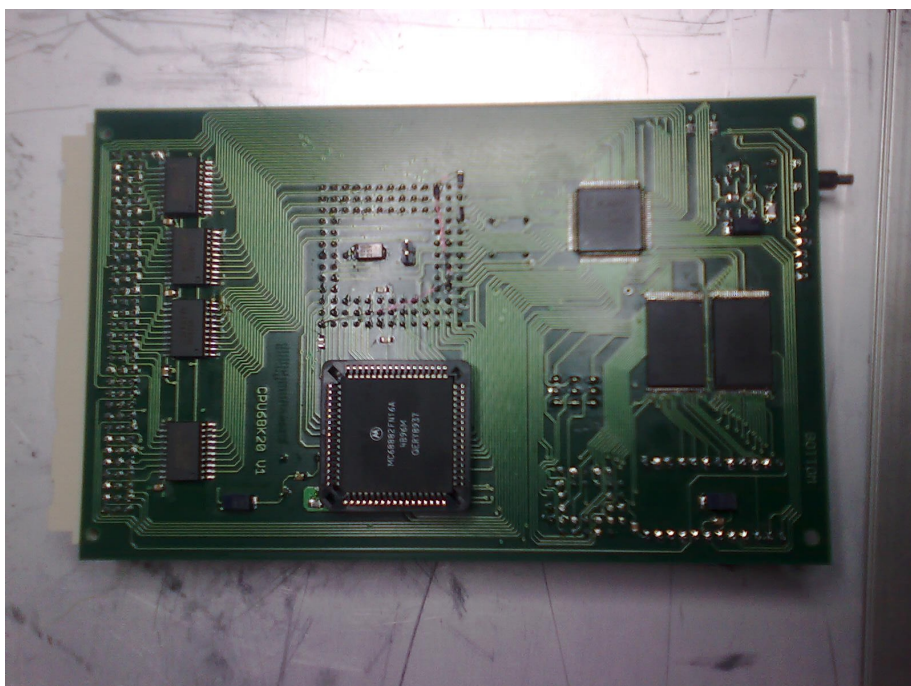
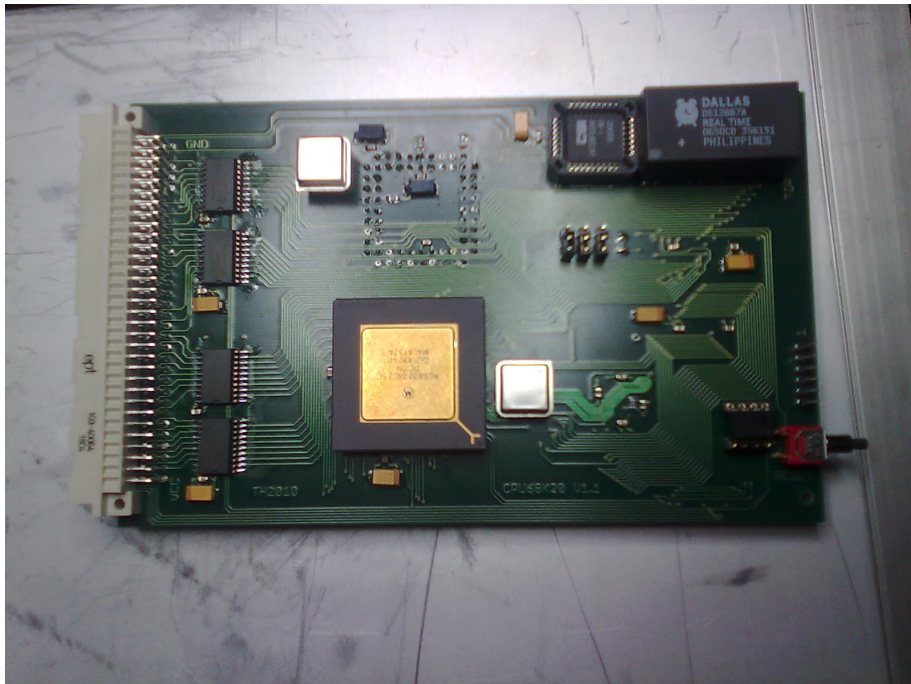


CPU68K020

68020 Baugruppe (nicht nur) für den NDR-Klein-Computer (ECB-Variante)

Stand: Nov 2010

Autor: Torsten Hemmecke



Wichtiger Hinweis:

Die in dieser Anleitung wiedergegebenen Schaltungen und Verfahren werden ohne Rücksicht auf die Patentlage oder Lizenzrechte Dritter mitgeteilt. Sie sind ausschließlich für private Zwecke und Lehrzwecke bestimmt und dürfen nicht gewerblich genutzt werden. *)

Alle Schaltungen und technische Angaben in dieser Anleitung wurden vom Autor sorgfältig erarbeitet bzw. zusammengestellt. Trotzdem sind Fehler nicht auszuschließen. Daher kann der Autor weder eine Garantie noch die juristische Verantwortung oder irgend eine Haftung für Folgen, die auf fehlerhafte Angaben zurückgehen, übernehmen. Für die Mitteilung eventueller Fehler ist der Autor jederzeit dankbar.

Die Rechte an Firmennamen, Logos und Warenzeichen, die in dieser Anleitung genannt werden, liegen bei den jeweiligen Inhabern.

*) Bei gewerblicher Nutzung ist vorher die Genehmigung des möglichen Lizenz- oder Rechteinhabers einzuholen.

