	Name	Fonction
Editor:	Fabrice Vély	
Approving:		

SUMMARY

ANNEX

1. Purpose and scope

This document presents the tools made available by watteco.

2. Reference documents

Downloadable tools on: ftp://nkesigfox:watteco@ftp.nke.fr\Tools

- SigfoxDecoder.exe
- DecompressionDelta.exe
- DeltaCompressionLib.dll
- DCL.cpp : source code of software using the Delta Decompression dll

3. Operating mode

3.1 Examples of frame decompression

3.1.1 1st example

Global compressed frame
0x0930a21902 (little Endian):

Hexa	Binary (Little Endian)	Binary (Big Endian)	
09	0000 1001	1001 0000	1001 = new absolute value. This code 1001 means that the following data is a new absolute value (and not a delta with respect to the previous value). See the probability tree.
30	0011 0000	0000 1100	0000 0000 1100 0100 = data 0x00C4. Here, we are in 16-bit base so the absolute value is on 16-bit and the next 16 bits.
A2	1010 0010	0100 0101	01 = the next probability code. So the data is based on the difference with the previous value with difference of ± 1. Associated data is 0 equivalent to +1 (so 0x00C5) 0b0 correspond to +1 and 0b1 correspond to -1 so here is +1 11 = the next probability code is « no difference » (so 0x00C5) the data does not evolve (so no associated data)
19	0001 1001	1001 1000	001 = the next probability code. So a difference between ±4 to 7. 000 => +4, 001 => +5, 010 => +6, 011 => +7 and the same for negative delta by replacing the first 0 with a 1 (100 => -4) And associated data 100 equivalent to -4 (so 0x00C1)
02	0000 0010	01 000 000	001= the next probability code. So the same that previous code

			And associated data 000 equivalent to +4 (so 0x00C5) 000 : The last byte is completed by bit value 0. No code in the probability tree contains only 0. So we use bit value 0 to complete the last byte.

Extract from SPDL Delta compression:

This probability tree, gives us for a code for the:
- « 0b11 » ➔ No difference
- « 0b01 » ➔ Difference of ± 1
- « 0b101 » ➔ A difference between ± 2 and 3
- « 0b001 » ➔ A difference between ± 4 and 7
- « 0b1001 » ➔ A new absolute value
- « 0b0001 » ➔ A difference between ± 8 and 15
- « 0b00001 » ➔ A difference between ± 16 and 31
- « 0b10001 » ➔ A difference between ± 32 and 63
- « 0b000001 » ➔ A difference between ± 64 and 127
- « 0b100001 » ➔ A difference between ± 128 and 255
- « 0b0000001 » ➔ A difference between ± 256 and 511
- « 0b10000010 » ➔ A difference between ± 512 and 1023
- « 0b10000011 » ➔ A difference between ± 1024 and 2047
- « 0b000000010 » ➔ A difference between ± 2048 and 4095
- « 0b000000011 » ➔ A difference between ± 4096 and 8191
- « 0b100000010 » ➔ A difference between ± 8192 and 16383
- « 0b100000011 » ➔ A difference between ± 16384 and 32767
- « 0b100000000 » ➔ A difference between ± 32768 and 65535
- « 0b000000000 » ➔ A difference between ± 65536 and 131071
- « 0b1000000011 » ➔ A difference between ± 131072 and 262143
- « 0b0000000010 » ➔ A difference between ± 262144 and 524287
- « 0b10000000100 » ➔ A difference between ± 524288 and 1048575
- « 0b10000000101 » ➔ A difference between ± 1048576 and 2097151
- « 0b00000000110 » ➔ A difference between ± 2097152 and 4194303
- « 0b00000000111 » ➔ A difference between ± 4194304 and 8388607

Thus probability codes represent the following fields in the memory:
0b11
0b01S
0b101SX
0b001SXX
0b1001XX….XX (Depending on the format of the base datum)
0b0001SXXX
0b00001SXXXX
0b10001SXXXXX
0b000001SXXXXXX
0b100001SXXXXXXX
0b0000001SXXXXXXXX
0b10000010SXXXXXXXXX
0b10000011SXXXXXXXXXX
0b000000010SXXXXXXXXXXX
0b000000011SXXXXXXXXXXXX
0b100000010SXXXXXXXXXXXXX
0b100000011SXXXXXXXXXXXXXX
0b100000000SXXXXXXXXXXXXXX
0b000000000SXXXXXXXXXXXXXXX
0b1000000011SXXXXXXXXXXXXXXXX
0b0000000010SXXXXXXXXXXXXXXXX
0b10000000100SXXXXXXXXXXXXXXXXX
0b10000000101SXXXXXXXXXXXXXXXXXX

0b00000000110SXXXXXXXXXXXXXXXXXXXX
0b00000000111SXXXXXXXXXXXXXXXXXXXX

With S as the sign and XX..XX a binary value.
If S = 1, then the delta is negative, otherwise it is positive.

3.1.2 2nd example (S0 device)

2 frames received
1st frame: CB0401010B09000000F6FFFF
2nd frame: CB14FFFFFF03
0xCB =203 : S0 device
0x04 = 0b0000 0100
00 : message typing : standard data
01 : Compression type : Delta compression
0000 : frame index : 0
0x01 = 1 : Function Number = 1 Pulse count measurement
0x01 = 0000 0001 : Measurement period
0 : reverse chronological order (1 = Chronological)
0 : measurement period set in hours
00 0001 : measurement period is 1
Conclusion : measurement period is 1h, in reverse chronological order
0x0B = 11 Number of bytes transmitted for compressed or uncompressed count measurements (§3.7.2)
0x09000000F6FFFF : Table of compressed count measurements (index frame n°1)

Frame n°2 CB14FFFFFF03
0xCB =203 : S0 device
0x14 = 0b0001 0100
00 : message typing : standard data
01 : Compression type : Delta compression
0001 : frame index : 1
0xFFFFFF03 : Table of compressed count measurements (index frame n°2)

