

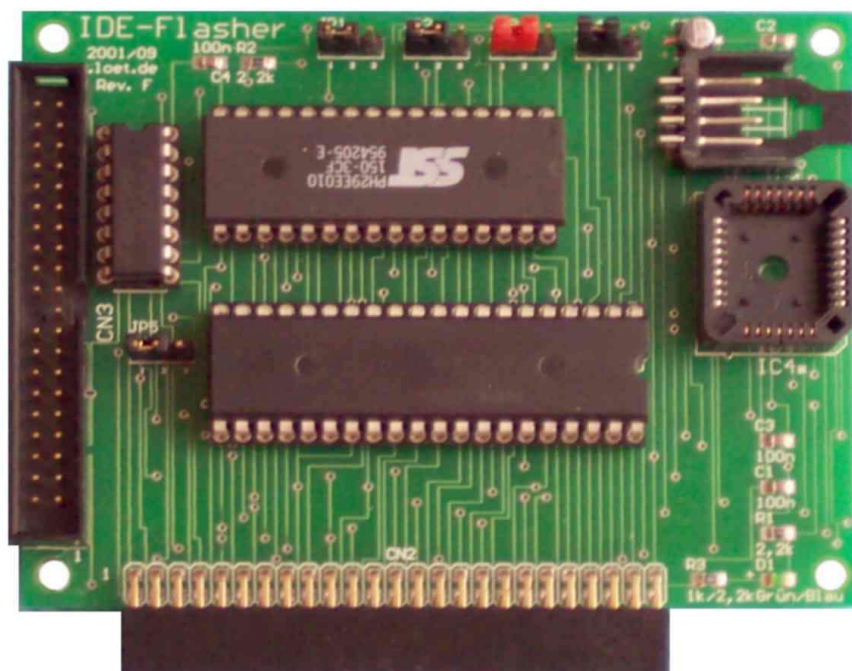
Handbuch

für den

IDE-Flasher Rev. F

Wichtiger Hinweis!!!

Low-Voltage, LPC oder FWH FlashROMs (Betriebsspannung 2,7V-3,6V) können **nicht ohne Spannungsadapter** mit dem *IDE-Flasher* programmiert werden. Bevor ein FlashROM programmiert werden soll, ist es notwendig zu wissen, ob er auch mit 5V Betriebsspannung arbeiten kann. Die Bezeichnung dieser Bausteine beinhaltet meist die Buchstabenkombination **LV, LF, LE oder 82802**. Das ist zumindest ein guter Hinweis, ob es sich um einen LV, LPC oder FWH FlashROM handeln könnte. Sollten Unklarheiten bestehen, hilft die Liste der unterstützten und nicht unterstützten Bausteine auf der LOET.DE Webseite weiter. Vor dem Einsetzen des ICs ist unbedingt zu prüfen, ob der IC mit 5V Betriebsspannung arbeiten kann. Andernfalls besteht sogar die Möglichkeit der Zerstörung des FlashROMs.



Letzte Änderung: 28/01/2004 20:22:15 Version 1.01

Copyright (C) 2003, 2004 LOET.DE

Dieses Dokument dient als ein Nachschlagematerial. Trotz großer Sorgfalt ist es dennoch möglich, dass sich darin Fehler befinden können. Es wird daher keine Garantie übernommen, dass alle Informationen korrekt sind. Insbesondere wird jede Haftung für falsche Informationen ausgeschlossen.

Inhalt:

1. Einleitung.....	5
2. Bestückungshinweise.....	6
2.1 Zusammenbau.....	6
2.2 Fehlersuche.....	8
3. Jumperbelegung.....	10
3.1 Technische Hintergründe.....	10
3.2 Jumperstellungen.....	11
4. Benutzung.....	13
5. Software.....	16
5.1 Arbeiten mit FlashROMs.....	16
5.2 Auslesen eines EPROMs.....	16
5.3 Auslesen der Hardware ID.....	17
5.4 Konfigurationsdatei.....	17
6. Liste der FlashROMs.....	18
6.1 Sortiert nach Kapazität.....	19
6.2 Sortiert nach Hersteller.....	21
7. Anlagen.....	23
7.1 Pinbelegung des IDE-Steckverbinders.....	23
7.2 Pinbelegung der PIO 82C55.....	24
7.3 Schaltplan.....	25
8. Erweiterungen und Programmierung.....	26
8.1 Eigene Erweiterungsmodule.....	26
8.2 Programmierung.....	26
9. Glossar.....	27
10. Quellennachweis.....	29
11. FAQ.....	30
12. Kontakt.....	31

1. Einleitung

Der *IDE-Flasher* ist eine allgemeine Hardware zum Beschreiben von FlashROMs. Darüber hinaus kann sie auch als preiswerte Multi-IO Karte und zum Auslesen von EPROMs verwendet werden. Es ist *kein ISA* oder *PCI Steckplatz notwendig*. Die Schaltung wird *einfach* an einem freien *IDE-Port* angeschlossen.

Mitunter passiert es, dass beim BIOS-Update etwas schief geht, die Telefonanlage oder die Netzwerkkarte eine neue Software braucht, dann ist es schön, wenn eine preisgünstige Lösung greifbar ist. Die hier vorgestellte Lösung eignet sich sogar als Nachbau auf einer Lochrasterplatine. Der Zeitaufwand für den Aufbau hält sich mit eins bis fünf Stunden beschränkt.

Der *IDE-Flasher* eignet sich zum Beschreiben von 32poligen Flash-ROMs mit 5V Betriebsspannung und 5V/12V Programmierspannung. Darüber hinaus können auch 28- und 32-polige EPROMs ausgelesen werden.

Die hier vorgestellte Schaltung ist ein Kompromiss zwischen einfachem und billigem Aufbau und trotzdem möglichst großem Anwendungsbereich.

Die Benutzung der Soft- und Hardware erfolgt auf eigenes Risiko. Seit Dezember 2000 wird die Schaltung verwendet. Das Handbuch wird weiter überarbeitet. Wir freuen uns daher über alle Hinweise und Fragen.

Vielen Dank an Keith für seine englische Übersetzung.

Andreas Ziermann – LOET.DE
18. Oktober 2003

2. Bestückungshinweise

2.1 Zusammenbau

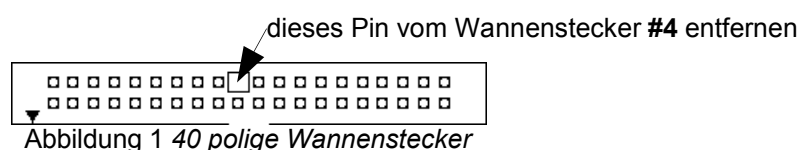
Jeder der den *IDE-Flasher* nicht erst zusammenbauen muss, kann dieses Kapitel getrost überspringen. Bevor man mit dem Löten beginnt, sind die Durchkontaktierungen unbedingt noch einmal zu testen. Insbesondere empfiehlt es sich an den Stellen, die später durch Bauelemente verdeckt werden. Herstellungsbedingte fehlerhafte Durchkontaktierungen sind in seltenen Fällen möglich.

Hinweis: Für das Bestücken der SMD Bauteile ist eine geeignete Lötstation oder FeinlötKolben mit kleiner Lötspitze und feines Lötzinn \varnothing 0,5 – 1,0 mm notwendig. Ansonsten wird mit dem LötKolben Pin und Lötauge erhitzt und anschließend vorsichtig, nicht zuviel, Lötzinn dazugegeben. Es sollte dann das Lötauge von allein umfließen.

Zuerst wird die Platine mit den SMD Bauteilen bestückt. Dazu wählt man den 1k Ω Widerstand **#13** aus, lötet ihn auf den Platz R3 (Abb. 6 auf Seite 16) und dann die grüne SMD-LED **#12**. Auf der Rückseite der LED ist ein kleines Dreieck. Es zeigt von + nach Masse.

Nachdem der Stromversorgungssteckverbinder **#16** aufgelötet wurde, werden die aufgelöteten Komponenten auf Kurzschlüsse getestet. Danach kann die Platine zum ersten mal angeschlossen werden. Die LED sollte jetzt ein deutlich vernehmbares grünes Licht ausstrahlen.

Das nächste Element mit einer speziellen Ausrichtung ist der SMD Elko **#11**. Die schwarze Markierung gehört an das + Zeichen. Die verbliebenen SMD Bauteile **#14** und **#15** haben keine besondere Ausrichtung. Abschliessend wird noch einmal auf Kurzschlüsse getestet.



Bei vielen IDE-Kabeln ist das Pin 20 verschlossen. Dies soll verhindern, dass das Kabel versehentlich falsch herum eingesteckt wird. Um Probleme mit dieser Art Kabel zu vermeiden, kann am Wannenstecker **#4** auch das Pin 20 entfernt werden.

