

Newsteo telemetry

RF-Monitor TCP/IP Server

Protocol specification

rev 1.3

■ Purpose:

This document defines the protocol used by RF-Monitor to relay RF-packets received from the RF-to-USB Key, to other applications using TCP-IP network protocol.

■ Required devices and software:

- any Newsteo telemeter (Data-logger, Builder, Tracker ...)
- one Newsteo RF-to-USB Key
- at least one computer with RF Monitor 3.1.7.7x and RF-to-USB Key drivers installed

■ Technical support:

For any question, remark or advice on product or communication protocol, please contact:

support@newsteo.com

+33 (0)4 42 01 82 23

■ Table of contents :

1.	Introduction
2.	Configure RF-Monitor
3.	TCP/IP frames format
3.1	Raw format
3.2	CSV format sample: ASCII measure for Builder
3.3	Header packets
4.	Measures format
4.1	LOG221: Temperature and humidity
4.2	BLD262: Temperature, humidity and constraint for concrete
4.3	EPEE
	Format trames de mesures

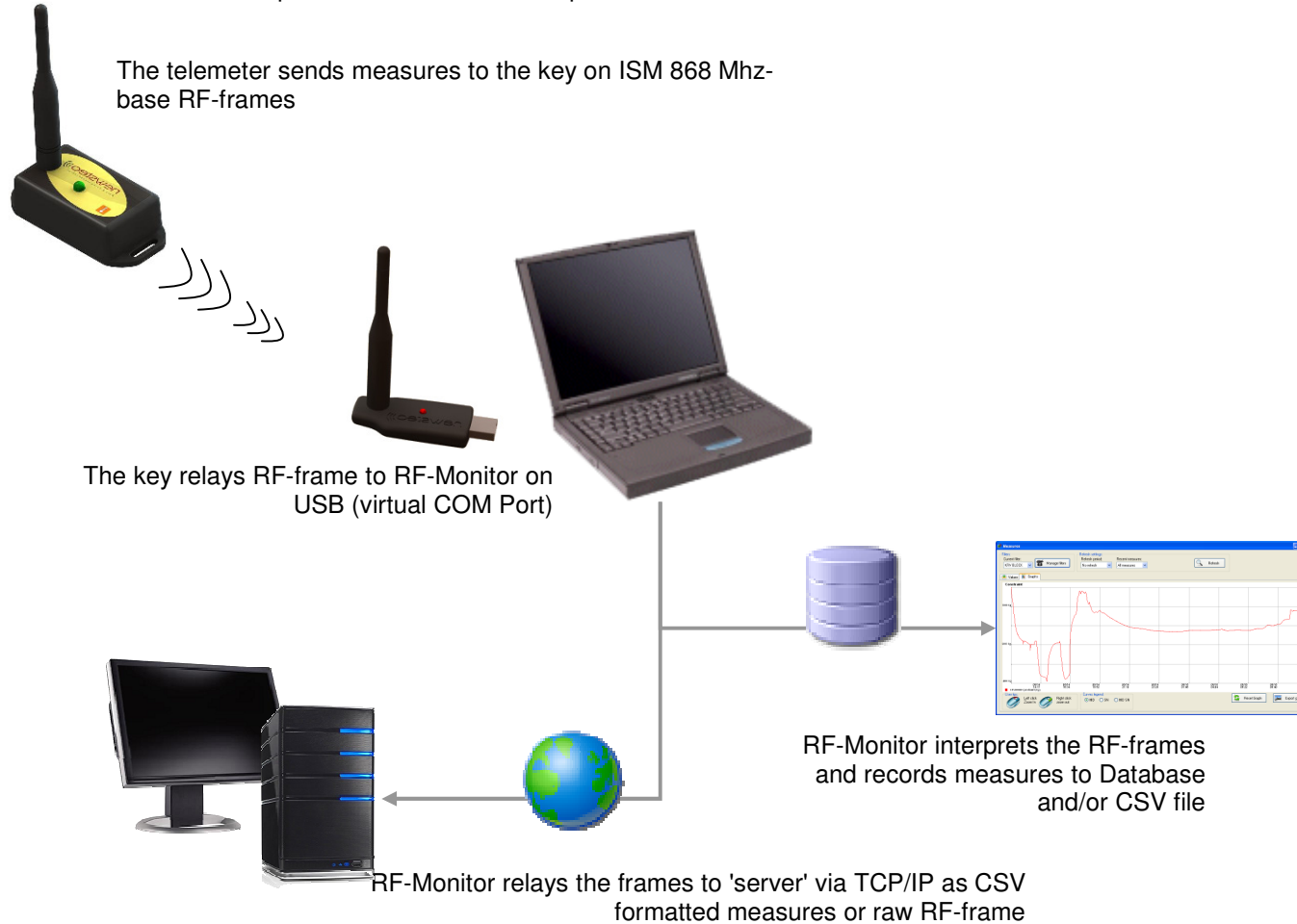
Exemples de trames.....	
5. TCP/IP server example	
6. TCP/IP daemon	

