Parameter Dictionary

copley
controls

Part Number 16-01091

Revision 01

September 2014

TABLE OF CONTENTS

ABOUT THIS MANUAL

Overview and Scope

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

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*

Also of related interest:

- *CME 2 User Guide*
- *Copley Indexer 2 Program User Guide* (describes use of Indexer 2 Program to create motion control sequences)
- *Copley ASCII Interface Programmer's Guide* (describes how to send ASCII format commands over an RS232 serial bus to control one or more drives)
- *Copley Camming User Guide* (describes the use of the Copley Controls Camming feature, and its setup through CME 2)
- *Extending Plus Module I/O* application note.
- Setting Outputs at Position application note.

Links to these publications, along with hardware manuals and data sheets, can be found under the *Documents* heading at http://www.copleycontrols.com/Motion/Downloads/index.html

Copley Controls software and related information can be found under the *Software* heading of the same page.

Comments

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

Copyrights

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.

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 CAN or DeviceNet node, the use of CME 2 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.

Revision History

Revision	Date	Comments
00	December 2013	Added new parameters and fixed existing content.
01	September 2014	Fixed units for parameter 0x5e

CHAPTER

1: INTRODUCTION

1.1: 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.

1.2: Organization of the Parameter Listings

The parameters are listed in tables consisting of the following columns:

The **ASCII** 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 **DvcNet** column contains the parameter's DeviceNet ID. 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 **MACRO** column contains the parameter's MACRO I-variable ID. The MACRO I-variable ID of a parameter is offset from the ASCII Interface parameter ID by decimal 1024 (hex 0x400).

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

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

- String and 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 DeviceNet variable ID, MACRO I-variable ID, and CANopen (and EtherCAT) object index and sub-index.

1.3: Important Notes

1.3.1: CME 2 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 2 session.

1.3.2: Input/Output Numbering

Inputs and Outputs on Copley drives are numbered starting from zero for all of the communication interfaces listed in this document. If a drive has 12 inputs, they are numbered 0 through 11. CME 2 software starts numbering at 1 (input 0 is called IN1 in CME 2 software).

CHAPTER

2: PARAMETERS

2.1: **Parameters Sorted by ASCII Interface Parameter ID**

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description
0x00	0x01	0x400	0x2380:1	RF	U16	Current Loop Proportional Gain (Cp).
0x01	0x02	0x401	0x2380:2	RF	U16	Current Loop Integral Gain (Ci).
0x02	0x03	0x402	0x2340	RF	INT16	Current loop programmed value. This current will be used to command the drive when Desired State (0x24) is set to 1. Units: 0.01 A.
0x03	0x04	0x403	0x2203	R*	INT16	Winding A Current. Actual current measured at winding A. Units: 0.01 A.
0x04	0x05	0x404	0x2204	R*	INT16	Winding B Current. Actual current measured at winding B. Units: 0.01 A.
0x05	0x06	0x405	0x2210	R*	INT16	Current Offset A. Offset value applied to the winding A current reading. This offset is calculated by the drive at startup. Units: 0.01 A.
0x06	0x07	0x406	0x2211	R*	INT16	Current Offset B. Offset value applied to the winding B current reading. This offset is calculated by the drive at startup. Units: 0.01 A.
0x07	0x08	0x407	0x2212	R*	INT16	X axis of calculated stator current vector. Units: 0.01 A.
0x08	0x09	0x408	0x2213	R*	INT16	Y axis of calculated stator current vector. Units: 0.01 A.
0x09	0x0A	0x409	0x221A	R*	INT16	Current loop output, X axis of stator space.
0x0A	0x0B	0x40A	0x221B	R*	INT16	Current loop output, Y axis of stator space.
0x0B	0x0C	0x40B	0x2214	R*	INT16	Actual Current, D axis of rotor space. Units: 0.01 A.
0x0C	0x0D	0x40C	0x2215	R*	INT16	Actual Current, Q axis of rotor space. Units: 0.01 A.
0x0D	0x0E	0x40D	0x2216	R*	INT16	Commanded current, D axis of rotor space. Part of the internal current loop calculation. Units: 0.01 A.
0x0E	0x0F	0x40E	0x2217	R*	INT16	Commanded Current, Q axis of rotor space. Part of the internal current loop calculation. Units: 0.01 A.
0x0F	0x10	0x40F	None	R*	INT16	Current Error, D axis of rotor space. Units: 0.01 A.
0x10	0x11	0x410	None	R*	INT16	Current Error, Q axis of rotor space. Units: 0.01 A.
0x11	0x12	0x411	None	R*	INT16	Current Integral Value, D axis of rotor space.
0x12	0x13	0x412	None	R*	INT16	Current Integral Value, Q axis of rotor space.
0x13	0x14	0x413	0x2218	R*	INT16	Current Loop Output, D axis of rotor space.
0x14	0x15	0x414	0x2219	R*	INT16	Current Loop Output, Q axis of rotor space.

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**
0x15	0x16	0x415	0x221D	R*	INT16	Commanded Motor Current. This is the value that is presently being sent to the current loop. It may come from the programmed value, analog reference, velocity loop, etc. depending on the drive's desired state. Units: 0.01 A.
0x17	0x18	0x417	0x6064	R	INT32	Actual Position. For dual encoder systems this is the load encoder position. See Actual Motor Position (0x32) for motor encoder position. Units: Counts. CANopen objects 0x6064 and 0x6063 hold the same value.
0x18	0x19	0x418	0x6069	R*	INT32	Actual Velocity. Units: 0.01 RPM for Junus, microsteps/s for Stepnet in stepper mode, 0.1 encoder counts/s for all other products.
0x19	0x1A	0x419	0x2310	RF	INT32	Analog reference scaling factor. This value is used to scale the analog reference input voltage to a command that will be used to drive the current or velocity loop (depending on drive state). When in current mode (Desired State (0x24) = 2), the value programmed specifies the amount of current to be commanded when ten volts is applied to the analog input. This value is input in 0.01 A units. For example, to command 12 amps at 10 volts the scaling factor would be 1200. When in velocity mode (Desired State (0x24) = 12), the value programmed specifies the commanded velocity corresponding to ten volts on the input. Units: 0.01 RPM for Junus, microsteps/s for Stepnet in stepper mode, 0.1 encoder counts/s for all other products. When in position mode (Desired State (0x24) = 22 or 32), the value programmed specifies the position (in encoder counts) that the drive will be commanded to for a +10 volt input.
0x1A	0x1B	0x41A	0x2311	RF	INT16	Offset Value applied to analog reference input. Units: mV.
0x1B	0x1C	0x41B	0x2205	R*	INT16	Analog Encoder Sine Input Voltage. Also known as Sine Feedback Voltage. Units: 0.1 mV.
0x1C	0x1D	0x41C	0x2206	R*	INT16	Analog Encoder Cosine Input Voltage. Also known as Cosine Feedback Voltage. Units: 0.1 mV.
0x1D	0x1E	0x41D	0x2200	R*	INT16	Analog Reference Input Voltage. Also known as A/D Reference Input Voltage. Units: mV.
0x1E	0x1F	0x41E	0x2201	R*	INT16	High Voltage A/D Reading. The voltage present on the high-voltage bus. Units: 100 mV.

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0x20</td><td>0x21</td><td>0x420</td><td>0x2202</td><td>R*</td><td>INT16</td><td>Dive temperature A/D Reading. Units: degrees C.</td></tr>
<tr><td>0x21</td><td>0x22</td><td>0x421</td><td>0x2110</td><td>RF</td><td>INT16</td><td>Peak Current Limit. Also known as Boost current on stepper drives. This value cannot exceed the peak current rating of the drive. Units: 0.01 A.</td></tr>
<tr><td>0x22</td><td>0x23</td><td>0x422</td><td>0x2111</td><td>RF</td><td>INT16</td><td>Continuous Current Limit. Also known as Run Current on stepper drives. This value should be less than the User Peak Current Limit. Units: 0.01 A.</td></tr>
<tr><td>0x23</td><td>0x24</td><td>0x423</td><td>0x2112</td><td>RF</td><td>U16</td><td>Time at Peak Current Limit. Also known as Time at Boost Current on stepper drives. Units: mS.</td></tr>
<tr><td>0x24</td><td>0x25</td><td>0x424</td><td>0x2300</td><td>RF</td><td>U16</td><td>Desired State:
<table>
<tr><td>Value</td><td>Description</td></tr>
<tr><td>0</td><td>Drive disabled.</td></tr>
<tr><td>1</td><td>The current loop is driven by the programmed current value.</td></tr>
<tr><td>2</td><td>The current loop is driven by the analog reference</td></tr>
<tr><td>3</td><td>The current loop is driven by the PWM input.</td></tr>
<tr><td>4</td><td>The current loop is driven by the function generator.</td></tr>
<tr><td>5</td><td>UV current mode.</td></tr>
<tr><td>6-10</td><td>Reserved for future use.</td></tr>
<tr><td>11</td><td>The velocity loop is driven by the programmed velocity value.</td></tr>
<tr><td>12</td><td>The velocity loop is driven by the analog reference.</td></tr>
<tr><td>13</td><td>The velocity loop is driven by the PWM.</td></tr>
<tr><td>14</td><td>The velocity loop is driven by the function generator.</td></tr>
<tr><td>15-20</td><td>Reserved for future use.</td></tr>
<tr><td>21</td><td>The position loop is driven by the trajectory generator.</td></tr>
<tr><td>22</td><td>The position loop is driven by the analog reference.</td></tr>
<tr><td>23</td><td>The position loop is driven by the digital input lines. (Pulse & direction, master encoder, etc.)</td></tr>
<tr><td>24</td><td>The position loop is driven by the function generator.</td></tr>
<tr><td>25</td><td>The position loop is driven by the cam tables.</td></tr>
<tr><td>26-29</td><td>Reserved.</td></tr>
<tr><td>30</td><td>The drive is controlled by the CANopen or EtherCAT interface</td></tr>
<tr><td>31</td><td>The microstepper is driven by the trajectory generator.</td></tr>
<tr><td>32</td><td>Reserved for future use.</td></tr>
<tr><td>33</td><td>The microstepper is driven by the digital input lines.</td></tr>
<tr><td>34</td><td>The microstepper is driven by the function generator.</td></tr>
<tr><td>35</td><td>The microstepper is driven by the cam tables.</td></tr>
<tr><td>36-39</td><td>Reserved for future use.</td></tr>
</table>
</td></tr>
</table>

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description
						40 The microstepper is controlled by the CANopen or EtherCAT interface.
						41 Reserved
						42 Simple micro-stepping mode. For diagnostic use only.
0x25	0x26	0x425	0x221E	R*	INT16	Limited Motor Current Command. Units: 0.01 A.
0x26	0x27	0x426	0x2313	RF	INT16	Analog Reference Input Deadband. Deadband window value applied to the analog command input. Units: mV.
0x27	0x28	0x427	0x2381:1	RF	U16	Velocity Loop Proportional Gain (Vp).
0x28	0x29	0x428	0x2381:2	RF	U16	Velocity Loop Integral Gain (Vi).
0x29	0x2A	0x429	0x2230	R*	INT32	Velocity Loop Limited Velocity. This is the commanded velocity after it passes through the acceleration and velocity limits and the velocity command filter. The velocity error used by the loop is the difference between the actual velocity and this value. Units: 0.01 RPM for Junus, microsteps/s for Stepnet in stepper mode, 0.1 encoder counts/s for all other products.
0x2A	0x2B	0x42A	None	R*	INT32	Velocity Loop Error.
0x2B	0x2C	0x42B	None	R*	INT32	Velocity loop Integral Sum.
0x2C	0x2D	0x42C	0x606B	R*	INT32	Commanded Velocity. Units: 0.01 RPM for Junus, microsteps/s for Stepnet in stepper mode, 0.1 encoder counts/s for all other products.
0x2D	0x2E	0x42D	0x6062	R*	INT32	Commanded Position. Also known as Limited Position in CME. Units: counts.
0x2E	0x2F	0x42E	0x2381:3	RF	U16	Velocity Loop Acceleration Feed Forward. The acceleration command from the trajectory generator is multiplied by this value and the result is added to the velocity loop output.
0x2F	0x30	0x42F	0x2341	RF	INT32	Programmed Velocity Command. Only used in Programmed Velocity Mode (Desired State (0x24) = 11) Units: 0.01 RPM for Junus, microsteps/s for Stepnet in stepper mode, 0.1 encoder counts/s for all other products.
0x30	0x31	0x430	0x2382:1	RF	U16	Position Loop Proportional Gain (Pp).
0x31	0x32	0x431	0x2381:4	RF	INT16	Velocity Loop Shift Value. After the velocity loop is calculated, the result is right shifted this many times to arrive at the commanded current value. This allows the velocity loop gains to have reasonable values for high resolution encoders.

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description
0x32	0x33	0x432	0x2240	R*	INT32	Actual Motor Position. Also known as Motor Encoder Position. For dual encoder systems, this parameter gives the motor encoder position. For single encoder systems, this is the same as the Actual Position Parameter (0x17). Units: counts.
0x33	0x34	0x433	0x2382:2	RF	U16	Position Loop Velocity Feed Forward (Vff). The Vff value is multiplied by the Instantaneous Commanded Velocity (0x3B).generated by the trajectory generator. The product is added to the output of the position loop. This gain is scaled by 1/16384. Therefore, setting this gain to 0x4000 (16384) would cause the input velocity to be multiplied by 1.0, and the result added to the output of the position loop.
0x34	0x35	0x434	0x2382:3	RF	U16	Position Loop Acceleration Feed Forward (Aff). The Aff value is multiplied by the Instantaneous Commanded Velocity (0x3B) generated by the trajectory generator. The product is added to the output of the position loop.
0x35	0x36	0x435	0x60F4	R*	INT32	Position Loop Error. The difference between Actual Position (0x17) and Commanded Position (0x2D). Units: counts.
0x36	0x37	0x436	0x2100	RF	U32	Velocity Loop Acceleration Limit. Used by the velocity loop limiter. Not used when velocity loop is controlled by the position loop. Units: 1000 counts/s^2.
0x37	0x38	0x437	0x2101	RF	U32	Velocity Loop Deceleration Limit. Used by the velocity loop limiter. Not used when velocity loop is controlled by the position loop. Units: 1000 counts/s^2.
0x38	0x39	0x438	0x221C	R*	INT16	Actual Motor Current. This current is calculated based on both the D and Q axis currents. Units: 0.01 A.
0x39	0x3A	0x439	0x2102	RF	U32	Velocity Loop Emergency Stop Deceleration Rate. Units: 1000 counts/s^2.
0x3A	0x3B	0x43A	0x2103	RF	INT32	Velocity Loop Velocity Limit. This value is a limit on the commanded velocity used by the velocity loop. Note that this limit is always in effect. Units 0.1 counts/s.
0x3B	0x3C	0x43B	0x2250	R*	INT32	Instantaneous Commanded Velocity. This velocity is the output of the trajectory generator and is the value by which the position loop's velocity feed forward is multiplied. Units: 0.1 encoder counts/s.
0x3C	0x3D	0x43C	0x2251	R*	U32	Instantaneous Commanded Acceleration. This acceleration is the output of the trajectory generator and is the value by which the position loop's acceleration feed forward is multiplied. Units: 10 encoder counts/s^2.

