B3M Series Software Manual
Command Reference

Kondo Kagaku Co., Ltd.

Ver. 1.2.0.3

Disclaimer

This manual is a reference manual of commands for communicating with the B3M Series.

Copyrights and all other legal rights on its contents belong to Kondo Kagaku Co., Ltd. Kondo Kagaku bears no responsibility for any results that may occur from use of these commands.

Please contact Kondo Kagaku if you discover any typographical errors in this document. Please note however that we may be unable to respond to questions regarding the content or programming.

The content of this manual and the various names herein may be changed without advance notice.

Contents

1. B3M Servo specification

1.1 Product Description

B3M is a servo developed for high reliability and long life using a brushless motor and non-contact magnetic encoder. The conventional KRS servo motor was developed with the goal of outputting as much power as possible at one time, but the B3M was developed with the goal of maximizing longevity and stable operation. It has been designed especially for durability, not only the motor and angle sensor, which tend to wear out easily, but the case and gears as well.

The system has also been updated. It is equipped with PID control settings and various limiters for temperature limits and operating voltage limits, etc., and resolution can be specified at a fineness four times that of the conventional model. Servo control is possible with just 6 types of commands.

1.2 Product Features and Overview

[Uses High-Precision 12 bit Magnetic Encoder]

The contactless magnetic encoder has no points of contacts for a long operating life.
Moreover, the minimum resolution at 12 bit resolution is 0.088° (0–4095).

[Trusted RS-485 Standard]

Communications conform with the RS-485 standard. It is a differential signal, so is highly resistant to noise, and 3 Mbps high-speed communications are possible. A maximum of 255 connections are possible with daisy chain connection (theoretical value).

[Simple Structure with Only 6 Command Types]

Servo control is possible with just 6 types of commands.
Memory-mapped IO is used, so the servo motor actually operates just by reading and writing memory.

[Multi-Cast Command]

ID numbers can be set for each servo. Also, ID=255 is used as the Broadcast ID. Commands can be executed simultaneously on all daisy-chain-connected servo motors.

[Positioning Gain, Differential Gain, Integral Gain]

PID control parameters can be adjusted when conducting position control or speed control. Also, using the gain preset function, the three types of gains can be registered in advance, so the gain can be quickly changed with a preset number.

[Sensing Function, Software Limit]

Various system errors are stored in memory, including input voltage, MCU temperature, motor temperature, and motor lock. There is also a limit function for safety. It is possible to reduce the causes of trouble by placing restrictions on motor and MCU temperatures.

1.3 Product Structure

The servo unit comes standard with an aluminum clamp horn. Also included are connection cables for the number of servo units.

[Servo Unit]

[Accessories]

<1 Unit>

XH connection cable (400mm) ×1

Aluminum clamp horn ×1

<5 Units>

XH connection cable (400mm) ×5

Aluminum clamp horn ×5

XH connection cable (400 mm)

Aluminum free horn with bearing for B3M/KRS6000 Series

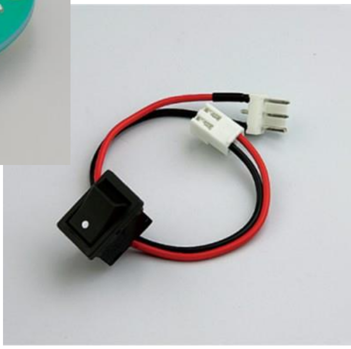
[Required Items]

These are the parts needed for servo control.

RS485USB/Serial conversion adaptor

No.02133

Relay adaptor for communicating with the servo.



HV Power Switch Cable

No.01213

It is a cable with a switch for connecting the USB adapter and the power supply.

Power supply

B3M servo also requires as its power supply at 9-12 V battery or AC power supply, etc.

・SB-1040-A: Ampacity of Max 2.8A required for each servo unit.

・SC-1040-A: Ampacity of Max 3.6A required for each servo unit.

・SC-1170-A: Ampacity of Max 5.4A required for each servo unit.

USB cable

A USB cable is also needed if sending commands to the servo from a PC.
The connector type is USB-A - mini-B.

1.3.1 Option

Hub for XH connector type A

No.02136

Convenient for connecting multiple servos.

Aluminum free horn with bearings (For B3M/KRS-6000 series)

No.02137

Horn connects to bottom case side.

4 cable set for servo

No.02135

XH cable connector set

No.02134

You can combine the cable set and the connector set to create the required length cable.

Servo arm

No.02150

Servo bracket S

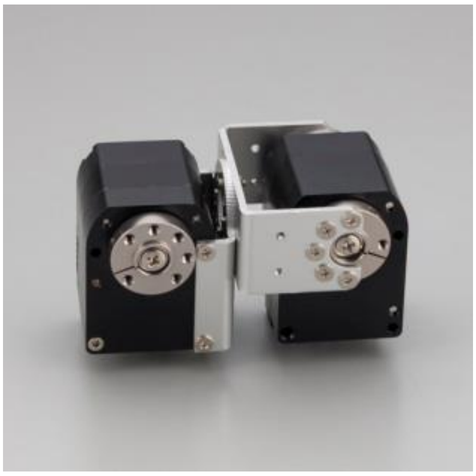
No.02151

Servo bracket B

No.02152

Combining the servo arm and the bracket makes it possible to realize various types of installation methods.

*These frame parts is for B3M-1000 series.
B3M-SB-1040-A
B3M-SC-1040-A



Sample usage

1.4 Servo Types and Specifications

The B3M Series has two main types of servos. B3M-SB uses a brushless motor and B3M-SC uses a coreless motor. The brushless motor has a long operating life because it does not have brushes that wear out. Also, the coreless motor is high spec but is set at a lower price than the SB Series for easy adoption.

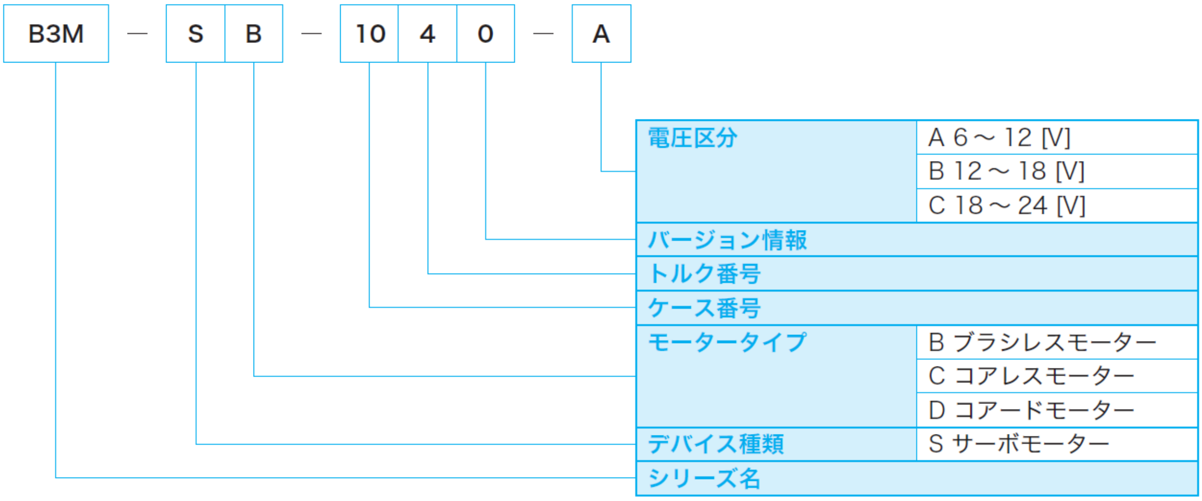
Item	SB-1040-A	SC-1040-A	SC-1170-A
Operating environment temp. [℃]	0–40		
Operating voltage [V]	6–12		
Standby current*1 (in free mode) [mA]	80	68	68
Stall current [A]	2.8	3.6	5.4
Maximum torque*1 [Nm]	4.1	4.6	7.6
[kgf･cm]	42	47	78
Maximum continuous torque*1 [Nm]	2.5	2.5	4.0
No-load rotation speed*1 [s/60°]	0.22	0.18	0.21
[rpm]	45	54	46
Gear Ratio	362.88:1	381.2:1	362.88:1
Operating temp. [°C]	0–40		
Motor type	**Brushless motor**	**Coreless motor**	
Weight [g]	86	82	105
Maximum operating angle*2 position control	±320° (±327.67°)		
Other control	Endlessly rotating*3		
Control resolution	12 [bit]/1 round (0–4095)		
Operation system	Position control, Speed control, Torque control		
Communication system	RS485-communication-compliant, half-duplex communication		
Main parameters	ID number, various control items, various PID control gains		
Unique IDs	0–254 (255 reserved as broadcast ID)		

*1 Specifications value at nominal voltage.

*2 Operating angle from starting position (starting position can be set freely).

*3 If acquiring angle, starting position ±180 deg.

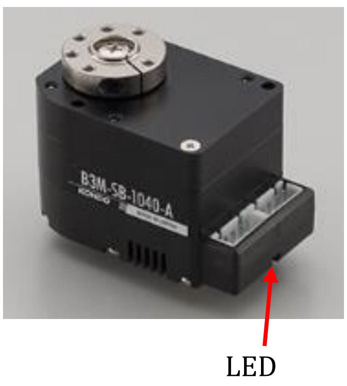
How to View B3M Series



電圧区分	A 6～12 [V]
	B 12～18 [V]
	C 18～24 [V]
バージョン情報	
トルク番号	
ケース番号	
モータータイプ	B ブラシレスモーター
	C コアレスモーター
	D コアードモーター
デバイス種類	S サーボモーター
シリーズ名	

1.5 LED Function

Servo status can be confirmed via the blink pattern of the LED light mounted on the main servo unit.



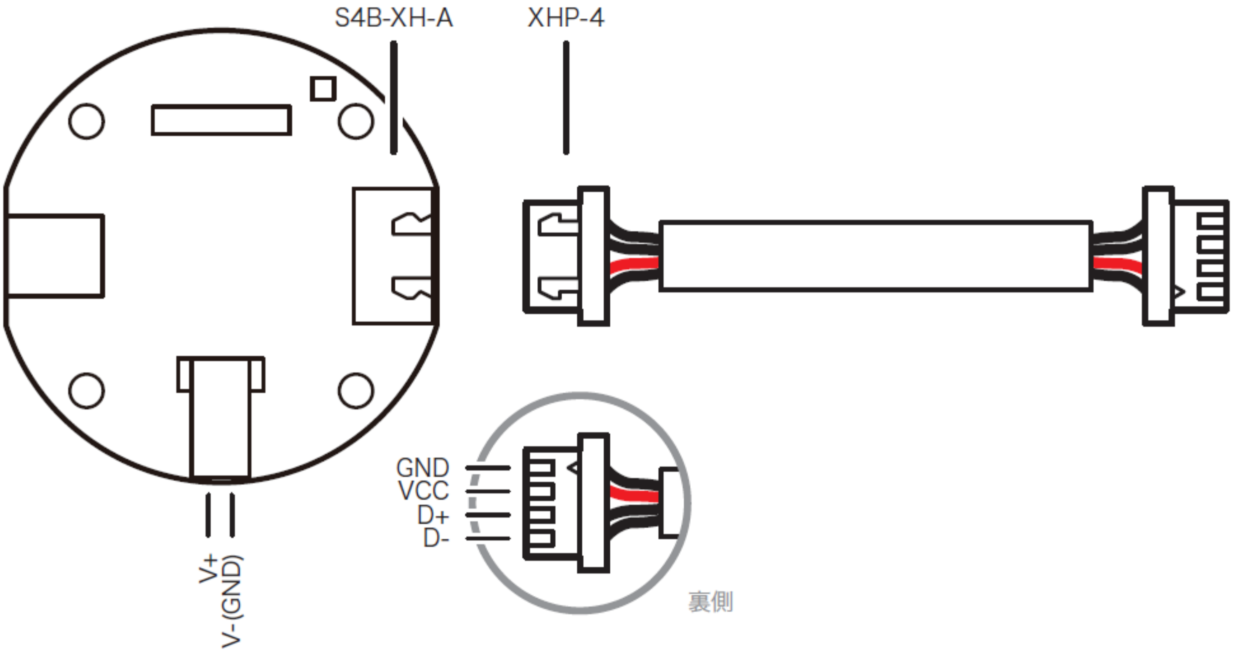
Status	Blink Pattern
Normal	On
System error	Blinks every 100 ms
Motor error	Blinks every 250 ms
UART error	Blinks every 500 ms
Command error	Blinks every second
Power Off	Off

*To clear the error, set 1 in CLEAR Bit of OPTION.

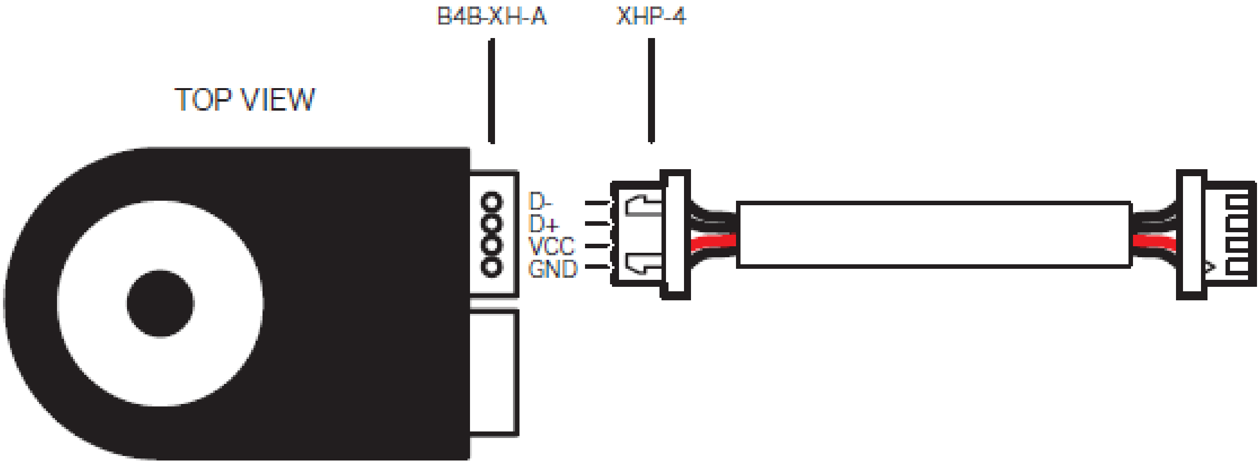
*In Normal mode, LED blinking is low priority, so will not blink regularly.
Confirm the status in Free mode or Hold mode.

1.6 Connector and Pin Placement

[S4B-XH-A Side Connection Diagram]



[Servo Side Connection Diagram]



[Cable Product Name/Code]

Product name: XH Connection Cable (400 mm)

Code: 02144

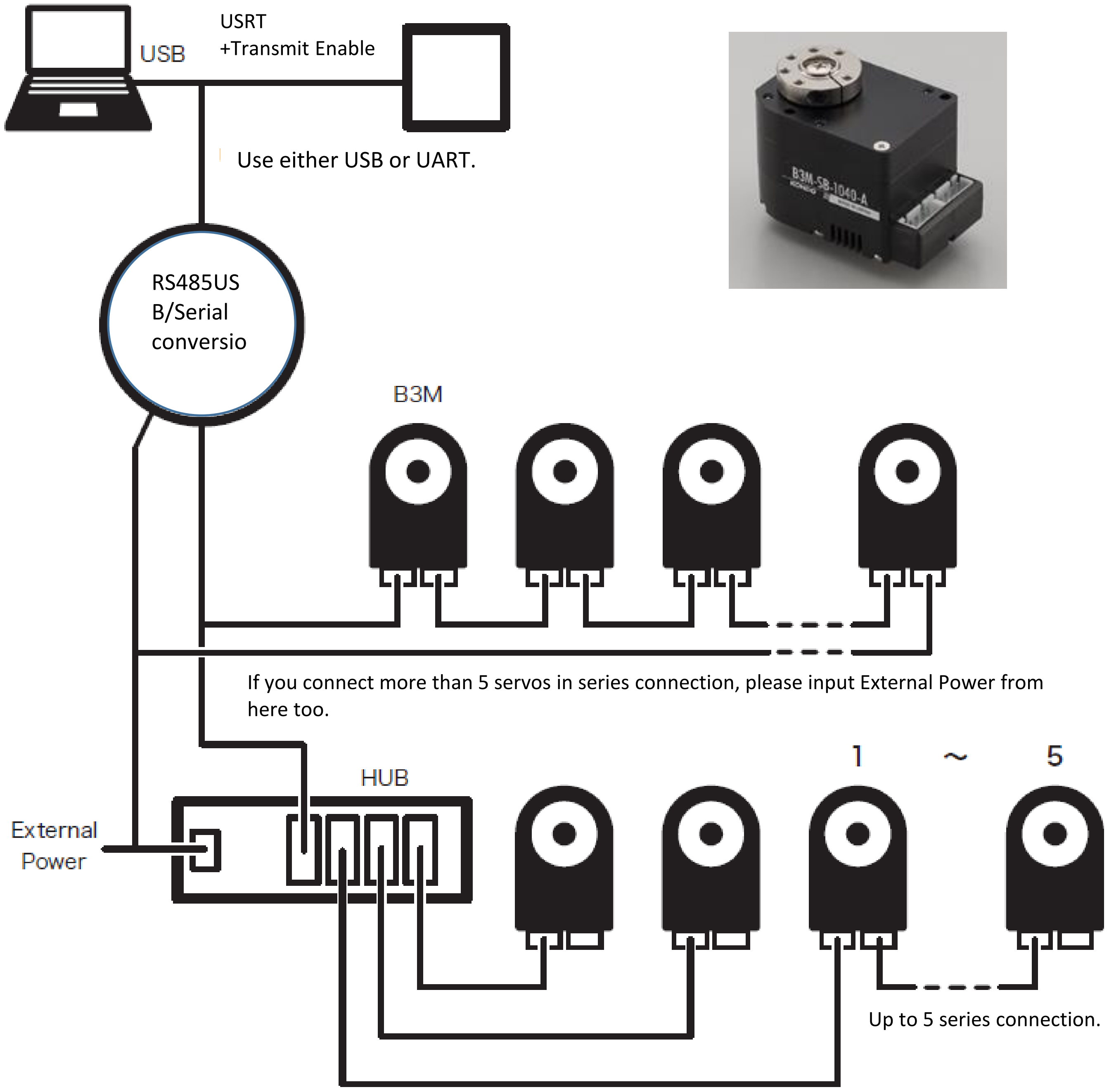
[Connector Set Product Name/Code]

Product Name: XH Cable Connector Set (contains 30 sets)

Code: 02134

1.7 System Structure

B3M can be controlled with the following system. The host that sends commands uses a PC USB connection or microcomputer board UART. Multiple servo units can be connected through multidrop connection and relayed with a RS485USB/serial conversion adaptor. Power can be supplied from a USB adaptor; it can also be supplied across a hub.



