NMPB08 – Road Emission Model

Reference Manual

Van Maercke Dirk
CSTB Grenoble
19/09/2012

The rest of this file was generated using Doxygen 1.7.4

Table of Contents

NMPB08 Road Emission Model

The NMPB Road Emission Model is defined in "Prévision du bruit routier, partie 1 - Calcul des émissions sonores dues au traffic routier" published by SETRA, June 2009)

Public functions defined in the RoadEmissionNMPB08 library

Calculate the overall sound power level (per meter) of the equivalent line source as a function of traffic and road surface characteristics

- NMPB08_Lwm

Calculate the traction an rolling components of the sound power level of the equivalent line source as a function of traffic and road surface characteristics

- NMPB08_Lwm_rolling
- NMPB08_Lwm_traction

Calculate the traction an rolling components of the sound power level of a single vehicle as a function of traffic conditions and road surface characteristics

- NMPB08_Lw_rolling
- NMPB08_Lw_traction

Declare a user-defined road surface type

- NMPB08_DefineRoadSurfaceType

Namespace Index

Namespace List

Here is a list of all namespaces with brief descriptions:

Class Index

Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

File Index

File List

Here is a list of all files with brief descriptions:

Namespace Documentation

CalculRoadEmissionNMPB Namespace Reference

Classes

- class CalculRoadEmission

Class used to road emission calculations. Functions

- bool DefineRoadSurfaceType (int idRoadSurface, double AVL, double BVL, double Vref_VL, double APL, double BPL, double Vref_PL, double const *spectrum)
  Definition of a new road surface type with data needed for rolling component and for spectrum.

Variables

- const int ThirdOctaveFrequencies [] = {100, 125, 160, 200, 250, 315, 400, 500, 630, 800, 1000, 1250, 1600, 2000, 2500, 3150, 4000, 5000}
  Frequencies in third octave band.
- const int OctaveFrequencies [] = {125, 250, 500, 1000, 2000, 4000}
  Frequencies in octave band.
- const int AllFrequencies [] = {0}
  Just the global value in dB(A)
- const int frequencyNumber = 18
  frequencies maximal number
- const double spectralDistribution_draining [] = {-22, -22, -20, -17, -15, -12, -10, -8, -9, -9, -10, -11, -12, -13, -16, -18, -20, -23}
  spectral distribution for draining ground (§ 2.8, array (2.23), p. 29)
- const double spectralDistribution_nonDraining [] = {-27, -26, -24, -21, -19, -16, -14, -11, -11, -8, -7, -8, -10, -13, -16, -18, -21, -23}
  spectral distribution for non draining ground (§ 2.8, array (2.23), p. 29)
- const double spectralFilterA [] = {-19.1, -16.1, -13.4, -10.9, -8.6, -6.6, -4.8, -3.2, -1.9, -0.8, 0.0, 0.6, 1.0, 1.2, 1.3, 1.2, 1.0, 0.5}
  Filter A corrections.
- map< int, RoadSurfaceDescription > RoadSurfaceDescriptionMap
  Road surface descriptions by id (RoadSurfaceType) : global variable.

Function Documentation

bool CalculRoadEmissionNMPB::DefineRoadSurfaceType (int*idRoadSurface*, double*AVL*, double*BVL*, double*Vref_VL*, double*APL*, double*BPL*, double*Vref_PL*, double const **spectrum*)

Definition of a new road surface type with data needed for rolling component and for spectrum.

Parameters:

idRoadSurface	The road surface id
AVL	The AVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)

BVL	The BVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
Vref_VL	The Vref_VL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
APL	The APL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
BPL	The BPL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
Vref_PL	The Vref_PL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
spectrum	The spectral distribution for this road surface (if null, non draining default values will be used)

Returns:

true if all OK

Definition at line 82 of file CalculRoadEmission.cpp.

Variable Documentation

const int CalculRoadEmissionNMPB::AllFrequencies[] = {0}

Just the global value in dB(A)

Definition at line 27 of file CalculRoadEmission.cpp.

const int CalculRoadEmissionNMPB::frequencyNumber = 18

frequencies maximal number

Definition at line 32 of file CalculRoadEmission.cpp.

const int CalculRoadEmissionNMPB::OctaveFrequencies[] = {125, 250, 500, 1000, 2000, 4000}

Frequencies in octave band.

Definition at line 23 of file CalculRoadEmission.cpp.

map<int,RoadSurfaceDescription> CalculRoadEmissionNMPB::RoadSurfaceDescriptionMap

Road surface descriptions by id (RoadSurfaceType) : global variable.

Definition at line 56 of file CalculRoadEmission.cpp.

const double CalculRoadEmissionNMPB::spectralDistribution_draining[] = {-22, -22, -20, -17, -15, -12, -10, -8, -9, -9, -10, -11, -12, -13, -16, -18, -20, -23}

spectral distribution for draining ground (§ 2.8, array (2.23), p. 29)

Definition at line 42 of file CalculRoadEmission.cpp.

const double CalculRoadEmissionNMPB::spectralDistribution_nonDraining[] = {-27, -26, -24, -21, -19, -16, -14, -11, -11, -8, -7, -8, -10, -13, -16, -18, -21, -23}

spectral distribution for non draining ground (§ 2.8, array (2.23), p. 29)

Definition at line 46 of file CalculRoadEmission.cpp.

const double CalculRoadEmissionNMPB::spectralFilterA[] = {-19.1, -16.1, -13.4, -10.9, -8.6, -6.6, -4.8, -3.2, -1.9, -0.8, 0.0, 0.6, 1.0, 1.2, 1.3, 1.2, 1.0, 0.5}

Filter A corrections.

Definition at line 50 of file CalculRoadEmission.cpp.

const int CalculRoadEmissionNMPB::ThirdOctaveFrequencies[] = {100, 125, 160, 200, 250, 315, 400, 500, 630, 800, 1000, 1250, 1600, 2000, 2500, 3150, 4000, 5000}

Frequencies in third octave band.

Definition at line 19 of file CalculRoadEmission.cpp.

RoadSurfaceDescriptionNMPB Namespace Reference

Classes

- class RoadSurfaceDescription

Class used to define a road surface.

Class Documentation

CalculRoadEmissionNMPB::CalculRoadEmission Class Reference

Class used to road emission calculations.
`#include <CalculRoadEmission.h>`

Public Member Functions

- CalculRoadEmission (void)
  Initialization of CalculRoadEmission class.
- CalculRoadEmission (RoadSpectrumType specType, RoadSurfaceType surfType)
  Initialization of CalculRoadEmission class with given SpectrumType and the given surface type.
- double * SoundPowerLevelPerMeter (RoadTraffic *roadTraffic)
  Calculation of the Sound power level per source line meter.
- double * Lwm_rolling (RoadTraffic *roadTraffic)
  Calculation of the rolling component of the Sound power level per source line meter.
- double * Lwm_traction (RoadTraffic *roadTraffic)
  Calculation of the traction component of the Sound power level per source line meter.
- double * Lw_rolling (RoadVehicleType vtype, double vehicleSpeed, RoadSurfaceType surfaceType, double surfaceAge)
  Calculation of the rolling component of the Sound power level per source line meter for a vehicle type.
- double * Lw_traction (RoadVehicleType vtype, double vehicleSpeed, RoadFlowType flowType, double ramp)
  Calculation of the traction component of the Sound power level per source line meter for a vehicle type.

Detailed Description

Class used to road emission calculations.

Definition at line 50 of file CalculRoadEmission.h.

Constructor & Destructor Documentation

CalculRoadEmissionNMPB::CalculRoadEmission::CalculRoadEmission (void)

Initialization of CalculRoadEmission class.

Definition at line 237 of file CalculRoadEmission.cpp.

CalculRoadEmissionNMPB::CalculRoadEmission::CalculRoadEmission (RoadSpectrumType *specType*, RoadSurfaceType *surfType*)

Initialization of CalculRoadEmission class with given SpectrumType and the given surface type.

Parameters:

specType	The spectrum type
surfType	The road surface type

Definition at line 263 of file CalculRoadEmission.cpp.

Member Function Documentation

double * CalculRoadEmissionNMPB::CalculRoadEmission::Lw_rolling (RoadVehicleType *vtype*, double *vehicleSpeed*, RoadSurfaceType *surfaceType*, double *surfaceAge*)

Calculation of the rolling component of the Sound power level per source line meter for a vehicle type.

