MAE 263B
Dynamics and Robotics

Lab 3

Lab 3: Offline Robot Programming

Instructions for Offline Programming of the Denso Robot

1 Task

The goal of Lab 3 is to write a program offline (e.g., in WINCAPSIII) to pick up three geometrically identical blocks and stack them to build a 'gate' structure as indicated in **Error! Reference source not found.** figure below. This is intended to give you some experience with an advanced industrial robot simulation and offline integrated development environment.

To achieve the task goal, you are required to understand the usage and instruction syntax of the Denso robot programming language WINCAPS. Some of the critical coordinates for your program are given, but you still need to figure out the rest to complete the program.

Figure 2 shows the layout of the fixture plate where the plastic blocks are initially located. It also shows the order in which the blocks should be moved. Blocks 1 and 2 are to be the 'pillars' of the gate while block 3 should be the 'cap'.

The dimensions of each plastic block are: 4''x3''x1''.

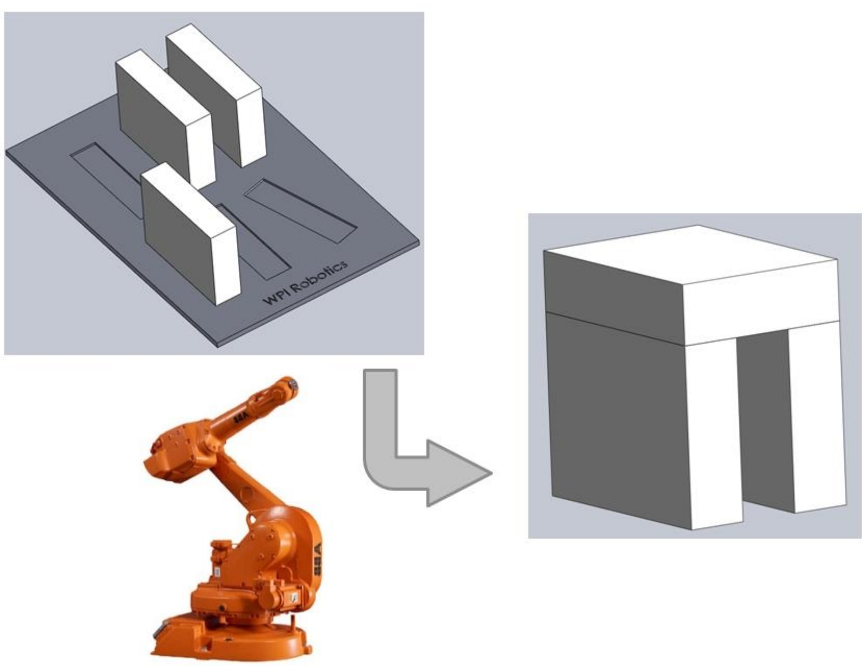


Figure 1. Stack Task

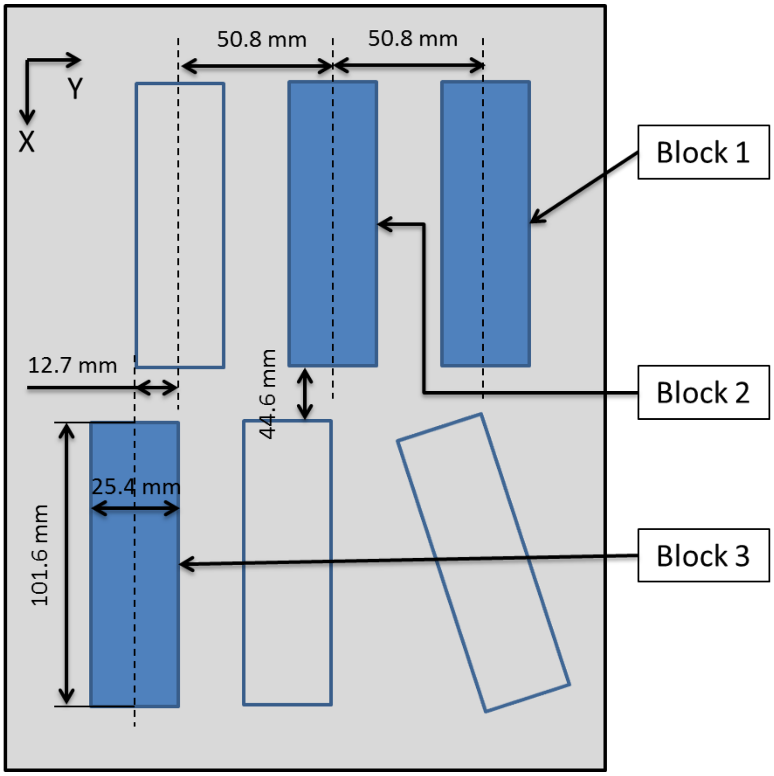


Figure 2: Fixture plate layout

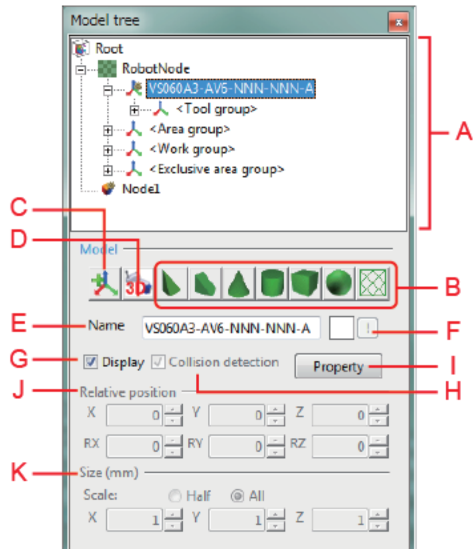
2 What you will have in the workcell

- The Denso robot (real as well as virtual)
- The Denso robot controller (real as well as virtual)
- Three white geometrically identical plastic blocks (real as well as virtual)
- A fixture plate to hold the plastic blocks (real as well as virtual)
- A computer equipped with the WINCAPSIII software
- In Lab 2 you used the Teach Pendant to write a program to move the plastic blocks from the fixture plate and stack them. In this lab you will write a similar program, but you will use a combination of station editing (what you did in the in-class exercises) as well as the integrated development environment in WINCAPSIII.

3 Instructions For Writing A Program Using WINCAPSIII

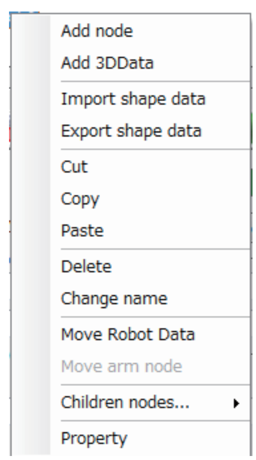
3.1.1 Arm Modeling

This window is for adding tools, work pieces, obstacles, and other objects to the equipment layout in the Arm 3D view window using 3D data from either the built-in simple modeling software or external sources.



