

EKG-Experimente

Ich fand es schon lange reizvoll, über eigene EKG-Systeme nachzudenken. Speziell interessieren mich Lösungen, die preiswert und einfach aufzubauen sind. Dieser Artikel zeigt einige Möglichkeiten, sich mit dieser Technik zu beschäftigen. Doch zunächst etwas Grundsätzliches:

Durch den direkten Kontakt der EKG-Elektroden mit dem Körper sind alle Komponenten ausschließlich mit Akkus oder Batterien zu betreiben. Kein Teil, also PC, Oszillograph, USB-Verbindungen dürfen eine Netzverbindung haben. Der PC darf nur mit Batterie betrieben werden (Laptop), alle Schaltungen dürfen nie mit Netzteil versorgt werden. Immer muss bedacht werden, dass Netzteile durchaus mal schadhaft sein können und dadurch die Gefahr besteht, einen Stromschlag zu bekommen

Wir werden im Folgenden einige Experimente finden, die uns näher an die EKG-Technik heranführen. Dabei sollten diese Experimente nie zu Diagnosezwecken benutzt werden, sondern lediglich interessante Bastelprojekte sein.

EKG-Simulator

Bevor wir uns selbst an ein EKG-Gerät anschließen, werden hier Experimente beschrieben, mit denen man ein „künstliches“ EKG erzeugen kann. Dies kann hilfreich sein, um die externen Komponenten, wie z.B. Oszillograph oder die serielle Übergabe der EKG-Daten zu testen.

Ausgangsbasis ist ein kleiner ATtiny13, der gespeicherte EKG-Daten entweder als PWM und damit analoge Werte oder als serielle Werte an eine serielle Schnittstelle ausgibt. Erstere Variante kann mit einem Oszillographen, die zweite mit einem Terminalprogramm zum PC und späterer Darstellung z.B. in Excel genutzt werden.

EKG-Simulator mit analoger Ausgabe

Der ATtiny 13 mit dem Programm „ekg_sim_pwm01.bas“ sendet autonom analoge EKG-Signale z.B. direkt an einen Oszillographen. Da hier kein Körperkontakt besteht, kann ein beliebiger Oszillograph auch mit Netzanschluss benutzt werden. Die folgende einfache Schaltung realisiert dies:

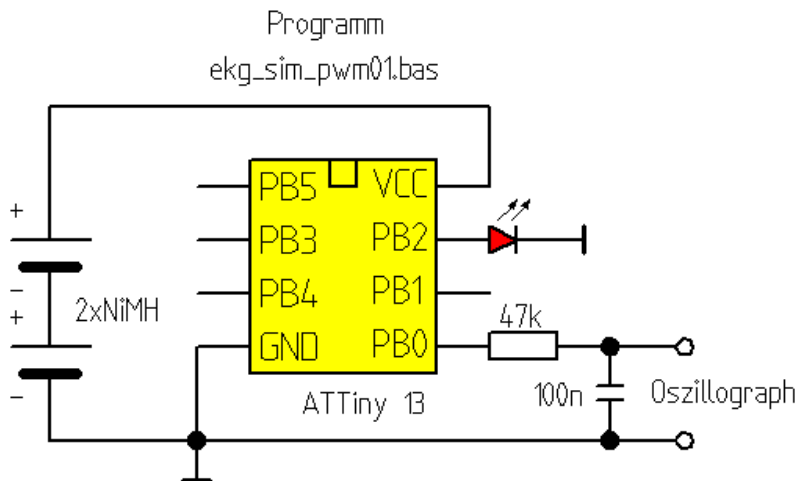


Bild Schaltung EKG-Simulator mit analoger Ausgabe

EKG-Simulator mit serieller Ausgabe

Will man simulierte EKG-Daten auf einem PC, z.B. über EXCEL darstellen, können die Daten über eine serielle Schnittstelle übertragen werden. Dazu wird wieder ein ATTiny 13 mit intern gespeicherten EKG-Werten verwendet. Diesmal überträgt er diese aber als seriellen Datenstrom. Das Programm „ekg_sim_ser01“ realisiert diese Funktion. Die Schaltung sieht dann so aus

