

```
1  /**
2      ****
3      * C Library for the BME280 Sensor
4      ****
5      * @author Samuel Ruhl, Alexander Meier
6      * @date 2017-04-04
7      * @file BME280.c
8      * @brief Contains Functions for using the SPI
9      ****
10 */
11
12 #include <Dave.h>
13 #include "spi.h"
14 #include "xmc4700.h"
15
16
17 /** SEND
18     /
19     uint8_t SPI_send(uint8_t data)
20     {
21         /*
22         while(!SPI_I2S_GetFlagStatus(SPI1, SPI_I2S_FLAG_TXE));
23         SPI_I2S_SendData(SPI1, data);
24         while(!SPI_I2S_GetFlagStatus(SPI1, SPI_I2S_FLAG_RXNE));
25
26         return SPI_I2S_ReceiveData(SPI1);
27         */
28         //uint8_t rx_dat;
29
30         SPI_MASTER_Transmit(&SPI_MASTER_0,&data,1);
31
32         return 0;
33     }
34
35 /** REC
36     */
37     uint8_t SPI_rec(void)
38     {
39         /*
40         while(!SPI_I2S_GetFlagStatus(SPI1, SPI_I2S_FLAG_TXE));
41         SPI_I2S_SendData(SPI1, address);
42         while(!SPI_I2S_GetFlagStatus(SPI1, SPI_I2S_FLAG_RXNE));
43         SPI_I2S_ReceiveData(SPI1);
```

```
44     while(!SPI_I2S_GetFlagStatus(SPI1, SPI_I2S_FLAG_TXE));
45     SPI_I2S_SendData(SPI1, 0x00);
46     while(!SPI_I2S_GetFlagStatus(SPI1, SPI_I2S_FLAG_RXNE));
47
48     return SPI_I2S_ReceiveData(SPI1);
49
50     */
51     uint8_t data;
52     SPI_MASTER_Receive(&SPI_MASTER_0,&data,1);
53     return data;
54
55 }
56
57
58 /** FT800 SPI select ↗
59     *****/
60 void FT_spi_select(void)
61 {
62     PORT3->OUT &= ~(1<<12);
63 }
64
65 /** FT800 SPI deselect ↗
66     *****/
67 void FT_spi_deselect(void)
68 {
69     for(int i = 0; i < 100; i++);
70     PORT3->OUT |= 1<<12;
71 }
72
73
74 void ms_delay(uint32_t millisec){
75     //Device running on 144MHz
76     millisec *= 31100;
77     while(millisec--){
78         __NOP();           //No Operation
79     }
80 }
81
82
83
84 void us_delay(uint32_t microsec){
85     //Device running on 144MHz
86     microsec *= 31;
87     while(microsec--){
88         __NOP();           //No Operation
89     }
90 }
91
92
```