Parameters:

vtype	The vehicle type
vehicleSpeed	The vehicle speed
surfaceType	The surface type
surfaceAge	The surface age

Returns:

the rolling component of the Sound power level per source line meter, for each third-octave band

Definition at line 636 of file CalculRoadEmission.cpp.

double * CalculRoadEmissionNMPB::CalculRoadEmission::Lw_traction (RoadVehicleType *vtype*, double *vehicleSpeed*, RoadFlowType *flowType*, double *ramp*)

Calculation of the traction component of the Sound power level per source line meter for a vehicle type.

Parameters:

vtype	The vehicle type
vehicleSpeed	The vehicle speed
flowType	The flow type
ramp	The ramp

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Definition at line 903 of file CalculRoadEmission.cpp.

double * CalculRoadEmissionNMPB::CalculRoadEmission::Lwm_rolling (RoadTraffic **roadTraffic*)

Calculation of the rolling component of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data

Returns:

The rolling component of the Sound power level per source line meter, for each third-octave band

Definition at line 570 of file CalculRoadEmission.cpp.

double * CalculRoadEmissionNMPB::CalculRoadEmission::Lwm_traction (RoadTraffic **roadTraffic*)

Calculation of the traction component of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Definition at line 837 of file CalculRoadEmission.cpp.

double * CalculRoadEmissionNMPB::CalculRoadEmission::SoundPowerLevelPerMeter (RoadTraffic *roadTraffic)

Calculation of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data

Returns:

the Sound power level per source line meter, for each third-octave band

Definition at line 322 of file CalculRoadEmission.cpp.

The documentation for this class was generated from the following files:

- CalculRoadEmission.h
- CalculRoadEmission.cpp

RoadSurfaceDescriptionNMPB::RoadSurfaceDescription Class Reference

Class used to define a road surface.

`#include <RoadSurfaceDescription.h>`

Public Member Functions

- RoadSurfaceDescription (void)
  Initialization of RoadSurfaceDescription class.
- RoadSurfaceDescription (double AVL, double BVL, double Vref_VL, double APL, double BPL, double Vref_PL, const double *spectrum, bool doNormalization)
  Initialization of RoadSurfaceDescription class with given data.
- double get_AVL ()
  *Gets the AVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)*
- double get_BVL ()
  *Gets the BVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)*
- double get_Vref_VL ()
  *Gets the Vref_VL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)*
- double get_APL ()
  *Gets the APL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)*
- double get_BPL ()
  *Gets the BPL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)*
- double get_Vref_PL ()
  *Gets the Vref_PL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)*
- map< int, double > get_spectralDistributionMap ()
  Gets the Sound power level by frequency.

Detailed Description

Class used to define a road surface.

Definition at line 20 of file RoadSurfaceDescription.h.

Constructor & Destructor Documentation

RoadSurfaceDescriptionNMPB::RoadSurfaceDescription::RoadSurfaceDescription (void)

Initialization of RoadSurfaceDescription class.

Definition at line 16 of file RoadSurfaceDescription.cpp.

RoadSurfaceDescriptionNMPB::RoadSurfaceDescription::RoadSurfaceDescription (double*AVL*, double*BVL*, double*Vref_VL*, double*APL*, double*BPL*, double*Vref_PL*, const double **spectrum*, bool*doNormalization*)

Initialization of RoadSurfaceDescription class with given data.

Parameters:

AVL	The AVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
BVL	The BVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
Vref_VL	The Vref_VL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
APL	The APL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
BPL	The BPL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
Vref_PL	The Vref_PL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
spectrum	The spectral distribution for this road surface
doNormalization	If true, do normalization

Definition at line 46 of file RoadSurfaceDescription.cpp.

Member Function Documentation

double RoadSurfaceDescriptionNMPB::RoadSurfaceDescription::get_APL () [inline]

Gets the APL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
Definition at line 73 of file RoadSurfaceDescription.h.

double RoadSurfaceDescriptionNMPB::RoadSurfaceDescription::get_AVL () [inline]

Gets the AVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
Definition at line 52 of file RoadSurfaceDescription.h.

double RoadSurfaceDescriptionNMPB::RoadSurfaceDescription::get_BPL () [inline]

Gets the BPL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
Definition at line 80 of file RoadSurfaceDescription.h.

double RoadSurfaceDescriptionNMPB::RoadSurfaceDescription::get_BVL () [inline]

Gets the BVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
Definition at line 59 of file RoadSurfaceDescription.h.

map<int,double> RoadSurfaceDescriptionNMPB::RoadSurfaceDescription::get_spectralDistributionMap () [inline]

Gets the Sound power level by frequency.
Definition at line 94 of file RoadSurfaceDescription.h.

double RoadSurfaceDescriptionNMPB::RoadSurfaceDescription::get_Vref_PL () [inline]

Gets the Vref_PL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
Definition at line 87 of file RoadSurfaceDescription.h.

double RoadSurfaceDescriptionNMPB::RoadSurfaceDescription::get_Vref_VL () [inline]

Gets the Vref_VL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
Definition at line 66 of file RoadSurfaceDescription.h.

The documentation for this class was generated from the following files:

- RoadSurfaceDescription.h
- RoadSurfaceDescription.cpp

RoadTraffic Struct Reference

Road traffic description.
`#include <RoadEmissionNMPB08.h>`

Public Attributes

- RoadSurfaceType surfaceType
  the road surface type
- double surfaceAge
  the surface age
- double ramp
  the ramp in percent (> 0 if rise, < 0 if down)
- int nbComponents
  the road traffic components number
- RoadTrafficComponent * traffic
  the road traffic components data (VL and PL)
- double _Lw_values [18]
  local buffer used for output values

Detailed Description

Road traffic description.
Definition at line 239 of file RoadEmissionNMPB08.h.

Member Data Documentation

double RoadTraffic::_Lw_values

local buffer used for output values
Definition at line 264 of file RoadEmissionNMPB08.h.

int RoadTraffic::nbComponents

the road traffic components number
Definition at line 256 of file RoadEmissionNMPB08.h.

double RoadTraffic::ramp

the ramp in percent (> 0 if rise, < 0 if down)
Definition at line 252 of file RoadEmissionNMPB08.h.

double RoadTraffic::surfaceAge

the surface age

Definition at line 248 of file RoadEmissionNMPB08.h.

RoadSurfaceType RoadTraffic::surfaceType

the road surface type
Definition at line 244 of file RoadEmissionNMPB08.h.

RoadTrafficComponent * RoadTraffic::traffic

the road traffic components data (VL and PL)
Definition at line 260 of file RoadEmissionNMPB08.h.

The documentation for this struct was generated from the following files:

- Linux/RoadEmissionNMPB08.h
- MingW/RoadEmissionNMPB08.h
- RoadEmissionNMPB08.h

RoadTrafficComponent Struct Reference

Road traffic component.
`#include <RoadEmissionNMPB08.h>`

Public Attributes

- RoadVehicleType vehicleType
  the vehicle type
- double trafficFlow
  the traffic flow : hourly vehicle flow rate, in vehicles/hour.
- double trafficSpeed
  the traffic speed
- RoadFlowType flowType
  the traffic type

Detailed Description

Road traffic component.

Definition at line 216 of file RoadEmissionNMPB08.h.

Member Data Documentation

RoadFlowType RoadTrafficComponent::flowType

the traffic type
Definition at line 233 of file RoadEmissionNMPB08.h.

double RoadTrafficComponent::trafficFlow

the traffic flow : hourly vehicle flow rate, in vehicles/hour.
Definition at line 225 of file RoadEmissionNMPB08.h.

double RoadTrafficComponent::trafficSpeed

the traffic speed
Definition at line 229 of file RoadEmissionNMPB08.h.

RoadVehicleType RoadTrafficComponent::vehicleType

the vehicle type
Definition at line 221 of file RoadEmissionNMPB08.h.

The documentation for this struct was generated from the following files:

- Linux/RoadEmissionNMPB08.h

- MingW/RoadEmissionNMPB08.h
- RoadEmissionNMPB08.h

File Documentation

CalculRoadEmission.cpp File Reference

Calculations of road emissions.

```
#include "../test_mem/safe_new.h"
#include "CalculRoadEmission.h"
#include <stdio.h>
```

Namespaces

- namespace CalculRoadEmissionNMPB

Defines

- #define NominalMedianFrequency ThirdOctaveFrequencies
  Nominal median frequencies array.

Functions

- bool CalculRoadEmissionNMPB::DefineRoadSurfaceType (int idRoadSurface, double AVL, double BVL, double Vref_VL, double APL, double BPL, double Vref_PL, double const *spectrum)
  Definition of a new road surface type with data needed for rolling component and for spectrum.

