DM-J4340P-2EC Geared Motor

User Manual V1.0 2025.01.10

Disclaimer

Thank you for purchasing the DM-J4340P-2EC geared motor (hereinafter referred to as "the motor").

Before using this product, please carefully read and follow the instructions in this document and all safety guidelines provided. Failure to do so may result in harm to yourself or others, or cause damage to the product or surrounding property.

By using this product, you are deemed to have read this document thoroughly and to have understood, acknowledged, and accepted all the terms and contents of this document and any related materials.

You agree to use this product only for legitimate purposes and assume full responsibility for its usage and any resulting consequences.

The manufacturer shall not be held liable for any damage, injury, or legal responsibility caused directly or indirectly by the use of this product.

Precautions

1. Please strictly operate the motor within the specified working environment and the maximum allowable winding temperature range. Failure to do so may result in permanent and irreversible damage to the product.
2. Prevent foreign objects from entering the rotor; otherwise, abnormal rotor operation may occur.
3. Before use, check whether all components are intact. Do not use the product if any parts are missing, aged, or damaged.
4. Ensure correct wiring and that the motor is installed properly and securely.
5. Do not touch the electronic rotor section during operation to avoid accidents. The motor may become hot during high-torque output; be cautious to prevent burns.
6. Users must not disassemble the motor without authorization, as this may affect control accuracy or lead to abnormal operation.

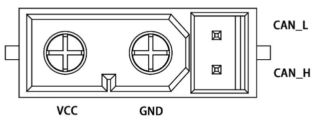
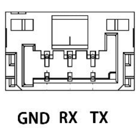
Motor Features

1.Dual encoders provide single-turn absolute position output on the output shaft, ensuring no loss of position even after power failure.
2.Integrated motor and driver design with a compact and highly integrated structure.
3.Supports upper-computer visual debugging and firmware upgrades.
4.Capable of reporting motor speed, position, torque, and temperature via the CAN bus.
5.Equipped with dual temperature protection.
6.Supports trapezoidal acceleration and deceleration in position control mode.

