E.D.A.

EVENT DETECTION AND ANALYSIS FOR PHYSIOLOGICAL SIGNALS

Manual of Version 1.04

Contents[1]

License Agreement iv

1 Basics 1

Overview 1
Main Modules and Additional Routines 1
Hardware Requirements 2
System Files 3
Data Format 4
Logic of File Names 4
✓Date Format, Version Number and Support 5

2 Event Detection 6

[1] ✓: indicates modifications to previous versions

Interactive Mode 6
Common Fields 8
Unique Fields 11
Graphics 12
Results 13
Parameter Pre-definitions 13
✓Batch Mode 14
✓Memory Resources 14
Methods 15
Electrodermal Activity (EDA) 15
Electrocardiogram (ECG) 18
Power Spectrum 20
Integration 21

3 Event Check 23
Getting Started 23
✓On-line Help and Interrupt Key 23
✓The Definition File 24
The Entry Screen 24
The Upper Screen 24
✓ The Lower Screen 26
Time Analysis and Time Files 27
The Graphic Interface 28
Viewing the Display 28
✓ Controlling the Appearance of a Single Channel 29
Scrolling the Current Display 30
Printing a Screen 30
Selection of Results 31
Measuring: Inserting and Revising Results 32
Time Markings 33
Changing Graphic Color Settings 33

4 Results and Calculations 34
The Result File Converting Utility 34
Listing a Result File 34
Options 36
Calculations 36
Number of Events 36
Heart Period, Heart Rate and IBI 36
Phasic Heart Rate Changes 37
Time Related IBI Statistics 38
✓ Plausibility Check 39
Result File Structure 40

5 Modifying CODAS Data Files 41

References 42

Appendix 43

✓ A. The Definition File ED.DEF 43

✓ B. The Definition File EC.DEF 43

C. An EDA Result File Listing Example 45

Index 47

	Jürgen Kayser	Hans-Jürgen Trosiener
	Poschingerstr. 22 D - 12157 Berlin	Motzstr. 9 D - 10777 Berlin
Tel.	+49 (0)30 855 95 07	+49 (0)30 216 95 71 +49 (0)30 314-25273
Mail:	kays0230@mailszrz.zrz.tu-berlin.de	trosienr@zrzsp3.gp.tu-berlin.de

LICENSE AGREEMENT

SOFTWARE LICENSE

THE SOFTWARE AND THE ACCOMPANYING WRITTEN MATERIALS ARE THE SUBJECT OF COPYRIGHT. IN CONSIDERATION OF PAYMENT OF THE LICENSE FEE THE HOLDERS OF THE COPYRIGHT TO THE SOFTWARE AND ACCOMPANYING WRITTEN MATERIAL GRANT YOU A NON-EXCLUSIVE RIGHT TO USE THE PRODUCT. THE SOFTWARE MAY BE RUN ON A SINGLE MACHINE AT A SINGLE LOCATION SO LONG AS YOU COMPLY WITH THE TERMS OF THIS LICENSE. AN ARCHIVAL COPY OF THE SOFTWARE IS PERMITTED.

LIMITED WARRANTY AND DISCLAIMER OF WARRANTY

WARRANTED IS THAT THE PRODUCT SHALL BE FREE FROM DEFECTS IN THE PHYSICAL MEDIA.

THE SOFTWARE AND ALL ACCOMPANYING WRITTEN MATERIALS ARE PROVIDED 'AS IS' WITHOUT WARRANTY OF ANY KIND. NO WARRANTY IS GIVEN THAT THE SOFTWARE IS FREE FROM ALL BUGS AND OMISSIONS. FURTHER, IT IS NOT WARRANTED OR GUARANTEED AND NO REPRESENTATIONS, EXPRESS OR IMPLIED, ARE MADE REGARDING THE USE, OR THE RESULTS OF USE, OF THE SOFTWARE OR WRITTEN MATERIALS IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, CURRENTNESS, FITNESS FOR A PARTICULAR PURPOSE, OR OTHERWISE. THE ENTIRE RISK AS TO THE RESULTS AND PERFORMANCE OF THE SOFTWARE IS ASSUMED BY YOU.

1 Basics

Overview

The software package *E.D.A.* (Event Detection and Analysis)[2] analyses digitized physiological recordings[3]. It was originally designed to analyze the CODAS data format (for CODAS see chapter 5, p. 41) but it is also possible to use a binary or ASCII format. Significant events will be detected (e.g., R-waves of an ECG or SCRs of an EDA recording) and stored in a result file. These events can be displayed with the original signal. Alterations or corrections of these events are possible.

The detected events or results are stored in relation to the beginning of recording, i.e. in real time. Besides a listing of the results it is therefore possible to look at particular parts of the recording in order to calculate several tonic or phasic parameters of interest from the events, e.g. SCR frequencies, heart rate, heart period, phasic heart rate second-by-second. If one of the recording channels contains time markings, e.g. for stimulus onset, the program is able to detect these and it is possible to consider the events in relation to these time markings.

In short, *E.D.A.* enables the user to generate parameters from the raw signal, the sort of data needed for statistical purposes.

Main Modules and Additional Routines

The package incorporates several independent programs. First, there is the routine *ED.EXE* (Event Detection). Till now, it is possible to investigate recordings of electrocardiography (ECG) and electrodermal activity (EDA), to integrate the data e.g. for electromyography (EMG), and to run a power spectrum for descriptive purposes. The user may specify several presets prior to the investigation, e.g., type of analysis (ECG, EDA, integration, power spectrum), time constant, amplitude criterion, kind of filter (high, low, band, notch, none). *ED.EXE* can show how your choices influence the detection of events (e.g., using filters with different upper or lower borders or using different amplitude criteria). A validated setup (the presets) can be stored in a definition file *ED.DEF* to run the routine in a batch job for several data files, e.g. overnight. *ED.EXE* also transforms the original data, i.e. you can choose between a binary output file containing the filtered signal, the original signal (converted from ASCII or CODAS), or the signal with the time constant eliminated. All features and the usage of the ED

[2] The first version of *E.D.A.* was presented at the 20th annual meeting of the German Psychophysiological Society (DGPA) in Hamburg (Trosiener & Kayser, 1992).

[3] Martin & Venables (1980) is recommended as a standard handbook for description and evaluation of methods in Psychophysiology.

routine are documented in chapter 2.

Second, the *EC.EXE* routine (Event Check) displays the data (original, filtered, time constant eliminated) together with the detected events. Up to eight data channels can be displayed at a time therefore making it possible: to inspect the recording; to compare the activity in several channels by overlaying the data curves; to change the resolution separately for each channel; to jump to different time points in the recording; to revise the detected results (change, add, or, delete an event), and, to select or to reject the detected events. Further, you can investigate a particular channel for time markings, i.e. rectangular impulses, which are simplest the ascents or descents or - more complicatedly - a pattern of two rectangular impulses determined by the time of and between the impulses. The detected discrete time points perform as 'anchors' in order to display a related time frame or time window on the screen, e.g. to collect interactively only those events which fit in your time window. The *EC* routine as well as the *PSCREEN.EXE* routine which prints stored graphic screens while working with *EC* are documented in detail in chapter 3.

The *RECAL.EXE* routine (results and calculations) converts the result files to ASCII code or calculates several parameters from the detected and/or corrected events, e.g. frequencies for a particular part of the recording. Of course, it is possible to direct the ASCII listing to an output file and to calculate own parameters from the results, or - even better - from the original result files for such a calculation. Refer to chapter 4 concerning the RECAL routine.

The *COSAM.EXE* routine (CODAS sampling) works on CODAS data files, i.e. to extract parts of a data file, to reduce a sampling rate, to combine two data files, or simply to inform you about the number of channels, sample rate, length of recording, etc. The COSAM routine is documented in chapter 5.

Hardware Requirements

All routines work in a DOS environment with a base memory of 640 Kb RAM on an ordinary IBM PC or any compatible PC. For the *EC.EXE* routine a color monitor with a VGA or EGA interface is recommended and is absolutely necessary if you are displaying more than one channel. For the *ED.EXE* routine such a color screen is recommended. The other routines will not need a special monitor type.

Depending on the amount of data a large, fast hard disk is recommended. *ED.EXE* will use available extended memory (EMS) which will speed up the analysis. All routines will operate with an installed co-processor which will also increase speed.

So, best is a machine with a 486 CPU, built-in co-processor, 16 MB RAM, 200 MB hard disk and a VGA adapter - but the package will work with a 286-AT, no co-processor, 640 Kb RAM, 20 MB hard disk and a EGA adapter - only much slower.

System Files

Before copying the files you are advised to make a back-up copy of the *E.D.A.* distribution diskette and to store the original data in a safe place as a reference.

The *E.D.A.* distribution diskette contains four self-extracting archive files which store the following files (archive names in the left column):

System Files

ED!.EXE	ED.EXE	- to analyze your raw data (EDA, ECG, integration, power spectrum)
	ED.DEF	- user defined default settings for ED.EXE
	ED.DBF/ED.DBT	- on-line help of the entry menu
EC!.EXE	EC.EXE	- to check the recorded and analyzed data
	EC.DEF	- user defined default settings for EC.EXE
	EC.DBF/ED.DBT	- on-line help of the entry screen
	EC.HLP	- on-line graphic command key reference
UTIL!.EXE	PSCREEN.EXE	- to print stored screens
	RECAL.EXE	- to list and to calculate on result files
	COSAM.EXE	- to modify CODAS data files
	RESULT.PAS	- result file structure (Turbo Pascal code)

Example Files

DEMO!.EXE	DEMO.COD	- CODAS data file
	DEMO.CAL	- the corresponding calibration file
	DEMO.DEF	- the corresponding user default setup (cf. EC.DEF)
	DEMO.D01	- time constant eliminated data channel 1
	DEMO.D02	- time constant eliminated data channel 2
	DEMO.G01	- EDA results of channel 1
	DEMO.G02	- EDA results of channel 2
	DEMO.G04	- ECG results of channel 4
	DEMO.ZMD	- the corresponding time file to channel 5
	DEMO_EMG.COD	- CODAS data file with an EMG recording
	DEMO_EMG.CAL	- the corresponding calibration file

Copy the system files after extraction from the archives to a subdirectory (e.g., C:\EDA) of your hard disk. If you would like to try the examples mentioned in this manual then copy the demo files to your hard disk (e.g, C:\EDA or C:\EDA\DEMO).

Data Format

ED.EXE and EC.EXE work on CODAS data files which are stored in an 12-bit integer format. The data is optionally calibrated to significant units, e.g., μ*mho* or *Mv*, and the calibration constants are stored in a CODAS calibration file (same file name with the extension *.CAL).

Alternatively, binary files may serve as an input to ED.EXE and EC.EXE consisting of real type values (range 2.9×10^{-39} to 1.7×10^{38}, accuracy 11-12 digits, i.e. 1 sign bit, 39 bit fraction and 8 bit exponent). The first data point holds the sample rate, the following data points represent the continuous data flow.

In addition, ASCII files may be used as a data source (only for *ED.EXE* which converts these files to the binary format). Each ASCII value must be separated by a carriage return and a line feed (ASCII decimal codes 13 and 10), i.e. each line contains one value. The first line value consists of the total amount of data points, the second value contains the sampling rate in Hertz. The following lines illustrate an ASCII file. Note, that there is 1) no need for an exponential notation and 2) the comments in the right column are not allowed to occur in the ASCII file.

```
20001                       line 1 (20001 data points)
 2.0000000000E+02           line 2 (frequency 200 Hz)
-6.3461538462E-02           line 3 (first data point at real time 0 sec)
-6.3781324248E-02           line 4 (second data point at 0.005 sec)
-6.6665196610E-02           .
    .                       .
-1.2444866899E-01           .
-1.2791017588E-01           line 20003 (last data point at 100.000 sec)
```

Logic of File Names

All output generated by the modules is named as follows:

<*file name*> . <*file type*> <*channel*>

<*file name*> is a string of 1 to 8 characters, <*file type*> a one-character string, and <*channel*> a two-character string. <*channel*> codes the number of the channel to which the file refers, e.g. *01* for channel one. <*file type*> refers to the type of data or to the kind of result the file exists of.

Table 1 lists the possibilities of <*file type*>.

Table 1.
E.D.A. file names

*.**B**??	binary data: converted ASCII data (ED)
*.**C**??	result file: control data (ED)
*.**D**??	binary data: filtered or time constant eliminated data (ED)
*.**F**??	ASCII overflow listing (ED)
*.**G**??	result file: detected events (ED)
*.**K**??	result file: corrected events (EC)
*.**R**??	result file: real time related phasic heart rates (RECAL)
*.**Z**??	result file: corrected events including time markings (EC)
*.**ZMD**	time file (EC)

Remarks. In brackets the routine which establishes the respective files. *: file name, ??: channel number.

Date Format, Version Number and Support

All dates in this manual and in the software are entered *mm/dd/yyyy* (month/day/year).

In case of inquiries communicate with us (see contact addresses on page iii) and refer to the respective routine, its version number and release date. For example, the following logo parts of the *EC.EXE* (shown with <*Alt-V*> in entry screen, cf. p. 26) and *ED.EXE* routines show the respective version numbers (*1.04* and *0.26*) and release dates (*Jan. 1st, 1993* and *Oct. 12th, 1992*).

```
6444444EC Version 1.04  Release 01/07/1993 E4444447
5                   Event Check                   5
5        Copyright (C) Jürgen Kayser 1992         5
```

```
C)))))))))))))))))))0))))))))))))))))))))))))))))))))0))))))))))))))))))))))))D
 5  00:00:00   step   *        Event Detection        *  Rel. 0.26 10/12/92  5
 5  00:00:00   total  *  (c) H.J.Trosiener  Berlin    *                      5
9444444444444444444444N44444444444444444444444444444444N444444444444444444444448
```

2 Event Detection

One of your first steps in analysis is the detection of interesting events in your data, e.g. the occurrence of R-waves in ECG or maxima in EDA. One of the best ways to become familiar with our programs is to test them with our 'demo' data files.

Interactive Mode

Start the program by typing "*ED* <-" and you will begin in the interactive mode of the program (as you will see later there is a batch mode as well). After the display of the identification screen (normally it takes 5 seconds, but you can skip this by pressing the <*Return*> key) the entry screen appears and ED prompts for information.

The <*Enter*> key terminates the entry process in the current field. Throughout the program the <*F1*> key will give you an on-line help. If you press <*Ctrl-Break*> the program quits as soon as possible.

In the entry screen you are prompted for the

- file name of your data file
- data type
- desired method.

```
  64444444444444444444  F1 = Help    Ctrl-Break = Exit 444444444444 14:41:25 47
5                                                                                5
5  File           DEMO.COD                                                       5
5  Datatype       COD    with   f=  200.00 [Hz]  T=  0: 1 [h:m] 40.01 [s]        5
5  Method          EDA                                                           5
```

Fig. 1. Partial ED entry mask.