Use either USB or UART.

If you connect more than 5 servos in series connection, please input External Power from here too.

Up to 5 series connection.

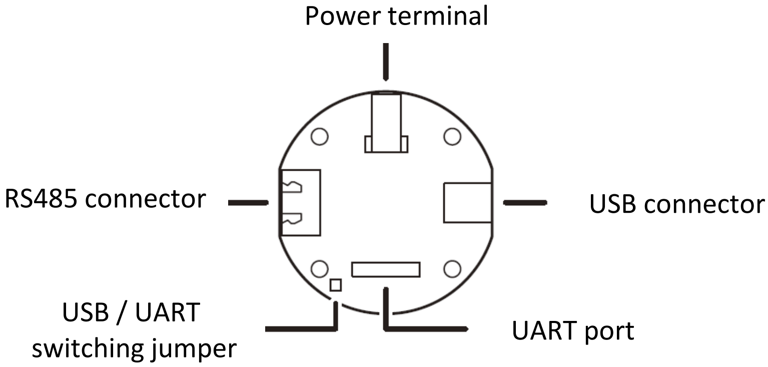
*If many servos are connected in series, the servo at the end becomes insufficient power.

If there are more than six serial connections, please use the hub to distribute the connections.

1.8 Interface

[RS485USB/Serial Conversion Adaptor]

With this adaptor, communications can be aligned to B3M servo's RS-485 standard. Whether connecting directly from a PC or controlled from a microcomputer board's UART terminal, it is compatible with both configurations. It is also equipped with a terminal for power supply to the servo.



[Mechanical Properties]

Item	Content
Usage Temperature	0–40 [℃]

[Electrical/Communication Properties]

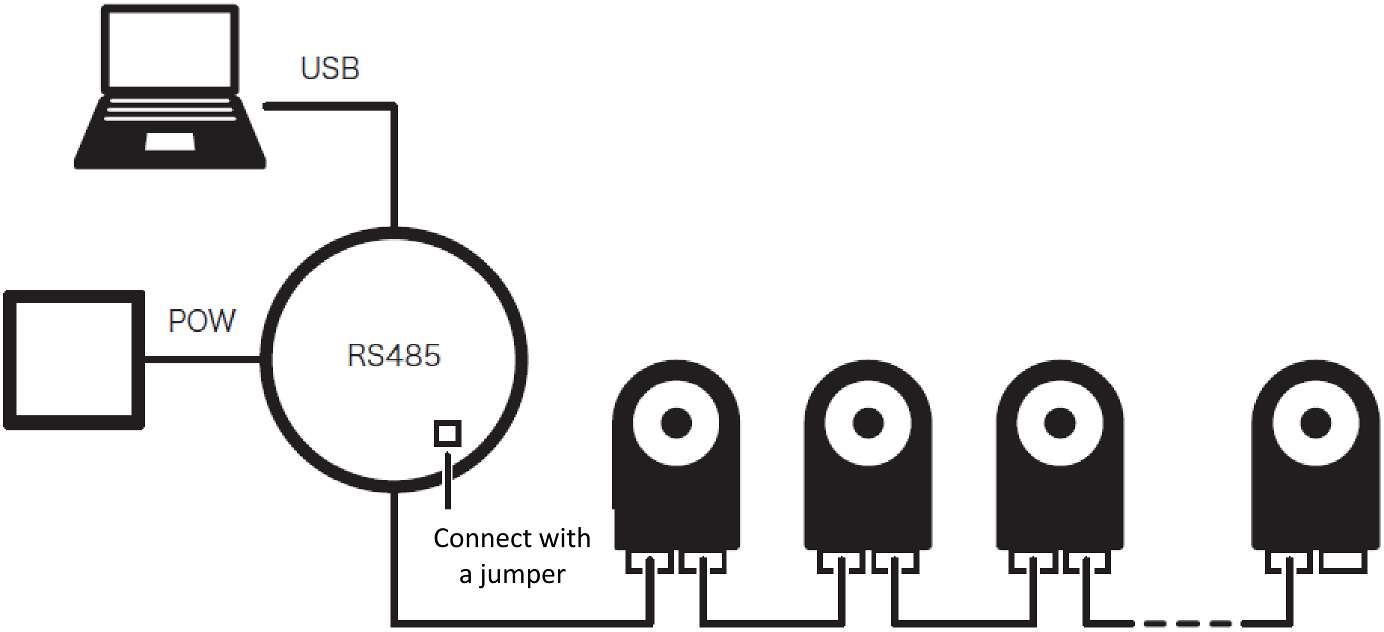
Item	Content
Connection device applied voltage	Depend on connection device
USB	USB2.0 and USB1.1
RS485	
Transfer mode	Half-duplex asynchronous transmission
Maximum Baud rate	3 [Mbps] (When using USB)
	16 [Mbps] (When using UART)
Nodes on bus	Up to 256
RS485 line (A, B)	-8 to 13 [V]
UART connection pin	
Vcc voltage	+3.0 to +6.0 [V]
R output voltage	H: Vcc -0.6 [V], L: 0.4 [V]
EN and D input voltage	0 to +6.0 [V]
Logic level	H: 2.0 [V] or higher, L: 0.8 [V] or lower

[Connector Specifications]

Item	Content
USB connector	USB mini B
POW connector	J.S.T. Mfg. Co., Ltd. B2PS-VH
1	V+ (power supply for device)
2	V- (GND)
RS485 connector	J.S.T. Mfg. Co., Ltd. S4B-XH-A
1	GND
2	Power supply output
3	A (D+ RS485 Line)
4	B (D- RS485 Line)
UART connection pin	(Layout at 2.54 [mm] pitch)
EN	Send/receive switch
R	Data receive pin
D	Data transfer pin
GND	GND
Vcc	Power supply for logic IC (3.0–6.0 [V])

[Usage Method]

With USB communication



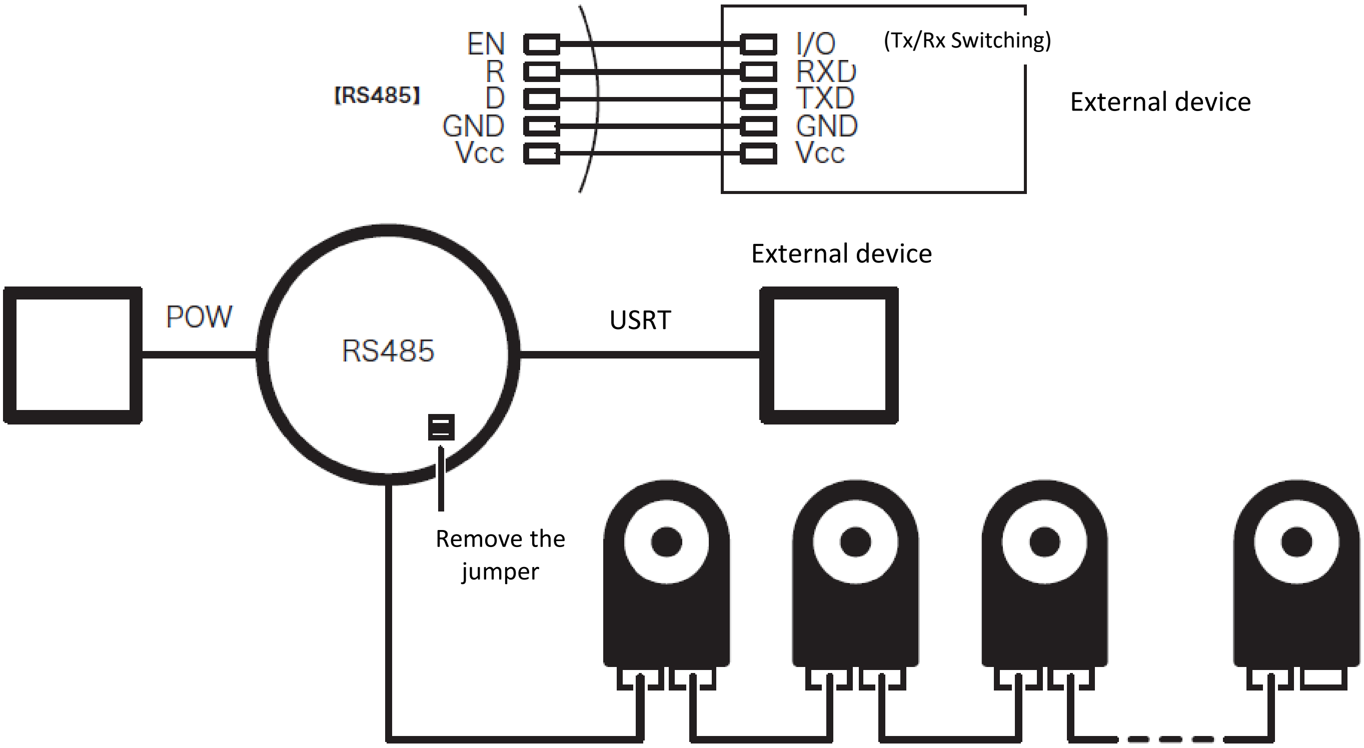
1. Connect JP1 with solder (connected at shipment).

2. Connect to PC and install KO Driver. After the driver is installed, communication is possible with the created virtual COM port. (Refer to the KO Driver installation manual for the KO Driver installation method.)

   KONDO website: Customer Center → Support Info → Software "KO Driver2015"
   http://kondo-robot.com/faq/ko-driver-2015

3. Install B3M Manager software. (Refer to the B3M Manager manual for how to use Manager software.)
   KONDO website: Customer Center → Support Info → B3M Development Documents → "B3M Manager Software"
   http://kondo-robot.com/faq/b3m-manager-software-2

With UART communication



1. Remove the JP1 solder jumper.
2. Connect the external connection device to the UART terminal.
3. When sending data from an external connection device to a 485-compliant device, set the EN terminal to H level and the others to L level.

When using external power supply

Use when supplying power to RS485-compliant devices. It is not necessary when supplying power from an external source using RS485's Vcc/GND line.

1.9 Power Supply

For the power supply, a 12 V DC power supply or battery is required. When connecting a power supply to B3M, make sure the polarity is correct. The required current value (amps) changes depending on the type and number, etc. of the motor used. For details, refer to the specifications, etc. of the motor used.

[Power supply]

B3M servo also requires as its power supply at 9-12 V battery or AC power supply, etc.

・SB-1040-A: Ampacity of Max 2.8A required for each servo unit.

・SC-1040-A: Ampacity of Max 3.6A required for each servo unit.

・SC-1170-A: Ampacity of Max 5.4A required for each servo unit.

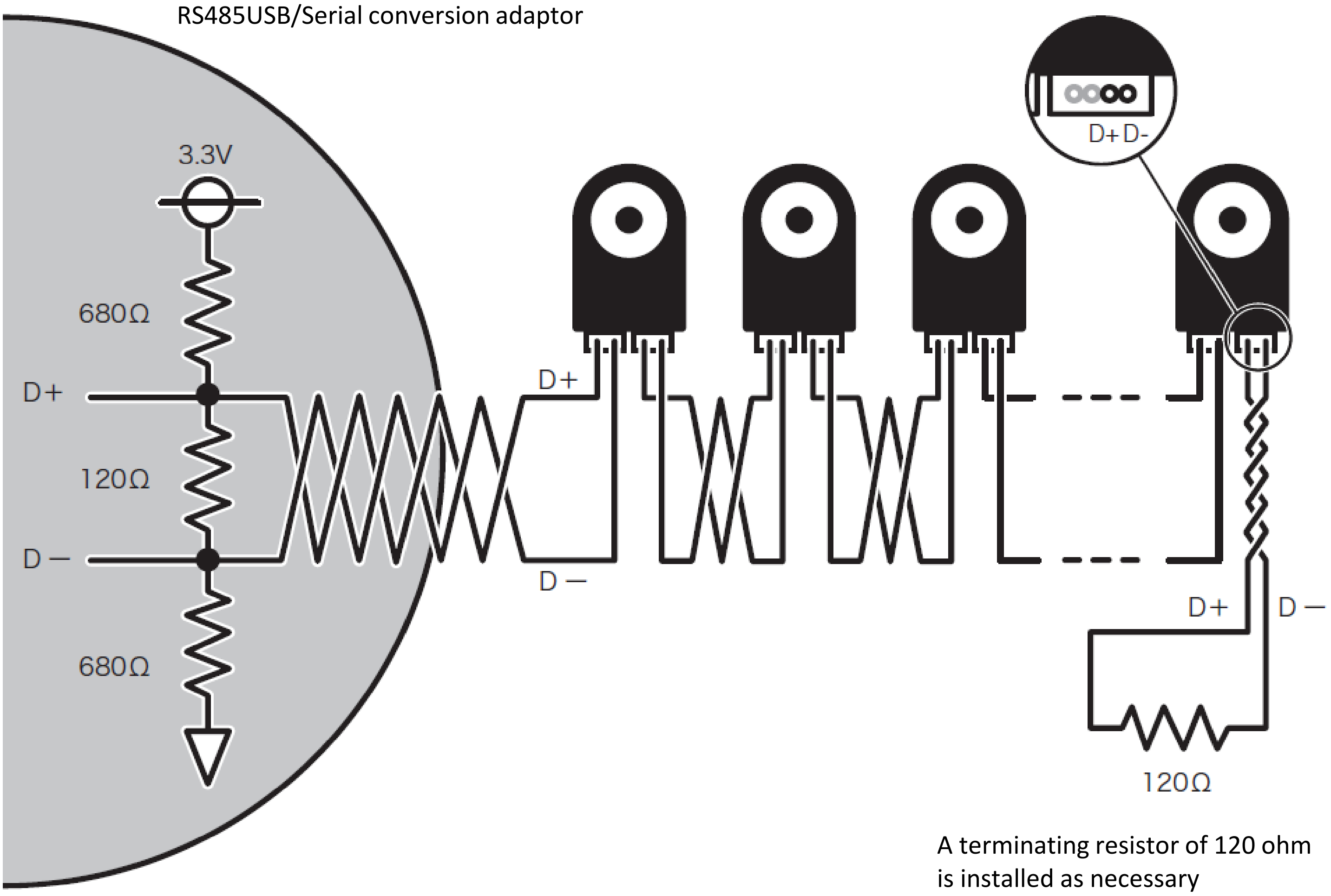
Notes.

Because the B3M series power supply circuit is equipped with a soft start function, the servo may not start properly if the power supply rise is slow. When using the stabilized power supply as the power supply, start the servo stabilization by turning on the switch between the servo and the power supply, turning on the regulated power supply and then turning on the switch.

1.10 About Multidrop Connection

Servos can be connected to each other with multidrop connection. For the connection, XH connection cable is recommended.

When there are multiple servos, a hub can also be used.


RS485USB/Serial conversion adaptor
3.3V
680Ω
D+
120Ω
D－
680Ω
D+
D－
D+D-
D+
D－
120Ω
A terminating resistor of 120 ohm
is installed as necessary

Note.
*Please twist D + and D- to reduce noise.
*To add a terminating resistor, short the D + and D- terminals to 120 ohms.
*To connect more than 10 servos in series when using a 40 cm connection cable, please attach a terminating resistor of 120 ohms. Even when the transmission distance or cable is long.

2. Communication Specifications

2.1 Serial Communication Setting

With the B3M Series, servo operations and servo status can be acquired through commands sent from the host.

Name	Function
Communication format	RS-485 compliant
Transfer mode	Half-duplex asynchronous communication (start-stop synchronous communication)
Baud rate	115200–3000000 bps (recommended 115200, 1000000, 1500000, 2000000, or 3000000bps), Guard time 200 [us] or higher Guard time 200us or higher
Start bit	1 bit
Data bit	8 bit
Parity bit	None
Stop bit	1 bit
Notes	*If communications are conducted at a Baud rate of 2Mbps or more, it may cause a communications error. In such cases, this can sometimes be improved by put the stop bit to 2. *Commands are not recognized as a series of commands if an interval greater than 2 bytes is open. *When communications are conducted consecutively, open the transmission interval to greater than 2 bytes of data + 220us. This differs depending on the transmission data volume and transmission speed.

2.2 Communication Modes

B3M, in addition to normal 1-to-1 communication (single mode), is equipped with Multi mode for communicating with multiple devices at once. For switching modes, when second and subsequent device IDs and data are sent consecutively after sending the first device ID and data, it is automatically sent in Multi mode.

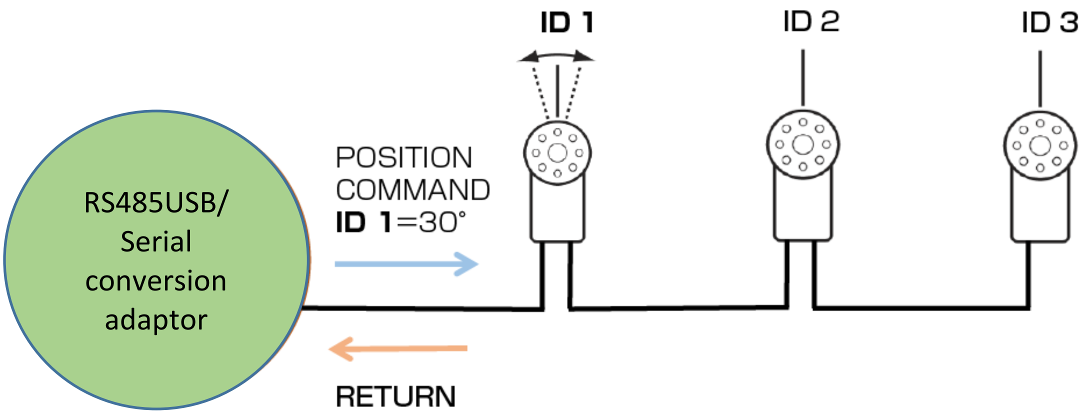
Name	Description
Single	Specify a single ID or the Broadcast ID and execute the command.
Multi	Specify two or more IDs and execute the command. The Broadcast ID cannot be used (is ignored).