Inhalt

1	Vorwort.....	4
2	Änderungshistorie.....	4
3	Kurzbeschreibung der Funktion.....	4
4	Technische Daten.....	4
5	Prinzipbeschreibung.....	5
5.1	Blockschaltbild.....	5
5.2	Beschreibung des Blockschaltbildes.....	5
5.3	Erweiterung des ECB-BUS auf 32 Bit.....	7
5.4	Das CPLD.....	8
5.4.1	Blockschaltbild des CPLD.....	8
5.4.2	Funktionsbeschreibung des CPLDs.....	8
5.4.3	Programmierung des CPLDs.....	10
6	Beschreibung der Jumper.....	11
6.1	EPROM-Offset.....	11
6.2	Cache-Disable.....	11
7	Aufbauanleitung.....	12
7.1	Umgang mit IC's.....	12
7.2	Aufbau Schritt für Schritt.....	12
8	Inbetriebnahme.....	13
8.1	Grundprogramme.....	13
8.1.1	68020 (32-Bit Grundprogramm).....	13
8.2	Betrieb mit CP/M-68K.....	13
8.3	Betrieb mit Jados.....	13
9	Fehlerkorrekturen.....	14
10	Quellenhinweise.....	14
11	Anhänge.....	15
11.1	Schaltplan.....	15
11.2	Stückliste.....	16
11.3	Bestückungsplan.....	17
11.4	Layout Lötseite.....	18
11.5	Layout Bestückungsseite.....	18

1 Vorwort

Die Baugruppe CPU68k020 entstand im Rahmen des Projektes „NKC Redesign“ das zum Ziel hat, den legendären NDR-Klein-Computer mit heute (Stand 2010) erhältlichen Bauelementen neu aufzubauen und bis hin zum Einsatz mit einem kleinen Linux-Kernel. Das ganze System verwendet einen ECB Bus, wobei die Belegung des Busses zum MC-Computer kompatibel ist und darüber hinaus 16/32bit Erweiterungen vorsieht.

Die CPU68k020 soll volle Software-Kompatibilität zum original NKC bieten.

Auf der Karte sind bis zu 4MB Speicher, ein 8bit-ROM für den BOOT-Loader / GP, eine Echtzeit-UHR (kompatibel zur Uhren-Baugruppe UHR3 von Jens Mewes), die komplette Banken-Logik und ein PLCC Sockel für eine FPU (MC68881/68882) integriert. Es steht ein 32-bit ECB-Bus-Interface zur Verfügung.

Zusammen mit der GDP-FPGA von Andreas Voggeneder steht damit ein vollständiges NKC System zur Verfügung, das bereits JADOS über eine SD-CARD bootet.

2 Änderungshistorie

In der vorliegenden Version 1.1.2 sind keine Fehler bekannt.

3 Kurzbeschreibung der Funktion

Die Baugruppe CPU68k020 stellt eine vollständige 32Bit CPU für den NKC dar. Sie enthält zusätzlich zur originalen CPU68020 bereits 4MB RAM, die komplette Banken-Logik sowie die UHR3 Baugruppe und ist voll softwarekompatibel zu diesen Baugruppen. Somit kann sämtliche Software des NKC darauf betrieben werden. Die Hardware ist auf einer Europakarte aufgebaut wird mit einen 32-Bit ECB-Bus-Anschluß betrieben werden.

4 Technische Daten

Europakarte 160 x 100 mm 4 Lagen (2 Supply Planes)
CPU 68020-16/25/33 MHz
FPU 68881/68882
ECB-Bus (32-bit)
Stromaufnahme ca. mA
Gerätetyp; Betriebsspannung 5V
4GB adressierbar
4MB RAM OnBoard
512K FLASH OnBoard
RTC 12(C)887

Die Bus Steuerung sowie DTACK/WAIT Generierung wird im CPLD erledigt. Es können die IRQx, NMI und INT Leitungen des ECB Busses im CPLD beliebig auf die IPL Leitungen der CPU geschaltet werden.

Darüber hinaus kümmert sich das CPLD um die notwendigen Adress-Dekodierungen von IO und MEM Zugriffen (lokal/nicht lokal etc.) und realisiert das BANKEN Register zum Ausblenden des ROM aus dem Adressraum des 68020.

Als Reset Baustein wird ein TLC7725 eingesetzt. Der Baustein sorgt bei Einschalten für einen genügend langen RESET Puls, so dass sich die Versorgungsspannung und die Signale stabilisieren können. Außerdem wird die Versorgungsspannung überwacht und bei Unterschreitung ein Reset ausgelöst um einen undefinierten Zustand des Systems zu vermeiden. Außerdem kann ein externer Reset-Taster angeschlossen werden.

Ein Sockel für eine FPU (68881 od. 68882) ist vorhanden und kann wahlweise bestückt werden.

Das 32Bit ECB Bus Interface stellt die Verbindung zu weiteren Karten über den ECB Bus her. So kann mit einer SRAM_24MB der verfügbare Speicher auf 28MB erweitert werden. Damit kann ohne Probleme ein uCLinux System betrieben werden. Der Zugriff auf die externe Speicherkarte (SRAM_24MB) ist mit einem 68020-25MHz getestet und benötigt keine weiteren WAIT-States.

5.3 Erweiterung des ECB-BUS auf 32 Bit

Damit der 68020 auch mit voller Bus-Breite auf Peripherie zugreifen kann, wurde der Bus auf 32-Bit erweitert. Er bleibt dabei Kompatibel zum Z80-Bus (siehe Baugruppe SBC4) und zum 16Bit Bus des 68000 (siehe Baugruppe CPU68K).

