**Copyright statement**

**Beijing Hollys Motor Technology Co., Ltd. All rights reserved.**

**The contents of the manual refer to relevant legal and industry benchmarks. When using the product, if you have any questions about the content provided in this manual, please contact the sales person who purchased the product, call the customer service hotline, or write to our mailbox.**

**Due to continuous product upgrades, Hollysys reserves the right to modify the product and product specifications in this manual without prior notice, prompting customers to use the latest version of the manual.**

**Hollysys Motors has the patent, copyright and other intellectual property rights of this product and its software. Without permission, this product and its related parts may not be copied, manufactured, processed, or used directly or indirectly.**

**Hollysys Motors owns the copyright of this manual, and may not modify or copy all or part of the manual without permission.**

I. Communication port settings:

UDP communication needs to open the port, the local port is 8401, the target port is 7408, and the local port 8410 is sent to the target port 7408. (8410 and 7408 are MBOX's factory default ports, which can be modified, see the MBOX manual for details)

# 二、动作指令 MBOXUDP 数据区举例

00 00 00 00 00 00 00 00 00 00

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 55 aa | | | |  | 00 00 |  | 13 01 |  | 00 01 |  | ff ff ff ff |  | 00 00 00 | |
|  | 00 00 00 00 |  | 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | | | | | | | | | | |  |
|  | | | | | | | |  | 12 34 |  | 56 78 ab cd |  | | |

01

**Annotation [z1]: Verification code, 2 bytes Annotation [z2]: Pass code, 2 bytes**

**Comment [z3]: Function code, 2 bytes. Divided into absolute time and relative time**

**Annotation [z4]: 2 bytes for 6 axes**

**Annotation [z5]: Make sure that the UDP data is received by that platform, and have a reply from that platform (all FF means full reception, full reply). 2 words each**

**Section, 4 bytes total**

**Annotation [z6]: the serial number of the UDP command sent,**

**2 bytes**

**Annotation [z7]: time code, 4 bytes in milliseconds, format is hexadecimal**

**Annotation [z8]: data for six axes, 4 bytes per axis, unit is pulse number, format is hexadecimal**

**Annotation [z9]: 2 bytes, 12 switches**

**Annotation [z10]: 4 bytes, two analog channels**

三、Analysis of UDP data area of ​​action instructions

1. Verification code

Fixed value, 0x55aa, hexadecimal, 2 bytes.

Pass code

Fixed value, 0x0000, hex, 2 bytes.

3. Function code

Divided into two categories: 0x1301, 0x1401, hex, 2 bytes.

The absolute time function code is 0x1301, and the relative time function code is 0x1401.

The absolute time 0x1301 matches the movie. When the absolute time is played, the execution time of the instruction is the absolute time in the current instruction minus the absolute time in the previous instruction. For example, the absolute time of your last instruction is 15000ms. It is 15100ms, then the execution time of this instruction is 100ms (15100ms-1500ms).

Relative time playback function code 0x1401 is the Delta Time in the current instruction, which is the execution time of the current instruction.

MBOX calculates the speed of the motor based on this execution time.

If the function code 1301 is played with absolute time and the time code is 0, then the playback time of each frame of data will be played according to the internal time (default 100ms); if the relative time function code is used and the time code is 0, then the The playback time will also be played in internal time (default 100ms).

But it is not recommended to set the time code to 0, please set it according to the actual execution time.

4.Channel number

Hex, 2 bytes.

Make sure that the current command is 3 axis, 6 axis, 10 axis control command, 3 axis command is 0x0000, 6 axis command is 0x0001, and 10 axis command is 0x0002.

5. Who accepts, who responds

Hex, 2 bytes each, 4 bytes in total.

Each MBOX has an IP address, the factory default is 192.168.15.201 (can be modified, see the MBOX for the modification method

Manual), and support response to each control instruction.

