1M23Z02002

User's Manual Ver.1.12

# RS303MR/RS304MD

Command Type Servo for Robot

## Instruction Manual


**Caution**

- Read this instruction manual before use.
- Keep this manual handy for immediate reference.

**For models**

Futaba®

# 1. FOR SAFETY

To use this product safely, please pay your full attention to the followings.
Be sure to read this instruction manual prior to using this product.

## Warning Symbols

The warning symbols used in this text are defined as follows:

| Indication | Meanings |
|---|---|
| ⚠Danger | Indicates a hazard that will cause severe personal injury, death, or substantial property damage if the warning is ignored. |
| ⚠Warning | Indicates a hazard that can cause severe personal injury, death, or substantial property damage if the warning is ignored. |
| ⚠Caution | Indicates a hazard that will or can cause minor personal injury, or property damage if the warning is ignored |

**Symbols:** ⃠ :**Prohibited** ❗ : **Mandatory**

## Cautions for Use

### ⚠ Caution

⃠ Do not disassemble or alter the servo.
Otherwise, it may cause breakage in the gear box, fire on the servo or explosion of the battery.

⃠ Do not use any battery as a power supply other than the specified battery.
The product is designed to be operated by Futaba's 9.6V Ni-MH battery only. Do not use any other batteries.

⃠ Do not touch the servo case during or some time after operating the servo.
Otherwise, you may get burned on the finger as the motor or electronic circuit in the servo gets very hot.

⃠ Do not let the servo get covered with sandy dust or water.
Otherwise, the servo may stop moving or have a short circuit. The servo is not designed to be waterproof or dust-proof.

Do not use this product for any application other than indoor hobby-robots.
Futaba is cleared of all responsibility to the results caused by the usage of this product for any application other than indoor hobby-robots.

Do not turn the servo horn forcibly.
Otherwise, the servo will be damaged.

Do not leave the servo locked.
If the servo continues to be locked due to a strong external force, it may cause smoke, fire or damage.

## Warnings in Handling Batteries

### ⚠ Warning

Do not use any battery charger other than the specified charger.
Otherwise, the battery may be get damaged, fire, smoke or liquid leakage. Be sure to use batteries recommended by Futaba.

Do not use battery packs connected in parallel.
Connecting battery packs in parallel may cause abnormal heat generation or explosion due to the differences between charging voltages.

Do not disassemble or alter battery packs.
Otherwise, it may cause fire, explosion or liquid leakage. And please be aware that such battery packs will be no longer guaranteed even if their warranties are not expired.

Do not use batteries if any abnormal symptoms are seen.
If you find any abnormal symptoms such as cracks in coating film, abnormal heat generation from batteries or deformation of batteries, never use the battery because it may cause serious consequences.

## Cautions for Storage

### ⚠ Caution

Do not store the servos in the following conditions.
Places where the temperature is over 60℃ or below －20℃.
Places where the Sun directly shines over the servos.
Places where it is very high in humidity.
Places where there is a strong vibration.
Places where there is a lot of dust.
Places where static electricity tends to be induced.
Places where infants can reach.

◆ Storing the servos in the places shown above may cause deformation and failure of the servos, or hazard.

# 2. Introduction

## Components

The following parts are included in RS303MR and RS304MD.

- 1) Servo(RS303MR or RS304MD) x1
  2) Servo Horn x1
  3) Screw to fix servo horn x1 (RS303MR：M2x6)
  (RS304MD：M2x8 Tapping Screw)

*Servo Horn and Screw are attached to the servo.

- 4) Free Horn for RS30x x1
- 5) USAGE PRECAUTION x1

---

Please refer to p.43 “Optional parts” for optional parts and repair parts.

# Features

RS303MR/RS304MD is especially designed for robotic applications, having the following features.

## ● Small and Lightweight

RS303MR/RS304MD is small and lightweight servos specially designed for robots, which are 21g and 28g respectively.

## ● Interactive High-Speed TTL Communication

TTL bi-directional communication of a maximum baud rate of 230kbps is possible and is used for the communication between the robot's processor and servos.

## ● Servos for Both Command Type and PWM Type

RS303MR/RS304MD can be controlled either as the Command Type Servo for robots or the PWM Type Servo for existing R/C servos. The type is automatically switched by the first signal sent after these products are connected.

## ● Feedback(Only as Command Type)

Information of sensors on the servo such as the position, load, temperature, electric current, and the alarm situation, etc. can be obtained via TTL communication.

## ● Compliance Control

With this feature, the movement of the RS303MR/RS304MD is controlled in accordance with the distance between the present and target positions. This enables the robot to move very smoothly without trembling its arms and legs and to absorb turbulence from external force.

## ● Calibration

Angular position of every servo is calibrated by our standard gauge before their shipment. Even if another RS303MR/RS304MD servo is employed, it does not require any bothersome work for adjusting its angular position.

## ● Wiring from the Output Shaft

In order to avoid wires becoming entangled during operation, they are drawn out from the rear of the output shaft. Another shaft is installed on the bottom case to make the center impeller structure easy to construct.

## ● Horn with a mark for reference position

凹 marks are placed on the circumference of the servo horn in order to make sure the reference position even after being built in. (one 凹 for 0° position, two 凹 s for 90° position, three 凹 s for 180 position and four 凹 s for 270° position)。

# Part Names / Handling Instructions

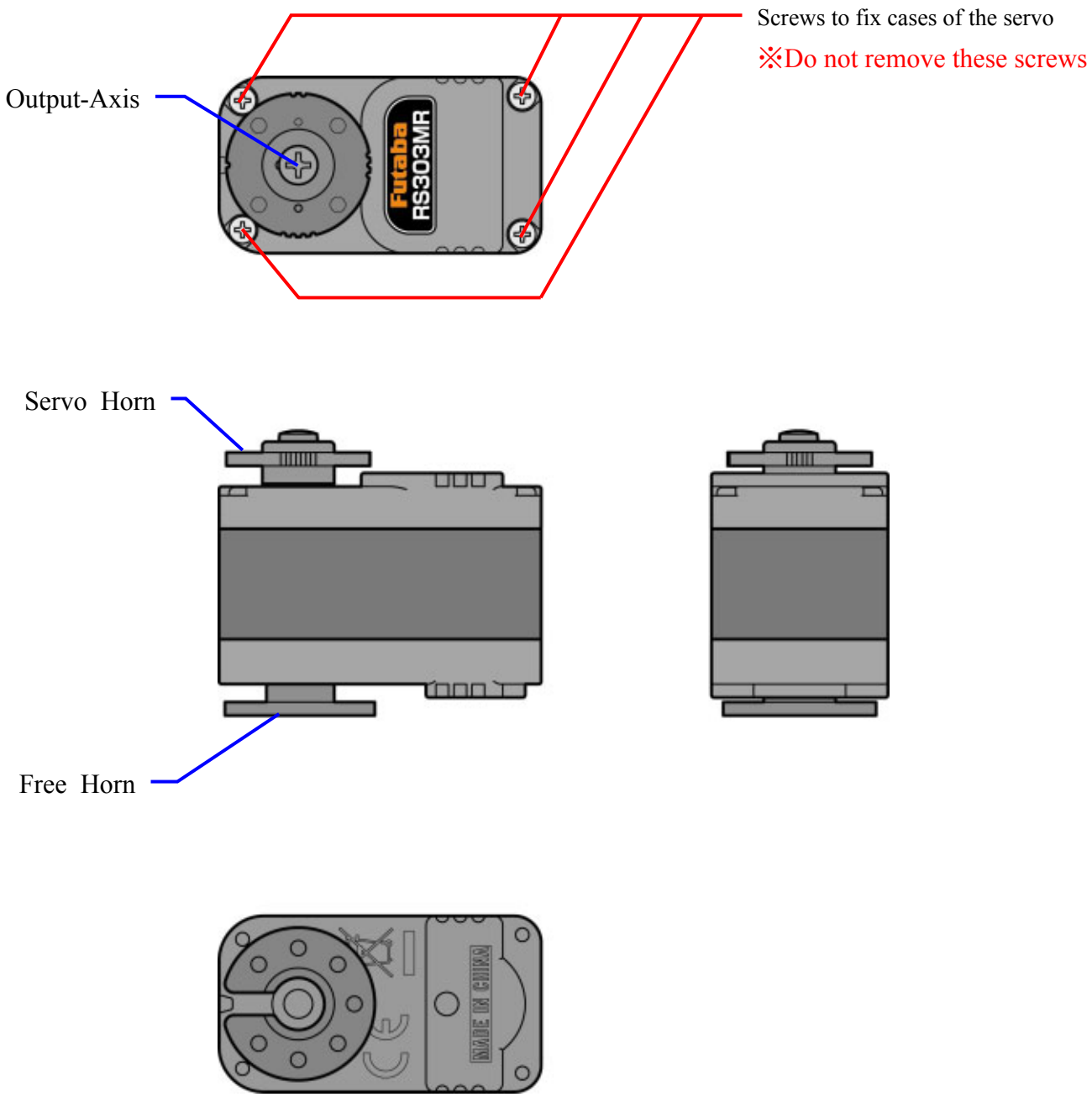



**Fig. 2.1 Part Names**

## Caution

Do not remove screws to fix cases of the servo.

Removing the screws for fixing the servo with a case may damage the servo.

# Pin Assignment of the Connector

The pin assignment of the connector of RS303MR/RS304 is shown in **Fig. 2.2**.

