Allied Motion Technologies, Inc.
495 Commerce Drive, Suite 3
Amherst, NY 14228
(716) 242-7535

Software Manual:

Parameters and Control Structure

Document Part Number: 34-2202 R3

Table of Contents

1.0 Tables

2.0 Figures

3.0 Preface

This manual describes the software parameters and control structure related to servo drives, controllers, and motor/drive combinations manufactured by Allied Motion Technologies, Inc.

Every effort has been made to ensure the accuracy of information in this manual. However, Allied Motion assumes no responsibility for any errors or omissions. The information contained within this document is regularly reviewed and we welcome your critical evaluation and suggestions for improvement. Allied Motion reserves the right to modify all documentation without prior notice. Be sure to download the most recent revision of this manual from the company website.

Table 1: Applicable Allied Motion Products

PRODUCT FAMILY	PART NUMBER	DESCRIPTION
EnduraMax iDrive	EMI-065xxxxx, EMI-075xxxxx, EMI-095xxxxx	65mm, 75mm, 95mm EnduraMax motors with integrated drive and encoder
Housed Megaflux motor with drive	MFH-xxxxxx	110mm, 140mm, 170mm Megaflux motors with integrated drive and high resolution optical encoder
AC xDrive	XDA-xxxxxxxx	AC powered brushless servo drive
SX Drives	SXD-xxxxxxxx	DC powered brush-brushless servo drive
SX Controllers	SXC-xxxxxxxx	DC powered brush-brushless single axis servo controller
MX Controllers	MXC-xxxxxxxx	DC powered brush-brushless multiple axis servo controller

4.0 Safety

CAUTION: GENERAL WARNING
Read all provided documentation before assembly and commissioning. Failure or incorrect or improper use of this equipment can cause property damage, personal injury, and death. The final responsibility for the safe use of Allied Motion products is solely that of the user.

CAUTION: HIGH VOLTAGE
When drives or motors are powered, high voltages are present on the connector and elsewhere in the drive and can be dangerous. The drive cover should not be removed with the power on, and then only by a competent and trained engineer. It is important that all system components be properly grounded.

5.0 Overview

Many Allied Motion controllers, drives, and motor/drive combinations are fully digital and so include a series of software parameters (read/write access) and variables (read only) which can be set and/or observed for specific motion applications. This manual describes these parameters/variables and their use.

The IN Control user interface is designed to access these parameters/variables and to assist the user in the setup and operation of Allied Motion devices. See “Document No. 34-2200 Software Manual: IN Control User Guide” for installation and use. Some software parameters/variables can also be accessed via network communications (e.g. CANopen, Modbus RTU). Note that accessible software parameters/variables are assigned object identifiers while others are not accessible and can only be accessed via RS-232 using IN Control or the ALLNET .NET Framework. See “Attachment A, Sortable Parameters and Variables List.xlsx” of this document for a complete listing.

The software parameters/variables are divided into ten functional categories: Motor, Commutation, Current Loop, Velocity Loop, Position Loop, Auxiliary PID Loop, I/O, Command, Configuration, and Status. For successful operation, be sure to follow the proceeding sections of this manual in order and per the instructions set the given parameters to their appropriate values. Note that depending on your specific hardware and/or application, not all parameters/variables will be available or used.

6.0 Motor

This category includes parameters related to the physical properties of the motor. Accurate parameters are essential for subsequent control loop tuning.

Protection for the motor can be applied by setting appropriate values for the locked rotor current limit **LRIL** and motor continuous rated current **IMAX**.

6.1 IMAX and I^2T Current Limiting

Current to the motor is limited using an I^2T current limiting scheme. This scheme allows for motor currents of up to the peak current rating of the drive for a two second interval, before ramping down to the continuous **IMAX** value thereafter. The shape of the fold-back curve is as follows, shown with an arbitrarily chosen scaling of 200% of the continuous **IMAX** level, then folding back to 100% with time (in seconds):

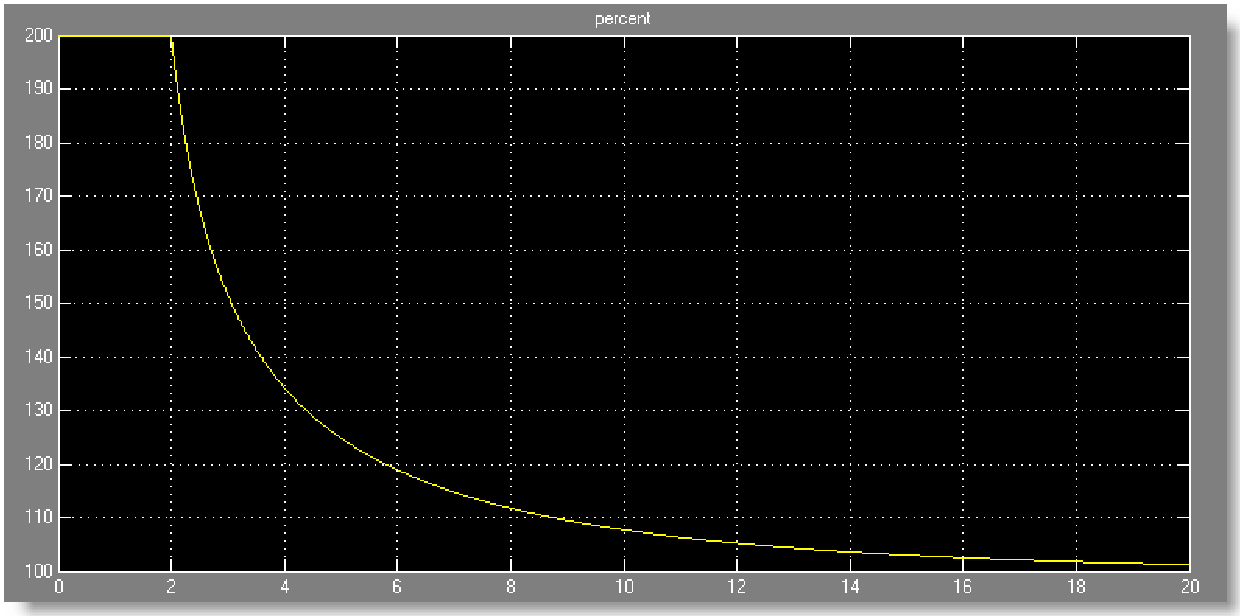

Figure 1: I^2T Current Limiting

Table 2: Motor Parameters

NAME	UNITS	DESCRIPTION	HOW TO USE
IMAX	A	Motor continuous rated current	Motor continuous rated current is defined in ADC when trapezoidally commutating (six-step commutation) or as the peak of the sine wave when commutating sinusoidally. The lesser of IMAX and the drive's continuous rated current are used with the drive's I^2*T current limiting scheme to limit the average current in the motor and drive. The output of the I^2*T limiting scheme is the current limit parameter ILIM.
INER	kg-m^2	Total motor + load inertia	This parameter is used to tune the velocity control loop when the advanced tuning switch is on VLAT=ON.
KT	N-m/A	Motor torque constant	Used to scale the torque command TRQC to the torque producing motor current reference IQREF.
LM	mH	Terminal motor inductance	Determined by measuring the inductance between any two of the motor leads with the third lead open. LM is used in the automatic calculation of current loop gains using the "Calc I-Loop Gains" function of IN Control.
LRIL	A	Locked rotor current limit	A locked rotor is defined as when the motor is stalled and the time averaged torque producing current reference IQAVG is greater than LRIL. When this condition persists for greater than 4 seconds, a locked rotor fault is produced. To disable all Locked Rotor faults, set LRIL=0.
MPOL	count	Motor pole count	Determines the number of electrical cycles per mechanical cycle. The number of electrical cycles per motor revolution is equal to the number of motor pole pairs (i.e. half the motor pole count).
RM	ohm	Terminal motor resistance	Determined by measuring the resistance between any two of the motor leads with the third lead open. RM is used in the automatic calculation of current loop gains using the Calc I-Loop Gains function of IN Control.

7.0 Commutation

For a permanent magnet motor, commutation refers to the process of switching motor currents so as to maintain the optimum angle between the permanent magnetic field and the electromagnetic field induced in the armature so as to generate maximum torque. For brushed-type motors, this process is performed mechanically using brushes and a physical commutator. For brushless-type motors, this can be performed electronically using a set of transistors and a microprocessor, which are the essential components in any modern servo drive.

For a servo drive to commutate a brushless motor effectively, the actual motor position must be known. Knowing the position, the drive then applies the correct voltages to the motor phases so that the desired motor torque output is produced. In Allied Motion drive products, calculation of these voltages is accomplished using a Field Oriented Control algorithm (refer to industry literature for explanation).

There are several kinds of devices available to measure motor position including halls sensors, encoders, and resolvers. The resolution of these devices will influence the quality of the currents produced in the motor. In other words, commutating using encoder plus hall sensor feedback will result in a smoother running system as compared to a motor commutating with hall sensor feedback alone.

With these considerations, use the commutation parameters to configure the drive to operate with the motor position feedback device available. Settings include drive operating mode (select feedback method), encoder resolution, encoder type, commutation offset, and feedback inversion bits. For Allied Motion motors properly wired to Allied Motion drives, the inversion bits and commutation offset should all be set to zero; otherwise use the “Auto Configure” tool to configure.

Table 3: Commutation Parameters

NAME	UNITS	DESCRIPTION	HOW TO USE
C1.INVE		Invert primary encoder direction	Bit value reverses the direction of the hardware counter for the primary feedback source FBC. When primary feedback is an incremental encoder plus hall sensors (or commutation track), for proper motor commutation FBC and the hall commutation angle HANG must proceed in the same direction. To confirm this is so, set drive operating mode to hall only DM=BLH, and confirm that FBC and HANG proceed in the same direction. If this is not the case, use the invert bit to make it so. C1.INVE is ignored when in hall operating mode DM=BLH. Be sure to set the inversion bits in the following order: commutation offset COFF and C1.INVH, then C1.ITRQ, C1.INVE, C1.INVF.
C1.INVF		Invert velocity/position feedback direction	Bit value reverses the direction of the velocity and position feedback parameters, VEL and FPOS respectively. For proper motor control when in velocity or position command modes, CM=VEL or CM=POS, a positive torque producing current reference IQREF must correspond to a positive change (acceleration) in the velocity feedback VEL. If this is not the case, use invert bit to make it so. C1.INVF is ignored when in hall operating mode DM=BLH. Be sure to set the inversion bits in the following order: commutation offset COFF and C1.INVH, then C1.ITRQ, C1.INVE, C1.INVF.
C1.INVH		Invert hall commutation angle direction	Bit value which reverses the direction of the hall commutation angle HANG. When in hall operating mode DM=BLH and current command mode CM=CUR, the no-load speed of the motor should be equivalent in both directions given a positive and negative current command of the same magnitude, and the motor feedback velocity VEL and the torque producing current command reference IQREF should always be of the same sign. If this is not the case, adjust COFF

NAME	UNITS	DESCRIPTION	HOW TO USE
			and C1.INVH to make it so. Be sure to set the inversion bits in the following order: commutation offset COFF and C1.INVH, then C1.ITRQ, C1.INVE, C1.INVF.
C1.ITRQ		Invert current command direction	Bit value which reverses the direction of the torque producing current reference IQREF. Use this bit to select chosen motor direction. Be sure to set the inversion bits in the following order: commutation offset COFF and C1.INVH, then C1.ITRQ, C1.INVE, C1.INVF.
COFF	deg	Commutation offset	Define the offset between the hall commutation angle HANG and the actual motor electrical angle. For most motors manufactured by Allied Motion, COFF should be zero; successful commutation of an Allied Motion motor using an offset angle which is some multiple of 120 degrees likely indicates that the hall sensor wiring was made out of order. For all motors, inaccurate values of COFF will result in decreased motor efficiency and/or problems controlling the motor. When in hall operating mode DM=BLH and current command mode CM=CUR, the no-load speed of the motor should be equivalent in both directions given a positive and negative current command of the same magnitude, and the motor feedback velocity VEL and the torque producing current command reference IQREF should always be of the same sign. If this is not the case, adjust COFF and C1.INVH to make it so.
DM		Drive operating mode	Select feedback method for motor commutation: DM=0 -> VF In variable frequency mode, the drive outputs a fixed-frequency, constant-amplitude, sinusoidal current to the motor. The amplitude is defined by the command value CV, and its frequency is defined by the parameter VFF. The motor current is independent of the motor position. This mode is used for testing and is almost never used in actual applications. DM=1 -> BLH In brushless hall mode, the motor is commutated using hall sensor position feedback only. If utilizing a motor with hall only feedback, an encoder is not required. Valid command modes are limited to current CM=CUR, torque CM=TRQ, or velocity CM=VEL. Note that slow speed operation (e.g. under 10 rev/s) will result in poor speed control due to the inadequate resolution provided by the hall sensors at low speeds. DM=2 -> BLE In brushless encoder mode, the motor is commutated using both hall sensor information and encoder feedback. All command modes are available. DM=3 -> BLS In brushless sensorless mode, the motor is commutated using position information estimated from the backEMF voltage of the motor as it rotates. Because the strength of the backEMF signal depends on the motor velocity, there is a lower bound on motor velocity below which the motor will not commutate. Valid command modes are limited to current CM=CUR, torque CM=TRQ, or velocity CM=VEL. DM=4 -> BLR In brushless resolver mode, the motor is commutated using resolver position feedback. All command modes are available. DM=5 -> BSC In brushless sin/cos mode, the motor is commutated using both hall sensor

NAME	UNITS	DESCRIPTION	HOW TO USE
			information and high resolution position feedback from a sin/cos encoder. All command modes are available. DM=6 -> BRU In brushed mode, the drive assumes the presence of a mechanical commutator. Phase C is deactivated and Phase A and B are utilized to control single-phase motor current. All command modes are available. DM=7 -> STEP Reserved. DM=8 -> BLE_NH In brushless encoder no halls mode, the motor is commutated using encoder feedback only. Note that on the first drive enable after power-up, the motor will move slightly to establish alignment between the encoder feedback and motor electrical angle, after which operation proceeds as normal. All command modes are available.
EPPR	count/rev	Encoder counts per revolution	Enter the resolution (number of observable positions per motor revolution) of the primary encoder.
PENCT		Primary encoder type	Declare output format of primary encoder (e.g. quadrature, sin/cos, Endat, Biss).
RPOL		Resolver pole count	Can be an even number up to 16. Parameter is required to be set if DM=BLR.
INTBITS		Sin/cos encoder interpolation degree	Feedback from a sin/cos encoder is processed through an analog to digital converter. The number of possible values output from the converter is defined as 2^INTBITS. The range of INTBITS is 0 to 12. Therefore, setting this parameter to zero indicates no interpolation and setting INTBITS to 12 results in the highest level of interpolation (4096 points). The total position feedback resolution is thus calculated as follows: multiply the number of encoder lines by four, in order to account for quadrature, to obtain the number of encoder counts per revolution EPPR. EPPR is then multiplied by the number of interpolated points (2^INTBITS) to obtain the total number of counts per revolution.
SENCT		Secondary encoder type	Declare output format of secondary encoder (e.g. quadrature, sin/cos, Endat, Biss).
TENCT		Tertiary encoder type	Declare output format of tertiary encoder (e.g. quadrature, sin/cos, Endat, Biss).
VFF	Hz	Open-loop commutation frequency	This is only used when DM=VF. This parameter is intentionally volatile (VFF=0 at power-up).

8.0 Current Loop

Depending on the chosen command mode, there may be up to three loops to tune in the drive: current, velocity, and position. The current loop must always be tuned. The easiest way to tune the current loop is first to have entered the motor resistance **RM** and inductance **LM**; if these parameters are accurate, the “Calc I-Loop Gains” function can be used to automatically calculate current loop gains.

Care should be taken if tuning the current loop manually, especially when attempting to increase the current loop bandwidth. Instability of the current loop can result if the current loop gains are increased too far. This will become apparent when listening to the motor once enabled; motor noise will increase as the current loop gains are increased. Eventually if the gain is too high, the loop will become completely unstable and an over-current fault will occur.

Over-current faults can also be triggered if the motor inductance is too low; see product datasheet for compatible minimum motor inductance values.

Table 4: Current Loop Parameters

NAME	UNITS	DESCRIPTION	HOW TO USE
CALIFA		Phase A current feedback calibration	Reserved.
CALIFB		Phase B current feedback calibration	Reserved.
CALIFC		Phase C current feedback calibration	Reserved.
CKID	ohm/s	Flux producing current integral gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.
CKIQ	ohm/s	Torque producing current integral gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.
CKPD	ohm	Flux producing current proportional gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.
CKPQ	ohm	Torque producing current proportional gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.
IKID	ohm/s	Flux producing current integral gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.
IKIQ	ohm/s	Torque producing current integral gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.
IKPD	ohm	Flux producing current proportional gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.
IKPQ	ohm	Torque producing current proportional gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.

