



Gebhard Raich BSc

## **Entwicklung und Bau eines 24 Kanal EEG Datenrekorders**

### **MASTERARBEIT**

zur Erlangung des akademischen Grades

Diplom-Ingenieur

Masterstudium Elektrotechnik

eingereicht an der

**Technischen Universität Graz**

Betreuer

Ass.Prof. Dipl.-Ing. Dr.techn. Gunter Winkler

Institut für Elektronik

Graz, Mai 2016

## **EIDESSTATTLICHE ERKLÄRUNG**

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

---

Datum

---

Unterschrift

## **Kurzfassung**

Diese Arbeit befasst sich mit der Entwicklung eines medizinischen Messsystems, das im wesentlichen EEG Signale misst und aufzeichnet. Das Gerät dient im klinischen Routinebetrieb zur Ableitung und Aufzeichnung ambulanter 24 Stunden EEG's. Neben einer kurzen, geschichtlichen Einleitung, werden alle Prozessstufen der Entwicklung, angefangen von Pflichtenheft über Gerätekonzept, Überlegungen zur Patientensicherheit, Hardwarerealisation, Software sowie Gehäuse detailliert behandelt. Im Schlusskapitel werden die erzielten Leistungen des Gerätes dargestellt und ein Ausblick auf die Weiterentwicklung des Gerätes gegeben.

## **Abstracts**

This paper deals with the development of a medical measurement unit, which measures and records EEG signals. This device is used in clinical routine treatment for ambulant 24 hour EEG.

After a short historical introduction, all processing steps of development, beginning with duty heft, device concept, considerations belonging to patient security, realizing of electronic hardware, software and at last housing of the device are treated in detail.

In the last chapter the performance of the developed device is shown and an outlook on further development is given.

# Inhaltsverzeichnis

## 1. Allgemeines

### 1.1 EEG Potentiale und Messung

#### 1.1.1 Geschichtliches

#### 1.1.2 Signaltechnische Betrachtung

#### 1.1.3 Elektroden

### 1.2 Aufgabenstellung

### 1.3 Spezifikation des Gerätes

## 2. Systemkonzept

### 2.1 Hardware

#### 2.1.1 Analoge Signalaufbereitung

#### 2.1.2 Mikrocontroller und angeschlossene Einheiten

#### 2.1.3 Stromversorgung über Batterie

#### 2.1.4 Stromversorgung über externes Netzteil

#### 2.1.5 Stromversorgung für analoge und digitale Schaltkreise

#### 2.1.6 Blockschaltbild

### 2.2 Software

#### 2.2.1 Allgemeine Überlegungen

#### 2.2.2 Blockdiagramm

## 3. Detaillösungen

### 3.1 Hardware

#### 3.1.1 Analoge Signalaufbereitung

##### 3.1.1.1 Referenzielle Eingangsverstärker

##### 3.1.1.2 Differenzielle Eingangsverstärker

##### 3.1.1.3 Referenzspannungserzeugung

##### 3.1.1.4 Kalibrier und Impedanzmesssignal

#### 3.1.2 Mikrocontroller und angeschlossene Einheiten

##### 3.1.2.1 Beschaltung des Mikrocontrollers

##### 3.1.2.2 Ethernet

##### 3.1.2.3 SD-Karte und Beschleunigungssensor

##### 3.1.2.4 24Kanal Analog Multiplexer

#### 3.1.3 Stromversorgung über Batterie

#### 3.1.4 Stromversorgung über externes Netzteil

#### 3.1.5 Stromversorgung für analoge und digitale Schaltkreise

### 3.2 Software

#### 3.2.1 Verwendete Datenstrukturen

#### 3.2.2 Verwendete Tasks

3.2.3 Verwendete Interrupts

3.2.4 Startsoftware für Mikrocontroller und Tasks

3.2.5 Hochpassfilter über exponentiell geglätteten Mittelwert

4. Layout der Leiterplatte

4.1 Grundlegende Überlegungen

4.2 Ausführung

5. Gehäuse

6. Messwerte, Diskussion, Ausblick

# 1. Allgemeines

## 1.1 EEG Potentiale und Messung

### 1.1.1 Geschichtliches

Geschichtlich geht das EEG bis in das Jahr 1875 zurück, wo R. Caton erstmals mit Hilfe eines Spiegelgalvanometers elektrische Potentiale des Gehirns von Tieren an der Kopfhaut messen konnte [1].

Der weitere Werdegang war vorgezeichnet, ab den 30er Jahren des letzten Jahrhunderts wurde mit Röhrenverstärkern gearbeitet, ab den 60er Jahren mit Transistorenverstärkern. Die Aufzeichnung erfolgte hier mittels eines Schreibers auf Papier.

In den 90er Jahren war dann die Digitalisierung und Speichermöglichkeit von digitalen Daten so weit fortgeschritten, dass die Messungen digitalisiert und auf der Hard-Disk oder optischen Disk eines PC aufgezeichnet werden konnten (papierloses EEG). Für mobile, tragbare Geräte wurden die Daten mittels eines Tonmodulationsverfahrens auf die damals verbreiteten Musik-Kassetten gespeichert. Halbleiterspeicher in entsprechender Größe waren damals noch nicht erhältlich.

[1] Richard Caton; The Electric Currents of the Brain; British Medical Journal; 28 August 1875 Volume 2, Issue 765; page 278

### 1.1.2 Signaltechnische Betrachtung

Die im klinischen Routinebetrieb abgeleiteten EEG Potentiale umfassen einen Frequenzbereich von 0,3 – 70Hz bei einer Amplitude von 15 – 200 $\mu$ V (nicht pathologisch). Bei Kleinkindern können die Amplituden bis zu 500 $\mu$ V erreichen.

Aus signaltechnischer Sicht handelt es sich beim EEG um ein Rauschsignal, vermischt mit wechselnden periodischen Anteilen.

Die periodischen Anteile werden als Rhythmus bezeichnet und nach ihrer Frequenz folgenderweise unterschieden:

#### a) Alpha – Rhythmus

Frequenzbereich 8 – 12 Hz

Amplitude 20 – 50  $\mu$ V, annähernd sinusförmig

Beobachtbar beim wachen, entspannten Menschen mit geschlossenen Augen

#### b) Beta – Rhythmus

Frequenzbereich 14 – 30 Hz

Amplitude 10 – 30  $\mu$ V, dreieckförmig

Beobachtbar bei erhöhter Aufmerksamkeit, geistiger Tätigkeit

c) Theta – Rhythmus

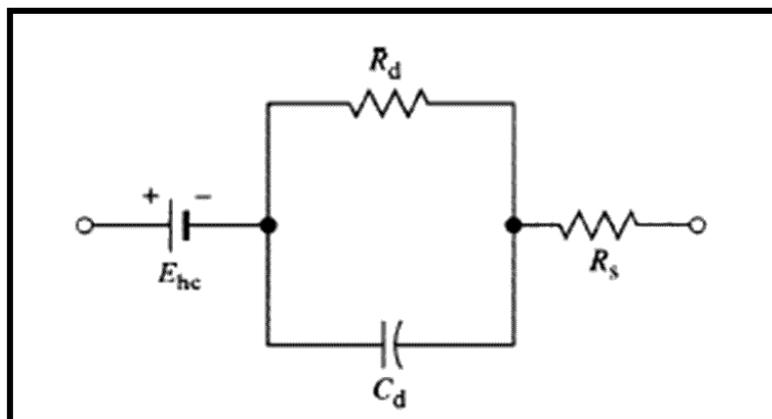
Frequenzbereich 4 – 8Hz  
Amplitude ca.  $20\mu\text{V}$ , bogen – bzw. trapezförmig  
Beobachtbar bei leichter bis mittlerer Schlafentiefe

d) Delta – Rhythmus

Frequenzbereich 1 – 4 Hz  
Amplitude ca.  $100\mu\text{V}$ , bogen – bzw. trapezförmig  
Beobachtbar bei tiefem Schlaf

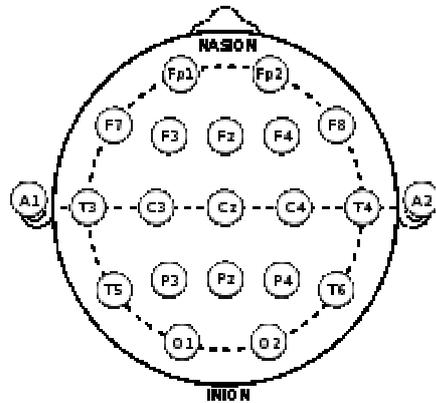
### 1.1.3 Elektroden

Die Elektroden stellen das Bindeglied zwischen Messgerät und dem menschlichen Körper dar. Die Eingänge des Messgerätes müssen an die Elektroden angepasst werden, um das Signal überhaupt erfassen zu können. Untenstehend ein vereinfachtes Ersatzschaltbild des Systems Elektrode-Haut.



Der Widerstand  $R_s$  bildet hierbei den Gewebewiderstand der unteren Hautschichten nach und liegt in einem Bereich von ca. 50 -200 Ohm.  $R_d$  und  $C_d$  bilden die Übergangsimpedanz zwischen Elektrode und Haut ab.  $R_d$  liegt in einem Bereich von 5-100kOhm und  $C_d$  in einem Bereich von 10-50nF. Die Spannung  $E_{hc}$  entsteht durch die Bildung von elektrischen Doppelschichten (Helmholzschichten) zwischen Elektrode und Haut. Die Größenordnung dieser Gleichspannung liegt bei einigen 10mV und ist somit um 2 bis 3 Zehnerpotenzen höher als das Nutzsignal. Aufgrund dieses Umstandes sind die angeschlossenen Verstärkerkanäle als AC Verstärker auszuführen.

Die Anordnung der Elektroden beim Routine – EEG erfolgt nach dem so genannten 10-20 System [2], das von H.H. Jasper im Jahre 1958 als Standard eingeführt wurde, um Messungen besser vergleichbar zu machen. Untenstehend eine Abbildung, wie die Elektroden in diesem System auf dem Kopf angeordnet sind.



[2] Jasper, H.H., The ten-twenty electrode system of the International Federation, *Electroenceph. clin. Neurophysiol.*, 1958, 10: 371-375.

## 1.2 Allgemeine Aufgabenstellung

Vorausschickend wäre zu erwähnen, dass ich seit 25 Jahren Entwickler und Hersteller medizinischer Messgeräte bin. Um auf diesem Geschäfts-Gebiet auch zu einem wirtschaftlichen Erfolg zu kommen, kam es vor 15 Jahren zu einer Zusammenarbeit mit der Fa. Grossegger&Drbal, die, in Wien ansässig, für Marketing, Vertrieb und Kundenservice zuständig ist, und Hr. Erwin Gabler (EPU), der für die Erstellung der Anwendersoftware verantwortlich zeichnet.

Im Jahre 2005 wurde das von mir entwickelte 32Kanal EEG Gerät EPV32 auf dem Markt eingeführt, welches im Jahre 2008 nochmals überarbeitet wurde, und seither gut 200 mal verkauft wurde. Dieses Gerät ist für stationäre EEG Messungen in Spitälern und bei niedergelassenen Neurologen konzipiert. Die Signalqualität und Robustheit der Geräte ist außerordentlich gut, der Aufbau ist allerdings relativ aufwändig und voluminös.

In letzter Zeit kam es zu einem Paradigmenwechsel in der medizinischen Versorgung, die jetzt weniger auf stationäre Versorgung ausgerichtet ist, und gleichzeitig zu einem Anstieg von Patienten mit Schlafstörungen. Dies führte dazu, dass es einen Bedarf an kleinen, mobilen EEG Rekordern gibt. Diese sollten nun nur noch ambulant am Patienten „montiert“ werden. Der Patient kann damit seinen gewohnten Tätigkeiten zu Hause größtenteils nachgehen. Der Rekorder zeichnet neben dem EEG auch weitere Parameter wie z.B. Atmung und Puls über 24 Stunden laufend auf. Die aufgezeichneten Daten können dann später ausgelesen und ausgewertet werden.

Damit erschien die Entwicklung eines eigenen kleinen, tragbaren EEG Rekorders, welcher aber auch im stationären Betrieb als kostengünstige Alternative zum 32 Kanal Gerät eingesetzt werden sollte, sinnvoll. Die Spezifikation des Gerätes wurde dann gemeinsam mit Dr. Grossegger festgelegt.

### 1.3 Spezifikation des Gerätes

- 21 Referenzkanäle<sup>(1)</sup>, 3 Differenzkanäle<sup>(2)</sup>
- 24h Laufzeit über Batterieversorgung
- 24h Aufnahmekapazität
- Datenkommunikation über Ethernet, TCP/IP
- Stromversorgung wahlweise über Datenkabel, handelsüblichen Akku oder Batterie
- 256Hz Abtastrate/Kanal mit Möglichkeit für 512Hz
- Interne Kalibrationsmöglichkeit
- Möglichkeit für Messung der Elektroden – Impedanz
- Eingebauter Beschleunigungssensor zur Feststellung von Patientenbewegungen und Patientenlage
- Robustes, möglichst kleines Gehäuse
- Kostengünstige Herstellung in Kleinserie
- Verwendung der genormten 1,5mm Elektrodenstecker
- Patienten Hand – Taster für die zeitliche Markierung von Ereignissen(z.B. Anfälle)
- Erweiterungsmöglichkeit für ein Pulsoximetermodul<sup>(3)</sup>
- Daten- und Befehlskompatibilität mit bestehendem 32Kanal EEG-System

<sup>(1)</sup> Die Referenzkanäle bilden die Differenzspannung zu der Referenzelektrode ab

<sup>(2)</sup> Die Differenzkanäle bilden die Differenzspannung zwischen den 2 Eingangselektroden ab

<sup>(3)</sup> Messmodul eines anderen Herstellers, welches im wesentlichen Sauerstoffsättigung und Pulsrate misst

## 2.1 Systemkonzept Hardware

Auf Basis des Pflichtenheftes konnte nun das Gesamtsystem entworfen werden.

### 2.1.1 Analoge Signalaufbereitung

Für den analogen Teil des Systems, der 21 Referenzkanäle und 3 Differenzkanäle enthalten soll, existierten vom bestehenden 32 Kanal EEG-System gut funktionierende Schaltungen, die aber dort mit +/- 12V versorgt wurden. Die Hauptschwierigkeiten bestanden nun darin, erstens die Schaltungen auf eine unipolare 3V Versorgung zu ändern und zweitens

passende Halbleiter Bauteile (Operationsverstärker, Instrumentenverstärker) auszuwählen, die eine angemessene Signalqualität gewährleisten. Im Fall der unipolaren 3V Versorgung wurde beschlossen einen „virtuellen“ Ground auf 1,2V einzurichten. Die maximale verarbeitbare ADC Eingangsspannung, die gleichzeitig die Referenzspannung darstellt, wurde mit 2,4V festgelegt. Damit war für den treibenden Verstärker kein rail to rail Typ notwendig, er sollte nur relativ nahe an 0V aussteuern können. Für den Eingangs OPV, der in der ersten Stufe das EEG Signal verstärkt, wären Eigenschaften wie

- geringer Bias Strom
- geringes Rauschen (0,1 – 70Hz)
- hohe Gleichtaktunterdrückung
- geringer Stromverbrauch
- Betriebsspannung bis 3V herab

äußerst günstig. Leider widersprechen sich so manche Eigenschaften wie z.B. geringes Rauschen und geringer Stromverbrauch. Rauscharme Verstärker haben einen relativ hohen Stromverbrauch aufgrund von vielen Parallelstrukturen im Differenzverstärkereingang.

Nach langem Suchen, wobei auch so scheinbar nebensächliche Dinge wie Verfügbarkeit eine große Rolle spielten, wurde für den OPA2244 in der Eingangsstufe und den AD8223 in der Differenzstufe entschieden.

## 2.1.2 Mikrocontroller und angeschlossene Einheiten

Der Microcontroller bildet das Kernstück des Systems. Zahlreiche Funktionen, die für das Gesamtsystem benötigt werden, sollten darin bereits enthalten sein:

- 12Kanal ADC 12Bit
- DMA
- DAC mit 12Bit
- Bus-Interface 16Bit
- SDIO-Interface für SD Speicher Karte
- 2 UART's
- SPI-Interface
- ca. 40 freie Portpins für Steuerungsaufgaben
- interner RAM und Flash-Speicher
- Betrieb mit 3,0V
- Nested Interrupts<sup>(1)</sup>

<sup>(1)</sup>Bezeichnet ein Interruptsystem, in dem ein Interrupt höherer Priorität einen bereits laufenden Interrupt niedriger Priorität unterbrechen kann.

Der 12 Kanal ADC wurde deshalb angedacht, weil es auch nach längerem Suchen kein passendes Mikrocontroller Modell gab, das 24 ADC Kanäle aufwies. Daher mussten die 24 Analog-Kanäle mittels eines Multiplexers auf 12 ADC Kanäle aufgeteilt werden.

Beim 32 Kanal EEG System hat sich gezeigt, dass durch „oversampling“ und anschließendes mitteln eine sehr gute Auflösung erzielt werden kann. Dieses Verfahren sollte auch hier zum Einsatz kommen. Ein 32 faches oversampling wie beim 32Kanal Gerät würde allerdings die Abtastrate auf 8192Hz/Kanal

erhöhen. Es war zum Entwurfszeitpunkt nicht klar, ob mit dieser hohen Abtastrate gemessen werden kann, weil vom Schreibtisch aus eine Prognose über die CPU Auslastung eher mit Kaffeesud Leserei vergleichbar ist.

Klar war aber, dass die CPU Auslastung sehr hoch sein wird und es daher sinnvollerweise eine DMA Einheit im Mikrocontroller geben sollte, um die Messdaten möglichst schnell zu verarbeiten.

Der DAC sollte das Kalibriersignal beziehungsweise das Impedanz Messsignal generieren.

Für den Ethernet Baustein W5300 des Herstellers Wiznet, der ebenfalls schon im 32 Kanal EEG System verwendet wurde, und der neben dem MAC(Media Access Controller) und PHY(Signalerzeugung) auch einen eingebauten TCP/IP Kommunikationsstack enthält, war ein 16Bit Bus Interface mit 9 Adressleitungen notwendig. Der Vorteil dieses Ethernet IC ist, dass die CPU für die Kommunikation mit dem PC praktisch keine Rechenleistung benötigt, da diese die Daten direkt in ein bis zu 64kByte großes FIFO des Ethernet IC schreiben beziehungsweise von dort auch lesen kann. Die Bandbreite dieser Kommunikationseinrichtung beträgt bis zu 5MByte/s, was sich vor allem in Hinblick auf die Zeitdauer für das Auslesen gespeicherter Messdaten günstig auswirkt.

Für das Speichern der Messdaten sollte eine Micro-SD-Karte eingesetzt werden. Die minimale Größe der Karte wurde folgendermaßen berechnet:

Laufzeit:	$24\text{h} = 86400\text{s}$
Datenstrom:	$24 \text{ Kanäle} \cdot 256 \text{ Messwerte}(16\text{Bit}) \text{ pro s} = 12288\text{Byte/s}$
Datenvolumen:	$12288 \text{ Byte/s} \cdot 86400\text{s} = 1,06 \text{ GByte}$

Um Daten-Kompatibilität mit dem bestehenden 32 Kanal EEG System zu ermöglichen, wurde ein Minimum von 1,6GByte Speicherplatz errechnet. Da es kaum noch SD-Karten mit 2GByte gibt, und vom Auftraggeber auch 512Hz Abtastrate angedacht wurden, was die Datenmenge verdoppeln würde, wurde schließlich eine 4GByte Karte eingesetzt. Diese SD-Speicherkarten fallen unter die so genannte „Consumer Elektronik“, was bedeutet, dass meist keine Datenblätter erhältlich sind, und, wie sich später zeigen sollte, wichtige Parameter wie z.B. Stromverbrauch je nach Typ äußerst unterschiedlich sind.

Für den Download der Geräte Software, der Gerätedaten wie z.B. Seriennummer oder Kalibrierwerte, und der Ausgabe von Variablenwerten für Debugzwecke muss ein UART Anschluss vorhanden sein. Um den Stromverbrauch niedrig zu halten und Platz zu sparen, werden keine RS232 Pegel erzeugt, sondern direkt 3V Pegel verwendet.

Weiters braucht es einen 2. UART Anschluss, um das optionale Chip-Oximeter-Modul zu betreiben. Dieses, von der Fa. Corscience gelieferte Modul, misst über einen Fingersensor Sauerstoffsättigung und Pulsrate und kommuniziert ebenfalls über eine UART Schnittstelle.

Der vorgesehene Beschleunigungssensor LIS302 von ST hat mit SPI und I<sup>2</sup>C 2 Schnittstellen eingebaut. Da die Kommunikation über I<sup>2</sup>C wesentlich höheren „Overhead“ besitzt und die CPU unnötig belastet hätte, wurde die SPI Schnittstelle verwendet.

Für den RAM Bedarf des Gerätes wurden Abschätzungen gemacht, die sich an der Funktionalität orientierten. Dabei spielen naturgemäß Bufferlängen und Stackgrößen eine maßgebliche Rolle. Die größten Buffer mussten für das Abspeichern der Messdaten auf SD Karte angelegt werden und waren über 8kByte groß. Dazu noch Buffer für ADC Daten und UART's. Weiters erschienen Stackgrößen von 4kByte aufgrund der hohen Funktionalität und der angedachten Verwendung eines Echtzeit-Betriebssystems ein absolutes Minimum zu sein. Eine minimale RAM Größe von 32kByte erschien auch im Hinblick auf Weiterentwicklung notwendig. Aus Platzgründen muss dieses RAM Teil des Mikrocontrollers sein.

Für die Größe des Flash Speicher, der im wesentlichen die Gerätesoftware enthält, gab es aus früheren Applikationen Abschätzungen, die nicht über 64kByte hinausgingen.

Aufgrund der hohen Funktionalität und der Vielzahl von zeitkritischen Interrupts sollte die CPU verschachtelte („nested“) Interrupts zulassen.

Als Mikrocontroller wurde schließlich der STM32F103ZET gewählt, weil dieser alle der benötigten Funktionen bereits integriert hat, gut verfügbar ist und ein sehr überzeugendes Preis/Leistungsverhältnis besteht.

### 2.1.3 Stromversorgung über Batterie

Das Gerät soll wahlweise über das Datenkabel von einer externen Stromquelle, oder mit handelsüblichen Akkus oder Batterien intern versorgt werden. Ersteres ist für die Anwendung als stationäres EEG Gerät gedacht, die interne Versorgung für ambulantes EEG und Schlafuntersuchungen. Vor allem im ersten Fall ist die Einhaltung der Patientensicherheit keine ganz einfache Sache. Aber auch der Batteriebetrieb wird von der Prüf und Zulassungsstelle durchaus kritisch betrachtet. Es ist im diesem Fall sicherzustellen, dass es zu keiner Überhitzung oder gar Explosion der Energiespeicher kommt. Damit scheidet die, vor allem im Modellbau beliebten, Lithium-Polymer Akkus mit hoher Energiedichte aus. Handy-Akkus wären von der Prüfstelle akzeptiert worden, weil diese eine interne Schutzeinrichtung besitzen, jedoch wurde auch dieses verworfen, weil eine entsprechende Halterung mit Kontaktierung selbst hergestellt werden müsste und die Spannungslage mit 3 - 4,2 V eher ungünstig war. Weiters hatte der Auftraggeber Bedenken, dass das regelmäßige Aufladen des Akkus vom anwendenden Personal nicht korrekt durchgeführt wird, und man daher möglicherweise mit stark verkürzten Messdauern rechnen muss. Schließlich wurde entschieden, dass 4Stk. handelsübliche AAA Zellen, entweder Batterie oder Akku die Energie für die interne Versorgung liefern. Die Vorteile waren die Sicherheit, die Verfügbarkeit und die Möglichkeit des Einsatzes einer fertigen Batteriehalterung. Als Nachteil wurde die eher geringe Energiedichte angesehen. Die zur Verfügung stehende Energie wurde überschlagsmäßig wie folgt berechnet:

4Stk. Akku NiMH 1,2V/1000mAh ergibt  $4,8V * 1Ah = 4,8Wh$

Dies sollte ein eher niedrig geschätzter Wert sein, weil die Akkus zum Teil auch deutlich höhere Kapazitäten aufweisen und bei Alkali Batterien ohnehin mit gut 50% mehr Kapazität gerechnet werden kann.

Die Spannungslage beträgt bei Serienschaltung der 4 Zellen 3,6 bis 6,5V.

Damit muss der Spannungsregler, der die Schaltung mit konstanter Spannung versorgt, zurechtkommen. Weiters muss der Spannungsregler einen sehr hohen Wirkungsgrad auch bei niedriger Stromabgabe (ca. 50mA) besitzen.

Linearregler scheiden aufgrund ihres Arbeitsprinzips aus, ebenso Schaltregler mit Freilaufdioden, hohem Eigenstrombedarf oder niedriger Schaltfrequenz (große Drossel). Die Höhe der Versorgungsspannung geht in die Verlustleistung der digitalen Schaltkreise quadratisch ein, bei den Operationsverstärkern (OPV) linear, aufgrund der intern vielfach verwendeten Konstantstromquellen..

Die Höhe der Versorgungsspannung für die gesamte Schaltung wurde mit 3,0V festgelegt. Dies deshalb, weil es bei dieser Spannung noch eine größere Anzahl von OPV's gibt, die für diese Applikation noch brauchbare Parameter aufweisen, und auch der Ethernet-IC eine minimale Betriebsspannung von 3,0V benötigt. Weiters brauchen Abwärtsschaltregler eine um einige 100mV höhere Eingangsspannung als die Ausgangsspannung. Es gibt auch Schaltreglerkonzepte, die sowohl eine höhere wie auch niedrigere Ausgangsspannung als die Eingangsspannung erzeugen können. Diese sind unter dem Namen SEPIC-Wandler (Single Ended Primary Inductance Converter) bekannt, besitzen aber einen deutlich schlechteren Wirkungsgrad als die Abwärtsschaltregler und scheiden daher aus. Die Stromersparnis der CPU bei 3,0V sollte in etwa 20% im Vergleich zu einer 3,3V Versorgung betragen.

Die beabsichtigte Taktfrequenz der CPU sollte 50MHz betragen. Aus dem Datenblatt der CPU entnimmt man auf Seite 50 des DB 34,6mA@3,3V bei 48MHz. Damit sollte man bei 3,0V auf etwa 28mA kommen. Somit kann man eine Tabelle mit den maßgeblichen Verbrauchern erstellen, um eine Abschätzung der Laufzeit im Falle der Batterieversorgung machen zu können.

Anz.	Bezeichnung	Stromverbrauch/Stk.	Stromverbrauch ges.
1	CPU	28mA	28mA
24	AD8223(Instr. Verst.)	0,5mA	12mA
27	OPA244(OPV)	0,06mA	1,62mA
1	LIS302 (Beschl.Sensor)	0,4mA	0,4mA
1	SD-Karte	70mA <sup>(1)</sup>	7mA <sup>(2)</sup>

<sup>(1)</sup> Durchschnittlicher Stromverbrauch beim Lese/Schreibe Vorgang

<sup>(2)</sup> Schreib/Lesedauer 10% (geschätzt)

Damit kommt man auf ca. 50mA Stromaufnahme auf der 3,0V

Versorgungsschiene und somit auf eine Verlustleistung von 150mW.

Wie oben bereits angeführt, stehen 4,8Wh Energie für 24 Stunden Laufzeit zur Verfügung. Damit darf die Leistungsaufnahme  $4,8\text{Wh}/24\text{h} = 0,2\text{W}$  nicht übersteigen. Dabei ist allerdings der Schaltreglerwirkungsgrad nicht miteinbezogen. Bei einem Wirkungsgrad von geschätzt 80% darf die Schaltung

demnach 160mW Leistungsaufnahme haben. Man sieht, dass die gewünschte Laufzeit von 24h im Bereich des Möglichen ist.

#### 2.1.4 Stromversorgung über externes Netzteil

Wie bereits erwähnt, ist diese Stromversorgung mit besonderer Hinsicht auf die Patientensicherheit zu konzipieren. Damit sind im wesentlichen folgende Kriterien einzuhalten:

Trennspannung: 1500Vrms für 60s

Trennstrecke: 8mm Kriechstrecke oder 2mm Luftstrecke

Da im Falle der externen Versorgung auch der Ethernet-IC W5300 in Betrieb ist, erhöht sich die notwendige Leistung drastisch. Laut DB muss man mit 250mA Stromaufnahme und damit 750mW Verlustleistung rechnen.

Zusammen mit der Verlustleistung der restlichen Schaltung muss man mit ca. 1W rechnen. Käufliche, isolierte DC/DC Wandler erfüllen die eingangs erwähnten Sicherheitskriterien nicht, oder sind deutlich zu groß.

Die Versuchung, das Gerät mit einer Standard POE (Power over Ethernet) Versorgung, die durch den IEEE-Standard 802.3af-2003 beschrieben wird, zu betreiben war groß, jedoch hätte der Spannungslevel von bis zu 57V zu noch höheren Anforderungen an Trennspannung und Trennstrecke geführt, die mit Standard Bauteilen kaum mehr einzuhalten gewesen wären.

