WHILL Control System Protocol Specification for Omni Platform

Contents:

1. Preface

This document describes the specification of the communication protocol and control system related to WHILL (Omni Platform) of WHILL, Inc.

2. Communication Overview

Communication interface between the host device and WHILL is RS232C.
In basic communication, the control command is sent from the host device to WHILL and WHILL takes actions according to the control command. The WHILL state data will be periodically sent from WHILL to the host device after WHILL receives the control command to send the WHILL state data. The WHILL state data includes speed of WHILL, battery level, battery current and so on. Figure 1 shows the communication overview.

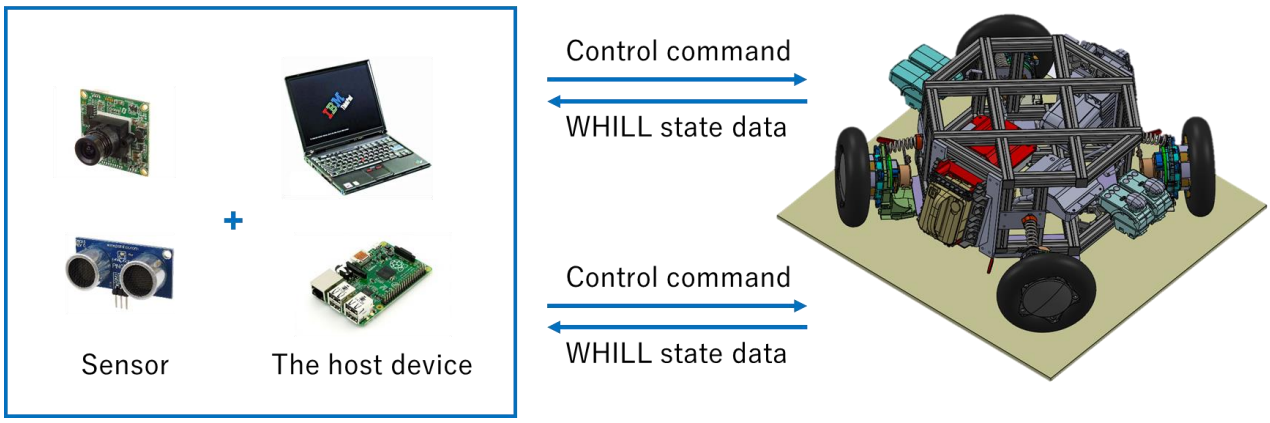

Figure 1 : Communication overview

Attention:
Since the Omni Platform is equipped with two motor controllers, it is equipped with two communication interfaces defined in this document, as shown in Figure 1.
To drive the four wheels for parallel movement and turning, synchronous control of these two systems of RS232C is required. Please refer to Appendix for specific usage information.

3. Control Command

3.1. Communication Format

Figure 2 shows the communication format of the control command. The communication format includes protocol sign, data length, control command and checksum.
Protocol sign: At first, protocol sign is sent from the host device. WHILL recognizes the start of control command by protocol sign. Protocol sign is always 0xAF.
Data length: Data length shows byte size behind itself, not includes protocol sign and data length.
Control Command: Control command consists of command ID and command data.
Checksum: Checksum is the value of XOR of protocol sign, data length and control command (command ID and all command data).

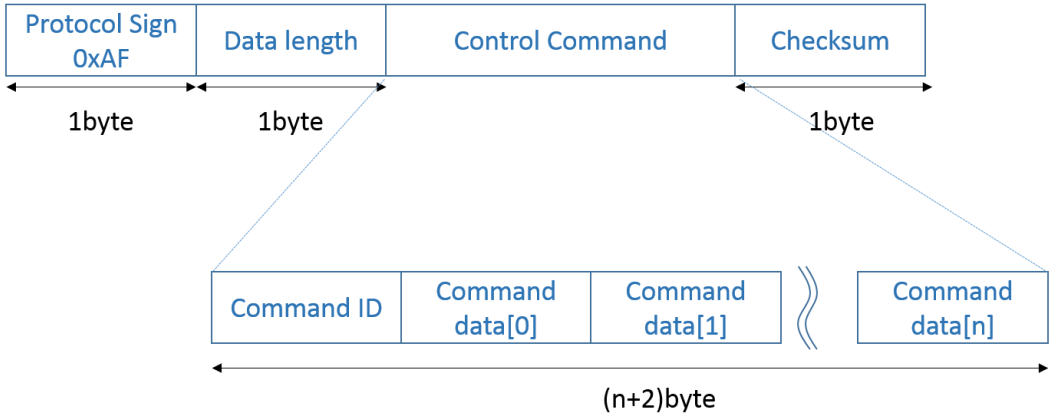

Figure 2 : Communication format of the control command

3.2. Control Commands

The host device can control WHILL using control commands. Table 1 is the list of control command. Each command has a command ID and command data. The detail of each command is described from next section.