Variables

- const int CalculRoadEmissionNMPB::ThirdOctaveFrequencies [] = {100, 125, 160, 200, 250, 315, 400, 500, 630, 800, 1000, 1250, 1600, 2000, 2500, 3150, 4000, 5000}
  Frequencies in third octave band.
- const int CalculRoadEmissionNMPB::OctaveFrequencies [] = {125, 250, 500, 1000, 2000, 4000}
  Frequencies in octave band.
- const int CalculRoadEmissionNMPB::AllFrequencies [] = {0}
  Just the global value in dB(A)
- const int CalculRoadEmissionNMPB::frequencyNumber = 18
  frequencies maximal number
- const double CalculRoadEmissionNMPB::spectralDistribution_draining [] = {-22, -22, -20, -17, -15, -12, -10, -8, -9, -9, -10, -11, -12, -13, -16, -18, -20, -23}
  spectral distribution for draining ground (§ 2.8, array (2.23), p. 29)
- const double CalculRoadEmissionNMPB::spectralDistribution_nonDraining [] = {-27, -26, -24, -21, -19, -16, -14, -11, -11, -8, -7, -8, -10, -13, -16, -18, -21, -23}
  spectral distribution for non draining ground (§ 2.8, array (2.23), p. 29)
- const double CalculRoadEmissionNMPB::spectralFilterA [] = {-19.1, -16.1, -13.4, -10.9, -8.6, -6.6, -4.8, -3.2, -1.9, -0.8, 0.0, 0.6, 1.0, 1.2, 1.3, 1.2, 1.0, 0.5}
  Filter A corrections.
- map< int, RoadSurfaceDescription > CalculRoadEmissionNMPB::RoadSurfaceDescriptionMap
  Road surface descriptions by id (RoadSurfaceType) : global variable.

Detailed Description

Calculations of road emissions.

Author:
CSTB
Version:
1.0
Definition in file CalculRoadEmission.cpp.

Define Documentation

#define NominalMedianFrequency ThirdOctaveFrequencies

Nominal median frequencies array.
Definition at line 38 of file CalculRoadEmission.cpp.

CalculRoadEmission.h File Reference

Calculations of road emissions.

```
#include "RoadEmissionNMPB08.h"
#include "RoadSurfaceDescription.h"
#include <vector>
#include <math.h>
#include <map>
```

Classes

- class CalculRoadEmissionNMPB::CalculRoadEmission

Class used to road emission calculations. Namespaces

- namespace CalculRoadEmissionNMPB

Functions

- bool CalculRoadEmissionNMPB::DefineRoadSurfaceType (int idRoadSurface, double AVL, double BVL, double Vref_VL, double APL, double BPL, double Vref_PL, double const *spectrum)
  Definition of a new road surface type with data needed for rolling component and for spectrum.

Detailed Description

Calculations of road emissions.

Author:
CSTB

Version:
1.0

Definition in file CalculRoadEmission.h.

DataStructuresNMPB.h File Reference

Definition of main enumerations, constants and structures used in the call of the library functions.

Detailed Description

Definition of main enumerations, constants and structures used in the call of the library functions.

Author:

CSTB

Version:

1.0

Definition in file DataStructuresNMPB.h.

Doxyfile.dox File Reference

Linux/RoadEmissionNMPB08.h File Reference

Classes

- struct RoadTrafficComponent
- *Road traffic component.* struct RoadTraffic

Road traffic description. Defines

- #define _COMPILE_NMPB extern "C"

Enumerations

- enum RoadVehicleType { VehicleType_VL = 1, VehicleType_PL = 2, VehicleType_VL = 1, VehicleType_PL = 2, VehicleType_VL = 1, VehicleType_PL = 2 }
- *Vehicle type.* enum RoadFlowType { FlowType_CONST = 0, FlowType_ACC = 1, FlowType_DEC = 2, FlowType_START = 3, FlowType_STOP = 4, FlowType_CONST = 0, FlowType_ACC = 1, FlowType_DEC = 2, FlowType_START = 3, FlowType_STOP = 4, FlowType_CONST = 0, FlowType_ACC = 1, FlowType_DEC = 2, FlowType_START = 3, FlowType_STOP = 4 }
- *Flow type.* enum RoadSurfaceType { RoadSurface_Default = 0, RoadSurface_R1 = 1, RoadSurface_R2 = 2, RoadSurface_R3 = 3, RoadSurface_DR1 = 4, RoadSurface_DR2 = 5, RoadSurface_DR3 = 6, RoadSurface_UserDefined = 7, RoadSurface_Default = 0, RoadSurface_R1 = 1, RoadSurface_R2 = 2, RoadSurface_R3 = 3, RoadSurface_DR1 = 4, RoadSurface_DR2 = 5, RoadSurface_DR3 = 6, RoadSurface_UserDefined = 7, RoadSurface_Default = 0, RoadSurface_R1 = 1, RoadSurface_R2 = 2, RoadSurface_R3 = 3, RoadSurface_DR1 = 4, RoadSurface_DR2 = 5, RoadSurface_DR3 = 6, RoadSurface_UserDefined = 7 }
- *Road surface type.* enum RoadSpectrumType { Spectrum_dBA = 0, Spectrum_oct_A = 1, Spectrum_3oct_A = 2, Spectrum_oct_lin = 3, Spectrum_3oct_lin = 4, Spectrum_dBA = 0, Spectrum_oct_A = 1, Spectrum_3oct_A = 2, Spectrum_oct_lin = 3, Spectrum_3oct_lin = 4, Spectrum_dBA = 0, Spectrum_oct_A = 1, Spectrum_3oct_A = 2, Spectrum_oct_lin = 3, Spectrum_3oct_lin = 4 }

Spectrum type. Functions

- _COMPILE_NMPB double * NMPB08_Lwm (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the Sound power level per source line meter.
- _COMPILE_NMPB double * NMPB08_Lwm_rolling (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the rolling component of the Sound power level per source line meter.
- _COMPILE_NMPB double * NMPB08_Lwm_traction (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the traction component of the Sound power level per source line meter.
- _COMPILE_NMPB double * NMPB08_Lw_rolling (RoadVehicleType type, double vehicleSpeed, RoadSurfaceType surfaceType, double surfaceAge, RoadSpectrumType specType)
  Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed and road surface characteristics.
- _COMPILE_NMPB double * NMPB08_Lw_traction (RoadVehicleType type, double vehicleSpeed, RoadFlowType flowType, double ramp, RoadSpectrumType specType)
  Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed, ramp and flow type.
- _COMPILE_NMPB bool NMPB08_DefineRoadSurfaceType (int idRoadSurface, double AVL, double BVL, double Vref_VL, double APL, double BPL, double Vref_PL, double const *spectrum)
  Definition of a new road surface type with data needed for rolling component and for spectrum.

Variables

- const int BBUM_0_6 = RoadSurface_R1
  BBUM 0/6 road surface type.
- const int BBDR_0_10 = RoadSurface_DR1
  BBDr 0/10 road surface type.
- const int BBTM_0_6_type2 = RoadSurface_R1
  BBTM 0/6-type2 road surface type.
- const int BBTM_0_6_type1 = RoadSurface_R1
  BBTM 0/6-type1 road surface type.
- const int BBTM_0_10_type2 = RoadSurface_R1
  BBTM 0/10-type2 road surface type.
- const int BBSG_0_10 = RoadSurface_R2
  BBSG 0/10 road surface type.
- const int BBTM_0_10_type1 = RoadSurface_R2
  BBTM 0/10-type1 road surface type.
- const int BBUM_0_10 = RoadSurface_R2
  BBUM 0/10 road surface type.
- const int ECF = RoadSurface_R2
  ECF road surface type.
- const int BBSG_0_14 = RoadSurface_R3
  BBSG 0/14 road surface type.
- const int BBTM_0_14 = RoadSurface_R3
  BBTM 0/14 road surface type.
- const int ES_6_10 = RoadSurface_R3
  ES 6/10 road surface type.
- const int BC = RoadSurface_R3
  BC road surface type.
- const int ES_10_14 = RoadSurface_R3
  ES 10/14 road surface type.

Define Documentation

#define _COMPILE_NMPB extern "C"

Definition at line 38 of file RoadEmissionNMPB08.h.

Enumeration Type Documentation

enum RoadFlowType

Flow type.

Enumerator:

FlowType_CONST Constant flow.

FlowType_ACC Acceleration section.