In Versuchsaufbauten konnte man feststellen, dass man auch mit einem Ethernetrafo, der im „Normalfall“ eigentlich nur der Signalübertragung dient, einen isolierten DC/DC Wandler mit bis zu 2W Ausgangsleistung aufbauen kann. Diese Trafos sind auch mit einer Trennspannung von 1500Vrms und 8mm Kriechstrecke zwischen den Ein – und Ausgangspins kostengünstig erhältlich. Allerdings gab es dabei noch gewisse Einschränkungen, die maximale Eingangsspannung durfte 5V nicht überschreiten, und ging, da der Wandler ungerichtet ist, unmittelbar in die Ausgangsspannung ein. Dies ergab das nächste Problem: Der Auftraggeber wollte verschieden lange Anschlusskabel verwenden, deren Länge bis zu 15m betragen sollte. Da die Standard Ethernetkabel nur für Signalübertragung verwendet werden, besitzen sie kleine Querschnitte (0,25mm<sup>2</sup>) und sind teilweise aus Kostengründen in Aluminium statt Kupfer ausgeführt. Daher muss man in diesem Fall mit unbestimmten und erheblichen Spannungsabfällen am Kabel rechnen. Damit war klar, dass man eine deutlich höhere Spannung als die 5V am Kabel braucht. Schließlich wurde die Spannungsebene am Kabel mit 12V festgelegt. Diese Spannung wird von einem medizinischen Steckernetzteil geliefert. Am Gerät selbst werden die 12V (oder das was davon noch übrig ist) mit einem Abwärtsschaltregler auf 4,6V gebracht und dann dem DC/DC Wandler als Eingangsspannung zur Verfügung gestellt.

#### 2.1.5 Stromversorgung für analoge und digitale Schaltkreise

Es hat sich bei früheren EEG Verstärker Entwicklungen gezeigt, dass eine Trennung der Stromversorgung für analoge und digitale Schaltungsteile im Hinblick auf die Signalqualität sehr sinnvoll ist. Dies deshalb, weil auch kleine Spannungsschwankungen im 20mV Bereich auf der Versorgungsschiene auf

das Messsignal einen deutlichen Einfluss haben. Dabei spielt die Frequenz der Spannungsschwankungen eine große Rolle.

Hohe Frequenzen von 100kHz aufwärts können durch einen einfachen Tiefpass vor dem ADC Eingang gut gedämpft werden und sind daher eher unkritisch. Frequenzen im unteren kHz Bereich machen sich durch Aliasing im abgetasteten Signal bemerkbar, weil die Dämpfung des Tiefpasses hier schon deutlich schlechter ist.

Frequenzen im Messbereich von 0,3 bis 70Hz verursachen die größten Probleme, weil es hier keine Dämpfung gibt. Das PSRR (power supply rejection ratio) der Verstärker wäre zwar im unteren Frequenzbereich hoch (ca. 100dB), nützt aber nicht viel, wenn es sich um einen plötzlichen, steilflankigen Spannungseinbruch handelt, der sich mit niedriger Frequenz wiederholt. Genau dieses Szenario ist beim Abspeichern der Messdaten auf die SD Karte zu befürchten, weil hier die Stromaufnahme der SD Karte sprunghaft ansteigt und die Periodizität des Vorgangs bei 4Hz liegt.

Aus diesen Gründen wurden 2 Abwärtsschaltregler vom Typ TPS62230 von TI für die getrennte Versorgung der analogen und digitalen Schaltkreise eingesetzt.

Für den Ethernet IC wurde ebenfalls ein eigener Schaltregler vom Typ TPS5420 von TI eingesetzt, der von der Ausgangsspannung des isolierten DC/DC Wandlers mit ca. 8V gespeist wird.

## 2.1.6 Blockschaltbild der Hardware

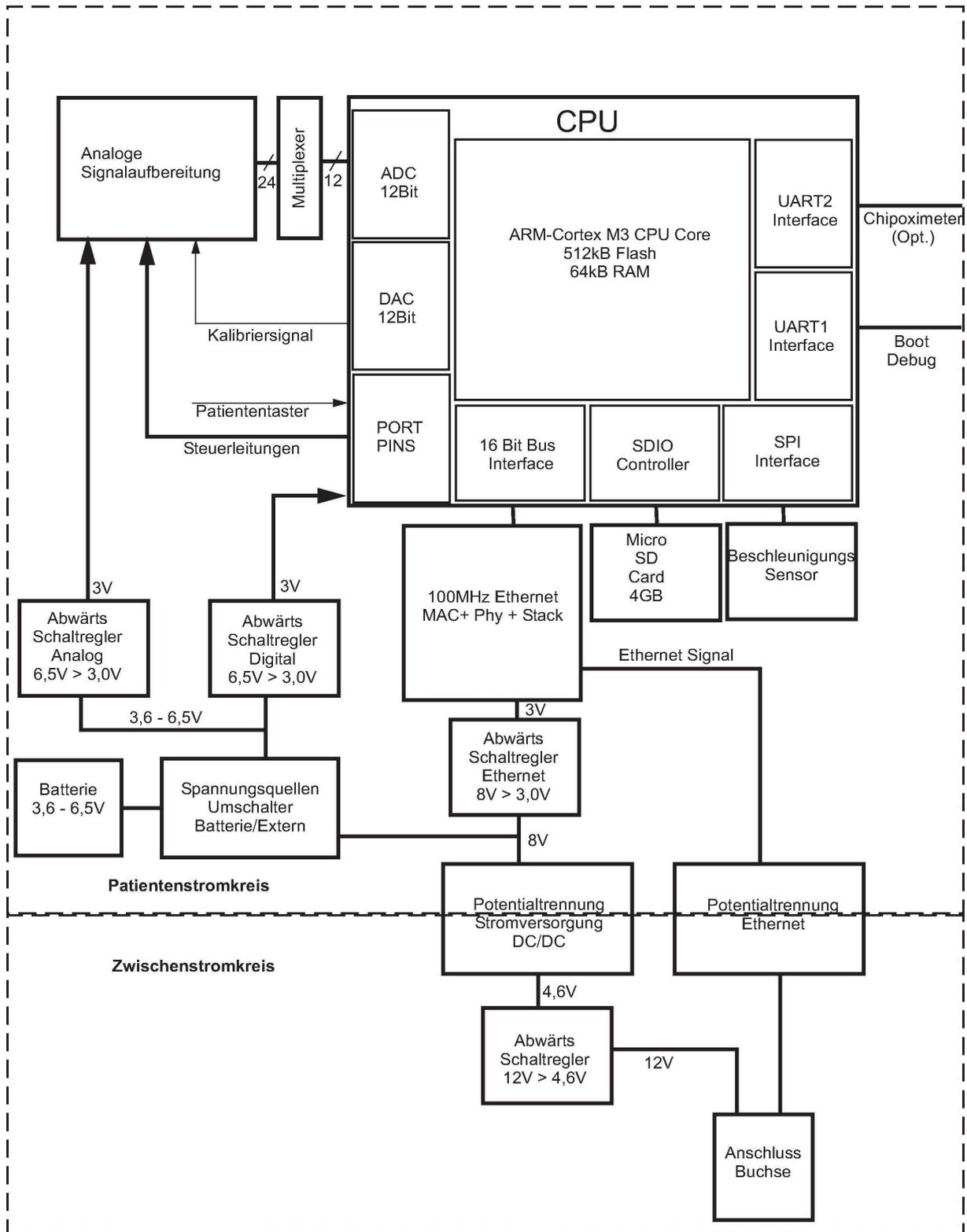


Bild 2.1 Blockschaltbild Hardware

## 2.2 Software

### 2.2.1 Allgemeine Überlegungen

Wie man aus dem Hardware Blockschaltbild erahnen kann, gibt es für die CPU vielerlei Aufgaben, die sich auch zeitlich überschneiden können. Die wesentlichste Belastung für die CPU sind die Werte des ADC, die bei einer Abtastrate von 8192Hz/Kanal immerhin mit 196608 Hz (bei 24Kanälen) generiert werden. Um diese Datenflut zu bewältigen ist das Vorhandensein einer DMA Einheit, die die Werte des ADC direkt in einen Buffer schreiben kann, sehr hilfreich. Weiters wird hier der Einsatz eines Doppelbuffer-Systems notwendig sein, damit immer ein Buffer zum Einfüllen der Daten bereitsteht, während der andere weiterverarbeitet wird. Diese Dinge würden jetzt einmal die Probleme unmittelbar hinter dem ADC weitgehend lösen.

Aber es gibt noch ein weiteres grundsätzliches Problem. Das Abspeichern der Daten auf die SD Karte dauert mehrere ms, im ungünstigsten Fall bis zu 100ms. Daraus ergibt sich die Forderung, dass dieser Prozess unterbrechbar und fortsetzbar sein muss. Eine Unterbrechung einer laufenden Prozedur ist generell nur über einen Interrupt möglich, der in diesem Fall von der Software ausgelöst werden müsste. Dies wäre prinzipiell auch möglich, würde aber zu einer eher unübersichtlichen, komplexen Software führen, bei der man sehr wahrscheinlich mit allen möglichen Seiteneffekten zu rechnen hat.

Eine andere Möglichkeit ist der Einsatz eines Echtzeit Betriebssystems. Hier weist man einer bestimmten Funktionalität einen „Task“ zu, der zum einen eine bestimmte Priorität hat, zum anderen einen eigenen Stack. Dies bewirkt, dass ein Task niedriger Priorität von einem Task höherer Priorität jederzeit unterbrochen werden kann, und später ohne Datenverlust wieder fortgesetzt werden kann (preemptives System). Tasks können auch sehr einfach periodisch ausgeführt werden, was z.B. bei der Ansteuerung einer blinkenden Status-LED enorm hilfreich ist. Die Abfolge der verschiedenen Tasks wird durch den so genannten Scheduler vorgegeben. Dieser wird seinerseits durch einen Timer-Interrupt regelmäßig (z.B. alle 1ms) aufgerufen und legt fest, welcher Task im Hinblick auf die Priorität als nächstes abgearbeitet wird.

Auf das Interruptsystem hat das Echtzeit Betriebssystem keinen Einfluss, bis auf die Möglichkeit, von einer Interruptroutine aus eine Semaphore (Signal) an einen bestimmten Task zu schicken, um diesen dann durch den Scheduler möglichst rasch zu starten.

Alles in allem führt so ein Betriebssystem dazu, dass die vorhandene Rechenleistung sehr gut zeitlich aufgeteilt werden kann, wobei der regelmäßige Aufruf des Schedulers kaum ins Gewicht fällt.

Aufgrund dieser Vorteile wurde entschieden, das Echtzeit Betriebssystem „Keil RTX“ einzusetzen, welches schon sehr lange am Markt ist und außerdem mit einer leicht überblickbaren Entwicklungsumgebung geliefert wird.

## 2.2.2 Blockdiagramm der Software

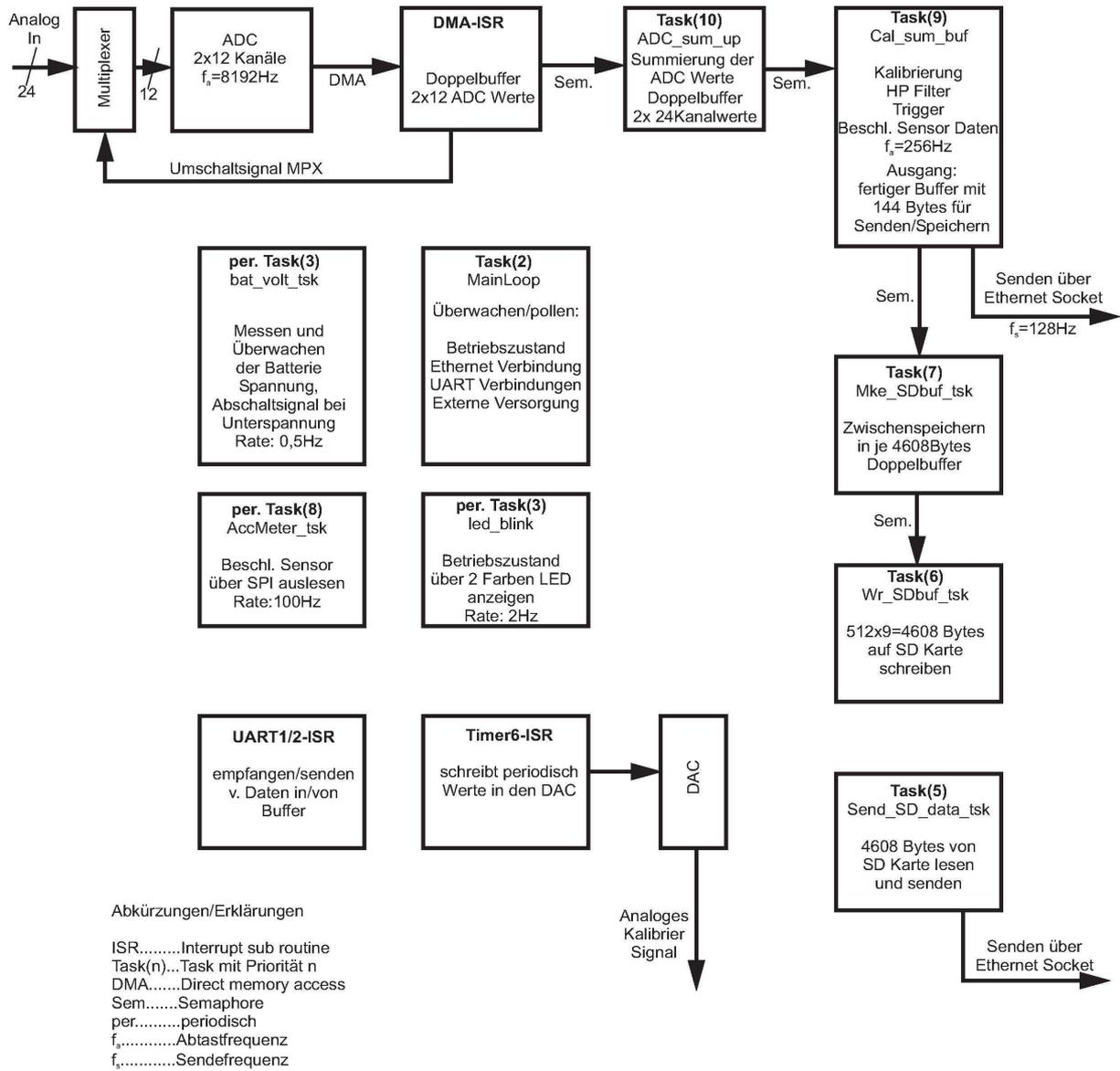


Bild 2.2 Blockdiagramm Software

### 3. Detaillösungen

#### 3.1.1 Analoge Signalaufbereitung

##### 3.1.1.1 Eingangsverstärker

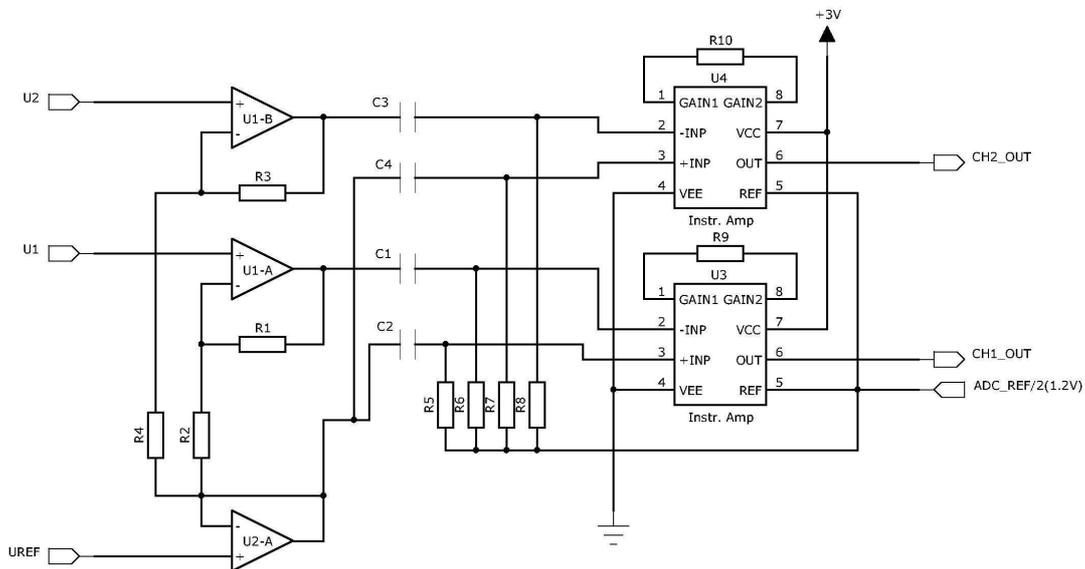


Bild 3.1 Prinzipschaltung referenzieller Eingangsverstärker

Grundsätzliche Schaltungsbeschreibung des referenziellen Eingangsverstärkers:

Wie auf obiger Abbildung zu sehen ist, handelt es sich um eine nicht invertierende Beschaltung eines OPV(U1), dessen Referenzpunkt von einem zweiten OPV(U2) angesteuert ist.

Der OPV U2 ist als Spannungsfolger ausgeführt, das heißt, dass die Ausgangsspannung gleich der Eingangsspannung  $U_{REF}$  ist. Die Ausgangsspannung  $U_{A1}$  am OPV U1-A berechnet sich folgendermaßen :

Da die Verstärkung eines idealen OPV unendlich ist, beträgt die Spannung am – Eingang von U1-A ebenfalls  $U1$ . Demnach fließt ein Strom  $I1$  durch R2, der ebenfalls durch R1 fließen muss, da auch die Eingangsströme des OPV als 0 angenommen werden. Es gilt:

$$I1 = (U1 - U_{REF}) / R2 = (U_{A1} - U1) / R1$$

Nach entsprechender Umformung erhält man:

$$U_{A1} = (R1/R2) * (U1 - U_{REF}) + U1$$

Diese Spannung wie auch  $U_{REF}$  wird über einen Hochpass ( $C1/R6$ ,  $C2/R5$ ) sehr niedriger Grenzfrequenz den Eingängen eines Instrumentenverstärkers U3 zugeführt. Dieser bildet die Differenz der 2 Spannungen, verstärkt diese und addiert noch die Gleichspannung  $ADC\_REF/2$  dazu. Für die Berechnung wird der Hochpass hier nicht berücksichtigt. Es gilt:

$$U_{\text{CHI\_OUT}} = (U_{\text{REF}} - U_{\text{A1}}) * G + \text{ADC\_REF}/2$$

Wobei  $G = F(R9)$  ... Verstärkung des Instrumentenverstärkers  
Nach Einsetzen von  $U_{\text{A1}}$  und Umformen erhält man:

$$U_{\text{CHI\_OUT}} = [(U_{\text{REF}} - U_1) * (R1/R2 + 1)] * G + \text{ADC\_REF}/2$$

Man sieht, dass nur die Differenz der Eingangsspannungen verstärkt wird. Ein Gleichtaktsignal ( $U_1 = U_{\text{REF}}$ ) wird theoretisch vollständig unterdrückt. Betrachtet man aber die realen Verhältnisse, so erhält man doch deutlich andere Werte für die Gleichtaktunterdrückung, die bekanntermaßen mit  $A_{\text{GL}}/A_{\text{DIFF}}$  definiert ist.

Zunächst werden die 2 Hochpässe ( $C1/R6$ ,  $C2/R5$ ) betrachtet. Für eine bestmögliche Gleichtaktunterdrückung wäre folgende Gleichung zu erfüllen:

$$C1 * R6 = \tau_1 = C2 * R5 = \tau_2$$

Nachdem SMD Keramikkondensatoren im Kapazitätsbereich von  $1\mu\text{F}$  eine Toleranz von  $\pm 10\%$  aufweisen, kann obige Forderung nur schlecht erfüllt werden. Eine Berechnung zur Abschätzung der realen Gleichtaktunterdrückung ist daher anzustellen.

Es wird zunächst die Spannungsdifferenz bezogen auf die Eingangsspannung an den Ausgängen der Hochpässe berechnet:

$$U_d/U_e = U_{a1}/U_e - U_{a2}/U_e = \frac{1}{1 + (1/j\omega\tau_1)} - \frac{1}{1 + (1/j\omega\tau_2)}$$

Nach der Umformung zu einem geschlossenen Ausdruck ergibt sich:

$$U_d/U_e = \frac{j\omega(\tau_2 - \tau_1)}{\omega^2\tau_1\tau_2 - j\omega(\tau_2 + \tau_1)}$$

Nach weiterer Umformung zu einem Ausdruck mit getrenntem Real und Imaginärteil:

$$U_d/U_e = \frac{(\tau_2^2 - \tau_1^2) + j\omega(\tau_2 - \tau_1)\tau_1\tau_2}{\omega^2\tau_1^2\tau_2^2 + (\tau_1 + \tau_2)^2}$$

Der interessierende Betrag von  $U_d/U_e$  errechnet sich wie folgt:

$$|U_d/U_e| = \frac{\sqrt{(\tau_2^2 - \tau_1^2)^2 + [(\tau_2 - \tau_1)\tau_1\tau_2]^2} \omega^2}{\omega^2\tau_1^2\tau_2^2 + (\tau_1 + \tau_2)^2}$$

Aufgrund der Grenzfrequenz von 0,2Hz beträgt der nominelle Wert für  $\tau_{\text{nom}} = 0,82\text{s}$  entsprechend  $R=820\text{k}$  und  $C=1\mu\text{F}$ .

Die Gleichtaktunterdrückung ist vor allem bei 50Hz interessant, daher ist  $\omega = 100\pi/\text{s}$ .

Folgende Tabelle macht die Auswirkungen der Bauteilvariationen ersichtlich:

$\tau_2/\tau_{\text{nom}}$	$ U_d/U_e [\text{dB}]$
1,2	-64
1,1	-69
1,05	-74,5
1,01	-86

Diese Werte stellen nicht die Gleitaktunterdrückung der gesamten Schaltung dar, da ja vor dem Hochpass ein Verstärker sitzt, der die Gleitaktspannung mit 1 und die Gegentaktspannung mit dem Faktor  $R1/R2 + 1$  verstärkt. Für eine hohe Gleitaktunterdrückung wäre also eine hohe Verstärkung dieser Stufe wünschenswert. Aufgrund von möglichen Gleichspannungspotentialen im 100mV Bereich am Eingang und der sehr niedrigen Betriebsspannung von 3V würde das aber sehr leicht zu einer Übersteuerung des Eingangsverstärkers führen. Daher wurde hier die Verstärkung mit  $4,8 = 13,7\text{dB}$  entsprechend  $R1=18\text{k}$  und  $R2=4,7\text{k}$  festgelegt.

Dieser Wert ist zu dem  $|U_d/U_e|$  Wert der Tabelle zu addieren.

Damit ergibt sich für die Gleitaktunterdrückung bei idealem Differenzbild U3 bei 5% Bauteilvariation am Hochpass (ein realistischer Wert)

$$74,5\text{dB} + 13,7\text{dB} = 88,2\text{dB}.$$

Für die Bestimmung der Gleitaktunterdrückung der gesamten Schaltung ist noch der Einfluss des nicht idealen Differenzbilders U3 (Instrumentenverstärker) zu berücksichtigen.

Aus dem Datenblatt des verwendeten AD8223 entnimmt man für das CMRR 90dB. Die Verstärkung dieser Stufe wurde mit 247,5 entsprechend einem R9 von 330 Ohm, festgelegt.

Die Gesamtverstärkung beträgt demnach  $247,5 * 4,8 = 1188$  womit ein Eingangsspannungsbereich von  $\pm 1\text{mV}$  erreicht wird.

Es wird eine Gleitaktspannung von 1V am Eingang des Gesamtverstärkers angenommen. An den Eingängen des AD8223 liegen somit:

$$U_{\text{GL}} = 1\text{V} \text{ (bleibt praktisch unverändert)}$$

$$U_{\text{Diff}} = 188\mu\text{V} \text{ (74,5dB Dämpfung)}$$

Am Ausgang des AD8223 ergibt sich somit:

$$U_{a1} = U_{\text{GL}} * \text{CMRR} = 1 * 3,16 * 10^{-5} = 31,6\mu\text{V}$$

$$U_{a2} = U_{\text{Diff}} * 247,5 = 0,188 * 247,5 = 46,5\text{mV}$$

Damit  $U_a = 46,5\text{mV} + 0,0316\text{mV} = 46,5316\text{mV}$  (worst case, gleichphasige Spannungen)

Man sieht, dass hier der Einfluss der CMRR vom AD8223 sehr gering ist ( $<0,1\%$ ). Damit ergibt sich die fast unveränderte CMRR des Gesamtverstärkers mit 88,15dB.

Die obige Prinzipschaltung erfüllt noch nicht alle Anforderungen bezüglich Funktionalität und wird deshalb mit zusätzlichen Schaltungselementen ergänzt:

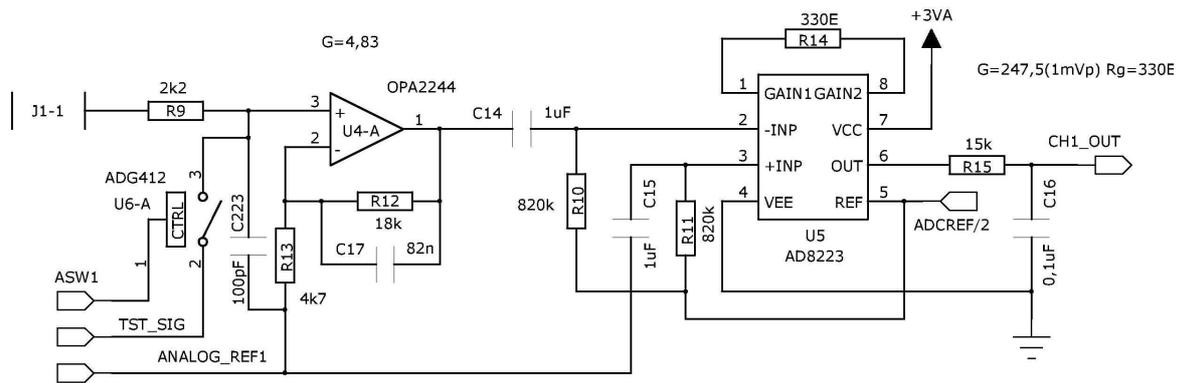


Bild 3.2 Ausgeführte Schaltung referenzieller EEG-Kanal

Der Kombination R9 mit C223 stellt zum einen einen Tiefpass dar, der hochfrequente Störungen unterdrücken soll und gleichzeitig die Abstrahlung von HF-Energie auf ein zulässiges Maß reduziert, und zum anderen einen Eingangsschutz gegen elektrostatische Entladungen bildet. Für den Eingangsschutz gibt es auch andere Varianten, die zumeist mit 2 zusätzlichen Schutzdioden gegen GND und Versorgung aufgebaut sind. Dies würde eventuell einen besseren Schutz des Eingangsverstärkers U4-A und des Schalters U6-A bieten, haben jedoch den Nachteil, dass die Leckströme der Dioden ein zusätzliches Rauschen am Eingang verursachen. Weiters scheint diese Variante auch nicht notwendig, weil beim mittlerweile seit 10 Jahren auf dem Markt befindlichen 32 Kanal Verstärker, der in sehr ähnlicher Weise aufgebaut wurde, noch nie ein Eingangsverstärker defekt wurde, und das bei mehr als 200 Einheiten, die praktisch tagtäglich im Einsatz sind. Die Tiefpässe R12/C17 und R15/C16 sorgen dafür, dass höherfrequente Spannungen, die nicht im EEG-Band liegen, entsprechend gedämpft werden. C16 hat zusätzlich die Aufgabe, die Ausgangsimpedanz der Schaltung für den nachfolgenden ADC im hochfrequenten Bereich niedrig zu halten. Der ADC arbeitet mit einem switched capacitor Verfahren, das prinzipbedingt impulsartige Belastungen der Messspannung zur Folge hat, die zu einem Messfehler führen können. Daher sind entsprechende Kapazitäten an den ADC Eingängen vorzusehen, damit der Messfehler idealerweise unter 1/10 LSB bleibt.

Die Aussteuergrenzen des Gesamtverstärkers sind ebenfalls zu ermitteln, um die maximal verarbeitbaren Eingangsamplituden angeben zu können. Zunächst wird der Eingangsverstärker OPA2244 betrachtet. Laut DB sind am Eingang max. 2,1V bzw. min. 0V erlaubt. Der Ausgang des OPV kann sich (garantiert) zwischen 2,1V und 0,5V bewegen. Der virtuelle GND und auch die Ausgangsspannung des OPV bei  $U_{\text{Diff}} = 0\text{V}$  liegt auf 1,2V.

