PerformanceMotionDevices
+1 781 674 9860
engineering@pmdcorp.com

Release Notes
MC24x0 version 3.3

Document last updated: 3/30/2007

Product names: MC2440, MC2420, MC2410
Version control label: mc24x0 version 3.3
IO device: MC2000ION
Date of build: 3/7/2007

Description:
The MC24x0 is a motion control processor for stepper motors and provides one to four axes of motion. This document details bug fixes and changes for this release.

Known Issues:

If the IO chip HostRdy signal (pin 8) is used for chip busy detection, the first instruction sent to the chipset after a power on or reset may be ignored or may produce a checksum error. It is recommended that in this configuration a NoOperation command be sent to the chipset as the first instruction after a power on or reset. If the *ReadStatus* operation is used to check the HostRdy state this problem does not occur.

Changes/Fixes:

Command Changes

Fixed - SrlEnable signal was intermittently staying active after an Update command.

Profile Changes

Fixed a problem where a negative actual position was resulting in positive capture values when a non-unity encoder to step ratio was set.

Version 3.2

Known Issues:

If the IO chip HostRdy signal (pin 8) is used for chip busy detection, the first instruction sent to the chipset after a power on or reset may be ignored or may produce a checksum error. It is recommended that in this configuration a NoOperation command be sent to the chipset as the first instruction after a power on or reset. If the *ReadStatus* operation is used to check the HostRdy state this problem does not occur.
To determine if the motion processor has generated an interrupt to the host, the host should check the state of the host interrupt signal. In most situations this signal should be connected to an external interrupt input on the host processor. The state of this signal is also reported by the *ReadStatus* operation but it is only accurate when the host is issuing actual chip commands and not simply polling *ReadStatus*.
A command sent via the serial interface that results in an instruction error does not set the instruction error bit in the event status register. The error code is set in the status byte of the serial response packet.

Changes/Fixes:

Command Changes

Fixed a 1st generation command incompatibility. SET_LMT_SENSE now behaves as expected. Previously the parameter for this command was inverted in comparison to its expected behavior.
Fixed a problem where an Update command that caused a command error would return a bad serial checksum.
Fixed a problem where zero parameter commands could return a bad checksum.
Fixed a problem that resulted in serial communications locking up in the event that a command packet was sent by the host with additional (unnecessary) bytes.
The parameter range for SetTrackingWindow was increased from 0 to 2^{15}-1, to 0 to 2^{16}-1.

Profile Changes

Fixed a problem with S-Curve when certain move settings would cause a “Fractional Period”>1, which would cause the move to fail (excessive velocity and/or overshoot).
Corrected a problem in velocity contouring profile mode related to very large velocity values. Prior to this fix, if the velocity was greater than.half of full-scale the velocity was interpreted as being negative instead of positive.
Corrected problems in s-curve profile mode when velocity or acceleration are set close to their maximum values.
Corrected a problem in s-curve profile mode related to move length. Previously, a move greater in length than half of full-scale position could only be done in the forward direction. Moves of greater than half of full-scale are now allowed in both directions.
Corrected problems in trapezoidal profile mode when velocity or acceleration are set close to their maximum values.
Corrected a problem where an axis in electronic gear profile mode could not be stopped using the SetStopMode command or a breakpoint with its action set to AbruptStop.
Corrected a problem in velocity contouring profile mode related to limit switch handling. Prior to this fix, if a limit switch had been activated an attempt to move out of the limit switch would generate an error.
Corrected a problem in velocity contouring profile mode related to a change in the motor mode. Prior to this fix, if the motor mode was turned off while and axis was in motion, the axis would resume motion when the motor mode was turned on. The axis will now remain stationary after the motor mode is turned on until a SetVelocity command has been issued with a value greater than zero.
Corrected a problem in trapezoidal profile mode where a deceleration value of 1 could cause the profile to overshoot the destination position. Following the overshoot the profile would change direction and correctly complete at the destination position.
Corrected a problem in trapezoidal profile mode related to changing direction on the fly. Prior to this fix, if the start velocity was non-zero and a change in the destination position on the fly would require a change in direction the profile would not change direction.

Registers and Signals Changes

The InMotion operation was changed to more accurately reflect the real state of motion. Now, after an Update command that will start motion the InMotion bit is immediately set. Previously the InMotion bit would not have been set for up to one chip cycle. This change corrected the situation where the MotionComplete bit would not be set for moves less than one step.
Corrected a problem in the activity status register related to the limit switches. Prior to this fix if both the positive and negative limit switches were active both switches would have to go inactive before either limit bit would be cleared. Now both bits are completely independent in operation.

Miscellaneous Changes

The buffered velocity is now zeroed when a motion error occurs. In previous versions this value was not cleared and so an Update after a motion error could result in unexpected motion.

Version 3.1

Changes/Fixes:

Command Changes

Fixed a 1st generation command incompatibility. SET_MTN_CMPLT_BRK now behaves as expected. Previously the command did not work.

Profile Changes

Corrected a problem in the SCurve profile that resulted in an overshoot in phase 1 of the profile for extremely short moves. This would in turn create an error in the trajectory in Phase 5/6, which caused the trajectory to overshoot but settle at the correct destination.
Corrected a problem in the SCurve profile that would cause the trajectory to automatically restart if a ClearPositionError command was given after a move in the negative direction trajectory completed.
Corrected a problem in velocity contouring profile mode that could cause the motion complete bit to not be set if the axis profile mode was changed to velocity contouring on-the-fly from either SCurve or Trapezoidal profile mode.

Miscellaneous Changes

Corrected a problem with analog inputs where the default conversion timing resulted in out of specification results being generated by the analog to digital converter.