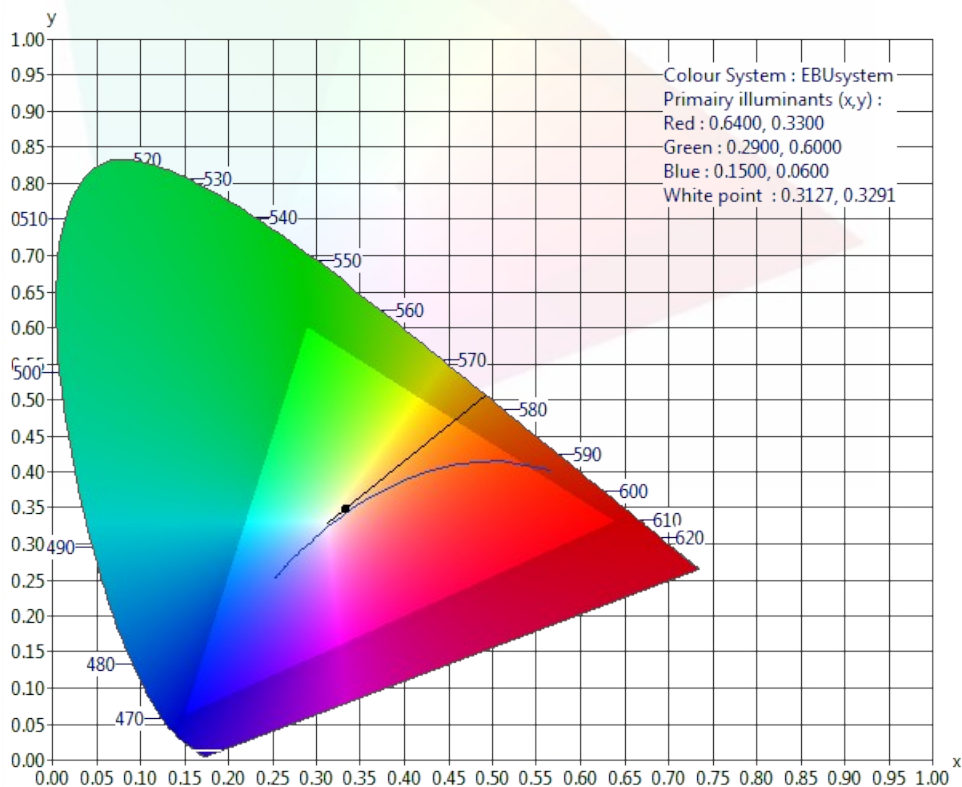




admesy

ADVANCED MEASUREMENT SYSTEMS

Admesy colour library manual





All rights reserved. No part of this document may be reproduced, stored in a database or retrieval system, or published in any form or way, electronically, mechanically, by print, photo print, microfilm or any other means without prior written permission from the publisher.

All correspondence regarding copyrights :

Admesy B.V.

Beneluxstraat 7

6014 CC Ittervoort

The Netherlands

Tel : +31 (0)475 600232

Fax : +31 (0) 45 5213156

URL : <http://www.admesy.nl>

Contents

1	General information	4
2	Colour conversions	5
2.1	CIE 1931 XYZ to CIE 1931 Yxy	5
2.2	CIE 1931 Yxy to CIE 1931 XYZ	5
2.3	CIE 1931 XYZ array to CIE 1931 Yxy array	6
2.4	CIE 1931 Yxy array to CIE 1931 XYZ array	6
2.5	CIE 1976 L*a*b* to CIE 1931 XYZ	7
2.6	CIE 1976 L*a*b* array to CIE 1931 XYZ array	7
2.7	CIE 1931 XYZ to CIE 1976 L*a*b*	8
2.8	CIE 1931 XYZ to Hunter Lab	9
2.9	Hunter Lab to CIE 1931 XYZ	10
2.10	CIE 1931 XYZ to CIE 1976 L*u*v*	11
2.11	CIE 1976 L*u*v* to CIE 1931 XYZ	12
2.12	CIE 1976 L*u*v* to CIE L*C*h*uv	13
2.13	CIE L*C*h*uv to CIE 1976 L*u*v*	13
2.14	CIE 1976 L*a*b* to CIE L*C*hab	14
2.15	CIE L*C*hab to CIE 1976 L*a*b*	14
2.16	CIE 1960 Yuv to CIE 1964 U*V*W*	15
2.17	ΔE CIE 1964 U*V*W*	15
2.18	ΔE CIE 1976	16
2.19	ΔE CIE CMC (1984)	17
2.20	ΔE CIE 1994	18
2.21	ΔE CIE 2000	19
2.22	CIE 1931 XYZ to sRGB	21
2.23	CCT to xy	21
2.24	Planckian locus to xy	22
2.25	Planckian locus to CIE 1960 uv	22
2.26	CIE 1931 XYZ to CCT	23
2.27	CIE 1931 XYZ to CCT Robertson's method	24
2.28	Chromaticity (xy) to Dominant wavelength	25
2.29	Lumen to Candela	26
2.30	Candela to Lumen	26

1 General information

All functions are exported as "C" functions. All the structs that are being used are located in the "admesy_colour.h" file, so when one is needed please refer this file.

All labview vi's functions are using the "libadmesy_colour.dll" and are tested to work well.