File Name and Data Type

Remember that you can choose different types of raw data. The data files can be one of three different types: ASCII, Binary or Codas. The file name extension can define the type of data if you use 'ASC' for ASCII-files, 'BIN' for binary files and 'COD' for Codas-files. If one of these extensions is specified then the field 'Datatype' is automatically set to that type. If your data file has another extension or if the extension does not correspond to your type of data the type must be specified individually.

If you have multiple channels recorded and your data type is not COD then you must choose one channel for analysis and prepare it as a binary or ASCII file.

Examples for valid filenames: CHANNEL1.BIN, VALUES2.ASC, EXPER3.COD, CHANNEL4, VP024, DATA0701.EXP

☞ In our examples please use **DEMO.COD** as filename. This file contains of 5 channels (cf. table 2).

Table 2.
Recording Channels of **DEMO.COD**

Channel	Recording	Unit
1	electrodermal activity (EDA) on left hand	[µS]
2	electrodermal activity (EDA) on right hand	[µS]
3	horizontal eye movements (EOG)	[°]
4	heart activity (ECG)	[mV]
5	time markings of stimulus exposure	[Mark]

Method

EDA calculates the time, the amplitudes and the half-value time of EDA reactions. The type of EDA-reaction is determined.

ECG1 calculates the time of the R-wave. The calculated values represent the absolute maxima of the R-wave complex. Under some circumstances (e.g. small sample rates) the absolute maximum does not represent the best estimation of the time of the R-wave.

ECG2 calculates the time of the R-waves as the best estimation, i.e. it represents a 'virtual' maximum. Use this method only if 'ECG1'-results are insufficient. The procedure is very time consuming and it might be very confusing for presentation purposes if the calculated maximum is not the absolute maximum.

POW calculates the power spectrum of the selected data channel. The data are divided in successive segments of 1024 data points. The spectra are linear averaged and the result is only presented visually.

INT integrates the selected data channel. The integration time constant can be selected over a wide range. The detection process is skipped.

After you have specified the main file name, the data type and the desired method you are prompted with the question: *OK (Y/N) ?* If you type 'N' you can modify all of your entries (you must use the *<RETURN>* key to skip through the fields).

```
  6444444444444444444  F1 = Help    Ctrl-Break = Exit 444444444444 14:41:25 47
5                                                                              5
5  File           DEMO.COD                                                      5
5  Datatype       COD    with   f=  200.00 [Hz]  T=  0: 1 [h:m] 40.01 [s]      5
5  Method          EDA                                                         5
5  Channel number        1    of         5                                     5
5  Range     from    -5.00    to       5.00 [ µS ]                             5
5  Time constant      1.00 [s ]      BESSEL   LOW-filt. 6. order(even)         5
5  Filter  f,flow     2.00 [Hz] fhi         [Hz]     save data: NONE           5
5  Displacement      43.00 pts  t=   0.215 [ s]                                5
5  Threshold : maximum               0.025 [ µS   ] Spec.ParamsN (Y/N)         5
5  Threshold : overlayed maximum     0.025 [ µS   ]                            5
5  Threshold : speed                 0.000 [ µS/s ]                            5
5  Threshold : acceleration          0.000 [ µS/s²]                            5
5                                                                              5
5  Auto scaleN ylo  -6.000      yhi    6.000           Overflows               5
5  Maximum number           at time          [s]      Point number             5
5  Time compression factor( 1 .. 9 ) : 1                                       5
   K))))))))))))))))))0))))))))))))))))))))))))))))))))0)))))))))))))))))))))M
5  00:00:00   step   *        Event Detection        *  Rel. 1.04 01/07/93     5
5  00:00:00   total  *  (c) H.J.Trosiener  Berlin    *                         5
   9444444444444444444N44444444444444444444444444444444N444444444444444444444448
```

Fig. 2. Complete ED entry screen with demo data.

Depending on your method the lower part of the screen is filled with different fields. Some fields are unique to the method, most are common to all methods (cf. fig. 2).

☞ For our demo-example choose the 'EDA'-option as method and after confirmation the screen will change to the 'EDA'-screen. If the data-files have been opened successfully, the lower part of the screen is filled with the appropriate entry-fields. The sample rate and the over-all sample time of your recording is displayed (cf. top of fig. 2).

Common Fields

Channel Number

Channel number must always be specified. If you have CODAS files the number of data channels is displayed and the requested data channel is selected. For all other files the channel number is for identification purposes (results, binary output).

Range (from, to)

Sometimes a signal may be cut because amplifiers were overranged. All parts of data outside a specified range are inspected for signal cuts. Cut is defined here as 3 or more successive points with equal amplitude. All cuts are reported in an error file. The number of cuts is shown in the field 'Overflows'. This is an example part of an error file listing (raw data from DEMO_EMG.COD):

```
Begin   Width    Value  Begin     Width
Point   Points  [µS|mV]  [s]       [s]
```

```
-----------------------------------------
  4330      3   -8.21     2.16    0.0015
  4963      3   -8.21     2.48    0.0015
 10378      4    8.17     5.19    0.0020
 12233      7   -8.21     6.12    0.0035
 12244      3    8.17     6.12    0.0015
```

Time Constant

The specification of this field is necessary. The (RC-)time constant (*tc*) must be ≥ 0.0. It is defined as the time a signal needs to decrease to 37 % of its stationary value. A *tc* of 0.0 indicates that your signal is not affected by a *tc*. A *tc* > 0.0 leads to an elimination of the effect of the specified *tc*. This is done by computing the signal as if it was affected by a *tc* of 100 seconds. This is nearly equivalent to DC amplification[4]. Some manufacturers of psychophysiological equipment use the term 'time constant' in a very confusing and unusual manner (e.g. AC time constant). In doubt determine the (RC-)time constant of your equipment by empirical tests.

Filter Type[5]

Filters are predefined as the Bessel type, which are recommended by constant group delay and low overswing. Conceptually the filter is designed as (recursive) IIR-filter. The precise determination of signals phase shift by bessel filters makes this type most interesting in event-related reaction designs. Furthermore, a Bessel filter has low signal distortion (passband ripple) but less selectivity (no sharp cut offs).

Filter Characteristic

You can choose high-pass, low-pass, band-pass and band-rejection (notch) filters.

LOW — Low-pass filters are used to eliminate high frequencies like EMG in some recordings.

HIGH — High-pass filters are used to eliminate low frequencies, e.g. slow potential moving artifacts.

BAND — Band-pass filters reject low and high frequencies and allow only frequencies in the passband (from flo to fhi) to pass like EEG-filters.

NOTCH — Band rejection filters cut out a frequency band (from flo to fhi) like line noise (50 Hz, 60 Hz).

---

4 Linear transformations of time constant affected data are discussed e.g. by Elbert & Rockstroh (1980) and Thom (1988).

5 See e.g. CookIII & Miller (1992) or Tietze & Schenk (1978) for digital filtering.

Filter Order

The sharpness of the cut-offs and the distortion increase with the order of the filter. If you need a very selective filter then choose a high order. The order of the filters are selectable between 2 and 10 for high- and low-pass filters in steps of 2. For both band filters this is equivalent to 1^{st} to 5^{th} order.

Cut-off Frequency (f,flo,fhi)

High- and low-pass filters need the cut-off frequency as a characteristic parameter. Cut-off frequency is the value where the signal gain is '3 Db down'. From this value the asymptotic gradient of the frequency characteristic has a value of -n x 20 dB per decade with n = order of the filter. Band filters also need the upper cut-off frequency.

Binary Output (save data)

You have different options to save your data in binary format:

ORIG The original data are saved. This is normally done if you have ASCII-data.

TCOR The time constant is eliminated from your original data

FILO You save the filtered signal and if requested the adjusted time constant, too. The group delay of the filter is not adjusted.

FILC You save the filtered signal and also, if requested, the adjusted time constant. The data are adjusted in respect to the group delay of the filter. This is done by cutting the first points in the length of the group delay (shift) and adding values at the end of the data. The last value is repeated until the original sampling time is reached.

NONE No data are saved.

In the interactive mode, existing binary data files with the same name are not destroyed. A message appears on the screen that prompts you to refuse or accept overwriting of binary data. In batch mode data files are overwritten unconditionally.

Displacement (group delay, shift)

The group delay of a filter is the phase shift of the incoming signal which is proportional to the signal frequency. Bessel filter group delay is practically frequency independent and determined by the order of the filter. A high order produces a large group delay or shift of the incoming signal. We estimate the group delay and report it in data points and seconds. Our estimate is the time the outcoming signal of the filters needs to raise from 0 to 50% of the stationary value. We must adjust our detected events with this value. This is already done in all our ED-result files. If you save data for further use with the 'FILO'-option the binary output data are not adjusted!

Unique Fields

Some fields are specific to the method(s), some very sophisticated. The 'special' parameters can improve the detection of events.

Threshold: Maximum

A maximum with an amplitude below this value is rejected, e.g. a value of 0.5 means that only the amplitude of R-waves equal or above this value are reported.

Threshold: Overlayed Maximum (special)

The amplitude of an overlayed EDA reaction must have at least this value. Default value is the maximum threshold. For an explanation of 'overlayed maximum' refer to fig. 5 and 6 (p. 17).

Threshold: Speed / Acceleration (special)

Speed (1^{st} derivate) and acceleration (2^{nd} derivate) control the selection of reactions. Only reaction above the predefined speed and acceleration are considered to be of interest. The values are normalized to speed and acceleration per second. In some special cases the distinction between reaction and recovery shifts can be forced by variation of acceleration threshold. Then only reactions which fit to a combined criterion of speed and acceleration are reported. Normally the parameters are not used.

Window (power spectrum)

Windowing means a weighting data to reduce spectral leakage error in the *Fast Fourier Transform* (FFT). This becomes necessary since the waveforms are sampled for a finite length of time. Discontinuities at the end of the segments produce spectral errors. We have implemented some common windows:

- RECT Rectangular window. No weighting. Leakage is very severe.
- HANN Hanning window. This window is most useful where good frequency resolution is needed and amplitude accuracy is not important.
- HAMM Hamming window. This window is useful for resolving closely spaced frequencies and amplitude accuracy is not important.
- PARZ Parzen window. The data are weighted with a triangle.

R-wave Enhancement (ECG2)

ECG2 convolves the data with a template of a specified frequency and enhances the R-waves. By convolution we get an enhancement of the relevant parts of the R-wave. Under some circumstances this leads to a better recognition of the maxima but the reported time values will not correspond necessarily to the absolute maximum in the R-wave complex.

Graphics

Results may be displayed graphically. A segment of 256 or 256 x time compression points (1 .. 99) is used. Time compression means that more data points are used for a clearer presentation. You can request graphics only in interactive mode by typing 'y' when *graphics (y/n)* appears on the screen.

The binary result files produced in graphical mode should not be used further, because the recursive filter algorithm leads to some problems if the segment lengths are small. We recommend choosing convenient parameters and checking this graphically. You should then run the same job without graphics to produce best results.

Scaling

The graph can be scaled automatically (*Auto scale*). Your data segment is displayed so that it fits exactly into the graphical surface. This is very useful if you have some unpredictably high signal amplitudes or if you want to inspect the visual results exactly.

Manual scaling requires the specification of upper (yhi) and lower (ylo) border of the scaling. Data out of that range are inspected but not displayed.

Results

Range Overflows

Cutting of your data by overranged amplifiers might occur. This is detected automatically and reported in an ASCII error file. The report gives information about the beginning and the length of the cut signal in units of data points and seconds. The amplitude is reported and the number of cuts is reported in the field 'Overflows' of the entry mask.

Event Detection

All detected events (SCRs, R-waves, ...) and related values are written to a result file. Two result files are generated. One (main) holds time, amplitude, type etc. of events, the other holds corresponding speeds and accelerations to give ideas for the sophisticated special parameters. The result files represent the input for program *EC*. Our EDA example produces the main binary result file DEMO.G01. This may is listed by the program RECAL with the command "*RECAL DEMO.G01 <-*" (cf. chapter 4 *Results and Calculations*, p. 34). Appendix C (p. 45) holds this listing.

Parameter Pre-definitions

All parameters can be preset if you use the ASCII file ED.DEF. Although all parameters have some default values which are set automatically by the program, it is not very handy to change some or all parameters manually for each new kind of analysis. Each time the program starts it searches for ED.DEF and uses - if found - all (valid) pre-definitions as default in the entry mask and for the batch mode. If ED.DEF is not present the program uses only internal presets and the batch mode is rejected.

The following example of ED.DEF shows you how such a definition file might look like for EDA method. It comments all possible presettings:

```
cod             ; type of Data(-File) : COD,BIN,ASC
eda             ; type of Analysis (eda, ecg1, ecg2, pow, int)
-5.0            ; lower border of signal range
 5.0            ; upper border of signal range
1.0             ; time constant in seconds, 0 = no time constant
bessel          ; filter type (for future versions)
low             ; filter characteristic (Low, High, Band,
                  Notch (Band rejection), none)
6               ; order of the filter (2, 4, 6, 8, 10)
2               ; upper cutoff-Frequency for LP/HP filter;
                  lower for band filters/notch
0               ; upper cutoff-frequency for band filters and band rejection
none            ; binary Output (tcor, filo, filc, orig, none)
.025            ; Threshold: maximum (smallest amplitude of a maximum)
0               ; Threshold: overlayed maximum
0               ; Threshold: speed
0               ; Threshold: acceleration
```

```
 .200          ; frequency of r-wave for enhancement / or time constant of
                 integrator
 0             ; segment length (DataPoints) in EMS/RAM/disk
                 (0 = automatic use)
```

In the 'real' definition files only one line is used for each parameter! Appendix A (p. 43) shows as another example a typical definition file for the ECG1 method. Any error suspends execution and an error message appears on the screen.

Batch Mode

You can use ED in batch mode as well. Thus, ED has to be used with several call parameters.

Parameters which define batch mode processing:

- -c channel number
- -i input file name

If both parameters are present and the data file can be opened successfully then processing continues in batch mode.

Parameters which work in batch mode and in interactive mode :

- -d data type (overrides all other specifications)
- -m method (overrides all other specifications)
- -p optional another (alternate) definition file than ED.DEF (very useful in case of frequent alterations of the predefinitions)

If a definition file (*ED.DEF* or an alternate file) or the channel number or the filename are missing the program automatically switches to the interactive mode. A wrong channel number stops execution with an error message.

Example: "*ED.EXE -iDEMO.COD -c4 -pECG.DEF -MECG1 -dCOD <-*"

Memory Resources

The program decides whether it can use RAM, EMS or virtual memory on disk for storage of the data segments. The decision depends on available resources. If there is not enough RAM or EMS and the anticipated consumption of disk space exceeds available disk space then the program is terminated with an error message.