9.0 Velocity Loop

If the drive is in velocity or position command mode **CM=VEL** or **CM=POS**, the velocity loop must be tuned. There are two modes for tuning the velocity loop: using the advanced tuning switch **VLAT=ON**, or setting the velocity loop gains manually **VLAT=OFF**.

With **VLAT=ON**, the drive software automatically calculates the velocity loop gains based on the parameter values **INER**, **WN**, and **ZETA**. Since the total system inertia **INER** is a factor in this calculation, an accurate value of **INER** is critical for ensuring smooth control of velocity. For the vast majority of applications, the **VLAT** function is effective in setting appropriate velocity loop gains.

With **VLAT=OFF**, velocity loop parameters must be set manually. As a conventional PID loop, the velocity feedback loop gains can be tuned by standard methods such as Ziegler-Nichols.

After setting the velocity loop gains, the system response can be observed while driving the motor with a square wave input from the drive's function generator (see Function Generator Command for how to use). Compare the system response to the command value by utilizing the "Capture" function under the "Analyze" tab of IN Control, and adjust velocity loop parameters as necessary for acceptable rise time, overshoot, settling time, and ringing.

A series of standard filters (Butterworth, Bessel, Chebyshev, and Linear Phase) are available to eliminate unwanted frequencies from either/both the velocity loop command input or the torque command output (refer to industry literature for explanation of each filter type).

Table 5: Velocity Loop Parameters

NAME	UNITS	DESCRIPTION	HOW TO USE
KA		Torque filter constant	In certain drive models there exists a single-pole filter on the output of the velocity control loop. This filter is directly influenced by KA. KA is exponentially weighted and small changes in KA result in large changes in the filter. A value of 1.0 means the filter is bypassed, whereas values such as 0.99 and 0.95 bring in a marked response to the torque command TRQC output of the velocity loop. In general, leaving KA at 1.0 (the default value) is desired. Dropping KA by small values in highly dynamic systems may improve controllability.
KB	N-m-s/rev	Pseudo-friction constant	Adds artificial viscous damping (pseudo-friction) to the velocity loop control system. Friction in a real physical system is usually undesired as it increases load, but it also adds stability to a control loop. By using KB, pseudo-friction can be added to the system without increase in load. Just as its real-world counterpart, KB adds a negative torque load proportional to speed and thus is scaled in N-m-s/rev.
KD	N-m-s^2/rev	Velocity loop derivative gain	It is the derivative gain used in classic PID to provide damping to the system. It is rarely used in PDF and is not included in most PDF-based control loops. It is available in the velocity loop, but it is useful only when KVF is at or near 1.0 (PID control mode).
KF	N-m-s/rev	Velocity loop feed-forward gain	The input velocity command VCMD is multiplied by this term and summed into the output torque command TRQC. For systems with high static-friction, adding a small amount of KF can improve response to step changes in velocity command. In applications where the velocity command is ramped, or in position modes, KF is rarely useful.
KI	N-m/rev	Velocity loop integral gain	With the advanced tuning switched activated VLAT=ON, this parameter is ignored. With VLAT=OFF, this parameter controls the integral gain with PID or PDF.
KV	N-m-s/rev	Velocity loop proportional gain	With the advanced tuning switched activated VLAT=ON, this parameter is ignored. With VLAT=OFF, this parameter controls the integral gain with PID or PDF.
KVF		Velocity loop PID/PDF control ratio	Controls the PID/PDF control-loop ratio. A value of 1.0 selects a classic PID and a value of 0.0 selects PDF. Values in between 1.0 and 0.0 select a combination of classic PID and/or PDF control.
TLIM	N-m	Velocity loop torque command limit	Limits the output of the velocity control loop at a given absolute torque value. The motor torque constant KT must be set accurately for TLIM to function properly.
TRQC.CUTOFF	Hz	Torque command filter cutoff frequency	Define the cutoff frequency for the torque command filter.

NAME	UNITS	DESCRIPTION	HOW TO USE
TRQC.METHOD		Torque command filter method	Define torque command filter type: TRQC.METHOD=BUTTERWORTH Implements Butterworth filter method. TRQC.METHOD=BESSEL Implements Bessel filter method. TRQC.METHOD=CHEBYSHEV_1DB Implements Chebyshev filter method with 1 dB of passband ripple. TRQC.METHOD=CHEBYSHEV_3DB Implements Chebyshev filter method with 3 dB of passband ripple. TRQC.METHOD=LINEAR_PHASE Implements a special form of the Butterworth filter which is linear in phase as much as possible.
TRQC.TYPE		Torque command filter type	Define torque command filter type: TRQC.TYPE=BYPASS Filter is not active. TRQC.TYPE=LPF Implements low-pass filter type TRQC.TYPE=NOTCH Implements notch filter type
VEL.CUTOFF	Hz	Velocity feedback filter cutoff frequency	Define the cutoff frequency for the velocity feedback filter.
VEL.METHOD		Velocity feedback filter method	Define velocity feedback filter type: VEL.METHOD=BUTTERWORTH Implements Butterworth filter method. VEL.METHOD=BESSEL Implements Bessel filter method. VEL.METHOD=CHEBYSHEV_1DB Implements Chebyshev filter method with 1 dB of passband ripple. VEL.METHOD=CHEBYSHEV_3DB Implements Chebyshev filter method with 3 dB of passband ripple. VEL.METHOD=LINEAR_PHASE Implements a special form of the Butterworth filter which is linear in phase as much as possible.
VEL.TYPE		Velocity feedback filter type	Define velocity feedback filter type: VEL.TYPE=BYPASS Filter is not active. VEL.TYPE=LPF Implements low-pass filter type

NAME	UNITS	DESCRIPTION	HOW TO USE
			VEL.TYPE=NOTCH Implements notch filter type
VLAT		Velocity loop advanced tuning switch	There are two modes for tuning the velocity loop: VLAT=ON With VLAT=ON, the parameters KV and KI are automatically calculated from the system inertia INER, motor torque constant KT, target bandwidth WN, and damping ratio ZETA through a proprietary means. In general, tuning with VLAT=ON is recommended as there is a complex interaction among all the velocity loop tuning parameters. With VLAT=ON, increasing WN increases the static and dynamic response of the velocity loop. Too high a value of WN will result in a noisy and unstable system. With VLAT=ON, increasing ZETA reduces over- and under-shoot when the drive is commanded with step-changes in velocity. A ZETA value equal to 1.0 will result in a critically-damped system with little or no overshoot depending on load and damping. VLAT=OFF With VLAT=OFF, the parameters KV and KI are not automatically calculated. The user is free to enter values for these parameters to tune the system. With VLAT=OFF and KVF=1.0, the velocity loop is a conventional PID loop with the following parameters: KV controls the proportional term, KI controls the integral term, and KD controls the derivative term. As a conventional PID loop, the velocity control loop can be tuned by standard methods such as Ziegler-Nichols.
WN	Hz	Velocity loop target bandwidth	Controls the desired target bandwidth of the velocity loop with VLAT=ON. When VLAT=ON, WN automatically calculates values for KV and KI based upon the inertia parameter INER and motor torque constant KT. It is therefore critical to properly calculate and enter the parameters INER and KT before using the velocity loop with VLAT=ON. A value of 100 Hz is usually acceptable in moderate-to-low inertia systems. In high-inertia systems, the value may need to be decreased for stability. In low-inertia systems the value may be increased for better dynamic and static response.
ZETA		Velocity loop damping ratio	Used to calculate values for KV and KI in conjunction with the parameter WN. In systems that ring (oscillate) with step changes in velocity command, increasing ZETA will dampen the system response at the expense of settling time. In systems where some overshoot can be tolerated, or where greater dynamic response is desired, ZETA may be reduced.

10.0 Position Loop

If the drive is in position command mode **CM=POS**, the position loop must be tuned. The proportional position loop uses the output of the drive's target generator **TPOS** to generate a command value for the velocity control loop. The feedback position **FPOS** is subtracted from the target position to produce a position error **PERR**. This position error is multiplied by **PPG** to produce one of the two terms that are summed to become the velocity command. As one can deduce, increasing **PPG** will increase "stiffness" and decrease the transient position loop error as a result.

The second term that is summed to produce the velocity command is the product of the time derivative of the target position and **PFF**. This feed-forward term provides a larger and instantaneous velocity command to the velocity control loop even when only a small position error is present. Adding **PFF** is useful in limited scenarios such as pick-and-place positioning machines where highly dynamic position response is desired for near-step changes in position command. In positioning systems where there is smooth acceleration and deceleration in the position profile, **PFF** is not useful and may decrease the performance and stability of the system.

Table 6: Position Loop Parameters

NAME	UNITS	DESCRIPTION	HOW TO USE
INPOSW	rev	In-position window	When the absolute value of the position error PERR is less than INPOSW at the end of a move in a motion sequence, the move is considered complete. INPOSW is ignored when not in position command mode CM=POS.
P0 to P9		Position loop register x	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.
PFF		Position loop feed-forward gain	The second term in a summation which becomes the velocity command VCMD is the product of the time derivative of the target position TPOS and PFF. This feed-forward term provides a larger and instantaneous velocity command to the velocity loop even with a small position error. PFF is unit-less and ranges from 0 to 1.
PPG	1000/min	Position loop proportional gain	PPG is a classic CNC-style tuning gain that is scaled in 1000/min. PPG is multiplied by the position error PERR to produce the first term in a summation which becomes the velocity loop command VCMD; the second term in this summation is the feed-forward term (see PFF). With a value of PPG=3.0 (1000/min) and PERR=0.1 (rev), the proportional term in the velocity loop command would be 300 rev/min or VCMD=5 (rev/s).

11.0 Auxiliary PID Loop

An auxiliary PID loop is available for process control. This function works to drive to zero the error measured between a given process setpoint and its feedback control variable.

The process setpoint is produced via either/both an analog signal and a digital setpoint parameter. Depending on your specific hardware, some Allied Motion drive products include a single-ended analog input compatible with many 4-20 mA output transducers. Using this as the process control feedback signal, the auxiliary PID loop can be effective in controlling variables such as pressure, temperature, air flow, etc.

To operate, the drive must be configured for velocity command mode **CM=VEL** with the command variable control switch set to digital command **CVCS=DCV**. The output of the auxiliary PID loop is then added to the digital command value **DCV** to become the total motor velocity command.

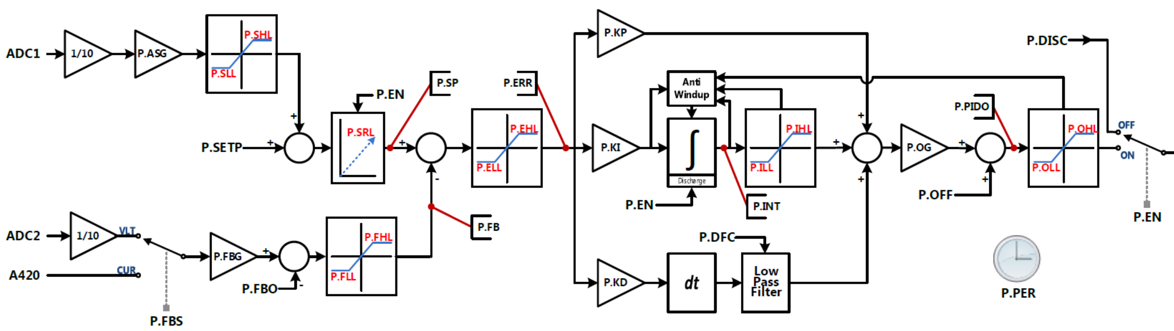

Figure 2: Auxiliary PID Control Loop Flow Chart

Table 7: Auxiliary PID Control Loop Parameters

NAME	UNITS	DESCRIPTION	HOW TO USE
P.ASG		Analog setpoint gain	The analog input signal ADC1 (-10 V to +10 V) is multiplied by 1/10 so that the resultant range is -1.0 to 1.0. P.ASG scales this signal to user defined units.
P.DFC	Hz	Derivative low-pass filter cutoff frequency	Used to eliminate high frequency noise from the derivative term in the Auxiliary PID control loop.
P.DISC		Disabled command output	This value is the command input VCMD to the velocity control loop, when the Auxiliary PID control loop is disabled.
P.EHL		Error high limit	Imposes upper bound on the error between the setpoint P.SP and feedback P.FB signals. The resulting signal P.ERR is the input to the Auxiliary PID control loop.
P.ELL		Error low limit	Imposes lower bound on the error between the setpoint P.SP and feedback P.FB signals. The resulting signal P.ERR is the input to the Auxiliary PID control loop.
P.EN		Auxiliary PID Enable	Bit value which enables the Auxiliary PID control loop. Note that the drive must also be set in velocity command mode DM=VEL for the Auxiliary PID control loop to operate. This is because the output of the Auxiliary PID control loop P.OUT is added to the digital command value DCV entered into the velocity control loop.
P.FBG		Feedback gain	The feedback signal ADC2 (-10 V to +10 V) is multiplied by 1/10 so that the resultant range is -1.0 to 1.0. The alternative feedback source A420 has a range of 0.0 to 1.0. P.FBG scales the feedback signal (chosen by P.FBS) to user defined units.
P.FBO		Feedback offset	Offset added to the scaled feedback signal.
P.FBS		Feedback source	Use to select the feedback source. P.FBS=0 -> ADC2 P.FBS=1 -> A420
P.FHL		Feedback high limit	Imposes upper bound on the scaled and offset feedback signal.
P.FLL		Feedback low limit	Imposes lower bound on the scaled and offset feedback signal.
P.IHL		Integrator high limit	Imposes upper bound on the output of the integrator in the Auxiliary PID control loop.
P.ILL		Integrator low limit	Imposes lower bound on the output of the integrator in the Auxiliary PID control loop.
P.KD		Auxiliary PID derivative gain	Tune Auxiliary PID control loop parameter for desired output. Tuning procedures include standard methods such as Ziegler–Nichols.
P.KI		Auxiliary PID integral gain	Tune Auxiliary PID control loop parameter for desired output. Tuning procedures include standard methods such as Ziegler–Nichols.
P.KP		Auxiliary PID proportional gain	Tune Auxiliary PID control loop parameter for desired output. Tuning procedures include standard methods such as Ziegler–Nichols.
P.OFF		Output offset	Offset added to the scaled pre-limited output of the Auxiliary PID control loop.
P.OG		Output gain	Scale factor applied to the pre-limited output of the Auxiliary PID control loop.
P.OHL		Output high limit	Imposes upper bound on the pre-limited output of the Auxiliary PID control loop.

NAME	UNITS	DESCRIPTION	HOW TO USE
P.OLL		Output low limit	Imposes lower bound on the pre-limited output of the Auxiliary PID control loop.
P.PER		Auxiliary PID period multiplier	Integer defining the cycle time of the Auxiliary PID control loop. When P.PER=1 control loop updates once every 1 ms, when P.PER=2 loop updates every 2 ms, etc. Range of permitted values is 1 to 1000.
P.SETP		Digital setpoint	Digital setpoint is summed with scaled and limited input from ADC1 to form the total process setpoint. Digital setpoint can be used to offset signal from ADC1, or used as the sole input source by zeroing the gain applied to ADC1 (i.e. let P.ASG=0). This parameter is volatile.
P.SHL		Analog setpoint high limit	Imposes upper bound on the signal from ADC1.
P.SLL		Analog setpoint low limit	Imposes lower bound on the signal from ADC1.
P.SRL		Set point rate limit	Imposes rate limit on the summation of the digital setpoint P.SETP and the scaled and limited analog setpoint parameter ADC1.

12.0 I/O

12.1 Static Brake

Certain devices are equipped with an integrated static brake and/or control. Configurable settings include brake engage delays, invert operation, override, and brake coil voltage.