The functions in this document are reviewed and recalculated with different calculators, to be sure the correct implementation is used.

Within every function description the formulas are given. When there is a question or a remark please contact us.

Note: when a function has a return, this will indicate whether there was an error occurred or not. 0 = success



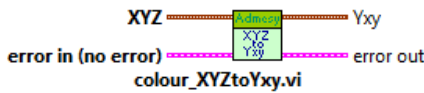
2 Colour conversions

The following commands can be used to convert the input values to a other colour space.

2.1 CIE 1931 XYZ to CIE 1931 Yxy

Program language:

Labview



C

```
int XYZtoYxy (XYZ_space_t *XYZ, Yxy_space_t *Yxy)
```

Function description:

Converts CIE XYZ colour space to CIE (1931) Yxy colour space.

Input:

```
struct XYZ_space_t;
```

Output:

```
struct Yxy_space_t;
```

Formula:

$$x = \frac{X}{X+Y+Z}$$

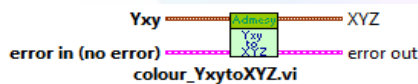
$$y = \frac{Y}{X+Y+Z}$$

$$z = \frac{Z}{X+Y+Z} = 1 - x - y$$

2.2 CIE 1931 Yxy to CIE 1931 XYZ

Program language:

Labview



C

```
int YxytoXYZ (Yxy_space_t *Yxy, XYZ_space_t *XYZ)
```

Function description:

converts CIE (1931) Yxy colour space to CIE XYZ.

Input:

```
struct Yxy_space_t;
```

Output:

```
struct XYZ_space_t;
```

Formula:

$$X = \frac{Y}{y} x$$

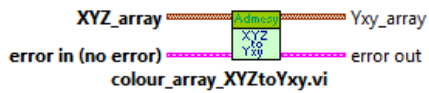
$$Y = Y$$

$$Z = \frac{Y}{y} (1 - x - y)$$

2.3 CIE 1931 XYZ array to CIE 1931 Yxy array

Program language:

Labview



C

```
int array_XYZtoYxy (XYZ_array XYZ_in, Yxy_array  
Yxy_out)
```

Function description:

Converts a CIE XYZ colour space array to CIE (1931) Yxy colour space array.

Input:

struct XYZ_array_t

Output:

struct Yxy_array_t

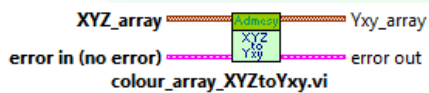
Formula:

see formula of 2.1

2.4 CIE 1931 Yxy array to CIE 1931 XYZ array

Program language:

Labview



C

```
int array_YxytoXYZ (Yxy_array Yxy_in, XYZ_array  
XYZ_out)
```

Function description:

Converts a CIE (1931) Yxy colour space array to CIE XYZ colour space array.

Input:

struct Yxy_array_t

Output:

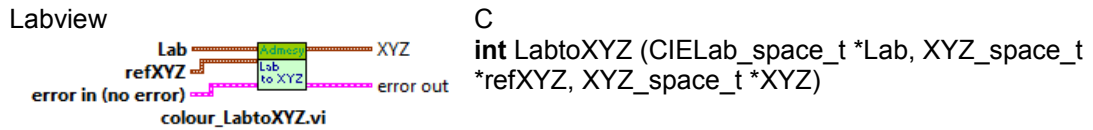
struct XYZ_array_t

Formula:

see formula of 2.2

2.5 CIE 1976 L*a*b* to CIE 1931 XYZ

Program language:



Function description:

Converts a CIE Lab colour space to CIE XYZ colour space regarding to the reference white value (d50 d65 ect..).

Input:

struct CIELab_space_t
struct XYZ_space_t (reference white)

Output:

struct XYZ_space_t

Formula:

$$e = \frac{216}{24389}, k = \frac{24389}{27}$$

$$X = x_r \text{WhiteRef}X, Y = y_r \text{WhiteRef}Y, Z = z_r \text{WhiteRef}Z$$

$$x_r = \begin{cases} f_x^3 & f_x^3 > e \\ \frac{116f_x - 16}{k} & f_x^3 \leq e \end{cases} \quad y_r = \begin{cases} \left(\frac{l+16}{116}\right)^3 & L > ke \\ \frac{L}{k} & L \leq ke \end{cases} \quad z_r = \begin{cases} f_z^3 & f_z^3 > e \\ \frac{116f_z - 16}{k} & f_z^3 \leq e \end{cases}$$

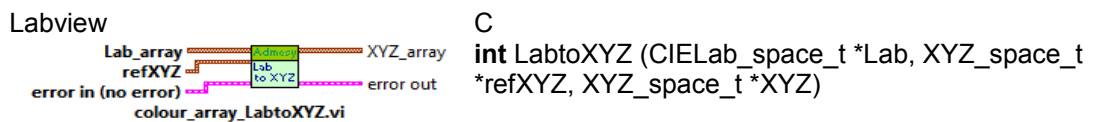
$$f_y = L + \frac{16}{116}$$

$$f_x = \frac{a}{500} + f_y$$

$$f_z = f_y - \frac{b}{200}$$

2.6 CIE 1976 L*a*b* array to CIE 1931 XYZ array

Program language:



Function description:

Converts a CIE Lab colour space array to CIE XYZ colour space array regarding to the reference white value (d50 d65 ect..).

