Navigation Laboratories Inc.
11 Pine Valley
Coto De Caza California, 92679
(949) 766-0444 / (949) 495-9551
Fax - (949) 713 - 2479

GPS Constellation Simulators

Support Tools

&

Navigation Products

Automotive package description (last Update FRB 11-03)

Product Overview

This document presents the modeling basis and operational description for the Tapestry **Automotive Dead Reckoning** simulation package. This package includes hardware and software upgrades to the basic Tapestry LabPro Constellation Simulator.

1.0 Overview of the Auxiliary Automotive Sensor.

To support multi-sensor GPS automotive products, the Tapestry system can be configured to generate voltages proportional to vehicle turn rate, acceleration, and ground speed. The simulator also provides serial data outputs at 1-10Hz that emulate proprietary ***D**ead* ***R**eckoning* *serial* *data* outputs in various formats.

Some aspects of the Tapestry implementation are:

- The auxiliary sensors - those sensors simulated in addition to the GPS RF signal - are precisely coordinated with the simulated dynamics of the vehicle. This is because the vehicle motion truth data and dynamics are used at a high rate to generate the auxiliary sensor outputs.
- The timing relationship between auxiliary sensors and the generated RF signal is controlled.
- The auxiliary sensor data is generated using our proprietary Multi-Function I/O expansion (MFIO) card. The card communicates with the Tapestry system using a custom Windows device driver we developed specifically for this application.
- The simulated sensor data is saved in formatted files that can be used post-test for data analysis and characterization.
- The system is turnkey with no user or third party software/hardware required.

The Automotive package includes the following features:

- 5 Volt analog output proportional to vehicle **Rate-of-Turn**. Proportionality factor and null rate offset are programmable. Nominally configured at 22.2 millivolts/ °/sec with a 2.5 Volt offset. The sense of the positive rate is programmable. Error models simulating noise, bias, scale factor, and non-linearity are provided.

- CMOS Level Pulse rate output proportional to **Vehicle Speed**. Characteristics are: 50% duty cycle square wave programmable from 0 to 80,000 transitions per second. Error models simulating wheel slip, wheel sliding, bias, scale factor, and noise are provided. 4 Independent wheel channels provided with

• NavLabs **, 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444** •

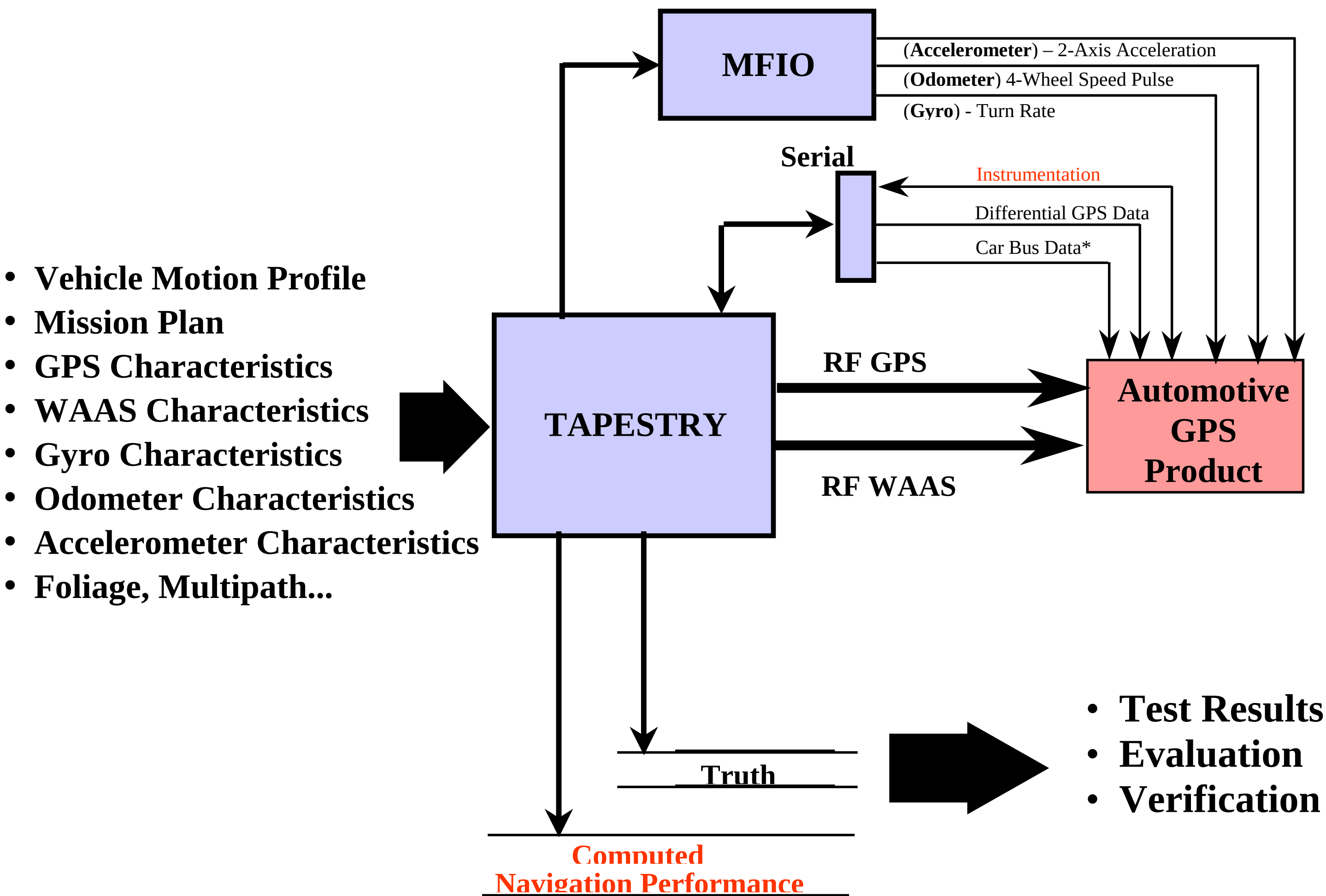
programmable wheel separation model.