A: Object Tree

This shows the object hierarchy behind the Arm 3D view. Right-clicking anywhere in the object tree area displays the following context menu.



1 Add node

This is for adding a node underneath the selected object.

2 Add 3D data

This is for adding an object using 3D data from an external source.

3 Import shape data

This is for reading in existing modeling data.

4 Export shape data

This is for saving the modeling data for the selected node or object as a file.

B: Add Buttons

These add an object of the shape indicated on the button. There are seven shapes available.

C: Node Button

This adds a node.

D: External Button

This reads in 3D data from an external source.

E: Relative Position

These fields specify the relative position for the currently selected object.

- X, Y, Z
  Relative x/y/z coordinates in mm.
- RX, RY, RZ
  Angles relative to the corresponding axis, indegree.

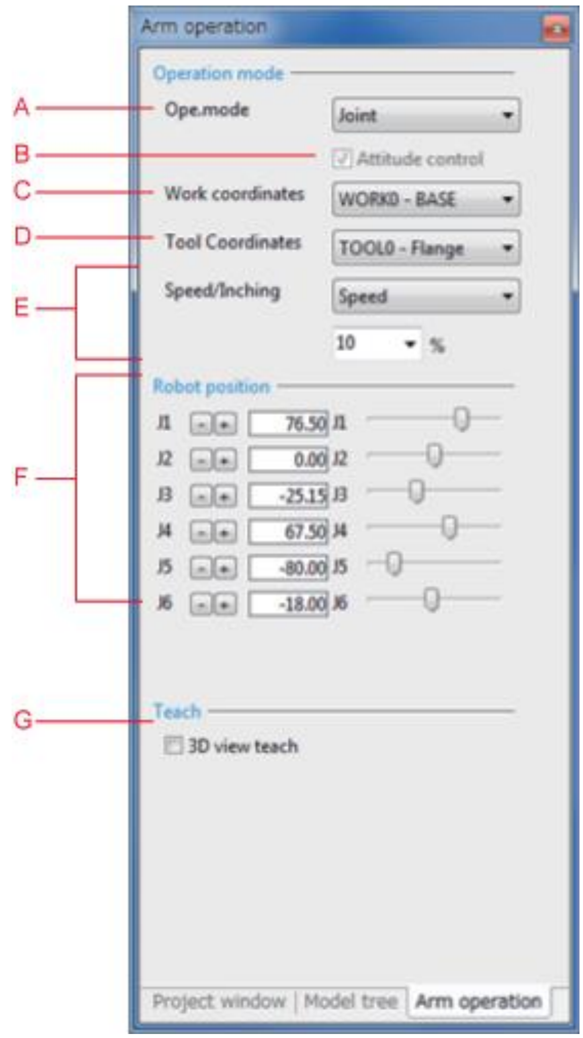
K: Size

These fields specify the dimensions for the object selected in the tree.

- X, Y, Z
  Dimensions along the x-, y-, and z-axes, in mm.

3.1.2 Arm Operation

This window is for moving the simulated robot in the Arm 3D view window.



A: Ope. Mode

Select movement mode.

- Axes: operates all axes.
- XY-bisection: move according to the selected work coordinates. RX/RY/RZ rotate around the work coordinates centered on the tool origin point.
- Tool: moves according to the selected tool coordinates. RX/RY/RZ rotate around the work coordinates centered on the toll origin point.

B: Attitude control

This option enables/disables the Attitude control function under X-Y mode or TOOL mode operation.

C: Work Coordinates

Select work coordinates.

D: Tool Coordinates

Select tool coordinates.
E: Speed/Inching
Select normal running or inching.

- Speed: While the “Move position” button is held down, the corresponding robot joint continuously moves at the specified speed (%).
- Inching: Each time the “Move position” button is pressed, the corresponding robot joint inches by the specified amount of movement (mm or degree)

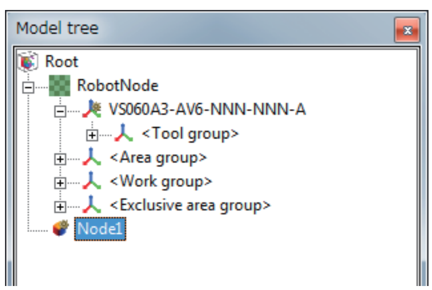
F: Move Position Buttons

Moves the robot displayed in Arm 3D view in accordance with the selected movement mode.

3.1.3 Simple Modeling

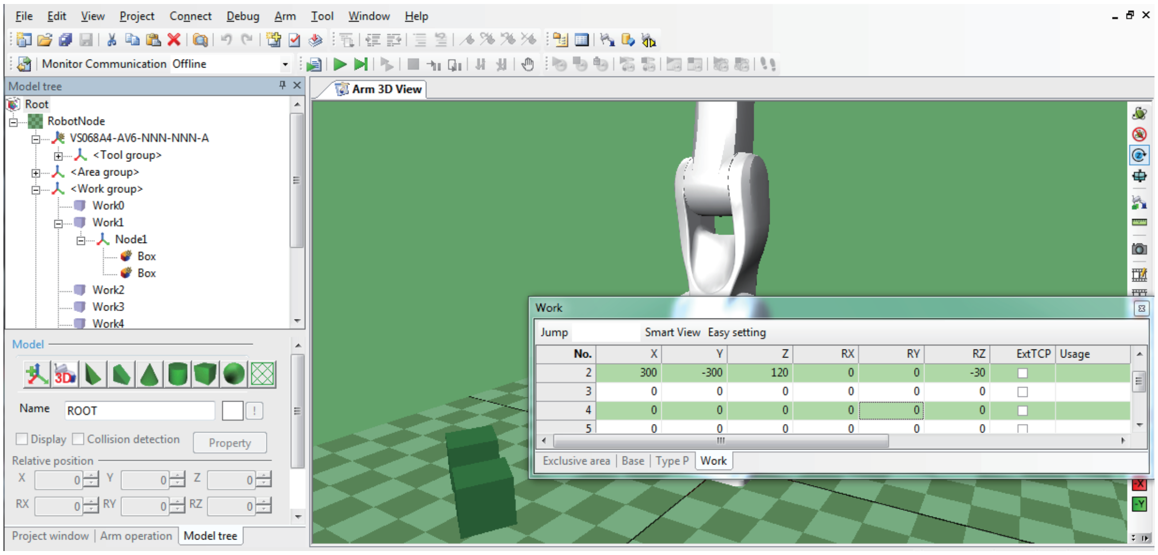
Simple modeling is a built-in -tool that allocates pallet bench, pallet, work piece, and other objects in Arm 3D view.

The Model tree window presents a hierarchical display of the objects in the Arm 3D view. The top node is Root, which starts out with the following predefined nodes. Noeds are virtual items to combine multiple objects and name them.



A: Work Group

This group contains the coordinates for eight work pieces, Work0 to Work 7. Se work coordinates on the [Work] screen. Note that Work0 cannot be edited.