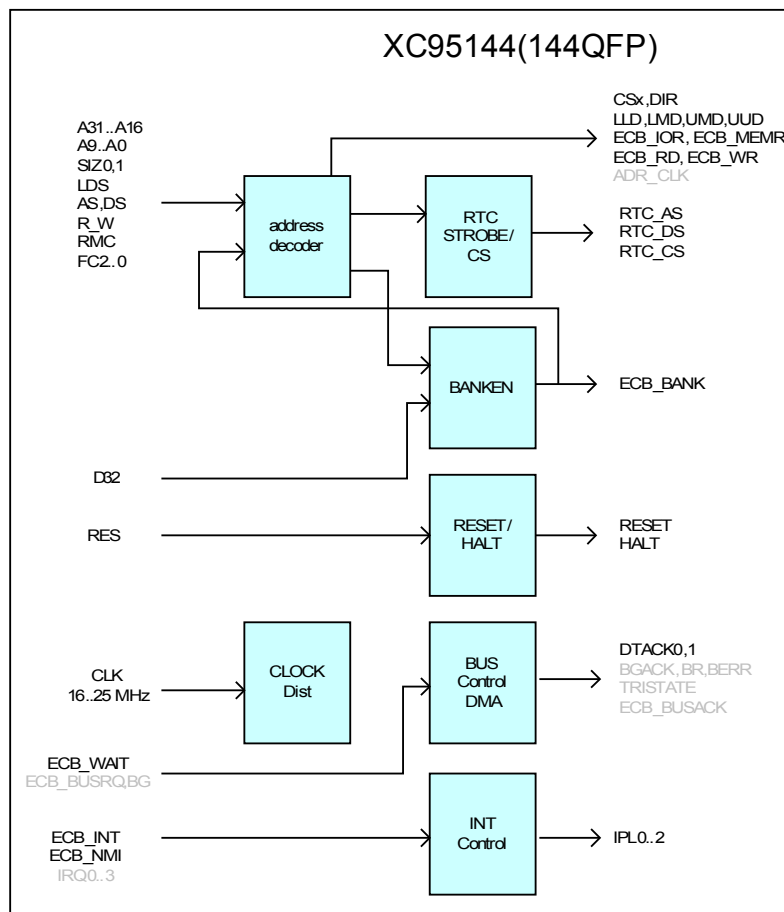
Reihe	A	B	C
1	5V	5V	5V
2	D5	A20	D0
3	D6	A21	D7
4	D3	A22	D2
5	D4	A23	A0
6	A2	D8	A3
7	A4	D9	A1
8	A5	D10	A8
9	A6	D11	A7
10	WAIT\	D12	A24
11	BUSRQ\	D13	A25
12	A18	D14	A19
13		D15	BANKEN
14	A26	D24	D1
15	(020-A0)	D25	A27
16	(020-A1)	D26	A28
17	A17	IRQ4	A11
18	A14	IRQ3	A10
19	A29	IRQ2	A16
20	M1\	IRQ1	NMI\
21	INT\	IRQ0	
22	D27	D16	WR\
23	D28	D17	
24	D29	D18	RD\
25	D30	D19	HALT/
26	D31	D20	
27	IORQ\	D21	A12
28		DS0/UDS/SIZ0	A15
29	A13	DS1/LDS/SIZ1	CLK
30	A9	D22	MREQ\
31	BUSAK\	D23	RESET\
32	GND	GND	GND

GRÜN = NKC-ECB-BUS Belegung
 ORANGE = 16Bit Erweiterung (68000)
 VIOLETT = 32Bit Erweiterung (68020 – asynchrones Bus Design)
 HBLAU = evtl. für synchrones Bus-Design (ab 68040) vorgesehen

5.4 Das CPLD

Im Folgenden wird im Wesentlichen das CPLD in der NKC Version beschrieben. Prinzipiell kann das Verhalten der CPU-Baugruppe auch an andere Anwendungsfälle angepasst werden, z.B. kann das Interrupthandling verfeinert und die BUS Arbitration für die Verwendung mehrerer CPUs angepasst werden. Auch die dekodierten Adressbereiche sind hier rekonfigurierbar.

5.4.1 Blockschaltbild des CPLD



Signale, die im Standard NKC Design nicht verwendet werden sind ausgegraut.

5.4.2 Funktionsbeschreibung des CPLDs

Das VHDL File definiert zunächst die Signale, die der Baustein nach Außen hin zur Verfügung stellt. Danach werden einige Konstanten definiert, die eine Anpassung/Änderung von Speicherbereichen und Register-Adressen erleichtern und das ganze Listing lesbarer machen.

Danach folgen einige Prozesse und ein wenig kombinatorische Logik:

Process STROBE:

Dient dem Erzeugen der für den Uhrenbaustein notwendigen Strobe Signale für Read und Write Zugriffe. Der Uhrenbaustein wird mit Motorola-Timing betrieben.

Process RESETPROC:

Dieser Prozess realisiert ein Tristate Gate, das beim Anlegen eines Reset Signals die Signale HALT und RESET des Prozessors auf NULL zieht. Die Verwendung als Tristate ist notwendig, da der Prozessor selbst diese Signale über Kommandos setzen kann.

Process BANKPROC

Realisiert das BANKEN Register.

Die Adress-Dekodierung wird über mehrere interne Signale generiert. Diese Art der Dekodierung gegenüber einer Switch-Anweisung oder If-then-else Verzweigung ist weniger fehleranfällig und besser lesbar.