- Serial Data Messages. Wheel Speed, Turn Rate, and other navigation data provided in serial format via PC Comport. 1 to 10 Hz rate supported in various formats.

- Analog Accelerometer outputs. 2 channels of 0-5 Volt accelerometer outputs are provided. Proportionality factor and null rate offset are programmable. Nominally configured at 600 millivolts/G with a 2.0 Volt offset. Error models simulating noise, bias, scale factor, and non-linearity are provided. Forward (X) and Lateral (Y) channels provided.

- User programmable wheel separation model provided. This support differential wheel ticks for inferring turn rate.

ILLUSTRATION OF AUTOMOTIVE SIMULATION PACKAGE



• NavLabs , 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444 •

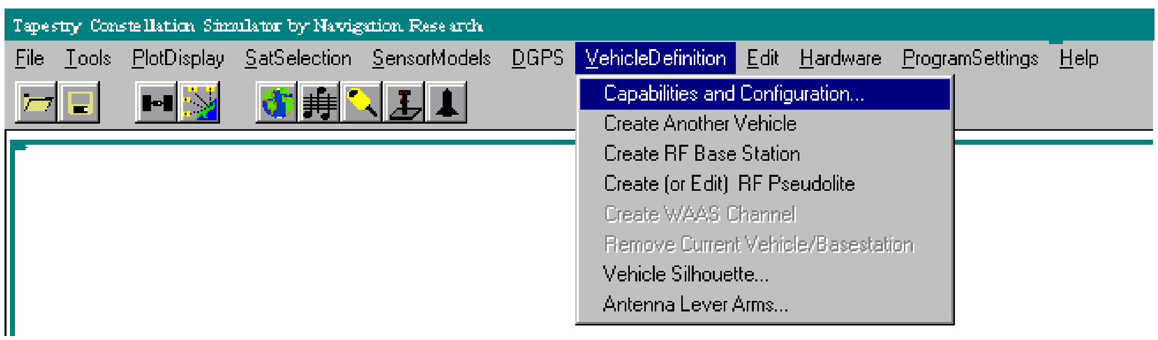
2.0 Using the Simulated Dead Reckoning Automotive Sensors

It will be assumed in this section that a working familiarization with the Tapestry Windows graphical user interface has been achieved. If you are not familiar with the basic operational characteristics of the Tapestry software please review the *Tapestry Users Guide(s)* provided with your simulator.

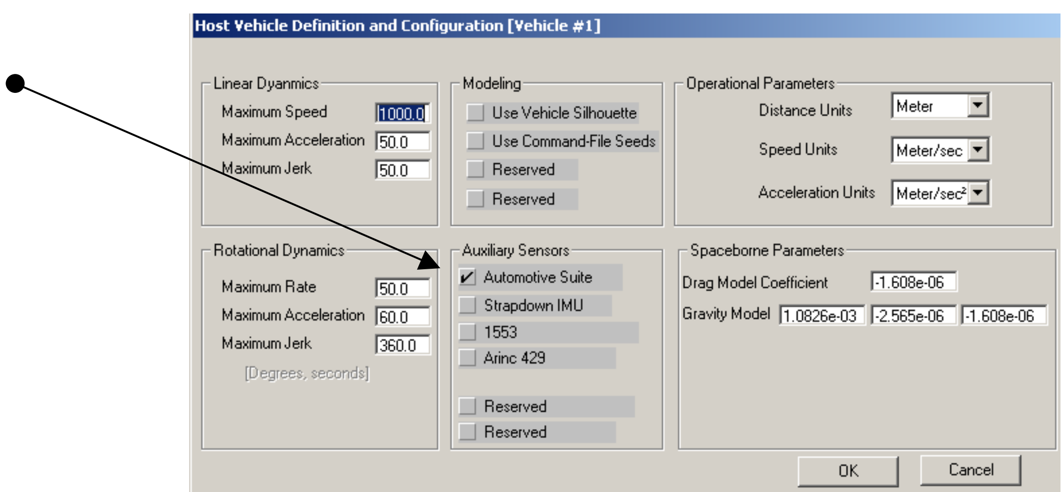
2.1. Adding the Sensors to your Simulated Vehicle

If the Automotive Sensors are not specifically added to scenario you are developing no sensor outputs will be generated. Add the automotive sensors to your simulation as follows:

From the Tapestry main display pull down the ***VehicleDefinition*** menu and select ***Capabilities and Configuration…***



From the PoP-Up, check the **Automotive Suite** check box as shown.

