Firmware Release Notes: Class 6 – EIP/EEC/EPN

Moog Animatics Firmware Update Revision History

Release Date	Software Version	Changes	Functions and items affected
3/28/2019	6.0.2.43	• Updated vendor specific objects to return error codes • Support for Combitronic added, requires FW850_523_4_0_5.nxf or newer. See website for download. -EIP models only. • Support for "keep alive" feature added, requires FW850_523_4_0_5.nxf or newer. See website for download. -EIP models only.	• 0x71, 0x72
10/29/2018	6.0.2.37	• Increased robustness of Combitronic commands • TCP channel number is 7 • BAUD(1) removed, no second com port	• PRINT7, RCHN7, SILENT7…
9/10/2018	6.0.2.36	• General bug fixes	
8/15/2018	6.0.2.35	• Additional range checking • Disabled SNAME on non –EPN models • Combitronic support over UDP/Ethernet on –EIP models • Added support for outputs 4/5 (MT2 models only) • Enable/Disable Modbus/TCP and Ethernet serial encapsulation	• INA, IN, X(), S(), PRINT(), KG, KL, PML, all synchronized motion commands • SNAME • ETHCTL(101,x), ETHCTL(100,x)
10/17/2017	6.0.2.34	• General bug fixes	
3/21/2017	6.0.2.33	• Out 9 (not faulted) is user configurable as general purpose output	• EOFT

		• EIP support for EOBK and EOFT	• Ethernet/IP Object 0x71
2/27/2017	6.0.2.32	• Prevent limits triggering in CSP, CSV, CST modes when output speed/torque is zero • Encrypted program download through CANopen/EtherCAT	 • CANopen/EtherCAT object 0x2500
2/9/2017	6.0.2.31	• EtherCAT object 0x60FE made more robust • SD card support added (if hardware available) • Allow user to invert home polarity	• Object 0x60FE • ETHCTL(60,x)
5/4/2016	6.0.2.30	• EtherCAT time sync enabled • EtherCAT object 0x6502 indicates homing is supported • EtherCAT object 0x6060 made more robust on transition to CSP, CSV, CST modes	 • EtherCAT Object 0x6502 • EtherCAT Object 0x6060
4/29/2016	6.0.2.29	• EtherCAT time sync disabled	

5/2/2016	6.0.2.28	• Use the EIP_APS_SET_CONFIGURATION_PARAMETERS_REQ message to allow full Type 1 Reset Processing. The previous command (EIP_APS_SET_CONFIGURATION_REQ) did not have all of the data values needed on reset (MCAST, TTL, etc.)
		• Add code to support the following remanent variables. They are stored in EE when changed, are loaded at power up and reset to default values on an EIP Type 1 Reset
		   • bTTLValue
		   • usEncapInactivityTimer
		   • tMcastConfig.bAllocControl
		   • tMcastConfig.bReserved
		   • tMcastConfig.ulMcastStartAddr
		   • tMcastConfig.usNumMCast
			• Passed ODVA Conformance Testing (CT13)

3/21/2016	6.0.2.27	• Increased allowable number of Modbus/TCP connections from 1 to 3.	•
3/2/2016	6.0.2.26	• Modbus/TCP support • Problem in Modbus RTU (serial) support where read requests returning longer than 32 bytes would give corrupt results fixed.	

		• Problem with hardware limit switch checking, starting a negative direction move while limit already asserted fixed.	
1/22/2016	6.0.2.25	• Added MAC address read for PROFINET protocol. • Added ability to run PROFINET firmware 4.1.0.2 on a Netx 52. • Added counters for Interrupt 1 and 2 used by PROFINET and ETHERCAT. RETH(58) and RETH(59) • Added way to read the synchronization correction being used RETH(57) • Changed USB to not send an ack when in state CMDLDCMPLT. • Fixed problem downloading netx firmware.	
1/14/2016	6.0.2.24	• Encrypted user program (SMXe) download support. Support for channels: COM0 (RS485), TCP/IP encapsulated serial, USB • Special calibration mode added for supporting raw resolution during calibration • Fixed issue with TS command under rare circumstances. • Added checks to program download to address concerns involving global program download. • Fixed issue with command Z(0,5) for clearing temperature fault. It was not correctly respecting the 5 degree cooldown as expected. •	

