The *PlotMicroheart* program
Micro Heart Result Visualization

Lars Hoff

Version: January 23, 2007

Contents

List of Figures

List of Tables

1 Warning: Preliminary Report

This report describes work still in progress. It is not completely updated on all details and parameters. Hence, use it with great care.

2 Operation of the Program

2.1 Preferred method: From Graphical User Interface

The program is run by starting Matlab in the directory containing the 'PlotMicroheart' m-files. This directory contains a Matlab *startup*-file which will set up the correct path-structure and call the GUI file, *PlotMicroheartGUI.m*. This is intended to be the main way to run the program.

The input values to the program are explained Table 1. Each of these input fields correspond to a data field in variable *measdata*.

2.2 Alternative: From Batch File, *PlotMicroheartBatch*

Alternatively, the program can also be run from a batch file, *PlotMicroheartBatch*. The file *PlotMicroheartBatch* gives an example on how this can be done. The program sets the fields in *measdata* to the selected parameters, and calls *PlotMicroheart(measdata)* to calculate and plot the results.

A listing of the example file is printed in Figure 3. The example shows how a loop can be made to select different parameters and plot the results in different figures. First, the *measdata* variable is initialized with common parameters. Then, the different parameters *acceleration*, *velocity*, and *position* are calculated for the same parameters, and the results plotted in figures '1', '2', and '3'. The example illustrates variation of the physical quantity (acceleration, velocity, position) to be plotted. Any parameter can of course be varied by this procedure, typical examples are different time intervals for one recording, and different sensors for the same interval.

2.3 Main program: *PlotMicroheart*

Either *PlotMicroheartGUI.m* or *PlotMicroheartBatch.m* will, when called, put the parameters to be used into a *struct*-variable *measdata*. The contents of *measdata* is given in Table 3. The program will then call the main function *PlotMicroheart(measdata)*, using the parameter fields in *measdata* to load the selected raw data file, calculate the traces, and plot the results in the selected figure window.

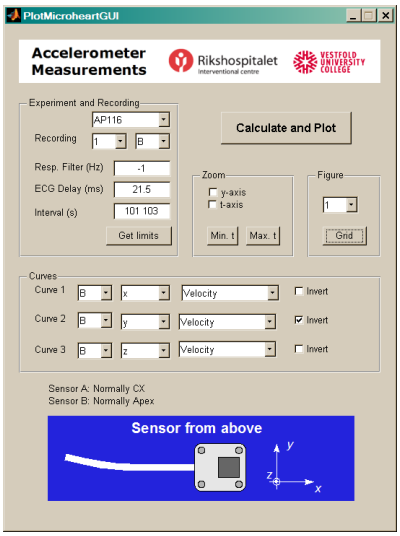

Figure 1. Graphical user interface to enter parameters and run the program *PlotMicroheart*.

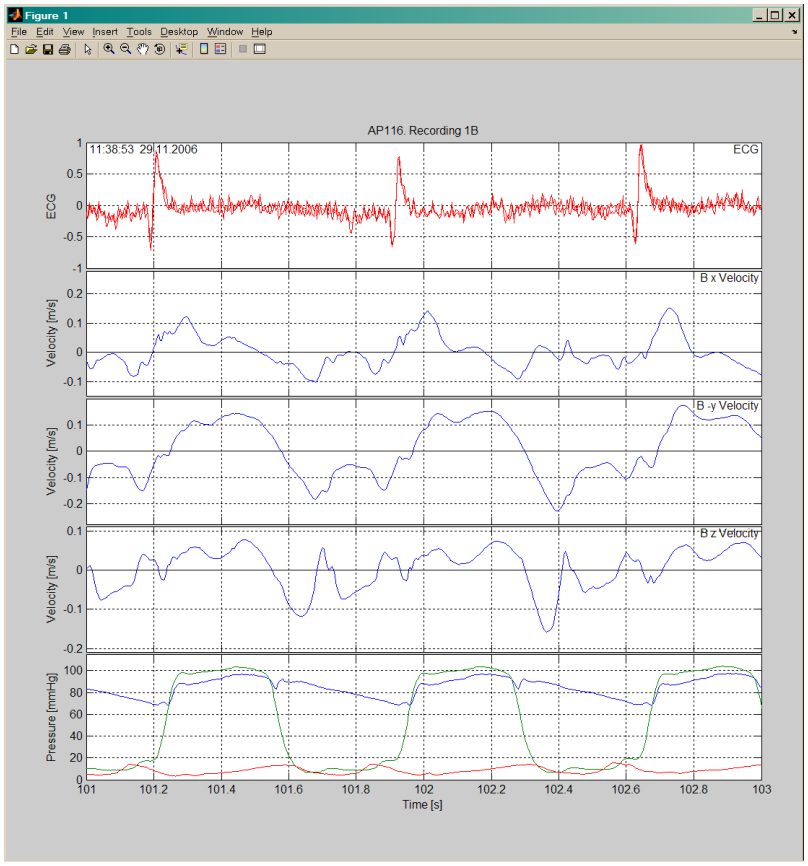

Figure 2. Example of result from *PlotMicroheart*, displaying the ECG signal (top), three velocity curves (middle), and the pressures (bottom).

