Overview and Characteristics of Servo A1-16

A1-16 is a modular actuator, which combines a gear reducer, a DC motor and an embedded control board in one small package. A1-16 provides the necessary torque for building a small robot. Also, A1-16 could give much information of internal condition such as the internal temperature, supply voltage and so on. A1-16 is much easier to use for beginners and advance users than a traditional servo motor

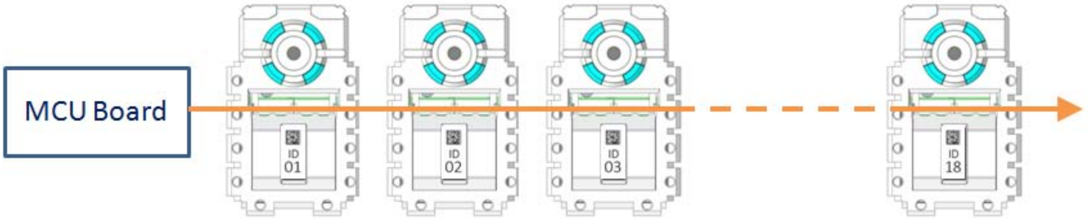
General Servo Motor Specifications

1. Operation voltage : 8 ~ 12 V(default)
2. Maximum speed : 70 ± 10 rpm
3. Stall torque : 25.0 kg-cm max
4. Rotary position feedback with 360° continuous rotation angle and maximum 330° effective position control range
5. Protocol type : Duplex UART 5V TTL serial communication(8, N, 1)
6. Communication Speed : 9600, 19200, 57600, 115200(default)
7. Feedback Information : Position, Temperature, Current, Voltage, etc

Dimensions of Servo Motor

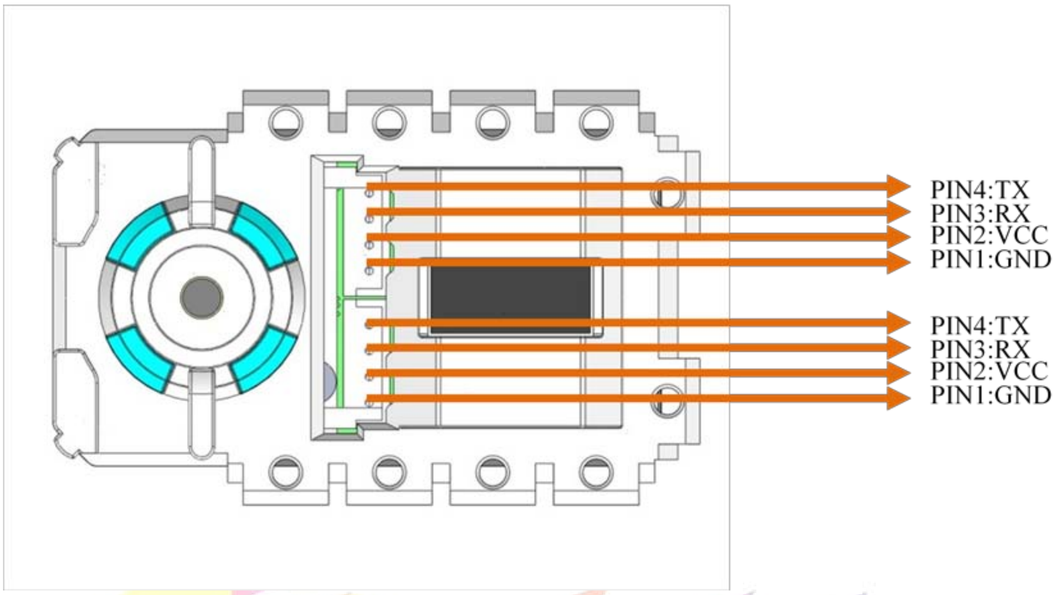
1. Size : 50 x 32 x 40.5 mm
2. Weight : 60 ± 2 grams
3. Material : POM casing with metal gear

