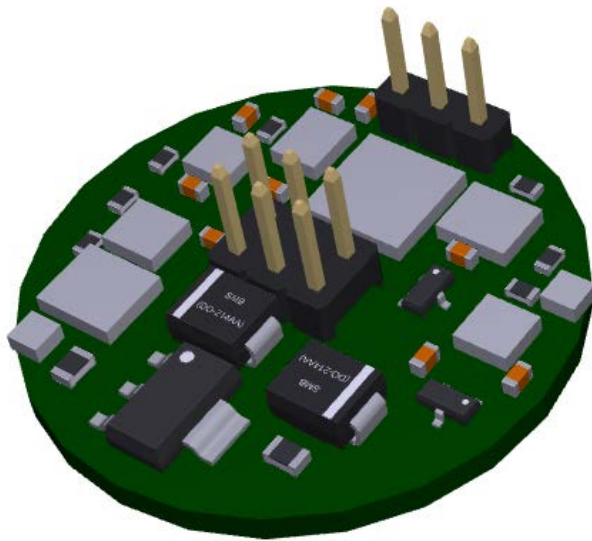
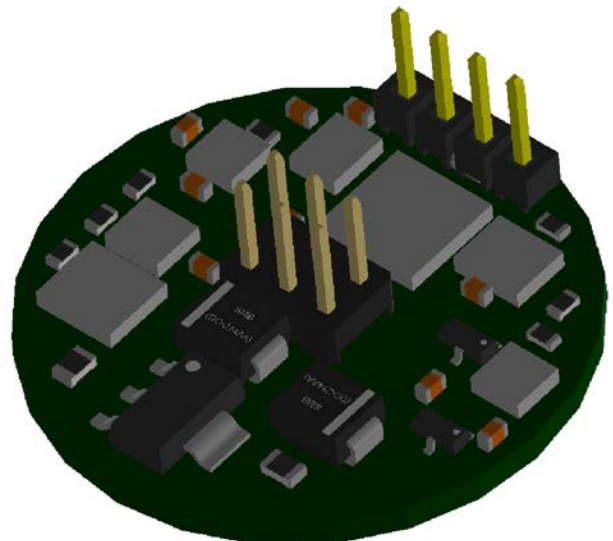


TX1 Sensor Controller Manual

Revision G



REVISION 2



REVISION 3



CO2METER.COM
CO2 MEASUREMENT SPECIALISTS

The information in this document is protected under applicable federal law as an unpublished work and is confidential and proprietary to CO2Meter.com. Its use, disclosure, reproduction, or publication, in whole or in part, without the express prior written consent of CO2Meter.com. is prohibited.

© 2019 CO2Meter.com. All Rights Reserved.

Document History

Date	Revision	Description
30 November 2018	Draft A	Initial Draft
1 December 2018	Draft A	Revised Draft
2 December 2018	Draft A	Added Appendix
15 December 2018	Rev. A	Firmware 1 build 0, clarification and detail; HR, IR and CMD
1 February 2019	Rev. B	Firmware 1.10 beta Hardware version 3.0: Addition of GSS CO2 capability. Dual use IR 2-5 and 16 - 20. Sensor restore type table. Commands 34, 40, 4. Support GSS CO2 sensors to 20Hz via Modbus
12 February 2019	Rev. C	Firmware 1.20 beta. Add limiter for 20 mA out, < = register 23 value. Test value register 19 limited < = register 19 prevent overrange out on miss configuration.
19 February 2019	Rev. C	Firmware 1.20 reference to CO2Meter and GasBox development tools. Confirm CO2/O2 version hardware 1.0, 2.0 and 3.0 compatible with firmware 1.2.
6 March 2019	Rev. D	Added Appendix with Revision History. Moved Version 1.1 Notes to Revision History. Added Version 1.2 Revision
8 March 2019	Rev. E	Firmware 1.20. Grammatical corrections, technical clarifications.
11 March 2012	Reve F	Firmware 1.20 clarity execution of Commands.

10 June 2019	Rev. G	Hardware REV3 release. Firmware 1.20. additional detail for compatibility w/ MX series. Elaboration in register definitions. Fix annotation of RS485 AB swapped.
---------------------	--------	--

Table of Contents

INTRODUCTION.....	1
ABOUT THIS MANUAL.....	1
SOFTWARE VERSION.....	1
CHAPTER 1 : OVERVIEW.....	2
GENERAL.....	2
MEASUREMENTS	2
CHAPTER 2 : COMMUNICATIONS INTERFACE.....	3
GENERAL.....	3
PHYSICAL CONNECTIONS	3
CONNECTOR ASSIGNMENTS.....	3
SENSOR CONNECTIONS.....	4
CHAPTER 3 : REGISTERS	5
INPUT REGISTERS.....	5
INPUT REGISTER DESCRIPTIONS	6
<i>IR30001 (Address 0) Busy/Error Status.....</i>	<i>6</i>
<i>IR30003 (Address 2) O2/ CO2 Concentration.....</i>	<i>7</i>
<i>IR30004 (Address 3) Partial Pressure O2.....</i>	<i>7</i>
<i>IR30005 (Address 4) Temperature</i>	<i>7</i>
<i>IR30006 (Address 5) Pressure.....</i>	<i>7</i>
<i>IR30007 (Address 6) Error Status.....</i>	<i>7</i>
<i>IR30008 (Address 7) Manufacturer Id</i>	<i>7</i>
<i>IR30009 (Address 8) Model Number.....</i>	<i>7</i>
<i>IR30010 (Address 9) Firmware Version.....</i>	<i>7</i>
<i>IR30011 (Address 10) Serial Number.....</i>	<i>7</i>
<i>IR30012 (Address 11) PDU Count</i>	<i>7</i>
HOLDING REGISTERS.....	7
HOLDING REGISTER DESCRIPTIONS.....	9
<i>HR40001 (Address 0) Checksum.....</i>	<i>9</i>
<i>HR40007 (Address 6) Gas Type</i>	<i>9</i>
<i>Gas type preset table.....</i>	<i>9</i>
<i>HR40016 ADR 15 Modbus Address</i>	<i>9</i>
<i>HR40017 ADR 16 Comm configuration (stop and parity).....</i>	<i>10</i>
<i>HR00020 ADR 19 Current loop test</i>	<i>10</i>
CHAPTER 4 DEVICE COMMANDS	11
GENERAL.....	11
COMMAND AND PARAMETER REGISTERS	11
COMMAND BUSY.....	11
COMMAND EXECUTION	11
COMMAND DESCRIPTIONS.....	11
<i>Device Command 1 (0x0001) Error Reset</i>	<i>11</i>
<i>Device Command 2 (0x0002) Factory Use.....</i>	<i>12</i>
<i>Device Command 16 (0x0010) Save Holding Registers to Flash.....</i>	<i>12</i>
<i>Device Command 17 (0x0011) Restore Factory Presets</i>	<i>12</i>
<i>Device Command 32 (0x0020) Zero Calibration</i>	<i>12</i>
<i>Device Command 33 (0x0021) Calibrate to Span</i>	<i>12</i>

