

Writing Scripts in MotoSim to Manipulate Models

Introduction

MotoSim has a scripting capability (see “Model Script Manager” in the Simulation tab.) The Model Script Manager allows you to perform functions, like:

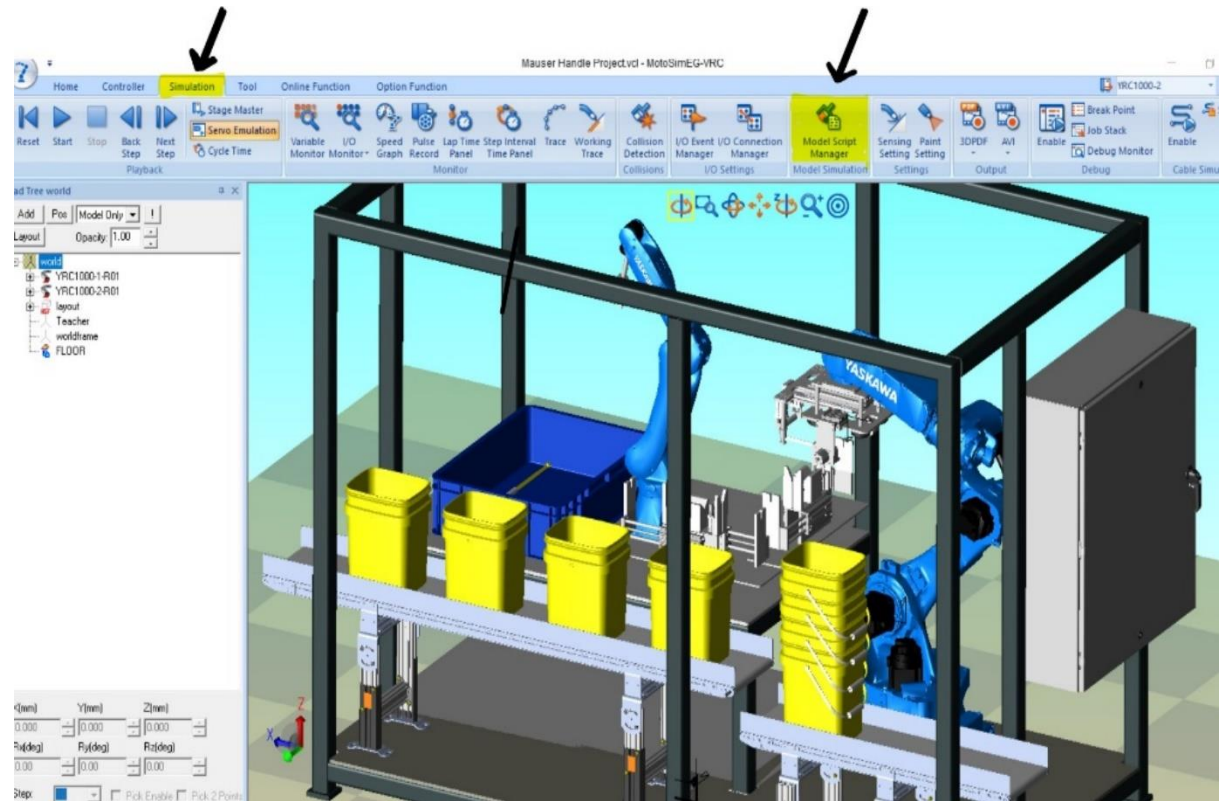
- HIDE or SEE models.
- Move a model, as if a conveyor is moving it.
- Actuate a tool model to make it look like you are picking product.

This document will describe how to use these scripting techniques.

This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

Creating a Script

First, build your Motosim cell and have all models in their correct locations. Make sure your cell has been saved before moving on. Once all that is complete, you will go to the Simulation tab, shown on the right. Then go to and open the Model Script Manager, as shown in the highlighted square.