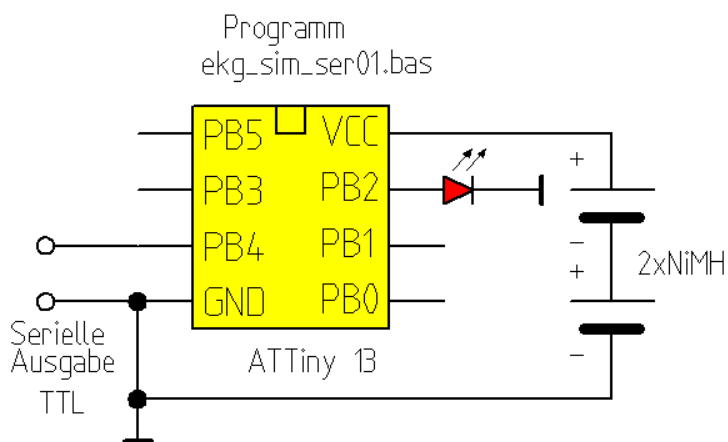


Bild Schaltung EKG-Simulator mit serieller Ausgabe

Für die Auswertung der seriellen Daten ist zunächst eine Wandlung der hier vorliegenden TTL-Daten notwendig. Ich benutze dazu den Pololu USB AVR Programmer V2.1., der neben der zum Programmieren der ATMEGA/ATTiny-Prozessoren notwendigen Schnittstelle eine zusätzliche RS232-Schnittstelle über USB bereitstellt:

<https://www.amazon.de/s?k=Pololu+USB-AVR+Programmer+V2%2F&camp=1638&creative=6742&linkCode=ur2&tag=mikroelektronik-21>

Damit ist eine Übertragung serieller Daten im TTL-Format an eine virtuelle (d.h. über USB) COM-Schnittstelle am PC einfach möglich. Ich benutze am PC das Programm hterm. Der Pololu-

Programmer liefert beim Verbinden mit dem PC zwei neue COM-Ports, die man über den Geräte-Manager findet:

Pololu USB AVR Programmer v2.1 Programming Port (COM17)

Pololu USMAVR Programmer v2.1 TTL Serial Port (COM16)

Der letztere ist der Kommunikationskanal für serielle Datenübertragung. Im Programm hterm werden im Bereich Input (Oberer Teil) folgende Einträge vorgenommen:

Port „COM16“ (bei mir, bitte im Geräte-Manager nachschauen)

Baud „9600“, Data „8“, Stop „1“, Parity „None“

Häkchen bei Ascii, Newline at „CR+LF“

Bei Save Output die Option „RAW“ auswählen

Ist der ATtiny13 programmiert, sollten nach „Connect“ sofort Werte herunter scrollen. Nach einigen Sekunden kann der Datenstrom mit „Disconnect“ gestoppt und mit „Save Output“ gespeichert werden. Die Log-Datei findet man im Ordner „hterm“ als „output_2022-04-27_11-29-05.log“ – hier nur als Beispiel. Diese Datei kann man direkt in EXCEL einlesen, muss aber vorher „alle Dateien“ aktivieren, da dies keine typische EXCEL-Datei ist. Sofort nach dem Einlesen kann über:

Einfügen -> Linie(ndiagramm) ein Diagramm der Datei erstellt werden, was dann etwa so aussieht:

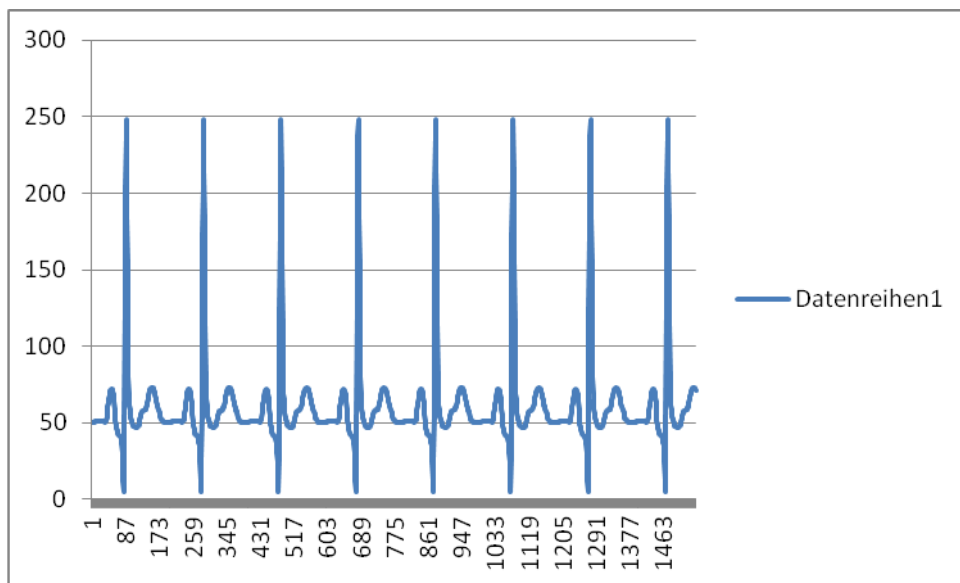


Bild EXCEL-Diagramm vom Simulator

Als Daten für das Diagramm können auch kleinere Bereiche ausgewählt werden, so dass z.B. nur zwei „Herzschläge“ erscheinen, was dann schon besser aussieht:

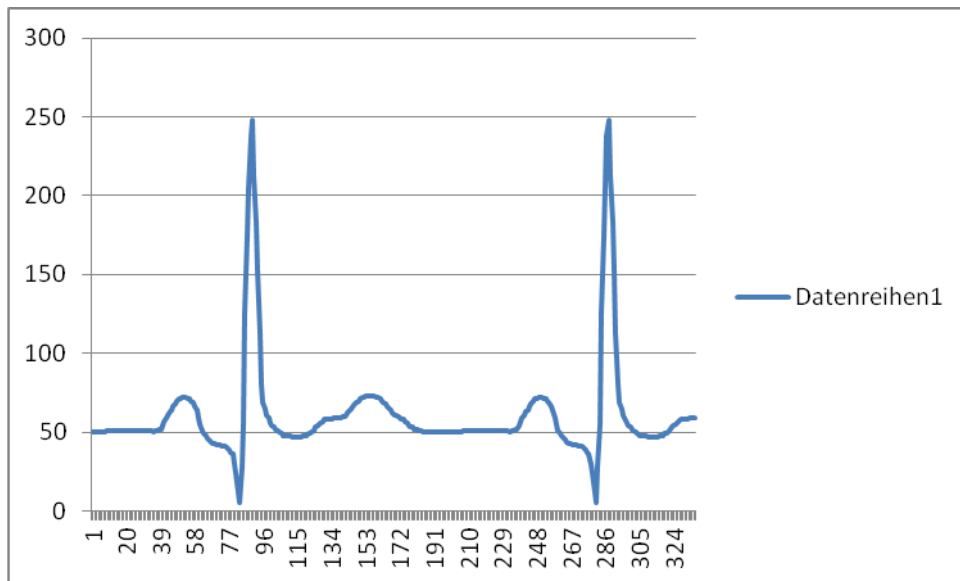


Bild Ausschnitt

Nun zum „echten“ EKG

Wie viele andere hat auch der Autor schon vor Jahren eigene EKG-Verstärker mit erheblichem Aufwand mit mehr oder weniger Erfolg entwickelt. Mittlerweile ist das alles viel einfacher geworden. So hat z.B. „Sparkfun“ ein einfaches Modul entwickelt, das sofort einsatzbereit ist und ein einfaches EKG ermöglicht. Mitgeliefert werden können die erforderlichen Kabel nebst selbstklebenden Körperkontakten. Auch ausführliche Anleitungen können bei Sparkfun eingesehen werden. Allerdings kosten die Dinge bei Sparkfun einiges (derzeit Modul 21,50€, Kabelsatz 5,50€, Zehnerpack Klebeelektroden 8,95€). Ich habe die gleichen Module bei Aliexpress sehr viel billiger bekommen, und zwar mit Kabelsatz für 3,99€:

https://de.aliexpress.com/item/33007711518.html?spm=a2g0o.order_detail.0.0.14036368S98Z2t&gatewayAdapt=glo2deu