Die Reihenfolge der anderen Komponenten (**# 1, 2, 3, 4, 6, 8, 10**) bereitet keine Schwierigkeiten, wohl aber ist auf die Ausrichtung der Fassungen und Steckverbinder zu achten. Jede Fassung hat eine Markierung, um die Ausrichtung zu verdeutlichen. Auf der Platine wird diese Richtung bei jeder Fassung durch ein * markiert. Die Markierung des Wannensteckers ▼ wird an '1' der Platine ausgerichtet. Die Buchsenleiste wird wie die anderen Bauelemente auf die Bestückungsseite gesteckt, nicht zu tief, so dass die Pins nicht auf der anderen Seite so weit herausragen. Die Abbildung 6 auf Seite 16 zeigt die Anordnung der Bauelemente auf der

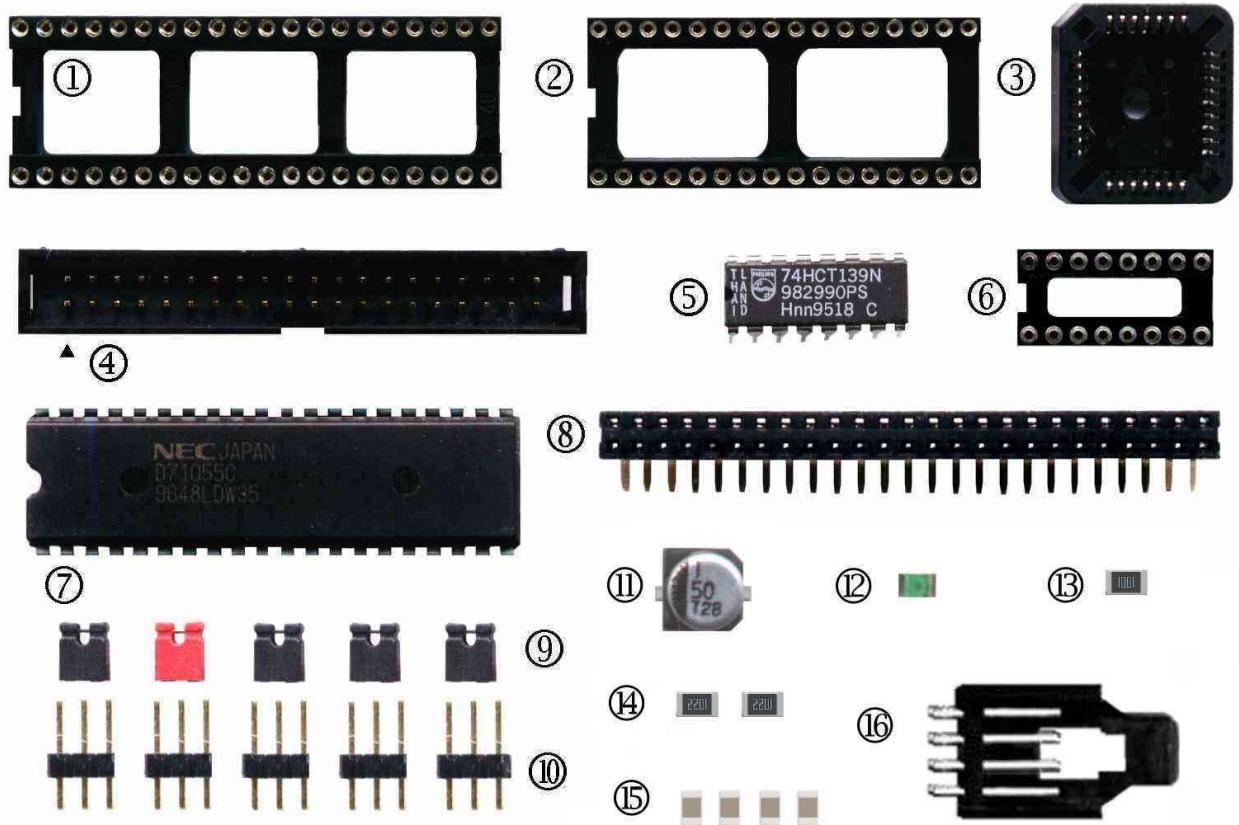


Abbildung 2 verwendete Bauteile

SMD Bauformen 1206 und 0805 möglich

lfd.Nr.	St.	Bezeichnung	lfd.Nr.	St.	Bezeichnung
1	1	Präzisionsfassung DIP40	9	5	Jumper (4 schwarz, 1 rot)
2	1	Präzisionsfassung DIP32	10	5	Stiflleiste 1x3
3	1	Fassung PLCC32	11	1	SMD Elko 1µF/16V
4	1	Wannenstecker 40polig, gerade	12	1	SMD LED grün
5	1	74HCT139 Dekoder DIL	13	1	SMD Widerstand 1kΩ
6	1	Präzisionsfassung DIP16	14	2	SMD Widerstand 2,2kΩ
7	1	82C55 PIO	15	4	SMD Kondensator 100nF
8	1	Buchsenleiste 2x25 gewinkelt (SCSI)	16	1	Platinensteckverbinder 4polig

Tabelle 1 Stückliste

Platine. Der weitere Aufbau und Test erfolgt wie gewohnt. Der erste Test erfolgt sicherheitshalber ohne FlashROM.

2.2 Fehlersuche

Ist alles korrekt aufgebaut, sollte der *IDE-Flasher* mit Hilfe der vorhandenen Software funktionieren. Was aber wenn nicht? Hier ist ein Beispiel für die Fehlersuche. Etwas verallgemeinert kann man damit auch weitere Fehler finden.

```
IDE-PORT1 ist frei. PIO-Hardware nicht gefunden
Fehler: 4e37f0 ceb7f0
```

Diese Fehlermeldung bedeutet, dass der *IDE-Flasher* prinzipiell funktioniert, wenn auch noch nicht korrekt. Der Fehlercode besteht aus zwei Teilen. Dabei wurde der Wert `4e37f0` zur Platine gesendet, aber `ceb7f0` empfangen. An dieser Stelle sieht man, dass das Bit 8 jedesmal gesetzt ist. Das deutet darauf hin, dass die Datenleitung DB0, Pin 17 an CN3, siehe Schaltplan auf Seite 26, nicht den korrekten Wert hat und immer H-Pegel aufweist.

Zur Sicherheit kann man noch die Ausgänge der PIO testen. Dazu kann man sich des von DOS mitgelieferten Programmes `debug` bedienen. Die Pinbelegung der PIO kann unter Kapitel 7.2 nachgeschlagen werden. Die Basisadresse ist `0x170` sollte der *IDE-Flasher* am *Secondary Port* angeschlossen sein und `0x1F0` für den *Primary Port*. Ein Testprogramm könnte folgendermaßen aussehen:

```
c:\debug
-o 173,80 ; alle PIO Leitungen auf Ausgang setzen
           ; an allen Leitungen PAX..PCx sollte der Wert 0 anliegen
-o 172,F0 ; an Leitungen PC0..7 sollte der Wert 0xf0 anliegen
-i 172    ; an dieser wird der Wert ausgelesen, der nach Port PBx
F0       ; geschrieben wurde
-o 171,37 ; an Leitungen PB0..7 sollte der Wert 0x37 anliegen
           ; nach unserem obigen Problem wird am Ausgang statt 0x37
           ; 0xB7 anliegen.
-i 171    ; an dieser wird der Wert ausgelesen, der nach Port PBx
B7       ; geschrieben wurde
-o 170,4e ; an Leitungen PA0..7 sollte der Wert 0x4e anliegen
-i 170    ; hier wird genau gesagt nicht der Wert ausgelesen, der
CE       ; nach Port PAX geschrieben wurde, sondern der an den Port-
           ; Leitungen PA0..7 anliegt. Idealerweise sollte es natürlich der
           ; gleiche Wert sein, da der in Port A geschriebene Wert auch an
           ; PA0..7 anliegen soll
```

Programm 1 Testprogramm für die PIO Leitungen

Nach dem Abarbeiten des Programmes darf man mit großer Sicherheit davon ausgehen, dass ein Fehler an der Datenleitung DB0 vorliegt. In diesem Fall war der Leiterzug unterbrochen.

Zur Verallgemeinerung hier noch ein zweites Beispiel: Dieser Fall ist mit einer anderen Fehlermeldung verbunden.

```
IDE-PORT1 ist frei. PIO-Hardware nicht gefunden
Fehler: 4e37f0 4f37f0
```

```
c:\debug
-o 173,80 ; alle PIO Leitungen auf Ausgang setzen
           ; an allen Leitungen PAX..PCx sollte der Wert 0 anliegen
-o 172,F0 ; an Leitungen PC0..7 liegt der Wert 0xf0 an
-i 172    ; an dieser wird der Wert ausgelesen, der nach Port PBx
F0       ; geschrieben wurde
-o 171,37 ; an Leitungen PB0..7 liegt der Wert 0x37 an
-i 171    ; an dieser wird der Wert ausgelesen, der nach Port PBx
37       ; geschrieben wurde
           ; das muss wieder 0x37 sein.
-o 170,4e ; an Leitungen PA0..7 sollte der Wert 0x4e anliegen
           ; hier wurden die Ausgänge werden der PIO vermessen und es
           ; lag der Wert 0x4f an.
-i 170    ; hier wird genau gesagt nicht der Wert ausgelesen, der
4f       ; nach Port PAX geschrieben wurde, sondern der an den Port-
           ; Leitungen PA0..7 anliegt. Idealerweise sollte es natürlich der
           ; gleiche Wert sein, da der in Port A geschriebe Wert auch an
           ; PA0..7 anliegen soll.
           ; Der fehlerhafte Wert 0x4f hilft bei der Fehlersuche weiter.
```

Programm 2 Testprogramm für die PIO Leitungen

Sollten an den Ausgängen von Port B und C jeweils 0x37 und 0xf0 anliegen oder jeder beliebige andere eingestellte Wert, kann es kaum ein Fehler in den Datenleitungen sein. Port B und C arbeiten ja nachweislich jeweils korrekt. Der Fehler, der in Programm 2 analysiert wird, kann vom einem Kurzschluss eines der Ausgänge von Port A stammen. Da offensichtlich Bit 0 vom PIO-Port A falsch gesetzt ist, kann man in diesem Fall das Netz vom PIO Pin 4 (PA0) untersuchen, ob es einen Fehler enthält. Meist handelt es sich dabei um einen Kurzschluss mit benachbarten Pins.

