PARAMETER DICTIONARY

copley controls

P/N 16-01091

Revision 04

March 2023

CONTENTS

1. ABOUT THIS MANUAL

1.1 Overview and Scope

This manual provides cross-referenced definitions of the parameters used to program and operate Copley Controls drives.

1.2 Related Documentation

CANopen-related documents:

- CANopen Programmer's Manual
- CML Reference Manual
- Copley Motion Objects Programmer's Guide

DeviceNet-related:

- Copley DeviceNet Programmer's Guide

Related interest:

- *CME User Guide*
- *Copley Indexer 2 Program User Guide*
- *Copley ASCII Interface Programmer's Guide*
- *Copley Camming User Guide*
- *AN102 - I/O Extension Features in Copley Modules*
- *AN137 - Setting Outputs at Position*

All these publications, along with hardware manuals and data sheets, can be found on www.copleycontrols.com

1.3 Comments

Copley Controls welcomes your comments on this manual.
See www.copleycontrols.com for contact information.

1.4 Copyrights

- Xenus, Accelnet, Stepnet, Accelus, and Junus are registered trademarks of Copley Controls.
- CME is a registered trademark of Copley Controls.
- MACRO is a registered trademark of Delta Tau Corp.

1.5 Document Validity

We reserve the right to modify our products. The information in this document is subject to change without notice and does not represent a commitment by Copley Controls. Copley Controls assumes no responsibility for any errors that may appear in this document.

1.6 Product Warnings

Observe all relevant state, regional, and local safety regulations when installing and using Copley Controls drives. For safety and to assure compliance with documented system data, only Copley Controls should perform repairs to drives.

DANGER

Hazardous voltages.

Exercise caution when installing and adjusting Copley drives.

Risk of electric shock.

On some Copley Controls drives, high-voltage circuits are connected to mains power. Refer to hardware documentation.

Risk of unexpected motion with non-latched faults.

After the cause of a non-latched fault is corrected, the drive re-enables the PWM output stage without operator intervention. In this case, motion may re-start unexpectedly. Configure faults as latched unless a specific situation calls for non-latched behavior. When using non-latched faults, be sure to safeguard against unexpected motion.

Latching an output does not eliminate the risk of unexpected motion with non-latched faults.

Associating a fault with a latched, custom-configured output does not latch the fault itself. After the cause of a non-latched fault is corrected, the drive re-enables without operator intervention. In this case, motion may re-start unexpectedly.

For more information, see Fault Mask (0xA7).

When operating the drive as a EtherCAT, MACRO, CAN or DeviceNet node, the use of CME or ASCII serial commands may affect operations in progress. Using such commands to initiate motion may cause network operations to suspend.

Operation may restart unexpectedly when the commanded motion is stopped.

Use equipment as described.

Operate drives within the specifications provided in the relevant hardware manual or data sheet.

FAILURE TO HEED THESE WARNINGS
CAN CAUSE EQUIPMENT DAMAGE, INJURY, OR DEATH.

1.7 REVISION HISTORY

Revision	Date	Comments
00	December 2013	Added new parameters and fixed existing content.
01	September 2014	Fixed units for parameter 0x5e
02	March 2019	Added new parameters and fixed the existing content
03	May 2021	Updated several parameters, updated tables, and made format adjustments
04	March 2023	Updated parameter 0x121 to include J1939 CANopen support for ARM and FPGA Plus drives.

2. INTRODUCTION

2.1 Scope and Purpose of this Document

This document provides a listing and definitions of the parameters used to program and operate Copley Controls drives. These parameters can be accessed using any of several communication interfaces, each with its own protocol and set of IDs for the parameters.

There are many CANopen and EtherCAT objects for which there are no direct correlations to Copley drive parameters. Refer to the *CANopen Programmer's Manual* for a complete list of supported objects.

2.2 Organization of the Parameter Listings

In section *3. Parameters*, table: *3.1 Parameters Sorted by ASCII Interface Parameter ID*, is organized into the following column headers / categories:

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
0x00	0x2380:1	R*	U16	Current Loop Proportional Gain (Cp).

Column header explanations:

The **ASCII** (American Standard Code for Information Exchange) column contains the parameter's Copley ASCII Interface parameter ID. This ID would also be used with Copley Controls Indexer 2 Program. The ID is listed in hex format.

The **CAN/ECAT IDX:SUB** column contains the CANopen and EtherCAT object index and sub-index of a parameter. The index is in hex format and the sub-index is in decimal format. Note that the CANopen and EtherCAT object libraries are identical.

The **Mem** column indicates whether the parameter is stored in drive RAM (R), drive flash memory (F), or both (RF).
An asterisk * next to R, F, or RF in this column indicates that the parameter is read-only. Parameters without an asterisk can be read and written.

The **Type** column indicates the parameter's data type. Types include:

- String: 20 words
- Integer (8, 16, 32, or 64-bit): INT8, INT16, INT32, INT64

- Unsigned (8, 16, 32, or 64-bit): U8, U16, U32, U64)

Cross references for each parameter include, where applicable, the equivalent CANopen (and EtherCAT) object index and sub-index.

The **Description** column includes object function and values.

It is important to note: that both the **DvcNet** column and the **MACRO** column have been removed from this revision of the *Parameter Dictionary.*

The DeviceNet ID can be derived from the ASCII ID by adding 1 to it.

Example: ASCII 0x00 = DvcNet 0x01 or ASCII 0x0F = DvcNet 0x10.

The MACRO ID can be derived from the ASCII ID by adding 0x400 to it.

Example: ASCII 0x00 = MACRO 0x400 or ASCII 0x0F = MACRO 0x40F

2.3 Important Notes

CME Refresh Behavior

When parameters are changed using one of the interfaces described in this manual, the changes will not necessarily be recognized by an active CME session.

Input/Output Numbering

Inputs and Outputs on Copley drives are numbered starting from zero for all the communication interfaces listed in this document. If a drive has 12 inputs, they are numbered 0 through 11. CME software starts numbering at 1.
(Input 0 is called IN1 in CME software).

3. PARAMETERS

The following table lists all available drive variables. The Mem column of the table identifies which banks of memory have instances of the variable. An R in this column indicates the variable is available in RAM, an F indicates the variable is available in flash memory. If this column contains an asterisk (*) then the parameter is read-only. Any ID values not listed are reserved for future use. All others are read and write parameters.

3.1 Parameters Sorted by ASCII Interface Parameter ID

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
0x00	0x2380:1	RF	U16	Current Loop Kp Proportional Gain (Cp).
0x01	0x2380:2	RF	U16	Current Loop Ki Integral Gain (Ci).
0x02	0x2340	RF	INT16	Current loop programmed value. Units: 0.01 A. This current will be used to command drive when Desired State (0x24) is set to 1.
0x03	0x2203	R*	INT16	Winding A Current. Units: 0.01 A. Actual current measured at winding A.
0x04	0x2204	R*	INT16	Winding B Current. Units: 0.01 A. Actual current measured at winding B.
0x05	0x2210	R*	INT16	Current Offset A. Units: 0.01 A. Offset value applied to winding A current reading. This offset is calculated by drive at startup.
0x06	0x2211	R*	INT16	Current Offset B. Units: 0.01 A. Offset value applied to winding B current reading. This offset is calculated by drive at startup.
0x07	0x2212	R*	INT16	X Axis of calculated stator current vector. Units: 0.01 A.
0x08	0x2213	R*	INT16	Y Axis of calculated stator current vector. Units: 0.01 A.
0x09	0x221A	R*	INT16	Current loop output, Stator Voltage, X axis. Units: 0.1 V
0x0A	0x221B	R*	INT16	Current loop output, Stator Voltage, Y axis. Units: 0.1 V
0x0B	0x2214	R*	INT16	Current reading. Actual Current, D axis of rotor space. Units: 0.01 A.
0x0C	0x2215	R*	INT16	Current reading. Actual Current, Q axis of rotor space. Units: 0.01 A. (Actual Current)
0x0D	0x2216	R*	INT16	Commanded current, D axis of rotor space. Part of internal current loop calculation. Units: 0.01 A.
0x0E	0x2217	R*	INT16	Commanded Current, Q axis of rotor space. Part of internal current loop calculation. Units: 0.01 A.
0x0F	None	R*	INT16	Current Error, D axis of rotor space. Units: 0.01 A.
0x10	None	R*	INT16	Current Error, Q axis of rotor space. Units: 0.01 A.
0x11	None	R*	INT16	Current Integral Value, D axis of rotor space.
0x12	None	R*	INT16	Current Integral Value, Q axis of rotor space.
0x13	0x2218	R*	INT16	Current Loop Output, D axis of rotor space. Units: 0.1 V (Terminal Voltage Stepper)

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
0x14	0x2219	R*	INT16	Current Loop Output, Q axis of rotor space. Units: 0.1 V (Terminal Voltage Servo)
0x15	0x221D	R*	INT16	Commanded Motor Current. Units: 0.01 A. This is value presently being sent to current loop. It may come from programmed value, analog reference, velocity loop, etc. depending on drive's desired state.
0x16	None	RF	INT16	Programmable Voltage Limit. Units: 100mV. This value limits the maximum PWM output duty cycle so that the max output will not exceed this limit. Note that the max PWM output duty cycle is recalculated approximately every 100ms based on the bus voltage, so quick increases in bus voltage may cause the limit to be exceeded until the PWM duty cycle is recalculated.
0x17	0x6063 0x6064	R	INT32	Actual Position. Units: Counts. Used to close position loop in drive every servo cycle. For single feedback systems, this value is same as Actual Motor Position (0x32). For dual feedback systems, this value is same as Load Encoder Position (0x112). CANopen objects 0x6064 and 0x6063 hold same value.
0x18	0x6069 0x606C	R*	INT32	Actual Velocity. Units: 0.1 encoder counts/s. For estimated velocity. Units: 0.01 RPM. For stepper mode: Units: 0.1 microsteps/s.
0x19	0x2310	RF	INT32	Analog Reference Scaling Factor. This value is used to scale analog reference input voltage to a command that will be used to drive current, velocity or position loop (depending on drive state). When in current mode (Desired State (0x24) = 2), value programmed specifies commanded current when 10 V is applied to analog input. Units: 0.01 A. For example, to command 12 A at 10 V, scaling factor would be 1200. When in velocity mode (Desired State (0x24) = 12), value programmed specifies commanded velocity when 10 V is applied to analog input. Units: 0.1 encoder counts/s. For estimated velocity. Units: 0.01 RPM. For stepper mode. Units: 0.1 microsteps/s. When in position mode (Desired State (0x24) = (22 or 32), value programmed specifies commanded position (in encoder counts) when 10 V is applied to analog input.
0x1A	0x2311	RF	INT16	Offset Value applied to Analog Input or Analog Reference Input. Units: mV.
0x1B	0x2205	R*	INT16	Analog 1Vpp Encoder Sine Input Voltage. Units: 0.1 mV. Also known as Sine Feedback Voltage.
0x1C	0x2206	R*	INT16	Analog 1Vpp Encoder Cosine Input Voltage. Units: 0.1 mV. Also known as Cosine Feedback Voltage.

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td>0x1D</td><td>0x2200</td><td>R*</td><td>INT16</td><td colspan="2">Analog Input. Units: mV.
Also known as Analog Reference Input Voltage.</td></tr>
<tr><td>0x1E</td><td>0x2201</td><td>R*</td><td>INT16</td><td colspan="2">High Voltage A/D Reading. Units: 100 mV.
Bus Voltage present on internal high-voltage bus.</td></tr>
<tr><td>0x1F</td><td>0x2207</td><td>R*</td><td>INT16</td><td colspan="2">Primarily of diagnostic interest, this parameter gives the offset value applied to the internal A/D unit. It is part of a continuous calibration routine that the drive performs on itself while running.</td></tr>
<tr><td>0x20</td><td>0x2202</td><td>R*</td><td>INT16</td><td colspan="2">Drive Temperature A/D Reading.
Units: degrees C. Range 0C to 99C.</td></tr>
<tr><td>0x21</td><td>0x2110</td><td>RF</td><td>INT16</td><td colspan="2">Peak Current Limit. Units: 0.01 A.
Used by I^2T algorithm to protect motor. Also known as Boost current on stepper drives.
This value cannot exceed Drive’s Peak Current (0xDE).
Peak current range 0 to peak overrides continuous current limit.</td></tr>
<tr><td>0x22</td><td>0x2111</td><td>RF</td><td>INT16</td><td colspan="2">Continuous Current Limit. Units: 0.01 A.
Used by I^2T algorithm to protect motor. Also known as Run Current on stepper drives. This value cannot exceed Drive’s Continuous Current Limit.</td></tr>
<tr><td>0x23</td><td>0x2112</td><td>RF</td><td>U16</td><td colspan="2">Time at Peak Current Limit. Units: ms.
Used by I^2T algorithm to protect motor. Also known as Time at Boost Current for stepper drives.</td></tr>
<tr><td rowspan="23">0x24</td><td rowspan="23">0x2300</td><td rowspan="23">RF</td><td rowspan="23">U16</td><td colspan="2">Desired State:</td></tr>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>0</td><td>Drive disabled</td></tr>
<tr><td>1</td><td>Programmed current value drives current loop</td></tr>
<tr><td>2</td><td>Analog reference drives current loop</td></tr>
<tr><td>3</td><td>PWM input drives current loop</td></tr>
<tr><td>4</td><td>Function generator drives current loop</td></tr>
<tr><td>5</td><td>UV current mode</td></tr>
<tr><td>6</td><td>Reserved</td></tr>
<tr><td>7</td><td>Current command slaved to lower axis</td></tr>
<tr><td>8-10</td><td>Reserved</td></tr>
<tr><td>11</td><td>Programmed velocity value drives velocity loop</td></tr>
<tr><td>12</td><td>Analog reference drives velocity loop</td></tr>
<tr><td>13</td><td>PWM input drives velocity loop</td></tr>
<tr><td>14</td><td>Function generator drives velocity loop</td></tr>
<tr><td>15-16</td><td>Reserved</td></tr>
<tr><td>17</td><td>Velocity command slaved to lower axis</td></tr>
<tr><td>18-20</td><td>Reserved</td></tr>
<tr><td>21</td><td>Trajectory generator drives position loop</td></tr>
<tr><td>22</td><td>Analog reference drives position loop</td></tr>
<tr><td>23</td><td>Digital input lines drive position loop (Pulse & direction, master encoder, etc.)</td></tr>
<tr><td>24</td><td>Function generator drives position loop</td></tr>
<tr><td>25</td><td>Cam tables drive position loop</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td>26</td><td>Analog reference commands velocity to position loop</td></tr>
<tr><td></td><td></td><td></td><td></td><td>27</td><td>Position command slaved to lower axis</td></tr>
<tr><td></td><td></td><td></td><td></td><td>28-29</td><td>Reserved</td></tr>
<tr><td></td><td></td><td></td><td></td><td>30</td><td>CANopen interface controls drive</td></tr>
<tr><td></td><td></td><td></td><td></td><td>31</td><td>Trajectory generator drives microstepper</td></tr>
<tr><td></td><td></td><td></td><td></td><td>32</td><td>Analog reference drives microstepper position</td></tr>
<tr><td></td><td></td><td></td><td></td><td>33</td><td>Digital input lines drive microstepper</td></tr>
<tr><td></td><td></td><td></td><td></td><td>34</td><td>Function generator drives microstepper</td></tr>
<tr><td></td><td></td><td></td><td></td><td>35</td><td>Cam tables drive microstepper</td></tr>
<tr><td></td><td></td><td></td><td></td><td>36</td><td>Analog reference drives microstepper velocity</td></tr>
<tr><td></td><td></td><td></td><td></td><td>37</td><td>Position slaved to another axis in microstepping mode</td></tr>
<tr><td></td><td></td><td></td><td></td><td>38-39</td><td>Reserved</td></tr>
<tr><td></td><td></td><td></td><td></td><td>40</td><td>CANopen interface controls microstepper</td></tr>
<tr><td></td><td></td><td></td><td></td><td>41</td><td>Reserved</td></tr>
<tr><td></td><td></td><td></td><td></td><td>42</td><td>Simple microstepping mode
For diagnostic use only.</td></tr>
<tr><td>0x25</td><td>0x221E</td><td>R*</td><td>INT16</td><td colspan="2">Limited Current. Units: 0.01 A. Limits the current to the current loop.</td></tr>
<tr><td>0x26</td><td>0x2313</td><td>RF</td><td>INT16</td><td colspan="2">Analog Reference Input Deadband. Units: mV.
Deadband window value applied to analog input.</td></tr>
<tr><td>0x27</td><td>0x2381:1</td><td>RF</td><td>U16</td><td colspan="2">Velocity Loop Kp Proportional Gain (Vp).</td></tr>
<tr><td>0x28</td><td>0x2381:2</td><td>RF</td><td>U16</td><td colspan="2">Velocity Loop Ki Integral Gain (Vi).</td></tr>
<tr><td>0x29</td><td>0x2230</td><td>R*</td><td>INT32</td><td colspan="2">Limited Velocity. This is commanded velocity after it passes through the velocity loop limiter and the velocity command filter. It is velocity value that the velocity loop will attempt to achieve.
Units: 0.1 encoder counts/s.
For estimated velocity. Units: 0.01 RPM. For stepper mode. Units: 0.1 microsteps/s.</td></tr>
<tr><td>0x2A</td><td>0x2233</td><td>R*</td><td>INT32</td><td colspan="2">Velocity Loop Error.</td></tr>
<tr><td>0x2B</td><td>None</td><td>R*</td><td>INT32</td><td colspan="2">Velocity Loop Integral Sum. Sum of the error multiplied by Ki (Vi) over time.</td></tr>
<tr><td>0x2C</td><td>0x606B</td><td>R*</td><td>INT32</td><td colspan="2">Commanded Velocity. Units: 0.1 encoder counts/s.
For estimated velocity (voltage). Units: 0.01 RPM.
For stepper mode. Units: 0.1 microsteps/s.</td></tr>
<tr><td>0x2D</td><td>0x6062</td><td>R*</td><td>INT32</td><td colspan="2">Limited Position. Units: counts.
In classical terms it is the commanded position that goes to the summing junction with the actual position to produce the position error.</td></tr>
<tr><td>0x2E</td><td>0x2381:3</td><td>RF</td><td>U16</td><td colspan="2">Velocity Loop Acceleration Feed Forward (Aff).
Acceleration command from trajectory generator is multiplied by this value and result is added to velocity loop input.</td></tr>
<tr><td>0x2F</td><td>0x2341</td><td>RF</td><td>INT32</td><td colspan="2">Programmed Velocity Command.
Only used in Programmed Velocity Mode (Desired State 0x24 = 11). Units: 0.1 encoder counts/s.
For estimated velocity (voltage). Units: 0.01 RPM. For stepper mode. Units: 0.1 microsteps/s.</td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
0x30	0x2382:1	RF	U16	Position Loop Proportional Gain (Pp).
0x31	0x2381:4	RF	INT16	Velocity Loop Shift Value. After velocity loop is calculated, result is right shifted this (value) many times to arrive at commanded current value. This allows velocity loop gains to have reasonable values for high resolution encoders.
0x32	0x2240	R*	INT32	Actual Motor Position. Units: counts. Gives feedback position of motor. For single feedback systems, this is same as Actual Position (0x17).
0x33	0x2382:2	RF	U16	Position Loop Velocity Feed Forward (Vff). Vff value is multiplied by Instantaneous Commanded Velocity (0x3B) generated by trajectory generator. Product is added to output of position loop. This gain is scaled by 1/16384. Therefore, setting this gain to 0x4000 (16384) would cause input velocity to be multiplied by 1.0 (100% Vff), and result added to output of position loop.
0x34	0x2382:3	RF	U16	Position Loop Acceleration Feed Forward (Aff). Aff value is multiplied by Instantaneous Commanded Velocity (0x3B) generated by trajectory generator. Product is added to output of position loop.
0x35	0x60F4	R*	INT32	Position Loop Error. Units: counts. Difference between Actual Position (0x17) and Limited Position (0x2D).
0x36	0x2100	RF	U32	Velocity Loop Acceleration Limit. Units: 1000 counts/s^2. Used by velocity loop limiter. Not used when velocity loop is controlled by position loop.
0x37	0x2101	RF	U32	Velocity Loop Deceleration Limit. Units: 1000 counts/s^2. Used by velocity loop limiter. Not used when velocity loop is controlled by position loop.
0x38	0x221C	R*	INT16	Actual Motor Current. Units: 0.01 A. This current is calculated based on both D and Q axis currents.
0x39	0x2102	RF	U32	Velocity Loop Emergency Stop Deceleration Rate. Units: 1000 counts/s^2.
0x3A	0x2103	RF	INT32	Velocity Loop Velocity Limit. Units 0.1 counts/s. This value limits commanded velocity used by velocity loop. Note that this limit is always in effect for safety to protect the motor from over speed command.
0x3B	0x2250	R*	INT32	Instantaneous Commanded Velocity. Units: 0.1 encoder counts/s. This velocity is output of trajectory generator and is value by which position loop's velocity feed forward is multiplied.
0x3C	0x2251	R*	U32	Instantaneous Commanded Acceleration. Units: 10 encoder counts/s^2. This acceleration is output