Ob es sich hier um Clone handelt oder um Restbestände der Produktion für Sparkfun, kann ich nicht beurteilen – jedenfalls funktionierten alle genauso wie von Sparkfun angegeben. Wer also einige Wochen auf die Lieferung warten kann, könnte hier sehr viel sparen.

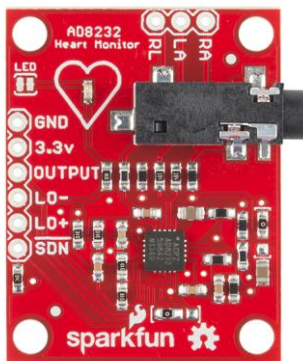


Bild SparkFun Single Lead Heart Rate Monitor - AD8232 (Foto Sparkfun)

Bevor es nun an erste Experimente gehen soll, noch ein wichtiger Hinweis. Die Module arbeiten mit nominal 3,3 V. maximal 3,6 V. Aus eigener Erfahrung weiß ich, dass höhere Spannungen „tödlich“ für die Module sind. Netzteile scheiden sowieso aus (s.o.), aber auch Lilonen-Akkus können gefährlich werden, da sie vollgeladen bis 4,2 V abgeben. Als brauchbar haben sich 2 NiMh-Akkus (max. 3V) oder 2 Alkaline-Zellen (max. 3 V) erwiesen. Das reicht für fast alle folgenden Experimente aus.

Erste Variante, Heart Rate Monitor - AD8232 mit Oszillograph

Hier ist nicht viel zu tun, die Klebe-Elektroden am Körper befestigen, den Stecker in die Buchse stecken und das Modul über die 6-polige Stiftleiste an 3,3 V und Masse anschließen. Danach OUTPUT und Masse mit einem Oszillographen verbinden. Die Anordnung der Elektroden zeigt das nächste Bild:

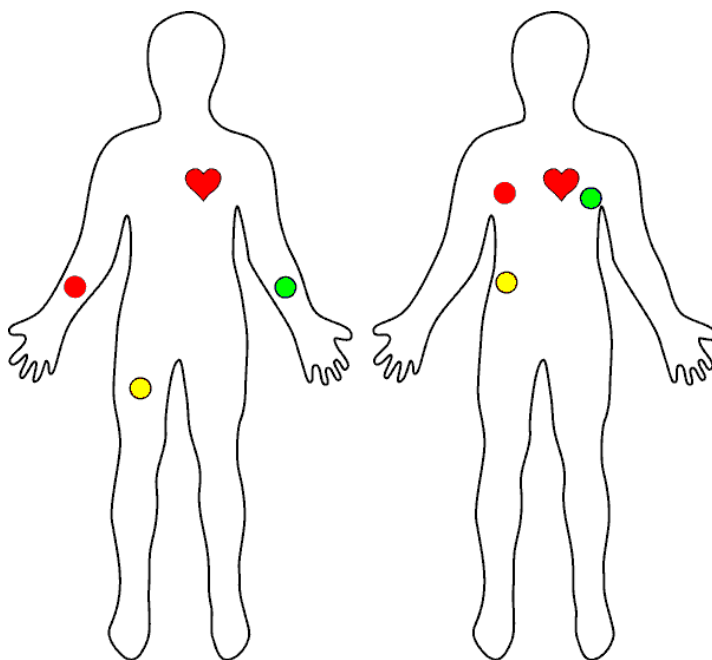


Bild Anordnung der Elektroden

Noch ein Wort zum Oszillographen. Ich benutze ein PicoScope 3204 mit der Software PicoScope 7 T&M Early Access. Dies ist ein USB-Oszillograph, der von der USB-Schnittstelle des Laptops mit Strom versorgt wird. Damit ist gesichert, dass beim Betreiben des Laptops über seine eigene Batterie keine Stromschlag-Gefahr ausgeht. Andere Oszillographen mit Netzanschluss sind gefährlich und dürfen nicht verwendet werden.