Zunächst wird festgestellt, ob ein gültiger Speicherzugriff der CPU vorliegt und das Signal *is_68k_mem_cycle* gesetzt. Die Signale *is_io_cs* und *is_mem_cs* unterscheiden weiter zwischen IO und Speicher-Zugriff (dieses Konzept ist aus historischen Gründen für den NKC so realisiert, obgleich der 68000 keinen Unterschied zwischen IO und MEM Zugriff kennt).

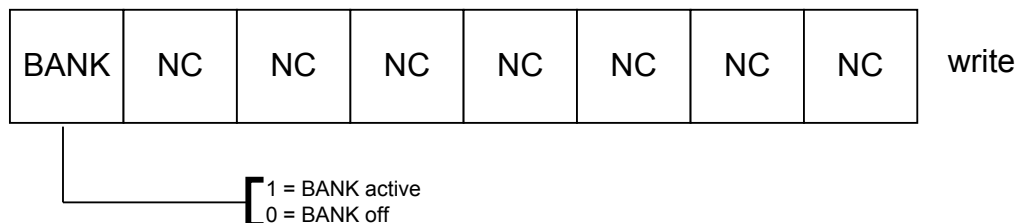
Falls ein IO Zugriff vorliegt wird überprüft, ob es ein lokaler Zugriff auf das BANKEN-Register oder die RTC ist (*rtc_data_cs*, *rtc_addr_cs*, *banken_cs*) oder ob es sich um einen Zugriff auf eine externe IO-Karte handelt (*ext_io_cs*).

Bei MEM Zugriffen wird je nach den Jumper-Einstellungen (siehe) und dem Zustand des BANKEN Registers das lokale RAM, ROM oder externer Speicher selektiert (*ram_cs*, *rom_cs*, *ext_mem_cs*). Bei lokalen Zugriffen werden die Chip-Select Signale entsprechend erzeugt, bei externen Zugriffen werden die Signale entsprechend auf den ECB BUS geschaltet.

Das BANKEN Register ist so realisiert wie im NKC. Es dient hier, anders als der Name vermuten lässt, nicht dem Banking sondern dem Ein- und Ausblenden des ROM Bausteins aus dem Speicherbereich des 68020. Der Namen BANKEN Register stammt aus der Verwendung als BANK-Umschaltregister im Z80 System.

Zu beachten ist, dass - anders als im original NKC - dieses Register hier auf 0xC9 gelegt wurde um nicht evtl. mit dem GP zu kollidieren. Das kann man bei Bedarf im CPLD ändern.

0xC9 – BANK register



5.4.3 Programmierung des CPLDs

Zum Programmieren des CPLDs benötigt man das ISE-Webpack von Xilinx, dieses kann man (nach kostenloser Registrierung) von der Webseite www.xilinx.com herunterladen. Weiterhin benötigt man einen Programmieradapter, hierfür kann z.B. die Schaltung von www.holger-klabunde.de verwendet. Der Programmierstecker auf der GIDE Platine hat die Belegung dieses Programmers. Die CPLD-Daten sind in der Datei Xilinx.zip, diese entpackt man am besten nach c:\xilinx und nicht nach Eigene Dateien, da versch. Versionen des ISE-Webpacks nicht mit Leerzeichen in Verzeichnisnamen klar kommen. Zum Programmieren startet man den (Xilinx) Projekt Navigator und öffnet die Datei CPU68K20.ise, dort startet man unter Processes Implement Design -> Generate Programming File – Configure Device (iMPACT), iMPACT das eigentliche Programmier Tool.

Bis zur Version 9.1.x ist ispIMPACT als Stand-Alone-Version vorhanden, das sich auch ohne ISE installieren lässt. Falls man also nur die fertigen JEDEC Files in's CPLD brutzeln möchte, eine echte Alternative.

6 Beschreibung der Jumper

Im Folgenden werden die Einstellungsmöglichkeiten der Jumper beschrieben. Grundlage dabei ist das Default NKC-CPLD Design.

- x bedeutet Jumper gesetzt
- bedeutet Jumper offen

6.1 EPROM-Offset

Der eingesetzte FLASH Baustein stellt max. 512K Speicher bereit. In der Grund-Konfiguration des CPLD werden davon nur 64K für den NKC Betrieb verwendet. Sollen für andere Anwendungen die ganzen 512K verwendet werden, muss das CPLD entsprechend angepasst werden. Über die Jumper J2,3,4 können verschiedene Bereiche innerhalb des FLASH Bausteines in die ersten 64K des NKC Speicherbereiches eingeblendet werden. Damit können mehrere GP Versionen oder eigene Programme gleichzeitig in einem FLASH Baustein gehalten werden.

J4	J3	J2	EPROM-Offset
A18	A17	A16	\$0.0000
A18	A17	1	\$1.0000
A18	1	A16	\$2.0000
A18	1	1	\$3.0000
1	A17	A16	\$4.0000
1	A17	1	\$5.0000
1	1	A16	\$6.0000
1	1	1	\$7.0000

Beispiel:

Es sollen 2 GP Versionen über Jumper ausgewählt werden können. Dazu wird GP1 ab \$00000 im EPROM abgelegt und GP2 ab \$10000. Wenn alle Adresssignale an das EPROM geschaltet werden, bootet GP1. Wird J2 auf ,1' ge-jumpert, bootet dagegen GP2.

6.2 Cache-Disable

Wenn JP100 (/CDIS) gesteckt ist der interne Befehls-Cache abgeschaltet (unabhängig vom Wert in CACR).

