

Entwicklung eines digitalen Bildspeichers mit
Speicherplatzoptimierung unter Verwendung von
Standard-DRAMs für ein Video-Schnitt-System

Diplomarbeit

im Studiengang Technische Informatik

an der

Fachhochschule Köln

Abteilung Gummersbach

Prüfer:

Prof. Dipl.-Phys. T.C.E. Drescher

Prof. Dipl.-Ing. H. Scheidt

Labor-Ing.:

Dipl.-Ing. K. Bock

vorgelegt von

Jürgen Loh

Matrikelnummer: 102994

Gummersbach, 1.März 1990

I n h a l t s v e r z e i c h n i s

1. <u>Abkürzungsverzeichnis</u>	1-1
2. <u>Einleitung</u>	2-1
1. Zielsetzung	2-1
2. Arbeitsumgebung	2-1
3. Konzeption	2-2
1. Auflösung	2-2
2. Adressierungsschema	2-3
3. <u>Technologien</u>	3-1
1. Logic Cell Array (LCA)	3-1
1. Configurable Logic Block (CLB)	3-1
2. Input/Output Block (IOB)	3-2
3. Verdrahtungskanäle	3-2
4. Entwicklungswerkzeug PC	3-2
2. Entwicklungswerkzeug Workstation	3-3
1. Symbole und Makros	3-3
2. Hierarchischer Schaltplan	3-3
3. Leitungen und Busse	3-4
3. In-Circuit-Emulator	3-5
4. Wire-Wrap-Technik	3-5
4. <u>Schaltungsbeschreibung</u>	4-1
1. Übersicht	4-1
2. Betriebsmodis	4-1
3. A/D-Wandler (ADC)	4-2
4. Ablauf- und D-RAM-Steuerung (CLK2)	4-3
1. Top-Level (CLK2/1)	4-3
1. Eingangssignale	4-3
2. Ausgangssignale	4-4
2. Prinzip der RAS/CAS-Erzeugung	4-4
3. RAS/CAS-Erzeugung (CLK2/LOGIC/1)	4-7
4. Differenzierer und CAS-Zähler (CLK2/LOGIC/2)	4-7
5. Aufnahmesteuerung und Refresh-timer (CLK2/LOGIC/3)	4-8
6. CLK2 - Zeitdiagramme	4-8
5. Erzeugung der D-RAM-Adressen (ADRE)	4-11
1. Top-Level (ADRE/1)	4-11
1. Eingangssignale	4-11
2. Ausgangssignale	4-11
2. ROW-Zähler (ADRE/LOGIC/1)	4-11
3. COL-Zähler (ADRE/LOGIC/2)	4-12
4. Adreßmultiplexer (ADRE/LOGIC/3)	4-13
5. ADRE - Zeitdiagramme	4-13
6. Erzeugung der Video-Synchronimpule (SYNC)	4-15
1. Top-Level (SYNC/1)	4-15
1. Eingangssignale	4-15
2. Ausgangssignale	4-15
2. Horizontalsteuerung (SYNC/LOGIC/1)	4-16
3. Vertikalsteuerung (SYNC/LOGIC/2)	4-17
4. CSync-Erzeugung (SYNC/LOGIC/3)	4-18
5. Aufnahmesteuerung (SYNC/LOGIC/4)	4-19
6. SYNC - Zeitdiagramme	4-19

7. Datenfluß- und CPU-Steuerung (PROZ)	4-20
1. Top-Level (PROZ/1)	4-20
1. Datenbusse	4-20
2. Eingangssignale	4-20
3. Ausgangssignale	4-21
2. Datenflußsteuerung (PROZ/LOGIC/1)	4-21
3. Datenmultiplexer (PROZ/LOGIC/2)	4-21
4. Interrupt-Steuerung (PROZ/LOGIC/3)	4-22
5. PROZ - Zeitdiagramme	4-23
8. Dynamischer Speicher (DRAM)	4-23
9. Mikroprozessor (CPU)	4-24
10. D/A-Wandler (DAC)	4-24
 5. <u>Programmbeschreibung</u>	5-1
1. Verwendete Hardware	5-1
2. Bedienung	5-1
 6. <u>Ungelöste Probleme</u>	6-1
1. Modus-Umschaltungen	6-1
2. Kritisches Timing	6-1
3. Aufnahmesteuerung	6-3
4. Umrahmung	6-3
 7. <u>Zusammenfassung</u>	7-1
1. Einsatzgebiete	7-1
2. Ausblick	7-1

Anhang

A. <u>Schaltpläne</u>	A-1
1. Bildspeicher, Top-Level (EDDI3)	A-2
2. Ablaufsteuerung (CLK2)	A-3
3. Adreßerzeugung (ADRE)	A-10
4. Video-Sync-Erzeugung (SYNC)	A-18
5. Prozessormodul, Datensteuerung (PROZ)	A-24
6. Sonstige Schaltpläne	A-28
B. <u>Zeitdiagramme</u>	B-1
1. CLK2 - Zeitdiagramme	B-2
2. ADRE - Zeitdiagramme	B-5
3. SYNC - Zeitdiagramme	B-8
4. PROZ - Zeitdiagramme	B-10
C. <u>LCA-Makrobibliothek</u>	C-1
1. Übersicht	C-1
2. Flip-Flops	C-3
1. RS-Flip-Flops	C-3
2. Toggle-Flip-Flops	C-4
3. Binärzähler	C-6
1. Binärzähler modulo 4	C-6
2. Binärzähler modulo 8	C-8
3. Binärzähler modulo 16	C-11
4. Binärzähler modulo 256	C-14
4. I/O-Pads	C-16
5. Addierer/Subtrahierer	C-16
D. <u>Flußdiagramme</u>	D-1
1. Flußdiagramme zu START.S	D-1
2. Flußdiagramme zu IRQ.S	D-4
3. Flußdiagramme zu TAST.S	D-5
4. Flußdiagramme zu UNTER.S	D-10
E. <u>Programmlisting</u>	E-1
1. Bibliothek (BIBL.S)	E-1
2. Initialisierung und Hauptprogramm (START.S)	E-5
3. Interrupt-Behandlung (IRQ.S)	E-10
4. Tastatur- und Anzeigesteuerung (TAST.S)	E-11
5. Unterprogramme (UNTER.S)	E-15
6. Low-Level Routinen (FREMD.S)	E-22
7. Referenztabelle	E-26
F. <u>Datenblätter</u>	F-1
1. UVC 3130: A/D-D/A-Wandler	F-1
2. XILINX XC3000-Serie, Zeitverhalten	F-5
3. uPD424256-12, 256K*4 Bit D-RAM	F-10
4. Video-Basisnormen der 625-Zeilen-Systeme	F-15
G. <u>Literaturverzeichnis</u>	G-1
H. <u>Sachwortverzeichnis</u>	H-1

Abbildungen

4-1: CLK2- Zustandsfolgediagramm.....	4-10
A-1: Übersichtsschaltplan (EDDI3/1).....	A-2
A-2: Ablaufsteuerung, Top-Level (CLK2/1).....	A-3
A-3: RAS/CAS-Erzeugung (CLK2/LOGIC/1).....	A-4
A-4: Differenzierer, CAS-Zähler (CLK2/LOGIC/2).....	A-5
A-5: Aufnahmesteuerung, Refresh-Timer (CLK2/LOGIC/3).....	A-6
A-6: Wartezyklus (CLK2/LOGIC/WAITZYK/1).....	A-7
A-7: RAS-Zyklus (CLK2/LOGIC/RASZYK/1).....	A-7
A-8: CAS-Zyklus (CLK2/LOGIC/CASZYK/1).....	A-8
A-9: Refresh-Zyklus (CLK2.LOGIC/RFSHZYK/1).....	A-9
A-10: Adreßerzeugung, Top-Level (ADRE/1).....	A-10
A-11: RAS-Zähler (ADRE/LOGIC/1).....	A-11
A-12: CAS-Zähler (ADRE/LOGIC/2).....	A-12
A-13: Adreßmultiplexer (ADRE/LOGIC/3).....	A-13
A-14: Addierer Nr.1 (ADRE/LOGIC/ADD_1/1).....	A-14
A-15: Addierer Nr.2 (ADRE/LOGIC/ADD_2/1).....	A-15
A-16: Multiplizierer * 24 (ADRE/LOGIC/MUL20/1).....	A-16
A-17: Multiplizierer * 80 (ADRE/LOGIC/MUL80/1).....	A-17
A-18: Sync-Erzeugung, Top-Level (SYNC/1).....	A-18
A-19: Horizontalsteuerung (SYNC/LOGIC/1).....	A-19
A-20: Vertikalsteuerung (SYNC/LOGIC/2).....	A-20
A-21: Composite-Sync-Erzeugung (SYNC/LOGIC/3).....	A-21
A-22: Aufnahmesteuerung (SYNC/LOGIC/4).....	A-22
A-23: Zeilenkomparator (SYNC/LOGIC/COMP9/1).....	A-23
A-24: Prozessormodul, Top-Level (PROZ/1).....	A-24
A-25: Datenwegesteuerung (PROZ/LOGIC/1).....	A-25
A-26: Datenmultiplexer (PROZ/LOGIC/2).....	A-26
A-27: Interruptsteuerung (PROZ/LOGIC/3).....	A-27
A-28: A/D- und D/A-Wandlerplatine (ADC/DAC).....	A-28
A-29: CPU-Platine.....	A-29
A-30: Anzeige- und Tastaturplatine.....	A-30
B-1: CLK2 - RAS/CAS-Erzeugung.....	B-2
B-2: CLK2 - Differenzierer.....	B-2
B-3: CLK2 - CPU-Modus (1).....	B-3
B-4: CLK2 - CPU-Modus (2).....	B-3
B-5: CLK2 - REC-Modus.....	B-4
B-6: ADRE - COL-Adreßerzeugung (1).....	B-5
B-7: ADRE - COL-Adreßerzeugung (2).....	B-5
B-8: ADRE - ROW-Adreßerzeugung.....	B-6
B-9: ADRE - Adreßmultiplexer.....	B-6
B-10: ADRE - CPU-Modus.....	B-7
B-11: SYNC - V1-Puls.....	B-8
B-12: SYNC - V2-Puls.....	B-8
B-13: SYNC - H-Puls, CSync-Erzeugung.....	B-9
B-14: PROZ - CPU-Modus.....	B-10

C-1: Übersicht über die LCA-Makrobibliothek.....	C-2
C-2: FCSR: Getaktetes RS-Flip-Flop, Vorrang Setzen.....	C-3
C-3: FCRS: Getaktetes RS-Flip-Flop, Vorrang Rücksetzen.....	C-3
C-4: FTCSR: Toggle-Flip-Flop, Synchron Reset.....	C-4
C-5: FTCRD: Toggle-Flip-Flop, Direct Reset.....	C-4
C-6: FTCPR: Ladbares Toggle-Flip-Flop, Synchron Reset.....	C-5
C-7: FTCPRD: Ladbares Toggle-Flip-Flop, Direct Reset.....	C-5
C-8: C4BCRX: Zähler modulo 4, Synchron Reset.....	C-6
C-9: BCRDX: Zähler modulo 4, Direct Reset.....	C-6
C-10: C4BCPRDX: Ladbarer Zähler modulo 4, Direct Reset.....	C-7
C-11: C8BCRX: Zähler modulo 8, Synchron Reset.....	C-8
C-12: C8BCPR: Ladbarer Zähler modulo 8, Synchron Reset.....	C-9
C-13: C8BCPRDX: Ladbarer Zähler modulo 8, Direct Reset.....	C-10
C-14: C16BCRX: Zähler modulo 16, Synchron Reset.....	C-11
C-15: C16BCPRX: Ladbarer Zähler modulo 16, Synchron Reset.....	C-12
C-16: C16BCPRDX: Ladbarer Zähler modulo 16, Direct Reset.....	C-13
C-17: C256BCRX: Zähler modulo 256, Synchron Reset.....	C-14
C-18: C256BCPRDX: Ladbarer Zähler modulo 256, Direct Reset.....	C-15
C-19: ILOZ: I/O-Pad mit Input-Latch und Tri-State-Output.....	C-16
C-20: HALF_ADD: Halbaddierer.....	C-16
C-21: ADDSUB2: 2-Bit-Addierer/Subtrahierer.....	C-17
C-22: ADDSUB4: 4-Bit-Addierer/Subtrahierer.....	C-17
C-23: ADDSUB8: 8-Bit-Addierer/Subtrahierer.....	C-18
D-1: STTA: Reset-Initialisierung.....	D-1
D-2: BFUELL: Bildspeicher mit Testdaten füllen.....	D-2
D-3: MKZEIL: Testbildzeile aufbauen.....	D-3
D-4: MKCH: Zeile eines Zeichens aufbauen.....	D-3
D-5: IRQA: Routine für maskierbaren Interrupt.....	D-4
D-6: NMIA: Routine für nicht maskierbaren Interrupt.....	D-4
D-7: ATINIT: Anzeige und Tastatur initialisieren.....	D-5
D-8: ANZNEU: Anzeige mit aktuellen Daten füllen.....	D-5
D-9: ANZNU: Binärzahl zweistellig darstellen.....	D-5
D-10: TASLOW: Low-Level-Tastaturfunktion ausführen (1).....	D-6
D-11: TASLOW: Low-Level-Tastaturfunktion ausführen (2).....	D-7
D-12: CONIN: Tastatur über Tabelle abfragen.....	D-8
D-13: TASIA: Physikalischen Tastencode ermitteln.....	D-9
D-14: BIN2BCD: Umwandlung Binär -> BCD.....	D-10
D-15: INCCAS: COL-Adresse erhöhen.....	D-10
D-16: DECCAS: COL-Adresse vermindern.....	D-10
D-17: INCRAS: ROW-Adresse erhöhen.....	D-11
D-18: DECRAS: ROW-Adresse vermindern.....	D-11
D-19: INCROW: Zeilennummer erhöhen.....	D-11
D-20: DECROW: Zeilennummer vermindern.....	D-11
D-21: ICCAS: Versch. der Bilder um ein Pixel nach links.....	D-12
D-22: DCCAS: Versch. der Bilder um ein Pixel nach rechts.....	D-12
D-23: ICRAS: Versch. der Bilder um ein Bild nach links.....	D-12
D-24: DCRAS: Versch. der Bilder um ein Bild nach rechts.....	D-12
D-25: ICRW1: Versch. der Bilder um eine Zeile nach oben.....	D-13
D-26: DCRW1: Versch. der Bilder um eine Zeile nach unten.....	D-13
D-27: CCCAS: COL-Adressen-Bereichsabfrage.....	D-13
D-28: CCRAS: ROW-Adressen-Bereichsabfrage.....	D-13
D-29: CCROW: Zeilennummer-Bereichsabfrage.....	D-14
D-30: XILUP: XILINX-Konfiguration laden.....	D-14
D-31: WAIT: Lange Warteschleife.....	D-15
D-32: WTLAUF: Kurze Warteschleife für Hauptprogramm.....	D-15

1. Abkürzungsverzeichnis

LCA	Logic Cell Array
CLB	Configurable Logic Block
IOB	Input/Output-Block
PC	Personal Computer
SYNC	Video-Synchronpuls-Erzeugung
CLK2	Ablaufsteuerung ("Clock")
ADRE	Adresserzeugung
PROZ	Prozessorsteuerung
CPU	Central Processing Unit
D-RAM	Dynamischer Schreib/Lese-Speicher (Dynamic Random Access Memory)
ROW	(Speicher-) Reihe
COL	(Speicher-) Spalte
A/D	Analog -> Digital
D/A	Digital -> Analog
ADC	Analog/Digital-Converter
DAC	Digital/Analog-Converter

2. Einleitung

2.1. Zielsetzung

Gegenstand der vorliegenden Arbeit ist die Entwicklung eines digitalen Bildspeichers mit folgenden Eigenschaften:

- Abspeichern von Vollbildern mit möglichst geringem Speicherplatzbedarf
- Wahlweise Einsatz einer Datenreduktion
- Darstellung der gespeicherten Bilder in horizontaler und vertikaler Richtung auf 1/4 verkleinert
- Möglichkeit des Zugriffs eines Mikroprozessorsystems auf den Speicher zum Ein- und Auslagern von Bildern von einem Massenspeicher

Das System soll in einem professionellen Videoschnittplatz zum Einsatz kommen, um die Nachbearbeitung von Schnittlisten zu verbessern. Diese Schnittlisten enthalten die Anfangs- und Endpositionen einzelner Einstellungen als Timecodewerte sowie deren Reihenfolge. Der Bildspeicher soll diesen Zahlenkolonnen repräsentative Standbilder (z.B. das erste, das letzte und ein typisches Bild einer Einstellung) zuordnen, um sie besser lesbar zu machen.

Falls durch eine Integration der digitalen Logik des Bildspeichers in ein Gate-Array eine ausreichende Kostenreduzierung möglich ist, soll der Bildspeicher mit reduzierter Speicherkapazität und/oder Bildauflösung auch im Consumer-Bereich eingesetzt werden. Die Steuerlogik muß daher flexibel genug sein, um beide Einsatzgebiete abdecken zu können.

2.2. Arbeitsumgebung

Die Diplomarbeit wurde bei der Firma Alpermann & Velte GmbH in Remscheid durchgeführt. Die Firma beschäftigt sich seit ca. 7 Jahren intensiv mit Verfahren und Hilfsmitteln zur Nachbearbeitung von Videofilmproduktionen. Dazu gehören in erster Linie Timecodegeneratoren und -leser für professionelle und semiprofessionelle Videostudios, die eine bildgenaue Adressierung von Videobildern ermöglichen.

In jüngerer Zeit wurden auch Schnittsteuergeräte mit Timecode-Steuerung für Heim-Videorecorder (VHS bzw. Video-8) entwickelt, was durch die Verwendung selbstentwickelter Timecode-Chips als Gate-Arrays ermöglicht wurde. Auch ein Schriftgenerator ist im Angebot, der ebenfalls eigene Gate-Arrays enthält.

2.3. Konzeption

2.3.1. Auflösung

Die Berechnung der benötigten Auflösung der gespeicherten Bilder basiert auf den Daten der 625-Zeilen PAL-Fernsehnorm [1]:

Zeilenzahl je Vollbild	$Z = 625$
Vollbildfrequenz:	$f_v/2 = 25\text{Hz}$
Zeilendauer:	$H = 64\text{fs}$

Da die Bilder in H- und V-Richtung bei der Wiedergabe auf 1/4 der Originalgröße verkleinert dargestellt werden sollen, muß bei der Aufnahme nur mit 1/4 der normalen Auflösung abgetastet werden.

Bei der Aufnahme wird nur ein Halbbild abgetastet. Bei der Wiedergabe werden zwei identische Halbbilder erzeugt. Dadurch wird ein Zeilenflackern bei der Wiedergabe, verursacht durch das Zeilensprungverfahren, vermieden. Die vertikale Auflösung halbiert sich dadurch auf 312 Zeilen. Von diesen 312 Zeilen wird nun jede 4. Zeile abgetastet. Das ergibt 78 Zeilen. Da nicht alle Zeilen Bildinhalt tragen, wurden 72 Zeilen gewählt.

Die Abmessungen des Fernsehbildes verhalten sich Höhe:Breite = 3:4. Um eine horizontal wie vertikal in etwa die gleiche Auflösung zu erhalten, wurde eine horizontale Auflösung von 80 Punkten gewählt. Der sichtbare Teil einer Videozeile dauert 52fs. Pro Pixel stehen also 650ns zur Verfügung. Bei einem zugrundeliegenden Quarztakt von 32MHz entspricht dies 20,8 Taktzyklen. Gewählt wurden 20 Taktzyklen. Bei der Wiedergabe wird mit vierfacher Pixelfrequenz ausgegeben, was 5 Taktzyklen pro Pixel entspricht, also 156ns. Dies liegt unter der RAS-Zykluszeit von 120ns-D-RAMs, die 180ns beträgt. Es muß daher, um diese kostengünstigen Speicherchips verwenden zu können, im Page-Modus auf die D-RAMs zugegriffen werden.

2.3.2. Adressierungsschema

Der Bildspeicher wird in Speicherseiten zu jeweils 256 Pixeln aufgeteilt. In jede dieser Seiten werden 3 Zeilen zu jeweils 80 Pixeln gespeichert. Das ergibt $3 \cdot 80 = 240$ Pixel. Die restlichen 16 Pixel einer Seite werden derzeit nicht genutzt. 24 aufeinanderfolgende Speicherseiten ergeben ein komplettes Bild mit $3 \cdot 24 = 72$ Zeilen.

Die Seitenadressen werden im folgenden ROW-Adressen genannt, die Pixeladressen innerhalb einer Seite COL-Adressen.

Es ergeben sich für das erste Bild im Speicher folgende Adressen (ROW-Adresse/COL-Adresse):

0/0	0/1	0/2	..	0/77	0/78	0/79
0/80	0/81	0/82	..	0/157	0/158	0/159
0/160	0/161	0/162	..	0/237	0/238	0/239
1/0	1/1	1/2	..	1/77	1/78	1/79
1/80	1/81	1/82	..	1/157	1/158	1/159
1/160	1/161	1/162	..	1/237	1/238	1/239
:	:	:		:	:	:
22/0	22/1	22/2	..	22/77	22/78	22/79
22/80	22/81	22/82	..	22/157	22/158	22/159
22/160	22/161	22/162	..	22/237	22/238	22/239
23/0	23/1	23/2	..	23/77	23/78	23/79
23/80	23/81	23/82	..	23/157	23/158	23/159
23/160	23/161	23/162	..	23/237	23/238	23/239

Das folgende Bild beginnt mit der ROW-Adresse 24:

24/0	24/1	24/2	..	24/77	24/78	24/79
24/80	24/81	24/82	..	24/157	24/158	24/159
24/160	24/161	24/162	..	24/237	24/238	24/239
:	:	:		:	:	:
:	:	:		:	:	:

3. Technologien

3.1. Logic Cell Array (LCA)

Für die Implementierung der digitalen Schaltung wurden Logic Cell Arrays (LCAs) eingesetzt. Es handelt sich dabei um anwender-programmierbare Logikbausteine hoher Komplexität. Sie ähneln in ihrer internen Struktur den Gate-Arrays. Die folgende Beschreibung ist angelehnt an [3]. Dort sind auch detailliertere Informationen zu den LCAs zu finden.

Hauptbestandteil der LCAs sind CLBs (**C**onfigurable **L**ogic **B**lock), die in einer Matrix (z.B. 8x8) angeordnet sind. Dies sind programmierbare Einheiten, die Flip-Flops und kombinatorische Logik enthalten. Zwischen den CLBs befinden sich Verdrahtungskanäle, durch die sie miteinander verbunden werden können. Die CLB-Matrix ist von IOBs (**I/O-Block**) umgeben, die auf die externen Anschlüsse (PADs) des LCAs führen.

Es gibt zwei LCA-Familien: die 2000er- und die 3000er-Serie. Sie unterscheiden sich im Aufbau der CLBs und der IOBs. So enthält beispielsweise jeder IOB bzw. CLB der 2000er LCAs je ein Flip-Flop. Bei den 3000ern sind es jeweils zwei. In der vorliegenden Arbeit wurden nur 3000er LCAs eingesetzt; die folgenden Beschreibungen beziehen sich daher auf diese LCA-Familie. In jeder Familie gibt es außerdem eine Reihe verschiedener Typen, die sich durch die Anzahl der CLBs und IOBs unterscheiden.

Jeder LCA-Typ ist in verschiedenen Geschwindigkeitsklassen erhältlich. Diese werden als obere Grenzfrequenz eines Toggle-Flip-Flops in MHz angegeben. LCAs mit Toggle-Frequenzen von 33, 50 und 70 MHz sind erhältlich; 100 MHz-Typen sind in Vorbereitung.

3.1.1. Configurable Logic Block (CLB)

Jeder CLB enthält zwei flankengetriggerte D-Flip-Flops mit Taktfreigabeeingang (Clock Enable) sowie entweder eine kombinatorische Verknüpfung mit fünf Eingängen oder zwei mit je vier Eingängen. Die Ein- und Ausgänge der CLBs sind über Pins mit den Verdrahtungskanälen und damit mit anderen CLBs oder IOBs verbunden. Die Verknüpfung ist als Look-Up-Table ausgeführt; die Durchlaufverzögerung ist daher von der Art der Verknüpfung unabhängig. Sie beträgt bei 70 MHz-Typen 8ns.

3.1.2. Input/Output Block (IOB)

Die IOBs enthalten ein Ein- und ein Ausgabe-Flip-Flop. Der Ausgabebuffer besitzt einen Three-State-Eingang, so daß auch bidirektionale Pads und Open-Collector-Ausgänge möglich sind. Der Eingangsbuffer kann mit TTL- und CMOS-Charakteristik gewählt werden. Das Eingangs-Flip-Flop kann flanken- oder pulsgetriggert sein.

3.1.3. Verdrahtungskanäle

CLBs und IOBs werden über Leitungen in den Verdrahtungskanälen miteinander verbunden. Einige diese Leitungen sind in Aluminium, andere in Poly-Silizium ausgeführt. Da die Aluminiumleitungen sehr viel schneller sind, sind sie insbesondere für den synchronen Takt aller Flip-Flops reserviert. Die Datenleitungen werden im Allgemeinen über Poly-Siliziumleitungen geführt. Die Verzögerungszeiten dieser Leitungen hängen in erster Linie von deren Länge ab. Im Schnitt sind Verzögerungen von 3ns bis zu 40ns und mehr möglich.

3.1.4. Entwicklungswerkzeug PC

Zur Implementierung der LCAs wurde das LCA-Entwicklungspaket der Fa. XILINX eingesetzt. Es läuft auf kompatiblen ATs bzw. 386ern mit mindestens 4MB Hauptspeicher im Protected Mode. Es umfaßt folgende Programme:

- Ein Netzlistenkonvertierungsprogramm, das die vom Schaltungsplaneditor gelieferte Netzliste minimiert und in CLBs aufteilt
- Einen Autorouter, der das Plazieren der CLBs und deren Verdrahtung übernimmt, sowie
- Einen interaktiven LCA-Editor, der Manipulationen des LCAs auf CLB-Ebene erlaubt.

Die Netzliste wurde auf einer Workstation erstellt, was im folgenden Abschnitt beschrieben wird.

3.2. Entwicklungswerkzeug Workstation

Zum Entwurf der digitalen Schaltungen stand eine Workstation der Firma Daisy zur Verfügung. Diese bietet leistungsfähige Werkzeuge zur Entwicklung digitaler Schaltungen. Dazu gehören ein Schaltplaneditor, der eine hierarchische Schaltungsentwicklung unterstützt, sowie ein Simulator. Die Schaltpläne und Timingdiagramme der LCAs sind sämtlich auf dieser Station entstanden. Die Schaltpläne enthalten einige Besonderheiten, die kurz erläutert werden sollen.

3.2.1. Symbole und Makros

Die Schaltungen basieren auf Grundbausteinen (sog. Primitives). Dies sind im Wesentlichen kombinatorische Gatter, Basis-D-Flip-Flops und Ein/Ausgabepuffer. Ein einfaches Beispiel ist das Symbol 'NOR2', ein NOR-Gatter mit zwei Eingängen.

Aus diesen Grundbausteinen sind komplexere Bausteine, sog. Makros, zusammengesetzt. Sie bestehen aus zwei Teilen: dem Symbol, das im Schaltplan erscheint, und einer Schaltung, aus der die Funktion hervorgeht. Die Symbole der Makros sind mit dem Buchstaben "N" ("Nested") gekennzeichnet. Zum LCA-Entwicklungspaket werden zahlreiche Makros mitgeliefert. Darunter befinden sich Dekoder, Multiplexer, Addierer, verschiedene Flip-Flops, Latches und Zähler, die in Bibliotheken zusammengefaßt sind. Ein Beispiel für einen Zähler ist das Makro 'C16BCPRD'. Es ist ein setzbarer (P) Binärzähler (B) modulo 16 mit Count-Enable (C) und asynchronem Reset (RD). Zusätzlich wurden eigene Makros erstellt, insbesondere Binärzähler und Flip-Flops, die den Clock-Enable-Eingang der LCA-Basis-Flip-Flops benutzen (Siehe Anhang C).

3.2.2. Hierarchischer Schaltplan

Um eine umfangreiche Schaltung übersichtlicher zu machen, kann sie auf viele einzelne Schaltpläne verteilt werden. Dies ist sowohl horizontal als auch vertikal möglich. Eine horizontale Aufteilung liegt vor, wenn eine Zeichnung aus Platzgründen auf mehrere Seiten verteilt ist, die durchnummeriert sind (z.B. Seite 1, 2 und 3). Da diese Seiten zu einer Zeichnung gehören, dürfen in ihnen vergebene Name für Leitungen und Bauteile jeweils nur ein-

mal vorkommen. Gemeinsame Leitungen auf verschiedenen Seiten werden über spezielle Steckersymbole mit einfachem Rand (Inter-Page-Connectors) miteinander verbunden.

Bei einer vertikalen Aufteilung wird ein sog. Block eingezeichnet, der eine Zeichnung auf tieferer Zeichnungsebene repräsentiert. Diese trägt den Namen der Ursprungszeichnung erweitert durch den Namen des Blocks. Falls sich z.B. in der Zeichnung /USER/ZEICHNUNG/1.DRW ein Block mit dem Namen "BLK10" befindet, so trägt die Zeichnung des Blocks den Namen /USER/ZEICHNUNG/BLK10/1.DRW. Leitungen aus dem Blockinneren werden über Pins am Block und über hierarchische Steckersymbole mit doppeltem Rand (Hierarchic Connectors) im Blockinneren referenziert. Hierarchische Stecker auf der obersten Zeichnungsebene entsprechen den physikalischen Anschlüssen einer Schaltung nach außen.

3.2.3. Leitungen und Busse

Einigen Leitungen sind in den Schaltplänen Parameter zugeordnet. Diese kontrollieren die Implementierung der Schaltung in das LCA.

Der "EXT"-Parameter ist Leitungen zugeordnet, die im LCA an Ein- oder Ausgabepads liegen sollen. Der Parameter kann folgende Werte annehmen:

- EXT=IPAD: Eingangsleitung
- EXT=OPAD: Ausgangsleitung
- EXT=BPAD: Bidirektionale Leitung
- EXT=UPAD: "Unbonded Pad"

Ein "Unbonded Pad" besitzt keine Verbindung zu einem externen IC-Pin. Es kann als internes Flip-Flop genutzt werden.

Der "CONST"-Parameter beeinflusst die Platzierung der CLBs und das Routing im Autorouter. Verwendet wurden folgende Werte:

- CONST=C: "Kritische Leitung". Diese Leitung soll während des Routings des LCAs bevorzugt behandelt werden, um möglichst kurze Verzögerungszeiten zu erhalten.
- CONST=L: "Longline". Für diese Leitung sollen die in Aluminium ausgeführten Leitungen verwendet werden. Dies ist besonders für den synchronen Takt in Verbindung mit dem Treiber GCLK sinnvoll.

Andere Werte kontrollieren die Partitionierung der Netzliste in CLBs und die Verteilung der CLB-Pins.

Mehrere Leitungen können zu Bussen zusammengefaßt werden. Die Namen der Leitungen, die in einem Bus enthalten sind, müssen als "CONTS"-Parameter dem Bus zugeordnet werden. Bei durchnummerierten Leitung ist eine Abkürzung möglich. Enthält z.B. ein Bus die Leitungen A3, A2, A1 und A0, wird ihm der Parameter CONTS=A(3:0) zugeordnet.

3.3. In-Circuit-Emulator

Das zum Bildspeicher gehörende Programm wurde auf einem Mikroprozessor-Entwicklungssystem der Fa. Dr. Krohn & Stiller erstellt. Zu diesem Entwicklungssystem gehört ein In-Circuit-Emulator, was die Programmentwicklung wesentlich erleichterte.

3.4. Wire-Wrap-Technik

Die LCAs wurden untereinander und mit den anderen Schaltungsteilen in Wire-Wrap-Technik verbunden. Diese Verdrahtungstechnik stellt einen gute Kompromiß zwischen Änderbarkeit und Haltbarkeit der Verbindungen dar. Nachteilig ist der recht große Platzbedarf der Wrap-Stifte, was bei der Anfertigung des Prototyps jedoch keine Rolle spielte.

4. Schaltungsbeschreibung

4.1. Übersicht

Der Bildspeicher besteht aus folgenden Komponenten:

- A/D-Wandler für die Aufnahme (ADC)
- Ablauf- und D-RAM-Steuerung (CLK2)
- Erzeugung der D-RAM-Adressen (ADRE)
- Erzeugung der Video-Synchronimpulse (SYNC)
- Datenfluß- und CPU-Steuerung (PROZ)
- Dynamischer Speicher (DRAM)
- Mikroprozessor (CPU)
- D/A-Wandler für die Wiedergabe (DAC)

Die Module CLK2, ADRE, SYNC und PROZ sind als LCAs (**L**ogic **C**ell **A**rray) ausgeführt. Dies sind anwenderprogrammierbare Logikbausteine, die sich, wie im vorliegenden Fall, gut zur Realisation komplexer Digitalschaltungen eignen. Auf die Besonderheiten der LCAs wurde im vorangegangenen Kapitel eingegangen. Näheres ist in [3] zu finden.

4.2. Betriebsmodis

Die Schaltung kann einen von drei möglichen Betriebsmodis einnehmen. Diese Modis werden von den Signalen CPIXI und RECAKI wie folgt gekennzeichnet:

CPIXI	RECAKI	Modus
0	0	Wiedergabemodus (PIX, 0)
1	0	CPU-Zugriffs-Modus (CPU, 1)
0	1	Aufnahmemodus (REC, 2)
1	1	(verboten)

Im Wiedergabemodus werden die zuvor abgespeicherten Bilder verkleinert auf dem Bildschirm angezeigt. Der CPU-Zugriffsmodus erlaubt das Auslesen bzw. Beschreiben des Bildspeichers vom Mikroprozessor aus. Im Aufnahmemodus wird schließlich ein Videosignal abgetastet und das enthaltene Bild abgespeichert.

Die Wiedergabe der gespeicherten Bilder erfolgt in einer Reihe nebeneinander. Diese Bildreihe ist horizontal in Schritten von einem Pixel verschiebbar. Dadurch können bis zu fünf verschiedene Bilder gleichzeitig sichtbar werden, wobei die beiden Bilder am linken und rechten Rand teilweise durch die Bildschirmmaske verdeckt sind. Die vertikale Position der Bildreihe ist ebenfalls wählbar.

4.3. A/D-Wandler (ADC)

Als A/D-Wandler wurde der Baustein UVC3130 eingesetzt. Er enthält einen 8-Bit-Flash-A/D- sowie einen 10-Bit-D/A-Wandler. Ihm ist ein Filter zur Begrenzung der Bandbreite des hereinkommenden Videosignals auf ca. 800kHz vorgeschaltet. Dies entspricht der halben Abtastrate von $(32/20)\text{MHz} = 1,6\text{MHz}$.

Die A/D-D/A-Wandlerplatine sowie das Eingangsfiler sind nicht Bestandteil der Diplomarbeit. Sie wurden begleitend von der Firma Alpermann&Velte entwickelt.

4.4. Ablauf- und D-RAM-Steuerung (CLK2)

Im LCA CLK2 werden das D-RAM-Timing und weitere Hilfstakte für den Bildspeicher gebildet. CLK2 wird vom LCA SYNC mit dem Video-Timing versorgt und veranlaßt seinerseits das LCA ADRE zur Erzeugung der D-RAM-Adressen.

4.4.1. Top-Level (CLK2/1)

4.4.1.1. Eingangssignale

Über den Anschluß CLKI wird das LCA mit einem Quarztakt von 32MHz versorgt. Über ihn werden alle Flip-Flops mit einem synchronen Takt versorgt.

Der Eingang RESI dient zum Initialisieren der Schaltung während der Logiksimulation. Im realen LCA wird er nicht benötigt. Vor der Implementation wird er über ein AND2B1-Gatter maskiert und fällt während der Übersetzung in das LCA heraus. Am LCA ist dieser Anschluß daher nicht mehr vorhanden.

Die Eingänge VSYNCI (V-Puls), VSPULSI (Vertikaler Startpuls), HSYNCI (H-Puls), und DISPLAYI (Horizontale Zeilenmaske) stammen aus dem LCA SYNC. Sie stellen das Video-Timing dar. Ihre Funktion wird im einzelnen am LCA SYNC besprochen.

Über den Anschluß WPCASI kann vom Prozessor-Datenbus, angeschlossen an DB(7:0)I, ein Register beschrieben werden. Über den in dieses Register einbeschriebenen Wert kann die horizontale Position des ersten, und damit auch aller weiteren Bilder einer Zeile bestimmt werden. Ist in diesem Register beispielsweise der Wert 30 enthalten, wird am Beginn einer Zeile als erstes das Pixel Nummer 30 dargestellt. Dadurch entsteht auf dem Bildschirm der Eindruck eines nach links verschobenen Bildes.

Die Eingänge CPIXI und RECAKI bestimmen den Betriebsmodus der Schaltung. Sie werden vom LCA PROZ geliefert.

CSCPUi kennzeichnet einen Bildspeicherzugriff im CPU-Modus vom Mikroprozessor aus. RAMRWNI entscheidet, ob es sich um einen Les- (RAMRWNI=1) oder Schreibzugriff (RAMRWNI=0) handelt.

4.4.1.2. Ausgangssignale

Die Ausgänge RASO, CASO, OEO und WEO sind mit den entsprechenden Eingängen des D-RAMs verbunden.

Über den Anschluß RSWAO wird das LCA ADRE zum Anlegen der Reihen-Adresse (Row-Address) veranlaßt. Der Anschluß CSWAO bewirkt das gleiche für die Spalten-Adresse (Column-Address).

VRESO, VSTARTO, HRESO und HSTARTO sind Impulse, die über digitale Differenzierglieder aus den Video-Timing-Signalen gewonnen werden. Sie beeinflussen die Erzeugung der D-RAM-Adressen im ADRE-Modul.

DXCLKO ist der Pixeltakt für den D/A-Wandler, AXCLKO entsprechend für den A/D-Wandler. Mit DACLKO und ADCLKO werden Zwischenspeicher in den Datenwegen im PCA PROZ kontrolliert. CLK40 ist der durch vier geteilte Eingangstakt mit 8MHz. Er versorgt das SYNC-Modul.

4.4.2. Prinzip der RAS/CAS-Erzeugung

Das RAS/CAS-Timing des dynamischen Speichers wurde in drei Grundzyklen zerlegt:

- 1) Anlegen der ROW-Adresse und fallende Flanke der RAS-Leitung
- 2) Anlegen der COL-Adresse und fallende Flanke der CAS-Leitung
- 3) Refresh-Zyklus

Der Refresh-Zyklus führt ein CAS-vor-RAS-Refresh durch, bei dem ein im D-RAM befindlicher Refresh-Zähler die Refresh-Adresse erzeugt. Von außen muß daher bei diesem Refresh-Modus keine Adresse angelegt werden.

Jeder der drei Zyklen ist als Schieberegister realisiert. Zum Starten eines Zyklus wird ein Impuls von einem Taktzyklus Breite in das Schieberegister eingespeist. Beim Durchtakten löst dieser Impuls zeitlich getrennte Aktionen aus, z.B. das Setzen bzw. Rücksetzen eines Signals oder das Anlegen einer Adresse an das D-RAM. Wenn der Impuls am Ausgang eines Schieberegisters erscheint, kann er in ein weiteres Schieberegister eingespeist werden. Dadurch können als den Grundzyklen komplexe Timings zusammengesetzt werden.

Neben den drei Grundzyklen existiert noch ein Wartezyklus, der selbst keine Aktionen auslöst, sondern nur das Aufnahmetiming anpaßt.

Das Aufeinanderfolgen der einzelnen Zyklen ist in Zustandsfolgediagrammen aufgezeichnet. (Abbildung 4-1) Die einzelnen Grundzyklen sind darin als Knoten dargestellt. Die Knoten CAS, RAS, RFSH und WAIT repräsentieren die erwähnten Zyklen. Ein Zustand ist aktiv, wenn das ihn darstellende Schieberegister einen Impuls enthält. IDLE ist ein Ruhezustand, von dem aus die anderen Zyklen gestartet werden. Er liegt vor, wenn keiner der vier anderen Zyklen aktiv ist. Die Bedingungen, die an die Aufeinanderfolge der Grundzyklen geknüpft werden, sind den Kanten zugeordnet.

Für jeden der drei Betriebsmodis existiert ein eigenes Zustandsfolgediagramm. Folgende Signale werden darin benutzt:

DISPLAY: Aktiv (=1) während des sichtbaren Teils einer Zeile

PAUSE: Aktiv (=1) vom Ende des H-Pulses bis zum Anfang des sichtbaren Teils der Zeile

CAS79: Aktiv (=1) vor dem jeweils 80. Pixel einer Bildzeile

HSTART: Impuls am Anfang des sichtbaren Teils einer Zeile
(steigende Flanke von DISPLAY)

CSCPU: Aktiv (=0) während eines Mikroprozessorzugriffs

CPUEND: Impuls am Ende eines Mikroprozessorzugriffs (steigende Flanke von CSCPU)

IRFSH: Impuls jeweils ca. 13fs nach dem letztem RAS-Zyklus, ausgelöst vom Refresh-Timer.

DIDIS: Aktiv (=1) während des sichtbaren Teils einer abzutastenden Zeile (während der Aufnahme)

Im Modus 0, dem Anzeigemodus muß besonders schnell auf den Bildspeicher zugegriffen werden, da hier die Pixelfrequenz am größten ist. Es wird daher im Page-Modus gearbeitet. Dabei kann, bei unveränderter ROW-Adresse, auf eine Zeile im D-Ram sehr schnell zugegriffen werden.

Wie aus dem Zustandsfolgediagramm für Modus 0 entnommen werden kann, wird am Anfang einer Zeile, ausgelöst durch HSTART zunächst ein RAS-Zyklus durchgeführt. Hierbei wird die Row-Adresse des ak-

tuellen Bildes und der aktuellen Zeile in das D-Ram geschrieben. Anschließend werden in CAS-Zyklen die Pixel einer Zeile, maximal 80, ausgelesen. Ist, angezeigt durch CAS79, das erste Bild vollständig angezeigt, wird in RFSH ein Refresh-Zyklus durchgeführt und mit dem nächsten Bild der Zeile fortgefahren. Diese Schleife: RAS - CAS (wiederholt) - RFSH wird solange wiederholt, bis die Zeile zuende ist. Bis zum Ende des H-Pulses der nächsten Zeile werden Refresh-Zyklen erzeugt und anschließend in IDLE auf den Anfang des sichtbaren Teils der folgenden Zeile gewartet.

Modus 1, der CPU-Modus, gestattet den Zugriff auf den Bildspeicher vom Mikroprozessor aus. Der Zugriff erfolgt nicht wahlfrei, vielmehr werden die einzelnen Pixel eines Bildes Byte-sequentiell gelesen und geschrieben. Die erforderlichen Adressen werden dabei im Modul ADRE automatisch hochgezählt.

Die Prozessorzugriffe auf den Bildspeicher erfolgen außerdem nicht direkt, sondern über Zwischenspeicher, die im Modul PROZ enthalten sind. Erst nach dem Ende des Zugriffs werden die Daten vom bzw. in den Bildspeicher übertragen. Bei Lesezugriffen muß daher zunächst ein Dummy-Zugriff durchgeführt werden, damit die Daten des ersten Pixels in den Zwischenspeicher übertragen werden können. Alle folgenden Zugriffen können normal ablaufen.

Das Zustandsfolgediagramm für Modus 1 beschreibt den Ablauf der Speicherzyklen. Ausgehend vom IDLE-Zustand wird nach erfolgtem CPU-Zugriff in einem RAS- und nachfolgendem CAS-Zyklus der eigentliche Speicherzyklus durchgeführt. Diesem folgt ein Refresh-Zyklus. Da zu dieser Zeit der Mikroprozessor mit der Verarbeitung der gelesenen bzw. dem Heranschaffen der folgenden Daten beschäftigt ist, stört dieser Refresh die CPU-Zugriffe nicht.

Nach dem Refresh wird wieder der IDLE-Zustand eingenommen. Falls dieser länger als ca. 13fs dauert, wird über IRFSH ein Refresh-Zyklus erzwungen, es sei denn, es wird gerade ein CPU-Zugriff durchgeführt. In diesem Fall ist kein Zwangs-Refresh nötig, da am Ende des CPU-Zugriffs ohnehin ein Refresh erfolgt.

Im Modus 2 wird ein Videobild aufgezeichnet. Die Abtastfrequenz beträgt dabei sowohl vertikal als auch horizontal nur 1/4 der Wiedergabefrequenz. Die horizontale Verlangsamung wird durch Einschleichen eines Refresh- und eines Wartezyklusses bei jedem abgetasteten Pixel erreicht.

Bei der Wiedergabe wurde jeden 5. Taktimpuls ein Pixel ausgegeben. Dies entspricht der Länge des CAS-Zyklus. Bei der Aufnahme

wird nun alle 20 Taktzyklen ein Pixel abgespeichert. Dies entspricht der Summe der Länge von RAS- (3), CAS- (5), RFSH- (9) und WAIT-Zyklus (3). Aufnahmefrequenz zu Wiedergabefrequenz verhalten sich also wie 1 zu 4.

Das Zustandsfolgediagramm zu Modus 2 verdeutlicht dies. Im IDLE-Zustand wird auf den Anfang einer Zeile gewartet. Durch HSTART wird die Schleife RFSH - RAS - CAS - WAIT ausgelöst. Diese wird bis zum Ende der Zeile wiederholt. Auf den Anfang der nächsten Zeile wird in RFSH-Zyklen gewartet.

4.4.3. RAS/CAS-Erzeugung (CLK2/LOGIC/1)

Die zuvor erläuterten Zustandsfolgediagramme sind in Abb. A-3 realisiert. Die 3-auf-1 - Multiplexer N3, N4, N13, N14 und N15 schalten zwischen den verschiedenen Modi um. Aus in den Blöcken RASZYK, CASZYK und RFSHZYK erzeugten Impulsen werden in getakteten RS-Flip-Flops die Signale RASY und CASY gewonnen. Sie entsprechen dem RAS- und CAS-Signal der D-Rams. RSWA und CSWA lösen im Modul ADRE das Anlegen der dazugehörigen Adressen aus.

Das im Block CASZYK enthaltene Schieberegister bestimmt mit seiner Länge im Wiedergabemodus den Pixeltakt. Derzeit beträgt dieser, wie bereits beschrieben, 1/5 des Systemtakts von 32MHz. Das D-RAM-Timing erlaubt eine CAS-Zykluszeit von minimal 85ns. Der CAS-Zyklus könnte also auf 4 oder sogar auf 3 Taktzyklen (=94ns) verkürzt werden. Falls die Vertikalsteuerung entsprechend angepaßt wird, wäre dadurch eine weitere Verkleinerung der dargestellten Bilder auf 1/5 oder ca. 1/7 der ursprünglichen Größe möglich. Durch die entsprechen verringerte Zeilenzahl könnten in dem vorhandenen Speicher entsprechend mehr Bilder untergebracht werden, was z.B. für die Anwendung des Bildspeichers im Consumer-Bereich interessant sein könnte.

4.4.4. Differenzierer und CAS-Zähler (CLK2/LOGIC/2)

Da die Adressen von ADRE um vier Taktzyklen verzögert erzeugt werden, geschieht dies auch mit dem Signal RASY. CASY ist bereits um einen Taktzyklus verschoben und erfährt daher eine Verzögerung um weitere drei. Ebenso das Signal OEWEY.

Aus den Signale CSCPU, VSYNC, VSPULS, HSYNC und DISPLAY werden Flankenimpulse von einem Taktimpuls Länge gewonnen, d.h. sie wer-

den digital differenziert. (Siehe auch Abb. B-2.) Aus HRES und HSTART werden PAUSE und DIDIS abgeleitet.

N55 und N56 bilden zusammen den CAS-Zähler, der die Pixel der angezeigten Bilder abzählt und beim 80. Pixel das nächste Bild einleitet. Der Zähler kann am Anfang einer Zeile vorgesetzt werden, was das horizontale Verschieben der Bilder ermöglicht.

Der Zähler N28 teilt den Systemtakt von 32MHz durch vier auf 8MHz herunter. Dieser wird dem LCA SYNC zugeführt. Außerdem arbeitet er als Vorteiler für den Refresh-Timer, der an das Signal CLK16CE angeschlossen ist.

4.4.5. Aufnahmesteuerung und Refreshtimer (CLK2/LOGIC/3)

Der eingesetzte A/D-Wandler benötigt zum Abtasten eines Pixels zwei Taktimpulse. Bei einer Erweiterung des Bildspeichers auf Farbe müssen pro Bildpunkt zwei Abtastungen durchgeführt werden, was insgesamt vier Taktimpulsen entspricht. Diese vier Impulse an AXCLK werden durch die Schieberegisterkette N58 bis N77 aus dem Signal RCK erzeugt.

N90 und N91 bilden den Refresh-Timer. Sie lösen im CPU-Modus über IRFSH ca. 13s nach dem letzten Speicherzugriff einen Zwangs-Refresh aus. Bei jedem RAS-Zyklus wird der Timer über RSWA zurückgesetzt.

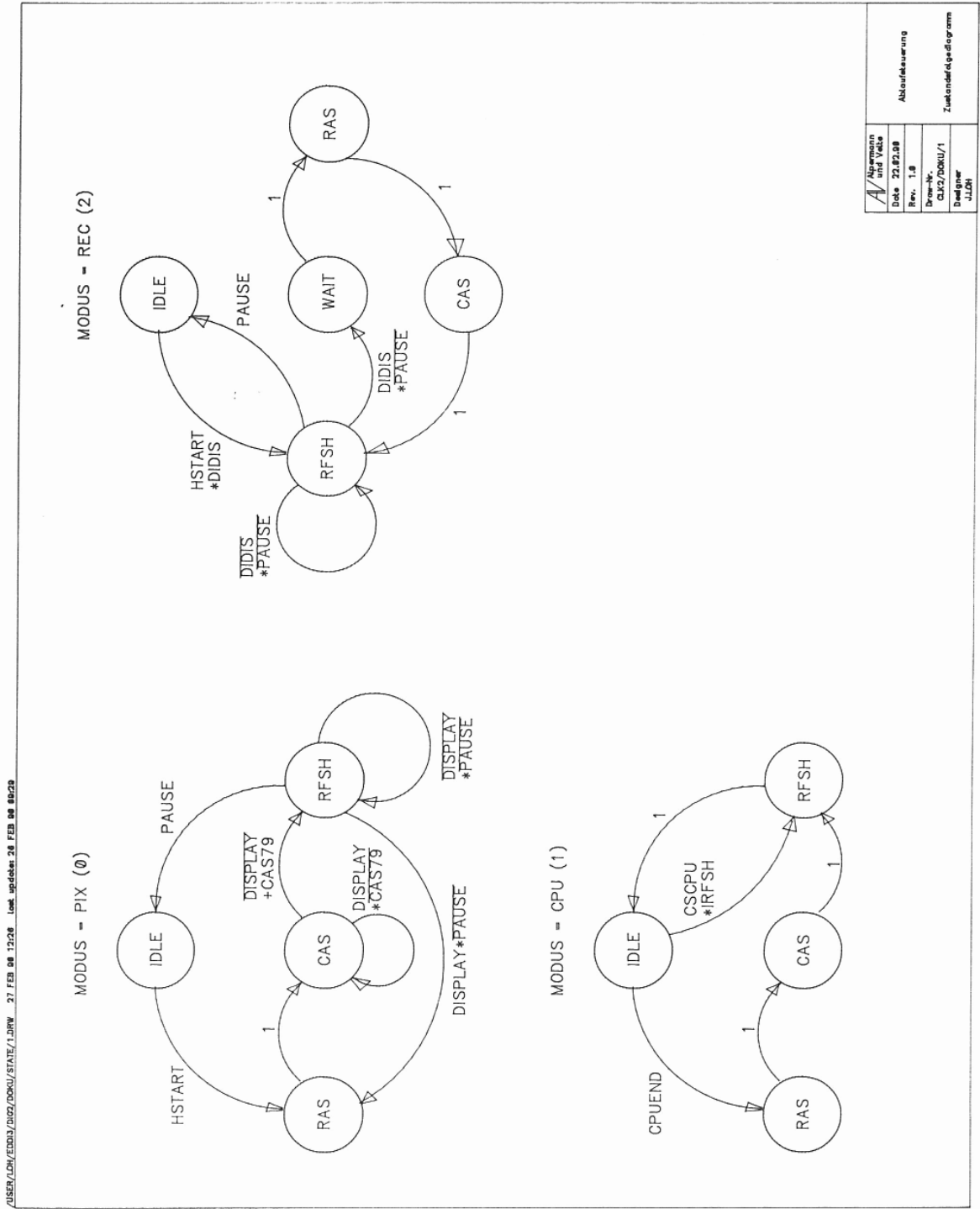
4.4.6. CLK2 - Zeitdiagramme

Das Zeitverhalten der Signale im Wiedergabemodus ist in Abb. B-1 dargestellt. Sie zeigt die RAS/CAS-Erzeugung während der H-Lücke. Vor dem H-Puls werden RFSH-Zyklen durchgeführt. Danach folgen das Warten auf HSTART und davon ausgelöst der D-Ram-Zugriff im Page-Modus.

Die Abbildungen B-3 und B-4 zeigen das RAS/CAS-Timing im CPU-Modus. In Abb. B-3 sind ein Schreib- und ein Lesezyklus dargestellt. Anschließend folgt ein Zwangs-Refresh-Zyklus ausgelöst durch IRFSH. B-4 zeigt die CPU-Zugriffe im Detail. Dabei sind die CAS-vor-RAS-Refresh-Zyklen im Anschluß an die Speicherzyklen gut zu erkennen.

In Abb. B-5 ist schließlich der Aufnahmemodus gezeigt. Hier beginnt jeder Speicherzyklus mit einem Refresh-Zyklus. Das Signal AXCLK mit vier Taktimpulsen pro Pixel ist ebenfalls dargestellt.

Abb. 4-1: CLK2- Zustandsfolgediagramm



4.5. Erzeugung der D-RAM-Adressen (ADRE)

Die Adressen des dynamischen Speichers werden im LCA ADRE erzeugt. Wie CLK2 wird es mit 32MHz synchron getaktet. Es enthält Zähler für die vertikalen bzw. horizontalen D-RAM-Adressen. Dabei entsprechen im wesentlichen die vertikalen den ROW- und die horizontalen den COL-Adressen. Ein Adreßmultiplexer ist ebenfalls enthalten. Die Steuersignale zum Weiterzählen bzw. Multiplexen erhält ADRE vom LCA CLK2.

4.5.1. Top-Level (ADRE/1)

4.5.1.1. Eingangssignale

CLKI liefert den Takt von 32MHz. RESI ist ein Rücksetzsignal für die Simulation. Es ist am realen LCA nicht vorhanden. CPIXI und RECAKI informieren ADRE über den jeweiligen Betriebsmodus.

Über WPCASI und den CPU-Datenbus DB(7:0)I erhält ADRE den gleichen horizontalen Positionswert in ein Eingangsregister einbeschrieben, der auch in CLK2 gespeichert wird. Die Pixelzähler in CLK2 und ADRE laufen also immer synchron. Mit WPRASI wird über den Datenbus bestimmt, welche Nummer das erste Bild am Anfang einer Zeile trägt. Dieser Wert wird ebenfalls in einem Register festgehalten.

Die folgenden Eingangssignale, sind jeweils nur eine Taktperiode aktiv. Sie kennzeichnen den Anfang eines Bildes (VRESI) und einer Zeile (HRESI) bzw. den vertikalen (VSTARTI) und horizontalen (HSTARTI) Start der Einblendung. RSWAI schaltet den Multiplexer auf die ROW-Adressen, CSWAI auf die COL-Adressen.

4.5.1.2. Ausgangssignale

Die Ausgangsleitungen A8o bis A00 führen die gemultiplexten Adressen für die dynamischen RAMs.

4.5.2. ROW-Zähler (ADRE/LOGIC/1)

Die Zähler N10 und N11 bilden zusammen den ROW-Zähler. Er liefert die 10 höherwertigen Adressen für den dynamischen Speicher. Im Aufnahme- und Wiedergabemodus wird er als Register benutzt. Am

Anfang jedes Bildes wird er über den YABUS mit einem neuen Wert geladen. Dieser Wert hängt ab von PRBUS, der vom Prozessor mit der Nummer des ersten anzuzeigenden Bildes geladen wird, von BBUS, der die Bilder einer Zeile zählt, sowie von ZDIV3, der jede dritte Zeile zählt.

Die zeilenrelative Bildnummer wird vom Bildzähler N3 erzeugt. Dieser zählt bei jedem RSWA-Impuls, also am Anfang jedes Bildes einer Zeile. Die relative Bildnummer wird mit der von der CPU gelieferten Nummer des ersten Bildes im Block ADD_1 addiert. Das Ergebnis ist in SBUS enthalten. Dieser enthält also die absolute Nummer des gerade angezeigten Bildes. SBUS wird im Block MUL20 mit 24 multipliziert, das entspricht einem Drittel der 72 Zeilen eines Bildes. Der so entstandene YBUS wird im Block ADD_2 mit ZDIV3 addiert. Dieser erhält aus N4/N5 die durch 3 geteilte Nummer der aktuellen Zeile.

Über die vor N10/N11 angeordnete Logik wird der ROW-Zähler im CPU-Modus vom Register in einen Zähler umgewandelt. Dadurch ist es möglich, in diesem Modus ein gesamtes Bild als fortlaufende Pixelfolge anzusprechen.

Der ROW-Zähler umfaßt im Prototyp nur 10 Bit. Das reicht zur Adressierung von 256KWorten, entsprechend 42 Bildern. Für eine größere Anzahl Bilder kann er leicht anstelle von P5/P6 aufgestockt werden. Hinzu kommt dann noch eine Dekodierlogik für verschiedene Speicherbänke.

4.5.3. COL-Zähler (ADRE/LOGIC/2)

Der Zähler N29 stellt den COL-Zähler dar. Anders als der ROW-Zähler wird er nur im Zählmodus betrieben. Gezählt werden die Pixel einer Zeile. In der ersten Zeile eines Bildes sind dies die Pixel 0-79, in der zweiten 80-159 und in der dritten 160-239. In der vierten Zeile ist der ROW-Zähler um eine Stufe höher, und der COL-Zähler zählt wieder 0-79.

Eine Ausnahme bildet das jeweils erste Bild einer Zeile. Da eine pixelweise horizontale Verschiebung der dargestellten Bilder möglich ist, muß mitten in einem Bild mit der Darstellung begonnen werden können. Die Nummer des ersten in einer Zeile dargestellten Pixels hängt von der Verschiebung des Bildes und von der Zeilennummer ab.

Die Berechnung wird im Addierer N16 durchgeführt. Addiert werden der vom der CPU gelieferte Versatz in CEBUS und die mit 80 multiplizierte Zeilennummer modulo 3 in ZMOD3. Mit dem so gewonnenen Wert in CAXBUS wird der COL-Zähler am Anfang jedes Bildes geladen. Da nur das jeweils erste Bild einer Zeile teilweise dargestellt werden darf, alle folgenden jedoch in voller Breite, wird für diese Bilder durch das Signal CEMA der Versatzwert PC(7:0) maskiert.

Die Schaltung vor N29 versetzt den COL-Zähler in den jeweiligen Zustand 'zählen', 'laden' und 'gesperrt', abhängig vom Betriebsmodus. Im CPU-Modus werden der COL- und der ROW-Zähler über CPUTC zu einem gemeinsamen ROW/COL-Zähler gekoppelt.

4.5.4. Adreßmultiplexer (ADRE/LOGIC/3)

Über die Multiplexer N30-N38 werden die erzeugten ROW- und COL-Adressen abhängig von RSWA und CSWA auf die D-RAM-Adreßpins geschaltet. Da der Refresh der D-RAMs mit CAS-vor-RAS-Zyklen durchgeführt wird, ist kein Refresh-Zähler notwendig, was den Schaltungsaufwand deutlich verringert.

4.5.5. ADRE - Zeitdiagramme

Die Abbildungen B-6 und B-7 zeigen die COL-Adreßerzeugung im Wiedergabemodus. Dargestellt wird die dritte Zeile eines Bildes. In Abb B-6 wurde der horizontale Versatz auf 60 eingestellt. Dieser Wert erscheint am Zeilenanfang während CEMA=1 auf dem CEBUS. Der Wert des Zeilenzählers ZMOD3 erscheint, multipliziert mit 80, auf CAABUS. Addiert mit CEBUS ergibt sich die Summe 220 auf CAXBUS. Mit diesem Wert wird der COL-Zähler geladen. Anschließend zählt er bei jedem CSWA-Impuls hoch.

Abb. B-7 zeigt die Situation am Ende des ersten Bildes. Hier ist CEMA=0, so daß der COL-Zähler diesmal auf 160 gesetzt wird. Das zweite Bild wird also voll angezeigt.

Die ROW-Adreßerzeugung wird in Abb. B-8 gezeigt. Dargestellt ist die H-Lücke vor der vierten darzustellenden Zeile. Über PRBUS wurde Bild Nr.20 angewählt. Die Bildnummer auf BBUS ist während der H-Lücke null, so daß die Summe auf SBUS ebenfalls gleich 20 ist. Multipliziert mit 24 ergibt das auf YBUS den Wert 480. Hinzu kommt der Zeilenzähler ZDIV3, so daß sich schließlich auf YABUS

der Wert 481 ergibt. Dieser wird in den ROW-Zähler geladen. Beim nächsten Bild wird BBUS=1. Dadurch erhöht sich YABUS um 24 auf 505, was das nächstfolgende Bild adressiert.

