XM430-W210

Part Photo

[XM430-W210]

PISHROBOT.COM

H/W Specification

- MCU : ST CORTEX-M3 (STM32F103C8 @ 72MHZ,32BIT)
- POSITION SENSOR : Contactless absolute encoder (12BIT,360 DEGREE)
    - Maker : ams (www.ams.com), Part No : AS5045
- Motor : Coreless Motor
- Baud Rate : 9600 bps ~ 4.5 Mbps
- Control Algorithm : PID CONTROL
- Degree of Precision : 0.088°
- Operating Modes
    - Torque Control Mode
    - Velocity Control Mode

    - Position Control Mode (0° ~ 360°)
    - Extended Position Control Mode(Multi-turn)
    - Current-based Position Control Mode
    - PWM Control Mode

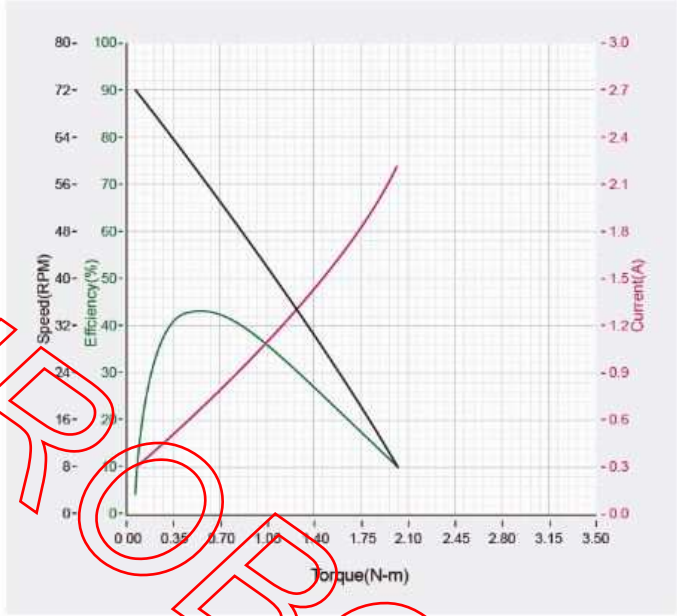
- Weight : 82g
- Dimensions(W x H x D) : 28.5mm x 46.5mm x 34mm
- Gear Ratio : 212.6 : 1
- Stall Torque
    - 2.7N.m (at 11.1V, 2.1A)
    - 3.0N.m (at 12.0V, 2.3A)
    - 3.7N.m (at 14.8V, 2.7A)
- No load speed
    - 70rpm (at 11.1V)
    - 77rpm (at 12.0V)
    - 95rpm (at 14.8V)
- Operating Temperature : -5℃ ~ +80℃
- **Input Voltage : 10 ~ 14.8V (Recommended : 12V)**
- Command Signal : Digital Packet
- Protocol Type : RS485 Asynchronous Serial Communication (8bit,1stop, No Parity)
- Link (Physical) : RS485 Multi Drop Bus
- ID : 254 ID (0~253)
- Feedback : Position, Velocity, Current, Realtime tick, Trajectory, Temperature, Input Voltage, etc.
- Material : Full Metal Gear, Metal Body(Top, Middle), Engineering Plastic Body(Bottom)
- Standby current : 40 mA

PISHROBOT.COM

Data

	Unit	Data
Weight	g	82
Dimension	mm	28.5 x 46.5 x 34
Gear Ratio	type/material	212.6(Spur/Metal)
Network	-	RS-485
Position Sensor (Resolution)	-	Contactless Absolute Encoder (360° / 4096)
Motor	-	Coreless Motor
Operation Voltage	V	10 ~ 14.8
Stall Torque	N.m	3.0 at 12V
Stall Current	A	2.3 at 12V
No Load Speed	RPM	77 at 12V

Performance Graph



Cautions for power supply

-It is recommended to use Robotis controllers or SMPS2Dynamixel for supplying power to Dynamixels.

-Please make sure to cut off the power supply before connecting Dynamixels. Then, use a switch to turn on or turn off the power.

ROBOTIS.COM

Control Table

The Control Table is a structure of data implemented in the Dynamixel. Users can read a specific Data to get status of the Dynamixel with Read Instruction Packets, and modify Data as well to control Dynamixels with WRITE Instruction Packets.

Control Table, Data, Address

The Control Table is a structure that consists of multiple Data fields to store status of the Dynamixel or to control the Dynamixel. Users can check current status of the Dynamixel by reading a specific Data from the Control Table with Read Instruction Packets. WRITE Instruction Packets enable users to control the Dynamixel by changing specific Data in the Control Table. The Address is a unique value when accessing a specific Data in the Control Table with Instruction Packets. In order to read or write data, users must designate a specific Address in the Instruction Packet. Please refer to the Protocol section of e-Manual for more details about Packets.

Area (EEPROM, RAM)

The Control Table is divided into 2 Areas. Data in the RAM Area is reset to initial values when the Dynamixel is turned on (Volatile). On the other hand, modified data in the EEPROM Area keeps their values even when the Dynamixel is turned off (Non-Volatile). Data in the EEPROM Area can only be changed when the value of Torque Enable(64) is cleared to '0'.

Access

The Control Table has two different access properties. 'RW' property stands for read and write access permission while 'R' stands for read only access permission. Data with the read only property cannot be changed by the WRITE Instruction. Read only property('R') is generally used for measuring and monitoring purpose, and read write property('RW') is used for controlling Dynamixels.

Initial Value

Each data in the Control Table is restored to initial values when the Dynamixel is turned on. Default values in the EEPROM area are initial values of the Dynamixel (factory default settings). If any values in the EEPROM area are modified by a user, modified values will be restored as initial values when the Dynamixels is turned on. Initial Values in the RAM area are restored when the Dynamixels is turned on.

Size

The Size of data varies from 1 to 4 bytes depend on their usage. Please check the size of data when updating the data with an Instruction Packet.

ROBOTIS

Area	Address	Size [byte]	Data Name	Description	Access	initial Value
EEPROM	0	2	Model Number	Model Number	R	1030
	2	4	Model Information	Model Information	R	-
	6	1	Version of Firmware	Firmware Version	R	-
	7	**1**	ID	Dynamixel ID	RW	1
	8	1	Baud Rate	Communication Baud Rate	RW	1
	9	1	Return Delay Time	Response Delay Time	RW	250
	11	1	Operating Mode	Operating Mode	RW	3
	13	1	Protocol Version	Protocol Version	RW	2
	20	4	Homing Offset	Home Position Offset	RW	0
	24	4	Moving Threshold	Velocity Threshold for Movement Detection	RW	10
	31	1	Temperature Limit	Maximum Internal Temperature Limit	RW	80
	32	2	Max Voltage Limit	Maximum Voltage Limit	RW	160
	34	2	Min Voltage Limit	Minimum Voltage Limit	RW	95
	36	2	PWM Limit	Maximum PWM Limit	RW	885
	38	2	Current Limit	Maximum Current Limit	RW	1193