Packing List

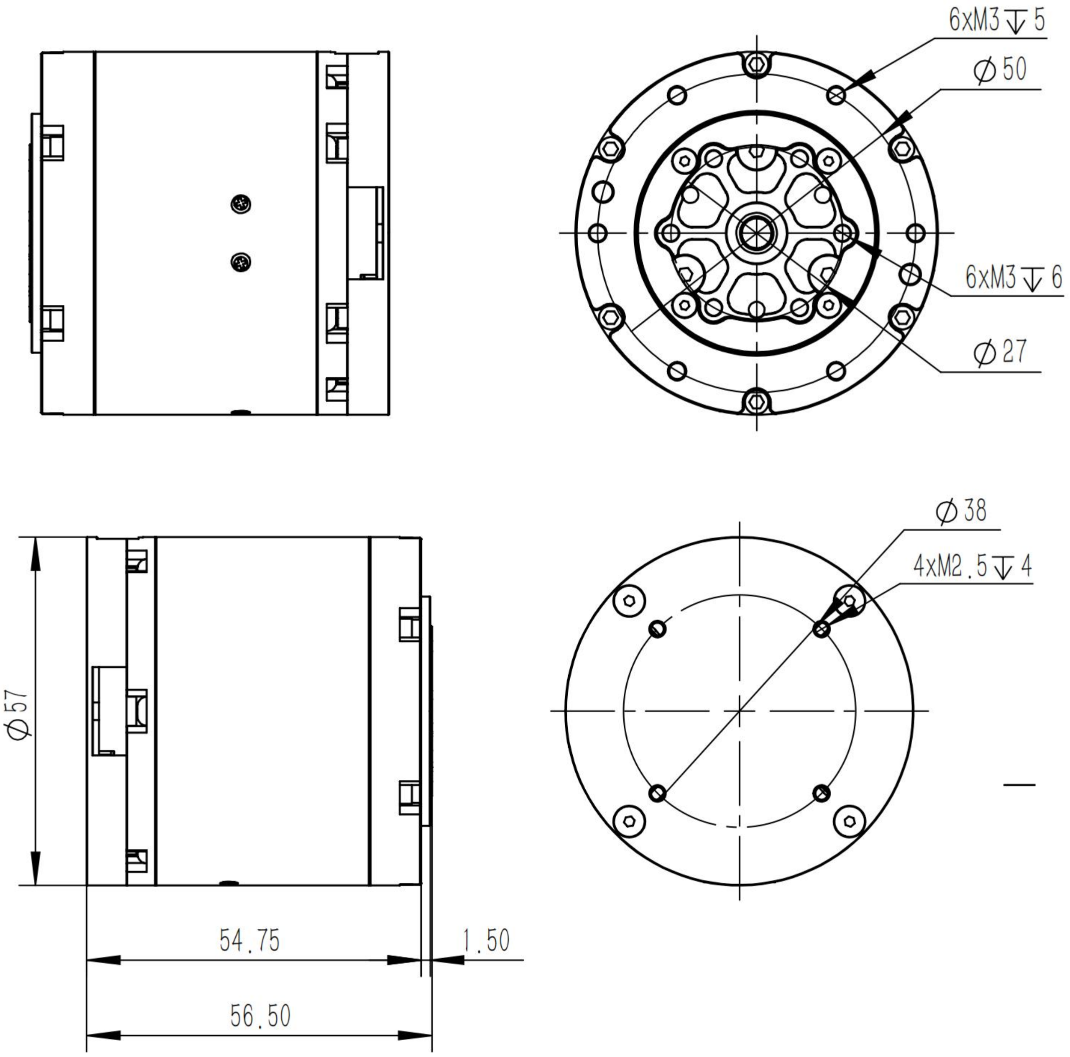
1. Motor (with driver) ×1
2. Power cable (with CAN communication terminal): XT30 (2+2)-F plug, single-ended cable (100mm) ×1
3. Debug serial signal cable: GH1.25 3-pin cable (opposite side, 300mm) ×1

Interface and Wiring Description

<table>
<tr><th>Specific Name – Number</th><th>Interface Label</th><th>Description</th></tr>
<tr><td>Power Interface – 1
(with integrated CAN Communication Terminal)</td><td rowspan="2">
</td><td rowspan="2">1. Connect the power supply using the XT30(2+2)-F plug power cable. The rated voltage is 24V, which powers the motor.
2. Use the CAN communication terminal to connect to an external control device. It can receive CAN control commands and transmit motor status information.
3. The motor is equipped with two power interfaces. Either interface can be used independently or connected in series with multiple motors for streamlined wiring.</td></tr>
<tr><td>Power Interface – 2
(with integrated CAN Communication Terminal)</td></tr>
<tr><td>Debug Serial Port – 3</td><td>
</td><td>Connect via GH1.25 3-pin cable. Use a USB to CAN debugging tool (or a general USB-to-serial module) to connect to a PC for parameter configuration and firmware upgrades via the debugging assistant.</td></tr>
</table>

Motor Dimensions and Installation

Please refer to the motor mounting hole dimensions and positions to install the motor onto the corresponding equipment.