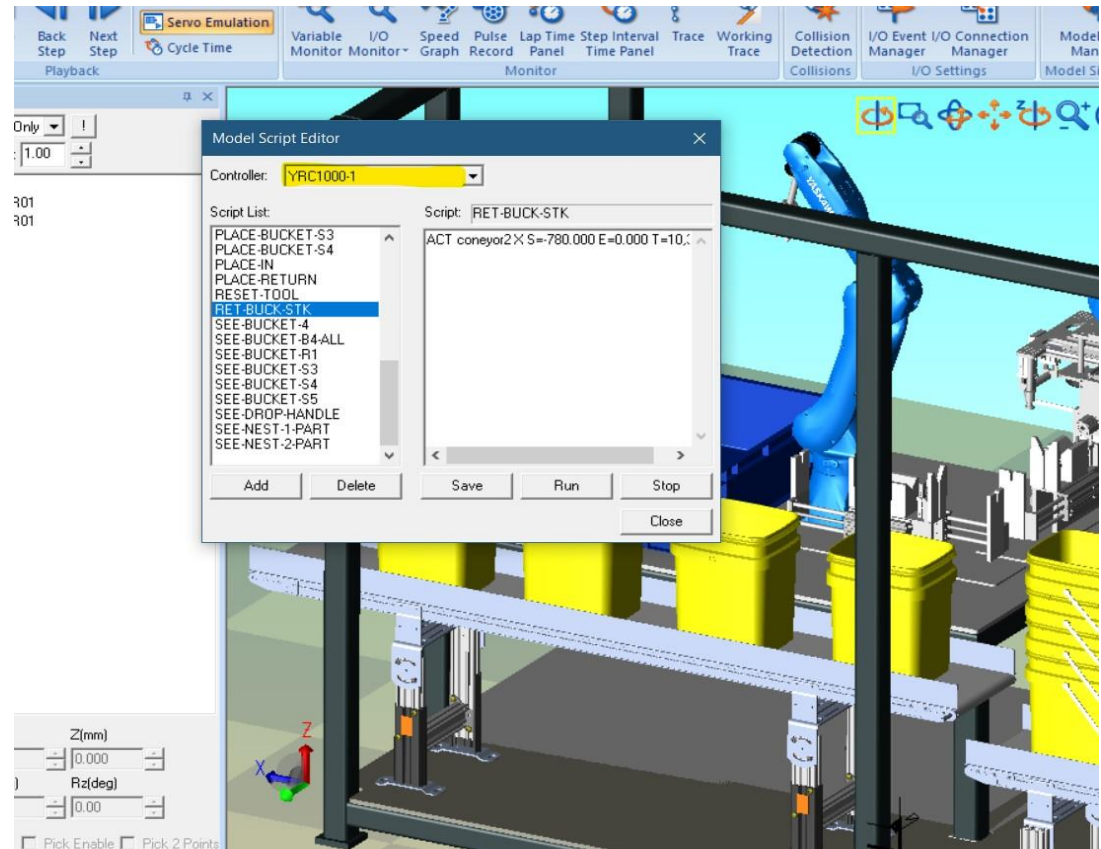


This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

Verify Active Controller

If you have more than 1 controller, you will need to make sure you have the controller active that the scripts will be used with.