■ History:

- 1.0 First Release
- 1.1 Frame examples and measures format
- 1.2 Information about EPEE Project: **important**, packet size and structure have been modified (BlockId)
Header packets frame at connection
- 1.3 Add TCP/IP daemon listener documentation
- To-do Exhaustive list of measures format for each product

1. Introduction

We introduce here the main process of TCP/IP data export in RF-Monitor.

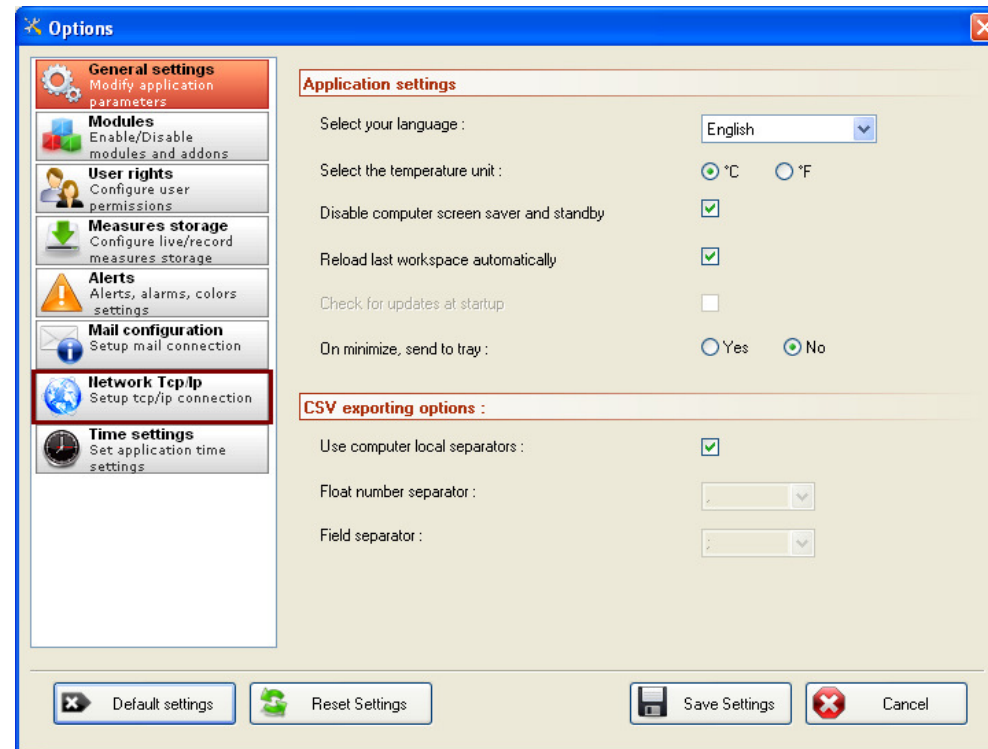


In next chapters, we will specify how to activate and configure TCP/IP export with RF-Monitor, and we'll explain the formats of transmitted data.