Am Oszillographen sollten folgende Einstellungen vorgenommen werden:

Zeitbasis	500ms/div
Kanal A	+/- 2V , Wechselspannung (bei Gleichspannung gibt es Auf- und Ab-Bewegungen der Kurven)

Wenn alles funktioniert, sieht das Ergebnis so aus:

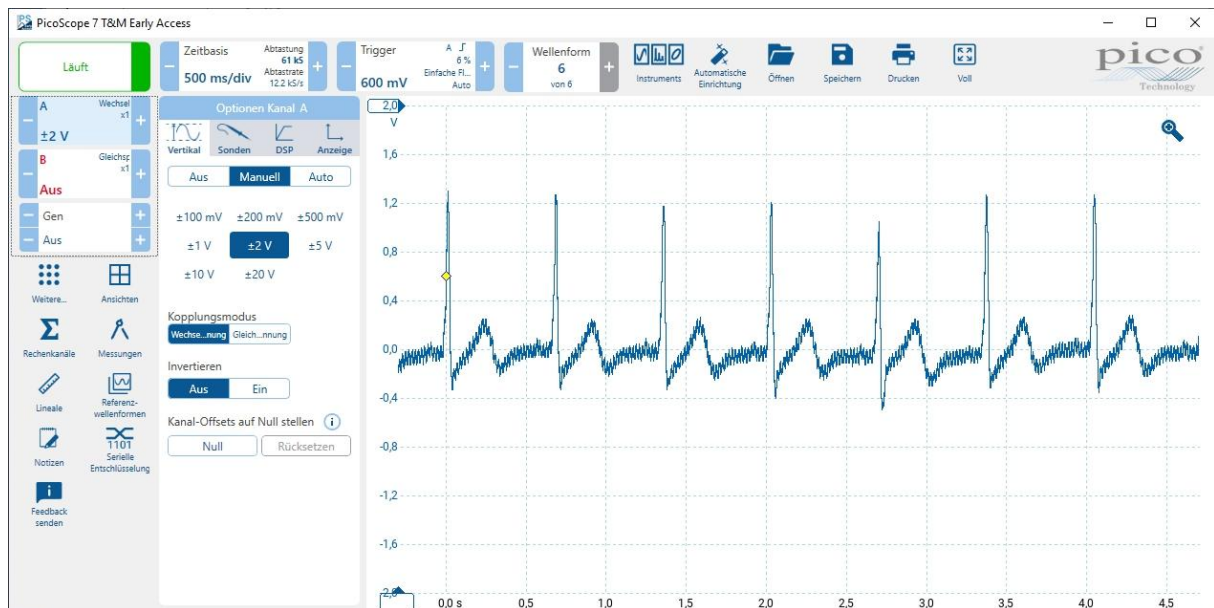


Bild EKG mit Heart Rate Monitor - AD8232 und PicoScope3204

Für den Fall, dass kein geeigneter Oszillograph verfügbar ist, gab es eine vergleichsweise günstige Alternative, Xprotolab Plain GT-0007 von Gabotronics. Dies ist eine kleine Leiterplatte mit einem 2-Kanal-Oszillographen, der über eine USB-Verbindung und eine passende Software Oszillogramme auf den PC und auf Androidgeräte bringt. Leider habe ich in der letzten Zeit keine Quelle (außer einer japanischen <https://www.elefine.jp/SHOP/XprotolabPlain.html>) für dieses Gerät mehr gefunden. Allerdings bietet Gabotronics auch andere Oszillographen an, teilweise sogar mit Display.

