

**README.DOC from Mar/4/2004**  
**Last minute information in addition to manuals**

## **1 ERTS V3.37**

- 1.1 Multiple Session definitions
- 1.2 Subject groups
- 1.3 Choosing blank character for screen echo
- 1.4 Limiting key coding to registration period
- 1.5 Managing bonus points
- 1.6 Displaying bonus points
- 1.7 Displaying time of block
- 1.8 User defined token delimiter for single word presentation
- 1.9 Extended CS-command for equi-distant random time interval
- 1.10 Defining Sets of Sets
- 1.11 Extended RK-command
- 1.12 Removing parallel element
- 1.13 Start message and warning tone for unstable tracking
- 1.14 Selective feedback for dual task situations
- 1.15 Immediate start of dual task
- 1.16 Periodic syncing with external counter
- 1.17 Writing all results at end of session

## **2 Analog joystick response in ERTS**

- 2.1 MM (Measure Movement)
- 2.2 MJ (Measure Joystick)
- 2.3 WJ (Wait for Joystick Position)
- 2.4 Additional columns in result file
- 2.5 Using Joystick keys
- 2.6 Calibration of joystick coordinates
- 2.7 Test utility TESTJOY

## **3 ERTSCODE**

- 3.1 NUMERIC keyword for declaring numeric variables
- 3.2 IMPLICIT keyword
- 3.3 New Start\_ and \_Final compute blocks
- 3.4 SHOWLIST
- 3.5 EXCLUDE
- 3.6 Extended /E (Export) command line option

## **4 Incompatibilities to older ERTS versions**

# **1 ERTS V3.37**

## **1.1 Multiple Session definitions**

You can now define different session structures for the same experiment by creating multiple session definitions within the same ERTS script.

Syntax: SESSION [Name]

Name    Name of Session which is used also as short description within the results file

In order to be compatible to previous ERTS releases, the name is optional but must be defined if multiple sessions are defined. Only the token up to the next delimiter (blank, comma, ..) will be used as the name.

If multiple sessions are defined, you must either assign session definitions to subject groups (see new GROUP command) or ERTS will randomly select one of the session definitions when you launch an experimental session. The name of the selected session is displayed before starting the experimental session and the session number is included in the result file.

## 1.2 Subject groups

In order to automate between-subject designs, the new GROUP command supports the definition of subject groups.

Syntax: GROUP Name  
SessName SubjNo1 [SubjNo2 [.. SubjNoN]]

Name	Name of the group. This will be listed in the result file
SessName	Name of the session definition which should be launched for this group
SubjNo	Subject numbers that belong to this group

A group is defined by simply enumerating the subject numbers that go into this group. The group is then assigned to a session definition and depending on the subject number, the corresponding session definition is selected. ERTS will display the selected group after you have typed in the subject number

Note that the GROUP command must be placed behind the session definitions because the command is referring to these definitions.

## 1.3 Choosing blank character for screen echo

In the SE trial command, you can now specify any symbol in addition to the underscore as the blank character

Syntax: SE Position [PosNo [Font [Case [FieldChar ]]]]

FieldChar	0 for blank
	1 for underscore
	> 1 decimal value of the ASCII character to be used as fieldchar

## 1.4 Limiting key coding to registration period

The CODEKEYS (only ERP-Version) introduces an additional parameter for limiting key coding (sending sync signals upon key presses) to periods of active response registration (RK-command).

Syntax: CODEKEY [Channel Keys ByteSequ [ToneSet [RegOnly]]]

RegOnly	0 code any key action during session
	1 code only key action within active response registration periods

## 1.5 Managing bonus points

For counting and displaying bonus points, ERTS has a special internal numerical bonus counter. This counter can be initialized or increased/decreased using the BO-trial command

Syntax: BO Operation Amount [FbWidth Precision]

Operation	0 Initialize internal counter with amount 1 Add amount to current value 2 Subtract amount from current value 3 Add Rt dependent value (value = Rt / value)
Amount	Numerical positive value. If precision is larger than 0 then the n right digits will be interpreted as decimal places (e.g. if precision=2, then 1050 will be interpreted as 10,50)
FBWidth	0: Do not store bonus value in FB colum n > 0: Store bonus value in FB column and set with of FB column to n characters.
Precision	0: Interpret value as defined n > 0: Interpret the n right numbers as decimal places

The default width of FB columns is 3 characters. If your bonus points are greater or equal to 1000 than you need to set width of 4 or higher.