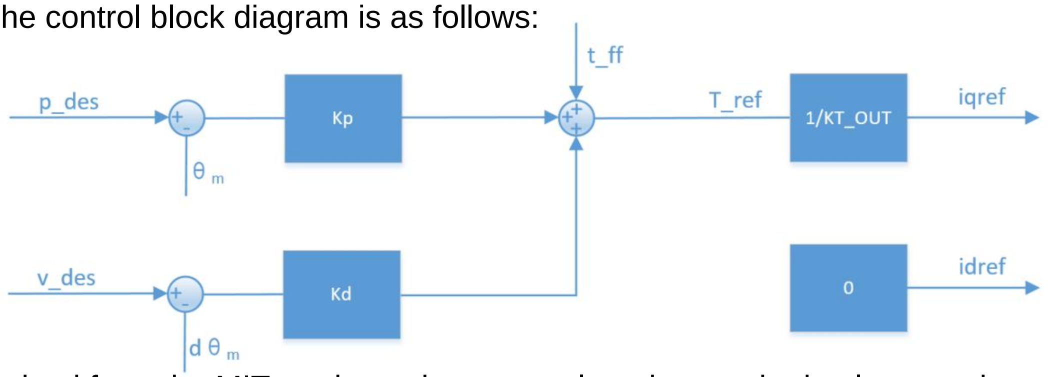
Indicator Light Status

Normal Status	Green-Solid On	ERR bit is 1, indicating enabled mode and normal operating status
	Red-Solid On	ERR bit is 0, indicating disabled mode
Abnormal Status	Red – Blinking	Indicates a fault. Corresponding fault types include: 8 – Overvoltage 9 – Undervoltage A – Overcurrent B – MOS Overtemperature C – Motor Coil Overtemperature D – Communication Loss E – Overload You can check the fault type via the feedback frame or through the debugging assistant interface.

Operating Modes

❖ MIT Mode

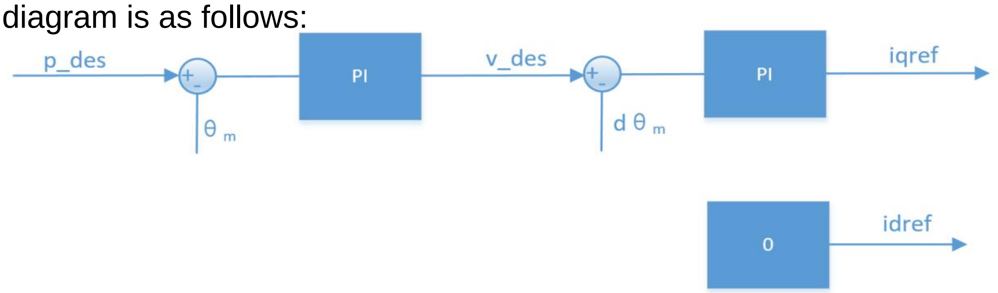
MIT mode is designed to be compatible with the original MIT mode. It allows seamless switching while enabling flexible configuration of control ranges (P _ MAX, V_MAX, T_MAX). The ESC converts received CAN data into control variables to calculate the torque value, which serves as the current reference for the current loop. The current loop then regulates to achieve the specified
torque current.
The control block diagram is as follows:



Derived from the MIT mode, various control modes can be implemented.
For example:
When kp = 0 and kd ≠ 0, setting v_des enables constant speed rotation;
When kp = 0 and kd = 0, setting t_ff enables constant torque output.
Note:
When controlling position, kd must not be set to 0, otherwise it may cause motor oscillation or even loss of control.

❖ Position-Velocity Mode

The position-velocity mode uses a three-loop cascaded control scheme. The position loop serves as the outermost loop, and its output is the setpoint for the velocity loop. The velocity loop's output is used as the setpoint for the inner current loop, which controls the actual current output. The control block diagram is as follows:

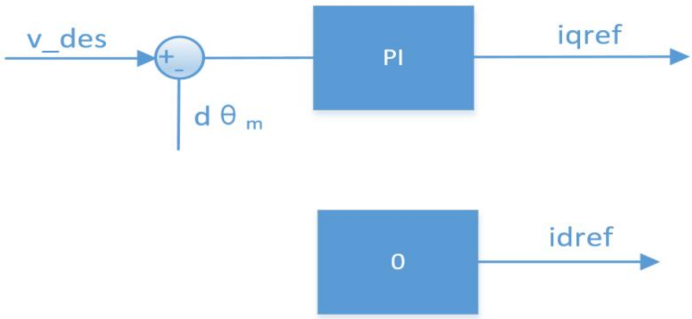


p_des is the target position for control, and v_des limits the maximum absolute speed during motion.
When using the recommended control parameters from the debugging assistant in position cascaded mode, high control accuracy can be achieved with relatively smooth operation. However, the response time is relatively longer.
Configurable parameters include not only v_des but also acceleration/ deceleration settings. If additional oscillations occur during control, increasing the acceleration/deceleration values may help.

