

## Annexe S.3

```

1  //  affichage.c
2  //
3  //  Description : fonctions liée à l'affichage E-paper
4  //  Auteur : Perret Mélissa
5  //  Création : 16/09/2024
6  //  Modifications : --
7
8  //  Version   : V1.0
9  /*-----*/
10
11
12 #include "affichage.h"
13 #include "EPD_2in13_V4.h" // pour EPD_2in13_V4_Init etc.
14 #include "GUI_Paint.h" // pour les fonctions Paint (Paint_NewImage, Paint_DrawString_EN etc.)
15 #include <stdlib.h> // pour malloc et free
16
17
18 ///// Fonction AffichageEPaper (affichage des données sur l'écran E-paper)
19 ///// Description: affiche les données sur l'écran E-paper (valeurs provenant du serveur, valeurs mesurées avec la sonde)
20 ///// Entrées: Pointeur:Mesures mesures (données mesurées), Tableau:DefinitionValeur valeursServeur (valeurs provenant du serveur)
21 ///// Sorties: uint16_t (-1 si problème, 0 si affiche réussi)
22 int AffichageEPaper(Mesures* mesures, DefinitionValeur valeursServeur[])
23 {
24     // Initialisation du module de développement
25     if(DEV_Module_Init() != 0)
26     {
27         return -1; // Initialisation échouée
28     }
29
30     // Créer une nouvelle image pour le cache
31     UBYTE *Image;
32     UWORD Imagesize = ((EPD_2in13_V4_WIDTH % 8 == 0)? (EPD_2in13_V4_WIDTH / 8 ) : (EPD_2in13_V4_WIDTH / 8 + 1)) * EPD_2in13_V4_HEIGHT; //instruction
    ternaire ?
33     if((Image = (UBYTE *)malloc(Imagesize)) == NULL)
34     {
35         return -1; // Allocation impossible. Problème de HEAP_SIZE insuffisante ?
36     }
37
38     // Mettre à jour les données considérées comme affichées
39     mesures->temperatureAffichee = mesures->temperatureActuelle;
40     mesures->humiditeAffichee = mesures->humiditeActuelle;
41
42     EPD_2in13_V4_Init(); // Initisaliation E-paper
43
44     // Initialisation de l'image avec des pixels blancs
45     Paint_NewImage(Image, EPD_2in13_V4_WIDTH, EPD_2in13_V4_HEIGHT, 90, WHITE);
46     Paint_SelectImage(Image);
47     Paint_Clear(WHITE);
48
49     // Première colonne
50     Paint_DrawString_EN(0, 45, "Actu", &Font16, WHITE, BLACK);
51     Paint_DrawString_EN(0, 65, "Min", &Font16, WHITE, BLACK);
52     Paint_DrawString_EN(0, 85, "Max", &Font16, WHITE, BLACK);

```

```
53 Paint_DrawString_EN(0, 105, "Ecart", &Font16, WHITE, BLACK);
54
55 Paint_DrawLine(60, 23, 60, 122, BLACK, DOT_PIXEL_1X1, LINE_STYLE_SOLID); // Ligne verticale pour séparer les colonnes
56
57 // Seconde colonne (température)
58 Paint_DrawString_EN(23, 3, "Temperature", &Font16, WHITE, BLACK); // Titre colonne
59 Paint_DrawString_EN(85, 23, "[", &Font16, WHITE, BLACK); // Unité colonne
60 Paint_DrawCircle(100, 23, 2, BLACK, DOT_PIXEL_1X1, DRAW_FILL_EMPTY); // Unité colonne (symbole degré °)
61 Paint_DrawString_EN(100, 23, "C]", &Font16, WHITE, BLACK); // Unité colonne
62
63 Paint_DrawNumDecimals(75, 45, mesures->temperatureActuelle, &Font16, NOMBRE_DECIMALES_MESURES, BLACK, WHITE); // Température actuelle
64 Paint_DrawNumDecimals(75, 65, valeursServeur[SEUIL_TEMPERATURE_MIN].valeur, &Font16, NOMBRE_DECIMALES_MESURES, BLACK, WHITE); // Température
min
65 Paint_DrawNumDecimals(75, 85, valeursServeur[SEUIL_TEMPERATURE_MAX].valeur, &Font16, NOMBRE_DECIMALES_MESURES, BLACK, WHITE); // Température max
66 Paint_DrawNumDecimals(75, 105, valeursServeur[ECART_TEMPERATURE].valeur, &Font16, NOMBRE_DECIMALES_MESURES, BLACK, WHITE); // Ecart température
67
68 Paint_DrawLine(150, 0, 150, 122, BLACK, DOT_PIXEL_1X1, LINE_STYLE_SOLID); // Ligne verticale pour séparer les colonnes
69
70 // Troisième colonne (humidité)
71 Paint_DrawString_EN(155, 3, "Humidite", &Font16, WHITE, BLACK); // Titre colonne
72 Paint_DrawString_EN(180, 23, "[%]", &Font16, WHITE, BLACK); // Unité colonne
73 Paint_DrawNumDecimals(170, 43, mesures->humiditeActuelle, &Font16, NOMBRE_DECIMALES_MESURES, BLACK, WHITE); // Humidité actuelle
74 Paint_DrawNumDecimals(170, 65, valeursServeur[SEUIL_HUMIDITE_MIN].valeur, &Font16, NOMBRE_DECIMALES_MESURES, BLACK, WHITE); // Humidité min
75 Paint_DrawNumDecimals(170, 85, valeursServeur[SEUIL_HUMIDITE_MAX].valeur, &Font16, NOMBRE_DECIMALES_MESURES, BLACK, WHITE); // Humidité Max
76 Paint_DrawNumDecimals(170, 105, valeursServeur[ECART_HUMIDITE].valeur, &Font16, NOMBRE_DECIMALES_MESURES, BLACK, WHITE); // Ecart humidité
77
78 Paint_DrawLine(0, 40, 250, 40, BLACK, DOT_PIXEL_1X1, LINE_STYLE_SOLID); // Ligne horizontale (séparation)
79
80 EPD_2in13_V4_Display_Base(Image); // Affichage
81
82 // Libérer la mémoire allouée pour l'image
83 free(Image);
84 Image = NULL;
85
86 DEV_Module_Exit(); // Fermeture
87
88 return 0;
89 }
90
```