Table of Content

Note:

The following descriptions contains links to online documents that provide more details. These links are *printed in blue and underlined*.

1. Introduction

The USBAMP-4 is an intelligent devices offering a wide dynamic range by using 4 calibrated measurement ranges for the measurement of photo currents from photodiodes of nearly any type. The photodiodes are operated in unbiased (photovoltaic) mode resulting in very low dark readings.

The devices can be connected to either PCs and Tablets with Windows, Linux or Mac-OS operating systems as well as Android and iOS. In all systems the sensor presents a standard serial peripheral for communication.

In this manual the underlying generic protocol is described independently of a specific programming language.

MODBUS RTU is used as logical protocol for the communication. The devices present a slave device according to the terms in the protocol specification. In depth information as well as the detailed MODBUS Specification can be found on the internet *www.modbus.org*.

The USBAMP-4 is compatible with most MODBUS compliant software available on the market, including commercial as well as free products (such as *Fenix Modbus* to name only one).

For further reading the *Protocol Specification* and the *Serial Line Protocol and Implementation* are recommended.

Prerequisites

For the USBAMP-4 to work it is required to install the mating serial driver. On many Windows- and Linux based systems the FTDI serial drivers may already be present and should work out of the box. The same is true for many but not all Android systems.

If these drivers are missing or pose problems a driver update is recommended. Please download the latest revision of FTDI's driver from their *VCP driver page* and install them according to their *Installation Guides*.

2. MODBUS Specifications

2.1. Connection Parameters

The serial port relating to the sensor must be opened with parameters given in table 1. The factory default is 115200 baud, 8 data bits, even parity, one stop bit and address 1.

Table 1: Virtual Serial Port Connection Parameters

Parameter	Default Value	Comment
Baud rate	115200	can not be changed
Bits and Parity	8E1	can not be changed
Sensor Address	1	can not be changed

For the USB based variant it is not possible to change any of these parameters.

2.2. Implemented Functions

From the MODBUS defined Function Codes only “Read Holding Registers” and “Write Holding Registers” for either a single or multiple registers at once are implemented in the device.

2.3. MODBUS Registers

2.3.1. Byte Order and Data Types

The devices use the LSB byte order for 16 Bit words and also for 32 Bit registers thus the lower significant byte or word is always placed first (Little Endian).

Table 2: List of all used data types

Abbreviation	Definition	Comment
uint16	Unsigned 16 Bit integer	Range 0...65535
int16	Signed 16 Bit Integer	Range -32767 … 32768
uint32	Unsigned 32 Bit integer	Range 0… 4294967295
float	IEE754 single precision Float	According to IEEE 754
string16	String of 16 Characters	Always consists of 8 consecutive registers of 16 Bit totaling 16 characters of 8 Bit each. Strings shorter than 16 characters end with an ASCII code of zero, characters behind this are not part of the actual string.

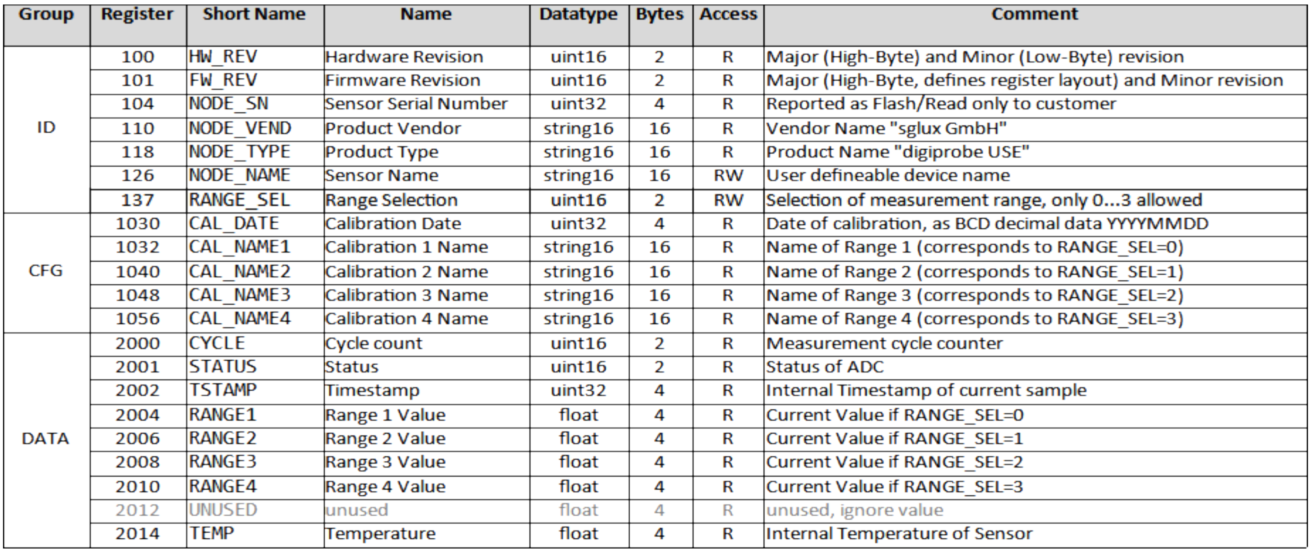
2.3.2. Register Mapping

In total these register ranges are defined:

- ID = Identification Data (readable, partially writable)
- CFG = Configuration Data (read only)
- DATA = Measurement Data (read only)

Register numbers (2nd column in table 3) are given as absolute numbers and not as offset within the holding register range.

