Name	Type	Address	Unit	Default value	Description	Setting
TML Registers						
AAR	UINT	0x030C			AAR is a 16-bit status register, containing information that defines the individual and group addresses of the motion axis.	This register is initialized during the default configuration phase of the motion system
ACR	UINT	0x0912			ACR (Auxiliary Control Register) is a 16-bit status register. It defines extra settings like: the configuration for automatic start and the external reference, operation options for the S-curve and the electronic camming modes.	ACR register ia automaticali configurated durind the Setup procedure
ASR	UINT	0x0201			ASR is a 16-bit status register. It allows you to reset controllers history and enable/disable the software limit switches	
CCR	UINT	0x030A			CCR (Communication Control Register) is a 16-bit status register, containing settings for the SPI link with the EEPROM memory.	This register is initialized during the default configuration phase of the motion system
CER	UINT	0x0301			CER is a 16-bit status register, containing status information about communication errors on CAN, SPI and SCI communication channels.	The CER is set by the drive
CSR	UINT	0x030B			CSR is a 16-bit status register, containing status information about the communication channels of the system, SPI and SCI.	This register is initialized during the default configuration phase of the motion system
DER	UINT	0x35D			DER - Detail Error Register	
ICR	UINT	0x0304			ICR is a 16-bit command register, enabling/disabling the TML interrupts.	All the unmasked bits of this register will allow the generation of a TML interrupt at the occurrence of the associated specific situation.
ISR	UINT	0x0306			ISR is a 16-bit status register, containing the interrupt flags for TML interrupts. Only unmasked TML interrupts (see Interrupt Control Register - ICR) will generate a TML interrupt request.	
MCR	UINT	0x0309			MCR is a 16-bit status register containing information about the motion modes, reference mode, active control loops, positioning type - absolute or relative, etc.	This register is initialized during the default configuration phase of the motion system
MCR_1	UINT	0x02B0			MCR_1 a is a copy of the MCR register	See the MCR setting
MER_MASK	UINT	0x0965			MER_MASK is a mask for MER register. A bit set to 1, enables to send MER when the same bit from MER changes	MREF is automatically set and updated by the master

Name	Type	Address	Unit	Default value	Description	Setting
MSR	UINT	0x0308			MSR (Motion Status Register) is a 16-bit status register, containing information about motion system status and some specific events like: control error condition, position wrap-around, limit switches and captures triggered by programmed transitions, etc.	MSR values are set by the drive during the motion is active
OSR	UINT	0x0302			OSR (Operating Settings Register) is a 16-bit configuration register, which defines some specific operating settings regarding motor control and data acquisition	This register is initialized during the default configuration phase of the motion system
PCR	UINT	0x0303			PCR (Motion Status Register) is a 16-bit command and status register, containing both masks and status information for TML protections.	This register is initialized during the default configuration phase of the motion system
SCR	UINT	0x0300			SCR (System Configuration Register) is a 16-bit configuration register, defines the basic application configuration regarding the motor type and the feedback sensors used	This register is initialized during the default configuration phase of the motion system
SR32	ULONG	0x090E			SR32 is the 32 bits Status Register. For nore details see the SRH and SRL register descriptions	SR32 is set during run time according to the motion events.
SRH	UINT	0x090F			SRH is the high part of a the status register grouping together all the key status information concerning the drive/motor	This register is initialized during the default configuration phase of the motion system
SRH_MASK	UINT	0x0963			The SRH_MASK is the mask of the SRH register and contain a bit which enables a message transmission when the same bit from the corresponding register changes	This register is initialized during the default configuration phase of the motion system
SRL	UINT	0x090E			SRL is the low part of a status register grouping together all the key status information concerning the drive/motor	This register is initialized during the default configuration phase of the motion system
SRL_MASK	UINT	0x0962			The SRL_MASK is the mask of the SRL register and contain a bit which enables a message transmission when the same bit from the corresponding register changes	This register is initialized during the default configuration phase of the motion system
UPGRADE	INT	0x0857			UPGRADE is a 16-bit status register, defining new options and extended features that are activated when their corresponding bits are set.	Defines new options and extended features which are activated when their corresponding bits are set to 1
TML Parameters						
AAR_table	UINT	0x0913			AAR_TABLE contains the Axis/Group ID as defined by the setup table. It is also modified after each Axis/Group ID change. Bit description is the same as AAR	The AAR_Table is initialized in the configuration (setup) phase
ACC_LIMIT	UINT	0x02C7			ACC_LIM parameter represent the acceleration limit for the open loop stepers, with encoder mounted on load	ACC_LIM can be set from Drive Setup dialog, during the commisioning

Name	Type	Address	Unit	Default value	Description	Setting
AD0OFF	UINT	0x0244		32736 (0x7FE0)	AD0OFF parameters represent the offset used to compute measure variables AD0, from the conversion data given by the analog-to-digital converters.	As the AD0 measure result falls within the range [0, 0xFFC0] (10bits, left aligned), the AD0OFF parameter enables the “polarization” of this range as unipolar or bipolar values. AD0OFF can be used as follows: Case 1: to define a positive unipolar range for AD0 measure in the range [0, Xmax]: set AD0OFF = 0; Case 2: to define a positive bipolar range for AD0 measure in the range [-Xmax, Xmax]: set AD0OFF =0x7FE0 Case 3: to define a negative unipolar range for AD0 measure in the range [-Xmax, 0]: set AD0OFF = 0xFFC0;
AD1OFF	UINT	0x245		32736 (0x7FE0)	AD1OFF parameters represent the offset used to compute measure variables AD1, from the conversion data given by the analog-to-digital converters.	As the AD1 measure result falls within the range [0, 0xFFC0] (10bits, left aligned), the AD1OFF parameter enables the “polarization” of this range as unipolar or bipolar values. AD1OFF can be used as follows: Case 1: to define a positive unipolar range for AD1 measure in the range [0, Xmax]: set AD1OFF = 0; Case 2: to define a positive bipolar range for AD1 measure in the range [-Xmax, Xmax]: set AD1OFF =0x7FE0 Case 3: to define a negative unipolar range for AD1 measure in the range [-Xmax, 0]: set AD1OFF = 0xFFC0;
AD2OFF	UINT	0x246		32736 (0x7FE0)	AD2OFF parameters represent the offset used to compute measure variables AD2, from the conversion data given by the analog-to-digital converters.	As the AD2 measure result falls within the range [0, 0xFFC0] (10bits, left aligned), the AD2OFF parameter enables the “polarization” of this range as unipolar or bipolar values. AD2OFF can be used as follows: Case 1: to define a positive unipolar range for AD2 measure in the range [0, Xmax]: set AD2OFF = 0; Case 2: to define a positive bipolar range for AD2 measure in the range [-Xmax, Xmax]: set AD2OFF =0x7FE0 Case 3: to define a negative unipolar range for AD2 measure in the range [-Xmax, 0]: set AD2OFF = 0xFFC0;
AD3OFF	UINT	0x0247		32736 (0x7FE0)	AD3OFF parameter represents the offset for the AD3 variable.	

Name	Type	Address	Unit	Default value	Description	Setting
AD4OFF	UINT	0x0248		32736 (0x7FE0)	AD4OFF parameters represent the offset used to compute measure variables AD4, from the conversion data given by the analog-to-digital converters.	As the AD4 measure result falls within the range [0, 0xFFC0] (10bits, left aligned), the AD4OFF parameter enables the “polarization” of this range as unipolar or bipolar values. AD4OFF can be used as follows: Case 1: to define a positive unipolar range for AD4 measure in the range [0, Xmax]: set AD4OFF = 0; Case 2: to define a positive bipolar range for AD4 measure in the range [-Xmax, Xmax]: set AD4OFF =0x7FE0 Case 3: to define a negative unipolar range for AD4 measure in the range [-Xmax, 0]: set AD4OFF = 0xFFC0;
AD5OFF	INT	0x0249		32736 (0x7FE0)	AD5OFF parameters represent the offset used to compute measure variables AD5, from the conversion data given by the analog-to-digital converters.	As the AD5 measure result falls within the range [0, 0xFFC0] (10bits, left aligned), the AD5OFF parameter enables the “polarization” of this range as unipolar or bipolar values. AD5OFF can be used as follows: Case 1: to define a positive unipolar range for AD5 measure in the range [0, Xmax]: set AD5OFF = 0; Case 2: to define a positive bipolar range for AD5 measure in the range [-Xmax, Xmax]: set AD5OFF =0x7FE0 Case 3: to define a negative unipolar range for AD5 measure in the range [-Xmax, 0]: set AD5OFF = 0xFFC0;
AD6OFF	UINT	0x024A		32736 (0x7FE0)	AD6OFF parameter represents the offset for the AD6 variable.	
AD7OFF	UINT	0x024B		32736 (0x7FE0)	AD7OFF parameters represent the offset used to compute measure variables AD7, from the conversion data given by the analog-to-digital converters.	As the AD7 measure result falls within the range [0, 0xFFC0] (10bits, left aligned), the AD7OFF parameter enables the “polarization” of this range as unipolar or bipolar values. AD7OFF can be used as follows: Case 1: to define a positive unipolar range for AD7 measure in the range [0, Xmax]: set AD2OFF = 0; Case 2: to define a positive bipolar range for AD7 measure in the range [-Xmax, Xmax]: set AD2OFF =0x7FE0 Case 3: to define a negative unipolar range for AD7 measure in the range [-Xmax, 0]: set AD2OFF = 0xFFC0;

Name	Type	Address	Unit	Default value	Description	Setting
ALPHA	LONG	0x0295		0 (0x0000)	The ALPHA(L) parameter sets the threshold value for the linear Hall signal missing detection. If the linear Hall signals results in a value below then ALPHA(L) the Position Wraparound interrupt is triggered and the PWM outputs are disabled.	The parameter ALPHA(L) is set with the threshold value read from the EEPROM memory. The threshold value is automatically determined and saved in the EEPROM by the linear Hall gains and offsets detection test.
BRAKELIM	UINT	0x028A		-1 (0xFFFF)	BRAKELIM parameter defines the DC voltage value for which the brake output becomes active	BRAKELIM must be set if the breaking mode will be activated in the OSR register. Its value is compared with the DC voltage value and the braking is activated once the DC voltage is bigger than the BRAKELIM value. If the DC voltage becomes smaller than BRAKELIM, the braking is deactivated.
CACC	FIXED	0x02A2	[rad/s^2]	0.5 (0x00008000)	CACC parameter represents the absolute value of the acceleration or deceleration for a positioning or speed profile movement.	CACC must be set before an immediate or event-based update command is issued. At the update command execution, the value of CACC is transferred to the reference generator module and is used to compute the position or speed reference.
CADIN	INT	0x025C		32767(0x7FFF)	CADIN parameter is used to scale the analog reference measured from the conversion data given by the analog-to-digital converter.	CADIN and SFTADIN depend on the peak value of the desired range and on the interpretation of the ADC input. The CADIN and SFTADIN parameters enable the “polarization” of ADC input range as unipolar or bipolar values. These parameters can be used as follows: Case 1: to define a positive unipolar range for analog reference measure in the range [0,Xmax]; Case 2: to define a positive bipolar range for analog reference measure in the range [Xmax, Xmax]; Case 3: to define a negative unipolar range for analog reference measure in the range [- Xmax, 0]; CADIN and SFTADIN parameters are computed as follows: Case 1: CADIN_fixed = Xmax * 32767 / 32736 (computed as a fixed) Case 2: CADIN_fixed = Xmax * 32767 / 16368 (computed as a fixed) Case 3: CADIN_fixed = Xmax * 32767 / 32736 (computed as a fixed) If CADIN_fixed < 32767 SFTADIN = 0 and CADIN = CADIN_fixed. If CADIN_fixed > 32767 SFTADIN = the least integer power of 2 for which CADIN_fixed <2^SFTADIN * 32767 CADIN = CADIN_fixed / 2^SFTADIN (computed as an integer)

