

The motor communication uses a CAN 2.0 communication interface with a baud rate of 1 Mbps and adopts the extended frame format, as shown below:

| Data Field | 29-bit ID | | | 8Byte Data Area |
|---------------|-----------|-------------|----------------|-----------------|
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | Data Type | Data Type 2 | Target Address | Data Area 1 |

Obtain Device ID (Communication Type 0x00)

| Data Field | 29-bit ID | | | 8Byte Data Area |
|-----------------|-----------|---|---------------------|-----------------|
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x00 | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | 0 |
| Response Frame: | | | | |
| Data Field | 29-bit ID | | | 8Byte Data Area |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x00 | Target Motor CAN ID | 0xFE | 0 |

Motor Feedback Data (Communication Type 0x02) Used to report motor operating status to the host

| Data Field | 29-bit ID | | | 8Byte Data Area |
|-------------|-----------|---|-------------|--|
| | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x02 | bit8~bit15: Current Motor CAN ID bit16~18: Fault Information (0: No, 1: Yes) bit18: Out of Range bit17: Stall bit16: Overvoltage Fault bit19~21: Mode 1: Speed Mode 2: Position Mode 3: Current Mode 4: Encoder Mode bit22~23: Status 0: Standby 1: Running 2: Error | Host CAN ID | Byte0~1: Current Speed [-32768~32767], Unit (rpm) Byte2~3: Current Position [-32768~32767], Unit (°) Byte4~5: Current Current [-32768~32767], Unit (mA) Byte6~7: Current Input Voltage [-32768~32767], Unit (V) |

Motor Enable Operation (Communication Type 0x03)

| Data Field | 29-bit ID | | | 8Byte Data Area |
|---|-----------|---|---------------------|-----------------|
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x03 | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | 0 |
| Response Frame: Motor Feedback Response Frame(See Communication Type 2) | | | | |

Motor Stop Operation (Communication Type 0x04)

| Data Field | 29-bit ID | | | 8Byte Data Area |
|---------------|-----------|---|---------------------|-----------------|
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x04 | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | 0 |

Response Frame: Motor Feedback Response Frame (See Communication Type 2)

| Set Motor CAN ID (Communication Type 0x07) Change the current motor CAN ID, effective immediately. | | | | |
|--|-----------|--|---------------------|-----------------|
| Data Field | 29-bit ID | | | 8Byte Data Area |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x07 | bit15~8: Used to Identify the Host CAN ID Bit16~23: Preset CAN ID | Target Motor CAN ID | 0 |

Response Frame: Motor Broadcast Response Frame (See Communication Type 0)

| Disable Motor Stall Protection (Communication Type 0x09) | | | | |
|--|-----------|---|---------------------|-----------------|
| Data Field | 29-bit ID | | | 8Byte Data Area |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x09 | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | 0 |

Response Frame: Motor Feedback Response Frame (See Communication Type 2)

| Save Parameters to Flash (Communication Type 0x0A) | | | | |
|--|-----------|---|---------------------|-----------------|
| Data Field | 29-bit ID | | | 8Byte Data Area |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x0A | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | 0 |

Response Frame: Motor Feedback Response Frame (See Communication Type 2)

| Set Motor CAN BPS (Communication Type 0x0B) Immediate Effect | | | | |
|--|-----------|---|---------------------|-----------------|
| Data Field | 29-bit ID | | | 8Byte Data Area |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x0B | bit15~8: Used to Identify the Host CAN ID Bit16~23: Preset CAN BPS (0:1Mbps, 1:500Kbps, 2:125Kbps) | Target Motor CAN ID | 0 |

| Response Frame: | | | | |
|-----------------|-----------|---|--------------|-----------------|
| Data Field | 29-bit ID | | | 8Byte Data Area |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x0B | bit15~8: Used to Identify the Host CAN ID Bit16~23: Preset CAN BPS (0:1Mbps, 1:500Kbps, 2:125Kbps) | Motor CAN ID | 0 |

| Enable Motor Stall Protection (Communication Type 0x0C) | | | | |
|---|-----------|---|---------------------|-----------------|
| Data Field | 29-bit ID | | | 8Byte Data Area |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x0C | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | 0 |

Response Frame: Motor Feedback Response Frame (See Communication Type 2)

| Disable Motor Stall Protection (Communication Type 0x0D) | | | | |
|--|-----------|--|--|-----------------|
| Data Field | 29-bit ID | | | 8Byte Data Area |

| | | | | |
|---------------|----------|---|---------------------|-------------|
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x0D | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | 0 |