Global compressed frame
0x09 00 00 00 F6 FF FF FF FF FF 03 Little Endian

Hexa	Binary (Little Endian)	Binary (Big Endian)	
09	0000 1001	1001 0000	0b1001 : it is a new absolute value
00	0000 0000	0000 0000	0b0000 0000 0000 0000 0000 0000 0000 0110 = 0x0006 is the new absolute value
00	0000 0000	0000 0000	
00	0000 0000	0000 0000	
F6	1111 0110	0110 1111	0b11 = probability code “No difference”. With this code there is no associated data. We count 23 times this code, that means that are 23 identical values to the absolute value (6) measured each hour.
FF	1111 1111	1111 1111	
FF	1111 1111	1111 1111	
FF	1111 1111	1111 1111	
FF	1111 1111	1111 1111	

FF	1111 1111	1111 1111	
03	0000 0011	1100 0000	To finish last byte 0b00 0000 is added to complete

In the end, there are 24 identical values measured during 1 day (24h)

3.1.3 3nd example (TH Device)

4 frames received

1st frame: ca040001100f0950624b9254
2nd frame: ca144aad0445531c2efe6a0d
3rd frame: ca2409a862dbadb5d65b737b
4th frame: ca3437e6cef40e

Frame n°1 : CA040001100F0950624B9254
0xCA =202 : TH device
0x04 = 0b0000 0100
00 : message typing : standard data
01 : Compression type : Delta compression
0000 : frame index : 0
0x00 = 0 : Function Number = 0 Temperature/Humidity measurement
0x01 = 0000 0001 : Measurement period
0 : reverse chronological order (1 = Chronological)
0 : measurement period set in hours
00 0001 : measurement period is 1
Conclusion: measurement period is 1h, in reverse chronological order
0x10 = 16 Number of bytes transmitted for compressed or uncompressed count measurements (Temperature)
0x0F = 15 Number of bytes transmitted for compressed or uncompressed count measurements (Humidity)
0x0950624B9254 : Table of compressed count measurements (index frame n°1)

Frame n°2 CA144AAD0445531C2EFE6A0D
0xCA =202 : TH device
0x14 = 0b0001 0100
00 : message typing : standard data
01 : Compression type : Delta compression
0001 : frame index : 1
0x4AAD0445531C2EFE6A0D : Table of compressed count measurements (index frame n°2)

Frame n°3 CA2409A862DBADB5D65B737B
0xCA =202 : TH device
0x24 = 0b0010 0100
00 : message typing : standard data
01 : Compression type : Delta compression
0010 : frame index : 2
0x09A862DBADB5D65B737B : Table of compressed count measurements (index frame n°3)

Frame n°4 CA3437E6CEF40E
0xCA =202 : TH device
0x34 = 0b0011 0100
00 : message typing : standard data
01 : Compression type : Delta compression

0011 : frame index : 4

0x37E6CEF40E : Table of compressed count measurements (index frame n°4)

Global compressed frame

0x 09 50 62 4B 92 54 4A AD 04 45 53 1C 2E FE 6A 0D | 09 A8 62 DB AD B5 D6 5B 73 7B 37 E6 CE F4 0E Little Endian

Hexa	Binary (Little Endian)	Binary (Big Endian)	
09	00001001	10010000	0b1001 : it is a new absolute value
50	01010000	00001010	0b0000 0000 1010 0100 = 0x00A4 = 164 is the new absolute value
62	01100010	01000110	0b01 is the next probability code. So the data is based on the difference with the previous value with difference of ± 1. Associated data is **1** equivalent to -1 (so 0x00A3 = 163) 0b0 correspond to +1 and 0b1 correspond to -1 so here is -1
4B	01001011	11010010	0b01 is the next probability code. Associated data is **1** equivalent to -1 (so 0x00A2 = 162) 0b01 is the next probability code. Associated data is **0** equivalent to +1 (so 0x00A3 = 163) 0b01 is the next probability code. Associated data is **0** equivalent to +1 (so 0x00A4 = 164)
92	10010010	01001001	0b01 is the next probability code. Associated data is **0** equivalent to +1 (so 0x00A5 = 165) 0b01 is the next probability code. Associated data is **0** equivalent to +1 (so 0x00A6 = 166) 0b01 is the next probability code. Associated data is **0** equivalent to +1 (so 0x00A7 = 167)
54	01010100	00101010	0b01 is the next probability code. Associated data is **0** equivalent to +1 (so 0x00A8 = 168) 0b101 is the next probability code. Associated data is **00** equivalent to +2 (so 0x00AA = 170)
4A	01001010	01010010	0b101 is the next probability code. Associated data is **00** equivalent to +2 (so 0x00AC = 172) 0b101 is the next probability code. Associated data is **01** equivalent to +3 (so 0x00AF = 175)
AD	10101101	10110101	0b101 is the next probability code. Associated data is **01** equivalent to +3 (so 0x00B2 = 178)
04	00000100	00100000	0b001 is the next probability code. Associated data is **000** equivalent to +4 (so 0x00B6 = 182) 0b001 is the next probability code. Associated data is **010** equivalent to +6 (so 0x00BC = 188)
45	01000101	10100010	0b001 is the next probability code. Associated data is **011** equivalent to +7 (so 0x00C3 = 195)
53	01010011	11001010	0b001 is the next probability code. Associated data is **010** equivalent to +6 (so 0x00C9 = 201)
1C	00011100	00111000	0b001 is the next probability code. Associated data is **110** equivalent to -6 (so 0x00C3 = 195)
2E	00101110	01110100	0b0001 is the next probability code. Associated data is **1101** equivalent to -12 (so 0x00B6 = 182) 0b0001 is the next probability code. Associated data is **1111** equivalent to -15 (so 0x00A7 = 167)