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0x3D</td><td>0x3E</td><td>0x43D</td><td>0x2122</td><td>R*</td><td>INT32</td><td>Trajectory Destination Position. This is the position that the trajectory generator is using as its destination. Units: encoder counts.</td></tr>
<tr><td>0x3E</td><td>0x3F</td><td>0x43E</td><td>0x2104</td><td>RF</td><td>INT32</td><td>Velocity Window. If the absolute value of the velocity loop error exceeds this, then the velocity window bit in the event status word will be set. Units: 0.1 counts/s.</td></tr>
<tr><td>0x3F</td><td>0x40</td><td>0x43F</td><td>0x2105</td><td>RF</td><td>U16</td><td>Velocity Window Time. The velocity window bit in the event status will be cleared when the absolute velocity error is less than the velocity window for this amount of time. Units: ms.</td></tr>
<tr><td>0x40</td><td>0x41</td><td>0x440</td><td>0x2383:1</td><td>F</td><td>U16</td><td>Motor Type. The type of motor connected to the drive. Bit-mapped as follows:
<table>
<tr><td>Bits</td><td colspan="2">Description</td></tr>
<tr><td>0</td><td colspan="2">Set for linear motor, clear for rotary.</td></tr>
<tr><td>1-3</td><td colspan="2">Reserved.</td></tr>
<tr><td rowspan="5">4-5</td><td colspan="2">Motor architecture:</td></tr>
<tr><td>0</td><td>Not specified.</td></tr>
<tr><td>1</td><td>Brushed servo motor.</td></tr>
<tr><td>2</td><td>Microstepper.</td></tr>
<tr><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>0x42</td><td>0x441</td><td>0x6404</td><td>F</td><td>String</td><td>Motor Manufacturer.</td></tr>
<tr><td>0x42</td><td>0x43</td><td>0x442</td><td>0x6403</td><td>F</td><td>String</td><td>Motor Model.</td></tr>
<tr><td>0x43</td><td>0x44</td><td>0x443</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>0x45</td><td>0x444</td><td>0x2383:9</td><td>F</td><td>INT32</td><td>Motor Inertia (Mass). Units: Rotary = 0.000001 Kg/cm^2; Linear = 0.0001 Kg.</td></tr>
<tr><td>0x45</td><td>0x46</td><td>0x445</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>0x47</td><td>0x446</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>0x48</td><td>0x447</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>0x49</td><td>0x448</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>0x4A</td><td>0x449</td><td>0x2383:7</td><td>F</td><td>INT16</td><td>Motor Resistance. Units: 10 mΩ.</td></tr>
<tr><td>0x4A</td><td>0x4B</td><td>0x44A</td><td>0x2383:8</td><td>F</td><td>INT16</td><td>Motor Inductance. Units: 10 µH.</td></tr>
<tr><td>0x4B</td><td>0x4C</td><td>0x44B</td><td>0x2383:13</td><td>F</td><td>INT32</td><td>Motor Peak Torque. Units: 0.00001 Nm units.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th colspan="3">Description</th></tr>
<tr><td>0x4C</td><td>0x4D</td><td>0x44C</td><td>0x2383:14</td><td>F</td><td>INT32</td><td colspan="3">Motor Continuous Torque. Units: 0.00001 Nm units.</td></tr>
<tr><td>0x4D</td><td>0x4E</td><td>0x44D</td><td>0x2383:11</td><td>F</td><td>INT32</td><td colspan="3">Motor Max Velocity. Units: 0.1 encoder counts/s.</td></tr>
<tr><td>0x4E</td><td>0x4F</td><td>0x44E</td><td>0x2383:3</td><td>F</td><td>U16</td><td colspan="3">Motor Wiring. 0=standard, 1= drive’s U and V outputs are swapped.</td></tr>
<tr><td>0x4F</td><td>0x50</td><td>0x44F</td><td>0x2383:6</td><td>RF</td><td>INT16</td><td colspan="3">Motor Hall Offset. Offset angle to be applied to the Hall sensors. Units: degrees.</td></tr>
<tr><td rowspan="5">0x50</td><td rowspan="5">0x51</td><td rowspan="5">0x450</td><td rowspan="5">0x2383:4</td><td rowspan="5">F</td><td rowspan="5">INT16</td><td colspan="3">Motor Hall Type.
The type of Hall effect sensors attached to the motor:</td></tr>
<tr><td>Value</td><td colspan="2">Description</td></tr>
<tr><td>0</td><td colspan="2">No Hall sensors available.</td></tr>
<tr><td>1</td><td colspan="2">Digital Hall sensors.</td></tr>
<tr><td>2</td><td colspan="2">Analog Hall sensors.</td></tr>
<tr><td rowspan="18">0x52</td><td rowspan="18">0x53</td><td rowspan="18">0x452</td><td rowspan="18">0x2383:5</td><td rowspan="18">F</td><td rowspan="18">INT16</td><td colspan="3">Motor Hall Wiring. Bit-mapped as follows:

NOTE: When analog Halls are used, only bit 8 is relevant.</td></tr>
<tr><td>Bits</td><td colspan="2">Description</td></tr>
<tr><td rowspan="9">0-2</td><td colspan="2">The Hall wiring code (see below).</td></tr>
<tr><td>Value</td><td>Hall Ordering</td></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>
<tr><td>3</td><td colspan="2">Reserved.</td></tr>
<tr><td>4</td><td colspan="2">Invert W Hall input if set. Inversion occurs after Halls wiring is changed by bits 0-2.</td></tr>
<tr><td>5</td><td colspan="2">Invert V Hall input if set. Inversion occurs after Halls wiring is changed by bits 0-2.</td></tr>
<tr><td>6</td><td colspan="2">Invert U Hall input if set. Inversion occurs after Halls wiring is changed by bits 0-2.</td></tr>
<tr><td>7</td><td colspan="2">Reserved.</td></tr>
<tr><td>8</td><td colspan="2">Reverse analog Halls if set.</td></tr>
<tr><td>9-15</td><td colspan="2">Reserved.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0x53</td><td>0x54</td><td>0x453</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>0x55</td><td>0x454</td><td>0x2383:18</td><td>F</td><td>U16</td><td>Motor Brake Delay Time. After the brake output is activated, the drive will stay enabled for this amount of time to allow the brake to engage. Units: ms.</td></tr>
<tr><td>0x55</td><td>0x56</td><td>0x455</td><td>0x2383:19</td><td>F</td><td>INT32</td><td>Motor Brake Activation Velocity. Also known as Motor Brake Velocity (CANopen). During the Motor Brake Activation Time (0x53), if the motor's actual velocity falls below this value the brake output is activated immediately.
Units: 0.1 counts/s.</td></tr>
<tr><td>0x56</td><td>0x57</td><td>0x456</td><td>0x2383:10</td><td>F</td><td>U32</td><td>Motor Back EMF Constant. Back EMF velocity estimation can be disabled by setting to zero. Units: rotary motor: 0.01 V/Krpm; linear motor: 0.01 V/mps.</td></tr>
<tr><td>0x57</td><td>0x58</td><td>0x457</td><td>0x2383:29</td><td>F</td><td>U32</td><td>Microsteps/Motor Rev. This parameter is used in true microstepping mode.
Units: microsteps.</td></tr>
<tr><td>0x58</td><td>0x59</td><td>0x458</td><td>0x2383:33</td><td>F</td><td>INT32</td><td>Motor Gear Ratio. This parameter may be used to store gear ratio information for dual encoder systems where a gearbox sits between the two encoders. This parameter is not used by the firmware and is supported as a convenience to the CME program.

Gear ratio is a ratio of two 16-bit values. The first word gives the number of motor turns and is the numerator. The second word gives the number of position turns and is the denominator.</td></tr>
<tr><td>0x59</td><td>0x5A</td><td>0x459</td><td>0x2107</td><td>RF</td><td>INT16</td><td>Hall Velocity Mode Shift Value. This parameter is only used in Hall velocity mode. It specifies a left shift value for the position and velocity information calculated in that mode.</td></tr>
<tr><td>0x5A</td><td>0x5B</td><td>0x45A</td><td>0x2241</td><td>RF</td><td>INT16</td><td>Encoder output configuration. This parameter determines the source of the buffered encoder output on drives which support it. Bit-mapped as follows:
<table>
<tr><th>Bits</th><th colspan="2">Description</th></tr>
<tr><td>0-1</td><td colspan="2">Mode of operation for encoder output lines.</td></tr>
<tr><td></td><td>0</td><td>Output buffered primary encoder (hardware buffering).</td></tr>
<tr><td></td><td>1</td><td>Configure pins as secondary encoder input.</td></tr>
<tr><td></td><td>2</td><td>Output simulated encoder outputs tracking motor encoder.</td></tr>
<tr><td></td><td>3</td><td>Output simulated encoder outputs tracking load encoder.</td></tr>
<tr><td>8-11</td><td colspan="2">For simulated encoder outputs, these bits configure a scaling value that adjusts the number of encoder output counts for each encoder count on the input.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td><table>
<tr><td>0</td><td>No adjustment, 1 count on the encoder is 1 output count.</td></tr>
<tr><td>1</td><td>Double the number of encoder counts.</td></tr>
<tr><td>2</td><td>Divide the encoder counts by 2.</td></tr>
<tr><td>3</td><td>Divide the encoder counts by 4.</td></tr>
<tr><td>4</td><td>Divide the encoder counts by 8.</td></tr>
<tr><td>5</td><td>Divide the encoder counts by 16.</td></tr>
<tr><td>6</td><td>Divide the encoder counts by 32.</td></tr>
<tr><td>7</td><td>Divide the encoder counts by 64.</td></tr>
<tr><td>8</td><td>Divide the encoder counts by 128.</td></tr>
<tr><td>9</td><td>Divide the encoder counts by 256.</td></tr>
</table></td></tr>
<tr><td>0x5B</td><td>0x5C</td><td>0x45B</td><td>0x2383:32</td><td>F</td><td>INT32</td><td>Load Encoder Resolution. Number of Motor Encoder Units (0x61) per encoder count. Linear motors only. Units: encoder units/count.</td></tr>
<tr><td>0x5C</td><td>0x5D</td><td>0x45C</td><td>0x2383:31</td><td>F</td><td>INT16</td><td>Load Encoder Direction. 0=normal, 1=reverse.</td></tr>
<tr><td>0x5D</td><td>0x5E</td><td>0x45D</td><td>0x2383:30</td><td>F</td><td>U16</td><td>Load Encoder Type. This parameter identifies the type of encoder used on the load when running in dual loop mode. The encoding of this parameter has changed over time to support more encoder types than were originally envisioned when the parameter was first defined. Bit 12 of the parameter is used to identify which encoding is active.

New encoding supported by Feature Set C (starting with version 2.10), Feature Set D, and Feature Set E:<table>
<tr><td>Bits</td><td>Meaning</td></tr>
<tr><td>0-11</td><td>Encoder hardware to use:<table>
<tr><td>0-15</td><td>Same encoder types as listed below in original encoding.</td></tr>
<tr><td>16</td><td>Simple analog potentiometer for feedback.</td></tr>
<tr><td>17</td><td>Reserved for custom encoder.</td></tr>
<tr><td>18</td><td>Reserved for custom encoder.</td></tr>
<tr><td>19</td><td>Reserved for custom encoder.</td></tr>
</table></td></tr>
<tr><td>12</td><td>Always set to identify new encoding.</td></tr>
<tr><td>13</td><td>Linear encoder if set, rotary encoder if clear.</td></tr>
</table></td></tr>
</table>

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description		
						14	If set, then don't use this encoder for position feedback.	
						15	Reserved.	
						Original encoding (bit 12 not set):		
						0-3	Encoder hardware to use:	
							0	No load encoder present.
							1	Primary (differential) quad encoder.
							2	Analog encoder.
							3	Secondary quad encoder from input lines.
							4	Servo tube / analog halls.
							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	Linear encoder if set, rotary encoder if clear.	
						5	If set, then don't use this encoder for position feedback.	
						6-15	Must be zero.	
0x5E	0x5F	0x45E	0x2231	R*	INT32	Load Encoder Velocity. Units: 0.1 encoder counts/s		
0x5F	0x60	0x45F	0x2106	RF	9 or 14	Velocity Loop Output Filter. A bi-quad filter which acts on the output of the velocity loop. 9 word parameter, see *Velocity Loop Filters* in the *CME 2 User Guide.* 14 word parameter (Plus product only), see Filter Coefficients (p. 73).		
0x60	0x61	0x460	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.	
						3	Secondary quad encoder from input lines.	
						4	Low frequency analog encoder.	
						5	Resolver.	
						6	Use digital hall signals for position & velocity estimates.	

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description	
						7	Analog encoder updated at current loop rate.
						8	Reserved for custom 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	Reserved for custom encoder.
						18	Reserved for custom encoder.
						19	Reserved for custom encoder.
0x61	0x62	0x461	0x2383:21	F	INT16	Motor Encoder Units. This value defines the units used to describe linear motor encoders. It is not used with rotary motors.	
						Value	Description
						0	Microns.
						1	Nanometers.
						2	Millimeters.
0x62	0x63	0x462	0x2383:23	F	INT32	Motor Encoder Counts/Rev. Rotary motor only. When a resolver is used as the motor feedback, this parameter sets the resolution of the interpolated position. Units: counts/rev.	
0x63	0x64	0x463	0x2383:24	F	INT16	Motor Encoder Resolution. Linear motor only. Units: encoder units/count.	
0x64	0x65	0x464	0x2383:25	F	INT32	Motor Encoder Electrical Distance. Linear motor only. Units: encoder units/electrical cycle.	
0x65	0x66	0x465	0x2383:22	F	U16	Motor Encoder Direction. 0=normal, 1=reverse.	
0x67	0x68	0x467	0x2383:28	F	INT16	Analog Encoder Shift Amount. This value gives the number of bits of interpolation to be applied to an analog encoder. The fundamental encoder resolution will be increased by a multiplier of 2^n where n is the value programmed in this parameter. The range of this value is 0 to 10, giving possible multipliers of 1 to 1024.	

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0x68</td><td>0x69</td><td>0x468</td><td>0x2402</td><td>R*</td><td>INT32</td><td>Captured Index Position. Provides the position that the axis was in when an index pulse was captured. Configured by setting bits in the Position Capture Control Register (0x6C), and the status of the captured data can be checked in the Position Capture Status Register (0x6D). Reading this variable resets bits 0 & 3 of the Position Capture Status Register (0x6D). Units: counts.</td></tr>
<tr><td>0x69</td><td>0x6A</td><td>0x469</td><td>0x2232</td><td>R*</td><td>INT32</td><td>Unfiltered Motor Encoder Velocity. Units 0.1 counts/s.</td></tr>
<tr><td>0x6A</td><td>0x6B</td><td>0x46A</td><td>0x2113</td><td>RF</td><td>INT32</td><td>Commanded Current Ramp Limit. Setting this to zero disables slope limiting. Units: mA/s.</td></tr>
<tr><td>0x6B</td><td>0x6C</td><td>0x46B</td><td>0x2108</td><td>RF</td><td>9 or 14</td><td>Velocity Loop Command Filter Coefficients. A bi-quad filter structure that acts on the command input of the velocity loop just after velocity & acceleration limiting. 9 word parameter, see *Velocity Loop Filters* in the *CME 2 User Guide.* 14 word parameter (Plus product only), see Filter Coefficients (p. 73).</td></tr>
<tr><td>0x6C</td><td>0x6D</td><td>0x46C</td><td>0x2400</td><td>RF</td><td>INT16</td><td>Position Capture Control Register. Sets up position capture based on the index or home input. Bit-mapped as follows:
<table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0</td><td>If set, the Captured Index Position (0x68) is captured on the falling edge of the index.</td></tr>
<tr><td>1</td><td>If set, the Captured Index Position (0x68) is captured on the rising edge of the index.</td></tr>
<tr><td>2</td><td>If set, a Captured Index Position (0x68) value will not be overwritten by a 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 the active to inactive edge of the home input switch. If clear, the home position will be captured on the inactive to active edge.</td></tr>
<tr><td>6</td><td>If set, Captured Home Position (0x10A) will not be overwritten by a 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..</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>12</td><td>Clear actual position on every encoder index pulse.</td></tr>
</table></td></tr>
<tr><td>0x6D</td><td>0x6E</td><td>0x46D</td><td>0x2401</td><td>R*</td><td>INT16</td><td>Position Capture Status Register. This register shows the current status of the index/home capture mechanism. Bit-mapped as follows:</td></tr>
</table>

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description	
						Bits	Description
						0	If set, an index position has been captured. Cleared when the captured position is read.
						1-2	Reserved.
						3	If set, a new index transition occurred when a captured position was already stored. Depending on the mode, the new position may have been discarded, or overwritten the previously stored position.
						4	If set, a home position has been captured. This bit is cleared when the captured position is read.
						5-6	Reserved.
						7	If set, a new home pin transition occurred when a captured position was already stored. Depending on the mode, the new position may have been discarded, or overwritten the previously stored position.
						8	New high speed capture data if set.
						9-10	Reserved.
						11	High speed capture overflow if set.
0x6E	0x6F	0x46E	0x2383:34	F	INT16	Number of Resolver Cycles/Motor Rev. This parameter is only used with resolver feedback devices.	
0x6F	0x70	0x46F	0x2140	RF	INT16	PWM Mode and Status. Bit-mapped as follows:	
						Bits	Description
						0	Force bus clamping if set, disable bus clamping if clear. If bit 1 is set, then this bit is ignored.
						1	Automatic bus clamping mode if set. Setting this bit causes bus clamping mode to be automatically selected based on the output voltage. Bit 0 is ignored if this bit is set
						2	Factory reserved. If set, DBrk mode is enabled.
						4	Use hexagonal voltage limiting if set, circular limiting if clear. This setting is only used with brushless motors.
						6	1. Double PWM frequency if set.
							3. Status bit, set when bus clamping is active.
0x70	0x71	0x470	0x2193:1	RF	See text	Output 0 Configuration. For notes on Output numbering see Input/Output Numbering. Data type is dependent on configuration and uses 1 to 5 words.	

