables on ARIES 32
4.5. Connection to SC-210/410....33
4.5.1. Communication Settings of SC-210/410 33
The position of the rotary switch of SC-410....33
Rotary switch setting for communication of SC-210/410 34
5. Getting Started! 35
5.1. Starting from the Icon on the Desktop 35
5.2. Starting from the Stored File....36
5.3. Start-up Window 37
5.4. Starting Up for the First Time....38
6. Connecting the Controller.... 39
6.1. Connecting the Controller Using RS-232C....39
6.1.1. Selecting RS-232C 39
6.1.2. Selecting COM Port 40
6.1.3. How to Check the COM PORT Number....41
6.2. Connection Using TCP/IP 42
6.2.1. Selecting TCP/IP (ARIES Only) 42
6.2.2. Specifying Connection Settings of Chamonix 43
6.2.3. Changing the IP Address of ARIES 44
6.3. Connecting the Controller 45
7. Checking Version Information.... 47
7.1. Checking Version Information about This Application 47
8. Returning to Origin.... 49
8.1. What is Returning to Origin? 49
8.2. Changing the Type to Return to Origin....50
8.3. Returning to Origin for All Axes....51
8.4. Status in Returning to Origin 52

8.5. Returning to Origin for Single Axis 53
9. Moving Stages. 54
9.1. Selecting a Axis to Control 54
9.2. Setting Axis Names 55
9.3. Switching the Current Position Display 56
9.4. Selecting Speed Table 57
9.5. Setting Resolution of Axes 58
9.6. Manual Movement 60
10. Deciding Position 61
10.1. Setting of Deciding Position with Absolute Position Management 61
10.2. Executing of Deciding Position with Absolute Position Management 62
10.3. Setting of Deciding Position with Relative Position Management 63
10.4. Executing of Deciding Position with Relative Position Management 64
10.5. Executing Multi-Axis Drive 65
11. Receiving Emergency Stop! 67
11.1. Receiving an Emergency Stop Signal (ARIES) 67
11.2. Recovering from Emergency Stop (ARIES) 68
11.3. Recovering from Emergency Stop (CRUX) 69
12. Triggr Output 70
12.1. Output of One Shot Trigger Signal (ARIES) 70
13. Changing System Parameters 71
13.1. Starting Parameter Setting 71
13.2. Setting Parameters 73
13.3. Saving System Parameters to Files 75
13.4. Printing Parameters 77
13.5. Properties of System Parameters 79
13.6. Change of Speed table 81
14. Using Programmable Grid 82
14.1. Starting Programmable Grid 82
14.2. Basic Features of Programmable Grid 83
14.3. Running Programs 84
14.4. Stopping Programs 85
14.5. Stopping Programs Temporarily 86

14.6. Inputting and Outputting Program Data 87
14.7. Commands of Programmable Grid 88
14.7.1. ORG: Returning to Origin....88
14.7.2. ABS: Deciding Position with Absolute Position Management 89
14.7.3. REL: Deciding Position with Relative Position Management 90
14.7.4. WOT: Controlling General-Purpose Output 91
14.7.5. FOR: Repeat Loop....92
14.7.6. IF: Running with Conditions....93
14.7.7. GOTO: Jumping to a Line....94
14.7.8. SBY: Standby 95
14.7.9. TFR : One Shot Trigger 96
14.7.10. TRG : Trigger Output Synchronized with Driving Axis....97
14.7.11. Command List of Programmable Grid 98
15. Entering Remote Commands Directly 99
15.1. Starting Command Control....99
15.2. Sending Commands from Command Control 100
16. Control from Outside Using API 102
16.1. Control from Outside with Windows API 102
16.2. Control from External Applications 103
16.3. Sending the APS Command from External Applications....104
16.4. Sending the RPS Command from External Applications 106
16.5. Sending the STP Command from External Applications....108
16.6. Sending the ORG Command from External Applications....110
16.7. Canceling Standby of Programmable Grid from External Applications 112
16.8. Sample Program for External Control 114
17. Exiting from Chamonix 115
17.1. Exiting from Chamonix 115
18. Others....116
18.1. Folder to Save Communication Logs....116
18.2. Checking Communication Logs 117
Revision History....118

1. Introduction

1.1. About Chamonix

Chamonix is an application to control motor controllers from KOHZU Precision. This application can control motor controllers from KOHZU Precision, KOSMOS series (ARIES, LYNX), and old SC series[1] (SC-20, SC-21, SC-200, SC-400, SC-210, SC-410).

Chamonix has GUI to control up to four axes at a time and a programing grid like simplified macro. It offers work with external application through Windows API.

RS-232C serial communication is used to communicate with the controller.

This application may be subject to modify without prior notice.
Read through the terms of the License Agreements carefully for use.

[1] Use of SC series may require upgrading of firmware.

1.2. About This User's Guide

This guide describes Chamonix, an application to control motor controller.

Please read and understand this document thoroughly to utilize the performance and functions of Chamonix in the best condition.

In addition, keep this document in a convenience place for future reference.
Please forward this user's guide to the end user.

Chamonix supports Japanese and English; however, this guide shows its GUI in Japanese.

1.3. Precautions for Use

Make sure that this application properly works in start of work and operation.

Provide sufficient safety measures to prevent various damage that may arise from accidental failure of this application.

If this product is used in improper conditions other than the terms provided in operation manuals and specifications decided separately, its features and performance cannot be guaranteed.

When this products is used with other devices, use the product after due consideration because the features and performance may not be satisfied depending on conditions of use and environment.

Please do not use to protect a human body.

Avoid the use with other application. Otherwise, features and performance may not be satisfied.

1.4. Terms of License Agreement

Please read this License Agreement carefully before using the Application.

Scope

- This License Agreement applies to Chamonix, a control application supplied as a package; various associated libraries; and all attached documents like manuals describing how to use them. Chamonix, associated libraries, and all attached documents are collectively referred to as the "Application". These documents include all media such as printed materials and electronic files.

Use Rights

- KOHZU Precision Co. grants to USER only non-exclusive right to use the Application if you agree to all the terms of this License Agreement. This non-exclusive right allows USER granted the license from KOHZU Precision Co. to install and use the Application on a single computer. USER shall not have any other rights except the non-exclusive right. This right does not include charge-free support for users.

Copyright

- Copyrights in and to the Application and all documents with the Application and libraries are owned by KHOZU Precison Co. or persons or entities having its applicable copyrights, if not specially provided. The Application is protected by copyright laws and international treaty provisions.

Forbidden Clause

- USER may not resell or distribute the Application , the libraries, or the attached documents to third parties by reproducing them. USER shall not reproduce the Application or the libraries unless for backup purpose. USER may not modify, reverse engineer, disassemble, or decompile the Application. These are strictly prohibited.

Limitation of Liability

- In no event shall KOHZU Precision Co. be liable for direct, indirect, or consequential damages of USER and third parties whatsoever arising out of use of the Application. KOHZU Precision Co. shall not be responsible for any damage of data, programs, or other intangible property, or any loss of business profits, of USER or third parties.

Others

- KOHZU Precision Co. may update the Application at any time. Terms to offer the update shall be decided by KOHZU Precision Co. If the Application has been updated, KOHZU Precision Co. shall be under no obligation to offer updates compatible with the old application. KHOZU Precision Co. may change this License Agreement anytime. If the License Agreement is changed, the former License Agreement shall be nullified and the latest License Agreement applied.

Termination of Agreement

- KHOZU Precision Co. shall have the right to immediately terminate this License Agreement if you fail to comply with or do not agree to the terms and conditions of this License Agreement. In such event, you shall return the software and all copies of the software to KOHZU Precision Co. or destroy them.

1.5. Supported OS and System Requirements

Supported OS

- Chamonix officially support Windows7SP1® and Windows10®.
- Supports both x86 and x64. Installation folders are "Program Files" for x86 and "Program Files(x86)" for x64.

Prerequisites

- Chamonix is developed by using Microsoft .NET Framework. For installation, you need Microsoft .NET Framework Full 4.7.2 or later.
- Your system must have 600MB or more free memory space on startup.
- You need interfaces to communicate with the controller, including RS-232C and TCP/IP.

2 3 4

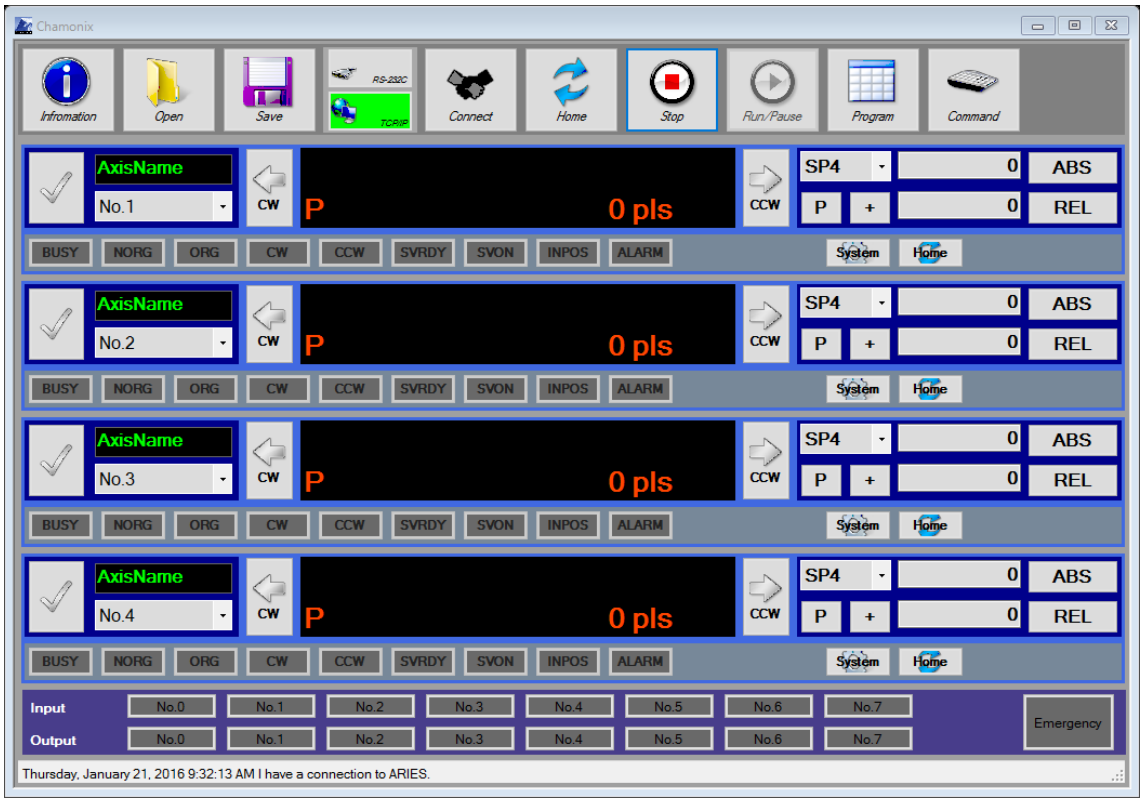
[2] The conditions are applicable to Chamonix version 1.2.2.1 or later.

[3] Windows® and .NET Framework are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

[4] WindowsXP®, Windows8® and Windows8.1® are not supported.

1.6. Languages

Chamonix supports Japanese and English. The language is dependent on the language type that is input in Windows.



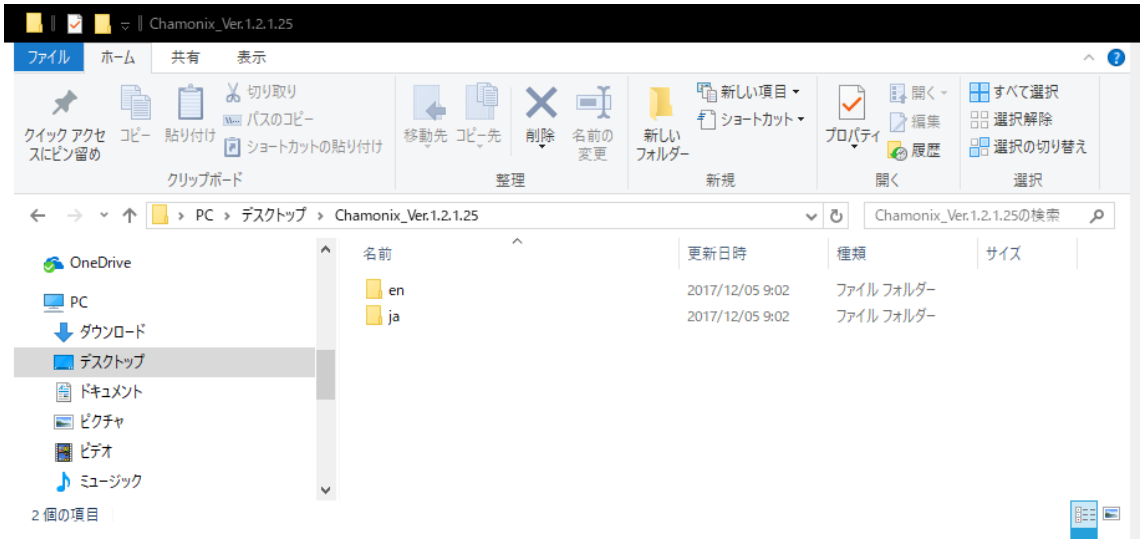
2. Installation

2.1. Installation Files

For installation, you must have administrator privileges. The installation DISK folder consists of files shown below. Click and run Setup to install.

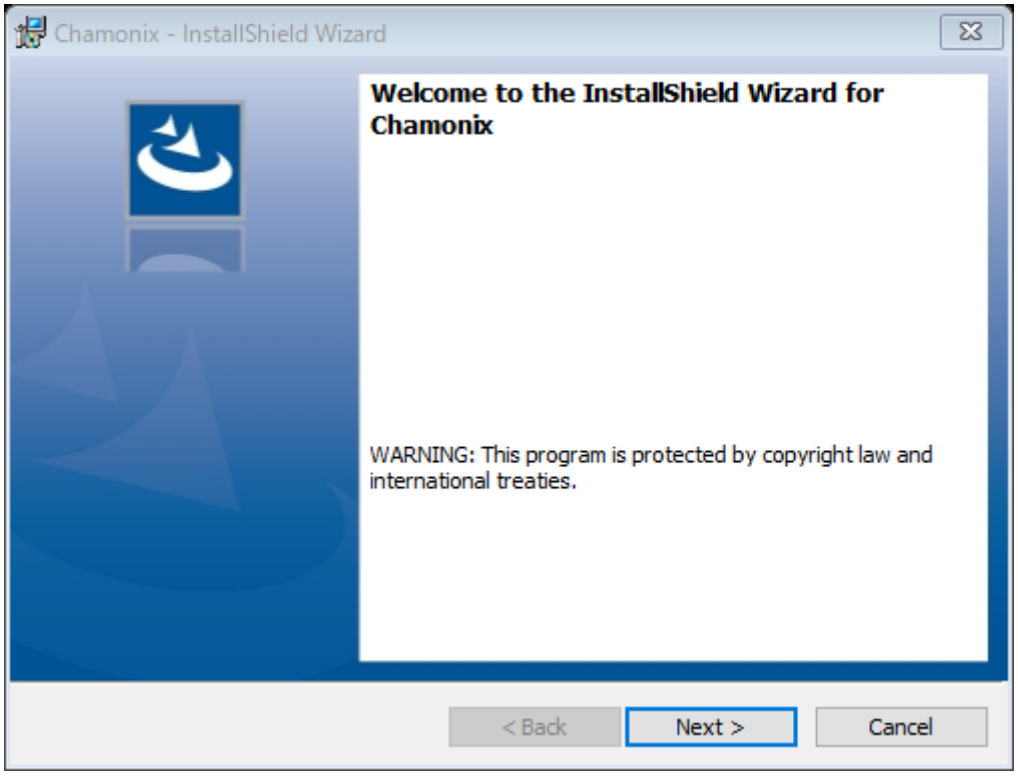
Installation files downloaded from our web site is compressed in a zip file. When you extract the zip file, folders will be created as shown below.

Please do not move and change these folders.



2.2. Executing Setup

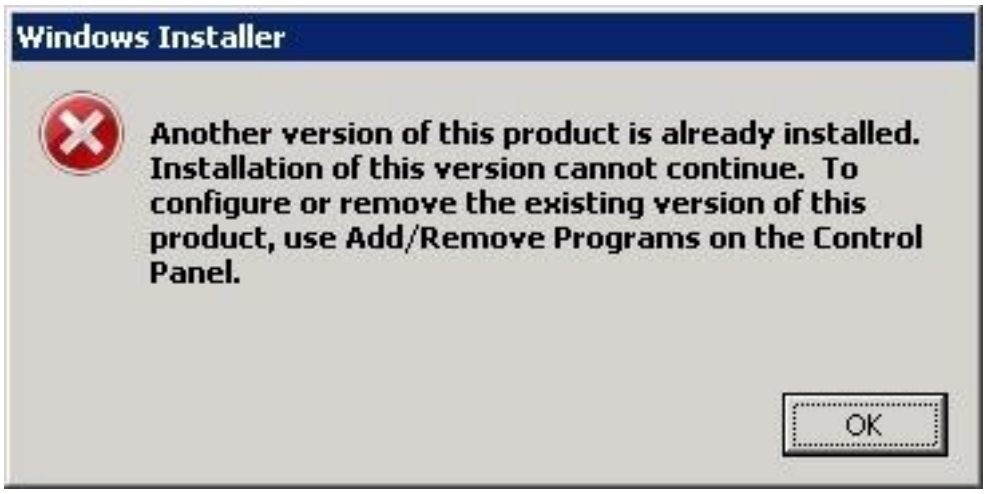
When you execute Setup.exe, the InstallShield®[5] wizard for Chamonix opens.



Press Next to continue the installation with this wizard. Press Cancel to finish the installation. Then, a window to confirm and accept License Agreement will appear.

[5] InstallShield is a registered trademark or trademark of Flexera Software, Inc. and/or InstallShield Co. Inc. in the United States and/or other countries.

2.3. Deleting an Old Version



If Chamonix or a part of Dynamic Link Library constituting Chamonix have been already installed, a message[6] may appear as shown above.

If this message appears, you must remove installed Chamonix using the Add or Remove Programs icon in Control Panel.

Refer to Uninstallation for deletion.

[6] A window for Microsoft Windows 8.1. A different message may appear depending on OS.

2.4. Confirming and Accepting License Agreement

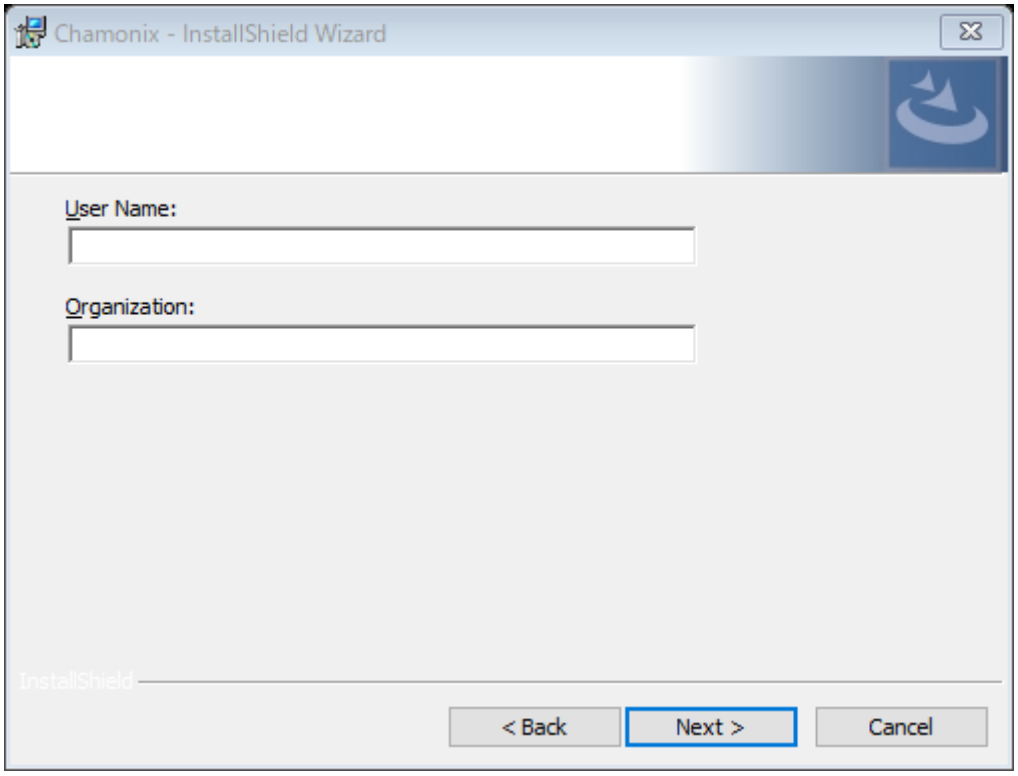
If you agree to the terms of the License Agreement, select "I accept the terms of the License Agreement."

If you do not agree to the terms of the License Agreement, select "I do not accept the terms of the License Agreement." In this case, the Next button is inactive.

Press the Print button to print "The License Agreement".

If you agree to the terms of the License Agreement, press Next to move to Entering User Information.

2.5. Entering User Information



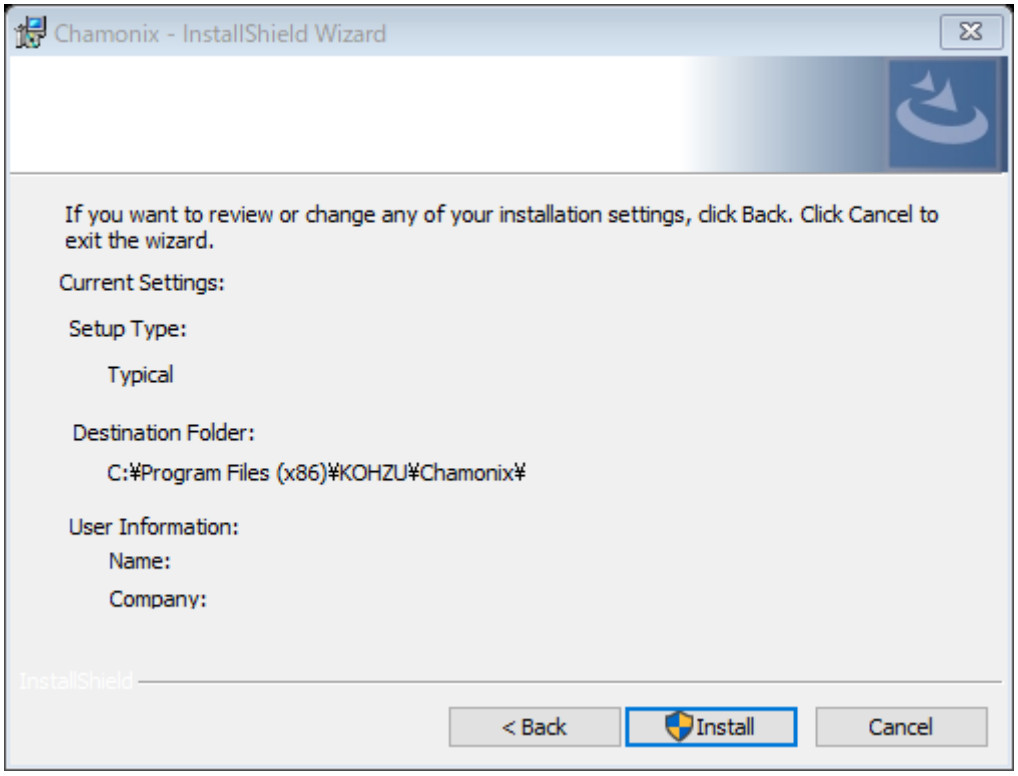
Enter a user name and an organization.

Entering user information is optional.

After entry, click Next. Click Back to check the terms of the License Agreement again.

Click Cancel to cancel the installation.

2.6. Starting Installation



Press Install to start installation.

For installation, you must have administrator privileges.

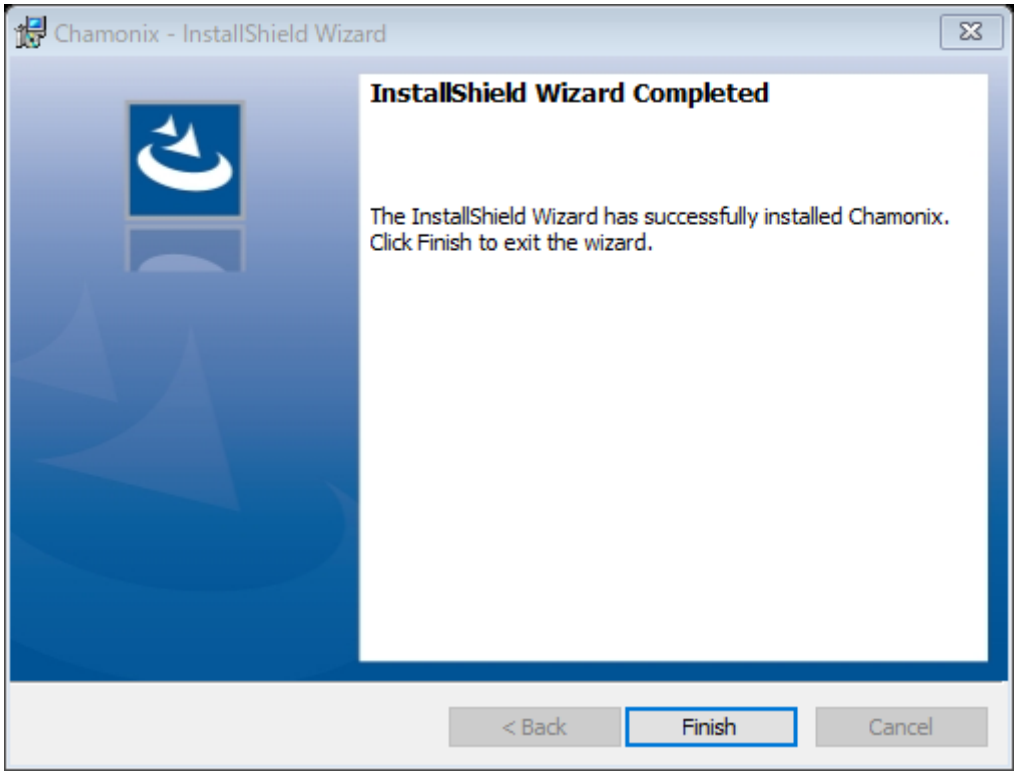
A user account control dialog may open and ask for starting installation during installation. In this case, check our code signing certificate of the Windows installer and then press Yes. Press No to stop the installation.

The destination folder for installation cannot be changed.

Press Back to back to Entering User Information.

Press Cancel to cancel the install wizard.

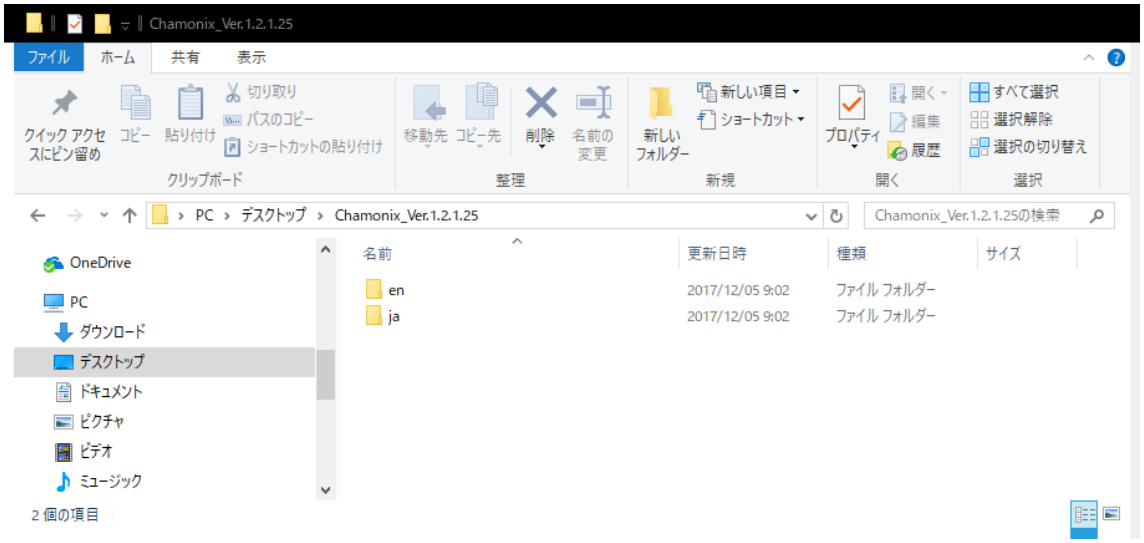
2.7. Completing Installation



Press Install in the previous window to start installing Chamonix. If the above window appears, installation is successfully complete.

Click Finish to finish the wizard.

2.8. Checking Installation

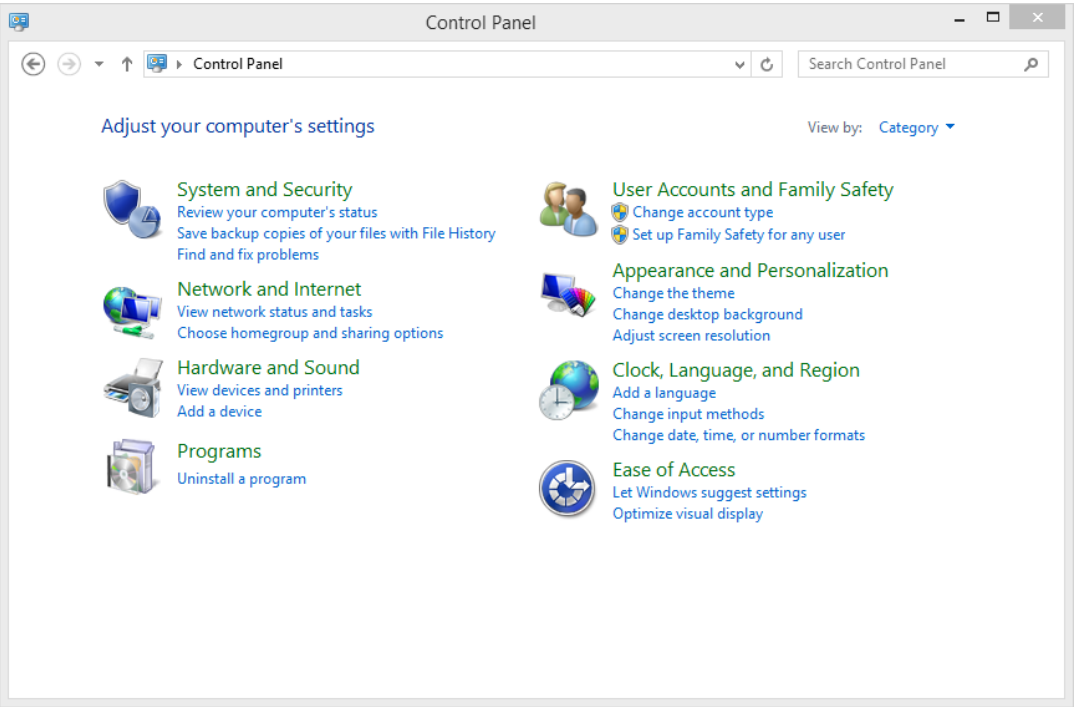


When the installation finished, the six files will be generated in the C:¥Program Files(x86)¥KOHZU¥Chamonix folder as shown above[7].

[7] A window for Microsoft Windows 8.1. A different message may appear depending on OS.

3. Uninstallation

3.1. Control Panel



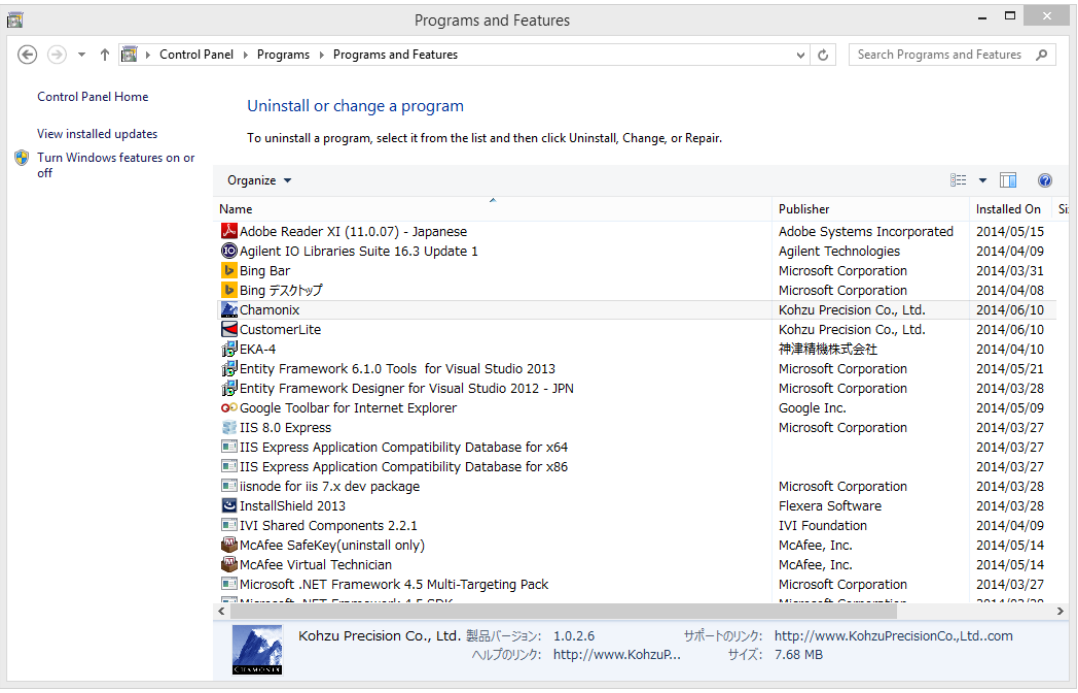
Open Programs in Control Panel[8]. Then click Programs and Features.

