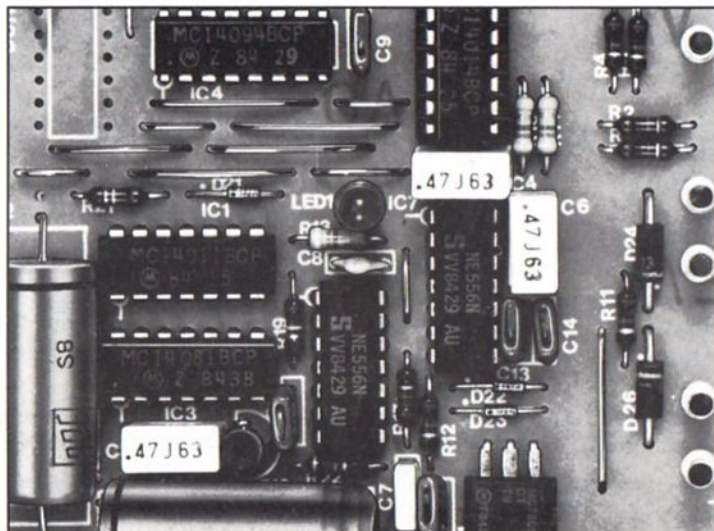


# fischertechnik<sup>®</sup>

## COMPUTING

Interface für Commodore Amiga Computer



## **Inhalt**

Anmerkung	4
Einführung	3
Anschluß des Interface	6
fischertechnik COMPUTING Software	8
Das Diagnoseprogramm	14
Amiga BASIC	18
Anschluß von zwei Interface	18
Benutzung von fischertechnik Elektromechanik und Elektronik	19
Funktionsweise des Interface und des Interface-Treiberprogramms	20
Checkliste	23
Technische Daten	24
Verdrahtungsplan der Interface Ein- und Ausgänge	47

## **Table of Contents**

Important Notice	26
Introduction	27
Connecting the Interface	28
fischertechnik COMPUTING Software	28
The Diagnostic Program	36
Amiga-BASIC	40
Control of two Interfaces	40
Using the fischertechnik Electro- mechanics and Electronics Kits	41
Operation of the Interface and the Interface Driver Program	42
Check List	45
Technical Data	46
Wiring Diagram of the Interface Inputs and Outputs	47

## Bitte kontrollieren Sie die Vollständigkeit Ihres fischertechnik COMPUTING Systems

Lieber fischertechnik Freund, bevor Sie mit den ersten Computer-Experimenten mit fischertechnik COMPUTING beginnen, kontrollieren Sie bitte die Vollständigkeit Ihrer Ausstattung:

Für den... benötigen Sie neben dem Bausatz noch...

- fischertechnik COMPUTING Baukasten**
- das fischertechnik Interface
  - den Adapter für Amiga (Gutschein)
  - die Diskette COMPUTING (Gutschein)
  - das fischertechnik Netzgerät COMPUTING

- fischertechnik COMPUTING Trainingsroboter**
- das fischertechnik Interface
  - den Adapter für Amiga (Gutschein)
  - die Diskette Trainingsroboter (Gutschein)
  - das fischertechnik Netzgerät COMPUTING

- fischertechnik COMPUTING Plotter/Scanner**
- das fischertechnik Interface
  - den Adapter für Amiga (Gutschein)
  - die Diskette Plotter/Scanner (Gutschein)
  - das fischertechnik Netzgerät COMPUTING

- fischertechnik COMPUTING EXPERIMENTAL**
- den Adapter für Amiga (Gutschein)
  - die Diskette EXPERIMENTAL (Gutschein)

Der Gutschein für den Adapter liegt beim Interface bei. Der Gutschein für die Diskette liegt entweder beim Bausatz oder Interface bei. Senden Sie die Gutscheine an die fischerwerke ein, um die benötigten Teile kostenlos zu erhalten. Bei künftigen Bausätzen fragen Sie Ihren Händler oder die fischerwerke.

fischerwerke Abt. fischertechnik D-7244 Tumlingen/Waldachtal

## Please check the completeness of your fischertechnik COMPUTING system.

Dear friend of fischertechnik, before starting right with your first computer control experiments using fischertechnik COMPUTING please check the completeness of your equipment:

Using the... you will require additionally...

- fischertechnik COMPUTING robotics kit**
- the fischertechnik interface
  - the adapter for Amiga (coupon)
  - the diskette COMPUTING (coupon)\*
  - an adequate power supply (see text)

- fischertechnik COMPUTING Training Robot**
- the fischertechnik interface
  - the adapter for Amiga (coupon)
  - the diskette Training Robot (coupon)\*
  - an adequate power supply (see text)

- fischertechnik COMPUTING Plotter/Scanner**
- the fischertechnik interface
  - the adapter for Amiga (coupon)
  - the diskette Plotter/Scanner (coupon)\*
  - an adequate power supply (see text)

- fischertechnik COMPUTING EXPERIMENTAL**
- the adapter for Amiga (coupon)
  - the diskette EXPERIMENTAL (coupon)\*

The coupon for the adapter is part of the interface package. The coupon for the diskette is either part of the construction kit or the interface package. Mail the coupons to the nearest address as given on the coupon to obtain the required items free of charge. Concerning future kits ask your dealer.

\*The above software is available in german language only. Use the programs GRUNDR and DIAGNOSE instead of the programs DRIVER and DIAGNOST referenced later in this manual.

## Anmerkung

Es wurden alle erdenklichen Maßnahmen getroffen, um die Richtigkeit dieser Produkt-Dokumentation zu gewährleisten. Da jedoch die fischerwerke Artur Fischer GmbH & Co. KG ständig an der Verbesserung ihrer Produkte arbeiten, können wir keine Garantie für die Vollständigkeit und Richtigkeit dieser Dokumentation seit ihrem Erscheinen übernehmen.

Amiga, Commodore 64, Commodore 128, Commodore VC20 und CBM sind Warenzeichen der Commodore Electronics Limited.

Apple ist ein eingetragenes Warenzeichen der Apple Computer Inc.

Atari ist ein eingetragenes Warenzeichen der Atari Corporation.

Centronics ist ein eingetragenes Warenzeichen der Data Computer Corporation.

CPC 464, CPC 664 und CPC 6128 sind Warenzeichen von Amstrad Consumer Electronics plc.

CP/M ist ein Warenzeichen von Digital Research Inc.

IBM und IBM PC sind eingetragene Warenzeichen der International Business Machines Corporation.

### Bescheinigung des Herstellers/Importeurs

Hiermit wird bescheinigt, daß das

#### **fischertechnik COMPUTING Interface**

(Gerät, Typ, Bezeichnung)

in Übereinstimmung mit den Bestimmungen der AmtsblVfg 1046/1984

funkentstört ist.

Der Deutschen Bundespost wurde das Inverkehrbringen dieses Gerätes angezeigt und die Berechtigung zur Überprüfung der Serie auf Einhaltung der Bestimmungen eingeräumt.

fischerwerke Artur Fischer GmbH & Co. KG, D-7244 Tumlingen/Waldachtal



## Das fischertechnik COMPUTING Interface

Lieber fischertechnik-Freund,

um mit einem Computer, in Erweiterung seiner Einsatzmöglichkeiten, auch technische Modelle ansteuern zu können, wurde fischertechnik COMPUTING entwickelt. Hierzu gehören die fischertechnik computing Bausätze (Baukasten COMPUTING, Trainingsroboter, Plotter/Scanner, EXPERIMENTAL) ebenso wie die fischertechnik COMPUTING Interfaces und die Software. Es ist jetzt möglich, technische Funktionen und Vorgänge zu simulieren, Aufgaben zu lösen und einfach viel Spaß an computer-gesteuerten Modellen zu haben.

Was braucht man zum Steuern der Modelle? Zunächst einmal das fischertechnik Modell zur Ausführung der Abläufe. Dann einen Personalcomputer, wie Sie ihn besitzen. Er dient der Steuerung und der Koordination. Und dann noch ein Interface als Bindeglied zwischen beiden. Was Sie in den Händen halten, ist das fischertechnik COMPUTING Interface. Steuersignale, die von dem Computer kommen, z.B. "Motor

einschalten!", werden von dem Interface in kräftige Ströme umgesetzt, die in der Lage sind, tatsächlich einen Motor zu bewegen. Wir sprechen in diesem Fall von einer Ausgabe. Die gedachte Blickrichtung verläuft von dem Computer nach außen. Aber auch der umgekehrte Weg ist denkbar und kommt vor. Die Modelle besitzen Taster, Potentiometer etc., um dem Computer Bericht zu erstatten, was an dem Modell draußen vorgeht. Auch hier greift das Interface wieder helfend ein und bereitet diese Signale dergestalt auf, daß sie eine für den Computer verständliche Eingabe darstellen.

Das fischertechnik Interface besitzt nun folgende Leistungsmerkmale:

- Mit ihm lassen sich vier fischertechnik Motoren, Lampen, Elektromagnete etc. steuern.
- Mit ihm kann man acht Taster oder Schalter abfragen.
- Darüberhinaus liefern zwei Eingänge die Werte von stufenlosen Signalgebern wie etwa Potentiometern.

Doch was würden alle elektrischen Verbindungen zwischen Computer und fischertechnik Modell mit Hilfe des Interface nutzen, wenn Sie keine Hilfsmittel hätten, jene zu aktivieren. Die Rede ist von der Software. Dieser Teil liegt in der Form einer Diskette vor. Auf ihr befindet sich ein Programm, das den Sprachschatz Ihres Computers derart erweitert, daß die Steuerung über das Interface tatsächlich erfolgen kann. Dieses Programm wird die Keimzelle Ihrer eigenen Programme sein. Doch damit nicht genug: damit Sie den Einsatz dieser neuen Hilfsmittel studieren und lernen können, sind Beispielprogramme für alle fischertechnik COMPUTING Modelle auch noch untergebracht.

Sie sehen, es wartet eine ganze Menge von interessanten Aufgaben auf Sie. Versuchen Sie es einmal, Sie werden viel Spaß haben.

Ihre  
Artur und Klaus Fischer

## Anschluß des Interface

Das fischertechnik Interface Centronics Port paßt, neben einer Reihe anderer Computer, sowohl an den Amiga 500 als auch an den Amiga 2000. Lediglich bei dem Amiga 1000 gilt es einige Besonderheiten zu beachten, da dieser Computer nicht über einen normgerechten Druckeranschlußstecker verfügt; wir kommen später noch darauf. Zum Zeitpunkt der Drucklegung dieses Anleitungsheftes sind weitere Amiga-Computer angekündigt. Wir gehen davon aus, daß sie weitgehend kompatibel zu den vorgenannten Computern sind und das Interface sich daher auch an diese Computer anschließen lassen wird.

