

dedicated software for small CPU board “VSRC003”

RobovieMaker2 User Guide

ATR

Vstone Co., Ltd.

Contents

| | |
|---|-----------|
| 1.Welcome | 5 |
| 1-1.Overview | 5 |
| 1-2.Differences between this software and the “RobovieMaker for VS-RC003” | 6 |
| 1-3.Using data from “RobovieMaker for VS-RC003” | 6 |
| 1-4.CPU board layout | 7 |
| 1-5.Connect the CPU board and your PC | 9 |
| 2.Installing this software | 12 |
| 2-1.Installing this software to your PC | 13 |
| 2-2.Creating Robot Project | 14 |
| 2-2-1.Starting up the software. | 14 |
| 2-2-2.Creating Robot Project | 16 |
| 2-2-3. Loading a Robot Project | 18 |
| 3.Making robot motions | 19 |
| 3-1.Outline of the Pose Area | 19 |
| 3-1-1.Pose sliders | 20 |
| 3-2.Calibrating servomotor angles | 21 |
| 3-2-1.Communicating with the CPU board | 22 |
| 3-2-2.Adjusting robot to reference poses | 23 |
| 3-2-3.Writing reference pose information to the CPU board | 24 |
| 3-3.Creating robot poses | 27 |
| 3-3-1.Creating poses constituting motions | 28 |
| 3-3-2.Playing created motion | 30 |
| 3-3-3.Saving/Loading motions | 31 |
| 3-4.Assembling motions | 32 |
| 3-4-1.Pose block | 32 |
| 3-4-2.Changing motion speed and play time | 33 |
| 3-4-3.Setting pose order in pose | 34 |
| 3-4-4.Loop block | 35 |
| 3-4-4-1.Inserting and setting loop blocks | 36 |
| 3-4-4-2.Loop counter | 37 |
| 3-4-4-3.Example of use of loop structure | 37 |
| 3-4-4-4.Advantages of loop structure and cautions | 38 |
| 3-4-5.Operation block | 39 |
| 3-4-5-1.Inserting operation block | 39 |
| 3-4-5-2.Setting for operation block | 40 |
| 3-4-6.Conditional branch block | 41 |
| 3-4-6-1.Conditionl branch block settings | 42 |
| 3-4-7.Start block and end block | 43 |
| 3-5.Advanced useful functions to create motions | 46 |
| 3-5-1.Pose area advanced control | 45 |
| 3-5-1-1.Selecting pose slider | 45 |
| 3-5-1-2.Using mouse wheel | 45 |
| 3-5-1-3.Inversing and mirroring using right-left settings | 46 |
| 3-5-1-4.Copy and paste of pose sliders | 46 |
| 3-5-1-5.Changing positions of pose sliders | 47 |
| 3-5-1-6.Changing step resolution temporary | 47 |

| | |
|--|----|
| 3-5-1-7.Output audio setting | 47 |
| 3-5-2.Motion area manipulation | 48 |
| 3-5-2-1.Moving blocks | 48 |
| 3-5-2-2.Copy, paste, paste, and cut blocks | 48 |
| 3-5-2-3.Inserting a intermediate motion in a motion | 49 |
| 3-5-2-4.Playing motions frame by frame advance | 49 |
| 3-5-2-5.Changing size of pose area and motion area | 50 |
| 3-5-2-6.Creating variable computation and branch only motion | 50 |
| 3-5-2-7.Combining multiple motion files | 51 |
| 3-5-2-8.Mirror projection of whole motion at once | 51 |
| 3-5-2-9.Offsetting pose sliders of multiple pose blocks | 51 |
| 3-6.Pose slider preferences | 52 |
| 3-6-1.Text preference | 53 |
| 3-6-2.Numerical settings | 53 |
| 3-6-3.Right-left preference | 54 |
| 3-6-4.Slider position and flag settings | 55 |
| 3-7.Reference poses preferences | 56 |
| 3-8.Importing motion file of VS-H8PWM28V2 | 58 |
| 3-8-1.File Import | 58 |
| 3-8-2.Changing setting of motion file import | 59 |

4.Mode switch and audio settings-----60

| | |
|--|----|
| 4-1.Auto demonstration | 61 |
| 4-1-1.Branching motions in auto demonstration | 64 |
| 4-2.Preferences of control map V1 | 65 |
| 4-2-1.Outline of Control map V1 | 66 |
| 4-2-2.Assigning a motion file to the input | 67 |
| 4-2-3.Cancellation preferences | 68 |
| 4-2-4.Analog input | 68 |
| 4-2-5.Save/Load Control map V1 | 70 |
| 4-2-6.How to use controller in Control map V1 | 71 |
| 4-3.Preferences of Control map V2 | 73 |
| 4-3-1.Outline of Control map V2 | 73 |
| 4-3-2.Assigning motion files to Control map V2 | 74 |
| 4-3-2-1.Motion assignment to Game pat input | 75 |
| 4-3-2-2.Motion assignment to a variable | 75 |
| 4-3-2-3.Map transition motion | 76 |
| 4-3-3.Operation in dialog of Control map V2 | 77 |
| 4-3-3-1.Idling motion preference | 78 |
| 4-3-3-2.Cancellation | 78 |
| 4-3-4.Save/Load Control map V2 | 79 |
| 4-3-5.How to use controller in Control map V2 | 79 |
| 4-4.Audio file setting | 80 |
| 4-5. Setting the ROM region for audio and motion data | 82 |
| 4-6.Transfer settings of mode switch, control map, and audio files | 83 |

5.CPU board configuration-----85

| | |
|---|----|
| 5-1.Variables in CPU board | 85 |
| 5-2.Basic preference, interpolation, and controller setting | 86 |
| 5-3.Servomotor preferences | 88 |

| | |
|---|------------|
| 5-4.Expanded board preferences----- | 91 |
| 5-5.Updating firmware of CPU board----- | 92 |
| 6.Display preferences----- | 95 |
| 6-1.Selecting displayed toolbar buttons----- | 95 |
| 6-2.Skin preferences----- | 96 |
| 6-3.Other preferences----- | 97 |
| 6-3-1.Display preferences in motion area----- | 98 |
| 6-4.Status window----- | 99 |
| 7.Support information----- | 100 |
| Appendix----- | 104 |
| Glossary----- | 104 |
| Name of buttons in toolbar----- | 107 |

1. Welcome

1-1. Overview

Thank you for using a RobovieMaker2 that is the dedicated software to manage a small CPU board, “VS-RC003” for controlling robots. Various settings and preferences, such as using servomotors, controllers and expanded boards, can be set from the RobovieMaker2. This software also includes a Graphical User Interface (GUI) panels making motions for robots build in “VS-RC003”. Before you start using this software and the CPU board, read carefully all the RobovieMaker2 users guide to know how to use them.

To use RobovieMaker2, you will need:

- **Operating system: Microsoft Windows 2000/XP/Vista**
- **Computer and processor: Pentium-III (recommended more than 1 gigahertz [GHz]) or higher**
- **Memory: 128 megabyte (MB) RAM or higher**
- **Interface: USB port**
- **Display: XGA (1024x768) or higher**

In the later part of this documentation, “RobovieMaker2” expresses as “this software”, and “**RobovieMaker2 user guide**” expresses as “this user guide”. Since the appendix explains about some other words and phrases, see them when you see unknown expressions in this user guide.

Note: Robovie and RobiveMaker are registered trademarks owned by ATR.

Note: Products and corporate name appearing in this manual my or my not be registered trademarks or copyright of this respective companies, and are used only for identification or explanation and to the owners’ benefit, without intent to infringe.

1-2.Differences between this software and the “RobovieMaker for VS-RC003”

This software is upward compatible of the “RobovieMaker for VS-RC003”. Since both versions share basic structures, all data from “RobovieMaker for VS-RC003” can be also used in this software. This software has many advanced specifications from the previous version, such as new control functions in the motion area, programmable branches using conditions, joint free functions for servomotors at each poses.

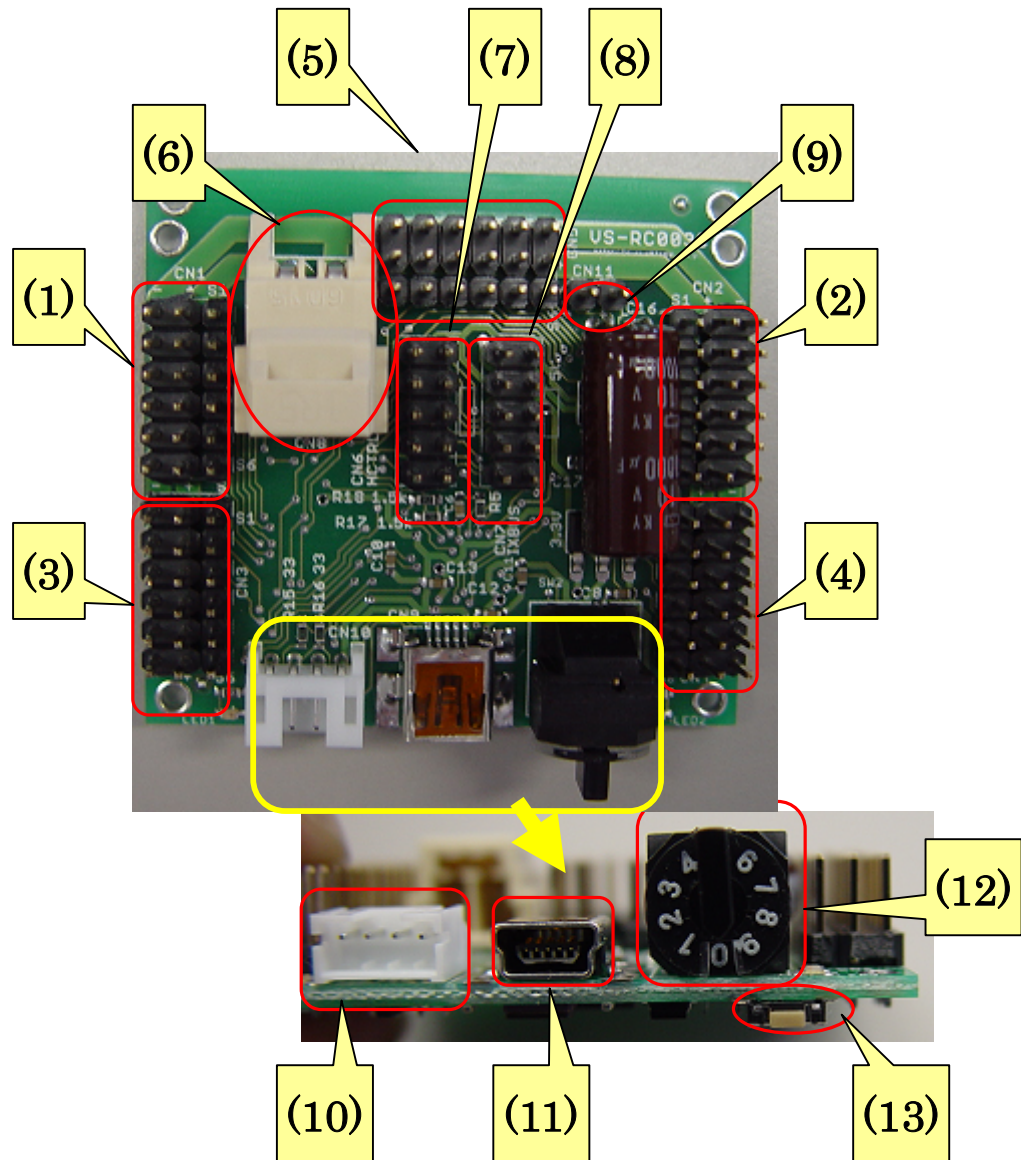
Caution: To use this software, the firmware of your CPU board needs to be the version 1.4 or later. You cannot use earlier versions of firmware with this software. In case using earlier versions, update the firmware of the CPU board using a firmware update function of this software before getting start.

1-3.Using data from “RobovieMaker for VS-RC003”

Robot-projects provided from “RobovieMaker for VS-RC003” are available in this software without any changes. This software is backward compatible with “RobovieMaker for VS-RC003”. However, robot-projects provided from “RobovieMaker2” are not available with “RobovieMaker for VS-RC003”. As a precaution, the user is recommended to back up own project when inheriting the project.

1-4. CPU board layout

This chapter contains outlines of how to handle the CPU board “VS-RC003”. The user needs to follow the handling guidelines.



Each part descriptions of the CPU board are as follows.

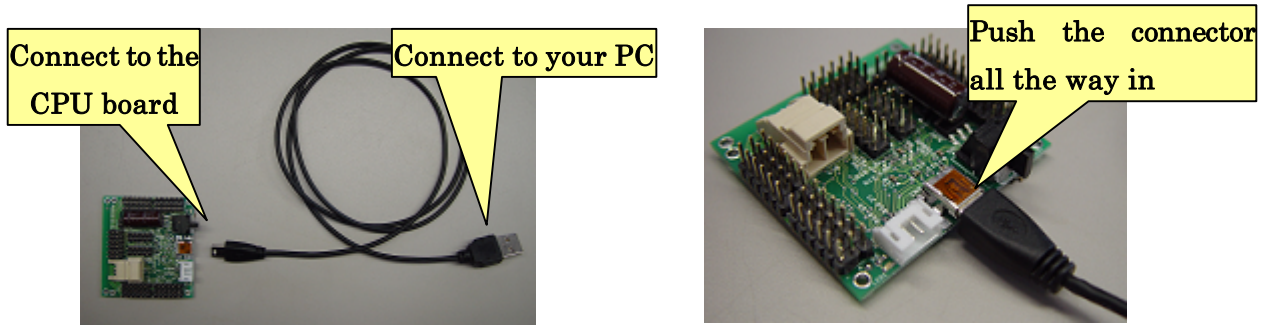
- (1) **Servomotor connector (CN1):** This connector is for connecting servomotors. Each connector named as CN1—5 respectively contains 6 channels outputs for servomotor, and is located separately on the CPU board.
- (2) **Servomotor connector (CN2):** This connector is for connecting servomotors.
- (3) **Servomotor connector (CN3):** This connector is for connecting servomotors.
- (4) **Servomotor connector (CN4):** This connector is for connecting servomotors.
- (5) **Servomotor connector (CN5):** This connector is for connecting servomotors.

- (6) **Power supply connector:** This connector is for supplying power. The CPU board works using the power supplied from USB connector. However, the other expanded boards, such as servomotors and speakers, require an additional power supply from this connector.
- (7) **Controller connector:** This connector is for the optional controller.
- (8) **Expanded board connectors:** This connector is for optional expanded boards. This connector also includes a port directory connected to the audio output without intermediate onboard amplifier.
- (9) **Audio output connector:** This connector is for audio output signal. To use onboard amplifier, connect a speaker directory to this connector. Do not connect any other parts but connect from a speaker. Also, note that you need to prepare the speaker.
- (10) **Communication board connector:** This connector allows you to use an optional intermediate communication board making a connection between your PC and the CPU board. This connector is unnecessary when the communication cable connector (11) accomplishes the connection.
- (11) **Communication cable connector:** This connector is for a communication between your PC and the CPU board through a communication cable. Since the connector is standard USB mini-B, the user can use commercially available standard USB cables.
- (12) **Mode switch:** This switch is for selecting auto-demo and control maps after transferring a user programs, in case the user want to boot the CPU board itself without this software. See “4. Mode switch and audio preference”) for more details about the preferences assigning the auto-demo and control map to number of the switch.
- (13) **Reset switch:** This switch is for rebooting the CPU board to reset the program. When the user initializes the CPU board, when your PC cannot recognize the CPU board, or when other problems occur, reboot the CPU board using this reset switch.

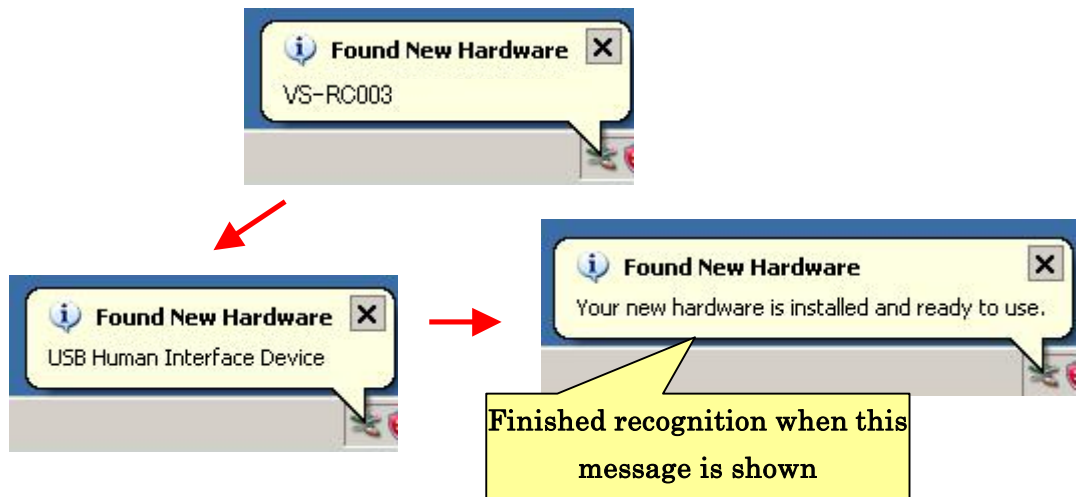
1-5. Connect the CPU board and your PC

Use USB port to connect your PC and the CPU board. Your PC will immediately recognize the CPU board using the USB hotplug support. You do not need to install device driver. If you use the CPU board at the first time, make your PC recognize the CPU board as directed below.

Connect your PC and the CPU board as you see in the following picture.

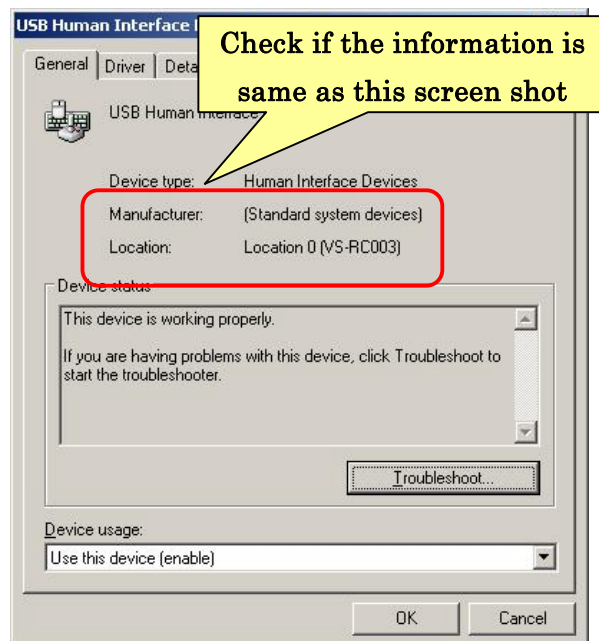
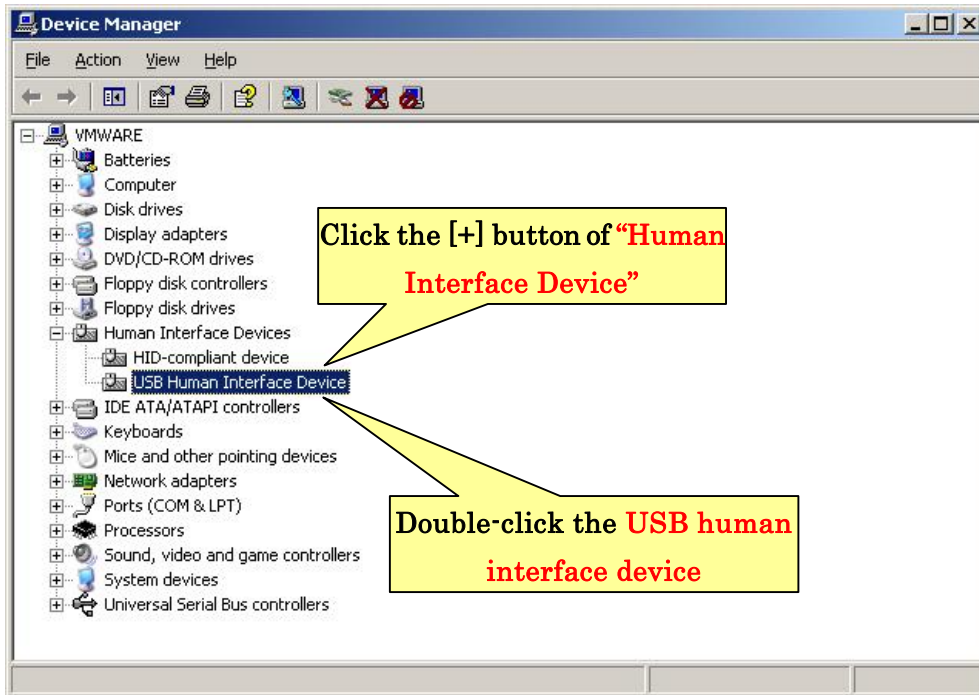


At the first connecting, your PC will take few seconds to recognize the CPU board. In your display, you will see the recognition process as following pictures. The final message will be “**Your new hardware is installed and ready to use**” meaning your PC finishes the processes for recognizing the CPU board. Then, install this software.

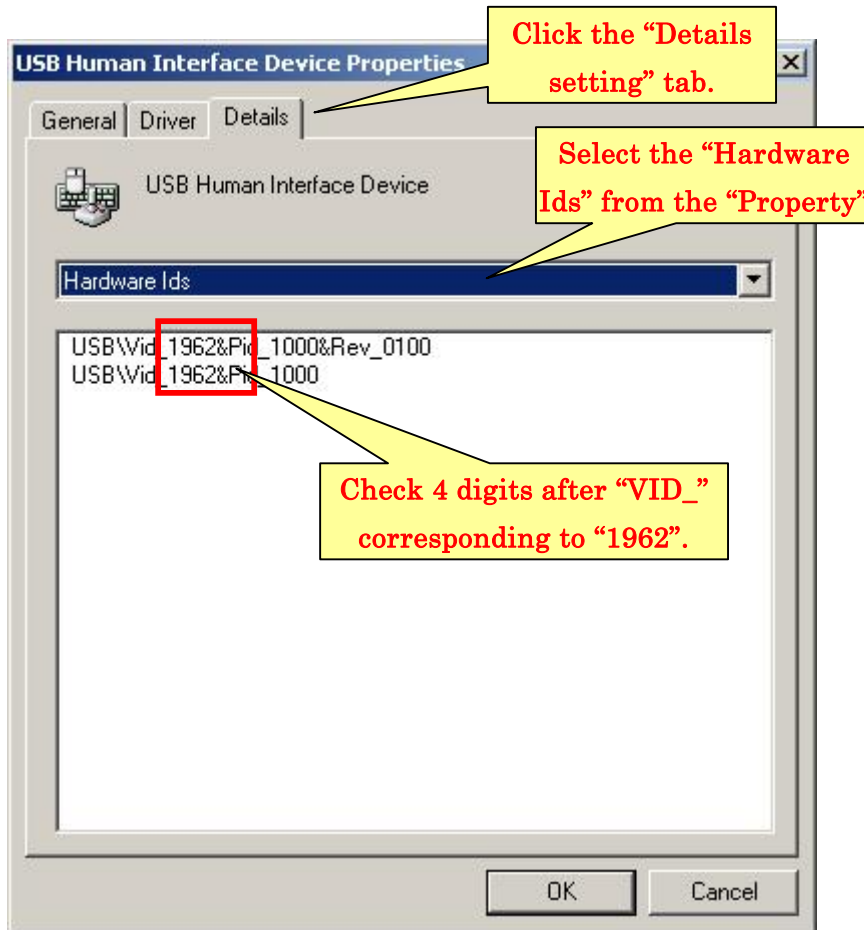


Once your PC recognizes the CPU board, your PC can immediately finish the recognition in next connection. (You will hear a Windows sound effect named “Windows XP Hardware Insert.wav” as the default.)

When you want to check if your PC recognizes the CPU board, confirm whether if "USB Human Interface Devices" exists in the "Device manager"—"Human Interface Devices". Then, double click it. In the dialog of the "USB Human Interface Devices", check if the device type, the product ID, and the place are corresponding to the following picture.



In case your OS is Windows Vista, since the displayed messages are a little bit different, confirm the connection of the CPU board as follows.



As you see from the above picture, you can confirm the connection whether if the "Human Interface Devices" property shows values containing the string "VID_1962".

2. Installing this software

This section explains how to install this software, and how to create “Robot-project” that is the first step to use this software.

2-1. Installing this software to your PC

Use the “**RobovieMaker2_inst_???.exe**” contained in an attached CD-ROM, where “???” are 3 digits, when you install this software. After open it, follow the instruction of setup wizard as following figures.

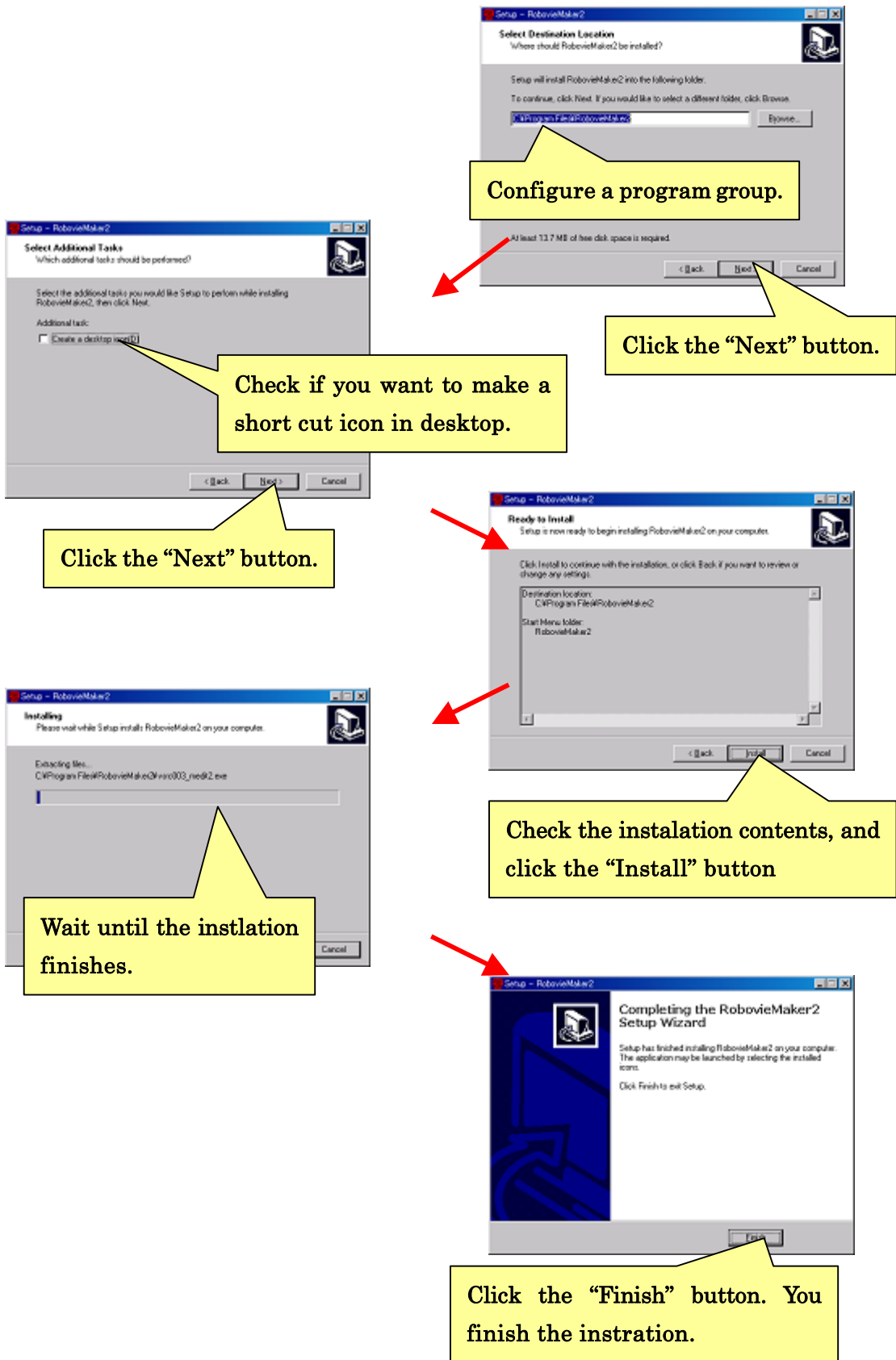
Execute the “RobovieMaker2_Inst_(version).exe”.

Click the “Next” button.

Click “I accept the agreement” if you accept the license agreement.

Select a folder you want to install in.

Click the “Next” button.



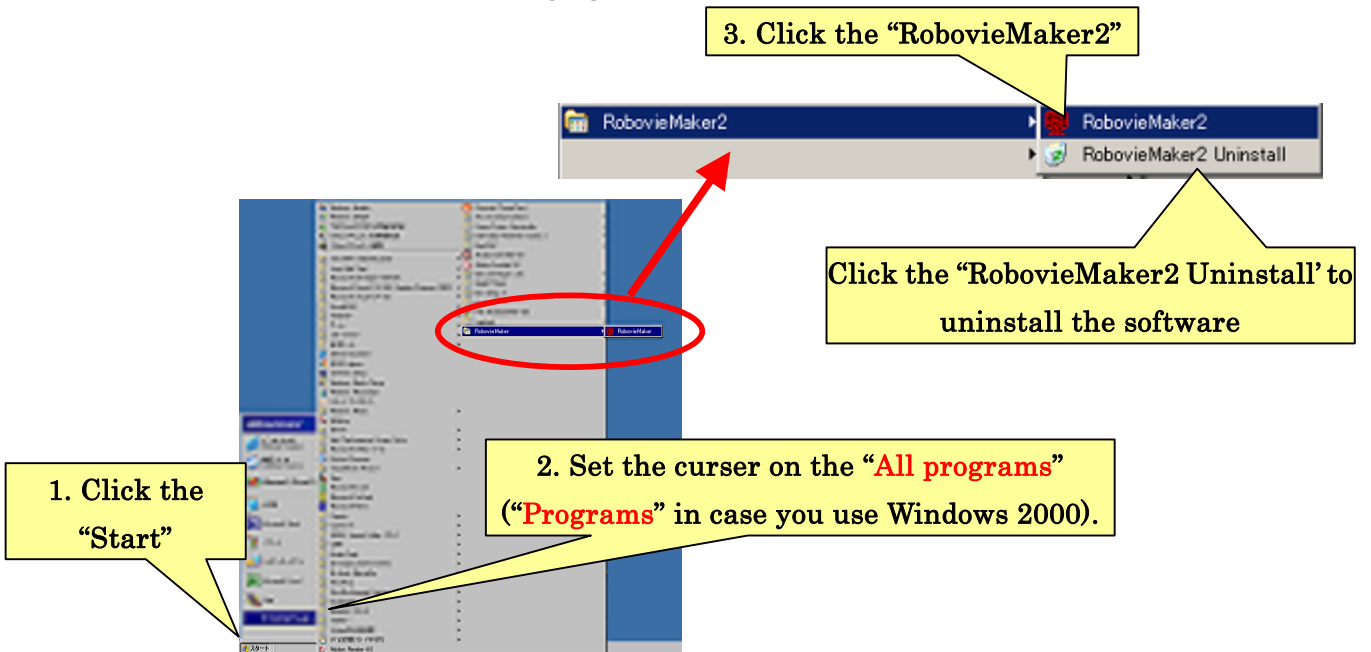
2-2. Creating Robot Project

After you install this software to your PC, you need to setup preferences of this software to use it. In this software, a set of preferences called “Robot Project” is a workspace for each robot. Create your own Robot Project for your robot as detailed below.

Note that, once you create the project, you do not need to create the project again. This software uses same project you previously used in the next time.

2-2-1. Starting up the software.

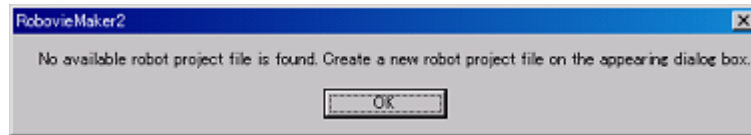
This software can start from the OS start menu to click “All programs” – “RobovieMaker2”. (See the following figure.)