Or click Uninstalla program directly in Control Panel.

The Uninstall or Change a program window opens as shown in the next page.

[8] The above illustration shows an example of Control Panel in Microsoft Windows 8.1.

3.2. Uninstalling the Program

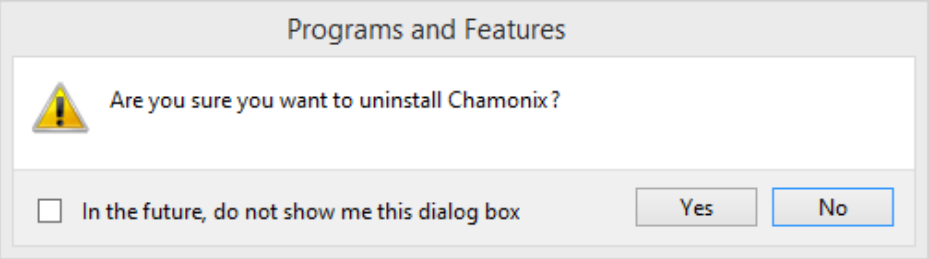


Select Chamonix from the displayed program list.

Click Uninstall to uninstall Chamonix.
A message dialog appears for confirmation.

Click Restore to restore Chamonix.

3.3. Confirmation Dialog



Click Yes in the message dialog to uninstall the program.

Installed Chamonix.exe, StageControl.dll, SerialControl.dll, SystemParam.dll, and Chamonix.chm are deleted in the uninstallation, but SerialContl.log and config.xml generated by application after installation are not deleted.

To delete these files, you must delete the files directly with Explorer.

The files are stored in C:¥Program Files(x86)[9]¥Common Files¥KOHZU.

[9] For 32-bit OS, the file is in C:¥Program Files ¥Common Files¥KOHZU¥Chamonix.

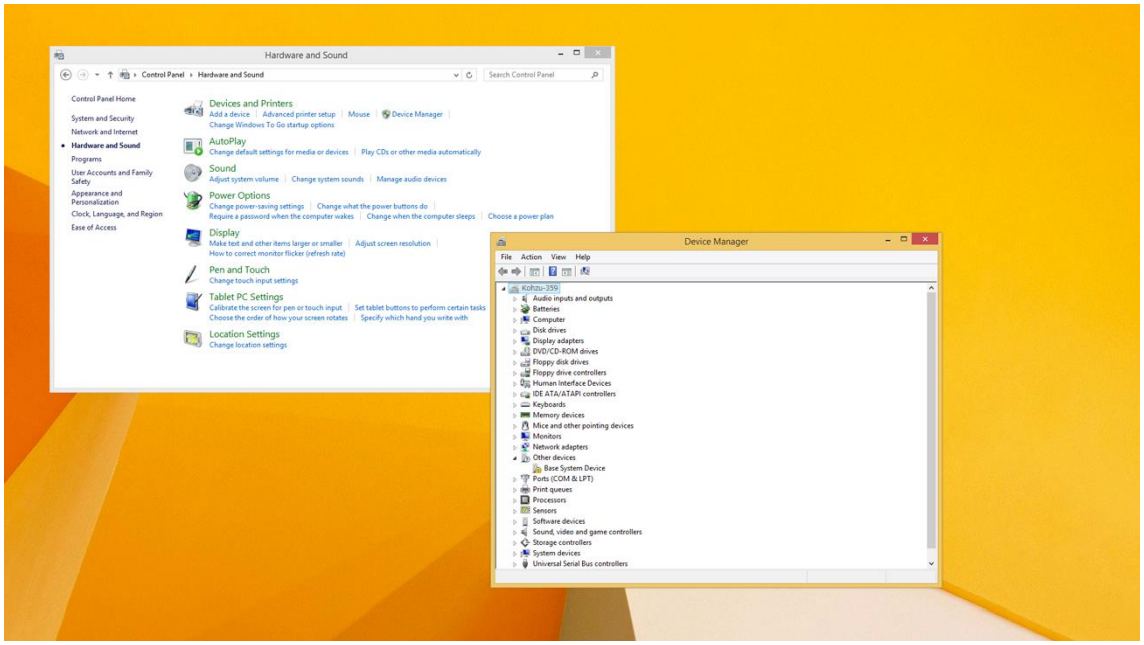
4. Wiring the Controller

4.1. RS-232C

ARIES, CRUX, and old SC series controllers can be connected by means of RS-232C. CRUX can also be connected by using USB (virtual COM port)[10].

Check the port connected with Controller in the device manager in advance.

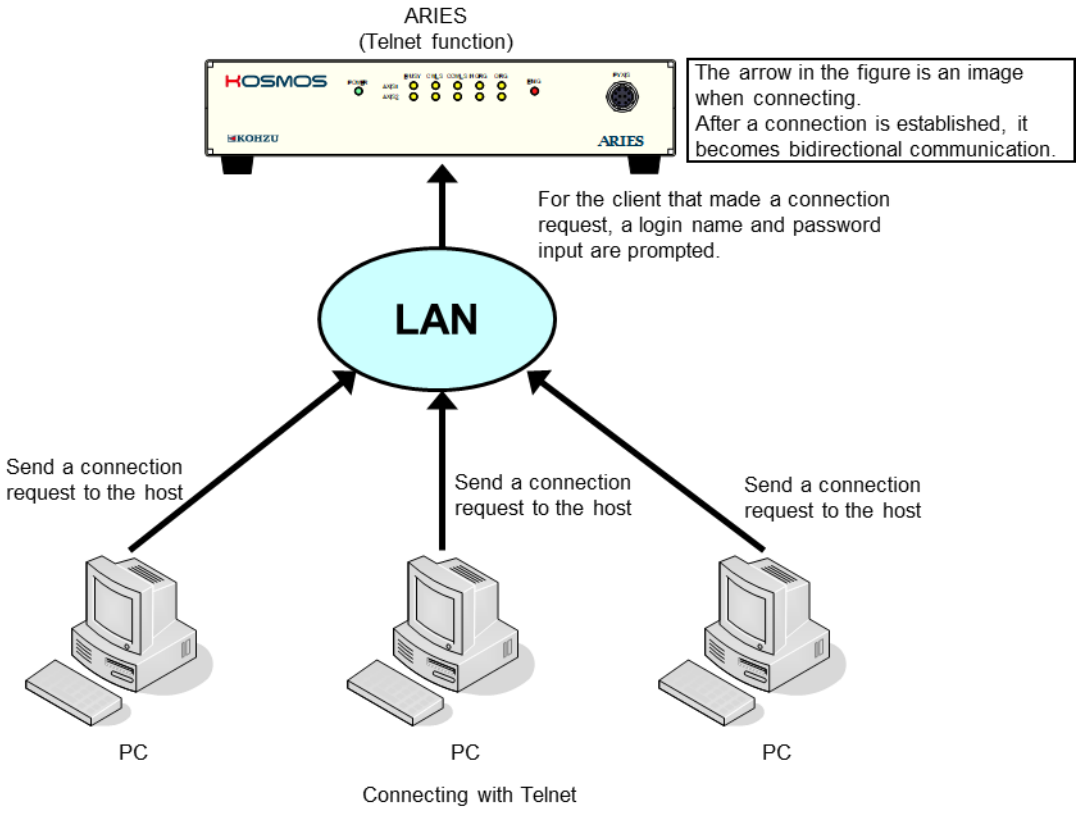
You can find in Control Panel > Hardware and Sound > Device Manager. To start the device manager, you must have administrator privileges.



[10] The USB virtual COM port is only available for CRUX.

4.2. TCP/IP

ARIES can be connected as a TCP/IP client. To connect ARIES as a TCP/IP client, the ARIES TCP/IP address, port number, and others[11] must be preset by means of RS-232C communication and the like. The illustration is an impression of this connection.



Connecting with Telnet

[11] There are two connection methods: necessary parameters are set to TCP/IP, or setting values shown on the next page are used with P2P.

Following are the values set in ARIES as the factory default settings: See the CRUX/LYNX User's Manual for more details.

Function	Default setting	Command	
		Write	Read
Host/Client/Telnet setting	Host (Set value 0)	WHC	RHC
Set IP Address of ARIES	192.168.1.120	WIP	RIP
Set subnet mask	255.0.0.0	WSN	RSN
Set a port number*	12321	WPT	RPT
Client limit setting	No limit (Set value of client No.1 777.0.0.0)	WCL	RCL
Specify the host IP address	192.168.1.102	PIP	RPI
Set the default gateway	Not setting (Set value 999.0.0.0)	WGW	RGW
Set a password (Write only)	KOSMOS	WPS	-
Set a number of application connections	32	WAP	RAP

4.3. Connection to CRUX

4.3.1. Communications Settings of CRUX

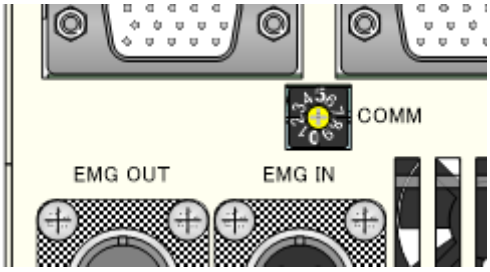
CRUX can set or change communication conditions with the rotary switch (COMM) in the rear panel. Default setting is 4 (Normal command, USB mode).

See the CRUX/CRUX-A User's Manual for more information.

* Settings of RS-232C communication except for speed (baud):

Parity	**NON**
Word length	8bit
Stop bit	1

Rotary switch setting for CRUX communication



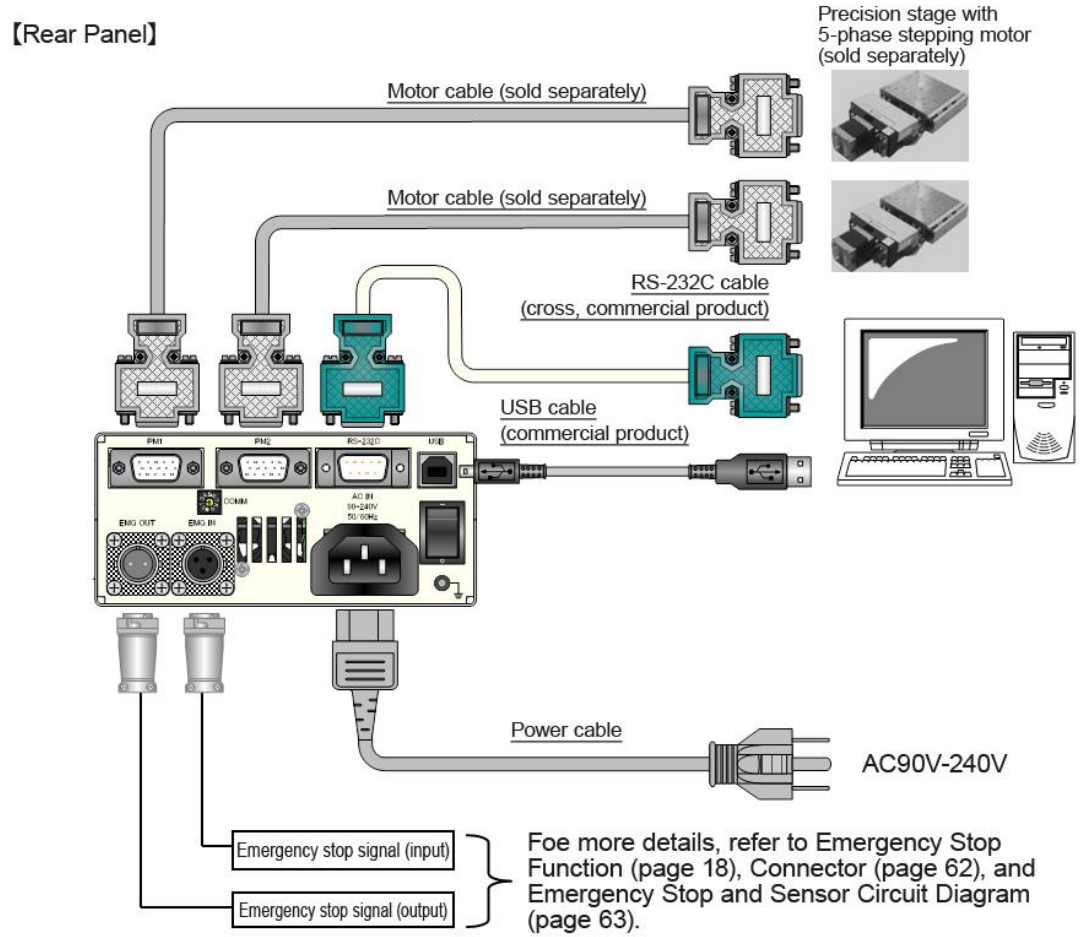
Positions of rotary switch for CRUX communication setting

Communication mode	Communications settings		
	RS-232C speed	USB	
0	38400	*	General Command
1	57600	*	
2	19200	*	
3	9600	*	
4	*	USB	
5	38400	*	Simple command
6	57600	*	
7	19200	*	
8	9600	*	
9	*	USB	

4.3.2. How to Connect Cables on CRUX

When pulling out or inserting at wire connections, make sure the power of main body is OFF.

Connection/connecting wires between CRUX and external equipment are explained.



See the CRUX/CRUX-A User's Manual for more information.

4.4. Connection to ARIES

4.4.1. Communication Settings of ARIES

When configuring communication, make sure the power of main body is OFF.

ARIES can set or change communication conditions with the rotary switch (COMM) in the rear panel. Default setting is Mode 4 (RS-232C 115200baud).

See the CRUX/LYNX User's Manual for more details.

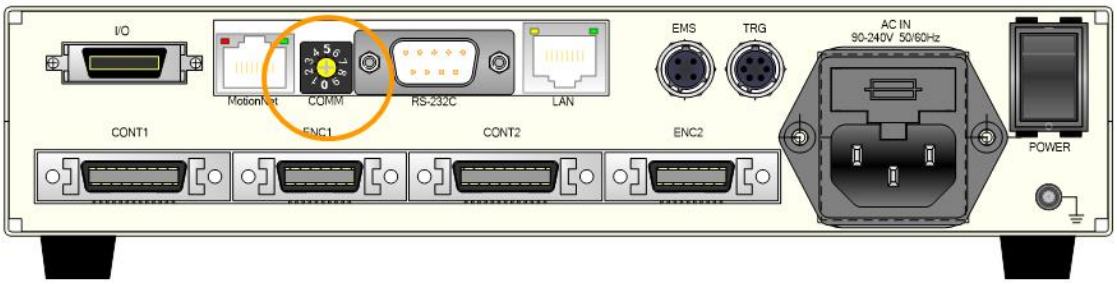
* Settings of RS-232C communication except for speed (baud):

Parity	**NON**
Word length	8bit
Stop bit	1

Position of Rotary Switch for ARIES Communication setting

The rotary switch for communication setting is located as shown below.

【Rear Panel】



Setting of Rotary Switch for ARIES Communication

Settings are as shown in the table below. (Mode 6 to 9 unusable)

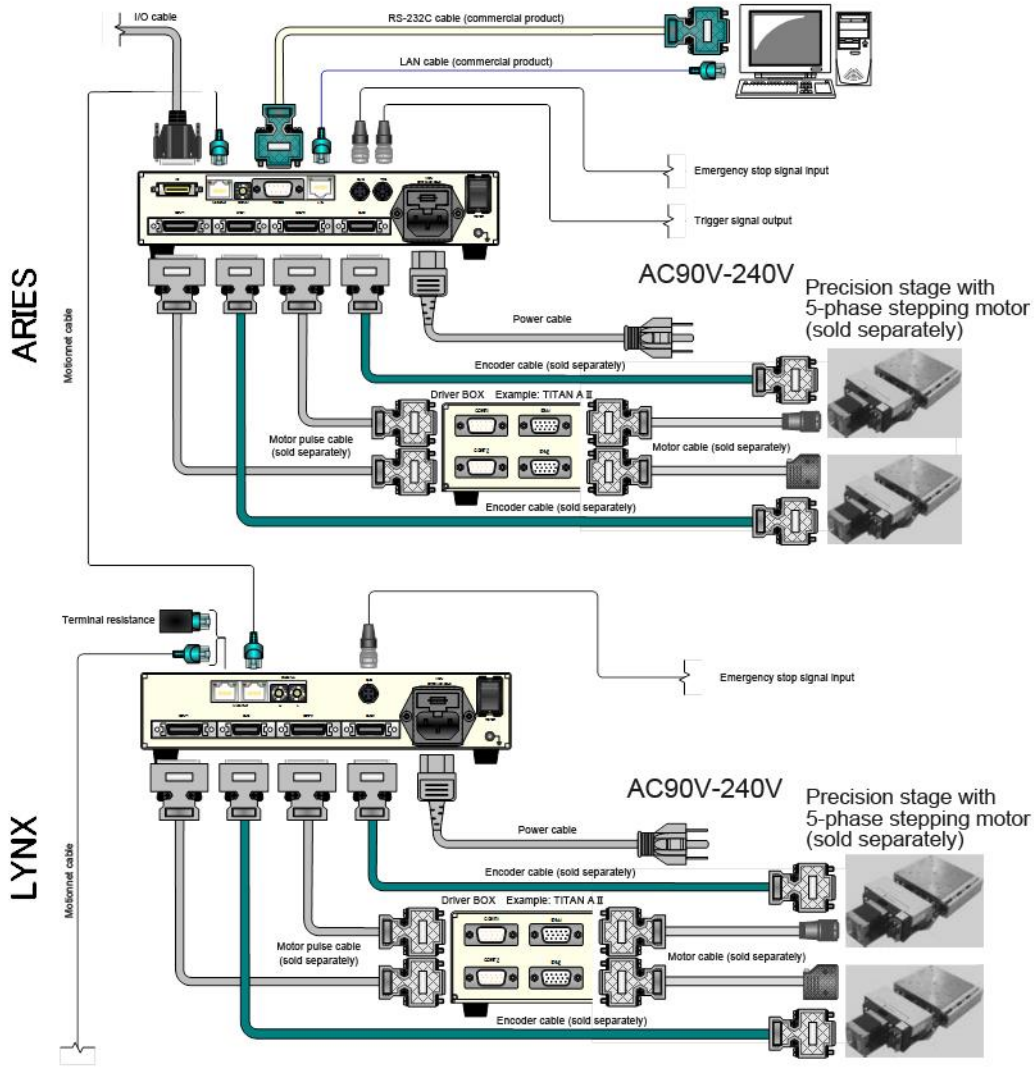
Communication mode	Communications settings	
	RS-232C speed (baud)	LAN
0	9600	*
1	19200	*
2	38400	*
3	57600	*
4	115200	*
5	*	LAN
6	*	*
7	*	*
8	*	*
9	*	*

4.4.2. How to Connect Cables on ARIES

⚠ When pulling out or inserting at wire connections, make sure the power of main body is OFF.

Connection/connecting wires between ARIES and external equipment are explained.

【Rear Panel】



* Do not use a hub between Motionnet® cable connections.

See the ARIES/LYNX User's Manual for more details.

4.5. Connection to SC-210/410

4.5.1. Communication Settings of SC-210/410

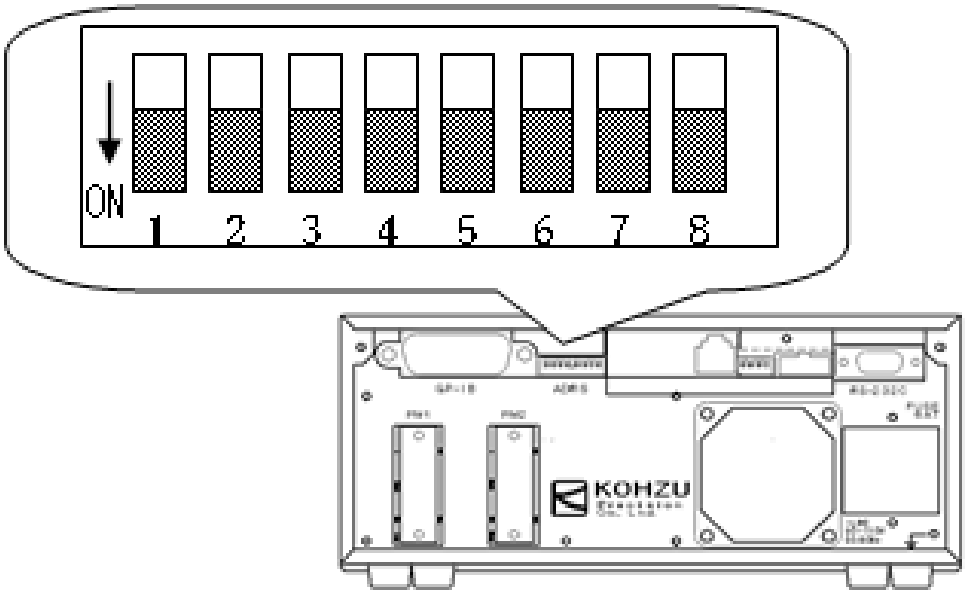
Please power off of your controller in doing communication settings.

In SC-210/410, it can set or change the communication setting by the dip-switches on their rear panel. The default setting is that the baud rate is 38400baud, Non-parity, the word length is 8bit and stop bit is 1.

Please refer to the user's manual of SC-210/410 to get more information.

The position of the rotary switch of SC-410

There is the rotary switch to select communication interface as the following.



About SC-020, 021, 200, 210 and 400, please refer to each user's manual to get the spec information.

Rotary switch setting for communication of SC-210/410

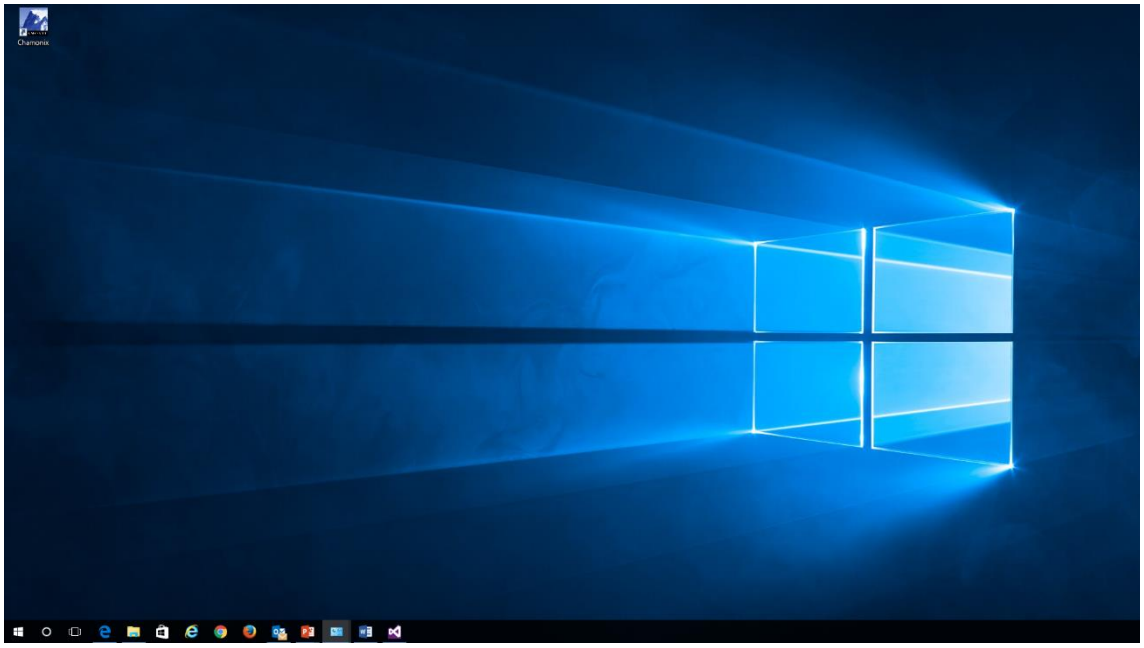
The setting is the followings.

Switch Settings								Comm	RS-232C Settings				GP-IB	
1	2	3	4	5	6	7	8		Baud rate	parity	Word length	Stop bit	delimiter	Adress
OFF	OFF	*	*	*	*	*	OFF	RS	38400	*	*	*	*	*
ON	OFF	*	*	*	*	*	OFF	RS	28800	*	*	*	*	*
OFF	ON	*	*	*	*	*	OFF	RS	19200	*	*	*	*	*
ON	ON	*	*	*	*	*	OFF	RS	9600	*	*	*	*	*

After just install, the baud rate on Chamonix is 115200baud. Since this baud rate cannot use by the SC series, we cannot connect to the controller. Please change the baud rate on the application to available baud rate of the SC controller. The change is described in Selecting COM Port.

5. Getting Started!

5.1. Starting from the Icon on the Desktop



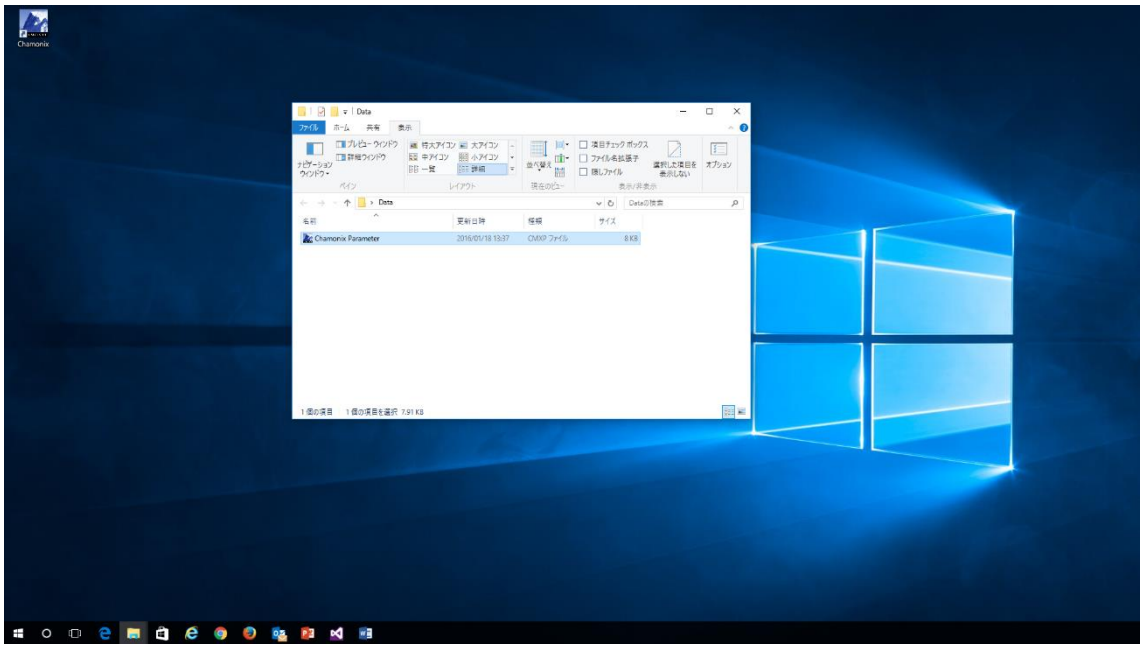
After the installation is successfully complete, the Chamonix icon is generated on the desktop.

Click this icon to start with communication settings that was used last time.

When you start Chamonix for the first time, the default communication settings is used.

By default, RS-232C is selected as the interface.
The COM port number is COM1 and the baud rate is 115200baud.

5.2. Starting from the Stored File



It is possible to start-up from the stored start-up file.

Information like a port number and baud rate is stored in the start-up file.

The extension of the stored start-up file is. cmxp.

5.3. Start-up Window

The screen similar to the one above should be displayed during start-up.

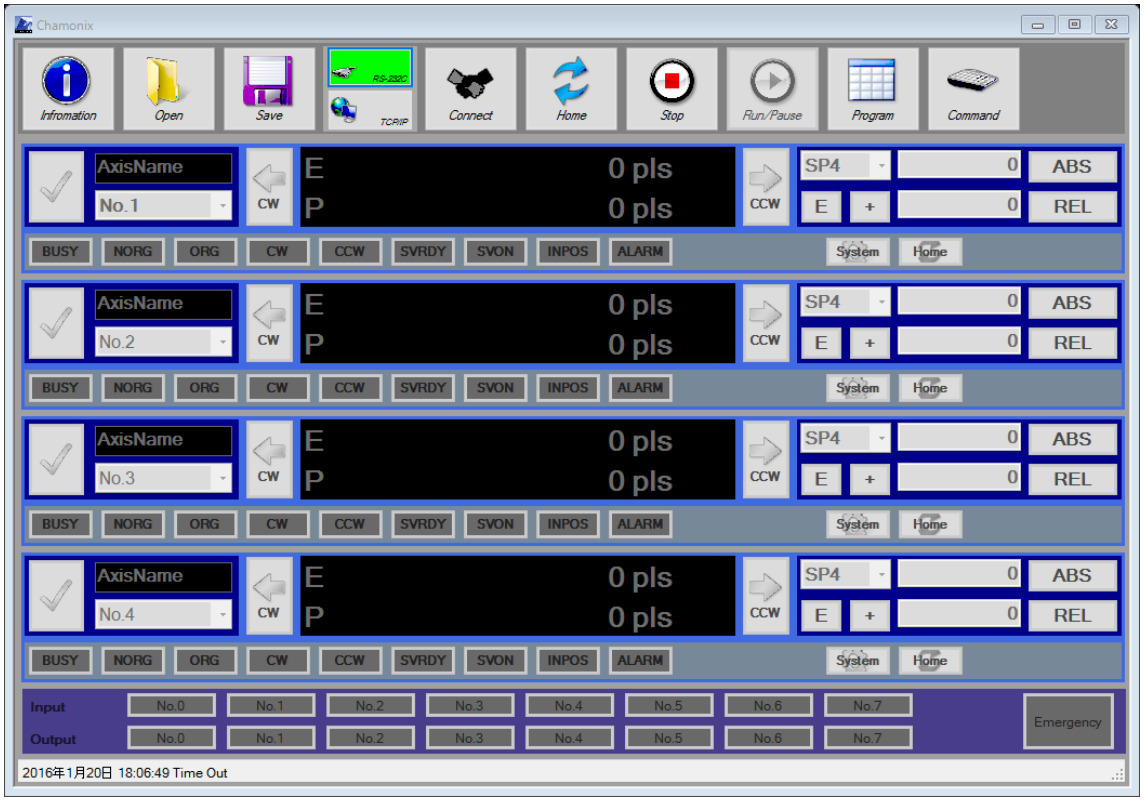
By default, the communication interface is RS-232C, port number is COM1, and baud rate is 115200baud for installation.

If the start-up is not for the first time, it starts with information that was used last time.

This value is stored in the Config file in C:¥ProgramData¥KOHZU¥Chamonix [12].

[12] The file format uses XML. Editing this file may cause a start-up failure about Chamonix.

5.4. Starting Up for the First Time



Chamonix should display the screen like the illustration when starting for the first time by default, the communication interface is RS-232C, the COM port number is COM1, and the baud rate is 115200 baud.