Abb. B-9 zeigt den Adreßmultiplexer in Aktion. Ausgelöst durch RSWA und CSWA schaltet das Signal RCSEL den Adreßmultiplexer um. Die so erzeugte Adresse wird mit ADRENA übernommen und erscheint an ABUS. Diese wird dem D-Ram zugeführt.

In Abb. B-10 wird der Zugriff auf den Bildspeicher im CPU-Modus gezeigt. Zunächst wird mit einem WPCAS-Impuls der horizontale Versatz gleich Null gesetzt. Anschließend wird mit einem WPRAS-Impuls die Nummer der Bildes, auf das zugegriffen werden soll, in PRBUS geschrieben. Dieser Wert, in diesem Fall 32, wird mit 24 multipliziert. Dies ergibt die erste ROW-Adresse des 32.Bildes. Sie wird mit einem weiteren WPCAS-Impuls in den ROW-Zähler geschrieben und erscheint als ROW-Adresse an RABUS. Da der Versatz gleich null gesetzt wurde, ist CABUS ebenfalls gleich null. Dies ist die Voraussetzung, um auf wirklich alle Pixel des Bildes zugreifen zu können. Bei jedem Zugriff auf den Bildspeicher, erkennbar an RSWA und CSWA, wird nun die D-Ram-Adresse um 1 erhöht, wodurch ein sequentieller Zugriff auf alle Pixel des Bildes möglich ist.

4.6. Erzeugung der Video-Synchronimpulse (SYNC)

Das LCA SYNC dient in erster Linie zur Erzeugung eines Composite-Sync-Signals, um abgespeicherte Bilder im Wiedergabemodus auf einem Monitor darstellen zu können. Dazu gehören die Steuerung des horizontalen und vertikalen Videotimings. Ein Startimpuls legt den vertikalen Beginn des dargestellten Bildes in einer wählbaren Zeile fest. Zusätzlich ist eine Zeilensteuerung integriert, die während der Aufnahme die aufzuzeichnenden Zeilen selektiert.

4.6.1. Top-Level (SYNC/1)

4.6.1.1. Eingangssignale

Der Anschluß CLKI versorgt das LCA mit einem synchronen Takt von 8Mhz, der im LCA CLK2 aus dem Systemtakt von 32MHz gebildet wird. Der Eingang RESI wird wie in den anderen LCAs nur während der Simulation benötigt und ist am realen LCA nicht vorhanden. Das gleiche gilt für den Eingang TESTI. Über diesen Anschluß kann der Zeilenzähler von Zeilentakt auf Systemtakt umgeschaltet werden, was während der Simulation die Anzahl der Simulationszyklen beim Testen des Zeilenzählers stark herabsetzt.

Über den Eingang WPROW1 kann vom Prozessor-Datenbus an DB(7:0)I ein Wert in das Eingangslatch einbeschrieben werden. Dieser bestimmt die Zeile, ab der die gespeicherten Bilder angezeigt werden sollen, also deren vertikale Position.

Für den Aufnahmemodus wird das LCA über die Eingänge VSEXTI und CSEXTI mit dem V-Sync bzw. Composite-Sync des aufzunehmenden Videosignals versorgt. Der Eingang RECRQI leitet die Aufnahme eines Einzelbildes ein.

4.6.1.2. Ausgangssignale

Die Ausgänge HSYNCO, VSYNCO und V1SYNCO liefern H-Puls, V-Puls und V1-Puls des in SYNC erzeugten Videosignals. DISPLAYO liefert ein Signal, das den für Bildinhalt nutzbaren Bereich einer Zeile kennzeichnet. Der Ausgang MASKO liefert dieses Signal nur für die Zeilen, die im Wiedergabemodus tatsächlich ein Bild enthalten.

VSPULSO markiert die Zeile, die über WPROW1 ausgewählt wurde. Sie leitet in CLK2 die Aufnahme bzw. die Wiedergabe eines Bildes ein.

Die bisher genannten Ausgangssignale dienen zur Synchronisation der D-RAM-Steuerung. CSYNCO liefert ein Composite-Sync-Signal zur Ansteuerung der Video-Mischstufe. RECAKO bestätigt die Einleitung eines Aufnahmezyklus durch RECRQI.

4.6.2. Horizontalsteuerung (SYNC/LOGIC/1)

Der Eingangstakt von 8MHz wird durch den 9Bit Horizontalzähler, aufgebaut aus N3 und N1, durch 512 auf 15625Hz (entsprechend 64fs), also auf Zeilenfrequenz heruntergeteilt. Diese wird als Signal VENA an den Zeilenzähler in der Vertikalsteuerung weitergeleitet. Eine Zeile im Videosignal beginnt mit der vorderen Flanke des H-Pulses. Dies entspricht einem Zählerstand des Horizontalzählers von Null. Aus den Zählerständen des Horizontalzählers werden alle horizontalen Signale abgeleitet. Dazu werden die Zählerstände, bei denen das jeweilige Signal eine steigende bzw. fallende Flanke zeigen soll, durch AND-Gatter auskodiert und auf getaktete RS-Flip-Flops geführt. Diese besitzen eine Vorrangschaltung auf dem Rücksetzeingang, so daß sie bei gleichzeitigem Setz- und Rücksetzsignal definiert zurückgesetzt werden. Dies wird in der vorhandenen Ansteuerung ausgenutzt, wodurch sich die vorgeschaltete AND-Verknüpfung vereinfacht.

Das Signal DISPLAY ist in dem Teil einer Zeile aktiv, in dem das Bild enthalten ist. HA bis HE sind Impulsfragmente, die in der CSync-Erzeugung zur vertikalen Austastlücke zusammengesetzt werden. HA entspricht außerdem in seinem Timing dem H-Puls. Mit CSXMASK werden im Aufnahmemodus die Trabanten, die sich in der V-Lücke befinden aus dem externen CSync-Signal ausmaskiert.

Die Schaltung vor dem Horizontalzähler schaltet diesen, gesteuert durch RECAK, in den Aufnahme- bzw. in den Wiedergabemodus. Im Wiedergabemodus läuft der Zeilenzähler frei. Im Aufnahmemodus wird er auf das externe Videosignal synchronisiert. Dies geschieht durch das Signal CSEXTp, das im Wiedergabemodus am Anfang jeder Zeile den Horizontalzähler zurücksetzt.

Durch das RS-Flip-Flop N2 wird verhindert, daß bei ausbleibenden CSEXTP-Impulsen der Horizontalzähler 'überläuft'. Bei Erreichen seines maximalen Zählerstands von 511 wird er angehalten und erst beim nächsten CSEXTP-Impuls wieder gestartet.

Das Signal MUXRES wird über den Buffer P3 mit VCC bzw. VSS, also mit logisch '1' bzw. '0' verbunden. Für die Implementation der Schaltung liegt es auf VCC und maskiert so den Rücksetzeingang RESI und den Testeingang TESTI, so daß diese im LCA nicht mehr vorhanden sind. Die daraus abgeleiteten Signale SRES und TEST liegen dann fest auf logisch '0'. Dies betrifft die OR-Verknüpfungen vor den Rücksetzeingängen der RS-Flip-Flops, von denen P18, P19, P26 und P34 dadurch entfallen. Sie sorgen während der Simulation dafür, die Schaltung zurückzusetzen, was im LCA ohnehin nach der Konfigurationsphase geschieht.

4.6.3. Vertikalsteuerung (SYNC/LOGIC/2)

N12 und N13 bilden zusammen den Zeilenzähler. Er wird mit dem Systemtakt von 8MHz getaktet, jedoch über das Signal VENAX nur am Anfang jeder Zeile für jeweils einen Taktimpuls freigegeben, so daß er effektiv mit Zeilenfrequenz zählt. Im Wiedergabemodus entspricht VENAX dem Signal VENA, das in der Horizontalsteuerung gebildet wurde.

N49 bildet den Halbbildzähler, das Signal FIELD zeigt durch log. '1' das zweite Halbbild an. Der Zeilenzähler wird nach 313 Zeilen im ersten bzw. 312 Zeilen im zweiten Halbbild zurückgesetzt. Dazu wird durch die AND-Gatter P40 bis P42 der Zählerstand 311 auskodiert, der über den Multiplexer N27 den Zeilenzähler nach den zweiten Halbbild zurücksetzt. Da das erste Halbbild eine Zeile länger ist, wird der Rücksetzimpuls in diesem Fall über N26 um eine Zeile verzögert.

Der Stand des Zeilenzählers, zusammengefaßt in VBUS, wird im Block COMP9 mit CBUS verglichen, der von der CPU mit der vertikalen Position vorbesetzt wurde. Bei Gleichheit wird über den Blockausgang EQ das Signal VSPULS erzeugt, das den vertikalen Start der Einblendung kennzeichnet.

Der Zeilenzähler zählt die Zeilen nicht nach der in der PAL-Norm festgelegten Weise. Um die Erzeugung der V-Lücke im CSync-Signal zu erleichtern, wurde der Zählerstand '0' auf den Beginn der V-Lücke gelegt. Der dadurch entstandene Versatz von 2 Zeilen

kann durch Software leicht ausgeglichen werden. Das Signal ENA ist während der ersten 8 Zeilen (Zeile 0 bis 7) jedes Halbbildes aktiv und veranlaßt in der CSync-Erzeugung die Bildung der vertikalen Austastlücke.

Ausgelöst vom vertikalen Startpuls VSPULS zählt der Zähler N18/N19 die 72 Zeilen, die zu einem abgetasteten Bild gehören. Während dieser Zeilen ist das RS-Flip-Flop N20 gesetzt. Es gibt dadurch das Signal MASK frei, mit dem, abgeleitet aus DISPLAY, das darzustellende Bild maskiert wird, sowohl horizontal als auch vertikal.

Im Aufnahmemodus werden über RECAK einige Schaltungseigenschaften modifiziert. Der Zeilenzähler erhält seine Freigabe nicht mehr vom Horizontalzähler sondern direkt von den H-Impulsen des abzutastenden Videosignals. Da nur jede vierte Zeile abgetastet wird, zählt der Zähler N18/N19 nur mit einem Viertel der H-Frequenz. Dies wird über das Gatter P44 erreicht.

4.6.4. CSync-Erzeugung (SYNC/LOGIC/3)

Aus den in der Horizontalsteuerung erzeugten Signalen HA bis HE werden über die Gatter P54 bis P59 die Signale VA bis VF gebildet. Diese stellen Fragmente der V-Lücke von jeweils einer Zeile Länge dar. Sie werden über die Multiplexer N31 für das erste und N32 für das zweite Halbbild in die zeitlich richtige Reihenfolge gebracht. Das am Ausgang der Multiplexer gewonnene Signal stellt einen V1- bzw. V2-Puls inklusive der Vor- und Nachtrabanten dar. FIELD wählt den jeweils richtigen V-Impuls aus, und ENA schaltet um zwischen V-Lücke und H-Pulsen, die im Signal HA vorliegen. Das Ergebnis ist ein komplettes Composite-Sync-Signal nach PAL-Norm. Es wird auf dem Ausgang CSYNC ausgegeben.

Während der Abtastung eines Videobildes wird vom internen auf das externe CSync-Signal umgeschaltet.

Aus ENA und FIELD werden noch der V-Puls VSYNC und der V1-Puls V1SYNC abgeleitet. Letzterer stellt einen Testausgang dar, er wird ansonsten nicht weiter verwendet. Aus HA wird der H-Puls HSYNC erzeugt. Im Wiedergabemodus wird für jede Zeile ein H-Puls erzeugt, da in jeder Zeile eine neue Bildzeile dargestellt wird. Im Aufnahmemodus wird nur jede vierte Zeile abgetastet. Es wird also auch nur jede vierte Zeile, gesteuert durch HPENA, ein H-Puls generiert.

4.6.5. Aufnahmesteuerung (SYNC/LOGIC/4)

Der Mikroprozessor fordert über RECRQ das Aufzeichnen eines Videobildes an. Diese Anforderung wird von N46 mit dem V-Puls des externen Videosignals synchronisiert und über das nachfolgende Flip-Flop N47 digital differenziert. Da diese Flip-Flops nur einmal pro Halbbild freigegeben werden, ist die Aufnahmebestätigung RECAK für genau ein Halbbild des externen Videos aktiv. An der fallenden Flanke von RECAK kann der Mikroprozessor die erfolgte Aufzeichnung eines Bildes erkennen.

Das aktive Signal RECAK gibt die Signale CSEXTP und VSEXTP frei. Diese steuern im Aufnahmemodus das horizontale bzw. vertikale Timing. Sie werden über digitale Differenzierer aus CSEXTI und VSEXTI gewonnen.

4.6.6. SYNC - Zeitdiagramme

In Abbildung B-11 wird die Erzeugung des V1-Pulses gezeigt. VA bis VF sind die C-Sync-Fragmente, die über den Zeilenzähler, dargestellt als V(8:0) ausgewählt werden. FIELD=0 zeigt das erste Halbbild an. CSYNCO ist das erzeugte Composite-Sync-Signal. Abbildung B-12 zeigt das gleiche für den V2-Puls. In Abbildung B-13 wird das horizontale Timing nochmals im Detail gezeigt.

4.7. Datenfluß- und CPU-Steuerung (PROZ)

Das Modul PROZ kontrolliert die Datenwege zwischen den Komponenten CPU, ADC, DAC und DRAM. Zusätzlich übernimmt es Dekoderrfunktionen und die Interrupt-Steuerung.

PROZ ist bereits weitestgehend für die Erweiterung der Schaltung auf Farbe vorbereitet

4.7.1. Top-Level (PROZ/1)

4.7.1.1. Datenbusse

Im Modul PROZ laufen alle Datenbusse des Bildspeichers zusammen. An die Busse sind folgende Komponenten angeschlossen:

```
AD(7:0)I: A/D-Wandler
P(7:0)IO: Mikroprozessor
L(7:0)IO: Luminanz-Teil des D-RAMs
C(7:0)IO: Chroma-Teil des D-RAMs
DA(7:0)O: D/A-Wandler
```

4.7.1.2. Eingangssignale

An den Eingang CLKI ist der Systemtakt von 32MHz angeschlossen. PHI2I, RWNi und A3 bis A0 kommen direkt vom Mikroprozessor. CSNI wird vom Adreßdekoder des CPU-Moduls geliefert. Es bestimmt die Basisadresse des Bildspeichers im Adreßraum des Mikroprozessors. VPULSI und VPEXTI sind der interne und der externe V-Puls. Sie können in PROZ Prozessorinterrupts auslösen.

ADCLKI und DACLKI steuern Zwischenspeicher in den Datenwegen vom A/D- und zum D/A-Wandler. LCILI entsprechend für den Luminanz- und den Chroma-Bus. Über LCOEI ist es möglich, diese beiden Busse hochohmig zu schalten. DASELI maskiert den D/A-Bus. PSRPI schaltet im Wiedergabemodus den D/A-Bus zwischen Luminanz- und Chroma-Bus um.

RECAKI und CPIXAKI informieren PROZ über den gerade gewählten Betriebsmodus.

4.7.1.3. Ausgangssignale

Über CPIXO und RECRQO werden Betriebsmodusänderungen angefordert. WPRASO, WPCASO und WPROW1O sind Latchimpulse, über die die Eingangsregister der Module CLK2, ADRE und SYNC vom Prozessor-Datenbus aus beschrieben werden können. CSCPUO meldet im CPU-Modus einen Bildspeicherzugriff der Mikroprozessors, RAMRWNO dessen Richtung (lesen oder schreiben). IRQO schließlich leitet Interrupt-Anforderungen an den Mikroprozessor weiter.

4.7.2. Datenflußsteuerung (PROZ/LOGIC/1)

Bei aktivem CSN-Signal des CPU-Moduls liefert der Adreßdekoder N1/N2 eines der Signale Y0 bis Y15. Y0, Y1 und Y2 erzeugen bei Schreibzugriffen (RWN=0) und aktivem PHI2-Signal die Schreibimpulse WPRAS, WPCAS und WPROW1.

Y4 und Y5 steuern die Erzeugung der Signale CPIX und RECRQ. Schreibzugriffe setzen das entsprechende Signal; Lesezugriffe löschen es. Das Setzen eines der Signale führt automatisch zum Löschen des jeweils anderen. Dadurch kann der verbotene Modus (CPIX=1 und RECRQ=1) nicht angewählt werden.

Über Y8 kann auf den Zwischenspeicher für den Luminanz-Anteil des Bildspeichers zugegriffen werden. Dabei wird über CSCPU auch ein D-RAM-Speicherzyklus ausgelöst, der das nächste Pixel schreibt bzw. liest. Y9 ermöglicht den Zugriff auf den Zwischenspeicher der Chroma-Information. Dabei wird kein Speicherzyklus erzeugt, wodurch in zwei Schritten über den 8Bit-Datenbus des Mikroprozessors auf den 16 Bit breiten Speicherbus zugegriffen werden kann.

Y11 kontrolliert ein Parallelport in PROZ, über das die Interruptverwaltung abgewickelt wird. Y12 dient zum Zurücksetzen von Interruptanforderungen.

4.7.3. Datenmultiplexer (PROZ/LOGIC/2)

Dieser Schaltungsteil stellt eine Art "Kreuzschienenverteiler" dar, über den die verschiedenen Busse miteinander verbunden werden können. Da innerhalb der LCA keine bidirektionalen Busse möglich sind, wurden die externen Busse in Ein- und Ausgangssignale aufgeteilt.

Der Multiplexer N15 bis N22 schaltet den Datenweg zum Luminanzteil des D-RAMs zwischen A/D-Wandler (Aufnahme) und zwischengespeichertem Prozessorbus (CPU-Modus, schreibende Zugriffe) um. Entsprechendes gilt für den Multiplexer N23 bis N30 und den Chroma-Anteil.

N31 bis N38 haben eine doppelte Funktion. Während der Wiedergabe schalten sie den D/A-Wandler zwischen Luminanz und Chroma um. Das gleiche geschieht bei lesenden CPU-Zugriffen im CPU-Modus. Während dieser Zeit kann der D/A-Bus über DASEL maskiert werden.

N39 bis N46 wählt bei lesenden CPU-Zugriffen zwischen Speicherzugriffen und dem Parallelport. Über P52 bis P59 kann schließlich der D/A-Wandler-Bus maskiert werden, um störende Ausgaben auf den Bildschirm, z.B. im CPU-Modus, zu verhindern.

4.7.4. Interrupt-Steuerung (PROZ/LOGIC/3)

In PROZ können zwei Interrupt-Quellen verwaltet werden. VPULS zeigt im Wiedergabemodus mit der steigenden Flanke den Beginn der V-Lücke an, um z.B. das Beschreiben der ROW-, COL- oder Zeilenregister zu veranlassen. Eine fallende Flanke an RECAK kennzeichnet die erfolgte Aufzeichnung eines Bildes.

VPULS ist ein Interrupt-Flip-Flop zugeordnet, das, falls es gesetzt ist, eine bestehende Interruptanforderung signalisiert. Sein Zustand kann über das Parallelport PPBUS auf Bit 7 ausgelesen werden. Über das Latch N50 wird die Interruptanforderung auf die IRQ-Leitung des Mikroprozessors durchgeschaltet. Die Anforderung kann durch Schreiben eines Bytes mit gesetztem 7. Bit in das Interrupt-Rücksetz-Register gelöscht werden. N51 und N52 selektieren für die Interruptanforderung die steigende Flanke von VPULS.

Die Interruptverwaltung für RECAKI ist identisch aufgebaut. Hier ist Bit 6 der Lese- und Schreibregister maßgebend.

4.7.5. PROZ - Zeitdiagramme

Abbildung B-14 zeigt eine Folge verschiedener CPU-Zugriffe. Im Einzelnen sind dies:

- Schreibzugriff auf PMCRO (9)
- Schreibzugriff auf PMRAS (0): Bildnummer setzen
- Schreibzugriff auf PMCAS (1): Versatz setzen
- Schreibzugriff auf PMROW1 (2): Zeile setzen
- Schreibzugriff auf PMCPIX (4): CPU-Modus einschalten
- Lesezugriff auf PMCPIX (4): CPU-Modus ausschalten
- Schreibzugriff auf PMREC (5): Aufnahmemodus einschalten
- Lesezugriff auf PMREC (5): Aufnahmemodus ausschalten
- Schreibzugriff auf PMLUM (8): Pixel schreiben
- Lesezugriff auf PMLUM (8): Pixel lesen
- Schreibzugriff auf PMCRO (9): Farbanteil schreiben
- Lesezugriff auf PMCRO (9): Farbanteil lesen
- Lesezugriff auf PMLUM (8): Pixel lesen
- Lesezugriff auf PMCRO (9): Farbanteil lesen

4.8. Dynamischer Speicher (DRAM)

Der dynamische Speicher ist mit Megabit-D-RAMs aufgebaut. Sie bieten den derzeit günstigsten Preis pro Bit. Zum Einsatz kommen sie in der Organisation 256K*4. Als maximale Zugriffszeit werden 120ns gefordert. In der vorliegenden Schaltungsversion wird nur das Helligkeitssignal mit 8Bit abgespeichert. Dazu sind zwei dieser ICs erforderlich. In der Farbversion wären es dann vier, um auf 16Bit Wortbreite zu kommen, oder, falls sich für Helligkeit und Farbe je 6Bit Auflösung als ausreichend herausstellt, drei ICs. Bei Einsatz einer Datenreduktion könnten diese wieder auf 2 ICs vermindert werden.

4.9. Mikroprozessor (CPU)

Die Mikroprozessorplatine wie auch die Anzeige, das Netzteil und das Gehäuse des Prototypen wurden aus dem Seriengerät 'TC12' der Firma Alpermann&Velte entnommen. Die CPU-Platine enthält Takt- und Reset-Erzeugung, die Adreßdekodierung, 2*8KByte EPROM, 2KByte RAM, 3 VIAs 65SC22 sowie Treiber für einen externen CPU-Bus, an den die Bildspeicherplatine angeschlossen wird. Von den VIAs kommt einer zur Ansteuerung der Tastatur- und Anzeigeplatine zum Einsatz.

4.10. D/A-Wandler (DAC)

Der D/A-Wandler ist mit im A/D-D/A-Wandler UVC3130 integriert. Von den zur Verfügung stehenden 10Bit werden nur die höherwertigen 8Bit ausgenutzt.

5. Programmbeschreibung

5.1. Verwendete Hardware

Die wesentlichen Funktionen des Bildspeichers werden von der Hardware übernommen. Der Mikroprozessor muß lediglich die Nummer des darzustellenden Bildes, dessen horizontalen Versatz und die vertikale Position in die dafür vorgesehene Register schreiben sowie die Aufzeichnung von Bildern überwachen. Nach dem Einschalten müssen außerdem die Konfigurationsdaten in die LCAs geladen werden.

Als Mikroprozessor wurde der 65SC02-2 von GTE gewählt. Er wird seit langem bei Alpermann&Velte eingesetzt. Es konnten daher einige Standardbausteine aus der Serienfertigung der Firma entnommen werden, insbesondere die CPU- und die Anzeige/Tastatur-Platine.

Die Anzeige bietet zwei achtstellige 7-Segment-Anzeigen, von denen eine benutzt wird, sowie einige LEDs. Die Tastatur umfaßt eine Dezimaleingabe sowie zahlreiche Funktionstasten.

5.2. Bedienung

Anzeige und Tastatur sind wie folgt aufgeteilt:

Anzeige:

```

-----
!  Anzeige  ! !  Anzeige  ! !  Anzeige  ! !  Anzeige  !
!   Bild    ! ! Horizontal ! ! Vertikal  ! ! Eingabe  !
-----

```

Tastatur:

```

-----
!Bild!Bild!Bild! Hor! Hor! Hor!Vert!Vert!Vert!    ...    ! REC!
! -  !SET ! +  ! -  !SET!  +  ! -  !SET ! +  !    ...    !   !
-----

```

Die Anzeigen 'Bild', 'Horizontal' und 'Vertikal' zeigen die aktuelle Bildnummer, den horizontalen Versatz und die vertikale Position an. In der Anzeige 'Eingabe' ganz rechts kann über die Zehnertastatur eine zweistellige Dezimalzahl eingegeben werden.

Durch Drücken einer der 'SET'-Tasten wird dieser Wert in die entsprechende Anzeige übernommen. Falls dabei der erlaubte Bereich einer der Anzeigewerte überschritten wird, erfolgt eine automatische Korrektur.

Mit den '+'- und '-'-Tasten kann der eingestellte Wert erhöht bzw. vermindert werden. Auch hier werden Bereichsüberschreitungen korrigiert.

Soll ein Bild aufgezeichnet werden, muß dessen Nummer in die Eingabeanzeige geschrieben und anschließend die 'REC'-Taste gedrückt werden. Zur Kontrolle leuchtet während der Aufnahme die LED in der 'REC'-Taste kurz auf.

Der genaue Programmablauf ist in Flußdiagrammen im Anhang sowie im Programmlisting beschrieben. Die Datei FREMD.S enthält dabei die Programmteile, die aus vorhandenen Programmen der Firma Alpermann&Velte entnommen wurden.

Die XILINX-Konfigurationsdaten werden vom Linker zum eigentlichen Programm hinzugeladen.

6. Ungelöste Probleme

In der vorliegenden Version des Bildspeichers sind einige Punkte noch nicht befriedigend gelöst. Im einzelnen sind dies:

6.1. Modus-Umschaltungen

Die Synchronisation der Modusumschaltungen mit der RAS/CAS-Erzeugung konnte in der zur Verfügung stehenden Zeit nicht letztendlich gelöst werden. Durch die Realisation der CAS/CAS-Erzeugung mittels Schieberegister dürfen Modusumschaltungen nur erfolgen, wenn die Schieberegister leergetaktet sind, d.h. wenn sich die Schaltung im IDLE-Zustand befindet. Die dazu nötige Steuerung muß noch implementiert werden. Sie wurde durch die Einführung von Modusanforderungs- (CPIXRQ, RECRQ) und -bestätigungssignalen (CPIXAK, RECAK) jedoch schon vorbereitet.

6.2. Kritisches Timing

Durch die Verwendung eines Systemtakts von 32MHz in drei der vier LCAs stehen pro Taktperiode ca. 31ns zur Verfügung. Viele Signale müssen in dieser Zeit vom Ausgang eines Flip-Flops über einige Gatter an den D-Eingang eines anderen Flip-Flops gelangen. Durch die Architektur der 3000er-LCAs ist die Durchlaufzeit für eine kombinatorische Verknüpfung zwar konstant, solange sie innerhalb eines CLBs aus maximal fünf Eingangssignalen gebildet wird. Die Verzögerungszeiten der Leitungen zwischen den Blöcken hängt jedoch stark von deren Länge ab. Diese wiederum wird von der Anordnung der Blöcke zueinander beeinflusst.

Für die verwendete 70MHz-Ausführung der 3030er-LCAs können folgende Verzögerungszeiten angenommen werden: [3]

Verzögerung vom Ausgang eines CLB-Flip
-Flops bis zum Ausgang des Blocks: 8ns

Verzögerung vom Blockeingang über eine
kombinatorische Verknüpfung bis zum D-
Eingang eines CLB-Flip-Flops (incl.
Setup-Time): 8ns

Bei 31ns Zykluszeit stehen für die Leitung zwischen den Blöcken also noch $(31-8-8)\text{ns} = 15\text{ns}$ zur Verfügung. Dies gilt jedoch nur für den günstigen Fall, in dem die Verknüpfung vollständig in einem CLB untergebracht werden kann. Kompliziertere Verknüpfungen müssen in mehreren CLBs kaskadiert werden, wodurch je Stufe eine Leitungsverzögerung und eine CLB-Durchlaufverzögerung von 9ns hinzukommt.

Dies betrifft insbesondere das LCA ADRE, in dem durch die verwendeten Addierer und Multiplizierer komplexe Verknüpfungen enthalten sind. Diese müssen zwar nicht in einem Taktzyklus bearbeitet werden (die schnellsten Vorgänge in ADRE laufen in 5 Taktzyklen = einer CAS-Periode ab), führen aber dennoch dazu, daß dieses LCA unter Normalbedingungen ($V_{CC}=5\text{V}$, $T_{Umg}=20^\circ\text{C}$) nicht einwandfrei arbeitet. Erst bei Kühlung des LCAs auf ca. -20°C stellt sich die gewünschte Funktion ein.

Für dieses Problem stehen mehrere Lösungsansätze zur Verfügung. Zunächst müssen durch Simulation des realen Timings des LCAs, inklusive der Leitungsverzögerungen, die kritischen Pfade identifiziert werden. Diese können durch Pipelinig-Methoden in mehrere unkritische Pfade aufgeteilt werden, wie es bereits jetzt an anderen Stellen der LCAs durchgeführt wurde. Mit den zur Verfügung stehenden Hilfsmitteln war eine solche Simulation nicht möglich, sie befindet sich bei der Fa. XILINX in Vorbereitung. Auf diese Weise könnte auch das Timing der restlichen Schaltung sichergestellt werden, das unter Worst-Case-Bedingungen ebenfalls kritisch werden könnte.

Die Platzierung und das Routing der LCAs wurde bisher von einem Autorouter übernommen. Bei der Analyse der Ergebnisse zeigte sich, daß diese durch manuelles Eingreifen wesentlich verbessert werden können. Dies setzt jedoch ebenfalls die Kenntnis der kritischen Pfade voraus.

Falls durch die genannten Maßnahmen das Timing nicht sichergestellt werden kann, ist auch noch der Einsatz der 100MHz-Ausführung der LCAs denkbar. Diese sind als 3030er LCA jedoch erst für KW21 in Musterstückzahlen angekündigt.

Schließlich ist ohnehin die Umsetzung der gesamten Schaltung in ein Gate-Array vorgesehen. Je nach verwendeter Technologie sind diese deutlich schneller als die verwendeten LCAs, was ebenfalls die Timing-Probleme lösen könnte. Zusätzlich kann bei der Gate-Array-Entwicklung von vorneherein das reale Zeitverhalten simuliert werden, wodurch das Timing wirklich sichergestellt werden kann.

6.3. Aufnahmesteuerung

Während der Aufzeichnung eines Bildes wird vom internen auf das externe CSync-Signal umgeschaltet. Da das externe Videosignal im Allgemeinen nicht auf das interne synchronisiert ist, fällt ein angeschlossener Monitor während der Aufnahme aus der Synchronisation. Wünschenswert wäre eine auch während der Aufnahme durchlaufende Synchronpulserzeugung.

6.4. Umrahmung

Zur besseren Abgrenzung der angezeigten Bilder untereinander und zum Hintergrund sollen diese mit einem möglicherweise farbigen Rahmen begrenzt werden. Derzeit ist während der Lücke zwischen den Einzelbildern das Ausgangssignal nicht definiert.

7. Zusammenfassung

7.1. Einsatzgebiete

In der vorliegenden Arbeit wurde gezeigt, wie durch Einsatz moderner Technologien (LCAs, Megabit-D-Rams) die Problemstellung 'Digitaler Bildspeicher' mit relativ geringem externen Schaltungsaufwand (der Digitalteil besteht aus nur 8 ICs!) gelöst werden konnte. Durch die derzeit noch recht hohen Preise der LCAs (Stückpreis der eingesetzten XC3030PC84-70 = ca. 160DM) ist die vorliegende Schaltung nur für den Einsatz im Profi- und Semiprofibereich interessant.

Wird die in den LCAs enthaltene Schaltung jedoch in ein Gate-Array umgesetzt (Stückpreis ca. 20DM bis 50DM, je nach Komplexität und Gehäuse), wird auch die Verwendung in Consumer-Geräten möglich. Durch dort erreichte Stückzahlen kann der Bildspeicher dann wiederum im Profi-Bereich günstiger angeboten werden.

Die Implementierung einer digitalen Schaltung in ein LCAs ähnelt der Gate-Array-Entwicklung. Sie hat jedoch den Vorteil, daß ein LCA unmittelbar in realer Umgebung getestet werden kann. Damit stellen LCAs eine gute Vorstufe zur Gate-Array-Entwicklung dar.

7.2. Ausblick

Für den Einsatz des Bildspeichers in einem Schnittsystem ist die Möglichkeit wünschenswert, Texte und Zahlenwerte zusammen mit den gespeicherten Bildern ausgeben zu können. Speicherplatz in den D-Rams wäre für diese Informationen noch vorhanden: für jeweils drei Bildzeilen werden 16 Bytes nicht genutzt.

Um den vorhandenen Speicher besser auszunutzen bzw. mit weniger Speicher-ICs auszukommen, bietet sich der Einsatz einer Datenreduktion an. Dazu wurden im Vorfeld der Diplomarbeit bereits einige Studien durchgeführt, die vermuten lassen, daß statt mit derzeit 16 Bit pro Pixel mit 8 Bit auskommen werden kann. Die notwendige Reduktionsschaltung würde bei Implementierung des Digitalteils in ein Gate-Array die Produktkosten nicht oder nur unwesentlich erhöhen und dadurch den Einsatz des Bildspeichers in Consumer-Produkten wesentlich erleichtern.

A. Schaltpläne

Die Schaltpläne sind nach folgendem Schema hierarchisch organisiert. Dabei ist links die höchste, rechts die niedrigste Hierarchiestufe.

```
EDDI3/1
```

```

!
!----- CLK2/1
!           !----- CLK2/LOGIC/1, /2, /3
!                   !----- CLK2/LOGIC/WAITZYK/1
!                   !----- CLK2/LOGIC/RASZYK/1
!                   !----- CLK2/LOGIC/CASZYK/1
!                   !----- CLK2.LOGIC/RFSHZYK/1
!
!----- ADRE/1
!           !----- ADRE/LOGIC/1, /2, /3
!                   !----- ADRE/LOGIC/ADD_1/1
!                   !----- ADRE/LOGIC/ADD_2/1
!                   !----- ADRE/LOGIC/MUL20/1
!                   !----- ADRE/LOGIC/MUL80/1
!
!----- SYNC/1
!           !----- SYNC/LOGIC/1, /2, /3, /4
!                   !----- SYNC/LOGIC/COMP9/1
!
!----- PROZ/1
!                   !----- PROZ/LOGIC/1, /2, /3

```

A.1. Bildspeicher, Top-Level (EDDI3)

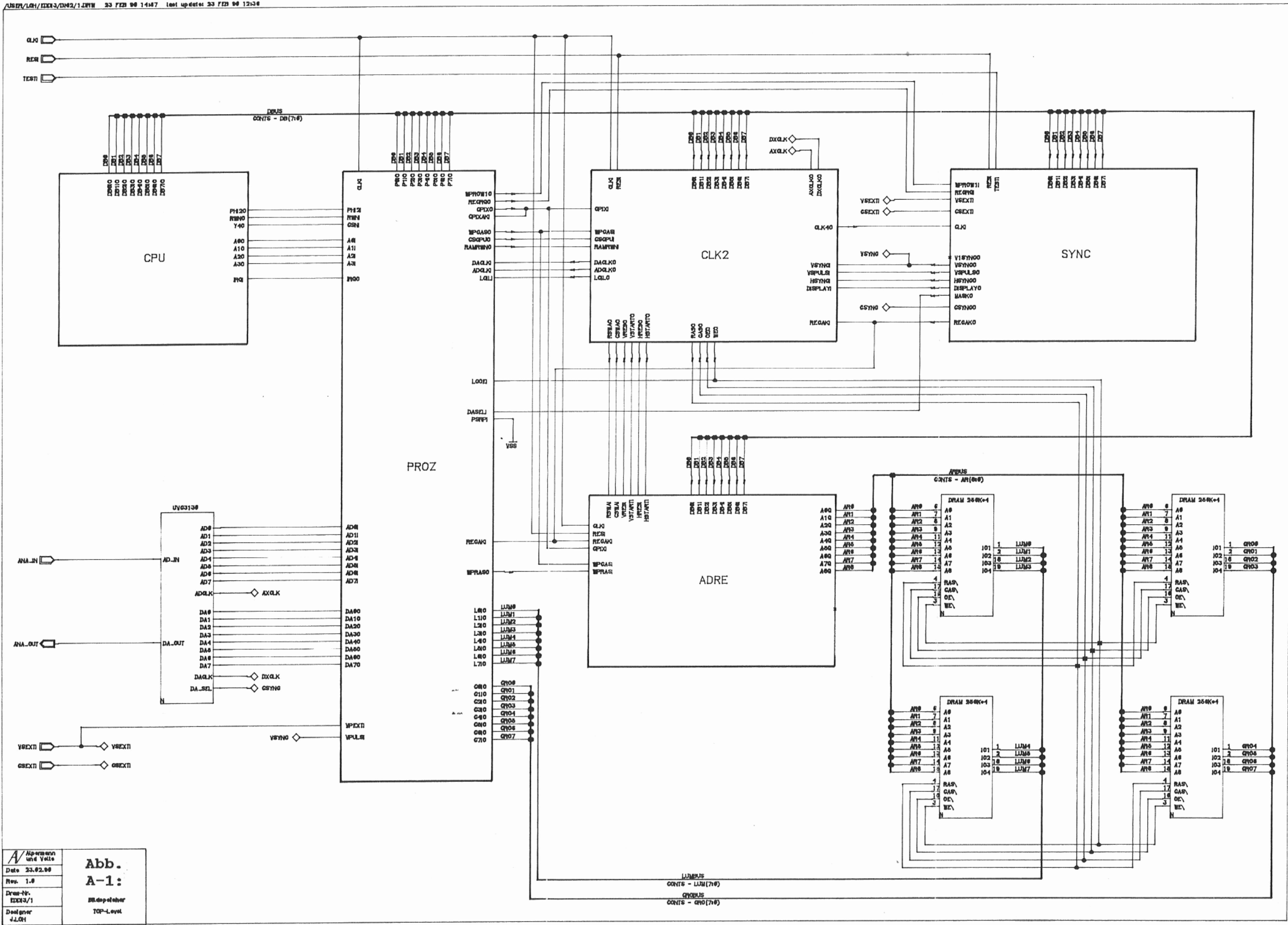
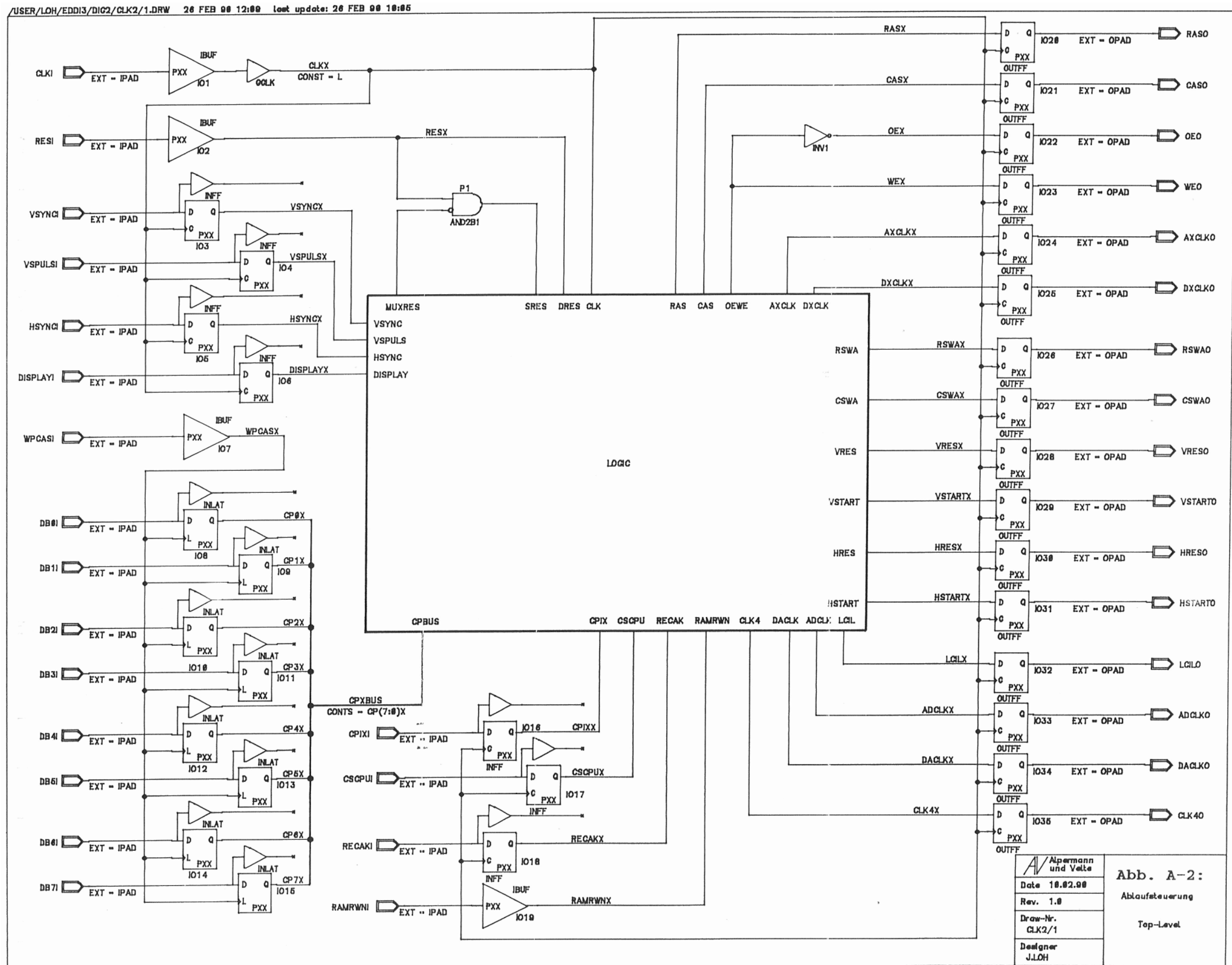


Abb. A-1: Übersichtsschaltplan (EDDI3/1)

A.2. Ablaufsteuerung (CLK2)



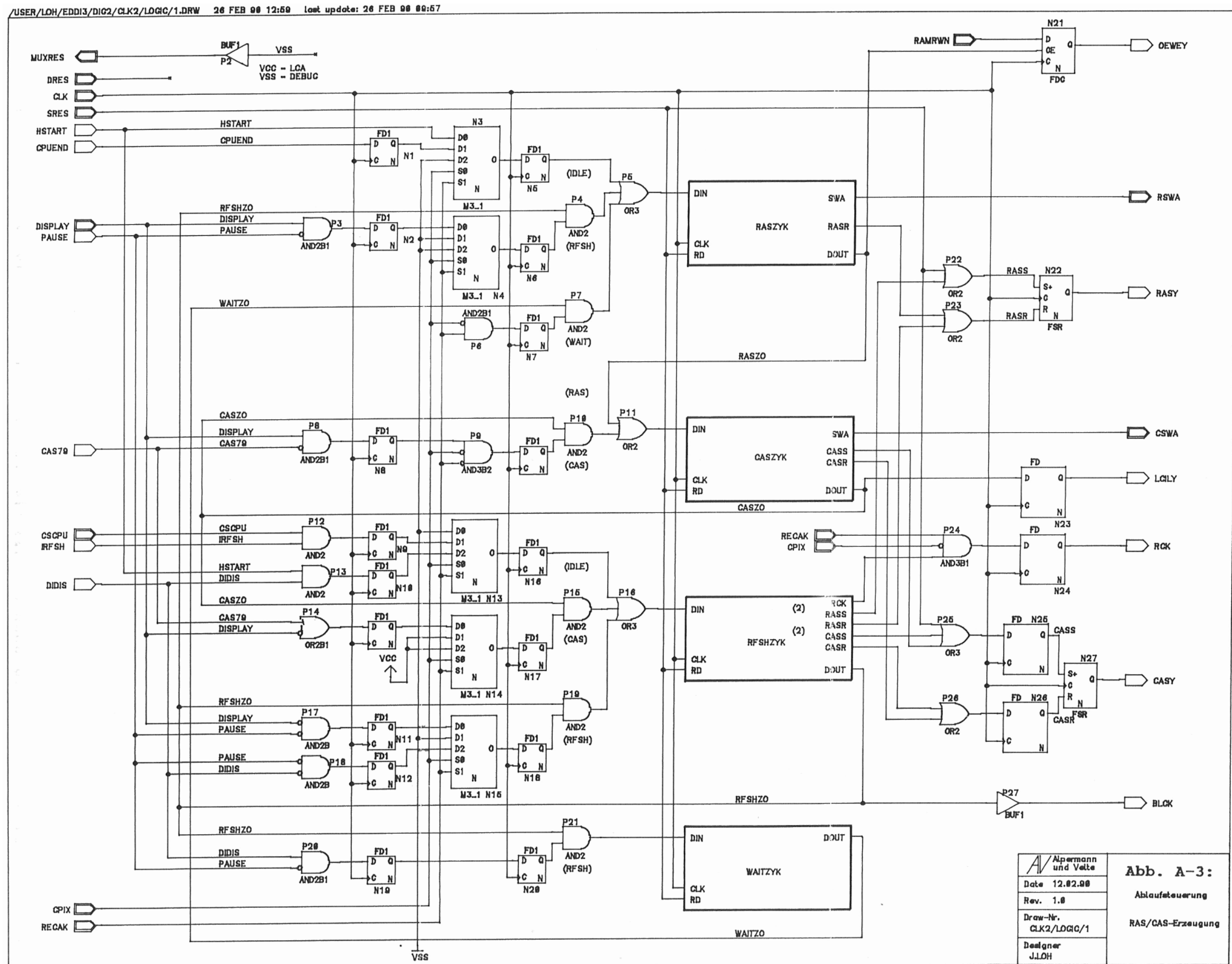
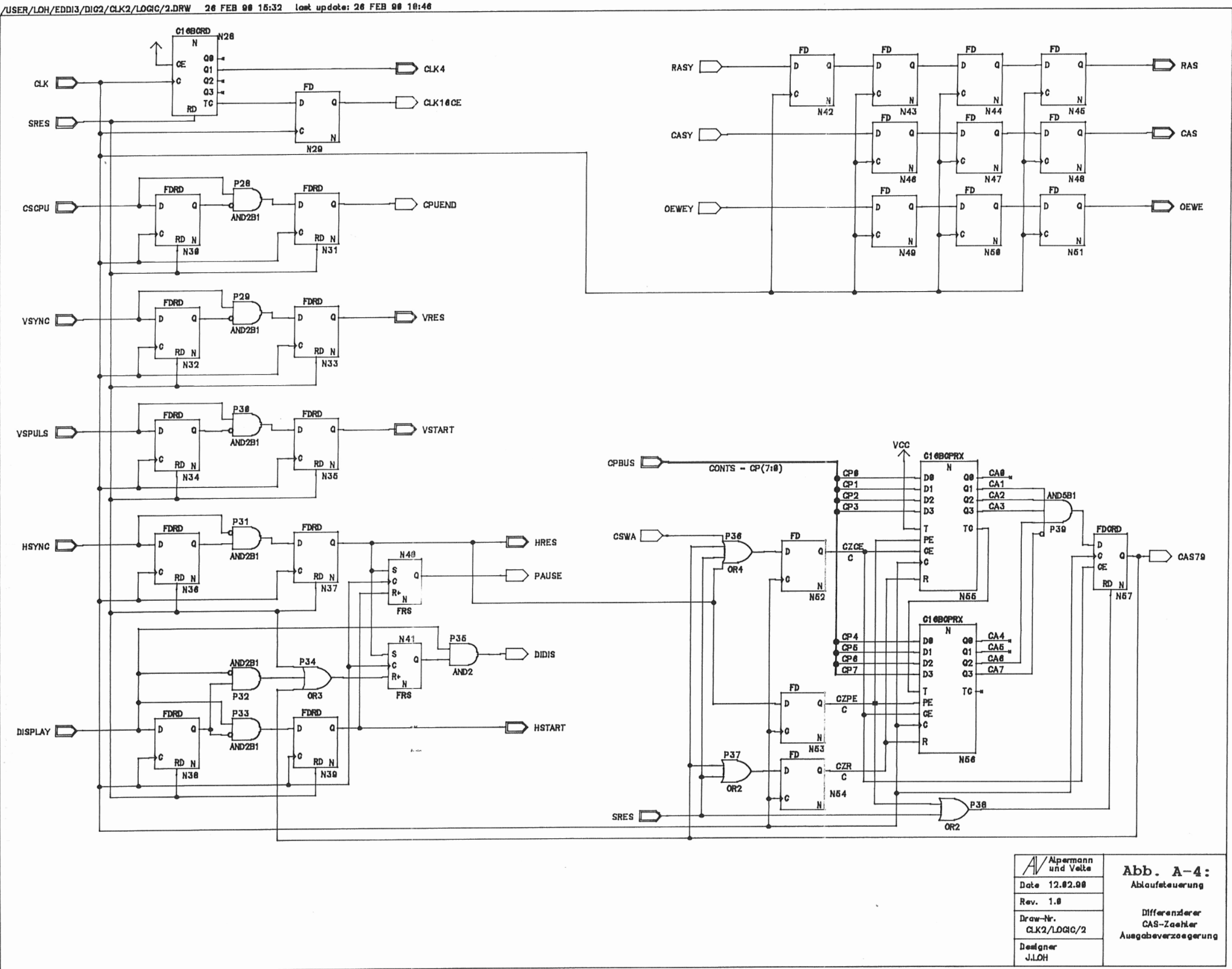
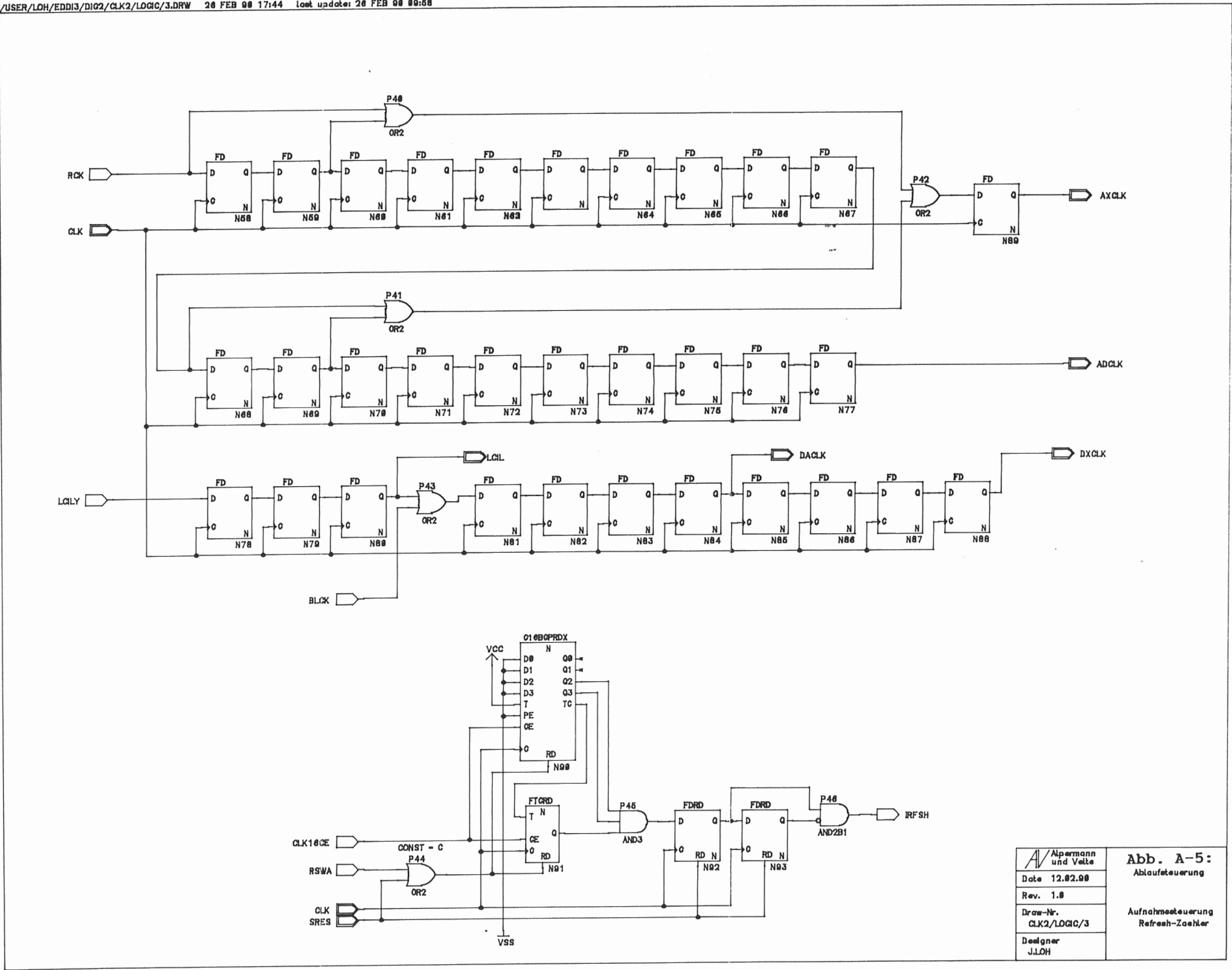


Abb. A-3: RAS/CAS-Erzeugung (CLK2/LOGIC/1)





/USER/LOH/EDDIS/DIG2/CLK2/LOGIC/WAITZYK/1.DRW 28 FEB 90 19:34 last update: 28 FEB 90 09:37

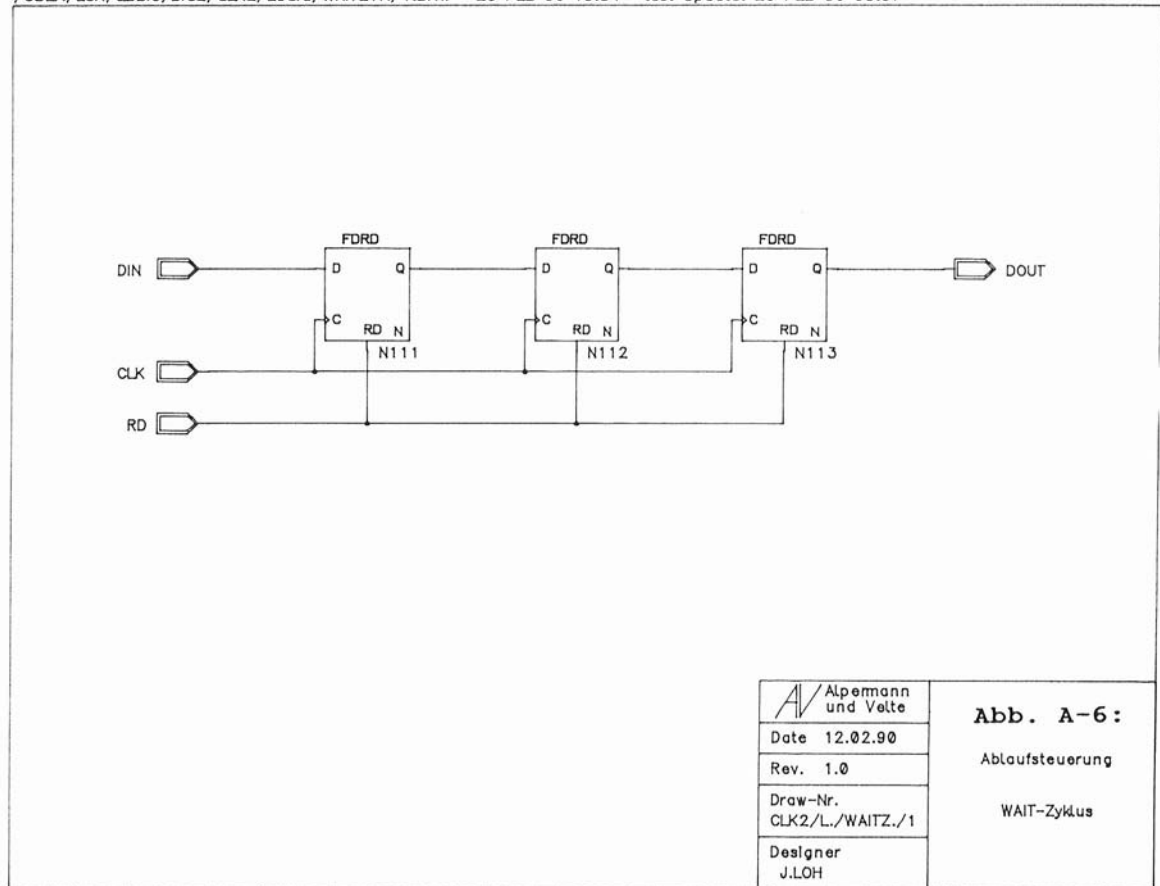


Abb. A-6: Wartezyklus (CLK2/LOGIC/WAITZYK/1)

/USER/LOH/EDDIS/DIG2/CLK2/LOGIC/RASZYK/1.DRW 28 FEB 90 18:26 last update: 28 FEB 90 09:39

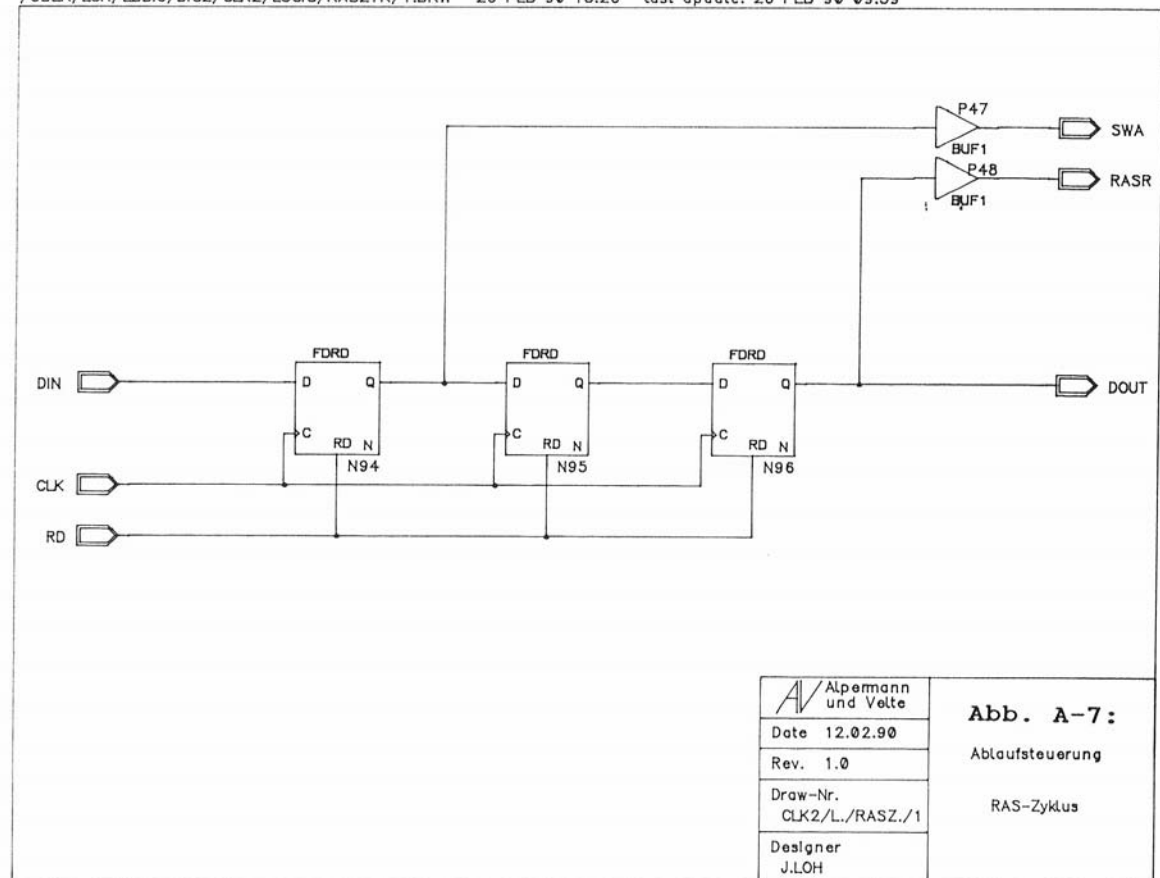
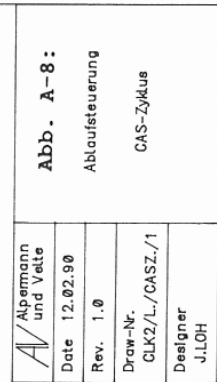
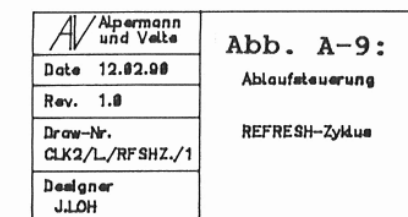


Abb. A-7: RAS-Zyklus (CLK2/LOGIC/RASZYK/1)



A-8



A-9

A.3. Adreßerzeugung (ADRE)

