

Konzeption, Entwicklung und Aufbau einer FADC-basierten Ausleseelektronik für das MAGIC-Teleskop

vom Fachbereich Elektrotechnik und Informatik
der Universität -Gesamthochschule Siegen
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
(Dr.Ing.)

genehmigte Dissertation

von

Diplom-Ingenieur Ralf Stiehler

1. Gutachter : Prof. Dr. Schwarte
2. Gutachter : Prof. Dr. Pavel

Tag der mündlichen Prüfung : 3. Dezember 2001

Inhalt

I. Einleitung

I.1. Einführung.....	1
I.2. Das MAGIC-Cherenkov-Teleskop.....	2
I.2.1. Der Cherenkov-Effekt.....	2
I.2.2. Das Detektorprinzip.....	3
I.2.3. Übersicht der Signalverarbeitungskette.....	5

II. Grundlagen

II.1. Grundlagen schneller Analog-Digital-Wandler.....	7
II.1.1. Funktionsweise von FADCs.....	7
II.1.2. Grundlegende Begriffe der Signalverarbeitung.....	8
II.2. Grundlagen schneller Elektronik.....	13
II.2.1. Theorie von Übertragungsleitungen schneller Signale.....	13
II.2.2. Übertragung differentieller und bipolarer Signale.....	19

III. Aufbau eines FADC-Systems

III.1. Gesamtübersicht.....	23
III.2. Das FADC-Modul.....	32
III.3. Das FADC-Motherboard.....	35
III.3. Die Signalverteilung im System.....	44
III.4. Die zentrale Clockerzeugung und Triggersynchronisation, Verteilung von Clock und Trigger	48
III.5. Das Delayboard zur Generierung von boardspez. Look-Up-Tabellen.....	52
III.6. Die Auslese eines einzelnen Boards im Testaufbau.....	55
III.7. Die Auslese des Gesamtsystems.....	57

IV. Messungen am FADC-System

IV.1. Testmessungen am einzelnen FADC-Modul.....	60
IV.2. Testmessungen an der Signalverteilung.....	67
IV.3. Erfassung des Jitters und Offsets an einem 8-Kanal-FADC-Motherboard.....	71
IV.4. Erfassung des Jitters an einem 32-Kanal-Prototypen.....	75

V. Zusammenfassung

Zusammenfassung.....	77
----------------------	----

Quellennachweis	78
Kurzfassung in englischer Sprache (Abstract).....	79
Kurzfassung in deutscher Sprache	80
Anhang	
Anhang A : Einführung in den FPGA-Logikentwurf.....	A1
Anhang B : Schematics und Abel-Dateien des FPGA-Logikentwurfs.....	B1
Anhang C : Schaltpläne der einzelnen Platinen.....	C1
Anhang D : C-Quell-Datei zur Auslese mit PCI-I/O-Karte.....	D1

Danksagung

I.1. Einführung

Im Rahmen dieser Arbeit wurde ein Konzept zur Auslese der in der Fokalebene eines Cherenkov-Teleskops stehenden Kamera erstellt und umgesetzt. Bildgebende atmosphärische Cherenkov-Teleskope verwenden große optische Reflektoren als Lichtkollektor und schnelle Photomultiplier in der Kamera zur Aufnahme von Cherenkov-Licht, das in der Erdatmosphäre von kosmischen hochenergetischen Teilchen bei sog. Gammabursts erzeugt wird.

Die Realisierung dieses neuartigen Systems mit 640 Kanälen basiert auf 300 MHz Analog-Digital-Wandlern (FADCs) zur Datenaufnahme, welche auf insgesamt 4 Schaltschränke (Racks) verteilt sind. Die Auslese erfolgt über ein eigens zu diesem Zweck entwickeltes System von Backplanes, Interfaceboards und Optokopplern über eine PCI-Schnittstelle in einen PC.

Die Herausforderung bei dieser Arbeit ist weniger die Entwicklung von schnellen Analog-Digital-Wandlern (welche ohnehin als industrielle Halbleiterchips für immer höhere Abtastfrequenzen verfügbar sind), vielmehr geht es um die Realisierung eines größeren Systems, in dem vielschichtige Probleme zu lösen sind. Die eigentliche Innovation ist als die Gesamtheit der Lösungen zu verstehen, wobei insbesondere der Verteilung und Verarbeitung eines relativ großen Datenflusses (bis 80 Mbyte/s) vom Frontend bis hin zum PC und der Verteilung einer hochfrequenten Systemclock mit einer Genauigkeit bis in den Picosekundenbereich hinein besondere Aufmerksamkeit gewidmet wurde.

Um die Rahmenbedingungen für die Ausleseelektronik besser verstehen zu können, soll im weiteren Verlauf dieser Einleitung näher auf die grundlegende Funktionsweise des „MAGIC-Teleskops“ eingegangen werden. Im zweiten Kapitel dieser Arbeit findet sich eine Zusammenstellung wichtiger elektrotechnischer Grundlagen, die unabdingbar zum Verständnis des Aufbaus des Gesamtsystems sind, welches in Kapitel 3 dieser Arbeit dargestellt und erläutert wird. Die Leistungsfähigkeit des Systems wird anhand von Testmessungen demonstriert, die im vierten Kapitel näher beschrieben werden.

I.2. Das MAGIC-Cherenkov-Teleskop

I.2.1. Der Cherenkov-Effekt

Die Bezeichnung des Experiments „MAGIC“ steht für **M**ajor **A**tmosphere **G**amma **I**maging **C**herenkov-Telescope. Der der Lichterzeugung zugrunde liegende Cherenkov-Effekt soll im folgenden kurz erläutert werden:

Bewegte Ladungsträger (z.B. Elektronen) erzeugen entlang deren Flugbahn ein Magnetfeld, an dem sich benachbarte Atome ausrichten, d.h. polarisiert werden. Es bildet sich ein Dipolmoment aus. Die zur Polarisation nötige elektromagnetische Strahlung kann sich maximal mit Lichtgeschwindigkeit in diesem Medium ausbreiten, d.h. solange die Geschwindigkeit der Elektronen klein im Verhältnis zur Lichtgeschwindigkeit in diesem Medium ist, erfolgt eine symmetrische Anordnung der polarisierten Atome und das resultierende Gesamtdipolmoment ist Null.

Ist die Geschwindigkeit der bewegten Ladung allerdings höher als die Lichtgeschwindigkeit in diesem Medium, dann ist keine symmetrische Anordnung der Dipole mehr gegeben. Das Gesamtdipolmoment ist größer als Null, es wird Energie in Form von Licht abgestrahlt.

Folgende Abb. soll dies erläutern :

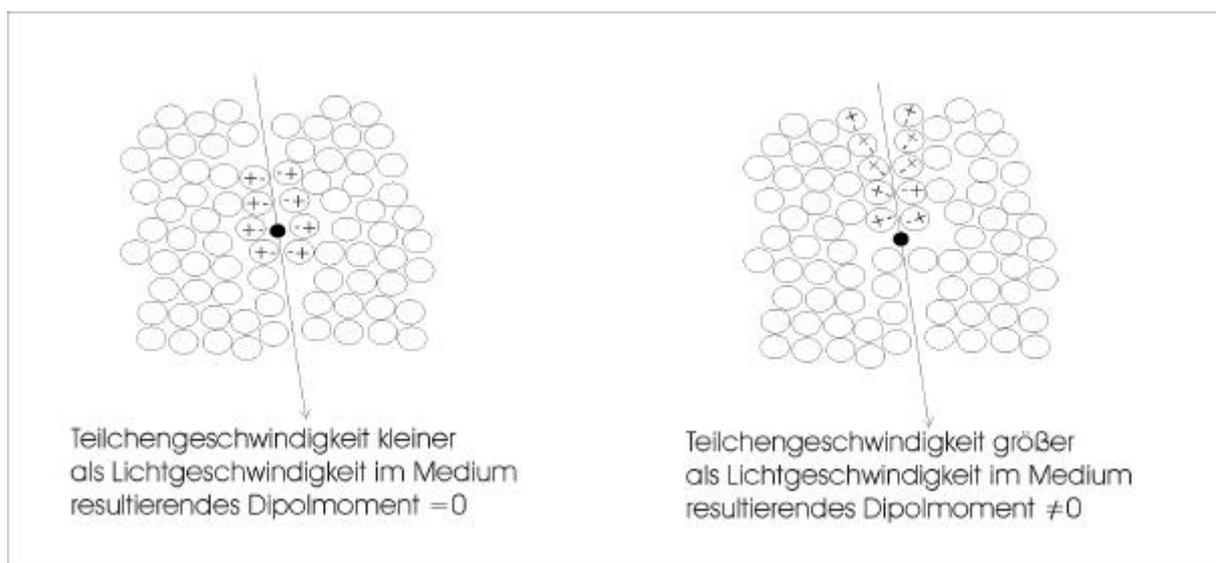


Abb.I.1. : Cherenkov-Effekt [GRUPEN]

I.2.2. Das Detektorprinzip

Da eine direkte Detektierung von Gammastrahlung nicht möglich ist, muss zunächst eine Konvertierung dieser Teilchen in direkt oder indirekt nachweisbare Elektronen erfolgen. Diese Konvertierung kann durch die physikalischen Effekte Paarbildung, Comptoneffekt oder Photoeffekt erfolgen.

Bei bildgebenden atmosphärischen Cherenkov-Teleskopen nutzt man aus, dass energiereiche kosmische Gammateilchen im oberen Teil der Erdatmosphäre Elektron-Positron-Paare erzeugen, welche ihrerseits wiederum durch Comptonstreuung und Bremsstrahlung hochenergetische Photonen erzeugen. Diese erzeugen wiederum über Paarbildung neue Photonen. So bildet sich eine in Richtung der Erdoberfläche bewegendes Lawine aus Photonen und Elektronen, wobei die Elektronen mit abnehmender Höhe an Energie verlieren. [MAGIC], [LOCHNER]

Die durch Paarbildung und Comptonstreuung erzeugten Elektronen sind sehr energiereich und führen aufgrund einer Polarisation von Gasatomen zur Emission von Cherenkov-Licht. Ein solcher „Lichtpool“, der auch als Cherenkov-Licht-Airshower bezeichnet wird, hat eine Breite von bis zu zweihundert Metern, ist aber nur wenige Meter „dick“.

In folgendem Bild sei das Detektorprinzip erläutert :

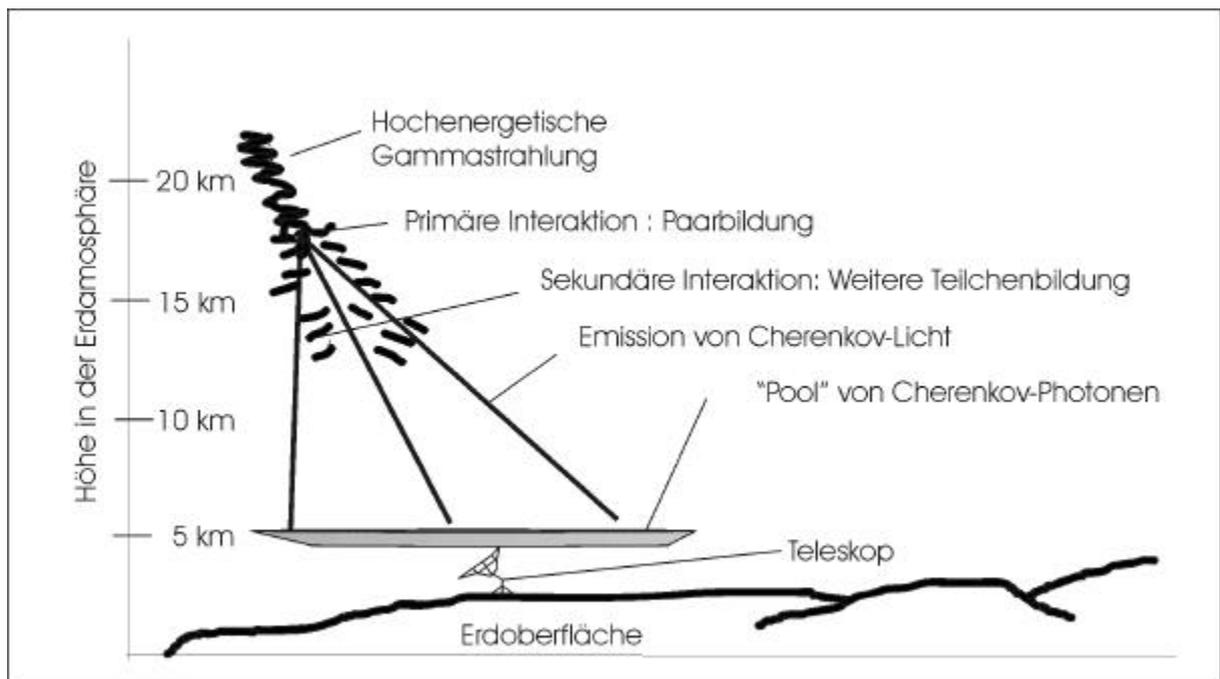


Abb. I.2. : Showerbildung [LOCHNER]

Das Licht wird in einem großen parabolischen Spiegel (Durchmesser ca. 17m) gesammelt und auf eine in der Fokalebene positionierte Kamera projiziert. Die Kamera besteht aus 576 schnellen Photomultipliern, die an sog. Winston-Cones angebunden werden, welche das Licht über eine hexagonale Öffnung zu den runden Eintrittsfenstern der Photomultiplier leiten. Auf diese Weise wird eine nahezu 100%ige Abdeckung der Kameraoberfläche sowie eine hohe Lichtsammeleffizienz erreicht. Jeder dieser 576 Photomultiplier wird an eine Ausleseelektronik angeschlossen, die in dieser Arbeit näher beschrieben wird.

Das durch den Cherenkov-Effekt erzeugte Licht ist nur sehr schwach und hebt sich in der Intensität praktisch nicht vom Hintergrund ab. Da die „Dicke“ des Airshowers aber zu schnellen Pulsen von wenigen Nanosekunden Länge führt, kann man das vom Cherenkov-Licht herrührende Signal vom überlagerten (poissonverteilten) Untergrund filtern.

Die Photomultiplier mit wabenförmigen Winston-Cone-Öffnungen sind in einem Kreis angeordnet. Licht, das in einem Cherenkov-Lichtshower erzeugt wurde und auf den Teleskopspiegel fällt, wird auf die Kamera projiziert, so dass in einer bestimmten Anzahl von Photomultipliern ein zeitlich kurzes Lichtsignal registriert wird. Der in diesem Experiment interessierende Cherenkov-Licht-Shower wird durch die Form und Ausdehnung des Gebiets, in dem die Photomultiplier Licht registriert haben, charakterisiert. Über die schnelle Erkennung von charakteristischen Mustern kann ein solcher Air-Shower angezeigt und ein entsprechendes digitales Triggersignal zur Abspeicherung des Kamerasignals erzeugt werden.

Die folgende Abbildung zeigt die Anordnung der Photomultiplier in der Kameraebene und ein typisches Ereignis eines Cherenkov-Licht-Showers :

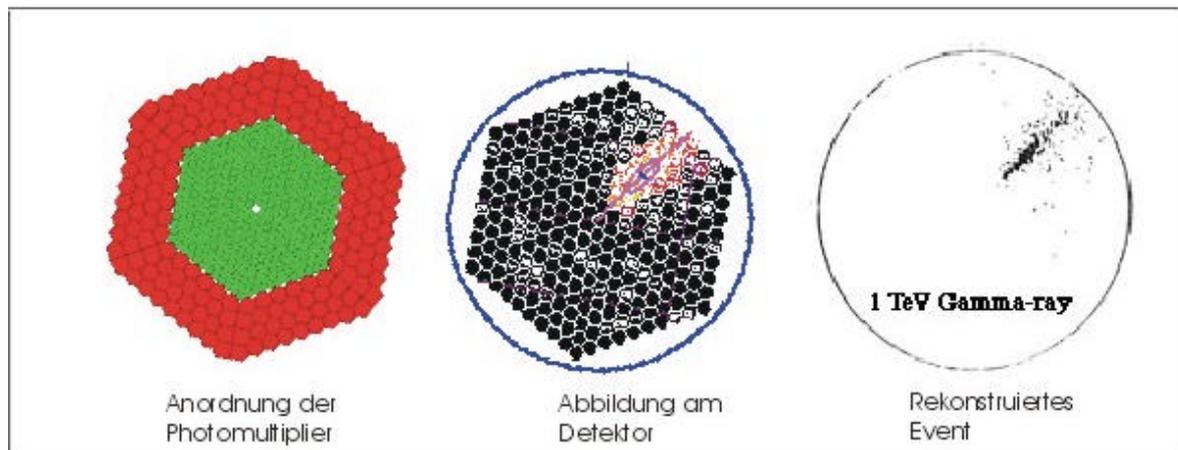


Abb. I.3. : Anordnung der Photomultiplier und typisches Ereignis [LOCHNER, MAGIC1]

I.2.3. Übersicht der Signalverarbeitungskette

Beim MAGIC-Experiment kommen klassische sechstufige Photomultiplier-Röhren (PMTs) zum Einsatz, die bei einfallendem Cherenkov-Licht am Ausgang Pulse von rund 2 ns Breite (FWHM) bereitstellen. Da Pulse einer solchen kurzen Dauer mit dem vorhandenen 300 MHz-System nur ungenügend genau rekonstruiert werden können, muss das Signal für die FADCs in einem Pulsformer künstlich verlängert werden.

Da die Cherenkov-Lichtsignale sich vor allem durch die kurze Dauer vom Untergrund abheben, kann man die Signale nicht beliebig verlängern, da sonst die Auflösungsdynamik zu schlecht würde. In der Praxis haben sich Pulsformer (Shaper / Stretcher) bewährt, die das Signal der Photomultiplier auf rund 5-7 ns Breite (FWHM) bringen.

Die gesamte Signalverarbeitungskette eines einzelnen Kanals bis zum Eingang der FADCs ist in folgender Skizze dargestellt :

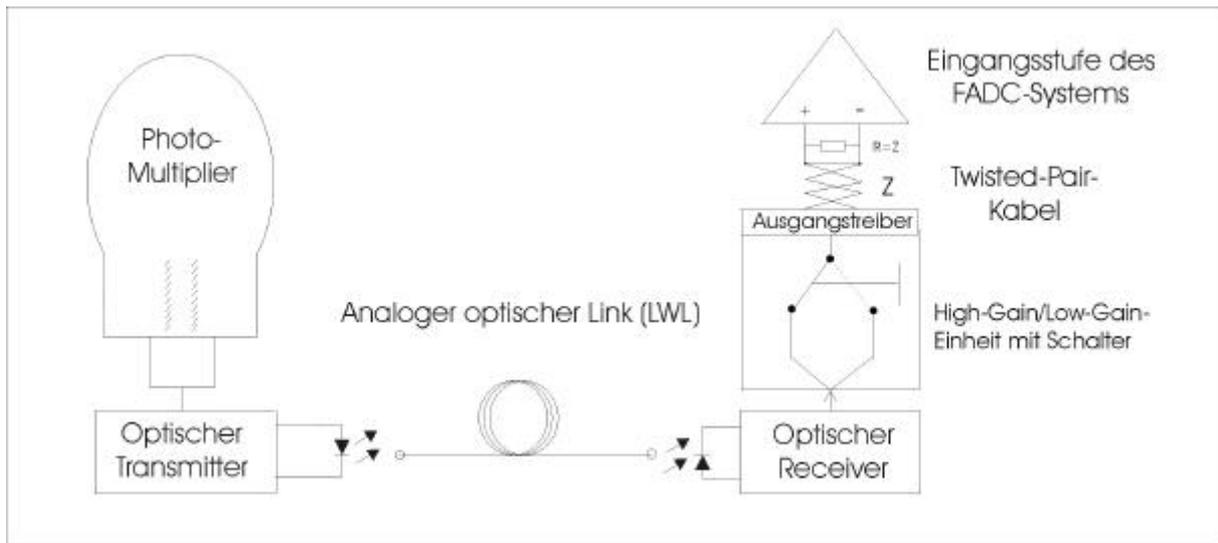


Abb. : Aufbau der Signalverarbeitungskette für einen von insgesamt 576 Kanälen

Nach einer ersten Vorverstärkerstufe wird das Photomultiplier-Signal über einen analogen optischen Link zur Signalaufnahmestation geleitet und dort auf zwei Signalzweige aufgeteilt. Die Einführung von zwei Signalzweigen mit unterschiedlicher Verstärkung führt zu einer höheren dynamischen Auflösung des Systems. Durch Einsatz eines GaAs-Schalters kann man ein Signal verzögern und beide Signale auf denselben FADC-Kanal leiten. Im ersten Signalzweig („High-Gain“) wird das Signal verstärkt und zur Triggerelektronik und den FADCs weitergeleitet. Der zweite Signalzweig („Low Gain“) unterwirft das Signal lediglich einer Verzögerung von ca. 50 ns und leitet es ohne weitere Verstärkung an die FADCs weiter. Eine tiefere Abhandlung dieser Thematik und erste Messergebnisse bei Tests mit Photomultipliern und einem 32-Kanal-Prototypensystem finden sich bei [MAGIC1], [MAGIC2].

II. Grundlagen

II.1. Grundlagen schneller Analog-Digital-Wandler (FADCs)

II.1.1.Funktionsweise von FADCs

Ein Analog-Digitalwandler (ADC) dient zum Umwandeln von analogen Signalen (z.B. Spannungsverlauf) in eine äquivalente digitale Form.

Ein ADC besteht in seiner einfachsten Form aus einem als Komparator geschalteten Differenzverstärker, von dem ein Eingang auf einer festen Spannung (Referenzspannung) liegt und dessen zweiter Eingang mit dem zu digitalisierendem Signal gespeist wird. Der Ausgang des Differenzverstärkers wird bei einer Signalspannung, die kleiner als die Referenzspannung ist, auf Nullniveau (L=logisch 0) verharren und bei einer Signalspannung, die größer ist als die Referenzspannung, in die Sättigung (H=logisch 1) gehen.

Ein so beschalteter ADC kann natürlich nur eine Auflösung von weniger als einem Bit erzielen, für eine bessere Auflösung ist es erforderlich, das Eingangssignal über sukzessiv-approximative Verfahren demselben oder aber über einen entsprechenden Spannungsteiler (Widerstandsleiter oder reference ladder) mehreren Komparatoren gleichzeitig zuzuführen. Letztere Methode ist ab einer gewissen Konversionsgeschwindigkeit unabdingbar und wird als parallele Umsetzung bezeichnet. Nach diesem Prinzip arbeitende Analog-Digitalwandler werden als parallele oder F-ADCs (Flash-ADCs) bezeichnet.

II.1.2. Grundlegende Begriffe der Signalverarbeitung

In den folgenden Abschnitten sollen grundlegende Begriffe der Signalverarbeitung kurz erläutert werden, die in engem Zusammenhang mit dem Auflösungsverhalten von ADCs stehen und in dieser Arbeit verwendet werden. Auf eine eingehende Herleitung der Zusammenhänge wurde verzichtet, diesbezüglich sei auf die gängigen Lehrbücher der Nachrichtentechnik und Signalverarbeitung (z.B. [SIGNAL1], [SIGNAL2]) verwiesen.

Das Abtasttheorem von Nyquist und Shannon

Das Abtasttheorem besagt, dass die minimale Abtastfrequenz mindestens doppelt so groß sein muss wie die höchste im Signal enthaltene Frequenzkomponente, damit ein Signal ohne Verlust an Informationen aus den digitalen Daten rekonstruiert werden kann.

Aliasing

Unter Aliasing oder auch Bandüberlappungseffekt versteht man eine (unerwünschte) Spiegelung höherer Frequenzen in das Frequenzspektrum des Eingangssignals. Dieser Effekt tritt prinzipiell bei jedem Verfahren auf, bei dem gleichmäßig Stichproben aus einem kontinuierlichem Vorgang entnommen werden, auch bei einem idealen ADC mit einer Umsetzzeit von Null. Geeignete Gegenmaßnahmen wären die Vorschaltung eines Anti-Aliasingfilters (Tiefpass) oder die Erhöhung der Abtastfrequenz (vgl. Shannon/Nyquist).

Quantisierungsrauschen

Bei der Analog-Digitalumsetzung wird der Eingangsspannungsbereich des Wandlers in diskrete Stufen unterteilt, bei einer Auflösung von n Bit ergeben sich somit 2^n Quantisierungsintervalle. Das bedeutet, dass sich der Ausgangscode des Wandlers erst dann ändern kann, wenn sich die Eingangsspannung um mindestens den Betrag eines halben minimalen Quantisierungsintervalls geändert hat. Man kann diesen Quantisierungsfehler als zufällige Größe auffassen und erhält somit nach den Regeln der Wahrscheinlichkeitslehre als zweites statistisches Moment einer Boxfunktion den Quantisierungsfehler zu

$$QF=1/12^{1/2}$$

Apertur

Bei den bisherigen Beschreibungen wurde zur besseren Anschauung davon ausgegangen, dass die Abtastpulse idealen Nadelpulsen (Dirac-Stoß) entsprechen. In der Realität besitzen aber alle Analog-Digital-Konverter eine endliche Abtastbreite, die als Öffnungszeit oder Apertur bezeichnet wird. Der Fehler, der sich aufgrund der endlichen Dauer der Einzelabtastung einstellt, wird als Aperturfehler bezeichnet.

Aperturjitter

Da man in einem realen System nicht einen idealen Taktimpulsgenerator für die Abtastpulse annehmen kann, muss man davon ausgehen, dass Varianzen der Abtastfrequenz auch zu Fehlern führen, die man als Aperturjitter bezeichnet. Hierzu sei angemerkt, dass in vielen Lehrbüchern bei der Herleitung der Systemübertragungsfunktion von ADCs von einer Mittelwertbildung während der Öffnungszeit ausgegangen wird. Da aber viele Analog-Digitalwandler systembedingt den Spitzenwert der Abtastung erfassen, erhält man schon oft auf diese Weise einen "Jitter".

Rauschen

Die Ursachen des Rauschens liegen in der Quantennatur elektrischer Vorgänge begründet, da Ströme und Spannungen das Resultat einzelner, stochastisch sich bewegendener Ladungsquanten (Elektronen) sind. Diese Ladungsbewegungen lassen sich mit Hilfe der Gesetze der statistischen Physik beschreiben.

Ein wichtiger Parameter ist die spektrale Rauschleistungsdichte $\rho = \Delta P / \Delta f$, welche als Rauschleistung pro Frequenzintervall definiert wird. Daraus ableiten lassen sich zwei weitere gebräuchliche Begriffe, die spektrale Rauschspannungsdichte ρ_u [$V \cdot \text{Hz}^{1/2}$] und Rauschstromdichte ρ_i [$A \cdot \text{Hz}^{1/2}$]

Die verschiedenen Arten des Rauschens lassen sich nach den physikalischen Ursachen, der Spektralverteilung oder der Amplitudenverteilung klassifizieren. Eine detaillierte Abhandlung über diese Thematik findet sich in [RAUSCHEN]. In der folgenden Tabelle soll ein Überblick gegeben werden:

Art des Rauschens	Vorkommen	Physikalische Ursachen	Spektralverteilung Signalgröße	Amplitudenverteilung
Thermisches Rauschen (Johnson/Nyquist-Noise)	Ohmsche Widerstände und PN-Übergänge im Halbleiter	Temperaturabhängige thermische Bewegung der Ladungsträger	weißes Rauschen $U_r = (4 \cdot k_B \cdot T \cdot R \cdot B)^{1/2}$	Gauß
Schrotrauschen, Generations- oder Rekombinationsrauschen (Schottkyrauschen)	Röhren und PN-Übergänge im Halbleiter	Stromabhängiges Rauschen aufgrund Bewegung diskreter Ladungsträger	weißes Rauschen $I_r = (2 \cdot e \cdot I \cdot B)^{1/2}$	Gauß
1/f – Rauschen oder Funkelrauschen	Kohlewiderstände und Halbleiter	Diverse Halbleitereffekte, z.B. Schwankungen der Rekombinationsrate	Rosa Rauschen Mit zunehmender Frequenz stark abnehmend	Gauß
Popcorn-Rauschen	„defekte Halbleiter“, vermeidbar durch Selektion	Sprunghafte Wechsel zwischen verschiedenen instabilen Zuständen eines Bauelements	unbestimmt	unbestimmt

k_B = Boltzmannkonstante, B =Bandbreite, T =absolute Temperatur

Integrale und differentielle Linearität

Bei Herstellerangaben von ADC-Bauelementen sind zwei Definitionen der Linearität gebräuchlich, die integrale und die differentielle Linearität. Folgendes Diagramm [FADC1] soll den Unterschied verdeutlichen :

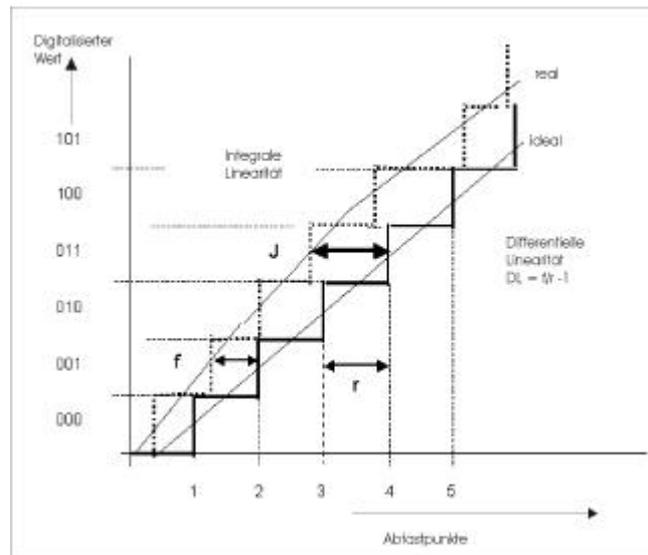


Abb. II.1. : Linearitätsbegriffe

Als integrale Linearität bezeichnet man die maximale Abweichung des tatsächlichen Kennlinienverlaufs von der idealen Kennlinie. Die differentielle Linearität ist der Quotient aus dem Betrag der fehlerbehafteten Schrittweite zur idealen Schrittweite.

Auflösung eines FADC-Systems

Es wird oft fälschlicherweise angenommen, dass die Genauigkeit einer Digitalisierung durch die Anzahl der Bit-Ausgänge des Analog-Digital-Wandlers bestimmt wird. In der Realität kann die absolute Genauigkeit sich aber so stark von dieser Angabe unterscheiden, dass einige der letzten Bits unrelevant sind (vgl. [STIEHLERDIPLOM]). Zur genauen Bestimmung der Genauigkeit eines Digitalisierungssystems ist die Aufstellung eines Fehleretats erforderlich, der sämtliche Fehlerquellen einschließt. Dazu gehören einerseits die Fehler, die bereits beim Einsatz eines idealen Umsetzers auftreten (Quantisierungsrauschen, Aliasing), andererseits sind störende Einflüsse durch Rauschen, Nichtlinearitäten und EMV zu berücksichtigen.

Die Auflösung eines ADC-Systems wird im allgemeinen in dB (Dezibel) angegeben. Das Dezibel ist eine dimensionslose Einheit. Man definiert die Auflösung als Signal-Rauschverhältnis zu

$$S/N = 10 \cdot \log \left(\frac{P_{\text{Signal}}}{P_{\text{Rausch}}} \right) [dB]$$

Man kann diese Angabe unter der Annahme einer sinusförmigen Spannung in sogenannte effektive Bits ENOB (effektive Bits, vgl. Kap.-IV.1., Sinustest) eines A-D-Wandlers umrechnen. Die Signalleistung einer sinusförmigen Spannung beträgt

$$P_{\text{sinus}} = \frac{1}{2 \cdot \sqrt{2}} \text{Amplitude}_{\text{spitze-spitze}}$$

der maximale Fehler beträgt $\frac{1}{2}$ LSB (Least Significant Bit), dessen Verteilung als Boxfunktion angenommen werden kann. Die Leistung dieser Gleichverteilung errechnet sich aus der Quadratwurzel des Mittelwerts aller Quadrate zu

$$P_{\text{rausch}} = \frac{1}{\sqrt{12}} \cdot \text{LSB}$$

Ein LSB entspricht

$$\frac{\text{AMPLITUDE}_{\text{spitze-spitze}}}{2^N}$$

Somit ergibt sich

$$\frac{P_{\text{signal}}}{P_{\text{rausch}}} = \frac{0,707 \cdot 0,5 \cdot \text{Amplitude}_{\text{spitze-spitze}}}{0,2886 \cdot \text{Amplitude}_{\text{spitze-spitze}}} \cdot 2^N$$

bzw. $P_{\text{signal}}/P_{\text{rausch}} = 1,227 \cdot 2^N$. Wenn man dieses Ergebnis in die Definition des Signal-Rauschverhältnis $S/N = 20 \cdot \log (P_{\text{signal}}/P_{\text{rausch}})$ einsetzt, ergibt sich die einfache Umrechnungsformel :

$$\text{Auflösung [ENOB]} = (\text{Auflösung [dB]} - 1,76) / 6,02$$

II.2. Grundlagen schneller Elektronik

II.2.1. Theorie von Übertragungsleitungen schneller Signale

Einführung

Die im Rahmen dieser Arbeit verwandte Technologie arbeitet mit Signalen, deren Anstiegszeit im Größenbereich von einigen hundert Picosekunden, der Jitter sogar im Bereich weniger Picosekunden liegt. Bei derartigen Signalen können die Übertragungswege zwischen einzelnen Bauelementen auf einer Platine und zwischen einzelnen Systemelementen nicht mehr als ideale Leitungen angesehen werden, sondern müssen als Hochfrequenzleitung betrachtet werden. Als Faustregel gilt, dass man ab einer Leitungslänge, bei der die Signallaufzeit größer als $1/8$ der Signalanstiegszeit wird, die Übertragungswege nicht mehr als ideale Leiter betrachten kann [MECL]. Für die verwendete LVDS (vgl. Kap.II.2., Abschnitt LVDS-Technologie, Seite 21)- bzw. ECLinps-Technologie (vgl. Kap.II.2., Abschnitt ECLinps-Technologie, Seite 20)- entspricht dies einer Leiterbahnlänge von wenigen Millimetern und trifft somit in der Praxis auf alle Signalübertragungswege von LVDS- und ECLinps-Signalen zu.

Vereinfachende Annahmen

Im folgenden soll nun eine kurze Einführung in die entsprechende Theorie gegeben werden. Da eine exakte Behandlung der Signaltheorie den Rahmen dieser Arbeit sprengen würde, werden bei den folgenden Betrachtungen einige vereinfachende Annahmen gemacht und Verluste durch Bandbreitenbegrenzungen, Dämpfung und Verzerrungen vernachlässigt. Diese Annahmen sind dann gerechtfertigt, wenn der Widerstand der Übertragungsleitung klein gegenüber dem Ausgangs- bzw. Eingangswiderstand des Senders bzw. Empfängers ist und wenn man gute dielektrische Eigenschaften des isolierenden Materials annimmt. Aufgrund dieser Vereinfachungen kann man das Verhalten von Übertragungsleitungen im wesentlichen durch die Parameter der charakteristischen Impedanz (Wellenwiderstand) Z sowie der Fortpflanzungsgeschwindigkeit v_{prop} bzw. dem daraus abgeleiteten Parameter $1/T_{\text{prop}}$ (vgl. folgender Abschnitt) beschreiben.

Die charakteristische Impedanz Z

Eine Hochfrequenzleitung kann als eine Kette von RCL-Gliedern aufgefasst werden, die auch verteilte Elemente genannt werden. Für jede Länge einer Leitung, bei der diese verteilten Elemente konstant sind, spricht man von charakteristischer Impedanz oder Wellenwiderstand der Übertragungsleitung. Da der ohmsche Widerstand –wie in der Einleitung dieses Kapitels ausgeführt– vernachlässigt werden kann, lässt sich die charakteristische Impedanz der Übertragungsleitung allein durch die Parameter L_{einh} und C_{einh} beschreiben. Man definiert den Wellenwiderstand als den Quotienten aus transienter Spannung $V_{\text{transient}}$ und dazugehörigen Strom $I_{\text{transient}}$ und erhält

$$Z = V_{\text{transient}} / I_{\text{transient}} = (L_{\text{einh}} / C_{\text{einh}})^{1/2},$$

wobei L_{einh} die Impedanz und C_{einh} die Kapazität pro Längeneinheit darstellt.

Die Fortpflanzungsgeschwindigkeit v_{prop}

Auch die Fortpflanzungsgeschwindigkeit v_{prop} bzw. der damit verbundene Parameter $1/T_{\text{prop}}$ lässt sich aus den Parametern L_{einh} und C_{einh} bestimmen. Man erhält

$$T_{\text{prop}} = (L_{\text{einh}} \cdot C_{\text{einh}})^{1/2}.$$

Aus den Gleichungen für den Wellenwiderstand und die Fortpflanzungsgeschwindigkeit lassen sich zwei Formeln zur einfachen experimentellen Bestimmung der Parameter L_{einh} und C_{einh} gewinnen :

$$L_{\text{einh}} = T_{\text{prop}} / Z \text{ und } C_{\text{einh}} = T_{\text{prop}} \cdot Z$$

Reflexionen

Im folgenden Schaltbild ist eine typische Signalübertragungsstrecke dargestellt:

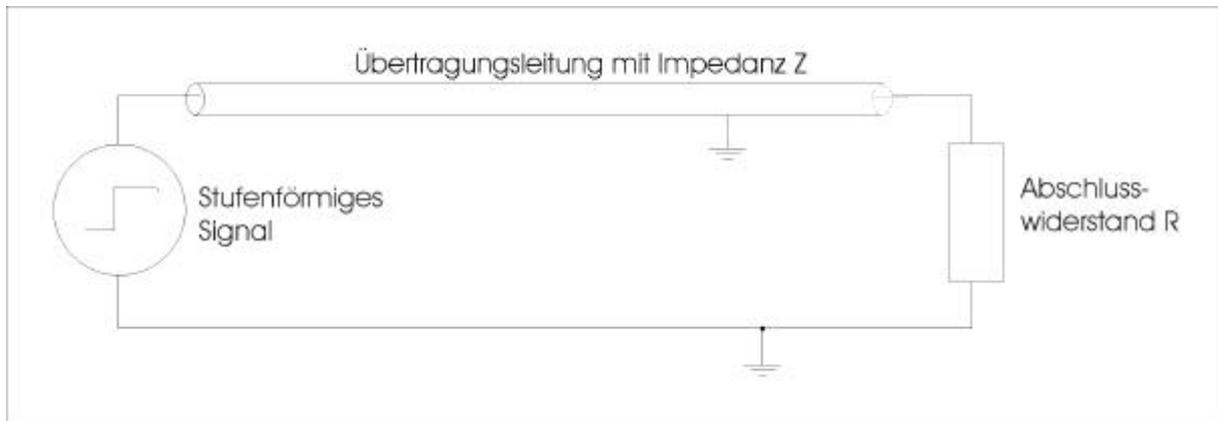


Abb. II.2. : Reflexion

In obiger Darstellung treibt eine Spannungsquelle über eine Übertragungsleitung Z einen ohmschen Widerstand R . Diese Schaltung entspricht einem häufig vorkommenden Signalübertragungsweg in der Praxis, wo eine Ausgangsstufe eines ECL-oder LVDS-Gatters über ein Kabel oder eine gedruckte Leiterbahn einen (von insgesamt zwei) entsprechenden differentiellen Eingang treibt. Wenn man den Ausgangswiderstand der Treiberstufe als vernachlässigbar und die Übertragungsstrecke als verlustfrei ansieht und den unteren Leiter als Spannungsreferenz nimmt, dann wird der Strom I_0 zur Zeit $t=0$ nach dem ohmschen Gesetz durch $I_0=V_0/Z$ bestimmt. Wenn die sich fortplanzende Welle den Widerstand R erreicht, dann muss auch zu diesem Zeitpunkt das ohmsche Gesetz erfüllt sein. Dies bedeutet, dass für den Fall $Z \neq R$ vom Verbindungspunkt der Übertragungsleitung Z mit dem Widerstand R aus eine Welle emittiert werden muss, die einlaufende Welle wird quasi reflektiert. Die von diesem Punkt rücklaufende Welle mit der Amplitude $V_0 \cdot \delta$ überlagert sich nun in jedem Punkt der Leitung mit der hinlaufenden Welle. Der Wert $\delta = (R-Z)/(R+Z)$ wird auch als Reflexionsfaktor bezeichnet. Wenn man den Ausgangswiderstand der Treiberstufe nicht –wie angenommen– vernachlässigen würde, dann gäbe dies wiederum Aufschluss zu einer weiteren auslaufenden Welle usw. [MECL].

Da Reflexionen die Störanfälligkeit der Schaltung vergrößern oder bei entsprechender Superpositionen gänzlich die Funktion der Schaltung beeinträchtigen können, muss man für einen definierten Wellenwiderstand Z sorgen und den sog. Abschlusswiderstand R daran anpassen. Wenn man in Ausnahmefällen aufgrund der Gegebenheiten keine kontrollierte

Impedanz erzeugen kann, so lassen sich zumindest weitere Reflexionen von einmalig reflektierten Wellen durch einen bei der Spannungsquelle in Reihe mit der Übertragungsleitung Z geschalteten Serien-Abschlusswiderstand mit $R_s = Z$ verhindern.

Da eine weitergehende und extensivere Behandlung dieses Themenbereiches den Rahmen dieser Arbeit sprengen würde, sei für eine Vertiefung der Thematik auf [MECL] und [WADELL] verwiesen.

Kontrollierte Impedanz

Wie bereits im vorherigen Kapitel angedeutet, ist bei Signalübertragungsleitungen darauf zu achten, dass Impedanzen kontrolliert werden. Im folgenden sollen Techniken vorgestellt werden, wie man dies realisieren kann.

Konsequenzen für Leiterbahnführung

Wie bereits ausgeführt, hängt die Impedanz Z von den Parametern L_{einh} und C_{einh} und somit direkt von der Geometrie der Übertragungsstrecke auf einer gedruckten Leiterplatte ab. Aus diesem Grunde sollte man bei der Leiterbahnführung von kritischen Signalwegen darauf achten, Winkel von 90° zu vermeiden. Das Verwenden von zwei kurz aufeinanderfolgenden 45° Winkeln oder im Idealfall eine viertelkreisförmige Leiterbahnführung stellt eine gleichbleibende Impedanz sicher und vermeidet somit Reflexionen.

Durchkontaktierungen bei differentiellen Signalen

Da Durchkontaktierungen in jedem Fall eine –wenn auch geringfügige– Änderung der Impedanz zur Folge haben, sollte mit Hinblick auf die Common-Mode-Rejection (vgl. Abschnitt “differentielle Signale“ dieses Kapitels) sowohl das invertierende Signal als auch das nichtinvertierte Signal eine Durchkontaktierung im selben Leiterbahnabschnitt durchlaufen.

Microstrip-Technik

Im folgenden Bild ist das Prinzip einer Microstripleitung dargestellt:

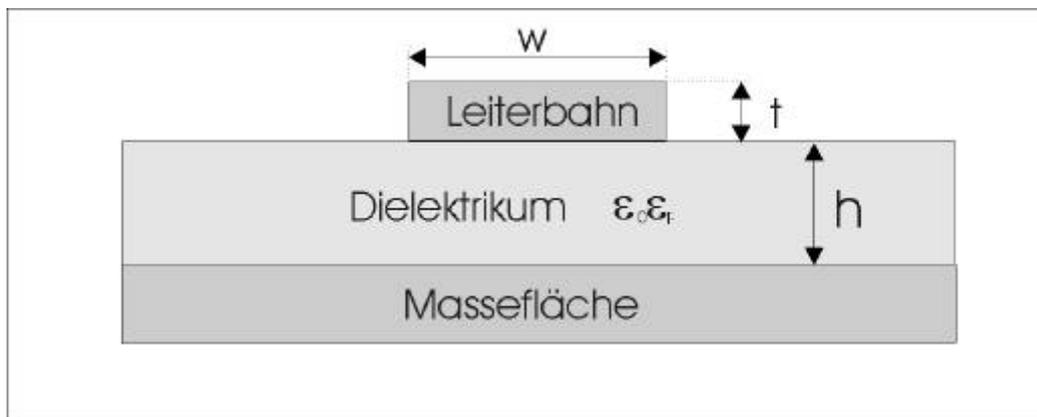


Abb. II.3. : Microstrip

Durch die offene Struktur des wellenleitenden Mediums und das geschichtete Dielektrikum breiten sich in Längsrichtung der Microstripline zwei Wellen mit unterschiedlicher Ausbreitungsgeschwindigkeit aus. Eine Welle bewegt sich im Innern des Substrats zwischen Leiter und Massefläche, die andere Welle bewegt sich im oberen Halbraum des Substrats. Die Wellenausbreitung ist aufgrund des Auftretens von Feldverzerrungen im geschichteten Dielektrikum nicht homogen. Dennoch wird dieser Leitungstyp bis zu einer gewissen Grenzfrequenz, bis zu der Feldkomponenten in Ausbreitungsrichtung vernachlässigbar klein sind, näherungsweise als TEM-Leiter (sog. Quasi-TEM-Leitung) behandelt. Die Berechnung des Wellenwiderstandes bei Verwendung eines Standard-FR4-Dielektrikums erfolgt nach der Formel

$$Z = 37,4 \cdot \ln \left(\frac{5,98 \cdot h}{0,8 \cdot w + t} \right)$$

Eine genaue Herleitung dieser Formel findet sich u.a. in [MECL] und [WADELL].

Microline-Technik

Im folgenden Bild ist das Prinzip einer Microlineleitung dargestellt :

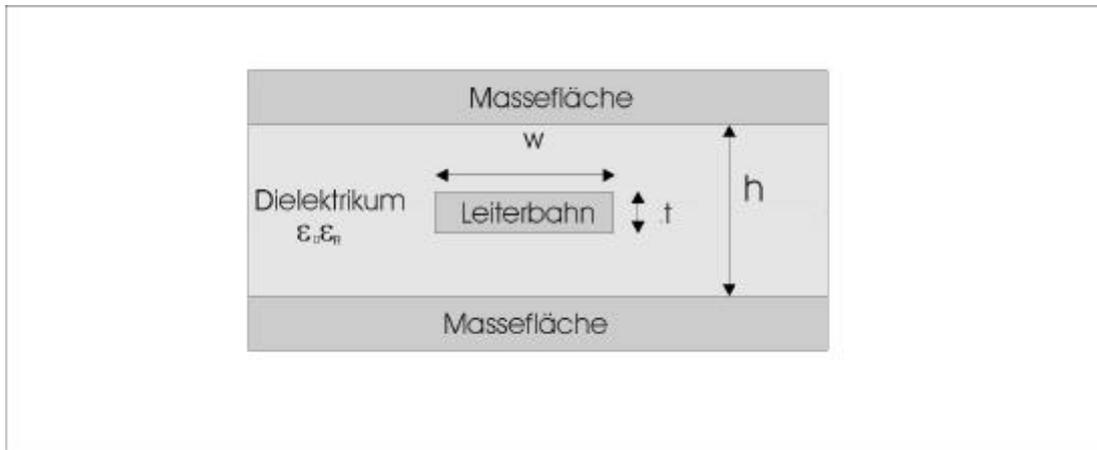


Abb. II.4. : Microline

Im Gegensatz zur Microstrip-Technik besitzt die Microline-Übertragungsleitung eine geschlossene Leiterstruktur mit einem homogenen Dielektrikum und kann aus diesem Grunde als TEM-Wellenleiter betrachtet werden

Die Berechnung des Wellenwiderstandes bei Verwendung eines Standard-FR4-Dielektrikums erfolgt nach der Formel

$$Z = 30 \cdot \ln \left(\frac{4 \cdot b}{2,01 \cdot (0,8 \cdot w + t)} \right)$$

Eine genaue Herleitung dieser Formel findet sich u.a. in [MECL] und [WADELL].

II.2.2. Übertragung differentieller und bipolarer Signale

Differentielle / bipolare Signale

Bei einer differentiellen bzw. bipolaren Signalübertragung wird das zu übertragende Signal vom Sender/Treiber sowohl im nicht-invertierten als auch in invertierten Zustand zur Verfügung gestellt. Beide Signale werden auf der Übertragungsstrecke eng nebeneinander geführt und liegen am Eingangsreceiver (Operationsverstärker) sowohl am invertierenden als auch nichtinvertierendem Eingang an. Durch Subtraktion beider Signale wird das Ausgangssignal gewonnen. Der große Vorteil bei der differentiellen oder bipolaren Signalübertragung liegt darin begründet, dass Störeinflüsse bei nahe aneinander liegenden Signalübertragungsleitungen sich gleichermaßen auf das invertierte Signal als auch das nicht-invertierte Signal auswirken (Common-Mode-Noise) und somit von der Subtraktionsstufe der Receivingereinheit eliminiert werden (Common-Mode-Rejection). Auch wenn äußere Störeinflüsse eher zu vernachlässigen sind, ist eine differentielle bzw. bipolare Signalübertragung von Vorteil, da von solchen Signalleitungen aufgrund destruktiver Interferenz der von beiden Leitern ausgehenden EMV-Strahlung weniger Störungen ausgehen. Die Nutzung der Vorteile dieser Technik hat einige Konsequenzen für das Leiterplattendesign, die in Kapitel 3 dieser Arbeit näher beschrieben werden.

ECLinps-Technologie

ECL ist eine Abkürzung für „Emitter-Coupled-Logic“ und bezeichnet mehrere Logikfamilien, bei deren Beschaltung das Ausgangssignal am Emitter der Ausgangsstufe entnommen wird. ECLinps ist eine dieser Logikfamilien, deren besondere Merkmale schnelle Anstiegszeiten (Größenordnung einige hundert ps) und ein geringer Jitter (Größenordnung $\ll 10$ ps) sind.