Input:

struct CIELab_space_t
struct XYZ_space_t (reference white)

Output:

struct XYZ_space_t

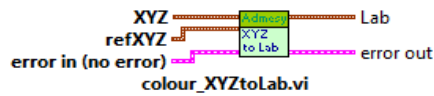
Formula:

see formula 2.5

2.7 CIE 1931 XYZ to CIE 1976 L*a*b*

Program language:

Labview



C

```
int XYZtoLab (XYZ_space_t *XYZ, XYZ_space_t
*refXYZ, CIELab_space_t *Lab)
```

Function description:

Converts a CIE XYZ colour space to CIE Lab colour space regarding to the reference white value (d50 d65 ect..).

Input:

struct XYZ_space_t
struct XYZ_space_t (reference white)

Output:

struct CIELab_space_t

Formula:

$$e = \frac{216}{24389}, k = \frac{24389}{27}$$

$$x_r = \frac{X}{WhiteRefX}, y_r = \frac{Y}{WhiteRefY}, z_r = \frac{Z}{WhiteRefZ}$$

$$f_x = \begin{cases} \sqrt[3]{x_r} & x_r > e \\ \frac{kx_r + 16}{116} & x_r \leq e \end{cases} \quad f_y = \begin{cases} \sqrt[3]{y_r} & y_r > e \\ \frac{ky_r + 16}{116} & y_r \leq e \end{cases} \quad f_z = \begin{cases} \sqrt[3]{z_r} & z_r > e \\ \frac{kz_r + 16}{116} & z_r \leq e \end{cases}$$

$$L^* = (116f_x) - 16$$

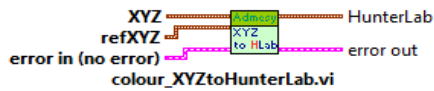
$$a^* = 500(f_x - f_y)$$

$$b^* = 200(f_y - f_z)$$

2.8 CIE 1931 XYZ to Hunter Lab

Program language:

Labview



C

```
int XYZtoHunterLab (XYZ_space_t *XYZ, XYZ_space_t
*refXYZ, CIELab_space_t *Lab)
```

Function description:

Converts a CIE XYZ colour space to Hunter Lab colour space regarding to the reference white value (d50 d65 ect..).

Input:

```
struct XYZ_space_t
struct XYZ_space_t (reference white XYZn)
```

Output:

```
struct CIELab_space_t
```

Formula:

$$L = 100 \sqrt{\frac{Y}{Y_n}}$$

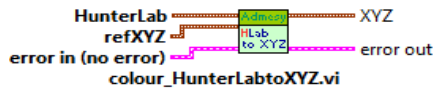
$$a = K_a \frac{\left(\frac{X}{X_n}\right) - \left(\frac{Y}{Y_n}\right)}{\sqrt{\frac{Y}{Y_n}}} \quad K_a \approx \frac{175}{198.04} (X_n + Y_n)$$

$$b = K_b \frac{\left(\frac{Y}{Y_n}\right) - \left(\frac{Z}{Z_n}\right)}{\sqrt{\frac{Y}{Y_n}}} \quad K_b \approx \frac{70}{218.11} (Y_n + Z_n)$$

2.9 Hunter Lab to CIE 1931 XYZ

Program language:

Labview



C

```
int HunterLabtoXYZ (CIELab_space_t *Lab,
XYZ_space_t *refXYZ, XYZ_space_t *XYZ)
```

Function description:

Converts a Hunter Lab colour space to CIE XYZ colour space regarding to the reference white value (d50 d65 ect..).

Input:

```
struct CIELab_space_t
struct XYZ_space_t (reference white XYZn)
```

Output:

```
struct XYZ_space_t
```

Formula:

$$Y = \left(\frac{L}{100} \right)^2 Y_n$$

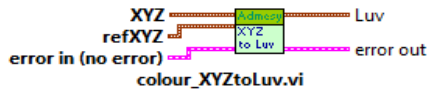
$$X = \left(\frac{a}{K_a} \sqrt{\frac{Y}{Y_n} + \frac{Y}{Y_n}} \right) X_n \quad K_a \approx \frac{175}{198.04} (X_n + Y_n)$$

$$Z = \left(\frac{Y}{Y_n} - \frac{b}{K_b} \sqrt{\frac{Y}{Y_n}} \right) Z_n \quad K_b \approx \frac{70}{218.11} (Y_n + Z_n)$$

2.10 CIE 1931 XYZ to CIE 1976 L*u*v*

Program language:

Labview



C

```
int XYZtoLuv (XYZ_space_t *XYZ, XYZ_space_t
*refXYZ, Luv_space_t *Luv)
```

Function description:

Converts a CIE XYZ colour space to CIE Luv colour space regarding to the reference white value (d50 d65 ect..).