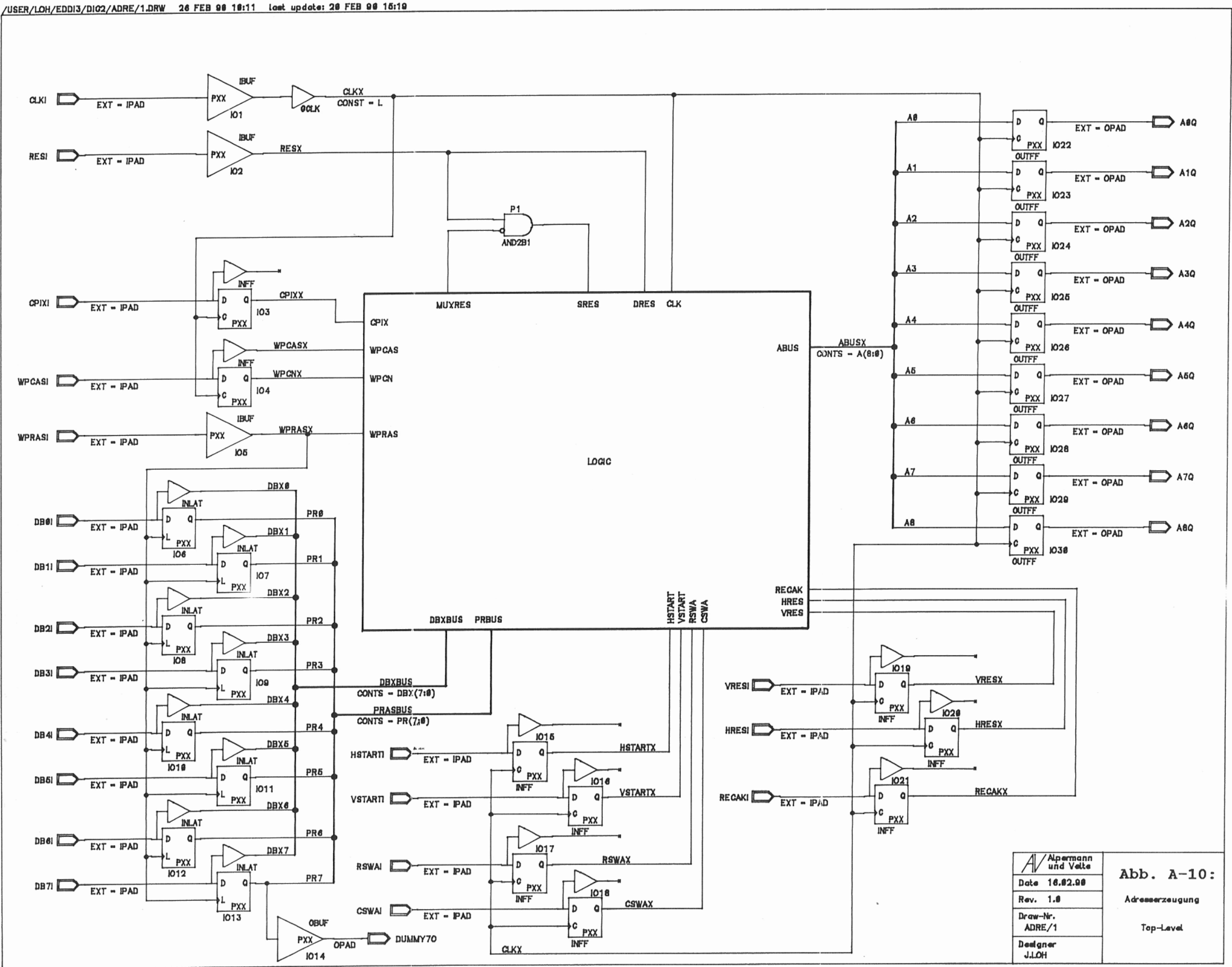


Abb. A-10: Adreßerzeugung, Top-Level (ADRE/1)

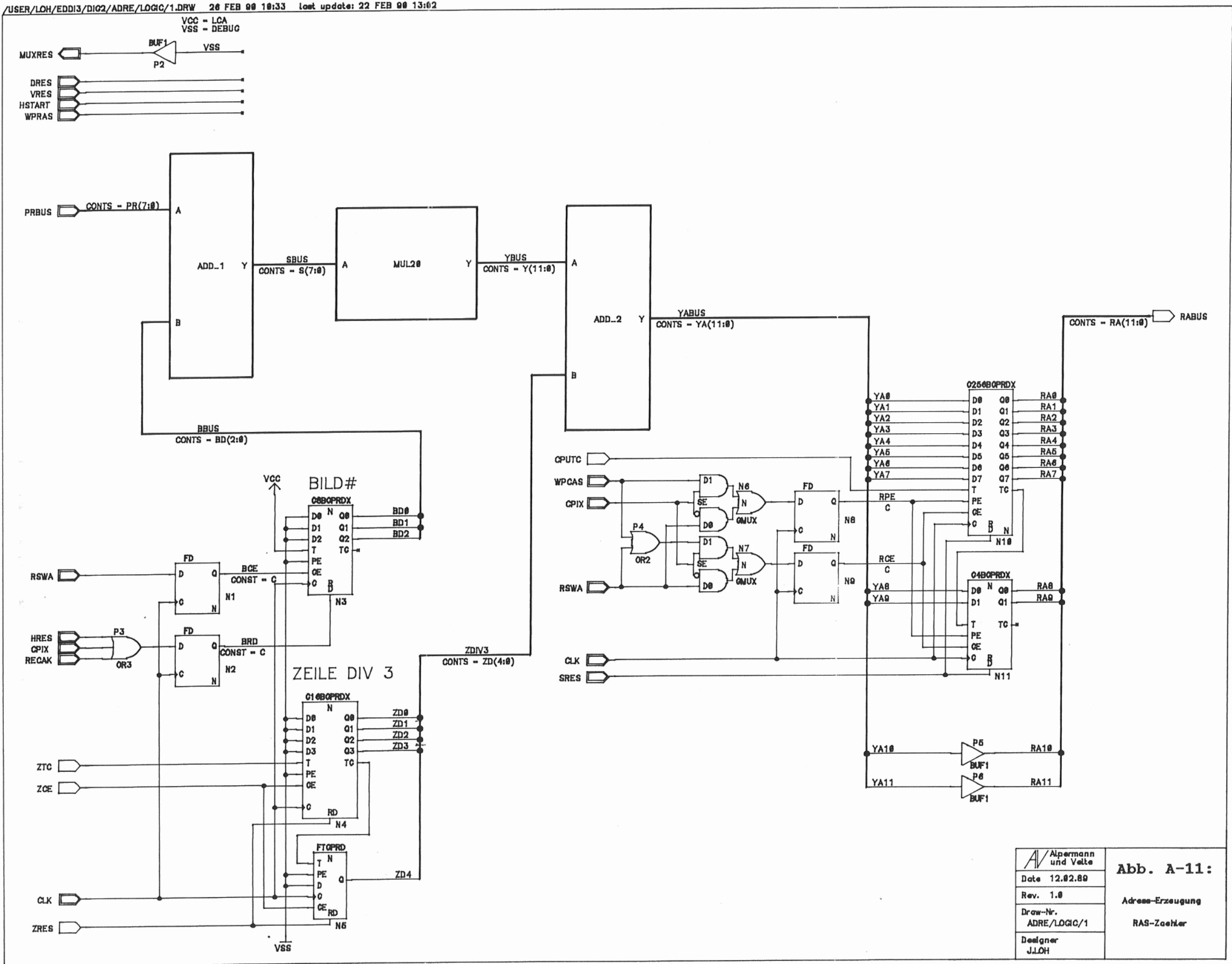


Abb. A-11: RAS-Zähler (ADRE/LOGIC/1)

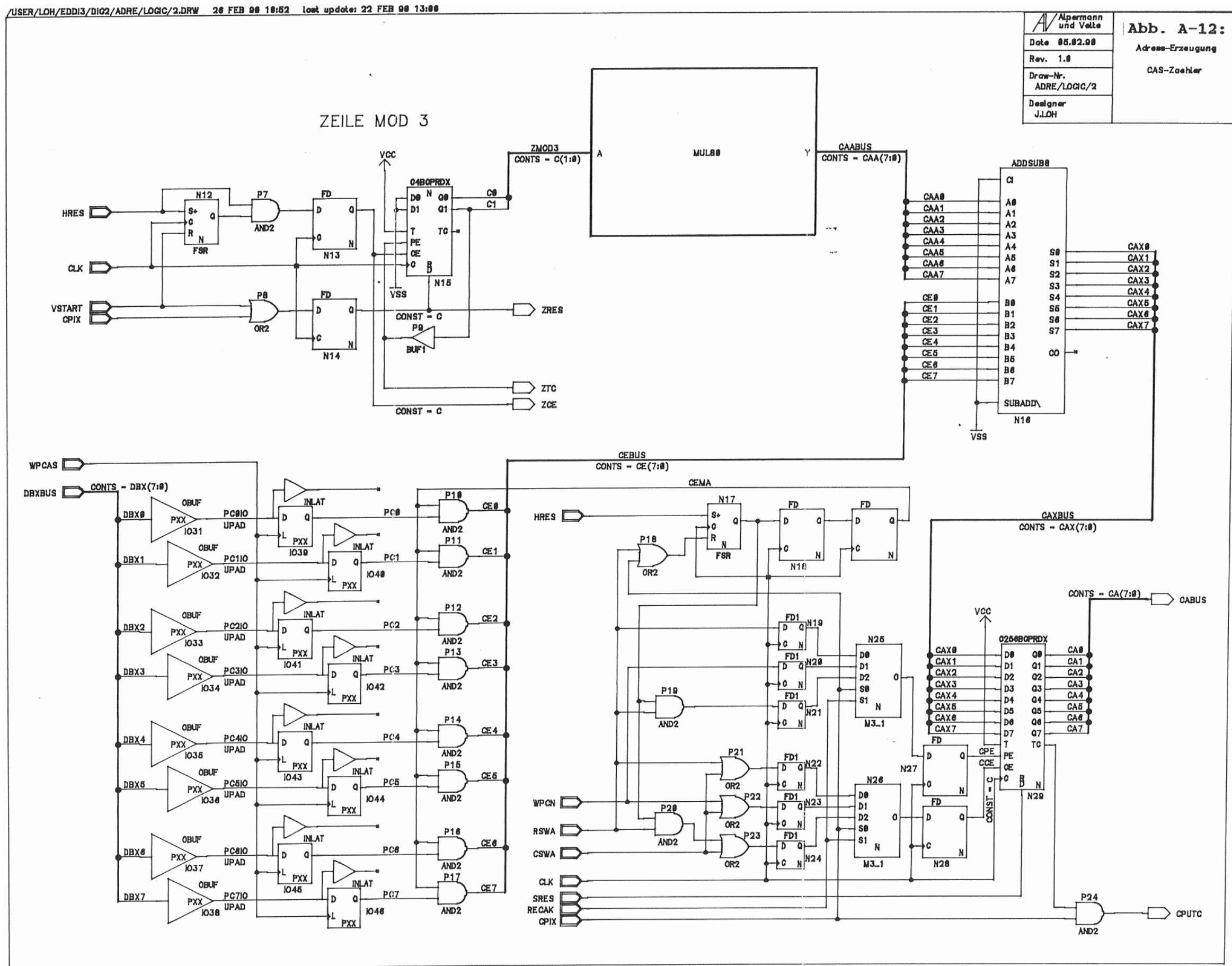


Abb. A-12: CAS-Zähler (ADRE/LOGIC/2)

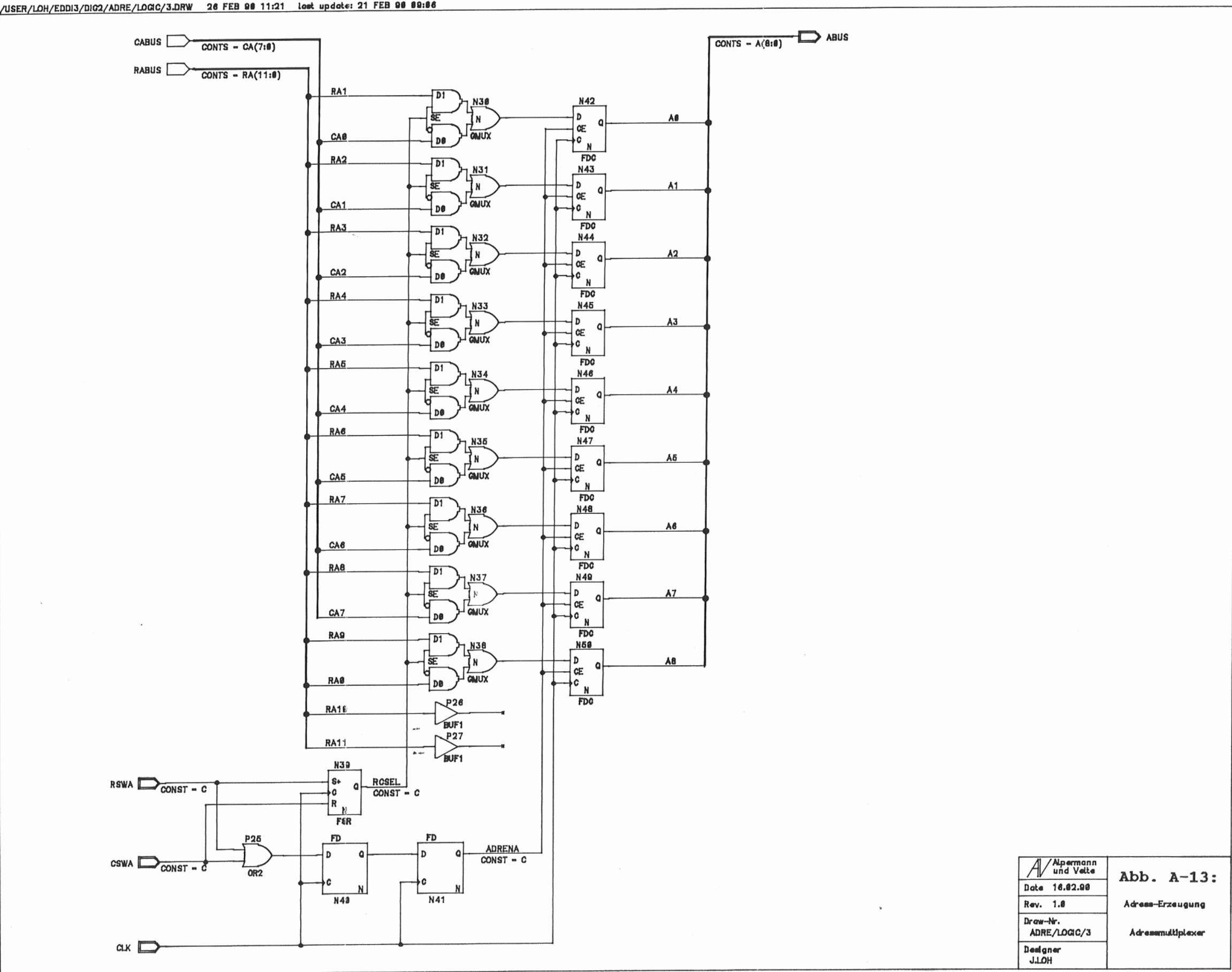


Abb. A-13: Adressmultiplexer (ADRE/LOGIC/3)

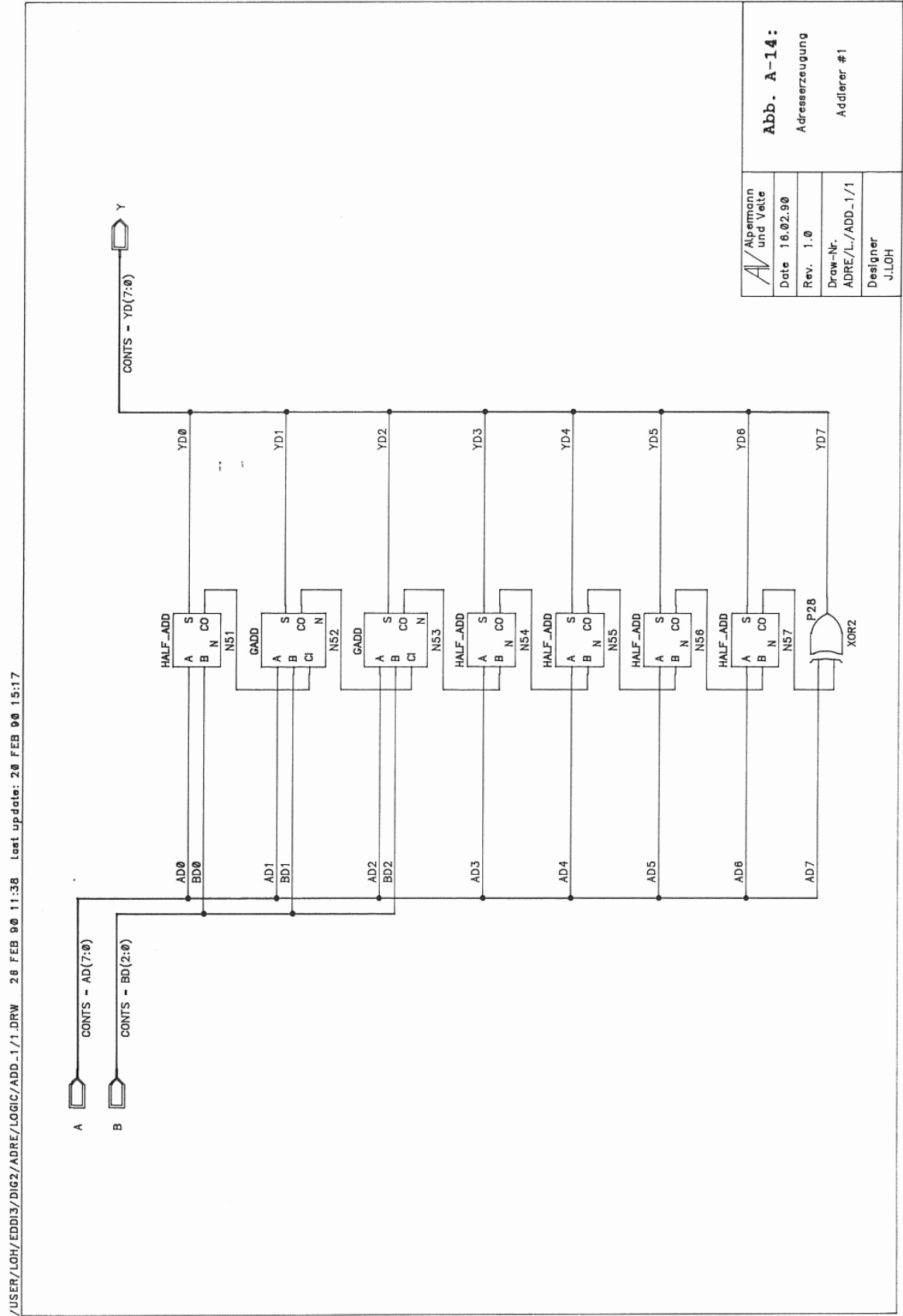
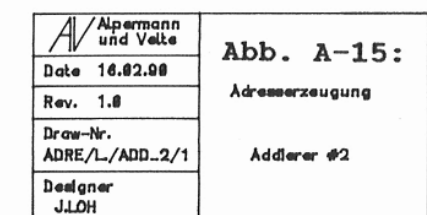


Abb. A-14: Addierer Nr.1 (ADRE/LOGIC/ADD_1/1)



A-15

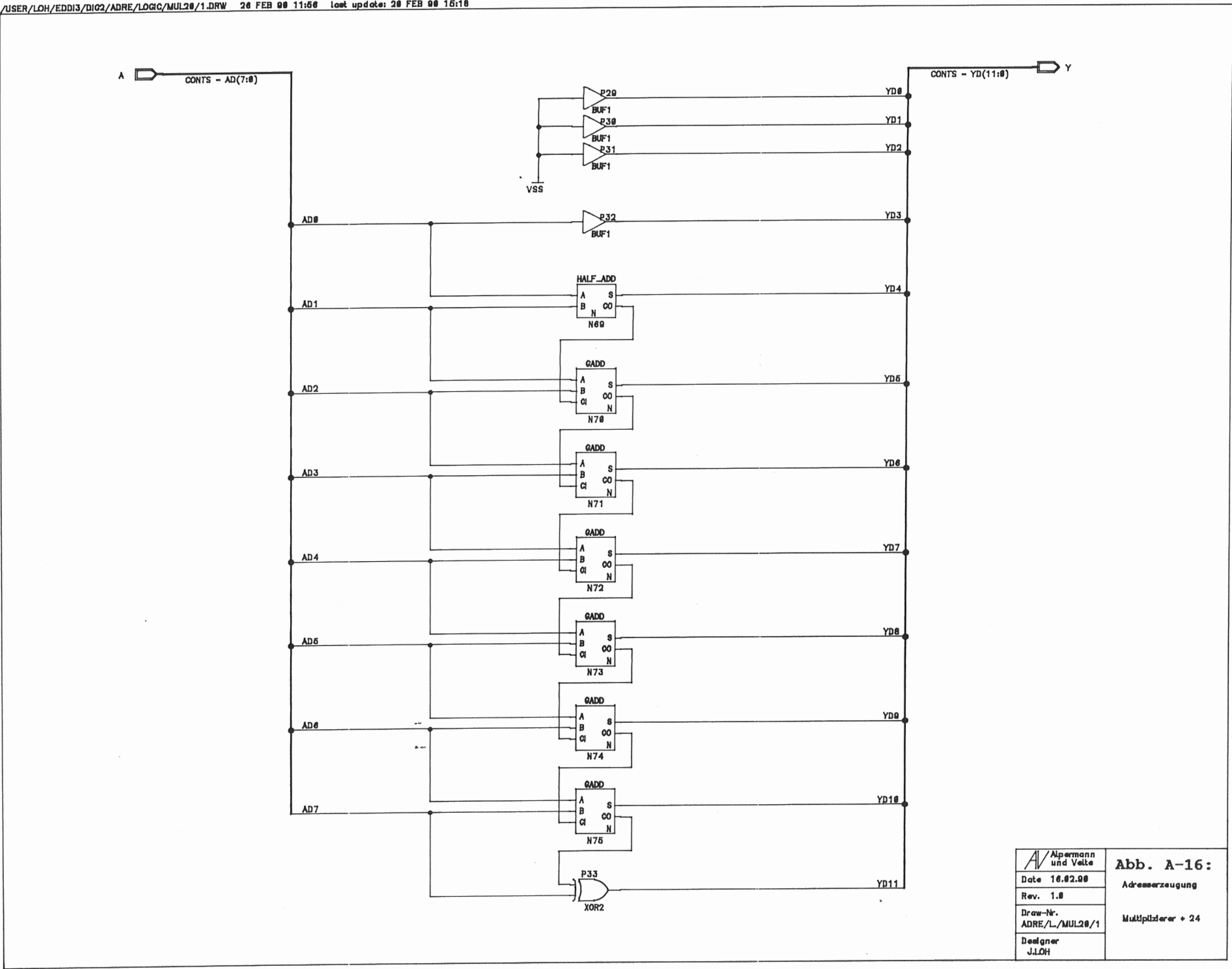
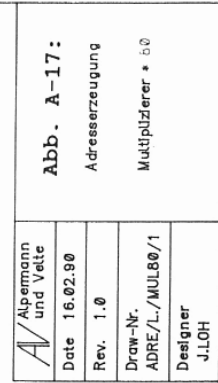
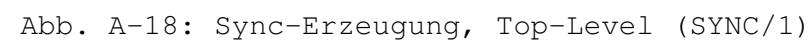
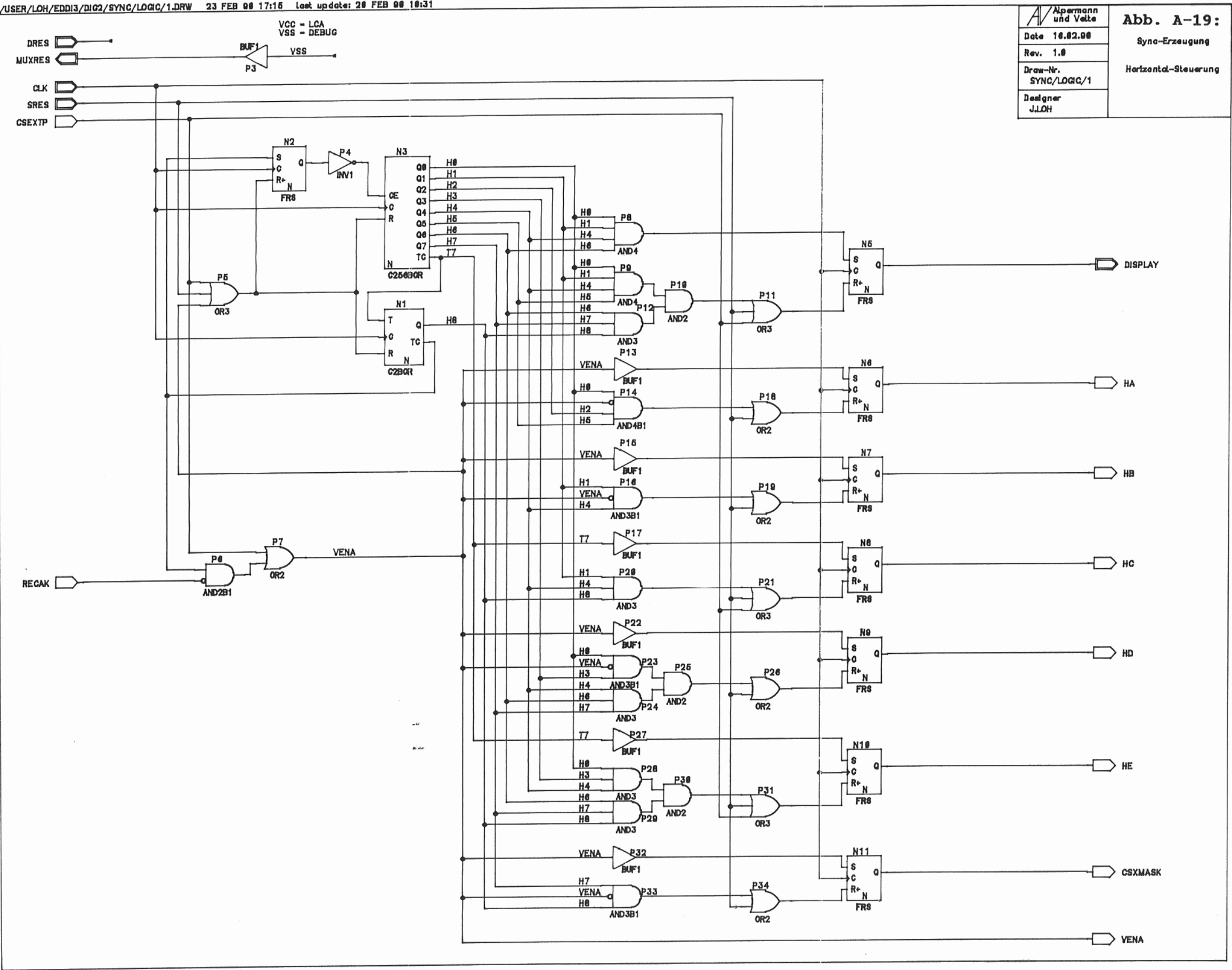


Abb. A-16: Multiplizierer * 24 (ADRE/LOGIC/MUL20/1)



A-17





Alpermann und Volke	Abb. A-19:
Date 16.02.90	Sync-Erzeugung
Rev. 1.0	Horizontal-Steuerung
Draw-Nr. SYNC/LOGIC/1	
Designer J.LOH	

Abb. A-19: Horizontalsteuerung (SYNC/LOGIC/1)

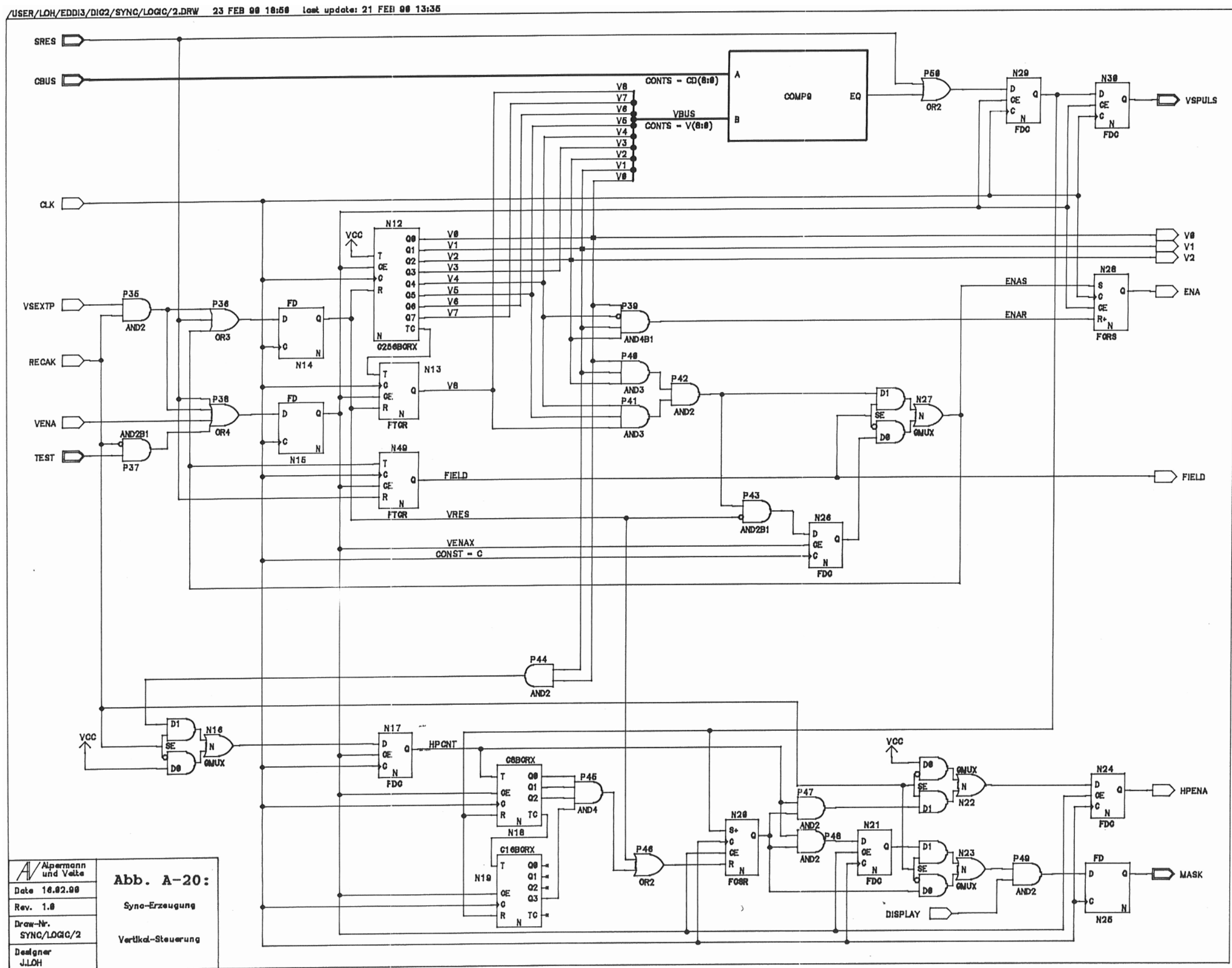


Abb. A-20: Vertikalsteuerung (SYNC/LOGIC/2)

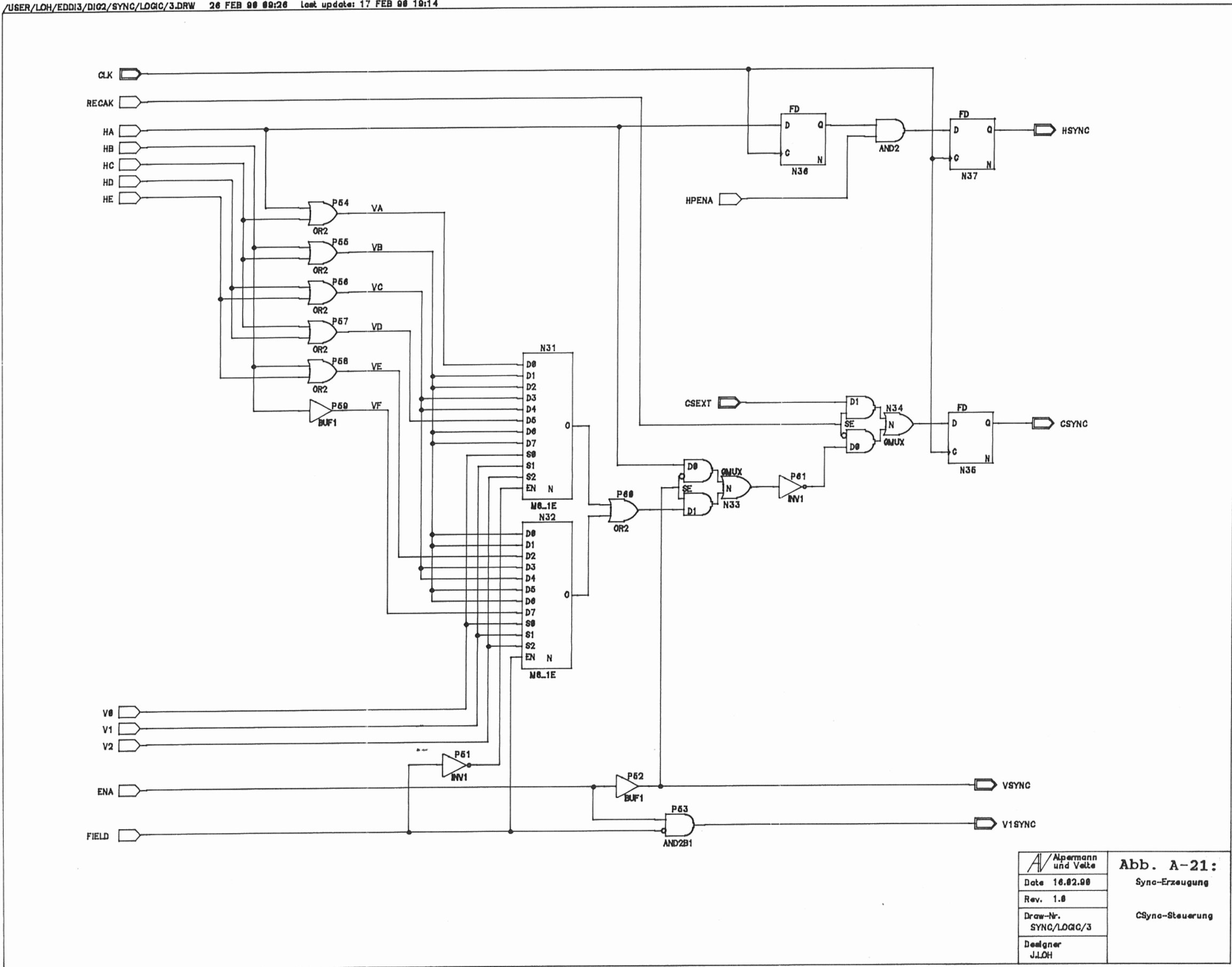


Abb. A-21: Composite-Sync-Erzeugung (SYNC/LOGIC/3)

/USER/LOH/EDDI3/DIG2/SYNC/LOGIC/4.DRW 28 FEB 90 00:45 Last update: 20 FEB 90 10:32

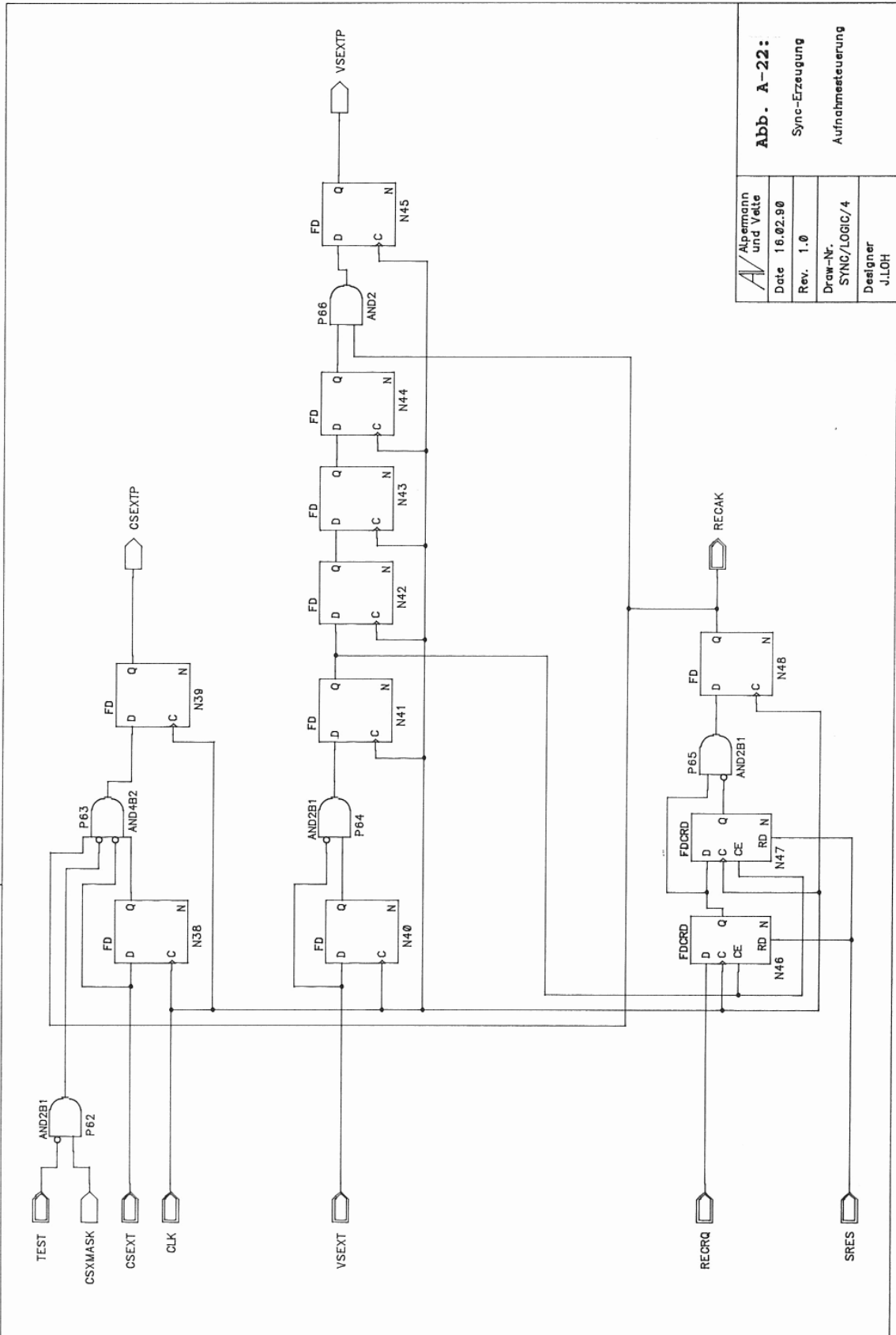
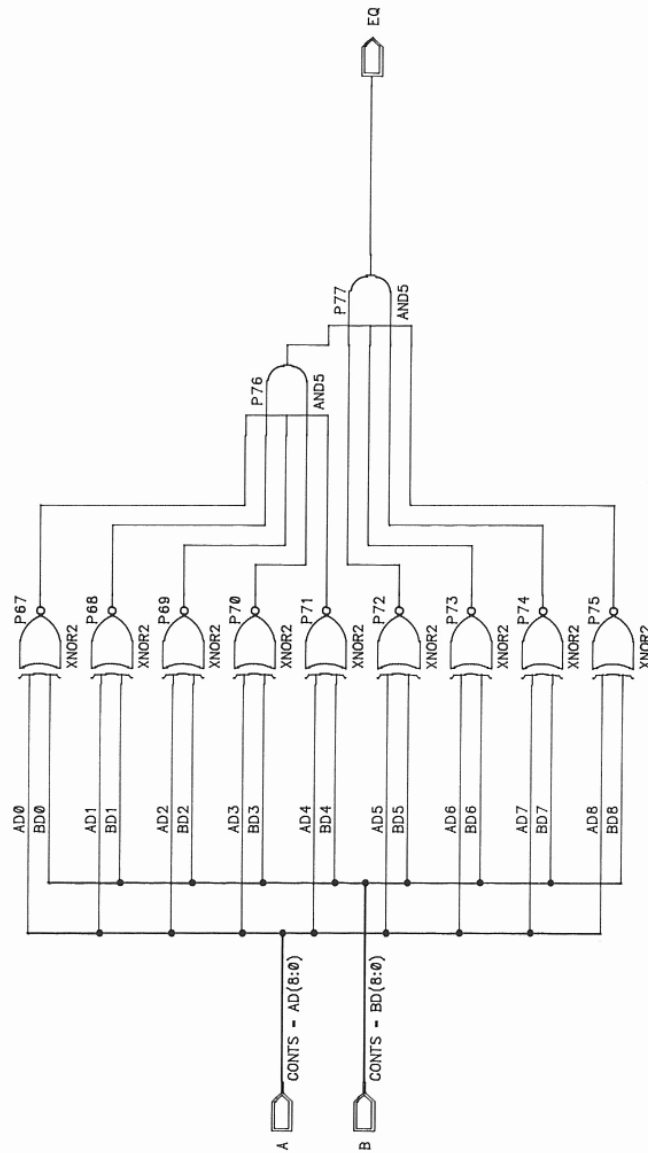


Abb. A-22: Sync-Erzeugung Aufnahmesteuerung	
Altmann und Vete	Date 16.02.90
Rev. 1.0	Draw-Nr. SYNC/LOGIC/4
Designer J.LOH	

Abb. A-22: Aufnahmesteuerung (SYNC/LOGIC/4)

/USER/LOH/EDD13/DIG2/SYNC/LOGIC/COMP9/1.DRW 26 FEB 90 10:00 Last update: 17 FEB 90 19:15



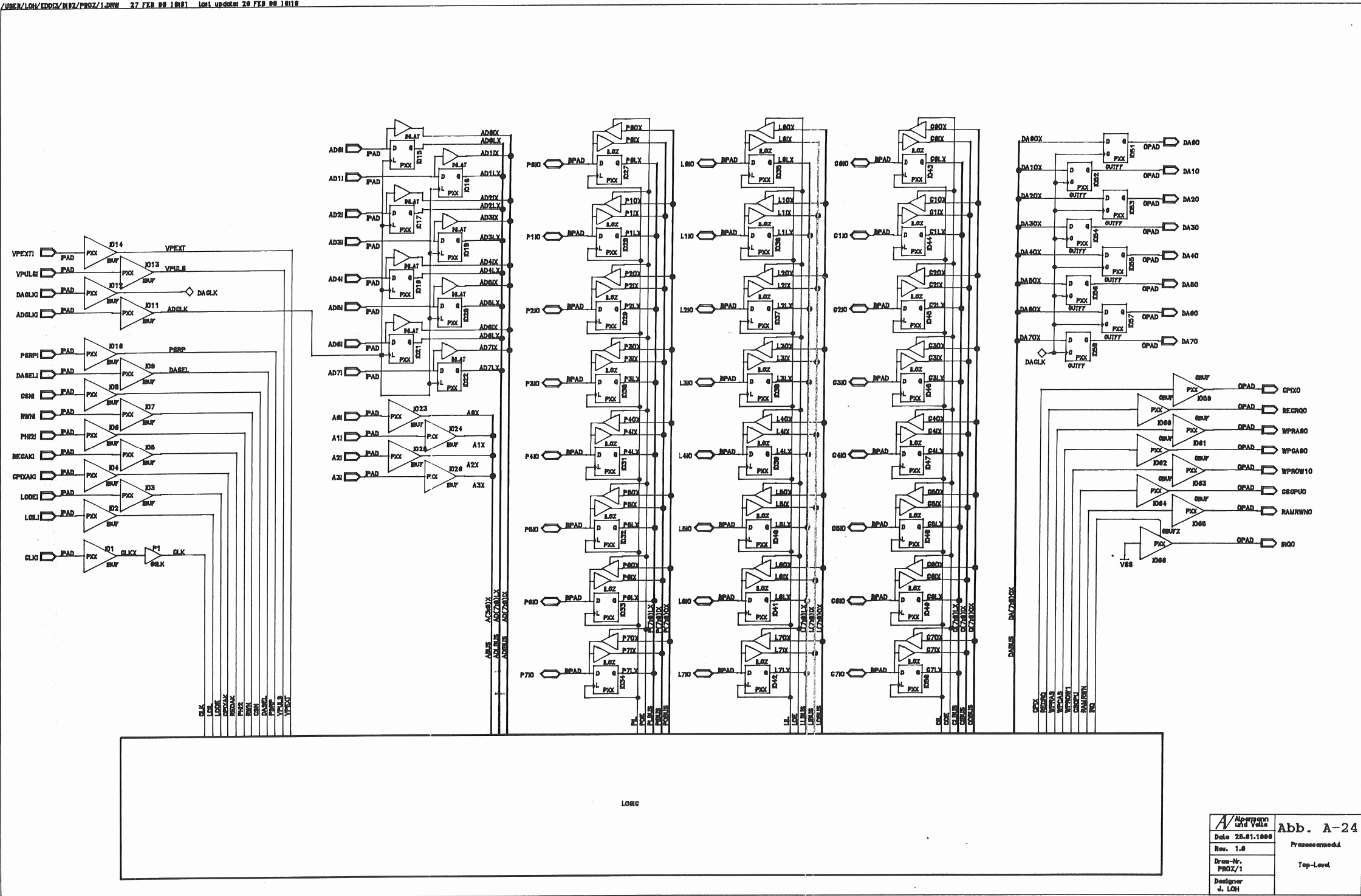
A/Alpemann und Vete	
Date	16.02.90
Rev.	1.0
Draw-Nr.	SYNC/L-/COMP9/1
Designer	J.LOH

Abb. A-23:

Sync-Erzeugung
Zeilenkomparator

Abb. A-23: Zeilenkomparator (SYNC/LOGIC/COMP9/1)

A.5. Prozessormodul, Datensteuerung (PROZ)



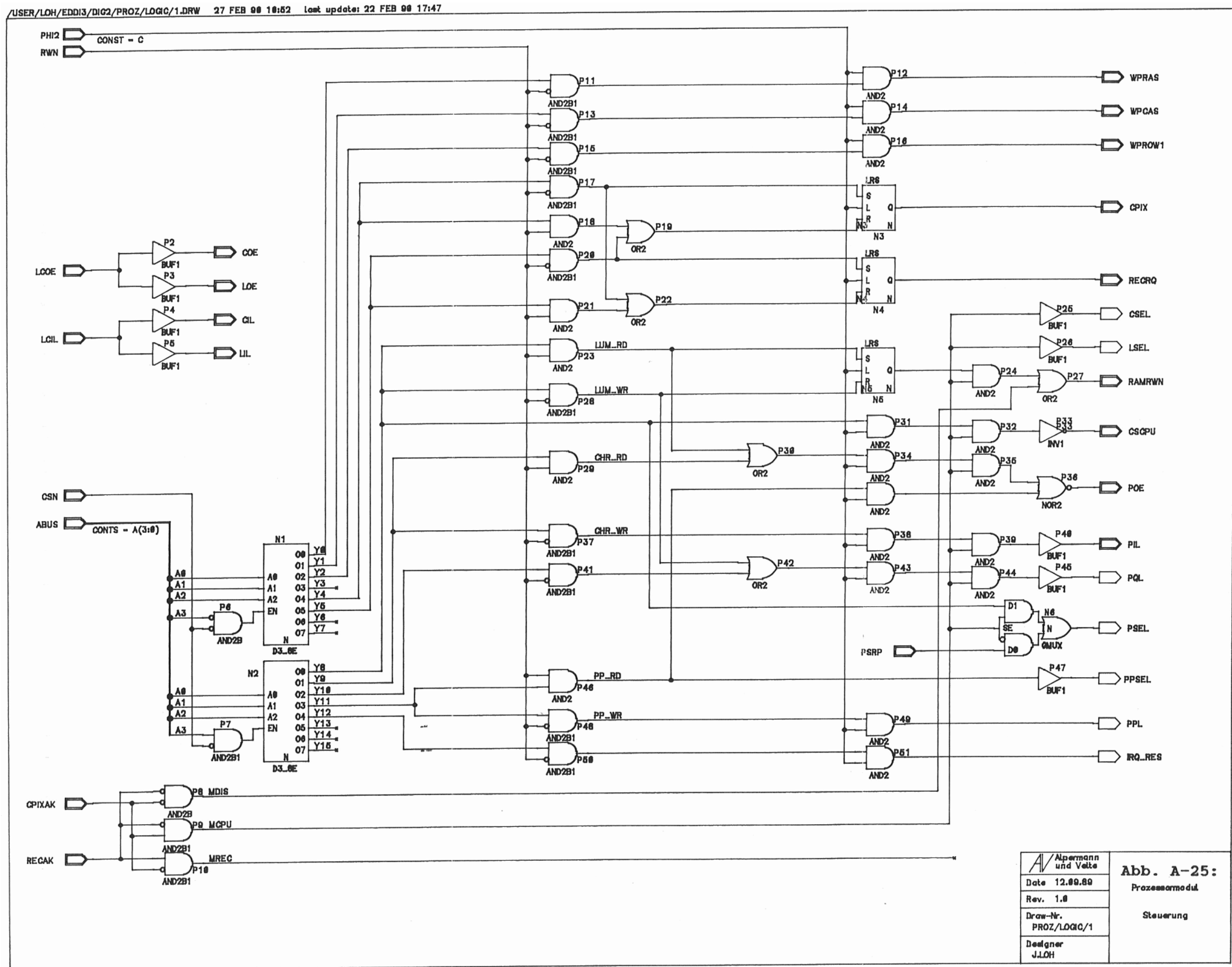


Abb. A-25: Datenwegesteuerung (PROZ/LOGIC/1)

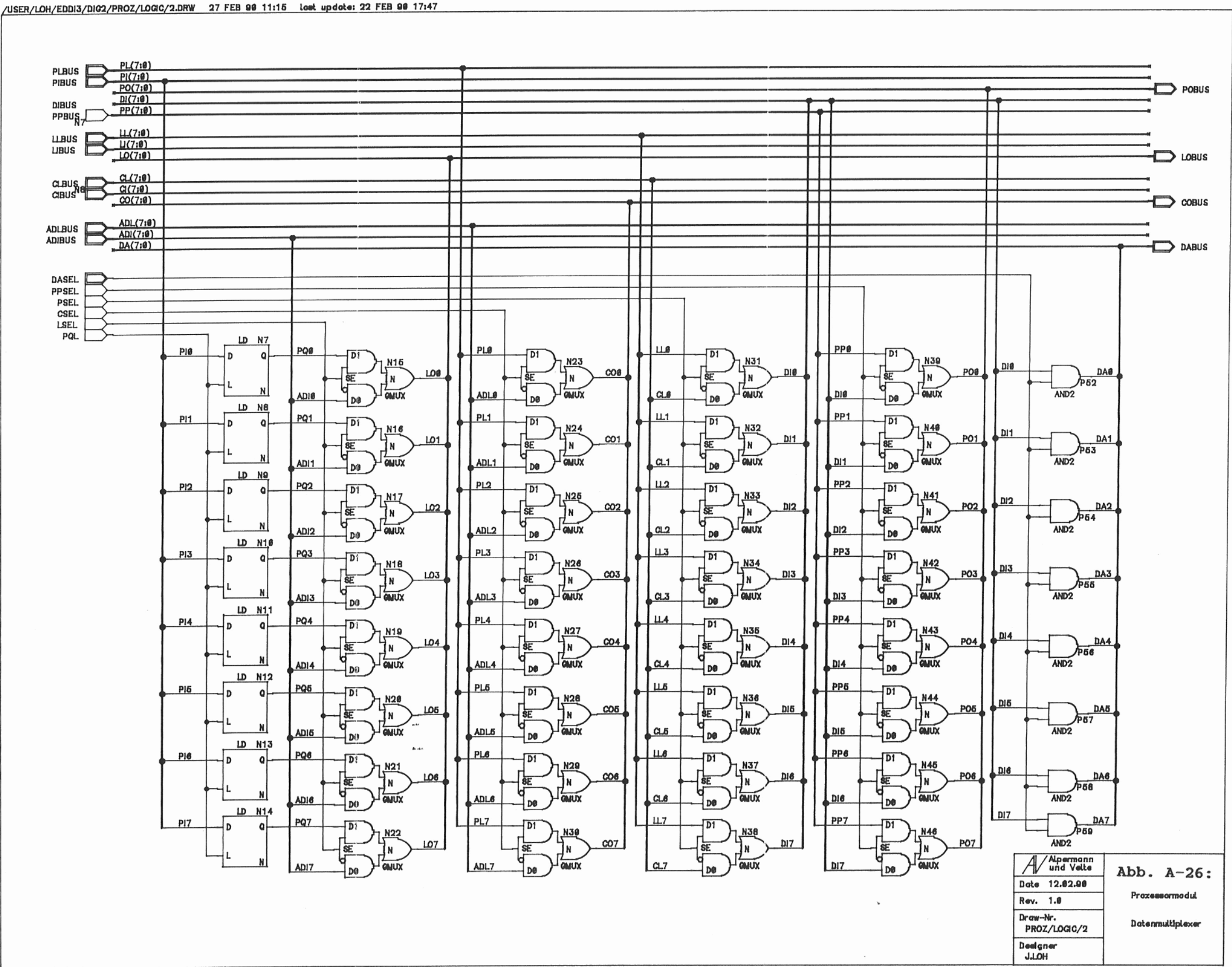
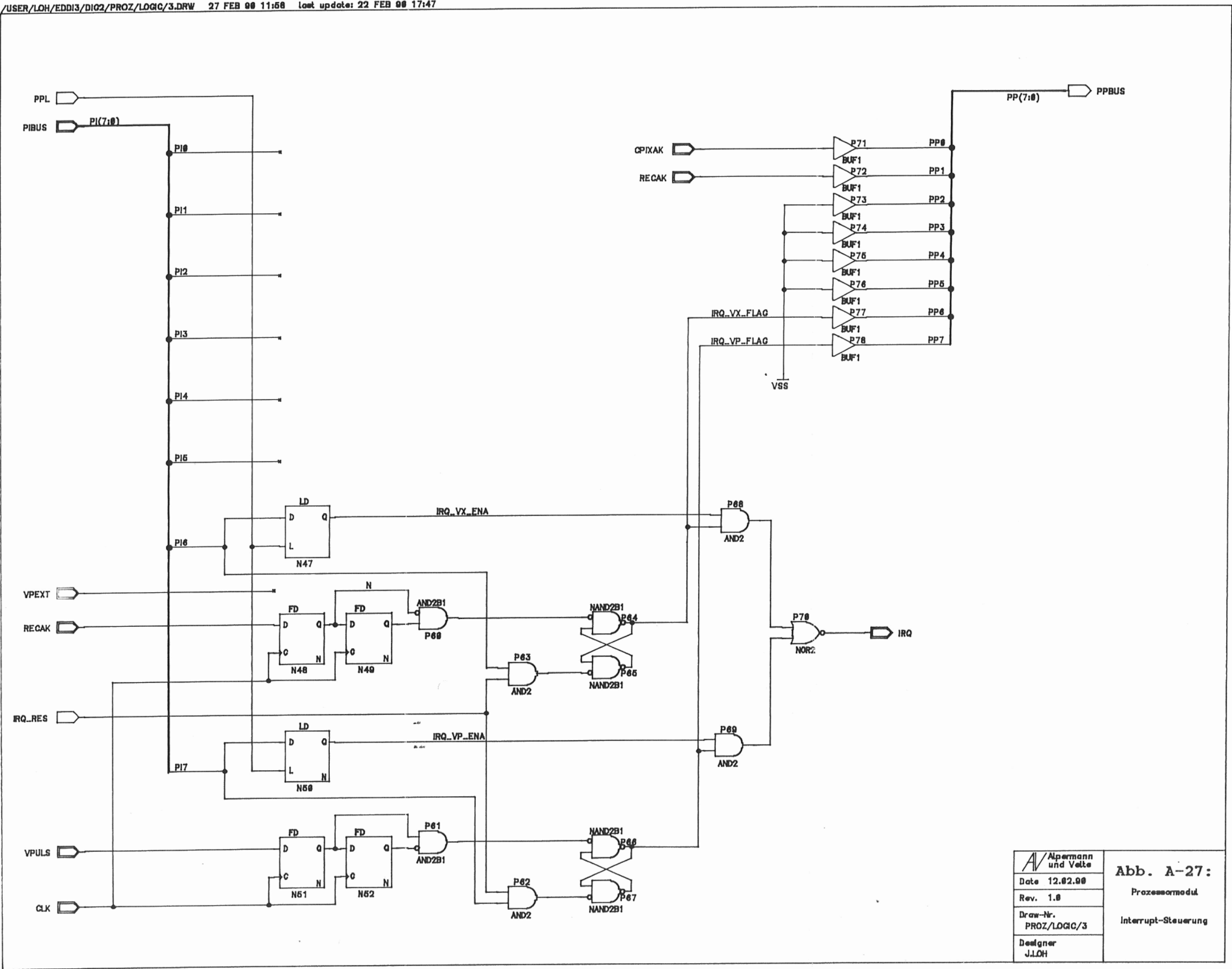


Abb. A-26: Datenmultiplexer (PROZ/LOGIC/2)



A.6. Sonstige Schaltpläne

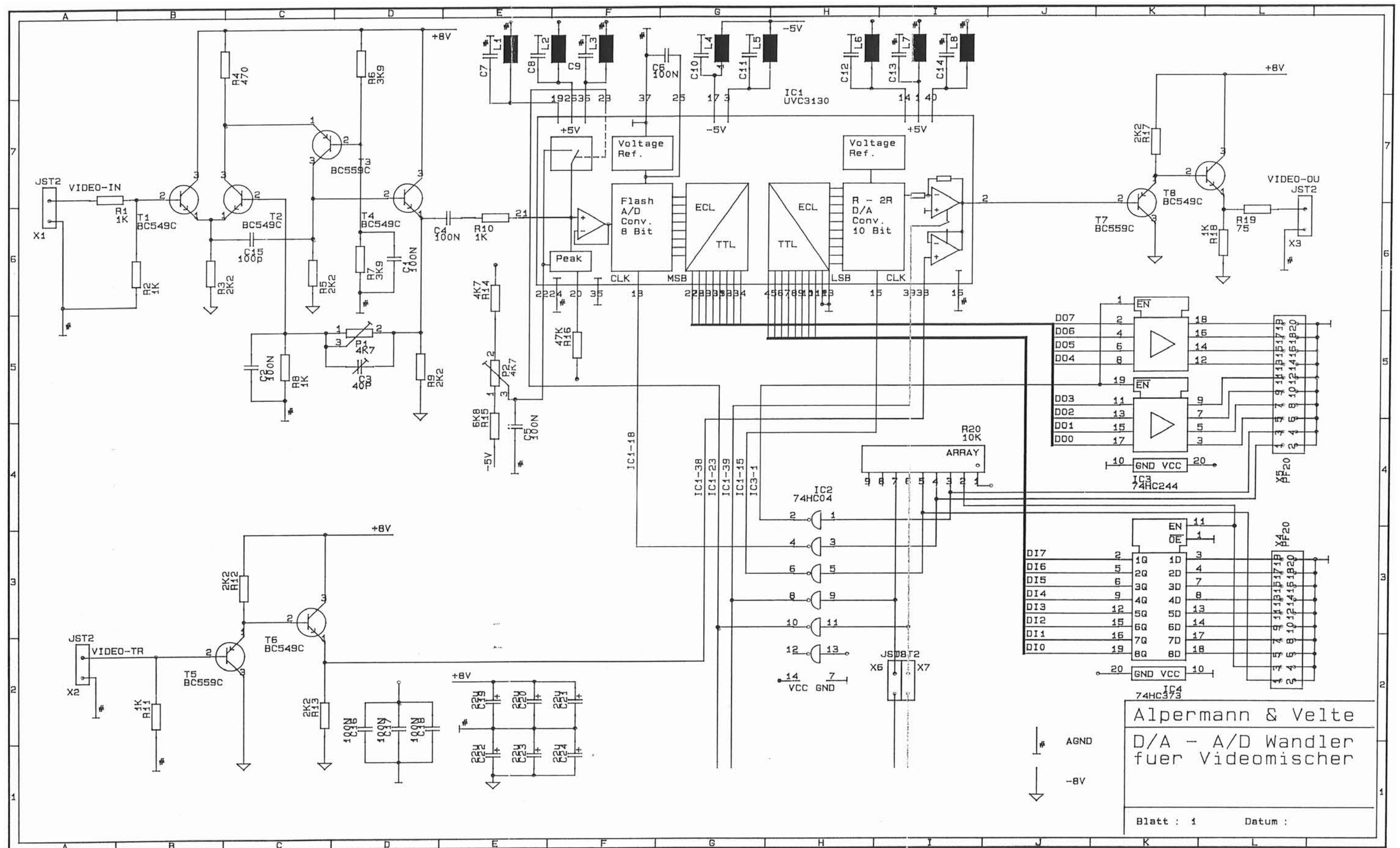


Abb. A-28: A/D- und D/A-Wandlerplatine (ADC/DAC)

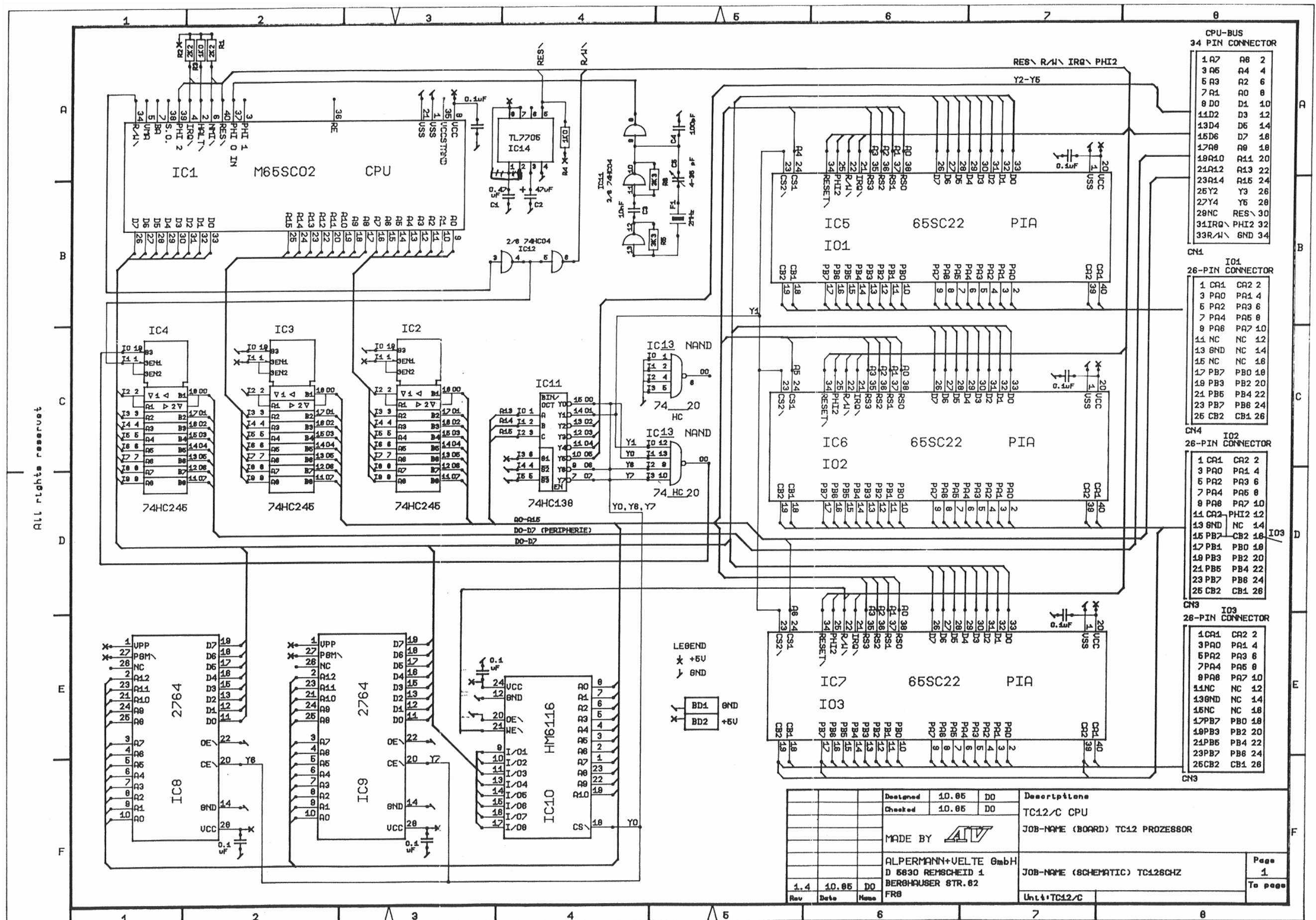


Abb. A-29: CPU-Platine

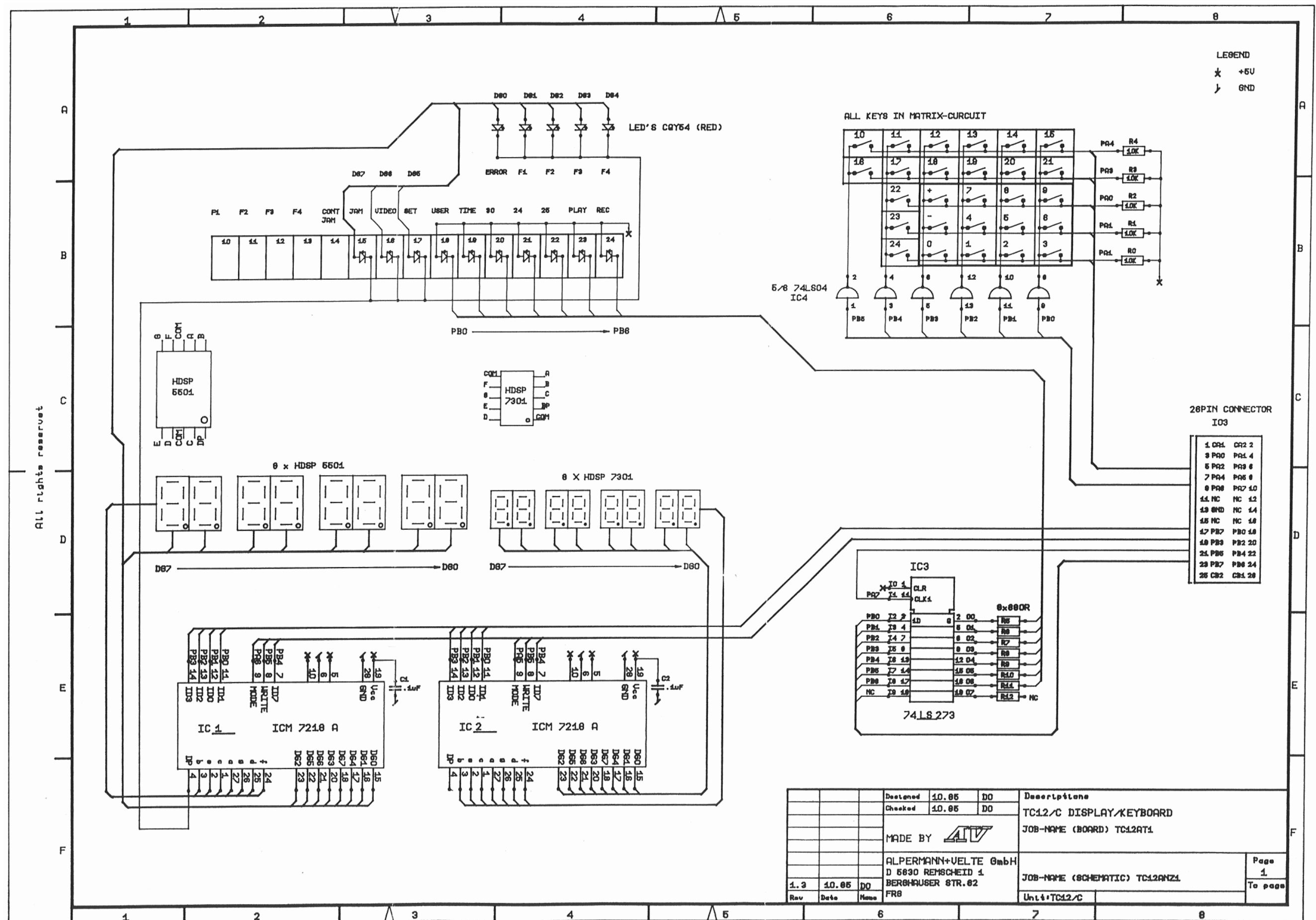


Abb. A-30: Anzeige- und Tastaturplatine

B. Zeitdiagramme

Die Zeitdiagramme sind Simulationsergebnisse der Daisy-Workstation. Sie geben unter Umständen nicht die realen Verzögerungszeiten der LCAs wieder. Die Zeitachse ist in 1ns-Schritte eingeteilt.