<table>
<tr><th>ASCII</th><th>CAN/ECAT
IDX: SUB</th><th>Mem</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td>of trajectory generator and is value by which position loop’s acceleration feed forward is multiplied.</td></tr>
<tr><td>0x3D</td><td>0x2122</td><td>R*</td><td>INT32</td><td>Trajectory Destination Position.
Units: encoder counts. This is position that the trajectory generator is using as its destination.</td></tr>
<tr><td>0x3E</td><td>0x2104</td><td>RF</td><td>INT32</td><td>Velocity Window. Units: 0.1 counts/s.
If absolute value of velocity loop error exceeds this, then velocity window bit in Event Status Register (0xA0) will be set.</td></tr>
<tr><td>0x3F</td><td>0x2105</td><td>RF</td><td>U16</td><td>Velocity Window Time. Units: ms.
Velocity window bit in Event Status Register (0xA0) will be cleared when absolute velocity error is less than velocity window for this amount of time.</td></tr>
<tr><td>0x40</td><td>0x2383:1</td><td>F</td><td>U16</td><td>Motor Type. Type of motor connected to drive.
Bit-mapped as follows:
<table>
<tr><th>Bits</th><th colspan="2">Description</th></tr>
<tr><td>0</td><td colspan="2">Set for linear, clear for rotary.</td></tr>
<tr><td>1-3</td><td colspan="2">Reserved.</td></tr>
<tr><td>4-5</td><td colspan="2">Motor architecture:</td></tr>
<tr><td></td><td>0</td><td>Not specified</td></tr>
<tr><td></td><td>1</td><td>DC Brush, 2 Wire Coil, or Voice Coil</td></tr>
<tr><td></td><td>2</td><td>Microstepper or Stepper motor</td></tr>
<tr><td></td><td>3</td><td>Brushless servo motor</td></tr>
<tr><td>6-15</td><td colspan="2">Reserved.</td></tr>
</table></td></tr>
<tr><td>0x41</td><td>0x6404</td><td>F</td><td>String</td><td>Motor Manufacturer Name.</td></tr>
<tr><td>0x42</td><td>0x6403</td><td>F</td><td>String</td><td>Motor Model Number.</td></tr>
<tr><td>0x43</td><td>0x2383:27</td><td>F</td><td>INT16</td><td>Motor Units. This is only used by CME for display. (0=metric, 1=English).</td></tr>
<tr><td>0x44</td><td>0x2383:9</td><td>F</td><td>INT32</td><td>Motor Inertia (Mass). Units: Rotary = 0.000001 Kg/cm^2 .
Units: Linear = 0.0001 Kg.</td></tr>
<tr><td>0x45</td><td>0x2383:2</td><td>F</td><td>INT16</td><td>Motor Poll Pairs (used only for rotary motors). Number of motor pole pairs (electrical phases) per rotation. For stepper motors, Poll Pairs = (360 deg / Motor deg/step) / 4.</td></tr>
<tr><td>0x46</td><td>0x2383:16</td><td>F</td><td>U16</td><td>Motor Brake Type. 0=present, 1=none.</td></tr>
<tr><td>0x47</td><td>0x2383:15</td><td>F</td><td>U16</td><td>Motor Temperature Sensor Type. 0=none, 1=present.</td></tr>
<tr><td>0x48</td><td>0x2383:12</td><td>F</td><td>INT32</td><td>Motor Torque Constant. Units: 0.00001 Nm/A.</td></tr>
<tr><td>0x49</td><td>0x2383:7</td><td>F</td><td>INT16</td><td>Motor Resistance. Units: 10 mΩ. (10-milliohms)</td></tr>
<tr><td>0x4A</td><td>0x2383:8</td><td>F</td><td>INT16</td><td>Motor Inductance. Units: 10 μH. (10-microhenrys)</td></tr>
<tr><td>0x4B</td><td>0x2383:13</td><td>F</td><td>INT32</td><td>Motor Peak Torque. Units: 0.00001 Nm units.</td></tr>
<tr><td>0x4C</td><td>0x2383:14</td><td>F</td><td>INT32</td><td>Motor Continuous Torque. Units: 0.00001 Nm units.</td></tr>
<tr><td>0x4D</td><td>0x2383:11</td><td>F</td><td>INT32</td><td>Motor Max Velocity. Units: 0.1 encoder counts/s.</td></tr>
<tr><td>0x4E</td><td>0x2383:3</td><td>F</td><td>U16</td><td>Motor Wiring. 0=standard, 1= drive’s U and V outputs are swapped. (0=normal, 1=reverse)</td></tr>
<tr><td>0x4F</td><td>0x2383:6</td><td>RF</td><td>INT16</td><td>Motor Hall Offset (Phase Offset). Units: degrees.
Offset angle to be applied to Hall Effect sensors or other feedback types.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th>Description</th></tr>
<tr><td>0x50</td><td>0x2383:4</td><td>F</td><td>INT16</td><td>Motor Hall Type.
Type of Hall Effect sensors attached to motor:
<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>0</td><td>No Hall Effect sensors available.</td></tr>
<tr><td>1</td><td>Digital Hall Effect sensors.</td></tr>
<tr><td>2</td><td>Analog Hall Effect sensors.</td></tr>
</table></td></tr>
<tr><td>0X51</td><td>0x2383:10</td><td>F</td><td>U16</td><td>Motor back EMF constant (<b>obsolete</b>, variable 0x56 is now used which accesses same data but with extended range)
Units: Rotary 0.01 V/krpm; Linear 0.01 V/m/s</td></tr>
<tr><td>0x52</td><td>0x2383:5</td><td>F</td><td>INT16</td><td>Motor Hall Effect Wiring. Bit-mapped as follows:
NOTE: When analog Halls are used, only bit 8 is relevant.
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0-2</td><td>The Hall wiring code (see below).
<table>
<tr><th>Value</th><th>Hall Ordering</th></tr>
<tr><td>0</td><td>U V W</td></tr>
<tr><td>1</td><td>U W V</td></tr>
<tr><td>2</td><td>V U W</td></tr>
<tr><td>3</td><td>V W U</td></tr>
<tr><td>4</td><td>W V U</td></tr>
<tr><td>5</td><td>W U V</td></tr>
<tr><td>6, 7</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>3</td><td>Reserved</td></tr>
<tr><td>4</td><td>Invert W Hall input if set. Inversion occurs after Halls wiring is changed by bits 0-2.</td></tr>
<tr><td>5</td><td>Invert V Hall input if set. Inversion occurs after Halls wiring is changed by bits 0-2.</td></tr>
<tr><td>6</td><td>Invert U Hall input if set. Inversion occurs after Halls wiring is changed by bits 0-2.</td></tr>
<tr><td>7</td><td>Reserved</td></tr>
<tr><td>8</td><td>If set, reverse analog Halls.</td></tr>
<tr><td>9-15</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>0x53</td><td>0x2383:17</td><td>F</td><td>U16</td><td>Motor Brake Activation Time. Units: ms.</td></tr>
<tr><td>0x54</td><td>0x2383:18</td><td>F</td><td>U16</td><td>Motor Brake Delay Time. Units: ms.
After brake output is activated, drive will stay enabled for this amount of time to allow brake to engage.</td></tr>
<tr><td>0x55</td><td>0x2383:19</td><td>F</td><td>INT32</td><td>Motor Brake Activation Velocity. Units: 0.1 counts/s.
During Motor Brake Activation Time (0x53), if motor's actual velocity falls below this value brake output is activated immediately.</td></tr>
<tr><td>0x56</td><td>0x2383:10</td><td>F</td><td>U32</td><td>Motor Back EMF Constant. Replaces (0x51), with 32 bits for extended range.

Units: Rotary 0.01 V/krpm
Units: Linear 0.01 V/m/s</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="3">Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="3">Back EMF velocity estimation can be disabled by setting to zero.</td></tr>
<tr><td>0x57</td><td>0x2383:29</td><td>F</td><td>U32</td><td colspan="3">Microsteps/Motor Rev. Units: microsteps.
This parameter is used in true microstepping mode.</td></tr>
<tr><td>0x58</td><td>0x2383:33</td><td>F</td><td>INT32</td><td colspan="3">Motor Gear Ratio.
This parameter may be used to store gear ratio information for dual encoder systems where gearbox sits between two encoders. This parameter is not used by firmware and is supported as convenience to CME program.
Gear ratio is ratio of two 16-bit values. First word gives number of motor turns and is numerator. Second word gives number of position turns and is denominator.</td></tr>
<tr><td>0x59</td><td>0x2107</td><td>RF</td><td>INT16</td><td colspan="3">Hall Velocity Mode Shift Value (Hall multiplier).
This parameter is only used in Hall velocity mode. It specifies left shift value (in multiples of 2) for position, velocity, and acceleration calculations.</td></tr>
<tr><td rowspan="13">0x5A</td><td rowspan="13">0x2241</td><td rowspan="13">RF</td><td rowspan="13">INT16</td><td colspan="3">Encoder Output Configuration.
This parameter determines the configuration of multi-mode encoder port output on drives that support the multi-mode encoder port.
Bit-mapped as follows:</td></tr>
<tr><td><b>Bits</b></td><td colspan="2"><b>Description</b></td></tr>
<tr><td rowspan="5">0-1</td><td colspan="2">Mode of operation for encoder output lines.</td></tr>
<tr><td>0</td><td>Output buffered primary encoder (hardware buffering).</td></tr>
<tr><td>1</td><td>Configure as secondary encoder input.</td></tr>
<tr><td>2</td><td>Output simulated (emulated) encoder outputs tracking motor encoder.</td></tr>
<tr><td>3</td><td>Output simulated (emulated) encoder outputs tracking load encoder.</td></tr>
<tr><td>4</td><td colspan="2">If set, force X and S channels to be inputs no matter what mode bits 0-1 specify. This is useful for some special modes that take commands on these lines while outputting encoder data on the A and B lines.</td></tr>
<tr><td rowspan="5">8-11</td><td colspan="2">For simulated (emulated) encoder outputs, these bits configure scaling value that adjusts number of encoder output counts for each encoder count on the input.
This setting also scales the max output frequency (nominally 10MHz) by the same amount.</td></tr>
<tr><td>0</td><td>No adjustment, 1 count on the encoder is 1 output count.</td></tr>
<tr><td>1</td><td>Multiply encoder counts by 2.</td></tr>
<tr><td>2</td><td>Divide encoder counts by 2.</td></tr>
<tr><td>3</td><td>Divide encoder counts by 4.</td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
					4	Divide encoder counts by 8.
					5	Divide encoder counts by 16.
					6	Divide encoder counts by 32.
					7	Divide encoder counts by 64.
					8	Divide encoder counts by 128.
					9	Divide encoder counts by 256.
				12-13	For simulated (emulated) encoder outputs, these bits allow the max output frequency to be reduced	
					0	No change to max output frequency
					1	Divide max output frequency by 2
					2	Divide max output frequency by 4
					3	Divide max output frequency by 8
0x5B	0x2383:32	F	INT32	Load Encoder Resolution. Units: Encoder unit/count. Used for linear motors only. Number of Motor Encoder Units (0x61) per encoder count.		
0x5C	0x2383:31	F	INT16	Load Encoder Direction. 0=normal, 1=reverse. Note: Change in direction will affect motor phasing.		
0x5D	0x2383:30	F	U16	Load Encoder Type. This parameter identifies type of encoder used on load when running in dual loop mode. Encoding of this parameter has changed over time to support more encoder types than were originally envisioned when parameter was first defined. Bit 12 is used to identify which encoding is active. Original encoding (bit 12 not set):		
				Bits	**Meaning**	
				0-3	Encoder hardware to use:	
					0	No load encoder present
					1	Primary (differential) quad encoder
					2	Analog encoder sine cosine
					3	Secondary quad encoder from input lines
					4	Low frequency analog encoder (Servo tube/analog halls/sine cosine)
					5	Resolver
					11	EnDat absolute encoder
					12	SSI serial encoder
					13	BiSS absolute encoder
					14	Various absolute encoders made by Sanyo Denki, Panasonic, and Harmonic Drives
					15	Harmonic Drives custom encoder
				4	If set, linear encoder. If clear, rotary encoder.	
				5	If set, do not use this encoder for closing position loop. Passively monitors load position.	
				6-15	Reserved. Must be set to zero.	

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description			
				New encoding supported by 8367 firmware starting with version 2.10:			
				0-11	Encoder hardware to use:		
						0-15	Same encoder types as listed above.
						16	Simple analog potentiometer for feedback
						17	Gurley virtual absolute encoder
						18	Custom encoder K
						19	S2 custom encoder
						20	Hiperface.
						22	Sankyo absolute encoder
				12	Always set to identify new encoding.		
				13	If set, linear encoder. If clear, rotary encoder.		
				14	If set, do not use this encoder for position feedback.		
				15	Reserved		
0x5E	0x2231	R*	INT32	Load Encoder Velocity. Units: 0.1 encoder counts/s			
0x5F	0x2106	RF	9 or 14	Velocity Loop Output Filter. Bi-quad filter which acts on output of velocity loop. 9- or 14-word parameters, see Filter Coefficients.			
0x60	0x2383:20	F	U16	Motor Encoder Type:			
				Value	**Meaning**		
				0	Primary (differential) quad encoder		
				1	No encoder (use motor back EMF for velocity estimation)		
				2	Analog encoder sine cosine		
				3	Secondary quad encoder from input lines (multimode port)		
				4	Low frequency analog encoder		
				5	Resolver		
				6	Use digital hall signals for position and velocity estimates		
				7	Analog encoder updated at current loop rate		
				8	Custom Y encoder		
				9	Panasonic		
				10	SPI command (reserved for custom firmware use).		
				11	EnDat		
				12	SSI		
				13	BiSS		
				14	Serial encoders from Sanyo Denki, Tamagawa, Panasonic and HD systems		
				15	Custom encoders from HD systems		
				16	Simple analog potentiometer feedback		
				17	Gurley virtual absolute encoder		
				18	Custom K encoder		

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				19	S2 custom encoder
				20	Hiperface
				21	Wire saving incremental encoder which outputs hall signals on encoder lines at power-up
				22	Sankyo absolute encoder
				23	Custom M encoder HG absolute
				24	Digital inputs used as tertiary encoder inputs. Inputs configured as single ended or differential by using Digital Input Command Configuration (0xA8). Not used in Desired State (0x24) modes 3, 13 and 23 (PWM or Digital Input Command Modes).
				25	Tachometer input
				26	Tamagawa TS5643 absolute encoder
				27	Hiperface DSL (using external adapter board)
0x61	0x2383:21	F	INT16	Motor Encoder Units. Value defines units used to describe linear motor encoders. Not used with rotary motors.	
				Value	Description
				0	Micrometers E-6
				1	Nanometers E-9
				2	Millimeters E-3
0x62	0x2383:23	F	INT32	Motor Encoder Counts/Rev. Units: Counts/rev. Used for rotary motors only. When resolver is used as motor feedback, sets resolution of interpolated position.	
0x63	0x2383:24	F	INT16	Motor Encoder Resolution. Linear motor only. Units: encoder units/count.	
0x64	0x2383:25	F	INT32	Motor Encoder Electrical Distance. Linear motor only. Units: encoder units/electrical cycle.	
0x65	0x2383:22	F	U16	Motor Encoder Direction. 0=normal, 1=reverse. Note: Change in direction will affect motor phasing.	
0x66	0x2383:26	F	U32	Encoder Index Marker Pulse Distance. Units: rotary, counts; linear, encoder units. Reserved.	
0x67	0x2383:28	F	INT16	Analog Encoder Shift Amount. This value gives number of bits of interpolation to be applied to an analog encoder. Encoder resolution with no interpolation (shift value of 0) is 4 encoder counts/encoder line. Setting this parameter to value of n would give total of 2^(n+2) counts/line.	
0x68	0x2402	R*	INT32	Captured Index Position. Units: counts. Provides position that axis was in when an index pulse was captured. Configured by setting bits in Position Capture Control Register (0x6C), and status of captured data can be checked in Position Capture Status Register (0x6D). Reading this variable resets *bits 0 & 3* of Position Capture Status Register (0x6D).	
0x69	0x2232	R*	INT32	Unfiltered Motor Encoder Velocity. Units 0.1 counts/s.	

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td>0x6A</td><td>0x2113</td><td>RF</td><td>INT32</td><td colspan="2">Commanded Current Ramp Limit. Units: mA/s.
Used when running in Current (Torque) mode. Setting this to zero disables slope limiting.</td></tr>
<tr><td>0x6B</td><td>0x2108</td><td>RF</td><td>9 or 14</td><td colspan="2">Velocity Loop Command Filter Coefficients.
Bi-quad filter structure that acts on command input of velocity loop just after velocity & acceleration limiting.
9- or 14-word parameters, see Filter Coefficients.</td></tr>
<tr><td rowspan="16">0x6C</td><td rowspan="16">0x2400</td><td rowspan="16">RF</td><td rowspan="16">INT16</td><td colspan="2">Position Capture Control Register. Sets up position capture based on index or home input.
Bit-mapped as follows:</td></tr>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0</td><td>If set, Captured Index Position (0x68) is captured on rising edge of index input.</td></tr>
<tr><td>1</td><td>If set, Captured Index Position (0x68) is captured on falling edge of index input.</td></tr>
<tr><td>2</td><td>If set, Captured Index Position (0x68) value will not be overwritten by new position until it has been read.
If clear, new positions will overwrite old positions.</td></tr>
<tr><td>3, 4</td><td>Reserved</td></tr>
<tr><td>5</td><td>If set, Captured Home Position (0x10A) will be captured on active to inactive edge of home input switch.
If clear, home position will be captured on inactive to active edge.</td></tr>
<tr><td>6</td><td>If set, Captured Home Position (0x10A) will not be overwritten by new position until it has been read.
If clear, new positions will overwrite old positions.</td></tr>
<tr><td>7</td><td>Reserved</td></tr>
<tr><td>8</td><td>If set, enable high-speed input position capture, Captured Position for High-Speed Position Capture (0x111)</td></tr>
<tr><td>9</td><td>If set, don't overwrite high-speed input capture positions</td></tr>
<tr><td>10</td><td>If set, latch high-speed position capture.</td></tr>
<tr><td>11</td><td>Reserved</td></tr>
<tr><td>12</td><td>Clear Actual Position (0x17) on every encoder index pulse</td></tr>
<tr><td>13</td><td>If set, reset phase angle every time index is captured.
Requires 4.40 or later Plus drive firmware, 1.80 or later ARM drive firmware. See description below.</td></tr>
<tr><td colspan="2">If bit 13 is set in firmware supporting this option, then the first time an index is captured after enabling this option the phase angle will be stored internally.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="2">On subsequent index captures the phase angle will be reset to this stored value. This requires phase mode 0 or 2 on incremental encoders.</td></tr>
<tr><td rowspan="11">0x6D</td><td rowspan="11">0x2401</td><td rowspan="11">R*</td><td rowspan="11">INT16</td><td colspan="2">Position Capture Status Register. This register shows status of index/home capture mechanism.
Bit-mapped as follows:</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>If set, index position has been captured. Cleared when captured position is read.</td></tr>
<tr><td>1-2</td><td>Reserved</td></tr>
<tr><td>3</td><td>If set, new index transition occurred when captured position was already stored. Depending on the value of bit 2 of parameter 0x6C, depending on mode, new position may have been discarded or overwritten previously stored position.</td></tr>
<tr><td>4</td><td>If set, home position has been captured. Cleared when captured position is read</td></tr>
<tr><td>5-6</td><td>Reserved</td></tr>
<tr><td>7</td><td>If set, new home pin transition occurred when captured position was already stored. Depending on the value of bit 6 of parameter 0x6C, depending on mode, new position may have been discarded or overwritten previously stored position.</td></tr>
<tr><td>8</td><td>If set, new high-speed capture data</td></tr>
<tr><td>9-10</td><td>Reserved</td></tr>
<tr><td>11</td><td>If set, high-speed capture overflow. Depending on the setting of bit 10 of parameter 0x6C, the new position may have been discarded or overwritten the previously stored position.</td></tr>
<tr><td>0x6E</td><td>0x2383:34</td><td>F</td><td>INT16</td><td colspan="2">Number of Resolver Cycles/Motor Rev. Used only with resolver feedback devices.</td></tr>
<tr><td rowspan="6">0x6F</td><td rowspan="6">0x2140</td><td rowspan="6">RF</td><td rowspan="6">INT16</td><td colspan="2">PWM Mode and Status. This bit-mapped register allows some details of the PWM output to be controlled and monitored.
Bit-mapped as follows:</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>If set, force bus clamping (0-100% modulation).
If clear, disable bus clamping (center weighted modulation).
If bit 1 set, this bit is ignored.</td></tr>
<tr><td>1</td><td>If set, automatic bus clamping.
Setting this bit causes bus clamping mode to be automatically selected based on output voltage.
Bit 0 ignored if this bit is set.</td></tr>
<tr><td>2</td><td>Reserved</td></tr>
<tr><td>3</td><td>Factory reserved (dynamic brake).
If set, short motor outputs when disabled.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="3">Description</th></tr>
<tr><td rowspan="5"></td><td rowspan="5"></td><td rowspan="5"></td><td rowspan="5"></td><td>4</td><td colspan="2">If set, use hexagonal voltage limiting.
If clear, use circular voltage limiting.</td></tr>
<tr><td>5</td><td colspan="2">Reserved</td></tr>
<tr><td>6</td><td colspan="2">If set, double PWM frequency.</td></tr>
<tr><td>7</td><td colspan="2">Reserved</td></tr>
<tr><td>8</td><td colspan="2">Status bit set when bus clamping is active.</td></tr>
<tr><td rowspan="10">0x70</td><td rowspan="10">0x2193:1</td><td rowspan="10">RF</td><td rowspan="10">3 -5</td><td colspan="3">Output 0 (OUT1) Configuration.
For notes on Output numbering see Input/Output Numbering.
Data type is dependent on configuration and uses 3- to 5-words.
First word is bit-mapped configuration value. Remaining words give additional parameter data used by output pin. Typically, second and third words are used as 32-bit bitmask to identify which bit(s) in Event Status Register (0xA0) output should follow. If any selected bits in Event Status Register (0xA0) are set, then output will go active. If no selected bits in Event Status Register (0xA0) are set, then output will be inactive.
Output 0 (OUT1) may be programmed as sync output for use in synchronizing multiple drives. In this configuration, first word of this variable should be set to 0x0200 (i.e., only bit 9 is set) and remaining words should be set to zero.
Note that only Output 0 (OUT1) has this feature. Attempting to program any other output pin as sync output will have no effect.
The first word is bit-mapped as follows:</td></tr>
<tr><th>Bits</th><th colspan="2">Configuration</th></tr>
<tr><td rowspan="8">0-4</td><td colspan="2">Define which internal register drives output. Acceptable values for these bits are as follows:</td></tr>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>0</td><td>Track bits in Event Status Register (0xA0)</td></tr>
<tr><td>1</td><td>Track bits in Latched Event Status Register (0xA1)</td></tr>
<tr><td>2</td><td>Track bits in Manual Output Control Register. See Output States and Program Control (0xAB)</td></tr>
<tr><td>3</td><td>Track bits in Trajectory Status Register (0xC9)</td></tr>
<tr><td>4</td><td>Go active if position is between the two positions specified in words 2, 3 (low) and 4, 5 (high). If bit 14 is set, commanded position is used.
If bit 14 is clear, actual position is used.</td></tr>
<tr><td>5</td><td>Go active on low to high crossing of position specified by words 2, 3. Stay high for number of ms specified by words 4, 5. If bit 14 is set,</td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
						commanded position is used. If bit 14 is clear, actual position is used.
					6	Same as 5, but for high-to-low crossings
					7	Same as 5 but for any crossing
					8	Go active if motor phase angle (plus an offset) is between 0 and 180 degrees. Offset is set using first word of extra data in units of degrees.
					9	Pulse output each time a position is crossed from an array of positions stored in trace memory
					10	Use output to trigger an external regen resister
					11	For EtherCAT drives, pulse on SYNC0 signal
					12	For EtherCAT drives, go active when an EtherCAT frame is being received.
					13	Track bits in the capture status register. Words 2 & 3 give the bit mask of bits to track. If bit 14 of the first word is set, then the tracked bits are automatically cleared when the output goes active. In this case, words 4 & 5 can be used to give an optional pulse duration in ms. A zero in words 4&5 causes a pulse 1 servo cycle long.
					16	Track Hardware Position Compare function on drives supporting it.
					17	Logical OR of function 0 and 2. Output will track both a set of selected Event Status Register (0xA0) bits and Output States and Program Control (0xAB). Bits 14 and 15 of configuration also effect operation. If any of selected Event Status bits are set then output is active (if bit 14 is clear) or inactive (if bit 14 is set). If selected Event Status bits aren't active, then if Output States and Program Control (0xAB) bit is set then output is either active (bit 15 is clear) or inactive (bit 15 is set). If neither of those conditions is true, then output is either active (if bit 15 is set) or inactive (if bit 15 is clear).
					18	Brake PWM foldback. Firmware 2.98 and later. FPGA Plus drives only support this special mode in which output is configured as a brake which goes active for programmable time after which it starts to PWM with programmable on and off times. Word 2 of output configuration gives PWM on time in microseconds. Word three

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
						gives PWM period in microseconds. Word four is reserved, word five gives delay before PWM starts in ms.
					19	EDM (External Device Monitor). Output is active if drive is being disabled by STO input.
					20	PWM Brake. This configuration is used to control a brake output which PWMs to control the voltage applied to the brake. The four 16-bit parameters used to configure this output give the initial voltage (in 0.1V units), the continuous voltage, the time (ms) to output initial voltage and the PWM frequency in Hz. Not all output pins support this mode, any output that doesn't will just act as a normal brake if configured this way. ARM firmware 1.78 added a new option to this mode which allows the PWM duty cycle to be directly set by if bit 12 of the config word is set. In this mode the two voltages are replaced with duty cycles in 0.1% units, i.e. 500 would be 50%.
					21	This is similar to output configuration 20, but is manually controlled rather than controlled as a brake output.
				5-7	Reserved	
				8	If set, inverts normal active state of output. E.g., outputs that are normally active low become active high. For programmed controls, see Output States and Program Control (0xAB). If using hardware position triggered output feature (bits 0-4=16), see Output Compare Configuration Module. For software triggered output at position see Output Configuration (x70).	
				9	If set, program output as sync output. This bit is reserved for all output pins except pin 0.	
				10-11	Reserved	
				12-13	Axis number for multi-axis drives	
				14-15	Usage depends on output function selected	
				Version 4.80 firmware added several advanced output pin configurations which required more parameter data. At that point, support for optional 5-word configuration was added to firmware. For these output pin configurations, words 2 and 3 define one 32-bit parameter and words 4 and 5 define second 32-bit parameter.		
0x71	0x2193:2	RF	See text	Output 1 (OUT2) Configuration. See Output 0 (OUT1) Configuration (0x70).		