Input:

```
struct XYZ_space_t
struct XYZ_space_t (reference white)
```

Output:

```
struct Luv_space_t
```

Formula:

$$e = \frac{216}{24389}, k = \frac{24389}{27}$$

$$y_r = \frac{Y}{WhiteRefY}, u'_r = \frac{4X}{X + 15Y + 3Z}, v'_r = \frac{9Y}{X + 15Y + 3Z}$$

$$u'_r = \frac{4X}{WhiteRefX + 15WhiteRefY + 3WhiteRefZ}$$

$$v'_r = \frac{9Y}{WhiteRefX + 15WhiteRefY + 3WhiteRefZ}$$

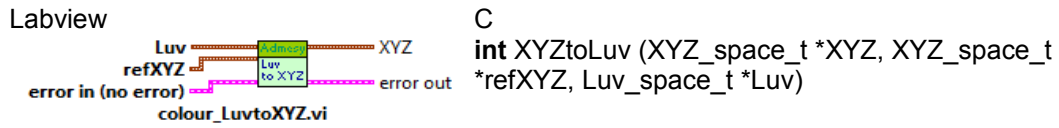
$$L = \begin{cases} 116y_r^{1/3} - 16 & y_r > e \\ ky_r & y_r \leq e \end{cases}$$

$$u = 13L(u' - u'_r)$$

$$v = 13L(v' - v'_r)$$

2.11 CIE 1976 L*u*v* to CIE 1931 XYZ

Program language:



Function description:

Converts a CIE Luv colour space to CIE XYZ colour space regarding to the reference white value (d50 d65 ect..).

Input:

```

struct Luv_space_t
struct XYZ_space_t (reference white)
    
```

Output:

```

struct XYZ_space_t
    
```

Formula:

$$e = \frac{216}{24389}, k = \frac{24389}{27}$$

$$u_0 = \frac{4X}{WhiteRefX + 15WhiteRefY + 3WhiteRefZ}$$

$$v_0 = \frac{9Y}{WhiteRefX + 15WhiteRefY + 3WhiteRefZ}$$

$$a = \frac{1}{3} \left(\frac{52L}{u + 13Lu_0} - 1 \right) \quad b = -5Y$$

$$c = \frac{-1}{3} \quad d = Y \left(\frac{39L}{v + 13Lv_0} - 5 \right)$$

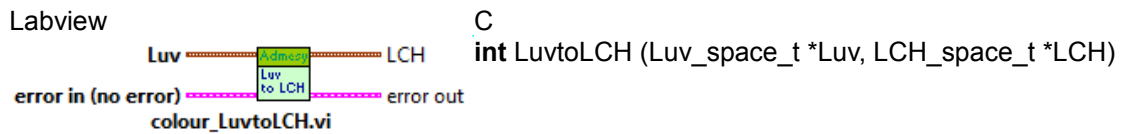
$$X = \frac{d - b}{a - c}$$

$$Y = \begin{cases} \left(\frac{L + 16}{116} \right)^3 & L > ke \\ \frac{L}{k} & L \leq ke \end{cases}$$

$$Z = Xa + b$$

2.12 CIE 1976 L*u*v* to CIE L*C*h*_{uv}

Program language:



Function description:

Converts a CIE Luv colour space to CIE LCH colour space.

Input:

struct Luv_space_t

Output:

struct LCH_space_t

Formula:

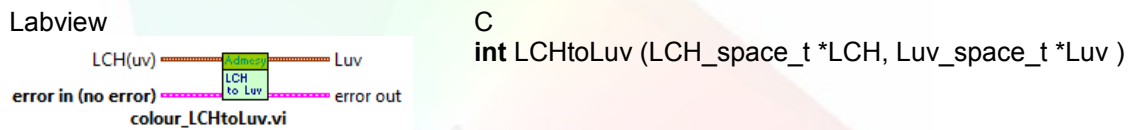
$$L = L$$

$$C = \sqrt{u^2 + v^2}$$

$$H = \tan^{-1}\left(\frac{v}{u}\right) \begin{cases} \text{if } H < 0^\circ & H + 360^\circ \\ \text{if } H \geq 360^\circ & H - 360^\circ \end{cases}$$

2.13 CIE L*C*h*_{uv} to CIE 1976 L*u*v*

Program language:



Function description:

Converts a CIE Luv colour space to CIE LCH colour space.

Input:

struct LCH_space_t

Output:

struct Luv_space_t

Formula:

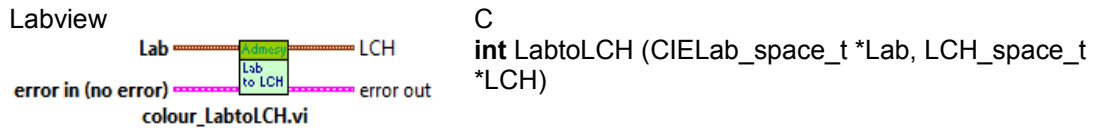
$$L = L$$

$$u = C \cos(H)$$

$$v = C \sin(H)$$

2.14 CIE 1976 L*a*b* to CIE L*C*h_{ab}

Program language:



Function description:

Converts a CIE Lab colour space to CIE LCH colour space.