Wir werden im folgenden Text den Amiga 500, Amiga 1000 oder Amiga 2000 oder auch ggf. noch erscheinende kompatible Entwicklungen kurz und bündig mit Amiga bezeichnen.

Der Anschluß des fischertechnik COMPUTING Interface erfolgt an die Parallel-Druckerschnittstelle des Amiga. Hierzu gehen Sie folgendermaßen vor:

- Vergewissern Sie sich, daß der Computer abgeschaltet ist.
- Drehen Sie den Computer herum, so daß die rückseitigen Anschlußstecker vor Ihnen liegen. Suchen Sie den Paral-

lel-Druckeranschluß. Es muß (außer beim Amiga 1000) ein 25-poliger Buchsenstecker sein. Sie werden den Parallel-Druckeranschluß mit anderen 25-poligen Schnittstellen, z.B. der Seriell-Schnittstelle, nicht verwechseln können, da diese Stifte und keine Buchsen besitzen.

- Durch Reibungselektrizität können Sie, ohne es zu merken und ohne daß es für Sie schädlich wäre, auf mehrere 1000 Volt aufgeladen sein. Diese Spannung ist jedoch schädlich für die Schaltkreise in dem Interface und dem Computer. Entladen Sie daher eine eventuell vorhandene elektrostatische Aufladung durch Berühren eines geerdeten Gegenstandes, z.B. eines Wasserhahns, eines Türgriffs, einer Heizung oder des Metallgehäuses des Computers.
- Legen Sie sich nun das Interface zu recht.
- Mit dieser Anleitung und der Diskette zum fischertechnik COMPUTING Modellbaukasten haben Sie auch den Steckadapter passend zu dem Amiga geschickt bekommen. Der Steckadapter wird zwischen Interface und Computer eingefügt. Auf der einen Seite trägt er einen Stecker, der genau zu Ihrem

Interfacekabel paßt, am anderen Ende einen Stecker, der zu dem Druckeranschluß des Amiga paßt. Das Flachbandkabel des Interfaces wird in die zwanzigpolige Steckerleiste des Adapters eingesteckt. Eine Aussparung am Gehäuse und eine Nase am Stecker gewährleisten, daß beides richtig herum zusammengesteckt wird. Der andere Stecker des Adapters wird auf den Parallel-Druckeranschluß aufgesteckt; ein Verpolen ist wegen der Trapezform des Steckers hier ebenfalls nicht möglich.

- Schließen Sie das Interface an das Netzgerät an. Das Interface erwartet ungesieberte Gleichspannung zwischen 6 und 10 Volt. Beim Einsatz von stabili-sierter Gleichspannung sind ca. 8 Volt (belastbar bis ca. 2 Ampere) einzustellen. Um ganz sicher zu gehen, sollten Sie jedoch das fischertechnik Netzgerät COMPUTING verwenden. Stecken Sie den Kabelausgang des Netzgeräts mit dem roten Flachstecker in die mit + gekennzeichnete Buchse des Interface und den Kabelausgang mit grünem Flachstecker in die mit - gekennzeichnete Buchse. Sie haben dabei die Auswahl zwischen zwei Buchsenpaaren;



- welches Sie verwenden, ist gleichgültig.
- Für Modelle mit bis zu zwei Motoren kann auch das fischertechnik Netzgerät mot4 eingesetzt werden, also insbesondere für die Modelle des fischertechnik COMPUTING Baukastens. Sollten Sie aber eigene Konstruktionen mit einer größeren Motorzahl oder die Bausätze Trainingsroboter oder Plotter/Scanner betreiben wollen, brauchen Sie Verstärkung. Entweder wechseln Sie dann doch zu dem kräftigeren fischertechnik Netzgerät COMPUTING oder Sie speisen mit einem zweiten fischertechnik Netzgerät mot4 in die noch freien Anschlußbuchsen ein.
  - Bei Einsatz der Netzgeräte mot4 verwenden Sie die den Baukästen beigelegte zweiadrige Leitung und Flachstecker zur Herstellung des Anschlußkabels.
  - Verbinden Sie das fischertechnik COMPUTING Modell mit dem Interface. Hierzu dient das den fischertechnik COMPUTING Bausätzen beigefügte zwanzigadrige Flachbandkabel. Dieses Kabel ist auch als Einzelteil erhältlich.
  - Die Reihenfolge, in der Sie das Interface und den Computer nun einschalten, spielt keine Rolle. Wenn Sie das Inter-

- face mal nicht benutzen und mit anderen Programmen arbeiten, brauchen Sie das Interface dennoch nicht abkabeln. Lassen Sie in diesem Fall einfach das Interface ausgeschaltet.
- Nebeneffekte des Interface: Solange das Interface eingesteckt ist, können Sie selbstverständlich keinen Drucker benutzen, der auch an diesem Anschluß eingesteckt wird. Allerdings sind im Handel Umschalter erhältlich, mit denen Sie manuell zwischen mehreren Druckern umschalten können. Einen solchen Umschalter können Sie auch zwischen den Amiga und das Interface schalten. Damit ersparen Sie sich, jedesmal die Anschlußkabel von Drucker und Interface auszutauschen. Der Ausgang solcher Druckerumschalter ist meist nach der 36-poligen Centronics-Norm aufgebaut. Bei Verwendung eines Druckerumschalters sollten Sie sich also den Steckadapter in 36-poliger Ausführung bestellen.
  - Bei dem Amiga 1000 wird anstelle des Druckeranschlusses mit 25-Buchsenöffnungen ein Druckeranschluß mit 25 Stiften verwendet. Nahezu alle Leitungen sind genauso wie bei dem normgerechten Stecker belegt. Wenn Sie

- Schwierigkeiten haben sollten, bei Ihrem Amiga-Händler ein Übergangsstück auf den normgerechten Druckeranschluß zu erwerben, so können Sie zwei Steckverbinder mit 25-poligen Buchsen nach folgender Vorschrift verbinden:
- Buchse 1 bis 13 und 23 des ersten Steckverbinders verbinden Sie mit der jeweiligen Buchse der gleichen Nummer des zweiten Steckverbinders.
  - Buchsen 14 bis 22 sind Masse. Sie können auf beiden Steckverbindern untereinander verbunden und mit einer Leitung zum anderen Steckverbinder geführt werden.
  - Die Buchsen 24 und 25 beider Steckverbinder bleiben frei.

Noch einmal zur Erinnerung:

Auch beim Arbeiten mit den Modellen sollten Sie sicherheitshalber immer zuerst eine eventuell vorhandene elektrostatische Aufladung ableiten, indem Sie einen geerdeten metallischen Gegenstand berühren. Doch nun genug von der Hardware, im nächsten Abschnitt wollen wir uns die fischertechnik COMPUTING Software vornehmen.

## fischertechnik COMPUTING Software

Wer sich schon einmal mit dem Gedanken befaßt hat, irgendwelche Geräte oder Modelle mit dem Computer zu steuern, wird aus eigener Erfahrung wissen oder von anderen Computerfreunden gehört haben, daß dies alles gar nicht so einfach sei. Man brauche eine genaue Kenntnis des Computers, des Mikroprozessors und der Ein- und Ausgabebausteine sowie der Maschinsprache für diese Aufgabe. Bislang stimmte diese Aussage und dadurch wurde leider auch mancher von diesem interessanten Kapitel der Computerei abgehalten. Jetzt gibt es diese Schwierigkeit nicht mehr. In dem Lieferumfang des jeweiligen fischertechnik COMPUTING Baukastens sind die passenden Steuerprogramme enthalten (Gutschein ausfüllen und an die fischertechnik schicken - die zu dem Baukasten passende Diskette kommt postwendend). Die vorliegende Anleitung sagt Ihnen zudem, wie Sie solche Steuerprogramme selbst schreiben.

Um sicher zu sein, daß Sie nicht bei einem versehentlichen Löschen oder einer Beschädigung der Diskette ohne Software dastehen, sollten Sie sich von der fischertechnik COMPUTING Diskette eine Kopie anfertigen. Benutzen Sie hierfür das auf der Amiga-DOS Systemdiskette (Workbench)

mitgelieferte Programm DISKCOPY. Verfahren Sie exakt nach den Angaben der Amiga-DOS Anleitung und des Programms. Am Schluß sollten Sie die Rename-Funktion benutzen, um das "copy of" vor dem Diskettenamen zu tilgen. Wenn Sie die Kopie vorliegen haben, verstauen Sie die fischertechnik COMPUTING Original-Diskette an einem sicheren Platz, an den keine natürlichen Feinde der Disketten, wie Sand, Hitze, Katzen oder Magnetfelder hinkommen können. Arbeiten Sie fortan mit der Kopie und benutzen Sie die fischertechnik COMPUTING Diskette nur, um gegebenenfalls eine weitere Kopie zu ziehen. Kopien sind vor allen Dingen dann nützlich, wenn Sie die Programme noch abändern wollen und die Originaldiskette bestehen bleiben soll. Das Kopieren beschränkt sich jedoch wohlgemerkt nur auf den eigenen Gebrauch.

Einige fischertechnik Programme legen Daten auf Diskette ab. Wenn Sie schon dabei sind, sollten Sie sich einige Disketten formatieren.

Wenn Sie das Interface noch nicht an Ihren Computer wie in dem vorigen Kapitel beschrieben angeschlossen haben, so sollten Sie es nun nachholen. Schalten Sie anschließend den Computer ein und legen Sie

die Workbench des Systems ein.

Nach dem Systemstart legen Sie die fischertechnik-Diskette in ein Laufwerk ein. Wenn Sie nur ein Diskettenlaufwerk besitzen, werden Sie dazu die Workbench wieder herausnehmen müssen. Öffnen Sie das Diskettenfenster. Sie finden in dem Fenster:

Eine Schublade mit den fischertechnik-Programmen; sie trägt den Namen COMPUTING, ROBOTER, PLOTTER, EXPERIMENTAL oder dergleichen und enthält die Beispielprogramme für die fischertechnik COMPUTING Modelle.

Eine Schublade mit den fischertechnik-Quelltexten (Sources); sie enthält Programme für erfahrene Programmierer, die eigene Ideen mit dem fischertechnik Interface realisieren wollen. Diese Schublade müssen sie normalerweise nicht öffnen.

Ein Buch mit dem Namen "Liesmich", das ergänzende Informationen und neueste Erkenntnisse zur fischertechnik COMPUTING Software und den Modellen enthält. Dieses Buch sollten Sie aufblättern (zweimal kurz hintereinander anklicken oder mit Hilfe der Open-Funktion).