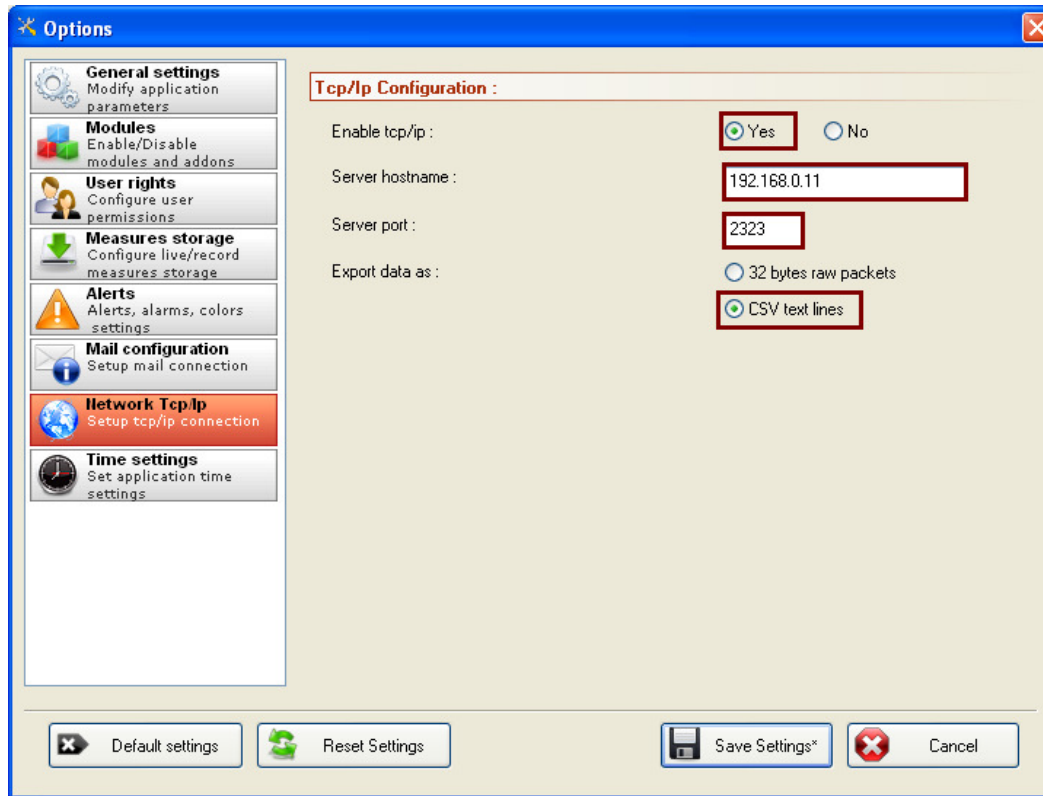
2. Configure RF-Monitor

To configure RF-Monitor TCP/IP client, please follow these steps:

- Run RF-Monitor application and select **Options** menu
- Select **Network TCP/IP** item



- **Enable TCP/IP** by selecting **Yes**



- Enter **Server hostname**: a valid computer IP or DNS such as **192.168.0.11** or **localhost**, (for loop connection) on which the target software that will receive the data is running.
- Enter **Server port** that target software is listening at. Default is **2323**. No value lower than 1024 should be set because they are used for standard protocols (mails, web server, ...)
- Select the data format you want to export :
 - **Raw packets**: a raw measures packet (hexadecimal) received by the RF-to-USB Key
 - **CSV text**: an ASCII text of interpreted measures, comma-separated.

3. TCP/IP frames format

The communication protocol used by the application is TCP connected mode.
RF-Monitor tries connection to the server using parameters set by the user until success.
Once the connection has succeeded, it sends measures packet to the server.
Otherwise, the packet is dumped on timeout.

BS (2B)	MF (1B)	PACKET CONTENT	ETX (2B)
---------	---------	----------------	----------

BS: total Buffer Size, on 2 Bytes. Minimum value is 0x00 0x05 (no data). Always 0x00 0x25 (37) if MF=raw format.