The following diagram illustrates static brake operation:

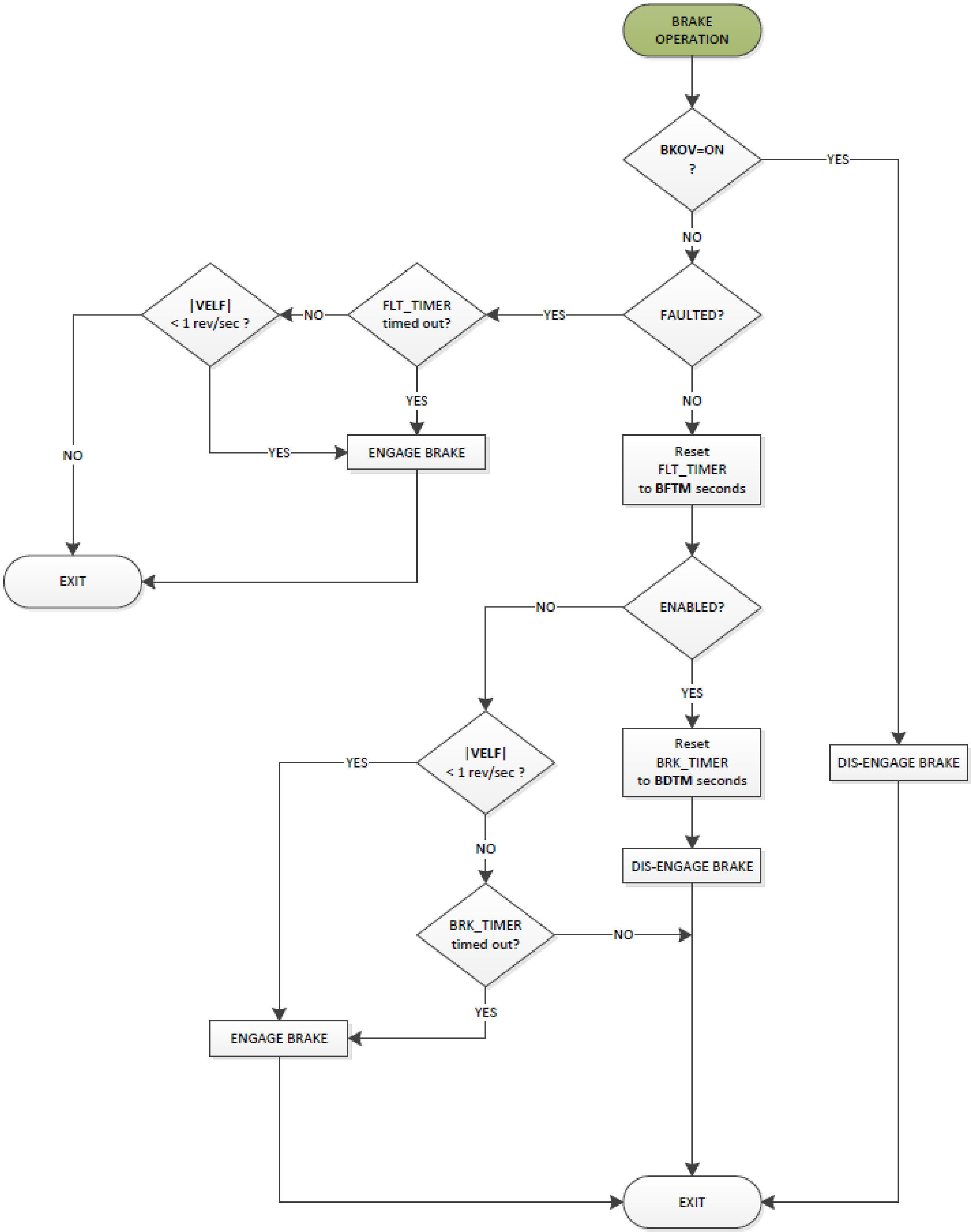

Figure 3: Brake Implementation Flow Chart

12.2 Calibrate Analog Inputs

Voltage drop across analog signal lines can cause the voltage at the signaling device and that measured downstream by the drive to differ significantly. Observe this difference by monitoring the analog input variables **ADC1**, **ADC2**, etc. This offset can be corrected by setting appropriate values to the calibration parameters **CAL1**, **CAL2**, etc. Note that these calibration values are intentionally not stored in the drive configuration file and must be set by hand for each device.

The "Calibrate ADC" button of IN Control allows the user to automatically calibrate the analog inputs. For this function to work properly, all analog inputs must be wired correctly and a zero volt command must be applied to each of the analog inputs when the calibration is being performed.

12.3 Configure Analog Output

Available analog outputs can be configured to signal any observable drive variable. Use **DAC1** to assign the desired drive variable for output, and scale using **DACMAX1** and **DACMIN1** parameters.

12.4 Digital Input Logic

Apply logic to assigned digital inputs using the **INOR.x** and **INAND.x** parameters.

12.5 Motor Thermistor

Winding temperature can be monitored by the drive using the thermistor inputs. Compatible thermistors are NTC type and characterized by the beta-parameter equation. Motor temperature is observed using variable **TMOT** and an over-temperature fault will be issued when **TMOT** exceeds the over-temperature threshold defined by **MTOT**.

Table 8: I/O Parameters

NAME	UNITS	DESCRIPTION	HOW TO USE
BDTM	s	Brake delay	Normally the brake engages at zero speed after disable. BDTM defines a time interval at drive disable, after which the brake will engage even if motor speed is still non-zero.
BFTM	s	Brake faulted delay	Normally the brake engages at zero speed after disable. BDTM defines a time interval at drive fault, after which the brake will engage even if motor speed is still non-zero.
BINV		Brake invert	Bit value inverts the operation of the brake (i.e. active power ON, or active power OFF).
BKOV		Brake override	Override disables all brake functionality.
BOUT		Brake output assignment	Defines which output controls the brake (BOUT=0 indicates no brake output assigned).
BVLT	V	Brake voltage	Defines the voltage output by the drive to the brake. For example, setting BVLT=12 allows a 12 V brake to be used with a 24 V supplied drive.
CAL1 to CAL4	V	Calibration offset for ADCx	Use to offset voltage losses between the signal source and analog input. Note that CALx cannot be saved to a drive configuration (.dcf) file using IN

NAME	UNITS	DESCRIPTION	HOW TO USE
			Control. Therefore when loading a configuration file, CALx will not be set and must be adjusted manually.
COUTS		Clear digital outputs	Clears output bits that were previously forced on with the force outputs parameter FOUTS. This parameter is volatile and cleared on power-up.
DAC1		Digital to analog converter 1	Select variable to output to analog output 1.
DACMAX1		Digital to analog converter 1 maximum	Define the value of DAC1 which is to be the maximum analog output voltage.
DACMIN1		Digital to analog converter 1 minimum	Define the value of DAC1 which is to be the minimum analog output voltage.
FOUTS		Force outputs	Forces on and latches one or more of the digital outputs. Unless deactivated by the drive firmware or by clear outputs COUTS, the output will remain active.
INAND.1 to INAND.12		Digital input "x" AND bit	Value can be either 0 or 1. The state of digital input "x" is ANDed with this parameter. If INAND.x=0, digital input "x" is effectively disabled regardless of the signal applied across digital input "x".
INOR.1 to INOR.12		Digital input "x" OR bit	Value can be either 0 or 1. The state of digital input "x" is ORed with this parameter. If INOR.x=1, digital input "x" is effectively enabled regardless of the signal applied across digital input "x".
MTB	°K	Motor thermistor beta	Is the beta parameter of an NTC thermistor. If no temperature measuring device is available, set this parameter to 4288 to avoid nuisance over-temperature faults.
MTOT	°C	Motor thermistor over-temperature limit	Defines the threshold at which a motor over-temperature fault will be produced. The detection algorithm assumes input from an NTC type thermistor which obeys the beta parameter equation. If no temperature measuring device is available on the motor, enter a value of 800 to avoid nuisance over-temperature faults.
MTR0	ohm	Motor thermistor resistance at 25C	Is the R0 parameter of an NTC thermistor. If no temperature measuring device is available, set this parameter to 50000 to avoid nuisance over-temperature faults.
SPD1	rev/s	Speed detect setpoint 1	When the speed (magnitude of the velocity) of the motor is greater than or equal to SPD1, the assigned speed-detect digital output will activate. Assign the speed detect digital output using the I/O Config (In) function of IN Control.
SPD2	rev/s	Speed detect setpoint 2	When the speed (magnitude of the velocity) of the motor is greater than or equal to SPD2, the assigned speed-detect digital output will activate. Assign the speed detect digital output using the I/O Config (In) function of IN Control.

13.0 I/O Config (In)

Each discrete input can be assigned a specific function under the “IO Config (In)” button of IN Control. The following functions are available:

13.1 RESET

When activated, any faults present in the drive are cleared.

13.2 + Overtravel

When activated, this function immediately halts motion in the positive direction when in velocity or position command modes. In actively decelerating the motor to zero speed, the torque command limit **TLIM** is applied. When "+ Overtravel" is deactivated, the motor resumes motion as before.

13.3 – Overtravel

When activated, this function immediately halts motion in the negative direction when in velocity or position command modes. In actively decelerating the motor to zero speed, the torque command limit **TLIM** is applied. When "– Overtravel" is deactivated, the motor resumes motion as before.

13.4 TLIM Control

TLIM is a parameter which limits the absolute torque command output of the velocity control loop. Normally, **TLIM** is always active when no digital inputs are assigned to "TLIM Control". However when a digital input is assigned, the torque command limit imposed by **TLIM** is ignored until that assigned digital input becomes active.

13.5 Quick Stop

When activated, this function immediately halts all motion when in velocity or position command modes. In actively decelerating the motor to zero speed, all command limits (i.e. **TLIM**, **ARL**, **DRL**, etc.) are ignored. When "Quick Stop" is deactivated, the motor resumes motion as before.

14.0 I/O Config (Out)

Each discrete output can be assigned a specific function under the "IO Config (Out)" button of IN Control. The following functions are available:

14.1 FAULT

The assigned output is active when the drive has faulted.

14.2 In Current Limit

The assigned output is active when the torque producing current reference **IQREF** is equal to the drive current limit **ILIM**. This will commonly occur when the motor is loaded with a torque requiring current higher than what the drive can supply, or when commanding a velocity beyond the no-load speed of the motor.

14.3 Zero-Speed

The assigned output is active when the motor speed is within +/- 1 rev/s.

14.4 At SPD1

The assigned output is active when motor speed exceeds the defined parameter **SPD1** (set value of **SPD1** under the "I/O" button of IN Control). This means that if the feedback velocity **VEL** is > **SPD1** or < **-SPD1** the output will be active.

14.5 At SPD2

The assigned output is active when motor speed exceeds the defined parameter **SPD2** (set value of **SPD2** under the "I/O" button of IN Control). This means that if the feedback velocity **VEL** is > **SPD2** or < **-SPD2** the output will be active.

15.0 Command

In order to operate, the drive must be provided with a signal to enable/disable and a command value. Utilize the command parameters to configure these settings.

Using the command mode parameter **CM**, first choose the method of operation for the drive. This selection will configure the drive to track either a current, torque, velocity, and/or position command.

15.1 Drive Enable

Once powered, the drive can be enabled to operate by writing to the software enable parameter **EN** (1=ON, 0=OFF), by configuring the drive to enable when a digital input is activated, or using a logical combination of the two. To configure the drive for one of these settings, input the appropriate value to the drive enable mode parameter **EM**.

15.2 Command Value

When the drive is configured for either current, torque, or velocity command modes, the target reference or command value **CV** is produced using the following structure:

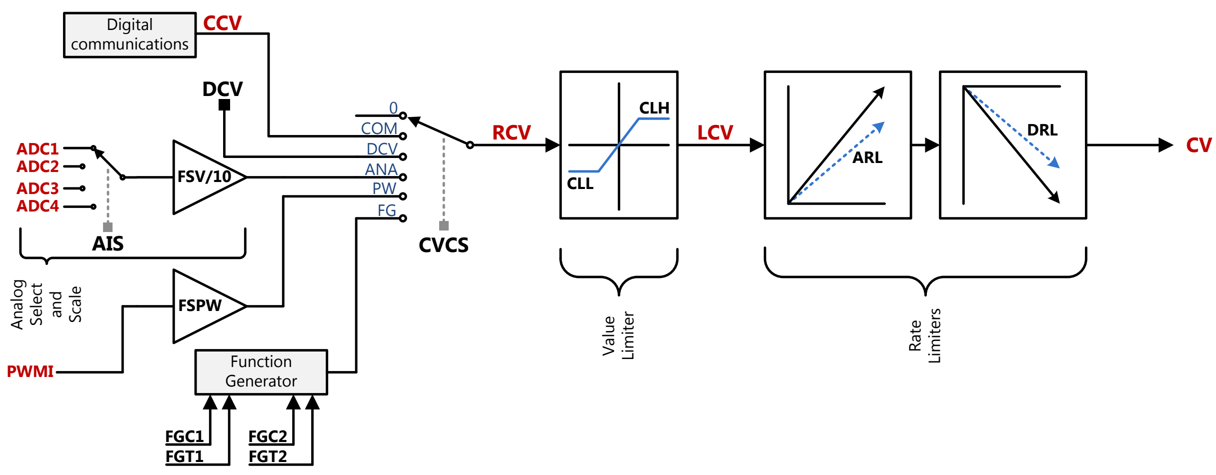

Figure 4: Current, Torque, and Velocity Command Structure

The inputs of this diagram are a variety of command sources. Select one of the sources using the command value control switch **CVCS**. For example, select analog input 1 to be the chosen command source be setting **CVCS=ANA** and **AIS=1**. A series of value (**CLL**, **CLH**) and rate (**ARL**, **DRL**) limiters are provided to condition the command source signal.

The output of the diagram is the command value **CV**. Its units are dependent upon the chosen command mode. For example, if the drive is in torque command mode **CM=TRQ**, **CV** is scaled to units of N-m and serves as the torque command **TRQC**. The torque command **TRQC** is divided by the motor torque constant **KT** to become the torque producing current reference **IQREF** for the current control loop.

$$\frac{\mathbf{TRQC}\,[\mathrm{N}\cdot\mathrm{m}]}{\mathbf{KT}\,\frac{[\mathrm{N}\cdot\mathrm{m}]}{[\mathrm{A}]}} = \mathbf{IQREF}\,[\mathrm{A}]$$

When the drive is in position command mode **CM=POS**, the preceding command structure is not used (see Motion Sequence Command for more information).

15.2.1 Analog Command

When using an analog input as the command source **CVCS=ANA**, two parameters control the input used and its scaling. **AIS** selects the source from one of the ADC channels, and the parameter **FSV** scales the analog input to the desired command value.

15.2.2 Digital Command

The drive can be provided with a digital command source by setting **CVCS=DCV**. Write to **DCV** from the IN Control user interface or using network communications (e.g. ALLNET .NET Framework, CANopen, or Modbus RTU).

15.2.3 Function Generator Command

The function generator is activated by setting **CVCS=FG**. The function generator programmed by the user to output a square wave command signal which alternates between two values (**FGC1** and **FGC2**) for two different periods (**FGT1** and **FGT2**).

For example, when in velocity command mode **CM=VEL**, the drive can be commanded to go between two velocities set by **FGC1** and **FGC2**. The speed set by **FGC1** is commanded for a period of **FGT1** seconds, and the speed set by **FGC2** is commanded for a period of **FGT2** seconds.

This command source can be useful in commissioning and tuning of the motor/drive system.

15.2.4 Motion Sequence Command

When the drive is in position command mode **CM=POS**, the target reference for the position control loop **TPOS** is produced by an internal target generator given a specific set of motion parameters; depending on your specific hardware, either **RUNMS** in conjunction with **REG00** to **REG07** and **STPMS** are used to execute a motion sequence, or the register parameters **R0** to **R15** are used.

To run a programmed motion sequence, set sequence to auto-run **EMS=1**. With this setting, the motion sequence will automatically execute upon drive enable.