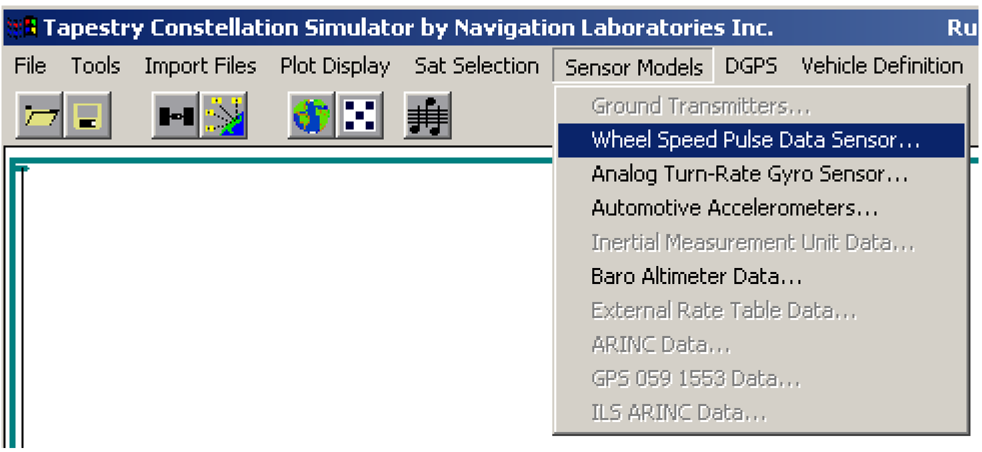


• NavLabs , 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444 •

Note: ***If you want your simulation to include these sensors by default, then select the "DEFAULT" scenario identifier and check the above boxes. Once this has been done all new simulations will include the automotive sensor outputs by default (when delivered your system is setup this way). You can also configure the DEFAULT serial port assignment and sensor configuration you desire by setting them in the DEFAULT scenario as well. Be careful changing the DEFAULT scenario- if an error is made you could corrupt all subsequent scenarios!***

2.2. Setting up the Sensor Models

Now that you have attached the sensors to your vehicle, you can change the model settings. From the main menu select the ***SensorModels*** pull down.
.



Select the sensor model you wish to configure and the appropriate data form will be created.

2.2.1. Wheel Speed Sensor Setup.

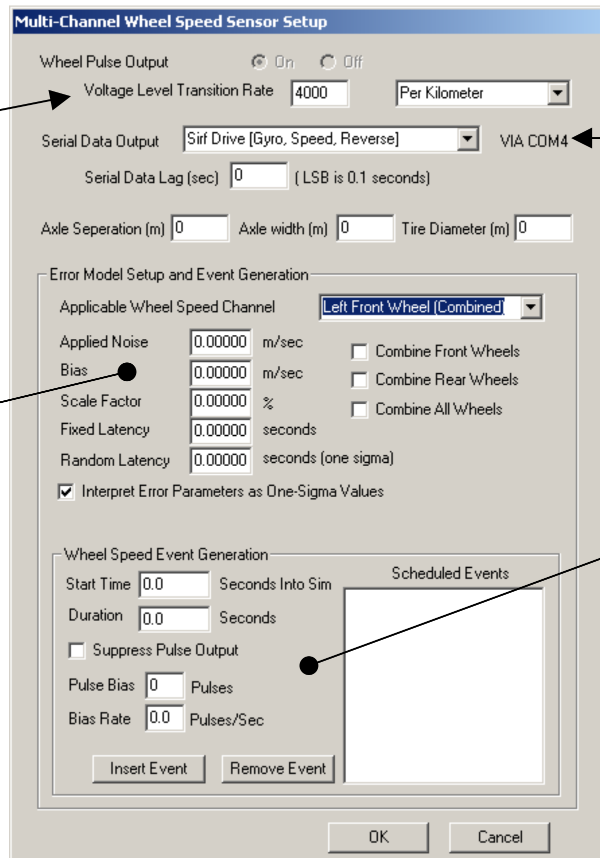
Selecting Wheel Speed Pulse Data Sensor displays the following Pop-Up data form. From the data form you may change the default characteristics of the sensor, apply error effects, or setup scheduled “events” which cause anomalies in the output data.

In addition, this menu provides a gateway for the configuration of the serial data stream COM port assignment. **All serialized Wheel Speed messages are configured to be output via COM4 on the computer back panel. COM4 is the TOP (vertical) DB9 connector on the installed auxiliary COM port card.**

Serial data is assigned to COM4. SirfDrive[R] and other formats supported.

Wheel Tick Outputs are @ CMOS levels. Four Wheel Independent settings

Enter the 3.8 Volt to 0 voltage level transition rate



Use this menu to configure each of 4 wheels

Use this menu to generate Wheel Speed anomalies

Wheel Pulse Output

Our Multi-Function Input Output (MFIO) expansion card installed with the automotive package provides a 50% duty cycle square wave pulse train for each of the wheels on the vehicle. The pulse train is from at CMOS level. The pulse output provides a direct measure of distance traveled for the vehicle. The output is automatic with the feature card pin-out provided in the appendix to this document.
For completeness the pin-out is:

MFIO DB25 feature card Wheel Tick Outputs

FUNCTION	PIN OUT	COMMENTS
Wheel Tick – 1	18	Left Front Wheel
Wheel Tick – 2	6	Right Front Wheel
Wheel Tick – 3	19	Left Rear Wheel
Wheel Tick – 4	7	Right Rear Wheel
Ground	11	

Voltage Level Transition Rate

You must enter a non-zero value into this field. It is used to compute the appropriate number of wheel pulse counts to output over a given output interval. You may configure the pulse count in either Pulse/Mile or Pulse/Kilometer.

