

# **In-Out**

Andreas Heinrich

Copyright © 1993-96 by Andreas Heinrich

**COLLABORATORS**

	<i>TITLE :</i> In-Out		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Andreas Heinrich	June 8, 2025	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1 In-Out</b>	<b>1</b>
1.1 In - Out - Board . . . . .	1
1.2 Was ist neu ? . . . . .	1
1.3 Einführung . . . . .	2
1.4 Technisches . . . . .	2
1.5 Anwendungsmöglichkeiten . . . . .	3
1.6 Aufbau der Schaltung . . . . .	3
1.7 Demoprogramme . . . . .	4
1.8 Step By Step - Demoprogramm . . . . .	4
1.9 Copyright und Vertrieb . . . . .	4
1.10 Warnung ! Bitte lesen. . . . .	5
1.11 Adresse des Autors . . . . .	5

---

# Chapter 1

## In-Out

### 1.1 In - Out - Board

Bauanleitung für ein In-Out-Board

von Andreas Heinrich

Erstveröffentlicht im August 1993  
1.Update am 29. August 1995  
2.Update am 21. Februar 1996

Was ist neu ?

Inhaltsverzeichnis

Einführung  
Technisches  
Anwendungsmöglichkeiten  
Aufbau der Schaltung  
Basic-Demoprogramme  
Step By Step - Demoprogramm  
Copyrights und Vertrieb  
Warnung ! Bitte lesen.  
Adresse des Autors

### 1.2 Was ist neu ?

Auf besonderen Wunsch wurden die Basic-Listings auch in  
>> Maxon Basic 3 << umgesetzt.

Da dieser Basic-Dialekt Amiga-Basic kompatibel sein soll,  
können evtl. auch die Maxon-Basic 3 Listings in den Amiga-  
Basic Interpreter geladen werden. Ob diese so lauffähig  
sind, kann ich allerdings nicht versprechen, da ich Amiga-  
Basic nicht mehr besitze.

---

## 1.3 Einführung

Was beim C 64 der Userport war, ist beim Amiga der .??.?..

Der Amiga hat zwar verschiedene Schnittstellen zur Außenwelt, die sich vom Rechner Typ mehr oder weniger unterscheiden, aber meistens lassen sich diese nur schwer programmieren und / oder der Hardwareaufwand wird zu aufwendig.

In dieser Baubeschreibung wird der Parallelport für die Ein- und Ausgabe mißbraucht.

## 1.4 Technisches

Mit diesem Multi-In-Out-Board ist es möglich, über die Parallel-Schnittstelle Daten oder Ereignisse abzufragen und diese in einem eigenen Programm zu verarbeiten. Die Ausgabe erfolgt optisch durch Leuchtdioden oder es werden Relais angesteuert. ( Man kann, wenn man es braucht, auch AD bzw. DA Wandler nehmen.)

\*\*\*\*\*

Technische Beschreibung:

In-Out-Board mit einen 8 Bit breiten Eingang, zwei 8 Bit breiten Ausgängen sowie eines bidirektionalen Ports.  
Auf der Platine sind noch zwei Adressen frei für zusätzliche Ports.

Schaltungsbeschreibung:

Die Signale Busy, Pout und Sel am Parallel - Port werden an den BCD - zu Dezimal - Decoder ( IC 1 ) geführt. Dieser gibt dann dementsprechend das Port IC zum Lesen oder Schreiben frei.  
Der Tri - State Octal Buffer ( IC 2 ) ist als Eingangsport 2 geschaltet.  
IC 3 ein Octal Bus Transc/N-NV ist der bidirektionale Port 1. Mit IC 6 erfolgt die Steuerung, ob gelesen oder geschrieben werden soll. ( Adresse 0 = Lesen, Adresse 1 = Schreiben )  
Wird auf diesem Port geschrieben, liegt das Signal nur solange an, wie die Freigabe durch IC 1 erfolgt. IC 4 und 5 jeweils ein Octal D-Flip-Flop speichert im Gegensatz zu IC 3 das Datenwort, bis es überschrieben wird. Die Leuchtdiode leuchtet, wenn keiner der Ports aktiv ist.