Table 9: Command Parameters

NAME	UNITS	DESCRIPTION	HOW TO USE
ADEAD	V	Analog input deadband	Deadband applied to analog input command signal selected by AIS.
AIS		Select analog input source	Use to select source of raw command signal RCV when in analog command mode CM=ANA: AIS=0 -> ADC1 AIS=1 -> ADC2 AIS=2 -> ADC3 AIS=3 -> ADC4
ARL	unit/s	Positive rate limit	Imposes positive rate limit on the limited command value LCV to produce the command value CV. If in velocity command mode CM=VEL then ARL limits the commanded rate of acceleration.
CLH		High command limit value	Imposes upper bound on the raw command value RCV to produce the limited command value LCV.
CLL		Low command limit value	Imposes lower bound on the raw command value RCV to produce the limited command value LCV.
CM		Command mode	Select the drive command mode: CM=0 -> CUR

NAME	UNITS	DESCRIPTION	HOW TO USE
			In current command mode, the command value CV becomes the torque producing motor current IQREF. CM=1 -> TRQ In torque command mode, the command value CV becomes the torque command TRQC. CM=2 -> VEL In velocity command mode, the command value CV becomes the velocity command VCMD. CM=3 -> POS In position command mode, the drive's target generator is active. Given register inputs (R0, R1, R2, or RUNMS, STPMS, REG00, REG01, etc.) to a UPOB loaded into memory (activated by motion sequence auto-run EMS), the target generator outputs a target position TPOS to the position control loop. CM=4 -> OLV In open-loop velocity command mode, the command value CV becomes the torque producing voltage demand VQ, while the flux producing voltage demand VD is held at zero.
CVCS		Command variable control switch	Select the drive command source: CVCS=0 -> 0 Sets the raw command value RCV to zero. CVCS=1 -> COM Sets the raw command value RCV to CCV. CVCS=2 -> DCV Sets the raw command value RCV to DCV. CVCS=3 -> ANA Sets the raw command value RCV to be an analog input. Select the specific analog input using AIS. CVCS=4 -> FG Sets the raw command value RCV to be the function generator output. The function generator outputs a square-wave signal defined by parameters FGC1, FGC2, FGT1, and FGT2. This mode can be useful in commissioning and tuning of the motor/drive system. CVCS=5 -> PW Reserved.
DCV		Digital command value	When CVCS=DCV, DCV is the input that corresponds to the raw command value RCV. The units of DCV are determined by the chosen command mode CM. For use with fieldbus communications other than Ethernet (e.g. RS-232, RS-485, or CAN). DCV is intentionally volatile (DCV=0 at power-up).
DRL	unit/sec	Negative rate limit	Imposes negative rate limit on the limited command value LCV to produce the command value CV. If in velocity command mode CM=VEL then DRL limits the commanded rate of deceleration.
EM		Drive enable mode	Select source of the drive enable signal: EM=0 -> SWE

NAME	UNITS	DESCRIPTION	HOW TO USE
			Software enable mode allows software to enable and disable the drive. In this mode the parameter EN is set to ON or OFF to enable or disable the drive. EN can be set through the IN Control user interface or through fieldbus communications. EM=1 -> HWE Hardware enable mode allows one of the drive's digital inputs to enable and disable the drive. This digital input selected by assigning a value to the hardware enable input ENI. EM=2 -> OR The drive is enabled by a logical OR of the hardware enable bit and the software enable parameter EN. EM=3 -> AND The drive is enabled by a logical AND of the hardware enable bit and the software enable parameter EN.
EMS		Motion sequence auto run	Bit value which enables a stored motion program to execute when the drive is enabled. Parameter is only used when in position command mode CM=POS.
EN		Software drive enable	Bit value which enables the drive when in software enable mode EM=SWE.
ENI		Hardware enable input	Select digital input (1 to 32) to enable the drive when in hardware enable mode EM=HWE.
FGC1	unit	Function generator constant amplitude 1	Constant amplitude value of the first level of the square-wave signal. Units are defined by the command mode CM.
FGC2	unit	Function generator constant amplitude 2	Constant amplitude value of the second level of the square-wave signal. Units are defined by the command mode CM.
FGT1	s	Function generator time duration 1	Duration of time at constant amplitude FGC1.
FGT2	s	Function generator time duration 2	Duration of time at constant amplitude FGC2.
FSPW		Full scale PWM command value	Reserved.
FSV		Full scale analog command value	When CVCS=ANA, FSV is used to scale the analog input command signal. It corresponds to the desired command value when the analog input is 10 V.
R0 to R15		Register x	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).
REG00 to REG07		Volatile memory register xx	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).
RUNMS		Run motion sequence.	A non-zero value starts a motion sequence: RUNMS=1 A single incremental index is performed. The motion sequence stops once the index is complete. REG00 is displacement (rev) from marker, REG01 is acceleration/deceleration rate in (rev/s^2), REG02 is maximum velocity in (rev/s). RUNMS=2 An incremental index is performed on every rise of digital input 1. REG00 is displacement (rev) from marker, REG01 is acceleration/deceleration rate in (rev/s^2), REG02 is maximum velocity in (rev/s). RUNMS=3 Zero's the target position maintaining following error.

NAME	UNITS	DESCRIPTION	HOW TO USE
			RUNMS=4 Motor is positioned relative to the encoder "Z" marker pulse. REG00 is displacement (rev) from marker, REG01 is acceleration/deceleration rate in (rev/s^2), REG02 is maximum velocity in (rev/s). RUNMS=5 Reserved. RUNMS=6 Reserved. RUNMS=7 Enters PV mode which allows the user to specify a position to move to on-the-fly together with a maximum velocity and acceleration/deceleration rate. REG00 is displacement (rev) from marker, REG01 is acceleration/deceleration rate in (rev/s^2), REG02 is maximum velocity in (rev/s). RUNMS=8 Exits PV mode and stops motion. RUNMS=9 A repeated out and back incremental index is performed with a delay after completion of each index. The motion sequence continues to execute unless the drive is disabled or STPMS is set to 1. REG00 is displacement (rev) from marker, REG01 is acceleration/deceleration rate in (rev/s^2), REG02 is maximum velocity in (rev/s), REG03 is delay time (ms).
STPMS		Stop motion sequence	Setting this bit to 1 stops a running motion sequence. STPMS is automatically set to 0 once the motion sequence has stopped. This parameter is intentionally volatile (STPMS=0 on power-up).

16.0 Fieldbus Configuration

Depending on the specific hardware available, Allied Motion products are capable of fieldbus communications. Refer to your product datasheet for specific networking capabilities. Utilize these parameters to input various communications settings including communication baud rate and node ID.

Table 10: Fieldbus Configuration Parameters

NAME	UNITS	DESCRIPTION	HOW TO USE
CANBR	kbps	CAN baud rate	Select baud rate of the CAN network.
CANID		CAN node identification number	Assign CAN node ID to drive.
MODID		Modbus identification number	Assign Modbus node ID to drive.

17.0 Status

Status variables provide monitors for drive board temperature, DC bus voltage, regenerative power output, enabled and fault status.

Note that drive faults are cleared by toggling the "Reset" button under the "Diagnose" tab of IN Control or by power cycling the drive. Faults can also be reset by so configuring a digital input. In addition, the parameter **FLTRSET** is available for clearing drive faults via fieldbus communications (e.g. CANopen, Modbus RTU, etc.).

A fieldbus watchdog is provided by which to monitor fieldbus communications. Should communications fail, the watchdog function will halt operation and issue a drive fault. See parameter **CWD** for details.

17.1 Fault Status Troubleshooting

17.1.1 Bad Current Offset

Return to Allied Motion Controls for repair.

17.1.2 Bus Over-Voltage

If the DC bus voltage **VBUS** exceeds the threshold limit (see product datasheet for device specific value), the drive is disabled and a bus over-voltage fault is declared.

This fault typically occurs during load deceleration when the regenerative energy produced exceeds the storage available in the DC bus capacitors. See "Document No. 34-2100 Hardware Manual: Wiring and Installation" for methods by which to properly dissipate regenerative energy, thus avoiding bus over-voltage faults.

17.1.3 Bus Under-Voltage

If the DC bus voltage **VBUS** falls below the threshold limit (see product datasheet for device specific value), the drive is disabled and a bus under-voltage fault is declared.

This fault typically occurs during load acceleration when the current demand of the drive exceeds the output available from the power supply. To avoid this scenario, a power supply with the proper current rating must be installed.

17.1.4 Drive Over-Temperature

The temperature of the drive power stage circuitry is monitored by observing the variable **TPWR**. When the value of **TPWR** exceeds 100 °C, a drive over-temperature fault is activated.

This fault typically occurs when the drive is operating under load in ambient temperatures above 50 °C and/or when proper heat sinking has not been applied to the drive housing. To avoid this scenario, mount the drive or motor/drive combination to a properly rated heat sink, and/or remove excess heat by blowing cool air across the drive housing or some similar method of forced convection.

17.1.5 Encoder Loss

If the drive mode is set for encoder position feedback (e.g. **DM=BLE**, **DM=BSC**, **DM=BLE_NH**, etc.), an encoder loss fault will be activated if no encoder is detected by the drive.

17.1.6 Fieldbus Watchdog Timeout

The fieldbus watchdog acts to confirm active fieldbus communications. If the timer has expired, this means that the watchdog parameter **CWD** has not been reloaded with an integer value, indicating that either the master controller has not issued a new value to **CWD** or that it cannot because fieldbus communications have failed.

17.1.7 Invalid Hall State

When commutating properly using hall sensor feedback, the variable **HALLS** proceeds in hexadecimal order 1, 3, 2, 6, 4, 5. States 0 and 7 are illegal and if present indicate incorrect wiring and/or electrical interference between the motor and drive. If an illegal hall state occurs, an invalid hall state fault will be activated.

17.1.8 Invalid ID Configuration

Return to Allied Motion Controls for repair.

17.1.9 Locked Rotor

A locked rotor is defined as when the motor is stalled and the time averaged torque producing current reference **IQAVG** is greater than **LRIL**. When this condition persists for greater than 4 seconds, a locked rotor fault is produced.

To disable all locked rotor faults, set **LRIL=0**.

17.1.10 Motor Over-Temperature

An over-temperature fault will be issued when the motor temperature **TMOT** exceeds the over-temperature threshold defined by **MTOT**. To continue operation, wait for the motor to cool before clearing the fault.

To avoid this scenario, mount the motor to a properly rated heat sink, and/or remove excess heat by blowing cool air across the motor housing or some similar method of forced convection.

17.1.11 Oscillator Failure

Return to Allied Motion Controls for repair.

17.1.12 Over-Current

Occurs when the output current of the drive exceeds its given rating.

An over-current fault typically is the result of poor current loop tuning. For proper current loop tuning methods, refer to the Current Loop section of this document.

An over-current fault might also indicate the presence of a motor which does not meet the drive's minimum inductance requirement for compatibility (refer to product datasheet for specific value).

17.1.13 Short Circuit

When a short circuit is detected between individual motor phases, phase to ground, or within the drive power stage circuitry, a short circuit fault is activated.

Check for a short circuit in the motor by measuring the resistance between all combinations of motor phase leads and phase leads to ground. If a short circuit is detected, return the device to Allied Motion for repair.

Table 11: Status Parameters

NAME	UNITS	DESCRIPTION	HOW TO USE
CWD	ms	Fieldbus Watchdog	CWD serves as a millisecond timer by which to confirm active fieldbus communications. The first write to CWD, after the drive is powered, both enables and loads the watchdog timer. The timer then counts down once every millisecond until it reaches zero, at which moment the Timeout Action described below is taken. Note that once the watchdog timer is enabled, its action cannot be cancelled. CWD can take on any integer value from 1 to 10,000. Once the timer is enabled, it is the master controller's responsibility to reload CWD with a new value periodically before the timer reaches zero, else the Timeout Action will be executed. The new value transmitted can differ from the previous value, which allows for dynamic timeout management if so desired. Timeout Action: If in position command mode CM=POS, the target generator is stopped immediately, and upon reaching zero-speed, the drive is faulted with "Fieldbus Watchdog Timeout". If in velocity command mode CM=VEL, a zero-speed command is issued, and upon reaching zero-speed, the drive is faulted with "Fieldbus Watchdog Timeout". The drive will decelerate the motor according to the negative rate limit DRL.

NAME	UNITS	DESCRIPTION	HOW TO USE
			If in current or torque command mode CM=CUR or CM=TRQ, the drive is faulted immediately with "Fieldbus Watchdog Timeout".
FLTRSET		Fault reset	Allows faults to be cleared over fieldbus network (e.g. CAN or RS-485). Faults are cleared when parameter transitions from 0 to 1. Parameter not visible through IN Control. To reset faults through IN Control, utilize the "Reset" button on the "Diagnose" tab.

18.0 Revision History

Revision	Description of Change	Date
R1	Initial release.	June 9, 2014
R2	Renumbered document sections. ENABLED parameter moved to Status category. KD parameter corrected to indicate KVF=1.0 is pure PID control. How to use description of COFF and C1 inversion bits updated. Added Fieldbus Watchdog CWD.	January 12, 2015
R3	Attachment A appended with corrected row heights.	March 10, 2015

19.0 Attachments

Attachment A, Sortable Parameters and Variables List.xlsx

Attachment A

Sortable Parameters and Variables List.xlsx

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RO	Auxiliary PID Loop	A420		Analog input 2 current	The value of A420 ranges between 0 and 1 for a 4 to 20 mA signal (i.e. 0=4 mA and 1=20 mA). This variable may be selected as a feedback input for the Aux PID control loop.	Real32	0x0008	0x2118	560
RO	Auxiliary PID Loop	P.ERR		PID error	Difference between the PID setpoint variable P.SP and the PID feedback variable P.FB. When the drive is operating properly, the steady state value is driven to zero.	Real32	0x0008	0x211B	566
RO	Auxiliary PID Loop	P.FB		PID feedback	Final PID feedback variable which is the summation of a conditioned analog input signal ADC2 or A420 with the feedback offset P.FBO, followed by the application of value limit parameters P.FLL and P.FHL.	Real32	0x0008	0x211A	564
RO	Auxiliary PID Loop	P.INT		PID integrator	Integrator output of the Aux PID control loop.	Real32	0x0008	0x211C	568
RO	Auxiliary PID Loop	P.OUT		PID output	Final output of the Aux PID control loop, produced from P.PIDO following the application of value limit parameters P.OLL and P.OHL. This value is added to the digital command value DCV, the total sum of which becomes the raw command value RCV input to the velocity control loop.	Real32	0x0008	0x211E	572
RO	Auxiliary PID Loop	P.PIDO		PID pre-limited output	Intermediate output of the Aux PID control loop, following the application of the output gain P.OG and summation with the output offset P.OFF.	Real32	0x0008	0x211D	570
RO	Auxiliary PID Loop	P.SP		PID setpoint	Final PID setpoint variable which is the summation of a conditioned analog input signal ADC1 and the digital setpoint P.SETP, followed by the application of rate limit parameter P.SRL.	Real32	0x0008	0x2119	562
RO	Command	CCV		Communication (high-speed fieldbus) based command value	When CVCS=COM, CCV is the input that corresponds to the raw command value RCV. The units of CCV are determined by the chosen command mode CM. Read/write access only available through high-speed fieldbus (e.g. Ethernet). To issue commands over RS-232, RS-485, or CAN utilize digital command value DCV.	Real32	0x0008	0x204F	158
RO	Command	CV		Command value	Final command value produced from LCV following the application of rate limit parameters ARL, DRL.	Real32	0x0008	0x2052	164
RO	Command	FGCV		Function generator command output	When CVCS=FG, FGCV is the input that corresponds to the raw command value RCV. Function generator mode CVCS=FG produces a square-wave signal which steps between amplitudes FGC1 and FGC2 of duration FGT1 and FGT2, respectively. FGCV can be useful for drive commissioning and control loop tuning.	Real32	0x0008	0x204D	154
RO	Command	LCV		Limited command value	Intermediate command value produced from RCV following the application of value limit parameters CLL, CLH.	Real32	0x0008	0x2051	162
RO	Command	PWMI		PWM input duty cycle	Represents the duty cycle of the PWM input command signal. PWMI=0.5 represents 50% duty cycle input command.	Real32	0x0008	0x2014	40
RO	Command	RCV		Raw command value	Represents the raw command value from the source selected by CVCS.	Real32	0x0008	0x2050	160
RO	Commutation	DQ	count	Previous encoder change	Last change in primary encoder feedback.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Commutation	DQI	count	Present encoder change	Change in primary encoder feedback this interrupt.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Commutation	DQV	count	Observer encoder change	Change in feedback position as seen by the velocity observer.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Commutation	EANG	rad	Electrical commutation angle in radians	Used for motor commutation. The number of electrical cycles per motor revolution is equal to the number of motor pole pairs.	Real32	0x0008	0x201A	52
RO	Commutation	EANGD	deg	Electrical commutation angle in degrees	Used for motor commutation. The number of electrical cycles per motor revolution is equal to the number of motor pole pairs.	Real32	0x0008	0x201B	54
RO	Commutation	ETHET	count	Encoder angle	When an encoder is available as the commutation feedback source, ETHET corresponds to the encoder angle over one motor revolution, counting from 0 to the number of encoder counts per revolution EPPR.	Integer32	0x0004	0x2020	64
RO	Commutation	FBC	count	Hardware counter for commutation feedback source	Represents a rolling pulse count from the commutation feedback source, ranging in value from 0 to the maximum integer value storable in memory.	Unsigned32	0x0007	0x2036	108
RO	Commutation	FCPR	count/rev	Resolution of feedback source	Represents the total number of pulse counts (observable positions) over one motor revolution of the selected feedback source.	Integer32	0x0004	0x2080	256

