OMRON

Programmable Multi-Axis Controller

Startup Guide for OMRON G5-Series Servo Drive

CK3M-CPU□
CK3W-AX□

SYSMAC
always in control

R193-E1-02

NOTE

- All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, mechanical, electronic, photocopying, recording, or otherwise, without the prior written permission of OMRON.
- No patent liability is assumed with respect to the use of the information contained herein. Moreover, because OMRON is constantly striving to improve its high-quality products, the information contained in this manual is subject to change without notice.
- Every precaution has been taken in the preparation of this manual. Nevertheless, OMRON assumes no responsibility for errors or omissions.
  Neither is any liability assumed for damages resulting from the use of the information contained in this publication.

Trademarks

- Microsoft, Windows, Windows Vista, Excel, and Visual Basic are either registered trademarks or trademarks of Microsoft Corporation in the United States and other countries.
- EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

Other company names and product names in this document are the trademarks or registered trademarks of their respective companies.

Copyrights

- Microsoft product screen shots reprinted with permission from Microsoft Corporation.
- This product incorporates certain third party software. The license and copyright information associated with this software is available at http://www.fa.omron.co.jp/nj_info_e/.

CONTENTS

Related Manuals **4**

Revision History **5**

Terms and Definitions **6**

Precautions **7**

Sections in this Manual **9**

Section 1 Summary of Materials

1-1 Summary of Materials ..... **1 - 2**

Section 2 Target Equipment and Device Configuration

2-1 Target Equipment ..... **2 - 2**

2-2 Device Configuration ..... **2 - 3**

Section 3 Analog I/F Connection Procedure

3-1 Work Flow ..... **3 - 2**

3-2 Controller Setting Preparations ..... **3 - 4**

3-2-1 Creation of a New Project ..... 3 - 4

3-2-2 Controller Initial Setting ..... 3 - 5

3-3 Various Equipment Wiring ..... **3 - 7**

3-3-1 Axial Interface Unit and Driver Wiring ..... 3 - 7

3-3-2 Wiring for Switch Mode Power Supply and Driver ..... 3 - 8

3-4 Various Controller Settings ..... **3 - 9**

3-5 Various Servo Drive Settings ..... **3 - 11**

3-6 Checking Operation ..... **3 - 12**

3-7 Motor Tuning ..... **3 - 13**

3-7-1 Open Loop Test ..... 3 - 13

3-7-2 Bandwidth Automatic Setting ..... 3 - 13

3-7-3 Manual Correction of Bandwidth ..... 3 - 14

3-7-4 Feed-Forward Value Setting ..... 3 - 16

3-7-5 Creation of Tuning Parameter Project ..... 3 - 18

3-8 Absolute Encoder System Home Setting ..... **3 - 19**

3-8-1 Absolute Encoder Setup ..... 3 - 19

3-8-2 Read Absolute Encoder Position ..... 3 - 19

3-8-3 Execute Homing ..... 3 - 22

3-9 Operations Check Based on Motion Program ..... **3 - 23**

3-9-1 Creation of Operations Check Program ..... 3 - 23

3-9-2 Transferring Project Data and Checking the Operation ..... 3 - 25

Index

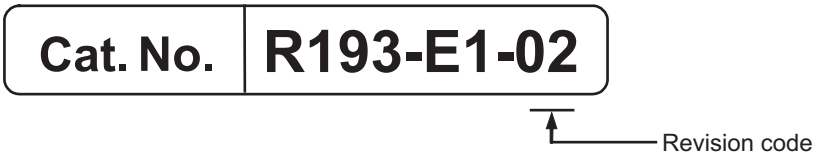
Related Manuals

To safely utilize the system, obtain a manual or user's guide for each device and piece of equipment, confirm their content, including "Safety Precautions", "Precautions for Safe Use", and other precautions related to safety, and then proceed with use.
The manuals for OMRON Corporation (hereafter, "OMRON") and Delta Tau Data Systems Inc. (hereafter "DT") are as shown below.

Manufacturer	Cat. No.	Model	Manual Name
OMRON	O036	CK3M-☐ CK3W-☐	Programmable Multi-Axis Controller Hardware User's Manual
DT	O014	---	Power PMAC User's Manual
DT	O015	---	Power PMAC Software Reference Manual
DT	O016	---	Power PMAC IDE User's Manual

Revision History

A manual revision code appears as a suffix to the catalog number on the front and back covers.



Revision code	Date	Revised content
01	April 2018	Original production
02	July 2018	Corrected mistakes.

Terms and Definitions

Terms	Descriptions and Definitions
Power PMAC IDE	Computer software that is used to configure the Controller, create user programs, and perform monitoring. PMAC is an abbreviation for Programmable Multi-Axis Controller.

Precautions

- For actual system construction, check the specifications for each device and piece of equipment that makes up the system, use a method with sufficient margin for ratings and performance, and adopt safety circuits and other safety measures to minimize risks even if a breakdown occurs.
- To safely utilize the system, obtain a manual or user's guide for each device and piece of equipment that makes up the system, confirm their content, including "Safety Precautions", "Precautions for Safe Use", and other precautions related to safety, and then proceed with use.
- The customer must check all regulations, laws, and rules that are applicable to the system themselves.
- Copying, duplication, or redistribution of part or all of these materials without the permission of OMRON Corporation is prohibited.
- The content listed in these materials is valid as of July 2018.
  The content listed in these materials may be changed without notice for purposes of improvement.

The marks used in these materials are defined as follows.

Precautions for Correct Use

Precautions on what to do and what not to do to ensure correct operation and performance.

Additional Information

Additional information to read as required.
This information is provided to increase understanding and make operation easier.

Sections in this Manual

1 Summary of Materials

2 Target Equipment and Device Configuration

3 Analog I/F Connection Procedure

1

Summary of Materials

This section lists a summary of these materials.

1-1 Summary of Materials ..... 1 - 2

1-1 Summary of Materials

These materials are a summary of the procedures and check methods for operations of the OMRON Servo Drive (G5-series Servo Drive, hereafter called Servo Drive) in the motion program of the OMRON Programmable Multi-Axis Controller CK3M-□ (hereafter, Controller).
In these materials, the general name for the Servo Drive together with the connected Servo Motor is the "motion control device". In addition, the Servo Drive is called the "slave", based on the description content.

By understanding the setting content and setting procedure points described in *Section 3 Analog I/F Connection Procedure* on page 3 - 1, you can use Analog I/F and torque control for a closed loop, enabling operation of the motion control device. In these materials, the motion program is used for the operations check.

2

Target Equipment and Device Configuration

This section lists the target equipment and system configurations for connections in these materials.

2-1 Target Equipment ..... 2 - 2
2-2 Device Configuration ..... 2 - 3

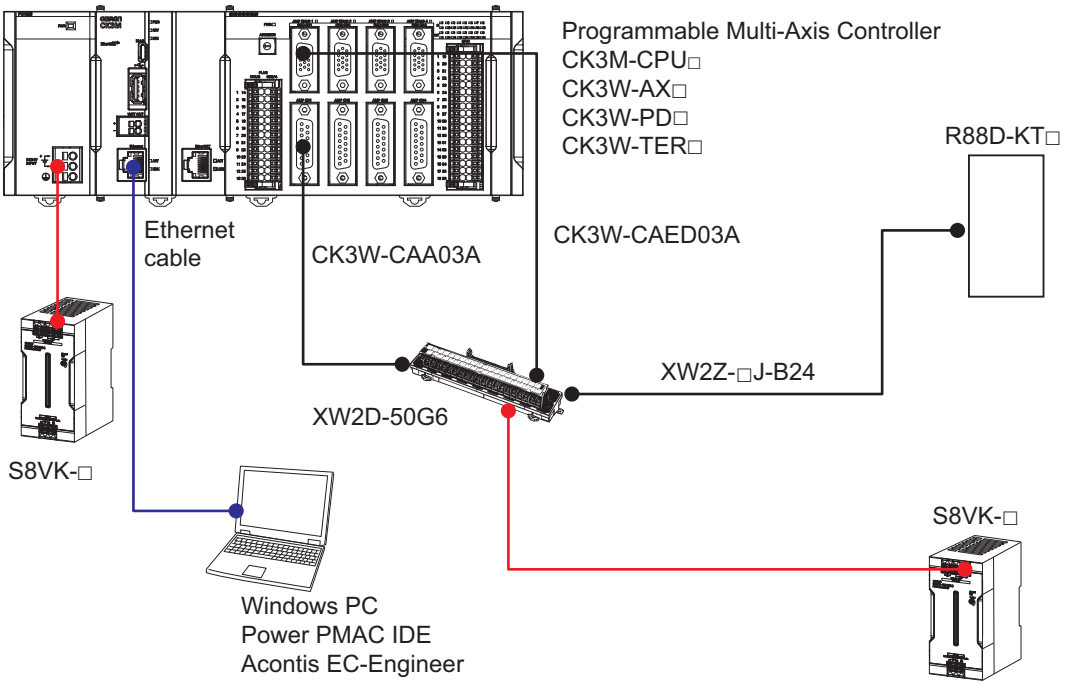
2-1 Target Equipment

The target equipment for connection is shown below.