11/10/2015	6.0.2.23	• RETH/ETHCTL case 56 (EIP only) - this case sets/returns the size of the Ethernet/IP Implicit I/O connections size. The value is initialized to zero in the EEPROM. The possible values are:
		  ○ 8 - Use the CIP Position Controller/Position Controller Supervisor size of fixed 8 bytes. Any other size in the connection request will be rejected with an error response.
		  ○ 32 - Use Class6 Extended size of fixed 32 bytes. This setting will enable the use of the extended 32 byte data frames and the Extended Move Position and Velocity command types (6 & 7). Any other size in the connection request will be rejected with an error response.
		  ○ 0 - Use the size of the Connect request from the host, 8 or 32 bytes. Any other sizes will be rejected with an error response. This mode will allow the user to change the I/O connection size in the RS-Logix or other host PLC software on the fly.	

		- Add I/O Object 0x71 to allow access to the drive's on-board I/O. We found that I/O object 0x70 (used in Class5 motors). Object 0x70 is not CIP compliant since it uses a class 0 attribute 1 which is reserved by CIP (class 0 attributes 1 - 7 are reserved by CIP in all objects). I/O Object 0x71 works similar to I/O Object 0x71 but uses class 0 Attribute 8 for the I/O data word. (Object 0x70 is still supported for backward compatibility but not documented in class6) - Changed how CIP Position Controller ProfileInProgress flag is calculated in Torque Mode to provide similar behavior to Position and Velocity moves. In Torque Mode the flag will be set when the motor is on and Torque Value is non-zero and be cleared when the Torque Value is zero. - Restricted the Network Mode to a value between 1 and 65534 on power-up. Previously only checked that the value was greater than 1. This change will detect an un-initialized motor. - Limited the number of possible analog inputs to inputs 0 and 1 instead of 0 - 7 to allow for implementing Safe Torque Off (STO) in the future. - Changed the Analog Input range 0 - 18V (from 0 - 24V) to match the actual hardware capability. - Hold off updating the NetX52 while there is an outstanding - Fixed Implicit I/O error handling to match the errors expected in the ODVA conformance test CT12.	
7/28/2015	6.0.2.21	- Add error handling to the Attribute to Get Extended feature - Added code to ETHCTL and RETH for the motor view network screen. - Change the handling of Position Controller attributes 50, 51,56,57. 50,51 - The Class 5 firmware responds with the instantaneous left and right limit states. The Class 6 firmware uses the historical, debounced left and right states. 56,57 - The Class 5 firmware responds with a value of "1" if either the left /right hardware or software limits are set. The Class 6 firmware uses only the left/right software limits. This matches the CIP specification for the Position Controller object.	

		- Change the name of the EtherCAT and Ethernet/IP EDS files to be more meaningful and include the network abbreviation in the file name. - EtherCAT: ST23306M-EtherCAT.xml -> SM6_0_EEC_D402P01_6v17.xml - Ethernet/IP: SM6_0_D16P10V1R1_4.eds -> SM6_0_EIP_D16P10_V2R1_6.eds - Watchdog error when exercising the motor using the playground using the USB interface fixed. - Overflow bug in move generator when very small velocity and distance. This adds a check to the two denormalize functions so that a mantissa of 0 with any exponent is handled as 0 fixed. - Renishaw encoder correction causes hunting (buzzing) when a missing encoder state is commanded fixed.	
6/24/2015	6.0.2.20	- Change IP Address logic to match ODVA Recommended Methods for EIP Devices. The motor ships from the factory with DHCP enabled and no default IP Address. - Using the SMI terminal window to set the IP Address to 0.0.0.0 enables DHCP, any other address disables DHCP. These changes are remembered after a power cycle. - Changed the Ethernet I/P communications from the standard 8 byte Position Controller format to an extended 32 byte format. The 32 byte format is a superset of the original 8 byte format. The EDS file can be configured to use the 8 byte format if necessary. - TCP Serial encapsulation, UDP discovery. - Debug information dump to non-volatile flash memory. - RETH/ETHCTL commands updated: - RETH(5) Needed to satisfy Production's request to be able to read RSP5 into a user program. So x=ETH(5) allows the NetX LFW version to be handled in a user program. - RETH(18) MAC ID reporting (Ethernet/IP only) as a string: RETH(18) - RETH(45), RETH(46), RETH(47)	

