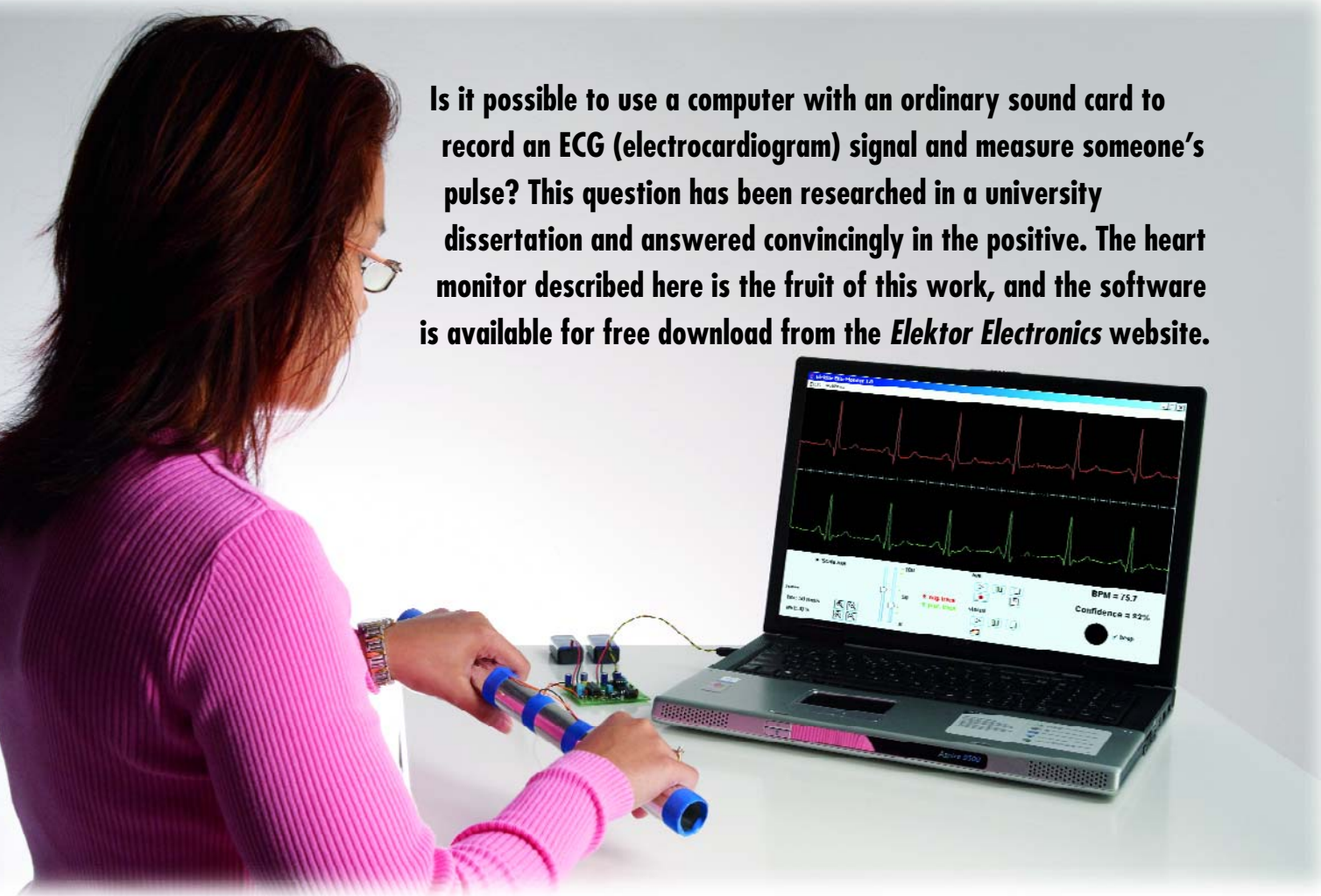


ECG using a Sound

Medical signal processing on a PC

Martin Klaper

Is it possible to use a computer with an ordinary sound card to record an ECG (electrocardiogram) signal and measure someone's pulse? This question has been researched in a university dissertation and answered convincingly in the positive. The heart monitor described here is the fruit of this work, and the software is available for free download from the *Elektor Electronics* website.