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
0x72	0x 2193:3	RF	See text	Output 2 (OUT3) Configuration. See Output 0 (OUT1) Configuration (0x70).
0x73	0x 2193:4	RF	See text	Output 3 (OUT4) Configuration. See Output 0 (OUT1) Configuration (0x70).
0x74	0x 2193:5	RF	See text	Output 4 (OUT5) Configuration. See Output 0 (OUT1) Configuration (0x70).
0x75	0x 2193:6	RF	See text	Output 5 (OUT6) Configuration. See Output 0 (OUT1) Configuration (0x70).
0x76	0x 2193:7	RF	See text	Output 6 (OUT7) Configuration. See Output 0 (OUT1) Configuration (0x70).
0x77	0x 2193:8	RF	See text	Output 7 (OUT8) Configuration. See Output 0 (OUT1) Configuration (0x70).
0x78	0x 2192:1	RF	U16	Input 0 (IN1) Configuration. Assigns function to input pin. All values not listed below are reserved for future use. For notes on Input numbering, See Input/Output Numbering. Sync Input function is only valid for high-speed input pins. In addition, input pins 2 & 3 of Accelus and Junus drives do not support this feature. The lower 8 bits define the input pin function:

Bits	Configuration	
0-7	**Value**	**Meaning**
	0	No function
	1	Reserved (no function)
	2	Reset drive on rising edge of input.
	3	Reset drive on falling edge of input.
	4*	Positive limit switch. Active high.
	5*	Positive limit switch. Active low.
	6*	Negative limit switch. Active high.
	7*	Negative limit switch. Active low.
	8*	Motor temperature switch. Active high.
	9*	Motor temperature switch. Active low.
	10*	Clear faults on rising edge, disable drive while high.
	11*	Clear faults on falling edge, disable drive while low.
	12*	Reset on rising edge, disable drive while high.
	13*	Reset on falling edge, disable drive while low.
	14*	Home switch. Active high.
	15*	Home switch. Active low.
	16*	Drive disable. Active high
	17*	Drive disable. Active low.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
					18	Sync input on rising edge. If bit 8 is set, pin switch debounce time is used as sync offset in 0.1 us units.
					19	Sync input on falling edge. If bit 8 is set, pin switch debounce time is used as sync offset in 0.1 us units.
					20*	Halt motor. Active high.
					21*	Halt motor. Active low.
					22	Scale analog input. Active high.
					23	Scale analog input. Active low.
					24*	High-speed position capture on rising edge. Only for high-speed inputs.
					25*	High-speed position capture on falling edge. Only for high-speed inputs.
					26	Count rising edges of input to indexer register. Register number identified by bits 8-11.
					27	Count falling edges of input to indexer register. Register number identified by bits 8-11.
					28*	Encoder fault input. Active high.
					29*	Encoder fault input. Active low.
					30-35	Reserved
					36	Abort move on rising edge if greater than *n* counts from destination position. Number of counts *n* is stored in an index register identified by bits 8-11.
					37	Abort move on falling edge if greater than *n* counts from destination position. Number of counts *n* is stored in an index register identified by bits 8-11.
					38*	Mark HV loss on rising edge, disable while high.
					39*	Mark HV loss on falling edge, disable while low.
					40*	Update trajectory on rising edge.
					41*	Update trajectory on falling edge.
					42*	Clear faults & event latch on rising edge.
					43*	Clear faults & event latch on falling edge.
					44*	Disable simulated encoder output when low. Burst current position on encoder output on rising edge.
					45*	Disable simulated encoder output when high. Burst current position on encoder output on falling edge.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
				46 Disable drive and act like safety input is active when high. Additionally, bits 8-11 of configuration word are set in Safety Status Register (0x139) bits 0-3. This input type is intended for custom hardware that implements a STO circuit external to drive.
				47 Like input type 46, but active low.
				8-11 Used to pass parameters to input pin functions.
				12-13 Used to select axis on multi-axis drives.
				* Input functions above use bit 8 to indicate that the input function should apply to all axes. This feature is enabled in FPGA Plus drives starting with version 1.72 firmware.
0x79	0x2192:2	RF	U16	Input 1 (IN2) Configuration. See Input 0 (IN1) Configuration (0x78).
0x7A	0x2192:3	RF	U16	Input 2 (IN3) Configuration. See Input 0 (IN1) Configuration (0x78).
0x7B	0x2192:4	RF	U16	Input 3 (IN4) Configuration. See Input 0 (IN1) Configuration (0x78).
0x7C	0x2192:5	RF	U16	Input 4 (IN5) Configuration. See Input 0 (IN1) Configuration (0x78).
0x7D	0x2192:6	RF	U16	Input 5 (IN6) Configuration. See Input 0 (IN1) Configuration (0x78).
0x7E	0x2192:7	RF	U16	Input 6 (IN7) Configuration. See Input 0 (IN1) Configuration (0x78).
0x7F	0x2192:8	RF	U16	Input 7 (IN8) Configuration. See Input 0 (IN1) Configuration (0x78).
0x80	0x6503	F*	String	Drive Model Number.
0x81	0x2384:1 or, 0x1018:4	F*	U32	Drive Serial Number.
0x82	0x2384:3	F*	INT16	Drive's rated Peak Current. Units: 0.01 A.
0x83	0x2384:4	F*	INT16	Drive's rated Continuous Current. Units: 0.01 A
0x84	0x2384:14	F*	INT16	Current Corresponding to Drive's Max A/D Reading. Units: 0.01 A.
0x85	0x2384:11	F*	U16	PWM Period (Current loop update rate). Units: 10 ns.
0x86	0x2384:12	F*	U16	Drive Servo Period (Position and velocity loop update rate). Units: Multiple of PWM Period (0x85).
0x87	None	F*	U16	Product Family. Identifies the drive product family. For specific drive hardware type, see Drive Hardware Type (0xAD).
0x88	0x2384: 5	F*	INT16	Drive's rated Time at Peak Current. Units: ms. (Default: 1000ms). Maximum 10 seconds.
0x89	0x2384:6	F*	INT16	Drive's rated Maximum Voltage. Units: 0.1 V. Maximum bus voltage rating. When HV (high voltage) is greater than the drive's maximum rated voltage the drive goes into overvoltage shutdown.
0x8A	0x2384:15	F*	INT16	Voltage Corresponding to HV Max A/D Reading.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
				Units: 0.1 V.
0x8B	0x2384:7	F*	INT16	Drive’s rated Minimum Voltage. Units: 0.1 V. Minimum bus voltage rating. When HV (high voltage) is less than the drive’s minimum rated voltage the drive goes into undervoltage shutdown.
0x8C	0x2384:9	F*	INT16	Drive’s rated Maximum Temperature. Units: degrees C. Range 0 to 100.
0x8D	0x2384:2	F*	String	Manufacturing info (date code) of drive. First two digits correspond to week and last two digits correspond to year.
0x8E	0x2384:16	F*	INT16	Analog Input Reference Scaling Factor. This is voltage applied to analog input which causes max A/D value on drive. Units: mV
0x90	None	R	U32	Serial Port Baud Rate. Units: bits/s. Defaults to 9600 at power up or reset.
0x91	None	R*	INT16	Maximum number of data words allowed per binary command over serial interface.
0x92	0x21A0	F	String	Axis label string (drive name).
0x93	None	F	U32	Reserved.
0x94	0x2384:24	R*	INT16	Firmware Version Number. Version number consists of major and minor version number. Minor number passed in bits 0-7; major number passed in bits 8-15. E.g. version 1.12 would be encoded 0x010C.
0x95	0x2421	F	String	Host Configuration State. Reserved for use by CME software.
0x96	0x2312	RF	INT16	Calibration Offset for Analog Input or Analog Reference. This voltage is added to analog reference input and is calibrated at factory to give zero reading for zero input voltage.
0x97	0x2384:10	F*	INT16	Hysteresis value for drive over temperature cut-out. Units: degrees C.
0x98	0x2330	RF	INT16	Function Generator Configuration. Configures drive’s internal function generator which drives current, velocity, or position loop. Bit-mapped as follows:

Bits	Description	
0-2	Function code (type of waveform to generate):	
	Value	**Description**
	0	None (disabled)
	1	Square wave output
	2	Sine wave output
	3	White noise (Plus & AFS products)
	4	Triangular waveform (Plus & AFS products)
3	Reserved	
4-5	Function generator injection into running loop. Allows output of function generator to be injected into input of either current or velocity loop while drive is operating in some mode of operation other than function generator mode. This feature is only available on Plus product	

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
					drives starting with firmware 3.34. This can be useful for testing system response in presence of a disturbance.	
					Mode	**Description**
					0	No function generator injection
					1	Inject function generator output into input of current loop
					2	Inject function generator output into input of velocity loop
					3	Reserved
				6-7	Reserved.	
				8	If set, use high resolution mode. In this mode Function Generator Frequency (0x99) is in units of 0.01 Hz. Plus & AFS products.	
				9-11	Reserved	
				12	If set, one shot mode. After one period function type resets to zero.	
				13	If set, invert every other period. After two periods function type resets to zero.	
				14-15	Reserved	
				Note drive is placed in function generator mode by setting Desired State (0x24).		
				4	(function generator drives current loop)	
				14	(function generator drives velocity loop)	
				24	(function generator drives position loop in servo- mode)	
				34	(function generator drives position loop in stepper mode).	
0x99	0x2331	RF	U16	Function Generator Frequency. Units: Hz. Plus & AFS products support high-resolution mode. Units: 0.01 Hz. See bit 8 of Function Generator Configuration (0x98).		
0x9A	0x2332	RF	INT32	Function Generator Amplitude. Amplitude of signal generated by internal function generator. Units depend on operating mode:		
				Mode	**Units**	
				Current	0.01 A.	
				Velocity	0.1 encoder counts/s.	
				Position	Encoder counts.	
0x9B	0x2333	RF	U16	Function Generator Duty Cycle (square wave only). Units: 0.1% Range 1 to 1000(100%).		
0x9C	0x2384:8	F*	U16	Hysteresis for Maximum Bus Voltage Cut-Out. Units: 0.1 V.		
0x9D	0x2384:18	F*	U16	PWM Dead Time at Continuous Current Limit.		

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td>Units: CPU cycles. Factory setting.
This parameter gives PWM dead time used at or above continuous current limit. Dead time below continuous current limit is defined by linear function of this parameter and PWM Dead Time at Zero Current (0x9F).</td></tr>
<tr><td>0x9E</td><td>0x2384:17</td><td>F*</td><td>U16</td><td>Drive Minimum PWM Off Time. Units: 10 ns. Factory Setting.
This parameter gives minimum amount of time for which all PWM outputs must be disabled for each current loop cycle.</td></tr>
<tr><td>0x9F</td><td>0x2384:19</td><td>F*</td><td>U16</td><td>PWM Dead Time at Zero Current. Units: CPU cycles. Factory setting.
This parameter gives PWM dead time at zero current. Dead time above zero current is defined by linear function of this parameter and PWM Dead Time at Continuous Current Limit (0x9D).</td></tr>
<tr><td>0xA0</td><td>0x1002</td><td>R*</td><td>U32</td><td>Event Status Register.
Bit-mapped as follows:
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0</td><td>Short circuit detected</td></tr>
<tr><td>1</td><td>Drive over temperature</td></tr>
<tr><td>2</td><td>Over voltage</td></tr>
<tr><td>3</td><td>Under voltage</td></tr>
<tr><td>4</td><td>Motor temperature sensor active</td></tr>
<tr><td>5</td><td>Feedback error or Encoder power error</td></tr>
<tr><td>6</td><td>Motor phasing error</td></tr>
<tr><td>7</td><td>Current output limited</td></tr>
<tr><td>8</td><td>Voltage output limited</td></tr>
<tr><td>9</td><td>Positive limit switch active</td></tr>
<tr><td>10</td><td>Negative limit switch active</td></tr>
<tr><td>11</td><td>Enable input not active</td></tr>
<tr><td>12</td><td>Drive is disabled by software (desired state is 0)</td></tr>
<tr><td>13</td><td>Trying to stop motor</td></tr>
<tr><td>14</td><td>Motor brake activated</td></tr>
<tr><td>15</td><td>PWM outputs disabled</td></tr>
<tr><td>16</td><td>Positive software limit condition</td></tr>
<tr><td>17</td><td>Negative software limit condition</td></tr>
<tr><td>18</td><td>Tracking (Following) Error Fault. A tracking (following) error has occurred, and drive is in tracking (following) error mode.</td></tr>
<tr><td>19</td><td>Tracking (Following) Error Warning. Indicates position error is greater than position tracking (following) warning.</td></tr>
<tr><td>20</td><td>Drive is currently in reset condition</td></tr>
</table>
</td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				21	Position has wrapped. Position variable cannot increase indefinitely. After reaching a certain value the variable rolls back. This type of counting is called position wrapping or modulo count
				22	Drive fault. Fault configured as latching in Fault Mask (0xA7) has occurred. Latched faults may be cleared using Latching Fault Status Register (0xA4).
				23	Velocity limit (0x3A) has been reached
				24	Acceleration limit (0x36) has been reached
				25	Position Tracking. Position Loop Error (0x35) is outside of Tracking (Following Error Fault Limit (0xBA).
				26	Home switch is active
				27	In motion. Bit is set if trajectory generator is running profile or Tracking (Following Error Fault Limit (0xBA) is outside tracking window. Clear when drive is settled in position.
				28	Velocity window. Set when velocity error is larger than programmed velocity window
				29	Phase not yet initialized. This bit is set until drive has initialized its phase. Drive is performing algorithmic phasing, or phase initialization has failed.
				30	Command fault. CANopen or EtherCAT master not sending commands in time as configured by the master, or PWM command not present. CANopen: Master configures guarding parameters 0x10C, 0x10D, 0x10E. EtherCAT: Master configures sync master. PWM: If *Allow 100% Output* option is enabled by setting Bit 3 of Digital Input Command Configuration (0xA8) this fault will not detect missing PWM command.
				31	Reserved.
0xA1	0x2181	R	U32	Latched Event Status Register. This is latched version of Event Status Register (0xA0). Bits are set by drive when events occur. Bits are only cleared by writing to this parameter as explained below: When writing to Latched Event Status Register, any bit set will cause corresponding bit in register to be cleared. For example, to clear latched event of over voltage, write decimal 4 or 0x04 to parameter 0xA1. To clear all bits, write 0xFFFFFFFF to parameter 0xA1.	
0xA2	0x2261	R*	INT16	Hall Input State. Lower three bits of returned value give present state of Hall input pins. Hall state is value of Hall lines AFTER ordering and inversions specified in Hall Wiring Configuration (0x52) have been applied.	
0XA3	None	R	U32	Drive test parameter. This parameter is reserved for use by Copley during drive test.	
0xA4	0x2183	R	U32	Latching Fault Status Register. Bit-mapped to show which latching faults have occurred in drive. When	

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
				latching fault has occurred, the *fault bit (bit 22)* of Event Status Register (0xA0) is set. Cause of fault can be read from this register. To clear fault condition, write a 1 to associated bit in this register. Events that cause drive to latch fault are programmable. See Fault Mask (0xA7) for details.
				Latched Faults

Bits	Fault Description
0	Data flash CRC failure. This fault is considered fatal and cannot be cleared. This bit is read-only and will remain latched. If drive detects corrupted flash data values on startup it will remain disabled and indicate fault condition.
1	A/D offset out of range (fatal fault). Drive internal error. This bit is read-only and will remain latched. If drive fails its power-on self-test, it will remain disabled and indicate fault condition.
2	Short circuit. If set: programs drive to latch a fault when short circuit is detected on motor outputs. If clear: programs drive to disable outputs for 100ms after short circuit and then re-enable.
3	Drive over temperature. If set: programs drive to latch a fault when drive over temperature event happens. If clear: programs drive to re-enable as soon as it cools sufficiently from over temperature event.
4	Motor over temperature. If set: programs drive to latch a fault when motor temperature sensor input activates. If clear: programs drive to re-enable as soon as over temperature input becomes inactive.
5	Over-voltage. If set: programs drive to latch a fault when excessive bus voltage is detected. If clear: programs drive to re-enable as soon as bus voltage is within normal range.
6	Under-voltage. If set: programs drive to latch a fault condition when inadequate bus voltage is detected. If clear: programs drive to re-enable as soon as bus voltage is within normal range.
7	Feedback fault. If set: programs drive to latch a fault when feedback faults occur. Feedback faults occur if too much current is drawn from 5 V source on drive, resolver or analog encoder is disconnected, or resolver or analog encoder has levels out of tolerance.
8	Phasing error. If set: programs drive to latch a fault when phasing errors occur.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
					If clear: programs drive to re-enable when phasing error is removed.
				9	Following error. If set: programs the drive to latch a fault and disable drive when following error occurs. If clear: programs drive to abort current move and remain enabled when following error occurs.
				10	If set: programs drive to latch a fault when output current is limited by I^2T algorithm.
				11	FPGA failure. This bit is read-only.
				12	Command input lost fault. If set: programs drive to latch a fault and disable when command input is lost.
				13	Unable to initialize internal drive hardware. This bit is read-only.
				14	If set, programs drive to latch a fault when there is safety circuit consistency check failure.
				15	If set, programs drive to latch a fault when drive is unable to control motor current.
				16	If set, programs drive to latch a fault when motor wiring is disconnected, see Open Motor Wiring Check Current (0x19D).
				17	Reserved.
				18	Safe torque off active
0xA5	0x2191	RF	U16	Input Pin Configuration Register. Some drives have one or more pull-up resistors associated with their general-purpose input pins. On these drives, state of pull-ups can be controlled by writing to this register. This register has one bit for each pull-up resistor available on drive. Setting bit causes resistor to pull any inputs connected to it up to high state when they are not connected. Bit 0 controls first pullup resistor on drive, bit 1 controls second pullup resistor, etc. Please refer to drive datasheet to determine how many pullup resistors are available for particular drive. On drives that allow groups of inputs to be configured as either single ended or differential, bit 8 controls this feature. Set bit 8 to 0 for single ended, 1 for differential. See also Input Pin Configuration Register, 32-Bit (0x15E) for newer drives which support more than 16 input pins.	
0xA6	0x2190	R*	U16	Input Pin States. The 16-bit value returned by this command gives current state (high/low) of drive's input pins after switch debounce. Each bit represents one input as shown below. See also Input Pin States, 32-Bit (0x15C) for newer drives which support more than 16 input pins.	
				Bits	**Description**

