

INHALTSVERZEICHNIS:

1.	Einführung	3
2.	Das R- 2R Netzwerk	4
3.	Der AD7111 von ANALOG DEVICES	5
4.	Die Entwicklung der DA- Karte	6
5.	Der Mikrokontroller PIC16F84	7
5.1	Der Mikrokontroller PIC16F84	8
6.	Abgleich und Inbetriebnahme der DA- Karte	9
7.	Messdiagramm $g = f(f)$	
8.	Datenbusplan μ Controller/ Attenuator	11
9.	Bedienungsanleitung	12
10.	Mechanische Zeichnung Frontplatte	13
10.1	Mechanische Zeichnung Netzbuchse	14
11.	Schaltplan Netzteil	15
11.1	Bestückungsplan Netzteil	16
12.	Schaltplan μ Controller	17
12.1	Bestückungsplan μ Controller	18
13.	Schaltplan Attenuator	19
13.1	Bestückungsplan Attenuator	20
14.	Flussdiagramm	21
14.1	Assemblerprogramm	22
14.2	Assemblerprogramm	23
14.3	Hexfile	24

Anhang:

A.....	Datenblatt LF411
B.....	Datenblatt AD7111
C.....	Layout's
D.....	Literaturnachweis

1. Einführung

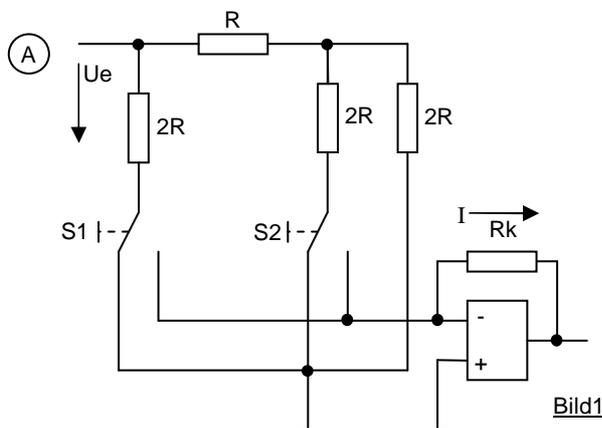
Der Einsatz von hochintegrierten Schaltungen nimmt immer mehr zu. Besonders davon betroffen sind die "klassischen" Gebiete der Nachrichtentechnik, speziell der Übertragungstechnik. So ist ISDN (Integrated Service Digital Network) in aller Munde. Dieses System bietet die Möglichkeit Sprache, Daten Text und Bilder in Digitalisierter Form über ein einziges Fernsprechnetz zu übertragen. Möglich wird dies nur durch den konsequenten Einsatz der μ P- Technik. Dabei steht der Gedanke im Vordergrund mit Hilfe dieser Technik sowohl die Digitalisierung analoger Signale und deren nachrichtentechnische Übertragung, als auch eine Steigerung der Leistungsdaten in Bezug auf Geschwindigkeit und Nutzungskomfort gegenüber konventioneller Übertragungstechnik zu realisieren. Dabei gelten die Bemühungen der Forschungs- und Entwicklungsabteilungen insbesondere den Schnittstellen zwischen analoger und digitaler Technik.

Meine Arbeit greift hier ein solches - sehr bescheidenes - Teilstück einer analog-digital Verknüpfung heraus: Das LOGDAC Projekt.

2. Das R- 2R Netzwerk

Wie schon eingangs erwähnt besteht das Hauptproblem der Nutzung von μP -Systemen in nachrichtentechnischen Anwendungen darin, eine Schnittstelle zwischen den analogen und den digitalen Signalen wie sie ein Prozessor versteht, zu schaffen.

Eine Lösungsmöglichkeit wie sie in diversen IC realisiert wird ist ein sogenanntes R-2R Netzwerk. Ein solches Netzwerk ist auch im AD7111 der Firma ANALOG DEVICES integriert. Bild 1 stellt das Prinzip eines solchen Netzwerkes in seiner einfachsten Form dar:



Beim Betrachten des Netzwerkes (Bild1) vom Punkt A ergibt sich unabhängig von den Schalterstellungen der Widerstand $2R$ (Grund: Der gegengekoppelte Operationsverstärker wirkt als "virtueller Nullpunkt"). Der Gesamtwiderstand vom Eingang der Spannungsquelle ergibt sich zu R . Der von der Spannungsquelle fließende Strom wird an jedem Knotenpunkt gleich geteilt. Abhängig von der Stellung der Schalter wird der Strom gegen Masse abgeleitet oder aufsummiert. Das Ergebnis ist eine binäre Gewichtung des Stromes. Der Ausgangsstrom von Bild1 errechnet sich allgemein mit:

$$I = (U_{\text{ref}}/R) * (\frac{1}{2} + \frac{1}{4} \dots +n)$$

Daraus ergibt sich, dass am Summationspunkt alle I 's aufaddiert werden und wir einen Strom- Spannungswandler erhalten.

Neben der Stabilität und der absoluten Genauigkeit der Referenzspannungsquelle U_{ref} stellen die Widerstände im vorliegenden Fall das schwächste Glied in der Kette dar, deren mit zunehmendem Absolutwert wachsende Wertetoleranzen einerseits wie deren Temperaturverhalten andererseits weitgehend die Qualitätseigenschaften des DAU bestimmen. Ein großer Vorteil unseres Leitungsnetzwerkes gegenüber anderen Verfahren besteht darin, dass sich absolute Fehler, hervorgerufen durch Fertigungstoleranzen nicht gravierend auswirken. Außerdem ist die Eingangsimpedanz des Leiternetzwerkes unabhängig vom Schaltzustand, also Ausgangsspannungsunabhängig konstant. Temperatureinflüsse sind nicht so gravierend, weil nur zwei Typen von Widerstandswerten implementiert werden müssen, die so ausgewählt werden können, dass sich ihr Temperaturverhalten gegenseitig kompensiert.

3. Der AD7111 von ANALOG DEVICES

Der AD7111 der Firma Analog Devices ist ein multiplizierender DA- Wandler, oder auch schlicht logarithmischer Dämpfungsregler. Er kann ein analoges Eingangssignal in einem Bereich von 0 bis -88,5 dB in 0,375 dB Schritten abdämpfen. Dabei ist der Dämpfungswert über ein 8 Bit Datenwort einzustellen, das in einen Zwischenspeicher mit mikroprozessorkompatiblen Steuersignalen geschrieben werden kann. Ein High Pegel muss mindestens 2,4V und ein Low Pegel darf maximal 0,8V betragen. Die Eingangsspannung am Analogeingang darf maximal +/- 35V groß sein.

Prinzip:

Das 8 Bit Datenwort wird über die Eingänge D0 bis D7 in einen Zwischenspeicher eingelesen. Nach dem Anlegen des Datenwortes wird über die Steuersignale CS/ und WR/ das Datenwort in den Zwischenspeicher geschrieben. Die Daten werden mit der steigenden Flanke des WR/ Signals übernommen. Nach dem Zwischenspeicher wird das eingelesene Datenwort über eine interne Logik zu einem 17 Bit Datenwort gewandelt und anschließend dem R- 2R Netzwerk zugeführt.

Um den AD7111 an einem μP - System anzuschließen, muss das Timing der Steuersignale und des anzulegenden Datenwortes mit dem Timing des μP - Systems verglichen werden. Die Minimalschaltzeiten müssen dabei unbedingt vom Prozessor eingehalten werden.

4. Die Entwicklung der DA- Karte

Die wichtigste Grundlage beim Design solcher Anwendungen stellt das Studium der Hersteller- Datenblätter dar. Wie schon erwähnt, muss sichergestellt sein, dass der Prozessor entsprechend seiner Taktfrequenz Steuersignale liefert, die für die Peripherie ausreichend lang sind.