Name	Type	Address	Unit	Default value	Description	Setting
CAMOFF	LONG	0x03AD		0 (0x0000)	CAMOFF parameter it's used in electronic camming modes allowing shifting the cam profile versus the master position. It represents the offset between the master position and slave axis positions in electronic camming.	Must be set at the same time with the cam table. The following relation exists between: the master position (MREF), the cam offset (CAMOFF), the cam table X input (MPOS0) and the master resolution (MASTERRES): MPOS0 = (MREF – CAMOFF) % MASTERRES
CAMSTART	INT	0x03AC		0 (0x0000)	CAMSTART parameter points to SRAM program memory start address for a cam table. When several cam tables are used, switching between them resumes to set CAMSTART to the right address i.e. the beginning of next the cam table to use. CAMSTART is automatically set by the INITCAM command, which copies the cam table from the EEPROM to the SRAM memory	CAMSTART parameter is initialized after the execution of INITCAM instruction with the RAM address where the cam table was copied.
CLPER	UINT	0x250		2 (0x0002)	The CLPER parameter is used for the setting of the fast control loop (current) sampling period.	CLPER depends on the PWM frequency and on the frequency of the fast control loop (current/field control loop). CLPER is expressed as a multiple of PWM periods, and is computed using the next relation: CLPER = PWM_freq · Ts_C
CPOS	LONG	0x029E	[rot]	0 (0x00000000)	CPOS parameter represents the value of the command position for a positioning profile movement. CPOS is considered relative, after a CPR command or absolute after a CPA command. After reset, CPOS is relative. By default, in a relative positioning, CPOS value is added to the actual position APOS of the motor at the moment when update command is executed. If TUM1 command is issued after the positioning profile mode is set, CPOS value is added to the target position TPOS at the moment when update command is executed.	CPOS must be set before an immediate or event-based update command is issued. At the update command execution, the value of CPOS is transferred to the reference generator module and is used to compute the position reference.
CSPD	FIXED	0x02A0	[rpm]	0 (0x00000000)	CSPD parameter represents the value of the slew speed for a positioning profile movement or the value of the jog speed for a speed profile movement. In a speed profile the CSPD sign indicates the direction of the movement. In a positioning profile, the CSPD sign is disregarded.	CSPD must be set before an immediate or event-based update command is issued. At the update command execution, the value of CSPD is transferred to the reference generator module and is used to compute the position or speed reference (depending on the motion mode settings).

Name	Type	Address	Unit	Default value	Description	Setting
CVDC	INT	0x025B			CVDC parameter is used to implement the compensation of DC link voltage (VDC) variation in the PWM command. Its value represents the ratio between the nominal and the measured VDC values.	CVDC is computed each control loop sampling if the UDC Compensation feature is activated. In the computation of the duty cycles of the PWM signals, the values are multiplied with the CVDC variable to compensate for the variations of the DC bus voltage: if the DC bus voltage is higher than the nominal value described by the VDCN parameter, the duty cycles of the phase oltages are decreased. The CVDC parameter is computed with the following formula: CVDC=PWM_per*[VDCN/(V_DC*Kuf_m)]
DBT	UINT	0x253		20 (0x014)	DBT parameter is used to set the dead band time for the PWM output commands	DBT depends on the dead band time of the power converter devices.
EFLEVEL	INT	0x02C7		32767 (0x7FFF)	EFLEVEL parameter activates or deactivates the synchronization mechanism between the master and slave in electronic gearing or camming. The master send it's position via communication.	The user must set EFLEVEL before starting the operation of the axis in the slavegearing mode.
ENC2TH	UINT	0x278			ENC2THL parameter is used to scale the measured position information, in order to obtain the electrical angle (theta).	ENC2THL depends on the number of encoder lines and pole pairs.
ENC2THL	LONG	0x024C			ENC2THL parameter is used to scale the measured position information, in order to obtain the electrical angle (theta).	ENC2THL depends on the number of encoder lines and pole pairs. ENC2THL is set as follows: ENC2THL = 2^32 * pp / (4*No_encoder_lines)
EREF	LONG	0x02A8			EREF parameter stores the value of the reference, for all external reference modes.	EREF must be set with the reference value, for all external "on-line" reference modes. Dependingon the reference type, its value will be considered as a long integer (for position references), or as a fixed-point 32-bit fractional value (for speed, torque or voltage references).
ERRMAX	UINT	0x02C5	[rot]	32767 (0x7FFF)	ERRMAX parameter represents the maximum accepted control error of the most outer loop of the controlled system. If the error between the reference and the feedback exceeds this value for a time that is greater than the value specified by the TERRMAX parameter, the corresponding control error flag (CTRERIF) is set in the Interrupt Status Register – ISR.	ERRMAX depends on the maximum accepted error for the variable that is controlled in the outer loop and on its scaling factor. It is expressed as follows: ERRMAX = (max. accepted error of the controlled variable) * (controlled variable scaling factor)
FFL	INT	0x223		0	FFL parameter is the current offset value for gravitational load compensation	FFL parameter is defined in the setup phase and it can be modified from a TML program or online by an external device

Name	Type	Address	Unit	Default value	Description	Setting
FILTER1	INT	0x029D		32767 (0x7FFF)	FILTER1 parameter represents the 1st order filter coefficient used to filter the reference signal, when measured from an A/D converter input.	FILTER1 is used to implement a first order filter on the measured A/D reference value. For the maximum value (0x7FFF), the measured value is not filtered. Complete attenuation is obtained for value 0. The filtering relation for an input x, output of the filter y, is: yn = yn-1 + (xn - yn-1) * FILTER1 Depending on the measurement environment (noise level), adjust the value of FILTER1, starting from its maximal value, until accurate measurements are obtained
FILTERQ	INT	0x0982			FILTERQ parameter represents the 1st order filter coefficient used to filter the current reference	FILTERQ parameter is defined in the setup phase and it can be modified from a TML program or online by an external device
GEAR	FIXED	0x02AC		1 (0x00000001)	GEAR, GEARMASTER and GEARSLAVE parameters are used to set the gearing factor used by an axis when operating in slave gearing mode. GEAR is multiplied with the master position increment (MREF – MPOS0), to give the target position increment of the slave. GEARSLAVE and GEARMASTER represent the numerator and denominator of the Slave / Master ratio. GEARSLAVE is a signed integer, while GEARMASTER is an unsigned integer. GEARSLAVE sign indicates the direction of movement: positive – same as the master, negative – reversed to the master. GEAR is a fixed value containing the result of the gear ratio i.e. the result of the division GEARSLAVE / ARMASTER. In order to eliminate any cumulative errors the electronic gearing slave mode includes an automatic compensation of the round off errors when the gear ratio has an irrational value like: Slave = 1, Master = 3, giving a ratio of 1/3 = 0.33333 which can’t be represented exactly.	The user must set GEAR before starting the operation of the axis in the slave-gearing mode. This parameter can be set directly using an assignment instruction, or is computed based on GEARMASTER and GEARSLAVE values.

Name	Type	Address	Unit	Default value	Description	Setting
GEARMASTER	UINT	0x255		-31072 (0x86A0)	GEAR, GEARMASTER and GEARSLAVE parameters are used to set the gearing factor used by an axis when operating in slave gearing mode. GEAR is multiplied with the master position increment (MREF – MPOS0), to give the target position increment of the slave. GEARSLAVE and GEARMASTER represent the numerator and denominator of the Slave / Master ratio. GEARSLAVE is a signed integer, while GEARMASTER is an unsigned integer. GEARSLAVE sign indicates the direction of movement: positive – same as the master, negative – reversed to the master. GEAR is a fixed value containing the result of the gear ratio i.e. the result of the division GEARSLAVE / ARMASTER. In order to eliminate any cumulative errors the electronic gearing slave mode includes an automatic compensation of the round off errors when the gear ratio has an irrational value like: Slave = 1, Master = 3, giving a ratio of 1/3 = 0.33333 which can't be represented exactly.	The user must set GEAR before starting the operation of the axis in the slave-gearing mode. This parameter can be set directly using an assignment instruction, or is computed based on GEARMASTER and GEARSLAVE values.
HALL30	INT	0x0877			HALL30 parameter specifie the digital Hall signals offset	HALL30 is automatically determined by the "Detect Hall Configuration" test in Easy SetUp / EasyMotion Studio
HALLAGAIN	INT	0x0287			HALLAGAIN parameter represents the gain factor used by MotionChip to interpret the values read from the linear Hall channel A.	HALLAGAIN is set by the "Detect Hall Gains and Offset" test in Easy SetUp / EasyMotion Studio
HALLAOFF	UINT	0x024A			HALLAOFF parameter represents the offset used by MotionChip to interpret the values read from the linear Hall channel A.	HALLAOFF is set by the "Detect Hall Gains and Offset" test in Easy SetUp / EasyMotion Studio
HALLBGAIN	INT	0x028F			HALLBGAIN parameter represents the gain factor used by MotionChip to interpret the values read from the linear Hall channel B.	HALLBGAIN is set by the "Detect Hall Gains and Offset" test in Easy SetUp / EasyMotion Studio
HALLBOFF	UINT	0x0247			HALLBOFF parameter represents the offset used by MotionChip to interpret the values read from the linear Hall channel B.	HALLBOFF is set by the "Detect Hall Gains and Offset" test in Easy SetUp / EasyMotion Studio
HALLCASE	INT	0x0259		32767 (0x7FFF)	HALLCASE parameter is used to set the Hall sensor configuration mode.	HALLCASE depends on the Hall sensor configuration, as related to the motor phases.
HALLCGAIN	INT	0x0279			HALLCGAIN parameter represents the gain factor used by MotionChip to interpret the values read from the linear Hall channel C.	HALLCGAIN is set by the "Detect Hall Gains and Offset" test in Easy SetUp / EasyMotion Studio
HALLCOFF	UINT	0x0258			HALLCOFF parameter represents the offset used by MotionChip to interpret the values read from the linear Hall channel C.	HALLCOFF is set by the "Detect Hall Gains and Offset" test in Easy SetUp / EasyMotion Studio