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				0	Programmable input pin 0 (IN1)
				1	Programmable input pin 1 (IN2)
				2	Programmable input pin 2 (IN3)
				3	Programmable input pin 3 (IN4)
				4	Programmable input pin 4 (IN5)
				5	Programmable input pin 5 (IN6)
				6	Programmable input pin 6 (IN7)
				7	Programmable input pin 7 (IN8)
				8	Programmable input pin 8 (IN9)
				9	Programmable input pin 9 (IN10)
				10	Programmable input pin 10 (IN11)
				11	Programmable input pin 11 (IN12)
				12	Programmable input pin 12 (IN13)
				13	Programmable input pin 13 (IN14)
				14	Programmable input pin 14 (IN15)
				15	Programmable input pin 15 (IN16)
0xA7	0x2182	RF	U32	Fault Mask. This variable is used to configure which drive events cause latching faults. For drive events see Latching Fault Status Register (0xA4). Setting fault mask bit to 1 causes associated drive event to cause latching fault when it occurs. Setting fault mask bit to 0 disables fault latching on associated event. Latched faults may be cleared using Latching Fault Status Register (0xA4).	
0xA8	0x2320	RF	INT16	Digital Input Command Configuration. Defines configuration of digital input commands when drive is running in a mode that uses them as a control source. The lower 8 bits control PWM input configuration for controlling current and velocity modes. Upper 8 bits configure digital inputs when running in position mode.	
				Bits	**Description**
				0	If set, use PWM in signed/magnitude mode. If clear, use PWM in 50% duty cycle offset mode.
				1	If set, invert the PWM input.
				2	If set, invert the signed input.
				3	If set, allow 100% duty cycle. If clear, treat 100% duty cycle as zero command, providing measure of safety in case of controller failure or cable break.
				4	If set, use PWM Input Frequency (0xB6) as deadband for PWM input. Note: Some newer products have dedicated parameter, PWM Input Deadband (0x13F) to hold deadband.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
				5	If set, allow longer PWM periods (up to 50ms).	
				6	For 8367 DSP products, setting this bit will cause Motor Hall Offset (0x4F) to be added to angle calculated in UV mode. For Plus & AFS products, see UV Configuration (0x180).	
				7	Reserved	
				8-9	Input pin interpretation for position mode (see below). Specifies the type of input signals. These bits should hold one of the following values:	
					Value	**Description**
					0	Step (Pulse) & Direction mode.
					1	Separate Pulse up & down counters.
					2	Quadrature encoder input.
					3	PWM input commands absolute position.
				10-11	Reserved	
				12	If set, pulses are counted on rising edge. If clear, pulses are counted on falling edge. This bit has no effect when inputs are configured as encoder inputs.	
				13	Causes direction of input to be reversed. Works for all three modes.	
				14-15	Identify which input pins to use. Input choices only valid for drives that support such inputs:	
					Value	**Description**
					0	Single ended high-speed inputs.
					1	Multi-mode encoder port.
					2	Differential high-speed inputs.
					3	Use primary encoder inputs.
0xA9	0x2321	RF	INT32	Digital Input Scaling Factor. This value gives amount of command at 100% PWM input. Scaling depends on what PWM input is driving: Current Mode Units: 0.01 A Estimated Velocity Mode Units: 0.01 RPM Velocity Mode Units: 0.1 encoder counts/second Position Mode Units: count ratio (output/input) In position mode scaling factor is a ratio of two 16-bit values. First word passed gives numerator and second word gives denominator. This ratio determines number of encoder units moved (output) for each pulse or encoder count (input). For example, a ratio of 1/3 would cause motor to move 1 encoder unit for every three input steps. When running in PWM position mode, scaling factor is single 32-bit integer which gives range of commanded		

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td>position in encoder counts. Minimum PWM Pulse Width (0x13C) corresponds to an absolute position of 0, Maximum PWM Pulse Width (0x13D) corresponds to an absolute position equal to this scaling factor.
Additionally, an offset position may be added using Registration Offset for Pulse & Direction Mode (0x10F).</td></tr>
<tr><td>0xAA</td><td>0x2196</td><td>R*</td><td>U16</td><td>Raw Input State. 16-bit value returned by this command gives current state (high/low) of drive’s input pins. Unlike Input Pin States (0xA6), no switch debounce is applied when reading inputs using this variable.
Bits are mapped in same order as Input Pin States (0xA6).
See also Raw Input Pin States, 32-Bit (0x15D) for newer drives which support more than 16 input pins.</td></tr>
<tr><td>0xAB</td><td>0x2194</td><td>R</td><td>U16</td><td>Output States and Manual (Program) Control. When read, this parameter gives active/inactive state of drive’s general-purpose digital outputs. Each bit represents an output number. Bit 0 = digital Output 0 (OUT1), bit 1 = digital Output 1 (OUT2), etc., up to output n (OUT(<i>n+1</i>)), number of digital outputs on drive. Additional bits are reserved, consult factory.
Outputs that have not been configured for external register control can be manually set by writing to output configuration parameter (0x70 - 0x77). Set bit to activate output. It will be activated high or low according to how it was programmed (Bit 8 of 0x70-0x77). Clear bit to make output inactive.
If an output was configured for internal register control, it will not be affected.</td></tr>
<tr><td>0xAC</td><td>0x2180</td><td>R*</td><td>U32</td><td>Sticky Drive Event Status Register. This read-only parameter is bit-mapped in exactly same way as Event Status Register (0xA0), but instead of giving present status of drive, sticky version indicates any bits in event status that has been set since last reading of sticky register.
Sticky register is similar to Latched Event Status Register (0xA1), but latched register must be cleared explicitly, whereas sticky register is cleared automatically each time it is read.</td></tr>
<tr><td>0xAD</td><td>0x1018:2
or
0x2384:13</td><td>F*</td><td>INT16</td><td>Drive Hardware Type. Also known as Product Code. Identifies specific drive model. This is an augmented version of Product Family (0x87).
<table>
<tr><th>Value (HEX)</th><th>Value (DEC)</th><th>Product</th></tr>
<tr><td>0x0000</td><td>000</td><td>ASC Accelus Card</td></tr>
<tr><td>0x0001</td><td>001</td><td>ASP Accelus Panel without pullup/pulldown on inputs (Obsolete)</td></tr>
<tr><td>0x0002</td><td>002</td><td>ASP Accelus Panel with pullup/pulldown on input pins</td></tr>
<tr><td>0x0100</td><td>256</td><td>JSP Junus Panel</td></tr>
<tr><td>0x0200</td><td>512</td><td>ACM Accelnet Module</td></tr>
<tr><td>0x0201</td><td>513</td><td>XSL Xenus Panel (Obsolete)</td></tr>
</table></td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
				0x0204	516	XSL-R Xenus Panel Resolver (Obsolete)
				0x0206	518	XSL-R Xenus Panel Resolver (Legacy)
				0x0207	519	XSL Xenus Panel (Legacy)
				0x0209	521	ACJ Accelnet Micro Panel
				0x0210	528	ACJ-S Accelnet Micro Panel Sin/Cos
				0x020C	524	ACK Accelnet Micro Module
				0x0240	576	STM Stepnet Module
				0x0242	578	STP Stepnet Panel
				0x0243	579	STL Stepnet Micro Module
				0x0300	768	ASP-X2 2-axis Accelus Panel (Obsolete)
				0x0310	784	XSJ Xenus Micro (8367DSP Obsolete)
				0x0314	788	XSJ Xenus Micro (ARM) AFS
				0x0320	800	XTL-R Xenus Resolver (8367DSP Obsolete)
				0x0330	816	XTL Xenus (8367DSP Obsolete)
				0x0331	817	Custom version of XTL prototype
				0x0334	820	XTL Xenus (ARM) AFS
				0x0340	832	XSJ-R Xenus Micro Resolver (8367DSP Obsolete)
				0x0350	848	STX Stepnet AC (8367DSP Obsolete)
				0x0351	849	STX Stepnet AC (8367DSP Obsolete)
				0x0360	864	ACJ-R Accelnet Micro Panel Resolver (8367DSP Obsolete)
				0x0370	880	ACK-R Accelnet Micro Module Resolver (8367DSP Obsolete)
				0x0380	896	AEP Accelnet EtherCAT Panel (8367DSP Obsolete)
				0x0390	912	AMP Accelnet Macro Panel (8367DSP Obsolete)
				0x03A0	928	ADP Accelnet Panel (8367DSP Obsolete)
				0x03A4	932	ADP Accelnet Panel (ARM) AFS
				0x03B0	944	ST3 3-axis Stepnet (8367DSP Obsolete)
				0x03C0	960	800-1638 Custom drive (8367DSP Obsolete)
				0x03D0	976	ADP-R Accelnet Panel Resolver (8367DSP Obsolete)
				0x03E0	992	ACM-R Accelnet Module (8367DSP Obsolete)
				0x03F0	1008	ACK-H High current Accelnet Micro Module ARM

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
				0x0400	1024	CAN I/O Module (8367DSP Obsolete)
				0x0404	1028	CAN I/O Module ARM
				0x1000	4096	XEL Xenus Plus EtherCAT (Obsolete)
				0x1001	4097	XEL Xenus Plus EtherCAT
				0x1008	4104	XEL-R Xenus Plus EtherCAT Resolver (Obsolete)
				0x1009	4108	XEL-R Xenus Plus EtherCAT Resolver
				0x1010	4112	XML Xenus Plus MACRO
				0x1018	4120	XML-R Xenus Plus MACRO Resolver (Obsolete)
				0x1020	4128	XPL Xenus Plus CAN
				0x1028	4136	XPL-R Xenus Plus Resolver CAN
				0x1030	4144	AEM Accelnet Plus EtherCAT Module (Obsolete)
				0x1031	4145	AEM Accelnet Plus EtherCAT Module
				0x1040	4160	APM Accelnet Plus CAN module
				0x1050	4176	AE2 2-axis Accelnet Plus EtherCAT module
				0x1060	4192	AP2 2-axis Accelnet Plus CAN module
				0x1070	4208	SEM Stepnet Plus EtherCAT module
				0x1080	4224	SPM Stepnet Plus CAN module
				0x1090	4240	SE2 2-axis Stepnet Plus EtherCAT module
				0x10A0	4256	SP2 2-axis Stepnet Plus CAN module
				0x10B0	4272	XE2 2-axis Xenus Plus EtherCAT
				0x10B8	4280	XE2-R 2-axis Xenus Plus Resolver EtherCAT
				0x10C0	4288	BE2 2-axis Accelnet Plus EtherCAT Panel
				0x10C8	4296	BE2-R 2-axis Accelnet Plus Resolver EtherCAT Panel
				0x10D0	4304	XP2 2-axis Xenus Plus CAN
				0x10D8	4312	XP2-R 2-axis Xenus Plus Resolver CAN
				0x10E0	4320	BP2 2-axis Accelnet Plus EtherCAT Panel
				0x10E8	4328	BP2-R 2-axis Accelnet Plus Resolver CAN Panel
				0x10F0	4336	TE2 2-axis Stepnet Plus EtherCAT Panel
				0x1100	4352	TP2 2-axis Stepnet Plus CAN Panel
				0x1110	4368	BEL Accelnet Plus EtherCAT Panel

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
				0x1118	4376	BEL-R Accelnet Plus Resolver EtherCAT Panel
				0x1120	4384	BPL Accelnet Plus CAN Panel
				0x1128	4392	BPL-R Accelnet Plus Resolver CAN Panel
				0x1130	4400	TEL Stepnet Plus EtherCAT Panel
				0x1150	4432	SP4 4-axis Stepnet CAN Module
				0x1170	4464	XM2 2-axis Xenus Plus MACRO
				0x1178	4472	XM2-R 2-axis Xenus Plus Resolver MACRO
				0x1180	4480	BML Accelnet Plus MACRO
				0x1190	4496	SE4 4-axis Stepnet EtherCAT Module
				0x11B0	4528	XEC Xenus Plus Compact EtherCAT
				0x11B8	4536	XEC-R Xenus Plus Compact Resolver EtherCAT
				0x11C0	4544	XPC Xenus Plus Compact CAN
				0x11C8	4552	XPC-R Xenus Plus Compact Resolver CAN
				0x11D0	4560	ME3 3-axis Module EtherCAT
				0x11E0	4576	MP3 3-axis Module CANopen
				0x11F0	4592	ME4 4-axis Module EtherCAT
				0x1200	4608	MP4 4-axis Module CANopen
				0x1240	4672	GEM Argus Plus EtherCAT Module
				0x1248	4680	GEM-R Argus Plus EtherCAT Resolver
				0x1250	4688	GPM Argus Plus CAN Module
				0x1258	4696	GPM-R Argus Plus CAN Resolver
				0x1260	4704	AEV Accelnet Plus Micro EtherCAT Module
				0x1270	4720	APV Accelnet Plus Micro CAN Module
				0x1280	4736	NEP Nano Plus EtherCAT
				0x12C0	4800	NPP Nano Plus CAN
				0x2050	8272	IES Integrated Servo Drive
				0x2070	8304	NPS Nano CAN
				0x2080	8320	NES Nano EtherCAT
0xAE	0x60F6:3	RF	INT16	Current Loop Offset. Units: 0.01 A. This value is added to commanded current. It can compensate for directional bias affecting current loop, such as gravity.		
0xAF	0x2420	RF	INT32	Miscellaneous Drive Options Register. This register allows various drive options to be selected. Bit-mapped as follows:		
				Bits	**Option**	

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
				0	If set, input pins 1, 2 and 3 are pulled high on drive. If clear, pins are not pulled up. Only available on Junus drive.	
				1	Reserved	
				2	If set, limit switch inputs will only abort trajectory in progress but will not affect current output. If clear, limit switches limit current.	
				3	If set, save PDO configuration to file in CVM file system when “Save to Flash” command is received over CANopen network. If clear, PDO is not saved.	
				4	If set, limit switch activation will be treated as fault in CANopen Status Word (CANopen index 0x6041 as described in *CANopen Programmer’s Manual*).	
				5-6	When encoder wrap is enabled, these bits control direction of motion for absolute moves in trapezoidal and S-curve profile modes.	
					Value	**Mode**
					0	Move in the shortest direction.
					1	Always move in positive direction.
					2	Always move in negative direction.
					3	Reserved
				7	If set, analog command values will use digital data written to an SPI serial peripheral interface connected to drive input pins & multimode port. This is available on some Plus drives for use in digitally interfacing with a Delta Tau controller.	
				8	If set, brake delay will be applied even in case of latching faults.	
				9	If set, voltage and current warnings are disabled.	
				10-31	Reserved	
0xB0	0x2260	R	INT16	Motor Phase Angle. Units: degrees. Writes are only useful when running in diagnostic microstepping mode.		
0xB1	0x21C1	RF	INT16	Increment Rate for Phase Angle When in microstepping Mode. Units: degrees/s. Only used in diagnostic mode. Desired State (0x24) = 42 (microstepping mode).		
0xB2	0x21C0	RF	U16	Commutation Mode (Phasing Mode). Configures mechanism by which the drive computes the motor phase angle. Determines what method the drive uses to initialize and maintain phase angle. Bit-mapped as follows:		
				Bits	**Mode**	

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				0	Standard Mode. Encoder-based sinusoidal commutation for brushless motors. Use digital Hall inputs (commutating encoder) to initialize phase, then switch to an encoder to maintain phase. Encoder is primary sensing device with Hall Effect sensors used to monitor and adjust phase angle as necessary during operation.
				1	Trapezoidal (Hall based) phasing. Hall Effect sensors are used for phasing at all times. This mode can be used if no encoder is available.
				2	Like mode 0 except that phase angle is not adjusted based on Hall inputs. Hall Effect sensors are still required to initialize phase angle at startup.
				3	Analog Halls (90 degrees). Only available on drives with necessary analog sine/ cosine inputs.
				4	DC brush motor mode. Note preferred way to configure an axis to drive DC brushed motor is by setting Motor Type (0x40). This method will continue to be supported for backward compatibility.
				5	Algorithmic Phase Initialization mode (wake & wiggle, no Halls). See *CME User Guide* for more information on Algorithmic Phase Initialization.
				6	Use with resolver or Servo-Tube motors. To determine the absolute position within the electrical cycle for phasing, much like encoder sinusoidal commutation.
				7	Trapezoidal commutation with phase angle interpolation (Estimated Sinusoidal).
				8	Reserved
				9	Manual phasing. Phase angle set to know position before enable. Commutation mode 9 is used in cases where the initial phase angle is known after power-up or reset and can be written to the drive before enable. In this mode we write to motor phase angle (0xB0) on startup after reading the absolute position from some external device such as absolute encoder, potentiometer, switch, or other method that provides a known physical position. As the motor moves, the drive will use the position from the incremental encoder count on the motor to update the phase angle.
0xB3	0x2384:23	F*	INT16	Analog Encoder Scaling Factor. This parameter selects resolution of analog encoder input. Parameter not used for other encoder types.	
0xB4	0x2263	R*	INT16	Motor Phase Angle. For feedback types that perform brushless commutation and generate phase angle	

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
				information. This parameter allows phase information to be read directly.
0xB5	0x2353	R*	INT32	Homing Adjustment. Units: counts. This parameter is updated after each successful homing operation. Value contained is size of actual position adjustment made in last home sequence.
0xB6	0x2322	RF	U16	PWM Input Frequency. This is frequency of PWM for use in UV commutation mode only. Units: 10 Hz. This parameter is also used to specify an optional PWM dead band when running in normal (not UV) PWM command modes. When used as deadband value, this input should be set in range 0 to 32767 which corresponds to deadband of 0 to 100% of PWM duty cycle. On Plus and AFS models, PWM Input Deadband (0x13F) is dedicated to holding PWM Input Deadband value. On products supporting that parameter, writing to this parameter will still modify deadband setting for backward compatibility but use of PWM Input Deadband (0x13F) is recommended.
0xB7	0x2141	R*	U32	System Time. Time since last start up (power-up or reset). Units: ms.
0xB8	0x607D:2	RF	INT32	Positive Software Limit value. Units: counts. This parameter is only available on drives that support trajectory generation and homing. Software limits are only in effect after drive has been referenced (i.e. homing has been successfully completed). Set to less than negative software limit to disable.
0xB9	0x607D:1	RF	INT32	Negative Software Limit. Units: counts. Software limits are only in effect after drive has been referenced (i.e. homing has been successfully completed). Set to greater than positive software limit to disable.
0xBA	0x2120	RF	INT32	Following Error Fault Limit. Units: counts. If Position Loop Error (0x35) exceeds this value then following error (bit 18*)* of Event Status Register (0xA0) is set and motor is stopped. Using Fault Mask (0xA7), following error event can be configured to either disable drive immediately or abort present move and continue holding position.
0xBB	0x6065	RF	INT32	Following Error Warning Limit. Units: counts. If Position Loop Error (0x35) exceeds this value then following warning (bit 19*)* of Event Status Register (0xA0) is set.
0xBC	0x6067	RF	INT32	Position Tracking Window Limit. Units: counts. If Position Loop Error (0x35) exceeds this value then tracking window (bit 25) of Event Status Register (0xA0) is set.
0xBD	0x6068	RF	U16	Time Delay For Following Error Fault Limit (0xBA). Units: ms

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="2">Tracking window (bit 25) of Event Status Register (0xA0) will not be cleared until Position Loop Error (0x35) has been within Following Error Fault Limit (0xBA) for at least this amount of time.</td></tr>
<tr><td>0xBE</td><td>0x2253</td><td>RF</td><td>U32</td><td colspan="2">Deceleration limit used with software limits. Set to 0 for non-trajectory-based software limits.</td></tr>
<tr><td>0xBF</td><td>0x2351</td><td>RF</td><td>U16</td><td colspan="2">Home to Hard Stop Delay Time. Units: ms.
When performing home to hard stop, drive will push against stop for this long before sampling the home position.</td></tr>
<tr><td>0xC0</td><td>None</td><td>R*</td><td>INT16</td><td colspan="2">CAN Network Node ID. This is drive's present ID as read at system startup. Node ID is only read at system startup, so this value will not change unless drive is reset. See CAN Network Node ID Configuration (0xC1).</td></tr>
<tr><td rowspan="5">0xC1</td><td rowspan="5">0x21B0</td><td rowspan="5">RF</td><td rowspan="5">INT16</td><td colspan="2">CAN Network Node ID Configuration.
Defines how drive's Node ID is calculated and specifies drive's network bit rate. Node ID is calculated at startup (and only at startup) using a combination of general-purpose input pins and programmed offset value. On certain models, an address switch is also used. The resulting value is clipped to a 7-bit ID in range 0 to 127.
For EtherCAT, this parameter can optionally hold network alias value to be loaded into ESC at power-up. See Network Options (0x121) for details.
Plus drives with firmware 2.82 or greater have an optional new method of setting Node IDs on multi-axis drives. This new method allows each axis to be assigned its own ID, and Node IDs don't have to be consecutive. See descriptions of parameters Input Pin Mapping, Node ID Selection (0x103) and Network Options (0x121) for details of this new method.
For multi-axis CANopen drives, first axis Node ID is set using this parameter. Subsequent axes are assigned consecutive Node ID's. For example, if first the axis was given Node ID 7 using this parameter, second would be Node 8, and third would be Node 9, etc.
Bit-mapped as follows:</td></tr>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0-6</td><td>Give Node ID offset value that will be added to value read from input pins</td></tr>
<tr><td>7</td><td>Used only on DeviceNet firmware.
If set, drive will be software disabled on startup and will remain disabled until enabled by DeviceNet I/O message with enable bit set.</td></tr>
<tr><td>8-10</td><td>Number of input pins (0-7) to read on startup for Node ID value. If input pins are used (i.e., value in bits 8-10 is not zero), inputs can be mapped to Node ID bits through Input Pin Mapping, Node ID Selection (0x103).</td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
				11	If set, the CAN address selector switch (if available) is used instead of the input pins. This bit is ignored on drives that do not have an address switch. On drives with an address switch, setting this bit programs drive to use address selector switch as part of address calculation. In this case, Node ID value is equal to sum of: ▪ Value read from designated input pins, shifted up 4 bits. ▪ Address switch value. ▪ Programmed offset value. Note that since Node ID is always clipped to lowest 7 bits, no more than three input pins will ever have an effect on Node address when address switch is used.	
				12-15	Set the bit rate for use on the CANopen Network. The valid values for this field are listed below. Network bit rate setting:	
					Value	**Bit Rate (bits/s)**
					0	1,000,000
					1	800,000
					2	500,000
					3	250,000
					4	125,000
					5	50,000
					6	20,000
					7-15	Reserved
0xC2	0x2352	RF	INT16	Homing Method Configuration. Bit-mapped as follows:		
				Bits	**Description**	
				0-3	Home function	
					Value	**Description**
					0	If bit 5 is not set, then just set current position as home. If bit 5 is set, then move in direction specified by bit 4 and set location of first index pulse as home. Bit 6 is not used in this mode.
					1	Move in direction specified by bit 4 until limit switch is encountered. Then move in other direction out of limit. If bit 5 is clear, then edge location is home. If bit 5 is set, then next index pulse is home. Bit 6 not used in this mode.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
					2	Home on constant home switch. Initial move is made in direction specified by bit 4. When home switch is encountered, direction is reversed. If bit 5 is clear, edge of home switch is set as home. If bit 5 is set, then an index pulse is used as home position. Bit 6 is used to define which index pulse is used.
					3	Home on intermittent home switch. This mode works same as mode 2 except that if limit switch is encountered when initially searching for home, then direction is reversed. In mode 2, hitting limit switch before finding home would be considered an error. Bit 8 identifies which edge of home to search for (positive or negative).
					4	Home to a hard stop. This moves in the direction specified in bit 4 until home current limit is reached. It then presses against hard stop using that current value until home delay time expires. If bit 5 (index) is set, drive away from the hard stop until an index is found.
					5-14	Reserved
					15	Immediate home. This value causes the amp to be referenced immediately on power-up. Once encoder is initialized, home offset value is added to encoder position and result is set as current referenced position. This is primarily useful with absolute encoders.
				4	Initial move direction (0=positive, 1=negative)	
				5	Home on index pulse if set	
				6	Selects which index pulse to use. If set, use pulse on DIR side of sensor edge. DIR is direction specified by bit 4 of this word.	
				7	If set, capture falling edge of index. If clear, capture rising edge.	
				8	When using momentary home switch, this bit identifies which edge of home switch to reference on. If set, use negative edge. If clear, use positive edge.	
				9	If set, move to zero position when homing is finished. If clear, zero position is found, but not moved to.	
				10	If set, homing sequence will run as normal, but actual position will not be adjusted at end of homing. Note that even though actual position is not adjusted, Homing Adjustment	