<i>Device Command 34 (0x0022) calibrate to 400 ppm CO2</i>	12
<i>Device Command 40 (0x0028) suspend streaming</i>	12
<i>Device Command 41 (0x0029) resume streaming</i>	12
<i>Device Command 9985 (0x2701) Factory Use</i>	12
CHAPTER 5 : TESTING	13
MODBUS TOOL.....	13
SERIAL PORT.....	13
CABLE CONNECTIONS.....	13
POWER CONNECTIONS.....	14
CONFIGURING MODBUS TOOL.....	14
USING THE MODBUS TOOL.....	14
COMMUNICATION LOG.....	14
READING HOLDING REGISTERS.....	15
READING INPUT REGISTERS.....	15
WRITING HOLDING REGISTERS.....	15
CHAPTER 6 CO2Meter GRAPHICAL CONFIGURATION AND RAPID DEPLOYMENT TOOL	18
CHAPTER 7 FIRMWARE UPDATES	20
GENERAL.....	20
PROGRAMMING CONFIGURATION.....	20
<i>Processor</i>	20
<i>Programming Interface</i>	20
<i>Programming Voltage</i>	20
<i>Device Power</i>	20
<i>Connections</i>	20
PROGRAMMING DEVICE.....	20
PROGRAMMING SOFTWARE.....	22
FIRMWARE FILES.....	22
PROGRAMMING PROCEDURE.....	22
CONNECT TO DEVICE.....	23
DEVICE ERASE.....	24
PROGRAMMING FIRMWARE.....	24
CHAPTER 8 : SUPPORT	26
WARRANTY.....	26
LIABILITY.....	26
RETURNS.....	26
CONTACT US.....	27
CHAPTER 9 : APPENDIX 1: REVISION HISTORY	28
VERSION 1.1 REVISIONS:.....	28
VERSION 1.2 REVISIONS:.....	29

Table of Figures

Figure 2-1 TX1 Sensor Controller.....	3
Figure 2-2 Sensor controller sensor mount side.....	4
Figure 5-1 Holding Registers.....	16
Figure 5-2 Input Registers.....	17

Figure 7-1 Programming Connections	21
Figure 7-2 ST-LINK/V2 Connections.....	21
Figure 7-3 Initial Screen.....	23
Figure 7-4 Screen After Connection.....	24
Figure 7-5 Erase and Programming Screen.....	25

Introduction

About this Manual

This document is a technical reference for the CO2Meter.com TX1 Sensor Controller. It provides the information necessary to use the sensor controller as well as information about configuring the controller for various applications. The sensor current supports the SST Luminox O2 sensors and the GSS CO2 MINIR sensors.

Software Version

This manual applies to Software Version 1.2 beta 2/12/2019

Chapter 1 : Overview

General

The TX1 Sensor Controller is a high-performance controller supporting various types of gas sensor modules. The controller supplies an RS485 Modbus RTU interface to the host system and a separate UART interface to the gas sensor module. The controller accepts a wide range (6 to 24Volt) power supply and provides a 4-20mA 3-wire current output. The controller is 24 volt and provides RS485 and 4-20 mA connection, used in a variety of industrial applications. The O2 Transmitter is UV and has zero cross to any of the mentioned gases.

Measurements

The TX1 receives continuous measurements streamed from the sensor, typically twice per second. The controller performs various computations at each measurement, including adjusting the raw controller reading to the field calibrated values. The measurements are made available to a host system through the Modbus interface. The primary gas sensor reading is made available as a 4-20ma current output, 24 volt and RS485 for industrial applications and many other industries.

Chapter 2 : Communications Interface

General

The TX1 communicates with a host system through an RS485 Modbus RTU interface. This interface allows many slave devices to be connected to a host controller over a single twisted cable. A common ground reference is normally required. For long lines, the twisted-pair should be terminated with a 120-ohm resistor at each end.

Physical Connections

The TX1 uses a 6 pin (2x3) header with 0.1-inch (2.54 mm) centers. In some cases, the header is not populated, and appropriate wire terminals are soldered directly to the header pads.

Connector Assignments

Pin	Function	Pin	Function
1	4-20 mA current output	2	Reserved
3	GND	4	RS485 A (Data -)
5	DC Power (+6 - 24V)	6	RS485 B (Data +)

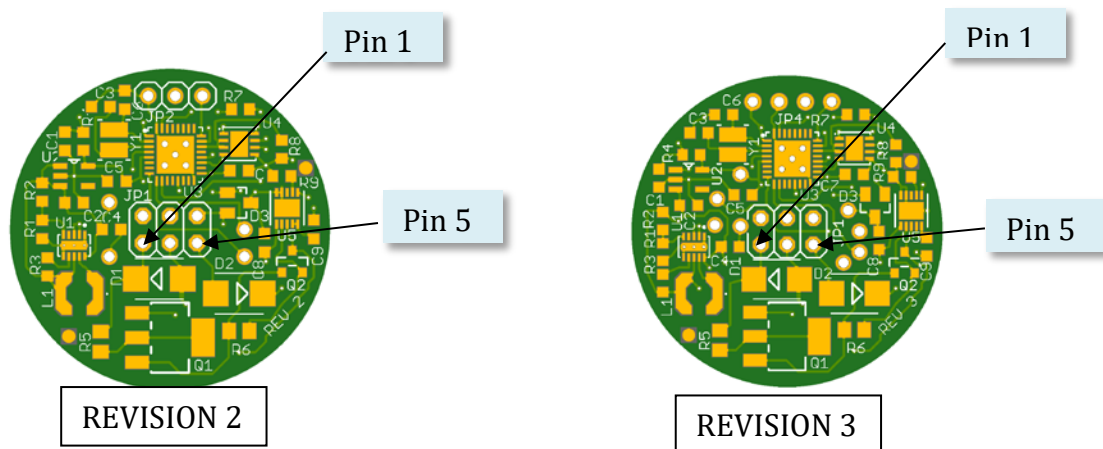


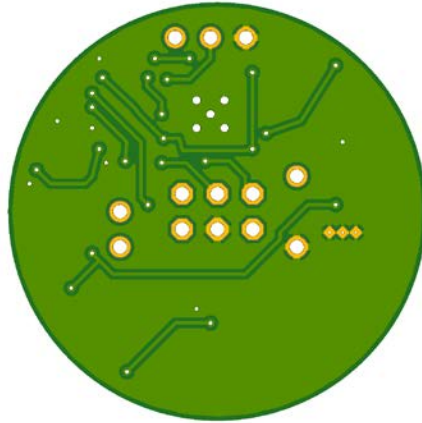
Figure 2-1 TX1 Sensor Controller

The DC Power and GND connections are protected against reverse polarity. The RS485 Data lines are protected against high voltage transients. The 4-20 mA output pin is also protected against transient voltages. There is no protection on reserved Pin 2 which connects directly to the microcontroller.

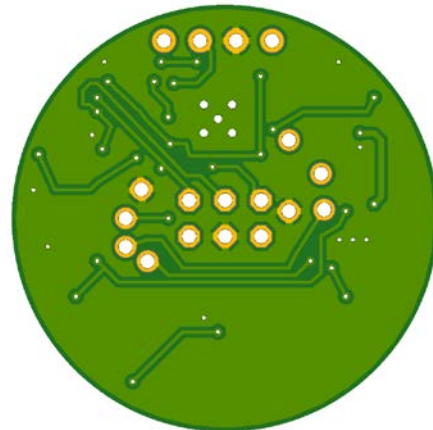
WARNING: Use care when connecting DC power. While the DC power and Ground connections are protected against reverse polarity, all other pins on the connector will be damaged if DC voltages > 5 V are continuously applied.

Sensor Connections

The Luminox sensor is installed centered on the solder side of the TX1 PCB. The sensor is normally soldered to the board. 5V power is supplied to the sensor from a switching power supply.



REVISION 2



REVISION 3

Figure 2-2 Sensor controller sensor mount side

Chapter 3 : Registers

The TX1 device provides a total of 32 Input Registers and 32 Holding Register that may be accessed through the Modbus RTU protocol. General information on the protocol as used on CO2Meter Inc. devices can be found in the RS485 Modbus RTU Interface manual. The following descriptions are specific to the register assignments and commands supported by the TX1.

Note that the register numbers are as defined in the Modbus Specifications. Some PLC controllers will reference these numbers. In the Modbus Specifications the Input Registers are numbered from 30001 to 30999. These correspond to the addresses 0 to 998 used with the “Read Input Register” (4) function. Similarly, the holding registers are numbered from 40001 to 40999. These correspond to the addresses of 0 to 998 used with the “Read Holding Register” (6) and “Write Multiple hold Registers” (16) commands. The TX1 currently uses addresses in the range 0 to 31. Addresses outside this range result in a Modbus error (Illegal Data Address).