Die Eingangsstufe eines ECL-Gatters ist als hochohmiger Differenzverstärker ausgeführt. Das Prinzipschaltbild einer ECL-Ausgangsstufe ist im folgenden Schaltbild dargestellt:

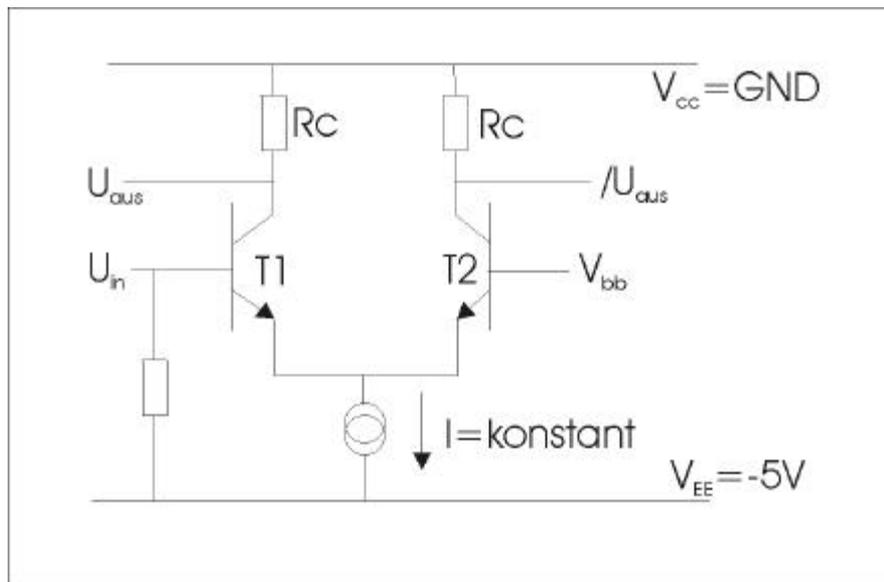


Abb. II.5. : ECL-Basisgatter (Ausgang)

Der durch die Konstantstromquelle fließende Strom wird über einen der beiden möglichen Pfade von V_{cc} (Masse) nach V_{ee} (-5V) geleitet. Die Verlustleistung eines Bausteins ist unabhängig von der Frequenz, da $P_v = V_{ee} \cdot I_s = \text{konstant}$ ist. Die Spannung V_{bb} legt die Schaltschwellen des Gatters fest. Da V_{bb} direkt auf Masse bezogen wird, ändern sich Schwankungsspannungen auf der Masseleitungen direkt auf die Rauschsicherheit aus, die um den Betrag des Spannungsabfalls auf der Masseleitung gemindert wird. An die niederohmigen Emitterfolger (Größenordnung 5-10 Ohm) lassen sich ECL-Gatter direkt an Übertragungsleitungen wie Koaxialkabel oder Twisted-Pair-Leitungen anschließen [MECL].

LVDS-Technologie

LVDS steht für die Bezeichnung Low Voltage Differential Signal. Im folgenden vereinfachten Schaltbild ist das vereinfachte Schaltbild einer LVDS-Übertragungsstrecke dargestellt:

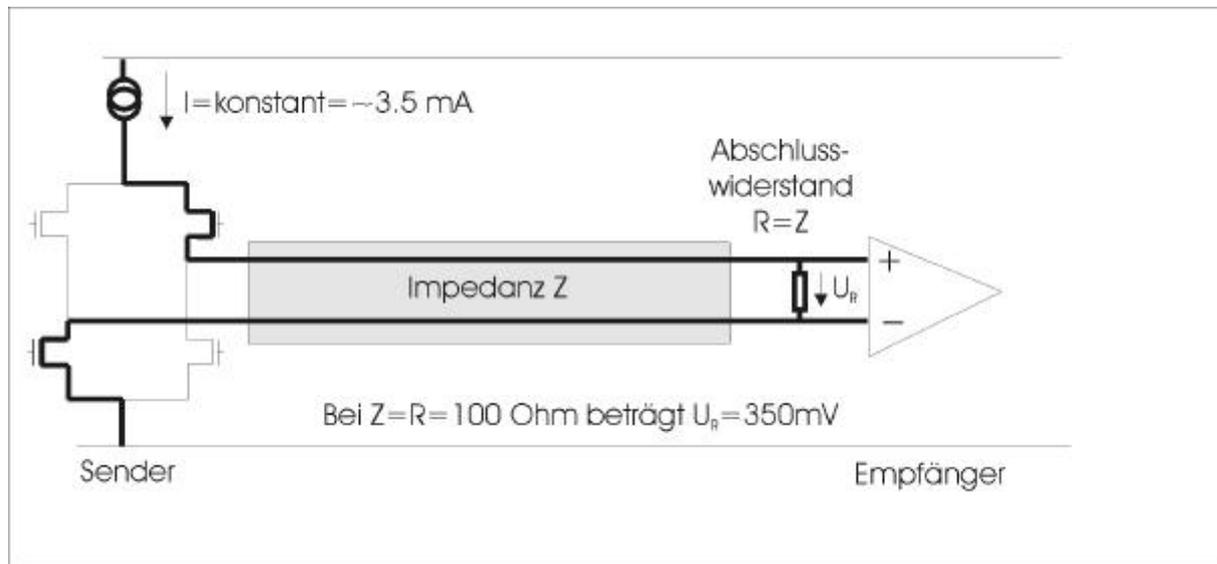


Abb. II.6. : LVDS-Übertragungsstrecke

Ein LVDS-Treiber besteht im wesentlichen aus vier Feldeffekttransistoren, von denen jeweils zwei durchgeschaltet sind. Eine Stromquelle treibt einen Strom von typischerweise 3-4 mA über die Übertragungsstrecke und den Receiveringang, die Richtung des Stroms ist im Diagramm angedeutet. Durch paarweises Umschalten der FETs des Treibers wird die Stromrichtung geändert, am Eingang des Receivers fällt eine positive (H=logisch 1) oder negative (L=logisch 0) Spannung ab, der Ausgang des Eingangsoperationsverstärkers geht in die Sättigung oder verharrt im Nullzustand.

Die Vorteile der LVDS-Technologie sind direkt aus diesem vereinfachten Schaltplan ersichtlich. Zum einen ist -wie bei ECL auch- die Leistungsaufnahme aufgrund gleichbleibender Abschlusswiderstände beim Receiver und einer konstanten Stromquelle im Sender über einen relativ großen Frequenzbereich konstant. Die aufgenommene Gesamtleistung ist bei LVDS aufgrund des Einsatzes von FETs und des relativ kleinen geschalteten Stroms in etwa eine Größenordnung niedriger als bei vergleichbar schnellen ECL-Schaltkreisen. Als weiterer Vorteil gegenüber ECL ist der geringer Spannungshub (300-400mV typ.) anzuführen, der eine geringere EMV-Emission zur Folge hat [LVDS1].

Ein anderer bemerkenswerter Aspekt hinsichtlich großer Systeme, in denen es leicht zu Differenzen von Massepotentialen kommen kann, ist der, dass LVDS-Elemente einen sehr hohen Störabstand von 0.05V-2-35V [LVDS2] haben, so dass sich diese Bausteine ideal zu Übertragungstrecken eignen, bei denen Empfänger und Sender geringfügige Unterschiede (<1V) im Massepotential haben.

Ein Nachteil von LVDS gegenüber ECL ist, dass bislang keine Logikfamilien mit geringen Jitter zur Verfügung stehen: Bei einigen ECL-Familien (ECLinps und höher) sind Jitter im Bereich von wenigen Picosekunden bereits Standard, bei LVDS liegt dieser Wert um Größenordnungen darüber. Als Konklusion dieses Abschnittes kann man zusammenfassend sagen, dass bei Frequenzen bis 400MHz und jitter-unkritischen Anwendungen der Einsatz von LVDS der ECL-Technik vorzuziehen ist.

III. Aufbau eines FADC - Systems

In diesem Kapitel werden die einzelnen Baugruppen und Module und deren Zusammenspiel im Gesamtsystem erläutert.

III.1. Gesamtübersicht

Photomultiplier und optischer Link

Zur Auslese des Magic-Teleskops werden Photomultiplier eingesetzt, die das Cherenkov-Licht in elektrische Pulse umwandeln. Zur Signalübertragung vom Teleskop zur Ausleseelektronik wurde ein analoger optischer Link gewählt, von konventionellen Koaxialkabeln wurde wegen des hohen Eigengewichts, hoher Kosten und relativ hoher Dämpfung abgesehen. Es sei an dieser Stelle auf [MAGIC3] und [MAGICPROP] sowie auf die Einleitung dieser Arbeit (Seite 4-6) verwiesen.

Optischer Receiver und Triggeregenerierung

Die Signale werden von einem optischen Receiver aufbereitet (vgl. Einleitung dieser Arbeit, Seite 6) und der FADC-Ausleseelektronik über Spezialkabel der Fa. 3M als bipolare Signale zugeführt. Ein Triggersignal zeigt das Vorhandensein eines Analogsignals an und veranlasst das FADC-Auslesesystem, die Signale auf allen Kanälen zeitgleich zu digitalisieren und über mehrere Zwischenspeicher in einen PC zur weiteren Datenverarbeitung weiterzuleiten. Pro auftretendem Triggersignal wird eine bestimmte Menge digitalisierter Daten gespeichert.

Gesamtübersicht des Transientenrekordersystems

Das Transientenrekordersystem besteht aus einzelnen FADC-Modulen, in denen die Analog-Digitalwandlung erfolgt. Aufgrund der räumlichen Größe der einzelnen Module und der großen Anzahl von Kanälen wurde ein eigenes LVDS-basiertes Bussystem entwickelt, auf dessen Basis die digitalisierten Signale aus insgesamt vier Racks über Backplanes, Interfaceboards und Hochfrequenzdatenkabeln zu einem Auslese-PC (DAQ-PC) gelangen. Die Signale werden durch Optokoppler galvanisch von PC-internen Signalen und Spannungen getrennt und über eine PCI-Karte in den PC übertragen.

Im Rahmen dieser Arbeit wurde das Konzept zur Datenaufnahme erstellt, die einzelnen Komponenten wurden entwickelt, getestet und deren Funktionsweise erläutert. Die vorliegende Arbeit ist eingebettet in den Gesamtkontext des „Magic-Projekts“, wodurch sich mehrere Schnittstellen mit anderen Systemkomponenten ergeben. Eine dieser Schnittstellen ist der Ausgang des optischen Receivers bzw. der Eingang des FADC-Motherboards, worauf bereits in der Einleitung dieser Arbeit eingegangen wurde. Weitere Berührungspunkte sind die Triggerbehandlung und zu einem gewissen Grad die PCI-Schnittstelle im DAQ-PC. Die Erstellung eines Konzepts zur Realisierung des Gesamtsystems mit FADC-Modulen zur Digitalisierung, Signalverarbeitung und Übertragung der Signale in einen PC zur Datenauswertung sind ebenso Thema dieser Arbeit wie Aufbau und Test der einzelnen Komponenten.

Schematische Darstellung der Funktionsweise des FADC-Transientenrekorders

Das Gesamtsystem besteht aus insgesamt 640 Kanälen, die Abtastfrequenz eines jeden Kanals beträgt 300 MHz. Auf die dynamische Auflösung wird im weiteren Verlauf dieser Arbeit näher eingegangen. Die Funktionsweise des Systems ist in folgenden Skizze dargestellt:

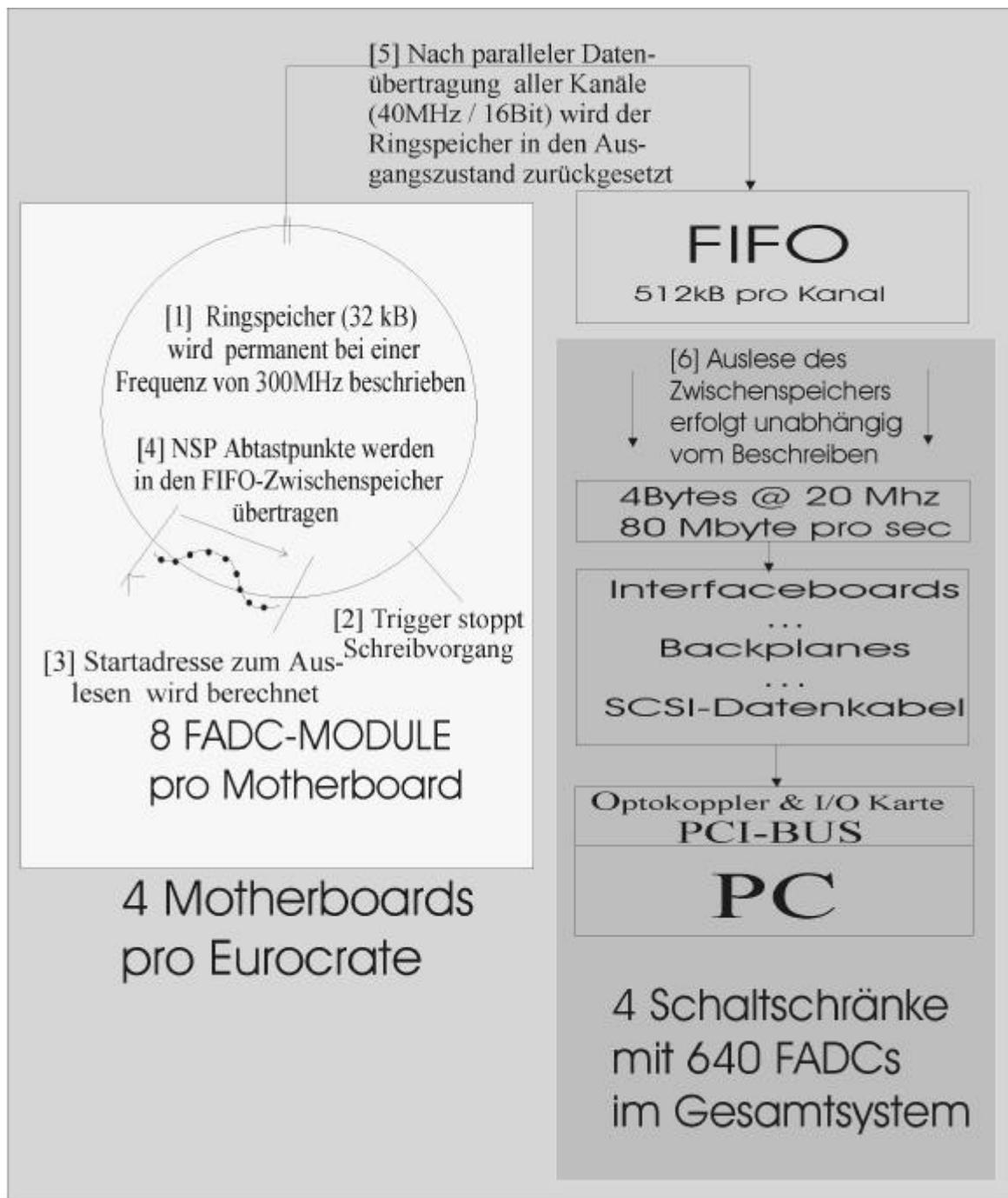


Abb. III.1. : Funktionsübersicht

Die Funktionsweise ist anhand der fortlaufenden Numerierung [1]...[6] der Auslesesequenz ersichtlich. Die verschiedenen Graustufen spiegeln die unterschiedlichen Ebenen des Systems (FADC-Modul, FADC-Motherboard, Multiplex-Auslesesystem) wider.

Das Prinzip der „digitalen Verzögerung“

Die FADCs digitalisieren permanent die anliegenden Analogsignale mit einer Abtastrate von 300MHz und legen die Daten in einem Ringspeicher ab. Wenn nun ein Triggersignal das Vorhandensein eines Analogsignals anzeigt, wird genau der Bereich des Ringspeichers ausgelesen, in dem sich das bereits digitalisierte Signal befindet. Diese Methode beruht auf folgender Überlegung : Da ein Triggersignal meistens aus dem Signal (z.B. ansteigende Flanke) abgeleitet wird, steht es in praktisch allen Fällen nach dem eigentlichen Signal an. In einigen seltenen Anwendungen kann ein Triggersignal auch zeitlich vor dem zu digitalisierenden Signal liegen.

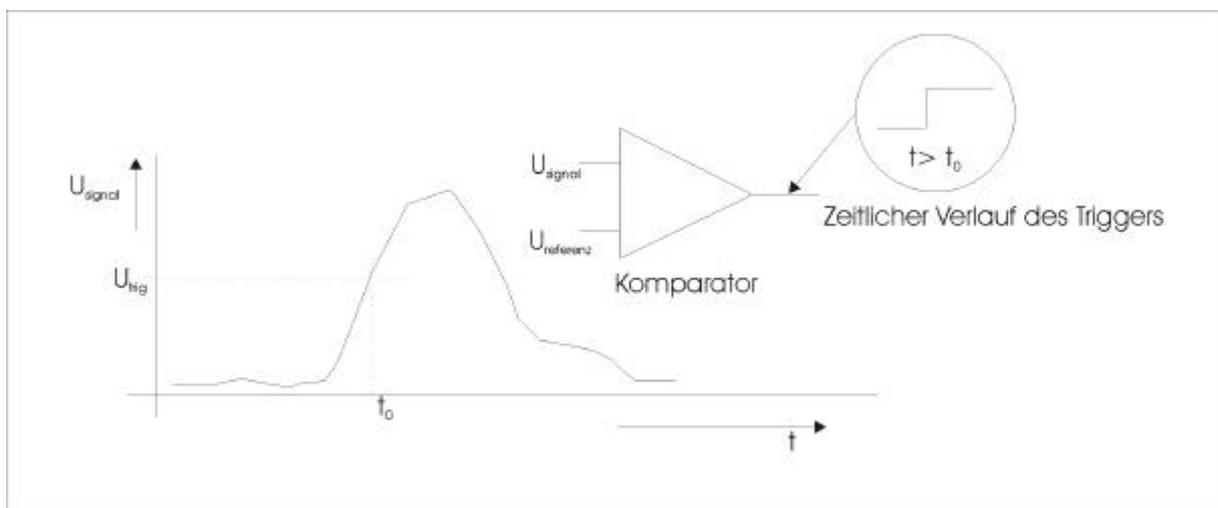


Abb. III.2. : Zeitverhalten des Triggers

Da man aber zur Datenauswertung möglichst den Gesamtverlauf des Signals erfassen möchte, muss man entweder das zu digitalisierende Signal verzögern oder unabhängig vom Triggersignal ständig digitalisieren. Nachteil einer Verzögerung des Analogsignals durch lange, relativ verzerrungsfreie Verzögerungsleitungen sind relativ hohe Kosten und eine nicht zu vernachlässigende Signaldämpfung. Je nach Aufbau des Experiments und auch aus Gründen der Kompatibilität des Transientenrekorders mit anderen Anwendungen liegt die erforderliche Verzögerung in einem Bereich von wenigen Nanosekunden bis hin zu einigen Mikrosekunden, was zu Kabellängen von bis zu mehreren hundert Metern führen würde. Bei Verwendung von diskreten Verzögerungselementen wie z.B. Eimerkettenspeichern (RLC-Glieder) treten neben hohen Kosten zusätzlich Bandbreitenprobleme und/oder Verzerrungen auf. Als einziger Vorteil einer Analogverzögerung wäre der geringe Aufwand auf Seiten der Ausleseelektronik anzuführen.

Bei der hier angewandten Methode einer „digitalen Verzögerung“ hingegen wird das Signal wie bereits erwähnt, ohne analoge Verzögerungsleitungen direkt digitalisiert und die Daten in einem Ringspeicher abgelegt. Gravierende Bandbreiten- oder Verzerrungsprobleme können praktisch nicht auftreten.

Die Parameter NSP und OFFS

Der Tiefe des Ringspeicher beträgt 32 Kbyte pro Kanal, was bei einer Abtastrate von 300 MHz rund 100 μ s entspricht. Aus der Gesamtheit des Ringspeichers wird nun ausschließlich der Bereich in einen weiteren Zwischenspeicher (FIFO) übertragen, in dem das digitalisierte Signal enthalten ist. Die ungefähre Gesamtdauer des zu digitalisierenden Signals sowie dessen zeitliche Lage in Bezug auf das Triggersignal, welches die Übertragung in die nächste Zwischenspeicherebene (FIFO) veranlasst, sind feste Parameter des Experiments. Beim „Magic-Projekt“ wird die Gesamtdauer eines aufzuzeichnenden Events rund 120 ns betragen, was bei der Abtastfrequenz von 300 MHz rund 40 Abtastpunkten entspricht, die Verzögerung beträgt $40 + \text{xoffset}$ wobei xoffset im wesentlichen durch Laufzeiten in der Triggerelektronik bestimmt wird und experimentell aus dem endgültigen Aufbau zu ermitteln ist. Die Parameter NSP (Anzahl der zu übertragenden Abtastpunkte) und OFFS (Verzögerung des Triggersignals) sind programmierbar und werden bei jedem Systemstart seriell in das FADC-System übertragen.

Die primäre Totzeit des Systems

Aus o.g. Systemparametern sowie der Übertragungsfrequenz f_{transfer} der Daten vom Ringspeicher in die nächste Zwischenspeicherebene (FIFO) kann man die sog. Totzeit des Systems bestimmen. Als primäre Totzeit T_{dead} bezeichnet man die Zeitdauer, für die das FADC-System aufgrund der internen Datenübertragung vom Ringspeicher zum FIFO keine Signale digitalisieren kann:

$$T_{\text{dead}} = \frac{20}{f_{\text{transfer}}} + \frac{0,5 \cdot \text{NSP}}{f_{\text{transfer}}}$$

Der erste Term $20/f_{\text{transfer}}$ wird durch innere Abläufe in der Steuerungselektronik eines FADC-Motherboards bestimmt. Hierzu zählen insbesondere die Berechnung der genauen Adresse der Ringspeichers, von der ab Daten übertragen werden sollen. Die Adresse zum Zeitpunkt des Triggersignals wird aus dem Adresszähler des Ringspeichers ausgelesen, ein fester Wert OFFS wird subtrahiert und das Ergebnis in den Adresszähler zurückgeschrieben.

Der zweite Term $0,5 \cdot \text{NSP} / f_{\text{transfer}}$ wird durch die Anzahl der zu übertragenden Abtastpunkte bestimmt, der Faktor 0,5 ergibt sich daraus, dass jeweils 2 Abtastpunkte ($2 \cdot 8 \text{Bit} = 16 \text{Bit}$) gleichzeitig übertragen werden. Wie man aus der Formel unschwer entnehmen kann, wird die Totzeit bei einer kleiner werdenden Anzahl von zu übertragenden Abtastpunkten zunehmend unabhängiger von NSP und immer deutlicher ausschließlich durch die Übertragungsfrequenz f_{transfer} bestimmt.

Die sekundäre Totzeit des Systems

Das Beschreiben und Auslesen der FIFOs geschieht völlig unabhängig (asynchron) voneinander. Aus diesem Grund tritt keine weitere Totzeit im System auf, sofern kein FIFO im Gesamtsystem überläuft. Der einzige variable Parameter hierbei ist die Frequenz der Triggersignale. Das System ist für eine bestimmte „Triggergrundfrequenz“ ausgelegt, bei der ein bestimmter innerer FIFO-Adresswert nicht überschritten wird. Kurzzeitig sind höhere Triggerfrequenzen zulässig, die dazu führen, dass der FIFO-Speicher kurzzeitig schneller gefüllt wird, als er ausgelesen werden kann. Die maximale Frequenz und Zeitdauer kurzzeitig erhöhter Triggerraten wird bestimmt durch die Speichertiefe der eingesetzten FIFOs sowie der durchschnittlichen Auslesefrequenz pro Kanal. Totzeiten, die dadurch entstehen, dass länger anhaltende Triggerraten über dem Nominalwert liegen, für die die Ausleseelektronik ausgelegt ist, werden im weiteren als sekundäre Totzeiten bezeichnet. Der Parameter NSP ist im MAGIC-Projekt auf $\text{NSP}=40$ festgelegt. Dies bedeutet, dass bei jedem Trigger eine Datenmenge von 40 Byte pro Kanal anfällt. Die Gesamtzahl der Kanäle beträgt 640. Die Auslese der FIFOs erfolgt standardmäßig mit einer Frequenz von 20 MHz, wobei immer 4 Byte parallel übertragen werden. Aus diesen Angaben kann man die eben eingeführte maximale Triggergrundfrequenz ermitteln:

$$f_{\text{trig max}} = \frac{80 \text{MByte/s}}{640 \cdot \text{NSP}} = 3,125 \text{ kHz}$$

Dies bedeutet, dass bei einer maximalen Triggerfrequenz $f_{\text{trig max}}$ keine sekundären Totzeiten auftreten. Die theoretisch für diesen Idealfall benötigte Speichertiefe der FIFOs betrüge $640 \cdot 40 \text{Byte} / 4 = 6,4 \text{ kB}$, ein Wert, der in Relation zur verwendeten Gesamtspeichertiefe der FIFOs von 512 kByte verschwindend gering ist. Allerdings kann das System nach Überschreiten von $f_{\text{trig max}}$ nur für eine begrenzte Zeit ohne sekundäre Totzeit arbeiten, da der FIFO-Speicher schneller beschrieben als ausgelesen wird.

Die maximale Zeit, mit der Daten ohne sekundäre Totzeit genommen werden können, beträgt somit

$$t_{\text{totzeitfrei}} = \frac{512 \text{ kByte} - 6,4 \text{ kByte}}{40 \text{ Byte} \cdot (f - 3,125 \text{ kHz})}$$

Diese Funktion ist im folgendem Diagramm graphisch dargestellt:

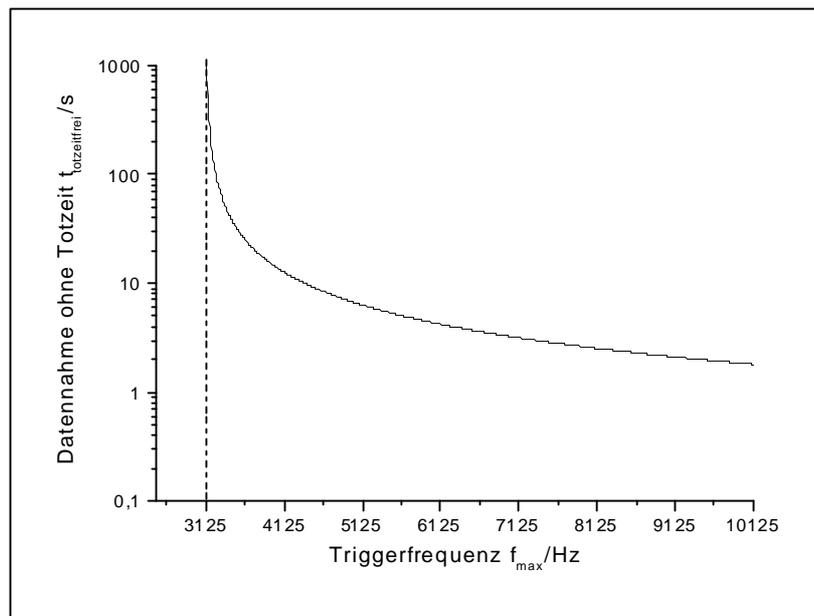


Abb. III.3. : Max. Dauer vs. max. Rate

Wie man erkennen kann, ist das System schon bei einem nur geringfügigen Überschreiten der Triggergrundfrequenz nur wenige Sekunden in der Lage, die hohe anfallende Datenmenge ohne Auftreten von Totzeiten (mit einhergehendem Datenverlust) zu verarbeiten.

Hierarchische Struktur des Auslesesystems

Nachdem in den vorherigen Abschnitten wesentliche Parameter definiert und erläutert wurden, soll im folgenden die Realisierung des FADC-Gesamtsystems dargestellt werden.

FADC-Module

Das Gesamtsystem für die Auslese des „Magic-Teleskops“ benötigt 640 Kanäle mit einer Abtastfrequenz von 300MHz bei einer Auflösung von $\text{ENOB} > 6,5$ Bit. Es werden an der Universität Siegen in konventioneller SMD-Technik entwickelte FADC-Module verwendet, die für die Anwendung geringfügig verändert wurden. Die Modifikationen betrafen zum einen die FPGA-Programmierung (vgl. Anhang B11) und eine Erweiterung des Ringspeichers, zum anderen wurde das Leiterplattenlayout des Analogteils optimiert.

8 FADC Module pro Motherboard

Jeweils 8 dieser Module befinden sich in einem FADC-Motherboard. Am Eingangsstecker des Boards anliegende Analogsignale werden zu den FADCs geleitet, digitalisiert und auf dem Ringspeicher der Module zwischengespeichert. Unterschiedliche DC-Level der Eingangssignale werden über eine programmierbare Baseline-Einstellung kompensiert, hierzu wird pro Kanal ein DAC zur Verfügung gestellt. Ein in Lattice-FPGAs implementiertes Steuerwerk stellt die Datenübertragung des Ringspeichers in die FIFOs sicher und übernimmt die Kommunikation mit der nächsten Interfaceebene (oder dem PC direkt). Programmierung, Auswertung und Multiplexing der FIFO-Flags sind in FPGA realisiert. Weiterhin wurde eine fortlaufende Triggernummerierung und ein sog. „Timestamp“ der digitalisierten Daten in FPGA implementiert, diese Daten werden in den Eingangsdatenstrom der FIFOs integriert. Der zeit- und jitterkritische Teil des Motherboards wurde in ECLinPS-Technologie auf dem Motherboard integriert. Dieser Teil umfasst vor allem die Verteilung und Verarbeitung von Clock- und Triggersignalen. Die Aufnahme dieser Signale erfolgt über 2 SMC-Stecker an der Frontseite jedes Motherboards. Bei der Auslese des „Magic-Teleskops“ wird aufgrund der langen Signalübertragungswege mit bipolaren LVDS-Signalen gearbeitet, die über impedanzangepasste Leitungen über ein System von Backplanes (impedanzangepasste Platinen), Interfaceboards und 68-polige SCSI-Standardkabel über Optokoppler in den Auslese-PC geleitet werden.

4 FADC-Motherboards und 1 Interfaceboard pro 19“-Crate

Jeweils 4 der FADC-Motherboards sind in Standard-19“-Crates untergebracht. Die bipolaren LVDS-Signale der Motherboards werden über eine eigens zu diesem Zweck entwickelte Backplane an ein Interfaceboard weitergeleitet. Dieses Interfaceboard steuert das Multiplexing der Daten- und Steuerleitungen und stellt an seiner Frontseite über 68-pol. SCSI-Standardstecker die entsprechenden Signale zur Verfügung. Die Interfaceboards sind so konzipiert, dass eine Datenübertragung sowohl von den Eingangssteckern auf der Frontseite zum Stecker der Backplane als auch umgekehrt möglich ist. Beide Varianten haben dasselbe Leiterplattenlayout, die Wahl der Richtung erfolgt über ein kleines FPGA. Auf diese Weise lassen sich mehrere Ebenen kaskadieren, eine Steuerung des Systems und Datenaufnahme per PC ist an jeder Zwischenebene des Systems möglich. Die in ECLinPS-Technik und mit Koaxialkabeln realisierte Clock- und Triggerverteilung ist räumlich gesehen auf der Front der Interfaceboards implementiert, elektrisch gesehen aber soweit wie nur möglich vom Rest der Schaltung getrennt.

5 FADC-Crates und 1 Interface-Crate pro Rack, 4 Racks für das gesamte System

Wie in den vorherigen Abschnitten erläutert, sind 32 Kanäle in einem Crate untergebracht. Das Gesamtsystem besteht aus 640 Kanälen, die Aufteilung der Crates auf verschiedene Racks wird zum einen durch die Spannungsversorgung der Module, zum anderen durch die begrenzten Räumlichkeiten der Messstation bestimmt. Folgende Skizze zeigt den Gesamtaufbau des Systems :

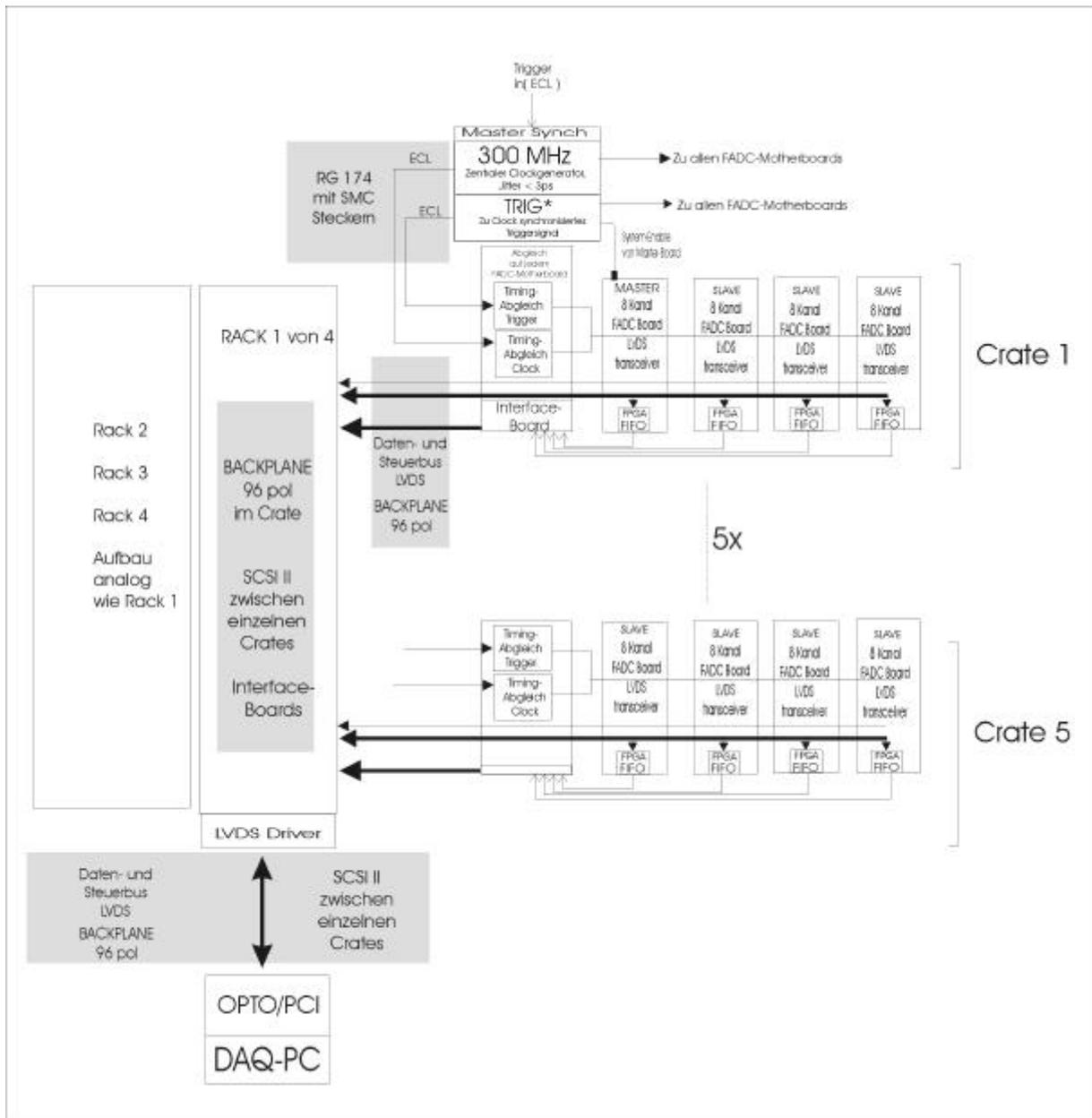


Abb. III.4. : Schematische Übersicht des Gesamtsystems

III.2. Das FADC – Modul

Als Basis des FADC-Auslesesystems wurde ein an der Universität Siegen in konventioneller SMD-Technik entwickeltes FADC-Modul verwendet. Im folgenden ist ein Bild des Moduls und seiner funktionalen Einheiten zu sehen, die im folgenden näher beschrieben werden:

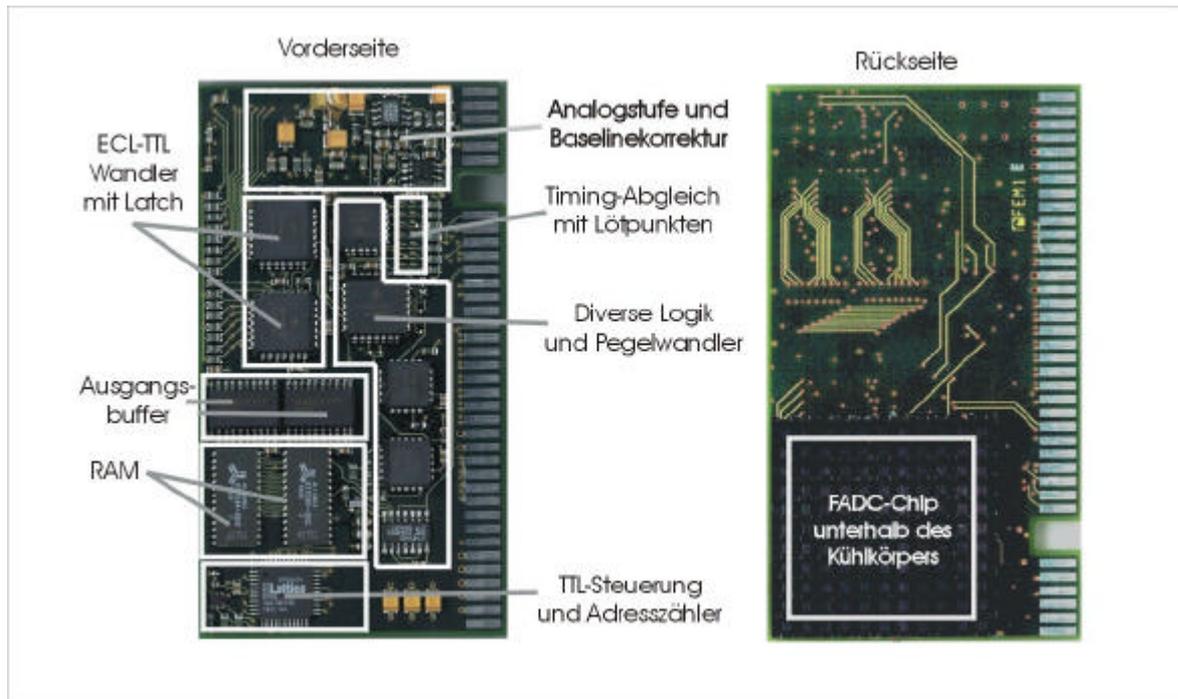


Abb. III.5. : Foto des FADC-Moduls

Dieses Modul wurde im Rahmen dieser Arbeit für die Anwendung modifiziert, hierzu wurden kleinere Änderungen am Digitalteil vorgenommen (Implementierung eines Emitterfolgers zur Ansteuerung des FPGA, FPGA-Programmierung und Erweiterung des Ringspeichers) und das Leiterplattenlayout des Analogteils optimiert.

Im folgenden sollen die wesentlichen Grundzüge des Moduls dargestellt werden, für eine weitergehende Beschreibung und die genauen Spezifikationen sei auf [VOLKOV] verwiesen. Im folgenden ist das grundlegende Blockschaltbild eines Moduls dargestellt, anhand dessen die Funktion erläutert wird:

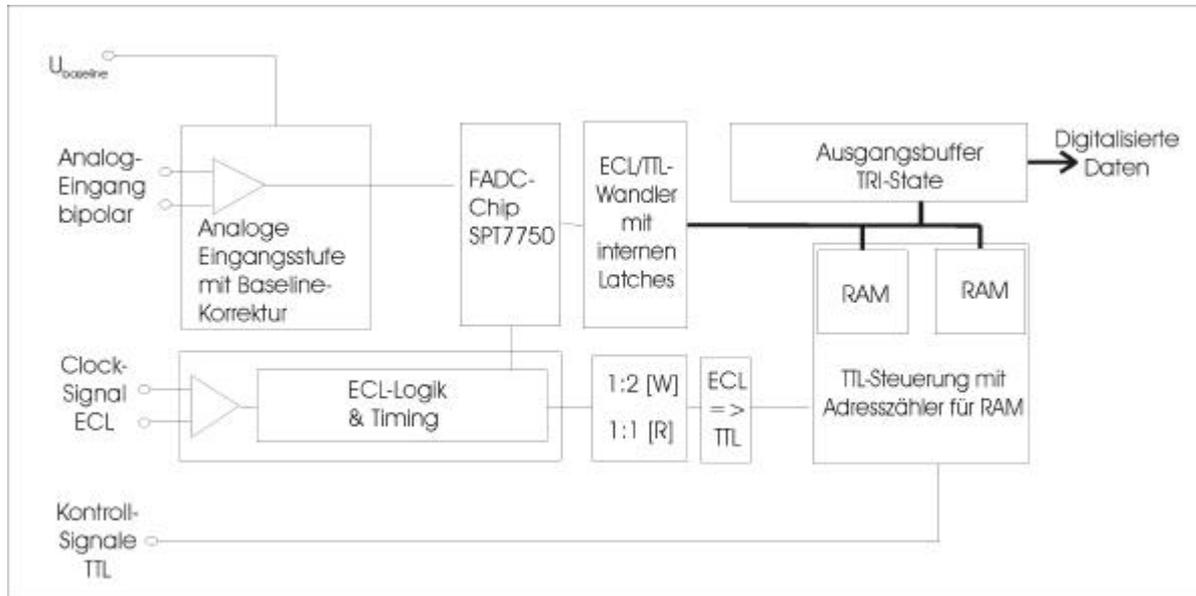


Abb. III.6. : Blockschaltbild des FADC-Moduls

Das FADC-Modul besteht im wesentlichen aus dem eigentlichen FADC-Chip samt vorgeschalteter Analogelektronik, diversen Pegelwandlern, dem Ringspeicher sowie einer Steuerungslogik. Das Modul arbeitet in zwei Modi, READ R und WRITE W, der aktuelle Modus wird der TTL-Steuerung über eines des TTL-Kontrollsignale (W/R) mitgeteilt. Ein gleichzeitiges Beschreiben und Auslesen ist nicht möglich, es kommt zu Totzeiten (vgl. Abschnitt zur primären Totzeit, Seite 27). Im WRITE-Modus wird die am Analogeingang anliegende Spannung mit der am ECL-Clockeingang anliegenden Abtastfrequenz digitalisiert. Die digitalisierten Daten werden am Ausgang des FADC-Chips als zwei 8-Bit-Wörter mit der halben Clockfrequenz (nominal 150 MHz) zur Verfügung gestellt, wobei eines der 8-Bit-Wörter um 180° phasenverschoben ist. Diese als ECL-Signale vorliegenden Daten werden über ECL-TTL-Konverter ins RAM übertragen, wobei das interne Latching dieser Pegelwandler die Phasenverschiebung der Datenwörter kompensiert. Da die TTL-Steuerungslogik im WRITE-Modus mit der halben Abtastfrequenz arbeitet, wird das DataReady-Signal des FADC-Chips verwendet.

Im READ-MODUS wird das Modul nicht mit der Abtastfrequenz (nominal 300MHz) gespeist, sondern mit einem Clocksignal, welches intern auf dem FADC-Motherboard (vgl. Kap.III. Das FADC-Motherboard, Seite 35) erzeugt wird. Dieses vom FADC-Motherboard zur Verfügung gestellte Clocksignal beträgt nominal 40 MHz und dient im READ-Modus lediglich zum Vortakten des Adresszählers.

Ein Herunterteilen der Frequenz für die TTL-Steuerung wie im WRITE-Modus erfolgt nicht. Die in der Skizze angedeuteten Kontrollsignale haben folgende Funktionen:

Kontrollsignal	Bedeutung	Funktion
W/R	Write/Read	Umschaltung zwischen Modi WRITE / READ
/PL	Parallel Load	Laden einer an den Adresseingängen liegenden Adresse in den internen Adresszähler
/RDA	Read Adress	Auslesen des aktuellen Zählerstandes
/ENA /ENB	Enable A/B	Anlegen der RAM-Daten über die Ausgangsbuffer an den Datenausgangsbus
/ENC oder Transfer	Transfer	Nur von Bedeutung im READ-Modus: 0 : zeigt dem Modul an, dass das FADC-Motherboard Adressberechnung durchführt (vgl. folgendes Kapitel) 1 : zeigt dem Modul an, dass das FADC-Motherboard Daten auslesen will (vgl. folgendes Kapitel)
/END	Keine	Zusätzliche Leitung zur TTL-Steuerung (FPGA), die zur Zeit nicht verwendet wird.

III.3. Das FADC – Motherboard

Das Motherboard ist räumlich gesehen in verschiedene Bereiche aufgeteilt, die bestimmte elektrische Funktionen und Eigenschaften erfüllen müssen. Folgende Aufnahme soll dies verdeutlichen:

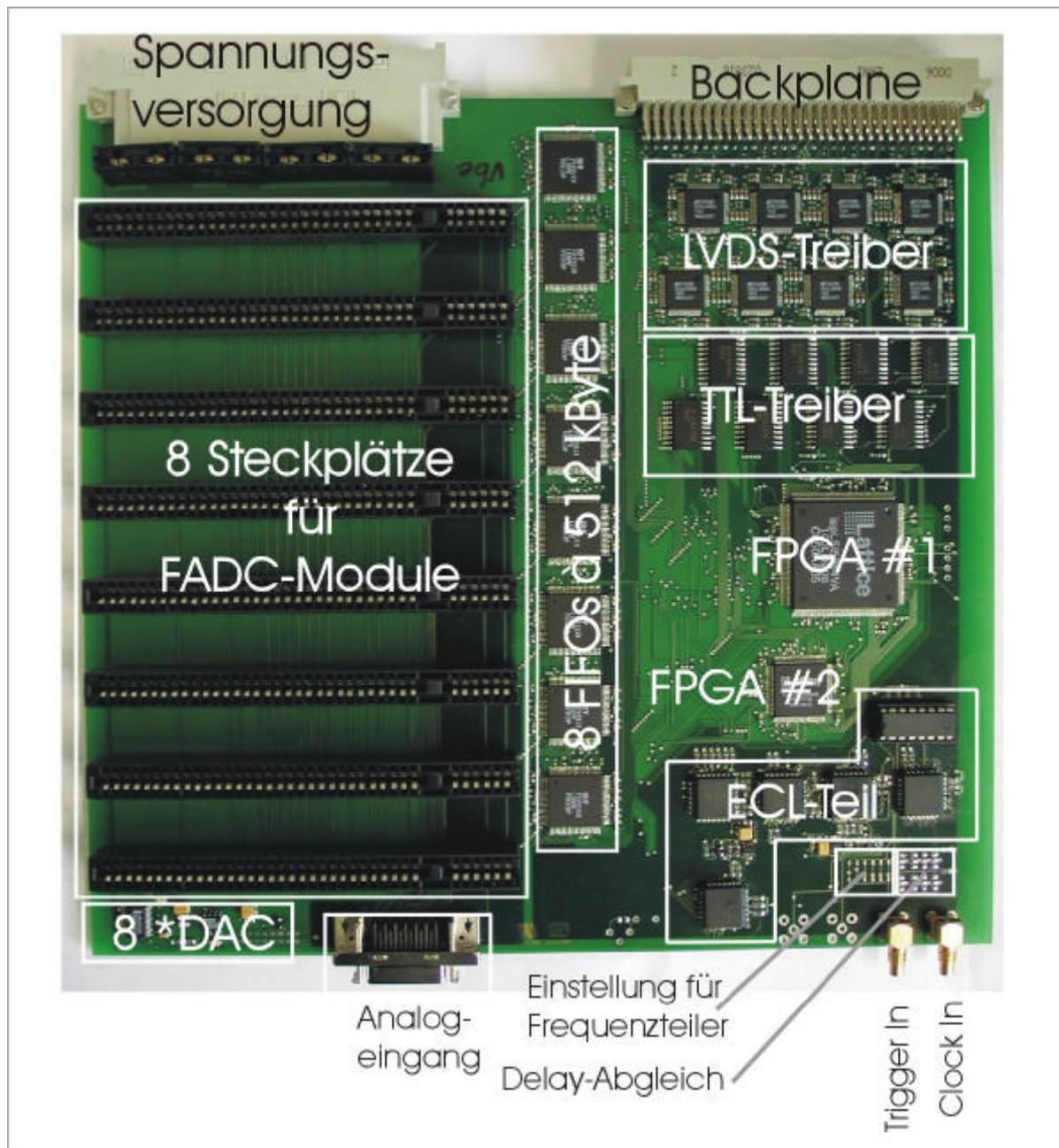


Abb. III.7. : Räumliche Aufteilung des FADC-Motherboards

FADC-Module und FIFOs

Fast 50% der Gesamtfläche wird durch die Steckplätze der FADC-Module und durch die FIFOs in Anspruch genommen. Bei den FIFOs handelt es sich um 512 kB große asynchrone Speicher, bei denen ein völlig voneinander unabhängiges Beschreiben und Lesen möglich ist.

