

Thermal Conductivity Conversions in DLL file

1 Introduction

The conversion DLL file of unit Thermal Conductivity 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 Thermal Conductivity conversions as follow:

```
double LP_UnitConversionsThermalConductivity_GetLeftValue (char* LeftUnit, char* RightUnit, double RightValue) ;
double LP_UnitConversionsThermalConductivity_GetRightValue(char* LeftUnit, char* RightUnit, double LeftValue ) ;
```

2 Problems in Thermal Conductivity conversion

The two functions in DLL file are used to handle all Thermal Conductivity 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:

$$\begin{array}{ccc} \text{left value} & \text{left unit} & \\ \swarrow & \searrow & \\ x & \text{Btu / (h.ft.}^\circ\text{F)} & = \\ \swarrow & \searrow & \\ 0.18 & \text{kcal / (min.m.}^\circ\text{C)} & \end{array}$$

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

- Method A : The code is :

```
double x ;
x = LP_UnitConversionsThermalConductivity_GetLeftValue( \
    "BtuPerHourFootFahrenheit", "KilocaloriePerMinuteMeterCelsius", 0.18);
```

- Method B : The code is :

```
char LeftUnit[] = "BtuPerHourFootFahrenheit" ;
char RightUnit[] = "KilocaloriePerMinuteMeterCelsius" ;

double RightValue = 0.18 ;

double x = LP_UnitConversionsThermalConductivity_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:

$$\begin{array}{ccc}
 \text{left value} & \text{left unit} & \\
 \swarrow & \searrow & \\
 2.0 & \text{Btu / (h.ft.}^\circ\text{F)} & = \text{y kcal / (min.m.}^\circ\text{C)} \\
 \swarrow & \searrow & \swarrow \quad \searrow \\
 & & \text{right value} \quad \text{right unit}
 \end{array}$$

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

- Method A : The code is :

```
double y ;
y = LP_UnitConversionsThermalConductivity_GetRightValue(
    "BtuPerHourFootFahrenheit", "KilocaloriePerMinuteMeterCelsius", 2.0);
```

- Method B : The code is :

```
char LeftUnit[] = "BtuPerHourFootFahrenheit" ;
char RightUnit[] = "KilocaloriePerMinuteMeterCelsius" ;

double LeftValue = 2.0 ;

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

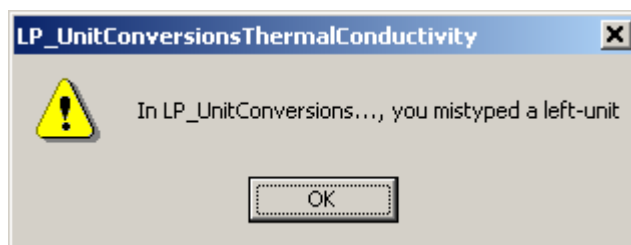
(...continue in next page)

3 Unit names in Thermal Conductivity conversions

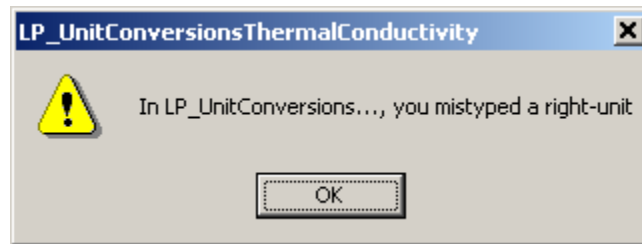
You can choose the unit name (case sensitive) in the following table for parameters, LeftUnit and/or RightUnit

BtuPerSecondInchFahrenheit	WattPerMillimeterCelsius
BtuPerSecondFootFahrenheit	WattPerCentimeterCelsius
BtuPerHourInchFahrenheit	WattPerMeterCelsius
BtuPerHourFootFahrenheit	KilowattPerMeterCelsius
BtuPerSecondInchRankine	MegawattPerMeterCelsius
BtuPerSecondFootRankine	KilojoulePerHourMeterCelsius
BtuPerHourInchRankine	MegajoulePerHourMeterCelsius
BtuPerHourFootRankine	KilojoulePerHourMeterKelvin
CaloriePerSecondMillimeterCelsius	MegajoulePerHourMeterKelvin
CaloriePerSecondCentimeterCelsius	WattPerMillimeterKelvin
CaloriePerSecondMeterCelsius	WattPerCentimeterKelvin
KilocaloriePerSecondMeterCelsius	WattPerMeterKelvin
KilocaloriePerMinuteMeterCelsius	KilowattPerMeterKelvin
KilocaloriePerHourMeterCelsius	MegawattPerMeterKelvin

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:



or



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_UnitConversionsThermalConductivity_GetLeftValue(..)` if your **unknown** value is on the **left hand side** of the equation.
3. Use function `LP_UnitConversionsThermalConductivity_GetRightValue(..)` if your **unknown** value is on the **right hand side** of the equation.