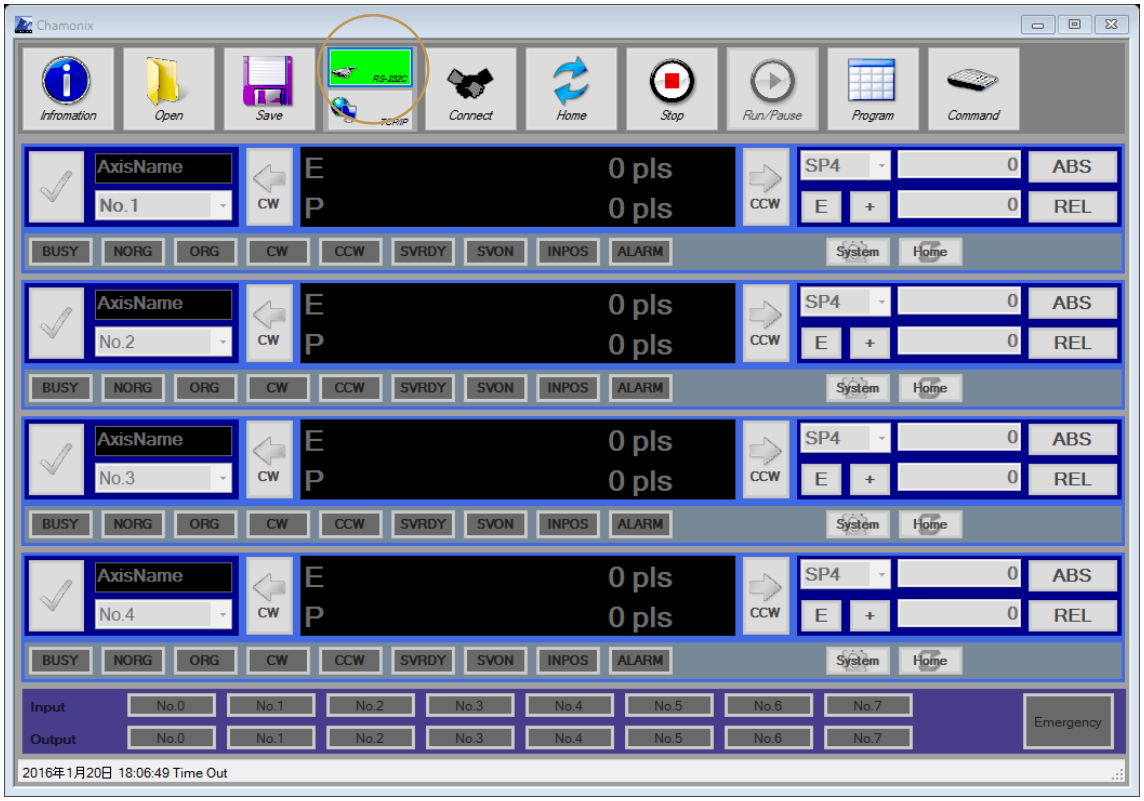
TCP/IP connects to the controller as a client, and the connected controller IP address is 192.168.1.120 and port is 12321.

If settings to connect with a controller are different from the default values, the controller will not be connected when Chamonix start up. Each character appears in monochrome when Chamonix cannot connect with the controller.

6. Connecting the Controller

6.1. Connecting the Controller Using RS-232C

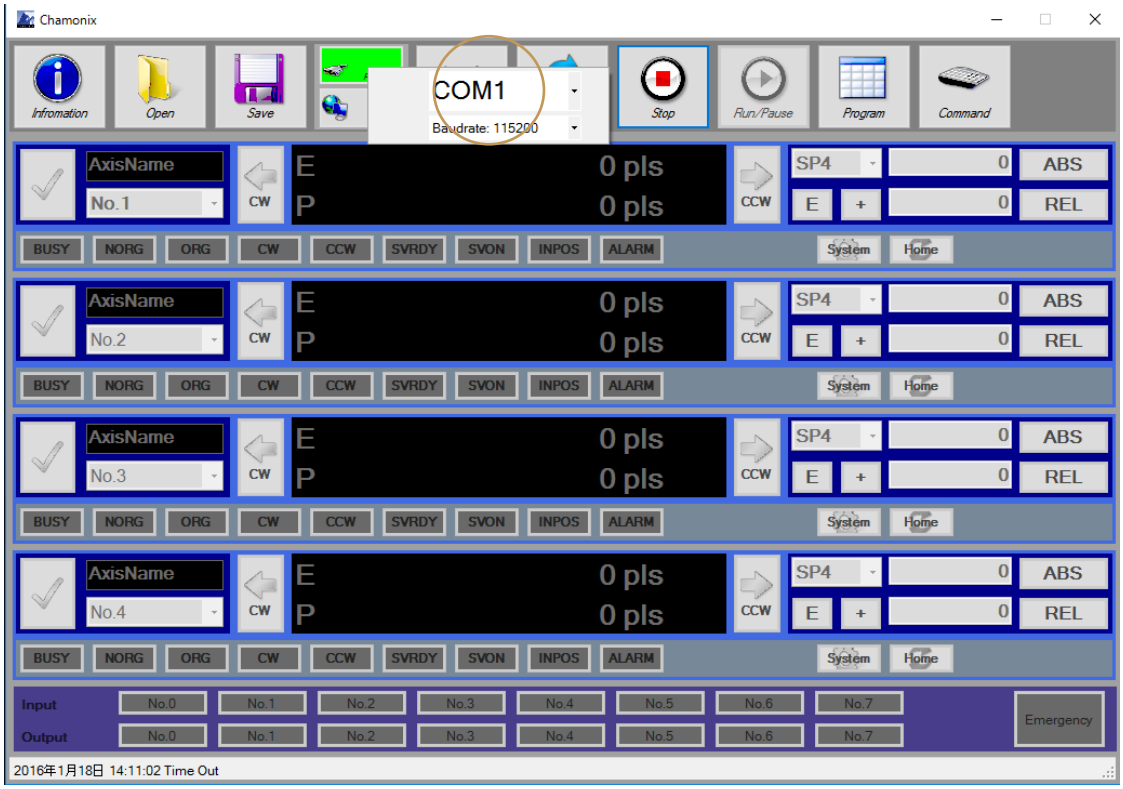
6.1.1. Selecting RS-232C



Press the RS-232 button to select serial communication through RS-232C as a communication interface to the controller. The selected interface illuminates like the illustration.

Also, RS-232C should be selected like this when CRUX is connected by using USB. Connect as a virtual COM port (CDC class). See the operation manual for CRUX for more details.

6.1.2. Selecting COM Port



If a COM port is not selected, control display of drive axes does not appear as shown above.

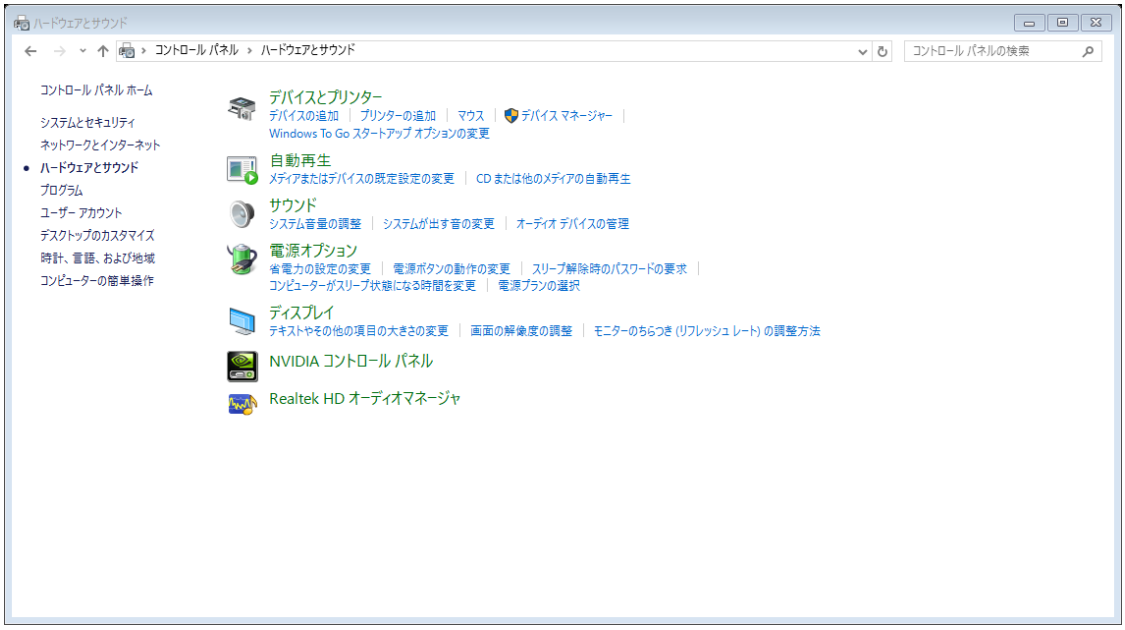
Right-click on the RS-232C button to display a menu and set the COM port.

The baud rate of RS-232C communication for our controller is 115200 as the factory default setting. See the operation manual for each controller for more details.

Left-click on the Connect button to perform connection check and initialization.

If this application is closed after a COM port is selected and connection check is successfully complete, the next start-up is performed by using set values.

6.1.3. How to Check the COM PORT Number

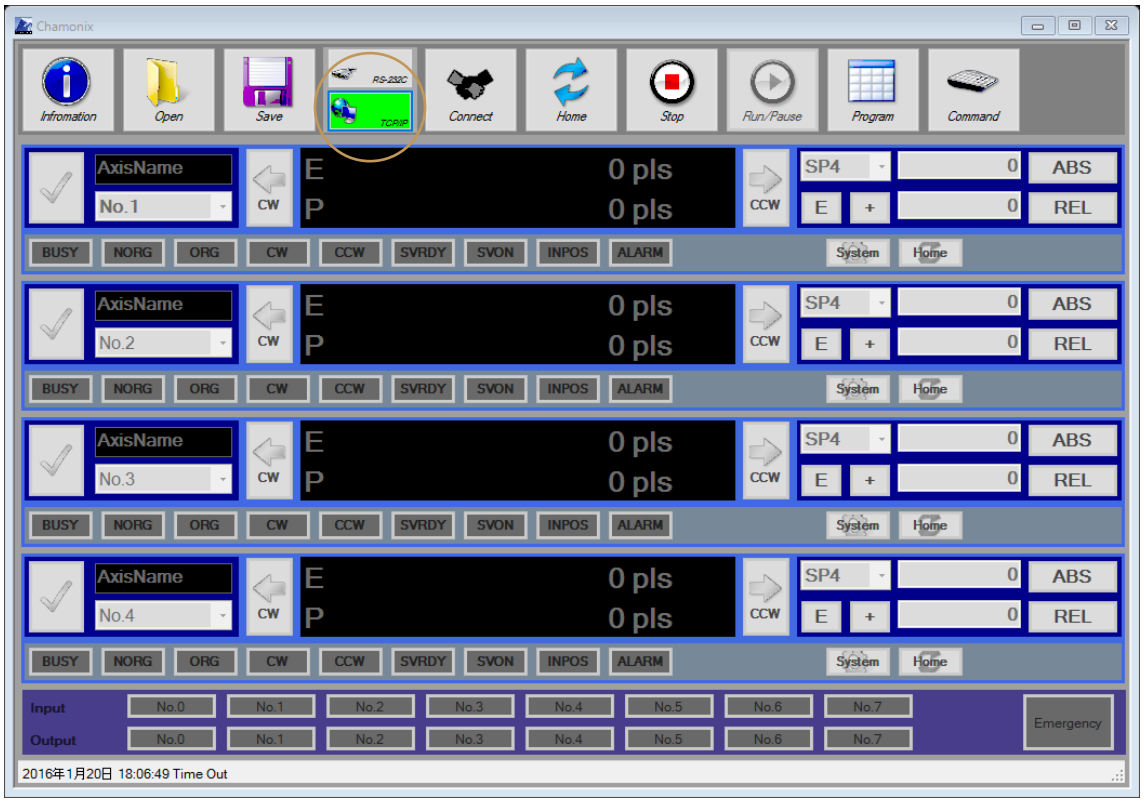


Check the port connecting controller in the Device Manager.

You can find in Control Panel > Hardware and Sound > Device Manager. To start the Device Manager, you must have administrator privileges.

6.2. Connection Using TCP/IP

6.2.1. Selecting TCP/IP (ARIES Only)



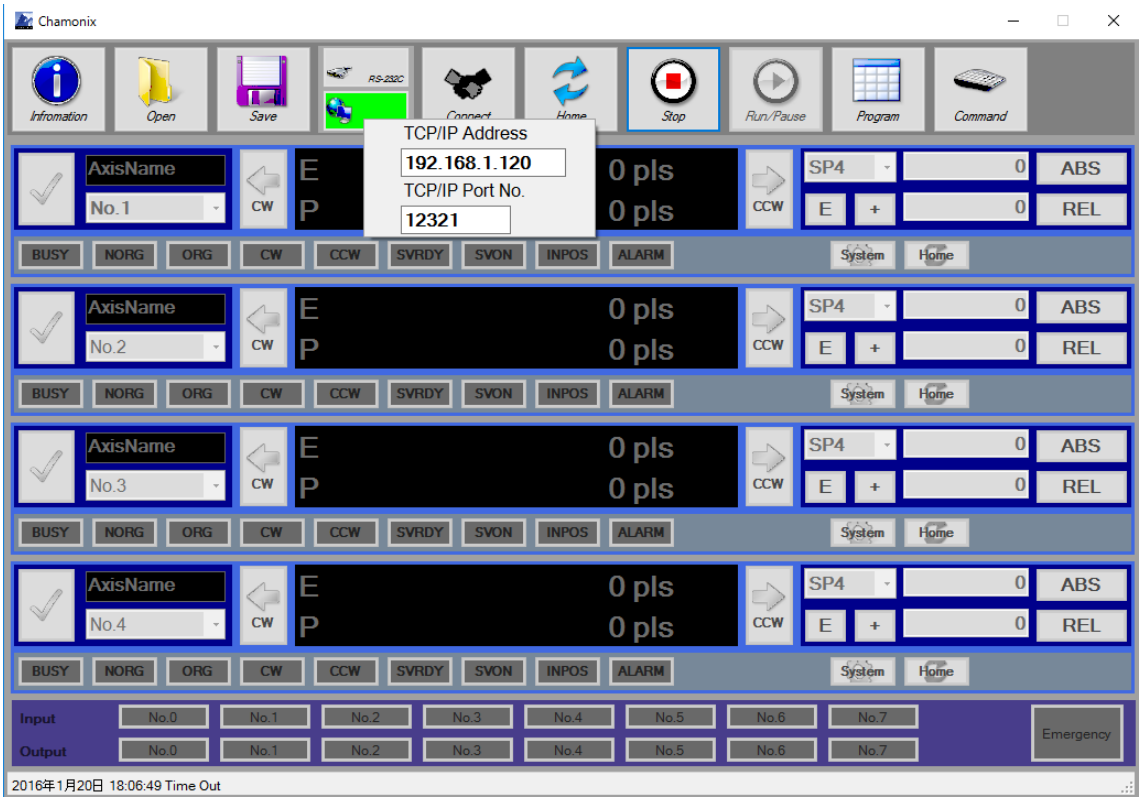
Press the TCP/IP button to select connection to the controller as a TCP/IP client. The selected communication interface illuminates as the illustration.

ARIES is the only controller that can select TCP/IP as the communication interface. Do not select it for the other controllers.

For "Host/Client/Telnet setting" of ARIES, set the host. Host is selected as the factory default setting of ARIES. To configure the ARIES settings, you must enter and send the sending command from CommandControl (see "15.Entering Remote Commands Directly“) and others directly while the controller is connected.

For more details on the sending command, see the operation manual for ARIES.

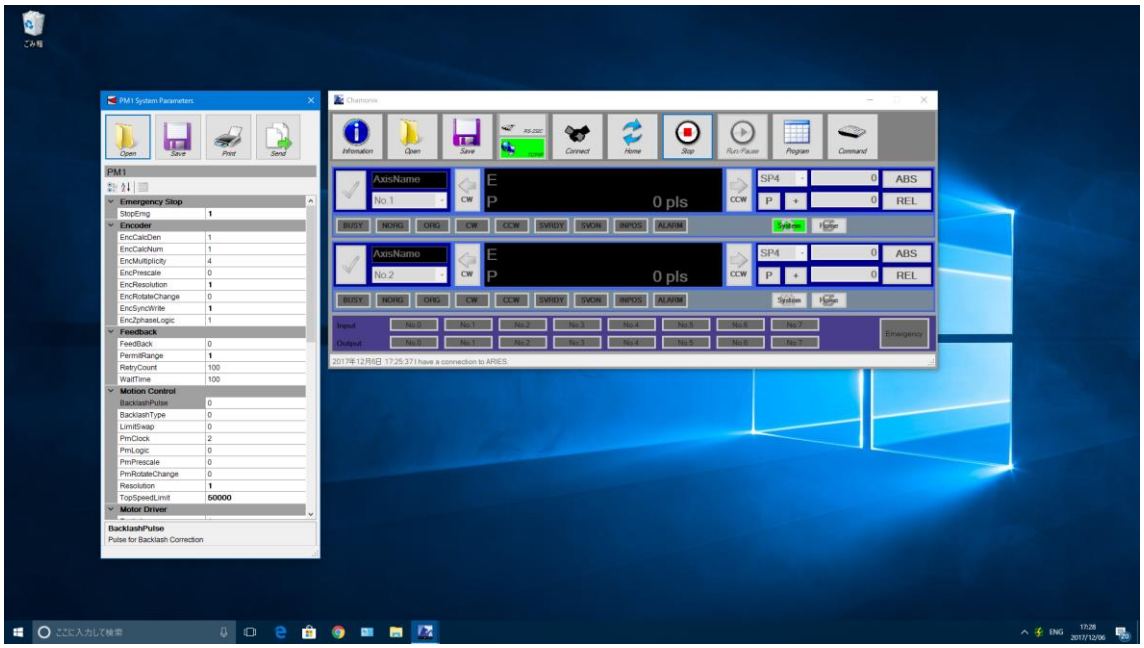
6.2.2. Specifying Connection Settings of Chamonix



Right-click the TCP/IP button to open a menu like the illustration. Enter the IP address and the port number of the controller to be connected to Chamonix.

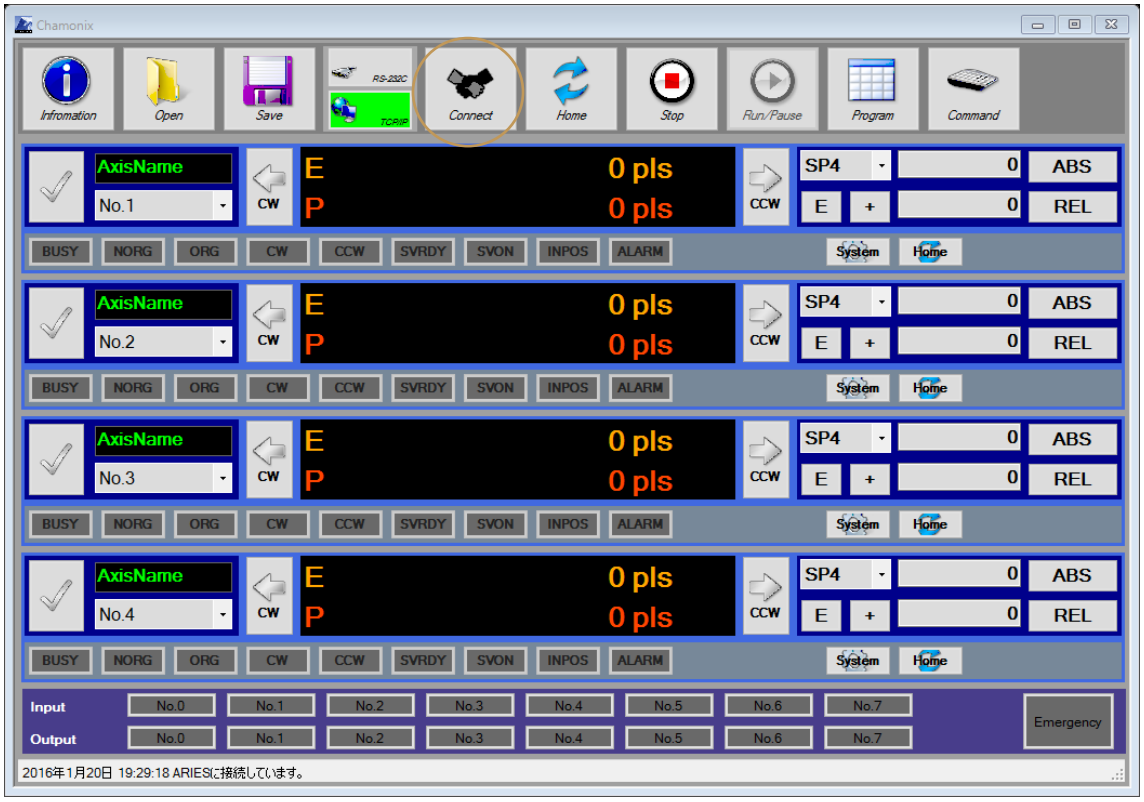
The IP address is 192.168.1.120 and the port number is 12321 as the factory default setting of ARIES. When you change the ARIES setting values in your environment, change the ARIES parameters by means of P2P or RS-232C and then restart Chamonix to make these settings.

6.2.3. Changing the IP Address of ARIES



After connecting to ARIES, the IP address, port number, and subnet mask of ARIES can be changed. Press any shown System button to open the System Parameters window. Set the TCP/IP items and press the Send button to send to the controller. (Also see "13.Changing System Parameters".)

6.3. Connecting the Controller



Press the Connect button to perform connection check and initialization.

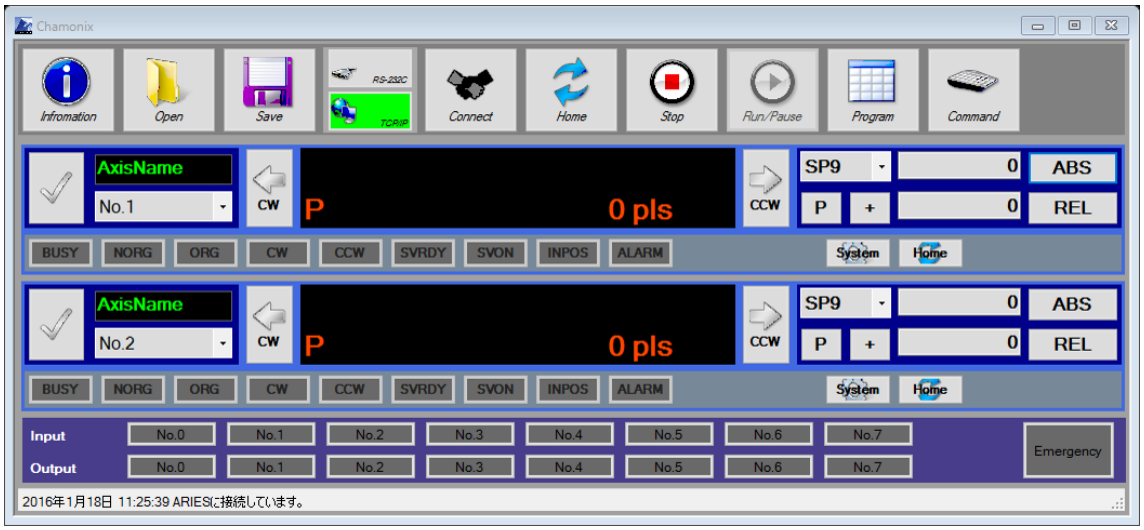
If the connection settings are successfully read during start-up, the system performs a connection check and initialization automatically.

It may take five or more seconds for start-up in a case where a connection check and initialization require time.

If the connection is established, control displays (StageControl) of drive axes will appear as shown above. Control displays of drive axes appears up to for 4 axes[13].

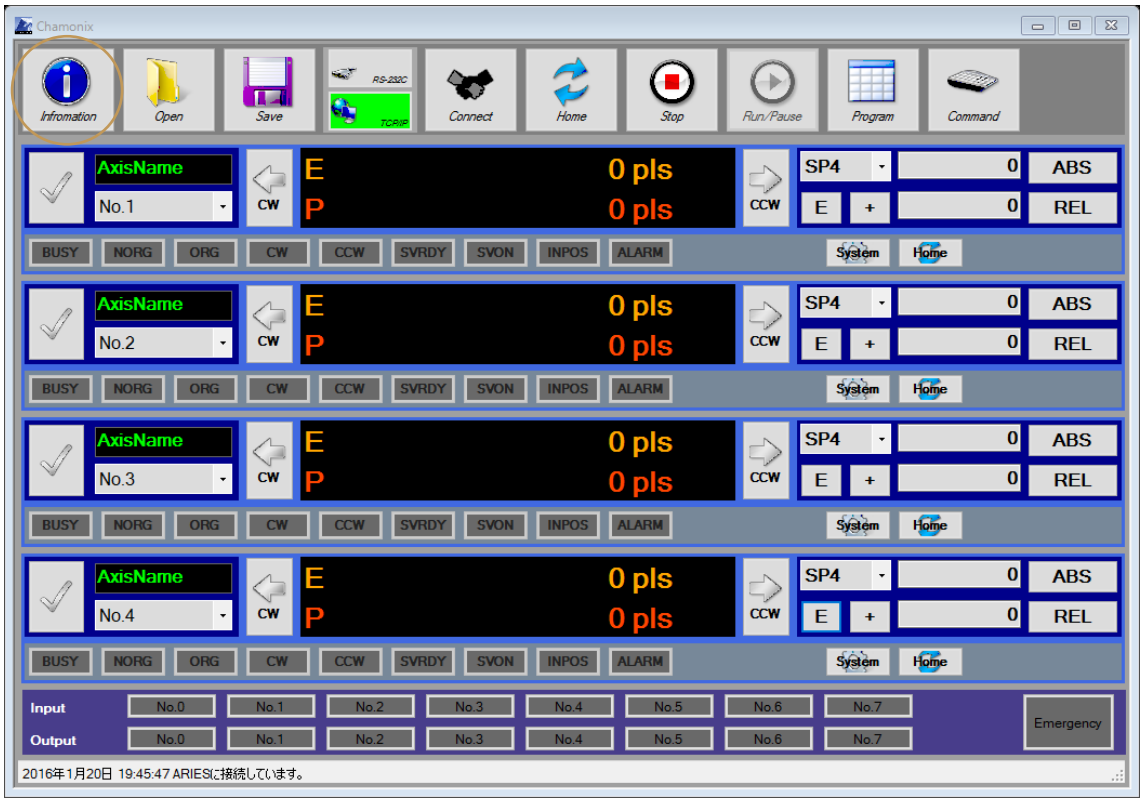
[13] Like cases using CRUX or not connecting slave controller LYNX to ARIES, the displays is for 2 axes if the total number of axes is 2.

Like cases using CRUX or not connecting slave controller LYNX to ARIES, the displays is for 2 axes if the total number of axes is 2.



7. Checking Version Information

7.1. Checking Version Information about This Application



Click the Information button to check version information[14] as shown on the next page.

[14] Information such as the version number may differ from the actual application.

About Chamonix
Chamonix
Ver. 1.2.1.25
Copyright © 2017 Kohzu Precision Co., Ltd.
神津精機株式会社(Kohzu Precision Co., Ltd.)
CHAMONIX
Chamonix is Windows based application. This application is protected by copylight laws in Japan or foreign countries and international treaties. This application requires the .NET Framework 4.5.2 or higher to execute.
Version of StageControl.dll is 1.5.2.7.
Version of SerialControl.dll is 1.2.6.5.
Version of SystemParam.dll is 1.1.2.4.
Version of SystemParamE.dll is 1.1.2.3.
Version of TCP_ClientControl.dll is 1.0.0.1.
OK

8. Returning to Origin

8.1. What is Returning to Origin?

"Retuning to Origin" is operation that moves near a specified sensor by using the speed table according to a type to return to origin set in the controller, moves with scan speed until the origin is detected, and then stops.

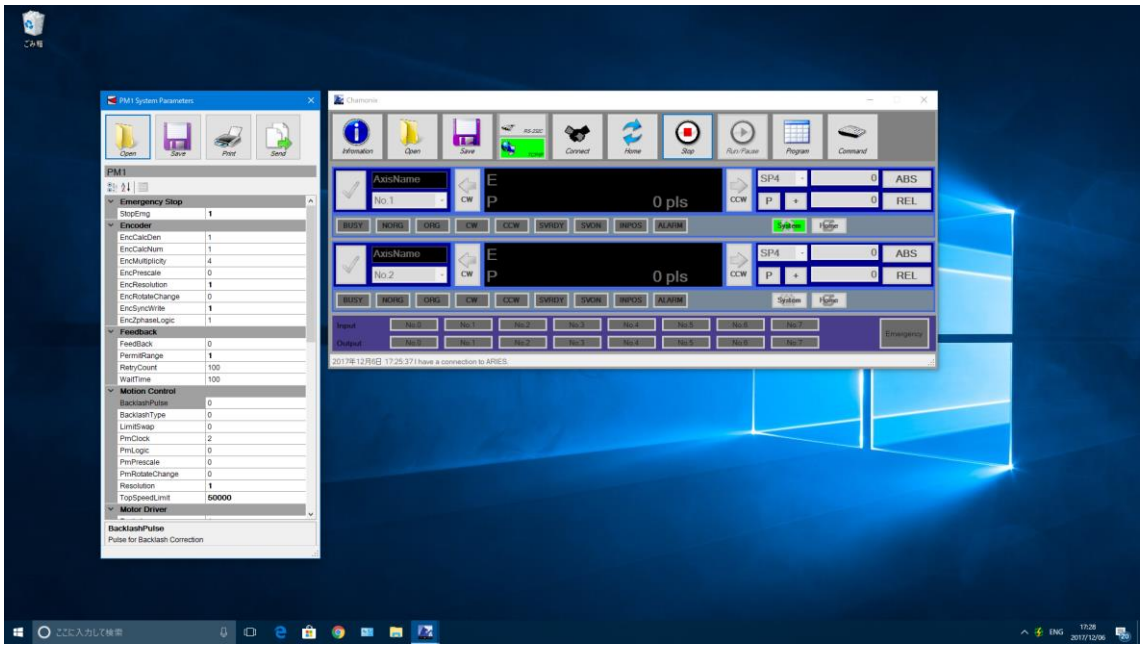
The types to return to origin available for controllers depends on individual controllers. See Types to Return to Origin for each operation manual.

All the strokes may drive when returning to origin.

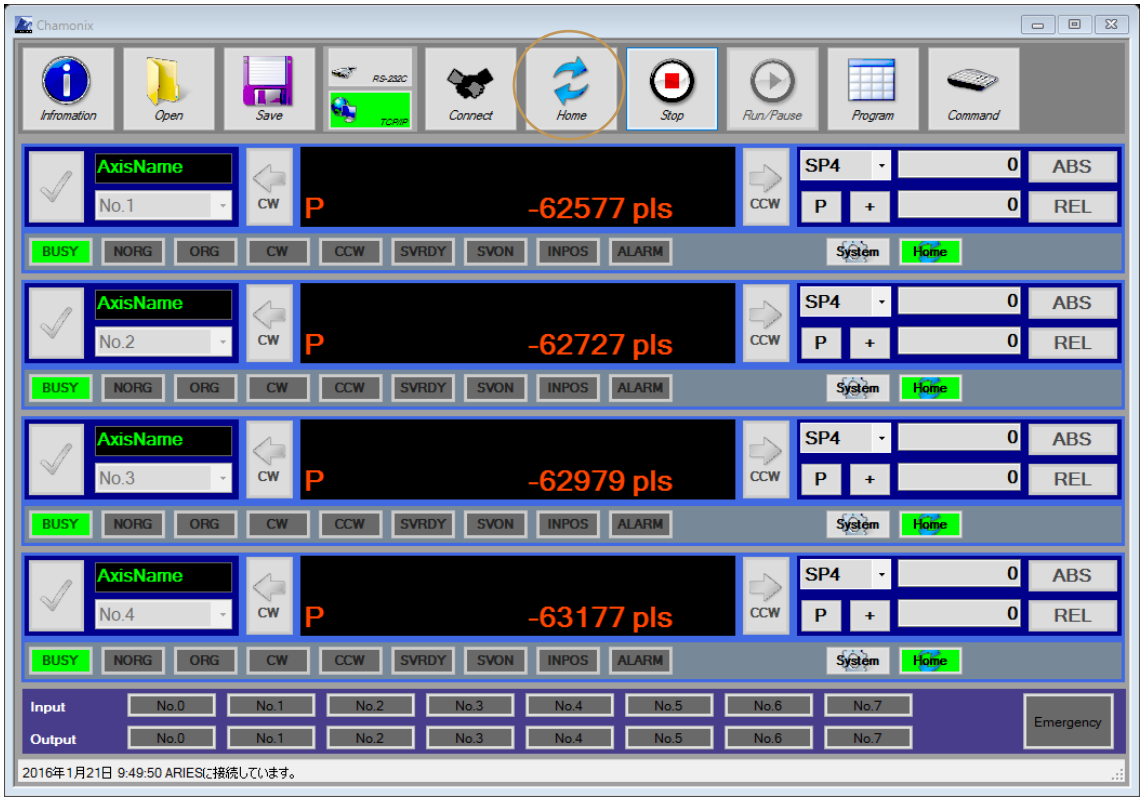
Pay careful attention to safety and perform the procedures.

8.2. Changing the Type to Return to Origin

You can set the type to return to origin of each axis by using the ORG Type property in the System Parameters window (also see "13.Changing System Parameters").
For more details on the type to return to origin, see Returning to Origin for the operation manual of the controller.