MF: measure Format :

0x00 = raw format

0x01 = CSV text

0x02 = header format

Packet content: the RF packet including some information about telemeters and measures, in raw data or CSV text format. In raw mode, always 32Bytes long.

ETX: end of packet, 2 fixed Bytes. Hexa: 0x24 0x24. ASCII: "\$\$".

3.1 Raw format

Packet is exact replication of RF-frame; measures and telemeter information is sent as hexadecimal raw data.

HEXA

00	25	00	A1 52 50 54 00 00 01 42 4C 44 00 00 00 10 00 00 21 6E E0 62 40 36 30 FA 30 09 42 13 02 36 8F FF	24	24
----	----	----	---	----	----

INTERPRETED

37	0	A1 RPT000001 BLD000000 10 0000 216EE062403630FA3009421302368FFF	\$	\$
----	---	---	----	----

37: size of the buffer

0: raw mode

Packet content: "A1 RPT000001 BLD000000 10 0000 216EE062403630FA3009421302368FFF"

In this example, A1 = live measure, RPT000001 is the repeater id that transmitted a 0x10Bytes long measure (216EE062403630FA3009421302368FFF) from builder BLD000000.

See measure interpretation for more information.

\$\$: end of the packet

3.2 CSV format sample: ASCII measure for Builder

Packet is interpreted; measures and telemeter information is ASCII-formatted.

HEXA

00	5B	01	66 76 68 48 48 48 48 48 48 59 56 53 54 56 44 48 48 48 59 54 50 57 48 44 48 48 48 59 56 56 57 44 48 48 48 59 53 50 44 50 57 53 59 51 52 56 48 56 52 44 54 56 57 59 54 44 53 50 50 59 51 44 53 55 54 59 50 51 44 51 48 48 59 50 57 44 54 53 52 59 52 44 53 55 52 59	24	24
----	----	----	---	----	----

INTERPRETED

91	1	BLD000000;8568,000;6290,000;889,000;52,295;348084,689;6,522;3,576;23,300;29,654;4,574;	\$	\$
----	---	--	----	----

91: size of the buffer

1: CSV format

Packet content : "BLD000000;8568,000;6290,000;889,000;52,295;348084,689;6,522;3,576;23,300;29,654;4,574;"

See measure interpretation for more information.

For BUILDER, in this example, values are:

Raw bridge: 8568

Raw temperature: 6290

Raw humidity: 889

Bridge: 52.295

RVishay: 348084.689

dL/L: 6.522 µm/m

Battery: 3.576V

Temperature: 23.300 °C

Relative humidity: 29.654

Dew Point: 4.574 °C

\$\$: end of the packet

3.3 Header packets

When the user connects to the application RF-Monitor, several header packets are sent to describe the protocol used or any other information. This is an enhancement from version 1.2.

Syntax of the ASCII packet content is: FIELD_LABEL = VALUE. Here is a list of fields:

FIELD LABEL	DESCRIPTION	since
PROTOCOL_VERSION	Version of the protocol	v1.2
FIELD_SEPARATOR	Character used to separate the CSV ASCII fields	v1.2

Examples:

HEXA

00	17	02	50 52 4F 54 4F 43 4F 4C 5E 56 65 72 73 69 6F 6E 3D 32	24	24
----	----	----	---	----	----

INTERPRETED

23	2	PROTOCOL_VERSION = 2	\$	\$
----	---	----------------------	----	----

4. Measures format

Here is described for each product, the way measures are transmitted and how to interpret them into PACKET CONTENT of TCP/IP frame, for each mode (RAW or ASCII).

