Mechanical Simulation

VehicleSim Products

755 Phoenix Drive, Ann Arbor MI, 48108, USA
Phone: 734 668-2930 • Fax: 734 668-2877 • Email: info@carsim.com

carsim.com

Change Units of VS Math Model Variables

Viewing Existing Units in a VS Math Model 1
Using Alternative Units for a Parameter 3
Change the Units for Output Variables 4
Define New Units..6

The VS Math Models in CarSim, TruckSim, and BikeSim maintain two sets of units for all parameters and variables.

1. All calculations during a simulation are based on all parameters and variables using the *International System of Units* (SI). This simplifies the equations because no scale factors are needed in the internal equations of motion. Length and distance variables use meters, mass variables use kilograms, force variables use Newtons, angular variables use radians, etc.

2. All machine-generated documentation files (Echo files, output motion files, etc.) show values of variables and parameters using *user display units*, which may be more convenient: mm for small motions, degrees for angles, g's for acceleration, etc. All model input parameters are provided with user display units.

VS Math Models allow the user display units to be modified. Also, new user display units can be created. The main references for how units are managed are the *VS Math Model Reference Manual* and the *VS Commands Manual*. This memo expands on the reference material to show examples for providing parameter values with alternative units and generating plots using alternative units.

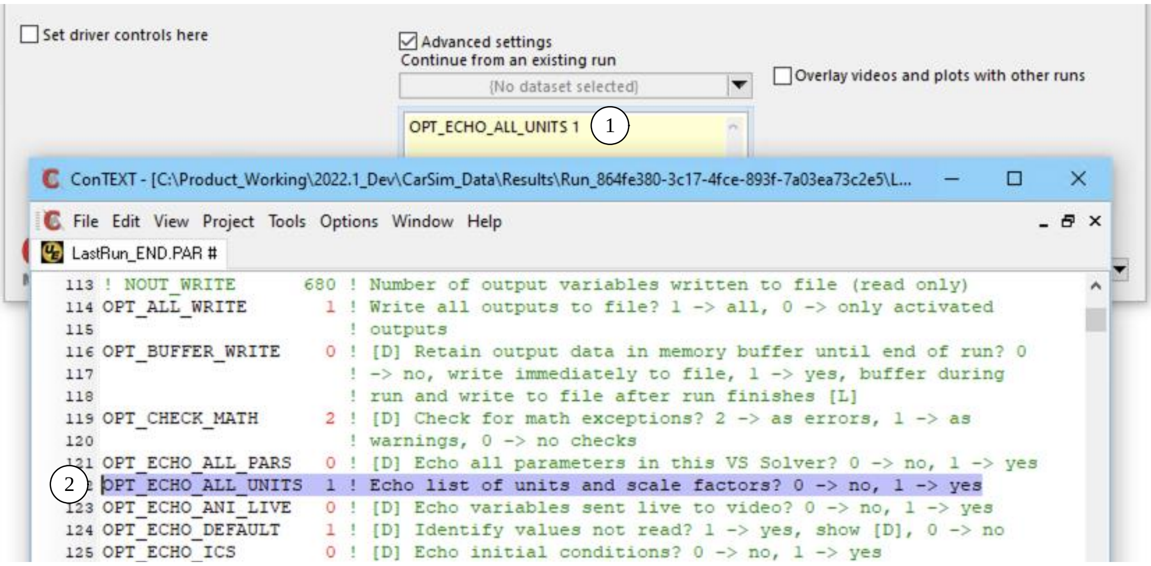
Viewing Existing Units in a VS Math Model

VS Math Models include some parameter for setting how much information to show in an Echo file. One of these, `OPT_ECHO_ALL_UNITS` ② (Figure 1), can be set to 1 (the default setting is 0) in a miscellaneous yellow field ① to cause the VS Math Model to list all available units in the Echo files written at the start and end of the run. For example, Figure 2 shows part of an Echo file made with `OPT_ECHO_ALL_UNITS` set to 1.

The information is shown with the `REDEFINE_UNITS` VS Command, such that the Echo file will install all units in the model if used as an input file to repeat or continue a run. The syntax is simple:

`REDEFINE_UNITS` *keyword name gain*

Keyword is a “word” (a set of non-blank characters) that will be interpreted by the VS Math Model to identify a type of units. As with most keywords in a VS Math Model, the keyword is not case sensitive. *Name* is a case-sensitive name for the units that will be written in outputs generated by the Math Model, such as Echo files and output motion files used for plots.