Achtung:

Die optische Ausgabekontrollschaltung zeigt die Daten invertiert an. Das heißt, wenn 0 geschrieben wird, leuchten alle LED's.

\*\*\*\*\*

Das In-Out-Board wurde mehrfach aufgebaut und funktionierte im Dauertest fehlerfrei an einem A500, A500+, A600, A4000, mit Kick/WB 1.3, 2.04, 2.1, 3.0, 3.1.

Ich nehme an, daß es bei den anderen Amiga-Modellen auch funktioniert. Eine Garantie dafür kann ich aber nicht geben.

\*\*\*\*\*

## 1.5 Anwendungsmöglichkeiten

Anwendungsmöglichkeiten wären:

Weckfunktion -> Radio wird zu einer bestimmten Zeit eingeschaltet und dann etwas später die Kaffeemaschine, Temperaturerfassung, Alarmanlage, Ansteuerung eines LCD-Displays, Aquariumbeleuchtung, Pumpen an und ausschalten, Aufbau einer Wetterstation, Stepmotoren ansteuern und ... und ... und.

Es gibt wahrscheinlich hunderte von Möglichkeiten.

## 1.6 Aufbau der Schaltung

Zum Aufbau:

Nach dem Bohren (0,8 mm) der Pins auf der Platine kann diese bestückt werden. Zuerst sollten die 7 Drahtbrücken gesetzt und eingelötet werden. Danach kann man die IC-Fassungen, die aktiven und die passiven Bauteile einsetzen bzw. verlöten.

Die Anschlüsse der Ports sind jeweils neben den Pins des entsprechenden IC's (siehe Platinenlayout). Das ist zwar nicht schön, aber durch diese Maßnahme konnte das Layout klein und übersichtlich gehalten werden. Außerdem wurden so ca.  $5 * 8 = 40$  Drahtbrücken eingespart. Ich habe z.B. Flachbandkabel direkt auf die Platine eingelötet und dieses mit einer 9 poligen Sub-D-Buchse versehen. So kann man bei Versuchsaufbauten die Verbindung schnell lösen.

Das Anschlußkabel zum Parallelport sollte nicht länger als 1,5 m sein, da es evtl. zu Störungen kommen kann. Aber man kann auch, wenn es nicht anders geht, ein längeres ausprobieren und bei Bedarf kürzen. Die nichtgebrauchten Gattereingänge von IC 6, müssen auf Masse gelegt werden. Die Gatter sind sonst in einem undefinierten Zustand und fangen an zu schwingen. Das sind die Pin's 1+2, 4+5 und 9+10.

Das Netzteil sollte mindestens 200 mA bringen. Besser wäre ein Stärkeres. Denn wenn viele Relais oder andere Schaltungen auch noch mit Strom versorgt werden müssen, reichen 200 mA nicht mehr aus. Wichtig ist eine stabilisierte Gleichspannung von 5 Volt. Da die meisten Steckernetzteile diese nicht haben, muß ein Spannungsregler und zwei Kondensatoren hinzugefügt werden. ( Siehe Schaltplan ) Beim Umgang mit 220 Volt ist mit äußerster Vorsicht umzugehen und die VDE-Vorschriften zu beachten.

Denn:

Man kann mit 220 Volt nicht nur seinen Rechner killen,  
sondern auch sich selbst.

Sollen die Relais höhere Lasten schalten, ist die Kontaktbelastbarkeit

zu beachten. Parallel zu den Relaiskontakten sollten Funkentstörkondensatoren von 0,1 uF/250V angebracht werden.

Wenn alles fertig aufgebaut und verdrahtet, sowie die LED-Kontrolle an Port 3 und die Eingabeschalter angeschlossen sind, steht einem Test nichts mehr im Wege. Man kann nun die Demoprogramme starten und die Hardware testen.

## 1.7 Demoprogramme

Zu den Demoprogrammen ist eigentlich nicht viel zu sagen.