3. Jumperbelegung

3.1 Technische Hintergründe

Bevor es zu den Bedeutungen der Jumperstellungen geht, sollen die jeweiligen technischen Grundlagen kurz erläutert werden. Aufgrund der unterschiedlicher Bauformen (DIP28 oder DIP32), Pinbelegung oder Programmierspannung sind jeweils andere Jumperpositionen notwendig. Das Verständnis der Bedeutungen für die einzelnen Jumper-Positionen ist sehr wichtig, um ein unbeabsichtigtes Zerstören der FlashROMs zu verhindern. Dies gilt insbesondere für neue bzw. unbekannte Typen.

Ohne Beschränkung der Allgemeinheit betreffen die nachfolgenden Aussagen im wesentlichen 8-Bit organisierte FlashROMs.

ROMs mit einer Kapazität von 64 KByte verfügen über 16 Leitungen zum Adressieren aller Speicherplätze. Mit diesen 16 Leitungen sind $2^{16} = 65536$ Byte oder $65536 * 8 = 512 * 1024 = 512$ KBit adressierbar. Diese Leitungen werden mit Adressleitung A_0 bis A_{15} bezeichnet. Dementsprechend werden die Leitungen für größere Bausteine durchnummeriert.

Kapazität		Adressleitungen	Beispiel
1 Mbit	128 KByte	$A_0..A_{16}$	FlashROM 28F010
2 Mbit	256 KByte	$A_0..A_{17}$	FlashROM 29C020
4 Mbit	512 KByte	$A_0..A_{18}$	FlashROM 29F040
8 Mbit	1 MByte	$A_0..A_{19}$	EPROM 27C080

Tabelle 2 Kapazität und Anzahl der Adressleitungen

An dieser Stelle sei noch bemerkt, dass die maximale Kapazität von EPROMs im 32poligen Gehäuse normalerweise 8 Mbit bzw. 1 MByte beträgt, während sie bei FlashROMs 4 Mbit bzw. 512 KByte beträgt. Der Unterschied begründet sich darin, dass alle Pins belegt sind und beim FlashROM aber eine Adressleitung für die Schreibleitung „geopfert“ wurde. Dieser Unterschied wird durch den Jumper JP5 behandelt. Auf die Bedeutung der Jumper wird im nächsten Kapitel detailliert eingegangen.

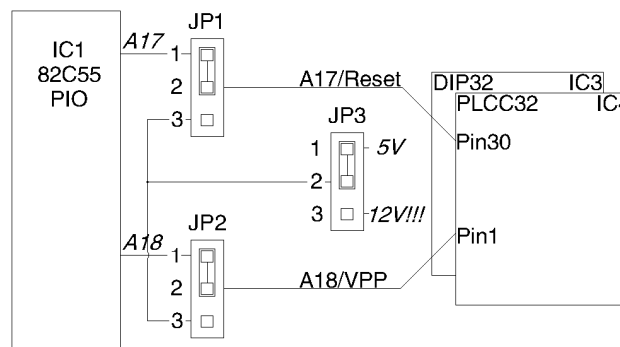


Abbildung 3 Funktion von JP1, JP2 und JP3

Die Abbildung 3 zeigt ein Detail aus dem Schaltplan Kapitel 7.3. Sie hebt die Bedeutung und Funktion der Jumper JP1, JP2 und JP3 hervor.

Um die Bedeutung der Jumper JP1 und JP2 zu verdeutlichen, wird als erstes Beispiel ein 4 Mbit FlashROM ausgewählt. Aus der Tabelle 2 geht hervor, dass bei 4 Mbit-Typen Adressleitungen A₀ bis A₁₈ verwendet werden. Diese Adressleitungen müssen dann auch über die Jumper JP1 und JP2 gesetzt werden. Dazu setzt man diese Jumper auf A17 und A18. Bei den meisten 1 Mbit FlashROMs werden die Pins die für A17/A18 vorgesehen sind nicht beschalten und können offen gelassen werden. Wenn die Jumper auf A17 und A18 belassen werden, entspricht das meist nicht den Spezifikationen, aber zumindest sollten die FlashROMs bei dieser Jumperstellung nicht zerstört werden.

Für den Intel 28F001BX Baustein ist die erforderliche Jumperstellung JP1 2-3, JP2 2-3 und JP3 2-3. Die Abbildung 3 zeigt, dass das Pin 1 und Pin 30 des Schaltkreises mit 12V verbunden werden. Die 12V Spannung wird gleichzeitig an beide FlashROM-Fassungen auf der Platine angelegt, deswegen ist es meist nicht möglich notwendig, ein 12V-FlashROM zu programmieren, während in der anderen Fassung noch ein 5V-Typ steckt.

3.2 Jumperstellungen

	Jumper PIN 1	PIN 2	PIN 3
JP1	A17 von der PIO	IC3/4: PIN 30	Spannung von JP3
JP2	A18 von der PIO	IC3/4: PIN 1	Spannung von JP3
JP3	5V	für JP1 und JP2	12V
JP4	A9 von der PIO	IC3/4: PIN 26	Hardware ID (12V)
JP5	/WR vom IDE port: PIN23	IC3/4: PIN 31	A19 von der PIO

Tabelle 3 Jumper Pins und Bedeutungen für den IDE-Flasher

Diese Tabelle erklärt die Bedeutung der Jumper. JP1 auf 1-2 bedeutet, dass die Adressleitung A17 von der PIO an PIN30 von IC3 und IC4 gelegt wird. Die Adressleitung A17 verwenden alle 2 oder 4 Mbit FlashROMs. Deswegen wird auch der Jumper JP1 auf Position 1-2 gesetzt. Für 1 Mbit FlashROMs wird das Pin 30 häufig als NC (not connected) beschrieben und zur Sicherheit sollte das Pin auch nicht beschalten, d.h. Position „offen“, werden.

Hinweis: Die Erfahrung im Umgang mit dem IDE-Flasher zeigt, dass die Jumperstellung 1111, d.h. jeder Jumper ist auf Position 1-2 gesetzt, eine Art Standardeinstellung ist, die für viele Typen funktioniert. Eine generelle Erklärung aber, dass diese Einstellung immer sicher ist oder nicht, kann nicht abgegeben werden. Wir haben eine Reihe von Bausteinen getestet und es funktioniert in den meisten Fällen. Aber wie und ob man ein „unbenutztes“ Pin beschalten kann, sei der jeweiligen Spezifikation des Bausteins überlassen.

Die nachfolgenden Erklärungen der generellen Funktionsweise der Jumper beziehen sich auf die von uns getesteten ICs. Bei der Vielzahl vorhandener ICs ist es nicht ausgeschlossen, dass die Jumperstellungen bei einigen Typen eine andere Bedeutung haben kann.

- Jumper JP1 – Position 1-2, 2-3 oder offen

1-2: wird bei FlashROMs mit einer Kapazität von 2 und 4 Mbit benötigt (z.B. AM29F020, AM29F040, MX29F002). Der Jumper verbindet A17 von der PIO und Pin 30 von IC3/4.

2-3: In dieser Stellung wird die von JP3 eingestellte Spannung mit Pin 30 an IC3 und IC4 verbunden. Diese Stellung wird **a)** für einige 28er-Typen (z.B. 28F001BX) zur Programmierung verwendet oder **b)**, wenn JP3 auf 1-2 steht, zum Auslesen von EPROMs im 28poligem Gehäuse (z.B. 27C64, ... , 27C512) gebraucht.

Offen: Für ICs mit einer Kapazität ≤ 1 Mbit sollte das Pin nicht beschalten werden.

- Jumper JP2 – Position 1-2, 2-3 oder open

1-2: wird bei FlashROMs mit einer Kapazität 4 Mbit benötigt (z.B. AM29F040). Der Jumper verbindet A18 von der PIO und Pin 1 von IC3/4.

2-3: In dieser Stellung wird die von JP3 eingestellte Spannung mit Pin 30 an IC3/4 verbunden. Es wird für einige 28er-Typen (z.B. 28F010, I28F001BX) als Programmiervoltage benötigt.

***Hinweis:** Die über JP1 und JP2 eingestellte Spannung wird an beide Sockel (DIP32 & PLCC32) angelegt. D.h. wenn an einem FlashROM 12V anliegen sollen, so wird diese Spannung an dem anderen auch auf 12V eingestellt.*

Offen: Für ICs mit einer Kapazität ≤ 2 Mbit sollte das Pin nicht beschalten werden.

- Jumper JP3 – Position 1-2 oder 2-3

1-2: Hiermit wird 5V für JP1 and JP2 eingestellt. Pin 3 von JP1 und JP2 wird mit 5V verbunden.