Axle Width (meters)

If a non-zero value (in meters) is entered for this field, wheel separation effects are applied to the output data pulse stream and serial data. This entry controls the **width** of the car from the left to the right hand side.

Axle Separation (meters)

This entry controls the **length** of the car from the back to the front.

Tire Diameter (meters)

This entry controls the **width** of the tire. It is used primarily to support differential wheel ticks and serial output data.

Serial Data Output.

By default, this item is set to NO OUTPUT. If you do not change the setting you will not have serial sensor data outputs for the simulation. All composite serial data is output to COM4. COM4 is the TOP DB9 on the back of the LabPro computer chassis.

The following choices are available.

Zodiac 1370 Serial Wheel Speed Format.

This format is the required serial input data format for the Rockwell (Conexant) Zodiac 12 channel Dead Reckoning receiver. The data consists of time strobed distance traveled and related sensor status information and T20 internal Rockwell

• NavLabs , 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444 •

clock timing. This is a proprietary format; consult the Rockwell Zodiac Developers kit documentation for the precise binary structure and data layout. The distance-traveled content of the message is described in section 2.2.2 of this addendum.

Car Bus Wheel Tick format.

This format generates 10 Hz serial data consisting of the following data structure;

```
unsigned char Start Of Header = 0xff
unsigned char Synch           = 0x81
unsigned char MessageId       = 0x10
unsigned char Number Data Bytes to Follow
struct {
        unsigned short int LeftFrontWheelTicks over last 100 mseconds
        unsigned short int RightFrontWheelTicks over last 100 mseconds
        unsigned short int LeftRearWheelTicks over last 100 mseconds
        unsigned short int RightRearWheelTicks over last 100 mseconds
        unsigned char Direction (0 = forward, 1 Reverse)
}
unsigned char Checksum ( byte-wise add of above structure,  then negate)
```

The data LSB is 1. In principal the data content of each wheel is different due to error characteristics that are independent for each wheel. If the wheels are locked (combined) then the error models will be locked in the same configuration you have defined.

See section 2.2.2 for a description of the modeling for this data.

Car Bus Speed format.

This format generates 10 Hz serial data consisting of the following data structure;

```
unsigned char Start Of Header = 0xff
unsigned char Synch           = 0x81
unsigned char MessageId       = 0x20
unsigned char Number Data Bytes to Follow
struct {
        unsigned short int Speed;  // LSB 0.01 m/s
        unsigned char  Direction;
}
unsigned char Checksum ( byte-wise add of above structure,  then negate)
```

The data LSB is 0.01 m/second. See section 2.2.2 for a description of the modeling for this data.

• NavLabs , 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444 •

Car Bus Odometer format.

This format generates 10 Hz serial data consisting of the following data structure;

```
unsigned char Start Of Header = 0xff
unsigned char Synch           = 0x81
unsigned char MessageId       = 0x30
unsigned char Number Data Bytes to Follow
struct {
        unsigned short int TraveledDistance;   // LSB 0.01 meters
        unsigned char  Direction;
}
unsigned char Checksum ( byte-wise add of above structure,  then negate)
```

The data LSB is 0.01 meter. See section 2.2.2 for a description of the modeling for this data.

SirfDrive (Gyro, Speed, Reverse)

This format generates 10 Hz accumulated data output at 1 Hz via COM4. The serial protocol is 38400,N, 8,1. This is a proprietary format. Please consult your Sirf developers' kit documentation for the details.

SirfDrive (Wheel Pulses, Reverse)

This format generates 10 Hz accumulated data output at 1 Hz via COM4. The serial protocol is 38400,N, 8,1. This is a proprietary format. Please consult your Sirf developer's kit documentation for the details.

SirfDrive (Wheel Speed, Reverse)

This format generates 10 Hz accumulated data output at 1 Hz via COM4. The serial protocol is 38400,N, 8,1. This is a proprietary format. Please consult your Sirf developer's kit documentation for the details.

SirfDrive (Wheel Angular Speed, Reverse)

This format generates 10 Hz accumulated data output at 1 Hz via COM4. The serial protocol is 38400,N, 8,1. This is a proprietary format. Please consult your Sirf developer's kit documentation for the details.

Serial Data Lag (LSB is 0.1 seconds)

Nominally serial data is output on-time, that is coincident with the GPS based timing mark. You may arbitrarily advance or delay output of the serial data with a granularity of 100 milliseconds. To advance serial data enter a (-) value into this field. This will cause the output data to be in the *future* by the entered amount of time (in 100 millisecond increments). A delay (lag) is programmed by entering a (+) time into this field. This will cause the data to be in the past by the entered time increment.