For example, if a command is sent and the IP address is 192.168.15.201 to receive and reply, whoever receives it is 0x0fc9 (0x0f = 15, 0xc9 = 201) and who responds as 0x0fc9 (0x0f = 15, 0xc9 = 201). If you control multiple platforms at the same time, hope that each platform can receive, and each platform responds, then whoever receives it is 0xffff, who responds as 0xffff. If you set the reply to 0x0000, the platform will not reply to the received instructions; if you set the reply to a fixed value, the response will be the corresponding one (if the reply is 0x0510, the corresponding IP address will be followed by 192.168.5.16 For command reply). Receive reply details for rows and columns

**See MBOX manual.**

**6.Serial number of UDP instruction**

**Hex, 4 bytes in total.**

**For verification only, it can be set to 0x00000000.**

**7, time code**

**Hex, 4 bytes in total. The unit is milliseconds.**

**If the third function code is selected as 0x1301, the time code is absolute time. For example, if you want this instruction to be executed within 100 milliseconds, you need to calculate the time difference between this instruction and the previous instruction.**

**If the third function code is selected as 0x1401, the time code is relative time. For example, if the instruction is to be executed within 100 milliseconds, the time code in this instruction must be set to 0x00000064.**

**The absolute time 0x1301 matches the movie. When the absolute time is played, the execution time of the instruction is the absolute time in the current instruction minus the absolute time in the previous instruction. For example, the absolute time of your last instruction is 15000ms. It is 15100ms, then the execution time of this instruction is 100ms (15100ms-1500ms).**

**Relative time playback function code 0x1401 is the Delta Time in the current instruction, which is the execution time of the current instruction.**

**MBOX calculates the speed of the motor based on this execution time.**

**If the function code 1301 is played with absolute time and the time code is 0, then the playback time of each frame of data will be played according to the internal time (default 100ms); if the relative time function code is used and the time code is 0, then the The playback time will also be played in internal time (default 100ms).**

**But it is not recommended to set the time code to 0, please set it according to the actual execution time.**

**8.Location information**

**Hex, 4 bytes for each position. The unit is the number of pulses.**

**How much position information is determined by the 4th channel number. If the channel number is the 3 axis command 0x0000, then the 3 position information is required to be X, Y, Z (corresponding to the 1-3 axis), a total of 12 bytes; if The channel number is 6-axis command 0x0001, then the 6 position information is required to be X, Y, Z, U, V, W (corresponding to 1-6 axes), a total of 24 bytes; if the channel number is 10-axis command 0x0002, Then 10 pieces of position information are needed: X, Y, Z, U, V, W, A, B, C, D (corresponding to 1-10 axes), a total of 40 bytes.**

**Pulse calculation method: the factory default is that the number of pulses per revolution of the motor is 10,000**

**(1) Example of electric cylinder:**

**Suppose the electric cylinder has a lead of 5 mm and the number of pulses per revolution is 10,000. The motor rotates once and the electric cylinder rises by one lead. The electric cylinders are all from 0 to 25 mm. Calculate the number of pulses required: 25/5 \* 10000 = 50000, converted into hexadecimal is 0x0000c350, (Note: You can set the length of each cylinder extension as required.)**

**(2) Example of rotation axis:**

**Generally, the rotating shaft has deceleration. If the reduction ratio is 1:60, the motor rotates 60 times, and the turntable rotates once. If the turntable needs to be rotated by 30 degrees, calculate the required pulse number 30/360 \* (60 \* 10000) = 50,000, and convert it to hexadecimal to be 0x0000c350.**

**9, special effects output**

**Hex, 2 bytes in total.**

**This is a 16-bit data. The upper 4 bits are reserved. The lower 12 bits represent 12 digital outputs. Each bit corresponds to an output. 1-bit special effect is on, and 0 is special effect off.**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| B15 | B14 | B13 | B12 | B11 | B10 | B9 | B8 | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
| save | | | | 12 digital output setpoints | | | | | | | | | | | |

Examples are as follows:

If you want to control 1, 3, 4 output, other channels are closed, this corresponds to the following

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| B15 | B14 | B13 | B12 | B11 | B10 | B9 | B8 | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |

# The hexadecimal is 0x000D. 10.Analog

# Output analog quantity, two outputs, 2 bytes each, 4 bytes in total. For details, please refer to the MBOX instruction manual for examples of action instructions:

# 1、Three-axis data and absolute time mode (function code is 1301)

55 aa 00 00 13 01 00 00 ff ff ff ff 00 00 00

01 00 00 00 00 00 00 c3 50 00 00 c3 50 00 00 c3 50 12 34

56 78 ab cd

# 2、Three-axis data and relative time mode (function code is 1401)

55 aa 00 00 14 01 00 00 ff ff ff ff 00 00 00

01 00 00 00 64 00 00 c3 50 00 00 c3 50 00 00 c3 50 12 34

56 78 ab cd

# 3、Six-axis data and absolute time mode (function code is 1301)

55 aa 00 00 13 01 00 01 ff ff ff ff 00 00 00

01 00 00 00 00 00 00 c3 50 00 00 c3 50 00 00 c3 50 00 00

c3 50 00 00 c3 50 00 00 c3 50 12 34 56 78 ab cd

# 4、Six-axis data and relative time mode (function code is 1401)

55 aa 00 00 14 01 00 01 ff ff ff ff 00 00 00

01 00 00 00 64 00 00 c3 50 00 00 c3 50 00 00 c3 50 00 00

c3 50 00 00 c3 50 00 00 c3 50 12 34 56 78 ab cd

# 5、Ten-axis data and absolute time mode (function code is 1301)

55 aa

00 01

55 aa 00 00 13 01 00 02 ff ff ff ff 00 00 00

Annotation [z11]: Confirmation code, same as action instruction

01 00 00 c3 50 00 00 c3 50 00 00 c3 50 00 00 c3

|  |  |  |  |
| --- | --- | --- | --- |
| 00 | 00 | 00 | 00 |
| 00 | 00 | c3 | 50 |
| 00 | 00 | c3 | 50 |

50 00 00 c3 50 00 00 c3 50 00 00 c3 50 00 00 c3

50 12 34 56 78 ab cd

# 6、Ten-axis data and relative time mode (function code is 1401)

55 aa 00 00 14 01 00 02 ff ff ff ff 00 00 00

Annotation [z12]: pass code, same as action instruction **Annotation [z13]: Function code, 2 bytes**

**Annotation [z14]: Object channel is two bytes**

01

|  |  |  |  |
| --- | --- | --- | --- |
| 00 | 00 | 00 | 64 |
| 00 | 00 | c3 | 50 |
| 00 | 00 | c3 | 50 |

50

50

# Fourth, read and write parameter instructions

01

1.Confirmation code

Same action instruction00 00 c3 50 00 00 c3 50 00 00 c3 50 00 00 c3

00 00 c3 50 00 00 c3 50 00 00 c3 50 00 00 c3

12 34 56 78 ab cd

**Annotation [z15]: who receives, who responds, same action instructions**

**Annotation [z16]: the start address of the parameter register,**

**2 bytes**

Annotation [z17]: Number of registers, 2 bytes

**Comments [z18]: Read register parameters, no such data; write register parameters, write registered values, each register value is 2 bytes, the total is related to the number of registers**

Pass code

Same action instruction

3. Function code

Hex, 2 bytes.

Divided into read register function code 0x1101, write register 0x1201

4.Channel

Hex, 2 bytes. Determined by the third function code.

If the function code reads register 0x1101, the channel is 0x0000 to read the DN parameter, and the channel is 0x0001 to read the FN parameter. If the function code is written to register 0x1201, then the channel is 0x0000. The FN parameter is written but not saved. The channel is 0x0001.

FN parameters are saved and the channel is 0x0002. Write CX control parameters.

5, who receives, who responds with the same action instruction

6.Parameter start address

Hex, 2 bytes.

Supports single read and continuous multiple reads, so it is the starting address of the parameter to be read.

7, read the number of registers

Hex, 2 bytes.

How many bytes are read consecutively, including the start address.

Parameter value

If it is a read parameter command, the command has no parameter value part. If it is a write parameter, it is the value of the register.