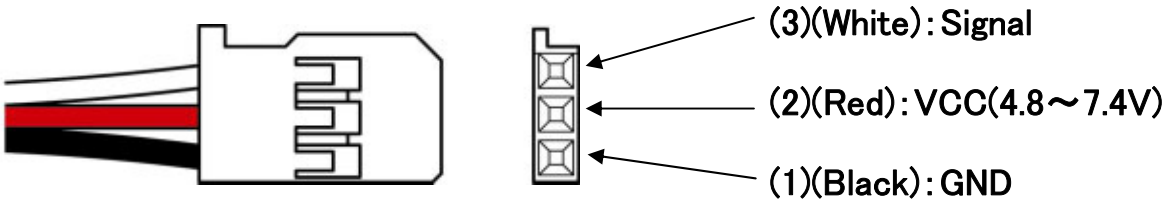



**Fig. 2.2 Connector Pin Assignment**

# Communication Settings

Set your Communication parameters as follows:

| | | | |
|---|---|---|---|
| Baud Rate | : | 115.2 | [kbps](9.6[kbps]～230.4[kbps], refer p.25) |
| Data bit length | : | 8 | [bit] |
| Parity | : | None | |
| Stop Bit | : | 1 | [bit] |
| Flow Control | : | None | |

# 3. Connections

## Systems

Systems of the robot with RS303MR/RS304MD are follows.

### ● System as Command-Type Servo

System with Command-Type Servo is shown in Fig. 3.1.

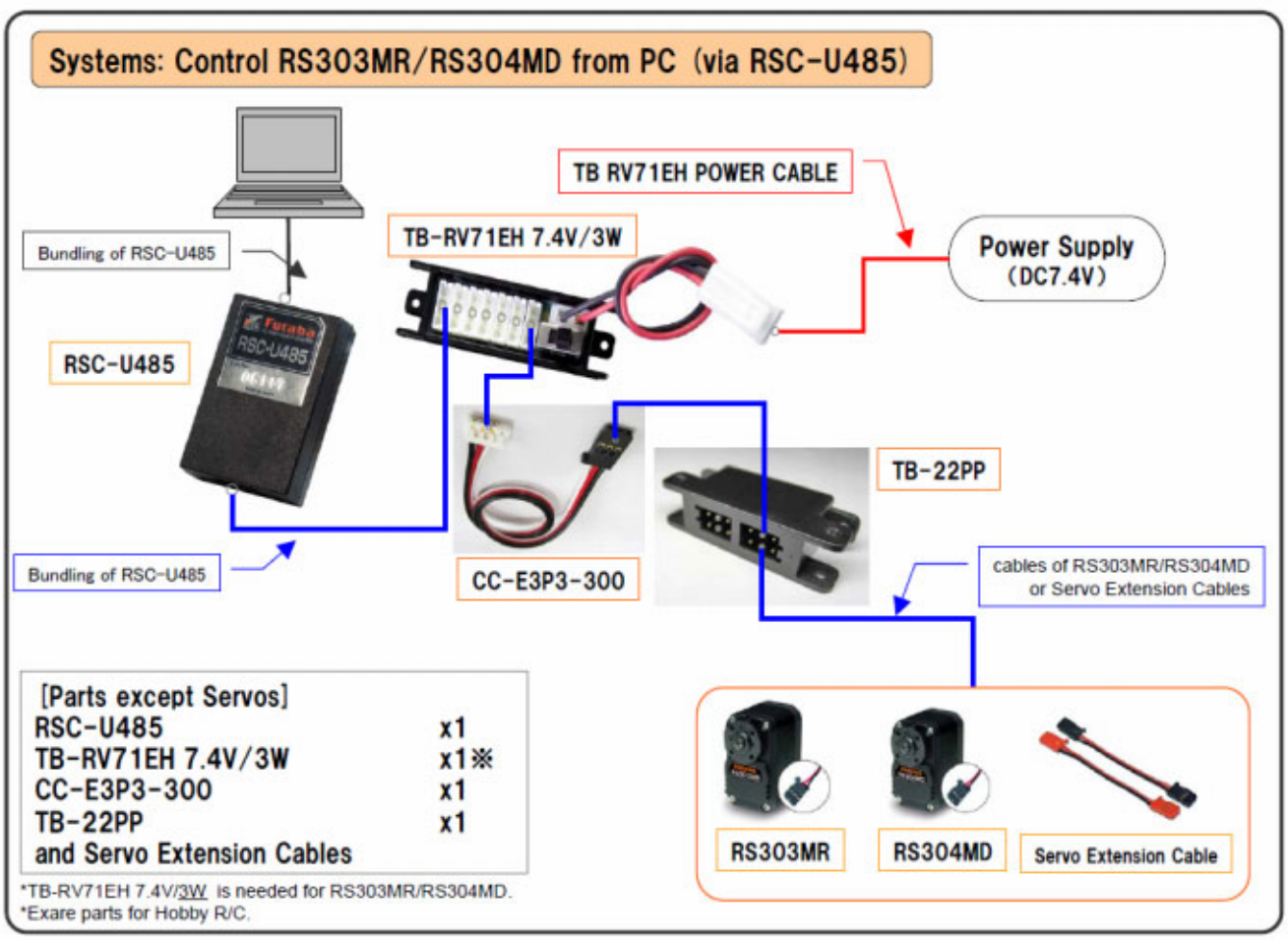



**Fig. 3.1 System with Command-Type Servo**

To Control RS303MR/RS304MD as Command-Type Servo with Processing Unit "RPU-10" or USB-RS485 Converter "RSC-U485", Power Hub "TB-RV71EH 7.4V/3W" is required somewhere in the system.

**Maximum number of RS303MR/RS304MD to be connected as TTL Command-Type Servo is 24.**

When these products are used together with RS485 Command-Type Servos, maximum number of the RS485 Command-Type Servo is 4.

Please note that when the Command-Type Servo are used together exceeding the maximum number, the servos may malfunction..

## ● Systems as PWM-Type Servo

System with Command-Type Servo is shown in Fig. 3.2.

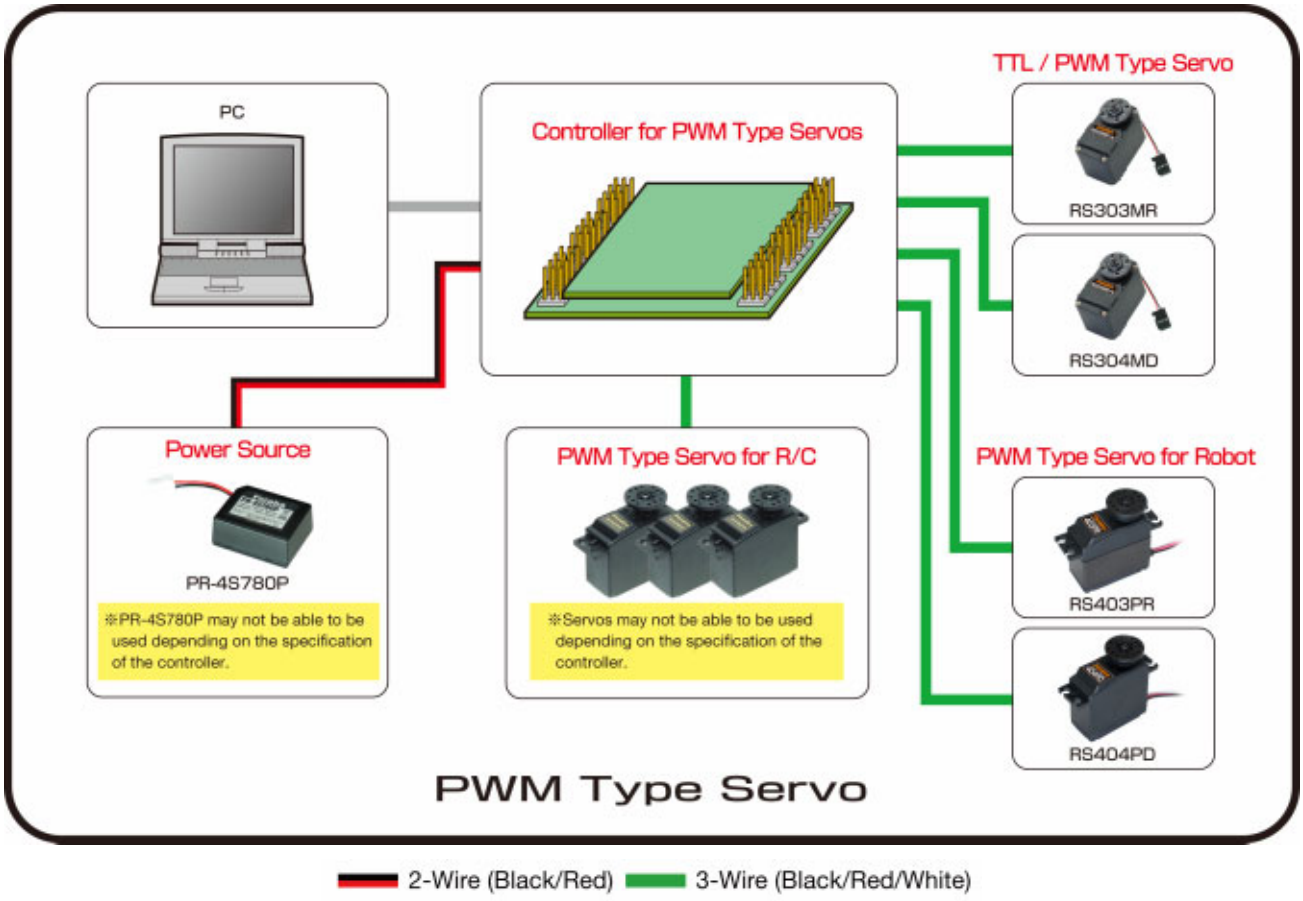



**Fig. 3.2 System with PWM-Type Servo**

Please note that the servos made by Futaba Corporation may malfunction, depending on the specification of the controller used for PWM servos. For details about the controller for

# Notes of Connection

RS303MR/RS304MD may malfunction, depending on the system configuration if they are used as TTL Command Type Servo.