Note also that only the Holding Registers can be written (06) from the host. Input Registers are read/only (4).

Input Registers

The TX1 supports 32 16-bit input registers. These registers are maintained by the controller during operation and can be read at any time using. The “int” type is a 16-bit signed quantity. The “uint” type is a 16-bit unsigned. The float type is a 32-bit IEEE754 formatted number. For floating point values, the lower numbered register contains the 16-bit word with the exponent and upper bits of the significant. The higher numbered register contains the remaining bits of the significant.

The status bits are described separately in a following section.

Table 3-1 Input Registers (Read Only)

Register	Address	Type	Description
IR30001	0	int	Busy/Error Status (see below)
IR30002	1	int	Reserved
IR30003	2	int	O2 Concentration in 10's of ppm/ CO2 filtered concentration.
IR30004	3	int	O2 Partial Pressure in tenths of millibars/ CO2 Raw unfiltered unscaled
IR30005	4	int	Sensor Internal Temperature; O2 in 10ths of a degree C/ CO2 in 10ths of a degree C
IR30006	5	int	O2 Sensor Pressure in millibars, CO2 sensor Unsigned RH in percentage.
IR30007	6	uint	Error Status received from Sensor
IR30008	7	uint	Manufacturer ID Code CO2Meter=35642

Register	Address	Type	Description
IR30009	8	uint	Model Number 1=TXZ 2=MX2 3=MX3 4=EC
IR30010	9	uint	Firmware Version high byte version low byte build i.e., 257 = 1.1
IR30011	10	uint	Controller Serial Number
IR30012	11	uint	Received PDU Count
IR30013	12	uint	Received PDU Errors
IR30014	13	uint	Transmit PDU Count
IR30015	14	uint	Count of Received Measurements from Sensor
IR30016	15	int	Raw %O2 as received from Sensor. NA CO2.
IR30017	16	float	Floating Point % O2/ filtered CO2 %
IR30018	17		
IR30019	18	float	Floating Point Partial Pressure O2/ unfiltered CO2 %
IR30020	19		
IR30021	20	float	O2 Floating Point Temperature/CO2 Temperature
IR30022	21		
IR30023	22	float	O2 Floating Point Pressure/ CO2 Relative Humidity %
IR30024	23		
IR30025	24	uint	BME1 TEMP Reserved
IR30026	25	uint	BME1 RH Reserved
IR30027	26	uint	BME1 Baro Reserved
IR30028	27	unit	BME2 TEMP Reserved
IR30029	28	unit	BME2 RH Reserved
IR30030	29	unit	BME2 Baro Reserved
IR30031	30	-	Reserved
IR30032	31	-	Current DAC value to current loop (0 - 4095) as calculated from IR30003

Input Register Descriptions

IR30001 (Address 0) Busy/Error Status

This register provides the current Busy/Error status of the device. The bit assignments are as follows:

- Bit 15 (MSB) Busy.
- Bits 4-14 Reserved.
- Bit 3 Command Failure.
- Bit 2 Flash Error.
- Bit 1 Invalid Command.
- Bit 0 (LSB) Power Fail. (Set at Power Up)

All bits can be cleared by executing the “Error Reset” device command.

IR30003 (Address 2) O2/ CO2 Concentration

This register provides the current concentration of O2 in 10's of ppm/Filter Co2 concentration un scaled %

IR30004 (Address 3) Partial Pressure O2

This register provides the current partial pressure of O2 in tenths of a millibar. / un filtered CO2 un scaled %

IR30005 (Address 4) Temperature

This register provides the current temperature in tenths of a Degree C Oxygen and CO2 sensors.

IR30006 (Address 5) Pressure

Barometric pressure from O2 sensor Barometer/ CO2 sensor if equipped % Relative humidity.

IR30007 (Address 6) Error Status

This register indicates an error status received from the sensor.

IR30008 (Address 7) Manufacturer Id

This register provides a manufacturer ID. All cCO2Meter Inc. devices will have 35642

IR30009 (Address 8) Model Number

This register provides a model number. TX1 devices have a value of 1.

IR30010 (Address 9) Firmware Version

This register indicates the firmware level of the device high byte (MSB) = version and low byte (LSB) = revision. 257 = version 1 revision 1 (Rev 1.1)

IR30011 (Address 10) Serial Number

This register provides the serial number of the controller. A value of 65535 indicates that a serial number has not yet been assigned.

IR30012 (Address 11) PDU Count

This register contains the total number of packets received by the device since power up.

Holding Registers

There are 32 Holding Registers. The Holding Registers are read hold register (03), write hold register (06) and write multiple Hold registers (16). They are initially loaded from non-volatile memory at power up. They may then be altered by the host. Note that holding registers are *not* automatically saved to non-volatile memory. Registers are saved to non-volatile memory only as a result of a specific write command.

Table 3-2 Holding Registers (Read/Write)

Reg	Address	Default	Description
HR40001	0	N/A	Checksum Value for FLASH Storage
HR40002	1	0	Reserved MX/EC output mask
HR40003	2	0	Reserved MX/EC logging Mask
HR40004	3	0	Reserved EC configuration
HR40005	4	5	Reserved configuration options MX/EC. !- RS485 base address 0-31, PWM.
HR40006	5	0	Reserved stream interval
HR40007	6	2	T & MX Gas Type see table, 1=CO2 2=O2
HR40008	7	0	O2 Zero Value, reported in zero calibration to N2
HR40009	8	0	O2 K Value slope of calibration K/32768 applied after subtracting 0 value.
HR40010	9	0	Calibration Concentration, value used in most recent calibration.
HR40011	10	25000	Span- not used in this application place holder for software compatibility.
HR40012	11	25000	Full Scale value for 4-20 mA current output. O2 20mA=25%. CO2 10000 unscaled input = 100% CO2
HR40013	12	10	Multiplier to get to PPM
HR40014	13	0	Reserved EC feature
HR40015	14	5865	Reserved PWM time base 1-16384. 0=342 mS0.1125 graduations. 5865=1 Sec.
HR40016	15	21	Modbus Address 1-247. 254 for set-up and discovery. Default 21 0x15
HR40017	16	0	Comm Configuration (Stop bits and Parity)
HR40018	17	8	Baud Rate (Multiple of 1200) 8 = 9600
HR40019	18	0	Reserved
HR40020	19	0	4-20 mA current output test (when non-zero) range 0-4096.
HR40021	20	0	Reserved
HR40022	21	550	Current output 4 mA Level, ADC Value corresponding to 4 mA current output.
HR40023	22	2740	Current output 20 mA Level, ADC Value corresponding to 20 mA current output.
HR40024	23	0	Reserved
HR40025	24	0	Reserved
HR40026	25	0	Reserved
HR40027	26	0	Reserved
HR40028	27	0	Reserved
HR40029	28	0	Reserved
HR40030	29	0	Reserved
HR40031	30	0	Command Parameter

Reg	Address	Default	Description
HR40032	31	0	Command

Holding Register Descriptions

HR40001 (Address 0) Checksum

This register contains the checksum of the initial holding register values as read from non-volatile memory. This register may be freely written from the host. When holding register content is saved to non-volatile storage, a new checksum value will be computed and written to non-volatile memory.

HR40007 (Address 6) Gas Type

This register contains a Value that indicates the type of Gas sensed by the sensor. Setting the method for communication and computations. 1=CO2 2=O2.

Gas type preset table

Sensor:	Label	1=CO2, 2=O2	Zero	K factor	Cal Val	Range	4-20	scaler
Type 0:	Luminox 25%	2	0	32768	20900	25000	25000	10
Type 1:	Luminox 50%	2	0	32768	20900	50000	50000	10
Type 2:	GSS 1-2%	1	0	32768	10000	10000	1000	1
Type 3:	GSS 5%	1	0	32768	5000	5000	5000	10
Type 4:	GSS 20%	1	0	32768	10000	20000	20000	10
Type 5:	GSS 60%	1	0	32768	10000	60000	60000	10
Type 6:	GSS 100%	1	0	32768	10000	10000	10000	100

HR40016 ADR 15 Modbus Address.