Name	Type	Address	Unit	Default value	Description	Setting
HALLFILT	INT	0x0809			HALLFILT parameter represents the 1st order filter coefficient used to filter the linear Hall signals	HALLFILT is used to implement a first order filter on the linear Hall signal. For the maximum value (0x7FFF), the measured value is not filtered. Complete attenuation is obtained for value 0. The filtering relation for an input x, output of the filter y, is: yn = yn-1 + (xn - yn-1) * HALLFILT.
I2TWARLIM_M	ULONG	0x097E			I2TWARLIM_M paremeter represents the warning level for the i2t motor protection.	I2TWARLIM_M paremeter is automatically computed during the setup phase based on the i2t motor protection limit.
II2TPROT_D	INT	0x0986	[A]		II2TPROT_D parameter is the reference for computing the I2t protection.	II2TPROT_D is computed based on motor nominal current. II2TPROT= In * Kif
II2TPROT_M	INT	0x0814	[A]	0 (0x0000)	II2TPROT_M parameter is the reference for computing the I2t protection.	II2TPROT_M is computed based on motor nominal current. II2TPROT= In * Kif
II2TWARLIM_D	ULONG	0x097C			II2TWARLIM_D paremeter represents the warning level for the i2t drive protection.	II2TWARLIM_D paremeter is automatically computed during the setup phase based on the i2t drive protection limit.
IMAXP	INT	0x266		0x0261	KIP, SFTKIP, IMAXP parameters are used to implement the integrative component of the PID position controller.	KIP, SFTKIP, IMAXP depend on the integrative component of the PID position controller coefficients, obtained from the tuning, on the controller maximum speed output (spd_max) and on the speed scaling factor. KDP, SFTKDP, KDFP are computed as follows: SFTKIP = the least integer power of 2 for which KIP < 2^SFTKIP KIP = Ki_P_scl * 32767 / 2^SFTKIP IMAXP = user defined. Defines the saturation value of the integrative part.
IMAXPROT	INT	0x297	[A]	32767 (0x7FFF)	IMAXPROT parameter represents the current values which, when exceeded for a time greater than the interval specified by TIMAXPROT, will activate the maximum current protection.	IMAXPROT depends on the motor and power converter maximum current. It is expressed as follows: IMAXPROT = IMAX_PROT* Kif
IMAXS	INT	0x026C		0 (0x0000)	KIS, SFTKIS, IMAXS parameters are used to implement the integrative component of the PI speed controller.	KIS, SFTKIS, IMAXS depend on the integrative component of the PI speed controller coefficients, obtained from the tuning, on the controller maximum current output (I_max) and on the current scaling factor. KIS, SFTKIS, IMAXS are computed as follows: SFTKIS = the least integer power of 2 for which KIS < 2^SFTKIS KIS = Ki_S_scl * 32767 / 2^SFTKIS IMAXS = user defined. Defines the saturation value of the integrative part.

Name	Type	Address	Unit	Default value	Description	Setting
INITANGLE	INT	0x89B			INITANGLE parameter is used by the motionless start algorithm. It defines the initial electrical angle that is searched during the motionless start.	INITANGLE parameter is set by the setup
INSTATUS	UINT	0x0908			INSTATUS variable provides the status of the special digital inputs of the drive	INSTATUS variable value is actualised by the drive
INTTABLE	INT	0x307		0 (0x0000)	INTTABLE parameter represents the starting address of the TML interrupt vector table	INTTABLE value will be set by the user to point to the starting address of the TML interrupt vector table, which contains pointers to the TML interrupt routines, associated with each of the TML interrupts. INTTABLE = address of interrupt vector table;
KDFP	INT	0x0264		0x0264	KDP, SFTKDP, KDFP parameters are used to implement the derivative component of the PID position controller.	KDP, SFTKDP, KDFP depend on the derivative component of the PID position controller coefficients, obtained from the tuning, on the controller maximum speed output and on the speed scaling factor. KDP, SFTKDP, KDFP are computed as follows: SFTKDP = the least integer power of 2 for which KD_P_scl < 2^SFTKDP KDP = Kd_P_scl * 32767 / 2^SFTKDP KDFP = filter_D*32767
KDP	INT	0x0262		0 (0x0000)	KDP, SFTKDP, KDFP parameters are used to implement the derivative component of the PID position controller.	KDP, SFTKDP, KDFP depend on the derivative component of the PID position controller coefficients, obtained from the tuning, on the controller maximum speed output and on the speed scaling factor. KDP, SFTKDP, KDFP are computed as follows: SFTKDP = the least integer power of 2 for which KD_P_scl < 2^SFTKDP KDP = Kd_P_scl * 32767 / 2^SFTKDP KDFP = filter_D*32767
KFFA	INT	0x026E		0 (0x0000)	KFFA and SFTAFFW parameters are used to implement the acceleration feedforward term in the acceleration & load feedforward block. They are used in the multiplication with the target acceleration issued by the reference generator module. The obtained term is added at the output of the speed controller, as an acceleration feedforward term.	KFFA represents the multiplication coefficient. The SFTAFFW parameter represents the shift used to extract the acceleration feedforward component from the target acceleration. These parameters depend on the system model and parameters. Theoretically, they can be computed from these parameters (mainly system inertia). Practically, try-and-error methods can be used to “tune” optimal values for these parameters. Use as start values the default ones (0).

Name	Type	Address	Unit	Default value	Description	Setting
KFFL	INT	0x026F		0 (0x0000)	KFFL parameter is used to implement the load torque feedforward term in the acceleration & load feedforward block. It is used in the multiplication with the measured load torque value. The obtained term is added at the output of the speed controller, as a load torque feedforward term.	KFFL represents the multiplication coefficient of the load torque value. This parameter depends on the system model and parameters. Theoretically, it can be computed from these parameters. Practically, try-and-error methods can be used to “tune” optimal values for this parameter. Use as start value the default one (0).
KFFS	INT	0x026D		0 (0x0000)	KFFS and SFTSFFW parameters are used to implement the speed feedforward term added at the input of the PI speed controller. They are used in the multiplication with the target speed issued by the reference generator module. The obtained term is added at the output of the speed controller, as a speed feedforward term.	KFFS represents the multiplication coefficient. The SFTSFFW parameter represents the shift used to extract the speed feedforward component from the target speed. These parameters depend on the system model and parameters. Theoretically, they can be computed from these parameters (mainly system viscous friction). Practically, try-and-error methods can be used to “tune” optimal values for these parameters. Use as start values the default ones (0).
KII	INT	0x273		0 (0x0000)	KII and SFTKII parameters are used to implement the integral component of the PI current controllers.	KII and SFTKII depend on the integral component of the PI current controller coefficients, obtained from the tuning and on the controller maximum voltage outputs (u_d_max, u_q_max). KII and SFTKII are computed as follows: SFTKII = the least integer power of 2 for which KII < 2^SFTKII KII = Ki_C_scl * 32767 / 2^SFTKII
KIP	INT	0x0260		0x0260	KIP, SFTKIP, IMAXP parameters are used to implement the integrative component of the PID position controller.	KIP, SFTKIP, IMAXP depend on the integrative component of the PID position controller coefficients, obtained from the tuning, on the controller maximum speed output (spd_max) and on the speed scaling factor. KDP, SFTKDP, KDFP are computed as follows: SFTKIP = the least integer power of 2 for which KIP < 2^SFTKIP KIP = Ki_P_scl * 32767 / 2^SFTKIP IMAXP = user defined. Defines the saturation value of the integrative part.

Name	Type	Address	Unit	Default value	Description	Setting
KIS	INT	0x0269		0 (0x0000)	KIS, SFTKIS, IMAXS parameters are used to implement the integrative component of the PI speed controller.	KIS, SFTKIS, IMAXS depend on the integrative component of the PI speed controller coefficients, obtained from the tuning, on the controller maximum current output (I_max) and on the current scaling factor. KIS, SFTKIS, IMAXS are computed as follows: SFTKIS = the least integer power of 2 for which KIS < 2^SFTKIS KIS = Ki_S_scl * 32767 / 2^SFTKIS IMAXS = user defined. Defines the saturation value of the integrative part.
KISPDEST	INT	0x095B			KISPDEST parameter is used to implement the integral component of the speed estimator	KISPDEST is set by the Setup
KITHT	INT	0x0888			KITHT parameter is used to implement the integral component of the electrical angle estimator which is used during the motionless start procedure.	KITHT has to be set by the user from the "Electric Angle Controller Tuning Test" from Easy SetUp/ EasyMotion Studio
KPI	INT	0x0271		0 (0x0000)	KPI and SFTKPI parameters are used to implement the proportional component of the PI current controllers.	KPI and SFTKPI depend on the proportional component of the PI current controller coefficients, obtained from the tuning and on the controller maximum voltage outputs (u_d_max, u_q_max). KPI and SFTKPI are computed as follows: SFTKPI = the least integer power of 2 for which KPI < 2^SFTKPI KPI = Kp_C_scl * 32767 / 2^SFTKPI
KPP	INT	0x025E		0 (0x0000)	KPP and SFTKPP parameters are used to implement the proportional component of the PID position controllers.	KPP and SFTKPP depend on the proportional component of the PID position controller coefficients, obtained from the tuning and on the controller maximum speed output (spd_max). KPP and SFTKPP are computed as follows: SFTKPP = the least integer power of 2 for which KPP < 2^SFTKPP KPP = Kp_P_scl * 32767 / 2^SFTKPP
KPS	INT	0x267		0 (0x0000)	KPS and SFTKPS parameters are used to implement the proportional component of the PI speed controllers.	KPS and SFTKPS depend on the proportional component of the PI speed controller coefficients, obtained from the tuning and on the controller maximum current output (I_max). KPS and SFTKPS are computed as follows: SFTKPS = the least integer power of 2 for which KPS < 2^SFTKPS KPS = Kp_S_scl * 32767 / 2^SFTKPS
KPSPDEST	INT	0x095C			KPSPDEST parameter is used to implement the proportional gain of the speed estimator	KPSPDEST is set by the Setup
KPTHT	INT	0x0887			KPTHT parameter is used to implement the proportional gain of the electrical angle estimator which is used during the motionless start procedure.	KPTHT has to be set by the user from the "Electric Angle Controller Tuning Test" from Easy SetUp/ EasyMotion Studio