About the "Broadcast ID"

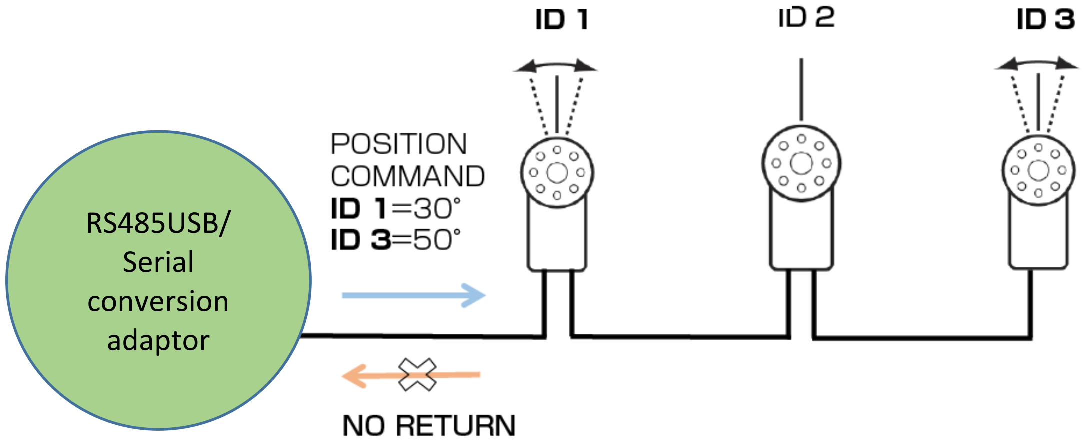
1. Using ID 255, commands can be sent to all devices all at once.
2. The "Broadcast ID" is a special ID number that is set permanently at 255.

*There is no reply data when multi command and the "Broadcast ID" are specified.

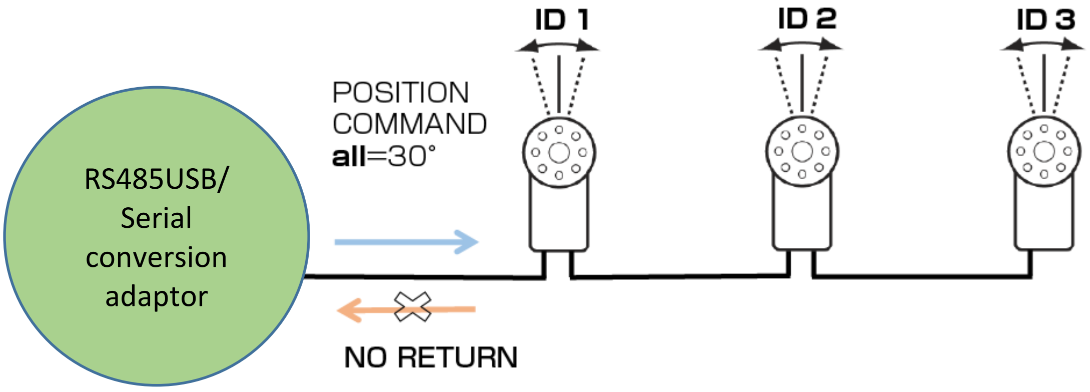
Single mode (Communicate with one servo)



Multi mode (Communicate with multiple servos)



Broadcast ID (Communicate with all servos)



2.3 Data Structure

The data structure consists of two format types, the "Send format" when sending from the host and "Reply format" when sending from the servo.

[Single Mode]

"Send Format"

Content	SIZE	COMMAND	OPTION	ID	S-DATA	SUM
Byte	1	1	1	1	0–N	1

"Reply Format"

Content	SIZE	COMMAND	STATUS	ID	R-DATA	SUM
Byte	1	1	1	1	0–N	1

[Multi mode]

"Send Format"

Content	SIZE	COMMAND	OPTION	ID1	S-DATA	SUM
Byte	1	1	1	1	0–N	1

In Multi mode, commands can be sent to multiple servo IDs by connecting to S-DATA.

"Reply Format"

・There is no reply data.

Refer to "3. Command Details" for formats in Multi mode.

2.4 Explanation of Data Format Names

Various items for each format are introduced here. Refer to this together with "3. Command Details."

"SIZE" Data size

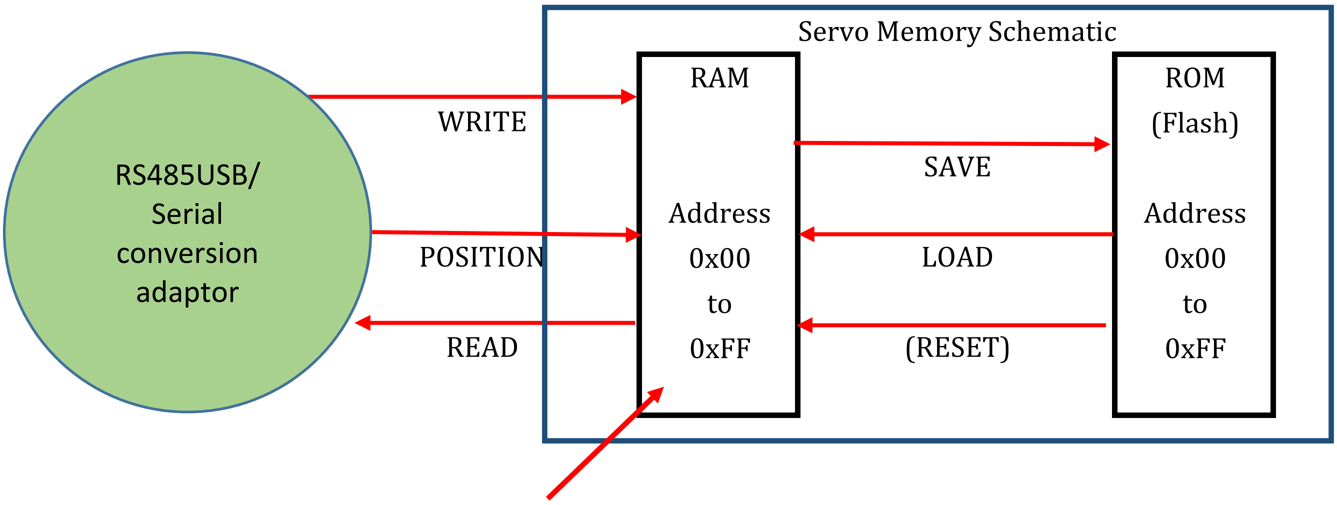
Specify the size of the format (SIZE-SUM). Up to 256 bytes.

"COMMAND" Command

With B3M, servo and data control can be conducted with six types of commands.
"3. Command Details" contains usage examples for each command.

Name	CODE	Function
LOAD	0x01	Loads ROM content into IO map (RAM)
SAVE	0x02	Saves IO map (RAM) content in ROM
READ	0x03	Reads data from device at IO map (RAM) address designation
WRITE	0x04	Writes data (RAM) on device at IO map (RAM) address designation.
RESET	0x05	Restarts device
POSITION	0x06	Changes servo motor position

*For data acquired with the READ command, the data value may change during transmission.



This RAM (0x00–0xFF) is called the memory map.

"OPTION" / "STATUS"

The status of the error item designated in send format OPTION is sent back in reply format STATUS. Also, the order to clear status errors is conducted with OPTION.
For details, refer to "4. OPTION/STATUS Details."

"ID" ID number

Device ID number.

"S-DATA" Data (Send Format)

Send data for each of the commands is sent as S-DATA.
With the READ command, the address and number of bytes to be read are the S-DATA, and in the WRITE command, the information to be written is the S-DATA.
With RESET, time (TIME) to reset is sent with S-DATA.
POSITION allows the operating angle and arrival time to be specified by sending the angle and time. However, the time setting takes effect only when the trajectory generation type (0x29) is set to something other than Nomal mode.
In Multi mode, operating instructions can be given simultaneously by inputted multiple servo IDs into LOAD and SAVE. With WRITE, it is possible simultaneously write to multiple units by alternatively sending multiple data and IDs. With RESET, multiple specified servos can be simultaneously reset. It is the same with POSITION. READ is not compatible with Multi mode.
For details on how to use S-DATA, refer to "3. Command Details".

Name	Single Mode	Multi mode
LOAD	No data	Specifies second and subsequent servo IDs after the first ID (ID2...IDn).
SAVE	No data	Specifies second and subsequent servo IDs after the first ID (ID2・・・IDn).
READ	Address (ADDRES) + Number of bytes (LENGTH)	
WRITE	Write data (DATA1)+ Address (ADDRES)	First write data (DATA1)+ Second and subsequent IDs (IDn)+ Second and subsequent write data (DATAn)+ Initial write address (ADDRES)+ Number of devices specified (COUNT)
RESET	Time (TIME) (How many seconds later to Reset)	Second and subsequent servo ID numbers (ID2...IDn)+Time (TIME)
POSITION	Angle (POS)+ Time (TIME)	First POS (POS1) + Second and subsequent IDs (IDn)+ Second and subsequent angle (POSn) + Time (TIME)

"R-DATA" Data (Reply Format)

With READ, data for each parameter is sent back; with POSITION, it is the Angle at data transfer. With LOAD, SAVE, and WRITE, no R-DATA is sent back. Also. RESET has no reply from servo. In Multi mode, there is no reply in all commands.

Name	Single Mode	Multi mode
LOAD	No data	
SAVE	No data	
READ	Actual data	
WRITE	No data	
RESET		
POSITION	Angle at data transfer (POS)	

"ADDRESS" Address of memory map

Specify the data on the memory map subject to WRITE (Write) and READ (Read) commands. If there are multiple consecutive addresses such as 4BYTE data, specify the start address. For details, refer to "7. Memory Map."

"LENGTH" Data length

Specify the number of R-DATA bytes to be read by the READ command.

"COUNT" Number of devices

Specify the ID number of the device to be specified with the WRITE command.

"POS" Position data

Specifies the angle with the SET POSITION command.

"TIME" Time of motion

In RESET, Specify the time until resetting.

Specifies the shaft arrival time with the SET POSITION command. However, the time setting takes effect only when the trajectory generation type (0x29) is set to something other than Nomal mode.

"SUM" Checksum

Low-order byte of command data sum, excluding SUM.

3. Command Details

3.1 List of Commands

Servo and data control can be conducted with six simple types of commands.
In Single mode, a reply can be received from the device after the command is sent. The commands include STATUS, which provides information on the device's status so it can be confirmed. In Multi mode, commands other than READ can be sent simultaneously to multiple servos. However, in this mode, it is not possible to receive a reply from a device.

Name	Mode	Command	Function
LOAD	Send (S/M)	0x01	ROM content is loaded into the Memory map (RAM).
	Reply	0x81	STATUS (servo status) and ID are sent back. *No reply in Multi mode.
SAVE	Send (S/M)	0x02	Memory map (RAM) content is saved in ROM.
	Reply	0x82	STATUS (servo status) and ID are sent back. *No reply in Multi mode.
READ	Send (S only)	0x03	Data (RAM) is read from the device at Memory map (RAM) address designation.
	Reply	0x83	Data read at the designated address is sent back.
WRITE	Send (S/M)	0x04	Data (RAM) is written to the device at the Memory map (RAM) address designation.
	Reply	0x84	STATUS (servo status) and ID are sent back. *No reply in Multi mode.
RESET	Send (S/M)	0x05	The device is restarted.
	Reply	–	No reply
POSITION	Send (S/M)	0x06	Servo motor position is changed.
	Reply	0x86	STATUS (servo status), ID and current shaft position are sent back. *No reply in Multi mode.

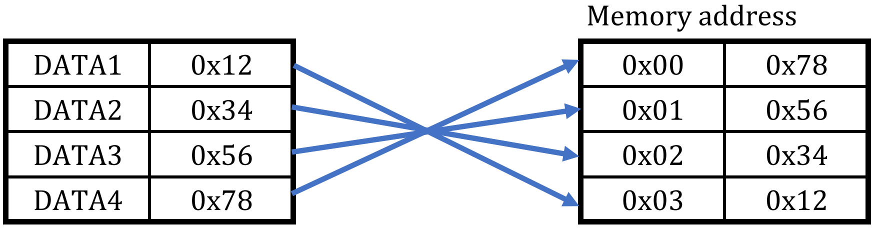
In the mode column, S=Single Mode, M=Multi mode

"S/M" means the function is available in both modes.

3.2 About Data Integration and Segmentation

When reading and writing Memory map (RAM) content using the READ and WRITE commands, with two-byte and four-byte items, data is stored in the little-endian format.

If the communication format data item is long (4-byte)



DATA1	0x12
DATA2	0x34
DATA3	0x56
DATA4	0x78

Memory address

0x00	0x78
0x01	0x56
0x02	0x34
0x03	0x12

For 2-byte data

When write data is 0x1234

DATA (1) (2)
0x1234

DATA (1)	DATA (2)
0x34	0x12

When read data is 0x5678

DATA (1)	DATA (2)
0x78	0x56

DATA (1) (2)
0x5678

For 4-byte data

When write data is 0x12345678

DATA (1) (2) (3) (4)
0x12345678

DATA (1)	DATA (2)	DATA (3)	DATA (4)
0x78	0x56	0x34	0x12

When read data is 0x12345678

DATA (1)	DATA (2)	DATA (3)	DATA (4)
0x78	0x56	0x34	0x12

DATA (1) (2) (3) (4)
0x12345678

3.3 LOAD Command

Read contents of ROM to Memory map (RAM)

Communication mode	Single	Multi

3.3.1 Send Format in Single Mode

Digit	1	2	3	4	5
Item	SIZE	COMMAND	OPTION	ID	SUM

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x05		Number of data bytes in a command sequence
2	COMMAND	0x01		
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID	0x00	0xFF	ID number of LOAD device
5	SUM			Low-order byte of sum of SIZE to ID

3.3.2 Multi mode Send Format

Instructions can be sent to more than one device at a time by specifying the first ID followed by the second and subsequent servo ID numbers.

Digit	1	2	3	4		n+2	n+3	n+4
Item	SIZE	COMMAND	OPTION	ID1	...	ID (n-1)	ID (n)	SUM

(n: the specified number of devices)

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x05	0xFF	Number of data bytes in a command sequence
2	COMMAND	0x01		
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID1	0x00	0xFE	ID number of LOAD device
...				
n+3	ID (n)	0x00	0xFE	ID number of LOAD device
n+4	SUM			Low-order byte of sum of SIZE to ID (n)

3.3.3 Reply Data (Single Mode Only)

*No reply when Broadcast ID designated or with Multi mode commands

Digit	1	2	3	4	5
Item	SIZE	COMMAND	STATUS	ID	SUM

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x05		Number of data bytes in a command sequence
2	COMMAND	0x81		
3	STATUS	Changes with OPTION		Refer to "4. OPTION/STATUS Details"
4	ID	Same as send ID		ID of device that sends reply
5	SUM			Low-order byte of sum of SIZE to ID

3.3.4 Example of use

Use the LOAD command to call the contents of ROM of ID1 servo to the Memory map (RAM).

Send Data

Digit	Item	Data
1	SIZE	0x05
2	COMMAND	0x01
3	OPTION	0x00
4	ID	0x01
5	SUM	0x07

Reply Data

Digit	Item	Data
1	SIZE	0x05
2	COMMAND	0x81
3	STATUS	0xXX
4	ID	0x01
5	SUM	0xXX

*0xXX is indefinite.

3.4 SAVE Command

Memory map (RAM) content is saved in ROM.

Communication mode	Single	Multi

3.4.1 Single Mode Send Format

Digit	1	2	3	4	5
Item	SIZE	COMMAND	OPTION	ID	SUM

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x05		Number of data bytes in a command sequence
2	COMMAND	0x02		
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID	0x00	0xFF	ID number of SAVE device
5	SUM			Low-order byte of sum of SIZE to ID

3.4.2 Multi mode Send Format

Instructions can be sent to more than one device at a time by specifying the first ID followed by the second and subsequent servo ID numbers.

Digit	1	2	3	4		n+2	n+3	n+4
Item	SIZE	COMMAND	OPTION	ID1	…	ID (n-1)	ID (n)	SUM

(n: the specified number of devices)

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x05	0xFF	Number of data bytes in a command sequence
2	COMMAND	0x02		
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID1	0x00	0xFE	ID number of SAVE device
…				
n+3	ID (n)	0x00	0xFE	ID number of SAVE device
n+4	SUM			Low-order byte of sum of SIZE to ID (n)

3.4.3 Reply Data (Single Mode Only)

*No reply when Broadcast ID designated or with Multi mode commands

Digit	1	2	3	4	5
Item	SIZE	COMMAND	STATUS	ID	SUM

Data Details

Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x05		Number of data bytes in a command sequence
2	COMMAND	0x82		
3	STATUS	Changes with OPTION		Refer to "4. OPTION/STATUS Details"
4	ID	Same as SEND ID		ID of device that sends reply
5	SUM			Low-order byte of sum of SIZE to ID

3.4.4 Example of use

Content of Memory map (RAM) with ID1 servo saved in ROM

Send Format

Digit	Item	Data
1	SIZE	0x05
2	COMMAND	0x02
3	OPTION	0x00
4	ID	0x01
5	SUM	0x08

Reply Format

Digit	Item	Data
1	SIZE	0x05
2	COMMAND	0x82
3	STATUS	0xXX
4	ID	0x01
5	SUM	0xXX

*0xXX is indefinite.

3.5 READ Command

Reads RAM data from the device at the Memory map (RAM) address designation.

Communication mode	Single

3.5.1 Single Mode Send Format

Digit	1	2	3	4	5	6	7
Item	SIZE	COMMAND	OPTION	ID	ADDRESS	LENGTH	SUM

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x07		Total number of command bytes
2	COMMAND	0x03		Command
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID	0x00	0xFE	ID number (0xFF (broadcast) gives an error)
5	ADDRESS	0x00	0xFF	Initial data read address (refer to the Memory map)
6	LENGTH	0x01	0xFA	Length of read data (0x00 gives an error; can only read up to 250 bytes)
7	SUM			Low-order byte of sum of SIZE to LENGTH

3.5.2 Reply Format

Digit	1	2	3	4	5	...	n+4	n+5
Item	SIZE	COMMAND	STATUS	ID	DATA (1)	...	DATA (n)	SUM

(n: number of data sent back)

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	For DATA (n), n+5		Total number of command bytes
2	COMMAND	0x83		Command
3	STATUS	Changes with OPTION		Refer to "4. OPTION/STATUS Details"
4	ID	Same as send ID		ID of device that sends reply
5	DATA	0x01	0xFF	Actual data (number of data determined by send format LENGTH)
...				
n+5	SUM			Low-order byte of sum of SIZE to DATA (n)

3.5.3 Example of use

Reads 4 bytes of data from 0xA2 address on ID0 servo.