2-3: Hiermit wird 12V für JP1 and JP2 eingestellt. Pin 3 von JP1 und JP2 wird mit 12V verbunden.

***Hinweis:** Wird die Stellung 2-3 zu Programmierung von 12V-Typen eingesetzt, sollte man danach den Jumper wieder in die Standardposition setzen.*

- Jumper JP4 – Position 1-2 oder 2-3

1-2: Das ist die Standardeinstellung für alle Typen.

2-3: Diese Stellung wird für die Funktion Hardware-ID-Erkennung benötigt, als zusätzliche Information um unbekannte FlashROM zu identifizieren. Die Software wird explizit zum Setzen dieser Stellung auffordern.

Hinweis: Bevor jedoch ein unbekannter Baustein eingesetzt wird, sollte man sich versichern, ob dieser auch mit 5V Betriebsspannung arbeiten kann.

- Jumper JP5 – Position 1-2 oder 2-3

1-2: Das ist die Standardeinstellung für alle FlashROM-Typen.

2-3: Um EPROMs mit einer Kapazität von 512 KByte oder 1 MByte auszulesen ist diese Position zu setzen.

4. Benutzung

Die Verbindungen zum Rechner (Stromversorgung, IDE-Kabel) müssen im ausgeschalteten Zustand des Rechners erfolgen.

Abbildung 4 zeigt den Anschluss der Stromversorgung an den IDE-Flasher. Der Steckverbinder verfügt nicht über eine Verriegelung, die normalerweise ein unbeabsichtigtes Lösen der Verbindung erschweren soll. Dies soll ein Wechseln der Platine erleichtern. An der Unterseite des Stromversorgungssteckerbinders (hier nicht sichtbar) befindet sich ein kleiner Widerhaken. Dies wird z. B. bei Floppy-Laufwerken benutzt.

Sollte der Steckverbinder verkehrt herum angeschlossen werden, ist eine Zerstörung der Platine oder Teile des Rechners möglich.

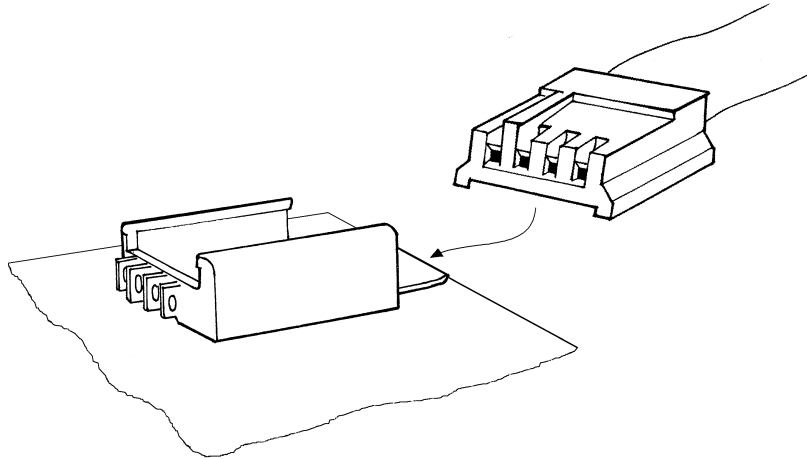


Abbildung 4 Anschluss der Stromversorgung an den IDE-Flasher

Um den *IDE-Flasher* anzuschließen, ist ein freier IDE-Port notwendig (Primary IDE oder Secondary IDE). Über diesen freien Port wird er mit einem handelsüblichen Festplattenkabel verbunden. Die Bezeichnungen Primary IDE und Secondary IDE stehen jeweils für den ersten und zweiten Anschluss auf dem Mainboard. Es ist nicht die Master/Slave-System, welches bei Festplatten üblich ist, zu verwechseln. Als Master und Slave bezeichnet man die zwei Geräte, die an einem Festplattenkontroller angeschlossen sind. Ist der *IDE-Flasher* angeschlossen, darf sich an diesem IDE-Kabel kein weiteres Gerät befinden. Der *IDE-Flasher* ist das einzige Gerät an diesem Port. Pin 1 auf dem Motherboard muss dabei mit Pin 1 auf dem *IDE-Flasher* verbunden werden. Der Wannenstecker ist mit ▼ markiert. Die rote Markierung des Festplattenkabels muss jeweils mit Pin 1 der Steckverbinder verbunden werden.

Auf den meisten Mainboards sind heutzutage zwei IDE-Ports vorhanden. Abbildung 5 demonstriert eine mögliche Anordnung der Steckplätze für die IDE-Ports. Das betrifft insbesondere Mainboards ab Pentium aufwärts. Ältere Platinen z.B. 80386er verfügen meist nur über einen IDE-Kontroller mit nur einem Port. Ein Zweiter kann meist problemlos über eine Soundkarte z.B. für CD-Laufwerke nachgerüstet

werden.

Die grüne LED leuchtet, sobald der Stromversorgungsstecker angeschlossen wurde und die 5V Betriebsspannung anliegt.

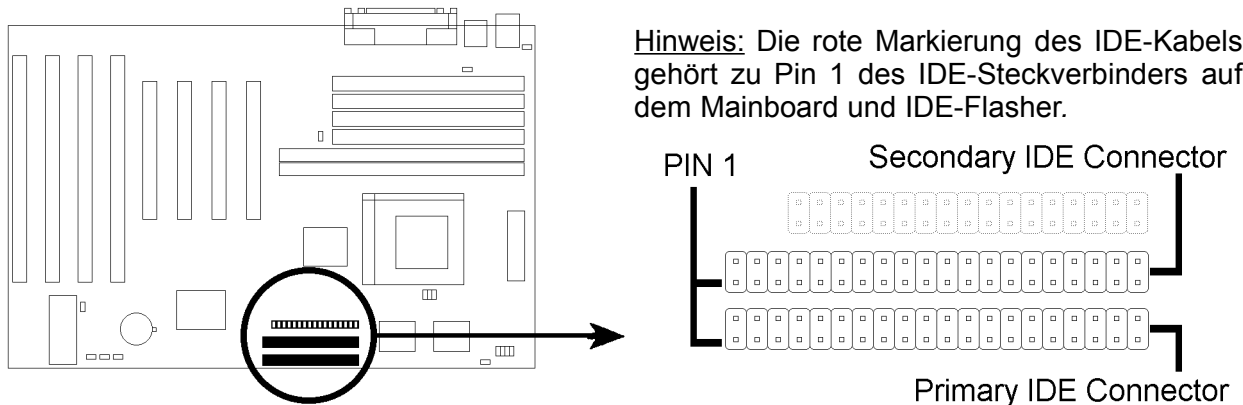


Abbildung 5 mögliche Anordnung der IDE-Ports auf dem Mainboard

Hinweis: Um eine Zerstörung des *IDE-Flashers* oder der ICs zu vermeiden, sollten folgende Richtlinien beachtet werden:

- Vor dem Wechseln der ROMs sollte der Jumper JP3 auf 1-2 gesetzt werden. Dies verhindert, dass unbeabsichtigt 12 Volt an den FlashROMs anliegt. Manche 5 Volt Typen der 29/39/49xxx Serie könnten sonst zerstört werden.
- Beim Einsetzen der ROMs ist unbedingt auf die korrekte Ausrichtung der ICs zu achten. Die Fassungen und ICs haben deshalb an Pin 1 eine kleine Aussparung oder Markierung am Gehäuse. Bei der PLCC-Fassung, kann man die Orientierung an Hand des Pfeils und der abgeschrägten Ecke erkennen. Pin 1 des ICs muss auch mit PIN1 der Fassung verbunden werden.

Sollte ein ICs verkehrt in die Fassung gesteckt worden sein, ist dies meist einfach an der starken Erwärmung des ICs zu bemerken. In diesem Fall ist sofort der Stromversorgungssteckverbinder und das IDE-Kabel zu entfernen.

Bei der *IDE-Flasher* Revision F zeigt die Markierung der Fassung nach rechts und der Pfeil der PLCC-Fassung nach unten.

- 32polige ICs werden dann wie oben beschrieben in die Fassung eingesetzt.
- Eine kleine Ausnahme bilden die 28poligen EPROMs. Vorher sind die Jumper JP1 auf 2-3 und JP3 auf 1-2 setzen! Die Abbildung 6 zeigt, dass diese EPROMs um 2 Pins nach links versetzt eingesetzt werden müssen. Bei den anderen Revisionen sind demnach sinngemäß die 2 ersten Pins freizuhalten.
- FlashROMs mit gleicher Programmierspannung, d.h. *beide* tragen jeweils die gleiche Serienbezeichnung z.B. 28xxx, können in den Fassungen kombiniert und programmiert werden.
- Werden zwei Flash-EPROMs mit unterschiedlichen Programmierspannungen z.B. 28xxx und 29xxx kombiniert, so darf auf keinen Fall der Jumper JP3 auf 2-3

eingestellt werden. Sonst liegt die 12V Programmierspannung auch am 5V-FlashROM an und kann möglicherweise diesen Chip zerstören.

