



## Release Notes for MP2000iec

### Firmware version 1.2.2.9

### Cumulative for changes from 1.1.3

Yaskawa Electric America, Inc.

#### Important!

If the controller has a firmware version less than 1.1.1, then it must first be upgraded to 1.1.1 before upgrading to a more recent version. This is due to a change in memory allocation that took place to expand the amount of memory available for application programs.

## 1. Important changes from 1.1.3 Release

### 1.1. New Features

- [SCR 5353] High Speed Outputs on LIO cards. LIO-01/02 and LIO-06 support coincidence output in hardware. This feature is exposed to the end user via the following new parameters:  
1050 coincidence enable  
  
1051 coincidence status  
  
1052 cyclic coincidence value  
1053 non-cyclic coincidence value
- [SCR 5418] External encoder feedback velocity sampling rate improvement. The controller now triggers the external encoder sampling in an interrupt service routine (ISR) rather than by hardware.
- [SCR 5595] Ease restriction on Master/Slave relationships. Changed restriction on master slave relationships to allow cascading master/slave relationships up to three levels. For example, the following situation is now allowed: Axis 1 is the master of Axis 2, and Axis 2 is the master of Axis 3 and Axis 4.
- [SCR 5696] Adjustments used for Y\_Cam Adjust blocks was different than MP940. Previously, the shift was determined by the equation:

$$y(t) = (f*t - \sin(f*t))/(2*PI), f=2*PI/End$$

This relationship results in smoother acceleration changes (i.e., limited change in jerk) but higher peak velocities. Now, the shift is determined by the equation:

$$y(t) = (\sin(f*t-PI/2) + 1)/2, f=PI/End$$

This relationship results in lower peak velocities but instantaneous changes in acceleration.

- [SCR 5728] Provide Scan Compensation Delay for Gear / Cam Masters. The controller now adds the current velocity multiplied by the feedback to the feedback position. As a result, parameters 1000, 1005 and 1006 will reflect the compensated value. Parameter 1305 is the feedback compensation, the units of which are in seconds, and the range is 0 to 10 times the motion update rate. To fetch the pre-compensated cyclic and non-cyclic values, use new parameters 1007 and 1008.

Additionally, when an external axis is used as a master, the controller adds master's velocity multiplied by the slave's command delay to the raw master position. The slave's command delay is a property of the slave axis is not exposed via a parameter.

- [SCR 5740] Add Parameters for reading LoadType and MachineCycle. Added parameter 1807 RotaryType with 0=Linear, 1=Rotary to MC\_ReadBoolParameter. Added parameter 1833 Machine Cycle to MC\_ReadParameter.
- [SCR 5761] Filter external encoder feedback velocity used for position compensation. The feedback velocity can be noisy due to the electro-mechanical characteristics (i.e., the axis is not running smoothly) or quantization. When the position is quantized due to the encoder resolution, the backward differences computation for velocity will amplify the quantization noise by dividing by the motion update rate. For example, if the position delta differs by one count, and the motion update rate 2ms, then the velocity estimate will differ by 500 counts/sec. This can be a problem because a noisy velocity estimate will cause the feedback delay compensation to introduce noise into the feedback position. To compensate for a noisy feedback velocity, an optional moving average filter was added. The moving average filter is set using parameter 1306, feedback velocity filter time constant. The units are in seconds; the maximum value is 0.1 seconds; and the default is 0.0, i.e., disabled. The filtered feedback velocity is observable with parameter 1001 and MC\_ReadActualVelocity, and the unfiltered feedback velocity is accessible via parameter 1009.

## 1.2. Function block fixes

- **Fixed:** [SCR 2810] A 95 issued when MC\_Power.Enable=FALSE while in motion
- **Fixed:** [SCR 3357] Can not re-execute non-periodic cams (Y\_CamIn.Periodic=FALSE)
- **Fixed:** [SCR 3461] CamScale for a one way cam

**Details:** Previously, the cumulative traveling distance was also scaled, resulting in potentially large changes in position. Now, the scaling is applied incrementally, and the current scale is used to increment the cumulative travel distance.