Wheel Speed Error Model

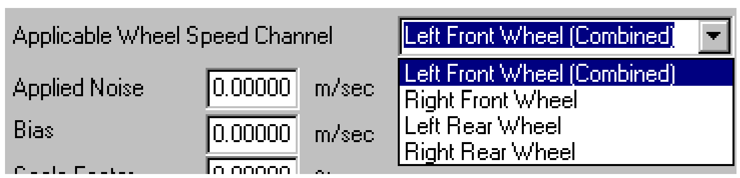
If these items are not changed the simulation will generate an exact value data stream. To corrupt the output data stream you may enter values into this menu. In addition, you may schedule specific wheel events to occur at a specified time and last for a specified duration.

Interpret Error Parameters as One Sigma Value.

If this item is checked, all of the following values are drawn randomly. To ascertain the values drawn see the description of the Monte Carlo capabilities of the simulator in the main section of the ***Build Scenario Users Guide***.

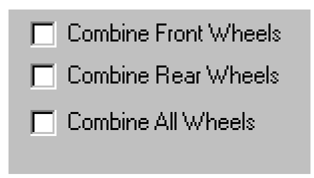
Applicable Wheel Channel.

The Tapestry models each wheel independently vis-à-vis the error model and special event schedule.



Use the pull down menu to select the applicable wheel for the error model parameters.

If you want to combine wheels into a composite signal, use the checkbox fields to do so. If you combine wheels, Tapestry will reset the error parameters you have specified such that the deterministic and random terms are forced equal. The priority of the re-assignment is channel 1 ⇒ channel 2 ⇒ channel 3 ⇒ channel 4.



Bias

The Bias applies a fixed offset to the output data. Note that the value applied is exactly the value you enter unless you have checked *Interpret Error Model…* check box described previously. It is entered in m/s.

Scale Factor

The Scale Factor applies an offset to the output data under load. Note that the value applied is exactly the value you enter unless you have checked *Interpret Error Model…* check box described previously. It is entered in %.

Fixed Latency

The Tapestry outputs serial data at 10Hz. This output rate is unwavering. To simulate the effect of data latency the output value is adjusted in magnitude based upon a presumed timing error. For example, if Wheel Speed were the selected output, Latency would have an effect of changing the output speed only under applied acceleration. The output Wheel Speed data is computed as;

```
Output_wheel_speed  = true_wheel_speed
                    + errors
                    - fixed_latency*acceleration_along_track
```

If a Distance Traveled data type were the output, the output data is,

```
Output_distance_increment   = true_distance_increment
                            + errors
                            - fixed_latency*speed
```

Random Latency

Tapestry outputs serial data at 10Hz. This output rate is unwavering. To simulate random latency the input value, interpreted as a one-sigma entry, is used to scale a gaussian random number generator. Output data is modified to simulate random latency effects as;

```
Output_wheel_speed  = true_wheel_speed
                    + errors
                    - fixed_latency*acceleration_along_track
                    + random_latency*acceleration_along_track*gauss()
```

or

```
Output_distance_increment   = true_distance_increment
                            + errors
                            - fixed_latency*speed
                            + random_latency*speed*gauss()
```

Wheel Speed Event Generation

You may schedule wheel events consisting of slipping, sliding, bias, and ramp. To schedule and event, enter the desired event start time (in seconds into week) and the duration in seconds. Fill out the data form and press ***Insert Event***. The event will be inserted into the schedule window. This window reflects the contents of the ASCII file wslip1/2/3/4.scn. (1,2,3,4 are for each wheel).

Suppress Pulse Output

This box is checked if you want NO pulse signal output over the defined interval. During this interval there will be 0 pulse count sent for serial data output. This simulates sliding.

Pulse Bias

This adds an instantaneous bias to the data output. This simulates effects of wheel slip. It is most useful if used in conjunction with *Bias Rate*.

Bias Rate

This value increases/decreases the instantaneous pulse bias over the defined data interval. This simulates effects of accelerating wheel slip.

2.2.2 Wheel Speed Model and Computation Method.

For all wheel speed models the following algorithm is used to compute the output data content for each channel.

True vehicle ground speed ($\mathbf{S_T}$), derived from continuous vehicle linear jerk, acceleration, and velocity, is generated at 10Hz within the main simulation program element. The true ground speed is deterministically corrupted using the following error model;

$$\mathbf{S} = (1 - \alpha)\, \mathbf{S}_T + \beta + \eta * G(seed) + (\tau + \tau_R * G(seed))\,(A_T)$$

i. Bias (β). This is entered in m/second and applied directly to the true ground speed.
ii. White noise(η). Entered in m/s and used directly as a multiplier for a 10Hz gaussian random noise process.
iii. Scale factor (α). Entered in % and used to decrease/increase the ground speed reading proportionally.
iv. Latency (τ,τ_R). τ is a fixed delay which scales true vehicle forward acceleration A_T to provide a simulated speed error. τ_R scales a seed driven gaussian noise process G (seed) to apply variability (jitter) to the time delay. This delay also multiplies true forward vehicle acceleration.

The wheel speed, in either analog or serial format, is truncated per the pulse rate entered by the user. This provides the quantization effect observed in an actual vehicle.