FE	11111110	11111111	0b11 is the next probability code is “no difference”. No associated data (so 0x00A7 = 167)
6A	01101010	01010110	0b01 is the next probability code. Associated data is **0** equivalent to +1 (so 0x00A8 = 168) 0b101 is the next probability code. Associated data is **10** equivalent to -2 (so 0x00A6 = 166)
0D	00001101	10110000	0b101 is the next probability code. Associated data is **10** equivalent to -2 (so 0x00A4 = 164) To finish last byte 0b000 is added to complete
09	00001001	10010000	0b1001 : it is a new absolute value
A8	10101000	00010101	0b0000 0001 0101 0100 = 0x0154 = 340 is the new absolute value
62	01100010	01000110	0b01 is the next probability code. Associated data is **1** equivalent to -1 (so 0x0153 = 339) 0b01 is the next probability code. Associated data is **1** equivalent to -1 (so 0x0152 = 338)
DB	11011011	11011011	0b01 is the next probability code. Associated data is **1** equivalent to -1 (so 0x0151 = 337) 0b01 is the next probability code. Associated data is **1** equivalent to -1 (so 0x0150 = 336)
AD	10101101	10110101	0b101 is the next probability code. Associated data is **10** equivalent to -2 (so 0x014E = 334) 0b101 is the next probability code. Associated data is **10** equivalent to -2 (so 0x014C = 332)
B5	10110101	10101101	0b101 is the next probability code. Associated data is **10** equivalent to -2 (so 0x014A = 330) 0b101 is the next probability code. Associated data is **10** equivalent to -2 (so 0x0148 = 328)
D6	11010110	01101011	0b101 is the next probability code. Associated data is **11** equivalent to -3 (so 0x0145 = 325)
5B	01011011	10110110	0b101 is the next probability code. Associated data is **10** equivalent to -2 (so 0x0143 = 323)
73	01110011	11001110	0b101 is the next probability code. Associated data is **10** equivalent to -2 (so 0x0141 = 321) 0b01 is the next probability code. Associated data is **1** equivalent to -1 (so 0x0140 = 320) 0b101 is the next probability code. Associated data is **10** equivalent to -2 (so 0x013E = 318)
7B	01111011	11011110	0b11 is the next probability code is “no difference”. No associated data (so 0x013E = 318) 0b11 is the next probability code is “no difference”. No associated data (so 0x013E = 318) 0b01 is the next probability code. Associated data is **1** equivalent to -1 (so 0x013D = 317)
37	00110111	11101100	0b101 is the next probability code. Associated data is **10** equivalent to -2 (so 0x013B = 315) 0b001 is the next probability code. Associated data is **100** equivalent to -4 (so 0x0137 = 311)
E6	11100110	01100111	0b11 is the next probability code is “no difference”. No associated data (so 0x0137 = 311) 0b101 is the next probability code. Associated data is **11** equivalent to

			-3 (so 0x0134 = 308)
CE	11001110	011100011	0b001 is the next probability code. Associated data is **100** equivalent to -4 (so 0x0130 = 304)
F4	11110100	001011111	0b101 is the next probability code. Associated data is **11** equivalent to -3 (so 0x012D = 301) 0b101 is the next probability code. Associated data is **11** equivalent to -3 (so 0x012D = 298)
0E	00001110	011100000	To finish last byte 0b0000 is added to complete

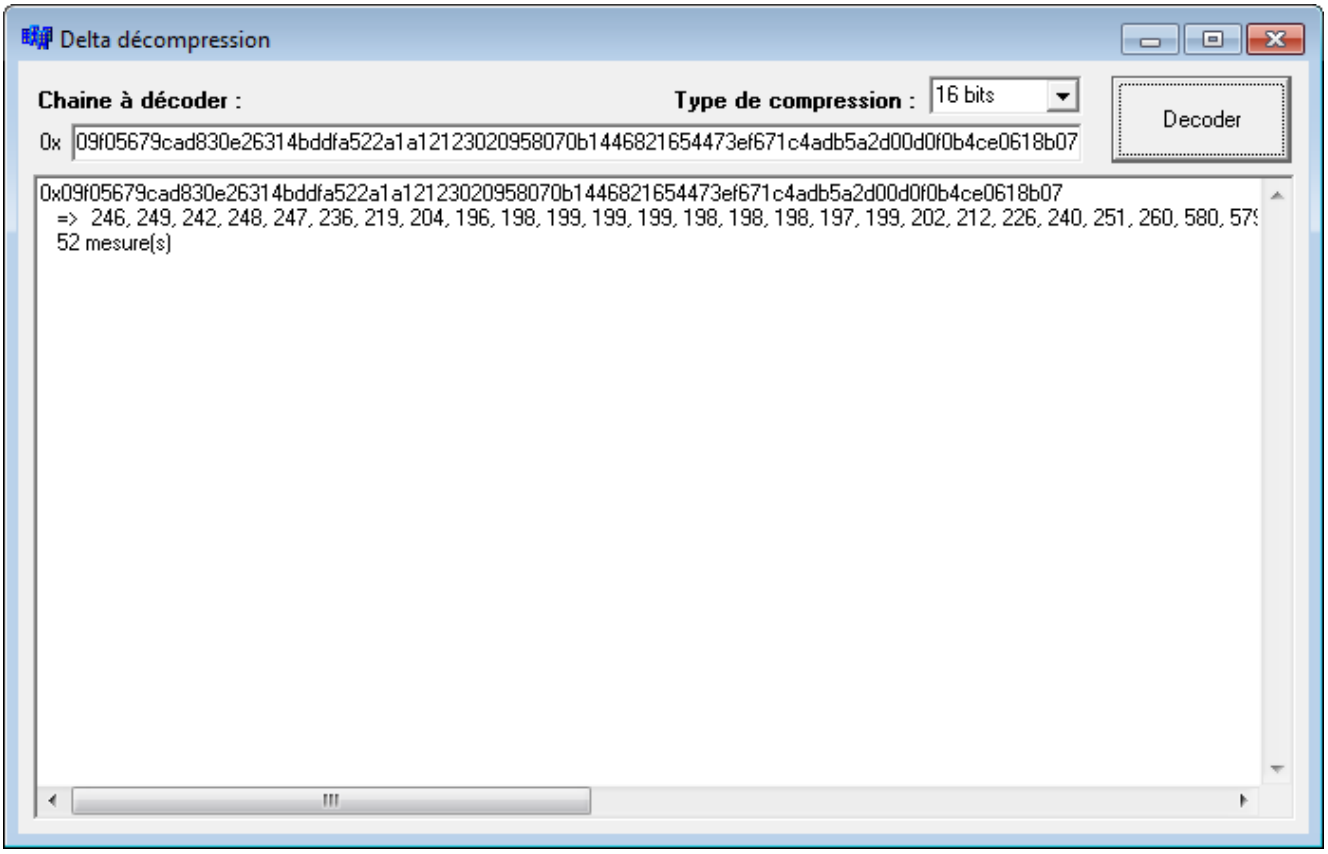
In conclusion, there are 24 temperature measurements (°C) measured during 1 day (24h), in reverse chronological order:
16,4 °C, 16,3 °C, 16,2 °C, 16,3 °C, 16,4 °C, 16,5 °C, 16,6 °C, 16,7 °C, 16,8 °C, 17,0 °C, 17,2 °C, 17,5 °C, 17,8 °C, 18,2 °C, 18,8 °C, 19,5 °C, 20,1 °C, 19,5 °C, 18,2 °C, 16,7 °C, 16,7 °C, 16,8 °C, 16,6 °C, 16,4 °C
And there are 24 humidity measurements (%) measured during 1 day (24h), in reverse chronological order:
34,0 %, 33,9 %, 33,8 %, 33,7 %, 33,6 %, 33,4 %, 33,2 %, 33,0 %, 32,8 %, 32,5 %, 32,3 %, 32,1 %, 32,0 %, 31,8 %, 31,8 %, 31,8 %, 31,7 %, 31,5 %, 31,1 %, 31,1 %, 30,8 %, 30,4 %, 30,1 %, 29,8 %