8.3. Returning to Origin for All Axes



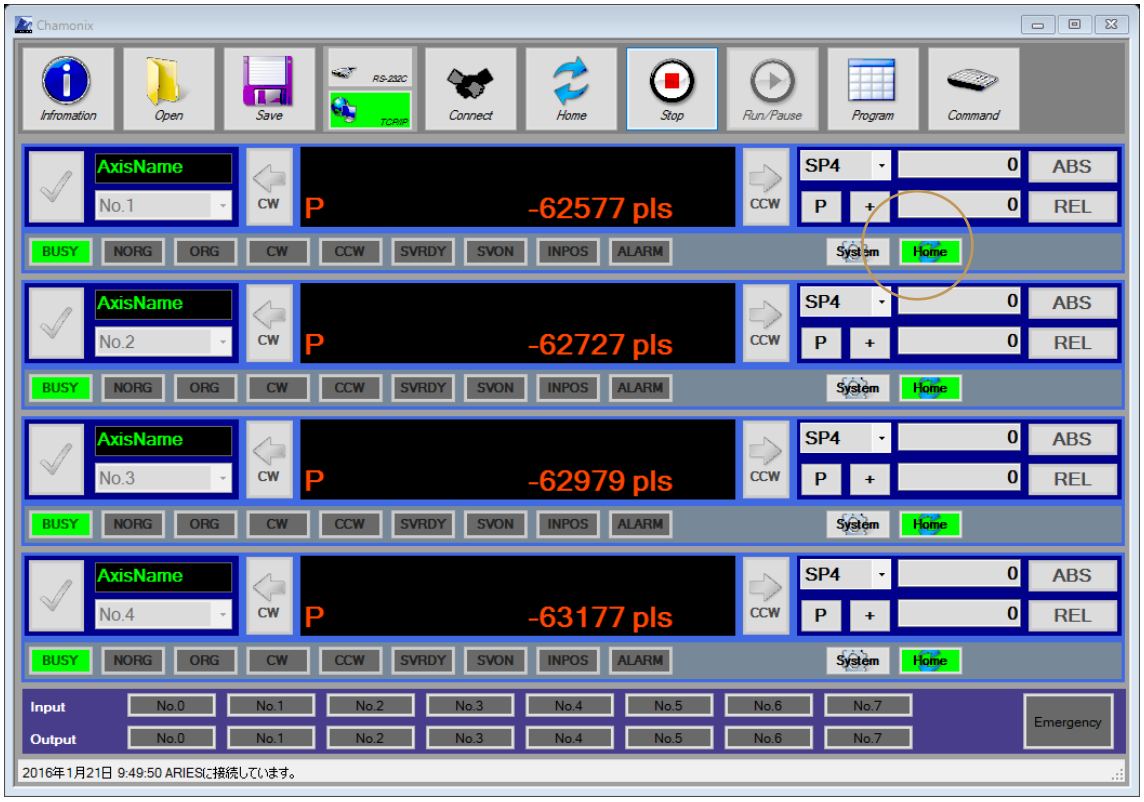
When you press Home Positions, all axes controlled by the controller return to origin.

Each axis returns to origin independently without consideration for movement of other axes. Pay great care to their respective interferences.

When interference of axes could arise, return to origin for single axis as described in the next section. See a manual for your controller for patters of returning to origin. A type to return to origin is subject to the setting value of the system parameter of the controller. Setting of system parameters is discussed in detail later.

Our company assumes no responsibility for a fault from axes interference caused by returning to origin for all axes.

8.4. Status in Returning to Origin



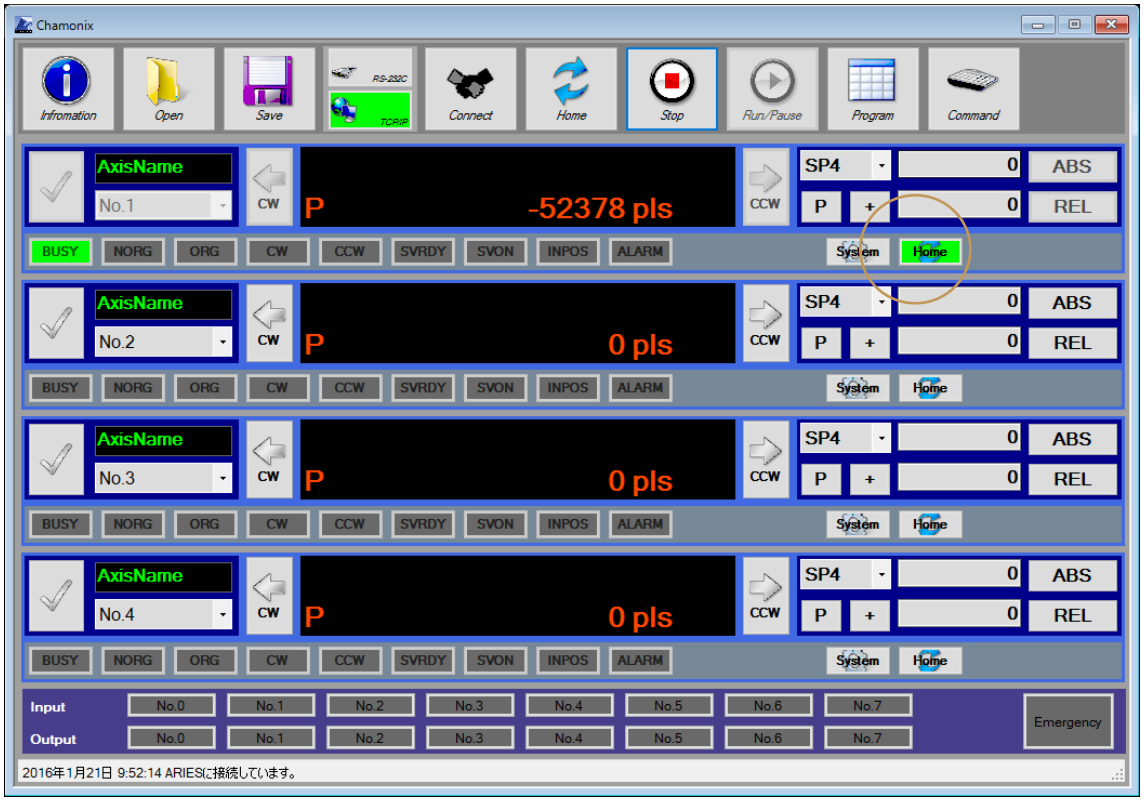
The Home button of each axis illuminates while returning to origin is performing.

The Home button goes out when retuning to origin is complete.

Click an illuminating Home button to stop returning to origin for the only axis.

The BUSY indicators also illuminate during the drive.

8.5. Returning to Origin for Single Axis



You can perform retuning to origin for only a specified axis by clicking the Recovery button of each axis.

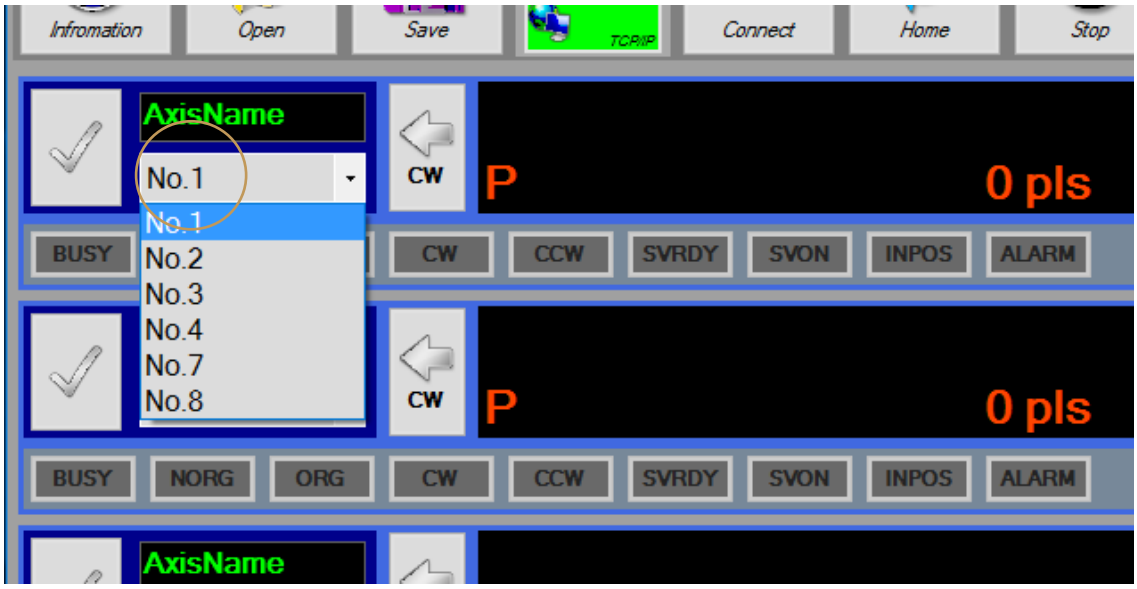
When interference may arise among axes, returning to originfor each axis one by one is recommended.

Press an illuminating button to stop.

Our company assumes no responsibility for a fault from axes interference caused by returning to origin for all axes.

9. Moving Stages.

9.1. Selecting a Axis to Control

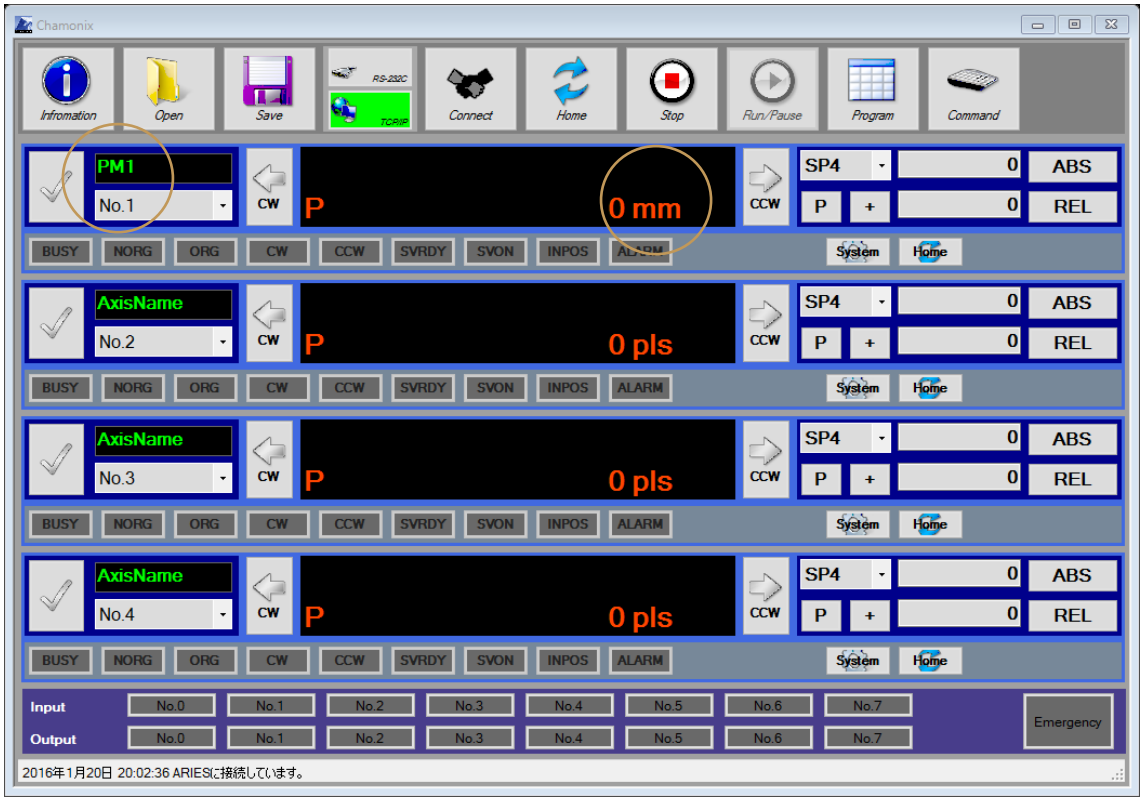


You can use a combo box to select an axis to control.
You can also change an axis to control with GUI-based display during drive.

The display is up to for four axes, and the control is up to for 32 axes.

For CRUX, the control is up to for two axes.
For ARIES, up to 32 axes can be controlled by adding LYNX.

9.2. Setting Axis Names

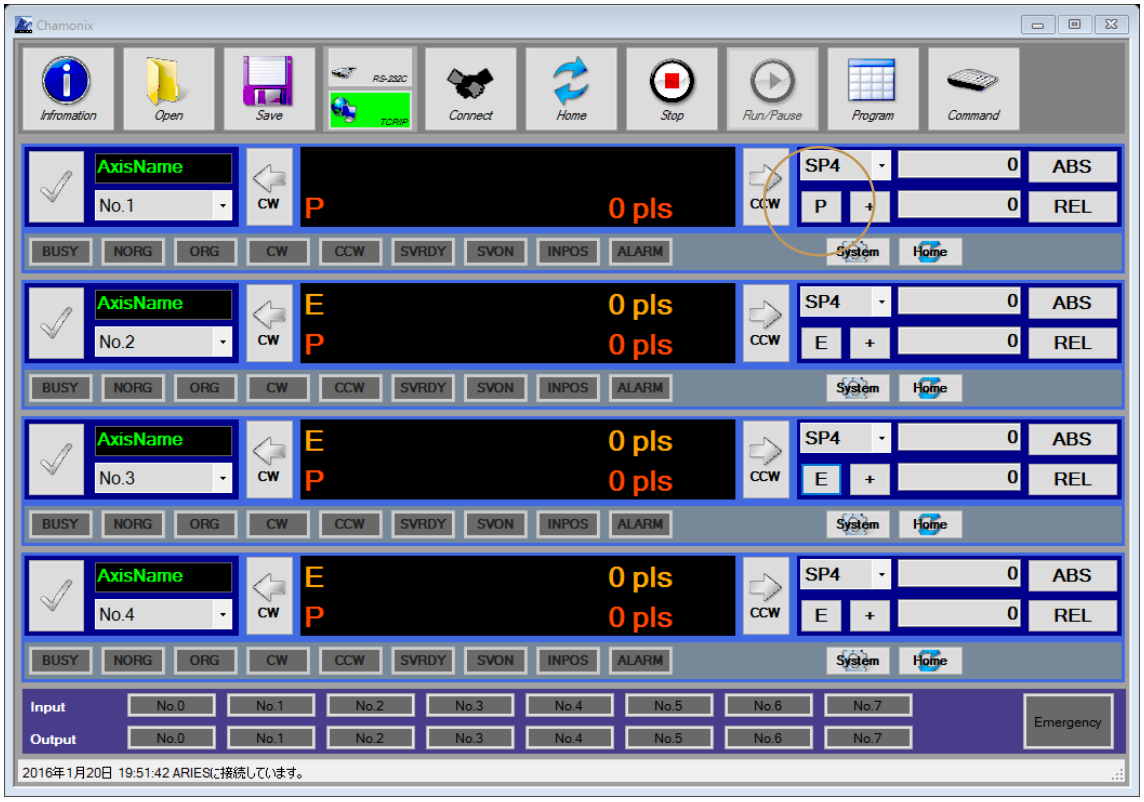


You can enter an axis name. You can also change an axis unit. Language that can be entered in Windows is available. The character code is UTF-8.

This value is stored in the Config file in C:¥Program Files (x86)[15]¥Common Files¥KOHZU¥Chamonix.

[15] For 32-bit OS, the file is in C:¥Program Files ¥Common Files¥KOHZU¥Chamonix.

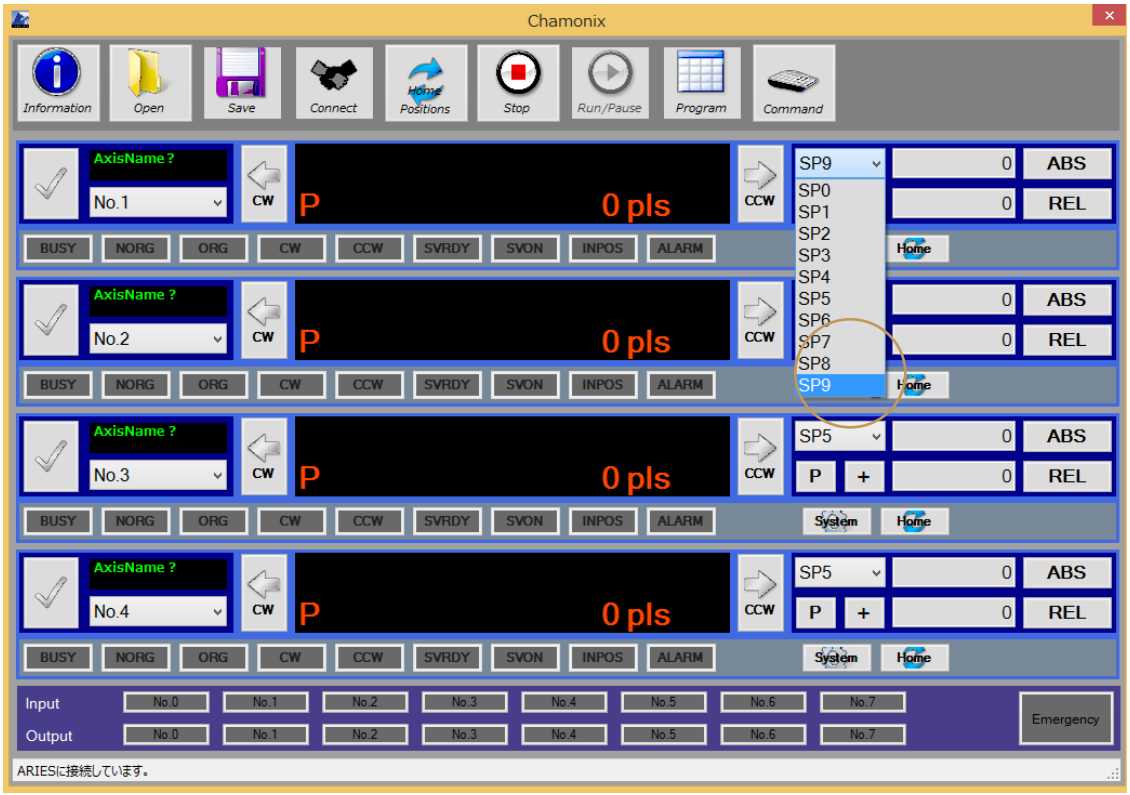
9.3. Switching the Current Position Display



Press the P button to switch to "E" and to display a current position of the encoder.

Press the E button to switch to "P" and to display a current position of the motor pulse.

9.4. Selecting Speed Table

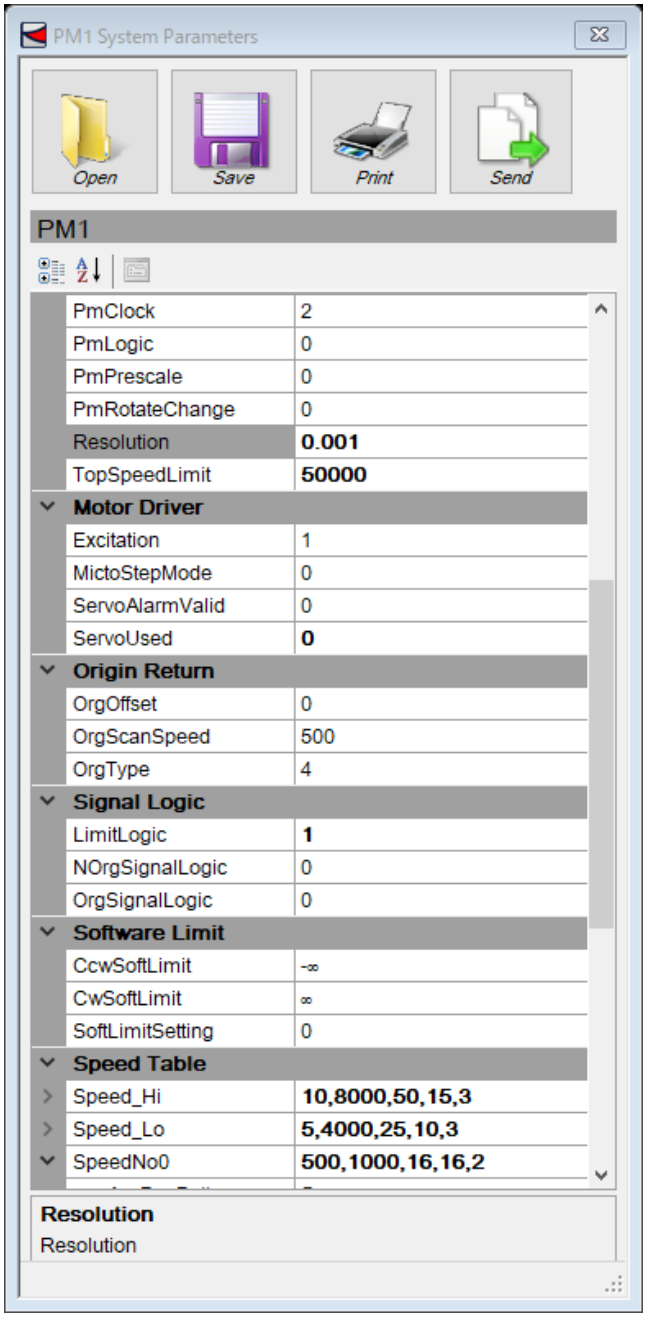


You can change the speed table of each axis from a combo box.

The setting value of a speed table can be set by changing system parameters. Changing system parameters is discussed in “Changing System Parameters”.

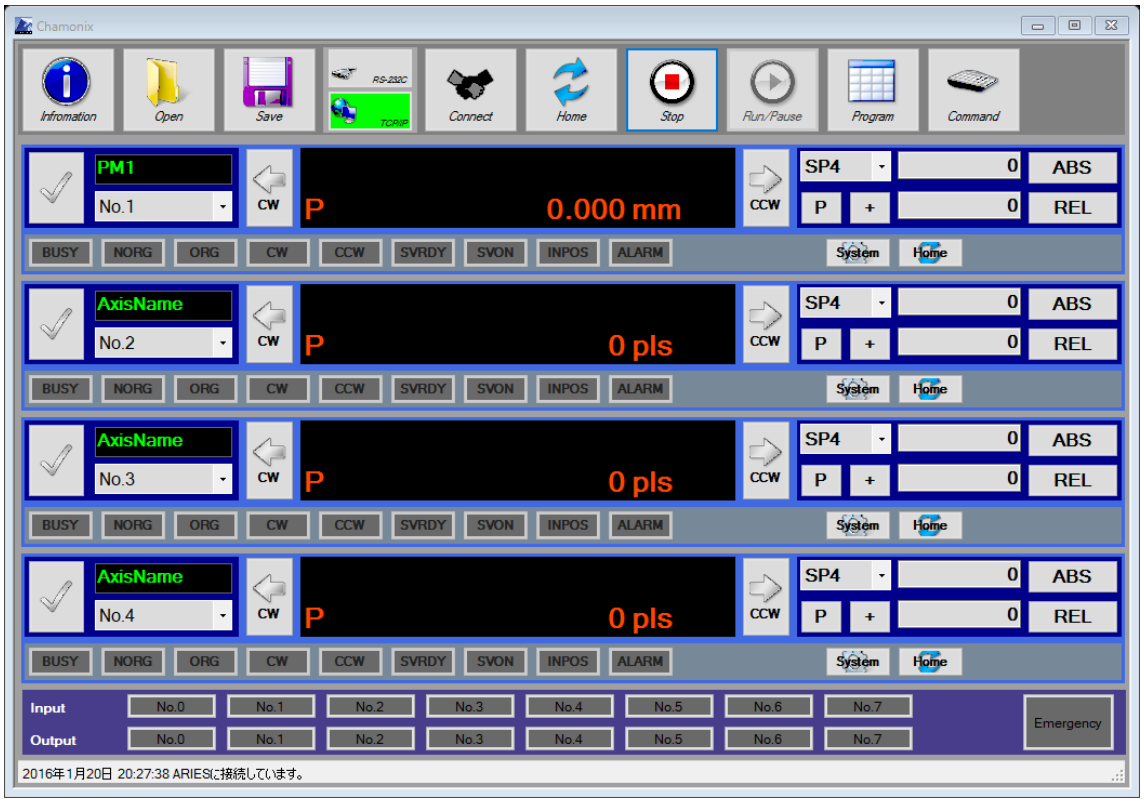
Each parameter of a speed table depends on types of controllers. See the operation manual for each controller for more information.

9.5. Setting Resolution of Axes

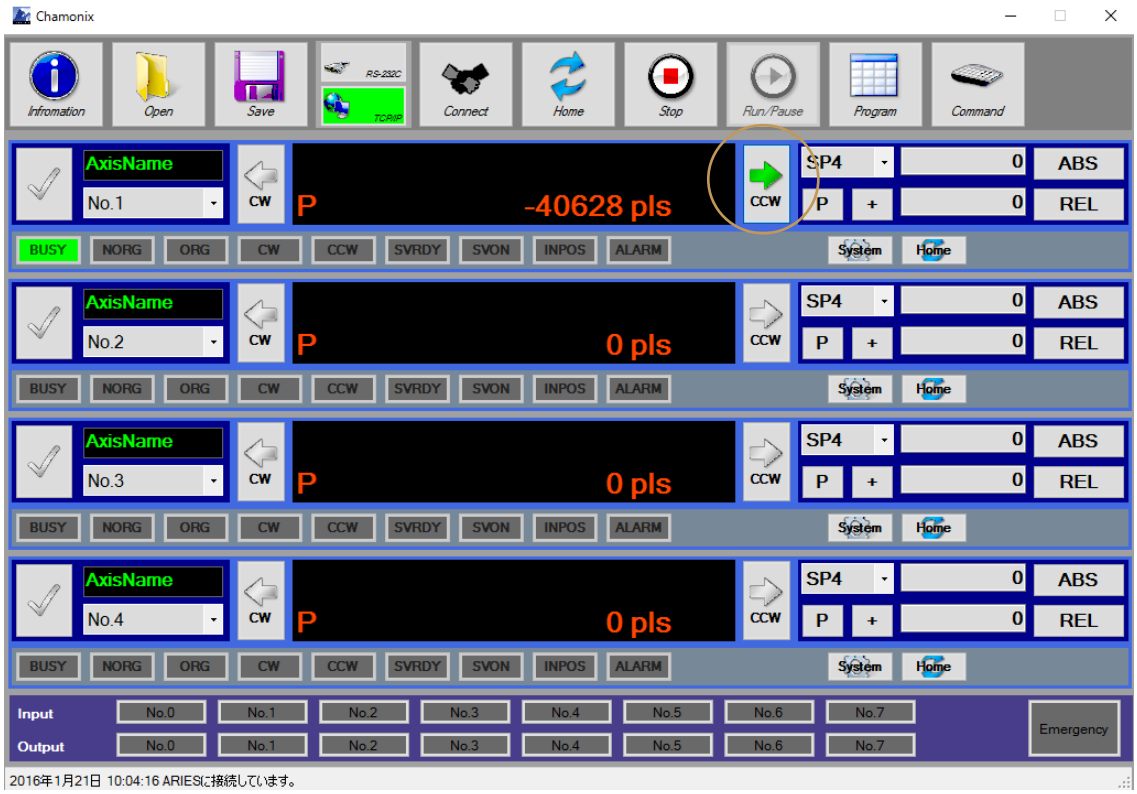


Resolution of axes can be set in the System Parameters window. Enter values per pulse in "Resolution" for the resolution of the motor pulse and "EncResolution" for the resolution of the encoder pulse. (Also see "13.13Changing System Parameters".)

The main window reflects the resolution value that has been set.



9.6. Manual Movement

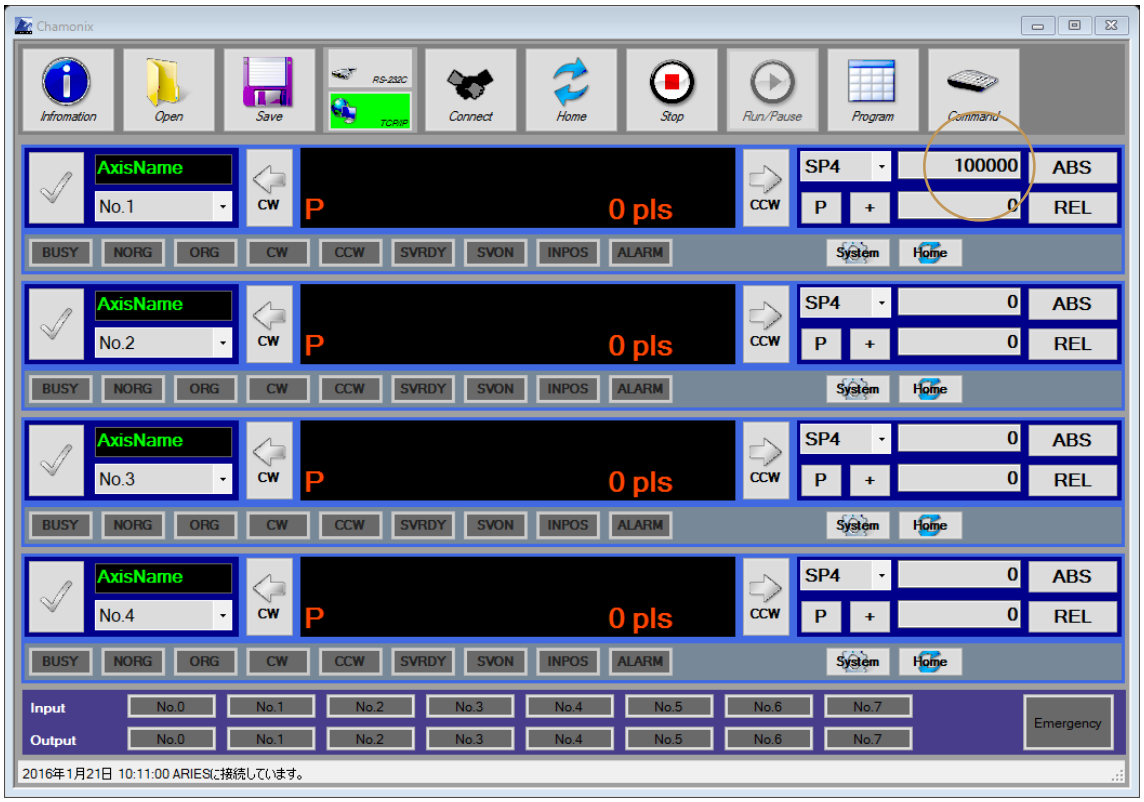


Press the CW button of each axis to move to the CW direction during the press.

Press the CCW button of each axis to move to the CCW direction during the press.

10. Deciding Position

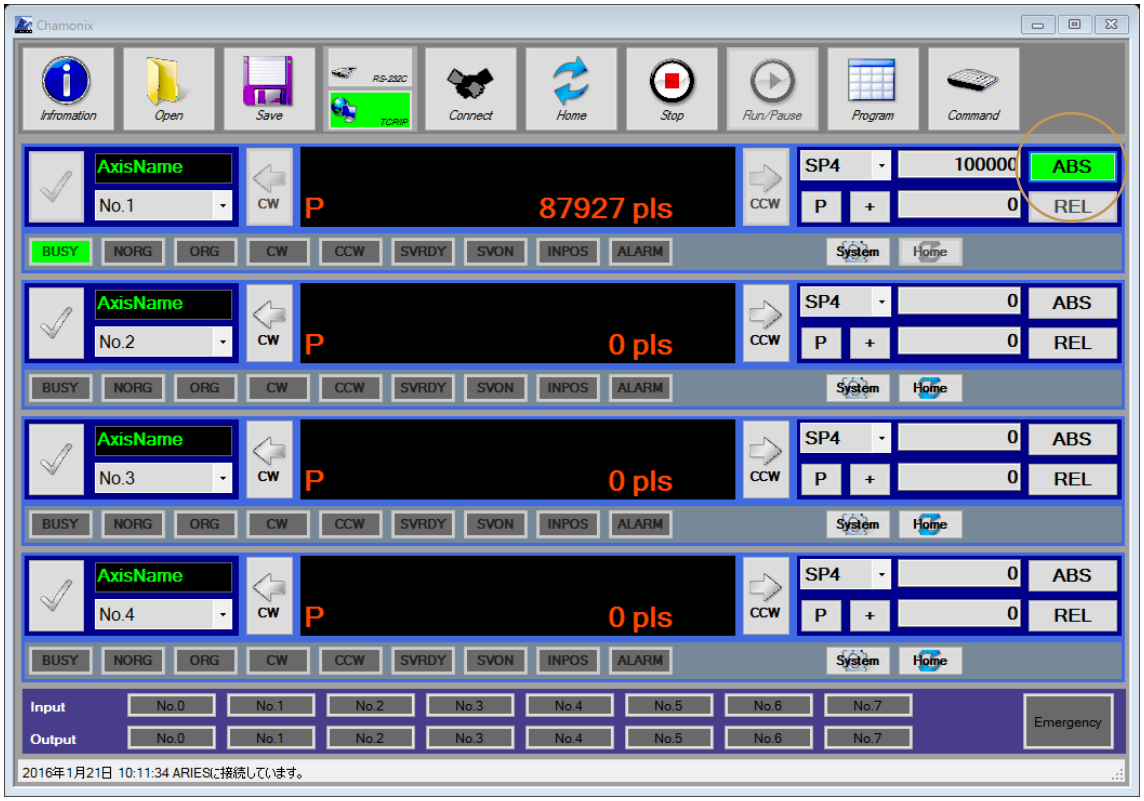
10.1. Setting of Deciding Position with Absolute Position Management



Enter an absolute position of its destination in this text box.