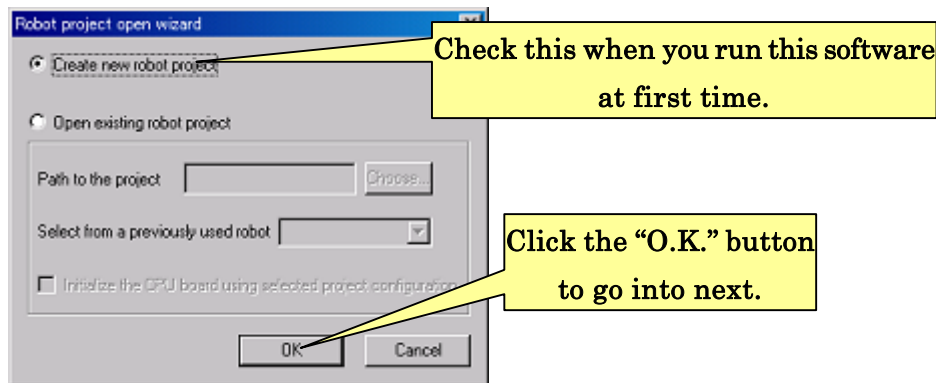
This software can also start from a desktop short cut created in case you choose “**Make a shortcut at Desktop <check>**” when you install the software.

If you want to uninstall this software, use the short cut to “RobovieMaker2 Uninstall” in Start menu.

Note that, in the first start, this software shows the dialog window with a message to show that you do not have any Robot Projects in your PC. Click “OK” to create your first Robot Project.



After clicking “OK”, you will see the following dialog. To create a new Robot Project, in this first case, check the radio button of “Create new robot project” and click “OK”.

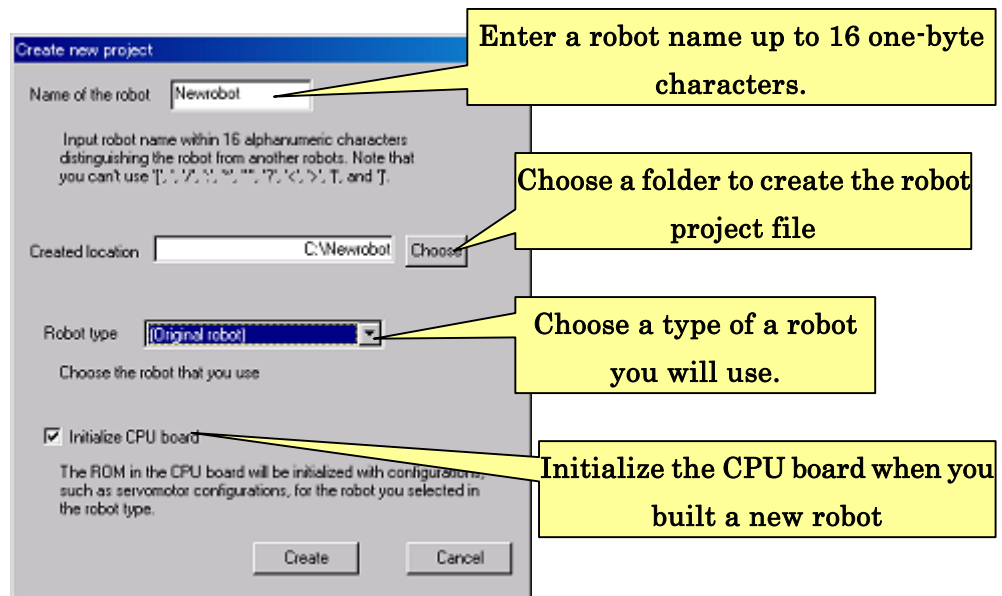


Note that; in case using a previously created Robot Project, you can load the Robot Project and start this software by checking the radio button of “Open existing robot project”. Choose the absolute location of the Robot Project file, (, such as *.rpj,) to open it. This will be happen when you reinstall this software or you copy the Robot Project file from another PC.

2-2-2. Creating Robot Project

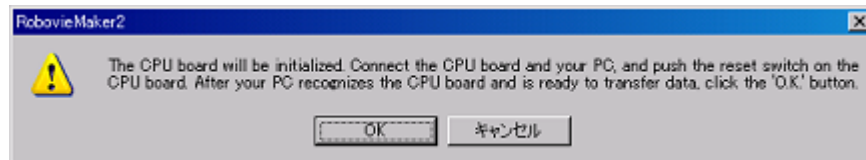
After selecting “Create new Robot Project” as the instruction in section 2-2-1, you will see the following dialog to create and configure a new Robot Project.

In this stage, you need to set a name of your robot and a location to create the project. Enter a robot name up to 16 one-byte characters and choose the location to create the project. Initialize the CPU board when you use it at the first time on your robot, in such case after you build a new robot. Even initialized; you need to initialize it again in case you use it in the different robot. Check “Initialize CPU board” to initialize your CPU board. After set all the configuration click “Create” to create the project.

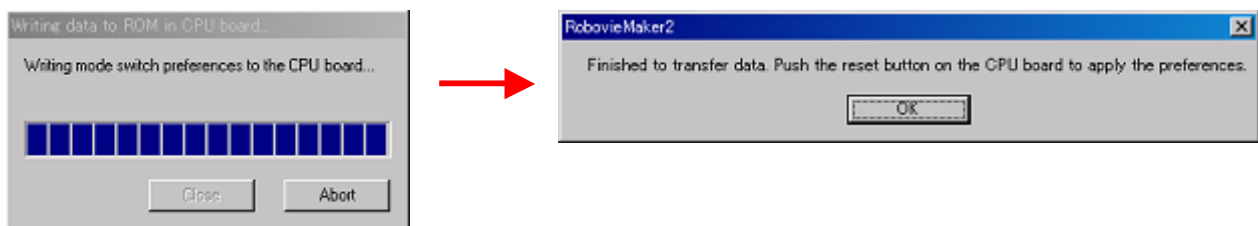


Create different projects for different robots respectively, in case that “you bought another robots” or “you build the CPU board in the different Robot”. To create another new robot project, click “File”—“Create robot project” from the software menu. Then follow the same instruction described above.

During creating a Robot Project, you will see the following dialog when you check “Initialize CPU board”. Connect the CPU board to your PC, then click “OK” after confirming that your PC recognizes the CPU board.

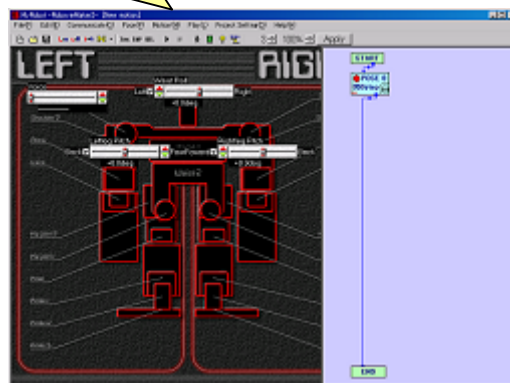


During the initialization, the dialog box shows you progresses as following picture. Since another dialog box will show you when the initialization finishes, click “OK” to close the dialog and push the reset switch on your CPU board.



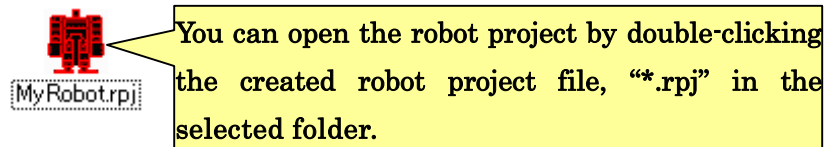
Now, you have created a new Robot Project at this stage. Then this software starts a project window (see below picture) to manage Robot Project. In second or later time, this software will start its program with this window using the project you created. Go to the section 3. “Creating motions of the robot”.

**A screen shot when the software starts with a robot project loaded.
*This screen shot can be different depending on the configuration you set in creating robot project.**

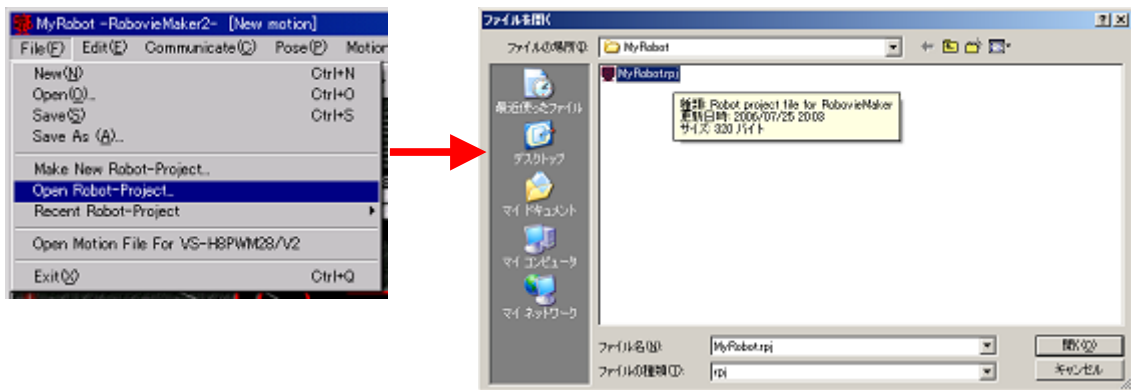


2-2-3. Loading a Robot Project

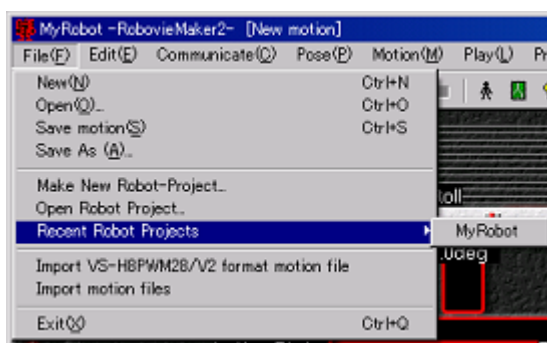
You can start this software from the project file (“Robot Name’.rj”), by double-clicking the file created in the folder you selected as “Location to create a project” when you create the new project.



To load the different project after starting this software, click “File”—“Load robot project...” from the top menu bar. Then you see the dialog to select Robot Project file (*.rpj). Choose the file and click “Open”.



To load a Robot Project you recently used in this software, click the Robot Project from logs shown by setting cursor in “File”--“Recent Robot Projects”. (See below picture.)



3. Making robot motions

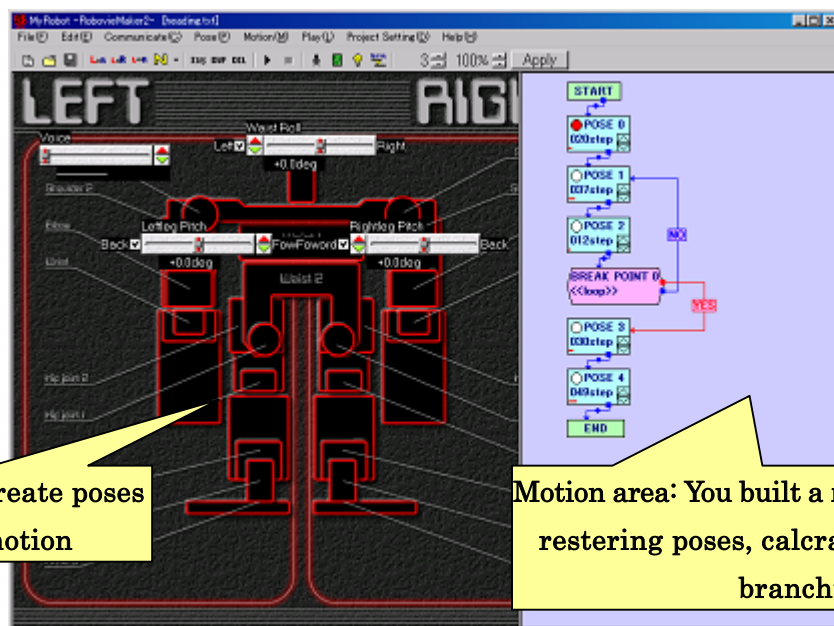
Making robot movements, such as a “bipedal walking” and a “get up motion”, is the main function provided in this software. This software can create various types of movement called “motion”, and allows you to use them for controller assigned motions or automated demonstrations.

Earlier part of this section details about basic knowledge for creating motions, before the section 3-4 “Assembling motions”. Read the part of this section at first, and try making your robot motions. The rest part of the section, form 3-4 “Assembling motions”, details mainly about convenient optional functions to provide motions.

3-1. Outline of the Pose Area

You can create motions in the “Pose area” and the “Motion area” that are the important areas to use this software. (See below picture). Imagine about animations to understand structures of the motion, since the motions consisting of poses arranged in time sequences. A robot performs poses at fixed periods.

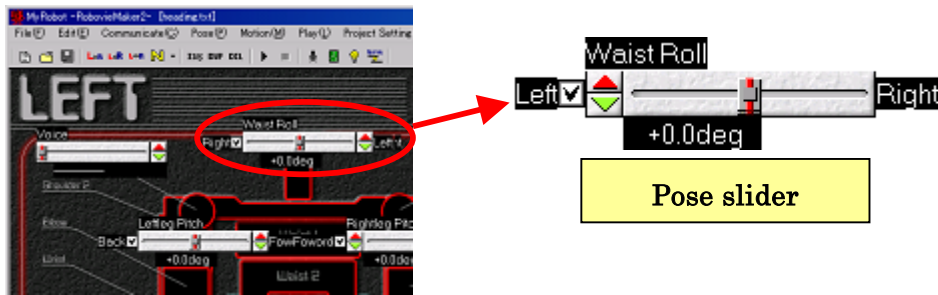
Create one pose in Pose Area, and register it in motion area. Repeat this to make motions using this software. Motion area has functions to change order of registered motions and to set transition time (, that decides motion speed). Motion area also has another function to register branch on condition and calculation of variables.



3-1-1.Pose sliders

The Pose Area show several “pose sliders (see below picture)” for controlling extended devices connected to the CPU board, such as servomotors, LED, and sensor gains. Using this, for example, you can make robot poses by setting servomotors angles.

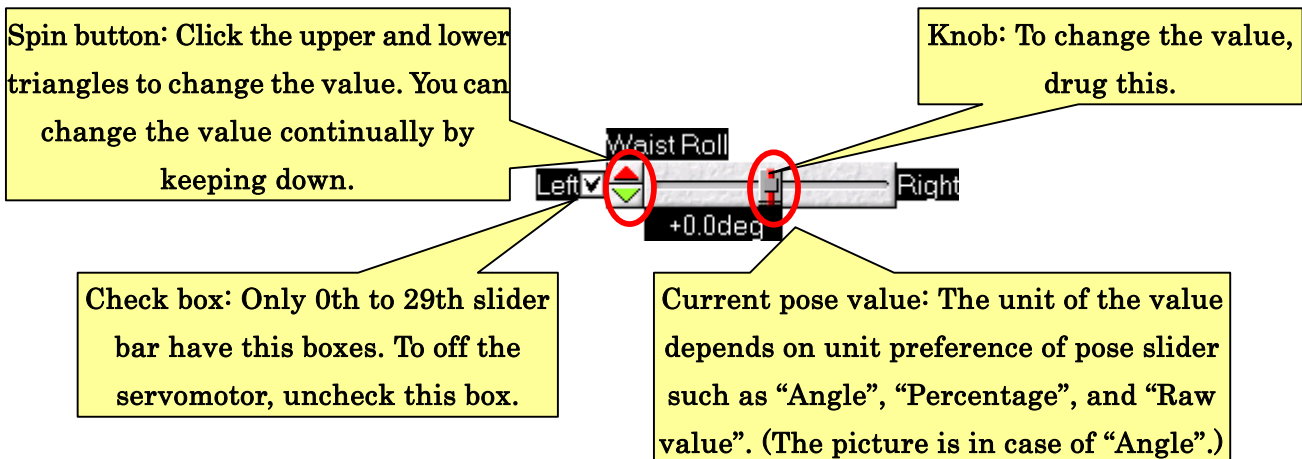
Pose area automatically configures pose sliders position and types as to selected robot. The pose area can show up to 62 sliders named from 0th to 61st, having the roles of controlling assigned devices servomotors respectively. For example, 0th to 29th pose sliders are for controlling angles of servomotors, the 30th pose slider is for selecting audio numbers to play, and 31st or later pose sliders are for expanded board settings, such as gain of gyro sensors.



This software uses the 62nd and the 63rd variables, which used to be used as pose sliders in "RobovieMaker for VS-RC003", as joint free function. Thus, you cannot use them for pose sliders in this software.

You can change values of pose sliders by controlling built-in “spin-button” and “knob” using mouse controls to set poses of robot. Built-in check boxes of 0-th to 29-th slider bars for controlling servomotors are on/off switches of servomotors.

Practice at least once to control pose sliders before starting control of robot. Move slider bar by your mouse control that are clicking spin button to change the value slightly and dragging knobs to change the value dynamically.




3-2. Calibrating servomotor angles

From this section, you will start controlling the robot.

Before creating poses and motions of the robot, you need to calibrate servomotors, meaning the adjustment of its reference angle. You first need to finish this adjustment at first because servomotors have differences in angle when same signals inputs to them. This angle difference occurs even within same types of servomotors.

You also need to adjust reference angle of servomotor when you replace servomotors, in such case of malfunctions.

When you start to calibrate servomotors, click  from toolbar or the top item from “Pose”—“Select reference pose” of menus. This operation controls all pose slider to initial angles.



Do not make servomotors “motor lock” detailed below since servomotor calibration requires you to supply power for robot indeed. “Motor lock” causes not only robot malfunctions but also injuries and some other accidents.

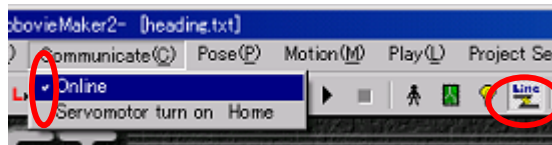
Warning: Do not make your servomotors “motor lock”!

“Motor lock” is a state of servomotor locked in a different angle from controlled angles, and suffered from tremendous loads. When “motor lock” happens, you will see several phenomena such as “large vibration of servomotors”, “large sound noises from servomotors”, or “heating up servomotors”. If you leave this state for long time, servomotors will burn out, resulting in malfunction of servomotors. In worse cases, the broken servomotors generate smokes or fires. Turn off the servomotor power switch when the “motor lock” happens. Do not turn on the power switch immediately until the servomotors become enough cool.

3-2-1. Communicating with the CPU board

Your PC can start communicating with the CPU board as follows.

Connect CPU board and PC with furnished accessory cable. Push the reset switch of the CPU board. Click “Communication”—“Online” from menu bar or  from tool bar, and your PC will start communicating with the CPU board. During the CPU board is online, “Communication”—“Online” is displayed with a check mark, and  is kept as pushed state. Click them again if you want to make it off line.





This is in case of keeping communication with the CPU board.

If your PC shows warning dialogs and fails to start communicating to the CPU board, see “Section 7.Support information” to check the cable connecting your PC and the CPU board or initialization of the CPU board.

Next step is to turn on the servomotors. Read carefully next warnings.

Warning: Be careful when you turn on the servomotors.

Your robot usually moves widely immediately after turning on, meaning starting controlling servomotors. This movement occurs when controlled poses of robot are different from the poses before turning on the servomotor, and can cause such accidents as stacking your fingers, falling down, and crushing. Lift and hold up a safe place with your one hand not to stack your finger.

In order to turn on the servomotor, click “Communication”—“Turn on servomotor” from menu bar or  from toolbar. During the servomotor is turned on, “Communication”—“Turn on servomotor” is displayed with a check mark, and  is kept as pushed state. Click them again if you want to turn off the servomotor. **Keep the servomotor turning off except as you are using it.**

3-2-2. Adjusting robot to reference poses

A reference pose, referred when creating robot poses, is a posture consisting of a set of servomotor angles. You will calibrate servomotors using this posture by observing differences of servomotor angles from supposed to be. In addition, you will start creating poses from this reference posture.

If you followed instruction of the previous section, that is clicking “N” from toolbar or the top item from “Pose”—“Select reference pose” of menus, your robot should be adjusted in a default reference pose. Since this default pose is close from the reference posture that you will set for your own robot. Adjust the actual pose of your robot to the reference posture.


Note that reference postures are different due to types of robot, but an appropriate default pose is automatically set when you select your robot type. If you use a commercially produced robot kit, check the reference posture usually shown in its attached manual as some pictures (that is erect posture in many cases). Prepare the documentation when you set a reference pose.

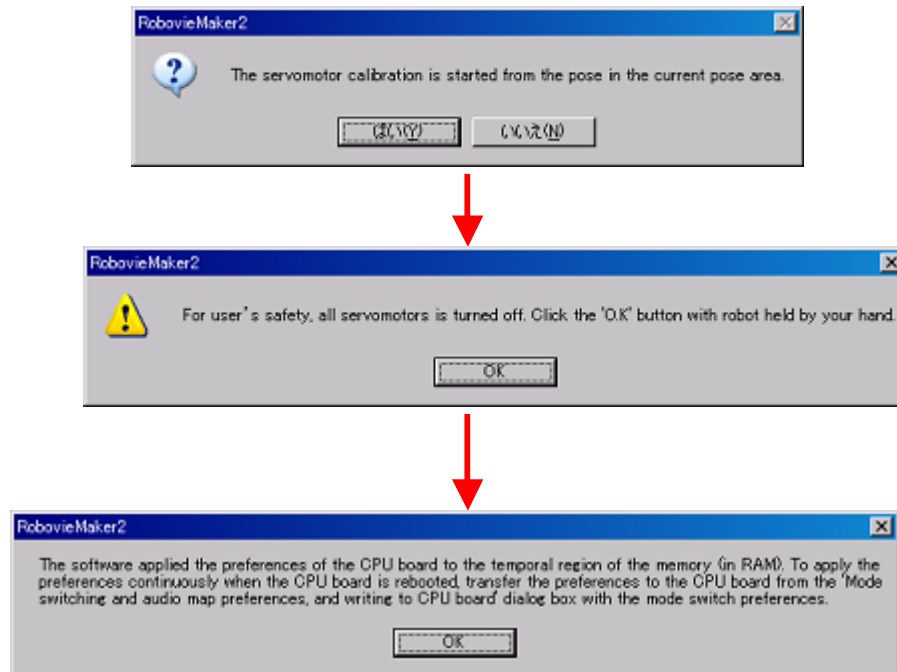
In case you create your own original robot (and want to use this software to control it), set all servomotor to the center of zero as a temporal default pose. See “3-7 Reference pose settings” to decide your original pose.

Use pose slider bar to adjust the robot pose to the reference posture. Since a servomotor along with your manipulation during your PC communicating with the CPU board, you see a current actual pose with your eyes.



- ※ **Note: During control, lift and grab your robot by your one hand. Click spin button to move servomotors slightly.**
- ※ **Note: In case you see a pose differ considerably from the reference posture, your robot was possibly assembled incorrect. Check if you really assembled your robot correctly, for example, that are how to attach servo hones or where to connect servomotor cables.**


After adjusting servomotors as the reference pose, click “” button from toolbar or “Project preferences”—“Calibrate servomotors” from menus. Since you will see the following dialog, click “O.K.” on each dialog **with lifting your robot with your one hand**.

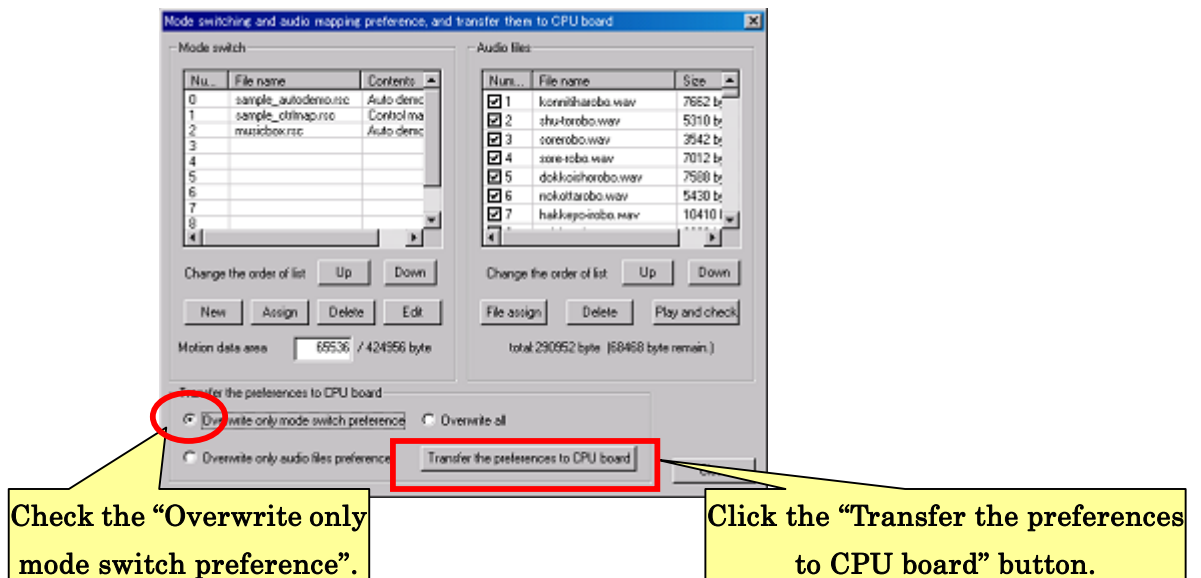


3-2-3. Writing reference pose information to the CPU board

The reference pose information is already set and translated in the RAM region of the CPU board. You need to write this reference information to the ROM region for keeping permanently, because the RAM region is a temporally memory and cleared when you reset or reboot the CPU board.

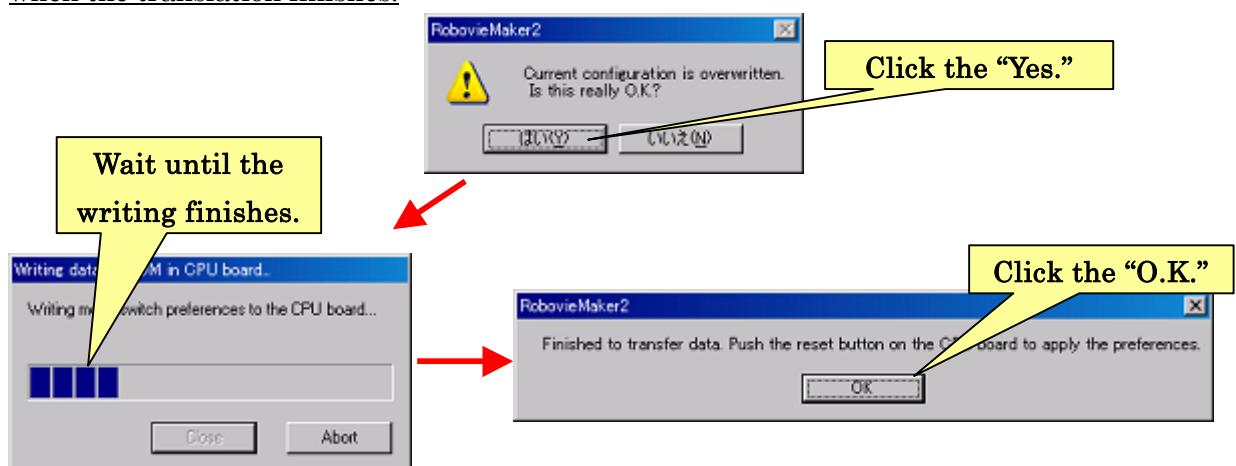
Use the dialog of “Mode switch/Audio map preference and translate it to CPU board” to write reference pose information to ROM region. This dialog is used for writing not only audio or controller preferences but also various configuration of the CPU board. The reference pose information is wrote using this function.

Click “” button from toolbar or “Project preferences”—“Mode switch/audio preferences and writing” from menu. You will see the dialog as a following figure.



Check “Overwrite only mode switch preference” from “Transfer the preferences to CPU board” on the “Mode switching and audio map preference and writing to CPU board”. Then click “Transfer the preferences to CPU board” button.

Follow the instruction shown in the following figures that are dialogs shown after clicking “Transfer the preferences to CPU board”. **Do not disconnect communication cable or push reset switch during writing data** to the CPU board. Push reset switch when the translation finishes.




Then you have finished servomotor calibration.

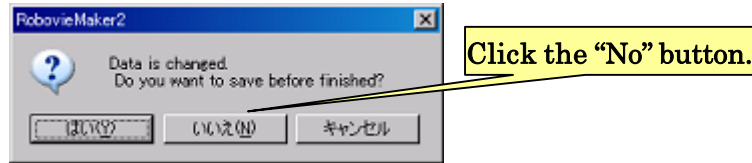
Note: You need to caliber servomotors regularly.

The errors of servomotor angles from the reference posture are increasing a little by little even you calibrate the servomotors once. Calibrate servomotors again if the errors become rather large. Although required calibration cycles are different between types of servomotors, in general, more intense moves and loads of servomotors require much frequent calibrations. In such case that “the robot fall down when playing the waking motion that should be success”, calibrate servomotors again.

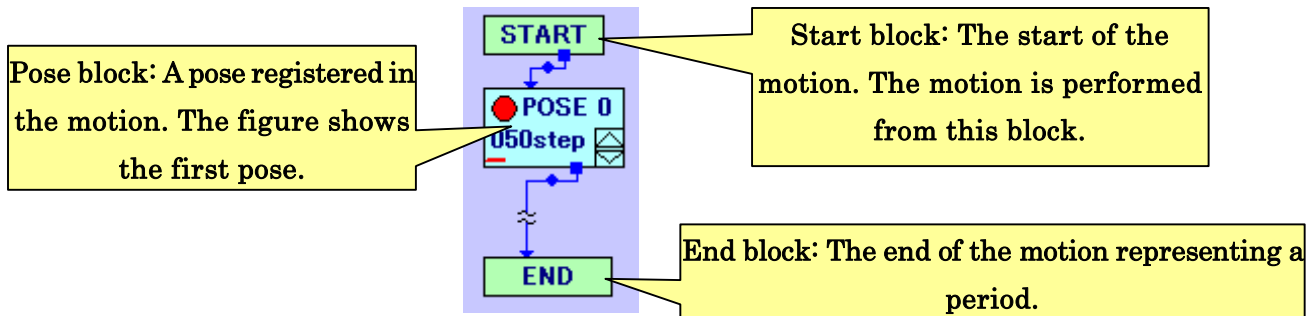
From next section, you make robot motions.

3-3. Creating robot poses


Click “File”—“New” from menus or “” from toolbar to create a new motion. If you have change slider bar positions, the software show you a dialog weather if you want to save them. Click “No”, since you don’t need to save them at this moment.



When you enter in creating motions, motion area registers a pose by default that you can edit in Pose Area. Creating motions starts from the reference pose as you see all pose slider is set to reference pose position.



Advanced information about blocks of Motion Area such as Pose Block, see “3-4. Assembling motions”.

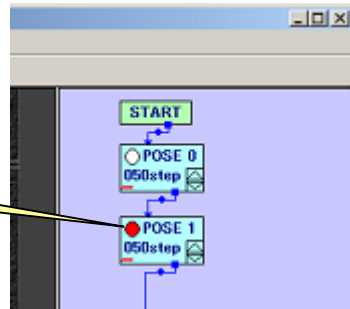
Click “Motion”—“Duplicate pose in editing” from menus or “” before editing first pose in Pose Area. This operation creates a new pose in Motion Area and is a function to insert a new pose by copying the current edited pose (called “Pose duplication”).

The reason why you duplicate reference pose is to start the created motion from reference pose regardless of previous poses. Using this strategy, you can play the motion robustly. Because of this reason, insert reference pose at the end of motion as much as possible.

3-3-1. Creating poses constituting motions

In this section, you create a pose next to the first pose that is the reference pose. The pose duplication makes an editable pose. The signals displayed on pose blocks indicate which one is in process of edit mode. A unique color of a pose block in Pose Area is a pose you are editing now. Click the other pose block to switch edited pose.

Signal: The pose in editing shows by an unique color from the others. To edit a pose, click the signal of it.



Keep your PC communicating with the CPU board, and turn on servomotors. It is convenient for users to move robot while creating poses. (Section “3-2-1.Communicating with the CPU board” details about how to do this.)

Now, make your original poses by controlling pose sliders. For the first step, it is better to try making simple motions moving the head or the arms so that the robot keeps its balance, such as “shaking head” or “put arms up”.