B: Tool Group

This group contains the coordinates for 64 tools, Tool0 to Tool63. Set tool coordinates on the [Tool] screen. (Tool0 cannot be edited.)

C: Area Group

[Area group] is used for 32 area functions from Area0 to Area31. Select the [Display] check box in [Model tree], and you can check each area in [Arm 3D View]. Set each area on the [Area] screen.

D: Exclusive Area Group

[Exclusive Area Group] is used for 32 exclusive area coordinates from ExArea0 to ExArea31. Select the [Display] check box in [Model tree], and you can check each exclusive area in [Arm 3D View]. Set each exclusive area on the [Exclusive area] screen.

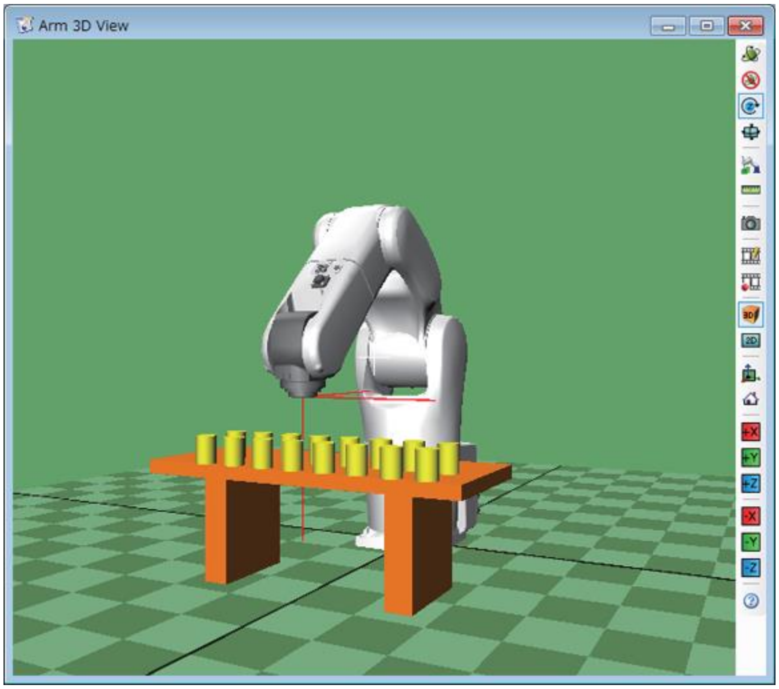
E: Adding Nodes

A node may be added anywhere in the object hierarchy. Select a node on the object tree, right-click, and choose Add node from the context menu that appears to add a new node as a child of the selected one.

3.2 Adding Objects

The following is the procedure for adding the pallet bench, pallet, work pieces, and other objects to the Arm 3D view window.

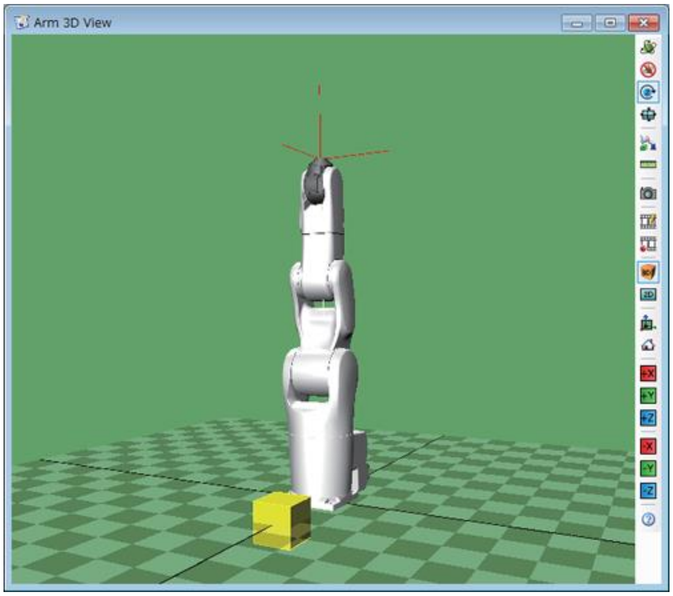
Example: pallet bench



Step 1: From the menu bar, click View, select Arm View to display Arm 3D View.

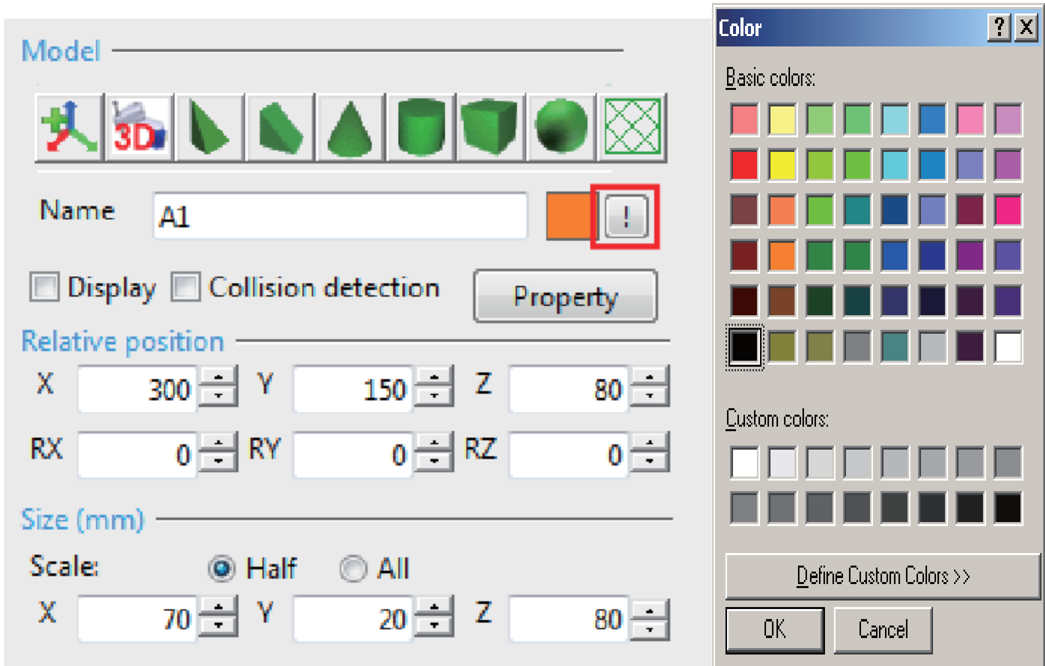
Step 2: On the object tree, select a node that will be parent of a newly created object.

Step 3: Press Box button in the Model tree window to add a box to the Arm view window.