Damit ergibt sich ein Spannungshub von  $2,1\text{V} - 1,2\text{V} = 0,9\text{V}$  in positiver sowie  $1,2\text{V} - 0,5\text{V} = 0,7\text{V}$  in negativer Richtung. Die kleinere dieser Spannungen ist durch den Verstärkungsfaktor 4,83 zu dividieren, um die maximale DC Differenzeingangsspannung zu bestimmen. Damit erhält man:

$$U_{\text{Diff DCmax}} = \pm 145\text{mV}$$

Die eigentliche AC Signalspannung (EEG-Signal), die laut Spezifikation innerhalb von  $\pm 1\text{mV}$  liegen darf, spielt hier für die Aussteuerung kaum eine Rolle.

Für die Gleichtaktaussteuerung wurde weiter oben bereits ein Verstärkungsfaktor von 1 für die Eingangsstufe berechnet. Damit entspricht die maximale Gleichtakteingangsspannung dieser Stufe dem möglichen Spannungshub des OPV am Ausgang.

$$U_{\text{Glmax (OPA2244)}} = \pm 700\text{mV}$$

Dies könnte aber an den Eingängen des nachfolgenden AD8223 einen zu hohen Spannungshub bewirken. Es sind 2 Kriterien zu überprüfen:

- 1.) max. Eingangsspannungsbereich
- 2.) interne Übersteuerung des Instrumentenverstärkers

Aus dem DB entnimmt man, dass maximal  $1,5\text{V}$  bzw. minimal  $-150\text{mV}$  an den Eingängen erlaubt sind. Damit reduziert sich hierbei die maximal erlaubte Gleichtakteingangsspannung auf  $1,5\text{V} - 1,2\text{V} = 300\text{mV}$

Für das 2. Kriterium ist laut DB folgende Bedingung einzuhalten:

$$-V_s + 0,01\text{V} < 0,6\text{V} + V_{\text{CM}} \pm (V_{\text{Diff}} * \text{Gain})/10 < +V_s - 0,1\text{V}$$

$$-V_s = 0\text{V}$$

$$+V_s = 3\text{V}$$

$$V_{\text{Diff}} = 4,83\text{mV}$$

$$\text{Gain} = 247,5$$

$$0,01\text{V} < 0,6\text{V} + V_{\text{CM}} \pm 0,1195 < 2,9\text{V}$$

bzw.

$$0,01\text{V} < 0,48\text{V} + V_{\text{CM}} < 2,9\text{V}$$

$$0,01\text{V} < 0,72\text{V} + V_{\text{CM}} < 2,9\text{V}$$

Somit ergibt sich  $V_{\text{CM min}} = -470\text{mV}$  und  $V_{\text{CM max}} = 2,18\text{V}$  bezogen auf GND (0V).

Es ist aber der virtuelle GND mit  $1,2\text{V}$  überlagert, daher errechnet sich

$$V_{\text{CM min}} = -1,67\text{V} \text{ und } V_{\text{CM max}} = 0,98\text{V}$$

Man sieht, dass das 1. Kriterium maßgeblich ist und somit endgültig:

$$U_{\text{Gl max}} = \pm 300\text{mV}$$

Für den Ausgangsspannungshub wird die Last mit  $100\text{k}\Omega$  angenommen, weil bei  $10\text{Hz}$  der  $0,1\mu\text{F}$  Kondensator am Ausgang eine  $166\text{k}\Omega$  Last darstellt.

Laut DB erreicht dabei die Ausgangsspannung minimal  $0,1\text{V}$  und maximal  $2,7\text{V}$ .

Damit ergibt sich in Bezug zur ADC Referenzspannung von  $2,4\text{V}$  eine Einschränkung im Spannungsbereich um etwa  $4\%$ , was durchaus tolerierbar ist.

Abschließend soll auch noch das Rauschverhalten des Verstärkers betrachtet werden. Der Frequenzbereich erstreckt sich laut Spezifikation von 0,3 bis 70Hz. Damit liegt dieser im Bereich des 1/f Rauschens, womit eine Integration über diesen Frequenzbereich notwendig wäre, jedoch ist es ausreichend, einen Mittelwert der Rauschgröße aus dem Diagramm zu bestimmen, und diesen dann mit der Wurzel des Frequenzbereichs zu multiplizieren.

Zunächst wird das Spannungsrauschen betrachtet, wobei die Berechnungen auf Peak to Peak Werte ausgerichtet sind.

Laut DB weist der OPA2244 im Frequenzbereich 0,1-10Hz ein Rauschen von  $0,4\mu\text{Vpp}$  auf. Für den Bereich 10Hz bis 70Hz entnimmt man dem Diagramm im DB etwa  $30\text{nV}/\sqrt{\text{Hz}}$ .

Damit ergibt sich für diesen Bereich  $30\text{nV} * \sqrt{60} = 232\text{nVeff}$

Mit einem Umrechnungsfaktor von 7 ergibt sich für die Peak to Peak Spannung etwa  $1,6\mu\text{Vpp}$ . Die gesamte Rauschspannung dieser Stufe ergibt sich durch eine quadratische Addition der beiden Bereiche:

$$U_{r(\text{OPA2244})} = \sqrt{0,4^2 + 1,6^2} = 1,65\mu\text{Vpp}$$

Man sieht, dass der niederfrequente Bereich von 0,1-10Hz nur einen sehr geringen Teil zum Gesamtrauschen beiträgt.

Für das Stromrauschen betrachte ich deshalb nur den Bereich von 1-70Hz, der im Diagramm des DB ersichtlich ist.

Man entnimmt dem Diagramm einen Mittelwert von ca.  $100\text{fA}/\sqrt{\text{Hz}}$

Damit ist  $I_r = 100\text{fA} * \sqrt{69} = 830\text{fAeff} \Rightarrow 5,8\text{pApp}$

An einer Signalimpedanz von angenommen  $10\text{k}\Omega$  entsteht dadurch ein Spannungsrauschen von  $58\text{nVpp}$ . Man sieht, dass dieser Wert doch gut eine Zehnerpotenz kleiner ist als das reine Spannungsrauschen und deshalb vernachlässigt werden kann.

Ebenfalls zu vernachlässigen sind die Rauschspannungen der Gegenkopplungs-Widerstände, die praktisch keinen Beitrag zum Gesamtrauschen liefern, da diese relativ niederohmig dimensioniert wurden.

Für die 2. Stufe mit dem AD8223 entnimmt man dem DB etwa  $35\text{nV}/\sqrt{\text{Hz}}$ .

Damit ergibt sich

$$U_{r(\text{AD8223})} = 35\text{nV} * \sqrt{69} = 291\text{nVeff} \Rightarrow 2\mu\text{Vpp}$$

Dieser Wert muss aber noch auf den Signal - Eingang bezogen werden, daher muss durch die Verstärkung der Eingangsstufe (4,83) dividiert werden und erhält damit  $0,42\mu\text{Vpp}$ .

Das gesamte Spannungsrauschen des Verstärkerkanals ist somit:

$$U_{\text{rges}} = \sqrt{0,42^2 + 1,65^2} = 1,7\mu\text{Vpp}$$

In der Schaltung finden sich auch noch die 2 Tiefpässe erster Ordnung, gebildet durch R12/C17 und R15/C16. Diese sollen so dimensioniert sein, dass eine Grenzfrequenz von 70Hz erreicht wird. Dabei ist R12 mit  $18\text{k}\Omega$  vorgegeben, weil damit die Verstärkung der 1.Stufe eingestellt wird. Es ist zwar grundsätzlich möglich, mit 2 verschiedenen Grenzfrequenzen der 2 Tiefpässe, die natürlich höher als die 70Hz sein müssen, ein Filter 2.Ordnung mit kritischer Dämpfung zu bilden, allerdings ist der Rechenaufwand sehr

hoch. Macht man die Grenzfrequenzen der 2 Tiefpässe gleich, so erzielt man erstens den steilsten Übergangsbereich und zweitens erleichtert das sehr die Dimensionierung der RC Glieder. In diesem Fall gilt nämlich:

$$f_{g(\text{Einzelfilter})} = \frac{1}{\sqrt{\sqrt{2}-1}} * f_{g(\text{gesamt})} = 1,55 f_{g(\text{gesamt})} \quad (1)$$

Damit wird  $f_{g(\text{Einzelfilter})} = 1,55 * 70\text{Hz} = 109\text{Hz}$

Die Berechnung von C17 erfolgt folgendermaßen:

$$C17 = \frac{1}{2\pi f_{gR12}} = 81\text{nF}$$

Für den 2. Tiefpass wird C16 mit 100nF vorgegeben, damit errechnet sich R15 mit:

$$R15 = \frac{1}{2\pi f_{gC16}} = 14,7\text{kOhm}$$

Verwendet wurden dann die nächsten Normwerte der E12 Reihe mit 82nF und 15kOhm.

Über den Analog-Schalter U6-A kann noch ein Testsignal, das entweder ein sinusförmiger Konstantstrom für Elektroden-Impedanzmessungen oder eine sinusförmige Spannung für Kalibrierzwecke sein kann, eingespeist werden. Die Ansteuerung des Schalters erfolgt direkt von einem CPU-Pin.

<sup>(1)</sup> Tietze/Schenk Halbleiterschaltungstechnik, Springer1986, Kap14, S377

### 3.1.1.2 Differenzielle Eingangverstärker

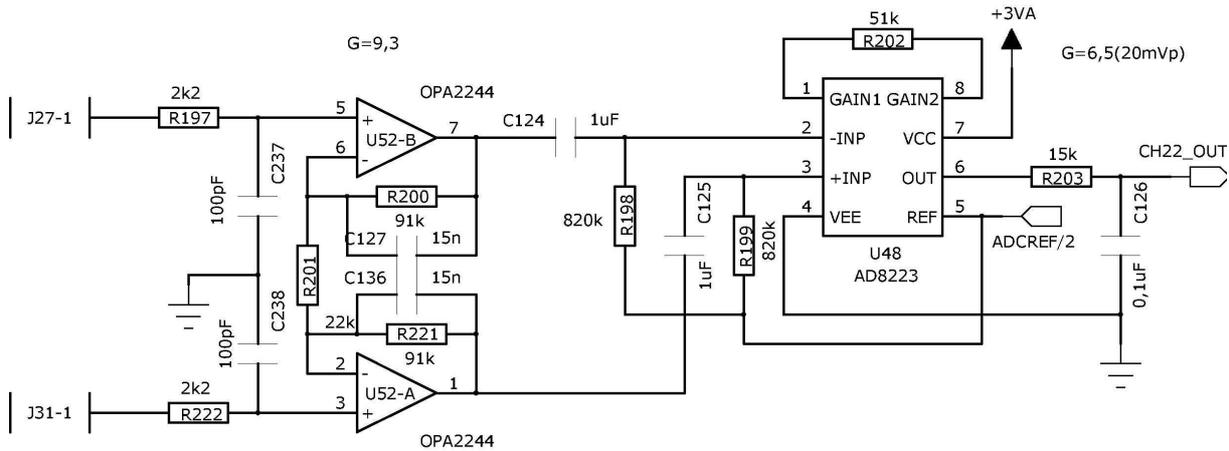


Bild 3.3 Ausgeführter differenzieller Verstärkerkanal

Vom Prinzip her handelt es sich hier um 2 hintereinander geschaltete Instrumentenverstärker, wobei in der ersten Stufe der Differenzbilder fehlt, und die zweite Stufe über einen Hochpass angekoppelt ist.

Für die Berechnung der Übertragungsfunktion werden die Kondensatoren C127 / C136 vorerst nicht berücksichtigt, diese bilden hier mit den Gegenkopplungswiderständen R200 / R221 einen Tiefpass erster Ordnung, der höherfrequente, außerhalb des EEG – Bandes liegende Anteile des Signals dämpfen soll.

Die Spannungen an den OPV Anschlüssen werden folgendermaßen bezeichnet:

Pin5:  $U_{e1}$

Pin7:  $U_{a1}$

Pin3:  $U_{e2}$

Pin1:  $U_{a2}$

Weiters wird ein idealer OPV angenommen mit  $A=\infty$  und  $R_e=\infty$ .

Durch den Widerstand R201 muss dann der Strom  $I_q = (U_{e1}-U_{e2})/R201$  fließen. Der gleiche Strom muss auch durch die Widerstände R200 und R221 fließen.

Damit gilt:

$$(U_{a1} - U_{e1})/R200 = (U_{e1}-U_{e2})/R201$$

$$(U_{e2} - U_{a2})/R221 = (U_{e1}-U_{e2})/R201$$

wobei  $R200 = R221$  ist.

Durch Addition der 2 Gleichungen ergibt sich :

$$(U_{a1}-U_{a2} +U_{e2}-U_{e1})/R200 = 2*(U_{e1}- U_{e2})/R201$$

Durch eine weitere Umformung erhält man :

$$U_{a1}-U_{a2} = (U_{e1}-U_{e2}) \cdot (2 \cdot R_{200}/R_{201} + 1)$$

Oder:

$$\frac{U_{aD}}{U_{eD}} = A = \frac{2R_{200}}{R_{201}} + 1$$

Und somit den bekannten Verstärkungsfaktor für die Instrumentenverstärkerschaltung.

Die 2. Stufe wurde so dimensioniert, dass eine maximale Eingangsamplitude von +/-20mV auf +/- 1,2V verstärkt wird. Damit liegt der Gesamtverstärkungsfaktor bei 60. Dividiert durch die Verstärkung der 1. Stufe mit 9,3 ergibt sich dann der Verstärkungsfaktor 6,5 für die 2. Stufe. Laut DB muss dann ein Widerstand mit 51kOhm an die Gain-Anschlüsse des AD8223.

Für die Hochpässe R198/C124 und R199/C125 gelten dieselben Überlegungen und Dimensionierungen wie beim referenziellen Verstärkerkanal.

In der Schaltung finden sich auch noch die 2 Tiefpässe erster Ordnung, gebildet durch R200/C127 und R203/C126. Diese sollen wieder so dimensioniert sein, dass eine Grenzfrequenz von 70Hz erreicht wird. Dabei ist R200 mit 91kOhm vorgegeben, weil damit die Verstärkung der 1.Stufe eingestellt wird. Die Überlegungen sind analog zu denen beim referenziellen Verstärker.

$$\text{Damit wird wieder } f_{g(\text{Einzelfilter})} = 1,55 \cdot 70\text{Hz} = 109\text{Hz}$$

Der Kondensator C127 berechnet sich dann wie folgt:

$$C_{127} = \frac{1}{2\pi f_g R_{200}} = 16\text{nF}$$

Beim 2.Tiefpass wird C126 mit 100nF vorgegeben, damit :

$$R_{203} = \frac{1}{2\pi f_g C_{126}} = 14,7\text{kOhm}$$

Der verwendete Wert für C127 ist 15nF und für R203 15kOhm.

Da die differenziellen Kanäle hauptsächlich für externe Sensoren (Lagesensor, Atemgurt, usw..) verwendet werden, wurde auf die Möglichkeit einer Impedanzmessung bzw. Kalibrierspannung verzichtet. Dies auch im Hinblick auf ein möglichst kompaktes Design des Verstärkers.

### 3.1.1.3 Referenz Spannungserzeugung

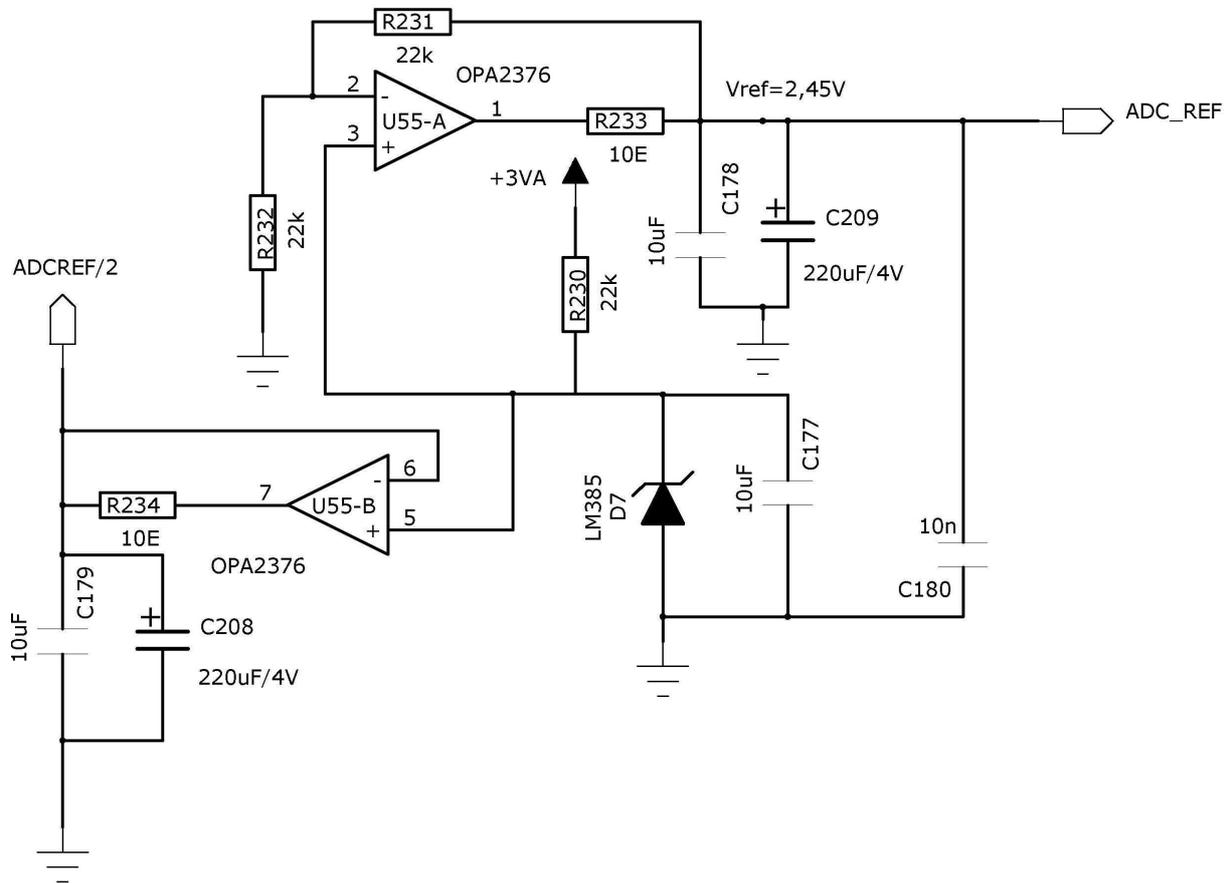


Bild 3.4 Referenzspannungserzeugung

Wie in den Konzeptüberlegungen bereits erwähnt, wird die gesamte Schaltung mit +3V versorgt. Da das analoge Eingangssignal sowohl positive als auch negative Werte annehmen kann, ist es notwendig einen so genannten virtuellen GND zu erzeugen. Weiters ist auch noch eine Referenzspannung für den ADC zur Verfügung zu stellen. Die Konstant – Spannung für den virtuellen GND wird von einer Bandgapreferenz LM385 (D7) erzeugt und beträgt ca. 1,22V. Dazu wird durch die Referenzdiode ein Strom von

$$(3V-1,22V)/22k\Omega = 81\mu A$$

getrieben.

Der C177 bildet mit dem R230 einen Tiefpass mit der Grenzfrequenz:

$$f_g = \frac{1}{2\pi R_{230} C_{177}} = 0,72 Hz$$

Dies dient vornehmlich zur Reduzierung von Rauschspannungen.

Um diese Spannung belastbarer zu machen, wird noch ein OPV als Spannungsfolger nachgeschaltet. Die Kondensatoren C179 und C208 sollen die vom LM385 und dem OPV erzeugten Rauschspannungen nochmals dämpfen und für eine niedrige Impedanz des virtuellen GND auch im höherfrequenten Bereich sorgen.

Diese Konfiguration neigt generell dazu, ein schwingfähiges System zu bilden, weil die rückgekoppelte Spannung eine nacheilende Phase besitzt und zusammen mit der Phasennacheilung des OPV aus einer Gegenkoppelung eine Mitkoppelung wird. Man kann das Schwingen verhindern, indem man einen ausreichend großen und damit niederimpedanten Kondensator an den Ausgang des OPV anschließt. Dies ist, wie in obiger Schaltung ersichtlich, mit C208 und C209 so ausgeführt worden.

Durch die Serienwiderstände R234 und R233, die nicht größer als ca. 100Ω sein sollen, hat man die Möglichkeit, mit einem Oszilloskop direkt am Ausgang des OPV die Wirksamkeit obiger Maßnahme zu überprüfen.

Für die Referenzspannung des ADC wird die Bandgap – Spannung mit Hilfe eines nicht invertierend beschalteten OPV auf ca. 2,45V verdoppelt. Eine präzise Spannung ist nicht erforderlich, weil die Messkanäle ohnehin alle einzeln kalibriert werden, und eine Amplitudenabweichung nach der A/D Wandlung sehr einfach durch eine Multiplikation mit einem Kalibrierfaktor ausgeglichen werden kann. Weiters wird auch keine allzu große Präzision bezüglich der Amplitude gefordert, nur sollten die Kanäle alle möglichst gleich sein.

Eine gute Temperaturstabilität sollte allerdings gegeben sein, und wird auch in diesem Fall mit 50ppm/K erfüllt.

Für den OPV kann der hier sonst beliebte OPA2244 nicht verwendet werden, weil dieser die 2,45V Ausgangsspannung nicht erreicht. Es wurde dafür der OPA2376 eingesetzt, ein Rail to Rail Präzisionsverstärker mit moderater Stromaufnahme.

### 3.1.1.4 Kalibrier und Impedanzmesssignal

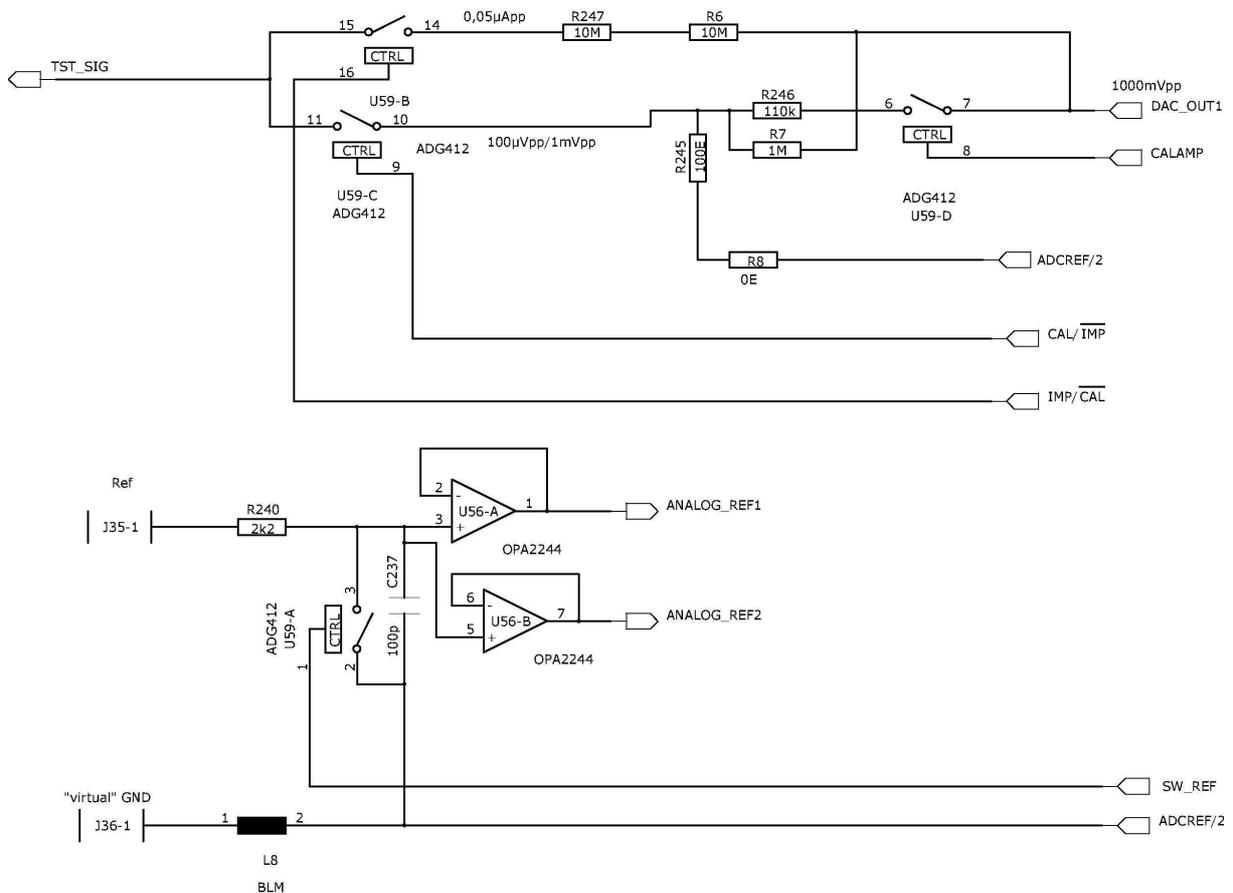


Bild 3.5 Kalibrierspannung, Impedanzmessung, Referenzverstärker

Wie oben ersichtlich wird vom DAC der CPU eine sinusförmige Spannung mit der Frequenz von 10Hz und der Amplitude von 1Vpp erzeugt.

Für die Impedanzmessung an der Elektrode wird der Schalter U59-B und U59-A aktiviert.

Damit ergibt sich für das Test-Signal TST\_SIG ein weitgehend konstanter Messstrom von  $0,05\mu\text{A}$ . Dieser wird in die Elektrode eingepreßt und verursacht je nach Übergangswiderstand einen Spannungsabfall, der mit dem jeweiligen Verstärkerkanal gemessen und ausgewertet wird. Der Messbereich beträgt hier  $20\text{k}\Omega$ , entsprechend der maximalen Eingangsamplitude von  $1\text{mV}$ .

Für die Kalibrierspannung wird der Schalter U59-C geschlossen, womit die Kalibrierspannung, deren Amplitude noch mit dem Schalter U59-D auf  $1\text{mVpp}$  oder  $100\mu\text{Vpp}$  eingestellt werden kann, über das TST\_SIG Netz an die Verstärkereingänge gelangt. Der Eingang des Referenzverstärkers muss dabei mit dem Schalter U59-A auf das fixe Potential ADCREF/2 bzw. virtuell GND gelegt werden.

Die Ferrit-Perle L8 bewirkt eine bessere EMV, weil durch die hohe Impedanz bei hohen Frequenzen nur wenig HF Energie über den virtual GND Anschluss abgestrahlt werden kann.

### 3.1.2 Mikrocontroller und angeschlossene Einheiten

#### 3.1.2.1 Beschaltung des Mikrocontrollers

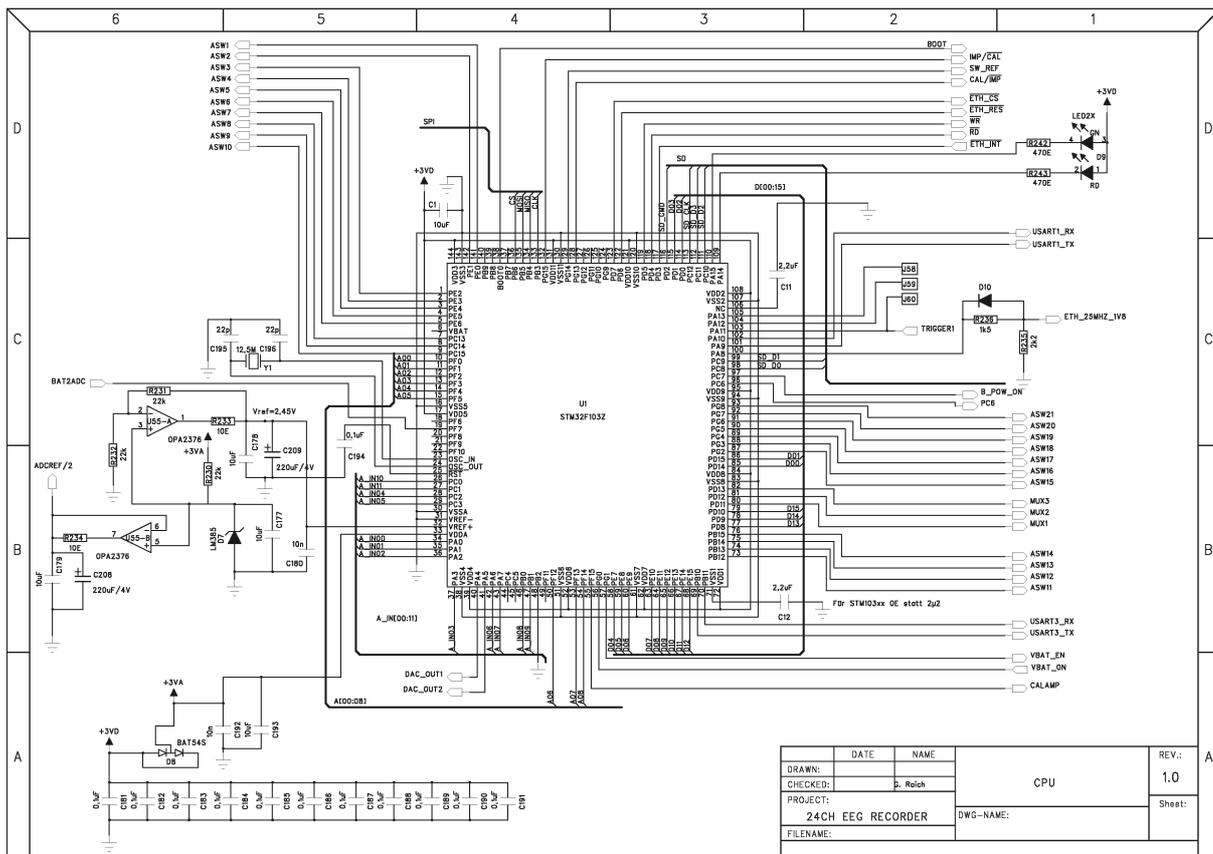


Bild 3.6 CPU Beschaltung

Als Mikrocontroller kommt der Typ STM32F103ZET6 zum Einsatz. Dieser hat auch noch ein verarbeitungsfreundliches LQFP144 Gehäuse, das allerdings mit einer 20x20mm Abmessung schon relativ groß ist. Es gibt denselben Controller auch im BGA Gehäuse mit 144 Anschlüssen, das gerade einmal ¼ der Größe aufweist. Das Layout wäre allerdings recht aufwändig, und würde beim Leiterplattenhersteller aufgrund der feinen Strukturen nicht mehr als Standard gefertigt werden können. Zudem würde man mindestens eine 6 Lagen Leiterplatte benötigen. Auch die Inbetriebnahme von Schaltungen mit BGA Bauteilen kann zu großen Problemen führen, da ja nicht mehr direkt an den Anschlüssen gemessen werden kann, und die Lötstellen für eine optische Inspektion oder Nacharbeit unzugänglich sind.