Make poses as well as when you have controlled pose sliders to adjust servomotors angle to a reference pose. However, in order to make outline of the pose, using “knob” is suitable to change servomotor angles dynamically rather than using spin buttons. (**Be careful in motor lock since the servomotor move widely!**)

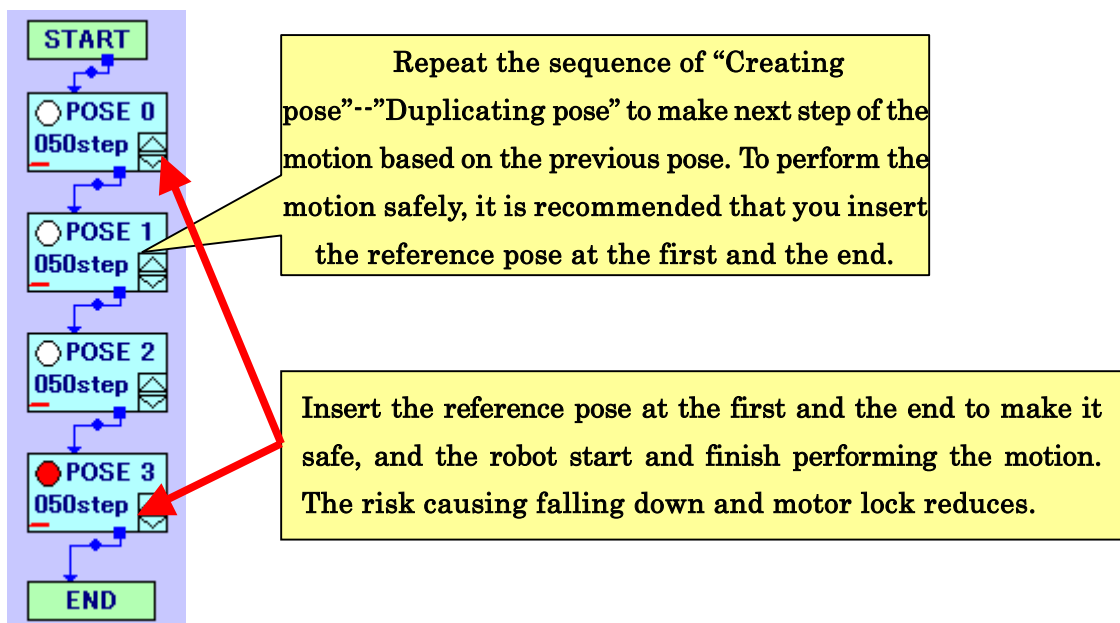
Checkboxes can switch servomotors on/off state during the pose. Uncheck these boxes to use joint free function during the motion.



After creating your original pose, duplicate again (by clicking “Motion”—“Duplicate pose in editing” from menu or “DUP”). This operation duplicates current pose and registers the copy in Motion Area. Edit this copy to make the next pose.

As you have seen in this procedure, motions are made by following steps that are repeat of “Duplicating current pose”—“Creating next pose based on the copy”—“Duplicating the pose”—“Creating next pose based on the copy”.

Therefore, you can work for creating motion along with repeat of “duplicating current pose”—“edit the copy”. A simple motion consisting of several poses is enough to practice how-to at the first time. Take a next step. Note that you insert the reference pose at the end. (Use “N” to set servomotor angles to the reference pose.)

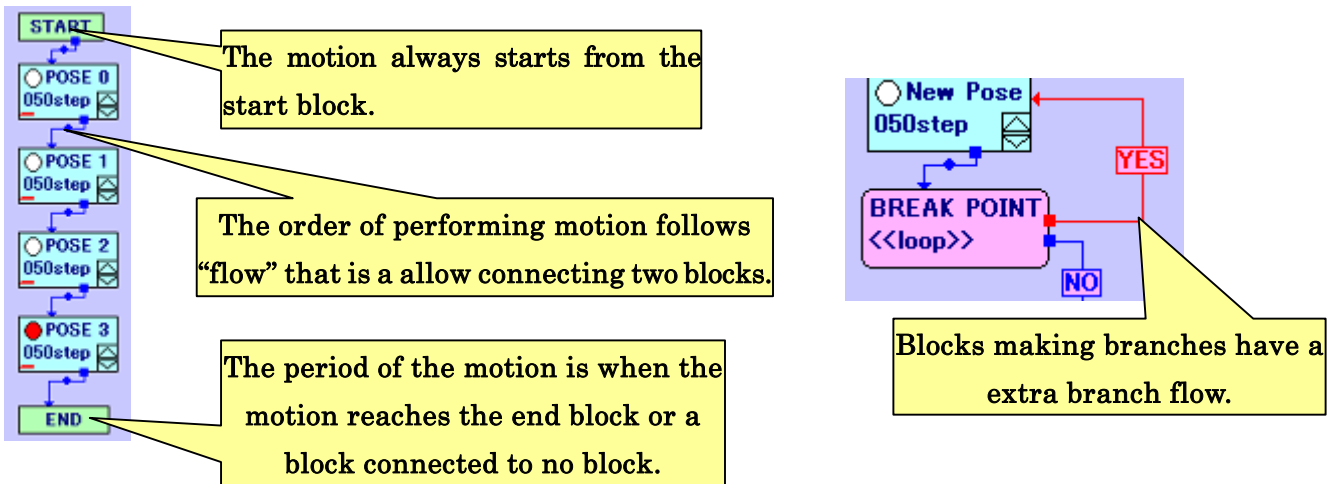


3-3-2. *Playing created motion*

The next step after editing poses is testing the created motion by playing it. To play motions, click “Play”—“Play motions” from menu or “▶” from toolbar. After your click, the robot starts motion from “start block” and play along with the order indicated by several arrows.

In this software, these arrows indicating the playing order of motion are called “flow”. There are two types of flows that are “normal flow” beginning from “pose block” (that is blue arrow by default) and “conditional flow” used for “loop block” or “conditional branch block” detailed below. “Conditional flow” select a branch arrow (that is red by default) when a conditional statement becomes true.

The motion ends when “end block” or a “block with no next flow” are played.

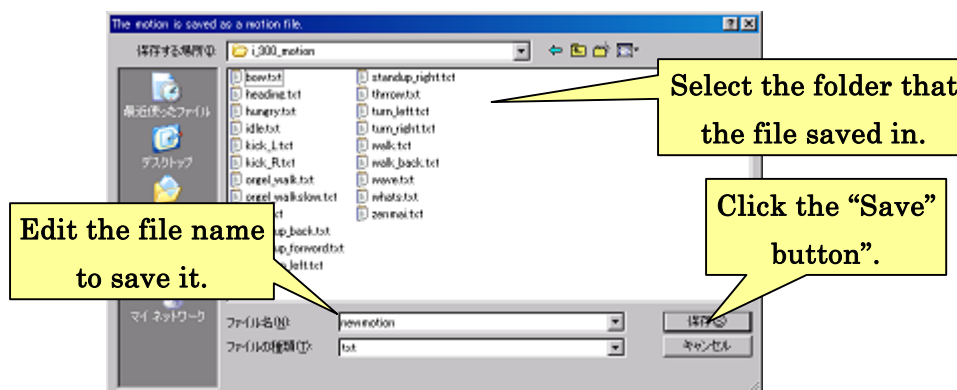


Note that no operation permitted during playing motions except for stopping to play motion or turning on/off servomotors. In order to stop playing motions, use “Play”—“Stop motion” from menus or “■” button from toolbar. When the motion reaches the end, the software automatically goes back from motion play mode to motion edit mode.

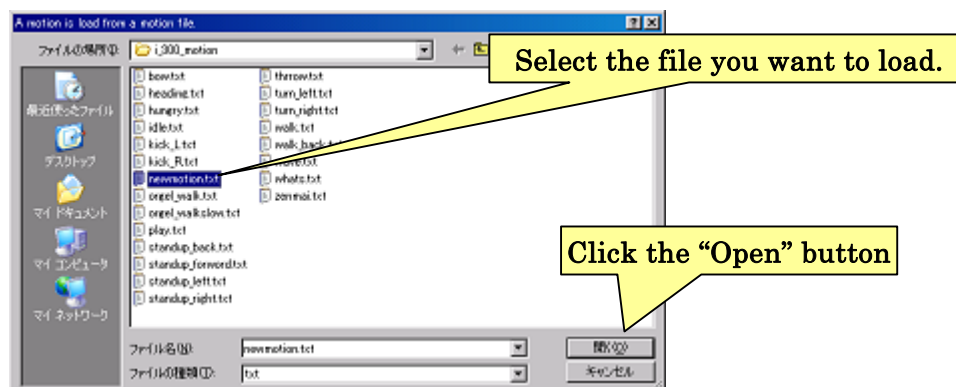
During motions playing, the signals in blocks change to indicate current pose.

3-3-3. Saving/Loading motions

To save motions to a motion file, click “File”—“Save as...” from menus. Select a folder and type name of the motion in opening dialog (see the below figure). Then click “Save” to save it.



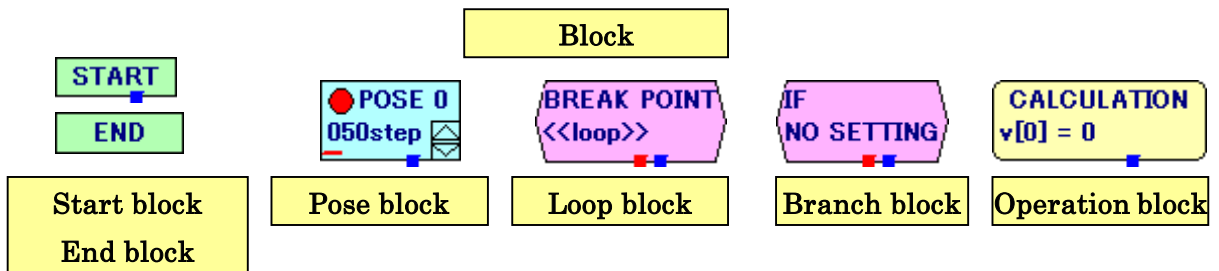
To load motion from a motion file, click “File”—“Open” from menus. Select the motion file you want to load in opening dialog (see the below figure). Then click “Load” to load it.



3-4. Assembling motions

Section “3-3.Creating robot poses” describes sequence procedure to make motion. However, the way for creating motion is still primary in terms that the motions consists of poses ordered at same time interval (or speed) and one order.

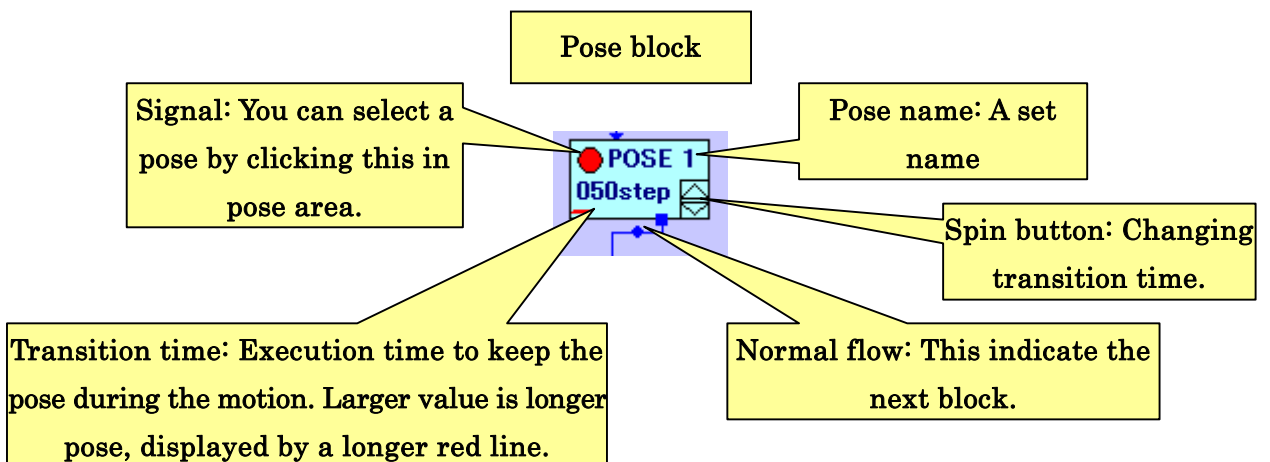
Pose blocks can change its speed and transit orders in Motion Area. There are another several blocks such as conditional branch blocks, loop blocks, and operation blocks (See below figure.). These “blocks” are for creating advanced motion, such as repeating same poses and branching motion due to sensor information.



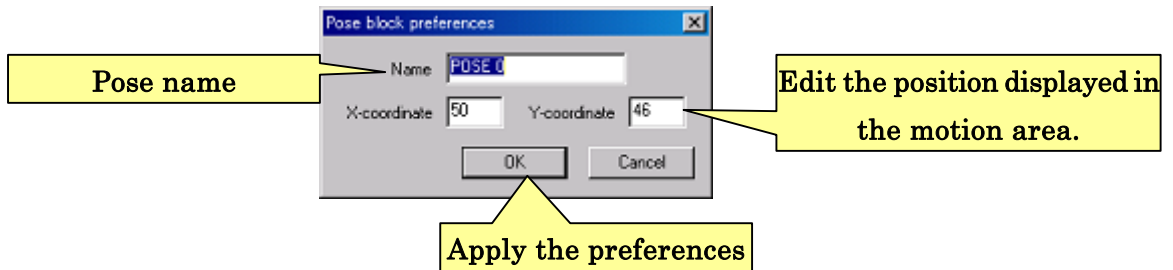
Next section details about each block.

3-4-1. Pose block

A pose block is a registered pose and settings as you have already known about it. The pose block consists of not only the “signal” as stated above, but also “transit time” setting speed of motion and “normal flow” for indicating next block. You can change pose name here.



By double-clicking the pose block, you can configure advanced settings that are the pose name and block position in the opening dialog. After editing them, close the dialog by clicking “O.K.” to apply or “Cancel” not to apply.

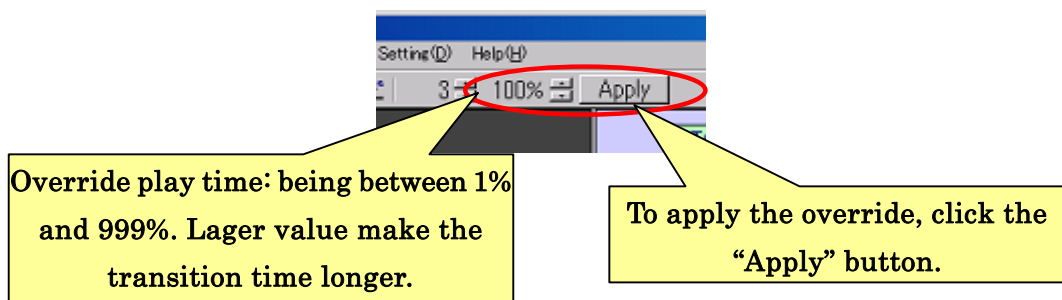


3-4-2. Changing motion speed and play time

A register pose has a setting to change motion speed, meaning how long the pose takes time to transform from previous pose, called “transition time” changed using spin buttons. Shorter transition time accelerates pose transition speed, and longer transition time decelerates transition speed. These settings on each pose fitting the motion enable you to create shape motion and several advanced acrobatic motions, such as “forward roll” and “cartwheel turn”.



Transition time of 060 is about 1 second, because the one unit of transition time corresponding to 60Hz (equal to about 0.01666... sec). (You can change this transition time in the CPU board preferences.) To decelerate (or accelerate) a whole motion, change the percentage of “Override play time” in toolbar larger (or smaller). (See below figure.)



Edit the value of overriding play time using spin buttons, and play motion. Changed transition times override transition time in the whole motion. Since transition times are at the ratio of overriding play time, larger value than 100% stretches transition time. (For example, 200% setting mean 2 times more than the original.)

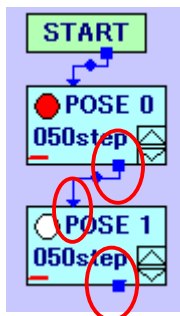
Overriding play time is optional software function during creating motions, the setting is not saved in motion files. In order to apply the setting, click the “Apply” button in toolbar, changing all transition time to new one.

3-4-3. Setting pose order in pose

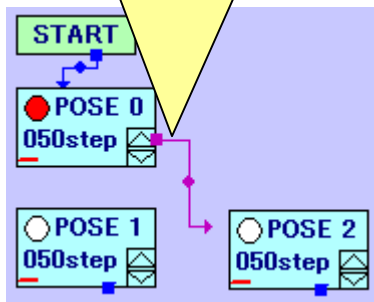
As you see “3-3-2. Playing motion”, pose blocks or another blocks displayed in Motion Area equip arrows called “flow” showing playing order of registered poses. A pose block has one “Normal flow”.

Flow connections can be changed by click the root or the end of arrow and dragging to the different blocks. Turnoff point can be changed by dragging a circle middle of arrow.

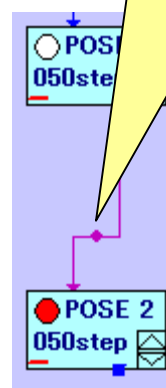
To drag the flow, click the bottom or top of the flow at first



During dragging, the flow's color is changed and automatically connected to the different block.



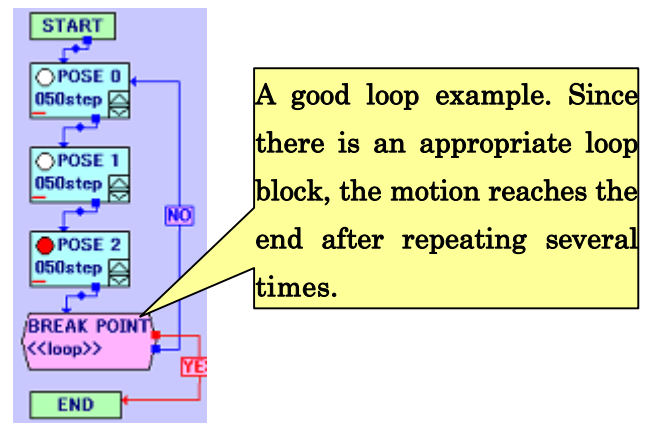
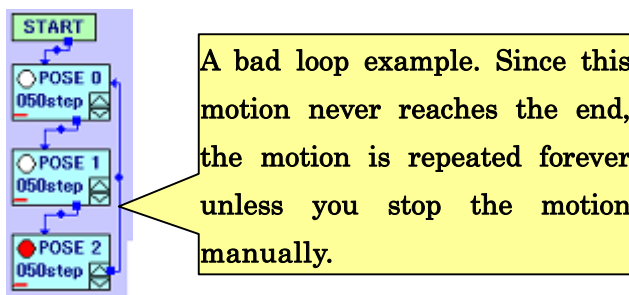
To change the flow shape, drag the circle middle of the flow.



3-4-4. Loop block

Thought you can realize repeating motion using normal flow connecting to the point you want to start the loop, you maybe do not want this wrong loop since the motion never reaches to the end. If you play this kind of motion, click “■” in toolbar to stop motion. Use “loop block” to create correct motion having break point in its loop instead.

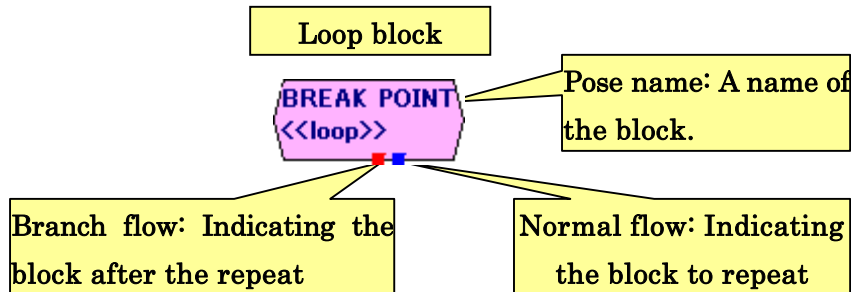
Using “Loop block”, you can make loop of repeating poses breaking after counting loops until set value automatically.



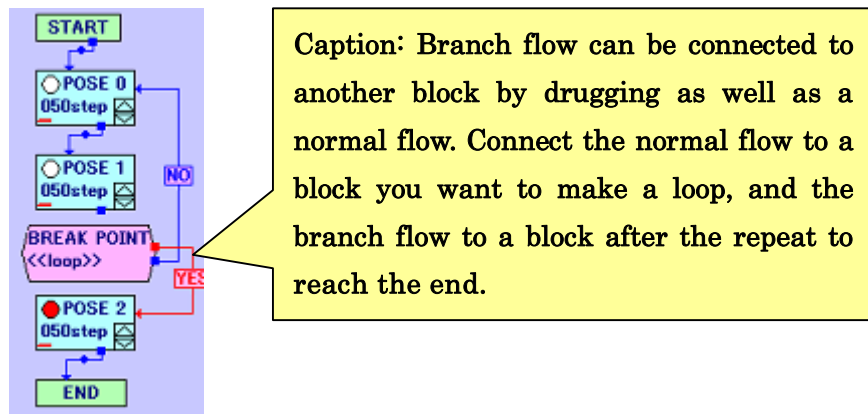
As above figures examples, a structure repeating poses having break point is called “loop structure”, and a structure repeating poses forever is called “infinite loop structure”.

3-4-4-1. Inserting and setting loop blocks

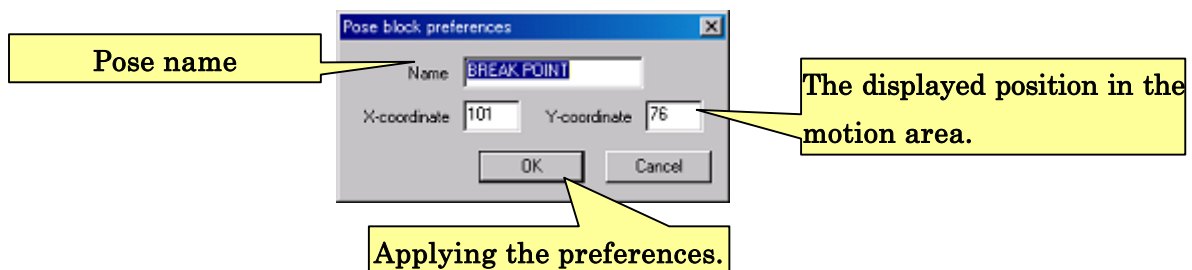
In order to insert a loop block, click “Motion”—“Add loop block” from menus, registering a loop block as below figure.



Loop blocks can change the name and flow connection as well as pose blocks, having a normal flow to make a loop and a branch flow to break the loop.



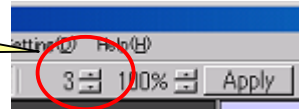
You can open the advanced setting dialog by double-clicking loop block to edit name and displayed position as well.



3-4-4-2. Loop counter

There are three different ways for setting loop counter counting loop of motion in “play from RobovieMaker”, “play from auto demonstration”, and “play from controller operation” respectively. When you play the motion from RobovieMaker, the counter is set using spin button of “loop counter” located on right top of window.

Loop counter: being between -1 and 100.
if you set -1, the motion repeat forever.

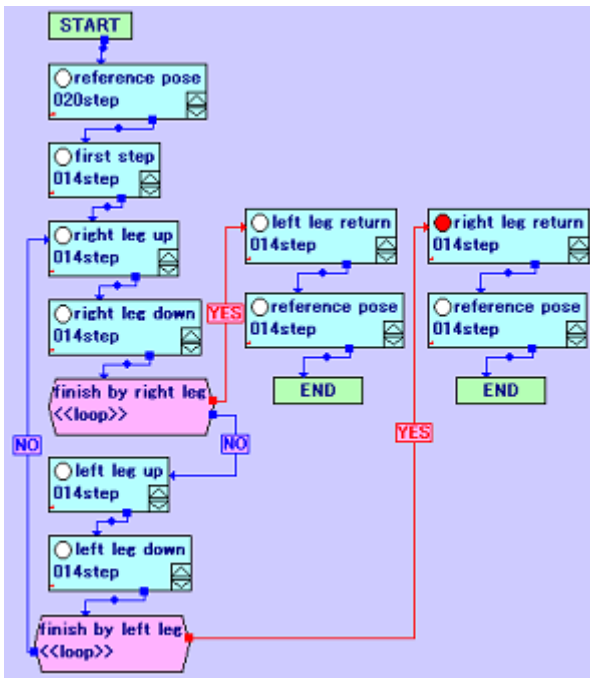


The loop blocks count number of execution times, and jump when the count reaches to the set loop counter. Therefore, if the loop contains several loop blocks, one loop is counted several loops.

In auto demonstrations, loop counts are set for each motion. In controller operations, the motion repeats during keeping the input (, for example, during keeping a button).

3-4-4-3. Example of use of loop structure

The use of loop structure is not only for easy loops but also for tricky combination of several motion ends. An advanced example is as the following figure.



A model case of a bipedal walking motion:
A bipedal walking motion need to start the different leg from the stepped leg at last time. Thus, as the right figure, make two loops and two loops of the motion in cases when the right leg is stepped and when the left is stepped. If this motion ends only when the right (or left) leg is stepped, the walking motion is not property performed because, if the counter set as 1 to make the one step, the steps become two steps.

3-4-4-4. Advantages of loop structure and cautions

In general, a program realizes a loop structure using a variable for the counter. During repeating a part of program, the counter is decreasing and compared to zero. If matching, the program breaks from loop. Robot motion loops in conventional software also use this way to realize loops. However, in controlling, there are demands that users want to repeat motion during pushing a button of controller or that users want to repeat motion predefined number of times; the conventional way could not share motion to deal with these controls, requiring the users to save same contents motions to different files that are difference only in the loop counter.

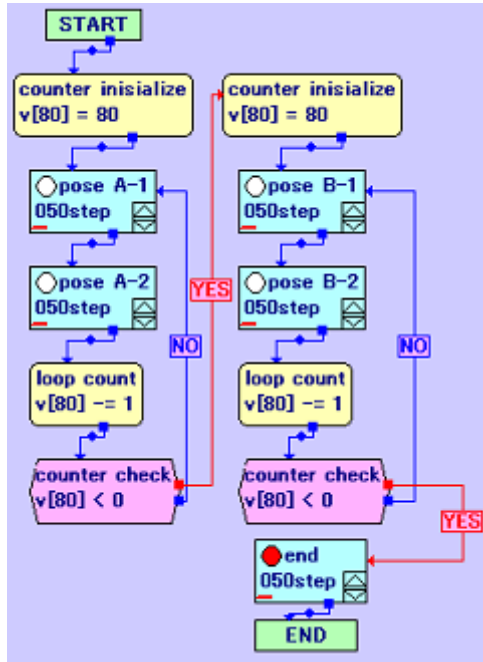
This software uses a loop structure that can change loop counter and conditional branch as to situations demanded in users' controls and auto demonstrations. The advantages are that the users can solve these demands with shared one motion, and that the users do not need to care about counter initializations and controller input judgments.

On the other hand, this software cannot use several loops in a motion, since first loop runs out of the counter, and in other word, sets to zero. Next loop will not repeat the poses but jump to the branch flow. Related with this, nested loop cannot be realized.

In order to edit these several loop combination, use "operation block" and "branch block" enabling you to make loop procedures as a general program.

3-4-5.Operation block

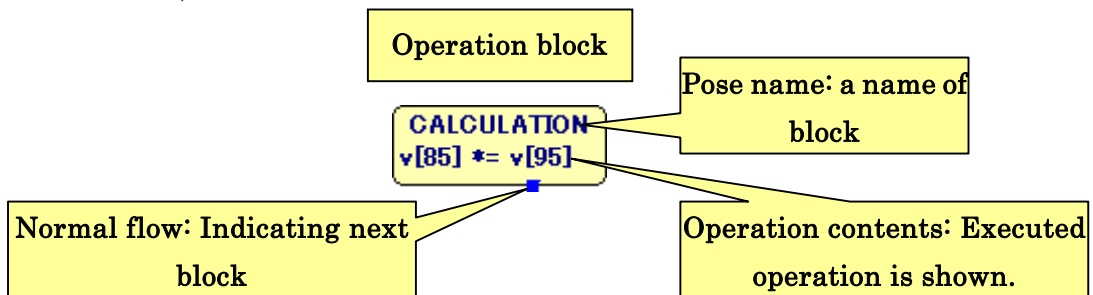
Operation block is for operating variable during motion playing. You can make much advanced motions by inserting this operation block. For example, by assigning a counter to operation block, you can create nested loops. By combination of conditional branch block detailed below, you can realize several loops in one motion.



An example that one motion has multiple loop structures

3-4-5-1.Inserting operation block

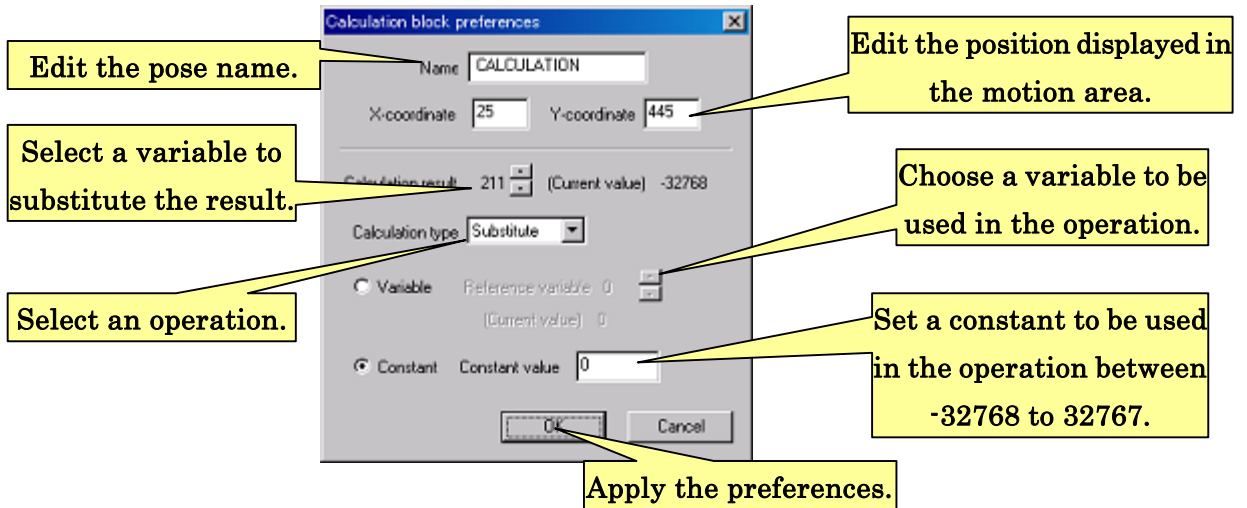
Click “Motion”—“Add operation block” from menus to insert an operation block in Motion Area. Then, one motion block is inserted in Motion Area.



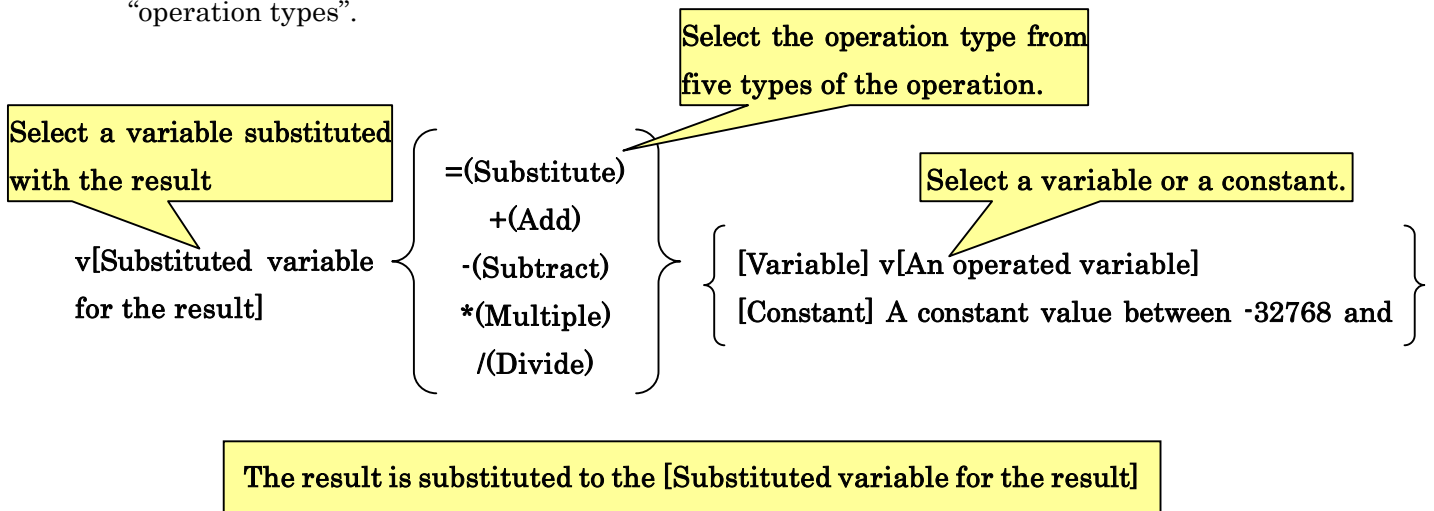
You can edit pose name and flow setting of operation block as well as pose block. The operation block shows an operation executing at the play.

