+D/2

# Transmission protocol

## Summary

# 1. Principle of functioning

There are three basic functioning modes:

1) Off-line data transmission
2) On-line data transmission
3) Interactive PC-REI2 communication

## *1.1. Off-line data transmission*

Off-line data transmission consists of the transference of data during a timing session after an interval of time. Naturally the timekeeper must enter the relative 'menu' on REI 2 to request data transmission.

The transfer of various types of data can be chosen:

- Net times
- Event times
- Non-starters
- Non-finishers
- Disqualified
- Speed

Each item can be filtered on the basis of a run, group or particular time interval (i.e. downloads all the event times of run 2 of those in group 3 between 12:00:00.0000 and 13:00:00.0000).
The protocol used for data transmission from REI2 to PC is ***Extended REI2 protocol*** (see description below) with the mode flag equal to 'F'.

## *1.2. On-line data transmission*

On-line data transmission consists of the transmission from REI2 to PC of the whole operation of acquisition, correction and annulment of times performed by the timekeeper on the machine (in practice, all the information given on the printout is transmitted). Once enabled, transmission takes place completely autonomously each time a time is acquired or modified. Each record transmitted is identified by a counter (from 0 to 99999 with *wrap-around*), which goes up automatically.

Also in this case, the protocol used for transmission is ***Extended REI2 protocol*** (see description below) with the mode flag equal to 'O'.

It is also possible to activate an output on the serial line corresponding to the output of the main displayboard, with intervals which can be set; in this case the transmission protocol used for transmission is ***Reduced REI2 protocol*** (see description below).

## *1.3. Interactive PC-REI2-PC communication*

The requests the PC can forward to REI2 can be subdivided into four types:

1) 'Static' requests
2) 'Dynamic' requests

3) Status requests
4) Break requests

Each request made by the PC to REI2 is identified not only by the type of request but also by a 5-figure identification number. The number concerned is used by the reply in such a way that the pairing is unique.

### 1.3.1. 'Static' request

A static request is made each time the PC needs to access one or more elements of the chronometer's database. The ***'Static' request,*** protocol allows you to filter the elements of the database according to the requirements of the moment. The possibility of obtaining running times in reply is not provided for in this type of request.

The ***'Static' request*** protocol also offers the possibilty of making requests to which the chronometer must reply with a number of records. (for example, a request for all the NPs of run 1). In this case the identification number given in the replies remains the same for all the replies corresponding to the same request.

The reply to a static request is in conformity with the ***Extended REI2 protocol***. The reply to a ***'Static' request*** differs from autonomous on-line and off-line transmission in the initial protocol identification character.

### 1.3.2. 'Dynamic' request

A 'dynamic' request allows you to activate running times on the specified serial line with the possibility of defining the time interval between one transmission and the next (from 1/100s a 999,99s, in steps of 1/100s) .

To guarantee maximum flexibility of use, the chronometer puts in line a Tout running time specified as follows:

Tout=Tnow-Tev-Taux

where

Tnow = present time of the machine (real time, as at initial synchronisation)

Tev = event time. The event time should be specified as type of event, competitor number and run. It is also possible not to specify this parameter (simply by assigning 0 to the athlete's number). In this case, REI2 assumes Tev=0.

Taux = generic time, communicated to REI2 by the PC. It is also possible to specify a negative sign for Taux

Taux = therefore allows you to 'shift' the running time of a competitor as desired by a fraction of a second. This is particularly useful during showing on TV.

REI2 can manage a maximum of 2 running times simultaneously.

The output of running times occurs in accordance with the ***'Dynamic' reply*** protocol.

### 1.3.3. Status request

A status request allows you to obtain information regarding the settings parameters of the machine (status of lines, line disactivation times, program set, etc.).

The request contains a code relative to the parameter you wish to check. The reply takes place in accordance with the ***Status reply*** protocol and although maintaining the same structure and dimension, can present differences depending on the parameter requested.

### 1.3.4. Request for break, transmission suspension/resumption, record repetition

A break request allows you to annul the reply to a particular static request. The reply to be interrupted is identified by its identification number. A reply to the break request is not required.

This command can be particularly useful for interrupting transmission after a static request for which there is more than one reply record.

The request for suspension and resumption of the communication makes it possible to implement a SW Xon/Xoff protocol. During suspension, the items of information are put in a queue. If the queue is full, the following records are lost. This possibility can easily be identified subsequently through requests which receive no reply or discontinuity in the incremental identification number for on-line information.

Any replies to requests which are lost can be recovered by repeating the request.

Lost on-line transmission records can be recovered by sending a request for repetition of the record.

### 1.3.5. Error in the request

If a request contains a syntactic error or cannot be interpreted correctly by REI2, a general error code of the consecutive number of the request in which the error was found is sent back.

## *1.4. Insertion of time events*

The request for insertion of time events, available from software version 1.07 on, allows insertion and annulment of time events and insertion of NP and NA indications in the REI2 database using a serial connection. The protocol specifications are given in chap. 4.3 Event transmission from PC to REI2 on p.29. The physical channel assigned to insertion from PC is 900.

REI2 does not do any kind of check on the congruity of data sent for insertion. All checking must be made with procedures external to the chronometer.

## 2. General information about the REI2- PC protocol

Some general information about the implementation of the REI2-PC protocol:

- The transmission of commands and the reception of replies contain only ASCII codes.
- Each request or reply has an initial header for each particular protocol with an ASCII control code (code character < 0x20 (space).
- Each request or reply ends with a 'carriage return' (CR, 0x0d).
- After the initial character which identifies the protocol, each request and reply (with the exception of 'dynamic' reply) has two characters to identify the type of device (REI2) and the device's address. This function makes it possible to connect a number of devices on the same communication line.