Name	Type	Address	Unit	Default value	Description	Setting
MASTERRES	LONG	0x081A		0 (0x00000000)	MASTERRES parameter indicates the master position sensor mechanical resolution for one motor rotation. The slaves need the master resolution to compute correctly the master position and speed (i.e. position increment).	MASTERRES depends on the position sensor type. For sensors connected to the QEP interface (quadrature encoders), MASTERRES is set as follows: MASTERRES = 4 * o_encoder_lines For other position sensors (not connected to the QEP interface), MASTERRES is set as follows: MASTRERRES = No_sensor_bits/ rotation
MECRESL	LONG	0x024E		7995494 (0x007A0066)	MECRESL parameter is used to indicate the position sensor mechanical resolution for one motor rotation.	MECRESL depends on the position sensor type. For sensors connected to the QEP interface (quadrature encoders), MECRESL is set as follows: MECRESL = 4 * No_encoder_lines For other position sensors (not connected to the QEP interface), MECRESL is set as follows: MECRESL = (No_sensor_bits/ rotation)
MPOS0	LONG	0x02E5		0 (0x00000000)	MPOS0 parameter is used by an axis when operating in slave gearing mode to compute its new target position with the formula: New TPOS = old TPOS + (MREF – MPOS0) x GEAR where, MREF is the master position, sent by the master and MPOS0 the previous master position.	Before activating the slave mode MPOS0 acts like a parameter that has to be initialized by the master with its target or actual position. After the slave mode is set, MPOS0 becomes a variable, which is updated by the slave at each position / speed sampling period with the previous value of MREF.
NI2T	LONG	0x255		100000 (0x000186	NI2T parameter represents the time after which one I2t protection iteration is executed.	NI2T depends on thermal model parameter values of the motor (mainly on the thermal time constant of the motor). NI2T depends on slow loop frequency analysis. NI2T will be set using the following relation: NI2T = (analysis time value) · Ktf
POS0	LONG	0x02B8		0 (0x00000000)	POS0 parameter is used to mark the absolute position value, from which the relative position will be measured when a relative position event is set. By default, POS0 is updated each time a new update command is executed. As effect, POS0 is set to the value of the actual position APOS of the motor. Using TUM1 command it is also possible to set POS0 to the value of the target position TPOS. After a RAOU (reset automatic origin update) command, POS0 remains unchanged when a new update is executed. This operation mode can be used to monitor a relative position event while the motion modes or parameters are changing. With command SAOU (set automatic origin update) the default operation mode is restored. POS0 is used as parameter only if TML relative positioning events are programmed with RAOU mode set.	POS0 must be set before setting a TML relative event, and only under RAOU command. In the other cases, POS0 is a variable automatically updated.

Name	Type	Address	Unit	Default value	Description	Setting
POSOKLIM	UINT	0x036A	[rot]	0 (0x0000)	POSOKLIM parameter is used to suspend (to lock) the execution of the control loops if the position error is lower than this value for a time greater than the one specified with the TONPOSOK parameter. This special feature of the position control loop can be useful in such applications where the vibrations of the motor are undesired from the moment when the load has arrived to its target position.	POSOKLIM can be set based on the application requirements and the position sensor resolution. If the position sensor resolution is good enough and the application accepts a certain position error, POSOKLIM could be set to a value slightly lower than the accepted position error.
PWMPER	UINT	0x0252	0	500	PWMPER parameter is used to set the value of the PWM period.	PWMPER depends on the values of the PWM frequency and of the DSP clock. It is computed as follows: PWMPER = DSP_clock / (2*PWM_freq)
REF_INC	INT	0x0897			REF_INC is a parameter used by the motionless start algorithm. It defines the slope of the current referance.	REF_INC parameter is set by the user from the "Electric Angle Controller Tuning Test" from Easy SetUp/ EasyMotion Studio
REF0	FIXED	0x02A8		0 (0x00000000)	REF0 parameter is used to set the initial value of the reference, when a contouring torque or voltage mode is activated.	REF0 must be set before starting a contouring torque or voltage mode. It will represent the initial value used by the reference generator, from which the contouring values are computed.
REF0_fixed	FIXED	0x02A8			REF0_fixed parameter is used to set the initial fixed value of the reference, when a contouring torque or voltage mode is activated.	REF0_fixed must be set before starting a contouring torque or voltage mode. See the REF0 parameter description
REF0_long	LONG	0x02A8			REF0_long parameter is used to set the initial long value of the reference, when a contouring torque or voltage mode is activated.	REF0_long must be set before starting a contouring torque or voltage mode. See the REF0 parameter description
REFTST	INT	0x0281		0 (0x0000)	REFTST parameter is used to implement the starting modes in Brushless Motor applications (AC control mode), or to implement the test-operating mode; it represents the saturation value of the generated reference.	REFTST depends on the reference type (current, voltage), on the input value and on the scaling factor associated to each type of input.
RINCTST	INT	0x0280		0 (0x0000)	RINCTST parameter is used to implement the starting modes in Brushless Motor applications (AC control mode), or to implement the test-operating mode; it represents the increment value for the reference generated.	RINCTST parameter is used to implement the starting modes in Brushless Motor applications (AC control mode), or to implement the test-operating mode; it represents the increment value for the reference generated.
SATID	INT	0x0275		0 (0x0000)	SATID parameter is used at the implementation of PI current controllers. It represents the D axis current controller output limit.	SATID depends on the current controller coefficients resulting from the tuning of (Kp_crt, Ki_crt) and on the maximum controller voltage outputs (u_d_max, u_q_max). SATID is computed as follows: SATID = 65535 * 2e-6 * PWM_freq

Name	Type	Address	Unit	Default value	Description	Setting
SATIQ	INT	0x0276		0	SATIQ parameter is used at the implementation of PI current controllers. It represents the Q axis current controller output limit.	SATIQ depends on the current controller coefficients resulting from the tuning of (Kp_crt, Ki_crt) and on the maximum controller voltage outputs (u_d_max, u_q_max). SATIQ is computed as follows: SATIQ = 65535 * 2e-6 * PWM_freq
SATP	INT	0x0265		0 (0x0000)	SATP parameter is used at the implementation of PID position controller. It represents the position controller output limit.	SATP depends on the PID position controller coefficients resulting from the tuning of (Kp_pos, Ki_pos, Kd_pos); on the controller maximum speed output (spd_max) and on the speed-scaling factor. SATP = 32767 – (N_max) * Kvf
SATPWM	INT	0x0254		2620 (0x0A3C)	SATPWM parameter is used to limit the maximum value of the reference voltage command. The reference voltage saturation is required for accurate current measurement.	SATPWM depends on the PWM frequency and on the saturation range. It is computed as follows: SATPWM = 65535 * (saturation range) * (PWM_freq) * 2
SATS	INT	0x026B		0 (0x0000)	SATS parameter is used at the implementation of PI speed controller. It represents the speed controller output limit.	SATS depends on the PI speed controller coefficients resulting from the tuning (Kp_spd, Ki_spd); on the maximum output current of the PI speed controller (I_max); on the current scaling factors; on the speed feedforward scaling factors. SATS is computed as follows: SATS = 32767 – I_max * Kif
SERRMAX	INT	0x0879	[rpm]		SERRMAX parameter represents the maximum accepted speed error. If the error between the speed reference and the speed feedback exceeds this value for a time that is greater than the value specified by the TSERRMAX parameter, the corresponding control error flag (CTRERIF) is set in the Interrupt Status Register – ISR.	SERRMAX depends on the maximum accepted error for the variable that is controlled in the outer loop and on its scaling factor. It is expressed as follows: ERRMAX = (max. accepted error of the controlled variable) * (controlled variable scaling factor).
SFI2T_D	INT	0x098C		0 (0x0000)	SFI2T_D is used to scale the integral computed for I2t drive thermal protection.	SFI2T_D is computed based on time at over current t_I2t defined by the user
SFI2T_M	INT	0x0819		0 (0x0000)	SFI2T_M is used to scale the integral computed for I2t motor thermal protection.	SFI2T_M is computed based on time at over current t_I2t defined by the user

Name	Type	Address	Unit	Default value	Description	Setting
SFTADIN	INT	0x025D		0 (0x0000)	AFTADIN parameter is used to scale the analog reference measured from the conversion data given by the analog-to-digital converter.	CADIN and SFTADIN depend on the peak value of the desired range and on the interpretation of the ADC input. The CADIN and SFTADIN parameters enable the “polarization” of ADC input range as unipolar or bipolar values. These parameters can be used as follows: Case 1: to define a positive unipolar range for analog reference measure in the range [0,Xmax]; Case 2: to define a positive bipolar range for analog reference measure in the range [Xmax, Xmax]; Case 3: to define a negative unipolar range for analog reference measure in the range [-Xmax, 0]; CADIN and SFTADIN parameters are computed as follows: Case 1: CADIN_fixed = Xmax * 32767 / 32736 (computed as a fixed) Case 2: CADIN_fixed = Xmax * 32767 / 16368 (computed as a fixed) Case 3: CADIN_fixed = Xmax * 32767 / 32736 (computed as a fixed) If CADIN_fixed < 32767 SFTADIN = 0 and CADIN = CADIN_fixed. If CADIN_fixed > 32767 SFTADIN = the least integer power of 2 for which CADIN_fixed <2^SFTADIN * 32767 CADIN = CADIN_fixed / 2^SFTADIN (computed as an integer)
SFTAFFW	INT	0x0291		0 (0x0000)	KFFA and SFTAFFW parameters are used to implement the acceleration feedforward term in the acceleration & load feedforward block. They are used in the multiplication with the target acceleration issued by the reference generator module. The obtained term is added at the output of the speed controller, as an acceleration feedforward term.	KFFA represents the multiplication coefficient. The SFTAFFW parameter represents the shift used to extract the acceleration feedforward component from the target acceleration. These parameters depend on the system model and parameters. Theoretically, they can be computed from these parameters (mainly system inertia). Practically, try-and-error methods can be used to “tune” optimal values for these parameters. Use as start values the default ones (0).