Note:
p_des and v_des are in units of rad and rad/s respectively, and both use the float data type.

❖ Velocity Mode

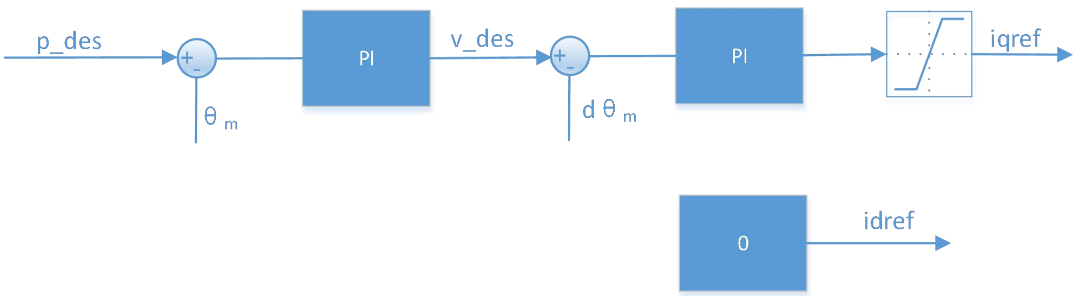
Velocity mode allows the motor to run steadily at the set speed. The control block diagram is as follows:



Note:
The unit of v_des is rad/s, and the data type is float.
To enable automatic parameter calculation using the debugging assistant, the damping factor must be set to a non-zero positive value.
Typically, it ranges from 2.0 to 10.0: A value that is too small may cause speed oscillation and significant overshoot. A value that is too large may result in a longer rise time. The recommended setting is 4.0.

❖ Force-Position Hybrid Control Mode

The force-position hybrid control mode dynamically regulates the output torque based on position and speed control. The control block diagram is shown below.



A current command saturation stage is added after the speed loop output, which limits the current loop command within a specified range.

Usage

Control is implemented using the standard CAN frame format with a fixed baud rate of 1 Mbps. Based on function, frames are categorized into command frames and feedback frames:
Command frames: These carry control data received by the motor to execute specific control commands.
Feedback frames: These are sent from the motor to the upper-level controller and contain the motor ’ s status data.
The format and frame ID of the command frames vary depending on the motor ’ s selected mode, whereas the feedback frame format remains consistent across all modes.

❖ Feedback Frame

The Feedback Frame ID is configured via the debugging assistant (Master ID) and defaults to 0. It mainly reports the motor's position, speed, and torque.
The specific frame format is defined as:

Feedback Message	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
MST_ID	ID|ERR<<4	POS[15:8]	POS[7:0]	VEL[11:4]	VEL[3:0]|T[11:8]	T[7:0]	T_MOS	T_Rotor

ID: Controller ID, corresponding to the lower 8 bits of the CAN ID.
ERR: Status indicator, with types as follows:
0 – Disabled
1 – Enabled
8 – Overvoltage
9 – Undervoltage
A – Overcurrent
B – MOS overtemperature
C – Motor coil overtemperature
D – Communication lost
E – Overload
POS: Motor position information
*VEL: Motor speed information
*T: Motor torque information
*T_MOS: Average temperature of the MOSFETs on the driver board (unit: °C)
T_Rotor: Average temperature of the motor ’ s internal coil (unit: °C)
* Position, speed, and torque are converted from floating-point to signed fixed-point format using a linear mapping:
Position uses 16-bit data
Speed and torque use 12-bit data each

❖ Control Frame in MIT Mode

Control Message	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
ID	p_des [15:8]	p_des [7:0]	v_des [11:4]	v_des[3:0]| Kp[11:8]	Kp [7:0]	Kd [11:4]	Kd[3:0]| t_ff[11:8]	t_ff[7:0]