In obiger CPU-Beschaltung werden Leitungen funktional Bussen zugeordnet. Die physischen Anschluss Stellen der Leitungen werden dabei dem DB unter Kapitel 3 „Pinouts and Pin descriptions“ entnommen.

Für die Verbindung mit dem Ethernetbaustein W5300 werden folgende Busse und Leitungen benötigt:

Datenbus:	D00-D15
Adressbus:	A00-A08
Chipselect:	ETH_CS
Read:	RD
Write:	WR
Reset-Ltg.:	ETH_RES
Interr.-Ltg.:	ETH_INT
Clock-Ltg.:	ETH_25MHZ_1V8

Die 25MHz Clock des Ethernetbausteins wird von der CPU zur Verfügung gestellt, weil es die Möglichkeit gibt, einen Pin der CPU so zu konfigurieren, dass die interne CPU Clock, oder ganzzahlige Teile davon, ausgegeben wird. Die CPU wird mit einem externen 12,5MHz Quarz betrieben, durch eine einstellbare, interne PLL wird diese Frequenz auf 50MHz erhöht und bildet die interne CPU-Clock. Durch einen „Prescaler“ wird diese Frequenz durch 2 dividiert und am Pin PA8 ausgegeben. Da die CPU mit einer Spannung von 3V betrieben wird, braucht es noch die Bauteilgruppe D10/R236/R235, um den Ausgang der CPU an den CLK-Eingang des Ethernetbausteins mit 1,8V Pegel anzupassen.

Der Grund für diese Beschaltung liegt darin, nur eine hochfrequente Clock zu haben, und nicht mehrere, die mit annähernd gleicher Frequenz laufen und damit alle möglichen Interferenz und EMV – Probleme hervorrufen könnten.

Für die Verbindung mit der SD-Speicherkarte wird das SDIO Interface mit 4 Datenleitungen und 2 Steuerleitungen verwendet:

Datenbus:	SD_D0 – SD_D3
Clock-Ltg.:	SD_CLK
Steuer-Ltg.:	SD_CMD

Für die Anbindung des Beschleunigungssensors ist ein SPI Bus mit folgenden Leitungen zuständig:

Chipselect:	CS
Daten-Ltg. Master:	MOSI
Daten-Ltg. Slave:	MISO
Clock-Ltg.:	CLK

Für die serielle Kommunikation nach aussen, die vor allem für Softwaredownload und Debug – Zwecke gebraucht wird, wird der USART1 verwendet, für die serielle Kommunikation mit dem optionalen Chipoximeter – Modul wird USART3 verwendet.

Für den Softwaredownload wird beim Hochfahren der Versorgungsspannung die Leitung BOOT auf high-Pegel gesetzt, was intern die CPU in den Boot – Modus versetzt und damit den Bootloader, der im System-Memory fix programmiert ist, startet.

Die Analogspannungen von den Verstärkerkanälen werden über einen Multiplexer und dann zum ADC des Controllers geführt. Diese 12 Leitungen sind als Bus A\_IN00 – A\_IN11 gezeichnet.

Die durch einen Spannungsteiler und Operationsverstärker konfigurierte Batteriespannung BAT2ADC wird direkt an einen ADC-Eingang des Controllers geführt.

Für die Aufschaltung des Testsignals auf die analogen Verstärkereingänge werden die Steuerleitungen ASW1 – ASW21 verwendet.

Für die Umschaltung der Analogsignal – Eingangsmultiplexer sind die Steuerleitungen MUX1 – MUX3 zuständig.

Die Betriebsspannung von 3V wird an 11 Stellen des CPU Gehäuses angelegt. Zur sauberen Entkopplung sind hier überall 100nF Stützkondensatoren, die möglichst nah am Gehäuse platziert werden um ihre volle Wirkung zu erzielen, vorgesehen. Weiters werden auch noch einige 10µF Keramikkondensatoren eingesetzt, die für zusätzliche Stabilität der Betriebsspannung sorgen. Diese Parallelschaltungen sind deshalb sinnvoll, weil die 100nF Kondensatoren eine weitaus höhere Resonanzfrequenz besitzen als die 10µF Typen. Letztere haben aber dafür im niederfrequenten Bereich eine sehr niedrige Impedanz. Bei CPU-Clocks von über 100MHz wären zusätzliche 1nF COG Kondensatoren im 0402 Gehäuse eine gute Wahl für die Entkopplung, aber in diesem Fall nicht notwendig.

Wie bereits in der Konzepterstellung angemerkt, wurden für den Analog und Digitalteil der Schaltung separate Stromversorgungen angedacht. So ist auch die analoge Betriebsspannung der CPU (VDDA), die hauptsächlich den A/D und D/A Konverter versorgt, und die digitale Betriebsspannung VDD1-VDD11 nicht direkt miteinander verbunden, sondern werden von separaten Schaltreglern gespeist.

Liest man das DB aufmerksam, so findet man im Kapitel 5.3.1 unter „general operating conditions“ einen Hinweis, dass die 2 Betriebsspannungen in keinem Betriebszustand mehr als 300mV voneinander abweichen dürfen, und überhaupt wird einem ans Herz gelegt („recommended“), dieselbe Betriebsspannung zu verwenden.

Um die obige Bedingung zu erfüllen, wurde eine Doppel-Schottky Diode (BAT54S, D8) antiparallel zwischen die Betriebsspannungsanschlüsse geschaltet.

Warum allerdings der Hersteller hier die „schlechte“ Lösung propagiert, bleibt wohl dessen Geheimnis, möglicherweise hat er selbst kein Vertrauen in sein Produkt. Jedenfalls ist es bis jetzt bei 25 in Betrieb befindlichen Exemplaren zu keinen Problemen mit der Stromversorgung gekommen.

Die CPU besitzt weiters auch eine interne Reset-Schaltung, damit braucht es nur noch einen 100nF Kondensator (C194) gegen GND damit die CPU ordnungsgemäß hochläuft.

Auch eine JTAG Schnittstelle zum download von Software und für Debugzwecke wäre grundsätzlich vorhanden, allerdings müssten dann einige Pins nur für diesen Zweck verwendet werden, was zu zusätzlichen Problemen geführt hätte. Außerdem erfolgt im gegenständlichen Fall der Softwaredownload über die USART1 Schnittstelle, und den Vorteil einer Debug-Schnittstelle sehe ich nicht. Für die Analyse von Softwareproblemen hat sich die Ausgabe von Variablen über die serielle Schnittstelle sehr bewährt, weil der Eingriff in den Programmablauf sehr gering ist. Ein weiteres Hilfsmittel stellen freie CPU-Pin's ( PA11-PA13) dar, die von der Software gesetzt bzw. rückgesetzt werden können und an denen mit einem Oszilloskop gemessen werden kann. Damit können Ausführungszeiten von bestimmten Funktionen oder Auslastungen sehr leicht gemessen werden, und das praktisch ohne Eingriff in das Programm.

Das Setzen eines Breakpoints mit Hilfe eines Debug-Adapters hält zwar die Programmausführung an einem bestimmten Punkt des Programms an, im Hintergrund gibt es aber massenweise Interrupts die noch immer laufen, aber nicht mehr abgearbeitet werden. Daher liefern solche Geräte gerne ein falsches Bild der wahren Zustände im Prozessor, beziehungsweise ist die Interpretation der Daten oft äußerst schwierig.

Ich bin jedenfalls die letzten 25 Jahre ohne Debugger sehr gut zurechtgekommen, die Softwarefehler findet man meiner Meinung nach nur durch nachdenken, und nicht automatisch durch anschließen eines Debug-Kästchens.

An den Pin's PA14 und PA15 ist eine 2 färbige LED (Status LED) angeschlossen, die eine Reihe von Betriebszuständen, durch Blinken mit unterschiedlichen Frequenzen und Farben, anzeigen kann.

### 3.1.2.2 Ethernet

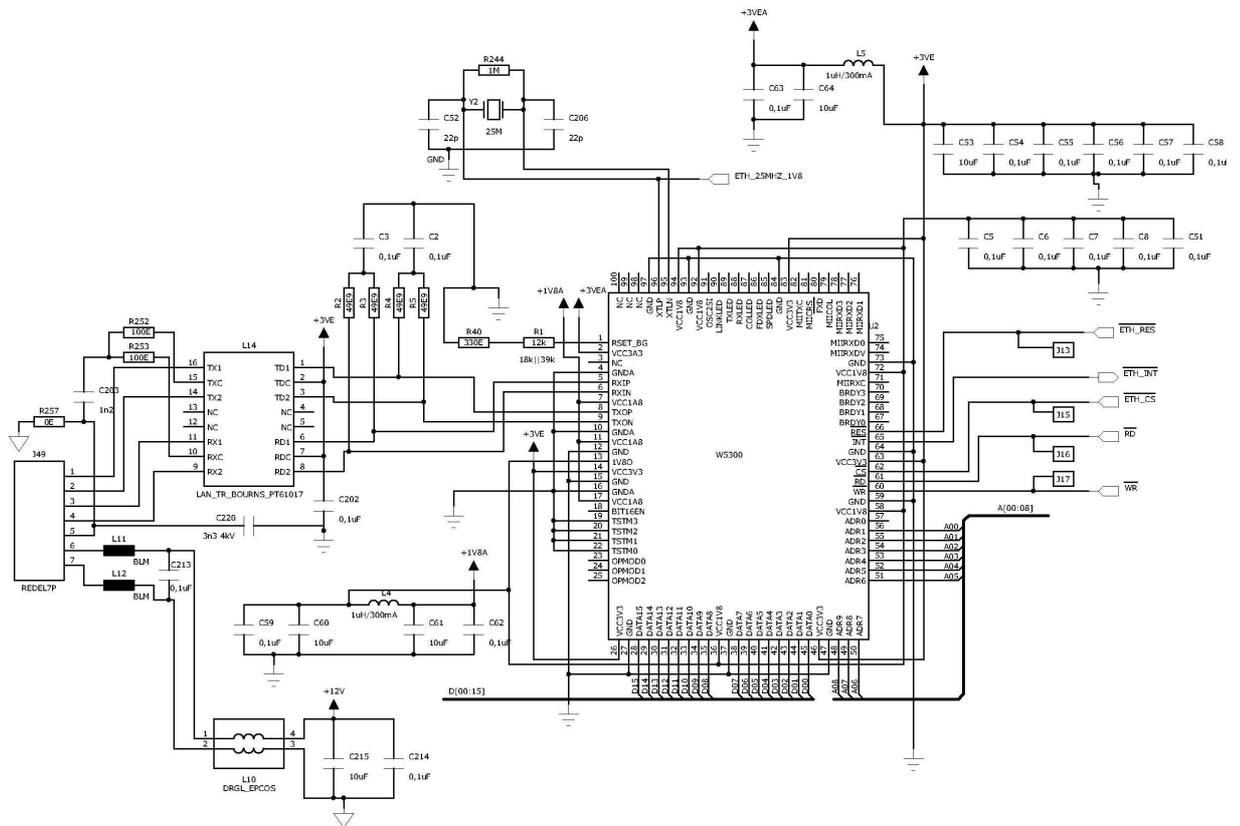


Bild 3.7 Ethernet Beschaltung

Für die Kommunikation über das Ethernet wurde der Baustein W5300 vom Hersteller Wiznet verwendet. Dieses Teil bietet einen 100MHz PHY, einen MAC und einen fix einprogrammierten TCP/IP Stack. Durch letzteres erspart man sich einen eigenen TCP/IP Stack, der dann auf dem Mikrocontroller laufen würde, und relativ viel Rechenleistung benötigt.

Der Stecker J49 (ganz links i. Bild 3.7) ist der kombinierte Versorgungs und Datenstecker des Gerätes. Die 4 Ethernetleitungen (je 1 differenzielles Paar für Senden und Empfangen) werden zunächst über einen Übertrager (L14) geführt. Dieser stellt gleichzeitig die Potentialtrennung zwischen Patientenstromkreis und Zwischenstromkreis dar, und muss deshalb den Anforderungen an die Patientensicherheit genügen. Diese sind in diesem Fall:

$$U_{\text{trenn}} = 1500VAC_{\text{rms}} / 60s$$

$$L_{\text{trenn}} = 8mm$$

Auch der Kondensator C220 ist ein Sicherheitsbauteil, das obigen Anforderungen genügen muss. Elektrisch gesehen dient dieses Bauteil zum kurzschließen hochfrequenter Spannungen zwischen Patienten- und Zwischenstromkreis und stellt daher eine EMV Maßnahme dar.

Auf der sekundär Seite des Übertragers L14 werden die Datenleitungen mit 50 Ohm, floatend durch C2/C3, abgeschlossen.

Der im W5300 vorhandene PHY erzeugt bzw. empfängt die 100MHz Signalspannungen für das Ethernet. Der Stromverbrauch ist deshalb recht hoch und die Versorgungspins sind für diesen Teil des Bauteils extra herausgeführt und müssen gegenüber dem restlichen Teil gut entkoppelt werden.

Dies geschieht mit der Drossel / Kondensator Kombination L3/C63,C64 für die 3V und L4/C61,C62 für die 1,8V.

An dem Pin1 (RSET\_BG) wird der Arbeitspunkt der analogen Schaltungsteile im W5300 eingestellt, laut DB soll ein Widerstand von 12,3kOhm verwendet werden.

Der 25MHz Quarz Y2 samt R/C Beschaltung wird nicht bestückt, dies war für den Fall vorgesehen, dass die von der CPU generierte, externe Clock (ETH\_25MHz\_1V8) nicht richtig funktioniert hätte.

Der Anschluss von Daten und Adressbus sowie der Steuerleitungen ist selbsterklärend.

Auf die externe Versorgungsspannung des Gerätes kann an den Pins 6 und 7 des kombinierten Versorgungs und Datensteckers J49 zugegriffen werden. Zur Verbesserung der EMV sind die Elemente L11/L12/C213 und die Gleichtaktdrossel L10 vorgesehen.

C214 und C215 stabilisieren die Versorgungsspannung von etwa 12V im höherfrequenten Bereich für den nachfolgenden Schaltregler, der aber hier nicht eingezeichnet ist, sondern in Kapitel 3.1.4 näher beschrieben wird.

### 3.1.2.3 SD-Karte und Beschleunigungssensor

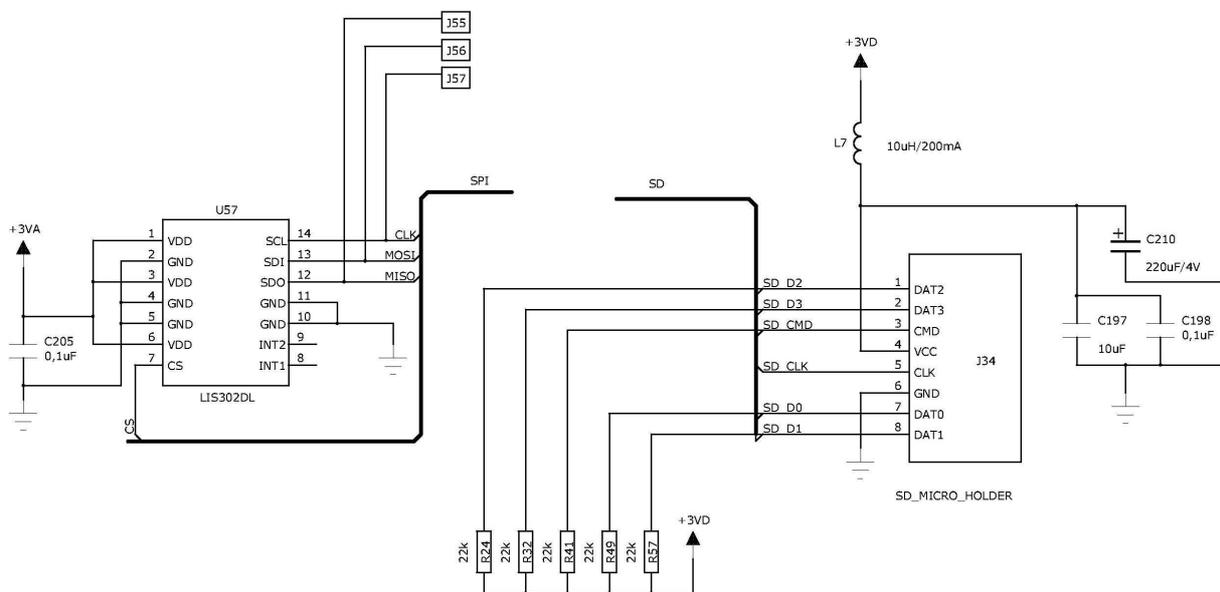


Bild 3.8 SD-Kartenanschluss und Beschleunigungssensor

Wie schon im Schaltungskonzept erwähnt, hat die SD-Karte einen, je nach Betriebsmodus, stark schwankenden Stromverbrauch. Vor allem beim Speichern der Daten steigt die Stromaufnahme auf ca. 70mA an, während im Lesebetrieb nur wenige mA gebraucht werden. Es schien daher sinnvoll, die Spannungsversorgung zu entkoppeln. Dies wurde durch die Kombination von

L7/C210/C197/C198 erreicht. Die Pull-up Widerstände an den Datenleitungen und an der CMD Leitung sind notwendig, weil es sonst Probleme bei der Initialisierung der SD Karte gibt. Wie aus der Schaltung ersichtlich, wird die Karte im 4 Bit Modus betrieben, wodurch der maximale Datendurchsatz erreicht wird. Dieser liegt mit dem verwendeten Controller bei rund 5 MByte/s.

Als Beschleunigungssensor wurde der LIS302 von STMicroelectronic eingesetzt. Dieses Teil ist recht universell einsetzbar, der Messbereich lässt sich auf +/-8g bzw. +/-2g über Registerbeschreibung einstellen. Im Sensor ist auch noch ein A/D Wandler integriert, der die Werte für X,Y,Z in entsprechende Register schreibt. Die Wandlungsrate kann auf 100 oder 400Hz eingestellt werden, was für die gegenständliche Anwendung mehr als ausreichend ist. Die Sensitivität beträgt im +/-2g Bereich ca. 18mg/Digit und der mögliche Offset 40mg. Letzteres ließe sich durch einen eingebauten Hochpass auf 0 reduzieren, jedoch könnte man dann die statische Lage des Patienten (Bauch, Rücken, Seitenlage) nicht mehr erkennen.

Die Anbindung an den Mikrocontroller erfolgt über die SPI Schnittstelle mit 4 Leitungen : MISO, MOSI, CLK, CS.

Das Gehäuse des Bauteils hat keine konventionelle Anschlussbeinchen, sondern plane Lötflächen an der Unterseite (LGA) und ist nur durch Lotpastenauftrag über eine Druckschablone verarbeitbar.

### 3.1.2.4 24Kanal Analog Multiplexer

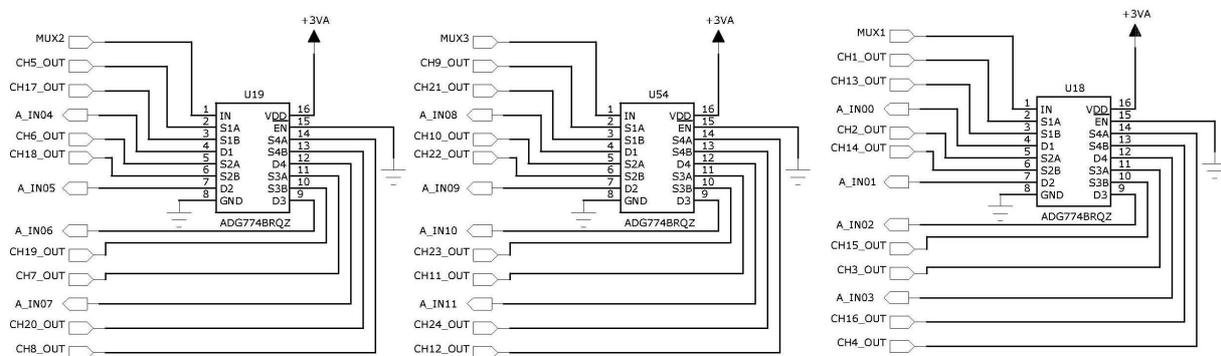


Bild 3.9 24Kanal Analog Multiplexer

Wie in der Schaltungskonzeptbeschreibung schon erwähnt, findet man keinen Mikrocontroller, der 24 A/D Eingänge besitzt. Daher wurden 3 Stück 4fach Wechselschalter vom Typ ADG774 eingesetzt, um die 24 Analogkanäle auf 12 A/D Wandlerkanäle zu bringen. Dieser Baustein hat einen sehr niedrigen on-Widerstand von nur 2,2 Ohm und Schaltzeiten von <10ns. Die Kondensatoren der Ausgangstiefpässe der Analogkanäle (Bild3.3) sind unmittelbar beim ADG774 platziert und dieser selbst möglichst nahe an den Analogeingängen des Mikrocontrollers, wodurch eine störungsarme Abtastung des Analogsignals gewährleistet ist. Ein kleines Problem stellt allerdings noch die „Charge-Injection“ dar, die aufgrund des niedrigen on-Widerstands mit 10pC doch gut doppelt so hoch ist, als bei den etwas höherohmigen Allzwecktypen wie z.B. ADG412. Trotzdem würden die 10pC an der 0,1uF Kapazität nur

einen Spannungssprung von  $100\mu\text{V}$  hervorrufen. Um sicher kein Problem entstehen zu lassen, muss einfach jeder einzelne Multiplexer im richtigen Zeitfenster, das heißt, dann, wenn der interne Multiplexer des A/D Wandlers nicht auf die Ausgänge des jeweiligen ADG744 zugreift, umgeschaltet werden (Steuersignale MUX1-3). Dies ist recht einfach in der Interruptroutine des A/D Wandlers zu bewerkstelligen.

### 3.1.3 Stromversorgung über Batterie

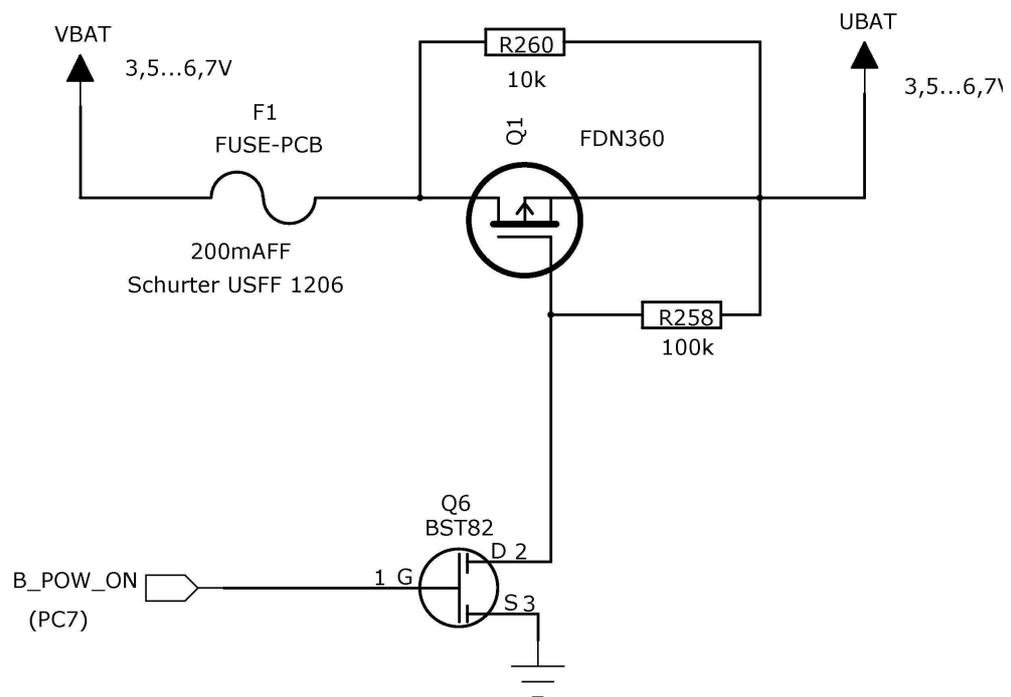


Bild 3.10 Batterie Schutzschaltung

Der Einsatz von Batterien oder Akkumulatoren zur Stromversorgung von elektromedizinischen Geräten wird von der Sicherheitsprüfstelle sehr argwöhnisch betrachtet. So darf es zu keiner Überhitzung oder gar Explosion der Energiespeicher kommen. Auch das zufällige Einprägen eines Ladestroms in die Batterie, infolge eines Fehlers, ist durch geeignete Maßnahmen zu verhindern.

Die Hitzeentwicklung ist generell nur durch eine zu hohe Stromentnahme oder Stromaufnahme im Fehlerfall möglich. Um dies zu verhindern wird die Schmelz-Sicherung F1 eingesetzt, die bei 200mA auslöst. Dieses Bauteil ist ein Sicherheitsbauteil, und darf nicht ohne weiteres durch einen anderen Typ ersetzt werden.

Die zweite Forderung, einen Stromfluss in die Batterie zu verhindern, wäre durch eine einfache Diode in Serie mit der Batterie leicht zu lösen. Der Spannungsabfall an dieser Diode hätte aber auch bei einer Schottky Ausführung gut 500mV betragen und damit die verfügbare Energie bzw. Laufzeit um mindestens 10% reduziert. Dies erschien nicht akzeptabel, und so wurde obige Lösung mit einem P-Kanal Mosfet realisiert. Ist der Mosfet ausgeschaltet, so verbleibt nur noch die Intrinsic Diode zwischen Drain und

Source, wobei die Kathode beim P-Kanal Typ auf Source liegt. Da Drain direkt über die Schmelzsicherung am +Pol der Batterie (VBAT) liegt, ist dies einmal die oben angeführte Diodenlösung. Der Widerstand R260 sorgt dafür, dass auch bei ausgeschaltetem Mosfet die Batteriespannung richtig, ohne wesentlichen Spannungsabfall gemessen werden kann. Der Widerstandswert von 10kOhm würde jetzt keinen „gefährlichen“ Strom in die Batterie zulassen. Wird das Gate des N-Kanal Mosfet Q6 auf high gesetzt, wird der P-Kanal Mosfet Q1 durchgeschaltet. Laut Datenblatt ist mit einem  $R_{on}$  von weniger als 0,3 Ohm zu rechnen und damit liegt der Spannungsabfall bei großzügig angenommenen 50mA Laststrom bei nur 15mV. Der Widerstand R258 sorgt dafür, dass Q1 sicher ausschaltet, wenn Q6 nicht durchgeschaltet ist. Da dieser Widerstand im Batteriebetrieb ständig von Strom durchflossen wird, ist er zur Erzielung höchster Effizienz hochohmig zu dimensionieren, allerdings auch mit Rücksicht auf die Drain-Source Leckströme des Q6.