You can override the automatic decision process in two ways:

- If you are in the interactive mode and not using graphic display mode (where some kind of fixed length segments are used) you are prompted for change of the segment length if EMS is not sufficient for storage (cf. fig. 3).

- In batch mode a segment length parameter can be set. If it is <> 0 and $\geq$ 1000 then this value is used as the segment length. A value greater than the maximum number of data points is truncated. Besides its usefulness to force EMS usage[6], the consequence of an early program termination produced by a small amount of available disk space can be overridden by specifying a smaller segment length.

```
 644444444 Segment Length 444444447
5 Data file       20001 data pts 5
5 EMS allows      11487 data pts 5
5 We recommend    20001 data pts 5
5                                5
5 Disk will be used. Decrease #  5
5 of data points to use EMS      5
5                                5
5 Your choice     20001 data pts 5
5                                5
  9444444444444444444444444444448
```

Fig. 3. Segment length mask

Methods

Electrodermal Activity (EDA)[7]

The meaning of all values in the entry mask is described under in the former section *Common Fields* (p. 8).

As a result of the detection process you receive the time of the maxima in relation to the beginning of your measurement, the amplitudes of the maxima, the maxima types and the half-value times if computing is possible (cf. *Results*, p. 13, and *Appendix C*, p. 45).

---

[6] It is strongly recommended to use a disk cache as SMARTDRV.EXE or a hardware cache, because this speeds up processing extremely when using virtual storage on disk.

[7] The EDA section was originally based on the algorithms described by Thom (1988) but substantially revised and extended to obtain a maximum of accuracy and reliability.

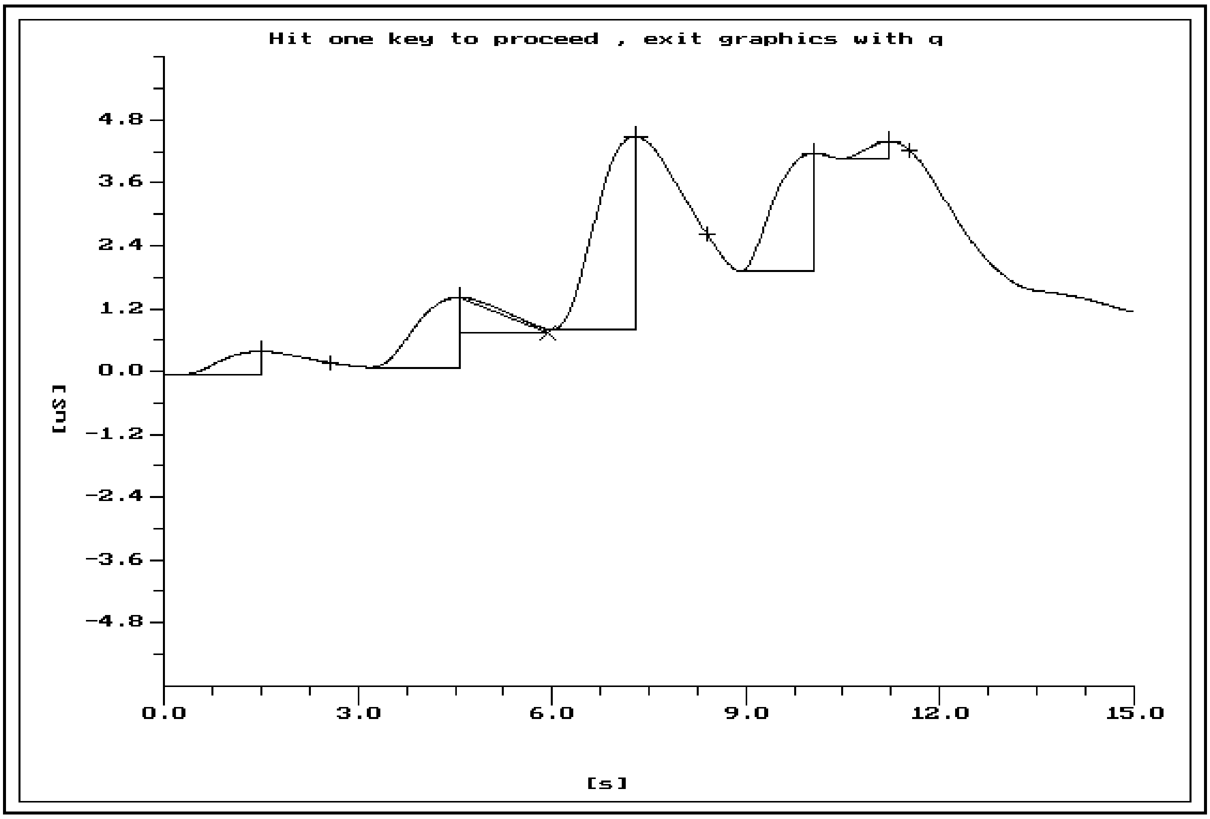

Fig. 4. Graphical presentation of the EDA method (results cf. *Appendix C*, p. 45). Graphic colors are given in table 3.

Fig. 4 shows a normal EDA recording, treated with a six pole 2 Hz bessel filter before the detection process. All maxima and overlayed maxima exceeding a particular amplitude criterion (here ≥ 0.025 µS) are treated as meaningful. In the graphical presentation the y-axis scale is fixed with a time compression value of 3.

Table 3.
<u>Graphic Colors for **EDA** method</u>

color	*graphic feature*
black:	(treated) data
yellow:	points that are shifted to the next segment
green:	angle sides of computed half recovery time
red plus (+):	maximum, reaction
white plus (+):	half value time
white cross (x):	computed half value time
light blue:	angle sides of the maximum, reaction

Five maxima were detected in this example. The 1st, 3rd and 5th. maximum are of type

1 and the half value times could be computed directly. The 2nd and 4th maximum are of type 2 and the half value times had to be estimated. In fig. 4 you can see that for maximum 2 the algorithm leads to a valid estimation. The presentation of the estimation of the 4th maximum was suppressed because the computed value is not very realistic (cf. fig. 5). An estimation is only realistic if we can extract a gradient from the data which leads to a practical determination of an approximation which is near to the assumed original value (cf. fig. 6). This is only possible if there are enough points for an estimation. Even if there are enough points particular circumstances can lead to enormous errors, e.g. a critical sample frequency, a critical AD-converter resolution or a very low amplitude criterion. A visual inspection or a programmatical control of plausibility is strongly recommended!

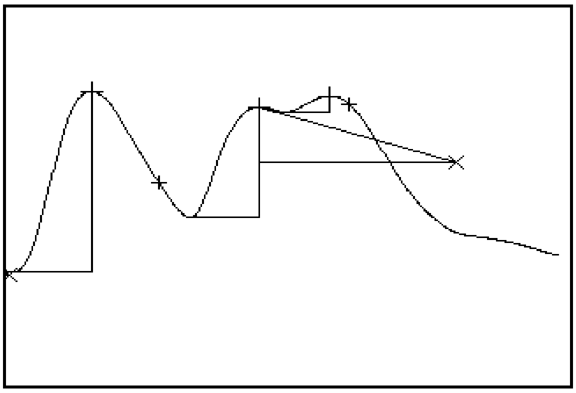

Fig. 5. SCR of type 2: Critical estimation of half value time (maximum 4) (for colors see table 3).

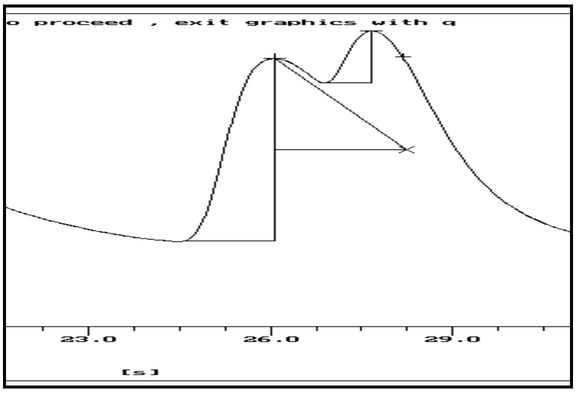

Fig. 6. SCR of type 2: Estimated half recovery time (for colors see table 3).

Classification of Maxima

After Boucsein (1988/1992) we distinguish three different types of maxima in the EDA method (cf. table 4).

Table 4.
<u>SCR Classification</u>

S))Q

Type 1:	Normal maximum, half value time could be computed.
Type 2:	Overlayed maximum, half value time had to be estimated.
Type 0:	Overlayed maximum, half value time could not be estimated because the SCR has no distinct maximum (see 1st reaction in fig. 7). Boucsein (1988/1992) labels this type different as type '3'.

S))Q

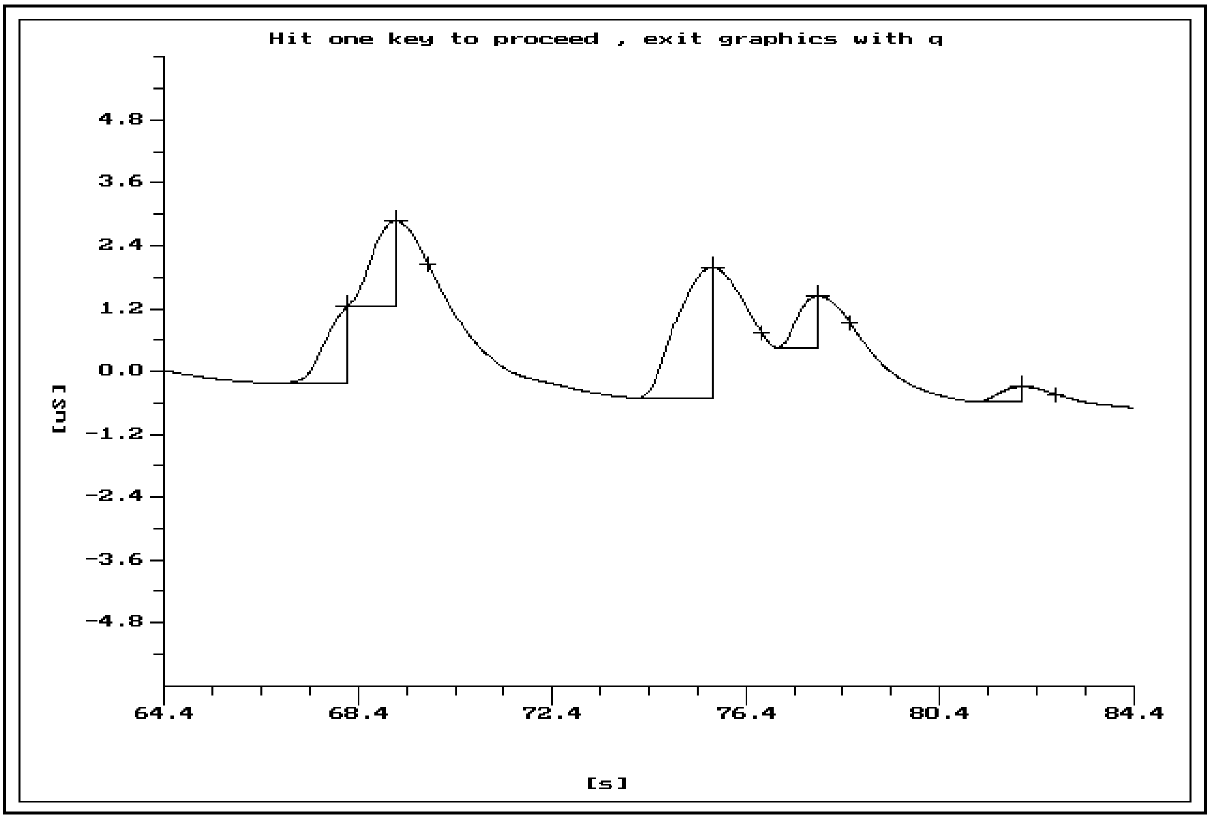

Fig. 7. Reaction of type 0 (1st SCR).

The half value times are always computed if possible, but no plausibility check is made. The graphical presentation of the half value time is suppressed if the value is greater than twice the rise time. A half value time of -1 is coded when an estimation was impossible (normally type 0 of reaction). A half value time of -2 is coded if there are not enough points for an estimation.

Electrocardiogram (ECG)

Most fields of the ECG screen are known (cf. *Interactive Mode*, p. 6). One field is specific for method *ECG2*: R-wave enhancement. Remember that you can choose between two different methods for the detection of R-waves: *ECG1* and *ECG2*. ECG1 determines the absolute maxima in the neighbourhood of a likely R-wave. A convolution of the data with a template of specified frequency is done in ECG2. A frequency value between 8 and 10 Hz is normal. You can determine the frequency when you measure the width of the R-wave where frequency equals the reciprocal of double width (in seconds). By convolution we get an enhancement of the relevant parts of the R-wave. Under some circumstances this leads to a better recognition of the maxima. The computed time values of ECG2 do not always point to the real maxima (i.e. values found in your data) but to hypothetical time values. Sometimes this estimation is better than the time of the absolute maxima, primarily if you have used a slow sample rate, so that the 'real' maximum lies between two data points. Method ECG1 is normally sufficient.

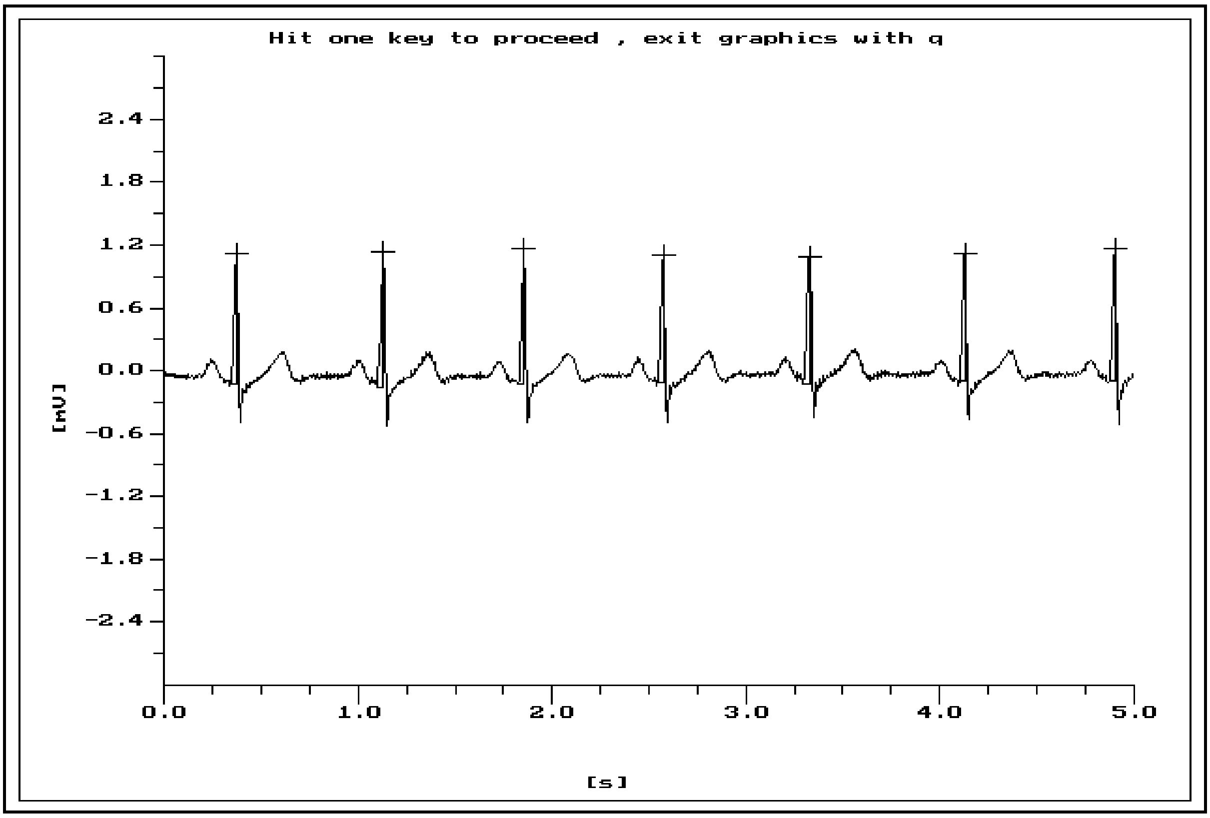

Fig. 8. ECG demo data (channel 4 of *DEMO.COD*). Graphic colors are given in table 5.