□ Set driver controls here

☑ Advanced settings

Continue from an existing run

{No dataset selected}

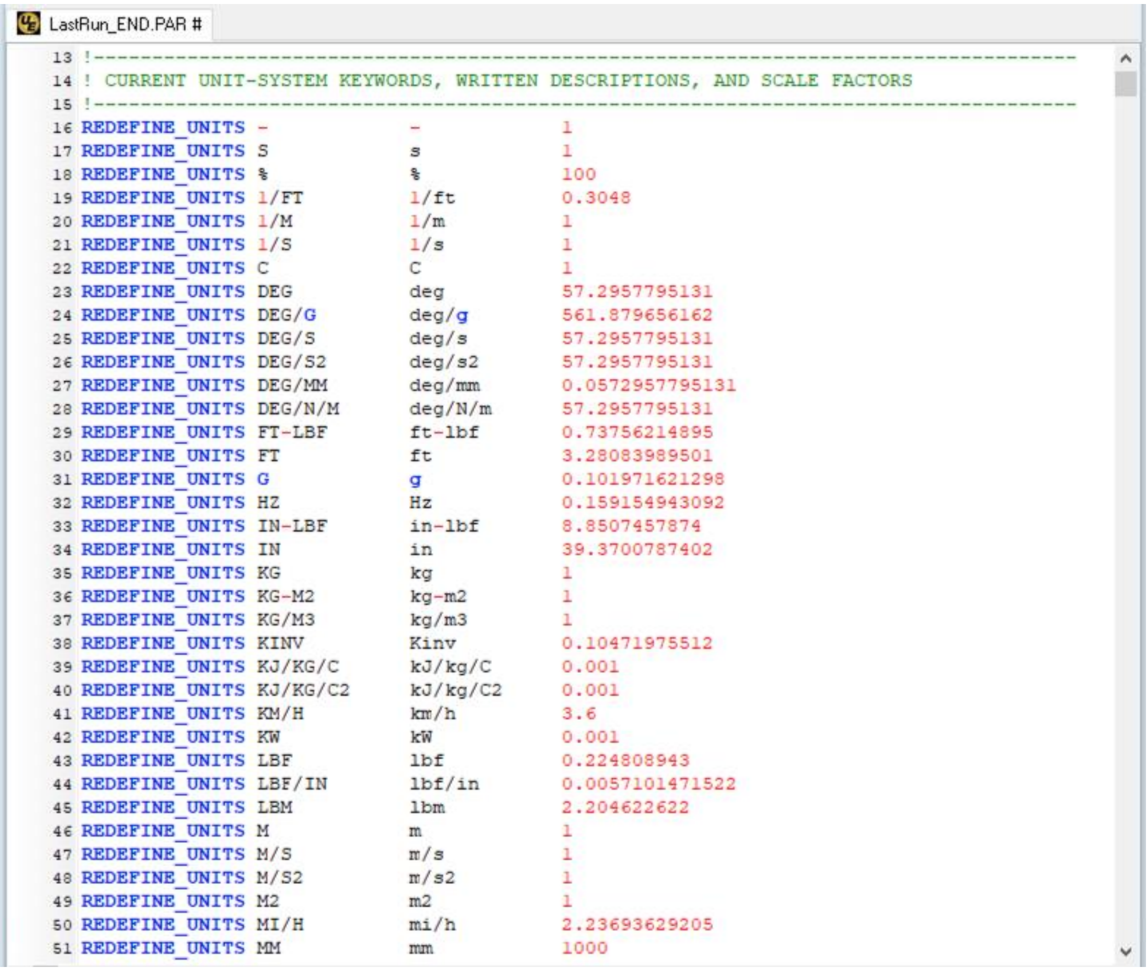
□ Overlay videos and plots with other runs

OPT_ECHO_ALL_UNITS 1

ConTEXT - [C:\Product_Working\2022.1_Dev\CarSim_Data\Results\Run_864fe380-3c17-4fce-893f-7a03ea73c2e5\L...

```
! NOUT_WRITE       680 ! Number of output variables written to file (read only)
OPT_ALL_WRITE        1 ! Write all outputs to file? 1 -> all, 0 -> only activated
                       ! outputs
OPT_BUFFER_WRITE     0 ! [D] Retain output data in memory buffer until end of run? 0
                       ! -> no, write immediately to file, 1 -> yes, buffer during
                       ! run and write to file after run finishes [L]
OPT_CHECK_MATH       2 ! [D] Check for math exceptions? 2 -> as errors, 1 -> as
                       ! warnings, 0 -> no checks
OPT_ECHO_ALL_PARS    0 ! [D] Echo all parameters in this VS Solver? 0 -> no, 1 -> yes
OPT_ECHO_ALL_UNITS   1 ! Echo list of units and scale factors? 0 -> no, 1 -> yes
OPT_ECHO_ANI_LIVE    0 ! [D] Echo variables sent live to video? 0 -> no, 1 -> yes
OPT_ECHO_DEFAULT     1 ! [D] Identify values not read? 1 -> yes, show [D], 0 -> no
OPT_ECHO_ICS         0 ! [D] Echo initial conditions? 0 -> no, 1 -> yes
```

Figure 1. Use the parameter OPT_ECHO_ALL_UNITS to show all units in a VS Math Model.