Wiring Connection

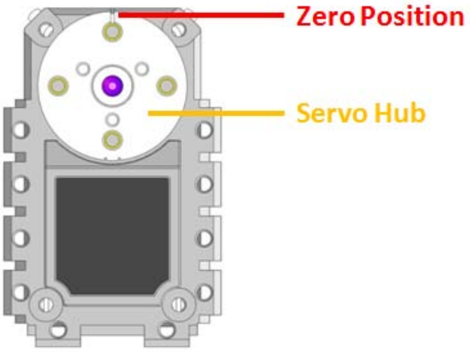


The A1-16 servos communicate with the main controller by daisy chain connection. Many A1-16 servos could be controlled by one single bus as shown above. Main controller provides power and sends control signal to A1-16 and receives respective data through the same bus. Every A1-16 servo has its unique ID value and communicates with the main controller by it, so user should be sure with the right ID before assembly. When power is successfully applied to A1-16, the status LED blinks in sequence with red, white, blue and green LED twice.

The pin assignment of A1-16 is described as below. Each pin of two connector is internal connected. So A1-16 could function with any connector attached.

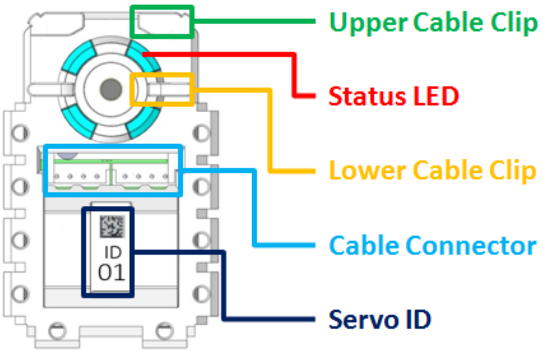


Front View of A1-16



1. Servo Hub: The servo hub is the rotation output part of A1-16.
2. Zero Position: The zero position shows the central position of A1-16 servo hub.

Back View of A1-16



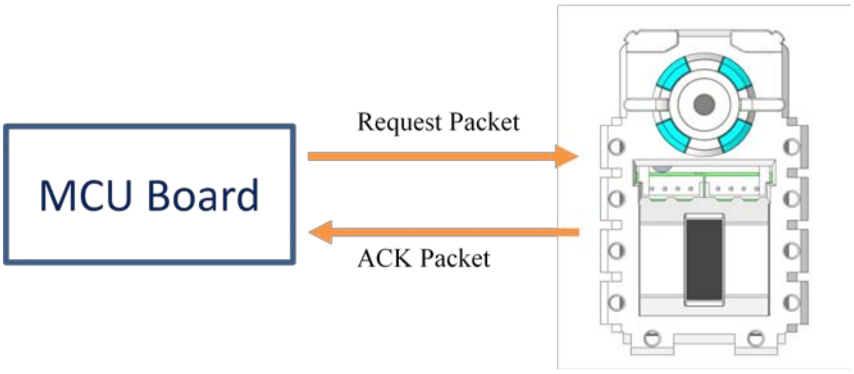
1. Cable Clip: The cable clip provide a route for cable.
2. Status LED: The status LED could indicate different error status to the users. The detail error information shows below.

Status Error	Error LED on/off
Normal Operation	White LED on
Exceed Potentiometer Range Error	Blue LED on
Over Voltage/Temperature/Current Limits Error	Red LED on/ White LED off
Requested Packet Error	Green LED on

3. Cable Connector: The cable connector provides power and communication signal for A1-16.
4. Servo ID: The servo ID shows default ID of A1-16.

Requested and ACK Packets

Main controller communicates with the servos in the UART network by sending a requested packet and receiving ACK packet back from the servo. Regardless of the number of servos in the network, only the servo with correct ID will acknowledge request packet and send the ACK packet to the main controller.



There are 9 UART command packets, as listed below, can be send from the master to servo controllers:

(1) EEP_WRITE	EEPROM parameters write
(2) EEP_READ	EEPROM parameters read
(3) RAM_WRITE	RAM parameters write
(4) RAM_READ	RAM parameters read
(5) I_JOG	independent control move
(6) S_JOG	synchronous control move
(7) STAT	read servo status
(8) ROLLBACK	reset all parameters to default values
(9) REBOOT	reset servo.

The servo controller may report ACK packets accordingly. The detail description of Requested and ACK packets are explained in Table1 through Table 9.

