ECB/ETS - PC-protocol 1.0
15.11.2010

1 General

Each line is wrapped with <STX> and <ETX>. These are ascii «start of text» (0x02) and «end of text» (0x03)

All fields are followed by <TAB> (0x09) which can be used to split the string into fields.

All fields can come in arbitrary order.

Additional fields may occur. The receiving software should ignore fields that are unknown.

> ***Please note!***
> *All examples are given in mode 0. That is /PP0. Variants will occur in other modes. Please refer to original protocol for further details.*

1.1 Status

Each 4 seconds when no other incident has occurred, the ECB will send a status message. In equipment with on board GPRS-module, this interval is increased to 60 seconds.

Sample:
<STX>IESD-HW1-SW4-V1.1<TAB>M1-740<TAB>W09:55:19.036<TAB>C0<TAB>X0<TAB>Y870100005<TAB>A116-151-+999-94<TAB>H01310<TAB><ETX>

It can consist of, but not limited to these fields:

"I" - info about the timing unit.
I<type>-HW<hwtype>-SW<swtype>-V<sw-version>

- type: ECB, ETS, ESD etc...
- HWtype: Which version of harware we're running (you don't need this)
- SWtype: What type of software (you don't need this)
- Version: Which SW-revision

M: Incidents kept in unit. M<first incident today>-<next incident>
W: Clock. The units time when the message is sendt
C: Postcode.
X: Operation mode. This is the number you set with /PP
Y: Serial number
A: Unit health. A<battery voltage>-<charger V>-<mA going in or out from battery>-<battery %>
H: More status

Habcde

- a: 0 = normal mode. 1= Turning off
- b: Loop 1 status. 0 = OK. 1 = no loop. 2 = loop is not operating correctly in some way. 3 =
- Loop has been malfunctioning, but now looks just fine.
- c: Loop 2. Same as b
- d: radio status: 1 = OK. 0 = radio not present
- e: gprs signal quality. 0-4. 0 = no signal or gprs not present

1.2 Emitag passing

<STX>N5<TAB>Y870100005<TAB>M740<TAB>C67<TAB>E09:55:30.112<TAB>T00:00:00.124<TAB>O0<TAB><ETX>

N Tag number
Y Serial number on unit
M incident number
C Postcode
E Time of incident
T Elapsed time since emitag last saw zero-post/start. Wraps at 4:39:37:215(!) (3 bytes of milliseconds)
O Number of radio transmissions before complete package

1.3 Start/finish

Incidents occurring on the start and finish connectors is shown to the PC as:
<STX>F1-1<SPACE>09:18:10.852<TAB>C67<TAB>M2094<TAB>W09:18:10.940<TAB><ETX>
<STX>F1-0<SPACE>09:18:10.998<TAB>C67<TAB>M2095<TAB>W09:18:11.128<TAB><ETX>
<STX>F0-1<SPACE>09:18:11.702<TAB>C67<TAB>M2096<TAB>W09:18:11.790<TAB><ETX>
<STX>F0-0<SPACE>09:18:11.748<TAB>C67<TAB>M2097<TAB>W09:18:11.930<TAB><ETX>

F: Change in gate status. Format: Fg-s HH:MM:SS.mmm
g: 0 = start. 1 = finish.
s: 1 = gate is short circuited
HH:MM:SS.mmm – time of the incident
C: Postcode
M: Incident number
W: Time on unit when message is sendt

1.4 Key Pads

<STX>K3-87654321-09:41:07.444<TAB>M2094<TAB>W09:41:07.548<TAB><ETX>
<STX>K3-22334455-09:41:21.412<TAB>M2095<TAB>W09:41:21.516<TAB><ETX>

Kk-dddddddd-HH:MM:SS.mmm
- k : Keypad number
- dddddddd: data from keypad
- HH:MM:SS.mmm time received in milliseconds

M: Incident number
W: Time on unit when message is sendt

1.5 Dump Emitag

<STX>N3<TAB>W10:15:01.531<TAB>V299-1829<TAB>S3002516<TAB>RemiTag v5<TAB>X0<TAB>P0-0-00:00:00.000<TAB>P1-67-00:00:00.128<TAB>P2-67-00:11:27.304<TAB>P3-67-116:48:03.805<TAB>P4-67-117:04:26.554<TAB>P5-67-117:04:57.054<TAB>P6-252-117:08:33.116<TAB><ETX>

N: Tag number
W: Time on unit when message is sent
V: Vxxx-yyy info about the tag. (you don't need this)
S: Tag serial number (originally given number) can be the same as N. But N can be a given
«nickname» e.g. To match starting number.
R: Text following R is a free text which is programmed into the tag.
P: Pn-p-[H]HH:MM:SS.mmm
- n: post number
- p: Post code for this post
- [H]HH:MM:SS.mmm: Time when emitag Passed the post. Wraps at 999 Hours.

2 Commands

To change settings in the timing unit, or to spool messages etc., they can receive commands over USB, RS232 and even RS485 where available.

These commands are somewhat different than the packages we receive from the unit. For

a start, commands is not enclosed by the <STX>...<ETX> pair. It ALWAYS starts with a / (Ascii 0x2F) and ends with <CR><LF> (Ascii 0x0D and 0x0A).

Between each byte sent to the unit, you should allow a 5 ms break to ensure stable reception.

2.1 Set Clock

Command: /SChh:mm:ss<CR><LF>

Sets the clock to hh:mm:ss.

2.1.1 Set Clock on Pulse

Command: /SCPhh:mm:ss<CR><LF>

Sets the clock to hh:mm:ss when either the start gate or the photofinishing terminal are short circuited .

2.2 Set CODE

Command: /SCc[cc]<CR><LF>

Set the code to integer c[cc]. The code should be in the range between 65 and 239. Confirm Emit for recommendation on how to choose codes.

I addition

- start has code 0 (caution; emitag interprets this as a new race)
- finish has code 248, which triggers the second loop.

2.3 Spool contents

The units send incidents (passings, start gate+++) as the happens, but occasionally we need to ask the units to resent one or more incidents.

2.3.1 Spool all

Command: /QD<CR><LF>

This will ask the unit for all its content. Use this with caution. It might be lots and lots of data. Remember the units remembers approximately the last 260.000 incidents!!

2.3.2 Spool new incidents today

Command: /QM<CR><LF>

This will ask the unit for all incidents “today”. That is; Since last time the unit was turned on, or, if is has been switched on overnight, since midnight.

2.3.3 Spool from message number

Command: /QFm<CR><LF>

This will ask the unit for all incidents with message number equal or greater than *m*.

2.3.4 Spool a specific message number

Command: /QCm<CR><LF>

This will ask the unit for one incident with message number *m*.

2.4 Clear unit memory

Command: /CL<CR><LF>

This command tells the unit to forget all incidents and start collecting at message number 1 again..

Caution! Never use this during a race!

2.5 Force status message

Command: */ST<CR><LF>*

This will force the unit to send a status message.

3 Connection info

Type	Speed (baud)	Data Bits	Stop Bits	Parity	Flow Control
USB	115200	8	1	None	None
RS232	9600				
RS485	19200				