- **Fixed:** [SCR 3490] Downloading application while Y\_CamFileSelect is executing crashes controller  
**Details:** Function blocks that spawned background tasks, like Y\_CamFileSelect, did not properly protect against cleanup before the background task had finished.
- **Fixed:** [SCR 3586] Sigma-5 HBB interaction problem with MC\_Power (SCR 3498) / MC\_Power gets 4370, cant recover unless power cycle  
**Details:** Triggering HBB caused A95 command error and axis disabled unexpectedly alarm. After clearing these alarms, MC\_Power could not re-enable the axis.  
**Note:** Setting HBB on and off rapidly will still cause an issue with MC\_Power. This is scheduled to be fixed in the next release as SCR 5702.
- **Fixed:** [SCR 3504] Y\_VerifyParamters has InvalidParameter set when Matches is TRUE  
**Details:** InvalidParameter was always being set to the last parameter checked, regardless of whether the parameter matched or not.
- **Fixed:** [SCR 3565] MC\_Stop and MC\_TorqueControl  
**Details:** Previously, MC\_Stop ramped down both the torque and velocity limit, but as the torque ramped down the holding power of the axis was reduced, which might not be appropriate for a vertical axis. Now, the torque is held constant and the velocity limit is ramped down.
- **Fixed:** [SCR 3581] Parameter 1031 LatchPositionNonCyclic not reflecting position offset  
**Details:** When MC\_SetPosition redefined the position, parameter 1031 was incorrect.
- **Fixed:** [SCR 3611] MC\_StepRefPulse outputs CommandAborted when opposite direction limit is ON  
**Details:** Changed to return ErrorID 4370 when the initial move fails due to an overtravel limit, and ErrorID 4397 ("failed to move away from overtravel limit") if the move-back-to-C-pulse fails due to OT.
- **Fixed:** [SCR 3621] MC\_TouchProbe unmodularized latch position rolling over at 32 bit encoder limit.  
**Details:** When the encoder traveled past 2,147,483,647 counts, the un-modularized latch position, parameter 1031 went negative.
- **Fixed:** [SCR 3627] MC\_MoveAbsolute / MC\_MoveRelative get stuck with 4370 when limit switch is on.

**Details:** If an axis is controlled with MC\_MoveRelative or MC\_MoveAbsolute and hits a limit switch (and limit remains on, which is normal), a reverse move may not be able to be performed. The function block reported 4370 even though the motion was commanded away from the active limit switch.

- **Fixed:** [SCR 3646] Y\_ReadDriveParameter on missing axis causes CPU exception  
**Details:** Y\_ReadDriveParameter on a missing axis caused PLC Watchdog Alarm and CPU exception.
- **Fixed:** [SCR 3649] Reading non cyclic position causes Controller Alarm 3301 0018  
**Details:** Executing MC\_ReadParameter for parameter 1016 sometimes corrupted the cyclic conversion causing a one cycle jump in the command position.
- **Fixed:** [SCR 3703] MC\_AbortTrigger causing watchdog  
**Details:** MC\_AbortTrigger always took a semaphore locking the axis resource. However, this semaphore might already be owned by another task, causing MC\_AbortTrigger to block.
- **Fixed:** [SCR 3778] Moving Average Filtering causes motion glitch w/ Rotary Shortest Path  
**Details:** The moving average filter was applied after the rotary axis calculation causing the motor to jump in the opposite direction when the command position rolled over.
- **Fixed:** [SCR 3789] ErrorID 4398 occurs when Y\_CamShift uses Y\_AdjustMode#WithinRange and the master is not moving.  
**Details:** When the master was not moving, then the Y\_CamShift incorrectly calculated the within range window.
- **Fixed:** [SCR 3805] Ease reporting of "Axis Disabled Unexpectedly" alarm.  
**Details:** When the axis disables due to HBB or loss of main bus power, the axis will not post the "Axis Disabled Unexpectedly" alarm.
- **Fixed:** [SCR 3836] Y\_CamFileSelect gets stuck Busy if filename greater than 32 characters  
**Details:** If the entire path length (including the automatically pre-pended base directory) was > 32 characters, the string copy operation caused a buffer overwrite condition.
- **Fixed:** [SCR 3844] MP2000iec: Jerk in motion at very high accel/decels  
**Details:** The check to determine if the axis could reach the end conditions within one Mechatrolink scan did not account for the velocity constraint.
- **Fixed:** [SCR 3891] MC\_StepLimitSwitch.DistanceLimit not correct  
**Details:** The distance limit was computed as the cumulative amount travelled, not the distance from the starting position. Consequently, in some test cases that poorly mimicked a real hardware limit switch by programmatically toggling POT/NOT inputs,

MC\_StepLimitSwitch would fail after moving off the limit switch and then approaching limit switch, despite never being far from the starting position. To be more robust for testing and clearer to understand, the distance limit is now based on the distance from the starting position.