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>
<p>The first word is a bit-mapped configuration value. The remaining words give additional parameter data used by the output pin. Typically the second and third words are used as a 32-bit mask to identify which bit(s) in the status register the output should follow. If any of the selected bits in the status register are set, then the output will go active. If none of the selected bits are set, then the output will be inactive.</p>
<p>Outputs may be programmed as a sync output for use in synchronizing multiple drives. In this configuration, the first word of this variable should be set to 0x0200 (i.e., only bit 9 is set), and the remaining words should be set to zero. Note that only output pin #0 has this feature. Attempting to program any other output pin as a sync output will have no effect.</p>
<p>The first word is bit-mapped as follows:</p>
<table>
<tr><td>Bits</td><td colspan="2">Configuration</td></tr>
<tr><td rowspan="13">0-4</td><td colspan="2">Define which internal register drives the output. The acceptable values for these bits are as follows:</td></tr>
<tr><td>Value</td><td>Description</td></tr>
<tr><td>0</td><td>Track bits in the event status.</td></tr>
<tr><td>1</td><td>Track bits in the latched event status.</td></tr>
<tr><td>2</td><td>Track bits in the manual output control register (see Output States and Program Control (0xAB)).</td></tr>
<tr><td>3</td><td>Track bits in the trajectory status register (see 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 12 is set, commanded position is used. If bit 12 is clear, actual position is used.</td></tr>
<tr><td>5</td><td>Go active on a low to high crossing of the position specified by words 2, 3. Stay high for the number of milliseconds specified by words 4, 5. If bit 12 is set, commanded position is used. If bit 12 is clear, actual position is used.</td></tr>
<tr><td>6</td><td>Same as 5, but for high to low crossings.</td></tr>
<tr><td>7</td><td>Same as 5 but for any crossing.</td></tr>
<tr><td>8</td><td>Go active if motor phase angle (plus an offset) is between 0 and 180 degrees. The offset is set using the first word of extra data in units of 32k/180 degrees.</td></tr>
<tr><td>9</td><td>Pulse output each time a position is crossed from an array of positions stored in trace memory.</td></tr>
<tr><td>10</td><td>Use the output to trigger an external regen resister.</td></tr>
</table>
</td></tr>
</table>

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**		
							11	For EtherCAT drives, pulse on the SYNC0 signal.
							16	Track hardware position compare function on drives supporting it.
						5-7	Reserved for future use.	
						8	Inverts normal active state of output if set. E.g., outputs that are normally active low become active high.	
						9	If set, program the output as a sync output. This bit is reserved for all output pins except pin 0.	
						10-11	Reserved for future use.	
						12-13	Axis number for multi-axis drives.	
						14-15	Usage depends on output function selected.	
0x71	0x72	0x471	0x2193:2	RF	See text	Output 1 Configuration. See Output 0 Configuration (0x70).		
0x72	0x73	0x472	0x 2193:3	RF	See text	Output 2 Configuration. See Output 0 Configuration (0x70).		
0x73	0x74	0x473	0x 2193:4	RF	See text	Output 3 Configuration. See Output 0 Configuration (0x70).		
0x74	0x75	0x474	0x 2193:5	RF	See text	Output 4 Configuration. See Output 0 Configuration (0x70).		
0x75	0x76	0x475	0x 2193:6	RF	See text	Output 5 Configuration. See Output 0 Configuration (0x70).		
0x76	0x77	0x476	0x 2193:7	RF	See text	Output 6 Configuration. See Output 0 Configuration (0x70).		
0x77	0x78	0x477	0x 2193:8	RF	See text	Output 7 Configuration. See Output 0 Configuration (0x70).		
0x78	0x79	0x478	0x 2192:1	RF	U16	Input 0 Configuration. Assigns a function to the input pin. All values not listed below are reserved for future use. For notes on Input numbering see Input/Output Numbering. The sync input function is only valid for high speed input pins. In addition, input pins 2 & 3 of the Accelus & Junus drives do not support this feature.		
						Bits	Configuration	
						0-7	Value	Meaning
							0	No function
							1	Reserved for future use (no function).
							2	Reset the drive on the rising edge of the input.
							3	Reset the drive on the falling edge of the input.
							4*	Positive side limit switch. Active high.
							5*	Positive side limit switch. Active low.

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description		
							6*	Negative side limit switch. Active high.
							7*	Negative side limit switch. Active low.
							8*	Motor temperature switch. Active high.
							9*	Motor temperature switch. Active low.
							10*	Clear faults on rising edge, disable while high.
							11*	Clear faults on falling edge, disable 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.
							19	Sync input on falling edge.
							20*	Halt motor, active high.
							21*	Halt motor, active low.
							22	Scale analog input when high.
							23	Scale analog input when 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.
							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.

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**		
							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.
							*These functions use bit 8 to indicate that the input function should apply to all axes. This feature is only available in Plus products with a firmware version of at least 1.72	
						8-11	Used to pass parameters to the input pin functions.	
						12-13	Bits 12-13 are used to select the axis on multi-axis drives.	
0x79	0x7A	0x479	0x2192:2	RF	U16	Input 1 Configuration. See Input 0 Configuration (0x78).		
0x7A	0x7B	0x47A	0x2192:3	RF	U16	Input 2 Configuration. See Input 0 Configuration (0x78).		
0x7B	0x7C	0x47B	0x2192:4	RF	U16	Input 3 Configuration. See Input 0 Configuration (0x78).		
0x7C	0x7D	0x47C	0x2192:5	RF	U16	Input 4 Configuration. See Input 0 Configuration (0x78).		
0x7D	0x7E	0x47D	0x2192:6	RF	U16	Input 5 Configuration. See Input 0 Configuration (0x78).		
0x7E	0x7F	0x47E	0x2192:7	RF	U16	Input 6 Configuration. See Input 0 Configuration (0x78).		
0x7F	0x80	0x47F	0x2192:8	RF	U16	Input 7 Configuration. See Input 0 Configuration (0x78).		
0x80	0x81	0x480	0x6503	F*	String	Model Number.		
0x81	0x82	0x481	0x2384:1 or 0x1018:4	F*	U32	Drive Serial Number.		
0x82	0x83	0x482	0x2384:3	F*	INT16	Drive’s rated Peak Current. Units: 0.01 A.		
0x83	0x84	0x483	0x2384:4	F*	INT16	Drive’s rated Continuous Current. Units: 0.01 A.		
0x84	0x85	0x484	0x2384:14	F*	INT16	Current Corresponding to Max A/D Reading. Units: 0.01 A.		
0x85	0x86	0x485	0x2384:11	F*	U16	PWM Period. Units: 10 ns.		
0x86	0x87	0x486	0x2384:12	F*	U16	Drive Servo Period (PWM periods). Servo loop update period as a multiple of the current loop period.		

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description
0x87	0x88	0x487	None	F*	U16	Product Family. Identifies the drive product family. For specific drive hardware type, see Drive Hardware Type (0xAD).
0x88	0x89	0x488	0x2384: 5	F*	INT16	Drive's rated Time At Peak Current. The maximum time for which the drive is rated to output peak current. Units: ms.
0x89	0x8A	0x489	0x2384:6	F*	INT16	Drive's rated Maximum Voltage. Maximum bus voltage rating. Units: 0.1 V.
0x8A	0x8B	0x48A	0x2384:15	F*	INT16	Drive's rated Voltage Corresponding To Max A/D Reading. Units: 0.1 V.
0x8B	0x8C	0x48B	0x2384:7	F*	INT16	Drive's rated Minimum Voltage. Minimum bus voltage rating. Units: 0.1 V.
0x8C	0x8D	0x48C	0x2384:9	F*	INT16	Drive's rated Maximum Temperature. Units: degrees C.
0x8D	0x8E	0x48D	0x2384:2	F*	String	Manufacturing info (date code, etc.).
0x8E	0x8F	0x48E	0x2384:16	F*	INT16	Analog Input Scaling Factor. This is the voltage applied to the analog input which causes the max A/D value on the drive.
0x90	0x91	0x490	None	R	U32	Serial Port Baud Rate. Units: bits/s. Defaults to 9600 at reset.
0x91	0x92	0x491	None	R*	INT16	The maximum number of data words allowed per binary command over the serial interface.
0x92	0x93	0x492	0x21A0	F	String	Drive Name. This object can assign an optional name to a drive. The data written here is stored to flash memory and is not used by the drive. Although this object is documented as holding a string (i.e. ASCII data), any values may be written here. Up to 40 bytes are stored.
0x94	0x95	0x494	0x2384:24	R*	INT16	Firmware Version Number. The version number consists of a major and a minor version number. The minor number is passed in bits 0-7; the major number is in bits 8-15. For example, the version 1.12 would be encoded 0x010C.
0x95	0x96	0x495	0x2421	F	20 words	Host Configuration State. Reserved for use by CME 2 software.
0x96	0x97	0x496	0x2312	RF	INT16	Calibration Offset For Analog Reference. This voltage is added to the analog command input. It is factory-calibrated to give a zero reading for zero input voltage.
0x97	0x98	0x497	0x2384:10	F*	INT16	Hysteresis Value For Drive Over Temperature Cut-Out. Units: degrees C.
0x98	0x99	0x498	0x2330	RF	INT16	Function Generator Configuration. Configures the drive's internal function generator, which can drive the current, velocity, or position loop. Bit-mapped as follows:

Bits	Description	
0-1	Function code (type of waveform to generate):	
	Value	Description

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>
<table>
<tr><td></td><td><table><tr><td>0</td><td>None (disabled).</td></tr><tr><td>1</td><td>Square wave.</td></tr><tr><td>2</td><td>Sine wave.</td></tr></table></td></tr>
<tr><td>2-7</td><td>Reserved for future use.</td></tr>
<tr><td>8</td><td>If set, the function generator frequency is in units of 0.01 Hz. If clear, the frequency is in units of Hz.</td></tr>
<tr><td>9-11</td><td>Reserved.</td></tr>
<tr><td>12</td><td>One-shot mode if set. If bit 12 is set and bit 13 is clear, the function code is reset to zero (disabled) after one complete waveform. If bits 12 and 13 are both set, the function code is reset to zero after two waveforms.</td></tr>
<tr><td>13</td><td>Invert every other period if set.</td></tr>
<tr><td>14-15</td><td>Reserved for future use.</td></tr>
</table>
The function code programmed into bits 0-1 defines the type of waveform to be generated:
<table>
<tr><td>Code</td><td>Description</td></tr>
<tr><td>0</td><td>None (disabled).</td></tr>
<tr><td>1</td><td>Square wave.</td></tr>
<tr><td>2</td><td>Sine wave.</td></tr>
</table>
Note that the drive is placed under control of the function generator by setting Desired State (0x24) to one of the following values:
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).

Note that if one-shot mode is selected, then after one period (two if invert is selected) the function type will reset to zero.
</td></tr>
<tr><td>0x99</td><td>0x9A</td><td>0x499</td><td>0x2331</td><td>RF</td><td>U16</td><td>Function Generator Frequency. Units: Hz. See bit 8 of Function Generator Configuration (0x98).</td></tr>
<tr><td>0x9A</td><td>0x9B</td><td>0x49A</td><td>0x2332</td><td>RF</td><td>INT32</td><td>Function Generator Amplitude. The amplitude of the signal generated by the internal function generator. The units depend on the mode:
<table>
<tr><td>Mode</td><td>Units</td></tr>
<tr><td>Current</td><td>0.01 A.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</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 rowspan="2"></td><td rowspan="2"></td><td>Velocity</td><td>0.1 counts/s.</td></tr>
<tr><td>Position</td><td>Counts.</td></tr>
<tr><td>0x9B</td><td>0x9C</td><td>0x49B</td><td>0x2333</td><td>RF</td><td>U16</td><td colspan="2">Function Generator Duty Cycle (square wave only).
Units: 0.1% (for instance, 1000 for 100%).</td></tr>
<tr><td>0x9C</td><td>0x9D</td><td>0x49C</td><td>0x2384:8</td><td>F*</td><td>U16</td><td colspan="2">Hysteresis For Maximum Bus Voltage Cut-Out. Units: 0.1 V.</td></tr>
<tr><td>0x9D</td><td>0x9E</td><td>0x49D</td><td>0x2384:18</td><td>F*</td><td>U16</td><td colspan="2">PWM Dead Time At Continuous Current Limit. This parameter gives the PWM dead time used at or above the continuous current limit. The dead time below the continuous current limit is a linear function of this parameter and PWM Dead Time At Zero Current (0x9F). Units: CPU cycles.</td></tr>
<tr><td>0x9E</td><td>0x9F</td><td>0x49E</td><td>0x2384:17</td><td>F*</td><td>U16</td><td colspan="2">Drive Minimum PWM Off Time. This parameter gives the minimum amount of time for which all PWM outputs must be disabled for each current loop cycle. Units: 10 ns.</td></tr>
<tr><td>0x9F</td><td>0xA0</td><td>0x49F</td><td>0x2384:19</td><td>F*</td><td>U16</td><td colspan="2">PWM Dead Time At Zero Current. This parameter gives the PWM dead time at zero current. The dead time above zero current is defined by a linear function of this parameter and PWM Dead Time At Continuous Current Limit (0x9D). Units: CPU cycles.</td></tr>
<tr><td rowspan="16">0xA0</td><td rowspan="16">0xA1</td><td rowspan="16">0x4A0</td><td rowspan="16">0x1002</td><td rowspan="16">R*</td><td rowspan="16">U32</td><td colspan="2">Drive Event Status Register. Bit-mapped as follows:</td></tr>
<tr><td>Bits</td><td>Description</td></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>Encoder feedback 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.</td></tr>
<tr><td>13</td><td>Trying to stop motor.</td></tr>
</table>

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description	
						14	Motor brake activated.
						15	PWM outputs disabled.
						16	Positive software limit condition.
						17	Negative software limit condition.
						18	Tracking error.
						19	Tracking warning.
						20	Drive is currently in a reset condition.
						21	Position has wrapped. The 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. A drive fault that was configured as latching has occurred. For information on latching faults, see the CME 2 User Guide.
						23	Velocity limit has been reached.
						24	Acceleration limit has been reached.
						25	Tracking Window. Position Loop Error (0x35) is outside of Position Tracking Error Limit (0xBA).
						26	Home switch is active.
						27	In motion. Set if the trajectory generator is running a profile, or the Position Tracking Error Limit (0xBA) is outside the tracking window. Clear when settled into position.
						28	Velocity window. Set if the absolute velocity error exceeds the velocity window value.
						29	Phase not yet initialized. If the drive is phasing with no Halls, this bit is set until the drive has initialized its phase.
						30	Command fault. PWM or other command signal (e.g., EtherCAT master) not present. If **Allow 100% Output** option is enabled by a setting Bit 3 of Digital Input Command Configuration (0xA8) this fault will not detect a missing PWM command.
						31	Not defined.
0xA1	0xA2	0x4A1	0x2181	R	U32	Latched Event Status Register. This is a latched version of the Drive Event Status Register (0xA0). Bits are set by the drive when events occur. Bits are only cleared by writing to this parameter as explained below: When writing to the Latched Event Status Register, any bit set will cause the corresponding bit in the register to be cleared. For example, to clear the Over Voltage bit, write a 1 to the register. To clear all bits, write 0x4 to the register.	

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0xA2</td><td>0xA3</td><td>0x4A2</td><td>0x2261</td><td>R*</td><td>INT16</td><td>Hall Input State. The lower three bits of the returned value give the present state of the Hall input pins. The Hall state is the value of the Hall lines AFTER the ordering and inversions specified in the Hall wiring configuration have been applied.</td></tr>
<tr><td>0xA4</td><td>0xA5</td><td>0x4A4</td><td>0x2183</td><td>R</td><td>U32</td><td>Latching Fault Status Register. Bit-mapped to show which latching faults have occurred in the drive. When a latching fault has occurred, the fault bit (bit 22) of the Drive Event Status Register (0xA0) is set. The cause of the fault can be read from this register.
To clear a fault condition, write a 1 to the associated bit in this register. The events that cause the drive to latch a fault are programmable. See Fault Mask (0xA7) for details.
<table>
<tr><td colspan="2">Latched Faults</td></tr>
<tr><td>Bits</td><td>Fault Description</td></tr>
<tr><td>0</td><td>Data flash CRC failure. This fault is considered fatal and cannot be cleared.</td></tr>
<tr><td>1</td><td>Drive internal error. This fault is considered fatal and cannot be cleared.</td></tr>
<tr><td>2</td><td>Short circuit.</td></tr>
<tr><td>3</td><td>Drive over temperature.</td></tr>
<tr><td>4</td><td>Motor over temperature.</td></tr>
<tr><td>5</td><td>Over voltage.</td></tr>
<tr><td>6</td><td>Under voltage.</td></tr>
<tr><td>7</td><td>Feedback fault.</td></tr>
<tr><td>8</td><td>Phasing error.</td></tr>
<tr><td>9</td><td>Tracking error.</td></tr>
<tr><td>10</td><td>Current limited by I^2T algorithm.</td></tr>
<tr><td>11</td><td>FPGA error type 1.</td></tr>
<tr><td>12</td><td>Command input lost.</td></tr>
<tr><td>13</td><td>FPGA error type 2.</td></tr>
<tr><td>14</td><td>Safety circuit fault.</td></tr>
<tr><td>15</td><td>Unable to control current.</td></tr>
<tr><td>16</td><td>Motor wiring disconnected (see Open Motor Wiring Check Current (0x19D)).</td></tr>
<tr><td>17-31</td><td>Reserved.</td></tr>
</table>
</td></tr>
<tr><td>0xA5</td><td>0xA6</td><td>0x4A5</td><td>0x2191</td><td>RF</td><td>U16</td><td>Input Pin Configuration Register. Some drives have one or more pull-up resistors associated with their general-purpose input pins. On these drives, the state of the pull-ups can be controlled by writing to this register.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="2">This register has one bit for each pull-up resistor available on the drive. Setting the bit causes the resistor to pull any inputs connected to it up to the high state when they are not connected. Bits 0 – 7 of this register are used to control pull-up resistor states. Each bit represents an input number. Bit 0 = IN1, bit 1 = IN2, etc.
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.</td></tr>
<tr><td rowspan="18">0xA6</td><td rowspan="18">0xA7</td><td rowspan="18">0x4A6</td><td rowspan="18">0x2190</td><td rowspan="18">R*</td><td rowspan="18">U16</td><td colspan="2">Input Pin States. The 16-bit value returned by this command gives the current state (high/low) of the drive’s input pins after debounce. Each bit represents one input as shown below.</td></tr>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0</td><td>Input 0.</td></tr>
<tr><td>1</td><td>Input 1.</td></tr>
<tr><td>2</td><td>Input 2.</td></tr>
<tr><td>3</td><td>Input 3.</td></tr>
<tr><td>4</td><td>Input 4.</td></tr>
<tr><td>5</td><td>Input 5.</td></tr>
<tr><td>6</td><td>Input 6.</td></tr>
<tr><td>7</td><td>Input 7.</td></tr>
<tr><td>8</td><td>Input 8.</td></tr>
<tr><td>9</td><td>Input 9.</td></tr>
<tr><td>10</td><td>Input 10.</td></tr>
<tr><td>11</td><td>Input 11.</td></tr>
<tr><td>12</td><td>Input 12.</td></tr>
<tr><td>13</td><td>Input 13.</td></tr>
<tr><td>14</td><td>Input 14.</td></tr>
<tr><td>15</td><td>Input 15.</td></tr>
</table>

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**
0xA7	0xA8	0x4A7	0x2182	RF	U32	Fault Mask. This variable is used to configure which drive events cause latching faults. Setting a fault mask bit to 1 causes the associated drive event to cause a latching fault when it occurs. Setting a fault mask bit to 0 disables fault latching on the associated event. Latched faults may be cleared using the Latching Fault Status Register (0xA4).