Each request can send the reply on the same serial channel on which it was received, on one of the two channels you can choose to make available (independently of which channel is used for requests) or on both serial outputs. It is possible to pilot the displayboard output by using the dynamic request identifier "T" requested. (see chap. 4.2.2 'Dynamic' request on p.25).

## 3. Observations regarding connection to TV

When running times are displayed (e.g. connection to TV) it is advisable to use the tick or dynamic responses and not on-line data as the latter might have a delay of a few tenths of a second.
In the following programs different types of information can be emitted for the tick simultaneously:

- PARALLEL SLALOM or DUAL TIMER: if the two competitors have already started the race, it is the running time for each track. When a competitor finishes, instead of running time tick output is the gap, positive or negative, between him/her and the competitor still racing.
- SHOW JUMPING: as well as running time, REI2 also sends the athlete's penalty.
- SIMPLE STOPWATCH: after the first competitor has finished, if "Displayboard block after first finish: Active" is set, the output is the net time of the first competitor plus the gap.

- When LinkPod or EncRadio devices are used with tick output, it is advisable to enter a delay time (for LinkPod 120ms is recommended, for EncRadio 200ms) which can be set in "Serial ports setup" by pressing <ALT>+<F2>.

# 4. Protocol detail

## *4.1. Data transmission from REI2 to PC*

### 4.1.1. Extended REI2 protocol

**`Total 52 bytes`**

| Description | N° bytes | ASCII Code (Dec, Hex) | Notes | |
|---|---|---|---|---|
| **DLE** | 1 | 16,10h | Protocol identifier | |
| Chronometer identifier | 1 | | R=REI2 | |
| Device address | 1 | 32,20h (space) | Reserved for future uses | |
| Dummy char | 1 | 32,20h (space) | For compatibility with 'static' replies | |
| Program in use | 1 | | S=Single Starts<br>G=Group starts<br>B=Simple stopwatch<br>P=Parallel<br>I=Show jumping<br>N=Swimming<br>T=Track Chase<br>O=Pc OnLine | |
| Mode | 1 | | O=OnLine<br>F=OffLine | |
| Progressive counter | 6 | | From 1 to 999999, with wrap around | |
| Competitor N° | 5 | | 00000<= N <=59999 | Zero in the case of PC OnLine without competitor number |
| Group/Category | 3 | | 000<= Ng <=199 | If the Group/Category is equal to zero or the Groups/Categories have not been defined or the information filtered by group is not being downloaded (it is not always downloaded because the competitor could belong to more than one group) |
| Run/Trial | 3 | | 001<= Nm <=250 | For horse racing the phase 1 run can range from 1 to 99, the phase 2 run from 100 to 198 |
| Physical channel[1] | 3 | | 000<= Physical channel <=999 | If the physical channel does not exist the output is " "<br>Channel Channel Description |

[1] Note: for the simple stopwatch and the parallel, the output has a different meaning:

- PARALLEL: The output data item is the progressive number of direct contests

| | | |
|---|---|---|
| Main Lines | 000 | Start Line |
| | 001 | Lap Line |
| | 015 | Stop Line |
| | 016 | Aux Line |
| | 100 | Start Key |
| | 101 | Lap key |
| | 115 | Stop key |
| | 116 | Aux key |
| | 200 | Auto Start |
| | 300 | Start (Manual keying in) |
| | 301 | Lap (Manual keying in) |
| | 315 | Stop (Manual keying in) |
| Pod Inputs | 400..407 | POD 0 |
| | 410..417 | POD 1 |
| | 420..427 | POD 2 |
| | 430..437 | POD 3 |
| | 440..447 | POD 4 |
| | 450..457 | POD 5 |
| | 460..467 | POD 6 |
| | 470..477 | POD 7 |
| | 480..487 | POD 8 |
| | 490..497 | POD 9 |
| Inputs Via Radio | 500 | RADIO START |
| | 501 | RADIO LAP 1 |
| | 502 | RADIO LAP 2 |
| | 503 | RADIO LAP 3 |
| | 504 | RADIO LAP 4 |
| | 505 | RADIO LAP 5 |
| | 506 | RADIO LAP 6 |
| | 507 | RADIO LAP 7 |
| | 508 | RADIO LAP 8 |
| | 509 | RADIO LAP 9 |
| | 510 | RADIO LAP A |
| | 511 | RADIO LAP B |
| | 512 | RADIO LAP C |
| | 513 | RADIO LAP D |
| | 514 | RADIO LAP E |
| | 515 | RADIO STOP |
| Retrieved from Encoder | 600 | ENC START |
| | 601 | ENC LAP 1 |
| | 602 | ENC LAP 2 |
| | 603 | ENC LAP 3 |
| | 604 | ENC LAP 4 |
| | 605 | ENC LAP 5 |