Table 1 : List of control commands

Command	Command ID	Description	Implementation
StartSendingData	0x00	WHILL starts sending WHILL state data	✓
StopSendingData	0x01	WHILL stops sending WHILL state data	✓
SetPower	0x02	Power On/Off WHILL	✓
SetJoystick	0x03	Disable/Enable joystick control and set joystick value	✓
SetSpeedProfile	0x04	Set max speed, acceleration and deceleration	✓
reserve	0x05	Don't use this Command ID	-
reserve	0x06	Don't use this Command ID	-
reserve	0x07	Don't use this Command ID	-
SetVelocity	0x08	Set velocity of WHILL	✓

3.2.1. StartSendingData command

StartSendigData command makes WHILL start sending the WHILL state data. Figure 3 shows StartSendingData command format.

Command ID: 0x00

D0: Data set number.

D0 = 0: Select data set 0

D1 = 1: Select data set 1

T0~T1: Interval of sending data [ms] (16bit value). T1:MSB8bit, T0:LSB8bit. Minimum interval is 10ms.

S0: Speed mode. Set speed mode in data set 0. See section4.2 Data set 0 for more detail.

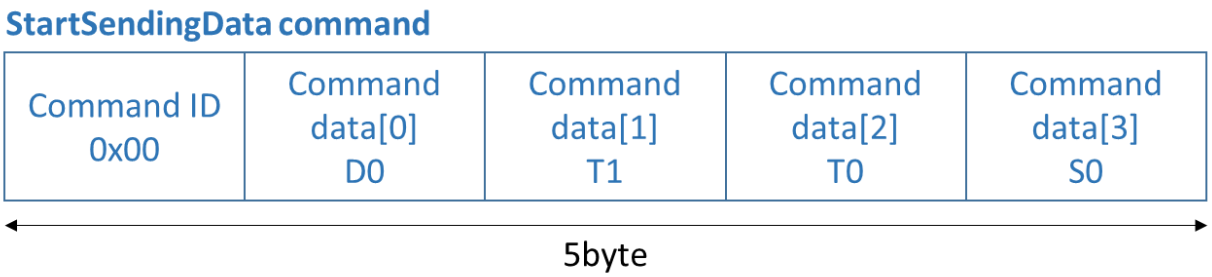

Figure 3 : StartSendingData format

3.2.2. StopSendingData command

StopSendigData command makes WHILL stop sending the WHILL state data. Figure 4 shows StopSendingData command format.
Command ID: 0x01

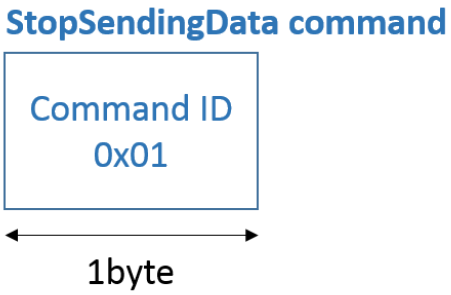

Figure 4 : StopSendingData format

3.2.3. SetPower

Function of SetPower command is to power ON/OFF WHILL. When WHILL finishes to power on after receiving SetPower command, WHILL sends Response data to the host device. Before receiving the response data, don't send any commands to WHILL. If the response data isn't sent for 5ms, re-issue SetPower Command to WHILL. See section5 WHILL Response Data for more detail. Figure 5 shows SetPower command format.
Command ID: 0x02
P0: Select Power ON or OFF
P0 = 0: Power OFF
P0 = 1: Power ON

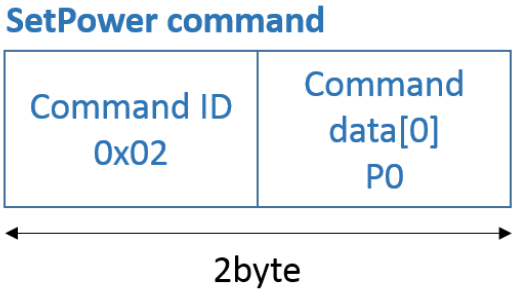

Figure 5 : SetPower command format

Note: After sending "POWER OFF", the host device must wait for more than 5s to send "POWER ON".

3.2.4. SetJoystick command

SetJoystick command has two functions. First function is to enable/disable the joystick control of a user who rides WHILL and disable/enable the joystick control of the host device. When the joystick control of a user is enabled (disabled), the joystick control of the host device is disabled (enabled). Second function is to set the value of joystick. The value is valid only when the joystick control of the host device is enabled. With a single SetJoystick command, the setting value is available for 200ms.Keep sending the command within 200ms if keeping WHILL moving more than 200ms.
Figure 6 shows SetJoystick command format.
Command ID: 0x03
U0: Disable/Enable joystick control of user/host device.
U0 = 0: Disable joystick control of user and enable joystick control of the host device
U0 = 1: Enable joystick control of user and disable joystick control of the host device
FB0 : Joystick value in front/back direction（-100～100）
LR0: Joystick value in left/right direction（-100～100）