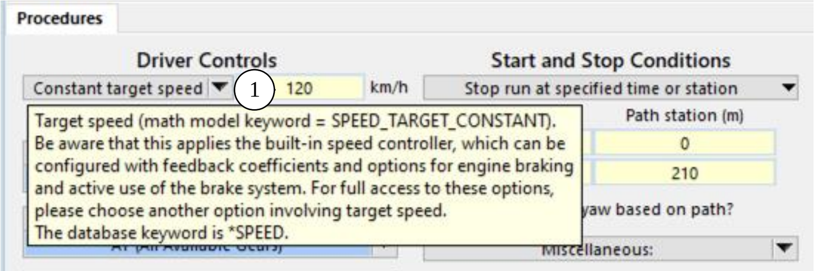
```
!----------------------------------------------------------------------------------
! CURRENT UNIT-SYSTEM KEYWORDS, WRITTEN DESCRIPTIONS, AND SCALE FACTORS
!----------------------------------------------------------------------------------
REDEFINE_UNITS -             -           1
REDEFINE_UNITS S             s           1
REDEFINE_UNITS %             %           100
REDEFINE_UNITS 1/FT          1/ft        0.3048
REDEFINE_UNITS 1/M           1/m         1
REDEFINE_UNITS 1/S           1/s         1
REDEFINE_UNITS C             C           1
REDEFINE_UNITS DEG           deg         57.2957795131
REDEFINE_UNITS DEG/G         deg/g       561.879656162
REDEFINE_UNITS DEG/S         deg/s       57.2957795131
REDEFINE_UNITS DEG/S2        deg/s2      57.2957795131
REDEFINE_UNITS DEG/MM        deg/mm      0.0572957795131
REDEFINE_UNITS DEG/N/M       deg/N/m     57.2957795131
REDEFINE_UNITS FT-LBF        ft-lbf      0.73756214895
REDEFINE_UNITS FT            ft          3.28083989501
REDEFINE_UNITS G             g           0.101971621298
REDEFINE_UNITS HZ            Hz          0.159154943092
REDEFINE_UNITS IN-LBF        in-lbf      8.8507457874
REDEFINE_UNITS IN            in          39.3700787402
REDEFINE_UNITS KG            kg          1
REDEFINE_UNITS KG-M2         kg-m2       1
REDEFINE_UNITS KG/M3         kg/m3       1
REDEFINE_UNITS KINV          Kinv        0.10471975512
REDEFINE_UNITS KJ/KG/C       kJ/kg/C     0.001
REDEFINE_UNITS KJ/KG/C2      kJ/kg/C2    0.001
REDEFINE_UNITS KM/H          km/h        3.6
REDEFINE_UNITS KW            kW          0.001
REDEFINE_UNITS LBF           lbf         0.224808943
REDEFINE_UNITS LBF/IN        lbf/in      0.0057101471522
REDEFINE_UNITS LBM           lbm         2.204622622
REDEFINE_UNITS M             m           1
REDEFINE_UNITS M/S           m/s         1
REDEFINE_UNITS M/S2          m/s2        1
REDEFINE_UNITS M2            m2          1
REDEFINE_UNITS MI/H          mi/h        2.23693629205
REDEFINE_UNITS MM            mm          1000
```

Figure 2. The top part of an Echo file showing units definitions.

Gain is a number or numerical expression that is multiplied by the SI value of a variable to convert it to the user units. For example, line 34 show the gain for the units `IN` (inches) is 39.3700787402, the scale factor between SI meters and inches. The gain could also be set with the more precise numerical expression 1/0.0254. The gain for `FT` (feet) could be set with the numerical expression 1/0.0254/12, which evaluates to 3.28083989501(see line 30).

Using Alternative Units for a Parameter

There are occasions when it is more convenient to use alternative units for a few parameters. For example, consider the constant target speed used in the Quick Start Guide examples in each product database. When the speed option in the Procedure screen is set to **Constant target speed** (1) (Figure 3), an adjacent yellow field appears with the label "km/h." Right-click on the field to view the information shown in the figure, including the keyword `SPEED_TARGET_CONSTANT`.



Constant target speed ▼ 120 km/h

Stop run at specified time or station ▼

Path station (m)