| | | | |
|---|---|---|---|
| | | 606 | ENC LAP 6 |
| | | 607 | ENC LAP 7 |
| | | 608 | ENC LAP 8 |
| | | 609 | ENC LAP 9 |
| | | 610 | ENC LAP A |
| | | 611 | ENC LAP B |
| | | 612 | ENC LAP C |
| | | 613 | ENC LAP D |
| | | 614 | ENC LAP E |
| | | 615 | ENC STOP |
| | Imputed by RadioModem | 700 | |
| | Serial A RADIO Inputs | 516 | RADIO START |
| | | 517 | RADIO LAP 1 |
| | | 518 | RADIO LAP 2 |
| | | 519 | RADIO LAP 3 |
| | | 520 | RADIO LAP 4 |
| | | 521 | RADIO LAP 5 |
| | | 522 | RADIO LAP 6 |
| | | 523 | RADIO LAP 7 |
| | | 524 | RADIO LAP 8 |
| | | 525 | RADIO LAP 9 |
| | | 526 | RADIO LAP A |
| | | 527 | RADIO LAP B |
| | | 528 | RADIO LAP C |
| | | 529 | RADIO LAP D |
| | | 530 | RADIO LAP E |
| | | 531 | RADIO STOP |
| | Serial B RADIO Inputs | 532 | RADIO START |
| | | 533 | RADIO LAP 1 |
| | | 534 | RADIO LAP 2 |
| | | 535 | RADIO LAP 3 |
| | | 536 | RADIO LAP 4 |
| | | 537 | RADIO LAP 5 |
| | | 538 | RADIO LAP 6 |
| | | 539 | RADIO LAP 7 |
| | | 540 | RADIO LAP 8 |
| | | 541 | RADIO LAP 9 |
| | | 542 | RADIO LAP A |
| | | 543 | RADIO LAP B |
| | | 544 | RADIO LAP C |
| | | 545 | RADIO LAP D |
| | | 546 | RADIO LAP E |
| | | 547 | RADIO STOP |
| | Imputed by PC | 900 | PC |
| Logical Channel | 3 | 000<= Logical channel <=255 | |

| | | | | |
|---|---|---|---|---|
| | | | 000=START | |
| | | | 001..240= LAP n | For simple stopwatch: SPLITs with LAP have a range from 001 to 200,<br>SPLITs without LAP have a range from 201 to 240.<br>If the maximum number is exceeded the previous item of data is cancelled. The finishes following the first are considered as LAP. |
| | | | 248=REAL_START_CBASE | Real time of start event in simple stopwatch mode |
| | | | 249=TIME_RESET_CBASE | Time of reset event on simple stopwatch when it counts down |
| | | | 250=Generic Lap | |
| | | | 254=AUX | |
| | | | 255=STOP | |
| | | | 245=SHOW JUMPING INFORMATION (only if a penalty is requested) | |
| Information | 1 | 48, 30h | 0=Time of day | |
| | | 49, 31h | 1=Run net time (split) | |
| | | 50, 32h | 2=Total net time (split) | |
| | | 51, 33h | 3=Lap net time | |
| | | 52, 34h | 4=Speed | |
| | | 53, 35h | 5=Time speed | |
| | | 54, 36h | 6=Air Temp. | |
| | | 55, 37h | 7=Snow Temp. | |
| | | 56, 38h | 8=Humidity | |
| | | 57, 39h | 9=Average speed (non radio) | |
| | | 84,54h | T=Average start-stop speed | |
| | | 65, 41h | A= A (non-finisher) | |
| | | 81, 51h | Q=SQ (disqualified) | |
| | | 80, 50h | P=NP (not started) | |
| | | 97, 61h | a=Annulled | |
| | | 83, 53h | S=Skipped not yet assigned | |
| | | 115, 73h | s=Skipped already assigned | |
| | | 75, 4Bh | K=Manually modified time event | |
| | | 71,47h | G=Effective time of phase is different from 0 only if the competition has two phases and phase 2 has been finished | |
| | | 72,48h | H=Total time tab. A | |
| | | 104,68h | h=Total time tab. C (without penalties) | |
| | | 73,49h | I=Penalties imposed tab. A | |
| | | 105,69h | i=Penalties imposed (seconds) tab. C | |
| | | 74,4Ah | J=Penalties for exceeding maximum time, tab. A | |
| | | 106,6Ah | j=Penalties in seconds for exceeding maximum time, tab. C | |
| | | 112, 70h | p=Total penalties, tab. A | |
| | | 107,6Bh | k=Final time tab C. (with penalties) | |
| | | 103,67h | g= Gundersen time (active only if it refers times to the first) | |

| | | | |
|---|---|---|---|
| | | 99, 63h | c= Time event substituted |
| | | 85, 55h | U= Duration of competition suspension in Show Jumping program |
| | | 87, 57h | W= Wind speed |
| | | 119, 77h | w= Wind direction |
| | | 88 ,58h | X= Brightness |
| | | 90 ,5Ah | Z= Net lap time (only for Basic Stopwatch) |
| Time/Speed | 10 | | Time in ten thousandths of a second 12345678980 corresponds to 12:34:56.7890<br>In the case of speed, the string means 123.456kmh<br>When the info field is equivalent to I,J,K,i,j the penalty (points or seconds) is transmitted in hundredths in the form #####.#### |
| Date | 8 | | Date in the following format 231201 corresponds to 23/12/2001<br>In the case of net time it is the number of days in the following format ±1234567<br>If a penalty comes up indicates if this is positive or negative |
| Dummy char | 2 | | Free bytes for future applications |
| **CR** | 1 | 13,0Dh | Carriage Return |
| **LF** | 1 | 10,0Ah | Line feed |