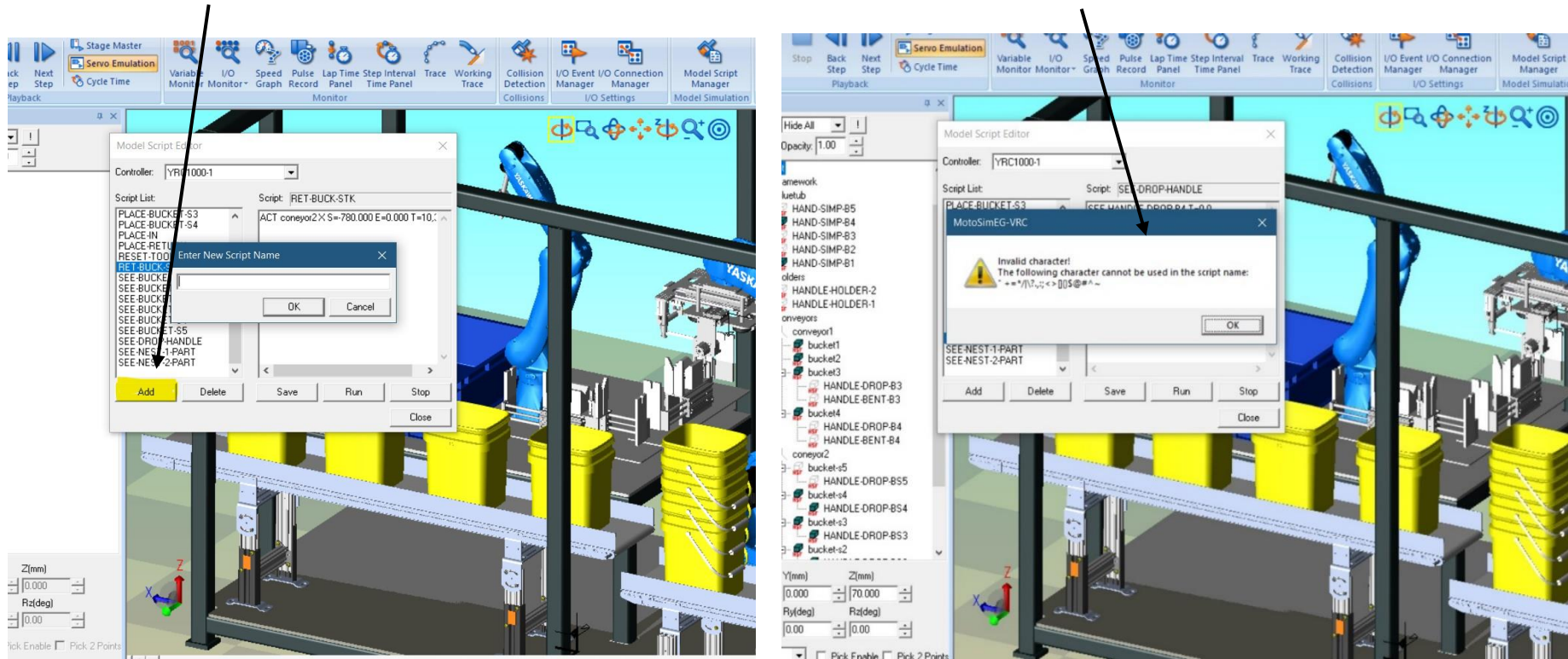
In the example here, you will see that the YRC1000-1 is active, which is controller 1 in this case.



This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

Next, click on the “Add” button, as highlighted. This will open a dialog box prompting you to enter a name for the script.

Use a descriptive name for the script. For example: “Gripper-open”. You cannot use spaces in the name. If you do, you will get the following error.



With the name Gripper-open, you might normally have two instructions: one to show the gripper open model and hide (HID) the gripper closed model; and one to hide the gripper open model and show the gripper closed model. (Your design may differ.) Gripper actions can then be scripted, making them look like the gripper opens smoothly.

This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

List of scripting commands:

SEE: Displays model listed in CAD tree

HID: Hides the model listed in CAD tree

MOV: Moves one model to another model in the CAD tree but does not relocate the model in the workspace.

AXIS6: Moves a named model from the CAD tree to a new location on the workspace

ADDX6: The position from its parent is changed by a value. IE: 10,20,30,0,0 in mm.

DUP: Duplicates a model. If Motosim is saved without deleting the copied model, then the description is written into the cell file. The copied model will reference the same model data file as the original when opened back up.

REF: Copies (references) the model M? to create model M? (M? is describing your model name) Even if Motosim is saved without deleting the copied model, the description (model data) is not written to the cell file.

DEL: Deletes the named model

ACT: Action. ACT M? D S=P1 E=P2 T=T1, T2 (ACT M? X S=0 E=1000 T=0,1000)

D: One of the directions X, Y, Z, Rx, Ry, or Rz

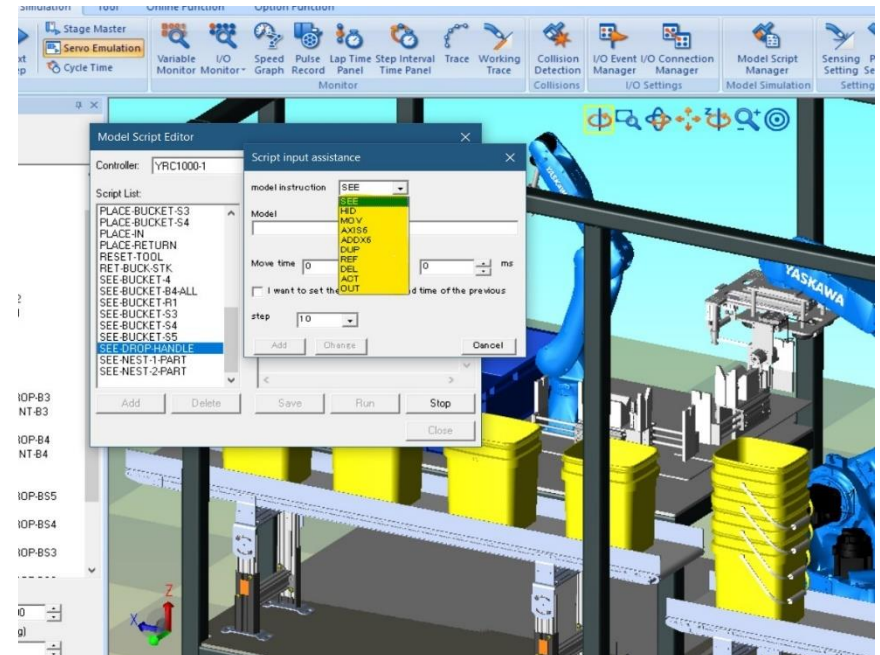
P1: Move start position (units: mm or deg.)