Read access in a program to IP
address/mask/gateway x=ETH(45),   x=ETH(46),   x=ETH(47)
- ○ ETHCTL(45), ETHCTL(46), ETHCTL(47)
  Ability to set IP address / mask / gateway
  numerically. ETHCTL(45,x), ETHCTL(46,x), ETHCTL(47,x)
- ○ RETH(46), RETH(47)
  MAC ID read access from a user program (Ethernet/IP
  only):   x=ETH(48) and x=ETH(49)
- CIP profile new objects to support registration, in-position, deadband,
  home-level:
  - ○ Class 0x24, instance 1, attributes: 16, 20, 21, 22, 24
  - ○ Class 0x25, instance 1, attributes: 12, 38
  - ○ Position controller implicit message adds support for bits: 'on
    target position', 'home level', 'reg level', 'reg arm'
- Enable Watchdog handling to catch software problems where the
  processor hangs
- Changed RETH(15), RETH(16), RETH(17) to use "." as a separator
  instead of ":".   i.e. 192.168.1.1 instead of 192:168:1:1
- RSP6 and above fixed to return syntax error because command not
  valid fixed
- Fixed problem with object 0x2309 where GOSUB gets stuck and won't
  run more than once.
- Corrected SPI Clock on Netx52 for 8.5MHz. Defined
  DEFAULT_MIN_CYCLE constant that is specific to the Netx processor
  type (Netx50 vs Netx51/52.
- Fixed CTE command that was always displaying command error even
  when success.
- Fixed issue when writing the maximum possible fixed-length cam
  records to flash, the last record failed to write correctly.
- DHCP processing is working if the DHCP server assigns the IP address
  or if the IP address is assigned statically with the IPCTL command in
  the SMI Terminal window. There is a problem when using the DHCP
  server to assign the address and then disabling DHCP (This should

		assign the address statically). I am working with Hilscher support to clear this up. • Doing a MovePosition command when already at the target position does not generate at least one packet with the Profile in Progress bit set. Similar problem in Velocity mode. • Processor watchdog correction • Issue in EtherCAT with object 0x60fd inputs (travel limits and home were not read.) • Various issues fixed in cam support. Memory range checks, issue with erasing full range of cam area. • Fixed issue in EtherCAT with object 0x2309 GOSUB hanging after first use.	

5/27/2015	6.0.2.19	• Changed the calls to create the program threads to use a RAM resident character string so that the name will show correctly in the RTOS Q-Viewer
		• Added FJT(9999,0) UI command to dump thread stack usage statistics into the AW variables.
		• Thread stack sizes have been reduced to approximately 2x the memory actually used on all threads.
		• Changed ECAT and EIP thread functions to use a global transmit / receive buffer instead of large local buffers on the thread stack.
		• Add code so that the Network processing threads (ECAT, EIP, PNET) exit when the LOAD function is executed.
		• Removed support for the SD Card interface.
		• Change the code to funnel all reads/writes to EE through the SPI2MGR thread instead of looping on SPI2Mananager throughout the code.
		• Changed Standard EtherCAT PDO definitions to match other vendor object types.
		• RS-485 Serial Port changes
		   ○ Fixed the serial port interrupt/fiber and Modbus Command Manager to work correctly.
		   ○ Fixed the serial port interrupt/fiber/thread to work correctly at high baud rates (> 56K).	

| | | ○ Fixed the Serial Command manager and Serial Port Thread to properly handle program download commands. ○ Change the RS-485 TX Time-off constant from 10 to 1. With the new serial interrupt processing the TX Enable line was being held too long after a transmit was complete causing lost messages. Fixed a problem when using a motor configured for EtherCAT. It was causing errors when the motor was repeatedly unplugged from the EtherCAT master and then plugged back in. Added check to limit .nxf file name length. Fix ShadowIVT to vector to the qKernel trap functions Fix 0x1A and 0x1B response processing for Ethernet/IP Position Controller. Was not generating proper responses Fixed the RAT/RDT and AT=/DT= commands to correctly report/change the acceleration/deceleration values for the motor. Previously all of the commands used the acceleration value. Changed the EtherCAT code to report actual string size lengths to support variable length strings RTOS changes: ○ Changed RTOS version number from 3353 to 3356. Was overlooked in last release of the RTOS ○ Task are added to the end of their priority list when placed in the ready list instead of being placed at the beginning ○ Add processor reset at end of RTOS error logging (yErrorReset) ○ Change Idle task create to use a RAM resident character string so that the name will show correctly in the RTOS Q-Viewer ○ Fix qKrnStack to correctly calculate the Interrupt Stack used/free counts ○ Fix yThrEvtSignal/yThrEvtSignal wait routines to eliminate a race condition | |