Robust Mini Linear Servo Actuator

mightyZAP

User Manual

iR ROBOT

INDEX

1 Before Use

1.1. Introduction

Thank you for purchasing mightyZAP mini Linear servo motors! Please peruse this manual before use to prevent any unexpected damage of product or serious injury of users.

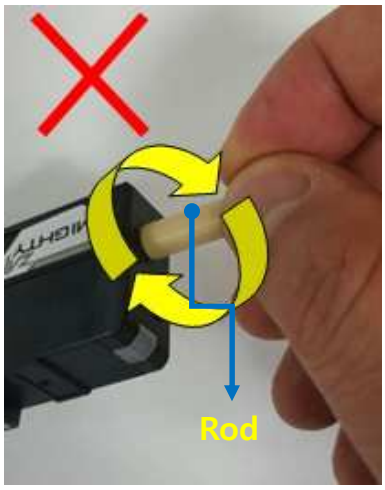
mightyZAP mini Linear servo motors have been developed to provide reliable, high quality linear solution in compact space. mightyZAP mini Linear servo motors can be applied in various fields such as factory automation, medical devices, robotics, professional UAV and radio control hobby.

[Features]

- Position Control (Positional Accuracy 50~90um – see spec chart of each model)
- Embedded Drive circuit
- 4096 Step High Resolution
- High Performance Coreless Motor
- Minimized Mechanical Backlash (30um)
- Excellent Substitute for pneumatic cylinder which does not support position control
- Reasonable Cost

1.2. For Safety

Please peruse safety instruction below to use mightyZAP safely. Please kindly note that abuse may invalidate your warranty.



1. **Do NOT turn the Rod with excessive force**. It may bring serious product damage.

2. **Do NOT press the Rod when the servo is being operated.** Motor may be damaged(burnt) if higher force than rated force is applied consistently.

3. **Apply proper input voltage** using power supply or correct battery. For instance, apply 4.0~7.4V for 7.4V input product(L7 series) and 7.0~12V for 12V input product(L12 series). The motor may be burnt when 12V is applied to 7.4V input product.

4. **Lifespan of motor can be varied according to the load and duty cycle and etc.**

    1) **Use under rated force.** For instance, rated force of L12-20PT-3 is 20N(approx. 2kg). That is, lifespan of L12-20PT-3 can be maximized when it is used less than 20N force condition. The lower load comparing to rated force, the longer lifespan of the motor.

    2) **Use under 50% of Duty Cycle** : If DC motor operates continuously without any interval (rest), motor will be overloaded and overload protection feature will cut off the power of servo motor. Therefore, user should consider “Duty cycle” which means the percentage of operating time against interval time. In other words, 50% duty cycle means that motor should rest 50% of time when motor operate during 50%

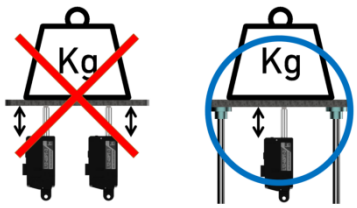
of time to manage motor lifespan more efficiently. Use under 50% of duty cycle for optimized lifespan. The less duty cycle, the longer lifespan.

3) **"Force Off" feature when servo motor is in standby mode may prolong the lifespan of servo motor.** However, this feature can be used when there is no problem in your system even if rod position is changed due to external force because "force off" makes servo force is released.

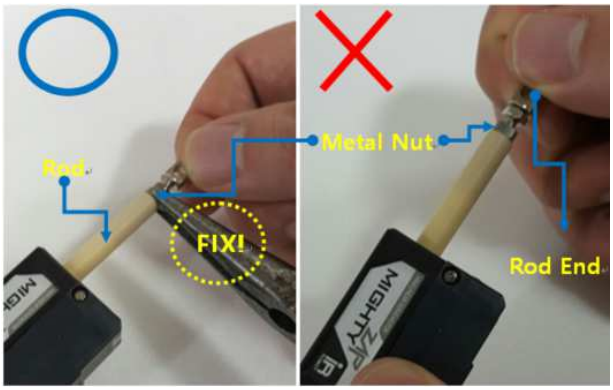
5. **Make sure not to damage servo motor case** such as perforation when you apply your own brackets to fix the servo on your system.

6. **Position command within mechanical limit :** There should be mechanical limit which servo rod can move when user install servo motor in their application. Make sure that positional command should be made within user's mechanical limit. It is too common to mention, but we could see this mistake from time to time. If positional command is out of mechanical limit, servo will be overloaded at certain point of time and power will be cut off to protect the servo due to overload protection feature. (if overload protection is inactivated by user, motor will be not be protected.) Considering precise position control, make sure to re-check this matter when servo is applied.

7. **It is strictly banned to use multiple qty actuators for single objective.** Due to DC motor characteristic, each actuator's speed can be slightly varied even if they are same model and goal position is same. (may cause overload to one of actuator)



8. **Fix the metal nut when screw the Rod-end onto the Rod using a long-nose or spanner**. : The metal nut is inserted onto the end of Rod with Loctite. When you screw rod-end onto the metal nut or you have your own rod-end, please make sure to fix the metal nut with tools like a long-nose. Unless you fix the metal nut with a tool, rod can be transformed slightly due to excessive power. This slight transformation may make some interruption between the rod & servo case which means useless load will be applied to the motor. In serious case, motor can be burnt and the lifespan of motor will be shortened at least. There would not be any problem if you use a tool to fix the metal nut.



9. **Use properly "Overload protection" feature to protect the servo and your system from damage.** Overload protection feature is activated from the factory, and for other protection setting, if necessary, set **"Alarm shutdown" feature** according to your system's condition.

10. **Do NOT touch the servo case right after servo operation**. It may hot.
11. **Keep away from water, humidity, dust and oil**.
12. It is designed for indoor purpose. **Do not use in outdoor.**
13. **Keep out of reach of children. Keep hands off when servo motor operates to avoid unexpected injury.**

1.3. For Storage

Do NOT store/use servo motor under below extreme condition. It may cause malfunction or damage of product.

- Direct light and High temperature more than 70 ˚C or Low temperature lower than minus 20˚C.
- Highly Humid space
- Space having Vibrating condition
- Space having Dust
- Space causing Electrostatic

Important Note : Constant load / Overload Protection / Force Off Function

Overload protection (Overload shutdown) feature is to prevent overload condition which greatly affect the service life and to motor burnout. Please read the following for proper protection and use it according to the conditions.

About mechanism of overload protection (overload shutdown) and terminology

- The overload protection feature will be activated when the motor exceeds 50% of duty cycle, the cumulative operating time is over 30 seconds without rest. Under these conditions, system will be shutdown.
- Duty cycle is the ratio of the time which motor is actually driven against
- Duty cycle 50% means that 50% of the time should be restored if the motor runs 50% of the time.
- Motor operating time includes the time which the motor rotates/moves normally as well as the time which motor draws current by stuck condition without motor rotating.

Overload condition and Shutdown case

For a better understanding, here are some examples of overload conditions and shutdown.

- When Servo actuator is continuously operated for more than 30 seconds without rest / Shutdown after 30 seconds continuous operation.
- When Goal position command is made outside the application's mechanical limit range / Shutdown after 30 seconds continuous operation.
- When stuck condition maintains more than 30 sec / Shutdown after 30 seconds continuous operation.
- When duty cycle is more than 50%. The more severe the duty cycle conditions are, the faster the shutdown to be made, and the lower than 50% duty cycle, the later the shutdown to be made.

Overload shutdown Disable

The overload shutdown function protects the servo actuator under overload conditions. Depending on the application, there are applications that need to be operated under severe conditions, even if the overall mechanism protection is more important than the servo actuator, or even if the life of the servo actuator is shortened. For this case, shutdown function can be disabled through mightZAP servo manager software so that shutdown does not work under overload situation.

Recovery after overload shutdown

Since the communication line is still functioning after shutdown, it can be restored to the initial state by “Restart” command or by reconnecting power. Be sure to remove the cause of overload before restoration.

Exception and Cautions

Exception) Overload shutdown does not work when the actual load is heavier than the rated power even though it is an overload environment. However, in order to manage the lifespan of actuator , please be careful not to put heavier load than the rated load.

Caution 1) When the spring is installed between the application and the actuator, or it is installed in Z axis (the direction of gravity), it makes external force to the actuator. Under these conditions, the servo actuator operates slightly, but continuously to keep its position. If this condition persists, actuator may invoke Overload shutdown in some cases. To prevent this, use Force Off command while external power is applied.

Caution 2) Each time overload shutdown is executed, it is not big but it damages the motor. Therefore, the cause of overload must be removed after the first overload shutdown so that shutdown does not occur frequently.

Overload Protection Function

Using Overload protection function, we can protect motor damage from overload condition.
Overload protection function is being activated from the factory. When it is activated, motor power will be cut off in case of overload condition to protect the servo actuator

The easiest way to set(activate/inactivate) Overload protection function is to connect servo with Servo Manager Software using IR-USB01 PC USB interface. Go to "Shutdown Alarm Setting" and simply click(activate) "Overload Error".

Another method to set Overload protection is to use a Command packet. You can send "Store data" command to the address(0X12) which is the address for Alarm Shutdown. Set bit 5 (see below) for Overload error from to "1"(Overload activation), then send "Store data" command to the servo motor. Servo force will be cut off under overload condition if the bit is set at "1".(1= Overload protection activation / 0=Inactivation)

Error	bit
RESERVED	7
Instruction Error	6
Overload Error	5
Checksum Error	4
Range Error	3
Stroke Limit Error	1
Input Voltage Error	0

Refer to below example for"Store Data" command.

Command Packet

HEADER	ID	Size	Command	Factor #1	Factor #2	Checksum
				Address	Data	
0xFFFFFF	0x00	0x04	0xF3	0x12	0x20	0xD6

: Command packet which designates Servo ID to '0'(0x00)

Caution **Use within Rated Force**

For proper performance and better lifespan of mightZAP, it is stronlgy requested to use it within the rated force range.