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RO	Commutation	FCPR1	count/rev	Resolution of primary feedback source	Represents the total number of counts (observable positions) over one motor revolution.	Integer32	0x0004	0x2081	258
RO	Commutation	FCPR2	count/rev	Resolution of secondary feedback source	Represents the total number of counts (observable positions) over one motor revolution.	Integer32	0x0004	0x2082	260
RO	Commutation	FCPR3	count/rev	Resolution of tertiary feedback source	Represents the total number of counts (observable positions) over one motor revolution.	Integer32	0x0004	0x2083	262
RO	Commutation	HALLS		Hall state	Binary representation of the motor hall state. IN Control converts this binary representation and displays a hexadecimal value. When commutating properly, HALLS proceeds in hexadecimal order 1, 3, 2, 6, 4, 5. States 0 and 7 are illegal and if present indicate incorrect wiring and/or electrical interference between the motor and drive.	Integer32	0x0004	0x2018	48
RO	Commutation	HANG	rad	Hall commutation angle in radians	Represents the electrical commutation angle as measured by hall sensor position feedback.	Real32	0x0008	0x201E	60
RO	Commutation	HANGD	deg	Hall commutation angle in degrees	Represents the electrical commutation angle as measured by hall sensor position feedback.	Real32	0x0008	0x201F	62
RO	Commutation	MAGE	count	Raw magnetic encoder reading	Raw magnetic encoder reading refers to an absolute angle from 0 to the number of encoder counts per revolution.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Commutation	MANG	rad	Mechanical commutation angle in radians	Used for motor commutation. Note that when using hall sensors only for commutation DM=BLH, MANG can only range from 0 to 2*pi/n radians, where 'n' equals the number of motor pole pairs.	Real32	0x0008	0x201C	56
RO	Commutation	MANGD	deg	Mechanical commutation angle in degrees	Used for motor commutation. Note that when using hall sensors only for commutation DM=BLH, MANGD can only range from 0 to 360/n degrees, where 'n' equals the number of motor pole pairs.	Real32	0x0008	0x201D	58
RO	Commutation	OPT1		Option register 1	Binary representation of the feedback type. If the least significant bit is 1, then the system has absolute type position feedback.	Unsigned32	0x0007	0x206D	218
RO	Commutation	S.COS	count	Scaled analog sine wave feedback	Is a scaled version of the raw sine-component input from a sin/cos encoder. Scaling is as follows: 32768 count = 0 V or the baseline of the sine wave, and 65535 corresponds to the peak of the sine wave.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Commutation	S.SIN	count	Scaled analog cosine wave feedback	Is a scaled version of the raw cosine-component input from a sin/cos encoder. Scaling is as follows: 32768 count = 0 V or the baseline of the cosine wave, and 65535 corresponds to the peak of the cosine wave.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Commutation	THET	count	Hardware counter for position feedback source	Represents a rolling pulse count for the position feedback source, ranging in value from 0 to the maximum integer value storable in memory. THET is used exclusively by the position control loop.	Unsigned32	0x0007	0x501F	24638
RO	Current Loop	IALPHA	A	Alpha-axis current feedback	Implementing a Field Oriented Control algorithm, IALPHA is the alpha-component of the Clarke transformation of the three-phase motor current vector (a,b,c) into the rotating two-phase current vector (alpha, beta).	Real32	0x0008	0x5001	24578
RO	Current Loop	IBETA	A	Beta-axis current feedback	Implementing a Field Oriented Control algorithm, IBETA is the beta-component of the Clarke transformation of the three-phase motor current vector (a,b,c) into the rotating two-phase current vector (alpha, beta).	Real32	0x0008	0x5002	24580
RO	Current Loop	ID	A	Flux producing current feedback	Implementing a Field Oriented Control algorithm, ID is the d-component of the Park transformation of the rotating two-phase motor current vector (alpha,beta) into the stationary two-phase current vector (d,q).	Real32	0x0008	0x5003	24582
RO	Current Loop	IDERR	A	Flux producing current error	Difference between the flux producing current reference IDREF and the flux producing current feedback ID. When the drive is operating properly, the steady state value is driven to zero.	Real32	0x0008	0x5005	24586
RO	Current Loop	IDREF	A	Flux producing current reference	Input flux producing current reference for the motor current control loop. For optimally efficient operation of permanent magnet motors, this value is zero.	Real32	0x0008	0x5007	24590
RO	Current Loop	IFA	A	Phase A motor current feedback	Implementing a Field Oriented Control algorithm, IFA is the a-component of the three-phase motor current vector (a,b,c).	Real32	0x0008	0x2022	68
RO	Current Loop	IFAHR	A	Phase A motor current feedback (high resolution)	If hardware for high resolution current feedback has been installed, IFAHR is a high resolution version of the Phase A motor current, ordinarily represented by IFA.	Real32	0x0008	0x2022	68
RO	Current Loop	IFB	A	Phase B motor current feedback	Implementing a Field Oriented Control algorithm, IFB is the b-component of the three-phase motor current vector (a,b,c).	Real32	0x0008	0x2023	70

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RO	Current Loop	IFBHR	A	Phase B motor current feedback (high resolution)	If hardware for high resolution current feedback has been installed, IFBHR is a high resolution version of the Phase B motor current, ordinarily represented by IFB.	Real32	0x0008	0x2023	70
RO	Current Loop	IFC	A	Phase C motor current feedback	Implementing a Field Oriented Control algorithm, IFC is the c-component of the three-phase motor current vector (a,b,c).	Real32	0x0008	0x2024	72
RO	Current Loop	ILIM	A	Current limit	Defines the operating current limit of the drive. ILIM is a moving target equal to the peak current rating of the drive, which reduces in value to IMAX according to an I^2*T current limiting scheme.	Real32	0x0008	0x203D	122
RO	Current Loop	IQ	A	Torque producing current feedback	Implementing a Field Oriented Control algorithm, IQ is the q-component of the Park transformation of the rotating two-phase motor current vector (alpha,beta) into the stationary two-phase current vector (d,q).	Real32	0x0008	0x5004	24584
RO	Current Loop	IQAVG	A	Time averaged torque producing current reference	Represents a time averaged value of the torque producing current reference IQREF.	Real32	0x0008	0x2021	66
RO	Current Loop	IQERR	A	Torque producing current error	Difference between the torque producing current reference IQREF and the torque producing current feedback IQ. When the drive is operating properly, the steady state value is driven to zero.	Real32	0x0008	0x5006	24588
RO	Current Loop	IQREF	A	Torque producing current reference	Target reference for the motor current control loop. When operating the drive in current command mode CM=CUR, the command value CV is equal to IQREF.	Real32	0x0008	0x5008	24592
RO	Current Loop	PWMA		Phase A PWM duty cycle	Represents the duty cycle of the PWM voltage applied across motor phase A. PWMA=0.5 represents 50% duty cycle across motor phase A.	Real32	0x0008	0x5009	24594
RO	Current Loop	PWMB		Phase B PWM duty cycle	Represents the duty cycle of the PWM voltage applied across motor phase B. PWMB=0.5 represents 50% duty cycle across motor phase B.	Real32	0x0008	0x500A	24596
RO	Current Loop	PWMC		Phase C PWM duty cycle	Represents the duty cycle of the PWM voltage applied across motor phase C. PWMC=0.5 represents 50% duty cycle across motor phase C.	Real32	0x0008	0x500B	24598
RO	Current Loop	UALPHA		Unitless alpha-axis voltage demand	Represents alpha-axis voltage demand divided by the DC bus voltage VBUS.	Real32	0x0008	0x5010	24608
RO	Current Loop	UBETA		Unitless beta-axis voltage demand	Represents alpha-axis voltage demand divided by the DC bus voltage VBUS.	Real32	0x0008	0x5011	24610
RO	Current Loop	VALPHA	V	Alpha-axis voltage demand	The voltage demand associated with the alpha-component of the rotating two-phase motor current vector (alpha, beta).	Real32	0x0008	0x500C	24600
RO	Current Loop	VBETA	V	Beta-axis voltage demand	The voltage demand associated with the beta-component of the rotating two-phase motor current vector (alpha, beta).	Real32	0x0008	0x500D	24602
RO	Current Loop	VD	V	Flux producing voltage demand	The voltage demand associated with the d-component of the stationary two-phase motor current vector (d,q).	Real32	0x0008	0x500E	24604
RO	Current Loop	VQ	V	Torque producing voltage demand	The voltage demand associated with the q-component of the stationary two-phase motor current vector (d,q).	Real32	0x0008	0x500F	24606
RO	I/O	ADC1	V	Analog input 1 voltage	When AIS=ADC1 and CVCS=ANA, the value of ADC1 is scaled by FSV to produce the raw command value RCV. See ADEAD and CAL1 to add deadband and calibrate ADC1. Alternatively, ADC1 can serve as a setpoint input to the Aux PID control loop.	Real32	0x0008	0x200C	24
RO	I/O	ADC2	V	Analog input 2 voltage	When AIS=ADC2 and CVCS=ANA, the value of ADC2 is scaled by FSV to produce the raw command value RCV. See ADEAD and CAL2 to add deadband and calibrate ADC2. Alternatively, ADC2 can serve as a feedback input for the Aux PID control loop.	Real32	0x0008	0x200D	26
RO	I/O	ADC3	V	Analog input 3 voltage	When AIS=ADC3 and CVCS=ANA, the value of ADC3 is scaled by FSV to produce the raw command value RCV. See ADEAD and CAL3 to add deadband and calibrate ADC3.	Real32	0x0008	0x200E	28
RO	I/O	ADC4	V	Analog input 4 voltage	When AIS=ADC4 and CVCS=ANA, the value of ADC4 is scaled by FSV to produce the raw command value RCV. See ADEAD and CAL4 to add deadband and calibrate ADC4.	Real32	0x0008	0x200F	30
RO	I/O	INS		State of digital inputs	Binary representation of the digital inputs. IN Control converts this binary representation and displays a hexadecimal value.	Integer32	0x0004	0x2015	42
RO	I/O	OUTS		State of the digital outputs	Binary representation of the digital outputs. IN Control converts this binary representation and displays a hexadecimal value.	Integer32	0x0004	0x2016	44
RO	I/O	TMOT	°C	Motor temperature	Measured temperature given input from an NTC type thermistor. Thermistor properties are entered into drive software as parameters MTB, MTOT, and MTR0.	Real32	0x0008	0x2048	144

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RO	Position Loop	CP		Code pointer	Used to view present code line being executed during motion sequence.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	CV0		Integer register 0	Integer memory storage available for motion sequences. Values are stored in volatile memory (set to 0 on power-up).	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	CV1		Integer register 1	Integer memory storage available for motion sequences. Values are stored in volatile memory (set to 0 on power-up).	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	CV2		Integer register 2	Integer memory storage available for motion sequences. Values are stored in volatile memory (set to 0 on power-up).	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	CV3		Integer register 3	Integer memory storage available for motion sequences. Values are stored in volatile memory (set to 0 on power-up).	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	CV4		Integer register 4	Integer memory storage available for motion sequences. Values are stored in volatile memory (set to 0 on power-up).	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	CV5		Integer register 5	Integer memory storage available for motion sequences. Values are stored in volatile memory (set to 0 on power-up).	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	CV6		Integer register 6	Integer memory storage available for motion sequences. Values are stored in volatile memory (set to 0 on power-up).	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	CV7		Integer register 7	Integer memory storage available for motion sequences. Values are stored in volatile memory (set to 0 on power-up).	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	FPOS	rev	Feedback position in revolutions	Position of the commutation feedback source. Note that FPOS is not a valid measure of motor position when using hall sensors only for commutation DM=BLH.	Real32	0x0008	0x2027	78
RO	Position Loop	FPOSC	count	Feedback position in degrees	Position of the commutation feedback source. Note that FPOSC is not a valid measure of motor position when using hall sensors only for commutation DM=BLH.	Integer32	0x0004	0x2028	80
RO	Position Loop	MFLT		Motion fault	Fault status of UPOB code execution.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	MSTAT		Motion sequence status bits	Binary representation of the motion status. IN Control converts this binary representation and displays a hexadecimal value. Bit 0=1 -> Running a motion sequence Bit 1=1 -> Idle but preparing to run a motion sequence Bit 2=1 -> Generating target positions: in a move Bit 3=1 -> Finished target generation: at end position Bit 4=1 -> Stopping Bit 5=1 -> Decelerating to a stop Bit 6=1 -> Changing speed mid-move Bit 7=1 -> Pre-calculating a spline table Bit 8=1 -> Pre-calculating a CAM table Bit 9=1 -> iterating through a spline table Bit 10=1 -> Iterating through a CAM table Bit 11=1 -> Ending a spline/CAM table-based move Bit 12=1 -> Invalid Hall State Bit 13=1 -> (NRE) Bad Current Offset Bit 14=1 -> Reserved Bit 15=1 -> Reserved	Integer16	0x0003	0x20AC	344
RO	Position Loop	PERR	rev	Position error	Difference between the target position TPOS and the feedback position FPOS. When the drive is operating properly, the steady state value is driven to zero.	Real32	0x0008	0x501C	24632
RO	Position Loop	POS1	rev	Primary feedback position in revolutions	Position of the primary feedback source.	Real32	0x0008	0x2029	82
RO	Position Loop	POS1C	counts	Primary feedback position in counts	Position of the primary feedback source.	Integer32	0x0004	0x202A	84

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RO	Position Loop	POS2	rev	Secondary feedback position in revolutions	Position of the secondary feedback source.	Real32	0x0008	0x202B	86
RO	Position Loop	POS2C	counts	Secondary feedback position in revolutions	Position of the secondary feedback source.	Integer32	0x0004	0x202C	88
RO	Position Loop	POS3	rev	Tertiary feedback position in revolutions	Position of the tertiary feedback source.	Real32	0x0008	0x202D	90
RO	Position Loop	POS3C	counts	Tertiary feedback position in revolutions	Position of the tertiary feedback source.	Integer32	0x0004	0x202E	92
RO	Position Loop	T		T register	Used as a counter in UPOB programming.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Position Loop	TPOS	rev	Target position in revolutions	Output position command of the target generator to the position control loop when in position command mode CM=POS.	Real32	0x0008	0x501A	24628
RO	Position Loop	TPOSC	count	Target position in counts	Output position command of the target generator to the position control loop when in position command mode CM=POS.	Integer32	0x0004	0x501B	24630
RO	Position Loop	ZHIT		Secondary index detection flag	Least significant bit is set to 1 after a full revolution and the index pulse is observed for a second time.	Unsigned8	0x0005	0x5033	24678
RO	Position Loop	ZPOSN	rev	Negative direction index position	Absolute position of the index pulse in the negative direction.	Real32	0x0008	0x5031	24674
RO	Position Loop	ZPOSP	rev	Positive direction index position	Absolute position of the index pulse in the positive direction.	Real32	0x0008	0x5032	24676
RO	Position Loop	ZPULSE		Primary index detection flag	Least significant bit is set to 1 when the index pulse is observed for the first time.	Unsigned8	0x0005	0x5034	24680
RO	Status	DBG1		Debug register 1	Reserved.	Real32	0x0008	0x2068	208
RO	Status	DBG2		Debug register 2	Reserved.	Real32	0x0008	0x2069	210
RO	Status	DBG3		Debug register 3	Reserved.	Real32	0x0008	0x206A	212
RO	Status	DBG4		Debug register 4	Reserved.	Real32	0x0008	0x206B	214
RO	Status	ENABLED		Enable status	When the least significant bit is 1, the drive is enabled and actively controlling the motor.	Unsigned8	0x0005	0x205F	190
RO	Status	FAULTED		Fault status	When the least significant bit is 1, the drive has faulted and has not been reset.	Unsigned8	0x0005	0x2060	192

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RO	Status	FHW		Hardware fault bits	Binary code indicating a specific hardware fault condition. Value is interpreted and displayed by the IN Control software under the Diagnose tab. Bit 0 is the least-significant bit and NRE means non-resettable error (if present, return product to Allied Motion Controls for repair). Bit 0=1 -> Short Circuit Bit 1=1 -> Drive Over-Temperature Bit 2=1 -> Reserved Bit 3=1 -> Bus Under-Voltage Bit 4=1 -> Over-Current Bit 5=1 -> Reserved Bit 6=1 -> Bus Over-Voltage Bit 7=1 -> Reserved Bit 8=1 -> Motor Over-Temperature Bit 9=1 -> Reserved Bit 10=1 -> Encoder Loss Bit 11=1 -> Reserved Bit 12=1 -> Invalid Hall State Bit 13=1 -> (NRE) Bad Current Offset Bit 14=1 -> Reserved Bit 15=1 -> Reserved Bit 16=1 -> Fieldbus Watchdog Timeout Bit 17=1 -> Reserved Bit 18=1 -> Reserved Bit 19=1 -> Reserved Bit 20=1 -> Reserved Bit 21=1 -> Locked Rotor Bit 22=1 -> (NRE) Oscillator Failure Bit 23=1 -> (NRE) Invalid ID Configuration Bit 24=1 -> Reserved Bit 25=1 -> Reserved Bit 26=1 -> Reserved Bit 27=1 -> Reserved Bit 28=1 -> Reserved Bit 29=1 -> Reserved Bit 30=1 -> Reserved Bit 31=1 -> Reserved	Unsigned32	0x0007	0x2063	198