P2 Movement end position (units: mm or deg.)

T1 Movement starting time after script is executed (units in mm/sec)

T2 Movement ending time after script is executed (units in mm/sec)

OUT: can use 2 different types of output. The logical # like 20030=ON or General I/O signal like IN#1=ON



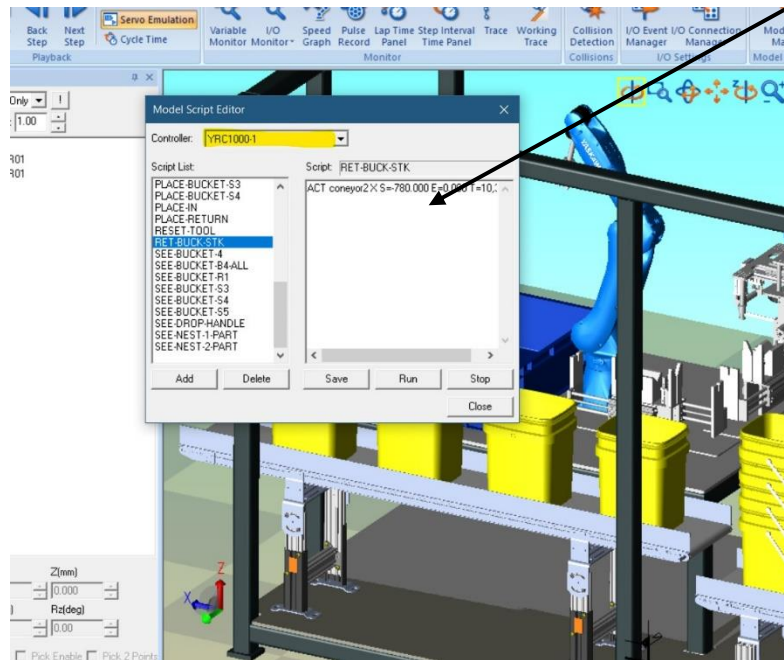
This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

NOTE: Only while the controller is actively running, will the ACT (action) command be executing. If the controller is stopped or the PLAY signal drops out prior to the ACT command to finish executing, it will stop with the controller, regardless if it has finished or not.

Assigning Script Commands

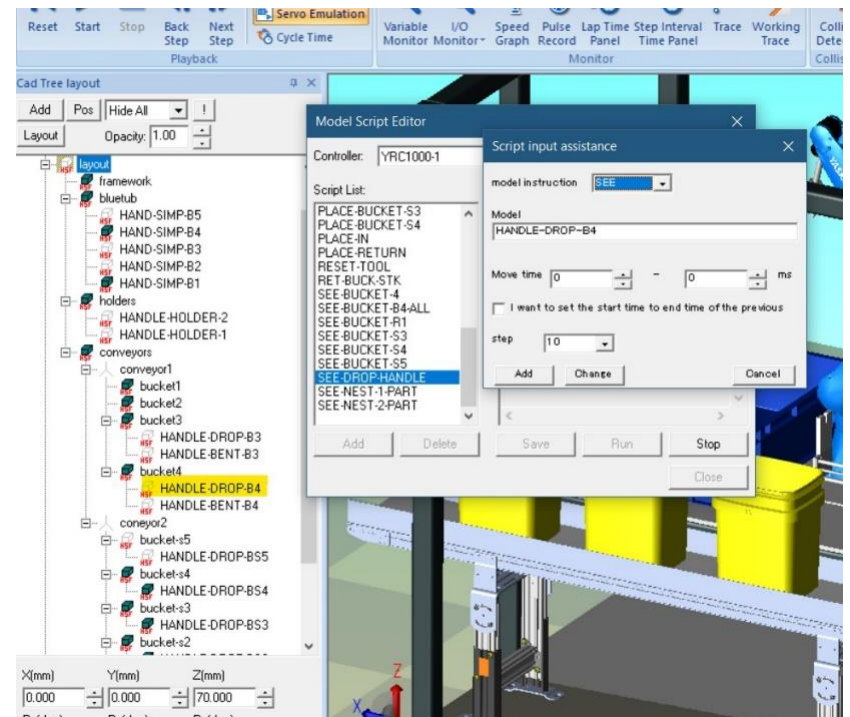
With the model script manager open, and a name assigned, you will double click in the right window