Frame ID is equal to the configured CAN ID value.
P_des: Position command
V_des: Velocity command
Kp: Position proportional gain
Kd: Position derivative gain
T_ff: Torque feedforward value

All parameters follow the mapping relationships described in the previous section.
The ranges of p_des, v_des, and t_ff can be configured via the debugging assistant. Kp range: [0, 500]   Kd range: [0, 5]
A standard CAN data frame contains only 8 bytes.
In MIT mode, the control command format packs five parameters — Position, Velocity, Kp, Kd, and Torque — into those 8 bytes.
Position: 16 bits (2 bytes)
Velocity: 12 bits
Kp: 12 bits
Kd: 12 bits

❖ Control Frame in Position-Velocity Mode

Control Message	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0x100+ID	p_des				v_des			

Frame ID = configured CAN ID value + 0x100
P_des: Position command, float, little-endian (low byte first, high byte last)
V_des: Velocity command, float, little-endian (low byte first, high byte last)
In this mode, the CAN ID used to send commands is 0x100 + ID. The velocity command (v_des) defines the maximum speed during the movement to the
target position—i.e., the speed during the constant-velocity phase.

❖ Control Frame in Velocity Mode

Control Message	D[0]	D[1]	D[2]	D[3]
0x200+ID	v_des			

Frame ID = configured CAN ID value + 0x200
V_des: Velocity command, float, little-endian (low byte first, high byte last)In this mode, the CAN ID used to send commands is
0x200 + ID.

❖ Control Frame in Force-Position Hybrid Mode

Control Message Format:

Control Message	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0x300+ID	p_des				v_des		i_des	

P_des: Position command, unit: rad, data type: float, little-endian (low byte first, high byte last)V_des: Speed limit value, unit: rad/s, scaled by 100, data type: unsigned 16-bit integer, little- endian.Range: 0 – 10000. Values above 10000 are clamped to 10000. Corresponds to an actual speed limit range of 0 – 100 rad/s.

I_des: Torque current limit (per unit), scaled by 10000, data type: unsigned 16-bit integer, little-endian.
Range: 0 – 10000. Values above 10000 are clamped to 10000. Corresponds to an actual per-unit current limit of 0 – 1.0.Per-unit current = actual current / maximum current; for DM-J4340P-2EC, max current = 10.26A.

❖ Read Parameters

Message ID	Attributes	D[0]	D[1]	D[2]	D[3]
0x7FF	STD	CANID_L	CANID_H	0x33	RID

RID refers to the register address. See appendix <Register List and Range>.
If the read is successful, the motor returns the data from the specified register with the following frame format:

Message ID	Attributes	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
MST_ID	STD	CANID_L	CANID_H	0x33	RID	Data			

The data is either a floating-point or unsigned integer, occupying 32 bits (4 bytes), with the lowest byte in D4 and highest byte in D7. Same format applies elsewhere unless stated.

❖ Write Parameters

Message ID	Attributes	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0x7FF	STD	CANID_L	CANID_H	0x55	RID	Data			

RID is the same as above. If the write is successful, the response frame returns the written data in the same format as the sent frame.

Message ID	Attributes	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
MST_ID	STD	CANID_L	CANID_H	0x55	RID	Data			

Register write takes effect immediately but is not stored persistently. After power-off, the changes will be lost.To store the modified parameters permanently, you must send a " store parameters" command to write all changes to onboard memory.

❖ Store Parameters

Message ID	Attributes	D[0]	D[1]	D[2]	D[3]
0x7FF	STD	CANID_L	CANID_H	0xAA	0x01

Upon successful storage, the return format is:

Message ID	Attributes	D[0]	D[1]	D[2]	D[3]
MST_ID	STD	CANID_L	CANID_H	0xAA	0x01

Note:
1. Parameters can only be stored in disabled mode.
2. Storing parameters will save all parameters at once.
3. This operation writes the parameters to internal flash. Each operation may take up to 30 ms, so allow sufficient time.
4. Flash write endurance is approximately 10,000 times. Avoid frequent " store parameter " operations.