Force Off Function

- When servo stops after moving position, servo is still working to stick to its position.
- So, if the period of time for stop position is much longer than moving time, to relieve the motor, you can use "Force Off" function so that servo holds its position only with mechanical friction(Self-lock) under power-off status.
- Under force off condition, communication is still alive while motor power is off, so servo will move again when servo gets new position command without giving "Force ON" command.
- Please see Self-lock force for each servo motor below. (Below chart shows self-lock force of 30mm stroke mightyZAP only. Refer to the separate specification for 41/56mm stroke version's Self lock force.)

Rated Force	Self-Lock Force
20N Lineup	10N
40N Lineup	40N
64N Lineup	64N
100N Lineup	100N

For Force Off, send 0x00 as a "Store data" command to the address(0X80) which is the address for Force ON/OFF. (For Force ON, send 0x01)

Refer to below example for "Store Data" command.

Command Packet

HEADER	ID	Size	Command	Factor #1	Factor #2	Checksum
				Address	Data	
0xFFFFFF	0x00	0x04	0xF3	0x80	0x00	0x88

- Command packet which designates Servo ID to '0'(0x00).

Under Force Off status, if user send "Goal Position" command, it is not necessary to send Force ON packet additionally because "Goal Position" command already includes "Force On" packet in it. .

2 Basic Information

2.1. Component

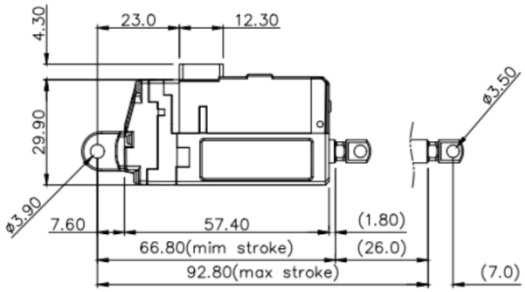
1 Hinge Base 1pc
2 Hinge 1pc
3 M3 NUT 2pcs
4 Hinge Shaft 1pc
5 Rod End Tip 1pc
6 M2.5x6 Screws 3pcs
7 Wire(PF Version) : 4Pin Molex to Molex (RS-485)
8 Wire(PT Version) : 3Pin Molex to Molex (TTL) & S-02 to Molex (PWM)

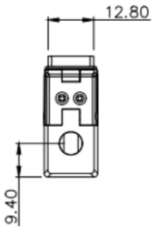
#3 M3 nut can be used to fix the hinge and hinge base. Also, M3 nut can be used between rod-end and rod-end tip as a stopper.

2.2. Dimension

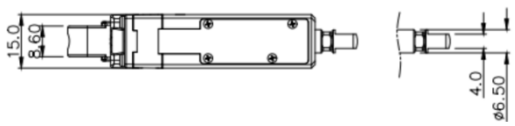
Please refer to detailed dimension from 3D drawing at our website. (www.irrobot.com →Digital Archives →mightyZAP)

[26mm Stroke Lineup - Economical]

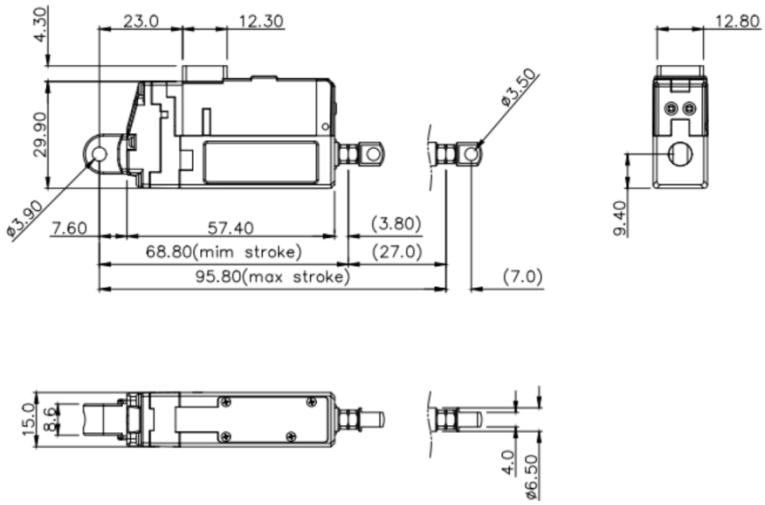




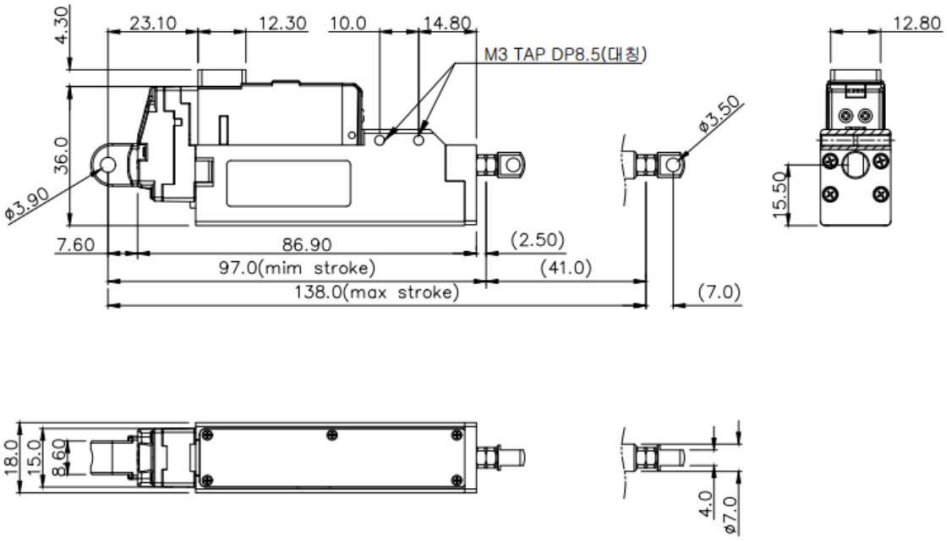
**** For 30mm stroke actuator, Factory default Stroke range sets at 27mm, and user may extend it to 30mm using Servo manager software if necessary. (for better mechanical stability, 27mm is recommended.)**



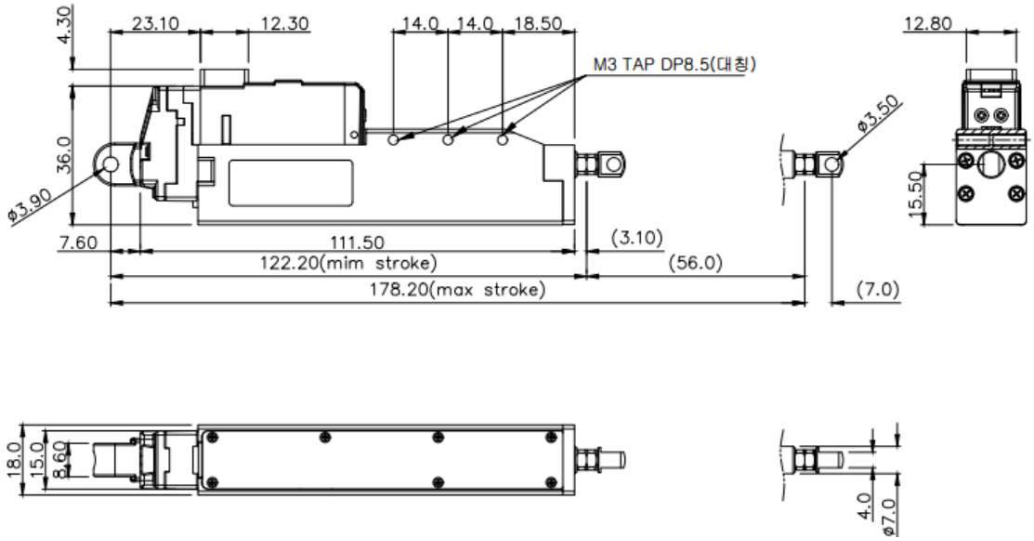
[27mm Stroke Lineup - Premium]



[41mm Stroke Lineup]



[56mm Stroke Lineup]



2.3. Specification

Input Voltage	Commu-nication	Force 20N / Stroke	Force 20N / Stroke				Force 30~40N / Stroke			
		27mm	27mm	41mm	56mm	96mm	27mm	41mm	56mm	96mm
12V	RS-485	L12-12F-3	L12-20F-3	L12-20F-4	L12-20F-6	L12-20F-10	L12-40F-3	L12-30F-4	L12-30F-6	L12-30F-10
	TTL/PWM	L12-12PT-3	L12-20PT-3	L12-20PT-4	L12-20PT-6	L12-20PT-10	L12-40PT-3	L12-30PT-4	L12-30PT-6	L12-30PT-10
7.4V		-	L7-20PT-3	-	-	-	L7-40PT-3	-	-	-
Rated Force / Max. Speed (No Load)		12N /120.0mm/s	20N /80.0mm/s	17N /80.0mm/s			40N /28.0mm/s	31N/ 28.0mm/s		
Self Lock Force / Z Axis Force		N/A	7N/7N				40N/30N	31N/23N		
Gear Ratio / Gear Type / Rod Type		10:1 / Engineering Plastic Gears / Metal Alloy Rod								
Motor Type / Watt / Duty Cycle		Coreless Motor / 26W / 50%								

Input Voltage	Commu-nication	Force 50N~60N / Stroke			Force 80N~100N / Stroke		
		27mm	41mm	56mm	27mm	41mm	56mm
12V	RS-485	L12-64F-3	L12-50F-4	L12-50F-6	L12-100F-3	L12-80F-4	L12-80F-6
	TTL/PWM	L12-64PT-3	L12-50PT-4	L12-50PT-6	L12-100PT-3	L12-80PT-4	L12-80PT-6
7.4V		L7-64PT-3	-	-	L7-100PT-3	-	-
Rated Force / Max. Speed(No Load)		64N / 10.5mm/s	50N / 10.5mm/s		100N / 7.7mm/s	78N / 7.7mm/s	
Self Lock Force / Z Axis Force		64N / 48N	50N / 37.5N		100N / 75N	78N / 58.5N	
Gear Ratio		30:1			50:1		
Gear Type / Rod Type		4 Metal & 2 Engineering Plastic Gears / Metal Alloy Rod					
Motor Type / Watt / Duty Cycle		Coreless Motor / 26W / 50%					