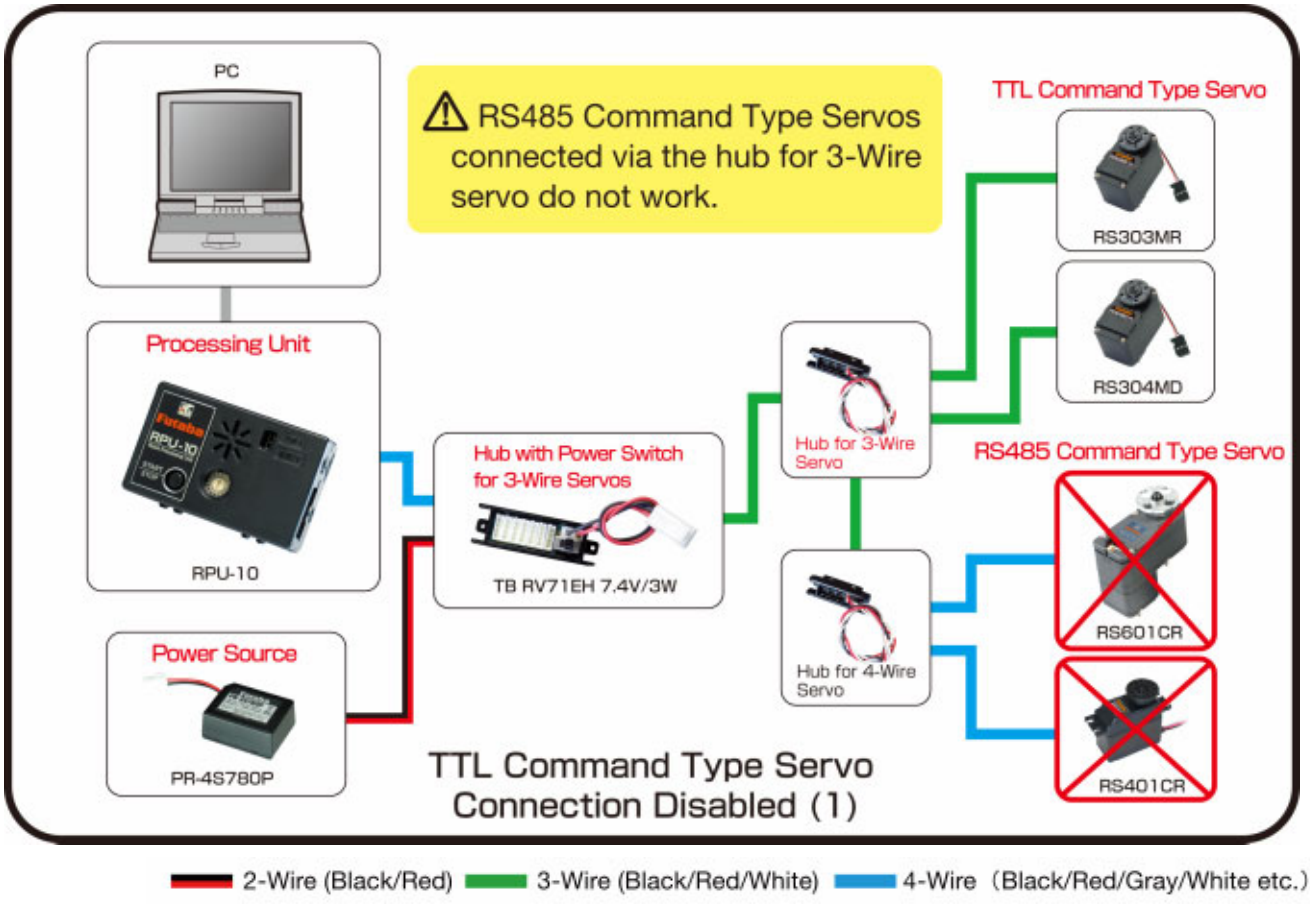



**Fig. 3.3 TTL Command-Type Servo / Example of Disabled Connection (1)**

RS485 Command Type Servos do not work if they are connected via the Hub for 3 Wire Servo.
When both RS485 Command Type Servos and TTL Command Type Servos are used in the same system, it is required not to create any part where three wires are placed on the route to the RS485 Command Type Servos from a processing unit (or, USB-RS485 converter).

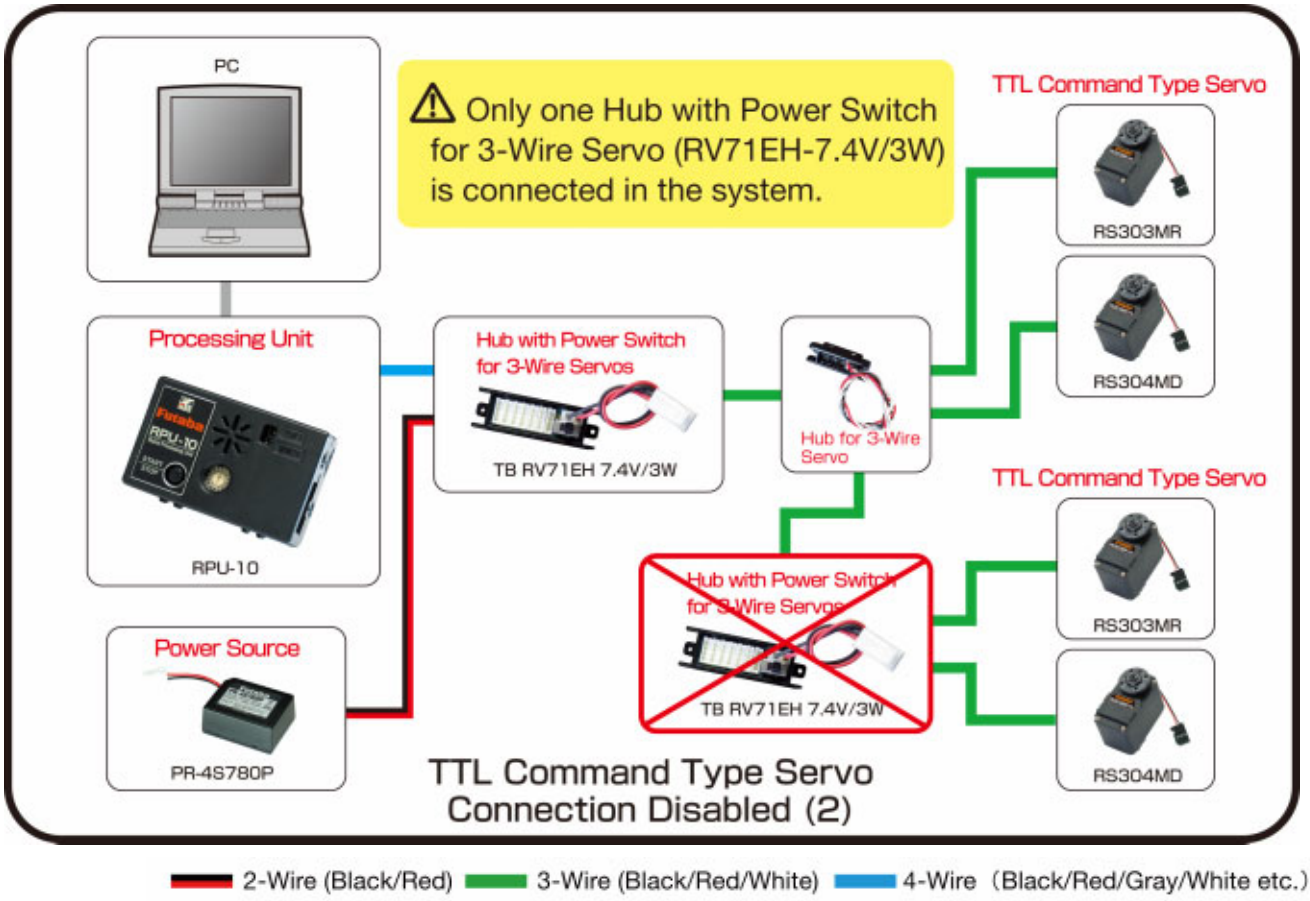



**Fig. 3.4 TTL Command-Type Servo / Example of Disabled Connection (2)**

The servos may malfunction if more than one Hub with Power Switch for TTL Command Type Servo (RV71EH-7.4V/3W) is connected in the same system.

# 4. Control

## Abstract

### ● Switch of Command-Type and PWM-Type

RS303MR/RS304MD can be controlled either as the Command-Type Servo of PWM-Type Servo.
Which Type will be applied is automatically decided by the first signal sent after the power is turned on. The Type once decided never changes until the power is turned off.

When RS303MR/RS304MD is controlled with a Robot Processing Unit "RPU-10", they are operated as Command Type Servo. When RS303MR/RS304MD is used by connecting to a radio control receiver or a controller for PWM-Type Servo, they are controlled as PWM-Type Servo.

### ● Communication Protocol (TTL Command-Type)

The communication protocol used for TTL Command Type Servo is asynchronous half-duplex communication. The signal line can be switched alternately for transmission or reception of data.
Normally, RS303MR and RS304MD stand by in a receiving mode. When they receive commands from RPU-10 to get the data of or the status of servos, the mode changes to a transmitting mode. After sending the data, they stand by again in a receiving mode.

### ● Control as PWM-Type Servo

To control RS303MR/RS304MD as a PWM-Type Servo, they are operated by changing the pulse width of the constant-frequency pulse (4ms-50ms). There is no direct relationship between the frequency, the torque, and the speed.
The relations between the pulse width and angles (positions) are as follows.

**Table 4.1 Pulse Width and Angles**

| Pulse Width | Angle(degrees) |
|---|---|
| 560 $\mu$s | +144 |
| 1520 $\mu$s | 0 |
| 2480 $\mu$s | -144 |

If there is no input for more than 80ms or an invalid value, less than 500μs or more than 2550μs, is input, the RS303MR and RS304MD will deactivate.(Refer p.27 No.22 Torque in Silence(1 Byte, Hex, Read/Write, for PWM)).

## ● Memory Map

RS303MR/RS304MD has its own memory area to store data necessary for its movement. This memory area is called “Memory Map”.
This memory map is divided into two groups. One is “RAM area” in which data will be erased when the power is turned off. The other one is “ROM area” in which data is held even after the power is turned off.

In “ROM area”, there are three different types of parameters; parameters used only for Command Type, parameters used only for PWM Type and those used for both Types. When these parameters are required to be rewritten, even parameters used only for PWM Type Servo, it is required to transfer the data as a Command Type Servo after connecting the servos to the PC with USB-RS485 converter (RSC-U485) etc.