0

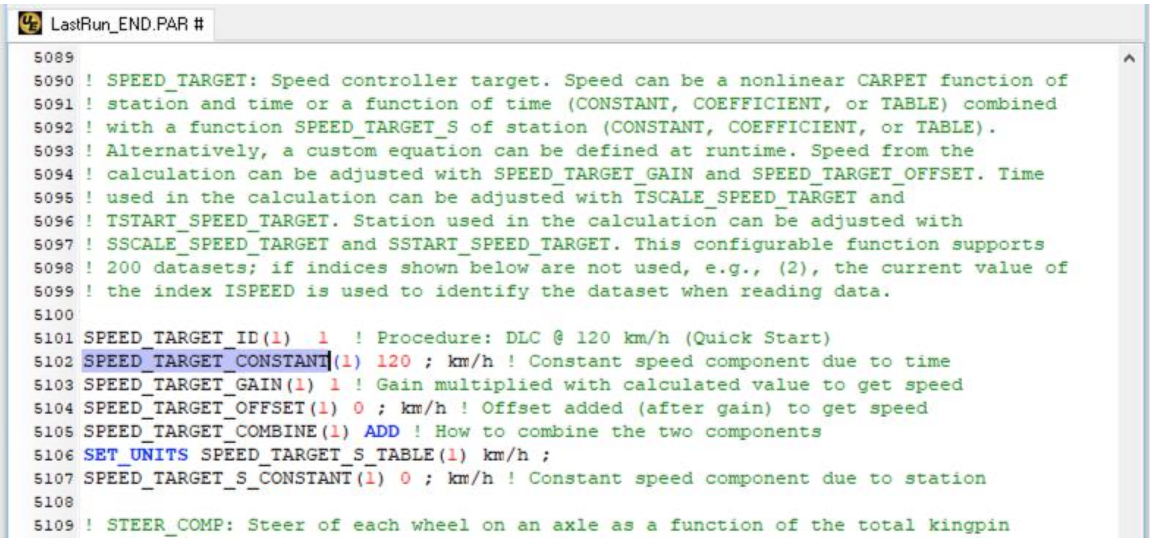
210

yaw based on path?

Target speed (math model keyword = SPEED_TARGET_CONSTANT).
Be aware that this applies the built-in speed controller, which can be configured with feedback coefficients and options for engine braking and active use of the brake system. For full access to these options, please choose another option involving target speed.
The database keyword is *SPEED.

Figure 3. Information about the constant target speed used in the Quick Start Guide example.

That setting appears in the Echo file for the run (line 5102, Figure 4).



LastRun_END.PAR #

```
5089
5090 ! SPEED_TARGET: Speed controller target. Speed can be a nonlinear CARPET function of
5091 ! station and time or a function of time (CONSTANT, COEFFICIENT, or TABLE) combined
5092 ! with a function SPEED_TARGET_S of station (CONSTANT, COEFFICIENT, or TABLE).
5093 ! Alternatively, a custom equation can be defined at runtime. Speed from the
5094 ! calculation can be adjusted with SPEED_TARGET_GAIN and SPEED_TARGET_OFFSET. Time
5095 ! used in the calculation can be adjusted with TSCALE_SPEED_TARGET and
5096 ! TSTART_SPEED_TARGET. Station used in the calculation can be adjusted with
5097 ! SSCALE_SPEED_TARGET and SSTART_SPEED_TARGET. This configurable function supports
5098 ! 200 datasets; if indices shown below are not used, e.g., (2), the current value of
5099 ! the index ISPEED is used to identify the dataset when reading data.
5100
5101 SPEED_TARGET_ID(1)  1   ! Procedure: DLC @ 120 km/h (Quick Start)
5102 SPEED_TARGET_CONSTANT(1) 120 ; km/h ! Constant speed component due to time
5103 SPEED_TARGET_GAIN(1) 1 ! Gain multiplied with calculated value to get speed
5104 SPEED_TARGET_OFFSET(1) 0 ; km/h ! Offset added (after gain) to get speed
5105 SPEED_TARGET_COMBINE(1) ADD ! How to combine the two components
5106 SET_UNITS SPEED_TARGET_S_TABLE(1) km/h ;
5107 SPEED_TARGET_S_CONSTANT(1) 0 ; km/h ! Constant speed component due to station
5108
5109 ! STEER_COMP: Steer of each wheel on an axle as a function of the total kingpin
```

Figure 4. SPEED_TARGET settings in the Echo file.