Positional Accuracy	27mm /41mm Stroke : 0.05mm (50μm) 56mm Stroke : 0.07mm (70μm) 96mm Stroke : 0.09mm (90μm)						Micro Controller	32bit ARM Core, 4096 Resolution (A/D converter)
Mechanical Backlash	0.03mm (30μm)						Pulse Range	900μs(Retracted)-1500 μs(Center)-2100μs (Extended)
Position Sensor	10KΩ linear Potentiometer						Parameter Setting	Programmable via PC Software
Input Voltage	7.4V or 12.1V (Rated)						Ingress Protection	IP-54 (Dust & Water Tight)
Current Consumption		7.4V			12V			
		Idle	Rated	Stall	Idle	Rated	Stall	
	Economical	30mA	250mA	0.48A	30mA	140mA	0.3A	
	Premium	30mA	280mA	2.4A	20mA	380mA	1.5A	
Size / Weight (Excluding rod-end & hinge)	27mm	57.5(L)x29.9(W)x15(H)mm /49~52g						
	41mm	86.9(L)x36(W)x18(H)mm / 96~99g						
	56mm	111.5(L)x36(W)x18(H)mm / 124~127g						
	96mm	151.5(L)x36(W)x18(H)mm / 177g						
Audible Noise	Approx. 50db at 1m						Operating Temp.	-10℃ ~ 60℃
Communication	Economical	RS-485 or TTL/PWM					Standard Accessory	1 Mounting Bracket, 2 types Rod end (Detachable linkage and Metal nut(M2.5) type), Wire Harnesses
	Premium	RS-485 or TTL/PWM						
LED Indication	7 Error Indications (Overload, Checksum, Data Range, Overheat, Stroke Limit, Input voltage, Instruction Error)						Wire Harness (Molex to S-02 PWM wire to be packed in Economical version only)	PWM/TTL(PT version) : Molex to S-02 and Molex to Molex Type (Molex 50-37-5033, 3pins) / 200mm length, 0.08×60(22AWG) or RS485(F version) : Molex to Molex Type (Molex 0510650400, 4pins) / 200mm length, 0.08×60(22AWG)

3 Application

Factory Automation

- Better Replacement of Pneumatic Cylinder
- Real-Time Automatic Width Adjustment Conveyer
- Real-Time Automatic Product Alignment (Up/Down or Left/Right)
- Automatic Value Control (oil or water)
- Automatic Dispensing with Syringe
- Automatic Clamping System
- Fitting or Adjusting Distance
- Pick & Place
- In & out / Extension & retraction
- Open & Closing (On-Off)
- Change of Direction
  Hexapod/Tripod movement

Production & Test JIGs

- Hole Punching Jig
- Hole Inspection Jig
- Switch Inspection Jig
- Touch Panel Inspection Jig
- PC Board Testing Jig

Robotics

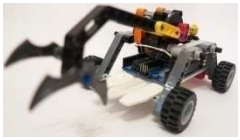
- Robot Joints
- Robot Grippers
- Linear Control Parts of Surgical Robot

UAV / Professional Drone

- Fixed wing (Aileron/Elevator/Throttle/Flap/Air Brake/ Rudder/ Throttle)
- Helicopter (Swash Plate Control/Rudder)
- Multicopter (Retract, Dropping Device)
- Linear control parts for Military products
- Pan/Tilt Camera control

Medical / Lab Equipment

- Linear position control for Medical Devices (HIFU, etc)
- Camera or Laser Focusing Control
- Laboratory Test Equipment



Education / Hobby

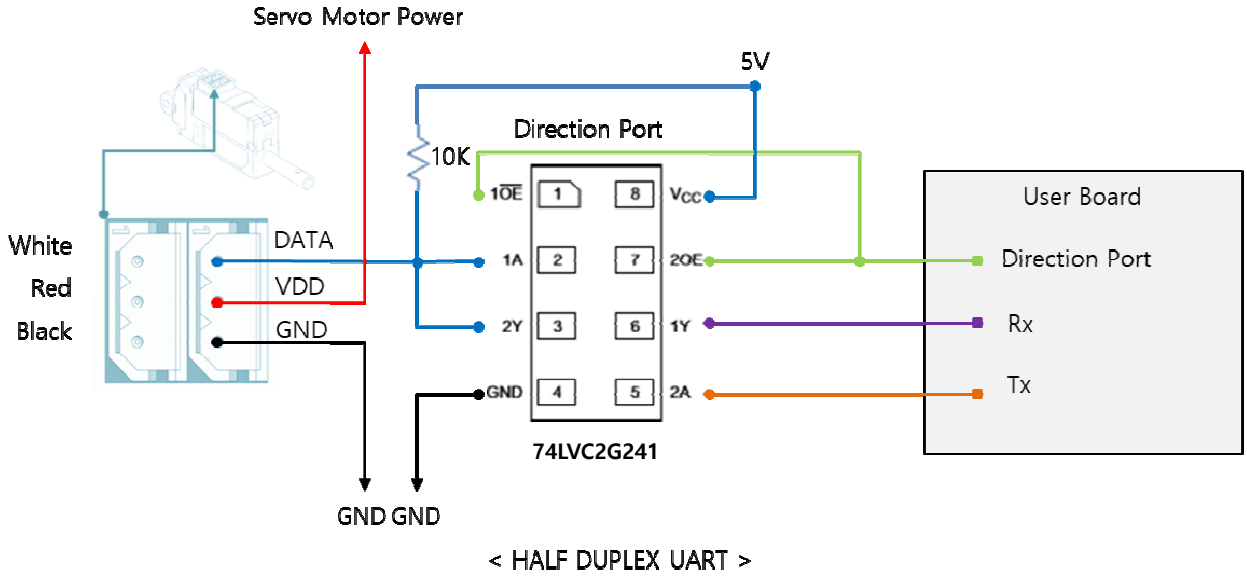
- 3D Printer
- Arduino or Rapsberry Pie Control
- Maker's DIY Project

4 Servo Control

4.1. Circuit Connection

mightyZAP supports both data communication(Half Duplux TTL) as well as simple pulse(PWM) control. For the control under data communication, UART signal of main board should be converted into Half Duplex Type signal. Conversion circuit will be as below.

■ TTL/PWM(3Pin Connector-Model L(D)xx-xxPT-x Series)



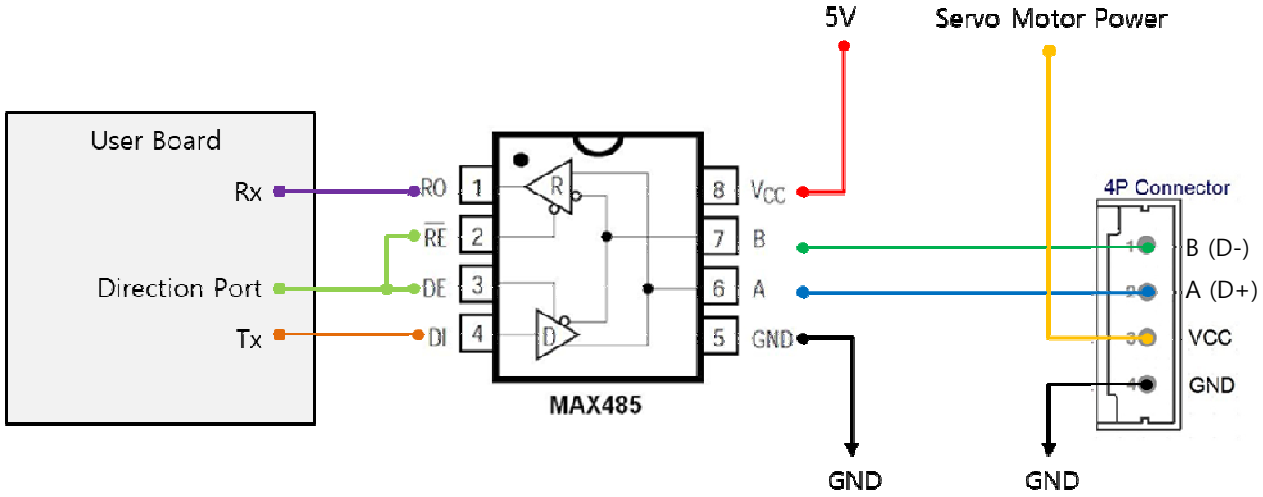
The direction of data signal for TxD and RxD of TTL level will be determined according to the level of direction_port as below.

- The level of "direction_port" is LOW :Data signal will be inputted to RxD.
- The level of "direction_port" is HIGH :TxD signal will be outputted as Data.

■ RS-485(4Pin Connector - Model Lxx-xxF-x Series)

Model Lxx-xxF-x Series uses RS-485 communication. Pin map and Conversion circuit will be as below.

PIN NUMBER(COLOR)	PIN NAME	FUNCTION(RS485)
1(Yellow)	D-	RS485 –
2(White)	D+	RS485 +
3(Red)	VCC	Power +
4(Black)	GND	Power -



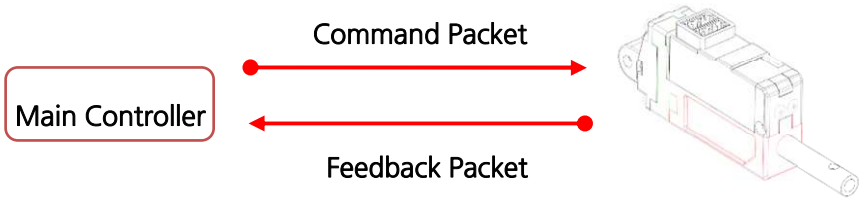
※If the power is supplied from outside, you can connect to 485 D+, 485 D- only.