Table 1: Requested and ACK packets data string

bytes	1	2	3	4	5	6	7	8~107
description	header	header	packet size N	packet ID	CMD	check_ sum_1	check_ sum_2	data[i]
Requested packet	0xFF	0xFF	7~107	1~20, 254 (#)	0x01~ 0x09	(*)	(**)	...
ACK packet	0xFF	0xFF	7~107	1~20	0x40~ 0x49	(*)	(**)	...

Note: (#) When packet ID=254, broadcast ID, none of any servo will send ACK packet

(*) check_sum_1 = (N^ID^CMD^data[0]^data[1]^...^data[N-8]) & 0xFE

(**) check_sum_2 = (~check_sum_1) & 0xFE

Table 2: Requested and ACK packets CMDs

Requested packet CMD								
EEP_ WRITE	EEP_ READ	RAM_ WRITE	RAM_ READ	I_JOG	S_JOG	STAT	ROLLBACK	REBOOT
0x01	0x02	0x03	0x04	0x05	0x06	0x07	0x08	0x09
ACK packet CMD								
EEP_ WRITE	EEP_ READ	RAM_ WRITE	RAM_ READ	I_JOG	S_JOG	STAT	ROLLBACK	REBOOT
0x41	0x42	0x43	0x44	0x45	0x46	0x47	0x48	0x49

Table 3: Requested and ACK packets for EEP_WRITE(0x01) and RAM_WRITE(0x03)

(1) Requested packet for EEP_WRITE and RAM_WRITE CMD

1	2	3	4	5	6	7	8	9	+L
header	header	size	ID	CMD	check_sum_1	check_sum_2	start addr.	length L	data[i]
0xFF	0xFF	9+L	1~20, 254	0x01, 0x03	0xXX	0xXX	0xXX	0xXX	...

Note: EEP_WRITE: 4 <= start addr. <= 53, , 5 <= start adds. + length <= 54

RAM_WRITE: 0 <= start addr. <= 47, 1 <= start adds. + length <= 48

(2) ACK packet for EEP_WRITE and RAM_WRITE CMD

1	2	3	4	5	6	7	8	9
header	header	size	ID	CMD	check_sum_1	check_sum_2	status error	Status detail
0xFF	0xFF	9	1~20	0x41, 0x43	0xXX	0xXX	0xXX	0xXX

Note: The status_error and status_detail are listed in RAM parameters section.

Table 4: Requested and ACK packets for EEP_READ(0x02) and RAM_READ(0x04) CMD

(1) Requested packet for EEP_READ and RAM_READ CMD

1	2	3	4	5	6	7	8	9
header	header	size	ID	CMD	check_sum_1	check_sum_2	start addr.	length L
0xFF	0xFF	9	1~20	0x02, 0x04	0xXX	0xXX	0xXX	0xXX

Note: EEP_READ: 0 <= start addr. <= 53, , 1 <= start adds. + length <= 54

RAM_READ: 0 <= start addr. <= 79, 1 <= start adds. + length <= 80

(2) ACK packet for EEP_READ and RAM_READ CMD

1	2	3	4	5	6	7	8	9	+L
header	header	size	ID	CMD	check_sum_1	check_sum_2	start addr.	length L	data[i]
0xFF	0xFF	9+L	1~20	0x42, 0x44	0xXX	0xXX	0xXX	0xXX	...

Table 5: Requested and ACK packets for I-JOG (0x05) CMD

(1) Requested packet for I-JOG CMD

1	2	3	4	5	6	7	+5	+(n-1)*5
header	header	size	ID	CMD	check_ sum_1	check_ sum_2	I-JOG data	...
0xFF	0xFF	7+5*n	1~20, 254	0x05	0xXX	0xXX	(5-1)	...

Note: n=number of motor IDs send.