3.2 Decompression Delta software

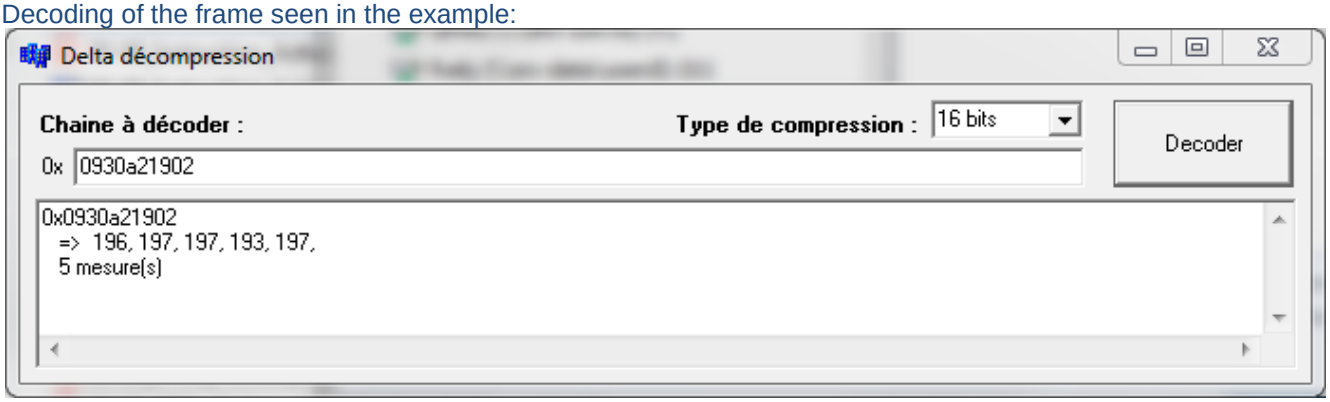
The **Delta Decompression** software can decode compressed data in 16 bits or 32 bits. If the data is on several frames, fill the field to be decoded with the useful data put end to end.
Example, for the following data:

Time	Delay (s)	Header	Data / Decoding
2018-04-18 17:33:44	3	0000	ca44d00d0f0b4ce0618b07
2018-04-18 17:33:37	3.9	0000	ca341654473ef671c4adb5a2
2018-04-18 17:33:30	3.4	0000	ca242123020958070b144682
2018-04-18 17:33:23	3.9	0000	ca1430e26314bddfa522a1a1
2018-04-18 17:33:16	2.8	0000	ca040001131a09f05679cad8

This gives:



Decoding of the frame seen in the example:

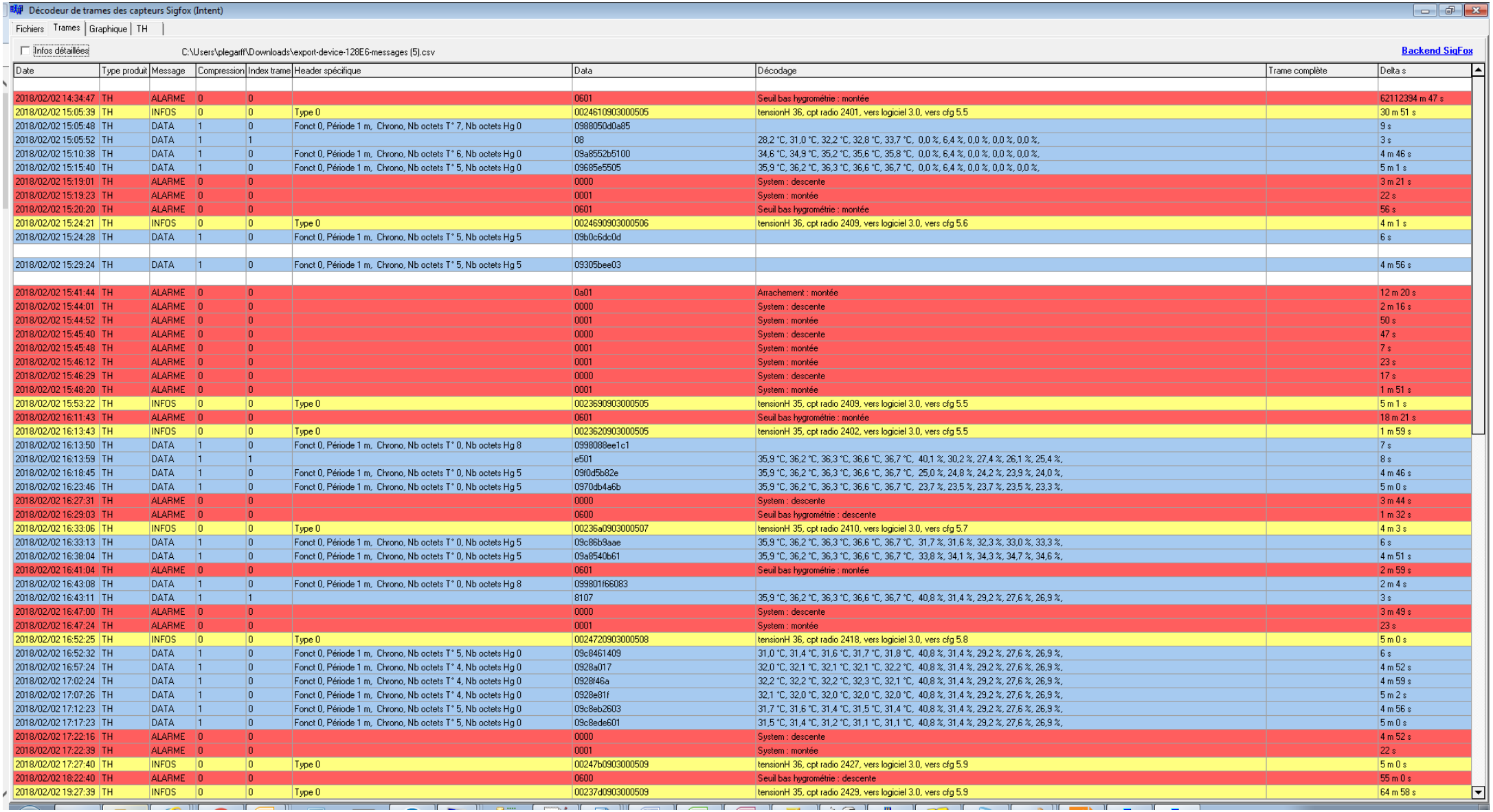


3.3 SigfoxDecoder software

The **SigfoxDecoder** software is a data interpreter. You must select a CSV file exported from the Sigfox backend of the product and double click on it. The frames are displayed.

NB : before export device message from Sigfox Backend, please select at least Country, Data, Device ID, Link quality, Sequence number and Timestamp

The white lines correspond to missing frames, the red ones to the alarms, the yellow ones to the frames of information and the blue ones to the data.



Décodeur de trames des capteurs Sigfox (Intent)

Fichiers | Trames | Graphique | TH

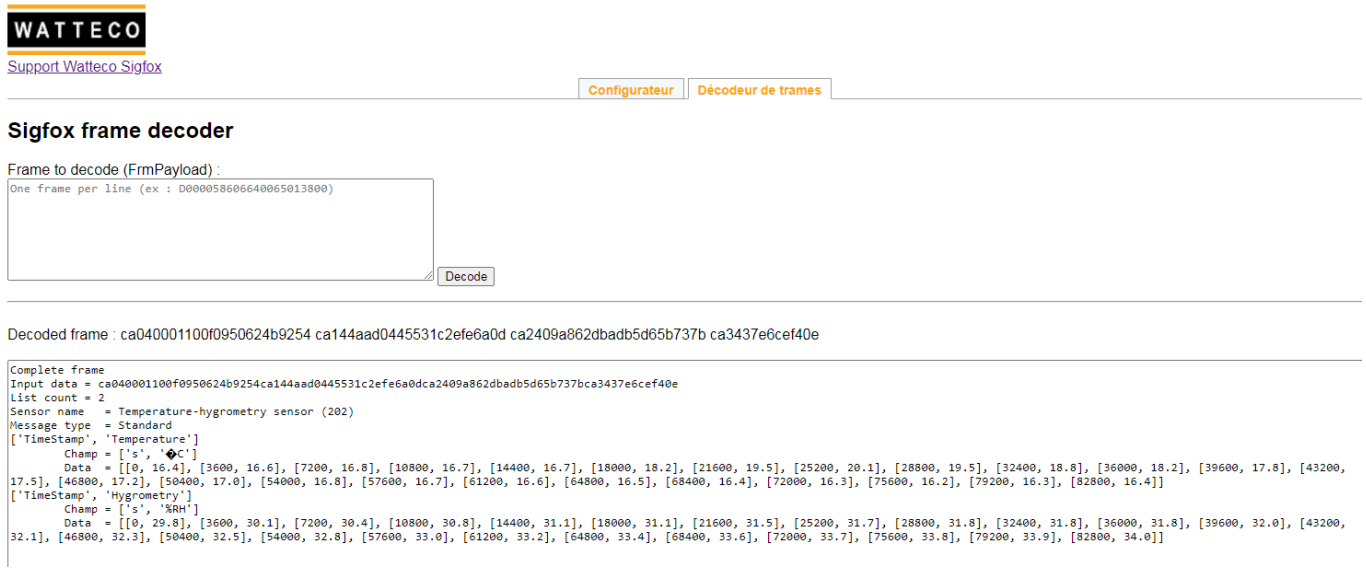
Infos détaillées — C:\Users\plegarff\Downloads\export-device-128E6-messages (5).csv — Backend SigFox