## 1.6 Displaying bonus points

Bonus points are displayed using special placeholders in a formatted picture definition (see PICTURE command with subtype F)

@BBBB	Display last amount
@bbbb	Display current points

## 1.7 Displaying time of block

Additional place holders in formatted pictures support the display of the elapsed or residual block execution time.

@tttt	Display elapsed block execution time in seconds
@TTTT	Display remaining block execution time in seconds (MAXTIME must be active)

## 1.8 User defined token delimiter for single word presentation

The following session command defines a token delimiter other than blank

SETSWDELIM	AsciiVal
AsciiVal	Numerical Value which is the decimal equivalent of the character that should be used as the delimiter

## 1.9 Extended CS-command for equi-distant random time interval

In addition to the exponential random time interval, the CS-command now supports the definition of an equi-distant random time interval. Whenever MaxRnd is less or equal to RndTime, ERTS will interpret these values as follows:

RndTime Time range that should be varied at random

MaxRnd Divisor for creating equi-distant time intervals within this range

For example, if MaxRnd=4 and RndTime=4000 then five different time values are generated at random: 0,1000,2000,3000,4000.

## 1.10 Defining Sets of Sets

The SET command now supports the definition of sets that exists of sets. The new sub-type for this new type of element is 'E'.

## 1.11 Extended RK-command

The RK-command has been extended by two arguments

Syntax: RK [Key [MaxRt [N(Keys) [Skey [EndKey [Echo [Reset AutoRep [DbIKey]]]]]]]]

AutoRep Auto repetition mode for PC-keyboard and EXKEY. For PC-keyboard the autorep rate is determined by BIOS setting. For external keys (EXKEY), the value sets the start delay in msec. If a key is pressed longer than AutoRep msec, ERTS will start to auto-repeat that key press with increasing rate. (default is 0, no auto rep).

DbIKey If you register multiple responses (N(Keys) > 1) and this flag is set to zero, each key can only be used once. (default 1)

## 1.12 Removing parallel element

If no element is specified in the RE-command, then the element with the same index that has been used by the most recent DE-command will be removed from the set. This is similar to parallel drawing of elements, but instead the parallel element will just be removed.

## 1.13 Start message and warning tone for unstable tracking

The RUNUNSTABLE command has been extended to support the display of a start message at the beginning of the tracking task and the presentation of a warning tone prior to hitting the borders.

Syntax: RUNUNSTABLE T(WarmUp) T(Record) [Lambda [Gain [Range [Key [Compress [Tone] [StartPic] [StartPos] [StartFont] [WarnRange] ]]]]]]]]

StartPic The name of the picture definition that should be displayed as the starting message at the beginning of the tracking task

StartPos The name of the position definition on which the start message should be displayed

StartFont The name of a font definition that should be used to render the message

WarnRange The "quiet" tracking range [mm/10]. If this range is exceeded by the vertical bar a warning tone will be presented as defined in the 'Tone' argument.

## 1.14 Selective feedback for dual task situations

The OK-command has been extended to support selected evaluations of responses with respect to time and key order.

Syntax: OK CompMode KeyIndex LowTime HighTime LowKey HighKey

Low/High Time	Limits the evaluation to responses that are greater than LowTime and lower than HighTime. All other responses will be ignored
Low/High Key	Limits the evaluation of responses to key presses that are inbetween LowKey and HighKey. All other responses will be ignored.

When cutting the evaluation range by defining a LowKey ERTS interpret all following latency relative to LowKey. This feature allows you to look at latencies relative to another keystroke. If both a key and a time range is defined, ERTS will first determine the key range and then will determine the time range relative to LowKey.

Typically, LowKey will refer to a key that is generated by the GR trial command to time stamp the onset of a stimulus within the response interval. This way the OK command will only interpret responses that are registered after the onset of that stimulus and will interpret latencies relative to the onset

Unlike FB you can use OK multiple times with different parameters. This enables you to evaluate all four error conditions one after the other. With +OK you could even display all messages at once if you use different screen positions.

