

```

1  #ifndef GesFifoTh32_H
2  #define GesFifoTh32_H
3  /*-----*/
4  //  GesFifoTh32.h
5  /*-----*/
6  //  Description :    Création entête pour utilisation
7  //                  de GesFifoTh32
8  //
9  //  Auteur       :    C. Huber
10 //
11 //  Version      :    V1.6
12 //  Compilateur  :    XC32 V1.42 + Harmony 1.08
13 //
14 //  Modifications :
15 //  CHR 19.12.2014  remplacement typedef32 par stdint
16 //  CHR 20.12.2016  fifosize en int32_t pour permettre des
17 //                  fifo de grande taille
18 //
19 /*-----*/
20
21
22
23
24 // structure décrivant un FIFO
25 typedef struct fifo {
26     int32_t fifoSize;    // taille du fifo
27     int8_t *pDebFifo;    // pointeur sur début du fifo
28     int8_t *pFinFifo;    // pointeur sur fin du fifo
29     int8_t *pWrite;      // pointeur d'écriture
30     int8_t *pRead;       // pointeur de lecture
31 } S_fifo;
32
33 /*-----*/
34 /* Définition des fonctions de gestion du fifo          */
35 /*-----*/
36
37 /*-----*/
38 /* InitFifo      */
39 /*=====*/
40
41 // Initialisation du descripteur de FIFO
42 void InitFifo ( S_fifo *pDescrFifo, int32_t FifoSize, int8_t *pDebFifo, int8_t InitVal
43 );
44
45 /*-----*/
46 /* GetWriteSpace */
47 /*=====*/
48
49 // Retourne la place disponible en écriture
50 int32_t GetWriteSpace ( S_fifo *pDescrFifo);
51
52 /*-----*/
53 /* GetReadSize  */
54 /*=====*/
55
56 // Retourne le nombre de caractères à lire
57
58 int32_t GetReadSize ( S_fifo *pDescrFifo);
59
60 /*-----*/
61 /* PutCharInFifo */
62 /*=====*/
63
64 // Dépose un caractère dans le FIFO
65 // Retourne 0 si OK, 1 si FIFO full
66
67 uint8_t PutCharInFifo ( S_fifo *pDescrFifo, int8_t charToPut );
68
69 /*-----*/
70 /* GetCharFromFifo */
71 /*=====*/
72

```

```
73 // Obtient (lecture) un caractère du fifo
74 // retourne 0 si OK, 1 si empty
75 // le caractère lu est retourné par référence
76
77 uint8_t GetCharFromFifo ( S_fifo *pDescrFifo, int8_t *carLu );
78
79 #endif
80
```