The Default address for TX transmitter boards is 21. Sensors will respond to the configured address or to the address 0xFE (254) allow for the connection to a sensor of a unknown sensor. Note that the address returned in the message is the address that is used by the host. This is necessary to comply with the protocol rules. Note also address 0xFE is outside the range of legal addresses.

Modbus broadcast address 0 is not supported at this time. Using address 0 will generate an error: invalid command.

HR40017 ADR 16 Comm configuration (stop and parity)

The low order three bits define the Comm Configuration.

Bit 0 (LSB) is 0 for 1 stop bit (default) and 1 for two stop bits.

Bit 1 0 = no parity, 1 = parity.

Bit 2 0 = even parity, 1 = odd parity.

NOTE: These are not implemented in Ver 1 Build 0.

HR00020 ADR 19 Current loop test.

When >0<4096 will cause 4-20 mA output to operate at that level for trim test. Used to establish level for 4 and 20 mA current output levels: HR 22-23

Chapter 4 Device Commands

General

Device commands initiate operations in response to messages from the host. Note that Device commands are not defined in the Modbus Specifications but are specific to the device (TX1).

Commands are used for operations such as calibration, error handling, and saving holding register content to non-volatile storage. Commands are also used for certain factory functions associated with the manufacturing and testing of the TX boards.

Command and parameter registers

Commands are written to Holding Register 40032 (Address 0x1F) parameters for commands are written to Hold Register 40031 (Address 0x1E). A parameter if used and the command are written in the same Modbus message.

Modbus function code 16 (0x10), Write Multiple Holding Registers, allows the parameter and command to be sent in a single PDU.

Command Busy

A command can only be sent to the device when it is not Busy. The Busy indication is provided as the most significant bit in the Status Register (Input Register 30001 at address 0).

Command Execution

Commands are executed as soon as they are received. If a command is not valid for the device then, one of the Command Error bits will be set in the status register. Error bits are only valid when the status indicates not busy.

Holding Register 40032 (Address 0x1F) is automatically cleared when the command is complete. Holding Register 40031 (Address 0x1E) is also cleared if the command was successful. If an error occurred, Holding Register 40031 (address 0x1E) may contain an additional error code.

Note that some commands execute immediately, never posting a busy state. The error status is always valid for the last command executed, Input Register 30001 address 0x00, and if read, the Hold register 40032 (address 0x31) will be cleared indicating the command executed.

Command Descriptions

Device Command 1 (0x0001) Error Reset

This command resets all error bits. Note that certain hardware errors may cause bits to be set again.

Device Command 2 (0x0002) Factory Use

This command is reserved for factory use.

Device Command 16 (0x0010) Save Holding Registers to Flash

This command saves the holding registers to Flash.

Device Command 17 (0x0011) Restore Factory Presets

This command will restore the factory presets to the holding registers. There are 7 groups of presets numbered 0 to 6. The command parameter indicates which group should be restored. Note that the retrieved values are not automatically saved to flash. To save the registers to flash, a separate Device Command 16 is required.

Device Command 32 (0x0020) Zero Calibration

This command sets the zero value for the sensor. The sensor should be shown 100% nitrogen for approximately 1 minute prior to issuing this command. Holding registers are automatically saved upon execution of this command.

Device Command 33 (0x0021) Calibrate to Span

This command calibrates the sensor based on the known concentration provided in the command parameter. The concentration is expressed in 10's of ppm. For example, for 20.9% the value should be 20900. All holding registers are automatically saved upon execution of this command.

Device Command 34 (0x0022) calibrate to 400 ppm CO2

Calibrate CO2 sensor to 400 PPM ambient, use the GSS "G" command.

Device Command 40 (0x0028) suspend streaming

Suspend streaming, prior to sending the calibrate command to CO2 sensor.

Device Command 41 (0x0029) resume streaming.

To resume streaming, send after sending the calibration command.

Device Command 9985 (0x2701) Factory Use

This command is reserved for factory use.

Chapter 5 : Testing

WARNING: These instructions are intended for engineering personnel familiar with the RS485 Modbus protocol.

Modbus Tool

The RS485 Modbus RTU protocol is a packet-oriented protocol. All the characters of the packet must be sent in a single burst with no gaps and a valid two-byte CRC. The absence of a character for 2.5-character times indicates the end of a packet.

The slave device will not react to a packet unless the CRC is correct, and the address field is that of the device. Due to these characteristics, it is not practical to test device operation from a terminal emulator program.

Fortunately, there are several tools available for testing Modbus RTU devices. The following sections describe the use of one of these tools, Modbus Tool.

Modbus Tool is a free and open source Modbus Protocol testing program available as a free download. The tool can be downloaded from:

<https://github.com/graham22/ModbusTool/blob/master/Setup/ModbusTool.msi>

This windows installer package includes two programs. One program, “Modbus Master” is used to simulate a host and a second program “Modbus Slave” is used to simulate a device. The following instructions will use only the Modbus Master program.

Reference RS485 Modbus RTU Interface Manual for further information.

Serial Port

Modbus tool, as well as most other PC based tools for Modbus RTU devices, requires a serial RS485 port that appears as a com port on the local system. FTDI manufactures a smart cable well suited for this purpose. The manufacturers part number is USB-RS485-WE-1800-BT. This cable is available from Digi-key, Mouser, and other electronic distributors. A version with a longer cable is also available.

Drivers for FTDI cables are pre-installed on most Windows systems. The latest drivers are also available from FTDI at the following link:

<https://www.ftdichip.com/FTDrivers.htm>

Cable Connections

The FTDI cables are color coded. The labeling on the TX1 uses “B Data +” and “A Data -” conforming to the definitions of A and B in the RS485 standard. The definitions on the FTDI cable are correct for the Data Polarity but opposite for A and B.

The yellow FTDI wire should be connected to Pin 6 of the TX1. The orange wire should be connected to Pin 4. Note that, in general, the Data+ and Data- labeling of devices is

consistent across most RS485 devices. The labelling for A and B is often opposite of that of the standard (Due to a different labeling convention used by manufacturers of the semiconductor interface devices).

Power Connections

The TX1 must be powered for proper operation. The DC power supply input is specified from 6V to 24V. Many devices will operate with a power supply set as low as 5V. If the 4-20 mA current output is not used, the TX1 device may be powered from the 5V USB available on the FTDI connector. Note, however, that this is not recommended.

WARNING: Use care when connecting DC power. While the DC power and Ground connections are protected against reverse polarity, all other pins on the connector will be damaged if DC power is applied.

Configuring Modbus Tool

Start the Modbus Master tool. A graphical interface will appear like Figure 5-1 Holding Registers.

The communication mode should be set to RTU. Select the COM port of your FTDI RS485 cable (or other RS485 device). The other communication settings should match the figure.

Before connecting, set the slave ID to 21. This is the default Modbus address of CO2Meter.com devices. Alternatively, you can set the ID to 254. All CO2Meter.com devices will respond to address 254, regardless of their configuration. Useful for operating/interrogating unknown address boards, do note that only one board may be connected at time.

Using the Modbus Tool

Press the connect button. This will connect the program to the FTDI smart cable. If the TX1 device is powered and the RS485 connections are correctly made, the Modbus Master tool will be able to issue commands to the device. The red and green led lights on the FTDI cable indicate transmit and receive of data.

The Modbus Master tool allows you to specify a starting address and a number of registers. As the TX1 has 32 registers in both the Holding Register and Input Register ranges, set the Start Address to 0 and the size to 32. You must hit the “Apply” button after changing any of these values.

The TX1 supports the commands; Reading Holding Registers, Read Input Registers, Write Single Register and Write Multiple Registers. The other functions are not supported.