Der Prozessor PIC16F84 von MICROCHIP wird mit $\frac{1}{4}$ seiner Oszillatorfrequenz von 4MHz getaktet. Die Taktfrequenz beträgt somit 1MHz, $T=1\mu s$. Die minimale Impulsdauer für das WR/ Signal beträgt 500ns. Die Periodendauer T von $1\mu s$ ist somit ausreichend lang um das Steuersignal WR/ anzusprechen. Es ist daher möglich gewesen, den AD7111 direkt an den Mikrokontroller anzuschließen. Der Vorteil dabei ist, dass der Programmieraufwand für das Timing gering ausfällt.

Ein weiterer wichtiger Punkt sind Angaben des Herstellerdatenblatts, die sich auf die Anordnung einzelner Bauelemente auf der Platine beziehen. Um parasitäre Störstrahlungen, Übersprechen von Leitungen und Noise zu begegnen, muss ein Layout die Vorgaben des Herstellers einbeziehen.

Allgemein kann ich meine gewonnenen Erkenntnisse bei der Layouterstellung und Schaltungsinbetriebnahme wie folgt darstellen:

- Herstellerdatenblatt in Bezug auf Anordnung beachten
- Konsequente Masseführung sternförmig (!) auf einen (!) Punkt.
- So kurze Leitungen wie möglich (zusätzliche Induktivität !)
- Um CMOS Latch Up zu vermeiden:
Power vs Ground Abblockung mittels Kondensatoren.
Die Anschlussleitungen der Kondensatoren müssen dabei so kurz wie möglich gehalten werden.
Möglichst große "Masseflächen" auf der Platine.
Die Erfahrungswerte für eine „Fünf- Punkt- Abblockung“ sind 750pF, 0,01 μ F, 1 μ F, 22 μ F. Dabei sollte mit dem kleinsten Wert begonnen werden.
Alle Versorgungsleitungen am Ursprung mit großen C's abblocken (100 μ F).
- Auf reflektionsfreie Abschlüsse achten.

Eine Hauptforderung an den LOGDAC sollte die Möglichkeit sein, unterschiedliche Signalquellen anzuschließen. Diese können sehr verschieden sein und fallweise Gleichspannungsanteile enthalten. Um diese Gleichspannungsanteile auszuschließen ist unmittelbar hinter dem Analogeingang ein Hochpass eingebaut. Dieser führt auf den invertierenden Eingang eines Operationsverstärkers mit einem eingestellten Verstärkungsfaktor $V= 1$. Dieser OP dient als Impedanzwandler um

Signalquellen belastungsunabhängig zu betreiben Allerdings ist das Signal das jetzt an den LOGDAC gelangt, um 180° phasenverschoben. Um das Originalsignal wiederzugewinnen wird daher am Ausgang des LOGDAC wiederum ein OP angebracht, als invertierender Summierverstärker mit $V= 1$ geschaltet. Der verwendete OP (Typ: LF 411 von NSC) muss einen geringen Offset haben, um die Gleichspannungsfehler gering zu halten. Selbstverständlich ist ein OP- Typ mit einem geringen Temperatur- und Langzeitdrift zu wählen. Außerdem war eine schnelle Anstiegsgeschwindigkeit (Slew Rate) erforderlich. Der LF 411 erfüllt alle diese Forderungen und besitzt darüber hinaus FET Eingänge, die den OP besonders hochohmig am Eingang machen.

Im Herstellerdatenblatt des AD7111 findet sich auch ein Layoutvorschlag wie und wo der OP anzubringen ist, um ein Übersprechen vom Eingang zum Ausgang zu vermeiden. Gleichzeitig wird auch darauf hingewiesen, dass die optimale Leistung des AD7111 nur mit einem guten Layout zu erzielen ist. Schaltungsprobleme ergeben sich in erster Linie durch ein schlechtes Platinenlayout, fehlerhafte Ground Beschaltungen oder auch durch einen ungeeigneten OP Typ. Zur zusätzlichen Phasenkompensation der Ausgangskapazität wird ein Kondensator am Ausgangs OP vorgeschlagen.

5. Der Mikrokontroller PIC16F84

Die Mikrokontroller der Firma Microchip (PIC) sind besonders einfache und preiswerte Mikrokontroller. Mit nur wenig Hardware können mit ihnen komplexe Aufgaben gelöst werden. Aber auch sehr einfache Anwendungen, die mit wenigen klassischen Digitalbausteinen realisiert werden könnten, lassen sich oft schneller mit einem einfachen PIC- Mikrokontroller verwirklichen. Die PIC's sind CMOS- Controller mit geringem Leistungsbedarf, die einen internen Programm und Datenspeicher mit Harvard Architektur besitzen.

Die I/O- Ports des PIC16F84: Für die Ansteuerung von externer Hardware, wie sie auch der AD7111 darstellt, besitzt der PIC16F84 zwei Ports. PORTA mit 5 Bit und PORTB mit 8 Bit. Einige Pins haben Doppelfunktionen, d. h. sie können entweder als Port- Pin oder z. B. als Timer- Eingang verwendet werden. In diesem Zusammenhang soll aber nur auf die Portpin Funktionen als Eingang oder Ausgang eingegangen werden. Die Port Pins RA0- RA3 haben alle vier Input Buffer sind TTL kompatibel und die Ausgangstreiber sind in CMOS- Technologie ausgeführt. Port Pin RA4 hat einen Schmitt- Trigger- Eingang und einen offenen Drain Ausgang. RA3 und RA4 verwende ich als Eingang und realisiere über sie die Tastaturabfrage. Wobei die gegenseitige Verriegelung Hardwaremäßig ausgeführt ist. Der PORTB ist ein bidirektionaler 8 Bit Port. An RB0- RB7 sind die 8 Bit Dateneingänge vom AD7111 angeschlossen. Die Ausgangstreiber können im Low- Zustand 25mA und im High Zustand 20mA treiben. RA0 ist mit CS/ belegt und RA1 mit WR/.

Entscheidend für die richtige Arbeitsweise ist das Programm. Ein Programm ist eine Folge von Befehlen (Anweisungen), die im Programmspeicher (ROM) im Maschinencode (0-, 1- Folgen) abgelegt sind. Der Adressbereich des

Programmspeichers geht von 000H bis 03FFH. Bei einem Reset (z. Bsp. Beim einschalten der Spannungsversorgung) wird der Befehlszähler auf 000H gesetzt. An dieser Adresse muss also beim PIC16F84 das Programm beginnen. Mit der Steueranweisung ORG<adresse> wird dem Assembler mitgeteilt ab welcher Adresse der nachfolgende Code im Programmspeicher abgelegt werden soll.

Jedem Portpin ist ein I/O Steuerregister zugeordnet, das TRIS Latch, in dem die Richtung von jedem Pin (Eingang/Ausgang) festgelegt wird. Um die Ports zu Initialisieren braucht man Zugang zu diesen Registern. Das TRISA und TRISB Register befindet sich jedoch in der Registerbank 1 (80H-FFH). Es muss also zunächst auf diese Bank umgeschaltet werden, dies geschieht über das Status-Register bit5 (RP0). Wird in ein TRIS Register eine "1" eingeschrieben, so ist der zugehörige Portpin als Eingang programmiert und der Ausgangstreiber im hochohmigen Zustand. Anschließend wird wieder auf die Registerbank 0 (00H- 7FH) zurückgeschaltet.

Im Einschaltzustand stellt sich das Datenwort 1011010B am PORTB ein. Die beiden Taster T3 und T4 werden abwechselnd abgefragt, durch betätigen eines Tasters kommt es zur Auslösung eines Sprungbefehls und das Bitmuster von Register PB wird an PORTB ausgegeben. Je nach Taster verringert oder erhöht sich der Wert von PB um eins. Wobei beide Mal auf Überlauf geachtet werden muss. Beim betätigen von T4 (RA4) ist es das Zero- Flag das abgefragt wird. Wenn es nicht gesetzt ist springt das Programm zur Marke ANF0 zurück, fragt den Taster 88,5dB (T4) erneut ab und gibt den Inhalt von PB an PORTB aus.