Date	Type produit	Message	Compression	Index trame	Header spécifique	Data	Décodage	Trame complète	Delta s
2018/02/02 14:34:47	TH	ALARME	0	0		0601	Seuil bas hygrométrie : montée		62112394 m 47 s
2018/02/02 15:05:39	TH	INFOS	0	0	Type 0	0024610903000505	tensionH 36, cpt radio 2401, vers logiciel 3.0, vers cfg 5.5		30 m 51 s
2018/02/02 15:05:48	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 7, Nb octets Hg 0	0988050d0a85			9 s
2018/02/02 15:05:52	TH	DATA	1	1		08	28,2 °C, 31,0 °C, 32,2 °C, 32,8 °C, 33,7 °C, 0,0 %, 6,4 %, 0,0 %, 0,0 %, 0,0 %,		3 s
2018/02/02 15:10:38	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 6, Nb octets Hg 0	09a8552b5100	34,6 °C, 34,9 °C, 35,2 °C, 35,6 °C, 35,8 °C, 0,0 %, 6,4 %, 0,0 %, 0,0 %, 0,0 %,		4 m 46 s
2018/02/02 15:15:40	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 5, Nb octets Hg 0	09685e5505	35,9 °C, 36,2 °C, 36,3 °C, 36,6 °C, 36,7 °C, 0,0 %, 6,4 %, 0,0 %, 0,0 %, 0,0 %,		5 m 1 s
2018/02/02 15:19:01	TH	ALARME	0	0		0000	System : descente		3 m 21 s
2018/02/02 15:19:23	TH	ALARME	0	0		0001	System : montée		22 s
2018/02/02 15:20:20	TH	ALARME	0	0		0601	Seuil bas hygrométrie : montée		56 s
2018/02/02 15:24:21	TH	INFOS	0	0	Type 0	0024690903000506	tensionH 36, cpt radio 2409, vers logiciel 3.0, vers cfg 5.6		4 m 1 s
2018/02/02 15:24:28	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 5, Nb octets Hg 5	09b0c6dc0d			6 s
2018/02/02 15:29:24	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 5, Nb octets Hg 5	09305bee03			4 m 56 s
2018/02/02 15:41:44	TH	ALARME	0	0		0a01	Arrachement : montée		12 m 20 s
2018/02/02 15:44:01	TH	ALARME	0	0		0000	System : descente		2 m 16 s
2018/02/02 15:44:52	TH	ALARME	0	0		0001	System : montée		50 s
2018/02/02 15:45:40	TH	ALARME	0	0		0000	System : descente		47 s
2018/02/02 15:45:48	TH	ALARME	0	0		0001	System : montée		7 s
2018/02/02 15:46:12	TH	ALARME	0	0		0001	System : montée		23 s
2018/02/02 15:46:29	TH	ALARME	0	0		0000	System : descente		17 s
2018/02/02 15:48:20	TH	ALARME	0	0		0001	System : montée		1 m 51 s
2018/02/02 15:53:22	TH	INFOS	0	0	Type 0	0023690903000505	tensionH 35, cpt radio 2409, vers logiciel 3.0, vers cfg 5.5		5 m 1 s
2018/02/02 16:11:43	TH	ALARME	0	0		0601	Seuil bas hygrométrie : montée		18 m 21 s
2018/02/02 16:13:43	TH	INFOS	0	0	Type 0	0023620903000505	tensionH 35, cpt radio 2402, vers logiciel 3.0, vers cfg 5.5		1 m 59 s
2018/02/02 16:13:50	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 0, Nb octets Hg 8	0998088ee1c1			7 s
2018/02/02 16:13:59	TH	DATA	1	1		e501	35,9 °C, 36,2 °C, 36,3 °C, 36,6 °C, 36,7 °C, 40,1 %, 30,2 %, 27,4 %, 26,1 %, 25,4 %,		8 s
2018/02/02 16:18:45	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 0, Nb octets Hg 5	09f0d5b82e	35,9 °C, 36,2 °C, 36,3 °C, 36,6 °C, 36,7 °C, 25,0 %, 24,8 %, 24,2 %, 23,9 %, 24,0 %,		4 m 46 s
2018/02/02 16:23:46	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 0, Nb octets Hg 5	0970db4a6b	35,9 °C, 36,2 °C, 36,3 °C, 36,6 °C, 36,7 °C, 23,7 %, 23,5 %, 23,7 %, 23,5 %, 23,3 %,		5 m 0 s
2018/02/02 16:27:31	TH	ALARME	0	0		0000	System : descente		3 m 44 s
2018/02/02 16:29:03	TH	ALARME	0	0		0600	Seuil bas hygrométrie : descente		1 m 32 s
2018/02/02 16:33:06	TH	INFOS	0	0	Type 0	00236a0903000507	tensionH 35, cpt radio 2410, vers logiciel 3.0, vers cfg 5.7		4 m 3 s
2018/02/02 16:33:13	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 0, Nb octets Hg 5	09c86b9aae	35,9 °C, 36,2 °C, 36,3 °C, 36,6 °C, 36,7 °C, 31,7 %, 31,6 %, 32,3 %, 33,0 %, 33,3 %,		6 s
2018/02/02 16:38:04	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 0, Nb octets Hg 5	09a8540b61	35,9 °C, 36,2 °C, 36,3 °C, 36,6 °C, 36,7 °C, 33,8 %, 34,1 %, 34,3 %, 34,7 %, 34,6 %,		4 m 51 s
2018/02/02 16:41:04	TH	ALARME	0	0		0601	Seuil bas hygrométrie : montée		2 m 59 s
2018/02/02 16:43:08	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 0, Nb octets Hg 8	099801f66083			2 m 4 s
2018/02/02 16:43:11	TH	DATA	1	1		8107	35,9 °C, 36,2 °C, 36,3 °C, 36,6 °C, 36,7 °C, 40,8 %, 31,4 %, 29,2 %, 27,6 %, 26,9 %,		3 s
2018/02/02 16:47:00	TH	ALARME	0	0		0000	System : descente		3 m 49 s
2018/02/02 16:47:24	TH	ALARME	0	0		0001	System : montée		23 s
2018/02/02 16:52:25	TH	INFOS	0	0	Type 0	0024720903000508	tensionH 36, cpt radio 2418, vers logiciel 3.0, vers cfg 5.8		5 m 0 s
2018/02/02 16:52:32	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 5, Nb octets Hg 0	09c8461409	31,0 °C, 31,4 °C, 31,6 °C, 31,7 °C, 31,8 °C, 40,8 %, 31,4 %, 29,2 %, 27,6 %, 26,9 %,		6 s
2018/02/02 16:57:24	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 4, Nb octets Hg 0	0928a017	32,0 °C, 32,1 °C, 32,1 °C, 32,1 °C, 32,2 °C, 40,8 %, 31,4 %, 29,2 %, 27,6 %, 26,9 %,		4 m 52 s
2018/02/02 17:02:24	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 4, Nb octets Hg 0	0928f46a	32,2 °C, 32,2 °C, 32,2 °C, 32,3 °C, 32,1 °C, 40,8 %, 31,4 %, 29,2 %, 27,6 %, 26,9 %,		4 m 59 s
2018/02/02 17:07:26	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 4, Nb octets Hg 0	0928e81f	32,1 °C, 32,0 °C, 32,0 °C, 32,0 °C, 32,0 °C, 40,8 %, 31,4 %, 29,2 %, 27,6 %, 26,9 %,		5 m 2 s
2018/02/02 17:12:23	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 5, Nb octets Hg 0	09c8eb2603	31,7 °C, 31,6 °C, 31,4 °C, 31,5 °C, 31,4 °C, 40,8 %, 31,4 %, 29,2 %, 27,6 %, 26,9 %,		4 m 56 s
2018/02/02 17:17:23	TH	DATA	1	0	Fonct 0, Période 1 m, Chrono, Nb octets T° 5, Nb octets Hg 0	09c8ede601	31,5 °C, 31,4 °C, 31,2 °C, 31,1 °C, 31,1 °C, 40,8 %, 31,4 %, 29,2 %, 27,6 %, 26,9 %,		5 m 0 s
2018/02/02 17:22:16	TH	ALARME	0	0		0000	System : descente		4 m 52 s
2018/02/02 17:22:39	TH	ALARME	0	0		0001	System : montée		22 s
2018/02/02 17:27:40	TH	INFOS	0	0	Type 0	00247b0903000509	tensionH 36, cpt radio 2427, vers logiciel 3.0, vers cfg 5.9		5 m 0 s
2018/02/02 18:22:40	TH	ALARME	0	0		0600	Seuil bas hygrométrie : descente		55 m 0 s
2018/02/02 19:27:39	TH	INFOS	0	0	Type 0	00237d0903000509	tensionH 35, cpt radio 2429, vers logiciel 3.0, vers cfg 5.9		64 m 58 s