This will pull up the commands list to pick from as mentioned above. You must pick a command, before choosing the model to interact with from the CAD tree. If you pick the model first, then pick the action to be taken, the model will be unassigned.



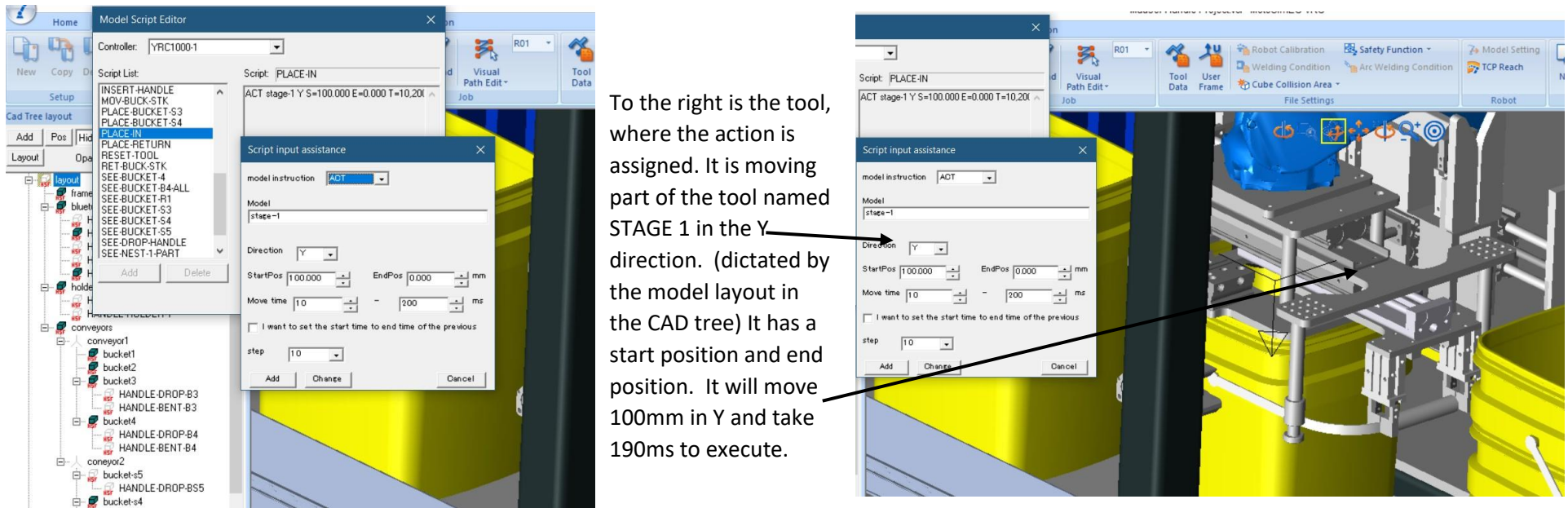
On the right, I am using the SEE command and assigning it to the model HANDLE-DROP-B4. The name of the script that will be used in the robot job is SEE-DROP-HANDLE.

This is used to see the handle models on the pails that are in the down position.



This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

In the following example, you can see the ACT script being used. It is one stage of this tool to place the handle on the pail.



To the right is the tool, where the action is assigned. It is moving part of the tool named STAGE 1 in the Y direction. (dictated by the model layout in the CAD tree) It has a start position and end position. It will move 100mm in Y and take 190ms to execute.

To help determine the direction the model you are trying to control will move, you can go to that model in the CAD tree and set position to determine what its X, Y or Z direction, as well as the distance you are wanting to move it. Then you can assign the direction and start and stop positions in the script manager for action.