Wird die 0dB (T3) Taste gedrückt steht eine "1" an es kommt zum überspringen vom nächsten Befehl, das Programm arbeitet beim über nächsten weiter. Wird beim verringern der Speicherzelle f um 1 das Ergebnis 0, dann wird der nachfolgende Befehl ignoriert. Das Programm kehrt wieder zum Programmanfang zurück. Zwischendurch wird jeweils das Unterprogramm ZEIT aufgerufen und über einen Bitmanipulations Befehl (BCF/BSF) das Timing für das Steuersignal WR/ erzeugt.

Seite 10 zeigt ein Messdiagramm $g = f(f)$

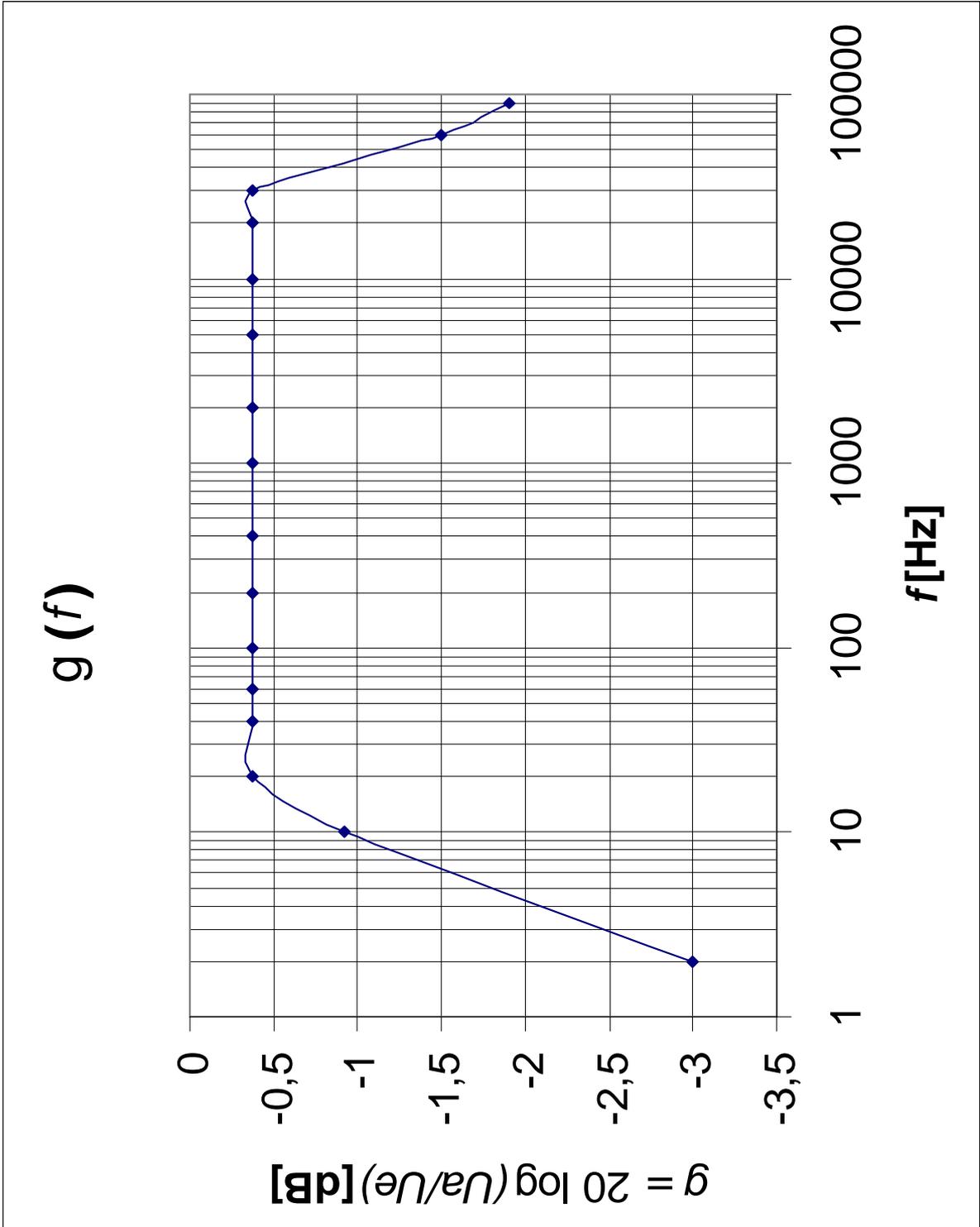
6. Abgleich und Inbetriebnahme der DA- Karte

1) Zu Seite 10 Übertragungsfunktion:

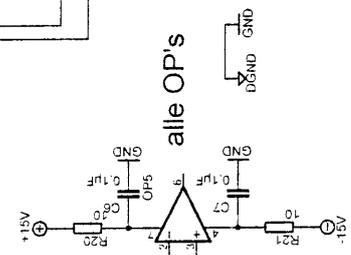
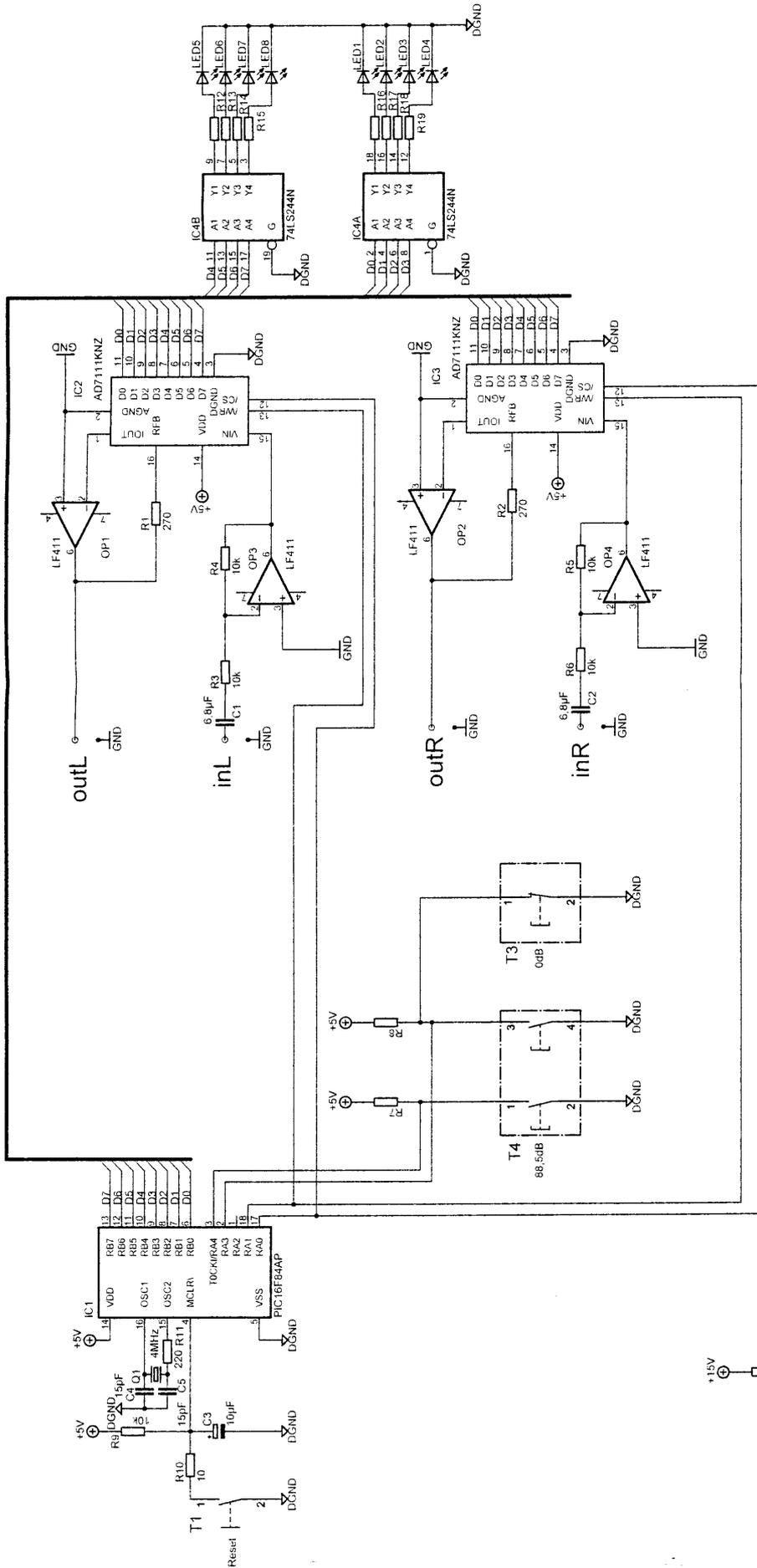
1.1) Mit Hilfe eines Funktionsgenerators (TG550) und einem Oszilloskop wurde die Übertragungsfunktion in Einfachlogarithmischer Darstellung aufgenommen. An den Dateneingang von AD7111 wurde das Datenwort $D0...D7 = 1000\ 0000$ angelegt, was einer Dämpfung von $-0,375\text{dB}$ entspricht. Da die DA- Karte für den NF- Bereich dimensioniert ist, war das Bandpassverhalten zu Überprüfen. Der untere -3dB Punkt stellte sich bei $f \approx 2\ \text{Hz}$ ein. Die obere Grenzfrequenz betrug $\approx 100\text{kHz}$. Somit war ein Linearer Amplituden- und Phasengang im Arbeitsbereich von $15\text{Hz}...20\text{kHz}$ gegeben.