Elsewhere in this issue we describe how to record an ECG signal using a Gameboy games console fitted with a special insertion card. Here we show how it can be done in a more experimental way using an ordinary sound card or the audio input of a laptop computer. For this we need a

sensor (see **Figure 1**), which in the simplest case can be just a tube with contact surfaces at either end that can be gripped like the handlebars of a bicycle. The weak signal that is picked up is amplified by a factor of 1,000 and presented to the audio input of the computer.

Subsequent processing is carried out using a Java program. It conditions the signal using a digital filter, stores it and displays it on the screen. The program also monitors the pulse and automatically calculates the pulse rate with a digital display and audio output. The individual readings can be stored in a

d Card

Specifications

Input impedance:	$> 1 \text{ M}\Omega$
Input dynamic range:	5 mV_{pp}
Amplifier current consumption:	approximately 11 mA
Optocoupler current consumption:	approx. 2.2 mA
Common mode rejection ratio (CMRR):	$> 70 \text{ dB}$
Gain:	approx. 1,000 (60 dB)
Bandwidth:	approx. 0.4 Hz to 35 Hz (depending on sound card)
Recording rate:	in practice unlimited, typically 60 kbyte per minute

file for later analysis. Test data sets are also available on the Internet [1] from medical databases, and these can also be processed and displayed using the program.

ECG

As described in detail in the Gameboy ECG article elsewhere in this issue, the heart muscle is controlled by electric currents. This electrical activity can be measured on the surface of the body using electrodes. The resulting plot against time is called an electrocardiogram, or ECG. **Figure 2** shows a typical ECG plot: the exact form of the curve is an important diagnostic aid. We will leave the job of diagnosis to the specialists, and concentrate here on making the measurements.

ECGs are normally displayed with 25 millimetres on the horizontal axis representing one second, or 40 milliseconds per millimetre. The vertical axis is usually 10 millimetres per millivolt. This means that a 1 mm square on the plot represents 0.04 s in time and 0.1 mV in voltage. There are conventions for labelling certain characteristic points on the ECG curve with letters [2]. The distance from one of the prominent 'R' peaks to the next represents exactly the time between two heartbeats: this lets us readily determine the pulse rate.

This rate, expressed in beats per minute (or BPM) is displayed by the computer, and the pulse itself can optionally be output as an audio signal. An interesting project would be to compare the measured rate against a preset value and use this to control the braking on an exercise bicycle, in order to maintain a constant pulse rate while exercising.

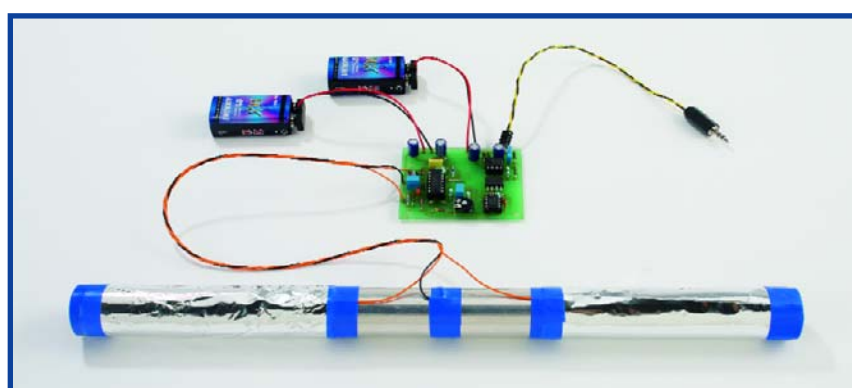


Figure 1. This simple sensor for heart signals consists of a pipe with a contact surface at either end.