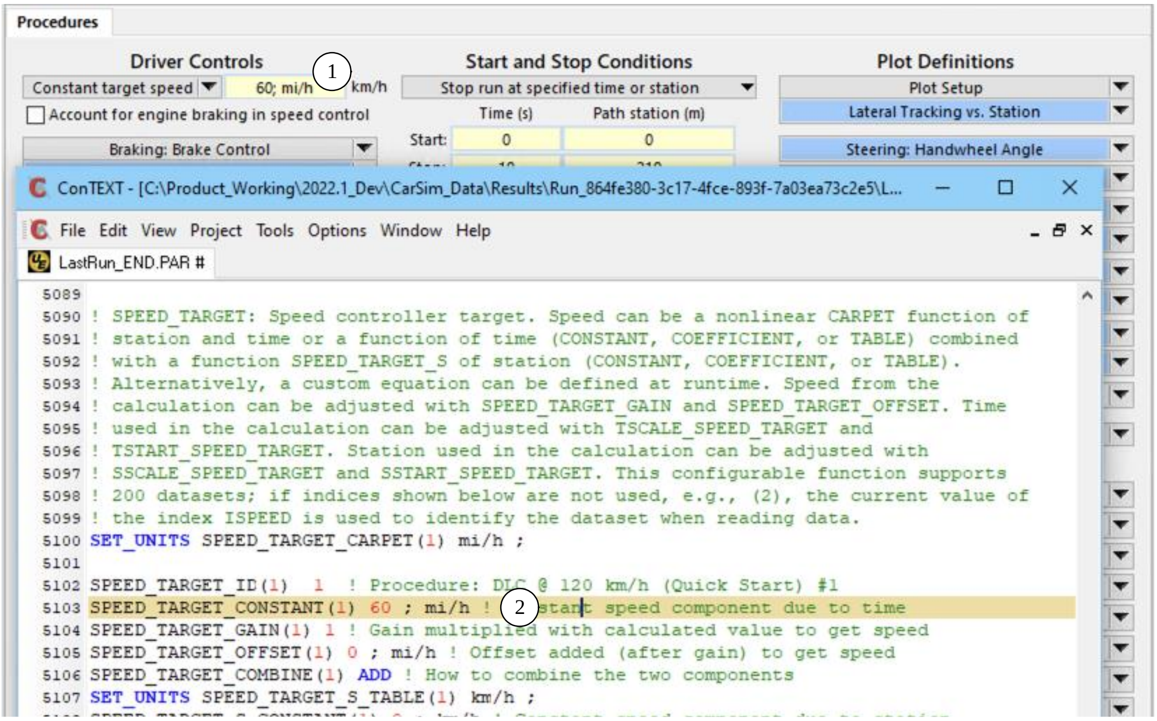
Notice line has four pieces of information:

1. The keyword `SPEED_TARGET_CONSTANT(1)` identifies a part of the model.
2. The number 120 provides a value.

3. The units are specified with the keyword `km/h`, separated from the numerical value by the semi-colon.

4. The parameter is described with a comment that follows the comment indicator ‘!’, such that the VS Math Model ignores that text if it were to be reused as an input.

Suppose you want to specify the constant target speed as 60 mi/h. This is done simply by mimicking the syntax shown in the Echo file. Provide the value, then a semicolon, then the units (1) (Figure 5). The figure also shows the Echo file for the run made with the new settings and confirms that the speed target was set with units of mi/h (2).


Procedures
1
2
ConTEXT - [C:\Product_Working\2022.1_Dev\CarSim_Data\Results\Run_864fe380-3c17-4fce-893f-7a03ea73c2e5\L...
File Edit View Project Tools Options Window Help
LastRun_END.PAR #

Driver Controls

Constant target speed: 60; mi/h km/h

☐ Account for engine braking in speed control

Braking: Brake Control

Start and Stop Conditions

Stop run at specified time or station

	Time (s)	Path station (m)
Start:	0	0

Plot Definitions

Plot Setup

Lateral Tracking vs. Station

Steering: Handwheel Angle

```

! SPEED_TARGET: Speed controller target. Speed can be a nonlinear CARPET function of
! station and time or a function of time (CONSTANT, COEFFICIENT, or TABLE) combined
! with a function SPEED_TARGET_S of station (CONSTANT, COEFFICIENT, or TABLE).
! Alternatively, a custom equation can be defined at runtime. Speed from the
! calculation can be adjusted with SPEED_TARGET_GAIN and SPEED_TARGET_OFFSET. Time
! used in the calculation can be adjusted with TSCALE_SPEED_TARGET and
! TSTART_SPEED_TARGET. Station used in the calculation can be adjusted with
! SSCALE_SPEED_TARGET and SSTART_SPEED_TARGET. This configurable function supports
! 200 datasets; if indices shown below are not used, e.g., (2), the current value of
! the index ISPEED is used to identify the dataset when reading data.
SET_UNITS SPEED_TARGET_CARPET(1) mi/h ;

SPEED_TARGET_ID(1)  1  ! Procedure: DLC @ 120 km/h (Quick Start) #1
SPEED_TARGET_CONSTANT(1) 60 ; mi/h !  stant speed component due to time
SPEED_TARGET_GAIN(1) 1 ! Gain multiplied with calculated value to get speed
SPEED_TARGET_OFFSET(1) 0 ; mi/h ! Offset added (after gain) to get speed
SPEED_TARGET_COMBINE(1) ADD ! How to combine the two components
SET_UNITS SPEED_TARGET_S_TABLE(1) km/h ;
```

Figure 5. Set units and value using a semi-colon separator in a yellow field.

Figure 6 show a video image and plot for the example run. Although the speed target was set with units of mi/h, the plot variables are still scaled to show km/h (with the target speed being 96.5606 km/h).

Suppose we want the plots to also show speeds using mi/h. This is also easy to do, as described next.