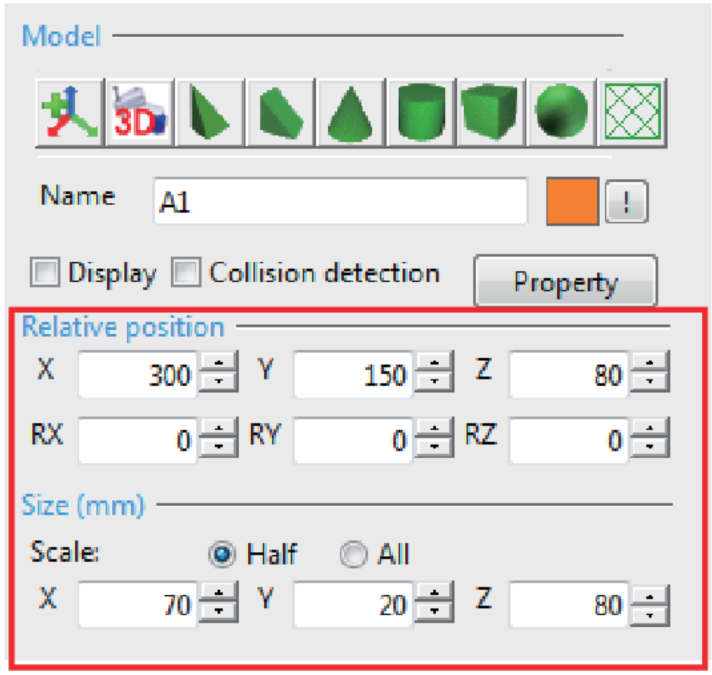


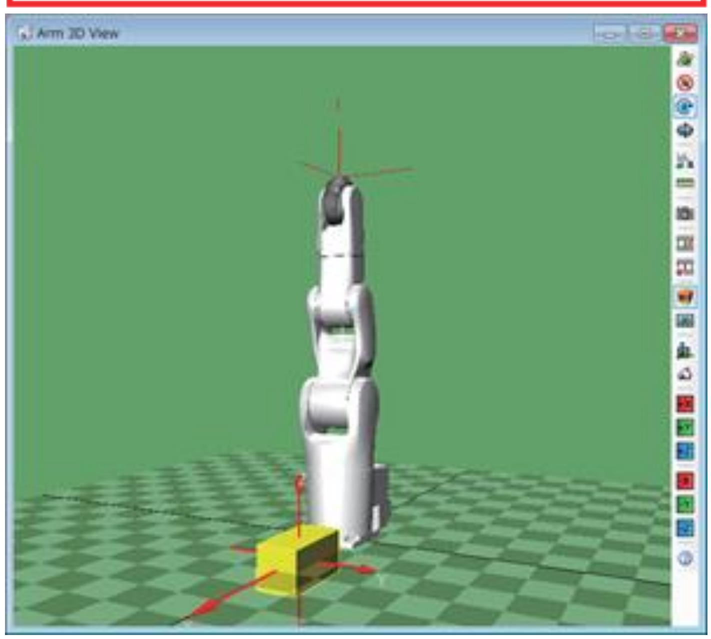
Step 4: Change the object name and color.

- Change name
  On the object tree, right-click on the newly created object, and then select Change name. You can rename the object.
- Changing color
  On the Model tree, click the [!] button and change the color on the [Color]dialog.

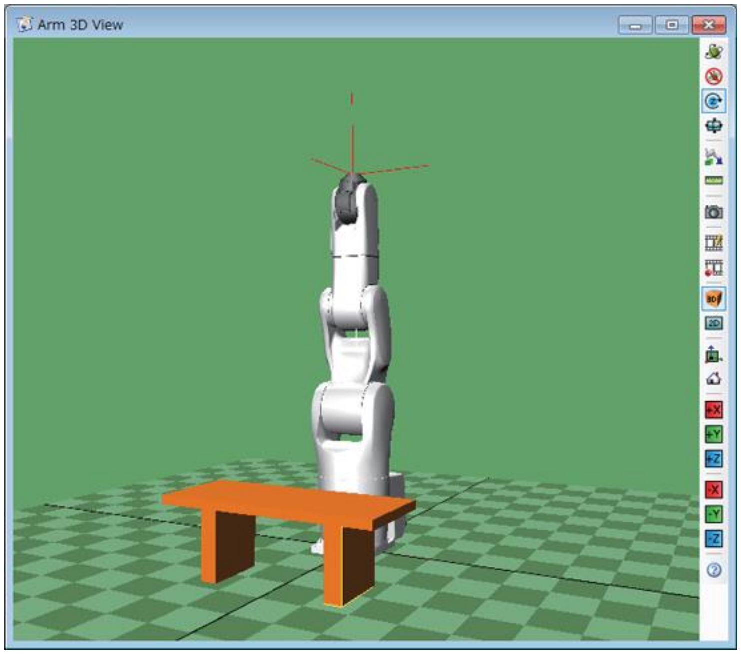


Step 5: Change the object's position and size



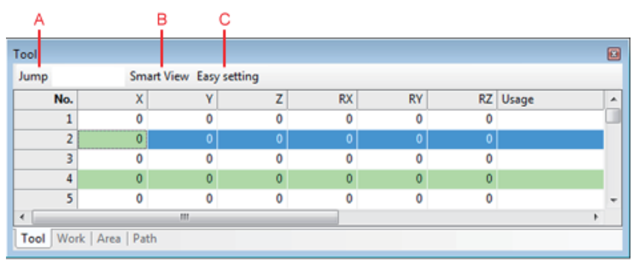


Step 6: Add objects to complete the pallet bench.
(Tip: Here, it is faster to copy and paste the above box as the second leg and then again as the starting point for the top.)



3.3 Arm Coordinates

3.3.1 Tool Setting



A: Jump

Display a selected line number

B: Smart View

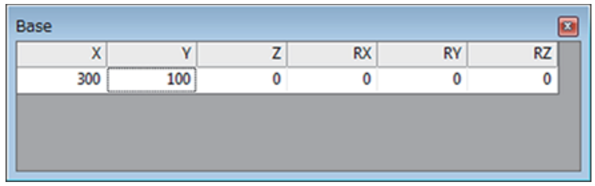
Display lines which Smart checkbox are selected

C: Easy setting

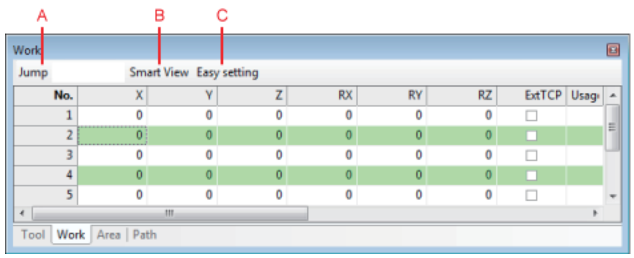
Display [Easy setting window] of the tool coordinates. The result of the Easy setting will be given in the tool coordinate of the selected number.

3.3.2 Base Setting

Set the base coordinate on the world coordinates system.