Einen Mülleimer (Trashcan, wie bei der Workbench auch). Um ihn zu sehen, müssen Sie das Fenster vergrößern. Dies ist



eine Sicherheitsmaßnahme, damit sie nicht versehentlich Dateien auf der Diskette löschen.

Ein Wohnzimmer mit der Unterschrift "Einrichten".

Um die Kopie Ihrer fischertechnik Diskette funktionsfähig zu machen, legen Sie sich die Workbench-Diskette und die Extras-Diskette Ihres Amiga zurecht. Sie werden sie im folgenden benötigen. Starten Sie das Programm mit dem "Einrichten"-Symbol. Das Programm wird von Ihnen immer das Einlegen der einen oder anderen Diskette fordern - wenn Ihr Amiga mit einem Diskettenlaufwerk ausgestattet ist häufiger, als wenn er mit zwei Diskettenlaufwerken ausgestattet ist. Verfahren Sie genau nach den Anweisungen des Programms. Der komplette Durchlauf des Einrichtens dauert knapp 10 Minuten.

Noch ein Hinweis: Wenn Sie Ihren Amiga nicht in Deutschland sondern in anderen Ländern gekauft haben, wird Ihre Extras-Diskette ggf. nicht ExtrasD heißen. Ändern Sie in diesem Fall die Datei "Einrichten" mit Hilfe eines Texteditors, so daß Ihre Extras-Diskette korrekt angesprochen wird.

Am Ende der Prozedur befinden sich auf der (Kopie der) fischertechnik-Diskette alle Programme, die sie zum Experimentieren

benötigen. Dies sind einige Bestandteile der Workbench und der AmigaBASIC-Interpreter. Legen Sie jetzt die Diskette in das Laufwerk DF0: und lösen Sie einen Neustart des Computers aus. Dies geschieht durch den Tastendruck <CTRL> <Amiga links> <Amiga rechts>. Bei späterer Benutzung legen Sie die so entstandene fischertechnik Workbench gleich zum Einschalten in das Diskettenlaufwerk DF0:. Sie sollten nicht versuchen, mit einer anderen Workbench als der fischertechnik Workbench zu starten, wenn Sie anschließend fischertechnik Programme aufrufen. Das Startprogramm der fischertechnik Diskette legt nämlich während des Startvorgangs einige Programme und Dateien in der Ram-Disk ab. Diese Programme würden fehlen, wenn das System nicht mit der fischertechnik Workbench gestartet würde.

Hier noch ein Hinweis für Besitzer von ausländischen Amigas und des Amiga 1000: Bei der Startprozedur wird die deutsche Tastaturbelegung installiert (dies betrifft insbesondere die Umlaute und die Lage der Sonderzeichen). Sollte dies für Ihren Amiga nicht geeignet sein, so ändern Sie das Kommando "setmap d" in der Datei "s/startup-sequence" der fischertechnik Diskette. Schauen Sie in der Datei "s/star-

tup-sequence" Ihrer Workbench nach, was dort verwendet wird (z.B.: setmap f).

Nach dem Laden der fischertechnik Workbench öffnen Sie erneut das Disketten-symbol. Jetzt sollte sich auch der AmigaBASIC interpreter im Fenster befinden. Öffnen Sie dann die fischertechnik Schublade (COMPUTING, ROBOTER, PLOTTER, EXPERIMENTAL oder dergleichen). Rufen Sie das Programm FISCHER mit einem Doppelklick auf. Das Programm FISCHER wird geladen. Dabei wird automatisch der Interpreter AmigaBASIC mitgeladen. Bei Start des Programms FISCHER erhalten Sie auch noch einmal Auskunft über Hardware und Software Ihres fischertechnik COMPUTING Baukastens.

Laden Sie nun das Grundprogramm GRUNDPR.BAS. Hierzu geben Sie entweder im Dialogfenster des AmigaBASICA den Befehl

### **RUN"GRUNDPR"**

ein oder Sie schließen das bisherige Programmfenster und doppelklicken das Symbol für GRUNDPR.

Das Programm GRUNDPR steht für Grundprogramm und richtet alles ein, was Sie zum Datenaustausch mit dem Interface benötigen. Das Grundprogramm ist auch Bestandteil aller folgenden Beispielpro-

gramme. Schauen Sie sich das Grundprogramm im Listfenster an. Sie sehen in einer der ersten Zeilen den Programmtext

#### **LIBRARY "ram:fischer.library"**

Diese Zeile ist der eigentliche Schlüssel zu dem Datenaustausch mit dem Interface. Sie verweist auf eine Bibliothek mit Ein- und Ausgabekommandos, die sich auf der Ram-Disk befindet. Während des Starts der Diskette wurde sie dorthin kopiert. Die Verwendung der Ram-Disk bringt Vorteile: Sie ist immer verfügbar und hat schnelle Zugriffszeiten.

Die Bibliothek fischer.library enthält die eingangs erwähnten Detailkenntnisse über die Ein- und Ausgabebausteine des Computers. Durch ihre Benutzung vereinfacht sich die Steuerung des Interfaces ganz wesentlich. Wir wollen an dieser Stelle anmerken, daß die fischer.library der verschiedenen Disketten durchaus unterschiedlich sind. Je nach Thema des Baukastens enthält sie spezielle Kommandos: Roboterbefehle für den Trainingsroboter, Meßwerterfassungsbefehle für EXPERIMENTAL etc. Sie dürfen somit die Bibliotheken der Disketten nicht austauschen.

Was wir im folgenden beschreiben, ist ein Grundgerüst an Befehlen, wie es in jeder Bibliothek vorkommt. Starten Sie hierzu

das Grundprogramm. Nach Durchlauf durch das Programm wird sich der Amiga wieder mit seinem Bereitzeichen melden. Äußerlich scheint sich nichts geändert zu haben. Dennoch besitzt Ihr Amiga durch Aktivierung der Bibliothek nun einige neue Befehle, die vorher in dieser Form nicht im BASIC enthalten waren. Diese Befehle sind genau auf Ihren Amiga und das fischertechnik COMPUTING Interface abgestimmt. Die Befehle benutzen das CALL-Kommando und die IN-Funktion für den Aufruf von Maschinenprogrammen. Das Grundprogramm selbst ist in BASIC geschrieben und erzeugt im Variablenspeicher des AMIGA die Parameter jener Maschinenprogrammaufrufe (Ex, Ey, E1 ... Rechts, Links, Aus).

Sie brauchen daher nur noch die folgend beschriebenen BASIC-Befehle zu beherrschen:

Der Motorausgang M1 wird angesteuert mit:

**CALL M1(EIN)      CALL M1(AUS)  
CALL M1(RECHTS) CALL M1(LINKS)**

Die Kommandoparameter bezeichnen den Motor, in diesem Fall Motor 1, und die Betriebsart.

Die entsprechenden Befehle mit M2, M3 und M4 steuern die übrigen drei Ausgänge.

Außerdem sollten Sie sich merken, daß "EIN" ebenfalls immer Rechtslauf bewirkt. Die 10 Eingänge des Interface werden mit Hilfe der IN-Funktion eingelesen. Die Funktion

#### **IN(E1)**

ist 1, wenn der Eingang E1 des Interface mit +5V verbunden ist. Sonst zeigt IN(E1) den Wert 0. Entsprechend erhält man mit IN(E2) ... IN(E8) die Zustände der übrigen Digitaleingänge.

Die Analogeingänge EX und EY werden über je einen Sensor, z.B. ein Potentiometer (4.7 k $\Omega$ ), mit +5V verbunden. Die Funktionen

#### **IN(EX)**

#### **IN(EY)**

führen dann einen Wert aus dem Bereich 20 bis 220, je nach Stellung der Potentiometer. Obige Grenzwerte können auch leicht anders ausfallen, je nach Streuung der Bauelemente.

Wird z.B. ein Roboterarm wie der aus dem fischertechnik COMPUTING Baukasten von einem Motor angetrieben und synchron mit der Bewegung des Arms das Potentiometer an EX verstellt, so kann das Programm, indem es immer wieder die Funktion

#### **IN(EX)**



aufruft, die Bewegung des Roboters genau verfolgen.

Wenn kein Sensor an den Eingang EX bzw. EY angeschlossen ist, wirkt sich aus, daß in der Interfaceschaltung ein 100 k $\Omega$  Widerstand parallel zu dem Eingang geschaltet ist. Der Funktionswert beträgt dann zwischen 4000 und 5000. Wir empfehlen aber, die Analogeingänge nie unbeschaltet zu lassen, wenn sie unbenutzt sind. Sie werden dann am besten direkt mit der Referenzspannung +5V verbunden; dies entspricht einem Widerstand von 0  $\Omega$ .

Der letzte der neuen Befehle ist

#### **CALL INIT**

Dieser wird benutzt, um das Interface in einen wohldefinierten Anfangszustand zu versetzen. Er kann auch benutzt werden, wenn alle Motorkanäle mit einem Male abgeschaltet werden sollen.

Doch nun genug der langen Vorrede. Schließen sie einen fischertechnik Motor über das zwanzigpolige farbcodierte Flachbandkabel an M1 an. Dies sind die gelbe und orange Leitung in der oberen Hälfte des Flachbandkabels. Geben Sie im Dialogfenster ein:

#### **CALL M1(EIN)**

Der Motor wird kurz anlaufen und dann wieder stehen bleiben. Geniessen Sie die-

sen Augenblick, er hat Ihnen das Gefühl gegeben, in kurzer Zeit die kompliziertesten fischertechnik Anlagen mit Ihrem Amiga zu steuern. Sollten Sie nicht zu den Glücklichen gehören, bei denen der Motor anlief? Haben Sie eine Fehlermeldung erhalten? In diesem Fall starten Sie das Grundprogramm erneut. Nach bisherigem Kenntnisstand baut der AmigaBASIC-Interpreter beim ersten Programmdurchlauf nicht alle Verweise auf die CALL-Kommandos auf, so daß die Direkteingabe erst nach mehreren Aufrufen korrekt durchgeführt wird. Im Ablauf eines Programms treten jedoch keine Schwierigkeiten mit dem CALL-Kommando auf.

Doch zurück zu dem Motor. Er war angelaufen und hatte sich kurz darauf wieder abgeschaltet. Wieso? Hatten wir ihn nicht richtig eingeschaltet? Gibt es zum Ausschalten nicht, wie oben beschrieben, einen eigenen Befehl? Nun, der Motor ist zwar stehen geblieben, aber in dem Interface ist nach wie vor gespeichert, daß er eigentlich laufen sollte. Das Interface hat sich selbst "schlafen gelegt". Dies tut es immer, wenn innerhalb einer halben Sekunde kein neuer Befehl kommt. Es geschieht aus Sicherheitsgründen. Stellen Sie sich vor, Sie erproben ein neues Pro-

gramm. Die Wahrscheinlichkeit, daß noch irgendwo ein Fehler im Programm versteckt ist, grenzt an Gewißheit. Der Computer bleibt mit einer leidigen Meldung wie

**Syntax error** **Ok**

stehen. Der Motor, der kurz vorher eingeschaltet wurde, bliebe jedoch nicht stehen und schicke sich an, das schöne Modell zu demolieren. Sie müßten zum Netzgerät hasten und schnell die Spannung abstellen.