Manufacturer	Name	Model
OMRON	Programmable Multi-Axis Controller CPU Unit	CK3M-CPU□
OMRON	Programmable Multi-Axis Controller Axial Interface Unit	CK3W-AX□
OMRON	Programmable Multi-Axis Controller Power Supply Unit	CK3W-PD□
OMRON	Programmable Multi-Axis Controller End Cover	CK3W-TER□
OMRON	Switch Mode Power Supply	S8VK-□
OMRON	AC Servo Drive	R88D-KT□
OMRON	Amplifier Cable	CK3W-CAA03A
OMRON	Encoder Cable	CK3W-CAED03A
OMRON	Connector-Terminal Block Conversion Unit	XW2D-50G6
OMRON	Control Cable	XW2Z-□J-B24

2-2 Device Configuration

The configuration devices for recreating the connection procedures in these materials are shown below.



Manufacturer	Name	Model	Version
OMRON	Programmable Multi-Axis Controller CPU Unit	CK3M-CPU☐	—
OMRON	Programmable Multi-Axis Controller Axial Interface Unit	CK3W-AX☐	—
OMRON	Programmable Multi-Axis Controller Power Supply Unit	CK3W-PD☐	—
OMRON	Programmable Multi-Axis Controller End Cover	CK3W-TER☐	—
OMRON	Switch Mode Power Supply	S8VK-☐	—
OMRON	AC Servo Drive	R88D-KT☐	—
OMRON	Amplifier Cable	CK3W-CAA03A	—
OMRON	Encoder Cable	CK3W-CAED03A	—
OMRON	Connector-Terminal Block Conversion Unit	XW2D-50G6	—
OMRON	Control Cable	XW2Z-☐J-B24	—
—	Windows PC		—
DT	Power PMAC Setting Tool	Power PMAC IDE	4.0 or higher
OMRON	Servo Drive Setting Tool	CX-Drive	—

3

Analog I/F Connection Procedure

This section describes the procedures for connecting the Controller and Servo Drive, and operating the motion control equipment.

3-1 Work Flow ..... 3 - 2

3-2 Controller Setting Preparations. ..... 3 - 4
3-2-1 Creation of a New Project ..... 3 - 4
3-2-2 Controller Initial Setting ..... 3 - 5

3-3 Various Equipment Wiring ..... 3 - 7
3-3-1 Axial Interface Unit and Driver Wiring ..... 3 - 7
3-3-2 Wiring for Switch Mode Power Supply and Driver ..... 3 - 8

3-4 Various Controller Settings ..... 3 - 9

3-5 Various Servo Drive Settings ..... 3 - 11

3-6 Checking Operation ..... 3 - 12

3-7 Motor Tuning ..... 3 - 13
3-7-1 Open Loop Test ..... 3 - 13
3-7-2 Bandwidth Automatic Setting ..... 3 - 13
3-7-3 Manual Correction of Bandwidth ..... 3 - 14
3-7-4 Feed-Forward Value Setting ..... 3 - 16
3-7-5 Creation of Tuning Parameter Project ..... 3 - 18

3-8 Absolute Encoder System Home Setting ..... 3 - 19
3-8-1 Absolute Encoder Setup ..... 3 - 19
3-8-2 Read Absolute Encoder Position ..... 3 - 19
3-8-3 Execute Homing ..... 3 - 22

3-9 Operations Check Based on Motion Program ..... 3 - 23
3-9-1 Creation of Operations Check Program ..... 3 - 23
3-9-2 Transferring Project Data and Checking the Operation ..... 3 - 25

3-1 Work Flow

The procedures for connecting the Controller and Servo Drive, and operating the motion control equipment, are shown below.

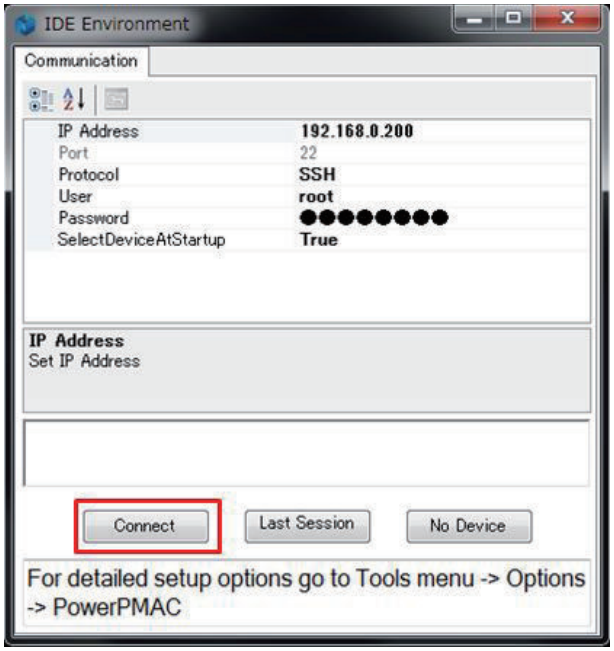
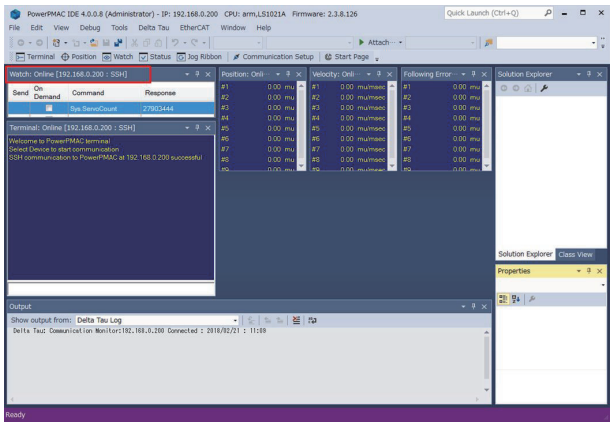
***3-2 Controller Setting Preparations.* on page 3 - 4**	Perform the Controller setting preparations.
▼	
3-2-1 Creation of a New Project on page 3 - 4	
▼	
3-2-2 Controller Initial Setting on page 3 - 5	
▽	
***3-3 Various Equipment Wiring* on page 3 - 7**	Perform wiring for each piece of equipment.
▼	
3-3-1 Axial Interface Unit and Driver Wiring on page 3 - 7	
▼	
3-3-2 Wiring for Switch Mode Power Supply and Driver on page 3 - 8	
▽	
***3-4 Various Controller Settings* on page 3 - 9**	Perform the Controller settings.
▽	
***3-5 Various Servo Drive Settings* on page 3 - 11**	Perform the Servo Drive settings.
▽	
***3-6 Checking Operation* on page 3 - 12**	Check that the settings up to here are correct.
▽	
***3-7 Motor Tuning* on page 3 - 13**	Perform motor tuning.
▼	
3-7-1 Open Loop Test on page 3 - 13	
▼	
3-7-2 Bandwidth Automatic Setting on page 3 - 13	
▼	
3-7-3 Manual Correction of Bandwidth on page 3 - 14	
▼	
3-7-4 Feed-Forward Value Setting on page 3 - 16	
▼	
3-7-5 Creation of Tuning Parameter Project on page 3 - 18	
▼	
***3-8 Absolute Encoder System Home Setting* on page 3 - 19**	Perform homing.
▼	
***3-8-1 Absolute Encoder Setup* on page 3 - 19**	Create a motion program, and perform an operations check.
▼	
3-8-2 Read Absolute Encoder Position on page 3 - 19	

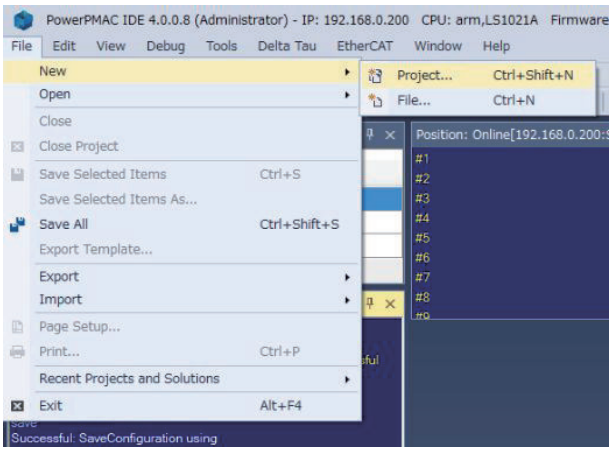
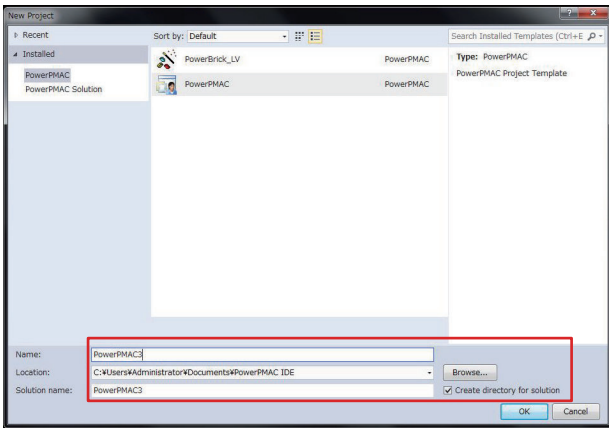
▼	
3-8-3 Execute Homing on page 3 - 22	

3-2 Controller Setting Preparations.