Somit lässt sich der 12V-Typ nicht programmieren, ohne den 5V Typen zu entfernen. Der 5V-Typ wiederum lässt sich in diesem beschriebenen Fall problemlos programmieren. Eine Möglichkeit ist es die Speicher-ICs einzeln einzusetzen, um von vornherein diese Probleme mit der Programmierspannung zu vermeiden.

- Für das Arbeiten mit FlashROMs im PLCC32-Gehäuse ist es sehr zu empfehlen, eine PLCC-Zange zu verwenden, um die Fassung/ICs zu schonen. Als temporäre Lösung kann auch Zahnseide Wunder wirken.
- Die ICs sollten nur im stromlosen Zustand der Schaltung gewechselt werden. Dazu empfiehlt es sich die Schaltung an den PC im ausgeschalteten Zustand anzuschließen und dann erst den Rechner zu starten. Für die Entfernung des IDE-Flashers ist der Rechner zuerst herunterzufahren. Die Software des IDE-Flashers erlaubt zwar das Wechseln der Platine, aber das IDE-Interface auf dem Mainboard ist aber per Spezifikation nicht darauf ausgelegt, dass im laufenden Betrieb Geräte gewechselt werden können.
- Wenn der Rechner nicht ausgeschaltet werden kann, so sollte zuerst das IDE-Kabel und dann der Stromverbinder entfernt werden.

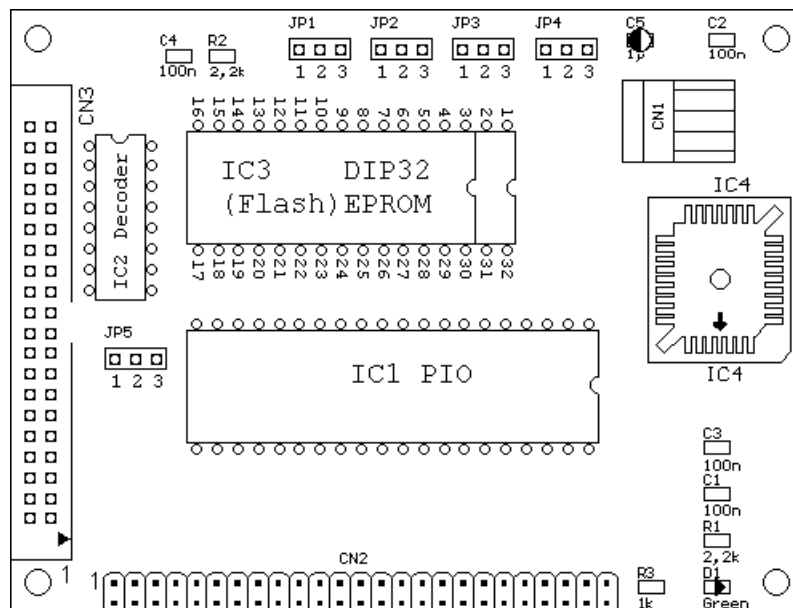


Abbildung 6 Layout des IDE-Flasher Rev. F

5. Software

Das Programm `ideflash.exe` ist eine reine DOS-Anwendung und sollte auch nur im DOS-Modus gestartet werden. Die Erfahrung zeigt aber, dass es unproblematisch ist, dieses Programm in einer DOS-Box unter Windows 95, Windows 98 respektive Windows ME auszuführen. Eine Garantie kann dabei nicht übernommen werden, da im Einzelfall die Spezifikationen der Timing-Parameter nicht immer eingehalten werden können. Das Programm funktioniert nicht unter Windows 2000 oder XP.

In diesem Kapitel werden kurz die wesentlichen Funktionen der Software beschrieben. Der Parameter `-port` teilt der Software mit, an welchem IDE-Port die Schaltung angeschlossen ist. Dabei steht 0 für den *Primary IDE-Port*, 1 für *Secondary IDE*, 2 und 3 für die darauffolgenden Ports. Über den Parameter `-socket` wird die zu verwendende IC-Fassung ausgewählt. Mögliche Werte sind `DIP32` oder `PLCC32`. Neu hinzugekommen ist der Parameter `-hardware`. Mögliche Werte sind `RevD`, `RevE` oder `RevF`. Für die meisten Benutzer ist der Standardwert `RevF`. Die anderen sind für frühere Revisionen der Hardware reserviert. Der Parameter wird unter anderem dafür benutzt, die erforderliche Jumper-Einstellungen anzuzeigen oder revisionsspezifische Parameter zu kontrollieren. Z.B. wird erst seit Hardware Rev. F das Auslesen von EPROMs mit 1 MByte Kapazität unterstützt.

Die Gross- und Kleinschreibung ist relevant. Es wird also zwischen `'dip32'` und `'DIP32'` unterschieden. Der korrekte Wert für den Parameter `-socket` ist demnach `DIP32`.

5.1 Arbeiten mit FlashROMs

Hinweis: Bevor man anfängt den womöglich einzigen BIOS-Chip zu überschreiben, sollte man über Möglichkeit untersuchen, das Update erst auf einem zweiten Chip zu probieren. Geht dabei irgendwas schief, so hat man immernoch den Original-Baustein. Auf alle Fälle ist ein Backup des alten Bausteins empfohlen, diesen kann man dann im Falle von Problemen wieder in ein FlashROM zurückschreiben.

FlashROM auslesen in eine Datei:

```
ideflash -port 1 -hardware RevF -socket DIP32 -f_read <Datei>
```

Vor dem Schreiben, das Löschen nicht vergessen:

```
ideflash -port 1 -hardware RevF -socket DIP32 -f_erase
```

Von Datei in FlashROM schreiben:

```
ideflash -port 1 -hardware RevF -socket DIP32 -f_write <Datei>
```

5.2 Auslesen eines EPROMs

Hinweis: Bei 28poligen Gehäuse ist **JP1 2-3** und **JP3 1-2** zu setzen.

`ideflash -port 1 -hardware RevF -socket DIP32 -e_read <Datei> -e_size XX`
 der Parameter hinter `-e_size` wird in KByte angegeben und muss der Größe des EPROMs entsprechen, andernfalls können falsche Daten ausgelesen werden.

Ein 8KByte EPROM 2764 braucht den Parameter `-e_size 8`, 27128 den Wert `16`, 27256 `32` und 27512 die `64` u.s.w.

5.3 Auslesen der Hardware ID

Diese Funktion wird normalerweise benutzt, wenn der IDE-Flasher einen Chip nicht ueber die normale ID-Funktion erkennen kann. Das betrifft insbesondere FlashROMs, die noch nicht von der Software unterstützt werden. Eine Hardware-ID-Erkennung für EPROMs ist nicht mehr vorgesehen. Dieses Feature wurde entfernt. Der Nutzen ist gering und trägt aber zur Verwirrung bei Hinsichtlich der Fähigkeiten des *IDE-Flashers* FlashROMs zu programmieren und EPROMs auszulesen bei. Der *IDE-Flasher* kann keine EPROMs programmieren.

Hinweis: Bevor man die Hardware-ID-Funktion benutzt, sollte man unbedingt sicherstellen, dass der ausgewählte Schaltkreis auch mit 5V Betriebsspannung arbeitet.

`ideflash -port 1 -hardware RevF -socket DIP32 -e_hardid`

5.4 Konfigurationsdatei

Um nicht ständig gleichbleibende Kommandozeilen-Parameter wie z.B. `-port`, `-socket` und `-hardware` angeben zu müssen, kann man alle Parameter in eine Datei `'ideflash.rc'` schreiben. Es besteht auch die Möglichkeit mehrere Konfigurationsdateien zu installieren. Die Suchreihenfolge ist dann wie folgt: Zuerst wird im lokalen Verzeichnis die Datei `'.\ideflash.rc'`, gesucht. Wird sie nicht gefunden, wird als nächstes der Pfad `'%HOME%\ideflash.rc'` angenommen und zum Schluss `'c:\etc\ideflash.rc'`. Die Umgebungsvariable `HOME` kann vom Administrator beim Login gesetzt werden.

Der Inhalt könnte folgendermaßen aussehen:

```
port      = 1
socket    = DIP32
hardware  = RevF
lang      = DE
```

6. Liste der FlashROMs

Die folgende Tabelle enthält die Schaltkreise, die von der Software explizit unterstützt werden. Sie enthält jeweils die Kapazität, Hersteller ID, Jumperstellung, Bezeichnung und den Namen des Herstellers soweit bekannt. Die Jumper JP4 und JP5 müssen in Stellung 1-2 verbleiben. Sie haben eine besondere Funktion und werden im normalen Betrieb nicht geändert. Die von der Software verwendete Kodierung der Jumperpositionen wird auch in dieser Tabelle verwendet. Die Positionen werden vom Programm vor dem Setzen angezeigt.

Die erste Ziffer der Zeichenfolge steht dabei für den Jumper JP1, die zweite für JP2 etc. Die Bedeutung ergibt sich dann folgendermaßen: 1 – steht für Position 1-2, 2 offen und 3 steht für die Position 2-3. Als Beispiel steht die Zeichenfolge '2211' des AT29C512 für die Jumperstellungen: 2211 – JP1 offen – JP2 offen – JP3 1-2 – JP4 1-2.