	40	4	Acceleration Limit	Maximum Acceleration Limit	RW	32767
	44	4	Velocity Limit	Maximum Velocity Limit	RW	480
	48	4	Max Position Limit	Maximum Position Limit	RW	4095
	52	4	Min Position Limit	Minimum Position Limit	RW	00
	63	1	Shutdown	Shutdown Dynamixel	RW	52
	64	1	Torque Enable	Motor Torque On/Off	RW	0
	65	1	LED	Status LED On/Off	RW	0
	68	1	Status Return Level	Select Types of Status Return	RW	2
	69	1	Registered Instruction	Check Reception of Instruction	R	0
	70	1	Hardware Error Status	Hardware Error Status	R	0
	76	2	Velocity I Gain	I Gain of Velocity	RW	1920
	78	2	Velocity P Gain	P Gain of Velocity	RW	50
	80	2	Position D Gain	D Gain of Position	RW	0
	82	2	Position I Gain	I Gain of Position	RW	0
	84	2	Position P Gain	P Gain of Position	RW	800
	88	2	Feedforward 2nd Gain	2nd Gain of Feed-Forward	RW	0
R A M	90	2	Feedforward 1st Gain	1st Gain of Feed-Forward	RW	0
	100	2	Goal PWM	Target PWM Value	RW	-
	102	2	Goal Current	Target Current Value	RW	-
	104	4	Goal Velocity	Target Velocity Value	RW	-
	108	4	Profile Acceleration	Acceleration Value of Profile	RW	0
	112	4	Profile Velocity	Velocity Value of Profile	RW	0
	116	4	Goal Position	Target Position Value	RW	-
	120	2	Realtime Tick	Count Time in millisecond	R	-
	122	1	Moving	Movement Status	R	0
	123	1	Moving Status	Detailed Information of Movement Status	R	0
	124	2	Present PWM	Current PWM Value	R	-
	126	2	Present Current	Current Current Value	R	-

ROBOTIS ROBOT.COM

128	4	Present Velocity	Current Velocity Value	R	-
132	4	Present Position	Current Position Value	R	-
136	4	Velocity Trajectory	Target Velocity Trajectory Generated by Profile	R	-
140	4	Position Trajectory	Target Position Trajectory Generated by Profile	R	-
144	2	Present Input Voltage	Current Input Voltage	R	-
146	1	Present Temperature	Current Internal Temperature	R	-
168	2	Indirect Address 1	Indirect Address 1	RW	224
170	2	Indirect Address 2	Indirect Address 2	RW	225
172	2	Indirect Address 3	Indirect Address 3	RW	226
…	…	…	…	…	…
218	2	Indirect Address 26	Indirect Address 26	RW	249
220	2	Indirect Address 27	Indirect Address 27	RW	250
222	2	Indirect Address 28	Indirect Address 28	RW	251
224	1	Indirect Data 1	Indirect Data 1	RW	0
225	1	Indirect Data 2	Indirect Data 2	RW	0
226	1	Indirect Data 3	Indirect Data 3	RW	0
…	…	…	…	…	…
249	1	Indirect Data 26	Indirect Data 26	RW	0
250	1	Indirect Data 27	Indirect Data 27	RW	0
251	1	Indirect Data 28	Indirect Data 28	RW	0
578	2	Indirect Address 29	Indirect Address 29	RW	634
580	2	Indirect Address 30	Indirect Address 30	RW	635
582	2	Indirect Address 31	Indirect Address 31	RW	636
…	…	…	…	…	…
628	2	Indirect Address 54	Indirect Address 54	RW	659
630	2	Indirect Address 55	Indirect Address 55	RW	660
632	2	Indirect Address 56	Indirect Address 56	RW	661

ROBOTSHOP.COM

634	1	Indirect Data 29	Indirect Data 29	RW	0
635	1	Indirect Data 30	Indirect Data 30	RW	0
636	1	Indirect Data 31	Indirect Data 31	RW	0
…	…	…	…	…	…
659	1	Indirect Data 54	Indirect Data 54	RW	0
660	1	Indirect Data 55	Indirect Data 55	RW	0
661	1	Indirect Data 56	Indirect Data 56	RW	0

Note : Protocol 1.0 does not support addresses greater than 256. Therefore, Indirect Address 29 ~ 56 and Indirect Data 29 ~ 56 can only be accessed with Protocol 2.0.

ROBOTSHOP.COM

Address Description

EEPROM Area

Note : Any Data in EEPROM Area can only be modified when the value of Torque Enable(64) is cleared to '0'.

Model Number (0)

This address stores model number of the Dynamixel.

Firmware Version (6)

This address stores firmware version of the Dynamixel.

ID (7)

The ID is a unique value in the network to identify each Dynamixel with an Instruction Packet.

0~252 (0xFC) values can be used as an ID, and 254(0xFE) is occupied as a broadcast ID. The Broadcast ID(254, 0xFE) can send an Instruction Packet to all connected Dynamixels simultaneously.

Note : Please avoid using an identical ID for multiple Dynamixels. In order to change the ID in the EEPROM Area, Torque Enable(64) has to be cleared to '0' in advance.

Note : Please avoid using an identical ID for multiple Dynamixels. In order to change the ID in the EEPROM Area, Torque Enable(64) has to be cleared to '0' in advance.

Baud Rate (8)

Baud Rate determines serial communication speed between a controller and Dynamixels.

Baud Rate	Baud Rate[bps]	Margin of Error
0	9,600	0.000%
1(Default)	57,600	0.000%
2	115,200	0.000%
3	1M	0.000%
4	2M	0.000%
5	3M	0.000%
6	4M	0.000%
7	4.5M	0.000%

Note : Less than 3% of the baud rate error margin will not affect to UART communication.

Return Delay Time (9)

After the Dynamixel receives an Instruction Packet, it delays transmitting the Status Packet for Return Delay Time (9). For instance, if the Return Delay Time(9) is set to '10', the Status Packet will be returned after 20[µsec] when the Instruction Packet is received.

	Values	Description
Unit	2[µsec]	-
Range	0 ~ 254	Default value '250'(500[µsec]), Maximum 508[µsec]

Operating Mode (11)

Operating Mode	Operating Mode	Description
0	Torque Control Mode	Dynamixel only controls current(torque) regardless of speed and position. This mode is ideal for a gripper or a system that only uses current(torque) control or a system that has additional velocity/position controllers.
1	Velocity Control Mode (0° ~ 360°)	This mode controls velocity. This mode is identical to the Wheel Mode(endless) from existing Dynamixels. This mode is ideal for wheel-type robots.
3(Default)	Position Control Mode	This mode controls position. This mode is identical to the Joint Mode from existing Dynamixels. Operating position range is limited by Max Position Limit(48) and Min Position Limit(52). This mode is ideal for articulated robots that each joint rotates less than 360 degrees.
4	Extended Position Control Mode(Multi-turn)	This mode controls position. This mode is identical to the Multi-Turn Mode from existing Dynamixels. 512 turns are supported(-256[rev] ~ 256[rev]).

ROBOTIS ROBOT.COM

		This mode is ideal for multi-turn wrists or conveyer systems or a system that requires an additional reduction gear.
5	Current-based Position Control Mode	This mode controls both position and current(torque). Up to 512 turns are supported(-256[rev] ~ 256[rev]). This mode is ideal for a system that requires both position and current control such as articulated robots or grippers.
16	PWM Control Mode	This mode directly controls PWM output.

Note 1 : Switching Operating Mode will reset gains(PID, Feedfoward) properly to the selected Operating Mode. The profile generator and limits will also be reset.

① Profile Velocity(112), Profile Acceleration(108) : Reset to '0'

② Goal PWM(100), Goal Current(102) : Reset to PWM Limit(36), Current Limit(38) respectively

③ Current-based Position Control Mode : Reset to Position Gain(PID) and PWM Limit(36) values

Changed Position Gain(PID) and PWM Limit(36) values can be read from the Control Table.

Note 2 : PWM is the abbreviation for Pulse Width Modulation that modulates PWM Duty to control motors. The PWM Control Mode changes pulse width to control average supply voltage to the motor and this technique is widely used in the motor control field. Therefore, PWM Control Mode uses Goal PWM(100) value to control supply voltage for Dynamixel. PWM Control Mode is similar to the Wheel Mode of Dynamixel AX and RX series.

ROBOTIS ROBOT.COM

Protocol version (13)

Users can select Dynamixel protocol version (1.0 and 2.0). It is recommended to use an identical protocol version for multiple Dynamixels.