Change the Units for Output Variables

The units for variables (import, output, state variable) and Configurable Functions may be modified using the `SET_UNITS` VS Command (documented in the *VS Commands Manual*).

Consider the **Plot Setup** screen used to plot the speed variables vs. station (distance travelled) in Figure 7. The second yellow field specifies the variables to cross-plot for the three plots (2).

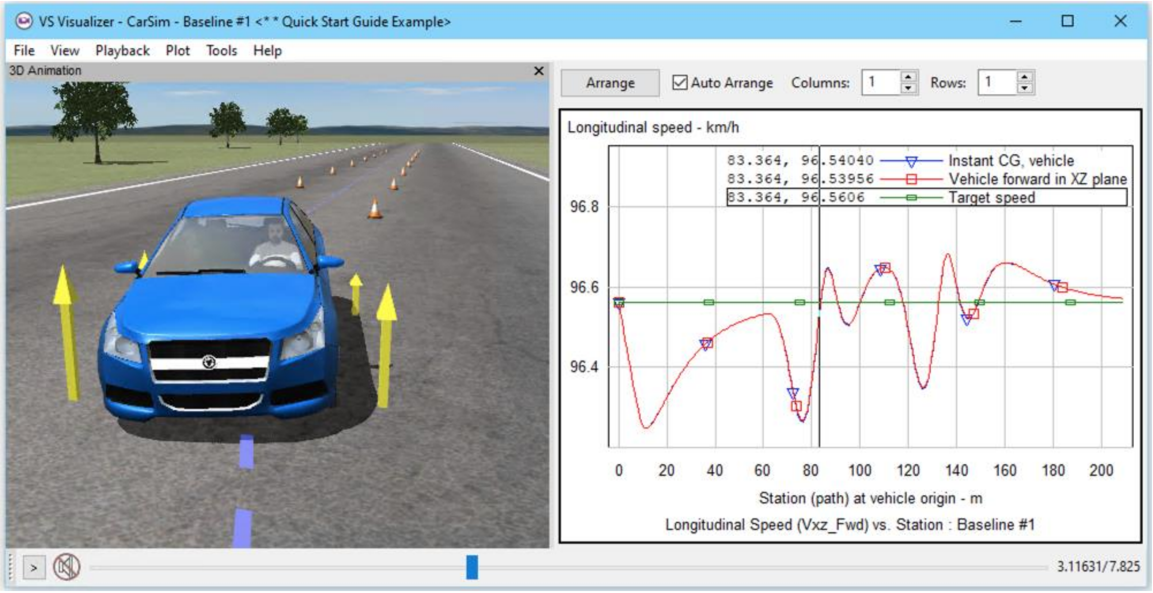
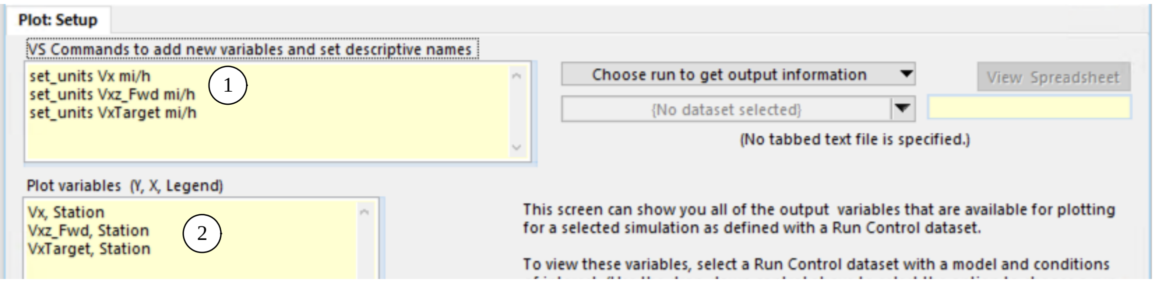

VS Visualizer - CarSim - Baseline #1 <* * Quick Start Guide Example>
File
View
Playback
Plot
Tools
Help
3D Animation
Arrange
☑ Auto Arrange
Columns: 1
Rows: 1
Longitudinal speed - km/h
83.364, 96.54040 Instant CG, vehicle
83.364, 96.53956 Vehicle forward in XZ plane
83.364, 96.5606 Target speed
96.8
96.6
96.4
0
20
40
60
80
100
120
140
160
180
200
Station (path) at vehicle origin - m
Longitudinal Speed (Vxz_Fwd) vs. Station : Baseline #1
3.11631/7.825

Figure 6. Video and plot for target speed of 60 mi/h.



Plot: Setup

VS Commands to add new variables and set descriptive names

```
set_units Vx mi/h
set_units Vxz_Fwd mi/h
set_units VxTarget mi/h
```

Choose run to get output information

(No dataset selected)

View Spreadsheet

(No tabbed text file is specified.)