You can convert TX and RX mode by controlling “Direction_Port pin” in above circuit.

- The level of "direction_port" is LOW : Data signal will be inputted to RxD.
- The level of "direction_port" is HIGH : TxD signal will be outputted as Data

4.2. Communication

mightyZAP and your main controller will communicate by exchanging data packet. The sorts of packet are Command packet (Main controller to mightyZAP) and Feedback packet(mightZAP to your main controller)



(1) Specification

① Communication specification

- **2 Mode in One** (Pulse / Data Mode Auto-Switching)
  mightyZAP will automatically recognize the input signal between data mode and pulse mode.

- **Data Mode**
  Asynchronous Serial communication (8 bit, 1 Stop bit, None Parity)

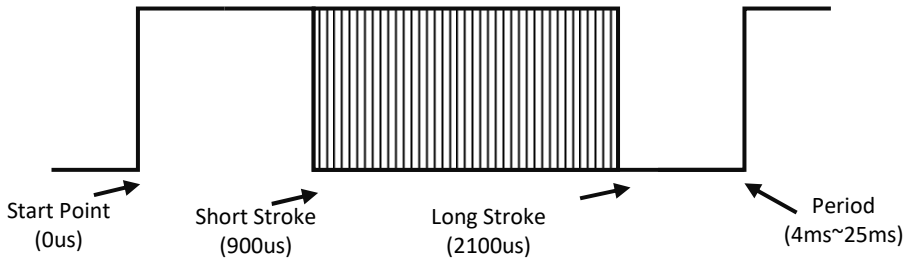
Item	Spec
Structure	Half-duplex UART
Baud Rate	57600bps(default)
Data Size	8bit
Parity	non-parity
Stop Bit	One bit

CAUTION!

- ***mightyZAP uses half duplex communication, and need to put proper delay time to prevent communication error.***
- ***Recommendable delay time is 5msec for data write, 10msec for data read.***
- ***Otherwise, there can be communication collision and motor failure.***
- ***Above delay time is not minimum, but proper delay time for safety.***

- **Pulse Mode**

  PPM(Pulse Position Modulation) Compatible [Radio-Control Servo Pulse Mode]
  (900us(Retracted)~1500 us(Center)~2100 us(Fully Extended)



※ Short stroke : Retract stroke / Long stroke : Extend stroke

② Data specification

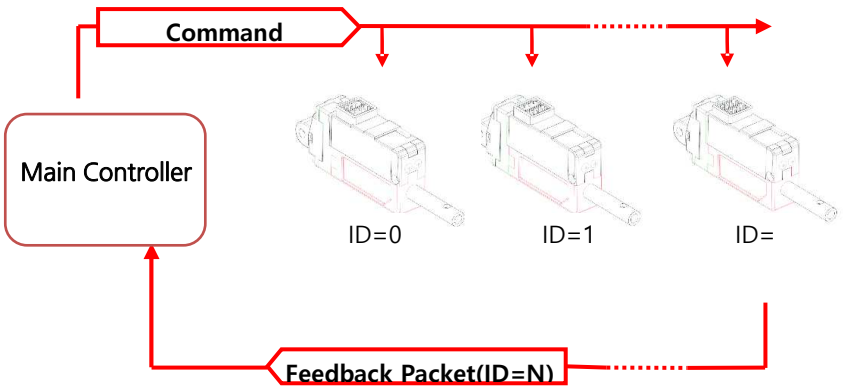
Data range is basically determined as below in both Data and Pulse modes. All factors are changeable & programmable.

Rod Stroke	Data Mode	Pulse Mode
Short Stroke	0	900us
Half Stroke	2047	1500us
Long Stroke (for 27mm)	3686 **	2100us
Long Stroke (for 30mm)	4095	2100us

****Factory default range : 27mm / Extendable by user to 30mm by Servo Manager Software.**
(for better mechanical stability, 27mm is recommended.)

③ Daisy-Chain Connection

After receiving Command Packet at multiple qty of mightZAPs, the servo whose ID is N will be operated only. (Only N ID servo will send Feedback packet and execute Command.)



Caution Unique ID

- Each mightZAP servo must have an individual ID to prevent interference between same IDs. Therefore, you need to set individual IDs for each servo in the network node.
- User may assign 253 different IDs and connect 253pcs servos in serial via TTL protocol. For RS-485 protocol, 253 IDs can be assigned, but available serial connection is upto 32pcs servo motors due to RS-485 node regulation.
- As factory default ID is 0, so please assign different, individual IDs for each servo. It will be easier if you assign each ID when you connect each servo in Daisy-chain network one by one.

(2) Packet Description

① Command Packet

It is command packets for servo operation. Its structure and elements are as below.

■ Structure

■ Element

Index	Data	Description
0	**Start Byte 1**	Start Byte 1 (0xFF)
1	**Start Byte 2**	Start Byte 2 (0xFF)
2	**Start Byte 3**	Start Byte 3 (0xFF)
3	**ID**	Servo ID (Range: 1 ~ 253, Broadcast ID: 254, Stand-alone ID: 0)
4	**SIZE**	Packet Size (COMMAND+FACTOR+CHECKSUM)
5	**COMMAND**	Instruction
5+1	**FACTOR #1**	First Parameter
5+m	**FACTOR #m**	"m"th Parameter
5+N	**FACTOR #N**	Last Parameter
5+N+1	**Check Sum**	Check Sum = BinaryInvert(LOWER_BYTE(ID + SIZE + COMMAND + FACTOR#1 + … + FACTOR#N))

■ Element Description

1. HEADER (3Byte)

- Code to recognize Packet start : 0xFFFFFF

2. ID (1Byte)

- The ID is an unique number of each servo to support Daisy Chain connection.
- Factory default value(ID) is 0.
- In case of ID = 0, it will be deemed as stand-alone(single) connection and communicate regardless of ID. (except for Echo, Load Data)
- In case of ID = 1 ~253, ID "N" which is stored in the servo will be operated.
- In case of ID = 254 (0xFE), it is operated under "Broadcasting Mode (move all servos)" and Feedback Packet does not work.

3. SIZE (1Byte)

- Packet length in Byte unit
- Data counting value after "Size" data (COMMAND+FACTOR+CHECKSUM)
- That is, Size value = Number of byte of "Factor" + 2

4. **COMMAND (1Byte)**

- Command codes defining the purpose of Packet

Function	CODE	Description
Echo	0xF1	Feedback Packet Reception
Load Data	0xF2	Send "Address" and get feedback of Data
Store Data	0xF3	Send "Address" and "Data". Then Save.
Send Data	0xF4	Send "Address" and "Data" for temporary storage
Execution	0xF5	Execute temporarily stored data that is made by SendData.
Factory Reset	0xF6	Reset to Factory default parameter value
Restart	0xF8	Restart servo system
Symmetric Store	0x73	Store data in the same address of multiple qty servos.

5. **FACTOR**

- Additional Packet factor according to Command

6. **CHECKSUM**

Verification data to check omission and any changes of Packet data. The interaction formula will be as below.

- Checksum = BinaryInvert(LOWER_BYTE(ID + SIZE + COMMAND + FACTOR#1 + … + FACTOR#N))

② Feedback Packet

After reception of command packet, servo sends Feedback packet including requested information. Its structure and factors are as below.

■ Structure

■ Element

Index	Data	Description
0	**Start Byte 1**	Start Byte 1 (0xFF)
1	**Start Byte 2**	Start Byte 2 (0xFF)
2	**Start Byte 3**	Start Byte 3 (0xFF)
3	**ID**	Servo ID (Range: 1 ~ 253, Broadcast ID: 254, Stand-alone ID: 0)
4	**SIZE**	Packet Size (COMMAND+FACTOR+CHECKSUM)
5	**ERROR**	Error Code
5+1	**FACTOR #1**	First Parameter
5+m	**FACTOR #m**	"m"th Parameter
5+N	**FACTOR #N**	Last Parameter
5+N+1	**Check Sum**	Check Sum = BinaryInvert(LOWER_BYTE(ID + SIZE + ERROR + FACTOR#1 + … + FACTOR#N))

Element Description

1. HEADER (3Byte)

- Recognizing "Packet start" code. 0xFFFFFF

2. ID (1Byte)

- Individual ID number for each servo (1 ~253)

3. SIZE (1Byte)

- Packet length in Byte unit
- Data counting value after "Size" data (ERROR+FACTOR+CHECKSUM)
- That is, Size value = Number of byte of "Factor" + 2

4. ERROR (1Byte)

- Error status during operation for each bit

Error	bit	Description	LED
RESERVED	7	TBD	LED Off
Instruction Error	6	In case that undefined instruction is sent, or Execution command is sent without Send Data command, it will be set as "1".	White
Overload Error	5	In case that current load cannot be controlled with the designated maximum force, it will be set as "1".	Cyan
Checksum Error	4	In case that transferred Checksum packet value is not correct, it will be set as "1".	Magenta
Range Error	3	In case that the command is out of Data Map address range, it will be set as "1".	Blue
Stroke Limit Error	1	In case that the goal position is written out of range between PULL Stroke Limit and PUSH Stroke Limit,it will be set as "1".	Green
Input Voltage Error	0	In case that the input voltage is out of operating voltage range designated in the Control table, it will be set as "1".	Red

5. FACTOR

- Additional Packet factor according to Feedback data.

6. CHECKSUM

Verification data to check omission and any changes of Packet data. The interaction formula will be as below.

- Checksum = BinaryInvert(LOWER_BYTE(ID + SIZE + ERROR + FACTOR#1 + … + FACTOR#N))

(3) Data Map

① Data Memory Map

■ Memory using data (Non-volatile)

- Data to be saved in non-volatile memory which maintains data even after power OFF/ON.
- All data will be reset to default value when Factory Reset command is executed.