Wie beruhigend ist es da zu wissen, daß der Motor von alleine stehen bleiben wird. Auch dann, wenn Sie mit der Stop-Funktion (RUN-Menü mit der rechten Maustaste am oberen Bildrand öffnen und Stop anwählen) den Programmablauf unterbrechen. Und wenn es dann wieder weitergeht, so wird mit dem ersten Befehl das Interface wieder "aufgeweckt" und hat keinen der Motoren vergessen. Der Ablauf kann fortgesetzt werden, als wäre nichts geschehen.

Daß das Interface mit dem Abschalten nicht sofort zur Hand ist, wurde mit Bedacht gewählt. Zwischen den Ein- und Ausgabebefehlen an das Interface werden sich immer wieder Pausen aufgrund von Berechnungen ergeben, die es zu überbrücken gilt. Ob das Interface durch Ein- oder Ausgabebefehle aktiviert wird, können Sie

auch durch einen Blick auf die Leuchtdiode des Interface sagen. Sie dient nicht nur der Spannungsanzeige sondern auch der Betriebsanzeige.

Nun wollen wir noch einen kurzen Blick auf die Eingabebefehle werfen. Schließen Sie zwischen E1 (der braunen Leitung am unteren Rand) und +5V (der roten Leitung in der Mitte des Flachbandkabels) einen Taster an.

Probieren Sie aus (bei Fehlermeldungen nicht verzagen, s.vorige Seite):

#### **PRINT IN(E1)**

Je nachdem, ob der Taster zwischen E1 und +5V bei der Betätigung der Returntaste des Computers gedrückt war oder nicht, wird auf dem Bildschirm eine 1 oder eine 0 ausgegeben.

Wenn an dem Ausgang noch von vorher der Motor angeschlossen war, wird er sich wieder rühren. Auch Eingabebefehle aktivieren wieder die Ausgänge des Interface! Nun schließen Sie bitte ein Potentiometer

4,7 k $\Omega$  wie es z.B. im fischertechnik COMPUTING Baukasten enthalten ist, zwischen EX und +5V an. Stellen Sie den Schleifer in eine mittlere Stellung und geben Sie ein:

#### **PRINT IN(EX)**

Die Zahl, die jetzt auf dem Bildschirm erschienen ist, muß zwischen 20 und ca. 220

liegen.

Das Grundprogramm, das Sie eben benutzt haben, werden Sie immer wieder brauchen. Jedes Programm, das mit dem fischertechnik COMPUTING Interface Modelle steuern soll, beginnt mit diesem oder einem ähnlichen Vorspann, der die neuen Befehle installiert. Bei den Beispielpogrammen auf der Diskette ist das Grundprogramm jeweils schon enthalten. Bei Programmen, die Sie selbst schreiben, beginnen Sie mit dem Grundprogramm und schreiben anschließend weiter.

Damit Sie das Potentiometer leichter beobachten können, wollen wir nun auf diese Weise das erste fischertechnik COMPUTING Programm schreiben. Das Grundprogramm befindet sich in dem Computer. Gehen Sie mit dem Mauszeiger in das Listfenster, drücken die linke Maustaste, setzen den Cursor an das Programmende und geben ein:

**lesen:**

#### **PRINT IN(EX)**

#### **GOTO lesen**

Geben Sie anschließend das Kommando

#### **RUN**

Es dauert einen ggf. kurzen Moment, bis die Bibliothek fischer.library und das Titelbild geladen sind. Und dann geht es los. Im

Nu wird der Bildschirm mit Zahlen gefüllt, die ständig nach oben hinausgeschoben werden. Wenn Sie jetzt das Potentiometer in die Hand nehmen und den Schleifer drehen, werden Sie die Veränderung der Zahlen beobachten. Drehen Sie von einem Anschlag zum andern. Die eingelesenen Zahlen sollten zwischen 20 und 220 liegen. Zum Benden des Programms müssen Sie die Stop-Funktion benutzen.

Für diejenigen, die etwas genauer die Abläufe verstehen wollen und nicht nur die auf der Diskette vorliegenden Programme benutzen wollen, halten wir hier nun noch Detailinformationen bereit. Die Funktion des Grundprogramms besteht darin, die Verbindung zu der fischertechnik Bibliothek herzustellen. Diese liegt ursprünglich als Datei fischer.library und fischer.bmap in dem Verzeichnis fischer/libs der fischertechnik-Diskette vor. Bei dem Einschalten des Computers wird sie gemäß den Anweisungen der Befehlsdatei startup-sequence in die Ram-Disk kopiert. Die Parameter INIT, M1, M2, M3, M4 und IN sowie ggf. weitere Namen sind Einsprungadressen der Bibliothek. Neben der Bibliothek selbst werden auch noch die genormten Parameter E1, E2, E3, E4, E5, E6, E7, E8, EX, EY, Ein, Rechts, Links und Aus gesetzt. Außer-



dem sind alle Variablen als Ganzzahlvariablen (DEFNG) definiert.

Für selbstgeschriebene BASIC-Programme sind daher unbedingt folgende Einschränkungen zu beachten:

Ähnlich wie Sie auch nicht die reservierten BASIC-Schlüsselwörter wie PRINT oder STOP als Variablen verwenden dürfen, ist auch der Gebrauch der obigen Parameter als Variable verboten. Die Einschränkung betrifft jedoch nicht Variablen anderen Typs (Strings, Real), so daß

#### **Aus\$, Rechts!, E1!...**

durchaus vorkommen dürfen.

Wer noch weiterforschen will, weil er z.B. die elektronischen Abläufe im Interface verstehen will oder eine Adaption an andere Programmiersprachen vornehmen will, wird auf die Beschreibung des Interface und des Quelltextes verwiesen.

Die letzte Funktion des Grundprogramms besteht in dem Einschalten des Interface und dem Ausschalten aller angeschlossenen Verbraucher. Dies wird durch den Befehl

#### **CALL INIT**

bewirkt. Damit wird das Interface sozusagen in betriebsbereitem Zustand an ein hier anschließendes Benutzerprogramm übergeben. Außerdem haben wir noch die La-

deroutine für das Überschriftenbild angeführt, die Sie vielleicht auch in eigenen Programmen verwenden wollen. Auch alle Beispielprogramme der Diskette sind nach diesem Muster aufgebaut. Studieren Sie diese, wenn Sie sich Anregungen holen wollen.

```
REM *****
REM ***   Interface-Treiber   ***
REM ***           für           ***
REM ***   Commodore AMIGA   ***
REM *****
REM
DECLARE FUNCTION In& LIBRARY
LIBRARY "ram:libs/fischer.library"
DEFNG a-z
EX=160
EY=144
REM
EA=255
REM Mit IN(EA) können alle Digital
      eingänge
REM gleichzeitig eingelesen werden.
E8=128
E7=64
E6=32
E5=16
E4=8
E3=4
E2=2
E1=1
```

```
Rechts=85
Links=170
Aus=255
Ein=85
REM Aufruf der Motoren mit CALL M1-M4
REM oder CALL MOTOR(1-4,Richtung)
CALL Init
REM *****
REM *** Bildaufbau ***
REM *****
SCREEN 1,640,256,2,2
WINDOW2,"                                GRUNDPROGRA
                                MM",,0,1
PALETTE 0,208/256,208/256,208/256
                                `hellgrau
PALETTE 1,0,0,0                                `schwarz
PALETTE 2,0,128/256,224/256                                `blau
PALETTE 3,240/256,0,0                                `rot
REM
REM Titel als BOB's einlesen
REM
OPEN "TITEL" FOR INPUT AS 1
OBJECT.SHAPE 1,INPUT$(6506,1)
OBJECT.SHAPE 2,INPUT$(1790,1)
OBJECT.SHAPE 3,INPUT$(794,1)
CLOSE 1
OBJECT.X 1,34
OBJECT.Y 1,10
OBJECT.X 2,145
OBJECT.Y 2,48
OBJECT.X 3,129
OBJECT.Y 3,233
OBJECT.ON
OBJECT.CLOSE
STOP
```

## Das Diagnoseprogramm

Wenn Sie ein fischertechnik computing Modell aufgebaut haben, werden Sie vielleicht die Erfahrung machen, daß es nicht so läuft, wie Sie sich das vorgestellt haben. Wen wundert das bei dieser großen Zahl von Leitungen, die zwischen Modell und Interface hin und her laufen. Und wenn nur ein Taster vertauscht wäre, die verblüffendsten Effekte könnte dies zur Folge haben. Doppelt schwierig wird die Situation, wenn die Programme selbst geschrieben sind. Wo soll man da mit der Suche anfangen? In der Hardware oder der Software?

Damit Sie die Hardware eindeutig und komfortabel testen können, wurde das Diagnoseprogramm entwickelt. Es liegt auf jeder fischertechnik Diskette als DIAGNOSE vor. Laden Sie dieses Programm immer zum Austesten eines Modells. Sie können mit ihm sämtliche Eingänge beobachten und feststellen, ob ihr Verhalten mit Ihren Vorstellungen übereinstimmt.

Die Motorausgänge werden mit der Maus angewählt. Jedem Motor ist ein Kasten zugeordnet. In dem jeweiligen Kasten befindet sich das Symbol für Motorstop in der Mitte. Die Symbole rechts davon wählen Rechtslauf an, die links davon den Linkslauf. Sie haben dann noch die Wahl zwischen zwei Betriebsarten. Das gestrichelte

Symbol bewirkt den Motorlauf nur solange die Maustaste gedrückt ist. In dieser Betriebsart kann präzise positioniert werden. Das durchgezogene Symbol wählt Dauerbetrieb an. Man wird es verwenden, wenn z.B. ein Greifmagnet eingeschaltet bleiben soll, während die Roboterachsen bewegt werden. Zum Beenden des Dauerbetriebs wählen Sie das Stoppsymbol in der Mitte.