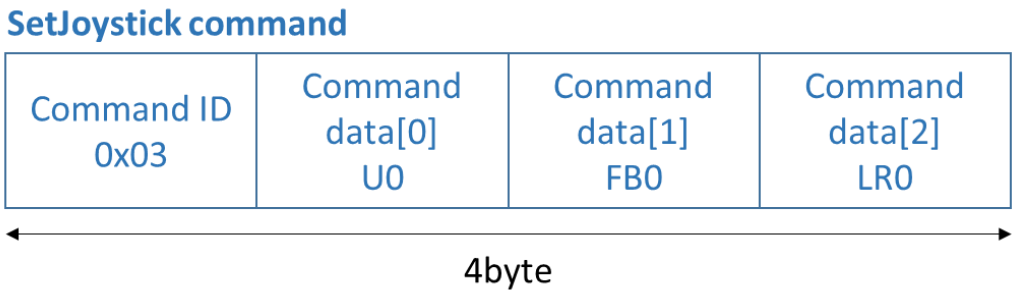

Figure 6 : SetJoystick command format

3.2.5. SetSpeedProfile command

SetSpeedProfile command is to set max speed, acceleration and deceleration at forward movement, reverse movement and turn movement in each speed mode.
Figure 7 shows SetSpeedProfile command format.
Command ID: 0x04
S1: Speed mode: Select speed mode to set
S1 = 0: Select speed mode when user uses joystick and “1” is displayed in indicator of WHILL
S1 = 1: Select speed mode when user uses joystick and “2” is displayed in indicator of WHILL
S1 = 2: Select speed mode when user uses joystick and “3” is displayed in indicator of WHILL

S1 = 3: Select speed mode when user uses joystick and “4” is displayed in indicator of WHILL

S1 = 4: Select speed mode when host device moves WHILL via RS232C

S1 = 5: Select speed mode when iPhone WHILL App moves WHILL

F_M1: Max speed at forward movement. The unit of value is 0.1km/h. If 30 is set, max speed is 3.0km/h. The range is (8~60).

F_A1: Acceleration at forward movement. The range is (10~90).

F_D1: Deceleration at forward movement. The range is (40~160).

R_M1: Max speed at reverse movement. The unit of value is 0.1km/h. If 30 is set, max speed is 3.0km/h. The range is (8~60).

R_A1: Acceleration at reverse movement. The range is (10~90).

R_D1: Deceleration at reverse movement. The range is (40~160).

T_M1: Max speed at turn movement. The unit of value is 0.1km/h. If 30 is set, max speed is 3.0km/h. The range is (8~60).

T_A1: Acceleration at turn movement. The range is (10~90).

T_D1: Deceleration at turn movement. The range is (40~160).

SetSpeedPrfile command

Command ID 0x04	Command data[0] S1	Command data[2] F_M1	Command data[3] F_A1	Command data[4] F_D1

Command data[5] R_M1	Command data[6] R_A1	Command data[7] R_D1	Command data[8] T_M1	Command data[9] T_A1	Command data[10] T_D1

11byte

Figure 7 : SetForward command format

3.2.6. SetVelocity command

SetVelocity command has two functions. First function is to enable/disable the joystick control of a user who rides WHILL and disable/enable the control of the host device. When the joystick control of a user is enabled (disabled), the control of the host device is disabled (enabled). Second function is to set the value of WHILL velocity. The value is valid only when control of the host device is enabled. With a single SetVelocity command, the setting value is available for 200ms.Keep sending the command within 200ms if keeping WHILL moving more than 200ms.

[Difference from SetJoystick command]

Using SetVelocity command, the host device can send velocity value directly independently of speed profile setting. Also, a finer speed can be set using SetVelocity command than using SetJoystick command.
The acceleration is const when the host device uses SetVelocity command. The acceleration is fast (1.7m/s2) in order for WHILL to follow the target speed as quick as possible.
WHILL moves so quickly using SetVelocity command and so pay enough attention to use SetVelocity command. Basically, send this command to increase speed gradually.
Figure 8 shows SetVelocity command format.
Command ID: 0x06
U0: Disable/Enable joystick control of user/host device.
U0 = 0: Disable joystick control of user and enable joystick control of the host device
U0 = 1: Enable joystick control of user and disable joystick control of the host device
Y0~1: Velocity value in in front/back direction (16bit value). The range is（-1500～1500）.The unit is 0.004[km/h]. Y1:MSB8bit, Y0:LSB8bit.
X0~1: Velocity value in in left/right direction (16bit value). The range is（-1500～1500）.The unit is 0.004[km/h]. X1:MSB8bit, X0:LSB8bit.

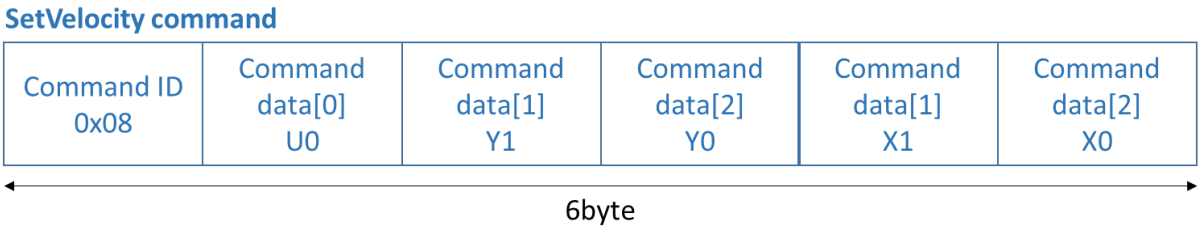

Figure 8 : SetVelocity command format

4. WHILL state data

4.1. Communication format

After receiving StartSendingData command, WHILL starts to send the WHILL state data to the host device. The host device selects the data set number of the WHILL state data. According to the data set number, contents which WHILL sends changes. The WHILL state data conforms to the communication format shown in Figure 9.
Protocol sign: At first, protocol sign is sent from WHILL. Protocol sign indicates the start of the WHILL state data. Protocol sign is always 0xAF.
Data length: Data length shows byte size behind itself, not includes protocol sign and data length.