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RO	Status	FSW		Software fault bits	Binary code indicating a specific software fault condition. Value is interpreted and displayed by the IN Control software under the Diagnose tab. Bit 0 is the least-significant bit and NRE means non-resettable error (if present, return product to Allied Motion Controls for repair). Bit 0=1 -> Reserved Bit 1=1 -> Reserved Bit 2=1 -> Reserved Bit 3=1 -> Reserved Bit 4=1 -> Reserved Bit 5=1 -> Reserved Bit 6=1 -> Reserved Bit 7=1 -> Reserved Bit 8=1 -> Reserved Bit 9=1 -> Reserved Bit 10=1 -> Reserved Bit 11=1 -> Reserved Bit 12=1 -> Reserved Bit 13=1 -> Reserved Bit 14=1 -> Reserved Bit 15=1 -> Reserved Bit 16=1 -> Reserved Bit 17=1 -> Reserved Bit 18=1 -> Reserved Bit 19=1 -> Reserved Bit 20=1 -> Reserved Bit 21=1 -> Reserved Bit 22=1 -> Reserved Bit 23=1 -> Reserved Bit 24=1 -> Reserved Bit 25=1 -> Reserved Bit 26=1 -> Reserved Bit 27=1 -> Reserved Bit 28=1 -> Reserved Bit 29=1 -> Reserved Bit 30=1 -> Reserved Bit 31=1 -> Reserved	Unsigned32	0x0007	0x2062	196
RO	Status	OCMV	A	Last over-current trip	Holds the last over-current value. If an over-current fault is set to trip at 48 A and the feedback current IQ actually achieves 49 A, OCMV will read 49 A.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Status	OVERTRQ		Over torque limit	If the least significant bit is 1, then the torque output command TRQC has reached the torque limit threshold defined by TLIM.	Unsigned8	0x0005	0x2061	194
RO	Status	REGP	W	Regeneration power	Power flowing through regenerative circuitry.	Real32	0x0008	0x2026	76
RO	Status	TPWR	°C	Power stage temperature	Measured temperature of the drive power stage circuitry.	Real32	0x0008	0x2019	50
RO	Status	TRAP		Active phase indicator	When driving a motor using trapezoidal (six-step) commutation (e.g. when hall only drive mode DM=BLH), TRAP indicates which motor phases are active. For example "CA" indicates that Phase C is at high potential, Phase A is at low potential, and Phase B is inactive.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Status	UVMV	V	Last under-voltage trip	Holds the last under-voltage value. If an under-voltage fault is set to trip at 10.0 V and DC bus voltage VDC actually achieves 9.8 V, UVMV will read 9.8 V.	Not accessible	Not accessible	Not accessible	Not accessible
RO	Status	VBUS	V	DC bus voltage	The DC bus voltage as measured by the drive.	Real32	0x0008	0x2025	74
RO	Velocity Loop	RPM	RPM	Scaled velocity feedback	Equivalent to the filtered velocity feedback VELF, scaled to units of RPM.	Real32	0x0008	0x2032	100

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RO	Velocity Loop	TRQC	N-m	Torque command	When operating the drive in torque command mode CM=TRQ, the command value CV is equal to TRQC. When operating in velocity or position modes, CM=VEL and CM=POS respectively, TRQC is the output of the velocity control loop. TRQC is scaled by the motor torque constant KT to become the torque producing current reference IQREF of the current control loop.	Real32	0x0008	0x2035	106
RO	Velocity Loop	VCMD	rev/s	Velocity command	Target reference for the motor velocity control loop. When operating the drive in velocity mode CM=VEL, the command value CV is equal to VCMD. When in position mode CM=POS, VCMD is the output of the position control loop.	Real32	0x0008	0x2034	104
RO	Velocity Loop	VEL	rev/s	Velocity feedback	Motor velocity derived from commutation feedback source.	Real32	0x0008	0x2030	96
RO	Velocity Loop	VELF	rev/s	Time averaged velocity feedback	Time averaged motor velocity derived from commutation feedback source.	Real32	0x0008	0x2031	98
RO	Velocity Loop	VERR	rev/s	Velocity error	Difference between the velocity command VCMD and the velocity feedback VEL. When the drive is operating properly, the steady state value is driven to zero.	Real32	0x0008	0x2033	102
RW	Auxiliary PID Loop	P.ASG		Analog setpoint gain	The analog input signal ADC1 (-10 V to +10 V) is multiplied by 1/10 so that the resultant range is -1.0 to 1.0. P.ASG scales this signal to user defined units.	Real32	0x0008	0x2100	512
RW	Auxiliary PID Loop	P.DFC	Hz	Derivative low-pass filter cutoff frequency	Used to eliminate high frequency noise from the derivative term in the Auxiliary PID control loop.	Real32	0x0008	0x210E	540
RW	Auxiliary PID Loop	P.DISC		Disabled command output	This value is the command input VCMD to the velocity control loop, when the Auxiliary PID control loop is disabled.	Real32	0x0008	0x2115	554
RW	Auxiliary PID Loop	P.EHL		Error high limit	Imposes upper bound on the error between the setpoint P.SP and feedback P.FB signals. The resulting signal P.ERR is the input to the Auxiliary PID control loop.	Real32	0x0008	0x2109	530
RW	Auxiliary PID Loop	P.ELL		Error low limit	Imposes lower bound on the error between the setpoint P.SP and feedback P.FB signals. The resulting signal P.ERR is the input to the Auxiliary PID control loop.	Real32	0x0008	0x210A	532
RW	Auxiliary PID Loop	P.EN		Auxiliary PID Enable	Bit value which enables the Auxiliary PID control loop. Note that the drive must also be set in velocity command mode DM=VEL for the Auxiliary PID control loop to operate. This is because the output of the Auxiliary PID control loop P.OUT is added to the digital command value DCV entered into the velocity control loop.	Unsigned8	0x0005	0x2106	524
RW	Auxiliary PID Loop	P.FBG		Feedback gain	The feedback signal ADC2 (-10 V to +10 V) is multiplied by 1/10 so that the resultant range is -1.0 to 1.0. The alternative feedback source A420 has a range of 0.0 to 1.0. P.FBG scales the feedback signal (chosen by P.FBS) to user defined units.	Real32	0x0008	0x2103	518
RW	Auxiliary PID Loop	P.FBO		Feedback offset	Offset added to the scaled feedback signal.	Real32	0x0008	0x2104	520
RW	Auxiliary PID Loop	P.FBS		Feedback source	Use to select the feedback source. P.FBS=0 -> ADC2 P.FBS=1 -> A420	Unsigned8	0x0005	0x2105	522
RW	Auxiliary PID Loop	P.FHL		Feedback high limit	Imposes upper bound on the scaled and offset feedback signal.	Real32	0x0008	0x2107	526
RW	Auxiliary PID Loop	P.FLL		Feedback low limit	Imposes lower bound on the scaled and offset feedback signal.	Real32	0x0008	0x2108	528
RW	Auxiliary PID Loop	P.IHL		Integrator high limit	Imposes upper bound on the output of the integrator in the Auxiliary PID control loop.	Real32	0x0008	0x210F	542
RW	Auxiliary PID Loop	P.ILL		Integrator low limit	Imposes lower bound on the output of the integrator in the Auxiliary PID control loop.	Real32	0x0008	0x2110	544
RW	Auxiliary PID Loop	P.KD		Auxiliary PID derivative gain	Tune Auxiliary PID control loop parameter for desired output. Tuning procedures include standard methods such as Ziegler–Nichols.	Real32	0x0008	0x210D	538
RW	Auxiliary PID Loop	P.KI		Auxiliary PID integral gain	Tune Auxiliary PID control loop parameter for desired output. Tuning procedures include standard methods such as Ziegler–Nichols.	Real32	0x0008	0x210C	536
RW	Auxiliary PID Loop	P.KP		Auxiliary PID proportional gain	Tune Auxiliary PID control loop parameter for desired output. Tuning procedures include standard methods such as Ziegler–Nichols.	Real32	0x0008	0x210B	534
RW	Auxiliary PID Loop	P.OFF		Output offset	Offset added to the scaled pre-limited output of the Auxiliary PID control loop.	Real32	0x0008	0x2112	548
RW	Auxiliary PID Loop	P.OG		Output gain	Scale factor applied to the pre-limited output of the Auxiliary PID control loop.	Real32	0x0008	0x2111	546
RW	Auxiliary PID Loop	P.OHL		Output high limit	Imposes upper bound on the pre-limited output of the Auxiliary PID control loop.	Real32	0x0008	0x2113	550
RW	Auxiliary PID Loop	P.OLL		Output low limit	Imposes lower bound on the pre-limited output of the Auxiliary PID control loop.	Real32	0x0008	0x2114	552

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Auxiliary PID Loop	P.PER		Auxiliary PID period multiplier	Integer defining the cycle time of the Auxiliary PID control loop. When P.PER=1 control loop updates once every 1 ms, when P.PER=2 loop updates every 2 ms, etc. Range of permitted values is 1 to 1000.	Integer16	0x0003	0x2116	556
RW	Auxiliary PID Loop	P.SETP		Digital setpoint	Digital setpoint is summed with scaled and limited input from ADC1 to form the total process setpoint. Digital setpoint can be used to offset signal from ADC1, or used as the sole input source by zeroing the gain applied to ADC1 (i.e. let P.ASG=0). This parameter is volatile.	Real32	0x0008	0x2117	558
RW	Auxiliary PID Loop	P.SHL		Analog setpoint high limit	Imposes upper bound on the signal from ADC1.	Real32	0x0008	0x2101	514
RW	Auxiliary PID Loop	P.SLL		Analog setpoint low limit	Imposes lower bound on the signal from ADC1.	Real32	0x0008	0x2102	516
RW	Auxiliary PID Loop	P.SRL		Set point rate limit	Imposes rate limit on the summation of the digital setpoint P.SETP and the scaled and limited analog setpoint parameter ADC1.	Real32	0x0008	0x211F	574
RW	Command	ADEAD	V	Analog input deadband	Deadband applied to analog input command signal selected by AIS.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Command	AIS		Select analog input source	Use to select source of raw command signal RCV when in analog command mode CM=ANA: AIS=0 -> ADC1 AIS=1 -> ADC2 AIS=2 -> ADC3 AIS=3 -> ADC4	Unsigned8	0x0005	0x205A	180
RW	Command	ARL	unit/s	Positive rate limit	Imposes positive rate limit on the limited command value LCV to produce the command value CV. If in velocity command mode CM=VEL then ARL limits the commanded rate of acceleration.	Real32	0x0008	0x2058	176
RW	Command	CLH		High command limit value	Imposes upper bound on the raw command value RCV to produce the limited command value LCV.	Real32	0x0008	0x2056	172
RW	Command	CLL		Low command limit value	Imposes lower bound on the raw command value RCV to produce the limited command value LCV.	Real32	0x0008	0x2057	174
RW	Command	CM		Command mode	Select the drive command mode: CM=0 -> CUR In current command mode, the command value CV becomes the torque producing motor current IQREF. CM=1 -> TRQ In torque command mode, the command value CV becomes the torque command TRQC. CM=2 -> VEL In velocity command mode, the command value CV becomes the velocity command VCMD. CM=3 -> POS In position command mode, the drive's target generator is active. Given register inputs (R0, R1, R2, or RUNMS, STPMS, REG00, REG01, etc.) to a UPOB loaded into memory (activated by motion sequence auto-run EMS), the target generator outputs a target position TPOS to the position control loop. CM=4 -> OLV In open-loop velocity command mode, the command value CV becomes the torque producing voltage demand VQ, while the flux producing voltage demand VD is held at zero.	Unsigned8	0x0005	0x2064	200

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Command	CVCS		Command variable control switch	Select the drive command source: CVCS=0 -> 0 Sets the raw command value RCV to zero. CVCS=1 -> COM Sets the raw command value RCV to CCV. CVCS=2 -> DCV Sets the raw command value RCV to DCV. CVCS=3 -> ANA Sets the raw command value RCV to be an analog input. Select the specific analog input using AIS. CVCS=4 -> FG Sets the raw command value RCV to be the function generator output. The function generator outputs a square-wave signal defined by parameters FGC1, FGC2, FGT1, and FGT2. This mode can be useful in commissioning and tuning of the motor/drive system. CVCS=5 -> PW Reserved.	Unsigned8	0x0005	0x2055	170
RW	Command	DCV		Digital command value	When CVCS=DCV, DCV is the input that corresponds to the raw command value RCV. The units of DCV are determined by the chosen command mode CM. For use with fieldbus communications other than Ethernet (e.g. RS-232, RS-485, or CAN). DCV is intentionally volatile (DCV=0 at power-up).	Real32	0x0008	0x204E	156
RW	Command	DRL	unit/sec	Negative rate limit	Imposes negative rate limit on the limited command value LCV to produce the command value CV. If in velocity command mode CM=VEL then DRL limits the commanded rate of deceleration.	Real32	0x0008	0x2059	178
RW	Command	EM		Drive enable mode	Select source of the drive enable signal: EM=0 -> SWE Software enable mode allows software to enable and disable the drive. In this mode the parameter EN is set to ON or OFF to enable or disable the drive. EN can be set through the IN Control user interface or through fieldbus communications. EM=1 -> HWE Hardware enable mode allows one of the drive's digital inputs to enable and disable the drive. This digital input selected by assigning a value to the hardware enable input ENI. EM=2 -> OR The drive is enabled by a logical OR of the hardware enable bit and the software enable parameter EN. EM=3 -> AND The drive is enabled by a logical AND of the hardware enable bit and the software enable parameter EN.	Unsigned8	0x0005	0x2066	204
RW	Command	EMS		Motion sequence auto run	Bit value which enables a stored motion program to execute when the drive is enabled. Parameter is only used when in position command mode CM=POS.	Integer16	0x0003	0x202F	94

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Command	EN		Software drive enable	Bit value which enables the drive when in software enable mode EM=SWE.	Unsigned8	0x0005	0x205D	186
RW	Command	ENI		Hardware enable input	Select digital input (1 to 32) to enable the drive when in hardware enable mode EM=HWE.	Integer16	0x0003	0x2067	206
RW	Command	FGC1	unit	Function generator constant amplitude 1	Constant amplitude value of the first level of the square-wave signal. Units are defined by the command mode CM.	Real32	0x0008	0x204A	148
RW	Command	FGC2	unit	Function generator constant amplitude 2	Constant amplitude value of the second level of the square-wave signal. Units are defined by the command mode CM.	Real32	0x0008	0x204C	152
RW	Command	FGT1	s	Function generator time duration 1	Duration of time at constant amplitude FGC1.	Real32	0x0008	0x2049	146
RW	Command	FGT2	s	Function generator time duration 2	Duration of time at constant amplitude FGC2.	Real32	0x0008	0x204B	150
RW	Command	FSPW		Full scale PWM command value	Reserved.	Real32	0x0008	0x205C	184
RW	Command	FSV		Full scale analog command value	When CVCS=ANA, FSV is used to scale the analog input command signal. It corresponds to the desired command value when the analog input is 10 V.	Real32	0x0008	0x205B	182
RW	Command	R0		Register 0	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5021	24642
RW	Command	R1		Register 1	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5022	24644
RW	Command	R10		Register 10	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x502B	24662
RW	Command	R11		Register 11	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x502C	24664
RW	Command	R12		Register 12	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x502D	24666
RW	Command	R13		Register 13	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x502E	24668
RW	Command	R14		Register 14	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x502F	24670
RW	Command	R15		Register 15	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5030	24672
RW	Command	R2		Register 2	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5023	24646
RW	Command	R3		Register 3	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5024	24648
RW	Command	R4		Register 4	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5025	24650
RW	Command	R5		Register 5	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5026	24652
RW	Command	R6		Register 6	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5027	24654
RW	Command	R7		Register 7	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5028	24656
RW	Command	R8		Register 8	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5029	24658
RW	Command	R9		Register 9	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x502A	24660
RW	Command	REG00		Volatile memory register 0	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5021	24642

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Command	REG01		Volatile memory register 1	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5022	24644
RW	Command	REG02		Volatile memory register 2	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5023	24646
RW	Command	REG03		Volatile memory register 3	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5024	24648
RW	Command	REG04		Volatile memory register 4	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5025	24650
RW	Command	REG05		Volatile memory register 5	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5026	24652
RW	Command	REG06		Volatile memory register 6	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5027	24654
RW	Command	REG07		Volatile memory register 7	Floating point memory storage available for motion sequences. Register values are stored in volatile memory (set to 0 on power-up).	Real32	0x0008	0x5028	24656