3.3.3 Work Setting

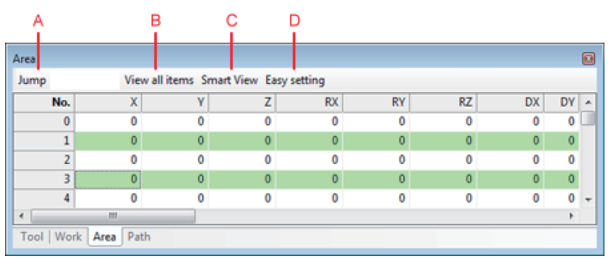


A: Display a selected line number

B: Display lines which smart checkbox are selected

C: Display [Easy setting window] of the work coordinates. The result of eacy setting will be given in the work coordinates of the selected number.

3.3.4 Area Setting



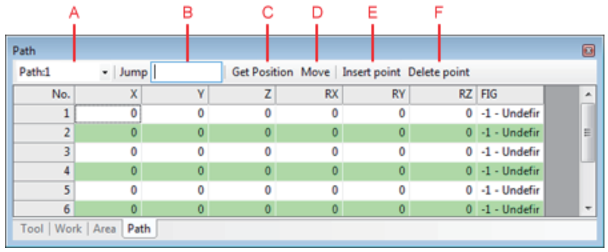
A: Display a selected line number

B: Switch between visible and invisible of the detail settings, such as the detection interval or detected pose

C: Display lines which smart checkbox are selected

D: Display of the area coordinates. The result of the easy setting will be given in the area coordinate of the selected number.

3.3.5 Path Point Setting



A: Select a path data from the combo box

B: Display the selected line number.

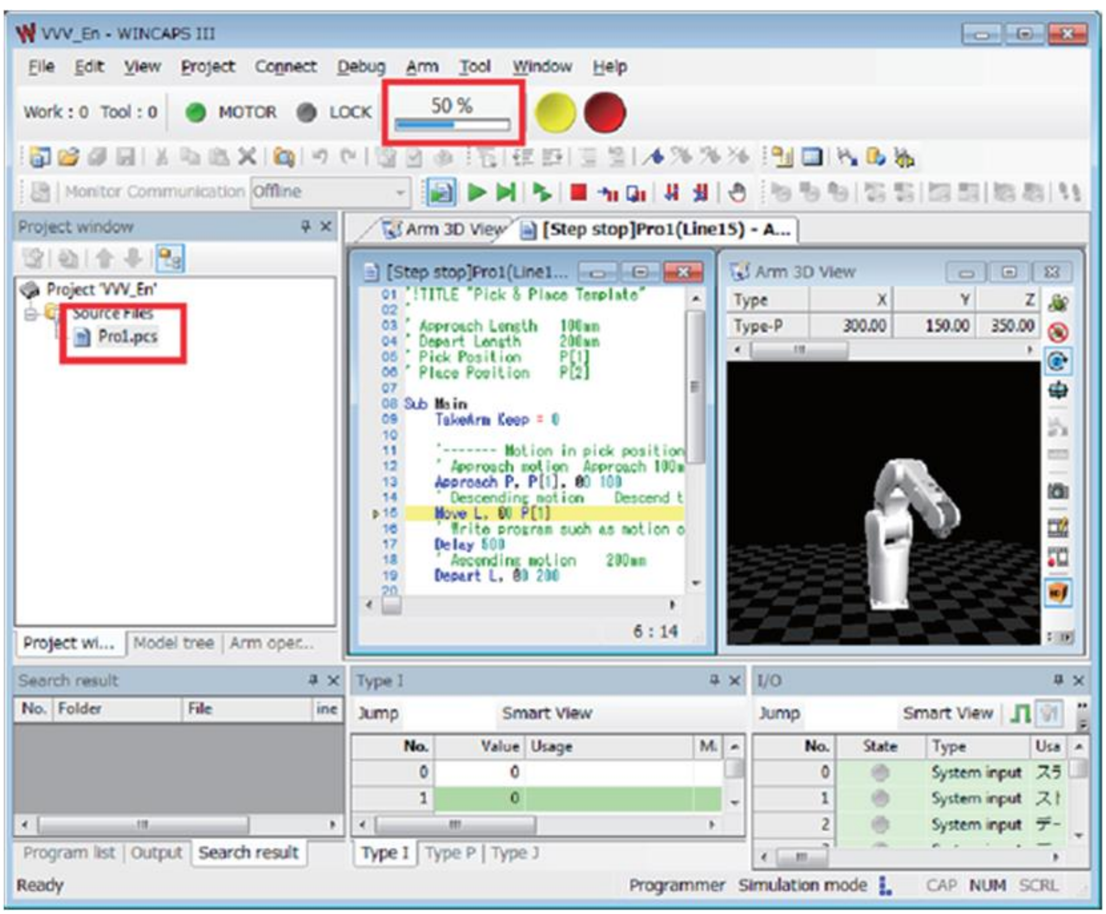
C: Overwrite the selected path point data with the current position data.

D: Move the robot to the selected path point

E: Insert a path point data in front of the selected path point.

3.4 Motion Check of the Program

You can check the cycle time, trajectory and posture of the robot by running the created program on the program.



Step 1: On the [Project window], select a program that you would like to check the motion. The selected program is displayed.

Step 2: On the [Debug] menu, click [Simulation mode].

Step 3: Set the external speed. On the [TP Status bar], click the speed button which is red-circled in the image above.

Step 4: On the [Debug] menu, select [Start a Task] or [Step in] to execute the program

Step 5: Check the program flows, the robot posture and the run time in each window.