### 1.3. PLC

The following PLC issues were fixed in this release:

- **Added:** [SCR 2976] Force outputs zero (configurable) when PLC stops  
**Details:** Outputs can now be configured device by device to output zero when the PLC stops. See the Hardware Configuration.
- **Fixed:** [SCR 3648] Crash on reboot  
**Details:** Prior to rebooting, the controller requests the PLC to stop and immediately afterwards freed the memory used by the PLC. If an application was running, there was a chance that the application would try to use the freed memory.
- **Fixed:** [SCR 3702] Webserver causes watchdog when the axis parameters tab is clicked  
**Details:** Several function blocks, like MC\_TouchProbe, could block on the Axis resource semaphore, which might already be taken by the communications module to read drive parameters. These function blocks now check to see if the resource semaphore is available before blocking on it. If not available, then the function blocks report busy.
- **Fixed:** [SCR 3814] LIO-04 outputs require writing to 2nd word to enable output of 1st word

### 1.4. Mechatrolink

The following Mechatrolink issues were fixed in this release:

- **Added:** [SCR 3614] Software reset to clear alarms that require power cycle  
**Details:** Y\_ResetMechatorlink now performs a servo drive software reset, which adds the capability to clear alarms that require power cycling. This is a complete restart of the amplifier, and position values for incremental encoders will be reset.
- **Fixed:** [SCR 3616] Eliminate A.95 alarms when possible  
**Details:** A.95 alarms no longer occur if the controller disables the drive while moving.

### 1.5. EtherNet/IP and Modbus/TCP

The following Ethernet/IP and Modbus/TCP issues were fixed in this release.

- **Fixed:** [SCR 3764] Unique serial number and product name for EIP products  
**Details:** The Ethernet/IP specification requires unique serial numbers and a product name.
- **Fixed:** [SCR 3873] Ethernet/IP scanner driver accepts data only if adapter increments PDU seq #  
**Details:** The Ethernet/IP scanner ignored packets in when the PDU seq # did not increment.

### 1.6. RMI for Configuration Tool

The following RMI issues were fixed in this release:

- **Fixed:** [SCR 3626] Connection server stops responding after createCurrentParamFiles() on missing axis  
**Details:** If the Hardware Configuration tried to connect to the controller while there were servo axes initially detected at power up but no longer responding, an exception propagated to the upper levels stopping the connection server. This exception is now caught and displayed as an error message.

### 1.7. Known Issues

The following are the known issues for function blocks compiled from the test case failures and test applications.

- MC\_GearIn
  - [SCR 3908] Slave cannot synchronize to a master with S curve applied.  
**Details:** The S-Curve filter is applied after the slave has sampled the master's command position. If the master has an S-curve filter, but the slave does not, then the slave position will lead the master position. For example, if the master has a 0.01 second filter, the master was travelling at 10 rev/s and the gear ratio was 1:1, then the slave would lead the master by 0.05 rev.  
**Mitigation:** Do not use an S-Curve filter on any master axis unless the slave has an identical S-Curve filter.
  - [SCR 3925] Issues moving off POT/NOT with very small accelerations  
**Details:** If on an overtravel and the initial change in command position is less than the amount the feedback position is dithering, then the move might be aborted. The following example illustrates the issue:
    - 1) Prior to the move, the last feedback position is 0.0000
    - 2) Because of dithering, the current feedback position 0.0010.
    - 3) Because the accelerations are very small, the first command position of a move in the negative direction could be 0.0005 (i.e., -0.0005 relative to the starting position.)

4) The overtravel limit check compares the new position (0.0005) to the old position (0.0000) and erroneously rejects the move because it appears to be moving into the POT.

**Mitigation:** To avoid this situation, adjust the drives to reduce the encoder dithering and/or increase the move's acceleration.