DAC zur Generierung der Baseline

Im vorderen Bereich auf der linken Seite befindet sich gut getrennt vom Rest der Elektronik eine 8-fache Digital-Analog-Konverter-Schaltung (DAC) zur Generierung einer Baseline für jedes FADC-Modul. Bei der Initialisierung des FADC-Motherboards werden die Daten seriell vom PC aus in den DAC getaktet. Im folgenden ist das Schaltbild dieses Teils der Schaltung dargestellt :

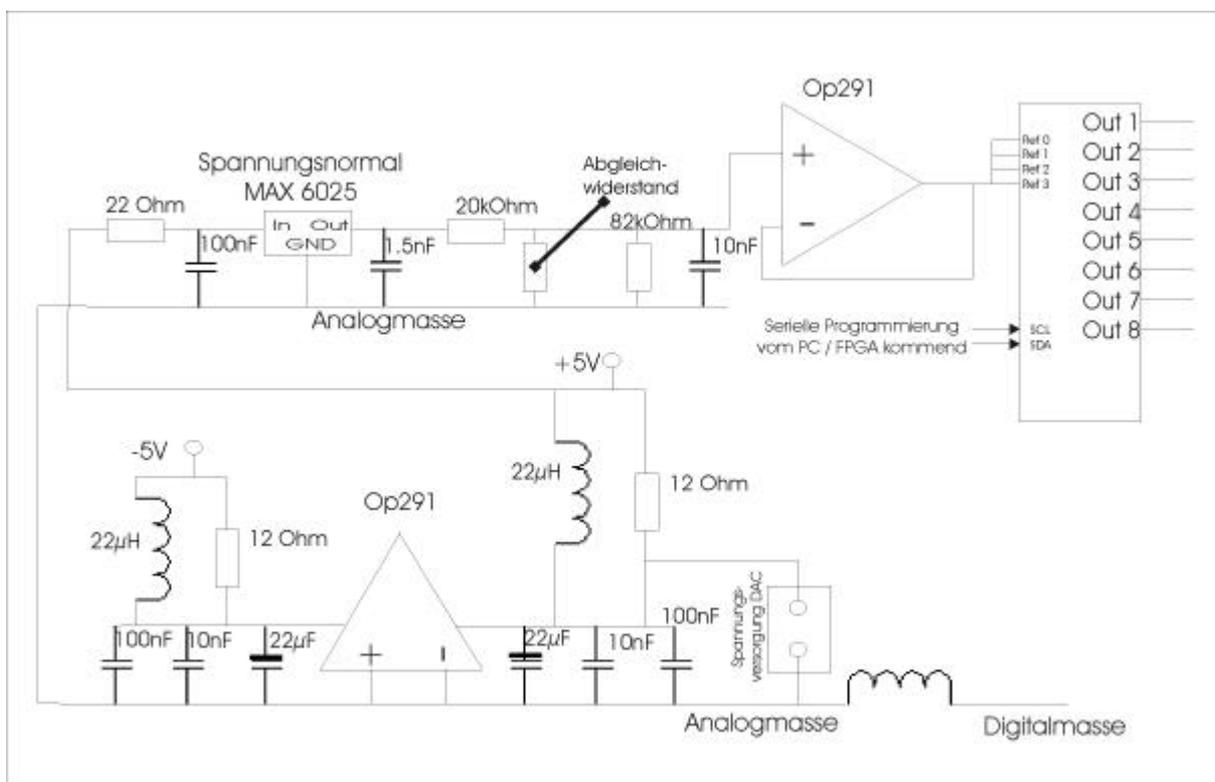


Abb. III.8. : Schaltplan der DAC-Ansteuerung

Die gesamte Schaltung ist durch mehrere Filter gut von Störeinflüssen auf der Spannungsversorgung abgeschirmt. Die Masse des Analogteils ist über eine niederohmige Impedanz von der digitalen Masse getrennt.

Dem eigentlichen DAC liegt ein Spannungsnormale zugrunde, das nur wenige Mikroampere treiben kann. Aufgrund des relativ niederohmigen Eingangs des DAC ist eine Operationsverstärkerschaltung zwischengeschaltet. Die Baseline steht als unipolares Signal zur Verfügung, die zugehörigen Massen werden von einem in der Nähe des DAC liegenden Sternpunktes aus zu den FADC-Modulen verteilt. Die analogen Ausgänge des DAC können die entsprechenden hochohmigen Eingänge der FADC-Module problemlos treiben.

Schnelle Elektronik : Der ECL-Teil

In diesem Bereich der Schaltung, der sich räumlich sehr begrenzt im vorderen Bereich der Leiterplatte nahe der Eingangsstecker befindet, werden die Clock- und Triggersignale empfangen und verarbeitet. Dieser Schaltungsteil lässt sich prinzipiell in die folgenden funktionellen Blöcke einteilen :

- Receiver für Clock und Trigger
- UND-Verknüpfung von Clock und Trigger
- Timing-Abgleich im ps-Bereich
- ODER-Verknüpfung von Abtast-Clock und Auslesetak
- Pegelwandler ECL-TTL / TTL-ECL
- Generation der FPGA-Systemclock
- Clockverteilung

Folgendes Blockschaltbild soll die Funktion verdeutlichen:

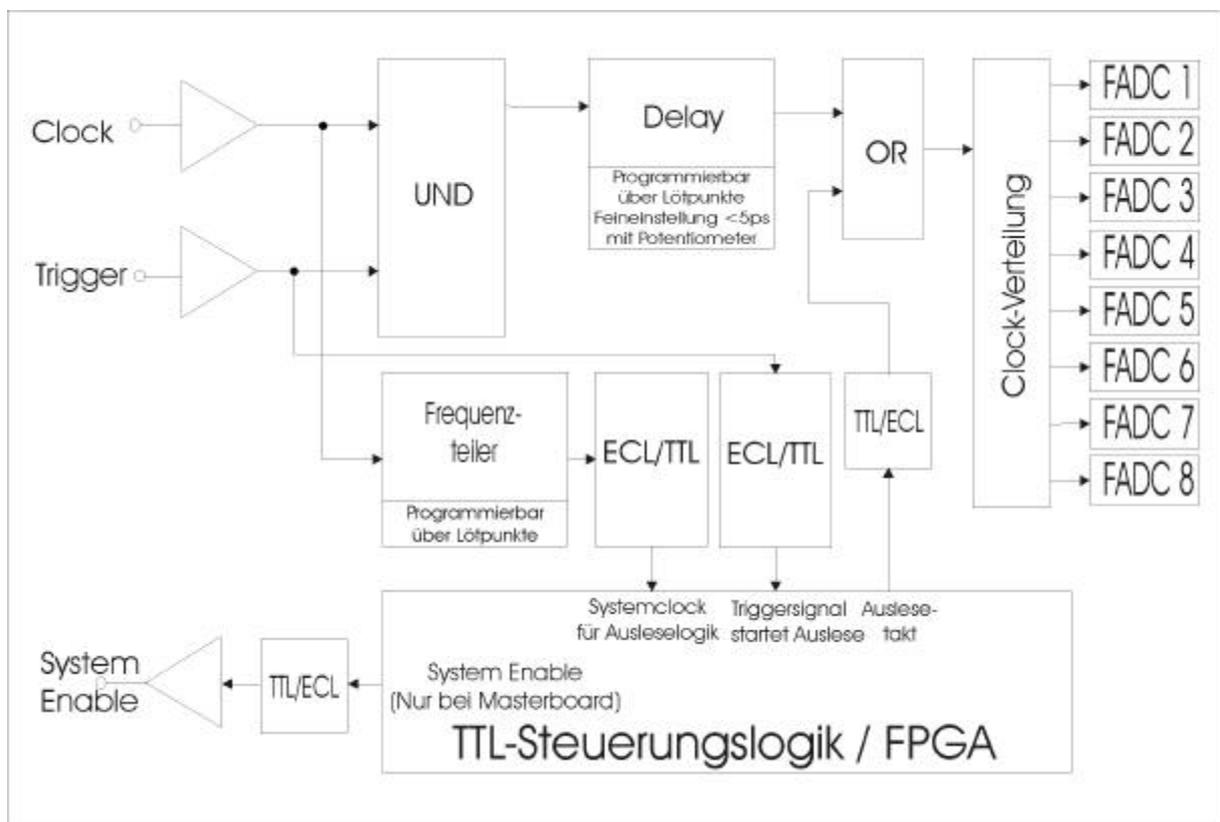


Abb. III.9. : Blockschaltbild des ECL-Teils des FADC-Motherboards

In den folgenden Abschnitten soll die Funktion der einzelnen Schaltungsblöcke erläutert werden.

Eingangsreceiver

Das Triggersignal und die Clock werden von ECL-Receiver aufgenommen, hierbei können die Signale sowohl als bipolare als auch unipolare Signale anliegen, eine Auswahl erfolgt über kleine Lötbrücken auf der Leiterplatte. Im unipolaren Betrieb wird eines der Eingangssignale des Differenzverstärkers auf den V_{bb} -Pin des Empfängerchips gelegt, der die internen Schaltschwellen festlegt. Bei anderen besonders zeitkritischen Anwendungen kann die Clock als bipolares Signal zugeführt werden, da nur in dieser Konfiguration vom Hersteller ein Jitter unter 5ps gewährleistet wird. Bei entsprechenden Messungen mit unipolarer Signalverteilung konnten allerdings keine schlechteren Werte gemessen werden. Aus diesem Grund (und Kosten- und Platzgründen) erfolgt die Verteilung von Trigger und Clock beim „Magic-Projekt“ als unipolare ECL-Signale. Auf dem Motherboard intern wurde auf zeitkritischen Signalwegen grundsätzlich mit bipolaren Signalen gearbeitet.

Erzeugung der Systemclock

Die interne Systemclock zur Steuerung des FADC-Motherboards wird über einen programmierbaren Frequenzteiler erzeugt. Als Eingangssignal dient die ständig am System anliegende Clock mit einer Frequenz von 300 MHz. Die Systemclock von $300 \text{ MHz} : 7 = 42,85 \text{ MHz}$ wird nach ECL-TTL Wandlung dem FPGA zugeführt. Die Ableitung der Systemclock aus dem Clocksignal ist zwar aufwendiger (und teurer) als die Verwendung eines zusätzlichen TTL-Clockbausteins auf dem Board, allerdings kann man hinsichtlich der Synchronisierung vielfältige Probleme vermeiden, da kein zusätzliches asynchrones Untersystem entsteht.

Weiterleitung des Triggers zur Steuereinheit / Pegelwandlung

Das Triggersignal dient als Initiator für die Übertragung der Daten auf dem Ringspeicher der FADC-Module in die FIFOs auf dem Motherboard. Diese Steuerung wird von einem FPGA übernommen, welches mit TTL-kompatiblen Ein- und Ausgängen arbeitet. Hier und auch bei der Systemclock ist eine Pegelumsetzung von ECL nach TTL erforderlich, welche aus Kostengründen diskret aufgebaut wird. Folgende Skizze zeigt den zugehörigen Schaltplan (Abblockkondensatoren nicht eingezeichnet) :

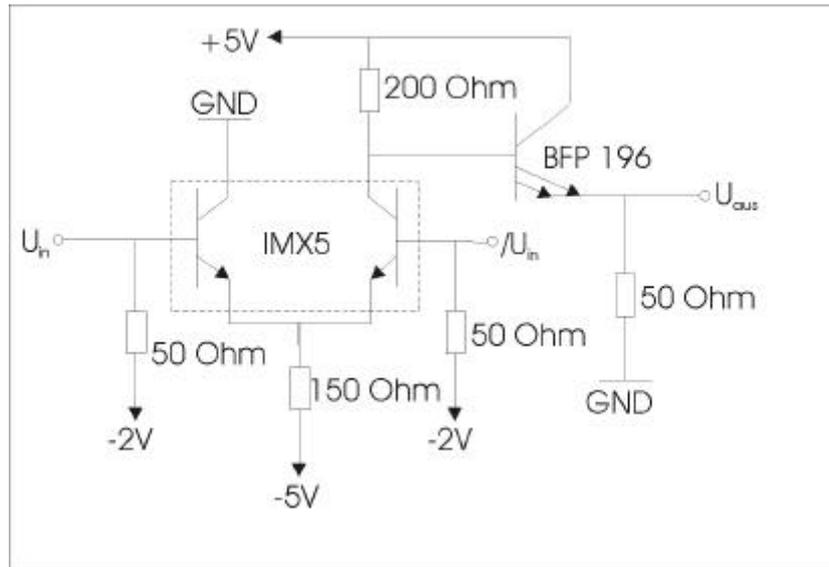


Abb. III.10. : ECL/TTL-Wandler

Abgleich von Trigger und Clock

Clock- und Triggersignal sind zwar durch die zentrale Einheit für Clockerzeugung und Triggersynchronisation synchronisiert, auf dem langen Signalweg von dieser Einheit zu den einzelnen FADC-Motherboards ist aber deren genaue Phasenbeziehung (z.B. wegen unterschiedlicher Kabellängen oder Fertigungstoleranzen bei den Clock- bzw. Trigger-Verteilungs-Chips) nicht mehr bekannt, sondern muss experimentell ermittelt und über Kabellängen abgeglichen werden. Ein Abgleich ist unbedingt erforderlich, da sonst durch eine Verknüpfung sehr kurze Impulse entstehen können, die vom nachfolgenden Gatter auf einem FADC-Motherboard als solche nicht mehr eindeutig zu erkennen sind. Dieser Abgleich ist relativ unproblematisch, da lediglich sichergestellt werden muss, dass die fallende Flanke des Triggers kommen muss, wenn sich die Clock im LOW-Zustand befindet. Die Ungenauigkeit darf an dieser Stelle somit mehrere 100ps betragen.

Das Clocksignal kann nach erfolgtem Abgleich mit dem Trigger in einem Bereich von 2ns mit einer Genauigkeit $<5\text{ps}$ verzögert werden, um somit die Synchronizität zwischen mehreren FADC-Motherboards untereinander sicherzustellen. Die Grobeinstellung (30ps-Bereich) erfolgt über ECL-Gatter, eine Feineinstellung bis $<5\text{ps}$ wird durch Anlegen einer über eine durch Potentiometer einstellbaren Spannung an die Schaltschwelle des letzten Gatters erreicht. Die Verzögerung in Abhängigkeit von der Steuerspannung kann folgendem Herstellerdiagramm [MECL] entnommen werden:

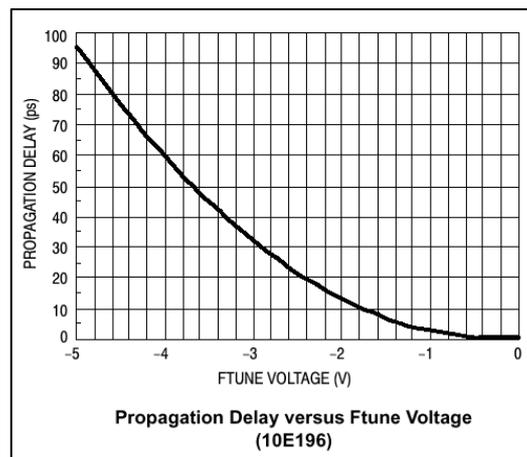


Abb. III.11. : Feineinstellung des Delays

Über das letzte ODER-Gatter gelangen entweder Clocksignale (300MHz) oder Systemclocksignale (42,85MHz) an die Clockverteilung, welche die Signale auf 8 FADC-Module verteilt. Diese Clockverteilung ist im vorderen Bereich der Leiterplatte angebracht, damit man den Clockleitungen dieselbe Länge zuordnen kann wie den zugehörigen Leitungen für die (auch im vorderen Bereich anstehenden) Analogsignale. Durch diesen Kunstkniff erscheinen für alle FADC-Module das Clocksignal und das Analogsignal synchron, unabhängig von der Entfernung des FADC-Moduls vom Eingangsstecker.

Verteilung des Analogsignals und der Auslese

Am Eingangsstecker des Boards anliegende Analogsignale werden über impedanzangepasste Leiterbahnen zu den FADCs geleitet. Diese Leiterbahnen sind räumlich weitestgehend von digitalen Leitungen getrennt.

Die Steuerung des Boards über 2 FPGAs

Die Abkürzung FPGA steht für „Field Programmable Gate Array“ und bedeutet eine konfigurierbare Ansammlung von programmierbaren Logikgattern in einem Halbleiterchip. Ein FPGA enthält eine große Anzahl an grundlegenden logischen Funktionen (Logikzellen). Eine Logikzelle selbst besteht (je nach FPGA-Typ) aus einer oder mehreren logischen Grundfunktionen und einem oder mehreren Flipflops. Die Verbindung der chipinternen Logikelemente erfolgt durch einen Programmiervorgang. Zur Steuerung des Boards werden FPGAs der Firma Lattice verwendet, deren Halbleiterchips aus der isp-Logikfamilie in der endgültigen Schaltung über die serielle Schnittstelle eines PCs programmiert werden. Die Entwicklung der zur Programmierung erforderlichen FPGA-Programmdatei (JEDEC-Format) wird im Anhang dieser Arbeit beispielhaft erläutert. Die Steuerung des Boards erfolgt über zwei FPGAs der Firma Lattice. Die Entscheidung zugunsten von zwei einzelnen statt einem einzigen FPGA liegt hauptsächlich darin begründet, dass die Anzahl der I/O-Pins begrenzt ist. Die Verwendung eines einzigen FPGAs hätte dazu geführt, dass man ausschließlich ein (besonders in der Entwicklungsphase) schwer zu handhabendes BGA-Gehäuse verwenden müsste. Funktional sind die beiden FPGAs so getrennt, dass der „kleine Chip“ vor allem das Multiplexing der FIFO-Flag-Logik und die Generation quasi-statischer Signale übernimmt, während der „große Chip“ die eigentliche Steuerung zur Aufgabe hat. Diese Steuerung ist stark vereinfacht im folgenden Blockschaltbild dargestellt:

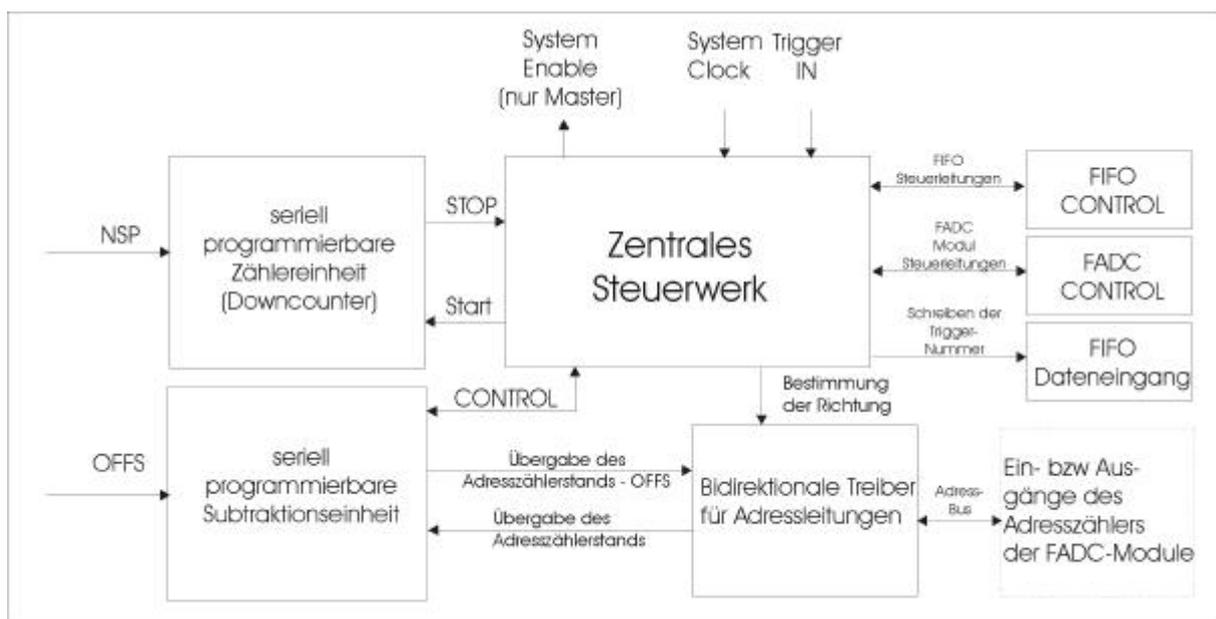


Abb.III.12. : FPGA-implementierte Steuerlogik

Der prinzipielle Ablauf der Steuerungslogik wurde bereits im Abschnitt „Schematische Darstellung der Funktionsweise des FADC-Transientenrekorders“ dieses Kapitels erläutert. Die in ein FPGA der Firma Lattice implementierte Steuerungslogik hat Schnittstellen zu den Kontrollleitungen der einzelnen FIFOs bzw. FADC-Module. 32 der insgesamt 128 FIFO-Dateneingänge sind über Tristate-Eingänge mit dem Steuerwerk verbunden, damit man die Triggernummer in den Datenstrom integrieren kann. Die Adressleitungen der FADC-Module sind über bidirektionale Leitungen mit dem Steuerwerk verbunden. Auf diese Weise kann der jeweilige Adressstand der einzelnen Module bestimmt und die Adresszähler der entsprechenden FADC-Module auf die Neuberechnete Adresse gesetzt werden. Die Parameter NSP (Anzahl der zu übertragenden Abtastpunkte) und OFFS (Verzögerung des Triggersignals) werden bei jedem Systemstart seriell in das FADC-System übertragen. Die einzelnen Module wurden in ABEL/HDL realisiert und mithilfe des Programmpakets Synario über mehrere Ebenen von Schematics zu einem Gesamtschaltplan verbunden. Die einzelnen Module, die Schematics und das kommentierte Programm des zentralen Steuerwerks befinden sich in Anhang B dieser Arbeit.

Schnittstellen des Boards

Das Board besitzt Schnittstellen sowohl für TTL als auch LVDS. Für die Auslese des „Magic-Projekts“ wird aufgrund der langen Signalübertragungswege mit bipolaren LVDS-Signalen gearbeitet, die über impedanzangepasste Leitungen über ein System von Backplanes (impedanzangepasste Platinen), Interfaceboards und 68poliges SCSI-Standardkabel über Optokoppler in den Auslese-PC geleitet werden. Die TTL –Schnittstelle dient zu Testzwecken und zur einfachen Auslese einzelner Boards bei kleineren Anwendungen.

III.3. Die Signalverteilung im System

Übersicht

Die Signalverteilung im System erfolgt über ein kaskadierbares System aus Interfaceboards, SCSI-Kabeln und Backplanes. Aufgrund der langen Signalübertragungswege wird mit differentiellen LVDS-Signalen gearbeitet. Folgende Skizze zeigt die Realisierung der Signalverteilung im System :

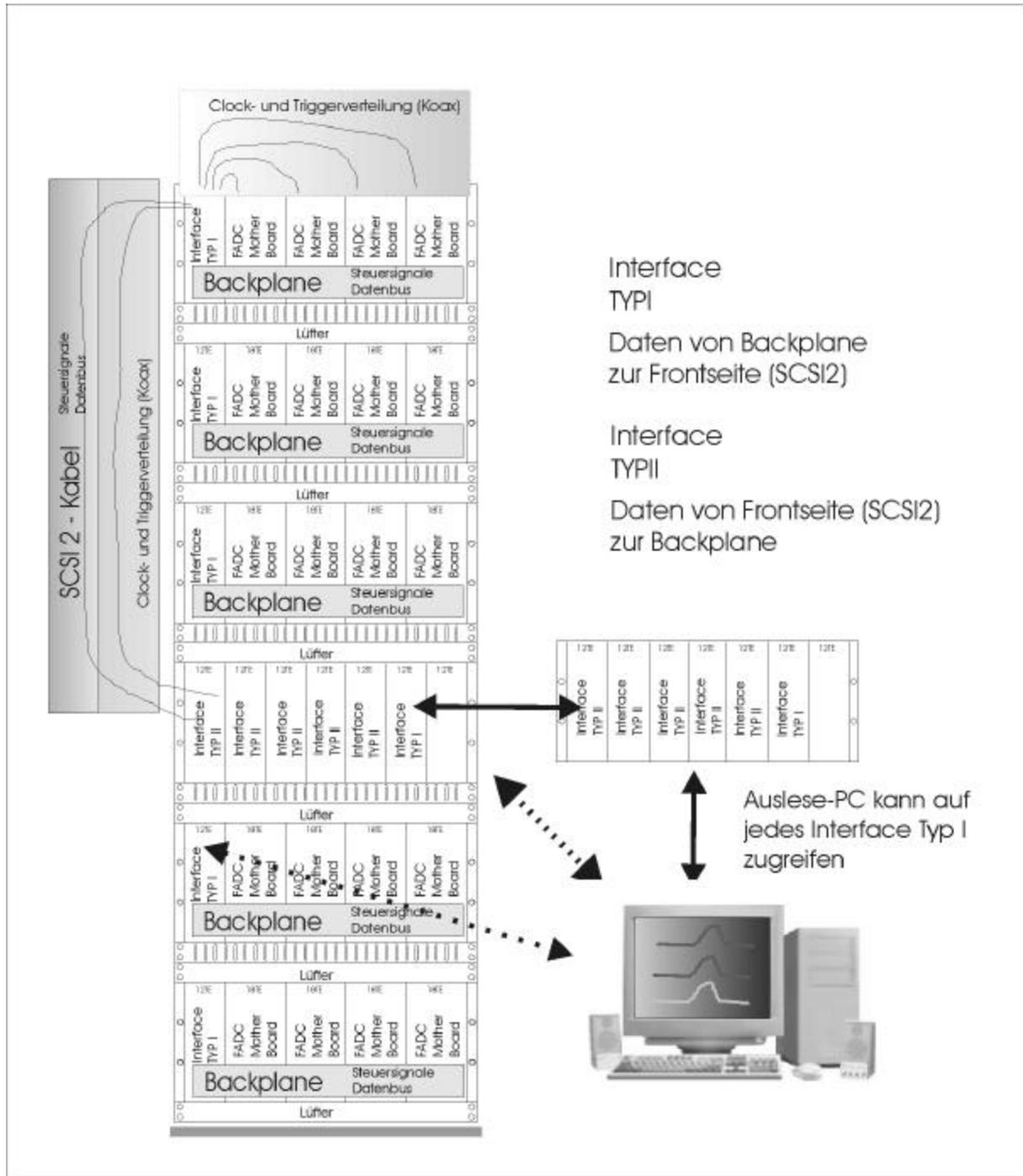


Abb. III.13. : Signalverteilung im System

Es lassen sich mehrere Ebenen kaskadieren, eine Steuerung des Systems und Datenaufnahme per PC ist an jeder Zwischenebene des Systems möglich.

Backplane

Die am FADC-Motherboard anstehenden Signale werden über impedanzangepasste Leitungen an eine Backplane übergeben. Diese impedanzangepasste Platine wurde eigens für die Auslese des Magic-Teleskops an der Universität Siegen entwickelt. An der Vorderseite werden die 128-poligen Stecker per Einpresstechnik mit der Platine verbunden, an der Rückseite wird beidseitig ein Stecker mit Abschlusswiderständen angebracht. Die räumliche Anordnung der Steckplätze wurde so vorgenommen, dass entweder 4 FADC-Boards und 1 Interfaceboard oder aber ausschließlich Interfaceboards verbunden werden können.

Interfaceboard

Das Interfaceboard arbeitet im Multiplexbetrieb und nimmt die Signale von jeweils 4 FADC-Boards seitens der Backplane auf und leitet sie an SCSI-Stecker an der Frontplatte weiter. Im folgenden ist das Interfaceboard und seine funktionale Unterteilung dargestellt :

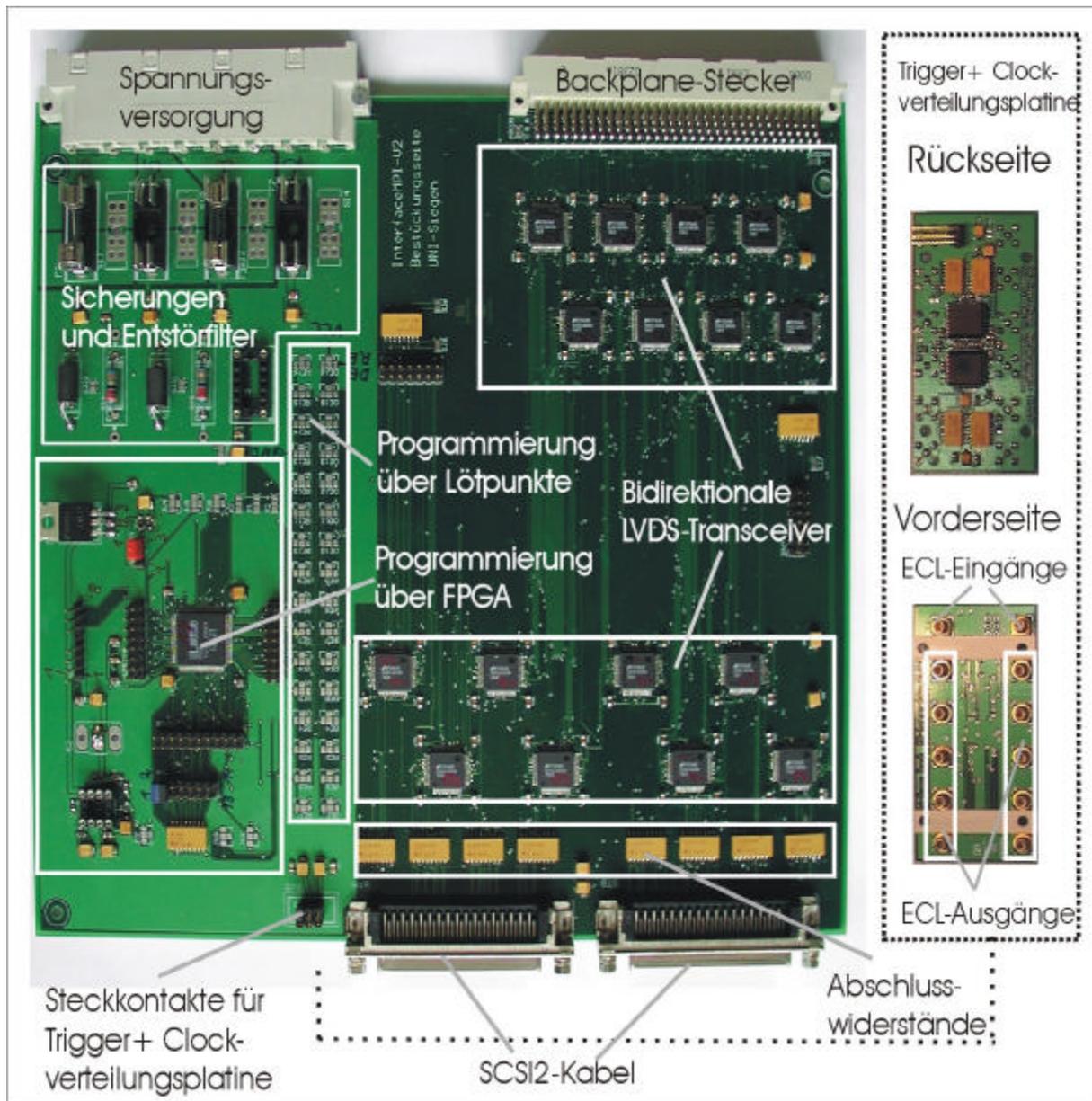


Abb.III.14. : Interfaceboard

Die genauen Aufgaben des jeweiligen Interfaceboards an seinem Platz im System werden über ein kleines FPGA vorgegeben. Dieses FPGA bestimmt Multiplexfunktionen und die Richtung der Daten- und Steuersignale. Wenn in anderen, kleineren Systemen keine Multiplexfunktionen benötigt werden, dann kann die Datenrichtung auch wahlweise über Lötunkte auf dem Board programmiert werden.

In einem reinen Interface-Crate nehmen jeweils bis zu 4 Interfaceboards die Daten- und Steuerleitungen aus 4 FADC-Crates mit je 32 Kanälen an der Frontseite auf und leiten die Signale im Multiplexbetrieb zur Backplane weiter. In einem weiteren Interfaceboard werden diese Signale wieder zur Frontseite geleitet und stehen dort der nächsten Ebene an Interfaceboards zur Verfügung.

Daten- und Steuerleitungen zwischen Crates

Als Bindeglied zur nächsten Ebene werden geschirmte SCSI-Kabel mit 34 verdrehten Aderpaaren (Twisted-Pair) verwendet, die Impedanz jedes Kabelpaars beträgt ca. 120 Ohm. An den Kabeln wurden Testmessungen hinsichtlich der maximalen Datenrate und des Übersprechverhaltens durchgeführt. Hierbei konnte gezeigt werden, dass die Kabel im System problemlos mit der angestrebten Auslesefrequenz von 20MHz (bzw. später bis max. 35MHz, vgl. Kap. 4, Messungen am SCSI-Kabel, Seite 70) betrieben werden können.

III.4. Die zentrale Clockerzeugung und Triggersynchronisation, Verteilung von Clock und Trigger

Gesamtüberblick

Im folgenden Bild sind die einzelnen Elemente der zentralen Clock- und Triggerverarbeitungseinheit zu sehen :

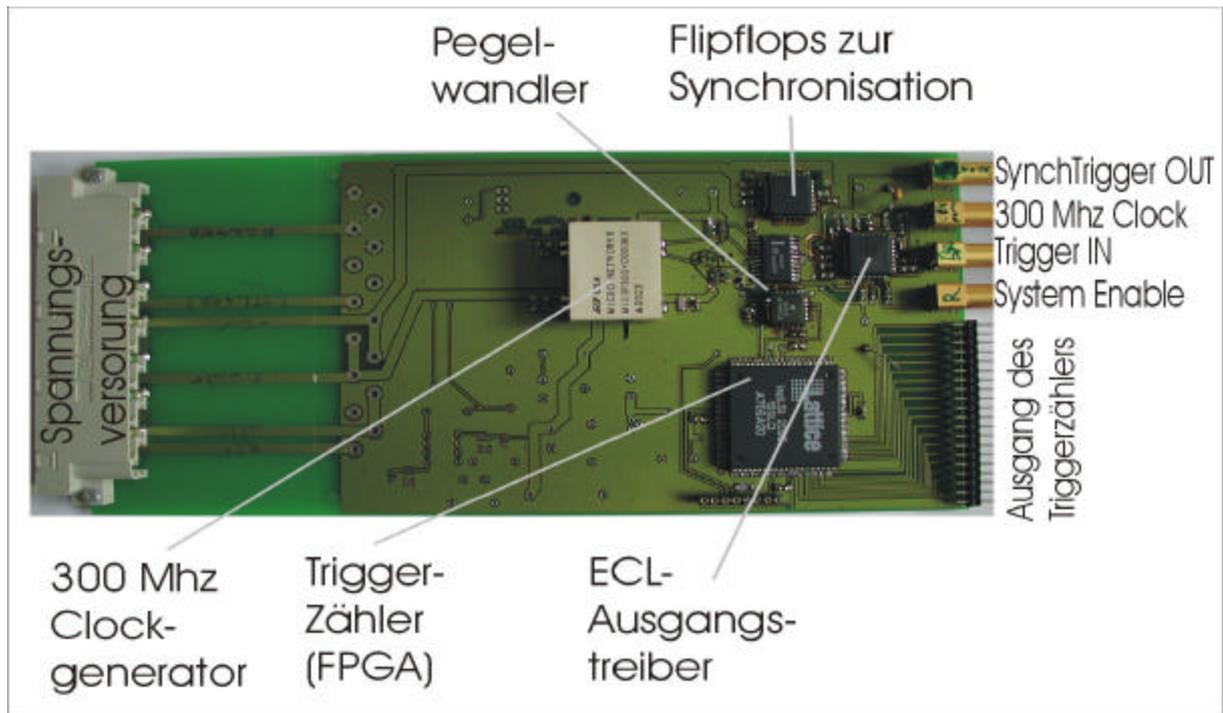


Abb.III.15. : Zentrale Clock- und Triggerverarbeitung

Die Funktionen dieser Einheit sollen anhand des folgenden Blockschaltbildes erläutert werden :

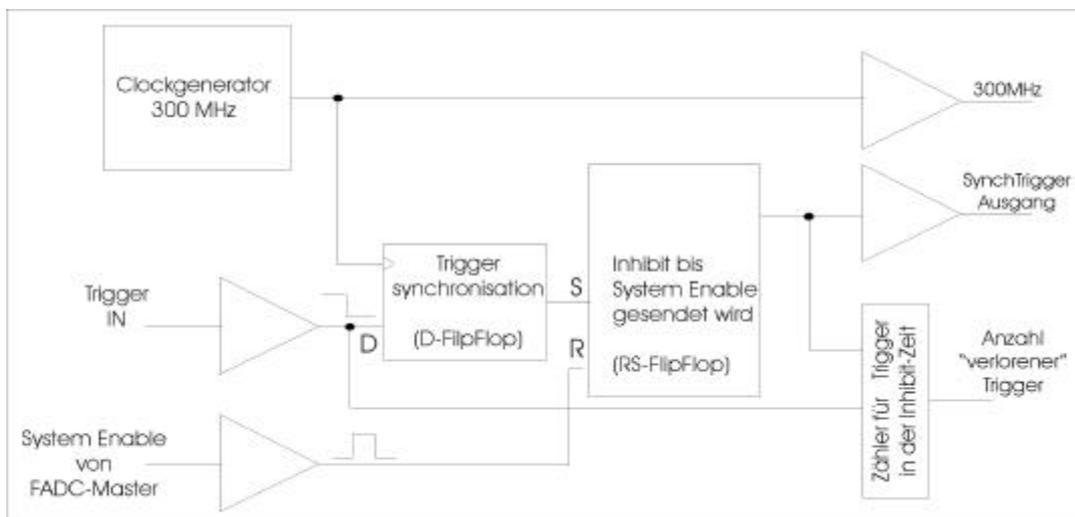


Abb. III.16. : Blockschaltbild der zentralen Clock- und Triggerverarbeitung

Das in einem zentralen Präzisionsoszillator erzeugte 300MHz-Signal wird aufgeteilt, wobei ein Zweig direkt zu den Ausgängen führt, ein anderer Zweig wird dem Synchronisationsflipflop zugeführt, welches die steigende Flanke des Triggersignals mit der Clock synchronisiert. Das zweite Flipflop dient dazu, das am Eingang der Schaltung anstehende Triggersignal solange zu verlängern, bis die Datenübertragung in den FADC-Motherboards (Ringspeicher => FIFOs) abgeschlossen ist. Das Triggersignal setzt dieses Flipflop und das Masterboard im System setzt es nach abgeschlossener Datenübertragung wieder zurück. Um während dieser primären Totzeit eingehende Trigger zahlenmäßig zu erfassen, wird auf der Platine ein Zähler in als FPGA implementiert, der seriell oder parallel über die Frontplatte ausgelesen werden kann.

Erzeugung der Clock

Die Abtastfrequenz von 300MHz wird in einem zentralen Modul erzeugt und über mehrere Stufen zu den einzelnen FADC-Motherboards verteilt. Als Frequenznormal wird ein über einen weiten Temperaturbereich langzeitstabiler Oszillator der Firma Mirco Networks verwendet, dessen maximaler Phasenjitter <5ps beträgt, die Realisierung der Schaltung erfolgt in ECLinPS-Technologie, wobei zeitkritische Teile konsequent bipolar aufgebaut sind, da nur bei dieser Beschaltungsart ein Jitter von <5ps vom Hersteller garantiert wird. Die Ausgänge wurden mit Hinblick auf die Kosten im Gesamtsystem unipolar ausgelegt, wobei durch Messungen gezeigt werden konnte, dass sich eine unipolare Beschaltung nur unwesentlich auf den Jitter auswirkt (vgl. Kap.4, Messung des Jitters des Clocksignals, Seite 71 ff.)

Synchronisation mit dem Triggersignal und Master Reset

Das am System zur Verfügung gestellte Triggersignal muss zur weiteren Verarbeitung mit dem 300MHz-Signal synchronisiert werden. Diese Phasensynchronisation erfolgt – vereinfacht ausgedrückt- mit einem Flipflop:

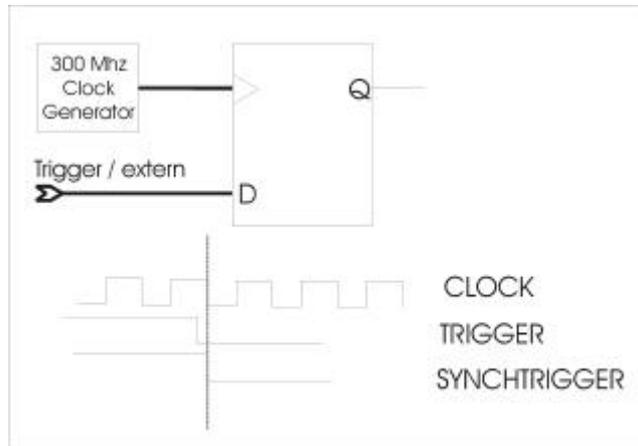


Abb. III.17 : Triggersynchronisation

Das Clocksignal und der dazu synchronisierte Synchtrigger (vgl. Abb.17) werden als unipolare ECL-Signale über Koaxialkabel und kleine Clock- und Triggereinheiten (bestehend aus einem einzigen ECL-Clockverteilerchip) verteilt, welche in der Frontseite der Interfaceboards integriert sind. Folgendes Blockschaltbild erläutert die Clockverteilung :

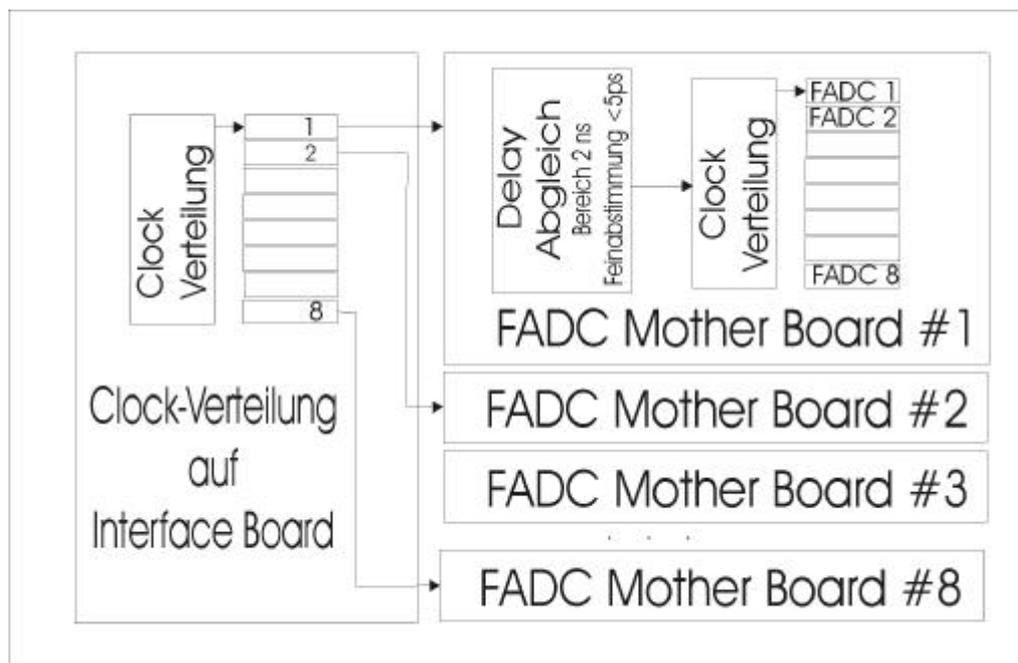


Abb. III.18 : Clockverteilung

Auf dem letzten Teilstück vor dem Eingang des FIFO-Motherboards müssen Clock- und das dazu synchronisierte Triggersignal durch Kabellängen abgeglichen werden (vgl. Abschnitt „Abgleich von Trigger und Clock“ dieses Kapitels, Seite 40). Ein Präzisionsabgleich der Clocksignale mehrerer FIFO-Motherboards untereinander im Endsystem erfolgt durch Lötunkte und ein Potentiometer auf jedem Board.

III.5. Das Delay-Board zur Generierung von boardspez. Look-Up-Tabellen

Die verwendeten Bausteine haben einen Jitter von typischerweise 5ps, jedoch liegt der systematische Fehler höher, bedingt durch eine konstante Verzögerung der einzelnen Gatter. Dieser Fehler wird aus Kostengründen hardwaremäßig nur pro Board bis hin zur letzten Clockverteilungstufe, nicht aber für jeden einzelnen Kanal korrigiert. Bedingt durch die letzte Clockverteilungstufe (vgl. Seite Abb.III.18.Clockverteilung), 2 nachfolgende Gatter und den FADC-Chip selbst tritt ein konstanter, systematischer Offset (Zeitfehler) auf, der in der Größenordnung von mehreren 10ps liegt. Um diesen konstanten Offset zu erfassen und später per Software zu korrigieren, wurde ein spezielles Delayboard (Eichboard) entwickelt und aufgebaut. In folgender Darstellung ist die räumliche Aufteilung des Delay-Boards zu sehen.

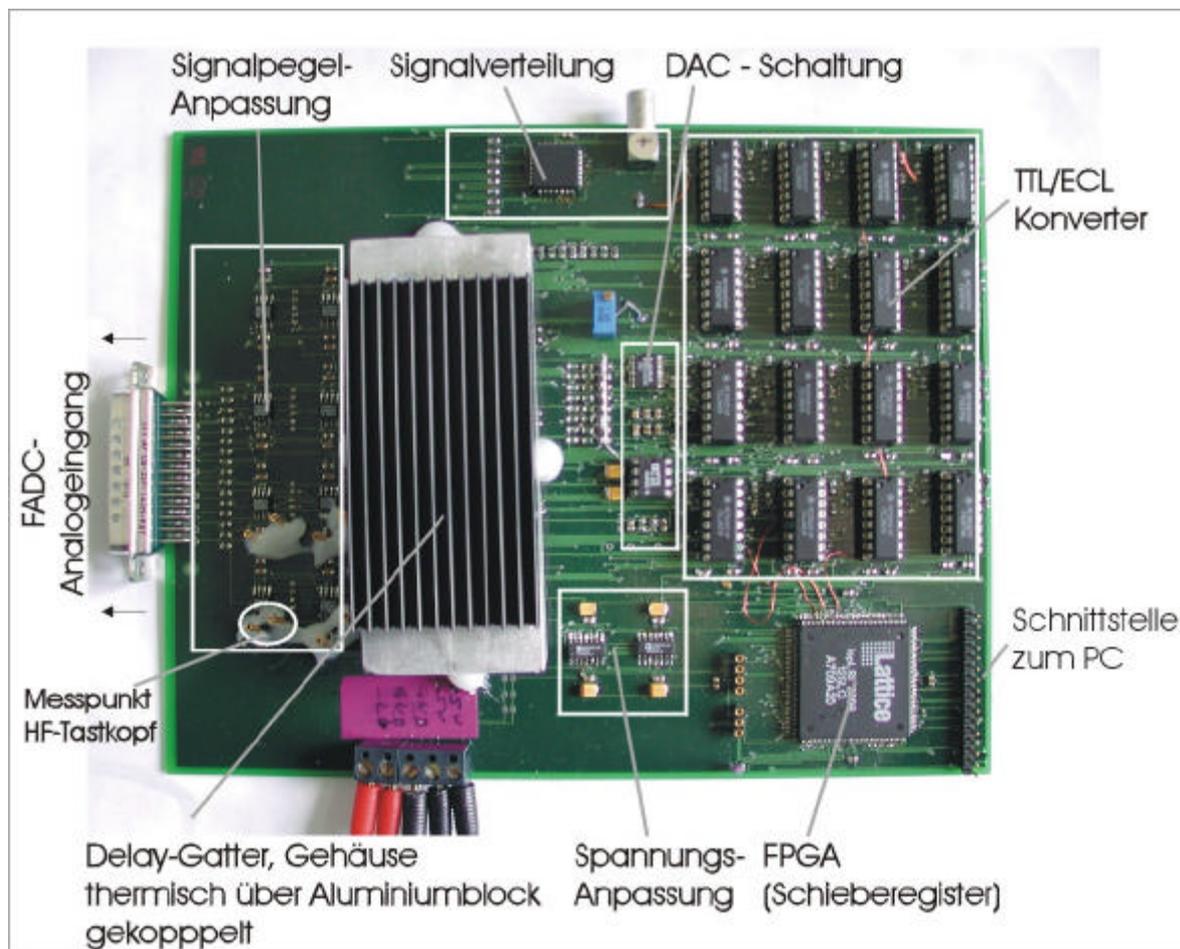


Abb. III.19. : Delay-Board

Auf diesem Delayboard können Pulse über den PC gesteuert bis auf wenige Pikosekunden genau gegeneinander verschoben werden.