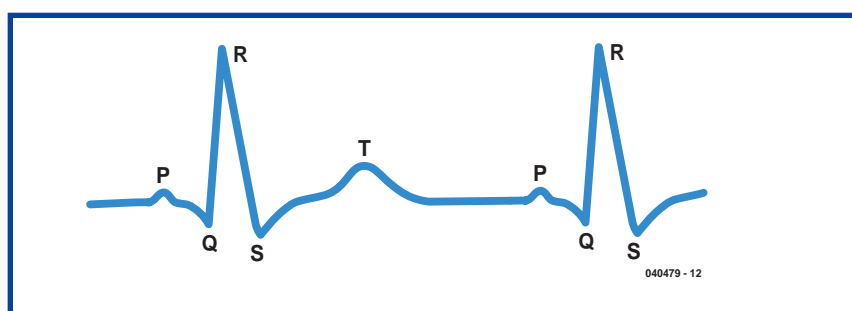


Figure 2. The interval between two consecutive points marked 'R' in the ECG trace gives the time between heartbeats.

Safety

The circuit and software are not approved and/or licensed for medical use. They are for private use only, for example for experimental purposes.

The circuit must only be used with a battery supply. This also applies to the connected computer whose sound card input is being used: the computer **must not be connected to the mains supply**. In practice this means that a notebook or laptop PC must be used, running on battery power. The mains adaptor must be disconnected!