• NavLabs , 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444 •

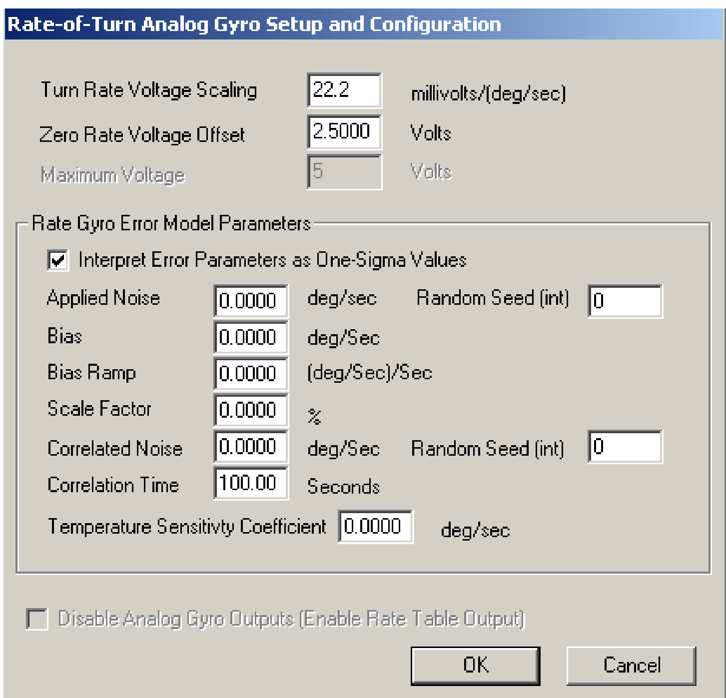
If wheels are combined, the random and deterministic errors are equated for the combined channels.

In practice, due to truncation, the wheel speed, distance traveled or pulse count is rounded down. To account for this effect, the output data type is computed in double precision and then cast to integer. The truncated value is reapplied on the next measurement cycle.

Effects from wheel separation are accounted for by computing the above measurement at the centerline of the modeled vehicle. Using the input wheel separation, effects can be applied on inner/outer wheels by using the instantaneous vehicle turn rate.

2.2.3 Analog Turn Rate Gyro Sensor.

The gyro model is used to generate an analog voltage output in addition to pumping the appropriate serial data messages.



Turn Rate Voltage Scaling

This value controls the transformation from ^{o}/Sec of turn rate to voltage output on the D2A expansion card. By default the setting is 22.2 milli-volts/ Degree/second. A positive voltage above the zero-point offset corresponds to a clock-wise rotation. To switch the sense, see using the A2DCONFG file in section 3.

• NavLabs , 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444 •

Zero Rate Voltage Output

For a typical turn rate gyro, 0 angular rate does not correspond to 0 output voltage. Typically the sensor outputs the maximum (counter-clockwise) turn rate at 0 volts and the maximum (clockwise) turn rate at 5.0 volts. The default setting is 2.5 volts at 0 ^{o}/second. To change the calibration of the 0-level see using the A2DCONFG file in section 3.

Rate Gyro Error Parameters

The error model for the gyro is controlled by the values input in the following data items.

Applied Noise

This value is interpreted as a one-sigma standard deviation used to scale a gaussian random number sequence. White noise is applied to the vehicle turn rate.

Output Rate = True Rate + Applied_Noise*gauss()

The value is input in ^{o}/Sec and computed at 10Hz.

Bias

This input provides a fixed offset to the output gyro turn rate. Note that this value is not interpreted as a standard deviation but is used directly as the applied rate bias.

Output Rate = True Rate +Bias

The value is input in ^{o}/Sec.

Bias Ramp

This value ramps the gyro drift over time. As is the case for the gyro bias, this value is directly applied to the gyro drift and is not interpreted as a standard deviation.

Output Rate = True Rate + (Bias + Bias_Ramp*ΔT)

The value is input in ^{o}/Sec/Sec.

Scale Factor

• NavLabs , 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444 •

This value is used to apply an offset to the output turn rate based upon the magnitude of the applied vehicle turn rate. This value is applied directly and not used as a standard deviation.

Output Rate = True Rate*(1.0 + Scale_Factor)

The value is input in %.

Correlated Noise and Time

These parameters control a first order Gauss-Markov process. The error model is of the form

$$\partial R/\partial t = - R/\tau + \sigma W$$

Where W is a white noise process scaled and computed at 10Hz.

2.2.4 Gyro Error Model and Computation Method

The following algorithm is used to compute the output data content for the turn rate analog channel.

True turn rate ($\mathbf{R}_T$), derived from continuous vehicle linear and angular jerk, acceleration, and velocity, is generated at 10Hz within the main simulation program element. The true turn rate can be deterministically corrupted using the following error model;