Perform the Controller setting preparations.
Install the Power PMAC IDE on the PC beforehand.

3-2-1 Creation of a New Project

1	Connect the Controller and computer with an Ethernet cable.	
2	Turn ON the power supply to the Controller.	
3	Start up Power PMAC IDE. • If a dialog for checking access rights is displayed at the time of startup, select the option for starting up.	
4	The Communication screen is displayed, so specify the IP address of the Controller to be connected to, and click **Connect**. • The default IP address for the Controller is 192.168.0.200. • If necessary, change the Windows IP address to 192.168.0.X.	
5	Power PMAC IDE starts up, and the Controller will come online.	

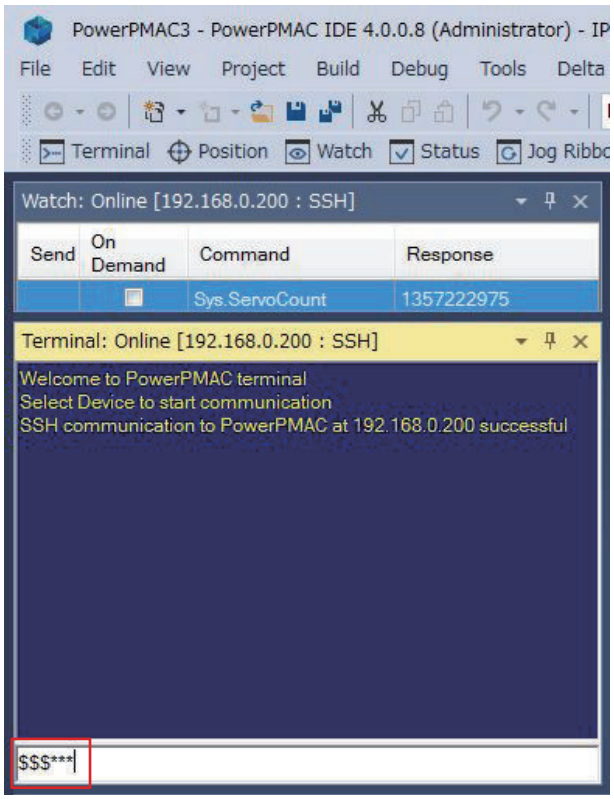
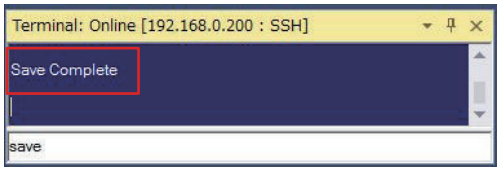
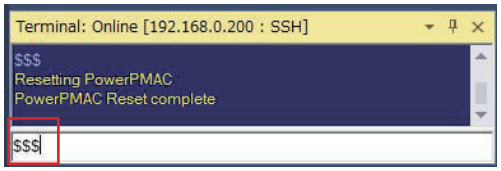
6	From the **File** menu, select **New**, **Project**.	
7	Input the desired project name and the save destination, and select **OK**.	

3-2-2 Controller Initial Setting

Perform the initial settings for the Controller.

Precautions for Correct Use

Since all memory is cleared by the initial settings, be sure to save any data remaining in the Controller that you may need.

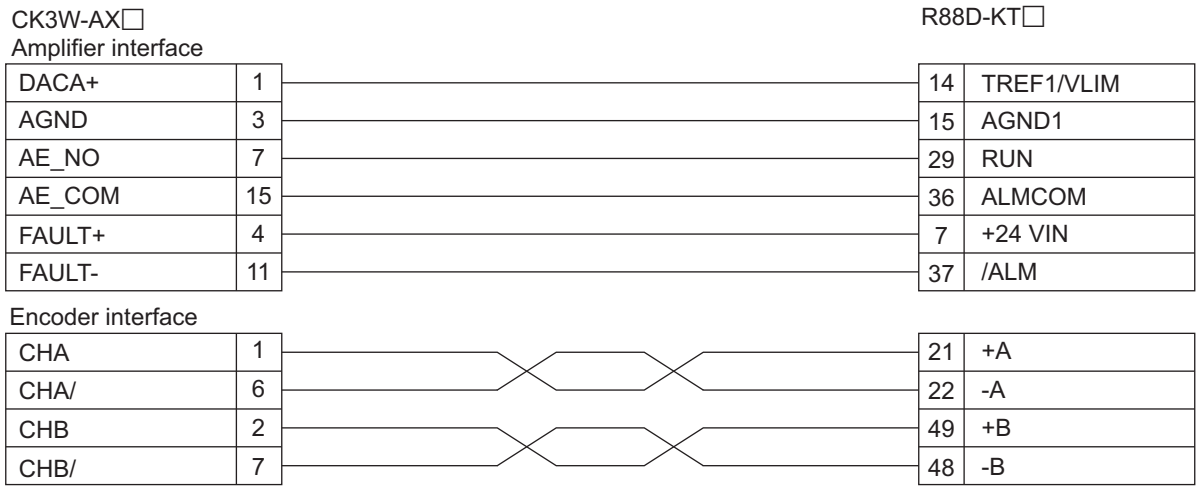
1	Type the **$$$***** command from the Terminal, and set the Controller to the factory default state.	
2	Type the **save** command in the Power PMAC IDE Terminal. When the save is completed, "Save Complete" is displayed in the Terminal.	
3	Type the **$$$** command in the Power PMAC IDE Terminal.	

3-3 Various Equipment Wiring

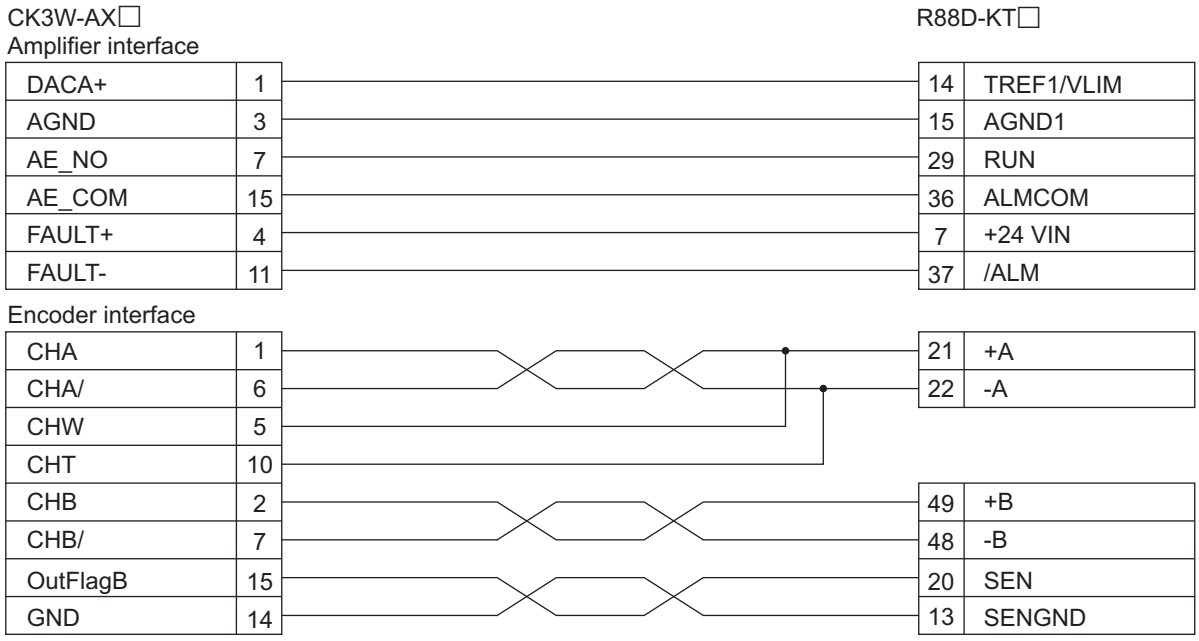
3-3-1 Axial Interface Unit and Driver Wiring

Perform wiring for the Axial Interface Unit and Servo Drive in accordance with the wiring diagram below.

If Using an Incremental Encoder



If Using an Absolute Encoder



The cables and units used are shown in the table below.

Manufacturer	Name	Model
OMRON	Amplifier Cable	CK3W-CAA03A

Manufacturer	Name	Model
OMRON	Encoder Cable	CK3W-CAED03A
OMRON	Connector-Terminal Block Conversion Unit	XW2D-50G6

3-3-2 Wiring for Switch Mode Power Supply and Driver

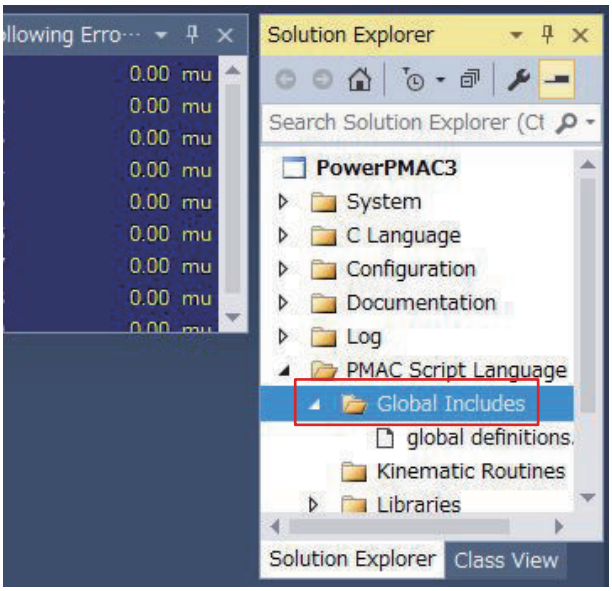
Perform the wiring for the switch mode power supply and the Servo Drive as shown below.