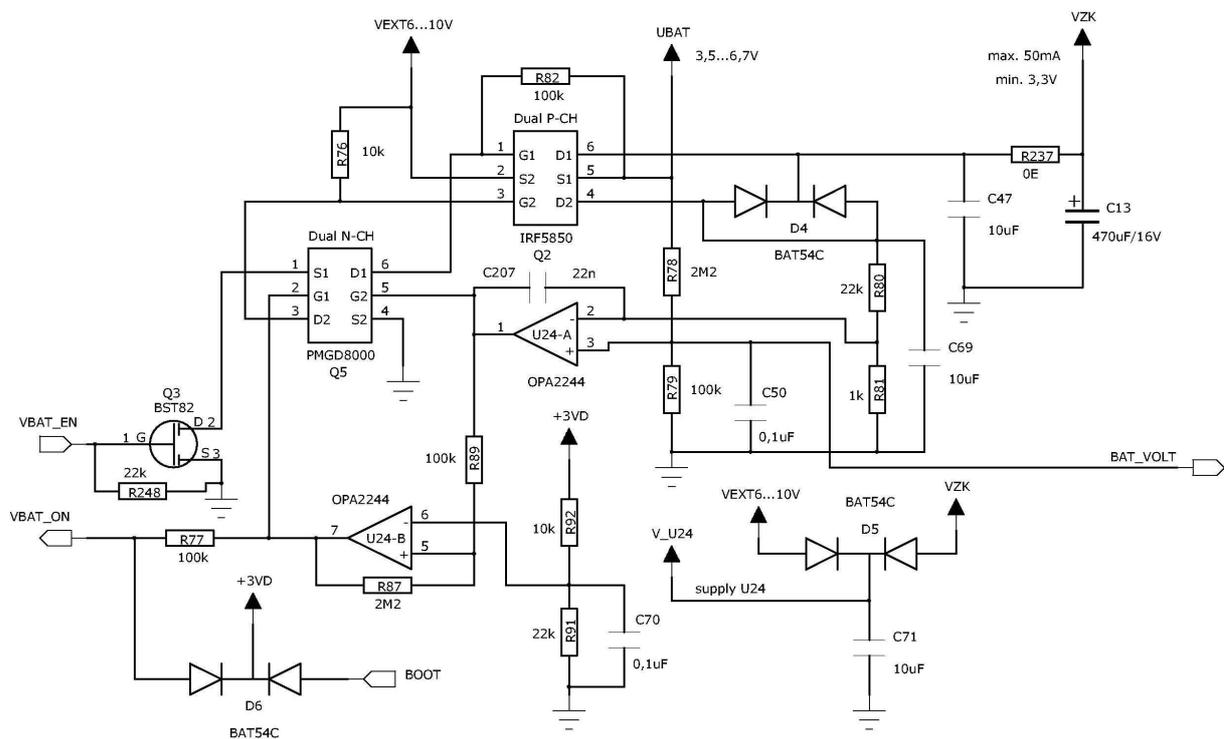


Bild 3.11 Stromquellenumschaltung und Zwischenkreis Spannungsregler

Funktionsweise der Stromquellenumschaltung:

Zunächst die Erklärung der Spannungssymbole:

VEXT6...10V externe Versorgungsspannung vom DC/DC Netzteil

UBAT: die von der Batterieschutzschaltung zur Verfügung gestellte

Batteriespannung

VZK: Ausgangsspannung der Schaltung, die bis auf ca. 200mV der Spannung UBAT entspricht, und die Eingangsspannung für die nachfolgenden Schaltregler zur Versorgung von CPU und Analoger Signalaufbereitung darstellt.

Das Hochfahren der Stromversorgung ist generell nur bei eingelegten Batterien möglich, wobei die Spannung UBAT zwischen 6,7 und 3,5V liegt, bzw. liegen muss. Der Steueranschluss VBAT\_EN liegt dabei auf LO. Wird der kombinierte Daten/Versorgungsstecker angesteckt, so fährt VEXT6..10V hoch. Die gemessene Spannung bei Normallast beträgt ca. 8V. Durch die Diode D5 wird der OPV U24 versorgt und ist damit aktiv. Am invertierenden Anschluss von U24-A liegt über die Diode D4 und den Spannungsteiler R80/R81 ein Teil der Ausgangs-Spannung VZK, die zu diesem Zeitpunkt noch 0 ist. Am nicht invertierenden Anschluss von U24-A liegt über den Spannungsteiler R78/R79 ein Teil der Batteriespannung. Dieser Spannungsteiler ist hochohmig zu dimensionieren, weil er ständig an der Batterie liegt und diese dadurch ständig entladen wird. In der gegenständlichen Dimensionierung fließen etwa 3µA, was unterhalb des Selbstentladungs-Stroms der Batterie liegt. Die Ausgangsspannung am Pin 1 von U24-A geht in die positive Sättigung und steuert G2 von Q5 an. Damit wird D2 von Q5 Richtung GND gezogen und damit auch G2 von Q2, ein Dual P-Kanal Leistungsmosfet. Dies bewirkt ein Durchschalten von S2/D2 von Q2 und damit auch der Spannung VEXT6..10V auf VZK.

Damit ist nun der Regelkreis, der die Spannung VZK der Batteriespannung UBAT nachführt, geschlossen.

R76 ist der Arbeitswiderstand von Q5(2) und soll so dimensioniert sein, dass Leckströme und parasitäre Kapazitäten von Q5 und Q2 die Funktion der Schaltung nicht beeinträchtigt.

Der Kondensator C207 bewirkt eine Phasenvoreilung in der Rückkoppelung des OPV und ist für die Stabilität des Regelkreises erforderlich, weil neben der Leerlaufverstärkung und Phasennacheilung des U24-A auch noch die Verstärkungen von Q5 und Q2 hinzu kommen, die beide in Source-Schaltung betrieben werden.

Ist der Regler im Eingriff, so bildet sich am Pin1 von U24-A eine Spannung in etwa der Höhe der Thresholdspannung (ca. 1bis 1,5V) von Q5 aus. Diese Spannung wird mit dem OPV U24-B, der als Komparator beschalten ist, mit einer fixen Spannung von ca. 2V (Pin6 von U-24-B) verglichen. Ist VEXT6..10V vorhanden, so liegt der Ausgang von U24-B auf LO Potential. Wird die Spannung VEXT6..10V entfernt, so liegt am Ausgang HI Potential an. Dieses Signal VBAT\_ON wird der CPU zugeführt, um den Batteriebetrieb vorzubereiten. Dazu muss die CPU das Signal VBAT\_EN auf HI setzen, um die Reihe der Transistoren Q3, Q5(1) und Q2(1) durchzuschalten, und somit UBAT auf VZK schalten zu können. Der Widerstand R87 bewirkt eine Hysterese von ca. 5%, damit ein sauberes Signal am Ausgang erzeugt wird.

Damit während des Umschaltvorgangs die Spannung VZK nicht zu stark einbricht, was einen Reset der CPU auslösen könnte, ist der Elko C13 mit 470µF eingefügt worden, der als Energiespeicher dient. Eine Abschätzung der Maximalzeit für die Spannungsumschaltung kann so berechnet werden:

$$Q = I \cdot t = C \cdot \Delta U \Rightarrow t = C \cdot \Delta U / I$$

Wenn man I mit 47mA annimmt und  $\Delta U$  mit 1V, so ergeben sich immerhin 10ms, in denen die CPU den Umschaltvorgang abgeschlossen haben muss.

Die Spannung des Batteriespannungsteilers R78/R79 wird auch noch, nach Verstärkung durch einen OPV, einem ADC- Eingang der CPU zugeführt, womit ein Monitoring der Batteriespannung erreicht werden kann.

### 3.1.4 Stromversorgung über externes Netzteil

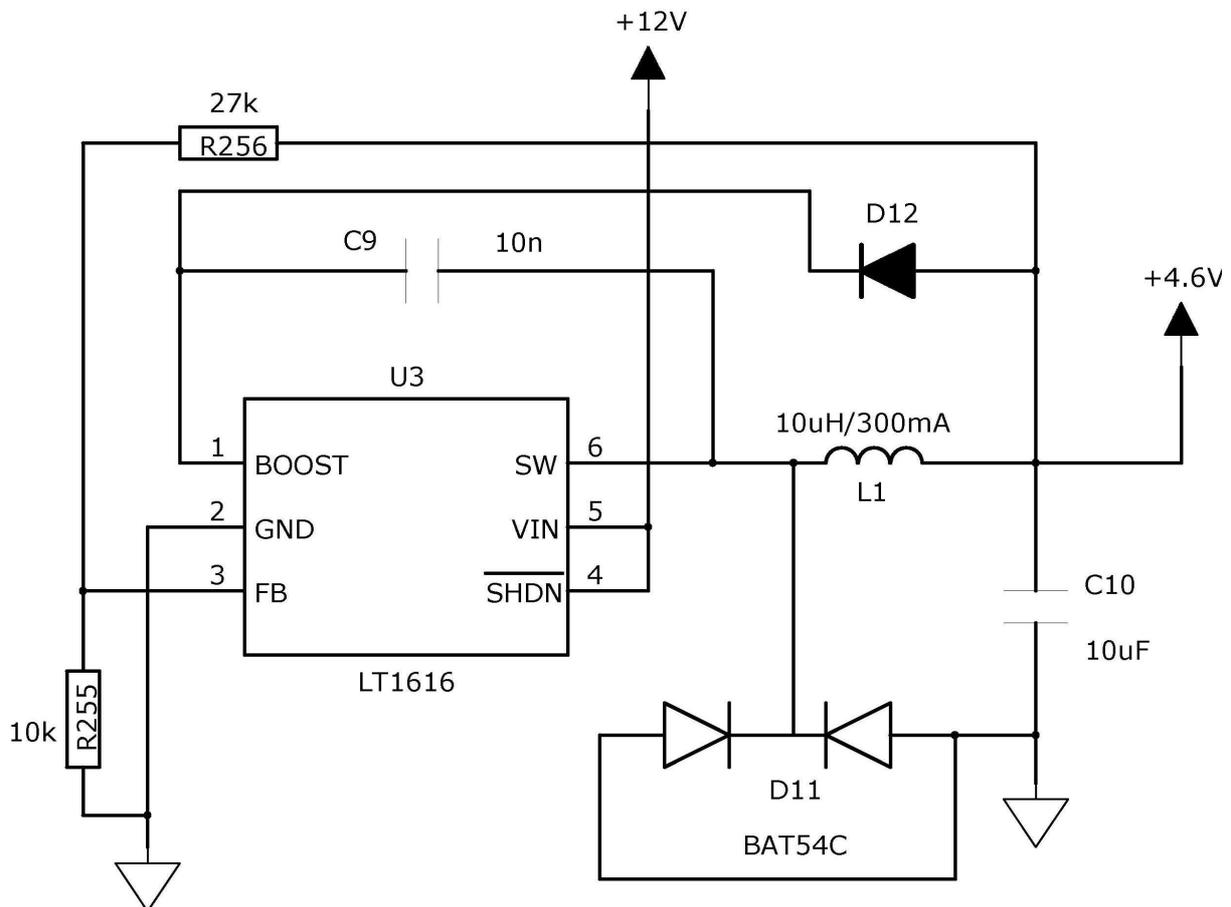


Bild 3.12 Schaltregler für die Spannungsanpassung des isolierten DC/DC Wandlers

Wie schon im Konzept erwähnt wird die externe Betriebsspannung von einem „medizinischen“ Steckernetzteil geliefert. Diese haben erhöhte Anforderungen an Spannungsfestigkeit und Leckströme. Die Nominalspannung beträgt 12V.

Da der isolierte DC/DC Wandler unregelt ist, braucht es zu dessen Versorgung eine gut definierte und deshalb geregelte Spannung, die in diesem Fall bei 4,6V liegt.

Der gemessene Eingangsstrom auf der 12V Schiene beträgt ca. 140mA im Normalbetrieb, bzw. 165mA im Hochlauf. Das entspricht einer Leistung von 1,7W bzw. 2W.

Für die 4,6V Schiene ergibt sich unter Berücksichtigung des Schaltreglerwirkungsgrades von ca. 80% eine Leistung von 1,36W oder ein Stromfluss von ca. 300mA im Normalbetrieb. Im Hochlauf sind allerdings bis zu 435mA zu erwarten, was schon sehr an der empfohlenen Grenze des verwendeten Schaltreglers LT1616 liegt. Der Überlastfall scheint aber sehr gut intern abgesichert zu sein, um eine Zerstörung des Bauteils zu vermeiden. Aufgrund der recht hohen Schaltfrequenz von typ. 1,4MHz kommt man mit einer kleinen Drossel aus, was in diesem Fall von großer Bedeutung war.

Für die Dimensionierung der Drossel gelten folgende Zusammenhänge:

$$L = \frac{(V_{in} - V_{out})V_{out}}{\Delta I_L V_{in} I_{out} f} = 9,6\mu\text{H} \quad [1]$$

Wobei:

$$V_{in} = 12\text{V}$$

$$V_{out} = 4,6\text{V}$$

$$I_{out} = 300\text{mA}$$

$$f = 1,4\text{MHz}$$

$$\Delta I_L = 0,7 \text{ (70\% Stromripple)}$$

[1] Abraham I. Pressman, Switching Power Supply Design, S65

Mit den Widerständen R255/R256 wird die Ausgangsspannung eingestellt, wobei die Widerstände so dimensioniert werden, dass die Spannung am FB Pin genau 1,25V beträgt.

Der LT1616 besitzt zwischen den Pins 5(VIN) und 6(SW) einen NPN Leistungstransistor, der mit dem Kollektor auf VIN liegt. Für diesen wird eine Treiberspannung am BOOST Pin benötigt, die von C9 und D12 erzeugt wird. Die Spannung am BOOST Pin muss höher sein als VIN, um den Transistor voll durchzuschalten. Damit ergibt sich die Einschränkung, dass dieser Schaltregler nicht auf 100% Tastverhältnis kommen kann, weil dann die BOOST Spannung nicht mehr generiert wird. Im Datenblatt ist ein maximales Tastverhältnis von 80% angegeben.

D11 stellt die Freilaufdiode dar, über die der Spulenstrom im ausgeschalteten Zustand des Leistungstransistors fließt.

Am Ein und Ausgang des Schaltreglers sind Kapazitäten mit einem niedrigen ESR vorzusehen. Im konkreten Fall wurden 10µF keramische Vielschichtkondensatoren verwendet.

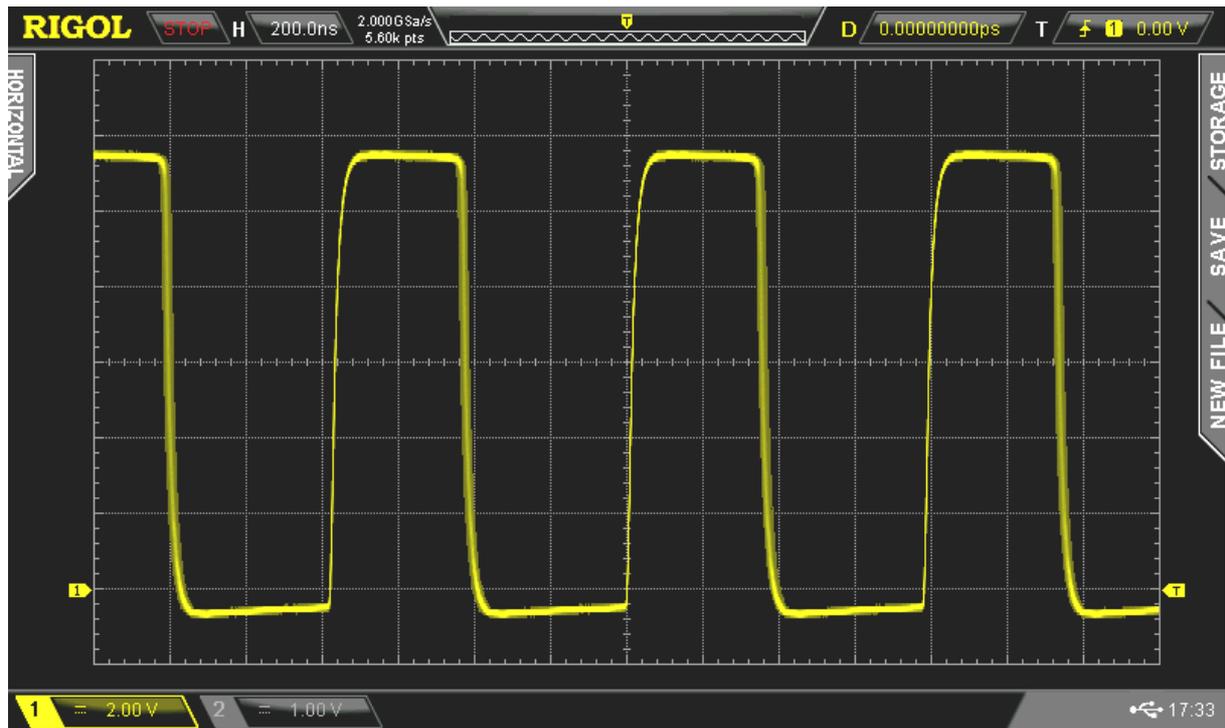


Bild 3.13

Signal am Pin 6 des LT1616, man erkennt, dass bei der vorhandenen Last von ca. 300mA am Ausgang der Stromfluss durch die Drossel kontinuierlich ist. Kriterium: in der Ausschaltphase ist die Spannung immer negativ (durch die Diode D11 geklemmt)

Der isolierte DC/DC Konverter stellt das Kernstück der isolierten Spannungsversorgung dar.

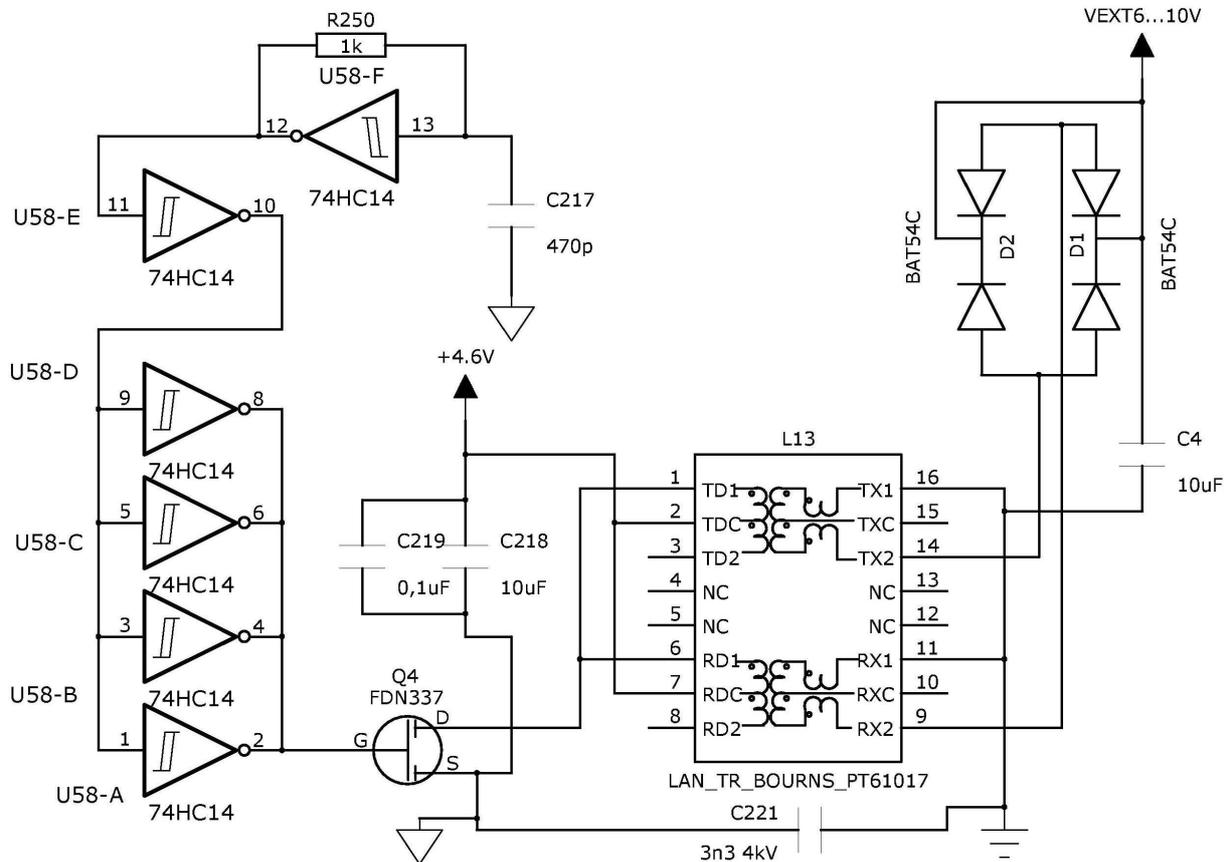


Bild 3.14 isolierter DC/DC Wandler

Funktionsweise:

Mit einem Schmitttrigger-Inverter Baustein 74HC14 wird ein Rechtecksignal erzeugt (U58-F). Die frequenzbestimmenden Bauteile R250/C217 sind so gewählt, dass eine Frequenz von ca. 1,2MHz am Pin 12 von U58 anliegt. Da das Rechtecksignal hier ein Tastverhältnis von etwas weniger als 50% hat, wird es zunächst invertiert (U58-E) und dann durch Parallelschalten von 4 Invertern der Ausgangsstrom erhöht und nochmals invertiert.

Ein Tastverhältnis von mehr als 50% am Gate des Leistungsmosfet Q4 hätte zur Folge, dass der Übertrager in die Sättigung geht, die Ströme sehr hoch werden und schließlich Q4 zerstört würde.

Der verwendete Leistungsmosfet FDN337 hat eine Strombelastbarkeit von 2,2A bei einer Spannungsfestigkeit von 30V in einem kleinen SOT23 Gehäuse. Vom Übertrager wird primärseitig nur die Hälfte der Wicklungen genutzt, sekundärseitig die volle Wicklung. Da die Übersetzung des Übertragers im Normalfall 1:1 ist, wird durch diese Maßnahme die Ausgangsspannung im Verhältnis zur Eingangsspannung in etwa verdoppelt.

Vom Prinzip her handelt es sich hier um eine vereinfachte Eintakt-Durchflusswandler Schaltung. Die Energie wird beim Einschalten von Q4

übertragen, eine Speicherung in Form einer magnetischen Energie findet aufgrund des sehr kleinen magnetischen Volumens kaum statt. Dadurch wird auch das Ausmaß des beim Ausschalten auftretenden Spannungsimpulses am Drain von Q4 recht moderat und übersteigt 12V nicht.

Mit den Schottky-Dioden D1/D2 wird der Ausgang des Übertragers gleichgerichtet und mit C4 geglättet.

C219/C218 dienen zur Stabilisierung der Betriebsspannung am Übertrager, und müssen ebenso wie Q4 räumlich eng bei diesem angeordnet sein.

C221 dient zur Unterdrückung von elektromagnetischer Abstrahlung, indem Primär- und Sekundärseite des Übertragers im HF Bereich kurzgeschlossen wird. Dieser Kondensator ist ein Sicherheitsbauteil mit 4kV Prüfspannung und Y1/500V~ klassifiziert.

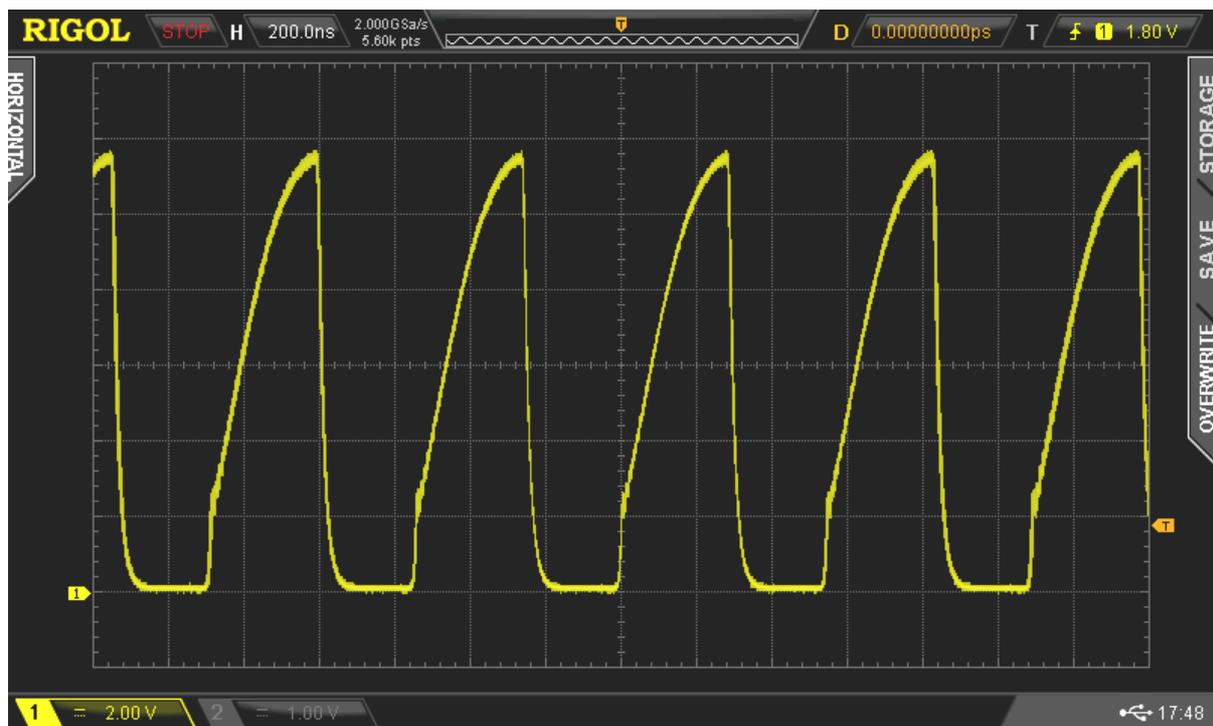


Bild 3.15

Signal am Drain des Leistungsmosfet Q4, Duty-Cycle ca. 40%,  $U_{Dmax} < 12V$

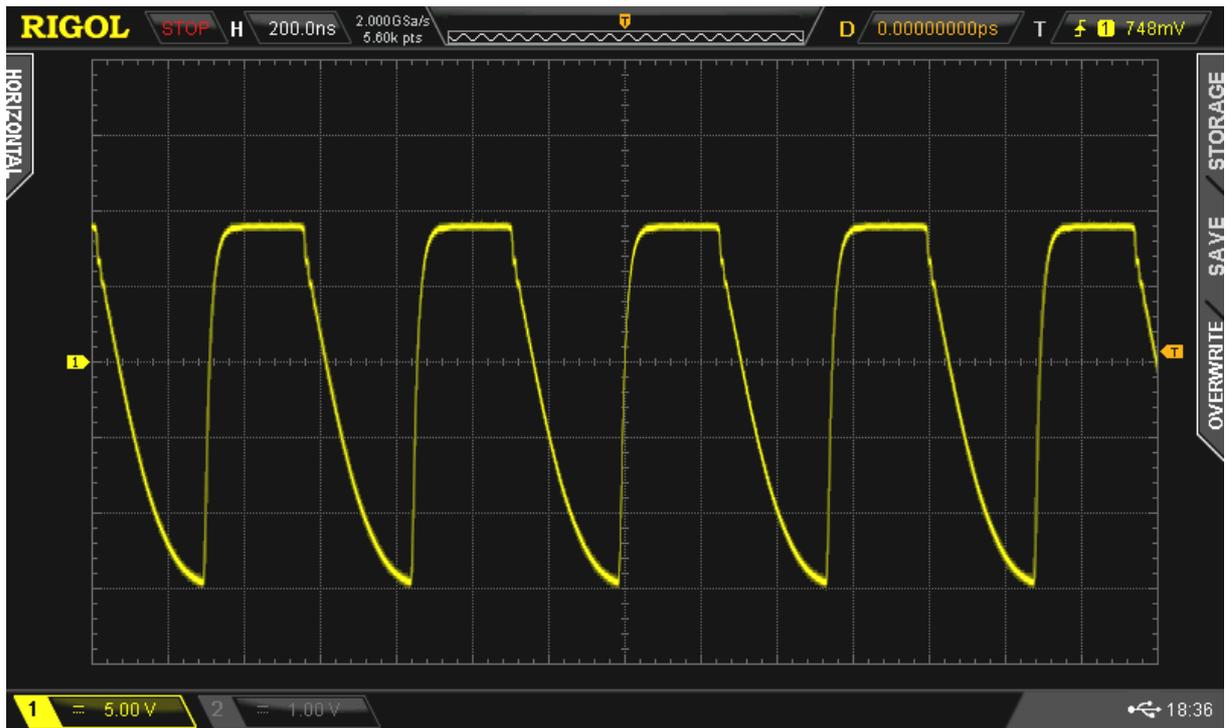


Bild 3.16  
Signal am Ausgang des Übertragers L13

### 3.1.5 Stromversorgung für analoge und digitale Schaltkreise

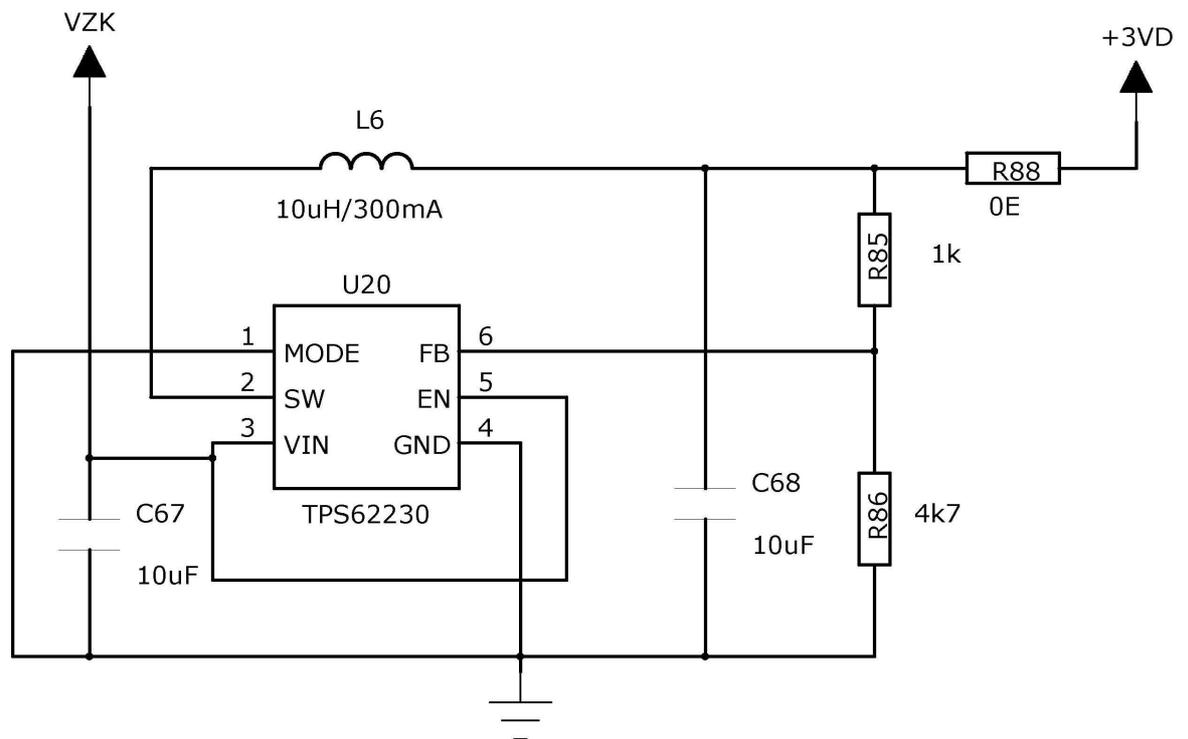


Bild 3.17 Schaltregler für die 3,0V Versorgungen

Für diese Aufgabe wurde der Typ TPS62230 von Texas Instruments ausgewählt. Die Gründe dafür waren der hohe Wirkungsgrad, die kleine Bauweise und eine Spannungsfestigkeit von 7V (absolute maximum ratings). Letzteres ist von Bedeutung, weil die meisten Schaltregler dieser Anwendungsklasse maximal 6V am Eingang verkraften, die Batteriespannung aber bis zu 6,7V erreicht.