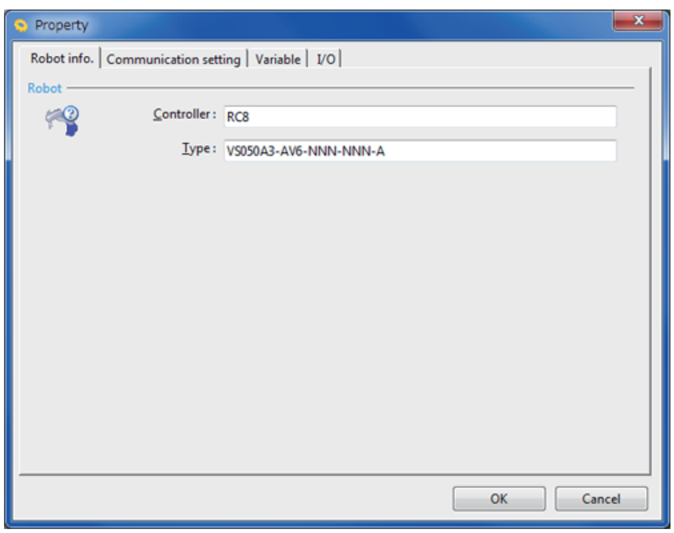
4 Procedure conducting the Lab

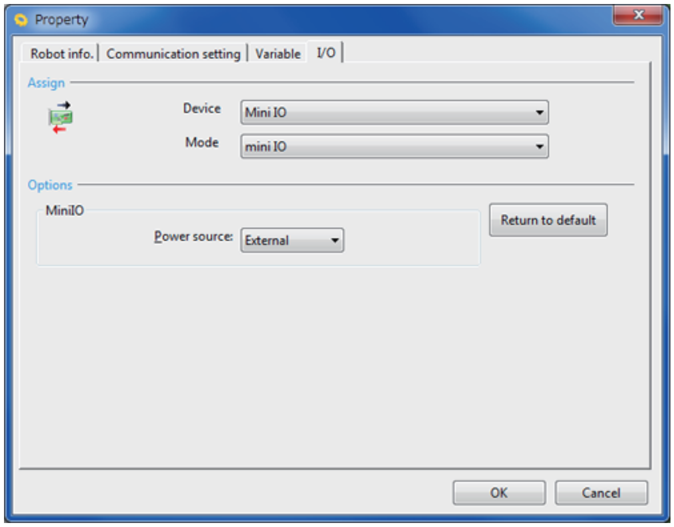
4.1 Creating a Project

Step 1: Choose Project Properties to display the property sheets for the project.

Step 2: Click the tab containing the desired configuration option for the settings.

Step 3: When configuration is complete, click OK button.





4.2 Import and position the Base Plate and Blocks

The next step is to position the base (fixture) plate in the virtual work cell. Import the base plate and rotate and position the plate to the position (-500, -1000,0) and orientation degree (90°,0,0) .

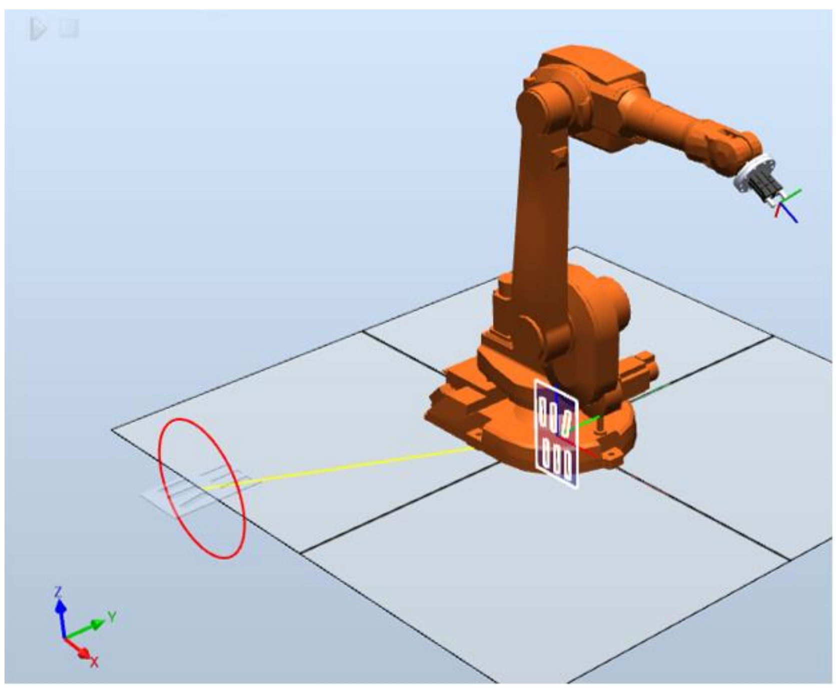


Figure 3. Base plate location parameters

Now import the plastic blocks in the same way as you imported the fixture plate. Import the first block and orient it as the figure showed below.

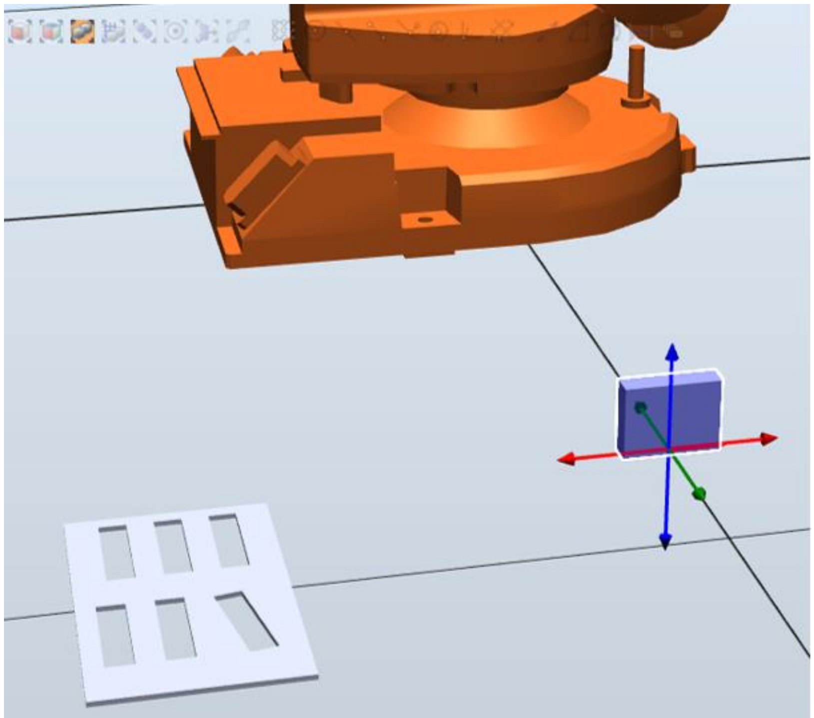


Figure 4. Imported and rotated block

The next step is to place the plastic block into position on the fixture plate. The plastic block needs to attach the two corners of a base plate slot. The same method used for the other two blocks.

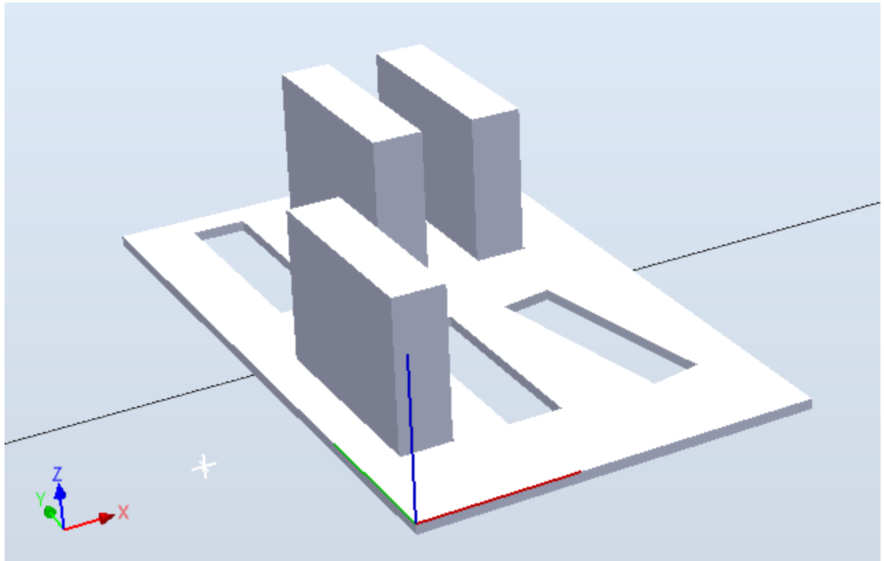


Figure 5: Starting positions for the 3 blocks and location of the workobject

4.3 Generate a workobject for the BasePlate

Create a Workobject in the corner of the base plate, shown in the figure above. The newly created Workobject should be positioned on the corner of the base plate.