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="3">Description</th></tr>
<tr><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td></td><td colspan="2">(0xB5) is updated with size of adjustment (in counts) that would have been made.
Also, if bit 10 is set then no move to zero is made regardless of setting of bit 9.</td></tr>
<tr><td>11</td><td colspan="2">If this bit is set, at end of home routine home configuration stored in flash will be set to 15, and home offset stored in flash will be updated to correct value necessary to calibrate an absolute encoder based on most recent home operation. This bit is used to automate calibration of absolute encoders.</td></tr>
<tr><td>0xC3</td><td>0x6099:1</td><td>RF</td><td>INT32</td><td colspan="3">Homing Velocity (fast moves). Units: 0.1 counts/s.
This velocity value is used during segments of homing procedure that may be handled at high speed. Generally, this means moves in which home sensor is being located, but edge of sensor is not being found.</td></tr>
<tr><td>0xC4</td><td>0x6099:2</td><td>RF</td><td>INT32</td><td colspan="3">Homing Velocity (slow moves). Units: 0.1 counts/s.
This velocity value is used for homing segments that require low speed, such as cases where edge of a homing sensor is being sought.</td></tr>
<tr><td>0xC5</td><td>0x609A</td><td>RF</td><td>U32</td><td colspan="3">Homing Acceleration/Deceleration.
Units: 10 counts/s^2.
This value defines acceleration used for all homing moves. Same value is used at beginning and ending of moves (i.e. no separate deceleration value).</td></tr>
<tr><td>0xC6</td><td>0x607C</td><td>RF</td><td>INT32</td><td colspan="3">Home Offset. Units: counts.
Home offset is difference between zero position for application and machine home position (found during homing). Once homing is completed, new zero position determined by homing state machine will be located sensor position plus this offset. All subsequent absolute moves shall be taken relative to this new zero position.</td></tr>
<tr><td>0xC7</td><td>0x2350</td><td>RF</td><td>INT16</td><td colspan="3">Homing Current Limit. Units: 0.01 A.
Used in Home to Hard Stop mode only, this current is used to determine when drive has reached end of travel (hard stop). Used in conjunction with Home to Hard Stop Delay Time (0xBF).
Note that the homing current value isn't the current limit that will be used when homing—it's the current threshold. The drive considers the motor to be in a hard stop condition when the actual current exceeds this amount for longer than the homing delay value (parameter 0xBF). During a home to hard stop move the motor current will be temporarily limited to a value that's 25% higher than this setting.</td></tr>
<tr><td rowspan="4">0xC8</td><td rowspan="4">None</td><td rowspan="4">RF</td><td rowspan="4">INT16</td><td colspan="3">Trajectory Profile Mode. To set profile in CANopen see CAN object 0x6086 in <i>CANopen Programmers Manual</i>.
Bit-mapped as follows:</td></tr>
<tr><td><b>Bits</b></td><td colspan="2"><b>Description</b></td></tr>
<tr><td rowspan="2">0-2</td><td colspan="2">Give trajectory profile mode. Possible trajectory modes are described below.</td></tr>
<tr><td><b>Value</b></td><td><b>Description</b></td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="3">Description</th></tr>
<tr><td rowspan="5"></td><td rowspan="5"></td><td rowspan="5"></td><td rowspan="5"></td><td rowspan="4"></td><td>0</td><td>Trapezoidal profile mode. Uses position/distance, velocity, acceleration and deceleration. Any parameters may be changed during move. Jerk is not used in this mode.</td></tr>
<tr><td>1</td><td>S-curve profile mode. Uses position/distance, velocity, acceleration, and jerk. No parameters may be changed while move is in progress (although move may be aborted). Acceleration parameter will be used for deceleration.</td></tr>
<tr><td>2</td><td>Velocity mode. Uses velocity, acceleration, and deceleration. Jerk is not used in this mode, and position is only used to define direction of move (zero or positive to move with a positive velocity, negative to move with a negative velocity). Any parameter may be changed during move. Set velocity to zero to stop.</td></tr>
<tr><td>3</td><td>PVT profile mode. Use of this mode through serial interface is not presently supported.</td></tr>
<tr><td>8</td><td colspan="2">If set, relative move.
If clear, absolute move.</td></tr>
<tr><td rowspan="10">0xC9</td><td rowspan="10">0x2252</td><td rowspan="10">R*</td><td rowspan="10">INT16</td><td colspan="3">Trajectory Status Register. This parameter gives status information about the trajectory generator.
Bit-mapped as follows:</td></tr>
<tr><td><b>Bits</b></td><td colspan="2"><b>Description</b></td></tr>
<tr><td>0-8</td><td colspan="2">Reserved</td></tr>
<tr><td>9</td><td colspan="2">Cam table underflow</td></tr>
<tr><td>10</td><td colspan="2">Reserved</td></tr>
<tr><td>11</td><td colspan="2">Homing error. If set, an error occurred in last home attempt. Cleared by a home command.</td></tr>
<tr><td>12</td><td colspan="2">Referenced. Set when homing command has been successfully executed. Cleared by home command.</td></tr>
<tr><td>13</td><td colspan="2">Homing. If set, drive is running home command.</td></tr>
<tr><td>14</td><td colspan="2">Set when move is aborted.
Cleared at start of next move.</td></tr>
<tr><td>15</td><td colspan="2">In-Motion Bit. If set, trajectory generator is presently generating profile.</td></tr>
<tr><td rowspan="4">0xCA</td><td rowspan="4">0x607A</td><td rowspan="4">RF</td><td rowspan="4">INT32</td><td colspan="3">Trajectory Generator Position Command.
Units: Counts.

This value gives destination position for absolute moves or move distance for relative moves.</td></tr>
<tr><td><b>Type</b></td><td colspan="2"><b>Meaning</b></td></tr>
<tr><td>Relative</td><td colspan="2">Move distance</td></tr>
<tr><td>Absolute</td><td colspan="2">Target position</td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
				Velocity Direction: 1 for positive, -1 for negative
0xCB	0x6081	RF	INT32	Trajectory Maximum Velocity. Trajectory generator will attempt to reach this velocity during a move. Units: 0.1 counts/s.
0xCC	0x6083	RF	U32	Trajectory Maximum Acceleration. Units: 10 counts/s^2. Trajectory generator will attempt to reach this acceleration during a move. For s-curve profiles, this value also used to decelerate at end of move.
0xCD	0x6084	RF	U32	Trajectory Maximum Deceleration. Units: 10 counts/s^2. In trapezoidal trajectory mode, this value used to decelerate at end of move.
0xCE	0x2121	RF	U32	Trajectory Maximum Jerk. Units: 100 counts/s^3. Also known as Trajectory Jerk Limit. S-curve profile generator uses this value as jerk (rate of change of acceleration/deceleration) during moves. Other profiles types do not use jerk limit.
0xCF	0x6085	RF	U32	Trajectory Abort Deceleration. Units: 10 counts/s^2. If move is aborted, this value will be used by trajectory generator to decelerate to stop.
0xD0	0x2192:9	RF	U16	Input 9 Configuration. See Input 0 (IN1) Configuration (0x78).
0xD1	0x2192:10	RF	U16	Input 10 Configuration. See Input 0 (IN1) Configuration (0x78).
0xD2	0x2192:11	RF	U16	Input 11 Configuration. See Input 0 (IN1) Configuration (0x78).
0xD3	0x2192:12	RF	U16	Input 12 Configuration. See Input 0 (IN1) Configuration (0x78).
0xD4	0x2192:13	RF	U16	Input 13 Configuration. See Input 0 (IN1) (Configuration (0x78).
0xD5	0x2192:14	RF	U16	Input 14 Configuration. See Input 0 (IN1) Configuration (0x78).
0xD6	0x2192:15	RF	U16	Input 15 Configuration. See Input 0 (IN1) Configuration (0x78).
0xD7	0x2192:16	RF	U16	Input 16 Configuration. See Input 0 (IN1) Configuration (0x78).
0xD8	0x2150	RF	U16	Regen Resistor Resistance. Units: 0.1 Ω.
0xD9	0x2151	RF	U16	Regen Resistor, Continuous Power. Units: W.
0xDA	0x2152	RF	U16	Regen Resistor, Peak Power. Units: W.
0xDB	0x2153	RF	U16	Regen Resistor, Time at Peak. Units: ms.
0xDC	0x2154	RF	INT16	Regen Turn on Voltage Units: 0.1 V.
0xDD	0x2155	RF	INT16	Regen Turn off Voltage. Units: 0.1 V.
0xDE	0x2384:20	F*	INT16	Drive’s Peak Current Rating for Internal Regen Transistor. Units: 0.01 A.
0xDF	0x2384:21	F*	INT16	Drive's Continuous Current Rating for Internal Regen Transistor. Units: 0.01 A.
0xE0	0x2384:22	F*	INT16	Drive's Time at Peak Current for Internal Regen Transistor. Units: ms.
0xE1	0x2156	F	String	Regen Resistor Model Number String.

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="6">0xE2</td><td rowspan="6">0x2157</td><td rowspan="6">R*</td><td rowspan="6">INT16</td><td colspan="2">Regen Resistor Status. Bit-mapped as follows:</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>Set if regen circuit is currently closed.</td></tr>
<tr><td>1</td><td>Set if regen is required based on bus voltage.</td></tr>
<tr><td>2</td><td>Set if regen circuit is open due to an overload condition. Overload may be caused by either resistor settings or internal drive protections.</td></tr>
<tr><td>3-15</td><td>Reserved</td></tr>
<tr><td>0xE3</td><td>0x2382:4</td><td>RF</td><td>U16</td><td colspan="2">Position Loop Output Gain Multiplier. Output of position loop is multiplied by this value before being passed to velocity loop. This scaling factor is calculated such that a value of 100 is a 1.0 scaling factor.
This parameter is most useful in dual loop systems.</td></tr>
<tr><td>0xE4</td><td>0x21C2</td><td>RF</td><td>INT16</td><td colspan="2">Maximum Current to use with algorithmic phase initialization. See <i>Value 5</i> of Commutation Mode (0xB2).
Units: 0.01 A.</td></tr>
<tr><td>0xE5</td><td>0x21C3</td><td>RF</td><td>U16</td><td colspan="2">Algorithmic Phase Initialization Timeout. See <i>Value 5</i> of Commutation Mode (0xB2). Units: ms.</td></tr>
<tr><td>0xE6</td><td>0x21D8</td><td>RF</td><td>INT32</td><td colspan="2">Max Step Rate. This is maximum velocity adjustment made by stepper outer position loop when enabled. This parameter is only used when stepper outer loop is engaged (bit 1 of Stepper Configuration & Status (0xEE) is set).
Units: 0.1 steps/s.</td></tr>
<tr><td>0xE7</td><td>0x21D7</td><td>RF</td><td>U16</td><td colspan="2">Proportional Gain for Stepper Outer Loop. (ECp) Encoder Corrections Proportional Gain. This parameter gives the gain used for calculating velocity adjustment based on Position Loop Error (0x35). This parameter is only used when stepper outer loop is engaged (bit 1 of Stepper Configuration & Status (0xEE) is set).</td></tr>
<tr><td>0xE8</td><td>0x21D0</td><td>RF</td><td>INT16</td><td colspan="2">Holding Current for Microstepping Mode.
Units: 0.01 A.</td></tr>
<tr><td>0xE9</td><td>0x21D1</td><td>RF</td><td>U16</td><td colspan="2">Run to Hold Time for Microstepping Mode. Units: ms.</td></tr>
<tr><td>0xEA</td><td>0x21D2</td><td>RF</td><td>U16</td><td colspan="2">Detent Correction Gain Factor for Microstepping Mode.</td></tr>
<tr><td>0xEB</td><td>0x21D3</td><td>RF</td><td>U16</td><td colspan="2">Damping Correction Gain Factor for Microstepping Mode</td></tr>
<tr><td>0xEC</td><td>0x21D4</td><td>RF</td><td>9 or 14</td><td colspan="2">Damping Correction bi-quad filter structure for Microstepping Mode.
For details on encoding of filter structure, please see Filter Coefficients.</td></tr>
<tr><td>0xED</td><td>0x21D5</td><td>RF</td><td>U16</td><td colspan="2">Holding Current to Fixed Voltage Output Time for Microstepping Mode. Time delay from entering hold current before entering special voltage control mode of operation. This mode trades normal tight control of current for very low jitter on motor position.
Used in stepper mode only. Set to 0 to disable this feature. Units: ms.</td></tr>
<tr><td rowspan="2">0xEE</td><td rowspan="2">0x21D6</td><td rowspan="2">RF</td><td rowspan="2">INT16</td><td colspan="2">Stepper Configuration & Status.
Bit-mapped as follows:</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
				0	Use encoder input for phase compensation if enabled. Pure stepper mode if disabled.	
				1	Use outer position loop to adjust stepper position based on Position Loop Error (0x35). When this bit is set, gain value Proportional Gain (ECp) (0xE7) is multiplied by Position Loop Error (0x35) and result is velocity that is added to Microstepping position limited by Max Step Rate (0xE6).	
				2-15	Reserved	
0xF0	0x2195:1	RF	U16	Switch Debounce Time For Input 1. Units: ms.		
0xF1	0x2195:2	RF	U16	Switch Debounce Time For Input 2. Units: ms.		
0xF2	0x2195:3	RF	U16	Switch Debounce Time For Input 3. Units: ms.		
0xF3	0x2195:4	RF	U16	Switch Debounce Time For Input 4. Units: ms.		
0xF4	0x2195:5	RF	U16	Switch Debounce Time For Input 5. Units: ms.		
0xF5	0x2195:6	RF	U16	Switch Debounce Time For Input 6. Units: ms.		
0xF6	0x2195:7	RF	U16	Switch Debounce Time For Input 7. Units: ms.		
0xF7	0x2195:8	RF	U16	Switch Debounce Time For Input 8. Units: ms.		
0xF8	0x2195:9	RF	U16	Switch Debounce Time For Input 9. Units: ms.		
0xF9	0x2195:10	RF	U16	Switch Debounce Time For Input 10. Units: ms.		
0xFA	0x2195:11	RF	U16	Switch Debounce Time For Input 11. Units: ms.		
0xFB	0x2195:12	RF	U16	Switch Debounce Time For Input 12. Units: ms.		
0xFC	0x2195:13	RF	U16	Switch Debounce Time For Input 13. Units: ms.		
0xFD	0x2195:14	RF	U16	Switch Debounce Time For Input 14. Units: ms.		
0xFE	0x2195:15	RF	U16	Switch Debounce Time For Input 15. Units: ms.		
0xFF	0x2195:16	RF	U16	Switch Debounce Time For Input 16. Units: ms.		
0x100	0x2184	RF	U32	CANopen Limit Status Mask. This parameter defines which bits in Event Status Register (0xA0) can set limit bit (bit 11) of CANopen Status Word (CANopen index 0x6041 as described in *CANopen Programmer's Manual*). If Event Register Status (0xA0) and its corresponding Limit Mask bit are both set, then CANopen Status Word limit bit is set. If all selected Event Status Register (0xA0) bits are clear, then limit bit is clear.		
0x101	0x2197	R*	INT16	Network Address Switch Value. This gives current state of address switch. For drives without a switch, value returned is undefined.		
0x102	0x21B4	R*	INT16	Network Status Word. Bit-mapped as follows:		
				CANopen		
				Bits	**Meaning**	
				0-1	CANopen node status. This field will take one of following values:	
					Value	**Status**
					0	CANopen interface is disabled.
					1	Stopped mode.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
					2	Preoperational mode.
					3	Operational mode
				4	Set if CANopen SYNC message is missing	
				5	Set on CANopen guard error	
				8	Set if CAN port is in 'bus off' state	
				9	Set if CAN port is in 'transmit error passive' state	
				10	Set if CAN port is in 'receive error passive' state	
				11	Set if CAN port is in 'transmit warning' state	
				12	Set if CAN port is in 'receive warning' state	
				15	Always clear for CANopen	
				DeviceNet		
				Bits	**Meaning**	
				0	Set if duplicate MAC ID check failed	
				1	Set if device is online	
				2	Set if at least one communication object timed out	
				3	Set if at least one communication object has been established	
				4-7	Reserved	
				8-14	Same bit mapping as for CANopen.	
				15	Always set for DeviceNet.	
				EtherCAT		
				0	Set if distributed clock is enabled	
				1	Set if distributed clock is locked	
				2	Set if SYNC0 period is multiple of drive's servo period	
				3	Set if invalid SYNC0 time	
				4-15	Reserved for future use	
				MACRO		
				0	Set if MACRO network is detected	
				1	Set if drive is being disabled by MACRO master	
				2	Set if MACRO network has been broken (i.e. once detected but now gone)	
				3	Set on heartbeat error	
				4	Ring break error received from upstream device	
				5-15	Reserved	
0x103	0x21B1	F	U32	Input Pin Mapping for Node ID Selection. When CAN Network Node ID Configuration (0xC1) indicates that 1 or more input pins will be used to select Node ID, this parameter is used to map input pins to ID bits.		
				Bits	**Meaning**	

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				0-3	Identify the general-purpose input pin associated with ID bit 0
				4-7	Identify the general-purpose input pin associated with ID bit 1
				8-11	Identify the general-purpose input pin associated with ID bit 2
				12-15	Identify the general-purpose input pin associated with ID bit 3
				16-19	Identify the general-purpose input pin associated with ID bit 4
				20-23	Identify the general-purpose input pin associated with ID bit 5
				24-27	Identify the general-purpose input pin associated with ID bit 6
				28-30	Reserved
				31	Set to enable this register. Clear to use default mapping
				If bit 31 is zero, then default bit mapping is used, and rest of this register is ignored. Default bit mapping uses top N input pins and maps them such that high numbered pins are used for higher numbered bits in Node ID. For example; Accelnet Panel drive has 12 general-purpose input pins (0 to 11). If 3 of these pins are used for Node ID configuration and default mapping is used, then highest 3 pins (9, 10 and 11) will be used for Node ID. In this case, pin 9 will be bit 0, pin 10 will be bit 1 and pin 11 will be bit 2. If bit 31 is set, then rest of this register will be used to define which input pin will be assigned to which bit of Node ID. Input pins are numbered from 0 to 15 and each nibble of register gives input pin number associated with one bit of Node ID. For example, if three input pins are configured for address selection and the mapping register is set to 0x80000012, then input pin 2 will be used for Node ID bit 0, input pin 1 will be used for Node ID bit 1 and input pin 0 will be used for Node ID bit 2. Note that CAN Node ID is calculated at startup only. Input pins assigned to Node ID will be sampled once during power up and used to calculate Node ID. These pins may be assigned other uses after power up if necessary. Starting with Plus drive firmware version 2.82, a new optional method of setting Node IDs of multi-axis drives is supported. This new method is enabled by setting bit 3 of Network Options (0x121). If this method of setting Node IDs is enabled, then parameter 0xC1 is not used for setting Node IDs. Instead, Node IDs of all nodes are set using this parameter. When this optional method of setting Node IDs is used, this parameter is bit-mapped as follows: This optional method of setting Node IDs allows multi-axis drives to have non-consecutive Node IDs. Note that	

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				it is possible to set multiple axes to same Node ID using this method which would result in errors.	
				Bits	**Meaning**
				0-6	Node ID of axis 1
				8-14	Node ID of axis 2
				16-22	Node ID of axis 3
				24-30	Node ID of axis 4
0x104	0x21C4	RF	INT16	Algorithmic Phase Initialization Config. See Value 5 of Commutation Mode (0xB2). Bit-mapped as follows:	
				Bits	**Description**
				0	If set, don't try to guess phase angle at startup, just force initial phase angle
				1	If set, increment initial phase angle by 90 degrees on each failed attempt
				2	If set, use Motor Hall Offset (0x4F) as the initial angle for first phase initialization attempt. If clear, first phase angle is zero.
				3	Ignore limit switches during phase initialization if switch is configured as trajectory based. Available in Feature set C only.
				4-15	Reserved
0x105	0x2360	RF	U16	Camming Configuration. For more information, see *Copley Camming User Guide.* Bit-mapped as follows:	
				Bits	**Description**
				0-3	ID Number of Cam Table to use (0-9)
				4	Reserved
				5	If set, only allow forward motion through CAM table
				6	If set, use Camming Internal Generator. Internal generator runs at constant velocity programmed in Camming Master Velocity (0x109). If clear, use digital command input as configured in CME software camming controls or Input Pin States (0xA6).
				7	If set, run tables stored in RAM. If clear, use tables stored in flash file system. This bit is used to select between running CAM tables stored in the flash file system (standard mode), and running tables stored in RAM. Tables stored in flash can be uploaded through the CME program. These tables will remain available between system starts. Tables stored in RAM will be lost each time the drive is powered down or reset.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description		
				8-11	Input number to use as Cam Trigger. Note: a value of 0 selects In1, 1 selects In2, etc.	
				12-14	Cam Trigger type: The input trigger identifies the type of input which will start CAM table operation. It should take one of the following values:	
					Value	**Type**
					0	None (Continuous): Active Cam Table is repeated continuously.
					1	Use Input, Edge: Active Cam Table begins executing on rising edge of input pin selected by bits 8-11.
					2	Use Input, Level: Active Cam Table will run if input selected by bits 8-11 is high.
					3	Use Master (Secondary) Encoder Index: Active Cam Table is executed when drive receives an index pulse from Master encoder. Index pulses received during execution are ignored.
					7	Never trigger. This can be used to stop CAM currently in progress.
0x106	0x2361	RF	INT16	Camming delay, forward motion. Units: master command counts. This gives delay used when entering cam table in forward direction.		
0x107	0x2362	RF	INT16	Camming delay, reverse motion. Units: master command counts. This gives delay used when entering a cam table in reverse direction.		
0x108	None	R	INT16	Writing any value to this parameter will cause any CANopen PDO objects configured with type code 254 to be sent. This parameter is primarily useful for triggering a PDO from within CVM program. Reading this parameter does not return any useful information.		
0x109	0x2363	RF	INT 32	Camming Master Velocity. Units: 0.1 counts/s. Constant velocity of Camming Internal Generator.		
0x10A	0x2403	R*	INT 32	Captured Home Position. Units: counts. Provides position that axis was in when an input pin configured as home switch input became active. Configured by setting bits in Position Capture Control Register (0x6C). Status of captured data can be checked in Position Capture Status Register (0x6D). Reading this variable resets bits 4 & 7 of Position Capture Status Register (0x6D).		
0x10B	0x2422	R*	U32	Firmware Version Number (extended). Upper 16 bits give same major/minor version number as Firmware Version Number (0x94). Lower 16 bits hold release number (upper byte) and reserved byte (lower).		
0x10C	0x1017	RF	U16	CANopen Heartbeat Time. Units: ms. Frequency at which drive will produce heartbeat messages. This parameter may be set to zero to disable heartbeat production. Note that only one of the two node-guarding methods may be used at once. If		