Sollte ein verwendeter IC nicht aufgelistet sein, werden wir versuchen zu helfen. Dazu senden sie uns folgende Informationen:

Hinweis: Bevor ein Schaltkreis in eine Fassung eingesetzt wird, ist unbedingt zu prüfen, ob er auch mit 5V Betriebsspannung arbeiten kann. Falls Unklarheiten bestehen, ist es auch möglich uns eine Email zu schicken und zu fragen.

Kann der verwendete Schaltkreis im *IDE-Flasher* arbeiten, schicken Sie uns noch folgende Informationen:

- 1) Hardware ID (JP4 auf 2-3 setzen)

```
ideflash -port <X> -socket <Y> -hardware RevF -e_hardid
```

- 2) Software-ID (JP4 wieder zurück auf 1-2 setzen)

```
ideflash -port <X> -socket <Y> -hardware RevF -f_id -debug
```

Hierbei steht <X> für den benutzten IDE-Port (0 oder 1) und <Y> für den verwendeten IC-Sockel (DIP32 oder PLCC32).

- 3) Schicken Sie uns auch andere Informationen über den IC, wie z.B. Hersteller oder Typenbezeichnung. Die Information AMI BIOS oder AWARD BIOS ist dabei nicht sehr aussagekräftig. Die Bezeichnung ist auf dem IC direkt aufgedruckt. Die wichtige Information zum Typ findet man leider erst, wenn man den Aufkleber ablöst.

6.1 Sortiert nach Kapazität

Kapazität	ID	Jumper	Name	Hersteller
64	5d1f	2211	AT29C512	Atmel
64	b889	1331	28F512	Intel
64	c8da	2211	W29EE512P	Winbond
128	2001	2211	AM29F010	Advanced Micro Devices
128	2001	2211	AM29F010B	Advanced Micro Devices
128	c1da	1111	AE29F1008	AE
128	c437	2311	A29001(1)B	AMIC Technology, Inc.
128	a137	2311	A29001(1)T	AMIC Technology, Inc.
128	d51f	2211	AT29C010A	Atmel
128	b431	1331	CAT28F010	Catalyst
128	1ac2	2331	M12116	H.T.
128	9589	3331	28F001BX-B	Intel
128	9489	3331	28F001BX-T	Intel
128	b489	1331	28F010	Intel
128	1ac2	2331	MX28F1000	Macronix
128	19c2	2211	MX29F001(N)B	Macronix
128	18c2	2211	MX29F001(N)T	Macronix
128	b5bf	2211	SST39SF010A	Silicon Storage Technology
128	07bf	1111	ST29EE010	Silicon Storage Technology
128	0720	1331	M28F101	STMicroelectronics
128	c1da	1111	W29EE011	Winbond
256	2a01	1331	AM28F020	Advanced Micro Devices
256	3401	1311	AM29F002(N)B	Advanced Micro Devices
256	b001	1311	AM29F002(N)T	Advanced Micro Devices
256	45da	1211	AE29F2008	AE
256	0d37	1311	A29002(1)B	AMIC Technology, Inc.
256	8c37	1311	A29002(1)T	AMIC Technology, Inc.
256	da1f	1211	AT29C020	Atmel
256	071f	1311	AT49F002(N)	Atmel
256	081f	1311	AT49F002(N)T	Atmel
256	bd31	1331	CAT28F020	Catalyst
256	971c	1311	EN29F002(N)B	EON Silicon Devices
256	921c	1311	EN29F002(N)T	EON Silicon Devices
256	b0ad	1311	HY29F002T	Hynix
256	bd89	1331	28F020	Intel
256	34c2	1311	MX29F002(N)B	Macronix
256	b0c2	1311	MX29F002(N)T	Macronix
256	2ac2	1331	MX28F2000	Macronix
256	a240	1311	V29C51002B	Mosel Vitelic o. SyncMos
256	0240	1311	V29C51002T	Mosel Vitelic o. SyncMos
256	10bf	1211	ST29EE020	Silicon Storage Technology
256	b6bf	1211	SST39SF020A	Silicon Storage Technology
256	45da	1211	W29C020	Winbond
256	0bda	1311	W49F002U	Winbond
512	a401	1111	AM29F040	Advanced Micro Devices
512	8637	1111	A29040A	AMIC Technology, Inc.
512	a41f	1111	AT29C040	Atmel
512	041c	1111	EN29F040	EON Silicon Devices

Kapazität	ID	Jumper	Name	Hersteller
512	a404	1111	F29F040A	Fujitsu
512	a4c2	1111	MX29F040	Macronix
512	b7bf	1111	SST39SF040A	Silicon Storage Technology
512	e220	1111	M29F040	STMicroelectronics
512	a401	1111	TMS29F040	Texas Instruments

Table 4 unterstützte FlashROMs und Jumperstellungen nach Kapazität sortiert

6.2 Sortiert nach Hersteller

Hersteller	Kapazität	ID	Jumper	Name
Advanced Micro Devices	128	2001	2211	AM29F010
Advanced Micro Devices	128	2001	2211	AM29F010B
Advanced Micro Devices	256	2a01	1331	AM28F020
Advanced Micro Devices	256	3401	1311	AM29F002(N)B
Advanced Micro Devices	256	b001	1311	AM29F002(N)T
Advanced Micro Devices	512	a401	1111	AM29F040
AE	128	c1da	1111	AE29F1008
AE	256	45da	1211	AE29F2008
AMIC Technology, Inc.	128	c437	2311	A29001(1)B
AMIC Technology, Inc.	128	a137	2311	A29001(1)T
AMIC Technology, Inc.	256	0d37	1311	A29002(1)B
AMIC Technology, Inc.	256	8c37	1311	A29002(1)T
AMIC Technology, Inc.	512	8637	1111	A29040A
Atmel	64	5d1f	2211	AT29C512
Atmel	128	d51f	2211	AT29C010A
Atmel	256	da1f	1211	AT29C020
Atmel	256	071f	1311	AT49F002(N)
Atmel	256	081f	1311	AT49F002(N)T
Atmel	512	a41f	1111	AT29C040
Catalyst	128	b431	1331	CAT28F010
Catalyst	256	bd31	1331	CAT28F020
EON Silicon Devices	256	971c	1311	EN29F002(N)B
EON Silicon Devices	256	921c	1311	EN29F002(N)T
EON Silicon Devices	512	041c	1111	EN29F040
Fujitsu	512	a404	1111	F29F040A
H.T.	128	1ac2	2331	M12116
Hynix	256	b0ad	1311	HY29F002T
Intel	64	b889	1331	28F512
Intel	128	9589	3331	28F001BX-B
Intel	128	9489	3331	28F001BX-T
Intel	128	b489	1331	28F010
Intel	256	bd89	1331	28F020
Macronix	128	1ac2	2331	MX28F1000
Macronix	128	19c2	2211	MX29F001(N)B
Macronix	128	18c2	2211	MX29F001(N)T
Macronix	256	2ac2	1331	MX28F2000
Macronix	256	34c2	1311	MX29F002(N)B
Macronix	256	b0c2	1311	MX29F002(N)T
Macronix	512	a4c2	1111	MX29F040
Mosel Vitelic o. SyncMos	256	a240	1311	V29C51002B
Mosel Vitelic o. SyncMos	256	0240	1311	V29C51002T
Silicon Storage Technology	128	b5bf	2211	SST39SF010A
Silicon Storage Technology	128	07bf	1111	ST29EE010
Silicon Storage Technology	256	b6bf	1211	SST39SF020A
Silicon Storage Technology	256	10bf	1211	ST29EE020
Silicon Storage Technology	512	b7bf	1111	SST39SF040A
STMicroelectronics	128	0720	1331	M28F101

Hersteller	Kapazität	ID	Jumper	Name
STMicroelectronics	512	e220	1111	M29F040
Texas Instruments	512	a401	1111	TMS29F040
Winbond	64	c8da	2211	W29EE512P
Winbond	128	c1da	1111	W29EE011
Winbond	256	45da	1211	W29C020
Winbond	256	0bda	1311	W49F002U

Tabelle 5 unterstützte FlashROMs und Jumperstellungen nach Hersteller sortiert

7. Anlagen

7.1 Pinbelegung des IDE-Steckverbinders

Dir	Signal	#	#	Signal	Dir
I	/Reset	1	2	Ground	
I/O	DD7	3	4	DD8	I/O
I/O	DD6	5	6	DD9	I/O
I/O	DD5	7	8	DD10	I/O
I/O	DD4	9	10	DD11	I/O
I/O	DD3	11	12	DD12	I/O
I/O	DD2	13	14	DD13	I/O
I/O	DD1	15	16	DD14	I/O
I/O	DD0	17	18	DD15	I/O
	Ground	19	20	KEYPIN	
O	DMARQ	21	22	Ground	
I	/IOWrite	23	24	Ground	
I	/IORead	25	26	Ground	
O	IOReady	27	28	CSEL	
I	/DMAAck	29	30	Ground	
O	INTRQ	31	32	/IOCS16	O
I	DA1	33	34	/PDIAG	
I	DA0	35	36	DA2	I
I	/CS0	37	38	/CS1	I
O	/DASP	39	40	Ground	

Tabelle 6 Pinbelegung des IDE-Steckverbinders [hATA]

Diese Tabelle dient als Gedankenstütze. Hier werden die Bezeichnungen für die einzelnen Signal und deren Pinnummer für ein generische IDE-Gerät aufgelistet. 'I/O' steht dabei für bidirektionalen Datatransfer, 'I' Eingang für ein IDE-Gerät und 'O' Datenausgang von einem IDE-Gerät zum IDE-Port des Controllers.