If a conversion value of control resolution is set in the system parameter, enter the conversion value. By default, setting for input uses motor pulse.

10.2. Executing of Deciding Position with Absolute Position Management

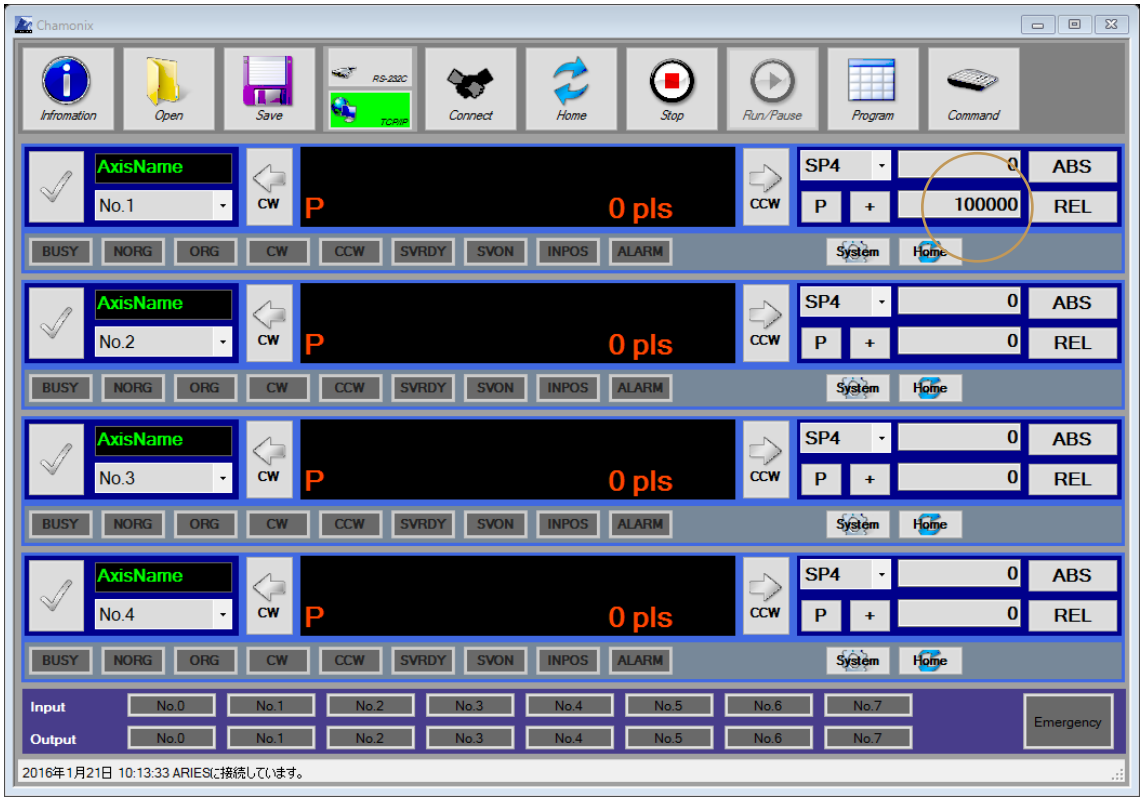


Press the ABS button with an absolute position to decide the position using absolute position management.

Press an illuminating ABS button to stop the only axis.

The BUSY indicator illuminates while the motor is driving.

10.3. Setting of Deciding Position with Relative Position Management

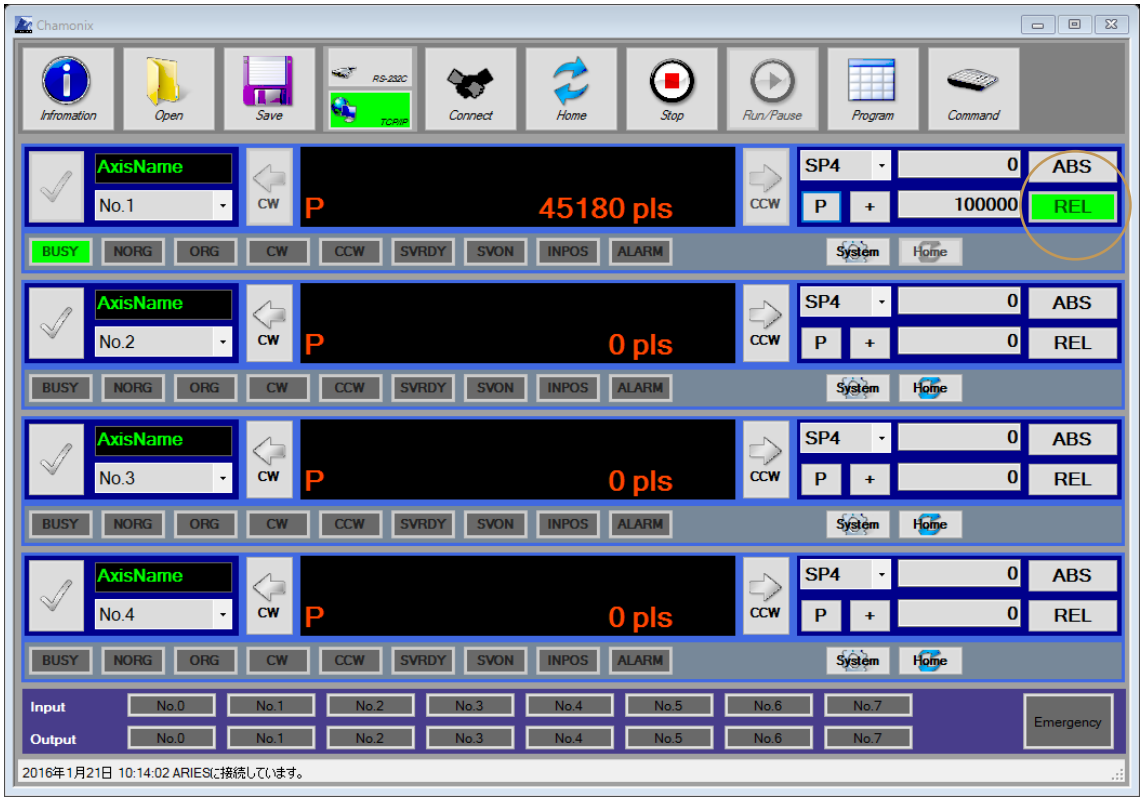


Enter a relative position of its destination in this text box.

Click a sign button to change the sign.

If a conversion value of control resolution is set in the system parameter, enter the conversion value. By default, setting for input uses motor pulse.

10.4. Executing of Deciding Position with Relative Position Management



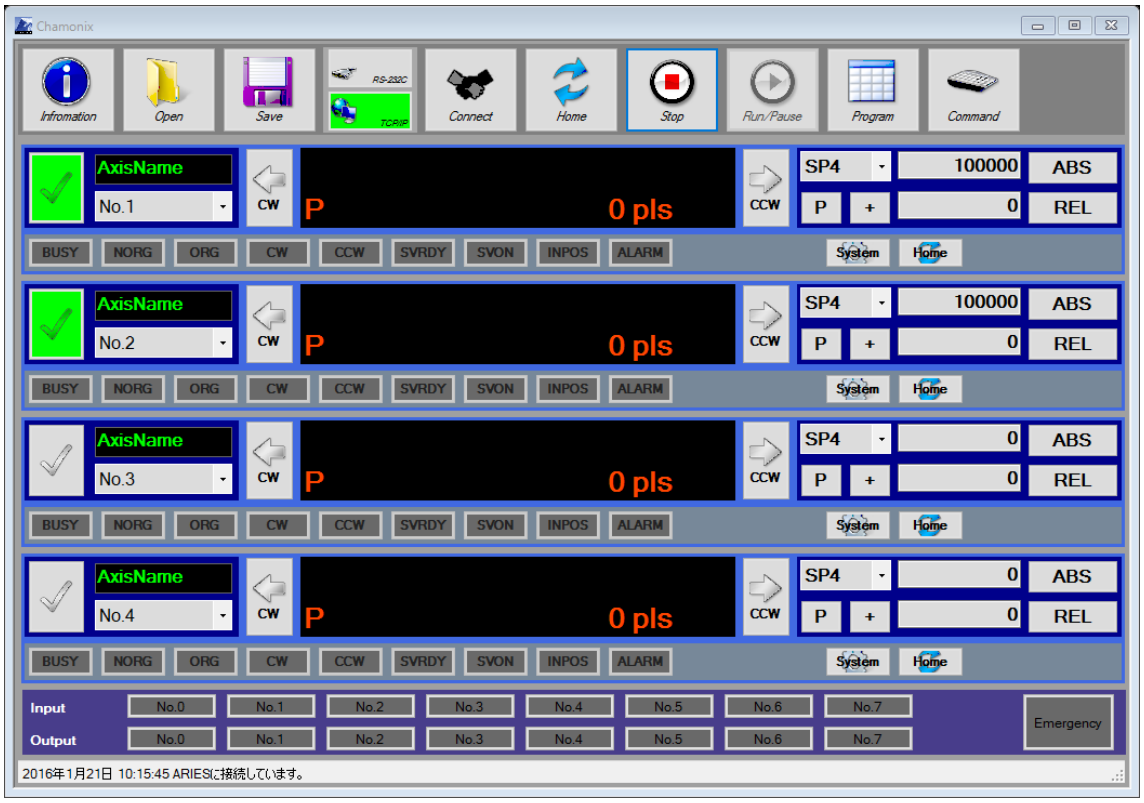
Click the REL button with a relative position to move the position using relative position management.

Press an illuminating REL button to stop the only axis.

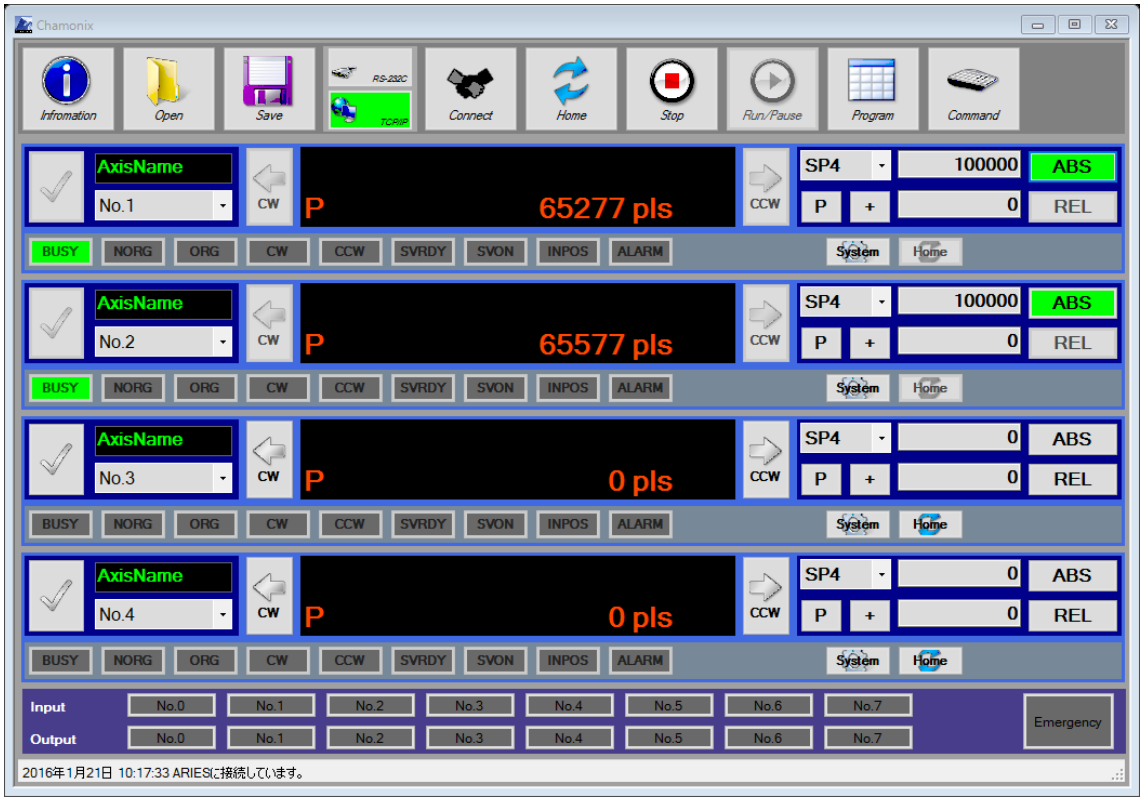
The BUSY indicator illuminates during drive.

10.5. Executing Multi-Axis Drive

Select the check button of each axis,



and press the ABS button or the REL button to drive the selected axes consecutively. The axes start driving at the same time.

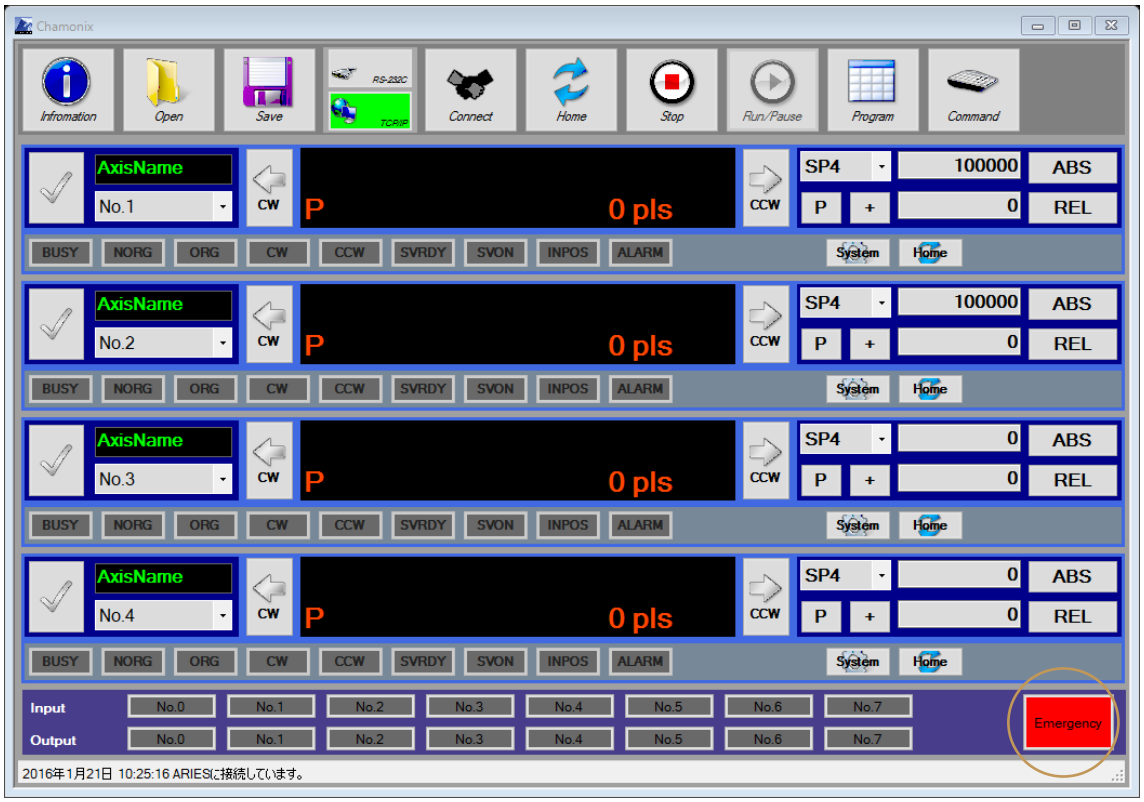


The BUSY indicator illuminates during drive, and the ABS (absolute position management) button or the REL (relative position management) button illuminates depending on the selected drive mode.

The GUI display performs axis-by-axis reading of the drive status and current position, and a time lag occurs during the reading. Updating will therefore not be conducted at the same time.

11. Receiving Emergency Stop!

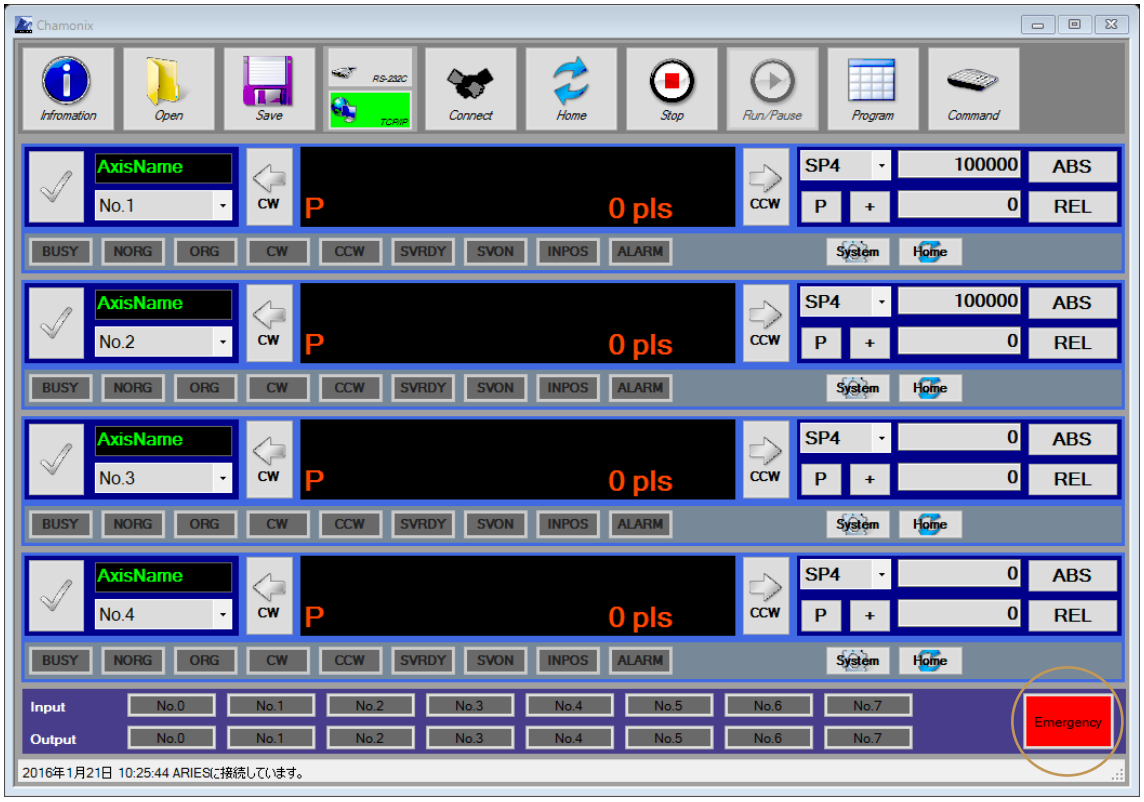
11.1. Receiving an Emergency Stop Signal (ARIES)



If ARIES is used for the controller, the Emergency indicator illuminates in red when an emergency stop signal is received.

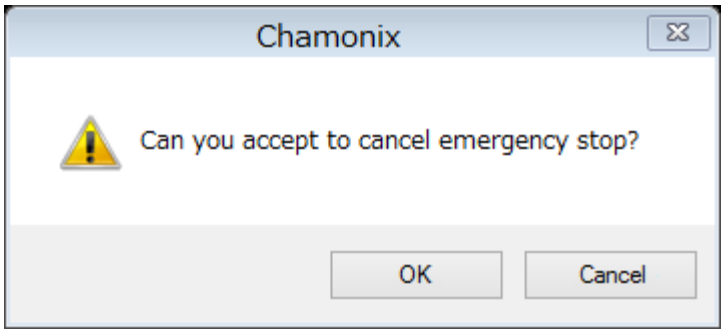
The controller will generate no motor pulse in this condition. Therefore, each motor cannot drive.

11.2. Recovering from Emergency Stop (ARIES)



For ARIES, press the Emergency button to call off the emergency stop after recovering the circuit of the emergency stop signal and checking safety.

When you press the Emergency button, the message is shown as below. Press OK to call off the emergency stop. Press Cancel to continue the emergency stop.



For recovery from an emergency stop, make sure that cause of the emergency stop is removed and safety is guaranteed.

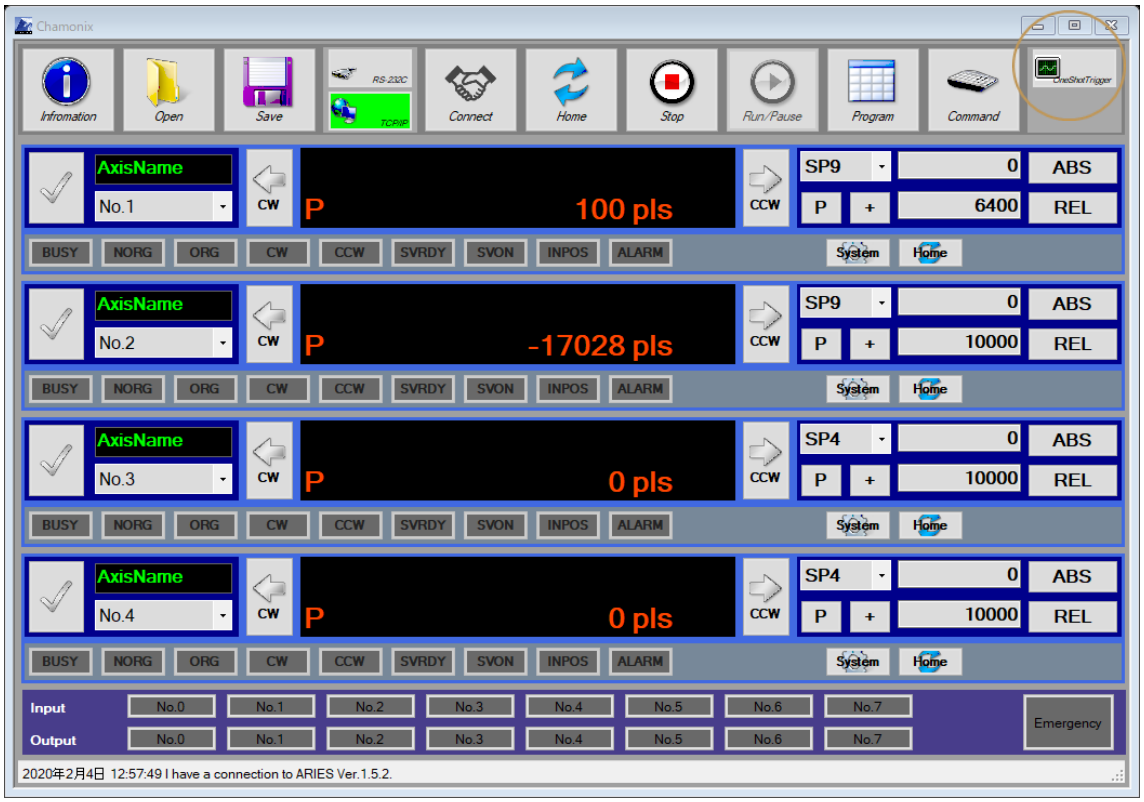
11.3. Recovering from Emergency Stop (CRUX)

For CRUX, Chamonix does not display an emergency stop signal. An emergency stop works if EMG illuminates in the front panel of CRUX.

Recover the circuit with the emergency stop signal.

12. Triggr Output

12.1. Output of One Shot Trigger Signal (ARIES)[16]

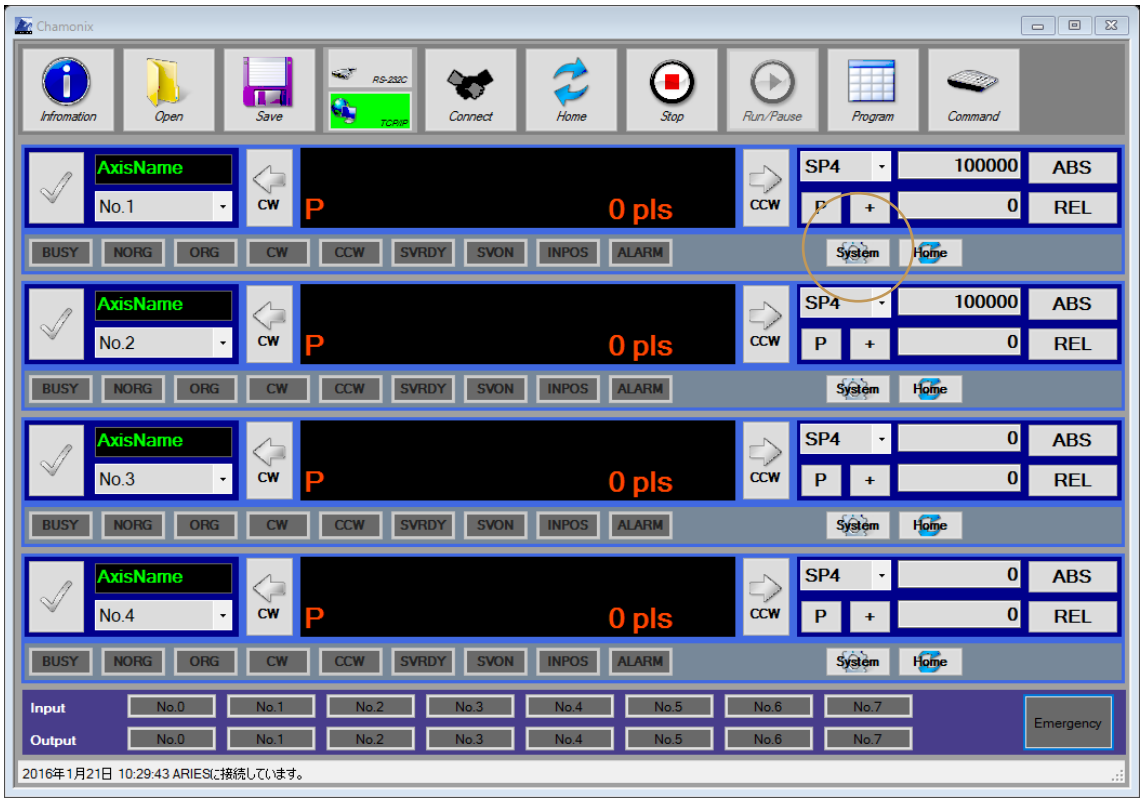


When ARIES is used, Chamonix can output trigger signal to outside device. By pressing One shot Trigger Button, trigger signal shall output. The pulse width can be set by right-click on One shot trigger button.

[16] This feature can be used Chamonix Ver.1.2.3 and after.

13. Changing System Parameters

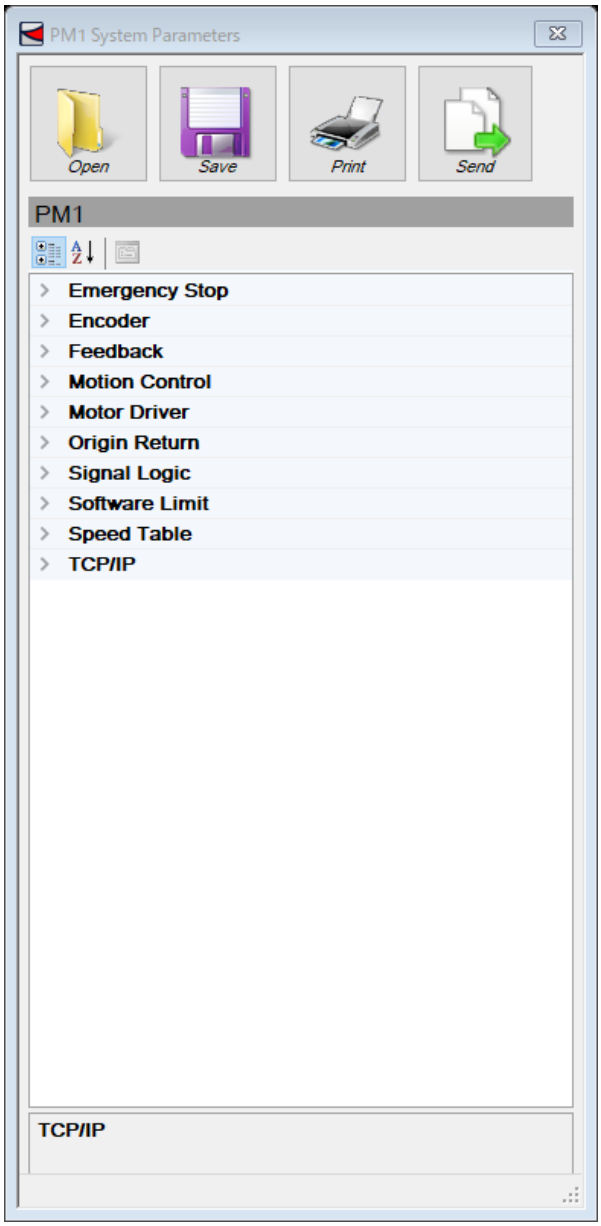
13.1. Starting Parameter Setting



Click the System button of each drive axis to start System Parameters, a window to set system parameters. System parameters are received from the controller during start. The system parameter of each axis opens as a property shown in the next section.

Parameter that can be set are a system parameter and speed table. Theses parameters is designed so that compatibility between controllers can be kept as much as possible[17].

[17] This does not mean that compatibility between different models is entirely



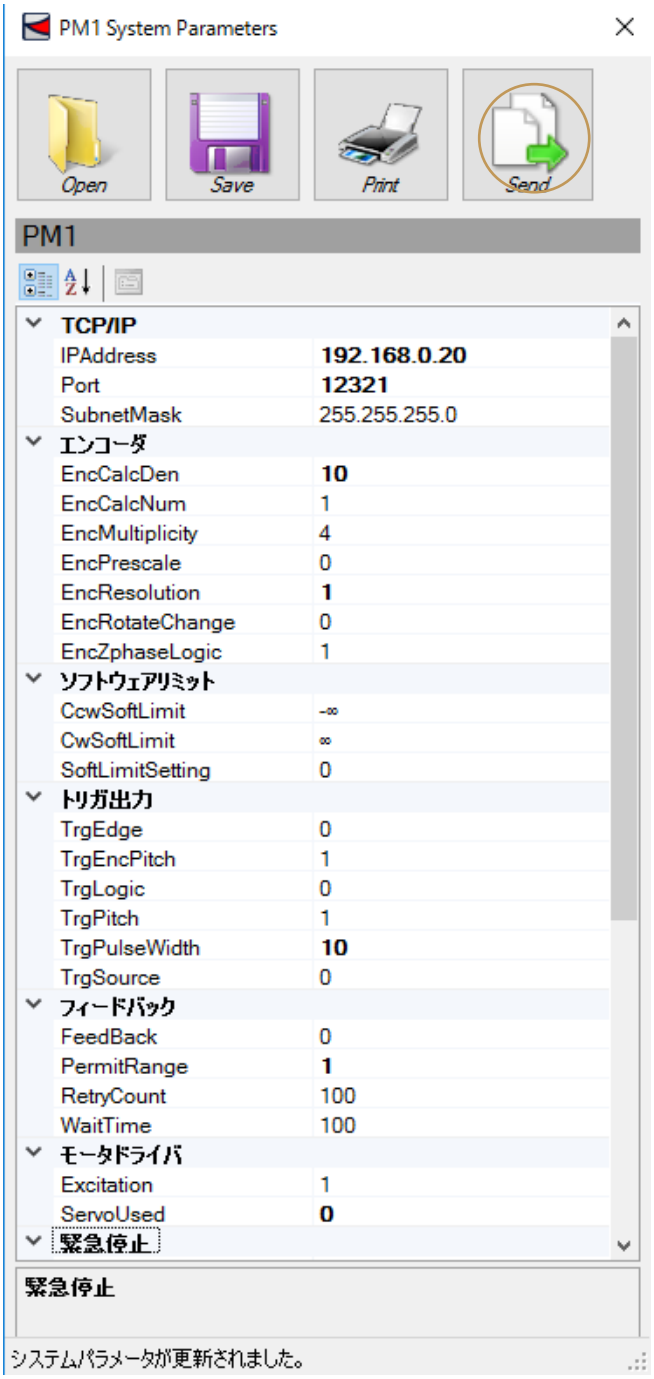
Features missed in a connected controller are not shown in the list. The above illustration shows an example of ARIES. The default value of Chamonix is saved in a stored file. [18]

---

guaranteed.