The reason for creating the Workobject is so that the targets you are about to create are tied to the BasePlate workobject rather than the default workobject wobj0 (which is located at the origin of the robot's base coordinate system). If the targets are created using newly created Workobject, then if the base plate is moved for some reason, the block pickup targets can be easily corrected by simply relocating newly created Workobject to the new corner position of the plate.

4.4 Generate targets

The target should be in the Workobject created in the above step. Because we are going to be creating multiple targets, you might want to rename the newly created target to something like "Pickup_1" so it will be clear later on as to which target is which.

It may be that the orientation or position of the target will not be what you want. Remember that the robot will orient the tool to match the orientation of the target. If the target orientation needs to be changed use the rotate tool to reorient the target. Remember that you can also view the tool (or the robot) at the target to help you in deciding how to orient and position the target. Continue to create pickup-related targets for the other two blocks in the same manner.

Note: Give some thought when orienting targets as to how the orientation may affect various joint values. For example, the gripper exhibits symmetry about its Z axis, so 180° rotations about the Z axis may appear to be irrelevant. But the gripper's Z-axis orientation does matter as it will affect the value of joint 6. In general, you will want to keep the joint angles as small as possible. These are reflected in lower (absolute) numbers in the configuration choices.

4.5 Generating clearance targets

Before moving on to generating dropoff targets, you should generate clearance targets for the pickup positions. A clearance target is used to get the gripper into a pose that has it located (usually) directly over the block to be picked up (and ready for a linear move down to the pickup target). An easy way to do this would be to copy and paste a pickup target to create a duplicate, and then modify the copy to offset it in the appropriate Z direction by the desired amount. The figure below shows clearance targets located over each of the pickup targets.

Note that the pickup targets have been adjusted to allow for proper gripping of the blocks.

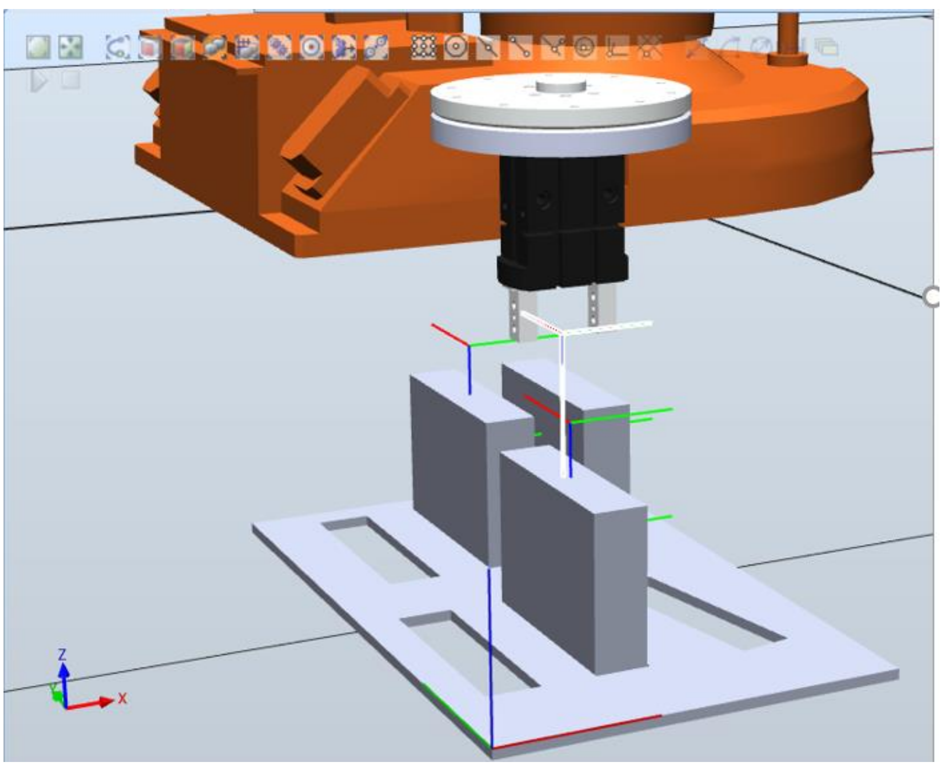


Figure 6. Clearance targets over pickup targets

4.6 Create a workobject and GateBase

First, we need to use Add Object command to create a GataBase and position it near the base plate. This is where you will have the robot construct the gate as shown in figure below. Create a workobject at one corner of the gate base.

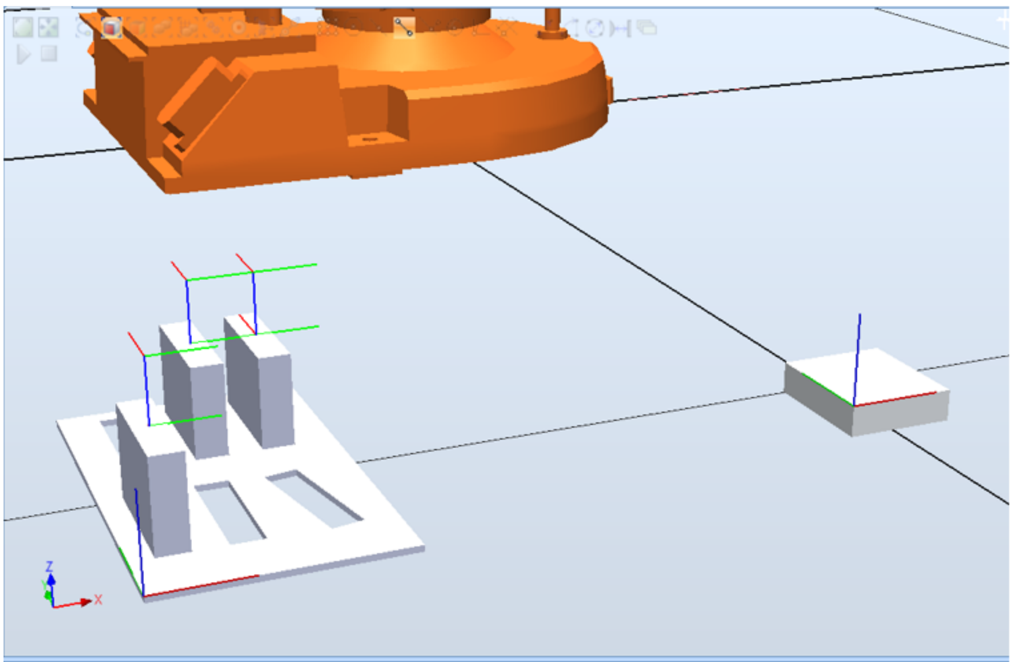


Figure 7. Location of Gate Base

4.7 Generating dropoff targets

You should also create a target for where each block will be placed as part of the gate being constructed – let's call these dropoff targets (e.g., "Dropoff_1"). Note that dropoff targets should be constructed with reference to the GateBase we have created above. Clearance targets will also be needed in this case too. [This is where the extra block(s) can come in handy. Use them to help create the dropoff targets. Once that has been completed, you can then delete those extra blocks or simply turn off their visibility.]