Table 5.
Graphic Colors for **ECG** method

color	*graphic feature*
black:	(treated) data
yellow:	points that are shifted to the next segment
red plus (+):	Maximum, R-wave
light blue:	angle sides of the R-wave

Fig. 8 shows the results of ECG1. No filtering was done and all maxima above 0.4 Mv are selected as R-waves. No time compression was requested. As a result you get the time values of the R-waves and their amplitudes:

number	time [sec]	type	amplitude	slope maximum	rise time	recovery /2
1	0.375	9	1.239	0.000	0.000	0.000
2	1.130	9	1.292	0.000	0.000	0.000

3	1.855	9	1.291	0.000	0.000	0.000
4	2.575	9	1.216	0.000	0.000	0.000
5	3.330	9	1.223	0.000	0.000	0.000
6	4.130	9	1.217	0.000	0.000	0.000
7	4.900	9	1.261	0.000	0.000	0.000

Fig. 9 shows an ECG (not part of the demo data !), where some other parts of the ECG have the same amplitude as the R-waves. The program detects the maxima without further intervention.

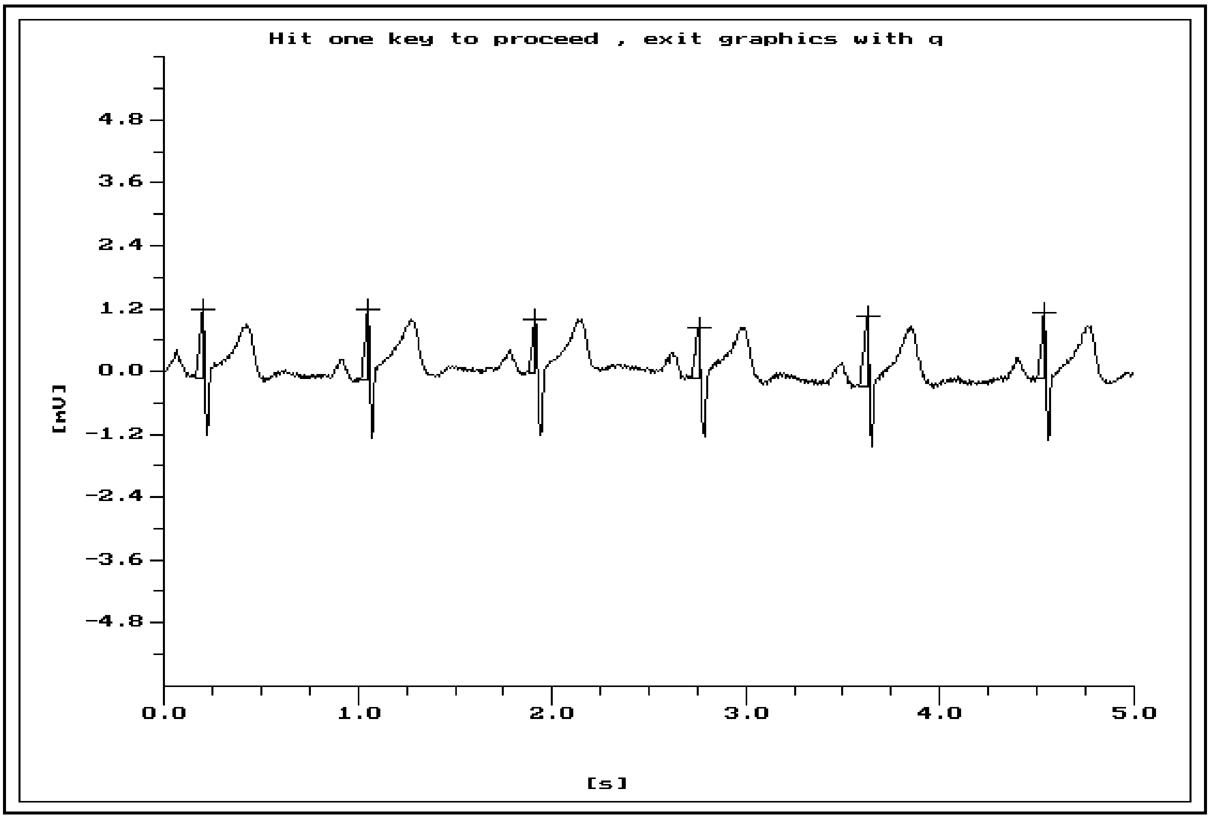

Fig. 9. R-waves together with large T-waves (for colors see table 5).

In both methods ECG1 and ECG2 the type of maximum is set to 9. The slope-maximum, rising and half recovery time are not computed and set to 0.0.

Power Spectrum

The power spectrum is used to control the effects of the requested filtering. As in the other methods you choose a data channel and specify the filter. A window can be requested. Windowing means a weighting of your data to reduce spectral leakage error in the Fast Fourier Transform (FFT). Normally this is not needed.

With the fields 'flow' and 'fhigh' you can change the x-scaling of the graph. If you specify e.g. 40 and 60 Hz you reduce the presentation to the range of 50 Hz noise (without harmonics). The y-scaling is done automatically. You can choose logarithmic or linear scaling.

The method itself divides your data in segments of 1024 points and computes the power spectrum of each segment. The segments are averaged without any weighting. The averaged power spectrum remains on your screen.

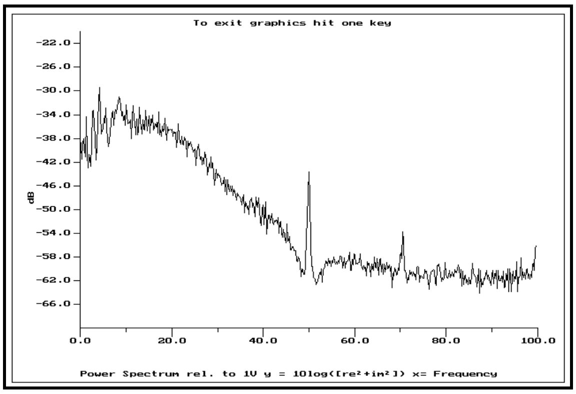

Fig. 10. Power spectrum of ECG-channel.

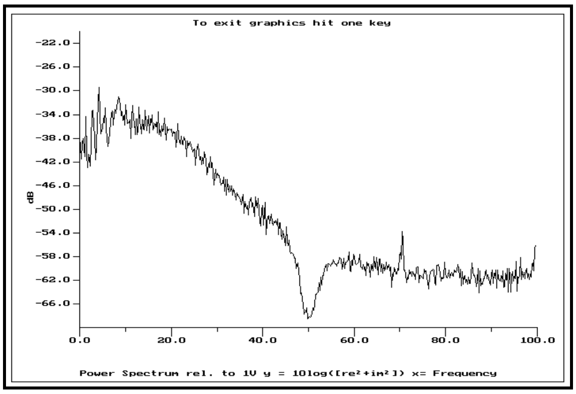

Fig. 11. Power spectrum of ECG-channel with notch filter (48-52 Hz).

Fig. 10 shows the power spectrum of the ECG-channel 4 without any treatment.The scaling is logarithmic and a rectangular window was used. You can see that the signal is superposed by a large amount of 50 Hz noise. If you choose a 2^{nd} order band rejection filter (notch) between 48 and 52 Hz the reduction of 50 Hz noise is obvious (cf. fig. 11).

Integration

The integration method integrates the chosen data channel. The minimum of the used integration time constant (*tc*) has to be the sample rate. If you use this sample rate as *tc* the signal is simply rectified. Larger time constants lead to a signal integration. The signal amplitude is weighted in respect to the time constant. No event detection is done.

The graphical presentation normally shows the original data together with the integrated data. For clarification the two figures below (fig. 12 and 13) show the signals separately. Graph scaling is done in according to the range of the integrated data. If you use a large *tc* the original data are mostly out of range.

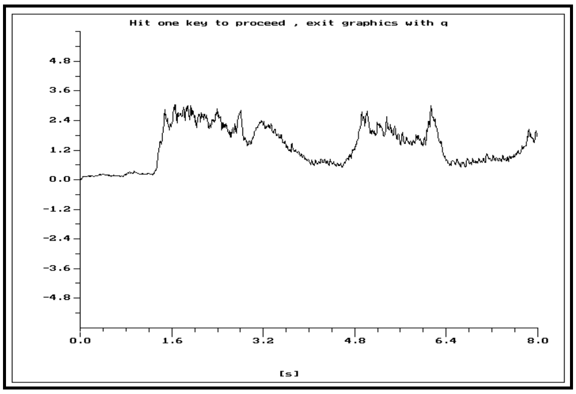

Fig. 12. Integrated EMG (for colors see table 6).

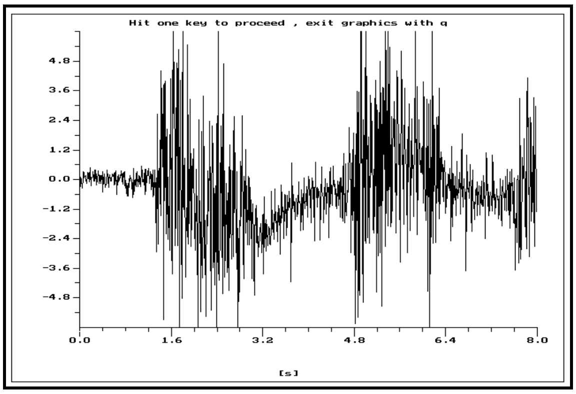

Fig. 13. Original EMG data (for colors see table 6).

Table 6.
<u>Graphic Colors for **Integration** method</u>

color	*graphic feature*
black:	(treated) data
red:	original data

Fig. 12 shows the integrated data of channel 1 (EMG) with an integration time constant of 50 ms. No filtering and *tc*-elimination was used. The time compression factor for the graphic display was 16. Fig. 13 shows the corresponding original EMG recording.

☞ The demo file **DEMO_EMG.COD** holds the original EMG recording data.

3 Event Check

The *EC.EXE* routine displays the data (original, filtered, time constant eliminated) together with events normally detected by *ED.EXE* and allows checking of the events, with optional modification or correction. Thus, this routine may create new event (result) files.

Getting Started

If you type "*EC /?* <-" you will get the following message:

```
EC Version 1.04  01/07/1993
Syntax:  EC [<fil>] [<def>[.DEF]] [<opt>]

 <fil>    data file name
 <def>    file name for definitions (default extension .DEF)
 <opt>    options:
            /?     this help
            /b     binary data
            /c     codas data
```

You may specify your input data file as the first call parameter and the data type either by the option (e.g. */b* for binary data) or by the extension of the data file, e.g. **.BIN* for binary data (see also p. 6).

You may set up your parameters for *EC* with the help of an ASCII text file. You tell *EC* the name of this definition file with the second call parameter (cf. chapter *The Definition File*, p. 24). If there is no second call parameter, *EC* will search per default for a file named *EC.DEF*.

When calling the program by typing "*EC* <-" you will start in the interactive mode. *EC* prompts you to name an input data file and to specify the data type (CODAS data or binary data). The <*Enter*> and <*Esc*> keys quit the entry field.

On-line Help and Interrupt Key

You are allowed to press the <*F1*> key throughout the program. This key will give you an on-line help. While using the on-line help you may switch to different help messages by pressing <*PgUp*> or <*PgDn*>, the <*Esc*> key quits the help. You will get a different set of help messages depending on whether you are working at the entry screen or at the graphic display.

The 'emergency' interrupt key <*Ctrl-Break*> will quit the program as soon as possible. Open files will be closed and occupied memory will be emptied. Use this key only if you are in trouble (!) because in contrast to the normal termination of EC with the <*Alt-X*> key no data are saved.

The Definition File

A definition file holds the particular parameters you need for *EC.EXE*, e.g. significant labels for your data, the display or operation mode, or the kind of printer you use. Normally this information is stored in an ASCII text file named *EC.DEF* but you may also specify another file name when calling the program (cf. chapter *Getting Started*, p. 23). If you have no definition file *EC* will take default settings. You may alter the default settings on-line.

An example file (EC.DEF) is given in Appendix B (p. 43) to show how such a definition file might look like. All possible presettings are commented and are further explained in this chapter.

If you have copied the demo files **DEMO.*** to your hard disk you can try to run EC with "*EC.EXE DEMO.COD DEMO.DEF* <-" or simply "*EC DEMO* <-".

The Entry Screen

The EC entry mask serves as a status report of your work. It is logically divided into an upper and a lower part. In the upper screen the user establishes fixed settings for the work with a particular data file. In the lower screen it is possible to change several parameters throughout the run of EC.

The Upper Screen

Besides the file name and the data type the top of the screen informs you about the sample frequency, sample rate, and, the total file length (cf. fig. 14). If the frequency per channel exceeds 1000 Hz time will be displayed in units of milliseconds (ms).

Below the file information you will get a channel listing of several columns. Depending on your data type there will be a display of eight (binary data) or of the total number of channels (CODAS data). The first column codes the channel number (*Ch.* ranges from 1 to 8). The second column codes whether a channel is active, i.e. to be included in the graphic (*act.* may be 'Y' or 'N'). The third column holds a user defined label for the channel which will be shown in the graphic (max. twelve characters).