Communication Log

Modbus Master provides a log of the TX and RX communication activity in the lower portion of the screen. This provides the hexadecimal values of the byte sent and received.

Reading Holding Registers

Begin by pressing the “Read holding register” button. If everything is operating correctly, the address 0-31 display should be populated similar to what is shown in figure 6-1.

Reading Input Registers

The Input Registers can be read by pressing the “Read input register” button. If everything is operating correctly, the address 0-31 should be populated similar to what is shown in figure 6-2. The values will differ than those in figure 6-2.

Writing Holding Registers

The TX1 supports both the Write Single Register and Write Multiple Register functions. When using the Modbus Master tool is usually easiest to use the Write Multiple Register function. This allows you to leave the starting address and size at 32.

Quickest way to work with the Hold Registers is to perform a “Read Holding Registers” to get the current value of registers. Modify the value of interest and press the “Write Multiple Registers” button. This will update all 32 registers.

Figure 5-1 Holding Registers

The screenshot shows the Modbus Master software interface. The 'Communication' section is set to RTU mode with Port COM8, Baud 9600, Parity None, Data Bits 8, and Stop Bits 1. The 'Display Format' is set to Integer. The 'Functions' section has 'Read holding register' selected. The 'Slave ID' is 21. Below these settings is a table of holding registers.

Start Address	Size	Value
0	12	21930
1	13	0
2	14	0
3	15	0
4	16	0
5	17	0
6	18	2
7	19	0
8	20	32768
9	21	20900
10	22	25000
11	23	25000

The 'Communication Log' at the bottom shows the following messages:

```

>5:31:01 PM: RX: 15 03 40 55 aa 00 00 00 00 00 00 00 00 00 02 00 00 80 00 51 a4 61 a8 61 a8 00 0a 00 00 00 00 15 00 00 00 08 00 00 00 80 00 03 20 0f a0 01
>5:31:01 PM: Read succeeded: Function code:3.
>5:58:33 PM: TX: 15 04 00 00 00 20 f2 c6
>5:58:33 PM: RX: 15 04 40 00 00 00 00 4e 8e 07 fb 01 12 03 f8 00 00 8b 3a 00 01 00 03 ff 0f 0c 00 00 00 00 08 30 4e 8e 1c 00 46 9d 60 00 44 ff 33 33 41 db 00 00 44 7e
>5:58:33 PM: Read succeeded: Function code:4.
>8:45:31 PM: TX: 15 03 00 00 00 20 47 06
>8:45:31 PM: RX: 15 03 40 55 aa 00 00 00 00 00 00 00 00 02 00 00 80 00 51 a4 61 a8 61 a8 00 0a 00 00 00 00 15 00 00 00 08 00 00 00 80 00 03 20 0f a0 01
>8:45:31 PM: Read succeeded: Function code:3.
  
```

Figure 5-2 Input Registers

The screenshot shows the Modbus Master software interface. At the top, there are communication settings for TCP (Port: 502, IP Address: 127.0.0.1) and RTU (Port Name: COM8, Baud: 9600, Parity: None, Data Bits: 8 Bits, Stop Bits: 1 Bit). Below this are display format options (LED, Binary, Hex, Integer) and function buttons (Read coils, Read holding register, Write single coil, Write multiple coils, Read discrete, Read input register, Write single register, Write multiple register). A Slave ID field is set to 21. The main area displays a table of input registers with Start Address 0 and Size 32. The table shows 12 rows of registers, each with 10 columns of values. The Communication Log at the bottom shows a successful read operation at 5:58:33 PM.

Communication

Mode: TCP UDP RTU

TCP: Port: 502, IP Address: 127.0.0.1

RTU: Port Name = COM8, Baud = 9600, Parity = None, Data Bits = 8 Bits, Stop Bits = 1 Bit

Display Format

LED Binary Hex Integer

Functions

Read coils, Read holding register, Write single coil, Write multiple coils, Read discrete, Read input register, Write single register, Write multiple register

Slave ID: 21

Input Registers Table

Register	0	12	24	36	48	60	72	84	96
0	0	0	0	0	0	0	0	0	0
1	0	13	0	25	0	37	0	49	0
2	20110	14	2096	26	0	38	0	50	0
3	2043	15	20110	27	0	39	0	51	0
4	274	16	7168	28	0	40	0	52	0
5	1016	17	18077	29	0	41	0	53	0
6	0	18	24576	30	0	42	0	54	0
7	35642	19	17663	31	0	43	0	55	0
8	1	20	13107	32	0	44	0	56	0
9	3	21	16859	33	0	45	0	57	0
10	65535	22	0	34	0	46	0	58	0
11	12	23	17534	35	0	47	0	59	0

Communication Log

```

5:29:48 PM: RX: 15 04 40 00 00 00 00 4e a2 07 fb 01 10 03 f7 00 00 8b 3a 00 01 00 03 ff 00 0a 00 00 00 01 49 4e a2 44 00 46 9d 60 00 44 ff 99 9a 41 d9 c0 00 44 7d
5:29:48 PM: Read succeeded: Function code:4.
5:31:01 PM: TX: 15 03 00 00 00 20 47 06
5:31:01 PM: RX: 15 03 40 55 aa 00 00 00 00 00 00 00 02 00 00 80 00 51 a4 61 a8 61 a8 00 0a 00 00 00 00 15 00 00 00 08 00 00 00 80 00 03 20 0f a0 00
5:31:01 PM: Read succeeded: Function code:3.
5:58:33 PM: TX: 15 04 00 00 00 20 f2 c6
5:58:33 PM: RX: 15 04 40 00 00 00 00 4e 8e 07 fb 01 12 03 f8 00 00 8b 3a 00 01 00 03 ff 00 0c 00 00 00 00 08 30 4e 8e 1c 00 46 9d 60 00 44 ff 33 33 41 db 00 00 44 7d
5:58:33 PM: Read succeeded: Function code:4.
    
```

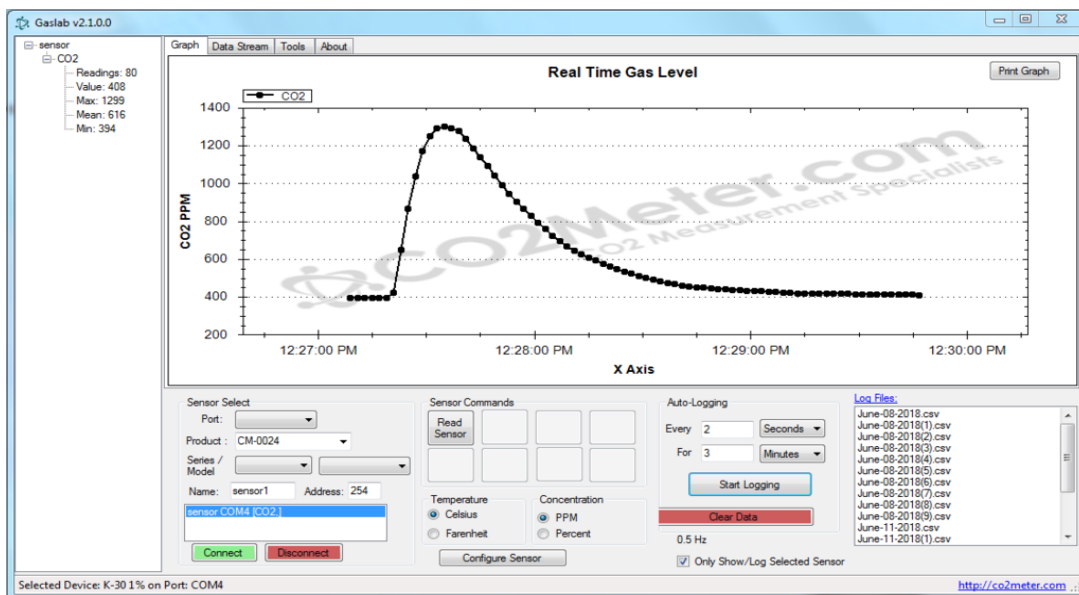
Chapter 6 CO2Meter Graphical configuration and Rapid deployment tool