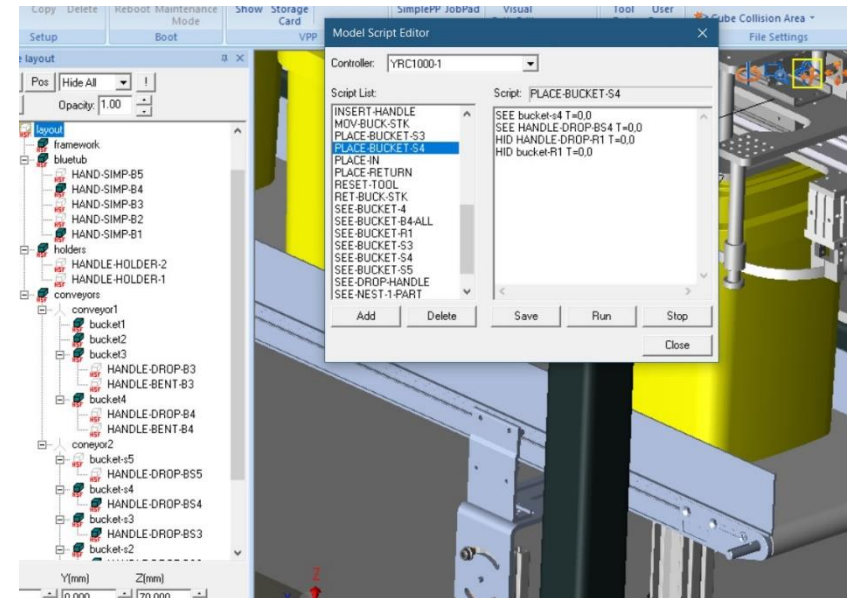
This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

Nesting Script Commands

You may use multiple commands under 1 script name. This is very useful to control multiple models with one instruction in the robot job.

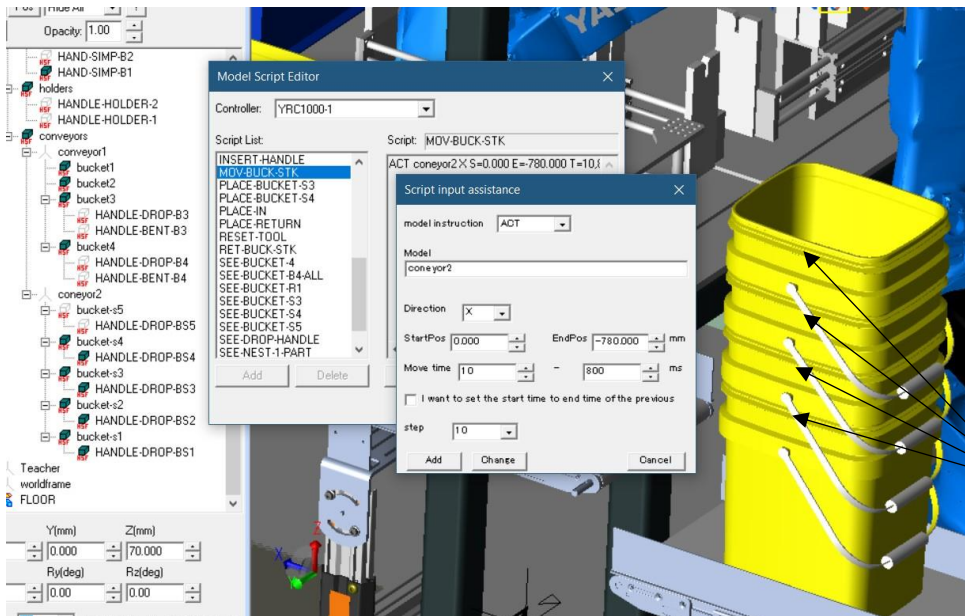
This script will be executed when the robot is placing the 4th bucket in the stack of buckets. In this example, the following will happen:

- Display the model bucket-s4 (4th. Bucket on stack)
- Display the model for the drop handle on bucket-s4
- Hide handle drop that is on R1 (bucket handle in tooling)
- Hide bucket that is in tool of R1 (bucket on robot tool)