[18]When a higher-level model reads a file saved in a lower-level model, parameters that is not saved in the file starts by using the Chamonix default value.

13.2. Setting Parameters



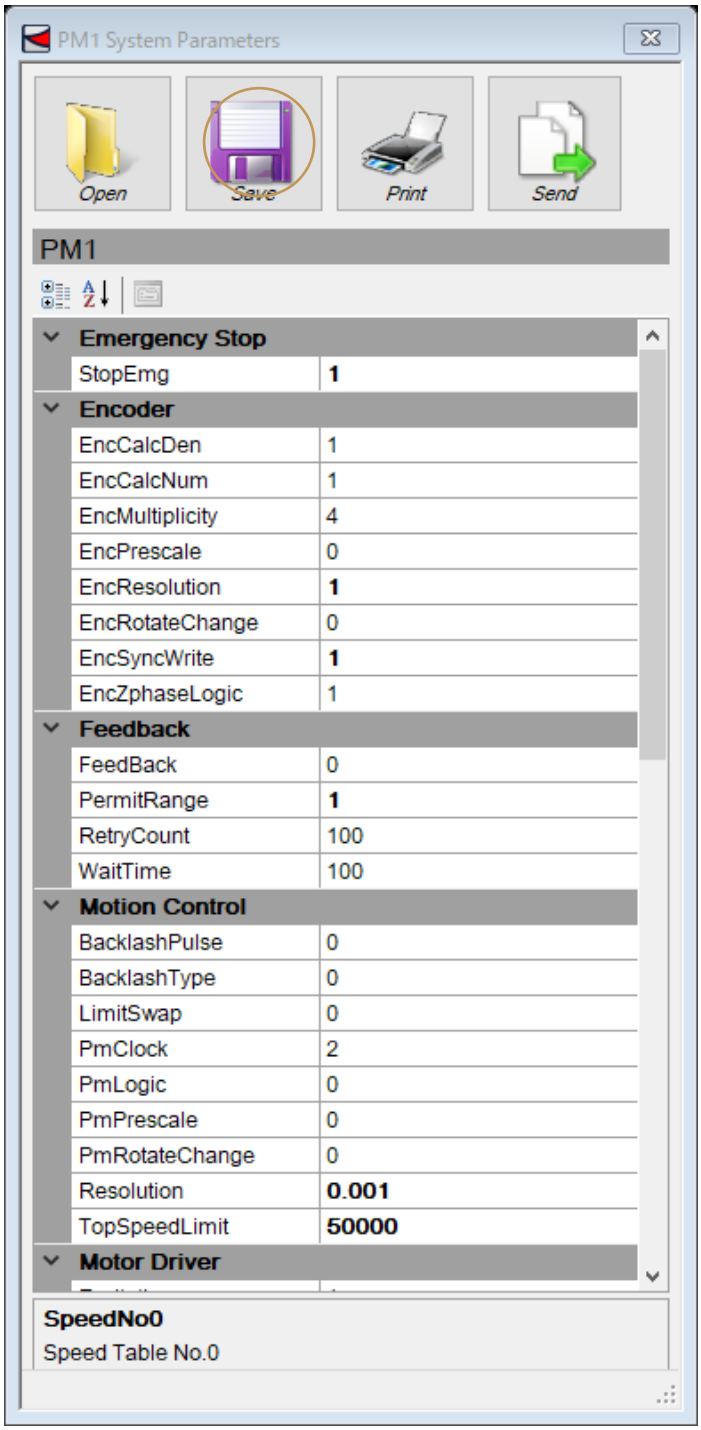
Enter values of system parameters of the controller in System Parameters to change them.

Press the Send button to reflect changed values for system parameters in the controller as a whole.

Number differences by models of controllers is absorbed in integration with application's properties.

A value of property is shown as normal when a default value is set in the controller and is shown as bold when value other than a default value is set.

13.3. Saving System Parameters to Files



Press the Save button to save system parameters to files by axis.

Press the Open button in the window to read a saved file.

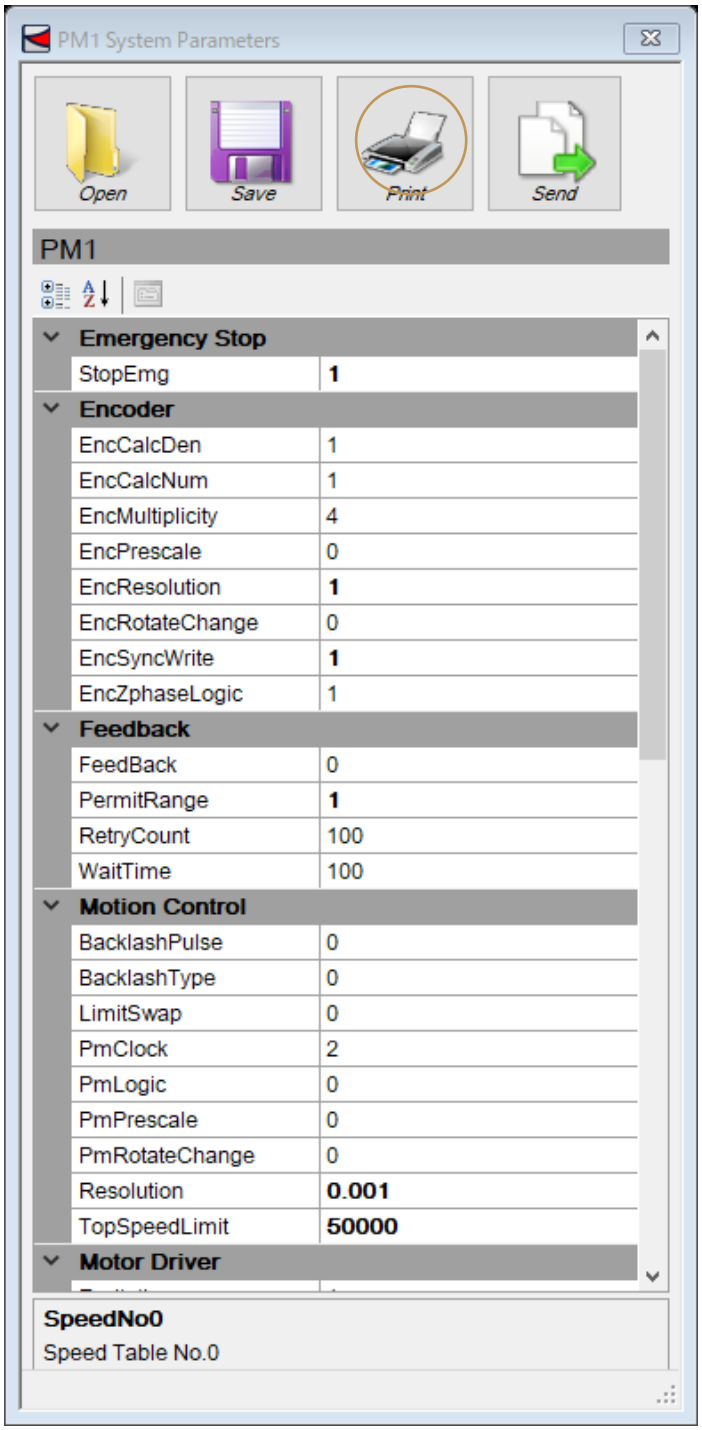
The file format is a bin file or XML file. These files are designed for common use in SC-210/410, CRUX, and ARIES[19].

Features missed in a connected controller are not shown in the list. The default value is stored in saving.

When a higher-level model reads a file saved in a lower-level model, parameters that is not saved in the file starts by using the Chamonix default value.

[19] This does not mean that compatibility between different models is entirely guaranteed.

13.4. Printing Parameters



Click the PRINT button to print parameters like the next page.

Date: 2017/12/06 17:52:58
Controller Type: ARIES

PM1

Emergency Stop	
StopEmg	1
Encoder	
EncCalcDen	1
EncCalcNum	1
EncMultiplicity	4
EncPrescale	0
EncResolution	1
EncRotateChange	0
EncSyncWrite	1
EncZphaseLogic	1
Feedback	
FeedBack	0
PermitRange	1
RetryCount	100
WaitTime	100
Motion Control	
BacklashPulse	0
BacklashType	0
LimitSwap	0
PmClock	2
PmLogic	0
PmPrescale	0
PmRotateChange	0
Resolution	0.001
TopSpeedLimit	50000
Motor Driver	
Excitation	1
MictoStepMode	0
ServoAlarmValid	0
ServoUsed	0
Origin Return	
OrgOffset	0
OrgScanSpeed	500
OrgType	4
Signal Logic	
LimitLogic	1
NOrgSignalLogic	0
OrgSignalLogic	0
Software Limit	
CcwSoftLimit	--
CwSoftLimit	--
SoftLimitSetting	0
Speed Table	
Speed_Hi	10,8000,50,15,3
Speed_Lo	5,4000,25,10,3
SpeedNo0	500,1000,16,16,2
SpeedNo1	500,2000,20,20,2
SpeedNo2	500,3000,24,24,2
SpeedNo3	500,4000,28,28,2
SpeedNo4	500,5000,32,32,2
SpeedNo5	500,6000,36,36,2
SpeedNo6	500,7000,40,40,2
SpeedNo7	500,8000,44,44,2
SpeedNo8	500,9000,48,48,2
SpeedNo9	500,10000,52,52,2
TCP/IP	
IPAddress	192.168.100.20
Port	12321
SubnetMask	255.0.0.0

13.5. Properties of System Parameters

This section summarizes properties of system parameters simply.

TCP/IP (ARIES only)

- **Port** TCP/IP port
- **SubnetMask** Subnet mask of the controller

Encoder (except CRUX and CRUX-A)

- **EncCalcDen** Motor pulse/encoder resolution ratio (denominator)
- **EncCalcNum** Motor pulse/encoder resolution ratio (numerator)
- **EncMultiplicty** Encoder multiplication setting
- **EncPrescale** Prescale of encoder values
- **EncResolutiion** Resolution per pulse of the encoder
- **EncRotateChanige** Change of direction to add the encoder
- **EncZphaseLogic** Switchover of the logic of the encoder Z phase

Software Limit

- **CwSoftLimit** Software limit for CCW direction (Enter this with pulse.)
- **CwSoftLimit** Software limit for CW direction (Enter this with pulse.)
- **CwSoftLimitSetting** Selection of use of software limit

Trigger Output (ARIES only)

- **TrgEdge** Selection of the edges of trigger signals
- **TrgEncPitch** Frequency dividing ratio of trigger signals for encoder pulse synchronous
- **TrgLogic** Switchover of the logic of trigger output
- **TrgPitch** Frequency dividing ratio of trigger signals for motor pulse synchronous
- **TrgPulseWidth** Pulse width of trigger output
- **TrgSource** Selection of trigger signal sources

Feedback

- **FeedBack** Feedback control formula
- **PermitRange** Tolerance level of encoder pulse
- **RetryCount** The number of retries for feedback
- **WaitTime** Waiting time for feedback

Motor Driver

- **CurrentNo** Battery setting number (CRUX only)
- **Excitation** Motor excitation
- **MicroStepNo** The number of divided number of micro steps (CRUX only)
- **ServoUsed** Use of the servo motor (ARIES only)

Emergency Stop

- **StopEmg** Emergency stop formula

Drive Control

- **BacklashPulse** The number of pulse of backlash compensation
- **BacklushType** Backlash compensation method
- **LimitSwap** Selection between limit signals of CW and CCW
- **PmClock** Switchover between pulse output formulas (ARIES only)
- **PmLogic** Switchover between pulse output logical formulas
- **PmPrescale** Pulse value prescale
- **PmRotateChange** Change of motor rotation direction
- **Resolution** Motor resolution per pulse
- **TopSpeedLimit** Maximum speed limit value (ARIES only)
- **UsingEMGModeForJog** Stop mode of jog button

Returning to Origin

- **EncSyncWrite** Zero reset of the encoder value for returning to origin (except CRUX and CRUX-A)
- **OrgOffset** Original point offset value
- **OrgScanSpeed** Speed for searching origin (except SC series)
- **OrgType** Type to return to origin

Logical Signals

- **LimitLogic** Selection between limit signals of CW and CCW
- **NOrgSignalLogic** Switchover of logic of NORG sensor signals
- **PrgSignalLogic** Switchover of logic of ORG sensor signals

Speed Table

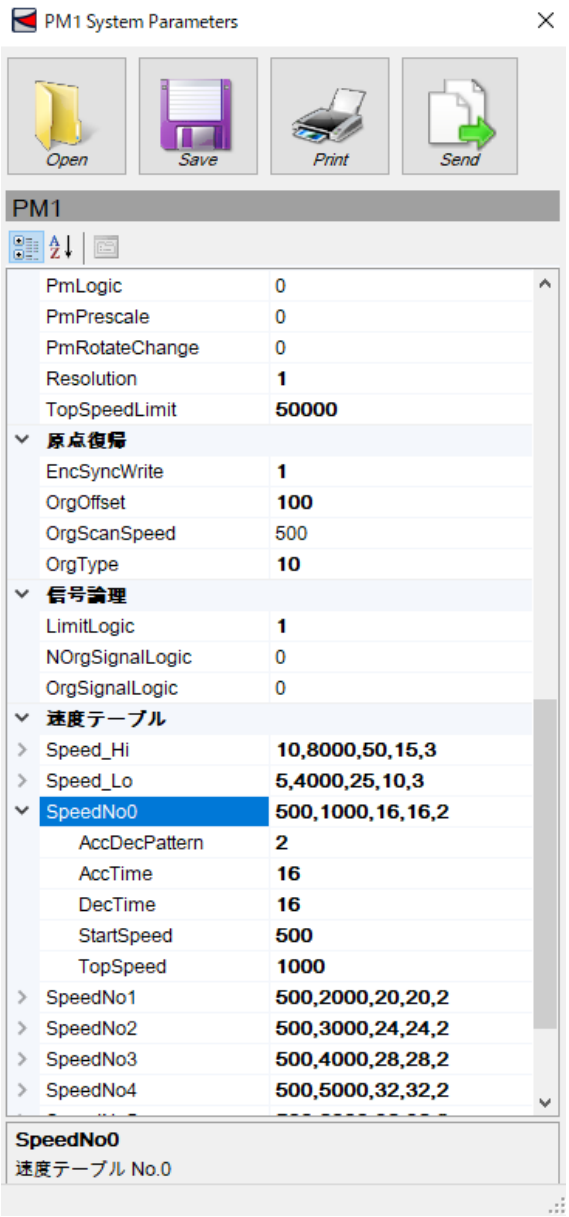
- **SpeedNo.0** Setting of speed table No.0
  - **AccDecPattern** Start speed
  - **AccTime** Acceleration time (x10 msec)
  - **DecTime** Deceleration time (x10 msec, except CRUX and CRUX-A)
  - **StartSpeed** Start speed
  - **TopSpeed** Top speed
- **SpeedNo.1** Setting of speed table No.1 (Defining parameters is the same as that of No.0)
- **SpeedNo.2** Setting of speed table No.2 (Defining parameters is the same as that of No.0)
- **SpeedNo.3** Setting of speed table No.3 (Defining parameters is the same as that of No.0)

- **SpeedNo.5** Setting of speed table No.5 (Defining parameters is the same as that of No.0)
- **SpeedNo.6** Setting of speed table No.6 (Defining parameters is the same as that of No.0)
- **SpeedNo.7** Setting of speed table No.7 (Defining parameters is the same as that of No.0)
- **SpeedNo.8** Setting of speed table No.8 (Defining parameters is the same as that of No.0)
- **SpeedNo.9** Setting of speed table No.9 (Defining parameters is the same as that of No.0)
- **Speed_Hi** High-speed speed table of jog operation with PYXIS
- **Speed_Lo** Low-speed speed table of jog operation with PYXIS

13.6. Change of Speed table

The way to change a speed table shall be described in this section. After name of speed table is clicked, their parameters shall be opened. Available parameter is different by each controller type.

In the case of ARIES/LYNX, AccDecPattern is Acceleration and deceleration pattern.



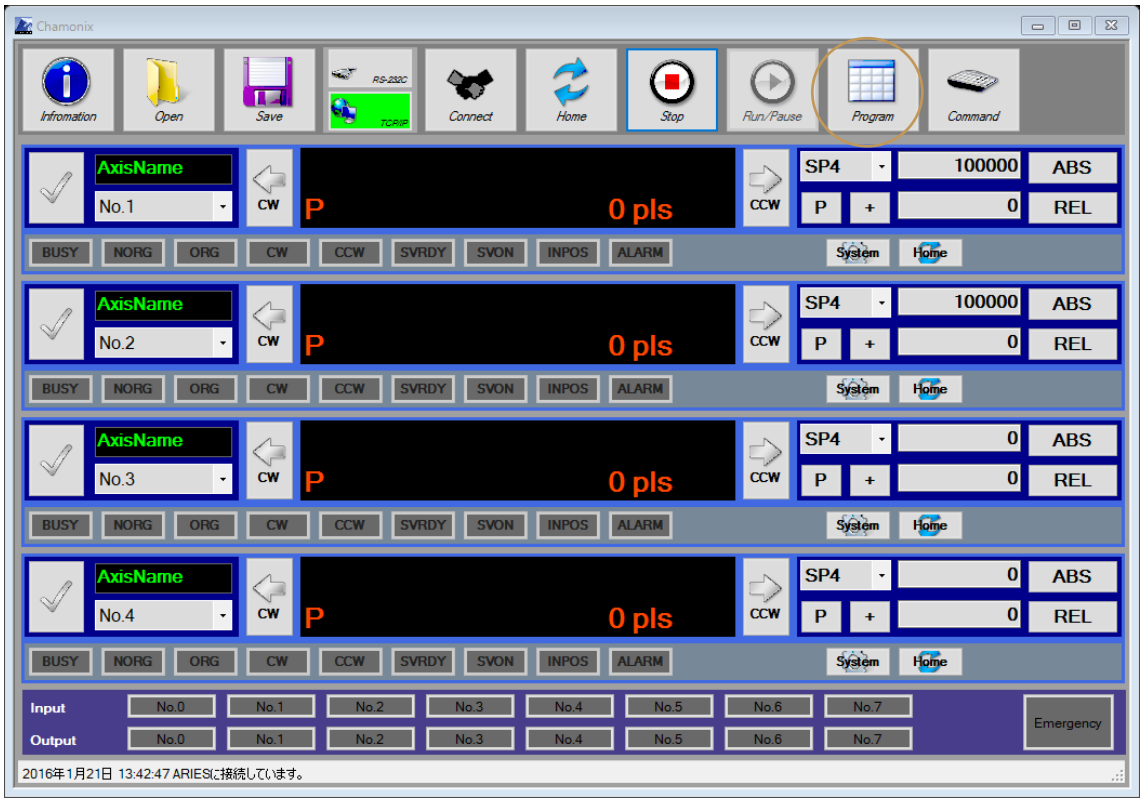
AccTime is acceleration time(unit of 10msec) , DecTime is deceleration time (unit of 10msec), StartSpeed is start speed(pps) and TopSpeed means maximum speed (pps).

In the case of CRUX/CRUX-A, AccTime is acceleration time(unit of 10msec), AccDecPattern is acceleration and deceleration pattern. StartSpeed is start speed(pps) and TopSpeed means maximum speed(pps).

Then, in CRUX/CRUX-A, DecTime is set automatically the same value of AccTime.

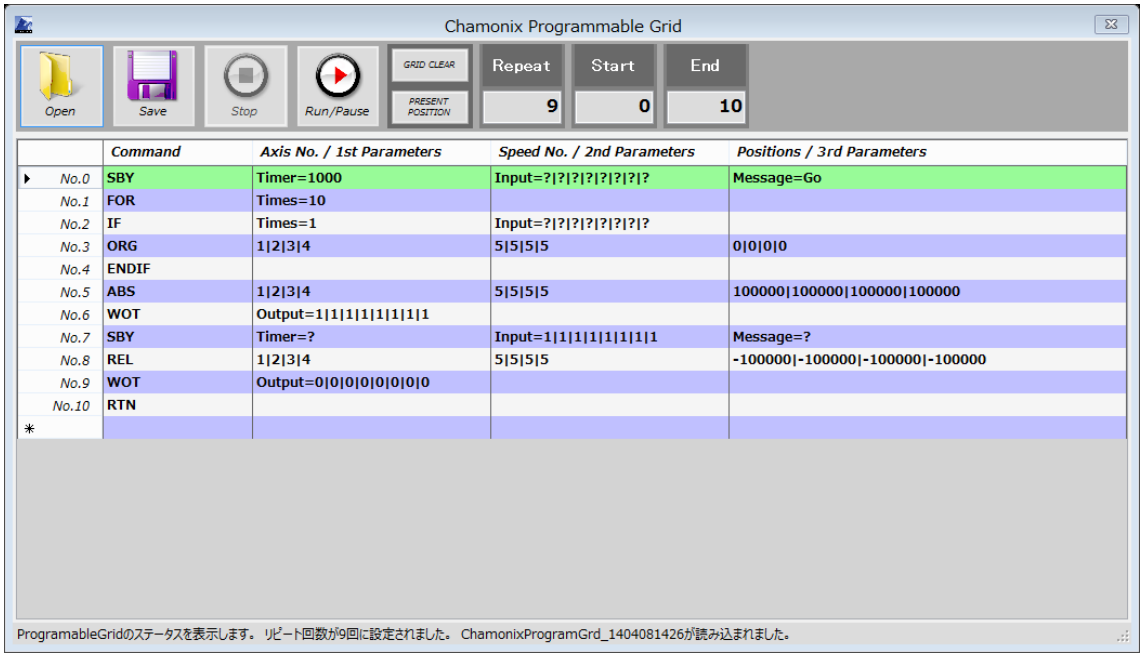
14. Using Programmable Grid

14.1. Starting Programmable Grid



Click the Program button to start Programmable Grid. Programmable Grid is a teaching feature that incorporated a feature like a simplified micro.

14.2. Basic Features of Programmable Grid



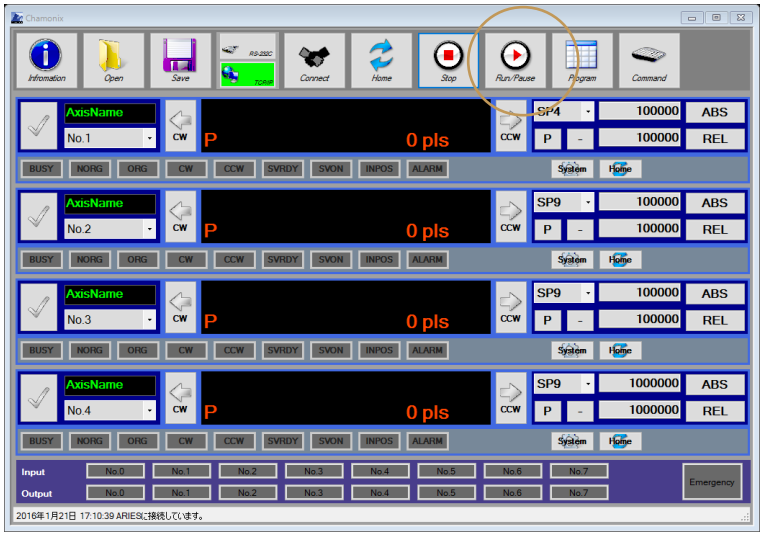
It drives automatically a stage connected with a controller according to a program written in the grid.

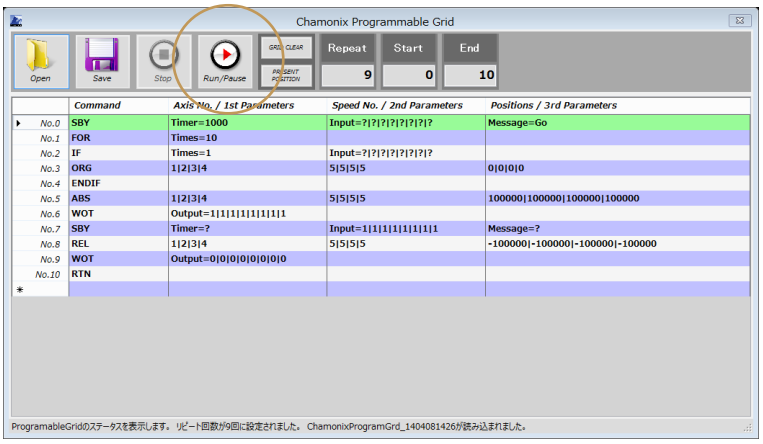
Commands in lines of a grid are executed in order from the top in principle.

Lines specified in Start and End are repeated the number of times specified in Repeat. If Endless or a negative number is entered in Repeat, the repeat is set to continue without end.

In the above example the lines of the program can be executed one-by-one basis, and the program will repeat nine times.

14.3. Running Programs



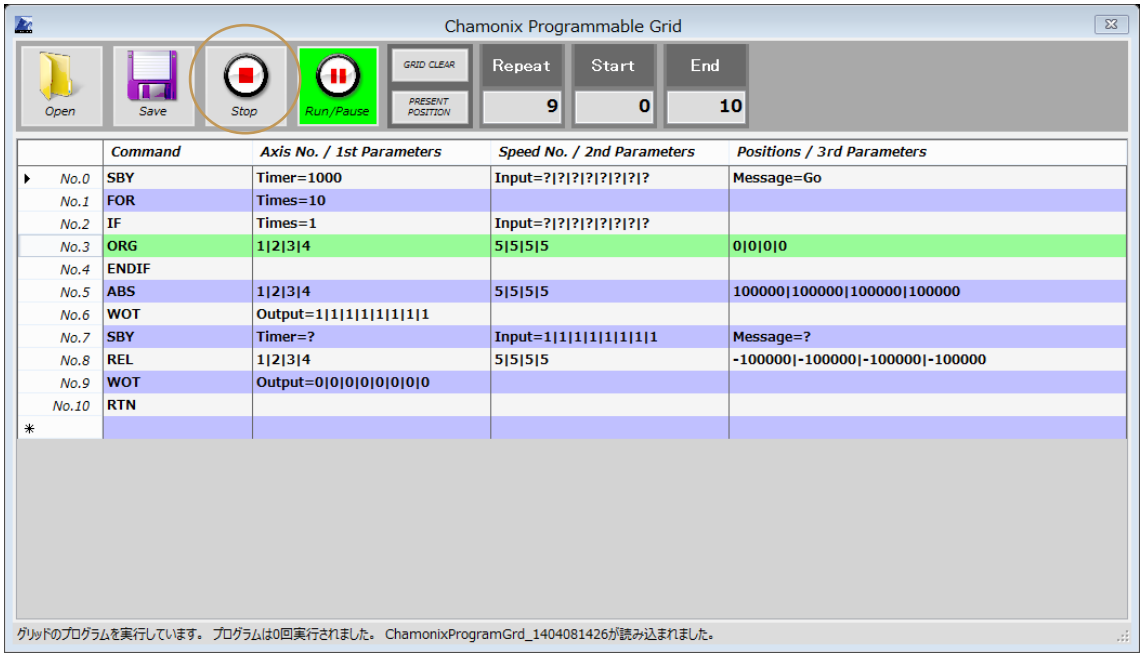


Click the RUN button in Chamonix or Chamonix Programmable Grid to run the program.

The PAUSE button replaces the RUN button and illuminates during the running of the program. The running line is colored green.

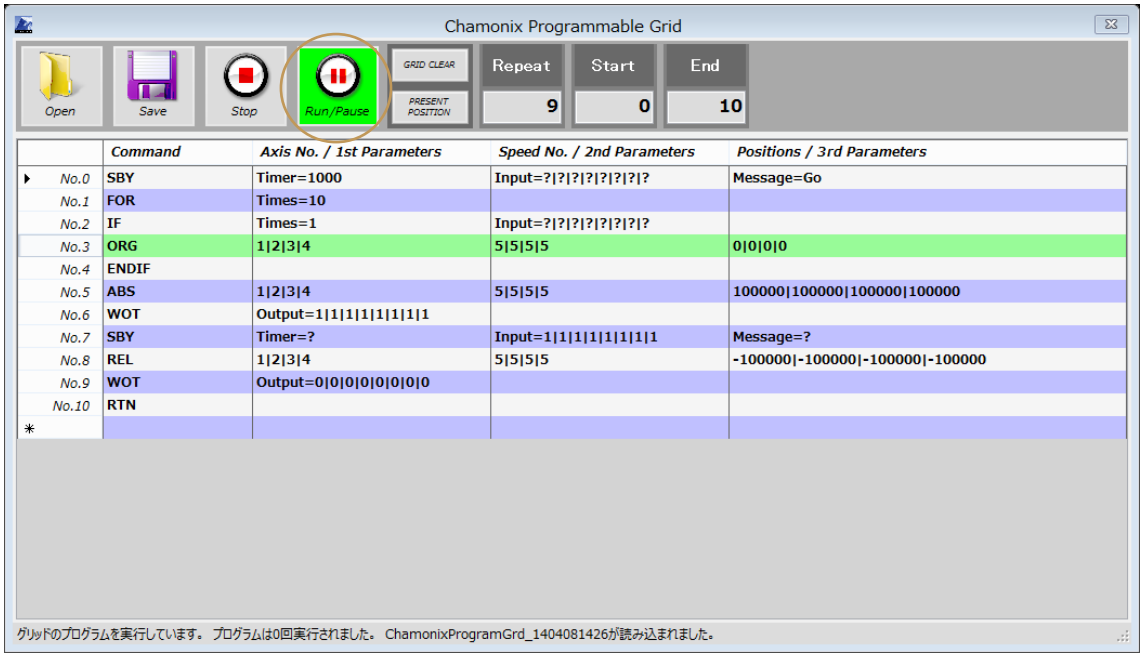
Click the PAUSE button to pause the program after finishing the running line. Press the RUN button to restart the paused program.

14.4. Stopping Programs

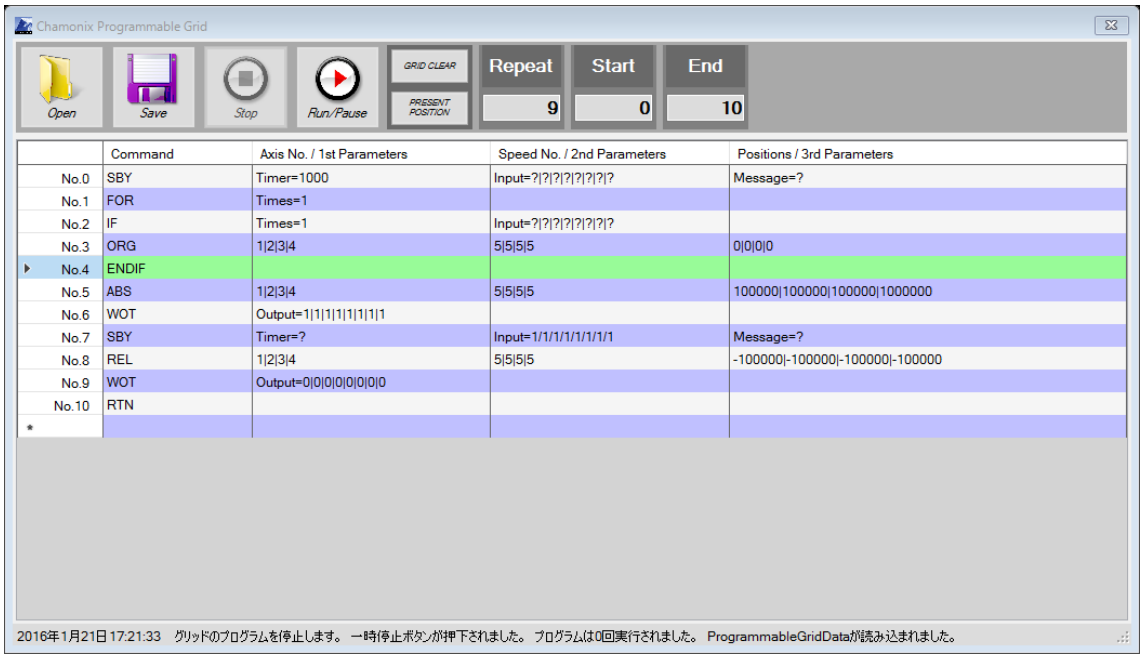


Click the STOP button to stop. Stop with the STOP button means an emergency stop, immediately stopping a program. It is impossible to restart from the line where the program stopped. When you may restart the program, please use the PAUSE button.