- MC\_GearInPos
  - [SCR 3596] MC\_GearInPos slave may slow down to match master position  
**Details:** MC\_GearInPos attempts to engage as quickly as possible subject to the velocity and acceleration inputs. Consequently, the slave may slow down, or briefly move in the opposite direction in order to engage quicker, but the slave will be engaged by the specified position if permitted according to the acceleration and deceleration constraints.  
**Mitigation:** If this behavior is undesirable, then decrease the MasterStartDistance.
- MC\_GearOut
  - [SCR 2808] MC\_GearOut holds current velocity even if not gearing.  
**Details:** For example, executing MC\_GearOut while a MC\_MoveAbsolute function block is active will abort the MC\_MoveAbsolute function and hold the current velocity.  
**Mitigation:** Only execute MC\_GearOut when gearing.
- MC\_Power
  - [SCR 3542] ErrorID 4371 on MC\_Power for drive power issue does not go off when enable goes low.  
**Details:** When MC\_Power can not enable the drive because, for example, it's disconnected from the network, the MC\_Power.ErrorID is set to 4371. When MC\_Power.Enable is set to false, this error code is not cleared. This alarm will be cleared when MC\_Power succeeds in enabling the drive.  
**Mitigation:**
  - [SCR 5717] MC\_Power gets errorID 57617 after download changes  
**Details:** Error ID 57617 can occur after download changes on PLCopenPlus functions blocks.  
**Mitigation:** Do not use download changes. Use patch POU instead.
- MC\_ReadAxisError
  - [SCR 2792] Alarm does not match alarm shown on drive  
**Mitigation:** The drive may have multiple alarms, and one of these is returned by MC\_ReadAxisError. Typically, the alarm matches the one shown on the drive, but this is not guaranteed.
- MC\_ReadStatus (Axis State Machine):
  - [SCR 2567] Incorrect axis state with MC\_MoveSuperImposed.  
**Mitigation:** Executing another motion block fixes the axis state.
  - [SCR 2822] No transition from ErrorStop to Disabled when MC\_Power.Enable=False.  
**Mitigation:** Technically this is not part of the PLCopen specification; the specification does not indicate any transitions to Disabled state.

- [SCR 3450] No transition from Disabled to ErrorStop when MC\_Power.Enable=True and there is an error on the axis.  
**Mitigation:** Use MC\_ReadAxisError to determine when the axis has an error.
- MC\_Reset
  - [SCR 2729] MC\_Reset does not clear A.ED on Sigma II
  - **Details:** A.ED alarm requires the servo network to be reset.
- MC\_StepRefPulse & MC\_StepLimitSwitch
  - [SCR 3131] MC\_StepLimitSwitch only supports one LimitSwitchMode: MC\_EdgeOn  
**Details:** MC\_StepLimitSwitch only works when detecting the rising edge of an input.  
**Mitigation:** Application can be wired to work within this limitation. The servopacks account for active high / active low via Pn 50A and 50B.
- MC\_TorqueControl
  - [SCR 3051] MC\_TorqueControl requires MC\_Stop before using any other motion function block.  
**Details:** MC\_TorqueControl can not be aborted by a 'position mode' motion block such as MC\_MoveAbsolute.  
**Mitigation:** Changing control modes while moving is not supported.
- Y\_CamIn
  - [SCR 3356] Y\_CamIn does not detect if the engage window is too small.  
**Details:** If the window is too small, Y\_CamIn will not engage. This will be based on the master speed.
  - [SCR 3413] Cam slaves engage one scan after detection of engage position  
**Details:** When the master crosses the engage position, the controller prepares the axis for camming, but the actual cam processing does not occur until one Mechatrolink cycle later, when all the axes are processed at the same time.
  - [SCR 3658] Two point, one-way cam tables do not work.  
**Details:** For one-way cam tables, the end condition occurs when the master position changes from the last defined interval to the first interval, but if there is only one interval, then this end condition is never detected.  
**Mitigation:** This issue can be avoided by using cam tables with four points or more.
- Y\_CamOut
  - [SCR 3671] Y\_CamOut.DisengageData.EndMode=Immediate is not supported.  
**Details:** This disengage mode is not supported and will result in ErrorID ???? – unsupported disengage mode.  
**Mitigation:** The user can implement this same behavior by using the current master position as disengage position..



## 1.8. Unsupported Function Block Inputs and Outputs

The following function block inputs and outputs are not supported and are reserved for future use:

- MC\_MoveAbsolute.Jerk
- MC\_MoveRelative.Jerk
- MC\_MoveSuperImposed.Jerk
- MC\_MoveVelocity.Jerk
- MC\_Stop.Jerk
- MC\_Stop.BufferMode (assumed BufferMode is *aborting*)
- MC\_Power.BufferMode
- MC\_ReadStatus.Busy (always FALSE)
- MC\_ReadAxisError.Busy (always FALSE)
- MC\_Read[Bool]Parameter.Busy (always FALSE)
- MC\_TorqueControl.Direction
- MC\_TorqueControl.Acceleration
- MC\_TorqueControl.Deceleration
- MC\_TorqueControl.Jerk
- MC\_Write[Bool]Parameter.Busy (always FALSE)
- MC\_ReadActualPosition (always FALSE)
- MC\_GearIn.Jerk
- MC\_TouchProbe.WindowOnly
- MC\_TouchProbe.FirstPosition
- MC\_TouchProbe.LastPosition
- MC\_ReadActualVelocity.Busy (always FALSE)
- MC\_ReadActualTorque.Busy (always FALSE)
- MC\_GearInPos.Jerk

## 1.9. Modbus/TCP

The following are the known issues for Modbus/TCP:

- [SCR 2739] Function code 15: write multiple coils is not supported  
**Details:** Write multiple coils is not supported, so each coil has to be written using a separate transaction. As a result, writing multiple coils is not recommended.
- [SCR 3897] Modbus/TCP server seems to stop communicating  
**Details:** A VB application running on a PC as the master can overload the controller and break Modbus/TCP communication.  
**Mitigation:** Adding a 5 ms timer between R/W queries avoids the issue.