Switch Mode Power Supply	Servo Drive
+24 V	7 : +24 VIN
GND	36 : ALMCOM

3-4 Various Controller Settings

Perform the settings for connecting the Controller to the Servo Drive.

1 Open Global Definitions.pmh under **PMAC Script Language** — **Global Includes** in the Solution Explorer.



2 Write the text on the right to Global Definitions.pmh.

- For CK3W-AX1111□

```
Sys.WpKey = $AAAAAAAA

Gate3[0].PhaseFreq = 64000
Gate3[0].ServoClockDiv = 3
Gate3[0].Chan[0].PwmFreqMult = 5
Sys.ServoPeriod= 1/16
Sys.PhaseOverServoPeriod = 1/4
Motor[1].ServoCtrl = 1
Gate3[0].Chan[0].OutputMode = 0
Gate3[0].Chan[0].PackOutData = 0
Motor[1].pDac = Gate3[0].Chan[0].Pwm[0].a
Motor[1].pLimits = 0
Motor[1].AmpFaultLevel = 1
```

- For CK3W-AX1212□

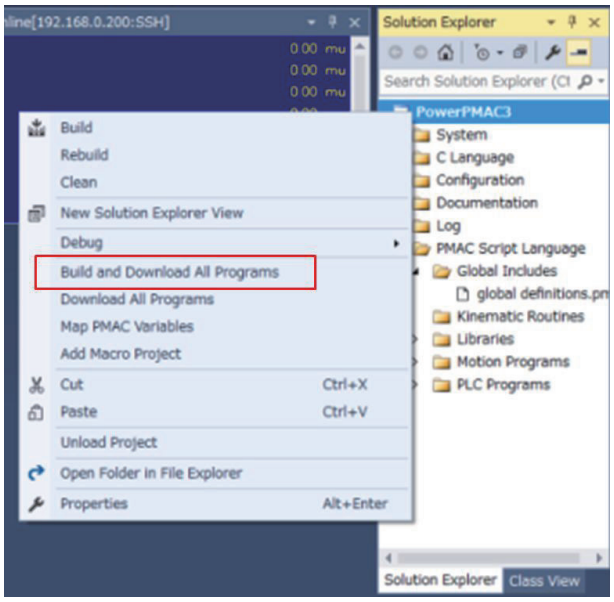
```
Sys.WpKey = $AAAAAAAA

Gate3[0].PhaseFreq = 64000
Gate3[0].ServoClockDiv = 3
Gate3[0].Chan[0].PwmFreqMult = 5
Sys.ServoPeriod = 1/16
Sys.PhaseOverServoPeriod = 1/4
Motor[1].ServoCtrl = 1
Gate3[0].Chan[0].OutputMode = 7
Gate3[0].Chan[0].PackOutData = 0
Motor[1].pDac = Gate3[0].Chan[0].Dac[0].a
Motor[1].pLimits = 0
Motor[1].AmpFaultLevel = 1
```

3 If using an absolute encoder, add the settings on the right to the bottom of the Global Definitions.pmh file.

```
Gate3[0].EncClockDiv = 3
Gate3[0].SerialEncCtrl = $82230005
Gate3[0].Chan[0].SerialEncCmd = $13000
Gate3[0].Chan[0].SerialEncEna = 1
Gate3[0].Chan[0].OutFlagD = 0
```

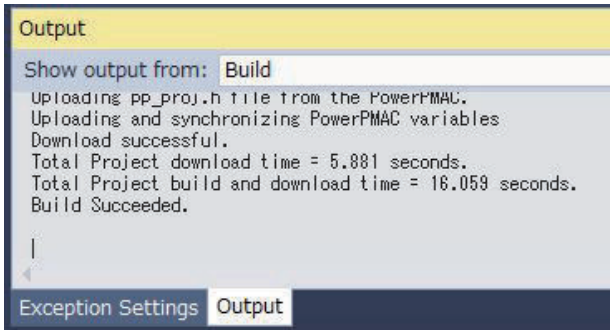
4 Downloading the project

Right click on the **Solution Explorer** project name at the upper right of the IDE screen, select **Build and Download All Programs**, and execute Build & Download.

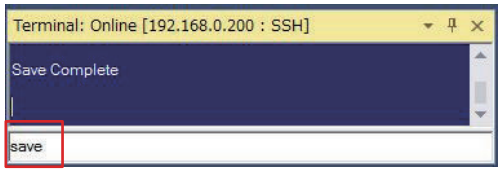


5 Make sure that there are no errors in the Output Window.

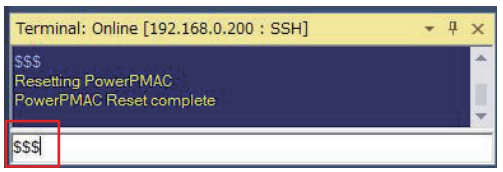
- If the transfer failed, check the content of the error in the Output Window. If there is a program error, review the program.



6 Type the **save** command in the Power PMAC IDE Terminal.
When the save is completed, "Save Complete" is displayed in the Terminal.



7 Type the **$$$** command in the Power PMAC IDE Terminal.



Precautions for Correct Use

Do not specify the servo cycle Sys.ServoPeriod setting as a decimal but rather in fraction form, as shown below. If this is not correctly set, there is a possibility that you may not be able to obtain synchronization with the Controller and EtherCAT driver.
(Example) At servo clock 12 kHz
Correct: Sys.ServoPeriod = 1/12
Incorrect: Sys.ServoPeriod = 0.083333

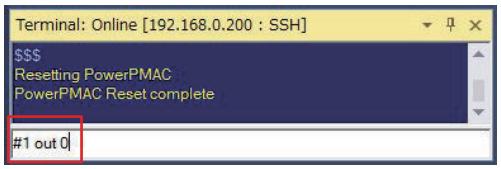
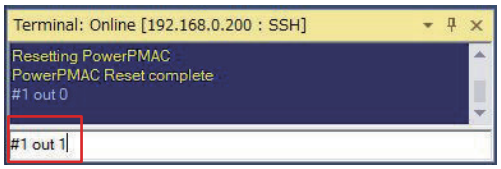
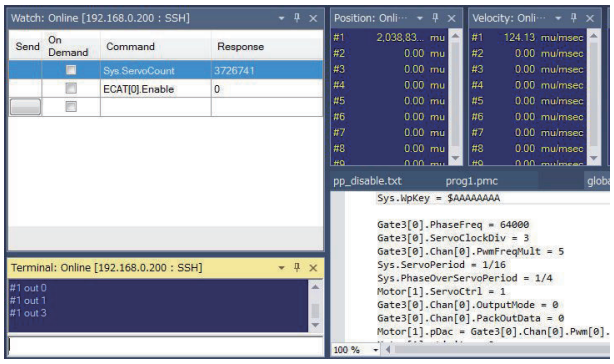
3-5 Various Servo Drive Settings

Use the CX-Drive to perform the R88D-KT□ setting. Change the drive parameters as shown in the table below. (Set parameters other than those shown in the table below to the factory settings.) For the R88D-KT□ operations method, refer to the attached manual.

No.	Name	After change
Pn001	Control mode selection	2: Torque control (analog command)
Pn319	Torque command scale	10
Pn321	Speed limit value setting	6000

3-6 Checking Operation

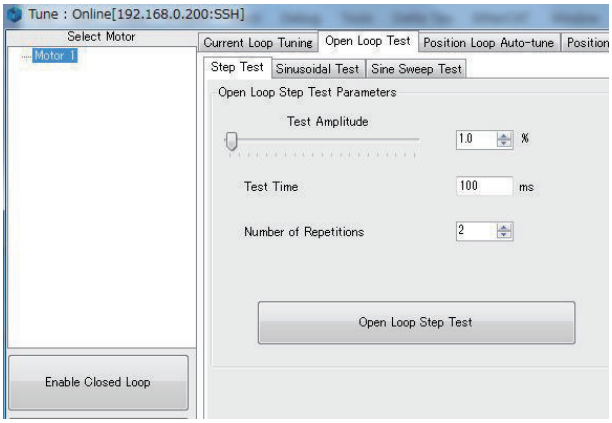
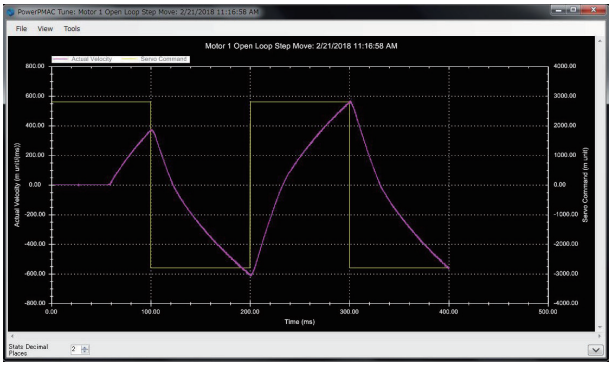
Check whether the settings up to here are correct.

