SENSUS Developer's Guide

Last Revised: Sept. 17, 2001

1.0 Introduction

This document presents the details of the hardware and software interface required to communicate with ReefNet's SENSUS data recorder. Interested third-party developers are encouraged to extend the SENSUS device with new software offerings. Please note however that although the interface described in this document is relatively straightforward, it is easy to corrupt a connected device if the outlined instructions are not followed. DO NOT attempt to 'discover' features of the protocol – everything required is included below.

2.0 Communication Overview

Communication with a SENSUS device consists of two parts: the hardware interface and the software protocol. This document focuses primarily on the software protocol. However to clarify the reasons for some of the software requirements, the hardware will be discussed briefly.

The connector on the SENSUS device is a micro-miniature MCX coaxial jack. Only two conductors are used to make the data link to the host system's 9-pin RS-232 serial port. Two-way communication is achieved by 'sharing' the available data line. This places certain timing requirements on the software, i.e., one must not attempt to send data while data is being received and vice versa.

> **CAUTION: ReefNet's download cables are NOT simply wire connections to the appropriate RS-232 pins – electronics provide for the sharing of the two wire connection to the three wire standard serial link. DO NOT attempt to wire a custom cable. The SENSUS electronics can easily be damaged by incorrect application of signals or voltages to the interface port.**

The software interface requires direct access to the serial port being used by the SENSUS device. The only signals used are RX, TX, and GND. No flow control signals are implemented, and these should be disabled by the software interface to prevent erroneous transmissions.

Connecting a SENSUS device to a host port 'awakens' its internal interface. The device waits for data in this state, and consumes significantly more current than when disconnected. Sending the appropriate instruction stream via the serial port causes the device to respond with the memory contents (dive data).

> **CAUTION: Due to the increased current drawn by a connected SENSUS device, ReefNet recommends that users not leave the interface cable connected for extended periods. Otherwise battery life may be affected.**

The format of the data contained within the memory dump is outlined later in this document.

3.0 Retrieving Data from a Device

3.1 Communication Settings

Communication with SENSUS devices is made at 19200 bps, 8 data bits, No parity, 1 stop bit. Using other settings will produce unpredictable results, and can easily render the connected device unresponsive.

> **NOTE: Ensure that the serial port receive buffer allocated for SENSUS data is AT LEAST 35000 bytes in size. It may be possible to receive data and process it in real-time on some systems, but it is much safer to receive all the data from the connected device and then process it afterward.**

3.2 Initial Handshake

Send the data byte 0x00 (NULL) to the serial port using the data settings above. This signal alerts the connected device that a data request is imminent. In reply to the alert, the SENSUS device will respond with 12 bytes of useful data:

Table 1 – Handshake data stream

Byte 0/1	"OK" ASCII text response header
Byte 2	Product Code = ASCII "1" for SENSUS
Byte 3	Product Version = ASCII "1" for current SENSUS devices
Byte 4	16 = Battery OK, 64 = Battery LOW
Byte 5	Sampling interval, in seconds
Byte 6/7	(WORD) Device ID = engraved serial number of device
Byte 8/9/10/11	(DWORD) Current SENSUS real-time clock value (in seconds)

If this response is not received within 100 ms, **resend** the 0x00 byte and wait again for the response. If no response is received after 1 second (polling with 0x00 after each 100 ms), there is either no device connected, or the connected device has too low a battery voltage to operate properly.

3.3 Data Retrieval

After successfully receiving the above handshake stream, the SENSUS device is *WAITING FOR A DATA REQUEST*. The user software MUST either request a data dump or cancel the dump operation. Disconnecting a device while in this wait-state may render it unresponsive.

> **CAUTION: ALWAYS respond to the handshake stream. Failing to respond to the stream will not 'damage' the SENSUS device, but it may be rendered unresponsive. In some cases the only remedy is a full reset (removing the battery).**

Wait at least 10 ms before sending your dump/cancel request to the device. This delay ensures the data line is clear before transmission from the host begins.

To **cancel** the data dump, respond to the handshake stream by sending the byte 0x00. The interface can then be safely disconnected.

To **confirm** the data dump, respond to the handshake stream by sending the byte 0x40. The device will then respond with the stream

"DATA" + 32768 bytes data + 16-bit checksum + "END"

The checksum is the 16-bit sum of the 32768 data bytes preceding it. Use the checksum to detect transmission errors that may have occurred.

After sending the data stream above, the SENSUS device returns to its normal mode and may be disconnected from the interface safely.

4.0 Parsing the Memory Dump

The 32 kB memory dump contains all dive data. In operation, the SENSUS electronics use the onboard memory in a circular fashion: when the end of the memory is reached, the memory pointer returns to the beginning to continue recording.

The memory dump must be processed sequentially. Dives each begin with a header of the following form:

Table 2 – Dive Header

SIZE	DESCRIPTION	VALUE
1 Byte	HEADER START	0xFF
1 Byte	Sampling interval (in s)	--
4 Bytes	Timestamp (in s)	--
1 Byte	HEADER END	0xFE

The actual dive data follows the header, and is stored in the following form (D = depth byte, T = temp. byte):

DT, D, D, D, D, D, DT, D, D, D, D, D, DT, D, D, D, D, D, ………

That is, temperature data is sampled once for every SIX depth samples (corresponding to once a minute at a 10 second sampling interval).

Temperature values are stored in **degrees Fahrenheit**.

Depth values are stored in **ADJUSTED feet of seawater**. A constant of 13 fsw is added to each actual depth to ensure that dives conducted at high altitudes do not underflow the depth byte. For dive data from sea level, simply subtract 13 from each depth value to get 'real' depth. If dive data was recorded at a higher altitude, subtract the appropriate amount to correct for that altitude.

The end of a dive is reached when 17 consecutive depth samples of less than 3 feet have been found.

CAUTION: The ONLY certain method of finding the end of a dive is that described above. Bytes may exist between the end of one dive and the header of the next – DO NOT simply continue reading dive data until the next header is encountered!

5.0 FOR MORE INFORMATION

For technical questions, please contact:

Kris Wilk
ReefNet Inc.
3610 Walnut Grove
Mississauga, Ontario L5L2W8
Canada

Tel:	1-888-819-REEF or
	(905)608-9373
Fax:	(905)820-1927

wilk@reefnet.on.ca
www.reefnet.on.ca

6.0 DISCLAIMER

ReefNet cannot be held responsible for damage to a connected SENSUS device caused by software other than ReefNet's Data Analyst. Nor can ReefNet be held responsible for damage caused by tampering with either a SENSUS device or its interface cable.