WHILL state data: WHILL state data consists of data set number and information.
Checksum: Checksum is the value of XOR of protocol sign, data length and WHILL state data.

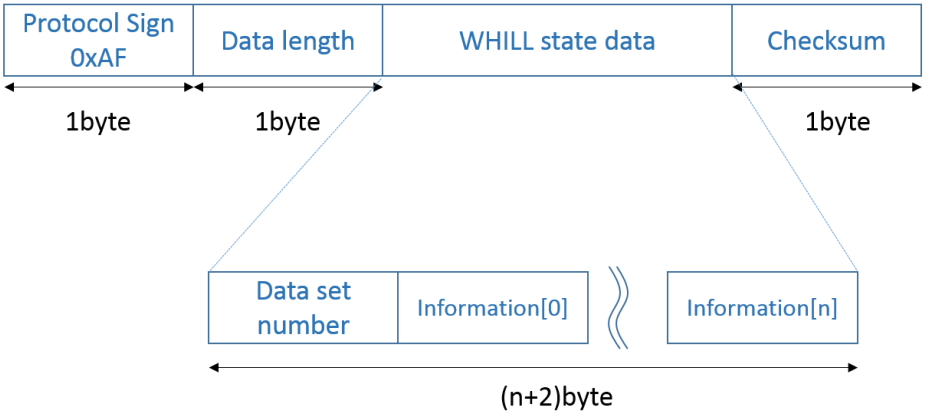

Figure 9 : Communication format of WHILL state data

4.2. Data set 0

1.1.1 When the host device select data set 0 in StartSendingData command, values of Table 2 will be sent from WHILL. This data is about speed profile. This data set will be changed only when the host device modifies the setting of WHILL. So, it is better to read the data set 0 once when WHILL power-on or the host device changes the settings of max speed, acceleration and deceleration. See section 3.2.5 SetSpeedProfile command to know items in table 2.

Table 2 : Data set 0 contents

Information Number	Value (8bit)	Description	Implementation
0	SPEED_MODE	Speed mode to send	✓
1	FORWARD_SPEED_MAX	Max speed at forward movement	✓
2	FORWARD_ACCEL	Acceleration at forward movement	✓
3	FORWARD_DECEL	Deceleration at forward movement	✓
4	REVERSE_SPEED_MAX	Max speed at reverse movement	✓
5	REVERSE_ACCEL	Acceleration at reverse movement	✓
6	REVERSE_DECEL	Deceleration at reverse movement	✓
7	TURN_SPEED_MAX	Max speed at turn movement	✓
8	TURN_ACCEL	Acceleration at turn movement	✓

9	TURN_DECEL	Deceleration at turn movement	✓

4.3. Data set 1

When the host device selects data set 1 in StartSendingData command, values of Table 3 will be sent from WHILL. This data set is changing continuously such as acceleration sensor value, battery current and so on.

Table 3 : Data set 1 contents

Information Number	Value (8bit)	Description	Implementation
0-13	Reserved	Always '0'	-
14	BATTERY_POWER	Battery level. 0 ~ 100%	✓
15	BATTERY_CURRENT (MSB 8bit)	Battery current (MSB 8bit) See Note3	✓
16	BATTERY_CURRENT (LSB 8bit)	Battery current (LSB 8bit) See Note3	✓
17	RIGHT_MOTOR_ANGLE (MSB 8bit)	Angle of right motor (MSB 8bit). See Note4	✓
18	RIGHT_MOTOR_ANGLE (LSB 8bit)	Angle of right motor (LSB 8bit). See Note4	✓
19	LEFT_MOTOR_ANGLE (MSB 8bit)	Angle of left motor (MSB 8bit). See Note4	✓
20	LEFT_MOTOR_ANGLE (LSB 8bit)	Angle of left motor (LSB 8bit). See Note4	✓
21	RIGHT_MOTOR_SPEED (MSB 8bit)	Right motor speed value (MSB 8bit). See Note5	✓
22	RIGHT_MOTOR_SPEED (LSB 8bit)	Right motor speed value (LSB 8bit). See Note5	✓
23	LEFT_MOTOR_SPEED (MSB 8bit)	Left motor speed value (MSB 8bit). See Note5	✓
24	LEFT_MOTOR_SPEED (LSB 8bit)	Left motor speed value (LSB 8bit). See Note5	✓
25	POWER_ON	State of WHILL power. ON:1, OFF:0	✓
26	SPEED_MODE_INDICATOR	Speed mode displayed on indicator of WHILL	✓
27	ERROR	Error code on WHILL. If there is no error, '0' is sent.	✓
28	ANGLE_DETECT_COUNTER	Counter of timing to detect motor angle.	✓

Note1: Acceleration value must be calculated below in order to be converted into unit [mg].

$$\mathrm{Vacc} \times 0.122 = \mathrm{a}$$

Vacc : Value about acceleration which is reported from WHILL. 16bit signed value.
a: Acceleration value which unit is [mg].