Address	Name	Description	Access	Default
0 (0x00)	Model Number(L)	Low byte of model number	R	
1 (0x01)	Model Number(H)	High byte of model number	R	
2 (0x02)	Version of Firmware	Firmware version info.	R	-
3 (0x03)	ID	Servo ID	RW	0 (0x00)
4 (0x04)	Baud Rate	Servo communication speed	RW	32 (0x20)
5 (0x05)	Return Delay Time	Return delay time	RW	250 (0xFA)
6 (0x06)	Short Stroke Limit(L)	Low byte of Retract direction limit value.	RW	0 (0x00)
7 (0x07)	Short Stroke Limit(H)	High byte of Retract direction limit value.	RW	0 (0x00)
8 (0x08)	Long Stroke Limit(L)	Low byte of Extension direction limit value.	RW	102 (0x66)
9 (0x09)	Long Stroke Limit(H)	High byte of Extension direction limit value.	RW	14 (0x0E)
13 (0x0D)	the Highest Limit Voltage	Highest limit of voltage	RW	individual SPEC
14 (0x0E)	Motor Operating Rate(L)	Low byte of max force	RW	255 (0xFF)
15 (0x0F)	Motor Operating Rate (H)	High byte of max force	RW	3 (0x03)
16 (0x10)	Feedback Return Mode	Feedback return mode	RW	1 (0x01)
17 (0x11)	Alarm LED	Alarm LED function	RW	36 (0x24)
18 (0x12)	Alarm Shutdown	Alarm Shut Down function	RW	36 (0x24)
19 (0x13)	Temperature	Servo Motor Temperature	R	
22 (0x16)	Resolution Factor	Resolution setting factor	RW	1 (0x01)
24 (0x18)	Calibration Short Stroke (L)	Low byte of short stroke	R	0 (0x00)
25 (0x19)	Calibration Short Stroke (H)	High byte of short stroke	R	0 (0x00)
26 (0x1A)	Calibration Long Stroke (L)	Low byte of long stroke	R	255 (0xFF)
27 (0x1B)	Calibration Long Stroke (H)	High byte of long stroke	R	15 (0x0F)
28 (0x1C)	Calibration Center Stroke (L)	Low byte of center stroke	RW	255 (0xFF)
29 (0x1D)	Calibration Center Stroke (H)	High byte of center stroke	RW	7 (0x07)
33 (0x21)	Acceleration Rate	Movement Acceleration Rate	RW	individual SPEC
34 (0x22)	Deceleration Rate	Movement Deceleration Rate	RW	individual SPEC
37 (0x25)	D Gain	Derivative Gain	RW	individual SPEC
38 (0x26)	I Gain	Integral Gain	RW	individual SPEC

39 (0x27)	P Gain	Proportional Gain	RW	individual SPEC
40 (0x28)	Short Stroke Pulse Width (L)	Low byte of Retract direction pulse width	RW	132 (0x84)
41 (0x29)	Short Stroke Pulse Width (H)	High byte of Retract direction pulse width	RW	3 (0x03)
42 (0x2A)	Long Stroke Pulse Width (L)	Low byte of Extension direction pulse width	RW	52 (0x34)
43 (0x2B)	Long Stroke Pulse Width (H)	High byte of Extension direction pulse width	RW	8 (0x08)
44 (0x2C)	Middle Stroke Pulse Width (L)	Low byte of middle stroke pulse width	RW	220 (0xDC)
45 (0x2D)	Middle Stroke Pulse Width (H)	High byte of middle stroke pulse width	RW	5 (0x05)
50 (0x32)	Center Difference (L)	Low byte of Zero point adjustment value	RW	255 (0xFF)
51 (0x33)	Center Difference (H)	High byte of Zero point adjustment value	RW	7 (0x07)
52 (0x34)	Punch Initial Value(L)	Low byte of Punch initial value	RW	individual SPEC
53 (0x35)	Punch Initial Value(H)	High byte of Punch initial value	RW	individual SPEC

② Parameter Map

■ Parameter Using Data (Volatile)

- All data to be reset to default value whenever power is On.

Address	Name	Description	Access	Default
0 (0x80)	Force ON/OFF	Force On/ Off	RW	0 (0x00)
1 (0x81)	LED	LED On/Off	RW	0 (0x00)
2 (0x82)	Start Compliance(Short stroke) Margin	Compliance margin of Start point (Retracting direction)	RW	7 (0x07)
3 (0x83)	Start Compliance (Long Stroke) Margin	Compliance margin of Start point (Extension direction)	RW	7 (0x07)
6 (0x86)	Goal Position(L)	Low byte of Goal position value	RW	-
7 (0x87)	Goal Position(H)	High byte of Goal position value	RW	-
12 (0x8C)	Present Position(L)	Low byte of present position value	R	-
13 (0x8D)	Present Position(H)	High byte of present position value	R	-
14 (0x8E)	Present Speed(L)	Low byte of present speed value	R	-
15 (0x8F)	Present Speed(H)	High byte of present speed value	R	-
16 (0x90)	Present Motor Operating Rate (L)	Low byte of present motor operating value	R	-
17 (0x91)	Present Motor Operating Rate (H)	High byte of present motor operating value	R	-
18 (0x92)	Present Voltage	Current voltage	R	-

20 (0x94)	Received Data	Reception status for "Send Data"	R	0 (0x00)
22 (0x96)	Moving	Moving status	R	0 (0x00)
23 (0x97)	Lock	Lock for Non-volatile Memory	RW	0 (0x00)
24 (0x98)	Punch(L)	Low byte of Punch value	RW	Punch Initial Value(L)
25 (0x99)	Punch(H)	High byte of Punch value	RW	Punch Initial Value(H)
26 (0x9A)	End Compliance Margin	End Point Compliance margin	RW	Varied Per model

(4) Data Description

1) Model Number

- The model number of MightyZAP
- "Read" only to discriminate & recognize concerned model

2) Version of Firmware

Check if current firmware is the latest version.

3) ID

ID to discriminate each servo. Different IDs should be assigned in Daisy-Chain system.

- In case of ID = 0, it will be deemed as stand-alone(single) connection and communicate regardless of ID. (except for Echo, Load Data)
- In case of ID = 1 ~253, ID "N" which is stored in the servo will be operated.
- In case of ID = 254 (0xFE), it is operated under "Broadcasting Mode (move all servos)" and Feedback Packet does not work.

4) Baud Rate

- Determining communication speed. Default value is 57600bps
- Servo system should be rebooted to apply changed baud rate to the servo.

[Setting Value]

Value	Baud Rate(bps)
16 (0x10)	115200
32 (0x20)	57600
64 (0x40)	19200
128 (0x80)	9600

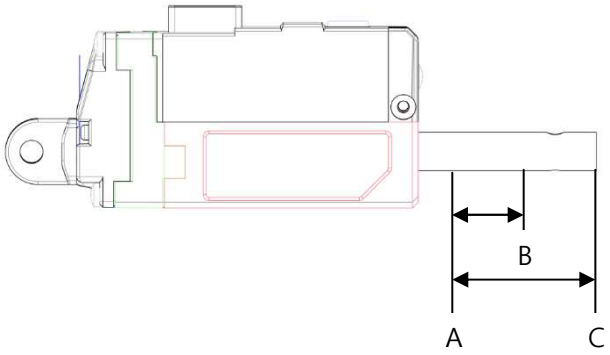
5) Return Delay Time

- Delay time to receive feedback packet after sending Command packet. (Unit : μs)

6) Stroke Limit

Stroke limit between Short Stroke (A) and Long Stroke (C) which is the max/min. value of Goal Position.

(Range: 0 ~ 4095)



7) The Highest / Lowest Limit Voltage

Max/Min. value of input voltage (unit : 0.1V)

For the servo with 7.4V input voltage :, 4V ~ 8.6V

For the servo with 12V input voltage : 7V ~ 13V

8) Motor Operating Rate (Old “Max Force”)

- Maximum Motor Operating value.[400~1023]
- 0 manes force OFF and 1023 is max force. (Value less than 400 will be displaced to 400.)
- When this value is decreased (increased), motor speed and STALL force will also be decreased (increased) (RATED Force will not be changed.)
- Changed value to be applied after power reboot or sending “Restart” command.

9) Feedback Return Mode

Feedback packet return mode after receipt of Command Packet

Mode	Feedback Packet Return or NOT
0	Do NOT sending Feedback packet for all Commands. (Except for Echo command)
1	Sending Feedback packet only for Load Data Command.
2	Sending Feedback packet for all Commands.

Under Broadcast ID(0xFE) mode, feedback packet will NOT be sent regardless values of Feedback Return Mode.

10) Alarm LED

If concerned bit is set as "1" when error occurs, error LED indication will be activated. (1 : activate, 0: deactivate)

Error	bit	LED Indicate
RESERVED	7	LED Off
Instruction Error	6	White
Overload Error	5	Cyan
Checksum Error	4	Magenta
Range Error	3	Blue
Stroke Limit Error	1	Green
Input Voltage Error	0	Red

In case that different errors are made at the same time, lower bit has a priority.

If Error is resolved, alarm will be deactivated after 2 sec and turns to previous status.

11) Alarm Shutdown

Force will be OFF if concerned bit is set as "1" when error occurs. (1 : activate, 0: deactivate)

Error	bit
RESERVED	7
Instruction Error	6
Overload Error	5
Checksum Error	4
Range Error	3
Stroke Limit Error	1
Input Voltage Error	0

Overload protection Alarm shutdown feature is activated from the factory. Other shutdown features can be enabled/disabled by user using mightyZAP manager software according to their wish.

12) Resolution Factor

Changing motor resolution.

Factor	Resolution	Long Stroke (for 30mm)	Long Stroke (for 27mm)
1	4096	4095	3686
2	2048	2047	1843
3	1024	1023	922
4	512	511	461

For 30mm stroke servo motor, Factory default Stroke range sets at 27mm, and user may extend it to 30mm using Servo manager software if necessary. (for better mechanical stability, 27mm is recommended.)