4.1 LOG221: Temperature and humidity

LOG221 - RAW PACKET CONTENT									
	measure type	destination id	source id	payload length	information	measure raw temperature rawT	measure raw humidity rawH	-	timestamp
Length (byte)	1	9	9	1	2	2	2	5	7
Precision (bits)						14	12	-	
Example	A1 (measure with ack)	4c 4f 47 00 00 93 (LOG000093)	4b 45 59 14 00 aa (KEY1400AA)	10	00 00 (live measure)	6a 08	55 70	-	06 13 11 04 18 86 08 (2008/06/18 11:13:06)
Default formula or format						T °C = -39.6+0.01*raw	-4+0.0405 * rawH - (0.0000028*rawH^2)		ss min hh - dd mm yy

LOG221 * ASCII CSV PACKET CONTENT									
Field	source id (telemeter serial)	raw temperature	raw humidity	raw relative humidity	raw dew point	temperature °C	sensor humidity	relative humidity (%)	dew point (°C)
Example	LOG000093;27048,000;22688,000;0,000;105,000;28,020;47,799;48,172;16,039;								

4.2 BLD262: Temperature, humidity and constraint for concrete

BLD262 * ASCII CSV PACKET CONTENT										
Field	source id (telemeter serial)	raw temperature	raw humidity	raw bridge	RVishay (ohm)	dL/L (µm/m)	Battery level (V)	temperature (°C)	relative humidity (%)	dew point (°C)
Example	BLD0000AA; 8728,000;6712,000;1482,000;53,271;350234,364;-4,759;3,620;27,520;50,195;16,227;									

4.3 EPEE

Format trames de mesures

Le format de la trame TCP/IP (ascii) est **FRAME_LENGTH;FRAME_TYPE;SENSOR/MODULE_ID;TIMESTAMP;[BLOCK_ID;]MEASURES;\$\$**

FRAME_LENGTH: longueur totale de la trame TCP/IP sur 2 octets. Transmis sous format hexa (ex 0x00 0x25 pour une longueur de 37)

FRAME_TYPE: indicateur du format des trames, 01 pour format ASCII. Transmis sous format hexa (01).

TIMESTAMP: Horodatage de la mesure. Dans le cadre de ce projet, toutes les mesures sont horodatées par RF-Monitor sur réception de la trame radio. Ex: 2008;07;06;13;45;44 pour le 06 juillet 2008 à 13h45min44s. Transmis sous format ASCII.

SENSOR/MODULE_ID: L'ID du module/capteur est l'identifiant radio du capteur ou du module regroupant plusieurs capteurs, et aura le format suivant: [TRIGRAMME (3B)] [SERIAL NUMBER (6B)]. Le trigramme permet de distinguer les types de télémètre et donc spécifie le format des trames de mesures correspondant. Les trigrammes possibles sont: ENV, RTA, SHK, FLW respectivement pour les modules Environnement, Roulis/Tangage, Contrainte et Débitmètre.

[BLOCK_ID*]MEASURES: Les mesures des capteurs sous format ASCII. Le Block_Id est optionnel, sa présence spécifie le numéro de block/capteur auquel se rapporte la mesure, pour les modules présentant plusieurs mesures. Par exemple pour le module environnement les valeurs 01, 02, 03, 04, 05 pour respectivement les mesures de anémométrie, GPS, Roulis/Tangage, Accélération, Température/Humidité et 06 pour la batterie. Le format et l'organisation des mesures sont décrits plus loin. Transmis sous format ASCII (0x45 0x4E 0x56 0x30 0x30 0x30 0x33 0x31 à interpréter comme ENV000031)

\$\$: fin de trame. Transmis sous format hexa

Le format des trames de mesures est décrit dans le tableau suivant, regroupé par module et par bloc.
Le séparateur décimal est spécifié dans les options de RF-Monitor.