(5-1) I-JOG data

1	2	3	4	5
goal.lsb	goal.msb	set	ID	playtime (unit:10 msec)
0xXX	0xXX	0xXX	1~20	0xXX

Note: (1) goal=0~1023; (2) play time may be modified for a long movement;
(3) set = 0 (position control) / 1 (speed control) / 2 (torque off) /3 (position control servo on)

(2) ACK packet for I-JOG CMD

1	2	3	4	5	6	7	8	9
header	header	size	ID	CMD	check_ sum_1	check_ sum_2	status error	status detail
0xFF	0xFF	9	1~20	0x45	0xXX	0xXX	0xXX	0xXX

Table 6: Requested and ACK packets for S-SOG (0x06) CMD

(1) Requested packet for S-JOG CMD

1	2	3	4	5	6	7	8	+4	+(n-1)*4
header	header	size	ID	CMD	check_ sum_1	check_ sum_2	play time	S-JOG data	...
0xFF	0xFF	8+4*n	1~20, 254	0x06	0xXX	0xXX	0xXX	(6-1)	...

Note: n=number of motor IDs send.

(6-1) S-JOG data

1	2	3	4
goal.lsb	goal.msb	set	ID
0xXX	0xXX	0xXX	1~20

Note: (1) goal=0~1023; (2) goal position may not be reached for a short play time;
(3) set = 0 (position control) / 1 (speed control) / 2 (torque off) / 3 (position control servo on)

(2) ACK packet for S-JOG CMD

1	2	3	4	5	6	7	8	9
Header	header	size	ID	CMD	check_ sum_1	check_ sum_2	status error	Status detail
0xFF	0xFF	9	1~20	0x46	0xXX	0xXX	0xXX	0xXX

Table 7: Requested and ACK packets for STAT(0x07) CMD

(1) Requested packet for STAT CMD

1	2	3	4	5	6	7
header	header	size	ID	CMD	check_sum_1	check_sum_2
0xFF	0xFF	7	1~20	0x07	0xXX	0xXX

(2) ACK packet for STAT CMD

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
header	header	size	ID	CMD	check_sum_1	check_sum_2	status_error	status_detail	PWM.lsb	PWM.msb	pos_ref.lsb	pos_ref.msb	position.lsb	position.msb	Ibus.lsb	Ibus.msb
0xFF	0xFF	17	1~20	0x47	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX	0xXX

Table 8: Requested and ACK packets for ROLLBACK(0x08) CMD

(1) Requested packet for ROLLBACK CMD

1	2	3	4	5	6	7
header	header	size	ID	CMD	check_sum_1	check_sum_2
0xFF	0xFF	7	1~20, 254	0x08	0xXX	0xXX

(2) ACK packet for ROLLBACK CMD

1	2	3	4	5	6	7	8	9
header	header	size	ID	CMD	check_sum_1	check_sum_2	status_error	status_detail
0xFF	0xFF	9	1~20	0x48	0xXX	0xXX	0xXX	0xXX

Table 9: Requested and ACK packets for REBOOT(0x09) CMD

(1) Requested packet for REBOOT CMD

1	2	3	4	5	6	7
header	header	size	ID	CMD	check_sum_1	check_sum_2
0xFF	0xFF	7	1~20, 254	0x09	0xXX	0xXX

(2) ACK packet for REBOOT CMD

1	2	3	4	5	6	7	8	9
header	header	size	ID	CMD	check_sum_1	check_sum_2	status_error	status_detail
0xFF	0xFF	9	1~20	0x49	0xXX	0xXX	0xXX	0xXX

3. EEPROM & RAM Parameters

The system parameters saved in EEOROM and RAM are shown in Table 10. There are 54 bytes parameter data in EEPROM and 80 bytes parameter data in RAM, in which the first 48 bytes of RAM data are same as the data in RAM from address of 6 to 54. The EEPROM data can be read and written to; some of the RAM data are read only.

Table 10: EEPROM & RAM Parameters

EEPROM Addr.	RAM Addr.	Parameter	Bytes	R/W /RO	Default Value
0		Model_No	1	RO	0x01
1		Year	1	RO	0x0F
2		Version/Month	1	RO	0x3A
3		Day	1	RO	0x01
4		Reserved	1	RO	0x01
5		Baud_Rate	1	R/W	0x0C
6	0	sID	1	R/W	0x01
7	1	ACK_Policy	1	R/W	0x02
8	2	Alarm_LED_Policy	1	R/W	0x00
9	3	Torque_Policy	1	R/W	0x01
10	4	SPDctrl_Policy	1	R/W	0x01
11	5	Max_Temperature	1	R/W	0x4B
12	6	Min_Voltage	1	R/W	0x77
13	7	Max_Voltage	1	R/W	0xE8
14	8	Acceleration_Ratio	1	R/W	0x00
15	9	Reserved	1	R/W	0xFF
16	10	Reserved	1	R/W	0x00
17	11	Reserved	1	R/W	0x00
18	12	Max_Wheel_Ref_Position	2	R/W	0x042E
20	14	Reserved	1	R/W	0x00
21	15	Reserved	1	R/W	0x00
22	16	Max_PWM	2	R/W	0x03FF
24	18	Overload_Threshold	2	R/W	0x00CC
26	20	Min_Position	2	R/W	0x00
28	22	Max_Position	2	R/W	0x03FF
30	24	Position_Kp	2	R/W	0x0F00
32	26	Position_Kd	2	R/W	0x0800