Name	Type	Address	Unit	Default value	Description	Setting
SFTCRT	INT	0x0290		0 (0x0000)	SFTCRT parameter is used to change the current scaling factor, and thus to increase the overall representation accuracy for speed controller coefficients.	SFTCRT parameter can be set between 0 and 6. Shifting of current values does not generate truncation errors, due to the limited, 10 bits accuracy, of the measured motor currents. On the other side, the maximum shift of a controller coefficient is 12. Thus, for values of controller coefficients bigger then 4096, one will be forced to use a truncated value for their representation. But the speed controller coefficients depend also on the current scaling factor. Then, if the current scaling factor can be modified through the use of a supplementary scaling factor (SFTCRT parameter), which do not alter the computational accuracy, an increased range can be used for speed controller coefficients, preserving the same overall mathematical accuracy.
SFTKDP	INT	0x0263		0 (0x0000)	KDP, SFTKDP, KDFP parameters are used to implement the derivative component of the PID position controller.	KDP, SFTKDP, KDFP depend on the derivative component of the PID position controller coefficients, obtained from the tuning, on the controller maximum speed output and on the speed scaling factor. KDP, SFTKDP, KDFP are computed as follows: SFTKDP = the least integer power of 2 for which KD_P_scl < 2^SFTKDP KDP = Kd_P_scl * 32767 / 2^SFTKDP KDFP = filter_D*32767
SFTKFF	INT	0x0270		0 (0x0000)	SFTKFF parameter is used to implement the feedforward term for PI current controller. It represents the global shift for the feedforward term that is implemented in the acceleration and load feedforward block.	SFTKFF depends on the PI speed controller coefficients resulting from the tuning of (Kp_spd, Ki_spd); on the maximum output current of the PI speed controller (I_max); on the current scaling factors; on the speed, acceleration & load feedforward scaling factors. SFTKFF is computed as follows: SFTKFF = the least integer power of 2 for which KFFA < 2^SFTAFFW + KFFL < 2^SFTKFF
SFTKII	INT	0x0274		0 (0x0000)	KII and SFTKII parameters are used to implement the integral component of the PI current controllers.	KII and SFTKII depend on the integral component of the PI current controller coefficients, obtained from the tuning and on the controller maximum voltage outputs (u_d_max, u_q_max). KII and SFTKII are computed as follows: SFTKII = the least integer power of 2 for which KII < 2^SFTKII KII = Ki_C_scl * 32767 / 2^SFTKII

Name	Type	Address	Unit	Default value	Description	Setting
SFTKIP	INT	0x0261		0x0266	KIP, SFTKIP, IMAXP parameters are used to implement the integrative component of the PID position controller.	KIP, SFTKIP, IMAXP depend on the integrative component of the PID position controller coefficients, obtained from the tuning, on the controller maximum speed output (spd_max) and on the speed scaling factor. KDP, SFTKDP, KDFP are computed as follows: SFTKIP = the least integer power of 2 for which KIP < 2^SFTKIP KIP = Ki_P_scl * 32767 / 2^SFTKIP IMAXP = user defined. Defines the saturation value of the integrative part.
SFTKIS	INT	0x026A		0 (0x0000)	KIS, SFTKIS, IMAXS parameters are used to implement the integrative component of the PI speed controller.	KIS, SFTKIS, IMAXS depend on the integrative component of the PI speed controller coefficients, obtained from the tuning, on the controller maximum current output (I_max) and on the current scaling factor. KIS, SFTKIS, IMAXS are computed as follows: SFTKIS = the least integer power of 2 for which KIS < 2^SFTKIS KIS = Ki_S_scl * 32767 / 2^SFTKIS IMAXS = user defined. Defines the saturation value of the integrative part.
SFTKITHT	INT	0x088A			SFTKITHT parameter is the shift of the integral component of the electrical angle estimator which is used during the motionless start procedure.	SFTKITHT has to be set by the user from the "Electric Angle Controller Tuning Test" from Easy SetUp/ EasyMotion Studio
SFTKPI	INT	0x0272		0 (0x0000)	KPI and SFTKPI parameters are used to implement the proportional component of the PI current controllers.	KPI and SFTKPI depend on the proportional component of the PI current controller coefficients, obtained from the tuning and on the controller maximum voltage outputs (u_d_max, u_q_max). KPI and SFTKPI are computed as follows: SFTKPI = the least integer power of 2 for which KPI < 2^SFTKPI KPI = Kp_C_scl * 32767 / 2^SFTKPI
SFTKPP	INT	0x025F		0 (0x0000)	KPP and SFTKPP parameters are used to implement the proportional component of the PID position controllers.	KPP and SFTKPP depend on the proportional component of the PID position controller coefficients, obtained from the tuning and on the controller maximum speed output (spd_max). KPP and SFTKPP are computed as follows: SFTKPP = the least integer power of 2 for which KPP < 2^SFTKPP KPP = Kp_P_scl * 32767 / 2^SFTKPP

Name	Type	Address	Unit	Default value	Description	Setting
SFTKPS	INT	0x268		0 (0x0000)	KPS and SFTKPS parameters are used to implement the proportional component of the PI speed controllers.	KPS and SFTKPS depend on the proportional component of the PI speed controller coefficients, obtained from the tuning and on the controller maximum current output (I_max). KPS and SFTKPS are computed as follows: SFTKPS = the least integer power of 2 for which KPS < 2^SFTKPS KPS = Kp_S_scl * 32767 / 2^SFTKPS
SFTSFFW	INT	0x0292		0 (0x0000)	KFFS and SFTSFFW parameters are used to implement the speed feedforward term added at the input of the PI speed controller. They are used in the multiplication with the target speed issued by the reference generator module. The obtained term is added at the output of the speed controller, as a speed feedforward term.	KFFS represents the multiplication coefficient. The SFTSFFW parameter represents the shift used to extract the speed. These parameters depend on the system model and parameters. Theoretically, they can be computed from these parameters (mainly system viscous friction). Practically, try-and-error methods can be used to “tune” optimal values for these parameters. Use as start values the default ones (0).
SLAVEID	INT	0x311		0 (0x0000)	SLAVEID parameter is used to define the ID of the axes that must get the reference from a master axis. This will be needed in gearing master mode operation.	SLAVEID must be set before starting a master operation mode. The SLAVEID parameter must contain the ID for the slave axis (if one slave is defined in the system), or the ID of the group of axes that will be slaves of the master one. The slave axis or group ID is set in SLAVEID using the standard 16-bit format of the ID field where ID value is set on bits 11 to 4 and bit 12 epresents the axis/group selection.
SLPER	UINT	0x251		20 (0x0014)	The SLPER parameter is used for the setting of the slow control loop (speed/position) sampling period.	SLPER depends on the PWM frequency and on the frequency of the slow control loop (speed / position control loop). SLPER is expressed as a multiple of PWM periods, and is computed using the following relation: SLPER = PWM_freq · Ts_S
START_POS	LONG	0x089C			START_POS parameter is used by the motionless start algorithm to detect motor movement.	START_POS parameter is initialized by the setup
SWNEGLS	LONG	0x084D			SWNEGLS is the parameter that defines the negative software limit switch (maximum negative position limit)	SWNEGLS is set from the Easy SetUp / EasyMotion Studion interface
SWPOSLS	LONG	0x0899			SWPOSLS is the parameter that defines the positive software limit switch (maximum positive position limit)	SWPOSLS is set from the Easy SetUp / EasyMotion Studion interface
T1MAXPROT	UINT	0x0298	[s]	(0xFFFF)	T1MAXPROT parameter represents the temperature value which, when exceeded, will activate the maximum motor temperature protection.	T1MAXPROT depends on the motor sensor type. The chip will set automatically T1MAXPROT with 32767 if the motor temperature sensor is PTC or NTC.

Name	Type	Address	Unit	Default value	Description	Setting
T1ONA	UINT	0x0284	[s]	1 (0xFFFF)	The T1ONA parameter controls the phase A supply time, at the start of a PMSM motor, using initial positioning through supply of phases A and B of the motor, with a controlled DC voltage or current.	T1ONA is related to the time needed to the motor to move when supplying phase A with a controlled DC voltage or current. In order to setup this parameter, a possible method is to use its maximal value (65535) - corresponding to a delay of approximate one second, and to educe it as long as the motor start is performed correctly. When the starting procedure is finished the value of T10NA is halved from the set value.
T1ONB	UINT	0x0285	[s]	-1 (0xFFFF)	The T1ONB parameter controls the phase B supply time, at the start of a PMSM motor, using initial positioning through supply of phases A and B of the motor, with a controlled DC voltage or current.	T1ONB is related to the time needed to the motor to move when supplying phase B with a controlled DC voltage or current. In order to setup this parameter, a possible method is to use its maximal value (65535) - corresponding to a delay of approximate 1 second, and to reduce it as long as the motor start is performed correctly. When the starting procedure is finished the value of T10NB is halved from the set value.
T2MAXPROT	UINT	0x0299	[°C]	(0xFFFF)	T2MAXPROT parameter represents the temperature values which, when exceeded, will activate the maximum drive/ power stage temperature protection.	T2MAXPROT depends on the power converter maximum temperature. T2MAXPROT depends on the protection temperature and on the temperature scale factor. It is expressed as follows: T2MAXPROT = Temp_2_PROT * KTf+OffAt0oC*65472/3.3V
TERRMAX	UINT	0x02C6	[s]	(0xFFFF)	TERRMAX parameter represents the trigger time for the control error protection. If the error between the reference and the feedback exceeds ERRMAX for a time that is greater than this parameter, the corresponding control error flag is set in the Interrupt Status Register – ISR.	TERRMAX depends on the time interval during which the error between the reference and the feedback can exceed the ERRMAX limit and the sampling time of the outer loop. It is expressed as follows: TERRMAX = (time interval for accepted error of the controlled variable) * Ktf
THTST	INT	0x282	[deg]	32768 (0x8000)	THTST parameter is used to implement the test mode for PMSM motors, or to implement the test-operating mode; it represents the electric angle value imposed to the motor.	THTST depends on the reference type (current, voltage). The user sets THTST as required by the test to be performed.
TIMAXPROT	UINT	0x02c4	[s]	(0xFFFF)	TIMAXPROT parameter represents the trigger time for the maximum current protection. If the error between the reference current IQREF and the feedback current IQ exceeds IMAXPROT for a time that is greater than this parameter, the maximum current protection is activated.	TIMAXPROT depends on the time interval during which the error in the current controller can exceed the IMAXPROT limit and the sampling time of the current loop. It is expressed as follows: TIMAXPROT = time_IMAX_PROT · Ktf
TIME_LIMIT	UINT	0x088B	[s]		TIME_LIM parameter defines the duration of the motionless start procedure	TIME_LIM parameter is set by the user from the "Electric Angle Controller Tuning Test" from Easy SetUp/ EasyMotion Studio