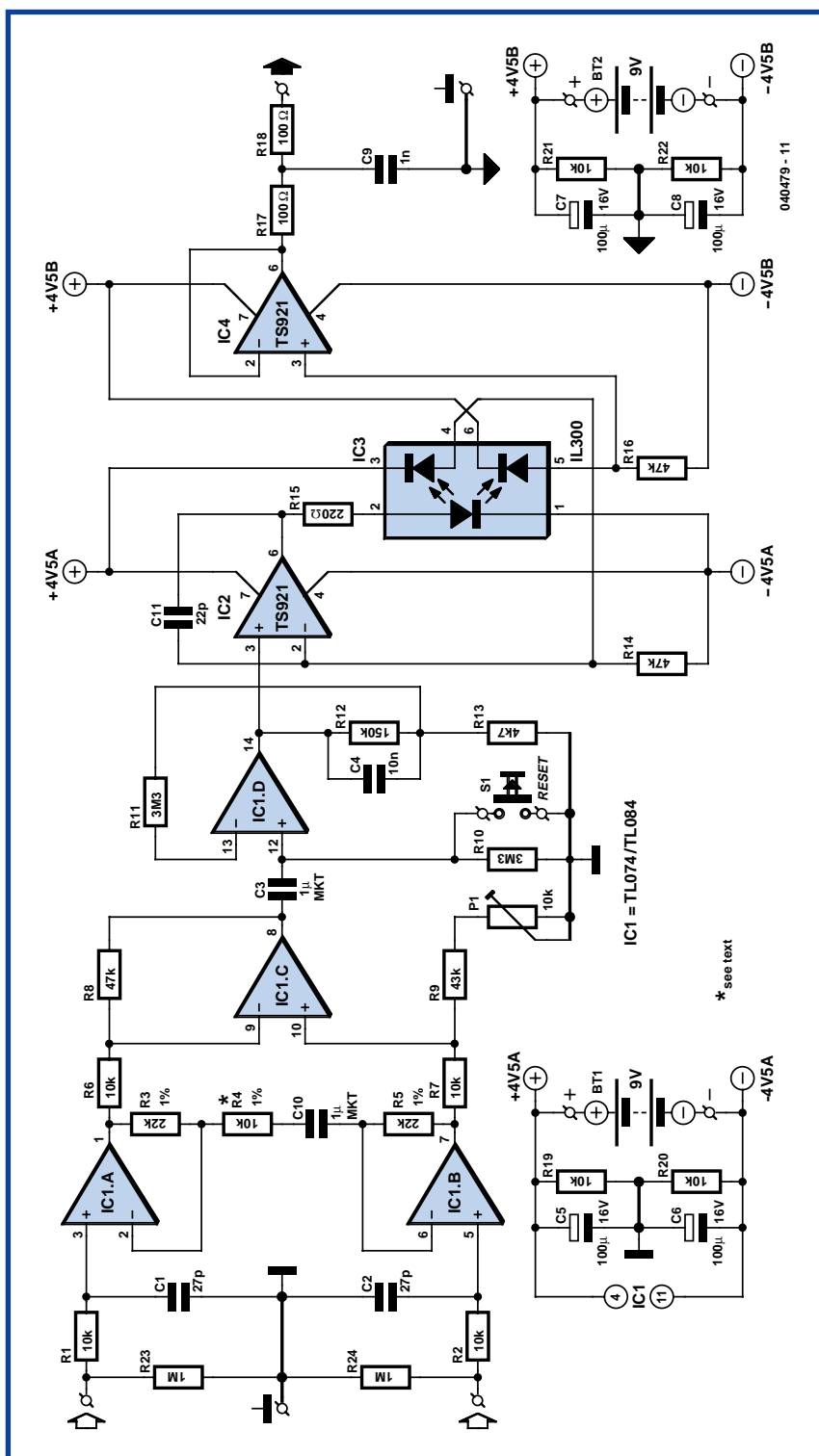


Figure 3. Circuit of the instrumentation amplifier, with galvanical isolation between input and output provided by an optocoupler.

Signal processing

Measuring an ECG using a computer requires demanding real-time processing, most of which is carried out in software. The hardware takes the form of an instrumentation amplifier

(Figure 3) and has the job of amplifying the weak signal from the sensor (which has an amplitude of approximately 1 mV) by a factor of around 1,000, and attenuating DC, common-mode, and high-frequency components.

To obtain an (at least relatively) clean ECG signal it is necessary carefully to filter out any interference. This is done in software using a biquad infinite impulse response filter. The filter can be configured for any of the required responses: low-pass, high-pass, band-pass and notch. A 50-Hz rejection filter removes interference originating from the mains, and other interference is attenuated using a further high-pass filter. Since the signal is obtained from electrodes on the skin it is possible that there will be a slowly-varying offset voltage: this is removed using a DC blocking filter. The main pulse of the ECG can be extracted using a band-pass filter, giving a signal from which it is straightforward to measure the pulse rate. The Java program allows display of either the original signal or the filtered version. The various filter functions can be selected and configured by the user, and the effect on the processed signal can be clearly seen. The pulse rate is calculated from its autocorrelation function, determining the period by comparing the signal against itself with a time offset.

Instrumentation amplifier

The circuit (Figure 3) can be divided into two blocks: the instrumentation amplifier itself at the input and the optocoupled isolation amplifier at the output.

The signal is amplified by quad operational amplifier IC1, type TL084 (or the lower-noise TL074). IC1.A and IC1.B are non-inverting amplifiers, feeding the inputs of differential amplifier IC1.C. This arrangement is known as an 'instrumentation amplifier'. P1 allows adjustment to obtain best common-mode rejection. Coupling capacitor C3 at the input to the next stage, built around IC1.D, blocks the DC component of the output of the instrumentation amplifier. To minimise the effect on low-frequency signals the time constant of the RC network formed by C3 and R10 is more than three seconds. This means that it will take at least this long for the voltage across the capacitor to stabilise when power is applied: this delay can be avoided by pressing reset button S1.

Optocoupler IC3 is driven by IC2. The type TS921 used is a 'rail to rail' opamp, which means that it can be driven to either extreme of its supply voltage range. Its output can deliver

COMPONENTS LIST

Resistors

R1,R2,R4,R6,R7,R19-22 = 10k Ω
 R3,R5 = 22k Ω 1%
 R8 = 47k Ω
 R9 = 42k Ω
 R10,R11 = 3M Ω 3
 R12 = 150k Ω
 R13 = 4k Ω 7
 R14,R16 = 47k Ω
 R15 = 220 Ω
 R17,R18 = 100 Ω
 R23, R24 = 1M Ω
 P1 = 10k Ω preset

Capacitors

C1,C2 = 27pF
 C3,C10 = 1 μ F 63V, 5mm lead pitch
 (no electrolytic cap)
 C4 = 100nF
 C5-C8 = 100 μ F 16V radial
 C9 = 1nF
 C11 = 22pF

Semiconductors

IC1 = TL074 DIP14
 IC2,IC4 = TS921 or TL071 DIP8
 IC3 = IL300

Miscellaneous

Two wire links
 Two 9-V PP3 batteries with connecting clips
 Two DIL8 IC sockets
 One DIL14 IC socket
 PCB, order code **040479-1**
 CD-ROM, contains PC software and source code, order code **040479-81** or free download from www.elektor.com

up to 80 mA, although only approximately 2.2 mA is needed to drive the transmit LED in the optocoupler. The current through the transmit LED is controlled using feedback from one of the receiver diodes in the optocoupler fed in to the inverting input of the operational amplifier. The result is that the voltage across R16 (the bias resistor for the second receiver diode) is equal to that across R14 and hence to the voltage at the non-inverting input to IC2. In other words, the voltage at the output of IC1.D appears across R16, but with galvanic isolation. C11 prevents high-frequency oscillation of the driver.