	Protocol Version	Compatible Dynamixels
1	1.0	AX, DX, RX, MX, EX Series
2(Default)	2.0	XL-320, Dynamixel-PRO Series

Note : The protocol 2.0 is greatly improved from the protocol 1.0. Accessing some of the Control Table area might be denied if protocol 1.0 is selected. This manual complies with protocol 2.0. Please refer to the Protocol section of e-Manual for more details about the protocol.

Homing Offset (20)

Users can adjust the Home position by setting Home Offset(20). The Homing Offset value is added to the Present Position(132).

Present Position(132) = Actual Position + Homing Offset(20).

	Values	Description
Unit	about 0.088[deg]	4096 resolution. All position related Data uses the same unit
Range	-1,044,479 ~ 1,044,479	±255[rev] range

Note : In case of the Position Control Mode(Joint Mode) that rotates less than 360 degrees, any invalid Homing Offset(20) values will be ignored(valid range : -1024 ~ 1024).

Moving Threshold (24)

This value helps to determine whether the Dynamixel is in motion or not. When the absolute value of Present Velocity(128) is greater than the Moving Threshold(24), Moving(122) is set to ‘1’, otherwise it is cleared to ‘0’.

	Values	Description
Unit	about 0.088[deg]	4096 resolution. All position related Data uses the same unit
Range	-1,044,479 ~ 1,044,479	±255[rev] range

Temperature Limit (31)

This value limits operating temperature. When the Present Temperature(146) that indicates internal temperature of Dynamixel is greater than the Temperature Limit(31), the Over Heating Error Bit(0x04) and Hardware Error Bit(0x80) in the Hardware Error Status(70) will be set. If Overheating Error Bit(0x04) is configured in the Shutdown(63), Torque Enable(64) is cleared to ‘0’ and Torque is disabled. For more details, please refer to the Shutdown(63) section.

	Values	Description
Unit	about 1[°C]	All temperature related Data uses the same unit
Range	0 ~ 100	

Max Voltage Limit (32), Min Voltage Limit (34)

These values are maximum and minimum operating voltages. When current input voltage acquired from Present Input Voltage(144) exceeds the range of Max Voltage Limit(32) and Min Voltage Limit(34), Voltage Range Error Bit(0x01) and Hardware Error Bit(0x80) in the Hardware Error Status(70) are set. If Input Voltage Error Bit(0x10) is configured in the Shutdown(63), Torque Enable(64) is cleared to ‘0’ and Torque is disabled. For more details, please refer to the Shutdown(63) section.

	Values	Description
Unit	about 0.1[V]	All voltage related Data uses the same unit
Range	95 ~ 160	9.5 ~ 16.0[V]

PWM Limit (36)

This value indicates maximum PWM output. Goal PWM(100) can’t be configured with any values exceeding PWM Limit(36). PWM Limit(36) is commonly used in all operating mode as an output limit, therefore decreasing PWM output will result in decreasing torque and velocity. For more details, please refer to the Gain section of each operating modes.

	Values	Description
Range	0 ~ 885	885 = 100[%] output

Current Limit (38)

This value indicates maximum current(torque) output limit. Goal Current(102) can’t be configured with any values exceeding Current Limit(38). The Current Limit(38) is used in

PLOTROBOT.COM

Torque Control Mode and Current-based Position Control Mode, therefore decreasing Current Limit(38) will result in decreasing torque of Dynamixel. For more details, please refer to the Position PID Gain(80 ~ 84).

	Values	Description
Unit	about 2.69[mA]	All current related Data uses the same unit
Range	0 ~ 1,193	-

Note : Current Limit(38) could be differ by each Dynamixel so please check the Control Table.

ROBOTIS

Acceleration Limit (40)

This value indicates maximum Profile Acceleration(108). Profile Acceleration(108) can't be configured with any values exceeding Acceleration Limit(40). Profile Acceleration(108) is used in all operating mode except Torque Control Mode in order to generate a target trajectory. For more details, please refer to the Profile Velocity(112).

	Values	Description
Unit	0.0036[Rev/min2]	All acceleration related Data uses the same unit, 1/64 of the velocity unit
Range	0 ~ 32,767	-

Note : Bit information of the Error field in the Status Packet is different from protocol 1.0 and protocol 2.0. This manual complies with protocol 2.0. Please refer to the Protocol section of e-Manual for more details about the protocol.

Velocity Limit (44)

This value indicates maximum velocity of Goal Velocity(104) and Profile Velocity(112). For more details, please refer to the Profile Velocity(112).

	Values	Description
Unit	0.229[RPM]	All velocity related Data uses the same unit
Range	0 ~ 1,223	-

Max Position Limit (48), Min Position Limit (52)

These values limit maximum and minimum target positions for Position Control Mode(Joint Mode) within the range of 1 rotation(0~4095). Therefore, Goal Position(116) should be configured within the position limit range. These values are not used in Extended Position Control Mode and Current-based Position Control Mode.

	Values	Description
Unit	0.088[deg]	All position related Data uses the same unit
Range	0 ~ 4095	The range is limited by 1 rotation

Note : Max Position Limit(48) and Min Position Limit(52) are only used in Position Control Mode with a single turn.

Shutdown (63)

The Dynamixel can protect itself by detecting dangerous situations that could occur during the operation. Each Bit is inclusively processed with the 'OR' logic, therefore, multiple options can be generated. For instance, when '0x05' (binary : 00000101) is defined as Shutdown(63), Dynamixel can detect both Input Voltage Error(binary : 00000001) and Overheating Error(binary : 00000100). If those errors are detected, Torque Enable(64) is cleared to '0' and the motor output becomes 0[%]. REBOOT is the only method to reset Torque Enable(64) to '1'(Torque ON) after the shutdown. The followings are detectable situations.

Shutdown		Definition	Description
bit 7	**0x80**	-	Unused, always '0'
bit 6	**0x40**	-	Unused, always '0'
bit 5	**0x20**	Overload Error(Default)	Detect persistent load that exceeds maximum output
bit 4	**0x10**	Electrical Shock Error(Default)	Detect electrical shock on the circuit, or input power is insufficient to operate the motor
bit 3	**0x08**	Motor Encoder Error	Detect malfunction of the motor encoder
bit 2	**0x04**	Overheating Error(Default)	Detect internal temperature exceeds the configured operating temperature
bit 1	**0x02**	Motor Hall Sensor Error(Unused)	Unused, always '0'
bit 0	**0x01**	Input Voltage Error	Detect input voltage exceeds the configured operating voltage

Note : If Shutdown occurs, use below method to REBOOT Dynamixels.

① H/W REBOOT : Turn off the power and turn on again

② S/W REBOOT : Transmit REBOOT Instruction (For more details, please refer to the Protocol section of e-Manual.)

RAM Area

Torque Enable (64)

Controls Torque ON/OFF. Writing '1' to this address will turn on the Torque and all Data in the EEPROM area will be protected.

Values	Description
0(Default)	Torque OFF(Free-run) and the motor does not generate torque.
1	Torque ON and all Data in the EEPROM area will be locked.

Note : Present Position(132) will be reset when Operating Mode(11) and Torque Enable(64) are updated. For more details, please refer to the Homing Offset(20) and Present Position(132).

LED (65)

Turn on or turn off the LED. Dynamixel LED can only be controlled by LED(65).

ROBOTIS ROBOT.COM

Values Description

Values	Description
0(Default)	Turn off the LED.
1	Turn on the LED.

Status Return Level (68)

This value decides how to return Status Packet when Dynamixel receives an Instruction Packet.