Bits	Fault Description
0	Data flash CRC failure. This bit is read-only and will always be set. If the drive detects corrupted flash data values on startup it will remain disabled and indicate a fault condition.
1	Drive internal error. This bit is read-only and will always be set. If the drive fails its power-on self-test, it will remain disabled and indicate a fault condition.
2	Short circuit. If set: programs the drive to latch a fault condition when a short circuit is detected on the motor outputs. If clear: programs the drive to disable its outputs for 100 ms after a short circuit and then re-enable.
3	Drive over temperature. If set: programs the drive to latch a fault condition when a drive over temperature event happens. If clear: programs the drive to re-enable as soon as it cools sufficiently from an over temperature event.
4	Motor over temperature. If set: programs the drive to latch a fault condition when a motor temperature sensor input activates. If clear: programs the drive to re-enable as soon as the over temperature input becomes inactive.
5	Over voltage. If set: programs the drive to latch a fault condition when excessive bus voltage is detected. If clear: programs the drive to re-enable as soon as the bus voltage is within normal range.
6	Under voltage. If set: programs the drive to latch a fault condition when inadequate bus voltage is detected. If clear: programs the drive to re-enable as soon as the bus voltage is within normal range.
7	Feedback fault. If set: programs the drive to latch a fault when feedback faults occur. Feedback faults occur if too much current is drawn from the 5 V source on the drive, a resolver or analog encoder is disconnected, or a resolver or analog encoder has levels out of tolerance.
8	Phasing error. If set: programs the drive to latch a fault when phasing errors occur. If clear: programs the drive to re-enable when the phasing error is removed.
9	Tracking error. If set: programs the drive to latch in the disabled state when a tracking error

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description	
							occurs. If clear: programs the drive to abort the current move and remain enabled when a tracking error occurs.
						10	If set: programs the drive to latch a fault when output current is limited by the I^2T algorithm.
						11	FPGA failure. This bit is read-only.
						12	Command input lost fault. If set: programs the drive to latch in the disabled state when the command input is lost.
						13	Unable to initialize internal drive hardware. This bit is read-only.
						14	If set, programs the drive to latch a fault when there is safety circuit consistency check failure.
						15	If set, programs the drive to latch a fault when the drive is unable to control motor current.
						16	If set, programs the drive to latch a fault when the motor wiring is disconnected (see Open Motor Wiring Check Current (0x19D).
						17-31	Reserved.
0xA8	0xA9	0x4A8	0x2320	RF	INT16	Digital Input Command Configuration. Defines the configuration of the digital input commands when the drive is running in a mode that uses them as a control source. The lower 8 bits control the PWM input configuration for controlling current and velocity modes. The upper 8 bits configure the 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	Invert the PWM input if set.
						2	Invert the sign input if set.
						3	Allow 100% duty cycle if set. If clear, treat 100% duty cycle as a zero command, providing a measure of safety in case of controller failure or cable break.
						4	If set, use parameter 0xB6 as deadband for PWM input.
						5	If set, allow longer PWM periods (up to 50ms).
						6	For Feature Set C products, setting this bit will cause the Motor Hall Offset (0x4F) to be added to the angle calculated in UV mode. For Feature Set E products, see UV Configuration (0x180).
						7	Reserved for future use.
						8-9	Input pin interpretation for position mode (see below).

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>
<table>
<tr><td></td><td><table>
<tr><td>Value</td><td>Description</td></tr>
<tr><td>0</td><td>Step & Direction mode.</td></tr>
<tr><td>1</td><td>Separate up & down counters.</td></tr>
<tr><td>2</td><td>Quadrature encoder input.</td></tr>
<tr><td>3</td><td>PWM input commands absolute position.</td></tr>
</table></td></tr>
<tr><td>10-11</td><td>Reserved for future use.</td></tr>
<tr><td>12</td><td>Pulses are counted on their falling edge if this bit is clear, rising edge if set. This bit has no effect when the inputs are configured as encoder inputs.</td></tr>
<tr><td>13</td><td>Causes the direction of the input to be reversed. Works for all three modes..</td></tr>
<tr><td>14-15</td><td>Identify which input pins to use.
<table>
<tr><td>Value</td><td>Description</td></tr>
<tr><td>0</td><td>Single ended high speed inputs.</td></tr>
<tr><td>1</td><td>Multi-mode encoder port.</td></tr>
<tr><td>2</td><td>Differential high speed inputs.</td></tr>
<tr><td>3</td><td>Use primary encoder inputs.</td></tr>
</table></td></tr>
</table>
</td></tr>
<tr><td>0xA9</td><td>0xAA</td><td>0x4A9</td><td>0x2321</td><td>RF</td><td>INT32</td><td>Digital Command Input Scaling Factor. This value gives the amount of current to command at 100% PWM input. The scaling depends on what the PWM input is driving:
Current mode: 0.01 A
Velocity (Accelus): 0.1 counts/s
In position mode the scaling factor is a ratio of two 16-bit values. The first word passed gives the numerator and the second word gives the denominator. This ratio determines the number of encoder units moved for each pulse (or encoder count) input.
For example, a ratio of 1/3 would cause the motor to move 1 encoder unit for every three input steps.</td></tr>
<tr><td>0xAA</td><td>0xAB</td><td>0x4AA</td><td>0x2196</td><td>R*</td><td>U16</td><td>Raw Input State. The 16-bit value returned by this command gives the current state (high/low) of the drive’s input pins. Unlike Input Pin States (0xA6), no debounce is applied when reading the inputs using this variable.
The bits are mapped in the same order as Input Pin States (0xA6).</td></tr>
<tr><td>0xAB</td><td>0xAC</td><td>0x4AB</td><td>0x2194</td><td>R</td><td>U16</td><td>Output States And Program Control. When read, this parameter gives the</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>active/inactive state of the drive’s general-purpose digital outputs. Each bit represents an input number. Bit 0 = digital output 0 (OUT0), bit 1 = output 1 (OUT1), etc., up to output n (OUT*n*), the number of digital outputs on the drive. Additional bits are ignored.
Outputs that have been configured for program control can be set by writing to this parameter. Set a bit to activate the output. It will be activated high or low according to how it was programmed. Clear a bit to make the output inactive. If an output was not configured for program control it will not be affected.</td></tr>
<tr><td>0xAC</td><td>0xAD</td><td>0x4AC</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 the same way as the Drive Event Status Register (0xA0), but instead of giving the present status of the drive, the sticky version indicates any bits in the event status that have been set since the last reading of the sticky register.
The sticky register is similar to the Latched Event Status Register (0xA1), but the latched register must be cleared explicitly, whereas the sticky register is cleared automatically each time it is read.</td></tr>
<tr><td>0xAD</td><td>0xAE</td><td>0x4AD</td><td>0x1018:2
or
0x2384:13</td><td>R*</td><td>INT16</td><td>Drive Hardware Type. Also known as Product Code. Identifies the specific drive model. This is an augmented version of Product Family (0x87).
<table>
<tr><td>Value</td><td>Product</td></tr>
<tr><td>0x0000</td><td>ASC: Accelus Card.</td></tr>
<tr><td>0x0002</td><td>ASP: Accelus Panel.</td></tr>
<tr><td>0x0100</td><td>JSP: Junus Panel.</td></tr>
<tr><td>0x0200</td><td>ACM: Accelnet Module.</td></tr>
<tr><td>0x0201</td><td>XSL: Xenus Panel (obsolete).</td></tr>
<tr><td>0x0203</td><td>ACP: Accelnet Panel (obsolete).</td></tr>
<tr><td>0x0206</td><td>XSL-R: Xenus Panel, resolver version.</td></tr>
<tr><td>0x0207</td><td>XSL: Xenus Panel.</td></tr>
<tr><td>0x0209</td><td>ACJ: Accelnet Micro Panel.</td></tr>
<tr><td>0x020B</td><td>ACP: Accelnet Panel.</td></tr>
<tr><td>0x020C</td><td>ACK: Accelnet Micro Module.</td></tr>
<tr><td>0x020E</td><td>Special.</td></tr>
<tr><td>0x020F</td><td>Special.</td></tr>
<tr><td>0x0210</td><td>ACJ-S: Accelnet Micro Panel, analog encoder version.</td></tr>
</table>
</td></tr>
</table>

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description	
						0x0240	STM: Stepnet Module.
						0x0242	STP: Stepnet Panel.
						0x0243	STL: Stepnet Micro Module.
						0x0300	ASP: Accelnet Panel, dual axis.
						0x0310	XSJ(S): Xenus Micro Panel.
						0x0320	XTL: Xenus Panel, resolver version.
						0x0330	XTL(S): Xenus Panel.
						0x0340	XSJ-R: Xenus Micro Panel, resolver version.
						0x0380	AEP: Accelnet EtherCAT Panel.
						0x0391	AMP: Accelnet Macro Panel.
						0x0350	STX: Stepnet AC Panel.
						0x0370	ACK-R: Accelnet Micro Module, resolver version.
						0x03A0	ADP: Accelnet Panel.
						0x1000	XEL: Xenus Plus EtherCAT.
						0x1010	XML: Xenus Plus MACRO.
						0x1020	XPL: Xenus Plus CAN.
						0x1030	AEM: 1 axis Accelnet Plus Module EtherCAT.
						0x1040	APM: 1 axis Accelnet Plus Module CAN.
						0x1050	AE2: 2 axis Accelnet Plus Module EtherCAT.
						0x1060	AP2: 2 axis Accelnet Plus Module CAN.
						0x1070	SEM: 1 axis Stepnet Plus Module EtherCAT.
						0x1080	SPM: 1 axis Stepnet Plus Module CAN.
						0x1090	SE2: 2 axis Stepnet Plus Module EtherCAT.
						0x10A0	SP2: 2 axis Stepnet Plus Module CAN.
						0x10B0	XE2: 2 axis Xenus Plus Dual EtherCAT
						0x10B8	XE2-R: 2 axis Xenus Plus Dual EtherCAT resolver version
						0x10C0	BE2: 2 axis Accelnet Plus Panel EtherCAT
						0x10C8	BE2-R: 2 axis Accelnet Plus Panel EtherCAT resolver version
						0x10D0	XP2: 2 axis Xenus Plus Dual CAN
						0x10D8	XP2-R: 2 axis Xenus Plus Dual CAN resolver version
						0x10E0	BP2: 2 axis Accelnet Plus Panel CAN

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="9"></td><td rowspan="9"></td><td rowspan="9"></td><td rowspan="9"></td><td rowspan="9"></td><td rowspan="9"></td><td>0x10E8</td><td>BP2-R: 2 axis Accelnet Plus Panel CAN resolver version</td></tr>
<tr><td>0x10F0</td><td>TE2: 2 axis Stepnet Plus Panel EtherCAT</td></tr>
<tr><td>0x1100</td><td>TP2: 2 axis Stepnet Plus Panel CAN</td></tr>
<tr><td>0x1110</td><td>BEL: 1 axis Accelnet Plus Panel EtherCAT</td></tr>
<tr><td>0x1118</td><td>BEL-R: 1 axis Accelnet Plus Panel EtherCAT resolver version</td></tr>
<tr><td>0x1120</td><td>BPL: 1 axis Accelnet Plus Panel CAN</td></tr>
<tr><td>0x1128</td><td>BPL-R: 1 axis Accelnet Plus Panel CAN resolver version</td></tr>
<tr><td>0x1130</td><td>TEL: 1 axis Stepnet Plus Panel EtherCAT</td></tr>
<tr><td>0x1150</td><td>SP4: 4 axis Stepnet Plus Module CAN</td></tr>
<tr><td>0xAE</td><td>0xAF</td><td>0x4AE</td><td>0x60F6:3</td><td>RF</td><td>INT16</td><td colspan="2">Current Loop Offset. This value is added to the commanded motor current. It can compensate for a directional bias affecting the current loop. Units: 0.01 A.</td></tr>
<tr><td rowspan="8">0xAF</td><td rowspan="8">0xB0</td><td rowspan="8">0x4AF</td><td rowspan="8">0x2420</td><td rowspan="8">RF</td><td rowspan="8">INT32</td><td colspan="2">Miscellaneous Drive Options Register. Bit-mapped as follows:</td></tr>
<tr><td>Bits</td><td>Option</td></tr>
<tr><td>0</td><td>If set, input pins 1, 2, and 3 are pulled high on the drive. If clear, the pins are not pulled up. This option is only available on the Junus drive.</td></tr>
<tr><td>1</td><td>Reserved.</td></tr>
<tr><td>2</td><td>If set, limit switch inputs will only abort a trajectory in progress, but will not affect current output. If clear, limit switches limit current.</td></tr>
<tr><td>3</td><td>If set, save PDO configuration to a file in the CVM file system when a “Save to Flash” command is received over the CANopen network. If clear, a PDO is not saved.</td></tr>
<tr><td>4</td><td>If set, a limit switch activation will be treated as a fault in the CANopen Status Word (CANopen index 0x6041 as described in the *CANopen Programmer’s Manual*).</td></tr>
<tr><td>5-31</td><td>Reserved.</td></tr>
<tr><td>0xB0</td><td>0xB1</td><td>0x4B0</td><td>0x2260</td><td>R</td><td>INT16</td><td colspan="2">Motor Phase Angle. Writes are only useful when running in diagnostic micro-stepping mode. Units: degrees.</td></tr>
<tr><td>0xB1</td><td>0xB2</td><td>0x4B1</td><td>0x21C1</td><td>RF</td><td>INT16</td><td colspan="2">Increment Rate For Phase Angle When In Micro Stepping Mode. Only used in diagnostic micro-stepping mode. Desired State (0x24) = 42 (microstepping mode). Units: degrees/s.</td></tr>
<tr><td rowspan="2">0xB2</td><td rowspan="2">0xB3</td><td rowspan="2">0x4B2</td><td rowspan="2">0x21C0</td><td rowspan="2">RF</td><td rowspan="2">U16</td><td colspan="2">Commutation Mode. Also known as Phasing Mode. Controls the mechanism used by the drive to compute the motor phasing angle. Determines what inputs the drive uses to initialize and maintain the phase angle, as follows:</td></tr>
<tr><td>Value</td><td>Mode</td></tr>
</table>

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description	
						0	Standard mode. Encoder-based sinusoidal commutation for brushless motors. Use digital Hall inputs to initialize phase, then switch to an encoder to maintain phase. The encoder is the primary sensing device with the Hall effect sensors used to monitor and adjust the phase angle as necessary during operation.
						1	Trapezoidal (Hall based) phasing. The Hall sensors are used for phasing all the time. This mode can be used if no encoder is available.
						2	Like mode 0 except that the phase angle is not adjusted based on the Hall inputs. Hall sensors are still required to initialize the phase angle at startup.
						3	Analog Halls (90 degrees). Only available on drives with the necessary analog Hall inputs.
						4	DC brush motor mode.
						5	Algorithmic Phase Initialization mode (wake & wiggle, no Halls). See the *CME 2 User Guide* for more information on Algorithmic Phase Initialization.
						6	Encoder based phasing. Use with resolver or Servo-Tube motors.
						7	Trapezoidal commutation with phase angle interpolation.
0xB3	0xB4	0x4B3	0x2384:23	F*	INT16	Analog Encoder Scaling Factor. This parameter selects the resolution of an analog encoder input. The parameter is not used for other encoder types.	
0xB4	0xB5	0x4B4	0x2263	R*	INT16	Encoder Phase Angle. For feedback types, such as resolver, that can also calculate phase angle information. This parameter allows the phase information to be read directly.	
0xB5	0xB6	0x4B5	0x2353	R*	INT32	Homing Adjustment. This parameter is updated after each successful homing operation. The value it contains is the size of the actual position adjustment made in the last home sequence. Units: counts.	
0xB6	0xB7	0x4B6	0x2322	RF	U16	PWM Input Frequency. This is the frequency of the 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 a dead band value, this input should be set in the range 0 to 32767 which corresponds to a dead band of 0 to 100% of the PWM duty cycle.	
0xB7	0xB8	0x4B7	0x2141	R*	U32	System Time. Time since start up. Units: ms.	
0xB8	0xB9	0x4B8	0x607D:2	RF	INT32	Positive Software Limit. This parameter is only available on drives that support trajectory generation and homing. Software limits are only in effect after the drive has been referenced (i.e. homing has been successfully completed). Set to less than negative software limit to disable. Units: counts.	
0xB9	0xBA	0x4B9	0x607D:1	RF	INT32	Negative Software Limit. Software limits are only in effect after the drive has been	