While the servos are operated as PWM Type Servo, rewriting the parameters cannot be executed. Therefore, the servos are operated with the prewritten parameters.

## ● Servo ID

You can set “ID” numbers to RS303MR/RS304MD servos individually.
“ID” is used to identify servos during communication.

The default number of every servo is set to “1”.
When you use plural servos in a single communication network, give them different “ID” numbers.

## ● Packet

“Packet” is a block that is used for sending a command to or receiving data from RS303MR/RS304MD servos. Packets are divided into the following three groups, having different formats.

### Short Packet

Short Packets are used for sending the data in the memory map toward a single servo.

### Long Packet

Long Packets are used for sending the data in the memory map toward multiple servos simultaneously.

### Return Packet

Return Packet is a packet that is sent from a servo when a return packet is requested.

# Format of Packets

## ● Short Packet

Short Packet is used for sending the data in the memory map to a single servo.

**Structure**

| Header | ID | Flag | Address | Length | Count | Data | Sum |
|---|---|---|---|---|---|---|---|

### Header

This is a line head of a packet. Set “FA AF” for short packets.

### ID

Set “ID” of the servo to be sent the packet.
By setting “FF” (=255), commands are commonly effective to all servos,

### Flag

“Flag” shows reaction of the servo such as sending Return Packet or write ROM Area and so on.
For details, refer subsequent pages.

### Address

Set the starting address of Memory Map to be changed.

### Length

“Length” is the length of the data. Set the number of bytes of “Data”.

### Count

“Count” is the number of servos to be sent “Data”. Set “1” for a short packet.

### Data

“Data” to be written in the memory map of the servo.

### Sum

“Sum” is the value obtained from XOR operation on all bytes from ID through Data in a packet by a unit of a byte.

Ex.)”Sum” of following packet is “1C”.

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 1E | 02 | 01 | 00 00 | 1C |

**01H** XOR **00H** XOR **1EH** XOR **02H** XOR **01H** XOR **00H** XOR **00H** = **1C**

## Details of Flag

Each bit has the following meaning.

**Table 4.2 bit of Flag**

| Bit | Function |
|---|---|
| 7 | (Reserved) |
| 6 | Write Flash ROM |
| 5 | Reboot Servo |
| 4 | Initialize memory map data |
| 3 | Direct Address of Return Packet |
| 2 | Direct Address of Return Packet |
| 1 | Direct Address of Return Packet |
| 0 | Direct Address of Return Packet |

## Bit 7 : Reserved

Set “0” to this bit always.

## Bit 6 : Write Flash ROM

By setting this bit to “1” (Flags=40H) and sending a packet of address = FFH, Length = 00H, Count = 00H to a servo, data of the memory map from No.4 to No.29 is written in Flash ROM.

ex)Write Flash ROM of the servo (ID:01)

| Hdr | ID | Flg | Adr | Len | Cnt | Sum |
|---|---|---|---|---|---|---|
| FA AF | 01 | 40 | FF | 00 | 00 | BE |

The servo’s memory should be renewed with the data you want to write into the Flash ROM by transferring the data beforehand.
The servo ID becomes effective only after receiving a packet. The ID returns to the previous number on the next boot up unless the ID is written into the Flash ROM.

## Caution

Never turn off the power while the Flash ROM is being written.

## Bit 5 : Reboot Servo

Setting this bit to “1” (Flags=20H), and sending a packet with Address = FFH, Length = 00H, Count = 00H to a servo will reboot a servo.

Ex)Reboot servo(ID:01)

| Hdr | ID | Flg | Adr | Len | Cnt | Sum |
|---|---|---|---|---|---|---|
| FA AF | 01 | 20 | FF | 00 | 00 | DE |

**The packet that “Write Flash ROM” and “Reboot Servo” has to be sent separately.**

“Reboot Servo” packet must be sent after finishing “Write Flash ROM”.

## Bit 4 : Initialize the memory map from No.4 to No.29

Setting this bit to “1” (Flags=10H), and sending a packet with Address = FFH, Length = 00H, Count = 00H and data = FFH to a servo will initialize the memory map from No.4 to No.29 to their default value. Please refer to default value in the “Memory Map of ROM Area” (p.24) for more details.

Ex)Initialize the memory map of the servo (ID:01)

| Hdr | ID | Flg | Adr | Len | Cnt | Sum |
|---|---|---|---|---|---|---|
| FA AF | 01 | 10 | FF | FF | 00 | 11 |

Caution

After initializing the servo, “ID” of the servo becomes “1”.

## Bit 3～0 : Direct Address of Return Packet

### (1) Direct Area of Memory Map

Setting Bit 3 to Bit 0 of a Short Packet as **Table 4.3**, you can receive return data of the specified area of servo's memory map.
The RS485 half duplex communication does not allow addressing more than one servo that can send a return packet. After requesting a return packet, do not send next data until completing receiving of the return packet.

**Table 4.3 Direct Area of memory map**

| Bit | 3 | 2 | 1 | 0 | Function |
|---|---|---|---|---|---|
| | 0 | 0 | 0 | 0 | No return Packet |
| | 0 | 0 | 0 | 1 | Return ACK/NACK Packet |
| | 0 | 0 | 1 | 1 | Return the data of memory map No.00～No.29 |
| | 0 | 1 | 0 | 1 | Return the data of memory map No.30～No.59 |
| | 0 | 1 | 1 | 1 | Return the data of memory map No.20～No.29 |
| | 1 | 0 | 0 | 1 | Return the data of memory map No.42～No.59 |
| | 1 | 0 | 1 | 1 | Return the data of memory map No.30～No.41 |
| | 1 | 1 | 1 | 1 | Return the specified number of bytes of data starting from the specified address |

### (2) Direct specified address

Setting the Bit 3 to Bit 0 to "1" and sending a short command with the starting address whose data you want to receive, the length of data and the count=00H makes it possible to return the specified number of bytes of the data starting from the specified address.
Available addresses in the memory map are from No.00 to No.139 (00H~8BH).

Ex)Return the data of addresses from No.42 (2AH) through No.43 (2BH) of the servo(ID:1).

| Hdr | ID | Flg | Adr | Len | Cnt | Sum |
|---|---|---|---|---|---|---|
| FA AF | 01 | 0F | 2A | 02 | 00 | 26 |

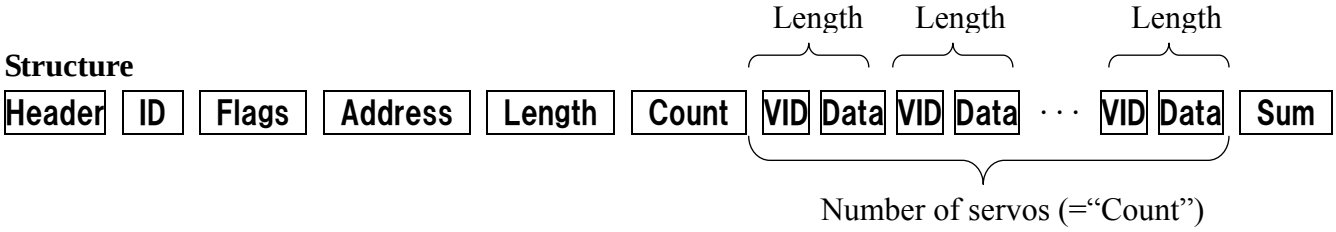
### (3) ACK/NACK Packet

By sending a Short Packet with the Bit 0=1, Bit1=0, Bit2=0, Bit3=0, the servo will return ACK/NACK packet.
ACK/NACK Packet is only 1 byte of data that is **07H** = "ACK" or **08H** = "NACK".

## ● Long Packet

"Long Packet" is used to send the data to multiple servos.
("Address" and "Length" are the same to all servos.)

**Structure**




**Header**

This notation indicates the front of a packet. Set "FA AF" for Long Packet.

**ID**

Set "0" for Long Packet always.

**Flags**

Set "0" for Long Packet always.

**Address**

Set the starting address of Memory Map to be changed.

**Length**

"Length" is the length of the data for each servos.
Length = the number of bytes of the data for each servos + 1(byte of VID)

**Count**

"Count" is the number of servos to be sent "Data".

**VID**

"VID" is the ID of each servo

**Data**

"Data" to be written in the memory map of each servo.

**Sum**

"Sum" is the value obtained from XOR operation on all bytes from ID through Data in a packet by a unit of a byte.

Ex)Set angle to 10 degrees to the Servos (ID:1 and ID:2) and 50 degrees to the servo (ID:5).

| Hdr | ID | Flg | Adr | Len | Cnt | VID | Dat | VID | Dat | VID | Dat | Sum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FA AF | 00 | 00 | 1E | 03 | 03 | 01 | 64 00 | 02 | 64 00 | 05 | F4 01 | ED |

The check sum of the send data above is:

**00H** XOR **00H** XOR **1EH** XOR **03H** XOR **03H** XOR **01H** XOR **64H** XOR **00H** XOR **02H** XOR **64H** XOR **00H** XOR **05H** XOR **F4H** XOR **01H** = **ED**

## ● Return Packet

"Return Packet" is the packet returned from the servo when the Flag field requests a servo to send a return packet.

**Structure**

| Header | ID | Flags | Address | Length | Count | Data | Sum |
|---|---|---|---|---|---|---|---|

### Header

This notation indicates the front of a packet. "FD DF" is set to Return Packet.

### ID

This is the "ID" of the servo that sent Return Packet.

### Flags

"Flags" of the Return Packet shows conditions of the servo.

**Table 4.4 Flags of Return Packet**

| Bit | Value | Meanings |
|---|---|---|
| 7 | 0: Normal / 1: Error | Temperature Limit (Torque OFF) |
| 6 | 0 | (Reserved) |
| 5 | 0: Normal / 1: Error | Temperature Alarm |
| 4 | 0 | (Reserved) |
| 3 | 0: Normal / 1: Error | Write Flash ROM Error |
| 2 | 0 | (Reserved) |
| 1 | 0: Normal / 1: Error | Received Packet Error |
| 0 | 0 | (Reserved) |

### Address

"Address" shows starting address of the data of return packet.