Values	When to return Status Packet	Description
0	PING Instruction	Status Packet will not be returned for all Instructions. (Exception : PING Instruction)
1	PING Instruction READ Instruction	Status Packet will only be returned for READ Instruction. exceptionally returns Status Packet to all Instruction.
2	All Instructions	Status Packet will be returned for all Instructions.

Note : If the ID of Instruction Packet is set to Broad Cast ID(0xFE), Status Packet will not be returned for READ and WRITE Instructions regardless of Status Return Level(68). For more details, please refer to the Protocol section of e-Manual.

Registered Instruction (69)

This value will be set to '1' when Dynamixel receives REG_WRITE Instruction Packet and processing ACTION Instruction Packet will clear the value to '0'.

Hardware Error Status (70)

This value indicates hardware error status. For more details, please refer to the Shutdown(63).

Velocity I Gain (76), Velocity P Gain (78)

These values indicate Gains of Velocity Control Mode. Gains of Dynamixel's internal controller can be calculated from Gains of the Control Table as shown below. The constant in each equations include sampling time. Velocity P Gain of Dynamixel's internal controller is abbreviated to K_VP and that of the Control Table is abbreviated to $K_VP_{(TBL)}$.

	Controller Gain	Conversion Equations	Range	Description
Velocity I Gain(76)	**K_VI**	$K_VI = K_VI_{(TBL)} / 65563$	0 ~ 32767	I Gain
Velocity P Gain(78)	**K_VP**	$K_VP = K_VP_{(TBL)} / 128$	0 ~ 32767	P Gain

Below figure is a block diagram describing the velocity controller in Velocity Control Mode. When the instruction transmitted from the user is received by Dynamixel, it takes following steps until driving the horn.

① An Instruction from the user is transmitted via Dynamixel bus, then registered to Goal Velocity(104).

ROBOTIS ROBOT.COM

② Goal Velocity(104) is converted to target velocity trajectory by Profile Acceleration(108).

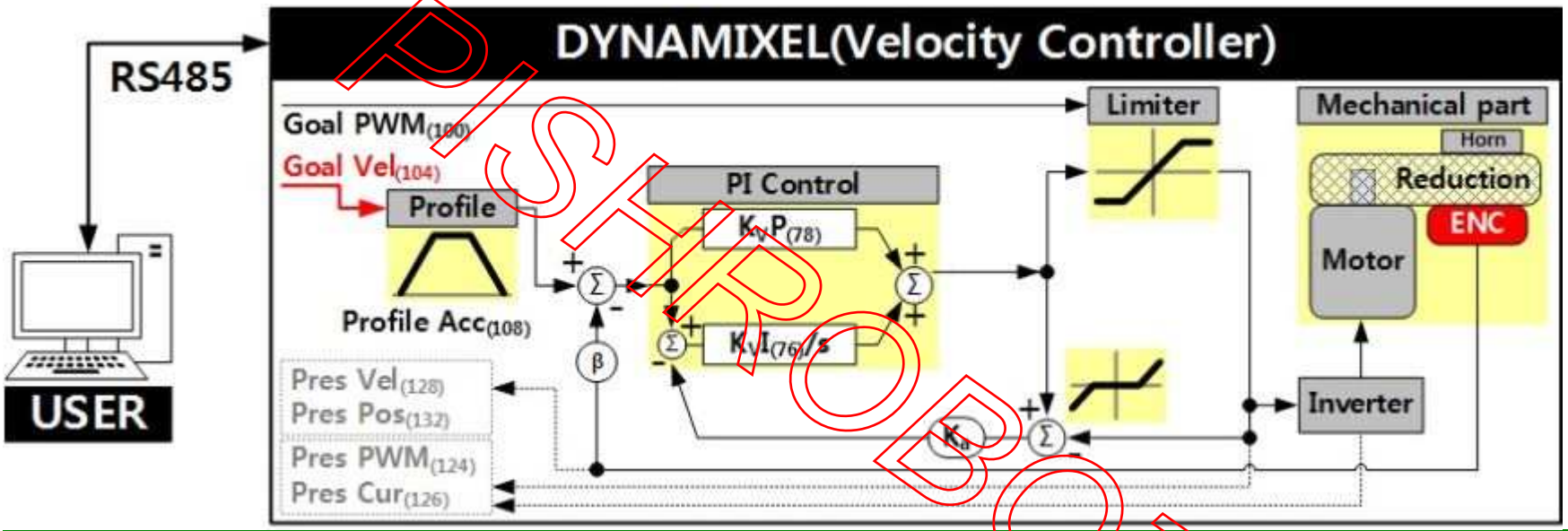
③ The target velocity trajectory is stored at Velocity Trajectory(136).

④ PI controller calculates PWM output for the motor based on the target velocity trajectory.

⑤ Goal PWM(100) sets a limit on the calculated PWM output and decides the final PWM value.

⑥ The final PWM value is applied to the motor through an Inverter, and the horn of Dynamixel is driven.

⑦ Results are stored at Present Position(132), Present Velocity(128), Present PWM(124) and Present Current(126).



Note : K_a stands for Anti-windup Gain and 'β' is a conversion coefficient of position and velocity that cannot be modified by users. For more details about the PID controller, please refer to the below website.

http://en.wikipedia.org/wiki/PID_controller

Position D Gain (80), Position I Gain (82), Position P Gain (84)

Feedforward 2nd Gain (88), Feedforward 1st Gain (90)

These Gains are used in Position Control Mode and Extended Position Control Mode. Gains of Dynamixel's internal controller can be calculated from Gains of the Control Table as shown below. The constant in each equations include sampling time. Position P Gain of Dynamixel's internal controller is abbreviated to K_PP and that of the Control Table is abbreviated to $K_PP_{(TBL)}$.

	Controller Gain	Conversion Equation	Range	Description
Position D Gain(76)	**KpD**	$K_PD = K_PD_{(TBL)} / 16$	0 ~ 32767	D Gain
Position I Gain(76)	**KpI**	$K_PI = K_PI_{(TBL)} / 65536$	0 ~ 32767	I Gain
Position P Gain(78)	**KpP**	$K_PP = K_PP_{(TBL)} / 128$	0 ~ 32767	P Gain
Feedforward 2nd Gain(88)	**K_{FF2nd}**	$K_{FF2nd} = K_{FF2nd(TBL)} / 4$	0 ~ 32767	Feedforward Acceleration Gain

ROBOTIS.COM

Feedforward 1st Gain(90)	**K_{FF1st}**	$K_{FF1st} = K_{FF1st(TBL)} / 4$	0 ~ 32767	Feedforward Velocity Gain

Below figure is a block diagram describing the position controller in Position Control Mode and Extended Position Control Mode. When the instruction from the user is received by Dynamixel, it takes following steps until driving the horn.

① An Instruction from the user is transmitted via Dynamixel bus, then registered to Goal Position(116).

② Goal Position(116) is converted to target position trajectory and target velocity trajectory by Profile Velocity(112) and Profile Acceleration(108).

