PerformanceMotionDevices
+1 781 674 9860
engineering@pmdcorp.com

Release Notes
MC55xx0 version 3.1

Document last updated: 9/6/2013

Product names: MC55420, MC55320, MC55220, MC55120, MC55110

Date of build: 08/23/2013

Device Checksum:

55110031	0xFD35873F
55120031	0xFCEF7CB8
55220031	0xFCD53517
55320031	0xFCB78907
55420031	0xFC9A9744

Compatible IO devices: MC50000IOAD8 (recommended) or MC50000IOABN

Description:
The MC55xx0 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. For details of bug fixes prior to 2.0 refer to the release notes for version 1.5.

This release includes support for 32 bit parallel encoders. The encoder data will be read on the parallel bus as two separate 16-bit words, high word first. *It is the responsibility of the hardware providing encoder data to ensure that the high and low words are consistent.* The low word is read immediately after the high word, within 200 ns.

32 bit parallel encoder addresses are as follows:

Axis 1	high	0x810
	low	0x811
Axis 2	high	0x812
	low	0x813
Axis 3	high	0x814
	low	0x815
Axis 4	high	0x816
	low	0x817

The maximum supported change in encoder reading is 32767 (positive or negative) in a single commutation period of 102 μs.

Known Bugs:

None

Known Issues

MC5xx20 only: If the IO chip HostRdy signal (pin 8) is used for chip busy detection, the first instruction sent to the chipset after a device 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.
When the encoder source for pulse and direction is set to loopback and the direction signal is inverted then the "A" encoder signal must also be inverted or the commanded position will change in the opposite direction to the "actual" position.
When using the parallel communication channel it is possible to send enough commands to Magellan to interfere with profile calculation, resulting in trajectory errors. It is recommended to delay sending commands so that at least 500 microseconds is allowed for each.

Command Changes:

SetEncoderSource	Now accepts an argument of 6 to denote "32 bit parallel".
SetTraceVariable	Trace variable 83 has been added: actual velocity in 32.0 format. Trace variable 6, actual velocity in 16.16 format, is still supported.
SetTraceVariable	Trace variable 84 has been added: raw parallel encoder reading. This value will differ from the actual position if SetActualPosition or AdjustActualPosition has been used.

Incompatibilities with previous version:

The behavior of the Activity status positive limit and negative limit bits is changed. In previous versions these bits were set only when the corresponding limit event was raised, and cleared only when the corresponding limit signal became inactive. This behavior was not well documented, and led to problems such as bugs 235 and 337, below. In the current release, the Activity status limit bits are set if the corresponding limit signal is active and there is a nonzero event action for the corresponding limit event.

Changes/Fixes:

179	In some cases, when the negative limit switch was active, a negative limit event was raised when moving in a positive direction using the s-curve profile. This is fixed.
232	When the stop and clear position error breakpoint action was configured nothing was actually done when the breakpoint triggered. This feature now works.
333	For step motors, the AdjustActualPosition command sometimes caused a jump in the motor phase angle, and a consequent jump in the physical rotor position. This bug is fixed.
335	An update causing a move in the direction of a tripped limit

	switch did not result in the “invalid move into limit” error unless a limit event had previously been raised. Limit error checking is now correctly done.
337	Setting the limit event action to zero would not prevent an “invalid move into limit” update error if a limit event had previously been raised. Limit error checking is now correctly done.
339	When the OutputMode0 pin was grounded and Magellan reset, the motor on Activity status bit was set after reset even though the operating mode was 1 (output disabled). OutputMode0 is used to select an Atlas-compatible set of reset defaults. This bug is fixed.
345	In some cases the commanded acceleration trace in trapezoidal profile mode showed some small but spurious oscillations. This is fixed by properly rounding the internal value.

Version 3.0:

Known Issues:

MC5xx20 only: If the IO chip HostRdy signal (pin 8) is used for chip busy detection, the first instruction sent to the chipset after a device 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.
When the encoder source for pulse and direction is set to loopback and the direction signal is inverted then the "A" encoder signal must also be inverted or the commanded position will change in the opposite direction to the "actual" position.
When using the parallel communication channel it is possible to send enough commands to Magellan to interfere with profile calculation, resulting in trajectory errors. It is recommended to delay sending commands so that at least 1 ms is allowed for each.

Pinout Changes:

pin 7, OutputMode0	This pin, formerly designated NC, is now used to enable non-backwards compatible changes in behavior. If this pin is grounded when Magellan comes out of reset the new mode will be selected; if it is not connected then a backwards-compatible mode will be selected.
pin 33, AxisOut1	This pin, formerly designated NC, now always carries the Axis 1 AxisOut signal.
pin 32 NC/AxisOut1	This pin, formerly designated AxisOut1, carries the AxisOut1 signal only if OutputMode0 is left floating.