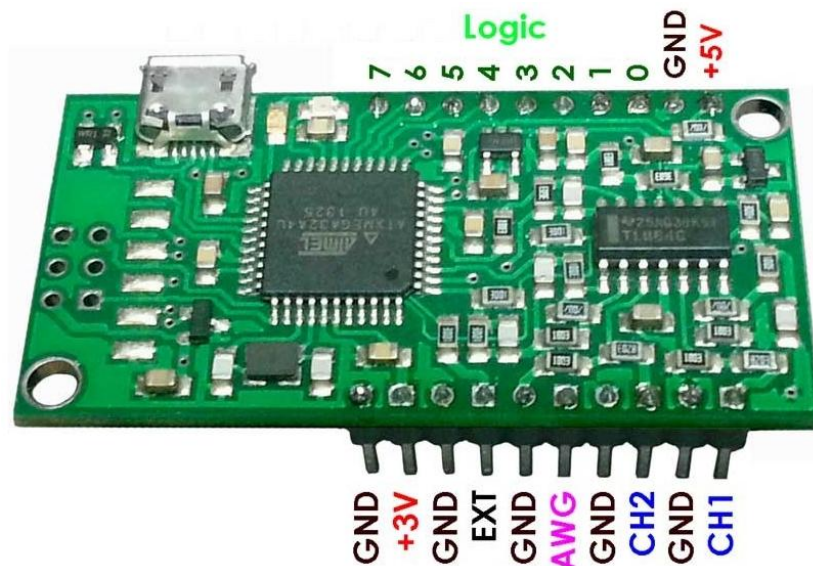


Bild Xprotolab Plain (Foto Gabotronics)

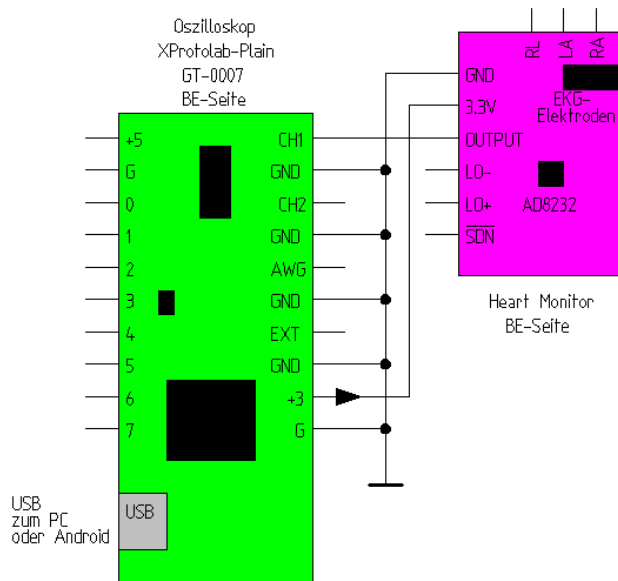


Bild Schaltung Heart Monitor und Xprotolab-Plain

Bei erfolgreicher Kopplung beider Komponenten kann das Ergebnis dann so aussehen. Die Software für PC und Androidgerät ist ein wenig unausgereift – da ist einige Fummelei angesagt.