### Length

"Length" shows the number of bytes of "Data".

### Count

"Count" Shows the number of servos. It is set to "1" for Return Packet.

### Sum

"Sum" shows check sum of the Return Packet, and its value is the XOR from "ID" to the end of "Data" in byte units.

# Memory Map

## 4.1. Invariable ROM Area

**Table 4.5 Invariable ROM Area**

| Area | Address DEC | HEX | Initial | Name | R/W |
|---|---|---|---|---|---|
| Invariable | 00 | 00H | 30H(40H) | Model Number L | R |
| | 01 | 01H | 30H | Model Number H | R |
| | 02 | 02H | 03H | Firmware Version | R |
| | 03 | 03H | -- | Reserved | - |

(*) for RS304MD

### ● No.0/No.1 Model Number(2 Byte, Hex, Read)

It is the Model Number (Name of the servo).
"30 30" means RS"303"MR, and "30 40" means RS"304"MD

| | RS303MR | RS304MD |
|---|---|---|
| Model_Number L | 30H | 40H |
| Model_Number H | 30H | 30H |

### ● No.2 Firmware Version(1Byte, Hex, Read)

It is the version of the servo's firmware.
Its value is depending on the version at production (0x03 in the example below).
Firmware Version = 03H

### ※Saving 2-Byte data

Two-byte data is stored to the memory map in two individual 8-bit bytes of H (High byte) and L (Low byte).

Ex) Set Angle to 29.2 degrees to servo (ID:23)
Target angle is stored in "Goal Position" (Address 30/31) with unit of 0.1 degrees.
(29.2 [degrees] = 292 [0.1degrees, DEC] = 0124[0.1degrees, HEX])
Stored data is bellow:

Goal Position (L) = 24H
Goal Position (H) = 01H

## 4.2. Variable ROM Area

**Table 4.6 Variable ROM Area**

| Area | Address DEC | Address HEX | Initial | Name | R/W |
|---|---|---|---|---|---|
| Variable ROM | 04 | 04H | 01H | Servo ID | RW |
| | 05 | 05H | 00H | Reverse | RW |
| | 06 | 06H | 07H | Baud Rate | RW |
| | 07 | 07H | 00H | Return Delay | RW |
| | 08 | 08H | DCH | CW Angle Limit L | RW |
| | 09 | 09H | 05H | CW Angle Limit H | RW |
| | 10 | 0AH | 24H | CCW Angle Limit L | RW |
| | 11 | 0BH | FAH | CCW Angle Limit H | RW |
| | 12 | 0CH | 00H | Reserved | - |
| | 13 | 0DH | 00H | Reserved | - |
| | 14 | 0EH | 4DH | Temperature Limit L | R |
| | 15 | 0FH | 00H | Temperature Limit H | R |
| | 16 | 10H | 00H | Reserved | - |
| | 17 | 11H | 00H | Reserved | - |
| | 18 | 12H | 00H | Reserved | - |
| | 19 | 13H | 00H | Reserved | - |
| | 20 | 14H | 00H | Reserved | - |
| | 21 | 15H | 00H | Reserved | - |
| | 22 | 16H | 00H | Torque in Silence | RW |
| | 23 | 17H | C8H | Warm-up Time | RW |
| | 24 | 18H | 02H | CW Compliance Margin | RW |
| | 25 | 19H | 02H | CCW Compliance Margin | RW |
| | 26 | 1AH | 08H | CW Compliance Slope | RW |
| | 27 | 1BH | 08H | CCW Compliance Slope | RW |
| | 28 | 1CH | 64H(58H) | Punch L | RW |
| | 29 | 1DH | 00H(02H) | Punch H | RW |

(*) for RS304MD

*C: Effective only for Command-Type Control

*P: Effective only for PWM-Type Control

## ● No.4 Servo ID (1Byte, Hex, Read/Write)

It is the "ID" of the servo.

Its Initial value is 01H and the settable range is from 1 to 127 (01H to 7FH).

Ex) Set ID to "5" to the servo (ID:1).

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 04 | 01 | 01 | 05 | 00 |

The servo begins to operate under the new ID as soon as the ID rewrite packet is received.

Note that the ID returns to previous ID if it was not written to the Flash ROM before turning off the power.

## ● No.5 Servo Reverse(1Byte, Hex, Read/Write)

It is the direction of rotation of the servo.

Its initial value is 00H that means the normal rotation, and the value of 01H means reverse rotation.

If it is set to 01H (reverse rotation), the Angle Limit is also reversed.

Ex) Reverse the servo (ID:1):

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 05 | 01 | 01 | 01 | 05 |

## ● No.6 Baud Rate(1Byte, Hex, Read/Write)

It is the baud-rate of communication.

Initial value is 07H (115,200bps) and the settable range is from 0 to 9 (00H to 09H)

The values and baud rate is assigned as shown in **Table 4.7**.

**Table 4.7 Baud Rate**

| Value | Baud Rate | Value | Baud Rate |
|---|---|---|---|
| 00H | 9,600bps | 05H | 57,600bps |
| 01H | 14,400bps | 06H | 76,800bps |
| 02H | 19,200bps | 07H | 115,200bps |
| 03H | 28,800bps | 08H | 153,600bps |
| 04H | 38,400bps | 09H | 230,400bps |

Even after the value is rewritten, the servos are operated at the previous baud rate.

In order to operate under the new baud rate, it is required to write Flash ROM and Reboot Servo.

Ex) Set baud rate as 38,400 bps to the servo (ID:1)

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 06 | 01 | 01 | 04 | 03 |

**After sending above packet, "Write Flash ROM" packet and "Reboot Servo" packet must be sent.**

## ● No.7 Return Delay(1Byte, Hex, Read/Write)

It is the delay time for reply when the Return Packet is required.

The servo sends the return packet in 100μs after receiving data with the setting of 0.
The parameters of No.7 are in units of 001H = 50μs.
If you want to set the delay time for reply to 1ms, write 18 (12H). (1ms=100μs+18x50μs)

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 60 | 07 | 01 | 01 | 12 | 74 |

## ● No.8/No.9/No.10/No.11 Angle Limit (2 Byte, Hex, Read/Write)

It is the maximum operating angle based on 0 degree (units: 0.1 degree).
No.8 and No.9 are used for CW (clockwise) direction and No.10 and No.11 are for CCW (counterclockwise) direction.

When the set angle is larger than the set Angle Limit, the servo rotates to the maximum operating angle.
Initial value and settable range is;

| Direction | Initial | range |
|---|---|---|
| CW | +150(05DCH) | 0 (0000H) ~ +150 (05DCH) |
| CCW | -150(FA24H) | 0 (0000H) ~ -150 (FA24H) |

Ex. 1) Set the CW Angle Limit of servo (ID: 1) to 100.0 degrees.
Since the angle is set in 0.1 degree units, 100.0 degrees = 1000(03E8H) is set.
CW Angle Limit L = E8H, CW Angle Limit H = 03H

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 08 | 02 | 01 | E8 03 | E1 |

Ex. 2) Set the CCW Angle Limit of servo (ID: 1) to -100.0 degrees.
-100.0 degrees = -1000(FC18H).
CCW Angle Limit L = 18H, CCW Angle Limit H = FCH

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 10 | 02 | 01 | 18 FC | F6 |

## ● No.14/ No.15 Temperature Limit(2 Byte, Hex, Read)

It is the limit value of the internal temperature of the servo.

When the internal temperature is increased by the heat of the motor, etc. and exceeds the set value, the servo will automatically turns off its torque to avoid troubles.
To turn on the torque again, cool the servo and reboot it.
*Note that rewriting this value will null and void the product warranty.

## ● No.22 Torque in Silence(1 Byte, Hex, Read/Write, for PWM)

It is the condition of the servo's output torque.
When there is no input for more than 80ms or continuous invalid input that the pulse width is less than 0.5ms or more than 2.55ms, the servos' torque condition changes to the specified condition.

The initial value is 0(00H) and the relationship between the set value and the condition is shown as below;

**Table 4.8 Torque Conditions**

| Value | Condition |
|---|---|
| 0(00H) | Torque OFF |
| 1(01H) | Torque ON and keep the same angle* |
| 2(02H) | Brake mode (refer p.32) |

*The servo continues maintain the angle directed just before the losing input or invalid input is started.

## ● No.23 Preparation Time (1 Byte, Hex, Read/Write, for PWM)

It is the time of the first movement after the power is turned on.
Setting this parameter enables the servo to avoid moving suddenly just after turning on the power.

Its unit is 10ms and the settable range is 0~255 (00H~FFH).
The Initial value is 200(C8H).

## ● No.24 / No.25 Compliance Margin (1 Byte, Hex, Read/Write)

It is the allowable range of the angle around the goal angle.
If the error between the present angle and the goal position is in the set range, the servo recognized itself to be in the goal position and stop moving.

No.24 is for CW and No.25 is for CCW.
The unit is 0.1 degree, initial value is 2(02H) and the settable range is 0~255(00H~FFH) for both directions.

## ● No.26 / No.27 Compliance Slope (1 Byte, Hex Read/Write)

It is the range that output torque of the servo increases in proportion to the error between the present angle and aim angle. The flexibility of the servo increases in proportion to this value.

No.26 is for CW and No.27 is for CCW.
The unit is 1.0 degree, initial value is 8(08H) and the settable range is 0~255(00H~FFH) for each directions.

## ● No.28 / No.29 Punch (2 Byte, Hex, Read/Write)

It is the minimum torque (electric current) that is generated when present angle of the servo exceeds the range of Compliance Margin.

The unit is 0.01% of the maximum torque and the settable range is 0~10,000(00H~2710H)
The initial value for RS303MR is 0064H (1.0%) and for RS304MD is 0258H (6.0%).

The relationship of the output torque, error (=between the present angle and the goal position) and compliance parameters are shown in **Fig. 4.1**.