1	Type the **#1 out0** command from the Terminal. At this time, check that the motor has the servo ON. • When using the absolute encoder, type **Gate3[0].Chan[0].OutFlagB=1** from the Terminal beforehand.	
2	Type the **#1 out1** command from the Terminal.	
3	Make sure that the motor is rotating. In addition, check that the Watch window **Position** value is increasing in the positive direction. • If the motor does not rotate even after typing the **#1 out1** command, input **#1 out2**, **#1 out3**, or another large value.	

3-7 Motor Tuning

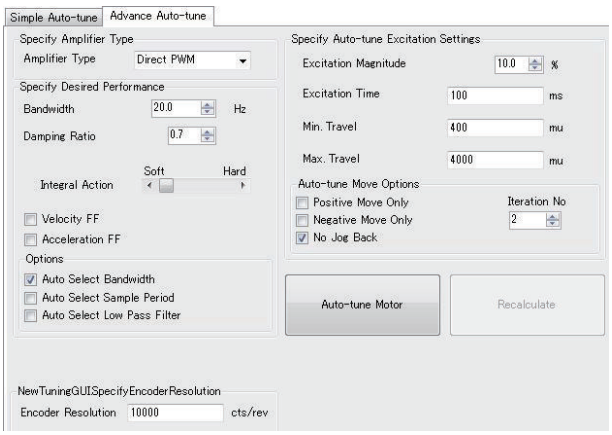
3-7-1 Open Loop Test

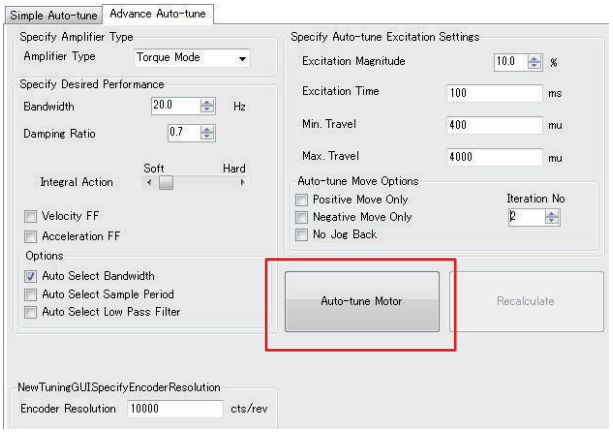
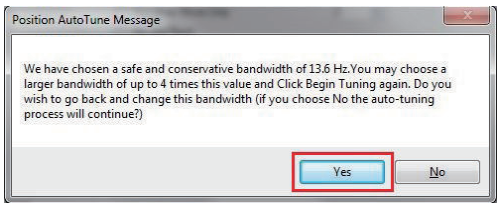
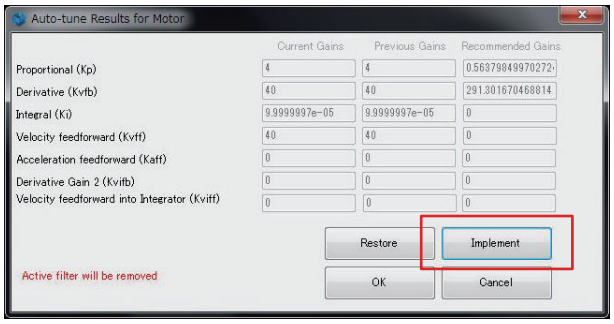
Operate the motor in an open loop, and check that each type of setting is correct.

1	Open the Tune screen on the right from the **Delta Tau** → **Tools** menu, and select **Open Loop Test** → **Step Test**. • When using the absolute encoder, input **Gate3[0].Chan[0].OutFlagB=1** from the Terminal before implementing tuning.	
2	Set the tuning parameters on the right.	**Test Amplitude**: 1.0% (If the motor is not rotating, set a large value.) **Test Time**: 100ms **Number Of Repetition**: 2
3	Click **Open Loop Step Test**, and check that the motor is performing reciprocating operation.	

3-7-2 Bandwidth Automatic Setting

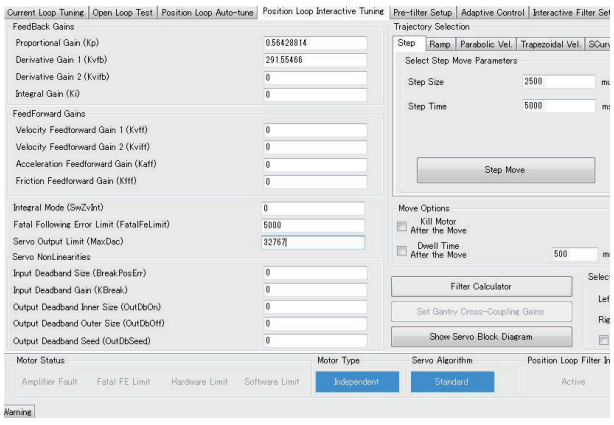
Use the Power PMAC IDE auto-tuning function to automatically set the servo loop bandwidth.

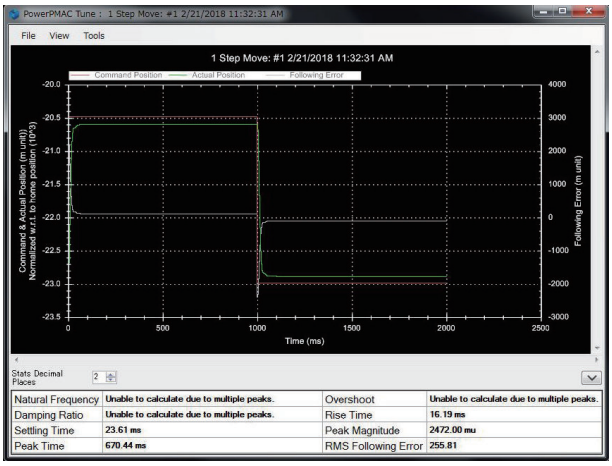
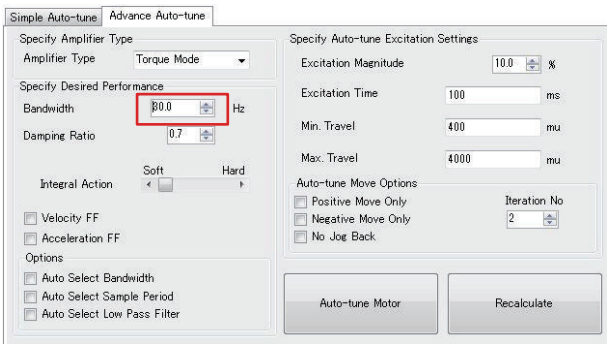
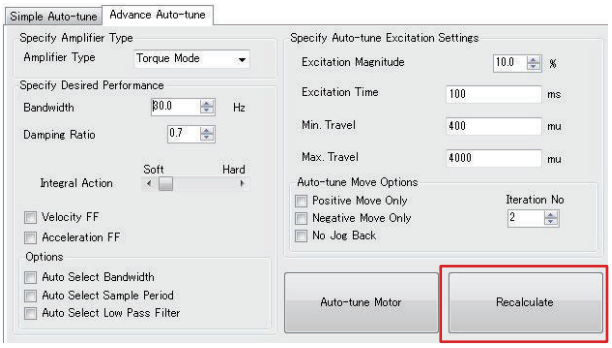
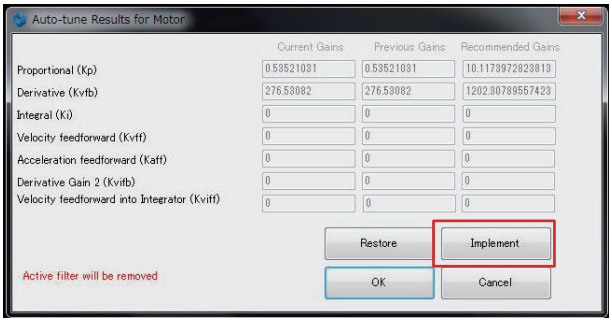
1	Select **Position Loop Auto Tune** → **Advance Auto-tune**.	

2	Set the tuning parameters on the right. • **Encoder Resolution** is determined by the resolution of the servomotor encoder being used, and by the electronic gear ratio of the Servo Drive. Set the value for the output pulse number per one motor rotation.	**Amplifier Type**: Torque Mode **Auto Select Bandwidth**: Check **Encoder Resolution**: 10000 **Excitation Magnitude**: 1% (Select the value rotated in the open loop.) **Iteration No.**: 2
3	Click **Auto-tune Motor**.	
4	If the message on the right appears, click **Yes**.	
5	If the screen on the right appears, click **Implement**.	