1.2) Nun wurde das Datenwort $D0...D3 = 1111$ und $D4...D7 = 0000$ eingestellt. Dies entspricht einer idealen Abschwächung von $-5,625\text{dB}$. Die Eingangsspannung betrug $20V_{\text{ss}}$, die Messfrequenz 1kHz . Mit dem Oszilloskop gemessen stellte sich eine Ausgangsspannung von $10,4V_{\text{ss}}$ ein. Dies entsprach einem Wert von $-5,68\text{dB}$, die Differenz wurde mit einem Spindeltrimmer auf der DA- Karte ausgeglichen. Dieser Abgleich wiederholte sich bei dem 2. Kanal.

1.3) Zur Linearitätsüberprüfung wurde nun noch ein Abschwächungswert von $-40,125\text{dB}$ eingestellt ($D0...D3 = 1101$ $D4...D7 = 0110$). Am Ausgang stellten sich 195mV_{ss} ein. Der Abgleich wurde nun wiederholt und das Gerät wurde mit diesem eingestellten Wert Probe laufen gelassen.



D00 71



Busplan µController/Attenuator

TITLE: logdac

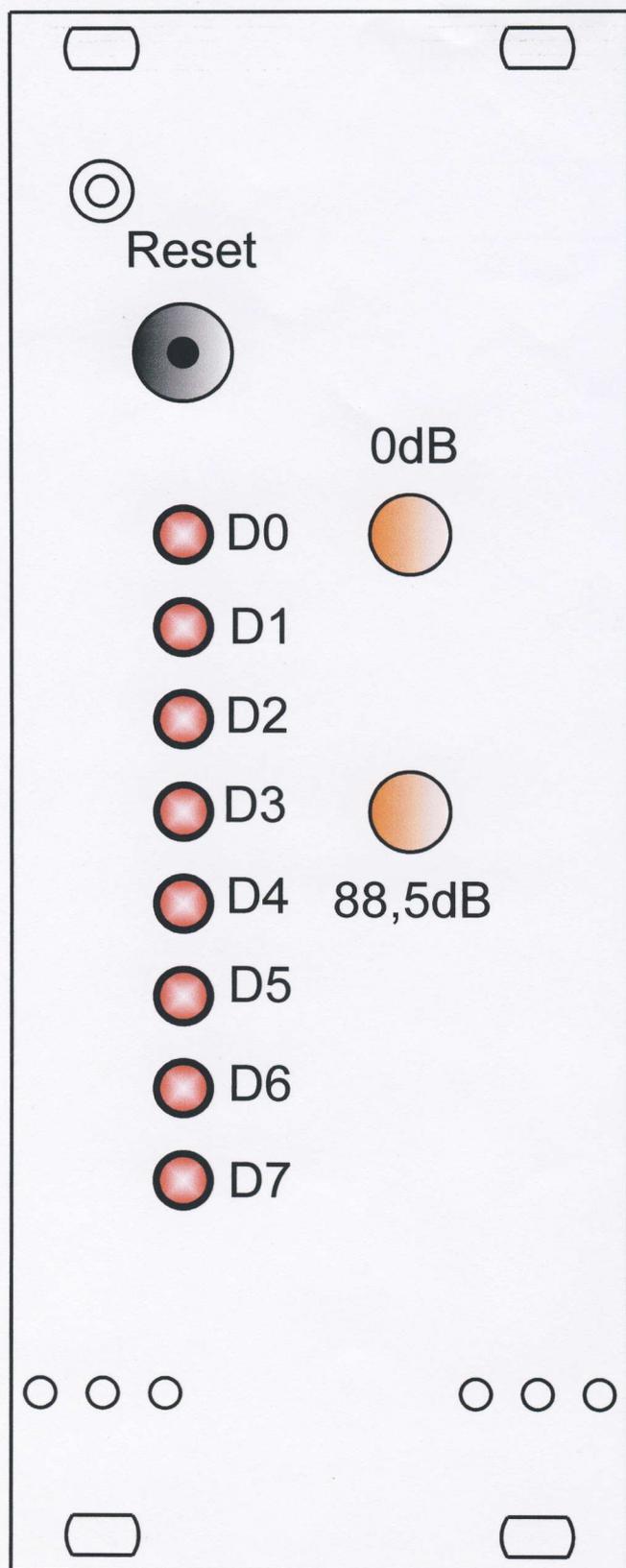