Die Diagramme werden im Kapitel 4 zusammen mit der zugehörigen Schaltung erläutert.

B.1. CLK2 - Zeitdiagramme

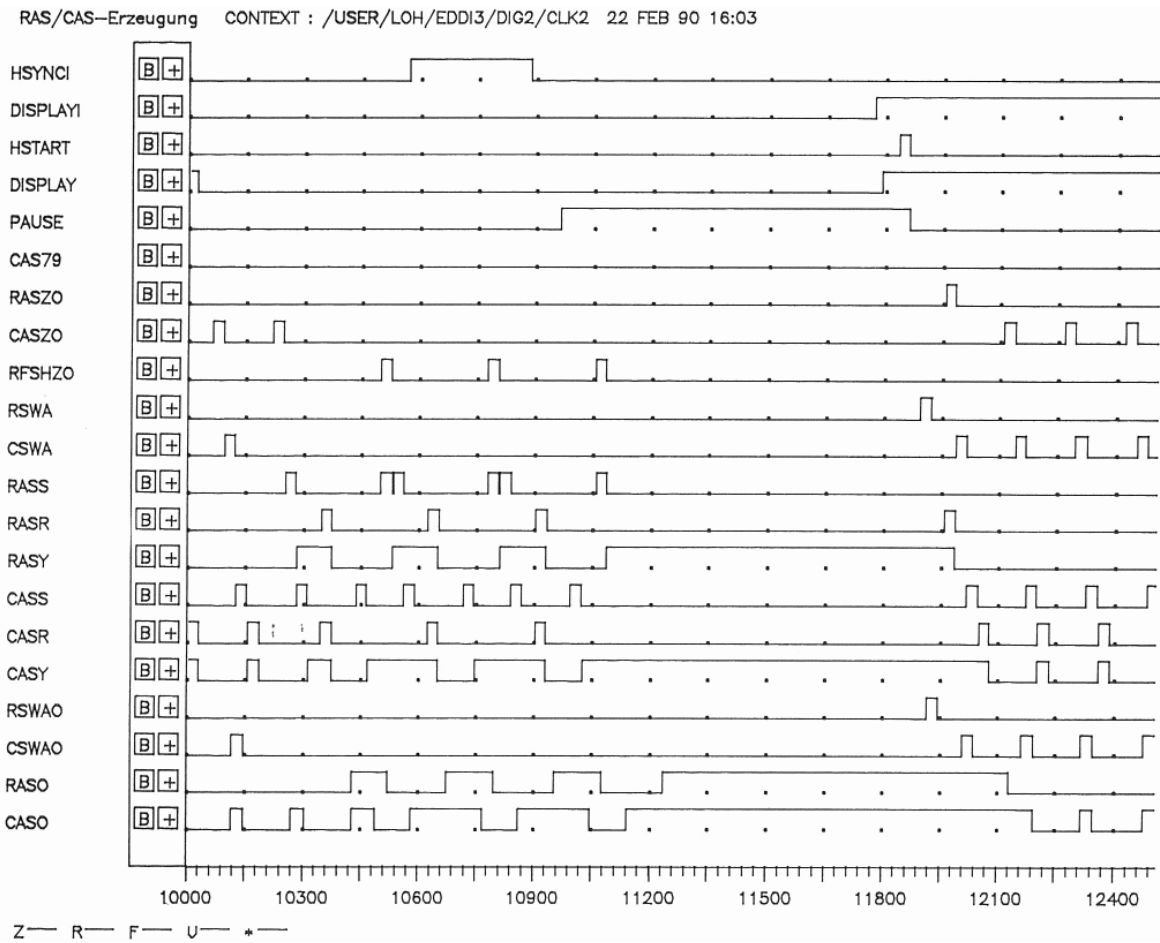


Abb. B-1: CLK2 - RAS/CAS-Erzeugung

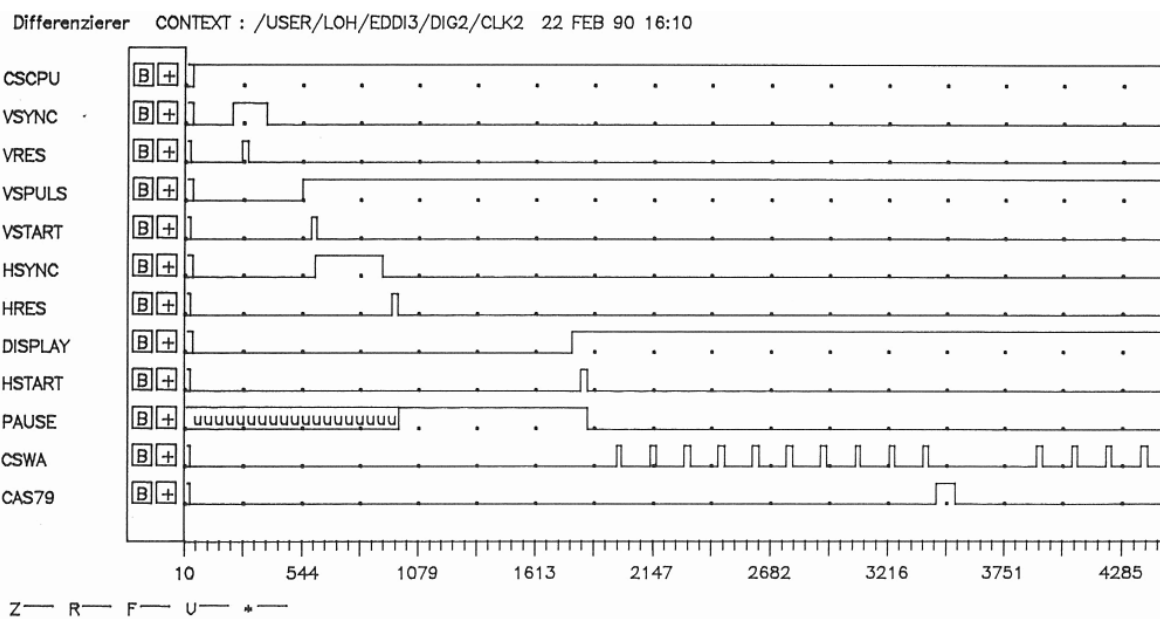


Abb. B-2: CLK2 - Differenzierer

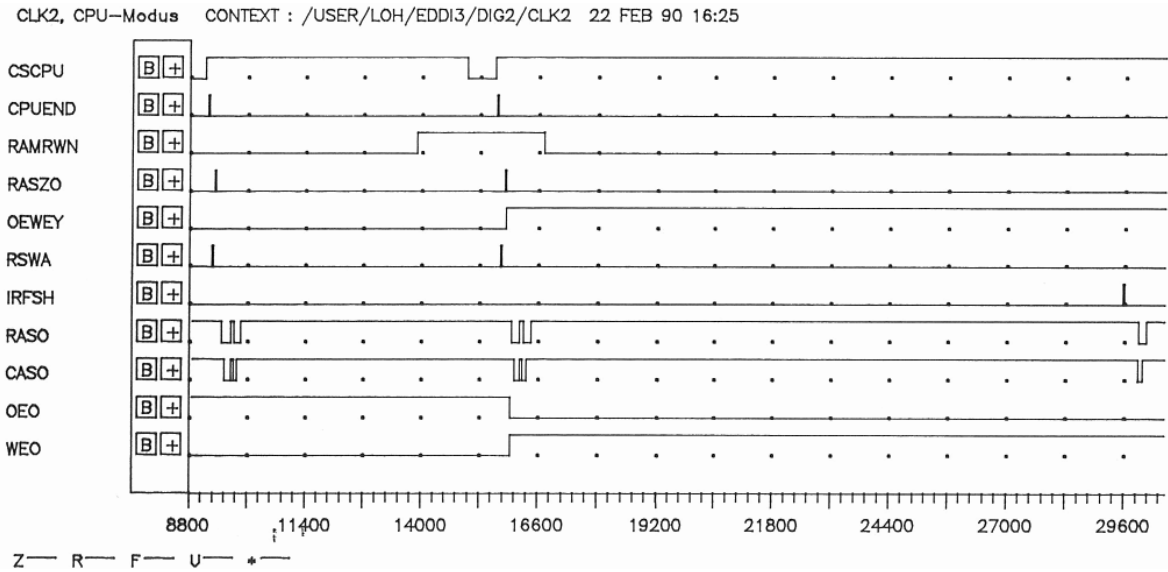


Abb. B-3: CLK2 - CPU-Modus (1)

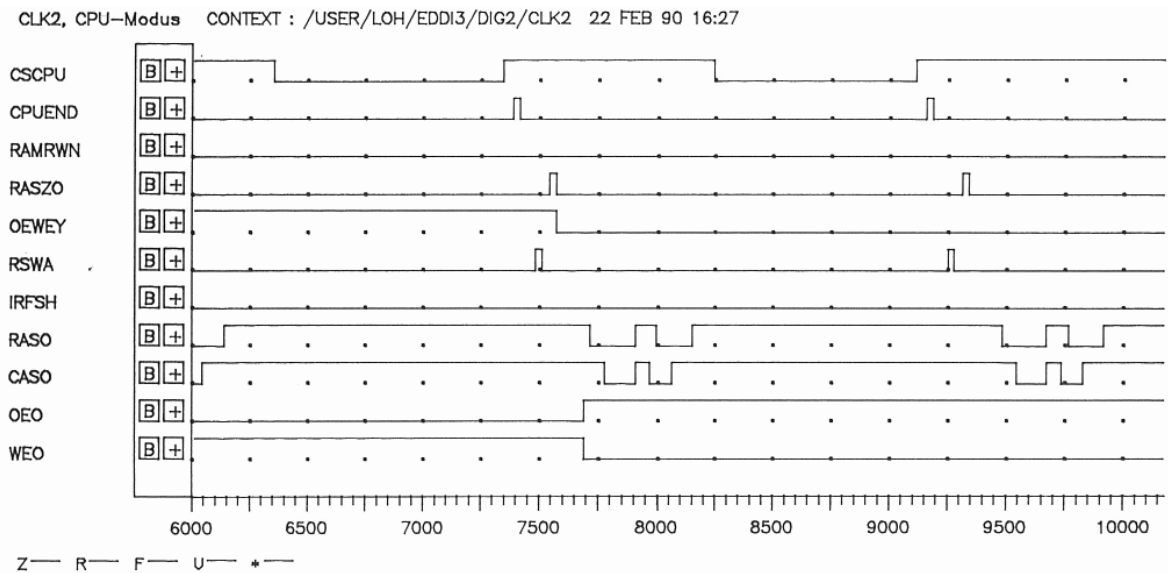


Abb. B-4: CLK2 - CPU-Modus (2)

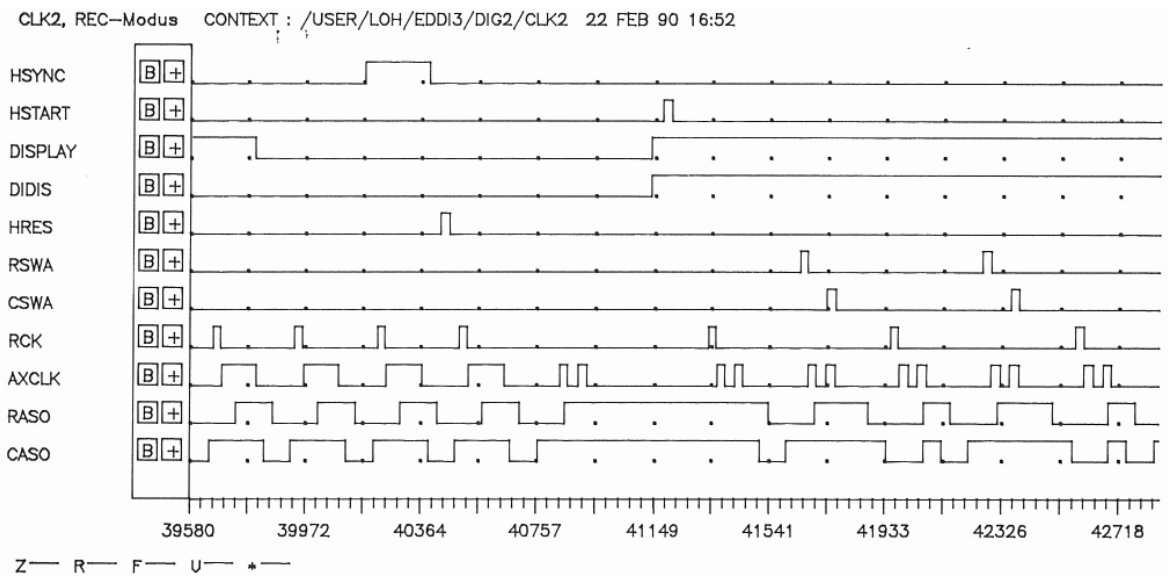


Abb. B-5: CLK2 - REC-Modus

B.2. ADRE - Zeitdiagramme

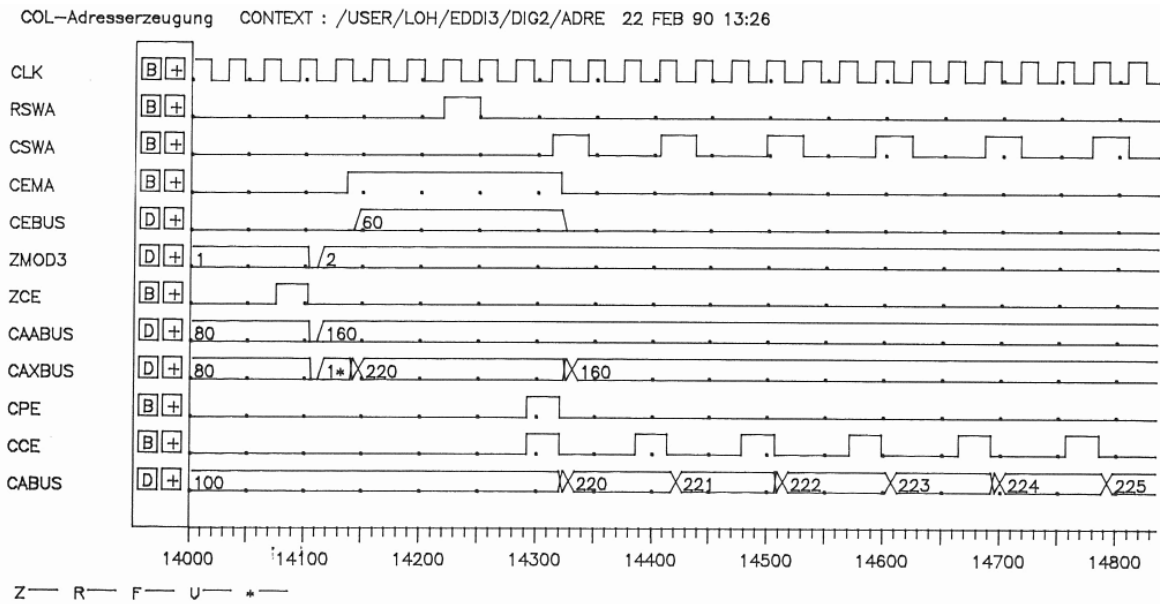


Abb. B-6: ADRE - COL-Adreßerzeugung (1)

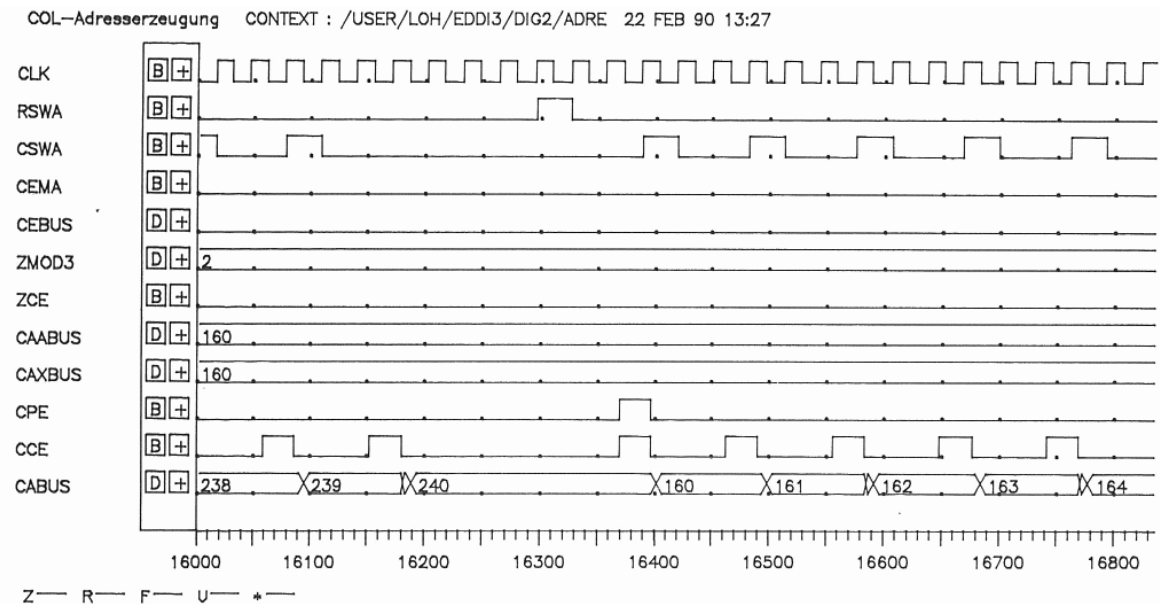


Abb. B-7: ADRE - COL-Adreßerzeugung (2)

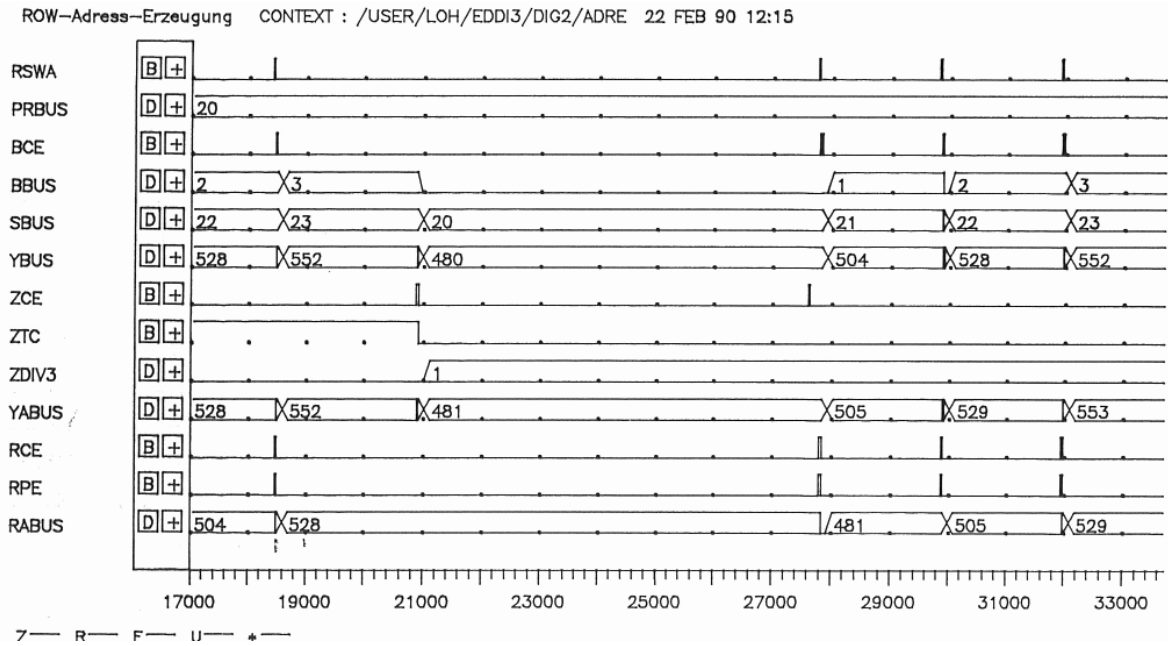


Abb. B-8: ADRE - ROW-Adreßerzeugung

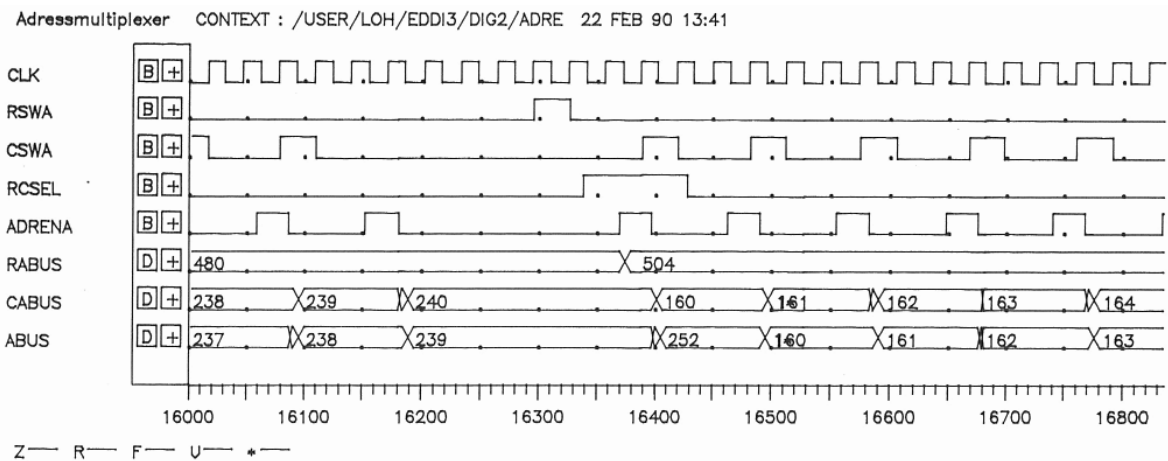


Abb. B-9: ADRE - Adreßmultiplexer

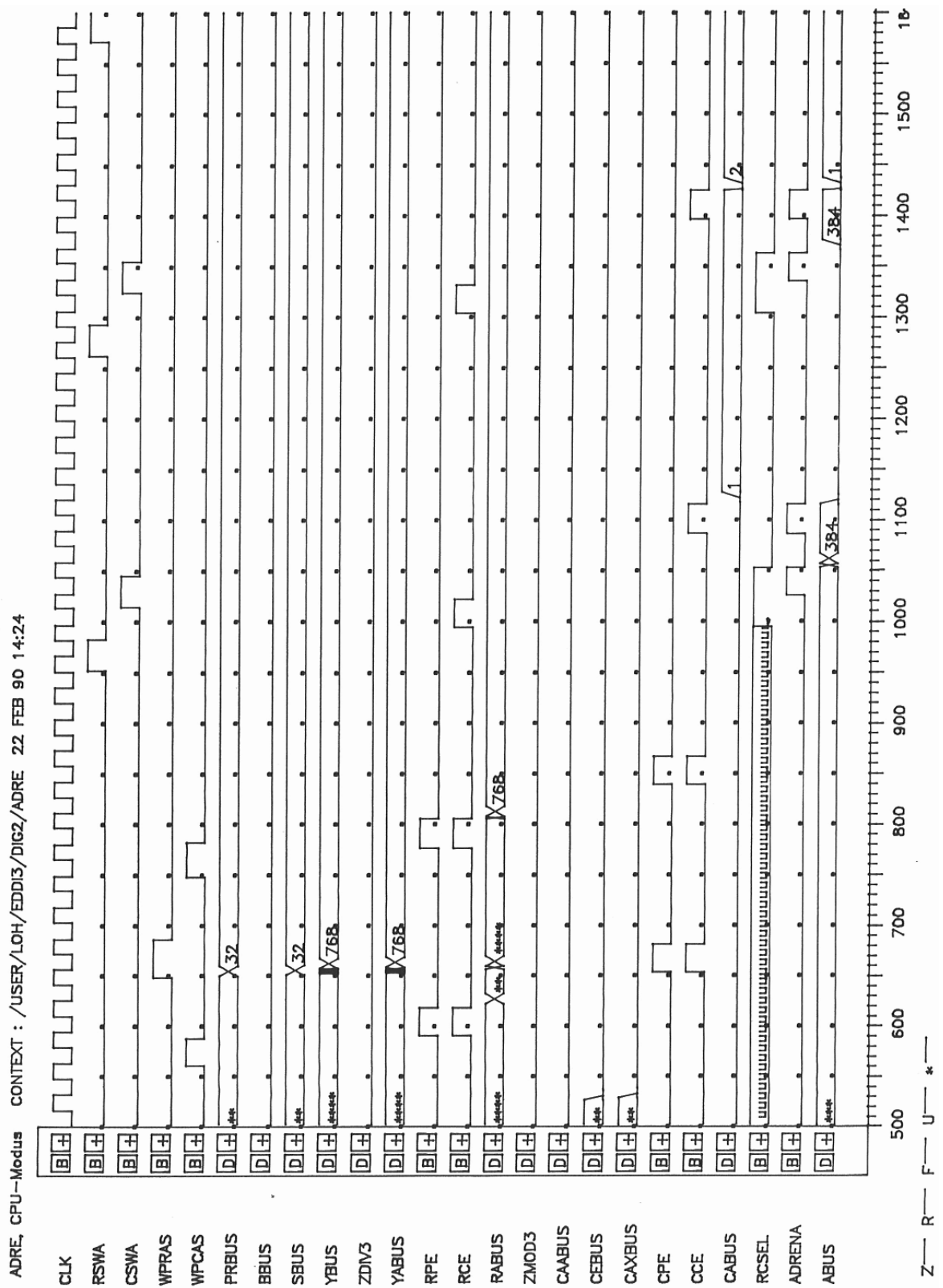


Abb. B-10: ADRE - CPU-Modus

B.3. SYNC - Zeitdiagramme

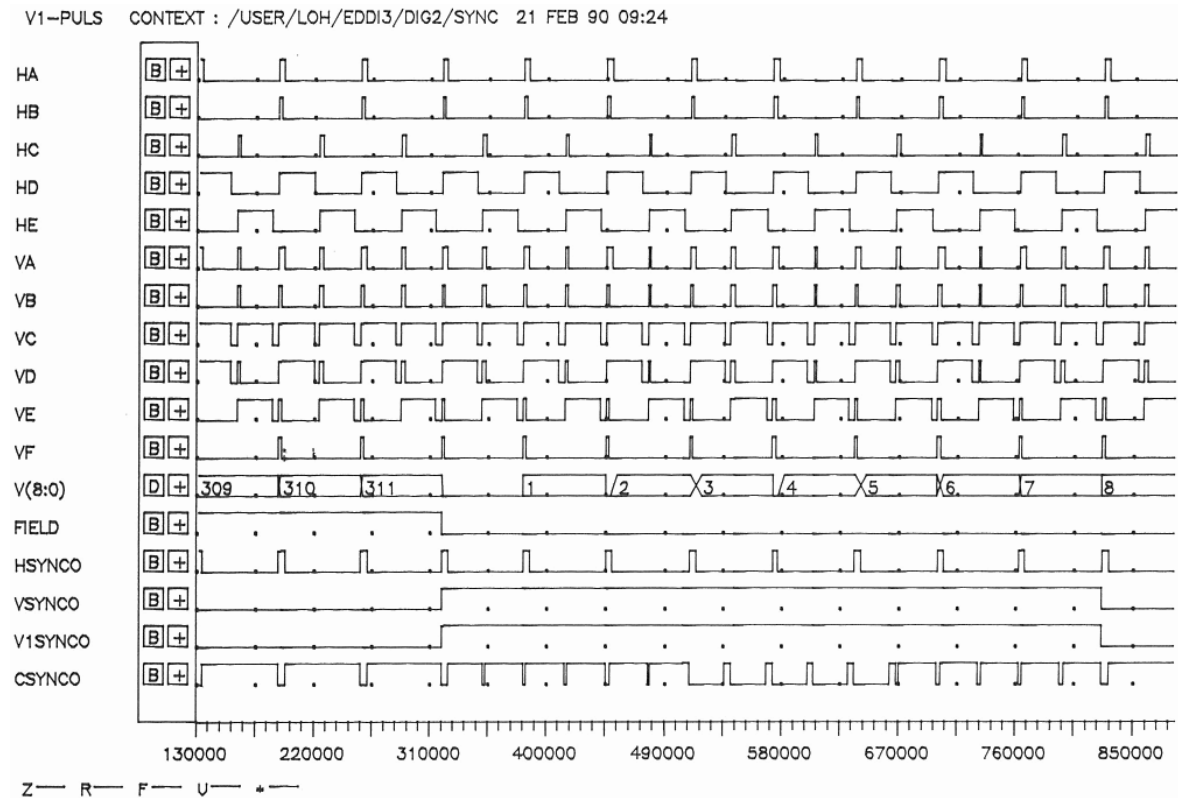


Abb. B-11: SYNC - V1-Puls

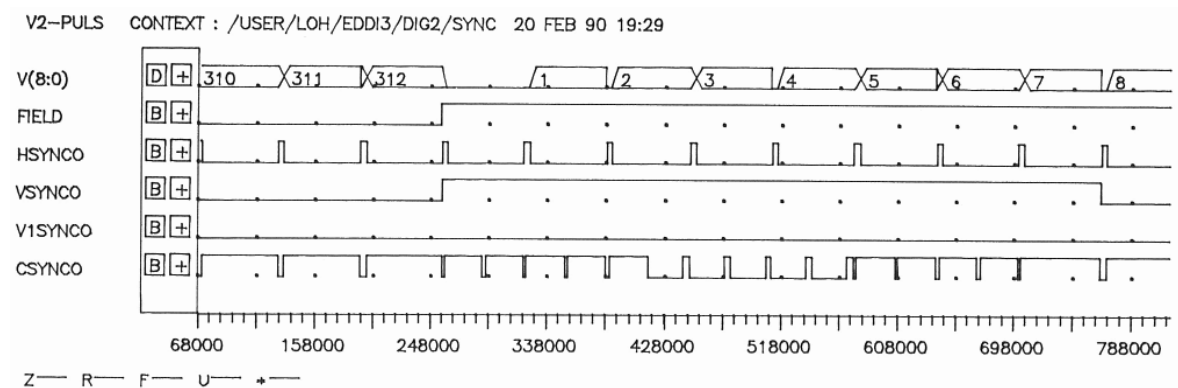


Abb. B-12: SYNC - V2-Puls

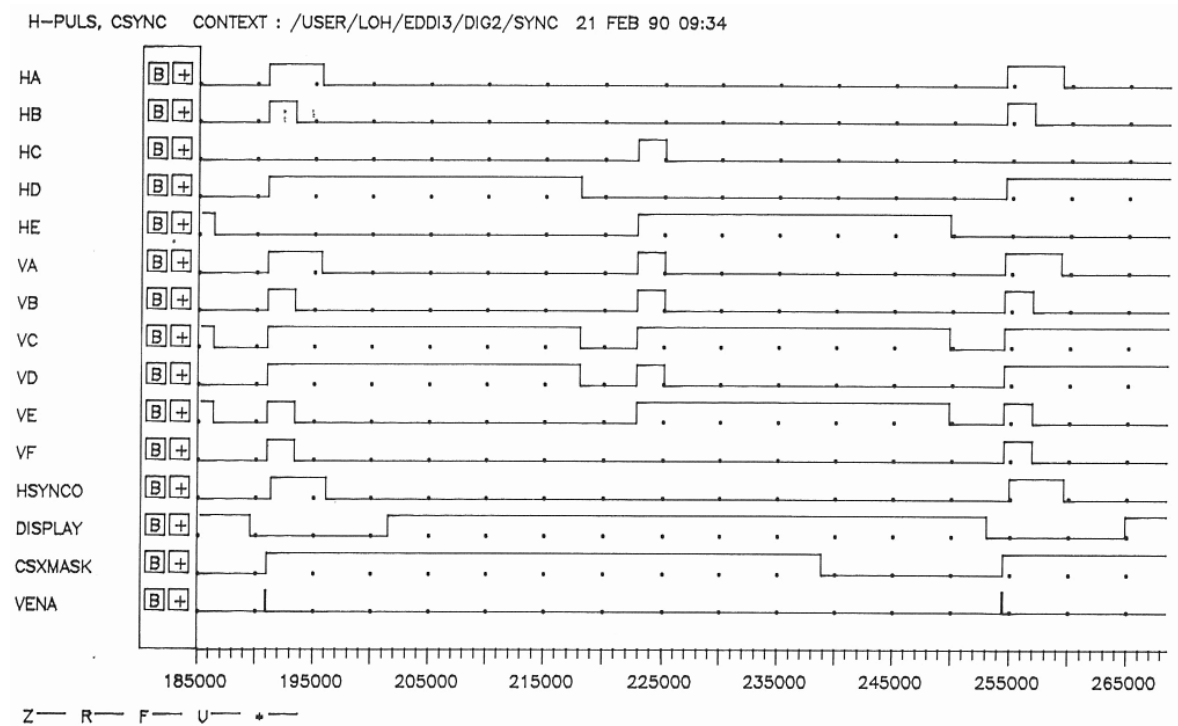


Abb. B-13: SYNC - H-Puls, CSync-Erzeugung

B.4. PROZ - Zeitdiagramme

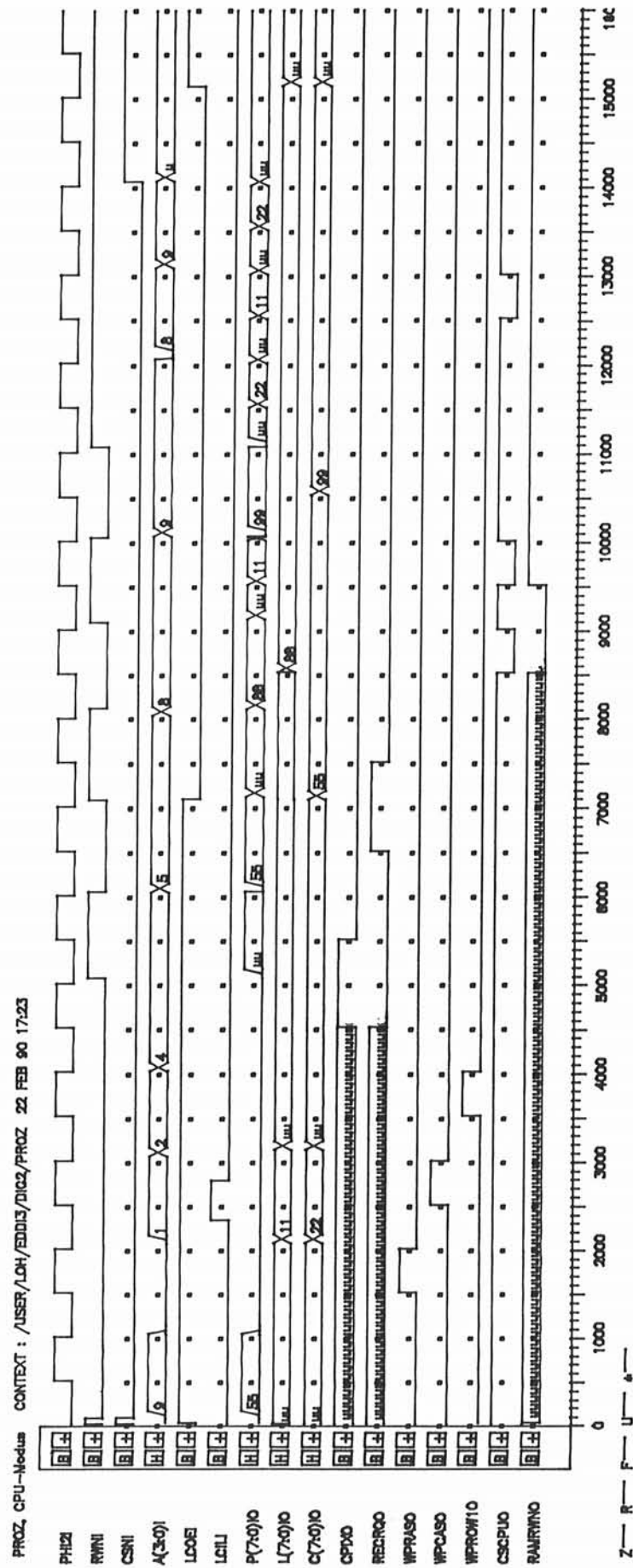


Abb. B-14: PROZ - CPU-Modus

C. LCA-Makrobibliothek

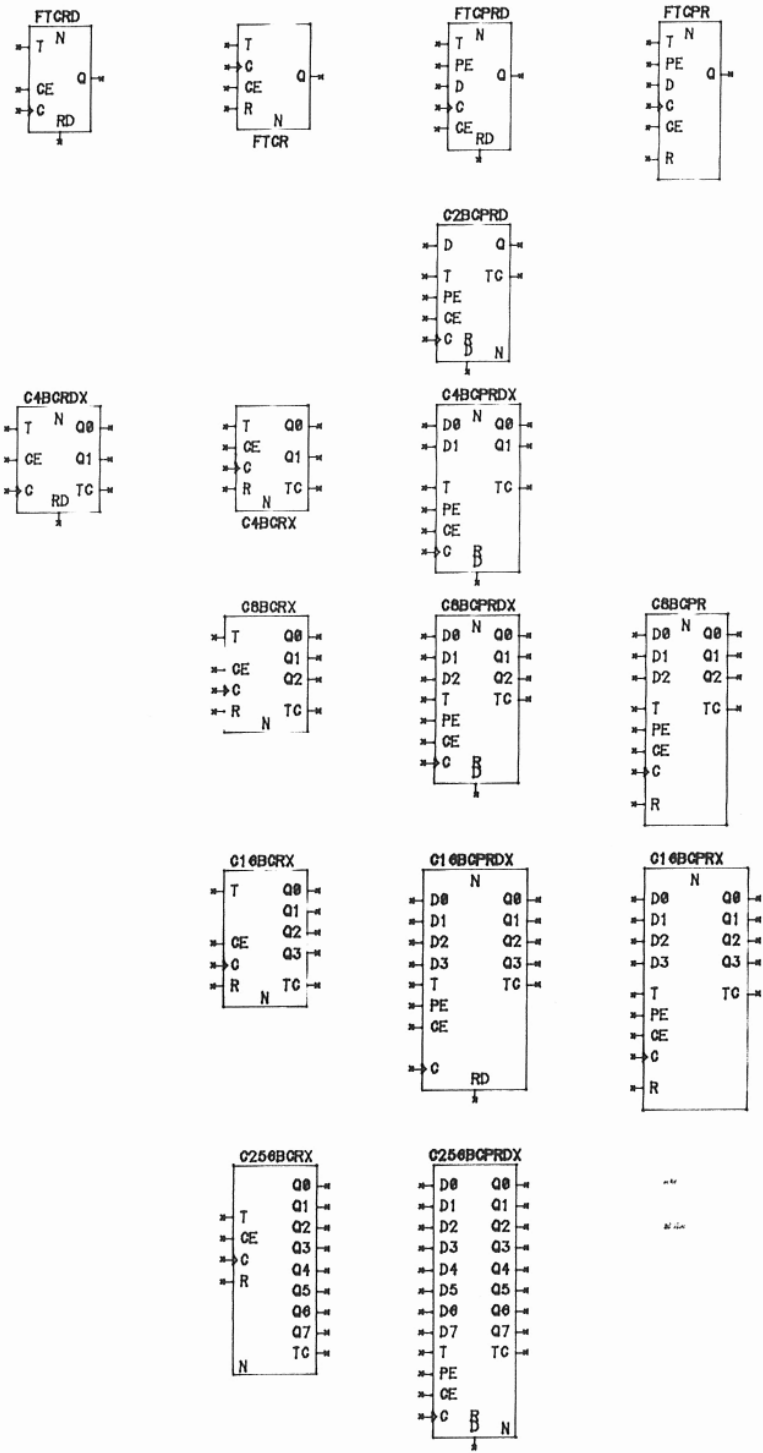
C.1. Übersicht

Mit dem LCA-Entwicklungspaket von XILINX wird eine Makrobibliothek geliefert, in denen die wichtigsten Grundsaltungen wie Dekoder, Multiplexer, Flip-Flops, Register und Zähler enthalten sind. Es fehlen jedoch fast vollständig Makros, die den Clock-Enable-Eingang der CLB-Basis-Flip-Flops der 3000er-LCA-Serie ausnutzen. Daher wurde im Rahmen der Diplomarbeit eine eigene Makrobibliothek angelegt, die in erster Linie Flip-Flops und Zähler mit Clock-Enable-Eingang enthält. Dadurch wird es möglich, die Flip-Flops mit einem synchronen Takt zu versorgen, während sie, kontrolliert durch den Clock-Enable-Eingang, effektiv mit verschiedenen Geschwindigkeiten arbeiten. Dies wird in der vorliegenden Schaltung sehr oft ausgenutzt.

Zusätzlich wurden verschiedene kombinierte Addierer/Subtrahierer erstellt, die über einen Steuereingang umgeschaltet werden können. Wird diese Umschalteigenschaft nicht benötigt und der Steuereingang fest auf VCC oder VSS gelegt, entfallen die dadurch überflüssig gewordenen Gatter in einem späteren Optimierungsvorgang. Sie belasten dadurch nicht die Implementierung der Schaltung ins LCA.

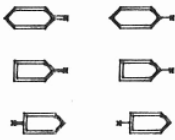
/USER/LCA/LIB3K/AV_LCA3K_01.LIB 23 FEB 90 12:32 last update: 21 FEB 90 10:11

Zaehler

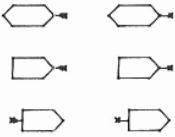


↑ · ⊥

VCC- und VSS-Symbole



Hierachische Konnektoren

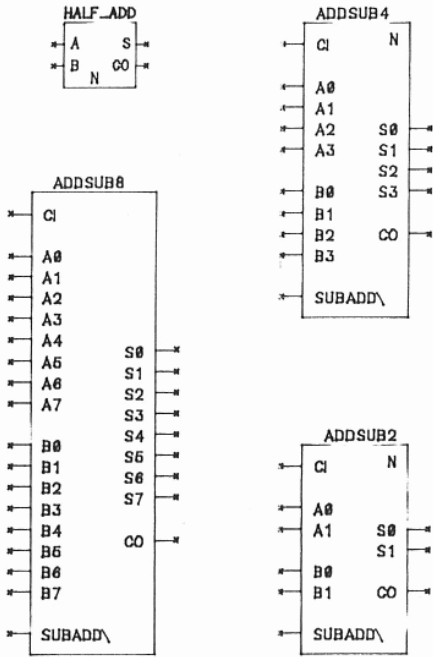


Inter-Page-Konnektoren

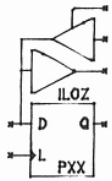


Intra-Page-Konnektoren

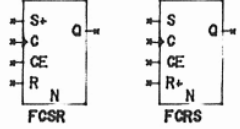
Addierer/Subtrahierer



PAD



RS-Flip-Flops



Alpermann und Vette	Abb. C-1:
Date 90.00.0000	AV_LCA_3K.LIB
Rev. 0.0	
Draw-Nr. 0000-0000-00	LCA-Makros
Designer 000000000000	

Abb. C-1: Übersicht über die LCA-Makrobibliothek

C.2. Flip-Flops

C.2.1. RS-Flip-Flops

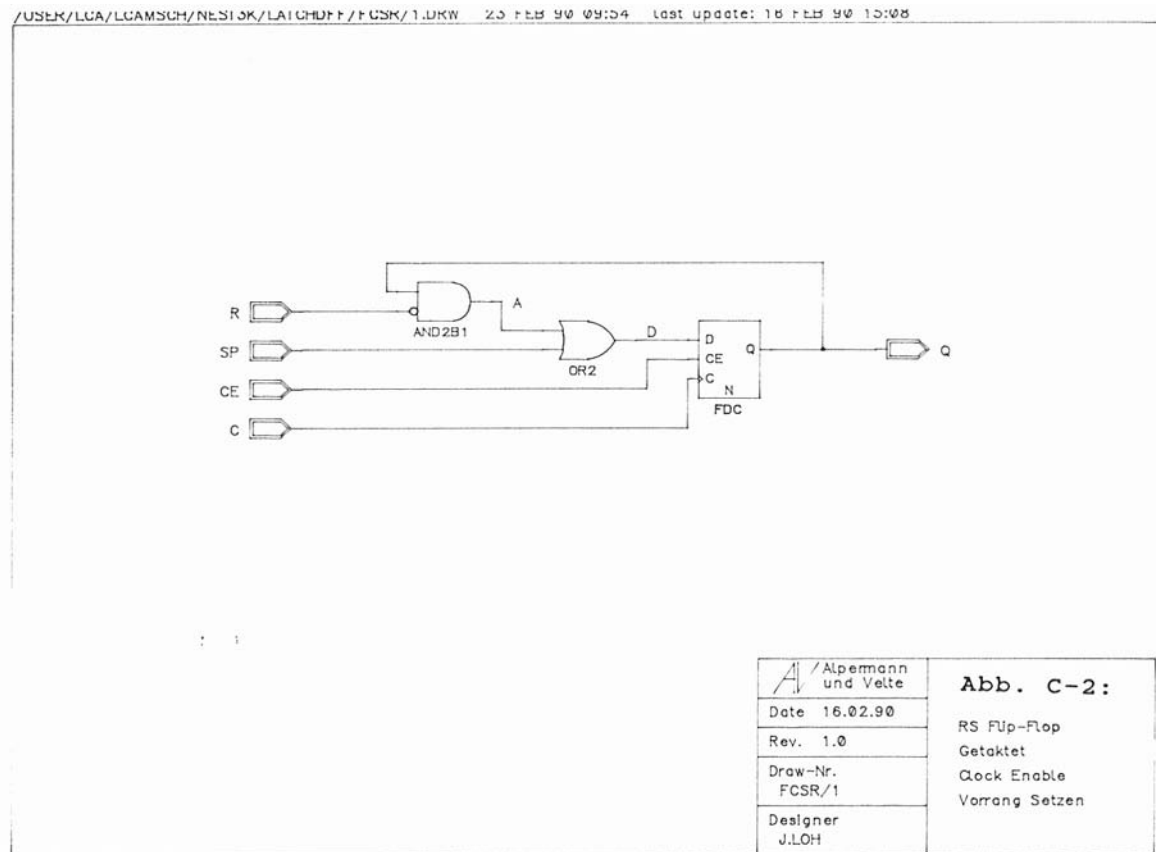


Abb. C-2: FCSR: Getaktetes RS-Flip-Flop, Vorrang Setzen

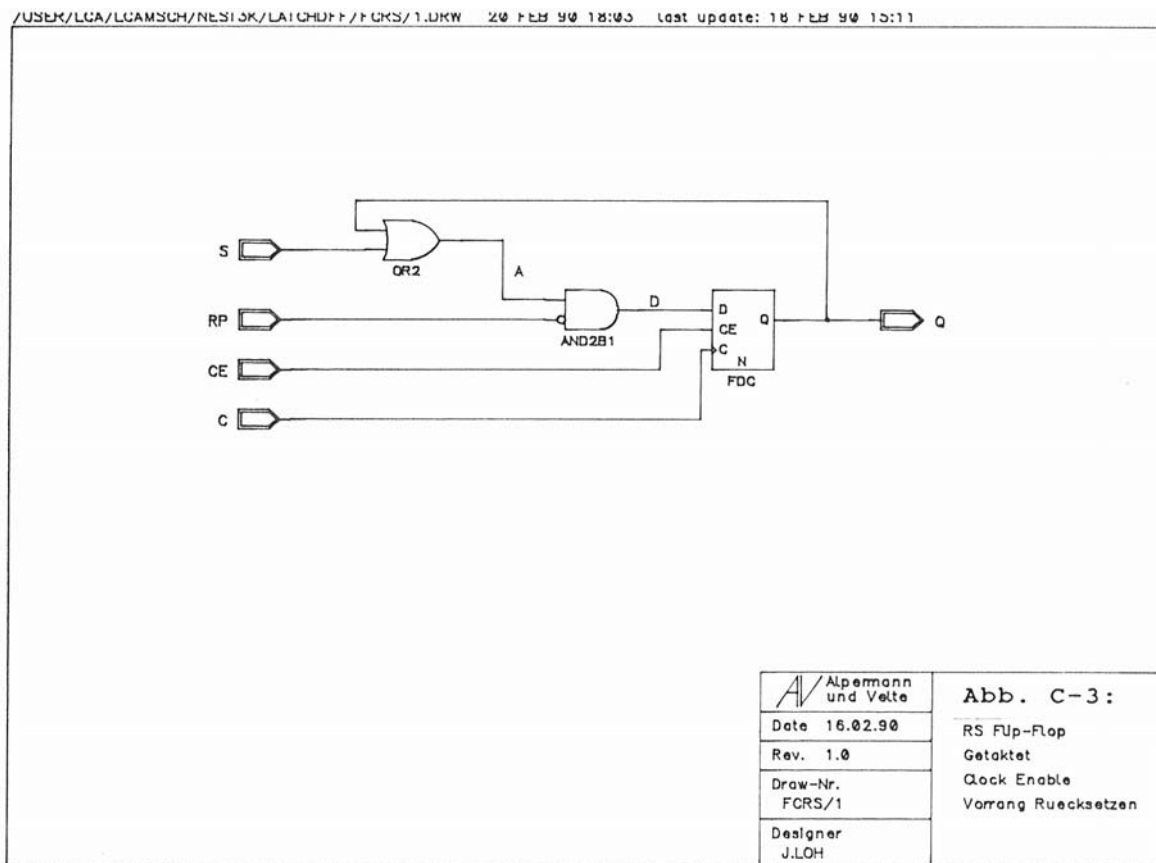


Abb. C-3: FCRS: Getaktetes RS-Flip-Flop, Vorrang Rücksetzen

C.2.2. Toggle-Flip-Flops

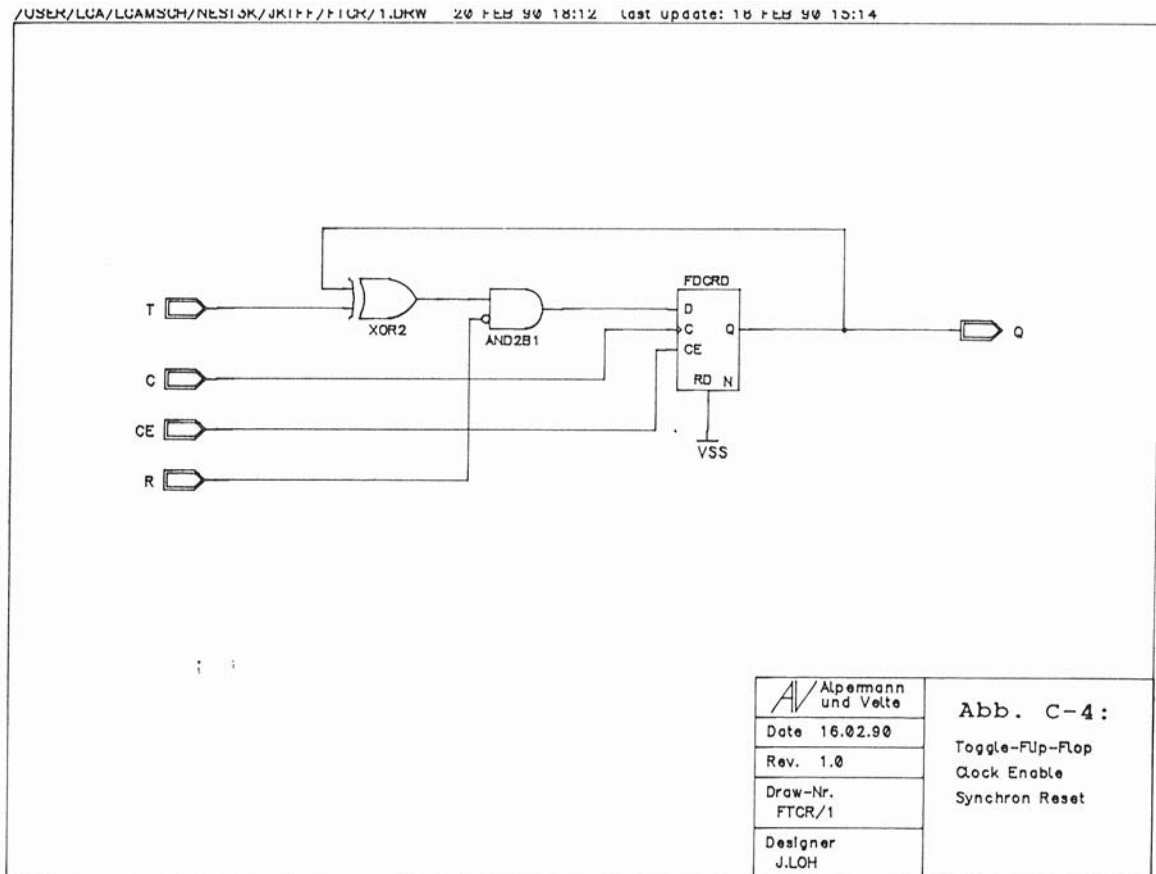


Abb. C-4: FTCR: Toggle-Flip-Flop, Synchron Reset

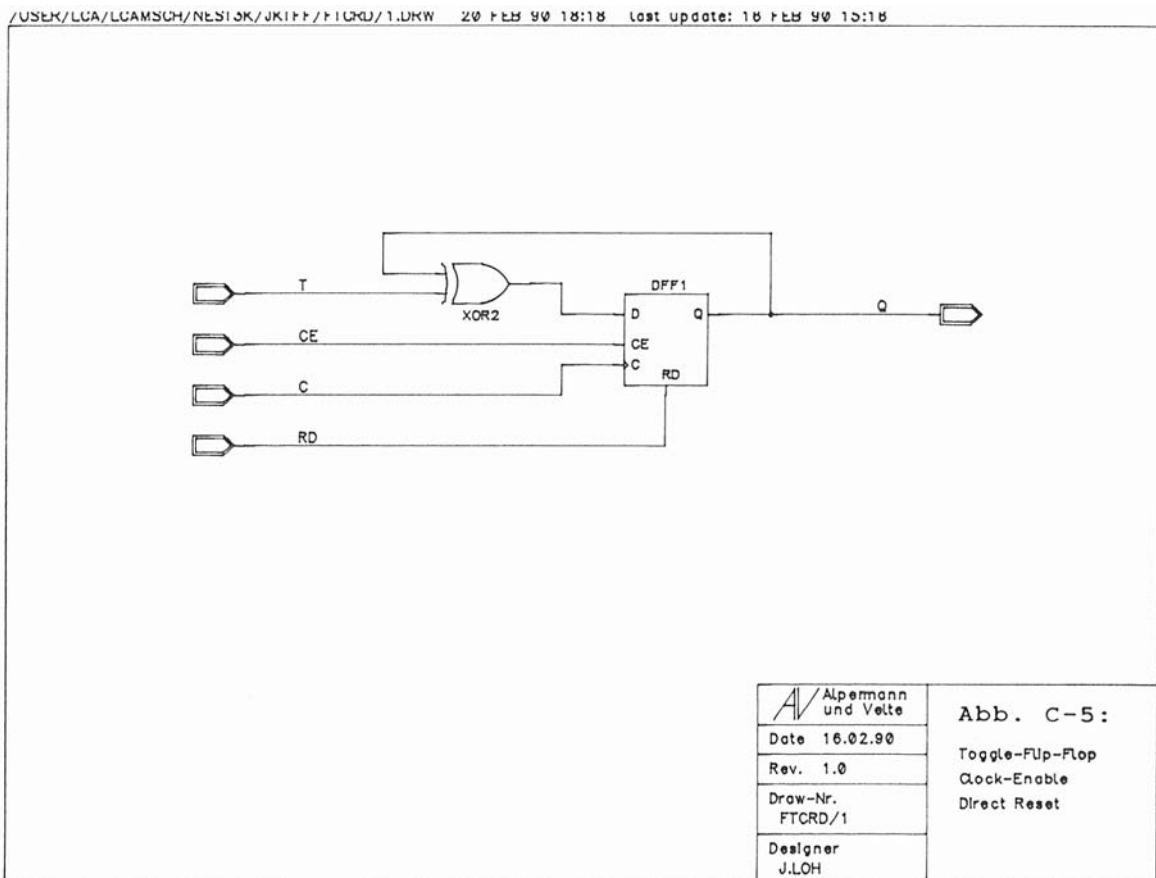


Abb. C-5: FTCRD: Toggle-Flip-Flop, Direct Reset

/USER/LCA/LCAMSCH/NESTOK/JKIFF/FTCPK/1.0KW 20 FEB 90 18:24 last update: 18 FEB 90 18:43