3-4-5-2. Setting for operation block

Double-click the operation block to open a dialog for editing settings. The upper side of dialog is for name and position settings as well as a pose block. The lower side is for operation preference.

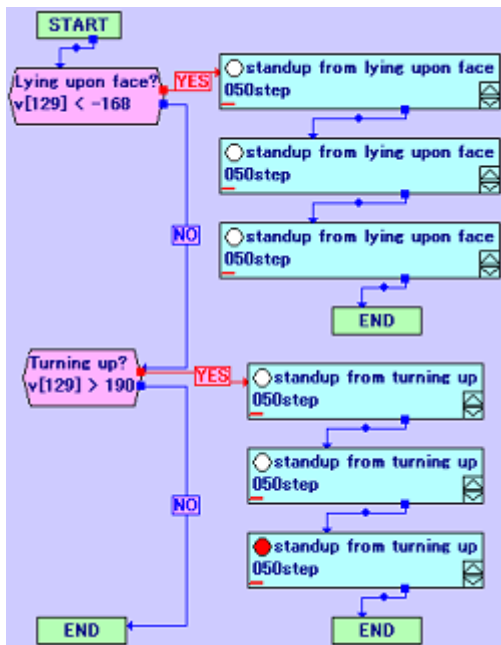


Several operations are available in operation blocks as the below figure. An editable number at the left side dialog is for assigning number to "results variable" holding an operation result. The "result variable" substitutes selected operation result calculated from itself and a user define variable or constant. You can select the operations from "operation types".



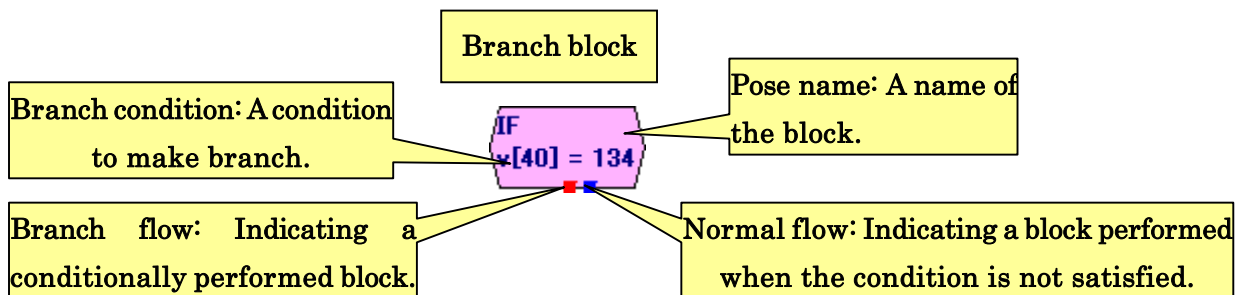
3-4-6. Conditional branch block

Conditional branch block has a function to make branch flows based on user define conditions. This block possesses a function similar to a loop block in terms of making a branch flow based on a condition. However, the condition is not just for breaking loops but also for checking a condition expression consisting of arbitrary variables. For example, you can make a motion consisting of two types of motions conditionally started weather if the robot lay on the face or the back, detected from an accelerometer state. You can execute this motion safety whenever the robot falls down by only pushing a button of controller, because an appropriate motion is selected based on laying condition by referring the accelerometer sampling at the beginning.



A waking up motion using branch blocks. An automated waking up motion is possible with one button event by putting two types of waking up motions branching depending on accelerometer status to recognize face-up lying or face-down lying.

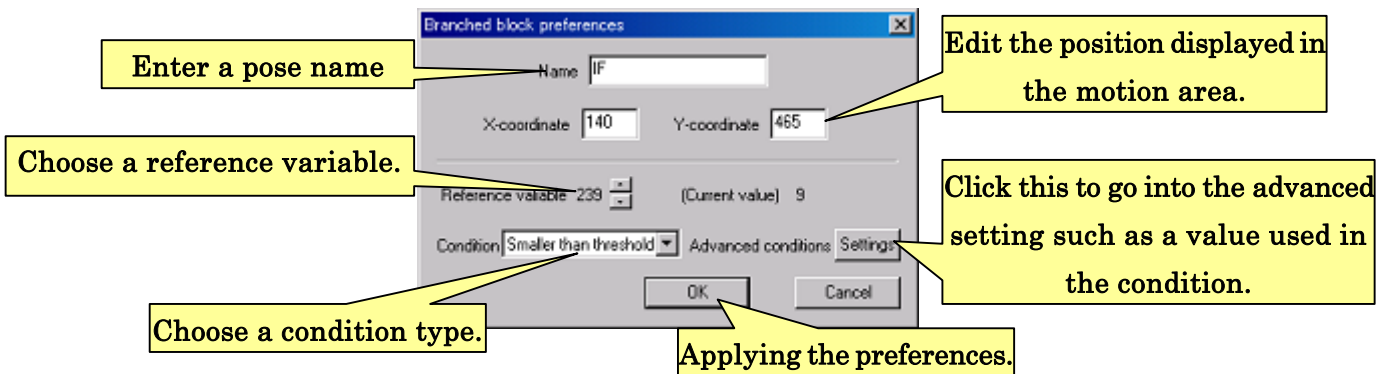
To add a conditional branch block, click “Motion”—“Add condition branch block” from menus. A conditional branch is added on Motion Area.



The Operation block has settings of a pose name and a normal flow as well as a pose block, and displays an operation content executed when the motion plays.

3-4-6-1. Conditionl branch block settings

To change a condition from a conditional branch block, double-click the conditional branch block to open a dialog as below. In upper side of the dialog, you can edit a block name and position as well as pose blocks. In lower side of the dialog, you can set advanced settings for branch condition.

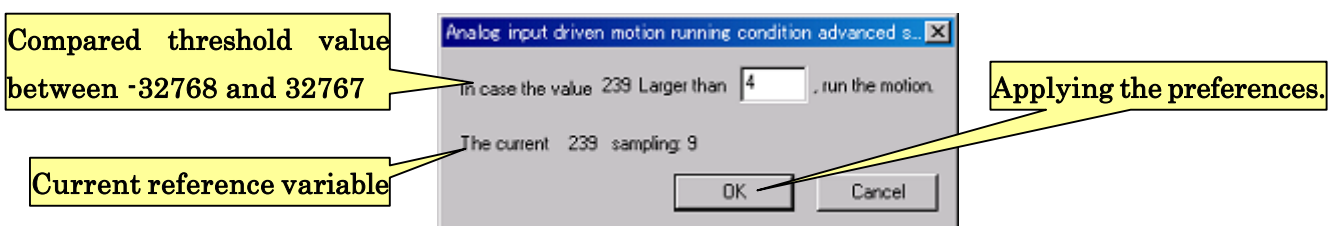


A conditional branch block branches “when an attribute variable satisfies a condition”. For branch condition settings, you can choose a referenced variable from items of “reference variable” and select a condition type from “condition type”. Condition types you can select in the “condition type” are as follows.

- Condition type**
- [Larger than threshold] Execute branch flow when chosen variable is larger than the threshold.
 - [Smaller than threshold] Execute branch flow when chosen variable is smaller than the threshold.
 - [Bit check] Execute branch flow when the masked variable bit all corresponds to set bits.
 - [Equal to threshold] Execute branch flow when the variable is equal to the threshold.
 - [Not equal to threshold] Execute branch flow when the variable is NOT equal to the threshold.
 - [Always true] Execute branch flow at any time
 - [Always false] Never execute branch flow at any time

Depends on the condition type selected in “Condition type”, you need advanced settings. Click “Setting” button when you want to enter the advanced setting dialogs.

If you select the condition types of “larger than threshold”, “smaller than threshold”, “equal to threshold”, and “not equal to threshold”, the following dialog will open.



If you select the condition type of “Bit check”, the following dialog allowing you to edit enabled bits in the variable and compared bits for the condition.

The screenshot shows the 'Analog input advanced setting' dialog box. It contains the following information:

- (1) Current value of 4 0001_0011
- (2) The result that only enabled bits are extracted from the current value 0000_0000
- (3) Motion running condition value 0000_1000

Below this is a note: "When the motion running condition value equal to the result that only enabled bits are extracted from the current value, the system run the motion."

The dialog is divided into two main sections:

- Enable bit preferences:** A table with columns for bit numbers 7 through 0. The 'Set enable' row has checkboxes for bits 7, 6, 5, 4, 3, 2, 1, and 0. Bits 3 and 2 are checked.
- Bit condition to be checked:** A table with columns for bit numbers 7 through 0. It has two rows: 'Set 0' and 'Set 1'. Each bit has a radio button. In the 'Set 0' row, bits 7, 6, 5, 4, 3, 2, 1, and 0 are all selected.

At the bottom are 'OK' and 'Cancel' buttons.

Callouts from yellow boxes point to specific parts of the dialog:

- "Select referred bits" points to the 'Enable bit preferences' section.
- "Choose status of referred bits." points to the 'Bit condition to be checked' section.
- "Applying the preferences." points to the 'OK' button.
- "If the 'Status of extracted enabled bit' matches to 'Motion running bit condition', the condition satisfied." points to the result line (2).

In “enable bit preferences”, you can choose bits you want to use for the comparison. In “motion running bit condition”, you can set referenced bit conditions. During keeping the communication between the CPU board and your PC, the current variable is masked to show only enabled bit status under the “Status of extracted enabled bit”, and is compared to bits status set in “Motion running bit conditions”.

3-4-7. Start block and end block

Start block and end block are the blocks to indicate motion start and end.

Only one start block exists in each motion. Motions always start from the next block to the start block. The start block is automatically added when you create a motion, and is not possible to be deleted and created in the motion. The flow from the other blocks cannot connect to the start block.

The diagram shows a green rectangular block labeled 'START' with a small blue square at its bottom center. Two callout boxes point to it:

- On the left: "Start block: You cannot edit the name. The start block cannot be connected from another blocks."
- On the right: "Normal flow: Indicating a block to be performed at first."

To add an end block, click “**Motion**”—“**Add end block**” from menus. The following end block is added to Motion Area after the click. Connect pose block to the end block when you want to make the end of the motion.

End block: You cannot edit the name. The end block has no flow while a flow is connected to it.

END

Note that you cannot change the name of the start block and the end block. No dialog opens with your double click on it. Your mouse dragging can change the coordination of the blocks.

As stated above, you have finished learning basic knowledge for creating motions. Custom makes all things easy – challenge various types of motions, and you will be able to create advanced motion such as a bipedal working.

3-5. Advanced useful functions to create motions

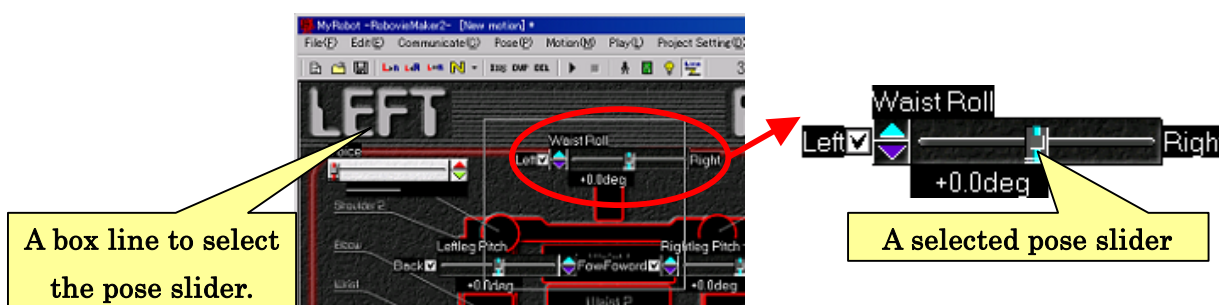
This section details about advanced useful functions allowing you to create motions rather conveniently.

3-5-1. Pose area advanced control

In the pose area, you can copy and paste pose slider values, or mirror the left/right slider values to the right/left sliders using “right/left settings”. Each function details below. Note that you will sometimes see the terms related with “right/left setting” and “step resolution” described in “3-6. Advanced setting of pose slider”. Refer the section when you meet the unknown terms.

3-5-1-1. Selecting pose slider

In motion area, select a rectangle region by dragging your mouse from the outside of pose sliders. A rectangle region is displayed during dragging mouse. The color of pose sliders contained and overlapped in the rectangle region changes. This state called “selected” state. You can manipulate pose sliders with selected state in several ways.



3-5-1-2. Using mouse wheel

Select pose sliders, and role the mouse wheel with mouse cursor inside the pose area. All selected pose sliders move at the same time. Regardless of pose slider settings, forward rolling moves pose sliders to the right, and backward rolling moves pose sliders to the left.



You can use this function, for example, to create symmetric poses by wheel manipulation of right and left joints at the same time. You can create leg bend or leg stretch pose easily using wheel manipulation with pitch axis of legs selected as well.


Note that, as stated above, the movement directions of pose slider are always same regardless of “mirror display” or “inverse sign” settings. Switch the flag of “mirror display” to inverse the real direction for creating symmetric poses depending on the axis types you want to move at the same time.

You can switch the flag of “mirror display” by clicking “switch to mirror display” from the hop-up menu. The menu is shown by right clicking in the pose area.

3-5-1-3. Inversing and mirroring using right-left settings



You can copy or interchange the right and left poses using “right-left settings” of properties of pose sliders.

To copy the pose of right body to the left body, click “Pose”—“Copy right to left” from menus, or  from the toolbar. After clicking, the pose of the left and the right should be same. Vice versa, when you want to copy the pose of left body to the right body, click “Pose”—“Copy left to right” from the menus, or  from the toolbar.

To interchange the right and left poses, click “Pose”—“Mirror inversion” from menus, or  from the toolbar. After clicking, the pose is interchanged.

To use this function with pose sliders selected, selected sliders and their pairs are the manipulation targets. Without pose sliders selected, the slider bars, enabled in the setting of “Enable slider”, are the targets.

3-5-1-4. Copy and paste of pose sliders

To copy software’s buffer, click “Pose”—“Copy joint value” from menus, or  from toolbar with pose sliders selected state. To paste the pose slider value, click “Pose”—“Paste joint value” from menus, or  from tool bar.

3-5-1-5. Changing positions of pose sliders

To change the position of a pose slider, select the pose slider and drag with pushing SHIFT key on the keyboard to new position. Note that, if you click the position of the knob or the spin buttons, the slider bar is unselected. Drag the pose slider using another position.

3-5-1-6. Changing step resolution temporary

Step resolutions can temporary be changed using keys on the keyboard. To increase step resolutions 16 times more the original, push the SHIFT key. To decrease step resolutions 1/16 times less the original, push the CONTROL key.

3-5-1-7. Output audio setting

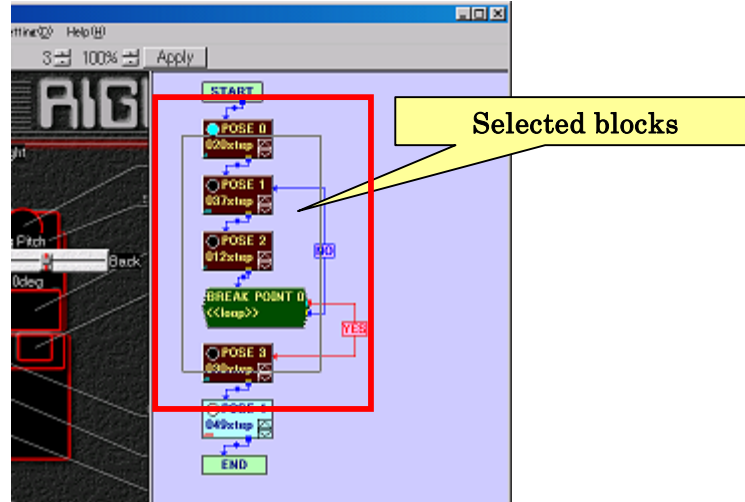
As stated in “3-1-1. Pose sliders”, the output audio from the CPU board can set using 30th pose slider. To assign an audio file to a pose, change the slider value to the file number. The assigned audio file is synchronized to the pose, and output it from an attached speaker on the CPU board.

Note that, to output the audio file, you need to connect a speaker to the CPU board. About the audio file settings written to the CPU board, see “4-4. Mode switching and audio map preference, and writing to CPU board”.

Also, note that the two audio files cannot play continuously in the two poses. To play the 2nd audio file after the 1st audio file, insert a pose assigned with audio number 0. Alternatively, save same audio file as different file name, and assign a different audio number on it. Then write the setting to the CPU board.

3-5-2. Motion area manipulation

In motion area, you can select blocks to move, copy & paste, and delete them as well as pose sliders' manipulations.



3-5-2-1. Moving blocks

To move a block, select the block and drag to the desired position. By selecting several blocks, selected blocks move at the same time.

By dragging blocks with the SHIFT push down, the blocks move along with the grid defined in every 10 pixels. To change the grid size by pixel, see “6-3. Another system related preferences”.

3-5-2-2. Copy, paste, paste, and cut blocks

To copy and paste a block in motion area, select the block and click “Motion”—“Copy selected block” from menus. This manipulation copies the block to a buffer of the software. Then click “Motion”—“Paste copied block” from menus. The copied block in the buffer will be pasted to motion area.

To delete selected blocks, click “Motion”—“Delete selected block”, and the blocks will be deleted.

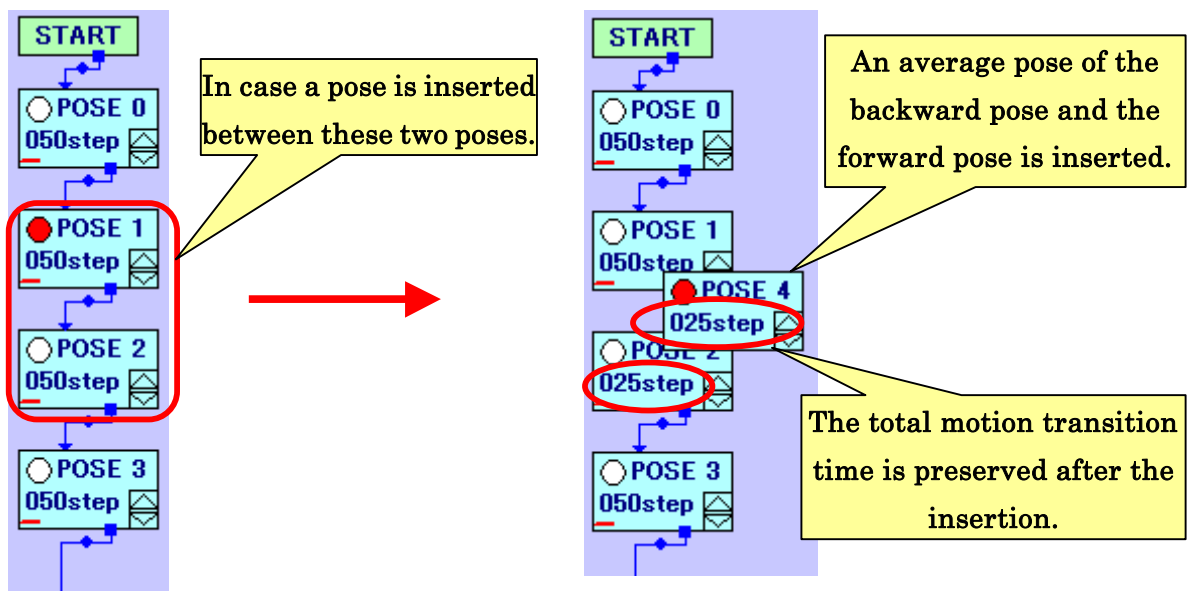
You can also use shortcut keys with this manipulations as well as a text editor. Move the mouse cursor into inside motion area, and you can use “CTRL+C – copy selected block”, “CTRL+V – Paste copied block”, “CTRL+X – Cut selected block”, and “DEL – Delete selected block”.

***Note that you need to keep mouse cursor in the motion area to use the short cut manipulation.**

3-5-2-3. Inserting a intermediate motion in a motion

Using “Duplicate pose”, you can duplicate an editing pose to resister it after the pose. You can also insert an intermediate pose between two poses having serial motion numbers. This intermediate pose insertion enables you to edit details of the motion at the time.

To insert an intermediate pose, click “Motion”—“Split pose in editing” from toolbar, or “INS” button from toolbar. This manipulation create a new intermediate pose of pose in editing (, that is a pose block having the unique color,) and the next pose (, that is a pose block having an 1 increased pose number), and insert the created pose between them. Pose slider positions of the created intermediate pose are the means of referenced two poses. In addition, the transition times of the first pose and inserted pose become to half time of original transition time of the first pose for keeping whole motion length.

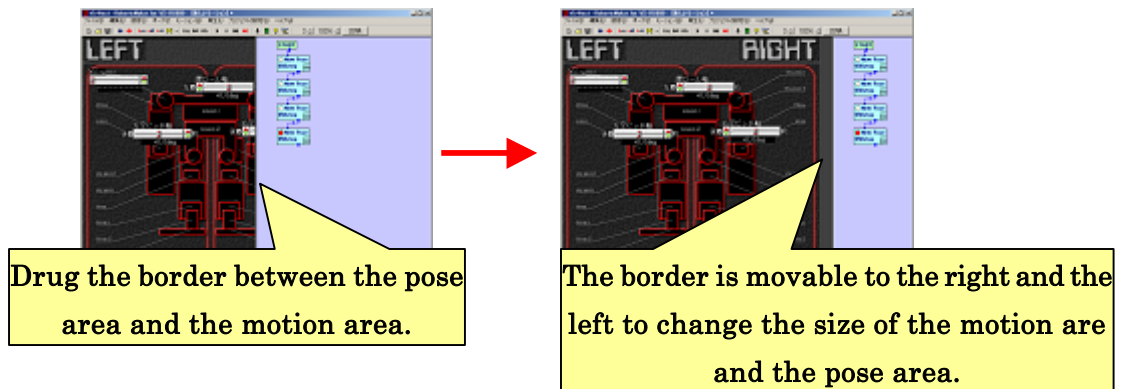


3-5-2-4. Playing motions frame by frame advance

You can view and play a motion one pose by one pose like a single step function in video. Click “Play”—“Next pose”, or “>>” from toolbar. The motion moves to the next pose connected with a normal follow. To move through branch flow, click “Play”—“Next pose (branch)”, or “>>” from toolbar.

3-5-2-5. Changing size of pose area and motion area

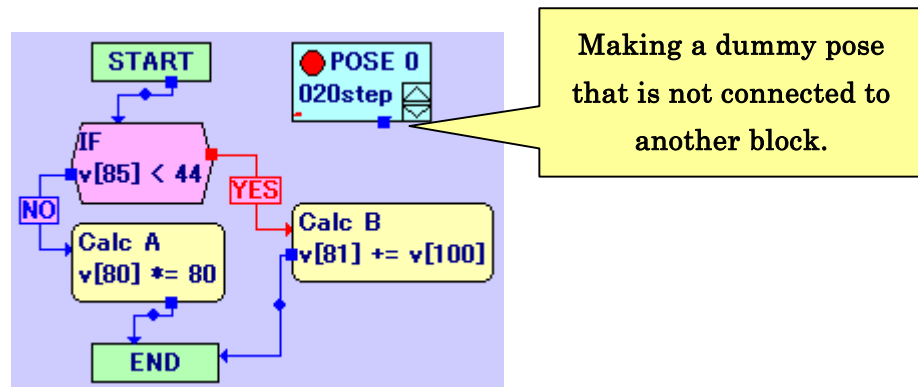
To change the area sizes, drag the boarder line between the pose area and the motion area. By double-clicking the motion area or the pose area, clicked area automatically expands its size to the maximum and the other becomes the minimum size. After this manipulation, the same double-click undo the manipulation, meaning the area size back to the original size. Using this function, you can conveniently maximize the pose area during creating poses, and maximize the motion area during assembling the motion vice versa.



3-5-2-6. Creating variable computation and branch only motion

This software constrains the motion to be registered one or more poses. However, this does not mean that the one or more poses must be executed. Using this rule, you can create the motions that branch or compute variables.

To make this type of motion put only one pose, that is a dummy pose, in the motion and delete all pose block. Then, add and assemble desired operation blocks and branch blocks in the motion area.



3-5-2-7. Combining multiple motion files

Click “File”—“Import motion files” from menus to open a motion file, the selected motion is appended to the motion in edit. Using this function, you can easily create a new motion combining multiple motions.

Note that the appended motion data using “Import motion files” no longer has link to the original file. Therefore, even if you edit the imported motion, the original motion saved as the motion file never changed with this import.

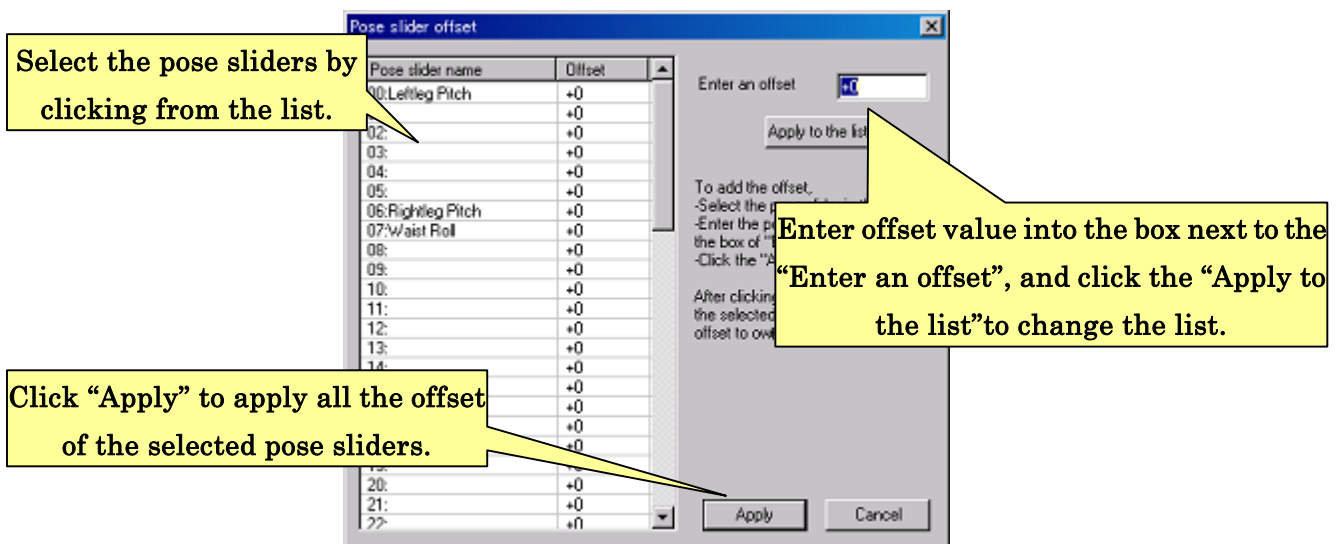
3-5-2-8. Mirror projection of whole motion at once

To change selected poses to mirror projected poses, click “Motion”—“Mirror projection of selected poses” with the pose blocks selected. Using this function, you can easily create symmetric motion.

3-5-2-9. Offsetting pose sliders of multiple pose blocks

To offset multiple pose sliders from multiple pose blocks, select multiple pose blocks, and click "Motion"--"Offset selected poses" from the menus. The following dialog box appears after clicking. In the left side of the dialog, the offset (incremental/decremental) values are shown. After selecting the pose slider you want to change, edit the value next to “Enter an offset”, and click the “Apply to the list” button. After checking that the list changes, click the “Apply” button to apply the offset all the pose slider values.

Note that, regardless of displaytypes of pose sliders or being plus or minus, the offset is applied using raw value of thepose sliders.



3-6. Pose slider preferences

You can change the settings, such as display types and manipulation methods, from the pose slider preferences.

Appropriate preferences usually automatically set by default based on specified robot type when you created the robot project. This preference should be changed when you append a servomotor, expanded board, or use your original robot. Change the preferences appropriately using following examples as reference.

The pose slider preferences can set from the “Pose slider preferences” dialog. To open the dialog, click “Pose”—“Pose slider property” from the menus.

The image shows a screenshot of the 'Pose slider preferences' dialog box. The dialog is titled 'Pose slider preferences' and has a dropdown menu at the top showing '07:Waist Roll'. Below this, there are four main sections: 'Text setting', 'Numerical setting', 'Right-left preference', and 'Slider bar position and flag setting'. Callouts point to various fields: 'Edit a name displayed around the pose slider' points to the 'Slider bar' field; 'Edit a value format of the pose slider.' points to the 'Format' radio buttons; 'Edit a right/left attribution of the pose slider.' points to the 'Attribution' radio buttons; and 'The position of the pose slider and the display format except for fonts.' points to the 'Display coordination' fields and the 'Enable slider bar', 'Inverse display', and 'Reverse bits' checkboxes.

Choose a pose slider for the configuration to be changed.

Edit a name displayed around the pose slider

Edit a value format of the pose slider.

Edit a right/left attribution of the pose slider.

The position of the pose slider and the display format except for fonts.

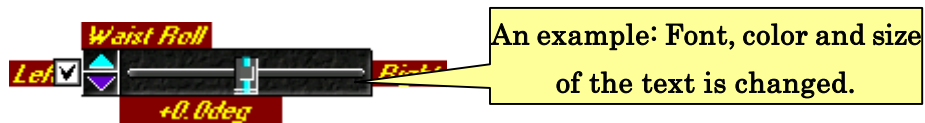
Select pose slider that you want to change the preferences from the dialog, and you can edit 4 categories of preferences that are “Text setting”, “Numerical setting”, “Right-left setting”, and “Slider bar position and flag setting”. The categories are detailed later.

After edit, click “Apply”, and the new settings will be applied. To close the dialog windows, click close button.

To open the preference dialog of a pose slider, right-click on the pose slider in the pose area, and you see a hop-up menu. Then, click “Property of pose servers” from the menu. To change the preferences of multiple pose sliders, open the preference dialog with the pose sliders selected.

3-6-1. Text preference

In the text preference of the preference dialog, you can set character sets to display around the pose slider. You can enter texts into “Slider name” to be displayed on the top side of the pose slider, and “Characters beside of the slider bar” to be displayed on the right side and the left side of the slider bar. Input the event type when you drag the slider bar to the right or the left, such as a moving direction of the servomotor. To change the character size, click “Font” button. To change the font sets or the font colors, click “Background color” button.



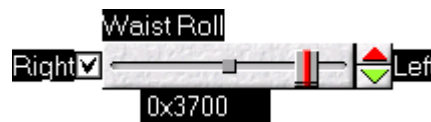
3-6-2. Numerical settings

In the numerical settings, you can set a pose various categories of numerical pose slider setting, such as the pose slider range, numerical type, and display type.

“Format” is a setting to decide displayed type. The preferences and its display examples are summarized below (, except for “Audio” the pose slider values are same).



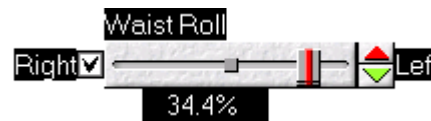
Decimal: The variable is displayed by decimal notation.



Hex: The variable is displayed by hexadecimal notation.



Degree [deg]: The variable is displayed after converted to a degree with the resolution configured in the “Ratio of value to 1 degree”.



Percentage: The variable is displayed after converted to a percentage with the range configured in the “Limitation of movable range”.



Audio: A sound file name is displayed corresponding to a played audio.

When the format is “Degree (deg)” or “Percentage”, you can set displayed decimal place (within 0 to 2). When the format is “Degree (deg)”, you can set the ratio of the one unit of pose slider and 1 degree displayed on the pose slider in “ratio of value to 1 degree”.

“Step resolution” is for a setting of a pose slider resolution. When you use the spin button or the knob of the pose slider, the differential of the pose slider value is always dividable by the step resolution value. Note that, regardless of selected format, this resolution unit is always decimal value.

3-6-3. Right-left preference

In the “Right-left preference” section, you can define the each servomotor right or light side of the body, and the pair of pose slider. These settings are not always necessary, but it is convenient for the users to copy pose sliders from the right to the left, and to interchange the left and the right pose sliders to get a mirror projection pose.

In the “Attribution” section, you can choose the pose slider types as for right-left setting, that are 4 types of attributions as follows.

- **No setting** – This setting is for the servomotor without right-left setting, such as a pitch axis of the center of body.
- **Center** – This setting is for a yaw or roll axis of center of the body, such as yaw and roll axes of the waist, a yaw axis of the neck. The pose sliders with this setting invert its value based on the reference pose when you use reverse the pose, but nothing happens when you copy right side pose to the left side.
- **Right** – Select this to define the servomotor as the right side of the body, and choose the “Paired pose slider” so that the pose slider can be copied or interchanged to the paired pose slider.
- **Left** – Select this to define the servomotor as the left side of the body, and choose the “Paired pose slider” so that the pose slider can be copied or interchanged to the paired pose slider.