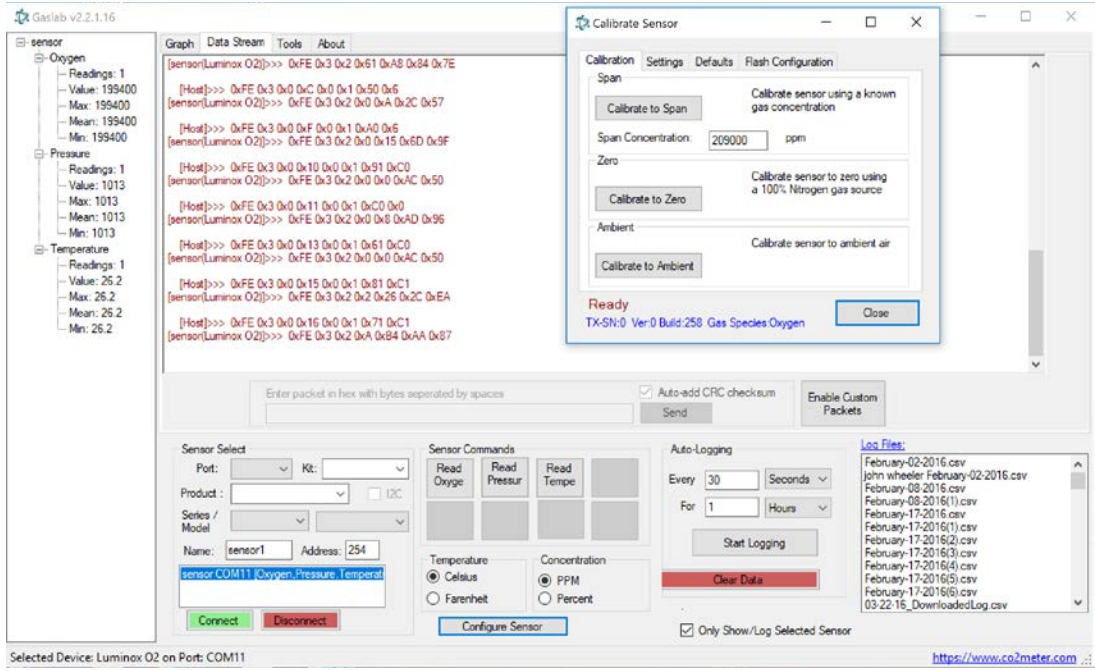
The CO2Meter software is available to download from CO2meter.com/ CO2Meter.com for Configuration, Calibration and Exercise, Laboratory applications and interface development.

For software download: <http://co2meters.com/register/>

For UART only operation a FTDI cable is required,

<https://www.co2meter.com/products/sdk-sensor-replacement-cables?variant=55789862>
For operation with RS485, RS485-Modbus see; GasBox at CO2Meter.com





Chapter 7 Firmware Updates

General

This chapter provides instructions for update the firmware of the device.

WARNING: Programming the device creates a risk of damage as the result of ESD (Electrostatic Discharge). These instructions are intended for engineering and manufacturing personnel trained in the proper control of ESD.

Programming Configuration

Processor

The TX1 uses an ST Microelectronics STM32F051K8U6 microcontroller. This device incorporates a 32-bit ARM Cortex M0 core and 64KB of flash memory.

Programming Interface

The device is programmed through a 3-wire SWD (Single Wire Debug) port. The interface consists of two signals (CLK and DIO) and GND.

Programming Voltage

The microcontroller is internally powered by a regulated 3.3V supply derived from the input voltage. The programming device must be set for 3.3V as target VCC is not available on the 3-wire interface. The means of establishing the target voltage levels are programmer specific.

Device Power

During programming, the TX1 device must be powered by an external 6V to 24VDC supply connected to the power pins. Input voltage must be applied prior to programming.

Connections

The correct connections for programming are illustrated in Figure 7-1 Programming Connections. As the SWD programming pins carry high speed signals, the wiring should be short. For open wires, it is recommended that the wires be twisted.

Programming Device

The recommended programmer is the ST-LINK/V2 from ST Microelectronics. This is a low cost device compatible with the recommended software. Other devices e.g. the ST-LINK V3SET may be used if configured for 3.3V targets. This may require a separate 3.3V power supply, depending on the device.

To configure the ST-LINK.V2 for 3.3 programming over a 3-wire interface, it is necessary to jumper Pins 1 and 19. Pin 19 contains 3.3V source from the programmer. Pin 1 is the target voltage input to the programmer. V1 STM programmers use pin 12, 18 or 20 for GND.

Figure 7-1 Programming Connections

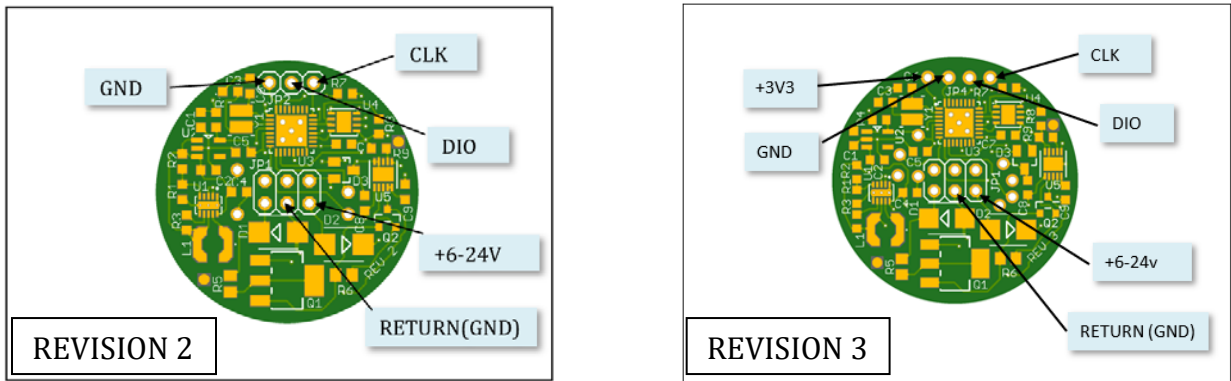
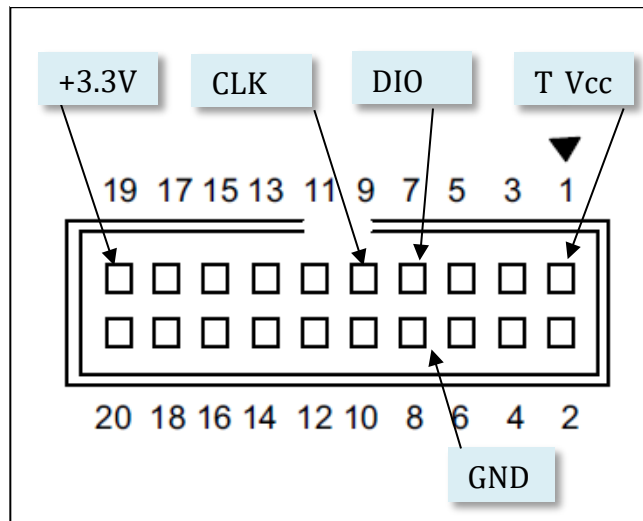


Figure 7-2 ST-LINK/V2 Connections



Programming Software

The recommended programming software is the STM32 CubeProgrammer from ST Microelectronics (Version 1.3 or later). This software is available for free from ST. Note that STM32 CubeProgrammer requires a Java Runtime Environment (8 or later) as a prerequisite.

It is also possible to program the device using the ST-LINK Utility version 4.2 or later.

The ST-LINK/V2 device should normally be updated to the latest version of its device firmware. This update is normally built-in to the utility.