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**
						referenced (i.e. homing has been successfully completed). Set to greater than positive software limit to disable. Units: counts.
0xBA	0xBB	0x4BA	0x2120	RF	INT32	Position Tracking Error Limit. Also known as Tracking Error Window. If the Position Loop Error (0x35) exceeds this value then the tracking error bit (bit 18) of the Drive Event Status Register (0xA0) is set and the motor is stopped. Using Fault Mask (0xA7), the tracking error event can be configured to either disable the drive immediately, or abort the present move and continue holding position. Units: counts.
0xBB	0xBC	0x4BB	0x6065	RF	INT32	Position Tracking Warning Limit. If the Position Loop Error (0x35) exceeds this value then the tracking warning bit (bit 19) of the Drive Event Status Register (0xA0) is set. Units: counts.
0xBC	0xBD	0x4BC	0x6067	RF	INT32	Position Tracking Window Limit. If the Position Loop Error (0x35) exceeds this value then the tracking window bit (bit 25) of the Drive Event Status Register (0xA0) is set. Units: counts.
0xBD	0xBE	0x4BD	0x6068	RF	U16	Time Delay For Position Tracking Error Limit (0xBA). The tracking window bit (bit 25) of the Drive Event Status Register (0xA0) will not be cleared until the Position Loop Error (0x35) has been within the Position Tracking Error Limit (0xBA) for at least this long. Units: ms.
0xBE	0xBF	0x4BE	0x2253	RF	U32	Software Limit Deceleration. The deceleration rate used to stop a motor when approaching a software limit. Units: 10 counts/s^2.
0xBF	0xC0	0x4BF	0x2351	RF	U16	Homing Current Delay Time (used with home to hard stop mode only). When performing a home to hard stop, the amplifier will push against the stop for this long before sampling the home position. Units: ms.
0xC0	0xC1	0x4C0	None	R*	INT16	Network Node ID. This is the drive's present ID as read at system startup. The node ID is only read at system startup, so this value will not change unless the drive is reset. See Network Node ID Configuration (0xC1).
0xC1	0xC2	0x4C1	0x21B0	RF	INT16	Network Node ID Configuration. Defines how a drive's node ID is calculated, and specifies the drive's network bit rate. The ID is calculated at startup (and only at startup) using a combination of general-purpose input pins and a programmed offset value. On certain models, an address switch is also used. The resulting value is clipped to a 7-bit ID in the range 0 to 127. Bit-mapped as follows: Bits | Description

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description		
						0-6	Give the node ID offset value that will be added to the value read from the input pins and address selector switch.	
						7	Used only on DeviceNet firmware. If this bit is set, then the drive will be software disabled on startup and will remain disabled until it is enabled by a DeviceNet I/O message with the enable bit set.	
						8-10	Number of input pins (0-7) to read on startup for the node ID value. If input pins are used (i.e., the value in bits 8-10 is not zero), the inputs can be mapped to node ID bits through the parameter Input Pin Mapping For Node ID Selection (0x103).	
						11	This bit is ignored on drives that do not have an address switch. On drives with an address switch, setting this bit programs the drive to use the address selector switch as part of the address calculation. In this case, the node ID value is equal to the sum of: ▪ The value read from the designated input pins, shifted up 4 bits. ▪ The address switch value. ▪ The programmed offset value. Note that since the node ID is always clipped to the lowest 7 bits, no more than 3 input pins will ever have an effect on the node address when the address switch is used.	
						12-15	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 for future use
0xC2	0xC3	0x4C2	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 the current position as home. If bit 5 is set, then move in the direction specified by bit 4 and set the location of the first index pulse as

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description		
								home. Bit 6 is not used in this mode.
							1	Move in the direction specified by bit 4 until a limit switch is encountered. Then move in the other direction out of limit. If bit 5 is clear, then the edge location is home. If bit 5 is set, then the next index pulse is home. Bit 6 not used in this mode.
							2	Home on a constant home switch. The initial move is made in the direction specified by bit 4. When the home switch is encountered, the direction is reversed. The edge of the home switch is set as home if bit 5 is clear. If bit 5 is set, then an index pulse is used as the home position. Bit 6 is used to define which index pulse is used.
							3	Home on an intermittent home switch. This mode works the same as mode 2 except that if a limit switch is encountered when initially searching for home, then the direction is reversed. In mode 2, hitting a limit switch before finding home would be considered an error. Bit 8 identifies which edge of the home to search for (positive or negative).
							4	Home to a hard stop. This moves in the direction specified in bit 4 until the home current limit is reached. It then presses against the hard stop using that current value until the home delay time expires. If bit 5 (index) is set, drive away from the hard stop until an index is found.
							15	Immediate home. This value causes the amp to be referenced immediately on power-up. Once the encoder is initialized, the home offset value is added to the encoder position and the result is set as the 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 the pulse on the DIR side of the sensor edge. DIR is the direction specified by bit 4 of this word.	
						7	If set, capture falling edge of index. Capture rising edge if clear.	
						8	When using a momentary home switch, this bit identifies which edge of the home switch to reference on. If set, then the negative edge is used; if clear the positive edge is used.	
						9	If set, make a move to the zero position when homing is finished. If clear, the zero position is found, but not moved to.	
						10	If set, the homing sequence will run as normal, but the actual position will not be adjusted at the end. Note that even though the actual position is not adjusted, the parameter Homing Adjustment (0xB5) is updated with the size of the adjustment (in counts) that would have been made. Also, if bit 10 is set then no move to zero is made regardless of the setting of bit 9.	

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0xC3</td><td>0xC4</td><td>0x4C3</td><td>0x6099:1</td><td>RF</td><td>INT32</td><td>Homing Velocity (fast moves). This velocity value is used during segments of the homing procedure that may be handled at high speed. Generally, this means moves in which the home sensor is being located, but the edge of the sensor is not being found. Units: 0.1 counts/s.</td></tr>
<tr><td>0xC4</td><td>0xC5</td><td>0x4C4</td><td>0x6099:2</td><td>RF</td><td>INT32</td><td>Homing Velocity (slow moves). This velocity value is used for homing segments that require low speed, such as cases where the edge of a homing sensor is being sought. Units: 0.1 counts/s.</td></tr>
<tr><td>0xC5</td><td>0xC6</td><td>0x4C5</td><td>0x609A</td><td>RF</td><td>U32</td><td>Homing Acceleration/Deceleration. This value defines the acceleration used for all homing moves. The same value is used at the beginning and ending of moves (i.e. there is no separate deceleration value). Units: 10 counts/s^2.</td></tr>
<tr><td>0xC6</td><td>0xC7</td><td>0x4C6</td><td>0x607C</td><td>RF</td><td>INT32</td><td>Home Offset. The home offset is the difference between the zero position for the application and the machine home position (found during homing). During homing the home position is found. Once the homing is completed the zero position is offset from the home position by adding the offset to the home position. All subsequent absolute moves shall be taken relative to this new zero position. Units: counts.</td></tr>
<tr><td>0xC7</td><td>0xC8</td><td>0x4C7</td><td>0x2350</td><td>RF</td><td>INT16</td><td>Homing Current Limit (home to hard stop mode only). Home current in hard stop mode, in which the drive drives the motor to the mechanical end of travel (hard stop). End of travel is recognized when the drive outputs the Hard Stop Mode Home Current for the Homing Current Delay Time (0xBF).Units: 0.01 A.</td></tr>
<tr><td>0xC8</td><td>0xC9</td><td>0x4C8</td><td>None</td><td>RF</td><td>INT16</td><td>Trajectory Profile Mode. To set profile in CANopen see CAN object 0x6086 in the CANopen Programmers Manual. Bit-mapped as follows:
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0-2</td><td>Give the trajectory profile mode. The possible trajectory modes are described below.
<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>0</td><td>Trapezoidal profile mode. Uses position/distance, velocity, acceleration and deceleration. Any of those parameters may be changed during the 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. None of these parameters may be changed while a move is in progress (although the move may be aborted). The 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 the 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 the move. Set velocity to zero to stop.</td></tr>
</table></td></tr>
</table></td></tr>
</table>

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**		
							3	PVT profile mode. Use of this mode through the serial interface is not presently supported.
						8	If set, this is a relative move. If clear, this is an absolute move.	
						Valid settings for this parameter:		
						Value	Description	
						0	Absolute move, trapezoidal profile.	
						1	Absolute move, S-curve profile.	
						256	Relative move, trapezoidal profile.	
						257	Relative move, S-curve profile.	
						2	Velocity profile.	
0xC9	0xCA	0x4C9	0x2252	R*	INT16	Trajectory Status Register. This parameter gives status information about the trajectory generator. Bit-mapped as follows:		
						Bits	Description	
						0-8	Reserved for future use.	
						9	Cam table underflow.	
						10	Reserved for future use.	
						11	Homing error. If set, an error occurred in the last home attempt. Cleared by a home command.	
						12	Referenced. Set when a homing command has been successfully executed. Cleared by a home command.	
						13	Homing. If set, the drive is running a home command.	
						14	Set when a move is aborted. Cleared at the start of the next move.	
						15	In-Motion Bit. If set, the trajectory generator is presently generating a profile.	
0xCA	0xCB	0x4CA	0x607A	RF	INT32	Trajectory Generator Position Command. Units: Counts. Meaning depends on move type as described below:		
						Move Type	Meaning	
						Relative	Move distance.	
						Absolute	Target position.	
						Velocity	Direction: 1 for positive, -1 for negative.	
0xCB	0xCC	0x4CB	0x6081	RF	INT32	Trajectory Maximum Velocity. The trajectory generator will attempt to reach this velocity during a move. Units: 0.1 counts/s.		

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**
0xCC	0xCD	0x4CC	0x6083	RF	U32	Trajectory Maximum Acceleration. The trajectory generator will attempt to reach this acceleration during a move. For s-curve profiles, this value is also used to decelerate at the end of a move. Units: 10 counts/s^2.
0xCD	0xCE	0x4CD	0x6084	RF	U32	Trajectory Maximum Deceleration. In trapezoidal trajectory mode, this value will be used to decelerate at the end of a move. Units: 10 counts/s^2.
0xCE	0xCF	0x4CE	0x2121	RF	U32	Trajectory Maximum Jerk. Also known as Trajectory Jerk Limit. The S-curve profile generator uses this value as the jerk (rate of change of acceleration/deceleration) during moves. Other profiles types do not use jerk limit. Units: 100 counts/s^3.
0xCF	0xD0	0x4CF	0x6085	RF	U32	Trajectory Abort Deceleration. If a move is aborted, this value will be used by the trajectory generator to decelerate to a stop. Units: 10 counts/s^2.
0xD0	0xD1	0x4D0	0x2192:9	RF	U16	Input 8 Configuration. See Input 0 Configuration (0x78).
0xD1	0xD2	0x4D1	0x2192:10	RF	U16	Input 9 Configuration. See Input 0 Configuration (0x78).
0xD2	0xD3	0x4D2	0x2192:11	RF	U16	Input 10 Configuration. See Input 0 Configuration (0x78).
0xD3	0xD4	0x4D3	0x2192:12	RF	U16	Input 11 Configuration. See Input 0 Configuration (0x78).
0xD4	0xD5	0x4D4	0x2192:13	RF	U16	Input 12 Configuration. See Input 0 Configuration (0x78).
0xD5	0xD6	0x4D5	0x2192:14	RF	U16	Input 13 Configuration. See Input 0 Configuration (0x78).
0xD6	0xD7	0x4D6	0x2192:15	RF	U16	Input 14 Configuration. See Input 0 Configuration (0x78).
0xD7	0xD8	0x4D7	0x2192:16	RF	U16	Input 15 Configuration. See Input 0 Configuration (0x78).
0xD8	0xD9	0x4D8	0x2150	RF	U16	Regen Resistor Resistance. Units: 0.1 Ω.
0xD9	0xDA	0x4D9	0x2151	RF	U16	Regen Resistor, Continuous Power. Units: W.
0xDA	0xDB	0x4DA	0x2152	RF	U16	Regen Resistor, Peak Power. Units: W.
0xDB	0xDC	0x4DB	0x2153	RF	U16	Regen Resistor, Time At Peak. Units: ms.
0xDC	0xDD	0x4DC	0x2154	RF	INT16	Regen Turn On Voltage Units: 0.1 V.
0xDD	0xDE	0x4DD	0x2155	RF	INT16	Regen Turn Off Voltage. Units: 0.1 V.
0xDE	0xDF	0x4DE	0x2384:20	F*	INT16	Drive’s Peak Current Rating For Its Internal Regen Transistor. Units: 0.01 A.
0xDF	0xE0	0x4DF	0x2384:21	F*	INT16	Drive's Continuous Current Rating For Its Internal Regen Transistor.
0xE0	0xE1	0x4E0	0x2384:22	F*	INT16	Drive's Time At Peak Current For Its Internal Regen Transistor. Units: ms.
0xE1	0xE2	0x4E1	0x2156	F	String	Regen Resistor Model Number String.

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0xE2</td><td>0xE3</td><td>0x4E2</td><td>0x2157</td><td>R*</td><td>INT16</td><td>Regen Resistor Status. Bit-mapped as follows:
<table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0</td><td>Set if the 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 the regen circuit is open due to an overload condition. The overload may be caused by either the resistor settings or the internal drive protections.</td></tr>
<tr><td>3-15</td><td>Reserved.</td></tr>
</table></td></tr>
<tr><td>0xE3</td><td>0xE4</td><td>0x4E3</td><td>0x2382</td><td>RF</td><td>U16</td><td>Position Loop Output Gain Multiplier. The output of the position loop is multiplied by this value before being passed to the 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>0xE5</td><td>0x4E4</td><td>0x21C2</td><td>RF</td><td>INT16</td><td>Maximum Current to use with algorithmic phase initialization. See code 5 of Commutation Mode (0xB2). Units: 0.01 A.</td></tr>
<tr><td>0xE5</td><td>0xE6</td><td>0x4E5</td><td>0x21C3</td><td>RF</td><td>U16</td><td>Algorithmic Phase Initialization Timeout. See code 5 of Commutation Mode (0xB2). Units: ms.</td></tr>
<tr><td>0xE6</td><td>0xE7</td><td>0x4E6</td><td>0x21D8</td><td>RF</td><td>INT32</td><td>Maximum Velocity Adjustment. This is the maximum velocity adjustment made by the stepper outer position loop when enabled. This parameter is only used when the stepper outer loop is engaged (when bit 1 of Stepper Configuration & Status (0xEE) is set). Units: 0.1 steps/s.</td></tr>
<tr><td>0xE7</td><td>0xE8</td><td>0x4E7</td><td>0x21D7</td><td>RF</td><td>U16</td><td>Proportional Gain For Stepper Outer Loop. This parameter gives the gain used for calculating a velocity adjustment based on Position Loop Error (0x35). This parameter is only used when the stepper outer loop is engaged (when bit 1 of Stepper Configuration & Status (0xEE) is set).</td></tr>
<tr><td>0xE8</td><td>0xE9</td><td>0x4E8</td><td>0x21D0</td><td>RF</td><td>INT16</td><td>Holding Current For Microstepping Mode. Units: 0.01 A.</td></tr>
<tr><td>0xE9</td><td>0xEA</td><td>0x4E9</td><td>0x21D1</td><td>RF</td><td>U16</td><td>Run to Hold Time For Microstepping Mode. Units: ms.</td></tr>
<tr><td>0xEA</td><td>0xEB</td><td>0x4EA</td><td>0x21D2</td><td>RF</td><td>U16</td><td>Detent Correction Gain Factor For Microstepping Mode.</td></tr>
<tr><td>0xED</td><td>0xEE</td><td>0x4ED</td><td>0x21D5</td><td>RF</td><td>U16</td><td>Holding Current To Fixed Voltage Output Time for Microstepping Mode. Time delay from entering hold current before entering the special voltage control mode of operation. This mode trades the normal tight control of current for very low jitter on the motor position. Used in stepper mode only. Set to 0 to disable this feature. Units: ms.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0xEE</td><td>0xEF</td><td>0x4EE</td><td>0x21D6</td><td>RF</td><td>INT16</td><td>Stepper Configuration & Status. Bit-mapped as follows:
<table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0</td><td>Use the encoder input for phase compensation if enabled. Pure stepper mode if disabled.</td></tr>
<tr><td>1</td><td>Use on outer position loop to adjust the stepper position based on Position Loop Error (0x35). When this bit is set, the gain value Maximum Velocity Adjustment (0xE6) is multiplied by the Position Loop Error (0x35), and the result is a velocity that is added to the microstepping position.</td></tr>
<tr><td>2-15</td><td>Reserved.</td></tr>
</table></td></tr>
<tr><td>0xF0</td><td>0xF1</td><td>0x4F0</td><td>0x2195:1</td><td>RF</td><td>U16</td><td>Debounce Time For Input 0. Units: ms.</td></tr>
<tr><td>0xF1</td><td>0xF2</td><td>0x4F1</td><td>0x2195:2</td><td>RF</td><td>U16</td><td>Debounce Time For Input 1. Units: ms.</td></tr>
<tr><td>0xF2</td><td>0xF3</td><td>0x4F2</td><td>0x2195:3</td><td>RF</td><td>U16</td><td>Debounce Time For Input 2. Units: ms.</td></tr>
<tr><td>0xF3</td><td>0xF4</td><td>0x4F3</td><td>0x2195:4</td><td>RF</td><td>U16</td><td>Debounce Time For Input 3. Units: ms.</td></tr>
<tr><td>0xF4</td><td>0xF5</td><td>0x4F4</td><td>0x2195:5</td><td>RF</td><td>U16</td><td>Debounce Time For Input 4. Units: ms.</td></tr>
<tr><td>0xF5</td><td>0xF6</td><td>0x4F5</td><td>0x2195:6</td><td>RF</td><td>U16</td><td>Debounce Time For Input 5. Units: ms.</td></tr>
<tr><td>0xF6</td><td>0xF7</td><td>0x4F6</td><td>0x2195:7</td><td>RF</td><td>U16</td><td>Debounce Time For Input 6. Units: ms.</td></tr>
<tr><td>0xF7</td><td>0xF8</td><td>0x4F7</td><td>0x2195:8</td><td>RF</td><td>U16</td><td>Debounce Time For Input 7. Units: ms.</td></tr>
<tr><td>0xF8</td><td>0xF9</td><td>0x4F8</td><td>0x2195:9</td><td>RF</td><td>U16</td><td>Debounce Time For Input 8. Units: ms.</td></tr>
<tr><td>0xF9</td><td>0xFA</td><td>0x4F9</td><td>0x2195:10</td><td>RF</td><td>U16</td><td>Debounce Time For Input 9. Units: ms.</td></tr>
<tr><td>0xFA</td><td>0xFB</td><td>0x4FA</td><td>0x2195:11</td><td>RF</td><td>U16</td><td>Debounce Time For Input 10. Units: ms.</td></tr>
<tr><td>0xFB</td><td>0xFC</td><td>0x4FB</td><td>0x2195:12</td><td>RF</td><td>U16</td><td>Debounce Time For Input 11. Units: ms.</td></tr>
<tr><td>0xFC</td><td>0xFD</td><td>0x4FC</td><td>0x2195:13</td><td>RF</td><td>U16</td><td>Debounce Time For Input 12. Units: ms.</td></tr>
<tr><td>0xFD</td><td>0xFE</td><td>0x4FD</td><td>0x2195:14</td><td>RF</td><td>U16</td><td>Debounce Time For Input 13. Units: ms.</td></tr>
<tr><td>0xFE</td><td>0xFF</td><td>0x4FE</td><td>0x2195:15</td><td>RF</td><td>U16</td><td>Debounce Time For Input 14. Units: ms.</td></tr>
<tr><td>0xFF</td><td>0x100</td><td>0x4FF</td><td>0x2195:16</td><td>RF</td><td>U16</td><td>Debounce Time For Input 15. Units: ms.</td></tr>
<tr><td>0x100</td><td>0x101</td><td>0x500</td><td>0x2184</td><td>RF</td><td>U32</td><td>CANopen Limit Status Mask. This parameter defines which bits in the Drive Event Status Register (0xA0) can set the limit bit (bit 11) of the CANopen Status Word (CANopen index 0x6041 as described in the *CANopen Programmer's Manual*). If a Drive Event Status Register (0xA0) bit and its corresponding Limit Mask bit are both set, then the CANopen Status Word limit bit is set. If all selected Drive Event Status</td></tr>
</table>