The final opamp at the output of the circuit acts as a buffer amplifier. This makes for a low-impedance output, which is also short-circuit proof thanks to the 100 Ω resistors R17 and R18, forming a low-pass filter at the output in combination with C9.

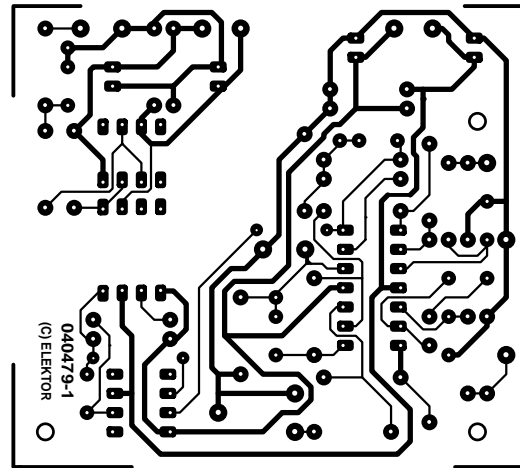
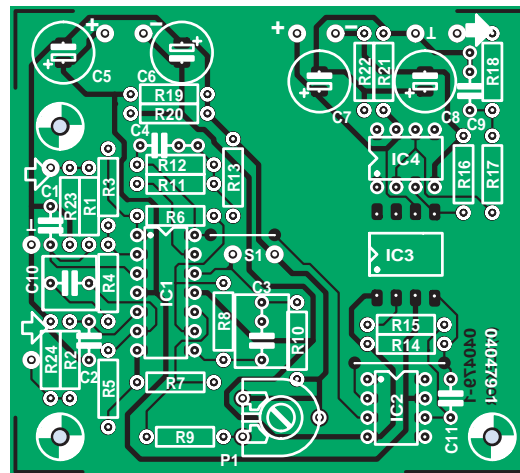


Figure 4. There are no SMDs on the single-sided printed circuit board, and so assembly is straightforward.

About the author

Martin Klaper studied electronic engineering at the Swiss Federal Institute of Technology in Zurich and worked in various positions in the telecommunications industry over a period of 20 years. Since 2000 he has been a lecturer in computing and telecommunications, until 2005 at the University of Applied Sciences of Northwestern Switzerland (FHSO), and since October 2005 at the School of Engineering and Architecture (HTA) in Horw, near Lucerne. This project was begun at FHSO and continued at HTA.



The author is also a keen radio amateur (callsign HB9ARK) and is currently interested in the ideas behind software defined radio. He is married with two sons.