Note2: Angular rate value must be calculated below in order to be converted into unit [mdps].

$$\mathrm{Vang} \times 4.375 = \omega$$

Vang: Value about angular rate which is reported from WHILL. 16bit signed value.
ω : Angular rate value which unit is [mdps].

Note3: BATTERY_CURRENT is 16bit singed value. The unit is 2mA.
Example: 0x035 represents 106mA; 0xFF97 represents –210mA.
The value will be 0 if battery current is in $\pm$75mA.
Sampling rate of battery current is 4Hz.

Note4: RIGHT_MOTOR_ANGLE and LEFT_MOTOR_ANGLE is 16bit singed value. The unit is 0.001rad. The range is $\pm\pi$ [rad].
Example: 0x0600 represents 1.536rad

Note5: The value of the item that Implementation is “-” is always 0x00.

Note5: RIGHT_MOTOR_SPEED and LEFT_MOTOR_SPEED is 16bit singed value. The unit is 0.004km/h.
Example: 0x01F4 represents 2km/h.
If higher accuracy of motor speed is needed, calculate motor speed from RIGHT_MOTOR_ANGLE(LEFT_MOTOR_ANGLE) and ANGLE_DETECT_COUNTER instead of using the value of RIGHT_MOTORO_SPEED(LEFT_MOTOR_SPEED).

Note6: ANGLE_DETECT_COUNTER indicates a timing of detecting motor angles. The unit is [ms].
The max of this value is 200. After this value reaches 200, this value will be 0 in next timing of detecting motor angles.
This value is used in order to improve the accuracy of odometry.
For Example: When the following data is received sequentially, right motor rotates 0.088rad in 11ms.
RIGHT_MOTOR_ANGLE: 1.536rad and ANGLE_DETECT_COUNTER: 110
RIGHT_MOTOR_ANGLE: 1.624rad and ANGLE_DETECT_COUNTER: 121

5. Response Data

When WHILL finishes to powers on after receiving SetPower command, WHILL sends the response data to the host device. The response data is always 0x52. Figure 10 shows the response data format.

Protocol Sign: At first, protocol sign is sent from WHILL. Protocol sign is always 0xAF.
Data length: it shows byte size behind itself, not includes protocol sign and data length. It's '2' at WHILL response data.
Response: It's always 0x52
Checksum: it is the value of XOR of protocol sign, data length and response.

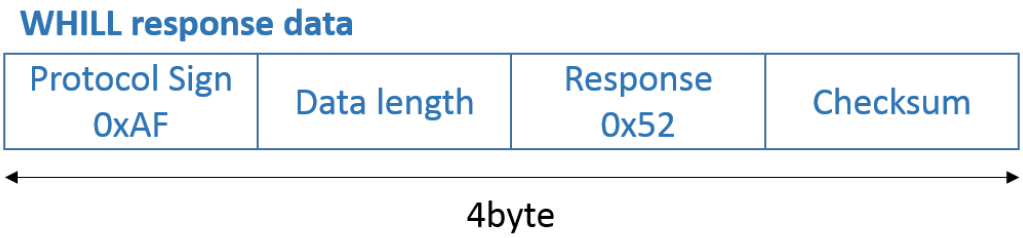

Figure 10 : WHILL response data format

After sending SetPower command to power on WHILL, don't send any commands to WHILL before receiving the response data. If the response data isn't sent for 15ms, re-issue SetPower command to WHILL (Figure 11).

