INNOVENT
INNOVENT e.V.
Technologieentwicklung
Oberflächen, Werkstoffe und Systeme

SENIS
magnetic & current measurement

Library

3mtslib

Version : 1.0

Contents

1. **Introduction** 2

2. **Standard structures** 2

3. **Function Reference** 2

3.1. **Count devices** 2

3.2. **Open device** 3

3.3. **Close device** 3

3.4. **Get sensor count** 3

3.5. **Get sensor values** 4

3.6. **Set range** 4

3.7. **Get range** 4

3.8. **Set speed** 5

3.9. **Get speed** 5

3.10. **Get firmware version** 5

3.11. **Get device name** 6

3.12. **Clear buffer** 6

1. Introduction

3mtsLib is a software library for easy data acquisition and communication with 3mts sensors. The software consists of a DLL to be imported into your data acquisition project.

2. Standard structures

```
typedef struct {
   int dimSize;
   int elt[x];
   } TD2;
typedef TD2 **TD2Hdl;


typedef struct {
   int dimSize;
   char elt[x];
   } TD3;
typedef TD3 **TD3Hdl;
```

3. Function Reference

All functions are declared in C syntax.

3.1. Count devices

Count the available devices.

Declaration: int count_devices(unsigned short* number_of_devices)

Output:

number_of_devices : Number of available devices

Return value:

0x0	ok
0x8000	Device not initialized

3.2. Open device

Open the measurement device.

Declaration: int open_device(int* device_number)

Input:

device_number : Device number

Return value:

0x0	ok
0x8000	Device not initialized

3.3. Close device

Close the opened measurement device.

Declaration: int close_device(int* device_number)

Input:

device_number : Device number

Return value:

0x0	ok
0x8000	Device not initialized

3.4. Get sensor count

Return the number of sensor channels.

Declaration: int get_sensor_count(int* device_number, int *sensor_count)

Input:

device_number : Device number

Output:

sensor_count: Number of sensor channels

Return value:

0x0	ok
0x8000	Device not initialized

3.5. Get sensor values

Get values from the measurement device. Before you call this function, you have to allocate space for the TD2Hdl structure.

Declaration: int get_sensor_values (int* device_number, unsigned long* timestamp, TD2Hdl values)

Input:

device_number : Device number

Output:

timestamp:	Time stamp
values:	Measurement values (4x int)

Return value:

0x0	ok
0x8000	Device not initialized

3.6. Set range

Set the measurement range of the sensor.

Declaration: int set_range (int* device_number ,unsigned short range)

Input:

device_number : Device number

range: Measurement range (0=0,1mT; 1=0,5mT; 2=3T; 3=20T)

Return value:

0x0	ok
0x8000	Device not initialized
0x8001	Range outside of value range

3.7. Get range

Get the measurement range of the sensor.

Declaration: int get_range (int* device_number, unsigned short* range)

Input:

device_number : Device number

Output:

range:	Measurement range (0=0,1mT; 1=0,5mT; 2=3T; 3=20T)

Return value:

0x0	ok
0x8000	Device not initialized

3.8. Set speed

Set measurement speed.

Declaration: int set_speed(int* device_number,unsigned short speed)

Input:

device_number : Device number

speed: Measure time period (1=1ms; 2=2ms; 3=3ms;…)

Return value:

0x0	ok
0x8000	Device not initialized

3.9. Get speed

Get measurement speed.

Declaration: int get_speed(int* device_number, unsigned short* speed)

Input:

device_number : Device number

Output:

speed: Measure time period (1=1ms; 2=2ms; 3=3ms;…)

Return value:

0x0	ok
0x8000	Device not initialized

3.10. Get firmware version

Get firmware version. Before you call this function, you have to allocate space for the TD3Hdl structure.

Declaration: int get_firmware_version(int* device_number,TD3Hdl values)

Input:

device_number : Device number

Output:

values: Firmware version fwx.x.x (7byte)

Return value:

0x0	ok
0x8000	Device not initialized

3.11. Get device name

Get the name of the measurement device. Before you call this function, you have to allocate space for the TD3Hdl structure.

Declaration: int get_device_name(int* device_number,TD3Hdl values)

Input:

device_number : Device number

Output:

values: Device name 3MTS REVx.x SNR xxx-YY (23 byte)

Return value:

0x0	ok
0x8000	Device not initialized

3.12. Clear buffer

Clear the internal value buffer.

Declaration: int clear_buffer(int* device_number)

Input:

device_number : Device number

Return value:

0x0	ok
0x8000	Device not initialized