### 4.1.2. Reduced REI2 protocol

**`Total 33 bytes`**

| Description | N° bytes | ASCII Code (Dec, Hex) | Notes |
|---|---|---|---|
| **DC4** | 1 | 20, 14h | Protocol identifier |
| Device address | 1 | 32,20h (spac) | Reserved for future uses |
| Identifier of requesting device | 1 | | 0..9 A..z<br><br>If the output is enabled by REI2 the code is 20h ' ' |
| Competitor N° | 5 | | 00000<= N <=59999 |
| Information | 1 | 65, 41h | A=Run running time (split) |
| | | 66, 42h | B=Total running time (split) |
| | | 67, 43h | C=Lap running time |
| | | 68, 44h | D=Dynamic output running time |
| | | 80, 50h | P=Running penalties [2] |
| | | 69, 45h | E= Gundersen running time |
| | | 84, 54h | T= Running gap positive |
| | | 83, 53h | S= Running gap negative |
| | | 97, 61h | a=Run net time (split) |
| | | 98, 62h | b=Total net time (split) |
| | | 99, 63h | c=Lap net time |
| | | 100,64h | d=Dynamic output net time |
| | | 112, 70h | p=Penalties total |
| | | 101,65h | e= Gundersen net time |
| | | 116, 74h | t= Net gap positive |
| | | 115, 73h | s= Net gap negative |
| Time | 10 | | Net time in ten thousandths of a second is padded with zeros depending on the precision set<br>0034567800 corresponds to 00:34:56.7800 |
| Number of days | 1 | 43, 2Bh | "-" = negative number of days [3] |
| | | (48,30h) .. (57,39h) | 0..9 Number of days |
| | | 45, 2Dh | "+" = number of days of net time is greater than 9 [3] |
| | | 82, 52h | R= in PARALLEL or DUAL TIMER program for RED track |
| | | 66, 42h | B= in PARALLEL or DUAL TIMER program for BLUE track |
| Run/Trial | 3 | | 001<= Nm <=250[2] |
| Lap | 3 | | 000<= Nlap <=240<br>If the information does not refer to an intermediate, 000 is sent |
| Position | 3 | | Position of first 999 competitors |
| | | 48, 30h (3 times) | 000= Calculation of ranking disabled |
| | | 45, 2Dh (3 times) | "---" = the ranking is being recalculated |
| | | 43, 2Bh (3times) | "+++" = the position of the competitor is greater than 999 |
| Dummy char | 2 | | Free bytes for future applications |
| **CR** | 1 | 13,0Dh | Carriage Return |
| **LF** | 1 | 10,0Ah | Line feed |

[2] For show jumping, the run of phase 1 can range from 1 to 99, the run of the second phase from 100 to 198
[2] The penalty comes up at the same time as running time
[3] If a penalty comes up indicates if this is positive or negative

### 4.1.3. 'Static' reply

**`Total 52 bytes`**

| Description | N° bytes | ASCII Code (Dec, Hex) | Notes | | |
|---|---|---|---|---|---|
| **DC2** | 1 | 18,12h | Protocol identifier | | |
| Chronometer identifier | 1 | | R=REI2 | | |
| Device address | 1 | 32,20h (space) | Reserved for future uses | | |
| Program in use | 1 | | S=Single Starts<br>G=Group starts<br>B=Simple stopwatch<br>P=Parallel<br>I=Show jumping<br>N=Swimming<br>T=Track Chase<br>O=Pc OnLine | | |
| Mode | 1 | | O=OnLine<br>F=OffLine | | |
| Status reply | 1 | | **R**: the record transmitted refers to the n _th request<br>**E**: the record transmitted is the last for the n _th request<br>**Z**: response not available for the n _th request | | |
| Identifier of requesting device | 1 | | 0..9 A..z | | |
| Identifier reply | 5 | | 00000<= Nresponse <=99999 | Progressive number which identifies the PC request, or progressive number for information sent autonomously in on-line and off-line modes. | |
| Competitor N° | 5 | | 00000<= N <=59999 | | |
| Group/Category | 3 | | 000<= Ng <=199 | If the Group/Category is equal to zero or the Groups/Categories have not been defined or the information filtered by group is not being downloaded (it is not always downloaded because the competitor could belong to more than one group) | |
| Run/Trial | 3 | | 001<= Nm <=250 | | |
| Physical channel[3] | 3 | | 000<= Physical channel <=255 | Channel | Channel Description |
| | | | Main Lines | 000 | Start Line |

[3] Note: for the simple stopwatch and the parallel, the output has a different meaning:

- SIMPLE STOPWATCH: The output data item is the number of events of the race set which have been memorised. If the event is a stop, it is the lap number
- PARALLEL: The output data item is the progressive number of direct contests

| | | |
|---|---|---|
| | 001 | Lap Line |
| | 015 | Stop Line |
| | 016 | Aux Line |
| | 100 | Start Key |
| | 101 | Lap Key |
| | 115 | Stop Key |
| | 116 | Aux Key |
| | 200 | Auto Start |
| | 300 | Keyb Start |
| | 301 | Keyb Lap |
| | 315 | Keyb Stop |
| Pod Inputs | 400..407 | POD 0 |
| | 410..417 | POD 1 |
| | 420..427 | POD 2 |
| | 430..437 | POD 3 |
| | 440..447 | POD 4 |
| | 450..457 | POD 5 |
| | 460..467 | POD 6 |
| | 470..477 | POD 7 |
| | 480..487 | POD 8 |
| | 490..497 | POD 9 |
| Via Radio Inputs | 500 | RADIO START |
| | 501 | RADIO LAP 1 |
| | 502 | RADIO LAP 2 |
| | 503 | RADIO LAP 3 |
| | 504 | RADIO LAP 4 |
| | 505 | RADIO LAP 5 |
| | 506 | RADIO LAP 6 |
| | 507 | RADIO LAP 7 |
| | 508 | RADIO LAP 8 |
| | 509 | RADIO LAP 9 |
| | 510 | RADIO LAP A |
| | 511 | RADIO LAP B |
| | 512 | RADIO LAP C |
| | 513 | RADIO LAP D |
| | 514 | RADIO LAP E |
| | 515 | RADIO STOP |
| Retrieved by Encoder | 600 | ENC START |
| | 601 | ENC LAP 1 |
| | 602 | ENC LAP 2 |
| | 603 | ENC LAP 3 |
| | 604 | ENC LAP 4 |
| | 605 | ENC LAP 5 |
| | 606 | ENC LAP 6 |
| | 607 | ENC LAP 7 |
| | 608 | ENC LAP 8 |
| | 609 | ENC LAP 9 |
| | 610 | ENC LAP A |
| | 611 | ENC LAP B |
| | 612 | ENC LAP C |
| | 613 | ENC LAP D |
| | 614 | ENC LAP E |
| | 615 | ENC STOP |
| Imputed by RadioModem | 700 | |
| Serial A RADIO Inputs | 516 | RADIO START |
| | 517 | RADIO LAP 1 |
| | 518 | RADIO LAP 2 |