7 Aufbauanleitung

Schaltplan, Layout, Stückliste und Bestückungspläne sind im Anhang bzw. in den PCB-Daten zu finden.

Damit die Datenleitungen möglichst kurz werden, wurde die FPU auf die Unterseite der Platine platziert.

7.1 Umgang mit IC's

CMOS-Bausteine sind hochempfindlich gegen elektrostatische Aufladung! Bewahren oder Transportieren Sie CMOS-Bausteine nur auf leitenden Schaumstoff! Alle Pins müssen kurzgeschlossen sein.

Achten Sie darauf, dass Sie Verbindung mit einer Erdungsmöglichkeit haben, bevor Sie mit diesen Bausteinen arbeiten. Geeignete ESD-Artikel gibt es im Fachhandel.

7.2 Aufbau Schritt für Schritt

Beim Layout der Platine wurde Wert auf eine möglichst hohe Nachbausicherheit gelegt. Dennoch war es notwendig TSSOP Bausteine zu verwenden, um die hohe Speicherdichte realisieren zu können. Daher muss ausdrücklich darauf hingewiesen werden, dass das Einlöten gerade bei diesen Bausteinen nur mit viel Erfahrung gelingen wird. Aufgrund der deutlich höheren Störsicherheit und der Anzahl integrierter Funktionen (UHR, Speicher etc.) ist ein 4-Lagen Design mit 0,2mm Strukturgröße notwendig geworden. Daher kann die Platine i.d.R. nicht vom Hobbyisten gefertigt werden.

Zunächst sollten die schwierigsten Bauteile eingelötet werden (IC1,11,15). Danach kommen die Treiber ICs und alle SMD Bauteile an die Reihe. Bevor der Sockel für den 68020 eingelötet wird darauf achten, dass der Jumper für CDIS und die anderen SMD Bauteile unter dem Sockel sauber eingelötet sind. Dann können der PGA Sockel und der PLCC Sockel (FPU) eingelötet werden. Danach werden alle IC Sockel, der VG Stecker eingelötet sowie alle Jumper und der Programmierstecker für das PLCC eingelötet.

Sind alle Sockel und fest einzulötenden Bauteile eingebaut, kann ein erster Test der Baugruppe erfolgen. Zunächst sollte geprüft werden, dass es keinen Kurzschluss zwischen 0 und 5V gibt (Ohmmeter). Dann 5V an die entsprechenden Klemmen des VG Steckers anlegen und überprüfen, ob irgendwelche Bauteile zu warm werden. Sollte dies der Fall sein, müssen die eingelöteten Bauteile überprüft werden (richtig herum eingelötet, korrekte Werte etc.).

Danach können alle gesockelten Bauteile aufgesteckt und der Test wiederholt werden.

Zuletzt muss noch das CPLD mit dem JEDEC File programmiert werden. Das kann entweder im System oder wieder durch Anschluss an eine 5V Stromversorgung geschehen.

8 Inbetriebnahme

8.1 Grundprogramme

8.1.1 68020 (32-Bit Grundprogramm)

Wenn das GP als bin File vorliegt kann es in das EPROM gebrannt werden.

Mit eingesetztem EPROM und Standard Einstellungen der Jumper sollte jetzt das GP Booten und das Menü des GP auf dem Bildschirm erscheinen.



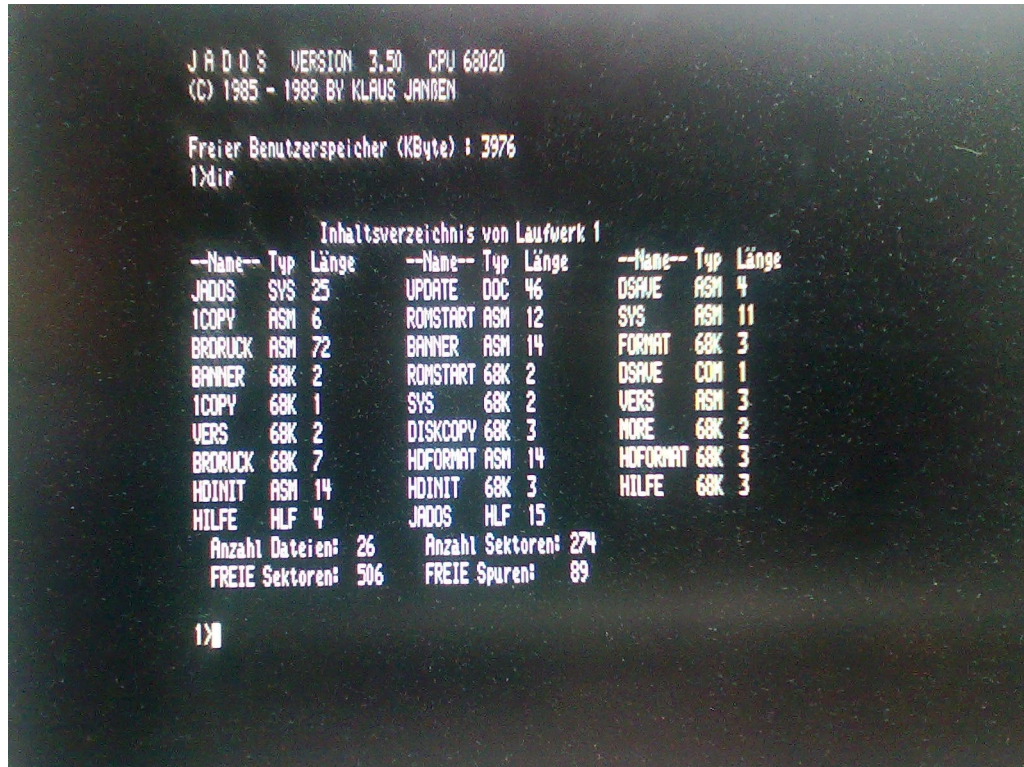
Im Bild ist das GP 6.3 abgebildet.

