Acceleration Conversions in DLL file

1 Introduction

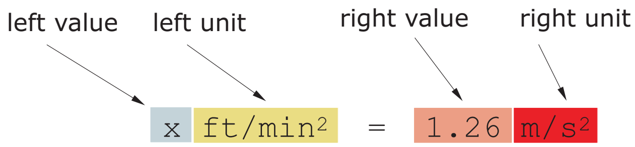
The conversion DLL file of unit Acceleration is written in C/C++ programming language style, and do not require any extra code in using this DLL file. This DLL file includes 2 functions to handle the Acceleration conversions as follow:

```
double LP_UnitConversionsAcceleration_GetLeftValue (char* LeftUnit, char* RightUnit, double RightValue)  ;
double LP_UnitConversionsAcceleration_GetRightValue(char* LeftUnit, char* RightUnit, double LeftValue )  ;
```

2 Problems in Acceleration conversion

The two functions in DLL file are used to handle all Acceleration conversions in two problems.

Problem 1 The unknown value is on the **left hand side** of equation
This problem in conversions is described in the figure:



The value x is obtained by either one of two methods:

- Method A : The code is :

  ```
  double x ;
  x = LP_UnitConversionsAcceleration_GetLeftValue("FootPerMinuteSquared", "MeterPerSecondSquared", 1.26);
  ```

- Method B : The code is :

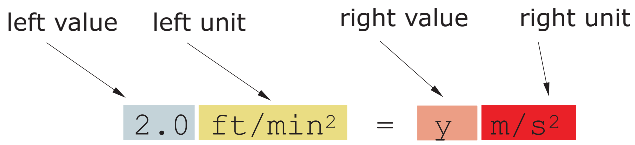
  ```
  char LeftUnit[]   = "FootPerMinuteSquared"                  ;
  char RightUnit[]  = "MeterPerSecondSquared"            ;

  double RightValue = 1.26              ;

  double x = LP_UnitConversionsAcceleration_GetLeftValue(LeftUnit, RightUnit, RightValue)   ;
  ```

Problem 2 The unknown value is on the **right hand side** of equation

This problem in conversions is described in the figure:



The value y is obtained by either one of two methods:

- Method A : The code is :

```
double y ;
y = LP_UnitConversionsAcceleration_GetRightValue("FootPerMinuteSquared", "MeterPerSecondSquared", 2.0);
```

- Method B : The code is :

```
char LeftUnit[]  = "FootPerMinuteSquared"            ;
char RightUnit[]   = "MeterPerSecondSquared"      ;

double LeftValue = 2.0                ;

double y = LP_UnitConversionsAcceleration_GetRightValue(LeftUnit, RightUnit, LeftValue)   ;
```

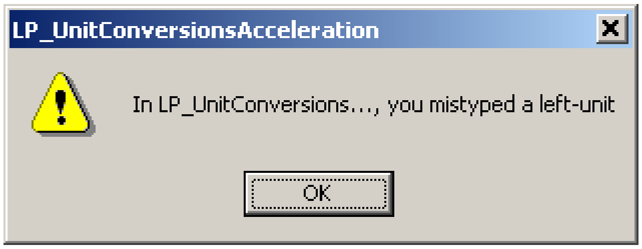
(...continue in next page)

3 Unit names in Acceleration conversions

You can choose the unit name (case sensitive) in the following table for parameters, LeftUnit and/or RightUnit

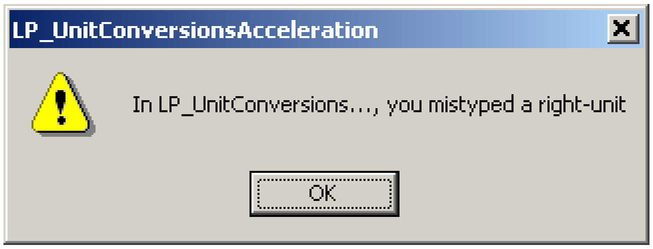
MillimeterPerSecondSquared	InchPerSecondSquared
MillimeterPerMinuteSquared	InchPerMinuteSquared
MillimeterPerHourSquared	InchPerHourSquared
CentimeterPerSecondSquared	FootPerSecondSquared
CentimeterPerMinuteSquared	FootPerMinuteSquared
CentimeterPerHourSquared	FootPerHourSquared
MeterPerSecondSquared	YardPerSecondSquared
MeterPerMinuteSquared	YardPerMinuteSquared
MeterPerHourSquared	YardPerHourSquared
KilometerPerSecondSquared	MilePerSecondSquared
KilometerPerHourSquared	MilePerMinuteSquared
Gal	MilePerHourSquared

When your unit name is not in this table, the returns of functions are -1 and the error message will issue as shown in the following figures:



In LP_UnitConversions..., you mistyped a left-unit

or



In LP_UnitConversions..., you mistyped a right-unit

4 Tip

1. The parameters in two functions have the same order of unit name (LeftUnit is first and RightUnit is second) and the last parameter is the known value.

2. Use function LP_UnitConversionsAcceleration_GetLeftValue(..) if your **unknown** value is on the **left hand side** of the equation.

3. Use function LP_UnitConversionsAcceleration_GetRightValue(..) if your **unknown** value is on the **right hand side** of the equation.