## 1.10. Web Interface

The following are the known issues for the Web Interface:

- Applet edit fields don't work reliably with IE8 (3829)

**Details:** Clicking around the grids results in odd highlighting and focus behaviors which hinders editing.

### 1.11. Output Variables

- [SCR 3475] Outputs can not be retained  
**Details:** In the IDE, the Global Variables table contains a column marked “Retain.” Selecting a check box within this column causes the corresponding variable to be allocated in SRAM. However, for outputs, this feature is not working. At this point, it is not clear if the problem is with the KWSOft’s MultiProg IDE or ProConOS firmware module. KWSOft has been notified of this issue, and they are tracking it with issue TID#5366.  
**Mitigation:** This issue is fairly easy to workaround by first storing the output value in a global variable that is retained and then copying the global variable to the output.

### 1.12. External encoders

The following are the known for external encoder axes:

- [SCR 3629, SCR 3909] External encoder feedback shows variability of sampling rate  
**Details:** .The controller triggers the external encoder sampling in an interrupt service routine (ISR) rather than by hardware. Consequently, the variability might cause a 1% variability in feedback velocity.

## 2. Known issues specific to MP2310iec and MP2300S

### 2.1. Function Blocks

- MC\_Reset
  - [SCR 3766] MC\_Reset does not recovery correctly from controller watchdog alarm  
**Details:** MC\_Reset behavior is different for clearing controller watchdog alarm for two different cases. 1) Pulling mlink cable and seeing A.E6 alarm on drive requires one MC\_Reset. 2) Power cycling the drive and then attempting MC\_Reset requires multiple MC\_Reset. On the first MC\_Reset, the Done bit is asserted immediately. On the second MC\_Reset the Done bit is asserted after a couple of seconds as expected and controller watchdog alarm is cleared.  
**Mitigation:** For now, the workaround is to have multiple MC\_Resets when a new axis is added.

## 2.2. Mechatrolink

- [SCR 2919] Controller reboots if gearing 16 axes with a 2ms Mechatrolink update rate  
**Details:** MC\_GearIn.Execute=TRUE for all axes, the controller reboots.  
**Mitigation:** Increase the Mechatrolink update rate. See Section 5.2 (MECHATROLINK Update Rate) for suggested Mechatrolink update rates.
- [SCR 3083] A94B alarm generated after Relative, Absolute, or Geared move.  
**Details:** With SGD V drives previously tuned with Sigma Win +, executing MC\_MoveRelative, MC\_MoveAbsolute or MC\_GearIn(Pos) at roughly half the rated speed causes an A94B warning. This is caused by “model following control” being enable in Pn140.  
**Mitigation:** Set SGD V drive Pn140 to the default value of 0x0100.

## 3. Known issues specific to MP2600iec

### 3.1. PLC

- [SCR 3623] PLC functions do not support EN/ENO inputs  
**Details:** On the MP2310iec and MP2300Siec, function could optionally have EN input and ENO output. On the MP2600iec, this option is not available.  
**Mitigation:** The user can duplicate the EN/ENO functionality by wrapping the functions with their own functions.
- [SCR 3732] CPU Utilization is wrong.  
**Details:** The CPU utilization always reports 0.1%.  
**Mitigation:** The individual task statistics are reported in microseconds, which is more useful for determining watchdog timers.
- [SCR 3927] eCLR task priorities are not correctly mapped.  
**Details:** PLC Task priorities 0, 1, 2, and 3 are all mapped to vxWorks task priority 42. Priorities 4-15 are then mapped linearly to vxWorks priorities 43 – 54.  
**Mitigation:** The user can adjust their priority assignments to account for this issue.
- [SCR 3931] CPU overload does not generate a controller alarm.  
**Details:** The PLC stops as expected and properly disables all axes, but no controller alarms are posted.  
**Mitigation:** The user can use event tasks to catch this condition.
- [SCR 3914] MP2600 1.2.1.5 WDT alarm and bad CRC in alarm history after restart.  
**Details:** To reboot, the controller sends a software reset command to the drive. Since the drive is rebooting, it does not acknowledge the command.  
**Mitigation:** Ignore these alarms in the alarm history.