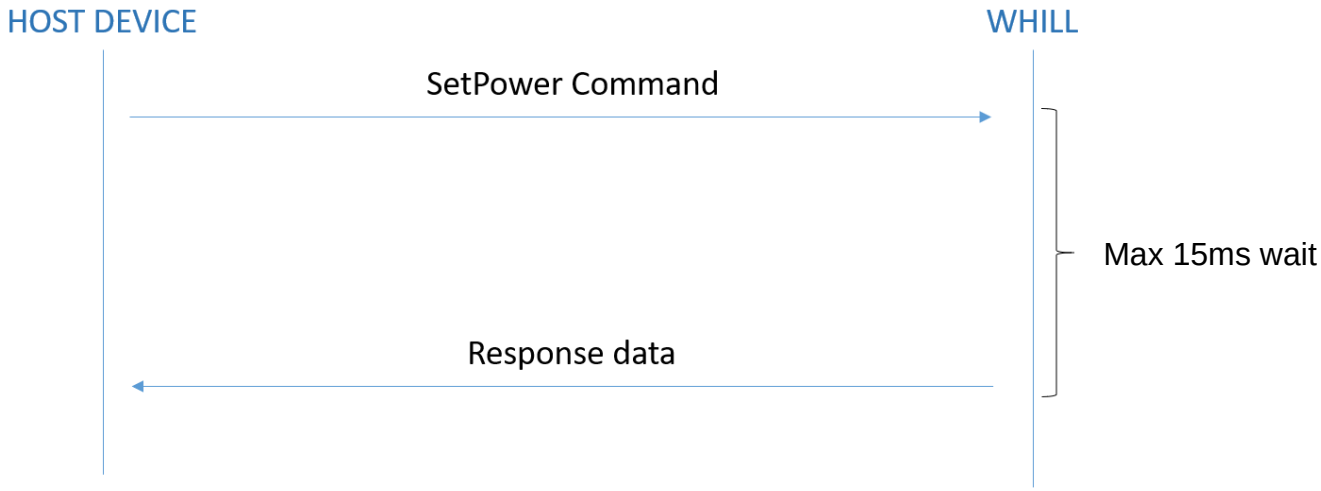

Figure 11 : Timing of Response data

6. Communication Electrical Specifications

6.1. RS232C specification

The interface of WHILL's CPU is RS232C base on Table 4. Figure 12 shows RS232C diagram.

Table 4 : UART specification

	Omni Platform
Baud rate	38400
Parity bit	No
Data length	8bit
Stop bit	2bit

Figure 12 : RS232C diagram

6.2. *Minimum time between control commands*

The host device must wait for 2ms to the send next control command after sending the previous control command (Figure 13).

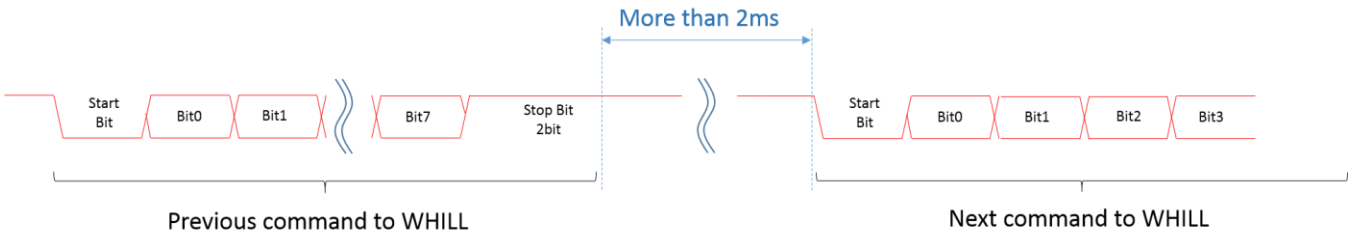

Figure 13 :Minimum time between control commands

6.3. *Maximum interval of byte data in control command*

The interval of byte data which constitutes a single control command must be less than 5ms. If the interval is more than 5ms, the previous 1 byte data will be ignored.

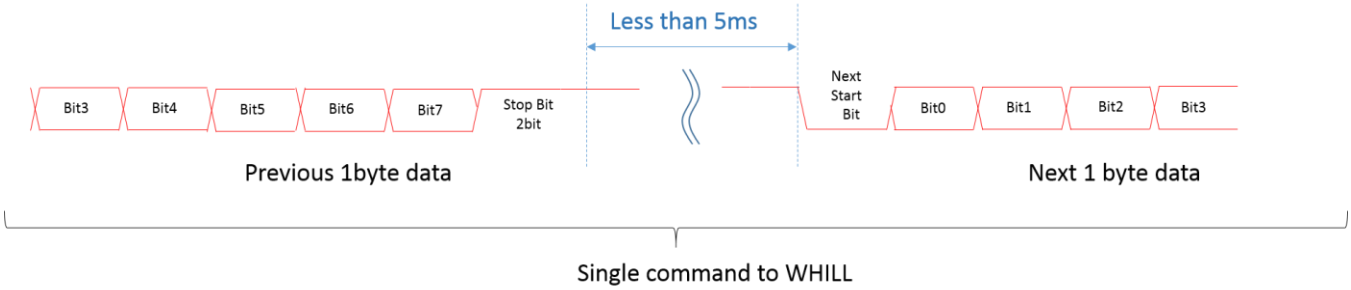

Figure 14 :Maximum interval of byte data in control command

7. Connector

7.1. Pin assignment

JEZ-9S-3 (JST) is used for interface connector. Figure 15 shows pin assignment.

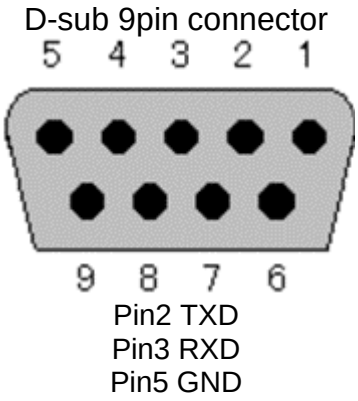

Figure 15 : Pin assignment of connector

8. Appendix

8.1. To move / rotate Omni Platform

At first, you need to use SetVelocity command for movement or rotation of Omni Platform. A set of motor (total of 2 motors on left and right) is controlled by SetVelocity command. Parameters of SetVelocity command and each motor speeds have the relationship as shown in Figure 16.