Zum gleichzeitigen Ausschalten aller Motoren existiert ein großer Not-aus-Kasten. Mit einem weiteren Kasten kann das Ende des Programms angewählt werden.

```
REM *****
REM *** Interface-Treiber ***
REM *** für ***
REM *** Commodore AMIGA ***
REM *****
REM
DECLARE FUNCTION IN% LIBRARY
LIBRARY "ram:libs/fischer.library"
DEFLNG a-z
Ex=160
Ey=144
EA=255
E8=128
E7=64
E6=32
E5=16
E4=8
E3=4
E2=2
E1=1
Rechts=85
Links=170
Aus=255
Ein=85
CALL Init
REM
REM *****
REM *** Bildaufbau ***
REM *****
SCREEN 1,640,256,3,2
WINDOW 2,"
GNOSE",,0,1
PALETTE 0,208/256,208/256,208/256
PALETTE 1,0,0,0
PALETTE 2,0,128/256,224/256
```

DIA

```

PALETTE 3,240/256,0,0
PALETTE 6,48/256,48/256,48/256
PALETTE 7,80/256,80/256,80/256
PALETTE 4,128/256,128/256,128/256
PALETTE 5,160/256,160/256,160/256
OPEN "TITEL" FOR INPUT AS 1
OBJECT.SHAPE 1,INPUT$(6506,1)
OBJECT.SHAPE 2,INPUT$(1790,1)
OBJECT.SHAPE 3,INPUT$(794,1)
CLOSE 1
OBJECT.X 1,34
OBJECT.Y 1,10
OBJECT.X 2,145
OBJECT.Y 2,48
OBJECT.X 3,129
OBJECT.Y 3,233
OBJECT.ON
OBJECT.CLOSE
REM*****
REM*
REM* fischertechnik computing
REM*
REM* Diagnose
REM*
REM*Copyright(c)fischerwerke 1987*
REM*
REM*****
REM
REM Funktion:
REM
REM Mit diesem Programm werden
REM sämtliche Funktionen der Model-
REM le überprüft.
REM Alle Eingänge werden angezeigt.
REM Alle Ausgänge werden über die

```

```

REM Maus angesteuert.
REM
REM *****
REM *** Steuerung vorbereiten ***
REM *****
REM
ON BREAK GOSUB ENDE
BREAK ON
GOSUB SPRITES
GOSUB BILDAUFBAU
REM alle Motoren aus
REM Status der 4 Motoren
DIM Sta(4),X(4)
FOR i=1 TO 4
  LET Sta(i)=Aus
NEXT i
REM Start mit Motor 1
LET MOT=1
REM
REM *****
REM *** Anzeige der Eingabe ***
REM *****
REM
Hauptschleife:
  LOCATE 15,10:PRINT IN(E1)
  LOCATE 15,13:PRINT IN(E2)
  LOCATE 15,16:PRINT IN(E3)
  LOCATE 15,19:PRINT IN(E4)
  LOCATE 15,22:PRINT IN(E5)
  LOCATE 15,25:PRINT IN(E6)
  LOCATE 15,28:PRINT IN(E7)
  LOCATE 15,31:PRINT IN(E8)
  LOCATE 15,36:PRINT USING "####";
    IN(Ex)
  LOCATE 15,41:PRINT USING "####";

```

```

IN(Ey)
REM *****
REM *** Steuerung der Ausgabe ***
REM *****
REM
REM Mausabfrage
k=MOUSE(0)
X=MOUSE(1)
Y=MOUSE(2)
REM Angewählter Motor
ZU=Sta(MOT)
IF k THEN
  IF Y>140 THEN
    IF Y>180 AND X<316 THEN
      MOT=2
    END IF
    IF Y>180 AND X>328 THEN
      LET MOT=4
    END IF
    IF Y<172 AND X<316 THEN
      LET MOT=1
    END IF
    IF Y<172 AND X>328 THEN
      LET MOT=3
    END IF
  IF ABS(X-188)<=20 OR ABS(X-452)
  <=20 THEN
    LET Sta(MOT)=Aus
    LET X(MOT)=0
    LET ZU=Aus
  END IF
  IF ABS(X-92)<=20 OR ABS(X-356)
  <=20 THEN
    LET Sta(MOT)=Links
    LET X(MOT)=96

```



```

LET ZU=Links
END IF
IF ABS(X-288)<=20 OR ABS(X-548)
<=20 THEN
LET Sta(MOT)=Rechts
LET X(MOT)=-96
LET ZU=Rechts
END IF
IF ABS(X-140)<=20 OR ABS(X-404)
<=20 THEN
LET ZU=Links
END IF
IF ABS(X-236)<=20 OR ABS(X-500)
<=20 THEN
LET ZU=Rechts
END IF
ELSE
IF X>376 THEN
IF ABS(Y-124)<=16 THEN
CALL Init
FOR i=1 TO 4
LET Sta(i)=Aus
LET X(i)=0
NEXT i
LET ZU=Aus
END IF
IF ABS(Y-100)<=16 THEN ENDE
END IF
END IF
END IF
REM Ausgabe an gewählten Motor
CALL MOTOR(MOT,ZU)
OBJECT.X 1,181-X(1)
OBJECT.X 2,181-X(2)
OBJECT.X 3,444-X(3)

```

```

OBJECT.X 4,444-X(4)
GOTO Hauptschleife
BILDAUFBAU:
pi = 3.141593
LINE(64,92)-(360,132),1,b
LOCATE 13,10:PRINT" E1 E2 E3 E4
E5 E6 E7 E8 Ex Ey";
LINE(376,92)-(576,108),1,b
LOCATE 13,55
PRINT"Programmende"
LINE(376,116)-(576,132),1,b
LOCATE 16,53:PRINT"Alle Motoren
aus"
LINE(64,182)-(316,214),1,b
LINE(64,142)-(316,174),1,b
LINE(328,182)-(576,214),1,b
LINE(328,142)-(576,174),1,b
REM *****
CIRCLE(356,198),20,1,1.25*pi,.9*pi,.5
CIRCLE(356,158),20,1,1.25*pi,.9*pi,.5
CIRCLE(92,198),20,1,1.25*pi,.9*pi,.5
CIRCLE(92,158),20,1,1.25*pi,.9*pi,.5
PATTERN 3
CIRCLE(404,198),20,5,1.25*pi,.9*pi,.5
CIRCLE(404,158),20,5,1.25*pi,.9*pi,.5
CIRCLE(140,198),20,5,1.25*pi,.9*pi,.5
CIRCLE(140,158),20,5,1.25*pi,.9*pi,.5
REM*****
CIRCLE(452,198),20,1,,,.5
CIRCLE(452,158),20,1,,,.5
CIRCLE(188,198),20,1,,,.5
CIRCLE(188,158),20,1,,,.5
PATTERN 3
CIRCLE(500,198),20,5,.1*pi,1.75*pi,.5
CIRCLE(500,158),20,5,.1*pi,1.75*pi,.5

```

```

CIRCLE(236,198),20,5,.1*pi,1.75*pi,.5
CIRCLE(236,158),20,5,.1*pi,1.75*pi,.5
REM*****
CIRCLE(548,198),20,1,.1*pi,1.75*pi,.5
CIRCLE(548,158),20,1,.1*pi,1.75*pi,.5
CIRCLE(284,198),20,1,.1*pi,1.75*pi,.5
CIRCLE(284,158),20,1,.1*pi,1.75*pi,.5
REM *****
OBJECT.X 1,181:OBJECT.Y 1,155
OBJECT.X 3,444:OBJECT.Y 3,155
OBJECT.X 2,181:OBJECT.Y 2,195
OBJECT.X 4,444:OBJECT.Y 4,195
FOR i=0 TO 1:PUT (69+i*48,154),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (69+i*48,194),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (333+i*48,154),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (333+i*48,194),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (250+i*48,154),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (250+i*48,194),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (514+i*48,154),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (514+i*48,194),
PF%,PSET:NEXT
OBJECT.ON
RETURN
SPRITES:
RESTORE ml
FOR i=0 TO 12:READ a%:
header$=header$+MKIS(a%):NEXT
FOR i=0 TO 8 :READ a%:

```



```

en$=en$+MKI$(a%):NEXT
FOR i=0 TO 6 :READ a%:
  mo1$=mo1$+MKI$(a%):NEXT
FOR i=0 TO 6 :READ a%:
  mo2$=mo2$+MKI$(a%):NEXT
FOR i=0 TO 6 :READ a%:
  mo3$=mo3$+MKI$(a%):NEXT
FOR i=0 TO 6 :READ a%:
  mo4$=mo4$+MKI$(a%):NEXT
RESTORE pfeil
FOR i=0 TO 6 :READ PF%(i):NEXT
m1$=header$+mo1$+en$
m2$=header$+mo2$+en$
m3$=header$+mo3$+en$
m4$=header$+mo4$+en$
OBJECT.SHAPE 1,m1$
OBJECT.SHAPE 3,m3$
OBJECT.SHAPE 2,m2$
OBJECT.SHAPE 4,m4$
RETURN
m1:
DATA 0,0,0,0,0,2,0,16,0,8,24,3,0
DATA 0,0,0,0,0,0,0,0,0
DATA -32232,-14792,-4584,-488,
-10728,-14824,-14722 :REM M1
DATA -32196,-14746,-4602,-484,
-10704,-14746,-14722 :REM M2
DATA -32196,-14746,-4602,-484,
-10746,-14746,-14788 :REM M3
DATA -32228,-14788,-4500,-308,
-10498,-14836,-14818 :REM M4
pfeil:
DATA 12,4,2
DATA 32640,16128,7680,3072
ENDE:

```

```

CALL Init
WINDOW CLOSE 2
SCREEN CLOSE 1
LIBRARY CLOSE
OBJECT.CLOSE

```

## Amiga BASIC

Die fischertechnik COMPUTING Programme der beigelegten Diskette sind meist in AmigaBASIC geschrieben. Die in den fischertechnik COMPUTING Programmieranleitungen dokumentierten Programme sind jedoch in dem BASIC eines anderen Computers formuliert. Die Unterschiede in den Programmlisten sind schon beträchtlich, insbesondere weil AmigaBASIC keine Zeilennummern als Strukturierungsmittel verwendet. Stattdessen stehen eine Reihe von leistungsfähigen Strukturbefehlen zur Verfügung, die in der vorliegenden Software auch optimal genutzt wurden. Das BASIC-typische GOTO wurde fast nirgends mehr verwendet.

Auch die Gestaltungsmöglichkeiten auf dem Bildschirm reichen weit über das hinaus, was in einem klassischen BASIC möglich ist. Hinzu kommt die Möglichkeit, auf die Routinen des Betriebssystems AmigaDOS und dessen Unterstützung von Bildschirmfenstern und der Maus zurückzugreifen.