| | | | | |
|---|---|---|---|---|
| | | | 519 | RADIO LAP 3 |
| | | | 520 | RADIO LAP 4 |
| | | | 521 | RADIO LAP 5 |
| | | | 522 | RADIO LAP 6 |
| | | | 523 | RADIO LAP 7 |
| | | | 524 | RADIO LAP 8 |
| | | | 525 | RADIO LAP 9 |
| | | | 526 | RADIO LAP A |
| | | | 527 | RADIO LAP B |
| | | | 528 | RADIO LAP C |
| | | | 529 | RADIO LAP D |
| | | | 530 | RADIO LAP E |
| | | | 531 | RADIO STOP |
| | | Serial B RADIO Inputs | 532 | RADIO START |
| | | | 533 | RADIO LAP 1 |
| | | | 534 | RADIO LAP 2 |
| | | | 535 | RADIO LAP 3 |
| | | | 536 | RADIO LAP 4 |
| | | | 537 | RADIO LAP 5 |
| | | | 538 | RADIO LAP 6 |
| | | | 539 | RADIO LAP 7 |
| | | | 540 | RADIO LAP 8 |
| | | | 541 | RADIO LAP 9 |
| | | | 542 | RADIO LAP A |
| | | | 543 | RADIO LAP B |
| | | | 544 | RADIO LAP C |
| | | | 545 | RADIO LAP D |
| | | | 546 | RADIO LAP E |
| | | | 547 | RADIO STOP |
| | | Imputed by PC | 900 | PC |
| Logical Channel | 3 | 000<= Logical channel <=255 | | |
| | | 000=START | | |
| | | 001..240= LAP n | | The SPLITs with LAP have a range from 001 to 200 (corresponding to the number of times the stop button is pressed)<br>SPLITs without LAP have a range from 201 to 240. If the maximum number is exceeded, the previous data item is cancelled<br>The finishes following the first are considered as LAP. |
| | | 248=REAL_START_CBASE | | Real time of start event in simple stopwatch mode |
| | | 249=TIME_RESET_CBASE | | Time of reset event on simple stopwatch when it counts down |

| | | | | |
|---|---|---|---|---|
| | | | 250=Generic Lap | |
| | | | 254= AUX | |
| | | | 255= STOP | Time of first competitor at finish |
| | | | 245= SHOW JUMPING INFORMATION (only if a penalty is requested) | |
| Information | 1 | 48, 30h | 0=Time of day | |
| | | 49, 31h | 1=Run net time (split) | |
| | | 50, 32h | 2=Total net time (split) | |
| | | 51, 33h | 3=Lap net time | |
| | | 52, 34h | 4=Speed | |
| | | 53, 35h | 5=Time speed | |
| | | 54, 36h | 6=Air Temp. | |
| | | 55, 37h | 7=Snow Temp. | |
| | | 56, 38h | 8=Humidity | |
| | | 57, 39h | 9=Average speed (non radio) | |
| | | 65, 41h | A=NA (non-finisher) | |
| | | 81, 51h | Q=SQ (disqualified) | |
| | | 80, 50h | P=NP (not started) | |
| | | 97, 61h | a=Time event deletion (only on-line) | |
| | | 110, 6Eh | n=Deletion of a previous non-finisher | |
| | | 113, 71h | q=Deletion of a previous disqualified athlete | |
| | | 112, 70h | p=Deletion of a previous non-starter | |
| | | 83, 53h | S=Skipped not yet assigned | |
| | | 115, 73h | s=Skipped already assigned | |
| | | 75, 4Bh | K=Manually modified time event | |
| | | 82,52h | R=Present position | |
| | | 84,54h | T=Start-stop | |
| | | 71,47h | G=Effective time of phase is different from 0 only if the competition has two phases and phase 2 has been finished | |
| | | 72,48h | H=otal time tab. A | |
| | | 104,68h | h=Total time tab. C (without penalties) | |
| | | 73,49h | I=Penalties imposed tab. A | |
| | | 105,69h | i=Penalties imposed (seconds) tab. C | |
| | | 74,4Ah | J=Penalties for exceeding maximum time, tab. A | |
| | | 106,6Ah | j=Penalties in seconds for exceeding maximum time, tab. C | |
| | | 112, 70h | p=Total penalties, tab. A | |
| | | 107,6Bh | k=Final time tab C. (with penalties) | |
| | | 103,67h | g= Gundersen time (active only if it refers times to the first) | |
| | | 99, 63h | c= Time event substituted | |
| | | 85, 55h | U= Duration of competition suspension in Show Jumping program | |
| | | 87, 57h | W= Wind speed | |
| | | 119, 77h | w= Wind direction | |
| | | 88 ,58h | X= Brightness | |
| Time/Speed | 10 | | Time in ten thousandths of a second 12345678980 corresponds to 12:34:56.7890<br>In the case of speed, the string means 123.456kmh | |

| | | | |
|---|---|---|---|
| Date | 8 | | Date in the following format 231201 corresponds to 23/12/2001<br>In the case of net time it is the number of days in the following format ±1234567<br>If a penalty comes up, indicates if this is positive or negative |
| Dummy char | 2 | | Free bytes for future applications |
| **CR** | 1 | 13,0Dh | Carriage Return |
| **LF** | 1 | 10,0Ah | Line feed |