Firmware Files

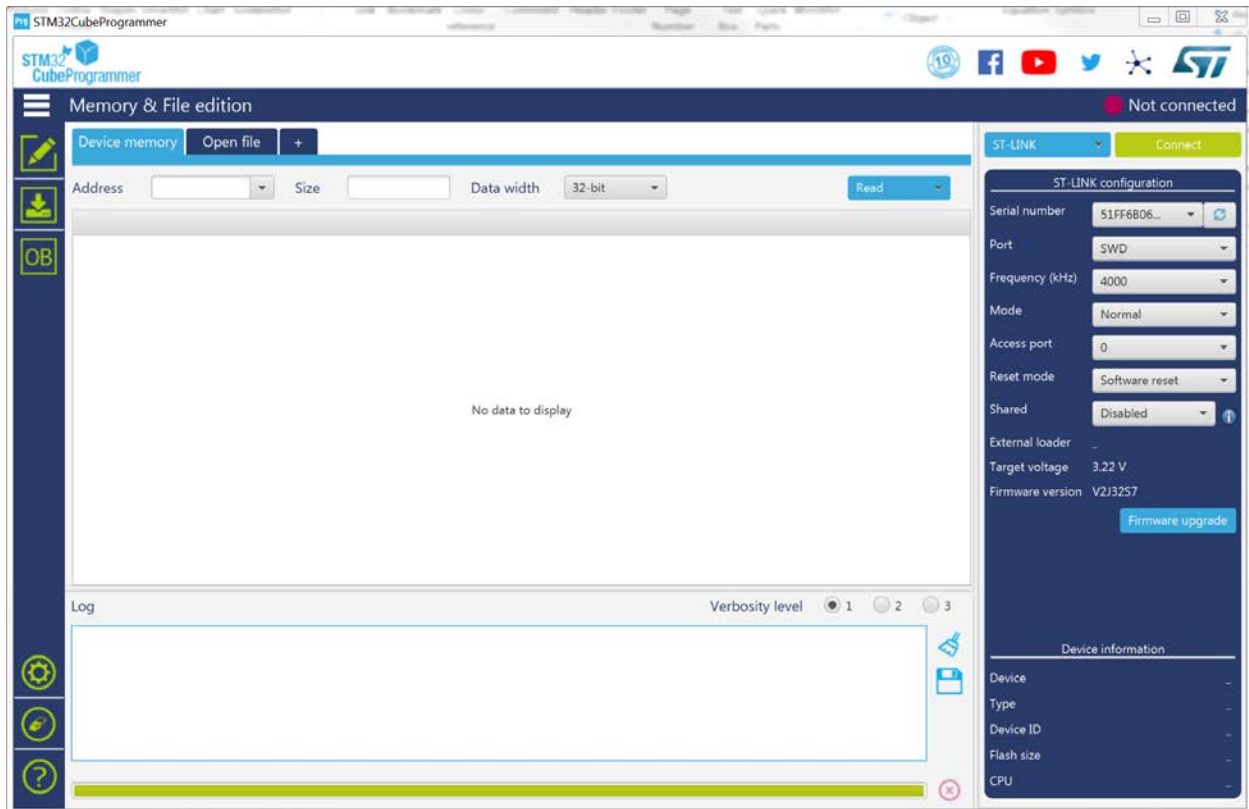
Firmware updates are supplied in the SREC file format which is compatible with both the STM32 CubeProgrammer and ST-LINK Utility tools.

Programming Procedure

Before starting the programmer verify the proper device connections are made and that the device to be programmed is powered up. Insure that the STM programming device is connected to host device by a USB port.

Start the STM Cube Programmer. The display should be similar to Figure 7-3 Initial Screen.

Figure 7-3 Initial Screen



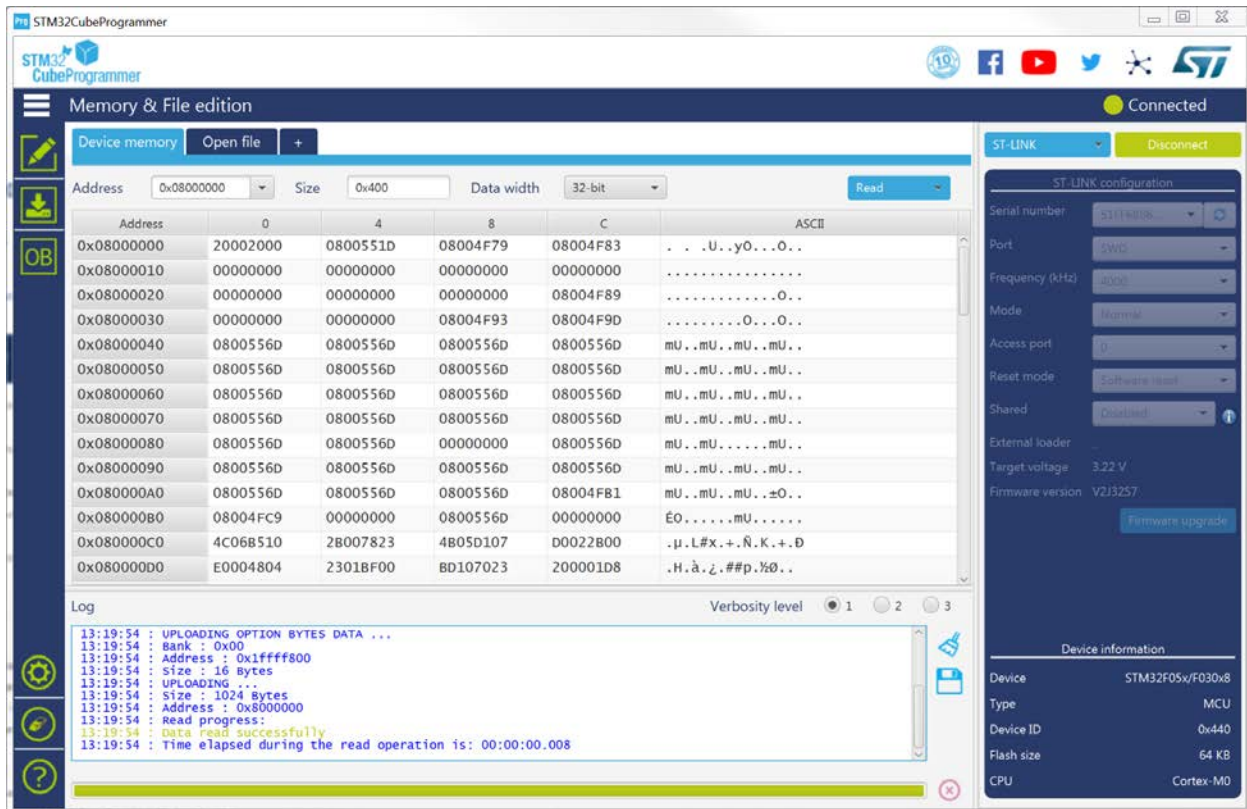
Verify that the ST LINK shows in the upper right corner of the screen and that there is a serial number present for the ST-LINK device.

Check the ST-LINK configuration. It should match the settings shown on the figure.

Connect to Device

Press the connect button. The screen similar to Figure 7-4 Screen After Connection will display.