$$\mathbf{R} = (1 - \alpha)\, \mathbf{R}_T + \beta + \eta * G(seed) + \eta_C$$

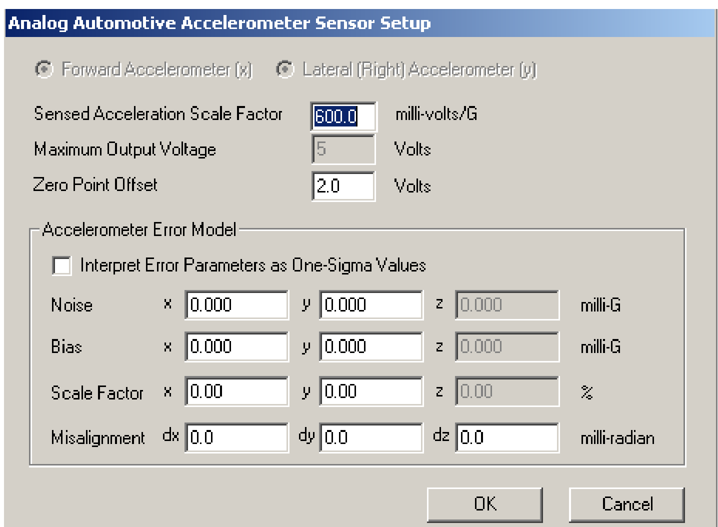
i. Gyro Bias (β). This is entered in degrees/second and applied directly to the true turn rate.
ii. White noise(η). Entered in °/s and used directly as a multiplier for a 10Hz gaussian random noise process G(seed).
iii. Scale factor (α). Entered in % and used to decrease/increase the turn rate proportionally.
iv. Correlated noise (η_C). Scales a 1^{st} order gauss Markov process. This error component can be used to simulated effects from residual temperature calibration error or to effectively provide a slowly time-varying gyro bias component.

These error models, were appropriate, are driven with a random seed. If the seed is initialized to zero the PC clock seeds the gaussian random number generator and each *build* (not execution - which is repeated exactly each time!) of the simulation results in another error ensemble. If non-zero, the seed causes the error ensemble to be deterministic.

FUNCTION	PIN OUT	COMMENTS
DAC –Chan 0	20	Reverse indicator
DAC –Chan 1	21	Gyro turn rate
DAC –Chan 2	9	Forward Accelerometer
DAC –Chan 3	22	Lateral Accelerometer
Ground	8	

2.2.5 Accelerometer Sensor Model Setup and Configuration

Tapestry incorporates two (2) analog accelerometers. The user entries are defined below with the output characteristics established by the user.



By construction the accelerometers are aligned one (1) sensing forward acceleration and the other sensing lateral acceleration.

Sensed Acceleration Scale Factor

This value scales the sensed acceleration into a voltage output. Typically a positive (forward) acceleration results in a positive voltage above the zero-point offset. A negative (reverse) acceleration generates a decreasing voltage relative to the zero-point offset.

• NavLabs , 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444 •

A lateral acceleration to the right produces a voltage above the zero-point offset. Acceleration to the left produces a decreasing voltage relative to the zero-point offset.

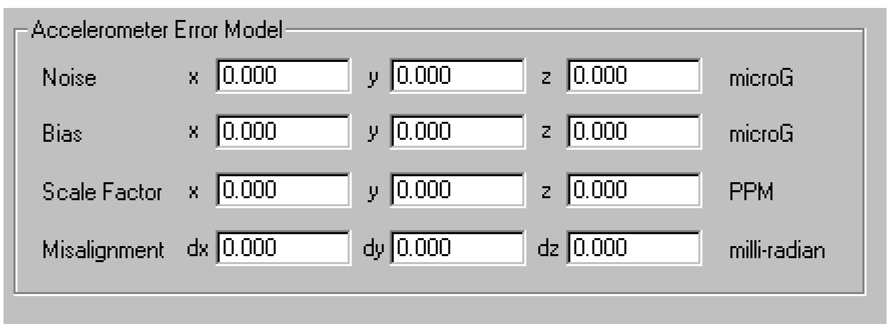
Zero Point Offset

To sense both directions of acceleration a zero point offset is required. Typically 2-3 volts corresponds to no sensed acceleration.

Accelerometer Error Model

The model simulates a two accelerometers that measure inertial reference vehicle acceleration and express the result in the vehicle sensor frame. Any acceleration dynamic range can be simulated with the LSB derived from the 12-bit mechanization of the function card. Typically ± 2G's is the maximum acceleration.

The following data form allows the user to control the applied errors for these sensors,.


Accelerometer Error Model

Noise	x	0.000	y	0.000	z	0.000	microG
Bias	x	0.000	y	0.000	z	0.000	microG
Scale Factor	x	0.000	y	0.000	z	0.000	PPM
Misalignment	dx	0.000	dy	0.000	dz	0.000	milli-radian

The error model equations for the output of the vehicle incremental velocity are;

$$\Delta\upsilon^M = \Delta\upsilon^T + \beta + (1 + \alpha)\Delta\upsilon^T + \varphi \times \Delta\upsilon^T + \sigma W$$

Where $\Delta\upsilon^T$ is the TRUE delta velocity based upon the simulated vehicle truth data, $\Delta\upsilon^M$ is the IRU-measured delta velocity output via the IRU expansion card. The remaining terms are defined as follows:

Noise (σ W)

σ is the standard deviation of a gaussian white noise process (W) used to apply jitter to the output delta velocity. This error is typically due to sensor measurement electronics. These errors are applied per each case axis (x-forward, y-right, z-down) in units of micro G (µg). A micro-G is given by

$$1\mu g = 1.0 \times 10^{-6} G \approx 0.00001 \text{ m/s}^2$$

Bias (β)

• NavLabs , 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444 •

This value corresponds to the non-zero accelerometer measurement output for a zero applied acceleration input. The units are μg's with 1 μg ≈ 0.00001 m/s^2

Scale Factor (α)

A non zero scale factor causes the output delta velocity to be in error proportional to the true acceleration input along the defined axis. The scale factor defines the proportionality constant. The input units are parts-per-million (PPM). 1 PPM = 1.0×10^{-6}

Misalignment (φ)

The accelerometers are mounted perfectly in an orthogonal configuration along the IRU case sensitive axis. However there is always some small residual error in the alignment. This error causes a sensed acceleration to be projected into one of the cross orthogonal case axis. This item simulates this effect with the input units being milli-radian or 0.001 radians.