Input:

struct CIELab_space_t

Output:

struct LCH_space_t

Formula:

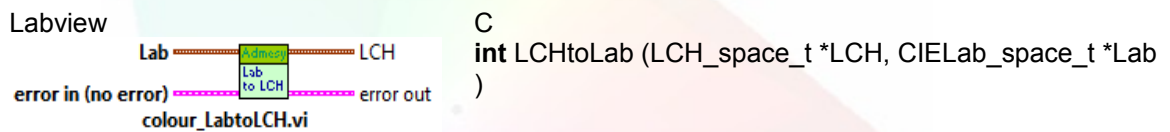
$$L = L$$

$$C = \sqrt{a^2 + b^2}$$

$$H = \tan^{-1}\left(\frac{b}{a}\right) \begin{cases} \text{if } H < 0^\circ & H + 360^\circ \\ \text{if } H \geq 360^\circ & H - 360^\circ \end{cases}$$

2.15 CIE L*C*h_{ab} to CIE 1976 L*a*b*

Program language:



Function description:

Converts a CIE LCH colour space to CIE Lab colour space.

Input:

struct LCH_space_t

Output:

struct CIELab_space_t

Formula:

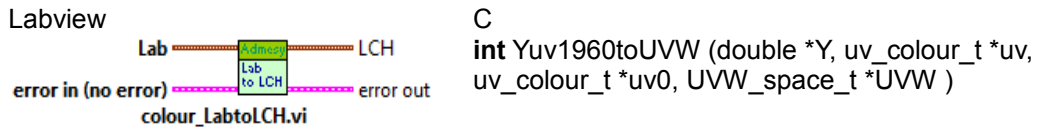
$$L = L$$

$$a = C \cos(H)$$

$$b = C \sin(H)$$

2.16 CIE 1960 Yuv to CIE 1964 U*V*W*

Program language:



Function description:

Converts a CIE Yuv colour space to CIE UVW colour space.

Input:

double Y
struct uv_colour_t
struct uv_colour_t reference uv0

Output:

struct UVW_space_t

Formula:

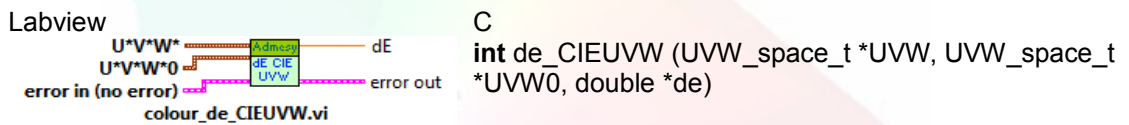
$$W = 25Y^{1/3}$$

$$U = 13W(u - u_0)$$

$$V = 13W(v - v_0)$$

2.17 ΔE CIE 1964 U*V*W*

Program language:



Function description:

calculate the colour difference (ΔE) of the UVW value regarding to the reference UVW

Input:

struct UVW_space_t
struct UVW_space_t reference UVW

Output:

double de

Formula:

$$\Delta E = \sqrt{(U - U_0)^2 + (V - V_0)^2 + (W - W_0)^2}$$

2.18 ΔE CIE 1976

Program language:

Labview



C

```
int de_CIE1976 (CIELab_space_t *Lab, CIELab_space_t
*refLab, double *de)
```

Function description:

calculate the colour difference (ΔE) of the Lab value regarding to the reference Lab

Input:

```
struct CIELab_space_t
struct CIELab_space_t reference Lab
```

Output:

```
double de
```

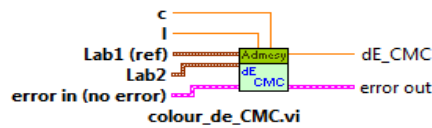
Formula:

$$\Delta E = \sqrt{(L - L_0)^2 + (a - a_0)^2 + (b - b_0)^2}$$

2.19 ΔE CIE CMC (1984)

Program language:

Labview



C

```
int de_CMC (CIELab_space_t *Lab, CIELab_space_t
*refLab, double l, double c, double *de)
```

Function description:

calculate the colour difference (ΔE) of the Lab value regarding to the reference Lab using two parameters l and c , typically expressed as CMC($l:c$). Most used values for acceptability are CMC(2:1) and for perceptibility are CMC(1:1)

Input:

```
struct CIELab_space_t
struct CIELab_space_t reference Lab
double l
double c
```

Output:

```
double de
```

Formula: sample color $L_2a_2b_2$ and a reference color $L_1a_1b_1$

$$C_1 = \sqrt{a_1^2 + b_1^2} \quad \Delta L = L_1 - L_2$$

$$C_2 = \sqrt{a_2^2 + b_2^2} \quad \Delta a = a_1 - a_2$$

$$\Delta C = C_1 - C_2 \quad \Delta b = b_1 - b_2$$

$$\Delta H = \sqrt{\Delta a^2 + \Delta b^2 - \Delta C^2}$$

$$H_1 = \tan^{-1} \left(\frac{b_1}{a_1} \right) \begin{cases} \text{if } H < 0^\circ & H + 360^\circ \\ \text{if } H \geq 360^\circ & H - 360^\circ \end{cases}$$

$$T = \begin{cases} 0.56 + |0.2 \cos(H_1 + 168)| & 164 \leq H_1 \leq 345 \\ 0.36 + |0.4 \cos(H_1 + 35)| & \text{otherwise} \end{cases}$$

$$F = \sqrt{\frac{C_1^4}{C_1^4 + 1900}}$$

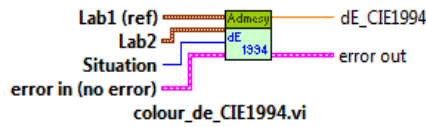
$$S_L = \begin{cases} 0.511 & L_1 < 16 \\ \frac{0.040975L_1}{1 + 0.01765L_1} & L_1 \geq 16 \end{cases} \quad S_C = \frac{0.0638C_1}{1 + 0.0131C_1} + 0.638 \quad S_H = S_C(FT + 1 - F)$$

$$\Delta E = \sqrt{\left(\frac{\Delta L}{l S_L} \right)^2 + \left(\frac{\Delta C}{c S_C} \right)^2 + \left(\frac{\Delta H}{S_H} \right)^2}$$