Die Demos sind in Basic geschrieben und man braucht nur Grundkenntnisse in dieser Sprache. Aber auch in Assembler, C oder sonstwas wird das Programmieren des In-Out-Boards keine Probleme verursachen. Es werden nur an bestimmte Adressen Daten geschrieben bzw. gelesen. In Basic werden dabei die Befehle Peek und Poke verwendet.

Nach dem Durchlesen des Listings dürfte der Programmablauf ersichtlich werden und das eigene Programmieren gelingen.

## 1.8 Step By Step - Demoprogramm

Mit dem Demoprogramm "Step By Step" ist eine Schrittmotoransteuerung möglich. Ausgelegt ist es für 4 Motoren und es kann auf dem Monitor die Drehung des Motors mit der Maus gesteuert werden.

Dieses Programm ist eine Beta-Demo-Version, die noch nicht alle Funktionen wie Editieren und Sichern integriert hat und noch eines graphischen Faceliftings bedarf. Das Programm kann allerdings auch ohne vorhandene Hardware angetestet werden, da die einzelnen Step's auf den 4 Motormonitoren verfolgt werden können.

Allerdings sollte ein angeschlossener Drucker abgeschaltet sein. Für jeden anzuschließenden Motor muß die Treiberschaltung wie im Verzeichnis Pictures/Schrittmotor-Ansteuerung nachgebaut werden.

Der Anschluß an das In-Out-Board muß so aussehen:

Motor 1	IC 4	D0 = Setzen	D1 = Takt	D2 = Links/Rechts
Motor 2	IC 4	D4 = Setzen	D5 = Takt	D6 = Links/Rechts
Motor 3	IC 5	D0 = Setzen	D1 = Takt	D2 = Links/Rechts
Motor 4	IC 5	D4 = Setzen	D5 = Takt	D6 = Links/Rechts

## 1.9 Copyright und Vertrieb

---

Diese Anleitung mit dem Amigaguidetext, den Demoprogrammen, sowie des Platinenlayouts und den entsprechenden Schaltplänen mit dazugehöriger Bauanleitungen & Bauteillisten und Texten sind  
Copyright © 1993-96 von Andreas Heinrich.

Alle Rechte vorbehalten.

Diese Bauanleitung ist Shareware und kann frei kopiert und weitergegeben werden. Sollte die Bauanleitung in einer FD-Serie aufgenommen werden, darf der Preis pro Einzeldisk nicht höher als 10.- DM sein.

Bei Gefallen und Benutzung des In-Out-Boards ist allerdings eine Sharewaregebühr von 25.- DM an mich zu zahlen.

Registrierte Anwender bekommen dann von mir die fertig geätzte Platine, das Platinenlayout, einen Bestückungsplan und weiteres auf Disk zugesandt.  
( Die Platine muß allerdings noch gebohrt werden ! )

Durch die Zahlung der Shareware ist der gewerbliche Nachbau in Serie trotzdem nicht erlaubt!

Siehe auch: Adresse des Autors .

## 1.10 Warnung ! Bitte lesen.

\*\*\*\*\*

!!!Achtung!!!Achtung!!!Achtung!!!Achtung!!!Achtung!!!Achtung!!!Achtung!!!

Der Aufbau und die Inbetriebnahme der Schaltung erfolgt auf eigene Gefahr. Obwohl der Aufbau nicht schwierig ist, sollte nur ein versierter Elektronikbastler, der mit einem Lötkolben umgehen und Schaltpläne lesen kann, das Projekt in Angriff nehmen. Kurzschlüsse, falsch angeschlossene Leitungen usw. können schwerwiegende Folgen für die Hardware haben. Gerade bei den neueren Modellen die in SMD-Technik gefertigt werden, kann das eine kostspielige Sache werden. Deshalb noch einmal meine Warnung: Wer sich nicht sicher ist, wie z.B. ein IC-Anschluß zu belegen ist, sollte bei einem Fachmann nachfragen.

\*\*\*\*\*

## 1.11 Adresse des Autors

Bitte schicken Sie das ausgefüllte Registrierungsformular an:

Andreas Heinrich  
Ellerstraße 34  
33615 Bielefeld

Ich bedanke mich im vorraus.