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**
						Register (0xA0) bits are clear, then the limit bit is clear.
0x101	0x102	0x501	0x2197	R*	INT16	Network Address Switch Value. This gives the current state of the CAN address switch. For drives without a switch, the value returned is undefined.
0x102	0x103	0x502	0x21B4	R*	INT16	Network Status Word. Bit-mapped as follows:

CANopen

Bits	Meaning	
0-1	CANopen node status. This field will take one of the following values:	
	Value	Status
	0	The CANopen interface is disabled.
	1	Stopped mode.
	2	Preoperational mode.
	3	Operational mode.
4	Set if the CANopen SYNC message is missing.	
5	Set on a CANopen guard error.	
8	Set if the CAN port is in 'bus off' state.	
9	Set if the CAN port is in 'transmit error passive' state.	
10	Set if the CAN port is in 'receive error passive' state.	
11	Set if the CAN port is in 'transmit warning' state.	
12	Set if the CAN port is in 'receive warning' state.	

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.

MACRO

0	Set if the MACRO network is detected.

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description	
						1	Set if the drive is being disabled by the MACRO master.
						2	Set if the MACRO network has been broken (i.e., once detected but now gone).
						3	Set on heartbeat error.
						4-15	Reserved.
0x103	0x104	0x503	0x21B1	F	U32	Input Pin Mapping For Node ID Selection. When Network Node ID Configuration (0xC1) indicates that 1 or more input pins will be used to select the node ID, this parameter is used to map input pins to ID bits.	
						Bits	Meaning
						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.

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>If bit 31 is zero, then a default bit mapping is used and the rest of this register is ignored. The default bit mapping uses the top N input pins and maps them such that the high numbered pins are used for higher numbered bits in the ID. For example; the Accelnet panel drive has 12 general-purpose input pins (0 to 11). If 3 of these pins are used for ID configuration and the default mapping is used, then the highest 3 pins (9, 10 and 11) will be used for the 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 the rest of this register will be used to define which input pin will be assigned to which bit of the ID. The input pins are numbered from 0 to 15 and each nibble of the register gives the input pin number associated with one bit of the 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 ID bit 0, input pin 1 will be used for ID bit 1, and input pin 0 will be used for ID bit 2.
Note that the CAN node ID is calculated at startup only. The input pins assigned to the node ID will be sampled once during power up and used to calculate the ID. These pins may be assigned other uses after power up if necessary.</td></tr>
<tr><td>0x104</td><td>0x105</td><td>0x504</td><td>0x21C4</td><td>RF</td><td>INT16</td><td>Algorithmic Phase Initialization Config. See code 5 of Commutation Mode (0xB2). This parameter is bit-mapped as follows:
<table>
<tr><td>Bits</td><td>Meaning</td></tr>
<tr><td>0</td><td>If clear, use algorithmic phase initialization. If set, force the phase angle to zero degrees.</td></tr>
<tr><td>1</td><td>If set, increment the initial phase angle by 90 degrees on each failed attempt.</td></tr>
<tr><td>2</td><td>If set, use Motor Hall Offset (0x4F) as the initial angle for the first phase initialization attempt. If clear, the first phase angle is zero.</td></tr>
<tr><td>3</td><td>Ignore limit switches during phase initialization if the switch is configured as trajectory based. Available in Feature set C only.</td></tr>
<tr><td>4-15</td><td>Reserved.</td></tr>
</table></td></tr>
<tr><td>0x105</td><td>0x106</td><td>0x505</td><td>0x2360</td><td>RF</td><td>U16</td><td>Camming Configuration. Bit-mapped as follows.
For more information, see the *Copley Camming User Guide.*
<table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0-3</td><td>ID Number of the Cam Table to use (0-9).</td></tr>
<tr><td>4</td><td>Reserved.</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>
<table>
<tr><td>5</td><td>If set, exit table in forward direction.</td></tr>
<tr><td>6</td><td>If set, use the Camming Internal Generator. The internal generator runs at the constant velocity programmed in Camming Master Velocity (0x109).
If clear, use digital command input as configured in using Copley's CME 2 software camming controls or Input Pin States (0xA6).</td></tr>
<tr><td>7</td><td>If set, run tables stored in RAM. If clear, use tables stored in the flash file system.</td></tr>
<tr><td>8-11</td><td>Input number to use as Cam Trigger. Note: a value of 0 selects IN1, 1 selects IN2, etc.</td></tr>
<tr><td>12-14</td><td>Cam Trigger type:
<table>
<tr><td>Value</td><td>Type</td></tr>
<tr><td>0</td><td>None (Continuous): The active Cam Table is repeated continuously.</td></tr>
<tr><td>1</td><td>Use Input, Edge: The active Cam Table begins executing on the rising edge of the input pin selected by bits 8-11.</td></tr>
<tr><td>2</td><td>Use Input, Level: The active Cam Table will run as long as the input selected by bits 8-11 is high.</td></tr>
<tr><td>3</td><td>Use Master (Secondary) Encoder Index: The active Cam Table is executed when the drive receives an index pulse from the Master encoder. Index pulses received during execution are ignored.</td></tr>
<tr><td>7</td><td>Never trigger. This can be used to stop a CAM currently in progress</td></tr>
</table>
</td></tr>
</table>
</td></tr>
<tr><td>0x106</td><td>0x107</td><td>0x506</td><td>0x2361</td><td>RF</td><td>INT16</td><td>Camming delay, forward motion. This gives the delay (in master encoder counts) used when entering a cam table in the forward direction. Units: master command counts.</td></tr>
<tr><td>0x107</td><td>0x108</td><td>0x507</td><td>0x2362</td><td>RF</td><td>INT16</td><td>Camming delay, reverse motion. This gives the delay (in master encoder counts) used when entering a cam table in the reverse direction. Units: master command counts.</td></tr>
<tr><td>0x108</td><td>0x109</td><td>0x508</td><td>None</td><td>R</td><td>INT16</td><td>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 a CVM program.
Reading this parameter does not return any useful information.</td></tr>
<tr><td>0x109</td><td>0x10A</td><td>0x509</td><td>0x2363</td><td>RF</td><td>INT 32</td><td>Camming Master Velocity. Constant velocity of the Camming Internal Generator. Units: 0.1 counts/s.</td></tr>
<tr><td>0x10A</td><td>0x10B</td><td>0x50A</td><td>0x2403</td><td>R*</td><td>INT 32</td><td>Captured Home Position. Provides the position that the axis was in when an input pin configured as a home switch input became active. Configured by setting bits in the Position Capture Control Register (0x6C). Status of the captured data can be checked in the Position Capture Status Register (0x6D). Reading this variable resets</td></tr>
</table>

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**
						bits 4 & 7 of the Position Capture Status Register (0x6D). Units: counts.
0x10B	0x10C	0x50B	0x2422	R*	U32	Firmware Version Number (extended). The upper 16 bits give the same major/minor version number as Firmware Version Number (0x94). The lower 16 bits hold a release number (upper byte) and a reserved byte (lower).
0x10C	0x10D	0x50C	0x1017	RF	U16	CANopen Heartbeat Time. The frequency at which the 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 the Heartbeat Time is non-zero, then the heartbeat protocol is used regardless of the settings of the CANopen Node Guarding Time (0x10D) and CANopen Node Guarding Time (0x10D). Units: ms.
0x10D	0x10E	0x50D	0x100C	RF	U16	CANopen Node Guarding Time. This parameter gives the time between node-guarding requests that are sent from the CANopen master to this drive. The drive will respond to each request with a node-guarding message indicating the internal state of the drive. If the drive has not received a node-guarding request within the time period defined by the product of the Node Guarding Time and the CANopen Node Guarding Life Time Factor (0x10E), the drive will treat this lack of requests as a fault condition. Units: ms.
0x10E	0x10F	0x50E	0x100D	RF	U8	CANopen Node Guarding Life Time Factor. This object gives a multiple of the CANopen Node Guarding Time (0x10D). The drive expects to receive a node-guarding request within the time period defined by the product of the Guarding Time and the Lifetime Factor. If the drive has not received a node-guarding request within this time period, it treats the lack of requests as a fault.
0x10F	0x110	0x50F	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 the master position. The offset will immediately be cleared once it has been applied to the master position, so this parameter will normally be read back as zero when running in pulse and direction mode 23.
0x110	0x111	0x510	0x2404	R	INT 32	Time Stamp of Last High Speed Position Capture. If high speed position capture is enabled, this parameter gives the time of the last capture. Setting this parameter causes the drive to calculate its position at the set time if

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>position capture is enabled and the time is recent enough for the data to be available. The calculated position may be read from *Captured Position for High Speed Position Capture.* Units: microseconds.</td></tr>
<tr><td>0x111</td><td>0x112</td><td>0x511</td><td>0x2405</td><td>R*</td><td>INT 32</td><td>Captured Position for High Speed Position Capture. Units: counts.</td></tr>
<tr><td>0x112</td><td>0x113</td><td>0x512</td><td>0x2242</td><td>R</td><td>INT 32</td><td>Position Encoder Position. This is also the passive load position when used in passive mode. Units: counts.</td></tr>
<tr><td>0x113</td><td>0x114</td><td>0x513</td><td>0x1015</td><td>RF</td><td>INT16</td><td>CANopen emergency inhibit time. Units: milliseconds.</td></tr>
<tr><td>0x114</td><td>0x115</td><td>0x514</td><td>0x2381:5</td><td>RF</td><td>U16</td><td>Velocity loop Drain (integral bleed). Modifies the effect of velocity loop integral gain. The higher the Vi Drain value, the faster the integral sum is lowered. Range: 0 to 32K Default: 0.</td></tr>
<tr><td>0x115</td><td>0x116</td><td>0x515</td><td>0x2010</td><td>R</td><td>5 Words</td><td>Trajectory buffer access. This object can be used to load data into the drive's internal trajectory buffer, or send commands used to control the buffer. The trajectory buffer holds trajectory segments used in PVT mode. Data passed to the parameter consists of a 16-bit command code, followed by up to two 32-bit parameters. The first word passed to this parameter is bit-mapped. The data contained in this word identifies this access as either a buffer command, or a trajectory segment to be loaded into the buffer. If the most significant bit of the first word is set, then the write is treated as a command code. In this case no additional data is passed and the first word is formatted as follows:
<table>
<tr><td>Bits</td><td>Description</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>
</table>
The following command codes are supported:
<table>
<tr><td>Code</td><td>Description</td></tr>
<tr><td>0</td><td>Clear the buffer and abort any move in progress.</td></tr>
<tr><td>1</td><td>Pop the N most recently sent segments off the buffer. PVT profiles will continue to run as long as the buffer doesn't underflow. The number of segments to pop (N) is passed in the command data area. If there are less than N segments on the buffer, this acts the same as a buffer clear, except that the profile is not stopped except by underflow.</td></tr>
</table>
To write data to the trajectory buffer, the MSB of the first word should be clear. In this case the first word is formatted as follows:</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>
<table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0-7</td><td>Segment time in milliseconds.</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>
</table>
When writing a new PVT segment to the trajectory buffer, this first word is always followed by a 32-bit position value. The position is specified in units of encoder counts and can be interpreted as either absolute or relative based on bit 12 of the command word.

Optionally, the position can be followed by a 32-bit velocity value. Velocity is specified in units of 0.1 encoder counts / second. If the velocity is supplied, then the drive will use cubic polynomial interpolation between points when running the trajectory (PVT mode). If velocity is not supplied, then linear interpolation will be used (PT mode). It's legal to mix PVT and PT segments within the same move.

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

The first returned word is formatted as follows:
<table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0-7</td><td>Number of free locations in the 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>0x117</td><td>0x516</td><td>0x605A</td><td>RF</td><td>INT16</td><td>CANopen quick stop option code.</td></tr>
<tr><td>0x117</td><td>0x118</td><td>0x517</td><td>0x605B</td><td>RF</td><td>INT16</td><td>CANopen shutdown option code.</td></tr>
<tr><td>0x118</td><td>0x119</td><td>0x518</td><td>0x605C</td><td>RF</td><td>INT16</td><td>CANopen disable option code.</td></tr>
<tr><td>0x119</td><td>0x11A</td><td>0x519</td><td>0x605D</td><td>RF</td><td>INT16</td><td>CANopen halt option code.</td></tr>
<tr><td>0x11A</td><td>0x11B</td><td>0X51A</td><td>0x2080</td><td>F*</td><td>U32</td><td>Drive scaling configuration. Defines the units used for current and voltage readings from the drive:</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>
<table>
<tr><td>Bits</td><td colspan="2">Description</td></tr>
<tr><td>0-1</td><td colspan="2">Identify units for current readings:</td></tr>
<tr><td></td><td>0</td><td>0.01 A</td></tr>
<tr><td></td><td>1</td><td>0.001 A</td></tr>
<tr><td></td><td>2</td><td>0.0001 A</td></tr>
<tr><td></td><td>3</td><td>0.00001 A</td></tr>
<tr><td>2-7</td><td></td><td>Reserved</td></tr>
<tr><td>8-9</td><td colspan="2">Identify units for voltage readings:</td></tr>
<tr><td></td><td>0</td><td>0.1 V</td></tr>
<tr><td></td><td>1</td><td>0.01 V</td></tr>
<tr><td></td><td>2</td><td>0.001 V</td></tr>
<tr><td></td><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>0x120</td><td>0x121</td><td>0x520</td><td>0x2384:25</td><td>R*</td><td>INT16</td><td>Returns the number of axis implemented by this drive.</td></tr>
<tr><td>0x121</td><td>0x122</td><td>0x521</td><td>0x21B3</td><td>RF</td><td>INT16</td><td>Network options. Configures the drive’s network. The details of its meaning depend on the type of network implemented in the drive.
<table>
<tr><td colspan="2">CANopen</td></tr>
<tr><td>Bits</td><td>Meaning</td></tr>
<tr><td>0</td><td>Must be clear to select CANopen networking.</td></tr>
<tr><td>1-15</td><td>Reserved.</td></tr>
<tr><td colspan="2">DeviceNet</td></tr>
<tr><td>Bits</td><td>Meaning</td></tr>
<tr><td>0</td><td>Must be set to select DeviceNet networking.</td></tr>
<tr><td>1-15</td><td>Reserved.</td></tr>
<tr><td colspan="2">MACRO</td></tr>
<tr><td>Bits</td><td>Meaning</td></tr>
<tr><td>0</td><td>If set, position data sent over the MACRO network is shifted up 5 bits for compatibility with Delta-Tau controllers.</td></tr>
<tr><td>1</td><td>If set, the drive will be disabled on startup until it is enabled through the MACRO interface. If clear, the drive can be used without the MACRO interface connected until it starts receiving MACRO messages.</td></tr>
</table>
</td></tr>
</table>

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description		
						2	If set, return the primary encoder index state (high/low) in the home status bit of the MACRO status word. If clear, the state of any general purpose input configured as a home input will be used.	
						3	If set, the drive will attempt to synchronize its current loop update period to the MACRO ring period. The ring period must be an integer multiple of the drive's PWM period (parameter 0x85).	
						4-7	Identify the type of data sent from the drive in the first 16-bit MACRO register. 0=digital inputs, 1=secondary analog input 1, 2-15=reserved	
						8-9	Identify the type of data sent in the second 16-bit MACRO register: 0 – primary analog input. 1 – Primary encoder reading. 2 – Secondary encoder reading. 3 – Pulse & direction hardware count.	
						10-15	Reserved.	
						EtherCAT		
						Bits	Meaning	
						0	If set, disable some extra checks of the SYNC0 configuration which were added for improved network conformance.	
						1	If set, the drive will follow the EtherCAT state machine even when running in a non EtherCAT mode of operation.	
						2	If set, object 0x1002 is the bit-wise OR of all axes event status for multi-axis drives. If clear, 0x1002 is for axis 1 only.	
						3-15	Reserved.	
0x122	0x123	0x522	0x2384:26	F*	INT16	Internal maximum regen current. Units: mA.		
0x123	0x124	0x523	0x2220	RF	INT32	Motor encoder wrap position. Actual motor position will wrap back to zero when this value is reached. Setting this value to zero disables this feature. Units: counts		
0x124	0x125	0x524	0x2221	RF	INT32	Load encoder wrap position. Actual load position will wrap back to zero when this value is reached. Setting this value to zero disables this feature. Units: counts.		
0x125	0x126	0x525	None	RF	INT16	Configures the MACRO drive's encoder capture circuit. This parameter is only used on MACRO drives. Bit-mapped as follows:		
						Bits	Meaning	
						0-3	Type of capture to use.	
							Value	Description

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description		
							0	Capture on edge of encoder index.
							1	Capture using a general purpose input pin.
							2-15	Reserved.
						4-7	Input pin number to use if using capture type 1.	
						8	Active level; high if clear, low if set.	
						9	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.	
						10	If set, passive load encoder, if configured, will be captured. Passive load encoder currently only supports capture type 1 (general purpose input).	
						11-15	Reserved.	
0x126	0x127	0x526	0x2384:27	R*	INT16	FPGA firmware version number.		