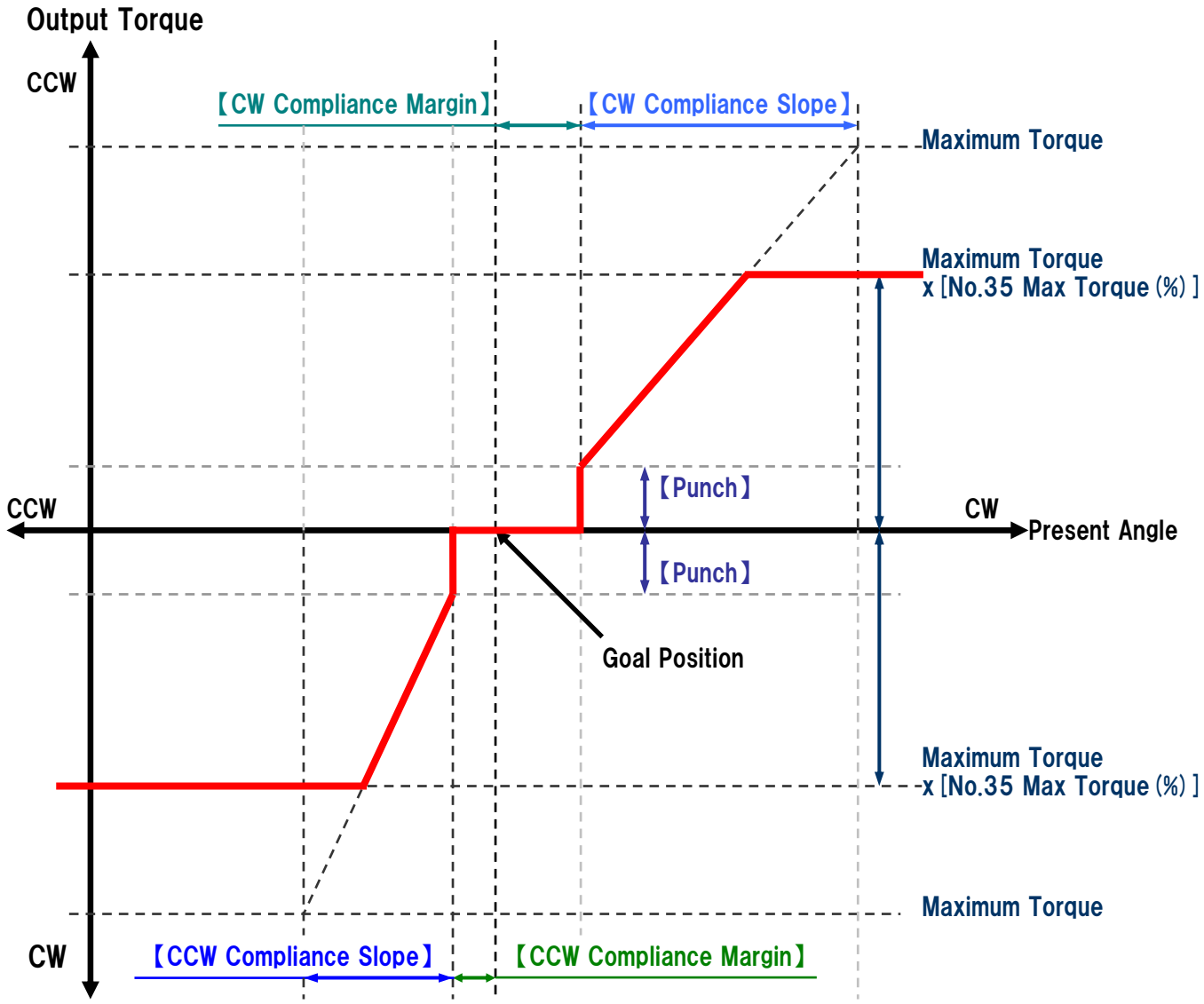



**Fig. 4.1 The relationship of Output Torque, Angle and Compliance settings.**

Ex.1) Set Punch of the servo (ID: 1) to 1[%]( = 100(0064H)).

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 1C | 02 | 01 | 64 00 | 7A |

Ex.2) Set the servo (ID: 1) as shown below;

| | | |
|---|---|---|
| CW Compliance Margin | = | 03H |
| CCW Compliance Margin | = | 03H |
| CW Compliance Slope | = | 14H |
| CCW Compliance Slope | = | 14H |
| Punch | = | 0064H |

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 18 | 06 | 01 | 03 03 14 14 64 00 | 7A |

## 4.3. Variable RAM Area

**Table 4.9 Variable RAM Area of the Memory Map of RS303MR/RS304MD**

| Area | Address DEC | Address HEX | Initial | Name | R/W |
|---|---|---|---|---|---|
| Variable RAM Area | 30 | 1EH | 00H | Goal Position L | RW |
| | 31 | 1FH | 00H | Goal Position H | RW |
| | 32 | 20H | 00H | Goal Time L | RW |
| | 33 | 21H | 00H | Goal Time H | RW |
| | 34 | 22H | 00H | Reserved | - |
| | 35 | 23H | 64H | Max Torque | RW |
| | 36 | 24H | 00H | Torque Enable | RW |
| | 37 | 25H | 00H | Reserved | - |
| | 38 | 26H | 00H | Reserved | - |
| | 39 | 27H | 00H | Reserved | - |
| | 40 | 28H | 00H | Reserved | - |
| | 41 | 29H | 00H | Reserved | - |
| | 42 | 2AH | 00H | Present Position L | R |
| | 43 | 2BH | 00H | Present Position H | R |
| | 44 | 2CH | 00H | Present Time L | R |
| | 45 | 2DH | 00H | Present Time H | R |
| | 46 | 2EH | 00H | Present Speed L | R |
| | 47 | 2FH | 00H | Present Speed H | R |
| | 48 | 30H | 00H | Present Current L | R |
| | 49 | 31H | 00H | Present Current H | R |
| | 50 | 32H | 00H | Present Temperature L | R |
| | 51 | 33H | 00H | Present Temperature H | R |
| | 52 | 34H | 00H | Present Volts L | R |
| | 53 | 35H | 00H | Present Volts H | R |
| | 54 | 36H | 00H | Reserved | - |
| | 55 | 37H | 00H | Reserved | - |
| | 56 | 38H | 00H | Reserved | - |
| | 57 | 39H | 00H | Reserved | - |
| | 58 | 3AH | -- | Reserved | - |
| | 59 | 3BH | -- | Reserved | - |

## ● No.30 / No.31 Goal Position (2 Byte, Hex, Read/Write)

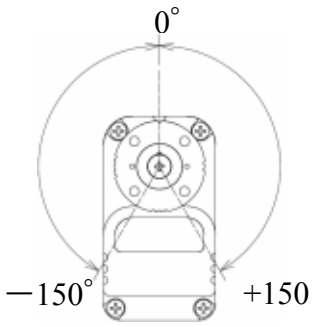



Fig. 4.2 Servo' s Angle

This parameter is the target angle of the servo.
Center of the movable range is 0 degrees and CW direction is "+" and CCW direction is "-" from the top of the servo (nameplate side).
Unit of Goal Position is 0.1 degree and its settable range is -1500~+1500.

If the set value is out of the range specified by the No.8~No.11 ("Angle Limit"), the servo moves to the set maximum or minimum angle.
When No.36 "Torque Enable" is set from 0 (Torque OFF) to 1 (Torque ON), "Goal Position" is automatically rewrite to the value of Present Position.
When a packet that direct Torque ON and set Goal Position in 1 packet (packet to write No.30~36 at once) is received, the servo turns on its torque but not moves.

Ex.1) Move servo (ID: 1) to 90.0 degree.

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 1E | 02 | 01 | 84 03 | 9B |

Ex.2) Move servo (ID: 1) to -90.0 degree

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 1E | 02 | 01 | 7C FC | 9C |

## ● No.32 / No.33 Goal Time (2 Byte, Hex, Read/Write)

This parameter is the time to move to "Goal Position".

The unit is 10ms and the settable range is 0~16,383(3FFFH), but note that error occurs about up to 5% when the set value is too big.
In the case that the speed required by "Goal Position" and "Goal Time" is faster than the maximum speed of the servo, the servo moves with its maximum speed.

Ex.1) Move the servo (ID: 1) to 90.0 degree in 5 sec.
90.0 degree = 900(0384H), 5 sec = 500 (01F4H)

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 1E | 04 | 01 | 84 03 F4 01 | 68 |

Ex.2) Move the servo (ID: 1) to -120.0 degree in 10 sec.
-120.0 degree = -12000(FB50H) , 10 sec = 1000(03E8H)

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 1E | 04 | 01 | 50 FB E8 03 | 5A |

## ● No.35 Maximum Torque (1 Byte, Hex, Read/Write)

This parameter set the maximum torque of the servo.
In PWM-Type Control, the servo always moves with initial value (100%) regardless of set value.

The unit is 1 % with the torque described in this manual (p.39) as 100%.
The initial value is 100(64H) and the settable range is 0~100(00H~64H)

Ex) Set Maximum Torque of the servo (ID: 1) to 80%(=50H).

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 23 | 01 | 01 | 50 | 72 |

## ● No.36 Torque Enable (1 Byte, Hex, Read/Write)

It is the condition of the servo's torque.
The relationship of the value and the condition is shown as below;

| Value | Condition |
|---|---|
| 0(00H) | Disable (Torque OFF) |
| 1(01H) | Enable (Torque ON) |
| 2(02H) | Brake mode |

The initial value is 0(00H) when the power is turned on.
In "Brake mode", the servo does not have output torque, but weak resistance torque occurs when it is turned from the outside.

Ex.1) Turn on the torque of the servo (ID: 1).

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 24 | 01 | 01 | 01 | 24 |

Ex.2) Turn off the torque of the servo (ID: 1)

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 24 | 01 | 01 | 00 | 25 |

Ex.3) Set the servo (ID: 1) to "Brake mode "

| Hdr | ID | Flg | Adr | Len | Cnt | Dat | Sum |
|---|---|---|---|---|---|---|---|
| FA AF | 01 | 00 | 24 | 01 | 01 | 02 | 27 |

## ● No.42 / No.43 Present Position (2 Byte, Hex, Read)

It is the angle of the servo.