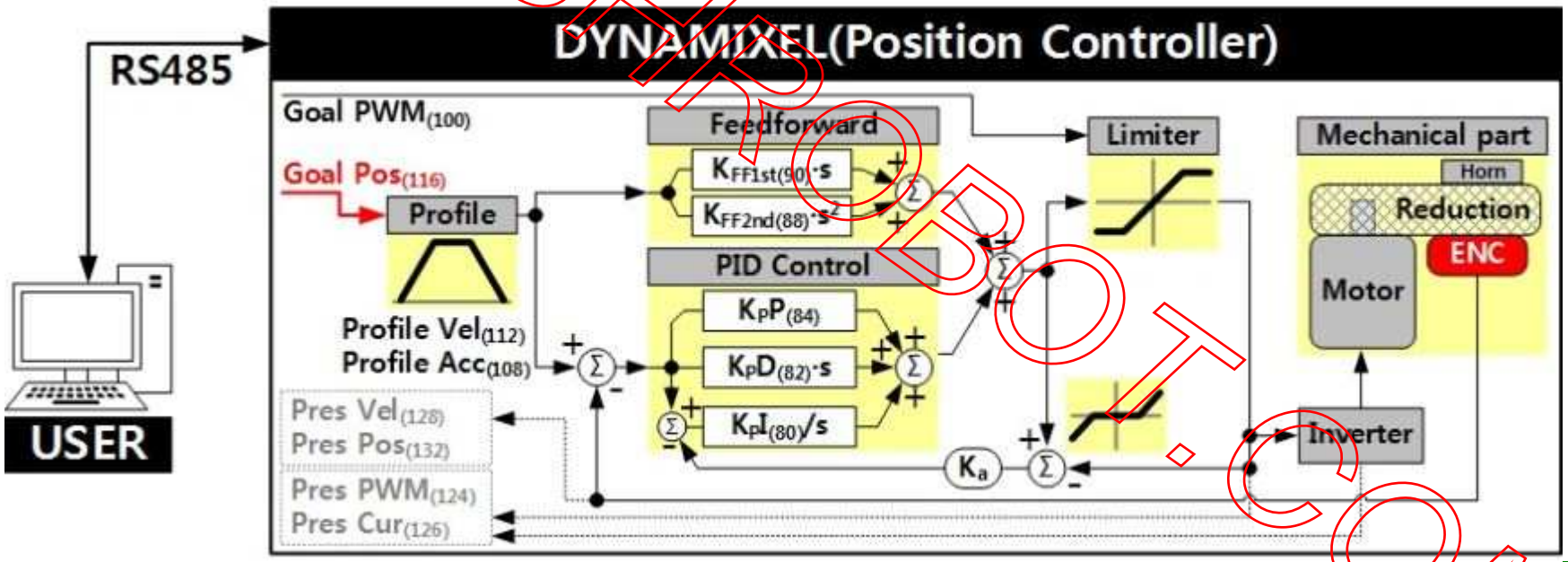
③ The target position trajectory and target velocity trajectory is stored at Position Trajectory(140) and Velocity Trajectory(136) respectively.

④ Feedforward and PID controller calculate PWM output for the motor based on target trajectories.

⑤ Goal PWM(100) sets a limit on the calculated PWM output and decides the final PWM value.

⑥ The final PWM value is applied to the motor through an Inverter, and the horn of Dynamixel is driven.

⑦ Results are stored at Present Position(132), Present Velocity(128), Present PWM(124) and Present Current(126).



Note1) : In case of PWM Control Mode, both PID controller and Feedforward controller are deactivated while Goal PWM(100) value is directly controlling the motor through an Inverter. In this manner, users can directly control the supplying voltage to the motor.

Note2) : K_a is an Anti-windup Gain that cannot be modified by users.

Below figure is a block diagram describing the current-based position controller in Current-based Position Control Mode. As Current-based Position Control Mode is quite similar to Position Control Mode, differences will be focused in the following steps. The differences are highlighted with a green marker in the block diagram as well.

① Feedforward and PID controller calculates target current based on target trajectory.

② Goal Current(102) decides the final target current by setting a limit on the calculated target current.

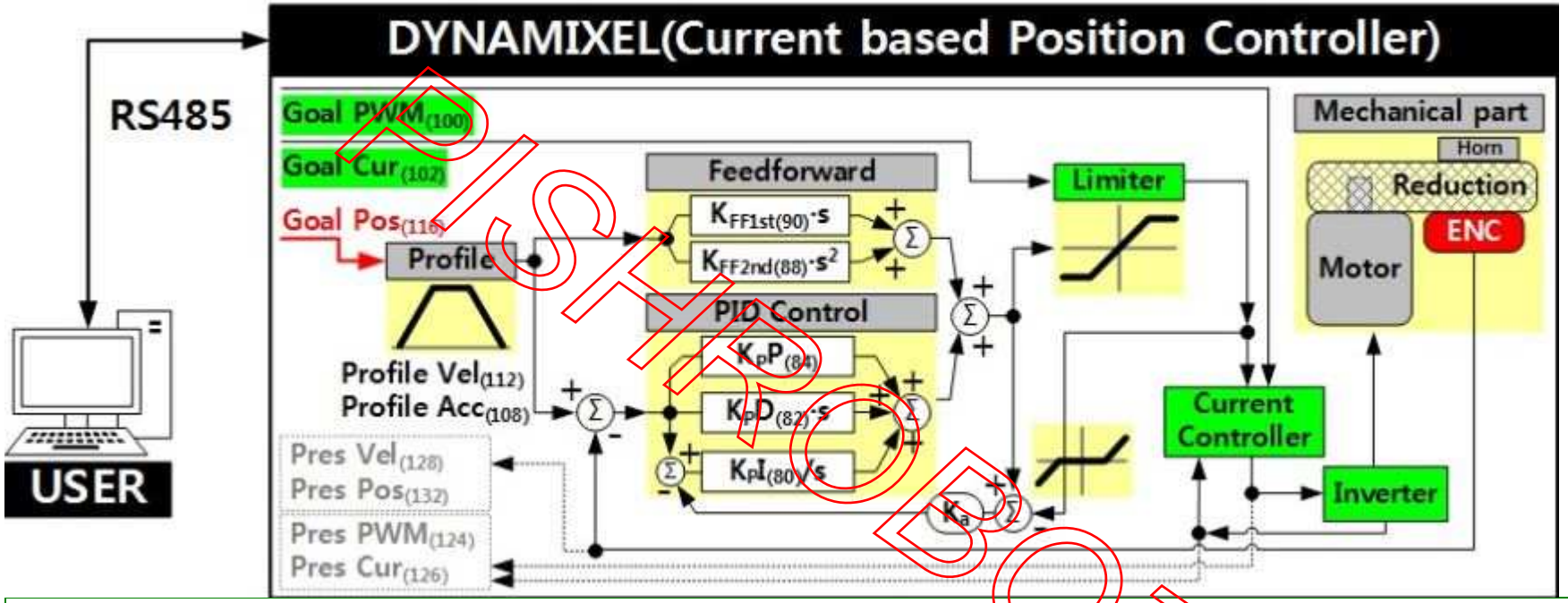
ROBOTIS

③ Current controller calculates PWM output for the motor based on the final target current.

④ Goal PWM(100) sets a limit on the calculated PWM output and decides the final PWM value.

⑤ The final PWM value is applied to the motor through an Inverter, and the horn of Dynamixel is driven.

⑥ Results are stored at Present Position(132), Present Velocity(128), Present PWM(124) and Present Current(126).



Note : Ka is an Anti-windup Gain that cannot be modified by users. For more details about the PID controller and Feedforward controller, please refer to the below websites.

http://en.wikipedia.org/wiki/PID_controller

https://en.wikipedia.org/wiki/Feed_forward_(control)

Goal PWM (100)

In case of PWM Control Mode, both PID controller and Feedforward controller are deactivated while Goal PWM(100) value is directly controlling the motor through an Inverter. In other control modes, this value is used to limit PWM value. This value cannot exceed PWM Limit(36). Please refer to the Gain section in order to see how Goal PWM(100) affects to different control modes.

	Values	Description
Range	0 ~ PWM Limit(36)	Initial Value of PWM Limit(36) : '885'

Goal Current (102)

In case of Torque Control Mode, Goal Current(102) can be used to set a target current. This value sets a limit to current in Current-based Position Control mode. This value cannot exceed Current Limit(38).

ROBOTIS ROBOT.COM

	Values	Description
Unit	about 2.69[mA]	All current related Data uses the same unit
Range	0 ~ Current Limit(38)	-

Note : Applying high current to the motor for long period of time might damage the motor.