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
				Heartbeat Time is non-zero, then heartbeat protocol is used regardless of settings of CANopen Node Guarding Time (0x10D) and CANopen Node Guarding Time Life Factor (0x10E).
0x10D	0x100C	RF	U16	CANopen Node Guarding Time. Units: ms. This parameter gives time between node-guarding requests that are sent from CANopen master to drive. Drive will respond to each request with node-guarding message indicating internal state of the drive. If drive has not received node-guarding request within time period defined by product of Node Guarding Time and CANopen Node Guarding Life Time Factor (0x10E), drive will treat this lack of requests as fault.
0x10E	0x100D	RF	U8	CANopen Node Guarding Lifetime Factor. This object gives multiple of CANopen Node Guarding Time (0x10D). Drive expects to receive node-guarding request within time period defined by product of CANopen Node Guarding Time (0x10D) and Lifetime Factor. If drive has not received node-guarding request within this time, it treats lack of requests as fault.
0x10F	0x2325	R	INT 32	Registration Offset for Pulse & Direction Mode. When running in pulse & direction mode (Desired State (0x24) = 23), this parameter may be used to inject an offset into master position. Offset will immediately be cleared once it has been applied to master position, so this parameter will normally be read back as zero when running in pulse and direction mode 23. When running in PWM position mode, offset value is added to absolute position calculated using Minimum PWM Pulse Width (0x13C) and Maximum PWM Pulse Width (0x13D) and Digital Input Scaling Factor (0xA9).
0x110	0x2404	R	INT 32	Time Stamp of Last High-Speed Position Capture. Units: us. If high-speed position capture is enabled, this parameter gives time of last capture. Setting this parameter causes drive to calculate its position at set time if position capture is enabled and time is recent enough for data to be available. Calculated position may be read from Captured Position for High-Speed Position Capture (0x111). This feature is mainly used when capturing position on multiple drives across network.
0x111	0x2405	R*	INT 32	Captured Position for High-Speed Position Capture. Units: counts.
0x112	0x2242	R	INT 32	Load Encoder Position. Units: counts. If set, this returns position of load encoder. When used in passive mode this returns passive load position.
0x113	0x1015	RF	INT16	CANopen emergency inhibit time. Units: ms.
0x114	0x2381:5	RF	U16	Velocity Loop Drain (integral bleed). Range: 0 to 32767, Default: 0.

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="2">Modifies effect of Velocity Loop Integral Gain (Vi).
Higher Vi Drain value, faster integral sum is lowered.</td></tr>
<tr><td rowspan="18">0x115</td><td rowspan="18">0x2010</td><td rowspan="18">R</td><td rowspan="18">5 Words</td><td colspan="2">Trajectory Buffer Access. This object can be used to load data into the drive's internal trajectory buffer or send commands used to control buffer. Trajectory buffer holds trajectory segments used in PVT mode.

Data passed to this parameter consists of a 16-bit command code, followed by up to two 32-bit parameters.

First word passed to this parameter is bit-mapped. Data contained in this word identifies this access as either buffer command or trajectory segment to be loaded into buffer. If most significant bit of first word is set, then write is treated as command code.

In this case no additional data is passed and first word is formatted as follows:</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0-7</td><td>Command data</td></tr>
<tr><td>8-9</td><td>Command code</td></tr>
<tr><td>10-14</td><td>Reserved</td></tr>
<tr><td>15</td><td>Always set for buffer commands</td></tr>
<tr><td colspan="2">Following command values are supported:</td></tr>
<tr><td>Value</td><td>Description</td></tr>
<tr><td>0</td><td>Clear buffer and abort any move in progress</td></tr>
<tr><td>1</td><td>Pop N most recently sent segments off buffer. PVT profiles will continue to run as long as buffer doesn't underflow. Number of segments to pop (N) is passed in command data area. If there are less than N segments on buffer, this acts same as buffer clear, except that profile is not stopped except by underflow.</td></tr>
<tr><td colspan="2">To write data to trajectory buffer, most significant bit of first word must be clear.

In this case, first word is formatted as follows:</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0-7</td><td>Segment time in ms.</td></tr>
<tr><td>8-11</td><td>Reserved</td></tr>
<tr><td>12</td><td>Set for relative positions.
clear for absolute positions.</td></tr>
<tr><td>13-14</td><td>Reserved</td></tr>
<tr><td>15</td><td>Always zero for data writes</td></tr>
<tr><td colspan="2">When writing new PVT segment to trajectory buffer, first word is always followed by a 32-bit position value. Position is specified in units of encoder counts and can be interpreted as either absolute or relative based on bit 12 of command word.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td>Optionally, position can be followed by a 32-bit velocity value. Velocity is specified in units of 0.1 encoder counts/second. If velocity value is
supplied, then drive will use cubic polynomial interpolation between points when running trajectory (PVT mode). If velocity is not supplied, then linear interpolation will be used (PT mode).
It is acceptable to mix PVT and PT segments within same move.

Reading this parameter always returns three words of status information about trajectory buffer.

First returned word is formatted as follows:
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0-7</td><td>Number of free locations in trajectory buffer.</td></tr>
<tr><td>8-15</td><td>Reserved.</td></tr>
</table>
The second two words are reserved for future use.</td></tr>
<tr><td>0x116</td><td>0x605A</td><td>RF</td><td>INT16</td><td>CANopen Quick Stop Option code.</td></tr>
<tr><td>0x117</td><td>0x605B</td><td>RF</td><td>INT16</td><td>CANopen Shutdown Option code.</td></tr>
<tr><td>0x118</td><td>0x605C</td><td>RF</td><td>INT16</td><td>CANopen Disable Option code.</td></tr>
<tr><td>0x119</td><td>0x605D</td><td>RF</td><td>INT16</td><td>CANopen Halt Option code.</td></tr>
<tr><td>0x11A</td><td>0x2080</td><td>F*</td><td>U32</td><td>Drive Scaling Configuration. Defines units used for current and voltage readings from drive:
<table>
<tr><th>Bits</th><th colspan="2">Description</th></tr>
<tr><td rowspan="5">0-1</td><td colspan="2">Identify units for current readings:</td></tr>
<tr><td>0</td><td>0.01 A</td></tr>
<tr><td>1</td><td>0.001 A</td></tr>
<tr><td>2</td><td>0.0001 A</td></tr>
<tr><td>3</td><td>0.00001 A</td></tr>
<tr><td>2-7</td><td></td><td>Reserved</td></tr>
<tr><td rowspan="5">8-9</td><td colspan="2">Identify units for voltage readings:</td></tr>
<tr><td>0</td><td>0.1 V</td></tr>
<tr><td>1</td><td>0.01 V</td></tr>
<tr><td>2</td><td>0.001 V</td></tr>
<tr><td>3</td><td>0.0001 V</td></tr>
<tr><td>10-31</td><td colspan="2">Reserved</td></tr>
</table></td></tr>
<tr><td>0x11B</td><td>0x6082</td><td>R</td><td>INT32</td><td>Trajectory Ending Velocity. For use with trap profile mode, gives velocity at end of moves. Primarily used when linking multiple moves together.</td></tr>
<tr><td>0x11C</td><td>0x2256</td><td>R</td><td>U32</td><td>Trajectory Sequence Buffer Status. Trajectory sequence buffer is used in CANopen profile position mode and stores trajectory segments added using the 'set of setpoints' method described in the CANopen specification. This parameter allows buffer status to be queried.

Bit-mapped as follows:
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0-7</td><td>Number of free locations in buffer</td></tr>
</table></td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				8-15	Number of full locations in buffer
				16-31	Reserved
0x11D	0x222B	RF	U32	Encoder Error Filter Configuration. Encoder error filter can be used to detect and ignore bad position data from an encoder or temporary encoder errors. Bad encoder readings are detected by comparing an expected position (based on extrapolation of previous readings) and actual reading from encoder.	
				Bits	**Description**
				0-3	Maximum number of consecutive bad samples to ignore. If zero then filter is disabled.
				4-15	Reserved
				16-27	Maximum error between extrapolated reading and actual reading to consider reading bad
				28-31	Reserved
0x11E	0x222C	R	U32	Encoder Error Filter Status. This can be cleared by writing zero to it.	
				Bits	**Description**
				0-3	Count of consecutive bad readings
				4-7	Reserved
				8	Set if encoder fault was generated by filter
				9-15	Reserved
				16-31	Total number of times extrapolated position has been used due to detected error
0x11F	0x21B5	RF	U32	IP address. Is a valid IPV4 address for the Ethernet network the drive is attached to. IP addresses are normally written out as a series of four decimal values separated by periods such as: 192.168.1.1. When passed to parameter 0x11F, the four decimal values should be packed into a single 32-bit value in little endian format. That is, the right-most digit in the IP address is the most significant byte in the 32-bit value. The IP address 192.168.1.1 would be formatted as 0x0101A8C0. When the drive is configured to obtain its own IP address using DHCP, this parameter will return 0 until an IP address has been assigned, at which point this parameter will return that address. The address assigned by the server is stored to flash and the drive will request the same address from the DHCP server the next time it powers up.	
0x120	0x2384:25	R*	INT16	Returns number of axis implemented by this drive	
0x121	0x21B3	RF	INT16	Network Options. Configures the drive’s network. Details of its meaning depend on type of network implemented in drive.	
				CANopen	
				Bits	**Meaning**

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				0	Must be clear to select CANopen networking
				1	If set, causes the drive to go to CANopen fault state when a fault occurs. Clear for backwards compatibility.
				2	If set, user CAN J1939 protocol for ARM and FPGA Plus drives.
				3	If set, use an alternative method of assigning Node IDs to each axis. See Input Pin Mapping for Node ID Selection (0x103) for details.
				4-7	Reserved
				8	If set, PDO mapping will be saved to flash when object 0x1010 is used to save drive state
				9	If set, PDO communications settings will be stored to flash when object 0x1010 is used to save drive state
				10-11	Reserved
				12	If set, makes drive conform to CANopen specs more strictly. Clear for backwards compatibility.
				13-15	Reserved
				DeviceNet	
				Bits	**Meaning**
				0	Must be set to select DeviceNet networking
				1-15	Reserved.
				MACRO	
				Bits	**Meaning**
				0	If set, position data sent over MACRO network is shifted up 5 bits for compatibility with Delta-Tau controllers.
				1	If set, drive will be disabled on startup until it is enabled through MACRO interface. If clear, drive can be used without MACRO interface connected until it starts receiving MACRO messages.
				2	If set, return primary encoder index state (high/low) in the home status bit of MACRO status word. If clear, state of any general-purpose input configured as home input will be used.
				3	If set, drive will attempt to synchronize its current loop update period to MACRO ring period. Ring period must be an integer multiple of drive's PWM Period (0x85).
				4-7	Defines what type of additional data is transmitted in the first auxiliary data register of every MACRO response message: 0 – Send digital input value 1 – Send secondary analog reference value

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
					2 – Send unfiltered secondary analog reference value 3 – Send motor encoder reading 4 – Send load encoder reading
				8-11	Defines what type of additional data is transmitted in second auxiliary data register of every MACRO response message: 0 – send analog input value 1 – send primary encoder reading 2 – send secondary encoder reading 3 – Pulse & direction hardware count. 4 – Unfiltered analog reference value
				12	If set, push synchronization point back ½ current loop period.
				13-15	Reserved.
				EtherCAT	
				Bits	**Meaning**
				0	If set, disable some extra checks of SYNC0 configuration which were added for improved network conformance.
				1	If set, drive will follow EtherCAT state machine even when running in a non- EtherCAT mode of operation.
				2	If set, object 0x1002 is bit-wise OR of all axes Event Status Register (0xA0) for multi-axis drives. If clear, 0x1002 is for axis 1 only.
				3	If set, value of Network Node Id Configuration (0xC1) will be used as network alias on powerup. If clear, alias will be set from address switches
				4-7	Reserved.
				8	If set, PDO mapping will be saved to flash when parameters are saved using object 0x1010
				9	If set, use standard Ethernet protocols (UDP, Modbus TCP, TCP/IP) rather than standard EtherCAT operation
				10-15	Reserved
				Ethernet	
				Bits	**Meaning**
				0	If set, the drive will request an IP address from a DHCP server on the network. The resulting IP address can be read from the IP address (0x11F)
				1-7	Reserved
				9	If set, use standard Ethernet protocols (UDP, Modbus TCP, TCP/IP) rather than standard EtherCAT operation
				10-15	Reserved
0x122	0x2384:26	F*	INT16	Internal Regen Current. Units: mA. Internal drive constant for factory use.	

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="3">Description</th></tr>
<tr><td>0x123</td><td>0x2220</td><td>RF</td><td>INT32</td><td colspan="3">Motor Encoder Wrap Position. Units: counts
Actual motor position will wrap back to zero when this value is reached.
Setting this value to zero disables this feature.</td></tr>
<tr><td>0x124</td><td>0x2221</td><td>RF</td><td>INT32</td><td colspan="3">Load Encoder Wrap Position. Units: counts
Actual load position will wrap back to zero when this value is reached.
Setting this value to zero disables this feature.</td></tr>
<tr><td rowspan="12">0x125</td><td rowspan="12">None</td><td rowspan="12">RF</td><td rowspan="12">INT16</td><td colspan="3">Configures MACRO drive's encoder capture circuit. This parameter is only used on MACRO drives.
Bit-mapped as follows:</td></tr>
<tr><td><b>Bits</b></td><td colspan="2"><b>Meaning</b></td></tr>
<tr><td rowspan="5">0-3</td><td colspan="2">Type of capture to use.</td></tr>
<tr><td>Value</td><td>Description</td></tr>
<tr><td>0</td><td>Capture on edge of encoder index.</td></tr>
<tr><td>1</td><td>Capture using a general-purpose input pin.</td></tr>
<tr><td>2-15</td><td>Reserved.</td></tr>
<tr><td>4-7</td><td colspan="2">Input pin number to use if using capture type 1.</td></tr>
<tr><td>8</td><td colspan="2">Active level; high if clear, low if set.</td></tr>
<tr><td>9</td><td colspan="2">If set, capture is re-enabled immediately when the capture position is read (using I-variable 921).
If clear, capture is only re-enabled on an explicit clear instruction.</td></tr>
<tr><td>10</td><td colspan="2">If set, passive load encoder, if configured, will be captured. Passive load encoder currently only supports capture type 1 (general purpose input).</td></tr>
<tr><td>11-15</td><td colspan="2">Reserved</td></tr>
<tr><td>0x126</td><td>0x2384:27</td><td>R*</td><td>INT16</td><td colspan="3">FPGA Version Number.</td></tr>
<tr><td rowspan="11">0x127</td><td rowspan="11">0x2370</td><td rowspan="11">RF</td><td rowspan="11">U32</td><td colspan="3">Gain Scheduling Configuration:</td></tr>
<tr><td><b>Bits</b></td><td colspan="2"><b>Meaning</b></td></tr>
<tr><td rowspan="9">0-2</td><td colspan="2">Key parameter for gain scheduling.</td></tr>
<tr><td><b>Value</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>None. Setting key parameter to zero disables gain scheduling.</td></tr>
<tr><td>1</td><td>Use value written to Gain Schedule Key Parameter (0x128) as the key</td></tr>
<tr><td>2</td><td>Use Instantaneous Commanded Velocity (0x3B).</td></tr>
<tr><td>3</td><td>Use Load Encoder Velocity (0x5F).</td></tr>
<tr><td>4</td><td>Use Commanded Position (0x2D).</td></tr>
<tr><td>5</td><td>Use Actual Position (0x17).</td></tr>
<tr><td>6-7</td><td>Reserved.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="3">Description</th></tr>
<tr><td rowspan="21"></td><td rowspan="21"></td><td rowspan="21"></td><td rowspan="21"></td><td rowspan="8"></td><td colspan="2">In addition to setting this parameter, a gain table must be loaded into the CVM file system. The table must be given the name '_GAINS' in CVM file system. When gain scheduling is active, drive will linearly interpret between rows of the table based on current value of key parameter. This table should contain at least two rows of gains. Each row must contain the following information:</td></tr>
<tr><td>1</td><td>Key value. This is 32-bit value which must increase for each entry in table. Most significant word is stored first.</td></tr>
<tr><td>2</td><td>Position Loop Proportional Gain (Pp)</td></tr>
<tr><td>3</td><td>Velocity Loop Proportional Gain (Vp)</td></tr>
<tr><td>4</td><td>Velocity Loop Integral Gain (Vi)</td></tr>
<tr><td>5</td><td>Current offset value</td></tr>
<tr><td>6</td><td>Position Loop Integral Gain (Pi)</td></tr>
<tr><td>7</td><td>Position Loop Derivative Gain (Pd)</td></tr>
<tr><td>3-7</td><td colspan="2">Reserved</td></tr>
<tr><td>8</td><td colspan="2">If set, use absolute value of key parameter for gain lookup</td></tr>
<tr><td>9</td><td colspan="2">If set, disable gain scheduling until position encoder is referenced</td></tr>
<tr><td>10-15</td><td colspan="2">Reserved</td></tr>
<tr><td>16</td><td colspan="2">Table includes position loop Pp if set</td></tr>
<tr><td>17</td><td colspan="2">Table includes velocity loop Vp if set</td></tr>
<tr><td>18</td><td colspan="2">Table includes velocity loop Vi if set</td></tr>
<tr><td>19</td><td colspan="2">Table includes current loop offset if set</td></tr>
<tr><td>20</td><td colspan="2">Table includes position loop Pi if set</td></tr>
<tr><td>21</td><td colspan="2">Table includes position loop Pd if set</td></tr>
<tr><td>22-31</td><td colspan="2">Reserved</td></tr>
<tr><td colspan="3" style="display:none"></td></tr>
<tr><td colspan="3" style="display:none"></td></tr>
<tr><td>0x128</td><td>0x2371</td><td>R</td><td>INT32</td><td colspan="3">Gain Scheduling Key Parameter Value.
When gain scheduling is enabled, current value of key parameter is stored here. When this parameter is selected as key parameter for gain scheduling, then it may be written to manually move through entries in gain scheduling table.</td></tr>
<tr><td>0x129</td><td>0x2384:29</td><td>R</td><td>U32</td><td colspan="3">Drive Hardware Options. Reserved for Copley Controls use.</td></tr>
<tr><td rowspan="3">0x12A</td><td rowspan="3">0x2222</td><td rowspan="3">F</td><td rowspan="3">U32</td><td colspan="3">Motor Encoder Options.
Used to specify various configuration options for motor encoder. Mapping of option bits to function depends on encoder type. Any bit not defined for an encoder should be considered reserved. Reserved bits should be set to zero to ensure compatibility with future firmware updates. Bit-mapped as follows:</td></tr>
<tr><td colspan="3">Quadrature Encoder</td></tr>
<tr><td><b>Bits</b></td><td colspan="2"><b>Description</b></td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				0	If set, ignore differential signal errors (if detected in hardware)
				1	If set, select single ended encoder inputs (if available in hardware)
				2	If set, ignore differential signal errors on index input only (if supported by hardware)
				3	If set, don't use index input at all. Useful when index input is being used by a different encoder interface
				4	Reserved
				5	If set, allows initialization of encoder type or options without resetting the position to 0. Normally the position would be set to 0.
				Resolver (encoder type 5):	
				Bit	**Description**
				16	Set for NSK custom incremental resolver
				17	Set for NSK custom absolute resolvers
				18	Set for NSK custom resolvers on normal brushless motors. Clear for normal resolvers, or NSK resolvers on custom NSK
				EnDat Encoder (Type 11)	
				Bits	**Description**
				0-5	Number of bits of single turn data available from encoder
				8-12	Number of bits of multi-turn data available from encoder
				16	If set, analog inputs are supplied by encoder
				17	If set, use multi-mode port
				18	If set, read position using EnDat 2.2 style commands rather than default 2.1 style
				19	If set, read encoder at current loop update rate. Otherwise, encoder is read at servo loop period.
				20-23	Number of least significant bits of encoder reading to discard
				SSI Encoder (Type 12)	
				Bits	**Description**
				0-5	Number of bits of position data available
				8-11	Number of extra bits sent with position data
				12	*If set, ignore first bit of data sent by encoder
				13	If set, encoder outputs position data using Gray code
				14	*If set, pull clock low briefly after data (custom for Codechamp encoder)
				15	If set, data is sent least signification bit first.
				16-21	Encoder Bit Rate. If set, use 100 kHz units. If zero, use default 1 MHz units.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				22	*If set, use setting of Motor Encoder Counts/Rev (0x62) to determine how many data bits to use
				23	If set, extra status bits are before position data. If clear, extra status bits are after position data. Default is clear.
				24	If set, first bit sent is 'data valid' bit
				25	If set, use multi-mode port for SSI interface
				26	If set, extra bits after position data are treated as fault bits and generate an encoder fault if any are set.
				* NOTE – these three bits are depreciated and will be removed in future firmware versions	
				BiSS (Type 13)	
				Bits	**Description**
				0-5	Number of bits of single turn data
				8-12	Number of bits of multiturn data
				15	If set, assume encoder position data wraps after number of encoder counts programmed in Motor Encoder Counts/Rev (0x62)
				16	Set for modeC encoder format
				17	Set to sample at servo loop rate (default at current loop rate)
				19	Set to treat the encoder error bit as a warning (no fault)
				20	If set, encoder error and warning bits are active low
				21	Set if encoder status bits are sent before position data, clear if status bits are sent after position data
				22	If set, encoder error bit is transmitted before warning bit. If clear, warning bit sent first.
				23	If set, error bits are sent after alignment bits. If clear, encoder error bits are sent between alignment bits and position data
				24-27	Number of alignment bits (reserved bits sent before position info)
				28	If set, use multi-mode encoder. If clear, use primary encoder.
				29	If set, use multi-mode encoder. If clear, use a primary encoder
				30	If set, use 2.5 MHz baud rate. If clear, use 4 MHz baud rate.
				BiSS encoders are not always consistent with order in which data is sent. We treat data as consisting of three fields, position data <P>, 2 status bits <S> and optional alignment bits <A> which we ignore. Formatting bits identify order of these three fields. Chart below shows order of fields based on format code. Note that data is always sent most significant bit first, so leftmost field is first transmitted. Format Order of fields	