2.20 ΔE CIE 1994

Program language:

Labview



C

```
int de_CMC (CIELab_space_t *Lab, CIELab_space_t
*refLab, double l, double c, double *de)
```

Function description:

calculate the colour difference (ΔE) of the Lab value regarding to the reference Lab.
This colour difference model uses different values for graphics and textile arts

Input:

```
struct CIELab_space_t
struct CIELab_space_t reference Lab
unsigned char textiles
```

Output:

```
double de
```

Formula: sample color $L_2a_2b_2$ and a reference color $L_1a_1b_1$

$$C_1 = \sqrt{a_1^2 + b_1^2} \quad \Delta L = L_1 - L_2$$

$$C_2 = \sqrt{a_2^2 + b_2^2} \quad \Delta a = a_1 - a_2$$

$$\Delta C = C_1 - C_2 \quad \Delta b = b_1 - b_2$$

$$\Delta H = \sqrt{\Delta a^2 + \Delta b^2 - \Delta C^2}$$

$$K_L = \begin{cases} 1 & \text{default} \\ 2 & \text{textiles} \end{cases} \quad K_c = 1 \quad K_H = 1$$

$$K_1 = \begin{cases} 0.045 & \text{graphic arts} \\ 0.048 & \text{textiles} \end{cases} \quad K_2 = \begin{cases} 0.015 & \text{graphic arts} \\ 0.014 & \text{textiles} \end{cases}$$

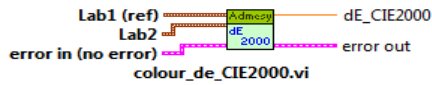
$$S_L = 1 \quad S_C = 1 + K_1 C_1 \quad S_H = 1 + K_2 C_2$$

$$\Delta E = \sqrt{\left(\frac{\Delta L}{K_L S_L}\right)^2 + \left(\frac{\Delta C}{K_c S_C}\right)^2 + \left(\frac{\Delta H}{K_H S_H}\right)^2}$$

2.21 ΔE CIE 2000

Program language:

Labview



C

```
int de_2000 (CIELab_space_t *Lab, CIELab_space_t
*refLab, double *de)
```

Function description:

calculate the colour difference (ΔE) of the Lab value regarding to the reference Lab.

Input:

struct CIELab_space_t

struct CIELab_space_t reference Lab

Output:

double de

Formula: sample color $L_2a_2b_2$ and a reference color $L_1a_1b_1$

$$C_1 = \sqrt{a_1^2 + b_1^2} \quad C_2 = \sqrt{a_2^2 + b_2^2}$$

$$\bar{C} = \frac{C_1 - C_2}{2} \quad G = \frac{\left(1 - \sqrt{\frac{\bar{C}^7}{\bar{C}^7 + 25^7}}\right)}{2}$$

$$\bar{L}' = \frac{L_1 + L_2}{2}$$

$$a'_1 = a_1(1 + G) \quad a'_2 = a_2(1 + G)$$

$$C'_1 = \sqrt{a'^2_1 + b_1^2} \quad C'_2 = \sqrt{a'^2_2 + b_2^2}$$

$$\bar{C}' = \frac{C'_1 + C'_2}{2}$$

$$h'_1 = \begin{cases} \tan^{-1}\left(\frac{b_1}{a'_1}\right) & \tan^{-1}\left(\frac{b_1}{a'_1}\right) \geq 0 \\ \tan^{-1}\left(\frac{b_1}{a'_1}\right) + 360^\circ & \tan^{-1}\left(\frac{b_1}{a'_1}\right) < 0 \end{cases}$$

$$h'_2 = \begin{cases} \tan^{-1}\left(\frac{b_2}{a'_2}\right) & \tan^{-1}\left(\frac{b_2}{a'_2}\right) \geq 0 \\ \tan^{-1}\left(\frac{b_2}{a'_2}\right) + 360^\circ & \tan^{-1}\left(\frac{b_2}{a'_2}\right) < 0 \end{cases}$$

$$H' = \begin{cases} \frac{h'_1 + h'_2 + 360^\circ}{2} & |h'_1 - h'_2| > 180^\circ \\ \frac{h'_1 + h'_2}{2} & |h'_1 - h'_2| \leq 180^\circ \end{cases}$$

$$T = 1 - 0.17\cos(\bar{H}' - 30^\circ) + 0.24\cos(2\bar{H}') + 0.32\cos(3\bar{H}' + 6^\circ) - 0.2\cos(4\bar{H}' - 63^\circ)$$

$$\Delta h' = \begin{cases} h'_2 - h'_1 & |h'_2 - h'_1| \leq 180^\circ \\ h'_2 - h'_1 + 360^\circ & |h'_2 - h'_1| > 180^\circ; h'_2 \leq h'_1 \\ h'_2 - h'_1 - 360^\circ & |h'_2 - h'_1| > 180^\circ; h'_2 > h'_1 \end{cases}$$