3.4 Online server with Python Codec

A server is configured at watteco (https://support.nke.fr/sigfox/SigfoxFrameDecode.php).
Insert frame to decode in windows and click on *Decode* button.

Result is displayed with all decoded datas:



The files to take into account are:

- SigfoxData_ X_YY.py for extract a compressed data from a bit stream thanks to an opcode
- SigfoxFifo_ X_YY.py for management of a bit FIFO
- SigfoxOpcode_ X_YY.py for extract an opcode from a bit stream
- SigfoxPayload_ X_YY.py for decode Sigfox payload
- SigfoxRawDecoder_ X_YY.py for decode Sigfox raw payload
- SigfoxZipDecoder_ X_YY.py for decode Sigfox compressed payload

SigfoxMain.py is used as example for Python platform.
These files are compressed in Python decoder V_X_YY.zip file

3.5 « DeltaCompressionLib.dll » DLL (Dynamic Link Librairy) use

A DLL for decoding data is available: DeltaCompressionLib.dll

The DLL uses only 3 functions to perform the decompression operations of the data received in SigFox.
Important: All function calls of the Dll are done in **stdcall**.

The first is the initialization function of the dll:

*bool **bFDeltaCompressionLibInitialization**(unsigned char);*

This function accepts a single input parameter which is the size of the data to be decompressed.

Thus, the possible values are 8, 16 or 32 bits.
This function responds "true" if the initialization was successful.
The main function of the decompression of the DLL is:

*bool **bFDeltaCompressionLibUnZip**(void *, unsigned short int *, const unsigned char [], unsigned short int *)*

The first parameter is the destination array, which can be an array of bytes, words, or long words.
It must be sized in order to collect the data to decompress.
The second parameter has two uses:
Enter the size of the data array passed in the first parameter. The size is expressed in number of elements of the table.
At the output of the function, the size indicates the number of elements actually converted.
The third parameter is the table of compressed data.
The last parameter has two uses:

- Indicate the size of the data (in bytes) contained in the table passed in 3rd parameter.
- At the output of the function, the number of bytes remaining to decompress

This function responds "true" if the decompression was successful.

The last function of initialization of the DLL is:
*bool **bFDeltaCompressionLibFinalization**(void);*

This function is to be called in order to complete the decompression phase.
This function responds "true" if the finalization was successful.