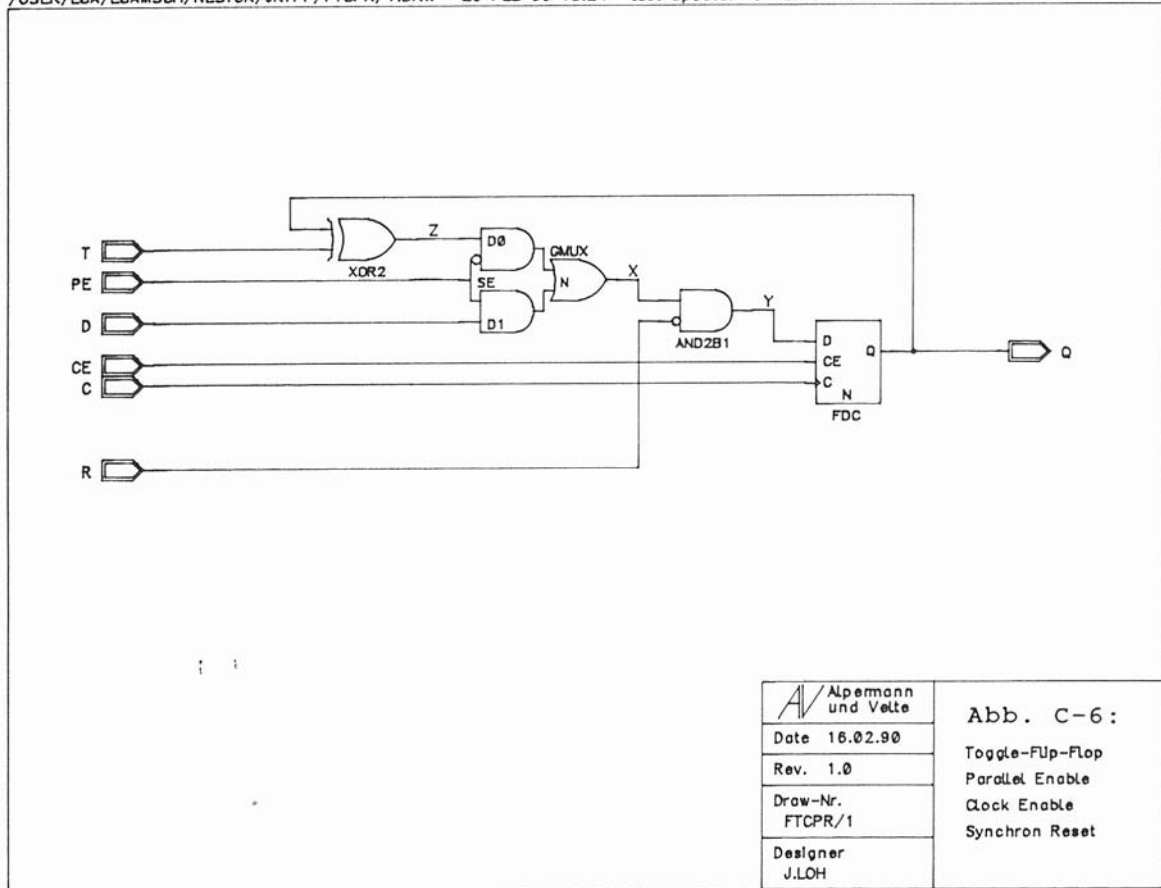


Abb. C-6: FTCPR: Ladbare Toggle-Flip-Flop, Synchron Reset

/USER/LCA/LCAMSCH/NESTOK/JKIFF/FTCPRD/1.0KW 20 FEB 90 19:17 last update: 18 FEB 90 15:24

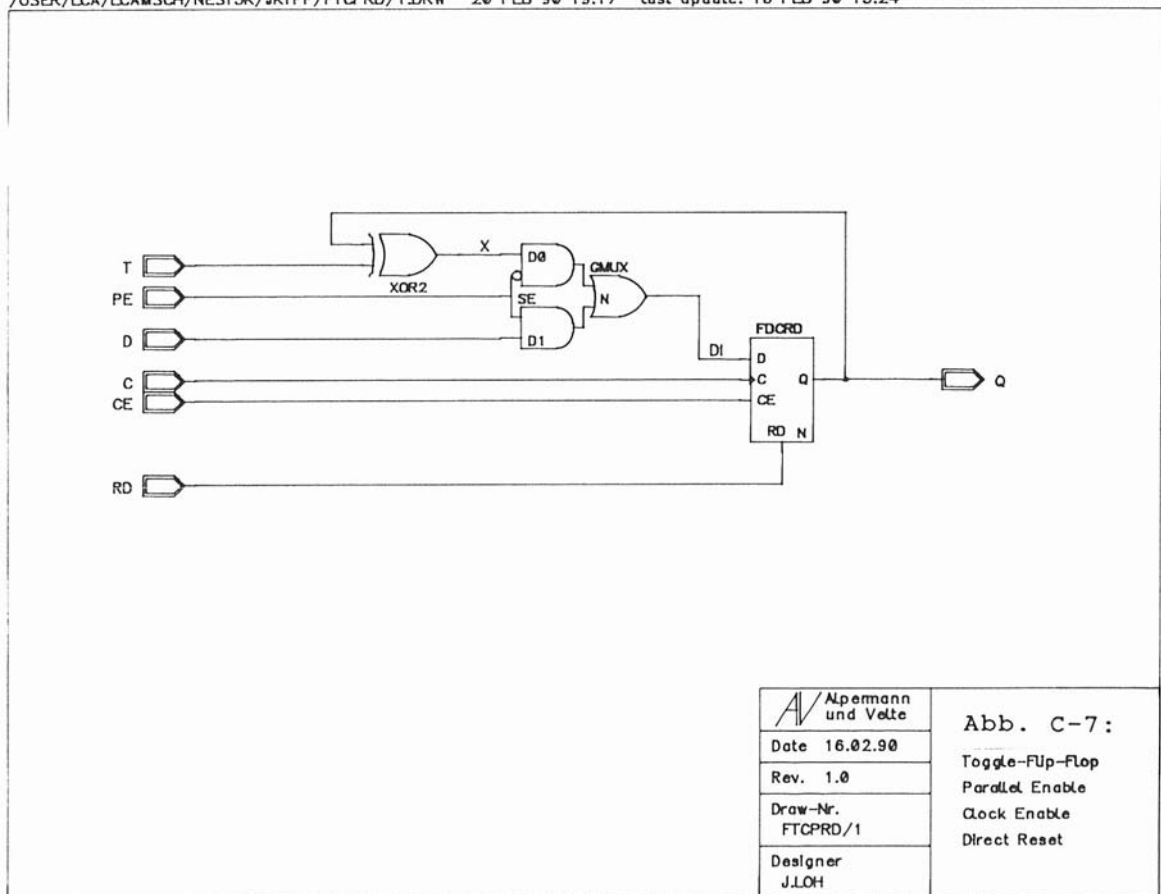


Abb. C-7: FTCPRD: Ladbare Toggle-Flip-Flop, Direct Reset

C.3. Binärzähler

C.3.1. Binärzähler modulo 4

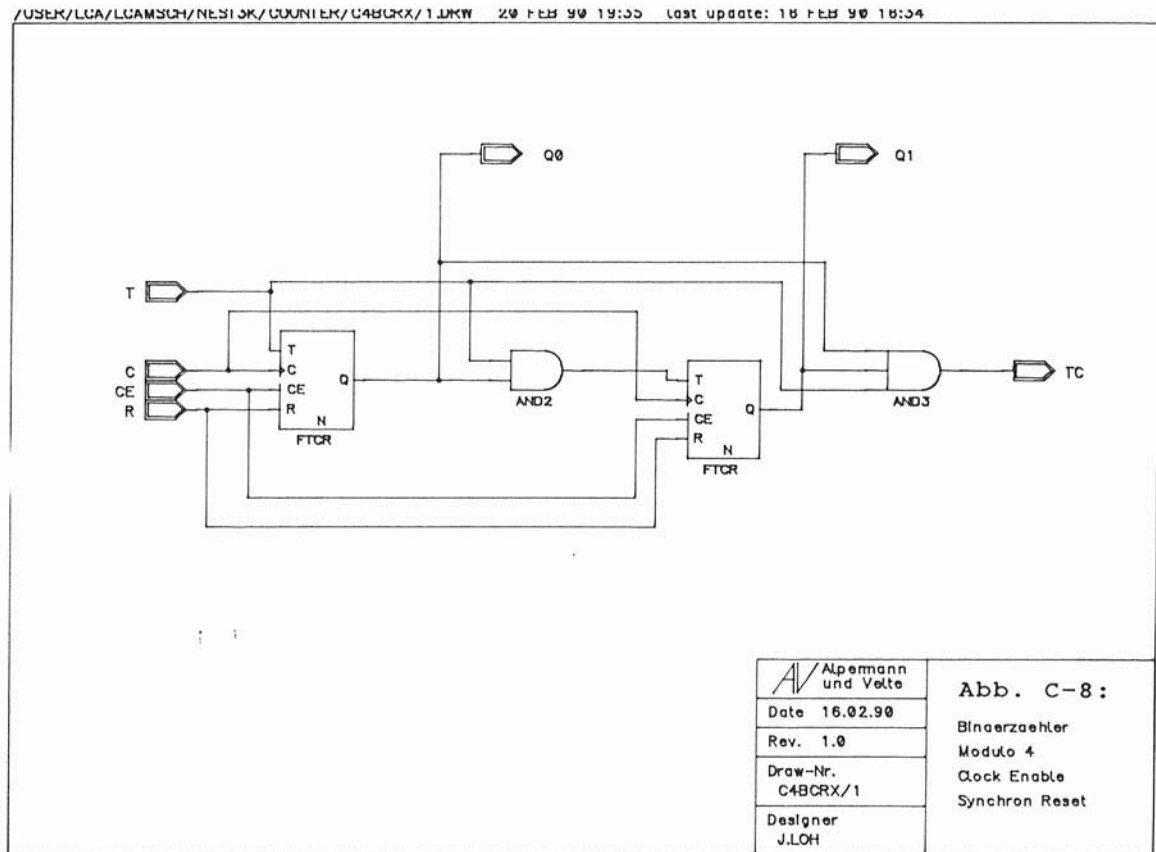


Abb. C-8: C4BCRX: Zähler modulo 4, Synchron Reset

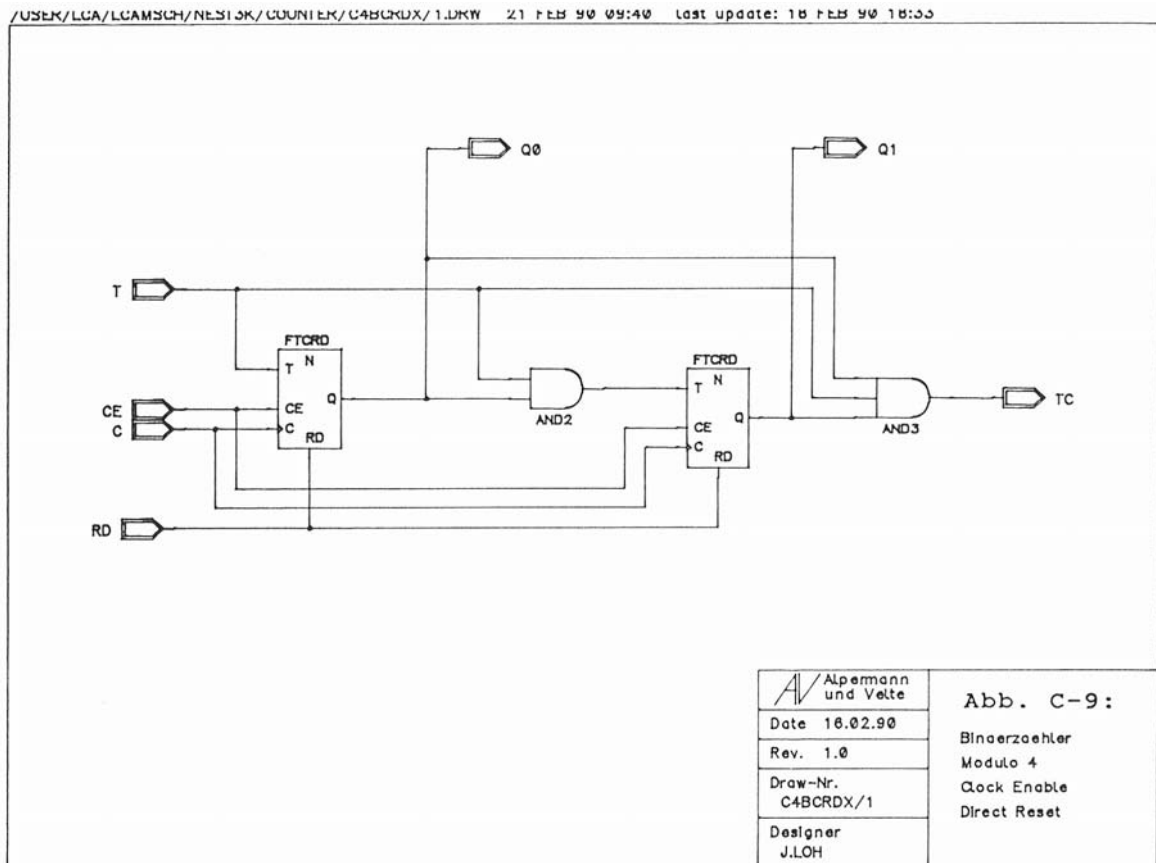


Abb. C-9: BCRDX: Zähler modulo 4, Direct Reset

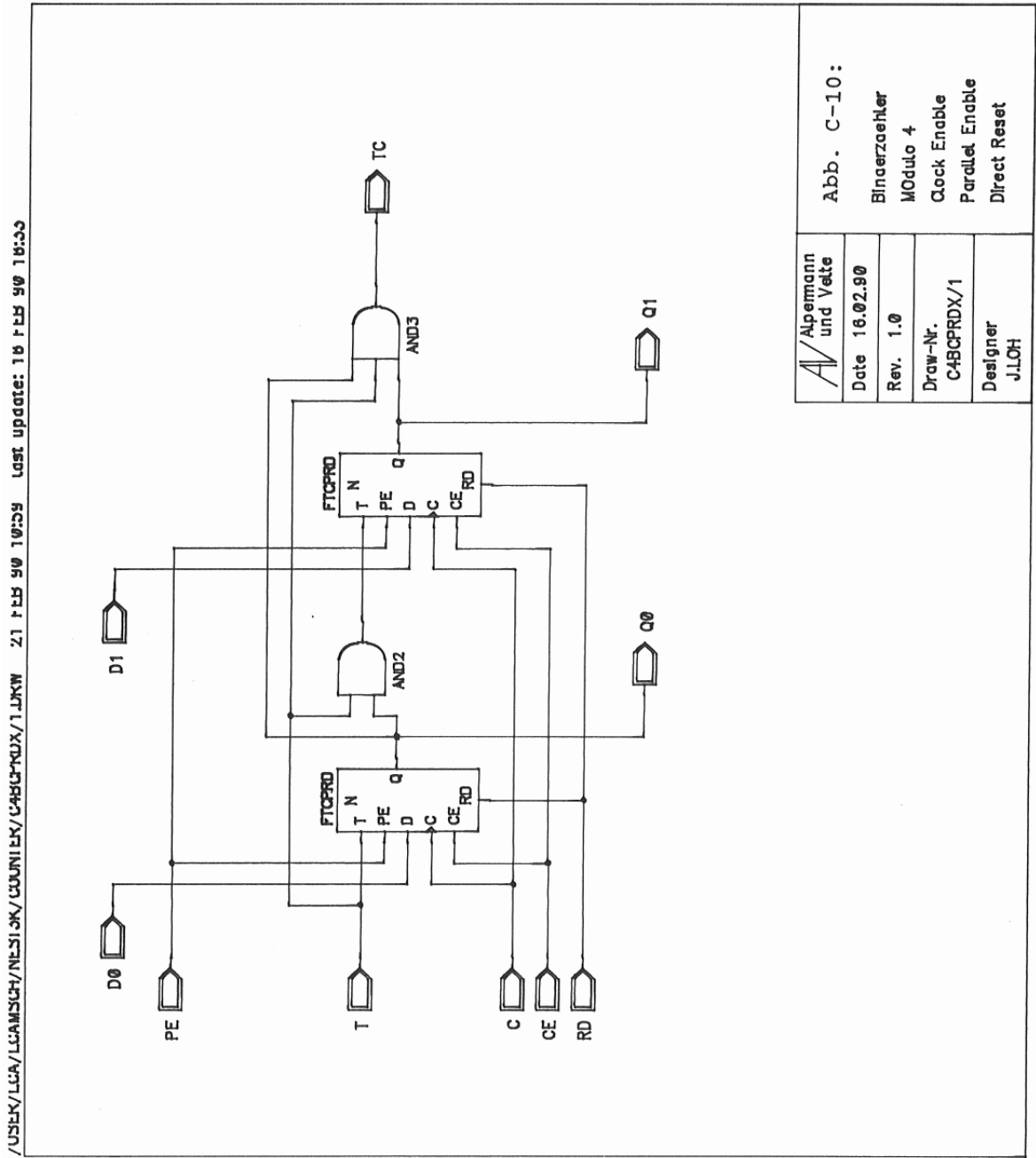


Abb. C-10: C4BCPRDX: Ladbarer Zähler modulo 4, Direct Reset

C.3.2. Binärzähler modulo 8

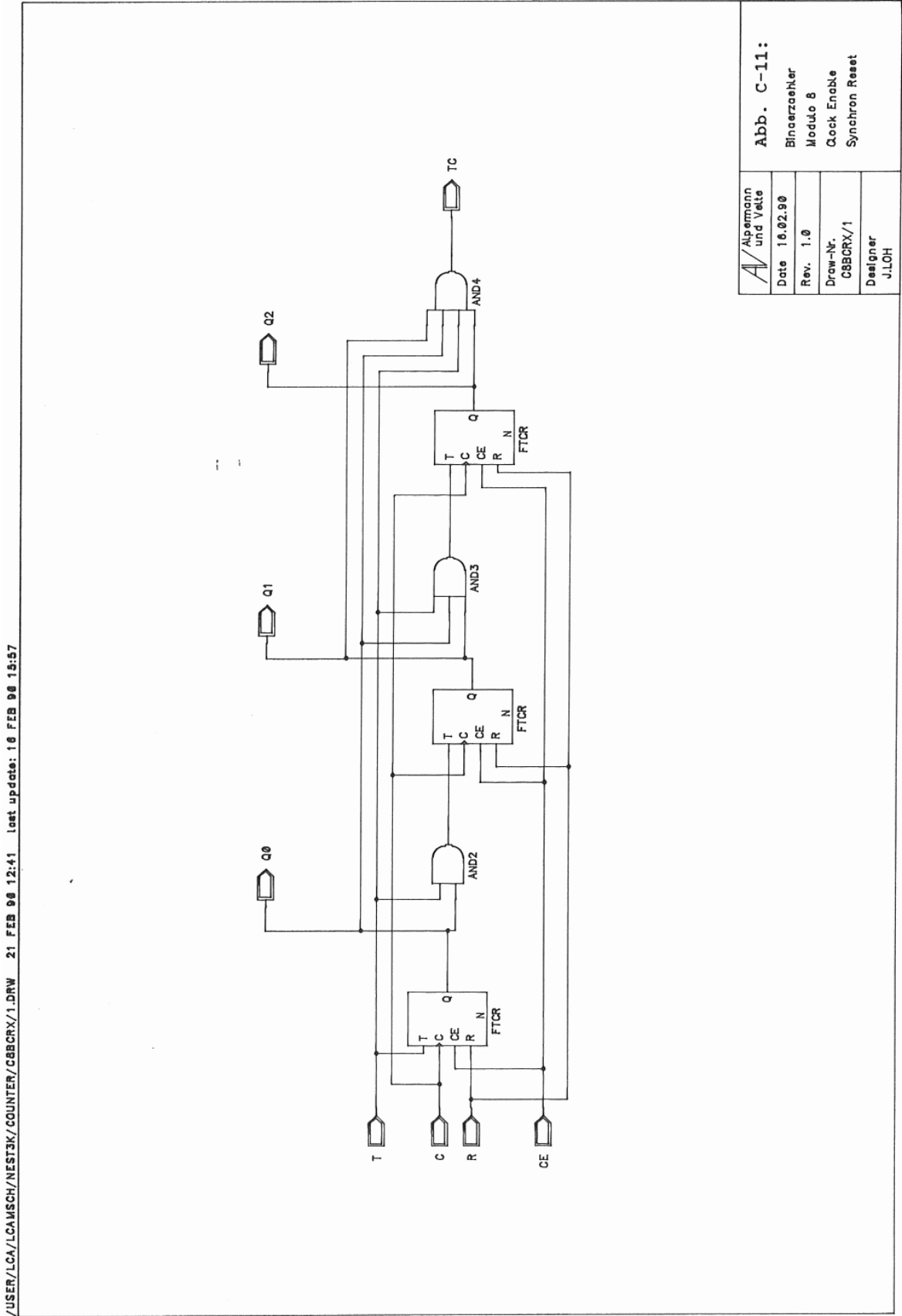


Abb. C-11: C8BCRX: Zähler modulo 8, Synchron Reset

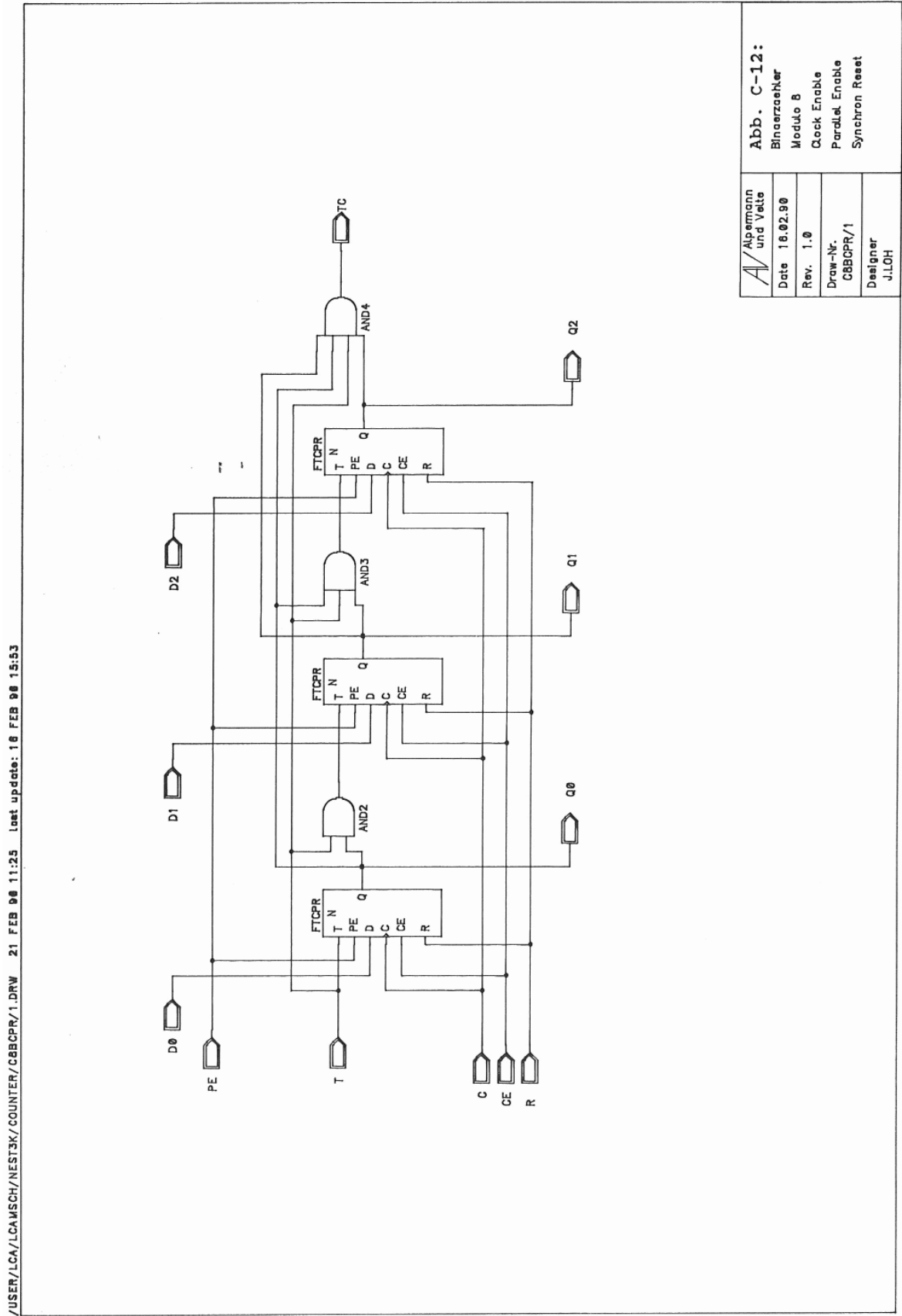
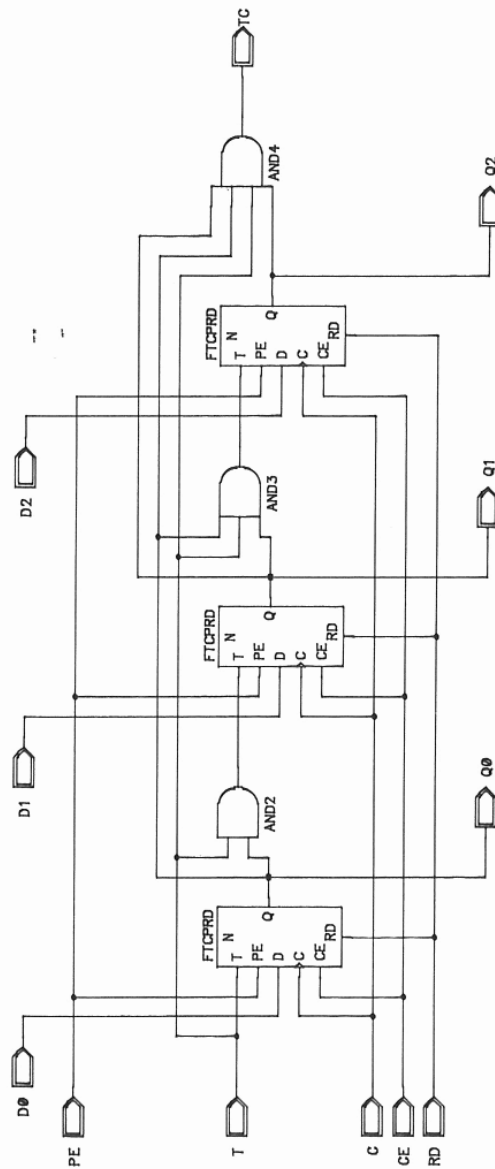


Abb. C-12: C8BCPR: Ladbarer Zähler modulo 8, Synchron Reset



<i>A</i> /Apermann und Vate	Abb. C-13:
Date 16.02.90	Binaerzeiher
Rev. 1.0	Modulo 8
Draw-Nr.	Lock Enable
C8BCPRDX/1	Parallel Enable
Designer	Direct Reset
J.LOH	

Abb. C-13: C8BCPRDX: Ladbarer Zähler modulo 8, Direct Reset

C.3.3. Binärzähler modulo 16

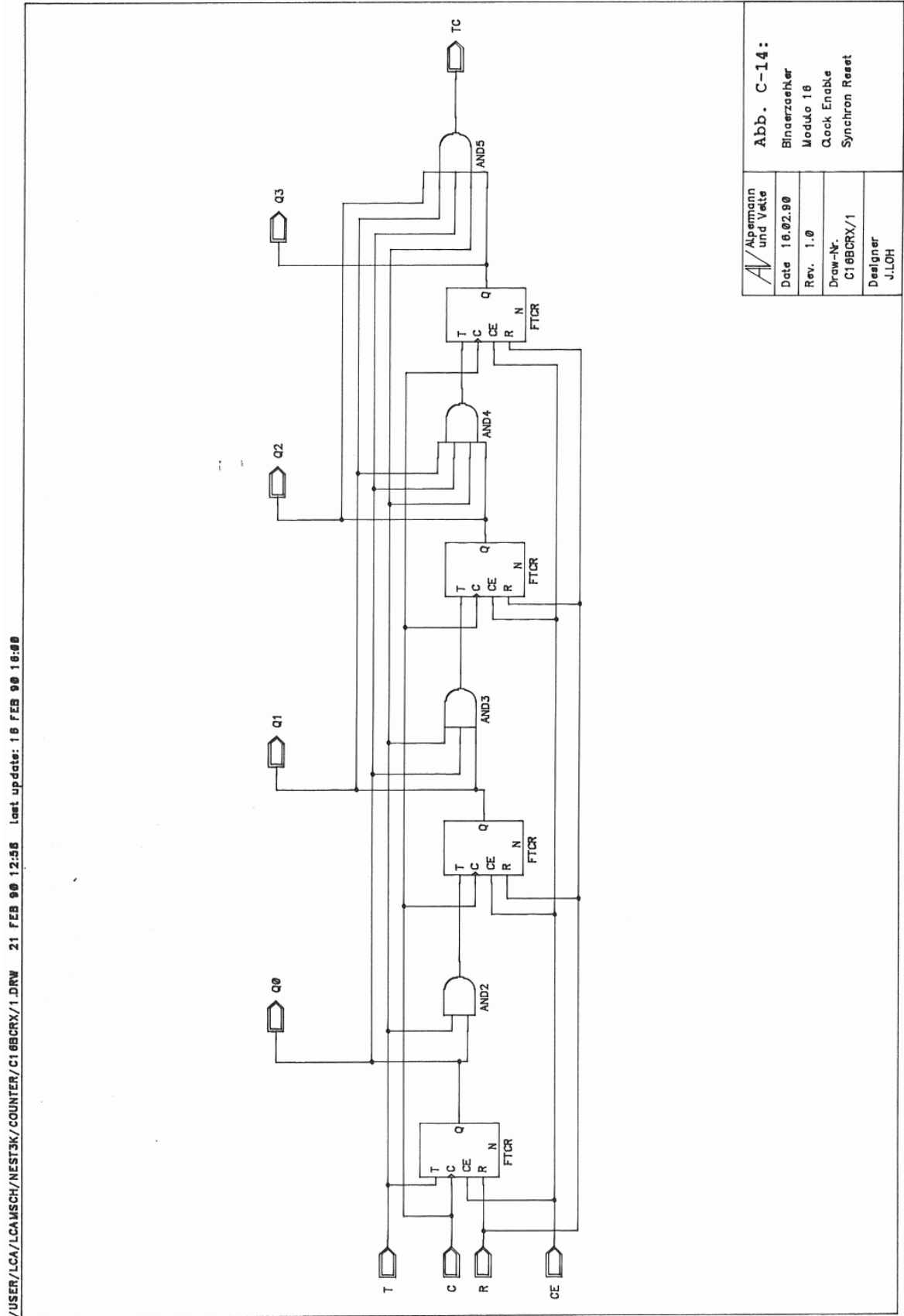


Abb. C-14: C16BCRX: Zähler modulo 16, Synchron Reset

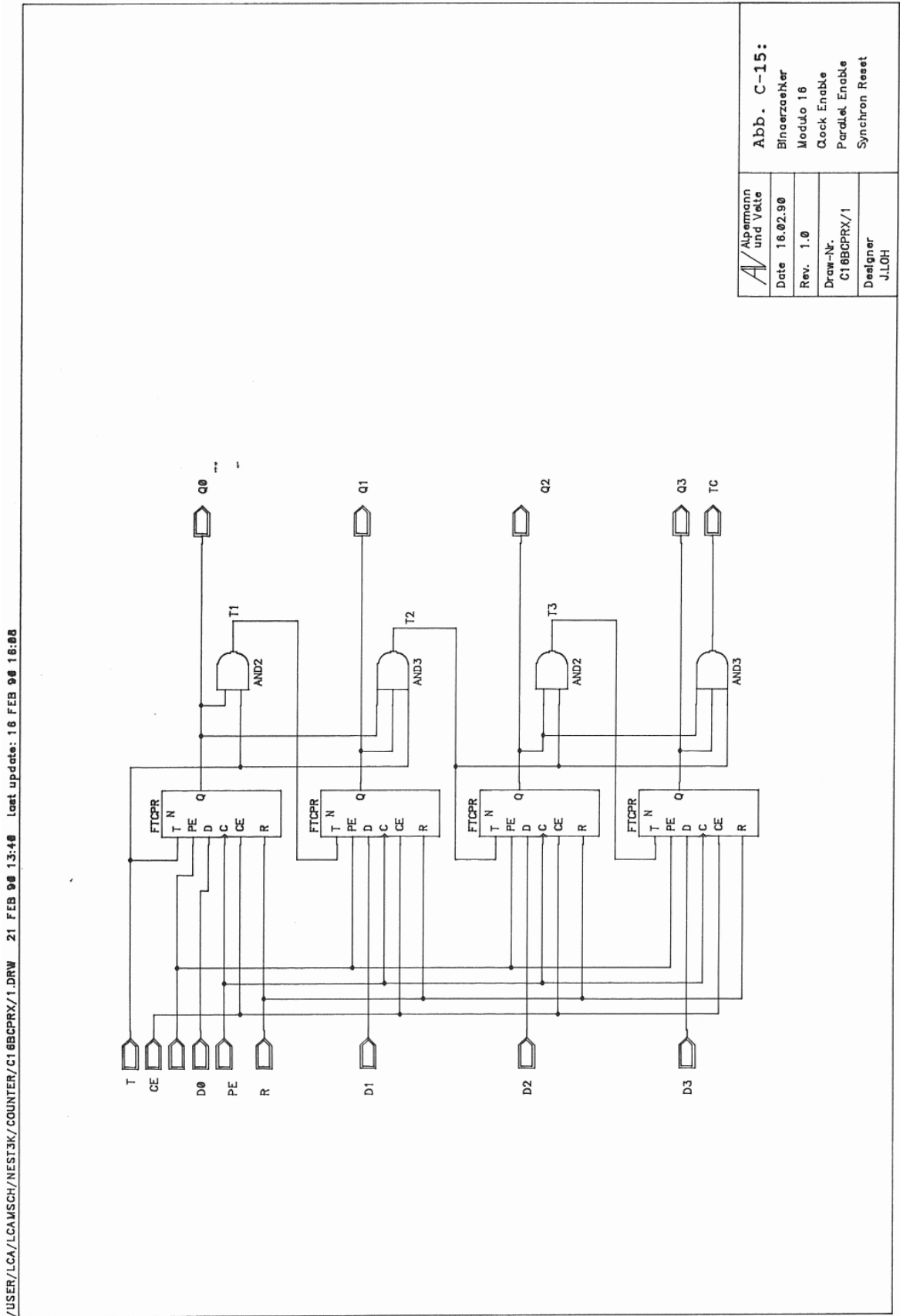


Abb. C-15: C16BCPRX: Ladbarer Zähler modulo 16, Synchron Reset

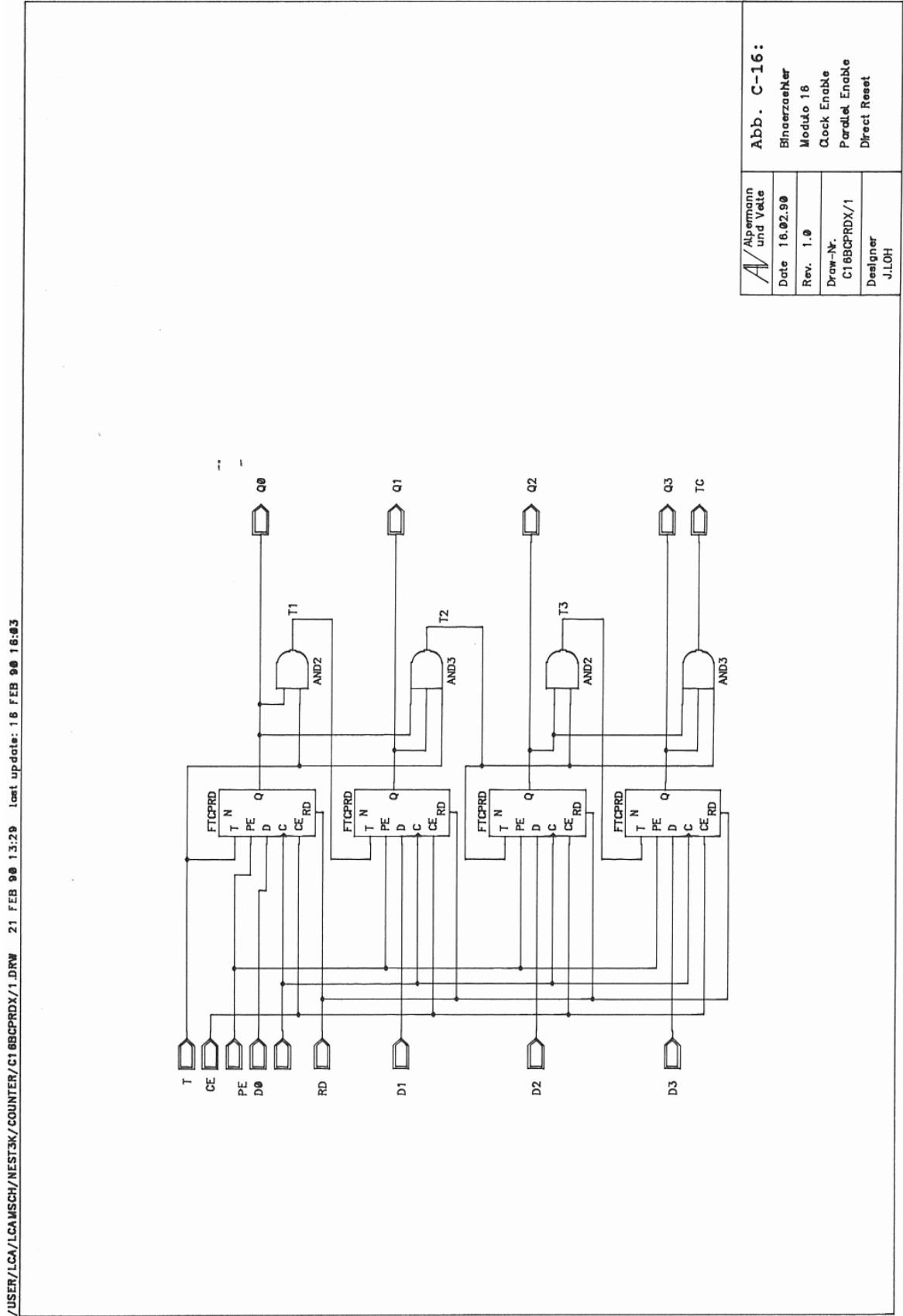


Abb. C-16: C16BCPRDX: Ladbarer Zähler modulo 16, Direct Reset

C.3.4. Binärzähler modulo 256

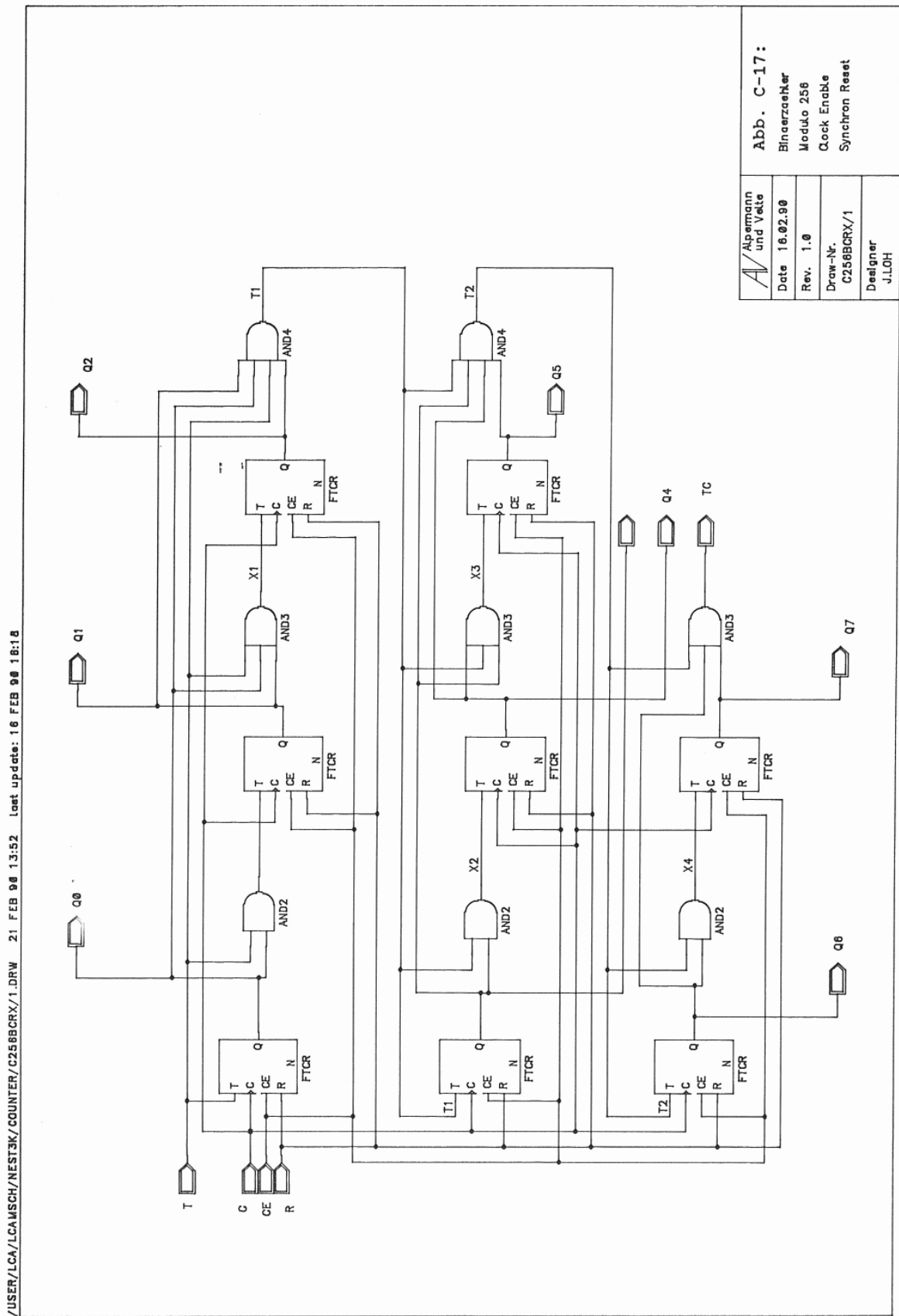
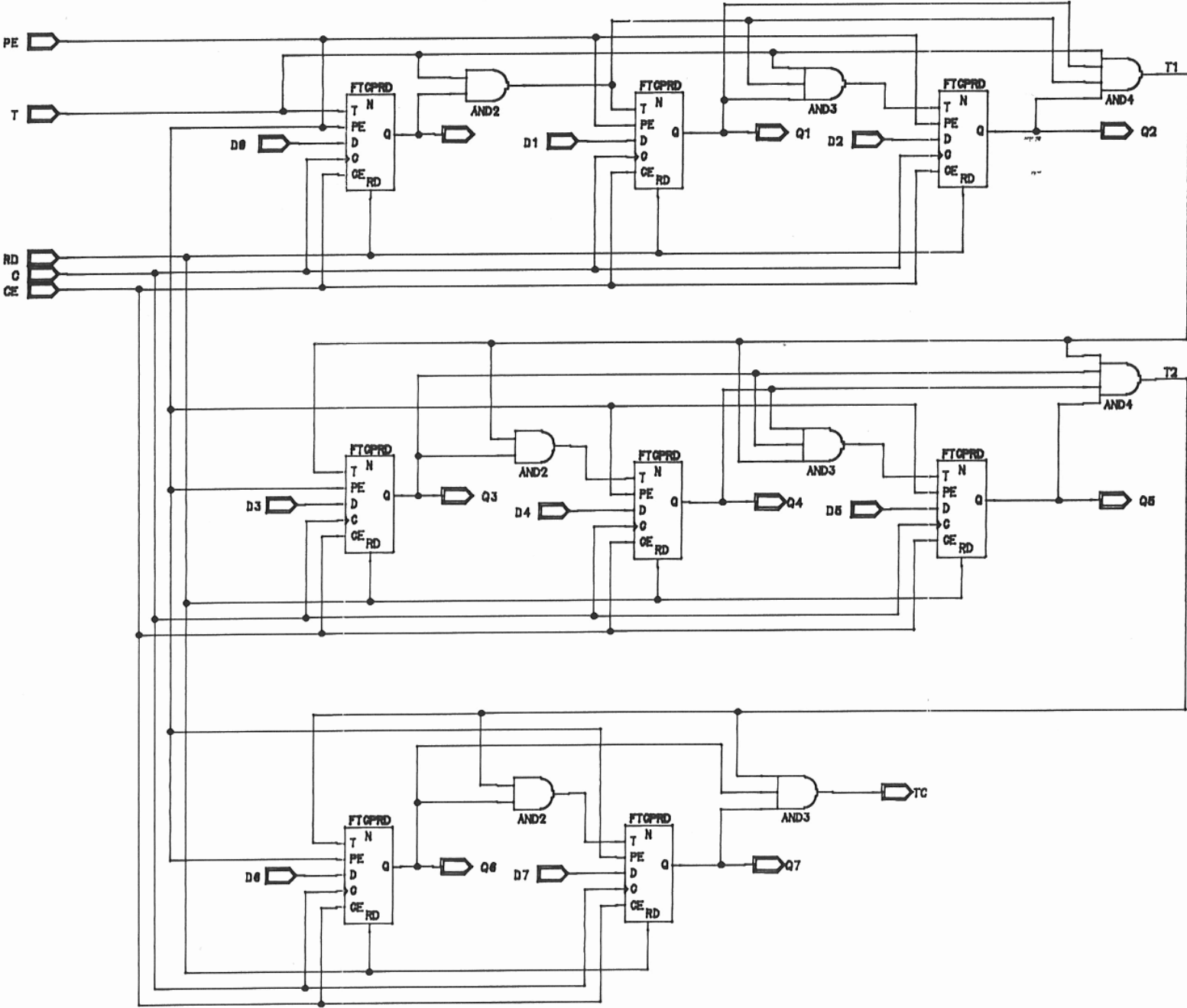


Abb. C-17: C256BCRX: Zähler modulo 256, Synchron Reset

USER/LCA/LCAMSCH/NEST3K/COUNTER/C256BCPRDX/1.DRW 21 FEB 90 14:04 last update: 10 FEB 90 10:17



Alpermann und Velle	Abb. C-18:
Date 10.02.90	Binärzähler
Rev. 1.0	Modulo 256
Draw-Nr. C256BCPRDX/1	Clock Enable
Designer J.LOH	Parallel Enable
	Direct Reset

Abb. C-18: C256BCPRDX: Ladbarer Zähler modulo 256, Direct Reset

C.4. I/O-Pads

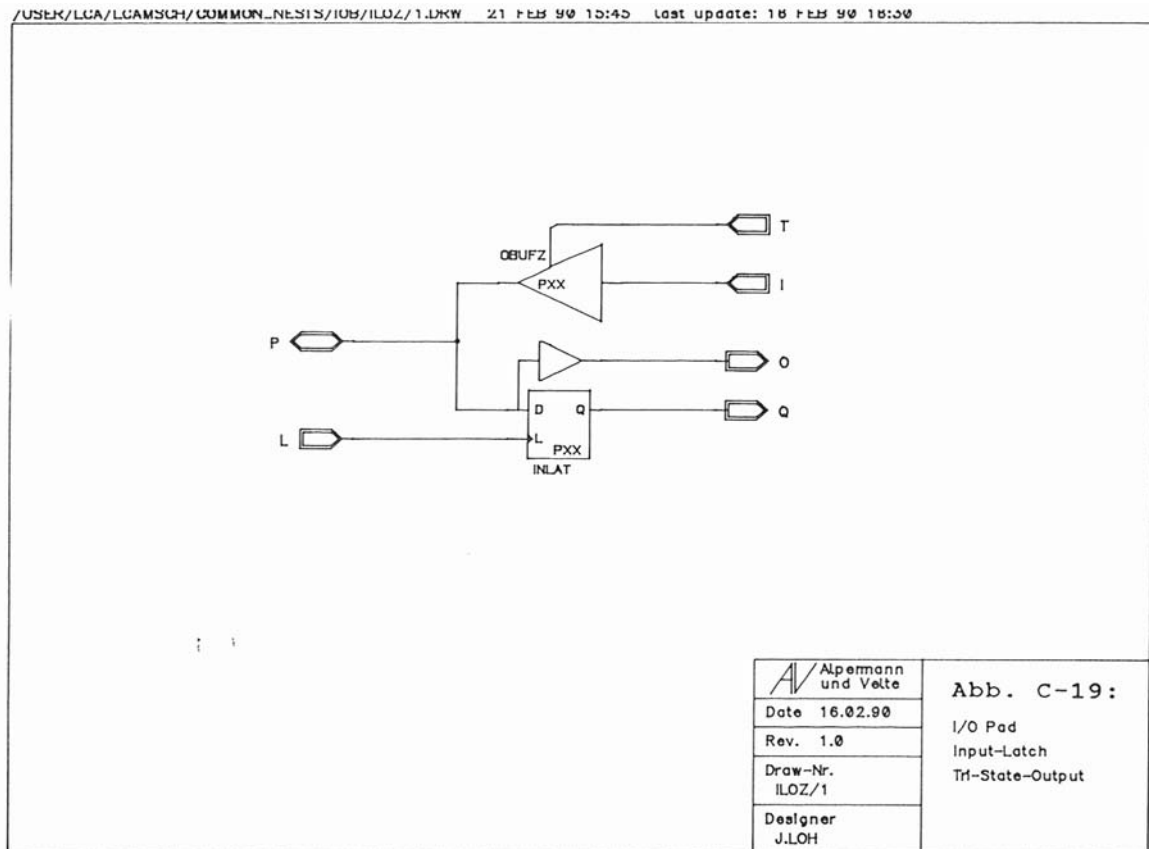


Abb. C-19: ILOZ: I/O-Pad mit Input-Latch und Tri-State-Output

C.5. Addierer/Subtrahierer

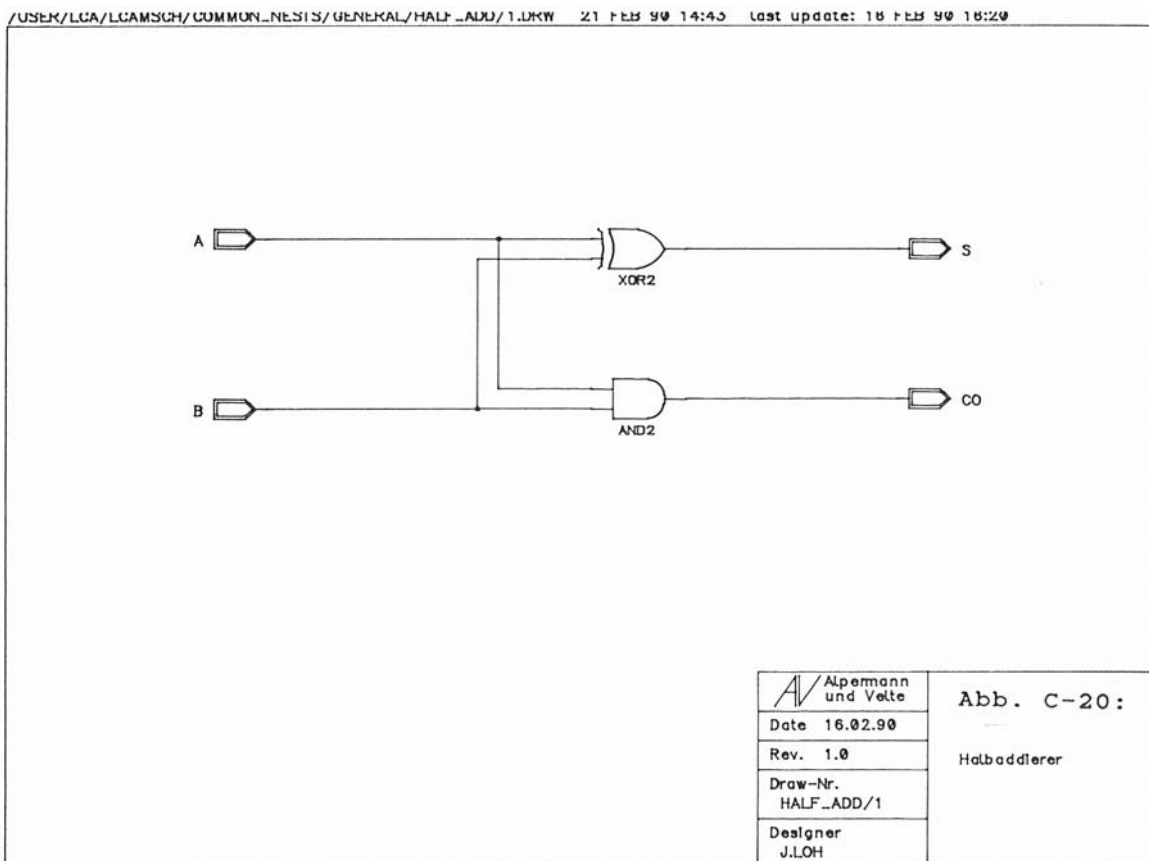


Abb. C-20: HALF_ADD: Halbaddierer

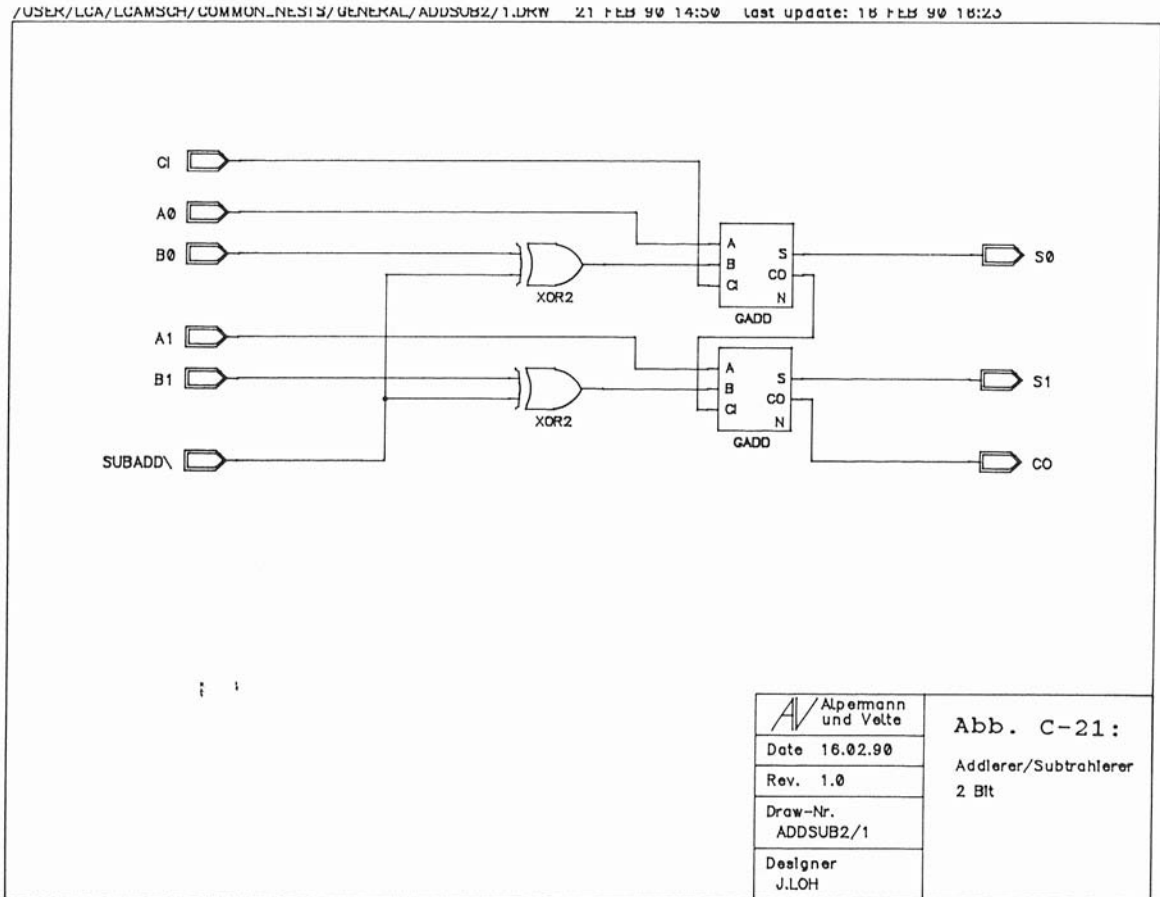


Abb. C-21: ADDSUB2: 2-Bit-Addierer/Subtrahierer

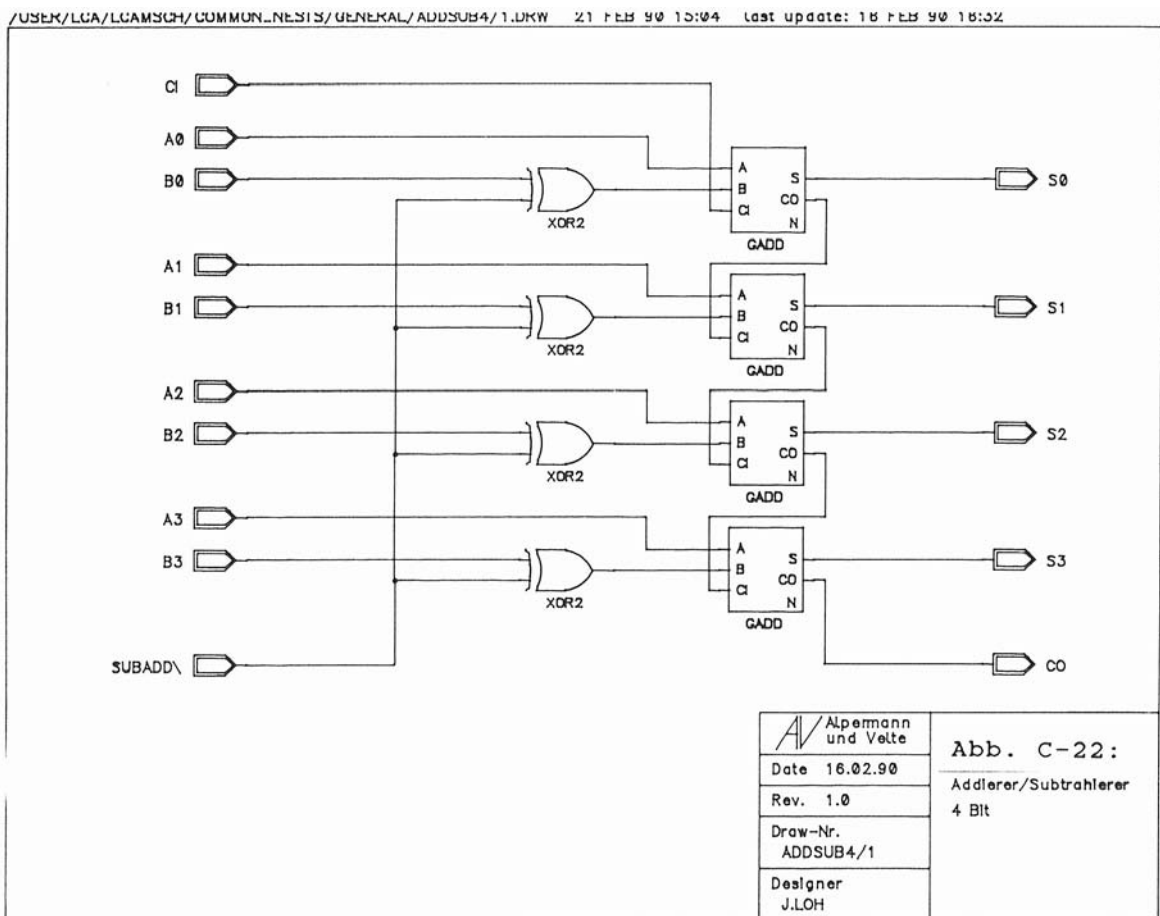
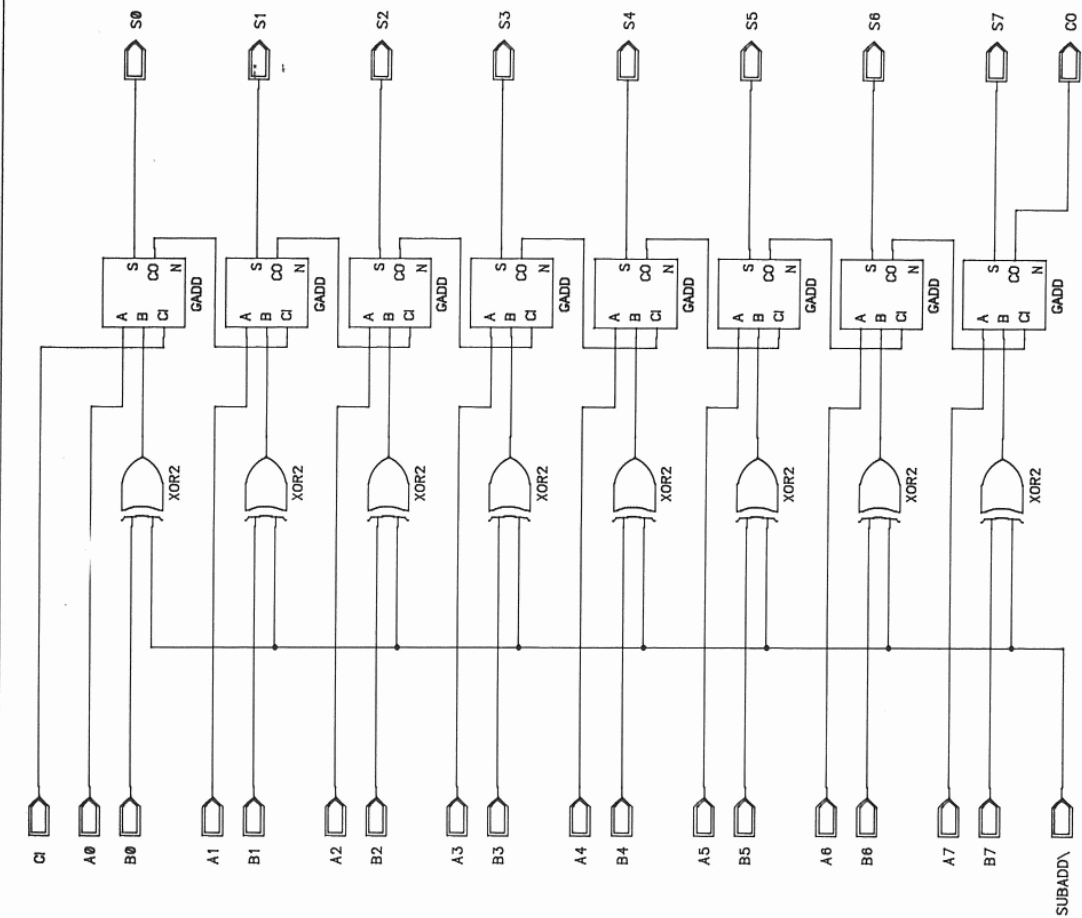