8.2 Betrieb mit CP/M-68K

Zum jetzigen Zeitpunkt (Stand Nov 2010) gibt es noch keine bootfähige Version für CPM/68K für den 6802

8.3 Betrieb mit Jados

Jados kann von GP aus über die Floppy (FLO-HD muss vorhanden sein) oder von der SD-CARD (ab GP Version 7.x) gebootet werden. Eine Anpassung an den 68020 ist nicht notwendig. Nach dem Booten sollte die Eingabeaufforderung zu sehen sein:



9 Fehlerkorrekturen

In der vorliegenden Version 1.1.2 sind bisher nur ein Problem bekannt:

Bei Betrieb des 68020 mit mehr als 16MHz läuft die FLOHD Baugruppe nicht sauber, es gibt da noch ein Timingproblem. Allerdings ist noch nicht ganz klar, ob das an der CPU oder der FLOHD Karte liegt.

Beim Aufbau sollte die Adresse des Banken-Registers im eingesetzten CPLD und der Software überprüft werden.

Da dieses Register im Grunde nur dazu verwendet wird, beim Boot-Vorgang das GP in den hohen Speicherbereich zu laden und dann das ROM auszublenden, wurde es auf 0xC9 gesetzt um mit Funktionen im GP nicht in Konflikt zu geraten

10 Quellenhinweise

M68020 User's Manual MOTOROLA
M68020 Addendum FREESCALE
CPU68020 / Graf Elektronik
UHR3 Jens Mewes 2007

1992
2003
Ausgabe 3

DS12C887 RTC Datasheet

DALLAS 020900

Die Bank Boot Baugruppe / GES

BOOTRAM68k Gerald Ebert 2007

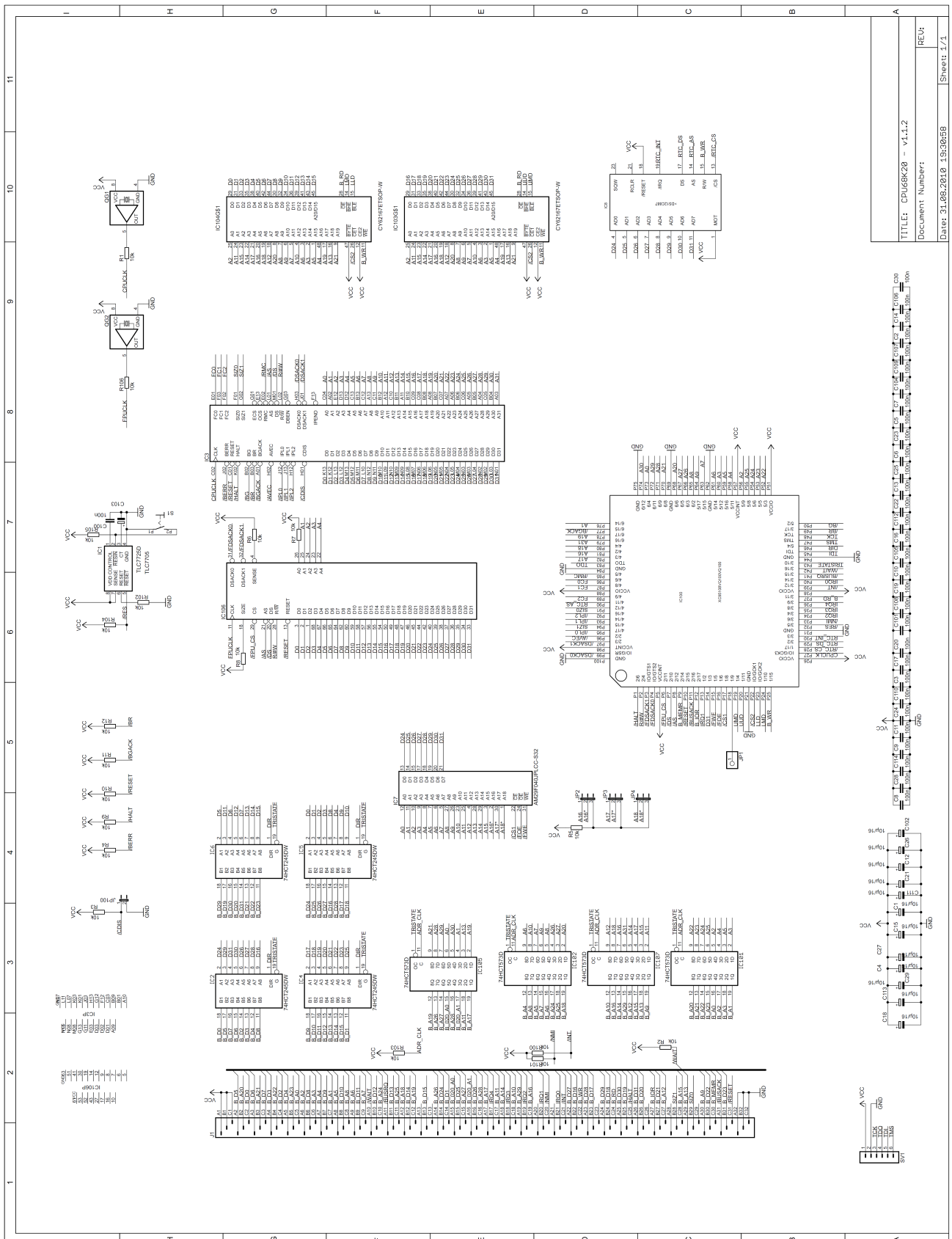
Handbuch Grundprogramm V 7.0 Jens Mewes 2007