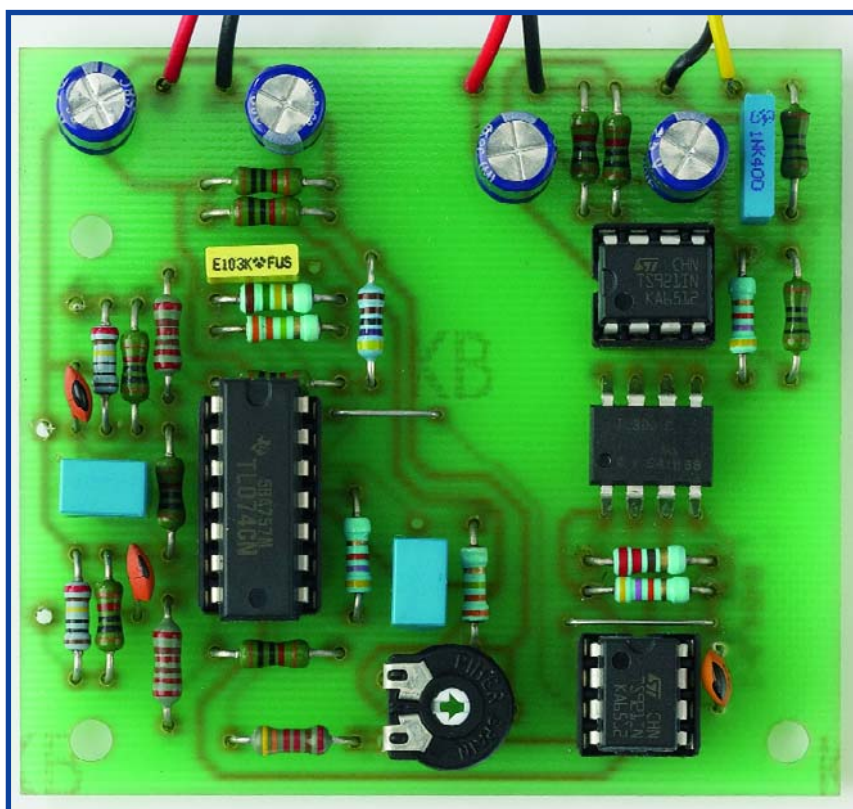


Figure 5. The populated *Elektor Electronics* prototype board. The IC not fitted in a socket is the optocoupler, whose pins are splayed out for mounting in order to increase the isolation gap.

The complete galvanic isolation between input and output of the circuit provides for extra safety (see the 'Safety' text box). The instrumentation amplifier and the output stage should be powered from separate batteries.

Components and construction

We look first at component selection. For resistors R3, R4 and R5 low-noise

metal film types are recommended. C10 provides DC decoupling for the input amplifiers, which, if the sensor shown in Figure 1 is used, prevents weak muscle signals from swamping the signal from the heart. If disposable self-adhesive ECG electrodes are used instead, the capacitor can be replaced by a wire link.

If the TS921 should prove hard to obtain, a type TL071 can be substi-

tuted at the cost of reducing the dynamic range of the circuit somewhat. The 43 k Ω resistor (value from the E24 series) can also be replaced by a different value, adjustment of P1 compensating for the difference.

Assembly of the printed circuit board (**Figure 4**) should commence with the wire links, since these are easy to overlook if left to a later stage of construction.

IC3 is supplied with its pins bent at a right angle. To fit the circuit board they should be bent apart (see **Figure 5**): this is necessary in order to guarantee the necessary isolation gap.

P1 can be adjusted for best common-mode rejection. Start with it set to its mid-position: the adjustment is not critical and so this will often be adequate. For best results connect the inputs of the instrumentation amplifier together and adjust P1 to minimise the amplitude at the output of the 50 Hz signal picked up by the circuit. This measurement can be done using the Java program (see **Figure 7**).

The heart signal sensor used by the author consists of two short pieces of conducting tube (zinc- or chrome-plated steel, for example from a vacuum cleaner head, or a length of water pipe) joined using a piece of insulating pipe. The *Elektor Electronics* prototype used a longer piece of conducting pipe with insulating tape wound around either end for a distance of about 10 cm. We then wrapped this insulating layer with aluminium foil to provide the two contact surfaces. Tinned copper wires wound around the two pieces of foil provide connection to the inputs of the instrumentation amplifier. This sensor construction, illustrated in Figure 1, has the advantage of not requiring any drilling or sawing. Screened audio cable can be used to wire the output of the circuit to a plug suitable for connecting in to an audio line input (or microphone input) of a computer. Most computers use a 3.5 mm stereo jack.

