Urban NaviGator DBW Serial Interface Specification

This document describes all the serial message structure used to perform drive-by-wire operation on the Urban NaviGator using the Apollo interface. The DBW device messages are different when using the old Tablet code.

Contents

[1 Hardware Requirements 1](#_Toc516836874)

[2 Braking and Acceleration Control 1](#_Toc516836875)

[2.1 Serial Communication 2](#_Toc516836876)

[2.1.1 DBW Device to Tablet: 2](#_Toc516836877)

[2.1.2 Tablet to DBW Device 2](#_Toc516836878)

[3 Steering Angle Control 3](#_Toc516836879)

[3.1 Serial Communication 3](#_Toc516836880)

[4 Shifting Control 4](#_Toc516836881)

[4.1 Serial Communication 4](#_Toc516836882)

# Hardware Requirements

# Braking and Acceleration Control

You will need to connect to the NI myRIO to control the braking and acceleration effort of the vehicle. The myRIO code has a state structure:

## Serial Communication

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rx: | Pin 10 Connector A | Baud Rate: | 115200 | Data Bits: | 8 |
| Tx: | Pin 14 Connector A | Parity | None | Stop Bits: | 1 |

### DBW Device to Tablet:

The DBW board will send two messages to the connected device. The first message is a periodic update message sent every 100 milliseconds. This message is a brief description of the current state of the DBW device. It is sent along with a Cyclic Redundancy Check value so that the receiver can test the validity of the data within the message.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Header | State | Pause | Left Blinker | Right Blinker | Brake  % | Throttle % | CRC | Terminator |
| 157 | Ref. Table | Ref. Table | Ref. Table | Ref. Table | 0-100 | 0-100 | 0-255 | 147 |

The second message is a response message that will be sent whenever a message is received. The response variable will either be 128 for a correct message or 64 for a message with corrupted data. This should instruct the connected device whether to resend the message.

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| Header | Response | Terminator |
| 137 | Ref. Table | 127 |

### Tablet to DBW Device

This message is sent to the DBW device. The DBW device will send a response when it receives this message. It is up to the sender of this message on how to respond to the response from the DBW device. You should repeat this exchange until the DBW device acknowledges the message.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Header | Auto Req. | Pause  Cmd. | Left Blinker | Right Blinker | Brake  % | Throttle % | CRC | Terminator |
| 137 | Ref. Table | Ref. Table | Ref. Table | Ref. Table | 0-100 | 0-100 | 0-255 | 127 |

The table below shows the values that the message components will take given their state. All states were assigned a non-zero value so that the CRC computation is easier.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Value | State | Pause | Response | Auto Req. | Blinkers | Pause Cmd. |
| 32 | Inactive | - | - | - | - | - |
| 64 | Manual-Active | No pause in effect | Correct  Message | No auto request | No blinker | Pause not commanded |
| 128 | Full-Auto | Pause is in effect | Incorrect  Message | Auto request | Yes Blinker | Pause Commanded |
| 255 | E-stop | - | - | - | - | - |

# Steering Angle Control

Steering angle control is achieved through connecting to the SmartMotor that is attached to the steering column. You will have to go through a SEALevel communication converter that turns RS-232 to USB. This will require a driver: (link the driver software).

## Serial Communication

|  |  |  |  |
| --- | --- | --- | --- |
| Baud Rate: | 38400 | Data Bits: | 8 |
| Parity | None | Stop Bits: | 1 |

Commands to SmartMotor:

|  |  |
| --- | --- |
| Command | Description |
| RUN | Command to begin the SmartMotor program. Must be sent if SmartMotor resets. |
| p=# | Command to request a certain angle from the SmartMotor. The number is in units of encoder counts. The conversion must be known ahead of time to get accurate angles. 155,000 encoder count/max turn. |
| f=2 | Command to shutdown SmartMotor |

Message sent by SmartMotor:

|  |  |
| --- | --- |
| Command | Description |
| c=# | Sent periodically to report the current encoder count position. |
| HOMING | Sent after receiving the “RUN” command |
| HOMING\_COMPLETE | Sent after completing the homing process |
| READY | Sent after homing is complete and signals the program is ready for inputs. |
| NORMAL\_SHUTDOWN | Sent when shutdown command is received, f=2 |
| EMERGENCY\_SHUTDOWN | Sent when shutdown is required due to errors/limits |
| OVER\_CURRENT | Sent if motor exceeds current limit i.e. motor stalled. Motor is shutdown |
| THERMAL\_LIMIT | Sent if motor exceeds thermal limit. Motor is shutdown |

Messages from the SmartMotor will be sent with only a carriage return at the end of each message.

# Shifting Control

Shifting angle control is achieved through connecting to the SmartMotor that is attached to the shifting mechanism. You will have to go through a SEALevel communication converter that turns RS-232 to USB. This will require a driver: (link the driver software).

## Serial Communication

|  |  |  |  |
| --- | --- | --- | --- |
| Baud Rate: | 38400 | Data Bits: | 8 |
| Parity | None | Stop Bits: | 1 |

Commands to SmartMotor:

|  |  |
| --- | --- |
| Command | Description |
| RUN | Command to begin the SmartMotor program. Must be sent if SmartMotor resets. |
| h=1 | One of the two requirements for the program to start. The other being the brake pressed. |
| s=0 | Command to request park gear |
| s=255 | Command to request reverse gear |
| s=128 | Command to request neutral gear |
| s=1 | Command to request drive gear |
| s=2 | Command to request regen gear |

Message sent by SmartMotor:

|  |  |
| --- | --- |
| Command | Description |
| BREAK\_PEDAL\_HIGH | Sent when the brake pedal is not pressed |
| BREAK\_PEDAL\_LOW | Sent when the brake pedal is pressed |
| HOMING | Sent after receiving the “RUN” command |
| HOMING\_COMPLETE | Sent after completing the homing process |
| NORMAL\_SHUTDOWN | Sent when shutdown command is received, f=2 |
| READY | Sent after homing is complete and signals the program is ready for inputs. |
| FAULT High: Pot reading is too high. | Error message when encoder value is not correct for a desired gear |
| FAULT Low: Pot reading is too low. | Error message when encoder value is not correct for a desired gear |
| PARK | Sent after completing the transition to park gear |
| REVERSE | Sent after completing the transition to reverse gear |
| NEUTRAL | Sent after completing the transition to neutral gear |
| DRIVE | Sent after completing the transition to drive gear |
| REGEN | Sent after completing the transition to regen gear |
| ACTUATING\_TO\_PARK | Sent after request for park is received and is actuating to park |
| ACTUATING\_TO\_REVERSE | Sent after request for reverse is received and is actuating to reverse |
| ACTUATING\_TO\_NEUTRAL | Sent after request for neutral is received and is actuating to neutral |
| ACTUATING\_TO\_DRIVE | Sent after request for drive is received and is actuating to drive |
| ACTUATING\_REGEN | Sent after request for regen is received and is actuating to regen |
| UNKNOWN\_COMMAND | Sent if request is not one of the specified values |

Messages from the SmartMotor will be sent with only a carriage return at the end of each message.