Abb. C-22: ADDSUB4: 4-Bit-Addierer/Subtrahierer

/USER/LCA/LCAMSCH/COMMON_NESTS/GENERAL/ADDSUB8/1.DRW 21 FEB 90 15:35 Last update: 16 FEB 90 18:31



A/Alpennann und Vette	
Date	16.02.90
Rev.	1.0
Draw-Nr.	ADDSUB8/1
Designer	J.LOH

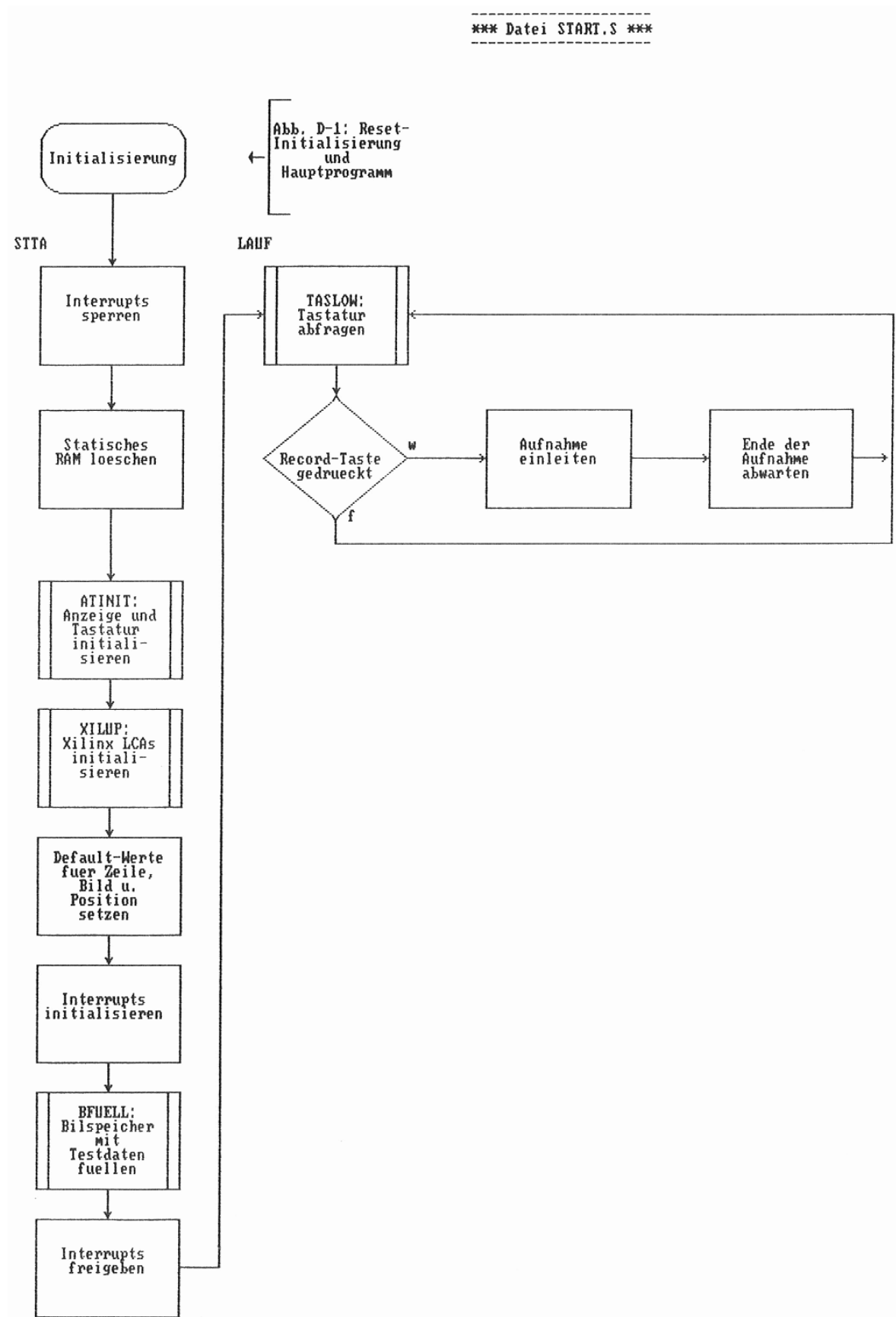
Abb. C-23:

Addierer/Subtrahierer
8 Bit

Abb. C-23: ADDSUB8: 8-Bit-Addierer/Subtrahierer

D. Flußdiagramme

D.1. Flußdiagramme zu START.S



*** Datei START.S ***

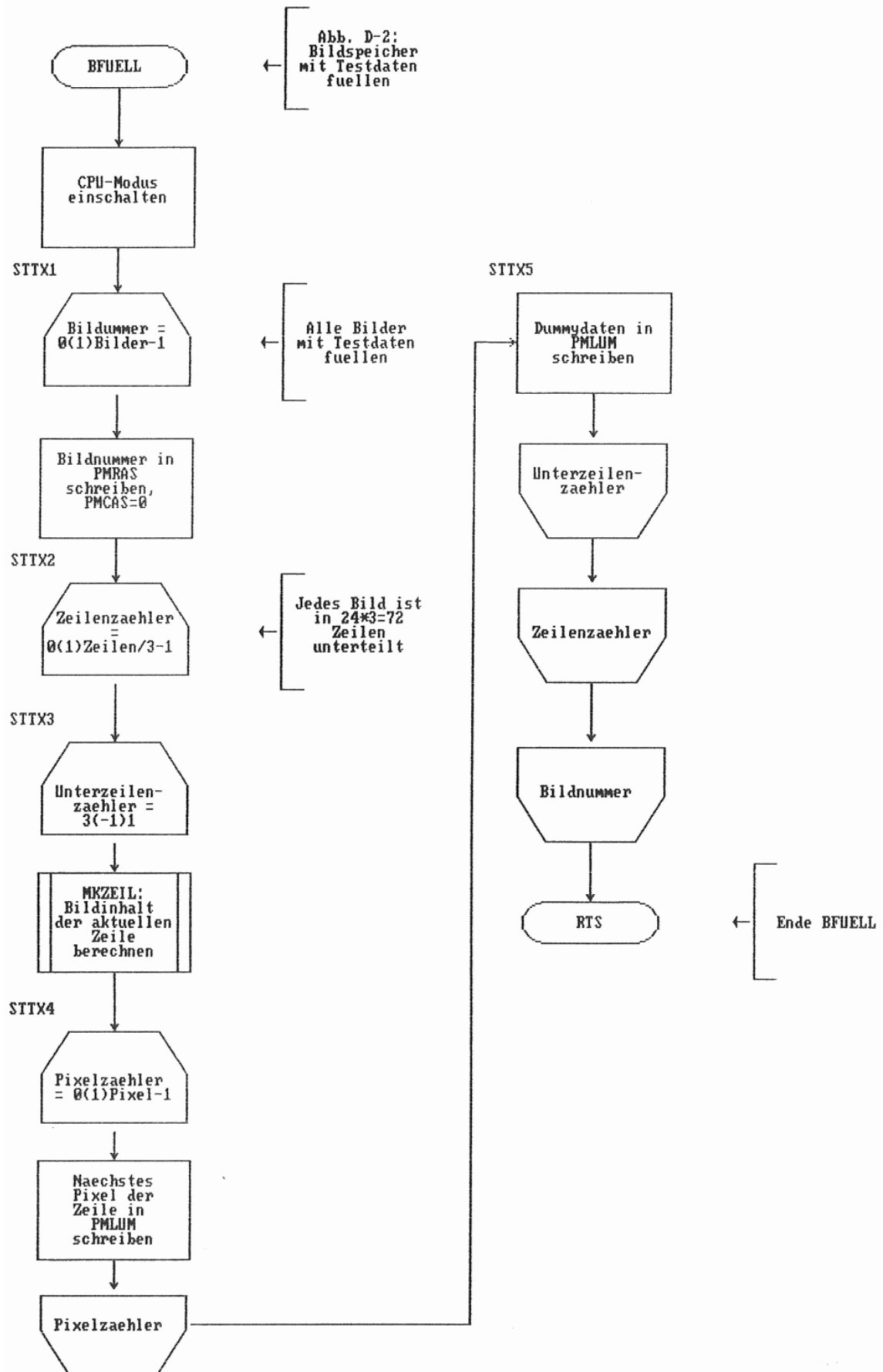


Abb. D-2: BFUELL: Bildspeicher mit Testdaten füllen

*** Datei START.S ***

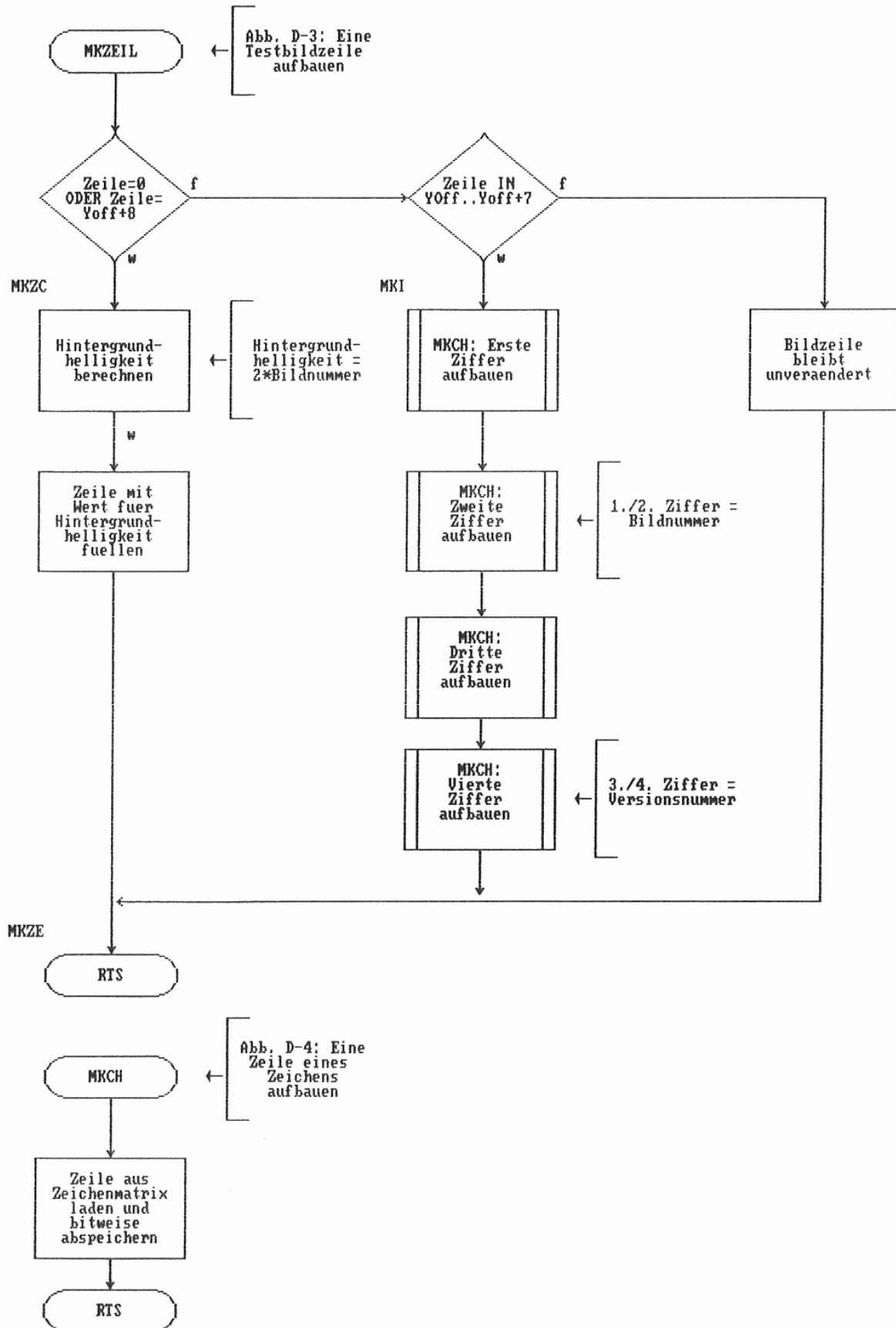


Abb. D-3: MKZEIL: Testbildzeile aufbauen

Abb. D-4: MKCH: Zeile eines Zeichens aufbauen

D.2. Flußdiagramme zu IRQ.S

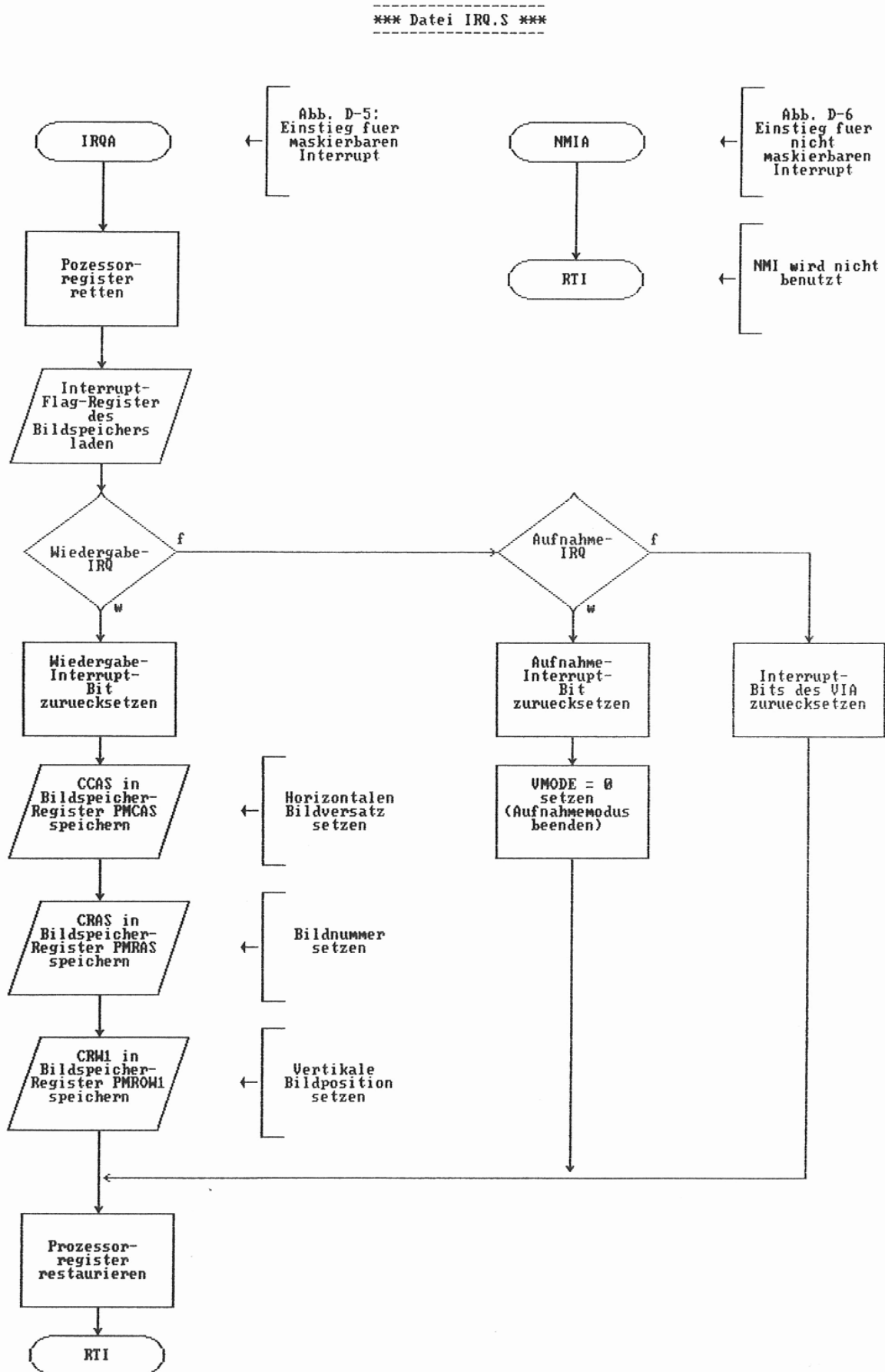


Abb. D-5: IRQA: Routine für maskierbaren Interrupt

Abb. D-6: NMIA: Routine für nicht maskierbaren Interrupt

D.3. Flußdiagramme zu TAST.S

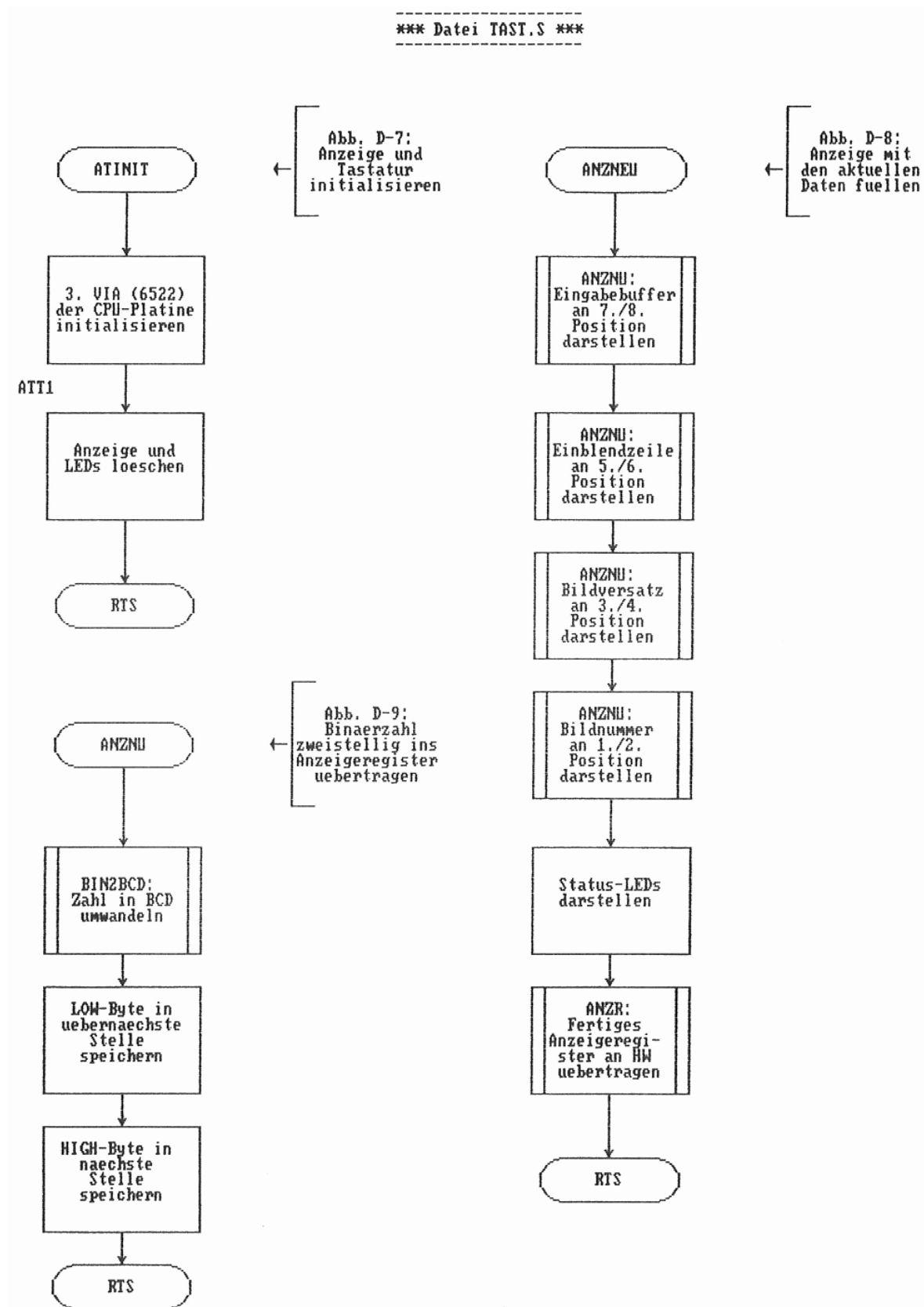


Abb. D-7: ATINIT: Anzeige und Tastatur initialisieren

Abb. D-8: ANZNEU: Anzeige mit aktuellen Daten füllen

Abb. D-9: ANZNU: Binärzahl zweistellig darstellen

*** Datei TAST.S ***

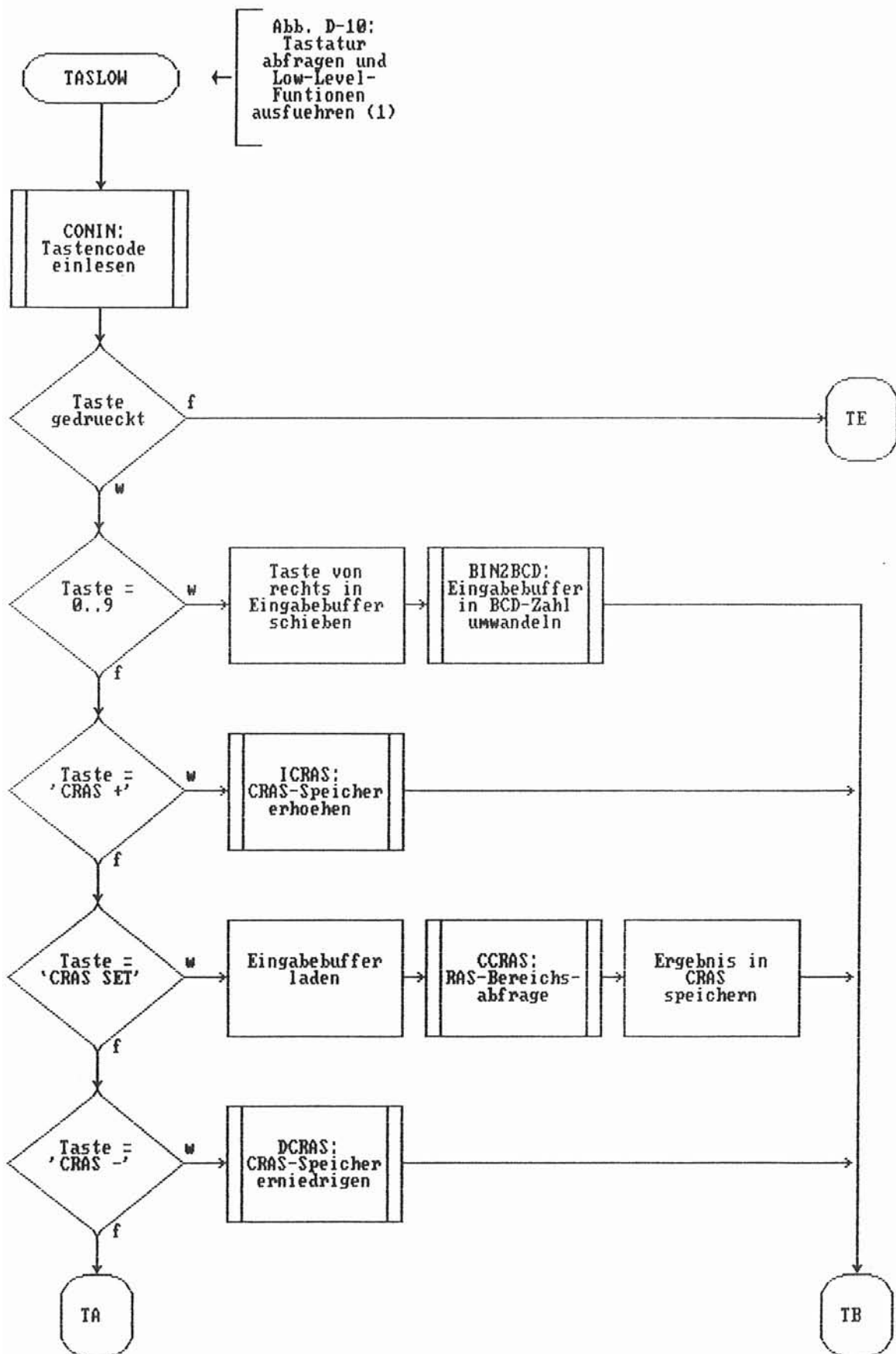


Abb. D-10: TASLOW: Low-Level-Tastaturfunktion ausfuehren (1)

*** Datei TAST.S ***

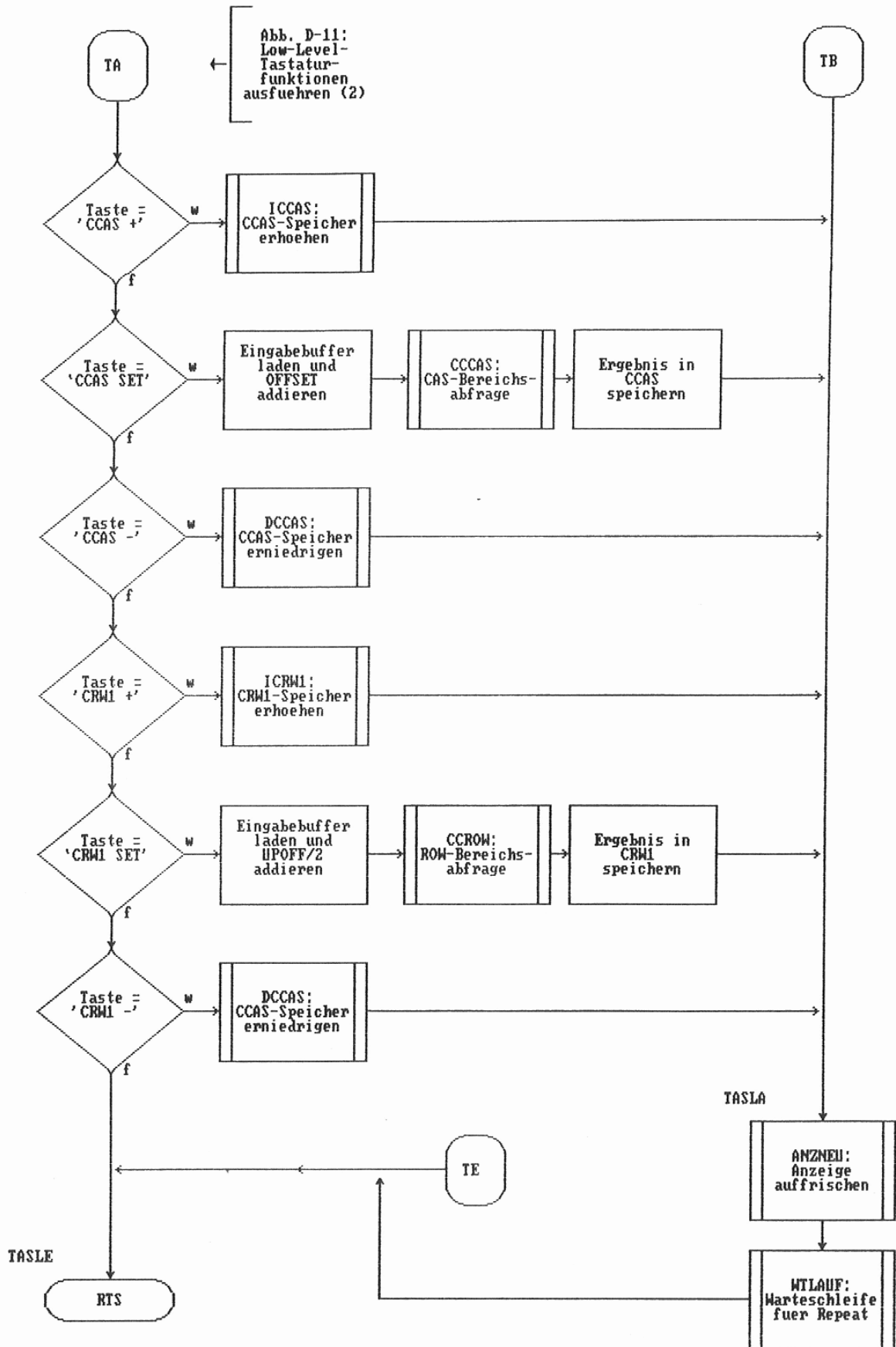


Abb. D-11: TASLOW: Low-Level-Tastaturfunktion ausführen (2)

*** Datei TAST.S ***

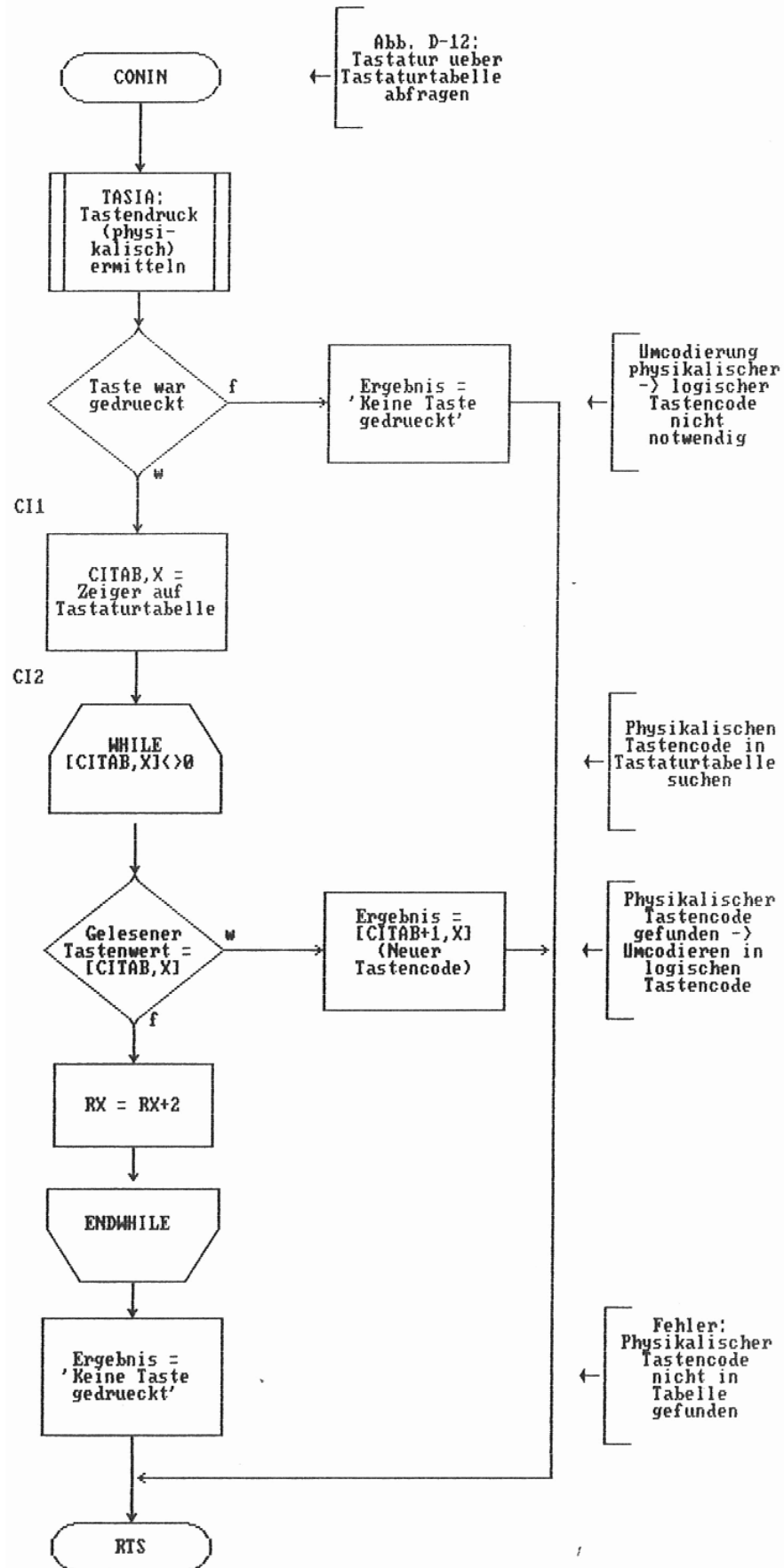


Abb. D-12: CONIN: Tastatur über Tabelle abfragen

*** Datei TAST.S ***

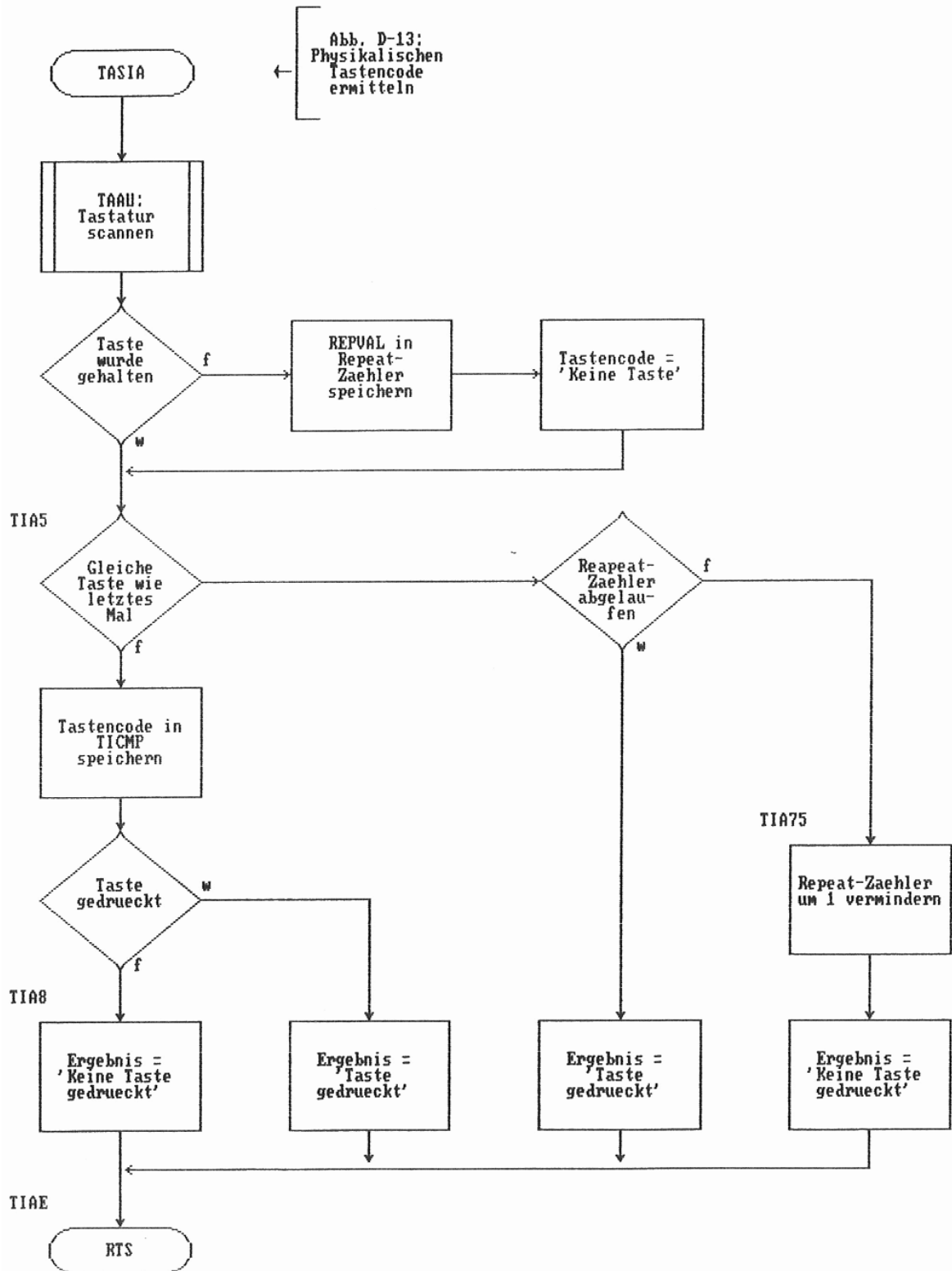
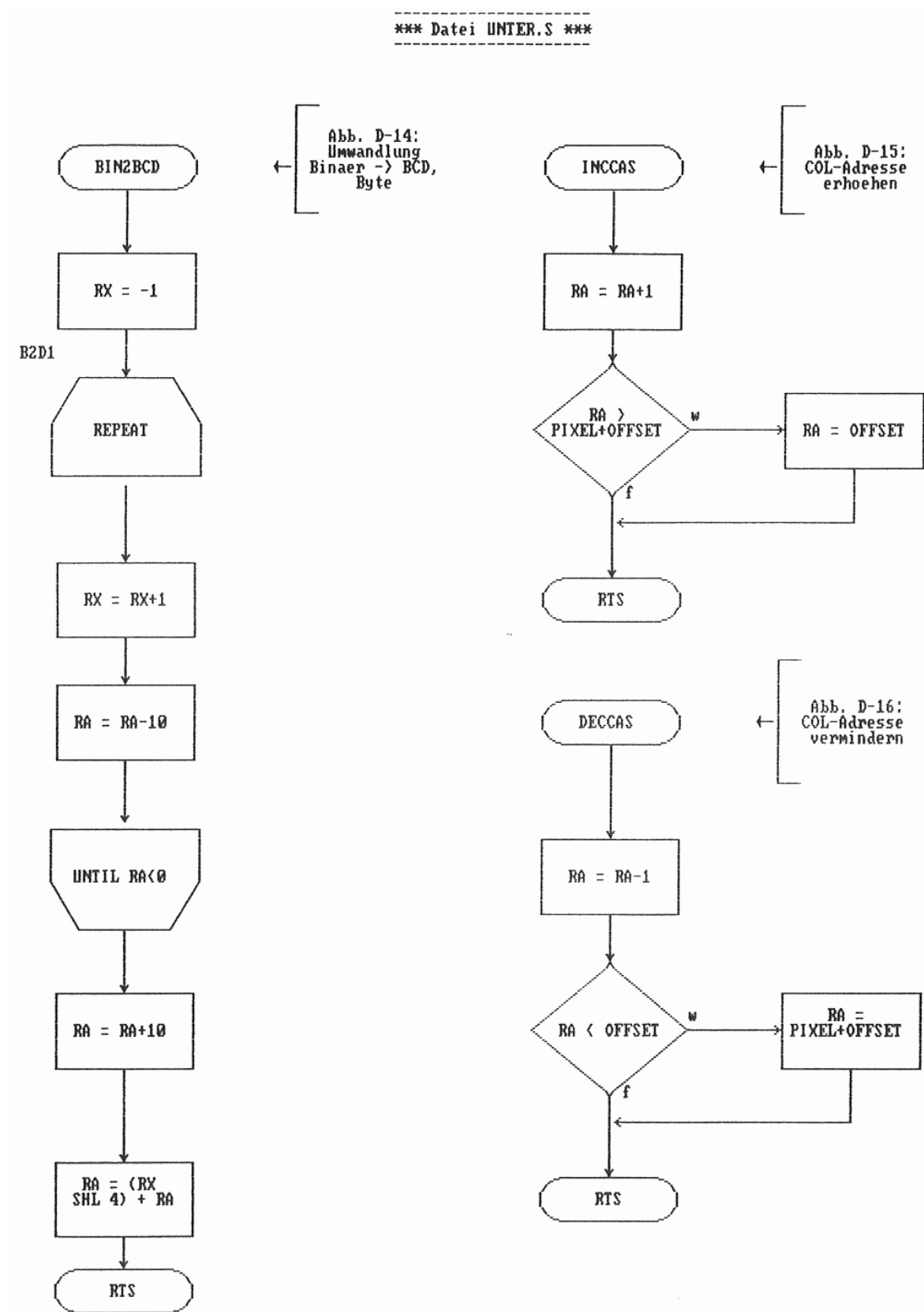


Abb. D-13: TASIA: Physikalischen Tastencode ermitteln

D.4. Flußdiagramme zu UNTER.S



*** Datei UNTER.S ***

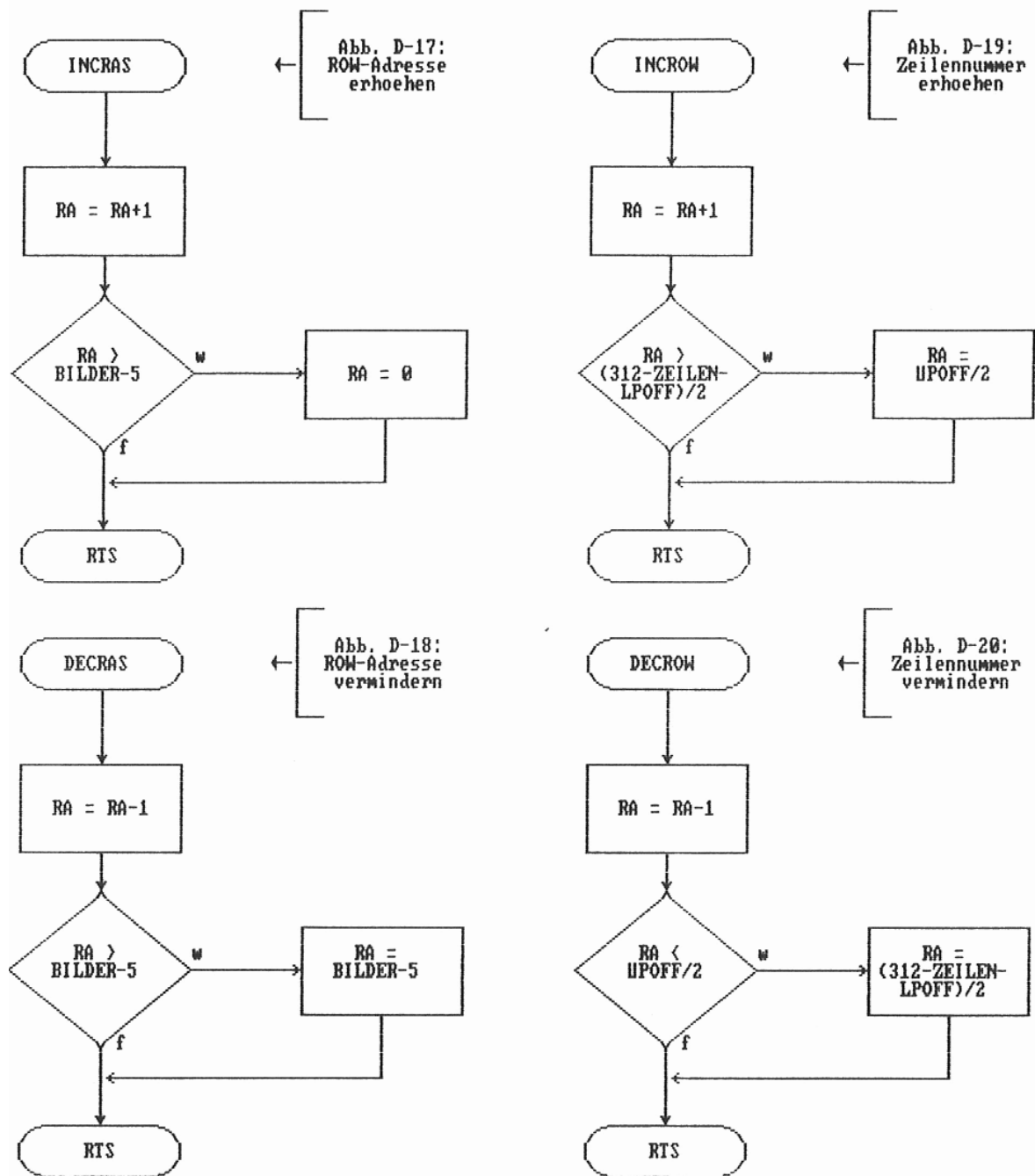


Abb. D-17: INCRAS: ROW-Adresse erhöhen

Abb. D-18: DECRAS: ROW-Adresse vermindern

Abb. D-19: INCROW: Zeilennummer erhöhen

Abb. D-20: DECROW: Zeilennummer vermindern

*** Datei UNTER.S ***

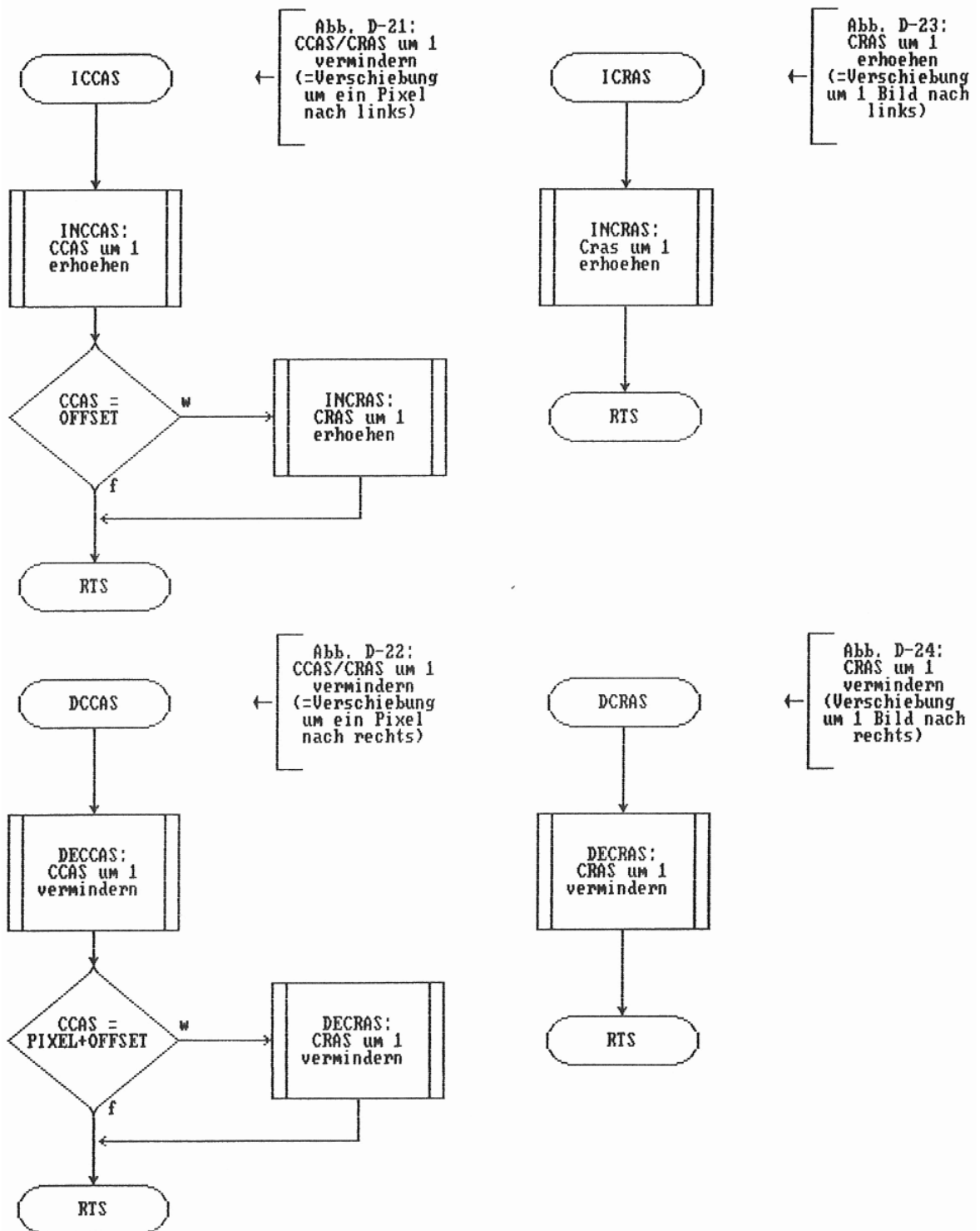


Abb. D-21: ICCAS: Versch. der Bilder um ein Pixel nach links
 Abb. D-22: DCCAS: Versch. der Bilder um ein Pixel nach rechts
 Abb. D-23: ICRAS: Versch. der Bilder um ein Bild nach links
 Abb. D-24: DCRAS: Versch. der Bilder um ein Bild nach rechts

*** Datei UNTER.S ***

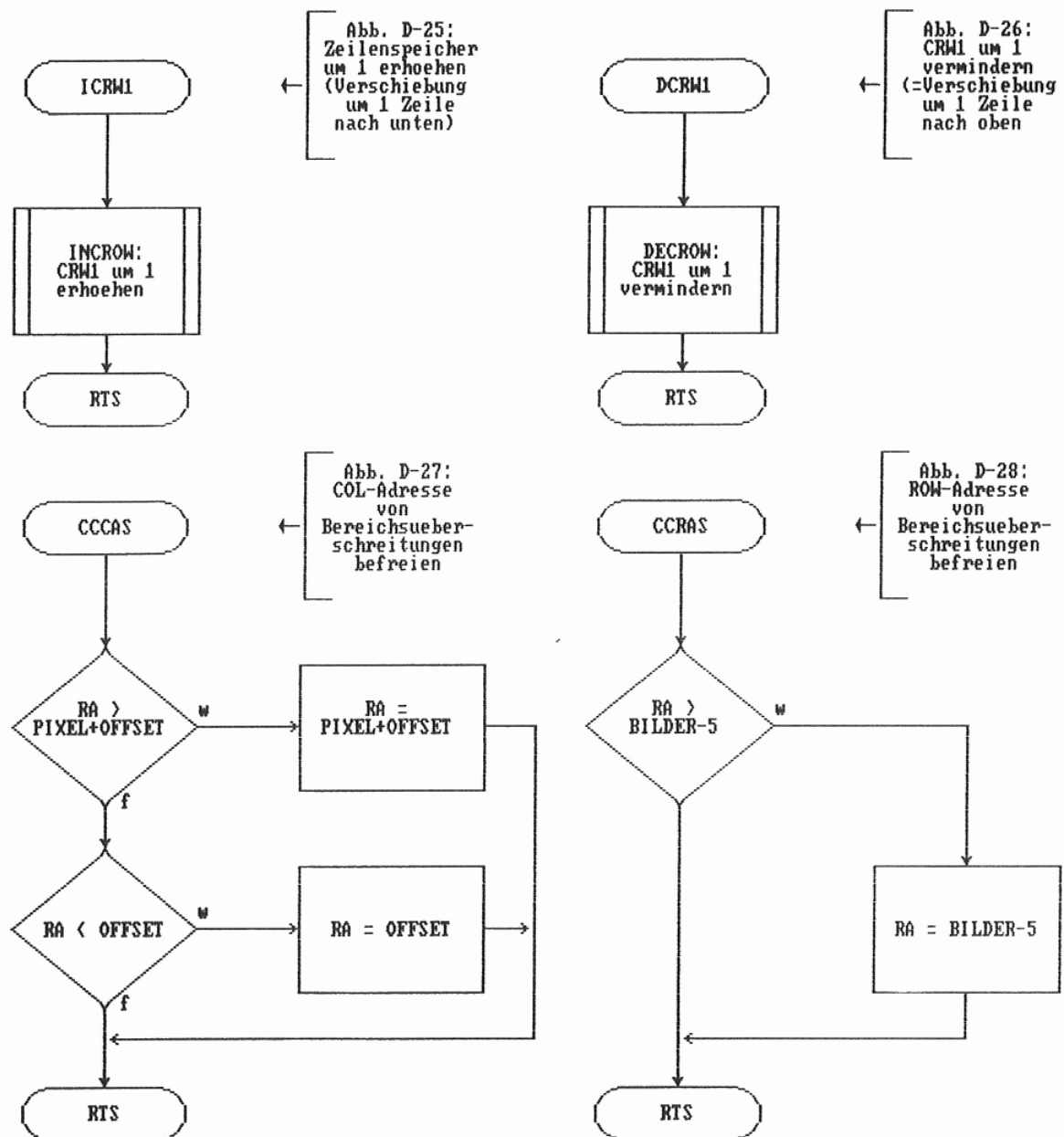


Abb. D-25: ICRW1: Versch. der Bilder um eine Zeile nach oben

Abb. D-26: DCRW1: Versch. der Bilder um eine Zeile nach unten

Abb. D-27: CCCAS: COL-Adressen-Bereichsabfrage

Abb. D-28: CCRAS: ROW-Adressen-Bereichsabfrage

*** Datei UNTER.S ***

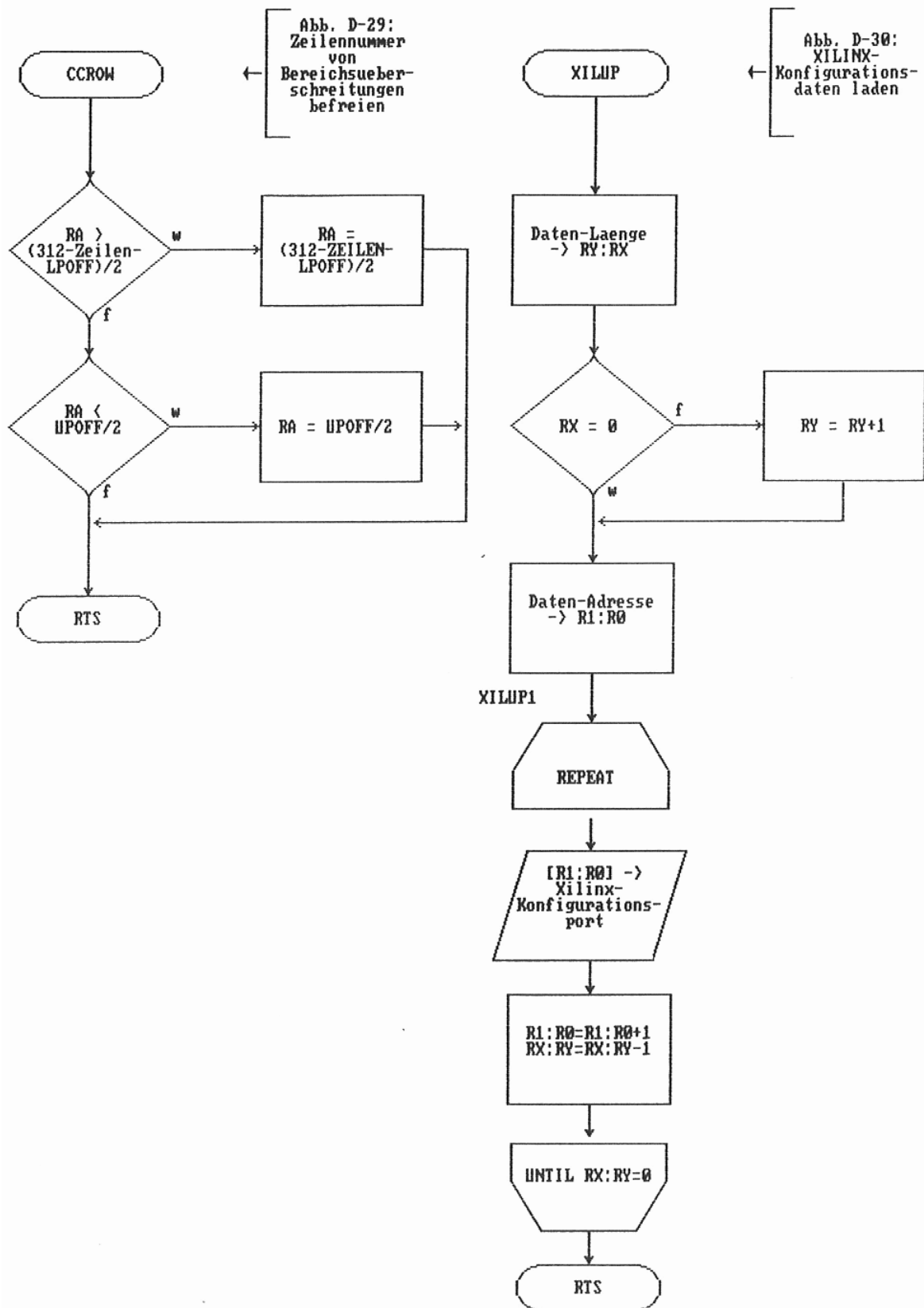


Abb. D-29: CCROW: Zeilennummer-Bereichsabfrage

Abb. D-30: XILUP: XILINX-Konfiguration laden

*** Datei UNTER.S ***

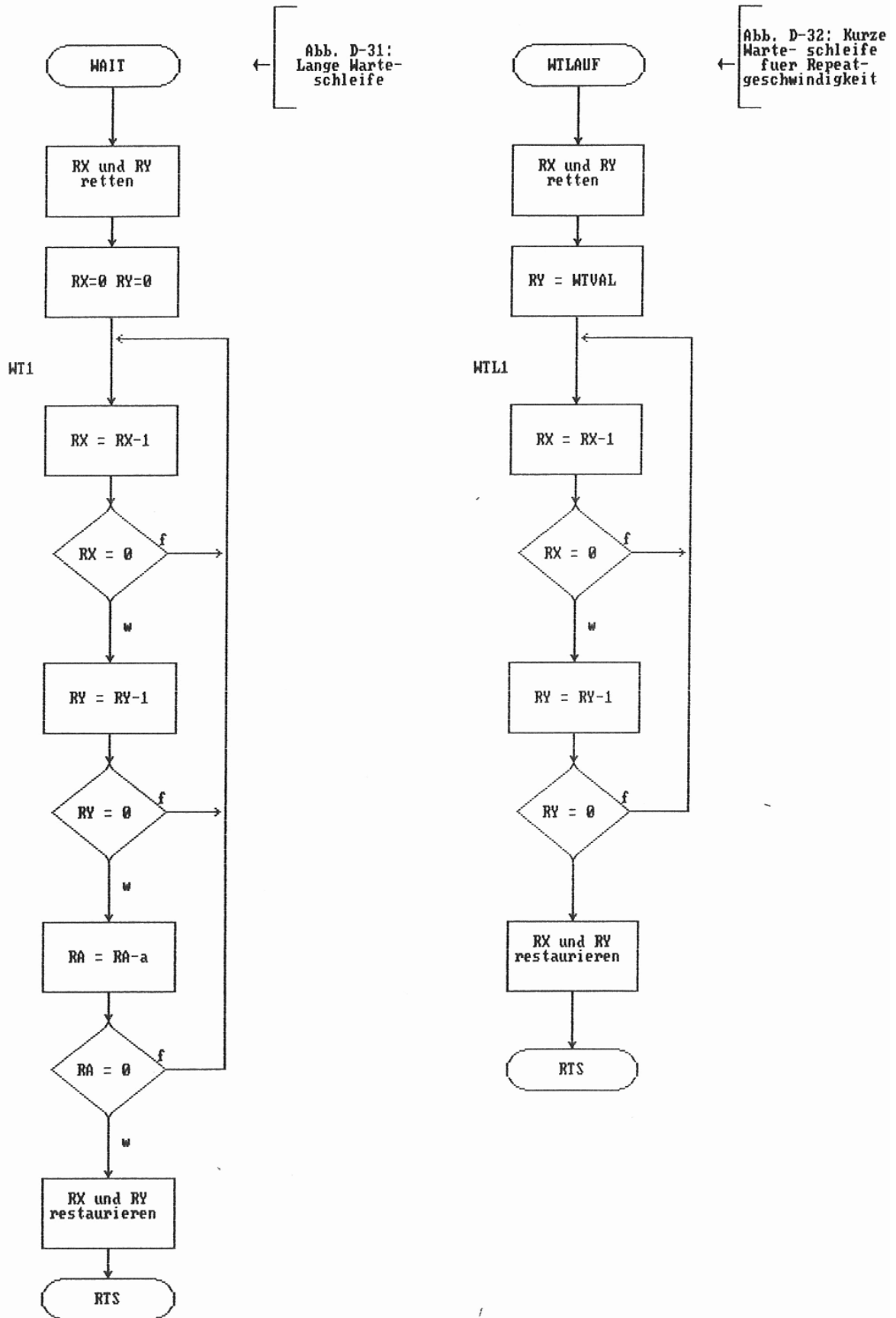


Abb. D-31: WAIT: Lange Warteschleife

Abb. D-32: WTLAUF: Kurze Warteschleife für Hauptprogramm

E. Programmlisting**E.1. Bibliothek (BIBL.S)**

```

; // dienstag, 20.februar 1990  16.17 uhr
;
; DATEI BIBL.S
; PROJEKT EDDI3

ADRE      MACRO

FALSE     EQU      0
TRUE      EQU      .NOT.FALSE

; PROGRAMM-OPTIONEN
GROSS     EQU      FALSE          VOLLBILDDARSTELLUNG
STEH      EQU      FALSE          STEHENDES BILD

; PROGRAMMPARAMETER
REPVAL    EQU      100            REPEAT-ZAEHLER-WERT (TASTATUR-WARTEZEIT)
WTVAL     EQU      50            LAUF-WARTESCHLEIFE (TASTATUR-REPEAT-RATE)

OFFSET    EQU      -5            KORREKTURWERT FUER WPCAS
PIXEL     EQU      80            ANZAHL PIXEL PRO BILD
ZEILEN    EQU      72            ANZAHL ZEILEN PRO BILD
BILDER    EQU      42            BILDER IM SPEICHER
UPOFF     EQU      34            OFFSET ZUM OBEREN BILDSCHIRM RAND
LPOFF     EQU      8            OFFSET ZUM UNTEREN BILDSCHIRM RAND

; BELEGUNG IO3PB
WPCAS     EQU      $01           PB0:  PRESET CAS
WP1       EQU      $02           PB1:  PRESET ZEILE
WPRAS     EQU      $04           PB2:  PRESET RAS
CPIX      EQU      $08           PB3:  CPU/PICTURE UMSCHALTUNG
CSCPU     EQU      $10           PB4:  CPU-ZUGRIFF AUF RAM
RAMOE     EQU      $20           PB5:  OE-SPEICHER
RAMWE     EQU      $40           PB6:  WE-SPEICHER
RECRQ     EQU      $80           PB7:  RECORD-REQUEST

; -----
; I/O-ADRESSEN
IO3        EQU      $2040        3. 6522 AUF TC12-CPU
IO3PB      EQU      IO3+0
IO3PA      EQU      IO3+1
IO3DB      EQU      IO3+2
IO3DA      EQU      IO3+3
IO3T1CL    EQU      IO3+4
IO3T1CH    EQU      IO3+5
IO3T1LL    EQU      IO3+6
IO3T1LH    EQU      IO3+7
IO3T2CL    EQU      IO3+8
IO3T2CH    EQU      IO3+9
IO3S       EQU      IO3+10
IO3A       EQU      IO3+11
IO3P       EQU      IO3+12
IO3I       EQU      IO3+13
IO3E       EQU      IO3+14
IO3O       EQU      IO3+15

; -----