Document Number: 1

REV: 1

Date: 10.07.2007 08:37:12

Sheet: 1/1

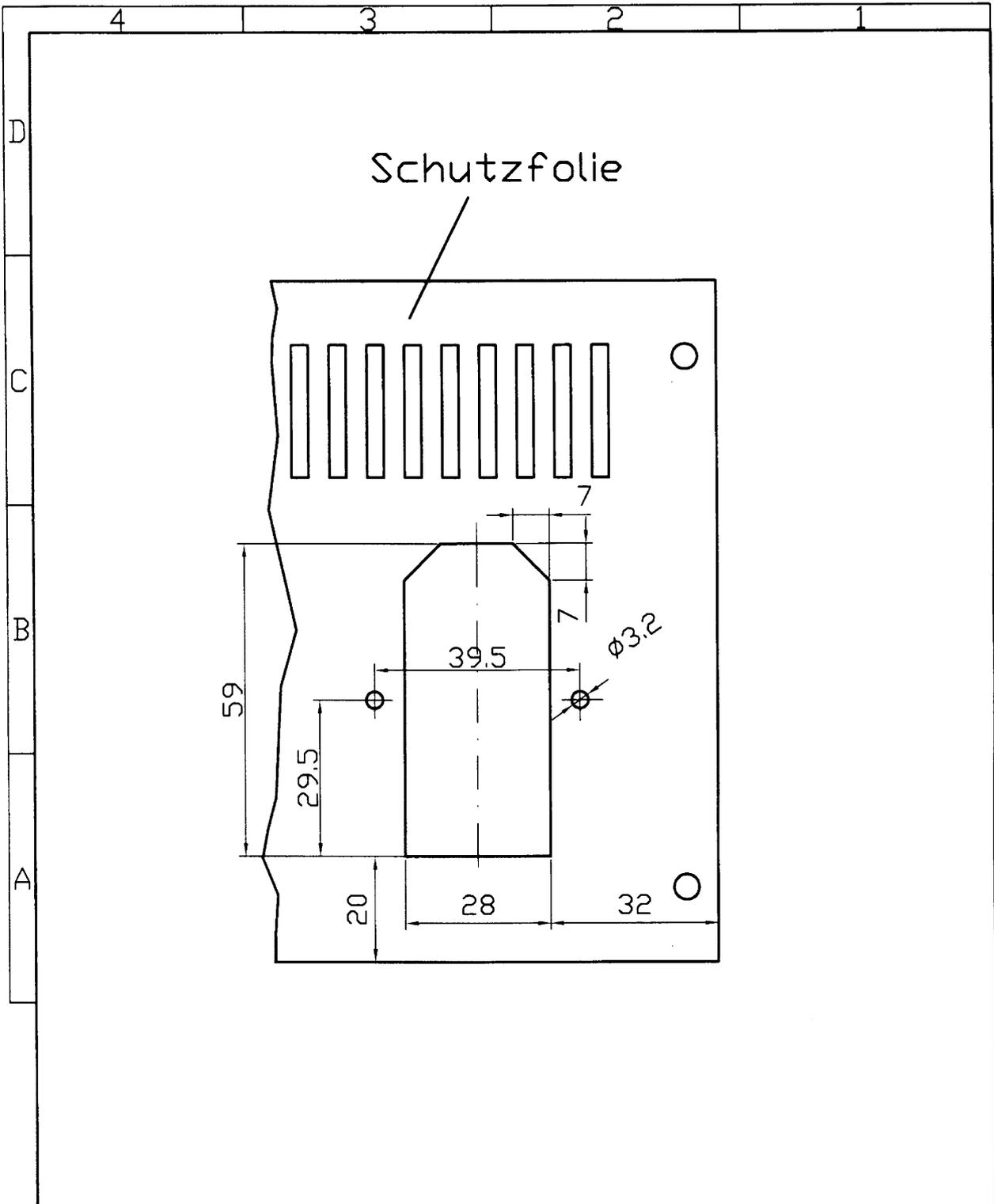
Bedienungsanleitung



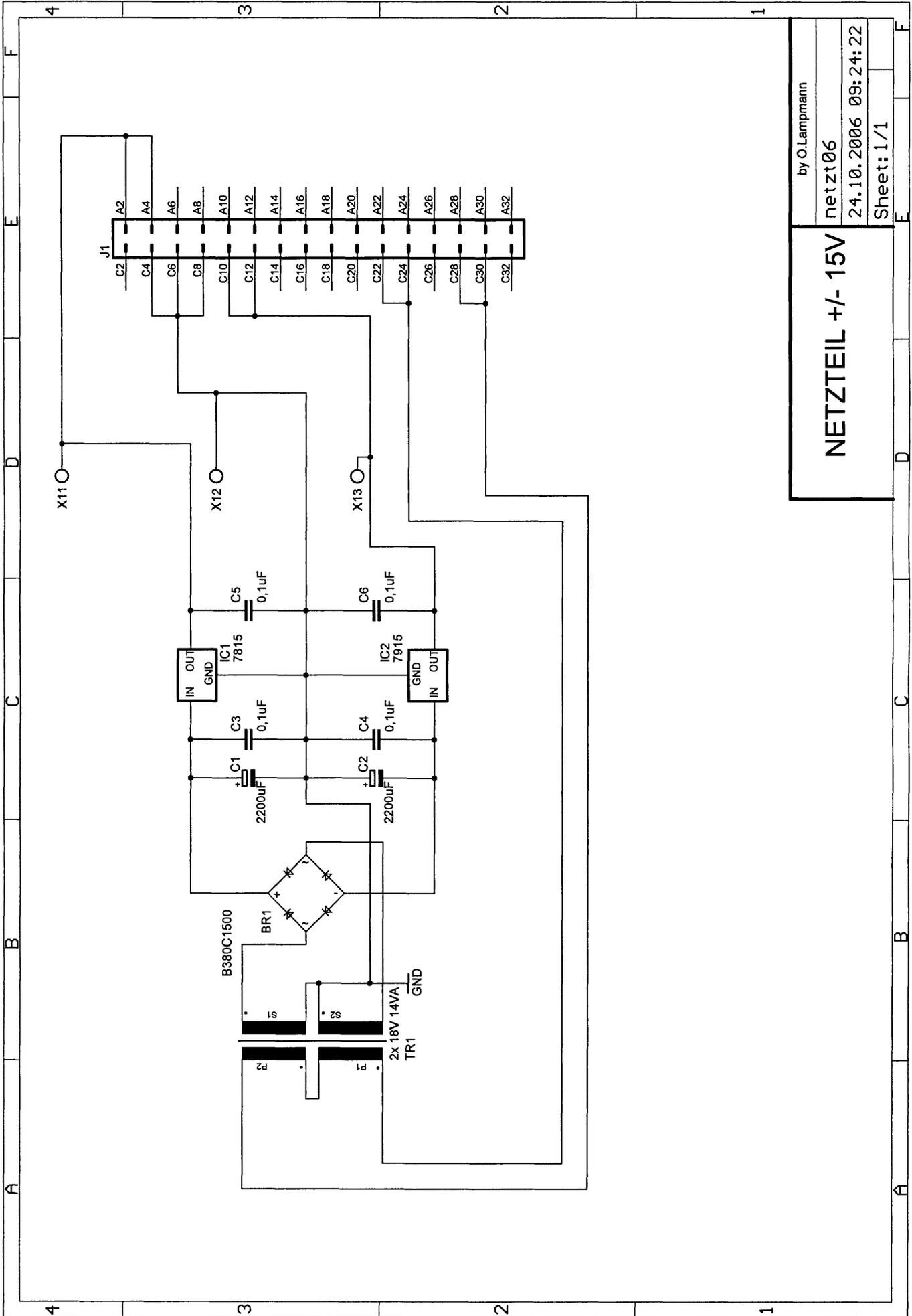
Mit der 0dB Taste wird die Lautstärke vom Ausgangssignal der beiden Audiokanäle in 0,375 dB Schritten angehoben.

Durch betätigen der 88,5 dB Taste wird der Wert der beiden Audiosignale in 0,375 dB Schritten verringert. Hat D4- D7 das Bitmuster 1111 erreicht wird das Ausgangssignal Stummgeschaltet, unabhängig von D0- D3.

Wenn die Resettaste gedrückt wird stellt diese den Lautstärkepegel vom Einschaltzustand wieder her.



		DIN 7168 fein mittel grob		Oberfläche	Maßstab 0	Position 1	Menge 1
					Werkstoff Alu		
			Datum	Name	Benennung Netzbuchse		
			Bearb. 20.06.07	Lampmann			
			Gepr.				
			Norm				
					Zeichnungs-Nr. 2	Blatt	
						Bl.	
Zust.	Änderung	Datum	Name	Dateiname rueckpl07.dwg			