Name	Type	Address	Unit	Default value	Description	Setting
TIME0	LONG	0x02BE	[s]	0 (0x00000000)	TIME0 parameter is used to mark the absolute time value, from which a relative time interval will be measured in RTIME variable. The computing formula is: RTIME = ATIME – TIME0 When a relative time event is activated, TIME0 is automatically updated with the value of ATIME - the absolute timer.	TIME0 can be used to memorize ATIME value at a certain moment. RTIME will then provide the time elapsed from this moment.
TINCTST	INT	0x283	[deg/s]	1 (0x0001)	TINCTST parameter is used to implement the starting modes for PMSM, or to implement the test-operating mode; it represents the increment value for the electric angle.	TINCTST depends on the reference type (current, voltage), on the input value and on the scaling factor associated to each type of input. TINCTST is computed as follows: TINCTST = (current/ voltage increment) * (scaling factor)
TONPOSOK	UINT	0x036B	[s]	-A277	TONPOSOK parameter is used to specify the time during which the position error must be lower than POSOKLIM to initiate the position control suspend feature.	TONPOSOK can be set based on the time given to the position to remain in the accepted position error limit. The value must be expressed in number of samplings of the position loop (slow loop).
TSERRMAX	UINT	0x087A	[s]		TSERRMAX parameter contains the value of the time of the speed control error protection	The TSERRMAX parameter is set by the drive
UMAXPROT	UINT	0x029A	[V]	(0xFFFF)	UMAXPROT parameter represents the voltage values which, when exceeded, will activate the over voltage protection.	UMAXPROT depends on the maximum protection voltage and on the voltage scale factor. UMAXPROT is expressed as follows: UMAXPROT = UMAX_PROT * Kuf_m
UMINPROT	UINT	0x029B	[V]	0 (0x0000)	UMINPROT parameter represents the voltage values which, when felled behind, will activate the under voltage protection.	UMINPROT depends on the maximum protection voltage and on the voltage scale factor. UMINPROT is expressed as follows: UMINPROT = UMIN_PROT * Kuf_m
TML Parameters						
ACCPL	INT	0x02FF			ACCPL variable contains the amplitude of the linear Hall sensors	ACCPL is computed at each position loop sampling time interval, based on the measured linear Hall signals
AD0	UINT	0x023C			AD0 variable represents the value of the measured A/D channel corresponding to phase A motor current. It corresponds to channel ADCIN6.	AD0 is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). Its value is used to set the variable IA, used by the control module.
AD1	UINT	0x023D			AD1 variable represents the value of the measured A/D channel corresponding to phase C motor current. It corresponds to channel ADCIN5.	AD1 is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). Its value is used to set the variable IC, used by the control module.

Name	Type	Address	Unit	Default value	Description	Setting
AD2	UINT	0x023E	[V]		AD2 variable represents the value of the measured A/D channel corresponding to an external analogue signal. It corresponds to channel ADCIN2.	Not used
AD3	UINT	0x023F			AD3 variable represents the value of the measured A/D channel. By default, it corresponds to channel ADCIN7.	
AD4	UINT	0x0240	[V]	32736 (0x7FE0)	AD4 variable represents the value of the measured A/D channel corresponding to DC link voltage. It is used if the DC link voltage compensation is activated (indicated by setting corresponding bit “UDC compensation” in OSR register). By default, it corresponds to channel ADCIN4.	AD4 is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). The control module uses this value, if the DC link voltage compensation is activated
AD5	UINT	0x0241	[V]		AD5 variable represents the value of the measured A/D channel corresponding to an analogue reference signal (indicated by using corresponding TML mode setup instructions). By default, it corresponds to channel ADCIN3.	AD5 is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). The control module uses this value, if an analogue reference must be used by the motion system
AD6	UINT	0x0242			AD6 variable represents the value of the measured A/D channel. By default, it corresponds to channel ADCIN0.	
AD7	UINT	0x243	[°C]		AD7 variable represents the value of the measured A/D channel corresponding to drive temperature sensor. By default, it corresponds to channel ADCIN1.	AD7 is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). The control module uses this value in the protections section, to check if parameter T2MAXPROT limit is not exceeded.
AD8	UINT	0x080B			AD8 variable represents the value of the measured A/D channel corresponding to drive temperature sensor..	AD8 is updated at each current control loop sampling time interval
APOS	LONG	0x0228	[rot]		APOS variable contains the value of the actual position of the motor, represented as a 32-bit integer variable. Depending on the position sensor, its value is directly set with the value read from the sensor, or the position increment between two sampling moments is added to the actual position value.	APOS is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). If the APOS variable wraps around, a “wrap around” bit is set in the MSR and ISR registers. APOS can be read at any moment using TML instructions or the data tracer capabilities of TML environment.
APOS_LD	LONG	0x0228	[rot]		APOS_LD variable contains the value of the actual position of the load. APOS_LD is a copy of APOS variable. For details see the APOS variable description	APOS_LD is updated at each speed / position control loop sampling time interval
APOS_MT	LONG	0x0988	[rot]		APOS_MT variable contains the value of the actual position of the motor.	APOS_MT is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter)
APOS2	LONG	0x081C			APOS2 variable contains the value of the actual position of the master.	APOS2 is computed based on master speed MSPD at each speed / position control loop sampling time interval.

Name	Type	Address	Unit	Default value	Description	Setting
ASPD	FIXED	0x022C	[rpm]		ASPD variable contains the value of the actual speed of the motor, represented as a 32-bit fixed-point value (16 bits integer part, 16 bits fractional part). Depending on the speed sensor, its value is directly set with the value read from the sensor, or a speed estimate is computed, using position variation between two sampling moments, or using the time length of a position sensor pulse.	ASPD is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). ASPD can be read at any moment using TML instructions or the data tracer capabilities of TML environment.
ASPD_LD	FIXED	0x098A	[rpm]		ASPD_LD variable contains the value of the actual speed of the load. See the ASPD variable.	ASPD_LD is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). ASPD_LD can be read at any moment using TML instructions or the data tracer capabilities of TML environment.
ASPD_MT	FIXED	0x022C	[rpm]		ASPD_MT variable contains the value of the actual speed of the motor. See the ASPD variable.	ASPD_MT is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). ASPD_MT can be read at any moment using TML instructions or the data tracer capabilities of TML environment.
ATIME	LONG	0x02C0	[s]		ATIME variable represents the value of the motion system actual time.	ATIME is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). ATIME is incremented with 1 at each update. This variable is used to implement time-based TML events.
AXISONACTIVE	UINT	0x037A			AXISONACTIVE variable signals the current execution of the AXISON instruction. Remark: While AXISONACTIVE=1 the UPD instruction is ignored	AXISONACTIVE is 1 during the execution of the AXISON instruction
BETA	LONG	0x0293			BETA variable it is used by the slave axes to synchronize with the master, when the reference is send via a communication channel.	BETA contains the reference send by the master. If the content of BETA it is not changed during one slow loop the slave inserts a variable delay in the slow loop (maximum one sampling). BETA is not used when the master send the reference from the second encoder.
BOOST	INT	0x027D			BOOST variable contain the value of the voltage uset during the initial alignment of the stepper motors	BOOST variable value is automatically set based on the motor electrichal parameters by Easy SetUp / EasyMotion Studio.
CAMX	FIXED	0x0903			Cam input correction factor, Cam input X (TML variable CAMINPUT) is: X = CAMINPUT = MPOS0 * CAMX where MPOS0 = MREF - CAMOFF	CAMX is set by the drive using the following formula: X = CAMINPUT = MPOS0 * CAMX where MPOS0 = MREF - CAMOFF
CAMY	FIXED	0x0905			Cam output correction factor. Cam table output Y is: Y = f(X) * CAMY	CAMY is set by the drive using the following formula: Y = f(X) * CAMY

Name	Type	Address	Unit	Default value	Description	Setting
CAPPOS	LONG	0x02BC	[rot]		CAPPOS variable represents a value of the measured motor position sensor, captured at the occurrence of an external capture on the DSP.	CAPPOS is set as the value of the position sensor, captured on one of the capture inputs Z1+ and Z1- of the drive (these capture inputs corresponds to DSP pins IOPA3/CAP1, respectively IOPA4/CAP2). It can be used to correct the position information, related to an external I/O input device (switch), installed in a specific position of the motion structure (homing functions, for example).
CAPPOS2	LONG	0x081E			CAPPOS2 variable represents the value of the master position read at second encoder inputs, captured at the occurrence of an external capture on the DSP.	CAPPOS2 is set as the value of the second position sensor, captured on one of the capture pins Z2+ and Z2- of the drive (these capture inputs corresponds to DSP pins IOPE7/CAP4, respectively IOPF0/CAP5).
CBR	UINT	0x030D			CBR variable contain the CAN baud-rate value	CBR value can be set from the Drive Setup dialog. The CBR default value for Technosoft drives is 500 kbits/s
CDEC	FIXED	0x0858	[rad/s^2]	0.5 IU	CDEC variable contain the deceleration rate for the quick stop mode (when a limit switch is hit)	CDEC variable calue is set from the Motion programing dialogs
CDREF	FIXED	0x02A4			CDREF variable is used for contouring references and represents the reference increment to be used for the actual contouring segment.	CDREF is updated with the last value of the reference increment, as defined with a SEG instruction.
COSRES	INT	0x227			COSRES variable contains the computed value of the resolver cosine signal	COSRES variable is computed each current loop sampling time interval.
COSTH	INT	0x226			COSTH presents the values of the cosine of the electric angle position.	COSTH is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). The control module uses this value for vector control of AC motors, for coordinate transformations.
CRTERR	INT	0x231	[A]		CRTERR variable represents the value of current error, on the Q axis (torque axis) of the motor, computed as the difference between the reference and the measured current in the Q axis of the motor.	CRTERR is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). The control module uses these values for current control loop implementation. This value is related to the Q (torque) axis current component of the motor.
CTIME	UINT	0x02A6	[s]		CTIME variable is used for contouring references and represents the number of samplings to be used for the actual contouring segment.	CTIME is updated with the last value of the number of samplings, as defined with a SEG instruction.
DIGIN_ACTIVE_LE\	UINT	0x090C			DIGIN_ACTIVE_LEVEL variable specifies the active level for limit switches and enable input	DIGIN_ACTIVE_LEVEL value is set from the Drive Setup dialog, based on the user selection
DIGIN_INVERSION_	UINT	0x090A			DIGIN_INVERSION_MASK variable contain the inversion mask for the digital inputs of the drive	A bit set to 1 in this mask, means that the corresponding input is inverted.
DIGOUT_INVERSIC	UINT	0x090B			DIGIN_INVERSION_MASK variable contain the inversion mask for the drive outputs	A bit set to 1 in this mask, means that the corresponding output is inverted.