Think carefully about how you will need to orient the last block when using it to cap off the gate. You may need to create some intermediate targets and perhaps even set the block down, release it, and then re-grasp it in a different way in order to be able to cap off the gate.

4.8 Verifying the target configurations

Before you go any further, you should verify the targets. The first step is to select all of the targets for each workobject and check that they are all reachable. A target that isn't reachable means that there's no inverse kinematic solution that will allow the robot to reach the target in the chosen tool orientation.

If you have an unreachable target the first thing you should check is the tool orientation. It may be that the orientation is the problem and if you correct that then the target will be reachable. If there is no tool orientation that allows you to reach one or more targets, then you will need to move the target(s). This most likely implies moving the fixture associated with the targets (the base plate or the gate plate). If you move the base plate, you'll also need to relocate the blocks to their new starting location.

Once you have verified that all of the targets are reachable, you then need to select a configuration for each target. You can manually configure each of the targets. Try viewing the robot at each target as you examine the different configurations. Keep in mind where each joint is and how much it has to rotate to get to its next target.

4.9 Generate paths

Once all of the targets have been verified and configurations selected, you can now create paths. As noted before, consider creating a path per block. For example, Path_10 might contain only the instructions for moving the first block into position, Path_20 handles the 2nd block, and Path_30 deals with the 3rd block.

It is suggested that you create an empty path and then drag targets onto it; doing so will generate a Move instruction for each target. Note that you can set up a template for the instructions that will be created by making appropriate selections on the toolbar at the bottom of the WINCAPSIII window.

Once you have created all of the instructions for the path you can then edit individual instructions to change things if the default wasn't correct.

Once you have added the instructions to the path, you can test the robot's motion. Use the instruction provided in the section 3.

Think about the fact that the last target of the previous path will be the starting point for the robot on the currently executing path. So, you may want to pre-position the robot there if possible as that's where the robot would normally be starting from. This is easy to accomplish – just enable View Robot At Target and then click on the target or an instruction referencing that target.

Ultimately, you should be able to start the robot at (for example) it's kinematic home location (or some other location of your choosing) and then work your way from path to path.

The figure below shows the path that was generated for picking up and dropping off the first block. Before proceeding further, the configurations for the targets involved in this path should be verified.

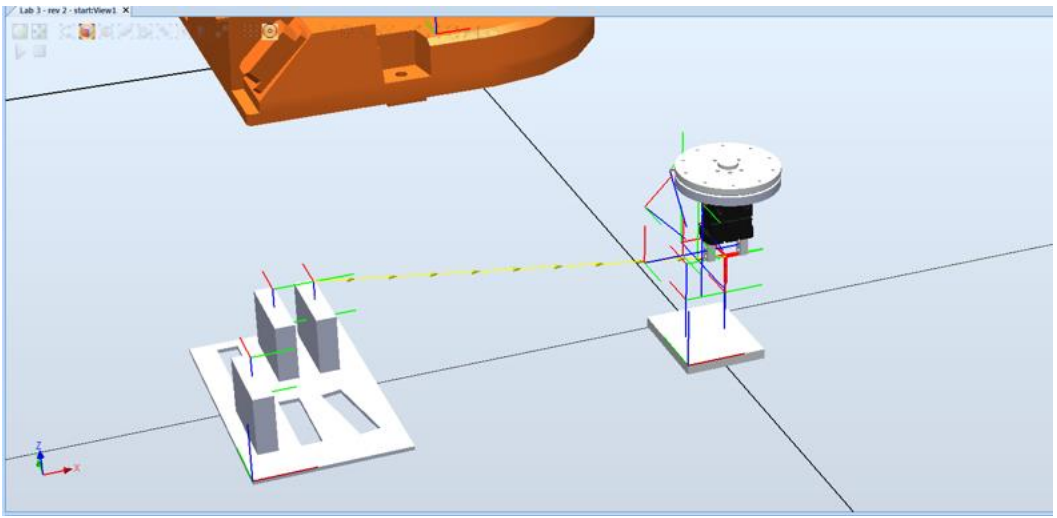
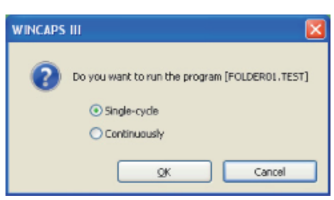


Figure 8. Example path

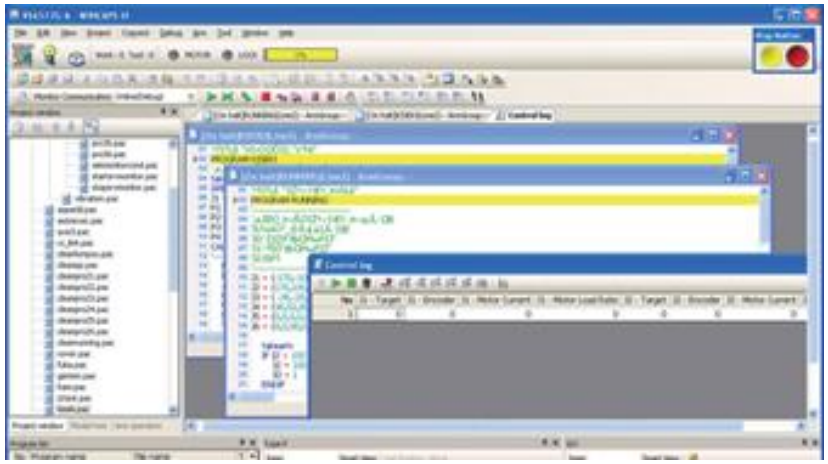
4.10 Launch the Program

Step 1: Double-click the program to execute in the Project or Program list window to display it in the Program view area.

Step 2: From the menu bar, click Debug, click Start a task. The following dialog box appears.



Step 3: Select an option and click OK button to run the robot program as specified. The robot will operate following the program's commands. The Arm 3D view, if open, tracks actual robot operation.



Step 4: Disable Debug function. From the menu bar, click Connect, point to Monitor Communication, and then select Offline.