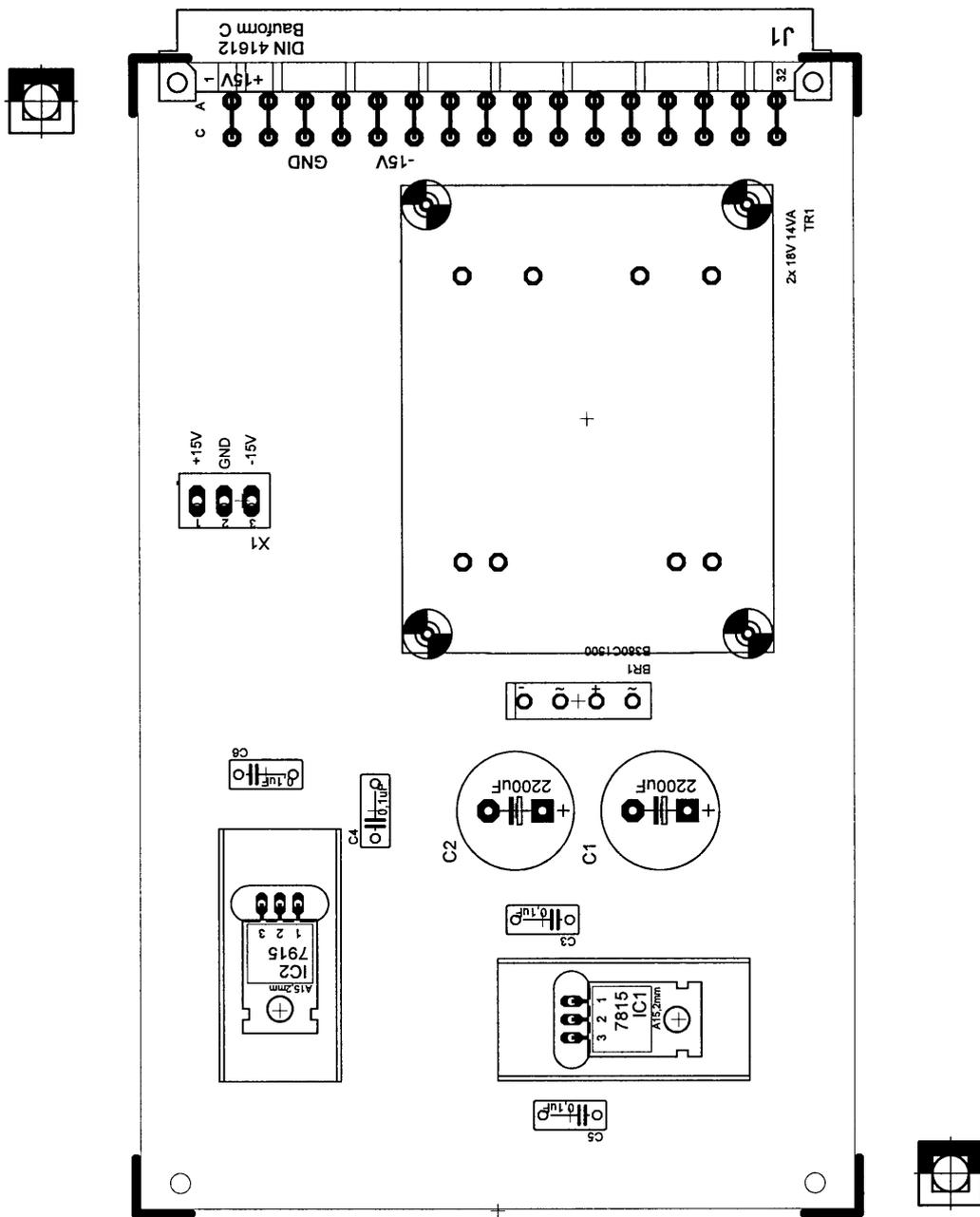
NETZTEIL +/- 15V

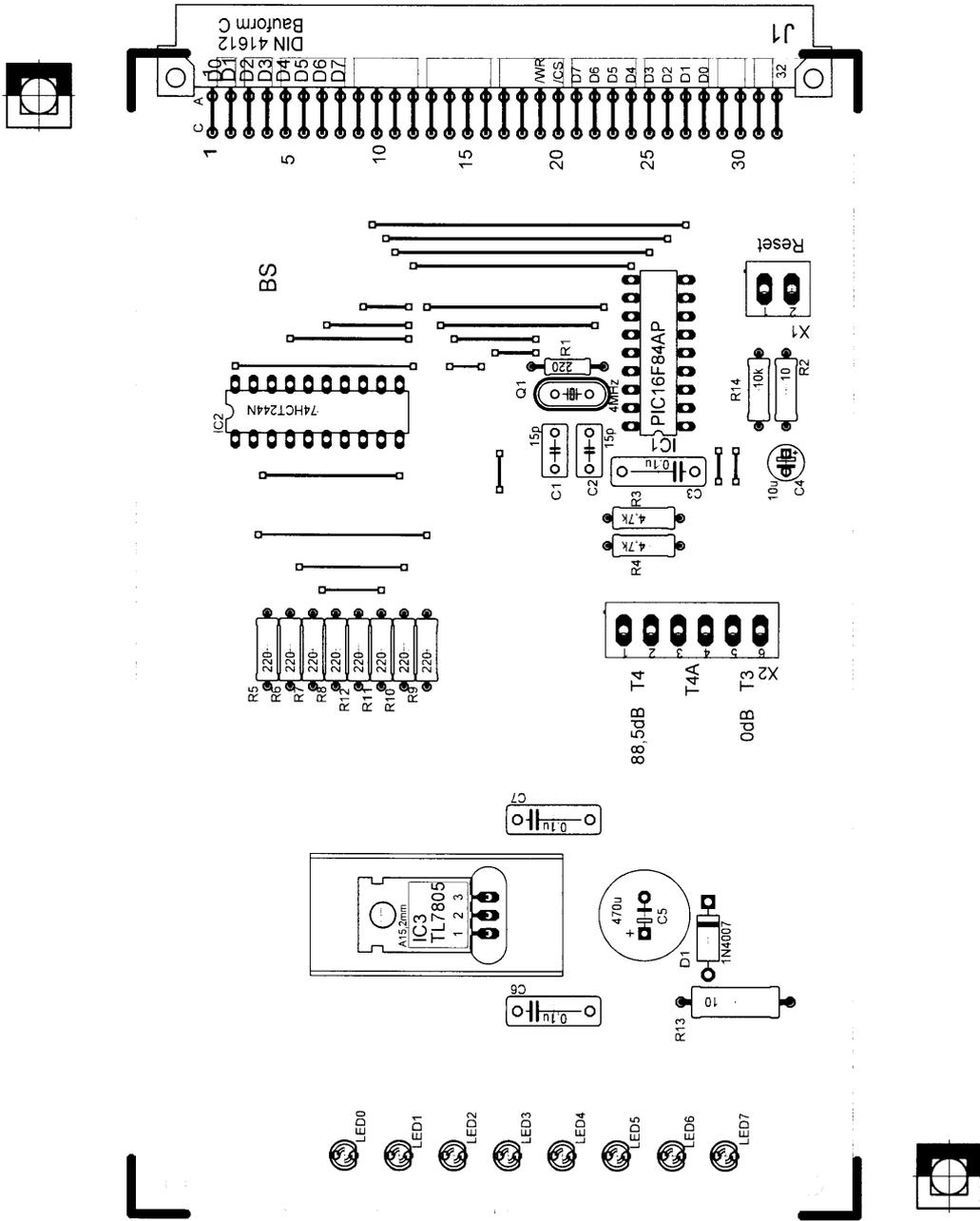
by O.Lampmann

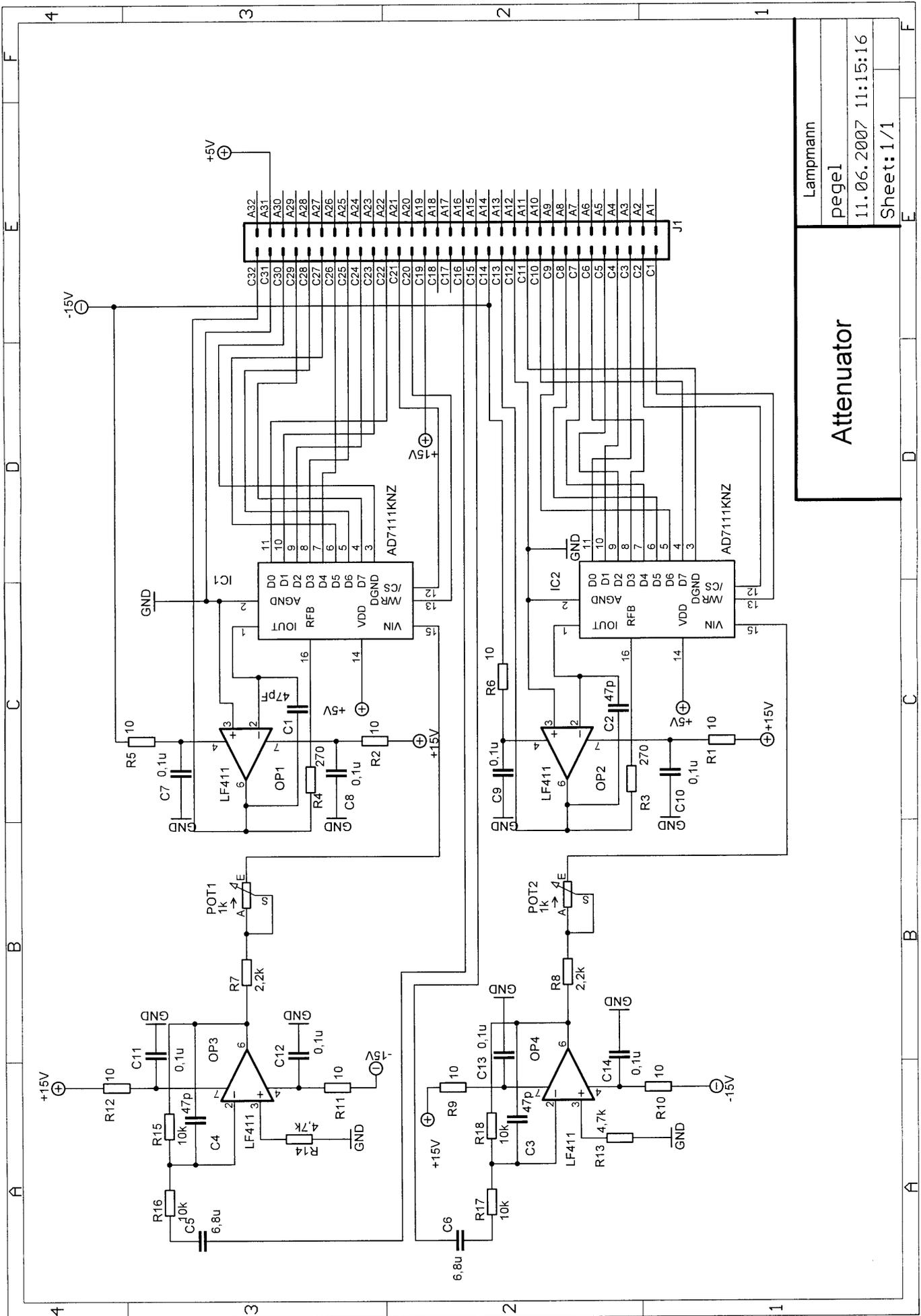
netz06

24.10.2006 09:24:22

Sheet: 1/1







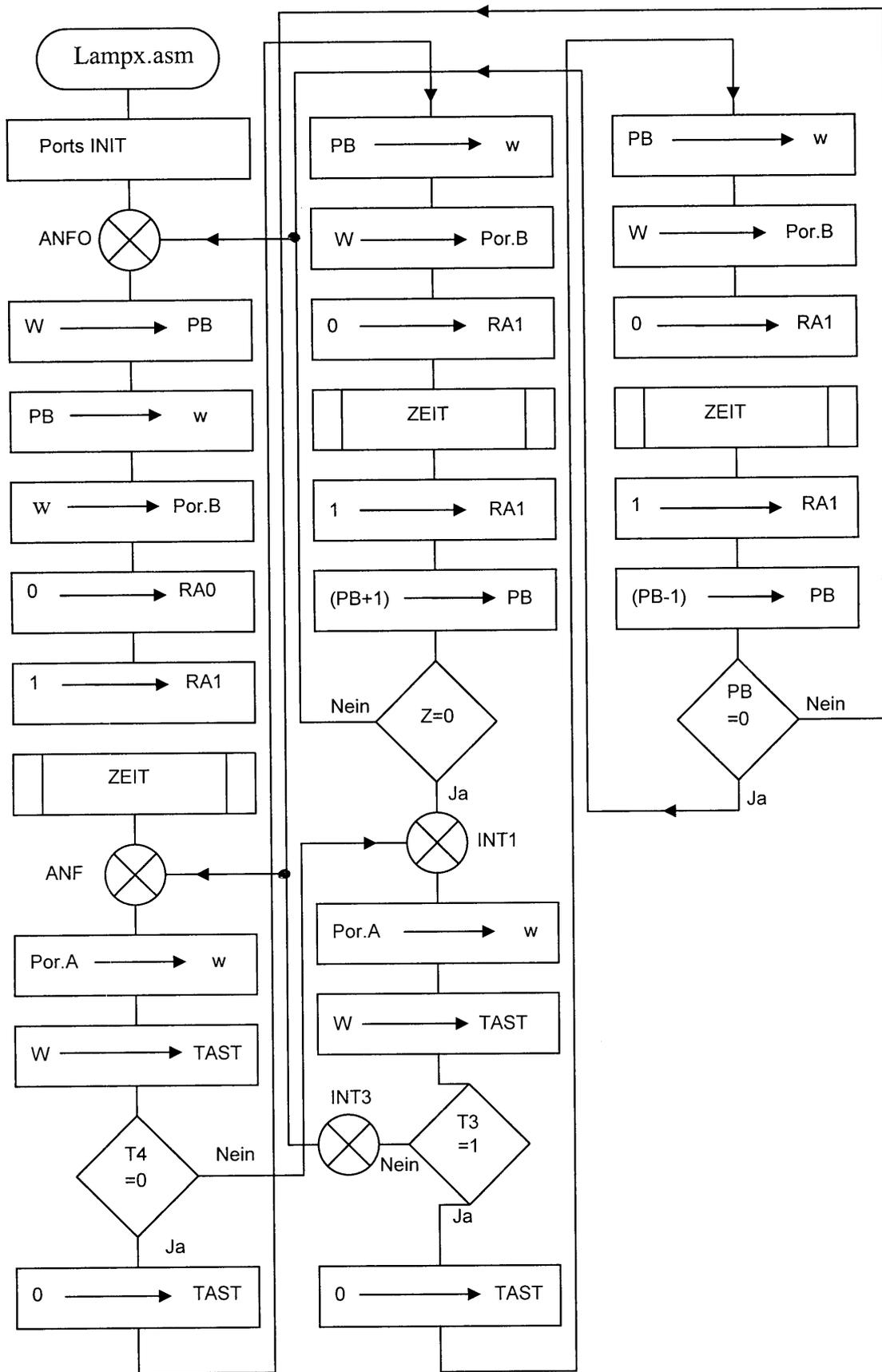
Attenuator

Lampmann

Pegel

11.06.2007 11:15:16

Sheet: 1/1



```
; *****
; * *Tastenabfrage Inkrement oder Dekrement Register PB am
; * Name lampx.ASM, 10.2.06 PORTB ausgeben fuer R-2R Netzwerk AD7111 *
; *****

#include "P16F84.INC" ;
__CONFIG __CP_OFF & __WDT_OFF & __PWRTE_ON & __XT_OSC

TAST EQU 0CH ;
PB EQU 0DH ; LED 0 - 7
K1 EQU D'128' ; Konstante K1 196
K2 EQU D'60' ; Konstante K2 150
K3 EQU D'1' ; Konstante K3 5
F1 EQU 0EH ; Zählvariable F1
F2 EQU 0FH ; Zählvariable F2
F3 EQU 10H ; Zählvariable F3

ORG 0
BSF STATUS,RP0 ; Auf Registerbank 1 umschalten
MOVLW 18H ; RA3 - RA4 Eingang der Rest Ausgang
MOVWF TRISA
MOVLW 0
MOVWF TRISB ; Port B als Ausgabeport schalten
BCF STATUS,RP0 ; Auf Registerbank 0 schalten
MOVLW 5AH ; LED0 - LED7 setzen Bitmuster im Einschaltzustand ins
; w-Register
ANF0 MOVWF PB ; Vom w-Register in Speicherzelle PB
MOVF PB,w
; *** MOVWF PORTB ***** ; Bitmuster für R-2R Netzwerk *****
; * BCF STATUS,RP0 ; Chipselect freigeben Register PB am
; * Name lampx.ASM, 10.2.06 WRE NICHT für kurze Zeit aktivieren * AD7111 *
; *** CALL ZEIT ***** ; Unterprogramm ZEIT aufrufen *****
ANF MOVF PORTA,w ; Taster einlesen und zum w-Register übergeben
MOVWF TAST ; w-Register in die Adresse 0CH mit dem Bezeichner TAST
__CONFIG __CP_OFF & __WDT_OFF & __PWRTE_ON & __XT_OSC
CALL ZEIT ; Unterprogrammaufruf Zeitschleife
TAST BSF PORTA,1 ; Bit des Registers PORTA (WRE NICHT) wieder sperren
PB EQU 0DH ; Erst nach Tasterbetätigung wieder freigeben
K1 BTFSC TAST,4 ; Übergehe nachfolgenden Befehl wenn Bit=0
K2 GOTO INT1' ; Rücksprung wenn TAST4 nicht Betätigt. (Am Taster liegt
K3 EQU D'1' ; in Ruhestellung eine 1)
F1 EQU 0EH ; Zählvariable F1
F2 EQU 0FH ; Zählvariable F2
F3 EQU 10H ; Zählvariable F3
MOVF PB,w ; PB nach W
BSF STATUS,RP0 ; Auf Registerbank 1 umschalten
MOVWF PORTB ; und nach PORTB eing der Rest Ausgang