Dieser Schaltregler besitzt am Ausgang intern eine Leistungs-Halbbrückenschaltung aus 2 komplementären Mosfets, wobei der high-side Mosfet einen  $R_{DSon}$  von ca. 0,6 Ohm und eine Strombelastbarkeit von ca. 1A aufweist. Weiters ist dieser Schaltregler für höchste Effizienz bei kleinen Laströmen entworfen worden. Dies wird durch einen extrem niedrigen Eigenstrombedarf des Bauteils erreicht. So beträgt der Wirkungsgrad selbst bei nur 1mA Laststrom gut 80%. Für diese kleinen Lasten, wird, wenn die  $T_{on}$  Zeit des Schaltreglers ein Minimum unterschreitet, die Schaltfrequenz reduziert (Pulse Frequency Modulation PFM). Dieser Modus wird durch das Anlegen von GND Potential an den MODE Pin eingestellt.

Der oben erwähnte Typ liefert bei direktem Anschluss der Ausgangsspannung an den Feedback Pin 2,5V Ausgangsspannung.

Es gäbe auch einen Typ mit genau 3,0V Ausgangsspannung, jedoch war dieser bei keinem der üblichen Distributoren erhältlich.

In Abweichung zur Beschaltung im Datenblatt, wurde deshalb der Feedbackpin über einen Spannungsteiler angeschlossen, um die gewünschten 3,0V Ausgangsspannung zu erhalten. Da es über die Innenbeschaltung dieses Pins keine Informationen gibt, erschien es ratsam, den Spannungsteiler niederohmig auszuführen, allerdings sollte dieser auch kein wesentlicher

Verlustfaktor sein. In der gegenständlichen Dimensionierung mit 1K/4K7 beträgt der Querstrom rund 0,5mA, was in etwa 1-2% Wirkungsgradverlust bedeutet.

Für die Dimensionierung der Drossel gelten folgende Zusammenhänge:

$$L = \frac{(V_{in} - V_{out})V_{out}}{\Delta I_L V_{in} I_{out} f} = 8,5\mu\text{H} \quad [1]$$

wobei:

$$V_{in} = 6,5\text{V}$$

$$V_{out} = 3,0\text{V}$$

$$f : \text{Schaltreglerfrequenz} = 3,8\text{MHz}$$

$$I_{out} : \text{Ausgangsstrom} = 50\text{mA}$$

$\Delta I_L$  : Ripplestrom der Spule = 100% (Grenzwert für den Übergang in den diskontinuierlichen Betrieb<sup>(1)</sup>)

[1] Abraham I. Pressman, Switching Power Supply Design, S65

<sup>(1)</sup> diskontinuierlicher Betrieb liegt dann vor, wenn der Strom durch die Drossel in der Ausschaltphase gleich 0 wird.

Entgegen dem Datenblatt, das Drosseln im Bereich von 0,7 bis 4,3 $\mu\text{H}$  empfiehlt, wurde eine 10 $\mu\text{H}$  Drossel mit reichlichen 300mA Sättigungsstrom ausgewählt. Dies deshalb, weil eine größere Induktivität generell den Ripplestrom der Schaltung herabsetzt, wodurch einerseits geringere elektromagnetische Störungen produziert werden, andererseits eine geringere Welligkeit der Ausgangsspannung erreicht wird.

Weiters ist bei dieser doch recht hohen Schaltfrequenz auch die Resonanzfrequenz der Drossel von Interesse. Idealerweise beträgt diese den etwa 10 fachen Wert der Schaltfrequenz. Liegt diese deutlich unterhalb dieses Wertes, so muss man von zusätzlichen relevanten Verlusten in der Drossel ausgehen, was hier natürlich unerwünscht ist.

Der 0-Ohm Jumper R88 ist bei der Inbetriebnahme der Schaltung nicht bestückt, weil eine fertigungsbedingte Fehlfunktion des Schaltreglers (Kurzschlüsse oder nicht korrekte Lötstellen) einen großen Schaden an der Gesamtschaltung anrichten könnte. Der R88 wird bei der Inbetriebnahme erst eingesetzt, wenn man sich vergewissert hat, dass die Ausgangsspannung 3,0V beträgt.



Bild 3.18  
Signal am Pin2 des TPS62230, Eingangsspannung ca. 6V,  $I_{out}$  ca. 15mA  
Gemessen mit einem 1:10 Tastkopf

## 3.2 Software

### 3.2.1 Verwendete Datenstrukturen

Um die Software für dieses doch relativ umfangreiche Projekt etwas übersichtlicher zu gestalten, wurden entsprechende Datenstrukturen angelegt.

USART/UART Datenstruktur:

```
typedef struct
{
    vu16 rxlen;
    vu16 txlen;
    u8 *txrdptr;
    u8 *txwrptr;
    u8 *rxrdptr;
    u8 *rxwrptr;
    u8 rxbuf[RXBUFLLEN];
    u8 txbuf[TXBUFLLEN];
}Uart_tt;
```

Diese Struktur enthält neben der aktuellen Anzahl von empfangenen bzw. zu sendenden Elementen (rxlen, txlen) noch 4 Pointer, mit denen auf die ebenfalls in der Struktur eingebetteten Empfangs bzw. Sendebuffer zugegriffen wird. Die Buffer werden linear verwaltet, weil eine zirkuläre Verwaltung erstens aufwändiger wäre und zweitens der Nutzen sehr beschränkt ist, weil das Senden bzw. Empfangen über die serielle Schnittstelle wesentlich langsamer vorangeht, als das Lesen bzw. Beschreiben der Buffer über die CPU selbst.

Gerätestatus Datenstruktur:

```
typedef struct
{
    vu8 eth_connected;           // Eth. Verbindung hergestellt
    vu8 adc1_run;                // Abtastung des Messsignals läuft
    vu8 SD_init;                 // SD Karte initialisiert
    vu8 SD_wr;                   // SD Karte Schreibvorgang aktiv
    vu8 SD_rd;                   // SD Karte Lesevorgang aktiv
    vu8 SD_rec_on;               // Speichermodus
    vu8 SD_TX_on;                // Übertragung SD-Karte => PC
    vu8 SD_Wr_buf_rdy;           // Buffer mit Messdaten fertig zum Speichern
    vu8 power;                   // Batterie/externe Vers.
    vu8 mux;                     // Multiplexer 0= CH1..12; 1= CH13..24
    vu8 num_avg;                 // Anzahl der gemittelten Abtastwerte/Kanal
    vu8 LED_mode;                // Akt. LED Modus
    vu8 LED_toggle;              // LED blinkt
    vu8 eth_prep;                // Vorbereitung der Ethernetverbindung
    vu32 last_stored_seq;        // Sequenznummer des zuletzt gesp. Datenblocks
}Status_t;
```

Diese interne Struktur enthält den aktuellen Betriebsstatus des Gerätes. Die Elemente sind selbsterklärend bzw. kommentiert.

Messdaten Datenstruktur:

```
typedef struct
{
    u16 ident1;          //0x7FFF
    u16 ident2;          //0x8000
    u16 ident3;          //0x7FFF
    u16 ident4;          //0x8000
    u16 numofbytes;     //144
    short data[31][2]; //ch1[0...1],ch2[0...1],ch3[0...1],ch4[0...1]
                       //Wertebereich +/-32767
    vu16 sequenz_L;     //fortlaufende Sequenznummer low Word
    vu16 sequenz_H;     //fortlaufende Sequenznummer high Word
    u16 trigger1;       //Aufzeichnung eines externen Ereignisses
    u16 trigger2;
    u16 chksum;
}eeg_data_t;
```

Mit dieser Struktur werden die Messdaten an den angeschlossenen PC via Ethernet übertragen beziehungsweise auf der SD-Karte gespeichert. Da immer nur 2 Messwerte eines Kanals übertragen werden, entspricht die Frequenz der Datenübertragung immer exakt der Hälfte der Ausgangs-Abtastrate und sorgt so für minimale Störungen im Messsignal.

Generelle Kommunikations Struktur:

```
typedef struct
{
    u16 ident;          //Identifizier gen. settings
    u16 num_of_bytes;   //ges. Anzahl der Bytes inkl. checksumme
    u16 sample_rate;    //2er Pot.
    char active_chan[64]; //1 für aktiv,0=Eingang kurzschließen für max. 64 Kan.
    char notch;         //Brummfilter 50,60 oder 0 für aus
    char data_req;      //1=Mess-Daten senden
    char checksum;      //summe der Bytes von ident bis data_req
}general_settings_t;
```

Diese Struktur wird vom PC an das Gerät gesendet und legt die grundlegenden Einstellungen des Gerätes fest. Die Struktur wurde vom bestehenden 32Kanal EEG-Gerät hier weiterverwendet, wobei das Element „notch“ hier keine Bedeutung hat, weil aufgrund der hohen Grundauslastung der CPU keine Brumm-Filter gerechnet werden können. Dies muss am auswertenden PC geschehen.

## Geräte - Eigenschafts - Struktur:

```
typedef struct //wird nur von PC gelesen
{
    u16 ident; //Identifizier device capabilities
    u16 num_of_bytes; //ges. Anzahl der Bytes inkl. checksumme
    u16 num_of_chan; //Anz. Kanäle
    char hw_vers[20]; //20 ASCII Zeichen
    char cpu_vers[20]; //20 ASCII Zeichen
    char dsp_vers[20]; //20 ASCII Zeichen
    char ser_num[20]; //Seriennummer(ASCII)
    char date_of_cal[10]; //Kalibrierdatum(ASCII) JJ/MoMo/TT/SS/MinMin
    u16 filt_cap256[8]; //4HP,4TP, 0xFFFF=kein Filter
    u16 filt_cap512[8]; //4HP,4TP
    u16 filt_cap1024[8]; //4HP,4TP
    u16 filt_cap2048[8]; //4HP,4TP
    u16 filt_cap4096[8]; //4HP,4TP
    u16 filt_cap8192[8]; //4HP,4TP
    u16 filt_cap16384[8]; //4HP,4TP
    u16 par[10]; //10 individuelle Parameter
    char checksum;
}device_cap_t;
```

Diese Struktur wird auf Anfrage vom Gerät an den PC gesendet und beschreibt die Einstellmöglichkeiten und Eigenschaften des Gerätes. Auch diese Struktur wurde vom bestehenden 32Kanal EEG-Gerät hier weiterverwendet wobei die Elemente „filt\_cap512“ bis „filt\_cap16384“, die die möglichen Grenzfrequenzen der digitalen Filterbank angeben, hier klarerweise nicht verwendet werden. Auch das Element „dsp\_vers“, das die Version der DSP-Software angibt, ist in diesem Fall obsolet.

## Messbereichs-Struktur:

```
typedef struct
{
    u16 ident; //75 dez.
    u16 num_of_bytes;
    u32 range_1; //nom. Messbereich 1 1dig = +/-1uV
    char channels_1[64]; //Kanalnummern für den Messbereich 1
    // (1...64)0xFF= "nicht möglich"
    u32 range_2;
    char channels_2[64]; //Kanalnummern für den Messbereich 2
    u32 range_3;
    char channels_3[64]; //Kanalnummern für den Messbereich 3
    u32 range_4;
    char channels_4[64]; //Kanalnummern für den Messbereich 4
    char checksum;
} range_cap_t;
```

Mit dieser Struktur, die auf Anfrage vom Gerät an den PC gesendet wird, werden die möglichen Messbereiche der einzelnen Messkanäle beschrieben. Bis zu 4 Messbereiche kann grundsätzlich ein Kanal aufweisen, allerdings gibt es bei diesem Gerät nur fixe Verstärkungsfaktoren und 2 Gruppen von Kanälen mit 2 unterschiedlichen Messbereichen.

Allgemeine Steuerstruktur für das Gerät:

```
typedef struct
{
    u16 ident;           //66 dez. für dat_req
    u16 num_of_bytes;   //ges. Anzahl der Bytes inkl. checksumme
    u16 channel_num;    //Kanalnummer nur sinnvoll wenn Anforderung Kanalsp.
    char checksum;
}com_request_t;
```

//Bedeutung von dat\_req.channel\_num in der Struktur com\_request\_t

```
#define START          1      //online Daten senden
#define STOP           2      //online Daten nicht senden
#define DISCONNECT     3      //Beenden der TCP/IP Verbindung
#define START_SD_REC   4      //Starten des SD Karten Rekorders
#define START_SD_TX    5      //Senden der SD Karten Daten
#define STOP_SD_REC    6      //Stoppen des SD Karten Rekorders
#define STOP_SD_TX     7      //Stoppen des Sendens der SD Daten
```

Diese Struktur wird vom PC an das Gerät gesendet und dient der allgemeinen Steuerung. Der Wert des Elementes „channel\_num“ bestimmt, welcher Vorgang im Gerät ausgelöst wird.

Kommunikationsstrukturen für das Aufzeichnen von Messdaten:

```
typedef struct
{
    u16 ident;           //19
    u16 num_of_bytes;   //72
    u32 num_of_eeg_data; //Anz. von eeg-data Blöcken zu 144Bytes bis
                        //abgeschalten wird (0= Abschalten bei Versorgung
                        //aus, oder SD-Karten overflow)
    char meas_desc[64]; //ASCII Daten Patient,Datum...
}app_meas_info_t;      //wird vom PC an Gerät gesendet
```

Diese Struktur wird vor dem Einleiten des Daten – Recordings an das Gerät gesendet und beinhaltet die vorgesehene Länge der Aufzeichnung und spezifische Messinformationen über Patient und Aufnahmedatum.

```

typedef struct
{
u16 ident;           //18
u16 num_of_bytes;   //73
u32 num_of_eeg_data; //akt. Anz. von eeg-data Blöcken zu 144Bytes
char meas_desc[64]; //ASCII Daten Patient,Datum...
char SD_rec_on;     //Datenrekorder ein/aus
}device_meas_info_t; //wird vom Gerät an PC gesendet

```

Diese Struktur gibt den momentanen Status der Aufzeichnung wieder.

```

typedef struct
{
u16 ident;           //20
u16 num_of_bytes;   //12
u32 SD_TX_start_seq; //bestimmt, ab welchem Block ausgegeben wird
u32 SD_TX_stop_seq; //bestimmt, bis zu welchem Block ausgegeben wird
}app_SD_TX_info_t; //wird vom PC an Gerät gesendet

```

Mit dieser Struktur hat man die Möglichkeit bestimmte Sequenzen aus dem aufgezeichneten Datenstrom von der SD-Karte zu lesen.

### 3.2.2 Verwendete Tasks

**Task "MainLoop"** , Priorität: 2 (niedrigste) , Aufruf: Grundtask, wird immer ausgeführt, wenn kein anderer Task oder ein Interrupt aktiv ist.

Dies ist der zentrale Task des Systems, der alle Gerätefunktionen überwacht und auf eingehende Signale reagiert.

Folgende Funktionen sind implementiert:

- Pollen der seriellen Schnittstellenbuffer
- Stromversorgungsumschaltung und Überwachung
- Ethernet Verbindungsaufbau
- Pollen des Ethernetbuffers und Interpretation der eingelangten Daten

**Task "Eth\_connect\_tsk"** Priorität: 3, Aufruf: Event

Stellt die TCP/IP Verbindung in Form von 2 Sockets<sup>(1)</sup> zwischen Gerät und PC her. Ein Socket mit der "default" Portadresse 200 dient zur Übertragung der Messdaten vom Gerät zum PC und ist quasi unidirektional.

Ein weiterer Socket mit der "default" Portadresse 300 dient zur allgemeinen Kommunikation. Dabei werden die im vorangegangenen Kapitel erwähnten Strukturen benutzt und gesendet.

Der Vorteil dieser 2 Socket Lösung ist, dass Messdaten und Kommunikations/Befehlsdaten von Haus aus getrennt sind und damit nicht aufwändig in einem Algorithmus getrennt werden müssen.

<sup>(1)</sup>Ein Socket kann als eine Struktur von 2 FIFO Buffern aufgefasst werden, wobei einer für das Senden beschrieben wird, der andere nach dem Empfang von Daten ausgelesen wird. Der Socket ist einmal zu initialisieren (IP-Adresse, Port-Nummer,...) dann werden nur noch Nutzdaten gelesen bzw. geschrieben.

**Task "led\_blink"** Priorität: 3, Aufruf: periodisch, 500ms

Optische Ausgabe des Betriebszustandes mittels einer 2 färbigen(rot/grün) LED. Folgende Betriebs- bzw. Fehlerzustände werden angezeigt:

- Rot blinken: Fehler SD-Karte
- Rot dauerleuchten: Unterspannung Stromversorgung
- Grün blinken: Gerät im batteriegespeisten Rekordermodus
- Grün dauerleuchten: Gerät im extern-gespeisten Rekordermodus
- Orange blinken: Gerät versucht eine TCP/IP Verbindung herzustellen
- Orange dauerleuchten: TCP/IP Verbindung hergestellt, Gerät im online-Modus

**Task "bat\_volt\_tsk"** Priorität:3, Aufruf: periodisch, 500ms

Dieser Task misst und mittelt die Batteriespannung und führt im Falle der Unterspannung das geordnete Abschalten des Gerätes herbei.

**Task " chipox\_contr\_tsk"** Priorität: 4 , Aufruf: periodisch, 50ms

Dieser Task bewerkstelligt die serielle Kommunikation mit dem optionalen Chipoximeter.

**Task "Send\_SD\_data\_tsk"** Priorität: 5, Aufruf: Event

Sendet die auf SD-Karte aufgezeichneten Messdaten an den PC.

**Task "Wr\_SDbuf\_tsk"** Priorität: 6, Aufruf: Event

Schreibt die gebufferten Messdaten auf die SD-Karte.

**Task "Mke\_SDbuf\_tsk"** Priorität: 7, Aufruf: Event

Buffert die Messdaten in zwei 4608 Byte (9 Sektoren zu je 512 Byte) große Buffer (Wechselbufferprinzip), damit sie dann "in einem Stück" auf die SD-Karte geschrieben werden können.

**Task "AccMeter\_tsk"** Priorität: 8, Aufruf: periodisch, 10ms

Liest den Beschleunigungssensor aus.

**Task "Cal\_sum\_buf"** Priorität: 9, Aufruf: Event mit 256Hz

Dieser Task wird mit der Ausgangsabrataste von 256Hz aufgerufen und multipliziert jeden Messwert mit dem zugehörigen Kalibrationsfaktor. Außerdem wird über einen gleitenden Mittelwert ein Hochpassfilter für alle 24 Kanäle gerechnet, um den DC Anteil des Signals zu minimieren. Weiters werden noch die Daten des Beschleunigungssensors und des Chipoximeters, sowie die aktuelle Sequenznummer des Datensatzes in die Messdatenstruktur eingefügt.

Wenn die komplette Messdatenstruktur befüllt ist (nach jedem 2. Aufruf), wird diese, falls sich das Gerät im online-Modus befindet, an den PC gesendet. Falls das Gerät im Rekordermodus mit Batteriebetrieb ist, werden die Daten an den Task " Mke\_SDbuf\_tsk " zur weiteren Bufferung weitergegeben.

**Task "ADC\_sum\_up"** Priorität: 10, Aufruf: DMA-Interrupt mit 16384Hz

Dieser Task wird nach der Abtastung von jeweils 12 Kanälen ausgeführt, und addiert die Messwerte kanalgetrennt zu den 24 Summenbuffern. Wenn dieses 32 mal geschehen ist (oversampling), kommt es zum Bufferwechsel und der fertige Buffer wird über den Task "Cal\_sum\_buf" weiterverarbeitet.

### 3.2.3 Verwendete Interrupts

DMA Interrupt:

```
void DMA1_Channel1_IRQHandler(void)
{u16 *tempptr;

if((DMA1->ISR & 0x0006) == 0x4){//nach 6 Kanälen MUX1 toggeln
    if(DevStatus.mux)CLR_MUX1;
    else SET_MUX1;
    DMA1->IFCR = DMA1_IT_HT1;
}
else{//TC interr. nach 12 Werten
    DMA1_Channel1->CCR &= CCR_ENABLE_Reset;
    if(DevStatus.mux){CLR_MUX2;
        CLR_MUX3;
        DevStatus.mux=0;
        tempptr=ADC_buf_rd_ptr;
        ADC_buf_rd_ptr = ADC_buf_wr_ptr;
        ADC_buf_wr_ptr = tempptr;
        DMA1_Channel1->CMAR= (u32)ADC_buf_wr_ptr;
        DMA1->IFCR = DMA1_IT_GL1;
        DMA1_Channel1->CNDTR =12;
        DMA1_Channel1->CCR |= CCR_ENABLE_Set;
        isr_evt_set (0x0001, adc_sum_up);
    }
}
```

```

else {
    SET_MUX2;
    SET_MUX3;
    DevStatus.mux=1;
    tempptr=ADC_buf_rd_ptr;
    ADC_buf_rd_ptr = ADC_buf_wr_ptr;
    ADC_buf_wr_ptr = tempptr;
    DMA1_Channel1->CMAR= (u32)ADC_buf_wr_ptr;
    DMA1->IFCR = DMA1_IT_GL1;
    DMA1_Channel1->CNDTR =12;
    DMA1_Channel1->CCR |= CCR_ENABLE_Set;
    isr_evt_set (0x0001, adc_sum_up);
}
}
}

```

#### Beschreibung:

Der DMA Controller bietet die Möglichkeit einen Datentransfer zwischen peripheren Einheiten und dem Speicher ohne Intervention der CPU durchzuführen.

Dazu müssen dem DMA Controller die Adresse der peripheren Einheit, sowie die Adresse des Datenbuffers in entsprechende Register geschrieben werden. Weiters muss die Anzahl der zu transferierenden Daten bekanntgegeben werden.

Der DMA Interrupt tritt grundsätzlich dann auf, wenn ein Datentransfer beendet ist. Weiters kann im DMA Controller auch noch ein Auftreten nach der Hälfte des vorgesehenen Datentransfers eingestellt werden. Dieses wird auch ausgenutzt, um den Multiplexer MUX1 rechtzeitig umzuschalten. In obiger Interruptroutine wird dazu das Bit2 des DMA Interrupt Status Registers geprüft:

```
if(DMA1->ISR & 0x0006) == 0x4)
```

Falls dieses Bit gesetzt ist, wird MUX1 umgeschaltet, und das Bit über das DMA Interrupt Flag Clear Register gelöscht. Weitere Aktionen sind nicht nötig, denn der DMA Datentransfer ist ja noch nicht abgeschlossen.

Nach 12 transferierten Abtastwerten tritt dann der "Transfer complete" Interrupt auf, in dem dann die Datenbuffer getauscht werden und ein Signal an den "adc\_sum\_up" Task gesendet wird, um die Daten weiterzuverarbeiten. Dazu sind 2 Buffer zu jeweils 12 Werten und 2 Pointer (ADC\_buf\_wr\_ptr, ADC\_buf\_rd\_ptr), die auf die Anfangsadressen der Buffer zeigen, angelegt worden. Mit einem Hilfspointer "tempptr" werden die 2 Pointer und damit die Buffer getauscht. Danach erfolgt die Neuinitialisierung des DMA Controllers und das Löschen des Interrupt-Flags.

## USART Interrupt:

```
void USART1_IRQHandler(void)
{
    vu8 tmp1;
    if(USART_GetITStatus(USART1, USART_IT_RXNE) != RESET)
    { //Byte empfangen
        if(Uart0.rxwrptr<(Uart0.rxbuf+sizeof(Uart0.rxbuf))){
            *Uart0.rxwrptr++ = USART1->DR;
            Uart0.rxlent++;
        }
        else tmp1=USART1->DR; //Buffer voll, trotzdem Lesen damit Flag rückgesetzt wird
    }

    if(USART_GetITStatus(USART1, USART_IT_TXE) != RESET)
    { if(Uart0.txlen>0){ //Daten Senden
        USART1->DR = *(Uart0.txrdptr++);
        Uart0.txlen--;
    }
    if(Uart0.txlen==0){ //Alle Daten gesendet
        Uart0.txrdptr=Uart0.txbuf;
        Uart0.txwrptr=Uart0.txbuf;
        USART1_TXIE_OFF;
    }
    }
}
```

### Beschreibung:

Die serielle Kommunikation für Debug oder Service Zwecke erfolgt über den USART1 der CPU. Dabei werden je ein Buffer für senden bzw. empfangen verwendet, weiters lese und schreib-Pointer die der Bufferverwaltung dienen. Die Bufferverwaltung ist linear. Der Ablauf eines Empfangs ist folgendermaßen:

1. Auslösung des USART1 Interrupts durch ein empfangenes Byte
2. Ausführung der Interruptroutine
  - 2a) Test ob ein TX oder RX Interrupt aufgetreten ist
  - 2b) falls der RX Buffer nicht voll ist, Byte mit RX-Schreibepointer in Buffer schreiben
  - 2c) RX-Schreibepointer und aktuelle Bufferlänge inkrementieren
3. Beendigung der Interruptroutine
4. Im "MainLoop" Task wird die aktuelle Bufferlänge „gepollt“ und das angekommene Byte weiterverarbeitet.

Das Senden von Daten erfolgt nach folgendem Ablauf:

1. Schreiben der Daten in den TX Buffer, TX Interrupts freigeben
2. Auslösung des USART1 Interrupts
3. Ausführung der Interruptroutine
  - 3a) Test ob ein TX oder RX Interrupt aufgetreten ist
  - 3b) falls der TX Buffer nicht leer ist, Byte mit TX-Lesepointer in USART1 Datenregister schreiben
  - 3c) TX-Lesepointer inkrementieren und aktuelle Bufferlänge dekrementieren
  - 3d) Falls alles gesendet, TX-Pointer rücksetzen und TX-Interrupts sperren
4. Beendigung der Interruptroutine

### Timer6 Interrupt:

```
void TIM6_IRQHandler(void)
{
    TIM_ClearITPendingBit(TIM6, TIM_IT_Update);
    DAC_SetChannel1Data(DAC_Align_12b_R, *DAC_sin_ptr);
    DAC_sin_ptr++;
    DAC_cntr++;
    DAC_SoftwareTriggerCmd(DAC_Channel_1, ENABLE);
    if(DAC_cntr>=64){DAC_sin_ptr=(u16*)&TableS;
                    DAC_cntr=0;
                    }
}
```

### Beschreibung:

Die Ausgabe des analogen Kalibrier Signals erfolgt durch den DAC des Mikrocontrollers. Um eine äquidistante Abtastung des Signals zu erzielen, wird der Timer6 Interrupt verwendet.