Table 1. Operation of the program *PlotMicroheart*.

GUI Field	*measdata* field	Description
Experiment and Recording		
Experiment	AP	A valid experiment code, e.g.ÁP116', 'AH006'.
Recording	recording	A valid recording number for the selected experiment, e.g. '1B', '2', '5B' Older measurements do not include the letter.
Resp. filter [Hz]	respfilter	Cutoff-frequency of the respiration high-pass filter. Negative input gives a default value depending on experiment: Pig, 'AP': f_c= 1.0 Hz Human, 'AH': f_c= 0.6 Hz
ECG delay [ms]	ECGdelay	Delay in the ECG monitoring unit.
Interval [s]	interval	Interval to plot, start and end values
Curves		
Sensor	curve(k).sensor	Sensor 'A' or 'B', or no result, '0'.
Axis	curve(k).ax	'x', 'y', or 'z' axis, or 'Length'. Ref. sensor image in Figure 1.
Quantity	curve(k).type	What to plot: 'Acceleration', 'Velocity' or 'Position'.
Invert	curve(k).invert	Invert y-axis scale.
Calculate and Plot		Load data, calculate results, and plot graphs.
Zoom		
y-axis		Toggles y-axis zoom.
t-axis		Toggles x-axis, or time-axis.
Min. t		Set time-scale on all plots equal to the shortest interval.
Max. t		Set time-scale on all plots equal to the longest interval.
Figure		Number of figure where the result is plotted.
Grid		Toggle grid lines in all plots.

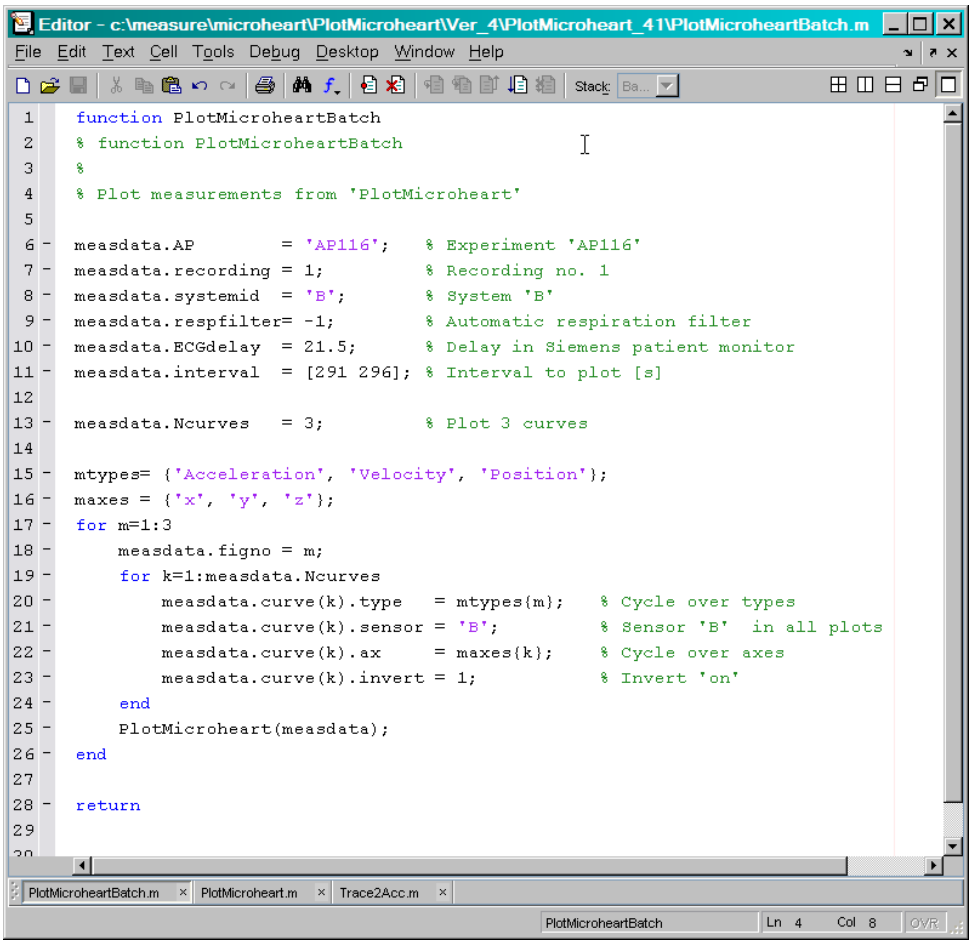

Figure 3. Example batch file *PlotMicroheartBatch* varying parameters to *PlotMicroheart* and plotting the results in different figures.