Software

The Windows-based software is written entirely in Java. The modular structure of the software (see **Figure 6**) is designed to provide the following functions:

- capture the signal from the sound card (labelled 'io');
- signal conditioning and filtering to

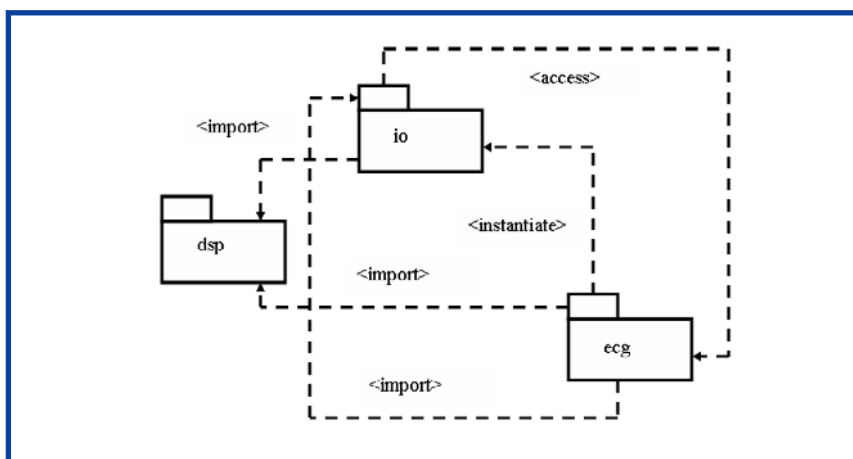


Figure 6. The module structure of the PC-based Java signal processing software.

- remove interference ('dsp');
- user interface ('ecg').

Figure 7 shows the appearance of the user interface, showing the main program window with the original signal plotted at the top and the processed signal plotted below. The time axis is shown in between the two plots. The BPM meter in the lower right corner of the window shows the pulse rate and the program's confidence in the displayed value. The audio output can also be enabled here. The program has many other features which we do not have space to describe here.

The source code is available for download from the *Elektor Electronics* website. There are several files in the download. The PC program is called 'EKG-MonitorV1.0.jar' and can be run with a double click of the mouse. The '.jar' extension indicates that this is a Java executable, analogous to a '.exe' file under Windows. A Java runtime system is required to execute the program: this can be found at [3]. You will need the version of the JRE (Java Runtime Environment) suitable for your operating system.

In practice

The circuit is suitable for use only with sound cards that have a frequency response that extends down to at least 0.1 Hz (at -3 dB). This should not present a problem if the input is DC coupled; however, many are provided with inputs that are AC coupled (i.e., with an input capacitor). In our laboratory the prototype worked perfectly with the audio input of an elderly laptop. In case of doubt the frequency response of the sound card can be measured using the free program RMAA [6].

Best results are obtained using commercial disposable ECG electrodes, but they can be tricky to attach to the skin. The pipe sensor described above and shown in Figure 1 is rather easier to use.

The filter (**Figure 8**) can be configured to minimise interference, and there is plenty of scope for experimentation, improvement, and research. Readers are encouraged to publish their improvements, and the Java software for the project is released under the GNU public licence.

As well as compiled hex code and source code, a User Guide is also available for download from the *Elektor Electronics* website. Look for archive file 040479-11.zip under month of publication.

(040479)



Figure 7. The main program window shows the original signal (above) and the processed signal (below).

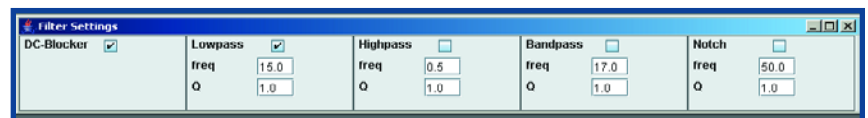


Figure 8. The 'Filter Settings' window allows configuration of the infinite impulse response biquad filter, implemented in software.

Links:

- [1] www.physionet.org/physiobank/ ECG sample data
- [2] <http://en.wikipedia.org/wiki/Electrocardiogram>
- [3] <http://java.sun.com/javase/downloads/index.jsp>
Java compiler and development environment. The Java Runtime Environment (JRE), current version 5.0, is required to run the program and the J2SE Development Kit (JDK), current version 5.0, is required to compile modified versions of the program.
- [4] www.bluej.org/download/download.html
Simple Java development environment with tutorial, ideal for beginners.
- [5] www.eclipse.org/downloads/
Eclipse is a full-featured Java development environment for professionals.
- [6] <http://audio.rightmark.org/download.shtml>
RMAA (RightMark Audio Analyser).
- [7] www.dspguru.com; www.musicdsp.org/archive.php?classid=0
Various topics in DSP.