Send Format

Digit	Item	Data
1	SIZE	0x07
2	COMMAND	0x03
3	OPTION	0x00
4	ID	0x00
5	ADDRESS	0xA2
6	LENGTH	0x04
7	SUM	0xB0

Reply Format

Digit	Item	Data
1	SIZE	0x09
2	COMMAND	0x83
3	STATUS	0xXX
4	ID	0x00
5	DATA1	0xXX
6	DATA2	0xXX
7	DATA3	0xXX
8	DATA4	0xXX
9	SUM	0xXX

*0xXX is indefinite.

3.6 WRITE Command

Writes on device RAM at the Memory map (RAM) address designation.

Communication mode	Single	Multi

3.6.1 Single Mode Send Format

Digit	1	2	3	4	5		m+4
Item	SIZE	COMMAND	OPTION	ID	DATA (1)	...	DATA (m)

Digit	(m+1)+4	(m+1)+5	(m+1)+6
Item	ADDRESS	COUNT	SUM

m: Number of bytes to be written on one device.

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x08	0xFF	Total number of command bytes
2	COMMAND	0x04		Command
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID	0x00	0xFF	ID number (*)
5	DATA (1)	0x00	0xFF	Data to be written (low-order byte comes first)
...				(short/long in little-endian format)
(m+1)+4	ADDRESS	0x00	0xFF	Initial address of data to be written (refer to Memory map)
(m+1)+5	COUNT	0x01	0x01	Number of specified devices (0x00 gives an error)
(m+1)+6	SUM			Low-order byte of sum of SIZE to COUNT

3.6.2 Multi mode Send Format

Digit	1	2	3
Item	SIZE	COMMAND	OPTION

Digit	4	5		m+4	(m+1)+3	(m+1)+4		(m+1)+3
Item	ID1	DATA1 (1)	...	DATA1 (m)	ID2	DATA2 (1)	...	DATA2 (m)

ID1

ID2

Digit	n*(m+1)+3	n*(m+1)+4		n*(m+1)+3	n*(m+1)+4	n*(m+1)+5	n*(m+1)+6
Item	ID (n)	DATA (n) (1)	...	DATA (n) (m)	ADDRESS	COUNT	SUM

ID(n)

m: Number of bytes to be written on one device.

n: Number of devices to be written on (same number as COUNT)

Data Details

Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x08	0xFF	Total number of command bytes
2	COMMAND	0x04		Command
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID1	0x00	0xFF	ID number (*)
5	DATA1 (1)	0x00	0xFF	Data to be written (low-order byte comes first)
...				(short/long in little-endian format)
n*(m+1)+4	ADDRESS	0x00	0xFF	Initial address of data to be written (refer to Memory map)
n*(m+1)+5	COUNT	0x01	0xFF	Number of specified devices (0x00 gives an error)
n*(m+1)+6	SUM			Low-order byte of sum of SIZE to COUNT

3.6.3 Reply Data (Single Mode Only)

*No reply when Broadcast ID designated or with Multi mode commands

Digit	1	2	3	4	5
Item	SIZE	COMMAND	STATUS	ID	SUM

Data Details

Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x05		Total number of command bytes
2	COMMAND	0x84		Command
3	STATUS	Changes with OPTION		Refer to "4. OPTION/STATUS Details"
4	ID	Same as send ID		ID of device that sends reply
5	SUM			Low-order byte of sum of SIZE to ID

*When the below conditions are met, the item with the earlier index number is processed; the rest are invalid.

1. The same ID is saved.
2. Both the normal ID and Broadcast ID are specified.

*If the length of ID (1) to DATA (n) (m) is indivisible by COUNT, the command is invalid.

3.6.4 Example of use

Writes 4 bytes of data from the 0x05 address to the servo with ID 1.3.5.

ID=1 Min. position limit (0x05.0x06 address)=-32000 (0x8300)
ID=1 Max. position limit (0x07.0x08 address)=32000 (0x7D00)
ID=3 Min. position limit (0x05.0x06 address)=-31000 (0x86E8)
ID=3 Max. position limit (0x07.0x08 address)=32000 (0x7918)
ID=5 Min. position limit (0x05.0x06 address)=-30000 (0x8AD0)
ID=5 Max. position limit (0x07.0x08 address)=30000 (0x7530)

Send Format

Digit	Item	Data	ID no.	Address
1	SIZE	0x07		
2	COMMAND	0x04		
3	OPTION	0x00		
4	ID(1)	0x01	1	
5	DATA1 (1)	0x00		0x05
6	DATA1 (2)	0x83		0x06
7	DATA1 (3)	0x00		0x07
8	DATA1 (4)	0x7D		0x08
9	ID(2)	0x03	3	
10	DATA2 (1)	0xE8		0x05
11	DATA2 (2)	0x86		0x06
12	DATA2 (3)	0x18		0x07
13	DATA2 (4)	0x79		0x08
14	ID(3)	0x05	5	
15	DATA3 (1)	0xD0		0x05
16	DATA3 (2)	0x8A		0x06
17	DATA3 (3)	0x30		0x07
18	DATA3 (4)	0x75		0x08
19	ADDRESS	0x05		
20	COUNT	0x03		
21	SUM	0x1A		

*With the above commands, there is no reply data.

3.7 RESET Command

Restarts the device.
ROM content is loaded into the Memory map (RAM).
*RAM content is overwritten, so if you want to save the data, execute the command after saving the data in ROM using the SAVE command.

Communication mode	Single	Multi

3.7.1 Single Mode Send Format

Digit	1	2	3	4	n+4	n+5
Item	SIZE	COMMAND	OPTION	ID	TIME	SUM

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x06	0xFF	Total number of command bytes
2	COMMAND	0x05		Command
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID	0x00	0xFF	ID number (0xFF resets all devices)
5	TIME	0x00	0xFF	Specified time of 0~25.5 sec (units of 100 ms)
6	SUM			Low-order byte of sum of SIZE to TIME

*There is no reply to the RESET command.

3.7.2 Multi mode Send Format

Digit	1	2	3
Item	SIZE	COMMAND	OPTION

Digit	4	5		n+2	n+3	n+4	n+5
Item	ID1	ID2	...	ID (n-1)	ID (n)	TIME	SUM

n: Number of devices specified

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x06	0xFF	Total number of command bytes
2	COMMAND	0x05		Command
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID	0x00	0xFF	ID number (0xFF resets all devices)
...				
n+4	TIME	0x00	0xFF	Specified time of 0~25.5 sec (units of 100 ms)
n+5	SUM			Low-order byte of sum of SIZE to TIME

*There is no reply to the RESET command.

3.7.3 Example of use

Reset ID0 servo after 300 ms.

Send Format

Digit	Item	Data
1	SIZE	0x06
2	COMMAND	0x05
3	OPTION	0x00
4	ID	0x00
5	TIME	0x03
6	SUM	0x0E

3.8 SET POSITION Command

Specifies the servo motor's operating angle and arrival time to change the position.

The arrival time is valid only when the orbit generation type (0x29) is set to "Even", "ThirdPoly", "ForthPoly", and "FifthPoly". It is not reflected in "Normal".

Communication mode	Single	Multi

3.8.1 Single Mode Send Format

Digit	1	2	3	4	5	6
Item	SIZE	COMMAND	OPTION	ID	POS_L	POS_H

POS

Digit	7	8	9
Item	TIME_L	TIME_H	SUM

TIME

Data Details

Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x09	0xFF	Total number of command bytes
2	COMMAND	0x06		Command
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID	0x00	0xFF	
5	POS_L	0x00	0xFF	Low-order byte of position data
6	POS_H	0x00	0xFF	High-order byte of position data
7	TIME_L	0x00	0xFF	Target time of 0~65535 ms; effective when Trajectory other than 0
8	TIME_H	0x00	0xFF	
9	SUM			Low-order byte of sum of SIZE to TIME_H

For details on position data, refer to "About Position Data" below.

"Trajectory generation type(0x29)"

The trajectory generation type is changed by rewriting the data at address 0x29 with the WRITE command.

For details, please refer to the section "Activation Generation Type" in "8. Detailed Memory map".

(0) Normal: Rotates at maximum speed.

(1) Even: Evenly divides target time and interpolates from start point to end point (trajectory generation).

(3) ThirdPoly: Interpolates from start point to end point using third-order polynomial.

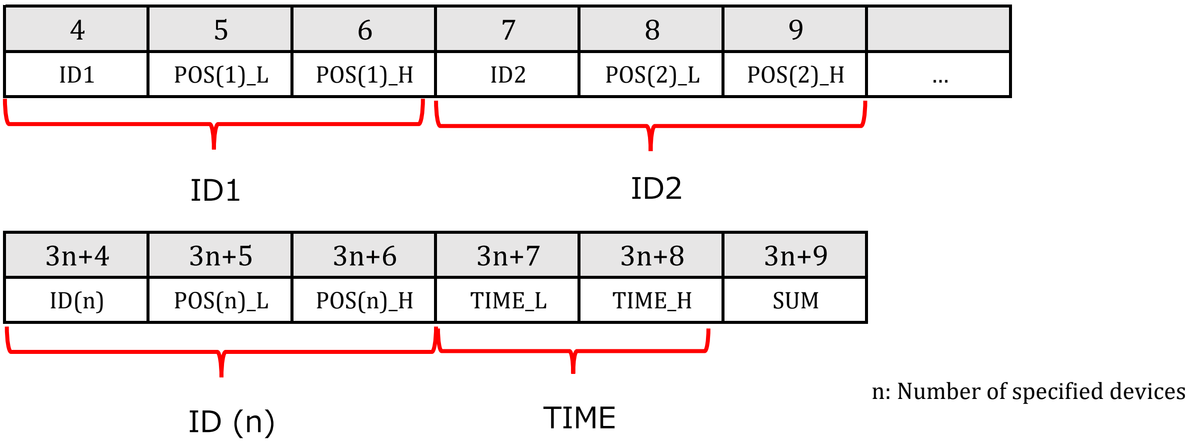
(4) ForthPoly: Interpolates from start point to end point with 4-1-4-order polynomial interpolation method.

(5) FifthPoly: Interpolates from start point to end point using fifth-order polynomial.

*Only effective in position control mode when operation mode is "Normal".

3.8.2 Multi mode Send Format

Digit	1	2	3
Item	SIZE	COMMAND	OPTION



4	5	6	7	8	9	
ID1	POS(1)_L	POS(1)_H	ID2	POS(2)_L	POS(2)_H	...

3n+4	3n+5	3n+6	3n+7	3n+8	3n+9
ID(n)	POS(n)_L	POS(n)_H	TIME_L	TIME_H	SUM

n: Number of specified devices

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x09	0xFF	Total number of command bytes
2	COMMAND	0x06		Command
3	OPTION			Refer to "4. OPTION/STATUS Details"
4	ID1	0x00	0xFF	ID number (prioritize subsequent data with same ID designation)
5	POS(1)_L	0x00	0xFF	Low-order byte of position data
6	POS(1)_H	0x00	0xFF	High-order byte of position data
...				
3n+7	TIME_L	0x00	0xFF	Target time of 0~65535 ms; effective when Trajectory other than 0
3n+8	TIME_H	0x00	0xFF	
3n+9	SUM			Low-order byte of sum of SIZE to TIME_H

For details on position data, refer to "About Position Data" below.

3.8.3 About Position Data

B3M operates within an operating angle range of -320° to 320°.

The parameter is the range of -32000 to 32000 in Decimal number, but in Hexadecima, the values specified for counter-clockwise and clockwise rotation differ. Refer to the following for details.

		Counter -clockwise Max.	Center	Clockwise Max.
Angle		-320°	0°	320°
Parameter	DEC	-32000	0	32000
	HEX	0x8300	0x00	0x7d00

The list of maximum values above are values at shipment.

The maximum operating angle can be changed at address 0x05 (min.) and 0x07 (max.).

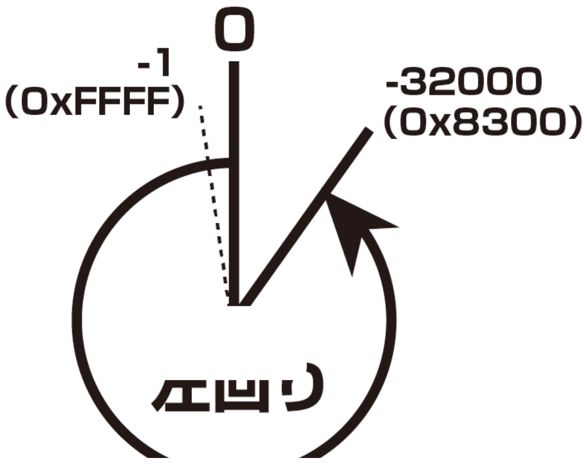
The operating angle is not changed beyond the set maximum even if a higher angle is specified.

With clockwise rotation, when the parameter is "1" in Decimal number, in Hexadecima, it is "0x0001," but with counter-clockwise rotation, when it is "-1" in Decimal number, it is "0xFFFF," and when the max. value is "-32000," it is "0x8300," so values are in the decreasing direction.

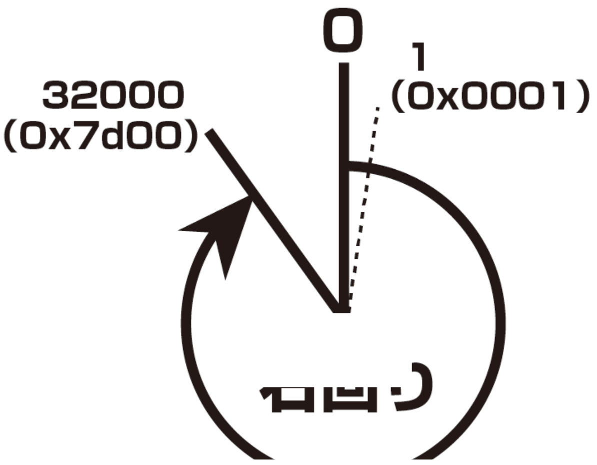
For both counter-clockwise and clockwise rotation, the range that can be set is 31999 of 1–32000 in Decimal number and 7CFF in Hexadecima.

The center is 0.

		Counter-clockwise	Center	Clockwise
Parameter	DEC	-1	0	1
	HEX	0xFFFF	0x00	0x0001



Counterclockwise



Clockwise

3.8.4 Reply Data (no reply when Broadcast ID designated or with Multi mode commands)

Digit	1	2	3	4	5	6	7
Item	SIZE	COMMAND	STATUS	ID	POS_L	POS_H	SUM

POS

Data Details				
Digit	Item	Setting Range		Content
		Min.	Max.	
1	SIZE	0x07		Total number of command bytes
2	COMMAND	0x86		Command
3	STATUS	Changes with OPTION		Refer to "4. OPTION/STATUS Details"
4	ID	Same as send ID		ID of device that sends reply
5	POS_L	0x00	0xFF	Low-order byte of present position
6	POS_H	0x00	0xFF	High-order byte of present position
7	SUM			Low-order byte of sum of SIZE to POS_H

3.8.5 Example of use

Move the servo position of ID 2 to the 180 deg at 3000 ms(3sec).

Send Format

Digit	Item	Data
1	SIZE	0x09
2	COMMAND	0x06
3	OPTION	0x00
4	ID1	0x02
5	POS(1)_L	0x50
6	POS(1)_H	0x46
7	TIME_L	0xB8
8	TIME_H	0x0B
9	SUM	0x6A

Reply Format

Digit	Item	Data
1	SIZE	0x07
2	COMMAND	0x86
3	STATUS	0xXX
4	ID	0x02
5	POS_L	0xXX
6	POS_H	0xXX
7	SUM	0xXX

*0xXX is indefinite.

4. OPTION/STATUS Details

OPTION and STATUS functions make it possible to acquire servo status information.
With OPTION, data to be read can be specified, and in response the servo sends back its STATUS.
They are not independent commands, but rather are used as a part of the send and reply formats of each command.

* Error status will not be cleared automatically. Be sure to manually clear with the optional CLEAR (7 bit).

4.1 OPTION

With OPTION, the status to be read can be specified from among five types: Error Status, System Status, Motor Status, Communication Status, and Command Status.

Bit	Name	Function	Notes
0	STATUS SELECT	Retrieves using 0~2 bit Select STATUS type	000: ERROR STATUS returned 001: SYSTEM STATUS returned 010: MOTOR STATUS returned 011: UART STATUS returned 100: COMMAND STATUS returned
1			
2			
3	–		
4	–		
5	–		
6	–		
7	CLEAR	Clears STATUS	Clears all STATUS to 0; status prior to CLEAR saved in reply command.

4.2 STATUS

STATUS specified in OPTION is sent back.

The following explains, in list format, the details of the status information that is sent back.

[ERROR STATUS (When STATUS SELECT in OPTION set at "000")]

Representative bits for each error status.

Bit	Content	
	0	1
0	No error	When any of the SYSTEM STATUS error bits is 1
1	No error	When any of the MOTOR STATUS error bits is 1
2	No error	When any of the UART STATUS error bits is 1
3	No error	When any of the COMMAND STATUS error bits is 1
4	－	－
5	－	－
6	－	－
7	－	－

[SYSTEM STATUS (When STATUS SELECT in OPTION set at "001")]

・System-related error bits.

Bit	Content	
	0	1
0	No error	1 when Watchdog Timer is started
1	No error	1 when a problem with data saved in MCU ROM
2	No error	1 when problem with data and RAM allocation fails; only checks once at start-up.
3	No error	1 when input voltage exceeds maximum or is lower than minimum value
4	No error	1 when MCU temperature exceeds maximum value
5	No error	1 when AD conversion fails
6	No error	1 when I2C communication fails (unused)
7	No error	1 when SPI communication fails

* Error status may have some invalid items depending on the servo motor type.

B3M-SB-1040-A	SYSTEM: Watchdog Timer	
B3M-SC-1040-A, B3M-SC-1170-A	SYSTEM: Watchdog Timer	Motor: Hall-IC

[MOTOR STATUS (When STATUS SELECT in OPTION is set at "010")]

Motor-related error bits.

Bit	Content	
	0	1
0	No error	1 when motor temperature exceeds maximum value
1	No error	1 when motor lock detected
2	No error	1 when current flowing to motor exceeds maximum value
3	No error	1 when problem with brushless motor's Hall-IC
4	No error	–
5	No error	–
6	No error	–
7	No error	–

[UART STATUS (When STATUS SELECT in OPTION set at "011")]

Communication-related error bits.