### 4.1.4. Error reply

**Total bytes: 10**

| Description | N° bytes | ASCII Code (Dec, Hex) | Notes |
|---|---|---|---|
| **ETB** | 1 | 23,17h | Protocol identifier |
| Chronometer identifier | 1 | | R=REI2 |
| Device address | 1 | 32,20h (space) | Reserved for future uses |
| Identifier of requesting device | 1 | | 0..9 A..z |
| Identifier requested | 3 | | 000<= Nrequest <=999<br>Progressive number which identifies the PC request<br>The reply is indicated by the same number<br>If an error has occurred before reception of the request code, 000 is shown |
| Type of error found | 1 | 48,30h | 0=request identifier |
| | | 49,31h | 1=type of information |
| | | 50,32h | 2=competitor number |
| | | 51,33h | 3=logical channel |
| | | 52,34h | 4=run |
| | | 53,35h | 5=group |
| | | 54,36h | 6=time |
| | | 55,37h | 7=date |
| | | 56,38h | 8=periodicity |
| | | 57,39h | 9=serial output |
| | | 66, 42h | B=periodicity |
| | | 67, 43h | C=status code |
| | | 68, 44h | D=identifier of requesting device |
| | | 69, 45h | E=identifier chronometer |
| | | 70, 46h | F=time sign |
| | | 71, 47h | G=device address |
| | | 72, 48h | H=dynamic request error A |
| | | 73, 49h | I=dynamic request error B |
| | | 74, 4Ah | J=competitor number reference for dynamic stop |
| | | 75, 4Bh | K=logical channel reference for dynamic stop |
| | | 76, 4Ch | L=run reference for dynamic stop |
| | | 77, 4Dh | M= start lists |
| **CR** | 1 | 13,0Dh | Carriage Return |
| **LF** | 1 | 10,0Ah | Line feed |

### 4.1.5. REI2 status reply

**Total bytes: 20**

| Description | N° bytes | ASCII Code (Dec, Hex) | Notes |
|---|---|---|---|
| **CAN** | 1 | 24,18h | Protocol identifier |
| Chronometer identifier | 1 | | R = REI2 |
| Device address | 1 | 32,20h (space) | Reserved for future uses |
| Identifier of requesting device | 1 | | 0..9 A..z |
| Identifier requested | 4 | | 0001<= Nrequest <=0999<br>Progressive number which identifies the PC request<br>If the first byte is equal to E, it identifies the end of the information requested. For example, E123 identifies that the replies sought by request 123 have finished |
| Information requested | 10 | | Contains case by case the value of the data requested |
| **CR** | 1 | 13,0Dh | Carriage Return |
| **LF** | 1 | 10,0Ah | Line feed |

### 4.1.6. REI2 status reply codes

**Request 0000= net times (totals runs,lap)**

Byte 0:

0= net times totals
1= net times runs
2= net times lap

Bytes 1..9 not used

**Request 1000=Precision set**

Byte 0:

0= 1s
1= 0.1s
2= 0.01s
3= 0.001s
4= 0.0001s

Bytes 1..9 not used

**Request 2000=Status of main lines**

0= Open (if in N/O configuration) Closed (if in N/C configuration)
1= Closed (if in N/O configuration) Open (if in N/C configuration)

Byte 0: START line status
Byte 1: LAP line status
Byte 2: STOP line status
Byte 3: AUX line status

**Request 3000=Status of pod lines**

Byte 0: Pod line

0= Open (if in N/O configuration) Closed (if in N/C configuration)
1= Closed (if in N/O configuration) Open (if in N/C configuration)

Byte 1: Status of line 0
Byte 2: Status of line 1
Byte 3: Status of line 2
Byte 4: Status of line 3
Byte 5: Status of line 4
Byte 6: Status of line 5
Byte 7: Status of line 6
Byte 8: Status of line 7
Byte 9: Not used

**Request 4000=Runs excluded from calculation of total time**

Bytes 0..2 Number of run excluded from calculation of total time
Bytes 3..9 not used

**Request 5xxx= Logical channel disactivation times xxx**

Bytes 0..2 logical channel of reference
Bytes 3..7 Disactivation time in thousandths of a second (12345 are 12.345 s)
Bytes 8..9 not used

**Request 6000=Main line N/O N/C configuration**

0= N/O
1= N/C

Byte 0: START line configuration
Byte 1: LAP line configuration
Byte 2: STOP line configuration
Byte 3: AUX line configuration

**Request 7000=Dynamic outputs status**

| | | |
|---|---|---|
| Byte 0: | Dynamic output 1 | 0=not active<br>1=active |
| Byte 1: | Dynamic output 2 | 0=not active<br>1=active |
| Byte 4: | Serials being used by dynamic output 1 | A= Serial PCA B=Serial PCB T=both |
| Byte 5: | Serials being used by dynamic output 2 | A= Serial PCA B=Serial PCB T=both |