Columns 4, 6, 8 hold file names, the adjacent columns 5, 7, 9 labelled with a question mark initiate the files ('Y' vs. 'N').

```
    64444444444444444444444444444444  <F1> help  44444444444444444444444444444447
5 file name  DEMO.COD                                total time     100.000 sec 5
5 data type  COD         frequency   200.00 Hz      sample rate      0.005 sec 5
     K)))))0)))))0))))))))))))0))))))))))))0)0))))))))))))0)0))))))))))))0)0))))))))M
5  Ch.* act.*label       *binary file *?*results     *?*correction  *?*select  5
5   1 *  Y  *SCR left    *DEMO.D01    *Y*DEMO.G01    *Y*DEMO.K01    *Y*   N    5
5   2 *  Y  *SCR right   *DEMO.D02    *Y*DEMO.G02    *Y*DEMO.K02    *Y*   N    5
5   3 *  Y  *EOG         *DEMO.D03    *N*DEMO.G03    *N*DEMO.K03    *N*   N    5
5   4 *  Y  *ECG         *DEMO.D04    *N*DEMO.G04    *Y*DEMO.K04    *Y*   Y    5
5   5 *  Y  *markings    *DEMO.D05    *N*DEMO.G05    *N*DEMO.K05    *N*   N    5
     K)))))2)))))2))))))))))))2))))))))))))2)2))))))))))))2)2))))))))))))2)2))))))))M
                                          :
5                                                                              5
5 Any modifications on entries (Y/N) ? N                                       5
     9444444444444444444444444444444444444444444444444444444444444444444444444448
```

Fig. 14. Upper part of EC entry screen and corresponding prompt.

Column 4 tells *EC* a name of a *binary file*. In the case of a CODAS data type such a specification is optional because the main data file contains the original data for all channels. But you may specify a binary file, e.g. filtered data by *ED*, to display other data than the original CODAS data for this channel. In the case of a binary data type *EC* expects at least one binary data file to be activated, i.e. the binary file name in column 4 must be followed by a 'Y' in column 5 (commonly the main file name). If EC is unable to find a specified file the corresponding column is set to 'N' (EC uses the current DOS path).

Column 6 (*results*) specifies a result file which may be used to display detected events by ED together with the former specified signal. Again, if EC is unable to find the file the corresponding column is set to 'N'.

Column 8 (*corrections*) gives a name to a new output file, i.e. an interactively checked and/or corrected result file. In the graphic display you may select or de-select a particular event from the input result file for the output correction file. Further, you are able to change an event, i.e. to change the base value or the maximum of an event, or to insert a new event (cf. chapter *Measuring*, p. 32). If you want to have the possibility to create a new output result file you must specify 'Y' in column 9. Before overwriting an existing file EC will ask for information. Results files generated by EC may serve as input result files in another EC session.

If you have decided to generate a correction file column 10 (*select*) permits you to determine to accept a priori all results from the input file for the output file (cf. chapter *Selection of Results*, p. 31).

After you have specified the main file name and the data type you are prompted with the question: "*any modifications on entries (Y/N) ?*". If you type 'Y' you can modify any of these entries (use the keys <*cursor up*> and <*cursor down*> to change the channel, <*Enter*> to skip to the next entry, and <*Esc*> to return to the prompt). You may restore the parameter settings from a definition file (cf. *The Definition File*, p. 24). After typing 'N' all channels will be initialized.

Due to the architecture of DOS a maximum of 15 open files at a time is possible. If you exceed this limit, i.e. try to open more than 15 files at a time, the initialization is terminated with the message "*too many open files*". Note that you will need files in addition to the ones you have specified: one for each correction, the time file, the CODAS data file, and the on-line help file.

The Lower Screen

Once you have established your channel entries you will look at a modified screen (cf. fig. 15).

```
5                                                                              5
5 compression:   7                            current time:         0.000 sec 5
5 events:      N                                                               5
5                                                                              5
5                                                                              5
5                                                                              5
5                                                                              5
5                                                                              5
5 <Enter> change      <Esc> graph      <Alt-X> end      <Alt-Y> new file ...   5
     9444444444444444444444444444444444444444444444444444444444444444444444448
```

Fig. 15. Lower part of EC entry screen and main menu.

By pressing <*Enter*> you can adjust the settings of the lower screen. The *compression* rate fixes the time resolution of the graphic display, e.g. to display only every 7^{th} data point as a single pixel (cf. fig. 15). A compression of 1 is the normal and largest time resolution. *Current time* represents the time of the first data point at the left border of the graphic display. You may specify any time within the range of the data file. *Events* sets the mode of correction, i.e. to use time markings as references for controlling results of a recording channel (cf. chapter *Time Analysis and Time Files*, p. 27).

The <*Esc*> key will force EC to read the data and to switch to the graphic display (cf. chapter *The Graphic Interface*, p. 28).

The <*Alt-X*> and <*Alt-Y*> keys will save all corrections (if any). While the former quits the program, the later returns to the upper screen and accepts a new file name.

Pressing <*Alt-V*> will show version number, release date and support address.

Time Analysis and Time Files

Very often it is necessary to look at physiological recordings in relation to stimulus presentation, e.g. specific SCRs to slides, or, phasic heart rate changes. Ongoing events (stimuli) are coded synchronically in form of rectangles in an extra channel. EC checks such a time marking channel for rectangular impulses and stores the times in a time file. These times serve as 'anchors' for the **.Z??* result output file and for a time window display in the graph.

```
5                                                                              5
5 compression:   7                              current time:         0.000 sec 5
5 events:      Y                                                                5
5 time file:   DEMO.ZMD        time window from:      1.000 to:      5.000 sec 5
 5 channel:      5           +)))),  +)))),    duration R1:         0.000 sec 5
 5 height:           1.000  )))- R1 .))- R2 .))) distance R1/R2:      0.500 sec 5
5 edge:        2              0    1  2    3    duration R2:         1.000 sec 5
5                                                                              5
5 detect events (time analysis) (Y/N) ? N                                      5
     944444444444444444444444444444444444444444444444444444444444444444444448
```

Fig. 16. Lower part of entry screen for an event (markings) related correction mode.

After specifying 'Y' for *events* EC prompts you to choose a name for a time file. The default is the main file name with an extension **.ZMD*. If there is an existing file EC will load the file and will display the total *number* of marking times and the first *time* reference (cf. fig. 16). The borders of the *time window* in the graphic display serve only as a visual help (two dotted lines). They are related to the current time reference. A time window border of zero indicates visually the current reference time point.

If there is no time file you may create a new file. For this purpose you must specify an active *channel* which keeps the rectangular impulses. EC assumes a base line of zero for this channel and searches for changes from this base line which are at least **b** of the *height* of the rectangles.

Rectangles of a fixed size only represent a binary information (*on* or *off*), but the duration of a rectangle can give additional information. EC may analyze a pattern of two following rectangles and consider the duration of both rectangles as well as the duration between the rectangular impulses. As you see in fig. 16 you may specify this pattern in the entry fields *duration/distance R1/R2*. A zero time codes any time, i.e. ignore this part of the pattern. For instance, a pattern of *0.000-0.000-1.000* searches for 1-sec impulses, a pattern of *2.000-1.000-0.500* searches for a 2-sec impulses followed by a ½-sec impulse with a delay of one second. The discrete 'anchor' is a *edge* within this pattern, coded by a number between zero and three (cf. schema of fig. 16). If all pattern times match zero, an even edge number (0 or 2) forces EC to search all rising edges, an odd number (1 or 3) to look for all falling edges. All of this specifications may be stored in advance in a definition file (cf. chapter *The Definition File*, p. 24).

When searching for time markings EC checks times with a precision of ±*2 x sample rate* (but at least 25 ms). An ongoing time analysis can be interrupted by the <*Esc*> key. If you then confirm the interruption the time file will contain all time points discovered prior to interruption otherwise the analysis will be continued

You may switch between reference time points in the graphic display (cf. chapter *Time Markings*, p. 33). In the entry screen you are no longer allowed to choose an *current time* (but you can do so from the graphic display using <*F10*>).

A time file may be converted to ASCII code with the RECAL.EXE routine (cf. chapter *Calculations and Other Options*, p. 34).

The Graphic Interface

Viewing the Display

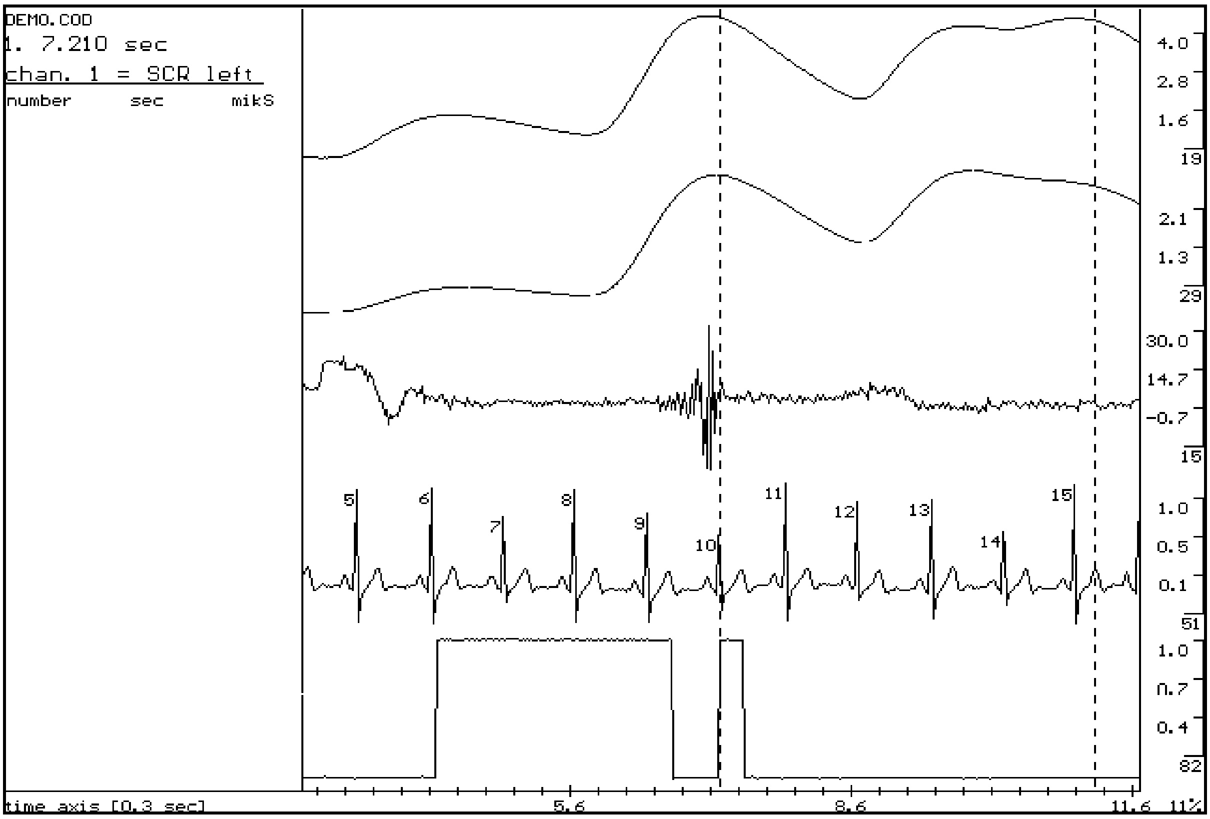

Fig. 17. The whole EC graphic display showing 5 active channels (*DEMO.COD*).

The <*Esc*> key toggles between the entry screen and the graphic display (cf. fig. 17).

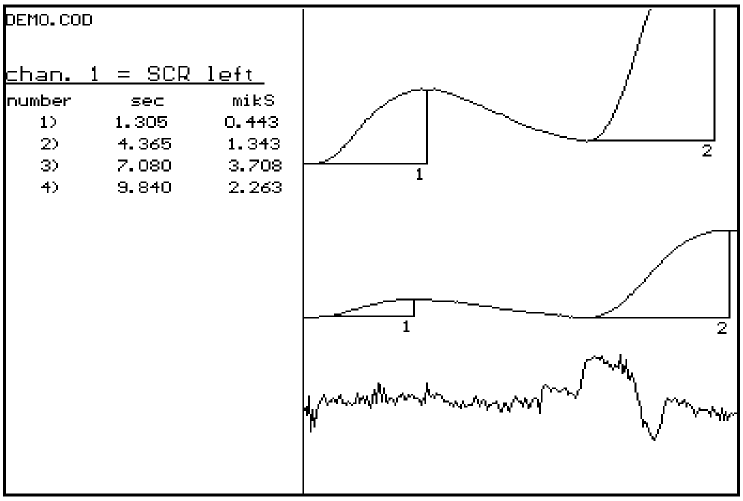

Fig. 18. Top left of graphic display: file name and currently activated channel.

When getting into the graphic mode for the first time or after changing the *actual time* EC will read the data. In the graphic the *<F1>* key always shows an on-line help, the *<F2>* redraws the screen.

The main part of the screen displays the signals of the recordings. If there are results within the time range of the display the results are integrated in each graph with orthogonal lines together with the last two digits of the result number. The top left corner of the screen exhibits the main file name (cf. fig. 18). Below the file name you see the number of the currently activated channel together with the corresponding label. Result parameters of this channel are listed below (number, time of maximum, amplitude in units).

At the bottom of the screen a time axis informs you about the time scaling, start and end time of the current display, and the percentage of data read so far (cf. fig. 19). The right panel displays the scaling of each channel in real units. The number at the bottom of each scale represents an arbitrary scaling factor which may range from 1 to 9999 (see next section).

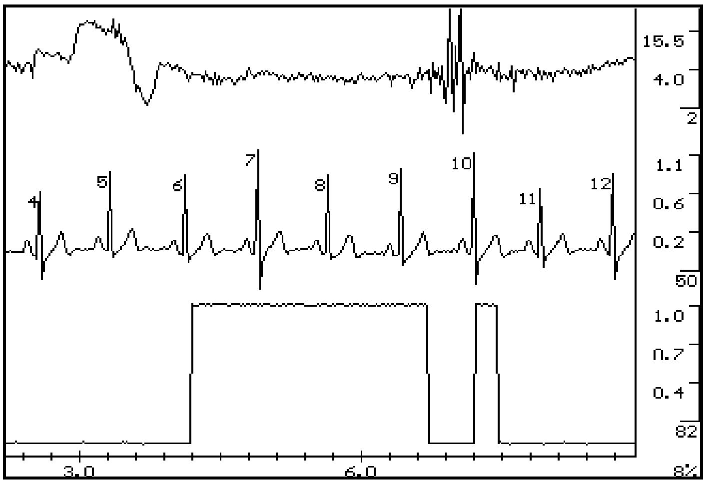

Fig. 19. Bottom right of graphic display: time axis, channel scaling, percentage of data read.