13) Calibration Stroke

- Calibration Short Stroke : Short Stroke calibration value, Short Stroke Calibration value which is set at the factory will be saved.
- Calibration Long Stroke : Long Stroke calibration value, Long Stroke Calibration value which is set at the factory will be saved.
- Calibration Center Stroke : Half Stroke calibration value, Half Stroke Calibration value which is set at the factory will be saved.

14) Short / Long Stroke Pulse Width

Pulse width setting for retract / extend position(Unit : µs). Setting Range is 900us ~ 2100us.

Rod Stroke	Goal Position (Based on Resolution 4096)	Default Setting
Short Stroke	0	900us
Half Stroke	2047	1500us
Long Stroke	4095	2100us

15) Center Difference

Zero point adjusting value of the center point. Setting range is within the Stroke Limit.

16) Force ON/OFF

Setting for Force On and OFF (0 : OFF, 1 : ON)

value	Description
0	Cut off power to the motor and Force is OFF.
1	Power to be supplied to the motor and Force is ON.

migtyZAP keeps its position due to mechanical design even after motor power is off. For instance, mightyZAP having more than 31N/40N force, rod sticks to its position firmly when motor power is off.
So, in case servo motor needs to keep certain position (if mechanical frictional

17) LED

Control LED when there is no Error indication.

bit	Description
0	LED Disable (All LEDs will be Off when it is 1.)
1	RED LED Control
2	GREEN LED Control
3	BLUE LED Control

18) Stroke Compliance Margin

Start Compliance Margin (Recommended margin value : 7 ~ 15)

- Minimum margin value for the servo actuator to start position movement.
- For example, if the compliance margin is 7 and the current position value is 400, motor start will be made when positional value between 407(400+7) and 393(400-7) is set.
- Likewise, when the positional change occurs by more than +/-7(out of 393~407) from the present position value due to physical external pressure or electrical noise, the motor starts to run to compensate position.
- For this reason, the larger this value means more stable operation without jittering even in the environment where the external pressure, electrical noise, or the clearance increases, but the sensitivity to drive to the desired position may be reduced. In other words, generally, increasing this value increases durability, and reducing it increases precision.
- This value must be equal to or greater than the "End compliance margin value" described below. Setting it to a lower value may cause an error.

End Compliance Margin (Recommended margin value : 4)

- Minimum margin value for the servo actuator to complete position movement.
- For example, if actuator is instructed to move to a position value of 400, and assuming that it cannot physically stop at a position value of 400 exactly due to software & mechanical clearance, acceleration, etc. of the servo, End compliance margin will be a criteria to judge if the positional command has been performed properly. If this value is set to 4 and the position command value is set to 400, actuator judges that positional movement has been made properly when it reaches within 396~404 range and then stop movement.
- If this value is increased for stable operation, you should not increase it beyond the "Start Compliance Margin" value which is described above, and if this value is decreased too much to increase the accuracy, it may bring adverse effect such as jitter.

19) Goal Position

Goal position value which is desired position value to move.

Rod Stroke	Resolution		
	4096	2048	1024
Short Stroke	0	0	0
Half Stroke	2047	1023	511
Long Stroke	4095	2047	1023

Resolution can be adjusted according to Resolution Factor setting.

20) Present Position

- Current Position value of stroke.
- Range is between 0~4095 and the value will be varied according to Resolution Factor setting.

21) Present Motor Operating Rate (Old “Present Load”)

- Current Motor operating rate value
- To be shown in the range of 0~2047
- Between 0~1023: Motor operating rate on short stoke direction (retract direction).
  Between 1024~2047: Motor operating rate on long stoke direction (extend direction).

22) Present Voltage

- Current input voltage. The unit is 0.1V
- For instance, 74 means 7.4V

23) Present Temperature

- Current temperature of servo inner space. The unit is 1 ˚C.
- For instance, 85 means 85℃.

24) Received Data

Send Data command reception status for Execution command.

Value	Description
0	Send Data command is NOT received.
1	Send Data command is received.

25) Moving

Moving status

Value	Description
0	Goal Position command execution is completed.
1	Goal Position command execution is under operation.

26) Lock

Value	Description
0	Non-volatile Memory Modification available
1	Non-volatile Memory Modification Unavailable

27) Punch

- Minimum current to the motor during the operation.
- Higher Punch value increase stall torque, but chattering can be made if punch value is too high.

(5) Command Example Packet

1) Echo Receiving Feedback Packet

Command Packet

HEADER	ID	Size	Command	Checksum
0xFFFFFF	0x00	0x02	0xF1	0x0C

- Command packet to recognize status of servo connection.

Feedback Packet

HEADER	ID	Size	Error	Checksum
0xFFFFFF	0x00	0x02	0x00	0xFD

- Feedback packet to inform status of servo connection. (Including Error information)

2) Factory Reset Reset to factory default parameter value.

Command Packet

HEADER	ID	Size	Command	Factor Option	Checksum
0xFFFFFF	0x01	0x03	0xF6	0x01	0x04

- Basic parameter (Memory & Parameter) to be reset to Default value. Additional Reset to be determined according to options.
- Servo ID to be reset to 0(ID Default) and Baud Rate to be maintained current status.
- If concerned bit is "1", it means Reset. If it is "0", it means Hold.

Option	bit	Description
Servo ID	0	Reset servo ID to 0
Baud Rate	1	Reset to 32 (57600 bps)

Feedback Packet

HEADER	ID	Size	Error	Checksum
0xFFFFFF	0x01	0x02	0x00	0xFC

- Feedback packet informing receipt of Factory reset.

3) Restart Servo system Restart

Command Packet

HEADER	ID	Size	Command	Checksum
0xFFFFFF	0x00	0x02	0xF8	0x05

- Command packet to reboot servo system.

Feedback Packet

HEADER	ID	Size	Error	Checksum
0xFFFFFF	0x00	0x02	0x00	0xFD

- Feedback packet informing receipt of Restart command of servo system

4) **Store Data** Store data after sending Address and Data to set ID, goal position, Force limit, Stroke limit, Speed, Force On/Off and etc.

- **ID change : Change ID'0' into ID '1'(0x01)**

HEADER	ID	Size	Command	Factor #1	Factor #2	Checksum
				Address	Data	
0xFFFFFF	0x00	0x04	0xF3	0x03	0x01	0x04

- ID : Servo motor ID
- Command : Save data at respective address in order.
- Address : the address which servo motor ID is saved. (see (3)Data Map)
- Data : Desired Servo ID (put 0x01 at address 0x03)

- **Goal Position command 1: Command packet to assign goal position to 2047(0x07FF)**

HEADER	ID	Size	Command	Factor #1	Factor #2	Factor #3	Checksum
				Address	Data #1	Data #2	
0xFFFFFF	0x01	0x05	0xF3	0x86	0xFF	0x07	0x7A

- ID : Servo motor ID
- Command : Save data at respective address in order.
- Address : the address which goal position value is saved. (see (3)Data Map)
- Data #1 : Desired goal position's lower byte (address 0x86 : 0xFF)
- Data #2 : Desired goal position's upper byte (address 0x87 : 0x07)

※Goal position value Hex change (decimal number→ hexadecimal number) : 2047 →0x07FF

- **Goal Position command 2 : Command packet to assign goal position to 1000 (0x03E8)**

HEADER	ID	Size	Command	Factor #1	Factor #2	Factor #3	Checksum
				Address	Data #1	Data #2	
0xFFFFFF	0x01	0x05	0xF3	0x86	0xE8	0x03	0x95

- ID : Servo motor ID
- Command : Save data at respective address in order.
- Address : the address which goal position value is saved. (see (3)Data Map)
- Data #1 : Desired goal position's lower byte (address 0x86 : 0xE8)
- Data #2 : Desired goal position's upper byte (address 0x87 : 0x03)

※ Goal position value Hex change (decimal number→hexadecimal number) : 1000 → 0x03E8

- **Motor Operating Rate 1 : Command packet to assign Motor operating rate to 512(0x0200)**

HEADER	ID	Size	Command	Factor #1	Factor #2	Factor #3	Checksum
				Address	Data #1	Data #2	
0xFFFFFF	0x01	0x05	0xF3	0x0E	0x00	0x02	0xF6

- ID : Servo motor ID
- Command : Save data at respective address in order.
- Address : the address which Motor Operating value is saved. (see (3)Data Map)
- Data #1 : Desired Motor Operating value's lower byte (address 0x0E : 0x00)
- Data #2 : Desired Motor Operating value's upper byte (address 0x0F : 0x02)

※ Motor Operating value Hex change(decimal number→hexadecimal number) : 512→0x0200

- **Motor Operating Rate 2: Command packet to assign Motor operating rate to 400(0x0190)**

HEADER	ID	Size	Command	Factor #1	Factor #2	Factor #3	Checksum
				Address	Data #1	Data #2	
0xFFFFFF	0x01	0x05	0xF3	0x0E	0x90	0x01	0x67

- ID : Servo motor ID

- Command : Save data at respective address in order.
- Address : the address which Motor Operating value is saved. (see (3)Data Map)
- Data #1 : Desired Motor Operating value's lower byte (address 0x0E : 0x90)
- Data #2 : Desired Motor Operating value's upper byte (address 0x0F : 0x01)

※ Motor Operating value Hex change(decimal number→hexadecimal number) : 400→0x0190

- **Stroke Limit 1 : Command packet to assign Short Stroke limit to 100(0x0064)**

HEADER	ID	Size	Command	Factor #1	Factor #2	Factor #3	Checksum
				Address	Data #1	Data #2	
0xFFFFFF	0x01	0x05	0xF3	0x06	0x64	0x00	0x9C

- ID : Servo motor ID
- Command : Save data at respective address in order.
- Address : the address which short stroke limit value is saved. (see (3)Data Map)
- Data #1 : Desired Short stroke limit value's lower byte (address 0x06 : 0x64)
- Data #2 : Desired Short stroke limit value's upper byte (address 0x07 : 0x00)

※Stroke limit value Hex change(decimal number→hexadecimal number) : 100 → 0x0064

- **Stroke Limit 2: Command packet to assign Long Stroke limit to 3800(0x0ED8)**

HEADER	ID	Size	Command	Factor #1	Factor #2	Factor #3	Checksum
				Address	Data #1	Data #2	
0xFFFFFF	0x01	0x05	0xF3	0x08	0xD8	0x0E	0x18

- ID : Servo motor ID
- Command : Save data at respective address in order.
- Address : the address which long stroke limit value is saved. (see (3)Data Map)
- Data #1 : Desired Long stroke limit value's lower byte (address 0x08 : 0xD8)
- Data #2 : Desired Long stroke limit value's upper byte (address 0x09 : 0x0E)

※ Stroke limit value Hex change(decimal number→hexadecimal number) : 3800 → 0x0ED8

- **Force On/Off : Command packet to cut off Motor power while communication is alive.**

HEADER	ID	Size	Command	Factor #1	Factor #2	Checksum
				Address	Data #1	
0xFFFFFF	0x01	0x04	0xF3	0x80	0x00	0x87

- ID : Servo motor ID
- Command : Save data at respective address in order.
- Address : the address which Force On/Off value is saved. (see (3)Data Map)
- Data : Desired Force On/Off Data byte (address0x86 : 0x00(Off) / 0x01(On))
- After force-off, automatically Force On when next goal position command is made.