## 1.15 Immediate start of dual task

The RUNMSEARCH and SETMSEARCH command has been extended by the AutoStart argument to support the immediate start of the task without displaying a target set and without waiting for further confirmation by the subject. This is useful if the task must be synchronized with EEG/MRI devices and when this command is used to display plain lists instead of a memory task (see ListMode argument).

Syntax: RUN/SETMSEARCH N(Valid) N(WarmUp) TargSet ProbeSet TargPos ProbePos StartKey ò  
YesNoKey MaxRt Pause [Sequence [Port [ParSet [ForceEndOfTracking [ListMode]  
[AutoStart]]]]]

AutoStart	0: Task is started by key press (default)
	1: Task is started automatically

## 1.16 Periodic syncing with external counter

The WC trial command (Wait for external Counter) has been extended to support a periodic syncing on every i-th trial

Syntax: WC [ [NTics [ResetCounter [Cycle ]]]

Cycle	Integer value that defines the number of repetitions in which the WC command should be executed. For example, a value of '2' means that every second time the WC command will wait for an external counter value.
-------	---

Use this new argument if you want to run several trials in-between two external MRI-signals but still want to re-calibrate with each new signal.

## 1.17 Writing all results at end of session

By default, ERTS is storing results after each block. If your block transitions are time critical, use the new /J command line option to hold all results in memory until the end of the session. This will, of course, require more working memory than the block-wise writing of results.

This new option does not work in combination with tracking tasks and staircases.

## 2 Analog joystick response in ERTS

ERTS provides some support for analog joystick responses that are based on movement. The registration of the analog joystick must follow different principles because the game port does not produce interrupts. ERTS must poll the game port in order to detect any changes.

Because of this, the time resolution depends on CPU speed and the response characteristic of the game port. You can test this speed by checking the port with the TESTJOY utility. In average, response time should be within a 5ms range.

Unlike with the RK command there exist some same restrictions for MM and MJ. Only the next trial command is executed and then ERTS waits for a joystick response. There is no continuation and no internal pre-loading of images before the response is made or MaxRT is expired.

There are two ways to think about joystick movements. One is in terms of the angle the joystick is moved away from the center position. The second is in terms of screen coordinates where the joystick cursor is located.

For recording joystick movements as a response, the following trial commands have been implemented:

MM	Record the leaving of a center position and crossing of circular boundaries
MJ	Record the leaving of a starting zone and arrival on target zone

Please note that all features described in this chapter are provided on an “as is” base

### 2.1 MM (Measure Movement)

The MM command measures the time until the joystick is moved across user-defined boundaries. Key names are assigned to each boundaries in order to code the event in the result file.

Syntax: MM Radius Angle Tolerance Keys [MaxRt [Cursor [ResultMode [Scale]]]]

Radius	Numeric set definition enumerating the boundaries which trigger a response when crossed (from center to outside). A negative value means that any crossing triggers a response. A positive value triggers a response only if the crossing is within a valid sector (see angle and tolerance)				
Angle	Numeric set definition enumerating the angles that define valid sectors				
Tolerance	Tolerance defines the size of the valid sectors angle plus/minus tolerance).				
Keys	A KEY definition that is used as a translation table for the angles. List of key names within KEY definition and list of angle in numeric key set must correspond.				
MaxRt	Maximum response time				
Cursor	Name of a SET definition consisting of at least on picture which will be used as the joystick cursor Valid picture types are P, G, R.				
ResultFormat	Determines the format of the result file <table> <tr> <td>0</td> <td>X/Y-Coordinates (default)</td> </tr> <tr> <td>1</td> <td>Radius and Angle</td> </tr> </table>	0	X/Y-Coordinates (default)	1	Radius and Angle
0	X/Y-Coordinates (default)				
1	Radius and Angle				
Scale	Determine metric of results for radius <table> <tr> <td>0</td> <td>Screen coordinates</td> </tr> <tr> <td>1</td> <td>Joystick angle (see STICKANGLE)</td> </tr> </table>	0	Screen coordinates	1	Joystick angle (see STICKANGLE)
0	Screen coordinates				
1	Joystick angle (see STICKANGLE)				

### 2.2 MJ (Measure Joystick)

This command measure the time of leaving a starting zone and the arrival of one or more target zones. The target zones are mapped to key names.