FlowType_DEC Deceleration section.
FlowType_START Startup section.
FlowType_STOP Stopping section.
FlowType_CONST Constant flow.
FlowType_ACC Acceleration section.
FlowType_DEC Deceleration section.
FlowType_START Startup section.
FlowType_STOP Stopping section.
FlowType_CONST Constant flow.
FlowType_ACC Acceleration section.
FlowType_DEC Deceleration section.
FlowType_START Startup section.
FlowType_STOP Stopping section.

Definition at line 62 of file RoadEmissionNMPB08.h.

enum RoadSpectrumType

Spectrum type.

Enumerator:

Spectrum_dBA The global value in dB(A) : only one value.
Spectrum_oct_A Octave bands (dB(A)) : 6 frequencies from 125 to 4000.
Spectrum_3oct_A Third-octave bands (dB(A)) : 18 frequencies from 100 to 5000.
Spectrum_oct_lin Octave bands : 6 frequencies from 125 to 4000.
Spectrum_3oct_lin Third-octave bands : 18 frequencies from 100 to 5000.
Spectrum_dBA The global value in dB(A) : only one value.
Spectrum_oct_A Octave bands (dB(A)) : 6 frequencies from 125 to 4000.
Spectrum_3oct_A Third-octave bands (dB(A)) : 18 frequencies from 100 to 5000.
Spectrum_oct_lin Octave bands : 6 frequencies from 125 to 4000.
Spectrum_3oct_lin Third-octave bands : 18 frequencies from 100 to 5000.
Spectrum_dBA The global value in dB(A) : only one value.
Spectrum_oct_A Octave bands (dB(A)) : 6 frequencies from 125 to 4000.
Spectrum_3oct_A Third-octave bands (dB(A)) : 18 frequencies from 100 to 5000.
Spectrum_oct_lin Octave bands : 6 frequencies from 125 to 4000.
Spectrum_3oct_lin Third-octave bands : 18 frequencies from 100 to 5000.

Definition at line 187 of file RoadEmissionNMPB08.h.

enum RoadSurfaceType

Road surface type.

Enumerator:

RoadSurface_Default default type

RoadSurface_R1 R1 type.
RoadSurface_R2 R2 type.
RoadSurface_R3 R3 type.
RoadSurface_DR1 Draining R1 type.
RoadSurface_DR2 Draining R2 type.
RoadSurface_DR3 Draining R3 type.
RoadSurface_UserDefined User defined type.
RoadSurface_Default default type
RoadSurface_R1 R1 type.
RoadSurface_R2 R2 type.
RoadSurface_R3 R3 type.
RoadSurface_DR1 Draining R1 type.
RoadSurface_DR2 Draining R2 type.
RoadSurface_DR3 Draining R3 type.
RoadSurface_UserDefined User defined type.
RoadSurface_Default default type
RoadSurface_R1 R1 type.
RoadSurface_R2 R2 type.
RoadSurface_R3 R3 type.
RoadSurface_DR1 Draining R1 type.
RoadSurface_DR2 Draining R2 type.
RoadSurface_DR3 Draining R3 type.
RoadSurface_UserDefined User defined type.

Definition at line 89 of file RoadEmissionNMPB08.h.

enum RoadVehicleType

Vehicle type.
Enumerator:
VehicleType_VL Light vehicle.
VehicleType_PL Heavy goods vehicle.
VehicleType_VL Light vehicle.
VehicleType_PL Heavy goods vehicle.
VehicleType_VL Light vehicle.
VehicleType_PL Heavy goods vehicle.

Definition at line 47 of file RoadEmissionNMPB08.h.

Function Documentation

_COMPILE_NMPB bool NMPB08_DefineRoadSurfaceType (int*idRoadSurface*, double*AVL*, double*BVL*, double*Vref_VL*, double*APL*, double*BPL*, double*Vref_PL*, double const **spectrum*)

Definition of a new road surface type with data needed for rolling component and for spectrum.

Parameters:

idRoadSurface	The road surface id
AVL	The AVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
BVL	The BVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
Vref_VL	The Vref_VL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
APL	The APL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
BPL	The BPL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
Vref_PL	The Vref_PL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
spectrum	The spectral distribution for this road surface (if null, non draining default values will be used)

Returns:

true if all OK

Definition at line 172 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lw_rolling (RoadVehicleType*type*, double*vehicleSpeed*, RoadSurfaceType*surfaceType*, double*surfaceAge*, RoadSpectrumType*specType*)

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed and road surface characteristics.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
surfaceType	The surface type
surfaceAge	The surface age
specType	The spectrum type

Returns:

the rolling component of the Sound power level of the vehicle, in third octave bands

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed and road surface characteristics.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
surfaceType	The surface type
surfaceAge	The surface age
specType	The spectrum type

Returns:

the rolling component of the Sound power level per source line meter, for each third-octave band

Definition at line 100 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lw_traction (RoadVehicleType *type*, double *vehicleSpeed*, RoadFlowType *flowType*, double *ramp*, RoadSpectrumType *specType*)

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed, ramp and flow type.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
flowType	The flow type
ramp	The ramp
specType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed, ramp and flow type.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
flowType	The flow type
ramp	The ramp
specType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Definition at line 133 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lwm (RoadTraffic **roadTraffic*, RoadSpectrumType *spectrumType*)

Calculation of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

The Sound power level per source line meter, for each third-octave band

Definition at line 25 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lwm_rolling (RoadTraffic **roadTraffic*, RoadSpectrumType *spectrumType*)

Calculation of the rolling component of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

The rolling component of the Sound power level per source line meter, for each third-octave band
Definition at line 48 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lwm_traction (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)

Calculation of the traction component of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band
Definition at line 71 of file RoadEmissionNMPB08.cpp.

Variable Documentation

const int BBDR_0_10 = RoadSurface_DR1

BBDr 0/10 road surface type.
Definition at line 133 of file RoadEmissionNMPB08.h.

const int BBSG_0_10 = RoadSurface_R2

BBSG 0/10 road surface type.
Definition at line 149 of file RoadEmissionNMPB08.h.

const int BBSG_0_14 = RoadSurface_R3

BBSG 0/14 road surface type.
Definition at line 165 of file RoadEmissionNMPB08.h.

const int BBTM_0_10_type1 = RoadSurface_R2

BBTM 0/10-type1 road surface type.
Definition at line 153 of file RoadEmissionNMPB08.h.

const int BBTM_0_10_type2 = RoadSurface_R1

BBTM 0/10-type2 road surface type.
Definition at line 145 of file RoadEmissionNMPB08.h.

const int BBTM_0_14 = RoadSurface_R3

BBTM 0/14 road surface type.
Definition at line 169 of file RoadEmissionNMPB08.h.

const int BBTM_0_6_type1 = RoadSurface_R1

BBTM 0/6-type1 road surface type.
Definition at line 141 of file RoadEmissionNMPB08.h.

const int BBTM_0_6_type2 = RoadSurface_R1

BBTM 0/6-type2 road surface type.
Definition at line 137 of file RoadEmissionNMPB08.h.

const int BBUM_0_10 = RoadSurface_R2

BBUM 0/10 road surface type.
Definition at line 157 of file RoadEmissionNMPB08.h.

const int BBUM_0_6 = RoadSurface_R1

BBUM 0/6 road surface type.
Definition at line 129 of file RoadEmissionNMPB08.h.

const int BC = RoadSurface_R3

BC road surface type.
Definition at line 177 of file RoadEmissionNMPB08.h.

const int ECF = RoadSurface_R2

ECF road surface type.
Definition at line 161 of file RoadEmissionNMPB08.h.

const int ES_10_14 = RoadSurface_R3

ES 10/14 road surface type.
Definition at line 181 of file RoadEmissionNMPB08.h.

const int ES_6_10 = RoadSurface_R3

ES 6/10 road surface type.
Definition at line 173 of file RoadEmissionNMPB08.h.