When you select “right” or “left” attribution, the paired pose slider attribution automatically defined as the other one to keep the pair relation.

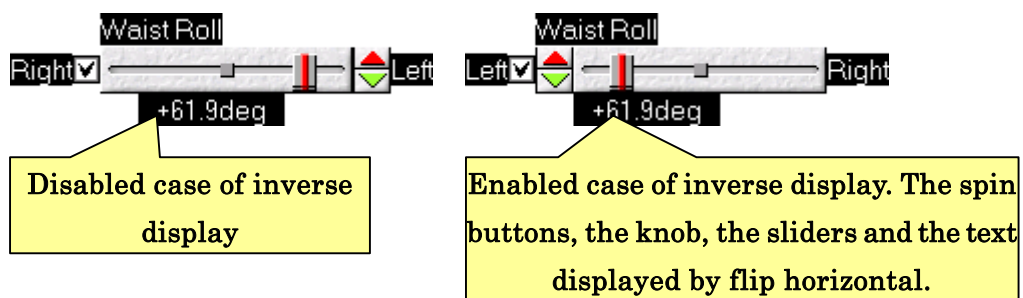
3-6-4.Slider position and flag settings

In the slider position and the flag settings, you can change displayed position of pose sliders, and a display type setting being different from the “numerical setting”.

To change the displayed position of pose slider, enter position in “Slider bar position and flag setting”. The position must be positive real values, and is automatically adjusted when you enter the position over the range of display area. Coordination of (0, 0) is the top-left of the dialog. The right-bottom of the dialog has a larger coordination.

In the “Flag settings”, you can set the display type of the pose sliders. Check the checkbox to enable the setting, and uncheck when you want to make the checkbox disabled. Each flags are as follows.

- **Enable slider bar** – To enable the pose slider, check this box. Only enabled pose sliders are displayed and can be manipulated. When you connect new servomotors to the CPU board, enable this flag to control them.
- **Inverse display** – To inverse the pose slider including spin buttons and sliders, check this box. The inverted pose slider is displayed like on the mirror as following figure.



- **Reverse sign** – To reverse sign of the displayed value, check this box.
- **Reference pose relative value** – To display the pose slider variable as a differential value from the reference pose, check this box. If checked, the pose slider shows the relative coordinate from the reference pose as the point of origin.

- **Display pose slider name** – To display pose slider name on top of the slider, check this box.

“Inverse display” and “Reverse sign” are not only for changing display type but also for changing direction when you manipulate pose slider with the mouse wheel. See “3-5.Advanced useful functions to create motions” to get more information.

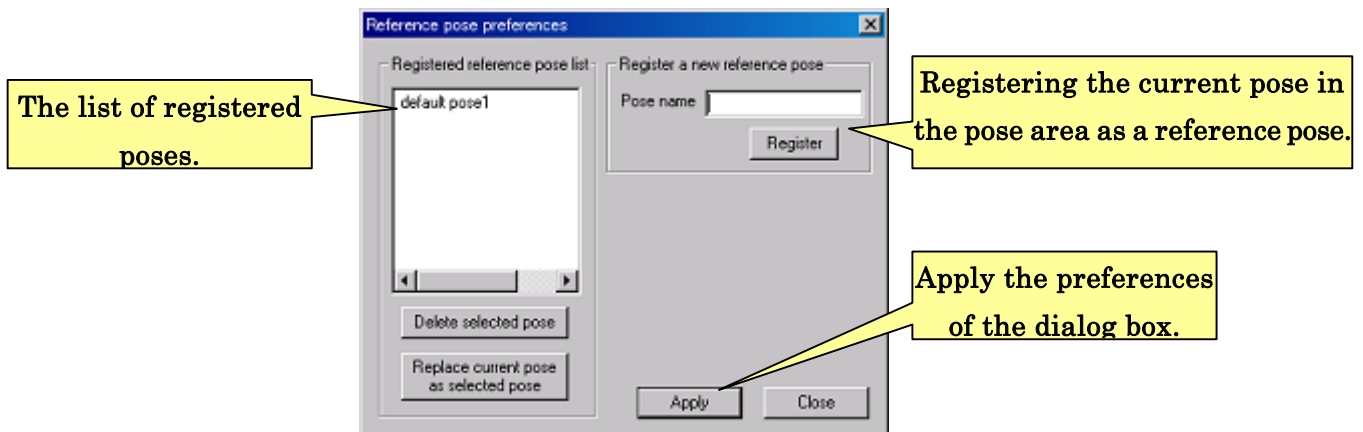
3-7.Reference poses preferences

The reference pose is automatically configured when you create a new robot project based on a selected robot type. Additionally, this software can change the reference pose and register multiple reference poses to switch them.

Switching reference poses is useful when you create new motions depending on its category, such as “battle mode” and “soccer mode”. The motion creation and the adjustment of servomotor become easier.

To register a new reference pose, or change the current reference pose, at first, create new reference pose using pose sliders. Adjust pose sliders to reverse signed value or same value as much as possible.

After creating the new reference pose, click “Pose”—“Register reference pose” from menus. The following dialog will be open.



To register the new reference pose, enter the reference pose in “Pose name”, and click “Register” button. After clicking, the reference pose is added to the list located on the right of the dialog.

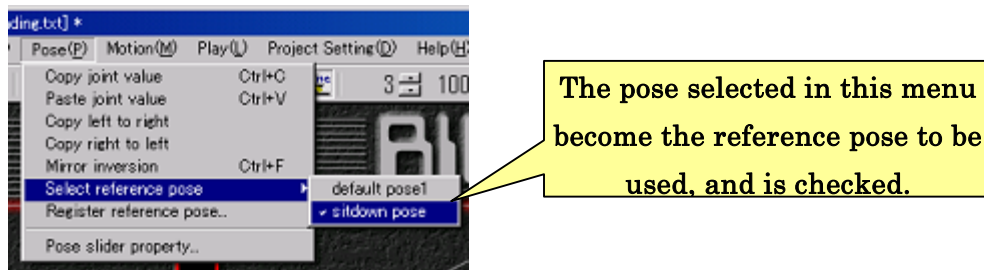
To replace the registered reference pose to current pose in the pose area, select the reference pose registered in the list located on the left of the dialog, and click “Replace

current pose as selected pose” button.

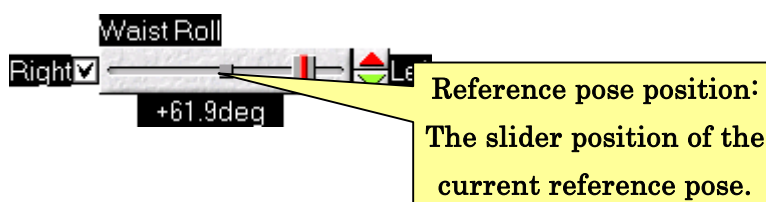
To delete the registered reference pose, select the reference pose registered in the list located on the left of the dialog, and click “Delete selected pose” button.

To apply these manipulations, click “Apply”. To close the dialog, click “Close”.

To adjust current pose sliders to the registered reference pose, click the registered pose from the shown list on “Pose”—“Select reference pose” from menu (see below figure). After clicking, the pose sliders are set with the values of the reference pose.



A pose slider value of the current reference pose is shown in its slider as a reference pose point. (See below picture.) Since the servomotor’s calibration depends on the current reference pose, set the pose sliders to the current reference pose before starting the calibration.

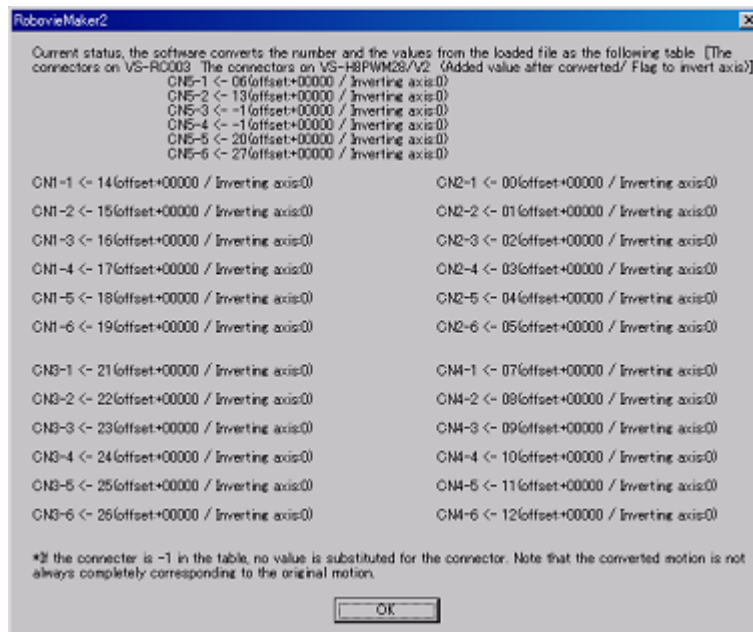


3-8. Importing motion file of VS-H8PWM28V2

3-8-1. File Import

This software can import motion files for the perversion CPU board “VS-H8PWM28/V2”. Transition time of imported motion file converted from 50Hz to 60Hz because of clock differences. The axis numbers also converted using the predefined table. You can add the reverse sign and offset to this converted motion file. Note that this convert function is for importing old formatted motion file to this software, meaning the original motion is not always imported.

To import a motion file of VS-H8PWM28/V2, click “File”—“Import VS-H8PWM28/V2 format motion file”. Current setting for converting motion will be displayed on the following dialog.



The dialog shows following setting contents; “The number of the port for servomotor in CPU board”, “Offset value”, and “Reverse or not reverse”. Click O.K. after confirming contents of the settings to open a file select dialog. Choose the motion file, and click “open”. The motion file of VS-H8PWM28/V2 will be imported. To save the imported motion, use not “Save” but “Save as ...” because the original file name no longer preserved.

3-8-2. Changing setting of motion file import

To change setting of motion file import, put a setting file named “**medit_loadoldmot.ini**” in the root directory of the robot project. (Some projects have this setting file in the root directories.)

The settings of “medit_loadoldmot.ini” are as follows. This setting file consists of multiple setting texts, such as “***_AXIS_CNVSETTING[1]-[23]-[9810]-[0]***”, each one line is corresponding to a sequence of settings that are “**Which servomotor output is destination**”, “**Which servomotor output is source**”, “**How large the offset is**”, and “**Reverse imported value or not reverse**”.

“***_AXIS_CNVSETTING***” – this is a prefix string to indicate this line as a convert setting. When a line in the setting file consists from this string, the line is regarded as a convert setting line.

First parameter – this value is the number of servomotor output in VS-RC003. The parameter should be between 0 and 29.

Second Parameter – this value is the number of servomotor output in VS-H8PWM28/V2. The parameter should be between -1 and 27. -1 means to ignore the servomotor (that is not converted).

Third parameter – this value is the number of servomotor output in VS-H8PWM28/V2. The parameter need to be in the range between -32768 and 32767 (signed 16bit). Since VS-H8PWM28/V2 motion data is between 0 and 255, at first, the converter subtracts 128 from the data to make the values between -128 and 127. The value is multiplied by 256 resulting in same range of the CPU board. The offset value is added to the value.

Forth parameter – this value decide if the converted value reverses its sign or not. The value should be 0 or 1. 0 means that the sign is not reversed, and 1 means the sign is reversed. The sign reverse is operated for the result after offset added.

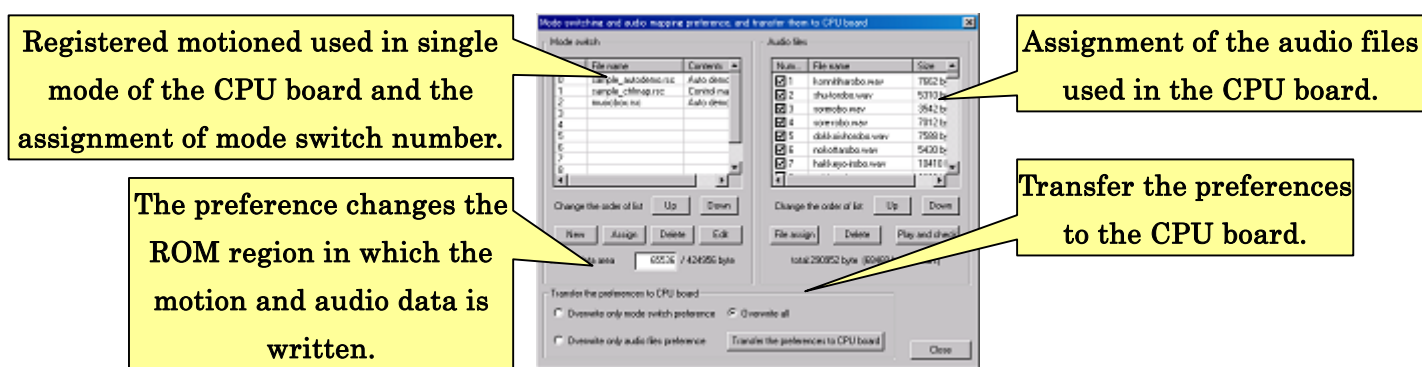
For example, the line of “***_AXIS_CONVEETTING[1]-[23]-[9810]-[0]***” substitutes 23rd servomotor value of VS-H8PWM28/V2 to 1st servomotor output. During the substitution, the original value x becomes “ $((x-128)*256)+9810$ ”. This equation result is applied to the final value of imported motion file.

4. Mode switch and audio settings

You can use the motion created in “3.Making robot motion” to control robots using controllers and to make “auto demonstration” that is a continual play of motions along with a sequence of scenario. You can set preferences of these controls from this software to translate it to the CPU board with several motion data. After the translation, you can disconnect the cable and turn on the CPU board to allow the CPU board running the firmware by itself. In this single mode, the CPU access to I/O connected to the mode switches behind on its circuit board to read the status. Based on switch status, CPU start motions assigned to each switch status. Since mode switch state can be selected from 10 types of 0-9, you can assign up to 10 motions.

You can set preferences, such as motion assignment of controller buttons and creating scenarios of an auto demonstration, from the dialog of “Mode switching and audio map preference and writing to CPU board” in this software. In this dialog, you can also set preferences of audio setting to be transferred to the CPU board.

To open the dialog, click “Project preferences”—“Mode switch/audio preferences and writing...” from menus. After clicking the following dialog is open.



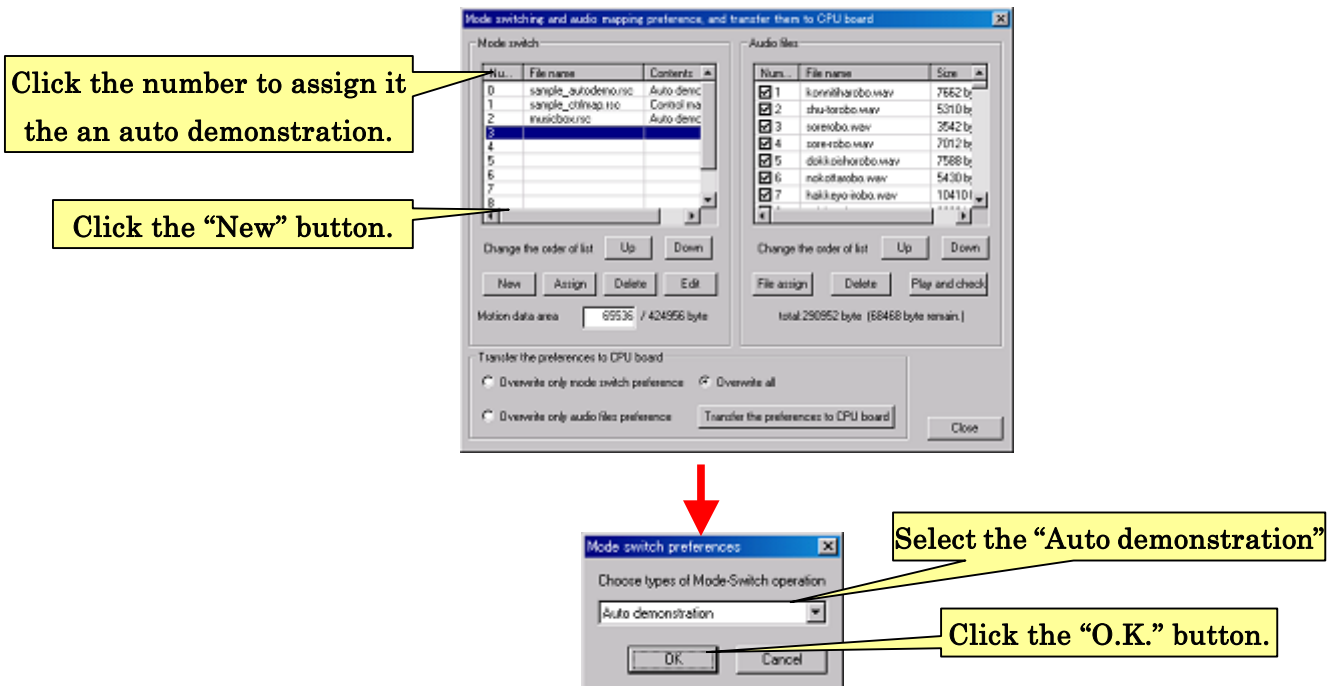
The left side of the dialog is for selecting motions to be assigned to mode switch status. The right side of the dialog is for audio setting to be transferred to the CPU board. After setting preferences, you can transfer it to the CPU board using the lower side of the dialog. You can set the size of the ROM region in which data file of motion, audio, and control map.

From next section, this manual details about this dialog.

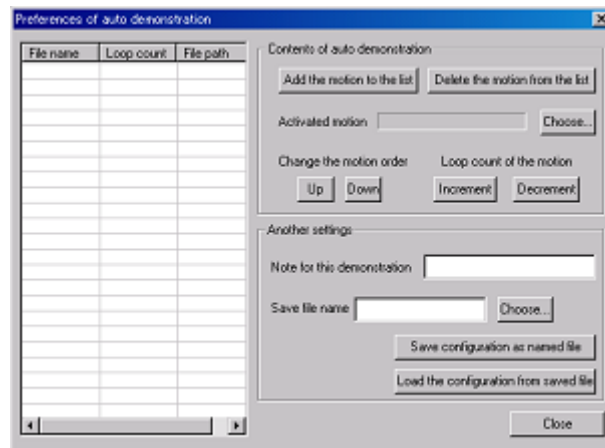
4-1. Auto demonstration

Auto demonstration is a function to play motions arranged on an order continuously and automatically. Each motion can repeat based on set loop counters respectively. To create an auto demonstration, use the “Preferences of auto demonstration” dialog opening from “Mode switching and audio map preference and writing to CPU board”.

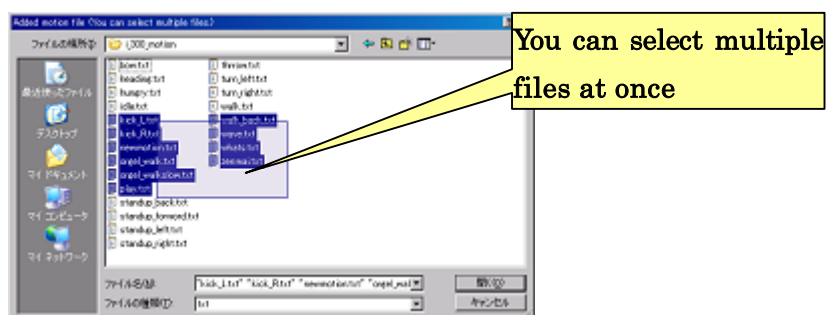
To create a new auto demonstration, select a mode switch number from the list, in the left side of the dialog, to be assigned to the auto demonstration. Then, click “New” button. After clicking, an opening dialog will ask you a type of added file. Select “Auto demonstration” and click “O.K.”



After clicking “O.K.”, the following dialog is open.



To add motion files to the auto demonstration, click “Add motions to list” from item of the “Auto demonstration contents”. After clicking, a dialog asks you to select a motion file. Choose a motion file and click “Open”. By selected several files, they can easily be added at the same time. (See below figure.)



Selected motion files will be added to the list in the left side of the dialog. The order of the list is the order to be executed in the auto demonstration. To change the order of motions, click a motion you want to change the order to select it, and click “Up” or “Down” from the item of “Change the motion order”. When the button is clicked, the item of the motion moves toward up or down and replaces its order.

To change repeating time, select the motion file from the list as well as changing order of the motion, and click “Increment” or “Decrement” from “Loop count of the motion”. When the button is clicked, the loop counter of selected motion increases/decreases.

To delete a motion file from the list, click the motion file and “Delete the motion from the list”.

Click a line to make it selected

Change the motion order by clicking the “Up” or the “Down” button.

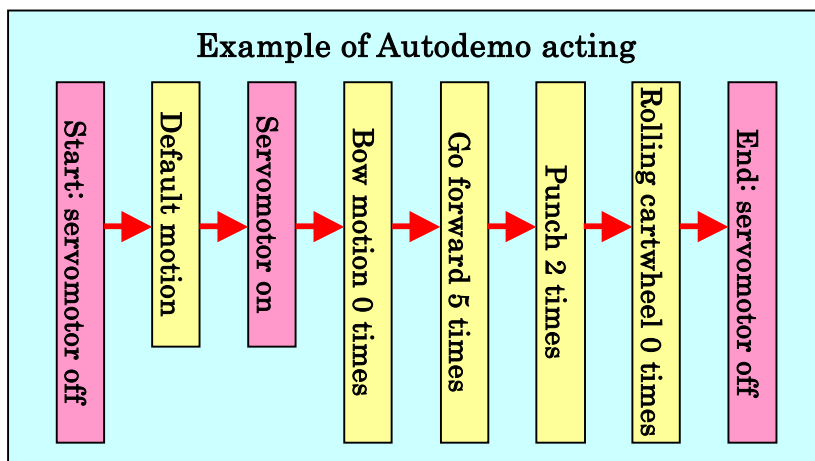
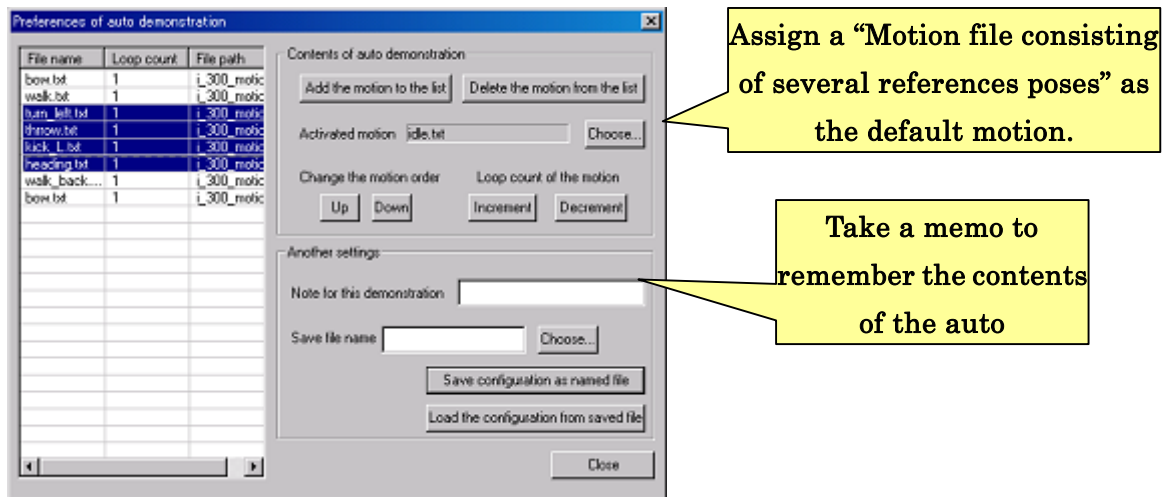
Deleting the selected item from the auto demonstration.

Changing the motion loop counter by clicking the “Increase” or “Decrease” button.

| File name | Loop count | File path |
|---------------|------------|-------------|
| bowl.tst | 1 | i_300_motic |
| walk.tst | 1 | i_300_motic |
| turn_left.tst | 1 | i_300_motic |
| throw.tst | 1 | i_300_motic |
| kick_L.tst | 1 | i_300_motic |
| heading.tst | 1 | i_300_motic |
| walk_back.tst | 1 | i_300_motic |
| bowl.tst | 1 | i_300_motic |

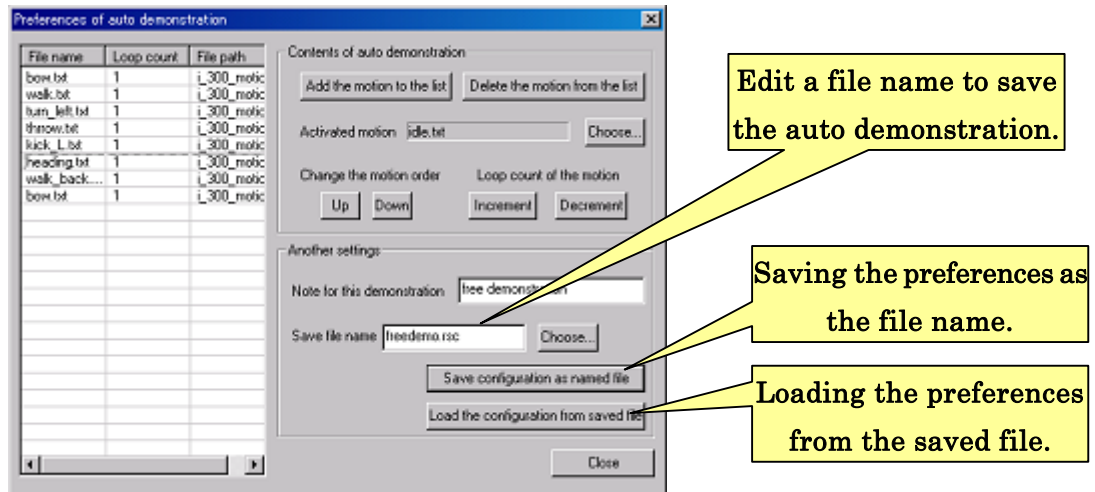
A “Default motion” including with “Preference of auto demonstration” is played before turning on the servomotor to start auto demonstration with safety poses. Immediate after waking up the CPU board, the robot pose is sometimes unstable because of unstable control signals. This sometimes causes danger situation such as a motor lock and a falling down, if servomotors are turned on with such unstable signals. Therefore, default motion is needed to start safe motion that is usually a “motion file consisting of several (4 or 5) reference poses” to start auto demonstration from the reference pose. **To run the auto demonstration safely, this default motion must be configured.**

You can note a content memo of auto demonstration at “Note for this demonstration”. Although this is optional and related nothing with auto demonstration, taking notes is convenience when you want to remember the contents of the auto demonstration.



To save the auto demonstration, click “Choose...” from the line of “Save file as”, and select destination and type the file name. Then, click “Save configuration as named file”.

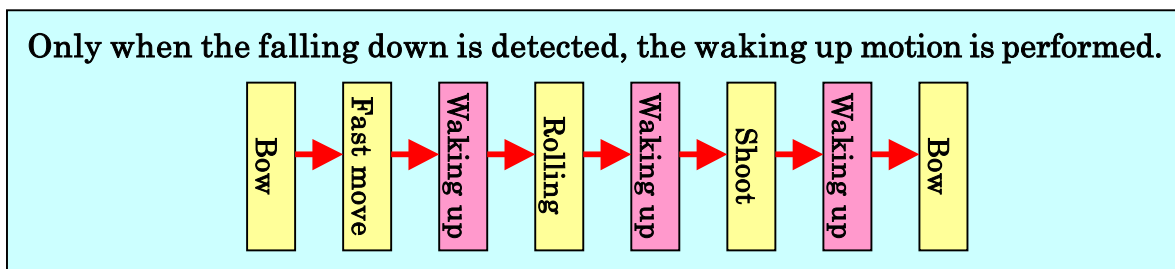
To edit saved auto demonstration, click “Load the configuration from saved file” and select saved auto demonstration file. After loading it, you can edit the preferences on same dialog.



To quit the dialog, click “Close”. The dialog will go back to the “Mode switching and audio map preference and writing to CPU board” dialog, and created auto demonstration is automatically assigned to the mode switch already selected from the list of “Mode switch preferences”.

4-1-1. Branching motions in auto demonstration

From this version of software, a function branching motions is available using branch blocks. For example, the robot can start a waking up motion when it falls down in the auto demonstration. See branch block examples detailed in “3-4-6. Conditional branch block”. By inserting a branch block that conditionally branches motion into a waking up motion in a falling down situation, the robot can wake up when it falls down and can proceed to the next motion when it stands up.



Another example is that; for example, a microphone is available for audio driven auto

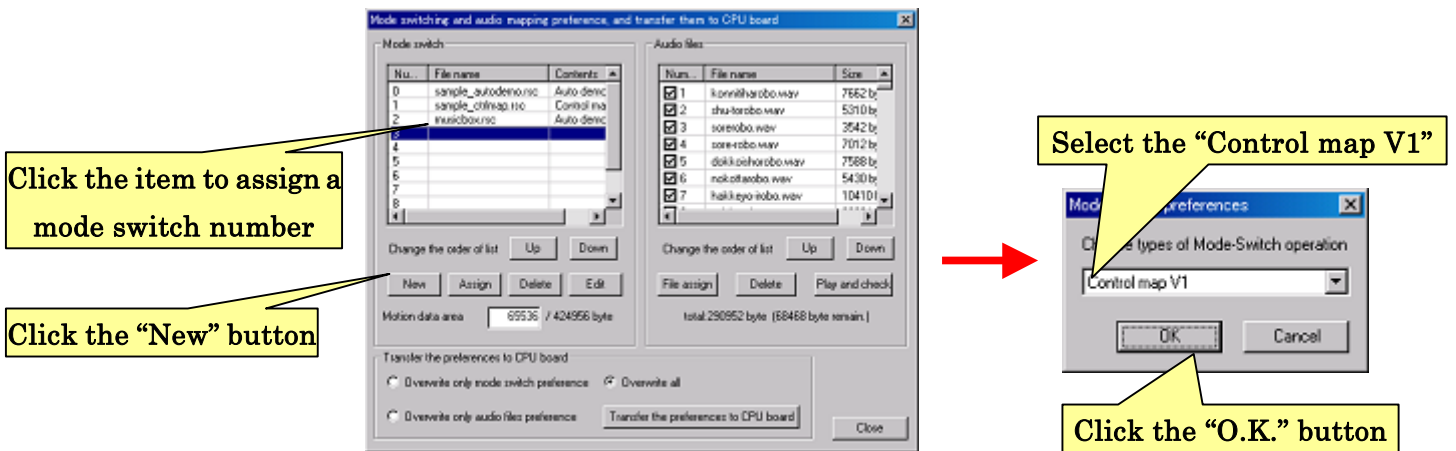
demonstration. A microphone can be connected into an expand analog-digital converter circuit to capture a loudness level of surrounding sound. Using this type of sensor, the auto demonstration that proceeds into next motions based on your voice by inserting a loop block repeating a motion during the sound level is less than a threshold.

4-2. Preferences of control map V1

An optional controller is available to control the robot. In this case, you need to assign motion files on each buttons of the controller. The assignment list of motions is called “Control map”.

In this software, the users can select a control map from “Control map V1” and “Control map V2”. “Control map V1” corresponds to the control map of the previous version of this software, RobovieMaker for VS-RC003. “Control map V2” is the latest control map consisting of same function of “Control map V1” and also new expanded features, such as analog input supported and a simultaneous button event supported. At first, this section reviews the “Control map V1”.

You can create a control map from the “Control map preferences” dialog to be opened from the “Mode switching and audio map preference, and writing to CPU board” dialog. On the dialog, to open the “Control map preferences” dialog, click a number of mode switch, that you want to assign a control map. Then, click “New”. After clicking, another dialog asks you a type of mode switch. Select “Control map V1” and click “O.K.”.



After clicking the following dialog will be opened.