Response Frame: Motor Feedback Response Frame (See Communication Type 2)

Read Single Parameter (Communication Type 0x11)

| | | | | |
|---------------|-----------|---|---------------------|--|
| Data Field | 29-bit ID | | 8Byte Data Area | |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x11 | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | Byte0~1: index Byte2~3: 00 Byte4~7: 00 |

Response Frame:

| | | | | |
|---------------|-----------|---|-----------------|--|
| Data Field | 29-bit ID | | 8Byte Data Area | |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x11 | bit15~8: Used to Identify the Host CAN ID | Motor CAN ID | Byte0~1: index , index, Parameter list, see Table 1 Byte2~3: 00 Byte4~7: Parameter data, 1 Byte, Data is in Byte 4 |

Write Single Parameter (Communication Type 0x12) (Lost on Power-Off)

| | | | | |
|---------------|-----------|---|---------------------|---|
| Data Field | 29-bit ID | | 8Byte Data Area | |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x12 | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | Byte0~1: index, see Table 1 Byte2~3: 00 Byte4~7: Parameter data |

Response Frame: Motor Feedback Response Frame (See Communication Type 2)

I2C Slave Parameter Read (Communication Type 0x13)

| | | | | |
|---------------|-----------|---|---------------------|---|
| Data Field | 29-bit ID | | 8Byte Data Area | |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x13 | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | Byte0~1: index see Table 1 Byte2~3: i2c address Byte4~7: 00 |

Response Frame:

| | | | | |
|---------------|-----------|---|-----------------|---|
| Data Field | 29-bit ID | | 8Byte Data Area | |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x13 | bit15~8: Used to Identify the Host CAN ID | Motor CAN ID | Byte0~1: index , index, Parameter list, see Table 1 Byte2~3: i2c address Byte4~7: Parameter data, 1 Byte, Data is in Byte 4 |

I2C Slave Parameter Write (Communication Type 0x14) (Lost on Power-Off)

| | | | | |
|---------------|-----------|---|---------------------|---|
| Data Field | 29-bit ID | | 8Byte Data Area | |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x14 | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | Byte0~1: index see Table 1 Byte2~3: i2c address Byte4~7: Parameter data |

Response Frame: Motor Feedback Response Frame (See Communication Type 2)

I2C Read Raw (Communication Type 0x15)

| | | | | |
|---------------|-----------|---|---------------------|--|
| Data Field | 29-bit ID | | 8Byte Data Area | |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x15 | bit15~8: Used to Identify the Host CAN ID | Target Motor CAN ID | Byte0: I2C Address Byte1: Number of Bytes Read, Maximum 8 |

| Response Frame: | | | | |
|-----------------|-----------|--|--------------|-----------------|
| Data Field | 29-bit ID | | | 8Byte Data Area |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x15 | bit15~8: Used to Identify the Host CAN ID bit16~23: is read success(0:failed, 1:success) | Motor CAN ID | Data Read |

| I2C Write Raw (Communication Type 0x16) | | | | |
|---|-----------|---|---------------------|--|
| Data Field | 29-bit ID | | | 8Byte Data Area |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x16 | bit15~8: Used to Identify the Host CAN ID Bit16~22: I2C Address Bit23: is stop bit | Target Motor CAN ID | Byte0: Number of Bytes Written, Max 7 Byte1-Byte7: Data Written |
| Response Frame: | | | | |
| Data Field | 29-bit ID | | | 8Byte Data Area |
| Byte Position | bit28~24 | bit23~8 | bit7~0 | Byte0~Byte7 |
| Description | 0x16 | bit15~8: Used to Identify the Host CAN ID bit16~23: is write success(0:failed, 1:success) | Motor CAN ID | 0 |

