

# 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.

## 1 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:

(Picture of myRIO states and transitions)

## 1.1 SERIAL COMMUNICATION PARAMETERS:

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

## 1.2 DBW BOARD TO TABLET:

Refer to the **Error! Reference source not found.** section for more details about the signals. The analog readings have been splits into two bytes because the converters are 12bit. Brake switch is the physical switch, Brake Lights is the current state of the brake lights. Current braking/throttle effort is send to the Tablet.

0	1	2	3
Header			
65	128	254	68

4	5	6	7		
E-Stop	Auto	Brake Switch	Brake Lights	Brake Effort	Throttle Effort
170 = Enable	200 = Enable	0 or 1	0 or 1	0-100	0-100

10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
DAC1		DAC2		DAC3		DAC4		DAC5		DAC6		DAC7		DAC8	
Low	High	Low	High	Low	High	Low	High	Low	High	Low	High	Low	High	Low	High

26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41
ADC1		ADC2		ADC3		ADC4		ADC5		ADC6		ADC7		ADC8	
Low	High	Low	High	Low	High	Low	High	Low	High	Low	High	Low	High	Low	High

42
Terminator
78

DAC/ADC Channel	Connection
1	Brake Stroke Sensor 1
2	Brake Stroke Sensor 2
3	Brake Pressure Sensor 1
4	Brake Pressure Sensor 2
5	Accelerator Stroke Sensor 1
6	Accelerator Stroke Sensor 2
7	N/A
8	N/A

### 1.3 TABLET TO DBW BOARD

0	1	2	3	4	5	6	7	8	9
Header				Auto	Brake	Throttle	Left Blinker	Right Blinker	Pause Command
55	93	200	30	128 = Enable	0-100	0-100	128 = Enable	128 = Enable	128 = enable
10	11	12	13	14	15	16	17	18	19
Termination Bytes									
77	77	77	77	77	77	77	77	77	88

## 2 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).

The SmartMotor also has a state machine. The states are initialize, ready. Describe the transition process.

### 2.1 STATE MACHINE

### 2.2 SERIAL COMMUNICATION

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

Commands to send 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. INCLUDE CONVERSION HERE
f=2	Command to shutdown SmartMotor

Commands 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 a carriage return at the end of each message.

## 3 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](#)).

### 3.1 STATE MACHINE

The SmartMotor also has a state machine. The states are initialize, ready. Describe the transition process.

### 3.2 SERIAL COMMUNICATION

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

Commands to send 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

Commands 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	
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 a carriage return at the end of each message.