Table 3: MODBUS Register Mapping (all numbers given are decimal numbers)



Group	Register	Short Name	Name	Datatype	Bytes	Access	Comment
ID	100	HW_REV	Hardware Revision	uint16	2	R	Major (High-Byte) and Minor (Low-Byte) revision
	101	FW_REV	Firmware Revision	uint16	2	R	Major (High-Byte, defines register layout) and Minor revision
	104	NODE_SN	Sensor Serial Number	uint32	4	R	Reported as Flash/Read only to customer
	110	NODE_VEND	Product Vendor	string16	16	R	Vendor Name "sglux GmbH"
	118	NODE_TYPE	Product Type	string16	16	R	Product Name "digiprobe USE"
	126	NODE_NAME	Sensor Name	string16	16	RW	User defineable device name
	137	RANGE_SEL	Range Selection	uint16	2	RW	Selection of measurement range, only 0...3 allowed
CFG	1030	CAL_DATE	Calibration Date	uint32	4	R	Date of calibration, as BCD decimal data YYYYMMDD
	1032	CAL_NAME1	Calibration 1 Name	string16	16	R	Name of Range 1 (corresponds to RANGE_SEL=0)
	1040	CAL_NAME2	Calibration 2 Name	string16	16	R	Name of Range 2 (corresponds to RANGE_SEL=1)
	1048	CAL_NAME3	Calibration 3 Name	string16	16	R	Name of Range 3 (corresponds to RANGE_SEL=2)
	1056	CAL_NAME4	Calibration 4 Name	string16	16	R	Name of Range 4 (corresponds to RANGE_SEL=3)
DATA	2000	CYCLE	Cycle count	uint16	2	R	Measurement cycle counter
	2001	STATUS	Status	uint16	2	R	Status of ADC
	2002	TSTAMP	Timestamp	uint32	4	R	Internal Timestamp of current sample
	2004	RANGE1	Range 1 Value	float	4	R	Current Value if RANGE_SEL=0
	2006	RANGE2	Range 2 Value	float	4	R	Current Value if RANGE_SEL=1
	2008	RANGE3	Range 3 Value	float	4	R	Current Value if RANGE_SEL=2
	2010	RANGE4	Range 4 Value	float	4	R	Current Value if RANGE_SEL=3
	2012	UNUSED	unused	float	4	R	unused, ignore value
	2014	TEMP	Temperature	float	4	R	Internal Temperature of Sensor

In normal operation it is only required to continuously read the Measurement Data registers in a row.

The registers in the groups ID and CFG do not change and need to be read only once, with the exception of the RANGE_SEL register which is used to request the current measurement range or to change the measurement range.

3. Application Information

3.1. General Hints

- The USBAMP-4 does not require any configuration by the user.
- If connected to a host it uses the standard FTDI VCP driver (exists on most systems) and installs a virtual serial port.
- The D2XX.DLL from FTDI may be used to identify the device in detail (see our examples). basically it installs as USB device with hardware ID VID_0403 and PID_6015.
- The names of the stored Ranges as well as the Product Name and Device Name should be read each with one multiple holding register read request (of 8 registers). Contained strings end with a zero-byte if they are shorter than 16 bytes.
- The measurement range in use can be read or set from Register 137. It is immediately active after a write operation. **The range parameter must have a value in the range 0-3 and must be known before taking a measurement – it gives the index to the correct result register. For range 0 the result has to be taken from register 2004 and for range 3 from register 2010.**
- The value of register 137 is not preserved after power off or disconnection. After power up (USB connection) it will always start with a value of 0.
- It may be required to drop some readings after switching the range, but this depends on multiple influences and must be checked for each specific application.
- The registers 2000...2014 should be read with a **multiple holding register read request** to ensure Timestamp and Cycle Count really relate to the same reading.
- The cycle count register can be used to avoid reading the same sample twice in case the data is requested too fast. The sensor generates samples with a rate of ~14 Hz. So in case your polling cycle is near to or smaller as 1/14th of a second you must check this register for an increment and reject any reading if it has not incremented yet.
- A polling cycle below 50 ms is not recommended.
- The polling routine can implement a timeout to detect loss of connectivity. The timeout typically is in the 2...10 second range. Nevertheless the sensor typically responds within 1/14th of a second or faster.
- The status Register can be used to detect sensor saturation (value over range)
    - Zero means everything is alright.
    - One denotes a negative reading which may indicate sub-optimal offset calibration or severe noise from the surrounding. In principal the sensor reading should be interpreted as being zero in this case!
    - 2 and 3 denote saturation events and applications should reject such readings. If possible switch to a less sensitive range. If the least sensitive range is already in use the input current is beyond the range of the device.

3.2. Typical Procedure

1. Detect serial port (by using USB ID “PID_0403&VID_6015”, optional check the USB serial number to start with “SGCD4”)
2. Open the serial port and read ID or CFG registers as needed. sglux recommends to check at least the sensor serial number. Furthermore it may be useful to also check the product vendor and product type strings to make sure it is the right and correct device. If it needs even more strictly also the calibration date and sensor name could be checked.
3. Read or write the RANGE_SEL register according to your application
4. Read DATA registers 2000-2015 with one request. The result has to be taken from register (2004 + RANGE_SEL)
5. Check if the value of register CYCLE has increased by 1 (or changed, at least), if not, repeat at 4.
6. Check the value of register STATUS. If an over-range condition is indicated a larger measurement range must be selected by changing register RANGE_SEL. If that is not possible, the input current is too large and can not be measured (is out of range).
7. process the measurement, repeat measurement if required
8. close serial port

3.3. Labview Applications

National Instruments offers two free implementations:

- the National Instruments MODBUS-API which has been tested by us and is working well. *Download and further information* is available on their website.
- They also offer an *older MODBUS Implementation* which is working but it offers lower performance and compatibility.
- Further Labview code is only provided on request. Please give a short description of your project to guarantee a good match.