3-7-3 Manual Correction of Bandwidth

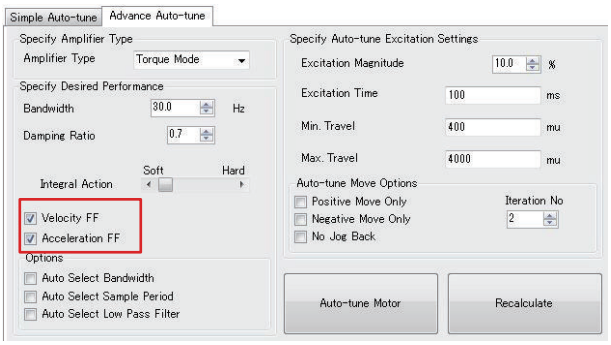
While monitoring the stepwise response, select the most suitable bandwidth.

1	Select **Position Loop Interactive Tuning**.	

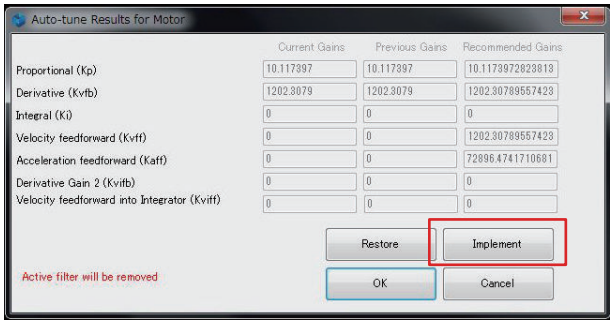
2	Set the tuning parameters on the right.	**Step Size**: 2500 **Fatal Following Error**: 5000 **Servo Output Limit**: 32767
3	Click **Step Move**, and check the step-wise response.	
4	If the target position has not been reached, return to the **Advance Auto-tune** screen, and set an even larger value for the Bandwidth.	
5	Click **Recalculate**.	
6	If the screen on the right appears, click **Implement**.	
7	Return to Step 1. Repeat until the desired responsiveness is obtained.	

3-7-4 Feed-Forward Value Setting

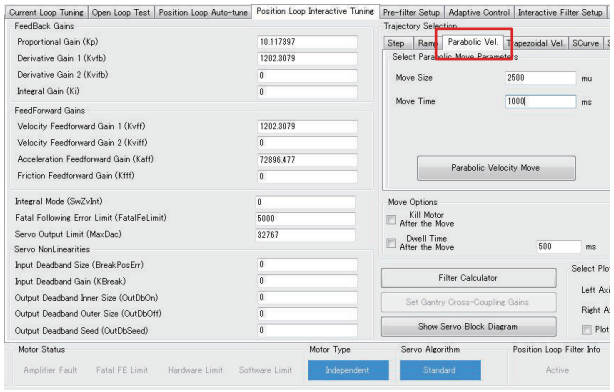
1 Select **Position Loop Auto Tune** → **Advance Auto-tune**, and insert checks into **Velocity FF** and **Acceleration FF**.



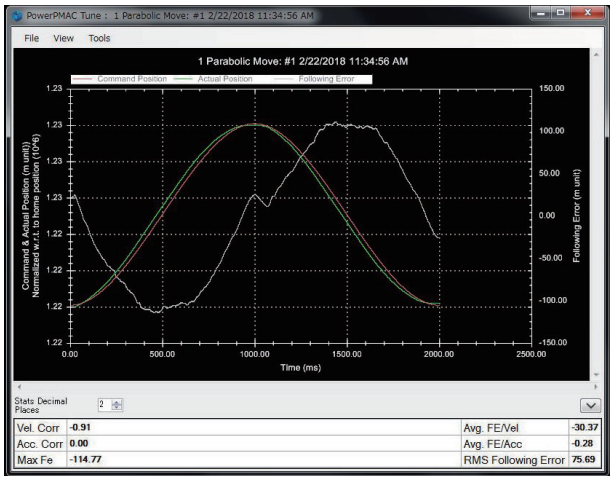
2 Click **Recalculate**, and click **Implement** in the pop-up window.



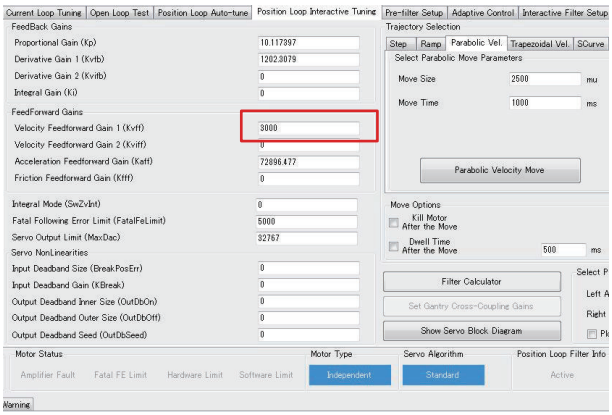
3 Select **Parabolic Vel.**. Use the same value for Move Size and Move Time.



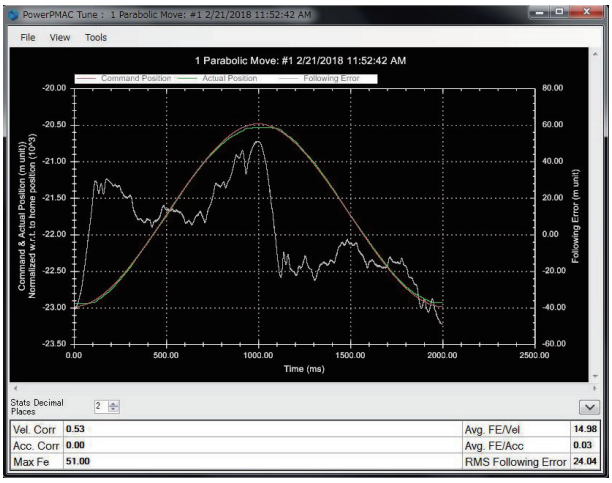
4 Click **Parabolic Velocity Move**.



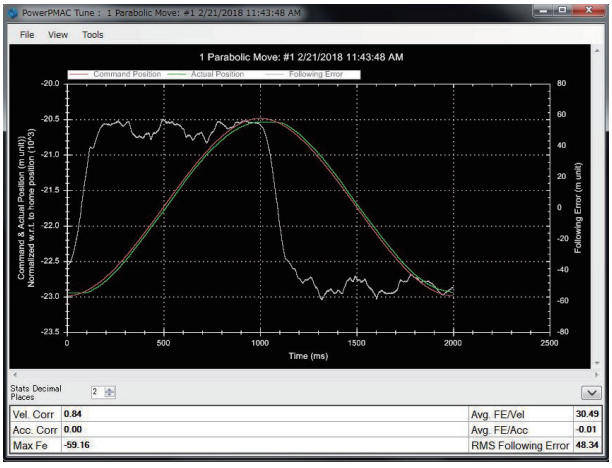
5 If the Following Error has a positive correlation to the speed, make Kvff larger. If it has a reverse correlation, make Kvff smaller.



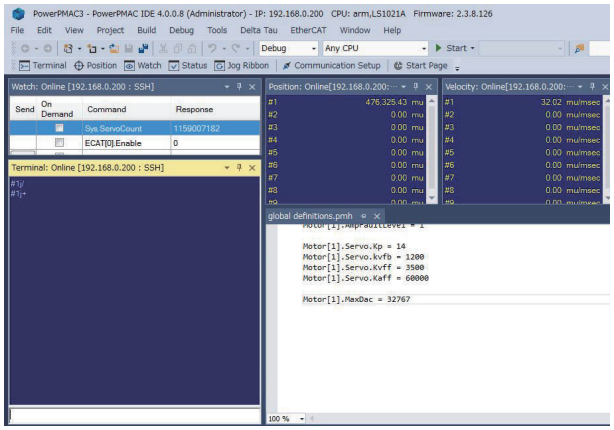
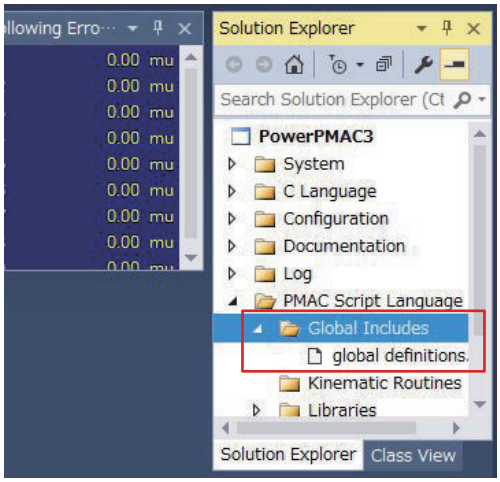
6 Click **Parabolic Velocity Move** again. Repeat this until the correlation of Following Error to the speed disappears.



7 In the same way, if Following Error has a correlation to acceleration, friction, etc., increase or decrease the Kaff and Kfff values. The figure on the right is an example of a correlation to friction.



3-7-5 Creation of Tuning Parameter Project

1	Type the **#1 j+** command from the Terminal.	
2	Make sure that the motor is rotating. In addition, confirm that the Watch window **Velocity** value is around +32.	
3	Open Global Definitions.pmh under **PMAC Script Language** — **Global Includes** in the Solution Explorer.	