MingW/RoadEmissionNMPB08.h File Reference

Classes

- struct RoadTrafficComponent
- *Road traffic component.* struct RoadTraffic

Road traffic description. Defines

- #define _COMPILE_NMPB extern "C"

Enumerations

- enum RoadVehicleType { VehicleType_VL = 1, VehicleType_PL = 2, VehicleType_VL = 1, VehicleType_PL = 2, VehicleType_VL = 1, VehicleType_PL = 2 }
- *Vehicle type.* enum RoadFlowType { FlowType_CONST = 0, FlowType_ACC = 1, FlowType_DEC = 2, FlowType_START = 3, FlowType_STOP = 4, FlowType_CONST = 0, FlowType_ACC = 1, FlowType_DEC = 2, FlowType_START = 3, FlowType_STOP = 4, FlowType_CONST = 0, FlowType_ACC = 1, FlowType_DEC = 2, FlowType_START = 3, FlowType_STOP = 4 }
- *Flow type.* enum RoadSurfaceType { RoadSurface_Default = 0, RoadSurface_R1 = 1, RoadSurface_R2 = 2, RoadSurface_R3 = 3, RoadSurface_DR1 = 4, RoadSurface_DR2 = 5, RoadSurface_DR3 = 6, RoadSurface_UserDefined = 7, RoadSurface_Default = 0, RoadSurface_R1 = 1, RoadSurface_R2 = 2, RoadSurface_R3 = 3, RoadSurface_DR1 = 4, RoadSurface_DR2 = 5, RoadSurface_DR3 = 6, RoadSurface_UserDefined = 7, RoadSurface_Default = 0, RoadSurface_R1 = 1, RoadSurface_R2 = 2, RoadSurface_R3 = 3, RoadSurface_DR1 = 4, RoadSurface_DR2 = 5, RoadSurface_DR3 = 6, RoadSurface_UserDefined = 7 }
- *Road surface type.* enum RoadSpectrumType { Spectrum_dBA = 0, Spectrum_oct_A = 1, Spectrum_3oct_A = 2, Spectrum_oct_lin = 3, Spectrum_3oct_lin = 4, Spectrum_dBA = 0, Spectrum_oct_A = 1, Spectrum_3oct_A = 2, Spectrum_oct_lin = 3, Spectrum_3oct_lin = 4, Spectrum_dBA = 0, Spectrum_oct_A = 1, Spectrum_3oct_A = 2, Spectrum_oct_lin = 3, Spectrum_3oct_lin = 4 }

Spectrum type. Functions

- _COMPILE_NMPB double * NMPB08_Lwm (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the Sound power level per source line meter.
- _COMPILE_NMPB double * NMPB08_Lwm_rolling (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the rolling component of the Sound power level per source line meter.
- _COMPILE_NMPB double * NMPB08_Lwm_traction (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the traction component of the Sound power level per source line meter.
- _COMPILE_NMPB double * NMPB08_Lw_rolling (RoadVehicleType type, double vehicleSpeed, RoadSurfaceType surfaceType, double surfaceAge, RoadSpectrumType specType)
  Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed and road surface characteristics.
- _COMPILE_NMPB double * NMPB08_Lw_traction (RoadVehicleType type, double vehicleSpeed, RoadFlowType flowType, double ramp, RoadSpectrumType specType)
  Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed, ramp and flow type.
- _COMPILE_NMPB bool NMPB08_DefineRoadSurfaceType (int idRoadSurface, double AVL, double BVL, double Vref_VL, double APL, double BPL, double Vref_PL, double const *spectrum)
  Definition of a new road surface type with data needed for rolling component and for spectrum.

Variables

- const int BBUM_0_6 = RoadSurface_R1
  BBUM 0/6 road surface type.
- const int BBDR_0_10 = RoadSurface_DR1
  BBDr 0/10 road surface type.
- const int BBTM_0_6_type2 = RoadSurface_R1
  BBTM 0/6-type2 road surface type.
- const int BBTM_0_6_type1 = RoadSurface_R1
  BBTM 0/6-type1 road surface type.
- const int BBTM_0_10_type2 = RoadSurface_R1
  BBTM 0/10-type2 road surface type.
- const int BBSG_0_10 = RoadSurface_R2
  BBSG 0/10 road surface type.
- const int BBTM_0_10_type1 = RoadSurface_R2
  BBTM 0/10-type1 road surface type.
- const int BBUM_0_10 = RoadSurface_R2
  BBUM 0/10 road surface type.
- const int ECF = RoadSurface_R2
  ECF road surface type.
- const int BBSG_0_14 = RoadSurface_R3
  BBSG 0/14 road surface type.
- const int BBTM_0_14 = RoadSurface_R3
  BBTM 0/14 road surface type.
- const int ES_6_10 = RoadSurface_R3
  ES 6/10 road surface type.
- const int BC = RoadSurface_R3
  BC road surface type.
- const int ES_10_14 = RoadSurface_R3
  ES 10/14 road surface type.

Define Documentation

#define _COMPILE_NMPB extern "C"

Definition at line 38 of file RoadEmissionNMPB08.h.

Enumeration Type Documentation

enum RoadFlowType

Flow type.

Enumerator:

FlowType_CONST Constant flow.

FlowType_ACC Acceleration section.

FlowType_DEC Deceleration section.
FlowType_START Startup section.
FlowType_STOP Stopping section.
FlowType_CONST Constant flow.
FlowType_ACC Acceleration section.
FlowType_DEC Deceleration section.
FlowType_START Startup section.
FlowType_STOP Stopping section.
FlowType_CONST Constant flow.
FlowType_ACC Acceleration section.
FlowType_DEC Deceleration section.
FlowType_START Startup section.
FlowType_STOP Stopping section.

Definition at line 62 of file RoadEmissionNMPB08.h.

enum RoadSpectrumType

Spectrum type.

Enumerator:

Spectrum_dBA The global value in dB(A) : only one value.
Spectrum_oct_A Octave bands (dB(A)) : 6 frequencies from 125 to 4000.
Spectrum_3oct_A Third-octave bands (dB(A)) : 18 frequencies from 100 to 5000.
Spectrum_oct_lin Octave bands : 6 frequencies from 125 to 4000.
Spectrum_3oct_lin Third-octave bands : 18 frequencies from 100 to 5000.
Spectrum_dBA The global value in dB(A) : only one value.
Spectrum_oct_A Octave bands (dB(A)) : 6 frequencies from 125 to 4000.
Spectrum_3oct_A Third-octave bands (dB(A)) : 18 frequencies from 100 to 5000.
Spectrum_oct_lin Octave bands : 6 frequencies from 125 to 4000.
Spectrum_3oct_lin Third-octave bands : 18 frequencies from 100 to 5000.
Spectrum_dBA The global value in dB(A) : only one value.
Spectrum_oct_A Octave bands (dB(A)) : 6 frequencies from 125 to 4000.
Spectrum_3oct_A Third-octave bands (dB(A)) : 18 frequencies from 100 to 5000.
Spectrum_oct_lin Octave bands : 6 frequencies from 125 to 4000.
Spectrum_3oct_lin Third-octave bands : 18 frequencies from 100 to 5000.

Definition at line 187 of file RoadEmissionNMPB08.h.

enum RoadSurfaceType

Road surface type.

Enumerator:

RoadSurface_Default default type

RoadSurface_R1 R1 type.
RoadSurface_R2 R2 type.
RoadSurface_R3 R3 type.
RoadSurface_DR1 Draining R1 type.
RoadSurface_DR2 Draining R2 type.
RoadSurface_DR3 Draining R3 type.
RoadSurface_UserDefined User defined type.
RoadSurface_Default default type
RoadSurface_R1 R1 type.
RoadSurface_R2 R2 type.
RoadSurface_R3 R3 type.
RoadSurface_DR1 Draining R1 type.
RoadSurface_DR2 Draining R2 type.
RoadSurface_DR3 Draining R3 type.
RoadSurface_UserDefined User defined type.
RoadSurface_Default default type
RoadSurface_R1 R1 type.
RoadSurface_R2 R2 type.
RoadSurface_R3 R3 type.
RoadSurface_DR1 Draining R1 type.
RoadSurface_DR2 Draining R2 type.
RoadSurface_DR3 Draining R3 type.
RoadSurface_UserDefined User defined type.

Definition at line 89 of file RoadEmissionNMPB08.h.

enum RoadVehicleType

Vehicle type.
Enumerator:
VehicleType_VL Light vehicle.
VehicleType_PL Heavy goods vehicle.
VehicleType_VL Light vehicle.
VehicleType_PL Heavy goods vehicle.
VehicleType_VL Light vehicle.
VehicleType_PL Heavy goods vehicle.

Definition at line 47 of file RoadEmissionNMPB08.h.

Function Documentation

_COMPILE_NMPB bool NMPB08_DefineRoadSurfaceType (int*idRoadSurface*, double*AVL*, double*BVL*, double*Vref_VL*, double*APL*, double*BPL*, double*Vref_PL*, double const **spectrum*)

Definition of a new road surface type with data needed for rolling component and for spectrum.

