

Thema, Ziele: Digitale Filterung, Integer vs. Floating Point

Aufgabe: Filterung von EKG-Daten, Darstellung mit gnuplot

(EKG-Daten und Filterung von Prof. F. Cellier, ETH Zürich)

Die im Verzeichnis Vorgabe beiliegende Datei `ecg.dat` enthält verrauschte EKG-Daten. Studieren Sie zunächst diese Datei in einem Texteditor. Die erste Zeile ist eine Kommentarzeile, dann folgen die Daten. Die Messungen wurden mit einer Frequenz von 1000 Hz durchgeführt. Jede Zeile entspricht einer Messung des EKG-Signals in mV. Die 10'000 Messungen entsprechen demnach einer Messdauer von 10 Sekunden.

Zur Visualisierung der Daten verwenden wir das Programm `gnuplot`. `gnuplot` ist erhältlich für Linux, Mac und Windows. Weitere Informationen zu `gnuplot` erhalten Sie unter:

<http://www.gnuplot.info/docs/gnuplot.html>

Nachdem Sie `gnuplot` im Terminal durch den Befehl `gnuplot` gestartet haben, müssen Sie unter Umständen noch das Ausgabeformat und die Zielfeile angeben.

Um das Ausgabeformat auf JPEG und die Zielfeile `plot.jpg` zu definieren, geben Sie folgende Befehle ein:

```
set terminal jpeg  
set output 'plot.jpg'
```

Statt `jpeg` können Sie auch z.B. `gif`, `postscript`, `png` und viele mehr wählen.

Geben Sie das folgende Kommando ein, um die Daten zu plotten:

```
plot 'ecg.dat' with lines
```

Sie können `gnuplot` mit dem Kommando `quit` wieder schliessen.

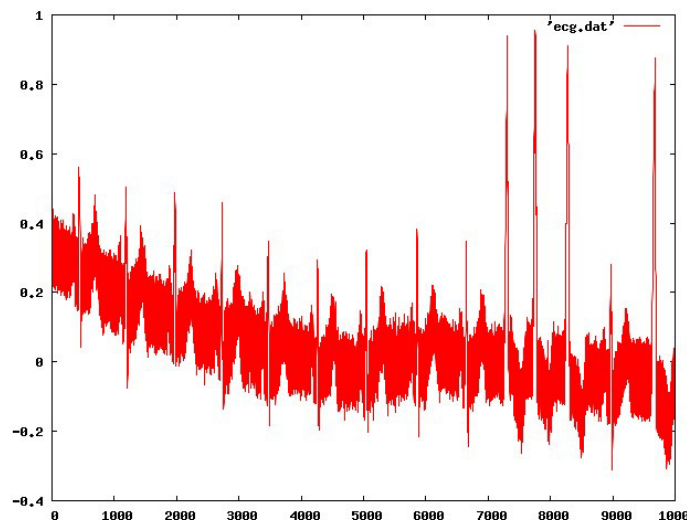


Abbildung 1: Verrauschtes EKG

Wie Sie sehen, ist das Signal stark verrauscht. Dieses Rauschen lässt sich in drei Hauptkomponenten aufteilen:

- Nulllinienschwankung (tieffrequentes Rauschen)
- feines Rauschen (hochfrequent)
- 50 Hz-Rauschen mit hoher Amplitude

Um das eigentliche EKG-Signal erkennen zu können, müssen als erstes diese störenden Frequenzen aus dem EKG-Messsignal herausgefiltert werden. Zu diesem Zweck finden Sie vorgefertigte Filter in den Dateien `filter1.dat`, `filter2.dat` und `filter3.dat`, mit denen sich das EKG-Signal aufbereiten lässt.

Die Filter sind im gleichen Format wie die EKG-Daten und lassen sich ebenfalls mit gnuplot betrachten. Die Filterung der Daten d mit dem Filter f der Breite l_f entspricht mathematisch einer diskreten Faltung:

$$d_{neu}[n] = \sum_{k=0}^{l_f-1} f[k] \cdot d[n - k + l_f/2]$$

An den Rändern greift diese Formel auf Werte ausserhalb der eigentlichen Daten zu (z.B. auf $d[-1]$). Deswegen müssen die Daten an den Rändern mit Nullen aufgefüllt werden.