34	28	Position_Ki	2	R/W	0x0000
36	30	Close_to_Open_Ref_Position	2	R/W	0x03FF
38	32	Open_to_Close_Ref_Position	2	R/W	0x00
40	34	Reserved	2	R/W	0x03FF
42	36	Ramp_Speed	2	R/W	0x03FF
44	38	LED_Blink_Period	1	R/W	0x00
45	39	Reserved	1	R/W	0x00
46	40	Packet_Timeout_Detection_Period	1	R/W	0x0A
47	41	Reserved	1	R/W	0x00
48	42	Overload_Detection_Period	1	R/W	0x19
49	43	Reserved	1	R/W	0x00
50	44	Inposition_Margin	1	R/W	0x01
51	45	Over_Voltage_Detection_Period	1	R/W	0xFF
52	46	Over_Temperature_Detection_Period	1	R/W	0x0A
53	47	Calibration_Difference	1	R/W	0xXX
	48	Status_Error	1	R/W	0x00
	49	Status_Detail	1	R/W	0x40
	50	Reserved	1	R/W	0x00
	51	Reserved	1	R/W	0x00
	52	Reserved	1	R/W	0x01
	53	LED_Control	1	R/W	0x00
	54	Voltage	1	RO	0xXX
	55	Temperature	1	RO	0xXX
	56	Current_Control_Mode	1	RO	0x02
	57	Tick	1	RO	0x00
	58	Reserved	2	RO	0xXXXX
	60	Joint_Position	2	RO	0xXXXX
	62	Reserved	2	RO	0x0000
	64	PWM_Output_Duty	2	RO	0x0000
	66	Bus_Current	2	RO	0x0000
	68	Position_Goal	2	RO	0xXXXX
	70	Position_Ref	2	RO	0xXXXX
	72	Omega_Goal	2	RO	0x0000
	74	Omega_Ref	2	RO	0x0000
	76	Requested_Counts	2	RO	0x0000
	78	ACK_Counts	2	RO	0x0000

The description of EEPROM and RAM parameters above are summarized below.

(E0) Model_No：Servo model name

(E1) Year：Year

(E2) Version/Month：bit0~3：month, bit4~8：version of servo firmware

(E5) Baud_Rate：

0x01：9600

0x02：19200

0x06：57600

0x0C：115200

(E6,R0) sID：Servo ID, 1, 2, ..., 19, 20 ... , 253

(E7,R1) ACK_Policy：

only STAT command reply：0

only EEPROM/RAM RAED and STAT commands reply：1

all commands reply：2

(E8,R2) Alarm_LED_Policy：bit i = 0 (System Alarm LED), 1 (User LED)

Bit 0：White LED

Bit 1：Blue LED

Bit 2：Green LED

Bit 3：Red LED

(E9,R3) Torque_Policy：Shut down Motor when Voltage/Load/Temperature

Torque Free Control：0

Torque Limited：1

(E10,R4) SPDctrl_Policy：Speed open/close loop control

Open Loop Control：0

Close Loop Control：1

(E11,R5) Max_Temperature：The limit of A1-16 servo operating temperature. The value is in Degrees Celsius.

(E12,R6) Min_Voltage：The min value of A1-16 servo operating voltage. The value is 16 times the actual voltage.

(E13,R7) Max_Voltage：The max value of A1-16 servo operating voltage. The value is 16 times the actual voltage.