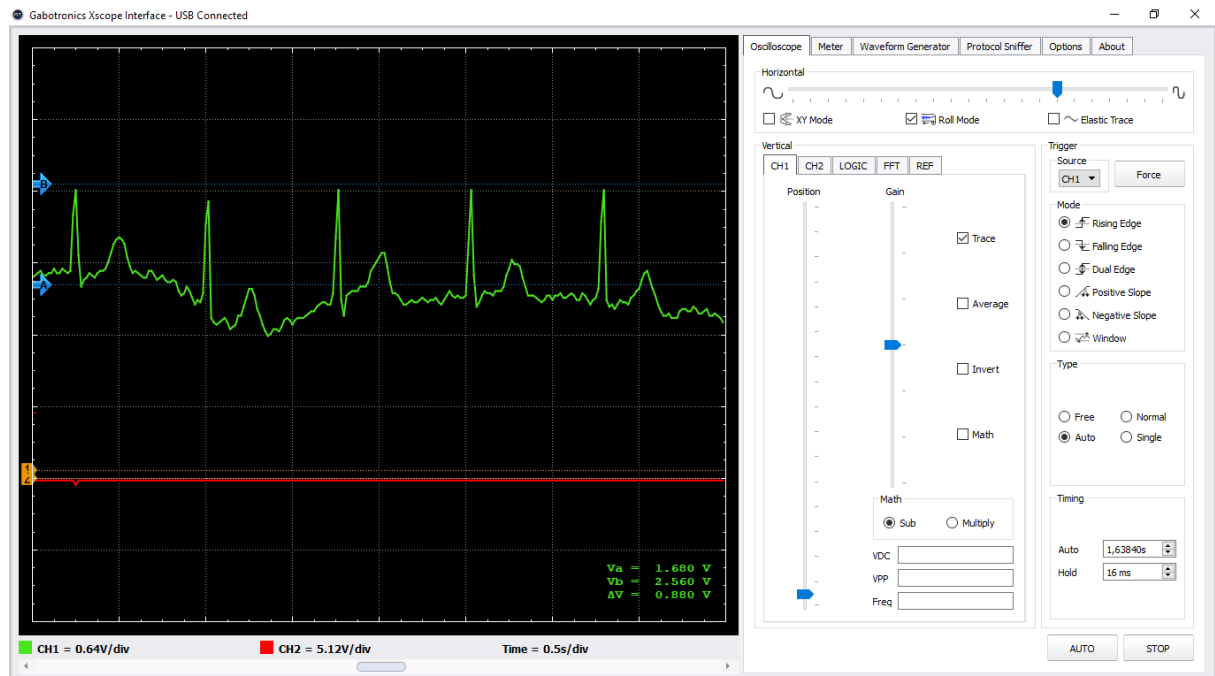


Bild Xprotolab-Plain GT-0007 am PC

Die nächste Stufe – EKG-Aufzeichnungen seriell übertragen

Wie schon beim EKG-Simulator werden wir nun echte EKG-Daten seriell an einen Laptop übertragen. Auch hier nutzen wir den ATtiny13, um erst die analogen Werte vom Heart-Monitor einzulesen und dann als seriellen Datenstrom zum PC zu senden. Voraussetzung ist wie oben ein Pololu-USB AVR Programmer V2.1 oder natürlich ein ähnliches Gerät, das serielle TTL-Daten an die USB-Schnittstelle übertragen kann.

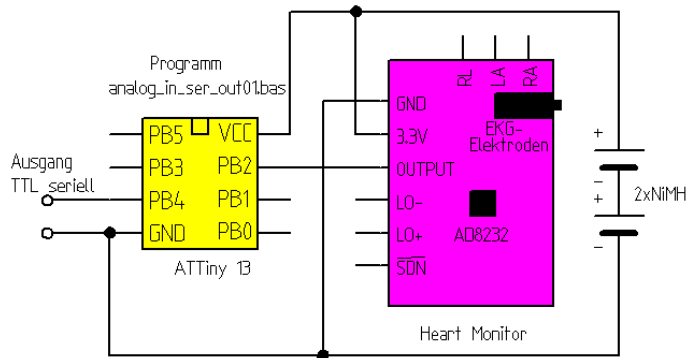


Bild echtes EKG an serielle Schnittstelle

Die Auswertung erfolgt wieder wie oben über ein Terminal-Programm wie hterm und danach z.B. in Excel.

EKG-Aufzeichnungen über längere Zeit mit OpenLog

So spannend das Beobachten eines EKG's ist, richtig interessant wird es, wenn man das EKG über längere Zeit aufnehmen und später analysieren kann. Dabei kommen dann z.B. schwankende Herzfrequenzen oder Aussetzer zum Vorschein. Nochmals der Hinweis, dass solche Beobachtungen nicht zu Diagnosen benutzt werden sollen. Dafür sind die verwendeten Ableitungen und die einfachen Komponenten nicht geeignet.