Die 64 Werte für eine Signalperiode werden in einem Buffer "TableS" bereitgehalten. Weiters wurde noch ein Pointer "DAC\_sin\_ptr" installiert, der bei Start der Software auf das erste Bufferelement von "TableS" zeigt und eine Zählvariable "DAC\_cntr". Die Interruptfrequenz des Timers 6 muss klarerweise so eingestellt werden, dass sie das 64 fache der Signalfrequenz beträgt.

Wird der Timer6 gestartet, werden in regelmässigen Abständen Interrupts generiert. Dabei wird als erstes das Interrupt Flag zurückgesetzt, danach der Wert des "DAC\_sin\_ptr" in das Datenregister des DAC geschrieben. In weiterer Folge wird der Pointer und die Zählvariable inkrementiert. Hat die Zählvariable "DAC\_cntr" den Wert 64 erreicht, wird der Pointer wieder auf das erste Bufferelement von "TableS" gesetzt und die Zählvariable auf 0 gesetzt. Somit ist eine äquidistante, zyklische Ausgabe der Bufferwerte möglich.

### ADC3 Interrupt:

```
void ADC3_IRQHandler(void) __irq
{
    ADC_ClearITPendingBit(ADC3, ADC_IT_EOC);
    isr_evt_set (0x0008, bat_volt );
}
```

### Beschreibung:

Die Wandlung am ADC3 wird vom periodischen Task "bat\_volt\_tsk" alle 2 ms gestartet. Wenn der Prozess beendet ist, tritt der Interrupt auf, um zu signalisieren, dass ein neuer Abtastwert vorhanden ist. Das Interrupt Flag ist zurückzusetzen und es wird ein Signal an den Task "bat\_volt\_tsk" gesendet, der dann den Abtastwert weiterverarbeitet.

### 3.2.4 Startsoftware für Mikrocontroller und Tasks

Der Mikrocontroller startet mit dem internen RC Oszillator aus dem Hardware-Reset in die "Main" Funktion, nachdem der Stack beziehungsweise der Stackpointer initialisiert wurde. Die Größe des Stacks, der die Daten bei Aufruf einer Unterfunktion oder eines Interrupts hält, muß der Anwendung angepasst werden. In diesem Fall, bei dem es sich überschneidende Interrupts gibt, empfiehlt es sich, hier großzügig zu sein, auch weil das gesamte RAM immerhin 64KByte groß ist. Ein Stacküberlauf würde sehr schnell zu einem völligen Fehlverhalten (Absturz) der Software führen und ist oft nur schwer zu identifizieren. Die Stackgröße wurde hier mit 4KByte festgelegt.

In der "Main" Funktion werden zunächst Initialisierungen im Mikrocontroller selbst ausgeführt.

Dazu wird ersteinmal die Funktion "RCC\_Configuration" ausgeführt, die den externen Quarz mit 12,5MHz zum Schwingen bringt und eine interne PLL speist, die den Systemclock auf 50MHz bringt.

Weiters müssen die Clocks für jede verwendete periphere Mikrocontroller Einheit explizit eingeschalten werden. Ohne diesen Vorgang, können keinerlei Register in den peripheren Einheiten beschrieben oder gelesen werden. Dieses Vorgehen erscheint umständlich, ist aber aus Stromspar - Gründen durchaus sinnvoll, denn die nicht mit dem Clock versorgten Einheiten brauchen praktisch keinen Strom.

Als nächstes werden mit der Funktion "NVIC\_Configuration" die Prioritäten der 6 Interrupts konfiguriert und die Interruptkanäle freigeschaltet:

- |            |  |
|------------|--|
| 1.) DMA1   | (DMA für Messsignale von ADC1/2)           |
| 2.) TIM6   | (Timerinterrupt für Ausgabe von DAC Daten) |
| 3.) SDIO   | (SD Speicherkarteninterface)               |
| 4.) USART1 | (serielle Schnittstelle für Debug)         |
| 5.) USART3 | (serielle Schnittstelle für Chipoximeter)  |
| 6.) ADC3   | (ADC für Batteriespannung)                 |

Die Reihenfolge hier gibt gleichzeitig auch die Prioritäten der Interrupts wieder, dh. DMA1 hat die höchste Priorität, ADC3 die niedrigste.

Die Auswahl der Priorität für einen Interrupt ist gut zu überlegen und hängt von der zur Verfügung stehenden Zeit vom Auftreten bis zum fertigen Abarbeiten des Interrupts ab. Nachdem der ADC1/2 mit 196608 Hz Abtastwerte liefert, und nach 6 Werten der DMA1 Interrupt auftritt, hat man hier maximal etwa 30µs Zeit, sonst gehen Daten und die Kanalzuordnung verloren.

Bei TIM6, der den Ausgabetak für den DAC liefert, hätte man bei 64 Stützwerten/Periode und 10Hz Signalfrequenz zwar rund 1,5ms Zeit, allerdings hätte eine zeitlich nicht äquidistante Ausgabe der Werte eine Verzerrung des Signals zur Folge, was höchst unerwünscht ist. Daher ist die Priorität hier aus signaltechnischen Gründen hoch.

Als nächstes werden dann die peripheren Einheiten konfiguriert und endgültig in Betrieb genommen.

„init\_GPIO()“

Konfiguriert die Prozessor-Pins auf digital - Eingang, digital - Ausgang, analog - Eingang oder alternative Funktion.  
Bei letzterem übernimmt eine periphere Einheit des Prozessors die Kontrolle über den Pin.

„init\_USART1()“  
„init\_USART3()“

Initialisiert die seriellen Schnittstellen für Debugzwecke und das optionale Chipoximeter. Dabei werden auch die im vorangegangenen Kapitel beschriebenen Softwarestrukturen für den USART initialisiert.

„init\_ADC()“

Initialisiert die internen ADC's des Mikrocontrollers. Für den ADC1/2, der für die Abtastung der EEG-Kanäle zuständig ist, wird außerdem der Timer4, der den Abtast - Trigger liefert und ein DMA - Kanal, der die Abtastwerte auf einen Buffer kopiert, initialisiert.

Der ADC3, der nur für die Messung der Batteriespannung verwendet wird, wird ebenfalls initialisiert, die Wandlungen werden dann über einen Software - Befehl ausgelöst, da eine exakt äquidistante Abtastung hier nicht notwendig ist.

„init\_DAC()“

Initialisiert den internen DAC des Mikrocontrollers für die Ausgabe eines analogen Kalibrationssignals. Weiters wird auch der Timer6, der die Interrupt's für die Ausgabe der Werte liefert, konfiguriert. Für die Ausgabe des Signals wird dann nur der Timer6 eingeschaltet.

„init\_FSMC()“

Initialisiert das 16 Bit Bus-System samt Kontroll-Leitungen für die Verbindung mit dem Ethernetbaustein W5300.

„init\_SPI3()“

Initialisiert die SPI Schnittstelle, für die Kommunikation mit dem Beschleunigungssensor.

„FLASH\_Unlock()“  
„EE\_Init()“

Initialisiert das virtuelle EEPROM, in dem die spezifischen Gerätedaten gespeichert sind. Dafür werden 2 Blöcke des internen Prozessor-Flash's verwendet.

In der weiteren Ausführungsfolge des Programms wird die serielle Schnittstelle USART0 auf den Empfang des ASCII Zeichens „x“ kontrolliert. Falls dieses empfangen wurde, geht das Gerät in den Service-Modus über, in dem über die serielle Schnittstelle gerätespezifische Daten (Kalibrierwerte, IP-Adresse, Port-Nummer, Seriennummer usw.) in das virtuelle EEPROM geschrieben werden können.

Schließlich wird dann auch noch die SD-Karte samt SDIO Controller initialisiert.

Als Letztes werden dann noch systeminterne Softwarestrukturen initialisiert und die Tasks des Betriebssystems initialisiert und gestartet.

### 3.2.5 Hochpassfilter über exponentiell geglätteten Mittelwert

Der analoge Hochpassfilter der Verstärkereingänge arbeitet aufgrund von unterschiedlichen Bias-Strömen und Offsetspannungen der Differenzbilder (Instrumentenverstärker) nicht optimal. Für die vorhandene Auswertesoftware ist es aber essentiell, dass im Signal kein DC Anteil vorhanden ist. Daher wurde es notwendig, einen Hochpassfilter digital zu implementieren.

Üblicherweise wird das über einen IIR Filter gemacht, dazu müsste man aber in der CPU double float Werte effizient verarbeiten können, was beim gegenständlichen Prozessor nicht der Fall ist. Die Problematik liegt in der numerischen Auflösung, weil in diesem Fall das Verhältnis von Abtastrate (256 Hz) und Grenzfrequenz (0,16 Hz) extrem ist. Es hat sich schon beim 32 Kanal EEG-System beim Implementieren von IIR Filtern auf 16 Bit DSP's gezeigt, dass bei Filtern 1.Ordnung mit 16Bit Auflösung das Verhältnis von Abtastfrequenz und Grenzfrequenz nicht größer als der Faktor 20 sein sollte, um ein ordnungsgemäßes Arbeiten des Filters sicherzustellen.

Die Lösung des Problems wurde über das Subtrahieren eines Signalmittelwertes vom Signal selbst erreicht.

Zunächst wurde das Verhalten der Mittelwertbildung betrachtet.

Der Mittelwert sollte in folgender Weise berechnet werden:

$$Y_{(n)} = Y_{(n-1)} * (1-K) + X_{(n)} * K$$

Wobei:

$Y_{(n)}$ .....Ausgangswert

$Y_{(n-1)}$ .....vorhergehender Ausgangswert

$X_{(n)}$ .....neuer Abtastwert

K.....Mittelwertkonstante,  $0 < K < 1$

Es findet hier also eine Gewichtung des vorhergehenden Mittelwertes  $Y_{(n-1)}$  und des neuen Wertes  $X_{(n)}$  statt.

Die graphische Darstellung der Ausgangswerte  $Y_{(n)}$  für den Einheitssprung am Eingang und ein  $K$  von 0,05 sieht dann folgendermaßen aus:

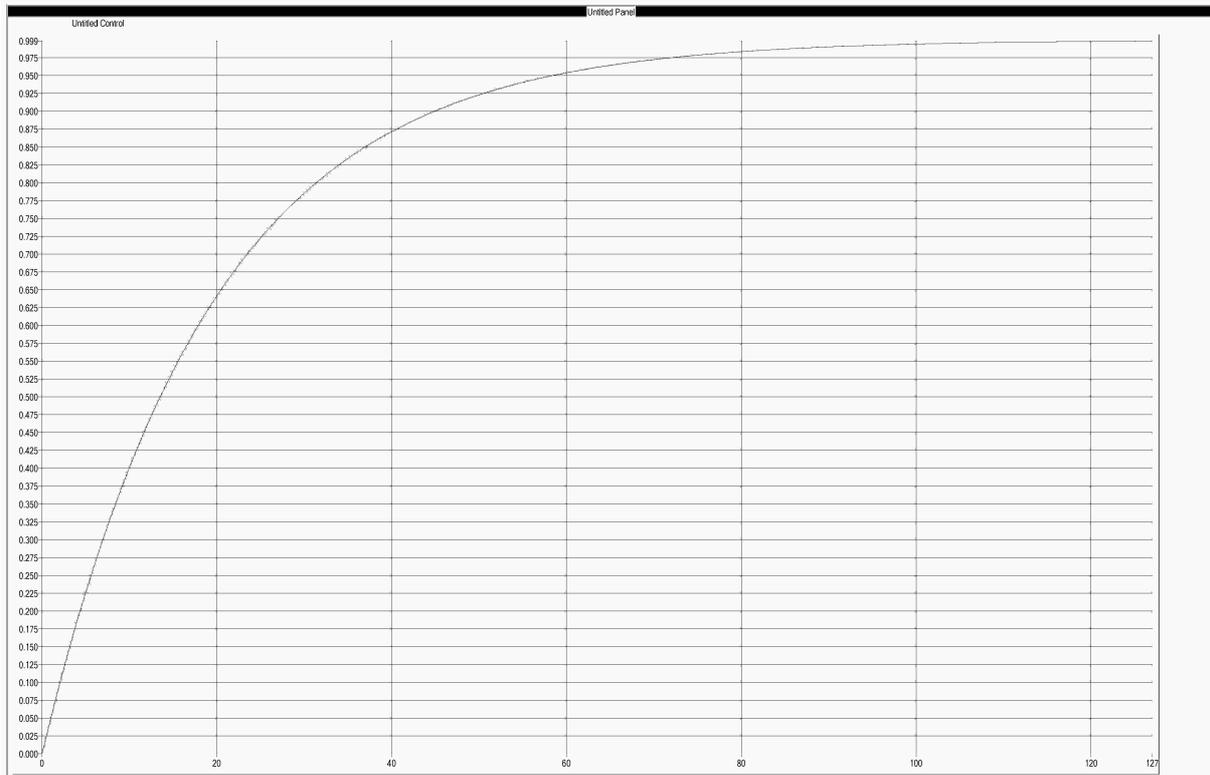


Bild 3.20  
Tiefpassverhalten des exponentiell geglätteten Mittelwertes im Zeitbereich

Das Verhalten ist dem eines analogen RC Tiefpasses sehr ähnlich. Dieser wird ja durch das Produkt  $R \cdot C = \tau$  beschrieben. Das  $\tau$  ist diejenige Zeit in der das Ausgangssignal 63% des Eingangssignals (Einheitssprung) erreicht hat. Die Grenzfrequenz eines solchen Tiefpasses ist bekanntermaßen:

$$f_g = \frac{1}{2\pi\tau}$$

Um jetzt die Grenzfrequenz dieses exponentiell geglätteten Mittelwertes zu bestimmen, muss ein Zusammenhang zwischen  $\tau$  und  $K$  gefunden werden.

Betrachtet man den Einheitssprung, dann wird:

$$Y_{(0)} = 0$$

$$Y_{(1)} = K$$

$$Y_{(2)} = K \cdot (1-K) + K$$

$$Y_{(3)} = [K \cdot (1-K) + K] \cdot (1-K) + K = K \cdot [(1-K)^2 + (1-K)^1 + (1-K)^0]$$

$$Y_{(4)} = Y_{(3)} \cdot (1-K) + K = K \cdot [(1-K)^3 + (1-K)^2 + (1-K)^1 + (1-K)^0]$$

Dies ist offenbar eine geometrische Reihe:

$$Y_{(m)} = K * \sum_{n=1}^{n=m} (1-K)^{(n-1)}$$

In mathematischen Formelbüchern findet man dafür die Summenformel:

$$S = 1 - (1-K)^m$$

Es ist nun zu berechnen, bei welchem „m“ die Summe S genau 0,63 beträgt.

$$(1-K)^m = 1 - 0,63 = 0,37$$

$$m = \frac{\ln 0,37}{\ln(1-K)} = -\frac{1}{\ln(1-K)} \approx \frac{1}{K} \quad \text{für } K \ll 1$$

Die Zeitkonstante  $\tau$  des exponentiell geglätteten Mittelwertes ist also  $m * T_a$ , wobei  $T_a$  die Abtastperiode darstellt.

Die Grenzfrequenz ergibt sich somit:

$$f_g = \frac{1}{2\pi m T_a}$$

oder für kleine K ( $< 0,01$ ):

$$f_g = \frac{K}{2\pi T_a}$$

Somit lässt sich der digitale Hochpass implementieren, indem man den exponentiell geglätteten Mittelwert vom Signal abzieht.

$$f_g = 0,16 \text{ Hz}, T_a = 1/256 = 0,0039 \text{ s}$$

damit ergibt sich mit obiger Formel:

$$K = 0,004$$

Nachdem es beim aktuellen Mikrocontroller keine floating point unit gibt, muss man Bruchzahlen mit einer Integerzahl darstellen. Dazu wird mit  $0x10000$ (hex) oder 65536 multipliziert. Dies hat den Vorteil, dass der „richtige Wert“ durch einen Shift-Befehl um 16 Bit, der nur sehr wenig Rechenzeit benötigt, wiederhergestellt werden kann.

Damit ergibt sich das normierte  $K_n$ :

$$K_n = 262$$

$$(1-K)_n = 65274$$

Für die Berechnung des exponentiell geglätteten Mittelwert sind nur 2 Programmzeilen nötig:

```
tmp = Sld_Avg[i]*65274 + 262*((*tempsumptr)<<2);  
Sld_Avg[i] = tmp>>16;
```

Nachdem es 24 Messkanäle gibt, gibt es auch einen Buffer für 24 Signal Mittelwerte (Sld\_Avg[24]). Diese Mittelwerte müssen aus numerischen Gründen 64Bit Werte sein. Ebenso die Hilfsvariable „temp“. Der Inhalt von „tempsumptr“ ist der jeweilige Kanalwert, der aus 32 aufsummierten 12 Bit Abtastwerten eines Kanals entstanden ist. Zur besseren numerischen Auflösung wird dieser mit 4 multipliziert bzw. um 2 Bit hinaufgeschiftet. Die endgültige Berechnung des jeweiligen Kanalwertes inklusive Hochpassfilterung und Kalibrierung findet in wenigen Zeilen statt:

```
tmp_data = (long)((((int*)tempsumptr++)>>4) * (*(u16*)calptr++)>>12) - (int)(Sld_Avg[i]/8);  
if (tmp_data > 0x00007FFF) tmp_data=0x7FFF;  
if (tmp_data < -32768) tmp_data=0x8000;  
*eeg_data_ptr = (short)tmp_data;
```

Dabei sind:

tmp_data	Ausgangswert (32Bit)
tempsumptr	Pointer auf 32 fach aufkummulierten Abtastwert(32Bit)
calptr	Pointer auf Kanal-Kalibrationswert (0x8000 =1,0)(16Bit)
Sld_Avg[]	Buffer mit den 24 Kanal-Mittelwerten (64 Bit)
eeg_data_ptr	Pointer auf die EEG Messdaten Datenstruktur (16Bit)

Das Ergebnis tmp\_data wird noch auf 16Bit Überlauf geprüft und eventuell begrenzt. Danach wird der endültige 16Bit Kanalwert mit dem „eeg\_data\_ptr“ in die Messdaten Datenstruktur eingefügt.

Da dieser Vorgang für alle 24 Kanäle stattfindet, werden die entsprechenden Pointer jeweils inkrementiert, bzw. der Bufferindex erhöht und eine Schleife 24mal durchlaufen.

## 4. Layout der Leiterplatte

### 4.1 Grundlegende Überlegungen

Nach der Erstellung des Schaltplans erhält man vom hier verwendeten Leiterplatten CAD (PADS) die Anzahl der verwendeten Bauteile (660). Diese bestimmt einmal in groben Zügen die Fläche der Leiterplatte. Ein weiteres Kriterium waren hier der Platzbedarf und die Form der standartisierten Anschlussbuchsen für die EEG-Elektroden. Für die Anzahl der Leiterplattenlagen ist hauptsächlich die maximale Anzahl und Abstand (pitch) von Pins an einem Bauelement (z.B. CPU) ausschlaggebend, weiters auch die Dichte von Bauteilen. Als Faustregel würde ich behaupten, dass fine-pitch Bauteile bis 64 Pins gerade noch auf einer 2-lagigen Leiterplatte geroutet werden können, darüberhinaus sind dann 4 oder mehr Lagen vorzusehen. Aus elektrotechnischer Sicht beeinflussen auch noch Dinge wie durchgehende Massefläche und Versorgungslage die Anzahl der Leiterplattenlagen.

Um fertigungstechnisch geringe Kosten zu erzielen wäre eine einseitige Bestückung zumindest von SMD Bauteilen vorzuziehen.

Ein weiteres Kriterium für Größe und Form der Leiterplatte ist, ob ein käufliches, fertiges Gehäuse für das Gerät verwendet wird, oder ein speziell angefertigtes.

Man sieht, dass hier viele Dinge schon vor dem Routen der Leiterplatte beachtet werden müssen, was eine gewisse Erfahrung erfordert.

Die endgültige Größe der Leiterplatte konnte durch versuchsweises Platzieren der "sperrigsten" Bauteile wie Buchsen/Stecker, CPU, Batteriefach herausgefunden werden.

Die Lagenanzahl wurde auf 4 festgelegt, wodurch aber auf eine durchgehende Versorgungslage verzichtet werden musste, die aber in diesem Fall nicht allzuviel gebracht hätte. Stattdessen wurde die Versorgungsspannung in einer netzartigen Struktur, mit jeweils einem 100nF Stützkondensator bei jedem aktiven Bauteil, ausgeführt.

Für das Gehäuse konnte kein käufliches gefunden werden, das auch nur annähernd geeignet gewesen wäre. Daher wurde beschlossen, hier aus Kunststoff-Plattenmaterial mit einer CNC Fräsmaschine ein gerätespezifisches Gehäuse herzustellen. Dies ist erstens auch bei kleinen Stückzahlen relativ kostengünstig und zweitens können Änderungen sehr leicht durchgeführt werden. Spritzgussgehäuse haben zwar einen gewissen "Charme", allerdings lohnt sich aufgrund der sehr teuren Formen diese Fertigungsweise nur für sehr große Stückzahlen(>10000).

## 4.2 Ausführung

Anhand der im vorigen Kapitel erwähnten Probeplatzierungen wurden die Abmaße der Leiterplatte mit 125x80mm festgelegt.

Zunächst wurden nun alle Buchsen/Stecker und das Batteriefach auf der Rückseite der Platine platziert. Für ein Buchsenpaar wurde dann das dazugehörige Analogverstärkerpaar auf der Vorderseite geroutet, wobei zu beachten war, dass das Layout dieses Analogverstärkerpaar nicht breiter als das Buchsenpaar sein durfte, weil man es sonst nicht anreihen könnte.

Da es 10 gleiche Analogverstärkerpaare gibt, konnte das Routen der übrigen, unterstützt durch das CAD-System, automatisiert werden, was sehr viel Aufwand erspart. Nachdem die Analogverstärker geroutet waren, wurden dann noch die Versorgungsleitungen verlegt.

Als nächstes wurde dann der 2x12Kanal Multiplexer und die CPU platziert. Der Multiplexer und die CPU sollten möglichst nah zueinander platziert werden, um eventuelles Einstreuen von Störsignalen zu unterbinden bzw. zu reduzieren.

Letzteres ist generell ein Problem, wenn analoge und digitale Schaltkreise auf einer Leiterplatte vorhanden sind. Es ist durch geeignetes Layout sicherzustellen, dass nicht etwa hochfrequente Ströme über die Masse-Lage nahe der Analogverstärker vorbeiführen. Daher sind analoge und digitale Schaltkreise samt ihren Masse-Lagen strikt zu trennen. Die Massen werden dann direkt bei den Analogeingängen der CPU miteinander verbunden.

In weiterer Folge wurden dann noch der Ethernetbaustein und die Stromversorgungen geroutet.

Eine sehr anspruchsvolle Aufgabe stellt das Routen des isolierten DC/DC Konverters und der nicht auf der Patientenseite befindlichen Schaltungsteile dar. Hierbei muss akribisch darauf geachtet werden, dass die vorgeschriebenen Abstände bzw. Kriechstrecken eingehalten werden. Die beiden Ethernet-Trafos, die die Aufgabe der galvanischen Trennung übernehmen, wurden auf der Rückseite der Leiterplatte platziert. Da es an einer Stelle unmöglich war, die 8mm Kriechstrecke einzuhalten, wurde dort ein Schlitz mit 2mm Breite in die Leiterplatte gefräst, um so zu einer zusätzlichen Luftstrecke zu kommen, wodurch die Sicherheitsbestimmungen wieder eingehalten wurden.

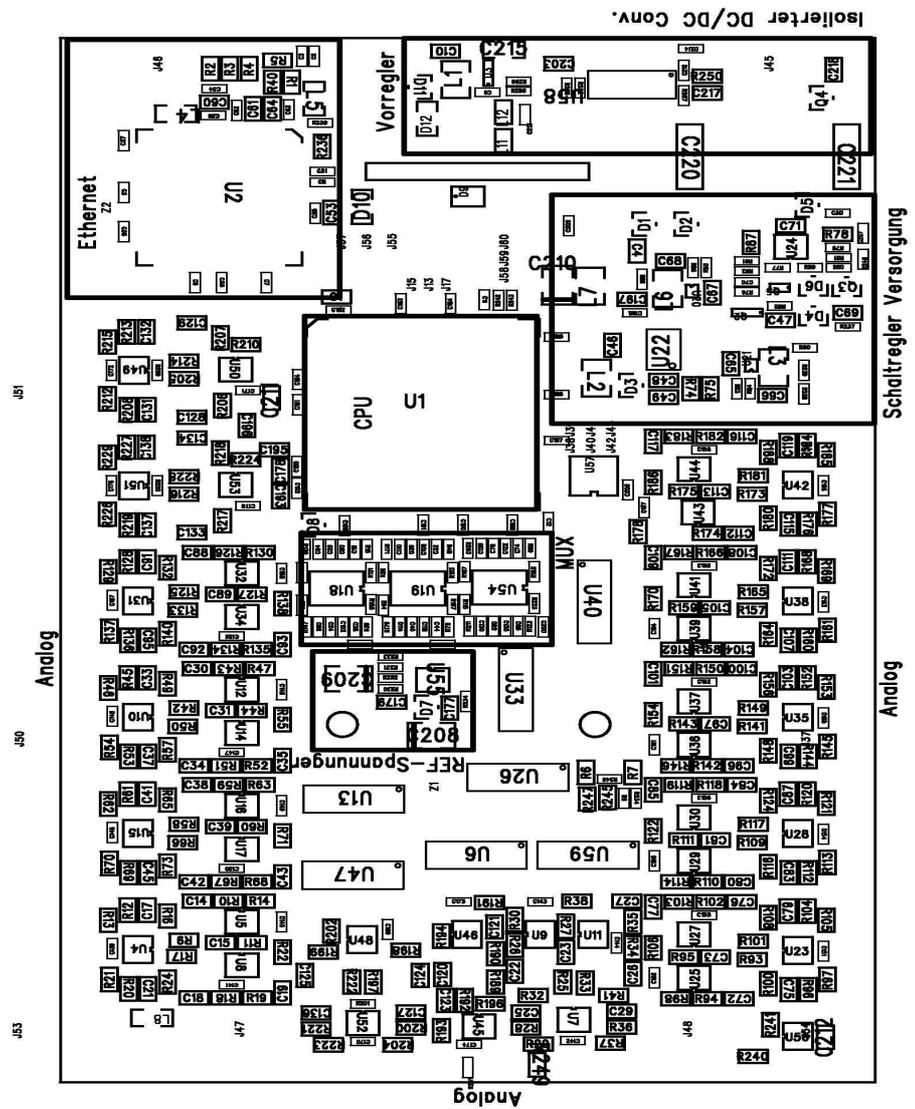


Bild 4.1 SMD Bestückungsseite der Leiterplatte mit umrandeten und bezeichneten Baugruppen