4	Add the values obtained from tuning to the Global Definitions.pmh.	`Motor[1].Servo.Kaff = ***`
		`Motor[1].Servo.Kvff = ***`
		`Motor[1].Servo.Kp = ***`
		`Motor[1].Servo.Kvfb = ***`
		`Motor[1].MaxDac = 32767`

3-8 Absolute Encoder System Home Setting

This section describes only the homing for the absolute encoder system. For the incremental encoder and homing commands (home and homez commands), refer to the attached DT manual.

Perform the home setting following the procedure below.

3-8-1 Absolute Encoder Setup

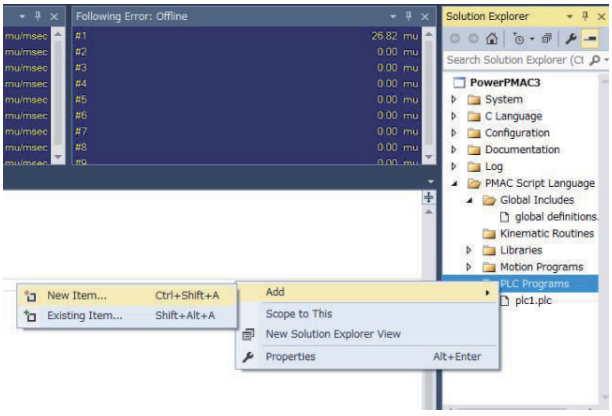
When using the absolute encoder for the first time, when wanting to initialize the rotation amount data to *0*, or when the absolute encoder has been left standing for a long period without connecting to a battery, etc., the absolute encoder setup is necessary. For details of the setup method, refer to the manual attached to the OMRON G5-series Servo Drive.

3-8-2 Read Absolute Encoder Position

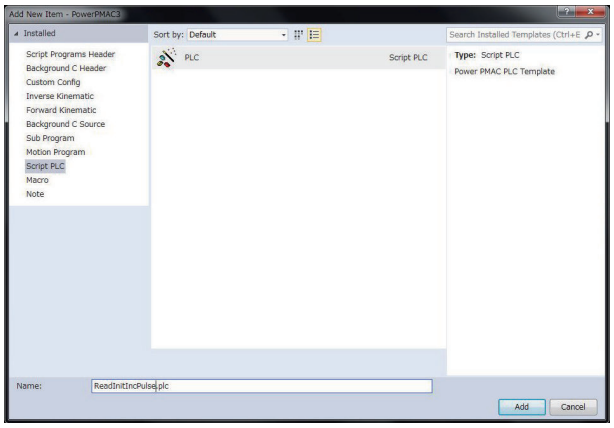
Read the absolute encoder position from the Servo Drive.
Carry out the absolute encoder wiring in *3-3-1 Axial Interface Unit and Driver Wiring* on page 3 - 7, then create a program to read multi-rotation data and initial incremental pulse in absolute encoder, and execute the program.
The procedure for creating a program is described below.

1	List the program on the right in the Solution Explorer in **PMAC Script Language** — **Libraries** subprog1.pmc.	`open subprog timer(delay_time)`
		`  local EndTime;`
		`  endtime = Sys.Time + delay_time;`
		`  while (endtime > Sys.Time){};`
		`close`

2	Add the code to the right to Global Definitions.pmh.	`Sys.Wpkey = $AAAAAAAA`
		`Gate3[0].SerialEncCtrl = $82230005`
		`Gate3[0].Chan[0].SerialEncEna = 1`

3	Right click on the Solution Explorer **PMAC Script Language** — **PLC Programs**, and select **Add** → **New Item…**.	

4 Select **Script PLC**, set Name to ReadInitIncPulse.plc, and then select **Add**.



5 List the program on the right in the Solution Explorer **PMAC Script Language — PLC Programs** ReadInitIncPulse.plc.

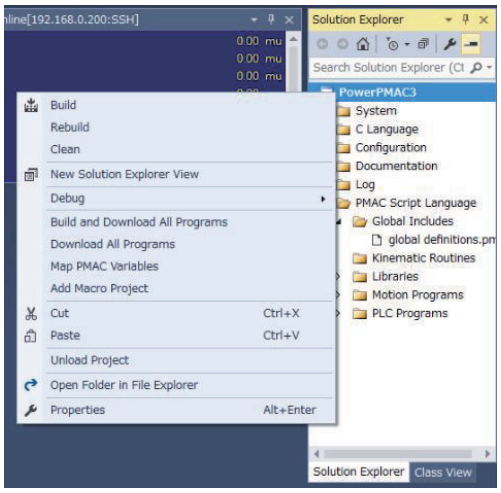
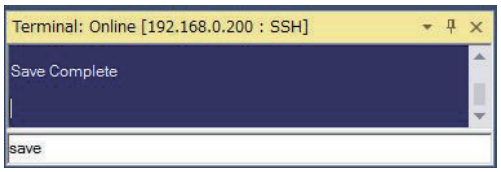
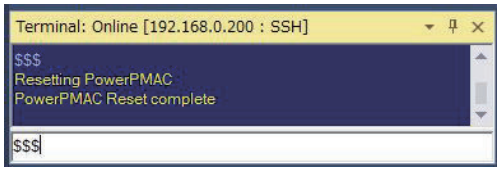
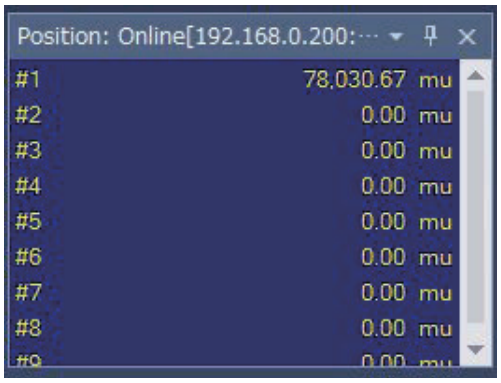
- The value shown below is determined by the resolution of the servomotor encoder being used, and by the electronic gear ratio of the Servo Drive. Set the value for the output pulse number per one motor rotation. **Global EncoderResolution = ****

```
Global MultiTurnCount, InitIncPulse;
Global EncoderResolution = 10000;

open plc 1

callsub sub.motorInitialize;
call timer(0.2);
callsub sub.requestMultiTurnData;
call timer(0.25);
callsub sub.readMultiTurnCount;
call timer(1.0);
callsub sub.readInitIncPulse;
Motor[1].HomePos = -Motor[1].HomeOffset

disable plc 1;
return;

////////////////////////////
sub: motorInitialize

 homez 1;
 kill 1;
 Gate3[0].Chan[0].OutFlagB = 0;
 Gate3[0].Chan[0].CountReset = 1;

return;
////////////////////////////
sub: requestMultiTurnData

 Gate3[0].Chan[0].SerialEncCmd=$13000
 Gate3[0].Chan[0].OutFlagB = 1

return;
```

```
//////////////////////////////
sub: readMultiTurnCount

 local tmpSerialEncDataA, tmpSerialEncDataB;
 local cAsciiOffset = 48;
 tmpSerialEncDataA = Gate3[0].Chan[0].SerialE
ncDataA;
 tmpSerialEncDataB = Gate3[0].Chan[0].SerialE
ncDataB;

 local calcMotorPos = 0;
 calcMotorPos =  (  tmpSerialEncDataA & $FF)-
cAsciiOffset
 calcMotorPos += (( tmpSerialEncDataA & $FF00
)>>8-cAsciiOffset)*10
 calcMotorPos += (( tmpSerialEncDataA & $FF00
00)>>16-cAsciiOffset)*100
 calcMotorPos += (( tmpSerialEncDataA & $FF00
0000)>>24-cAsciiOffset)*1000
 calcMotorPos += (( tmpSerialEncDataB & $FF)-
cAsciiOffset)*10000
 MultiTurnCount = calcMotorPos;
 if(((tmpSerialEncDataB & $FF00) >> 8) == 45)
 MultiTurnCount *= -1

return;
//////////////////////////////
sub: readInitIncPulse

 local tmpInitIncPulse;
 tmpInitIncPulse = (Gate3[0].Chan[0].ServoCap
t) >> 8;
 InitIncPulse = tmpInitIncPulse;
 Motor[1].Pos = MultiTurnCount * EncoderResol
ution + tmpInitIncPulse;

return;
//////////////////////////////

close;
```

6 List the program on the right in the Solution Explorer **Configuration** pp_startup.txt.

```
enable plc 1
```

7	Downloading the project Right click on the **Solution Explorer** project name at the upper right of the IDE screen, select **Build and Download All Programs**, and execute Build & Download.	
8	Type the **save** command in the Power PMAC IDE Terminal. When the save is completed, "Save Complete" is displayed in the Terminal.	
9	Type the **$$$** command in the Power PMAC IDE Terminal.	
10	Check that the current position is reflected in the Power PMAC IDE Watch window.	

3-8-3 Execute Homing

Execute homing. For the homing method, refer to the attached DT manual.