Attention: Giving the parameter where 'Y' is equal to 'X' (not 0) is not recommended as it will result in a pivot turn and increase the risk of overcurrent on the pivot side.

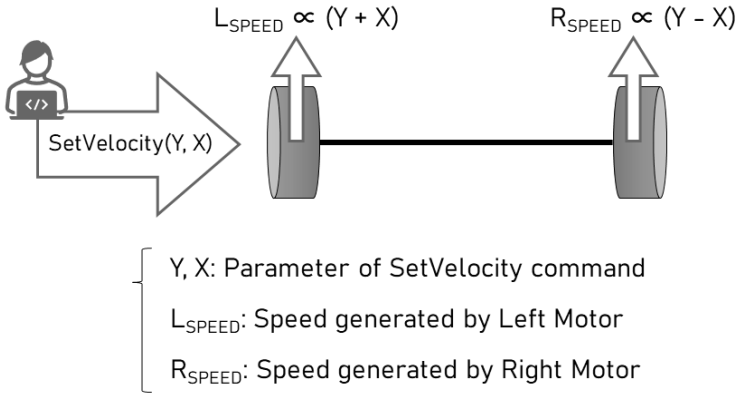

Figure 16 : Relationship between SetVelocity command and motor speed

And Omni Platform is implemented with two motor pairs orthogonal. So, the movement and rotation of Omni Platform can be achieved by combining the speeds of each motor, as shown in Figure 17.

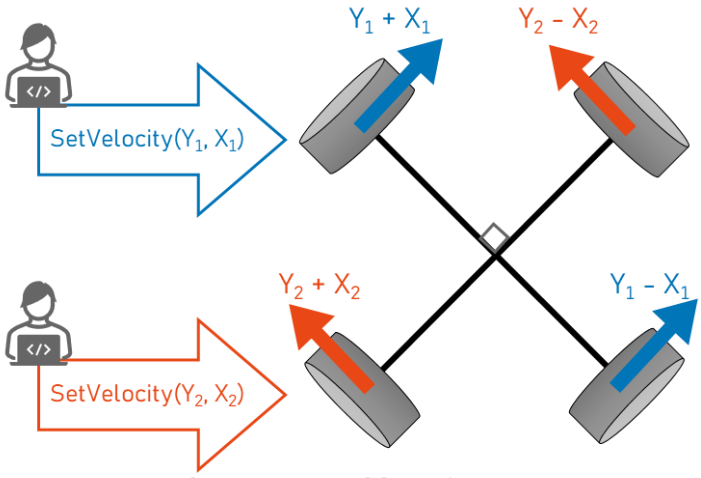

Figure 17 : Position of 4 motors

Example 1) Movement

Omni Platform moves by simultaneous input of SetVelocity command with ‘X’ is 0.

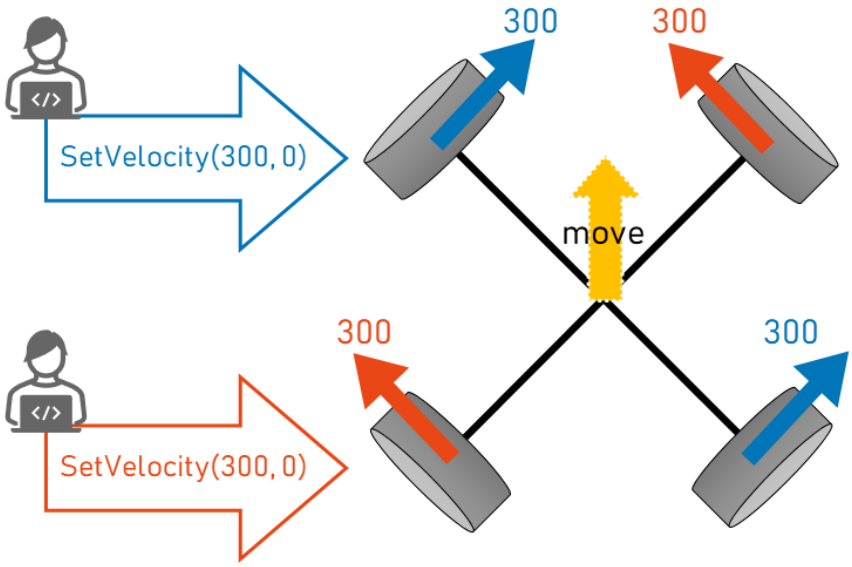

Figure 18 : Example of movement (Forward)