- **Feedback Return Mode 1: Command packet to send Feedback packet for Load Data command only.**

HEADER	ID	Size	Command	Factor #1	Factor #2	Checksum
				Address	Data #1	
0xFFFFFF	0x01	0x04	0xF3	0x10	0x01	0xF6

- ID : Servo motor ID
- Command : Save data at respective address in order.
- Address : the address which Feedback Return Mode value is saved. (see (3)Data Map)
- Data : Feedback Return Mode Data (address 0x10 : 0x01)
  (1: Send Feedback packet only to Load Data(0xF3) Command)

- **Feedback Return Mode 2: Command packet to send Feedback packet for All commands.**

HEADER	ID	Size	Command	Factor #1	Factor #2	Checksum
				Address	Data #1	
0xFFFFFF	0x01	0x04	0xF3	0x10	0x02	0xF5

- ID : Servo motor ID
- Command : Save data at respective address in order.
- Address : the address which Feedback Return Mode value is saved. (see (3)Data Map)
- Data : Feedback Return Mode Data (address 0x10 : 0x02)
  (2: Send Feedback packet to All)

5) Load Data Send address and Get data feedback

- **Present Position : Command packet to read present Position**

Command Packet

HEADER	ID	Size	Command	Factor #1	Factor #2	Checksum
				Address	Length	
0xFFFFFF	0x00	0x04	0xF2	0x8C	0x02	0x7B

- ID : Servo motor ID
- Command : Read byte (equivalent to the Length number) from Address
- Address : Address where present position value is saved. (see (3)Data Map)
- Length: The number of byte to read from Address (present position value consists of 2byte.)

Feedback Packet

HEADER	ID	Size	Error	Factor #1	Factor #2	Checksum
0xFFFFFF	0x00	0x04	0x00	0xFF	0x07	0xF5

- ID : Servo motor ID
- Error : Error indication during operation
- Factor 1 : Present position value's lower byte (ex> 0xff)
- Factor 2 : Present position value's upper byte (ex> 0x07)

※ Present position value Hex change(hexadecimal number → decimal number) : 0x07ff→2047

- **Present Load: Command packet to read present Load.**

Command Packet

HEADER	ID	Size	Command	Factor #1	Factor #2	Checksum
				Address	Length	
0xFFFFFF	0x00	0x04	0xF2	0x90	0x02	0x77

- ID : Servo motor ID
- Command : Read byte (equivalent to the Length number) from Address
- Address : Address where present load value is saved. (see (3)Data Map)
- Length: The number of byte to read from Address (present load value consists of 2byte.)

Feedback Packet

HEADER	ID	Size	Error	Factor #1	Factor #2	Checksum
0xFFFFFF	0x00	0x04	0x00	0xFF	0x03	0xF9

- ID : Servo motor ID
- Error : Error indication during operation
- Factor 1 : Present Load value lower byte (ex> 0xff)
- Factor 2 : Present Load value upper byte (ex> 0x03)

※ Present Load value Hex change(hexadecimal number → decimal number) : 0x03ff→1023

- **Present Volt: Command packet to read present input Voltage**

Command Packet

HEADER	ID	Size	Command	Factor #1	Factor #2	Checksum
				Address	Length	
0xFFFFFF	0x00	0x04	0xF2	0x92	0x01	0x76

- ID : Servo motor ID
- Command : Read byte (equivalent to the Length number) from Address
- Address : Address where present voltage value is saved. (see (3)Data Map)
- Length: The number of byte to read from Address (present voltage value consists of 2byte.)

Feedback Packet

HEADER	ID	Size	Error	Factor #1	Checksum
0xFFFFFF	0x00	0x03	0x00	0x7B	0x81

- ID : Servo motor ID
- Error : Error indication during operation
- Factor 1 : Present Voltage value byte (ex> 0x7B)

※ Present voltage value Hex change(hexadecimal number → decimal number):
0x7B→123(12.3V)

6) Send Data **Send "Address" and "Data", Then temporarily store it.**

Command Packet

HEADER	ID	Size	Command	Factor #1	Factor #2	Factor #3	Checksum
				Address	Data #1	Data #2	
0xFFFFFF	0x01	0x05	0xF4	0x86	0xFF	0x07	0x79

- Command packet for temporary store of goal position as 2047(0x07FF).

Feedback Packet

HEADER	ID	Size	Error	Checksum
0xFFFFFF	0x01	0x02	0x00	0xFC

- Feedback packet informing receipt of temporary store for servo goal position.

7) Execution **Execute temporarily stored data that is made by Send Data.**

Command Packet

HEADER	ID	Size	Command	Checksum
0xFFFFFF	0x01	0x02	0xF5	0x07

- Command packet to execute all temporarily stored data at the same time.

Feedback Packet

HEADER	ID	Size	Error	Checksum
0xFFFFFF	0x01	0x02	0x00	0xFC

- Feedback packet informing receipt of execution command for temporarily stored data.

8) Symmetric Store Save data in the same address of multiple servos.

- Goal Position : Command packet to assign multiple servo's goal positions.
  - Servo ID 1 : 1023(0x03FF), Servo ID 2 : 2047(0x07FF)

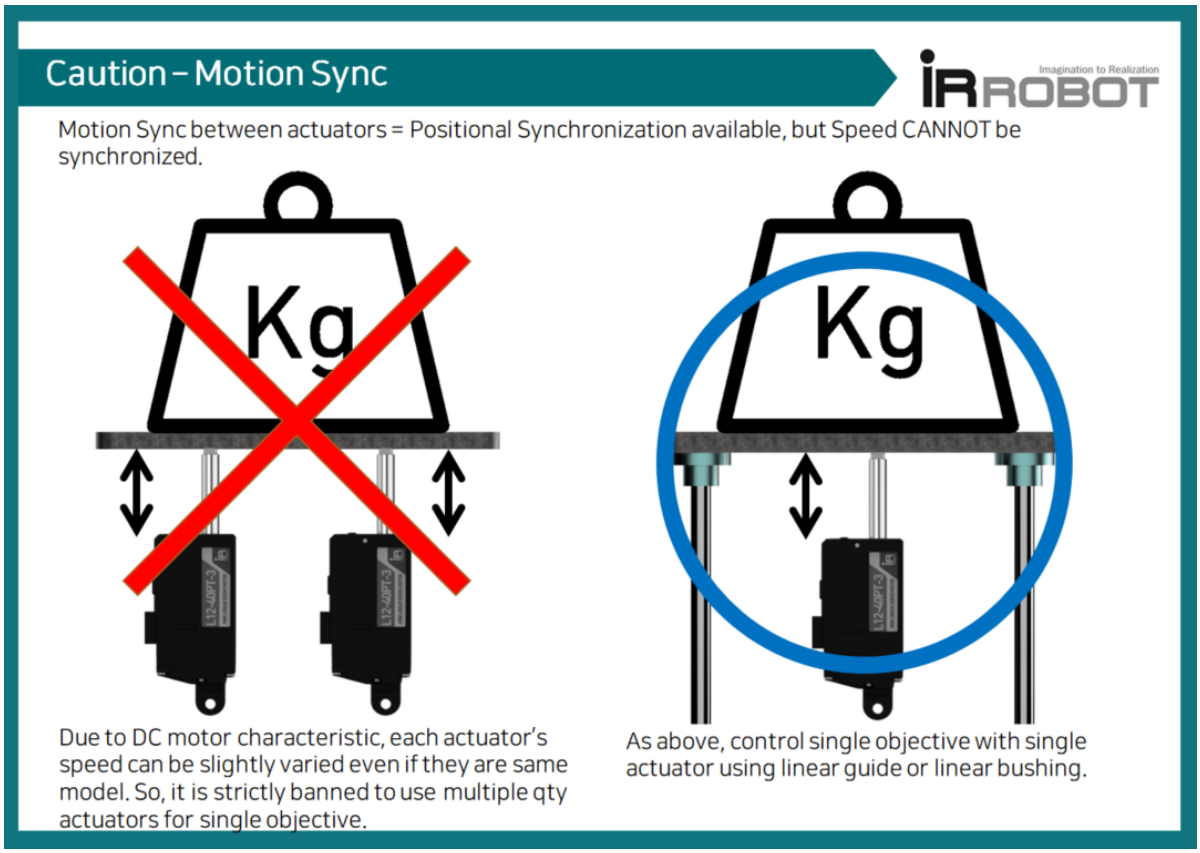
HEADER	ID	Size	Command	Factor #1	Factor #2
				Address	Length
0xFFFFFF	0xFE	0x0S	0x73	0x86	0x02

Factor #3	Factor #4	Factor #5	Factor #6	Factor #7	Factor #8	Checksum
1> ID	1> Data #1	1>Data #2	2>ID	2> Data #1	2> Data #2	
0x01	0xff	0x03	0x02	0xFF	0x07	0xF1

- Command packet to assign respective goal position to multiple qty servo motors at the same time.
- Better synchronization without delay than respective command is made for each servo motor.
- ID : Broadcast ID (Command to all connected IDs)
- Command : Send data at the same time to the ID defined in Factor (1>ID, 2>ID …)
- Address : Address present position value is saved. (See (3)Data Map)
- Length: The number of byte to read from Address (present position value consists of 2byte.)
- feedback Packet : No Feedback.