3 Organization of Functions

The structure of the program is illustrated in Table 2 and in Figure 4.

Table 2. Organization of functions used in *PlotMicroheart*.

Function	Description
Shell	
`varargout= PlotMicroheartGUI(varargin)`	Graphical user interface. Generated by Matlab's *Guide*, with variable initialization and callback functions added.
`PlotMicroheartBatch`	Batch operation.
Main function	
`[sample,wfmr,scaled]=PlotMicroheart(measdata)`	Load, calculate and plot results.
Sub functions	
`[sample,wfm,par]=InterpretTRC(sourcefile)`	Load and interpret raw data data files.
`scaled= Trace2Acc(sample,wfmr,ch)`	Scale results, from Volts to acceleration.
`PlotMHTraces(curve,ECG,sample,measdata,pressure)`	Plot calculated results.
Utility functions	
`trace= BuildTrace(t0,dt,y)`	Build waveform trace from components.
`h=PlotWaveform(trace,method,varargin)`	Plot waveform trace.
`y=MHintegrate(x,fs,fc,N)`	Integration with high pass filter.
`cal=ReadCalibration(sensor)`	Read sensor calibration data

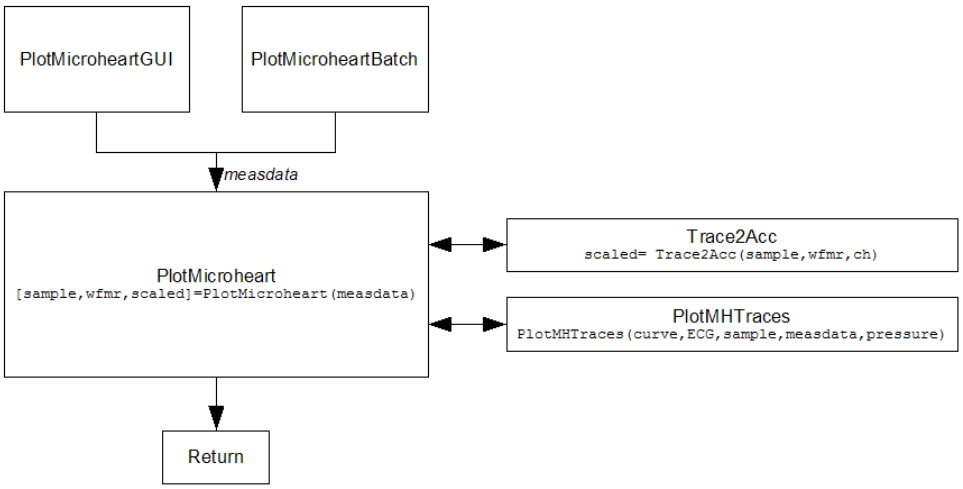

Figure 4. Illlustration of the organization of main functions in *PlotMicroheart*.

4 Organization of Data

Data are stored in the *struct* variables listed in Table 3.

Table 3. *Struct*-variables used in *PlotMicroheart*.

	Name	Description
Loaded from raw data file	sample	Data identifying the measurement. Stored when data is recorded.
	wfm	Acquired waveform data, results of the measurement.
	par	Free, for extra parameters read from 'filename.dat'.
Entered by operator, or from GUI	measdata	Data specifying which parameters to calculate.
Calculated	wfmr	Reduced version of wfm, containing. 1) Fields from wfm with only selected channels and intervals. 2) New fields needed for identification of data.
	scaled	Calculated and scaled results.

5 Contents of the *struct*-Variables.

Table 4. Fields of the *struct*-variable *measdata*.

Field	Size	Type	Description
AP		string	A valid experiment code, e.g.ÁP116', 'AH006'.
recording	1	double	Recording number.
systemid	1	char	System id letter: 'A', 'B', ...
figno	1	double	Number of figure where the result is plotted.
respfilter	1	double	f_c of respiration high-pass filter [Hz]. Negative value gives a default f_c: Pig, 'AP': f_c= 1.0 Hz Human, 'AH': f_c= 0.6 Hz
ECGdelay	1	double	Delay in the ECG monitoring unit [ms].
interval	2×1	double	Interval to plot, start and end values
Ncurves	1	double	No. of curves to plot.
curve	$1 \times Ncurves$	struct	Description of the curves to be plotted
Fields of *curve*			
curve(k).sensor	1	char	Sensor 'A' or 'B', or no result, '0'.
curve(k).ax	1	char	'x', 'y', or 'z' axis, or 'Length'.
curve(k).type		string	What to plot: 'Acceleration', 'Velocity' or 'Position'.
curve(k).invert	1	boolean	Invert y-axis scale.