Handbuch Jados 3.50 Klaus Janßen 1985-1990

<http://www.schuetz.thtec.org/index.html>

<http://www.drcrazy.de/nkc>

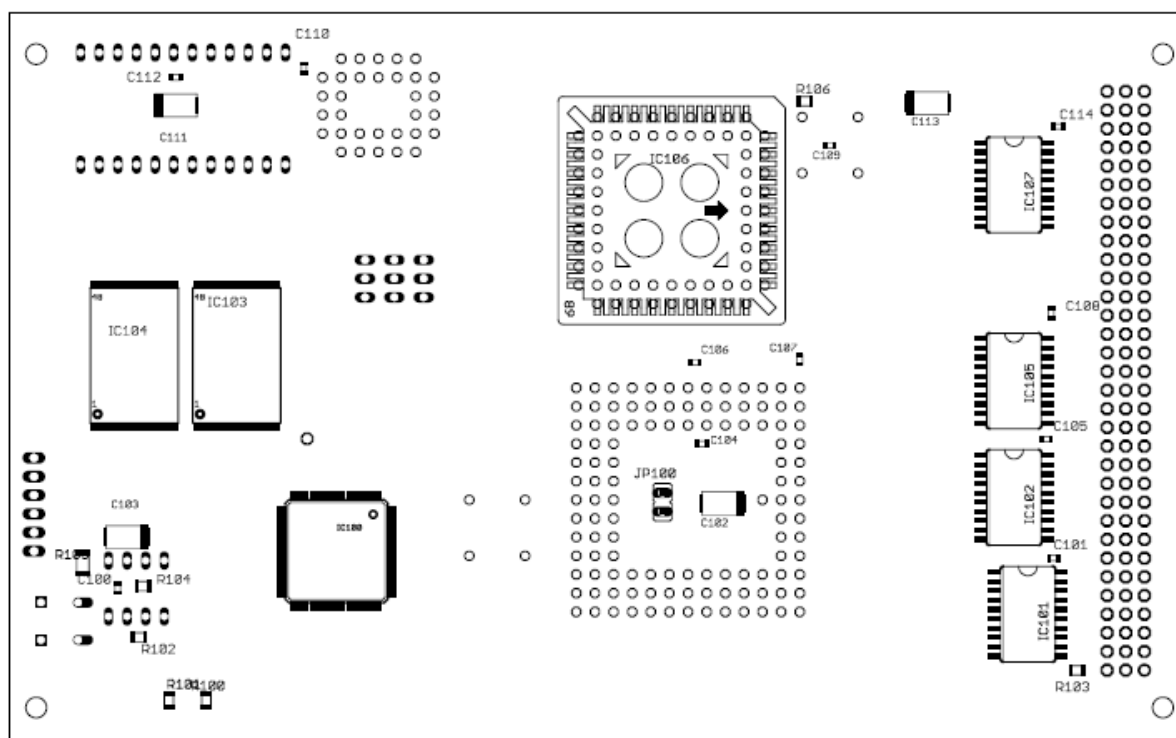
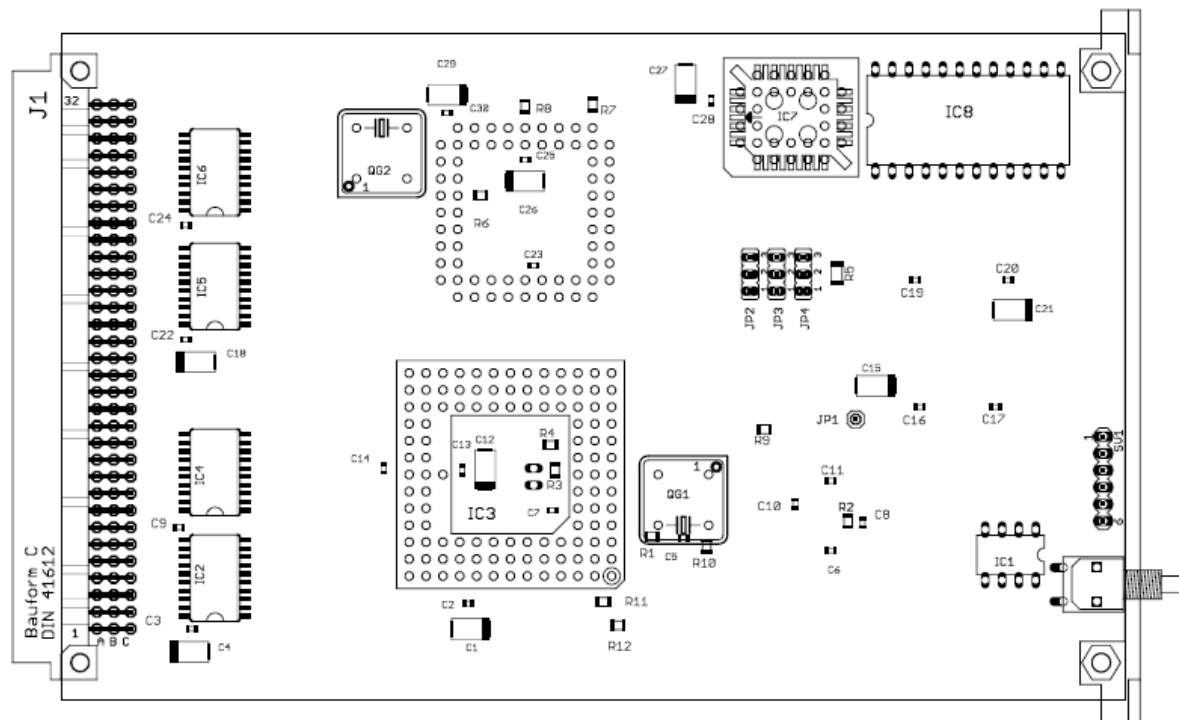
11.1 Schaltplan