Die Funktion des Delay-Boards soll anhand des folgenden Blockschaltbilds erläutert werden :

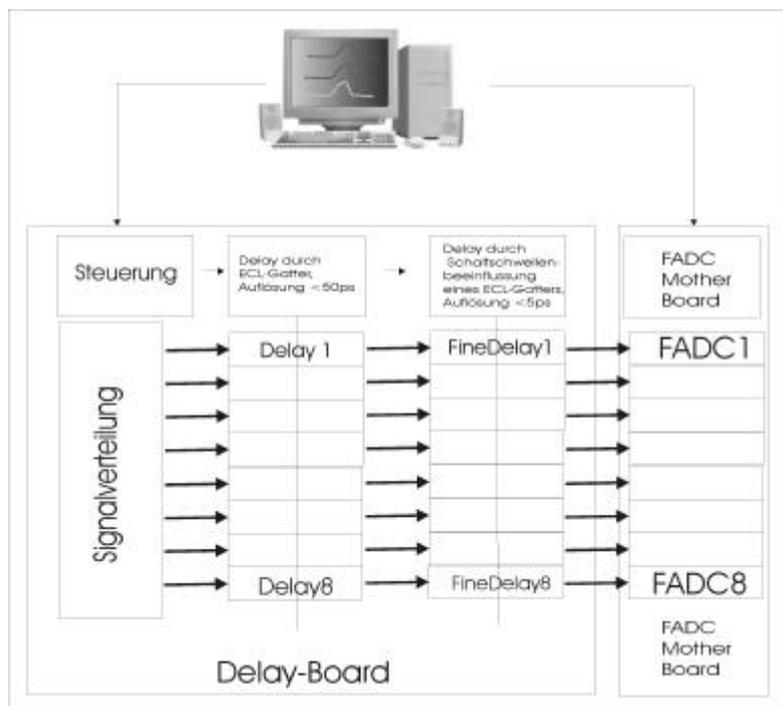


Abb. III.20. : Blockschaltbild des Delayboards

Ein am Eingang anliegendes Signal wird auf insgesamt 8 Kanäle verteilt, wobei jeder dieser 8 Kanäle in einem Bereich von 2ns mit einer Genauigkeit $< 5ps$ verzögert werden kann. Die Grobeinstellung (30ps-Bereich) erfolgt über ECL-Gatter. Eine Feineinstellung bis $< 5ps$ wird durch Anlegen einer über eine Steuerspannung an die Schaltschwelle des letzten Gatters erreicht, welche in einem DAC erzeugt wird. Die Steuerung der Verzögerung erfolgt per PC. Es wird dieselbe DAC-Schaltung wie in (Kap. III.2., Seite 36, DAC zur Generierung der Baseline) verwendet. Die Anpassung der Steuerspannung 0 bis -5V an die DAC-Ausgangsspannung erfolgt durch eine einfache Operationsverstärkerschaltung:

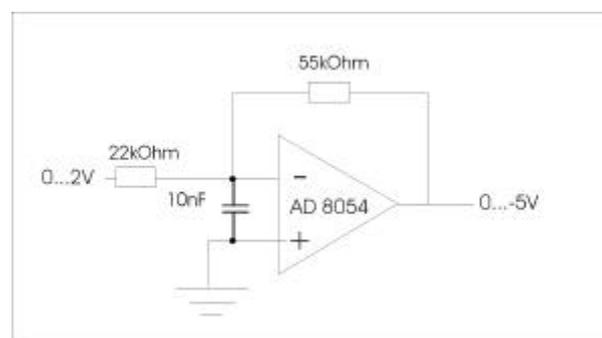


Abb. III.21. : Spannungsanpassung

Mit einem Oszilloskop ausreichend hoher Bandbreite und Abtastfrequenz kann man die einzelnen Verzögerungen so einstellen, dass die Pulse am Ausgang –im Rahmen der Messgenauigkeit und eines Jitters von ca. 5ps- gleichzeitig erscheinen. Man kann nun diese Pulse auf den Eingang des FADC-Motherboards geben und digitalisieren. Da die Pulsform bekannt ist, können die Messwerte mit einer entsprechenden Funktion angefittet werden und der zeitlich konstante Offset aus den Parametern der Fitfunktion ermittelt werden. Genauere Erläuterungen zu dieser Messung und deren Ergebnisse finden sich in Kapitel 4 (Kap. IV.3. ff, ab Seite 71) dieser Arbeit.

III.6. Die Auslese eines einzelnen Boards im Testaufbau

Das in Kapitel III.2. (Seite 35 ff) beschriebene FADC-Motherboard wird in einem eigens dazu aufgebauten Teststand getestet. Der Aufbau ist im folgenden schematisch dargestellt:

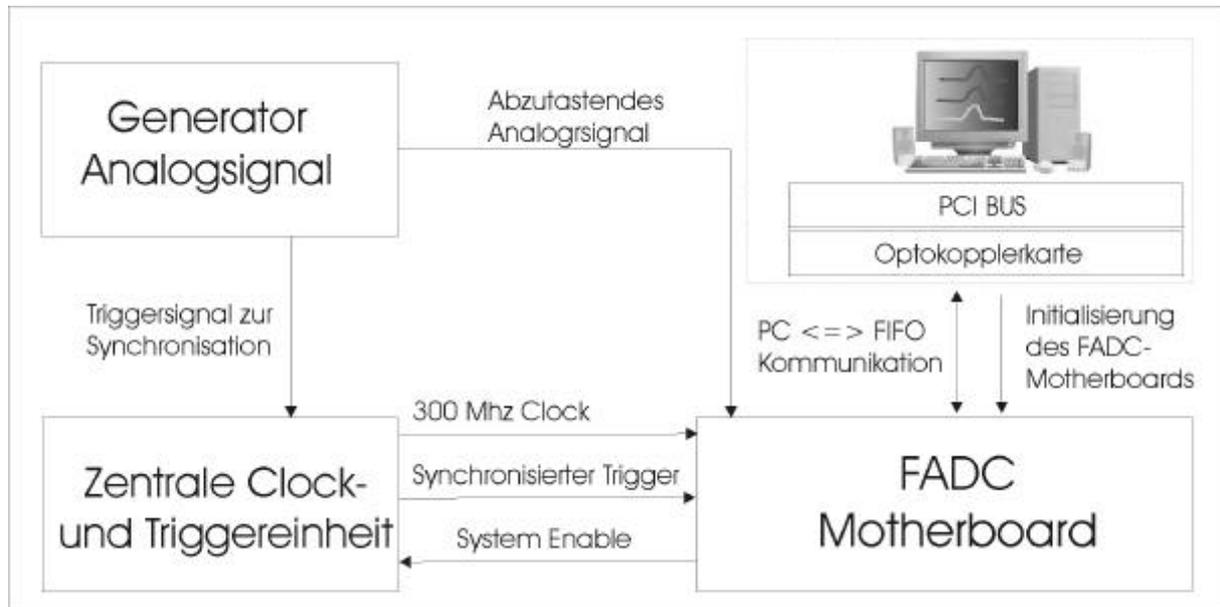


Abb.III.22. : Blockschaltbild des Testaufbaus (einzelnes Board)

Der Teststand beinhaltet Schaltnetzteile zur Spannungsversorgung, die unter Kapitel III.4. (Seite 48 ff) beschriebene zentrale Clock- und Triggereinheit und das zu testende FADC-Motherboard. Als Steuerkarte im PC dient eine einfache, kommerzielle PCI-I/O-Karte der Fa. Meilhaus. Zwischen der PCI-Karte und dem FADC-Board befindet sich eine im Rahmen dieser Arbeit entwickelte Optokopplerkarte, welche durch eine galvanische Trennung von PC und Ausleseelektronik für mehr Störsicherheit sorgt. Da die PCI-Karte nicht genügend I/O-Kanäle zur Verfügung stellt, ist auf dieser Optokopplerkarte ein Multiplexer integriert.

Die verwendete PCI-I/O-Karte stellt 32 Ein- und Ausgänge zur Verfügung. Da ein PCI-Bus typischerweise 32-Bit-Worte überträgt, wäre es wünschenswert gewesen, wie bei der Auslese im Gesamtsystem 4 FADC-Kanäle à 8 Bit zusammenzufassen und pro Taktzyklus zu übertragen. Da aber auch FIFO-Steuersignale an den PC gesendet werden müssen, erfolgt im Testsystem ein Multiplexing des 32 Bit breiten Datenbusses, welcher am Ausgang des FADC-Motherboards zur Verfügung steht. Dieser 32:8 Multiplexer ist in einem FPGA implementiert. Die Anzahl der Ausgänge ist für die Testschaltung ausreichend.

Die Datenaufnahme über die FADC-Module und das Beschreiben der FIFOs geschieht im Normalbetrieb völlig eigenständig, die Ablaufsteuerung ist in einem FPGA auf dem FADC-Motherboard integriert. Zweck der Testschaltung ist es, das FADC-Motherboard zu initialisieren und die am Ausgang der FIFOs anstehenden Daten auszulesen. Da der PC quasi direkt mit den einzelnen FIFOs kommuniziert, wurde das Ausleseprotokoll dem Datenblatt des FIFOs entnommen und im PC-Ausleseprogramm in C implementiert. Vor Beginn der Datenaufnahme muss das System über das PC-Ausleseprogramm initialisiert werden, d.h. die On-Board-Zähler müssen zurückgesetzt werden (FIFO-Master-Reset, Reset der State-Machine für die Ablaufsteuerung, Reset des Triggerzählers) und bestimmte Kenngrößen wie z.B. Baselineeinstellung, die Hauptparameter NSP und OFFS (vgl. Kap.III.2., Seite 27 ff) und ggf. die FIFO-Flagprogrammierung werden über serielle Schieberegister in das System getaktet.

III.7. Die Auslese des Gesamtsystems

Die Auslese des Gesamtsystems erfolgt mit einer PCI-Karte der Fa. Silicon Software. Die verwendete microEnable-Karte ist nicht nur eine einfache I/O-Karte, sondern es befindet sich ein vom Benutzer programmierbarer FPGA und ein Cache auf der Karte. Dies erlaubt eine funktionale Zweiteilung der PCI-Karte: Zum einen wird das Ausleseprotokoll der FIFOs direkt in FPGA implementiert, zum anderen kann der Cache, wenn er ausreichend mit Daten aufgefüllt ist, mit maximaler Frequenz über DMA und sog. PCI-Bursts in den Speicher des PCs ausgelesen werden. Wie bereits in den vorangegangenen Kapiteln beschrieben, wird als Übertragungstechnologie LVDS eingesetzt. Zur Pegelanpassung und zur galvanischen Trennung zwischen Auslese-PC und der Elektronik wurde die microEnable-Karte um ein entsprechendes LVDS-Optokopplerboard erweitert. Folgendes Blockschaltbild soll die Funktionsweise erklären :

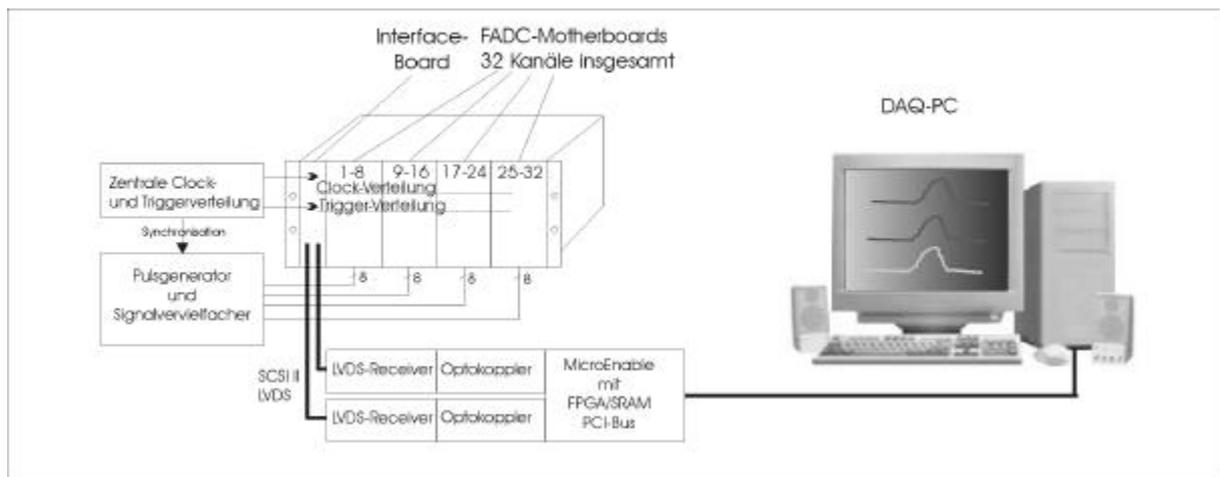


Abb. III.23 : Blockschaltbild 32-Kanal-Prototyp-Auslese

Die LVDS-Optokoppler-Einheit besteht aus 2 identischen Modulen mit je 31 Kanälen (20 Eingänge und 11 Ausgänge), welche am PC-Ausgang an zwei 68-polige SCSI-Kabel angeschlossen und im Normalbetrieb mit einem Interfaceboard Clockverteilung (vgl. Kap.III.3., Abschnitt Interfaceboard, Seite 46 ff) verbunden werden.

Die zentrale Clock- und Triggerverteilung erzeugt eine hochpräzise 300MHz-Clock und ein Triggersignal, welches mit der Clock und einem synchronisierten Signalgenerator in einer festen Phasenbeziehung steht. Die im ECL-Standard übermittelten Trigger- und Clock-Signale werden auf der Interfaceplatine auf der dort frontseitig realisierten Trigger- und Clockverteilung auf die 4 einzelnen FADC-Motherboards verteilt. Das vom Signalgenerator zur Verfügung gestellte Signal wird auf ausgewählte Analogeingänge der 4 FADC-Motherboards geleitet. Die Verteilung der Analogsignale wurde mit dem in Kap. III.5. (Seite 52 ff) dargestellten Delayboard realisiert, somit können mit diesem Aufbau max. jeweils 8 Kanäle gleichzeitig getestet werden.

Zweck des Prototypenaufbau ist es, die Funktionsfähigkeit eines größeren Systems zu demonstrieren. Hierzu muss wie auch bei dem Test eines einzelnen Boards im Testaufbau (vgl. Kap.III.6.) das FADC-Motherboard initialisiert und die am Ausgang der FIFOs anstehenden Daten auszulesen werden. Vor Beginn der Datenaufnahme muss das System über das PC-Ausleseprogramm initialisiert werden, d.h. die On-Board-Zähler müssen zurückgesetzt werden (FIFO-Master-Reset, Reset der Statemachine für die Ablaufsteuerung, Reset des Triggerzählers) und bestimmte Kenngrößen wie z.B. Baselineeinstellung, die Hauptparameter NSP und OFFS (vgl. Kap.III.1, Seite 27 ff) und ggf. die FIFO-Flagprogrammierung werden über serielle Schieberegister in das System getaktet.

Der prinzipielle Unterschied zur unter Kap. III.6. (Seite 55/56) beschriebenen Auslese im Testaufbau besteht darin, dass der PC nicht über Software mit den einzelnen FIFOs kommuniziert, sondern dass das Ausleseprotokoll im FPGA der microEnable-PCI-Karte implementiert ist. Die Ergebnisse der Testmessungen an dem 32-Kanal-Prototypen wurden mit den Einzelplatzmessungen verglichen, die Ergebnisse sind in Kapitel 4 (Seite 74 ff) dieser Arbeit näher erläutert.

Aufgrund der langen Übertragungswege vom jeweils auszulesenden FIFO bis zur PCI-microEnable-Karte ist eine Kompensation von Kabellaufzeiten erforderlich. Bereits bei einem 10m langen Kabel liegt die Kabellaufzeit in der Größenordnung von $1/\text{Clockfrequenz}$. Zur Laufzeitkompensation wird einfach der Auslesetakt (CLK) am Eingang des FIFO-Boards zurück zur PCI-Karte (CLK') gesendet. Alle von der PCI-Karte ausgehenden Signale werden dann zu CLK, alle vom FIFO ausgehenden Signale zu CLK' synchronisiert sein. Die Länge des Signalwegs spielt somit keine Rolle mehr, es müssen lediglich gleichlange Paare von SCSI-Kabeln sichergestellt werden, so dass Daten und Clock die gleiche Verzögerung erfahren.

Im Blockschaltbild der 32-Kanal-Prototyp-Auslese nicht eingezeichnet sind zusätzliche LVDS-Treiber, die eine Umkehr der Datenrichtung erlauben. Sie sind als alternative Bestückung vorgesehen, um ein „Gegenstück“ zu erhalten, mit dem die Schaltung ohne FADCs getestet werden kann. Bei einem solchen Testaufbau werden 2 PCs über die o.g. SCSI-Kabel verbunden, ein PC mit microEnable-Karte emuliert die FADC-Boards (im wesentlichen das FIFO-Ausleseprotokoll) und der andere PC agiert als Auslese-PC. Damit lassen sich kontrolliert unterschiedliche zeitliche Eventverteilungen (z.B. kurze Bursts mit hoher Eventdichte) und die Reaktion des Auslese-PCs darauf testen.

IV. Messungen am FADC - System

Die Komponenten des Systems wurden einzeln und in kleineren Unter-Systemen auf Ihre Funktions- und Leistungsfähigkeit überprüft. In diesem Kapitel werden die jeweiligen Testmessungen erläutert.

IV.1. Testmessungen am einzelnen FADC – Modul

Übersicht über den Test der FADC-Module

Die einzelnen FADC-Module werden bei der Herstellung eingestellt und getestet. Diese Testmessungen erfolgten unabhängig von dem im Rahmen dieser Arbeit aufgebauten FADC-System und wurden in einem nahezu idealen Umfeld durchgeführt:

- Die Wärmeabführung war vergleichsweise sehr gut (freistehendes Modul mit eigenem Lüfter).
- Die Zuleitung der Analogsignale konnte fast vollständig per Koaxialkabel realisiert werden (im Gegensatz von bis zu 20 cm Leiterbahn auf dem FADC-Motherboard).
- EMV-Störungen verursachende Elektronik konnte aufgrund der geringen Packungsdichte im Vergleich zum FADC-Motherboard gut räumlich getrennt werden.
- Weniger Probleme der Masseführung (Potentialunterschiede, Erdschleifen) als bei einem großen und verteilten System.

Diese standardmäßig durchgeführten Testmessungen umfassen einen einfachen Funktionstest und die Bestimmung des Auflösungsvermögens ENOB (**E**ffective **N**umber **o**f **B**its), die sowohl für den quasi-statischen Fall als auch für den dynamischen Fall ermittelt wurde. Weiterhin wurde die maximale Abtastfrequenz der einzelnen FADC-Module ermittelt. Die Standardtests hinsichtlich des Auflösungsverhaltens der einzelnen FADC-Module sind nicht eigentliches Thema dieser Arbeit und sollen hier nur grob erläutert werden. Nähere Ausführungen hierzu finden sich in [VOLKOV] und [STIEHLERDIPLOM]

Messung der maximalen Abtastfrequenz

Die maximale Abtastfrequenz der FADC-Module ist begrenzt. Der eigentliche FADC-Chip ist für Abtastfrequenzen von bis zu 500 MHz ausgelegt und auch die für die Clockverteilung verwendete ECLinPS-Technologie der FADC-Module würde Frequenzen in dieser Größenordnung zulassen. Der FADC-Chip gibt jeweils 2·8Bit parallel mit halber Abtastfrequenz aus, dementsprechend muss die nachfolgende Digitalelektronik nur mit halber Frequenz arbeiten. Die Bauteile, die die maximale Betriebsfrequenz limitieren sind zum einen der FPGA (laut Timing-Simulation maximale Frequenz von 163 MHz) sowie die statischen RAM-Bausteine, die nur als 8ns-Sortierung erhältlich sind. Da die Nominalfrequenz der FADCs im System auf 300 MHz festgelegt ist, ist man mit dem FPGA (Limitierung auf $2 \cdot 163 \text{ MHz} = 326 \text{ MHz}$) auf der sicheren Seite.

Probleme könnten vielmehr seitens der statischen RAMs auftauchen. RAM-Zugriffszeiten von 8 ns als Worst-Case-Scenario würden eine Limitierung auf 250 MHz bedeuten. Da dies ein unakzeptabler Wert ist, muss während der Herstellung der FADC-Module eine Vorselektion von RAMs mit Zugriffszeiten von $t_{\text{zugriff}} < 6.5 \text{ ns}$ erfolgen. RAMs mit Zugriffszeiten von 6.5 ns würden eine maximale Frequenz von 307,7 MHz bedeuten. Da die RAMs der eigentliche limitierende Faktor im System sind, nimmt man zum Test der maximalen Abtastfrequenz der FADC-Module einen beliebigen Kurvenverlauf (z.B. Sinusfunktion) über den gesamten dynamischen Bereich auf und bestimmt die Anzahl der Bitausfälle (d.h. unkorrekte Digitalisierungswerte aufgrund fehlerhafter RAM-Schreibvorgänge) bei der jeweiligen Abtastfrequenz.

Um fundierte statistische Daten zu erhalten, wurden jeweils 10^5 Messpunkte aufgenommen. Es ergibt sich folgender Kurvenverlauf :

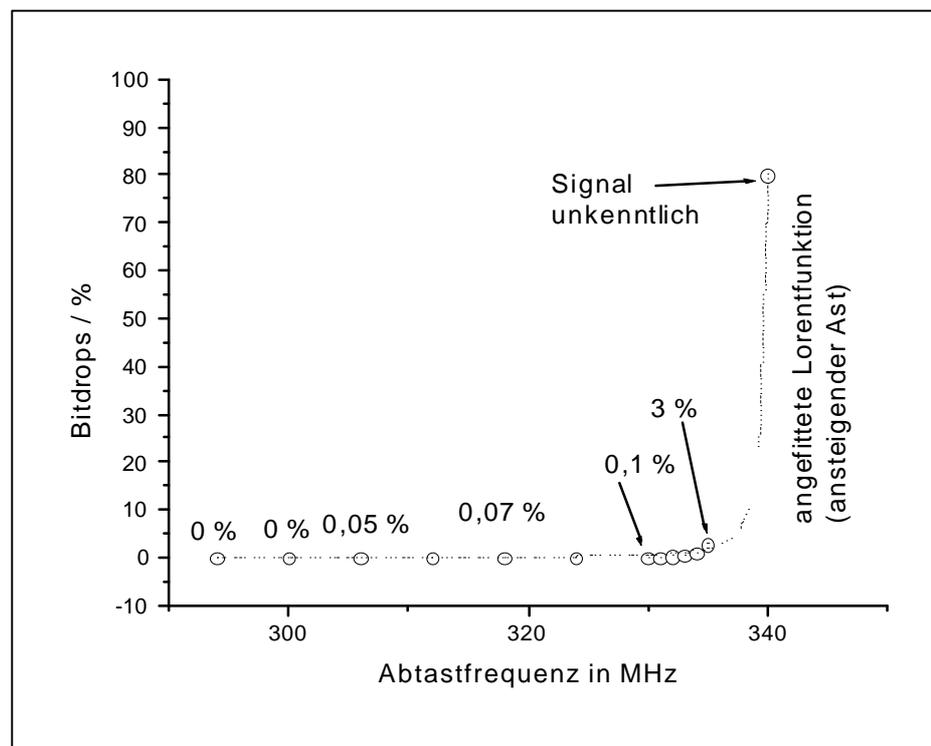


Abb.IV.1. :Abtastfrequenz vs. Bitdrops

Bei dem hier ausgemessenen FADC-Modul wurden bis zu einer Frequenz von ca. 305 MHz keine Bitausfälle registriert, im Bereich von 305MHz-330MHz steigt die Zahl der Bitausfälle nahezu linear von 0,05% auf 0,1% an. Im Bereich von 330MHz-340MHz nimmt die Zahl der Bitdrops schlagartig auf über 50% zu, so dass das Signal unkenntlich wird. Dieser für das Verhalten typische Kurvenverlauf kann gut mit einer Lorentzfunktion angenähert werden.

Statisches und dynamisches Auflösungsverhalten

Messungen zum statischen (Rampentest) und dynamischen (Sinustest) Auflösungsverhalten wurden an einem Modul in einem voll bestückten Board durchgeführt und die Ergebnisse mit den Ergebnissen der Standardtests verglichen, die in der Einführung dieses Kapitels bereits erwähnt wurden. Erfahrungsgemäß können aufgrund der dort aufgeführten fast idealen Rahmenbedingungen die bei diesen Tests gemessenen Werte nur bedingt in einem System mit einer Vielzahl von FADC-Modulen erzielt werden.

Vergleich im quasi-statischen Fall (Rampentest)

Für den Fall, dass das analoge Eingangssignal quasi-statisch ist, sich also auf einem bestimmten Gleichspannungsniveau befindet, ist das Auflösungsvermögen der FADC-Module besonders gut. Das Auflösungsvermögen wird im Rahmen der Herstellung der FADCs gemessen, indem eine Spannung digitalisiert wird, deren Genauigkeit wesentlich höher ist wie die Genauigkeit des FADCs. Eine durch einen 12 Bit genauen DAC erzeugte Gleichspannung, die als Baseline-Signal für die FADCs dient, wird schrittweise erhöht (daher die Bezeichnung „Rampentest“). Hierbei werden bei jeder DAC-Stufe 1000 Digitalisierungen durchgeführt und die Einträge in den jeweiligen FADC-Stufen gezählt. Der Übergang von einer FADC-Stufe zur nächsten ist keine reine Stufenfunktion, vielmehr sind die Einträge im Übergangsbereich statistisch verteilt. Man kann diese Verteilungsfunktion als eine Faltung von einer durch Rauschen bzw. statistischen Störeinflüssen bedingten Distribution mit einer idealen Stufenfunktion ansehen:

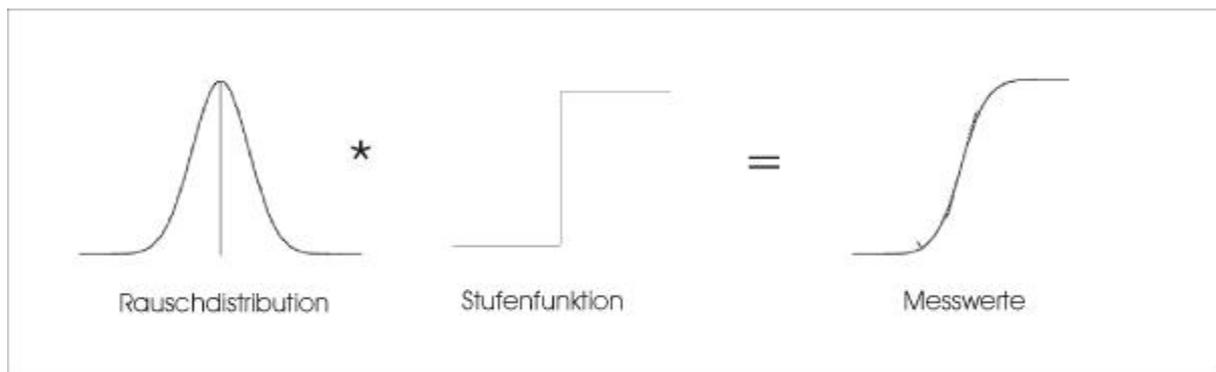


Abb.IV.2. : Faltung einer Rauschdistribution mit Stufenfunktion

Die Breite (z.B. der Wert bei FWHM Full Width Half Maximum) der Distribution ist ein Maß für das Auflösungsvermögen des getesteten FADCs. Die Breite der Distribution muss aus den digitalisierten Messwerten entfaltet werden. Per Definition entspricht die Faltung der Rauschdistribution mit der Stufenfunktion dem Integral:

$$MESSWERTE = \int_{-\infty}^{+\infty} RAUSCHDISTRIBUTION(x') \cdot HEAVISIDE(x - x') \cdot dx'$$

Aufgrund der einfachen Beschaffenheit der Stufenfunktion ($\text{HEAVISIDE}(x-x')=1$) entspricht die Entfaltung der Rauschdistribution aus den Messwerten einer Differentiation der Messwertverteilungsfunktion. Durch Anfitzen einer Gaußverteilung an die so gewonnene Rauschdistribution und Bestimmung deren Breite (FWHM) kann man die Auflösung des gemessenen Systems bestimmen. Dieses Verfahren ist näher in [STIEHLERDIPLOM] erläutert.

Für die FADCs wurde auf diese Weise im Teststand für einzelne FADCs [vgl. VOLKOV] für den quasi-statischen Fall ein Auflösungsvermögen von $>7,8$ ENOB festgestellt. Da im Messaufbau des Gesamtsystems aufgrund von PC-internen Interrupt/I-O-Konflikten kein geeigneter DAC (d.h. Auflösung $\gg 8\text{Bit}$) zur Verfügung stand, wurde eine grobe Abschätzung durchgeführt, indem man die zur Baseline-Einstellung ohnehin zur Verfügung stehende Spannung des 8-Bit-On-Board-DAC als Messgrundlage verwendet hat. Die Genauigkeit dieses DAC wird vom Hersteller auf 7,75 Bit angegeben, was 1,2 FADC Stufen entspricht. Wie auch bei der Vergleichsmessung im Einzelteststand wurde nun auch mit dem 8-Bit-DAC der DAC-Wert jeweils um eine Stufe erhöht und die entsprechenden Einträge in den FADC-Stufen ausgewertet. Da für jede DAC-Stufe nie mehr als in zwei FADC-Stufen Ereignisse gezählt werden konnten, beträgt die effektive Auflösung im System für den quasi-statischen Fall mindestens 7 Bit.

Vergleich im dynamischen Fall (Sinustest)

Das dynamische Auflösungsvermögen der FADCs wird bestimmt, indem man eine mit einem Funktionsgenerator erzeugte sinusförmige Spannung digitalisiert. Die Abweichung der Messwerte von den Funktionswerten einer angefitteten Sinusfunktion ist ein Maß für das dynamische Auflösungsvermögen des betreffenden FADCs. Die effektive Anzahl der Bits eines N Bit breiten AD-Wandlers errechnet sich aus

$$ENOB = N - \log_2 \sqrt{\left(\frac{1}{M} \sum (y_n - y'_n)^2 \right)} \cdot \sqrt{12}$$

wobei y_n die Messwerte und y'_n die angefittete Sinusfunktion darstellen.

In diesem Ausdruck ist

$$\sqrt{\left(\frac{1}{M} \sum (y_n - y_{n'})^2\right)}$$

die Standardabweichung der Messwerte und $12^{1/2}$ der ideale Quantisierungsfehler (2. Statistisches Moment einer Boxfunktion). $N=8$ ist die Anzahl der Digitalausgänge des FADCs und M die Anzahl der Messpunkte.

Die FADCs werden mit einer Frequenz von 300MHz getaktet. Nach dem Nyquist-Kriterium kann aus den aufgenommenen Daten eine Sinusfunktion von max. 150 MHz korrekt rekonstruiert werden.

Im Rahmen der Testmessungen wurden mit den FADCs in einem voll bestückten Board sinusförmige Spannungen von 0,1 MHz – 150 MHz digitalisiert und die errechnete Auflösung mit den Messdaten der in der Einleitung dieses Kapitels erwähnten Standardtests verglichen. Die Messergebnisse sind in folgendem Graphen dargestellt :

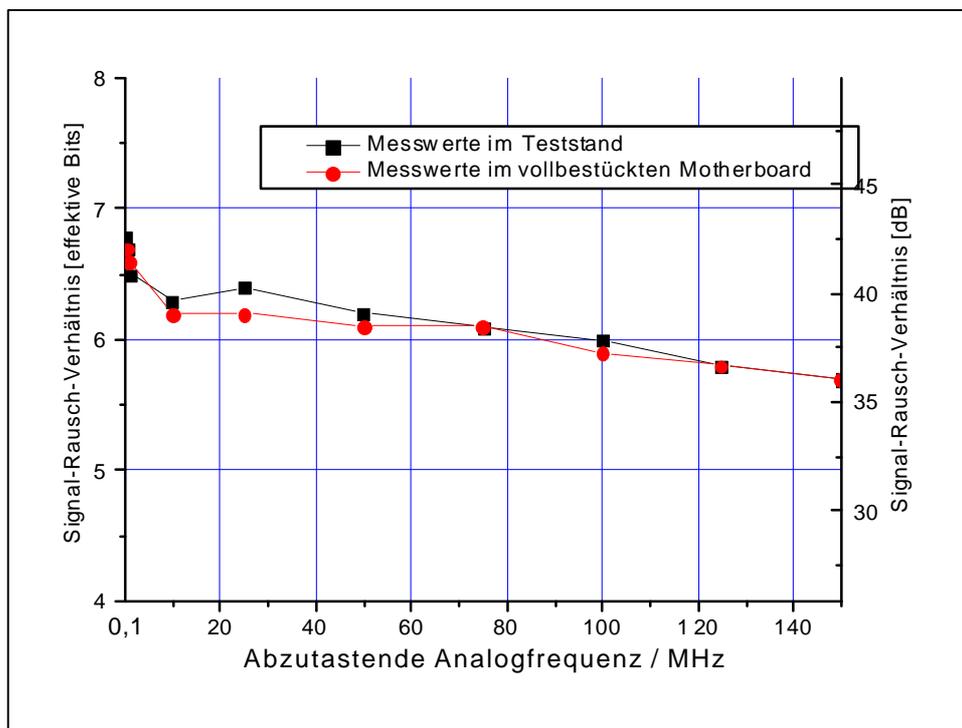


Abb. IV.3. : Vergleich des Dynamiktests bei Teststand und Motherboard

Die erzielten Messergebnisse der FADC-Module im bestückten FADC-Motherboard weichen nur unwesentlich von den Werten ab, die bei den Standardmessungen erzielt wurden. Dies spricht mit Hinblick auf die bereits erwähnten schlechteren Randbedingungen eines großen Systems im Vergleich zu einem Einzeltestaufbau (vgl. Einleitung dieses Kapitels) für die Qualität des Aufbaus. Es sei in diesem Zusammenhang nochmals anzumerken, dass die obige Messung zum Signal-Rauschverhältnis eigentlich eine Messung des Verhältnisses des Signals zur Summe aller Störeinflüsse ist (vgl. Kap.2, Seite 11 ff, Auflösung eines FADC-Systems). Die Bezeichnung Signal-Rauschverhältnis ist hier irreführend, da andere Störeinflüsse (EMV-Störungen, Oberwellen, etc.) dominant sind.

IV.2. Testmessungen an der Signalverteilung

Problematik bei der Signalverteilung

Bei der Auslese des „Magic-Teleskops“ wird aufgrund der langen Signalübertragungswege mit bipolaren LVDS-Signalen gearbeitet, die über impedanzangepasste Leitungen über ein System von Backplanes (impedanzangepasste Platinen), Interfaceboards und 68-polige SCSI-Standardkabel über Optokoppler in den Auslese-PC geleitet werden. Die Systemfrequenz im Magic-Auslesesystem wird 20 MHz betragen, es ist evtl. mit einer Erhöhung auf bis zu 40 MHz zu rechnen. Begrenzendes Element sind hier die Optokoppler, die bis max. 40 MHz spezifiziert sind. Die Signalwege können aufgrund der hohen Frequenzen und somit relativ schnellen Signalanstiegszeiten nicht mehr als ideale Leiter angesehen werden, sondern müssen als Hochfrequenzleitungen betrachtet werden. Genauere Ausführungen hierzu finden sich in Kapitel 2 dieser Arbeit. Im folgenden sind die Testmessungen beschrieben, die mit der Backplane und an den verwendeten Signalkabeln (SCSI-II-Standard) durchgeführt wurden. Es wurden insbesondere die Signaldämpfung und die Phasentreue der einzelnen Signalaare gegeneinander untersucht.

Messungen an der Backplane

Die Backplane wurde eigens für dieses Projekt an der Uni-GH-Siegen entwickelt. Sie ist in Microstrip-Technik aufgebaut und besteht aus 4 Lagen. Die äußeren Lagen sind signalführend, die inneren Lagen sind als Masseflächen ausgeführt. Beim Layout wurde darauf geachtet, paarweise zusammengehörende Leitungen möglichst dicht zusammenzulegen, damit Störungen sich möglichst sowohl auf das Signal als auch auf das invertierte Signal auswirken und vom Receiver eliminiert werden können (Common-Noise-Rejection). Weiterhin wurde darauf geachtet, dass die Signalwege „elektrisch symmetrisch“ verlaufen, d.h. wenn z.B. das invertierte Signal eine Durchkontaktierung durchlaufen muss, dann sollte das nichtinvertierende Signal auch eine Durchkontaktierung im entsprechenden Leiterbahnsegment haben, damit die Auswirkungen durch Reflexion oder Dämpfung auf die Signalform für das invertierte und nichtinvertierte Signal identisch sind und eliminiert werden können. Genauere Betrachtungen zu dieser Thematik finden sich in Kapitel 2 (Seite 15 ff) dieser Arbeit.

Im folgenden ist der Messaufbau dargestellt, der zur Messung verwendet wird:

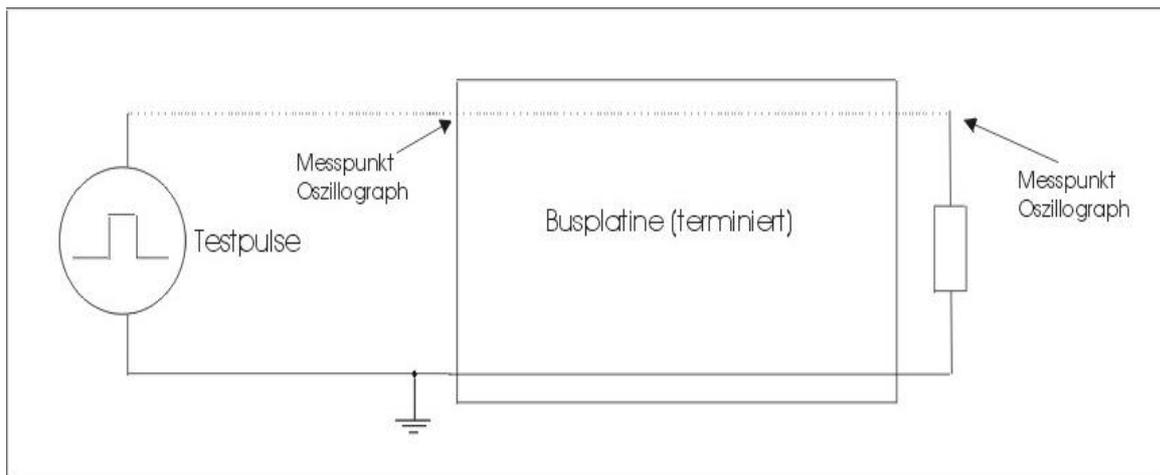


Abb.IV.4. : Aufbau zu den Messungen an der Backplane

Die Signale werden an einem Ende der Backplane eingespeist, die Messpunkte liegen am anderen Ende der Backplane. Die Frequenz des eingespeisten Signals wird nun schrittweise von 0,1MHz – 50 MHz erhöht und die jeweilige Signaldämpfung aufgenommen. Die Ergebnisse sind in folgendem Diagramm dargestellt :

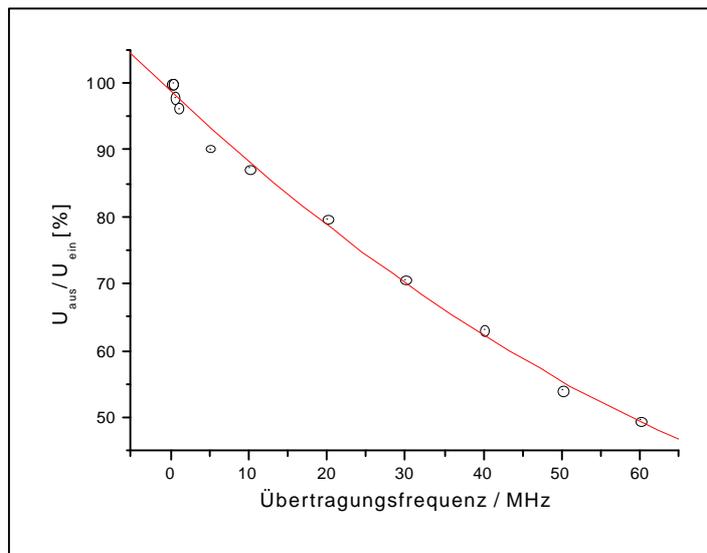


Abb. IV.5. : Messergebnisse Signaldämpfung der Backplane

Die hohe Dämpfung, die den Signalhub bei höheren Frequenzen auf über 50 % des Anfangswertes begrenzen, ist dennoch vollkommen ausreichend für eine LVDS-Übertragungsstrecke. Laut Spezifikation reicht ein Eingangshub von 100mV am Eingang eines differenziellen LVDS-Receiver für einen fehlerfreien Betrieb aus, die typische Ausgangsspannung beträgt (je nach Hersteller) über 300 mV. Aufgrund der Common-Noise-Rejection wirken sich Störeinflüsse von außen trotz des geringen Spannungshubs des

gedämpften LVDS-Signals nicht auf die Funktion der Schaltung aus. Für andere Signalwege auf der Backplane ergeben sich im Rahmen der Messgenauigkeit identische Messwerte. Weiterhin konnten für unterschiedliche Signalwege über den gesamten Frequenzbereich keine nennenswerten Phasenunterschiede ($\ll 1\text{ns}$) festgestellt werden, so dass ein Betrieb der LVDS-Backplane bis über den 50 MHz-Bereich hinaus möglich wäre.

Messungen am SCSI-Kabel

Analog zum Messaufbau bei den Messungen zur Backplane werden Signale an einem Ende des 1.5m langen zu untersuchenden SCSI-Kabels eingespeist, die Messpunkte befinden sich bei dem Abschlusswiderstand am anderen Ende des Kabels. Die Frequenz des eingespeisten Signals wird nun schrittweise von 0,1MHz bis über 50 MHz erhöht und die jeweilige Signaldämpfung aufgenommen. Die Ergebnisse sind in folgendem Diagramm dargestellt :

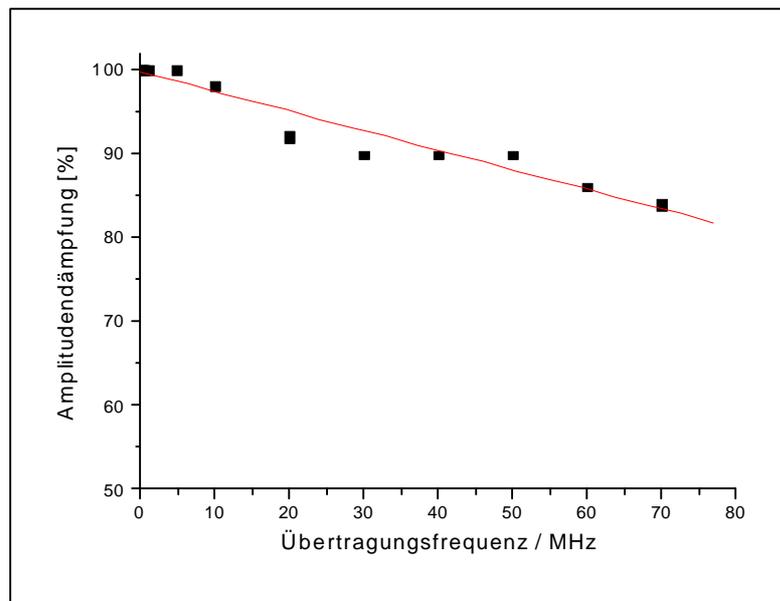


Abb. IV.6. : Dämpfung des Signalübertragungskabels

Bei Signalübertragungswegen mit mehr als einem Kanal kann man die Kabeldämpfung nicht als einziges Qualitätskriterium heranziehen. Da innerhalb des Kabels mehrere Signale parallel laufen, müssen auch gewisse Mindestanforderungen an die Phasengleichheit gestellt werden. Aus diesem Grund wurde das (frequenzabhängige) Phasenverhalten von drei typischen Kabellagen (Innen, Mitte, Außen) untersucht:

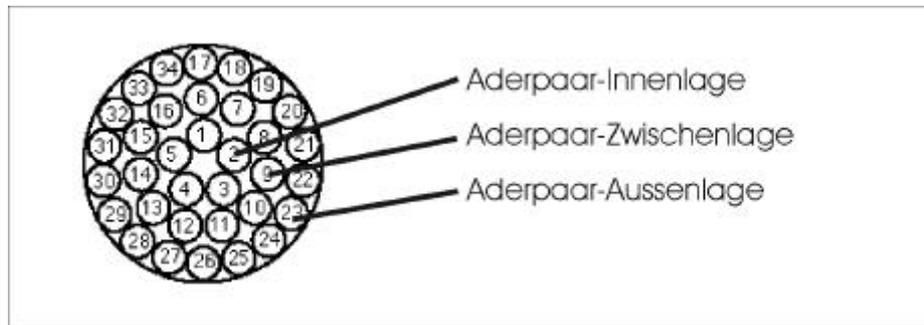


Abb.IV.7. : Lage der Aderpaare [PARALAN]

Typische Herstellerangaben für Unterschiede in der Signalverzögerung (Skew) bei Twisted-Pair-Leitungen liegen im Bereich 60ps bis über 500 ps pro Meter [CABLE], für das untersuchte SCSI-Kabel lagen allerdings keine diesbezüglichen Angaben vor. Im Aufbau muss man von Kabellängen von bis zu 10m ausgehen, so dass mit Hinblick auf die Elektronik ein maximaler Skew von 0,15ns/m einen sinnvollen Grenzwert darstellt. Bei der Untersuchung des vorliegenden SCSI-Kabels musste man leider feststellen, dass ab einer Frequenz von ca. 35 MHz die Phasenunterschiede vergleichbarer Signalwege in der Innenlage und Außenlage so groß werden, dass (unter Zugrundelegung des Grenzwertes von 0,15ns/m) kein ordnungsgemäßes Funktionieren der Schaltung mehr gewährleistet werden kann. Da im Nominalbetrieb die maximale Frequenz bei 20 MHz liegt, ist diese Grenzfrequenz jedoch unkritisch. Es sei noch angemerkt, dass die Messung nur mit geerdeter Ummantelung durchgeführt wurde, die jeweils unbenutzten Aderpaare wurden nicht beschaltet. Im späteren Betrieb hingegen hat man sowohl quasistatische Signale und einige masseführende Aderpaare, was zu einem „konstanterem“ Wellenwiderstand und evtl. leicht besseren Ergebnissen führen könnte.

Bewertung der Ergebnisse

Die erzielten Ergebnisse entsprechen noch voll und ganz den Erfordernissen im Gesamtsystem, das mit nominal 20MHz betrieben werden soll. Eine Steigerung dieser Auslesefrequenz bis auf ca. 35MHz ist möglich, sofern der Auslese-PC den höheren Datenfluss verarbeiten kann. Begrenzende Elemente sind aufgrund der stark unterschiedlichen Phasenverhaltens für verschiedene, vergleichbare Signalwege das SCSI-Kabel (Limitierung auf 35MHz) sowie die Optokoppler, die zur Zeit nur für 20MHz ausgelegt sind, aber durch pinkompatible Austauschtypen mit einer max. Durchsatzrate von 40MHz ersetzt werden können.

IV.3. Erfassung des Jitters und Offsets an einem 8-Kanal- FADC-Motherboard

Einleitung

Mit Hilfe des in Kapitel 3 dieser Arbeit beschriebenen Delayboards können Pulse über den PC gesteuert bis auf wenige Pikosekunden genau gegeneinander verschoben werden. Nach einer Eichmessung mit einem Digital-Oszillographen hoher Bandbreiter und ausreichender Abtastfrequenz kann man mit diesem Delayboard den Abtast-Jitter der einzelnen FADC-Kanäle auf einem FADC-Motherboard sowie das zeitliche Verhalten der FADC-Kanäle untereinander (Offset bedingt durch Fertigungstoleranzen der verwendeten Bausteine) bestimmen. Aufgrund eines falsch gewählten FPGA (I/O-Routing aufgrund hoher Packungsdichte nicht mehr beliebig möglich) konnten leider nur jeweils 4 Kanäle miteinander verglichen werden.

Messung des Jitters des abzutastenden Signals

Das abzutastende Analogsignal wird –wie bereits erläutert- auf insgesamt 8 Kanäle verteilt, wobei die Signale in jedem Teilzweig beliebig verzögert werden. Diese 8 Signale durchlaufen mehrere Halbleiter, was jeweils zu einem bestimmten Jitter führt. Man kann diesen Jitter erfassen, indem man den zeitlichen Abstands zweier verzögerter Pulse mit Hilfe eines Digitaloszillographen ausreichend hoher Bandbreite und Abtastfrequenz histogrammiert. Die Anbringung der HF-Tastköpfe erfolgt an speziellen Messpunkten auf dem Delayboard (vgl. Kap III.5, Abb. Delay-Board, Seite 52), bei dem verwendeten Oszillographen ist eine entsprechende Histogrammierungsfunktion fest implementiert.

An jedem dieser Messpunkte ergab sich bei der Histogrammierung ein maximaler Jitter von weniger als 10ps, nach Anfitzen einer Gaußverteilung ergibt sich eine Halbwertsbreite der Verteilung von deutlich weniger als $\sigma < 5\text{ps}$.