Plot variables (Y, X, Legend)

```
Vx, Station
Vxz_Fwd, Station
VxTarget, Station
```

This screen can show you all of the output variables that are available for plotting for a selected simulation as defined with a Run Control dataset.

To view these variables, select a Run Control dataset with a model and conditions

Figure 7. Plot Setup screen for speed variables.

The first yellow field (1) is normally blank. However, in this example, we have added three VS Commands to set the units of the three speed variables to mi/h. Figure 8 shows that the plots now show the speeds in mi/h, with the target speed exactly 60.0 mi/h.

> **Note** The **Plot Setup** screen is a convenient place to use the `SET_UNITS` commands for these plot variables, but the settings could be placed in any miscellaneous fields that were used in setting up the simulation.

The internal calculations made by the VS Math Model for the two runs (with results in Figure 6 and Figure 8) are identical. The only difference is that the internal speed variables, with SI units of m/s, were scaled differently when written to the output files used by VS Visualizer to make the plots.

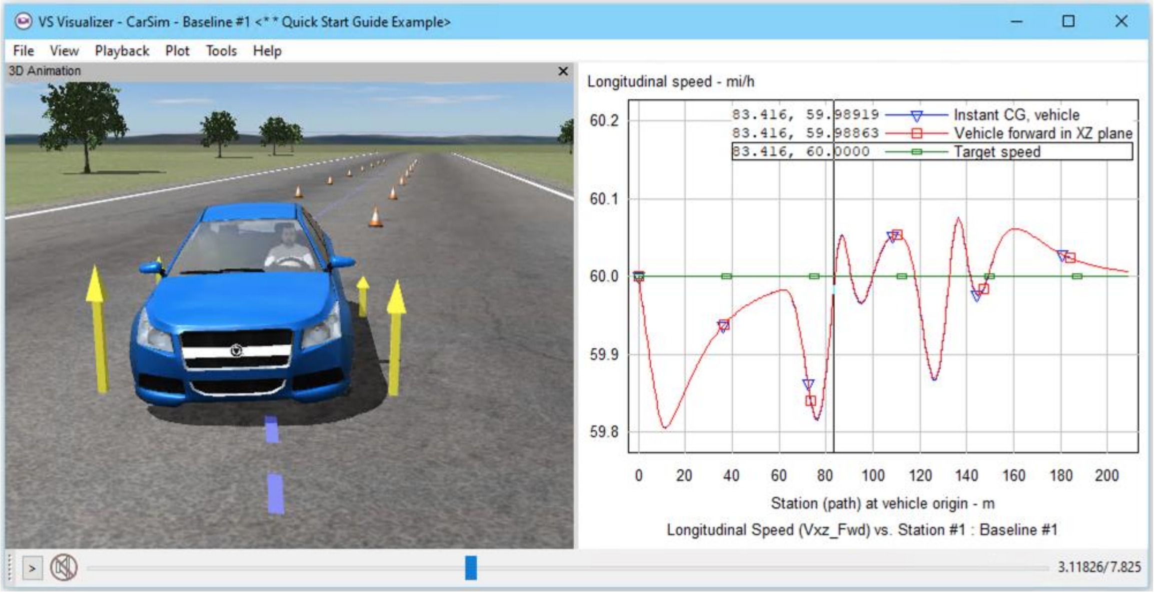

VS Visualizer - CarSim - Baseline #1 <* * Quick Start Guide Example>
File View Playback Plot Tools Help
3D Animation
Longitudinal speed - mi/h
83.416, 59.98919 Instant CG, vehicle
83.416, 59.98863 Vehicle forward in XZ plane
83.416, 60.0000 Target speed
60.2
60.1
60.0
59.9
59.8
0 20 40 60 80 100 120 140 160 180 200
Station (path) at vehicle origin - m
Longitudinal Speed (Vxz_Fwd) vs. Station #1 : Baseline #1
3.11826/7.825

Figure 8. Video and plot showing speeds with mi/h.

Define New Units

Now suppose you want to show the cross-plots using miles, rather than meters.

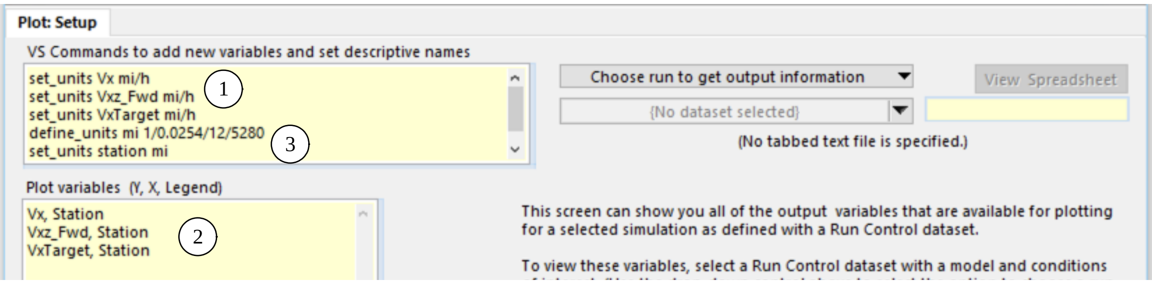
The listing of available units shown earlier (Figure 2, page 2) does not list units for miles. (The only units involving miles are mi/h.)