MODULE / TRIGRAM	BLOC K ID	CAPTEUR(S)	MESURE	UNIT E	PLAGE/ RESOLUTION	ENCODI NG	PRECIS ION	PER IOD E	FORMAT TRAME ASCII: MESURE	
ENVIRONNEMENT ENV	01	ANEMOMETRE	Vitesses du vent moyenne, max, min	Knot	0.0 à 99.0	2B	0.1 Knot	60s	01; WND_SPD_AV; WND_SPD_MAX; WND_SPD_MIN; WND_DIR_AV;	
		GIROUETTE	Direction moyenne du vent	°	0 à 360	1B	5°			
	02	GPS	Latitude (°)	°	0 à 90	1B	-	5s	02; LAT_DEG; LAT_MIN; LAT_SEC; LAT_CARD; LONG_DEG; LONG_MIN; LONG_SEC; LONG_CARD; SPD;	
			Latitude (')	min	0 à 59		-			
			Latitude (")	s	0.000 à 59.000		0.006s			
			Cardinalité Latitude	-	[N, S]	2B	-	1s		
			Longitude (°)	°	0 à 180		-			
			Longitude (')	min	0 à 59		-			
			Longitude (")	s	0.000 à 59.000		0.006s			
			Cardinalité Longitude	-	[E, W]		-			
	03	ACCELEROMETR E	Accélérations moyennes X, Y, Z	g	-3.000 à +3.000		0.1g	1s	03; ACC_X_AV; ACC_Y_AV; ACC_Z_AVG; ROLL_AVG; PITCH_AVG;	
		ROULIS TANGAGE	Roulis Tangage	° °	-90 à +90 -90 à +90		1° 1°			
	04	ACCELEROMETR E	Accélérations max X, Y, Z Accélérations min X, Y,	g	-3.000 à +3.000		0.1g	1s	04; ACC_X_MAX; ACC_Y_MAX; ACC_Z_MAX; ACC_X_MIN; ACC_Y_MIN; ACC_Z_MIN;	

	05	TEMPERATURE-HUMIDITE	Température Humidité relative	°C %	-20.00 à 80.00 0.000 à 100.000		0.4 °C 3%	10 min	05; TEMP; REL_HUM;
	06	BATTERIE	<i>*à définir*</i>	V			0.1V	10 min	<i>*à définir*</i>
ROULIS TANGAGE RTA	01	ACCELEROMETRE ROULIS TANGAGE	Accélérations moyennes X, Y, Z Roulis Tangage	g ° °	-3.000 à +3.000 -90 à +90 -90 à +90		0.1g 1° 1°	1s	ACC_X_AVG; ACC_Y_AVG; ACC_Z_AVG; ROLL_AVG; PITCH_AVG;
	02	BATTERIE	<i>*à définir*</i>	V			0.1V	10 min	<i>*à définir*</i>
CONTRAINTE SHK	01	CONTRAINTE	Déformations 1 moyenne, max, min Déformations 2 moyenne, max, min Déformations 3 moyenne, max, min	µm/m	<i>*à définir*</i>		1 µm/m	1s	SHK_1_AVG; SHK_1_MAX; SHK_1_MIN; SHK_2_AVG; SHK_2_MAX; SHK_2_MIN;
	02	BATTERIE	<i>*à définir*</i>	V			0.1V	10 min	<i>*à définir*</i>
DEBITMETRE TRE FLW	01	DEBITMETRE	Débit direct Débit reflux	L/h	0.0 à 1000.0		1%	1 s	DELTA; [DIRECT_FLOW; BACK_FLOW;]**
	02	BATTERIE	<i>*à définir*</i>	V			0.1V	10 min	<i>*à définir*</i>

** Les débits directs et de reflux seront intégrés si possible, confirmation sera donnée après étude du capteur. Le Delta sera quant-à lui systématiquement fourni.

Rappel: pour le 'module' ENV, le 1er octet de mesure indique le Type de Capteur / ID Block. Il DOIT correspondre à l'identifiant du bloc descripteur qu'utilise RF-Monitor pour récupérer la mesure.

