

## Note concerning the MotoROS with HC robots

Update: 2025-Feb-17

### **WARNING!!!**

*Even though the motion of a Human Collaborative robot using the MotoRos driver is possible with the current driver, IT DOES NOT MAKE THE ROBOT INTRINSICALLY SAFE for all operational conditions. Proper risk assessment must be conducted following the current industrial standards. Refer to the proper documentation for PFL settings and usage.*

### Compatibility Check

Yaskawa Human Collaborative robots (HC) make use of the PFL (Power and Force Limiting) function. For safety reasons, the use of MotoROS with Human Collaborative robots was prevented in early controller software versions until proper safety testing was performed. In those versions, the MotoROS driver generates a systematic alarm and is prevented from running.

The latest MotoROS driver checks controller software to ensure it is version YAS2.80 or newer on YRC1000. For the YRC1000micro, YBS2.30 or newer is required. The MotoROS driver is only prevented from running if the software is older than the required version.

### PFL Interaction

When the robot detects an unexpected contact (with a human or object), PFL causes the current motion to be cancelled and if clamping is detected the robot will move slightly in the opposite direction to release pressure. Further motion from ROS will not be accepted until the PFL sequence is completed and the ROS motion is restarted.

Following the activation of the PFL, further motion command sent by ROS will be rejected with a return value of ROS\_RESULT\_NOT\_READY (5). A new subcode ROS\_RESULT\_NOT\_READY\_PFL\_ACTIVE (5011) was added. Depending of the timing in the sequence, the return might be the new code or one of the existing subcode such as ROS\_RESULT\_NOT\_READY\_NOT\_STARTED or ROS\_RESULT\_NOT\_READY\_SKILLSEND.

### Tool Change Simple Message

The PFL function is highly dependent on proper tool definition and selection. In the case where the robot is handling parts, it may be required to change tool selection to properly reflect the current load on the robot. Until now the MotoROS driver was limited to the use of a single tool, Tool#0. To increase

flexibility, a new Yaskawa/Motoman specific simple message was added to select tool. **The implementation was done for the MotoROS driver side, but it is left to the ROS user to implement the necessary message on the ROS side.** Below are the new message specifications:

Message Type:

```
ROS_MSG_MOTO_SELECT_TOOL = 2018
```

Structure:

Standard header plus:

```
struct SmBodySelectTool
{
    // Robot/group ID; 0 = 1st robot
    int groupNo;

    // Tool no (0 to 63) for the selected group
    int tool;

    // Optional message tracking number that will be echoed back in the response.
    int sequence;
}
```

The message is sent to the MotionServer. The MotionServer will return a message of type:

```
ROS_MSG_MOTO_MOTION_REPLY = 2002
```

Return values can be:

```
ROS_RESULT_SUCCESS = 0
```

```
ROS_RESULT_INVALID = 3; ROS_RESULT_INVALID_GROUPNO = 3004
```

```
ROS_RESULT_INVALID = 3; ROS_RESULT_INVALID_DATA_TOOLNO = 3017
```

Once a tool is selected, the selected tool is used for all following motion until a new selection command is sent or the controller is rebooted. **Every time the controller is rebooted, the selected tool will default to 0 for all control group.** Only robot group can be assigned tools.

## FSU Interaction

The FSU (Functional Safety Unit) works independently from the controller, therefore all FSU functions will work properly whether MotoROS is used or not.

In the case that alarms are generated by the FSU, the ROS side will be notified of the alarm state like any other alarm. The alarm will need to be reset, and motion reinitialized from the current position.

**Since April 2023**, MotoROS is fully compatible with the Speed Limit functionality of the FSU.

In **old versions** of the MotoROS application, when a function such as Speed Limit is enabled, any motion sent by ROS at speed higher than the speed limit will be reduced. This will affect the robot motion and the robot will not reach it's expected end position. The only feedback to ROS is the standard position-monitor stream. There is no explicit feedback that the speed was dynamically reduced. This is a limitation that we are aware of. So it is not recommended to use FSU speed limitation with ROS. If you do need to use it, be careful to properly monitor feedback position to detect if the robot trajectory is not following the expected commanded trajectory.

## Conveyor Tracking

The HC series of robots is not compatible with standard Yaskawa Conveyor Tracking function. If Conveyor Tracking is enabled, then MotoROS will alarm and fail during initialization.