$$\Delta L' = L_2 - L_1 \quad \Delta C' = C'_2 - C'_1 \quad \Delta H' = 2\sqrt{C'_1 C'_2} \sin\left(\frac{\Delta h'}{2}\right)$$

$$S_L = 1 + \frac{0.015(\bar{L}' - 50)^2}{\sqrt{20 + (\bar{L}' - 50)^2}} \quad S_C = 1 + 0.045\bar{C}' \quad S_H = 1 + 0.015\bar{C}' T$$

$$\Delta\theta = 30 \exp\left\{-\left(\frac{H' - 275^\circ}{25}\right)^2\right\} \quad R_C = \sqrt{\frac{\bar{C}'^7}{\bar{C}'^7 + 25^7}} \quad R_T = -2R_C \sin(2\Delta\theta)$$

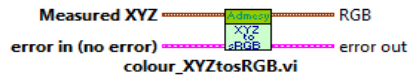
$$K_L = 1 \quad K_C = 1 \quad K_H = 1$$

$$\Delta E = \sqrt{\left(\frac{\Delta L}{K_L S_L}\right)^2 + \left(\frac{\Delta C}{K_C S_C}\right)^2 + \left(\frac{\Delta H}{K_H S_H}\right)^2 + R_T \left(\frac{\Delta C}{K_C S_C}\right) \left(\frac{\Delta H}{K_H S_H}\right)}$$

2.22 CIE 1931 XYZ to sRGB

Program language:

Labview



C

`int XYZtosRGB (XYZ_space_t *XYZ, RGB_data_t *RGB)`

Function description:

Converts CIE XYZ colour space to sRGB values.

Input:

struct XYZ_space_t

Output:

struct RGB_data_t

Formula:

$$\begin{bmatrix} R_{linear} \\ G_{linear} \\ B_{linear} \end{bmatrix} = \begin{bmatrix} 3.2410 & -1.5374 & -0.4986 \\ -0.9692 & 1.8760 & 0.0416 \\ 0.0556 & -0.2040 & 1.0570 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$R = \begin{cases} 12.92R_{linear} & R_{linear} \leq 0.0031308 \\ (1 + 0.55)R_{linear}^{1/2.4} & R_{linear} > 0.0031308 \end{cases}$$

$$G = \begin{cases} 12.92G_{linear} & G_{linear} \leq 0.0031308 \\ (1 + 0.55)G_{linear}^{1/2.4} & G_{linear} > 0.0031308 \end{cases}$$

$$B = \begin{cases} 12.92B_{linear} & B_{linear} \leq 0.0031308 \\ (1 + 0.55)B_{linear}^{1/2.4} & B_{linear} > 0.0031308 \end{cases}$$

2.23 CCT to xy

Program language:

Labview

C

`void D_illuminant_T_to_xy(unsigned int *temp, cxcy_t *xy)`

Function description:

Converts the correlated colour temperature of a CIE D-illuminant to the chromaticity (xy) value.

Input:

unsigned int temp (colour temperature)

Output:

struct cxcy_t

Formula:

$$x_D = \begin{cases} \frac{-4.6070 * 10^9}{T^3} + \frac{2.9678 * 10^6}{T^2} + \frac{0.09911 * 10^3}{T} + 0.244063 & 4000K \leq T \leq 7000K \\ \frac{-2.0064 * 10^9}{T^3} + \frac{1.9018 * 10^6}{T^2} + \frac{0.24748 * 10^3}{T} + 0.227040 & 7000K < T \leq 25000K \end{cases}$$

$$y_D = -3.000x_D^2 + 2.870x_D - 0.275$$

2.24 Planckian locus to xy

Program language:

Labview

C

void plackian_xy(unsigned int *temp, cxcy_t *xy)

Function description:

Get the planckian x,y values of a specified colour temperature.

Input:

unsigned int temp (colour temperature)

Output:

struct cxcy_t

Formula:

$$x_C = \begin{cases} -0.2661239 \frac{10^9}{T^3} - 0.2343580 \frac{10^6}{T^2} + 0.8776956 \frac{10^3}{T} + 0.179910 & 1667\text{K} \leq T \leq 4000\text{K} \\ -3.0258469 \frac{10^9}{T^3} + 2.1070379 \frac{10^6}{T^2} + 0.2226347 \frac{10^3}{T} + 0.24039 & 4000\text{K} < T \leq 25000\text{K} \end{cases}$$
$$y_C = \begin{cases} -1.1063814 x_C^3 - 1.34811020 x_C^2 + 2.18555832 x_C - 0.20219683 & 1667\text{K} \leq T \leq 2222\text{K} \\ -0.9549476 x_C^3 - 1.37418593 x_C^2 + 2.09137015 x_C - 0.16748867 & 2222\text{K} \leq T \leq 4000\text{K} \\ +3.0817580 x_C^3 - 5.87338670 x_C^2 + 3.75112997 x_C - 0.37001483 & 4000\text{K} \leq T \leq 25000\text{K} \end{cases}$$

2.25 Planckian locus to CIE 1960 uv

Program language:

Labview

C

void plackian_uv(unsigned int *temp, cxcy_t *xy)

Function description:

Get the planckian CIE 1960 u,v values of a specified colour temperature.