**Request 9999=Basic Configuration of device**

Byte 0: Type of device (R=Rei2)
Byte 1: Address
Byte 2: Program set

| | |
|---|---|
| SINGLE STARTS | 0 |
| GROUP STARTS | 1 |
| SIMPLE_STOPWATCH | 2 |
| PARALLEL | 3 |

| | |
|---|---|
| SHOW JUMPING | 4 |
| SWIMMING | 5 |
| TRACK CHASE | 6 |
| PC_ONLINE | 7 |
| None | 9 |

Byte 3: Program configuration
To be defined

Byte 4: Number of devices connected to REI2NET

Bytes 5..8 Serial number

Byte 9 not used

## *4.2. Data transmission from PC to REI2*

### 4.2.1. 'Static' request

**`Total 24 bytes`**

| Description | N° bytes | ASCII Code (Dec, Hex) | Notes |
|---|---|---|---|
| **DC1** | 1 | 17,11h | Protocol identifier |
| Chronometer identifier | 1 | | R = REI2 |
| Device address | 1 | 32,20h (space) | Reserved for future uses |
| Identifier of requesting device | 1 | | 0..9 A..z |
| Identifier requested | 3 | | 000<= Nrequest <=999<br>Progressive number which identifies the PC request<br>The reply is marked by the same number |
| Competitor N° | 5 | | 00000<= N <=59999<br>00000= Sends the type of information wanted for all the numbers which satisfy the request<br>In the case of a group start, if you are searching for a start you must specify the group number |
| Information | 1 | 48, 30h | 0=Time of day |
| | | 49, 31h | 1=Run net time (split) |
| | | 50, 32h | 2=Total net time (split) |
| | | 51, 33h | 3=Lap net time |
| | | 52, 34h | 4=Speed |
| | | 53, 35h | 5=Time speed |
| | | 54, 36h | 6=Air Temp. |
| | | 55, 37h | 7=Snow Temp. |
| | | 56, 38h | 8=Humidity |
| | | 57, 39h | 9=Average speed (non radio) |
| | | 65, 41h | A=NA (non-finisher) |
| | | 81, 51h | Q=SQ (disqualified) |
| | | 80, 50h | P=NP (not started) |
| | | 97, 61h | a=Annulled |
| | | 83, 53h | S=Skipped not yet assigned |
| | | 115, 73h | s=Skipped already assigned |
| | | 84,54h | T=Average start-stop speed |
| | | 75, 4Bh | K=Manually modified time event |
| | | 76,4Ch | L=Last lap of competitor set |
| | | 116,74h | t=All laps of competitor set |
| | | 82,52h | R=Present position |
| | | 42,2Ah | * =All time events, including NA,SQ,NP, skipped |
| | | 71,47h | G=Effective time of phase is different from 0 only if the competition has two phases and phase 2 has been finished |
| | | 72,48h | H=Total time tab. A |
| | | 104,68h | h=Total time tab. C (without penalties) |
| | | 73,49h | I=Penalties imposed tab. A |
| | | 105,69h | i=Penalties imposed (seconds) tab. C |

| | | | |
|---|---|---|---|
| | | 74,4Ah | J=Penalties for exceeding maximum time, tab. A |
| | | 106,6Ah | j=Penalties in seconds for exceeding maximum time, tab. C |
| | | 112, 70h | p=Total penalties, tab. A |
| | | 107,6Bh | k=Final time tab C. (with penalties) |
| | | 103,67h | g= Gundersen time (active only if it refers times to the first) |
| | | 87, 57h | W= Wind speed |
| | | 119, 77h | w= Wind direction |
| | | 88, 58h | X= Brightness |
| | | 108, 6Ch | l= All time events not yet sent |
| | | 98, 62h | b= All time events not yet sent (waiting for acknowledge) |
| | | 99, 63h | c= All net run times not yet sent (waiting for acknowledge) |
| | | 100, 64h | d= All total net times not yet sent (waiting for acknowledge) |
| | | 113, 71h | q= acknowledge of last static request with type of information ‘’b’, ‘c’, or ‘d’ |
| Logical channel | 3 | | 000<= Logical channel <=255<br>000=START<br>001..240= LAP n<br>248=REAL_START_CBASE[4]<br>249=TIME_RESET_CBASE[5]<br>251=All events<br>255= STOP |
| Run | 3 | | 0<= Nm <=250[6]<br>0= all runs |
| Group | 3 | | 0<= Ng <=199<br>If the Group/Category is equal to zero, this means all the groups |
| Output | 1 | | Serial port on which response is sent<br>S=same serial port as request<br>A=serial port A<br>B=serial port B<br>T=both ports |
| **CR** | 1 | 13,0Dh | Carriage Return |

[4] Real time of start event in basic stopwatch mode
[5] Time of reset event in basic stopwatch when it counts down
[6] For show jumping the phase 1 run can range from 1 to 99, the second phase run from 100 to 198

### 4.2.2. 'Dynamic' request

**`Total 46 bytes`**

| Description | N° bytes | ASCII Code (Dec, Hex) | Notes |
|---|---|---|---|
| **DC1** | 1 | 19,13h | Protocol identifier |
| Chronometer identifier* | 1 | | R = REI2 |
| Device address* | 1 | 32,20h (space) | Reserved for future uses |
| Identifier of requesting device | 1 | | 0..9 A..z |
| Identifier requested* | 1 | | A= activation dynamic output 1/ tick A<br>B= activation dynamic output 2 / tick B<br>a= disactivation dynamic output 1/ tick A<br>b= disactivation dynamic output 2 / tick B<br>T= activation of data output of competitor specified on displayboard[7]<br>t= disactivation of data output of competitor specified on displayboard |
| Competitor N°* | 5 | | 1<= N <=59999<br>0= generic time request: Tev=0 Taux=0<br>60000=tick activation request |
| Logical channel | 3 | | Logical channel of reference for Tev time<br>0=START<br>1..240= LAP n<br>250=Generic Lap<br>254= AUX<br>255= STOP |
| Run | 3 | | 0<= Nm <=250[8]<br>0=present run |
| Competitor N° of Stop ref. | 5 | | 1<= N <=59999<br>60000= time reference disactivation |
| Logical channel of Stop ref. | 3 | | Logical channel of stop reference<br>0=START<br>1..240= LAP n<br>250=Generic Lap<br>254= AUX<br>255= STOP |
| Run of Stop ref. | 3 | | 0<= Nm <=250[9]<br>0=present run |
| Sign | 1 | | Taux time sign (0=positive, 1=negative) |
| Time | 10 | | Taux time in ten thousandths of a second padded with zeros depending on the precision set<br>0034567800 corresponds to 00:34:56.7800 |
| Date | 1 | (48,30h) .. (57,39h) | 0..9 Number of days |
| Periodicity* | 5 | | Period in hundredths of a second<br>12345 corresponds to 123.45 seconds |