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="29"></td><td rowspan="29"></td><td rowspan="29"></td><td rowspan="29"></td><td colspan="2">0 <P> <S> <A>
1 <S> <P> <A>
2 <P> <A> <S>
3 <S> <A> <P></td></tr>
<tr><td colspan="2">Absolute A format. Tamagawa, Panasonic, Harmonic Drives, Sanyo Denki, N-A format. (Type 14)</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0-5</td><td>Number of bits of single turn data</td></tr>
<tr><td>8-12</td><td>Number of bits of multi-turn data</td></tr>
<tr><td>16-19</td><td>Number of LSB to discard from reading</td></tr>
<tr><td>20-22</td><td>Number of consecutive CRC errors to ignore before generating an error</td></tr>
<tr><td>24-27</td><td>Encoder sub-type (0=Tamagawa, 1=Panasonic absolute, 2=HD systems, 3=Panasonic Incremental, 4=Sanyo Denki, 5=Tamagawa Single Turn)</td></tr>
<tr><td>28</td><td>Bit rate (set for 4 Mbit, clear for 2.5 Mbit)</td></tr>
<tr><td>29</td><td>If set, use multi-mode encoder.
If clear, use a primary encoder.</td></tr>
<tr><td>30</td><td>If set, treat encoder battery errors as warnings.</td></tr>
<tr><td>31</td><td>Read the encoder's internal temperature sensor. Currently for Sanyo Denki and Panasonic encoders. Temperature value read from encoder can be read as encoder register 0.</td></tr>
<tr><td colspan="2">Incremental Type E (Type 15)</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>If set, incremental encoder.
If clear, absolute encoder.</td></tr>
<tr><td>8</td><td>If set, disable interpolation of position</td></tr>
<tr><td colspan="2">Gurley Virtual Absolute (Type 17)</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>If set, invert sine/cosine signals</td></tr>
<tr><td>1</td><td>If set, invert virtual absolute signal</td></tr>
<tr><td>2</td><td>If set, use custom interface board (customer specific)</td></tr>
<tr><td>3</td><td>If set, use encoder digital index input for VABS. If clear, use encoder analog index (if available).</td></tr>
<tr><td>8</td><td>If set, switch from algorithmic phase initialization to encoder-based phasing as soon as absolute position is found</td></tr>
<tr><td>9</td><td>If set, treat any VABS warnings as encoder fault. If clear, these warnings set status bits but aren’t treated as encoder errors.</td></tr>
<tr><td colspan="2">Custom Absolute Encoder (Type 18)</td></tr>
<tr><td><b>Bit</b></td><td><b>Description</b></td></tr>
<tr><td>28</td><td>If set, use multimode input.
If clear, use primary encoder input.</td></tr>
<tr><td colspan="2">S2 Custom Encoder (Type 19)</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT
IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="26"></td><td rowspan="26"></td><td rowspan="26"></td><td rowspan="26"></td><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0-4</td><td>Number of bits of single turn position data / rev.</td></tr>
<tr><td>8</td><td>Set for incremental encoders, clear for absolute</td></tr>
<tr><td>9</td><td>Use multimode port if set. If clear use primary encoder interface</td></tr>
<tr><td>10</td><td>If set, treat encoder battery errors as warnings</td></tr>
<tr><td colspan="2">Wire saving incremental (type 21):</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0-15</td><td>These bits are the same as a normal incremental encoder (type 0)</td></tr>
<tr><td>16</td><td>If set, reverse direction of simulated hall signals after powerup</td></tr>
<tr><td>17</td><td>If set, sample halls 10ms after they stabilize on power-up. If clear, sample them after 100ms.</td></tr>
<tr><td>18</td><td>If set, force the simulated hall signals to transition coincident with the index signal</td></tr>
<tr><td colspan="2">Sankyo Absolute Encoder (Type 22):</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>Ignore battery errors if set</td></tr>
<tr><td>1-31</td><td>Reserved</td></tr>
<tr><td colspan="2">Custom Absolute Encoder M (Type 23)</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>Use the multimode port if set</td></tr>
<tr><td>1</td><td>Ignore battery errors</td></tr>
<tr><td colspan="2">Tachometer input (type 25):</td></tr>
<tr><td><b>Bit</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>If set, read tach from analog encoder sine input.
If clear, read from analog reference input.</td></tr>
<tr><td colspan="2">Tamagawa TS5643 (type 26):</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>If set, use the multi-mode port</td></tr>
<tr><td>1</td><td>Don’t generate faults on error bits reported by encoder</td></tr>
<tr><td>0x12B</td><td>0x2223</td><td>F</td><td>U32</td><td colspan="2">Load Encoder Options. Same details as Motor Encoder Options (0x12A) but affects load or position encoder.</td></tr>
<tr><td>0x12C</td><td>0x2384:28</td><td>R*</td><td>U32</td><td colspan="2">Nios Processor Firmware Version Number.
Currently only used on three-axis drives.</td></tr>
<tr><td>0x12D</td><td>0x2109</td><td>RF</td><td>9 or 14</td><td colspan="2">Analog Input Filter Coefficients. A bi-quad filter which acts on the analog reference input. 9- or 14-word parameters, see <i>Analog Input Filters</i> in <i>CME User Guide.</i> 14-word parameter (Plus and AFS products only), see Filter Coefficients.</td></tr>
<tr><td>0x12E</td><td>0x2224</td><td>R*</td><td>U32</td><td colspan="2">Motor Encoder Status. This parameter gives additional status information for encoder. Bits set in status word are latched and cleared when status value is read. Format of this status word is dependent on encoder type. Many error bits are taken directly from encoder data stream. For full description of what these error bits mean, please consult encoder manufacturer.</td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				Quadrature	
				Bits	**Description**
				0	Only used for custom incremental encoders. Set on startup if encoder did not transmit hall information successfully
				1	Set on bad differential signal levels on any of encoder inputs
				EnDAT (Type 11)	
				Bits	**Description**
				0	CRC error on data received from encoder
				1	Failed to detect encoder connected to drive
				2	Error bit on encoder stream is active
				3	Encoder failed to respond to request for position
				SSI (Type 12)	
				Bits	**Description**
				0-6	Fault flags returned from encoder
				15	Encoder data invalid bit set
				BiSS (Type 13)	
				Bits	**Description**
				0	CRC error on data received from encoder
				1	Encoder failed to transmit data to drive
				2	Error bit on encoder stream is active
				3	Warning bit on encoder stream is active
				4	Encoder transmission delay is too long
				Tamagawa & Panasonic (Type 14)	
				Bits	**Description**
				0	Over-speed error reported by encoder
				1	Absolute position error reported by encoder
				2	Counting error reported by encoder
				3	Counter overflow reported by encoder
				5	Multi-turn error reported by encoder
				6	Battery error reported by encoder
				7	Battery warning reported by encoder
				8	Error bit 0 reported by encoder
				9	Error bit 1 reported by encoder
				10	Comm error 0
				11	Comm error 1
				15	CRC error on data received from encoder
				Sanyo Denki & Harmonic Drives (Type 14)	
				Bits	**Description**
				0	Battery warning reported by encoder
				1	Battery error reported by encoder
				3	Over speed reported by encoder
				4	Memory error reported by encoder

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				5	STERR reported by encoder
				6	PSERR reported by encoder
				7	Busy error reported by encoder
				8	Memory busy reported by encoder
				9	Over temperature reported by encoder
				15	CRC error on data received from encoder
				Harmonic Drives (Encoder Type 15)	
				Bits	**Description**
				0	System error reported by encoder
				1	Overflow error reported by encoder
				2	Mode error reported by encoder
				3	Battery error reported by encoder
				4	CRC error on data received from encoder
				5	No data received from encoder on read
				Gurley Virtual Absolute (Encoder type 17)	
				Bits	**Description**
				0	Amplitude of Sine/ Cosine signals is out of range
				1	Encoder power current limited
				2	Encoder moving too fast during initialization
				3	Missing trigger signal (only occurs when using custom interface hardware).
				4	Virtual absolute signal changed state at incorrect time
				5	Invalid virtual absolute data received.
				6	Encoder has not finished initializing position
				Custom Absolute Encoder K (Type 18)	
				Bits	**Description**
				0	Busy bit from encoder set
				1	ABSALM bit from encoder set
				2	INPALM bit from encoder set
				8	CRC error on data received from encoder
				S2 Custom Encoder (Type 19)	
				Bits	**Description**
				0	Battery error alarm bit from encoder
				1	Encoder error alarm bit from encoder
				2	Battery warning alarm bit from encoder
				3	Absolute error alarm bit from encoder
				4	Over speed error alarm bit from encoder
				5	Overheat error alarm bit from encoder
				8	CRC error on data received from encoder
				9	Encoder not responding to queries from drive
				Hiperface encoder (Type 20):	

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				Bits	**Description**
				0	Analog sin/cos signals missing or too low
				1	Error bit received from encoder
				2	No response received from encoder
				3	Checksum error on encoder response
				4	Digital data from encoder doesn't agree with analog angle
				Sankyo Absolute Encoder (Type 22)	
				Bits	**Description**
				0	Set if encoder is not responding to commands
				1	Set if error bit is returned by encoder
				2	Set if encoder returns incorrectly formatted data
				3-7	Reserved
				8	Encoder reports "MR sensor amplitude error"
				9	Encoder reports "Multi rotation data error"
				10	Encoder reports "battery error"
				11	Encoder returned reserved error bit
				12	Encoder reports "MR sensor error"
				13	Encoder reports "Over speed error"
				14	Encoder reports "Temperature error"
				15	Encoder returned reserved error bit
				Custom Absolute Encoder M (Type 23)	
				Bits	**Description**
				0	Encoder reported "CPU alarm"
				1	Encoder reported "reserved alarm"
				2	Encoder reported "Data alarm"
				3	Encoder reported "Thermal alarm"
				4	Encoder reported "Thermal warning"
				5	Encoder reported "Multi revolution Alarm"
				6	Encoder reported "Absolute position lost warning"
				7	Encoder reported "Battery disconnect"
				8-12	Reserved
				13	Incorrect data type returned from encoder
				14	Encoder not responding to reads
				15	Encoder CRC data error
				Tamagawa TS5643 (type 26):	
				Bits	**Description**
				0	Encoder reported a "battery error"
				1	Encoder reported an "overflow error"

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
				2 Encoder reported an “over-speed error” 3 Encoder reported a “battery alarm” 4 Encoder preload status bit 5 Encoder reported a counter error 8 CRC error reading data from encoder 9 Encoder is not sending data
0x12F	0x2225	R*	U32	Load Encoder Status. Same details as Motor Encoder Status (0x12E), but for load encoder.
0x130	0x2114	RF	INT16	RMS Current Calculation Period. Units: ms. This sets period over which RMS current is calculated. If this value is set to zero, then RMS current will be updated each time it is read for period since the last read. In this case, RMS current must be read at least once every 65536 current loop periods (about every 4 seconds) for returned RMS values to be accurate.
0x131	0x2115	R*	INT16	RMS Current Value. Units: 0.01 A. See RMS Current Calculation Period (0x130).
0x132	0x2116	R*	INT16	Running Sum of User Current Limit. Units: 0.01%. Values will be 0 to 10000 (100 %).
0x133	0x2117	R*	INT16	Running Sum of Drive Current Limit. Units: 0.01% Values will be 0 to 10000 (100 %).
0x134	0x21E0	RF	U32	Analog Output D/A converter configuration. This parameter sets mode for D/A converter on drives with an analog output. **Bits** **Description** 0-3 Defines mode of D/A converter 16 If set, current outputs will be scaled based on motor peak current setting rather than drive's internal scaling. Currently supported modes are: **Mode** **Description** 0 Manual configuration. Set using Analog Output D/A (0x135) 1 Actual Current of configured axis. If bit 16 is clear, then output voltage is scaled so that full 5V output on D/A will correspond to Current Corresponding to Max A/D Reading (0x84). If bit 16 is set, then voltage is scaled based on motor peak current setting. 2 Actual Velocity of configured axis, ratio of actual velocity to Velocity Loop Velocity Limit (0x3A) 3 U winding current, scaled same as mode 1 4 V winding current, scaled same as mode 1 5 W winding estimated current, scaled same as mode 1
0x135	0x21E1	R	INT16	Analog Output D/A Converter Output Value. Units: mV. For drives that support auxiliary D/A converter, this sets output value when D/A is in manual mode. In other modes, current value being output on D/A can be read here.

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th>Description</th></tr>
<tr><td>0x136</td><td>0x2208</td><td>R*</td><td>INT16</td><td>Second Analog Input. Units: mV.
Also known as Secondary analog reference value</td></tr>
<tr><td>0x137</td><td>0x2314</td><td>RF</td><td>INT16</td><td>Offset for Second Analog Input (Secondary analog reference value). Units: mV.</td></tr>
<tr><td>0x138</td><td>0x2315</td><td>RF</td><td>INT16</td><td>Calibration offset, second analog input. Units: mV.
Factory-calibrated to give zero reading for zero input voltage.</td></tr>
<tr><td>0x139</td><td>0x219D</td><td>R</td><td>INT32</td><td>Drive Safety Circuit Status (STO).
This parameter allows status of safety circuit built into some drives to be queried. For drives without safety circuit, this parameter is reserved.
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0</td><td>Set when safety Input 0 (STO-IN1) is preventing drive from enabling.</td></tr>
<tr><td>1</td><td>Set when safety Input 1 (STO-IN2) is preventing drive from enabling.</td></tr>
<tr><td>8</td><td>This read/write bit can be used to force ‘drive is unsafe’ output of safety circuit to go active for testing purposes. Write 1 to force this output active. Write zero for normal operation.</td></tr>
<tr><td>16-19</td><td>On the NxS drives these bits give information about the safety
circuit status transmitted from the safety micro-controller to
the main processor. Bit-mapped as follows:
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0</td><td>Working normally</td></tr>
<tr><td>1</td><td>Timeout waiting for safety status info from micro</td></tr>
<tr><td>2</td><td>Invalid status infor received from micro</td></tr>
<tr><td>8-15</td><td>Safety micro is reporting a failure code. The code is stored in the lower three bits of this field.</td></tr>
</table></td></tr>
</table></td></tr>
<tr><td>0x13A</td><td>0x2209</td><td>R*</td><td>INT16</td><td>Present Voltage at Analog Motor Temperature Sensor. Units: mV.
If thermistor characteristics have been programmed in Steinhart Contants (0x19A), then temperature is returned in degrees C. (This parameter is currently under development and is reserved for future use.)
Note that this parameter is only valid for drives that include an analog temperature sensor input.</td></tr>
<tr><td>0x13B</td><td>0x220A</td><td>RF</td><td>INT16</td><td>Limit for Analog Motor Temperature Sensor.
Units: mV.
If this parameter is set to zero, then analog motor temperature sensor is disabled.
If this parameter is set to positive value, then motor temperature error will occur any time voltage on motor temperature input exceeds this value.
If this parameter is set to negative value, then motor temperature error will occur any time voltage on the</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td>motor temperature input is lower than absolute value of this limit.
If thermistor characteristics have been programmed in Steinhart Constants (0x19A), then this gives maximum motor temperature in degrees C. (This parameter is currently under development and is reserved for future use.)</td></tr>
<tr><td>0x13C</td><td>0x2323</td><td>RF</td><td>INT16</td><td>Minimum PWM Pulse Width. Units: ms.
Used when running in PWM position mode. In this mode PWM input pulse width is captured by drive and used to calculate an absolute position using the following formula:
pos = ((PW-MIN) / (MAX-MIN)) * SCALE + OFFSET
Where this parameter is minimum pulse width (MIN), <i>parameter 0x13D</i> is maximum pulse width (MAX), <i>parameter 0xA9</i> is scaling factor (SCALE) and <i>parameter 0x10F</i> is offset (OFFSET).</td></tr>
<tr><td>0x13D</td><td>0x2324</td><td>RF</td><td>INT16</td><td>Maximum PWM Pulse Width. Units: us.
Used only when running in PWM position mode.</td></tr>
<tr><td>0x13E</td><td>0x222A</td><td>RF</td><td>U32</td><td>Encoder Adjustment Table Configuration.
See applications note for additional details.
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0</td><td>Set to enable encoder adjustment table.</td></tr>
<tr><td>1</td><td>If set, use resolver angle adjustment tables. If clear, use normal encoder adjustment tables.</td></tr>
</table></td></tr>
<tr><td>0x13F</td><td>0x232B</td><td>RF</td><td>INT16</td><td>PWM Input Deadband. Range of 0 to 32767 equals deadband of 0 to 100%.
This parameter was added to Plus drives starting with version 2.75.</td></tr>
<tr><td>0x141</td><td>0x2243</td><td>R</td><td>INT16</td><td>Resolver angle scaled so 180 deg is 32767.
Only valid when using resolver as motor encoder feedback. Reserved for other encoder types.</td></tr>
<tr><td>0x142</td><td>None</td><td>RF</td><td>INT32</td><td>This parameter is used in ARM based drives to support backward compatibility options to make them more consistent with the obsolete DSP based drives that they replaced. Bit-mapped as follows:
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0</td><td>If set, then don’t generate a phase error for invalid hall states (000 or 111).</td></tr>
<tr><td>1</td><td>If set, limit PVT buffer size to 32 points.</td></tr>
</table></td></tr>
<tr><td>0x143</td><td>0x2302</td><td>RF</td><td>U16</td><td>Watchdog Timeout (in ms). If non-zero, then an error will occur if a serial port command hasn’t been received within this much time. When such an error occurs, the drive will be disabled.
This parameter is supported on Plus drives starting with version 4.18 firmware.</td></tr>
<tr><td>0x150</td><td>0x210A</td><td>RF</td><td>14</td><td>Second chained bi-quad filter on output of velocity loop. For 14-word parameter, see Filter Coefficients.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>CAN/ECAT
IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="3">Description</th></tr>
<tr><td>0x151</td><td>0x210B</td><td>RF</td><td>14</td><td colspan="3">Third chained bi-quad filter on output of velocity loop. For 14-word parameter, see Filter Coefficients.</td></tr>
<tr><td>0x152</td><td>0x210C</td><td>RF</td><td>14</td><td colspan="3">First chained bi-quad filter on input of current loop. For 14-word parameter, see Filter Coefficients.</td></tr>
<tr><td>0x153</td><td>0x210D</td><td>RF</td><td>14</td><td colspan="3">Second chained bi-quad filter on input of current loop. For 14-word parameter, see Filter Coefficients.</td></tr>
<tr><td rowspan="8">0x154</td><td rowspan="8">0x2301</td><td rowspan="8">RF</td><td rowspan="8">INT32</td><td colspan="3">Servo Loop Configuration. This parameter allows various parts of drive servo loops to be enabled/disabled. Bit-mapped as follows:</td></tr>
<tr><td><b>Bits</b></td><td colspan="2"><b>Description</b></td></tr>
<tr><td>0</td><td colspan="2">If set, this disables Velocity loop gains. Velocity Feed Forward (0x157) is still active as are velocity loop output filters.</td></tr>
<tr><td>1</td><td colspan="2">If set, this enables <i>Position Loop I (0x155)</i> and <i>Position Loop D (0x156)</i> gains.
If clear, these are treated as zeros.</td></tr>
<tr><td>2</td><td colspan="2">If set, velocity error windows will be calculated using filtered version of the motor velocity.
If clear, unfiltered velocity will be used.</td></tr>
<tr><td>3</td><td colspan="2">If set, the velocity loop will be used to stop the motor when the drive is disabled. If clear, the position loop will be used in velocity mode.</td></tr>
<tr><td>4</td><td colspan="2">If set, the analog reference input can be used to add a current offset. Parameter 0x19 is used to scale the current in the same way it would be used when running in mode 2. For Plus drive firmware 4.48 and later.</td></tr>
<tr><td>Other</td><td colspan="2">Reserved</td></tr>
<tr><td>0x155</td><td>0x2382:5</td><td>RF</td><td>INT16</td><td colspan="3">Position Loop Integral Gain (Pi).</td></tr>
<tr><td>0x156</td><td>0x2382:6</td><td>RF</td><td>INT16</td><td colspan="3">Position Loop Derivative Gain (Pd)</td></tr>
<tr><td>0x157</td><td>0x2381:6</td><td>RF</td><td>INT16</td><td colspan="3">Velocity Loop Command Feed Forward (Vcff).