Parameters:

idRoadSurface	The road surface id
AVL	The AVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
BVL	The BVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
Vref_VL	The Vref_VL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
APL	The APL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
BPL	The BPL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
Vref_PL	The Vref_PL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
spectrum	The spectral distribution for this road surface (if null, non draining default values will be used)

Returns:

true if all OK

Definition at line 172 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lw_rolling (RoadVehicleType*type*, double*vehicleSpeed*, RoadSurfaceType*surfaceType*, double*surfaceAge*, RoadSpectrumType*specType*)

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed and road surface characteristics.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
surfaceType	The surface type
surfaceAge	The surface age
specType	The spectrum type

Returns:

the rolling component of the Sound power level of the vehicle, in third octave bands

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed and road surface characteristics.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
surfaceType	The surface type
surfaceAge	The surface age
specType	The spectrum type

Returns:

the rolling component of the Sound power level per source line meter, for each third-octave band

Definition at line 100 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lw_traction (RoadVehicleType *type*, double *vehicleSpeed*, RoadFlowType *flowType*, double *ramp*, RoadSpectrumType *specType*)

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed, ramp and flow type.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
flowType	The flow type
ramp	The ramp
specType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed, ramp and flow type.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
flowType	The flow type
ramp	The ramp
specType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Definition at line 133 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lwm (RoadTraffic **roadTraffic*, RoadSpectrumType *spectrumType*)

Calculation of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

The Sound power level per source line meter, for each third-octave band

Definition at line 25 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lwm_rolling (RoadTraffic **roadTraffic*, RoadSpectrumType *spectrumType*)

Calculation of the rolling component of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

The rolling component of the Sound power level per source line meter, for each third-octave band
Definition at line 48 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lwm_traction (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)

Calculation of the traction component of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band
Definition at line 71 of file RoadEmissionNMPB08.cpp.

Variable Documentation

const int BBDR_0_10 = RoadSurface_DR1

BBDr 0/10 road surface type.
Definition at line 133 of file RoadEmissionNMPB08.h.

const int BBSG_0_10 = RoadSurface_R2

BBSG 0/10 road surface type.
Definition at line 149 of file RoadEmissionNMPB08.h.

const int BBSG_0_14 = RoadSurface_R3

BBSG 0/14 road surface type.
Definition at line 165 of file RoadEmissionNMPB08.h.

const int BBTM_0_10_type1 = RoadSurface_R2

BBTM 0/10-type1 road surface type.
Definition at line 153 of file RoadEmissionNMPB08.h.

const int BBTM_0_10_type2 = RoadSurface_R1

BBTM 0/10-type2 road surface type.
Definition at line 145 of file RoadEmissionNMPB08.h.

const int BBTM_0_14 = RoadSurface_R3

BBTM 0/14 road surface type.
Definition at line 169 of file RoadEmissionNMPB08.h.

const int BBTM_0_6_type1 = RoadSurface_R1

BBTM 0/6-type1 road surface type.
Definition at line 141 of file RoadEmissionNMPB08.h.

const int BBTM_0_6_type2 = RoadSurface_R1

BBTM 0/6-type2 road surface type.
Definition at line 137 of file RoadEmissionNMPB08.h.

const int BBUM_0_10 = RoadSurface_R2

BBUM 0/10 road surface type.
Definition at line 157 of file RoadEmissionNMPB08.h.

const int BBUM_0_6 = RoadSurface_R1

BBUM 0/6 road surface type.
Definition at line 129 of file RoadEmissionNMPB08.h.

const int BC = RoadSurface_R3

BC road surface type.
Definition at line 177 of file RoadEmissionNMPB08.h.

const int ECF = RoadSurface_R2

ECF road surface type.
Definition at line 161 of file RoadEmissionNMPB08.h.

const int ES_10_14 = RoadSurface_R3

ES 10/14 road surface type.
Definition at line 181 of file RoadEmissionNMPB08.h.

const int ES_6_10 = RoadSurface_R3

ES 6/10 road surface type.
Definition at line 173 of file RoadEmissionNMPB08.h.

RoadEmissionNMPB08.h File Reference

Classes

- struct RoadTrafficComponent
- *Road traffic component.* struct RoadTraffic

Road traffic description. Defines

- #define _COMPILE_NMPB extern "C"

Enumerations

- enum RoadVehicleType { VehicleType_VL = 1, VehicleType_PL = 2, VehicleType_VL = 1, VehicleType_PL = 2, VehicleType_VL = 1, VehicleType_PL = 2 }
- *Vehicle type.* enum RoadFlowType { FlowType_CONST = 0, FlowType_ACC = 1, FlowType_DEC = 2, FlowType_START = 3, FlowType_STOP = 4, FlowType_CONST = 0, FlowType_ACC = 1, FlowType_DEC = 2, FlowType_START = 3, FlowType_STOP = 4, FlowType_CONST = 0, FlowType_ACC = 1, FlowType_DEC = 2, FlowType_START = 3, FlowType_STOP = 4 }
- *Flow type.* enum RoadSurfaceType { RoadSurface_Default = 0, RoadSurface_R1 = 1, RoadSurface_R2 = 2, RoadSurface_R3 = 3, RoadSurface_DR1 = 4, RoadSurface_DR2 = 5, RoadSurface_DR3 = 6, RoadSurface_UserDefined = 7, RoadSurface_Default = 0, RoadSurface_R1 = 1, RoadSurface_R2 = 2, RoadSurface_R3 = 3, RoadSurface_DR1 = 4, RoadSurface_DR2 = 5, RoadSurface_DR3 = 6, RoadSurface_UserDefined = 7, RoadSurface_Default = 0, RoadSurface_R1 = 1, RoadSurface_R2 = 2, RoadSurface_R3 = 3, RoadSurface_DR1 = 4, RoadSurface_DR2 = 5, RoadSurface_DR3 = 6, RoadSurface_UserDefined = 7 }
- *Road surface type.* enum RoadSpectrumType { Spectrum_dBA = 0, Spectrum_oct_A = 1, Spectrum_3oct_A = 2, Spectrum_oct_lin = 3, Spectrum_3oct_lin = 4, Spectrum_dBA = 0, Spectrum_oct_A = 1, Spectrum_3oct_A = 2, Spectrum_oct_lin = 3, Spectrum_3oct_lin = 4, Spectrum_dBA = 0, Spectrum_oct_A = 1, Spectrum_3oct_A = 2, Spectrum_oct_lin = 3, Spectrum_3oct_lin = 4 }

Spectrum type. Functions

- _COMPILE_NMPB double * NMPB08_Lwm (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the Sound power level per source line meter.
- _COMPILE_NMPB double * NMPB08_Lwm_rolling (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the rolling component of the Sound power level per source line meter.
- _COMPILE_NMPB double * NMPB08_Lwm_traction (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the traction component of the Sound power level per source line meter.
- _COMPILE_NMPB double * NMPB08_Lw_rolling (RoadVehicleType type, double vehicleSpeed, RoadSurfaceType surfaceType, double surfaceAge, RoadSpectrumType specType)
  Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed and road surface characteristics.
- _COMPILE_NMPB double * NMPB08_Lw_traction (RoadVehicleType type, double vehicleSpeed, RoadFlowType flowType, double ramp, RoadSpectrumType specType)
  Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed, ramp and flow type.
- _COMPILE_NMPB bool NMPB08_DefineRoadSurfaceType (int idRoadSurface, double AVL, double BVL, double Vref_VL, double APL, double BPL, double Vref_PL, double const *spectrum)
  Definition of a new road surface type with data needed for rolling component and for spectrum.

Variables

- const int BBUM_0_6 = RoadSurface_R1
  BBUM 0/6 road surface type.
- const int BBDR_0_10 = RoadSurface_DR1
  BBDr 0/10 road surface type.
- const int BBTM_0_6_type2 = RoadSurface_R1
  BBTM 0/6-type2 road surface type.
- const int BBTM_0_6_type1 = RoadSurface_R1
  BBTM 0/6-type1 road surface type.
- const int BBTM_0_10_type2 = RoadSurface_R1
  BBTM 0/10-type2 road surface type.
- const int BBSG_0_10 = RoadSurface_R2
  BBSG 0/10 road surface type.
- const int BBTM_0_10_type1 = RoadSurface_R2
  BBTM 0/10-type1 road surface type.
- const int BBUM_0_10 = RoadSurface_R2
  BBUM 0/10 road surface type.
- const int ECF = RoadSurface_R2
  ECF road surface type.
- const int BBSG_0_14 = RoadSurface_R3
  BBSG 0/14 road surface type.
- const int BBTM_0_14 = RoadSurface_R3
  BBTM 0/14 road surface type.
- const int ES_6_10 = RoadSurface_R3
  ES 6/10 road surface type.
- const int BC = RoadSurface_R3
  BC road surface type.
- const int ES_10_14 = RoadSurface_R3
  ES 10/14 road surface type.