Goal Velocity (104)

In case of Velocity Control Mode, Goal Velocity(104) can be used to set a target velocity. This value cannot exceed Velocity Limit(44). For now, Goal Velocity(104) is used for target velocity, but this value is not used to limit the velocity.

	Values	Description
Unit	0.229[RPM]	All velocity related Data uses the same unit
Range	0 ~ Velocity Limit(44)	-

Note1) : The maximum velocity and maximum torque of Dynamixel is affected by supplying voltage. Therefore, if supplying voltage changes, so does the maximum velocity. This manual complies with recommended supply voltage(12[V]).

Note2) : If Profile Acceleration(108) and Goal Velocity(104) are modified simultaneously, modified Profile Acceleration(108) will be used to process Goal Velocity(116).

Profile Acceleration (108)

The acceleration of Profile can be set with this value. Profile Acceleration(108) can be used in all control modes except Torque Control Mode. Profile Acceleration(108) cannot exceed Acceleration Limit(40). For more details, please refer to the Profile Velocity(112).

	Values	Description
Unit	0.0036[Rev/min2]	All acceleration related Data uses the same unit, 1/64 of the velocity unit
Range	0 ~ Acceleration Limit(40)	The value '0' on Profile Acceleration(108) means infinite acceration.

Profile Velocity (112)

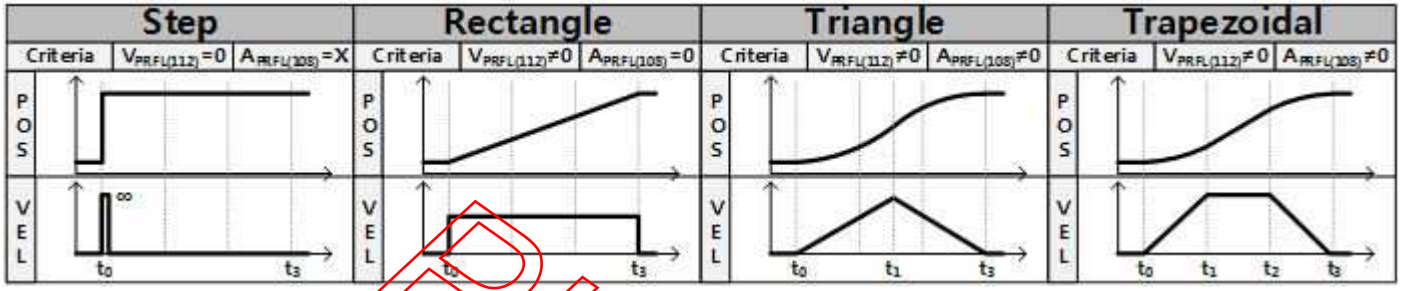
The Maximum velocity of Profile can be set with this value. Profile Velocity(112) can be used in all control modes except Torque Control Mode and Velocity Control Mode. Profile Velocity(112) cannot exceed Velocity Limit(40). Velocity Control Mode only uses Profile Acceleration(108) instead of Profile Velocity(112).

	Values	Description
Unit	0.229[RPM]	All velocity related Data uses the same unit
Range	0 ~ Velocity Limit(44)	If Profile Velocity(112) is set to ‘0’, it stands for infinite velocity.

The Profile is an acceleration/deceleration control method to reduce vibration, noise and load of the motor by controlling dramatically changing velocity and acceleration. It is also called Velocity Profile as it controls acceleration and deceleration based on velocity. Dynamixel provides 4 different types of Profile. The following explains 4 Profiles and how to select them. Profiles are usually selected by a combination of Profile Velocity(112) and Profile Acceleration(108). Triangular and Trapezoidal Profiles exceptionally consider total travel distance(ΔPos, the distance difference between target position

ROBOTIS.COM

and current position) as an additional factor. For convenience, Profile Velocity(112) is abbreviated to V_{PRFL} and Profile Acceleration(108) is abbreviated to V_{PRFL}. 'X' stands for "Don't Care" case.



When given Goal Position(116), Dynamixel's profile creates target velocity trajectory based on current velocity(initial velocity of the Profile). When Dynamixel receives updated target position from a new Goal Position(116) while it is moving toward the previous Goal Position(116), velocity smoothly varies for the new target velocity trajectory. Maintaining velocity continuity while updating target velocity trajectory is called Velocity Override. For a simple calculation, let's assume that the initial velocity of the Profile is '0'. The following explains how Profile processes Goal Position(116) instruction in Position Control mode, Extended Position Control Mode, Current-based Position Control Mode.

① An Instruction from the user is transmitted via Dynamixel bus, then registered to Goal Position(116).

② Acceleration time(t1) is calculated from Profile Velocity(112) and Profile Acceleration(108).

③ Types of Profile is decided based on Profile Velocity(112), Profile Acceleration(108) and total travel distance(ΔPos, the distance difference between target position and current position)

Condition	Types of Profile
$V_{PRFL(112)} = 0$	Profile not used (Step Instruction)
$(V_{PRFL(112)} \neq 0)$ & $(A_{PRF(108)} = 0)$	Rectangular Profile
$(V_{PRFL(112)} \neq 0)$ & $(A_{PRF(108)} \neq 0)$ & $(V_{PRFL_TRI} \leq V_{PRFL(112)})$	Triangular Profile
$(V_{PRFL(112)} \neq 0)$ & $(A_{PRF(108)} \neq 0)$ & $(V_{PRFL_TRI} > V_{PRFL(112)})$	Trapezoidal Profile

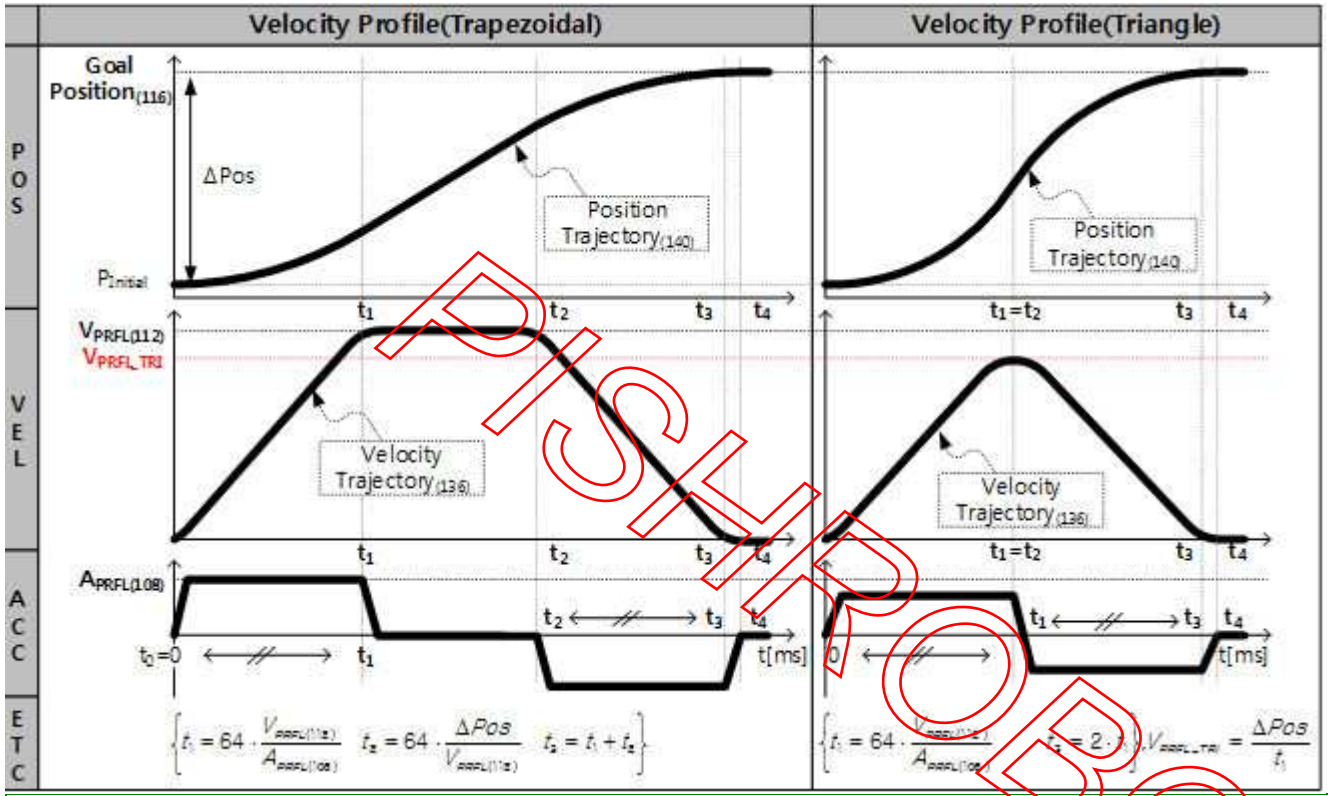
④ Selected Profile type is stored at Moving Status(123).(Refer to the Moving Status(123))