This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

Using ACT to Move Models, Like Being on a Conveyor



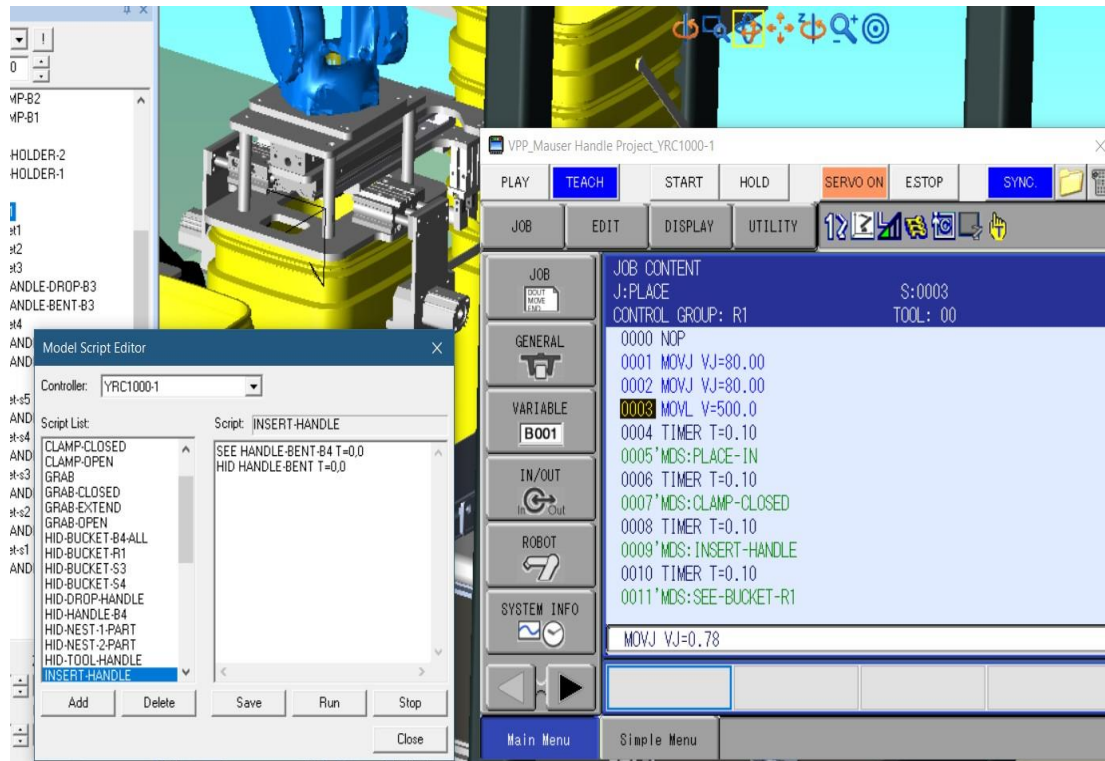
This is another ACT command, where it is moving a dummy model (conveyor2). Conveyor2 in the CAD tree has many other models under it, that make up all the buckets and handles in the stack. So, by doing an action of the dummy model, all child models under that parent move with it. It is starting at position 0 and moving -780mm in the X direction of that dummy model.

All these models, (buckets and handles), will move together.

This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

Inserting Scripts into Robot Jobs

Model scripts in robot jobs are entered as a comment. That way, if you take the job to a real robot, you do not get a syntax error.