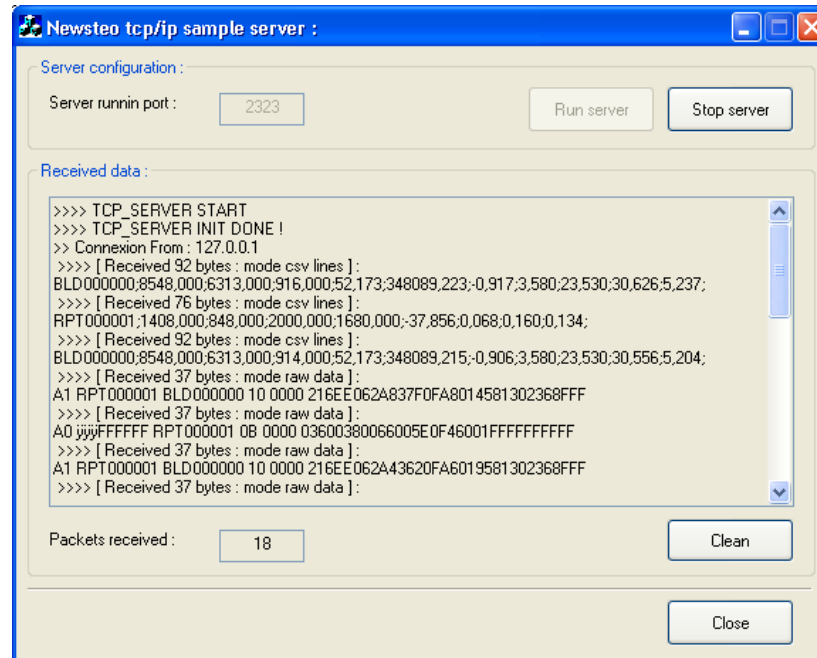
Exemples de trames

Les données en *italique* (ex: 0x17 0x01) sont hexadécimales. Les autres sous format ASCII (1Byte/caractère). Les mesures capteurs sont données à titre illustratif et non forcément cohérentes d'un point de vue physique ou mécanique.

MODULE	TRAME - INTERPRETATION MESURE
ENV WIND	0x0039 0x01 ENV000001; 2008; 07; 28; 17; 40; 25; 01; 18.7; 31.5; 15.4; 113; 0x24 0x24 Capteur: ENV000001 / 01(VENT) Date: 28/07/2008 à 17:40:25 Vitesses du vent moyenne = 18.7 nœuds, max = 31.5 nœuds, min = 15.4 nœuds Direction moyenne du vent: 113°
ENV GPS	0x0047 0x01 ENV000001; 2008; 07; 28; 17; 40; 26; 02; 46; 06; 50.990; N; 05; 53; 31.930; 11.7; 0x24 0x24 Capteur: ENV000001 / 02(GPS) Date: 28/07/2008 à 17:40:26 Latitude = 46°06'50.990" N Longitude = 05°53'31.930" E Vitesse = 11.7 nœuds.
ENV RTA	0x0037 0x01 ENV000001; 2008; 07; 28; 17; 40; 27; 03; 1.7; 0.8; -1.2; 7; 3; 0x24 0x24 Capteur: ENV000001 / 03 (ROULIS/TANGAGE) Date: 28/07/2008 à 17:40:28 Accélérations moyennes en X = 1.7g; Y = 0.8g, Z = -1.2g Roulis moyen = 7° Tangage moyen = 3°
ENV ACC	0x0041 0x01 ENV000001; 2008; 07; 28; 17; 40; 28; 04; 2.1; 1.2; -1.7; 0.1; -0.7; -1.4; 0x24 0x24 Capteur: ENV000001 / 04 (ACCELERATIONS) Date: 28/07/2008 à 17:40:28 Accélérations maximales en X = 2.1g; Y = 1.2g, Z = -1.7g Accélérations minimales en X = 0.1g; Y = -0.7 g, Z = 1.4g
ENV TH	0x002E 0x01 ENV000001; 2008; 07; 28; 17; 40; 29; 05; 17.5; 72; 0x24 0x24 Capteur: ENV000001 / 05 (TEMPERATURE/HUMIDITE) Date: 28/07/2008 à 17:40:29 Température: 17.5°C Humidité relative: 72%
RTA	0x0034 0x01 RTA000001; 2008; 07; 28; 17; 40; 30; 01; 1.7; 0.8; -1.2; 7; 3; 0x24 0x24 Capteur: RTA000001 Date: 28/07/2008 à 17:40:30 Accélérations moyennes en X = 1.7g; Y = 0.8g; Z = -1.2g Roulis moyen = 7° Tangage moyen = 3°
SHK	0x0041 0x01 SHK000001; 2008; 07; 28; 17; 40; 31; 01; 25; 31; 15; 28; 29; 27; 17; 19; 12; 0x24 0x24 Capteur: SHK000001 Date: 28/07/2008 à 17:40:31 Déformation 1 moyenne = 25µm/m, max = 31µm/m, min = 15µm/m Déformations 2 moyenne = 28µm/m, max = 29µm/m, min = 27µm/m Déformations 3 moyenne = 17µm/m, max = 19µm/m min = 12µm/m
FLW	0x2C 0x01 FLW000001; 2008; 07; 28; 17; 40; 32; 01; 15.70 [32.80; 17.10]; 0x24 0x24 Capteur: FLW000001 Date: 28/07/2008 à 17:40:32 Débit différentiel = 15.70 L/h [Débit direct = 32.80 L/h] [Débit reflux = 17.10 L/h]