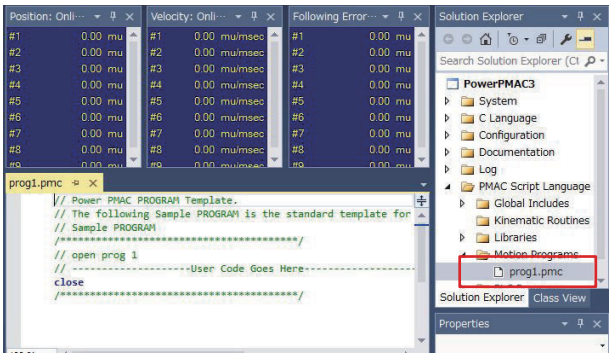
3-9 Operations Check Based on Motion Program

3-9-1 Creation of Operations Check Program

Create a program for the operations check.
The operations check program uses the specific language. For details, refer to *Power PMAC User's Manual (Cat. No. O014)* and *Power PMAC Software Reference Manual (Cat. No. O015)*.

1 Creating the Motion Program

In the Solution Explorer window, open **Project Name — PMAC Script Language — Motion Programs — prog1.pmc**.



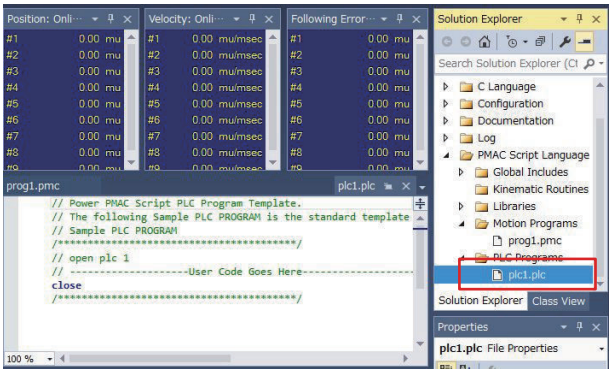
2 In the prog1.pmc tab programming area, write in the program listed on the right.

- This program example repeatedly rotates the motor in the clockwise direction and stops, and then rotates in the counterclockwise direction and stops.

```
&1;
#1->131072X;

OPEN PROG 1

   INC;
   TA800;
   TS300;
   LINEAR;
   While (1 < 2)
   {
      TA800;
      TS300;
      TM3000;
      X20;
      DWELL2000;
      X-20;
      DWELL2000;
   }

CLOSE
```

3 Creating the PLC Program

In the Solution Explorer window, open **Project Name — PMAC Script Language — PLC Programs — plc1.plc**.



4

In the **plc1.plc** tab programming area, write in the program listed on the right.

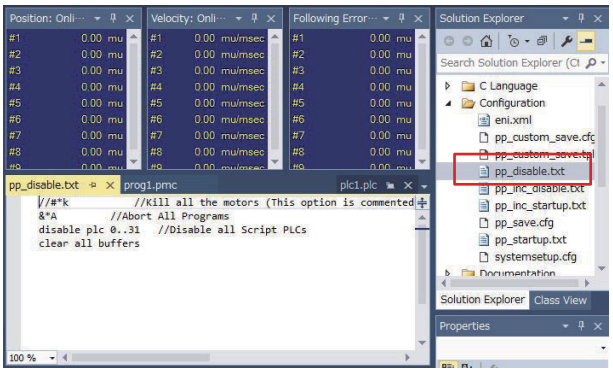
- This program example switches ON the servo, and starts up the motor user program 1, and then ends the execution of the PLC user program cycle.

```
open plc 1

    P1000=Sys.Time+1;
    while(P1000>Sys.Time){};

    cmd"&1enable";

    P1000=Sys.Time+5;
    while(P1000>Sys.Time){};

    cmd"&1b1r";

    disable plc 1;

close
```

5

User Program Startup Settings

In the Solution Explorer window, open **Project Name** — **Configuration** — **pp_disable.txt**.



6

In the **pp_disable.txt** tab programming area, add in the program listed on the right.

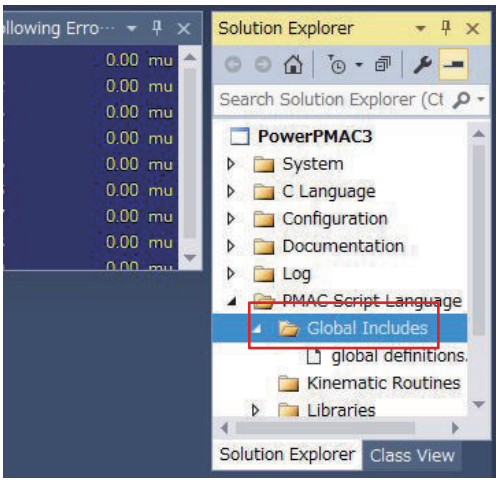
- The **pp_disable.txt** is automatically executed when the Controller starts up.
- In the listed example, execute the PLC1 script.

```
    enable plc 1;
```

7

Parameter Settings for Motor Control

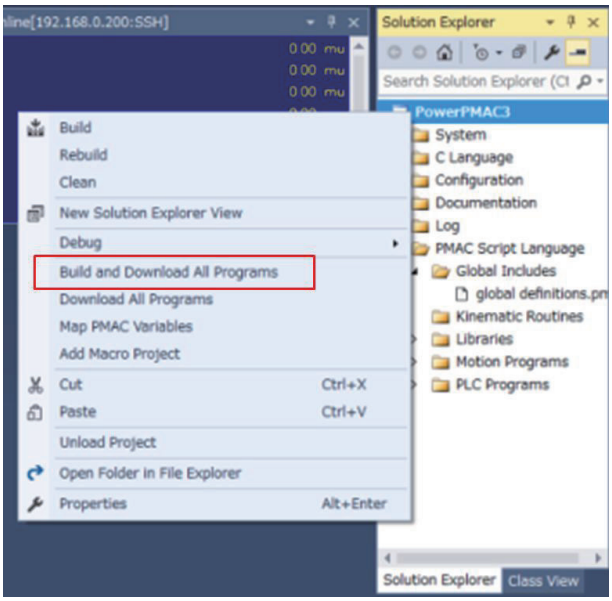
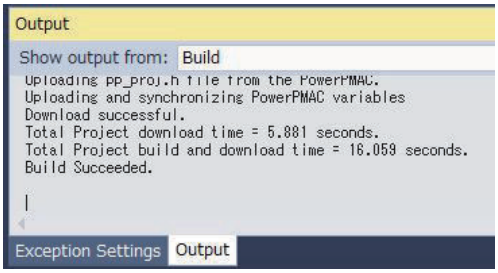
In the Solution Explorer window, open **Project Name** — **PMAC Script Language** — **Global Includes** — **global definitions.pmh**.

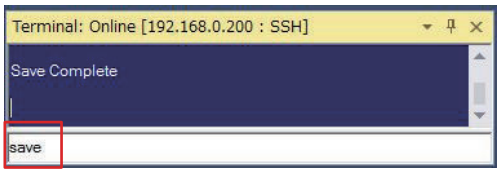


8	In the **global definitions.pmh** tab programming area, input the setting values to be set by automatic execution when the power is switched ON.
	• An example of the settings is shown on the right.
		`Motor[1].FatalFeLimit=0;`
		`Motor[1].AbortTa= -0.1;`
		`Motor[1].AbortTs= 0;`
		`Motor[1].MaxSpeed= 5000;`
		`Motor[1].JogTa= -0.1;`
		`Motor[1].JogTs= -1;`
		`Motor[1].JogSpeed= 1000;`
		`Motor[1].HomeVel= 1000;`
		
		`Coord[1].Tm=100;`
		`Coord[1].FeedTime=60000;`
		`Coord[1].MaxFeedRate=5000;`
		`Coord[1].Td=-0.1;`
		`Coord[1].Ta=-0.1;`
		`Coord[1].Ts=-1;`

3-9-2 Transferring Project Data and Checking the Operation

Transfer the created project data to the Controller.
When you transfer the project, the program automatically starts up, and the motor rotates.

1	Downloading the project Right click on the **Solution Explorer** project name at the upper right of the IDE screen, select **Build and Download All Programs**, and execute Build & Download.	
2	Make sure that there are no errors in the Output Window. • If the transfer failed, check the content of the error in the Output Window. If there is a program error, review the program.	
3	When download is successful, the program executes.	

4	Confirm that it is operating correctly, and then save the project to the Controller. • Execute the "save" command from the Terminal. • Transfer alone will not save the project to the Controller. If the power to the Controller is switched OFF without executing the "save" command, the transferred project is destroyed.	

OMRON Corporation **Industrial Automation Company**
Kyoto, JAPAN

Contact: www.ia.omron.com

Regional Headquarters
OMRON EUROPE B.V.
Wegalaan 67-69, 2132 JD Hoofddorp
The Netherlands
Tel: (31)2356-81-300/Fax: (31)2356-81-388

OMRON ELECTRONICS LLC
2895 Greenspoint Parkway, Suite 200
Hoffman Estates, IL 60169 U.S.A.
Tel: (1) 847-843-7900/Fax: (1) 847-843-7787

OMRON ASIA PACIFIC PTE. LTD.
No. 438A Alexandra Road # 05-05/08 (Lobby 2),
Alexandra Technopark,
Singapore 119967
Tel: (65) 6835-3011/Fax: (65) 6835-2711

OMRON (CHINA) CO., LTD.
Room 2211, Bank of China Tower,
200 Yin Cheng Zhong Road,
PuDong New Area, Shanghai, 200120, China
Tel: (86) 21-5037-2222/Fax: (86) 21-5037-2200

Authorized Distributor:

Cat. No. R193-E1-02 0718