Syntax: MJ StrtRadius TargPos PosNo PosRadius Keys [MaxRt [Cursor [Scale]]]

StrtRadius	Radius which triggers first response
TargPos	Position definitions that define valid target zones

PosNo	Index of position (within the position definition) which is used as target zone. If zero, all positions of TargPos are valid.
PosRadius	Defines the size of the valid area around each target zone
Key	KEY definition that is used as mapping table for leaving the start zone and target zones. First key is used for leaving the target zone. The remaining keys are mapped to the target locations in ascending order.
MaxRt	Maximum response time (like RK)
Cursor	SET Definition that contains at least two pictures. The first picture is used outside of valid zones, the second on target zones. Valid picture types are P, G, R.
Scale	Determines metric of radius <ul style="list-style-type: none"> <li>0 Screen position</li> <li>1 Angle of joystick position (see STICKANGLE)</li> </ul>

## 2.3 WJ (Wait for Joystick Position)

This command waits until the joystick is positioned on a target position for a given amount of time. Note that this event is not recorded.

Syntax: WJ TargPos PosNo [PosRadius [OnTarget [Cursor [Scale]]]]

TargPos	POSITION definition that defines the target position
PosNo	Index of position within TargPos that is used as target position
PosRadius	Radius around TargPos that defines the size of the target area
OnTarget	Time in milliseconds which defined the duration at which the cursor must be on target before the response is accepted
Cursor	Set definition that contains at least two pictures The first picture is used outside of valid zones, the second on target zones. Valid picture types are P, G, R.
Scale	Determines metric of radius <ul style="list-style-type: none"> <li>0 Screen position</li> <li>1 Angle of joystick position (see STICKANGLE)</li> </ul>

## 2.4 Additional columns in result file

Two new columns 'Wn' and 'Rn' will be generated when you use the MM or MJ commands. For each radius one column will be produced

Rn	Latency of crossing boundaries
Xn	X-coordinate of joystick
Yn	Y-coordinate of joysticks
Dn	Radius of joysticks
Wn	Angle of joystick
Tn	Key name (mapped to target zones)

## 2.5 Using Joystick keys

The WK command also handles J1 and J2.

The RK command also handles J1 and J2 to measure latencies but the same restrictions exist like for MM and MJ which means that only the next trial command is executed and then ERTS waits for a joystick response.

There is no continuation and no internal pre-loading of images before the response is made.

## 2.6 Calibration of joystick coordinates

The SETSTICK session command and the AS trial command have been extended to support vertical calibration as well

Syntax: SETSTICK center left right top bottom

Syntax: AS Key PosFlag

PosFlag	Determines what position will be checked
1	left (Xmin)
2	center (X/Y middle)
3	right (Xmax)
4	top (Ymin)
5	bottom (Ymax)

The new session command STICKANGLE can be used to set range of overall movement in terms of angle position

Syntax: STICKANGLE Xangle [Yangle]

Xangle Overall horizontal range from left to right

Yangle Overall vertical range from top to bottom

## 2.7 Test utility TESTJOY

This utility display the current coordinates, radius and angle and the time needed to sample the joystick position. The later determines the timing resolution of MJ and MM.

When starting TESTJOY the current joystick position is interpreted as the center position.

# 3 ERTSCODE

## 3.1 NUMERIC keyword for declaring numeric variables

The implicit declaration of numeric variables has been removed from ERTSCODE to provide better checking of ERSTCODE-jobs. You must now declare each numeric variable at the top of a definition file by using the new NUMERIC keyword:

Syntax: NUMERIC VarName

VarName Any user-defined name for this variable

## 3.2 IMPLICIT keyword

The old-style implicit declaration of numeric variables can be enabled by inserting the new IMPLICIT at the top of a definition file:

Syntax: IMPLICIT

Note that allowing implicit declaration has the potential danger of not recognizing misspelled variable names or non-existing columns in the result file because ERTSCODE interprets them implicitly as new variables.