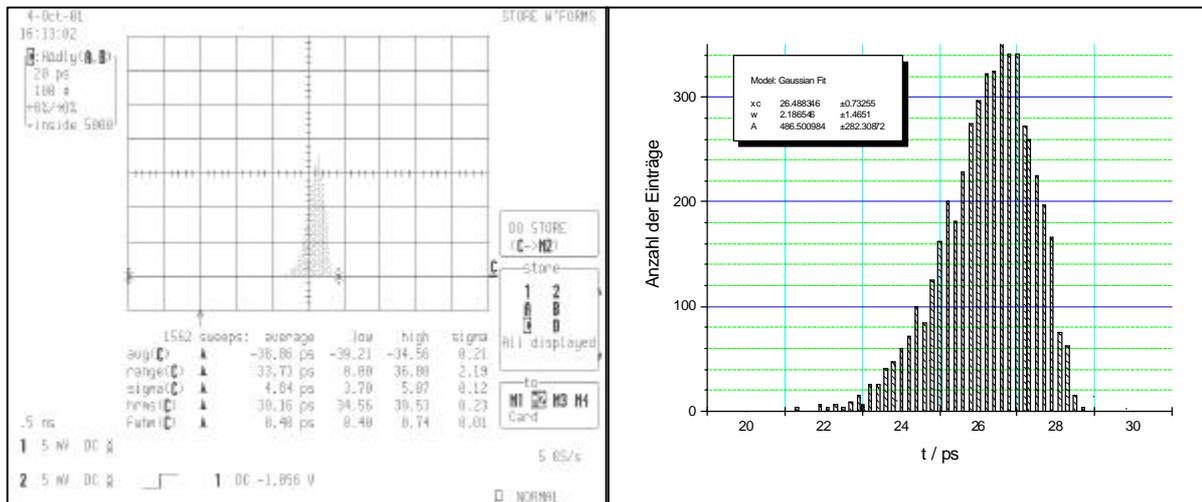


Abb. IV.8. : Jittermessung

Da sich Temperaturunterschiede auf dem Delayboard in den 8 einzelnen Signalzweigen stark auf das Timingverhalten auswirken, wird ein Temperatenausgleich geschaffen, indem ein ca. 3 cm starker Quader aus Aluminium (vgl. Kap III.5, Abb. Delay-Board, Seite 52) mit Wärmeleitpaste auf allen Halbleiterbauelementen befestigt wird, so dass diese thermisch miteinander verbunden werden. Wenn man die thermische Kopplung (Aluminiumblock) der Kanäle untereinander entfernt, dann können sich die Halbleiter der einzelnen Kanäle unterschiedlich erwärmen (ungleiches dT/dt). Man stellt für diesem Fall bei Langzeitmessungen (mehrere 10 s bis mehrere Minuten) fest, dass die Verteilung auf dem histogrammierenden Oszilloskop nach links bzw. rechts „wandert“. Dieses Verhalten ist nicht auf einen Jitter, sondern auf temperaturabhängige Laufzeitunterschiede zurückzuführen. Diese temperaturabhängigen Laufzeitunterschiede sind auch bei (zeitlich) unterschiedlich stark gekühlten FADCs zu erwarten, entsprechende Messungen hierzu stehen noch aus. Als „Offset“ ist dieser Fehler nur schwer zu erfassen, da im Gesamtsystem nicht alle FADC-Kanäle einzeln temperaturüberwacht sind und eine thermische Kopplung aufgrund der räumlichen Dimensionen praktisch möglich ist. Für den Benutzer der Schaltung erscheint es dann so, als ob sich durch das zeitliche Verschieben der Histogramm-Mitte der Jitter von 5 ps auf rund 20 ps vergrößern würde.

Messung des Jitters des Clocksignals

Die Abschätzung des Jitters des Clocksignals erfolgt analog zur Jittermessung des Analogsignals durch einfache Histogrammierung des Flankenabstands des 300MHz-Clocksignals. Messungen mit der in dem verwendeten Oszillographen implementierten Histogrammierungsfunktion führen wie bei der analogen Messung des Analogsignals zu einem Histogramm mit einer Halbwertsbreite von $\sigma < 5\text{ps}$ und ist praktisch mit dem Histogramm der Jittermessung des Analogsignals (vgl. voriger Abschnitt) identisch. Mit Hinblick auf die Herstellerangaben der verwendeten ECL-Bausteine ist es für den vorliegenden unipolaren Fall ein bemerkenswert gutes Ergebnis, dass der Jitterwert praktisch mit den Angaben für rein differenziellen Betrieb übereinstimmt.

Nullabgleich und Bestimmung des zeitlichen Versatzes unterschiedlicher Kanäle (Offset)

Man kann nun mit Hilfe eines Oszillographen die einzelnen Kanäle des Delayboards so verzögern, dass am Analogeingang des FADC-Motherboards die Signale im Rahmen der Messgenauigkeit und des Jitters alle Analogsignale zeitgleich anliegen. Wenn man nun die eingehenden Analogsignale mit den FADCs digitalisiert und die Abtastpunkte mit einer entsprechenden Funktion fittet, dann ist der Abstand der einzelnen Fitkurven untereinander ein Maß für den Offset des jeweiligen Kanals. Der gemessene Offset liegt im Bereich 0-30ps und ist im Rahmen des ermittelten Jitters konstant. Durch Vertauschen zweier FADC-Module im Motherboard kann gezeigt werden, dass der Offset größtenteils durch die FADC-Module selbst und nicht durch das Motherboard bestimmt wird. Dies ist darauf zurückzuführen, dass auf dem Motherboard nur ein Halbleiterbaustein für alle 8 Kanäle gleichzeitig unterschiedliche Laufzeiten verursachen kann, auf einem FADC-Modul steuern hingegen (für jeden einzelnen Kanal getrennt) sowohl der FADC-Chip selbst als auch ein Halbleiterbauelement zu Laufzeitschwankungen bei.

Einstellung eines beliebigen Offset-Wertes und Langzeitmessung

Bei 4 Kanälen wurde per PC jeweils ein beliebiger, konstanter Offset eingestellt und der Abstand der digitalisierten Pulse nach erfolgtem Fitten des Kurvenverlaufs histogrammiert. In folgender Abbildung ist der gefittete Kurvenverlauf (steigende Flanke) an die Messwerte eines beliebigen Events zu sehen.

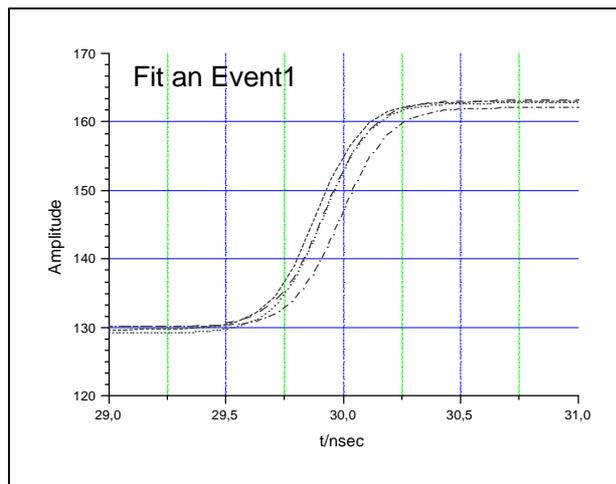


Abb.IV.9. : Fit eines Flankenanstiegs

Diese Messung wurde über einen längeren Zeitraum mit sehr geringer Wiederholfrequenz der Pulse durchgeführt. Die folgenden Ergebnisse erhält man bei einer Wiederholfrequenz von 500mHz über einen Zeitraum von ca. 30 Minuten.

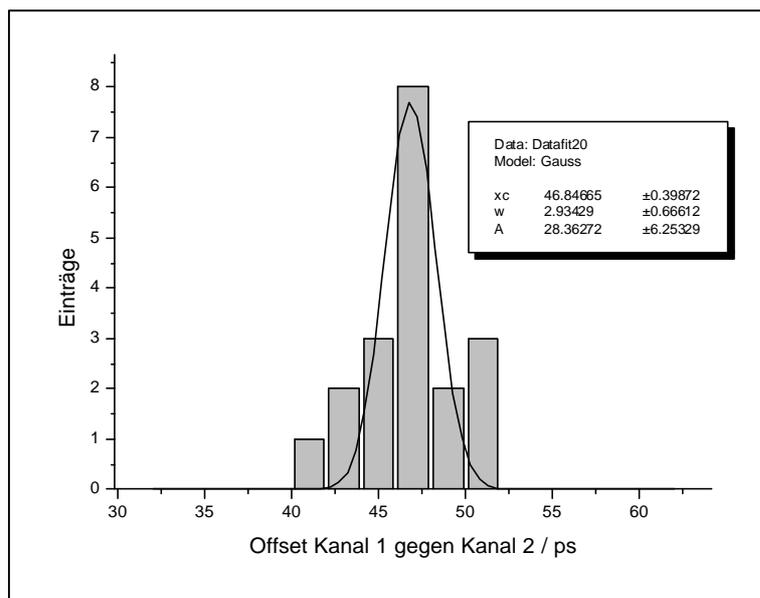


Abb. IV.10. : Jitter 2 Kanäle an einem FADC-Motherboard

Der Jitter zweier Kanäle untereinander liegt (auch bei der Langzeitmessung) unter 5ps.

IV.4. Erfassung des Jitters an einem 32-Kanal-Prototypen

Messungen zur Erfassung des Jitters (entsprechend den Messungen aus dem vorigen Kapitel IV.3.) wurden auch an einem 32-Kanal-Prototypen durchgeführt. Bei diesen Messungen wurde prinzipiell derselbe Aufbau verwendet, wie er schon im vorigen Kapitel beschrieben wurde: Per PC wird am Delay-Board ein beliebiger, konstanter Offset eingestellt und der Abstand der digitalisierten Pulse nach erfolgtem Fitten des Kurvenverlaufs histogrammiert. Über eine Adapterplatine können jeweils 4 der insgesamt 32 Kanäle zur vergleichenden Messung ausgewählt werden. Diese Messung wurde über einen längeren Zeitraum mit sehr geringer Wiederholfrequenz der Pulse durchgeführt. Die folgenden Ergebnisse erhält man bei einer Wiederholfrequenz von 1kHz über einen Zeitraum von ca. 15 Minuten:

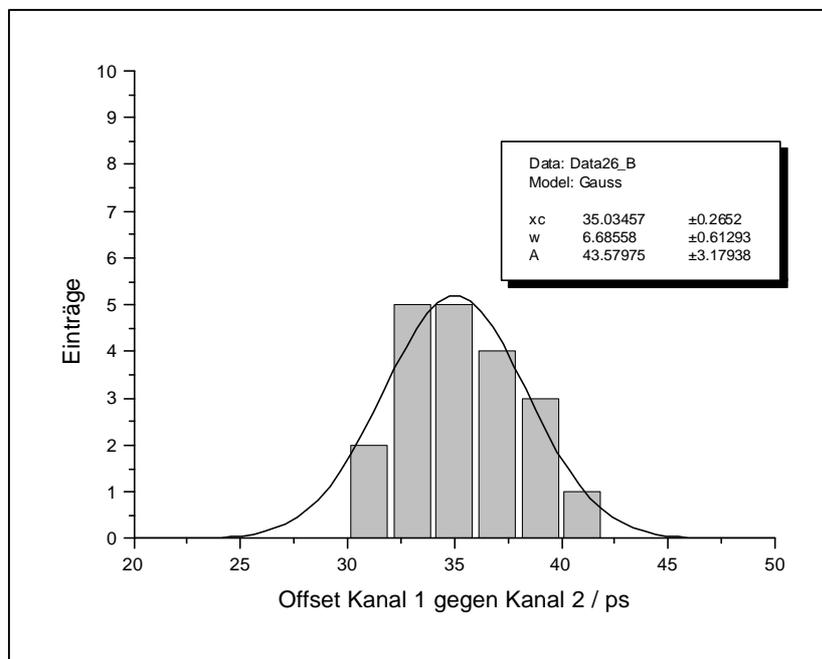


Abb. IV.11. : Jitter bei 32-Kanal-Prototypen (FADCs auf 2 versch. Motherboards)

Der Jitter bei Kanälen, die sich auf unterschiedlichen FADC-Motherboards im System befinden, ist um einige Pikosekunden größer als bei dem Jitter von Modulen auf demselben FADC-Motherboard, liegt aber deutlich unterhalb von 10 ps.

Bewertung der Ergebnisse

Die Messungen zum Offset und Jitter wurden sowohl an einem einzelnen FADC-Board als auch an einem größeren Prototypensystem (32 Kanäle, Signalverteilung auf 4 verschiedene FADC-Motherboards) durchgeführt. Entsprechende Messungen am Gesamtsystem (Verteilung auf unterschiedliche Racks) sollten zu einem späteren Zeitpunkt noch durchgeführt werden. Ein evtl. vorhandener großer Offset ist unproblematisch, da dieser auf jedem einzelnen FADC-Motherboard abgeglichen werden kann (vgl. Kap.III.3., Abschnitt Abgleich von Trigger und Clock). Zur noch ausstehenden Jitter-Messung in einem größeren System sei angemerkt, dass bei den bisherigen Messungen die Signale des Delayboards bei der Einzelmessung direkt (Steckverbindung ohne Kabel) an den Eingangsstecker des FADC-Motherboards und bei der Messung am Prototypensystem über sehr kurze (und impedanzkontrollierte) Signalleitungen angekoppelt wurden. Eine Verteilung dieser Signale auf mehrere FADC-Motherboards in unterschiedlichen Racks ist mit unverhältnismässig hohem Aufwand verbunden, wenn man die Ergebnisse aus den Vergleichsmessungen mit einem Jitter von $t_{\text{jitter}} \ll 10$ ps auch nur näherungsweise reproduzieren wollte.

Der Offset zwischen unterschiedlichen Kanälen ist stark abhängig von Temperaturdifferenzen, die bei einzelnen FADCs untereinander bestehen, da unterschiedliche Halbleitertemperaturen zu unterschiedlichen Durchschaltzeiten führen. Der kanalspezifische Jitter ändert sich im Rahmen der im System möglichen Temperaturschwankungen hingegen verhältnismässig gering. Bei den Darstellungen zur Langzeitmessung sind temperaturabhängige Schwankungen des Offsets und der eigentliche kanalspezifische Jitter nicht trennbar. Die Messung zeigt jedoch, dass sich weder der Jitter noch der Offset der Kanäle untereinander selbst über längere Zeiträume nicht verändert. Dieses bemerkenswert gute Ergebnis ist auf die gute gleichmäßige Kühlung der einzelnen FADC-Module im Teststand zurückzuführen. Da bei den Jittermessungen am Delayboard gezeigt werden konnte, dass ohne gleichmäßige Kühlung sich die zeitliche Auflösung durch Temperaturschwankungen von rund 5 ps auf bis zu 25 ps verschlechtert, ist im Gesamtsystem unbedingt auf eine gute Kühlung zu achten, die nicht ausschließlich die Maximaltemperatur begrenzen soll, sondern auch für gleichmäßige Temperaturverhältnisse sorgen muss. Sind gleichmäßige Temperaturverhältnisse nicht gegeben, so kann nur noch bei Kurzzeitmessungen im Sekundenbereich von einer Zeitauflösung im Bereich von 5ps ausgegangen werden.

V. Zusammenfassung

Im Rahmen dieser Arbeit wurde ein Konzept zur Auslese des Cherenkov-Teleskops MAGIC erstellt, die einzelnen Komponenten wurden entwickelt und sowohl einzeln als auch im Gesamtsystem getestet. Die grundlegenden Anforderungen dieses Projekts sind eine Gesamtzahl von 640 8-Bit FADC-Kanälen, die mit einer Abtastrate von 300 MHz arbeiten.

Als Grundlage für das System diente ein an der Universität Siegen entwickeltes FADC-Modul, welches für die Befürfnisse des Systems leicht modifiziert wurde. Die Modifikationen umfassen Layoutverbesserungen im Analogteil zur Verminderung von Störeinflüssen, die Implementierung eines Emitterfolgers zur Vermeidung von Fehlzählungen des Adresszählers und die Adaption des moduleigenen FPGAs an die Ausleseelektronik.

Kern des Systems ist ein FADC-Motherboard, auf dem 8 dieser FADC-Module Platz finden. Ein Teststand zum Test einzelner FADC-Boards mit optogalvanischer Trennung und Auslese über eine Standard-PCI-I/O-Karte (AD 7200, deutscher Vertrieb über Fa. Meilhaus) wurde entwickelt und an einem vollbestückten Board Messungen zum Auflösungsvermögen durchgeführt. Das gemessene Auflösungsvermögen entspricht der bei einer Einzelplatzmessung erzielten Genauigkeit. Bei Messungen zum Aperturjitter konnte gezeigt werden, dass dieser im Bereich von wenigen Pikosekunden liegt und mit den verwendeten ECLInps-Bausteinen aufgrund deren Spezifikationen nicht mehr zu verbessern ist.

Die Auslese von insgesamt 640 Kanälen in einen einzigen PC über ein eigenentwickeltes, LVDS-basiertes System bestehend aus Backplanes, Interfaceboards und Standard-SCSI-2-Kabeln wurde erfolgreich an einem Prototypensystem demonstriert. Die hierzu erforderliche MicroEnable-PCI-Karte mit FPGA-basierter Auslese und Opto-LVDS-Schnittstelle wurde von der Drittfirma Silicon Software entwickelt nach meinen Vorgaben an das Auslesesystem angepasst. Die Funktionsfähigkeit wurde an einem 32-Kanal-Prototypen demonstriert.

Es konnte gezeigt werden, dass das Konzept und das entwickelte Gesamtsystem den Anforderungen des MAGIC-Projekts entspricht.

Quellennachweis

- [CABLE] Hess, Goldie, Handbuch der Fa. National Semiconductors, „A practical guide to Cable selection“, 1993
- [FADC1] Zander, „Analog-Digital-Wandler in der Praxis“, Markt&Technik-Verlag, 1983, ISBN 3-922120-24-5
- [GRUPEN] Grupen, Claus, „Teilchendetektoren“, 1993, BI-Wissenschaftsverlag, ISBN 3-411-16571-5
- [LOCHNER] Lochner, Jim, „Air Cherenkov Detectors“ (Internet-Publikation), 2001, <http://imagine.gsfc.nasa.gov/>
- [LVDS1] Handbuch der Fa. National Semiconductors, „LVDS Owner’s Guide“, 1997
- [LVDS2] Application Note der Fa. Fairchild Semiconductors, „AN-5017, LVDS Fundamentals“, 2000
- [MAGIC1] González, J.C., „The Energy Resolution of the MAGIC Telescope“, 1999, interne Veröffentlichung des Max-Planck-Instituts für Physik in München
- [MAGIC2] Cortina, Lorenz, Mirzoyan, „The FADC Readout of the MAGIC Telescope“, 2000, interne Veröffentlichung des Max-Planck-Instituts für Physik in München
- [MAGIC3] Lorenz, Weißbach, Rose, Bond, Karle, Tran „Fast analogue signal transmission for an air Cerenkov photomultiplier camera using optical fibers“, 1999, interne Veröffentlichung des Max-Planck-Instituts für Physik in München
- [MAGICPROP] Proposal des MAGIC-Projekts „The Magic Telescope, A design study of a 17m Cerenkov telescope for Gamma-Astronomy above 10GeV“, 1998
- [MECL-PS] Handbuch der Fa. Motorola, „ECL in PS Data“, 1991
- [PARALAN] Online-Datenblatt der Fa. Paralan, <http://www.paralan.de/de/index.html>
- [RAUSCHEN] Müller, „Rauschen (Halbleiterelektronik Bd. 15)“, Springer-Verlag, 1989, ISBN 3540511458
- [SIGNAL1] Zinke, Otto, Brunswig, Heinrich, „Elektronik und Signalverarbeitung“, Springer-Verlag, 1999, ISBN 3-540-64728-7
- [SIGNAL2] Lüke, „Signalübertragung“, Springer-Verlag, ISBN 3540651977
- [STIEHLERDIPLOM] Stiehler, „Entwicklung und Test einer universell einsetzbaren computergesteuerten Ausleseelektronik für 100MHz-Parallel-ADC-Hybridmodule zur Anwendung in der Proteinkristallographie“, Diplomarbeit 1996, Universität Siegen
- [VOLKOV] Volkov, Universität Siegen , Dissertation in Vorbereitung
- [WADELL] Wadell, „Transmission Line Design Handbook“, Artech House Publishers, 1991, ISBN 0890064369

Abstract

The subject of this thesis is the development of a readout system for the camera of the MAGIC telescope. MAGIC is an acronym for Major Atmosphere Gamma Imaging Cherenkov-Telescope.

Imaging Atmosphere Cherenkov-Telescopes (IACTs) are ground-based telescopes for the detection of very high energy particles, in particular gamma rays. Such cosmic particles are indirectly detected by the Cherenkov light which is produced by particles of a shower induced by primary cosmic in the high atmosphere. The light pulses arriving at the telescope are extremely short and can only be registered by digitizing with a high sampling frequency. For the MAGIC experiment a parallel sampling of all 640 camera pixel output signals with 300 MHz FADCs is foreseen. The design and testing has been carried out particularly with regard to enable a relatively high data flux of up to 80 MByte/s and to facilitate a highly synchronous clock distribution with a precision in the order of a few picoseconds.

Within the scope of the present work, a first 32 channel subsystem containing all components of the final system has been developed and tested successfully. This prototype system can be considered as a nucleus for the 640 channel system, which will be implemented at the MAGIC experiment site throughout the year 2002.

Kurzfassung

Thema dieser Arbeit ist die Entwicklung eines Auslesesystems für die Kamera des MAGIC-Teleskops. MAGIC ist eine Abkürzung für Major Atmosphere Gamma Imaging Cherenkov-Telescope.

Bildgebende atmosphärische Cherenkov-Teleskope (englisch IACTs) sind erdgebundene Teleskope zur Detektierung von hochenergetischen Teilchen, insbesondere von Gammastrahlung. Diese kosmischen Teilchen werden indirekt über Cherenkov-Licht detektiert, welches durch Teilchenshower primärer Teilchen im oberen Bereich der Atmosphäre erzeugt wird. Die am Teleskop eintreffenden Lichtpulse sind von extrem kurzer Dauer und können nur durch sehr schnell abtastende A-D-Wandler registriert werden. Beim MAGIC-Experiment ist eine parallele Abtastung aller 640 Kamerapixel-Ausgangssignale mit 300 MHz FADCs vorgesehen. Bei Entwicklung und Test wurde besonderer Wert darauf gelegt, dass ein sehr hoher Datenfluss bis 80 MByte/s und eine synchrone Clockverteilung mit einer Genauigkeit im Bereich weniger Pikosekunden ermöglicht wird.

Im Rahmen der vorliegenden Arbeit wurde ein 32-Kanal-Teilsystem, das alle Komponenten des endgültigen Aufbaus beinhaltet, aufgebaut und erfolgreich getestet. Dieses Prototypensystem kann als Kern der 640-Kanal-Auslese angesehen werden, welches am Bestimmungsstandort des Teleskops im Laufe des Jahres 2002 aufgebaut wird.

Anhang A – Einführung in den FPGA-Logikentwurf

Die hohen Anforderungen an die Steuerlogik legen die Verwendung von hochintegrierten FPGAs nahe. Die Baureihen ispLSI 2000 und ispLSI 5000 des Herstellers Lattice erfüllen problemlos die Forderungen nach einer Auslesetaktfrequenz von rund 40 MHz und gewährleisten durch deren hohe Integrationsdichte eine platzsparende Bauweise der Ausleseelektronik. Im folgenden soll weniger auf den internen Aufbau dieser Bausteine eingegangen werden (hierzu sei auf die entsprechenden Datenblätter verwiesen), vielmehr soll kurz der Logikentwurf mit dem Programmpaket Synario dargestellt werden.

Man unterscheidet beim Logikentwurf unterschiedliche Entwurfsebenen, beginnend bei hardwaremäßigen logischen Verknüpfungen im FPGA-Chip selbst bis hin zur einer Funktionseingabe, die sich an der Hochsprache Abel/HDL und an Blockschaltbildern orientiert. Für den Entwickler sind die unteren Ebenen von untergeordneter Bedeutung, da die gesamte Schaltung sowohl funktional als auch unter Berücksichtigung von chipinternen Gatterlaufzeiten simuliert werden kann. Die Entwicklung der als FPGA implementierten Elektronik soll im folgenden anhand von Teilschaltungen des Hauptsteuerungs-FPGA ispLSI5384 erläutert werden. Der komplette Logikentwurf findet sich im Anhang B dieser Arbeit.

In einer Übersicht sind beim verwendeten Programmpaket Synario alle oberen Ebenen zusammengestellt :

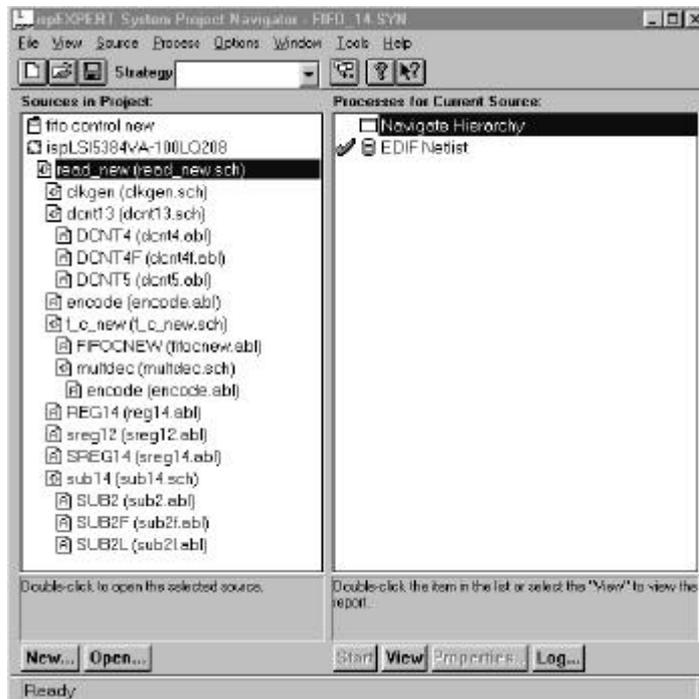


Abb.A1 : Übersicht Navigationshierarchie bei Synario

Die im linken Teil des Fensters angezeigten Dateinamen mit der Endung *.abl bezeichnen Dateien im Abel-Format, einer einfachen HDL-Programmiersprache (Hardware Description Language). Dateien mit der Endung *.sch bezeichnen Schematics, in welchen einfache Logikgatter und komplexere Blockschaltbilder zu einer funktionalen Einheit zusammenschaltet werden können. Die oberste Schematics-Datei legt zudem die Pinbelegung des FPGAs fest.

Im folgenden Screenshot ist ein Ausschnitt einer Seite des obersten Schematics dargestellt :

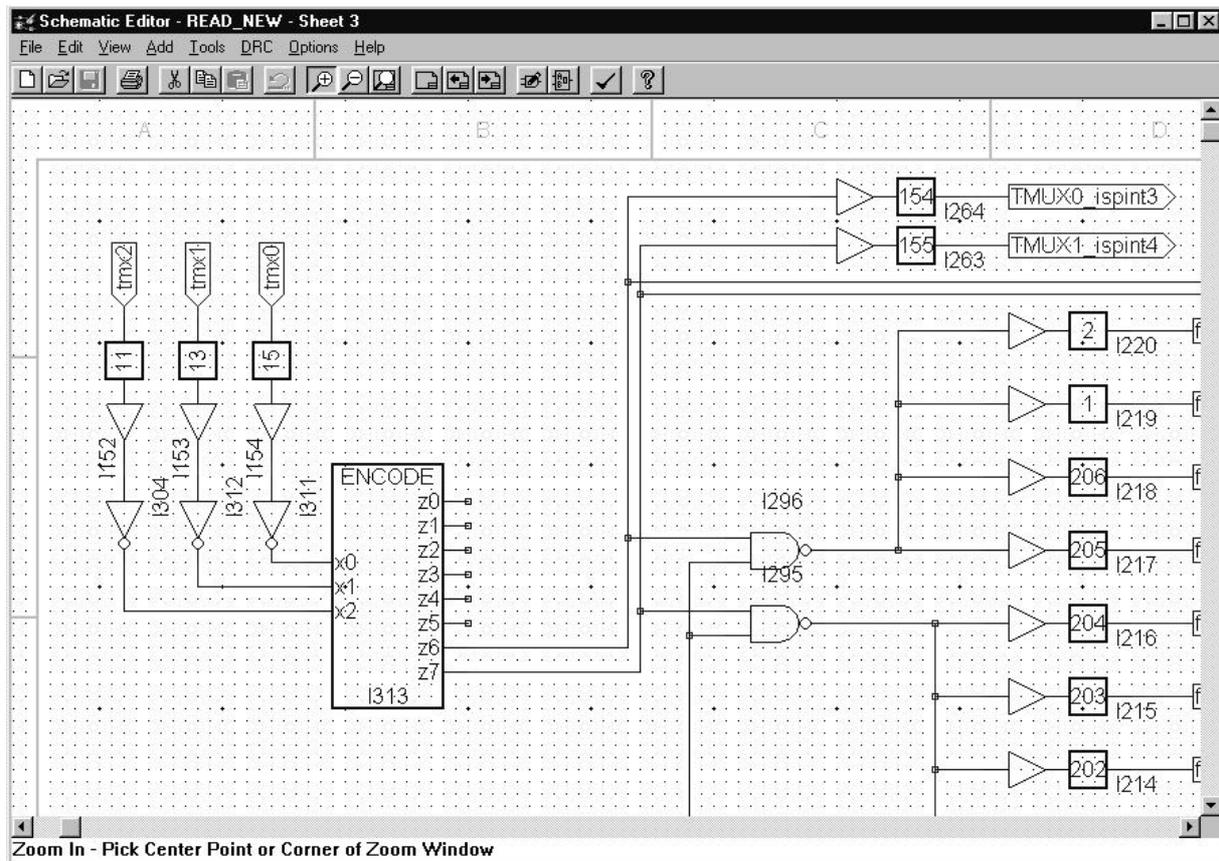


Abb. A2: Auszug aus Schematic READ_NEW.sch

Schematics stellen die einfachste und intuitivste Eingabemöglichkeit dar. In jedem Schematic können entweder Bauteile aus vorhandenen Bibliotheken eingefügt werden oder neue Symbole (Blockschaltbilder) definiert werden, die dann auf einer tieferliegenden Ebene z.B. als Abel-Quell-Datei definiert sein muss. Beispielsweise ist der in der Darstellung als Blockschaltbild angezeigte 3-zu-8-Encoder ebenfalls in der Übersichtsdarstellung (vgl. Abb. A1) enthalten.

Die Abel-Datei encode.abl beschreibt die Funktionen dieses Encoders :

```

MODIFIE encode
TITLE '3-to-8 Encoder'

*inputs
  z0, z1, z2 pin;
  z = [z2, z0];

*outputs
  z0, z1, z2, z3, z4, z5, z6, z7 pin istype 'com';

truth_table (z -> [z0, z1, z2, z3, z4, z5, z6, z7])
  0 -> [1, 0, 0, 0, 0, 0, 0, 0];
  1 -> [0, 1, 0, 0, 0, 0, 0, 0];
  2 -> [0, 0, 1, 0, 0, 0, 0, 0];
  3 -> [0, 0, 0, 1, 0, 0, 0, 0];
  4 -> [0, 0, 0, 0, 1, 0, 0, 0];
  5 -> [0, 0, 0, 0, 0, 1, 0, 0];
  6 -> [0, 0, 0, 0, 0, 0, 1, 0];
  7 -> [0, 0, 0, 0, 0, 0, 0, 1];

END encode

```

Abb. A3 :Beispiel einer Abel-Datei

Im Kopf der Abel-Datei werden der Name des Moduls sowie ein Titel eingegeben. Anschließend werden Ein- und Ausgangssignale definiert, bei komplizierteren Entwürfen kann man an dieser Stelle auch Konstanten für die Simulation definieren. Bei der Definition können wie im obigen Beispiel mehrere Eingänge (oder Ausgänge) zu Vektoren zusammengefasst werden. Bei Ausgängen muss die Art des Signals angegeben werden. So bezeichnet in obigem Beispiel `pin istype com` einen einfachen kombinatorischen Ausgang. Im Hauptteil des Programms (EQUATIONS oder TRUTH_TABLE) werden die Funktionsgleichungen oder wie im obigen Beispiel eine Wahrheitstabelle zur eigentlichen funktionalen Beschreibung des Moduls eingegeben. Bei komplexeren Entwürfen schliessen sich an dieser Stelle die Testvektoren der Schaltung an. Das Modul wird mit der Zeile `END` abgeschlossen.

Nachdem alle Module erstellt, einzeln compiliert und deren einwandfreie Funktion verifiziert wurde, kann man in dem Übersichtsfenster eine chipspezifische Jedec-Datei erstellen, welche über die serielle Schnittstelle in den FPGA übertragen werden kann. Das Erzeugen einer chipspezifischen Datei wird als „Fitten“ bezeichnet.

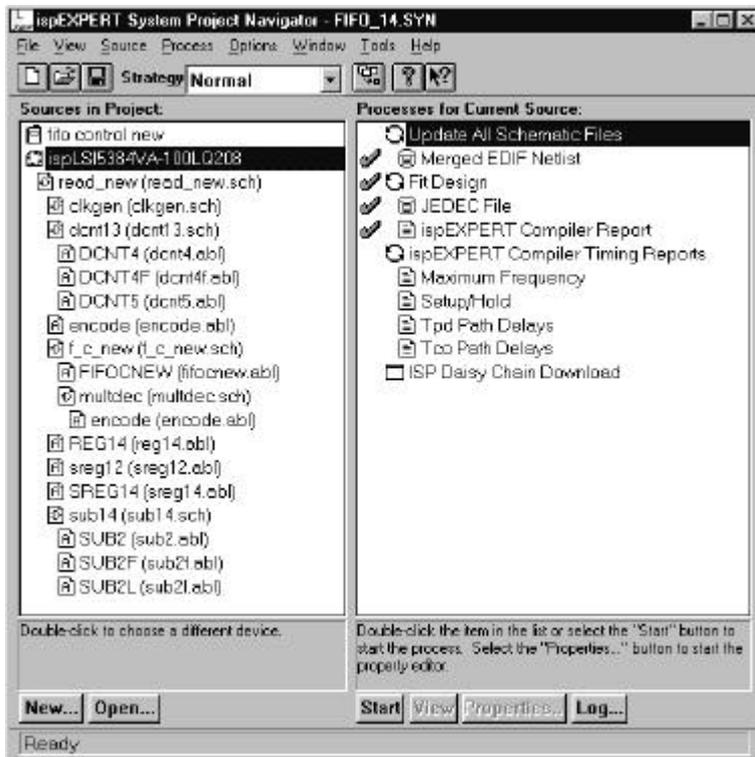


Abb. A4 : Übersicht Netzliste erzeugen/Fitten/Simulation

Im linken Fenster wird unterhalb des Projektnamens der FPGA-Typ und dessen Gehäusebauform festgelegt. Im rechten Fenster kann man dann (nacheinander) eine Netzliste erzeugen, diese Netzliste („Merged EDIF Netlist“) an den gewählten FPGA-Typ anpassen („Fit Design“) und erhält neben der eigentlichen Jedec-Datei automatisch mehrere aus den Testvektoren erzeugte Zwischendateien, mit denen eine einfache funktionale und auch eine chipspezifische Timing-Simulation durchgeführt werden kann. Der Menüpunkt „Maximum Frequency“ ruft ein Programm auf, das unter Berücksichtigung chipinterner Gatterlaufzeiten die maximale Frequenz des Entwurfs ermittelt und die begrenzenden langsamsten Pfade aufzeigt.

Anhang B –Schematics und Abel-Dateien des FPGA-Logikentwurfs

FADC-Motherboard : Schematics für Lattice ispLSI 2096V

Lattice ispLSI 2096V – Flag-Multiplexing und quasi-statische Leitungen..... B2

FADC-Motherboard : Schematics für Lattice ispLSI 5384V

Lattice ispLSI 5384V – Haupt-Schematic Ebene 1 – Blatt 1/3..... B3

Lattice ispLSI 5384V – Haupt-Schematic Ebene 1 – Blatt 2/3..... B4

Lattice ispLSI 5384V – Haupt-Schematic Ebene 1 – Blatt 3/3..... B5

Lattice ispLSI 5384V – Funktionales Untermodul..... B6

Lattice ispLSI 5384V – 13-Bit-Zähleinheit..... B7

Lattice ispLSI 5384V – Frequenzteiler..... B8

Lattice ispLSI 5384V – 14-Bit-Subtraktionseinheit..... B9

Lattice ispLSI 5384V – Interne Zähler- und Multiplexing-Einheit..... B10

FADC-Modul : Schematics für Lattice ispLSI 2032

Lattice ispLSI 2032 – Schematic für FADC-Modul (Zähler und Logik)..... B11

FADC-TTL-Optokopplerboard : Schematics für Lattice ispLSI 2096

Lattice ispLSI 2096 – Schematic für TTL-Optokopplerboard (32 zu 8 Multiplexer)..... B12

FADC-Interfaceboard : Schematics für Lattice ispLSI 2096V

Lattice ispLSI 2096V – Beschaltung quasi-statischer Leitungen..... B13

Delay-Board : Schematics für Lattice ispLSI 2096

Lattice ispLSI 2096 – Schematic für Delayboard..... B14

Abel-Quell-Dateien für Lattice ispLSI 5384V

dcnt4.abl (4 Bit-Abwärts-Zähler)..... B15

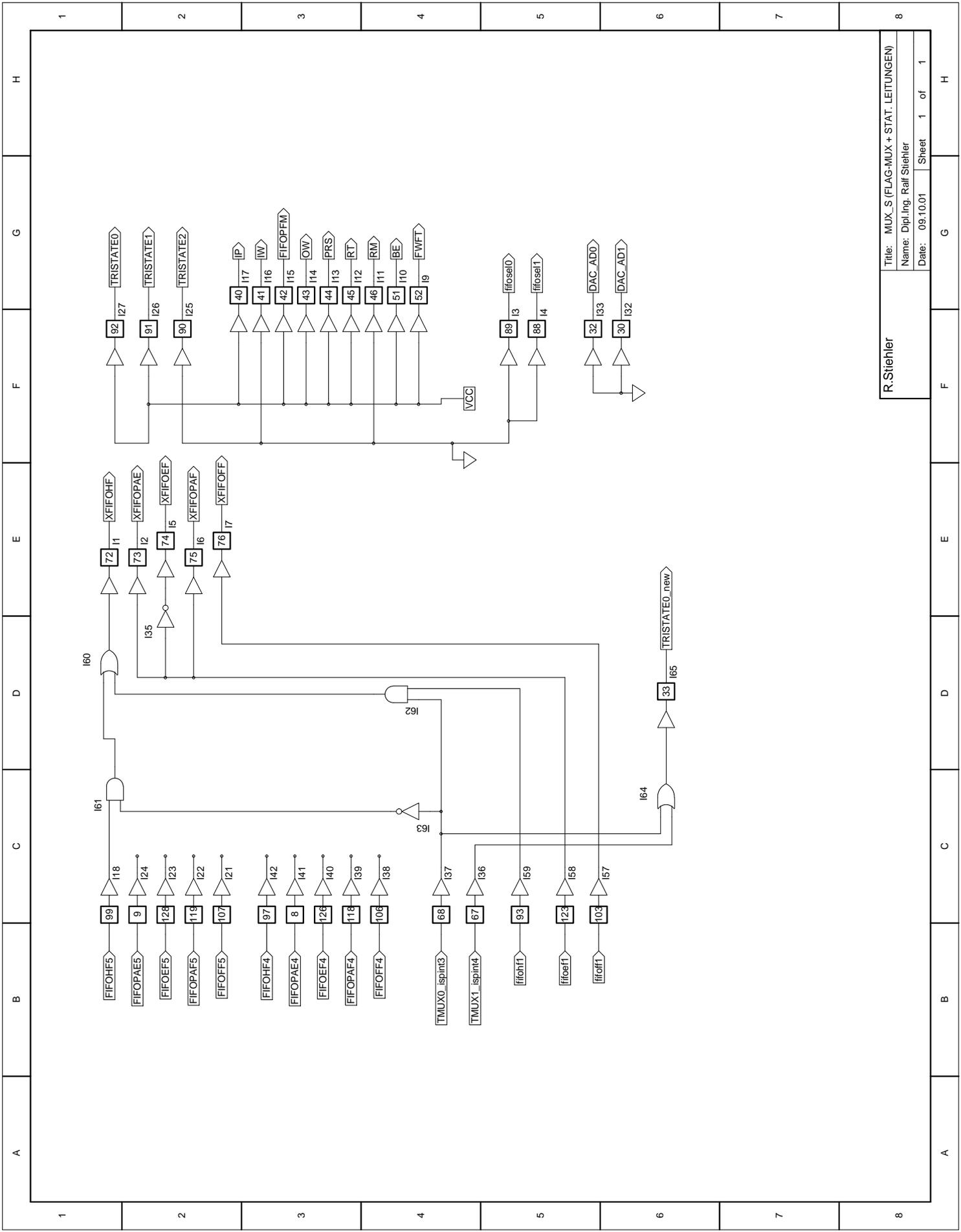
encode.abl (3-8-Encoder)..... B16

reg14.abl (14 Bit Synchrones Register mit LOAD-Funktion)..... B17

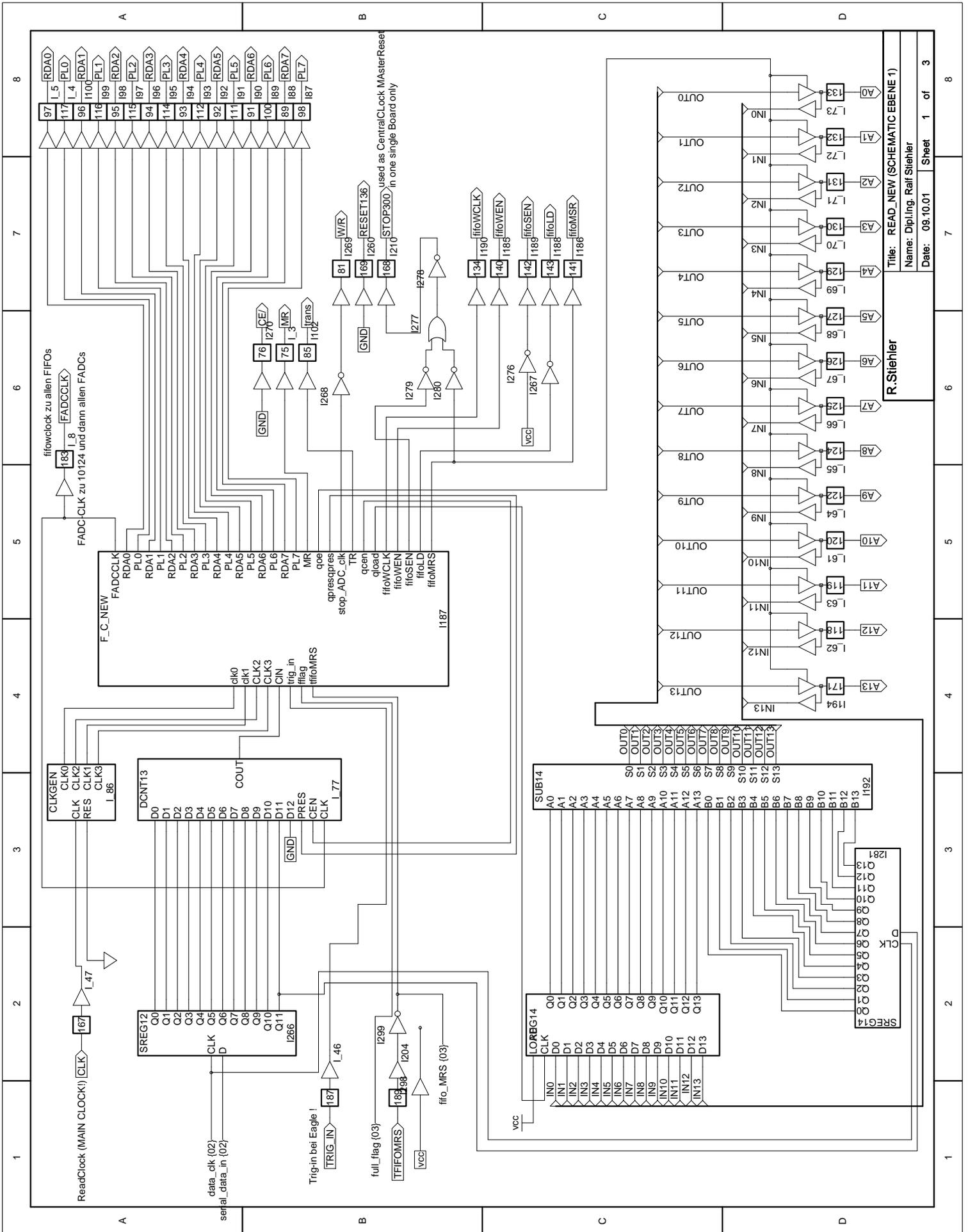
sreg12.abl (12-Bit-Schieberegister)..... B18