New units are defined with the VS Command `DEFINE_UNITS`, with the syntax:

`DEFINE_UNITS` *name gain*

where *name* is the case-sensitive name written in Echo files and used in plots and *gain* is the scale factor that is multiplied by the SI value to obtain a value in the new units. The command also makes an all-caps version of *name* that is installed so the new units name can be recognized regardless of case.

Going back to the **Plot Setup** screen for the plots shown (Figure 9), we add two lines (3): a `DEFINE_UNITS` command to define `mi` as miles, and a `SET_UNITS` command to set the units of the output variable `Station` to miles. In defining the units, the gain divides by 0.0254 (meters per inch), 12 (inches per foot), and 5280 (feet per mile).


Plot: Setup
VS Commands to add new variables and set descriptive names
1
3
Plot variables (Y, X, Legend)
2
Choose run to get output information
{No dataset selected}
View Spreadsheet
(No tabbed text file is specified.)

```
set_units Vx mi/h
set_units Vxz_Fwd mi/h
set_units VxTarget mi/h
define_units mi 1/0.0254/12/5280
set_units station mi
```

```
Vx, Station
Vxz_Fwd, Station
VxTarget, Station
```

This screen can show you all of the output variables that are available for plotting for a selected simulation as defined with a Run Control dataset.

To view these variables, select a Run Control dataset with a model and conditions

Figure 9. Defining new units (mile) and setting station to use them.

Figure 10 shows the resulting plots in which station is now displayed as miles travelled.

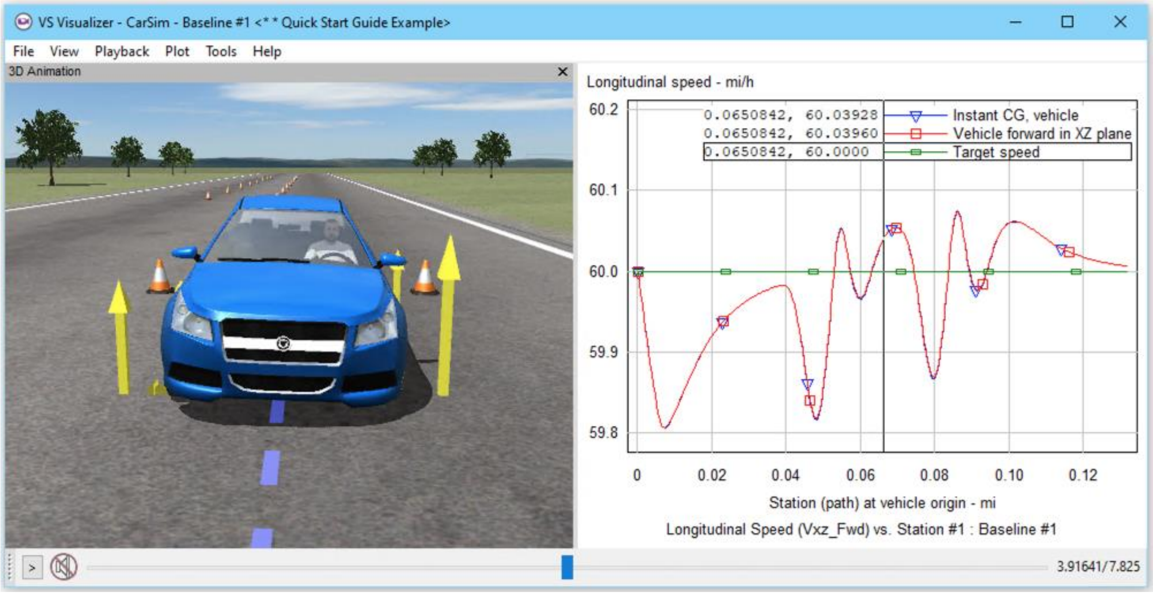

VS Visualizer - CarSim - Baseline #1 <* * Quick Start Guide Example>
File
View
Playback
Plot
Tools
Help
3D Animation
Longitudinal speed - mi/h
0.0650842, 60.03928
Instant CG, vehicle
0.0650842, 60.03960
Vehicle forward in XZ plane
0.0650842, 60.0000
Target speed
60.2
60.1
60.0
59.9
59.8
0
0.02
0.04
0.06
0.08
0.10
0.12
Station (path) at vehicle origin - mi
Longitudinal Speed (Vxz_Fwd) vs. Station #1 : Baseline #1
3.91641/7.825

Figure 10. Video and plot showing speed (mi/h) versus distance (miles).