Input command (after limiting) to velocity loop is scaled by this value and added into output of velocity loop.</td></tr>
<tr><td>0x158</td><td>0x2382:7</td><td>RF</td><td>INT16</td><td colspan="3">Position Loop Integral Drain (Pi Drain).</td></tr>
<tr><td>0x159</td><td>0x6007</td><td>RF</td><td>INT16</td><td colspan="3">Abort Option Code, CANopen/EtherCAT drives.</td></tr>
<tr><td rowspan="7">0x15A</td><td rowspan="7">0x2198</td><td rowspan="7">RF</td><td rowspan="7">U32</td><td colspan="3">I/O Options. This parameter is used to configure optional features of general purpose I/O.</td></tr>
<tr><td><b>Bits</b></td><td colspan="2"><b>Description</b></td></tr>
<tr><td rowspan="3">0-3</td><td colspan="2">For Plus drives, these bits determine whether several I/O pins are used as serial interface for expanded I/O features, and how they are configured.</td></tr>
<tr><td>0</td><td>Normal I/O</td></tr>
<tr><td>1</td><td>Plus drive development board LEDs and address switches</td></tr>
<tr><td>4-7</td><td colspan="2">Reserved</td></tr>
<tr><td>8</td><td colspan="2">For Plus drives, setting this bit allows the STO LED to be illuminated even if the drive is disabled by firmware if the STO inputs are connected.</td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				9-15	Reserved
				16	On AC powered Plus drives, this bit disables AC line drop detection if set.
				17	Reserved
				18	Starting with firmware 4.40, setting this bit causes the firmware to switch debounce the safety status for 3ms. If the safety input causes the drive to be disabled for less than 3ms, the firmware keeps working normally and will not abort moves or perform any other actions.
				19-31	Reserved
0x15B	0x2199	F	INT16	Motor Brake Enable Delay Time. Units: ms. This parameter gives delay between enabling drive PWM outputs and releasing brake. Positive values mean PWM is enabled first and brake is released later. Negative values cause brake to be released before PWM outputs are enabled.	
0x15C	0x219A	R*	U32	Input Pin States, 32-bit. 32-bit version of Input Pin States (0xA6). Each bit gives high/low state of one general purpose input pin. Lower 16 bits of this parameter are equivalent to value returned by Input Pin States (0xA6). This parameter is primarily used for drives with more than 16 general purpose input pins.	
0x15D	0x219B	R*	U32	Raw Input State, 32-bit. 32-bit version of Raw Input State (0xAA). Gives current high/low state of all general-purpose inputs before any switch debounce is applied.	
0x15E	0x219C	RF	U32	Input Pin Configuration, 32-bit. 32-bit version of Input Pin Configuration (0xA5). Used to configure pull up/down resistors on drives with more than 16 such resistors.	
0x15F	0x237B	RF	U32	Motor Cogging Compensation. This was added to Plus drives starting with version 3.18 firmware. Scales current command to motor based on sine of phase angle plus programmable offset. Bit-mapped as follows:	
				Bits	**Description**
				0-7	Gives an angular offset in units of 360/256 degrees.
				8-15	Reserved
				16-31	Gives scaling value. Scale = 1.0 + X/16384 where X is unsigned value programmed in these bits. Resulting scale ranges from 0 <= scale < 5.0.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
0x160	0x2192:17	RF	U16	Input Pin Configuration, General Purpose Input 17. See Input 0 (IN1) Configuration (0x78).
0x161	0x2192:18	RF	U16	Input Pin Configuration, General Purpose Input 18. See Input 0 (IN1) Configuration (0x78).
0x162	0x2192:19	RF	U16	Input Pin Configuration, General Purpose Input 19. See Input 0 (IN1) Configuration (0x78).
0x163	0x2192:20	RF	U16	Input Pin Configuration, General Purpose Input 20. See Input 0 (IN1) Configuration (0x78).
0x164	0x2192:21	RF	U16	Input Pin Configuration, General Purpose Input 21. See Input 0 (IN1) Configuration (0x78).
0x165	0x2192:22	RF	U16	Input Pin Configuration, General Purpose Input 22. See Input 0 (IN1) Configuration (0x78).
0x166	0x2192:23	RF	U16	Input Pin Configuration, General Purpose Input 23. See Input 0 (IN1) Configuration (0x78).
0x167	0x2192:24	RF	U16	Input Pin Configuration, General Purpose Input 24. See Input 0 (IN1) Configuration (0x78).
0x170	0x2195:17	RF	U16	Switch Debounce Time, General Purpose Input 17. Units: ms.
0x171	0x2195:18	RF	U16	Switch Debounce Time, General Purpose Input 18. Units: ms.
0x172	0x2195:19	RF	U16	Switch Debounce Time, General Purpose Input 19. Units: ms.
0x173	0x2195:20	RF	U16	Switch Debounce Time, General Purpose Input 20. Units: ms.
0x174	0x2195:21	RF	U16	Switch Debounce Time, General Purpose Input 21. Units: ms.
0x175	0x2195:22	RF	U16	Switch Debounce Time, General Purpose Input 22. Units: ms.
0x176	0x2195:23	RF	U16	Switch Debounce Time, General Purpose Input 23. Units: ms.
0x177	0x2195:24	RF	U16	Switch Debounce Time, General Purpose Input 24. Units: ms.
0x180	0x2326	RF	U32	UV configuration. Used to configure drive when running in UV mode, Desired State (0x24), Mode 5. Bit-mapped as follows:

Bits	Meaning		
0-1	Define source of UV command inputs:		
		Value	Description
		0	PWM inputs
		1	Analog reference inputs (for drives with two analog reference inputs)
		2	Analog encoder inputs.
		3	Directly set over serial/network interface
2-7	Reserved		
8-9	Define format of UV inputs:		
		Value	Description
		0	120 degree current commands
		1	90 degree current commands
		2	Angle/Magnitude form. U input gives magnitude. V gives angle.
10-15	Reserved		

<table>
<tr><th>ASCII</th><th>CAN/ECAT IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td>16</td><td>If set, value of Motor Hall Offset (0x4F) is added to UV angle</td></tr>
<tr><td>17</td><td>If set, drive will use field oriented control. Normally FOC is disabled in UV mode due to ambiguity of phase angle with zero inputs. This is best used when running in angle/magnitude format.</td></tr>
<tr><td>0x181</td><td>0x2327</td><td>R</td><td>INT16</td><td colspan="2">U input when running in UV mode.

This parameter can be used to read calculated U value or to set U value when UV inputs are being directly set over serial/network interface.</td></tr>
<tr><td>0x182</td><td>0x2328</td><td>R</td><td>INT16</td><td colspan="2">V input when running in UV mode.
Same as 0x181 but for V Input.</td></tr>
<tr><td>0x183</td><td>0x2329</td><td>R</td><td>INT16</td><td colspan="2">Raw Counter Value From Pulse & Direction Input.

This can be read when running in any mode, not just pulse & direction modes. This parameter can be written as well, but should not be written when drive is being controlled by pulse & direction inputs. Writing in that mode will cause drive to treat change in counter as real pulse inputs resulting in possible unexpected motion.</td></tr>
<tr><td>0x184</td><td>0x2254</td><td>RF</td><td>8 to 40</td><td colspan="2">Input Shaping Filter.

This filter is used to modify trajectory before it is input into position loop. This can be used to compensate for low frequency resonances in loads.

Parameter is an array of 32-bit values. First four values are used to store information about input shaping filter (filter type, frequency, etc.) and are mostly unused by firmware. The only exception is that most significant bit (MSB) of first word should not be set to ensure compatibility with future firmware versions.

The remaining 32-bit values are pairs of IEEE floating point values. Each pair defines a time (first value) and an impulse amplitude (second value).
Up to eight pairs may be passed for up to 8 impulses in input shaping filter.
Time values are specified in seconds and must be >= 0.0. Impulse values are unit-less and must have an absolute magnitude of &lt; 16.0.</td></tr>
<tr><td rowspan="5">0x185</td><td rowspan="5">0x2160</td><td rowspan="5">R</td><td rowspan="5">U32</td><td colspan="2">Output Compare Configuration Module. Used to configure hardware triggered output pulses at position. For software triggered output at position see Output Configuration (0x70).

For detailed description of output compare function, see [Setting Outputs at Position, AN137] application note.</td></tr>
<tr><td><b>Bits</b></td><td><b>Description</b></td></tr>
<tr><td>0</td><td>If set, enables module.</td></tr>
<tr><td>1</td><td>If set, inverts normal active state of output. E.g. outputs that are normally active-low become active-high.</td></tr>
<tr><td>2</td><td>If set, toggle output on compare match. If clear, pulse output for programmable time.</td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				3-4	Define mode of compare module. See below.
				5-15	Reserved
				16-17	Selects which encoder to use for position comparisons. See below.
				18-31	Reserved
				For firmware versions 4.18 and later, bits 16-17 can be used to select which encoder to use for hardware-triggered outputs when dual encoders are being used.	
				Value	**Encoder**
				0	Use the encoder that feeds the position loop. This is the default for earlier firmware versions.
				1	Always use the motor encoder, even on dual encoder setups.
				2	Always use the load encoder, even if it is passive.
				3-4	Reserved
0x186	0x2161	R	U32	Compare Module Status Register. Bit-mapped as follows:	
				Bits	**Description**
				0	Current value of compare output (read only).
				1	If set, position matches compare register 0. Write 1 to clear.
				2	If set, position matches compare register 1. Write 1 to clear.
				3-31	Reserved
0x187	0x2162	R	INT32	Output Compare Value 0.	
0x188	0x2163	R	INT32	Output Compare Value 1.	
0x189	0x2164	R	INT32	Output Compare Increment. Signed 32-bit value used to update compare values in some modes.	
0x18A	0x2165	R	INT32	Output Compare Pulse Width. The lower 20-bits of this parameter give the period of the compare output pulse in 10ns units.	
0x18B	0x2255	RF	INT32	Trajectory Options. This parameter is used to modify behavior of some trajectory modes. Interpretation depends on trajectory mode being used. The following trajectory modes currently make use of this parameter:	
				EtherCAT CSP mode:	
				Bits	**Description**
				0-7	Number of extra loop cycles to extrapolate trajectories if input data from master is not received.
				8-15	Reserved
				16	If set, jump to quick stop mode if master data is not received within number of cycles set in bits 0-7.

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description	
				17	If set, and *Interpolation Time object* (0x60C2) is non-zero, then calculated velocity will be filtered, and trajectory acceleration will also be calculated. If clear, velocity is unfiltered, and acceleration is not calculated (zero).
				18-31	Reserved
0x18C	0x21A1	RF	U32	I/O Extension Configuration for Plus Modules. This parameter is used to configure I/O extension feature on Plus Modules which support it. For detailed description of this I/O extension feature, see I/O Extension Features in Copley Modules, AN102 application note.	
				Bits	**Description**
				0-7	Number of bits to transfer less 1 (e.g., set to 19 to transfer 20 bits).
				8	Reserved
				9	If set, automatically restart transmission.
				10	If set, leave CS line low after transfer.
				11	Status bit indicating new receive data is available. Auto-cleared when data is read via parameter 0x18E
				12	Clock polarity setting
				13	Data phase setting
				14-15	Reserved
				16-23	Clock period. Units: 100 ns.
				24-27	Reserved
				28	If set, enable SPI I/O extension feature. If clear, enable LED/Switch interface
				29-31	Reserved
0x18D	0x21A2	R	INT32*	I/O Extension Transmit Data. Data to be transferred over SPI port is sent immediately after being written here. Refer to Extending Plus Module I/O AN102 application note.	
0x18E	0x21A3	R	INT32*	I/O Extension Receive Data. Afer transimissoin, data received from SPI port can be read here. Refer to Extending Plus Module I/O AN102 application note.	
0x18F	0x220B	RF	INT16	Encoder Sine Offset. This is set in A/D units and only used with resolvers and servo-tube motors. It gives an offset which is added to encoder sine signal before calculating position. Note that parameter 0x191 must be non-zero for this to be used.	
0x190	0x220C	RF	INT16	Encoder Cosine Offset. Similar to 0x18F, but for encoder cosine signal.	
0x191	0x220D	RF	U16	Encoder Cosine Scaling Factor. Used by resolver & Servotube encoder calculations. This scaling factor is used to adjust cosine signal amplitude so it is same as sine signal amplitude.	

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
				If set to zero, both Encoder Sine Offset (0x18F) and Encoder Cosine Offset (0x190) will be ignored. If non-zero the cosine is scaled by N/32768 where N is the value of this parameter.
0x192	0x2226	RF	U32	Motor Encoder Calibration settings. The meaning of this value is dependent on encoder type. See Motor Encoder Options (0x12A) for motor encoder type.
0x193	0x2227	RF	U32	Load Encoder Calibration settings. Same as 0x192, but applied to load encoder. See Load Encoder Options (0x12B) for load encoder type.
0x194	0x232A	R*	INT16	PWM Input Duty Cycle. This can be used to read duty cycle of PWM input. Returned 16-bit value gives duty cycle in range +/-32767. Digital Input Command Configuration (0xA8) is used to configure PWM input.
0x195	0x2123	RF	INT32	Jerk Abort Value. Units: 100 counts/s^3. Value to use during trajectory aborts. If this is zero, abort will be calculated without any jerk limits.
0x196	0x220E	R*	INT32	Returns magnitude squared of analog encoder signals (sin*sin + cos*cos)
0x197	0x2378	RF	INT16	Cross Coupling XPp Gain. On dual axis drives this gain is applied to difference in position error of two axes.
0x198	0x2379	RF	INT16	Cross Coupling XPi Gain. On dual axis drives this gain is applied to difference in position error of two axes.
0x199	0x237A	RF	INT16	Cross Coupling XPd Gain. On dual axis drives this gain is applied to difference in position error of two axes.
0x19A	0x220F	RF	5 words	Reserved.
0x19B	0x2384:30	F*	INT16	Current at which minimum PWM deadtime is used.
0x19C	0x2406	R*	INT32	High-Speed Position Capture, Passive Load Encoder.
0x19D	0x2142	RF	INT16	Motor Wire Open Circuit Test. If Motor Brake Enable Delay Time (0x15B) is greater than zero, then during that time period on enable this current will be applied to motor wiring to check that motor is connected. If programmed current cannot be applied to motor, then a motor disconnected fault will be flagged.
0x19E	0x6066	RF	U16	Position Tracking Window Warning Time. Units: ms.
0x19F	0x2264	RF	INT16	Phase Advance. Scaled so 32000 is 180 degrees. Adjusted using gain scheduling with key parameter absolute value of actual velocity. This produces field weakening thereby increasing the motor's top speed.
0x1A0	0x2193:9	RF	3-5	Output 8 (OUT9) Configuration. See Output 0 (OUT1) Configuraton (0x70).
0x1A1	0x2193:10	RF	3-5	Output 9 (OUT10) Configuration. See Output 0 (OUT1) Configuraton (0x70).
0x1A2	0x2193:11	RF	3-5	Output 10 (OUT11) Configuration. See Output 0 (OUT1) Configuraton (0x70).

<table>
<tr><th>ASCII</th><th>CAN/ECAT
IDX: SUB</th><th>Mem</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td>0x1A3</td><td>0x2193:12</td><td>RF</td><td>3-5</td><td colspan="2">Output 11 (OUT12) Configuration.
See Output 0 (OUT1) Configuraton (0x70).</td></tr>
<tr><td>0x1A8</td><td>0x2228</td><td>RF</td><td>INT16</td><td colspan="2">Motor Encoder Downshift.

This parameter is useful when using very high resolution encoders that would otherwise have limited speed and travel distance due to range of INT32 position and velocity parameters. Setting downshift causes position read from encoder to be right-shifted before being used.

For example, setting this parameter to value of 2 effectively cuts the encoder resolution by a factor of 4. If set, servo loops use fractional encoder counts, therefore encoder resolution is not completely lost.

Consult factory for development of FP32 floating point velocity, acceleration, deceleration, and jerk parameters.</td></tr>
<tr><td>0x1A9</td><td>0x2229</td><td>RF</td><td>INT16</td><td colspan="2">Load Encoder Downshift.
Same as Motor Encoder Downshift (0x1A8), but for load encoder.</td></tr>
<tr><td>0x1AA</td><td>0x21E2</td><td>RF</td><td>INT16</td><td colspan="2">Fan Turn On Temperature. Units: Degrees C.
For products with software controlled internal fan, this value is temperature when fan will first turn on.</td></tr>
<tr><td>0x1AB</td><td>0x21E3</td><td>RF</td><td>INT16</td><td colspan="2">Fan Max Speed Temperature. Units: Degrees C.

For products with software controlled internal fan, this value is temperature when fan will run at top speed.

Must be >= Fan Turn On Temperature (0x1AA) or value will be ignored</td></tr>
<tr><td>0x1AD</td><td>0x21E4</td><td>RF</td><td>INT16</td><td colspan="2">Encoder Cosine Angular Offset. Units: 0.1 degree

This parameter gives angular error of encoder cosine signal. Used to compensate for imperfections in encoder signals. This adjustment is only used if Encoder Cosine Scaling Factor (0x191) is non-zero.</td></tr>
<tr><td rowspan="5">0x1AE</td><td rowspan="5">0x21A4</td><td rowspan="5">RF</td><td rowspan="5">U32</td><td colspan="2">Inter-drive communication configuration.
This parameter is only used on drives that support the IDC bus.</td></tr>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0</td><td>Set for IDC master.
Clear for IDC slave devices</td></tr>
<tr><td>1</td><td>Set to disable serial command forwarding via IDC</td></tr>
<tr><td>8-10</td><td>Address of partner axis for cross coupling</td></tr>
<tr><td rowspan="7">0X1AF</td><td rowspan="7">0x21A5</td><td rowspan="7">R</td><td rowspan="7">U32</td><td colspan="2">Inter-drive communication status.</td></tr>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0</td><td>Synchronized to IDC bus if set</td></tr>
<tr><td>1</td><td>Address assignment complete if set</td></tr>
<tr><td>2</td><td>IDC running normally if set</td></tr>
<tr><td>8</td><td>Set if IDC is reset</td></tr>
<tr><td>16-18</td><td>Assigned IDC address</td></tr>
<tr><td>0x1B0</td><td>None</td><td>RF</td><td>U32</td><td colspan="2">Common device profile warning mask.
This parameter is not actually used.</td></tr>
</table>

ASCII	CAN/ECAT IDX: SUB	Mem	Type	Description
0x1B1	None	RF	U32	Common device profile error mask. This parameter is not actually used.
0x1B2	None	R*	U32	Absolute position from motor encoder. This is the value read from the encoder and isn't affect by homing or setting the actual position.
0x1B3	None	R*	U32	Absolute position from the load encoder.

4. Filter Coefficients

There are several drive parameters which are used to define filters. These filters are implemented as generic bi-quadratic filter structures. Filters of this type implement the following formula to transform the input parameter x(n) at time n to an output parameter y(n):

$$y(n) = b_0x(n) + b_1x(n-1) + b_2x(n-2) + a_1y(n-1) + a_2y(n-2)$$

Values a_1, a_2, b_0, b_1, b_2 are constants known as filter coefficients. They define the type of filter being implemented.

Values passed to these drive filter parameters are used to define filter coefficients. Formatting of these parameters varies depending on drive product family being interfaced to.

All first-generation Copley drives use 16-bit integer math to implement their filters internally. Filter coefficients are given as 16-bit signed integer values. To increase resolution of these coefficients, an additional unsigned scaling coefficient (k) is also specified. Actual filter formula used within these drives is as follows:

$$y(n) = \frac{K}{32{,}768 * 4{,}096} * (b_0x(n) + b_1x(n-1) + b_2x(n-2) + a_2y(n-2))$$

To set filter coefficients on drives of this category, 9 words of parameter data are passed. The first three words of data are informational parameters which are used by CME software to describe the filter. If the upper 3 bits of the first word are all set, then filter will be disabled. Otherwise, the first three words of data are not used in any way by the firmware. These three words are reserved for CME use.

Word	Description
1	Filter info. Set to 0xFFFF to disable filter. Otherwise, reserved for CME use.
2	Filter info. Reserved for CME use.
3	Filter info. Reserved for CME use.
4	b_2 coefficient
5	b_1 coefficient
6	b_0 coefficient
7	a_2 coefficient
8	a_1 coefficient
9	K scaler

For Plus family of drives (Accelnet Plus, Stepnet Plus, Xenus Plus, AEM), a new format is used to describe bi-quad filter coefficients. These drives include ability to design filters in firmware using Cephes filter design library (http://www.netlib.org/cephes/ellf.tgz).

Filters on these families of drives are calculated internally using 32-bit IEEE floating point coefficients. Format of parameter information passed when setting filter parameters on these drives consists of an array of up to fourteen 16-bit words. First 4 words describe filter and remaining 10 words give filter coefficients as 32-bit IEEE floating point values. Filter coefficient words are optional and are only necessary if firmware is not calculating coefficients internally.

Word	Description	
1	**Bits**	**Usage**
	0-3	Filter family
	4	If set, filter will not be designed. Always set by firmware after successfully designing filter. This prevents filter from being redesigned when copied from flash at startup.
	5-7	Reserved
	8	Number of poles – 1 (i.e. 0 for single pole, 1 for two pole)
	9-12	Reserved
	13-15	Filter type
	All reserved bits should be set to zero. Filter family should be one of following values:	
	0	Custom Bi-quad filter. Coefficients must be passed; firmware will not design filter.
	1	Butterworth filter
	2	Chebychev filter
	3	Elliptic filter
	4-15	Reserved
	Filter type should be one of the following:	
	0	Custom Bi-quad filter. Coefficients must be passed; firmware will not design filter.
	1	Low pass
	2	High pass
	3	Band reject (notch)
	4	Band pass
	5-6	Reserved
	7	Disabled. The filter will have no effect in system.
	If legal values are passed for filter type and family, the firmware will attempt to design specified filter and fill in coefficient values itself. Firmware can calculate 1- or 2-pole low-pass or high-pass filters. For notch and band pass filters firmware can only calculate 2-pole filter. For these filter types, bit 8 must be set.	
2	This word gives cut off frequency for low pass and high pass filters. Units: Hz. For notch and band pass filters this gives first filter frequency.	
3	This word gives second filter frequency for notch and band pass filters. Units: Hz.	
4	**Bits**	**Usage**
	0-7	Rp. Units: 0.1 dB
	8-15	Rs. Units: dB
	Rp is pass band ripple. This parameter is only used for Chebychev and Elliptic filters. Rs used only with elliptic filters. Defines stop band as Rs dB down from peak value in pass band.	
5-6	Coefficient a1. All filter coefficients are passed as 32-bit IEEE floating point numbers. The upper 32-bits should be passed first. If firmware designs filter, then coefficients will be filled in by firmware and need not be passed.	
7-8	Coefficient a_2	
9-10	Coefficient b_0	
11-12	Coefficient b_1	
13-14	Coefficient b_2	

Parameter Dictionary

P/N 16-01091

Revision 04

March 2023

Copley Controls
20 Dan Road
Canton, MA 02021 USA