Define Documentation

#define _COMPILE_NMPB extern "C"

Definition at line 38 of file RoadEmissionNMPB08.h.

Enumeration Type Documentation

enum RoadFlowType

Flow type.

Enumerator:

FlowType_CONST Constant flow.

FlowType_ACC Acceleration section.

FlowType_DEC Deceleration section.
FlowType_START Startup section.
FlowType_STOP Stopping section.
FlowType_CONST Constant flow.
FlowType_ACC Acceleration section.
FlowType_DEC Deceleration section.
FlowType_START Startup section.
FlowType_STOP Stopping section.
FlowType_CONST Constant flow.
FlowType_ACC Acceleration section.
FlowType_DEC Deceleration section.
FlowType_START Startup section.
FlowType_STOP Stopping section.

Definition at line 62 of file RoadEmissionNMPB08.h.

enum RoadSpectrumType

Spectrum type.

Enumerator:

Spectrum_dBA The global value in dB(A) : only one value.
Spectrum_oct_A Octave bands (dB(A)) : 6 frequencies from 125 to 4000.
Spectrum_3oct_A Third-octave bands (dB(A)) : 18 frequencies from 100 to 5000.
Spectrum_oct_lin Octave bands : 6 frequencies from 125 to 4000.
Spectrum_3oct_lin Third-octave bands : 18 frequencies from 100 to 5000.
Spectrum_dBA The global value in dB(A) : only one value.
Spectrum_oct_A Octave bands (dB(A)) : 6 frequencies from 125 to 4000.
Spectrum_3oct_A Third-octave bands (dB(A)) : 18 frequencies from 100 to 5000.
Spectrum_oct_lin Octave bands : 6 frequencies from 125 to 4000.
Spectrum_3oct_lin Third-octave bands : 18 frequencies from 100 to 5000.
Spectrum_dBA The global value in dB(A) : only one value.
Spectrum_oct_A Octave bands (dB(A)) : 6 frequencies from 125 to 4000.
Spectrum_3oct_A Third-octave bands (dB(A)) : 18 frequencies from 100 to 5000.
Spectrum_oct_lin Octave bands : 6 frequencies from 125 to 4000.
Spectrum_3oct_lin Third-octave bands : 18 frequencies from 100 to 5000.

Definition at line 187 of file RoadEmissionNMPB08.h.

enum RoadSurfaceType

Road surface type.

Enumerator:

RoadSurface_Default default type

RoadSurface_R1 R1 type.
RoadSurface_R2 R2 type.
RoadSurface_R3 R3 type.
RoadSurface_DR1 Draining R1 type.
RoadSurface_DR2 Draining R2 type.
RoadSurface_DR3 Draining R3 type.
RoadSurface_UserDefined User defined type.
RoadSurface_Default default type
RoadSurface_R1 R1 type.
RoadSurface_R2 R2 type.
RoadSurface_R3 R3 type.
RoadSurface_DR1 Draining R1 type.
RoadSurface_DR2 Draining R2 type.
RoadSurface_DR3 Draining R3 type.
RoadSurface_UserDefined User defined type.
RoadSurface_Default default type
RoadSurface_R1 R1 type.
RoadSurface_R2 R2 type.
RoadSurface_R3 R3 type.
RoadSurface_DR1 Draining R1 type.
RoadSurface_DR2 Draining R2 type.
RoadSurface_DR3 Draining R3 type.
RoadSurface_UserDefined User defined type.

Definition at line 89 of file RoadEmissionNMPB08.h.

enum RoadVehicleType

Vehicle type.
Enumerator:
VehicleType_VL Light vehicle.
VehicleType_PL Heavy goods vehicle.
VehicleType_VL Light vehicle.
VehicleType_PL Heavy goods vehicle.
VehicleType_VL Light vehicle.
VehicleType_PL Heavy goods vehicle.

Definition at line 47 of file RoadEmissionNMPB08.h.

Function Documentation

_COMPILE_NMPB bool NMPB08_DefineRoadSurfaceType (int*idRoadSurface*, double*AVL*, double*BVL*, double*Vref_VL*, double*APL*, double*BPL*, double*Vref_PL*, double const **spectrum*)

Definition of a new road surface type with data needed for rolling component and for spectrum.

Parameters:

idRoadSurface	The road surface id
AVL	The AVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
BVL	The BVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
Vref_VL	The Vref_VL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
APL	The APL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
BPL	The BPL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
Vref_PL	The Vref_PL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
spectrum	The spectral distribution for this road surface (if null, non draining default values will be used)

Returns:

true if all OK

Definition at line 172 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lw_rolling (RoadVehicleType*type*, double*vehicleSpeed*, RoadSurfaceType*surfaceType*, double*surfaceAge*, RoadSpectrumType*specType*)

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed and road surface characteristics.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
surfaceType	The surface type
surfaceAge	The surface age
specType	The spectrum type

Returns:

the rolling component of the Sound power level of the vehicle, in third octave bands

Note:

this function returns a pointer to an internal memory buffer maintained within the library and shared by calls to NMPB08_LW_traction and NMPB08_Lw_traction. Any call to these functions will invalidate the results obtained by previous calls. It is highly recommended that applications take a copy of the return values into dedicated memory immediately after the call to any of these functions.

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed and road surface characteristics.

Parameters:

type	The vehicle type

vehicleSpeed	The vehicle speed
surfaceType	The surface type
surfaceAge	The surface age
specType	The spectrum type

Returns:

the rolling component of the Sound power level per source line meter, for each third-octave band

Definition at line 100 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lw_traction (RoadVehicleType*type*, double*vehicleSpeed*, RoadFlowType*flowType*, double*ramp*, RoadSpectrumType*specType*)

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed, ramp and flow type.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
flowType	The flow type
ramp	The ramp
specType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Note:

this function returns a pointer to an internal memory buffer maintained within the library and shared by calls to NMPB08_LW_traction and NMPB08_Lw_traction. Any call to these functions will invalidate the results obtained by previous calls. It is highly recommended that applications take a copy of the return values into dedicated memory immediately after the call to any of these functions.

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed, ramp and flow type.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
flowType	The flow type
ramp	The ramp
specType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Definition at line 133 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lwm (RoadTraffic **roadTraffic*, RoadSpectrumType*spectrumType*)

Calculation of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

The Sound power level per source line meter, for each third-octave band

Note:

this function returns a pointer to an internal memory buffer maintained within the library and shared by calls to NMPB08_Lwm_traction, NMPB08_Lwm_traction and NMPB08_Lwm. Any call to these functions will invalidate the results obtained by previous calls. It is highly recommended that applications take a copy of the return values into dedicated memory immediately after the call to any of these functions.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

The Sound power level per source line meter, for each third-octave band

Definition at line 25 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lwm_rolling (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)

Calculation of the rolling component of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

The rolling component of the Sound power level per source line meter, for each third-octave band

Note:

this function returns a pointer to an internal memory buffer maintained within the library and shared by calls to NMPB08_Lwm_traction, NMPB08_Lwm_traction and NMPB08_Lwm. Any call to these functions will invalidate the results obtained by previous calls. It is highly recommended that applications take a copy of the return values into dedicated memory immediately after the call to any of these functions.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

The rolling component of the Sound power level per source line meter, for each third-octave band

Definition at line 48 of file RoadEmissionNMPB08.cpp.

_COMPILE_NMPB double* NMPB08_Lwm_traction (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)

Calculation of the traction component of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Note:

this function returns a pointer to an internal memory buffer maintained within the library and shared by calls to NMPB08_Lwm_traction, NMPB08_Lwm_traction and NMPB08_Lwm. Any call to these functions will

invalidate the results obtained by previous calls. It is highly recommended that applications take a copy of the return values into dedicated memory immediately after the call to any of these functions.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Definition at line 71 of file RoadEmissionNMPB08.cpp.

Variable Documentation

const int BBDR_0_10 = RoadSurface_DR1

BBDr 0/10 road surface type.

Definition at line 133 of file RoadEmissionNMPB08.h.

const int BBSG_0_10 = RoadSurface_R2

BBSG 0/10 road surface type.

Definition at line 149 of file RoadEmissionNMPB08.h.

const int BBSG_0_14 = RoadSurface_R3

BBSG 0/14 road surface type.