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Command	RUNMS		Run motion sequence.	A non-zero value starts a motion sequence: RUNMS=1 A single incremental index is performed. The motion sequence stops once the index is complete. REG00 is displacement (rev) from marker, REG01 is acceleration/deceleration rate in (rev/s^2), REG02 is maximum velocity in (rev/s). RUNMS=2 An incremental index is performed on every rise of digital input 1. REG00 is displacement (rev) from marker, REG01 is acceleration/deceleration rate in (rev/s^2), REG02 is maximum velocity in (rev/s). RUNMS=3 Zero's the target position maintaining following error. RUNMS=4 Motor is positioned relative to the encoder "Z" marker pulse. REG00 is displacement (rev) from marker, REG01 is acceleration/deceleration rate in (rev/s^2), REG02 is maximum velocity in (rev/s). RUNMS=5 Reserved. RUNMS=6 Reserved. RUNMS=7 Enters PV mode which allows the user to specify a position to move to on-the-fly together with a maximum velocity and acceleration/deceleration rate. REG00 is displacement (rev) from marker, REG01 is acceleration/deceleration rate in (rev/s^2), REG02 is maximum velocity in (rev/s). RUNMS=8 Exits PV mode and stops motion. RUNMS=9 A repeated out and back incremental index is performed with a delay after completion of each index. The motion sequence continues to execute unless the drive is disabled or STPMS is set to 1. REG00 is displacement (rev) from marker, REG01 is acceleration/deceleration rate in (rev/s^2), REG02 is maximum velocity in (rev/s), REG03 is delay time (ms).	Integer16	0x0003	0x20AA	340
RW	Command	STPMS		Stop motion sequence	Setting this bit to 1 stops a running motion sequence. STPMS is automatically set to 0 once the motion sequence has stopped. This parameter is intentionally volatile (STPMS=0 on power-up).	Integer16	0x0003	0x20AB	342
RW	Commutation	C1.INVE		Invert primary encoder direction	Bit value reverses the direction of the hardware counter for the primary feedback source FBC. When primary feedback is an incremental encoder plus hall sensors (or commutation track), for proper motor commutation FBC and the hall commutation angle HANG must proceed in the same direction. To confirm this is so, set drive operating mode to hall only DM=BLH, and confirm that FBC and HANG proceed in the same direction. If this is not the case, use the invert bit to make it so. C1.INVE is ignored when in hall operating mode DM=BLH. Be sure to set the inversion bits in the following order: commutation offset COFF and C1.INVH, then C1.ITRQ, C1.INVE, C1.INVF.	Unsigned32	0x0007	0x2071	226

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Commutation	C1.INVF		Invert velocity/position feedback direction	Bit value reverses the direction of the velocity and position feedback parameters, VEL and FPOS respectively. For proper motor control when in velocity or position command modes, CM=VEL or CM=POS, a positive torque producing current reference IQREF must correspond to a positive change (acceleration) in the velocity feedback VEL. If this is not the case, use invert bit to make it so. C1.INVF is ignored when in hall operating mode DM=BLH. Be sure to set the inversion bits in the following order: commutation offset COFF and C1.INVH, then C1.ITRQ, C1.INVE, C1.INVF.	Unsigned32	0x0007	0x2073	230
RW	Commutation	C1.INVH		Invert hall commutation angle direction	Bit value which reverses the direction of the hall commutation angle HANG. When in hall operating mode DM=BLH and current command mode CM=CUR, the no-load speed of the motor should be equivalent in both directions given a positive and negative current command of the same magnitude, and the motor feedback velocity VEL and the torque producing current command reference IQREF should always be of the same sign. If this is not the case, adjust COFF and C1.INVH to make it so. Be sure to set the inversion bits in the following order: commutation offset COFF and C1.INVH, then C1.ITRQ, C1.INVE, C1.INVF.	Unsigned32	0x0007	0x2070	224
RW	Commutation	C1.ITRQ		Invert current command direction	Bit value which reverses the direction of the torque producing current reference IQREF. Use this bit to select chosen motor direction. Be sure to set the inversion bits in the following order: commutation offset COFF and C1.INVH, then C1.ITRQ, C1.INVE, C1.INVF.	Unsigned32	0x0007	0x2072	228
RW	Commutation	COFF	deg	Commutation offset	Define the offset between the hall commutation angle HANG and the actual motor electrical angle. For most motors manufactured by Allied Motion, COFF should be zero; successful commutation of an Allied Motion motor using an offset angle which is some multiple of 120 degrees likely indicates that the hall sensor wiring was made out of order. For all motors, inaccurate values of COFF will result in decreased motor efficiency and/or problems controlling the motor. When in hall operating mode DM=BLH and current command mode CM=CUR, the no-load speed of the motor should be equivalent in both directions given a positive and negative current command of the same magnitude, and the motor feedback velocity VEL and the torque producing current command reference IQREF should always be of the same sign. If this is not the case, adjust COFF and C1.INVH to make it so.	Real32	0x0008	0x2037	110

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Commutation	DM		Drive operating mode	Select feedback method for motor commutation: DM=0 -> VF In variable frequency mode, the drive outputs a fixed-frequency, constant-amplitude, sinusoidal current to the motor. The amplitude is defined by the command value CV, and its frequency is defined by the parameter VFF. The motor current is independent of the motor position. This mode is used for testing and is almost never used in actual applications. DM=1 -> BLH In brushless hall mode, the motor is commutated using hall sensor position feedback only. If utilizing a motor with hall only feedback, an encoder is not required. Valid command modes are limited to current CM=CUR, torque CM=TRQ, or velocity CM=VEL. Note that slow speed operation (e.g. under 10 rev/s) will result in poor speed control due to the inadequate resolution provided by the hall sensors at low speeds. DM=2 -> BLE In brushless encoder mode, the motor is commutated using both hall sensor information and encoder feedback. All command modes are available. DM=3 -> BLS In brushless sensorless mode, the motor is commutated using position information estimated from the backEMF voltage of the motor as it rotates. Because the strength of the backEMF signal depends on the motor velocity, there is a lower bound on motor velocity below which the motor will not commutate. Valid command modes are limited to current CM=CUR, torque CM=TRQ, or velocity CM=VEL. DM=4 -> BLR In brushless resolver mode, the motor is commutated using resolver position feedback. All command modes are available. DM=5 -> BSC In brushless sin/cos mode, the motor is commutated using both hall sensor information and high resolution position feedback from a sin/cos encoder. All command modes are available. DM=6 -> BRU In brushed mode, the drive assumes the presence of a mechanical commutator. Phase C is deactivated and Phase A and B are utilized to control single-phase motor current. All command modes are available. DM=7 -> STEP Reserved. DM=8 -> BLE_NH In brushless encoder no halls mode, the motor is commutated using encoder feedback only. Note that on the first drive enable after power-up, the motor will move slightly to establish alignment between the encoder feedback and motor electrical angle, after which operation proceeds as normal. All command modes are available.	Unsigned8	0x0005	0x2065	202
RW	Commutation	EPPR	count/rev	Encoder counts per revolution	Enter the resolution (number of observable positions per motor revolution) of the primary encoder.	Integer32	0x0004	0x2038	112

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Commutation	INTBITS		Sin/cos encoder interpolation degree	Feedback from a sin/cos encoder is processed through an analog to digital converter. The number of possible values output from the converter is defined as 2^INTBITS. The range of INTBITS is 0 to 12. Therefore, setting this parameter to zero indicates no interpolation and setting INTBITS to 12 results in the highest level of interpolation (4096 points). The total position feedback resolution is thus calculated as follows: multiply the number of encoder lines by four, in order to account for quadrature, to obtain the number of encoder counts per revolution EPPR. EPPR is then multiplied by the number of interpolated points (2^INTBITS) to obtain the total number of counts per revolution.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Commutation	PENCT		Primary encoder type	Declare output format of primary encoder (e.g. quadrature, sin/cos, Endat, Biss).	Not accessible	Not accessible	Not accessible	Not accessible
RW	Commutation	RPOL		Resolver pole count	Can be an even number up to 16. Parameter is required to be set if DM=BLR.	Integer16	0x0003	0x203F	126
RW	Commutation	SENCT		Secondary encoder type	Declare output format of secondary encoder (e.g. quadrature, sin/cos, Endat, Biss).	Not accessible	Not accessible	Not accessible	Not accessible
RW	Commutation	TENCT		Tertiary encoder type	Declare output format of tertiary encoder (e.g. quadrature, sin/cos, Endat, Biss).	Not accessible	Not accessible	Not accessible	Not accessible
RW	Commutation	VFF	Hz	Open-loop commutation frequency	This is only used when DM=VF. This parameter is intentionally volatile (VFF=0 at power-up).	Real32	0x0008	0x205E	188
RW	Current Loop	CALIFA		Phase A current feedback calibration	Reserved.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Current Loop	CALIFB		Phase B current feedback calibration	Reserved.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Current Loop	CALIFC		Phase C current feedback calibration	Reserved.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Current Loop	CKID	ohm/s	Flux producing current integral gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Current Loop	CKIQ	ohm/s	Torque producing current integral gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Current Loop	CKPD	ohm	Flux producing current proportional gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Current Loop	CKPQ	ohm	Torque producing current proportional gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Current Loop	IKID	ohm/s	Flux producing current integral gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.	Real32	0x0008	0x203A	116
RW	Current Loop	IKIQ	ohm/s	Torque producing current integral gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.	Real32	0x0008	0x203C	120
RW	Current Loop	IKPD	ohm	Flux producing current proportional gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.	Real32	0x0008	0x2039	114
RW	Current Loop	IKPQ	ohm	Torque producing current proportional gain parameter	Current loop gain is based on motor terminal resistance RM and inductance LM. Having input accurate values for each, gain value is set using the "Calc I-Loop Gains" function of IN Control.	Real32	0x0008	0x203B	118
RW	Fieldbus Configuration	CANBR	kbps	CAN baud rate	Select baud rate of the CAN network.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Fieldbus Configuration	CANID		CAN node identification number	Assign CAN node ID to drive.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Fieldbus Configuration	MODID		Modbus identification number	Assign Modbus node ID to drive.	Not accessible	Not accessible	Not accessible	Not accessible
RW	I/O	BDTM	s	Brake delay	Normally the brake engages at zero speed after disable. BDTM defines a time interval at drive disable, after which the brake will engage even if motor speed is still non-zero.	Not accessible	Not accessible	Not accessible	Not accessible

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	I/O	BFTM	s	Brake faulted delay	Normally the brake engages at zero speed after disable. BDTM defines a time interval at drive fault, after which the brake will engage even if motor speed is still non-zero.	Not accessible	Not accessible	Not accessible	Not accessible
RW	I/O	BINV		Brake invert	Bit value inverts the operation of the brake (i.e. active power ON, or active power OFF).	Not accessible	Not accessible	Not accessible	Not accessible
RW	I/O	BKOV		Brake override	Override disables all brake functionality.	Unsigned8	0x0005	0x20C0	384
RW	I/O	BOUT		Brake output assignment	Defines which output controls the brake (BOUT=0 indicates no brake output assigned).	Not accessible	Not accessible	Not accessible	Not accessible
RW	I/O	BVLT	V	Brake voltage	Defines the voltage output by the drive to the brake. For example, setting BVLT=12 allows a 12 V brake to be used with a 24 V supplied drive.	Not accessible	Not accessible	Not accessible	Not accessible
RW	I/O	CAL1	V	Calibration offset for ADC1	Use to offset voltage losses between the signal source and analog input. Note that CAL1 cannot be saved to a drive configuration (.dcf) file using IN Control. Therefore when loading a configuration file, CAL1 will not be set and must be adjusted manually.	Real32	0x0008	0x2010	32
RW	I/O	CAL2	V	Calibration offset for ADC2	Use to offset voltage losses between the signal source and analog input. Note that CAL2 cannot be saved to a drive configuration (.dcf) file using IN Control. Therefore when loading a configuration file, CAL2 will not be set and must be adjusted manually.	Real32	0x0008	0x2011	34
RW	I/O	CAL3	V	Calibration offset for ADC3	Use to offset voltage losses between the signal source and analog input. Note that CAL3 cannot be saved to a drive configuration (.dcf) file using IN Control. Therefore when loading a configuration file, CAL3 will not be set and must be adjusted manually.	Real32	0x0008	0x2012	36
RW	I/O	CAL4	V	Calibration offset for ADC4	Use to offset voltage losses between the signal source and analog input. Note that CAL4 cannot be saved to a drive configuration (.dcf) file using IN Control. Therefore when loading a configuration file, CAL4 will not be set and must be adjusted manually.	Real32	0x0008	0x2013	38
RW	I/O	COUTS		Clear digital outputs	Clears output bits that were previously forced on with the force outputs parameter FOUTS. This parameter is volatile and cleared on power-up.	Unsigned32	0x0007	0x20A9	338
RW	I/O	DAC1		Digital to analog converter 1	Select variable to output to analog output 1.	Not accessible	Not accessible	Not accessible	Not accessible
RW	I/O	DACMAX1		Digital to analog converter 1 maximum	Define the value of DAC1 which is to be the maximum analog output voltage.	Real32	0x0008	0x20C1	386
RW	I/O	DACMIN1		Digital to analog converter 1 minimum	Define the value of DAC1 which is to be the minimum analog output voltage.	Real32	0x0008	0x20C2	388
RW	I/O	FOUTS		Force outputs	Forces on and latches one or more of the digital outputs. Unless deactivated by the drive firmware or by clear outputs COUTS, the output will remain active.	Unsigned32	0x0007	0x20A8	336
RW	I/O	INAND.1		Digital input 1 AND bit	Value can be either 0 or 1. The state of digital input 1 is ANDed with this parameter. If INAND.1=0, digital input 1 is effectively disabled regardless of the signal applied across digital input 1.	Unsigned32	0x0007	0x2090	288
RW	I/O	INAND.10		Digital input 10 AND bit	Value can be either 0 or 1. The state of digital input 10 is ANDed with this parameter. If INAND.10=0, digital input 10 is effectively disabled regardless of the signal applied across digital input 10.	Unsigned32	0x0007	0x20A2	324
RW	I/O	INAND.11		Digital input 11 AND bit	Value can be either 0 or 1. The state of digital input 11 is ANDed with this parameter. If INAND.11=0, digital input 11 is effectively disabled regardless of the signal applied across digital input 11.	Unsigned32	0x0007	0x20A4	328
RW	I/O	INAND.12		Digital input 12 AND bit	Value can be either 0 or 1. The state of digital input 12 is ANDed with this parameter. If INAND.12=0, digital input 12 is effectively disabled regardless of the signal applied across digital input 12.	Unsigned32	0x0007	0x20A6	332
RW	I/O	INAND.2		Digital input 2 AND bit	Value can be either 0 or 1. The state of digital input 2 is ANDed with this parameter. If INAND.2=0, digital input 2 is effectively disabled regardless of the signal applied across digital input 2.	Unsigned32	0x0007	0x2092	292
RW	I/O	INAND.3		Digital input 3 AND bit	Value can be either 0 or 1. The state of digital input 3 is ANDed with this parameter. If INAND.3=0, digital input 3 is effectively disabled regardless of the signal applied across digital input 3.	Unsigned32	0x0007	0x2094	296