Input:

unsigned int temp (colour temperature)

Output:

struct cxcy_t

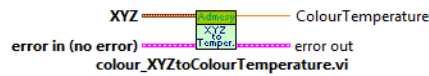
Formula:

$$u_T = \frac{0.860117757 + 1.54118254 * 10^{-4} T + 1.28641212 * 10^{-7} T^2}{1 + 8.42420235 * 10^{-4} T + 7.08145163 * 10^{-7} T^2}$$
$$v_T = \frac{0.317398726 + 4.22806245 * 10^{-5} T + 4.20481693 * 10^{-8} T^2}{1 - 2.89741816 * 10^{-5} T + 1.61456053 * 10^{-7} T^2}$$

2.26 CIE 1931 XYZ to CCT

Program language:

Labview



C

```
int XYZtoCorColorTemp(XYZ_space_t *XYZ, double *temp)
```

Function description:

Calculate the Correlated colour temperature of the CIE XYZ values.

Input:

```
struct XYZ_space_t
```

Output:

```
double temp
```

formula:

$$x = \frac{X}{(X+Y+Z)}$$

$$y = \frac{Y}{(X+Y+Z)}$$

$$n = \frac{x-0.3366}{y-0.1735}$$

$$CCT_{xy} = -949.86315 + 6253.80338 \exp\left(\frac{-n}{0.92159}\right) + 28.70599 \exp\left(\frac{-n}{0.20039}\right) + 0.00004 \exp\left(\frac{-n}{0.07125}\right)$$

if

$$CCT_{xy} > 50000 \quad CCT_{xy} = \begin{cases} n = \frac{x-0.3356}{y-0.1691} \\ CCT_{xy} = 36284.48953 + 0.00228 \exp\left(\frac{-n}{0.07861}\right) + 5.4535 * 10^{-36} \exp\left(\frac{-n}{0.01543}\right) \end{cases}$$

$$CCT_{xy} < 2500K$$

$$CCT_{xy} = 0$$

$$CCT_{xy} > 80000K$$

$$CCT_{xy} = 0$$

2.27 CIE 1931 XYZ to CCT Robertson's method

Program language:

Labview

C

```
int XYZtoCorColorTemp_Rob(XYZ_space_t *XYZ, double *temp)
```

Function description:

Calculate the Correlated colour temperature of the CIE XYZ values (Robertson's method).

Input:

```
struct XYZ_space_t
```

Output:

```
double temp
```

formula:

The Robertson's method work with linear interpolation of the isotherm's mired values.

$$u_T = \frac{4Y}{X + 15Y + 3Z}$$

$$v_T = \frac{6Y}{X + 15Y + 3Z}$$

$$m_i = \frac{-1}{l_i} \text{ where } l_i \text{ is the slope of the locus at } (u_i, v_i)$$

$$d_i = \frac{(v_T - v_i) - m_i(u_T - u_i)}{\sqrt{(1 + m_i^2)}}$$

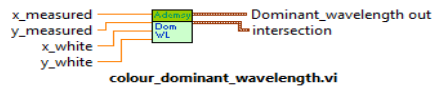
$$\frac{1}{T_c} = \frac{1}{T_i} + \frac{d_i}{d_i - d_{i+1}} \left(\frac{1}{T_{i+1}} - \frac{1}{T_i} \right)$$

where T_i and T_{i+1} are the color temperatures of the lookup isotherms.

2.28 Chromaticity (xy) to Dominant wavelength

Program language:

Labview



C

```
int dominant_wl (DWL *domwl, double meas_x, double meas_y, double white_x, double white_y)
```

Function description:

Calculate the Dominant wavelength of the chromaticity (xy) values regarding to the reference white chromaticity (xy).

Input:

double meas_x
double meas_y
double white_x
double white_y

Output:

struct DWL

formula:

A colour measurement provides us with the chromaticity (xy) values, the white reference is set to a specific illuminant that provides the reference chromaticity (xy) values. These points are “plotted” in the CIE 1931 chromaticity diagram and drawing a line from the reference illuminant through the measured chromaticity (xy) values until the intersection with the spectrum locus and the dominant wavelength can be read.

The purity P_e is defined by the ratio a / b , where “a” is the distance between the measured chromaticity values and the reference illuminant and “b” is the distance between the reference illuminant to the point on the spectrum locus where the dominant wavelength is read.

2.29 Lumen to Candela

Program language:

Labview



C

```
int LumentoCandela (double lm, double beam_angle,  
double *cd)
```

Function description:

Converts Lumen (lm) to Candela (cd) of the measured source regarding the beam angle in degrees.

Input:

double lm

double beam_angle

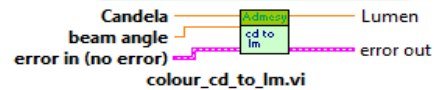
Output:

double cd

2.30 Candela to Lumen

Program language:

Labview



C

```
int LumentoCandela (double lm, double beam_angle,  
double *cd)
```

Function description:

Converts Candela (cd) to Lumen (lm) of the measured source regarding the beam angle in degrees.

Input:

double cd

double beam_angle

Output:

double lm