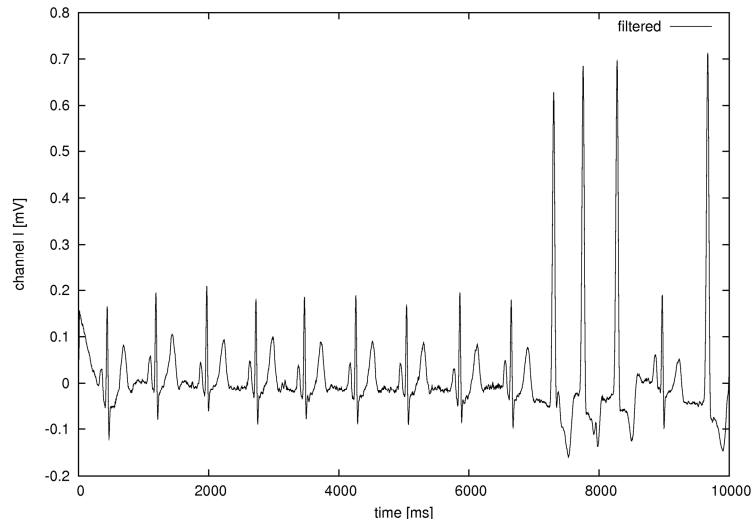


Abbildung 2: Gefiltertes EKG (alle Filter)

- a) Erstellen Sie ein Programm `filter`, das die EKG-Daten und einen Filter aus Dateien einliest und eine Faltung der Daten mit diesem Filter durchführt. Anschliessend sollen die gefilterten Daten wieder in eine Datei ausgegeben werden, so dass sie mit gnuplot betrachtet werden können.

Aufrufbeispiel:

```
filter filter1.dat ecg.dat output.dat
```

Hinweise:

- Verwenden Sie `filestreams` (Bibliothek `<fstream>`) zur Ein- und Ausgabe von Dateien:
`ifstream fin("dateiname");`
öffnet eine Datei zum Lesen und
`ofstream fout("dateiname");`
öffnet eine Datei zur Ausgabe und löscht den vorherigen Inhalt.
Die Variablen `fin` und `fout` verhalten sich dann genau wie `cin` und `cout`.
- Beachten Sie die Vorlagedatei `filterTMPL.cpp`, die Ihnen einen Teil der Aufgabe bereits abnimmt. Der Code bestimmt die Länge einer Filterdatei. So können Sie die Buffergrösse berechnen. Erzeugen Sie vorerst aus dieser Vorlage das Programm `filter.exe` und ermitteln Sie die Anzahl der Koeffizienten, beispielsweise von `filter2.dat` mittels des Befehls
`filter filter2.dat`
- In dieser Vorlagedatei sehen Sie auch die Übergabe von Parametern an `main()` mittels `argc` und `argv`.
- Die Headerdatei `fpfilter.h` beinhaltet die Funktionsprototypen der Funktionen, die Sie implementieren müssen.

- b) Verwenden Sie ihr Programm, um die Wirkungsweise der einzelnen Filter zu testen. Starten Sie das Programm in der Console, nicht aus Eclipse. Wenden Sie alle drei Filter auf `ecg.dat` an. Beantworten Sie zudem die Frage, welche Filterdatei welcher Funktion entspricht.

Funktion	Filterdatei
Hochpass	
Tiefpass	
50 Hz Notch	

- c) Die EKG-Daten liegen als Floating Points vor. Untersuchen Sie, welche Operationen für die Berechnung der Faltung notwendig sind und welche Hardware für die Berechnung am besten geeignet wäre. Benötigen Sie Floating Point- oder Integeroperationen?
- d) Sie führen Ihr Programm auf einem Controller mit folgenden Eigenschaften aus

Operation	Zyklen
FP Addition	8
FP Subtraktion	8
FP Multiplikation	12
FP Division	24
Integer-Addition (32 Bit)	1
Integer-Subtraktion	1
Integer-Multiplikation	2
Integer-Division	10

Wie viele Zyklen dauert die Berechnung mit Filter 2, wenn Sie mit Floating Points rechnen, wie viele würde es dauern, wenn Sie mit Integern rechnen könnten?

- e) Implementieren Sie die Faltung mit geeigneter Skalierung nur mit ganzen Zahlen. Schätzen Sie ab, wie gross der Genauigkeitsverlust ist. Beurteilen Sie durch Betrachten der beiden Kurven, ob Sie einen qualitativen Unterschied zwischen den beiden Implementationen feststellen.