❖ Mode Switching

Multiple control modes are supported and can be switched between.
The currently supported modes are:

Code	Mode
1	MIT
2	Position-Velocity
3	Velocity
4	Force-Position Hybrid

You can switch modes by modifying the mode register (0x0A).
When switching modes, the motor will first reset all command values (including position, speed, torque feedforward in MIT mode, KP, and KD).
When switching to a position control mode, to avoid impact, it is recommended to first read the precise position (register 0x50), and preferably switch when the motor is at zero speed.

Note: Mode changes are not stored to flash. After power-off, the motor will revert to the last mode saved in flash.

❖ CAN Baud Rate Modification

You can change the CAN communication baud rate by writing a specific value to the baud rate register (address: 0x23).
The following baud rates are currently supported:

Code	Baud Rate
0	125K
1	200K
2	250K
3	500K
4	1M
5	2M
6	2.5M
7	3.2M
8	4M
9	5M

After successfully modifying the baud rate, the driver will first respond using the original baud rate, and then switch to communication at the new baud rate. Upon power-up, the motor will check the stored baud rate: if it is greater than 5 Mbps, it will automatically default to 1 Mbps. For baud rates greater than 1 Mbps (excluding 1 Mbps), the system will automatically switch to CAN FD mode. If the baud rate is less than or equal to 1 Mbps, it will switch to CAN 2.0B mode.Motors set to CAN FD mode can still receive CAN 2.0B data frames, but will use CAN FD to send feedback frames. As a result, the upper-level controller will be unable to receive feedback data, and the driver will continuously report errors. If a controller using CAN 2.0B sets an incorrect ID, the baud rate can still be restored by sending a baud rate modification command.

Using the Debugging Assistant

Use the USB-to-CAN debugging tool to connect the motor to the computer, and configure motor parameters or perform firmware upgrades via the debugging assistant.
The motor's serial port connects to the PC through a GH1.25 3-pin cable. The CAN communication terminal on the motor's power interface is connected
to the USB-to-CAN debugging tool using an XT30 (2+2)-F plug cable. Parameter configuration and firmware upgrades can be carried out through the debugging assistant.
After connecting the motor's serial port, CAN port, and power interface, open the debugging assistant on the computer, select the appropriate serial port device, and open the port. Once the motor is powered on, the serial port will output information, with " Control Mode " indicating the current drive mode.

Key Parameters

Please refer to the following parameters for proper motor usage:

Motor Parameters	Rated Voltage	24V/48V
	Rated Phase (Power) Current	2.5 A
	Peak Phase (Power) Current	8A
	Rated Torque	9 NM
	Peak Torque	27 NM
	Rated Speed	36 rpm
	Max No-Load Speed	52 rpm
Motor Characteristics	Gear Ratio	40 : 1
	Pole Pairs	14
	Phase Inductance	360 uH
	Phase Resistance	880 mΩ
Structure & Weight	Outer Diameter	57 mm
	Height	56.5 mm
	Motor Weight	≈ 375 g
Encoder	Encoder Resolution	14-bit
	Number of Encoders	2
	Encoder Type	Magnetic encoder (single-turn)
Communication	Control Interface	CAN@1Mbps
	Tuning Interface	UART @ 921 600 bps

Control & Protection	Control Modes	MIT Mode/Speed Mode/Position Mode /Force-Position Hybrid Control Mode
	Driver Over-temperature Protection (threshold)	120 °C — exits "enabled mode" when exceeded
	Motor Over-temperature Protection (user-set, recommended ≤ 100 °C)	Exits "enabled mode" when exceeded
	Motor Over-voltage Protection (user-set, recommended ≤ 32(24V Version)/52V(48V Version)	Exits "enabled mode" when exceeded
	Communication-loss Protection	If no CAN command within set period → exits "enabled mode"
	Motor Over-current Protection (user-set, recommended ≤ 9.8 A)	Exits "enabled mode" when exceeded
	Motor Under-voltage Protection (supply must be ≥ 15 V)	Exits "enabled mode" when below threshold