(E14,R8) Acceleration_Ratio = 0, 1, 2, ..., 50

Play_time	Acceleration_Ratio	Referenced position trajectory
0		Ramp-to-step position command, see (36)
1~255	0	Constant speed profile
1~255	1~50	T-curve speed profile

Note: acceleration_time = deceleration_time = play_time * Acceleration_Ratio/100

(E18,R12) Max_Wheel_Ref_Position：Start virtual position for speed close loop control.

(E22,R16) Max_PWM：The max value of A1-16 servo output torque.

(E24,R18) Overload_Threshold：The max value of A1-16 servo output torque.

(E26,R20) Min_Position：Min operational angle

(E28,R22) Max_Position：Max operational angle

(E30,R24) Position_Kp：msb is the integer number and lsb is the decimal number. The P control law is implemented below with a sampling time of 10 msec

(E32,R26) Position_Kd：msb is the integer number and lsb is the decimal number. The PD control law is implemented below with a sampling time of 10 msec

(E34,R28) Position_Ki：msb is the integer number and lsb is the decimal number. The PID control law is implemented below with a sampling time of 10 msec,

(E36,R30) Close_to_Open_Ref_Position：close loop continuous rotate mode close to open position.

(E38,R32) Open_to_Close_Ref_Position：close loop continuous rotate mode open to close position.

(E42,R36) Ramp_Speed = 0 (step position command), 1~1023 (slope of ramp-to-step)

(E44,R38) LED_Blink_Period：Blinking Period of LED with a sampling time of 10 msec.

(E46,R40) Packet_Timeout_Detection_Period：Packet Timeout Detection Period of LED with a sampling time of 10 msec. 1 = 10ms

(E48,R42) Overload_Detection_Period：Overload Detection Period of servo with a sampling time of 10 msec. 1 = 10ms

(E51,R45) Over_Voltage_Detection_Period：Over Voltage Detection Period of servo with a sampling time of 10 msec. 1 = 10ms

(E52,R46) Over_ Temperature _Detection_Period：Over Temperature Detection Period of servo with a sampling time of 10 msec. 1 = 10ms

(E53,R47) Calibration_Difference：The difference between newtral point and position raw data.

(R48) status_error

bit	Mask	Default	Status Error	Error LED on/off
1	0x01	0	Exceed Potentiometer Range Error	Blue LED on
2	0x02	0	Over Voltage Limits Error	Red LED on/ White LED off
3	0x04	0	Over Temperature Error	Red LED on/ White LED off
4	0x08	0	Overload/Over-current Error	Red LED on/ White LED off
5	0x10	0	Reserved	None

6	0x20	0	Requested Packet Checksum Error	Green LED on
7	0x40	0	Requested Packet Data Error	Green LED on
8	0x80	0	Requested Packet RX FIFO Error	Green LED on

(R49) status_detail

bit	Mask	Default	Status Detail
1	0x01	0	Reserved
2	0x02	0	Reserved
3	0x04	0	Reserved
4	0x08	0	Reserved
5	0x10	0	Motor Moving
6	0x20	0	Motor In-Position (Position control mode only)
7	0x40	0	1: Torque on (Position/Speed control), 0: Torque off
8	0x80	0	Motor Braked

(R53) LED_Control：bit i = 0 (LEDi off), 1 (LEDi on); (see Alarm_LED_Policy)

Bit 0：White LED

Bit 1：Blue LED

Bit 2：Green LED

Bit 3：Red LED

(R54) Voltage：The voltage currently applied to servo. The Value is 16 times the actual voltage.

(R55) Temperature：The internal temperature of motor in Degrees Celsius.

(R56) Current_Control_Mode：0 (position control), 1 (speed control), 2 (torque off)

(R57) Tick：Time servo operation. 1 = 10ms

(R60) Joint Postion：Servo Position

(R64) PWM_Output_Duty：The torque applied to motor

(R66) Bus_Current：The Current applied to motor. The Value is 200 times the actual current.

(R68) Position_Goal：Servo goal of position control mode

(R70) Position_Ref：Ref point for position control

(R72) Omega_Goal：Goal speed of speed close-loop control

(R74) Omega_Ref：Ref speed of speed close-loop control

(R76) Requested_Counts：Total # of requested packets received since power on.

(R78) ACK_Counts：Total # of ACK packets send since power on.