

## I<sup>2</sup>C Library

I<sup>2</sup>C Library für den I2C1 des STM32F10x

Mit dieser Library kann man beliebig viele Bytes senden, und 1 – 3 Bytes empfangen.

- **char I2C1\_init(void)**

Dieses Upro muss immer als erstes aufgerufen werden um die Start –, Sende – und Empfangsroutinen verwenden zu können.

Initialisiert den I2C-Bus als Master mit einer Taktfrequenz von 100kHz.

Return – Wert: Error-Code, kann mit I2C\_error\_message(char errorcode) ausgewertet werden.

- **char I2C1\_start(char address, uint8\_t read\_anz)**

Mit diesem Upro wird die Start-Bedingung des I2C-Buses erzeugt und anschließend die

7-bit Slave-Address + W/R-bit gesendet

Return - Wert: Error-Code, kann mit I2C\_error\_message(char err\_msg) ausgewertet werden.

char address: 7-bit Slave Address + W/R-bit

uint8\_t read\_anz: Anzahl der zulesenden Bytes wenn Bytes empfangen werden.(1-3)

Wenn gesendet wird soll eine 0 übergeben werden.

- **char I2C1\_send(char \*data, uint8\_t anz)**

Mit diesem Upro können Daten gesendet werden.

Return - Wert: Error-Code, kann mit I2C\_error\_message(char err\_msg) ausgewertet werden.

char \*data: Byte-Array mit dem zusendenden Daten.

uint8\_t anz: Länge des Arrays.

- **char I2C1\_read(char \*data)**

Mit diesem Upro kann ein Byte gelesen werden.

Return - Wert: Error-Code, kann mit I2C\_error\_message(char err\_msg) ausgewertet werden.

char \*data: empfangenes Byte

- **char I2C1\_read\_n(char \*data, uint8\_t anz)**

Mit diesem Upro können 2 – 3 Byte empfangen werden.

Return - Wert: Error-Code, kann mit I2C\_error\_message(char err\_msg) ausgewertet werden.

char \*data: Byte-Array mit den empfangenen Daten.

uint8\_t anz: Anzahl der zu empfangenen Bytes.

Library Files: I2C\_LIB\_KOE.lib I2C\_LIB\_KOE.h

Beispiel Code:

//1 Byte Empfangen

#include "I2C\_LIB\_KOE.h"

int main (void)

```
{
    char daten;
    I2C1_init();
    I2C1_start(0xD1,1);
    I2C1_read(daten);
    for(;;)
    {
    }
}
```

Beispiel Code:

//2 Byte Empfangen

#include "I2C\_LIB\_KOE.h"

int main (void)

```
{
    char daten[2];
    I2C1_init();
    I2C1_start(0xD1,2);
    I2C1_read_n(daten, 2);
    for(;;)
    {
    }
}
```

Beispiel Code:

//3 Byte Senden

#include "I2C\_LIB\_KOE.h"

int main (void)

```
{
    char daten[3]=0x20,0xFF,0xFF;
    I2C1_init();
    I2C1_start(0x82,0);
    I2C1_send(daten, 3);
    for(;;)
    {
    }
}
```