Controlling the Appearance of a Single Channel

If you want to select another channel instead of the one currently activated you simply have to type the number of this channel (1-8). Only the channel displayed at the left panel may be modified.

The keys <*Cursor up*>, <*Cursor down*>, <*PgUp*>, <*PgDn*>, <*Ctrl-PgUp*> and <*Ctrl-PgDn*> adjust the scaling factor of this channel. Alternatively, you use the <*F4*> or <*Alt-S*> keys and specify the scaling factor directly (press <*Backspace*> or <*Delete*> to change the former scaling factor and enter the new one). The key <*F3*> recalculates the scaling factor and the best fitting horizontal position of the graph line (this is the initial default display). The <*F5*> key allows you to set the scaling factor of all measuring channels at once, i.e. of channels creating a new correction result file.

Per default all channel are 'auto scaled' which is exactly what the key <*F3*> does. But it is also possible to preset the scaling factor for each channel via a definition file (cf. the example in the chapter *The Definition File*, p. 24). In such a file scaling factor settings are coded by the initial characters "SF" followed by the channel number (two digits) and a non-significant character (e.g. double point) and the scaling factor value. The scaling factor has to be in the range 1 to 9999.

Under some circumstances you need to adjust your input data in order to get a good resolution and scaling on the graphic display. If your data represents very coarse or very refined units you can multiply your original data by any factor by using the keys <*F6*> or <*Alt-A*>. A value smaller than one will divide the original data which will fit for a coarse differentiation. Values larger than one increase the original data which should be done with very refined input data. The adjustment process will not affect your data, and it operates only as an adaptation for the graphic display. Again, you can preset the adjustment factor for each channel via a definition file (cf. the example in the chapter *The Definition File*, p. 24) where the adjustment factor settings are coded by the initial characters "AF" followed by the channel number (two digits) and a non-significant character (e.g. a double point) and the adjustment factor value. There is no limit for the adjustment factor (a value equalling one has no effect).

With the help of the key <*Home*>, <*Ctrl-Home*>, <*End*> and <*Ctrl-End*> you move the graph in a vertical direction up and down, respectively.

Scrolling the Current Display

To look through the whole recording the keys <-> and <+> move the current display half a screen back and forth, respectively, i.e. the former centre of the screen will be left or right in the next screen. By pressing the <*F10*> key you may jump to any time position (start time of the display) within the recording range.

Printing a Screen

Typing <*Ctrl-PrtScr*> forces EC to draw a pixel image of the screen. The drawing may be directed to the printer or to a print file. The printing process is displayed on the screen (cf. fig. 20) and may be cancelled by pressing <*Esc*>. You may setup your printer in the definition file (cf. chapter *The definition file*, p. 24) or on-line (after <*Ctrl-*

PrtScr> type <*Esc*> and then the printer code; use <*F1*> for help).

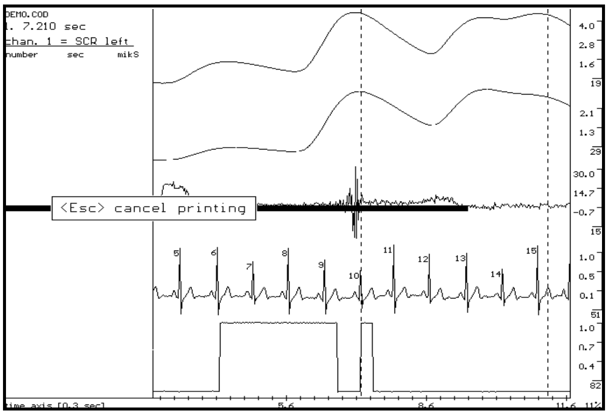

Fig. 20. Screen printing in progress.

If you confirm the prompt question (which also shows the printer model) you are asked whether to send the screen image to a file or to the printer. Files are named SRCEEN00.PRN, SCREEN01.PRN, etc. in regard of existing print files. You may print the files later with the screen file printing utility PSCREEN.EXE. Typing "*PSCREEN* <-" will show the message below. Remember to delete your screen print files after printing when you are no longer using them.

```
EC Screen File Printing Utility  Version 1.04  01/07/1993
Syntax:  PSCREEN <PrintFile>[.PRN]  [<lpt-no>]

The default is LPT1.
Example: "PSCREEN screen01 2"   -   print file SCREEN01.PRN at LPT2
```

Selection of Results

EC will indicate with different colors whether a particular result is selected for the output result file or not. Per default the selected results are highlighted, i.e. they are in the same color as the graph line (you may change the color settings, cf. chapter *Changing Graphic Color Settings*, p. 33). Depending on column 10 of the entry screen the results of a channel are either selected of unselected (cf. chapter *The Upper Screen*, p. 24).

You will activate a result in the display by pressing <*cursor left*> (left arrow key) or <*cursor right*> (right arrow key) which shifts between adjacent results. An activated result is highlighted in white in the list on the left as well as in the graph.

Pressing <*Return*> will select or de-select the activated result for the result output file (depending on the former status quo). An acoustic signal of either a high (now selected) or low (now unselected) frequency indicates your operation.

An activated result can be deleted by pressing <*Delete*>. You have to confirm your decision because the activated result will be removed from this EC session and it cannot be reinserted - but the original input result file will not be affected.

Note that when working with time markings results are <u>not</u> (!) implicitly selected when fitting the time window. You have to do this explicitly by activating the respective results and pressing <*Return*> which is followed by the high frequency tone.

Measuring: Inserting and Revising Results

The keys <*Insert*> or <*Alt-I*> are used to activate the measuring panel on the left of the screen as well as a small cross in the middle of the screen. The cross behaves like a cursor: if you press the <*cursor left*> or <*cursor right*> keys the cross moves and the actual value and the corresponding time are displayed in the measuring panel (cf. fig. 21). Together with the <*Ctrl*> key the cursor movement is speeded up.

In fact, when determining an amplitude you need to specify two data points, one for the base value (onset) and the other for the maximum of the amplitude. Pressing <*Space*> fixes one data point and moves to the other data point which is represented by a second cross. Again, you can move the second cross with the help of the cursor keys. As before, <*Space*> will fix this data point and jump back to the first data point (and so on). Values and times of both data point together with their differences are shown on the panel.

If you have specified any compression larger than 1, *EC* will zoom the area around the actual data point when pressing <*Space*>. This gives you the opportunity to select each data value by moving a new cross within the zoom window. Pressing <*Space*> or <*Esc*> closes the zoom window and shifts to the other data point.

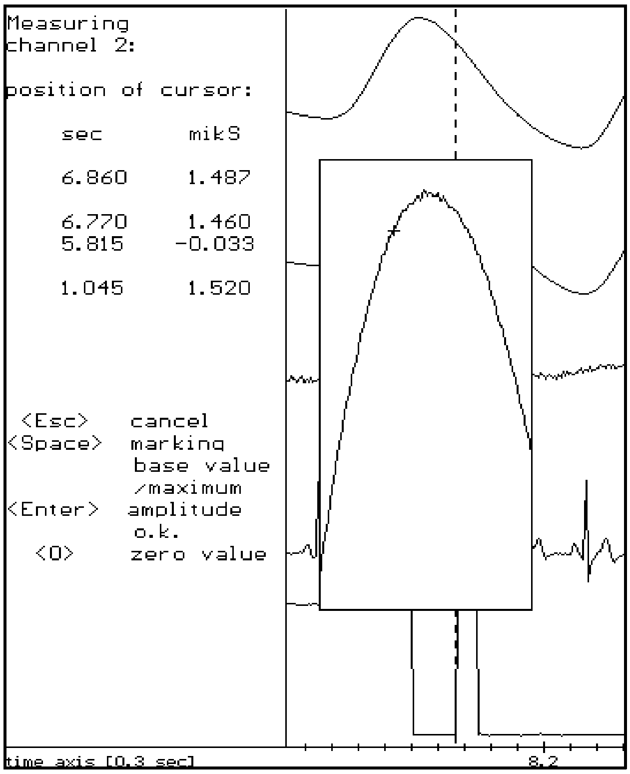

Fig. 21. Measuring an event: Zoom window with cursor (right hand) and corresponding status panel (left hand).

After choosing two data points the key <*Enter*> enters your decision as a choice for a new result. If you confirm your decision after prompting the result is automatically selected for the output result file. The <*Esc*> key will cancel the measuring.

Sometimes you do not need two data points, e.g. when specifying the R-wave peak or another zero amplitude. In this case you simply press the key <*0*> (otherwise you have to set both data points exactly to the same position). Note that when correcting ECG results you should not measure the R-wave amplitude. Calculations of RECAL.EXE refer to the difference of *time* (onset time of a particular amplitude) minus *rise time* (cf. chapter *Calculations and more*, p. 36). Normally you take the R-wave peak as the reference time, and therefore the rise time should be zero.

In contrast to the key <*Alt-I*> which will always insert a new result the <*Insert*> key will modify an existing result if activated. In this case you may change the base value and the maximum of the activated result. The cursor will start at the base value of the particular result and swap to its maximum.

Time Markings

If you are working with a time file (cf. chapter *Time Analysis and Time Files*, p. 27) the current time marker (number and time) is displayed at the top left corner between file name and channel label (cf. fig. 17). All manual result selection will store this time as the event time, i.e. the results of the *.Z?? files are related to this marker time (cf. example 2 in chapter *Listing a Result File*, p. 35). When working with time markings the <*Insert*> key will activate the cursor cross in the middle of the time window (when no result is formerly activated).

You may jump to different time markers by either using the <*F9*> key (choose any marker number within the range of detected markers) or the keys <*Tab*> and <*Shift-Tab*> to skip to the next and previous time marker, respectively.

It is also possible to insert or delete a time marker manually. The key <*F8*> insert a new time marker at the specified time. The key <*F7*> deletes the time marker. Both keys adjust the time file. The keys <*F7*> to <*F9*> as well as the <*Tab*> keys are only functional when using a time file.

Changing Graphic Color Settings

The default color settings of EC may be altered by the user. Pressing the <*Shift-F1*> key toggles between eight different background colors (using EGA or VGA).

The <*Shift-F2*> key activates a left panel to modify the colors of the currently activated channel. The keys <*Space*>, <*Home*> and <*End*> toggle between fifteen foreground colors (EGA/VGA) for the graph line, the selected, and, the unselected results, respectively. The <*Esc*> key quits the channel color panel.

You will save the current color settings with <*Shift-F3*> to a file named EC.PAL. When starting the program EC searches in the current DOS directory for a file EC.PAL with the color settings.

4 Results and Calculations

The Result File Converting Utility

The RECAL.EXE routine is a multi-purpose program to operate on result files. By typing "*RECAL <-*" (or with */?* as any call parameter) you will get the following help screen:

```
EC Result File Converting Utility  Version 1.04  01/07/1993
Syntax:  RECAL [<switches>] <input file> [<output file>]

switches:
  /a              file converting: ASCII => binary (default: binary => ASCII)
  /index           show sample rate index of events in listing
                    (default: show event number)
  /n[<t1>:<t2>]   number or events                       * optional from time
  /stat           time related statistics (only with /n)  * <t1> to time <t2>
  /p[<t1>:<t2>]    IBI ("heart time")                       * in seconds
  /pk[<ll>:<ul>]   IBI plausibility control
                    <ll> lower HR limit (interrupts listing)
                    <ul> upper HR limit (interrupts listing)
  /r[<t0>:<tu>]    IBI phasic changes (real time)
                    <t0> reference time (default: times from time file *.ZMD)
                    <tu> calculation time         (default:  1 [sec])
                   => creates a new result file *.R??
  /rt<t1>:<t2>     specifications for IBI phasic changes (real time)
                    <t1> calculations before <t0> (default:  5 [sec])
                    <t2> calculations after  <t0> (default: 10 [sec])
  /z               convert time file
  /?               this help
```

RECAL expects at least a result file name (*<input file>*) to convert the binary information into readable ASCII code. All ASCII output to the screen may be directed to an ASCII text file if you specify an *<output file>* name.

Listing a Result File

Example 1 shows an EDA result file created by ED. The column *type* refers to the kind of SCR, i.e. *1* codes a normal SCR, *2* codes an SCR with an interpolated *recovery time/2* (because of a new SCR), and *0* codes an SCR with no recovery at all (*recovery time/2* is set to *-1.000*). The column *time* holds the real time of the maximum of the SCR amplitude. The onset of the SCR (base value) is this *time* minus the *rise time*.

```
file name (binary)            DEMO.G01
file name (ASCII)             DEMO.EDA
threshold: maximum            0.0250
threshold: overlayed maximum  0.0250
threshold: speed (slope)      0.0750
```

```
percentage of maximum = onset    5.00
time constant in seconds         0.98        sample rate  200.00  Hz
delay in sample points          43.00    filter frequency    2.00  Hz

  number         time  type   amplitude      slope        rise   recovery
                 [sec]                      maximum       time    /2
---------------------------------------------------------------------------
       1        1.305     1       0.443      0.671       1.210      1.080
       2        4.365     2       1.343      1.770       1.350      1.387
       3        7.080     1       3.708      4.938       1.265      1.110
       4        9.840     2       2.263      3.631       1.125      3.393
       5       10.995     1       0.331      0.708       0.710      0.325
                                    :
      21       86.350     1       0.151      0.234       1.070      0.790
      22       91.490     1       1.878      2.500       1.695      1.235
      23       94.130     1       1.388      2.314       1.065      0.865
      24       97.725     1       2.291      3.107       1.355      1.025
ok
```

Example 2 shows an EDA output result file created by EC while using time markings. Only a few results from the input result file have been selected. The column *recovery time/2* is replaced by *event time*, i.e. the actual reference time of the time file when selecting a particular result. The column *slope maximum* is replaced by *latency*, i.e. *time* minus *rise time* minus *event time*. As there has been an interactive correction (cf. chapter *Measuring*, p. 32) there is a *-1* in the column *type*.

```
file name (binary)             DEMO.Z01
file name (ASCII)

  number         time  type   amplitude    latency        rise      event
                 [sec]                                    time
---------------------------------------------------------------------
       1        9.840     2       2.263      1.505       1.125      7.210
       2       10.995     1       0.331      3.075       0.710      7.210
       3       39.660    -1       1.913      3.225       1.245     35.190
       4       62.710     1       0.485      2.310       1.160     59.240
```

Example 3 shows an ECG result file created by ED. Note that *time* holds the peak of the R-wave and that there is always a *rise time* of zero.

```
file name (binary)             DEMO.G04
file name (ASCII)
threshold: maximum             0.5000
threshold: overlayed maximum   0.5000
threshold: speed (slope)       0.0750
percentage of maximum = onset    5.00
time constant in seconds         0.00        sample rate  200.00  Hz
delay in sample points           0.00    filter frequency    0.00  Hz

  number         time  type   amplitude      slope        rise   recovery
                 [sec]                      maximum       time    /2
---------------------------------------------------------------------
       1        0.375     9       1.216      0.000       0.000      0.000
       2        1.130     9       1.230      0.000       0.000      0.000
                          :
      10        7.200     9       1.201      0.000       0.000      0.000
      11        7.910     9       1.209      0.000       0.000      0.000
```

Options

You may use the switch */a* to reconvert from an ASCII file generated by RECAL to a binary file but this will only work if the ASCII listing contains the head entries, i.e. sample rate, etc.