BCFWF PORTA,1 ; Lösche Bit 1 des Registers PORTA
MOVLW 0
CALLF ZEITB ; Unterprogrammaufruf port schalten
BSF PORTA,1 ; Setze Bit 1 des Registers PORTA
MOVLW 5AH ; LED0 - LED7 setzen Bitmuster im Einschaltzustand ins
INCF PB,f ; Erhöhe den Wert aus der Speicherzelle PB um 1. Flags Z
ANF0 MOVWF PB ; Vom w-Register in Speicherzelle PB
MOVLW 0FFH ; Lade Konstante FF in das w-Register
; *** ANDWF PB,FF ***** ; UND Verknüpfung mit der Konstanten FF und PB
; * BCF STATUS,RP0 ; Umschalten Ausgabe Register PB am
; * Name lampx.ASM, 10.2.06 WRE NICHT für kurze Zeit aktivieren * AD7111 *
; *** CALL ZEIT ***** ; Unterprogramm ZEIT aufrufen *****
ANF GOTO ANF0,w ; Wenn es nicht gesetzt ist ignoriere den nächsten
MOVWF TAST ; Befehlster in die Adresse 0CH mit dem Bezeichner TAST
__CONFIG __CP_OFF & __WDT_OFF & __PWRTE_ON & __XT_OSC
CALL ZEIT ; sc - skip if clear. Mache beim übernächsten weiter
INT1 MOVF PORTA,w ; Taster einlesen und im w-Register speichern sperren
PB EQU 0DH ; Erst nach Tasterbetätigung wieder freigeben
K1 BTFSC TAST,4 ; Übergehe nachfolgenden Befehl wenn Bit=0
K2 GOTO INT1' ; Rücksprung wenn TAST4 nicht Betätigt. (Am Taster liegt
K3 EQU D'1' ; in Ruhestellung eine 1)
F1 EQU 0EH ; Zählvariable F1
```

```
MOVWF TAST ; w- Register in Fileregister TAST übertragen
BTFSS TAST, 3 ; Übergehe nachfolgenden Befehl wenn Bit=1 und mache
; beim übernächsten weiter.
; TAST3 in Ruhestellung eine 0 es wird
; mit dem nächsten Befehl Sprung zu ANF weitergemacht
; (Abfrage ist TAST4 Betätigt?)
GOTO INT3 ;
CLRF TAST ; Übernächster Befehl: Registerinhalt von TAST löschen
MOVF PB, w ; Registerinhalt nach w- Kopieren
MOVWF PORTB ; An PORTB ausgeben
BCF PORTA, 1 ; Bit 1 des Registers PORTA auf 0 setzen
CALL ZEIT ; Unterprogrammaufruf ZEIT
BSF PORTA, 1 ; Bit 1 der Speicherzelle PORTA setzen
; Wird kein Taster betätigt WRE NICHT PB übernahme
; gesperrt Störungsunabhängiger
DECFSZ PB, f ; verringere den Wert aus der Speicherzelle f um 1.
; Falls das 0 ergibt, dann ignoriere den nachfolgenden
; Befehl. Es wird mit dem übernächsten weitergemacht.
INT3 GOTO ANF ; Sprungbefehl zur Marke ANF