### 3.3 New Start\_ and \_Final compute blocks

The concept of executing BEGIN and END compute blocks has been extended by START\_ and FINAL\_ to improve handling of multiple input files.

```
START_   is executed in the first data row of the first file before BEGIN and normal block
BEGIN_   is executed in the first data row of each file before normal block
END_     is executed in the last data row of each file after normal block
FINAL_   is executed in the last data row of the last file after normal and END block
```

### 3.4 SHOWLIST

A new SHOWLIST command simplifies the creation of output tables that list data from multiple files. Similar to the OUTPUT command you may list several text variables which will be included in the output list as columns. Global delimiter and justification settings are used.

Syntax: SHOWLIST varname1 [varname2 ... varnameN]

ERTSCODE produces one label line and then one output line per file. Using the /E option you may redirect this output into a file. In contrast to the OUTPUT command you are completely free to arrange and include certain statistical parameters. To extract parameters from an factorial analysis, declare a text variable and assign the value to this variable within an END\_ block:

```
COLUMN MeanRt 10
END_COMPUTE MeanRt = CELL(„Mean“, „Response.Correct“)
```

### 3.5 EXCLUDE

A new EXCLUDE command allows filtering of values that are out of a range whereas the range is computed by using all data. The command functions similar to the SKIP command but causes ERTSCODE to process each file in two phases. In phase one, the EXCLUDE command is not executed, thus all data are processed and all statistics are computed based on the entire data set. In phase two, the EXCLUDE command is processed, thus data that do not meet the WHERE condition are excluded.

Syntax: EXCLUDE WHERE condition

Note that in order to function as a two phase exclusion criteria you must compute variables based on the statistical analysis of the first phase. This is typically done by using END\_COMPUTE statements to set the variables:

```
NUMERIC Mini
NUMERIC Maxi

EXCLUDE WHERE num(R1) GT Maxi OR num(R1) LT Mini

...

END_COMPUTE Mini = CELL("Mean") - CELL("STDV")
END_COMPUTE Maxi = CELL("Mean") + CELL("STDV")
```

### 3.6 Extended /E (Export) command line option

The command line option /E (re-directing SHOW output to a file) has been extended to support RTF-file format and the appending of new output to existing files.

Syntax: /E[WriteMode[:rtf ]]

The WriteMode is one of the symbols ':', '\*', '+', '='.

- : Normal (Overwrite existing files)
- \* First (Overwrite existing files, do not close rtf-file)
- + Continued (Append to existing file, do not close rtf file)
- = End (Append to existing file, close rtf-file)

Adding the string „rtf“ will enable RTF-output format. For example

```
/E:rtf
```

will activate RTF-format. For non-rtf format only WriteModes ':' and '+' make sense.

## 4 Incompatibilities to older ERTS versions

- Default handling of AutoRepeat in RK-commnd has been changed. Unless explicitly enabled by AutoDelay flag of RK command, the autorepeat of PC keyboard will be ignored.
- New DROPOUT session command substitutes the Block randomization modes 4 and 5. These modes are not supported anymore
- The meaning of the 'Sequence' argument of the FB-command has been slightly changed.
- The Level=1 argument of the CC-trial command (Change color) now manipulates the first loaded foreground color and not necessarily the standard foreground color.
- The SW-trial command has been extended by including an argument that defines the index of the critical word within the sentence. The EC and GR trial commands are sensitive to this index.
- The KEYCODE session command is changed to CODEKEYS. Syntax has been changed and only direct writing to port is supported (corresponds to Bios=1).
- Since ERTSCODE V3.08 numeric variables must be declared explicitly by using the new NUMERIC keyword. To use old-style implicit declaration insert the new keyword IMPLICIT at the top of the ERTSCODE-definition file.
- Since ERTS V3.26 the SW command has been extended by an additional argument CritWords. If you have used this command in earlier version you must add a '0' as the third argument.
- Since ERTS V3.29 the German keyword „WC“ for the CC-command (Change Color) is not supported anymore. This keyword now has a new meaning (Wait for External Counter)