The switch */index* will show the sample rate index of each event instead of the event number.

Example 4 shows a time file (DEMO.ZMD) listing generated with the switch */z*.

```
1            7.21000 sec
2           35.19000 sec
3           59.24000 sec
4           82.21500 sec
```

Calculations

The switches */n*, */stat*, */p*, */r* and */pk* are used to calculate on a result file.

Number of Events

The switch */n* counts the number of events.

```
n = 24 results in DEMO.G01 ...
```

Together with a time range specification only the events fitting the time range are regarded. The critical time for fitting is column *time* minus column *rise time*. For example, */n50.0:100.0* calculates on a 50 seconds interval.

```
n = 73 results (50.000 - 100.000 sec => 87.600 n/min) in DEMO.G04 ...
```

Heart Period, Heart Rate and IBI

The same logic holds for the switch */p* which calculates heart periods (HP) and heart rate (HR) on IBIs, i.e. the time between adjacent results.

```
IBI  time [sec]        HP [sec]      HR [bpm]
  1      20.810         0.72000        83.333
  2      21.560         0.75000        80.000
  3      22.285         0.72500        82.759
  4      23.015         0.73000        82.192
  5      23.765         0.75000        80.000
  6      24.480         0.71500        83.916
```

```
       7      25.145          0.66500        90.226
       8      25.765          0.62000        96.774
       9      26.365          0.60000       100.000
      10      26.970          0.60500        99.174
      11      27.620          0.65000        92.308
      12      28.360          0.74000        81.081
      13      29.065          0.70500        85.106
      14      29.760          0.69500        86.331
14 IBI (n = 15, 20.000 - 30.000 sec) in DEMO.G04 ...
ok
```

Phasic Heart Rate Changes

The switch */r* calculates on IBIs phasic changes for real time. For example, the switch */r50.0* calculates for the reference time 50.0 seconds for steps of 1 seconds (default) heart period and heart rate. Additionally, EC calculates a mean heart rate for the interval before the reference time and the difference heart rate to the mean heart rate.

```
          Interval [sec]      HP [sec]    HR [bpm]      [bpm] Ø
 -5    45.000-    46.000    0.73385000      81.761
 -4    46.000-    47.000    0.73315000      81.839
 -3    47.000-    48.000    0.67200000      89.286
 -2    48.000-    49.000    0.68990000      86.969
 -1    49.000-    50.000    0.69505000      86.325
  0    50.000-    51.000    0.73050000      82.136      85.132 Ø
  1    51.000-    52.000    0.69207500      86.696       1.564
  2    52.000-    53.000    0.65802500      91.182       6.050
  3    53.000-    54.000    0.71585000      83.816      -1.315
  4    54.000-    55.000    0.72975000      82.220      -2.912
  5    55.000-    56.000    0.70805000      84.740      -0.392
  6    56.000-    57.000    0.70665000      84.908      -0.224
  7    57.000-    58.000    0.69680000      86.108       0.976
  8    58.000-    59.000    0.70410000      85.215       0.083
  9    59.000-    60.000    0.72160000      83.149      -1.983
```

You may use a time file as a reference time input. If the first time of the switch */r* equals zero (*/r* or */r0.0:1.0*) RECAL searches for a time file *.ZMD. If there is no file you are prompted for a time file name. RECAL will calculate real time phasic changes for all times of the time file.

Together with the switch */rt* you may change the default settings of the pre- and post-intervals. The next example shows a listing with the switch settings */r0:0.5* and */rt6:10* while DEMO.G04 served as the input result file.

```
time file: DEMO.ZMD
          Interval [sec]      HP [sec]    HR [bpm]      [bpm] Ø
  1     7.210
 -6     4.210-     4.710    0.77500000      77.419
 -5     4.710-     5.210    0.75640000      79.323
 -4     5.210-     5.710    0.74955000      80.048
 -3     5.710-     6.210    0.78000000      76.923
 -2     6.210-     6.710    0.77715000      77.205
 -1     6.710-     7.210    0.77370000      77.549
  0     7.210-     7.710    0.71000000      84.507      78.061 Ø
```

```
  1     7.710-     8.210    0.74300000      80.754       2.693
  2     8.210-     8.710    0.76675000      78.252       0.192
  3     8.710-     9.210    0.79000000      75.949      -2.111
  4     9.210-     9.710    0.78265000      76.663      -1.398
  5     9.710-    10.210    0.77500000      77.419      -0.641
  6    10.210-    10.710    0.75150000      79.840       1.780
  7    10.710-    11.210    0.72360000      82.919       4.858
  8    11.210-    11.710    0.69240000      86.655       8.594
  9    11.710-    12.210    0.73000000      82.192       4.131
  2    35.190
 -6    32.190-    32.690    0.73060000      82.124
                        :
```

In case of the switch */r* a new result file *.R?? is generated by RECAL which may be later inspected with RECAL. Compare the following listing of the file DEMO.R04 with the previous example.

```
file name (binary)            DEMO.R04
file name (ASCII)

  number  event time   No.   off-time   cal. IBI         HR   mean HR/
               [sec]            [sec]      [sec]              delta HR
------------------------------------------------------------------------
       1       7.210    -6      4.210      0.775     77.419      0.000
       2       7.210    -5      4.710      0.756     79.323      0.000
       3       7.210    -4      5.210      0.750     80.048      0.000
       4       7.210    -3      5.710      0.780     76.923      0.000
       5       7.210    -2      6.210      0.777     77.205      0.000
       6       7.210    -1      6.710      0.774     77.549      0.000
       7       7.210     0      7.210      0.710     84.507     78.061
       8       7.210     1      7.710      0.743     80.754      2.693
       9       7.210     2      8.210      0.767     78.252      0.192
      10       7.210     3      8.710      0.790     75.949     -2.111
      11       7.210     4      9.210      0.783     76.663     -1.398
      12       7.210     5      9.710      0.775     77.419     -0.641
      13       7.210     6     10.210      0.752     79.840      1.780
      14       7.210     7     10.710      0.724     82.919      4.858
      15       7.210     8     11.210      0.692     86.655      8.594
      16       7.210     9     11.710      0.730     82.192      4.131
      17      35.190    -6     32.190      0.731     82.124      0.000
      18      35.190    -5     32.690      0.735     81.633      0.000
      19      35.190    -4     33.190      0.742     80.917      0.000
                         :
```

Time Related IBI Statistics

The switch */stat* calculates several time specific statistics for IBIs, i.e. mean (MEAN), standard deviation (STDEV), variance coefficient (VARCOEF), mean sum of squares of successive differences (MSSD), instantaneous heart rate (IHR). For a detailed explanation of these statistical parameters and their relevance we recommend the article of van Dellen, Aasman, Mulder & Mulder (1985) as a reference. In contrast to the switch */n* all calculations refer to the time interval between the first and the last heart beat. For clarification look at the following three examples (corresponding commands are entered after the DOS prompt):

```
C:\EDA\DEMO>recal demo.g04 /n15:75 /stat

n = 86 events (15.000 - 75.000 sec => 86.000 n/min) in DEMO.G04 ...

n = 85 IBI (15.435 - 74.660 sec => 86.112 n/min)
MEAN    =       696.7647058824 msec
STDEV   =        43.9338374472 msec
VARCOEF =         6.3054051212
MSSD    =      1245.8333334122
IHR     =         1.4352047277 Hz

ok

C:\EDA\DEMO>recal demo.g04 /n15:75

n = 86 events (15.000 - 75.000 sec => 86.000 n/min) in DEMO.G04 ...
ok


C:\EDA\DEMO>recal demo.g04 /n /stat

n = 143 events in DEMO.G04 ...

n = 142 IBI (0.375 - 99.925 sec => 85.585 n/min)
MEAN    =       701.0563380281 msec
STDEV   =        45.4794435129 msec
VARCOEF =         6.4872737105
MSSD    =      1128.5460992791
IHR     =         1.4264188850 Hz

ok
```

Plausibility Check

The switch */pk* is used to check the IBIs for extra beats (systoles) or artifacts. For example, the switch */pk55:85* will list all IBIs and their respective HR but will pause the listing whenever an IBI of an HR below 55 bpm or over 85 bpm appears.

```
C:\EDA\DEMO>recal demo.g04 /pk55:85

     IBI         HP [sec]      HR [bpm]
       1       1.130          0.75500        79.470
       2       1.855          0.72500        82.759
       3       2.575          0.72000        83.333
       4       3.325          0.75000        80.000
       5       4.125          0.80000        75.000
       6       4.900          0.77500        77.419
       7       5.645          0.74500        80.537
       8       6.425          0.78000        76.923
       9       7.200          0.77500        77.419
      10       7.910          0.71000        84.507
      11       8.675          0.76500        78.431
      12       9.465          0.79000        75.949
      13      10.240          0.77500        77.419
      14      10.990          0.75000        80.000
      15      11.680          0.69000        86.957
```

```
---- <Return> ----
      16      12.410          0.73000        82.192
      17      13.180          0.77000        77.922
      18      13.950          0.77000        77.922
      19      14.695          0.74500        80.537
      20      15.435          0.74000        81.081
      21      16.125          0.69000        86.957
---- <Return> ----
                   :
```

Result File Structure

Experienced programmers can use the result files created by ED.EXE, EC.EXE and RECAL.EXE to calculate their own parameters. For this reason the storage format of the events in the result file is given as a part of a **BORLAND® Turbo Pascal** source code in the file *RESULT.PAS*. Regard the comments in the source code!

```
{ RESULT.PAS                          (C) 1992 H.-J. Trosiener & J. Kayser }

{ The following declarations refer to the file structure of result files
  created by ED.EXE, EC.EXE and RECAL.EXE.

  The first record of a result file consists of <ResultType> = 2,
  the second record of <ResultType> = 3, and all following records
  are of <ResultType> = 1.

  Depending on the kind of <CodeType> the variables of <ResultType> = 1
  differ from the meaning of the variable name which hold only for
  <CodeType> = NormalType.                                               }

TYPE CodeType    = (NormalType,
                    SlopeType,
                    LatencyType,
                    IBI_RealtimeType);

     ResultType  = RECORD CASE INTEGER OF
                   1:(Index           : LongInt;
                      TimePoint       : REAL   ;
                      AmplitudeType   : INTEGER;
                      Amplitude       : REAL   ;
                      SlopeMaximum    : REAL   ;
                      RiseTime        : REAL   ;
                      RecoveryTime    : REAL   );
                   2:(Min_Ampl_Crit   : REAL   ;
                      Min_Inter_Top   : REAL   ;
                      Min_Slope_Crit  : REAL   ;
                      Percentage      : REAL   ;
                      Code            : CodeType);
                   3:(TimeConstant    : REAL   ;
                      SampleRate      : REAL   ;
                      FilterDelay     : REAL   ;
                      Frequency_low   : REAL   ;
                      Frequency_high  : REAL   );
                   END;
```

```
VAR   ResultFile : FILE of ResultType;
```

5 Modifying CODAS Data Files

The COSAM.EXE routine allows you to reduce the original sample rate of a CODAS[8] data file in order to reduce the amount of data, to extract parts from the original file, and, to join different parts to a new CODAS data file. By typing "*COSAM /?*" you will get a command help screen.

```
CODAS File Operating Utility  Version 1.04  01/07/1993
Syntax:  COSAM <arg> [<inp>] [<out>] [<joi>]

 <arg>   arguments:
            /r    interactive data reduction to a smaller sample rate
            /e    extract parts
            /j    join with second CODAS input file
            /i    information about CODAS input file
            /?    this help
 <inp>   CODAS input file
 <out>   CODAS output file
 <joi>   second CODAS input file for joining
```

If you are creating a new CODAS data file, i.e. using one of the arguments */r*, */e*, or */j*, then the calibration file *.CAL of the CODAS input file is copied.

8 The CODAS package (Computer-based Oscillograph and Data Acquisition System) is a software and hardware utility that turns a personal computer into a high performance data acquisition system (Dataq Instruments, Inc., 1988).

References

Boucsein, W. (1988). Elektrodermale Aktivität: Grundlagen, Methoden und Anwendungen. Berlin: Springer. (English translation 1992: Electrodermal Activity. New York: Plenum Press.)

CookIII, E.W. & Miller, G.A. (1992). Digital Filtering: Background and Tutorial for Psychophysiologists. Psychophysiology, 29, 350-367.

Dataq Instruments, Inc. (1988). CODAS User's Manual. Software Release Level 4. Manual Revision H. 825 Sweitzer Ave., Akron, Ohio 44311, USA.

van Dellen, H.J., Aasman, J., Mulder, L.J.M. & Mulder, G. (1985). Time Domain versus Frequency Domain Measures of Heart-Rate Variability. In Orlebeke, J.F., Mulder, G. & Doornen, L.J.P. van (Eds.), Psychophysiology of Cardiovascular Control, pp. 353-374. New York: Plenum Press.

Elbert, T. & Rockstroh, B. (1980). Some Remarks on the Development of a Standardized Time Constant. Psychophysiology, 17, 504-505.

Martin, I. & Venables, P.H. (1980). Techniques in Psychophysiology. Wiley: Chichester.

Thom, E. (1988). Die Hamburger EDA-Auswertung. In Boucsein, W. (Hrsg.), Elektrodermale Aktivität: Grundlagen, Methoden und Anwendungen, S. 501-514. Berlin: Springer.

Tietze, U. & Schenk, C. (1978). Halbleiter - Schaltungstechnik (4. Aufl.). Berlin: Springer.

Trosiener, H.-J. & Kayser, J. (1992). Analysis and Interactive Control of Electrodermal and Cardiac Activity: A Computer Program for IBM PC's and Compatibles. Poster presented at the 20th annual meeting of the DGPA, June 18-20, Hamburg.