Figure 7-4 Screen After Connection



Device Erase

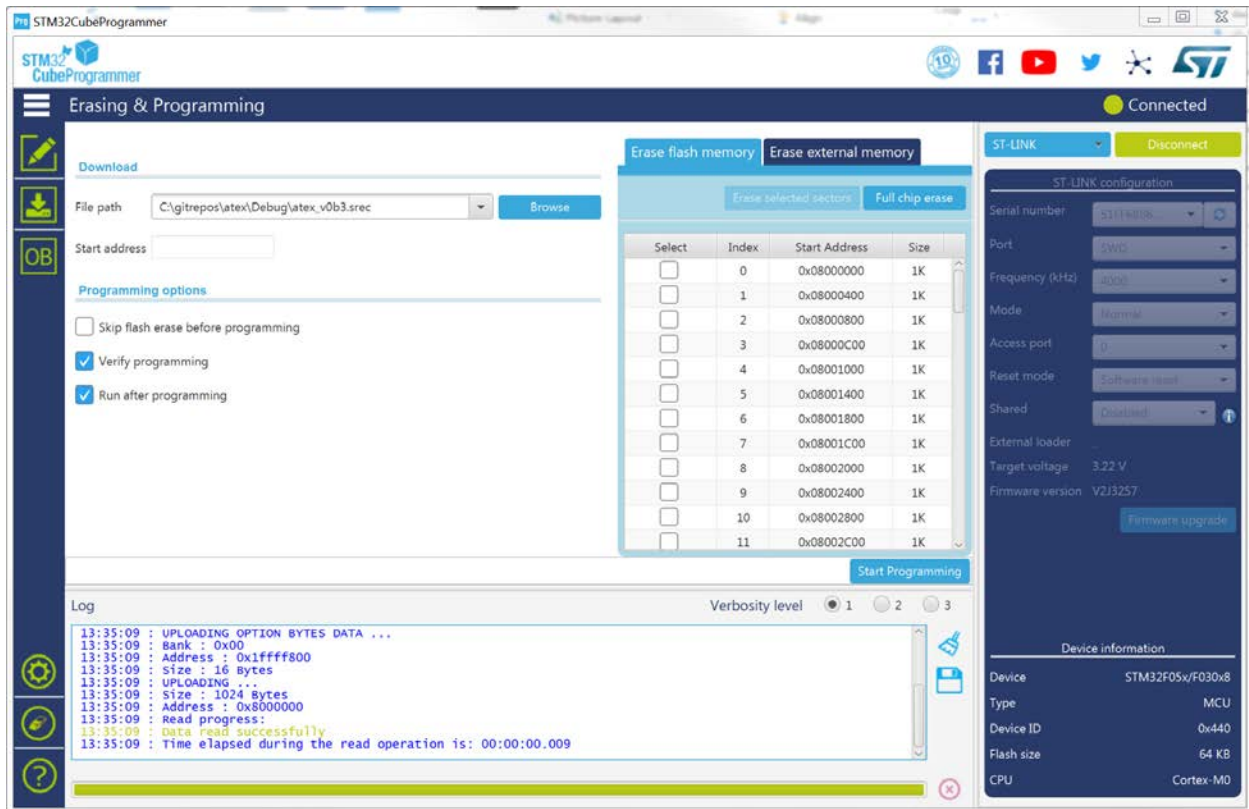
A mass chip erase function is available via a button on the lower left corner of the window. The user should not normally perform a chip erase. Doing so will overwrite data stored in the flash memory outside of the program space. This includes the configuration and the calibration information and the device serial number. When performing a normal programming operation, the tool will erase those sectors associated with the programming.

Programming Firmware

To program new firmware, click on the download ICON on the left side of the screen. This is the second green ICON from the top. This will display the “Erasing and Programming” window as illustrated in Figure 7-5 Erase and Programming Screen.

Press the browse button next to the “File path” to select the file to be programmed. This file should end with a “.srec” extension. Make sure that the “Verify programming” button is checked and that the “Skip Flash erase” button is *not* checked.

Figure 7-5 Erase and Programming Screen



Chapter 8 : Support

The quickest way to obtain technical support is via email. Please send all support inquiries to Support@CO2Meter.com

Please include a clear, concise definition of the problem and any relevant troubleshooting information or steps taken so far, so we can duplicate the problem and quickly respond to your inquiry.

Warranty

This device comes with a 90 day (warranty period) limited manufacturer's warranty, starting from the date the meter was shipped to the buyer.

During this period, CO2Meter.com warrants our products to be free from defects in materials and workmanship when used for their intended purpose and agrees to fix or replace (at our discretion) any part or product that fails under normal use. To take advantage of this warranty, the product must be returned to CO2Meter.com at your expense. If, after examination, we determine the product is defective, we will repair or replace it at no additional cost to you.

This warranty does not cover any products that have been subjected to misuse, neglect, accident, modifications or repairs by you or by a third party. No employee or reseller of CO2Meter.com's products may alter this warranty verbally or in writing.

Liability

All liabilities under this agreement shall be limited to the actual cost of the product paid to CO2Meter.com. In no event shall CO2Meter.com be liable for any incidental or consequential damages, lost profits, loss of time, lost sales or loss or damage to data, injury to person or personal property or any other indirect damages as the result of use of our products.

Returns

If the product fails under normal use during the warranty period, an RMA (Return Material Authorization) number must be obtained from CO2Meter.com. After the item is received CO2Meter.com will repair or replace the item at our discretion.

To obtain an RMA number, call us at or email us at (386) 256-4910 support@co2meter.com.

When requesting an RMA please provide reason for return and original order number. If we determine that the product failed because of improper use (water damage, dropping, tampering, electrical damage etc.), or if it is beyond the warranty date, we will inform you of the cost to fix or replace the product.

For more information visit our website: www.CO2Meter.com/pages/faq

Contact us: We're here to help!

If the troubleshooting guide above doesn't help you solving your problem or for more information, please contact us using the information below.



Support@CO2Meter.com

(386) 256-4910 (M-F 9:00am–5:00pm EST)



www.CO2Meter.com



See CO2Meter, Inc. Terms & Conditions at:

www.CO2Meter.com/pages/terms-conditions

CO2Meter, Inc.

131 Business Center Drive

Ormond Beach, FL 32174

Phone: 386-872-7665 | Fax: 866-422-2356

Email: Sales@CO2Meter.com



131 BUSINESS CENTER DRIVE
ORMOND BEACH, FL 32174
SUPPORT (386) 256-4910
SALES (877) 678-4259
WWW.CO2METER.COM | SALES@CO2METER.COM

Chapter 9 : Appendix 1: Revision History

Version 1.1 Revisions:

A. 5 New Factory configurations are added to support GSS sensors. These are in addition to the Luminox configurations. The available standard configurations are:

Type 0: Luminox 25%

Type 1: Luminox 50%

Type 2: GSS 1-2% Scaling 1

Type 3: GSS 5% Scaling 10

Type 4: GSS 20% Scaling 10

Type 5: GSS 60% Scaling 10

Type 6: GSS 100% Scaling 100

B. The GSS sensor support is enable when the gas type is 1. Luminox sensors have a gas type of 2. The gas type is in holding register 6 (0 based).

C. Luminox sensors default to streaming mode. For the GSS sensors the mode is saved between power ups. To insure proper operation, the sensor is always initialized at power up. The sequence is Streaming Off (K 0), Streaming Configuration (M 4166), Streaming On (K 1). The sensor is always set to report Z, z, T and H although the values only valid when the T and H are fitted in the sensor.

D. Luminox sensors are calibrated by the TX board by maintaining zero and span values in Flash. For the GSS sensors, the calibration is in the sensor. The existing zero and span calibration commands work properly with the GSS by sending the appropriate commands to the sensor. Additional commands have been added as described below is support GSS calibration.

E. Command 40 has been added to stop the sensor from streaming. This is recommended prior to doing calibration commands.

F. Command 41 has been added to resume the sensor streaming. This should be done after the calibration is performed.

G. Command 34 has been to perform an ambient calibration. This uses the G command of the sensor which presumes a 400ppm environment.

Bug Fixes:

H. Parameter 14 sensor packet count was only being updated once per second. It now correctly reflects every packet received from the sensor. When operating at high polling speeds on the Modbus interface this can be used to determine when the sensor value has

been updated. Modbus polling has been tested at 20 polls per second so the sprinter update rates should be supported.

Version 1.2 Revisions:

- A. In previous Firmware Revisions, when the Gas Concentration exceeded that of the 4-20mA max ADC Value, the output could be above 20mA. Revision 1.2 will cap the output to 20mA and all Gas Concentrations above the max ADC Value set in Holding Register Address 22 (HR40023 Current Loop 20mA Level) will be read as 20mA.