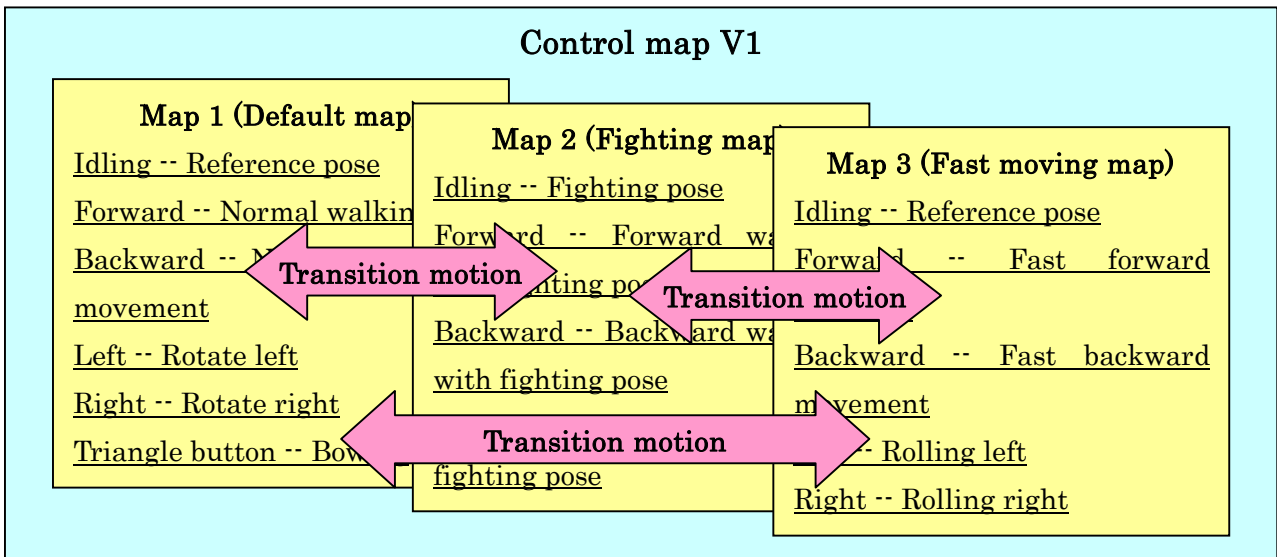


4-2-1. Outline of Control map V1

The left side of the dialog shows the list of available inputs to be able to configure in Control map V1. The inputs are defined as not only buttons of the controller but also as the following lists having specific meanings.

- **Idling** – This is a default motion performed when there is no input. **The idling motion must be defined to start the program when you use the controller.**
- **Transition motion into map [x]** – This is a transition motion performed when the control map switches. (“Map [x]“ is detailed below.)
- **Analog 1-4** – This is a sensor driven motion performed based on signals of the expanded sensor board devices. (The performed condition is detailed below.)
- **Disable** – Unsupported input in the latest Control map V1. No motion is performed even if you assign the motion to this input.

In the Control map V1, one motion file can be assigned to one input including the special input as detailed above. The motion assignment list on each input called “Map”. Control map V1 can contain several maps. This enables a button of the controller to perform multiple motions by changing the map in the control. (See the figure below.)

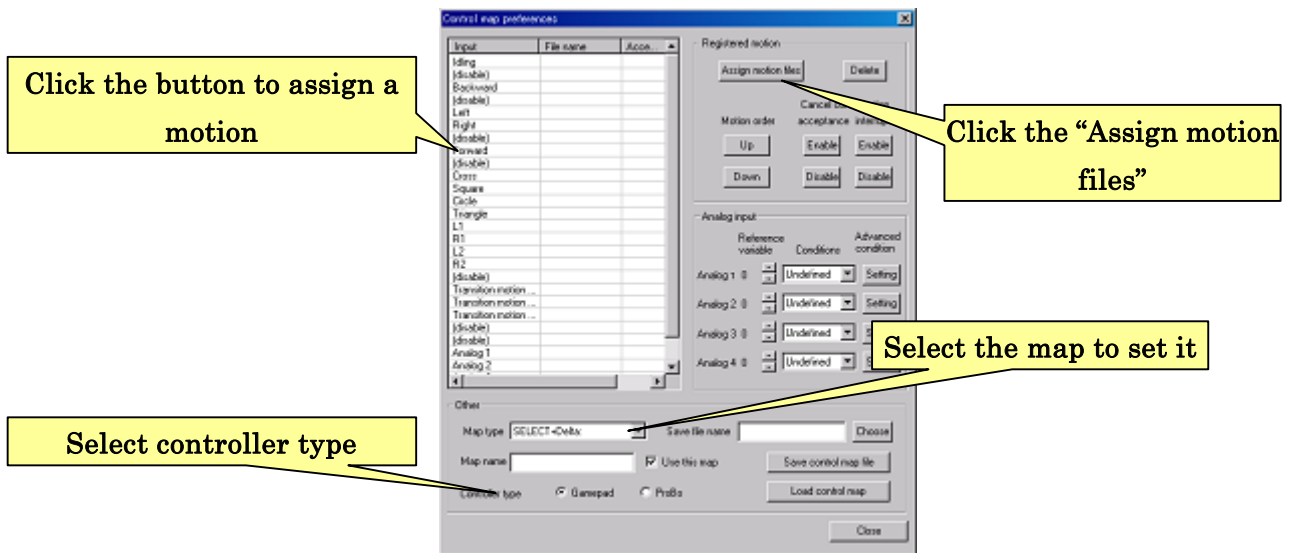


Structure of Control Map V1

You can assign maps to situations up to 4 maps. In control, the map is switched from the controller. When the map changed, the transition motion is performed.

4-2-2. Assigning a motion file to the input

To assign a motion file to a button of the controller, click the motion that you want to assign from the list in the left side. Then click “Assign motion files” from “Registered motion”. After clicking, an opening dialog asks you to choose a file assigned to the selected button. Select the motion file and click “Open”.



The motion file should be added to the list after the motion assignment. **As stated in Section “4-2-1.Outline of Control map V1”, an idling motion must be assigned. If you do not want to make a complex motion, a simple motion consisting of one reference pose is enough for the idling motion.** The total time length of the idling motion corresponds to a sampling time of the button input. Therefore, in case the simple motion, short transition time of the pose improves the response time. However, note that, shorter transition time causes a sudden robot movement when a performed motion finished and transited to the idling pose.

The input list is set to an appropriate one based on “Controller type” in lower side of the dialog. Select controller type you use.

To delete the motion from the list, click the motion from the list to be selected, and click “Delete”.

To change the order of the motion, click the motion from the list as well, and click “Up” or “Down” from “Motion order”. Note that the order of buttons is not changed.

In the low side of the dialog, select a type of the map you want to assign the motions at “Map name”.

4-2-3. Cancellation preferences

When the robot is performing the motion, in case you controlled it from the controller, the next order of the motion from the controller will be performed after the current motion finished. To perform the motion immediate when the button is pushed, enable the cancellation. Using this, the order of the motion cancels the current motion so that the order can be executed.

The cancellation preferences consist of two types of configurations that are “acceptance” and “interrupt”. A motion, set as an “acceptance” motion, is always cancelled and accepts another motion if the order is provided from the controller. The next motion is immediately performed. On the other hand, a motion, set as an “interrupt” motion, is always accepted and cancels the current motion to start the next motion.

To enable/disable the cancellations, click the motion you want to change the settings to select the motion. Then, click “Enable”/“Disable” from “Acceptance cancellation” or “Interrupt cancellation”.

Although enabling cancellation improves response of the controller, it also causes an immediate servomotor movement as well as the start of the motion and the stop of the motion, because the cancellation of the motion causes decoupling motion. Since this immediate movement of the servomotors consumes lots of energy, voltage supplied to the CPU board sometimes drops. Be careful not to immediately move the servomotor. In a worse case, the CPU board reboots because of the low voltage.

4-2-4. Analog input

Using the four channels of “Analog input” in Control map V1, you can make the motion automatically performed when a variable of the CPU board satisfies a condition. The condition can be set in “Analog input preferences”. (See “5-1.Variables of the CPU board” about details of available variables in the CPU board.)

The image shows a screenshot of the 'Analog input' dialog box. The dialog box has a title bar 'Analog input' and three columns: 'Reference variable', 'Conditions', and 'Advanced condition'. There are four rows, each representing an analog input channel (Analog 1 0, Analog 2 0, Analog 3 0, and Analog 4 0). Each row has a dropdown menu for the reference variable, a dropdown menu for the conditions (all set to 'Undefined'), and a 'Setting' button. Three yellow callout boxes with black text point to specific parts of the dialog box: one points to the reference variable dropdown for Analog 1 0, another points to the 'Setting' button for Analog 1 0, and a third points to the 'Conditions' dropdown for Analog 1 0.

Set a variable number to be used for comparison with threshold value.

Another dialog box appears to set advanced conditions.

Select a main condition to perform the motion.

An analog input has settings consist of a reference variable number set in “Reference variable” and a condition to start the motion set in “Condition type” and “Advanced conditions”.

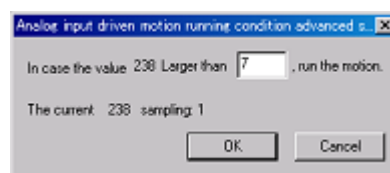
The “Condition type” is for a setting to start the motion if what kind of condition satisfied. A type of condition can be selected from the four conditions that are “More than threshold”, “Less than threshold”, “Bit check”, and “Undefined”. In the condition type of “More than threshold”/“Less than threshold”, the motion is performed if selected variable is more/less than a threshold value. For example, this condition type can be used to perform a waking up motion automatically based on the accelerometer.

In the condition type of “Bit check”, the motion is performed if chosen bits of selected variable correspond to a bit condition. For example, this condition type can be used to perform a motion driven by digital switch inputs.

The “Undefined” is for disabling the analog input.

“Advanced condition” is for setting threshold value and bit condition. After selecting the type of the condition, click “Setting” button of “Advanced condition”. Appropriate dialog is open to set the conditions.

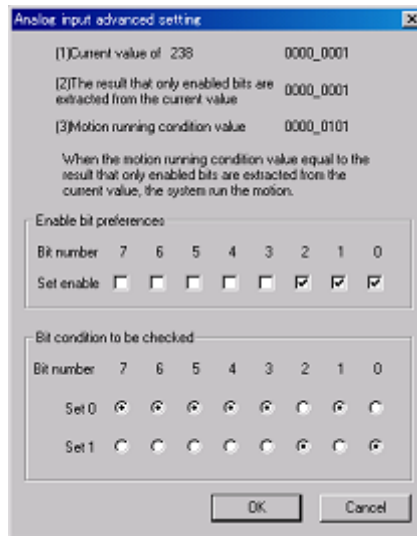
If you selected “More than threshold” or “Less than threshold”, the dialog below picture is opened. Threshold is a border value to compare to the variable. Type the threshold value at the dialog (between -32768 and 32767). Then click “O.K.” of the dialog to close the window.



The dialog below line shows the current variable of analog input. Use this monitoring value as reference.

If you selected “Bit check”, the dialog is opened as below figure. The bits compared with the variable is not 16 bits but 8 bits. Select referred bits from the “Enable bit preferences” and bit conditions. The result of AND operation of this conditions and enabled bits is compared with the result of AND operation of the variable and enabled bits.

In this condition, the motion starts if “Value of extracted variable with only enabled bits” corresponds to the “Motion running condition”. Click “O.K.” to close the dialog.



The current value of the variable is shown in the dialog as well. Use this monitoring value as reference.

4-2-5. Save/Load Control map V1

To save the Control map V1 to a file, after finishing the setup of Control map V1, click “Chose” button next to “Save as” to select the location and to edit saved file name and click “Save control map file”. To load and edit the Control map V1 from a saved file, click “Load control map from saved file” and choose a saved file of Control map V1.

Click “Close” to finish editing the control map and to close the dialog.

4-2-6. How to use controller in Control map V1

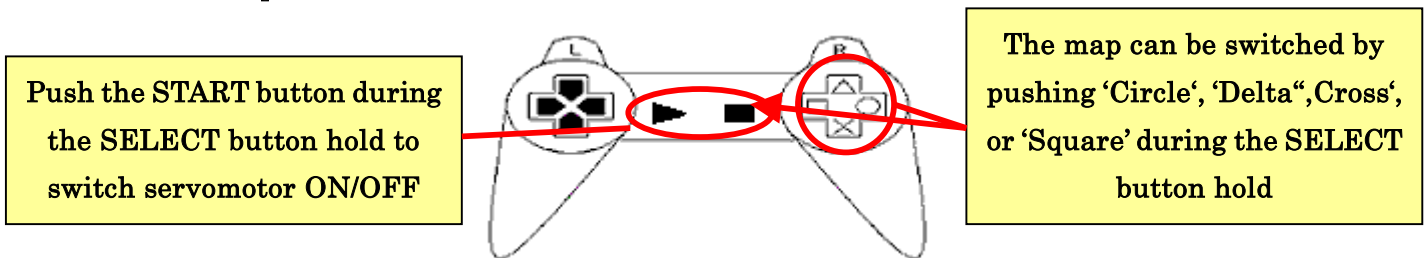
This section details how to control the robot using Control map V1. Basic controls of the controller are that, when you push the button in the controller, an assigned motion is performed. The motion, having loop structure, repeats the motion during button kept down, and finishes the motion when the button is released.

The motions, starting with the event of four channels of “Analog inputs”, are performed during only the robot motion idling if the set conditions is satisfied.

In the Control map V1, you can use not only the basic control but also two specific controls. After running the CPU board as a controller mode, you need to turn on the servomotors by operation of “Servomotor ON/OFF”. This operation is accepted even the motions are performed. Turn off the servomotors immediately when the robot is in accident such as it falls down or causes motor lock.

The input method of the controller is as follows.

Game pad case



The Game pad with vibration function vibrates when the voltage of the CPU board is less than an alarming threshold voltage.

ProBo case



* Be careful to connect the controller with appropriate direction. If you connect the

receiver with wrong direction, the CPU board will not run properly. In worse case, if you leave the wrong connection for a long time, the CPU board could malfunction.

4-3. Preferences of Control map V2

This section details the Control map V2.

4-3-1. Outline of Control map V2

In this software, the latest version of control map, “Control map V2” having new additional feature of control from Control map V1, is available.

Control map V2 have expanded analog input that is as well as the analog input of Control map V1. Motion starting condition can be set in all of button event. Thus, new features of button event are supported such as multiple button input on the Game pad. More features of motion starting based on the events from expanded boards. As for motion start conditions, new additionally conditions are supported. The conditions are “More than threshold”, “Less than threshold”, and “Bit check” that are supported in Control map V1, but also “Equal to threshold”, “Not equal to threshold”, “Always true”, and “Always false”.

These new features of Control map V2 enable the robot to response to several events of multiple inputs simultaneously happened. An autonomic control of the robot, that is rather hard to be accomplished, is possible using the control map. The multiple button events satisfy a request that users desire more number of input.

To open the dialog of “Control map V2”, click mode switch number, which you want to assign the map, from the list in the left side of the dialog, and click “New” button. After clicking, opening dialog asks a map type. Select “Control map V2” and click “O.K.”. The dialog of “Control map V2 preferences” is opened.

Click the mode switch number to assign the motion.

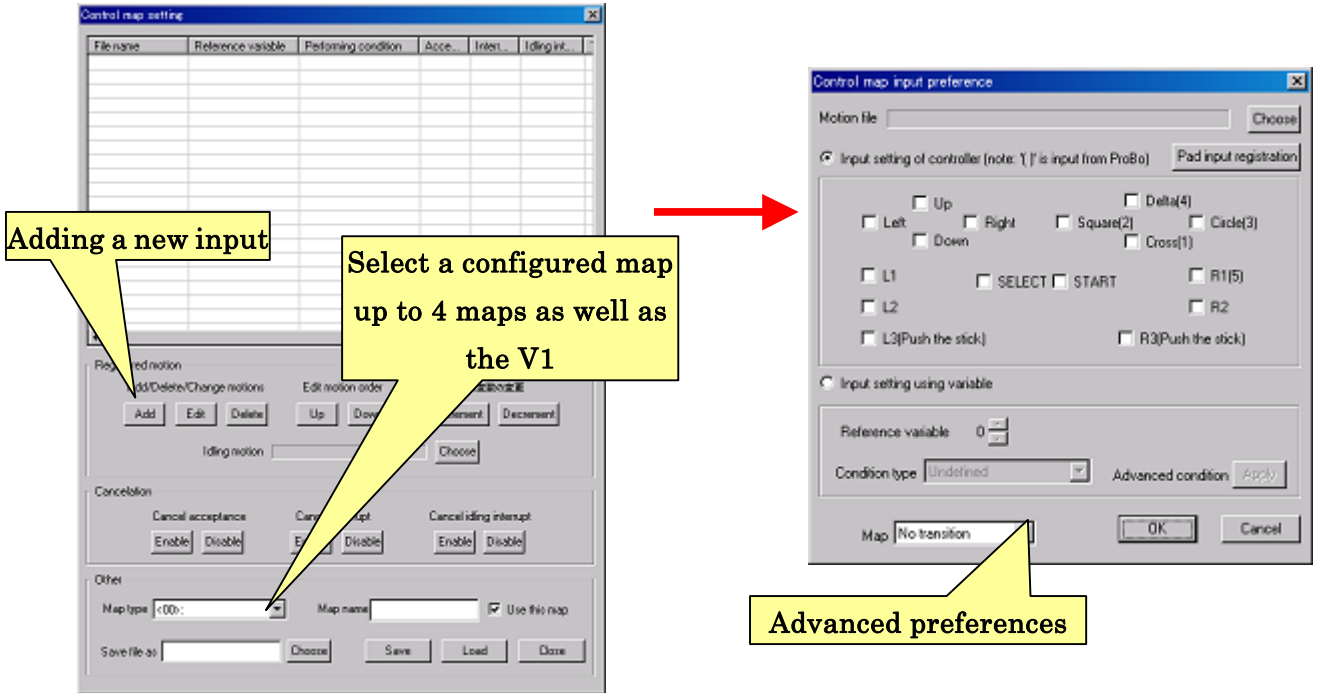
Click the “New” button

Select the “Control map V2”

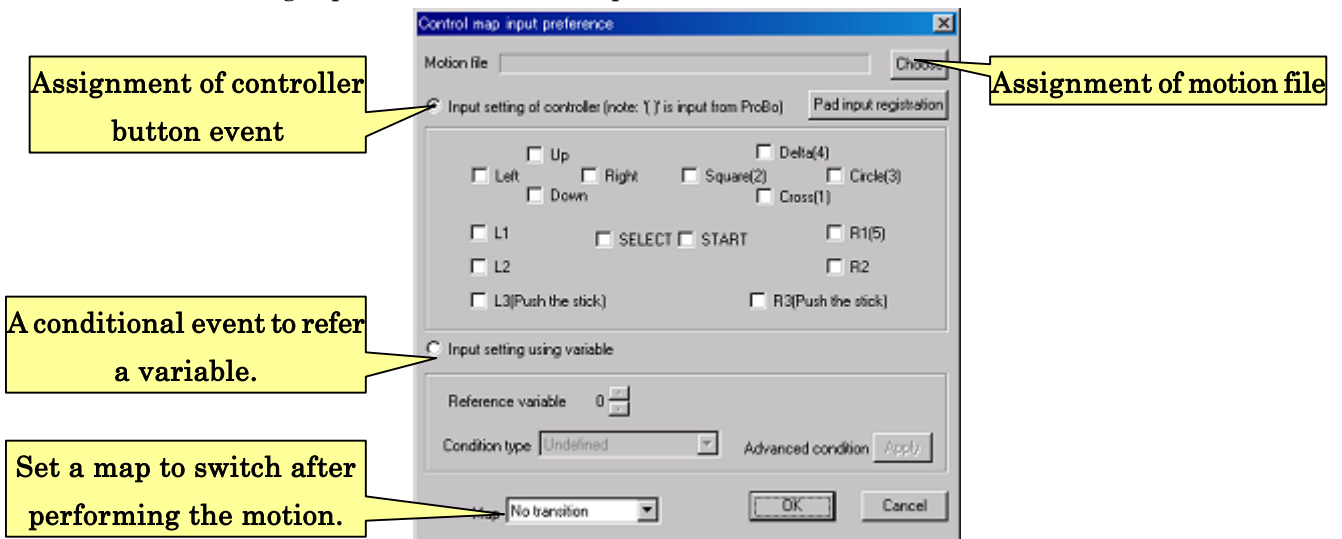
Click the “O.K.”

4-3-2. Assigning motion files to Control map V2

To assign a motion file to the Control map V2, click “Add” button from “Add/Delete/Change motion” in the registered motion. After clicking, the dialog of “Control map input preference” is opened. In this dialog, you can set a button event to a motion file.

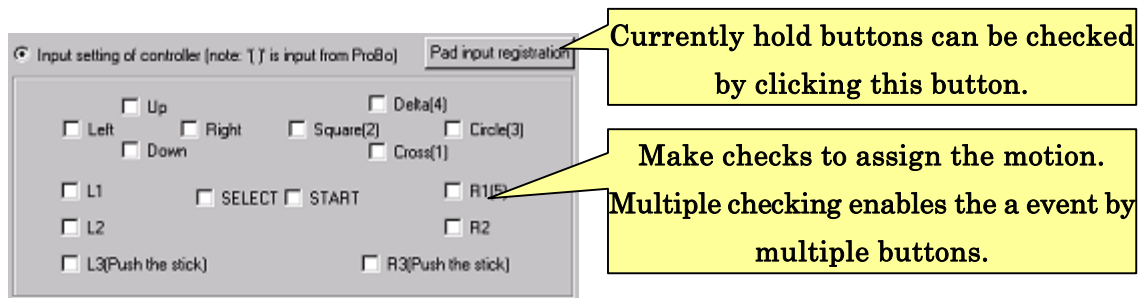


In the “Control map input preference” dialog, you can use two types of motion assigning method. One of them is a method to assign an event of button of controller to a motion file. The other is to assign an event, when a condition is satisfied, to a motion file as well as analog input of the Control map V1.



4-3-2-1. Motion assignment to Game pad input

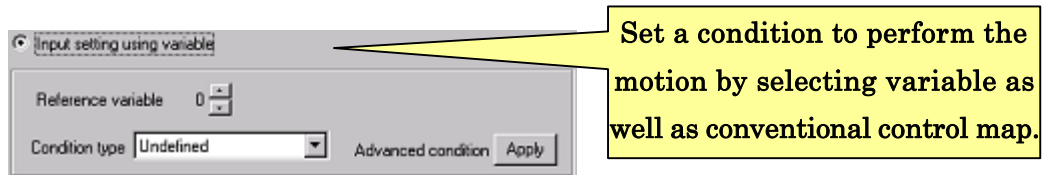
To assign a motion file to a button of the controller, check “Input setting of controller” and the buttons you want to assign the motion. By checking multiple buttons, the motion file is performed in the event that the buttons are pushed down at the same time.



Optionally, if your PC is communicating with the CPU board, click “Pad input registration” button with the buttons being pushed down. The hold buttons are checked in the dialog.

4-3-2-2. Motion assignment to a variable

To start a motion based on a variable condition, click “Input setting using variable”. Select a reference variable in “Reference variable”, and choose a type of condition from “Condition type”. To set an advanced condition, click “Setting” of “Advanced condition”.



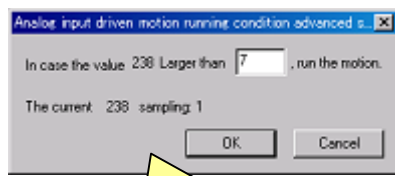
Condition types you can choose in “Condition type” are as follows.

- **Undefined** – No motion is performed with this undefined condition. You do not need to set an advanced condition.
- **Larger than threshold** – The assigned motion is performed when the reference variable is larger than the threshold value. You need to set the threshold value compared to the variable in the advanced setting.
- **Smaller than threshold** – The assigned motion is performed when the reference variable is smaller than the threshold value. You need to set the threshold value compared to the variable in the advanced setting.
- **Bit check** --- The assigned motion is performed when the enabled bits of reference variable correspond to set value. You need to set enabled bits and bit

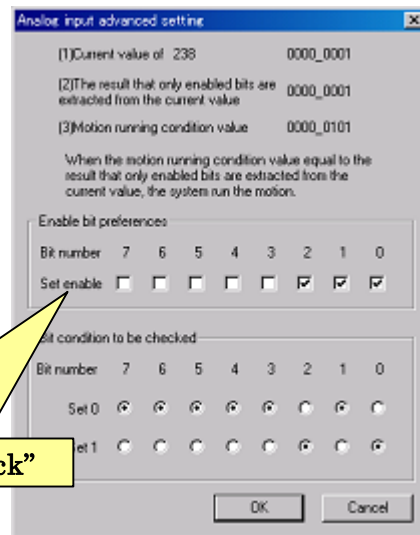
conditions compared to the reference variable.

- **Equal to threshold** – The assigned motion is performed when the reference variable is equal to the threshold value. You need to set the threshold value compared to the variable in the advanced setting.
- **Not equal to threshold** – The assigned motion is performed when the reference variable is NOT equal to the threshold value. You need to set the threshold value compared to the variable in the advanced setting.
- **Always true** – The condition is always satisfied. The assigned motion is always performed. You do not need to set the advanced condition.
- **Always false** – The condition is always not satisfied. The assigned motion is disable and is not performed. You do not need to set the advanced condition.

The opening dialog box types, when you click “Setting” button of “Advanced condition”, are as below two figures.



Set an advanced condition of “Larger than threshold”, “Smaller than threshold”, “Equal to threshold”, or “NOT equal to threshold”.



Advanced condition of “bit check”

4-3-2-3. Map transition motion

In Control map V1, there was a specific setting for a transition motion performed when the map is switched. In Control map V2, the motion performed by the button event is assigned to a map transition. Select a control map in “Map transition” to be switched after performing the motion.

Note that, in Control map V1, there is no default button event to switch a control map, such as button events of “SELECT+’Delta’,’Circle’,’Cross’,’Square” defined in Control map V1. You must defined the map transition to switch the map.



Define the motion to switch the map at the map transition.

4-3-3. Operation in dialog of Control map V2

Motion files added to the Control map V2 are listed in the top of the dialog. Listed information are as follows; motion file name, reference variable, performed condition, cancellation configurations, and transition map.

The screenshot shows the 'Control map setting' dialog box. It contains a table of registered motions, a section for managing the list, cancellation preferences, and other settings. Callouts point to specific parts of the dialog:

- The list of the registered motion in the control map. When the control map is used, the satisfied condition is searched from the top of the list. (It means that upper motions have higher priorities.)** - Points to the table of registered motions.
- Registering the motion, and changing the order in the list.** - Points to the 'Registered motion' section with buttons like 'Add', 'Edit', 'Delete', 'Up', 'Down', 'Increment', and 'Decrement'.
- Cancellation preferences.** - Points to the 'Cancellation' section with 'Enable' and 'Disable' buttons for 'Cancel acceptance', 'Cancel interrupt', and 'Cancel idling interrupt'.
- Saving and loading Control map V2 file.** - Points to the 'Other' section with 'Save file as', 'Save', 'Load', and 'Close' buttons.

| File name | Reference variable | Performing condition | Acc. | Inter... | Idling int. |
|----------------|--------------------|----------------------|---------|----------|-------------|
| 後退(19軸)制御-A | 241(Control) | Backwardinput | Disable | Disable | Disable |
| 左移動(19軸)制御... | 241(Control) | Leftinput | Disable | Disable | Disable |
| 右移動(19軸)制御... | 241(Control) | Rightinput | Disable | Disable | Disable |
| 前進(19軸)制御-A | 241(Control) | Forwardinput | Disable | Disable | Disable |
| 直交移動制御 | 241(Control) | Crossinput | Disable | Disable | Enable |
| 直交移動制御 | 241(Control) | Squareinput | Disable | Disable | Enable |
| 直交移動制御 | 241(Control) | Circleinput | Disable | Disable | Enable |
| 円周移動制御(19軸... | 241(Control) | Triangleinput | Disable | Disable | Enable |
| 左方向移動(19軸... | 241(Control) | Linput | Disable | Disable | Disable |
| 右方向移動(19軸... | 241(Control) | Rinput | Disable | Disable | Disable |
| 左側移動制御(19軸... | 241(Control) | L2input | Disable | Disable | Disable |
| 右側移動制御(19軸... | 241(Control) | R2input | Disable | Disable | Disable |
| 前後移動制御上移動(1... | 241(Control) | L3input | Enable | Disable | Enable |
| 前後移動制御下移動(1... | 241(Control) | R3input | Enable | Disable | Enable |
| 前進(19軸)制御-A | 241(Control) | ForwardLeftinput | Disable | Disable | Disable |
| 前進(19軸)制御-A | 241(Control) | ForwardRightinput | Disable | Disable | Disable |
| 後退(19軸)制御-A | 241(Control) | BackwardLeftinput | Disable | Disable | Disable |
| 後退(19軸)制御-A | 241(Control) | BackwardRightinput | Disable | Disable | Disable |
| 直交移動制御-A | 241(Control) | RightBackwardinput | Disable | Disable | Disable |
| 直交移動制御-B | 241(Control) | CircleSELECTinput | Disable | Disable | Enable |

In case the CPU board is running the Control map V2 by itself, it searches a motion satisfying a set condition from the top of the list to perform the motion. If no condition is satisfied, it performs the idling motion. Thus, the upper lines of the list have higher priorities.

To change the motion order of the list, click the motion you want to change the order, and click “Up” or “Down” button from “Edit motion order” in “Registered motion”. Since a multiple motion selection is supported, you can select several lines to change their priorities simultaneously.

The number of reference variable can be easily changed by clicking the “Increment” or “Decrement” button of “Change reference variable”.

4-3-3-1. Idling motion preference

In Control map V2, you have to define an idling motion on each map as well as Control map V1, because the idling motion is needed for the controller operation to be started. To set an idling motion, click “Choose” button of “Idling motion” in “Registered motion”. After clicking, opening dialog ask you to select a motion file. Select an appropriate motion, such as a standing up motion consisting of one pose, and click “Open”.

4-3-3-2. Cancellation

In Control map V2, the cancellation of the motion is supported as well as Control map V1. Using cancellation, the next motion can interrupt the performing motion instead of waiting the end of it. Using the cancellation, the operation responses of control are improved.

Cancellations can be set as “Enable”/“Disable” on each motion registered in Control map V2. The cancellation of “Acceptance” and “Interrupt” are supported as well as Control map V1. In additionally, “Idling interrupt” is also supported. Thus, there are three types of cancellations.

“Acceptance” – If this cancellation is enabled, the motion accepts another motions of button events regardless of cancellation setting of the next motion.

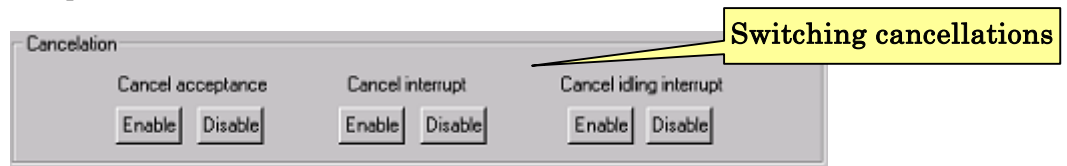
“Interrupt” – If this cancellation is enabled, the motion interrupts the another performing motion regardless of cancellation setting of the current motion.

“Idling interrupt” – If this cancellation is enabled, the motion interrupts only the idling motion. This type of the cancellation is used when you want to limit the situations for the motion to be performed, such as a waking up motion started based on accelerometers.

Although enabling cancellation improves response of the controller, it also causes an immediate servomotor movement as well as the start of the motion and the stop of the motion, because the cancellation of the motion causes decoupling motion. Since this immediate movement of the servomotors consumes lots of energy, voltage supplied to the CPU board sometimes drops. Be careful not to immediately move the servomotor. In a worse case, the CPU board reboots because of the low voltage.

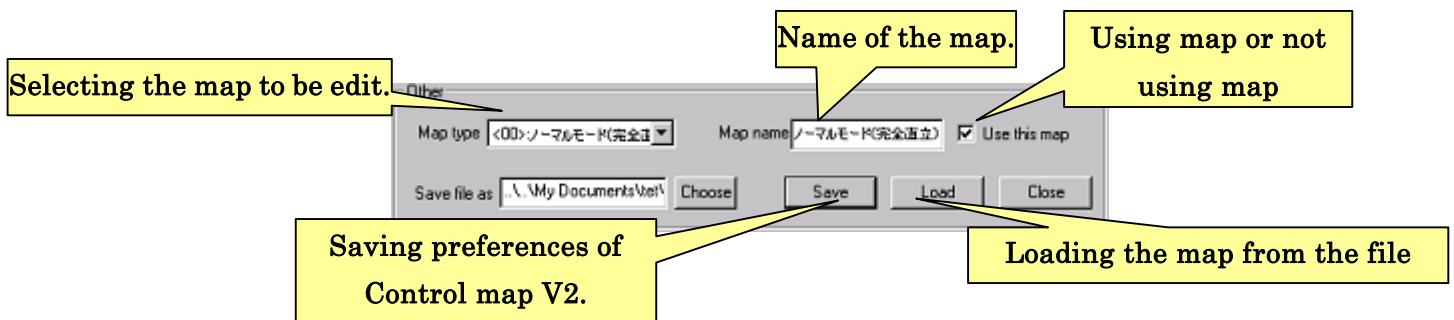
To change the setting of the cancellations, select the motion you want to change the

cancellation setting, and click the “Enable”/“Disable” button of “Acceptance”, “Interrupt”, and “Idling interrupt” from the “Cancellation”.