Center of the movable range is 0 degrees and CW direction is “+” and CCW direction is “-” from the top of the servo (nameplate side).Unit of Goal Position is 0.1 degree and its range is -1500~+1500.

Ex) Get “Present Position” of the servo (ID: 1)
To get the value of “Present Position”(No.42 and No.43 of the Memory Map) as a Return Packet, a Short Packet (“Flag”:bit3=1, bit2=0, bit1=0, bit0=1,”Address”=0, “Length”=0, “Count”=1 and no ”Data” ) is required to be sent.

A Short Packet to require Return Packet with Memory Map No.42 ~ No.49 is;

| Hdr | ID | Flg | Adr | Len | Cnt | Sum |
|---|---|---|---|---|---|---|
| FA AF | 01 | 09 | 00 | 00 | 01 | 09 |

An example of Return Packet is;

| | | | | | | Data | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hdr | ID | Flg | Adr | Len | Cnt | 42 | 43 | ··· | ··· | 58 | 59 | Sum |
| FD DF | 01 | 00 | 2A | 12 | 01 | 84 03 | 00 00 00 00 06 00 | | ···00 00 00 00 00 00 | | | B9 |

2 Byte from the top of “Data” is “Present Position” (No.42 and No.43 of Memory Map), then “Present Position” is 0384H=900= 90.0degrees.
*The value of "Data" varies according to the conditions of the real servo.

## ● No.44/No.45 Present Time (2 Byte, Hex., Read)

It is an elapsed time after a servo receives a packet to move.

When movement is completed, it maintains the last value.
If the “Goal Time” of the movement is “0(00H)”, “Present Time” will not be rewritten (maintain last value).

Ex) Get “Present Time” of the servo (ID: 1)
A Short Packet to require Return Packet with Memory Map No.42 ~ No.49 is the same as the previous paragraph.

A Short Packet to require Return Packet with Memory Map No.42 ~ No.49 is;

| Hdr | ID | Flg | Adr | Len | Cnt | Sum |
|---|---|---|---|---|---|---|
| FA AF | 01 | 09 | 00 | 00 | 01 | 09 |

An example of Return Packet is;

| | | | | | | Data | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hdr | ID | Flg | Adr | Len | Cnt | 42 | 43 | 44 | 45 | ··· | ··· | 58 | 59 | Sum |
| FD DF | 01 | 00 | 2A | 12 | 01 | 5C FF | 37 | 02 | 00 00 07 00 | · · · | 00 00 00 00 00 00 | | | A9 |

The 3rd and 4th Byte of the “Data” is “Present Time”(No.44 and No.45 of Memory Map), then “Present Time” is 0237H=567=5670msec.

*The value of "Data" varies according to the conditions of the real servo.

## ● No.46/No.47 Present Speed (2 Byte, Hex, Read)

It is the rotational speed and its unit is deg/sec.

Ex) Get “Present Speed” of the servo (ID: 1)

A Short Packet to require Return Packet with Memory Map No.42 ~ No.49 is the same as the previous paragraphs.

A Short Packet to require Return Packet with Memory Map No.42 ~ No.49 is;

| Hdr | ID | Flg | Adr | Len | Cnt | Sum |
|---|---|---|---|---|---|---|
| FA AF | 01 | 09 | 00 | 00 | 01 | 09 |

An example of Return Packet is;

| | | | | | | Data | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hdr | ID | Flg | Adr | Len | Cnt | 42 | 43 | 44 | 45 | 46 | 47 | ··· | ··· | 58 | 59 | Sum |
| FD DF | 01 | 00 | 2A | 12 | 01 | 5C | FF | 37 | 02 | 2C | 01 | 07 | 00 | · · · | 00 | 00 | 00 | 00 | 00 | 00 | 84 |

2 Byte as No.46 and No.47 of “Data” is the “Present Speed”, then “Present Speed” is 012CH=300 deg/sec.

*The value of "Data" varies according to the conditions of the real servo.

## ● No.48/No.49 Present Current (2 Byte, Hex, Read)

It is the electric current of the servo and its unit is 1mA.

It is almost proportional to output torque, but does not become 0 even in the condition of Torque-OFF.

Ex) Get “Present Current” of the servo (ID: 1)

A Short Packet to require Return Packet with Memory Map No.42 ~ No.49 is the same as the previous paragraphs.

A Short Packet to require Return Packet with Memory Map No.42 ~ No.49 is;

| Hdr | ID | Flg | Adr | Len | Cnt | Sum |
|---|---|---|---|---|---|---|
| FA AF | 01 | 09 | 00 | 00 | 01 | 09 |

An example of Return Packet is;

| | | | | | | Data | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hdr | ID | Flg | Adr | Len | Cnt | 42 | 43 | ··· | | ··· | 48 | 49 | ··· | | | ··· | 58 | 59 | Sum |
| FD DF | 01 | 00 | 2A | 12 | 01 | 4E FB 00 00 00 06 00 00 BA 03 00 00 00 00 00 00 00 00 | | | | | | | | | | | | | 32 |

2 Byte as No.48 and No.49 of “Data” is the “Present Current”, then “Present Current” is 0006H=6mA.

*The value of "Data" varies according to the conditions of the real servo.

## ● No.50/No.51 Present Temperature (2 Byte, Hex, Read)

It is the temperature of the board in the servo.
The sensor has individual difference about up to ±3 degrees Celsius.

When “Present Temperature” reaches low temperature 10 degrees Celsius than a “Temperature Limit” (No.14 and No.15 of Memory Map), “Temperature Alarm” (Bit 5 of Flag of Return Packet) becomes “1”.
When “Present Temperature” Exceeds “Temperature Limit”, “Temperature Limit” (Bit 7 of Flag of Return Packet) becomes “1” and the servo will be “Brake mode”(No.36 of Memory Map becomes to “2”) automatically.

The temperature reaches “Temperature Limit” once, the servo will not accept Torque-ON command until it is rebooted or is turned off-and-on the power again.

When “Present Temperature” reaches “Temperature Limits”, temperature around the motor of the servo reaches to 120~140 degrees Celsius. Please be careful about burns and use the servo after the temperature fell enough.

Ex) Get “Present Temperature” of the servo (ID: 1)
A Short Packet to require Return Packet with Memory Map from No.42 and No.59 is;

| Hdr | ID | Flg | Adr | Len | Cnt | Sum |
|---|---|---|---|---|---|---|
| FA AF | 01 | 09 | 00 | 00 | 01 | 09 |

An example of Return Packet is;

| | | | | | | Data | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hdr | ID | Flg | Adr | Len | Cnt | 42 | 43 | ・・・ | ・・・ | 50 | 51 | ・・・ | ・・・ | 58 | 59 | Sum |
| FD DF | 01 | 00 | 2A | 12 | 01 | 4E FB | 00 | 00 | 00 | 00 | 06 | 00 | 2D | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | A6 |

2 Byte as No.50 and No.51 of “Data” is the “Present Temperature”, then the value is 002DH=45degrees Celsius.
*The value of "Data" varies according to the conditions of the real servo.

## ● No.52/No.53 Present Voltage (2 Byte, Hex, Read)

It is the voltage of the power applied to the servo and its unit is 10mV.
The sensor has individual difference about up to ±0.3V.

Ex) Get “Present Voltage” of the servo (ID: 1)
A Short Packet to require Return Packet with Memory Map from No.42 and No.59 is;

| Hdr | ID | Flg | Adr | Len | Cnt | Sum |
|---|---|---|---|---|---|---|
| FA AF | 01 | 09 | 00 | 00 | 01 | 09 |

An example of Return Packet is;

| | | | | | | Data | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hdr | ID | Flg | Adr | Len | Cnt | 42 | 43 | ・・・ | ・・・ | 52 | 53 | ・・・ | ・・・ | 58 | 59 | Sum |
| FD DF | 01 | 00 | 2A | 12 | 01 | 4E FB 00 00 00 00 06 00 2D 00 | | | | E4 | 02 | 00 00 00 00 00 00 | | | | A6 |

2 Byte as No.52 and No.53 of “Data” is the “Present Voltage”, then the value is 02E4H=7.4V.
*The value of "Data" varies according to the conditions of the real servo.

# 5. References

## Specs

| | | | | | | |
|---|---|---|---|---|---|---|
| Application | | Actuators for Robots | | | | |
| Dimensions | | 35.8 x 19.6 x 25.0 [mm]<br>*Refer next page for more details. | | | | |
| Weight | | RS303MR | 28 | [g] | | |
| | | RS304MD | 21 | [g] | | |
| Consumption Current<br>(Room Temperature, No Load) | | (in suspension) | 16 | [mA] | | |
| | | (in operation) | 90 | [mA] | | |
| Maximum Output Torque<br>(at 7.4V) | | RS303MR | 6.5 | [kgf·cm] | | |
| | | RS304MD | 5.0 | [kgf·cm] | | |
| Maximum Speed<br>(at 7.4V, No Load) | | RS303MR | 0.11 | [sec/60degree] | | |
| | | RS304MD | 0.16 | [sec/60degree] | | |
| Direction | CW | Present Position < Goal Position | | | | |
| | CCW | Present Position > Goal Position | | | | |
| Angle Range | CW | 150[degree] (Command-Type) / 144 [degree] (PWM-Type) | | | | |
| | CCW | 150[degree] (Command-Type) / 144 [degree] (PWM-Type) | | | | |
| Supply Voltage | | 4.8 | ~ | 7.4 | [V] | |
| Temperature Range | | (to operate) | 0 | ~ | +40 | [degrees Celsius] |
| | | (to store) | -20 | ~ | +60 | [degrees Celsius] |
| Power Source | | Lithium Polymer Battery (2 cells) | | | | |
| Communication | | Baud Rate: | Maximum 230.4 kbps | | | |
| | | Protocol: | 8bit, Stop bit 1, None Parity, Asynchronous | | | |