5. TCP/IP server example

A full example of Windows (Xp/Vista) binary and source code (VC++) of TCP/IP server is available for download at:
<http://www.newsteo.com/private/releases/release.php#tcpipserver>



6. TCP/IP daemon

Pour répondre aux besoins du projet EPEE, un démon TCP/IP va être implémenté. Ce démon répondra à une liste limitée de commandes qui permettront entre autre d'arrêter le processus de maitre / esclave et de terminer l'application.

Ce démon répondra sur le port 2324 par défaut, et sera configurable via les options du logiciel (sur le même écran que précédemment) au niveau de l'intitulé (daemon port).

La liste des commandes interprétées est la suivante :

HELLO

- l'application RFMonitor répond sur la socket 2323 : *HELLO_ACK*

L'octet Measure Format devient 0x03 (pour command ack)

HEXA

00	0E	03	48 45 4C 4C 4F 5F 41 43 4B	24	24
----	----	----	----------------------------	----	----

INTERPRETED

14	3	HELLO_ACK	\$	\$
----	---	-----------	----	----

STOP_MASTER_THREAD_PROCESS

- l'application RFMonitor répond sur la socket 2323 : une réponse en plusieurs séquences.

// à réception de la commande l'application envoie les 2 trames suivantes

STOP_MASTER_THREAD_PROCESS_BEGIN
REMAINING_DEVICES=k

L'application envoie alors des trames d'hibernates aux différents produits

// lorsque un paquet d'hibernate nous parvient on envoie les infos suivantes sur la socket 2323

DEVICE_SN HIBERNATE_ACK
REMAINING_DEVICES=k-1

...

// lorsque l'on reçoit le dernier hibernate on envoie les infos suivantes avant de s'arrêter

DEVICE_SN HIBERNATE_ACK (le dernier)
REMAINING_DEVICES=0
STOP_MASTER_THREAD_PROCESS_END

INTERPRETED

37	3	STOP_MASTER_THREAD_PROCESS_BEGIN	\$	\$
24	3	REMAINING_DEVICES=4	\$	\$
27	3	ENV000001 HIBERNATE_ACK	\$	\$
24	3	REMAINING_DEVICES=3	\$	\$
35	3	STOP_MASTER_THREAD_PROCESS_END	\$	\$

KILL_APP

- l'application RFMonitor répond sur la socket 2323 KILL_APP_ACK et se termine dans l'état. (Si le stop master thread n'a pas été effectué il ne sera pas effectué et les produits ne seront pas en hibernate).

Cependant, si on est en cours de stop master thread et que l'un des produits n'a pas répondu, c'est le seul moyen de terminer l'application.

HEXA

00	11	03	4B 49 4C 4C 5F 41 50 50 5F 41 43 4B	24	24
----	----	----	-------------------------------------	----	----

INTERPRETED

17	3	KILL_APP_ACK	\$	\$
----	---	--------------	----	----