4-3-4. Save/Load Control map V2

To save the Control map V2 to a file, click the “Choose” button of “Save file as” in “Other”, and click the “Save” button to save it. To load the Control map V2 from a saved file, click “Open” and select a saved file of Control map V2.



In the dialog of “Other” settings, you can select the map, change the map name, and switch the map as enabled or disabled.

4-3-5. How to use controller in Control map V2

To control the robot using a controller, turn on/off the servomotors by pushing down SELECT and START button of Game pad simultaneously, or by switching SW4 of ProBo as well as Control map V1.

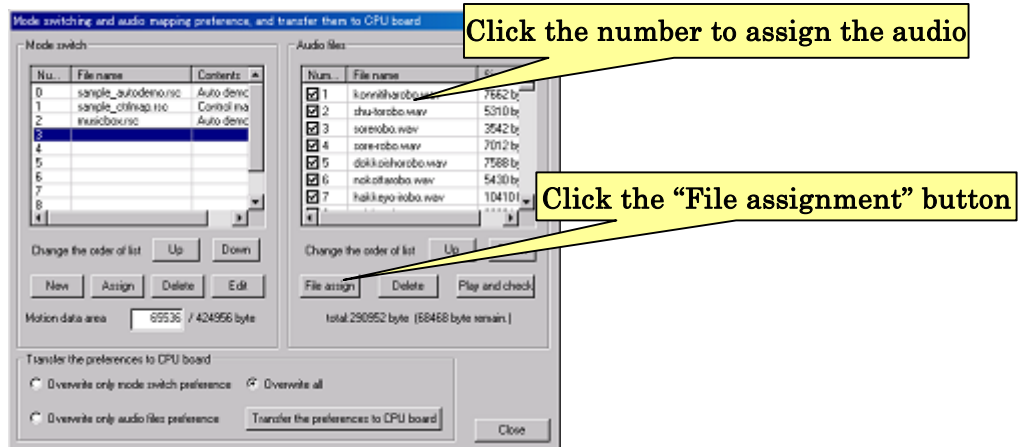
Note, however, that the Control map V2 has no default operation to switch the map, such as operations of “pushing down SELECT+’Delta’,’Circle’,’Cross’,’Square’ button simultaneously” or switching SW1 of ProBo. To switch the control map, a motion need to be registered as a transition motion. Using a normal button event from the controller, the control map is able to be switched.

To switch the map using the Control map V2, several motions need to be registered in “Map transition”.

4-4. Audio file setting

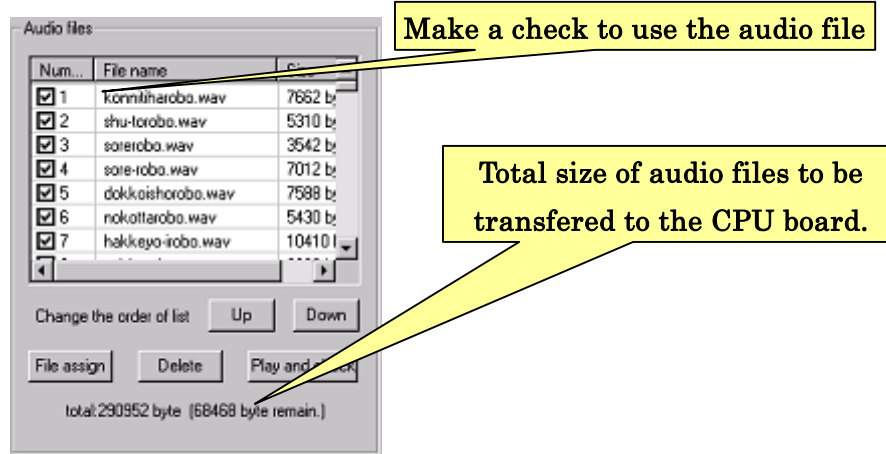
This software can transfer the audio data to the CPU board so that the CPU board output the audit from the speaker. (Note that you need to connect an optional speaker to the CPU board to output the audio.) The translated audio is allocated as a unique number. The number can be selected by the #30 slider bar variable for the audio to be assigned to a pose.

Only two types of audio wave sound format, “PCM” and “Microsoft ADPCM”, are supported up to 352kbyte (360448byte) because of the memory limitation. Since “PCM” format has good sound quality but is usually large size, a few data can transfer to the CPU board. If you want to use various audio files, convert the audio file to “Microsoft ADPCM” format. This format is not so good sound quality but is usually smaller size.



To select the audio file to be transferred, select the number of audio file from the list of “Audio files”. Then click “File assignment” button, and select the file assigning the number to it.

Selected audio file is added to the list. The check boxes in the columns of the number list are for enabling or disabling transfer of the audio file to the CPU board. Thus, checked files are indeed transferred to the CPU board. The total size of the audio file is shown in lower side of the dialog. **The audio file can be transferred up to 352kbyte (360448byte)**. Check out the audio files not to exceed this maximum size.



To delete the audio file from the list, click the line of the audio file to be selected, and click the "Delete" button.

To change the order of the audio file list, click the line of the audio file to be selected, and click the "Up" or "Down" button from "Change the order of list". Note, however, that, since the coupling of the audio files and the motion files are not synchronized, this operation can cause the inconsistency of the file mapping.

To listen the audio file from the speaker of your PC, click the "Play and check" button with the audio file selected.

4-5. Setting the ROM region for audio and motion data.

All the audio and motion data are written into the ROM region of the CPU board. The available size of the ROM region is 415 Kbyte. The data size is appropriately set by default. This software aborts the transferring the data due to the over limitation. You will see the warning message of the abort. If another ROM region is physically available, you can transfer both of motion data and audio data by changing the size of the ROM region to fit both data.

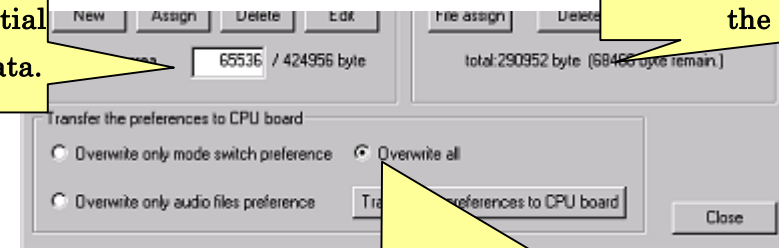
You may want to use more sizes for audio data because audio data is usually larger than motion data. In such cases, assign much more ROM region for audio data to transfer the larger audio data.

To change the assignment of ROM region, change the size of the value set in the “Motion data region” item of the “Mode switch” preference in the dialog. This value indicate the assigned size for keeping initial setting data and motion data in the ROM region.

After editing this value, and transfer the data into the CPU board with new size assignment. According as this change, the shown size of the audio changes.

Assigned size of the motion data which is used for keeping initial configuration and motion data.

Assigned size of the audio data in the ROM region.



When you change the setting of the ROM region, the “Overwrite all” is automatically

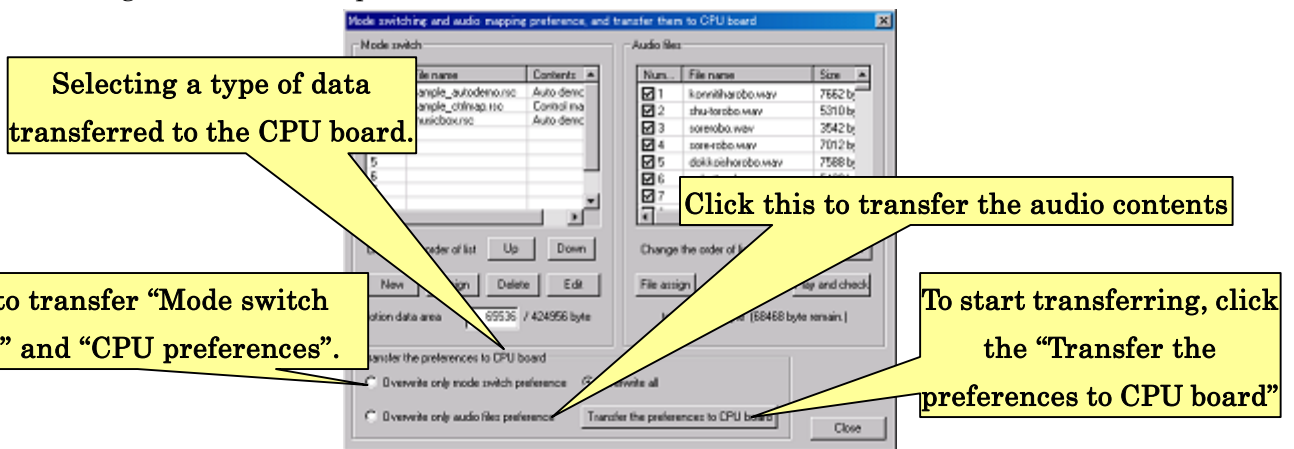
When you change the value of “Motion data region”, overwrite both of motion data and audio data. When you enter the size that is not dividable with “32768”, you only can select the “Overwrite all” in the “Transfer the preferences to CPU board” due to limitation of transferring program.

You can set the value experimentally, since the software aborts the transfer and shows a warring message even if you enter the wrong value. This message helps you to know the size of the limitation of the ROM region. Check it to set an appropriate size.

Note that there are also minimum size limitations both in motion data size and audio data size because the initial configuration at least have to be kept in that regions.

4-6. Transfer settings of mode switch, control map, and audio files

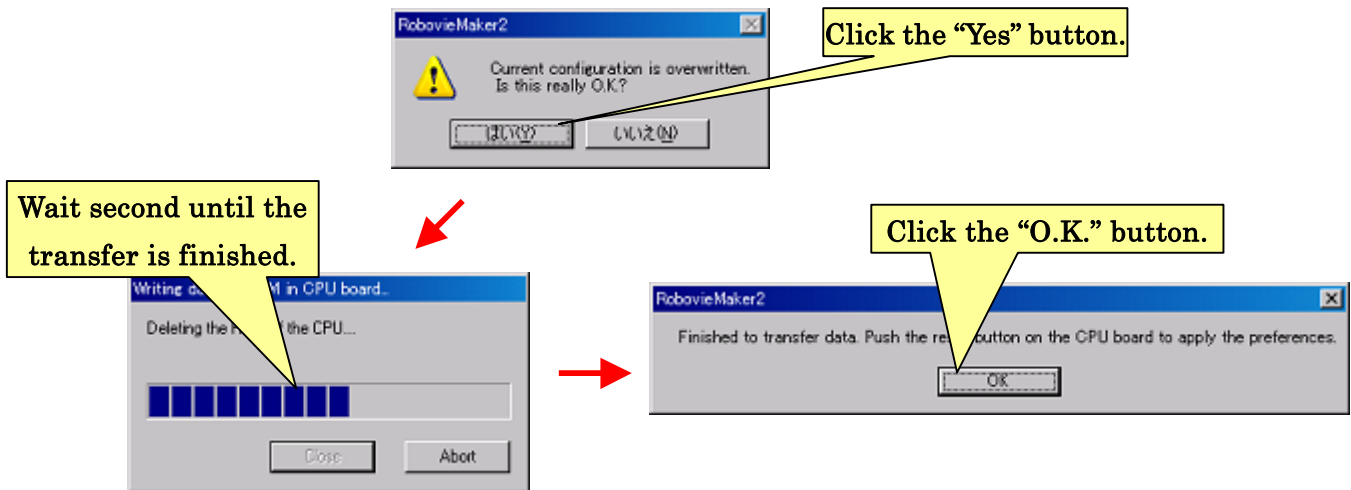
This section details the proceedings to transfer the settings of control map, auto demonstration, and audio files to the CPU board. These operations are conducted from the dialog of “Transfer the preferences to CPU board” in the lower side.



Before transferring the preferences, you need to select the transferred data types. Check “Overwrite only mode switch preferences” if you want to the preferences related with CPU configurations, such as control map, auto demonstration, and calibration result of servomotors. To transfer only the audio preference, check “Overwrite only audio files preference”. To transfer both of the settings, click “Overwrite all”.

Because the audio file is usually larger size than the control map and auto demonstration, it may take long time to be finished. If you don't change the audio files, uncheck the “Overwrite only mode switch preference” to save your time.

To start transferring the preferences to the CPU board, confirm the wire connection between your PC and the CPU board, and click the “Transfer the preference” button. During transferring the data, the below figures are shown. Follow the instruction in the figures to proceed the translation. **Do not disconnect the CPU board during transferring the data, nor push the reset switch.**



After all data transferred, the CPU board reboots automatically to apply the new preferences. (The CPU board may not reboot appropriately due to some environment in rare cases. Please reboot the CPU board to push the reset switch manually.)

5. CPU board configuration

This software can configure advanced settings of the CPU board. To change the preferences, click “Project preferences”—“CPU preferences” from the menu. The following dialog box appears after clicking.



A dialog box to configure advanced settings.

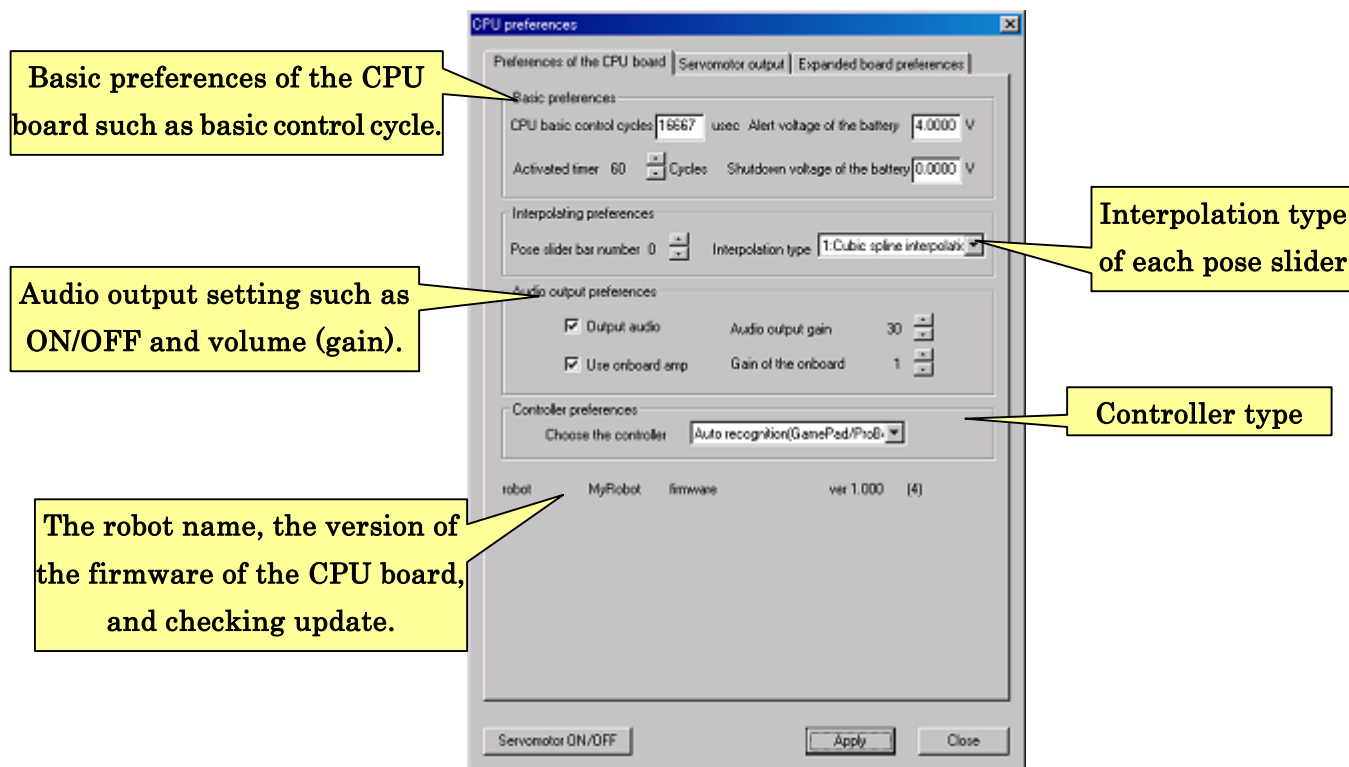
5-1. Variables in CPU board

The CPU board equips 256 of 16-bit variables in the RAM region. A lot of information is wrote and read using the RAM region. In this software’s preferences, you often need to select a variable to be accessed by selecting the number of the variable so that the CPU board can control the servomotors and expanded board using sensor feedback control.

Note that sharing variable between different objectives sometimes causes unexpected behavior of the CPU board. The additional volume of the manual, “VS-RC003VariableList.pdf”, details about the variable assignments. Refer the manual not to cause the interfering of the variable when you select the variable to be assigned.

5-2. Basic preference, interpolation, and controller setting

In the “CPU preference” tab dialog, the preferences of the control frequency, interpolation type, and audio output are available.



In the “Basic preferences”, the following preferences can be set:

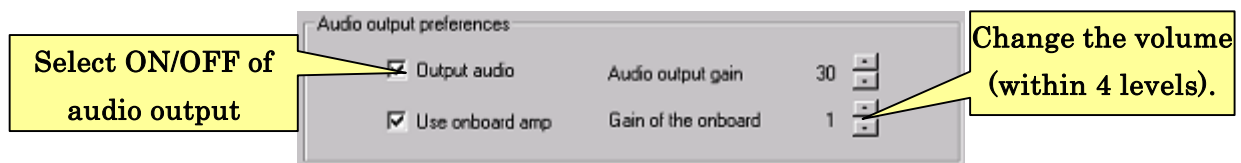
- **Basic control cycle of CPU board** – This is a basic cycle for the CPU to control the devices. By the default, the frequency is set to 1.6667 about equal 60Hz. The unit of “Cycle” appeared in the dialog corresponds to a unit of basic cycle time.
- **Boot timer** – This is the waiting time to start the single mode after the CPU board boots. During the waiting time, the CPU board quit to start the single mode. If the value set to 0, the CPU board never goes into the single mode.
- **Battery alarm voltage** – This voltage is a threshold to alarm low voltage supplied to the CPU board.
- **Battery shutdown voltage** – This voltage is a threshold to reboot the CPU board immediately when supplied voltage is lower than this threshold.

In “Interpolation preference”, you can set an interpolation type that is a method of constructing new control position between motion steps as the discrete poses. The CPU interpolates the poses using this method that you can set different types for each pose

sliders respectively.

To change the interpolation type, select a number of pose slider and a type of interpolation. You can choose an interpolation type from the following 5 types.

- **No interpolation** – No interpolation is performed. In this type, the CPU board doesn't change the value.
- **Cubic spline interpolation** -- This is a spline interpolation based on cubic polynomials between the data points. Servomotor control discrete signal are usually interpolated by this method.
- **Linear spline interpolation** -- This is a simplest form of spline using linear interpolation.
- **Switching before transition** -- Current value is switched to the next when the interpolation starts. This method is usually used for audio output.
- **Switching after transition** -- Current value is switched when a scheduled time passed.



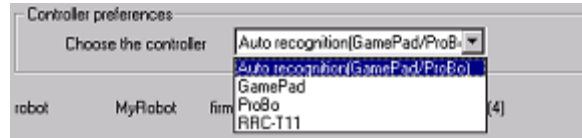
In “Audio output preferences”, you can set an audio output type from the CPU board.

- **Use audio output** -- Check this box for the CPU board to output the audio signal.
- **Audio output slider number** -- Select the pose slider number used for selecting audio output.
- **Use onboard amp**-- Check this box for the CPU board to use onboard amp.
- **Audio output gain** -- Select gain value in built-in amplifier between 0 to 3.

In “Controller preferences”, you can set an controller type connected to the CPU board.

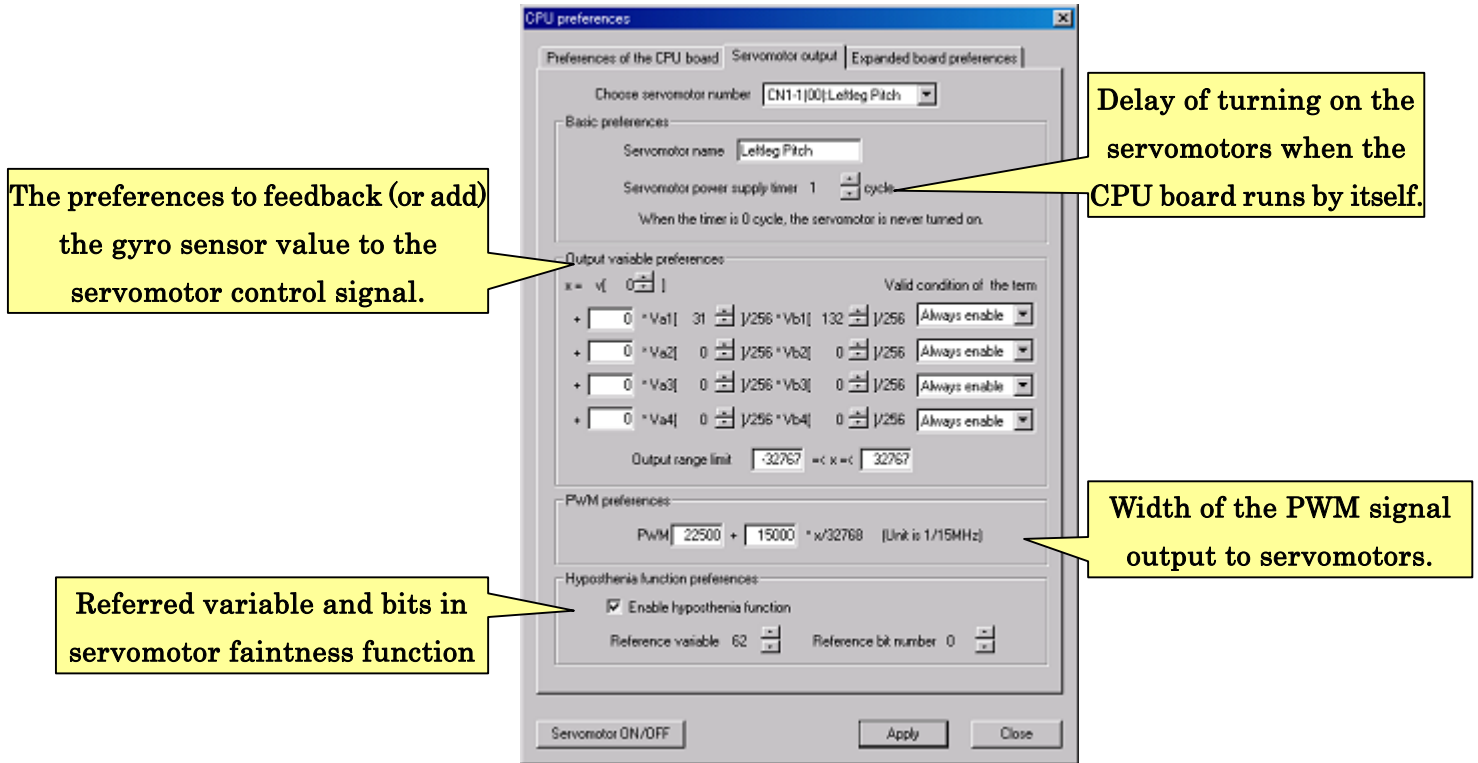
- **Auto select (GamePad/ProBo supported)** -- In this mode, a connected controller is automatically recognized if the controller is GamePad or ProBo. (RRC-T11 is not recognized in this mode.)
- **Gamepad** -- Used controller is limited in a GamePad
- **ProBo** -- Used controller is limited in a ProBo
- **RRC-T11** -- Used controller is limited in a “RRC-T11”

You can set this controller type as auto select mode except for using RRC-T11 manufactured by Futaba co., ltd. Select RRC-T11 in that case.

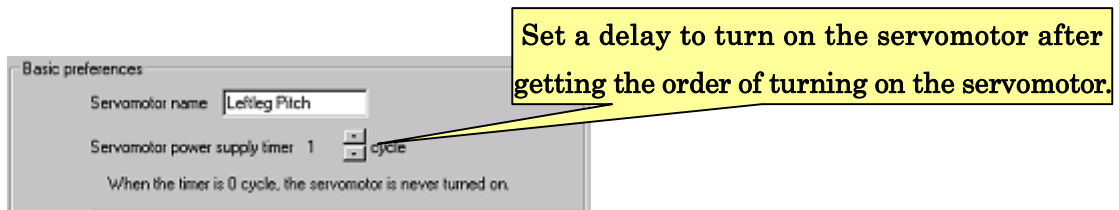


5-3. Servomotor preferences

In the “Servomotor control” tab, you can set preferences of 30-ch servomotor output respectively that are control signals.



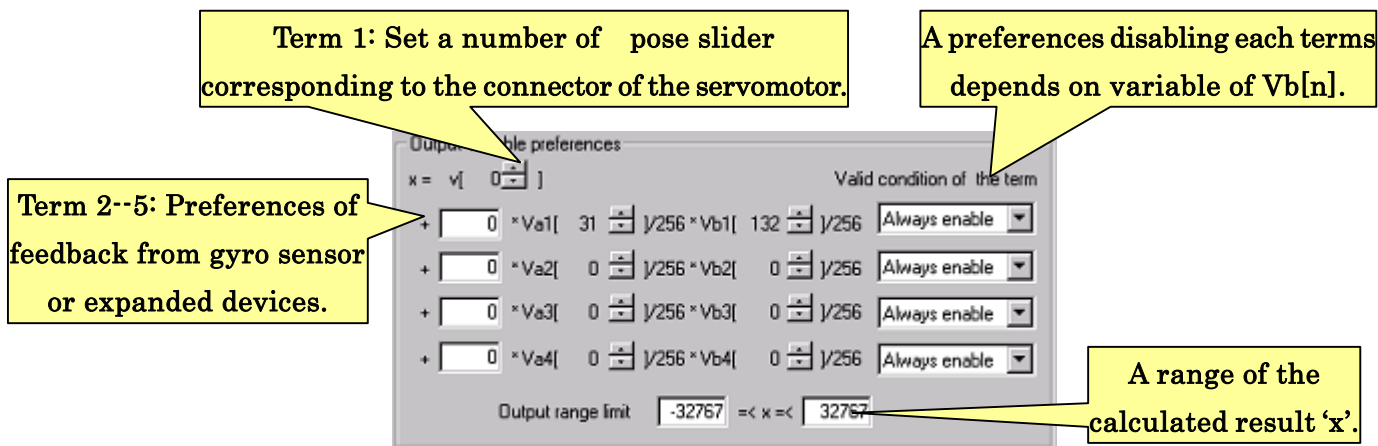
The top of dialog, select a servomotor to be set.



In the “Basic preferences” section, you can edit a name for the number of servomotor, and set a power supply timer when to start turning on the servomotor after the CPU boots. The value of the power supply timer is delay of turning on the servomotor from the CPU board start the sequence of power supply. The servomotor indeed moves after this delay.

If the CPU board turns on all servomotor at once, the total rush current is extremely high resulting the CPU in a forced reboot due to absence voltage. Thus, using this preference, the rush current can be de-concentrated to solve the power supply problem.

The sequence of power supply is recommended to be from the end of body. This reduce the risk of causing motor lock.



In the “Output signal preferences” section, you can set preferences that define the output value. Angle of servomotors is usually set by pose sliders. However, in case you connect the gyro sensors in expanded board, you need to set the sensor signal to feedback to servomotor control.

One section of setting composes an equation with variable parameter including constant values and coefficients. By default setting, the top of the lines includes “ $x=v[]$ ” that is already set a number of the servomotor connector of the CPU board, and the other variables are usually set to 0. The calculation result, that is a control signal, corresponds to the pose slider value by default, since the terms of 2nd to 5th in the equation ignore because of multiplied by 0.

In the equation, the terms of 2nd to 5th have the following structure.

$$(+[a \text{ constant value}] * [a \text{ real number of } Va[n]] / 256 * [a \text{ real number of } Vb[n]] / 256)$$

You can set an integer number between -32768 to 32767 as a constant value, and between 1 to 255 as $Va[n]$ and $Vb[n]$. (Setting of 0 means substituting a real number 0.)

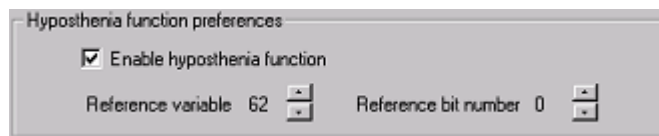
The sensor signal incoming from gyro sensor can feedback to control the servomotors by setting 2nd to 5th terms in the equation. For example, the parameters are 220 as a constant gain for a joint, 31 as a $Va[n]$ value for a gain enable of changing in each pose, and 132 as a variable number to preserve the sensor sampling result.

Using “Condition of enabling term”, the term can be ignored when the $Vb[n]$ is a certain range. “Always true” means that the term is always used for the calculation. “ $Vb[n] \geq 0$ ” or “ $Vb[n] \leq 0$ ” mean that the term is used when value of $Vb[n]$ is larger than 0 or smaller than 0 respectively.



In the “PWM preferences” section, you can set a conversion equation to calculate the PWM (Pulse-Width Modulation) from ‘x’ that is a result ‘x’ of “Output signal preferences.” The equation has two blank boxes to substitute integer value between -32768 and 32767. The boxes are the gradient and the intercept. The ‘x’ in the equation substituted with the result of the ‘x’ from the equation of “Output signal preferences.” Since range of ‘x’ is between -32768 and 32767, the range of 2nd term is between -1.0 to 1.0 of real number.

To check range of time-width that PWM outputs, after editing two blank boxes, calculate the result of the equation with maximum ‘x’ (-32768) and minimum of ‘x’ (32767). The results show maximum and minimum pulses of PWM. Convert the results to the unit of time. The unit is 1/15 [MHz].



In the “Joint free function preferences” section, you can on or off joint free function for each servomotor in each pose. When it is enabled, the reference bit of the reference variable switches servomotor on/off by 0/1.

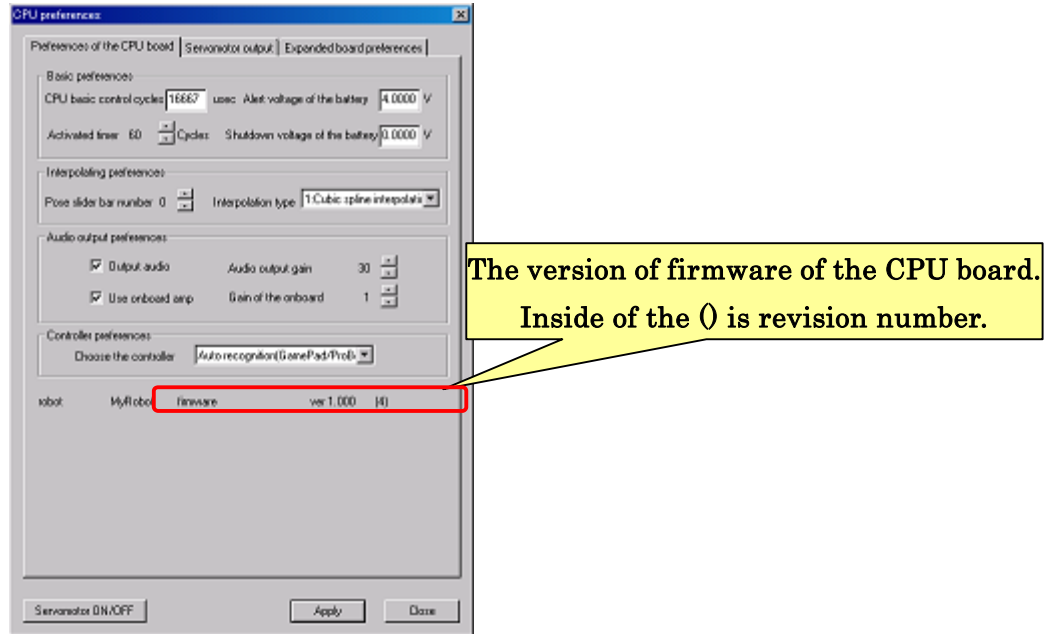
By default, joint free function of CN1-1 to CN3-4 is controlled by 0th to 15th bits of referenced 62nd variable, and the function of CN3-5 to CN5-6 is controlled by 0th to 13th bits of referenced 63rd variable.

The referenced variable number is editable. If you set the referenced variable 241st variable, you can use joint free function from your controller.

Note that, however, this setting is not synchronized with the checkbox of the pose slider meaning that the checkbox of pose sliders changes status of default reference variable. In this case, the checkboxes of pose sliders are ignored.