3.5.1 Example of use on a C ++ code [C++ Builder 6.0]

3.5.1.1 Dynamic loading of the DLL into memory

```
//---------------------------------------------------------------------------
HINSTANCE hinstDLL = 0;
bool ( __stdcall * bpFDeltaCompressionLibInitialization )( unsigned char );
bool ( __stdcall * bpFDeltaCompressionLibUnZip )( void * , unsigned short int *, unsigned char [], unsigned short int * );
bool ( __stdcall * bpFDeltaCompressionLibFinalization )( void );
//---------------------------------------------------------------------------
void __fastcall TFMain::FormCreate( TObject *Sender )
{
 // On charge le librairie en mémoire / We load the library in memory
 if( ( hinstDLL = LoadLibrary( "DeltaCompressionLib.dll" ) ) )
  {
   bpFDeltaCompressionLibInitialization = ( bool ( __stdcall * )( unsigned char ) )GetProcAddress( hinstDLL, "bFDeltaCompressionLibInitialization" );
   bpFDeltaCompressionLibUnZip          = ( bool ( __stdcall * )( void * , unsigned short int *, unsigned char [], unsigned short int * ) )GetProcAddress( hinstDLL, "bFDeltaCompressionLibUnZip" );
   bpFDeltaCompressionLibFinalization   = ( bool ( __stdcall * )( void ) )GetProcAddress( hinstDLL, "bFDeltaCompressionLibFinalization" );
  }
}
```

3.5.1.2 Using the decompression functions

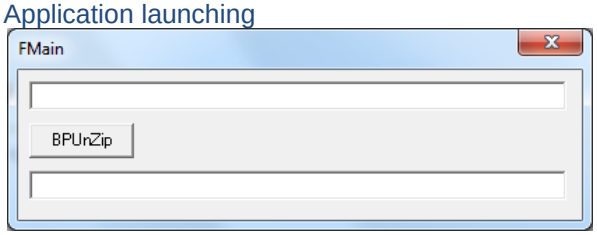
```
//---------------------------------------------------------------------------
void __fastcall TFMain::BPUnZipClick(TObject *Sender)
{
 unsigned char tucInput[ 64 ];
 unsigned short int uiInputSize;
 unsigned short int tuiOutput[ 512 ];
 unsigned short int uiOutputSize;
 char tcView[ 1024 ];

 // Intégrité / Integrity
 if(   ( bpFDeltaCompressionLibInitialization == NULL )
    || ( bpFDeltaCompressionLibUnZip          == NULL )
    || ( bpFDeltaCompressionLibFinalization   == NULL ) )
  {
   MessageDlg( "Les fonctions de la Dll n'ont pas été chargées",
               mtError,
               TMsgDlgButtons() << mbOK,
               0 );
   return;
  }
 // On regarde s'il y a qques chose à décompresser / We look for something to decompress
 if( eInput->Text == "" )
  {
   MessageDlg( "Il n'y a pas de données à décompresser",
               mtWarning,
               TMsgDlgButtons() << mbOK,
               0 );
   return;
  }
 // On ouvre la session en mode 16 bits / We open the session in 16-bit mode
 bpFDeltaCompressionLibInitialization( 16 );
 // On formate
 uiInputSize  = HexToBin( eInput->Text.c_str(), tucInput, eInput->Text.Length() );
 // On indique la taille du buffer disponible pour les données à décompresser
 // We indicate the size of the buffer available for the data to decompress
 // ... et en fonction de la taille du type de donnée
 // ... and depending on the size of the data type
 uiOutputSize = sizeof( tuiOutput ) / sizeof( unsigned short int );
 // On appelle la fonction de décompression / We call the decompression function
 bpFDeltaCompressionLibUnZip( tuiOutput, &uiOutputSize, tucInput, &uiInputSize );
 // On ferme la session / We close the session
 bpFDeltaCompressionLibFinalization;
 // On convertit en qque chose de visualisable / We convert to something visualizable
 BinToHex( ( char * )tuiOutput, tcView, uiOutputSize * sizeof( unsigned short int ) );
 // On affiche le résultat / We display the result
 eOutput->Text = tcView;
}
```

3.5.1.3 DLL unloading

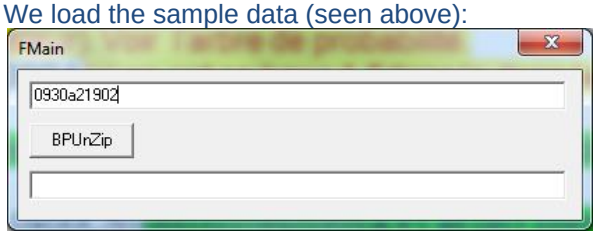
```
//---------------------------------------------------------------------------
void __fastcall TFMain::FormClose(TObject *Sender, TCloseAction &Action)
{
 if( hinstDLL ) FreeLibrary( hinstDLL );
}
//---------------------------------------------------------------------------
```

3.5.1.4 Software test

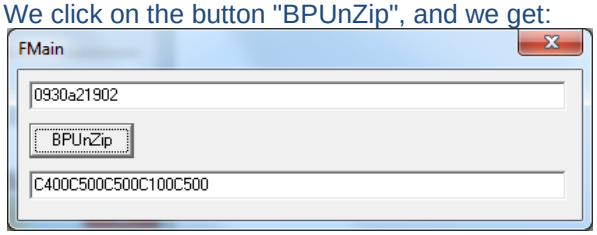
Application launching



We load the sample data (seen above):



We click on the button "BPUnZip", and we get:



The result seen above is identical and gives:
0x00C4 = 19,6°C
0x00C5 = 19,7°C
0x00C5 = 19,7°C
0x00C1 = 19,3°C
0x00C5 = 19,7°C

APPENDICES

Annex 1: Document evolutions

Date	Revision	Document evolutions	Writer
2018/04/19	0	Creation	FV
2018/11/28	1	Adding information about the DLL decompression	SD
2018/11/30	2	Adding another decompressing example	FV
2019/01/25	3	Adding another decompressing example (TH)	FV
2019/06/05	4	Adding Python Codec	FV
2022/04/25	5	Update support page URL	PLG