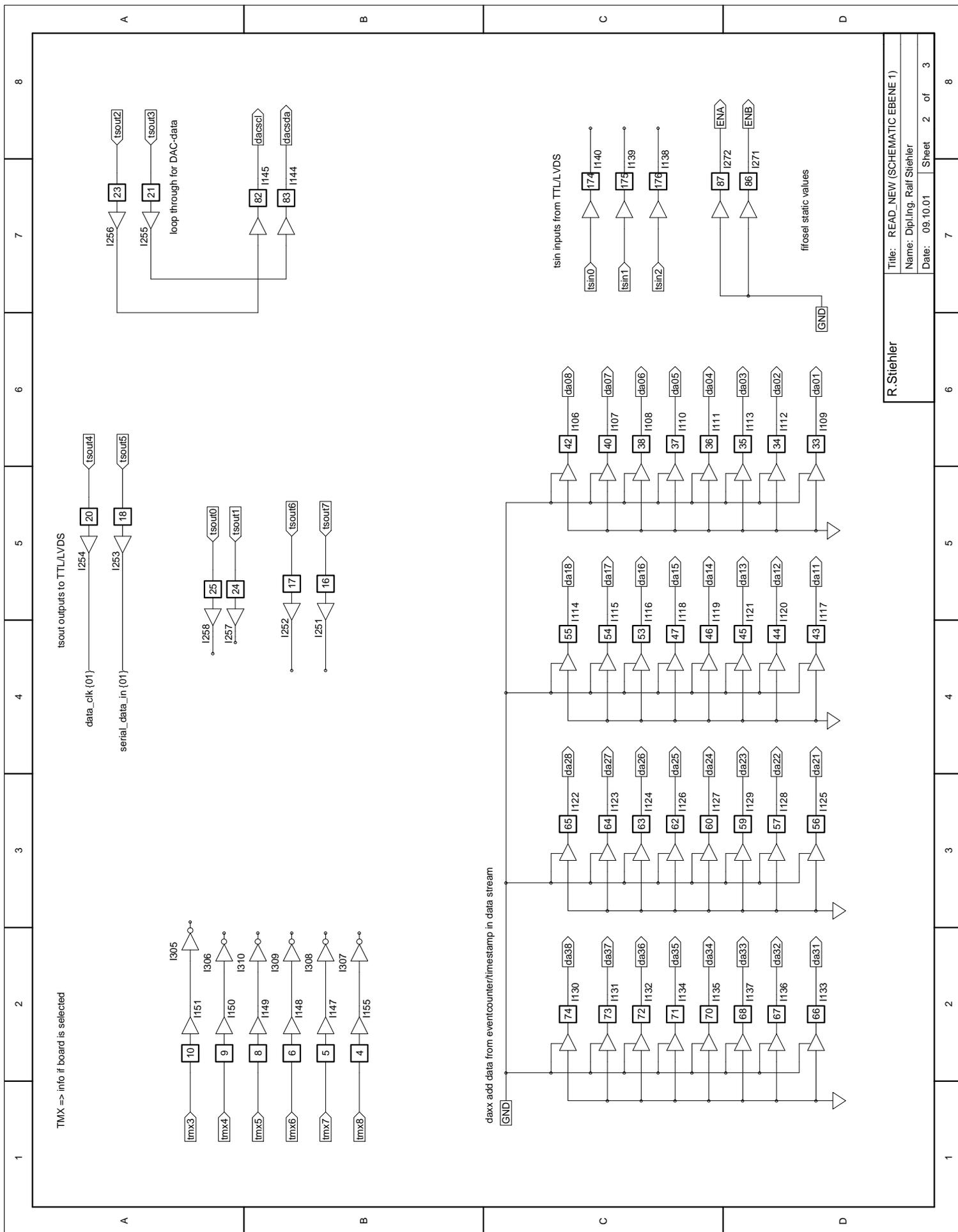
sub2f.abl (2-Bit-Binär-Subtraktionsmodul)..... B19

fadcnew.abl (Hauptprogramm zur Steuerung)..... B20..B24

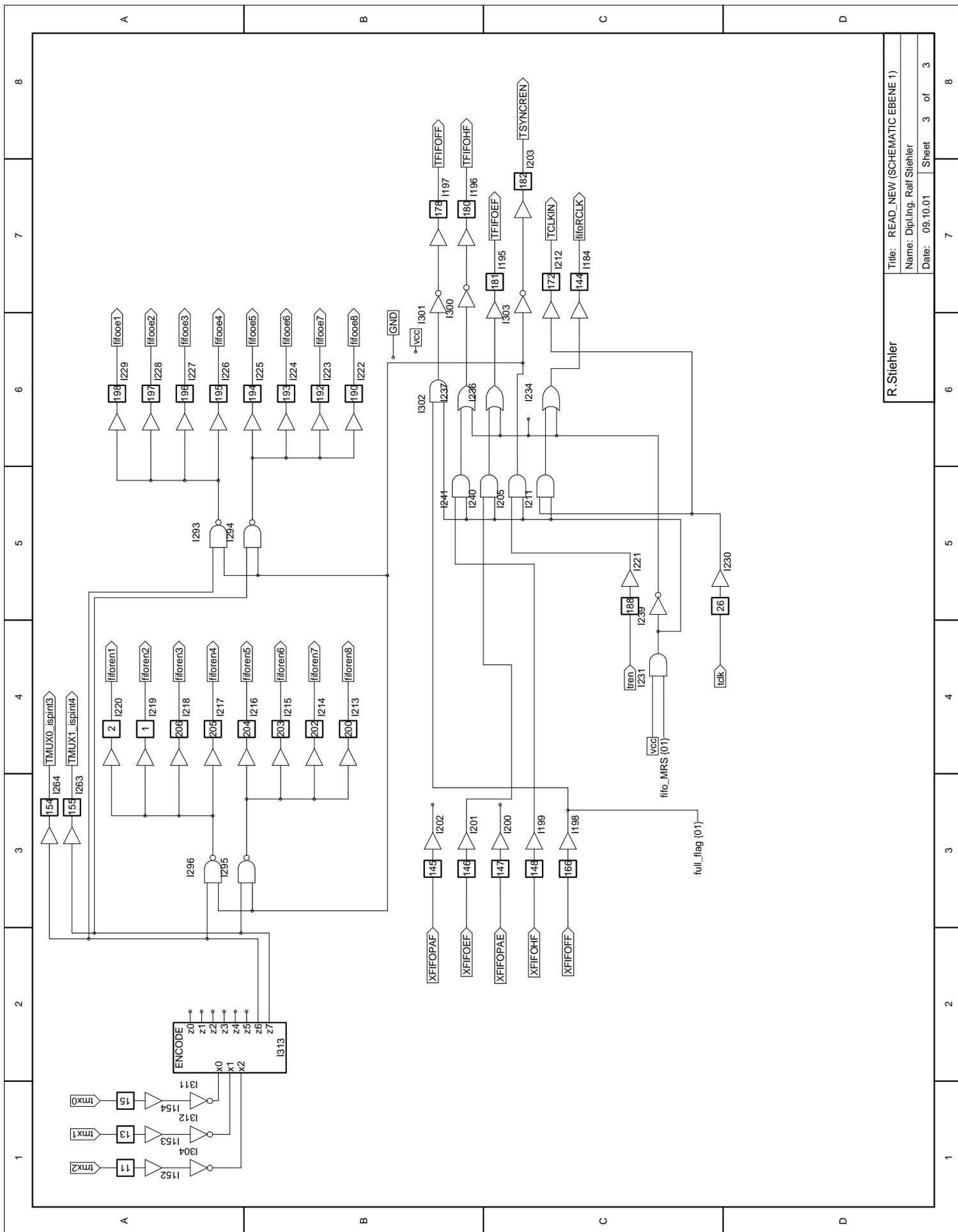


R. Stiehler	Title: MUX_S (FLAG-MUX + STAT. LEITUNGEN)		
	Name: Dipl.Ing. Ralf Stiehler		
	Date: 09.10.01	Sheet 1	of 1

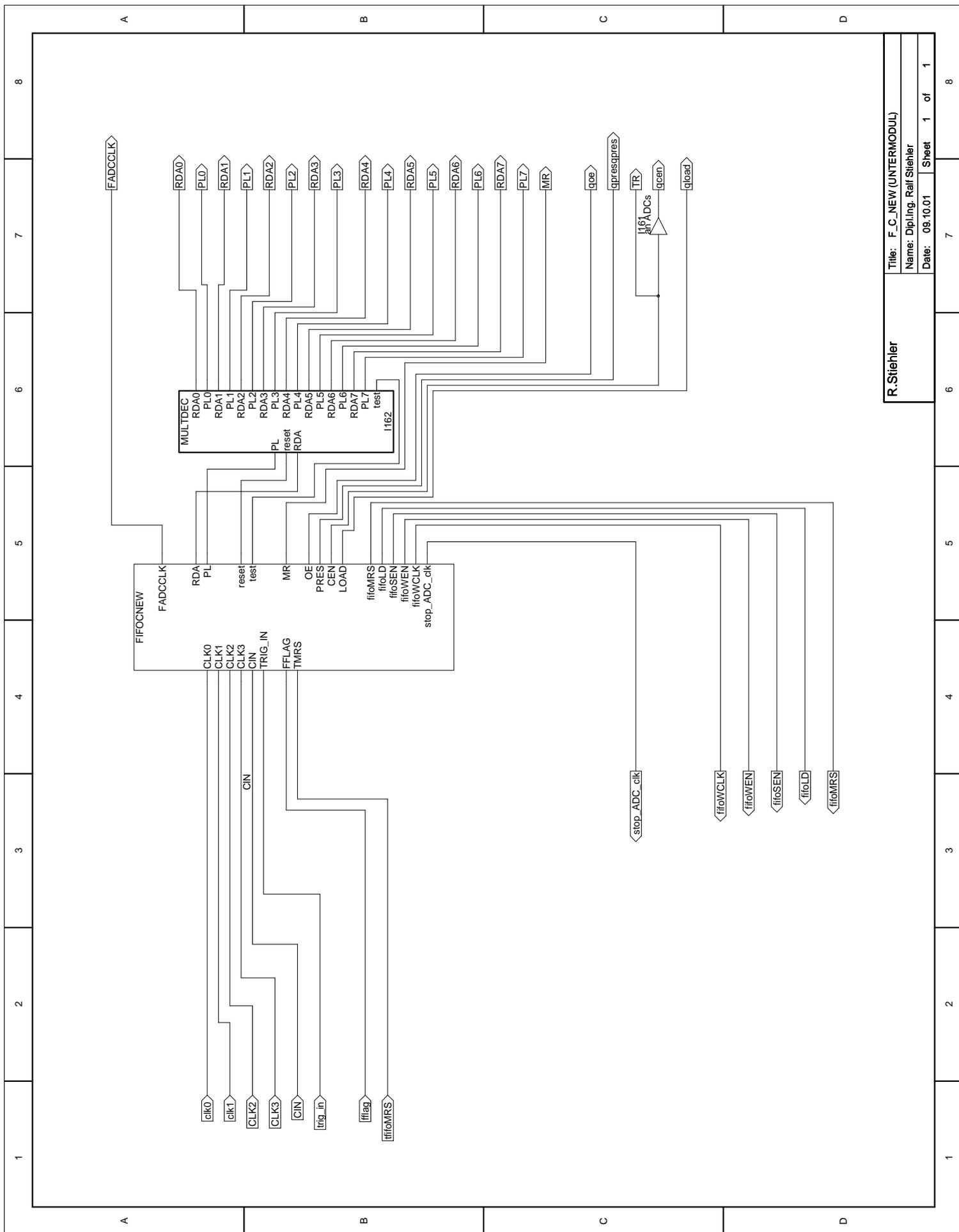




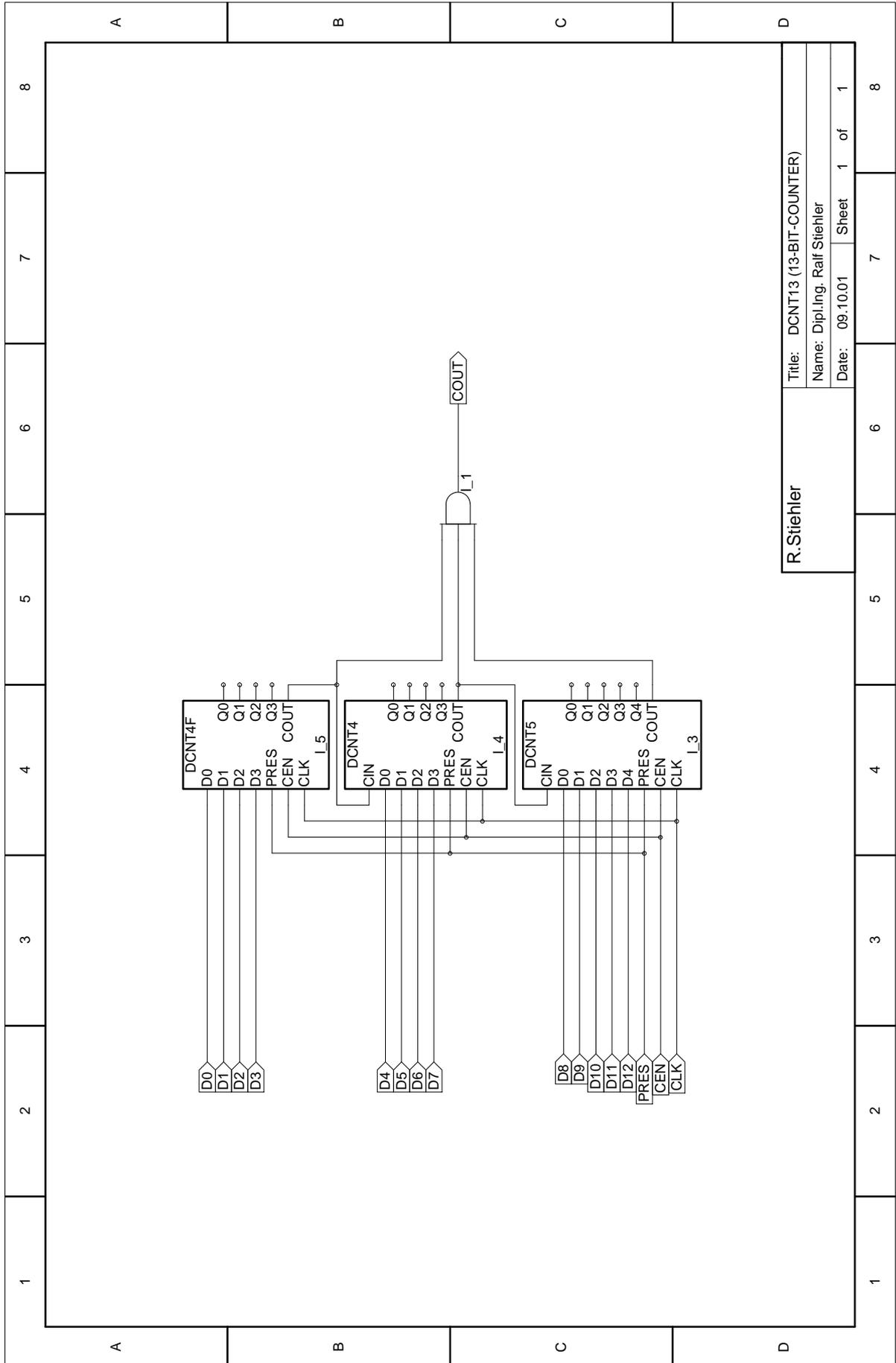
R.Stiehl			
Title: READ_NEW (SCHEMATIC EBENE 1)			
Name: Dipl.Ing. Ralf Stiehl			
Date: 09.10.01	Sheet 2	of 3	8



R. Stiehler		
Title: READ_NEW (SCHEMATIC EBENE 1)		
Name: Dipl.Ing. Ralf Stiehler		
Date: 09.10.01	Sheet 3	of 3

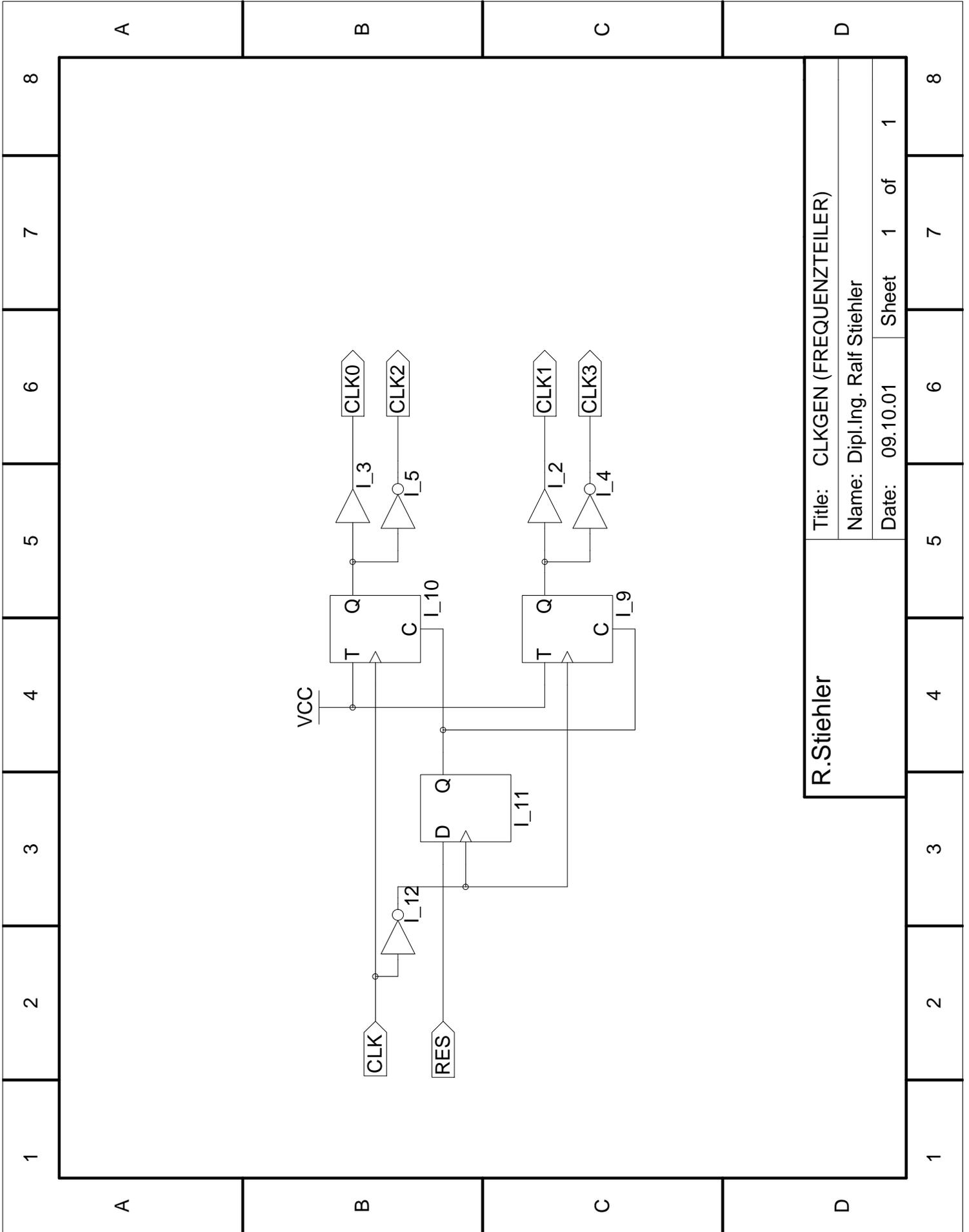


R. Stiehler
Title: F_C_NEW (UNTERMODUL)
Name: Dipl.Ing. Ralf Stiehler
Date: 09.10.01 Sheet 1 of 1



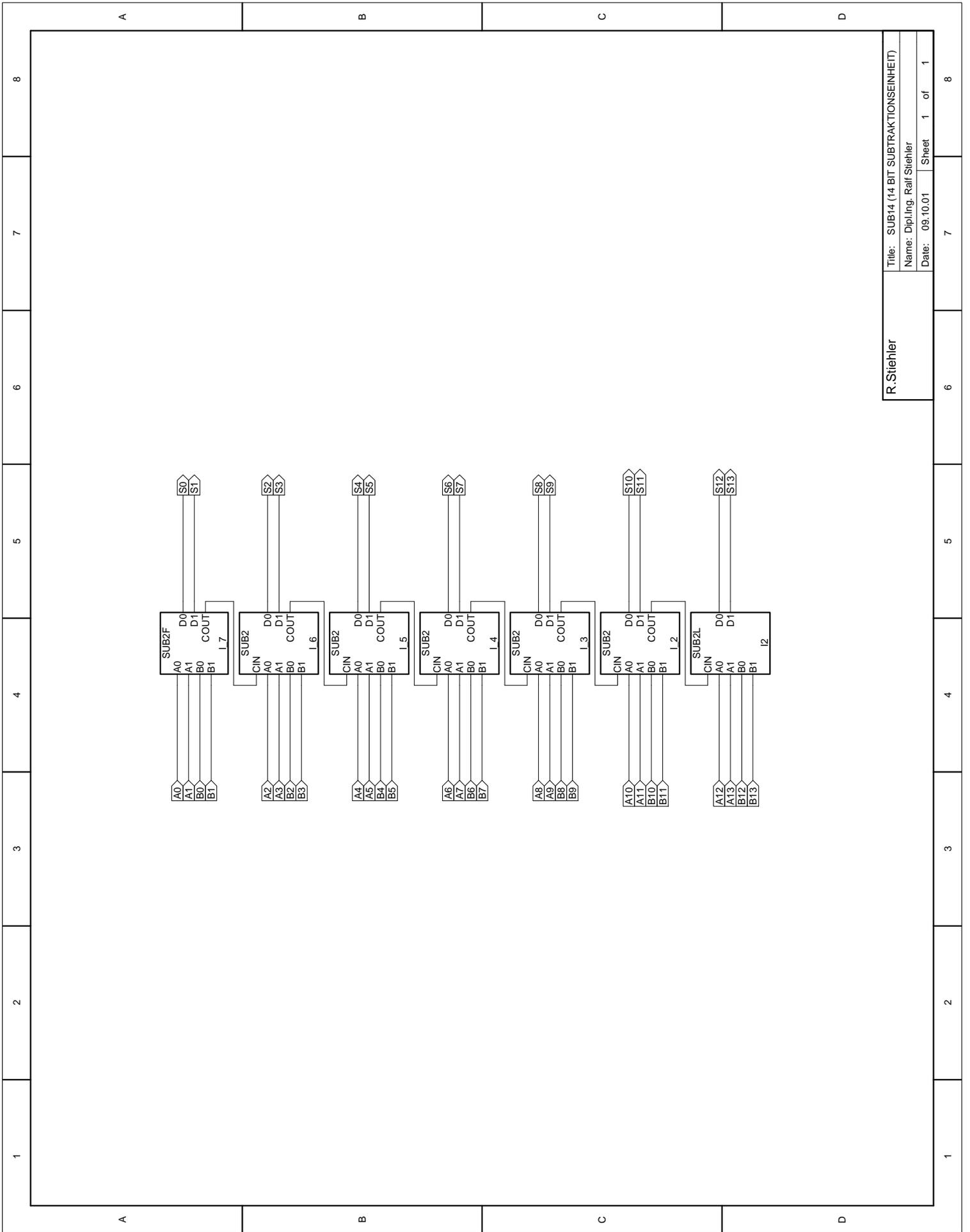
R. Stiehler	Title: DCNT13 (13-BIT-COUNTER)	
	Name: Dipl.Ing. Ralf Stiehler	
	Date: 09.10.01	Sheet 1 of 1

1	2	3	4	5	6	7	8
---	---	---	---	---	---	---	---

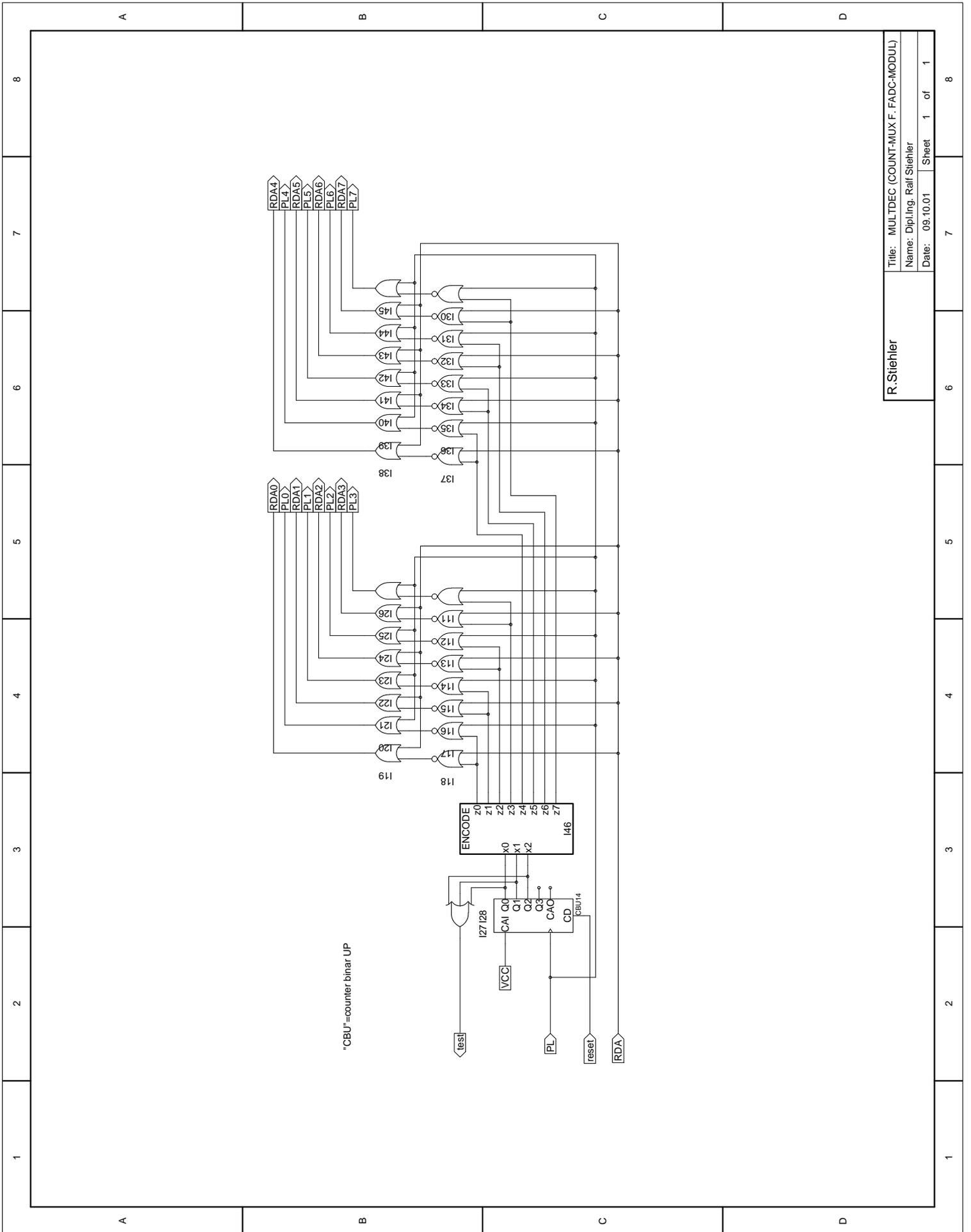


R.Stiehler	Title: CLKGEN (FREQUENZTEILER)
Name: Dipl.Ing. Ralf Stiehler	
Date: 09.10.01	Sheet 1 of 1

1	2	3	4	5	6	7	8
A	B	C	D				

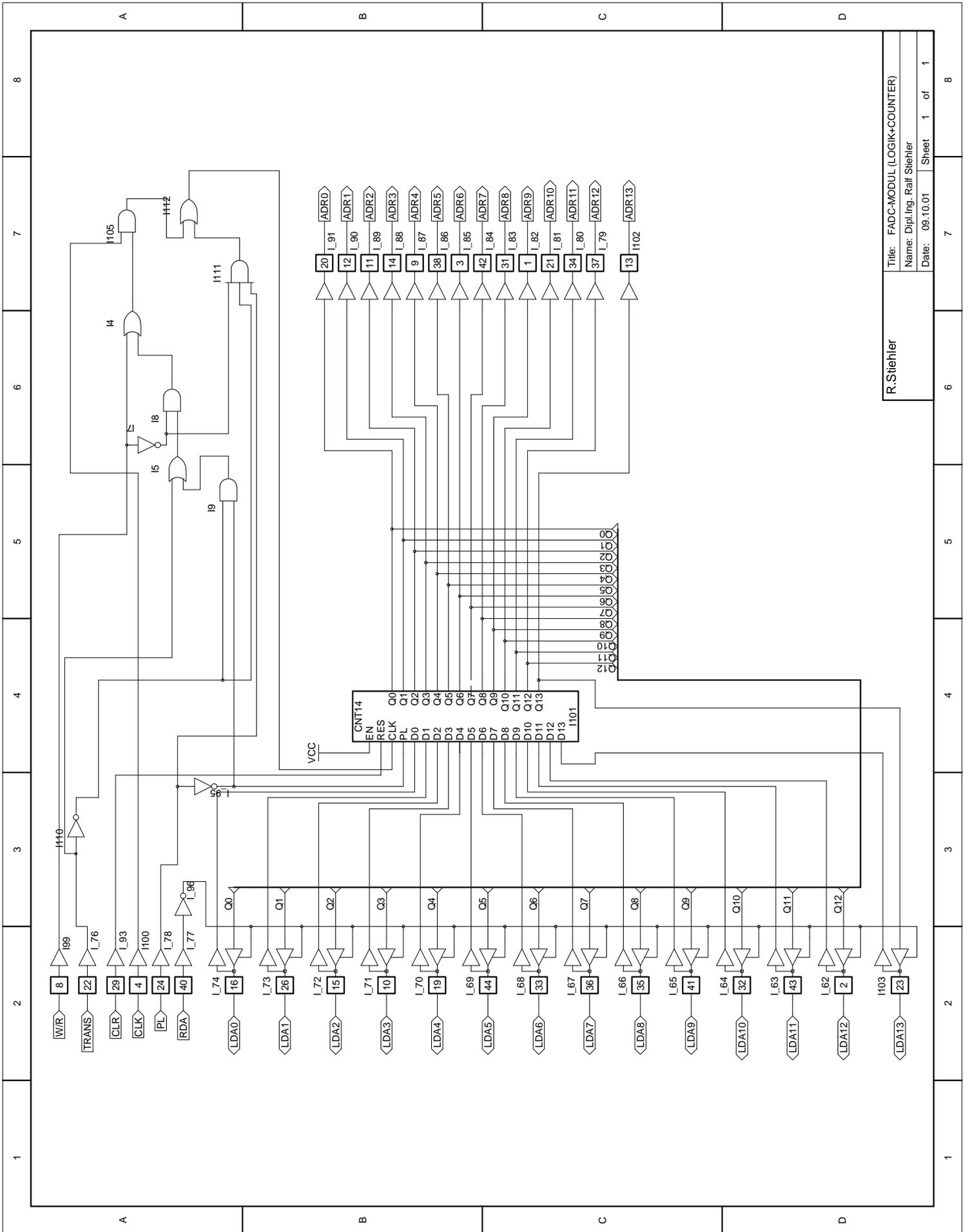


R. Stiehler	Title: SUB14 (14 BIT SUBTRAKTIONSEINHEIT)		
	Name: Dipl.Ing. Ralf Stiehler		
	Date: 09.10.01	Sheet 1	of 1



R. Stiehler	
Title: MULTDEC (COUNT-MUX F. FADC-MODUL)	
Name: Dipl.Ing. Ralf Stiehler	
Date: 09.10.01	Sheet 1 of 1

1	2	3	4	5	6	7	8
A	B	C	D				



R. Stiehler

Title: FADC-MODUL (LOGIK+COUNTER)

Name: Dipl.-Ing. Ralf Stiehler

Date: 09.10.01

Sheet 1 of 1

8

7

6

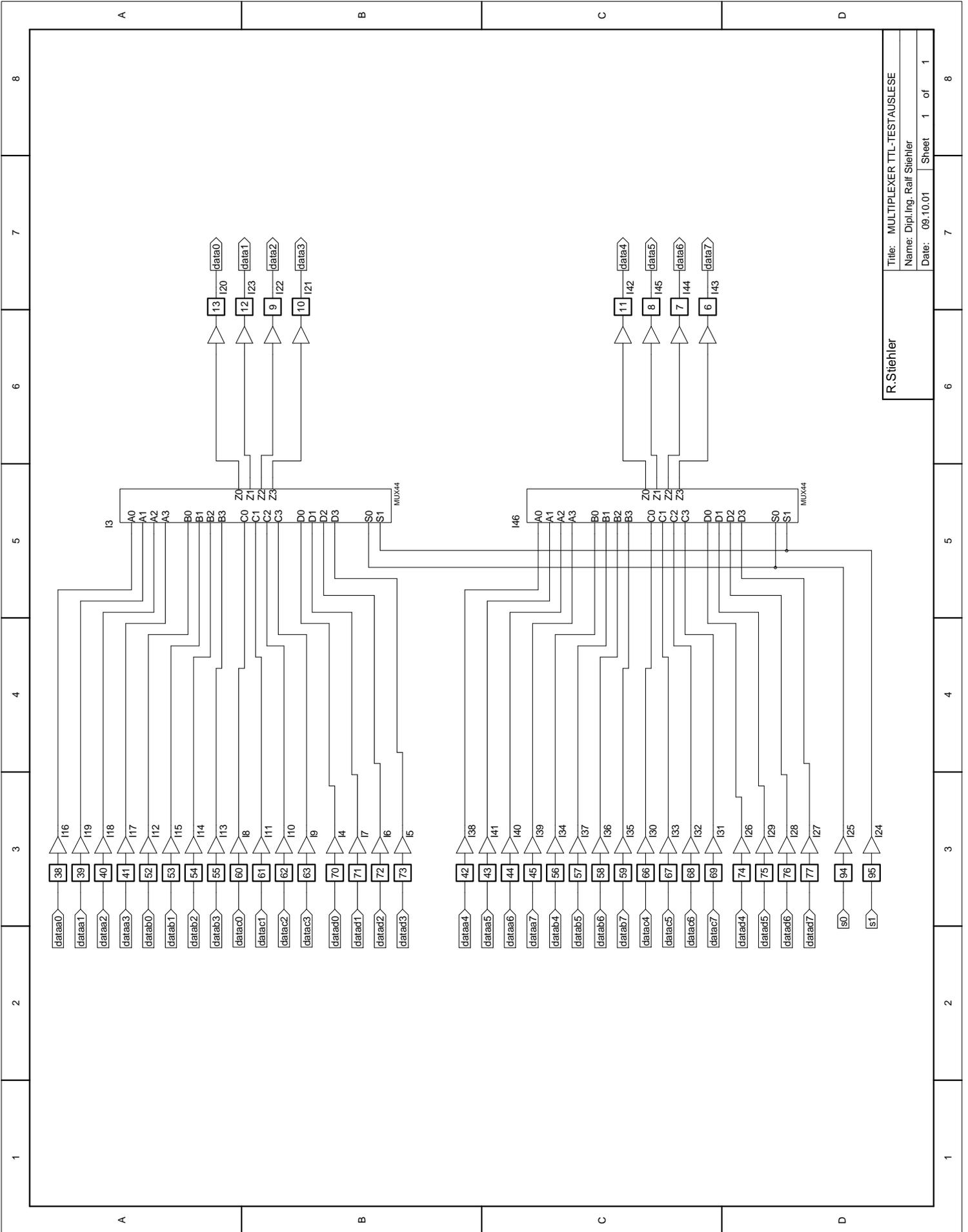
5

4

3

2

1



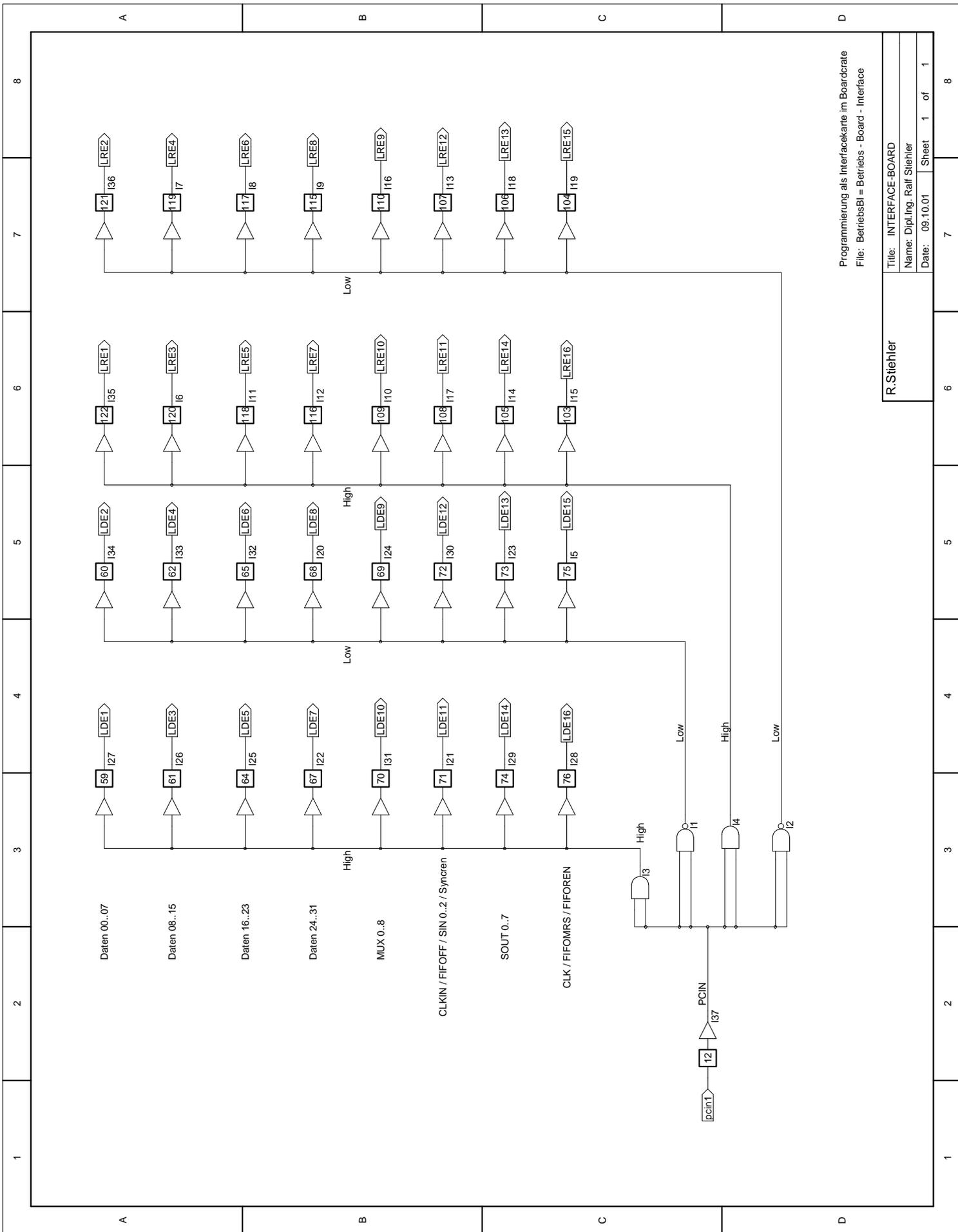
R. Stiehler

Title: MULTIPLEXER TTL-TESTAUSLESE
Name: Dipl.Ing. Ralf Stiehler
Date: 09.10.01

Sheet 1 of 1

7 6 5 4 3 2 1

8 7 6 5 4 3 2 1



Programmierung als Interfacekarte im Boardrate
 File: BetriebsBI = Betriebs - Board - Interface

R.Stiehler

Title: INTERFACE-BOARD
 Name: Dipl.Ing. Ralf Stiehler
 Date: 09.10.01

Sheet 1 of 1

8

7

6

5

4

3

2

1

8

7

6

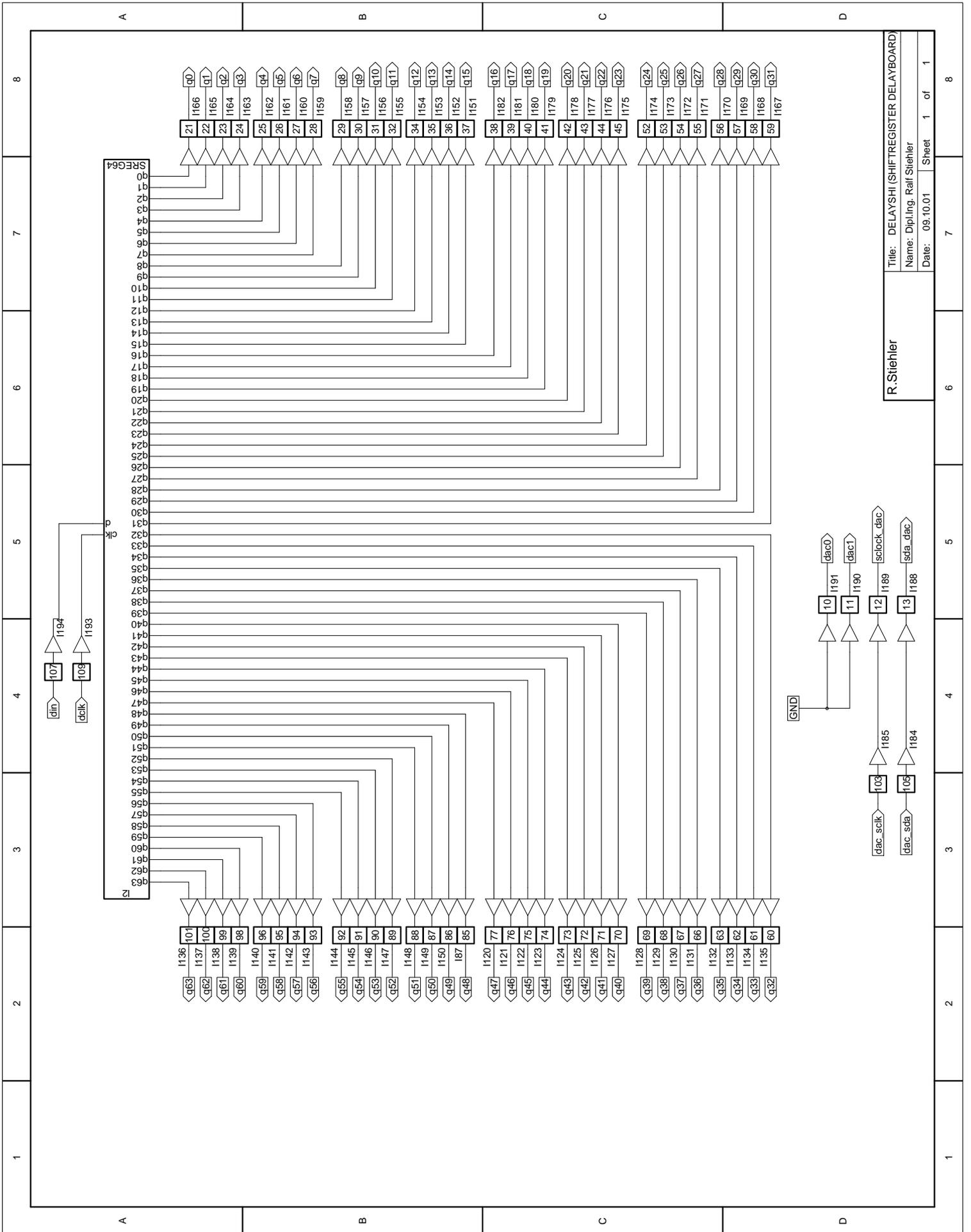
5

4

3

2

1



R. Stiehler

Title: DELAYSHI (SHIFTREGISTER DELAYBOARD)
Name: Dipl.Ing. Ralf Stiehler
Date: 09.10.01 Sheet 1 of 1

Abel-Quell-Dateien für Lattice ispLSI 5384V**dcnt4.abl (4 Bit-Abwärts-Zähler)**

(aus Gründen der Übersichtlichkeit ohne Testvektoren dargestellt)

```

MODULE DCNT4

"Modul ist validiert!

TITLE '4 bit binary down counter with count enable preset and async reset'

DECLARATIONS

"Constants
    X,C=.x.,.c.;

"Inputs
    CIN    pin;
    D3..D0 pin;
    PRES   pin;
    CEN    pin;
    CLK    pin;

"Outputs
    Q3..Q0 pin istype 'reg';
    COUT   pin istype 'com';

"Sets
    data    =[D3..D0];
    count   =[Q3..Q0];

    mode    =[CEN,PRES];
    hold     =(mode==[0,0]);
    cnt      =(mode==[1,0]);
    set      =(mode==[X,1]);

EQUATIONS

    count.clk=CLK;
    when hold then count:=count
    else when cnt then count:=count-[0,0,0,CIN]
    else when set then count:=data;

    COUT=!Q3 & !Q2 & !Q1 & !Q0 & CIN;

END

```

Anmerkung : Die Module dcnt4f.abl und dcnt5.abl sind analog dazu strukturiert, auf eine Auflistung wurde daher verzichtet.

Abel-Quell-Dateien für Lattice ispLSI 5384V**encode.abl (3-8-Encoder)**

```
MODULE encode
TITLE '3-to-8 Encoder'

"inputs
    x0,x1,x2 pin;
    x = [x2..x0];

"outputs
    z0, z1, z2, z3, z4, z5, z6, z7 pin istype 'com';

truth_table (x -> [z0, z1, z2, z3, z4, z5, z6, z7])
    0 -> [1, 0, 0, 0, 0, 0, 0, 0];
    1 -> [0, 1, 0, 0, 0, 0, 0, 0];
    2 -> [0, 0, 1, 0, 0, 0, 0, 0];
    3 -> [0, 0, 0, 1, 0, 0, 0, 0];
    4 -> [0, 0, 0, 0, 1, 0, 0, 0];
    5 -> [0, 0, 0, 0, 0, 1, 0, 0];
    6 -> [0, 0, 0, 0, 0, 0, 1, 0];
    7 -> [0, 0, 0, 0, 0, 0, 0, 1];

END encode
```

Abel-Quell-Dateien für Lattice ispLSI 5384V**reg14.abl (14 Bit Synchrones Register mit LOAD-Funktion)**

```
MODULE REG14

TITLE '14 Bit synchron Register with LOAD'

"Universität-gesamthochschule Siegen
"Fachbereich Physik

DECLARATIONS

"Constants
    C,X=.C.,.X.;

"Inputs
    CLK    pin;
    LOAD   pin;
    D0..D13 pin;

"Outputs
    Q0..Q13 pin istype 'reg';

"Sets
    input  =[D13..D0];
    output =[Q13..Q0];

EQUATIONS

    output.clk=CLK;
    when LOAD then output:=input;
    else output:=output;

TEST_VECTORS ([CLK,LOAD,input]->[output]);

    [C,1,8191]->[8191];
    [C,0, X]->[8191];
    [C,1, 0]->[ 0];
    [C,0, X]->[ 0];

END
END encode
```

Abel-Quell-Dateien für Lattice ispLSI 5384V**sreg12.abl (12-Bit-Schieberegister)**

```
MODULE sreg12
TITLE '12 bit shift register with par. out'
DECLARATIONS
"Constants
    C,X=.C.,.X.;
"Inputs
    D    pin;
    CLK  pin;
"Outputs
    Q0..Q11 pin istype 'reg';
"Sets
    input  =[Q10..Q0,D];
    output =[Q11..Q0];
EQUATIONS
    output.clk=CLK;
    output:=input;
END
```

Anmerkung : Die Module sreg14.abl und sreg64.abl sind analog dazu strukturiert, auf eine Auflistung wurde daher verzichtet.

Abel-Quell-Dateien für Lattice ispLSI 5384V**sub2f.abl (2-Bit-Binär-Subtraktionsmodul)**

(aus Gründen der Übersichtlichkeit ohne Testvektoren dargestellt)

```
MODULE SUB2L
"Modul ist validiert!
TITLE '2 bit binary subtract with carry in'
DECLARATIONS
"Constants
    X,C=.x.,c.;
"Inputs
    CIN    pin;
    A1..A0 pin;
    B1..B0 pin;
"Outputs
    D1..D0 pin istype 'com';
"Sets
    va=[A1..A0];
    vb=[B1..B0];
    vd=[D1..D0];
EQUATIONS
    vd=((va-vb)-[0,CIN]);
END
```

Anmerkung : Die Module sub2.abl und sub2f.abl sind analog dazu strukturiert, auf eine Auflistung wurde daher verzichtet.