[7] The competitor is shown on the displayboard until the function is disabled,regardless of operations made on REI2.
[8] For show jumping the phase 1 run can range from 1 to 99, the phase 2 run from 100 to 198
[9] For show jumping the phase 1 run can range from 1 to 99, the phase 2 run from 100 to 198

| | | | |
|---|---|---|---|
| Output* | 1 | | Serial port on which response is set<br>S=same serial port as request<br>A= serial port A<br>B= serial port B<br>T= both ports |
| **CR*** | 1 | 13,0Dh | Carriage Return |

### 4.2.3. Request for break, suspension and resumption of transmission, record repetition

**Total bytes: 9**

| | | | | |
|---|---|---|---|---|
| **NAK** | 1 | 21,15h | Protocol identifier | |
| Chronometer identifier | 1 | | R = REI2 | Break request mode of use: |
| Device address | 1 | 32,20h (space) | Reserved for future uses | If the break request is sent before the relative static request -> nothing happens |
| Identifier of requesting device | 1 | | 0..9 A..z | If the static request has finished sending the data -> the break request has no effect |
| Operation specifier | 1 | | C: interrupts the reply to the PC request xxx | If the break request is sent after the relative static request -> the static request is immediately blocked |
| Identifier requested | 3 | | 001<= Nrequest <=999 | |
| **CR** | 1 | 13,0Dh | Carriage Return | |

### 4.2.4. Status request

**Total bytes: 13**

| Description | N° bytes | ASCII Code (Dec, Hex) | Notes |
|---|---|---|---|
| **SYN** | 1 | 22,16h | Protocol identifier |
| Chronometer identifier | 1 | | R = REI2 |
| Device address | 1 | 32,20h (space) | Reserved for future uses |
| Identifier of requesting device | 1 | | 0..9 A..z |
| Identifier requested | 3 | | 001<= Nrequest <=999<br>The reply is indicated by the same number |
| Information requested | 10 | | Contains case by case the value of the data requested |
| Code of status requested | 4 | | 0000= net times (totals, runs,lap)<br>1000=Precision set<br>2000=Status of main lines<br>3000=Pod lines status<br>4000=Runs excluded from calculation of total time<br>5xxx= Logical channel disactivation times xxx<br>6000= N/O N/C configuration of main lines<br>9999=Basic device information |
| Output | 1 | | Serial port on which response is set<br>S=same serial port as request<br>A= serial port A<br>B= serial port B<br>T= both ports |
| **CR** | 1 | 13,0Dh | Carriage Return |

## *4.3. Event transmission from PC to REI2*

### 4.3.1. Time insertion

**`Total bytes 34`**

| Description | N° bytes | ASCII Code (Dec, Hex) | Notes |
|---|---|---|---|
| **DC4** | 1 | 23, 17h | Protocol identifier |
| Chronometer identifier | 1 | | R = REI2 |
| Device address | 1 | 32,20h (space) | Reserved for future uses |
| Information | 1 | 48,30h | 0= Chronological time |
| | | 65,41h | A = NA (not finished) |
| | | 80,50h | P = NP (not started) |
| | | 97,61h | a = Annulled |
| Competitor N° | 5 | | 00001<= N <=59999 |
| Logical channel | 3 | | 000<= Logical channel <=255<br>000=START<br>001..240= LAP n<br>255= STOP |
| Physical channel | | | 900 PC |
| Run | 3 | | 0<= Nm <=250[10] |
| Time | 10 | | Time in ten thousandths of a second padded with zeros according to the precision set<br>0034567800 corresponds to 00:34:56.7800 |
| Date | 8 | | Date in the following format 231201 corresponds to 23/12/2001<br>In the case of net time it is the number of days in the following format ±1234567<br>If a penalty comes up, indicates if this is positive or negative |
| **CR** | 1 | 13,0Dh | Carriage Return |

[10] For show jumping the phase 1 run can range from 1 to 99, the phase 2 run from 100 to 198

## *4.4. Printout transmission from PC to REI2*

For sending strings to the REI2 printer.

| **Description** | **N° byte** | **Codice ASCII (Dec, Hex)** | **Notes** |
|---|---|---|---|
| **STR** | 1 | 25, 19h | Protocol identifier |
| | … | | Text |
| **CR** | 1 | 13,0Dh | Carriage Return |
| **LF** | 1 | 10,0Ah | Line feed |

## 5. Modification history

The following table summarises the main modifications made to the present document.

| Program version | Chapter | Page | Description of intervention |
|---|---|---|---|
| 1.03 | | | Specific protocol updates, change in order of chapters. |
| 1.07 | 1.4 | 5 | New function Insertion of time events |
| 1.07 | 2 | 6 | Inserted in chap. General information about the REI2- PC protocol indications for displayboard piloting<br>. |
| 1.07 | 4.3 | 29 | Insertion of specifications for Event transmission from PC to REI2 |
| 1.07.9 | 1.4 | 5 | Running time explanation added |

## Copyright



Collaborators in the creation of REI2 and the preparation of the relative manuals are:
**Ing. Roberto Biasi,**
**Dr. Vinicio Biasi**
**Ing. Federico Gori**
**Ing. Alessandro Miorelli**
**Giuliano Menestrina**
**Daniele Veronese**

The software and manuals are available in the following languages: Italian, English, German and French.

**Microgate S.r.L**
Via Stradivari, 4 Stradivaristr.
39100 BOLZANO - BOZEN
ITALY

Tel. +39 471 501532 - Fax +39 471 501524
e-mail info@microgate.it