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0x127</td><td>0x128</td><td>0x527</td><td>0x2370</td><td>RF</td><td>U32</td><td>Gain scheduling configuration:
<table>
<tr><td>Bits</td><td>Meaning</td></tr>
<tr><td>0-2</td><td>Key parameter for gain scheduling.
<table>
<tr><td>Value</td><td>Description</td></tr>
<tr><td>0</td><td>None. Setting the key parameter to zero disables gain scheduling.</td></tr>
<tr><td>1</td><td>Use value written to Gain scheduling key parameter value (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></td></tr>
<tr><td>3-7</td><td>Reserved.</td></tr>
<tr><td>8</td><td>If set, use the absolute value of key parameter for gain lookup.</td></tr>
<tr><td>9</td><td>If set, disable gain scheduling until the position encoder is referenced.</td></tr>
<tr><td>10-31</td><td>Reserved.</td></tr>
</table></td></tr>
<tr><td>0x128</td><td>0x129</td><td>0x528</td><td>0x2371</td><td>R</td><td>INT32</td><td>Gain scheduling key parameter value. When gain scheduling is enabled, the current value of the key parameter is stored here. When this parameter is selected as the key parameter for gain scheduling, then it may be written to manually move through entries in the gain scheduling table.</td></tr>
<tr><td>0x129</td><td>0x12A</td><td>0x529</td><td>0x2384:29</td><td>R</td><td>U32</td><td>Drive Hardware Options. Reserved for Copley Controls use.</td></tr>
<tr><td>0x12A</td><td>0x12B</td><td>0x52A</td><td>0x2222</td><td>F</td><td>U32</td><td>Motor encoder options. Used to specify various configuration options for the motor encoder. The mapping of option bits to function depends on the encoder type. Any bit not defined for a particular encoder should be considered reserved. Reserved bits should be set to zero to ensure compatibility with future firmware updates. Bit-mapped as follows:
<table>
<tr><td colspan="2">Quadrature Encoder</td></tr>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0</td><td>If set, ignore differential signal errors (if detected in hardware).</td></tr>
<tr><td>1</td><td>If set, select single ended encoder inputs (if available in hardware).</td></tr>
<tr><td>2</td><td>If set, ignore differential signal errors on index input only (if supported by hardware).</td></tr>
</table></td></tr>
</table>

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**	
						3	If set, don’t use index input at all. Useful when the index input is being used by a different encoder interface.
						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	Set if 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 the default 2.1 style.
						20-23	Number of least significant bits of the encoder reading to discard.
						SSI Encoder (Type 12)	
						Bits	Description
						0-5	Number of bits of position data available. A value of zero is invalid for these bits.
						8-11	Number of extra bits sent with position data.
						12	If set, ignore the first bit of data sent by the 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 LSB first.
						16-21	Encoder bit rate in 100 kHz units. If zero, default to 1 MHz.
						22	If set, use setting of encoder counts/rev 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.
						BiSS (Type 13)	
						Bits	Description
						0-5	Number of bits of single turn data.
						8-12	Number of bits of multi-turn data.
						15	If set, assume the encoder position data wraps after the number of encoder counts programmed in parameter Motor Encoder Counts (0x62).
						16	Set for mode-C encoder format.
						20	Set if encoder error and warning bits are active low.

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description	
						21	Set if encoder status bits are sent before position data. Clear if status bits are sent after position data.
						22	Set if encoder error bit is transmitted before warning bit. Clear for warning bit sent first.
						24-26	Number of alignment bits (reserved bits sent before position info).
						28	Use multi-mode port if set. If clear use primary encoder interface.
						30	Set for 2.5 MHz baud rate, clear for 4 MHz baud rate.
						Tamagawa, Panasonic, Harmonic Drives, etc. (Type 14)	
						Bits	Description
						0-5	Number of bits of single turn data.
						8-12	Number of bits of multi-turn data.
						16-19	Number of LSB to discard from reading.
						20-22	Number of consecutive CRC errors to ignore before generating an error.
						24-27	Encoder sub-type (0=Tamagawa, 1=Panasonic absolute, 2=HD systems, 3=Panasonic Incremental, 4=Sanyo Denki, 5=Tamagawa Single Turn).
						28	Bit rate (set for 4 Mbit, clear for 2.5 Mbit).
						30	If set, treat encoder battery errors as warnings.
0x12B	0x12C	0x52B	0x2223	F	U32	Load Encoder Options. Same as 0x12A, but affects load or position encoder.	
0x12C	0x129	0x52C	0x2384:28	R*	U32	Secondary firmware version for drives equipped with two processors.	
0x12D	0x12E	0x52D	0x2109	RF	9 or 14 words	Analog Input Filter Coefficients. A bi-quad filter which acts on the analog reference input. 9 word parameter, see *Analog Input Filters* in the *CME 2 User Guide.* 14 word parameter (Plus product only), see Filter Coefficients (p. 73).	
0x12E	0x12F	0x52E	0x2224	R*	U32	Motor encoder status. This parameter gives additional status information for the encoder. Bits set in the status word are latched and cleared when the status value is read. The format of this status word is dependent on the encoder type. Many error bits are taken directly from encoder data stream. For a full description of what these error bits mean, please consult the encoder manufacturer.	
						Quadrature	
						Bits	Description
						0	Reserved.
						1	Set on bad differential signal levels on any of the encoder inputs.
						EnDAT (Type 11)	
						Bits	Description

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**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 amp.
						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)	

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td><table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0</td><td>Battery warning reported by encoder.</td></tr>
<tr><td>1</td><td>Battery error reported by encoder.</td></tr>
<tr><td>3</td><td>Over speed reported by encoder.</td></tr>
<tr><td>4</td><td>Memory error reported by encoder.</td></tr>
<tr><td>5</td><td>STERR reported by encoder.</td></tr>
<tr><td>6</td><td>PSERR reported by encoder.</td></tr>
<tr><td>7</td><td>Busy error reported by encoder.</td></tr>
<tr><td>8</td><td>Memory busy reported by encoder.</td></tr>
<tr><td>9</td><td>Over temperature.</td></tr>
<tr><td>15</td><td>CRC error on data received from encoder.</td></tr>
</table></td></tr>
<tr><td>0x12F</td><td>0x130</td><td>0x52F</td><td>0x2225</td><td>R*</td><td>U32</td><td>Load encoder status. Same as parameter 0x12E, but for the load encoder.</td></tr>
<tr><td>0x130</td><td>0x131</td><td>0x530</td><td>0x2114</td><td>RF</td><td>INT16</td><td>RMS current calculation period. This sets the period over which the RMS current is calculated. If this value is set to zero, then the RMS current will be updated each time it is read for the period since the last read. In this case, the RMS current must be read at least once every 65536 current loop periods (about every 4 seconds) for the returned RMS values to be accurate. Units: milliseconds.</td></tr>
<tr><td>0x131</td><td>0x132</td><td>0x531</td><td>0x2115</td><td>R*</td><td>INT16</td><td>RMS current over the period set in parameter 0x130. Units: 0.01 A.</td></tr>
<tr><td>0x132</td><td>0x133</td><td>0x532</td><td>0x2116</td><td>R*</td><td>INT16</td><td>Running sum of user current limit in 0.01% units. (i.e., 0 to 10000).</td></tr>
<tr><td>0x133</td><td>0x134</td><td>0x533</td><td>0x2117</td><td>R*</td><td>INT16</td><td>Running sum of amp current limit in 0.01% units. (i.e., 0 to 10000).</td></tr>
<tr><td>0x134</td><td>0x135</td><td>0x534</td><td>0x21E0</td><td>RF</td><td>U32</td><td>Analog output configuration. This parameter sets the mode for the Analog output on drives so equipped.<table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0-3</td><td>Define the mode of the Analog output.</td></tr>
<tr><td>16-17</td><td>Identify the axis associated with the Analog output.</td></tr>
<tr><td>Mode</td><td>Description</td></tr>
<tr><td>0</td><td>Manual configuration (set using parameter 0x135).</td></tr>
<tr><td>1</td><td>Actual current of configured axis.</td></tr>
<tr><td>2</td><td>Actual velocity of configured axis. The output is a +/- 5V range and is a ratio of the actual velocity to the velocity loop maximum velocity.</td></tr>
</table></td></tr>
<tr><td>0x135</td><td>0x136</td><td>0x535</td><td>0x21E1</td><td>R</td><td>INT16</td><td>Analog output value. For drives that support an Analog output, this parameter sets</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>the output value in mV units when the D/A is in manual mode. In other modes, the current value being output on the D/A can be read here.</td></tr>
<tr><td>0x136</td><td>0x137</td><td>0x536</td><td>0x2208</td><td>R*</td><td>INT16</td><td>Secondary analog reference value. Units: millivolts</td></tr>
<tr><td>0x137</td><td>0x138</td><td>0x537</td><td>0x2314</td><td>RF</td><td>INT16</td><td>Secondary analog reference offset. Offset for secondary analog reference input. Units: millivolts</td></tr>
<tr><td>0x138</td><td>0x139</td><td>0x538</td><td>0x2315</td><td>RF</td><td>INT16</td><td>Calibration offset for second analog reference input. It is factory-calibrated to give a zero reading for zero input voltage. Units: millivolts</td></tr>
<tr><td>0x139</td><td>0x140</td><td>0x539</td><td>0x219D</td><td>R</td><td>INT32</td><td>Status of drive safety circuit. This parameter allows the status of the safety circuit built into some drives to be queried. For drives without a safety circuit, this parameter is reserved.
<table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0</td><td>Set when safety input 0 is preventing the drive from enabling.</td></tr>
<tr><td>1</td><td>Set when safety input 1 is preventing the drive from enabling.</td></tr>
<tr><td>8</td><td>This read/write bit can be used to force the ‘drive is unsafe’ output of the safety circuit to go active for testing purposes. Write 1 to force.</td></tr>
</table></td></tr>
<tr><td>0x13A</td><td>0x13B</td><td>0x53A</td><td>0x2209</td><td>R*</td><td>INT16</td><td>Present voltage at analog motor temperature sensor. Units: millivolts.</td></tr>
<tr><td>0x13B</td><td>0x13C</td><td>0x53B</td><td>0x220A</td><td>RF</td><td>INT16</td><td>Limit for analog motor temperature sensor.
If this parameter is set to zero, then the analog motor temperature sensor is disabled.
If this parameter is set to a positive value, then a motor temperature error will occur any time the voltage on the motor temperature input exceeds this value (in millivolts).
If this parameter is negative, then a motor temperature error will occur any time the voltage on the motor temperature input is lower than the absolute value of this limit in millivolts.</td></tr>
<tr><td>0x13C</td><td>0x13D</td><td>0x53C</td><td>0x2323</td><td>RF</td><td>INT16</td><td>Minimum PWM pulse width in microseconds. Used when running in PWM position mode. In this mode the PWM input pulse width is captured by the drive and used to calculate an absolute position using the following formula:
pos = ((PW-MIN) / (MAX-MIN)) * SCALE + OFFSET
where this parameter is the minimum pulse width (MIN), parameter 0x13D is the maximum pulse width (MAX), parameter 0xA9 is the scaling factor (SCALE) and parameter 0x10F is the offset (OFFSET).</td></tr>
<tr><td>0x13D</td><td>0x13E</td><td>0x53D</td><td>0x2324</td><td>RF</td><td>INT16</td><td>Maximum PWM pulse width used when running in PWM position mode. Units:</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>microseconds.</td></tr>
<tr><td>0x150</td><td>0x151</td><td>0x550</td><td>0x210A</td><td>RF</td><td>14 words</td><td>Second chained biquad filter on output of velocity loop, see Filter Coefficients (p. 73).</td></tr>
<tr><td>0x151</td><td>0x152</td><td>0x551</td><td>0x210B</td><td>RF</td><td>14 words</td><td>Third chained biquad filter on output of velocity loop. 14 word parameter, see Filter Coefficients (p. 73).</td></tr>
<tr><td>0x152</td><td>0x153</td><td>0x552</td><td>0x210C</td><td>RF</td><td>14 words</td><td>First chained biquad filter on input of current loop. 14 word parameter, see Filter Coefficients (p. 73).</td></tr>
<tr><td>0x153</td><td>0x154</td><td>0x553</td><td>0x210D</td><td>RF</td><td>14 words</td><td>Second chained biquad filter on input of current loop. 14 word parameter, see Filter Coefficients (p. 73).</td></tr>
<tr><td>0x154</td><td>0x155</td><td>0x554</td><td>0x2301</td><td>RF</td><td>INT32</td><td>Servo loop configuration. This parameter allows various parts of the drive servo loops to be enabled/disabled. Bit-mapped as follows:
<table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0</td><td>If set, this disables the velocity loop gains. The velocity loop command feed forward gain (parameter 0x157) is still active as are the velocity loop output filters.</td></tr>
<tr><td>1</td><td>If set, this enables the position loop I and D gains (parameters 0x155 and 0x156). If clear, these parameters are treated as zeros.</td></tr>
<tr><td>2-31</td><td>Reserved for future use.</td></tr>
</table></td></tr>
<tr><td>0x155</td><td>0x156</td><td>0x555</td><td>0x2382:5</td><td>RF</td><td>INT16</td><td>Position loop integral gain (KI).</td></tr>
<tr><td>0x156</td><td>0x157</td><td>0x556</td><td>0x2382:6</td><td>RF</td><td>INT16</td><td>Derivative gain for position loop (KD).</td></tr>
<tr><td>0x157</td><td>0x158</td><td>0x557</td><td>0x2381:6</td><td>RF</td><td>INT16</td><td>Velocity loop command feed forward. The input command (after limiting) to the velocity loop is scaled by this value and added in to the output of the velocity loop.</td></tr>
<tr><td>0x158</td><td>0x159</td><td>0x558</td><td>0x2382:7</td><td>RF</td><td>INT16</td><td>Integral drain for position loop.</td></tr>
<tr><td>0x159</td><td>0x15A</td><td>0x559</td><td>0x6007</td><td>RF</td><td>INT16</td><td>Abort option code for CANopen / EtherCAT drives.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0x15A</td><td>0x15B</td><td>0x55A</td><td>0x2198</td><td>RF</td><td>U32</td><td>I/O options. This parameter is used to configure optional features of the general purpose I/O.
<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>0-3</td><td>For AEM/APM, these bits determine whether several I/O pins are used as a serial interface for expanded I/O features, and if so how they are configured.
0 – normal I/O
1 – AEM/APM development board LEDs and address switches.
2 – LEDs wired the same as the developer's kit board, but using separate red & green LEDs for the network status.</td></tr>
<tr><td>4-31</td><td>Reserved for future use.</td></tr>
</table></td></tr>
<tr><td>0x15B</td><td>0x15C</td><td>0x55B</td><td>0x2199</td><td>F</td><td>INT16</td><td>Motor brake enable delay time (milliseconds). This parameter gives a delay between enabling the drive PWM outputs and releasing the brake. Positive values mean the PWM is enabled first and then the brake is released N milliseconds later. Negative values cause the brake to be released before PWM outputs are enabled.</td></tr>
<tr><td>0x15C</td><td>0x15D</td><td>0x55C</td><td>0x219A</td><td>R*</td><td>U32</td><td>32-bit version of parameter 0xA6.</td></tr>
<tr><td>0x15D</td><td>0x15E</td><td>0x55D</td><td>0x219B</td><td>R*</td><td>U32</td><td>32-bit version of parameter 0xAA.</td></tr>
<tr><td>0x15E</td><td>0x15F</td><td>0x55E</td><td>0x219C</td><td>RF</td><td>U32</td><td>32-bit version of parameter 0xA5.</td></tr>
<tr><td rowspan="8">0x160-0x167</td><td rowspan="8">0x161-0x168</td><td rowspan="8">0x560-0x567</td><td>0x2192:17</td><td>RF</td><td>U16</td><td>Input pin config for general purpose input 16. See Input 0 Configuration (0x78)</td></tr>
<tr><td>0x2192:18</td><td>RF</td><td>U16</td><td>Input pin config for general purpose input 17. See Input 0 Configuration (0x78)</td></tr>
<tr><td>0x2192:19</td><td>RF</td><td>U16</td><td>Input pin config for general purpose input 18. See Input 0 Configuration (0x78)</td></tr>
<tr><td>0x2192:20</td><td>RF</td><td>U16</td><td>Input pin config for general purpose input 19. See Input 0 Configuration (0x78)</td></tr>
<tr><td>0x2192:21</td><td>RF</td><td>U16</td><td>Input pin config for general purpose input 20. See Input 0 Configuration (0x78)</td></tr>
<tr><td>0x2192:22</td><td>RF</td><td>U16</td><td>Input pin config for general purpose input 21. See Input 0 Configuration (0x78)</td></tr>
<tr><td>0x2192:23</td><td>RF</td><td>U16</td><td>Input pin config for general purpose input 22. See Input 0 Configuration (0x78)</td></tr>
<tr><td>0x2192:24</td><td>RF</td><td>U16</td><td>Input pin config for general purpose input 23. See Input 0 Configuration (0x78)</td></tr>
<tr><td rowspan="3">0x170-0x177</td><td rowspan="3">0x171-0x178</td><td rowspan="3">0x570-0x577</td><td>0x2195:17</td><td>RF</td><td>U16</td><td>Debounce time for general purpose input 16.</td></tr>
<tr><td>0x2195:18</td><td>RF</td><td>U16</td><td>Debounce time for general purpose input 17.</td></tr>
<tr><td>0x2195:19</td><td>RF</td><td>U16</td><td>Debounce time for general purpose input 18.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td>0x2195:20</td><td>RF</td><td>U16</td><td>Debounce time for general purpose input 19.</td></tr>
<tr><td></td><td></td><td></td><td>0x2195:21</td><td>RF</td><td>U16</td><td>Debounce time for general purpose input 20.</td></tr>
<tr><td></td><td></td><td></td><td>0x2195:22</td><td>RF</td><td>U16</td><td>Debounce time for general purpose input 21.</td></tr>
<tr><td></td><td></td><td></td><td>0x2195:23</td><td>RF</td><td>U16</td><td>Debounce time for general purpose input 22.</td></tr>
<tr><td></td><td></td><td></td><td>0x2195:24</td><td>RF</td><td>U16</td><td>Debounce time for general purpose input 23.</td></tr>
<tr><td>0x180</td><td>0x181</td><td>0x580</td><td>0x2326</td><td>RF</td><td>U32</td><td>UV configuration. This parameter is used to configure the drive when running in UV mode (desired state 5). Bit-mapped as follows (undocumented bits are reserved for future use):
<table>
<tr><td>Bits</td><td>Meaning</td></tr>
<tr><td>0-1</td><td>Define the source of the UV command inputs:
<table>
<tr><td>Value</td><td>Description</td></tr>
<tr><td>0</td><td>PWM inputs.</td></tr>
<tr><td>1</td><td>Analog reference inputs (for drives with two analog reference pins).</td></tr>
<tr><td>2</td><td>Analog encoder inputs.</td></tr>
<tr><td>3</td><td>Directly set over the serial/network interface.</td></tr>
</table></td></tr>
<tr><td>2-7</td><td>Reserved.</td></tr>
<tr><td>8-9</td><td>Define the format of the UV inputs:
<table>
<tr><td>Value</td><td>Description</td></tr>
<tr><td>0</td><td>120 degree current commands.</td></tr>
<tr><td>1</td><td>90 degree current commands.</td></tr>
<tr><td>2</td><td>Angle/Magnitude form. U input gives magnitude, V gives angle.</td></tr>
</table></td></tr>
<tr><td>10-15</td><td>Reserved.</td></tr>
<tr><td>16</td><td>If set, the value of the Motor Hall Offset (0x4F) is added to the UV angle</td></tr>
<tr><td>17</td><td>If set, the drive will use field oriented control. Normally FOC is disabled in UV mode due to the ambiguity of the phase angle with zero inputs. This is best used when running in angle/magnitude format.</td></tr>
</table></td></tr>
<tr><td>0x181</td><td>0x182</td><td>0x581</td><td>0x2327</td><td>R</td><td>INT16</td><td>U input when running in UV mode. This parameter can be used to read the calculated U value, or to set a U value when the UV inputs are being directly set over the serial/network interface.</td></tr>
<tr><td>0x182</td><td>0x183</td><td>0x582</td><td>0x2328</td><td>R</td><td>INT16</td><td>V input when running in UV mode.</td></tr>
<tr><td>0x183</td><td>0x184</td><td>0x583</td><td>0x2329</td><td>R</td><td>INT16</td><td>Raw counter value from pulse & direction input hardware. This can be read when running in any mode, not just pulse & direction modes.
This parameter can be written also, but should not be written when the amp is being driven by the pulse & direction inputs. Writing in that mode will cause the amp to treat the change in the counter as real pulse inputs resulting in possible unexpected motion.</td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0x184</td><td>0x185</td><td>0x584</td><td>0x2254</td><td>RF</td><td>8 to 40 Words</td><td>Input shaping filter. This filter is used to modify the trajectory before its input to the position loop. This can be used to compensate for low frequency resonances in the load.