Name	Type	Address	Unit	Default value	Description	Setting
E_LEVEL_AD5	INT	0x0870			DBTLIM variable contains the value of the dad band limit applied on the analog reference	DBTLIM variable value is set in the Drive Setup dialog. Dead-band range in internal units computed with formula: E_LEVEL_AD5 = DB_Range * 65472 / InputRange, where DB_Range – is the desired dead-band range expressed in V InputRange – is the drive/motor analogue input range expressed in V.
ELPOS	INT	0x221			ELPOS variable represents the value of the electric angle position as measured from the position sensor.	ELPOS is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). The control module uses this value to compute the scaled field position angle – THETA, by multiplying it with the ENC2THL parameter.
ELPOSL	LONG	0x080E			ELPOSL variable represents the value of the electric angle position, on 32 bits, as measured from the position sensor.	ELPOSL is updated at each current control loop sampling time interval. See ELPOS variable description
ELRES	UINT	0x279			ELRES variable represent the number of the encoder counts for one pole pair	ELRES variable value is automatically set by EasyMotion Studio / Easy SetUp based on the encoder resolution.
ELRESL	LONG	0x0875			ELRESL variable represent the number of the encoder counts, on 32 bits, for one pole pair	ELRESL variable value is automatically set by EasyMotion Studio / Easy SetUp based on the encoder resolution.
EREFP	LONG	0x02A8	[rot]		EREFP variable contain the position referance writen by an external device, in the external mode on-line.	The EREFP variable can be modified from a TML program or online by an external device
EREFS	FIXED	0x02A8	[rpm]		ERFS is a TML variable where an external device writes the speed reference in external mode on-line. Measured in speed units	The EREFS variable can be modified from a TML program or online by an external device
EREFT	INT	0x02A9	[A]		EREFT is a TML variable where an external device writes the torque reference in external mode on-line. Measured in current units	The EREFT variable can be modified from a TML program or online by an external device
EREFV	INT	0x02A9	[V]		EREFV is a TML variable where an external device writes the voltage reference in external mode on-line. Measured in voltage command units	The EREFV variable can be modified from a TML program or online by an external device
FIRST_POS	LONG	0x089E			FIRST_POS variable is used by the motionless start algorithm to detect motor movement.	FIRST_POS parameter is initialized by the setup
FLAGUV	INT	0x02FB			FLAGUV variable is used to detect the moment when the motor voltage is applied to the drive	FLAGUV is set during the Setup and its value is modified when the motor voltage is applied to the drive

Name	Type	Address	Unit	Default value	Description	Setting
GEARSLAVE	INT	0x256		1 (0x0001)	GEAR, GEARMASTER and GEARSLAVE parameters are used to set the gearing factor used by an axis when operating in slave gearing mode. GEAR is multiplied with the master position increment (MREF – MPOS0), to give the target position increment of the slave. GEARSLAVE and GEARMASTER represent the numerator and denominator of the Slave / Master ratio. GEARSLAVE is a signed integer, while GEARMASTER is an unsigned integer. GEARSLAVE sign indicates the direction of movement: positive – same as the master, negative – reversed to the master. GEAR is a fixed value containing the result of the gear ratio i.e. the result of the division GEARSLAVE / ARMASTER. In order to eliminate any cumulative errors the electronic gearing slave mode includes an automatic compensation of the round off errors when the gear ratio has an irrational value like: Slave = 1, Master = 3, giving a ratio of 1/3 = 0.33333 which can't be represented exactly.	The user must set GEAR before starting the operation of the axis in the slave-gearing mode. This parameter can be set directly using an assignment instruction, or is computed based on GEARMASTER and GEARSLAVE values.
HALL	INT	0x0227			HALL variable codifies the position information read from Hall sensors, in the case of BLDC motors.	HALL is set corresponding to the position information read from the Hall sensors and Hall configuration
HOMEPOS	LONG	0x0992	[rot]		HOMPOS variable contains the new home position value	HOMPOS variable is set at the end of the homing procedure
HOMINGTABLE_PC	UINT	0x096A			HOMINGTABLE_POINTER - *will be added later*	
I2TINT_D	ULONG	0x097A			I2TINT_D variable contains the current value of the drive I2T integral	I2TINT_D varianle value is updated at each slow loop sampling time interval
I2TINT_M	ULONG	0x0817			I2TINT_M variable contains the current value of the motor I2T integral	I2TINT_M varianle value is updated at each slow loop sampling time interval
I2TINTLIM_D	ULONG	0x0980			I2TINTLIM_D is the maximum value of the I2t drive thermal protection; exceeding this value triggers the corresponding bit in PCR register.	I2TINTLIM_D is computed based on over current level, time at over current (I_I2t respectively t_I2t) specific for each driver and nominal current
I2TINTLIM_M	ULONG	0x0815		0 (0x00000000)	I2TINTLIM_M is the maximum value of the I2t motor thermal protection; exceeding this value triggers the corresponding bit in PCR register.	I2TINTLIM_M is computed based on over current level, time at over current (I_I2t respectively t_I2t) defined by the user and nominal current
IA	INT	0x239	[A]		IA, IB, IC variables represent the values of motor currents. These values are normally represented as signed fractional numbers. Only analog to digital current measurement is	IA, IB, IC are updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). Two currents, IA
IB	INT	0x023A	[A]			
IC	INT	0x023B	[A]			
ID	INT	0x234	[A]		ID variable represents the value of the measured field current component. The control module uses this value for vector control of AC motors.	ID is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). It is obtained from the measured motor currents, as projected on the field axis of the motor (for vector control schemes).

Name	Type	Address	Unit	Default value	Description	Setting
IDMAX	INT	0x027B	[A]		IDMAX variable represents the value of the maximum allowed current.	IDMAX is set by the current control loop
IDMIN	INT	0x027C	[A]		IDMIN variable represents the value of the mimimum allowed current.	IDMIN is set by the current control loop
IDREF	INT	0x233	[A]		IDREF variable represents the value of the reference field current component. The control module uses this value for vector control of AC motors.	IDREF is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). It is obtained as the output of the field control block, and is usually constant up to the rated speed of the motor, and is modified at speeds higher than the rated value, in the field weakening operating region.
IDRSTEP	INT	0x027B	[A]		IDRSTEP variable contains the value of the current used by the drive to control the step motor in open-loop.	The run current must be set less or equal with the motor nominal current.
IQ	INT	0x230	[A]		IQ variable represents the value of the measured torque current component. The control module uses this value for vector control of AC motors.	IQ is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). It is obtained from the measured motor currents, as projected on the torque axis of the motor (for vector control schemes).
IQREF	INT	0x022F	[A]		IQREF variable represents the value of the reference torque current component. The control module uses this value for vector control of AC motors.	IQREF is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). It is obtained as the output of the outer loop connected to the current controller on the Q axis, i.e.: speed controller (if configured in the motion structure); position / user controller (if no speed controller was configured in the motion structure); reference generator (if no speed or position / user loops were configured).
IQREFFILTERED	INT	0x0987	[A]		IQREFFILTERED variable contains the value of the filtred current referance	IQREFFILTERED is updated at each current loop sampling time interval.
LEVEL_AD5	INT	0x086F			LEVEL_AD5 variable represent the dead-band point in internal units	LEVEL_AD5 is computed with the folowinf formula: LEVEL_AD5=(DB_Point–InputLow)*65472/InputRange
LOG_PTR	UINT	0x0365			LOG_PTR variable is used internally by the logger	The LOG_PTR is set by the drive
LSACTIVE	INT	0x0832		0	LSACRIVE variable enable/disable the functionality of the limit switches that is to prevent the movement outside the working area.	The LSACRIVE variable can be set and retrieved anywhere in a TML instruction
MASTERID	INT	0x0927			MASTERID variable drovides the host ID (address)	MASTERID variable value is computed usinf the following formula: MASTERID = host ID * 16 + 1
MAX_CURRENT	INT	0x0890	[A]		MAX_CURRENT represents the current used during the motionless start procedure	MAX_CURRENT is set by the user from the "Electric Angle Controller Tuning Test" from Easy SetUp/ EasyMotion Studio
MECRES	UINT	0x0277			MECRES variable contains the mechanical resolution of the position feedback	MECRES depends on the position sensor type

Name	Type	Address	Unit	Default value	Description	Setting
MER	UINT	0x08FC			MREF variable stores the value of the master position, in electronic gearing multi-axis configurations. The master updates MREF variable on all slave axes, once at each position / speed loop sampling period	MREF is automatically set and updated by the master
MOTORON	INT	0x0312			MOTORON variable contains the sate of the control loops	The MOTORON variable value is initialized by setup
MOVESTATE	INT	0x0898			MOVESTATE variable indicates if the motionless start procedure completed successfully	MOVESTATE is set 1 if the motionless start procedure completed successfully.
MREF	LONG	0x02AA			MREF variable contains the master position received or computed by the slave(s). Measured in master position units	The MREF variable value is set by the drive based on the information received from the master
MSPD	INT	0x0820			MSPD variable stores the value of the master speed, in electronic gearing	MSPD is computed by the slave axis, at each speed / position control loop sampling time interval. If the master sends position information via communication channels the speed is computed as follows: MSPD=MREF-MPOS0 If the slave axis receives position reference from second encoder, the speed is computed as the difference between two consecutive captured values from timer 4.
MTSTYPE	UINT	0x028C		0 (0x0000)	MTSTYPE is used to indicate the type of the motor temperature sensor, PTC or NTC.	For sensors with positive temperature coefficient (PTC) MTSTYPE = 1 and for sensors with negative temperature coefficient (NTC) MTSTYPE = 0.
PHASEADV	INT	0x0257		0x1FFF	PHASEADV it is used in field position computation for high speed PMSM applications, to eliminate measurement and computation delays.	PHASEADV must be set before AXISON instruction.
POSERR	INT	0x022A	[rot]	0 (0x0000)	POSERR variable represents the value of motor position error, computed as the difference between the reference and the measured position of the motor.	POSERR is updated at each speed / position control loop sampling time interval (i.e.the slow sampling time, see SLPER parameter). The control module uses these values for position control loop implementation.
POSINC	INT	0x0222		0 (0x0000)	POSINC variable contents the position increment value	POSINC is updated each position control loop
POSTRIGG1	LONG	0x091A	[rot]		POSTRIGG1 variable contains the position of the trigger 1	POSTRIGG1 value is update at each speed / position control loop sampling time
POSTRIGG2	LONG	0x091C	[rot]		POSTRIGG2 variable contains the position of the trigger 2	POSTRIGG1 value is update at each speed / position control loop sampling time
POSTRIGG3	LONG	0x091E	[rot]		POSTRIGG3 variable contains the position of the trigger 3	POSTRIGG1 value is update at each speed / position control loop sampling time
POSTRIGG4	LONG	0x0920	[rot]		POSTRIGG4 variable contains the position of the trigger 4	POSTRIGG1 value is update at each speed / position control loop sampling time
PROD	LONG	0x030E			PROD variable contains the result of the last TML multiply operation performed, including the left or right shift.	PROD is updated after each TML multiply operation. Either the 32MSB of the PROD or the 32LSB may be copied into a 32-bit variable using a 32-bit assignment instruction. Use PROD(H) mnemonic to access the 32MSB of the result and PROD or PROD(L) for the 32LSB.