Definition at line 165 of file RoadEmissionNMPB08.h.

const int BBTM_0_10_type1 = RoadSurface_R2

BBTM 0/10-type1 road surface type.

Definition at line 153 of file RoadEmissionNMPB08.h.

const int BBTM_0_10_type2 = RoadSurface_R1

BBTM 0/10-type2 road surface type.

Definition at line 145 of file RoadEmissionNMPB08.h.

const int BBTM_0_14 = RoadSurface_R3

BBTM 0/14 road surface type.

Definition at line 169 of file RoadEmissionNMPB08.h.

const int BBTM_0_6_type1 = RoadSurface_R1

BBTM 0/6-type1 road surface type.

Definition at line 141 of file RoadEmissionNMPB08.h.

const int BBTM_0_6_type2 = RoadSurface_R1

BBTM 0/6-type2 road surface type.
Definition at line 137 of file RoadEmissionNMPB08.h.

const int BBUM_0_10 = RoadSurface_R2

BBUM 0/10 road surface type.
Definition at line 157 of file RoadEmissionNMPB08.h.

const int BBUM_0_6 = RoadSurface_R1

BBUM 0/6 road surface type.
Definition at line 129 of file RoadEmissionNMPB08.h.

const int BC = RoadSurface_R3

BC road surface type.
Definition at line 177 of file RoadEmissionNMPB08.h.

const int ECF = RoadSurface_R2

ECF road surface type.
Definition at line 161 of file RoadEmissionNMPB08.h.

const int ES_10_14 = RoadSurface_R3

ES 10/14 road surface type.
Definition at line 181 of file RoadEmissionNMPB08.h.

const int ES_6_10 = RoadSurface_R3

ES 6/10 road surface type.
Definition at line 173 of file RoadEmissionNMPB08.h.

RoadEmissionNMPB.cpp File Reference

```
#include "stdafx.h"
```

Functions

- BOOL APIENTRY DllMain (HMODULE hModule, DWORD ul_reason_for_call, LPVOID lpReserved)

Function Documentation

BOOL APIENTRY DllMain (HMODULE *hModule*, DWORD *ul_reason_for_call*, LPVOID *lpReserved*)

Definition at line 11 of file RoadEmissionNMPB.cpp.

RoadEmissionNMPB08.cpp File Reference

Definition of the dll functions that can be called by external software.

```
#include "RoadEmissionNMPB08.h"
#include "CalculRoadEmission.h"
#include <vector>
#include <math.h>
#include <stdio.h>
```

Functions

- double * NMPB08_Lwm (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the Sound power level per source line meter.
- double * NMPB08_Lwm_rolling (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the rolling component of the Sound power level per source line meter.
- double * NMPB08_Lwm_traction (RoadTraffic *roadTraffic, RoadSpectrumType spectrumType)
  Calculation of the traction component of the Sound power level per source line meter.
- double * NMPB08_Lw_rolling (RoadVehicleType type, double vehicleSpeed, RoadSurfaceType surfaceType, double surfaceAge, RoadSpectrumType specType)
  Calculation of the rolling component of the Sound power level per source line meter for a vehicle type.
- double * NMPB08_Lw_traction (RoadVehicleType type, double vehicleSpeed, RoadFlowType flowType, double ramp, RoadSpectrumType specType)
  Calculation of the traction component of the Sound power level per source line meter for a vehicle type.
- bool NMPB08_DefineRoadSurfaceType (int idRoadSurface, double AVL, double BVL, double Vref_VL, double APL, double BPL, double Vref_PL, double const *spectrum)
  Definition of a new road surface type with data needed for rolling component and for spectrum.

Detailed Description

Definition of the dll functions that can be called by external software.

Author:
CSTB

Version:
1.0

Definition in file RoadEmissionNMPB08.cpp.

Function Documentation

bool NMPB08_DefineRoadSurfaceType (int *idRoadSurface*, double *AVL*, double *BVL*, double *Vref_VL*, double *APL*, double *BPL*, double *Vref_PL*, double const **spectrum*)

Definition of a new road surface type with data needed for rolling component and for spectrum.

Parameters:

idRoadSurface	The road surface id

AVL	The AVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
BVL	The BVL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
Vref_VL	The Vref_VL value for the Lr_w_VL calculation where Lr_w_VL = BVL + AVL * log10(V/Vref_VL)
APL	The APL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
BPL	The BPL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
Vref_PL	The Vref_PL value for the Lr_w_PL calculation where Lr_w_PL = BPL + APL * log10(V/Vref_PL)
spectrum	The spectral distribution for this road surface (if null, non draining default values will be used)

Returns:

true if all OK

Definition at line 172 of file RoadEmissionNMPB08.cpp.

double* NMPB08_Lw_rolling (RoadVehicleType*type*, double*vehicleSpeed*, RoadSurfaceType*surfaceType*, double*surfaceAge*, RoadSpectrumType*specType*)

Calculation of the rolling component of the Sound power level per source line meter for a vehicle type.

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed and road surface characteristics.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
surfaceType	The surface type
surfaceAge	The surface age
specType	The spectrum type

Returns:

the rolling component of the Sound power level per source line meter, for each third-octave band

Definition at line 100 of file RoadEmissionNMPB08.cpp.

double* NMPB08_Lw_traction (RoadVehicleType*type*, double*vehicleSpeed*, RoadFlowType*flowType*, double*ramp*, RoadSpectrumType*specType*)

Calculation of the traction component of the Sound power level per source line meter for a vehicle type.

Calculation of the rolling component of the Sound power level of a vehicle type as a function of speed, ramp and flow type.

Parameters:

type	The vehicle type
vehicleSpeed	The vehicle speed
flowType	The flow type
ramp	The ramp
specType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Definition at line 133 of file RoadEmissionNMPB08.cpp.

double* NMPB08_Lwm (RoadTraffic *roadTraffic*, RoadSpectrumType*spectrumType*)

Calculation of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

The Sound power level per source line meter, for each third-octave band

Definition at line 25 of file RoadEmissionNMPB08.cpp.

double* NMPB08_Lwm_rolling (RoadTraffic *roadTraffic*, RoadSpectrumType*spectrumType*)

Calculation of the rolling component of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

The rolling component of the Sound power level per source line meter, for each third-octave band

Definition at line 48 of file RoadEmissionNMPB08.cpp.

double* NMPB08_Lwm_traction (RoadTraffic *roadTraffic*, RoadSpectrumType*spectrumType*)

Calculation of the traction component of the Sound power level per source line meter.

Parameters:

roadTraffic	The road traffic data
spectrumType	The spectrum type

Returns:

the traction component of the Sound power level per source line meter, for each third-octave band

Definition at line 71 of file RoadEmissionNMPB08.cpp.

RoadSurfaceDescription.cpp File Reference

RoadSurfaceDescription Class used to define a road surface.

```
#include "RoadSurfaceDescription.h"
#include <math.h>
```

Namespaces

- namespace RoadSurfaceDescriptionNMPB

Detailed Description

RoadSurfaceDescription Class used to define a road surface.

Author:

CSTB

Version:

1.0

Definition in file RoadSurfaceDescription.cpp.

RoadSurfaceDescription.h File Reference

RoadSurfaceDescription Class used to define a road surface.

```
#include <vector>
#include <map>
```

Classes

- class RoadSurfaceDescriptionNMPB::RoadSurfaceDescription

Class used to define a road surface. Namespaces

- namespace RoadSurfaceDescriptionNMPB

Detailed Description

RoadSurfaceDescription Class used to define a road surface.

Author:

CSTB

Version:

1.0

Definition in file RoadSurfaceDescription.h.

stdafx.cpp File Reference

```
#include "stdafx.h"
```

stdafx.h File Reference

`#include <windows.h>`

Defines

- #define WINVER 0x0501
- #define _WIN32_WINNT 0x0501
- #define _WIN32_WINDOWS 0x0410
- #define _WIN32_IE 0x0600
- #define WIN32_LEAN_AND_MEAN

Define Documentation

#define _WIN32_IE 0x0600

Definition at line 23 of file stdafx.h.

#define _WIN32_WINDOWS 0x0410

Definition at line 19 of file stdafx.h.

#define _WIN32_WINNT 0x0501

Definition at line 15 of file stdafx.h.

#define WIN32_LEAN_AND_MEAN

Definition at line 26 of file stdafx.h.

#define WINVER 0x0501

Definition at line 11 of file stdafx.h.

Index

INDEX