Aus all diesen Gründen lehnt sich die Gestaltung der Amiga Software für die fischertechnik COMPUTING Baukästen nicht sehr eng an die Vorlage an, sondern schöpft aus diesen Möglichkeiten. Der eigentliche Ablauf der Steuerung der Modelle

hält sich jedoch immer noch an die ausgedruckte Vorlage. Wir stellen hier noch die wichtigsten unterschiedlichen Kommandos gegenüber:

fischertechnik COMPUTING Programmieranleitung	Amiga BASIC
<b>SYS M1,EIN</b>	<b>CALL M1(EIN)</b>
<b>PRINT USR(E1)</b>	<b>PRINT IN(E1)</b>
<b>SYS INIT</b>	<b>CALL INIT</b>
<b>PRINT CHR\$(147)</b>	<b>CLS</b>
<b>GET A\$</b>	<b>A\$=INKEY\$</b>

Cursorsteuerungen vereinfachen sich im Amiga-BASIC, da das Kommando LOCATE zur Verfügung steht. Die Ansteuerung von inversen Schriftzeilen kann nicht im PRINT-Kommando erfolgen, sondern erfordert jeweils den Aufruf des COLOR - Kommandos. Andererseits vereinfachen sich einige PRINT-Kommandos durch die Benutzung des PRINT-Kommandos mit Formatangabe (PRINT USING).

Die Überschrifttexte werden nicht gedruckt, sondern von Diskette (in diesem Fall von der Ram-Disk) geladen.

Studieren Sie die Programme auf der fischertechnik Diskette, um die Programmierung unter Amiga-BASIC zu verstehen.

## Anschluß von zwei Interface

Wenn für größere Projekte die Zahl der Interface-Eingänge oder Ausgänge nicht ausreicht, so können zwei Interface an einen Computer angeschlossen werden. Dabei wird ein Interface wie bisher an den Computer angeschlossen. Das zweite Interface wird an eine Steckverbindung des ersten Interface angeschlossen.

Bei den fischertechnik COMPUTING Interface wird als Steckverbinder ein 20-poliger Anschluß verwendet (zwei Reihen à 10 Stifte). An diesen Anschluß kann ein weiteres fischertechnik COMPUTING Interface, Artikel 30 566, direkt (ohne Steckadapter) angeschlossen werden.

Die Kombination zweier Interface erlaubt die Steuerung von bis zu acht Motoren und die Abfrage von 16 Digitaleingängen. Die Analogeingänge des zweiten Interface können nicht genutzt werden; nur die Analogeingänge des ersten Interface sind verfügbar.

Zur Steuerung beider Interface ist ein spezielles Treiberprogramm notwendig. Dieses ist per Gutschein bei den Fischerwerken abrufbar.

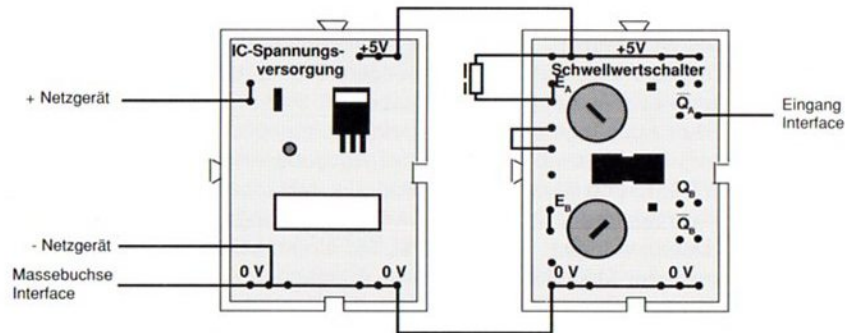
## Benutzung von fischertechnik Elektromechanik und Elektronik

Das fischertechnik COMPUTING Interface ist kompatibel zu den Bauteilen und Elektronikbausteinen der obengenannten Baukästen. Anstelle der bei den fischertechnik COMPUTING Modellen verwendeten mini-Taster können Sie genauso gut Taster und Schalter anderer Bauart anschließen. Z.B. den großen Taster oder den Polwendschalter aus dem fischertechnik **Elektromechanik** Baukasten, aber auch den Reedkontakt oder den Schaltkontakt eines Relais (aus der **ec** Serie). Aufpassen müssen Sie jedoch bei der Verwendung von selbstgebauten Tastern und Schaltern aus Gelenkbausteinen und Federn. Hier könnten eventuell Prellerscheinungen auftreten. Wir empfehlen, in diesen Fällen den Taster mehrmals abzufragen und den Wert nur dann als gültig zu erachten, wenn zweimal hintereinander der gleiche Wert erschien. Die Analogeingänge des Interface können mit Sensoren beschaltet werden, die einen Widerstandswert zwischen 0 und 5 k $\Omega$  als Ausgang liefern. Zunächst bieten sich die Potentiometer aus dem Baukasten fischertechnik COMPUTING an. Genauso können aber auch andere Bauelemente, wie z.B. der Fotowiderstand aus dem fischertechnik **Elektronik** Baukasten, verwendet werden. Die Motorausgänge des Interface sind kräftig

belastbar. Nicht nur die mini-Motoren, auch der S-Motor und der 6V-Motor lassen sich mit dem Interface ansteuern, wobei noch eine Lampe zur Funktionsanzeige parallelgeschaltet sein darf. Außer Motoren eignen sich noch der Elektromagnet und das Relais **RBII**.

Die Signale der Elektronikbausteine mit integriertem Schaltkreis aus der TTL-Familie (z.B. Schwellwertschalter aus dem fi-

schertechnik **Elektronik** Baukasten) können ebenfalls in die Eingänge des Interface eingespeist werden. Als gemeinsamer Bezugspunkt ist jedoch zuvor die Masseschiene des Elektronikbausteins mit der Massebuchse des Interface zu verbinden. In der Abbildung zeigen wir, wie eine Lichtschranke aufgebaut wird. Der Schwellwertschalter dient dazu, die Ansprechschwelle der Lichtschranke einzustellen.





## Funktionsweise des Interface und des Interface-Treiberprogramms

Wenn Sie die fischertechnik COMPUTING Software benutzen oder selbst Programme entsprechend der Hinweise in den vorigen Kapiteln erstellen, werden Sie kaum die nun folgende Information benötigen. Wenn Sie aber die Programme in anderen Sprachen als BASIC formulieren wollen, die Programme durch komplexe Abläufe in Maschinensprache beschleunigen wollen, die Funktionen des Interface erweitern wollen oder auch nur einfach einen Blick hinter die Kulissen werfen wollen, so wird Ihnen das Nachfolgende sicherlich hilfreich sein. Allerdings sollten Sie dann auch ein paar Kenntnisse der Maschinensprache und der Digitalelektronik mitbringen, denn hier geht es an die "bits and pieces".

Das fischertechnik Interface erfüllt eine Reihe von Aufgaben, die wir anhand des Blockdiagramms besprechen wollen. Am linken Rand sind die Signale von und zu dem Computer aufgeführt. Es fällt auf, daß diese recht wenig mit den Ausgängen M1 bis M4 und Eingängen E1 bis E8 sowie EX und EY gemein haben. Der Grund ist darin zu suchen, daß am Computeranschluß wesentlich weniger Datenleitungen zur Verfügung stehen, als auf der Modellseite des Interface benötigt werden. Diese wenigen Datenleitungen müssen deshalb so

eingesetzt werden, daß alle Signale auf der Modellseite gesteuert werden können. Das Konzept sieht eine Mehrfachverwendung der Datenleitungen mit Hilfe von Schieberegistern vor. Auf diese Weise werden z.B. nur drei Datenleitungen für die Steuerung der Ausgabe notwendig. Eine parallele Anschlußweise hätte acht Datenleitungen benötigt.

Schauen wir uns gleich die Ausgabe an den Anschlüssen M1 bis M4 genauer an. Die dafür benötigten Datenleitungen werden mit DATA-OUT, CLOCK und LOAD-OUT bezeichnet. Bei einer Ausgabe werden immer die Daten für alle vier Motoren übertragen, d.h. ein ganzes Byte (Ein Byte deswegen, weil jeder der vier Motoren zwei Bits zur Steuerung der Drehrichtung benötigt.). Die von dem Kommando nicht betroffenen Motorausgänge erhalten somit den derzeitigen Stand, der im Computer als Ausgabewort zwischengespeichert ist, erneut eingeschrieben.

Bei der Ausgabe werden der Reihe nach die Bits des Ausgabeworts an die Leitung DATA-OUT angelegt, das höchstwertige zuerst. Mit einem Übergang von low nach high am Ausgang CLOCK wird das Bit in ein Schieberegister übernommen. Danach folgt das nächste Bit an DATA-OUT, das

ebenfalls in das Schieberegister mit dem nächsten CLOCK-Impuls übernommen wird. Das vorangegangene Bit ist dabei aber auch um eine Position im Schieberegister nach rechts gerutscht, um dem nachfolgenden Platz zu machen. Nach insgesamt acht solchen Datenübertragungen ist das ganze Ausgabewort im Schieberegister abgelegt. Das zuerst übertragene Bit ist im Verlaufe des Datentransfers ganz nach rechts durch geschoben worden. Von der Aktivität im Schieberegister ist aber bislang an seinen Ausgängen noch nichts spürbar. Die Ausgangsverstärker werden nicht direkt über das Schieberegister angesteuert, sondern über ein zwischengeschaltetes Speicherregister, das auch noch im Schieberegister-Baustein integriert ist. Erst mit dem Übergang von low nach high am Ausgang DATA-OUT erfolgt die Übernahme in das Speicherregister. Die zeitliche Abfolge der Signale können Sie dem Impulsdiagramm entnehmen.