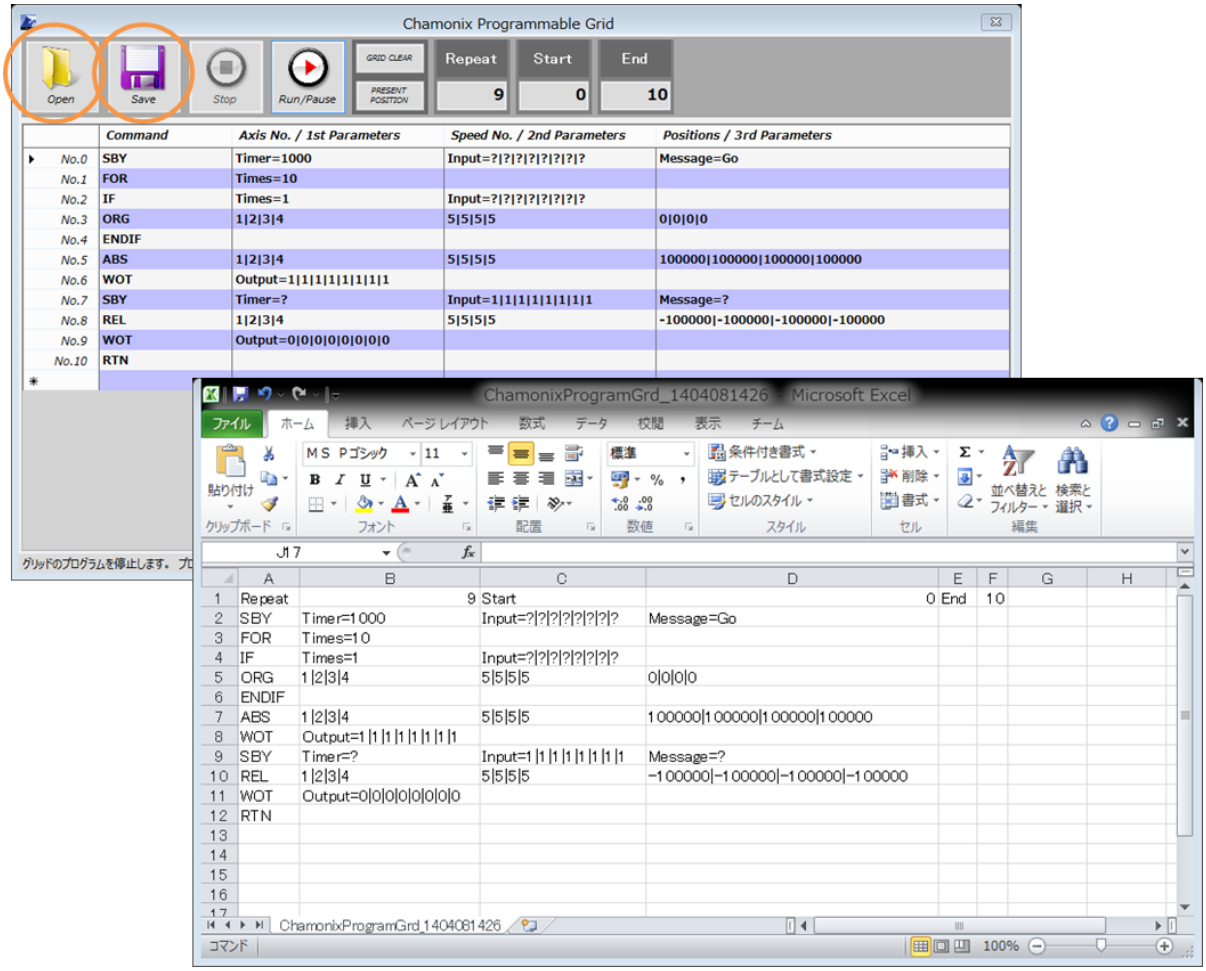
14.5. Stopping Programs Temporarily



The RUN button turns on light and changes to the PAUSE button. When the PAUSE button is pressed, the program is paused, and the screen should look like the illustration below after the current line is executed. Then, Press the RUN button to start the program again.



14.6. Inputting and Outputting Program Data



You can save program data to files.

Data can be read from files.

Files support DAT (standard), XML, and CSV (CSV1) formats. A DAT format is recommended.

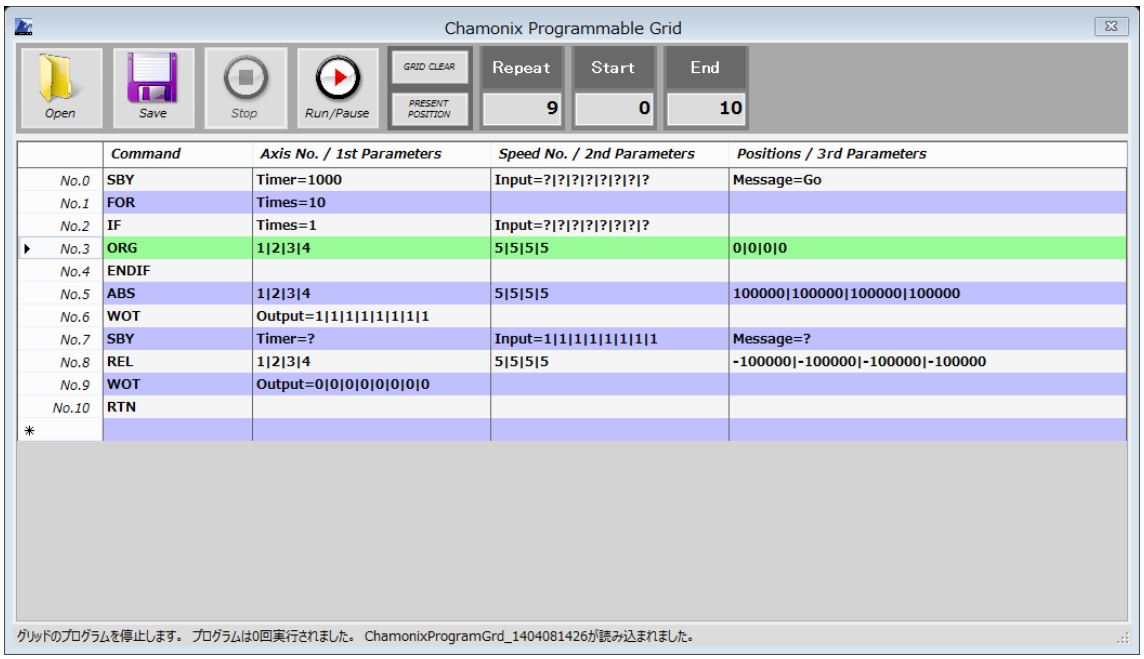
A break character of each parameter in Parameters uses "|" or "/". It is interpreted as "|" during an internal process. This reason is the inconvenience[20] in processing

[20] By default, Excel converts "1/2 " to "January 2" with a date function

CSV with Microsoft Excel.

14.7. Commands of Programmable Grid

14.7.1. ORG: Returning to Origin



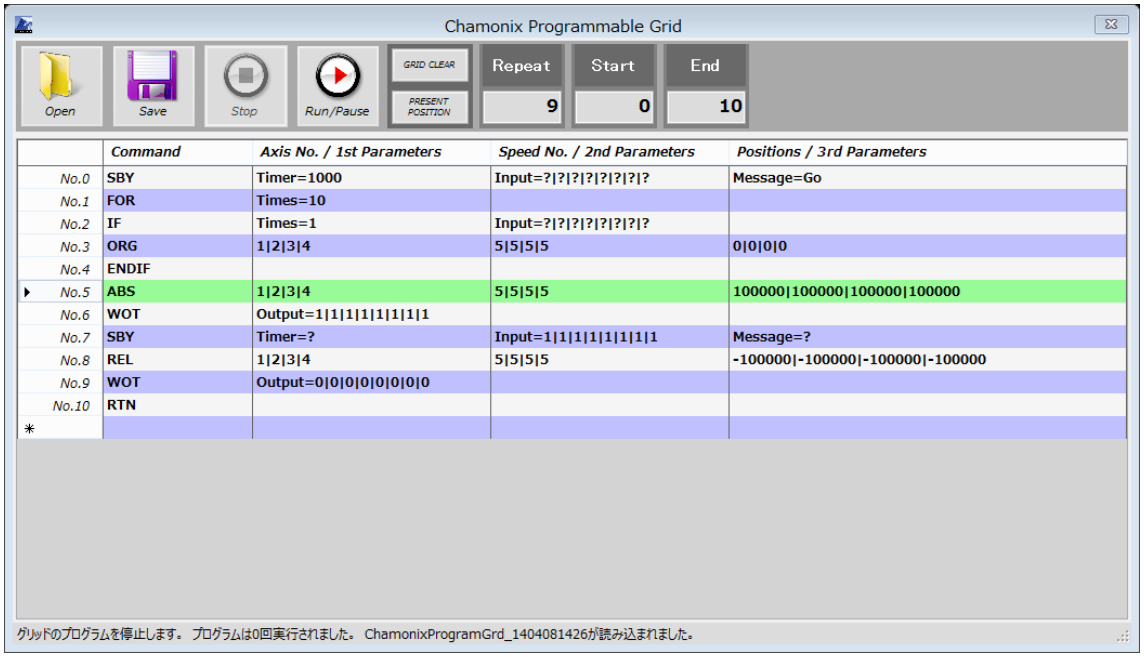
ORG performs returning to origin according to the settings of system parameters.

After completion of returning to origin,
it moves an axis specified by 1st Parameters
with a speed table specified by 2nd Parameters
to an absolute position specified by 3rd Parameters.

If the Resolution property is set in system parameters, specify a conversion value[21].

[21] By default, setting for input uses pulse.

14.7.2. ABS: Deciding Position with Absolute Position Management

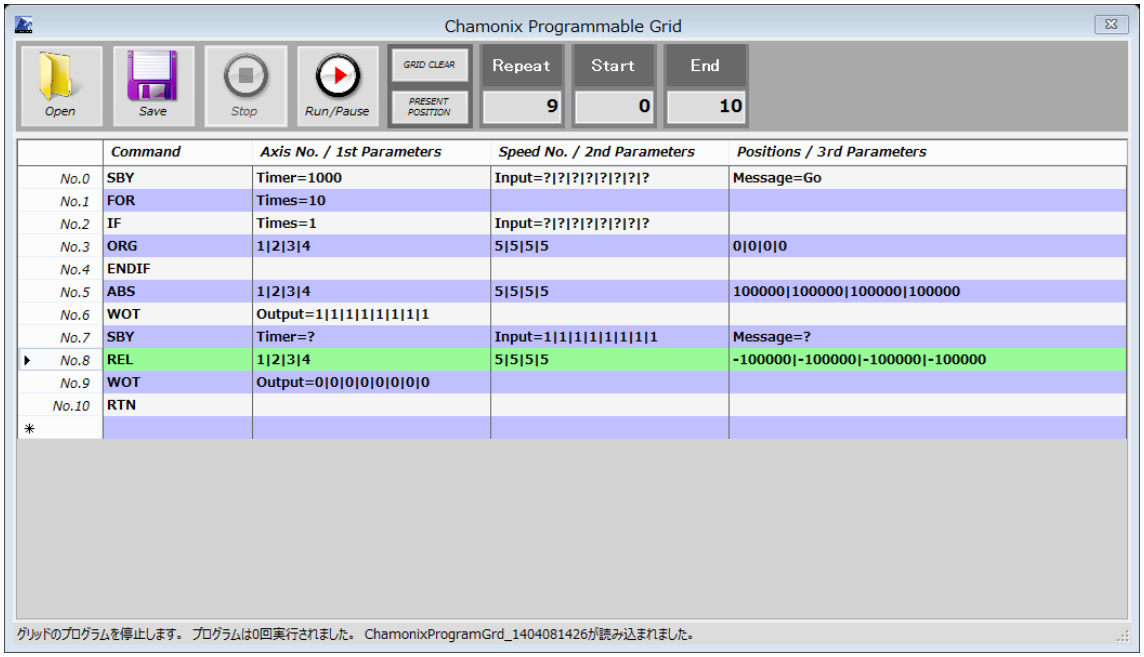


ABS allows moving with absolute position management
to an axis specified by 1st Parameters
at a speed specified by 2nd Parameters
to position specified by 3rd Parameters.

If the Resolution property is set in system parameters, specify a conversion value[22].

[22] By default, setting for input uses pulse.

14.7.3. REL: Deciding Position with Relative Position Management

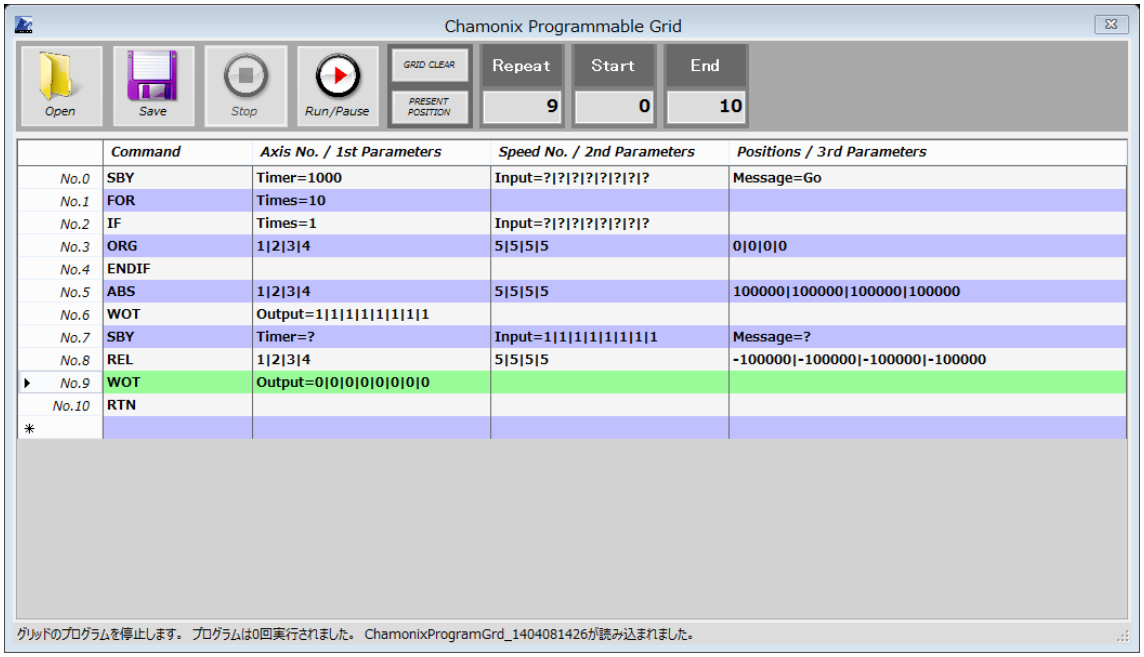


REL allows moving with relative position management
to an axis specified by 1st Parameters
at a speed specified by 2nd Parameters
to position specified by 3rd Parameters.

If the Resolution property is set in system parameters, specify a conversion value[23].

[23] By default, setting for input uses pulse.

14.7.4. WOT: Controlling General-Purpose Output



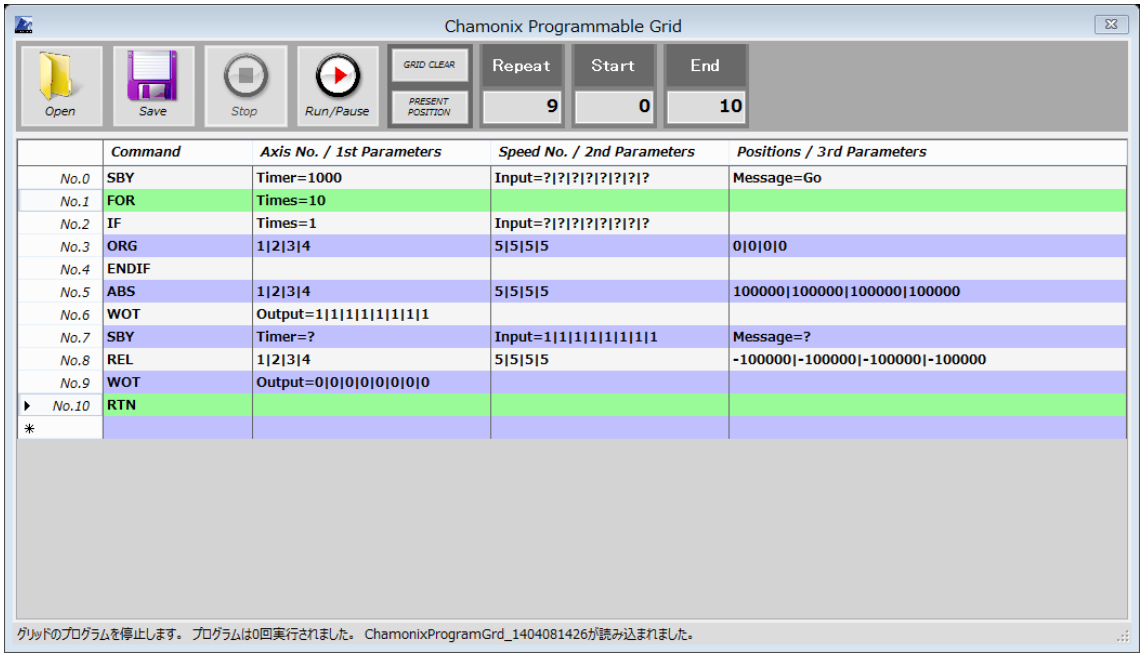
You can control the output of I/O for ARIES. The above example shows all the outputs of ARIES with 8ch are turned to ON.

Setting 1st Parameters to "Output=0|0|0|0|0|0|0|0" indicates that all outputs are turned to OFF.

The ? symbol indicates that the parameter is NULL.

When a controller that does not have I/O is connected, this command will be ignored.

14.7.5. FOR: Repeat Loop



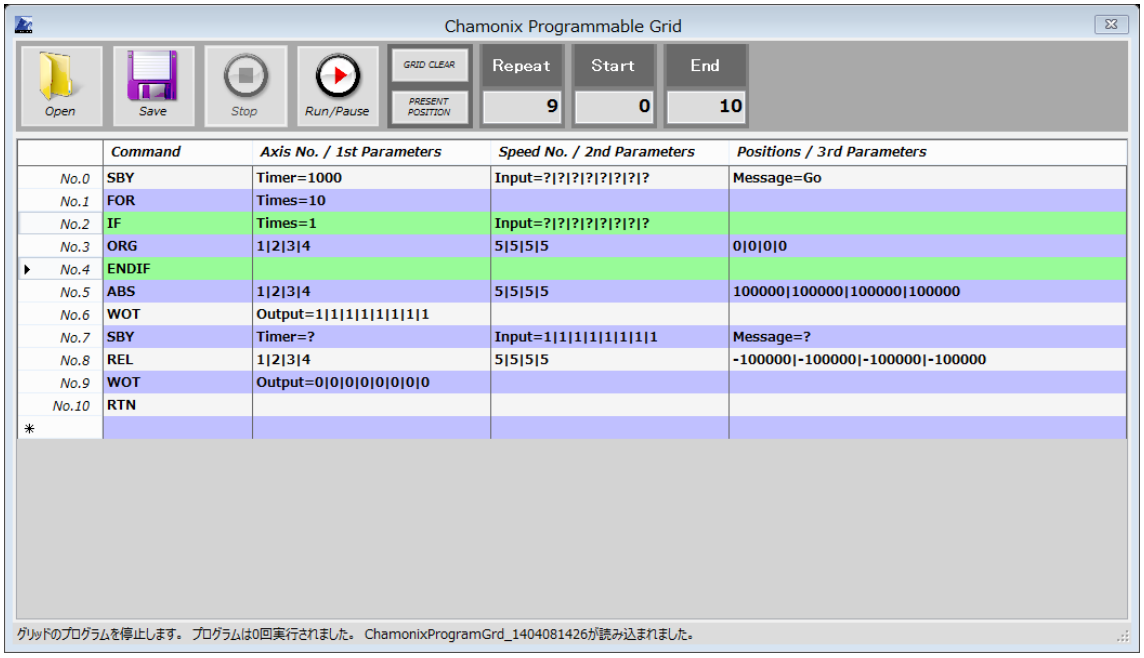
The FOR command allows repeating the number of times specified by the "Times=" parameter specifier of 1st Parameters until first RTN command in lower line than its line.

The FOR commands can nest. When the FOR commands nest, the inner loop of a FOR command runs at the beginning.

The "Times=" parameter specifier of the IF command also covers the number of times of a running loop.

If specifying the FOR command, you must insert the RTN command in the lower line than the FOR command.

14.7.6. IF: Running with Conditions



When the conditions written in 1st Parameters or 2nd Parameters is met, the IF command runs lines until the ENDIF command.

You can specify the number of times of the loop by using Times= to 1st Parameters. The above example shows that the ORG command (returning to origin) is executed only if the loop is for the first time

You can specify input status of I/O for the controller as the condition to 2nd Parameters. This is ignored for a controller that does not have I/O.

If "?" is entered, the item will be ignored.

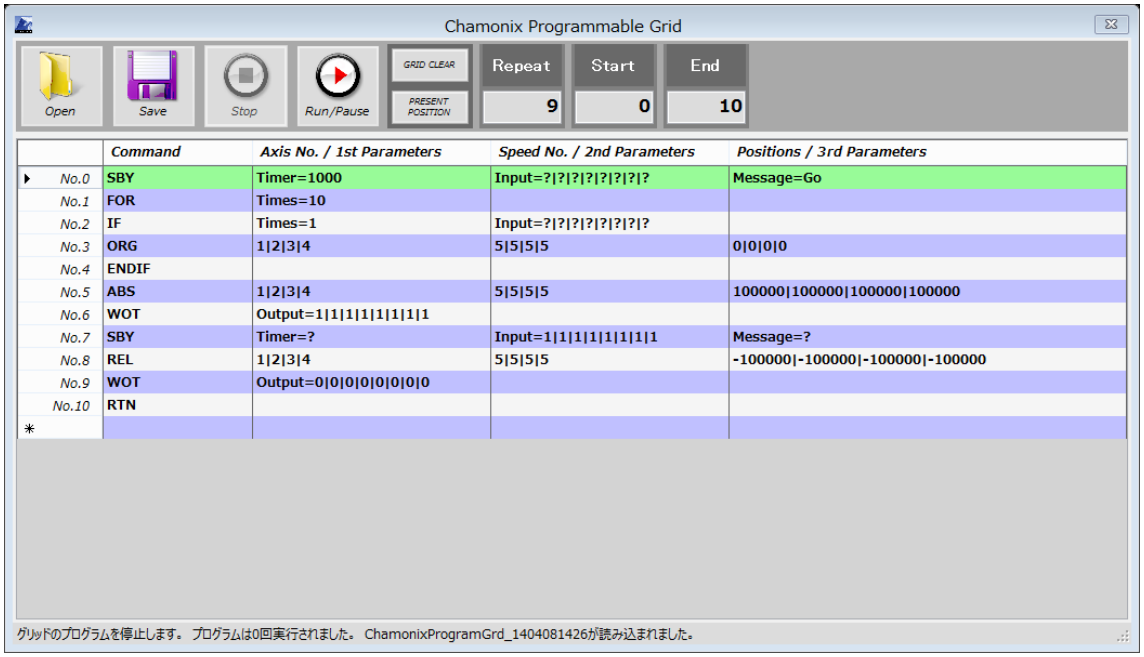
The above example shows a loop that repeats 10 times from No.1 to No.10 is set to run the No.3 line only if the loop is for the first time.

14.7.7. GOTO: Jumping to a Line

The GOTO command jumps to a line specified by 1st Parameters.

Although the GOTO is convenient command, it should be noted that the program may become complicated and an incorrect program may be set.

14.7.8. SBY: Standby

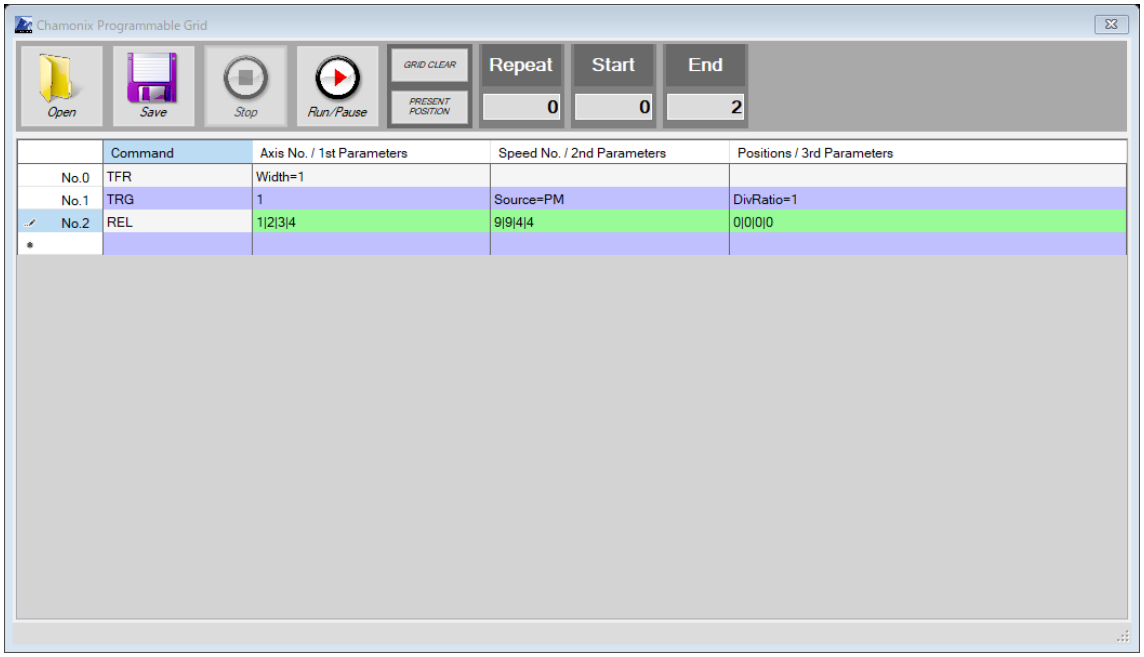


The SBY command stands by until any of conditions of 1st to 3rd Parameters is met.

Parameter specifier "Time=" of 1st Parameters can specify the timer in milliseconds. Parameter specifier "Input=" of 2nd Parameters can specify input of I/O of ARIES. Parameter specifier "Message=" of 3rd Parameters allows going through stop when specified character strings are send to the main window of Chamonix with SendMessage of WindowsAPI.

The ? symbol indicates that a parameter is not specified.

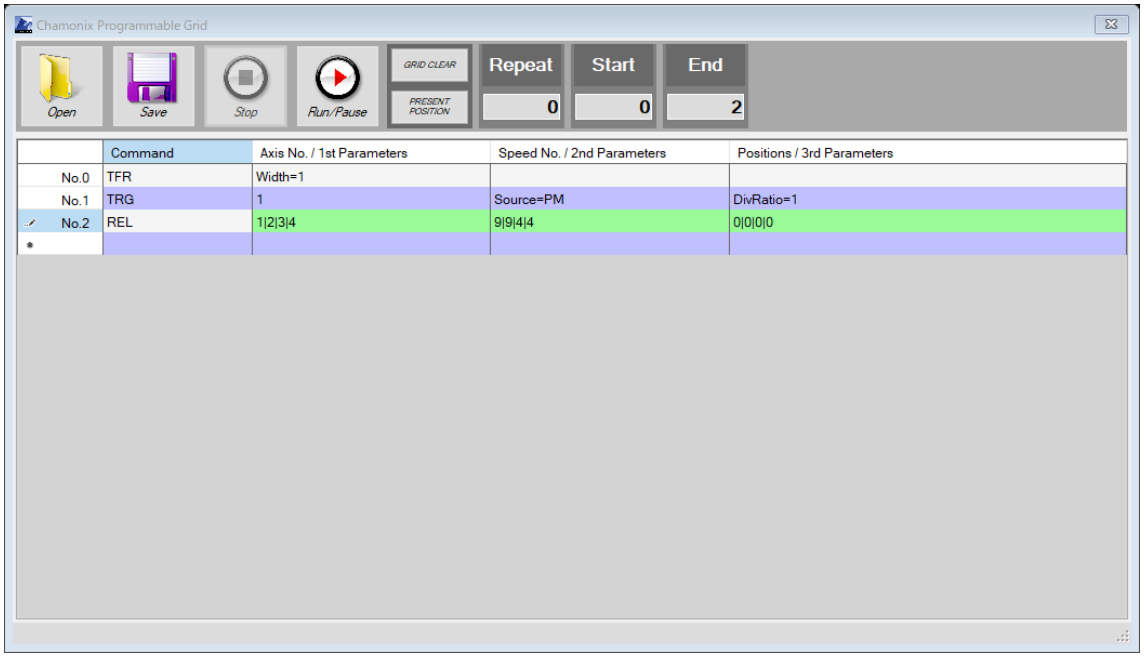
14.7.9. TFR : One Shot Trigger



In case of using ARIES, TFR command is available. By TFR command , on shot trigger output shall be executed.

Parameter specifier "Width=" of 1st parameters can specify pulse width by milli second.

14.7.10. TRG : Trigger Output Synchronized with Driving Axis



In the case of using ARIES, TRG command is available.TRG command set trigger output synchronized with driving of 1st or 2nd axis of ARIES. synchronized trigger shall output when driving by the conntroller on the first time after setting by this command. This setting is valiable for only one time driving.

1st parameters specify the based axis for trigger signal. It shall 1 or 2.Pa
Parameter specifier "Surce=" of 2nd parameters can specify signak source. where is the signals ource is pointed by "Source=PM": motor pulse, "Source=ENC": encoder pulse, respectively.
Parameter specifier "DivRatio=" can specify divided ratio of trigger signal. For exsample, When this parameters will set on "DivRatio=1000", Trigger signal shall output every 1000 pulse.

14.7.11. Command List of Programmable Grid

The following table lists the commands[24] of Programmable Grid. See descriptions of each command for more details.

ABS
- Drives with deciding position by using absolute position management

REL
- Drives with deciding position by using relative position management

ORG
- Returns to origin and decides position by using absolute position management after returning to origin is finished.

SBY
- Stands by until conditions are met.

FOR
- Repeats the number of times specified by the Times specifier until a line specified by RTN.

RTN
- Specifies a line where the FOR loop returns.

IF
- Runs lines until ENDIF if the conditions are met.

ENDIF
- Indicates the end of range where is run with conditions by the IF command.

WOT
- Sets output of I/O.

GOTO
- Jumps to specified line.

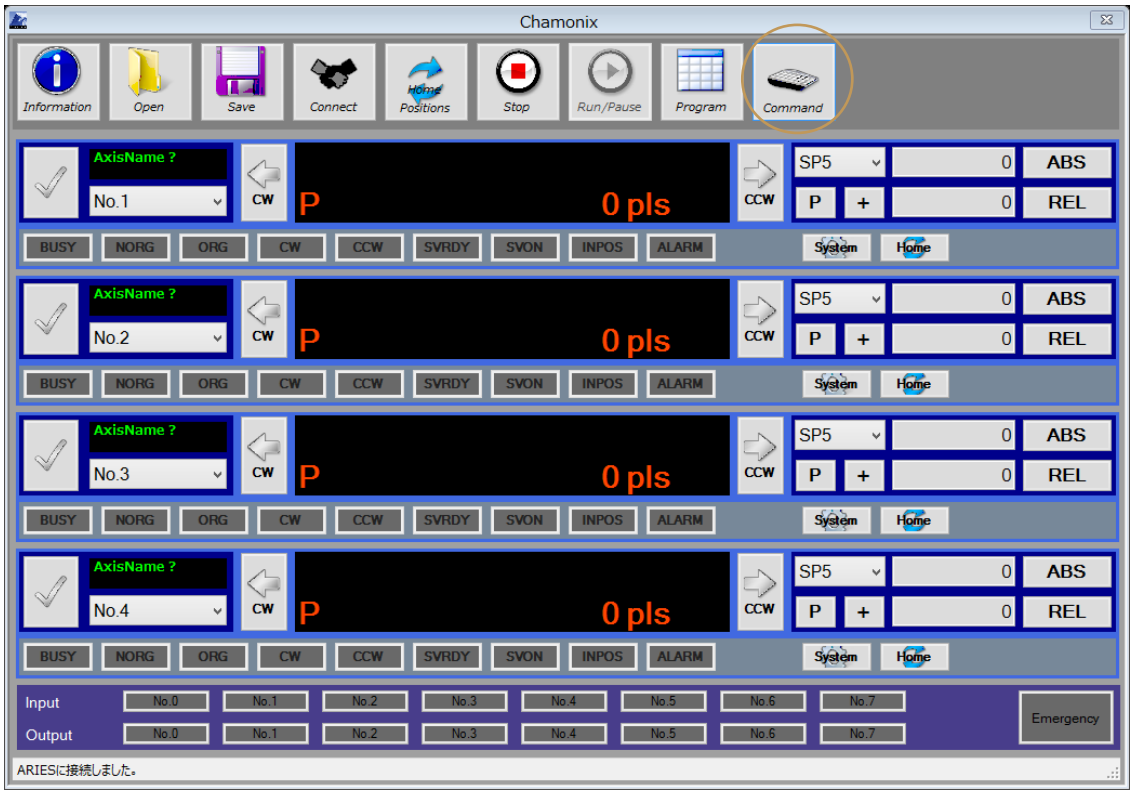
TFR
- Oneshot trigger output

TRG
- Trigger can be set shynchronized with 1st or 2nd axis of ARIES.