Least Significant Bit effects (LSB).

This is a very *important* component of the error model. This effect results from scaling of measured velocity increment into a data length compatible with the data interface specification. There are no user entered parameters as the LSB error is fixed by the interface specification and dynamic range specified by the user.

FUNCTION	PIN OUT	COMMENTS
DAC –Chan 0	20	Reverse indicator
DAC –Chan 1	21	Gyro turn rate
DAC –Chan 2	9	Forward Accelerometer
DAC –Chan 3	22	Lateral Accelerometer
Ground	8	

3.0 Configuration of the Dead Reckoning Upgrade Package

The Automotive package is an optional add-on to the Tapestry Windows software delivered with the Navigation Laboratories LabPro constellation simulators. The automotive package is a turn-key system including all of the following elements:

- Tapestry Windows Automotive Software package. The software replaces the scenario building and run-time applications described in the Tapestry Windows Users Manual. This software includes a 1 year support and upgrade privileges

- MFIO PCI function generation card. This expansion card and software device driver is used to generate 4 variable analog channels and 4 discrete pulse channels (wheel speed pulse train). The output connector is a DB25.

- Comprehensive instruction manual and optional training.

Using the A2DCONFG file

There is a configuration file associated with the automotive package that can be used to:

- Input calibration values for the analog output channels.

- Change the sense of the 5 Volt reverse line, the turn direction of the gyro and the sense of the accelerometers.

Details of the A2DCONFG.SCN file.

The A2DCONFG.SCN is and ASCII file located in c:\voyager\runs\default. Note that this file is not in the scenario folder as the other control files for your simulation. It **MUST** be found in the default scenario only and will apply to all new scenarios' you build. If you change the settings in the configuration file you will have to rebuild the scenario's you've previously constructed for the effects to apply.

The format of the file looks like this:

```
; Multi-function I/O card configuration file - End user should edit this
; card only with great care.  The data is as follows
; NOANALOG = 0, GYRO = 1, REVERSEA = 2, FACCELEROMETER = 3, VACCELEROMETER = 4
; TEMPERATURE = 5, RATETABLE = 6
; Channel MinVolt MaxVolt HWScale OffsetCal AssignedSensor (see above scheme), switch
; switch - 0 is only used for the gyro and the reverse signal - it allows the user
; to switch the positive sense of the voltage 0 is default ( clockwise turn negative rate
; 5 volts when moving forward, 1 sets clockwise turn positive, and 5 volts in reverse
0 0.0 5.0 4096 0.0 2 0       ; reverse
1 0.0 5.0 4096 0.0005 1 0  ; gyro
2 0.0 5.0 4096 0.0 3 0       ; forward accel
3 0.0 5.0 4096 0.0 4 0       ; lateral accel
```

Note the comment lines that begin with a semi-colon (;). You <u>cannot</u> add comment lines of your own after the data entries have started. Basically it is not a good idea to add any lines to the file. Lets consider the entry data lines.

This floating-point value is for calibration. The zero point offset may not be at the precise correct value programmed. The value in this filed is ADDED to the sensor identified in the comment line

```
0 0.0 5.0 4096 0.0 2 0       ; reverse
1 0.0 5.0 4096 0.0005 1 0  ; gyro
2 0.0 5.0 4096 0.0 3 0       ; forward accel
3 0.0 5.0 4096 0.0 4 0       ; lateral accel
```

This integer value controls the sense of the output. 0/1 reverses the sense of either the acceleration or gyro outputs.

<u>Digital to Analog Expansion card Pin Out.</u>

DB25 Male

FUNCTION	PIN OUT	COMMENTS
DAC –Chan 0	20	Reverse indicator
DAC –Chan 1	21	Gyro turn rate
DAC –Chan 2	9	Accelerometer (linear)
DAC –Chan 3	22	Accelerometer (Vertical)
Ground	8	All grounds tied together
Wheel Tick – 1	18	Left Front Wheel
Wheel Tick – 2	6	Right Front Wheel
Wheel Tick – 3	19	Left Rear Wheel
Wheel Tick – 4	7	Right Rear Wheel
Ground	11	All grounds tied together
Discrete chan 0	1	Programmable Output discrete
Discrete chan 1	14	Programmable Output discrete
Discrete chan 2	2	Programmable Output discrete
Discrete chan 3	15	Programmable Output discrete
Discrete chan 4	3	Programmable Output discrete
Discrete chan 5	16	Programmable Output discrete
Discrete chan 0	4	Programmable Input discrete (unused)
Discrete chan 1	17	Programmable Input discrete (unused)

• NavLabs , 11 Pine Valley, Coto De Caza Ca. 92679 (949) 766 -0444 •