```

; BILDSPEICHER-ADRESSEN

PMEM	EQU	\$8000	PMEM: PAGE-MEMORY
PMCF	EQU	PMEM+0	PMEM: XILINX-CONFIGURATION
PMRAS	EQU	PMEM+0	PMEM: RAS-SETZPULS
PMCAS	EQU	PMEM+1	PMEM: CAS-SETZIMPULS
PMROW1	EQU	PMEM+2	PMEM: ZEILEN-SETZPULS #1
PMCP1X	EQU	PMEM+4	PMEM: CPU-MODUL UMSCHALTEN
PMREC	EQU	PMEM+5	PMEM: REC-MODUS UMSCHALTEN
PMLUM	EQU	PMEM+8	PMEM: LUMINANZ
PMCRO	EQU	PMEM+9	PMEM: CHROMA
PMPORT	EQU	PMEM+11	PMEM: I/O-PORT
PMIRES	EQU	PMEM+12	PMEM: IRQ-RESET WIEDERGABE

ENDM ; ADRE

; *****

; BEFEHLS-MAKROS

; *****

VIERLSR MACRO

```

    LSR    A
    LSR    A
    LSR    A
    LSR    A
ENDM

```

VIERASL MACRO

```

    ASL    A
    ASL    A
    ASL    A
    ASL    A
ENDM

```

DREILSR MACRO

```

    LSR    A
    LSR    A
    LSR    A
ENDM

```

DREIASL MACRO

```

    ASL    A
    ASL    A
    ASL    A
ENDM

```

ZWEILSR MACRO

```

    LSR    A
    LSR    A
ENDM

```

ZWEIASL MACRO

```

    ASL    A
    ASL    A
ENDM

```

BNE_L MACRO

```

    N
    BEQ    BNE&INDX
    JMP     &N

```

BNE&INDX;

```

        ENDM

BEQ_L   MACRO   N
        BNE     BEQ&INDX
        JMP     &N
BEQ&INDX;
        ENDM

BCS_L   MACRO   N
        BCC     BCS&INDX
        JMP     &N
BCS&INDX;
        ENDM

BCC_L   MACRO   N
        BCS     BCC&INDX
        JMP     &N
BCC&INDX;
        ENDM

BVS_L   MACRO   N
        BVC     BVS&INDX
        JMP     &N
BVS&INDX;
        ENDM

BVC_L   MACRO   N
        BVS     BVC&INDX
        JMP     &N
BVC&INDX;
        ENDM

BMI_L   MACRO   N
        BPL     BMI&INDX
        JMP     &N
BMI&INDX;
        ENDM

BPL_L   MACRO   N
        BMI     BPL&INDX
        JMP     &N
BPL&INDX;
        ENDM

; *****
; GLOBALE ADRESSEN
; *****
GLOBAL  MACRO
;
;               START.S
        GLBL    R0,R1,R2,R3,R4,R5,R6,R7    UNIVERSALREGISTER
        GLBL    VMODE    VIDEOMODUS
;
;               IRQ.S
        GLBL    IRQA      EINSPRUNG MASKIERBARER INTERRUPT
        GLBL    NMIA      EINSPRUNG NICHT MASKIERBARER INTERRUPT
;
;               UNTER.S
        GLBL    BIN2BCD    KONVERTIERUNG BINAER -> BCD
        GLBL    INCCAS      CAS-ADRESSE ERHOEHEN
        GLBL    DECCAS      CAS-ADRESSE ERNIEDRIGEN
        GLBL    INCRAS      RAS-ADRESSE ERHOEHEN
        GLBL    DECRAS      RAS-ADRESSE ERNIEDRIGEN
        GLBL    ICCAS       CCAS ERHOEHEN

```

```

        GBL  DCCAS      CCAS ERNIEDRIGEN
        GBL  ICRAS      CRAS ERHOEHEN
        GBL  DCRAS      CRAS ERNIEDRIGEN
        GBL  ICRW1      ZEILE ERHOEHEN
        GBL  DCRW1      ZEILE ERNIEDRIGEN
        GBL  CCCAS      CAS-BEREICHSABFRAGE
        GBL  CCRAS      RAS-BEREICHSABFRAGE
        GBL  CCROW      ROW-BEREICHSABFRAGE
        GBL  WAIT       WARTEN
        GBL  WTLAUF      WARTESCHLEIFE FUER LAUF
        GBL  CHSET       ZEICHENSATZ-TABELLE
        GBL  XILUP       XILINX-KONFIGURATION LADEN
        GBL  CCAS        CAS-ZAEHLER (OFFSET)
        GBL  CRAS        RAS-ZAEHLER (BILDNUMMER)
        GBL  CRW1        ZEILENZAEHLER
;
        XILINX.S
        GBL  XILINXS     STARTADRESSE XILINX-DATEN
        GBL  XILINXL     LAENGE XILINX-DATEN
;
        FREMD.S
        GBL  ANZR        ANZEIGEREGISTER DARSTELLEN
        GBL  TAAU        TASTATUR SCANNEN
        GBL  ANRE        ANZEIGEREGISTER
;
        TAST.S
        GBL  ATINIT      ANZEIGE UND TASTATUR INITIALISIEREN
        GBL  CONIN       ZEICHEN VON TASTATUR EINLESEN
        GBL  TASLOW      LOW-LEVEL-TASTATURFUNKTIONEN AUSFUEHREN
        GBL  ANZNEU      ANZEIGE AUFRISCHEN
        GBL  IBUFB       SCRATCH-EINGABEREGISTER
        GBL  ANZLED1     STATUS-LED'S

ENDM ; GLOBAL

END

```

E.2. Initialisierung und Hauptprogramm (START.S)

```

; // dienstag, 27.februar 1990  18.34 uhr
;
; DATEI START.S
; PROJEKT EDDI3

        ADRE
        GLOBAL

        RSEG    START

STTA    ;                                RESET-EINSTIEG
        SEI
        LDX     #$FF                    STACK SETZEN
        TXS
        CLD                                DEZIMAL-MODUS AUS

        LDA     DCMP
        LDX     #0                      RAM LOESCHEN
STTC    ;
        STZ     $0,X
        STZ     $100,X
        STZ     $200,X
        STZ     $300,X
        STZ     $400,X
        STZ     $500,X
        STZ     $600,X
        STZ     $700,X
        DEX
        BNE     STTC
        STA     DCMP
        INC     DCMP

        JSR     ATINIT                  ANZEIGE U. TASTATUR INITIALISIEREN
        JSR     XILUP                   XILINX-KONFIGURATION LADEN
        LDA     PMCPIX                  CPIX LOESCHEN
        LDA     PMREC                   RECRQ LOESCHEN
        LDA     #UPOFF/2+50
        STA     CRW1                    ZEILE RUECKSETZEN
        STA     PMROW1

        LDA     #10000000B
        STA     PMPORT                  WIEDERGABE-IRQ FREIGEBEN

        JSR     BFUELL                  BILDSPEICHER FUELLEN
        LDA     PMCPIX                  CPU-MODUS AUS

        LDA     #OFFSET
        STA     CCAS                    BILD-OFFSET RUECKSETZEN
        JSR     ANZNEU                  ANZEIGE AUFRISCHEN

        CLI                                IRQ FREIGEBEN
; HAUPT-PROGRAMM-SCHLEIFE #####
LAUF    ;
        JSR     TASLOW                  LOW-LEVEL-TASTATURFUNKTIONEN AUSFUEHREN
        CMP     #36                     REC ?
        BNE     LAUF1
; RECORD-FUNKTION
        LDA     #01000000B
        TRB     ANZLED1

```

```

        JSR      ANZNEU      REC-LED EINSCHALTEN
        STZ      PMCAS      AUFNAHME VORBEREITEN
        LDA      IBUF8
        STA      PMRAS      BILD-# ANWAEHLEN
        LDA      #20/2
        STA      PMROW1     ZEILE SETZEN
        LDA      #01000000B
        STA      PMPORT     AUFNAHME-IRQ FREIGEBEN
        STA      PMREC      REC-MODUS EIN
        LDA      #1
        STA      VMODE      VIDEO-MODUS UMSCHALTEN
LAUF01  ;
        LDA      VMODE      AUFNAHME BEENDET ?
        BNE      LAUF01     NEIN
        LDA      #10000000B
        STA      PMPORT     WIEDERGABE-IRQ FREIGEBEN
        LDA      PMREC      AUFNAHME-MODUS AUS
        LDA      #01000000B
        TSB      ANZLED1
        JSR      ANZNEU      REC-LED AUSSCHALTEN
        BRA      LAUF
LAUF1   ;
        BRA      LAUF

; #####
; BILDSPEICHER FUELLEN
BFUELL  ;
        STA      PMCPIX      CPIX EINSCHALTEN

        STZ      DPUT      BILD-NUMMER

STTX1   ;
; ADRESSZAEHLER INITIALISIEREN
        STZ      PMCAS
        LDA      DPUT
        STA      PMRAS      BILD-# ANLEGEN
        STZ      PMCAS

        LDA      #ZEILEN/3
        STA      R1
STTX2   ;
        LDA      #3
        STA      R2
STTX3   ;
; ZEILE AUFBAUEN
        SEC
        LDA      #ZEILEN+3
        SBC      R1
        SBC      R1
        SBC      R1
        SBC      R2      ZEILENNUMMER BERECHNEN
        JSR      MKZEIL    ZEILE BERECHNEN

; ZEILE AUSGEBEN
        LDX      #0      PIXELZAEHLER
STTX4   ;
        LDA      ZEILE,X
        STA      PMCRO      CHROMA EINSCHREIBEN
        EOR      #$FF
        STA      PMLUM      LUMINANZ EINSCHREIBEN
        INX

```



```

        CPX      #PIXEL
        BNE      STTX4          PIXELZAEHLER

        DEC      R2
        BNE      STTX3          ZEILENZAEHLER L

        LDX      #256-(3*PIXEL)  PIXELOFFSET
STTX5    ;
        STZ      PMCRO
        STZ      PMLUM          DUMMY-DATEN EINSCHREIBEN
        DEX
        BNE      STTX5

        DEC      R1
        BNE      STTX2          ZEILENZAEHLER H
        INC      DPUT
        LDA      DPUT
        CMP      #BILDER
        BNE      STTX1          BILDNUMMER

        RTS                      ENDE BFUELL

; EINBLENDZEILE AUFBAUEN #####
MKZEIL  ;
XOFF    EQU      10             HORIZONTALE POSITION DER EINBLENDUNG
YOFF    EQU      1             DTO. VERTIKALE POSITION
        CMP      #0
        BEQ      MKZC          LOESCHEN
        CMP      #YOFF+8
        BEQ      MKZC          LOESCHEN
        CMP      #YOFF
        BCC      MKZE          UEBER
        CMP      #YOFF+8
        BCS      MKZE          UNTER
MKI      ;
        SEC
        SBC      #YOFF
        STA      R4            ZEILENOFFSET

; 1.ZIFFER
        LDA      #L(ZEILE+XOFF+10)
        STA      MKCHP
        LDA      #H(ZEILE+XOFF+10)
        STA      MKCHP+1
        LDA      DPUT
        JSR      BIN2BCD
        VIERLSR
        JSR      MKCH

; 2.ZIFFER
        LDA      #L(ZEILE+XOFF+19)
        STA      MKCHP
        LDA      #H(ZEILE+XOFF+19)
        STA      MKCHP+1
        LDA      DPUT
        JSR      BIN2BCD
        AND      #$F
        JSR      MKCH

; 3.ZIFFER
        LDA      #L(ZEILE+XOFF+30)
        STA      MKCHP
        LDA      #H(ZEILE+XOFF+30)
        STA      MKCHP+1

```

```

        LDA      DCMP
        JSR      BIN2BCD
        VIERLSR
        JSR      MKCH
; 4.ZIFFER
        LDA      #L(ZEILE+XOFF+39)
        STA      MKCHP
        LDA      #H(ZEILE+XOFF+39)
        STA      MKCHP+1
        LDA      DCMP
        JSR      BIN2BCD
        AND      #$F
        JSR      MKCH

        BRA      MKZE

; ZEILE MIT ZEICHEN AUFBAUEN #####
MKZC      ;
          LDX      #0
MKZC1     ;
          LDA      DPUT
          ZWEIASL
        IF STEH
          LDA      #0
        ENDIF
          STA      ZEILE,X HINTERGRUND-HELLIGKEIT
          INX
          CPX      #PIXEL
          BNE      MKZC1
          BRA      MKZE

MKZE      ;
          RTS

; ZEICHEN AUFBAUEN #####
MKCH      ;
          DREIASL      * 8
          CLC
          ADC      R4
          TAX
          LDA      CHSET,X PIXELMUSTER
          STA      R5
          LDY      #8
MKCH1     ;
          LDA      #0
          ROR      R5
          SBC      #0
        IF .NOT.STEH
;          EOR      DPUT
        ENDIF
          STA      (MKCHP),Y
          DEY
          BNE      MKCH1
          RTS

; SPEICHERBELEGUNG #####
RSEG      STARTZ      ZEROPAGE
R0      DS      1
R1      DS      1
R2      DS      1
R3      DS      1

```

```

R4      DS      1
R5      DS      1
R6      DS      1
R7      DS      1      8 UNIVERSALREGISTER
MKCHP   DS      2      2 BYTE PTR FUER MKZEIL
          RSEG    STARTD  RAM
ZEILE   DS      80      80 BYTE ZEILENSPEICHER
DPUT    DS      1
DGET    DS      1
DCMP    DS      1
VMODE   DS      1      VIDEO-MODUS
;          0 = WIEDERGABE
;          1 = AUFNAHME

; INTERRUPT-VEKTOREN #####
          RSEG    VEKT
          DW      NMIA
          DW      STTA
          DW      IRQA

          END

```

E.3. Interrupt-Behandlung (IRQ.S)

```

; // freitag, 23.februar 1990  19.51 uhr
;
; DATEI IRQ.S
; PROJEKT EDDI3

        ADRE
        GLOBAL

        RSEG    IRQ

IRQA    ;
        PHA
        PHX
        PHY
        LDA     PMPORT
        AND     #$80
        BEQ     IRQB
; WIEDERGABE - V-PULS - INTERRUPT #####
IRQA1   ;
        STA     PMIRES          IRQ RUECKSETZEN
        LDA     CCAS
        STA     PMCAS          CAS SETZEN
        LDA     CRAS
        STA     PMRAS          RAS SETZEN
        LDA     CRW1
        STA     PMROW1         ROW SETZEN
        BRA     IRQE           ENDE WIEDERGABE - V-PULS - INTERRUPT

IRQB    ;
        LDA     PMPORT
        AND     #$40
        BEQ     IRQC

; AUFNAHME - V-PULS - INTERRUPT #####
IRQA2   ;
        STA     PMIRES          IRQ RUECKSETZEN
        STZ     VMODE          AUFNAHME-MODUS BEENDEN
        BRA     IRQE           ENDE AUFNAHME - V-PULS - INTERRUPT

IRQC    ;
; ##### UNBEKANNTER INTERRUPT #####
        LDA     #$80
        STA     IO3I           IO3-INTERRUPTS LOESCHEN

IRQE    ;
        PLY
        PLX
        PLA
        RTI

NMIA    ;
        RTI

; #####

        RSEG    IRQZ
        RSEG    IRQD

        END

```

E.4. Tastatur- und Anzeigesteuerung (TAST.S)

```

; // dienstag, 20.februar 1990  16.23 uhr
;
; DATEI TAST.S
; PROJEKT EDDI3

        ADRE
        GLOBAL

        RSEG    TAST

; ANZEIGE AUFRISCHEN #####
ANZNEU ;
        LDX      #0
        LDA      IBUF'B          EINGABEBUFFER
        JSR      ANZNU           -> 'FRAMES'
        LDX      #2
        LDA      CRW1           EINBLENDZEILE
        CLC
        ADC      #-UPOFF/2
        JSR      ANZNU           -> 'SECONDS'
        LDX      #4
        LDA      CCAS           BILDVERSATZ
        CLC
        ADC      #-OFFSET
        JSR      ANZNU           -> 'MINUTES'
        LDX      #6
        LDA      CRAS           BILDNUMMER
        JSR      ANZNU           -> 'HOURS'
        LDA      #$FF
        LDA      ANZLED1 STATUS-LED'S LADEN
        STA      ANRE+16  -> LED'S
        JSR      ANZR           ANZEIGEREGISTER DARSTELLEN
        RTS

ANZNU ;
        JSR      BIN2BCD ZAHL IN BCD KONVERTIEREN
        TAY
        AND      #$0F           LOW-BYTE MASKIEREN
        ORA      #$10           ZUGEHOERIGE LED LOESCHEN
        STA      ANRE,X         SPEICHERN INS ANZEIGEREGISTER
        TYA
        VIERLSR           HIGH-BYTE MASKIEREN
        ORA      #$10           ZUGEHOERIGE LED LOESCHEN
        STA      ANRE+1,X SPEICHERN INS ANZEIGEREGISTER
        RTS

; LOW-LEVEL-TASTATURFUNKTIONEN AUSFUEHREN #####
TASLOW ;
        JSR      CONIN           TASTATURWERT EINLESEN
        STA      TASBUF
        CMP      #-1            TASTE GEDRUECKT ?
        BEQ_L    TASLE          NEIN: FERTIG
        CMP      #10            TASTE 0..9 ?
        BCS      TASL1          NEIN
        TAX
        LDA      IBUFT          LOW-BYTE
        STA      IBUFT+1        SPEICHERN IN HIGH-BYTE
        STX      IBUFT          NEUES LOW-BYTE
;                               UMWANDELN IN BINAERZAHL

```

	LDA	IBUFT+1	HIGH-BYTE
	ASL	A	* 2
	STA	IBUFB	ZWISCHENERGEBNIS
	ZWEIASL		* 8
	ADC	IBUFB	* 10
	ADC	IBUFT	ENDERGEBNIS
	STA	IBUFB	ABSPEICHERN
	BRA	TASLA	
TASL1	;		
	CMP	#24	CRAS + ?
	BNE	TASL2	
	JSR	ICRAS	CRAS ERHOEHEN
	BRA	TASLA	
TASL2	;		
	CMP	#23	CRAS SET ?
	BNE	TASL3	
	LDA	IBUFB	
	JSR	CCRAS	
	STA	CRAS	CRAS SETZEN
	BRA	TASLA	
TASL3	;		
	CMP	#22	CRAS - ?
	BNE	TASL4	
	JSR	DCRAS	CRAS ERNIEDRIGEN
	BRA	TASLA	
TASL4	;		
	CMP	#27	CCAS + ?
	BNE	TASL5	
	JSR	ICCAS	CCAS ERHOEHEN
	BRA	TASLA	
TASL5	;		
	CMP	#26	CCAS SET ?
	BNE	TASL6	
	LDA	IBUFB	
	CLC		
	ADC	#OFFSET	
	JSR	CCCAS	
	STA	CCAS	CCAS SETZEN
	BRA	TASLA	
TASL6	;		
	CMP	#25	CCAS - ?
	BNE	TASL7	
	JSR	DCCAS	CCAS ERNIEDRIGEN
	BRA	TASLA	
TASL7	;		
	CMP	#30	CRW1 + ?
	BNE	TASL8	
	JSR	ICRW1	CRW1 ERHOEHEN
	BRA	TASLA	
TASL8	;		
	CMP	#29	CRW1 SET ?
	BNE	TASL9	
	LDA	IBUFB	
	CLC		
	ADC	#UPOFF/2	
	JSR	CCROW	
	STA	CRW1	CRW1 SETZEN
	BRA	TASLA	
TASL9	;		
	CMP	#28	CCW1 - ?
	BNE	TASL10	

```

        JSR      DCRW1      CRW1 ERNIEDRIGEN
        BRA      TASIA
TASL10  ;
        BRA      TASLE      KEINE LOW-LEVEL FUNKTION
TASIA   ;
        JSR      ANZNEU      ANZEIGE AUFFRISCHEN
        JSR      WTLAUF      WARTEN
TASLE   ;
        LDA      TASBUF      TASTATURCODE RESTAURIEREN
        RTS          ENDE TASLOW

; ZEICHEN VON TASTATUR EINLESEN #####
CONIN   ;
        PHX
        PHY
        JSR      TASIA
        BEQ      CI1          TASTE GEDRUECKT!
        LDA      #-1          RUCKGABEWERT FUER 'KEINE TASTE GEDRUECKT'
        BRA      CIE
CI1      ;
        LDX      #0
CI2      ;
        LDY      CITAB,X TABELLENENDE ERREICHT ?
        BEQ      CI5          JA
        CMP      CITAB,X TASTENWERT MIT TABELLE
        BNE      CI3          UNGLEICH
        LDA      CITAB+1,X    NEUEN TASTENCODE LADEN
        BRA      CIE          FERTIG
CI3      ;
        INX
        INX
        BRA      CI2
CI5      ;
        LDA      #-1          TASTE NICHT IN TABELLE GEFUNDEN
CIE      ;
        PLY
        PLX
        RTS          ENDE CONIN

; TASTENDRUCK ERMITTELN
TASIA   ;
        JSR      TAAU          TASTATUR ABSCANNEN
        BEQ      TIA5          TASTE GEHALTEN!
        LDA      #REPVAL
        STA      REPCNT
        LDA      #0
        TIA5      ;
        CMP      TICMP          GLEICHE TASTE GEDRUECKT ?
        BEQ      TIA7          JA!
        STA      TICMP          NEUEN VERGLEICHSWERT SPEICHERN
        CMP      #0              TASTE GEDRUECKT ?
        BEQ      TIA8          NEIN
        LDY      #0              Z-FLAG SETZEN: TASTENDRUCK!
        BRA      TIAE          FERTIG
TIA7    ;
        TAX
        LDA      REPCNT          TASTENCODE MERKEN
        BNE      TIA75          REPEAT-ZAEHLER NOCH NICHT ABGELAUFEN
        TXA
        LDY      #0              TASTENCODE WIEDERHOLEN
        BRA      TIAE          Z-FLAG SETZEN: TASTENDRUCK!
        FERTIG

```

```

TIA75      ;
           DEC      REPCNT      REPEAT-ZAEHLER WEITERZAEHLEN
           LDY      #-1         Z-FLAG LOESCHEN: KEIN TASTENDRUCK
           BRA      TIAE        FERTIG

TIA8      ;
           LDY      #-1         Z-FLAG LOESCHEN: KEIN TASTENDRUCK

TIAE      ;
           RTS                  ENDE TASIA

CITAB     ;
           DB      $7B,0        0
           DB      $5B,1        1
           DB      $3B,2        2
           DB      $1B,3        3
           DB      $5D,4        4
           DB      $3D,5        5
           DB      $1D,6        6
           DB      $5E,7        7
           DB      $3E,8        8
           DB      $1E,9        9
           DB      $7E,20       +
           DB      $7D,21       -
           DB      $AF,22       F1
           DB      $8F,23       F2
           DB      $6F,24       F3
           DB      $4F,25       F4
           DB      $2F,26       CONT.JAM
           DB      $0F,27       JAM
           DB      $B7,28       VIDEO
           DB      $97,29       SET
           DB      $77,30       USER
           DB      $57,31       TIME
           DB      $37,32       30
           DB      $17,33       24
           DB      $9E,34       25
           DB      $9D,35       PLAY
           DB      $9B,36       REC
           DB      0            TABELLENENDE CITAB

           RSEG      TASTZ
           RSEG      TASTD

TICMP     DS      1            VERGLEICHSWERT FUER TASIA
REPCNT    DS      1            REPEAT-ZAEHLER FUER TASIA
IBUFT     DS      2            EINGABEBUFFER ALS TASTENWERT
IBUFB     DS      1            DTO. ALS BINAERWERT
TASBUF    DS      1            BUFFER FUER TASLOW
ANZLED1   DS      1            ANZEIGE-LED'S

END

```


E.5. Unterprogramme (UNTER.S)

```

; // freitag, 23.februar 1990  19.50 uhr
;
; DATEI UNTER.S
; PROJEKT EDDI3

        ADRE
        GLOBAL

        RSEG    UNTER

; BINAER -> BCD #####
BIN2BCD ;
        PHX
        LDX     #-1
B2D1    ;
        INX
        SEC
        SBC     #10
        BCS     B2D1
        CLC
        ADC     #10
        STA     B2DSTO
        TXA
        VIERASL      * 16
        ADC     B2DSTO
        PLX
        RTS                                ENDE BIN2BCD

; CAS-ADRESSE ERHOEHEN #####
INCCAS  ;
        INC     A
        CMP     #PIXEL+OFFSET+1
        BMI     INCCE
        LDA     #OFFSET ZAEHLER STARTEN
INCCE   ;
        RTS

; CAS-ADRESSE ERNIEDRIGEN #####
DECCAS  ;
        DEC     A
        CMP     #OFFSET
        BPL     DECCE
        LDA     #PIXEL+OFFSET
DECCE   ;
        RTS

; RAS-ADRESSE ERHOEHEN #####
INCRAS  ;
        INC     A
        CMP     #BILDER-4
        BCC     INCRE
        LDA     #0
INCRE   ;
        RTS

; RAS-ADRESSE ERNIEDRIGEN #####
DECRAS  ;
        DEC     A
        CMP     #BILDER-4

```

```

        BCC      DECRET
        LDA      #BILDER-5
DECRET  ;
        RTS

; ZEILE ERHOEHEN #####
INCROW  ;
        INC      A
        CMP      # (312-ZEILEN-LPOFF) /2+1
        BCC      INCWE
        LDA      #UPOFF/2
INCWE   ;
        RTS

; ZEILE ERNIEDRIGEN #####
DECROW  ;
        DEC      A
        CMP      #UPOFF/2
        BCS      DECWE
        LDA      # (312-ZEILEN-LPOFF) /2
DECWE   ;
        RTS

; CCAS-RCAS ERHOEHEN #####
ICCAS   ;
        LDA      CCAS          CAS-ZAEHLER LADEN
        JSR      INCCAS          ERHOEHEN
        STA      CCAS          UND ABSPEICHERN
        CMP      #OFFSET CAS-GRENZE ERREICHT ?
        BNE      ICRE          NEIN
        LDA      CRAS          RAS-ZAEHLER LADEN
        JSR      INCRAS          ERHOEHEN
        STA      CRAS          UND ABSPEICHERN
ICRE    ;
        RTS

; CCAS/CRAS ERNIEDRIGEN #####
DCCAS   ;
        LDA      CCAS          CAS-ZAEHLER LADEN
        JSR      DECCAS          ERNIEDRIGEN
        STA      CCAS          UND ABSPEICHERN
        CMP      #PIXEL+OFFSET CAS-GRENZE ERREICHT ?
        BNE      DCRE          NEIN
        LDA      CRAS          RAS-ZAEHLER LADEN
        JSR      DECRAS          ERNIEDRIGEN
        STA      CRAS          UND ABSPEICHERN
DCRE    ;
        RTS

; CRAS ERHOEHEN #####
ICRAS   ;
        LDA      CRAS          RAS-ZAEHLER LADEN
        JSR      INCRAS          ERHOEHEN
        STA      CRAS          UND ABSPEICHERN
        RTS

; CRAS ERNIEDRIGEN #####
DCRAS   ;
        LDA      CRAS          RAS-ZAEHLER LADEN
        JSR      DECRAS          ERNIEDRIGEN
        STA      CRAS          UND ABSPEICHERN

```

```

        RTS

; CRW1 ERHOEHEN #####
ICRW1   ;
        LDA     CRW1          ROW-ZAEHLER LADEN
        JSR     INCROW        ERHOEHEN
        STA     CRW1          UND ABSPEICHERN
        RTS

; CRW1 ERNIEDRIGEN #####
DCRW1   ;
        LDA     CRW1          ROW-ZAEHLER LADEN
        JSR     DECROW        ERNIEDRIGEN
        STA     CRW1          UND ABSPEICHERN
        RTS

; CAS-BEREICHSABFRAGE #####
CCCAS   ;
        CMP     #PIXEL+OFFSET+1 OBERE GRENZE UEBERSCHRITTEN ?
        BMI     CCC1
        LDA     #PIXEL+OFFSET
        BRA     CCCE
CCC1    ;
        CMP     #OFFSET UNTERE GRENZE UEBERSCHRITTEN ?
        BPL     CCCE
        LDA     #OFFSET
CCCE    ;
        RTS

; RAS-BEREICHSABFRAGE #####
CCRAS   ;
        CMP     #BILDER-4      OBERE GRENZE UEBERSCHRITTEN ?
        BCC     CCRE
        LDA     #BILDER-5
CCRE    ;
        RTS

; ROW-BEREICHSABFRAGE #####
CCROW   ;
        CMP     #(312-ZEILEN-LPOFF)/2+1 OBERE GRENZE UEBERSCHRITTEN ?
        BCC     CCW1
        LDA     #(312-ZEILEN-LPOFF)/2
        BRA     CCWE
CCW1    ;
        CMP     #UPOFF/2 UNTERE GRENZE UEBERSCHRITTEN ?
        BCS     CCWE
        LDA     #UPOFF/2
CCWE    ;
        RTS

        RTS

; UPLOAD XILINX-KONFIGURATION #####
XILUP   ;
        LDY     #H(XILINXL)    LAENGE XILINX-DATEN (MSB)
        LDX     #L(XILINXL)    " " " (LSB)
        BEQ     XILUP0
        INY     BYTEZAEHLER KORRIGIEREN
XILUP0  ;
        LDA     #L(XILINXS)    ADRESSE DER XILINX-DATEN (LSB)
        STA     R0

```

```

        LDA    #H(XILINXS)          " " " " (MSB)
        STA    R1
XILUP1  ;
        LDA    (R0)                 XILINX-BYTE LADEN
        STA    PMCF                 UND INS KONFIGURATIONSREGISTER SPEICHERN
        INC    R0                   ADRESSE ERHOEHEN (LSB)
        BNE    XILUP2
        INC    R1                   ADRESSE ERHOEHEN (MSB)
XILUP2  ;
        PHA
        PLA
        PHA
        PLA

        DEX                         BYTE-ZAEHLER (LSB)
        BNE    XILUP1
        DEY                         BYTE-ZAEHLER (MSB)
        BNE    XILUP1
        RTS

; WARTESCHLEIFE (LANG) #####
WAIT    ;
        PHX
        PHY
        LDX    #0
        LDY    #0
WT1     ;
        DEX
        BNE    WT1
        DEY
        BNE    WT1
        DEC    A
        BNE    WT1
        PLY
        PLX
        RTS

; WARTESCHLEIFE FUER LAUF-ROUTINE (HAUPTPROGRAMM) #####
WTLAUF  ;
        PHX
        PHY
        LDX    #0
        LDY    #WTVL
WTL1    ;
        DEX
        BNE    WTL1
        DEY
        BNE    WTL1
        PLY
        PLX
        RTS

; ZEICHENSATZ ZUR DARSTELLUNG IM BILDSPEICHER #####
CHSET   ;
        DB      00111000B          0
        DB      01000100B
        DB      01001100B
        DB      01010100B
        DB      01100100B
        DB      01000100B
        DB      00111000B

```

DB	00000000B	
DB	00010000B	1
DB	00110000B	
DB	01010000B	
DB	00010000B	
DB	00010000B	
DB	00010000B	
DB	01111100B	
DB	00000000B	
DB	00111000B	2
DB	01000100B	
DB	00000100B	
DB	00001000B	
DB	00010000B	
DB	00100000B	
DB	01111100B	
DB	00000000B	
DB	00111000B	3
DB	01000100B	
DB	00000100B	
DB	00011000B	
DB	00000100B	
DB	01000100B	
DB	00111000B	
DB	00000000B	
DB	00001000B	4
DB	00011000B	
DB	00101000B	
DB	01001000B	
DB	01111100B	
DB	00001000B	
DB	00011100B	
DB	00000000B	
DB	01111100B	5
DB	01000000B	
DB	01111000B	
DB	00000100B	
DB	00000100B	
DB	01000100B	
DB	00111000B	
DB	00000000B	
DB	00111000B	6
DB	01000100B	
DB	01000000B	
DB	01111000B	
DB	01000100B	
DB	01000100B	
DB	00111000B	
DB	00000000B	
DB	01111100B	7
DB	01000100B	
DB	00000100B	
DB	00001000B	
DB	00010000B	

DB	00100000B	
DB	00100000B	
DB	00000000B	
DB	00111000B	8
DB	01000100B	
DB	01000100B	
DB	00111000B	
DB	01000100B	
DB	01000100B	
DB	00111000B	
DB	00000000B	
DB	00111000B	9
DB	01000100B	
DB	01000100B	
DB	00111100B	
DB	00000100B	
DB	01000100B	
DB	00111000B	
DB	00000000B	
DB	00010000B	A
DB	00101000B	
DB	01000100B	
DB	01000100B	
DB	01111100B	
DB	01000100B	
DB	01000100B	
DB	00000000B	
DB	01111000B	B
DB	01000100B	
DB	01000100B	
DB	01111000B	
DB	01000100B	
DB	01000100B	
DB	01111000B	
DB	00000000B	
DB	00111000B	C
DB	01000100B	
DB	01000000B	
DB	01000000B	
DB	01000000B	
DB	01000100B	
DB	00111000B	
DB	00000000B	
DB	01111000B	D
DB	01000100B	
DB	01000100B	
DB	01000100B	
DB	01000100B	
DB	01000100B	
DB	01111000B	
DB	00000000B	
DB	01111100B	E
DB	01000000B	
DB	01000000B	

```

DB      01111000B
DB      01000000B
DB      01000000B
DB      01111100B
DB      00000000B

DB      01111100B      F
DB      01000000B
DB      01000000B
DB      01111000B
DB      01000000B
DB      01000000B
DB      01000000B
DB      00000000B

RSEG    UNTERZ          ZERO-PAGE
RSEG    UNTERD          RAM-BEREICH
B2DSTO  DS      1      ZWISCHENSPEICHER FUER BIN2BCD
CCAS    DS      1      CAS-ZAEHLER, 7 BIT + VORZEICHEN
CRW1    DS      1      ZEILENZAEHLER, 8 BIT
CRAS    DS      1      RAS-ZAEHLER, 8 BIT

END

```

E.6. Low-Level Routinen (FREMD.S)

```

; // dienstag, 27.februar 1990  18.37 uhr
;
; DATEI FREMD.S
; PROJEKT EDDI3
; HARDWARETREIBER FUER TC12-ANZEIGE/TASTATUR
; (c) ALPERMANN & VELTE GmbH

        RSEG      FREMD

        ADRE
        GLOBAL

; ANZEIGE UND TASTATUR INITIALISIEREN #####
ATINIT  ;
;
;                                I/O-BAUSTEIN INITIALISIEREN
        LDA      #01000000B
        STA      IO3A
        STZ      IO3P
        LDA      #01111111B
        STA      IO3E
        STA      IO3I
        STZ      IO3T1CL
        LDA      #80H          T2 ANZEIGETAKT
        STA      IO3T2CH

        STZ      IO3PA
        LDA      #11100000B
        STA      IO3DA
        LDA      #01100000B
        STA      IO3PB
        LDA      #7FH
        STA      IO3DB

        LDX      #16          16-STELLIGE ANZEIGE
        LDA      #$10        LEUCHTDIODE AUS; ANZEIGE='0'
AT11    ;
        STA      ANRE-1,X
        DEX
        BNE      AT11
        LDA      #$FF
        STA      ANRE+16 STATUS-DIODEN AUS
        JSR      ANZR          ANZEIGE AUFRISCHEN
        LDA      #$FF
        STA      ANZLED1 LED'S LOESCHEN
        RTS              ENDE ATINIT

; ANZEIGE-STEUERUNG #####
; INHALT VON ANRE IN ANZEIGE SCHREIBEN
ANZR    ;

ANZRT1  LDA      #20H          CS
        TSB      IO3PB
        LDY      #40H
        STY      IO3PA
        LDY      #30H
        STY      IO3PB
        TRB      IO3PB
        TSB      IO3PB
        STA      IO3PB

```



```

        STZ      IO3PA
        LDX      #0
ANZR2   LDY      ANRE,X          WERTE
        STY      IO3PB
        TSB      IO3PB
        INX
        CPX      #8
        BNE      ANZR2

ANZRT2  STA      IO3PA          MODE,KLEINE ANZEIGE
        LDY      #30H
        STY      IO3PB
        TRB      IO3PB
        TSB      IO3PB
        STA      IO3PB
        STZ      IO3PA
        LDX      #8
ANZR3   LDY      ANRE,X
        STY      IO3PB
        TSB      IO3PB
        INX
        CPX      #16
        BNE      ANZR3
        LDA      ANRE+16 STATUS
        STA      IO3PB
        LDA      #80H
        TSB      IO3PA
        TRB      IO3PA
        RTS

; TASTATUR ABFRAGEN #####
; ERGEBNIS IM AKKU
; Z-FLAG GESETZT = TASTE WAR GEDRUECKT
TAAU    ;
        JSR      TASI
        BEQ      TA10          TASTE ?
        STZ      TSTA+1        VERGLEICHSWERT=0
        RTS

TA10    STA      TSTA          TASTENWERT SICHERN
        JSR      ENTP
        CMP      #0FFH        TASTE O.K. ?
        BNE      TA20
        RTS

TA20    ;
        LDA      TSTA
        CMP      TSTA+1        TASTE GEHALTEN ?
        BNE      TA21
        RTS

TA21    STA      TSTA+1

TASI    LDX      #01
        STX      IO3PB        ZEILE
        LDA      IO3PA
        AND      #1FH
        CMP      #1FH
        BNE      TASI1        SPALTE ?
        LDX      #02
        STX      IO3PB

```

	LDA	IO3PA	
	AND	#1FH	
	ORA	#20H	ZEILEN-INDEX
	CMP	#3FH	
	BNE	TASI1	
	LDX	#04	
	STX	IO3PB	
	LDA	IO3PA	
	AND	#1FH	
	ORA	#40H	
	CMP	#5FH	
	BNE	TASI1	
	LDX	#08	
	STX	IO3PB	
	LDA	IO3PA	
	AND	#1FH	
	ORA	#60H	
	CMP	#7FH	
	BNE	TASI1	
	LDX	#10H	
	STX	IO3PB	
	LDA	IO3PA	
	AND	#1FH	
	ORA	#80H	
	CMP	#9FH	
	BNE	TASI1	
	LDX	#20H	
	STX	IO3PB	
	LDA	IO3PA	
	AND	#1FH	
	ORA	#0A0H	
	CMP	#0BFH	
	BNE	TASI1	
	LDY	#0FFH	KEINE TASTE
	RTS		
TASI1	LDY	#00	
	STY	IO3PB	
	RTS		
;ENTPRELLEN			
ENTP	;		
	LDA	#0FFH	
	STA	TSTA+2	
ENTP1	JSR	TASI	
	CMP	TSTA	GLEICHE TASTE ?
	BEQ	ENTP2	
	LDA	#0FFH	
	STA	TSTA+1	
	RTS		
ENTP2	DEC	TSTA+2	
	BNE	ENTP1	
	LDA	#00	
	RTS		
	RSEG	FREMDZ	
	RSEG	FREMDD	
ANRE	DS	17	ANZEIGENSPEICHER
TSTA	DS	3	TASIATUR-FLAGS

END

E.7. Referenztablelle

---- LINKER V2.6k ----- DR. KROHN & STILLER ----- 1 --

memory allocation for LINK.R

file	START.R	rseg	STARTZ	length	000A	at	0000	to	0009
	IRQ.R		IRQZ		0000		000A		
	UNTER.R		UNTERZ		0000		000A		
	TAST.R		TASTZ		0000		000A		
	FREMD.R		FREMDZ		0000		000A		
	START.R		STARTD		0054		0200		0253
	IRQ.R		IRQD		0000		0254		
	UNTER.R		UNTERD		0004		0254		0257
	TAST.R		TASTD		0007		0258		025E
	FREMD.R		FREMDD		0014		025F		0272
	START.R		START		0194		C000		C193
	IRQ.R		IRQ		003A		C194		C1CD
	UNTER.R		UNTER		018F		C1CE		C35C
	TAST.R		TAST		0179		C35D		C4D5
	FREMD.R		FREMD		0142		C4D6		C617
	XILINX.R		XILINX		2B56		C618		F16D
	START.R		VEKT		0006		FFFA		FFFF

----- LINKER V2.6k ----- DR. KROHN & STILLER ----- 2 --

referencetable

file	START.R	rseg	START	length	0194	at	C000	to	C193
			STARTZ		000A		0000		0009
			STARTD		0054		0200		0253
			VEKT		0006		FFFA		FFFF

R0	0000	R1	0001	R2	0002	R3	0003	R4	0004
R5	0005	R6	0006	R7	0007	START	C000	STARTD	0200
STARTZ	0000	VEKT	FFFA	VMODE	0253				

file	IRQ.R	rseg	IRQ	length	003A	at	C194	to	C1CD
			IRQZ		0000		000A		
			IRQD		0000		0254		

IRQ	C194	IRQA	C194	IRQD	0254	IRQZ	000A	NMIA	C1CD
-----	------	------	------	------	------	------	------	------	------

file	UNTER.R	rseg	UNTER	length	018F	at	C1CE	to	C35C
			UNTERZ		0000		000A		
			UNTERD		0004		0254		0257

BIN2BCD	C1CE	CCAS	0255	CCAS	C26D	CCRAS	C27C	CCROW	C283
CHSET	C2DD	CRAS	0257	CRW1	0256	DCCAS	C22E	DCRAS	C24F
DCRW1	C263	DECCAS	C1EF	DECRAS	C1FF	ICCAS	C217	ICRAS	C245
ICRW1	C259	INCCAS	C1E7	INCRAS	C1F7	UNTER	C1CE	UNTERD	0254
UNTERZ	000A	WAIT	C2BC	WTLAUF	C2CE	XILUP	C293		

file	TAST.R	rseg	TAST	length	0179	at	C35D	to	C4D5
			TASTZ		0000		000A		
			TASTD		0007		0258		025E

ANZLED1	025E	ANZNEU	C35D	CONIN	C449	IBUFB	025C	TASLOW	C3A5
TAST	C35D	TASTD	0258	TASTZ	000A				

file	FREMD.R	rseg	FREMD	length	0142	at	C4D6	to	C617
			FREMDZ		0000		000A		
			FREMDD		0014		025F		0272

ANRE	025F	ANZR	C518	ATINIT	C4D6	FREMD	C4D6	FREMDD	025F
FREMDZ	000A	TAAU	C576						

file	XILINX.R	rseg	XILINX	length	2B56	at	C618	to	F16D
------	----------	------	--------	--------	------	----	------	----	------

XILINX	C618	XILINXE	F16E	XILINXL	2B56	XILINXS	C618
--------	------	---------	------	---------	------	---------	------

startaddress: C000

F. Datenblätter

F.1. UVC 3130: A/D-D/A-Wandler

Entnommen aus [7]

UVC 3730

Pin	Connections
1	+ 5 V Supply D/A Converter Output Amplifier (Buffer)
2	Analog Input D/A Converter
3	- 5 V Supply D/A Converter
4	Digital Input Bit 5 (MSB)
5	Digital Input Bit 8
6	Digital Input Bit 7
7	Digital Input Bit 6
8	Digital Input Bit 5
9	Digital Input Bit 4
10	Digital Input Bit 3
11	Digital Input Bit 2
12	Digital Input Bit 1
13	Digital Input Bit 0 (LSB)
14	+ 5 V Supply D/A Converter digital
15	Digital Input Bit 0 (LSB)
16	GND of Ref. Voltage A/D Converter
17	+ 5 V Supply A/D Converter analog
18	Clock Input A/D Converter
19	+ 5 V Supply Analog Input Amplifier and Voltage
20	Reference A/D
21	Analog Input A/D Converter
22	Clamping Level Input
23	Key Pulse Input
24	Analog Ground A/D Converter
25	Reference Voltage A/D Converter
26	+ 5 V Supply A/D Converter digital
27	Digital Output Bit 7 (MSB)
28	Digital Output Bit 6
29	Digital Output Bit 5
30	Digital Output Bit 4
31	Digital Output Bit 3
32	Digital Output Bit 2
33	Digital Output Bit 1
34	Digital Output Bit 0 (LSB)
35	Digital Input Bit 0 (LSB)
36	+ 5 V Supply A/D Converter analog
37	GND of Ref. Voltage A/D Converter
38	External Analog Input
39	Output Signal Switchover Input
40	+ 5 V Supply D/A Converter Output Amplifier (Final stage)

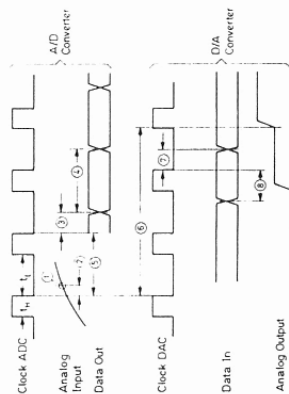


Fig 3: Timing diagram of the UVC 3130 A/D-D/A Converter

- ① Sample
- ② Aperture delay
- ③ Digital output delay
- ④ Data valid (after sample ①)
- ⑤ Transfer time A/D
- ⑥ Total transfer time A/D-D/A with common clock
- ⑦ Input register hold time
- ⑧ Input register setup time

of the A/D converter. The reference voltage for the A/D converter is generated on-chip, but both the ground of that circuit and the reference voltage are fed to pins, so that an external filter capacitor may be connected. Further, the input is equipped with switches which optionally provide operation with keyed clamping or peak clamping or without clamping (see also section 6.)

Also the D/A converter's reference voltage is generated on-chip, and a gated amplifier is arranged at the output of the D/A converter so that an external analog signal can be fed in instead of the signal delivered by the D/A converter

Separate clock inputs are provided for the A/D converter and the D/A converter thus enabling the application of time compression procedures. All inputs and outputs are TTL compatible.

2. Outline Dimensions and Pin Connections

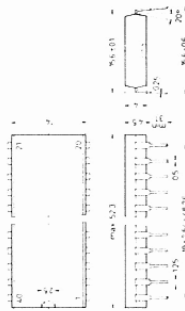


Fig. 2:
UVC 3130 in 40-pin DII Plastic Package,
20 B 40 according to DIN 41870

Weight approx. 6 g Dimensions in mm

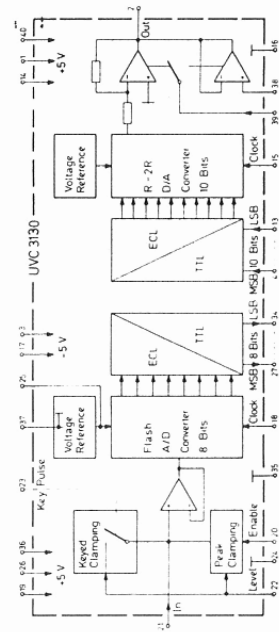


Fig. 1: UVC 3130 block diagram

UVC 3130

High-Speed A/D-D/A Converter

VLSI circuit in CI technology featuring the following circuits:

- a high-speed flash type 8-bit A/D converter
- a high-speed low-glitch 10-bit D/A converter, designed as an R-2R network with switched current sources
- various auxiliary circuits, such as reference voltage sources, preamplifier, input clamping circuit, and feed-in output amplifier

UVC 3130 has been developed for use in all applications which call for a high-speed A/D-D/A converter. For instance, the device can be used to advantage to decode television signals in Pay-TV converters or D2-MAC converters used in direct satellite broadcast. Other promising applications can be seen in industrial electronics, e.g. in conjunction with digital signal processing. Although UVC 313130 was initially designed as high-speed codec for the video-range, it can be used with equal benefits for lower frequencies, even down to zero.

To meet different application requirements, the UVC 3130 is classified into four groups with respect to the linearity of the D/A converter

Brand	Linearity D/A
UVC 3130-10	10 Bit
UVC 3130-09	9 Bit
UVC 3130-08	8 Bit
UVC 3130-07	7 Bit

1. General Information

The above auxiliary circuits contained on-chip provide versatile potential applications needing a minimum of external components. For example, an impedance converter is connected upstream of the A/D converter to provide a high-impedance signal input, in spite of the high input capacitance

UVC 3130

3. Electrical Characteristics

All voltages are referred to pins 16, 24, 35 and 37

Absolute Maximum Ratings

Symbol	Value	Unit
$+V_B$	6	V
$-V_B$	6	V
V_i	-0.5 V to (+ V_B + 0.5 V)	-
V_i	-1.5 V to (+ V_B + 0.5 V)	-
I_O	± 10	mA
T_A	0 to +65	°C
T_S	-40 to +125	°C

Recommended Operating Conditions

Symbol	Min	Typ	Max	Unit
Positive Supply Voltage	4.75	5	5.25	V
Negative Supply Voltage	4.75	5	5.25	V
A/D Converter				
Analog Input Voltage	0	-	2	V
f_i	-	-	$< \frac{f_{cl}}{2}$	-
V_{iBH}	2.4 V	-	+ V_B	V
V_{iBL}	0	-	0.8	V
f_{1H}	0	-	30	MHz
t_H	10	-	-	ns
t_L	23	-	-	ns
V_{22}	-1	-	+2	V
V_{23H}	2.4 V	-	+ V_B	V
V_{23L}	0	-	0.8	V
Resistor of 20 to 60 k Ω from Pin 20 to +5 V				
V_{1SH}	2.4 V	-	+ V_B	V
V_{1SL}	0	-	0.8	V
f_{1S}	0	-	30	MHz
V_{1H}	2.4 V	-	+ V_B	V
V_{1L}	0	-	0.8	V
V_{38}	-1	-	+3	V
V_{39}	0	-	0.8	V
V_{39}	2 V	-	+ V_B	V

Analog Input Voltage at pin 38

Control Voltage for the Output Gate Amplifier

Input Signal from Pin 21 at the Output Pin 2

Input Signal from Pin 38 at the Output Pin 2

UVC 3130

Characteristics at + V_B = 5 V, - V_B = 5 V, f_{1S} = 25 MHz, f_{1B} = 25 MHz, T_A = 25 °C

	Symbol	Min	Typ	Max	Unit
Current Consumption	I_B	-	-	150	mA
	$-I_B$	-	-	150	mA
Power Dissipation	P_{tot}	-	-	1.5	W
Total Transfer Time A/D-D/A ((a) in Fig. 3)	t_{tot}	-	see Fig. 3	-	-
A/D Converter					
Input Current Pin 21	I_i	-	1	-	μ A
Input Impedance Pin 21	Z_i	-	20	-	M Ω
at f = 1 MHz	Z_i	-	40	-	k Ω
at f = 10 MHz	Z_i	-	4	-	k Ω
Input Capacitance Pin 21	C_i	-	4	-	pF
3 dB Bandwidth of the Input Amplifier	V_{23}	-	30	-	MHz
Clamping Active at	R_{on}	2.4	-	-	V
ON Resistance of the Clamping Switch		-	300	-	Ω
Between Pins 21 and 22		-	-	-	-
Input Current of the Clamping Level Input 22	I_{22}	-	200	-	μ A
Aperture Delay (z in Fig. 3)	t_{sd}	-	-	10	ns
Digital Output Delay (a in Fig. 3)	t_{dv}	-	18	-	ns
Transfer Time (s in Fig. 3)	t_{tr}	-	one Clock Period	-	-
Differential Non-Linearity	-	-	$\pm 1/2$ LSB	-	-
Absolute Non-Linearity	-	-	1	-	%
Number of Bits	-	-	8	-	-
Code of the Digital Output Signals	-	-	binary	-	-
Output Signal	-	-	0 0 0 0 0 0 0	-	-
at V_{21} = 0 V	-	-	1 1 1 1 1 1 1	-	-
Internal Reference Voltage, accessible from outside	V_{25}	-	≥ 0	-	V
D/A Converter					
Input Register Hold Time ((7) in Fig. 3)	t_{rh}	7.5	-	-	ns
Input Register Setup Time ((8) in Fig. 3)	t_{is}	10	-	-	ns
Differential Non-Linearity (referred to 10 bit)	L_{in}	-	$\pm 1/2$ LSB	-	-
UVC 3130-10					
UVC 3130-09	L_{in}	-	± 1 LSB	-	-
UVC 3130-08	L_{in}	-	± 2 LSB	-	-
UVC 3130-07	L_{in}	-	± 4 LSB	-	-
Absolute Non-Linearity	-	-	1	-	%
Number of Bits	-	-	10	-	-
Code of the Digital Input Signal	-	-	binary	-	-
Output Signal	V_2	-	0	-	V
with 0 0 0 0 0 0 0 at the Inputs	V_2	-	2	-	V
with 1 1 1 1 1 1 1 at the Inputs	Z_O	-	15	-	Ω
Output Impedance Pin 2	I_i	-	1	-	μ A
Input Current Pin 38	V_{ref}	-	2	-	V
Internal Reference Voltage		-	-	-	-

UVC 3130

4. Pin Circuits

The following figures schematically show the circuitry at the various pins. X = protection diode

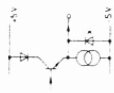


Fig. 4:
Pin 2, Output

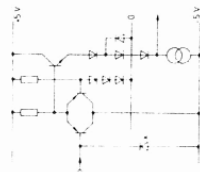


Fig. 5:
Pins 4 to 13 and 15, Inputs

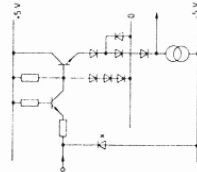


Fig. 6:
Pins 18, 23 and 39, Inputs

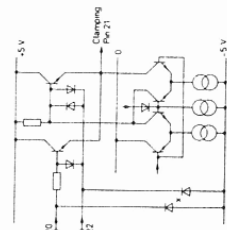


Fig. 7:
Pins 20 and 22, Inputs

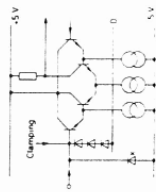


Fig. 8:
Pin 21, Input

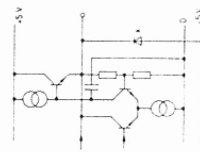


Fig. 9:
Pin 25, Reference Voltage Pin

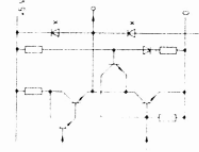


Fig. 10:
Pins 27 to 34, Outputs

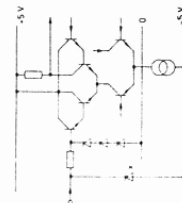


Fig. 11:
Pin 38, Input

UVC 3130

5. Pin Descriptions

Pin 1 – +5 V Supply D/A Converter Output Amplifier
By this pin, the buffer stage of the D/A converter's analog output amplifier is powered

Pin 2 – Analog Output D/A Converter
This pin whose diagram is shown in Fig. 4, is the output for the processed analog signal either originating from the D/A converter or from the external analog input pin 38

Pin 3 – –5 V Supply D/A Converter Analog
This pin gets the negative supply for the analog part of the D/A converter

Pins 4 to 13 – Digital Inputs Bit 9 to Bit 0
The diagram of these pins is shown in Fig. 5. They are the inputs of the D/A converter. Not-used inputs should be connected to ground

Pin 14 – +5 V Supply D/A Converter Digital
This pin gets the positive supply for the digital part of the D/A converter

Pin 15 – Clock Input D/A Converter
This pin whose diagram is shown in Fig. 5 must be supplied with the clock signal for the D/A converter

Pin 16 – GND D/A Converter and Clock A/D Converter
This pin serves as ground pin for the D/A converter and for the clock of the A/D converter

Pin 17 – –5 V Supply A/D Converter Analog
This pin is the negative supply pin for the analog part of the A/D converter

Pin 18 – Clock Input A/D Converter
The diagram of this pin is shown in Fig. 6. Pin 18 is supplied with the clock of the A/D converter

Pin 19 – +5 V Supply Analog Input Amplifier and Voltage Reference A/D
Via this pin the analog input amplifier and the voltage reference of the A/D converter are powered

Pin 20 – Peak Clamping Enable Input
Via pin 20 whose diagram is shown in Fig. 7, the peak clamping facility can be enabled

Pin 21 – Analog Input A/D Converter
Fig. 8 is the diagram of this input. To pin 21 the analog signal to be converted into digital is applied

Pin 22 – Clamping Level Input
Via this pin whose diagram is shown in Fig. 7, the input of the A/D converter is supplied with the desired clamping level

Pin 23 – Key Pulse Input
Fig. 6 is the diagram of this input. Pin 23 must be supplied with the key pulse if keyed clamping is required

Pin 24 – Analog Ground A/D Converter
This pin serves as ground pin for the analog part of the A/D converter. It must be connected with pin 37 as short as possible

Pin 25 – Reference Voltage A/D Converter

This pin whose diagram is shown in Fig. 9, is intended for connecting a decoupling capacitor to the A/D converter's reference voltage, the other end of this capacitor to pin 37

Pin 26 – +5 V Supply A/D Converter Digital
This pin is the positive supply pin for the digital part of the A/D converter

Pins 27 to 34 – Digital Outputs Bit 7 to Bit 0
Fig. 10 shows the diagram of these outputs which supply the digitalized analog signal in parallel 8-bit code

Pin 35 – Digital Ground A/D Converter
This pin is the ground connection for the digital part of the A/D converter

Pin 36 – +5 V Supply A/D Converter Analog
This pin is the positive supply pin for the analog part of the A/D converter

Pin 37 – Ground of Reference Voltage A/D Converter
To this pin must be connected the ground end of the decoupling capacitor which is at pin 25

Pin 38 – External Analog Input
The diagram of this input is shown in Fig. 11. Pin 38 serves for feeding an external analog signal into the output amplifier of the UVC 3130 instead of the D/A-converted signal originating from pins 4 to 13

Pin 39 – Output Signal Switchover Input
This pin whose diagram is shown in Fig. 6, is intended for enabling the external analog signal fed to pin 38

Pin 40 – +5 V Supply D/A Converter Output Amplifier
By this pin, the final (power) stage of the D/A converter's analog output amplifier is powered

UVC 3130

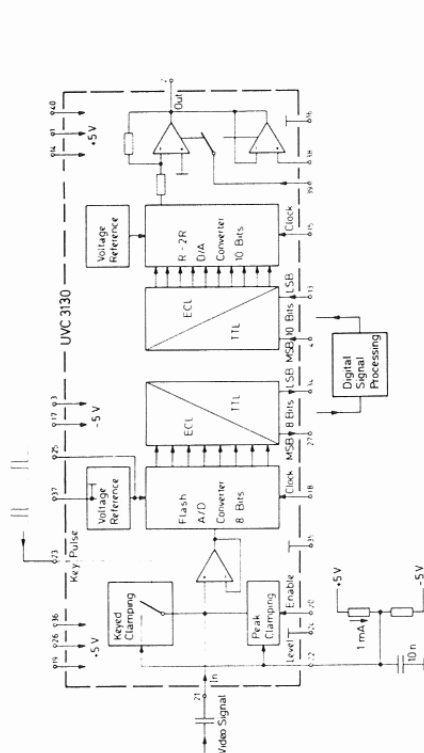


Fig. 13: Operation with keyed clamping. During the key pulse, the input signal must have that level to which it is to be clamped. Pin 20 must not be connected.