[24] Custom-order commands are excepted.

15. Entering Remote Commands Directly

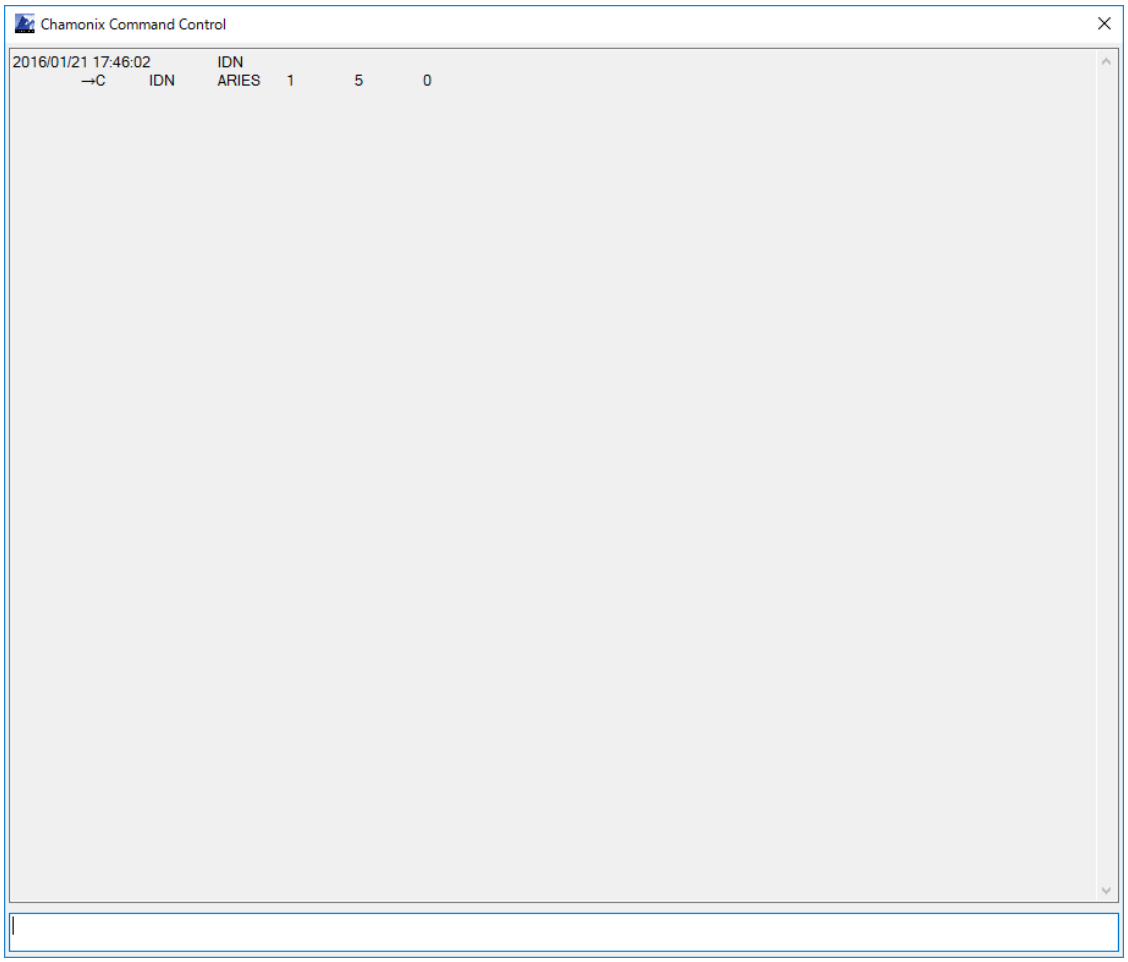
15.1. Starting Command Control



Click this button to start Chamonix Command Control.

Chamonix Command Control allows you to enter commands to send to a controller with a keyboard.

15.2. Sending Commands from Command Control



Enter a command in the send box and press Enter to send the command to the controller. The STX character[25] of a header for RS-232 communication and the CRLF character[26] of a delimiter are added automatically. See the operation manual for each controller for remote control formats. Entered lowercase letters will be converted into uppercase ones.

[25] "STX" is not three characters of "S", "T", and "X", but one control character indicating "start of text".

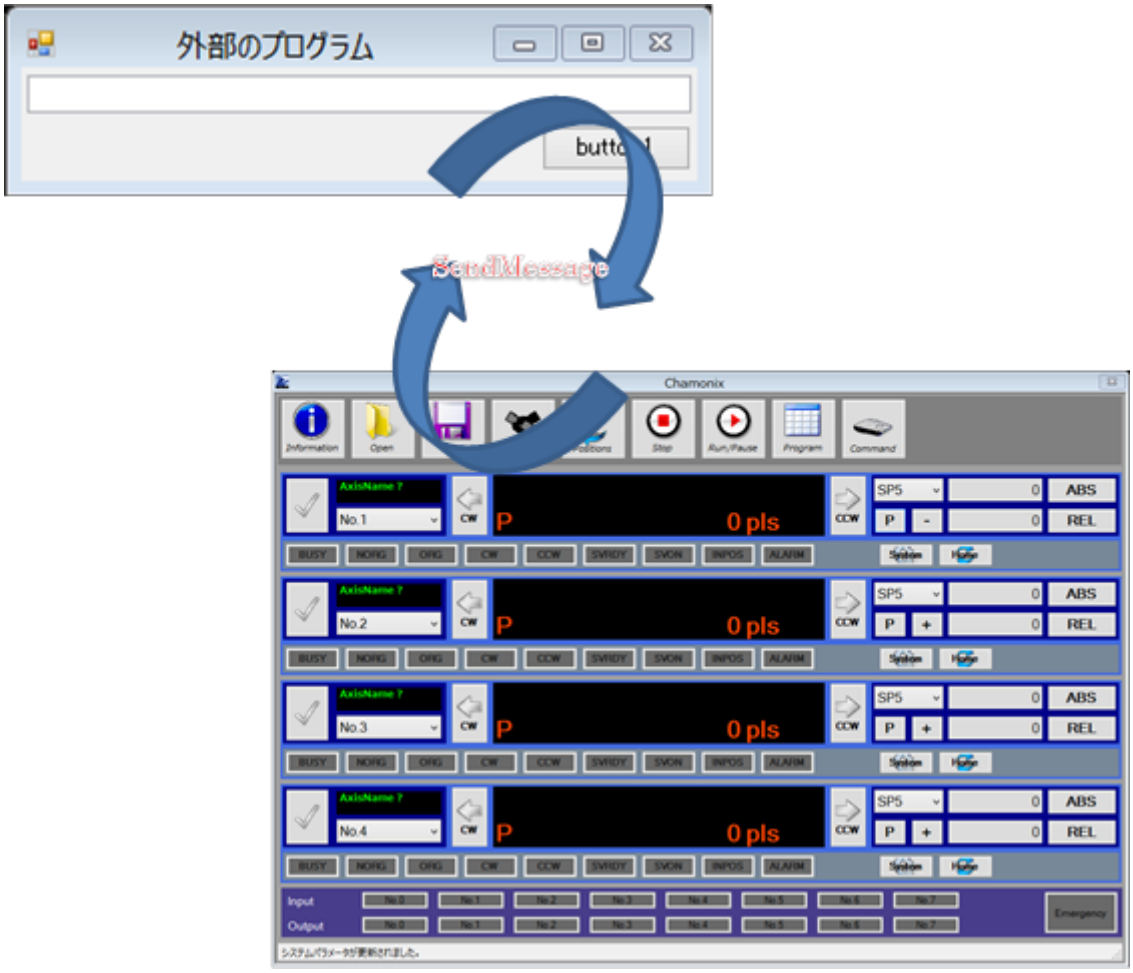
[26] "CR" is not two of Roman letters, but a control character indicating return. Control character "LF" indicates line feed in a similar way.

Time a command was sent and response from controller will appear in this window.

The application does not perform any error checking. Enter correct commands. IF an incorrect character string is sent to the controller, an error will be returned by controller's specifications. See the operation manual for the controller for more details.

16. Control from Outside Using API

16.1. Control from Outside with Windows API



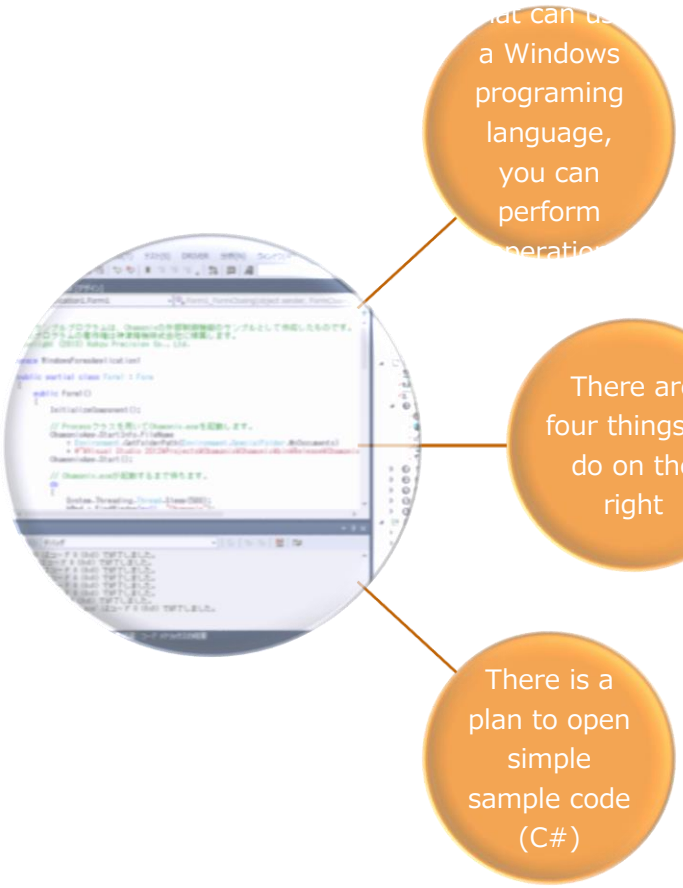
In Chamonix, you can control drives using WM_COPYDATA of Windows API from another application.

For example, refer to the Microsoft Web site for Windows API.

It supports APS, RPS, ORG, and STP remote commands.

The format is in accordance with that of ARIES remote commands.
Parameters of a response method are omitted.

16.2. Control from External Applications



- Obtain a Chamonix window handler
- Import SendMessage() of USER32.DLL
- Contain the outbound message in the COPYDATASTRUCT structure
- Send to Chamonix

Chamonix is easily controlled by adding only one code of SendMessage() in an external application.

16.3. Sending the APS Command from External Applications

```
IntPtr hWnd;

private int WM_COPYDATA = 0x004A;

USER32.dllのインポート

private void button1_Click(object sender, EventArgs e)
{
    // ChamonixのWindowハンドラを取得します。
    hWnd = FindWindow(null, "Chamonix");

    COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

    // 構造体stに値を入れます。
    copyStruct.dwData = (IntPtr)0;
    copyStruct.cbData = (uint)textBox1.Text.Length + 1;
    copyStruct.lpData = this.textBox1.Text;

    // SendMessageでChamonixにコマンドを送信します。
    SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

    //Marshal.FreeCoTaskMem(lp);
    textBox1.Text = null;
}
/// <summary>
/// SendMessageで送信する構造体です。
/// </summary>
private struct COPYDATASTRUCT
{
    public IntPtr dwData;
    public uint cbData;
    public string lpData;
}
```

Commands are passed to window procedures of Chamonix directly by using the SendMessage() function[27] of Windows API from external applications.

[27] See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the SendMessage function.

The structure of the SendMessage() function is as follows:

SendMessage(HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)	
hWnd	Enter a handle of Chamonix (main window)
Msg	WM_COPYDATA（0x004A）
wParam	Window handle to pass a message from Chamonix
lParam	Sending command (COPYDATASTRUCT[28]) lParam.dwData: 0 lParam.cbData: size of sending character string (for example, 13) lParam.lpData: Sending character string (for example, APS1/9/10000)

When you want to perform deciding position of PM1 axis with the No.9 speed table to the 10000pls position using absolute position management for Chamonix, enter 12 in cbData and APS1/9/10000 in IpParam.

Parameters of a response method are omitted.

[28]See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the COPYSTRUCT structure.

16.4. Sending the RPS Command from External Applications

```
IntPtr hWnd;

private int WM_COPYDATA = 0x004A;

USER32.dllのインポート

private void button1_Click(object sender, EventArgs e)
{
    // ChamonixのWindowハンドラを取得します。
    hWnd = FindWindow(null, "Chamonix");

    COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

    // 構造体stに値を入れます。
    copyStruct.dwData = (IntPtr)0;
    copyStruct.cbData = (uint)textBox1.Text.Length + 1;
    copyStruct.lpData = this.textBox1.Text;

    // SendMessageでChamonixにコマンドを送信します。
    SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

    //Marshal.FreeCoTaskMem(lp);
    textBox1.Text = null;
}
/// <summary>
/// SendMessageで送信する構造体です。
/// </summary>
private struct COPYDATASTRUCT
{
    public IntPtr dwData;
    public uint cbData;
    public string lpData;
}
```

Commands are passed to window procedures of Chamonix directly by using the SendMessage() function[29] of Windows API from external applications.

[29] See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the SendMessage function.

The structure of the SendMessage() function is as follows:

SendMessage(HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)	
hWnd	Enter a handle of Chamonix (main window)
Msg	WM_COPYDATA（0x004A）
wParam	Window handle to pass a message from Chamonix
lParam	Sending command (COPYDATASTRUCT[30]) lParam.dwData: 0 lParam.cbData: size of sending character string (for example, 13) lParam.lpData: Sending character string (for example, RPS1/9/100000)

When you want to perform deciding position of PM1 axis with the No.9 speed table to the 10000pls position using relative position management for Chamonix, enter 12 in cbData and RPS1/9/10000 in IpParam.

Parameters of a response method are omitted.

[30]See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the COPYSTRUCT structure.

16.5. Sending the STP Command from External Applications

```
IntPtr hWnd;

private int WM_COPYDATA = 0x004A;

USER32.dllのインポート

private void button1_Click(object sender, EventArgs e)
{
    // ChamonixのWindowハンドラを取得します。
    hWnd = FindWindow(null, "Chamonix");

    COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

    // 構造体stに値を入れます。
    copyStruct.dwData = (IntPtr)0;
    copyStruct.cbData = (uint)textBox1.Text.Length + 1;
    copyStruct.lpData = this.textBox1.Text;

    // SendMessageでChamonixにコマンドを送信します。
    SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

    //Marshal.FreeCoTaskMem(lp);
    textBox1.Text = null;
}
/// <summary>
/// SendMessageで送信する構造体です。
/// </summary>
private struct COPYDATASTRUCT
{
    public IntPtr dwData;
    public uint cbData;
    public string lpData;
}
```

Commands are passed to window procedures of Chamonix directly by using the SendMessage() function[31] of Windows API from external applications.

[31] See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the SendMessage function.

The structure of the SendMessage() function is as follows:

SendMessage(HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)	
hWnd	Enter a handle of Chamonix (main window)
Msg	WM_COPYDATA（0x004A）
wParam	Window handle to pass a message from Chamonix
lParam	Sending command (COPYDATASTRUCT[32]) lParam.dwData: 0 lParam.cbData: size of sending character string (for example, 7) lParam.lpData: Sending character string (for example, STP1/1)

When you stop the PM1 axis in an emergency for Chamonix, enter 7 in cbData and STP1/1 in ipParam.

[32]See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the COPYSTRUCT structure.

16.6. Sending the ORG Command from External Applications

```
IntPtr hWnd;

private int WM_COPYDATA = 0x004A;

USER32.dllのインポート

private void button1_Click(object sender, EventArgs e)
{
    // ChamonixのWindowハンドラを取得します。
    hWnd = FindWindow(null, "Chamonix");

    COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

    // 構造体stに値を入れます。
    copyStruct.dwData = (IntPtr)0;
    copyStruct.cbData = (uint)textBox1.Text.Length + 1;
    copyStruct.lpData = this.textBox1.Text;

    // SendMessageでChamonixにコマンドを送信します。
    SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

    //Marshal.FreeCoTaskMem(lp);
    textBox1.Text = null;
}
/// <summary>
/// SendMessageで送信する構造体です。
/// </summary>
private struct COPYDATASTRUCT
{
    public IntPtr dwData;
    public uint cbData;
    public string lpData;
}
```

Commands are passed to window procedures of Chamonix directly by using the SendMessage() function[33] of Windows API from external applications.

[33] See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the SendMessage function.

The structure of the SendMessage() function is as follows:

SendMessage(HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)	
hWnd	Enter a handle of Chamonix (main window)
Msg	WM_COPYDATA（0x004A）
wParam	Window handle to pass a message from Chamonix
lParam	Sending command (COPYDATASTRUCT[34]) lParam.dwData: 0 lParam.cbData: size of sending character string (for example, 7) lParam.lpData: Sending character string (for example, ORG1/9)

When you return the PM1 axis to origin with the No.9 speed table for Chamonix, enter 7 in cbData and ORG1/9 in ipParam.

[34]See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the COPYSTRUCT structure.

16.7. Canceling Standby of Programmable Grid from External Applications

```
IntPtr hWnd;

private int WM_COPYDATA = 0x004A;

USER32.dllのインポート

private void button1_Click(object sender, EventArgs e)
{
    // ChamonixのWindowハンドラを取得します。
    hWnd = FindWindow(null, "Chamonix");

    COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

    // 構造体stに値を入れます。
    copyStruct.dwData = (IntPtr)0;
    copyStruct.cbData = (uint)textBox1.Text.Length + 1;
    copyStruct.lpData = this.textBox1.Text;

    // SendMessageでChamonixにコマンドを送信します。
    SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

    //Marshal.FreeCoTaskMem(lp);
    textBox1.Text = null;
}
/// <summary>
/// SendMessageで送信する構造体です。
/// </summary>
private struct COPYDATASTRUCT
{
    public IntPtr dwData;
    public uint cbData;
    public string lpData;
}
```

Commands are passed to window procedures of Chamonix directly by using the SendMessage() function[35] of Windows API from external applications.

[35] See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the SendMessage function.

The structure of the SendMessage() function is as follows:

SendMessage(HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)	
hWnd	Enter a handle of Chamonix (main window)
Msg	WM_COPYDATA（0x004A）
wParam	Window handle to pass a message from Chamonix
lParam	Sending command (COPYDATASTRUCT[36]) lParam.dwData: 0 lParam.cbData: size of sending character string lParam.lpData: Sending character string

When a standby state is occurring with the SBY command during running of programmable grid for Chamonix, the standby state will be canceled if the string of Message= specifier agree with the string of IParam on the received command.

[36]See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the COPYSTRUCT structure.

16.8. Sample Program for External Control

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();

        // Processクラスを用いてChamonix.exeを起動します。
        ChamonixApp.StartInfo.FileName
            = Environment.GetFolderPath(Environment.SpecialFolder.CommonProgramFilesX86)
            + @"¥KOHZU¥Chamonix¥Chamonix.exe";
        ChamonixApp.Start();

        // Chamonix.exeが起動するまで待ちます。
        do
        {
            System.Threading.Thread.Sleep(500);
            hWnd = FindWindow(null, "Chamonix");
        }
        while (hWnd.ToInt32() == 0);

    }

    Process ChamonixApp = new Process();

    IntPtr hWnd;

    private int WM_COPYDATA = 0x004A;

    #region USER32.dllのインポート

    [DllImport("USER32.dll")]
    private static extern IntPtr SendMessage(IntPtr hWnd, int Msg, IntPtr wParam, ref COPYDATASTRUCT lParam);

    [DllImport("USER32.dll")]
    static extern IntPtr FindWindow(string lpClassName, string lpWindowName);

    #endregion

    private void button1_Click(object sender, EventArgs e)
    {
        // Chamonixのwindowハンドラを取得します。
        hWnd = FindWindow(null, "Chamonix");

        COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

        // 構造体stに値を入れます。
        copyStruct.dwData = (IntPtr)0;
        copyStruct.cbData = (uint)textBox1.Text.Length + 1;
        copyStruct.lpData = this.textBox1.Text;

        // SendMessageでChamonixにコマンドを送信します。
        SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

        //Marshal.FreeCoTaskMem(lp);
        textBox1.Text = null;
    }
    /// <summary>
    /// SendMessageで送信する構造体です。
    /// </summary>
    private struct COPYDATASTRUCT
    {
        public IntPtr dwData;
        public uint cbData;
        public string lpData;
    }

    protected override void WndProc(ref Message m)
    {
        if (m.Msg.Equals(WM_COPYDATA))
        {
            try
            {
                COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();
                copyStruct = (COPYDATASTRUCT)m.GetLParam(typeof(COPYDATASTRUCT));
                string text = copyStruct.lpData;
            }
            catch
            {
                throw;
            }
            textBox1.Text = copyStruct.lpData;
        }

        base.WndProc(ref m);
    }

    private void Form1_FormClosing(object sender, FormClosingEventArgs e)
    {
        ChamonixApp.CloseMainWindow();
    }
}
```

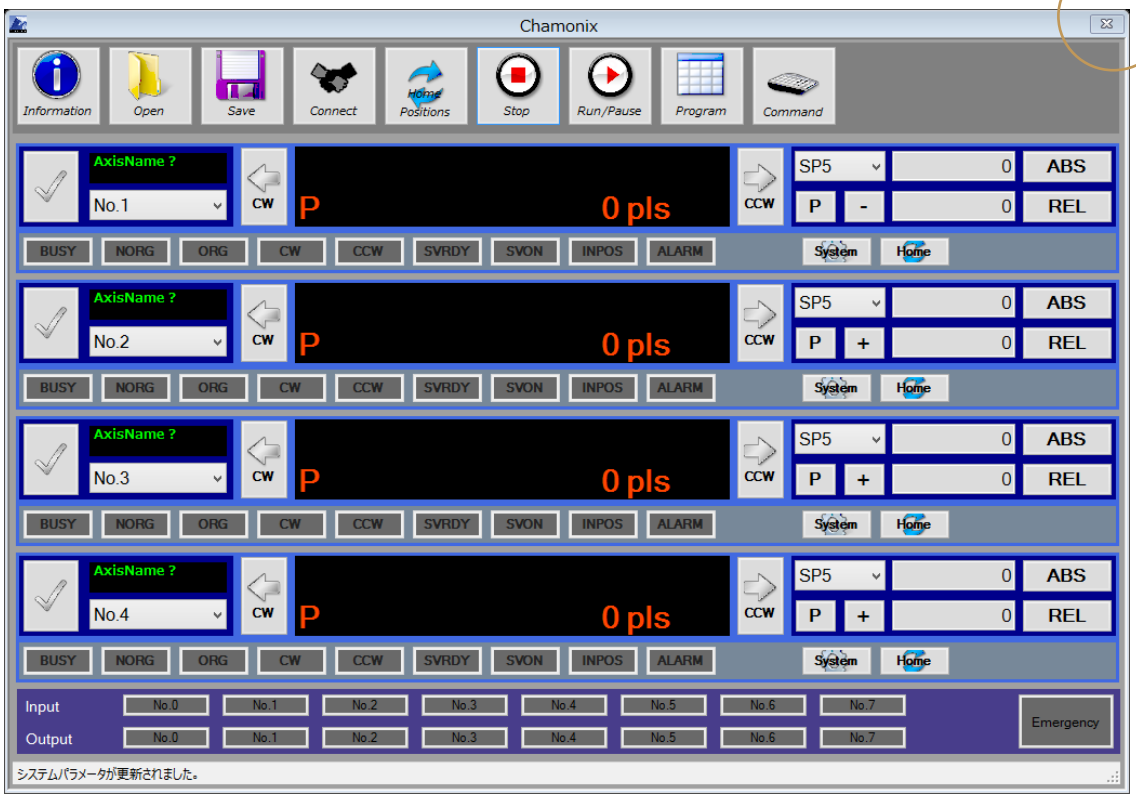
Sample code of an external application using SendMessage will be offered in C# at a Web site. The sample code is not designed to be suitable to practical use by incorporating itself. This is only an example for use of SendMessage.

Our company assumes no responsibility for customer's use of the sample code.

Copyrights in and to the sample code are owned by KOHZU Precision Co.

17. Exiting from Chamonix

17.1. Exiting from Chamonix



Click the close box to exit from Chamonix.

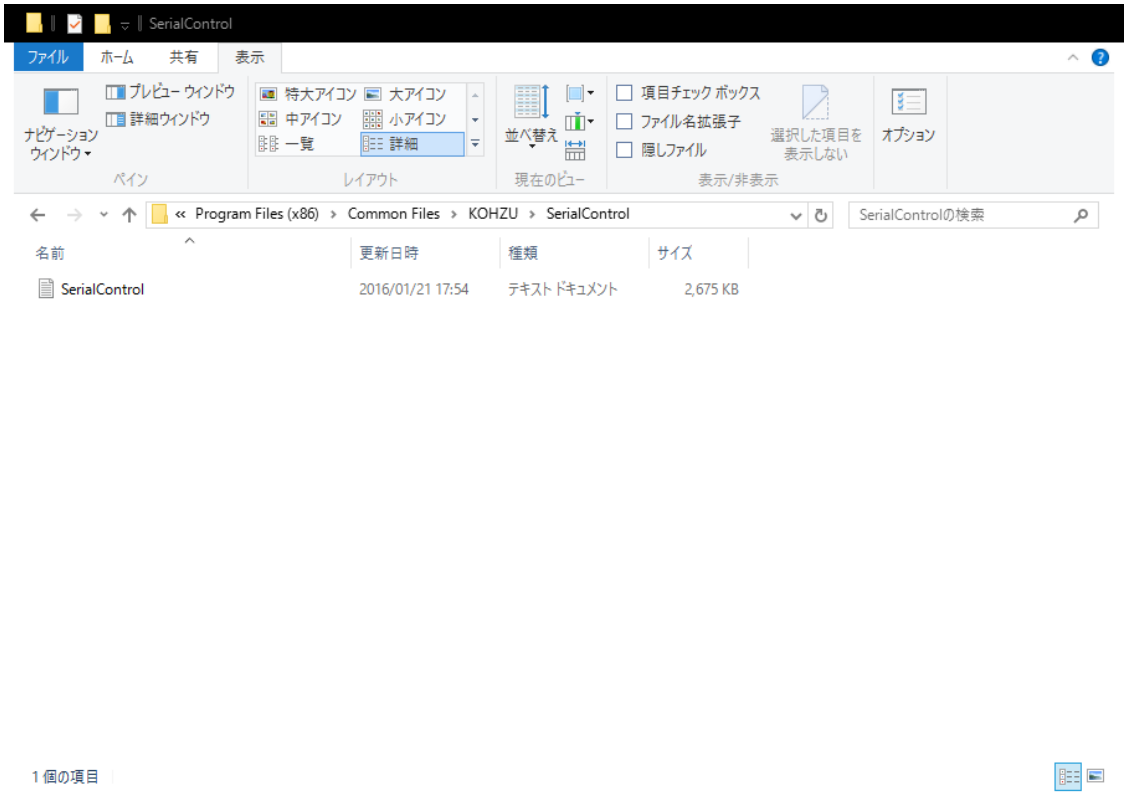
Parameters at time of the exiting are written into Config.xml of program Data in exiting. The COM port, baud rate, conversion coefficient, name, and others are saved. System parameters of the controller give setting values of the controller priority over saved values. It starts with conditions saved in this Config.xml unless starting from a start-up file.

The STP command is sent to all the axes of the controller when Chamonix exits. When Chamonix is shut down during control of drive with Chamonix, the drive is meant to be canceled.

When CRUX have been connected and this application is closed, it sends SAV commands. Please do not power off till finished to write for ROM.

18. Others

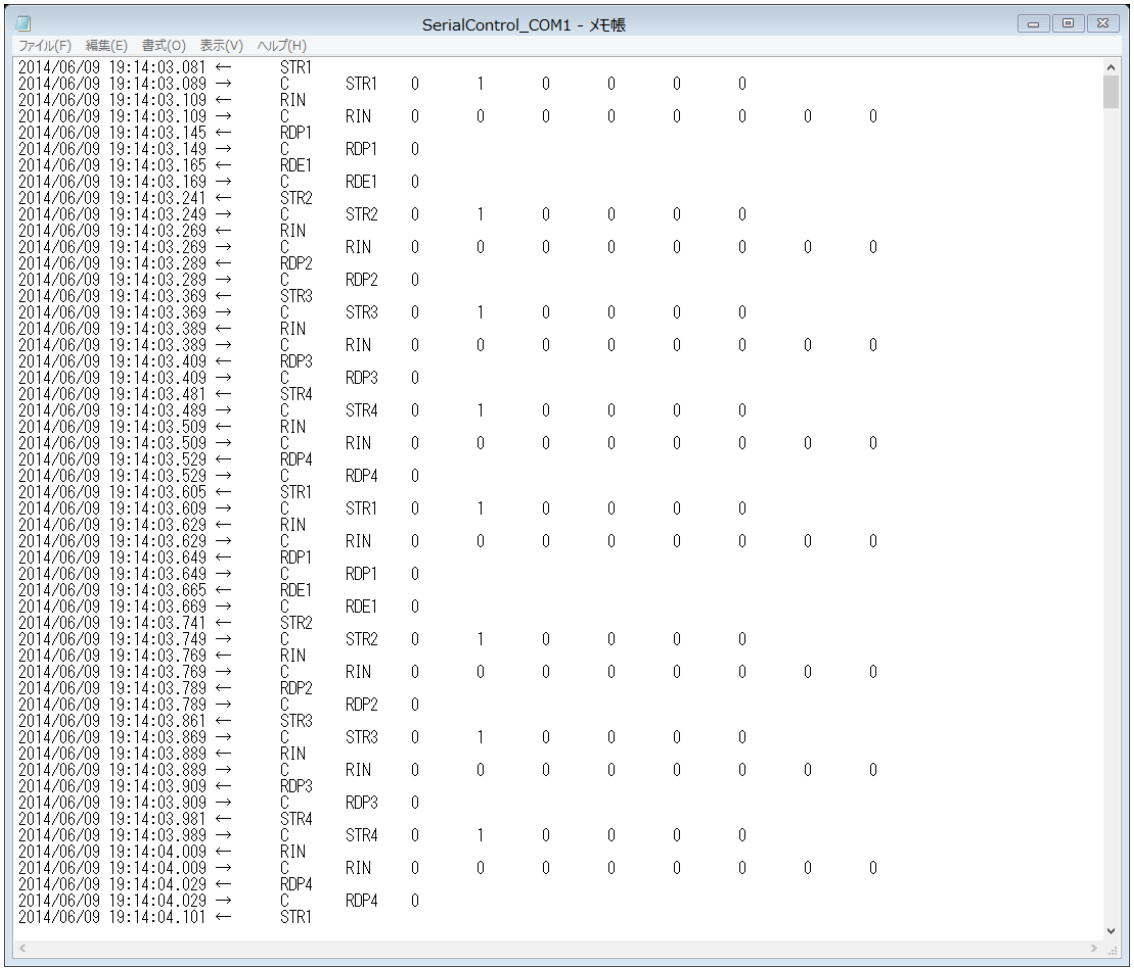
18.1. Folder to Save Communication Logs



Communication logs for the controller remain in C:¥ProgramData¥KOHZU ¥SerialControl [37] folder.

[37] In the case of the before than Ver.1.1.015, the config file is saved at C:¥Program Files¥Common Files¥KOHZU¥SerialControl l.

18.2. Checking Communication Logs



This log file contains records of transmission to and reception from a controller.

Transmission and reception time is recorded in milliseconds. The time is not accurate because of using the system clock.

It is expected that troubleshooting will become more efficient by analyzing this file when a trouble is arising.

Revision History

Date	Version	Others
June 30, 2014	Rev.1.0.0	M.O.
October 30, 2014	Rev.1.0.1	M.O.
December 25, 2014	Rev.1.0.2	M.O.
January 21, 2016	Rev.1.1.0	M.O.
December 25, 2017	Rev.1.2.1	M.O.
June 25, 2019	Rev.1.2.2	M.O
Feb 4, 2020	Rev.1.2.3	M.O

記録欄

ご購入日　　年　　月　　日

購入先

担当者　　電話番号

製造番号

特　記

・　・	
・　・	
・　・	
・　・	
・　・	
・　・	
・　・	
・　・	

KOHZU

Kohzu Precision Co., Ltd.

Headquarters

215-8521

2-6-15 Kurihira Asao-ku, Kawasaki City Kanagawa

Tel: **+81-44-981-2131**

Fax: **044-981-2181**

E-mail: sale@kohzu.co.jp

Web Site: http://www.kohzu.co.jp/

Osaka branch

532-0004

2-7-38 Nishi Miyahara Yodogawa-ku, Osaka City

Shin Osaka Nishiura Bldg. 202

Tel: **06-6398-6610**

Fax: **06-6398-6620**