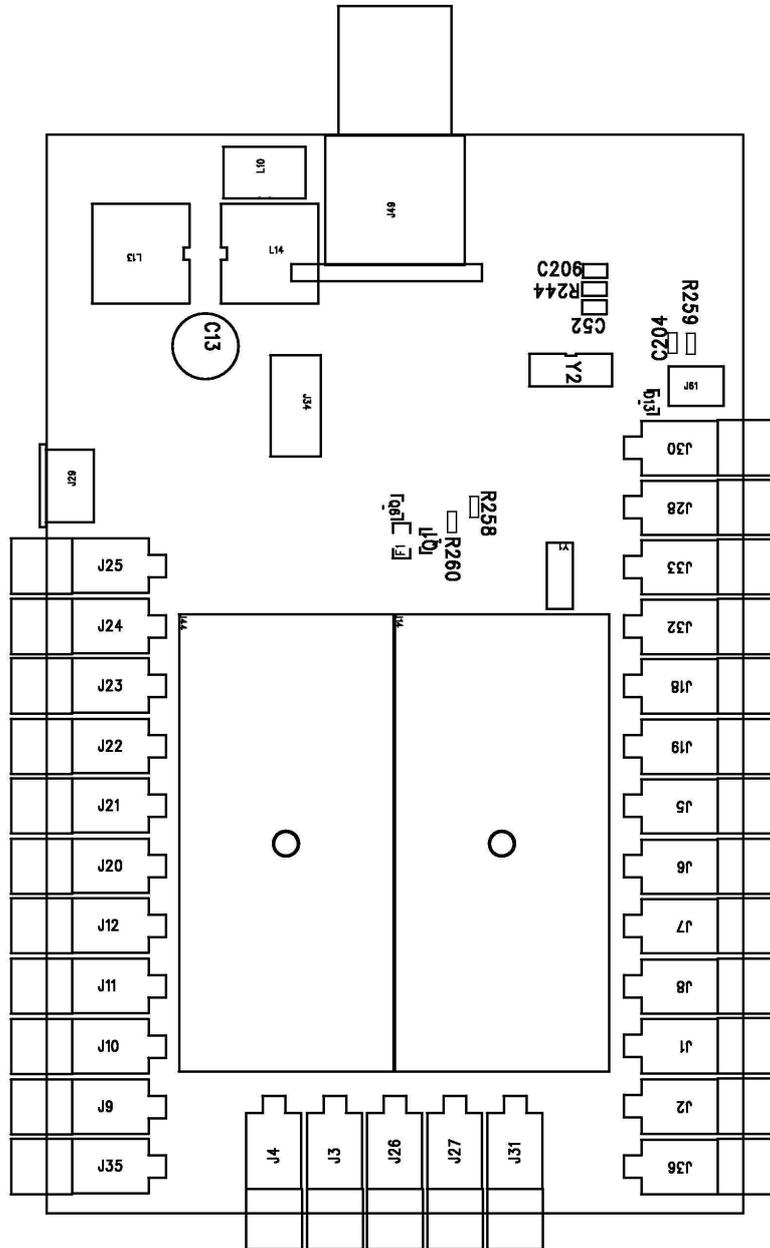


Bild 4.2 Rückseite der Leiterplatte mit hauptsächlich bedrahteten Bauteilen

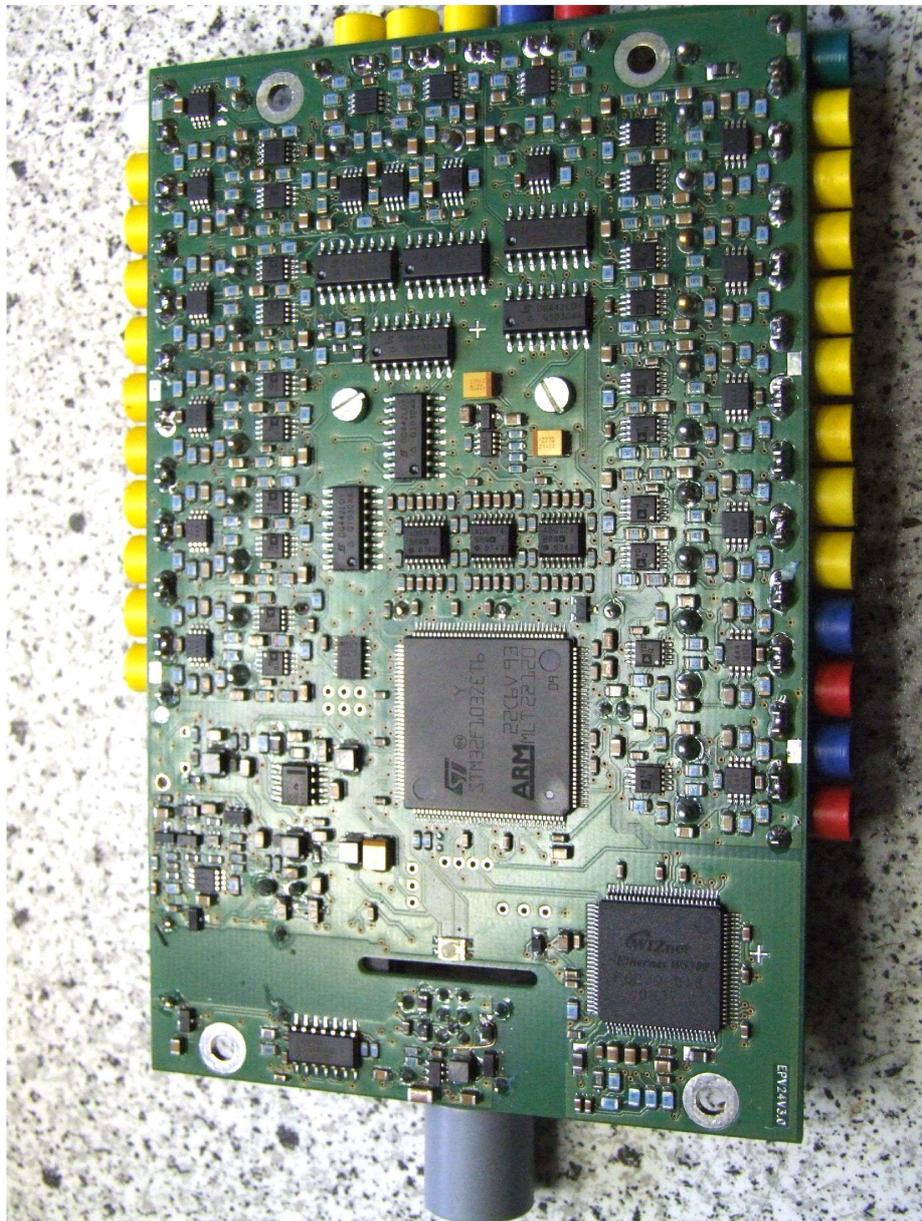


Bild 4.3 Fertig bestückte Leiterplatte

## 5. Gehäuse

Wie im vorhergehenden Kapitel schon erwähnt, wird ein gerätespezifisches Gehäuse angefertigt. Als Ausgangsmaterial werden dabei Kunststoffplatten aus POM (Polyoxymethylen) in der Stärke 16mm und 10mm verwendet. POM zeichnet sich durch eine geringe Wasseraufnahme, hohen Schmelzpunkt und sehr gute mechanische Stabilität und Bearbeitbarkeit aus. Damit können auch Gewinde mit guter Haltbarkeit in das Material geschnitten werden.

Da von der Sicherheitsnorm eine Kriechstrecke von 4mm von berührbaren Isolationsteilen zum Patientenstromkreis gefordert wird, wurden die äußeren Abmessungen des Gehäuses um 8mm größer als die Leiterplatte.

Das dünnere Unterteil wird zur Abdeckung der SMD Bauteile verwendet, wobei einfach eine entsprechende Vertiefung in das Plattenmaterial eingefräst wird.

Das Licht der 2-färbigen Status-Leuchtdiode wird mittels Lichtleiter aus Polycarbonat durch die Abdeckung geführt.

Das Oberteil deckt dann die Anschlussbuchsen und weitere hohe Bauteile ab. Dazu wurden wieder entsprechende Vertiefungen in das Material gefräst.

Einige Überlegungen sind auch für den Batteriefachdeckel notwendig. Eine Öffnungsmöglichkeit ohne Werkzeug ist unzulässig, weil dann Teile des Patientenstromkreises "zufällig" berührt werden können. Daher wird dieser mit einer M4 Kunststoffschraube geschlossen. Durch entsprechende Einfräsungen an Oberteil und Deckel ergibt sich dann ein geschlossenes Gehäuse mit glatten Außenflächen.

Ein weiteres Teil ist dann noch das Verschlussplättchen, das den Debug/Downloadstecker abdeckt. Dieses wird aus 3mm starkem POM Material gefertigt und mit einer M3 Kunststoffschraube seitlich befestigt.

Die Teile sind so konstruiert, dass eine sehr schnelle Montage möglich ist. Für den Zusammenbau wird zunächst die fertig bestückte Leiterplatte in das Oberteil geschoben. Dann wird das Unterteil aufgesetzt und mit 4 Stück Kunststoff-Senkkopfschrauben befestigt. Danach werden noch Batteriefach und Steckerabdeckung angebracht. Die Aussenflächen des Gehäuses haben eine Vertiefung von 0,5mm, in welche dann noch die Beschriftungsfolien geklebt werden.

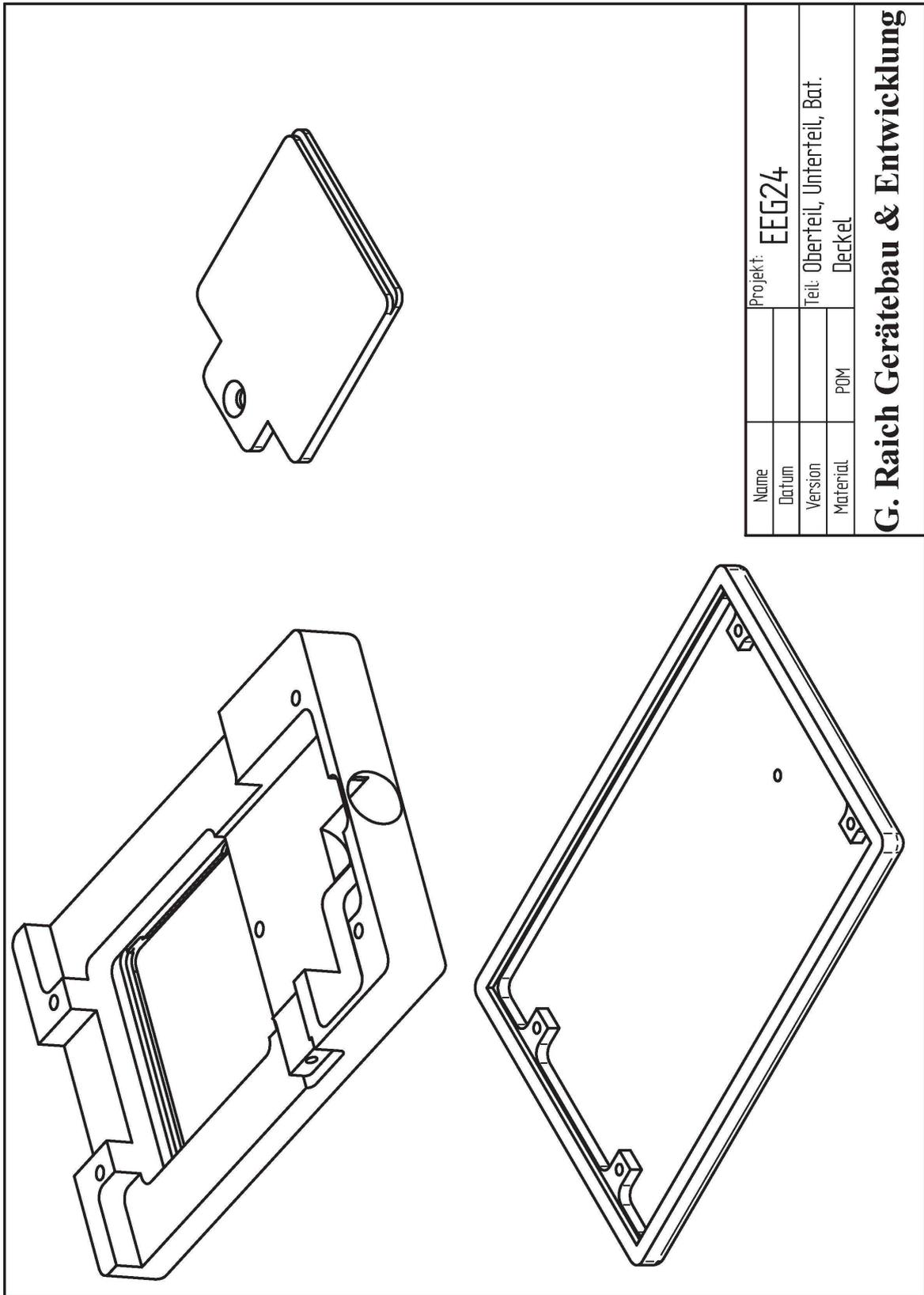


Bild 5.1 Gehäuseteile (Innenansicht)

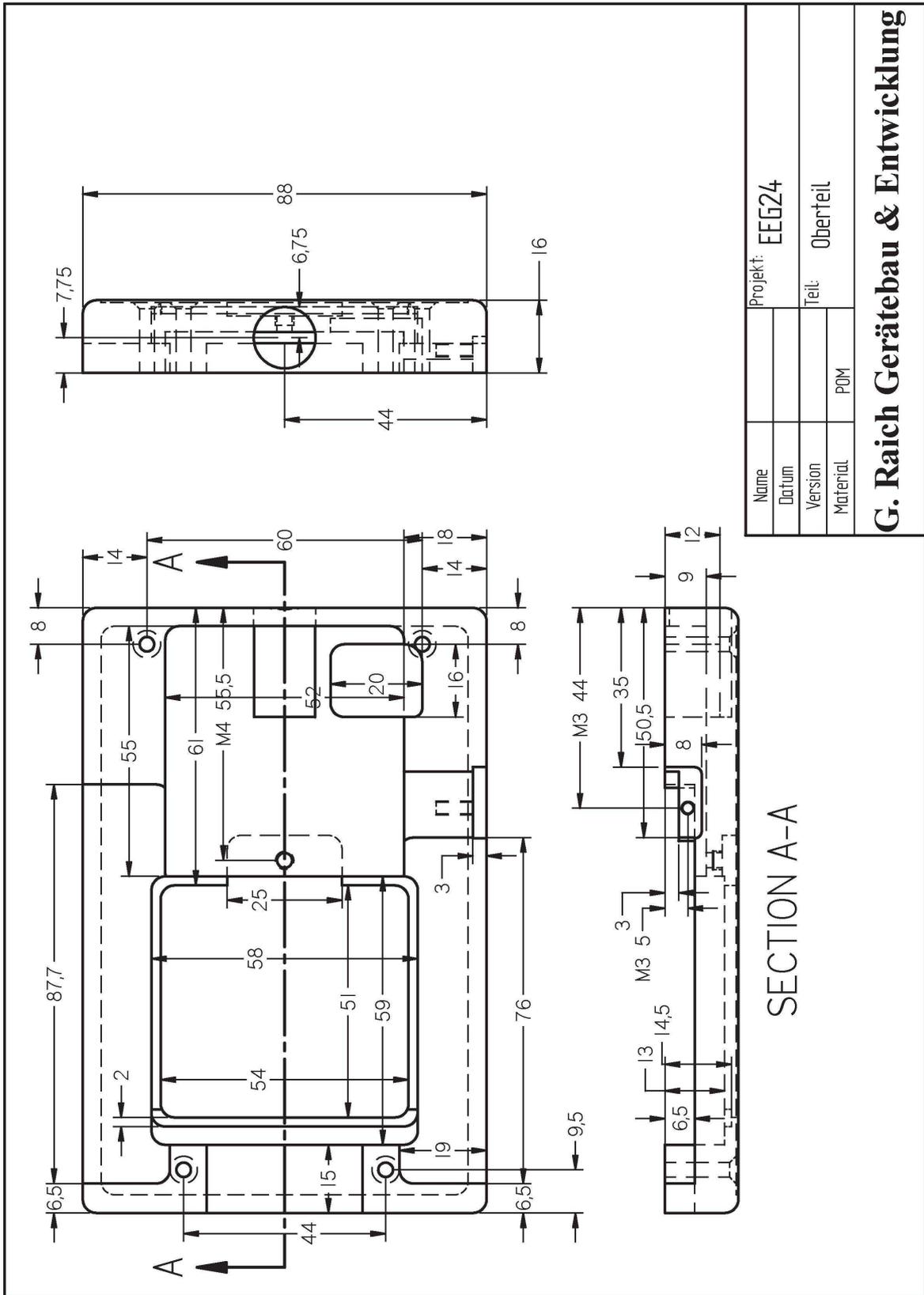


Bild 5.2 Werkstattzeichnung Oberteil

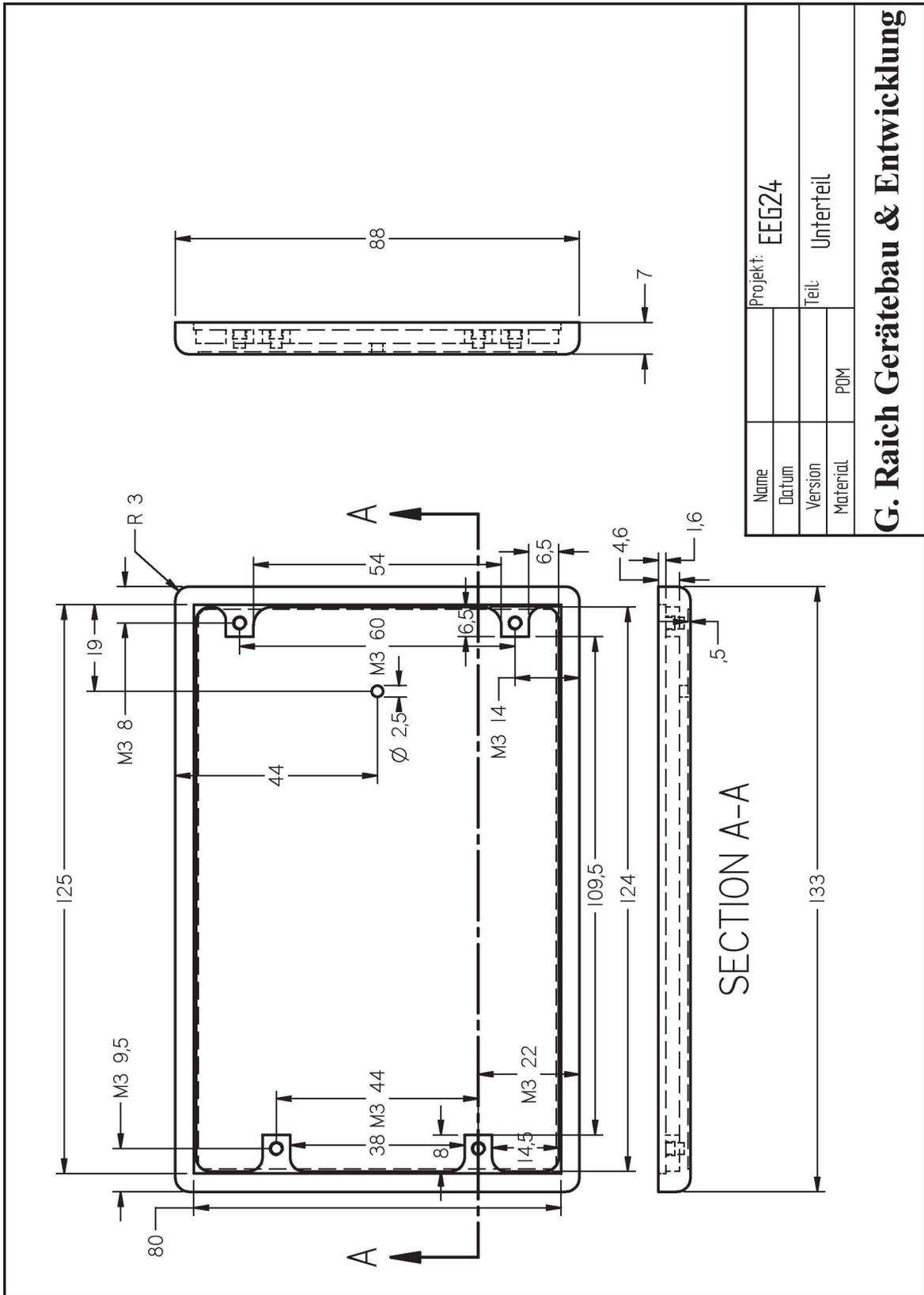


Bild 5.3 Werkstattzeichnung Unterteil

## 6. Messwerte, Diskussion, Ausblick

Im folgenden werden die erzielten Ergebnisse dargestellt und besprochen. Für die Messungen wurde das von mir ursprünglich für das 32 Kanal EEG Gerät entwickelte Interface und Testprogramm für dieses 24Kanal Gerät adaptiert. Die Amplituden werden in Digits dargestellt, wobei 32000 Digits 1mV (Messbereich) entsprechen.

Das obere schwarze Fenster stellt das Signal im Zeitbereich dar, das untere weiße Fenster zeigt die dazugehörige FFT.

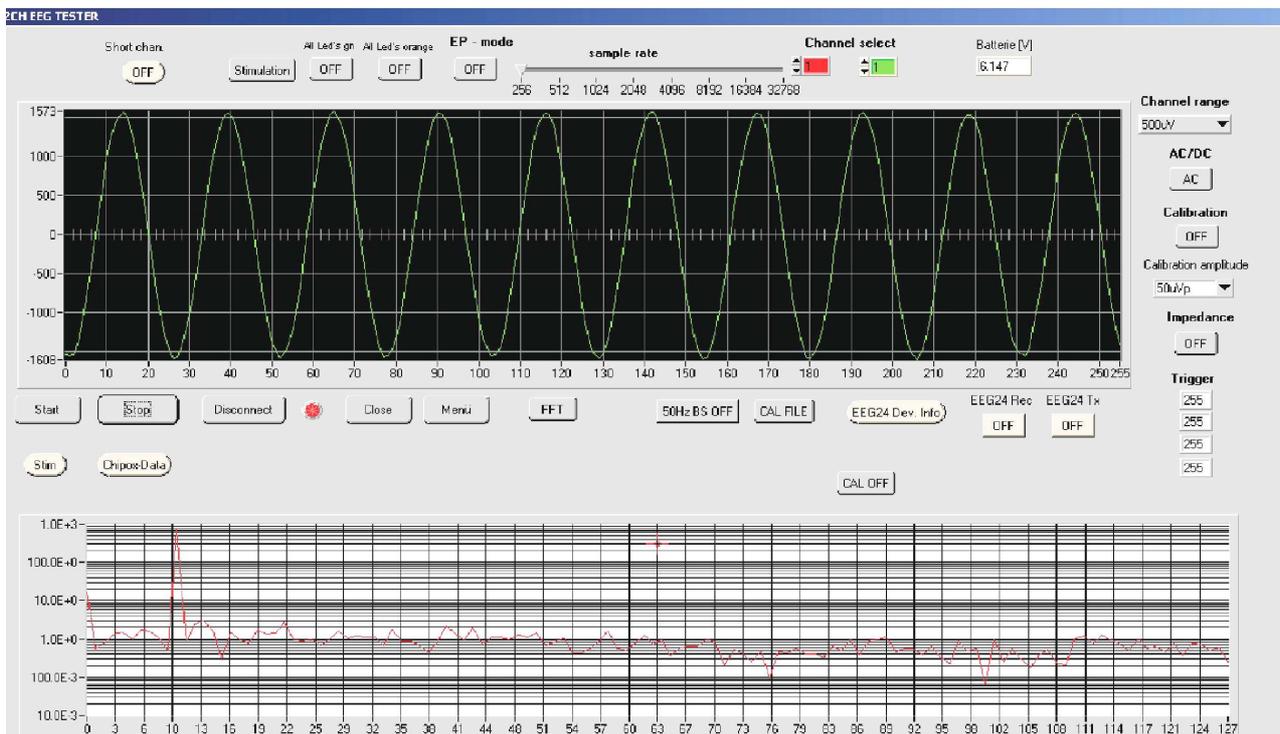


Bild 6.1

Kalibriersignalsinus mit 10Hz /  $50\mu\text{V}_{\text{peak}}$  am Eingang des Verstärkers  
Leichte Verformungen an den Sinusscheiteln infolge Grundrauschens  
Spektrale Analyse des Signals weist keine unerwünschten Peaks auf  
Offset etwa 16 Digits oder  $0,5\mu\text{V}$   
Signalqualität insgesamt akzeptabel

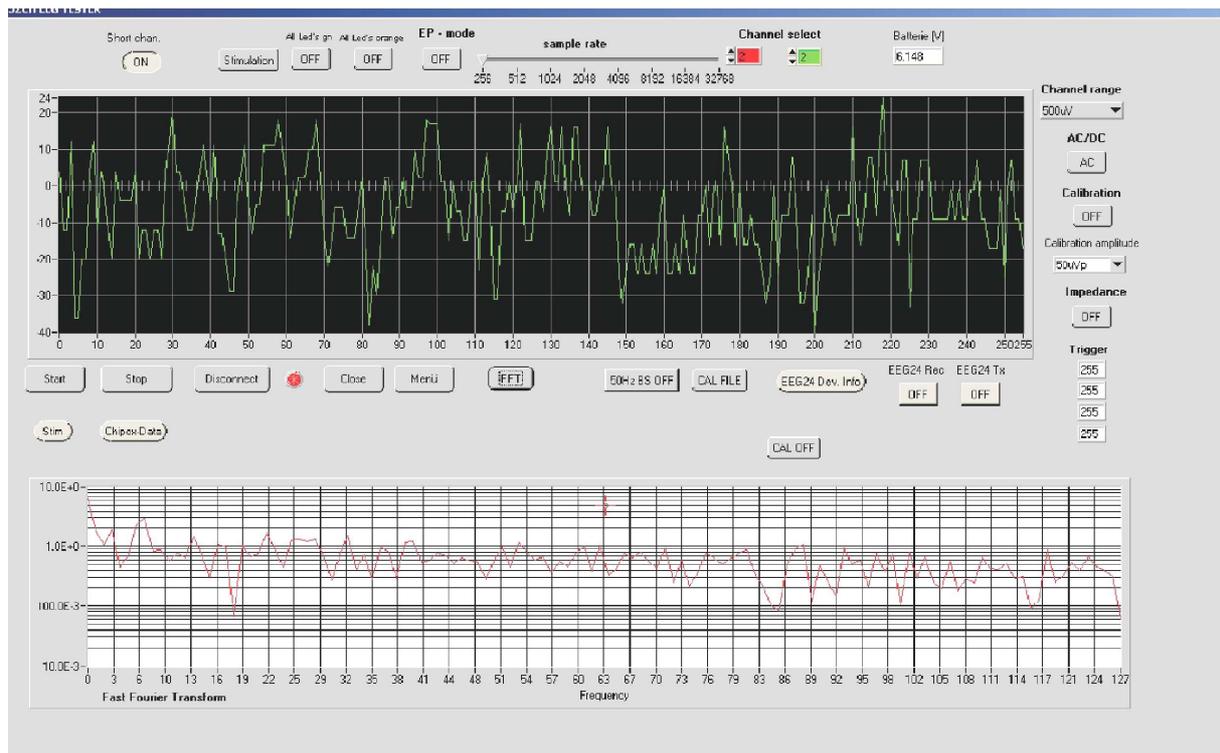


Bild 6.2

Rauschspannungsmessung, Eingänge kurzgeschlossen

Das Rauschen beträgt ca. 60 Digits entsprechend  $1,9\mu V_{pp}$ .

Es ist damit etwas höher als angenommen, der Grund liegt im relativ flachen Übergangsbereich des 70Hz Tiefpassfilters

In der Spektralanalyse findet sich ein Peak bei 6Hz, vermutlich das alias Signal der 5. Netzerwelle.

Weiters wurde noch die Gleichtaktunterdrückung gemessen. Dazu wurde für die referenziellen Kanäle ein Sinus-Signal mit  $55\text{Hz} / 200\text{mV}_{pp}$  zwischen virtuellen GND und dem Referenzanschluss eingespeist. Die Messkanäle wurden mit dem Referenzanschluss verbunden. Für die referenziellen Kanäle wurde so eine Gleichtaktunterdrückung von 78dB gemessen. Dies blieb doch etwas unter den Erwartungen und ist noch Gegenstand weiterer Verbesserungen.

Für die differentiellen Kanäle wurden die Eingänge kurzgeschlossen und das Gleichtaktsignal zwischen virtuellem GND und den Eingängen eingespeist. Dabei konnte eine Gleichtaktunterdrückung von 90dB gemessen werden.

Ebenso wurde auch noch die Stromaufnahme des Gerätes im batteriegespeisten Rekordermodus gemessen. Dabei ergab sich ein Wert von ca. 31mA bei 6,4V Batteriespannung, was einer Leistung von 198mW entspricht und sich fast exakt mit den gerechneten Werten deckt. Probemessungen mit Alkali-Zellen des Herstellers Duracell brachten Aufnahmezeiten von 28-29 Stunden. Mit NiMH Akkus konnte eine Laufzeit von 25-26 Stunden erzielt werden.

Als letzter Punkt wurde noch die Bandbreite der Ethernetverbindung beim Auslesen der Messdaten getestet. Dabei konnte ein Datenstrom von 3,5MByte/s erzielt werden. Damit kann eine 24 Stunden Messung mit ca. 1,6GB Daten in etwa 8 Minuten ausgelesen werden.

Von diesen 24Kanal EEG-Rekordern wurden bis heute 25Stk. gefertigt und sind im Einsatz. Dabei ist es da und dort zu Problemen gekommen, die hauptsächlich mit der Software zusammenhängen. Das Problem für den Entwickler ist jenes, dass nicht von vornherein alle möglichen Zustände und Einwirkungen auf das Gerät berücksichtigt werden können. Damit ist man auf die Rückmeldungen der Anwender angewiesen, um restliche Schwachstellen des Gerätes zu beheben. Dieser Prozess kann durchaus auch mehrere Jahre dauern, wobei grobe Fehlfunktionen rasch behoben werden müssen, um den Kunden nicht allzusehr zu verärgern.

Als weiteren Ausblick sollte die hier verwendete CPU mit dem ARM M3 Core durch eine weitgehend pincompatible CPU mit dem ARM M4 Core mit floating point unit (FPU) ersetzt werden. Diese sind erst seit kurzem am Markt und bieten neben der FPU auch höhere Taktfrequenzen bei niedrigerem Stromverbrauch. Damit könnten auch noch IIR Tiefpassfilter und/oder ein Brummfilter gerechnet werden.

Auch auf der Analogseite wären Verbesserungen angebracht, um den Rauschpegel noch zu senken oder die Gleichtaktunterdrückung zu erhöhen.

Alles in allem ist hier doch ein brauchbares Gerät entwickelt worden, das auch als Grundlage für weitere Entwicklungen dient.

Die Arbeitsaufwendungen wurden festgehalten, wobei insgesamt bis heute zur Leiterplattenrevision 3.1 etwa 350 Arbeitsstunden angefallen sind. Dabei ist das Verhältnis von Hardwarearbeiten zu Softwarearbeiten etwa 30% zu 70%.