⑤ Dynamixel is driven by the calculated target trajectory from Profile.

⑥ Target velocity trajectory and target position trajectory from Profile are stored at Velocity Trajectory(136) and Position Trajectory(140) respectively.

⑦ V_{PRFL_TRI} of ③ and Travel time(t3) to reach Goal Position(116) is calculated as below.

ROBOTIS.COM



Note1) : Dynamixel supports Jerk control in order to minimize dramatic change of acceleration. Therefore, actual travel time by the target trajectory of Profile could be longer than t3(t4 of above figure).

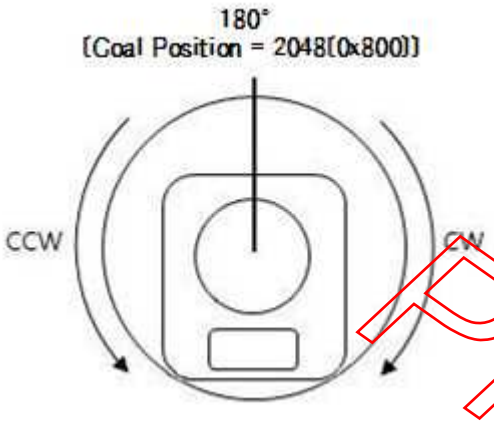
Note2) : Velocity Control Mode only uses Profile Acceleration(108). Step and Trapezoidal Profiles are supported. Velocity Override and Jerk control are supported as well. Acceleration time(t1) can be calculated as below equation.

$$\frac{Goal\ Velocity_{(104)}}{Profile\ Acceleration_{(108)}} t_1 = 64$$

PISHROBOT.COM

Goal Position (116)

Target position can be set with Goal Position(116). From the front view of Dynamixels, CCW is an increasing direction whereas CW is a decreasing direction. The way to reaching Goal Position(116) is differ by 4 Profiles provided by Dynamixels. Please refer to the Profile Velocity(112) for more details.



	Values		Description
Unit	0.088[deg]		1[rev] : 0 ~ 4,095
Range	Position Control Mode	Min Position Limit(52) ~ Max Position Limit(48)	Initial Value : 0 ~ 4,095
	Extended Position Control Mode	1,048,575 ~ 1,048,575	-256[rev] ~ 256[rev]
	Current-based Position Control Mode	-	-

Note : If Profile Acceleration(108), Profile Velocity(112) and Goal Position(116) are modified simultaneously, Goal Position(116) is processed based on updated Profile Acceleration(108) and Profile Velocity(112).

Realtime Tick (120)

This value indicates Dynamixel's time.

	Values	Description
Unit	1[ms]	-
Range	0 ~ 32,767	The value resets to '0' when it exceeds 32,767

Moving (122)

This value indicates whether Dynamixel is in motion or not. If absolute value of Present Velocity(128) is greater than Moving Threshold(24), Moving(122) is set to '1'. Otherwise, it will be cleared to '0'. However, this value will always be set to '1' regardless of Present Velocity(128) while Profile is in progress with Goal Position(116) instruction.

Moving	Description
0	Movement is not detected
1	Movement is detected, or Profile is in progress(Goal Position(116) instruction is being processed)

PISHROBOT.COM

Moving Status (123)

This value provides additional information about the movement. Following Error Bit(0x08) and In-Position Bit(0x01) only work with Position Control Mode, Extended Position Control Mode, Current-based Position Control Mode.

Moving Status		Details	Description
Bit 7	**0x80**	-	Unused
Bit 6	**0x40**	-	Unused
Bit 5 ~ Bit 4	**0x30**	Profile Type(0x30)	Trapezoidal Velocity Profile
		Profile Type(0x20)	Triangular Velocity Profile
		Profile Type(0x10)	Rectangular Velocity Profile
		Profile Type(0x00)	Rectangular Velocity Profile
Bit 3	**0x08**	Following Error	Dynamixel fails to reach target position trajectory
Bit 2	**0x04**	-	Unused
Bit 1	**0x02**	Profile Ongoing	Profile is in progress with Goal Position(116) instruction
Bit 0	**0x01**	In-Position	Dynamixel is reached to target position

Present PWM (124)

This value indicates current PWM. For more details, please refer to the Goal PWM(100).

Present Current (126)

This value indicates current Current. For more details, please refer to the Goal Current(102).

Present Velocity (128)

This value indicates current Velocity. For more details, please refer to the Goal Velocity(104).

Present Position (132)

This value indicates current Position. For more details, please refer to the Goal Position(116)

> Note : Present Position(132) represents 4[byte] continuous range(-2,147,483,648 ~ 2,147,483,647) when Torque is turned off. Updating Operating Mode(11) and Torque Enable(64) will initialize Present Position(132) to relative position value within 1[rev] (0~4,095). Initializing value is affected by Homing Offset(20).

Velocity Trajectory (136)

This is a target velocity trajectory created by Profile. Operating method can be changed based on control mode. For more details, please refer to the Profile Velocity(112).

ROBOTIS ROBOT.COM

① Velocity Control Mode : When Profile reaches to the endpoint, Velocity Trajectory(136) becomes equal to Goal Velocity(104).

② Position Control Mode, Extended Position Control Mode, Current-based Position Control Mode : Velocity Trajectory is used to create Position Trajectory(140). When Profile reaches to endpoint, Velocity Trajectory(136) is cleared to '0'.

Position Trajectory (140)

This is a target position trajectory created by Profile. This value is only used in Position Control Mode, Extended Position Control Mode, Current-based Position Control Mode. For more details, please refer to the Profile Velocity(112).

Present Input Voltage (144)

This value indicates current voltage that is being supplied. For more details, please refer to the Max/Min Voltage Limit(32, 34).

Present Temperature (144)

This value indicates current temperature of Dynamixel. For more details, please refer to the Temperature Limit(31).

Indirect Address 1 ~ 28 (168, 170 ~ 220, 222), Indirect Address 29 ~ 56 (578, 580 ~ 630, 632)

Indirect Data 1 ~ 28 (224, 225 ~ 250, 251), Indirect Data 29 ~ 56 (634, 635 ~ 660, 661)

Indirect Address and Indirect Data are useful when accessing two remote addresses in the Control Table as sequential addresses. Sequential addresses increase Instruction Packet efficiency. Addresses that can be defined as Indirect Address is limited to RAM area(Address 64 ~ 661).