Appendix

A. The Definition File ED.DEF

```
cod            ; type of Data(-File) : COD,BIN,ASC
ecg1           ; type of Analysis (eda, ecg1, ecg2, pow, int)
-5.0           ; lower border of signal range
 5.0           ; upper border of signal range
0              ; time constant in seconds, 0 = no time constant
bessel         ; filter type (for future versions)
notch          ; filter characteristic ( Low, High, Band,
                 Notch (Band rejection), none)
2              ; order of the filter (2,4,6,8,10)
48             ; upper cutoff-Frequency for LP/HP filter;
                 lower for Bandfilters/notch
52             ; upper cutoff-frequency for bandfilters and bandrejection
none           ; binary Output( tcor , filo, filc, orig, none)
.4             ; Threshold : maximum (smallest amplitude of a maximum)
0              ; Threshold : overlayed maximum
0              ; Threshold : speed
0              ; Threshold : acceleration
0              ; frequency of r-wave for enhancement / or time constant of
                 integrator (Only for methods ECG2 and INT )
0              ; segment length (DataPoints) in EMS/RAM/disk
                 (0 = automatic use )
```

B. The Definition File EC.DEF

```
; EC.DEF - parameter settings (definitions) of EC.EXE  (C) Kayser 01/07/1993
; ======   ==========================================

; EC.EXE searches for a file EC.DEF (another file name may be
; specified as the second call parameter when starting the program).

; This file is an example of a parameter setting file:

; - empty lines are ignored
; - comments start with a semicolon

; - the order of all parameters for the head settings is fixed
; - head settings occur only once and refer to all channels (the default
;   is coded by a minus sign)
; - sequenced settings may be left, i.e. it is possible to have no head
;   settings at all (default settings)

; - head settings ----------------------------------------------------
7              ; rate of compression (default is 1, not smaller than 1,
               ; whole number)
y              ; event related analysis (using time markings)
-              ; file name of time markings (- = *.ZMD)
5              ; channel number for time analysis
1.0            ; start of time window [sec] in relation to a time marker
5.0            ; end of time window [sec] in relation to a time marker
1.0            ; height of markings (a base of zero is supposed)
```

```
0              ; duration [sec] of the 1st rectangular impulse (0 = any time)
0.5            ; duration [sec] between both impulses (0 = any time)
1.0            ; duration [sec] of the 2nd rectangular impulse (0 = any time)
2              ; time marking in pattern (0 = rise edge first rectangular
               ; impulse, 1 = decay edge first rectangular impulse, 2 = rise
               ; edge second rectangular impulse, 3 = decay edge second
               ; rectangular impulse; if each duration equals zero an even
               ; number (0 or 2) will lead to a detection of all rise edges,
               ; an odd number (1 or 3) will detect all decay edges)

; - adjustment factor settings ------------------------------------
; AF02:0.01      ; adjustment factor channel 02 (data divided by 100)
; AF03:0.1       ; adjustment factor channel 03

; - scaling factor settings (default = auto scaling) are coded by the
;   initial characters "SF" followed by the channel number (two digits)
;   and a not significant character (e.g. double point), i.e. the
;   scaling factor value must be on column 6-15

; - scaling factor settings (range 1 to 9999) -----------------------
; SF01:100       ; scaling factor channel 01
; SF02:100       ; scaling factor channel 02
; SF03:200       ; scaling factor channel 03
; SF04:50        ; scaling factor channel 04


; - parameter settings of a particular channel (channel entries) are
;   introduced by 'CHANNEL[n]' ([n] = channel number with two digits,
;   e.g. '01', max. 8 channels are possible)
; - entries for a particular channel are optional; if a channel is
;   mentioned all entries for that channel have to be explicit
; - the default is coded by a minus sign

;-------------------------------------------------------------------
CHANNEL01      ; start entries of channel 1
y              ; y = show data of channel, n = ignore data of channel
SCR left       ; label (comment max. 12 characters)
-              ; binary file name (- = *.D01)
y              ; y = use binary data, n = do not use binary data
-              ; result file name (- = *.G01)
y              ; y = show results, n = do not show results
-              ; file name for corrected results (- = *.K01)
y              ; y = correct results, n = do not correct results
n              ; selection of results (y = all, n = none)

;-------------------------------------------------------------------
CHANNEL02      ; start entries of channel 2
y              ; y = show data of channel, n = ignore data of channel
SCR right      ; label (comment max. 12 characters)
-              ; binary file name (- = *.D01)
y              ; y = use binary data, n = do not use binary data
-              ; result file name (- = *.G01)
y              ; y = show results, n = do not show results
-              ; file name for corrected results (- = *.K01)
y              ; y = correct results, n = do not correct results
n              ; selection of results (y = all, n = none)

;-------------------------------------------------------------------
CHANNEL03      ; start entries of channel 3
```

```
y              ; y = show data of channel, n = ignore data of channel
EOG            ; label (comment max. 12 characters)
-              ; binary file name (- = *.D01)
n              ; y = use binary data, n = do not use binary data
-              ; result file name (- = *.G01)
n              ; y = show results, n = do not show results
-              ; file name for corrected results (- = *.K01)
y              ; y = correct results, n = do not correct results
-              ; selection of results (y = all, n = none)

;----------------------------------------------------------
CHANNEL04      ; start entries of channel 4
y              ; y = show data of channel, n = ignore data of channel
ECG            ; label (comment max. 12 characters)
-              ; binary file name (- = *.D01)
y              ; y = use binary data, n = do not use binary data
-              ; result file name (- = *.G01)
n              ; y = show results, n = do not show results
-              ; file name for corrected results (- = *.K01)
y              ; y = correct results, n = do not correct results
y              ; selection of results (y = all, n = none)

;----------------------------------------------------------
CHANNEL05      ; start entries of channel 5
y              ; y = show data of channel, n = ignore data of channel
markings       ; label (comment max. 12 characters)
-              ; binary file name (- = *.D01)
n              ; y = use binary data, n = do not use binary data
-              ; result file name (- = *.G01)
n              ; y = show results, n = do not show results
-              ; file name for corrected results (- = *.K01)
n              ; y = correct results, n = do not correct results
-              ; selection of results (y = all, n = none)

; the entry 'PRINTER[n]' initializes the printer ([n] = printer number
; with three digits, e.g. '001')
; the default setting is a standard printer
; up to now the following printer are supported:
;     001 - Epson MX 100
;     002 - Epson RX-80, Epson 80F/T
;     003 - Epson LQ-850/1050
;     004 - IBM 4207-002 and compatible IBM printers
;     005 - Hewlett_Packard LaserJet II, Hewlett_Packard DeskJet PLUS
;     006 - Panasonic KX-P1092

; - printer ----------------------------------------------------------
PRINTER004     ; initializes an IBM printer
```

C. An EDA Result File Listing Example

```
file name (binary)              DEMO.G01
file name (ASCII)               DEMO.EDA
threshold: maximum              0.0250
threshold: overlayed maximum    0.0250
threshold: speed (slope)        0.0000
percentage of maximum = onset     5.00
time constant in seconds          1.00         sample rate  200.00  Hz
```

```
delay in sample points         43.00    filter frequency    2.00  Hz
```

number	time [sec]	type	amplitude	slope maximum	rise time	recovery /2
1	1.295	1	0.441	0.668	1.235	1.080
2	4.360	2	1.331	1.762	1.350	1.373
3	7.070	1	3.675	4.910	1.255	1.105
4	9.835	2	2.254	3.630	1.120	3.249
5	10.990	1	0.324	0.698	0.710	0.325
6	16.790	2	6.364	7.326	1.610	20.091
7	25.845	2	4.584	5.843	1.490	2.174
8	27.465	1	1.311	2.776	0.775	0.500
9	36.645	1	0.054	0.080	1.050	0.580
10	39.675	1	2.043	2.753	2.100	1.265
11	42.365	1	1.576	2.812	0.965	0.730
12	44.475	1	1.670	2.939	0.975	0.700
13	49.160	1	1.533	2.817	0.935	0.690
14	51.275	1	0.416	0.922	0.740	0.430
15	53.640	1	1.821	2.725	1.160	0.785
16	62.705	1	0.482	0.662	1.170	1.485
17	67.980	0	1.441	2.129	1.195	-1.000
18	68.970	1	1.635	2.649	0.985	0.675
19	75.510	1	2.498	2.822	1.520	1.005
20	77.680	1	0.981	2.015	0.825	0.650
21	81.895	1	0.279	0.519	0.945	0.670
22	86.350	1	0.154	0.234	1.135	0.800
23	91.480	1	1.860	2.485	1.690	1.220
24	94.125	1	1.381	2.307	1.060	0.865
25	97.720	1	2.274	3.096	1.355	1.020

Index

<+>	30
<->	30
<0>	32
<Alt-A>	30
<Alt-I>	32, 33
<Alt-V>	5, 26
<Alt-X>	24, 26
<Alt-Y>	26
<Backspace>	30
<Ctrl-End>	30
<Ctrl-Home>	30
<Ctrl-PgDn>	30
<Ctrl-PgUp>	30
<Ctrl-PrtScr>	30, 31
<Cursor down>	26, 30
<Cursor left>	31, 32
<Cursor right>	31, 32
<Cursor up>	26, 30
<Delete>	30, 31
<End>	30, 33
<Esc>	23, 26, 28, 30-33
<F1>	6, 23, 25, 29, 31
<F10>	28, 30
<F2>	29
<F3>	30
<F4>	30
<F5>	30
<F6>	30
<F7>	33
<F8>	33
<F9>	33
<Home>	30, 33
<Insert>	32, 33
<Shift-F1>	33
<Shift-F2>	33
<Shift-F3>	33
<Shift-Tab>	33
<Space>	32, 33
<Tab>	33
acceleration	8, 11, 13, 43
actual time	29
adjustment factor	30, 44
amplitude criterion	1, 15, 17
arbitrary scaling factor	29
ASCII	1, 2, 4-7, 10, 13, 23, 24, 28, 34-36, 38, 45
BAND	1, 9, 10, 13, 21, 43
Band-pass	9, 10
base value	25, 32-34
batch mode	6, 10, 13-15
Bessel filter	9, 11, 15
binary data	5, 10, 23-25, 44, 45
binary output	1, 8, 10, 11, 13, 43
BORLAND	40
bpm	39
calculations	2, 13, 28, 33, 34, 36, 38
calibration file	3, 4, 41
channel number	5, 8, 14, 24, 30, 43, 44
co-processor	2, 3
CODAS	1-4, 6, 8, 23-26, 41, 42
compression	8, 12, 15, 19, 22, 26, 27, 32, 43
correction mode	27
COSAM.EXE	2, 3, 41
cut-off frequency	10
data format	1, 4
data point	4, 26, 32
Dataq Instruments, Inc.	41, 42
datatype	6, 8
definition file	1, 13, 14, 23, 24, 26, 27, 30, 31, 43
demo files	3, 24
displacement	8, 11
dotted lines	27
duration	27, 43, 44
E.D.A.	i, 1, 3, 5
EC	2-5, 13, 23-35, 37, 40, 43
EC.DEF	3, 23, 24, 43
EC.EXE	2-5, 23, 24, 40, 43
ECG	1, 3, 6, 7, 18-21, 25, 33, 35, 45
ECG1	7, 13, 14, 18-20, 43
ECG2	7, 12, 13, 18, 20, 43
ED	1-6, 8, 11, 13, 14, 23, 25, 34, 35, 40, 43
ED.DEF	1, 3, 13, 14, 43
ED.EXE	1-5, 14, 23, 40
EDA	1, 3, 6-8, 11, 13, 15, 16, 15-17, 34, 35, 39, 42, 43, 45
edge	27, 44
electrocardiogram	18
electrodermal activity	1, 7, 15, 42
EMG	1, 3, 9, 21, 22, 21, 22
EMS	2, 14, 15, 43
entry screen	3, 5, 6, 8, 23-28, 31
event detection	i, 1, 5, 6, 8, 13, 21
events	1, 2, 5, 6, 11, 13, 23, 25-27, 34, 36, 39, 40
file name	4-6, 8, 14, 23-27, 29, 33-35, 37, 38, 43-45
file type	4
filter	1, 8-13, 15, 20, 21, 35, 43, 45
filter characteristic	9, 13, 43
filter order	10
filter type	9, 13, 43
frequency	4, 10-13, 17, 18, 24, 25, 31, 32, 35, 40, 42, 43, 45
graphic	2, 3, 15, 16, 19, 22-31, 33
graphic colors	16, 19, 22
group delay	9-11
HAMM	12
Hamming	12
HANN	12
Hanning	12
hardware requirements	2
heart period	1, 36, 37
heart rate	1, 27, 36-38
heart time	34
help	3, 6, 8, 23, 25-27, 29-32, 34, 41
HIGH	1, 9-13, 31, 32, 40, 41, 43
horizontal position	30
IBI	34, 36-40
IHR	38, 39
INT	7, 13, 43
integration	1, 3, 7, 21, 22

interactive mode 6, 10, 12, 14, 15, 18, 23
label 24, 25, 29, 33, 44, 45
LOW 1, 8-10, 13, 17, 31, 40, 43
manual scaling 12
maximum 7, 8, 11-13, 15-20, 25, 26, 29, 32-35, 43, 45
mean sum of squares of successive differences 38
memory 2, 14, 24
method 6-8, 11, 13, 14, 16-19, 21, 22
MSSD 38, 39
NOTCH 1, 9, 10, 13, 21, 43
on-line help 3, 6, 23, 26, 29
onset 1, 32-35, 45
open files 24, 26
parameter pre-definitions 13
Parzen 12
phasic changes 34, 37
POW 7, 13, 43
power spectrum 1, 3, 7, 11, 20, 21
precision 28
print file 30, 31
printer 24, 30, 31, 45
PSCREEN.EXE 2, 3, 31
R-wave 7, 12, 13, 18, 19, 32, 33, 35, 43
R-wave enhancement 12, 18
RAM 2, 3, 14, 43
range 4, 7, 8, 12, 13, 21, 26, 29, 30, 33, 36, 43, 44
range overflows 13
RECAL.EXE 2, 3, 28, 33, 34, 40
recovery time 16, 17, 20, 34, 35
RECT 12
rectangular window 12, 21
result file 1, 3, 5, 13, 25, 30-38, 40, 44, 45
RESULT.PAS 3, 40
rise time 17, 33-36
sample rate 2, 4, 8, 18, 21, 24, 25, 28, 34-36, 41, 45
scaling 12, 21, 29, 30, 44
scaling factor 29, 30, 44
SCR 1, 17, 18, 25, 34, 44
scrolling 30
segment 12, 14-16, 19, 21, 43
selection 11, 25, 31, 33, 44, 45
speed 2, 8, 11, 13, 35, 43, 45
STDEV 38, 39
system files 3
tc 9, 21, 22
threshold 8, 11, 13, 34, 35, 43, 45
time analysis 26-28, 33, 43
time axis 29
time constant 1-3, 5, 7-10, 13, 21-23, 35, 42, 43, 45
time file 3, 5, 26-28, 33-37
time markings 1, 2, 5, 7, 26, 28, 31, 33, 35, 43
time window 2, 27, 31, 33, 43
total time 25
Turbo Pascal 3, 40
VARCOEF 38, 39
variance coefficient 38
virtual memory 14
zero amplitude 32
zoom window 32