# Dimensions

## ● RS303MR/RS304MD

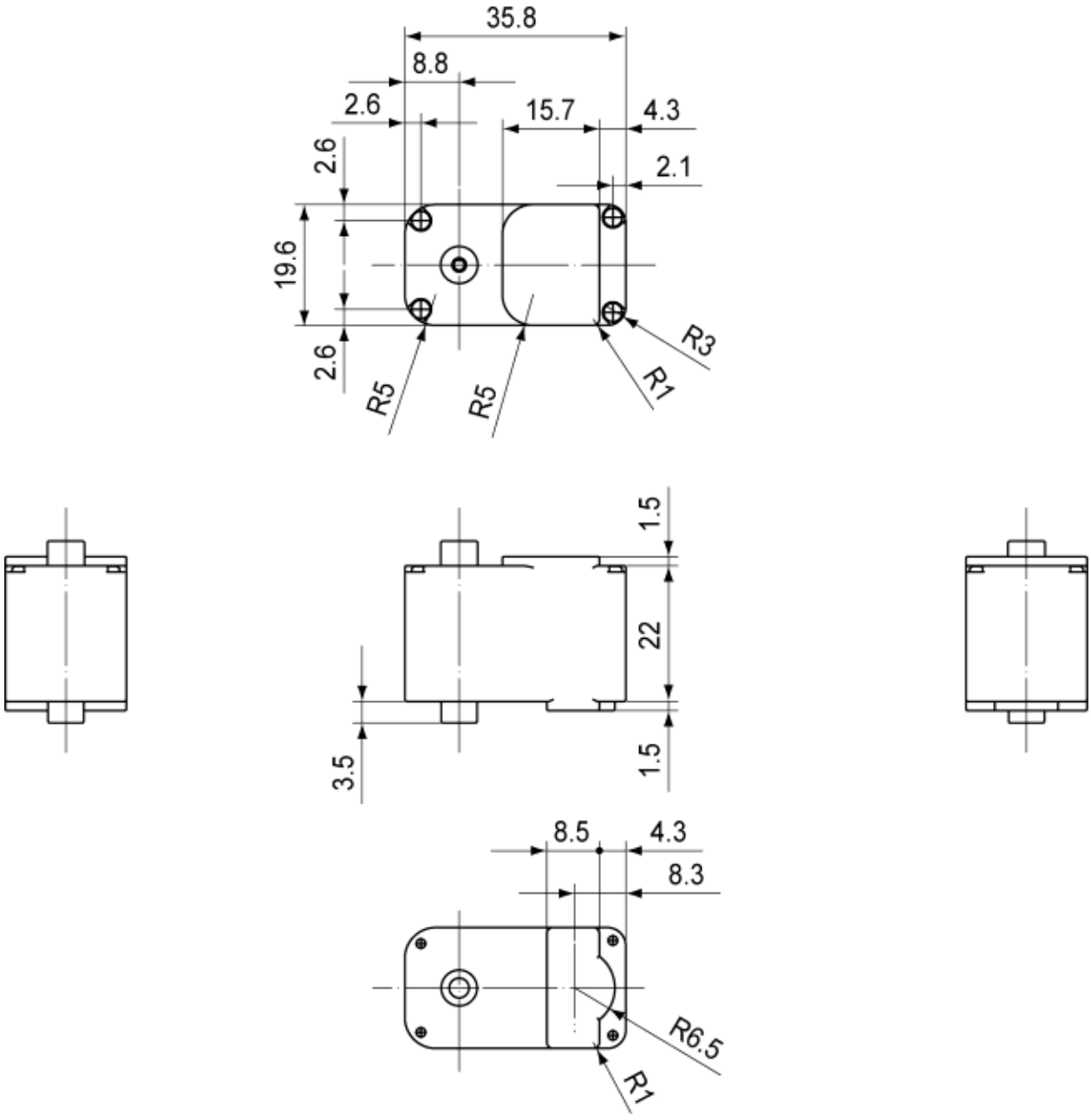



**Fig. 5.1 RS303MR/ RS304MD (unit: mm)**

## ● RS303MR/RS304MD with Servo Horn and Free Horn

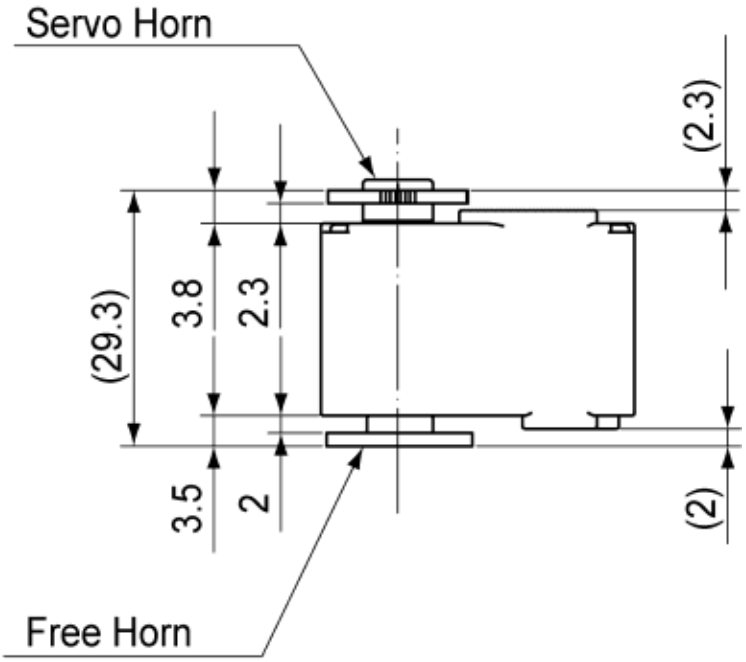



**Fig.5.2 RS303MR/RS304MD with Servo Horn and Free Horn (unit: mm)**

*A gap of 0.3mm occurs between Servo Horn and the body of servo when the Servo Horn is attached exactly.

*Free Horn is not fixed to the servo. It turns freely around the axis of the bottom case and is unlocked along to the axis of the bottom case.

## ● Servo Horn and Free Horn of RS30x Series

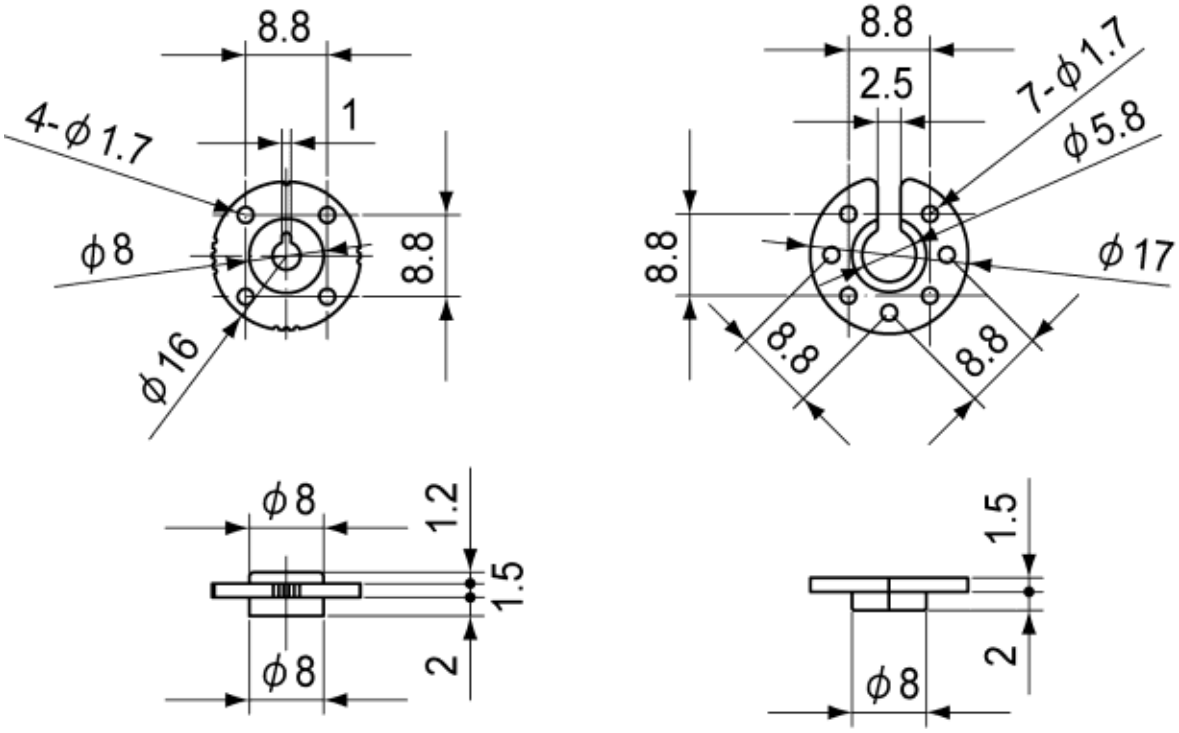



**Fig. 5.3 (L) Servo Horn (R) Free Horn**

# Option Parts

**Table 5.1 Option Parts for RS303MR/RS304MD**

| Name | |
|---|---|
| RS301_303 SCREW 2x6 | Screws to fix servo horn for RS301CR/RS303MR (10pcs) |
| RS302_304 SCREW 2x8 | Screws to fix servo horn for RS302CD/RS304MD (10pcs) |
| ROBOT SERVO HORN RH01 | Servo Horns for RS30x Series (5pcs) |
| RS301_303 GEAR SET | Gears for RS301CR/RS303MR |
| RS302_304 GEAR SET | Gears for RS302CD/RS304MD |
| RS30x CASE SET | Cases for RS30x Series Servo |
| RS30x BOTTOM CASE-SHAFT | Bottom Case with Shaft for RS30x Series |
| RS30x FREE HORN SET | Free Horns for RS30x Series (5pcs) |
| CC-E3P3-300 | Cable to connect TB22PP to other hubs, 300mm |
| TB22PP | Hub for TTL-Command Type Servo |

**Fig. 5.4 (L) BB0131 CC-E3P3-300 (R) BB0132 TB22PP**

Futaba®

© FUTABA CORPORATION 2011. June.