Caution Motion Synchronization of multiple actuators

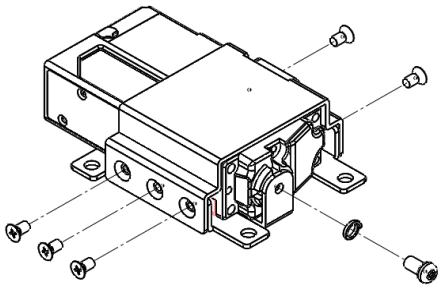
It is strictly banned to use multiple qty actuators for single objective. Due to DC motor characteristic, each actuator's speed can be slightly varied even if they are same model and goal position is same. (may cause overload to one of actuator)



5 Optional Accessories

Metal Bracket (IR-MB02/IR-MB03)

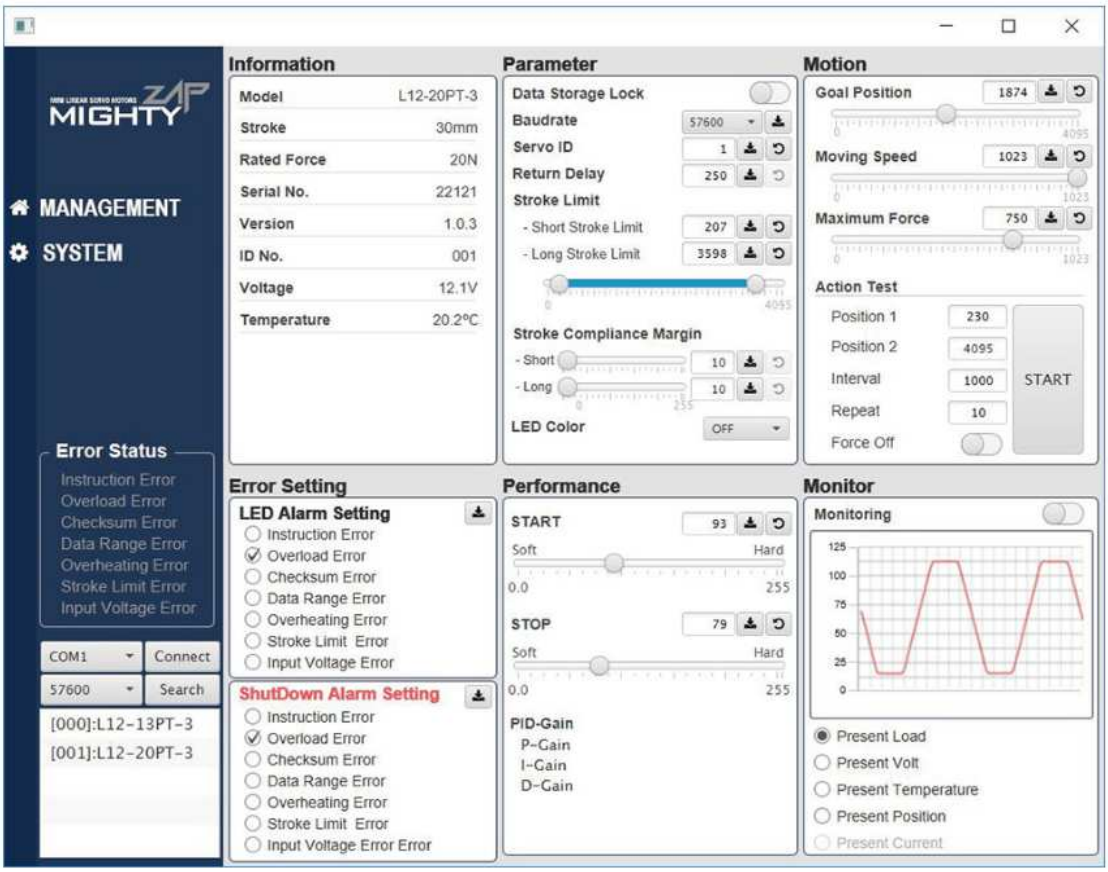
IR-MB02 is the mounting bracket for 27mm stroke lineup only. For 41mm & 56mm stroke lineup, they can be mounted via built-in mounting holes on the case. Or, if you wish more flexible mounting, you can use IR-MB03 for 41/56/96mm stroke versions. The drawing is open at our website, so you may make this bracket at their end.



PC USB Interface (IR-USB01)

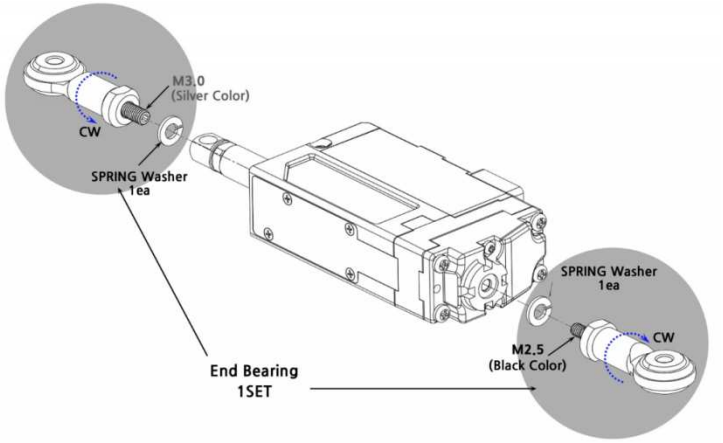
USB Interface between mightyZAP and user's PC. Through PC software, mightyZAP manager, user is able to control below.

- Parameter and Memory setting
- Motion test
- Voltage, temperature, present position, force monitoring
- System initialization and Firmware update



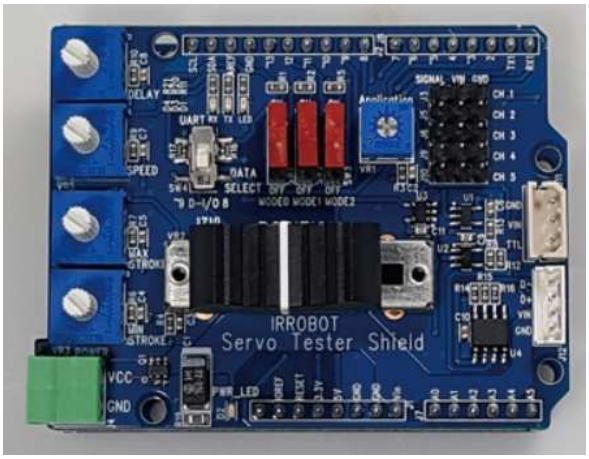
End-Bearing (IR-EB01)

Mount mightyZAP on applications using this end bearings for most optimal installation. Put it on the rod end(M3) and on the end of servo case(M2.5). Two end bearings (M3 & M2.5) to be packed in a set.



Arduino Servo Tester Shield (IR-STS01)

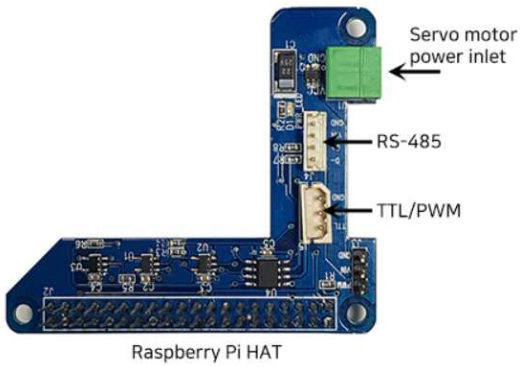
Control servo motions without PC software. Built with Ardunio Leonardo and our dedicated servo shield, user is able to control servo motor using Arduino API & library more conveniently.
For parameter setting, user still need IR-USB01, PC USB interface to utilize mightyZAP manager software.



Raspberry Pi HAT (IR-STS02)

IR-STS02 is a Raspberry Pi HAT(Hardware Attached on Top) which is compatible with Raspberry Pi B3 or Raspberry Pi Zero.

With TTL/RS-485/PWM communication interface, power connector and GPIO pins, user is able to control mightyZAP on Raspberry Pie.
API and Library can be downloaded from our web.



Raspberry Pi HAT

Extension Wire (IR-EW01~04)

Optional extension wires for applications which need longer wire harness.
IR-EW01 : Extension wire - 3pin TTL 1000mm
IR-EW02 : Extension wire - 3pin TTL 2000mm
IR-EW03 : Extension wire - 4pin RS-485 2000mm
IR-EW04 : Extension wire - 4pin RS-485 4000mm

6 Warranty Service

6.1. Warranty & Service

The warranty period of mightZAP is 1 year from the date of purchasing the goods. Please prepare some evidence showing the date of purchase and contact your product supplier or IR Robot.

Warranty service will not cover the malfunctions of product which are derived from customer's abuse, mistake, or carelessness (including normal wearing of gear train, tear of wire harness and motor burnt-out). Please kindly note that all service should be processed by designated engineers and voluntary disassembly or maintenance may void warranty.

IR Robot Customer Service Team :

- Tel : +82- 070-7600-9466
- Address : (ZIP 14502) 1303, Bucheon Techno Park 401, Pyeongcheon-Ro 655, Wonmi-Gu, Gyeonggi-Do, Korea.
- E-mail : irsales@irrobot.com

Thank you.