7.2 Pinbelegung der PIO 82C55

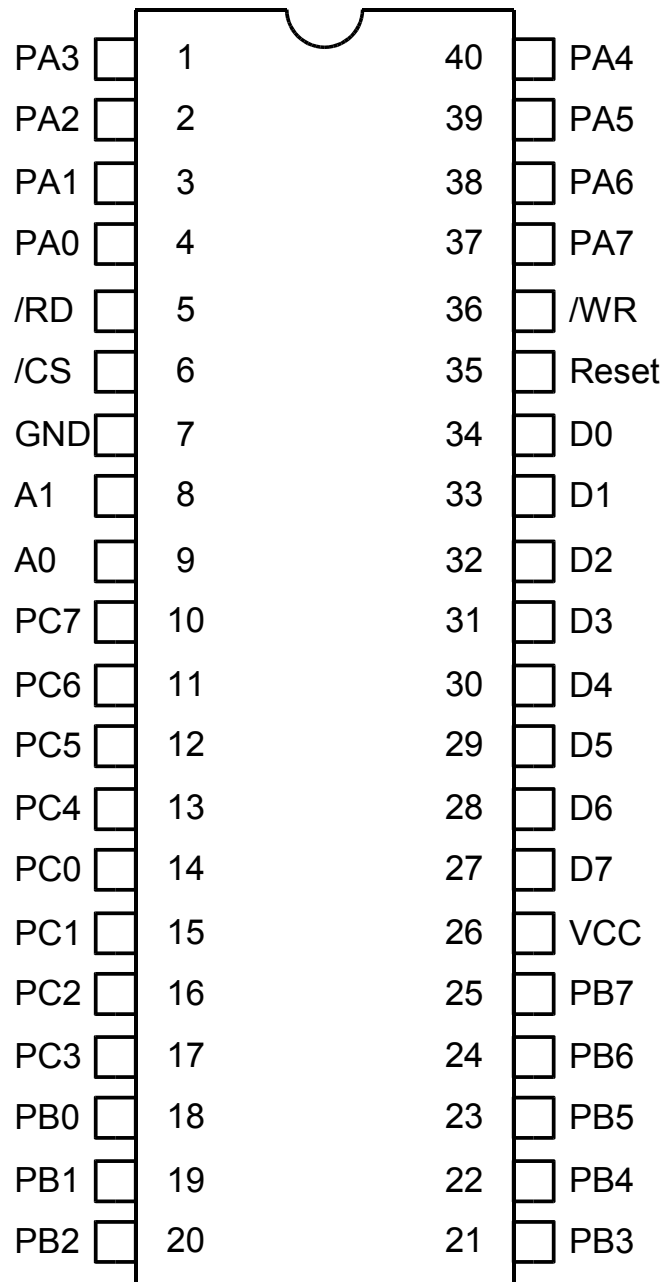


Abbildung 7 Pinbelegung 82C55

7.3 Schaltplan

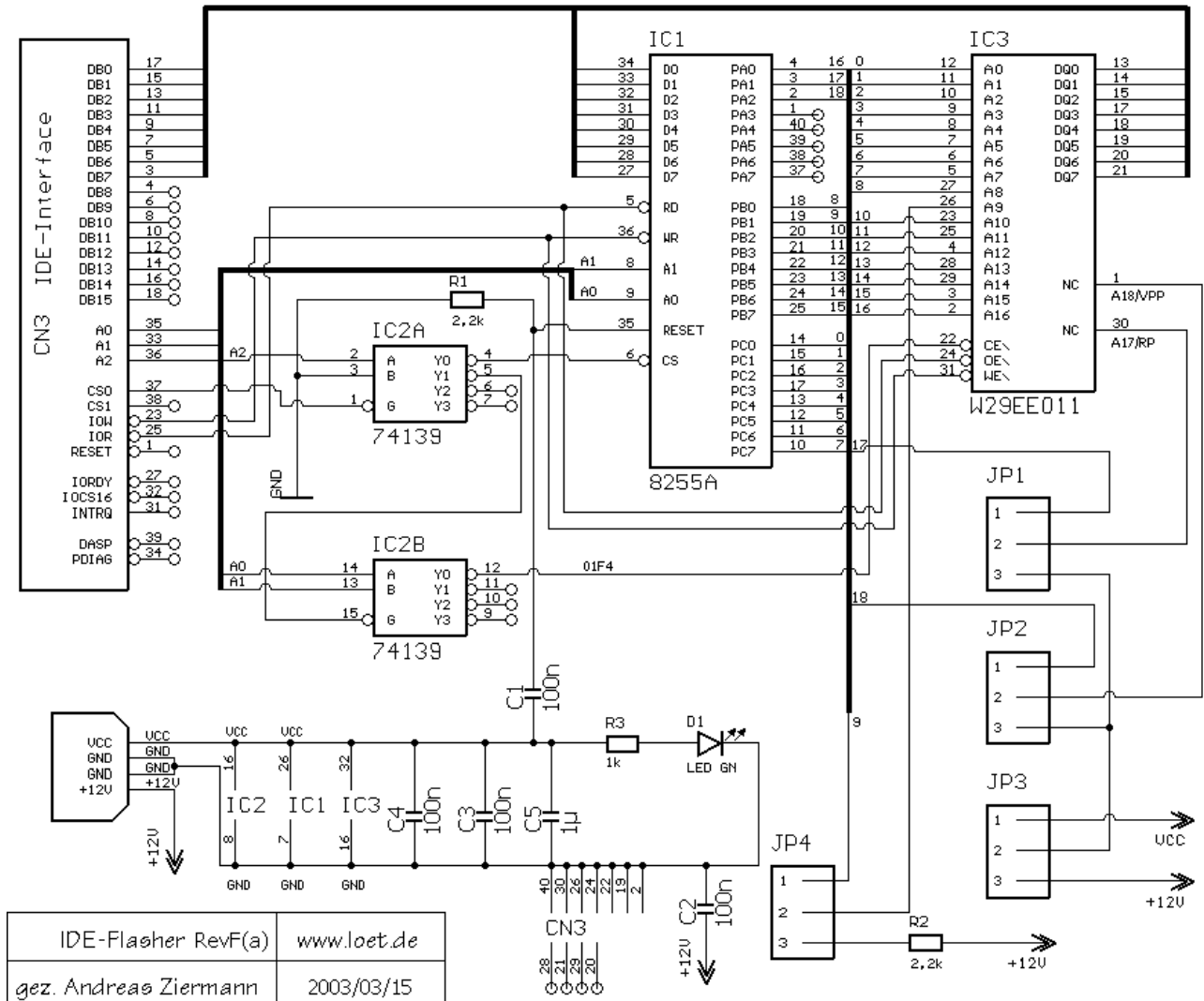


Abbildung 8 Schaltplan

8. Erweiterungen und Programmierung

8.1 Eigene Erweiterungsmodule

Es wurde häufig angefragt, ob es möglich ist, den *IDE-Flasher* für andere Zwecke einzusetzen. Ja, der *IDE-Flasher* kann für eigene Schaltungen benutzt werden. Wir selbst hatten als Beispiel einen Aufsatz zum Bestimmen der Kapazität von SMD-Kondensatoren (*CTestPro*) mittels eines NE555. In Planung sind eine Spannungsanpassung an LPC und FWH FlashRoms, A/D-Wandler-Aufsatz und ein einfacher EPROM-Programmer.

Also prinzipiell ist dagegen nichts einzuwenden, den Platine als Interface für eigene Schaltungen zu verwenden. Man sollte sich aber dabei immer vor Augen halten, dass dieses Gerät am IDE-Port angeschlossen wird und gegebenenfalls bei einem Defekt der Schaltung der IDE-Port auf dem Board beschädigt werden kann.

Ein einfacher Schutz vor Problemen dieser Art ist es nur die Ausgangsleitungen der PIO zu verwenden. Geht sie kaputt, ist ein einfacher Austausch möglich.

Man kann die Ausgangsleitungen PC0..7, PB0..7, PA0..2 über einen Adapter von der DIP32-Fassung herausführen oder verwendet den Erweiterungsstecker. Bei letztem wird ein 50poliger Wannestecker verwendet, wie er z.B. für SCSI-Geräte verwendet wird.

Hinweis: Um Kompatibilität mit älteren NMOS 8255 zu wahren, sollte man die Stromstärke im Blick behalten. Bei über 0,2 mA (Ausgang H), kann bei einigen 8255-Typen die Ausgangsspannung unter 2,4V sinken. Weitere Hinweise liefert das Datenblatt für den 8255 und 82C55.

8.2 Programmierung

Ist der *IDE-Flasher* an einem IDE-Port angeschlossen, ergibt sich folgende IO-Port konfiguration.

Der Variable für `adr` enthält die Basisadresse des ausgewählten IDE-Ports. Der Wert ist `0x1F0` für den *Primary Port* und `0x170` für den *Secondary Port*. Dadurch ergeben sich für die Ports der PIO folgende Adressen PIO-Port A hat die Adresse `adr+0`, PIO-Port B `adr+1`, PIO-Port C `adr+2` und PIO Controlwort `adr+3`.