Ob die Daten allerdings auch die Leistungsverstärker durchsteuern, hängt wiederum von der Freigabesteuerung des Speicherbausteins ab. Die Freigabesteuerung erfolgt durch ein Monoflop. Diese Schaltung erzeugt ein Freigabesignal von einer halben Sekunde Dauer, wenn ein Impuls auf

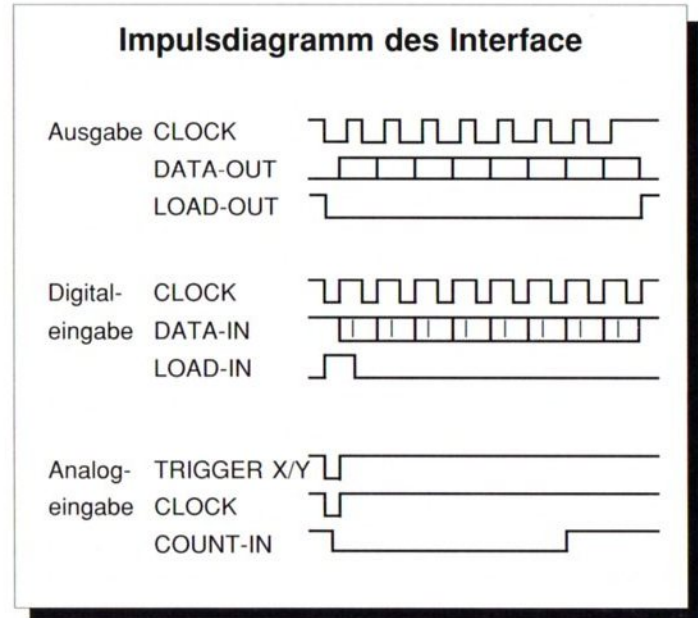
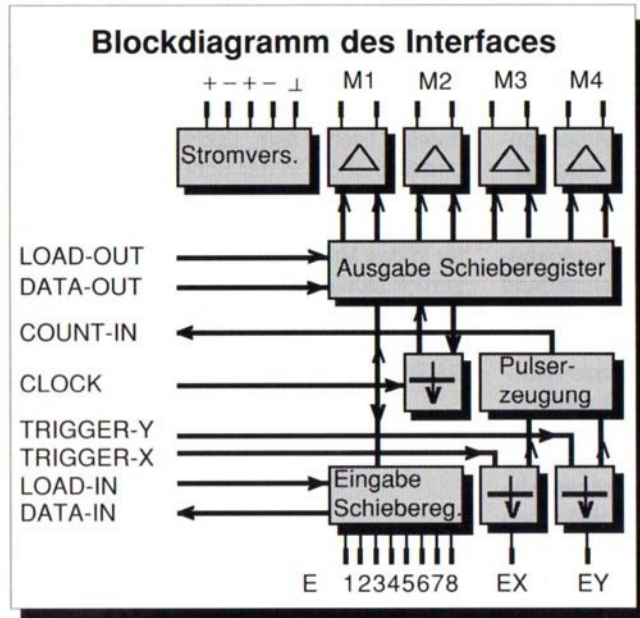


der CLOCK-Leitung vorliegt. Wir können davon ausgehen, daß zunächst die Leistungsverstärker angesteuert werden, da zuvor gerade die Daten mit Hilfe der CLOCK-Leitung übertragen wurden. Sollte aber innerhalb der nächsten halben Sekunde kein weiterer Datentransfer erfolgen, so kippt das Monoflop wieder in seinen stabili-

len Zustand zurück und das Freigabesignal wird zurückgenommen. Das Monoflop ist übrigens nachtriggerbar, d.h. die Zeitdauer von einer halben Sekunde rechnet sich jeweils vom Zeitpunkt des letzten CLOCK-Impulses an.

Nun zu der Übertragung der digitalen Signale an E1 bis E8. Im Prinzip findet bei der

Eingabe eine Umkehrung des oben Beschriebenen statt. Durch das Ausgabe-Signal LOAD-IN werden die an den Eingängen anstehenden Signale in das Eingabeschieberegister übernommen. Dies erfolgt wiederum für alle acht Eingänge, auch wenn nur ein einziger abgefragt werden soll. In dem Schieberegister angelangt,



bringt jeder Impuls auf der CLOCK-Leitung ein Bit auf der Eingabeleitung DATA-IN zum Vorschein, jenes von E8 zuerst und das von E1 zuletzt. Durch Testen dieser Leitung kann der Computer die Bits "auf-sammeln" und wieder ein Datenwort bilden. Das gewünschte Bit wird anschließend herausgefiltert und dem BASIC-Programm übergeben.

Da zur Übertragung der Daten dieselbe CLOCK-Leitung wie bei der Ausgabe benutzt wird, wird auch bei der digitalen Eingabe das Monoflop aktiviert, das das Freigabesignal für die Ausgabedaten steuert. Eine Fehlfunktion des Ausgabeschieberegisters durch die Mehrfachfunktion der CLOCK-Leitung steht nicht zu befürchten, denn die aktuellen Ausgabedaten stehen ja nicht im Ausgabeschieberegister sondern im Speicherregister. Ersteres wird zwar wohl durch die CLOCK-Impulse beeinflusst, nicht aber letzteres, das ja nur auf das Signal LOAD-OUT reagiert.

Bleiben zum Schluß die Analogeingänge EX und EY. Die Potentiometer oder sonstigen veränderlichen Widerstände dienen als zeitbestimmendes Bauelement in zwei weiteren Monoflop-Schaltungen. Ein niedriger Widerstandswert wird in einen Impuls kurzer Dauer, ein hoher Widerstandswert

in einen Impuls langer Dauer umgesetzt. Der Impuls selbst wird durch Startsignal TRIGGER-X bzw. TRIGGER-Y (mit negativer Logik) ausgelöst und erscheint dann auf der Leitung COUNT-IN. Ein Maschinenprogramm stellt die Impulsdauer anhand der Zahl der Schleifendurchläufe fest, die während der Impulsdauer durchgeführt werden können. Diese Zahl wird in das aufrufende BASIC-Programm zurückgegeben. Sie sehen also, daß der Analogwert weder etwas mit der Winkelstellung noch dem Widerstandswert der Potentiometer zu tun hat. Dagegen geht z.B. die Arbeitsgeschwindigkeit des Prozessors ein. Zwischen der letztlich ermittelten Zahl und dem Widerstandswert besteht jedoch ein linearer Zusammenhang. Dieser muß gegebenenfalls im BASIC-Programm noch anhand einer Eichung in Winkelgrade oder Widerstandswerte umgerechnet werden.

Wir wollen jetzt noch kurz zusammenfassen, wie das Interface mit dem Amiga verbunden wird. Es wird, wie Sie ja schon von dem Kapitel über den Anschluß des Interface wissen, die Parallel-Druckerschnittstelle verwendet. Von deren acht Datenleitungen werden die untersten sechs für die oben besprochenen Ausgabesignale verwendet, siehe auch die nachstehende Tabelle.

Ein kleines Problem gilt es für die Eingabeleitungen zu lösen. Die Parallel-Druckerschnittstelle beinhaltet nur eine für alle PC verbindliche Eingabeleitung, das Aktiv-Signal des Druckers, BUSY. Es entsteht jedoch kein Konflikt, wenn beide Eingabeleitungen, DATA-IN und COUNT-IN, mithilfe einer ODER-Schaltung zusammengefaßt werden. Da das Maschinenprogramm ja "weiß", welche Eingabefunktion es angefordert hat, kann es die Signale an dieser einzigen Eingabeleitung auch korrekt interpretieren. Ein Maschinenprogramm (Quelldateien fischer.asm, fischer.def, fischer.lib und fischer.src in der Schublade "sources" der Diskette) steuert die zuvor beschriebenen Abläufe. Zu der Übergabe der Parameter in das Maschinenprogramm und wieder zurück studieren Sie die Beschreibung der Einbindung von Maschinensprach-Unterprogrammen und des CALL-Befehls in der BASIC-Anleitung Ihres Amiga.



## Checkliste

### Anschluß des fischertechnik Interface an die Parallel-Druckerschnittstelle

Interface-Signal	Drucker-Signal	Stift
LOAD-OUT	Datenbit1	2
LOAD-IN	Datenbit2	3
DATA-OUT	Datenbit3	4
CLOCK	Datenbit4	5
TRIGGER-X	Datenbit5	6
TRIGGER-Y	Datenbit6	7
DATA/COUNT-IN	Busy	11

Sollte das fischertechnik COMPUTING Interface sich einmal widerwillig zeigen und nicht so arbeiten, wie Sie es erwarten, so überprüfen Sie bitte folgende Punkte mit dem Diagnoseprogramm:

- Das Diagnoseprogramm zeigt bei E1 bis E8 überall 1 an, obwohl kein Modell angeschlossen ist. - Das Interface ist nicht an den Computer oder nicht an das Netzgerät angeschlossen.
- Einer der Eingänge E1 bis E8 zeigt bei Betätigung des Tasters gerade das umgekehrte Ergebnis. - Öffner- und Schließfunktion des Tasters sind vertauscht.
- Einer der Eingänge E1 bis E8 zeigt immer das Ergebnis 0, obwohl er angeschlossen ist und betätigt wird. - Prüfen Sie die Verkabelung.
- Einer der Eingänge E1 bis E8 zeigt immer das Ergebnis 1, selbst wenn kein Modell angeschlossen ist. - Vermutlich das Eingangsgatter IC 4014 durch Überspannung (elektrostatische Aufladung) beschädigt.
- Ein Motorausgang arbeitet nicht. - Bitte Verkabelung überprüfen.
- Ein Motorausgang arbeitet nur in einer Richtung. - Leistungsstufe des Motors defekt.

- Motor läuft sehr langsam oder setzt aus. - Entweder Netzgerät durch zuviel Motoren überlastet (zwei Netzgeräte mot4 oder das stärkere COMPUTING Netzgerät verwenden) oder Netzgerät bei Verwendung des regelbaren Ausgangs nicht voll aufgedreht.
- Die Abfrage eines Analogkanals benötigt eine spürbare Zeit. - Es wurde ein unbeschalteter Analogeingang abgefragt.

Bei Defekten schicken Sie das Gerät bitte an folgende Adresse ein:

**fischerwerkeArtur Fischer GmbH&Co.KG**  
**Abteilung Service**  
**D-7244 Tumlingen / Waldachtal**

## Technische Daten

fischertechnik COMPUTING Interface zum Anschluß an Druckerschnittstellen nach Centronics-Norm, Art.Nr. 30566.

Das Interface kann unter anderem an den Computern Amiga 500 und Amiga 2000 verwendet werden. Der Betrieb am Amiga 1000 ist bei Adaption des Druckeranschlusses auf den 25-poligen Normanschluß möglich.

### **Ausgabe:**

4 Ausgänge zum Anschluß von Motoren, Lampen, Elektromagneten.. (M1 bis M4). Polarität des Ausgangs steuerbar.

Belastbarkeit: 1A Dauerstrom, 1,5A Spitzenstrom.

Schutzmaßnahmen: Bei einem Kurzschluß am Ausgang schaltet eine Kurzschlußsicherung den Ausgangsverstärker ab. Das Einschalten erfolgt automatisch beim Entfernen des Kurzschlusses. Bei Überlastung des Ausgangs steigt die Temperatur im Ausgangsverstärker. Bei Überschreitung der zulässigen Temperatur spricht eine Temperatursicherung im Ausgangsverstärker an.

Überwachungsschaltung des Daten-

stroms: Bei Ausbleiben von Datensignalen vom Computer schaltet das Interface nach 0,5 Sekunden alle Ausgänge inaktiv. Die Signale bleiben jedoch gespeichert.

### **Eingabe:**

8 Eingänge für digitale Signale (E1 bis E8). Durch interne Beschaltung sowohl Anschluß von elektromagnetischen Artikeln (Taster, Schalter, Relais) in positiver Logik als auch Anschluß von TTL-Ausgängen möglich.