Name	Type	Address	Unit	Default value	Description	Setting
PRODH	LONG	0x030F			PRODH variable contains the 32 most significant bits of the product register	PRODH is updated after each TML multiply operation
PVTBUFBEGIN	INT	0x0864	0		PVTBUFBEGIN Specifies the start address of the PT buffer	PVTBUFLEN is initialized by the setup
PVTBUFLEN	INT	0x0865	0		PVTBUFLEN specifies the PT buffer length expressed in PT points	PVTBUFLEN is initialized by the setup
PVTMODE	UINT	0x086B	0		PVTMODE contains the PVT operation mode as was set with the SETPVT command	PVTMODE is set based on SETPVT commands
PVTPOS0	LONG	0x0869	0	0	PVTPOS0 variable specifies the initial position (for absolute mode) from which to start computing the distance to move up to the first PT point	PVTBUFLEN is initialized by the setup
PVTSENDOFF	INT	0x092B	0	0	PVTSENDOFF is used to enable/disable the message transmission durind the PT mode	When the PVTSRNDOFF is set to 1, the transmission of messages, during the PT mode is disabled.
PVTSTS	UINT	0x0863	0		PVTSTS variable contents the PT motion status	PVTBUFLEN is initialized by the setup
REFTST_A	INT	0x0281	[A]		REFTST_A variable contents the value of the maximum current referance	REFTST_A is set by the drive based on the Drive Setup dialog setings
REFTST_V	INT	0x0281	[V]		REFTST_A variable contents the value of the maximum voltage referance	REFTST_A is set by the drive based on the Drive Setup dialog setings
RINCTST_A	INT	0x280	[A/s]		RINCTST_V variable contents the value of the current referance increment	RINCTST_A is update at each slow-loop sampling period
RINCTST_V	INT	0x0280	[V/s]		RINCTST_V variable contents the value of the voltage referance increment	RINCTST_V is update at each slow-loop sampling period
RPOS	LONG	0x02BA			RPOS variable represents the difference between the actual position of the motor APOS and the origin of relative position measurement, stored in variable POS0.	RPOS is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). It is used in relative position events tests, in order to decide if an event must be set or not (in RPO or RPU instructions). Also, it can be used to test if the motor has reached a specific relative position as compared with a reference point.
RTIME	LONG	0x02C2	[s]		RTIME variable represents the difference between the actual motion system absolute time ATIME and the origin of the relative time measurement, stored in variable TIME0.	RTIME is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). It can be used to test the reaching of a specific relative delay as compared with a reference time moment.
SFTKPTHT	INT	0x0889			SFTKPTHT variable contain the shift factor for the electrical angle estimator	The SFTKPTHT is set by the drive
SINRES	INT	0x213			SINRES variable contains the computed value of the resolver sine signal	SINRES variable is computed each current loop sampling time interval.
SINTH	INT	0x225			SINTH variable represent the values of the sine of the electric angle position.	SINTH is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). The control module uses this value for vector control of AC motors, for coordinate transformations.

Name	Type	Address	Unit	Default value	Description	Setting
SPDERR	INT	0x022E	[rpm]		SPDERR variable represents the value of motor speed error, computed as the difference between the reference and the measured speed of the motor.	SPDERR is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). The control module uses these values for speed control loop implementation.
SPDREF	INT	0x022B	[rpm]		SPDREF variable represents the value of the motor speed reference.	SPDREF is updated at each speed control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). It is obtained as the output of the outer loop connected to the speed controller, i.e. position / user controller, or reference generator (if no position / user loops were configured).
TACC	FIXED	0x02B6	[rad/s^2]		TACC variable represents the value of the reference acceleration.	TACC is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). It is computed in the reference generator module, for position / speed profile or contouring modes. TACC is used by the acceleration feedforward term in the speed controller.
THETA	INT	0x0224	[deg]		THETA variable represents the value of the electric angle position.	THETA is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). The control module uses these values for vector control of AC motors, for coordinate transformations.
THETAINC	INT	0x020E	[deg/s]		THETAINC variable represents the increment of the electric angle as measured from the position sensor, before scaling	THETAINC is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). The control module uses this value to compute the scaled field position angle – THETA, by adding it to ELPOS variable and then multiplying the result with the ENC2THL parameter.
TIMESBC	UINT	0x028B	[s]		TIMESBC variable contain the value of the standby current, which is the current value used when the motor is at standstill.	TIMESBC variable value is set in the Drive Setup dialog. In the same dialog is requested also a time from reaching the commanded position, after which the drive switches from the run current to the standby current.
TJERK	LONG	0x08D2	[s]		TJERK variable contains the Jrek time value needed to accelerate from zero up to the CACC parameter value.	TJERK variable value is set in the Drive Setup dialog
TMLINTPER	UINT	0x0983	[s]		TMLINTPER variable represent the value of the TML interrupt period	TMLINTPER can be set and retrieve anywhere in a TML application
TPOS	LONG	0x02B2	[rot]		TPOS variable represents the value of the reference position	TPOS is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). It is computed in the reference generator module for all positioning modes and also during speed profile or contouring modes.

Name	Type	Address	Unit	Default value	Description	Setting
TPOSFAST	LONG	0x0959	[rot]		TPOSFAST variable represents the value of the reference position when the position control is executed in the fast control loop (current loop)	TPOSFAST is updated at each current control loop sampling time interval.
TREF	LONG	0x02AE			TREF variable represents the value of the reference value, for all internal reference types, in any of the accepted motion modes (position, speed, torque or voltage mode).	TREF is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). It is computed in the reference generator module and is used in the control module as an input to the outer controller of the motion structure.
TREF_fixed	FIXED	0x02AE			TREF_fixed variable represents the value of the reference value, in fixed point, for all internal reference types, in any of the accepted motion modes (position, speed, torque or voltage mode).	TREF is updated at each speed / position control loop sampling time interval. See the TREF variable description
TSPD	FIXED	0x02B4	[rpm]		TSPD variable represents the value of the reference speed	TSPD is updated at each speed / position control loop sampling time interval (i.e. the slow sampling time, see SLPER parameter). It is computed in the reference generator module, for all speed modes and also during position profile or contouring modes.
UAREF	INT	0x0236	[V]		UAREF, UBREF, UCREF variables represent the values of the command motor voltages. These values are the output of the inner active control loop (if any is defined) or, for voltage operating mode, the output of the reference generator module.	UAREF, UBREF, UCREF are updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). They are computed depending on the motion mode, specifically depending on the active control loops. Corresponding to their values, the PWM outputs are set and will command the power transistors.
UBREF	INT	0x0237	[V]			
UCREF	INT	0x0238	[V]			
UDREF	INT	0x0235	[V]		UDREF variable represents the value of the reference field voltage component. The control module uses this value for vector control of AC motors.	UDREF is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). It is obtained as the output of the inner loop connected to the D axis, i.e.: D-axis current controller (if configured in the motion structure); field control block (if no current controllers are configured in the motion structure).
UQREF	INT	0x0232	[V]		UQREF variable represents the value of the reference field voltage component. The control module uses this value for vector control of AC motors.	UQREF is updated at each current control loop sampling time interval (i.e. the fast sampling time, see CLPER parameter). It is obtained as the output of the inner loop connected to the Q axis, i.e.: Q-axis current controller (if configured in the motion structure); speed controller (if no current controllers are configured in the motion structure); position / user controller (if neither current, nor speed controller were configured in the motion structure); reference generator (if no current, speed or position / user loops were configured).

Name	Type	Address	Unit	Default value	Description	Setting
VAR_I1	INT	0x0366			VAR_I1, VAR_I2 and VAR_I3 variables represent predefined integer variables used during system setup, i.e. interrupts service routines.	The three variables can be set and retrieve anywhere in a TML instruction accepting an integer variable as an operand, as long they do not interfere with a predefined functionality.
VAR_I2	INT	0x0367				
VAR_I3	INT	0x0979				
VAR_LF	LONG	0x0368			VAR_LF variable represents a predefined variable used during the system setup, as a fixed or long variable.	The VAR_LF variable can be set and retrieved anywhere in a TML instruction accepting a fixed or long variable as an operand, as long they do not interfere with a predefined functionality.
VAR_LF_fixed	FIXED	0x0368			VAR_LF_fixed variable represents a fixed predefined variable used during the system setup, as a fixed or long variable.	The VAR_LF_fixed variable can be set and retrieved anywhere in a TML instruction accepting a fixed variable as an operand
VAR_LF_long	LONG	0x0368			VAR_LF_log variable represents a predefined long variable used during the system setup, as a fixed or long variable.	The VAR_LF variable can be set and retrieved anywhere in a TML instruction accepting a long variable as an operand
VAR_U1 VAR_U2 VAR_U3	UINT UINT UINT	0x0366 0x0367 0x0979			VAR_U1, VAR_U2 and VAR_U3 variables represents predefined unsiigned integer variables used during system setup, i.e. interrupts service routines.	The three variables can be set and retrieve anywhere in a TML instruction accepting an integer variable as an operand, as long they do not interfere with a predefined functionality.
VDCN	UINT	0x025A		46000 (0XB3B)	VDCN parameter is used to implement the compensation of DC-link/supply voltage (Vdc) variation by changing the PWM command. Its value represents the nominal Vdc value.	VDCN depends on the voltage scaling factor and on the rated DC bus voltage values: VDCN = Vdc * Kuf_m
ZAOFF	UINT	0x0289			ZAOFF variable contain the offest between the position feedback and the A phase	The ZAOFF is updated at each current control loop sampling time interval
HOMESPD	FIXED	0x0994	[rpm]		HOMESPD variable contains the low speed command for the final approach towards the home position	HOMESPD value is set in the homing procedure
TONPOSOK_FC	UINT	0x09FD	[s]	(0xFFFF)	TONPOSOK_FC parameter is used to specify the time during which the position error must be lower than POSOKLIM_FC to initiate the position control suspend feature. This parameter is used in FreezeControl.	TONPOSOK_FC can be set based on the time given to the position to remain in the accepted position error limit. The value must be expressed in number of samplings of the position loop (slow loop).
POSOKLIM_FC	UINT	0x09FC	[rot]	(0x0001)	POSOKLIM_FC parameter is used to suspend (to lock) the execution of the control loops if the position error is lower than this value for a time greater than the one specified with the TONPOSOK_FC parameter. This special feature of the position control loop can be useful in such applications where the vibrations of the motor are undesired from the moment when the load has arrived to its target position. This parameter is used in FreezeControl.	POSOKLIM_FC can be set based on the application requirements and the position sensor resolution. If the position sensor resolution is good enough and the application accepts a certain position error, POSOKLIM_FC could be set to a value slightly lower than the accepted position error.