Abel-Quell-Dateien für Lattice ispLSI 5384V**fadcnew.abl (Hauptprogramm zur Steuerung)**

```
MODULE FIFOCNEW

TITLE 'Main-Control-Unit'

"Ralf Stiehler

DECLARATIONS

"Constants
    X,C=.X.,.C.;

    "Gray-Code
    "S0=0; S1=1; S2=3; S3=2;
    "S4=6; S5=7; S6=5; S7=4;
"    S8=12; S9=13; S10=15; S11=14;
"    S12=10; S13=11; S14=8; S15=9;
"    S16=18; S17=17; S18=16;

    "Gray-Code
    S0=0; S1=1; S2=3; S3=2;
    S4=6; S5=7; S6=5; S7=4;
    S8=12; S9=15; S10=31; S11=27;
    S12=19; S13=14; S14=18; S15=13;
    S16=16; S17=17; S18=23; S19=30;
    S20=21 ; S21=29;

    "Für Fehlersuche binär
"    S0=0; S1=1; S2=2; S3=3; S4=4; S5=5; S6=6; S7=7; S8=8;
"    S9=9; S10=10; S11=11; S12=12; S13=13; S14=14; S15=15;

"Inputs
    CIN                pin;
    CLK0                pin;
    CLK1                pin;
    CLK2                pin;
    CLK3                pin;
    TRIG_IN            pin;
    TMRS                pin;
    FFLAG                pin; "the Input Ready signal from FIFO
    test                pin;
```

(Fortsetzung nächste Seite)

"Outputs

```

fifoMRS          pin istype 'com';
fifoWEN          pin istype 'com';
fifoWCLK         pin istype 'com';
fifoSEN         pin istype 'com';
                "used as Masterreset for CentralClock -, fifosen @ Latticepin == HIGH
fifoLD          pin istype 'com';
FADCCLK         pin istype 'com';
RDA             pin istype 'com';
PL              pin istype 'com';
reset           pin istype 'com';
MR              pin istype 'com';
OE             pin istype 'com';
PRES           pin istype 'com';
CEN            pin istype 'com';
                " NSP-DowncounterEnable used as TRANS-Signal to
                " indicate datatransfer to FADC Module
LOAD           pin istype 'com';
stop_ADC_clk   pin istype 'com';" used as W/R which goes to FADC Module

```

```

Q0..Q4 pin istype 'reg';

```

"Sets

```

sreg   =[Q4..Q0];

```

EQUATIONS

```

sreg.clk=CLK0;

state_diagram sreg;

state S0: fifoWCLK=0; fifoWEN=1; FADCCLK=0;          "Master Reset cycle for FIFO's
        RDA=1; PL=1; reset=1;
        MR=1;
        OE=0; PRES=0; CEN=0; LOAD=0;
        fifoMRS=0; fifoSEN=1; stop_ADC_clk=0;
        fifoLD=0;
        if (TMRS==0) then S0
            else S1;

state S1: fifoWCLK=0; fifoWEN=1; FADCCLK=0;          "warten auf TRIG_IN=1,FFLAG=0
        RDA=1; PL=1; reset=1;
        MR=1; fifoMRS=1; fifoSEN=1; stop_ADC_clk=0;
        OE=0; PRES=0; CEN=0; LOAD=0;
        fifoLD=0;
        if (TMRS==0) then S0
            else if (TRIG_IN==1) & (FFLAG==0) then S2
                else S1;

state S2: fifoWCLK=0; fifoWEN=1; FADCCLK=0;          "START DER AUSLESE
        RDA=0; PL=1; MR=1;
        fifoSEN=1; fifoMRS=1; stop_ADC_clk=1; reset=0;
        fifoLD=0;
        OE=0; PRES=0; CEN=0; LOAD=0;          "RDA0=0 um FADC-RDAesszähler
        if (TMRS==0) then S0                  "Auszulesen
            else S3;

```

(Fortsetzung nächste Seite)

```

state S3: fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0; "Adresse in REG13 laden
        RDA=0; PL=1; MR=1;
        fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
        fifoLD=0;
        OE=0; PRES=0; CEN=0; LOAD=1;
        if (TMRS==0) then S0
            else S4;

state S4: fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0; "Auf Subtraktion warten...
        RDA=1; PL=1; MR=1;
        fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
        fifoLD=0;
        OE=1; PRES=0; CEN=0; LOAD=0;    "OE=1 um Adresse zu laden
        if (TMRS==0) then S0
            else S5;

state S5: fifoWCLK=0; fifoWEN=1; FADCCLK=0;reset=0; "dito...
        RDA=1; PL=1; MR=1;
        fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
        fifoLD=0;
        OE=1; PRES=0; CEN=0; LOAD=0;
        if (TMRS==0) then S0
            else S6;

state S6: fifoWCLK=0; fifoWEN=1; FADCCLK=0;reset=0; "dito...
        RDA=1; PL=1; MR=1;
        fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
        fifoLD=0;
        OE=1; PRES=0; CEN=0; LOAD=0;
        if (TMRS==0) then S0
            else S7;

state S7: fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0; "DCNT13 mit NSP laden
        RDA=1; PL=0; MR=1;
        fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
        fifoLD=0;
        OE=1; PRES=1; CEN=0; LOAD=0;
        if (TMRS==0) then S0
            else S8;

state S8: fifoWCLK=0; fifoWEN=1; FADCCLK=1; reset=0;    "1 Taktimpuls an FADC um
        RDA=1; PL=0; MR=1;    "Adresse in Counter zu laden
        fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
        fifoLD=0;
        OE=1; PRES=1; CEN=0; LOAD=0;
        if (TMRS==0) then S0
            else S15;

state S15:    fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0;
        RDA=1; PL=0; MR=1;
        fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;    " PL=0 and FADCCLK = 0

        fifoLD=0;
        OE=1; PRES=1; CEN=0; LOAD=0;
        goto S9;

```

(Fortsetzung nächste Seite)

```

state S9: fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0;
        MR=1; RDA=1; PL=1;
        fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
        fifoLD=0;
        OE=0; PRES=0; CEN=0; LOAD=0;
        if (TMRS==0) then S0
            else if (test==1) then S2
            else S13;

state S13:    fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0;
            MR=1; RDA=1; PL=1;
            fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
            fifoLD=0;
            OE=0; PRES=0; CEN=0; LOAD=0;
            if (TMRS==0) then S0
            else S19;

state S19:    fifoWCLK=0; fifoWEN=0; FADCCLK=0; reset=0; // WEN=0 vorzubereiten
                                                    //4.4. NUN AUF WEN==1
            MR=1; RDA=1; PL=1;
            fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
            fifoLD=0;
            OE=0; PRES=0; CEN=0; LOAD=0;
            goto S10;

state S10:    fifoWCLK=0; fifoWEN=0; FADCCLK=0; reset=0; //NORMAL OPERATION
                                                    "Samples in RAM schreiben
            //fifoWCLK=CLK1; fifoWEN=0; FADCCLK=CLK1; reset=0;
            ///STAY IN LOOP TEST //vorher clk1 clk1
            RDA=1; PL=1; MR=1;
            fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
            fifoLD=0;
            OE=0; PRES=0; CEN=1; LOAD=0;
                                                    "bis CIN oder FFlag==1

            if (TMRS==0) then S0
            else if ((CIN==1) # (FFLAG==1)) then S11
            //     else if (FFLAG==1) then S11
                //else S10; //STAY IN LOOP TEST
                else S18; //Normal Operation

state S11:    fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0;
            "MR=0 um Zähler zu synchronisieren
            RDA=1; PL=1; MR=1;
                                                    "MR und Trans Vorhaltephase
            fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
            fifoLD=0;
            OE=0; PRES=0; CEN=1; LOAD=0;
                                                    "CEN=1 um Zustandsänder. zu unterdr.
            goto S12;

state S12:    fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0;
            "MR=0 um Zähler zu synchronisieren
            RDA=1; PL=1; MR=1;
            fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
            fifoLD=0;
            OE=0; PRES=0; CEN=0; LOAD=0;
            goto S14;

```

(Fortsetzung nächste Seite)

```

state S14:    fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0;
              "MR=0 um Zähler zu synchronisieren
              RDA=1; PL=1; MR=0;
              fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
              fifoLD=0;
              OE=0; PRES=0; CEN=0; LOAD=0;
              goto S16;

state S16:    fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0;
              "SYSEN=1, Übertragung abgeschlossen
              RDA=1; PL=1; MR=1;
              fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;
              fifoLD=0;
              OE=0; PRES=0; CEN=0; LOAD=0;
              goto S17;

state S17:    fifoWCLK=0; fifoWEN=1; FADCCLK=0; reset=0;
              "fifosel=0, Reset to CentralCLock
              RDA=1; PL=1; MR=1;
              fifoSEN=0; fifoMRS=1; stop_ADC_clk=1;
              fifoLD=0;
              OE=0; PRES=0; CEN=0; LOAD=0;
              goto S1;

state S18:    fifoWCLK=0; fifoWEN=0; FADCCLK=0; reset=0; "Samples in RAM schreiben
              RDA=1; PL=1; MR=1;
              fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;           //BEIDES AUF 0
              fifoLD=0;
              OE=0; PRES=0; CEN=1; LOAD=0;                   "bis CIN==1
              goto S20;

state S20:    fifoWCLK=1; fifoWEN=0; FADCCLK=1; reset=0; "Samples in RAM schreiben
              RDA=1; PL=1; MR=1;
              fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;           //BEIDES AUF 1
              fifoLD=0;
              OE=0; PRES=0; CEN=1; LOAD=0;                   "bis CIN==1
              goto S21;

state S21:    fifoWCLK=0; fifoWEN=0; FADCCLK=0; reset=0; "Samples in RAM schreiben
              RDA=1; PL=1; MR=1;
              fifoSEN=1; fifoMRS=1; stop_ADC_clk=1;           //BEIDES AUF 0
              fifoLD=0;
              OE=0; PRES=0; CEN=1; LOAD=0;                   "bis CIN==1
              goto S10;

```

END FIFOCNEW

Anhang C – Schaltpläne der einzelnen Platinen

FADC-Motherboard

Eingangs- und Ausgangstreiber, Steckverbinder.....	Seite 1/8	C2
8-fach-Digital-Analog-Wandler für Baseline.....	Seite 2/8	C3
Datenbezogener Teil der FADC-Module und FIFOs.....	Seite 3/8	C4
Steuerungsbezogener Teil der FADC-Module und FIFOs und FPGAs.....	Seite 4/8	C5
ECL-Teil der Schaltung, Pegelwandler, Clockverteilung, Frequenzteiler, Logik.....	Seite 5/8	C6
Spannungsversorgung und Spannungsfilterung.....	Seite 6/8	C7
Spannungsfilterung (Abblock-Kondensatoren).....	Seite 7/8	C8
Spannungsfilterung (Abblock-Kondensatoren).....	Seite 8/8	C9

FADC-TTL-Optokopplerboard

Eingangs-Optokoppler, Steckverbinder, Logik.....	Seite 1/2	C10
Ausgangs-Optokoppler, Steckverbinder, Logik.....	Seite 2/2	C11

FADC-Interfaceboard :

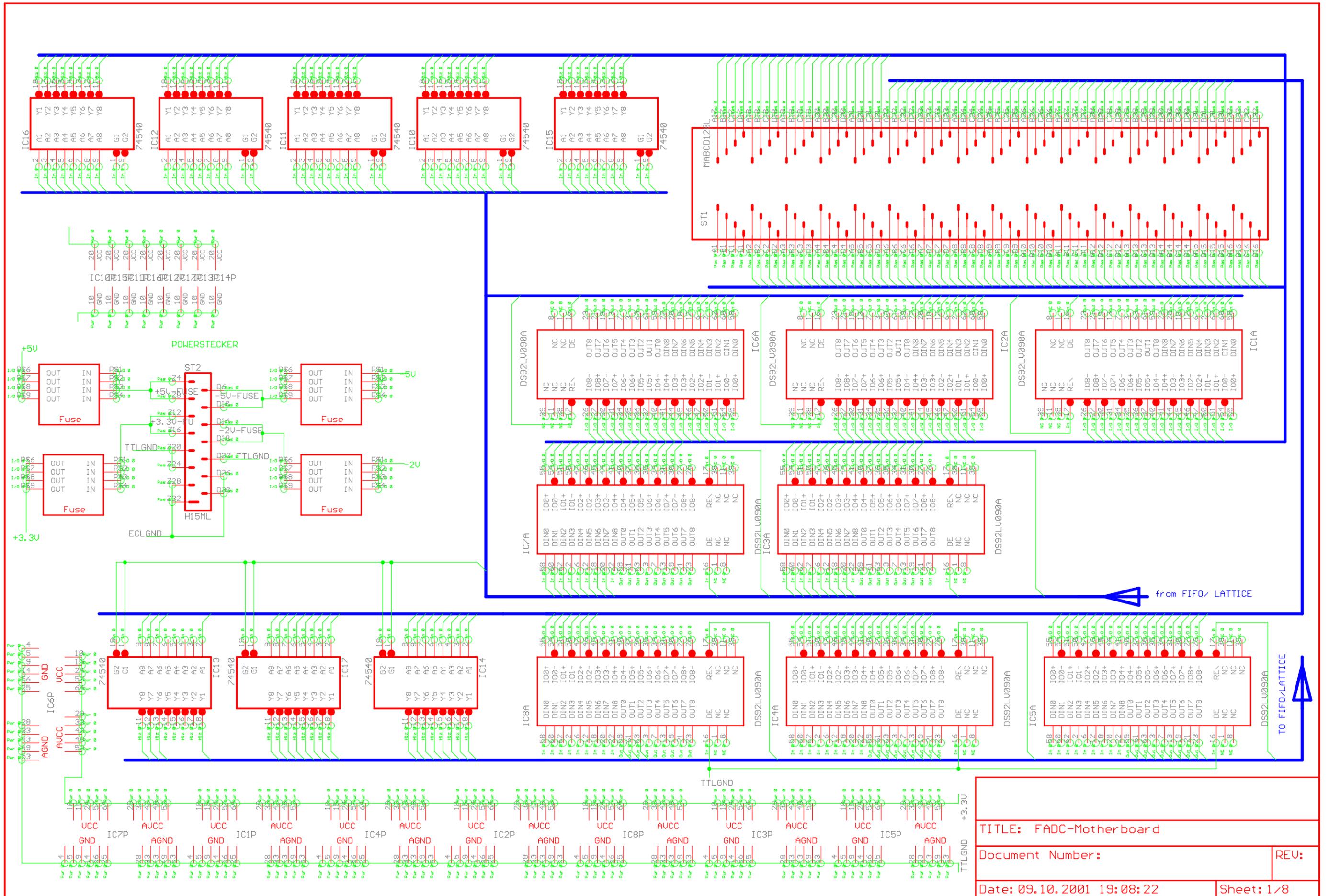
LVDS-Treiber, Spannungsfilterung, Logik.....	Seite 1/2	C12
Spannungsfilterung, Logik.....	Seite 2/2	C13

Delayboard :

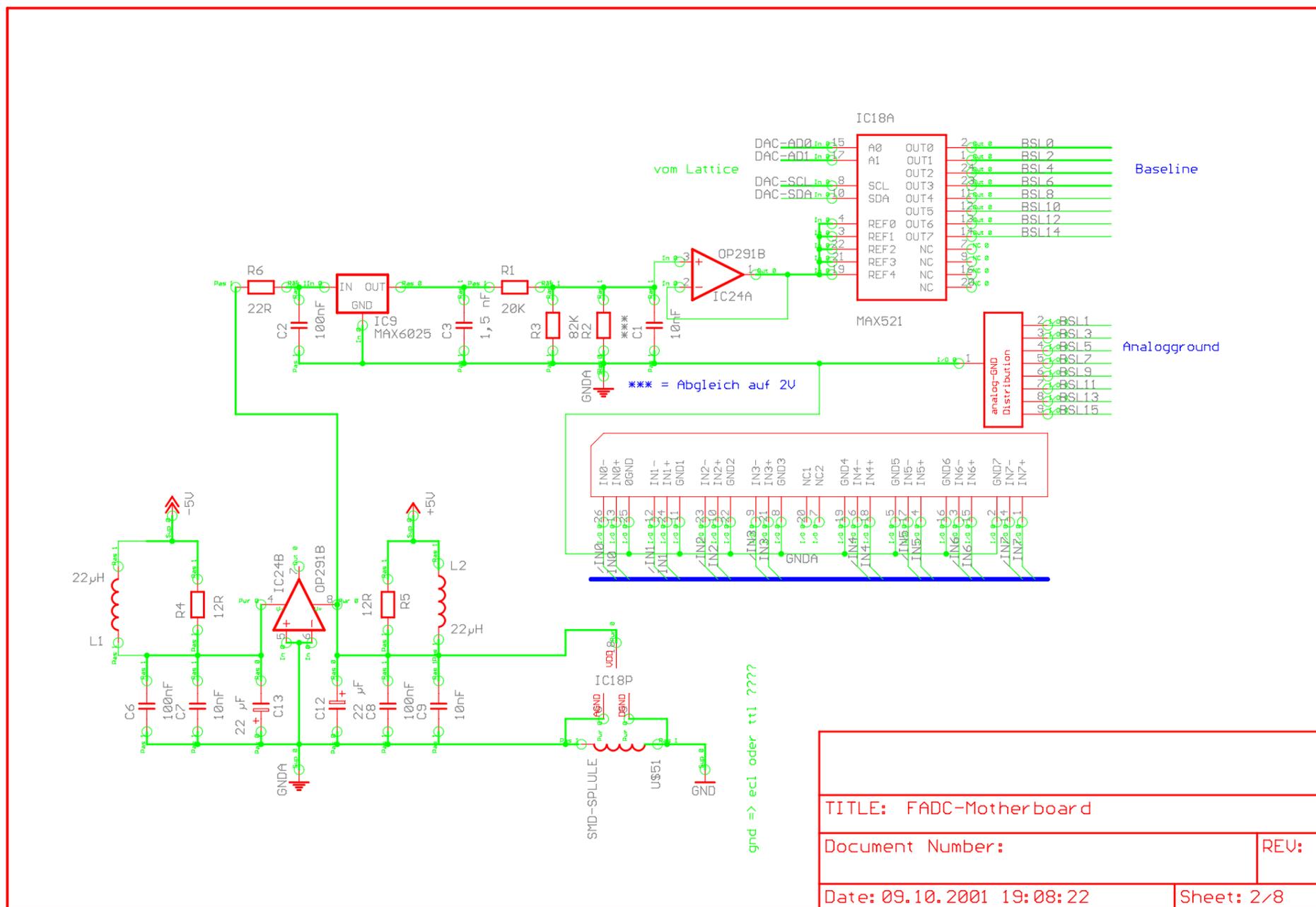
Pulsverzögerung, Pegelanpassung (1 von 8 identischen Kanälen dargestellt).....	Seite 1/4	C14
8-fach-Digital-Analog-Wandler zur Feineinstellung der Verzögerung.....	Seite 2/4	C15
Logik.....	Seite 3/4	C16
Verteilungslogik, Steckverbinder.....	Seite 4/4	C17

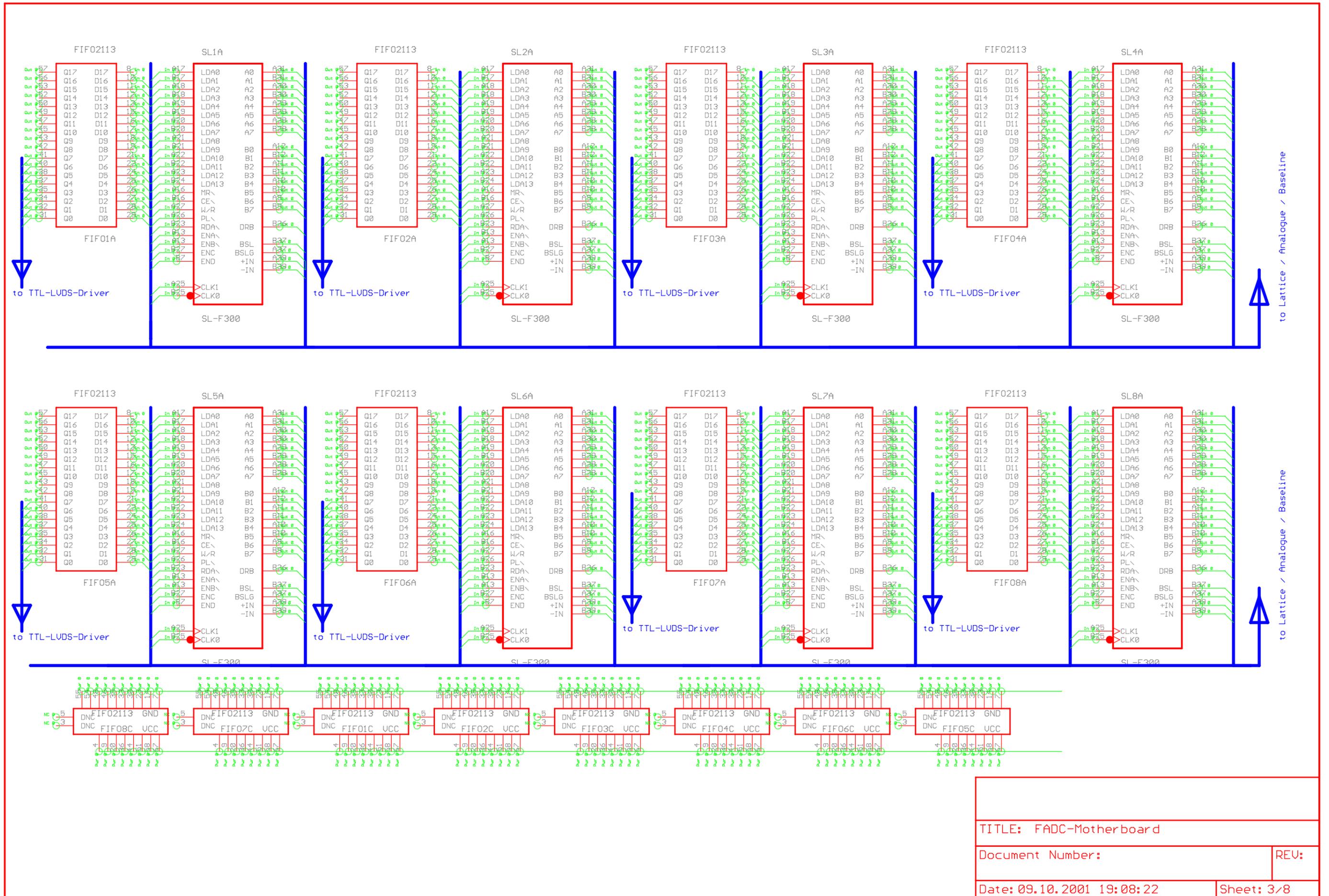
Board zur zentralen Clockerzeugung, Triggersynchronisation und Verteilung :

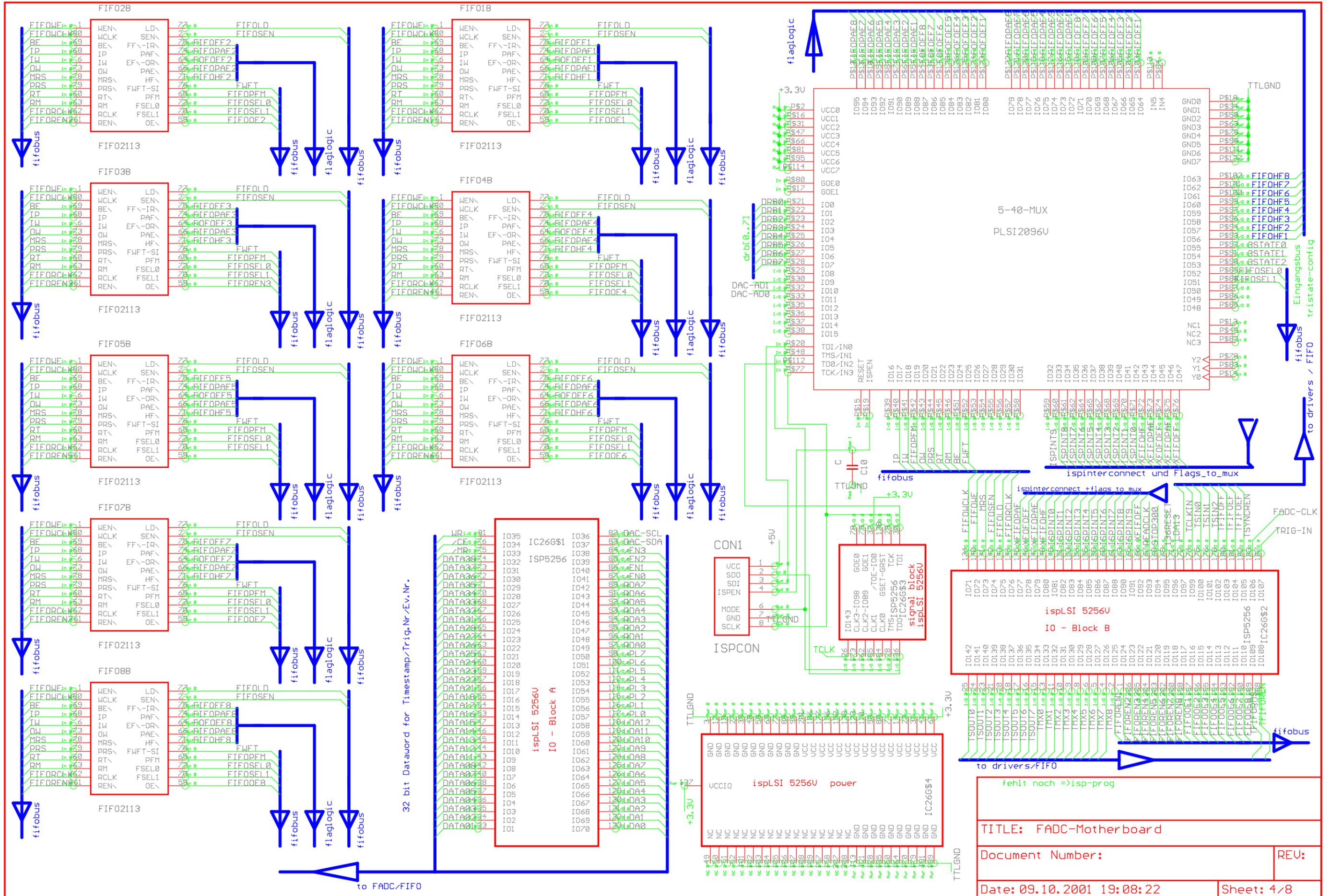
Spannungsversorgung, Clockgenerator, Logik.....	Seite 1/2	C18
Temperaturregelung mit Fühler und Lastwiderstand.....	Seite 2/2	C19



TITLE: FADC-Motherboard	
Document Number:	REV:
Date: 09.10.2001 19:08:22	Sheet: 1/8



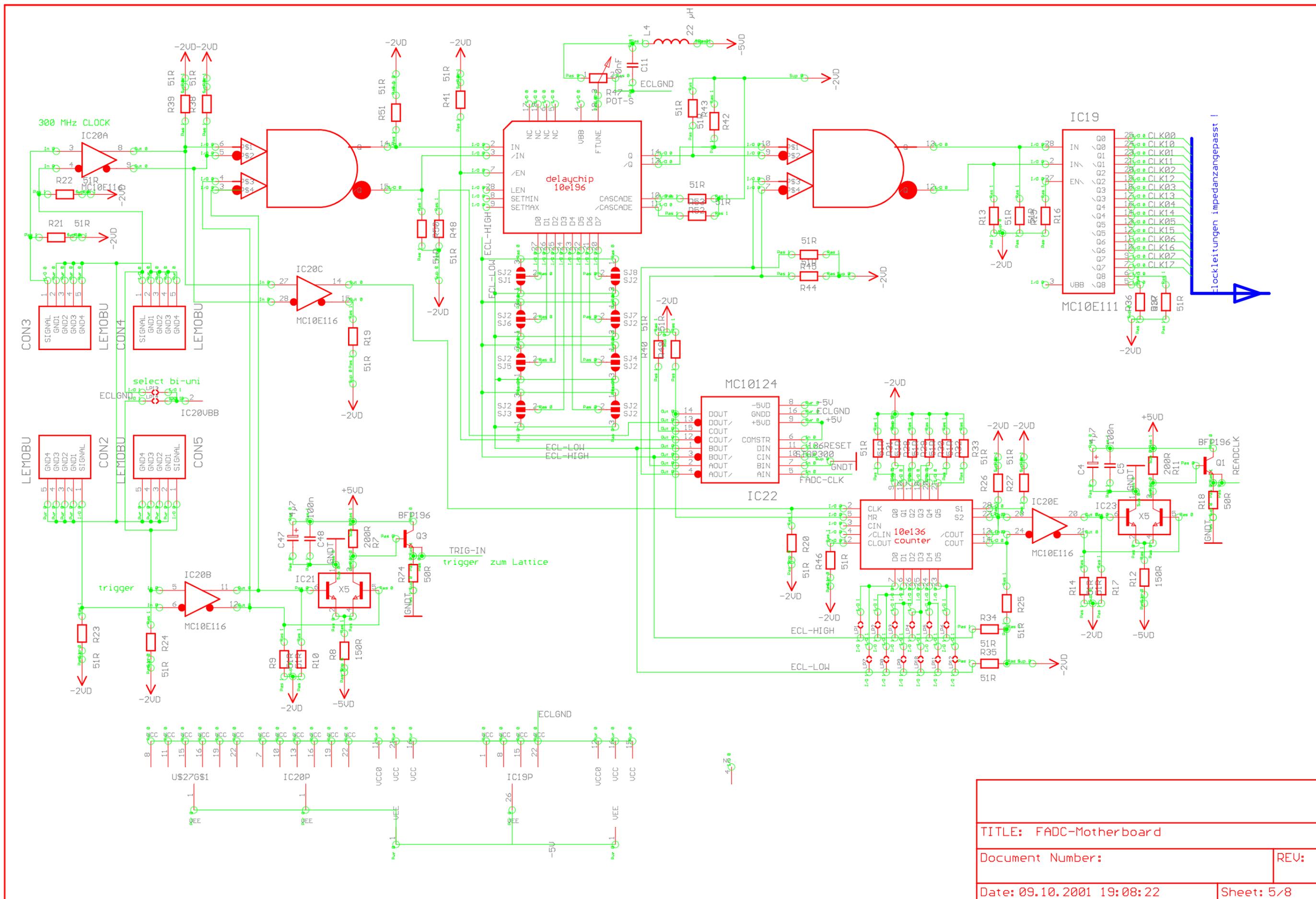




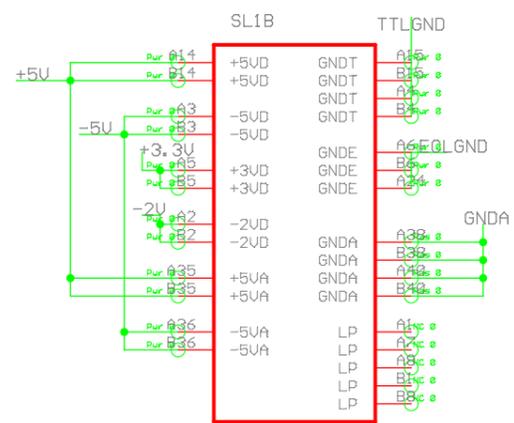
32 bit Databord for Timestamp/Trig.Nr./Ev.Nr.

fehlt noch =>isp-prog

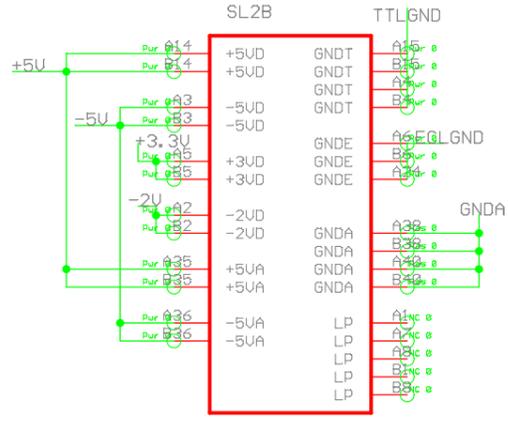
TITLE: FADC-Mother board	
Document Number:	REV:
Date: 09.10.2001 19:08:22	Sheet: 4/8



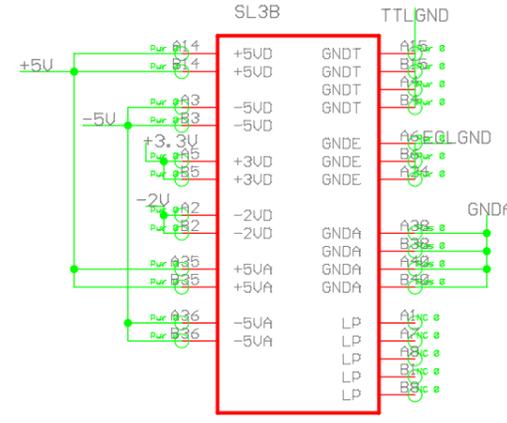
TITLE: FADC-Motherboard	
Document Number:	REV:
Date: 09.10.2001 19:08:22	Sheet: 5/8



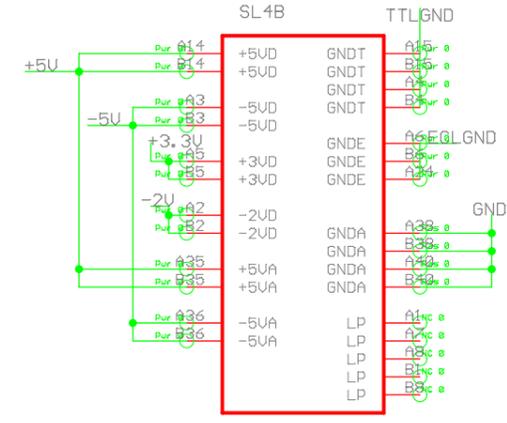
SL-F300



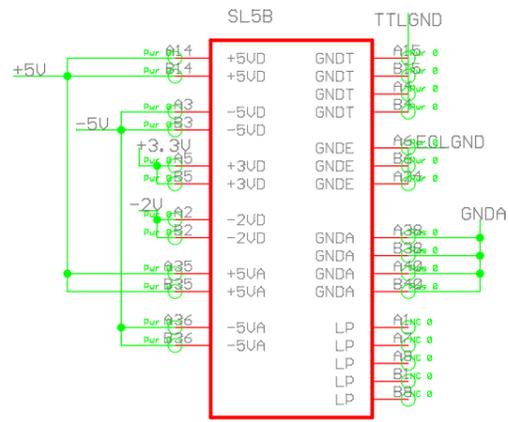
SL-F300



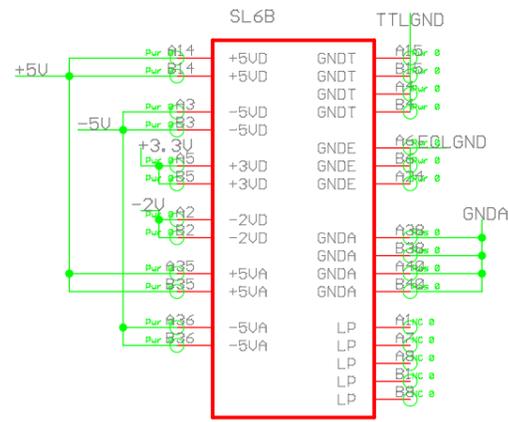
SL-F300



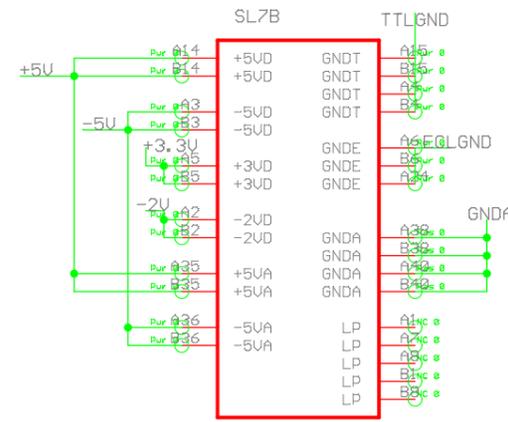
SL-F300



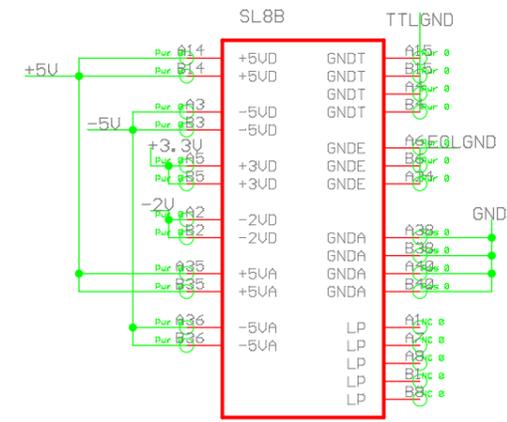
SL-F300



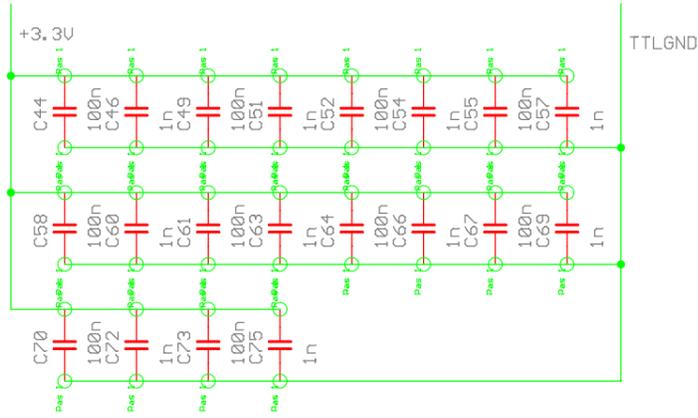
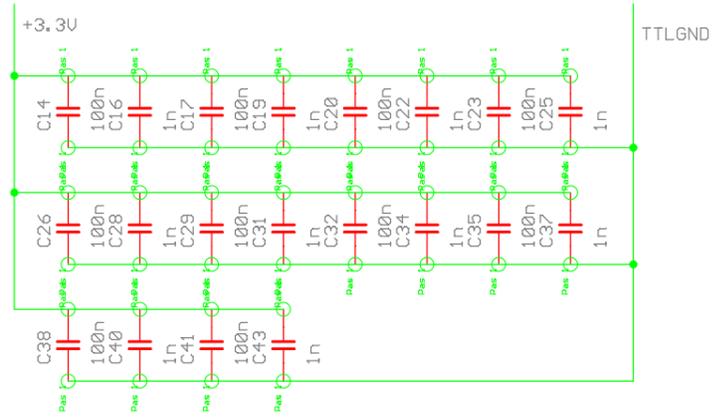
SL-F300



SL-F300



SL-F300



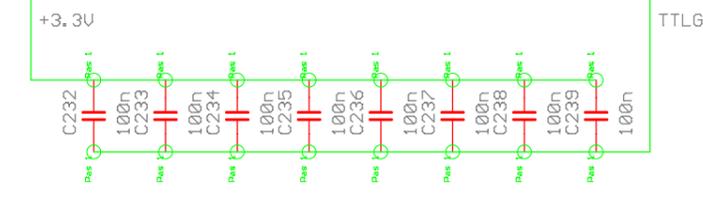
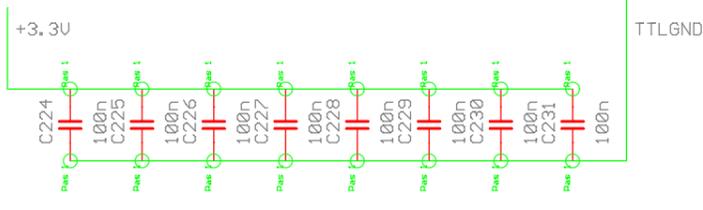
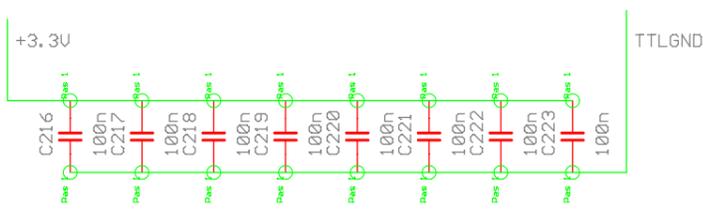
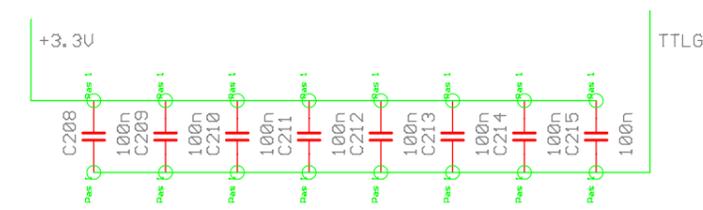
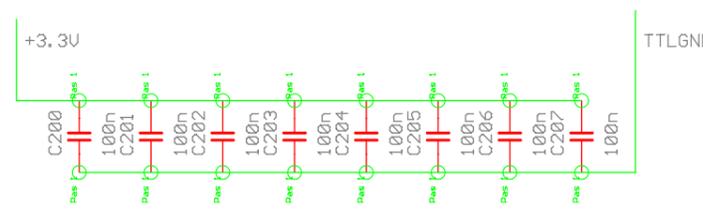
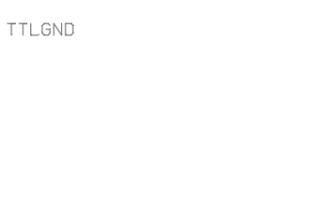
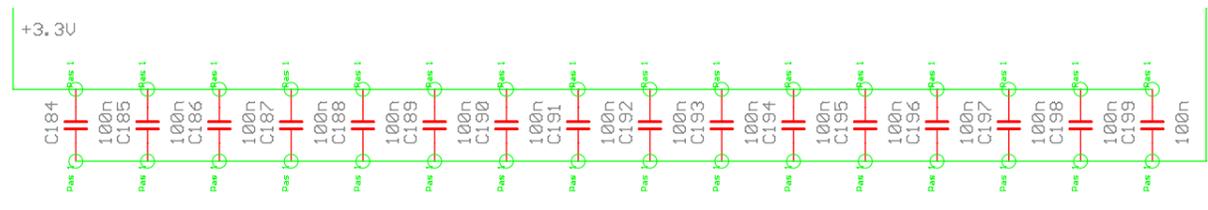
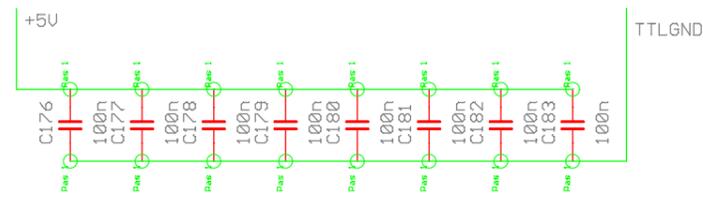
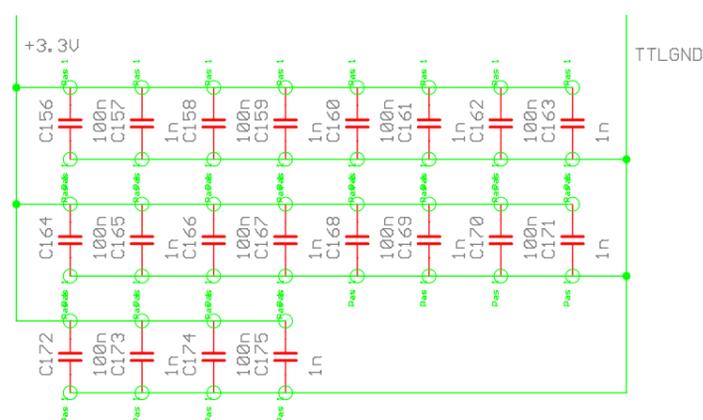
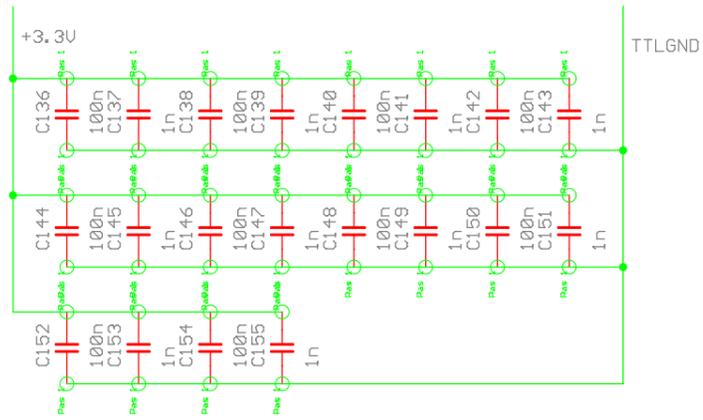
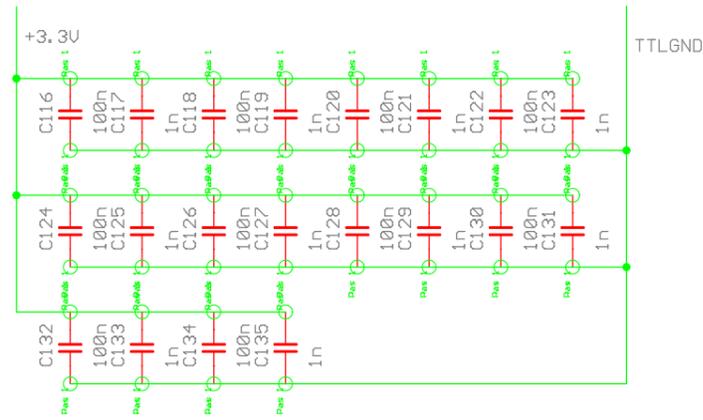
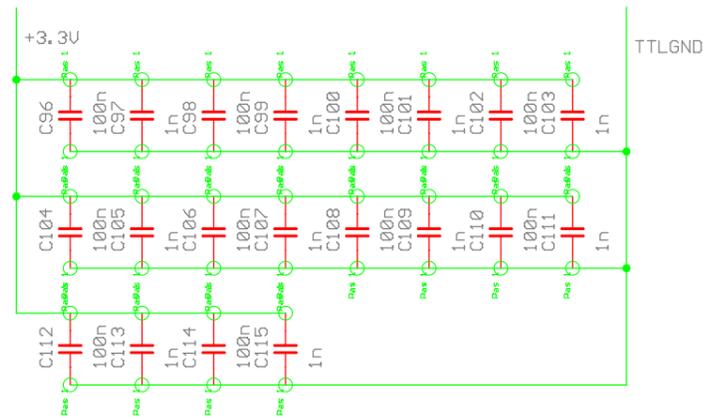
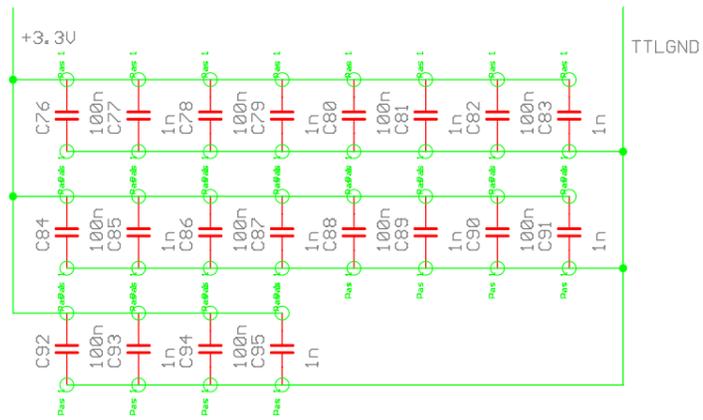
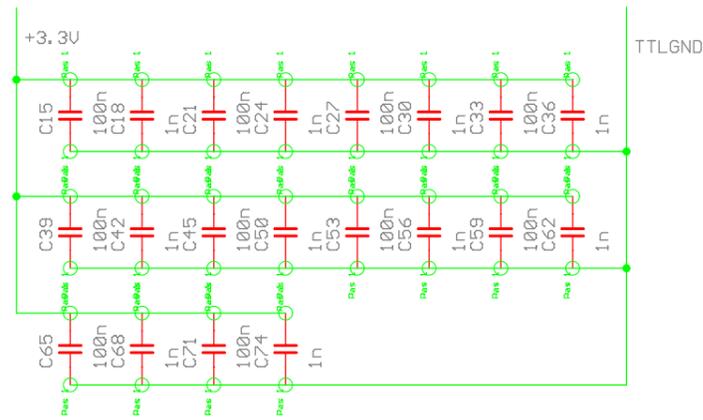
TITLE: FADC-Motherboard

Document Number:

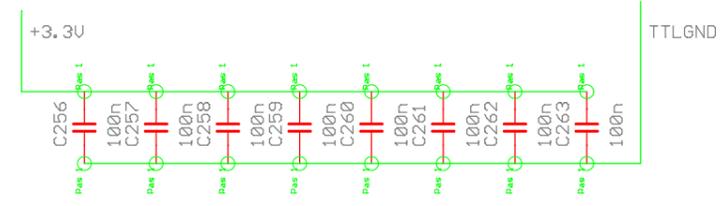
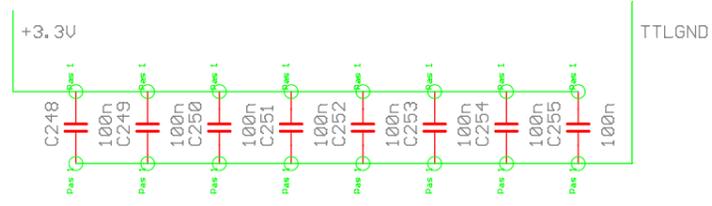
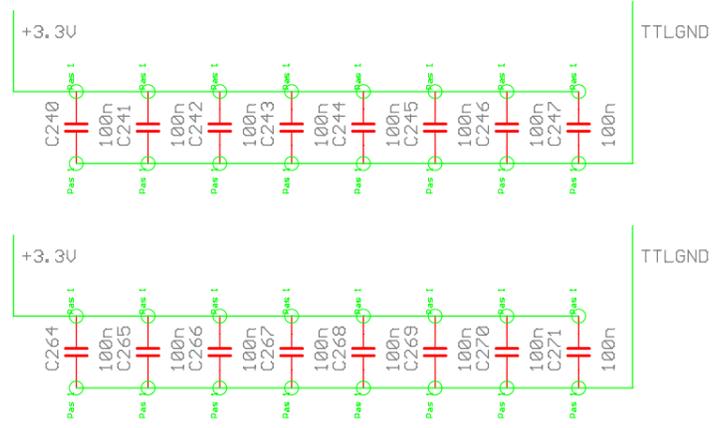
REV:

Date: 09.10.2001 19:08:22

Sheet: 6/8



TITLE: FADC-Motherboard	
Document Number:	REV:
Date: 09.10.2001 19:08:22	Sheet: 7/8



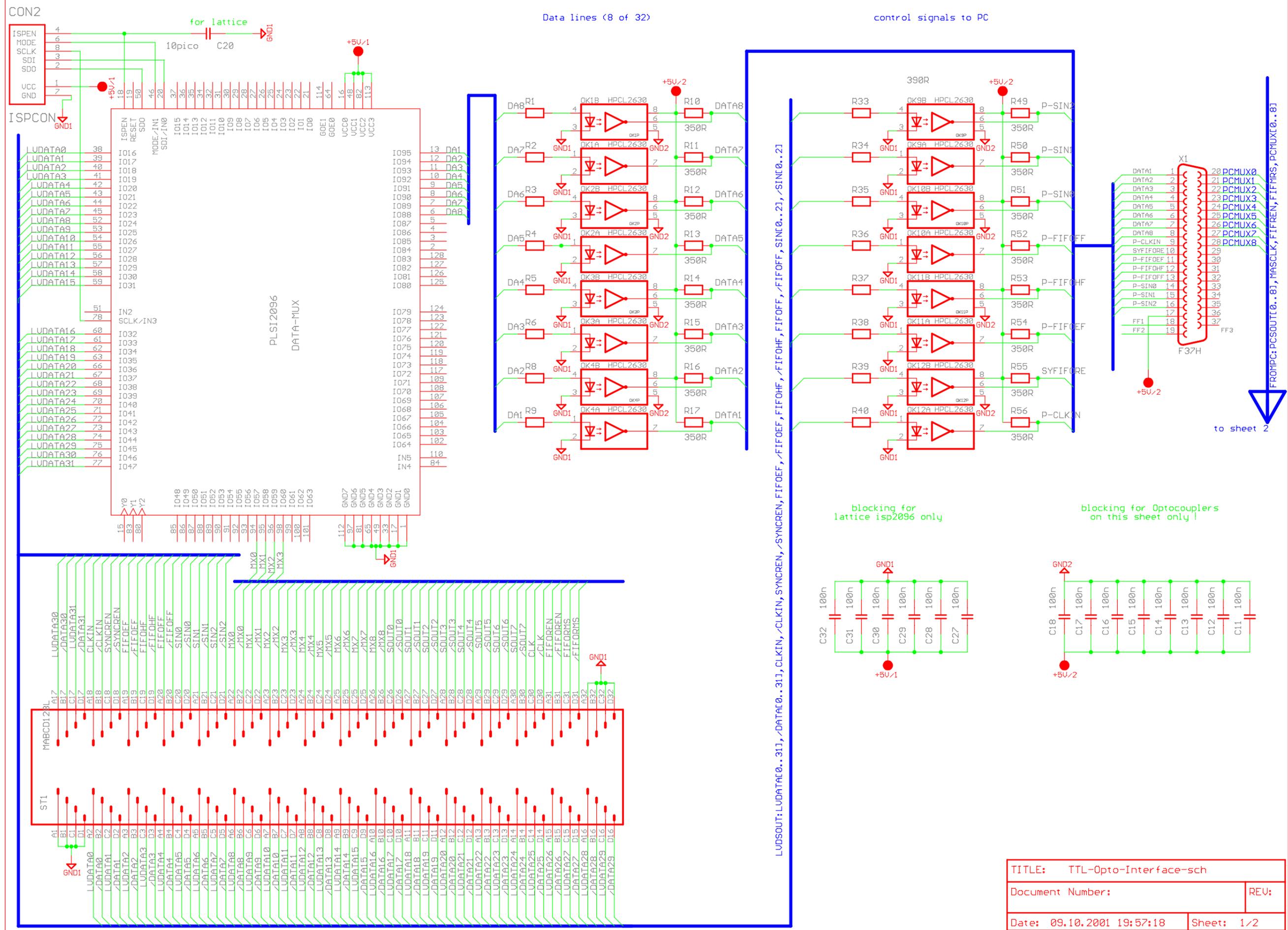
TITLE: FADC-Motherboard

Document Number:

REV:

Date: 09.10.2001 19:08:22

Sheet: 8/8

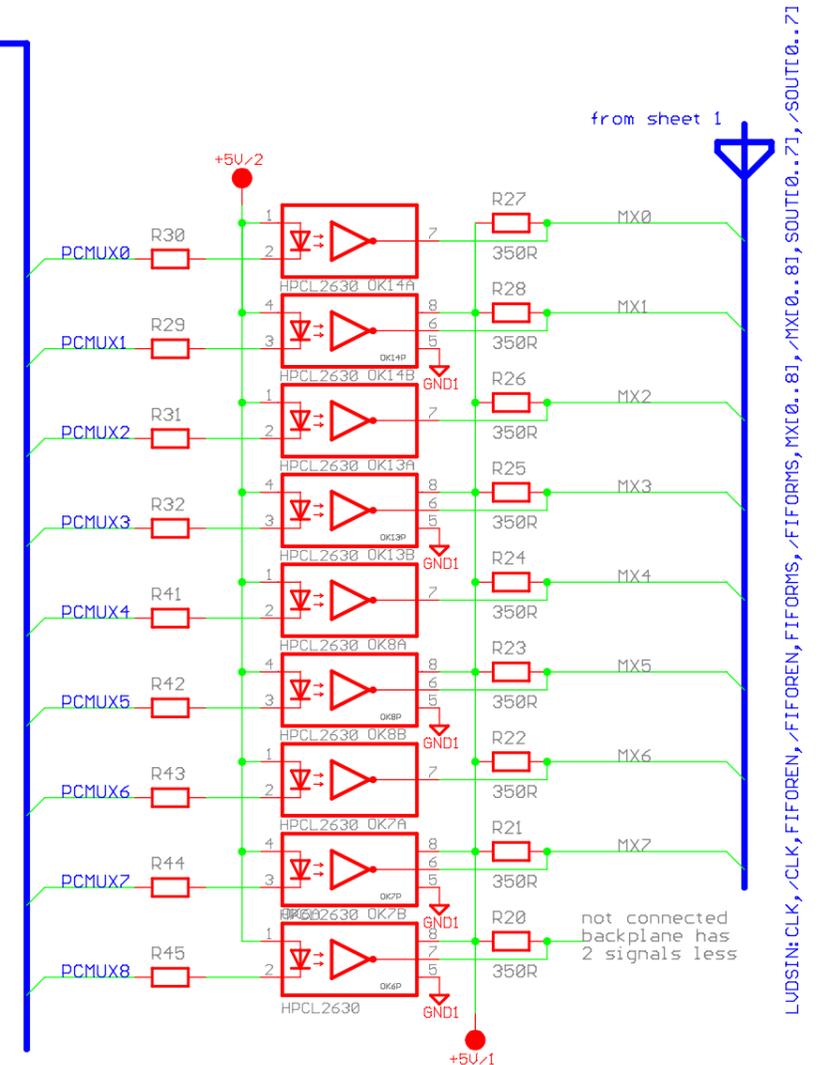
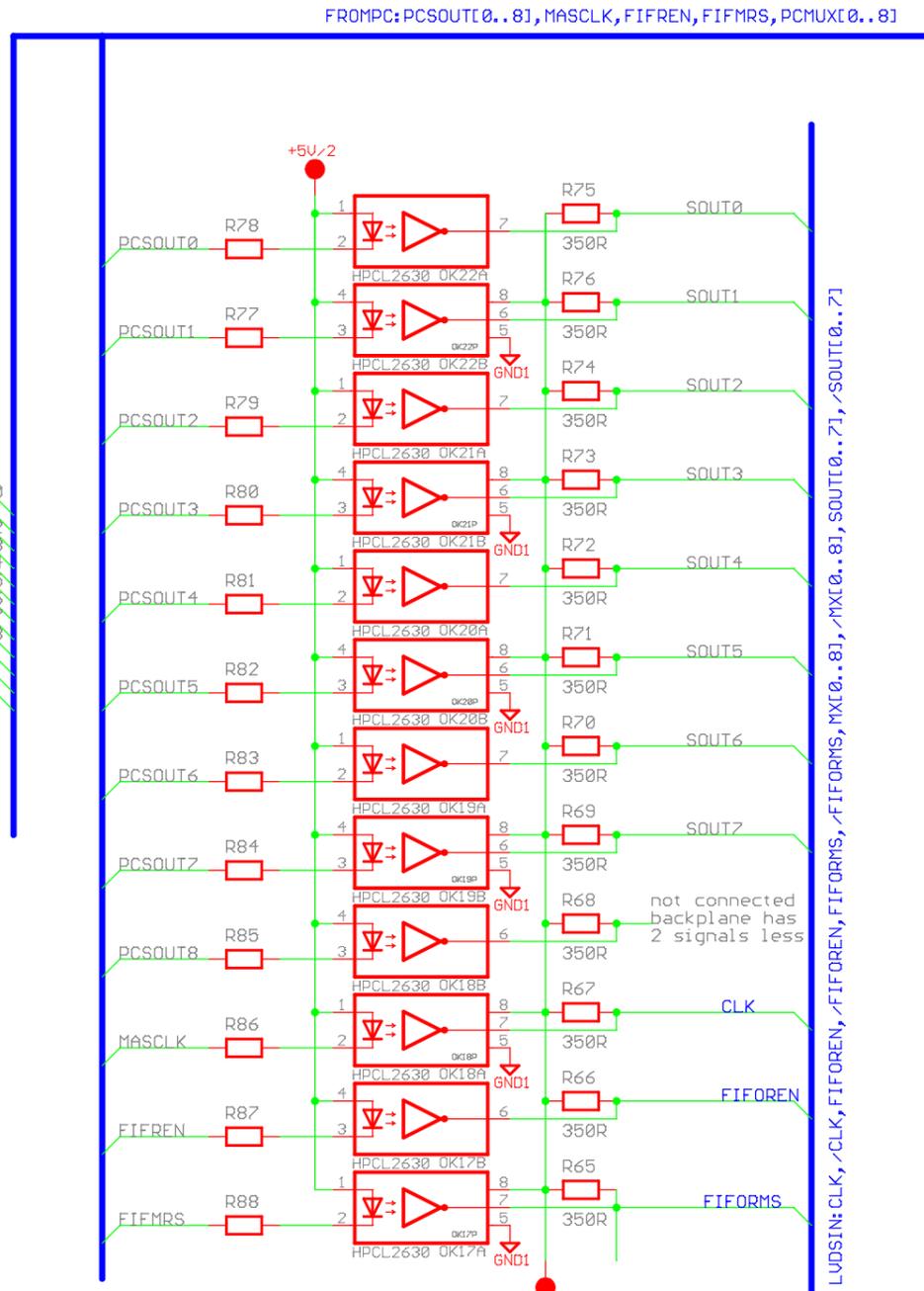
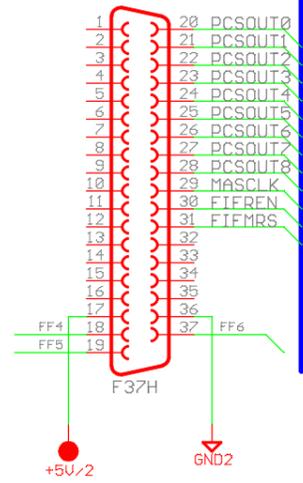
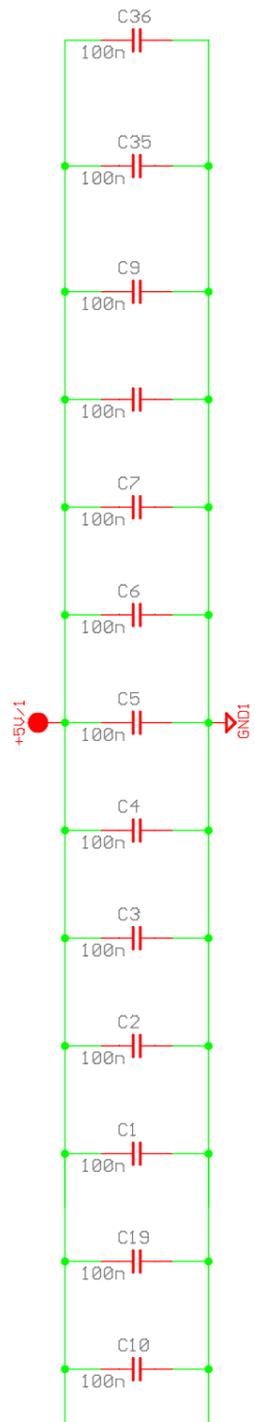


TITLE: TTL-Opto-Interface-sch	
Document Number:	REV:
Date: 09.10.2001 19:57:18	Sheet: 1/2

FROM PC: PCSOUT[0..8], MASCLK, FIFREN, FIFRIS, PCMUX[0..8]

to sheet 2

blocking for
optocouplers
on this sheet only



LUDSIN:CLK, /CLK, FIFOREN, /FIFOREN, FIFORMS, /FIFORMS, MX0..8, /MX0..8, SOUT0..7, /SOUT0..7

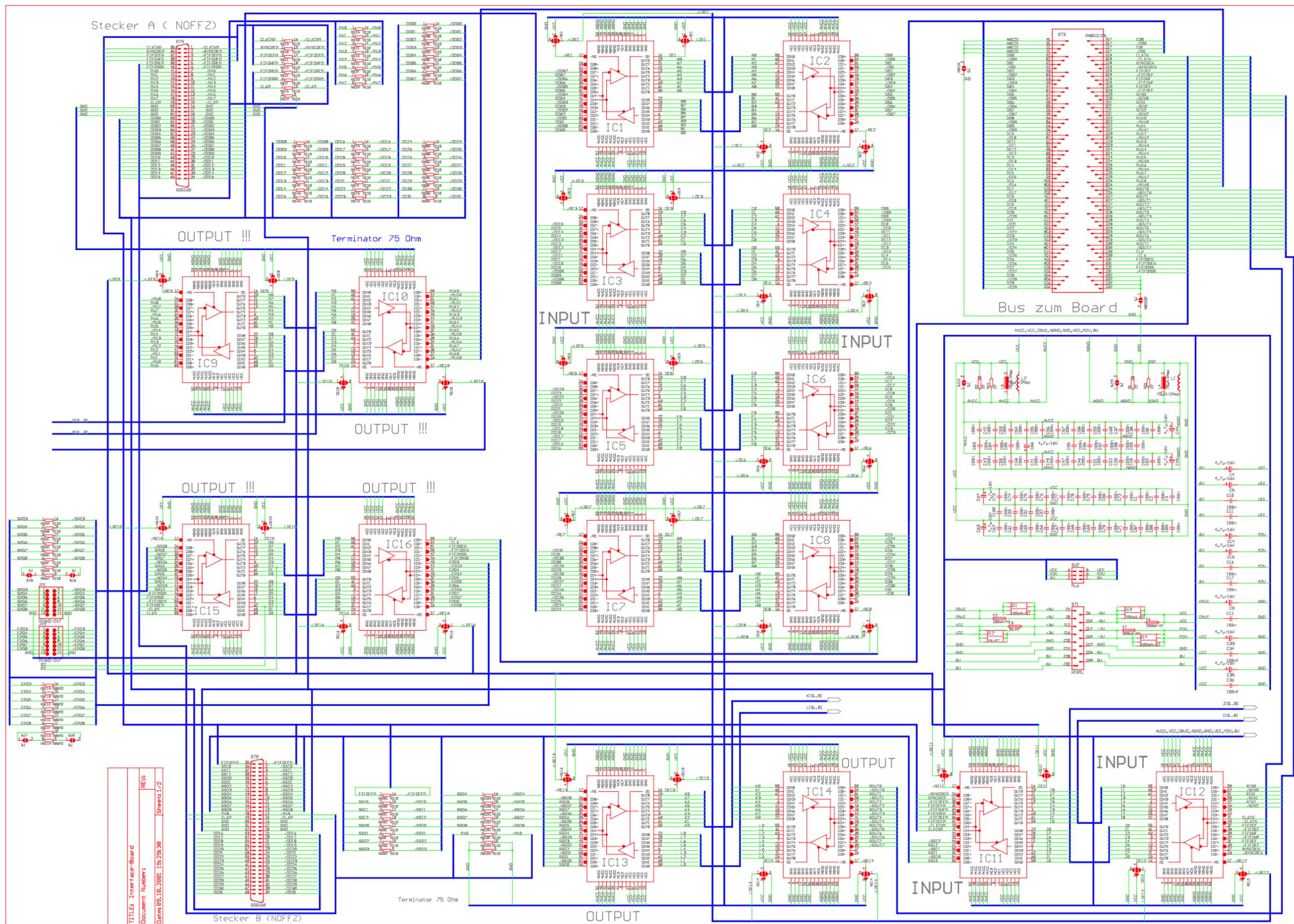
TITLE: TTL-Opto-Interface-sch

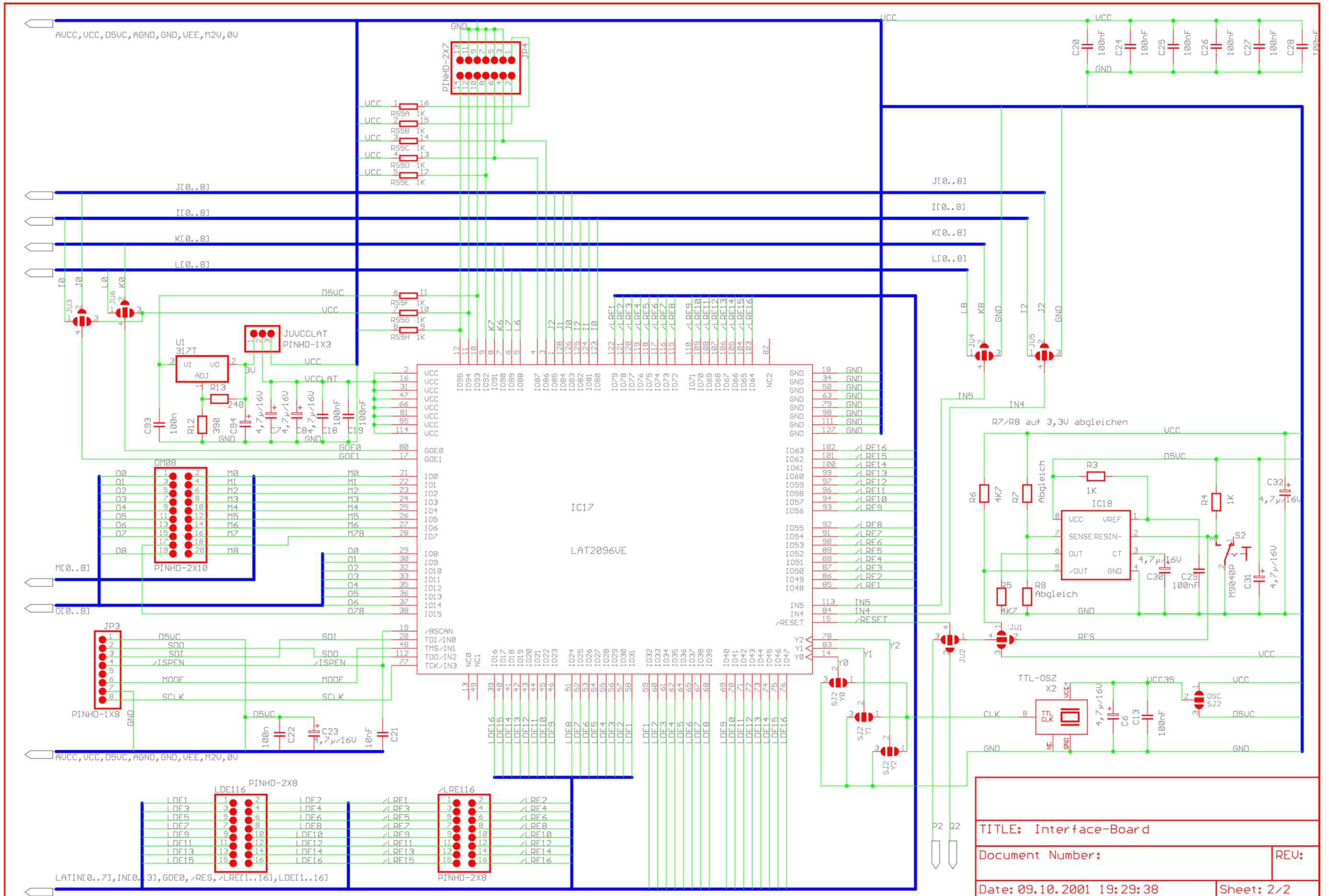
Document Number:

REV:

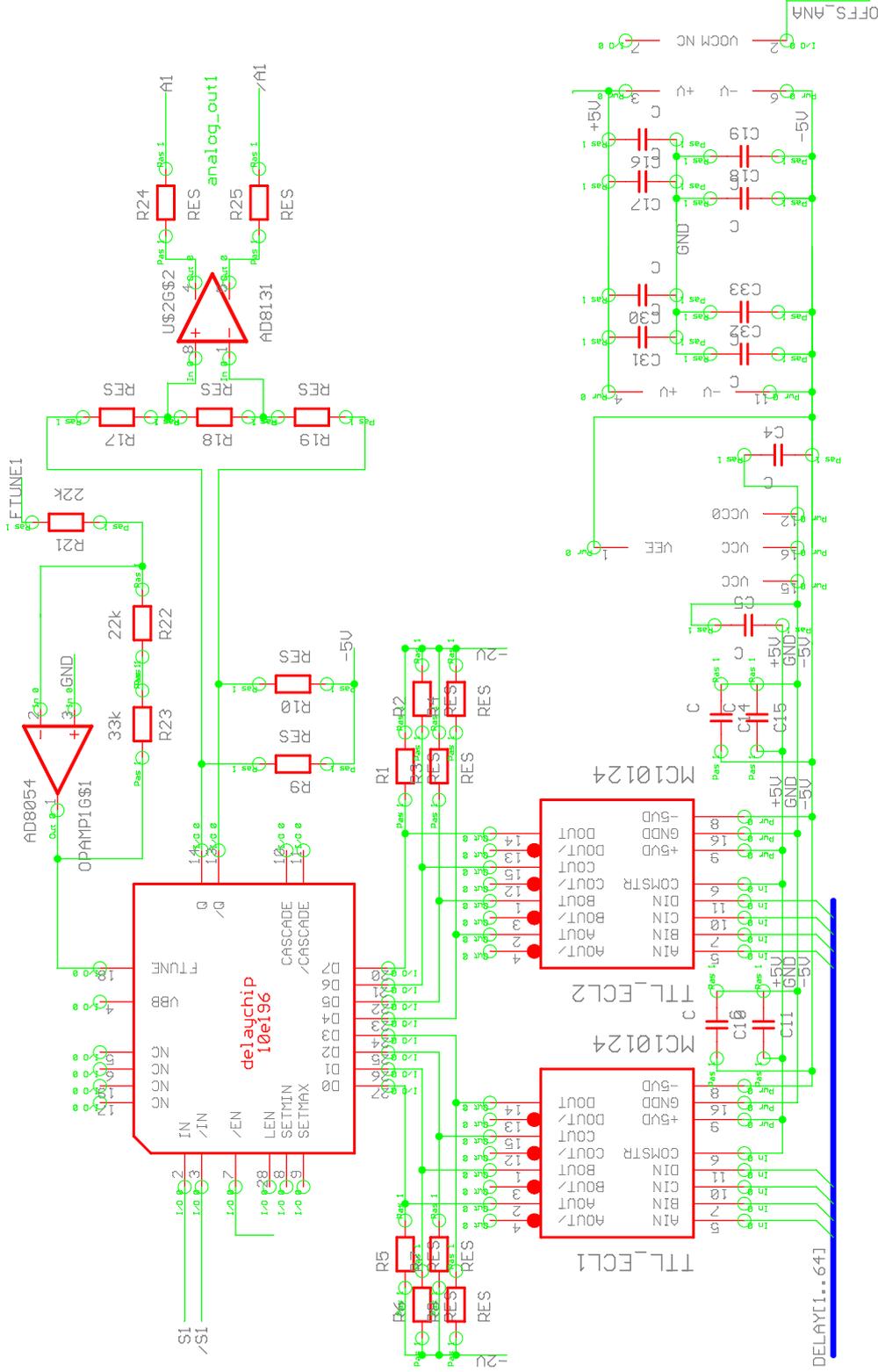
Date: 09.10.2001 19:57:18

Sheet: 2/2





TITLE: Interface-Board
 Document Number: _____ REV: _____
 Date: 09.10.2001 19:29:38 Sheet: 2/2



unit 1/8

TITLE: delayboard

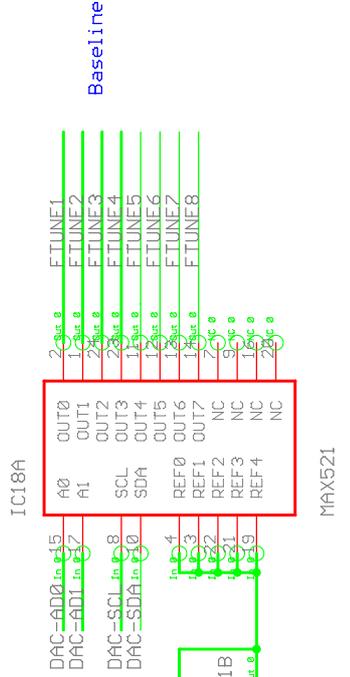
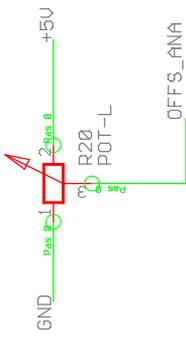
Document Number:

REV:

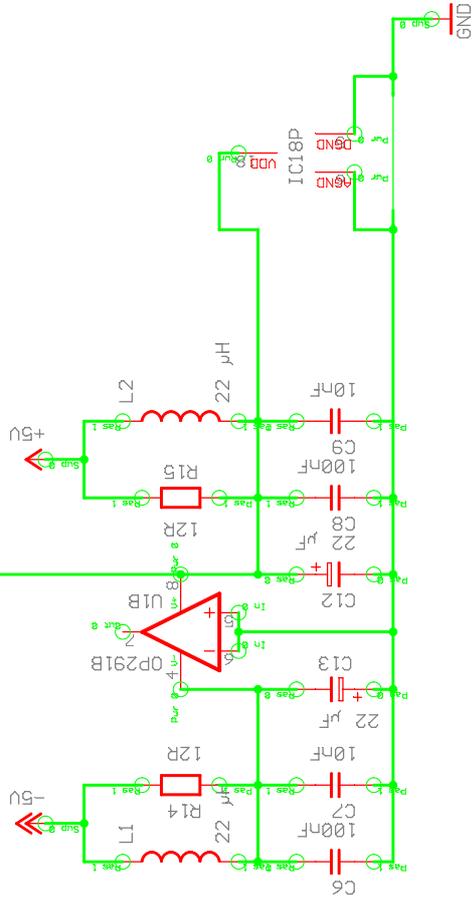
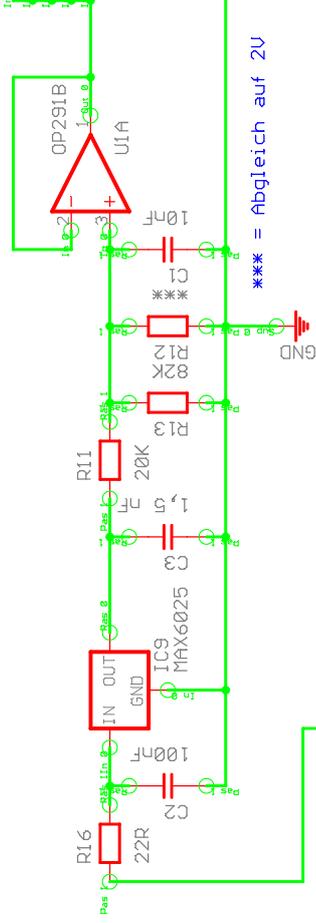
Date: 18.09.2000 14:48:06

Sheet: 1/11

analogue offset (to AD8131)



vom Lattice



DAC for 196 Fine Tuning

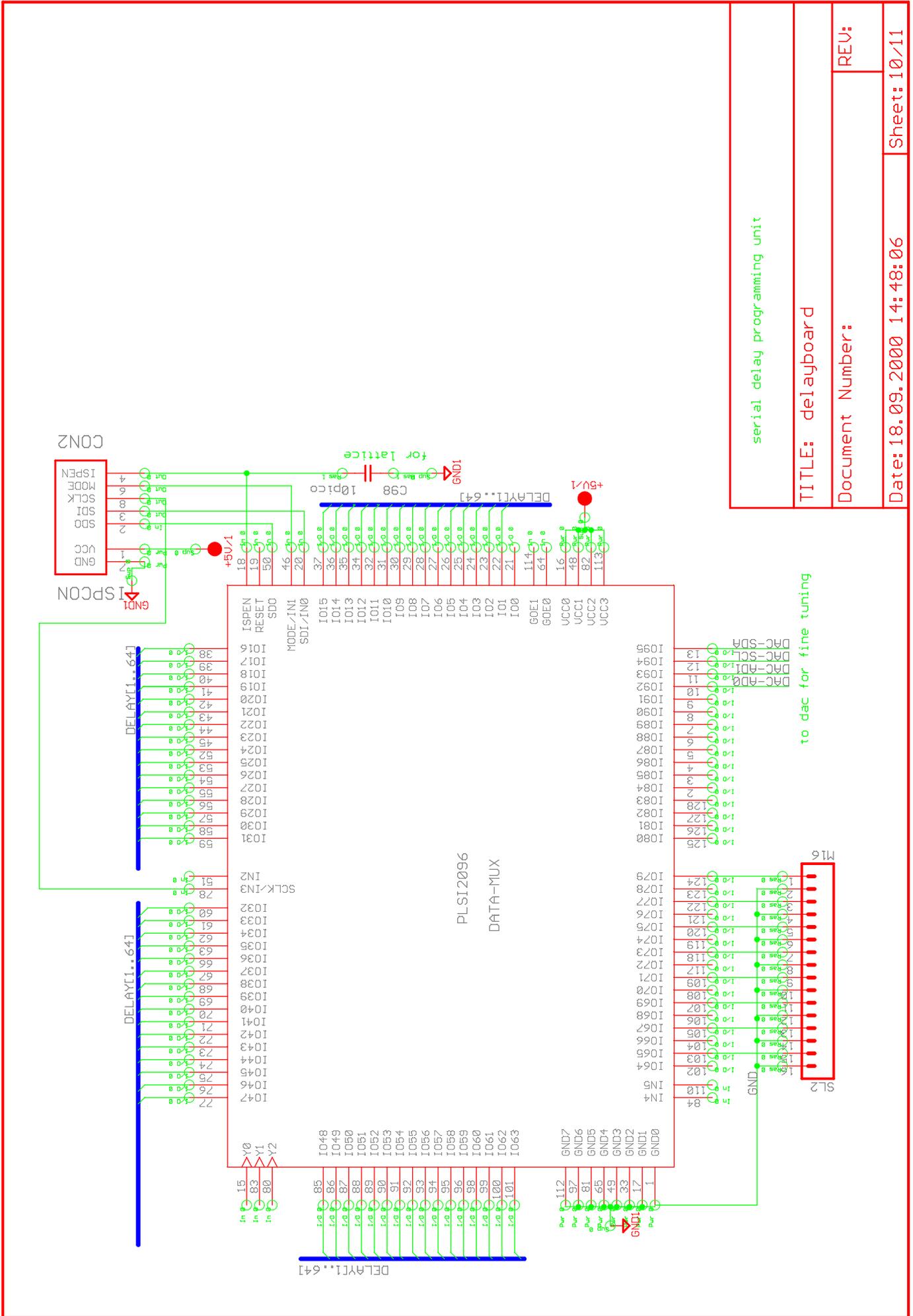
TITLE: delayboard

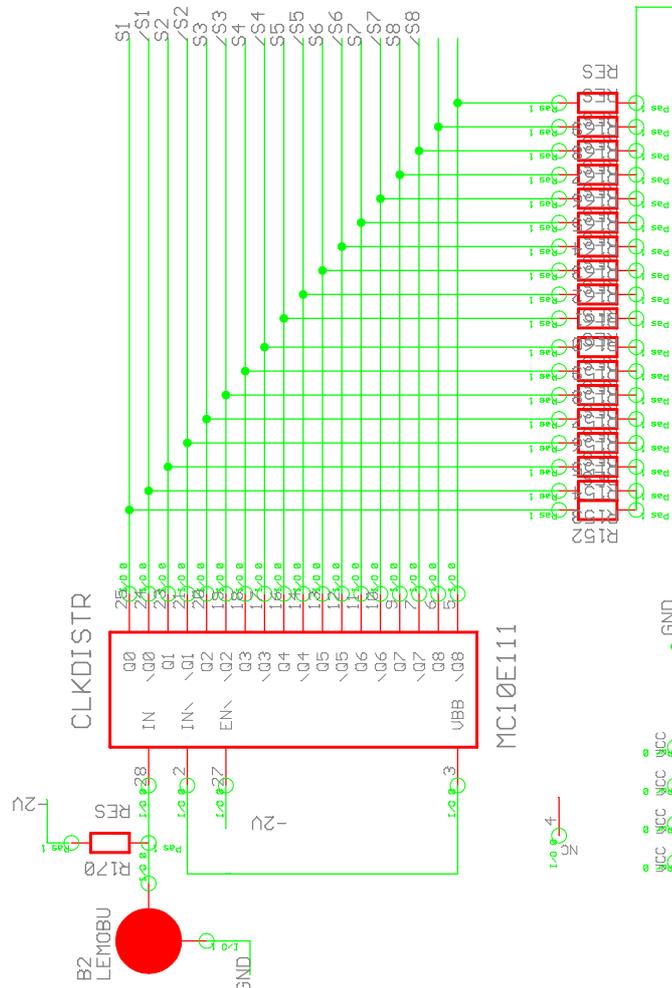
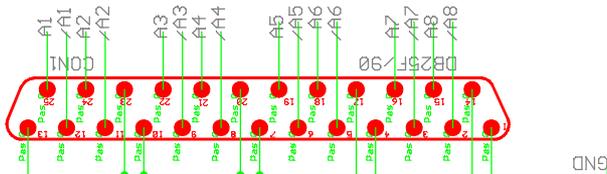
Document Number:

REV:

Date: 18.09.2000 14:48:06

Sheet: 2/11





clock distribution
analogue outputs
Power

TITLE: delayboard

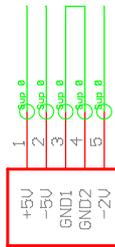
Document Number:

REV:

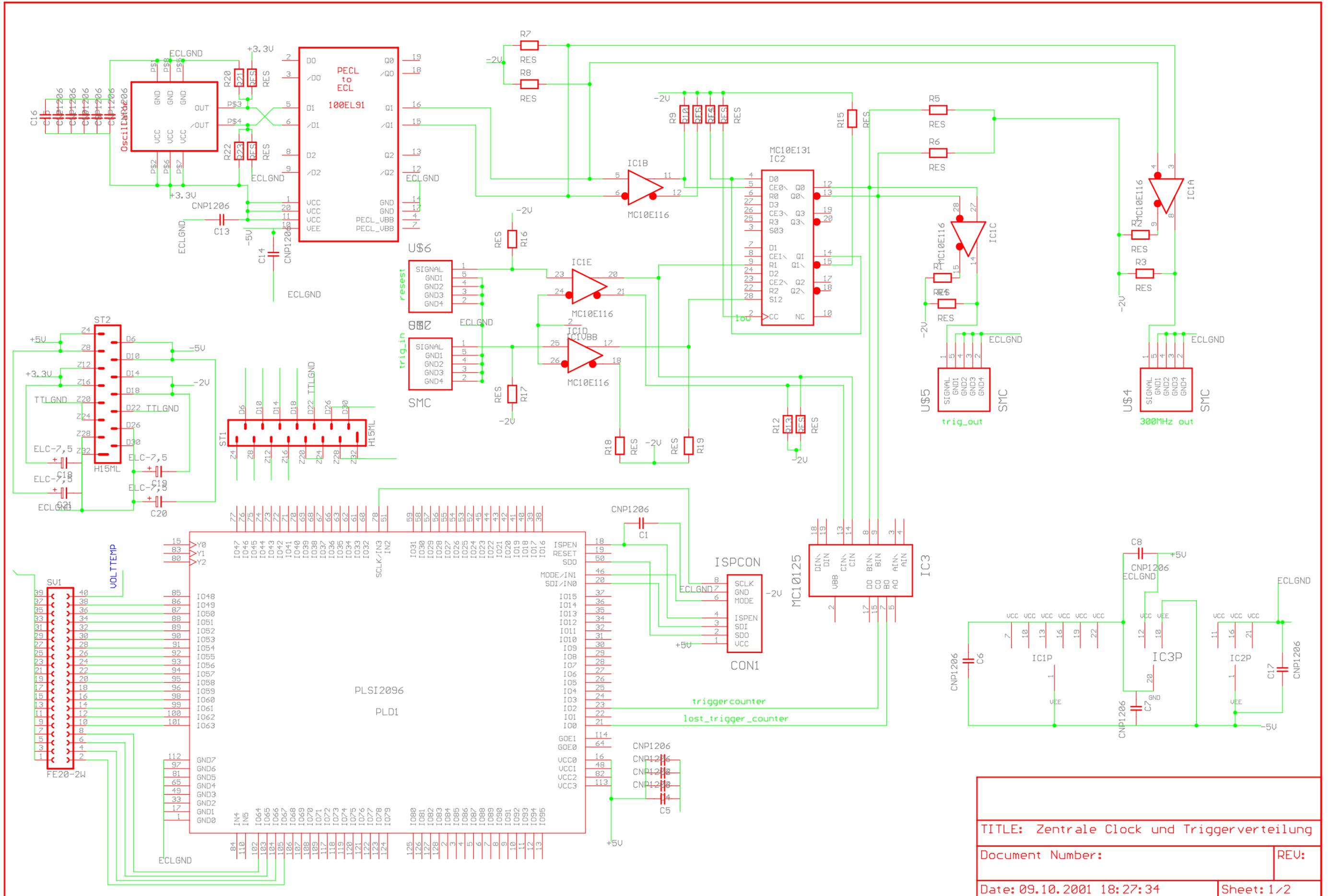
Date: 18.09.2000 14:48:06

Sheet: 11/11

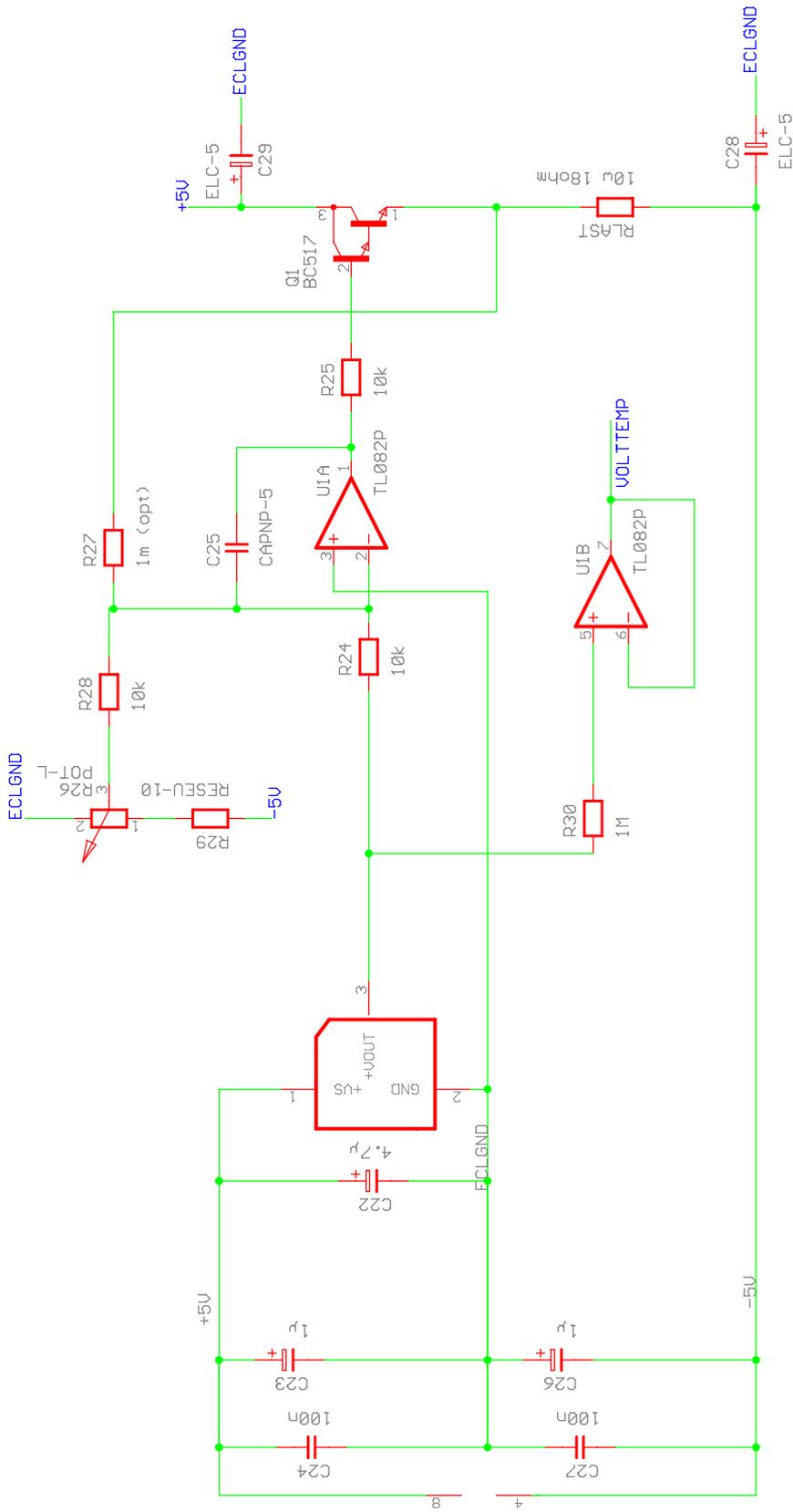
CON3



POWERCON



TITLE: Zentrale Clock und Triggerverteilung	
Document Number:	REV:
Date: 09.10.2001 18:27:34	Sheet: 1/2



TITLE: Zentrale Clock und Triggerverteilung

Document Number: REV:

Date: 09.10.2001 18:27:34 Sheet: 2/2

Anhang D :C-Quell-Datei zur Auslese mit PCI-I/O-Karte

Zweck : Initialisierung, Auslese und Tests der FADC-Motherboards im Teststand

fiforeadout.cpp

```

#include "StdAfx.h"
#include "7200.h"
#include "grey.h"
#include <conio.h>
#include <stdio.h>

// full fadc sampling routine
// pin connection:
//      DI 00 ... DI 07   : 8 bit data bus board 1
//      DI 10             : FIFO_EF
//      DI 11             : FIFO_HF
//      DI 12             : FIFO_FF
//      DI 08 ... DI 15   : unused
//      DI 16 ... DI 23   : unused
//      DI 24 ... DI 31   : unused
//      D0 00             : S0
//      D0 01             : S1
//      D0 02             : TMUX0
//      D0 16             : data_clk (SOUT0)
//      D0 17             : serial_data_in (SOUT1)
//      D0 20             :
//      D0 22             : MUX 3
//      D0 23             : MUX 2
//      D0 24             : MUX 1
//      D0 25             : MASCLK
//      D0 26             : FIFO_REN
//      D0 27             : FIFO_MRS
//      D0 28             :
//      D0 29             :
//      D0 30             :
//      D0 31             :
//      D0 21             :

#define channel_size      4           // no of channels to read out
#define event_size        100
// Anzahl auszulesender Events ( 1 Trigger => nr_channels* Samplepoints ==1 Event)
#define delay             4
#define signalsize        50
#define channels_on_board 8           // number of ADC board on one FIFO board
#define events_transferred 10

```

```

U32 TDIN, TDCLK, TFIFO_REN, TFIFO_Sync_REN, TCLKIN, TRCLK, TFIFO_EF, TFIFO_MRS, SDA,
SCL, S0, S1, TMUX;
U32 bit[32], SS[4], val, dummy;
U16 base_addr;
U16 outp_addr, inp_addr;
U8 irq, MAX_data[2*channels_on_board+1];

```

Fortsetzung nächste Seite

```

//U8 GreytoBinary(U8 grey);

void ShiftDataIn(void);
int BaseLineProgrammer(void);

void main()
{
FILE *output_file;
U8 data, baseline;
int i, k, select;

for (k=0;k<32;k++) bit[k]=0x00000001<<k;

    TFIFO_REN          = bit[26];
    TFIFO_Sync_REN    = bit[9];
    TCLKIN             = bit[8];
    TRCLK              = bit[25];
    TFIFO_EF           = bit[10]; // we use /OR signal from FIFO
    TFIFO_MRS          = bit[27];
    TDCLK              = bit[16]; // TSOUT0 CLK for NSP OFFS
    TDIN               = bit[17]; // TSOUT1 NSP OFFS
    SCL                = bit[18]; // TSOUT2 Baseline
    SDA                = bit[19]; // TSOUT3 Baseline
    MAX_data[0]        = 80; //address of MAX circuit 80=%01010000
//put the address of each DAC
for(k=0;k<8;k++)
    MAX_data[2*k+1]    = k;

//the base line of each DAC board
// High value => high DAC-Voltage => LOW value

//printf("Input baseline value:");scanf("%d",&baseline);
baseline=128;
MAX_data[2]           = baseline;
MAX_data[4]           = baseline;
MAX_data[6]           = baseline;
MAX_data[8]           = baseline;
MAX_data[10]          = baseline;
MAX_data[12]          = baseline;
MAX_data[14]          = baseline;
MAX_data[16]          = baseline;

    TMUX = bit[2];
    S0   = bit[0];
    S1   = bit[1];

    SS[0]=0; SS[1]=S0; SS[2]=S1; SS[3]=S0|S1;

```

Fortsetzung nächste Seite

```
//open the output file
output_file=fopen("events.dat","w");

// init PCI interface
W_7200_Initial(0,&base_addr,&irq);

// get out and in address from base address
outp_addr=base_addr+0x14;
inp_addr =base_addr+0x10;

//usually FIFO_MRS and TFIFO_REN are not active
//val=TFIFO_MRS+TFIFO_REN;
_outpd(outp_addr,val|SCL|SDA);

//call function for MAX circuit programming
printf("BASELINE programming \n \n");
BaseLineProgrammer();
val=(SCL|SDA); //keep the ICL bus on "1" logic

//send offset and sample points number
printf("\nSend NSP and OFFSET.\n");
ShiftDataIn();
val&=~TDCLK; val&=~TDIN; //keep the data and clock line on "0" logic

printf("FIFO Master Reset Cycle.\n");
//Master Reset Cycle for FIFO's
_outpd(outp_addr,val|TFIFO_MRS|TFIFO_REN);
_outpd(outp_addr,val|TFIFO_REN);
val=(TFIFO_MRS|TFIFO_REN);

//while(1) {printf("writing into FIFO\n");}

printf("Select the board via MUX signals.\n");
//select first FIFO group
_outpd(outp_addr,(val|=TMUX));
printf("Start the Read Cycle by TFIFO_REN=0.\n");

for(k=0;k<3;k++)
{
_outpd(outp_addr,(val&=~TFIFO_REN));
_outpd(outp_addr,(val|=TFIFO_REN));
_outpd(outp_addr,val|=TRCLK);
_outpd(outp_addr,val&=~TRCLK);
}
```

Fortsetzung nächste Seite

```

//printf("Start the Read Cycle by TFIFO_REN=0.\n");
//send read enable to FIFO, FIFO_REN is active on zero
//while(1){printf(" fifo status: %u\n",(_inpd(inp_addr)&TFIFO_EF));}

_outpd(outp_addr,(val&=~TFIFO_REN));

for(i=0;i<events_transfered;i++)
{
    //read one event
    for(k=0;k<event_size;k++)

//read out procedure from one group FIFO's : 1 sample read
{

//printf("check if OUTPUT READY is 0.\n");
//check if the /OUTPUT READY is active
_outpd(outp_addr,(val|=TFIFO_REN));
    while((_inpd(inp_addr)&TFIFO_EF)!=0)
    {
        _outpd(outp_addr,(val|TRCLK));
        _outpd(outp_addr,val);
    }
_outpd(outp_addr,(val&=~TFIFO_REN));

printf("Send Read Clock to FIFO.\n");
//send to FIFO PC_read_CLK
_outpd(outp_addr,(val|TRCLK));
_outpd(outp_addr,val);

//wait for CLKIN to come, this can be done only hardware
/*    while(_inpd(inp_addr)&CLKIN); */
//read data from all 4 channels
for(select=0;select<channel_size;select++)
{
    _outpd(outp_addr,(val|SS[select]));
//
    data=(U8)(_inpd(inp_addr)<<0);
    data= ~(U8)(_inpd(inp_addr));

    data= (GreyToBinary(data));
    fprintf(output_file,"%d\t", data);

}
fprintf(output_file,"\n");
}

}

```

Fortsetzung nächste Seite

```
printf("Disable TFIFO_REN.\n");
//send read disenable to FIFO
_outpd(outp_addr,val|=TFIFO_REN);
//disable board select
_outpd(outp_addr,val&=~TMUX);

printf("%d event(s) stored.\n",i);
//getch();
fclose(output_file);
}

/*
U8 GreytoBinary(U8 grey)
{
    U8 SetBit[8]    = {1,2,4,8,16,32,64,128};
    U8 ClearBit[8] = {254,253,251,247,239,223,191,127};

    U8 tbin;
    int i;

    tbin=(grey & SetBit[7]);

    for (i=6; i>=0; i--)
    {
        if ((grey & SetBit[i])>0)
        {
            if ((tbin & SetBit[i+1])==0)
            {
                tbin=(tbin | SetBit[i]);
            }
            else
            {
                tbin=(tbin & ClearBit[i]);
            }
        }
        else
        {
            if ((tbin & SetBit[i+1])>0)
            {
                tbin=(tbin | SetBit[i]);
            }
            else
            {
                tbin=(tbin & ClearBit[i]);
            }
        }
    }
    return(tbin);
}
*/
```

Fortsetzung nächste Seite

```

void ShiftDataIn(void)
{
    int i;    U32 qa,qb;

    qa=delay;
    qb=(signalsize-1);

    _outpd(outp_addr,val&(~TDCLK)); //TDCLK auf 0 um def. Anfangszustand zu haben

    printf(" OFFS: ");

    for(i=12;i>=0;i--)
    {
        if ((qa>>i)&0x01){ val=val|TDIN ;_outpd(outp_addr,val); }
        else {val=val&(~TDIN); _outpd(outp_addr,val); }

        _outpd(outp_addr,val|TDCLK);
        _outpd(outp_addr,val&(~TDCLK));
    }

    printf("\n NSP: \n");
    for(i=11;i>=0;i--)
    {
        if ((qb>>i)&0x01)    _outpd(outp_addr,val|=TDIN);
        else {
            _outpd(outp_addr,val&=~(TDIN)); }

        _outpd(outp_addr,val|=TDCLK);
        _outpd(outp_addr,val&=~(TDCLK));
    }
}

int BaseLineProgrammer(void)
{
    //function to program the MAX circuit
    int i,k,send_data;
    bool ok=1; //control variable for ACK signal from MAX circuit
    U8 bit_mask[8];

    for (i=0;i<8;i++)
    {
        bit_mask[i]=1<<i;
        //printf("bitmask-%d=%d\n",i,bit_mask[i]);
    }
    //start condition
    _outpd(outp_addr,val|SCL|SDA);
    _outpd(outp_addr,(val|SCL)&(~SDA));

    //put serial clock to zero
    _outpd(outp_addr,val&(~SCL));
}

```

Fortsetzung nächste Seite

```
//send 1+2xNr_of_dacs bytes to the MAX; 1 adress byte and 2 times 8 (nr of DAC's) command bytes
for(k=0;k<17;k++)
{
    //printf("send_data=%d\n",MAX_data[k]);
    send_data=MAX_data[k];
    for(i=0;i<8;i++)
    {
        if ((send_data&bit_mask[7-i])==0)
        {
            //printf("0");
            _outpd(outp_addr,val&(~SDA)&(~SCL));
            //send the clock pulse
            _outpd(outp_addr,(val|SCL)&(~SDA));
            _outpd(outp_addr,val&(~SCL)&(~SDA));
        }
        else
        {
            //printf("1");
            _outpd(outp_addr,(val|SDA)&(~SCL));
            //send the clock pulse
            _outpd(outp_addr,val|SCL|SDA);
            _outpd(outp_addr,(val|SDA)&(~SCL));
        }
    }
    //the 9th pulse and verify if is zero
    _outpd(outp_addr,(val|SCL)&(~SDA));
    if (SDA&(_inpd(inp_addr))!=0)
        ok=0;
    _outpd(outp_addr,val&(~SCL)&(~SDA));
    //printf("\n");
}

//stop conditon
_outpd(outp_addr,(val|SCL)&(~SDA));
_outpd(outp_addr,val|SDA|SCL);
.. //printf("ok=%d\n",ok);
//while(1) {};
return ok;
}
```

Danksagung

Ich bedanke mich bei Prof. Pavel und Prof. Schwarte für die Themenstellung und die gute Betreuung dieser Arbeit und für die Hilfestellung bei der Prüfungsvorbereitung. Durch Prof. Walenta und Dr. Besch wurde schon seit meinen späten Studienjahren die notwendige Basis für diese Dissertation gelegt, für die gute Betreuung insbesondere in der Anfangsphase meiner Tätigkeit als Doktorand bedanke ich mich recht herzlich.

Ich bedanke mich bei all denjenigen, die mit viel Geist und Arbeitseinsatz zum Gelingen der Elektronik beigetragen haben, insbesondere bei Martin Adamek und Adrian Alexandrescu. Auch den Herren Dipl.Ing. Sergei Volkov und Werner Marschik sei in diesem Zusammenhang gedankt.

Ich bedanke mich bei Herrn Dieter Junge und Alfred Peper für die Übernahme vieler kleiner Arbeiten und bei der Elektronikwerkstatt für die freundschaftliche Zusammenarbeit. Insbesondere Herrn Dipl.Ing. Rudi Seibert sei für sein dort einzigartiges Engagement gedankt.

Ich bedanke mich auch bei den anderen Mitgliedern der Walenta/Pavel-Gruppe, insbesondere den Herren Dipl.Phys. Wolf Meissner, Hendrik Wagner, Reinhard Langer, Andre Orthen, Thomas Hengstebeck, Carsten Strietzel und Norbert Sauer für das gute Arbeitsklima.

Dank für eine stets gute Zusammenarbeit geht auch an die Mitarbeiter der MAGIC-Kollaboration, insbesondere an Dr. Eckart Lorenz, Dr. Juan Cortina und Dr. Toni Coarasa vom Max-Planck-Institut in München und Herrn Dr. Rudi Böck am CERN.

Ich danke meiner Frau und meinen Kindern für ihre Geduld mit einem vielbeschäftigten Mann und Vater und dafür, mir stets auch die Augen für die anderen wichtigen Dinge des Lebens zu öffnen. Meinen Eltern danke ich dafür, dass sie meine privaten und beruflichen Entscheidungen stets akzeptiert und meine Eigenständigkeit in allen Bereichen und Lebenslagen immer gefördert und unterstützt haben. Ohne sie wäre es nicht möglich gewesen, diesen Weg zu gehen.