Zunächst – einige der bisher verwendeten Baugruppen und Programme lassen bereits eine längere Aufzeichnung zu, so z.B. Speicheroptionen der verschiedenen Oszillographen. Einfacher geht es aber mit einem weiteren sehr simplen Baustein, auch von Sparkfun:



Bild OpenLog von Sparkfun (Foto Sparkfun)

Von billigen chinesischen Clones ist abzuraten – der Autor hat mit diesen gleich aussehenden Bauteilen schlechte Erfahrungen gemacht – sie waren vollkommen unprogrammiert und damit nicht geeignet. Wenn einer weiß, wie man die passende Firmware aufspielt – ich wäre dankbar...

Dieser Baustein OpenLog ermöglicht auf einfache Weise das Abspeichern von Text (auch unsere EKG-Werte sind im weitesten Sinne Text) über eine serielle Schnittstelle im TTL-Format auf eine µSD-Karte. Dabei kann man OpenLog mit einer Textdatei auf dieser µS-SD-Karte an verschiedene Baudraten anpassen – im Auslieferungszustand ist die richtige Konfiguration schon auf 9600 Bd eingestellt und es muss nichts gemacht werden. Falls schon mit anderen Einstellungen gearbeitet wurde, wird auf der µSD-Karte eine Text-Datei „config.txt“ mit folgendem Inhalt gespeichert „9600,26,3,0,1,1,0“, um jetzt Daten mit 9600 Baud aufzunehmen.

Nun brauchen wir noch eine Komponente, die aus dem analogen Signal des Heart Monitor einen seriellen Datenstrom erzeugt. Auch dies wird von einem ATTiny13 übernommen:

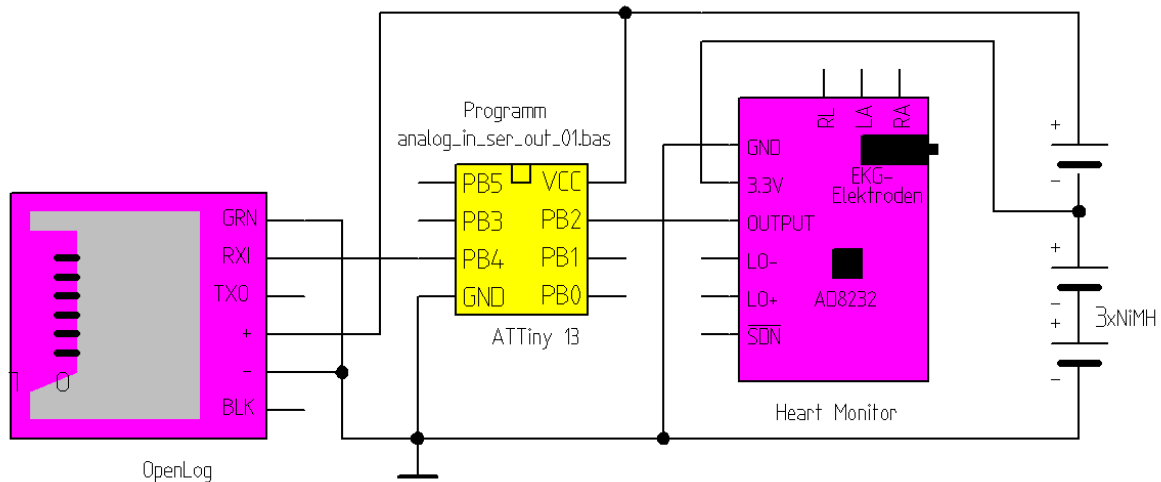


Bild EKG über ATTiny13 an OpenLog

Allerdings gibt es mit OpenLog und den bisherigen Komponenten ein Problem. Während der Heart Monitor mit maximal 3,6 V betrieben wird, verlangt OpenLog mindestens 3,3 V, also deutlich mehr als 2 Zellen NiMH oder Alkanline. Wir helfen uns mit einer zusätzlichen Zelle für die Komponenten OpenLog und ATTiny 13. Wird die obige Schaltung mit Strom versorgt, beginnt sofort die Aufzeichnung und zwar jeweils als Textdatei mit laufender Numerierung.