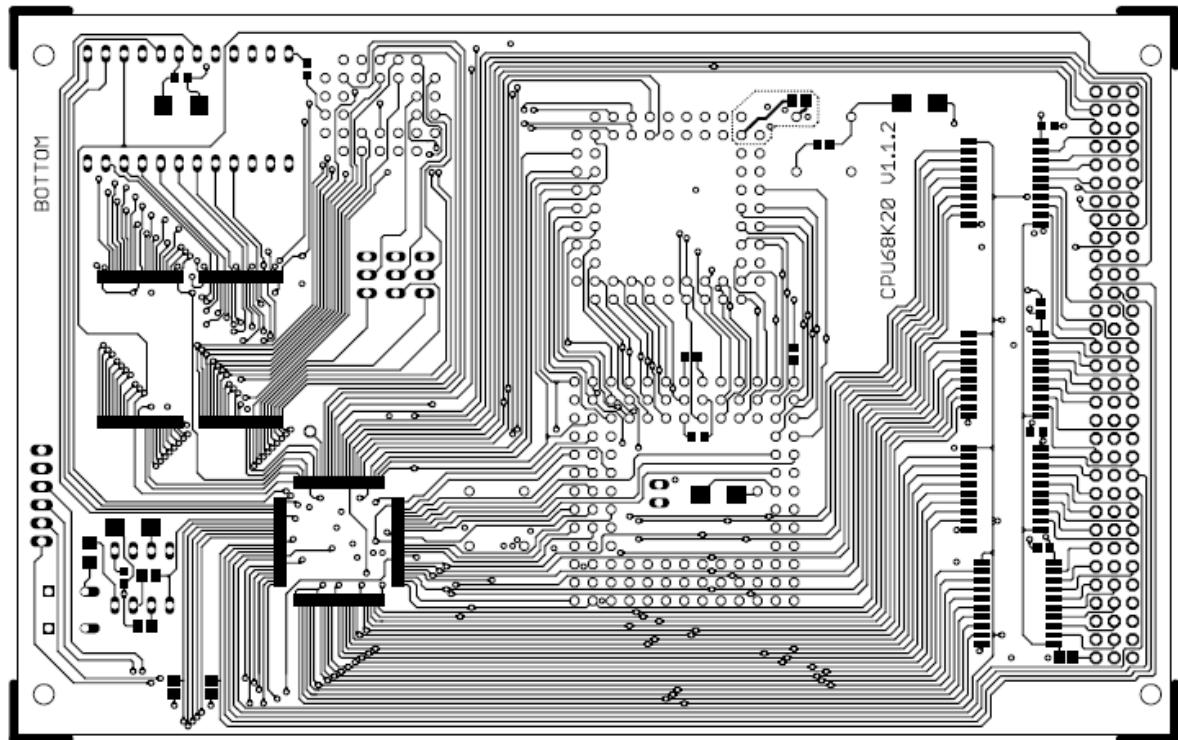
11.2 Stückliste

Menge	Wert	Device	Bauteile
3		JP2E	JP2, JP3, JP4
1		MA06-1	SV1
1		MC68020RC	IC3
1		MC68881FN-SOC	IC106
1		PINHD-1X1	JP1
2		QG5460	QG1, QG2
1		TS-21A	S1
1		VG96P	J1
13	10µ/16	CPOL-EUSMCC	C1, C4, C12, C15, C18, C21, C26, C27, C29, C102, C103, C111, C113
17	10k	R-EU_M0805	R1, R2, R3, R4, R6, R7, R8, R9, R10, R11, R12, R100, R101, R102, R103, R104, R106
2	10k	R-EU_M1206	R5, R105
4	74HCT245DW	74HCT245DW	IC2, IC4, IC5, IC6
4	74HCT573D	74HCT573D	IC101, IC102, IC105, IC107
32	100n	C-EUC0603K	C2, C3, C5, C6, C7, C8, C9, C10, C11, C13, C14, C16, C17, C19, C20, C22, C23, C24, C25, C28, C30, C100, C101, C104, C105, C106, C107, C108, C109, C110, C112, C114
1	AM29F040JPLCC-S32	AM29F040JPLCC-S32	IC7
1	CDIS	JP1E	JP100
2	CY62167ETSOP-W	CY62167ETSOP-W	IC103, IC104
1	DS12C887DIL24	DS12C887DIL24	IC8
1	TLC7725D	TLC7725D	IC1
1	XC95108VQ100VQ100	XC95108VQ100VQ100	IC100

11.3 Bestückungsplan



11.4 Layout Lötseite



11.5 Layout Bestückungsseite