MOVWF TAST ; w- Register in Fileregister TAST übertragen
GOTO ANF0 ; wieder zum Programmstart
BTFSS TAST, 3 ; Übergehe nachfolgenden Befehl wenn Bit=1 und mache
ZEIT MOVLW K1 ; F1 mit K1 laden
MOVWF F1 ; TAST3 in Ruhestellung eine 0 es wird
M1 MOVLW K2 ; F2 mit K2 laden ; mit dem nächsten Befehl Sprung zu ANF weitergemacht
MOVWF F2 ; Abfrage ist TAST4 betätigt?
M2 MOVLW K3 ; F3 mit K3 laden
MOVWF F3 ;
M3 DECFSZ TAST, F3, f ; F3 - 1, Sprung, wenn F3 = 0 ; von TAST löschen
GOTO M3 ;
DECFSZ PB, w, F2, f ; F2 - 1, Sprung, wenn F2 = 0
GOTO M2 ;
DECFSZ PORTB, F1, f ; F1 - 1, Sprung, wenn F1 = 0
GOTO M1 ; RTA, 1 ; Bit 1 des Registers PORTA auf 0 setzen
RETURN ; Unterprogrammaufruf KEIN
BSF PORTA, 1 ; Bit 1 der Speicherzelle PORTA setzen
END ; Programmende
; gesperrt Störungsunabhängiger
```

```
DECFSZ PB, f ; verringere den Wert aus der Speicherzelle f um 1.
; Falls das 0 ergibt, dann ignoriere den nachfolgenden
; Befehl. Es wird mit dem übernächsten weitergemacht.
INT3 GOTO ANF ; Sprungbefehl zur Marke ANF
MOVWF TAST ; w- Register in Fileregister TAST übertragen
GOTO ANF0 ; wieder zum Programmstart
BTFSS TAST, 3 ; Übergehe nachfolgenden Befehl wenn Bit=1 und mache
ZEIT MOVLW K1 ; F1 mit K1 laden
MOVWF F1 ; TAST3 in Ruhestellung eine 0 es wird
M1 MOVLW K2 ; F2 mit K2 laden ; mit dem nächsten Befehl Sprung zu ANF weitergemacht
MOVWF F2 ; Abfrage ist TAST4 betätigt?
M2 MOVLW K3 ; F3 mit K3 laden
MOVWF F3 ;
M3 DECFSZ TAST, F3, f ; F3 - 1, Sprung, wenn F3 = 0 ; von TAST löschen
GOTO M3 ;
DECFSZ PB, w, F2, f ; F2 - 1, Sprung, wenn F2 = 0
GOTO M2 ;
GOTO M1 ;
```


LITERATURNACHWEIS:

Analog Devices, Datenbuch Linear Products 1990/91

Dieter Kohtz, Messen, Steuern und Regeln mit PIC- Mikrokontroller, Franzis V.

König, A, Das PICmicro Profi- Buch, Franzis Verlag

E. Heilmayer, AD- DA Wandler, Markt und Technik Verlag

K. Block, H.- J. Hölzel, L. Wölfing, P, Zachert, Kontaktlose Signalverarb. H. Stamm V.

Tietze Schenk, Halbleiterschaltungstechnik, Springer Verlag

Referenz- Handbuch, EAGLE 4.0, CadSoft Verlag

Rudolf Burkhardt, Mikrokontroller PIC16F84, Kursskriptum, VHS- Kurs Nr. 9692