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	I/O	INAND.4		Digital input 4 AND bit	Value can be either 0 or 1. The state of digital input 4 is ANDed with this parameter. If INAND.4=0, digital input 4 is effectively disabled regardless of the signal applied across digital input 4.	Unsigned32	0x0007	0x2096	300
RW	I/O	INAND.5		Digital input 5 AND bit	Value can be either 0 or 1. The state of digital input 5 is ANDed with this parameter. If INAND.5=0, digital input 5 is effectively disabled regardless of the signal applied across digital input 5.	Unsigned32	0x0007	0x2098	304
RW	I/O	INAND.6		Digital input 6 AND bit	Value can be either 0 or 1. The state of digital input 6 is ANDed with this parameter. If INAND.6=0, digital input 6 is effectively disabled regardless of the signal applied across digital input 6.	Unsigned32	0x0007	0x209A	308
RW	I/O	INAND.7		Digital input 7 AND bit	Value can be either 0 or 1. The state of digital input 7 is ANDed with this parameter. If INAND.7=0, digital input 7 is effectively disabled regardless of the signal applied across digital input 7.	Unsigned32	0x0007	0x209C	312
RW	I/O	INAND.8		Digital input 8 AND bit	Value can be either 0 or 1. The state of digital input 8 is ANDed with this parameter. If INAND.8=0, digital input 8 is effectively disabled regardless of the signal applied across digital input 8.	Unsigned32	0x0007	0x209E	316
RW	I/O	INAND.9		Digital input 9 AND bit	Value can be either 0 or 1. The state of digital input 9 is ANDed with this parameter. If INAND.9=0, digital input 9 is effectively disabled regardless of the signal applied across digital input 9.	Unsigned32	0x0007	0x20A0	320
RW	I/O	INOR.1		Digital input 1 OR bit	Value can be either 0 or 1. The state of digital input 1 is ORed with this parameter. If INOR.1=1, digital input 1 is effectively enabled regardless of the signal applied across digital input 1.	Unsigned32	0x0007	0x2091	290
RW	I/O	INOR.10		Digital input 10 OR bit	Value can be either 0 or 1. The state of digital input 10 is ORed with this parameter. If INAND.10=0, digital input 10 is effectively enabled regardless of the signal applied across digital input 10.	Unsigned32	0x0007	0x20A3	326
RW	I/O	INOR.11		Digital input 11 OR bit	Value can be either 0 or 1. The state of digital input 11 is ORed with this parameter. If INAND.11=0, digital input 11 is effectively enabled regardless of the signal applied across digital input 11.	Unsigned32	0x0007	0x20A5	330
RW	I/O	INOR.12		Digital input 12 OR bit	Value can be either 0 or 1. The state of digital input 12 is ORed with this parameter. If INAND.12=0, digital input 12 is effectively enabled regardless of the signal applied across digital input 12.	Unsigned32	0x0007	0x20A7	334
RW	I/O	INOR.2		Digital input 2 OR bit	Value can be either 0 or 1. The state of digital input 2 is ORed with this parameter. If INAND.2=0, digital input 2 is effectively enabled regardless of the signal applied across digital input 2.	Unsigned32	0x0007	0x2093	294
RW	I/O	INOR.3		Digital input 3 OR bit	Value can be either 0 or 1. The state of digital input 3 is ORed with this parameter. If INAND.3=0, digital input 3 is effectively enabled regardless of the signal applied across digital input 3.	Unsigned32	0x0007	0x2095	298
RW	I/O	INOR.4		Digital input 4 OR bit	Value can be either 0 or 1. The state of digital input 4 is ORed with this parameter. If INAND.4=0, digital input 4 is effectively enabled regardless of the signal applied across digital input 4.	Unsigned32	0x0007	0x2097	302
RW	I/O	INOR.5		Digital input 5 OR bit	Value can be either 0 or 1. The state of digital input 5 is ORed with this parameter. If INAND.5=0, digital input 5 is effectively enabled regardless of the signal applied across digital input 5.	Unsigned32	0x0007	0x2099	306
RW	I/O	INOR.6		Digital input 6 OR bit	Value can be either 0 or 1. The state of digital input 6 is ORed with this parameter. If INAND.6=0, digital input 6 is effectively enabled regardless of the signal applied across digital input 6.	Unsigned32	0x0007	0x209B	310
RW	I/O	INOR.7		Digital input 7 OR bit	Value can be either 0 or 1. The state of digital input 7 is ORed with this parameter. If INAND.7=0, digital input 7 is effectively enabled regardless of the signal applied across digital input 7.	Unsigned32	0x0007	0x209D	314
RW	I/O	INOR.8		Digital input 8 OR bit	Value can be either 0 or 1. The state of digital input 8 is ORed with this parameter. If INAND.8=0, digital input 8 is effectively enabled regardless of the signal applied across digital input 8.	Unsigned32	0x0007	0x209F	318
RW	I/O	INOR.9		Digital input 9 OR bit	Value can be either 0 or 1. The state of digital input 9 is ORed with this parameter. If INAND.9=0, digital input 9 is effectively enabled regardless of the signal applied across digital input 9.	Unsigned32	0x0007	0x20A1	322
RW	I/O	MTB	°K	Motor thermistor beta	Is the beta parameter of an NTC thermistor. If no temperature measuring device is available, set this parameter to 4288 to avoid nuisance over-temperature faults.	Real32	0x0008	0x2046	140

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	I/O	MTOT	°C	Motor thermistor over-temperature limit	Defines the threshold at which a motor over-temperature fault will be produced. The detection algorithm assumes input from an NTC type thermistor which obeys the beta parameter equation. If no temperature measuring device is available on the motor, enter a value of 800 to avoid nuisance over-temperature faults.	Real32	0x0008	0x2047	142
RW	I/O	MTR0	ohm	Motor thermistor resistance at 25C	Is the R0 parameter of an NTC thermistor. If no temperature measuring device is available, set this parameter to 50000 to avoid nuisance over-temperature faults.	Real32	0x0008	0x2045	138
RW	I/O	SPD1	rev/s	Speed detect setpoint 1	When the speed (magnitude of the velocity) of the motor is greater than or equal to SPD1, the assigned speed-detect digital output will activate. Assign the speed detect digital output using the I/O Config (In) function of IN Control.	Real32	0x0008	0x2053	166
RW	I/O	SPD2	rev/s	Speed detect setpoint 2	When the speed (magnitude of the velocity) of the motor is greater than or equal to SPD2, the assigned speed-detect digital output will activate. Assign the speed detect digital output using the I/O Config (In) function of IN Control.	Real32	0x0008	0x2054	168
RW	Motor	IMAX	A	Motor continuous rated current	Motor continuous rated current is defined in ADC when trapezoidally commutating (six-step commutation) or as the peak of the sine wave when commutating sinusoidally. The lesser of IMAX and the drive's continuous rated current are used with the drive's I^2*T current limiting scheme to limit the average current in the motor and drive. The output of the I^2*T limiting scheme is the current limit parameter ILIM.	Real32	0x0008	0x2040	128
RW	Motor	INER	kg-m^2	Total motor + load inertia	This parameter is used to tune the velocity control loop when the advanced tuning switch is on VLAT=ON.	Real32	0x0008	0x2044	136
RW	Motor	KT	N-m/A	Motor torque constant	Used to scale the torque command TRQC to the torque producing motor current reference IQREF.	Real32	0x0008	0x2043	134
RW	Motor	LM	mH	Terminal motor inductance	Determined by measuring the inductance between any two of the motor leads with the third lead open. LM is used in the automatic calculation of current loop gains using the "Calc I-Loop Gains" function of IN Control.	Real32	0x0008	0x2041	130
RW	Motor	LRIL	A	Locked rotor current limit	A locked rotor is defined as when the motor is stalled and the time averaged torque producing current reference IQAVG is greater than LRIL. When this condition persists for greater than 4 seconds, a locked rotor fault is produced. To disable all Locked Rotor faults, set LRIL=0.	Real32	0x0008	0x206C	216
RW	Motor	MPOL	count	Motor pole count	Determines the number of electrical cycles per mechanical cycle. The number of electrical cycles per motor revolution is equal to the number of motor pole pairs (i.e. half the motor pole count).	Integer16	0x0003	0x203E	124
RW	Motor	RM	ohm	Terminal motor resistance	Determined by measuring the resistance between any two of the motor leads with the third lead open. RM is used in the automatic calculation of current loop gains using the Calc I-Loop Gains function of IN Control.	Real32	0x0008	0x2042	132
RW	Position Loop	INPOSW	rev	In-position window	When the absolute value of the position error PERR is less than INPOSW at the end of a move in a motion sequence, the move is considered complete. INPOSW is ignored when not in position command mode CM=POS.	Real32	0x0008	0x208A	276
RW	Position Loop	P0		Position loop register 0	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.	Real32	0x0008	0x20B0	352
RW	Position Loop	P1		Position loop register 1	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.	Real32	0x0008	0x20B1	354
RW	Position Loop	P2		Position loop register 2	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.	Real32	0x0008	0x20B2	356
RW	Position Loop	P3		Position loop register 3	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.	Real32	0x0008	0x20B3	358
RW	Position Loop	P4		Position loop register 4	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.	Real32	0x0008	0x20B4	360
RW	Position Loop	P5		Position loop register 5	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.	Real32	0x0008	0x20B5	362

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Position Loop	P6		Position loop register 6	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.	Real32	0x0008	0x20B6	364
RW	Position Loop	P7		Position loop register 7	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.	Real32	0x0008	0x20B7	366
RW	Position Loop	P8		Position loop register 8	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.	Real32	0x0008	0x20B8	368
RW	Position Loop	P9		Position loop register 9	Floating point memory storage available for motion sequences. Position loop register values are stored in non-volatile memory.	Real32	0x0008	0x20B9	370
RW	Position Loop	PFF		Position loop feed-forward gain	The second term in a summation which becomes the velocity command VCMD is the product of the time derivative of the target position TPOS and PFF. This feed-forward term provides a larger and instantaneous velocity command to the velocity loop even with a small position error. PFF is unit-less and ranges from 0 to 1.	Real32	0x0008	0x2089	274
RW	Position Loop	PPG	1000/min	Position loop proportional gain	PPG is a classic CNC-style tuning gain that is scaled in 1000/min. PPG is multiplied by the position error PERR to produce the first term in a summation which becomes the velocity loop command VCMD; the second term in this summation is the feed-forward term (see PFF). With a value of PPG=3.0 (1000/min) and PERR=0.1 (rev), the proportional term in the velocity loop command would be 300 rev/min or VCMD=5 (rev/s).	Real32	0x0008	0x2088	272
RW	Status	CWD	ms	Fieldbus watchdog	CWD serves as a millisecond timer by which to confirm active fieldbus communications. The first write to CWD, after the drive is powered, both enables and loads the watchdog timer. The timer then counts down once every millisecond until it reaches zero, at which moment the Timeout Action described below is taken. Note that once the watchdog timer is enabled, its action cannot be cancelled. CWD can take on any integer value from 1 to 10,000. Once the timer is enabled, it is the master controller's responsibility to reload CWD with a new value periodically before the timer reaches zero, else the Timeout Action will be executed. The new value transmitted can differ from the previous value, which allows for dynamic timeout management if so desired. Timeout Action: If in position command mode CM=POS, the target generator is stopped immediately, and upon reaching zero-speed, the drive is faulted with “Fieldbus Watchdog Timeout”. If in velocity command mode CM=VEL, a zero-speed command is issued, and upon reaching zero-speed, the drive is faulted with “Fieldbus Watchdog Timeout”. The drive will decelerate the motor according to the negative rate limit DRL. If in current or torque command mode CM=CUR or CM=TRQ, the drive is faulted immediately with “Fieldbus Watchdog Timeout”.	Integer32	0x0004	0x2120	576
RW	Status	FLTRSET		Fault reset	Allows faults to be cleared over fieldbus network (e.g. CAN or RS-485). Faults are cleared when parameter transitions from 0 to 1. Parameter not visible through IN Control. To reset faults through IN Control, utilize the "Reset" button on the "Diagnose" tab.	Unsigned8	0x0005	0x20C3	390
RW	Velocity Loop	KA		Torque filter constant	In certain drive models there exists a single-pole filter on the output of the velocity control loop. This filter is directly influenced by KA. KA is exponentially weighted and small changes in KA result in large changes in the filter. A value of 1.0 means the filter is bypassed, whereas values such as 0.99 and 0.95 bring in a marked response to the torque command TRQC output of the velocity loop. In general, leaving KA at 1.0 (the default value) is desired. Dropping KA by small values in highly dynamic systems may improve controllability.	Real32	0x0008	0x2004	8

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Velocity Loop	KB	N-m-s/rev	Pseudo-friction constant	Adds artificial viscous damping (pseudo-friction) to the velocity loop control system. Friction in a real physical system is usually undesired as it increases load, but it also adds stability to a control loop. By using KB, pseudo-friction can be added to the system without increase in load. Just as its real-world counterpart, KB adds a negative torque load proportional to speed and thus is scaled in N-m-s/rev.	Real32	0x0008	0x2003	6
RW	Velocity Loop	KD	N-m-s^2/rev	Velocity loop derivative gain	It is the derivative gain used in classic PID to provide damping to the system. It is rarely used in PDF and is not included in most PDF-based control loops. It is available in the velocity loop, but it is useful only when KVF is at or near 1.0 (PID control mode).	Real32	0x0008	0x2002	4
RW	Velocity Loop	KF	N-m-s/rev	Velocity loop feed-forward gain	The input velocity command VCMD is multiplied by this term and summed into the output torque command TRQC. For systems with high static-friction, adding a small amount of KF can improve response to step changes in velocity command. In applications where the velocity command is ramped, or in position modes, KF is rarely useful.	Real32	0x0008	0x2001	2
RW	Velocity Loop	KI	N-m/rev	Velocity loop integral gain	With the advanced tuning switched activated VLAT=ON, this parameter is ignored. With VLAT=OFF, this parameter controls the integral gain with PID or PDF.	Real32	0x0008	0x200B	22
RW	Velocity Loop	KV	N-m-s/rev	Velocity loop proportional gain	With the advanced tuning switched activated VLAT=ON, this parameter is ignored. With VLAT=OFF, this parameter controls the integral gain with PID or PDF.	Real32	0x0008	0x200A	20
RW	Velocity Loop	KVF		Velocity loop PID/PDF control ratio	Controls the PID/PDF control-loop ratio. A value of 1.0 selects a classic PID and a value of 0.0 selects PDF. Values in between 1.0 and 0.0 select a combination of classic PID and/or PDF control.	Real32	0x0008	0x2005	10
RW	Velocity Loop	TLIM	N-m	Velocity loop torque command limit	Limits the output of the velocity control loop at a given absolute torque value. The motor torque constant KT must be set accurately for TLIM to function properly.	Real32	0x0008	0x2006	12
RW	Velocity Loop	TRQC.CUTOFF	Hz	Torque command filter cutoff frequency	Define the cutoff frequency for the torque command filter.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Velocity Loop	TRQC.METHOD		Torque command filter method	Define torque command filter type: TRQC.METHOD=BUTTERWORTH Implements Butterworth filter method. TRQC.METHOD=BESSEL Implements Bessel filter method. TRQC.METHOD=CHEBYSHEV_1DB Implements Chebyshev filter method with 1 dB of passband ripple. TRQC.METHOD=CHEBYSHEV_3DB Implements Chebyshev filter method with 3 dB of passband ripple. TRQC.METHOD=LINEAR_PHASE Implements a special form of the Butterworth filter which is linear in phase as much as possible.	Not accessible	Not accessible	Not accessible	Not accessible

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Velocity Loop	TRQC.TYPE		Torque command filter type	Define torque command filter type: TRQC.TYPE=BYPASS Filter is not active. TRQC.TYPE=LPF Implements low-pass filter type TRQC.TYPE=NOTCH Implements notch filter type	Not accessible	Not accessible	Not accessible	Not accessible
RW	Velocity Loop	VEL.CUTOFF	Hz	Velocity feedback filter cutoff frequency	Define the cutoff frequency for the velocity feedback filter.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Velocity Loop	VEL.METHOD		Velocity feedback filter method	Define velocity feedback filter type: VEL.METHOD=BUTTERWORTH Implements Butterworth filter method. VEL.METHOD=BESSEL Implements Bessel filter method. VEL.METHOD=CHEBYSHEV_1DB Implements Chebyshev filter method with 1 dB of passband ripple. VEL.METHOD=CHEBYSHEV_3DB Implements Chebyshev filter method with 3 dB of passband ripple. VEL.METHOD=LINEAR_PHASE Implements a special form of the Butterworth filter which is linear in phase as much as possible.	Not accessible	Not accessible	Not accessible	Not accessible
RW	Velocity Loop	VEL.TYPE		Velocity feedback filter type	Define velocity feedback filter type: VEL.TYPE=BYPASS Filter is not active. VEL.TYPE=LPF Implements low-pass filter type VEL.TYPE=NOTCH Implements notch filter type	Not accessible	Not accessible	Not accessible	Not accessible

ACCESS	CATEGORY	NAME	UNITS	DESCRIPTION	HOW TO USE	DATATYPE (ENGLISH)	DATATYPE (CANOPEN INDEX)	CANOPEN OBJECT ID	MODBUS REGISTER ADDRESS
RW	Velocity Loop	VLAT		Velocity loop advanced tuning switch	There are two modes for tuning the velocity loop: VLAT=ON With VLAT=ON, the parameters KV and KI are automatically calculated from the system inertia INER, motor torque constant KT, target bandwidth WN, and damping ratio ZETA through a proprietary means. In general, tuning with VLAT=ON is recommended as there is a complex interaction among all the velocity loop tuning parameters. With VLAT=ON, increasing WN increases the static and dynamic response of the velocity loop. Too high a value of WN will result in a noisy and unstable system. With VLAT=ON, increasing ZETA reduces over- and under-shoot when the drive is commanded with step-changes in velocity. A ZETA value equal to 1.0 will result in a critically-damped system with little or no overshoot depending on load and damping. VLAT=OFF With VLAT=OFF, the parameters KV and KI are not automatically calculated. The user is free to enter values for these parameters to tune the system. With VLAT=OFF and KVF=1.0, the velocity loop is a conventional PID loop with the following parameters: KV controls the proportional term, KI controls the integral term, and KD controls the derivative term. As a conventional PID loop, the velocity control loop can be tuned by standard methods such as Ziegler-Nichols.	Unsigned8	0x0005	0x2007	14
RW	Velocity Loop	WN	Hz	Velocity loop target bandwidth	Controls the desired target bandwidth of the velocity loop with VLAT=ON. When VLAT=ON, WN automatically calculates values for KV and KI based upon the inertia parameter INER and motor torque constant KT. It is therefore critical to properly calculate and enter the parameters INER and KT before using the velocity loop with VLAT=ON. A value of 100 Hz is usually acceptable in moderate-to-low inertia systems. In high-inertia systems, the value may need to be decreased for stability. In low-inertia systems the value may be increased for better dynamic and static response.	Real32	0x0008	0x2008	16
RW	Velocity Loop	ZETA		Velocity loop damping ratio	Used to calculate values for KV and KI in conjunction with the parameter WN. In systems that ring (oscillate) with step changes in velocity command, increasing ZETA will dampen the system response at the expense of settling time. In systems where some overshoot can be tolerated, or where greater dynamic response is desired, ZETA may be reduced.	Real32	0x0008	0x2009	18