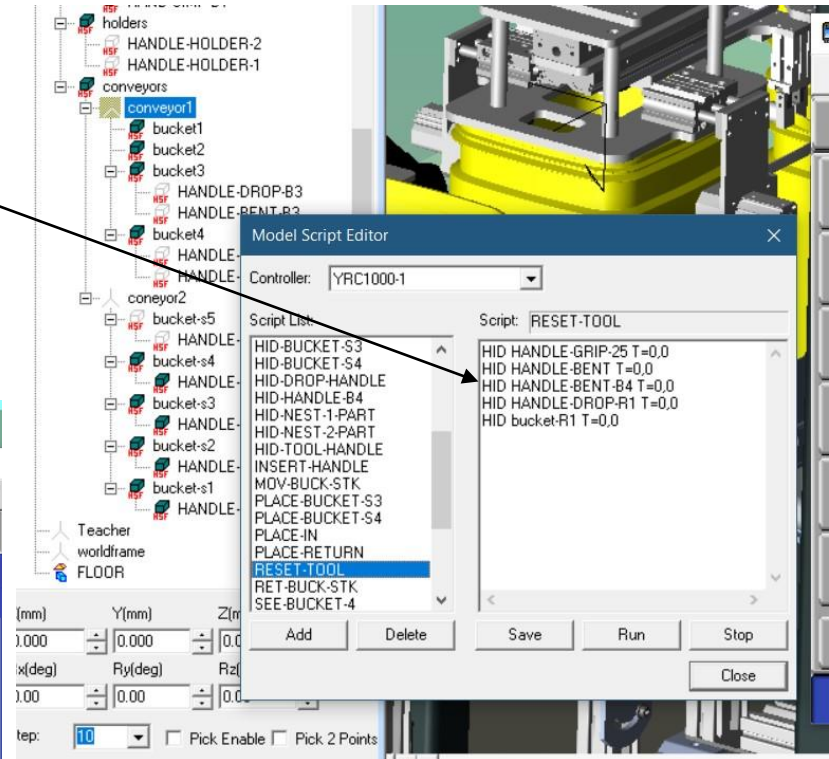
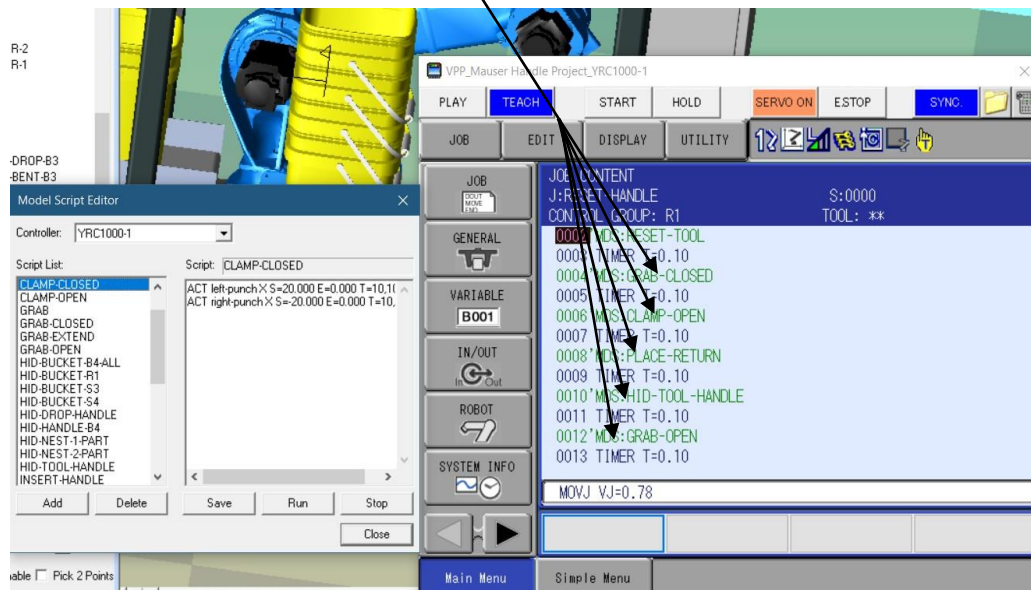
To insert a script into a robot job:

- Start programming your job. When you are at a location you want to preform a function, you will add a comment from the Inform list.
- Go to detail edit of that comment and then enter MDS:name (name of script).
- It is always a good idea to add a small timer (.01 sec. works) before and after the comment.
- If you have multiple scripts to run together, you only need a timer before and after all the scripts.
- In the case to the left, there is a timer between each script. This is because it is actuating the tool, and it must be sequenced. Completion of a script prior to running the next is important in this case.
- Without the timer being used with the scripts, the controller in most cases will run right through that comment without executing it.
- No errors will be displayed. The controller will just pass through the comment with no action and continue to run the job.

This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

It is also a good idea to have reset scripts. Those scripts will be assigned to set all models back to their starting positions or hide some and display others. Whatever you need at the start of the main job, so all sequences are correct. In the example to the right, I am resetting the tool models for what the tool is picking.

For the tool itself. That takes 5 scripts to reset the tool back to starting posture.





PARTNER SUPPORT

Shared Integration Experience

This document captures ideas, experiences, and informal recommendations from the Yaskawa Partner Support team. It is meant to augment – not supersede manuals or documentation from motoman.com. Please contact the Partner Support team at partnersupport@motoman.com for updates or clarification.

CONCLUSION:

Using scripts is a very powerful tool to help customers, or anyone to visualize what the process is doing. Great sales tool!