Bit	Content	
	0	1
0	No error	1 when framing error occurs
1	No error	1 when parity error occurs
2	No error	1 when break error occurs
3	No error	1 when overrun error occurs
4	No error	–
5	No error	–
6	No error	–
7	No error	–

[COMMAND STATUS (When STATUS SELECT in OPTION set at "100")]

Command-related error bits.

Bit	Content	
	0	1
0	No error	1 when problem with command checksum
1	No error	1 when command device number is too many or too few
2	No error	1 when length of data to be acquired is longer than address
3	No error	1 when address out of specified range
4	No error	1 when problem with command itself
5	No error	–
6	No error	–
7	No error	–

[CLR (Clear bits in OPTION)]

Bit	Content	
	0	1
7	No clear	All STATUS cleared to 0, and status prior to clear saved in reply command.

* The various error statuses are not automatically cleared.

5. Servo Operations

5.1 Operating mode

1.There are three types of servo status.
Free: No power applied to the motor.
Hold: Magnetic brakes applied to the motor.
Normal: Control starts based on set point of each control mode.

2.When switched on, the servo starts up in Free mode.
The servo can be changed to Free mode in the address (0x28) run mode.

3.Switching to each operation mode can be changed by rewriting the address 0x28 with the WRITE command.

5.2 Control Mode

There are four types of servo control modes: Position Control Mode, Speed Control Mode, Currency (Torque) Control Mode, and Feedforward Mode. They can be selected depending on the application used.
*PID control is conducted in all modes.

Control Mode (0x28)	Set Point	Trajectory Generation Type (0x29)	Target Time (0x36)	Control Parameter (0x5c–0x8c)
Position Control Mode	Target Position (0x2A・0x2B)	(0) Normal: Rotation at max. speed.		Proportional gain Derivative gain Integral gain Coefficient of static friction Coefficient of dynamic friction
		(1) Even: Interpolates from start-point to end-point by evening segmenting target time (trajectory generation).	Target Travel Time	
		(3) ThirdPoly: Interpolates from start-point to end-point using third-order polynomial.		
		(4) FourthPoly: Interpolates from start-point to end-point with 4-1-4-order polynomial interpolation method.		
		(5) FifthPoly: Interpolates from start-point to end-point using fifth-order polynomial.		
Speed Control Mode	Target Speed (0x30・0x31)			
Current (Torque) Control Mode	Target Torque (0x3C・0x3D)			
Feedforward Mode	Target Position (PWM) (0x2A・0x2B)			

5.3 Change of Operating mode mode / Control mode

Switching between Operating mode mode and Control mode is set by rewriting address 0x28.

Address:0x28	Mode
BIT0	Operating mode
BIT1	
BIT2	Control mode
BIT3	

[Operating mode]
(00) Normal: Mode in which servo motor is active.
(10) Free: Mode in which servo motor is de-powered. Motor is not controlled.
(11) Hold: Mode in which servo motor maintains the current position. Motor is not controlled.

[Control mode]
(00) Position: Position control mode.
(01) Speed: Speed control mode.
(10) Current: Current control mode.
(11) Feed Forward: Feedforward mode. In feedforward, if the PWM count value is inputted in the target position, the motor rotates.

*Control mode: Motor does not operate unless operating mode is set to Normal mode.
*To switch to speed control mode, switch first to position control mode, set the current position and then switch to the mode.
*When switching the control mode, always move to FREE, HOLD and set initial values and then switch the mode.
*When switching modes, change the gain preset or adjust the gain.
*Can only be accessed in 1 byte units. When writing, set all bits.

5.4 Gain Preset

The following data is registered at address 0x5C as a preset of motion gain.

Preset 0: Position Control Mode
Preset 1: Speed Control Mode
Preset 2: Current (Torque) Control Mode

When gain preset numbers(address:0x5C) are changed with WRITE command, gains are immediately changed and incorporated into servo motor operations. Gains differs with position control, speed control and torque control, so when changing the control mode, always change the gain or gain preset.

5.5 Position Control Mode

1. Moves to specified position (angle).
2. Control conducted based on target position and current position.
3. Control can be conducted in a range of ±320 degrees from the starting point (value set with medium value

   *In Free and Hold, it is restricted to current position of ±320 degrees, and angles greater than this cannot be assured.

   *When reset is applied, and it is not within the range of ±180 degrees, it is forcefully converted to a position of ±180 degrees.
4. To specify the target position, there are two methods: specifying using the SET POSITION command, and writing data on the target position (0x2A・0x2B).
5. Trajectory generation type can be specified with (0x29).
6. By changing the target time (0x36) of the trajectory generation type, it is possible to change servo operations.
7. The initial gain status is saved in gain 0 (specified with preset number address (0x5C)).

5.6 Speed Control Mode

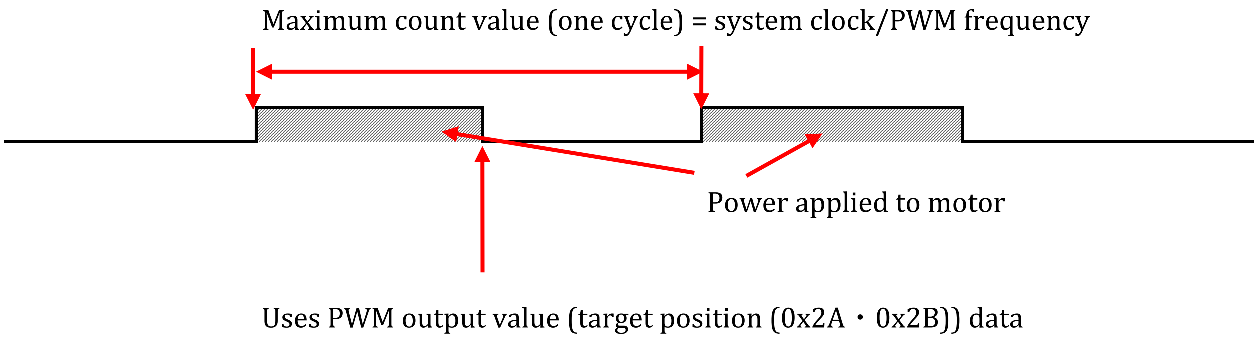
1. Rotation is conducted at the fixed speed that is specified.
2. Control is conducted based on the target speed and current speed.
3. To change the target speed, change the value of (0x30・0x31).
4. Current position (0c2C・0x2D) can be acquired, but the range is within ±180 deg.
5. When acquiring the cumulative rotation angle, it can be calculated backwards from the encoder's total count number (0x52–0x55).
6. The initial gain status is saved in gain 1 (specified with preset number address (0x5C)).

5.7 Current (Torque) Control Mode

1. The control method is to calculate the current value from the target torque and feed back the current value.
2. Control is conducted to maintain certain fixed current.
3. Current (workload) and torque have a fixed relationship, so it is possible to apply power at a fixed torque.
4. To output a fixed power, it is endlessly rotated.
5. To change the target torque, change the value of (0x3C・0x3D).
6. It is possible that the accurate torque will not be outputted depending on the motor temperature, etc.
7. When acquiring the cumulative rotation angle, it can be calculated backwards from the encoder's total count number (0x52–0x55).
8. The initial gain status is saved in gain 2 (specified with preset number address (0x5C)).

5.8 Feedforward Mode

1. Motor output is specified directly.
2. When acquiring the cumulative rotation angle, it can be calculated backwards from the encoder’s total count number (0x52–0x55).


Maximum count value (one cycle) = system clock/PWM frequency
Power applied to motor
Uses PWM output value (target position (0x2A・0x2B)) data

3. The maximum count value at a frequency of 8 kHz is 6250, so in feedforward mode, even if the target position (PWM count value) is set above 6250, it is automatically corrected to the maximum value.

Frequency (Hz)	Max. Count
1000	50000
1250	40000
1562	32010.24328
1563	31989.76328
1600	31250
2000	25000
2500	20000
4000	12500
5000	10000
8000	6250
10000	5000
12500	4000
20000	2500
25000	2000
40000	1250
50000	1000

Settable limit

(Maximum value of target position)

6. Operating B3M Servo Motor

6.1 To operate B3M Servo Motor

The following procedures are necessary to operate the B3M servo motor.

1. Set servo status to Free mode.
2. Change the control mode.
3. Change the gain based on the control mode.
4. Set servo status to Normal mode.
5. Set target position.

This section explains how to operate the servo motor in position control mode.

*Explained with a ID=0 B3M servo motor.

[packet names]

SIZE: Data length
CMD ＝ COMMAND
OP ＝ OPTION
STTS ＝ STATUS
ID: ID number
DATA ＝ S-DATA
ADR ＝ ADDRESS
CNT ＝ COUNT: Number of devices
SUM: Checksum

6.1.1 Set servo status to free mode.

First, decrease power so servo motor does not abruptly operate. The motor is not controlled.

[Set operating mode to Free mode and send command]

Write 0x02 value to 0x28 address with WRITE command.

Send Format

Tx	1	2	3	4	5	6	7	8
	SIZE	CMD	OP	ID	DATA	ADR	CNT	SUM
	0x08	0x04	0x00	0x00	0x02	0x28	0x01	0x37

Address:0x28				Control		Operating	
BET7	BET6	BET5	BET4	BET3	BET2	BET1	BET0
0	0	0	0	0	0	1	0

> Operating mode

(00)Normal

(10)Free

(11)Hold

Reply Format

Rx	1	2	3	4	5
	SIZE	CMD	STTS	ID	SUM
	0x05	0x84	0x00	0x00	0x89

*In the above case there is no error in servo operation. If an error is detected, STATUS changes.

*The following WRITE command also returns 5 BYTE data from the servo regardless of the number of DATA transmitted.

6..1.2 Change the control mode

Set the control mode to Position Control Mode.

[Set Operating mode to Position Control Mode and send command]

Write 0x02 value to 0x28 address with WRITE command.

*Becomes value of OR operation between Operation mode 0x02 and Control mode 0x00.

* The contents of the following command is the same as the command set to the previous Free, but it is posted as a control procedure.

Send Format

Tx	1	2	3	4	5	6	7	8
	SIZE 0x08	CMD 0x04	OP 0x00	ID 0x00	DATA 0x02	ADR 0x28	CNT 0x01	SUM 0x37

Address:0x28				Control		Operating	
BET7	BET6	BET5	BET4	BET3	BET2	BET1	BET0
0	0	0	0	0	0	1	0

※ The Operating mode is Free.

> Control Mode

(00)Position Control Mode

(01)Speed Control Mode

(10)Current (Torque) Control Mode

(11)Feedforward Mode

2.2 Set trajectory generation type

Change the Position Control Mode and then set the trajectory generation type.

(Please skip to use the trajectory generation type as it is in normal mode.)

[Set trajectory generation type as Even and send command]

Write 0x01 value in 0x29 address with WRITE command.

Send Format

Tx	1	2	3	4	5	6	7	8
	SIZE 0x08	CMD 0x04	OP 0x00	ID 0x00	DATA 0x01	ADR 0x29	CNT 0x01	SUM 0x37

> Trajectory generation type (0x29)

(0)Normal: Rotates at maximum speed.

(1)Even: Evenly divides target time and interpolates from start point to end point (trajectory generation).

(3)ThirdPoly: Interpolates from start point to end point using third-order polynomial.

(4)ForthPoly: Interpolates from start point to end point with 4-1-4-order polynomial interpolation method.

(5)FifthPoly: Interpolates from start point to end point using fifth-order polynomial.

6.1.3 Change the gain based on the control mode.

Next, set the gain. At shipment, position control parameters are set in preset No. 0.

[Set gain to preset No. 0 and send command]

Write 0x00 value in 0x5C address with WRITE command.

Send Format

Tx	1	2	3	4	5	6	7	8
	SIZE	CMD	OP	ID	DATA	ADR	CNT	SUM
	0x08	0x04	0x00	0x00	0x00	0x5C	0x01	0x69

> PID gain preset number (Address:0x5C)

(0):Position Control Mode

(1):Speed Control Mode

(2):Current (Torque) Control Mode

6.1.4 Set servo status to Normal mode.

Next, when servo status is set to Normal mode, control begins and the servo begins to operate.

[Operating mode: Send Normal mode setting command]

Write 0x00 value to 0x28 address with WRITE command.

Send Format

Tx	1	2	3	4	5	6	7	8
	SIZE	CMD	OP	ID	DATA	ADR	CNT	SUM
	0x08	0x04	0x00	0x00	0x00	0x28	0x01	0x35

アドレス：0x28				Control		Operating	
BET7	BET6	BET5	BET4	BET3	BET2	BET1	BET0
0	0	0	0	0	0	0	0

> Control Mode

(00)Position Control Mode

(01)Speed Control Mode

(10)Current Control Mode

(11)Feedforward Mode

> Operating mode

(00)Normal

(10)Free

(11)Hold

6.1.5 Set target position.

Next, the servo position can be changed by changing the set points.

[Send command to set target position]

There are two ways to set the target position. One way is to write the target position to the address 0x2A / 0x2B with the WRITE command, or specify the target position and target movement time with the SET POSITION command.

Specify target position with WRITE command

Write 0x50 value to 0x2A address and 0x46 value to 0x2B address using the WRITE command.

Ex.) Set target position to 180 deg.: 18000 (0x4650)

Send Format

Tx	1	2	3	4	5	6	7	8	9
	SIZE	CMD	OP	ID	DATA1	DATA2	ADR	CNT	SUM
	0x09	0x04	0x00	0x00	0x50	0x46	0x2A	0x01	0xCE

Little-endian (2 bytes)
(Data sent from low-order byte)

Specify the target position and the target movement time with the SET POSITION command

Ex.) Set target position at 180 deg.: 18000 (0x4650)
Set target time at 3 sec.: 3000 (0x0BB8)

Send Format

Tx	1	2	3	4	5	6	7	8	9
	SIZE	CMD	OP	ID	DATA1	DATA2	DATA3	DATA4	SUM
	0x09	0x06	0x00	0x00	0x50	0x46	0xB8	0x0B	0x68

Target position
(2 bytes)
Little-endian

Target time
(2 bytes)
Little-endian

Reply Format (SET POSITION)

Rx	1	2	3	4	5	6	7
	SIZE	CMD	STTS	ID	DATA1	DATA2	SUM
	0x07	0x86	0x00	0x00	0xXX	0xXX	0xXX

*In the above case there is no error in servo operation.

Position data at command transmission is included in reply data.

*The arrival time is valid only when the orbit generation type (0x29) is set to "Even", "ThirdPoly", "ForthPoly", and "FifthPoly". It is not reflected in "Normal".

7. Memory Map

B3M has memory for processing data.

This memory is called the memory map.

The memory has a RAM area, where the data is erased when the power is turned off, and a ROM area, where the data is maintained even when the power turned off.

7.1 Unit Notation

This explains memory area units.

Unit	Content
bps	Unit of data transfer rate
deg	Angle unit (degrees)
℃	Temperature unit
mA	Currency unit
mV	Voltage unit
deg/sec	Angle moved in 1 second
ns	1/1,000,000,000 second (10–^{9} sec.)
ms	1/1,000 second
sec	Seconds unit
Hz	Frequency unit

7.2 Attribute Notation

This explains memory area attributes.

Attribute	Content
RW	(Read Write) Can read and write. Parameters that change system behavior when rewritten.
RO	(Read) Can only read. Parameters for overwriting data on system side.
#	Items with # mark in notes are written, and effective after restart. To make them effective, they must be saved in ROM.
x	When 100x, etc. is indicated, a value 100 times larger than the actual value is used. For example, to specify 90 deg. for the servo angle, 9000 is used.

7.3 Type Notation

This explains memory area types.

Type	Content
char	Signed char-type variable, 1 byte
byte	Unsigned char-type, 1 byte
short	Signed short in-type variable, 2 bytes (little-endian)*
ushort	Unsigned short in-type variable, 2 bytes (little-endian)*
long	Signed long in-type variable, 4 bytes (little-endian)*
ulong	Unsigned long in-type variable, 4 bytes (little-endian)*

*Little-endian is a format that segments numerical data with data volume of two bytes or more by each byte and records then sequentially started with the lowest order byte.

- Data signs are signed (char, short, long) and unsigned (byte, ushort, ulong).

Ex.) With 1-byte data

With data of 100

DATA (1)
100

→

DATA (1)
0x64

- With data of -100

DATA (1)
-100

→

DATA (1)
0x9C

Ex.) With two-byte data

With data of 1000

1000
0x03E8

→

DATA (1)	DATA (2)
0xE8	0x03

With data of -1000

-1000
0xFC18

→

DATA (1)	DATA (2)
0x18	0xFC

Ex.) With four-byte data

With data of 100000

100000
0x000186A0

→

DATA (1)	DATA (2)	DATA (3)	DATA (4)
0xA0	0x86	0x01	0x00

With data of -100000

-100000
0xFFFE7960

→

DATA (1)	DATA (2)	DATA (3)	DATA (4)
0x60	0x79	0xFE	0xFF

7.4 Memory Areas

・Memory is segmented into SYSTEM area, servo parameters area, control parameters area, STATUS area, version information area, and system initialization area.