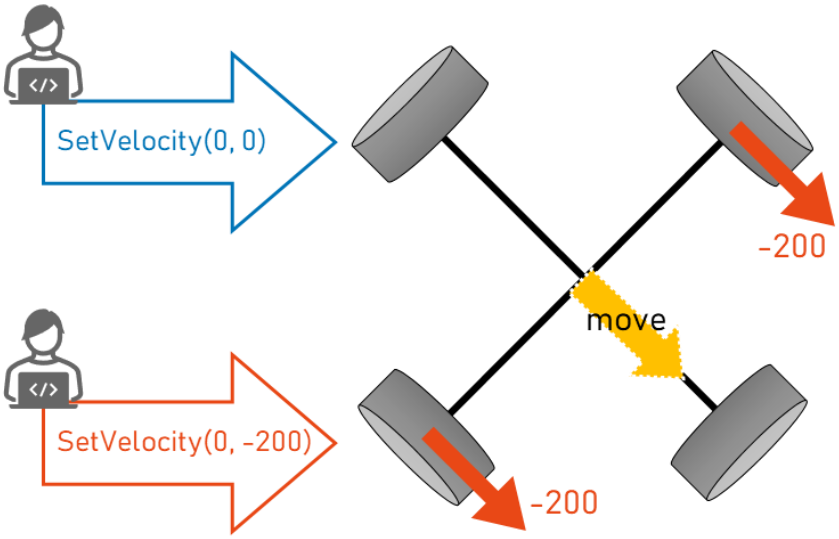

Figure 19 : Example of movement (Right backward)

Example 2) Rotation

Omni Platform rotates by simultaneous input of SetVelocity command with 'Y' is 0.

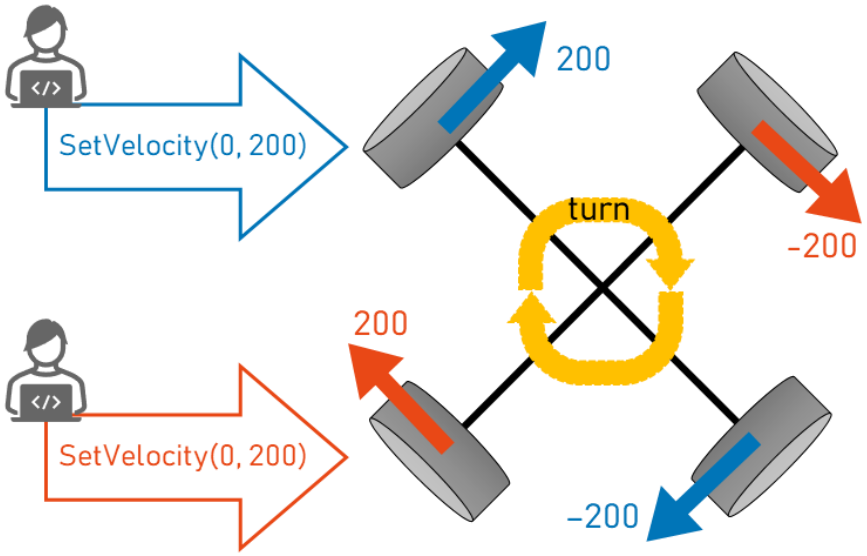

Figure 20 : Example of rotation (Clockwise)

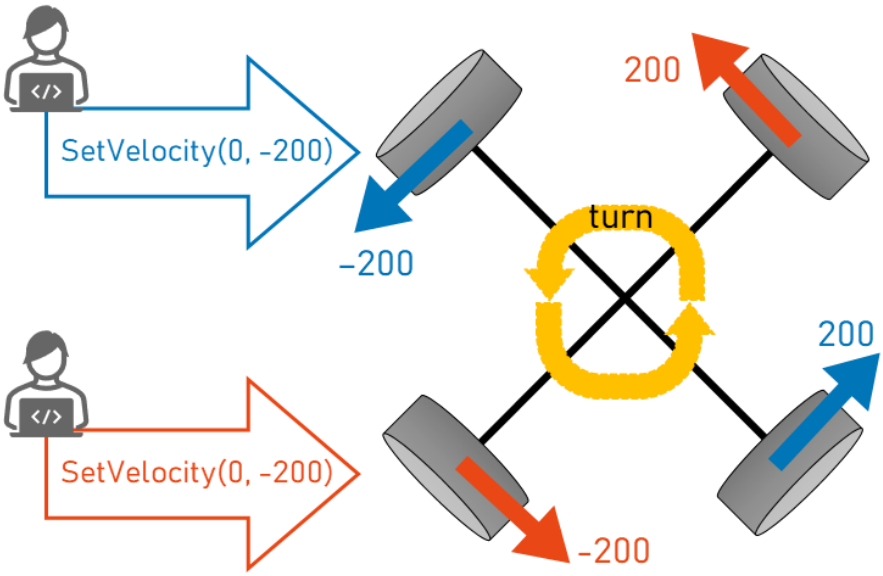

Figure 21 : Example of rotation (Counterclockwise)

9. Revision History

Revision	Date	Note
Rev 5	2024/02/07	Revert to Rev 3 due to physical configuration re-changes. (8) Add attention and example in Appendix. (8.1)
Rev 4	2024/01/26	Update Appendix due to physical configuration changes. (8)
Rev 3	2024/01/17	- Update figure 1 and add attention that Omni Platform has two communication interfaces. (2) - Extend limits of Control commands to enable equal movement forward, backward, left and right. (3.2.5, 3.2.6) - Add Appendix. (8)
Rev 2	2024/01/15	Delete descriptions of Model CR because the base of Omni Platform is same as that of Model CR2.
Rev 1	2024/01/11	Copy of “WHILL_Control_System_Protocol_Specification.docx Rev 8”