Appendix: Register List and Parameter Ranges

Address (HEX)	Address (DEC)	Variable	Description	R/W	Range / Value	Type
0x00	0	UV_Value	Undervoltage Protection Value	RW	(10.0, fmax]	float
0x01	1	KT_Value	Torque Constant	RW	[0.0, fmax]	float
0x02	2	OT_Value	Overtemperature Protection	RW	[80.0, 200)	float
0x03	3	OC_Value	Overcurrent Protection Value	RW	(0.0, 1.0)	float
0x04	4	ACC	Acceleration	RW	(0.0, fmax)	float
0x05	5	DEC	Deceleration	RW	[-fmax, 0.0)	float
0x06	6	MAX_SPD	Maximum Speed	RW	(0.0, fmax]	float
0x07	7	MST_ID	Feedback ID	RW	[0, 0x7FF]	uint32
0x0A	8	ESC_ID	Receive ID	RW	[0, 0x7FF]	uint32
0x09	9	TIMEOUT	Timeout Alarm Duration	RW	[0, 2^32−1]	uint32
0x0A	10	CTRL_MODE	Control Mode	RW	[0, 4]	uint32
0x0B	11	Damp	Motor Damping Coefficient	RO	/	float
0x0C	12	Inertia	Motor Inertia	RO	/	float
0x0D	13	hw_ver	Reserved	RO	/	uint32
0x0E	14	sw_ver	Software Version	RO	/	uint32
0x0F	15	SN	Reserved	RO	/	uint32
0x10	16	NPP	Motor Pole Pairs	RO	/	uint32
0x11	17	Rs	Motor Phase Resistance	RO	/	float
0x12	18	Ls	Motor Phase Inductance	RO	/	float
0x13	19	Flux	Motor Flux	RO	/	float
0x14	20	Gr	Gear Reduction Ratio	RO	/	float
0x15	21	PMAX	Position Mapping Range	RW	(0.0, fmax]	float
0x16	22	VMAX	Speed Mapping Range	RW	(0.0, fmax]	float
0x17	23	TMAX	Torque Mapping Range	RW	(0.0, fmax]	float
0x18	24	I_BW	Current Loop Bandwidth	RW	[100.0, 1.0e4]	float
0x19	25	KP_ASR	Speed Loop Kp	RW	[0.0, fmax]	float
0x1A	26	KI_ASR	Speed Loop Ki	RW	[0.0, fmax]	float
0x1B	27	KP_APR	Position Loop Kp	RW	[0.0, fmax]	float
0x1C	28	KI_APR	Position Loop Ki	RW	[0.0, fmax]	float
0x1D	29	OV_Value	Overvoltage Protection Value	RW	TBD	float
0x1E	30	GREF	Gear Torque Efficiency	RW	(0.0, 1.0]	float
0x1F	31	Deta	Speed Loop Damping Coefficient	RW	[1.0, 30.0]	float
0x20	32	V_BW	Speed Loop Filter Bandwidth	RW	(0.0, 500.0)	float
0x21	33	IQ_c1	Current Loop Boost Coefficient	RW	[100.0, 1.0e4]	float
0x22	34	VL_c1	Speed Loop Boost Coefficient	RW	(0.0, 1.0e4]	float
0x23	35	can_br	CAN Baudrate Code	RW	[0, 4]	uint32
0x24	36	sub_ver	Sub-Version	RO	/	uint32
0x32	50	u_off	U Phase Offset	RO	/	float
0x33	51	v_off	V Phase Offset	RO	/	float

0x34	52	k1	Compensation Factor 1	RO	/	float
0x35	53	k2	Compensation Factor 2	RO	/	float
0x36	54	m_off	Angle Offset	RO	/	float
0x37	55	dir	Direction	RO	/	float
0x50	80	p_m	Motor Current Position	RO	/	float
0x51	81	xout	Output Shaft Position	RO	/	float

RW (Read/Write)

RO (Read Only)