[System information is stored]

SYSTEM Area						
Address	Bytes	Content	Unit	Type	Attribute	Note
0x00	1	ID number	−	byte	RW	*1
0x01	4	Baud rate, signal speed	bps	ulong	RW #	*2
0x05	2	Min. position limit	100x deg	short	RW	*3
0x07	2	Max. position limit	100x deg	short	RW	*3
0x09	2	Medium value offset	100x deg	short	RW	*3
0x0B	2	MCU temp. limit	100x℃	short	RW	
0x0D	1	PWM limit ratio at MCU temp. limit	%	byte	RW	
0x0E	2	Motor temp. limit	100x℃	short	RW	
0x10	1	PWM limit ratio at motor temp. limit	%	byte	RW	
0x11	2	Current limit	mA	ushort	RW	
0x13	1	PWM limit ratio at current limit	%	byte	RW	
0x14	1	Time until motor lock recognized	100 ms	byte	RW	
0x15	1	Percentage from max. output when lock recognized	%	byte	RW	
0x16	1	Degree of power reduction at lock detection	%	byte	RW	*4
0x17	2	Min. effective input voltage	mv	ushort	RW	*5
0x19	2	Max. effective input voltage	mv	ushort	RW	*5
0x1B	1	PWM limit	%	byte	RW	
0x1C	2	Dead bandwidth width	100x deg	ushort	RW	
0x1E	2	Reserved	−	short	−	
0x20	2	Reserved	−	short	−	
0x22	1	Motor rotation characteristics CW-direction ratio	−	byte	RW #	
0x23	1	Motor rotation characteristics CCW-direction ratio	−	byte	RW #	

[Servo control information is stored]

Servo Parameters Area						
Address	Bytes	Content	Unit	Type	Attribute	Note
0x27	1	Servo option	−	byte	RW #	*6
	Bit	Content	0		1	
	0	−				
	1					
	2					
	3					
	4					
	5					
	6	Clone (effective from Ver. 1.0.2.0)	Has reply data		No reply data	
	7	Reverse (effective from Ver. 1.0.2.0)	OFF		ON	
0x28	1	Normal/FREE/Hold mode switch	−	byte	RW	*6
	Bit	Content	0		1	
	0	Operation mode	00: Normal 10: Free 11: Hold			
	1					
	2	Control mode (*)	00: Position control, 01: Speed control 10: Torque (current) control 11: Feedforward			
	3					
	4	−				
	5	−				
	6	Servo (effective to Ver.1.0.1.0)	00: Normal 01: Clone 10: Reverse (11 also possible)			
	7					
0x29	1	Trajectory generation type	−	byte	RW	*6
	Bit	Content	0		1	
	0	Trajectory	0: Direct movement (no acceleration/deceleration interval) 1: Even segmentation (movement in equal intervals by time) 3: Third-order polynomial interpolation (acceleration interval generated with third-order polynomial and stopped after reaching max. speed) 4: 4-1-4-order polynomial interpolation (acceleration, uniform velocity, deceleration intervals generated in combination of fourth-order polynomial and first-order; uniform velocity interval is fixed) 5: Fifth-order polynomial interpolation (acceleration/deceleration intervals generated with fifth-order polynomial and stopped at velocity 0.)			
	1					
	2					
	3	−				
	4	−				
	5	−				
	6	−				
	7	−				

* With feedforward control, the PWM duty factor is input with the PWM count number (direction changes with sign) in the target position (address 0x2A). However, max. count number = system clock / PWM cycle, and even if it is set in excess of this value, it will be automatically revised to the upper limit.

Servo Parameters Area						
0x2A	2	Target position	100x deg	short	RW	
0x2C	2	Current position	100x deg	short	RO	
0x2E	2	Position one sampling prior	100x deg	short	RO	
0x30	2	Target speed	100x deg/sec	short	RW	*7
0x32	2	Current speed	100x deg/sec	short	RO	
0x34	2	Speed one sampling prior	100x deg/sec	short	RO	
0x36	2	Target time	ms	ushort	RW	
0x38	2	Command run time	ms	ushort	RO	*8
0x3A	2	Time elapsed from start-up	sec	ushort	RO	*8
0x3C	2	Target torque	mN.m	short	RW	*7
0x3E	4	System clock	20x ns	ulong	RO	
0x42	2	Sampling time	20x ns	ushort	RO	
0x44	2	Current MCU temp.	100x℃	short	RO	
0x46	2	Current motor temp.	100x℃	short	RO	
0x48	2	Current (load) current value	mA	short	RO	
0x4A	2	Current input voltage value	mV	ushort	RO	
0x4C	2	Current PWM duty factor		ushort	RW	
0x4E	2	PWM cycle	Hz	ushort	RW #	
0x50	2	Current value of encoder	－	ushort	RO	
0x52	4	Total count number of encoder	－	long	RW	
0x56	1	Status of Hall-IC	－	byte	RO	

[Gain information is stored]

Control Parameters Area						
Address	Bytes	Content	Unit	Type	Attribute	Note
0x5C	2	Change PID gain pattern	－	ushort	RW	
0x5C	1	PID gain preset number		byte		
0x5D	1	Reserved	－	byte		
0x5E	4	Proportional gain 0	100x	ulong	RW	
0x62	4	Differential gain 0	100x	ulong	RW	
0x66	4	Integral gain 0	100x	ulong	RW	
0x6A	2	Static friction coefficient 0	－	ushort	RW	
0x6C	2	Kinetic friction coefficient 0	－	ushort	RW	
0x6E	4	Proportional gain 1	100x	ulong	RW	
0x72	4	Differential gain 1	100x	ulong	RW	
0x76	4	Integral gain 1	100x	ulong	RW	
0x7A	2	Static friction coefficient 1	－	ushort	RW	
0x7C	2	Kinetic friction coefficient 1	－	ushort	RW	
0x7E	4	Proportional gain 2	100x	ulong	RW	
0x82	4	Differential gain 2	100x	ulong	RW	
0x86	4	Integral gain 2	100x	ulong	RW	
0x8A	2	Static friction coefficient 2	－	ushort	RW	
0x8C	2	Kinetic friction coefficient 2	－	ushort	RW	

[Status information is stored]

STATUS Area						
Address	Bytes	Content	Unit	Type	Attribute	Note
0x9D	1	Error status	－	byte		
0x9E	4	Error, etc. status	－	ulong		
0x9E	1	System error	－	byte		
0x9F	1	Motor status abnormality	－	byte		
0xA0	1	UART receive error	－	byte		
0xA1	1	Command error, etc.	－	byte		

[Version information is stored]

Version Information Area						
Address	Bytes	Content	Unit	Type	Attribute	Note
0xA2	4	Model segment	—	ulong	RO	*9
0xA2	1	Voltage segment	—	char	RO	
0xA3	1	Model version	—	byte	RO	*9
0xA4	1	Torque number	—	byte	RO	*9
0xA5	1	Case number	—	byte	RO	*9
0xA6	4	Model type	—	ulong	RO	*9
0xA6	1		—		RO	
0xA7	1		—		RO	
0xA8	1	Motor type	—	char	RO	*9
0xA9	1	Device type	—	char	RO	*9
0xAA	4	Firmware information	—	ulong	RO	
0xAA	1	Build number	—	byte	RO	
0xAB	1	Revision number	—	byte	RO	
0xAC	1	Minor version	—	byte	RO	
0xAD	1	Major version	—	byte	RO	

[Corrects difference between standard value and actual position]

System Initialization Area						
0xAE	2	Encoder error from absolute 00 position	—	short	RW #	
0xB0	72	Corrects encoder error at relative position	—	short*36	RW #	
0xB0	2	Corrects error in relative position no. 1	—	short	RW #	
⋮	⋮	⋮	⋮	⋮	⋮	
0xF6	2	Corrects error in relative position no. 36	—	short	RW #	

*The system initialization area (0xAE–0xF7) is set at factory shipment, so do not change it.

Notes

*1 When 255 (broadcast ID) is specified, the command is for all connected devices.

*2 115200/1000000/1250000/1500000/2000000/3000000 can be set.

When a value is set above the maximum or below the minimum, the value is revised to the maximum or minimum value.

*3 In reverse mode, it is set to the same position as normal mode and no reserve impact is received.

*4 It takes approx. 50 ms until the lock detection time starts.

*5 It is detected even when the current is insufficient and the voltage drops abruptly.

*6 Can only be accessed in 1-byte units. When writing, set all bits.

*7 Cannot output a value over specifications.

*8 The margin of error is max. ±0.5%.

*9 Refer to B3M Series specifications.

Overall Even if theoretical max. and min. values can be input in excess of specified values, do not input in excess of specified values.

8. Memory Map Details

This section explains individual memory address functions.

8.1 SYSTEM Area

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
ID number	0x00	byte	—	0	0	255	RW

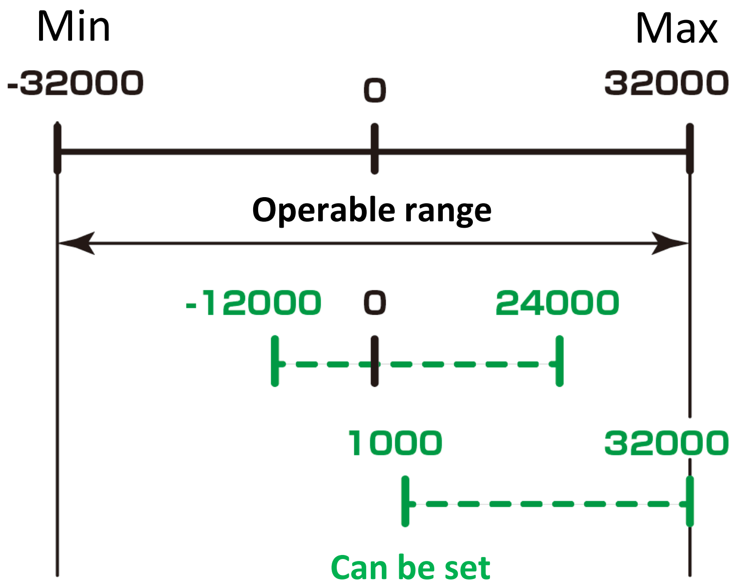
ID number of servo.
Ex.) ID number set to 12: 12 (0x0c)
*When 255 is specified, the command goes to all connected devices.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Baud rate	0x01	ulong	bps	1500000	115200	3000000	RW #

115200/1000000/1250000/1500000/2000000/3000000 are settable.
Ex.) Baud rate set to 1500000 bps: 1500000 (0x0016E360)
*When a value is set beyond the max. or min. value, it is revised to the max. or min. value.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Min. position limit	0x05	short	100×deg	-32000	-32000	32000	RW
Max. position limit	0x07	short	100×deg	32000	-32000	32000	RW

Max./min. values of movable range.
Ex.) Min. value set to -30000: -30000 (0x8AD0)
Ex.) Max. value set to 30000: 30000 (0x7530)
*Operations not guaranteed if max./min. values are invalid.
*In reverse mode, it is set to the same position as normal mode, and there is no reverse impact.



Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Medium value offset	0x09	short	100×deg	0	-18000	18000	RW

Center position (installation position offset)
Ex.) Offset set to 30 deg.: 3000 (0x0bb8)
*Operations not guaranteed if values invalid.
*In reverse mode, it is set to the same position as normal mode, and there is no reverse impact.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
MCU temp. limit	0x0B	short	100×C	8000	-32768	32767	RW

Temperature threshold that causes an error when MCU temperature rises too high.
Output torque is restricted when the specified temperature is exceeded.
If limit values (temperature limit, current limit, etc.) are exceeded and output is restricted, the servo will not operate normally.
Ex.) Limit set to 50 deg.: 5000 (0x1388)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
PWM limit ratio when MCU temp. restricted	0x0D	byte	%	0	0	100	RW

When an MCU temperature limit error occurs, motor power is made the specified percentage.
Ex.) Output torque set to 80% limit: 80 (0x50)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Motor temperature limit	0x0E	short	100×C	10000	-32768	32767	RW

Temperature threshold that triggers an error when motor temperature rises too high.
When the specified temperature is exceeded, output torque is restricted.
If limit values are exceeded and output is restricted, the servo will not operate normally.
Ex.) Limit is set to 50 deg.: 5000 (0x1388)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
PWM limit ratio when motor temp. restricted	0x10	byte	%	0	0	100	RW

When an MCU temperature limit error occurs, motor power is made the specified percentage.
Ex.) Output torque set to 80% limit: 80 (0x50)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Current limit	0x11	ushort	mA	*	0	65535	RW

Threshold that triggers an error when too much current flows to the motor.
When the specified current limit is exceeded, output torque is restricted at the specified value.
If limit values are exceeded and output is restricted, the servo will not operate normally.
Ex.) Current limit set to 6A: 6000 (0x1770)
*Differs depending on the product.

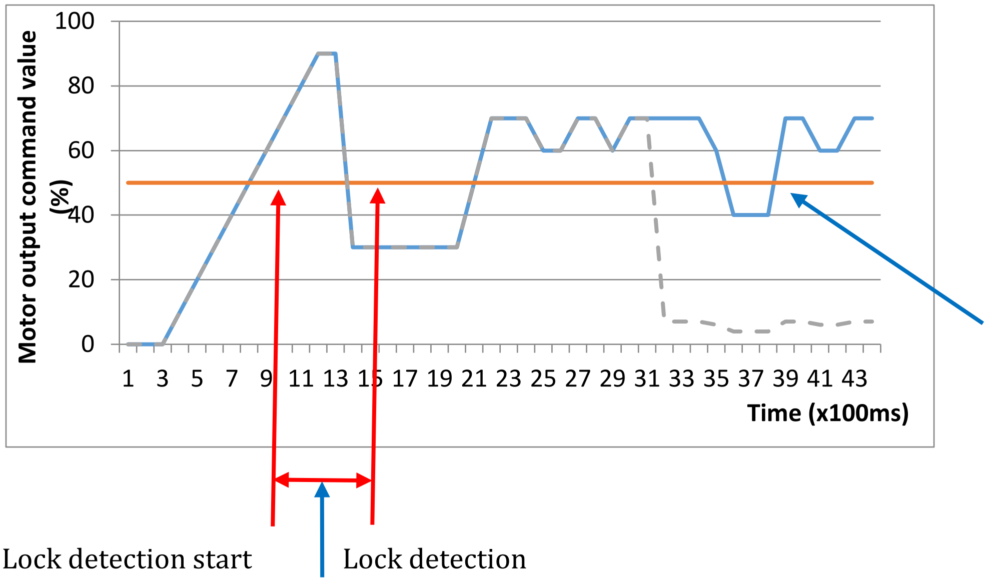
Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
PWM limit ratio when current restricted	0x13	byte	%	0	0	100	RW
When a current limit error occurs, motor power is made the specified percentage. Ex.) Output torque set to 80% limit: 80 (0x50)							

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Time until motor lock recognized	0x14	byte	100 ms	0	0	255	RW
Time from when motor lock is detected to when an error occurs. When the servo stops at a specified time, it is regarded as locked. Ex.) Time set to 1s: 10 (0x0A)							

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Percentage from max output when lock recognized	0x15	byte	%	50	0	100	RW
When a motor lock error occurs, motor power is made the specified percentage. Ex.) Output torque is set to 90% limit: 90 (0x5A)							

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Degree of power reduction when lock detected	0x16	byte	%	100	0	100	RW
A percentage to the maximum output, when this value continues for the time until motor lock is recognized, lock is detected. Ex.) Degree of power reduction set to 90%: 90 (0x5A) *When degree of power reduction is set to 100 when lock status is detected, the lock is not released. *It takes around 50ms until the lock detection timer starts.							

This is the timing chart for when the motor lock function does not function and when it does function.



Orange=Percentage of maximum output

Blue=Motor output command value

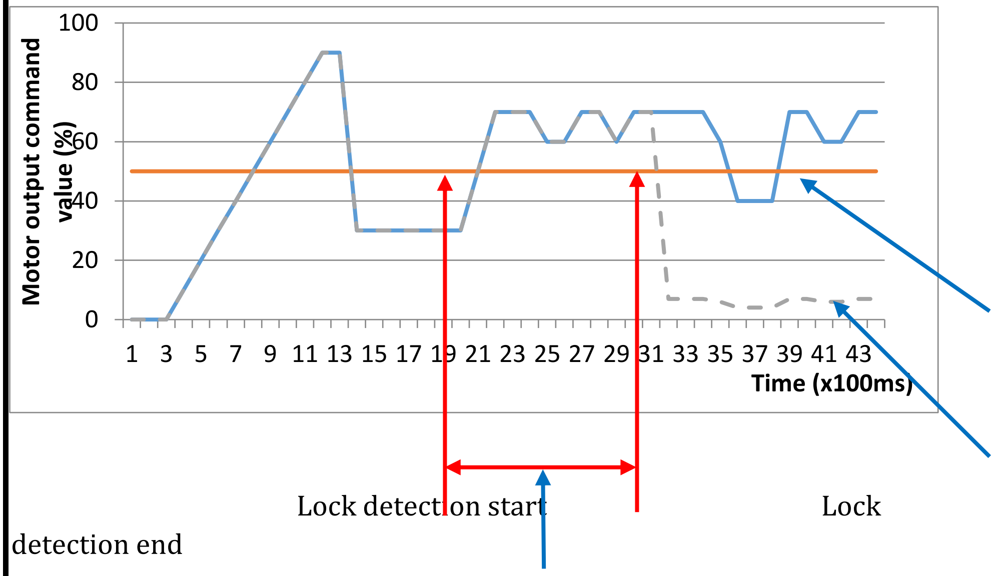
Gray=Output value after lock detected

Percentage from the maximum output recognized as lock (0x15)

Lock detection start

Lock detection

Lock function does not function because short time between start of lock detection and lock end (0x14)



Orange=Percentage of maximum output

Blue=Motor output command value

Gray=Output value after lock detected

Percentage from the maximum output recognized as lock (0x15)

Lock detection start

Lock

Ratio to remove torque at lock detection (0x16)

detection end

Time from lock detection start to error recognition exceeded (0x14)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Effective input voltage min.	0x17	ushort	mv	5500	0	65535	RW
Effective input voltage max.	0x19	ushort	mv	18000	0	65535	RW

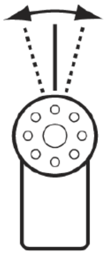
Motor completely de-powered when servo input voltage goes below/above the specified voltages.
Ex.) Min. value set to 5 V: 5000 (0x1388)
Ex.) Max. value set to 13 V: 13000 (0x32C8)
*This is detected even when current is insufficient and voltage drops abruptly.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
PWM limit	0x1B	byte	%	100	0	100	RW

Puts upper limit on PWM duty that drives motor.
Ex.) PWM limit set to 90%: 90 (0x5A)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Dead bandwidth width	0x1C	ushort	100×deg	15	0	65535	RW

Operating dead bandwidth (deviation, statistically determinate error).
Ex.) Set width to 5 deg.: 500 (0x01F4)
*If value set too small, there may be vibrating.



Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Motor rotation characteristics CW-direction ratio	0x22	byte	%	100	0	100	RW #

Upper limit of PWM duty of motor in clockwise direction.
Ex.) PWM value ratio set at 90%: 90 (0x5A).

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Motor rotation characteristics CCW-direction ratio	0x23	byte	%	100	0	100	RW #

Upper limit of PWM duty of motor in counter-clockwise direction.
Ex.) PWM value ratio set to 80%: 80 (0x50)

8.2 Servo Parameters Area

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Servo option	0x27	byte	–	0			RW #

BIT6	Clone
BIT7	Reverse

Operating mode

Reverse

・When the servo is installed, use this mode if you want it to operate in the opposite direction of the command value.

・Medium value offset (0x09-0x0A), maximum position limit (0x07-0x08), and minimum position limit (0x05-0x06) are not impacted by reverse and operate within the range specified when normal.

・When this mode is used, set the 7th bit to 1, save in ROM and re-start.