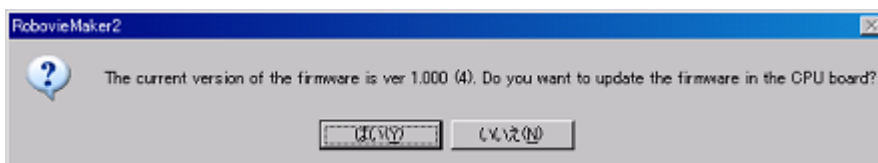
5-5. Updating firmware of CPU board

You can check the version of the firmware in the CPU board from the “CPU preferences” tab of the CPU preference dialog box. (See below picture.)



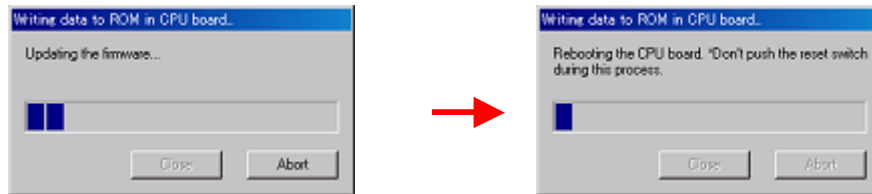
If the following message appears in the low side of the dialog box, the firmware is older version than this software expects. In this case, the software cannot use all the function, such as not supporting some expanded board.

Finishing download, then update the firmware from this software. Connect your PC and the CPU board. Then Click “Help”--”Update firmware” from the menu. The following dialog box appears, click “Yes” to step next.

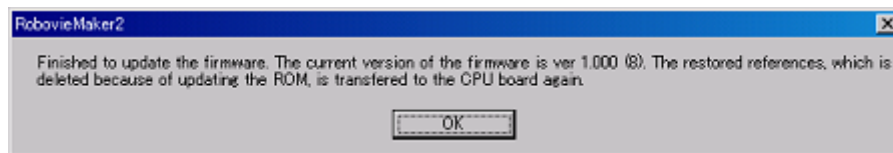


After clicking “Yes”, Since an appearing dialog asking you to select a firmware file, select the downloaded latest firmware (*.bin; if the file is compressed, uncompress it). Then click “Open”. **Be careful not to select wrong program files. In the worst case, the CPU board gets to be unable to communicate with PC resulting in requiring the CPU board repaired. Select a correct program file.**

After selecting the firmware file, the update sequence starts. During the update, since the following dialog boxes appear to show you update progress, do neither disconnect the cable connecting your PC and the CPU board, nor push the reset switch.

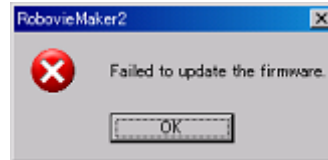


After the firmware is updated, the following dialog appears. Confirm if the firmware is correctly updated.



After verify the version of the firmware changed to latest version, click "O.K." button in the dialog. Then, the data restoration of the CPU board starts. The settings of the mode switch and audio files are transferred to the CPU board. Wait seconds until the transfer finishes.

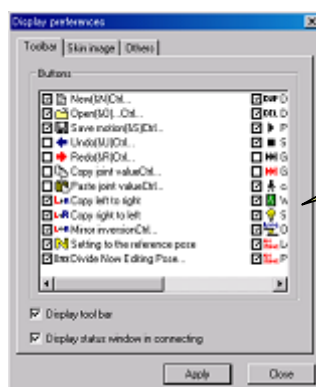
Note that, after updating the firmware, if the following dialog box appears, the firmware you selected was maybe not correct file type. Click “O.K.” to close the dialog, and try update again to use a correct firmware file.



When you try to update the firmware again after the update is failed, if you get a message, “The CPU board is not connected...” in a dialog, the firmware is destroyed because of the wrong firmware. You cannot use the CPU board unless the firmware repaired in the makers. Call to the contact list in the end of this manual.

6. Display preferences

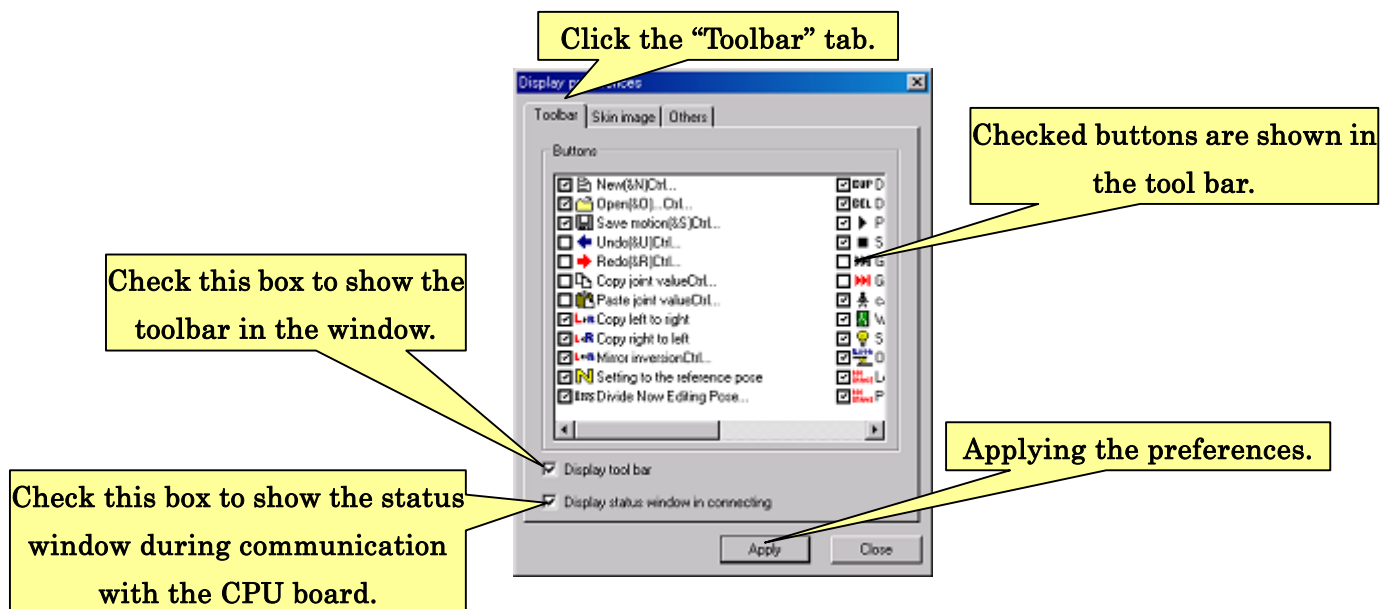
This section details about screen displayed settings in pose area and motion area, and choosing button types. To change them, click “Project preferences”--”Display preferences” from the menu. After clicking, the following dialog box appears.



A dialog to configure display preferences.

6-1. Selecting displayed toolbar buttons

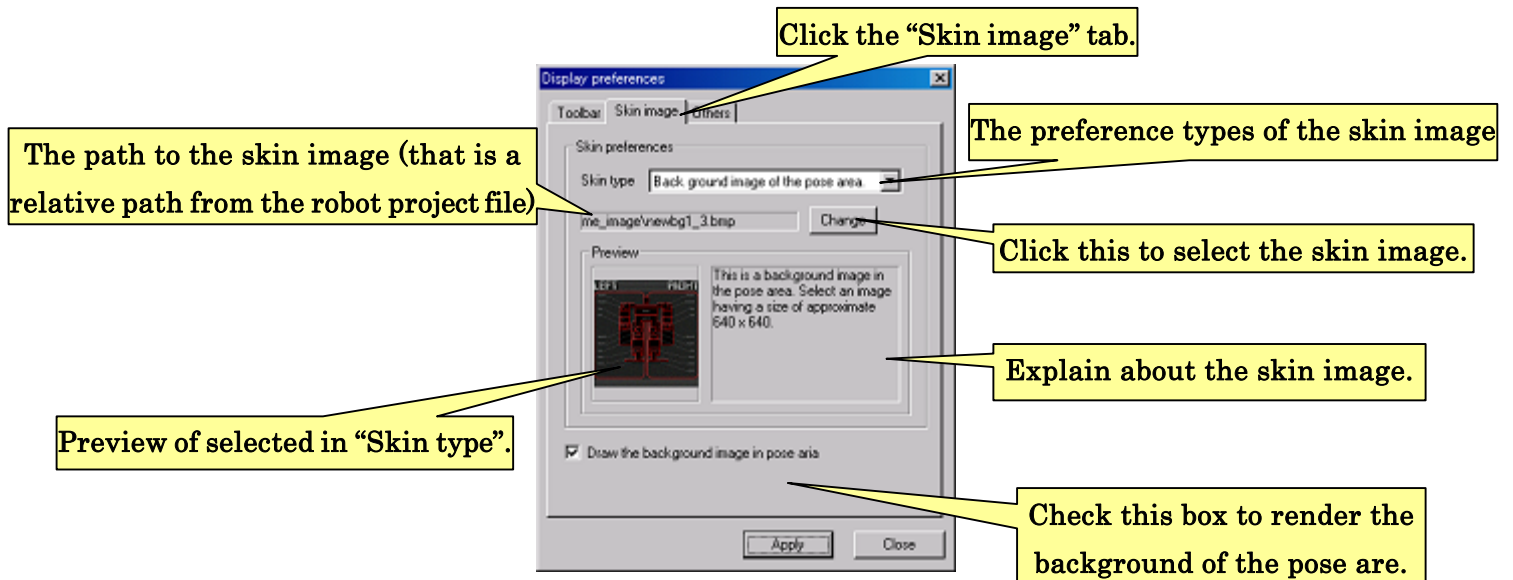
To select displayed buttons in the toolbar, click the “Toolbar” tab. After clicking, the following dialog box appears.



6-2. Skin preferences

Since this software can use general types of image files, you can select any images for pose sliders and background of the pose area. These image files are called “Skin.” The skin can be changed in this software. Note that, however, if you use too big image as the skin, the rendering of dialog will collapse, and it is sometimes out of control.

To change the skin, click “Skin” from the tab in the dialog.

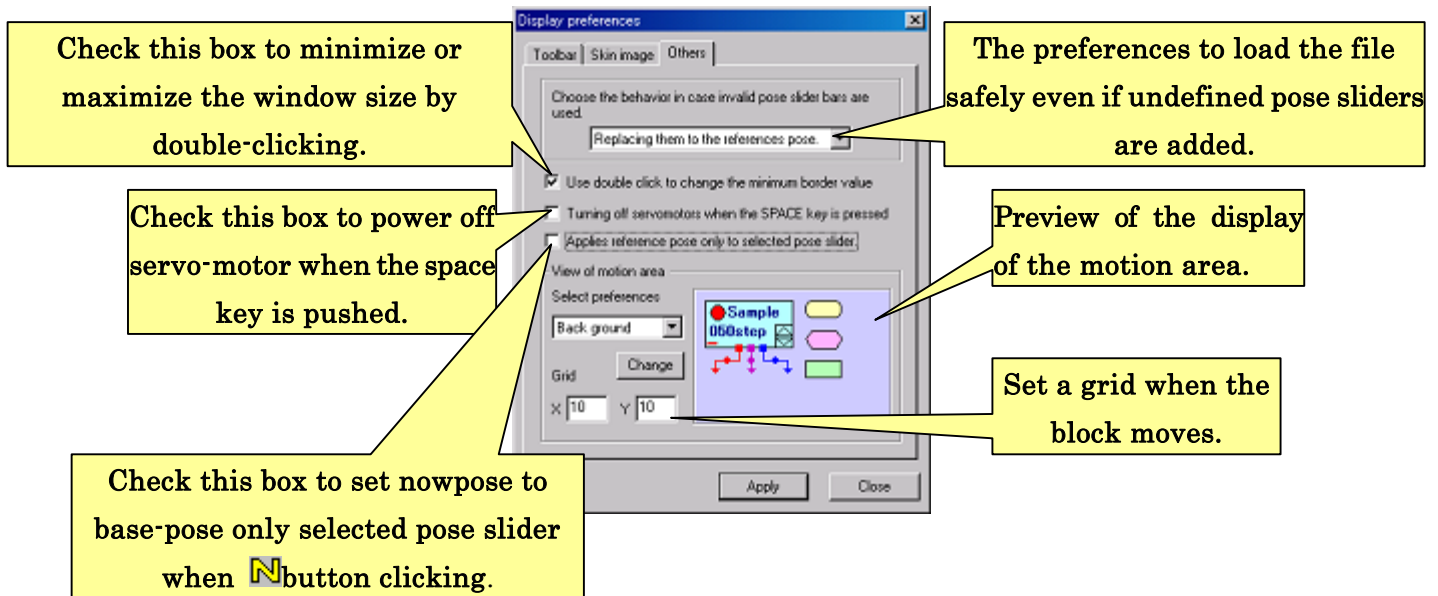


Supported image format is only the Windows Bitmap. Once you select the skin image, the location preserved as relative path. Thus, the image file is recommended copy into the project folder, since another location of the image file causes decoupling of the path to the image file when the project folder moved. In this case, the software will fail to render the skin.

Note that a large size image, such as the background image in pose area, consumes the CPU power to be rendered on the display. When you feel the software running slow because of using low power PC, uncheck the “Display the background” checkbox, and the skin image is not displayed resulting in improving the performance.

6-3. Other preferences

You can set preferences of software system and the motion area in the “Other” tab in the “Display preferences.”



The combo box below of the message, that is “If loaded motion file include the pose slider you currently do not use,” you can choose the behavior of the software when a saved motion file is loaded and does not include the all the pose slider bars currently displayed. How to import the values, that is undefined pose slider in the motion file, has to be chosen.

This is a setting for preventing the undefined pose sliders from being loaded and decoupling with the loaded motion file. If the “Undefined pose sliders are replaced by reference pose” selected, all the values of undefined pose slider currently displayed are replaced by the values of the reference pose. If the “Undefined pose slider is preserved” selected, all the value of undefined pose slider are left and preserved. If the “Open the dialog to choose the setting” selected, when you load the motion file, a dialog appears to ask you to select which of two settings. (If there are no decoupling, the dialog box does not appear.)

If you make a check in the box of “Switch the border to minimum motion area when double-clicked,” by double-clicking the motion area or the pose area, the size is minimized. If you double-click again, the size is restored.

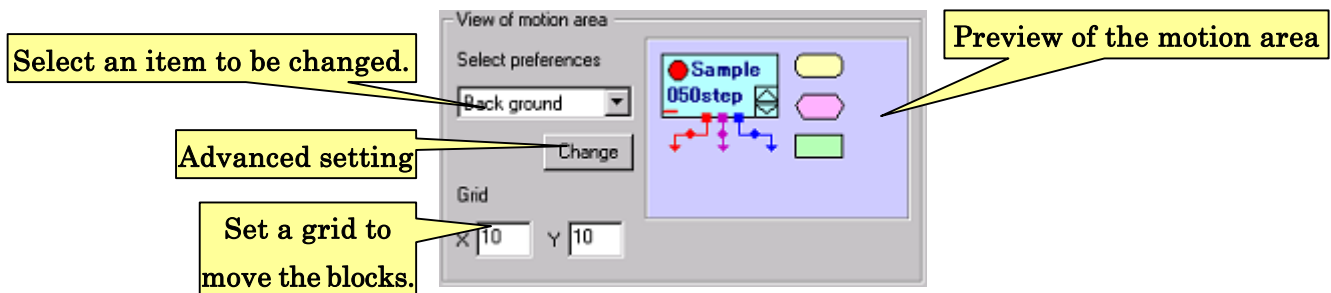
If you make a check in the box of “Turning off servomotors when thhe SPACE key is pressed”, you can turn off the servomotor by pressing SPACE key during editing motions. This can be used to avoid a malfunction of the servomotors in emergency.

If you make a check in the box of “Applies reference pose only to selected pose slider”, when you click the **N** button some pose slider are selected, they are set to the reference pose. If you didn’t select any pose sliders, all the pose sliders set to the reference pose.

6-3-1. Display preferences in motion area

In the “Motion area preferences” section of the low side of the “Skin preferences” dialog, you can set font colors, font types, and block and back ground color. In this preferences, you can change “Background color in motion area”, “Block color”, Block border color”, “Font color and font in block”, and “Arrow color of flows”.

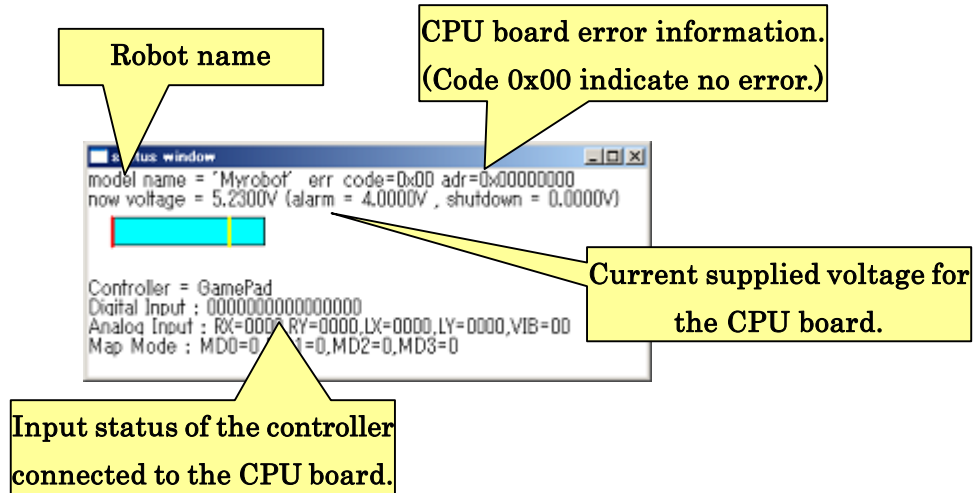
After selecting the item from the list, when you click the “Change” button, the following dialog box, in which you can change the preferences, appears. The previews is also change based on the item you selected.



In the “Grid”, you can change the resolution that is used for the blocks to move with SHIFT key down by pixel.

6-4. Status window

To show the status window when the CPU board starts the communication with this software, check the “Toolbar” tab and the box displayed on the low side of the appearing dialog.



In the status windows, the following items are displayed; name of the robot, error status of the CPU board, current voltage of the power supplied to the CPU board, and the button event from connected controller.

The voltage supplied to the CPU board is displayed as a horizontal bar in real time. You can easily monitor the status. The red vertical line on the left is the shutdown voltage the CPU reboots, and the yellow line at the middle is the alarm voltage the software notifies you the warning. The color changes based on the voltage status as follows. If the voltage is larger than the alarming voltage, the bar graph is aqua blue. If the voltage is larger than the shutdown voltage and smaller than the alarming voltage, the bar graph is yellow. If the voltage is smaller than the shutdown voltage, the bar graph is red.

Note that the status window will not appropriate display if the lower voltage than 10V is supplied because the CPU board status can be obtained due to limitation of the specification.

7.Support information

In case you meet the following problems during you use this software, at first, see the FAQs in this section that state the problems and how to solve them. When you can't them even if you try the how-to, or when you can't find your problems on the FAQ list, please contact the help desk that is listed on the end of this manual.

Official support web site

In the official support web site, the latest information about the CPU board, its latest firmware, and the latest software. Check the web site.

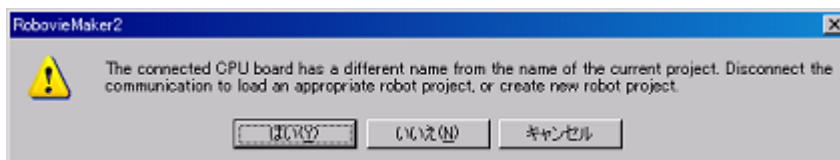
URL of the official support web site:

http://www.vstone.co.jp/top/products/robot/support_i.html

Q: Question, A: Answer

Q. In the case that the built-in firmware seems be broken due to failed transferring a firmware or the initialization of the CPU board.

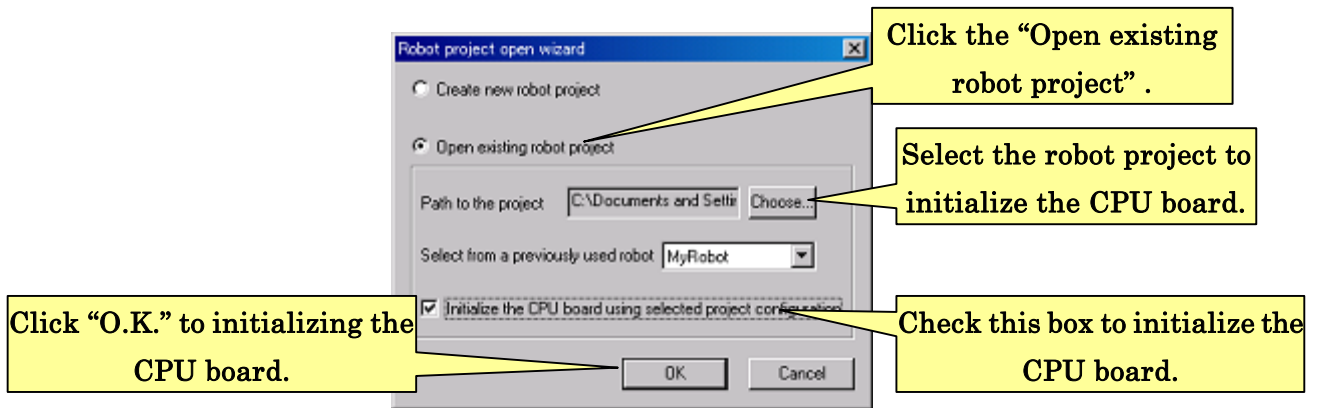
Q. The following dialog box appears when your PC starts communication with the CPU board.



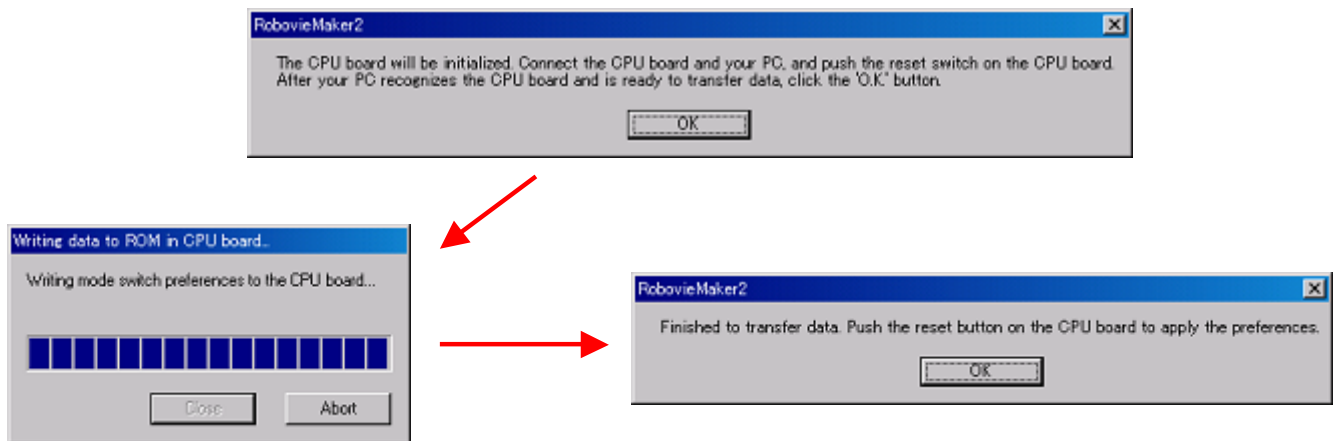
A1: After transferring the firmware data or initializing the CPU board, there are cases that the CPU runs the old firmware due to failing to reboot. Push the reset switch on the CPU board and check if the problem is solved.

A2: Confirm that you initialized the CPU board with an appropriate robot type (or name). If you already have a robot project and want to use it, click the "Cancel" button, when the above dialog box appears, to close it. Then load the robot project you want to use. To load the project file, click "File"--"Load robot project" from the menu and select the project file which is initialized with an appropriate type of the CPU board.

A3: In case the transferring firmware data to the CPU board is failed, the built-in firmware can be destroyed. ROM in the CPU board need to be initialized to repair it. Click the “Yes” button. After clicking, since the following dialog box appears, configure the project as the following figure shows.



After configuring, click the “O.K. button. The following dialog box appears to initialize the CPU board.



After the initialization has finished, push the reset switch on the CPU board to reboot it.

Q: This software does not recognize the CPU board.

A1: Confirm that your PC recognizes the CPU board. If your PC does not recognize the CPU board, check the cable connection at first. Then, try pushing the reset switch and confirm that the CPU board works.

A2: Check the connection between connectors and cables on the CPU board or expanded boards. If the connector is connected with wrong direction, the CPU board can be failed to reboot after the.

Q: The servomotors won't move.

A1: Check the cable connection between the CPU board and the servomotors.

A2: Check the power supply to the CPU board.

A3: Check that the power supply has enough voltage and current.

A4: Check the setting in "CPU preferences" is not "0: No interpolation".

A5: Check that the CPU board preferences are appropriate. If you doubt that the preferences are wrong, you can create a new robot project instead of repairing them.

Q: The supplied voltage drops because of not enough power supply when all the servomotor move, resulting in rebooting the CPU board.

A1: Set delays on timers being difference in servomotors to turn on in the CPU board preferences.

Q: The controller won't work connected to the CPU board.

A1: Check that the connector is connected appropriate direction. You can check if the CPU board receives information from the controller by seeing LED blinking orange on the CPU board.

Q: The CPU board won't output the audio.

A1: The CPU board has no built-in speaker. Connect a speaker to the CPU board for audio output.

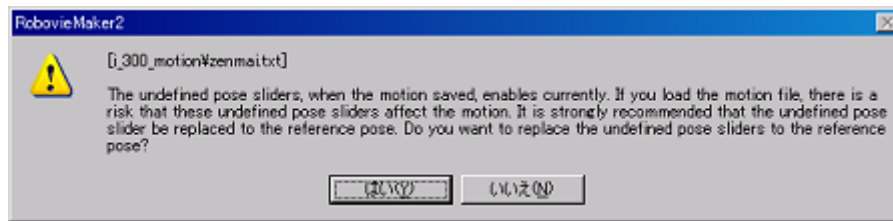
A2: Check that, in the CPU board preferences, you set neither "No use of audio output" or "No use of built-in amplifier". Also check that you set appropriate gain for the amplifier.

A3: The CPU board only outputs the audio from the speaker only when the power supply is enough from the connector. Check the power supply is enough.

Q: The CPU board doesn't work with a configuration and data transferred from my PC.

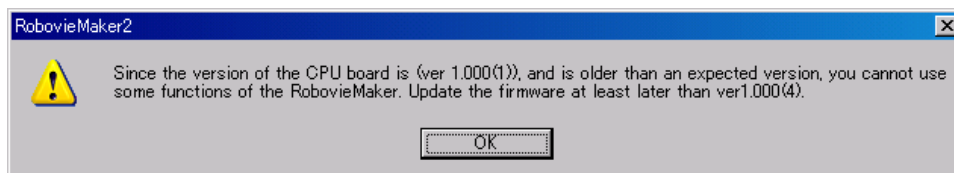
A1: When you transferred configuration data to the CPU board, the CPU board need to reboot. Push the reset switch after transferring configuration data to reboot the CPU board.

Q: The following dialog box appears when an extra pose slider is added after an expanded board is connected to the CPU board.



A1: This dialog box appears because there is a decoupling in a number of pose sliders between a loaded file and currently software status. It means that the software configuration of the number of the pose sliders is changed after the loaded file is saved. This dialog box warns you against risks that the undefined slider bar, in the motion file, can cause a motor lock and unexpected behaviors of expanded boards when the motion is performed. To replace all the undefined slider values as the reference pose, click the “yes” button. Since the undefined pose sliders preserved when you save the motion file after this replacement, the dialog will not appears again unless you don’t add extra pose sliders.

Q: The following dialog box appears when this software try to start communication between the PC and the CPU board. The communication is failed.



A1: This is because that the firmware version of the CPU board is older than the expected version, and that this software cannot deal with the firmware. This software support the firmware later than version 1.4. If the CPU board runs with older version than this version, update the firmware of the CPU board to the latest version. Download the latest version from our support web page, and update the firmware as detailed in the section “5-5. Updating firmware of CPU board”.

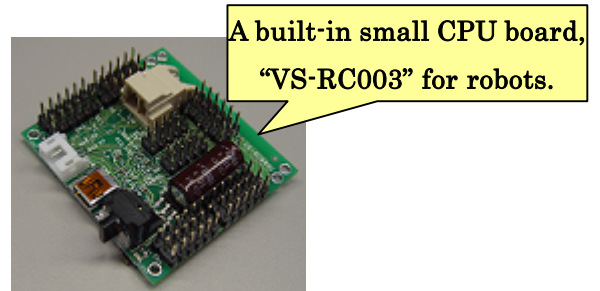
Appendix

Glossary

Technical words and codes upcoming in this manual is detailed in this glossary. If you meet some words and codes, refer this section.

(The) CPU board

A built-in small CPU board, called “VS-RC003”, for a robot to be controlled from this software.



Auto demonstration

A function that the CPU board performs a sequence of the programmed motions in order by itself.

Audio output

A function that the CPU board output sequence of the sound using the WAVE formatted sound data.

Expanded board

An optional device to expand functions of the CPU board, such as expanded boards of LED, accelerometer, and gyro.

Joint

An movable link of a robot that is usually with built-in servomotor to be controlled from the CPU board.

Game-pad

A commercially available controller pad. The Game-pad is usually connected to the CPU board through a connector to fit the connector type of the CPU board.

Servomotor

A type of motor used in radio control. Servomotors are connected to the CPU board to make movements of joint angles. Be careful neither to be cause motor lock nor to be injured at joints when servomotors move. Some types of servomotor produce

enough large power to harm you.

Servomotor calibration

A function to flat servomotors' angle variations. The servomotors have individual angle variations even with same control signal. This software can adjust the variations using servomotor calibration.

Skin image

An image file to be rendered as a background of this software.

Status window

A window to show current status of parameters in the CPU board during this software keeping communication with the CPU board.

Communication cable

A USB (Universal Serial Bus) cable for the PC to connect with the CPU board. A built-in connector on the CPU board is the mini-B type. Another commercially available USB cables can be used instead of the accessory USB cable.

Toolbar

A series of buttons displayed on the top of this software window consisting of several icons. Each function of buttons are detailed in this manual.

ProBo

A controller made, produced by Sanwa electronic instrument co. ltd. ProBo can be connected to the CPU board using a conversion connector to control a robot.

Pose

A posture of a robot realized by built-in servomotors. As an animation, a series of several poses performed continually is a motion.

Pose area

A sub-window to create a pose in this software. To move joint angles of a robot, you can manage it using pose sliders of this software. This software can output sound file and has configuration of expanded boards. These configurations are set in the pose area.

Motion

A series of robot movements such as walking and moving hands. A motion is usually composed of several poses performing continually to make a meaningful movement.

Motion area

A sub-window to register poses to construct motions. Orders and speeds of performed poses are configured in this window.

Motor lock

A state of a servomotor locked in a different angle from a controlled angle. During motor lock, the servomotor is suffered from tremendous loads. When “motor lock” happens, you will see several phenomena such as “large vibration of servomotors”, “large sound noises from servomotors”, or “heating up servomotors”. If you leave this state for long time, servomotors will burn out, resulting in malfunction of the servomotor. In worse cases, the broken servomotor generates smokes or fires.

Mode switch

A built-in rotary switch on the back side of the CPU board to select mode 0--9. When the CPU board runs by itself, the robot performs the action assigned to the switch number.

Robot project





A set of preserved configurations to control a robot. The robot project is discriminated from the others using name of the robot set on the CPU board. To use this software, you need to load at least one robot project. If this software cannot find any robot project in search paths, you need to create a new robot project fitting to your robot. You also create it when you want to use a new robot.

Robot project file

A file, generated when you create a robot project, to save all preferences. (The extensions of the file is “*.rpj”.) If this software is installed in the PC, by double-clicking the robot project file, this software starts with the robot project loaded.

Name of buttons in toolbar

The names of button icons in toolbar are shown as the following list. Although this manual uses the buttons as the icon figures in explanations, use the texts to mention about the button icons when you contact or e-mail to the support center.

-  -- New button
-  -- Open button
-  -- Save button
-  -- Undo button
-  -- Redo button
-  -- Copy button
-  -- Paste button
-  -- Left to right button
-  -- Right to left button
-  -- Inverse button
-  -- Neutral button
-  -- Insert pose button
-  -- Duplicate pose button
-  -- Delete pose button
-  -- Play motion button
-  -- Stop motion button
-  -- Go into normal flow button
-  -- Go into branch flow button
-  -- Servomotor calibration button
-  -- Write data to CPU button
-  -- Servomotor on/off button
-  -- Online button

(2008.9.25)