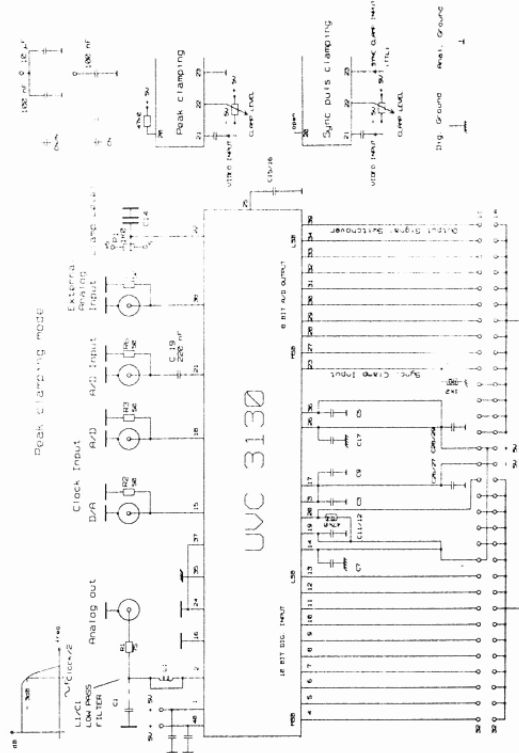


Fig. 14: Application diagram including power supply and associated decoupling elements.

UVC 3130

6. Appendix: Application Circuits

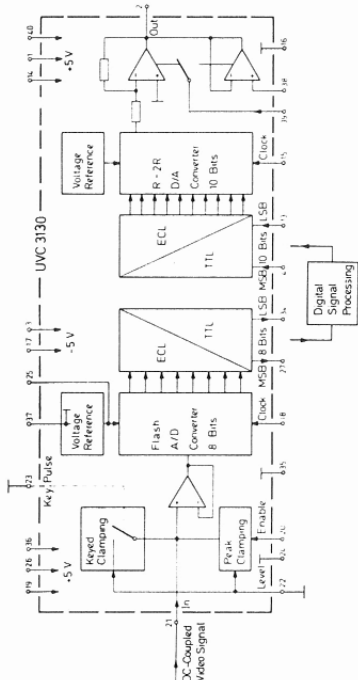


Fig. 11: Operation without clamping of the input signal. Pin 20 must not be connected.

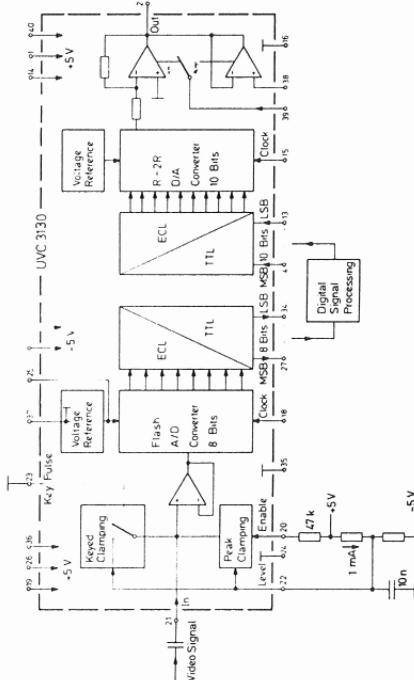
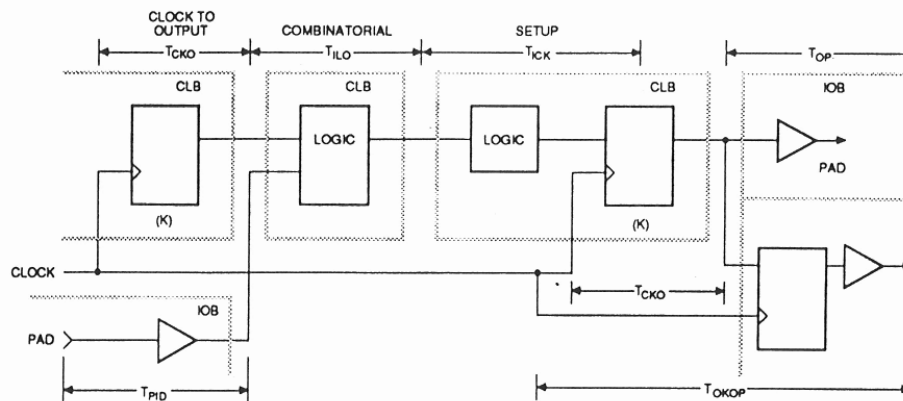
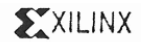


Fig. 12: Operation with peak clamping. The input signal is clamped automatically to the negative peak value. No key pulse is needed.

F.2. XILINX XC3000-Serie, Zeitverhalten

Entnommen aus [3]



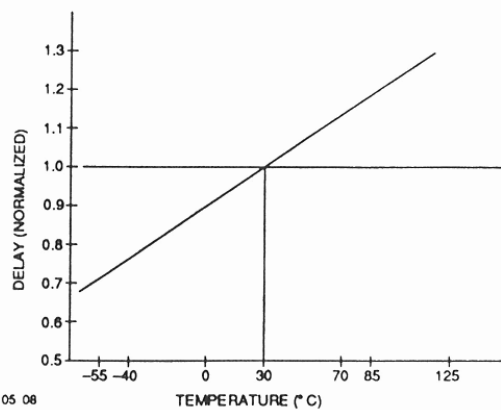
1105 21

Speed Grade			-50		-70		-100		Units
	Description	Symbol	Min	Max	Min	Max	Min	Max	
Logic input to Output	Combinatorial	T _{ILO}		14		9		7	ns
K Clock	To output	T _{CKO}		12		8		7	ns
	Logic-input setup	T _{ICK}	12		8		7		ns
	Logic-input hold	T _{CKI}	0		0		0		ns
Input/Output	Pad to input (direct)	T _{PID}		9		6		4	ns
	Output to pad (fast)	T _{OP}		15		9		6	ns
	I/O clock to pad (fast)	T _{OKPO}		18		13		10	ns
FF toggle frequency		F _{CLK}		50		70		100	MHz

Examples of Primary Block Speed Factors.

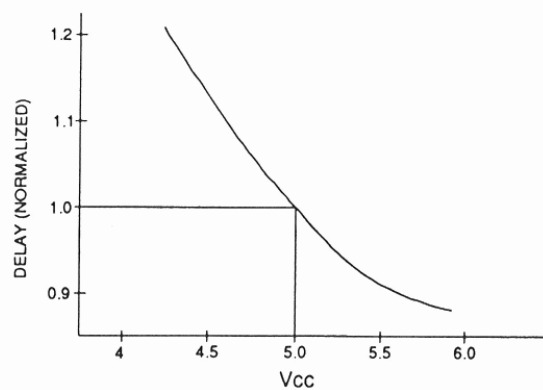
Actual timing is a function of various block factors combined with routing factors.

Overall performance can be evaluated with the XACT timing calculator or by an optional simulation.



1105 08

Change in speed performance as a function of temperature, normalized for 30°C.

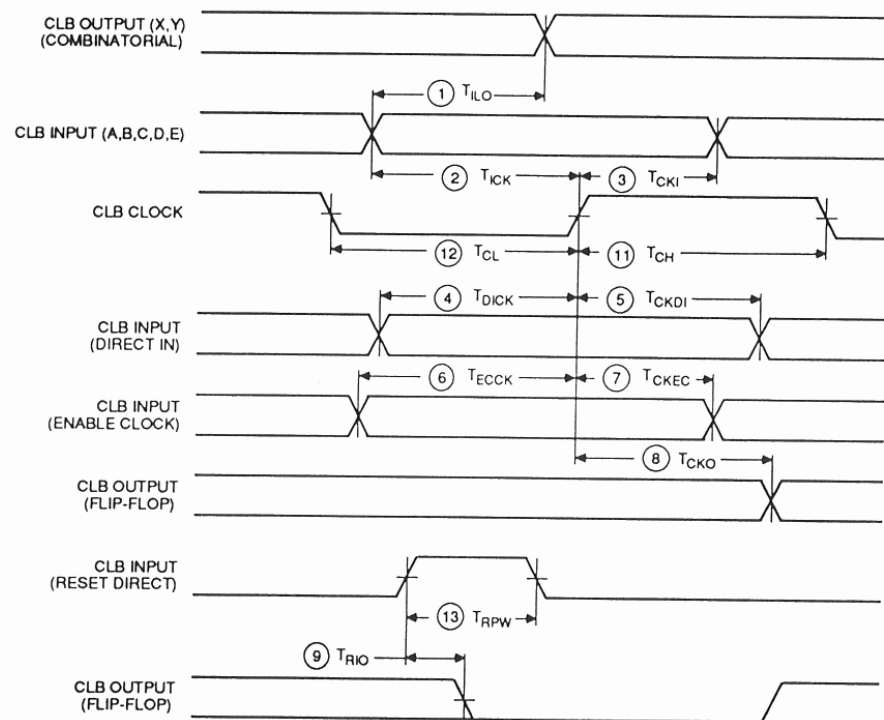


1105 22

The speed performance of a CMOS device increases with V_{CC} within the operating range.

XC3000 Logic Cell Array Family

CLB SWITCHING CHARACTERISTIC GUIDELINES

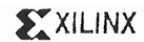


1105 26

BUFFER (Internal) SWITCHING CHARACTERISTIC GUIDELINES

Speed Grade		-50		-70		-100		Units
Description	Symbol	Min	Max	Min	Max	Min	Max	
Global and Alternate Clock Distribution** Either: Normal IOB input pad to clock buffer input Or: Fast (CMOS only) input pad to clock buffer input Plus: Clock buffer input to any clock k	T_{PID}		9		6		4	ns
	T_{PIDC}		5		3		2	ns
			9		6		5	ns
TBUF driving a Horizontal Longline (L.L.)** I to L.L. while T is Low (buffer active) T↓ to L.L. active and valid T↑ to L.L. (inactive) with single pull-up resistor with pair of pull-up resistors	T_{ID}		8		5		4	ns
	T_{ON}		15		9		7	ns
	T_{PUS}		34		22		14	ns
	T_{PUF}		17		11		7	ns
BIDI Bi-directional buffer delay			6		4		3	ns

** Timing is based on the XC3020, for other devices see XACT timing calculator.



CLB SWITCHING CHARACTERISTIC GUIDELINES (Continued)

Testing of the switching characteristic guidelines is modeled after testing specified by MIL-M-38510/605. Devices are 100% functionally tested. Benchmark timing patterns are used to provide correlation to the switching characteristic guideline values. Actual worst-case timing is provided by the XACT Timing calculator or Simulation modeling.

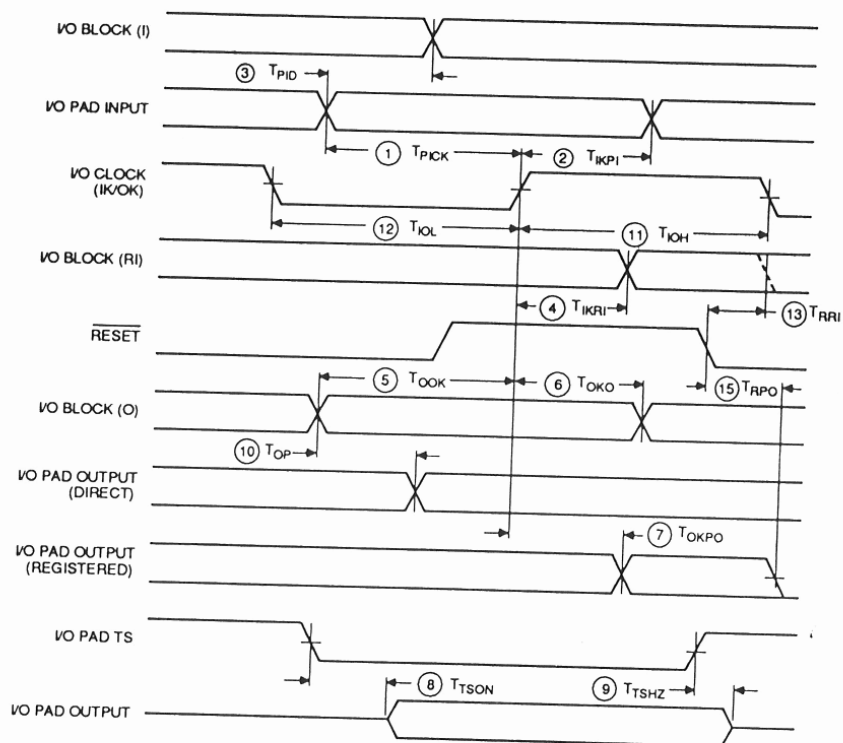
		Speed Grade		-50		-70		-100		Units
Description	Symbol	Min	Max	Min	Max	Min	Max	Min	Max	
Combinatorial Delay										
Logic Variables a, b, c, d, e, to outputs x, y	1 T _{LO}		14		9		7			ns
Sequential delay										
Clock k to outputs x, y	8 T _{CKO}		12		8		7			ns
Clock k to outputs x,y when Q is returned through function generators F or G to drive x, y			23		15		12			ns
Set-up time before clock K										
Logic Variables a, b, c, d, e	2 T _{ICK}	12		8		7				ns
Data In di	4 T _{DICK}	8		5		4				ns
Enable Clock ec	6 T _{ECCK}	10		7		5				ns
Reset Direct inactive rd		2		1		1				ns
Hold Time after clock k										
Logic Variables a, b, c, d, e	3 T _{CKI}	0		0		0				ns
Data In di	5 T _{CKDI}	6		4		2				ns
Enable Clock ec	7 T _{CKEC}	0		0		0				ns
Clock										
Clock High time*	11 T _{CH}	9		7		5				ns
Clock Low time*	12 T _{CL}	9		7		5				ns
Max. flip-flop toggle rate*	F _{CLK}	50		70		100				MHz
Reset Direct (rd)										
rd width	13 T _{RPW}	12		8		7				ns
delay from rd to outputs x, y	9 T _{RIO}		12		8		7			ns
Master Reset (MR)										
MR width	T _{MRW}	38		25		21				ns
delay from MR to outputs x, y	T _{MRO}		30		20		17			ns

* These timing limits are based on calculations.

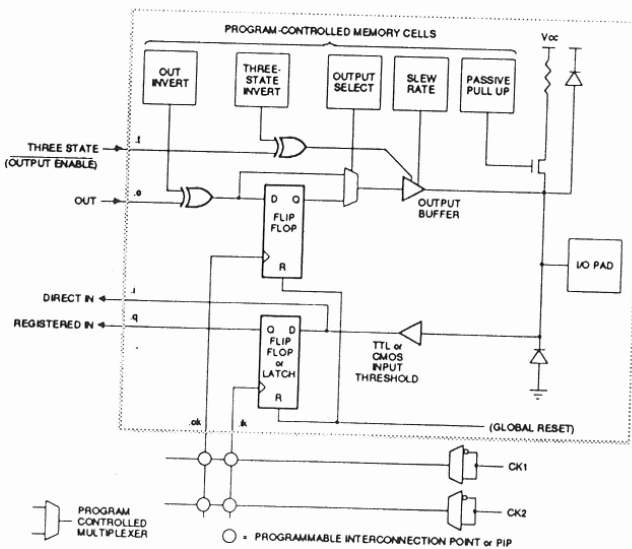
Note: The CLB K to Q output delay (T_{CKO}, #8) of any CLB, plus the shortest possible interconnect delay, is always longer than the Data In hold time requirement (T_{CKDI}, #5) of any CLB on the same die.

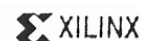
XC3000 Logic Cell Array Family

IOB SWITCHING CHARACTERISTIC GUIDELINES



1105 27





IOB SWITCHING CHARACTERISTIC GUIDELINES (Continued)

Testing of the switching characteristic guidelines is modeled after testing specified by MIL-M-38510/605. Devices are 100% functionally tested. Benchmark timing patterns are used to provide correlation to the switching characteristic guideline values. Actual worst-case timing is provided by the XACT Timing calculator or Simulation modeling.

			-50		-70		-100		Units
Description	Symbol		Min	Max	Min	Max	Min	Max	
Propagation Delays (Input)									
Pad to Direct In (i)	3	TPID		9		6		4	ns
Pad to Registered In (q) with latch transparent		TPTG		34		21		17	ns
Clock (ik) to Registered In (q)	4	TIKRI		11		7		6	ns
Set-up Time (Input)									
Pad to Clock (ik) set-up time	1	TPICK	30		20		17		ns
Propagation Delays (Output)									
Clock (ok) to Pad (fast)	7	TOKPO		18		13		10	ns
same (slew rate limited)	7	TOKPO		43		33		27	ns
Output (o) to Pad (fast)	10	TOPF		15		9		6	ns
same (slew-rate limited)	10	TOPS		40		29		23	ns
Three-state to Pad begin hi-Z (fast)	9	TTSHZ		12		8		8	ns
same (slew-rate limited)	9	TTSHZ		37		28		25	ns
Three state to Pad active and valid (fast)	8	TTSON		20		14		12	ns
same (slew -rate limited)	8	TTSON		45		34		29	ns
Set-up and Hold Times (Output)									
Output (o) to clock (ok) set-up time	5	TOOK	15		10		9		ns
Output (o) to clock (ok) hold time	6	TOKO	0		0		0		ns
Clock									
Clock High time	11	TCH	9		7		5		ns
Clock Low time	12	TCL	9		7		5		ns
Max. flip-flop toggle rate		FCLK	50		70		100		MHz
Master Reset Delays									
RESET Pad to Registered In (q)	13	TRRI		35		23		20	ns
RESET Pad to output pad	15	TRPO		50		33		28	ns

Notes: 1. Timing is measured at pin threshold, with 50 pF external capacitive loads (incl. test fixture).

Typical fast mode output rise/fall times are 2 ns and will increase approximately 2%/pF of additional load.

Typical slew rate limited output rise/fall times are approximately 4 times longer.

A maximum total external capacitive load for simultaneous fast mode switching in the same direction is 500 pF per power/ground pin pair. For slew-rate limited outputs this total is 4 times larger.

2. Voltage levels of unused (bonded and unbonded) pads must be valid logic levels. Each can be configured with the internal pull-up resistor or alternatively configured as a driven output or driven from an external source.

3. Input pad set-up time is specified with respect to the internal clock (.ik)


In order to calculate system set-up time, subtract clock delay (pad to ik) from the input pad set-up time value.

Input pad holdtime with respect to the internal clock (ik) is negative. This means that pad level changes immediately before the internal clock edge (ik) will not be recognized.

For a more detailed description see the discussion on "LCA Performance" in the Applications chapter (6-14 to 18).

F.3. μ PD424256-12, 256K*4 Bit D-RAM

Entnommen aus [6]



μ PD424256C/LA

μ PD424256 (FAST PAGE)

262.144 W x 4 BIT DRAM

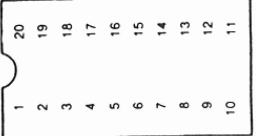
FEATURES

- 262.144 words by 4 bit organization
- Low power dissipation CMOS DRAM
- Single 5 V \pm 10% power supply
- CAS before RAS internal address refresh mode
- 512 cycle, 8 ms refresh
- High density 20-pin plastic DIP (μ PD424256C) or 26/20-pin plastic SOJ (μ PD424256LA)
- Fast page mode

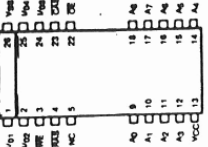
FAMILY	TRAC	TCAC	TAA	ICC1
μ PD424256 - 8	80 ns	20 ns	45 ns	70 mA
μ PD424256 - 10	100 ns	25 ns	55 ns	60 mA
μ PD424256 - 12	120 ns	30 ns	65 ns	50 mA

* ICAC = IOEA

20 PIN DIP



26/20 PIN SOJ



PIN CONFIGURATION

<p>1 2 3 4 5 6 7 8 9 10</p> <p>I/O1 I/O2 WE RAS NC A0 A1 A2 A3 VCC</p>	<p>11 12 13 14 15 16 17 18 19 20</p> <p>VSS I/O4 I/O3 CAS OE A8 A7 A6 A5 A4</p>
--	---

<p>1 2 3 4 5 6 7 8 9 10</p> <p>I/O1 I/O2 WE RAS NC A0 A1 A2 A3 VCC</p>	<p>11 12 13 14 15 16 17 18 19 20</p> <p>VSS I/O4 I/O3 CAS OE A8 A7 A6 A5 A4</p>
--	---

ADDRESS INPUT

DATA INPUT/OUTPUT

CAS

RAS

ROW ADDRESS STROBE

OE

WRITE ENABLE

NO CONNECTION

POWER SUPPLY

GROUND

NEC**μPD424256C/LA****ABSOLUTE MAXIMUM RATINGS**

Voltage on any pin relative to GND -----
 Operating temperature, T_{OPT} (ambient) -----
 Storage temperature, T_{STG} (ambient) -----
 Short circuit output current -----
 Power dissipation -----

-1.0 to + 7.0 V
 0 to + 70 °C
 -55 to + 125 °C
 50 mA
 1 W

DC CHARACTERISTICS
($T_A = 0$ to 70 °C, $V_{CC} = 5V \pm 10\%$)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	TEST CONDITION
SUPPLY VOLTAGE	V_{CC}	4.5	5.0	5.5	V	ALL VOLTAGES REFERENCED TO GND
HIGH LEVEL INPUT VOLTAGE	V_{IH}	2.4		5.5	V	
LOW LEVEL INPUT VOLTAGE	V_{IL}	-1.0		0.8	V	
STANDBY CURRENT	I_{CC2}			3.0	mA	$\overline{RAS} = \overline{CS} = V_{IH}$
INPUT LEAKAGE CURRENT	$I_{I(L)}$	-10		1.0	mA	$\overline{RAS} = \overline{CAS} = V_{CC} - 0.2 V$
OUTPUT LEAKAGE CURRENT	$I_{O(L)}$	-10		10	μA	
OUTPUT LOW VOLTAGE	V_{OL}	0		0.4	V	$I_{OL} = 4.2 mA$
OUTPUT HIGH VOLTAGE	V_{OH}	2.4		V_{CC}		$I_{OH} = -5 mA$

CAPACITANCE
($T_A = 25^\circ C$, $F = 1 MHz$)

PARAMETER	SYMBOL	MAX	UNITS
ADDRESS	C_{I1}	6	pF
\overline{RAS} , \overline{CAS} , \overline{WE} , \overline{OE}	C_{I2}	8	pF
INPUT/OUTPUT	C_D	7	pF

NEC**μPD424256C/LA****AC CHARACTERISTICS**
($T_A = 0^\circ C$ to 70 °C, $V_{CC} = 5V \pm 10\%$)

PARAMETER	SYM-BOL	μPD4256 · 8		μPD4256 · 10		μPD4256 · 12		UNITS
		MIN	MAX	MIN	MAX	MIN	MAX	
VCC OPERATING CURRENT	I_{CC1}		70		60		50	mA
VCC \overline{RAS} ONLY REFRESH CURRENT	I_{CC3}		60		50		40	mA
VCC OPERATING CURRENT (PAGE MODE)	I_{CC4}		60		50		40	mA
VCC CBR REFRESH CURRENT	I_{CC6}		60		50		40	mA
RANDOM READ OR WRITE CYCLE TIME	t_{RC}	170		190		220		ns
READ WRITE CYCLE TIME	t_{RWC}	240		265		305		ns
FAST PAGE MODE CYCLE TIME	t_{PC}	60		70		85		ns
ACCESS TIME FROM \overline{RAS}	t_{RAC}		80		100		120	ns
ACCESS TIME FROM \overline{CAS}	t_{CAC}		20		25		30	ns
ACCESS TIME FROM COLUMN ADDRESS	t_{AA}		45		55		65	ns
ACCESS TIME FROM \overline{CAS}	t_{ACP}		55		65		80	ns
PRECHARGE								
OUTPUT BUFFER TURN-OFF DELAY	t_{OFF}	0	20	0	25	0	30	ns
TRANSITION TIME (RISE AND FALL)	t_T	3	50	3	50	3	50	ns
\overline{RAS} PRECHARGE TIME	t_{RP}	80		80		90		ns
\overline{RAS} PULSE WIDTH	t_{RAS}	80	10000	100	10000	120	10000	ns
\overline{RAS} HOLD TIME	t_{RSH}	20		20		25		ns
\overline{CAS} PULSE WIDTH	t_{CAS}	20	10000	20	10000	25	10000	ns
\overline{CAS} HOLD TIME	t_{CSH}	80		100		120		ns
\overline{RAS} TO \overline{CAS} DELAY TIME	t_{RCD}	25	60	25	75	25	90	ns
\overline{RAS} TO COLUMN ADDRESS DELAY TIME	t_{RAD}	20	35	20	45	20	55	ns
\overline{CAS} TO \overline{RAS} PRECHARGE TIME	t_{CRP}	10		10		10		ns
\overline{CAS} PRECHARGE TIME	t_{CP}	10		10		15		ns
ROW ADDRESS SET-UP TIME	t_{ASR}	0		0		0		ns
ROW ADDRESS HOLD TIME	t_{RAH}	15		15		15		ns
COLUMN ADDRESS SET-UP TIME	t_{ASC}	0	20	0	30	0	35	ns

AC CHARACTERISTICS
($T_A = 0^\circ\text{C}$ to 70°C , $V_{CC} = 5\text{V} \pm 10\%$)

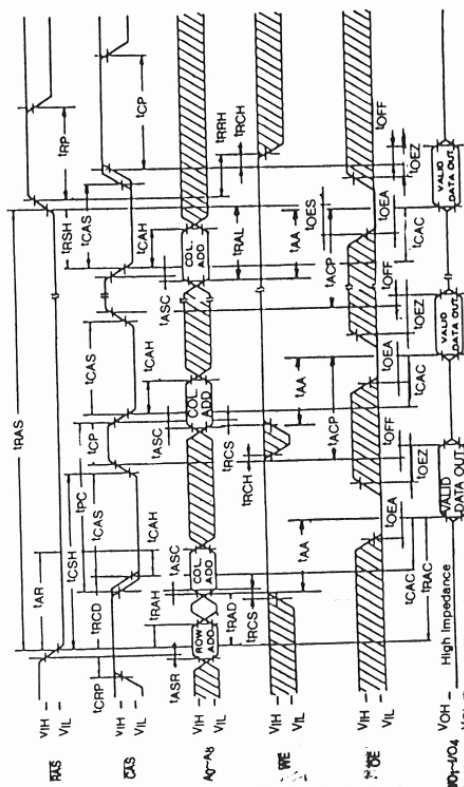
NFC

ИД PD424256C/A

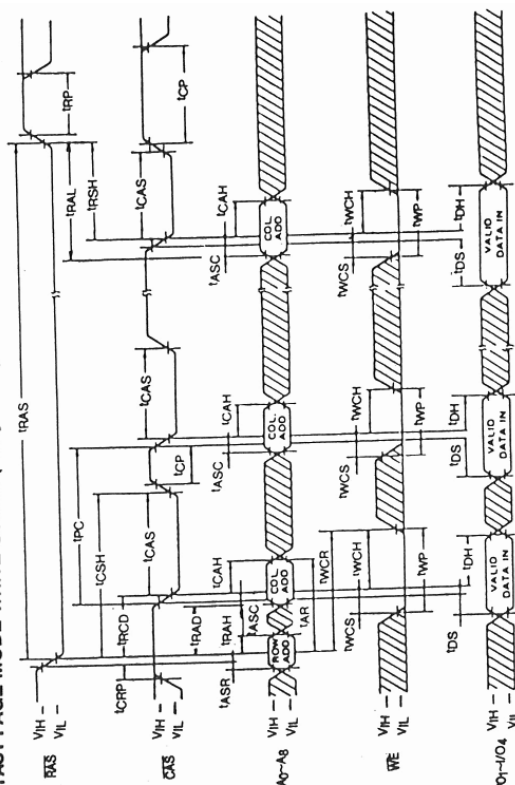


PD424256C/LA

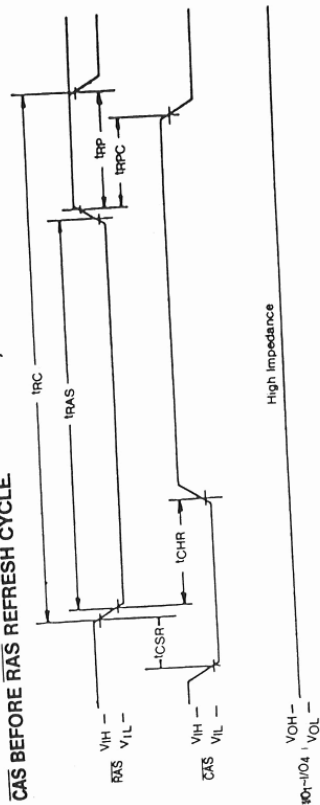
FAST PAGE MODE READ CYCLE



FAST PAGE MODE WRITE CYCLE (Early Write)

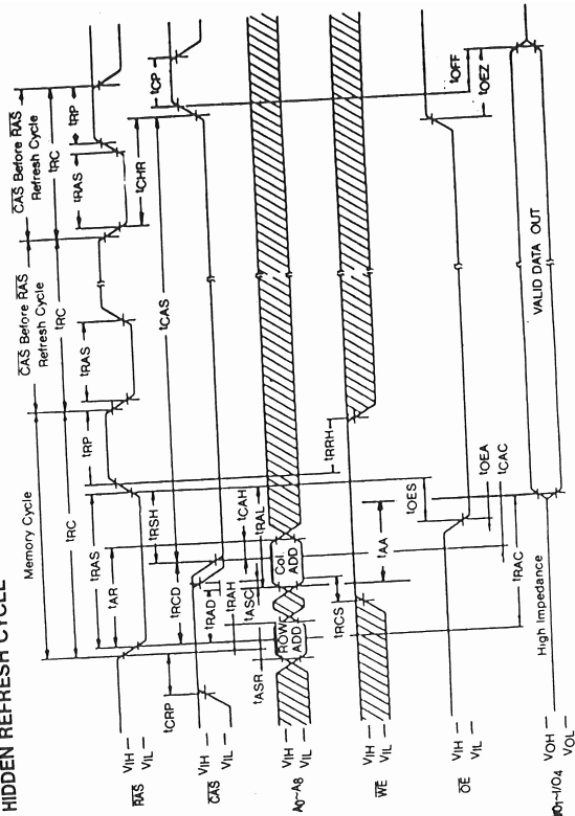


2.0 BEFORE BAS REFRESH CYCLE



WE OF ADDRESS: Don't Care

HIDDEN REFRESH CYCLE



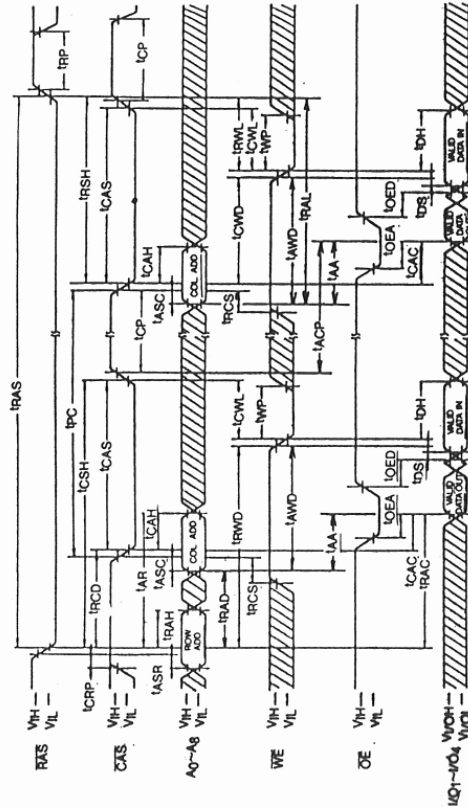
VALID DATA	OUT
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	20
21	21
22	22
23	23
24	24
25	25
26	26
27	27
28	28
29	29
30	30
31	31
32	32
33	33
34	34
35	35
36	36
37	37
38	38
39	39
40	40
41	41
42	42
43	43
44	44
45	45
46	46
47	47
48	48
49	49
50	50
51	51
52	52
53	53
54	54
55	55
56	56
57	57
58	58
59	59
60	60
61	61
62	62
63	63
64	64
65	65
66	66
67	67
68	68
69	69
70	70
71	71
72	72
73	73
74	74
75	75
76	76
77	77
78	78
79	79
80	80
81	81
82	82
83	83
84	84
85	85
86	86
87	87
88	88
89	89
90	90
91	91
92	92
93	93
94	94
95	95
96	96
97	97
98	98
99	99
100	100

1. How

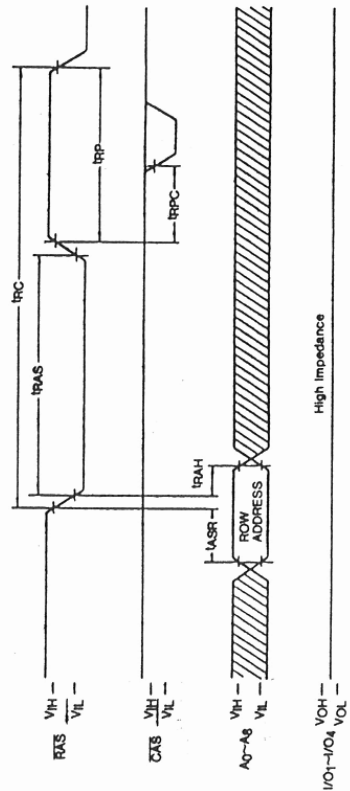
11111111

11

FAST PAGE MODE READ-WRITE/READ-MODIFY-WRITE CYCLE



RAS ONLY REFRESH CYCLE



WE, OE: Don-t Care

F.4. Video-Basisnormen der 625-Zeilen-Systeme

Entnommen aus [2]

Auszug aus den Fernsehnormen des CCIR

Videofrequente Basisnormen der 625-
Zeilen-Systeme [nach CCIR-Bericht 624-1]
(Comité Consultatif International des Radiocommuni-
cations)

Parameter	Nennwert, Toleranz
Zeilenzahl je Vollraster Z	625
Vollrasterfrequenz $f_v/2$	25 Hz
Halbrasterfrequenz f_v	50 Hz
Zeilendauer H	64 μ s ^a
Zeilenfrequenz f_H	15 625 Hz $\left\{ \begin{array}{l} \text{SW: } \pm 0,02 \% \\ \text{Farbe: } \pm 0,0001 \% \end{array} \right.$
Videobandbreite f_g	s. ^b
<i>Zeitintervalle im horizontalen</i>	
<i>Austastintervall (vgl. Bild 2-62)</i>	
Austastintervall a	12 μ s \pm 0,3 μ s
Intervall b (berechn. Mittelwert)	10,7 μ s
Intervall c (vordere Schwarzscher) d	1,5 μ s ^c \pm 0,2 μ s ^c
Zeilensynchronimpuls d	4,7 μ s \pm 0,2 μ s
Steig-/Fallzeit der Kanten e des horizontalen Austastintervalls	0,3 μ s \pm 0,1 μ s
Idem, Zeilensynchronimpuls f	0,2 μ s ^c \pm 0,1 μ s ^c
<i>Zeitintervalle im vertikalen</i>	
<i>Austastintervall (vgl. Bild 2-62)</i>	
Austastintervall j	25 H + a
Steig-/Fallzeit der Kanten j' des vertikalen Austastintervalls	0,3 μ s \pm 0,1 μ s
Intervall zwischen erster Flanke des vertikalen Austastintervalls und erster Kante des ersten Aus- gleichsimpulses k	3 μ s \pm 2 μ s
Erste Sequenz von Ausgleichs- impulsen l	2,5 H
Vertikale Synchronimpulse m	2,5 H
Zweite Sequenz von Ausgleichs- impulsen n	2,5 H
Ausgleichsimpulse p	2,35 μ s \pm 0,1 μ s
Vertikaler Synchronimpuls q	27,3 μ s ^c
Intervall zwischen vertikalen Synchronisierimpulsen r	4,7 μ s \pm 0,2 μ s ^c
Steig-/Fallzeit der Synchron- und Ausgleichsimpulse s	0,2 μ s ^c \pm 0,1 μ s ^c

^a Frankreich und Osteuropa: Kurzzeittoleranz
 $\pm 0,032 \mu$ s.

^b Normen B und G (Westeuropa und andere Länder,
ohne Frankreich und Großbritannien): 5 MHz,
übrige Normen s. Tabellen 2-IV und 2-IX.

^c I-Norm: $c: 1,65 \pm 0,032; f: 0,25 \pm 0,05; q: 27,3 \pm 0,1;$
 $r: 4,7 \pm 0,1; s: 0,25 \pm 0,05.$

Weitere Basisdaten für die B/G-Normen
(625 Zeilen; nach CCIR-Bericht 624-1)

Relativpegel BAS-Signal: 2 Darstellungsarten:

Austastwert A	30 % (0)
Weiß (peak white)	100 % (100)
Synchronwert	0 % (-43)
Differenz zwischen Schwarzwert und Austastwert	0; Toleranz $\left\{ \begin{array}{l} +5\% \\ -0 \end{array} \right.$

Mittleres γ der Wiedergabeseite 2,8 ($\gamma_{tot} \approx 1,2$)

Modulation und RF-Kanal:

Art und Polarität der Bildmodulation:	A5C negativ
Bild-/Ton-Trägerabstand	5,5 MHz
Breite Übertragungskanal	B: 7; G: 8 MHz
Bildträger über unterer Kanalgrenze	1,25 MHz
Breite des Restseitenbandes	0,75 MHz

Tonmodulation: Art: F 3 (Frequenzmodulation)

Hub	± 50 KHz
Pre-Emphase	50 μ s
Bildträgerleistung (Synchronspitze)/Tonträgerleistung	10/1 ^a

Ausgestrahltes Signal in % des Spitzenträgers

Synchronpegel	100
Austastpegel	75 \pm 2,5
Differenz zwischen Schwarz- und Austastpegel	0 bis 2
Weißpegel	10 bis 12,5

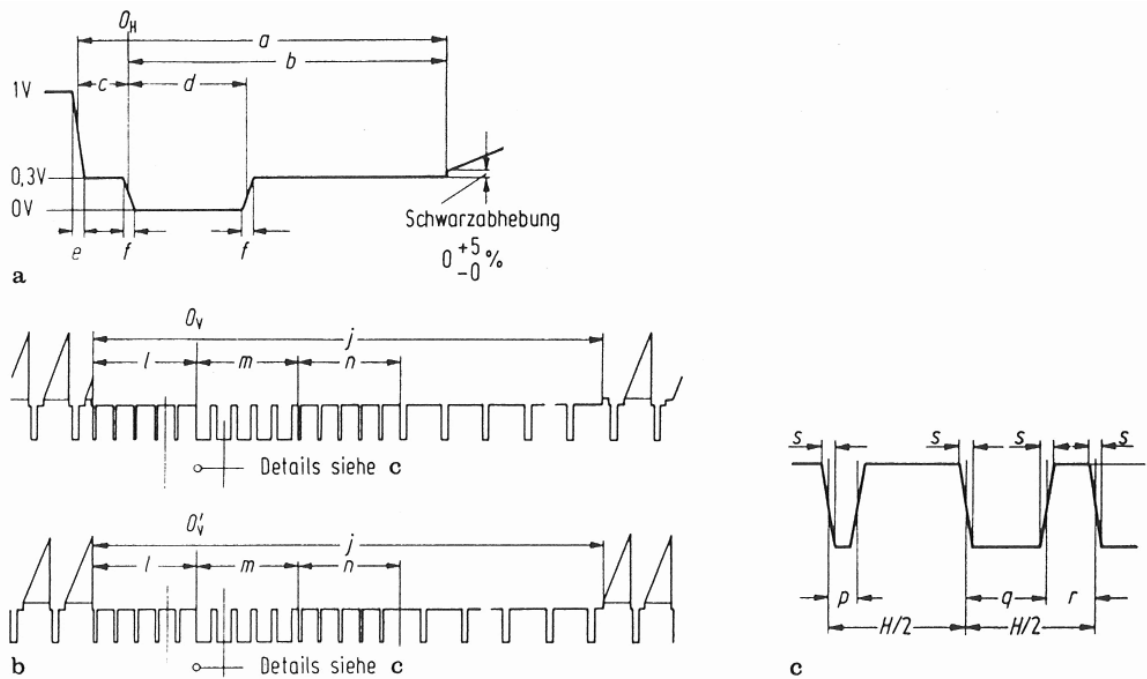
^a Bundesrepublik Deutschland 20 : 1.

Sendeseitige Vorentzerrung der Grup-
penlaufzeit-Charakteristik des Empfängers in μ s
(nach CCIR-Bericht 624-1)

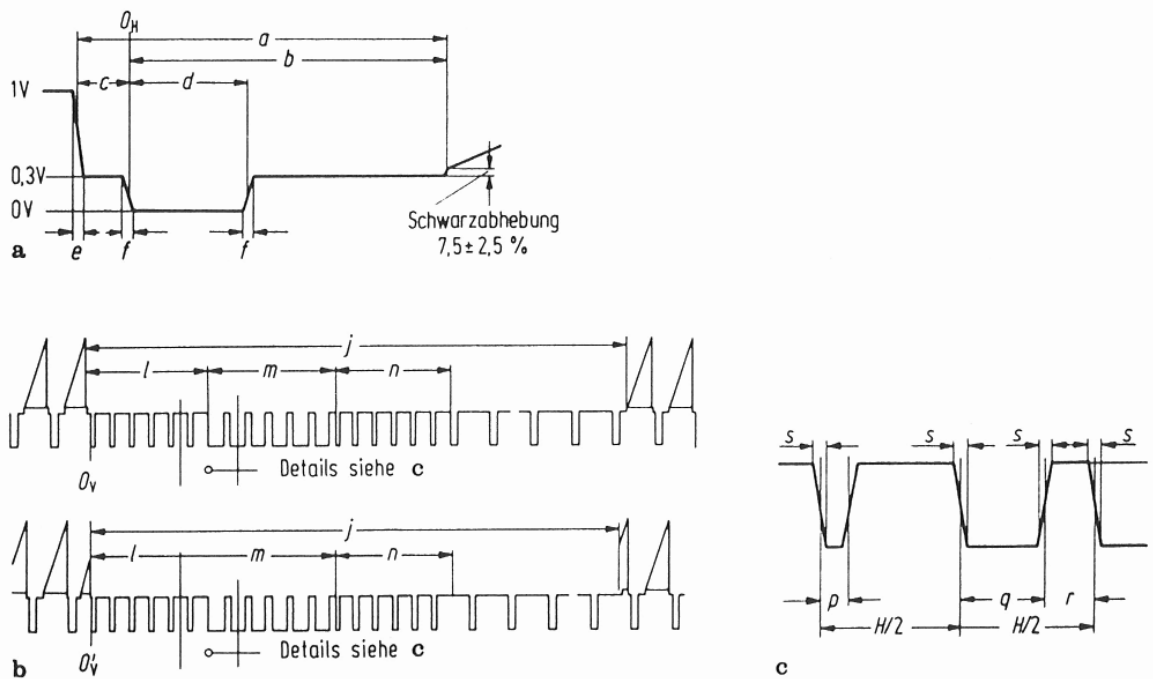
f_{video} in MHz	B/G-Normen		M-Normen (Farbsysteme)
	A ^a	B ^b	
0,25	—	5 \pm 0	0 \pm 100
1,00	30 \pm 50	53 \pm 40	0 \pm 100
2,00	60 \pm 50	90 \pm 40	0 \pm 100
3,00	60 \pm 50	75 \pm 40	0 \pm 60
3,75	0 \pm 50	0 \pm 40	—
3,58	—	—	-170 \pm 30
4,00	—	—	-295 \pm 85
4,43	-170 \pm 35	-170 \pm 40	—
4,80	-260 \pm 75	-400 \pm 90	—

^a Niederlande, Spanien.

^b Bundesrepublik Deutschland und andere Länder.



Gestaltung der Austastlücken bei den 625-Zeilen-Systemen. a) horizontale Austastlücke mit Zeilensynchronzeichen, O_H Zeilenbeginn; b), c) vertikale Austastlücke mit entsprechendem Synchronzeichen (beide Halbraster). O_V Beginn des ersten, O'_V Beginn des zweiten Halbrasters.



Gestaltung der Austastlücken bei den 525-Zeilen-Systemen. a) horizontale Austastlücke mit Zeilensynchronzeichen, O_H Zeilenbeginn; b), c) vertikale Austastlücke mit entsprechenden Synchronzeichen (beide Halbraster). O_V Beginn des ersten, O'_V Beginn des zweiten Halbrasters.

G. Literaturverzeichnis

- [1] CCIR Recommendations and Reports of the XIVth Plenary Assembly, Vol. XI (Television). Genf: UIT 1978 (in Englisch, Französisch u. Spanisch)

- [2] Bernath, K.W.: Grundlagen der Fernseh-System- und Schaltungstechnik, Berlin Heidelberg, New York: Springer 1982

- [3] XILINX Inc.: The Programmable Gate Array, Data Book, San Jose: 1989

- [4] Drescher, Prof. T.C.E.; Reckert, Prof.: Vorlesung zum Fach 'Technik der Datenverarbeitungsanlagen'. Fachhochschule Köln, Abt. Gummersbach 1988

- [5] Drescher, Prof. T.C.E.: Vorlesung zum Fach 'Maschinenorientierte Programmiersprachen'. Fachhochschule Köln, Abt. Gummersbach 1988

- [6] NEC Inc.: Memory Products, Data Book, USA: 1987

- [7] ITT Semiconductors: UVC 3130 High-Speed A/D-D/A Converter, Freiburg: 1988

- [8] GTU Microcircuits: 65SCxx Data Book, Tempe, Arizona 1987

- [9] Poenicke, K.: Die schriftliche Arbeit. 2. Auflage Mannheim/Wien/Zürich: Dudenverlag 1989

H. Sachwortverzeichnis

A

A/D- und D/A-Wandlerplatine, A-28
 A/D-Wandler (ADC), 4-2
 Abkürzungsverzeichnis, 1-1
 Ablauf- und D-RAM-Steuerung (CLK2), 4-3
 Ablaufsteuerung (CLK2), A-3
 Ablaufsteuerung (CLK2/1), A-3
 ADC, 4-1, 4-2, A-28
 Addierer Nr.1 (ADRE/LOGIC/ADD_1/1), A-14
 Addierer Nr.2 (ADRE/LOGIC/ADD_2/1), A-15
 Addierer/Subtrahierer, C-16
 ADDSUB2: 2-Bit-Addierer/Subtrahierer, C-17
 ADDSUB4: 4-Bit-Addierer/Subtrahierer, C-17
 ADDSUB8: 8-Bit-Addierer/Subtrahierer, C-18
 ADRE, 4-1, 4-11
 Adreßmultiplexer, B-6
 COL-Adreßerzeugung, B-5
 CPU-Modus, B-7
 ROW-Adreßerzeugung, B-6
 ADRE - Zeitdiagramme, 4-13, B-5
 Adressierungsschema, 2-3
 Adreßerzeugung (ADRE), A-10
 Adreßerzeugung (ADRE/1), A-10
 Adreßmultiplexer (ADRE), B-6
 Adreßmultiplexer (ADRE/LOGIC/3), 4-13, A-13
 Anzeige, A-30
 Anzeige/Tastaturplatine, A-30
 Anzeigesteuerung, E-11
 ANZNEU: Anzeige mit aktuellen Daten füllen, D-5
 ANZNU: Binärzahl zweistellig darstellen, D-5
 Arbeitsumgebung, 2-1
 ATINIT: Anzeige und Tastatur initialisieren, D-5
 Auflösung, 2-2
 Aufnahmesteuerung, 6-3
 Aufnahmesteuerung (CLK2/LOGIC/3), A-6
 Aufnahmesteuerung (SYNC/LOGIC/4), 4-19, A-22
 Aufnahmesteuerung und Refreshtimer (CLK2/LOGIC/3), 4-8
 Ausblick, 7-1
 Ausgangssignale, 4-4, 4-11, 4-15, 4-21
 Autorouter, 3-2, 6-2

B

Bedienung, 5-1
 Betriebsmodis, 4-1
 BFUELL: Bildspeicher mit Testdaten füllen, D-2
 BIBL.S, E-1
 Bibliothek, E-1
 Bibliothek (BIBL.S), E-1
 Bildspeicher
 Top-Level (EDDI3), A-2
 BIN2BCD: Umwandlung Binär -> BCD, D-10
 Binärzähler, C-6
 Binärzähler modulo 16, C-11
 Binärzähler modulo 256, C-14
 Binärzähler modulo 4, C-6
 Binärzähler modulo 8, C-8

C

C16BCPRDX: Ladbarer Zähler modulo 16
 Direct Reset, C-13
C16BCPRX: Ladbarer Zähler modulo 16
 Synchron Reset, C-12
C16BCRX: Zähler modulo 16
 Synchron Reset, C-11
C256BCPRDX: Ladbarer Zähler modulo 256
 Direct Reset, C-15
C256BCRX: Zähler modulo 256
 Synchron Reset, C-14
C4BCPRDX: Ladbarer Zähler modulo 4
 Direct Reset, C-7
C4BCRDX: Zähler modulo 4
 Direct Reset, C-6
C4BCRX: Zähler modulo 4
 Synchron Reset, C-6
C8BCPR: Ladbarer Zähler modulo 8
 Synchron Reset, C-9
C8BCPRDX: Ladbarer Zähler modulo 8
 Direct Reset, C-10
C8BCRX: Zähler modulo 8
 Synchron Reset, C-8
CAS-Zyklus (CLK2/LOGIC/CASZYK/1), A-8
CAS-Zähler (ADRE/LOGIC/2), A-12
CAS-Zähler (CLK2/LOGIC/2), A-5
CCCAS: COL-Adressen-Bereichsabfrage, D-13
CCRAS: ROW-Adressen-Bereichsabfrage, D-13
CCROW: Zeilennummer-Bereichsabfrage, D-14
CLB, 3-1
CLK2, 4-1, 4-3
 CPU-Modus, B-3
 Differenzierer, B-2
 REC-Modus, B-4
CLK2
 RAS/CAS-Erzeugung, B-2
CLK2 - Zeitdiagramme, 4-8, B-2
COL-Adreßerzeugung (ADRE), B-5
COL-Zähler (ADRE/LOGIC/2), 4-12
Composite-Sync, 4-15
Composite-Sync-Erzeugung (SYNC/LOGIC/3), A-21
Configurable Logic Block (CLB), 3-1
CONIN: Tastatur über Tabelle abfragen, D-8
CPU, 4-1, 4-24, A-29
CPU-Modus
 ADRE, B-7
 CLK2, B-3
 PROZ, B-10
CSync-Erzeugung (SYNC), B-9
CSync-Erzeugung (SYNC/LOGIC/3), 4-18

D

D/A-Wandler (DAC), 4-24
DAC, 4-1, 4-24, A-28
Daisy Workstation, 3-3
Datenblätter, F-1
Datenbusse, 4-20
Datenfluß- und CPU-Steuerung (PROZ), 4-20
Datenflußsteuerung (PROZ/LOGIC/1), 4-21
Datenmultiplexer (PROZ/LOGIC/2), 4-21, A-26
Datenreduktion, 4-23, 7-1

Datensteuerung (PROZ), A-24
 Datenwegesteuerung (PROZ/LOGIC/1), A-25
 DCCAS: Versch. der Bilder um ein Pixel nach rechts, D-12
 DCRAS: Versch. der Bilder um ein Bild nach rechts, D-12
 DCRW1: Versch. der Bilder um eine Zeile nach unten, D-13
 DECCAS: COL-Adresse vermindern, D-10
 DECRAS: ROW-Adresse vermindern, D-11
 DECROW: Zeilennummer vermindern, D-11
 Differenzierer (CLK2), B-2
 Differenzierer (CLK2/LOGIC/2), A-5
 Differenzierer und CAS-Zähler (CLK2/LOGIC/2), 4-7
 DRAM, 4-1, 4-23
 Dynamischer Speicher (DRAM), 4-23

E

Eingangssignale, 4-3, 4-11, 4-15, 4-20
 Einleitung, 2-1
 Einsatzgebiete, 7-1
 Entwicklungswerkzeug PC, 3-2
 Entwicklungswerkzeug Workstation, 3-3
 EPROM, 4-24
 Erzeugung der D-RAM-Adressen (ADRE), 4-11
 Erzeugung der Video-Synchronimpulse (SYNC), 4-15

F

FCSR: Getaktetes RS-Flip-Flop
 Vorrang Rücksetzen, C-3
 FCSR: Getaktetes RS-Flip-Flop
 Vorrang Setzen, C-3
 Flip-Flops, C-3
 Flußdiagramme, D-1
 Flußdiagramme zu IRQ.S, D-4
 Flußdiagramme zu START.S, D-1
 Flußdiagramme zu TAST.S, D-5
 Flußdiagramme zu UNTER.S, D-10
 FREMD.S, E-22
 FTCPR: Ladbares Toggle-Flip-Flop
 Synchron Reset, C-5
 FTCPRD: Ladbares Toggle-Flip-Flop
 Direct Reset, C-5
 FTCCR: Toggle-Flip-Flop
 Synchron Reset, C-4
 FTCD: Toggle-Flip-Flop
 Direct Reset, C-4

H

H-Puls (SYNC), B-9
 HALF_ADD: Halbaddierer, C-16
 Hauptprogramm, E-5
 Hierarchie der Schaltpläne, A-1
 Hierarchischer Schaltplan, 3-3
 Horizontalsteuerung (SYNC/LOGIC/1), 4-16, A-19

I

I/O-Pads, C-16
 ICCAS: Versch. der Bilder um ein Pixel nach links, D-12
 ICRAS: Versch. der Bilder um ein Bild nach links, D-12
 ICRW1: Versch. der Bilder um eine Zeile nach oben, D-13
 ILOZ: I/O-Pad mit Input-Latch und Tri-State-Output, C-16
 In-Circuit-Emulator, 3-5
 INCCAS: COL-Adresse erhöhen, D-10

INCRAS: ROW-Adresse erhöhen, D-11
INCROW: Zeilennummer erhöhen, D-11
Initialisierung, E-5
Initialisierung und Hauptprogramm (START.S), E-5
Input/Output Block (IOB), 3-2
Interrupt-Behandlung, E-10
Interrupt-Behandlung (IRQ.S), E-10
Interrupt-Steuerung (PROZ/LOGIC/3), 4-22
Interruptsteuerung (PROZ/LOGIC/3), A-27
IOB, 3-2
IRQ.S, E-10
IRQA: Routine für maskierbaren Interrupt, D-4

K

Konzeption, 2-2
Kritisches Timing, 6-1

L

LCA, 3-1
LCA-Editor, 3-2
LCA-Makrobibliothek, C-1
LCA-Makrobibliothek (Übersicht), C-2
Leitungen und Busse, 3-4
Literaturverzeichnis, G-1
Logic Cell Array (LCA), 3-1
Low-Level Routinen (FREMD.S), E-22

M

Mikroprozessor, A-29
Mikroprozessor (CPU), 4-24
MKCH: Zeile eines Zeichens aufbauen, D-3
MKZEIL: Testbildzeile aufbauen, D-3
Modus-Umschaltungen, 6-1
Multiplizierer * 24 (ADRE/LOGIC/MUL20/1), A-16
Multiplizierer * 80 (ADRE/LOGIC/MUL80/1), A-17

N

Netzliste, 3-2
NMIA: Routine für nicht maskierbaren Interrupt, D-4

P

Plazierung, 6-2
Prinzip der RAS/CAS-Erzeugung, 4-4
Programmbeschreibung, 5-1
Programmlisting, E-1
PROZ, 4-1, 4-20
 CPU-Modus, B-10
PROZ - Zeitdiagramme, 4-23, B-10
Prozessormodul
 Datensteuerung (PROZ), A-24
Prozessormodul (PROZ/1), A-24

R

RAM, 4-24
RAS-Zyklus (CLK2/LOGIC/RASZYK/1), A-7
RAS-Zähler (ADRE/LOGIC/1), A-11
RAS/CAS-Erzeugung (CLK2), B-2
RAS/CAS-Erzeugung (CLK2/LOGIC/1), 4-7, A-4
REC-Modus
 CLK2, B-4
Referenztafel, E-25

Refresh-Timer (CLK2/LOGIC/3), A-6
 Refresh-Zyklus (CLK2.LOGIC/RFSHZYK/1), A-9
 Routing, 6-2
 ROW-Adreßerzeugung (ADRE), B-6
 ROW-Zähler (ADRE/LOGIC/1), 4-11
 RS-Flip-Flops, C-3

S

Sachwortverzeichnis, H-1
 Schaltpläne, A-1
 Schaltungsbeschreibung, 4-1
 Sonstige Schaltpläne, A-28
 START.S, E-5
 STTA: Reset-Initialisierung, D-1
 Symbole und Makros, 3-3
 SYNC, 4-1, 4-15
 CSync-Erzeugung, B-9
 H-Puls, B-9
 V1-Puls, B-8
 V2-Puls, B-8
 SYNC - Zeitdiagramme, 4-19, B-8
 Sync-Erzeugung (SYNC/1), A-18

T

TASIA: Physikalischen Tastencode ermitteln, D-9
 TASLOW: Low-Level-Tastaturfunktion ausführen (1), D-6
 TASLOW: Low-Level-Tastaturfunktion ausführen (2), D-7
 TAST.S, E-11
 Tastatur, A-30
 Tastatur- und Anzeigesteuerung (TAST.S), E-11
 Tastaturabfrage, E-11
 Technologien, 3-1
 Toggle-Flip-Flops, C-4
 Top-Level (ADRE/1), 4-11, A-10
 Top-Level (CLK2/1), 4-3, A-3
 Top-Level (PROZ/1), 4-20, A-24
 Top-Level (SYNC/1), 4-15, A-18

U

Umrahmung, 6-3
 Unbonded Pad, 3-4
 Ungelöste Probleme, 6-1
 UNTER.S, E-15
 Unterprogrammbibliothek, E-15
 Unterprogramme (UNTER.S), E-15
 uPD424256-12
 256K*4 Bit D-RAM, F-10
 UVC 3130: A/D-D/A-Wandler, F-1

V

V1-Puls (SYNC), B-8
 V2-Puls
 SYNC, B-8
 Verdrahtungskanäle, 3-2
 Vertikalsteuerung (SYNC/LOGIC/2), 4-17, A-20
 Verwendete Hardware, 5-1
 VIA, 4-24
 Video-Basisnormen der 625-Zeilen-Systeme, F-15
 Video-Sync-Erzeugung (SYNC), A-18

W

WAIT: Lange Warteschleife, D-15

Wartezyklus (CLK2/LOGIC/WAITZYK/1), A-7

Wire-Wrap-Technik, 3-5

Workstation, 3-3

WTLAUF: Kurze Warteschleife für Hauptprogramm, D-15

X

XILINX, 3-2

XILINX XC3000-Serie

Zeitverhalten, F-5

XILUP: XILINX-Konfiguration laden, D-14

Z

Zeilenkomparator (SYNC/LOGIC/COMP9/1), A-23

Zeitdiagramme, B-1

Zielsetzung, 2-1

Zusammenfassung, 7-1

Zustandsfolgediagramm, 4-10

Ü

Übersicht, 4-1, C-1

Übersicht über die LCA-Makrobibliothek, C-2

Übersichtsschaltplan (EDDI3/1), A-2

Hinweise zu dieser Ausgabe

Die Diplomarbeit hat der Verfasser 1990 auf einem CP/M plus Computer (mc-CP/M plus Computer, ein System 8000 EPC von oettle+reichler) unter Verwendung von WordStar, MailMerge und StarIndex erstellt. Die Flussdiagramme wurden mit EasyFlow unter MS-DOS gezeichnet, die Schaltpläne und Zeitdiagramme entstanden auf einer DNIX Workstation von Daisy. Die Grafiken wurde manuell im Cut+Paste-Verfahren in den Text integriert, d.h. mit Schere und Klebstoff. Das so entstandene Original wurde mit einem Fotokopierer vervielfältigt.

Die Wordstar-Textdateien sowie die Quelltexte der Programme hat der Verfasser jetzt, fast 20 Jahre nach ihrer Entstehung, von einer alten CP/M-Backup-Diskette auslesen können, sie bildeten die Basis dieser Ausgabe. Alle Grafiken wurden aus einer Kopie der Diplomarbeit von 1990 gescannt. Die Easy-Flow-Grafiken gibt es zwar noch, es konnte aber kein Programm aufgetrieben werden, das sie importieren könnte. Die Daisy-Workstation existiert schon lange nicht mehr, auch keine Kopien der dort entstandenen Dateien.

Aus den vorliegenden Dateien wurde diese Ausgabe der Diplomarbeit rekonstruiert. Dazu wurde sie mit Microsoft Word 2000 neu gesetzt. Sie ist inhaltlich mit der Originalausgabe identisch, auch die Paginierung stimmt überein. Es gibt lediglich kleine Unterschiede in der Formatierung, insbesondere fehlt im Original die durchgehende Titelzeile auf den Seiten, die ausschließlich Grafiken enthalten. Außerdem wurde diese Hinweisseite angefügt.

Die Rechte dieser Diplomarbeit verbleiben beim Verfasser. Eine kommerzielle oder nicht-kommerzielle Verwertung bedarf der ausdrücklichen schriftlichen Zustimmung.

Wuppertal, den 27. Juni 2009

Jürgen Loh