If specific address is allocated to Indirect Address, Indirect Address inherits features and properties of the Data from the specific Address. Property includes Size(Byte length), value range, and Access property(Read Only, Read/Write). For instance, allocating 65(Address of LED) to Indirect Address 1(168), Indirect Data 1(224) can perform exactly same as LED(65).

① Example 1) Allocating Size 1[byte] LED(65) to Indirect Data 1(224)

A. Indirect Address 1(168) : change the value to '65' which is the address of LED

B. Set Indirect Data 1(224) to '1' → LED(65) also becomes '1' and LED is turned on.

C. Set Indirect Data 1(224) to '0' → LED(65) also becomes '0' and LED is turned off.

② Example 2) Allocating Size 4[byte] Goal Position(116) to Indirect Data 2(225), all 4[byte] has to be allocated.

A. Indirect Address 2(170) : change the value to '116' which is the first address of Goal Position.

B. Indirect Address 3(172) : change the value to '117' which is the first address of Goal Position.

C. Indirect Address 4(174) : change the value to '118' which is the first address of Goal Position.

D. Indirect Address 5(176) : change the value to '119' which is the first address of Goal Position.

E. Set 4[byte] value '1024' to Indirect Data 2 → Goal Position(116) also becomes '1024 and Dynamixel moves.

Indirect Address	Values	Description
Range	64~661	Indirection Address can't be allocated with EEPROM area

PISHROBOT.COM

Note 1) : In order to allocate Data in the Control Table longer than 2[byte] to Indirect Address, all address must be allocated to Indirect Address like the above Example 2.

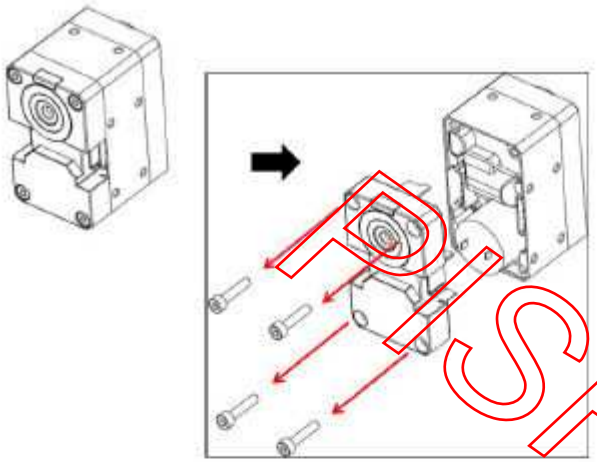
Note 2) : Indirect Address 29 ~ 56 and Indirect Data 29 ~ 56 can only be accessed with Protocol 2.0.

Combination

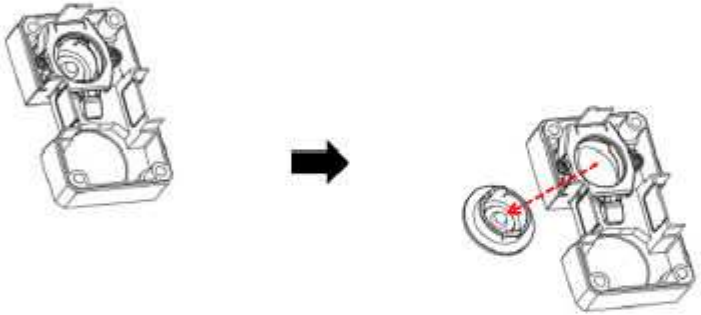
The following example shows the combination structure of option frames and horns.

PISHROBOT.COM

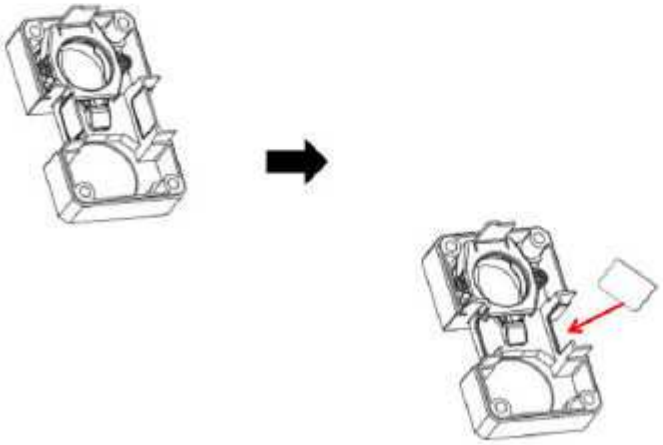
1.



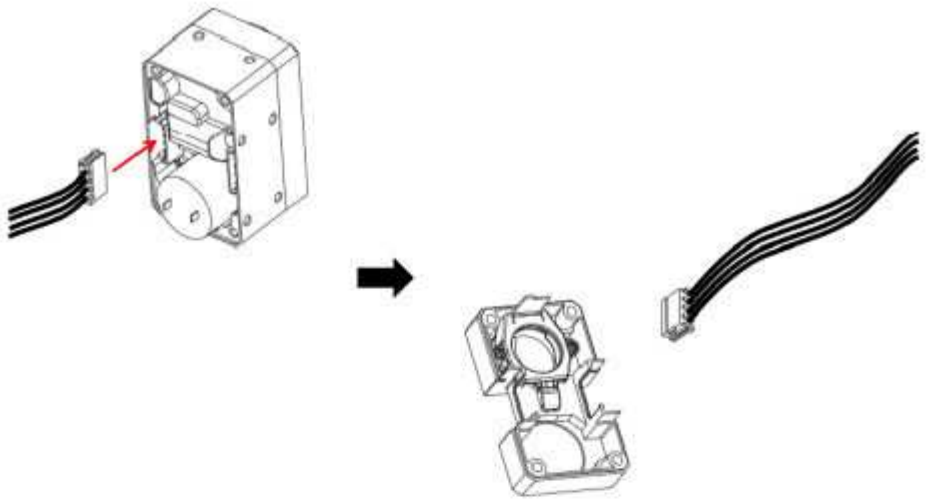
2.



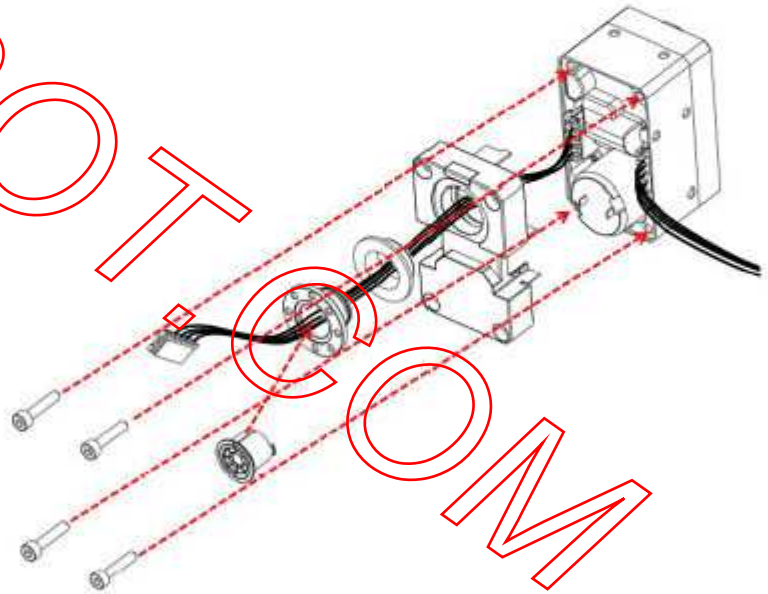
3.



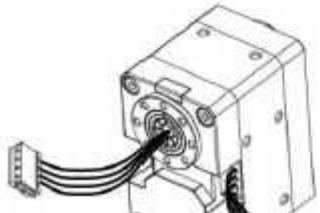
4.



5.



6.



DFSHROBOT.COM