The parameter is an array of 32-bit values. The first four values are used to store information about the input shaping filter (filter type, frequency, etc.) and are mostly unused by the firmware. The only exception is that the MSB of the 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 the input shaping filter. The time values are specified in seconds and must be >= 0.0. The impulse values are unit-less and must have an absolute magnitude of < 16.0.</td></tr>
<tr><td>0x185</td><td>0x186</td><td>0x585</td><td>0x2160</td><td>R</td><td>U32</td><td>Output compare configuration. For a detailed description of the output compare function, please see the Setting Outputs at Position application note describing it.</td></tr>
<tr><td>0x186</td><td>0x187</td><td>0x586</td><td>0x2161</td><td>R</td><td>U32</td><td>Output compare status.</td></tr>
<tr><td>0x187</td><td>0x188</td><td>0x587</td><td>0x2162</td><td>R</td><td>INT32</td><td>Output compare value 1.</td></tr>
<tr><td>0x188</td><td>0x189</td><td>0x588</td><td>0x2163</td><td>R</td><td>INT32</td><td>Output compare value 2.</td></tr>
<tr><td>0x189</td><td>0x18A</td><td>0x589</td><td>0x2164</td><td>R</td><td>INT32</td><td>Output compare increment.</td></tr>
<tr><td>0x18A</td><td>0x18B</td><td>0x58A</td><td>0x2165</td><td>R</td><td>INT32</td><td>Output compare pulse width.</td></tr>
<tr><td>0x18B</td><td>0x18C</td><td>0x58B</td><td>0x2255</td><td>RF</td><td>INT32</td><td>Trajectory options. This parameter is used to modify the behavior of some trajectory modes. Its interpretation depends on the trajectory mode being used. The following trajectory modes currently make use of this parameter:
<table>
<tr><td colspan="2">EtherCAT CSP mode:</td></tr>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0-7</td><td>Number of extra loop cycles to extrapolate trajectories if input data from the master is not received.</td></tr>
<tr><td>8-15</td><td>Reserved.</td></tr>
<tr><td>16</td><td>If set, jump to quick stop mode if master data is not received within the number of cycles set in bits 0-7.</td></tr>
<tr><td>17</td><td>If set, and the interpolation time object (0x60C2) is non-zero, then the calculated velocity will be filtered, and a trajectory acceleration will also be calculated. If not set, velocity is unfiltered and acceleration is not calculated (zero).</td></tr>
<tr><td>18-31</td><td>Reserved.</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>ASCII</th><th>DvcNet</th><th>MACRO</th><th>CAN/ECAT IDX:SUB</th><th>Bank</th><th>Type</th><th>Description</th></tr>
<tr><td>0x18C</td><td>0x18D</td><td>0x58C</td><td>0x21A1</td><td>RF</td><td>U32</td><td>I/O extension configuration for modules. This parameter is used to configure the I/O extension feature on plus modules which support it.
For a detailed description of this I/O extension feature, see the *Extending Plus Module I/O* application note.
<table>
<tr><td>Bits</td><td>Description</td></tr>
<tr><td>0-4</td><td>Number of bits to transfer less 1 (e.g., set to 19 to transfer 20 bits).</td></tr>
<tr><td>5-7</td><td>Reserved.</td></tr>
<tr><td>8</td><td>Set to enable SPI I/O extension feature. Clear for LED/Switch interface.</td></tr>
<tr><td>9</td><td>Automatically restart transmission if set.</td></tr>
<tr><td>10</td><td>Leave the CS line low after transfer if set.</td></tr>
<tr><td>11</td><td>Reserved.</td></tr>
<tr><td>12</td><td>Clock polarity setting.</td></tr>
<tr><td>13</td><td>Data phase setting.</td></tr>
<tr><td>14-15</td><td>Reserved.</td></tr>
<tr><td>16-23</td><td>Clock period (100ns units).</td></tr>
<tr><td>24-31</td><td>Reserved.</td></tr>
</table></td></tr>
<tr><td>0x18D</td><td>0x18E</td><td>0x58D</td><td>0x21A2</td><td>R</td><td>INT32*</td><td>I/O extension transmit data. Data to be transferred over the SPI port is sent immediately after being written to this parameter.

*Data size is variable dependent on drive configuration. Refer to the Extending Plus Module I/O application note.</td></tr>
<tr><td>0x18E</td><td>0x18F</td><td>0x58E</td><td>0x21A3</td><td>R</td><td>INT32*</td><td>I/O extension receive data. Data received from the SPI port can be read from this parameter.

*Data size is variable dependent on drive configuration. Refer to the Extending Plus Module I/O application note.</td></tr>
<tr><td>0x18F</td><td>0x190</td><td>0x58F</td><td>0x220B</td><td>RF</td><td>INT16</td><td>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 the encoder sine signal before calculating position.</td></tr>
<tr><td>0x190</td><td>0x191</td><td>0x590</td><td>0x220C</td><td>RF</td><td>INT16</td><td>Encoder cosine offset. Like 0x18F, but for encoder cosine signal.</td></tr>
<tr><td>0x191</td><td>0x192</td><td>0x591</td><td>0x220D</td><td>RF</td><td>U16</td><td>Encoder cosine scaling factor. Used by the resolver & servo tube encoder calculations. This scaling factor is used to adjust the cosine signal amplitude so that it's the same as the sine signal amplitude.
If set to zero, both the scaling and offsets (0x18F,0x190) will be ignored. If non-zero the cosine is scaled by N/32768 where N is the value of this parameter.</td></tr>
</table>

ASCII	**DvcNet**	**MACRO**	**CAN/ECAT IDX:SUB**	**Bank**	**Type**	**Description**
0x192	0x193	0x592	0x2226	RF	U32	Motor encoder calibration settings. The meaning of this parameter is dependent on the encoder type.
0x193	0x194	0x593	0x2227	RF	U32	Load encoder calibration settings. Same as 0x192, but applied to the load encoder.
0x194	0x195	0x594	0x232A	R*	INT16	PWM input duty cycle. This parameter can be used to read the duty cycle of the PWM input. The returned 16-bit value gives the duty cycle in the range +/-32k. Parameter 0xA8 is used to configure the PWM input.
0x195	0x196	0x595	0x2123	RF	INT32	Jerk value to use during trajectory aborts. If this is zero, then the abort will be calculated without any jerk limits. Units are 100 counts / s^3.
0x196	0x197	0x596	0x220E	R*	INT32	Returns the magnitude squared of the analog encoder signals (sin*sin + cos*cos)
0x197	0x198	0x597	0x2378	RF	INT16	Cross coupling KP gain. On dual axis drives this gain is applied to the difference in position error of the two axes.
0x198	0x199	0x598	0x2379	RF	INT16	Cross coupling KI gain.
0x199	0x19A	0x599	0x237A	RF	INT16	Cross coupling KD gain.
0x19A	0x19B	0x59A	0x220F	RF	5 words	Steinhart constants for motor analog motor temperature sensor.
0x19B	0x19C	0x59B	0x2384:30	F*	INT16	Current at which minimum PWM deadtime is used.
0x19C	0x19D	0x59C	0x2406	R*	INT32	High speed capture for passive load encoder.
0x19D	0x19E	0x59D	0x2142	RF	INT16	Open motor wiring check current. If parameter 0x15B is greater than zero, then during that time period on enable this current will be applied to the motor wiring to check that the motor is connected. If the programmed current cannot be applied to the motor, then a motor disconnected fault will be flagged.
0x19E	0x19F	0x59E	0x6066	RF	U16	Position error timeout. The time that a position error must persist before it is triggered. Units: ms
0x1A0	0x1A1	0x5A0	0x2193:9	RF	See text	Output 9 Configuration. See Output 0 Configuraton (0x70).
0x1A1	0x1A2	0x5A1	0x2193:10	RF	See text	Output 10 Configuration. See Output 0 Configuraton (0x70).
0x1A2	0x1A3	0x5A2	0x2193:11	RF	See text	Output 11 Configuration. See Output 0 Configuraton (0x70).
0x1A3	0x1A4	0x5A3	0x2193:12	RF	See text	Output 12 Configuration. See Output 0 Configuraton (0x70).

ASCII	DvcNet	MACRO	CAN/ECAT IDX:SUB	Bank	Type	Description
0x1A8	0x1A9	0x5A8	0x2228	RF	INT16	Motor encoder down-shift. This parameter is useful when using very high resolution encoders that would otherwise have limited speed and travel distance due to the range of position and velocity parameters. Setting the down-shift causes the position read from the encoder to be right-shifted before being used. For example, setting this parameter to a value of 2 effectively cuts the encoder resolution by a factor of 4. When this parameter is set, the servo loops use fractional encoder counts, therefore the encoder resolution is not completely lost.
0x1A9	0x1AA	0x5A9	0x2229	RF	INT16	Load encoder down-shift. Same as parameter 0x1A8, but for the load encoder.

CHAPTER

3: FILTER COEFFICIENTS

There are several drive parameters which are used to define filters. These filters are implemented as generic biquadratic 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):

$$y(n) = b0 \cdot x(n) + b1 \cdot x(n-1) + b2 \cdot x(n-2) + a1 \cdot y(n-1) + a2 \cdot y(n-2)$$

The values a1, a2, b0, b1, b2 are constants known as filter coefficients. They define the type of filter being implemented.

The values passed to these drive filter parameters are used to define the filter coefficients. The formatting of these parameters varies depending on the drive 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 the resolution of these coefficients, an additional unsigned scaling coefficient (k) is also specified. The actual filter formula used within these drives is as follows:

$$y(n) = K \cdot (b0 \cdot x(n) + b1 \cdot x(n-1) + b2 \cdot x(n-2) + a1 \cdot y(n-1) + a2 \cdot y(n-2)) / 32768 / 4096$$

To set the 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 the CME software to describe the filter. If the upper 3 bits of the first word are all set then the filter will be disabled. Other than that, 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 the filter. Otherwise, reserved for CME use.
2	Filter info. Reserved for CME use.
3	Filter info. Reserved for CME use.
4	b2 coefficient.
5	b1 coefficient.
6	b0 coefficient.
7	a2 coefficient.
8	a1 coefficient.
9	K scaler.

For the plus family of drives (Xenus+, AEM, etc), a new format is used to describe the biquad filter coefficients. These drives include the ability to design the filters in firmware using the Cephes filter design library.

Filters on this family of drives are calculated internally using 32-bit IEEE floating point coefficients. The format of the parameter information passed when setting filter parameters on these drives consists of an array of up to fourteen 16-bit words. The first 4 words describe the filter, and the remaining 10 words give the filter coefficients as 32-bit IEEE floating point values. The filter coefficient words are optional and are only necessary if the firmware is not calculating the 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 the filter. This prevents the 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. The filter family should be one of the following values:	
	0	Custom biquad filter. Coefficients must be passed; the firmware will not design the filter.
	1	Butterworth filter.
	2	Chebychev filter.
	3	Elliptic filter.
	4-15	Reserved.
	The filter type should be one of the following:	
	0	Custom biquad filter. Coefficients must be passed; the firmware will not design the 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 the system.
	If legal values are passed for the filter type and family, the firmware will attempt to design the specified filter and fill in the coefficient values itself. The firmware is capable of calculating 1 or 2 pole low pass or high pass filters. For notch and band	

Word	Description	
	pass filters the firmware can only calculate a two pole filter. For these filter types, bit 8 must be set.	
2	This word gives the cut off frequency (in Hz) for low pass and high pass filters. For notch and band pass filters this gives the first filter frequency.	
3	This word gives the second filter frequency (in Hz) for notch and band pass filters.	
4	Bits	Usage
	0-7	Rp given in 0.1 dB units.
	8-15	Rs given in dB units.
	Rp is the pass band ripple. This parameter is only used for Chebychev and Elliptic filters. Rs is only used with elliptic filters. It defines the stop band as Rs dB down from the peak value in the 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 the firmware designs the filter then the coefficients will be filled in by the firmware and need not be passed.	
7-8	Coefficient a2.	
9-10	Coefficient b0.	
11-12	Coefficient b1.	
13-14	Coefficient b2.	

Copley Controls Parameter Dictionary

P/N 16-01091

Revision 01

September 2014

Copley Controls
20 Dan Road
Canton, MA 02021 USA