表1 可读写单个参数列表

| 参数 index | 参数名称 | 描述 | 类型 | 字节数 | 单位/说明 | R/W 读写权限 |
|----------|----------------------------|---|----------|-----|--|----------|
| 0x7002 | save flash | save flash | uint8 | 1 | 写入大于1的数, 保存参数到 flash | W |
| 0x7003 | release protection | release protection | uint8 | 1 | 写入大于1的数, 解除电机堵转保护 | W |
| 0x7004 | on/off | on/off | uint8 | 1 | 1:电机使能 0:电机关闭 | W/R |
| 0x7005 | run_mode | 1: Speed Mode 2: Position Mode 3: Current Mode 4: Encoder Mode | uint8 | 1 | 设置电机模式 | W/R |
| 0x7006 | current (mA) | 电流 X100 Int | int32_t | 4 | -120000~120000 | W/R |
| 0x700A | speed(rpm) | 速度 X100 Int | int32_t | 4 | -2100000000~2100000000 | W/R |
| 0x7016 | position(°) | 位置 X100 Int | int32_t | 4 | -2100000000~2100000000 | W/R |
| 0x7017 | position_max_current | 位置模式最大电流 X100 Int | int32_t | 4 | -120000~120000 | W/R |
| 0x7018 | speed_max_current | 速度模式最大电流 X100 Int | int32_t | 4 | -120000~120000 | W/R |
| 0x7020 | speed_kp | 速度模式的Kp X100000 Int | uint32_t | 4 | For example: P=0.00001, P setting value=0.00001*100000=1, P-byte0=1, P-byte1=0, P-byte2=0, P-byte3=0 | W/R |
| 0x7021 | speed_ki | 速度模式的Ki X1000000 Int | uint32_t | 4 | For example: I=0.00001, I setting value=0.00001*1000000=100, I-byte0=100, I-byte1=0, I-byte2=0, I-byte3=0 | W/R |
| 0x7022 | speed_kd | 速度模式的Kd X100000 Int | uint32_t | 4 | For example: D=0.00001, D setting value=0.00001*100000=1, D-byte0=1, D-byte1=0, D-byte2=0, D-byte3=0 | W/R |
| 0x7023 | position_kp | 位置模式的Kp X100000 Int | uint32_t | 4 | For example: P=0.00001, P setting value=0.00001*100000=1, P-byte0=1, P-byte1=0, P-byte2=0, P-byte3=0 | W/R |
| 0x7024 | position_ki | 位置模式的Ki X1000000 Int | uint32_t | 4 | For example: I=0.00001, I setting value=0.00001*1000000=100, I-byte0=100, I-byte1=0, I-byte2=0, I-byte3=0 | W/R |
| 0x7025 | position_kd | 位置模式的Kd X100000 Int | uint32_t | 4 | For example: D=0.00001, D setting value=0.00001*100000=1, D-byte0=1, D-byte1=0, D-byte2=0, D-byte3=0 | W/R |
| 0x7030 | Speed Readback X100 Int | 实时速度 | int32_t | 4 | Speed Readback = Speed Readback-byte0 + Speed Readback-byte1 * 256 + Speed Readback-byte2 * 65536 + Speed Readback-byte3 * 16777216 Actual Speed Readback = Speed Readback/100 | R |
| 0x7031 | Position Readback X100 Int | 实时位置 | int32_t | 4 | Position Readback = Position Readback-byte0 + Position Readback-byte1 * 256 + Position Readback-byte2 * 65536 + Position Readback-byte3 * 16777216 Actual Position Readback = Position Readback/100 | R |
| 0x7032 | Current Readback X100 Int | 实时电流 | int32_t | 4 | Current Readback = Current Readback-byte0 + Current Readback-byte1 * 256 + Current Readback-byte2 * 65536 + Current Readback-byte3 * 16777216 Actual Current Readback = Current Readback/100 | R |
| 0x7033 | Encoder Counter | 编码器值 | int32_t | 4 | Encoder Counter = Encoder Counter-byte0 + Encoder Counter-byte1 * 256 + Encoder Counter-byte2 * 65536 + Encoder Counter-byte3 * 16777216 Can only use in Encoder mode | W/R |
| 0x7034 | VIN X100 Int | 输入电压值 | int32_t | 4 | VIN X100 = VIN X100-byte0 + VIN X100-byte1 * 256 + VIN X100-byte2 * 65536 + VIN X100-byte3 * 16777216 Actual VIN = VIN X100/100 | R |
| 0x7035 | Temp Int | SOC温度 | int32_t | 4 | Temp = Temp-byte0 + Temp-byte1 * 256 + Temp-byte2 * 65536 + Temp-byte3 * 16777216 | R |
| 0x7050 | rgb mode | RGB Mode | uint8 | 1 | RGB Mode: 0, Sys-default 1, User-define | W/R |
| 0x7051 | rgb color | rgb color | uint32_t | 4 | rgb color = RGB-B + RGB-G * 256 + RGB-R * 65536 | W/R |
| 0x7052 | rgb brightness | rgb brightness | uint8_t | 1 | RGB Brightness: 0-100 | W/R |