Zum Schluss ein kleines Beispiel, um einen Zugriff auf den IDE-Flasher zu demonstrieren:

```
PORT_CONTROL = 3;
PORT_C       = 2;
PORT_B       = 1;
PORT_A       = 0;
adr          = 0x170;           // secondary port
outp(adr+PORT_CONTROL,0x80); // alle Pins auf Ausgang 0 setzen
outp(adr+PORT_C,0x67);       // PC0-PC7 wird auf 0x67 gesetzt
```

9. Abkürzungen

Liste von verwendeten Abkürzungen aus diesem Handbuch.

- AT - "IBM PC AT" or "Advanced Technology" [hDIE]:
Eine Version des IBM PC, die ab August 1984 ausgeliefert wurde. Er hatte einen Intel 80286 Prozessor, einen 16-bit Bus, eine mittelschnelle Festplatte und ein 1.2 MByte Diskettenlaufwerk.
- ATA - "Advanced Technology Attachment" or "AT Attachment" [hDIE]:
Ein Standard für ein Festplattenanschluss. Es basiert auf dem IBM PC ISA 16-bit Bus. Die ATA Spezifikation befasst sich mit der Stromversorgung als auch mit dem Datenaustausch zwischen dem Mainboard, dem integrierten Festplattenkontroller und dem Laufwerk. Der ATA "Bus" unterstützt nur zwei Geräte - Master and Slave genannt.
- EPROM "Erasable Programmable Read Only Memory" [hDIE]:
Ein Nur-Lese-Speicherbaustein, der mit ultraviolettem Licht gelöscht wird und danach wieder neu programmiert werden kann.
- EEPROM "Electrically Erasable Programmable Read-Only Memory" [hDIE]:
Ein nicht flüchtiger Speicherbaustein, der eine ähnliche Technik zur Speicherung verwenden wie in EPROMs, aber mit dem Unterschied, dass die Speicherzellen mittels eines elektrischen Impulses zurückgesetzt werden.
- Flash memory – "flash ROM", "flash EPROM" oder "FEEPROM" [hDIE]:
Ein spezieller nicht flüchtiger Speicherbaustein ähnlich dem EEPROM, aber er kann nur blockweise oder komplett gelöscht werden. Von 1995 an ersetzte diese neue Technologie die EPROMs. Einer großer Vorteil ist, das das Programmieren dieser Bausteine kann im System erfolgen kann. Der Schaltkreis muss dazu nicht mehr ausgewechselt werden.
- FWH Firmware Hub Interface:
Besonderer Typ eines FlashROMs. Er kann zusaetzlich neben der Firmware auch andere spezielle Funktionen besitzen, wie z.B. Zufallszahlen generieren. Ähnlich wie LPC-Speichern erfolgt hier die Kommunikation über ein spezielles proprietäres Protokoll. Der FWH is über den ICH mit Hilfe eines speziellen synchronisierten 33MHz Buses mit dem PCI-Bus verbunden. Die Abbildung 9 gibt einen kurzen Überblick über die logische Anordnung der FHW Komponente auf dem PC Mainboard.
- ICH "I/O Controller Hub":
Spezieller Schaltkreis auf vielen Mainboards, der die Kommunikation mit den externen Geräten übernimmt.

- IDE "Integrated Drive Electronics" or "Integrated Disc Electronic"
Synonym für ATA.
- LPC Low Pin Count Interface:

Eine Möglichkeit die Anzahl der Pins für ein FlashROM zu reduzieren. Der Schaltkreis wird über einen speziellen 4-Bit 33MHz LPC-Bus mit dem ICH verbunden. Die Abbildung 9 gibt einen kurzen Überblick über die logische Anordnung von LPC Komponenten im PC Mainboard.

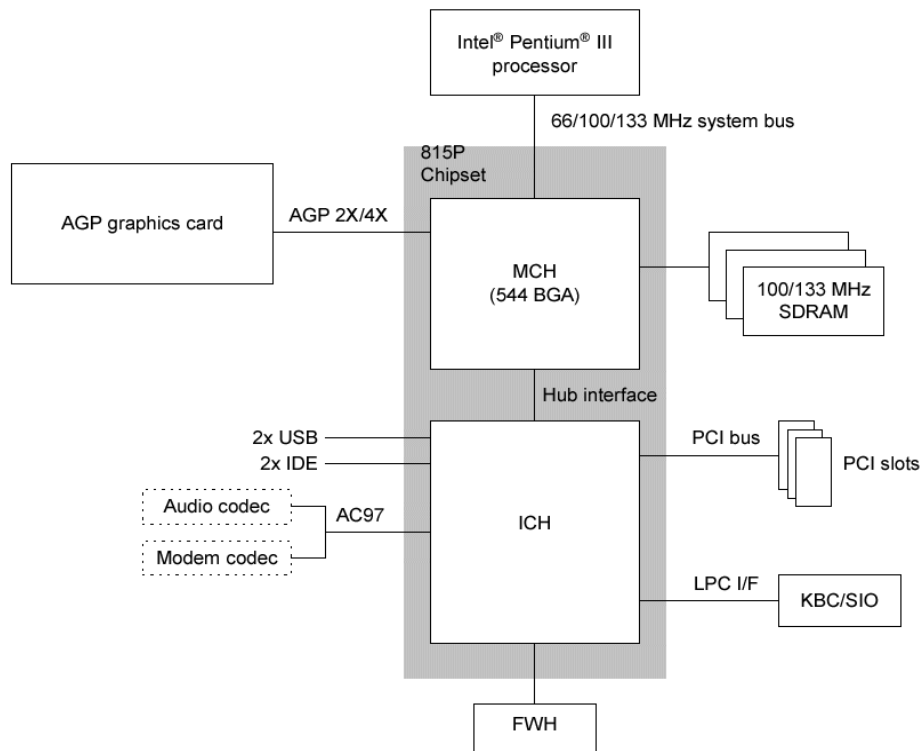


Abbildung 9 Anschluss der FWH and LPC Komponenten [815]

- PCB "Printed Circuit Board" [hDIE]:
Eine dünne Platte auf der elektronische Komponenten festgelötet sind. Die Anschlüsse der Bauelemente oder Pins der Schaltkreise können über Bohrungen auf die Platine gesteckt werden oder auch direkt auf der Oberfläche befestigt werden (SMD).
- PIO "parallel input output":
Besonderer Peripherieschaltkreis, der byteweisen Datentransfer unterstützt. Der *IDE-Flasher* verwendet den 82C55 PIO.
- SMD "surface mounted devices":
Kleine elektronische Bauelemente, die direkt auf die Leiterplatte aufgelötet werden. Im Gegensatz zur konventionellen Bestückung braucht man keine Bohrungen für die Pins. SMD-Bauelemente benötigen daher weniger des wertvollen Platzes auf der Leiterplatte.

10. Quellennachweis

[hATA] ATA-Schnittstelle im Detail: <http://www.tecchannel.de/hardware/459/>

[hDIE] Online Dictionary: <http://dict.die.net/>

[815] Intel 82815 Chipset Platform Design guide: 29825101.pdf

11. FAQ

F: Ist die Jumperstellung 1111 eine Art Standardeinstellung?

A: Ja, diese Stellung kann in den meisten Fällen verwendet werden. Lediglich bei der Programmierung einiger Chips sind besondere Einstellungen notwendig. Wie auch immer, eine allgemeine Aussage, dass diese Stellung immer funktioniert kann nicht gegeben werden.

F: Fehlermeldung "IDE-PORT1 ist frei. PIO-Hardware nicht gefunden." Was kann man jetzt tun?

A: Die genaue Fehlermeldung sieht meist folgendermaßen aus:

```
...  
IDE-PORT1 ist frei. PIO-Hardware nicht gefunden.  
- Kabel überprüfen  
- Ist die Orientierung Kabels korrekt?  
- Ist die Hardware angeschlossen?  
Fehler: 4e37f0 ffffff  
...
```

An der Stelle ist noch einmal genau der Anschluss des IDE-Flasher zu prüfen und ob der richtige Port für die Software ausgewählt wurde.

In seltenen Fällen wird aber auch die Hardware auf dem Mainboard vom BIOS deaktiviert, wenn kein IDE-Gerät gefunden wurde. Eine andere Möglichkeit für die automatische Deaktivierung ist z.B. auch die BIOS-Option „Primary/Secondary IDE-Port – **AUTO**“. Diese Option sorgt u.a. auch dafür, dass wenn kein CD oder Festplattenlaufwerk beim Startup gefunden wird, auch keine Hardware aktiviert wird.

Probleme treten u.a. bei einigen DELL-Computern auf. Hier wird in jedem Fall die Hardware deaktiviert, sollte kein CD oder Festplattenlaufwerk gefunden werden.

Ein möglicher Weg dieses Verhalten zu umgehen, ist es während des Einschaltvorganges vorübergehend ein CD-Laufwerk als Dummy anzuschliessen und so das BIOS zu überlisten. Danach wird gewartet, bis das BIOS seine Checks beendet hat und kurz bevor Windows gestartet wird, muss das Laufwerk wieder entfernt werden.

12. Kontakt

LOET.DE
Andreas Ziermann
Marktstrasse 3
06369 Wulfen
Germany
<http://www.loet.de>
EMail: b@loet.de