Bug Fixes:

309	After a trace trigger the start of tracing was delayed by the trace period. In the current release tracing begins immediately.
125	Actual position breakpoints sometimes triggered incorrectly when the actual position was near the wrap around value, this is fixed.
153	When in "actual position" motion complete mode the motion complete event was raised each time an axis transitioned from unsettled to settled, even without a new move. In the current release motion complete is signaled only once per move.
157	The AdjustActualPosition command sometimes corrupted the actual position, this is now fixed.
176	A reset command using multi-drop serial communications could result in a spurious checksum error. This is fixed.

329	Under certain circumstances pulse and direction step output was, rarely, incorrect. This is fixed.

Incompatibilities with previous version:

Trace start timing is changed, see bug 309.
Motion complete events are no longer signaled multiple times in “actual position” complete mode, see bug 153.
When OutputMode0 is grounded, the default operating mode for all motor types is 1.

Version 2.4

Known Issues:

MC5xx20 only: If the IO chip HostRdy signal (pin 8) is used for chip busy detection, the first instruction sent to the chipset after a device 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.
When the encoder source for pulse and direction is set to loopback and the direction signal is inverted then the “A” encoder signal must also be inverted or the commanded position will change in the opposite direction to the “actual” position.

Incompatibilities with previous version:

Commanded positions produced by the external trajectory generator will be slightly different – in all cases they should be slightly more accurate than in previous versions.
Conversion of steps into actual position and vice versa will be slightly different – in all cases they should be slightly more accurate than in previous versions.

Known Bugs:

None	

Changes/Fixes:

90179	Unexpected commanded position values when using external profile mode, particularly with small negative velocities. Trajectory position, velocity, and acceleration are computed internally to 48 bits, however only 32 bits are used for motor control or user-visible. This version uses more accurately rounded position values than previous versions.
90180	Commanded position incorrectly rounded when using a non-unity step to count ratio.
90181	Commanded position wrong when using a non-unity step to count ratio and a negative actual position.

90182	Non-unity step to count ratio sometimes not applied to actual position after a commanded change in the ratio.

Versions 2.2 and 2.3

Not released

Version 2.1

Known Issues:

MC5xx20 only: If the IO chip HostRdy signal (pin 8) is used for chip busy detection, the first instruction sent to the chipset after a device 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.

Incompatibilities with previous version:

If the CP bus is tied high (or left floating – which is the same) the default serial port baud was 9600 in 1.x and 2.0. It is now 57600.

Known Bugs:

90166	When the encoder source for P&D is set to loopback and the MotorOutput bit in the signal sense is active, the actual position goes in the opposite direction of the commanded position.

Changes/Fixes:

90170	Default serial port baud was 9600 instead of 57600 as documented for 2.0. Fixed.
90171	MultiUpdate was only affecting one axis. Fixed.

Version 2.0

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 device 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.

Incompatibilities with previous version:

AxisOut	In version 1.x of Magellan, when the axis out register is set to SignalStatus the output sense of AxisOut was inverted to indicate the active LOW sense of the signal status register. This has been maintained when the SetAxisOutSource command is used, but when the new API command SetAxisOutMask is used the user determines the sense of the output.
Default CAN configuration	If the CP bus is tied high (or left floating – which is the same) the default CAN configuration on Magellan 1.x was 500K, NodeID=0. For Magellan 2.x it is 20K, NodeID=0.
Chip response to the reset command	For compatibility with ION and future products, when the reset command is executed it now does not send the "Processor reset" error code in the response. Instead, it responds "No error" and sets the InstructionError to "Processor reset". Following the reset the host can execute GetInstructionError to verify that the chip reset correctly and also to clear the instruction error register.

Known Bugs:

None

Changes/Fixes:

Command Changes

90165	Performing a SetMotorMode On would cause the internal step position counter to be filled with an incorrect value. This problem would not result in any loss of steps or observable problems. Fixed.

Communication Changes

90163	When using parallel communication, a bad axis error would not be returned for invalid axis selection. Fixed.
90160	When using parallel communication, if a command is sent with an invalid axis value, if the host then attempts to read data it would receive data from the previous command. Fixed.
90155	In multi-drop serial mode the chip would not automatically re-sync each time it received the address byte of a new frame. Fixed.
90147	In CAN communication an invalid axis error code was returned in the high byte of the returned status word instead of in the low byte. Fixed.

Trajectory Generation Changes

90151	If the SetEncoderToStepRatio command was issued during a trapezoidal move with 1 count (or step) or less left to the destination position, the chip could stop responding. Fixed.

Step Signal Output Changes

	none

Commutation Changes

	none

Registers and Signals Changes

90162	The AxisOut1 or AxisOut2 signals could exhibit glitches. Fixed.

Position Feedback Changes

90164	Selecting the parallel encoder as source does not store the first value read to the actual position. Fixed.
90152	MC55110 only. A position capture will store an incorrect value if encoder channel A has been inverted using SetSignalSense. Fixed.

Miscellaneous Changes

	none