Table 5. Fields of the *struct*-variable *sample*.

Field	Size	Type	Description
version	64	char	Acquisition program version number
sensor	2	cell	Sensor identifiers
operator	64	char	Operator initials
journalno	64	char	Reference to measurement, free text
id	64	char	Reference to test subject, free text
comment	64	char	Any comment, free text
code	64	char	Measurement code
no	1	double	Measurement number
date	64	char	Measurement date: *dd.mm.yyyy*
time	64	char	Measurement time: *hh.mm.ss*
filename	64	char	Name of original raw data file
path	64	char	Path where original raw data was stored
sensortype	1	double	Sensor type no. 0: 'MHP0xx', 1: 'MHP1xx'

Table 6. Fields of the *struct*-variable *wfm*.

Field	Size	Type	Description
description		char	Short description of contents
ver	M	double	Version number
N	1	double	Total no. of channels, including merged channels.
inputconfig	M	double	Configuration of ADC, from NI-DAQ
Lscalingcoeff	2	double	Length of scalingcoeff
scalingcoeff	Lscalingcoeff	double	Conversion of ADC data to volts, from NI-DAQmx. Offset and multiplier for each channel,
Lchannel	1	double	Length of channel
channel	Lchannel	double	ADC channel names, from NI-DAQmx.
samplemode	M	double	Code for 'Sample mode', from NI-DAQmx.
samplerate	1	double	Sample rate [S/s].
Lverstring	2	double	Length of verstring.
verstring	Lverstring	char	Recording program(s) version number.
scaleoffset	N	double	ADC scaling coefficients, from 'scalingcoeff'. Included for compatibility between NI-DAQ 7 and NI-DAQmx
scalemultiplier	N	double	ADC scaling coefficients, from 'scalingcoeff'. Included for compatibility between NI-DAQ 7 and NI-DAQmx
trace	$N \times Nt$	double	Measured traces. 12 bit data stored as 'uint16'.
tracename	N	cell	Trace names, codes giving channel contents.
axname	N	cell	Axis names,roughly identical to 'tracename'.
cmax		cell	Correlation coefficients, when merging data from different systems.

M is the number of measurement systems used to obtain the data.
$M = 1$: All data sampled on one system.
$M = 2$: Data merged from two systems. Typically, acceleration data is from the first system and pressure data from the second.

Table 7. Organization of channels in *wfm.trace*.

Row no.	Symbol	Alt. symbol	Description
1	V_{DD}	VDD	Supply voltage
2	ECG	ECG	ECG recorded with acceleration data
3	a_{Ax}	Ax	Acceleration, sensor A, axis x
4	a_{Ay}	Ay	Acceleration, sensor A, axis y
5	a_{Az}	Az	Acceleration, sensor A, axis z
6	a_{Bx}	Bx	Acceleration, sensor A, axis x
7	a_{By}	By	Acceleration, sensor A, axis y
8	a_{Bz}	Bz	Acceleration, sensor A, axis z
9	ECG_p	ECGp	ECG recorded with pressure data
10	p_1	p1	Pressure channel *1*, normally aorta
11	p_2	p2	Pressure channel *2*, normally left ventricle
12	p_3	p3	Pressure channel *1*, normally left atrium

Table 8. Fields of the *struct*-field *selection* in variable *wfmr*. This variable *wfmr* contains the same fields as *wfm*, but *trace* has been shortened to include only necessary sample points, and a new field *selection* contains time and sample infor for the selected part included in *trace*.

Field	Size	Type	Description
t0	1	double	[s] Start time of *trace*, relative to the *trace* in *wfm*.
dt	1	double	[s] Sample interval.
n0	1	double	Start index of *trace*, relative to the *trace* in *wfm*.
N	1	double	Number of points in *trace*.

Table 9. Fields of the *struct*-variable *scaled*.

Field	Size	Type	Description
bias	Na	double	Bias values, B_k, for calibration of acceleration.
sensitivity	Na	double	Sensitivity values, S_k, for calibrating acceleration.
Na	1	double	Number of acceleration channels.
excluded	$dynamic$	double	List of excluded sample points.
Vdd	1	waveform	Supply voltage [V].
ECG	1	waveform	ECG signal [V].
a	1	waveform	Scaled acceleration vectors [g].
p	1	waveform	Scaled pressure vectors [mmHg].

Table 10. Definition of the *waveform* variable.

Field	Size	Type	Description
Required t0	1	double	Start time.
dt	1	double	Sample interval.
y	$N \times M$	double	Data vectors.
Optional			
delay	1	double	Possible system delay.
name		$string$	Name of trace