Clone

・If there is a servo you want to operate exactly the same, use the same ID and set it to this mode.

・When this mode is used, there is no reply data, so even if values are changed in Manager, etc., the values are changed on the servo side, but they may be recognized as errors on the Manager side.

・When using this mode, set the 6th bit to 1, save in ROM and re-start.

Control mode (control in endless rotation)

・B3M servo motor rotates endlessly except when in position control mode.

・B3M servo motor includes a magnetic encoder that can read 360 deg.

・In modes other than position control, when the current position is acquired, it is ±180 deg. of the starting position.

・When acquiring the cumulative angle, it can be calculated based on the current value (ushort) of the encoder at address 0x50 while in operation, or based on the total count number of the encoder at address 0x52.

・B3M servo motor's encoder is 12 bit, so 4096 values can be obtained per one revolution (360 deg.). The total count calculates the cumulative value of this, so it can count from -2^31 to 2^31-1.

・It is possible to also input any given value in the total count, so when resetting the count, enter 0. If using in a wheel, etc., sliding cannot be detected, so be aware of sliding when calculating distance.

*Effective from Ver.1.0.2.0.
*Only accessible in 1-byte units. When writing, set all bits.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Normal/FREE/Hold mode switch	0x28	byte	–	0x02			RW

BIT0	Operating mode
BIT1	
BIT2	Control mode
BIT3	
BIT6	Servo option
BIT7	

Operating mode
(00) Normal: Mode in which servo motor is active.
(10) Free: Mode in which servo motor is de-powered. Motor is not controlled.
(11) Hold: Mode in which servo motor maintains the current position. Motor is not controlled.

Control mode
(00) Position: Position control mode.
(01) Speed: Speed control mode.
(10) Current: Current control mode.
(11) Feed Forward: Feedforward mode. In feedforward, if the PWM count value is inputted in the target position, the motor rotates.
*Control mode: Motor does not operate unless operating mode is set to Normal.

Servo option (*Moved to 0x27 effective up to Ver. 1.0.1.0)
(00) Normal: Normal operating option
(01) Clone: No reply to any command. Used when simultaneously operating servos with the same ID.
(10) Reverse: Reverses operating direction. Effective only in position control mode.
*To switch to speed control mode, switch first to position control mode, set the current position and then switch to the mode.
*When switching the control mode, always move to FREE, HOLD and set initial values and then switch the mode.
*When switching modes, change the gain preset or adjust the gain.
*Can only be accessed in 1 byte units. When writing, set all bits.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Trajectory generation type	0x29	byte	–	0	0	255	RW

BIT0	Trajectory generation type
BIT1	
BIT2	

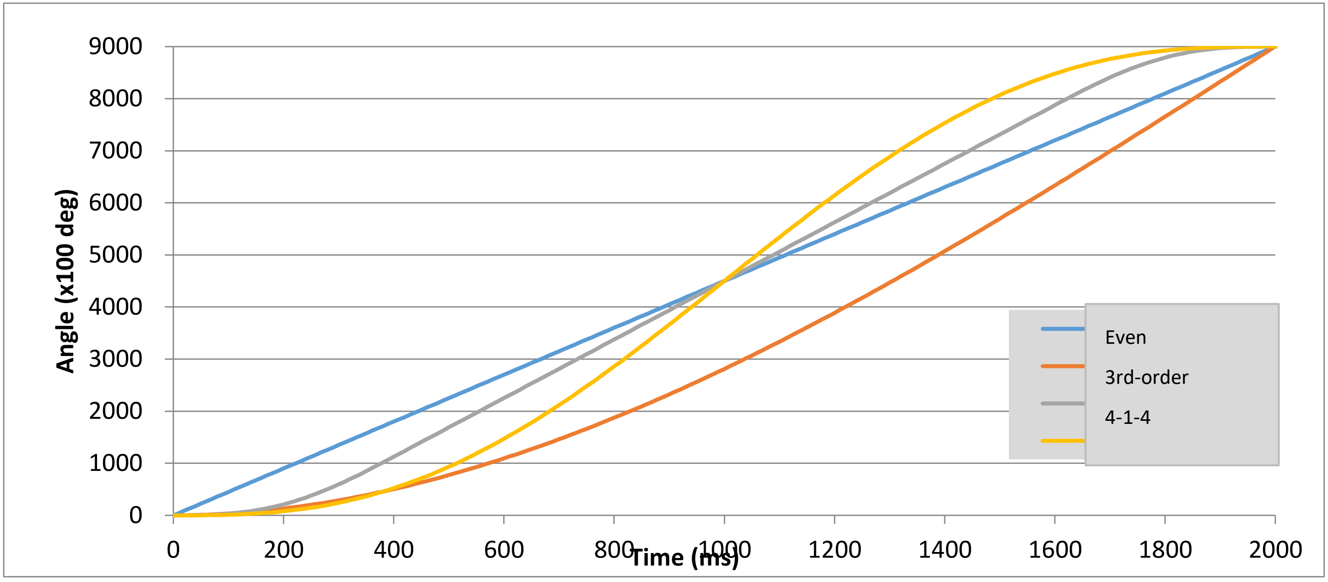
(0) Normal: Rotates at maximum speed.
(1) Even: Evenly divides target time and interpolates from start point to end point (trajectory generation).
(3) ThirdPoly: Interpolates from start point to end point using third-order polynomial.
(4) ForthPoly: Interpolates from start point to end point with 4-1-4-order polynomial interpolation method.
(5) FifthPoly: Interpolates from start point to end point using fifth-order polynomial.

*Only effective in position control mode when operation mode is Normal.
In Even, ThirdPoly, FourthPoly, FifthPoly modes, operations Normal when target time is set to 0.

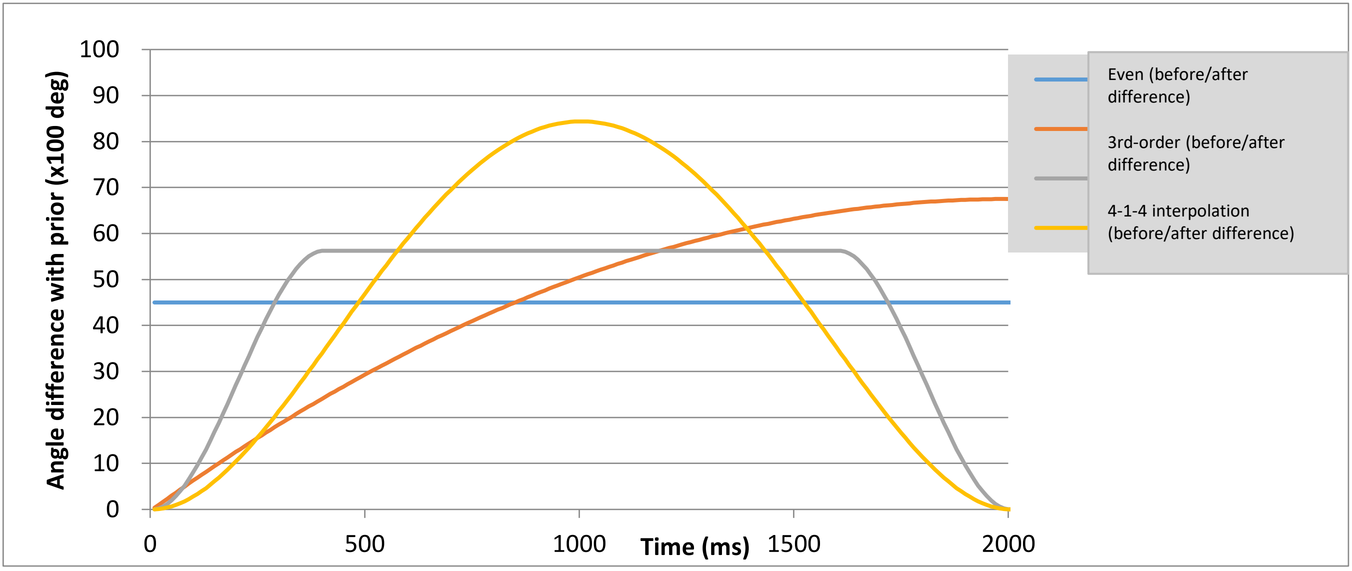
Servo trajectory graphed for different trajectory generation types.

Target time=2000 ms   Target position=90 deg

Graph 1: Servo movement trajectory over time



Graph 2: Servo angle difference over time



Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Target position	0x2A	short	100×deg	0	-32000	32000	RW

Writes the target position. Position control in position control mode.
PWM count value in feedforward mode.

Position control

・Control method in which input value (manipulated variable) is determined based on target position and current position. PID control is conducted, so operational behavior changes with P gain, D gain, and I gain described below.

・Can be controlled within range of ±320 deg. from starting point (value set with center value offset) (When switching to Free or Hold, it is restricted to current position ±180 deg.)

Ex.) Target position set at 180 deg.: 18000 (0x4650)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Current position	0x2C	short	100×deg	0	-32000	32000	RO

Shows current position when data read from servo.
Current position angle information can be known in units of 0.01 deg. in the range of -320 deg. in CCW (counter-clockwise) direction and 320 deg. in CW (clockwise) direction, with the center of the movable range at 0 deg.

Ex.) When current position at 90 deg.: 9000 (0x2328)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Previous sampling position	0x2E	short	100×deg	0	-32000	32000	RO

Shows value one time prior on servo control loop.

Ex.) When prior sampling servo position is 89 deg.: 8900 (0x22C4)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Target speed	0x30	short	100×deg/sec	0	-32768	32767	RW

Writes servo target speed. Effective only in speed control mode.

Speed control

・Control method in which input value (manipulated variable) is determined based on target speed and current speed. PI control is conducted.

Ex.) Target speed set to 30 deg/sec: 3000 (0x0BB8)

*Values exceeding specifications cannot be outputted.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Current speed	0x32	short	100×deg/sec	0	-32768	32767	RO

Current speed can be known in units of 0.01 deg/sec.

Ex.) When current speed is 40 deg/sec: 4000 (0x0FA0)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Prior sampling speed	0x34	short	100×deg/sec	0	-32768	32767	RO

Shows speed one time prior on servo control loop.
Ex.) When prior sampling servo speed is 39 deg/sec: 3900 (0x0F3C)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Target time	0x36	ushort	ms	0	0	65565	RW

Writes target movement time when in trajectory generation mode in position control mode.
Ex.) Target time set at 3 sec.: 3000 (0x0BB8)
*When 0 is specified, moves at maximum speed.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Command run time	0x38	ushort	ms	0	0	65565	RO

Shows current time in position control mode.
When SET POSITION command is issued, it is 0.
Ex.) When command run time is 2 sec.: 2000 (0x07D0)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Time elapsed from start-up	0x3A	ushort	sec	0	0	65535	RO

Shows control time elapsed.
Ex.) When elapsed time is 30 sec.: 30 (0x001e)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Target torque	0x3C	short	mN.m	0	-32768	32767	RW

Current (torque) control

・Control method that calculates current value from target torque and feeds back current value. PI control is conducted.
Ex.) Target torque set to 1000 mN.m: 1000 (0x03E8)
*Values exceeding specifications cannot be outputted.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
System clock	0x3E	ulong	20×ns	0	0	10000000	RO

Reads system (MCU) clock value.
Ex.) When clock value is 12345678: 12345678 (0x00BC614E)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Sampling time	0x42	ushort	20×ns	0	0	10000000	RO

Shows current loop sampling time.
Ex.) When data processing time is 30 ms: 1500 (0x05DC)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Current MCU temp.	0x44	short	100×℃	0	-5500	12800	RO

Shows MCU temp.
Ex.) When current MCU temp. is 40 deg.: 4000 (0x0FA0)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Current motor temp.	0x46	short	100×℃	0	-12800	12800	RO

Shows motor temp.
Ex.) When current motor temp. is 60 deg.: 6000 (0x1770)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Current (load) current value	0x48	short	mA	0	0	32767	RO

Shows motor current value.
When CW (clockwise), and CCW (counter-clockwise) 0–32767
(both way same value)
Ex.) When current value is 300 mA: 300 (0x012C)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Current input voltage value	0x4A	ushort	mV	0	0	65535	RO

Shows input voltage value.
Ex.) When input is 5.5 V: 5500 (0x157C)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Current PWM duty ratio	0x4C	ushort	–	0	0	65535	RO

Shows PWM duty value.
The value is the count number with respect to PWM frequency; meaning it is not the duty ratio.
Ex.) When the count number is 2000: 2000 (0x07D0)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
PWM frequency	0x4E	ushort	Hz	*	0	65535	RW #

PWM frequency. The duty ratio can be calculated as the max. counter value.
PWM frequency
　Max. PWM count value=system clock / PWM frequency (1)
　PWM signal duty ratio = specified PWM count value / max. PWM count value
　Max. PWM count value depends on formula (1).
　When the PWM frequency count value increases, the maximum PWM count per 1 cycle decreases, so the duty ratio can no longer be specified narrowly.
Ex.) PWM frequency set to 8 KHz: 8000 (0x1F40)
*Differs depending on the product.
*When rewritten, the PWM frequency changes.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Current encoder value	0x50	ushort	–	0	0	4095	RO

Encoder value installed on the servo.
Data before data correction to acquire current position.
Ex.) When current value is 2048: 2048 (0x0800)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Total count number of encoder	0x52	long	–	0	-2^31	2^31-1	RW

Cumulative count number of encoder.
Can measure from -2147483648 to 2147483647 (-524288–524287) rotations.
When initializing, make this parameter 0.
In reverse mode, it is set the same as normal mode, and reverse has no impact.
Ex.) When total count number is 123456: 123456 (0x0001E240)

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Hall-IC status	0x56	byte	–	0	0	7	RO

Expresses input status of brushless motor's Hall-IC (ON=1, OFF=0) in 3 bits.

8.3 Control Parameters Area

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
PID gain preset number	0x5C	byte	–	0	0	2	RW
Reserved	0x5D	byte					

Each PID gain opened from pre-specified presets.

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Proportional gain 0	0x5E	ulong	100x	*	0	2^32-1	RW
Differential gain 0	0x62	ulong	100x	*	0	2^32-1	RW
Integral gain 0	0x66	ulong	100x	*	0	2^32-1	RW
Static friction coefficient 0	0x6A	ushort	–	*	0	2^16-1	RW
Kinetic friction coefficient 0	0x6C	ushort	–	*	0	2^16-1	RW
Proportional gain 1	0x6E	ulong	100x	*	0	2^32-1	RW
Differential gain 1	0x72	ulong	100x	*	0	2^32-1	RW
Integral gain 1	0x76	ulong	100x	*	0	2^32-1	RW
Static friction coefficient 1	0x7A	ushort	–	*	0	2^16-1	RW
Kinetic friction coefficient 2	0x7C	ushort	–	*	0	2^16-1	RW
Proportional gain 2	0x7E	ulong	100x	*	0	2^32-1	RW
Differential gain 2	0x82	ulong	100x	*	0	2^32-1	RW
Integral gain 2	0x86	ulong	100x	*	0	2^32-1	RW
Static friction coefficient 2	0x8A	ushort	–	*	0	2^16-1	RW
Kinetic friction coefficient 2	0x8C	ushort	–	*	0	2^16-1	RW

About gains

With the B3M servo motor, when gain values are changed, there are immediately incorporated into control.
*Differs depending on the product.

P gain (Kp0–Kp2)

・P gain is proportional gain; it is a scale factor (gain) that changes the input value (manipulated variable) proportionally with respect to the target position and current position's position error (variation).

・When P gain is reduced, the input value is proportionally reduced, and the motor's power and speed both decrease. When it is increased, operations become oscillatory.

D gain (Kd0–Kd2)

・D gain is differential gain; it is a scale factor (gain) that changes the manipulated variable proportionally with respect to the current speed and target speed (with position control, the target speed is 0).

・D gain contributes to the incline of position change, which makes it easy to keep the response from becoming oscillatory and facilitate convergence.

I gain (Ki0–Ki2)

・I gain is integral gain; it is a scale factor (gain) that changes the manipulated variable proportionally with respect to the cumulative value (integral value) of the variation between the current position and target position (in position control, the target speed is 0).

・When I gain is increased, position error is decreased, so the motor's retention force increases. However, the reaction when the response begins will generally slow, so that the error will be 0 or the integral value small,

<u>Gain presets</u>

・Gains are set at shipment in presets 0, 1 and 2 in line with position control, speed control and torque control.

・When gain preset numbers are changed, gains are immediately changed and incorporated into servo motor operations. Gains differs with position control, speed control and torque control, so when changing the control mode, always change the gain or gain preset.

・Gains for position control are entered in all the presets and can also be switched and used.