Example of read register instruction

Read DN parameters, starting address is 0x0010 (corresponding to DN10), read 0x0010 parameters (corresponding to continuous read)

16 parameters), that is, continuous reading (DN10, DN11 ,,,,,, DN1F total 16 parameters)

|  |
| --- |
| **Comments [z19]: the start address of the parameter register,**  **2 bytes starting at 16** |
| **Annotation [z20]: Read 16 consecutive DN parameters** |

00

00 10

10

55 aa 00 00 11 01 00 00 ff ff ff ff

FN parameter read start address 0x0001 (corresponding FN001), 0x0002 continuously read parameters (corresponding to the continuous reading 2

Parameters), i.e. the continuous reading (FN001, FN002 of 2 parameters)

|  |
| --- |
| **Annotation [z21]: the start address of the parameter register,**  **2 bytes starting at 1** |
| **Annotation [z22]: Read 2 FN parameters consecutively** |

00

00 01

02

55 aa 00 00 11 01 00 01 ff ff ff ff

The DN parameter and FN parameter have the same reply command, only the channels are different. Here take the FN parameter as an example。

|  |
| --- |
| **Comment [z23]: 0x0000 returns the value of DN, 0x0001 returns the value of FN** |
| **Annotation [z24]: the start address of the parameter register,**  **2 bytes starting at 1** |
| **Annotation [z25]: Reply 2 parameters in a row** |
| **Annotation [z26]: Parameter value of reply, each register**  **Device corresponds to 2 Bytes. FN01, FN02** |

55 aa 00 00 11 02

00 01

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 00 | 00 | 00 | 00 | 00 01 |  | 00 |

02

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 00 | 00 | 00 | 00 |  |

Example of write register instruction

Write FN parameter start address is 0x0020 (corresponding to FN020), write 0x0004 parameters continuously (corresponding to write 4 parameters continuously), that is, continuous write (4 parameters of FN020, FN021, FN022, FN023)

12 01

00 01

|  |
| --- |
| **Annotation [z27]: 0x0000 then write FN without saving, 0x0001 write FN and save, 0x0002 write CX parameter** |
| **Annotation [z28]: start address is FN020** |
| **Annotation [z29]: Write 4 parameters in a row** |
| **Annotation [z30]: Each parameter is 2 Bytes in total**  **8 bytes** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ff | ff | ff | ff | 00 20 |  | 00 |

55 aa 00 00

04

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 00 | c0 | 00 | a8 | 00 | 0f | 00 | c9 |  |
|  |  |  |  |  |  |  |  |  |  |

# **Examples of commonly used instructions for writing parameters**

# **1. Platform emergency stop command (write to FN register, modify and not save)**

55 aa 00 00 12 01

00 00

00 01

|  |
| --- |
| **Annotation [z31]: The object channel is 0, which means that the parameter register is modified but not saved** |
| **Comment [z32]: Start address of parameter register**  **0x0090** |
| **Annotation [z33]: The length of the register is 1, which means**  **Access only one register** |
| **Annotation [z34]: The value of the parameter register**  **Address 0x0090 write 0x0001, make the platform stop suddenly** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ff | ff | ff | ff | 00 90 |  | 00 |

01

# 2、Platform cancel emergency stop command (write to FN register, modification is not saved)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ff  55 aa 00 00 12 01  00 00  Comment [z36]: Start address of parameter register  0x0090  **Annotation [z37]: The length of the register is 1, which means that only one register is accessed**  **Annotation [z35]: The object channel is 0, which means that the parameter register is modified but not saved**  00 00 | ff | ff | ff | 00 90 |  | 00 |

01

**3、Platform reset command (write to CX register)**

55 aa 00 00 12 01

01

00 02

00 00

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ff | ff | ff | ff | 00 00 |  | 00 |

|  |
| --- |
| **Annotation [z38]: The object channel is 2, which means the command register** |
| **Comment [z39]: The starting address of the command register is**  **0, indicating the playback control register** |
| **Annotation [z40]: The length of the register is 1, which means**  **Operate on only 1 register** |
| **Comment [z41]: The command register data is:**  **0x0000 for reset operation** |

# 4、4. Control special effects by writing registers

55 aa 00 00 12 01

00 02

ff ff

|  |
| --- |
| **Annotation [z42]: The object channel is 2, which means the command register** |
| **Comment [z43]: The starting address of the command register is**  **1, indicating the special effect control register** |
| **Annotation [z44]: The length of the register is 1, which means**  **Operate on only 1 register** |
| **Annotation [z45]: The command register data is:**  **0xffff, full effects** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ff | ff | ff | ff | 00 01 |  | 00 |

01

# Dynamic platform test software download address: https://yunpan.cn/crgVXxwLRhfeM Access password ec6d