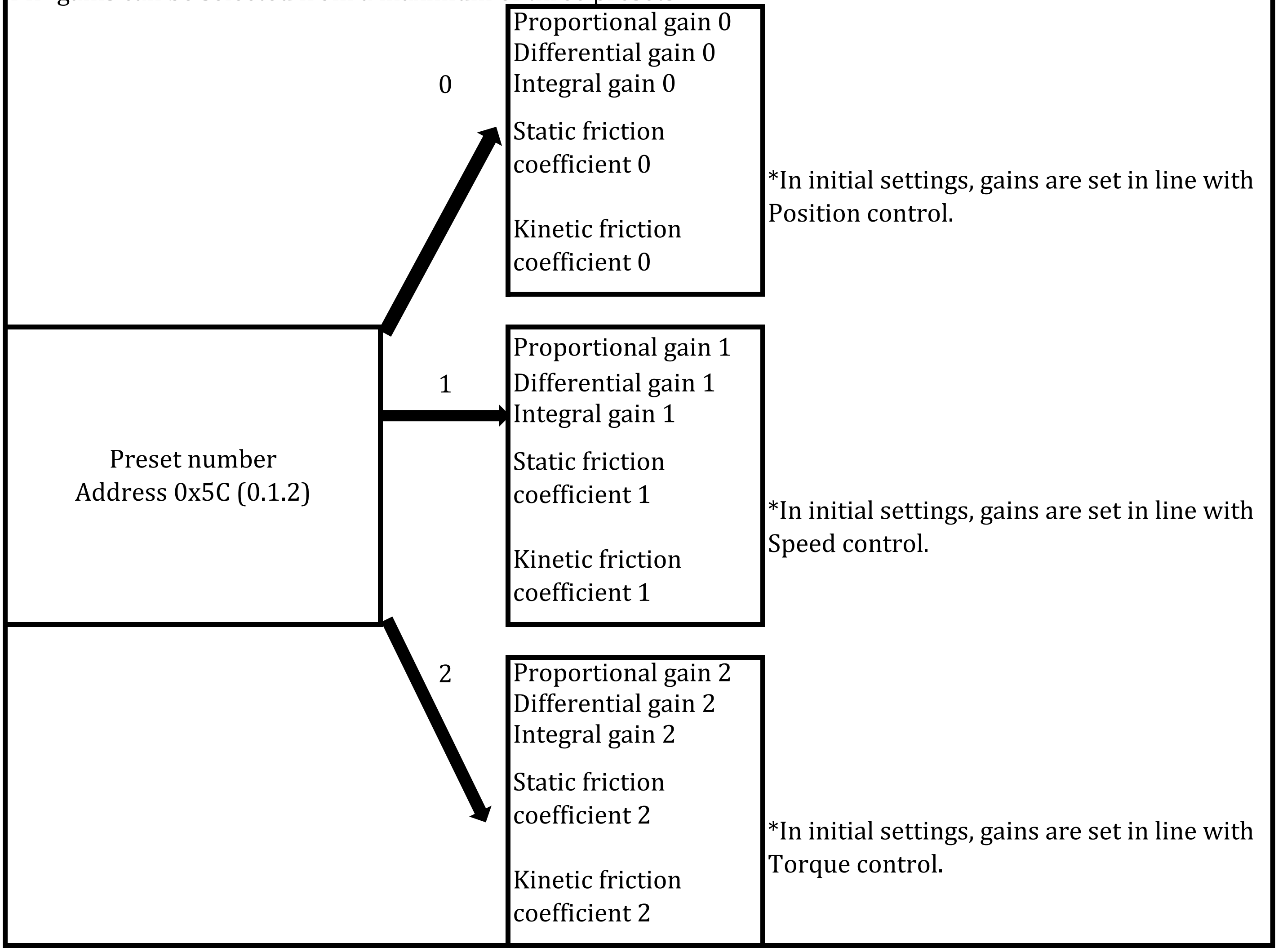
<u>Friction coefficient</u>

・The friction coefficient compensates for motor and gear friction; it is a dimensionless quantity.

・When moving from a static state, a static friction coefficient is added to the manipulated variable. Also, when moving, a kinetic friction coefficient is applied. The static friction coefficient and kinetic friction coefficient switches automatically based on the motor speed.

In initial settings, gain preset No. 0 is allocated to position control, gain preset No. 1 to speed control, and gain preset No. 2 to torque control.

PID gains can be selected from a maximum of three presets.



*In initial settings, gains are set in line with Position control.

*In initial settings, gains are set in line with Speed control.

*In initial settings, gains are set in line with Torque control.

8.4 STATUS Parameters Area

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Error status	0x9D	byte	–	0			
System error	0x9E	byte	–	0			
Motor status abnormality	0x9F	byte	–	0			
UART receive error	0xA0	byte	–	0			
Command error, etc.	0xA1	byte	–	0			

8.5 Version Information Area

Name (Explanation)	Symbol	Content
Voltage segment	A	6–12 [V]
	B	12–18 [V]
	C	18–24 [V]
Torque number	4	40 [kgf.cm]
	7	70 [kgf.cm]
Case number	10	Case height is low
	11	Case height is high
Motor type	B	Brushless motor
	C	Coreless motor
	D	Cored motor
Device type	S	Servo motor

Model segment

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Voltage segment	0xA2	char	–	*	0x00	0xFF	RO
Model version	0xA3	byte	–	*	0x00	0xFF	RO
Torque number	0xA4	byte	–	*	0x00	0xFF	RO
Case number	0xA5	byte	–	*	0x00	0xFF	RO

*Differs depending on the product.

Model type

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
	0xA6	byte	–	*	0x00	0xFF	RO
	0xA7	byte	–	*	0x00	0xFF	RO
Motor type	0xA8	char	–	*	0x00	0xFF	RO
Device type	0xA9	char	–	*	0x00	0xFF	RO

*Differs depending on the product.

Firmware information

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Build number	0xAA	byte	–	*	0x00	0xFF	RO
Revision number	0xAB	byte	–	*	0x00	0xFF	RO
Minor version	0xAC	byte	–	*	0x00	0xFF	RO
Major version	0xAD	byte	–	*	0x00	0xFF	RO
*Differs depending on the product.							

8.6 System Initialization Area

Encoder error correction at relative position

Name (Explanation)	Address	Form	Unit	Initial Value	Min. Value	Max. Value	Attribute
Encoder error from absolute 00 position	0xAE	short	–	*	-32768	32767	RW #
Error correction at relative position no. 01	0xB0	short	–	*	-32768	32767	RW #
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Error correction at relative position no. 36	0xF6	short	–	*	-32768	32767	RW #
Corrects difference between standard value and actual position. *Already set at factory shipment, so do not change.							

9. B3M Series PC Library

1. B3M servo motor's NET library for PC.
2. B3MLib: Two DLL files for creating commands for B3M servo motor: B3MLib.dll and Extensions.dll.

9.1 Usage Environment

1. Windows XP (SP2 or later)/Windows Vista/Windows 7 / 8 / 8.1 / 10, 32 bit/64 bit versions.
2. B3MLib.DLL and Extensions.DLL (B3M library set)
3. .NET Framework 2.0 and higher
4. With Windows XP Service Pack 2 and later, Windows Vista, Windows 7, installed in OS.
5. Visual Studio 2005 and later, or Visual Basic Express/Visual C# Express/Visual C++ Express 2005 or later version is necessary (this manual uses Visual Basic Express 2010).

   Available here:
   http://kondo-robot.com/faq/b3m-pclibset

9.2 Memory Map

SYSTEM Area				
Address	Content	Type	Address Number Definition Name	Property Name
0x00	ID number	byte	SYSTEM_ID	ID
0x01	Baud rate, signal speed	ulong	SYSTEM_baud rate	baud rate
0x05	Min. position limit	short	SYSTEM_POSITION_MIN	PositionMinLimit
0x07	Max. position limit	short	SYSTEM_POSITION_MAX	PositionMaxLimit
0x09	Medium value offset	short	SYSTEM_POSITION_CENTER	PositionCenterOffset
0x0B	MCU temp. limit	short	SYSTEM_MCU_TEMP_LIMIT	MCUTempLimit
0x0D	PWM limit ratio at MCU temp. limit	byte	SYSTEM_MCU_TEMP_LIMIT_PR	MCUTempPowerLimit
0x0E	Motor temp. limit	short	SYSTEM_MOTOR_TEMP_LIMIT	MotorTempLimit
0x10	PWM limit ratio at motor temp. limit	byte	SYSTEM_MOTOR_TEMP_LIMIT_PR	MotorTempPowerLimit
0x11	Current limit	ushort	SYSTEM_CURRENT_LIMIT	CurrentLimit
0x13	PWM limit ratio at current limit	byte	SYSTEM_CURRENT_LIMIT_PR	CurrentPowerLimit
0x14	Time until motor lock recognized	byte	SYSTEM_LOCKDETECT_TIME	LockDetectTime
0x15	Percentage from max. output when lock recognized	byte	SYSTEM_LOCKDETECT_OUTRATE	LockDetectOutputRate
0x16	Degree of power reduction at lock detection	byte	SYSTEM_LOCKDETECT_TIME_PR	LockDetectTimePowerLimit
0x17	Min. effective input voltage	ushort	SYSTEM_INPUT_VOLTAGE_MIN	InputVoltageMin
0x19	Max. effective input voltage	ushort	SYSTEM_INPUT_VOLTAGE_MAX	InputVoltageMax
0x1B	PWM limit	byte	SYSTEM_TORQUE_LIMIT	TorqueLimit
0x1C	Dead bandwidth width	ushort	SYSTEM_DEADBAND_WIDTH	DeadBandWidth
0x22	Motor rotation characteristics CW-direction ratio	byte	SYSTEM_MOTOR_CW_RATIO	MotorCWRatio
0x23	Motor rotation characteristics CCW-direction ratio	byte	SYSTEM_MOTOR_CCW_RATIO	MotorCCWRatio

Servo Parameters Area				
Address	Content	Type	Address Number Definition Name	Property Name
0x27	Servo option (*)	byte	SERVO_SERVO_OPTION	ServoOption
0x28	Operation mode	ushort	SERVO_SERVO_MODE	ServoMode
0x28	Normal/FREE/Hold mode switch	byte	SERVO_TORQUE_ON	TorqueON
0x29	Trajectory generation type	byte	SERVO_RUN_MODE	RunMode
0x2A	Target position	short	SERVO_DESIRED_POSITION	DesiredPosition
0x2C	Current position	short	SERVO_CURRENT_POSITION	CurrentPosition
0x2E	Position one sampling prior	short	SERVO_PREVIOUS_POSITION	PreviousPosition
0x30	Target speed	short	SERVO_DESIRED_VEROCITY	DesiredVelosity
0x32	Current speed	short	SERVO_CURRENT_VELOCITY	CurrentVelosity
0x34	Speed one sampling prior	short	SERVO_PREVIOUS_VELOCITY	PreviousVelosity
0x36	Target time	ushort	SERVO_DESIRED_TIME	DesiredTime
0x38	Command run time	ushort	SERVO_RUNNING_TIME	RunningTime
0x3A	Time elapsed from start-up	ushort	SERVO_WORKING_TIME	WorkingTime
0x3C	Target torque	short	SERVO_DESIRED_TORQUE	DesiredTorque
0x3E	System clock	ulong	SERVO_SYSTEM_CLOCK	SystemClock
0x42	Sampling time	ushort	SERVO_SAMPLING_TIME	SamplingTime
0x44	Current MCU temp.	short	SERVO_MCU_TEMP	MCUTemperature
0x46	Current motor temp.	short	SERVO_MOTOR_TEMP	MotorTemperature
0x48	Current (load) current value	short	SERVO_CURRENT	Current
0x4A	Current input voltage value	ushort	SERVO_INPUT_VOLTAGE	InputVoltage
0x4C	Current PWM duty factor	ushort	SERVO_PWM_DUTY	PwmDuty
0x4E	PWM cycle	ushort	SERVO_PWM_FREQUENCY	PwmFrequency
0x50	Current value of encoder	ushort	SERVO_ENCODER_VALUE	EncoderValue
0x52	Total count number of encoder	long	SERVO_ENCODER_COUNT	EncoderCount
0x56	Status of Hall-IC	byte	SERVO_HALLIC_STATE	HallICState

Control Parameters Area				
Address	Content	Type	Address Number Definition Name	Property Name
0x5C	Change PID gain pattern	ushort	CONTROL_CONTROL_LOW	ControlRow
0x5C	PID gain preset number	byte	CONTROL_GAIN_PRESETNO	GainPresetNo
0x5D	Reserved	byte	CONTROL_TYPE	ControlType
0x5E	Proportional gain 0	ulong	CONTROL_KP0	Kp0
0x62	Differential gain 0	ulong	CONTROL_KD0	Kd0
0x66	Integral gain 0	ulong	CONTROL_KI0	Ki0
0x6A	Static friction coefficient 0	ushort	CONTROL_STATIC_FRICTION0	StaticFriction0
0x6C	Kinetic friction coefficient 0	ushort	CONTROL_DYNAMIC_FRICTION0	DynamicFriction0
0x6E	Proportional gain 1	ulong	CONTROL_KP1	Kp1
0x72	Differential gain 1	ulong	CONTROL_KD1	Kd1
0x76	Integral gain 1	ulong	CONTROL_KI1	Ki1
0x7A	Static friction coefficient 1	ushort	CONTROL_STATIC_FRICTION1	StaticFriction1
0x7C	Kinetic friction coefficient 1	ushort	CONTROL_DYNAMIC_FRICTION1	DynamicFriction1
0x7E	Proportional gain 2	ulong	CONTROL_KP2	Kp2
0x82	Differential gain 2	ulong	CONTROL_KD2	Kd2
0x86	Integral gain 2	ulong	CONTROL_KI2	Ki2
0x8A	Static friction coefficient 2	ushort	CONTROL_STATIC_FRICTION2	StaticFriction2
0x8C	Kinetic friction coefficient 2	ushort	CONTROL_DYNAMIC_FRICTION2	DynamicFriction2

STATUS Area				
Address	Content	Type	Address Number Definition Name	Property Name
0x9D	Error status	byte	STATUS_BASE_ADDR	StatusError
0x9E	Error, etc. status	ulong	STATUS_SYSTEM	Status
0x9E	System error	byte	STATUS_SYSTEM	StatusSystem
0x9F	Motor status abnormality	byte	STATUS_MOTOR	StatusMotor
0xA0	UART receive error	byte	STATUS_UART	StatusUart
0xA1	Command error, etc.	byte	STATUS_COMMAND	StatusCommand

System Initialization Area / Version Information Area				
Address	Content	Type	Address Number Definition Name	Property Name
0xA2	Model segment	ulong	CONFIG_MODEL_NUMBER	ModelNumber
0xA2	Voltage segment	char	CONFIG_MODEL_NUMBER_VOLTAGE_CLASS	ModelNumberVoltageClass
0xA3	Model version	byte	CONFIG_MODEL_NUMBER_VERSION	ModelNumberVersion
0xA4	Torque number	byte	CONFIG_MODEL_NUMBER_TORQUE	ModelNumberTorque
0xA5	Case number	byte	CONFIG_MODEL_NUMBER_CASE	ModelNumberCase
0xA6	Model type	ulong	CONFIG_MODEL_TYPE	ModelType
0xA6				
0xA7				
0xA8	Motor type	char	CONFIG_MODEL_TYPE_MOTOR	ModelTypeMotor
0xA9	Device type	char	CONFIG_MODEL_TYPE_DEVICE	ModelTypeDevice
0xAA	Firmware information	ulong	CONFIG_FW_VERSION	FwVersion
0xAA	Build number	byte	CONFIG_FW_BUID	FwVersionBuild
0xAB	Revision number	byte	CONFIG_FW_REVISION	FwVersionRevision
0xAC	Minor version	byte	CONFIG_FW_MINOR	FwVersionMinor
0xAD	Major version	byte	CONFIG_FW_MAJOR	FwVersionMajor
0xAE	Error from absolute 00 position	short	CONFIG_ENC_OFFSET_CENTER	EncoderOffsetCenter
0xB0	Error at relative 01 position	short	CONFIG_ENC_OFFSET	EncoderOffset

* Effective from Ver. 1.0.2.0

10. List of Enumerated Type Variables (bitwise Option)

Address (0x27.0x28)	Operation mode		
Category	Content	Enum name	Value
Options	Operation mode normal	RunNormal	0x00
	Free mode	RunFree	0x02
	Hold mode	RunHold	0x03
	Position control mode	ControlPosition	0x00
	Speed control mode	ControlVelosity	0x04
	Torque control mode	ControlTorque	0x08
	Feedforward control mode	ControlFForward	0x0C
	Normal	ServoNormal	0x00
	Clone mode	ServoClone	0x40
	Reverse mode	ServoReverse	0x80

Address (0x9E)	System error		
Category	Content	Enum name	Value
SystemErrors	Whether Watchdog timer restarted	Watchdog	0x01
	Flash memory access error	FlashAccess	0x02
	Flash memory initialization error	MemoryAllocation	0x04
	Input voltage error	InputVoltage	0x08
	MCU temp. outside range	MCUTemperature	0x10
	AD conversion failed	ADConversion	0x20
	I2C access failed	I2C	0x40
	SPI access failed	SPI	0x80

Address (0x9F)	Motor status abnormality		
Category	Content	Enum name	Value
MotorErrors	Motor temp. outside range	ExceedMotorTemperature	0x01
	Motor lock detected	LockDetect	0x02
	Motor current outside range	ExceedCurrentLimit	0x04
	Hall-IC error detected	HallIC	0x08

Address (0xA0)	UART receive error		
Category	Content	Enum name	Value
UARTErrors	START/STOP bit invalid	Framing	0x01
	Parity check failed	Parity	0x02
	Data not finished	Break	0x04
	Data bit overrun	Overrun	0x08

Address (0xA1)	Command error, etc.		
Category	Content	Enum name	Value
CommandErrors	Checksum error	CheckSum	0x01
	Error in specified data length	Length	0x02
	Specified data length is different	Size	0x04
	Address overflow	Address	0x08
	Command error	WrongCommand	0x10

Address (0x29)	Trajectory generation type		
Category	Content	Enum name	Value
Trajectory	Linear movement, no acceleration/deceleration control	Normal	0x00
	Even segmentation (movement in even intervals over time)	Even	0x01
	Third-order polynomial interpolation	ThirdPoly	0x03
	4-1-4-order polynomial interpolation	FourthPoly	0x04
	Fifth-order polynomial interpolation	FifthPoly	0x05

	Command		
Category	Content	Enum name	Value
CommandTypes	Data saved in flash memory loaded into RAM	Load	0x01
	RAM data saved in flash memory	Save	0x02
	RAM data read	Read	0x03
	Data written on RAM	Write	0x04
	Position specification command	Position	0x05
	Reset command	Reset	0x06

Precautions

When returning bitwise option to byte variable, cast as (byte) category.Enum name. For example, when retrieving Options RunFree in Byte variable, it is as follows.

```
byte options = (byte) Options.RunFree;
```

11. Change Log

Version	Date	Change
Ver. 1.0.0.0		β release
Ver. 1.0.1.0	2012/12/15	Switched position command and reset command data
		Revised some typographical errors in Status address
Ver. 1.1.0.0	2013/07/18	Revised typographical errors in error status
		Revised typographical errors in LockDetectTime
		Revised servo option (clone, reverse) address. Applied from B3M version 1.0.2.0. Changed to make effective after restart.
		Servo option bitwise option error revised
		Internal calculation routine revised
		Other manual errors revised
Ver. 1.1.3.0	2014/04/07	Except for position control, current position revised so that it is ±180 deg.
		Revised so that current position is ±180 deg. in Free or Hold status.
		Changed so that current position entered as target position so that motor does not move abruptly in status other than position control.
		Changed so not dependent on voltage during torque control.
		Added area on operations to manual.
		Corrected the following bugs.
		・Command run time (0x38) not incorporated
		・Lock detection not incorporated normally
		・Encoder count not 0 immediately after start-up
		・Encoder difference (0xAE–0xF6) incorporated even if changed during start-up
		・Does not function normally when multimode command sent with Save, Load commands.
		Other manual typographical errors revised.
Ver. 1.2.0.0	2015/11/27	Manual revised for readability
Ver. 1.2.0.2	2018/8/24	Information added.
Ver.1.2.0.3	2020/2/25	Information correction