2 Eingänge für analoge Signale (EX und EY).

Anschließbar sind Geber mit Widerstandswerten zwischen 0 und 5 k $\Omega$ , z.B. Potentiometer, Fotowiderstände...

### **Stromversorgung:**

Betriebsspannung: 6 bis 10 V ungesieberte Gleichspannung oder 8 V gesieberte Gleichspannung.

Die Verwendung des fischertechnik Netzgeräts COMPUTING wird empfohlen.

### **Anschlüsse:**

Interface-Anschlußkabel.  
Netzgerät-Anschluß (doppelt), 2 Buchsen, 2,5 mm  $\rightarrow$ . Massebuchse, 2,5 mm  $\rightarrow$ .  
Modellanschluß, 20-polig nach DIN 41651.  
Anschluß für weiteres Interface.

### **Abmessungen:**

150 x 90 x 30 mm,  
1200 mm Anschlußkabel,  
transparenter Deckel.



## Table of Contents

Important Notice	26
Introduction	27
Connecting the Interface	28
fischertechnik COMPUTING Software	28
The Diagnostic Program	36
Amiga-BASIC	40
Control of two Interfaces	40
Using the fischertechnik Electro- mechanics and Electronics Kits	41
Operation of the Interface and the Interface Driver Program	42
Check List	45
Technical Data	46
Wiring Diagram of the Interface Inputs and Outputs	47

## Important Notice

All measures and efforts have been taken to guarantee the correctness of this product documentation. However, since fischerwerke Artur Fischer GmbH & Co. KG constantly work on the improvement of their products, we cannot guarantee the completeness and correctness of this documentation from the time of printing onwards.

Amiga, Commodore 64, Commodore 128, Commodore VIC20 und CBM are trademarks of Commodore Electronics Limited.

Apple is a registered trademark of the Apple Computer Inc.

Atari is a registered trademark of the Atari Corporation.

Centronics is a registered trademark of the Data Computer Corporation.

CPC 464, CPC 664 und CPC 6128 are trademarks of the Amstrad Consumer Electronics plc.

CP/M is a trademark of the Digital Research Inc.

IBM and IBM PC are registered trademarks of the International Business Machines Corporation.

This equipment generates and uses radio frequency energy. If it is not properly installed and used in strict accordance with the manufacturer's instructions, this equipment may interfere with radio and television reception. This device has been tested and found to comply with the limits for a class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC rules, which are designed to provide reasonable protection against such interference in a residential installation. If you suspect interference, you can test this equipment by turning it off and on. If you determine that there is interference with radio or television reception, try one or more of the following measures to correct it:

- reorient the receiving antenna
- move the interface and the computer away from the receiver
- change the relative positions of the equipment and the receiver
- plug the power supply of the interface (as well as the computer equipment) in a different outlet so that the equipment and the receiver are on different branch circuits.

If necessary, consult your fischertechnik dealer or an experienced radio/television technician for additional suggestions. You may also wish to consult the following booklet, which was prepared by the Federal Communications Commission:

"How to Identify and Resolve Radio-TV Interference Problems"

This booklet is available from the U.S. Government Printing Office, Washington, D.C. 20402, Stock No. 004-000-00345-4.



## The fischertechnik COMPUTING Interface

Dear friend of fischertechnik:

fischertechnik COMPUTING has been developed to allow computer owners to control technical models with a computer, thereby extending the computer's capabilities. fischertechnik COMPUTING includes the fischertechnik COMPUTING construction sets (fischertechnik COMPUTING kit, Training Robot, Plotter/Scanner, EXPERIMENTAL), the fischertechnik COMPUTING interfaces and software. Computer owners can now simulate technical functions and processes, solve problems and have a lot of fun controlling models with the computer.

What is needed for controlling the models? First, the fischertechnik model which will perform the processes. Second, a personal computer such as the one you own to provide for control and coordination. Third, an interface which will link the two. The item in your hands is the fischertechnik COMPUTING interface. Control signals from the

computer, e.g. "Turn on motor!" are converted by the interface into currents strong enough to make a motor move. This is called "output". The point of view is from the computer outward.

The opposite direction is also possible and occurs in practice. The models are equipped with pushbutton switches, potentiometers, etc., in order to inform the computer about what is happening on the model. Here again, the interface provides assistance by modifying the signals in such a way that they represent an input which is meaningful to the computer.

The fischertechnik interface provides the following functions:

- With it you can control four fischertechnik motors, lamps, electromagnets etc.
- With it you can interrogate eight push-buttons or switches.
- Additionally, two inputs provide values from variable signal transmitters such as potentiometers.

What would be the use of all the electrical connections between the computer and the fischertechnik model provided by the interface without a means of activating them? We are referring to software, of course. The software is located on a diskette. The diskette contains a program which expands the language of your computer in such a way that it can control the interface. This program will be the core of your own programs. The diskette contains a lot more, though: in order for you to study and learn the use of these new resources, sample programs for all fischertechnik COMPUTING models are also included on it.

As you can see, a lot of challenging problems are waiting to be solved. Just try it out - you will have lots of fun!

Sincerely yours  
Artur and Klaus Fischer

## Connecting the Interface

The fischertechnik interface for Centronics Ports can be connected to a whole range of computers, among of which are also the Commodore Amiga 500 and Amiga 2000. It may also be connected to the Amiga 1000; due to the special printer port of this computer you have to consider the use of a special plug adaptor. We will discuss this later in this chapter. When editing this manual some more computers of Commodore's Amiga series are announced. We assume that they will be compatible to the Commodore Amiga computers listed above and that you will not run into difficulties to hook up the interface to those computers.

In the ongoing text we will refer to the Amiga 500, the Amiga 1000, the Amiga 2000 and future computers of the Amiga series using the short term Amiga.

The fischertechnik COMPUTING interface is connected to the parallel printer port of the Amiga. Use the following procedure:

- Make sure the computer is turned off.
- Turn the computer around so that you are facing the ports on the rear panel. Locate the parallel printer port. This port is jack connector which will accept 25 pins (except Amiga 1000, see later in

this text). You cannot mistake it for the other ports, for example, the serial port, because they have pins, not jacks.

- You may be electrostatically charged to several thousand volts by frictional electricity without noticing it and without it doing you any harm. However, these voltages are harmful to the circuits contained in the interface and the computer. It is therefore necessary to discharge any electrostatic charges by touching a grounded object, i.e. the metal housing of the computer.
- Place the interface in front of you.
- Together with this instruction manual and the diskette for the fischertechnik COMPUTING model kit you got the adapter connector fitting to the printer port of the Amiga. The adapter is inserted between the interface and the computer. The adapter carries two connectors. One plug fits to the connector of the interface cable. The other plug fits to the printer port. Insert the cable of the interface into the adapter connector. A reversed insertion is prohibited by the shape of the connector. Now plug the adapter onto the parallel printer port of your Amiga. Also in this case plugging in the wrong way is impossible due to the

trapezoidal shape of the connector.

- Connect the interface to the power supply unit. The interface expects an unfiltered direct current with a voltage between 6 and 10 volts. If a stabilized direct voltage source is used the power supply must be adjusted to approximately 8 volts (loadable to approximately 2 amps). Using the fischertechnik COMPUTING power supply unit will guarantee the correct voltage. Plug the output cable of the power supply unit (red flat plug) into the jack marked with a "+" on the interface, and the output cable with the green flat plug into the jack marked with a "-". Two pairs of jacks are provided; it does not matter which pair you use.
- For models with up to two motors, the fischertechnik power supply units mot 4 may also be used, i.e. especially for the models of the fischertechnik COMPUTING kit. If you intend to use your own constructions with a greater number of motors or the Training Robot or Plotter/Scanner kits, you will need a more powerful power source. In this case, you can either switch to the more powerful fischertechnik COMPUTING power supply unit or plug a second fischertechnik power supply unit mot4 into the



empty jacks.

- When using the mot4 power supply unit, you use the two-wire cable and flat plugs supplied with the kits to make the power cord.
- Connect the fischertechnik COMPUTING model to the interface. Use the twenty-wire ribbon cable supplied with the fischertechnik COMPUTING kits. This cable can also be ordered as a spare part.
- The turn-on sequence of the interface and the computer does not matter. If you are not using the interface while working with other programs, it is not necessary to unplug the interface. Just leave it turned off.
- Side effects caused by the interface: as long as the interface is plugged in, you cannot use a printer which is to be connected to the same port. However, there are switches available which allow you to connect several printers to a single computer. Such a switch may also be used with your printer and your interface, by this avoiding to remove and install the cables when changing from printer to interface and vice versa. The output port of those switches frequently uses the 36-pole version of the

Centronics connector. In order to connect the interface to one of the outputs of the switch, you should order the 36-pole version of the fischertechnik interface adaptor.

- The Amiga 1000 uses a printer port, which employs a 25-pole pin-connector. Most pins correspond to the jacks of the original connector. If you run into difficulties to order a gender changer connector for the Amiga 1000 at your computer dealer, you may try to construct one using the following procedure:
  - Use two 25-pole female connectors. Connect the lines 1 through 13 and 23 of one connector to the lines of same number of the other connector.
  - Lines 14 through 24 are ground lines. They may be connected to each other on both connectors. A single line will be sufficient to lead from one connector to the other.
  - Plugs 24 and 25 of both connectors must not be used.

Once more for your remembrance:  
Prior to working with the models you should also always discharge possible electrostatic charges by touching a grounded metal

object, e.g. the metal housing of the PC.  
Enough about hardware - in the next section we'll deal with the fischertechnik COMPUTING software.

## fischertechnik COMPUTING Software

Anyone who has toyed with the idea of controlling some sort of device or model with a computer may know from experience or may have heard from friends that this is not easiest of tasks. That it requires a thorough knowledge of the computer, the microprocessor, the input/output devices and machine language. These difficulties, which were considerable, were a good reason for many not to experiment with this fascinating field of computing - up to now, that is. fischertechnik COMPUTING provides the proper interface for your computer, however. And with the fischertechnik COMPUTING kits comes the software you need to make it all work smoothly and easily. Just tick the appropriate box on the coupon and mail it to the closest fischertechnik representative as given in the address list. On return of post there will be mailed the disk fitting to your construction kit and containing all the control programs. Additionally this manual details you how to write such control programs yourself.

In order to avoid losing the software through accidental deletion or damage to the diskette, you should make a copy of the fischertechnik COMPUTING diskette. For this purpose you use the program DISKCOPY contained on the AmigaDOS system diskette

(workbench). Proceed exactly according to the instructions of the AmigaDOS manual and the program prompts. At the end of the copy procedure you should use the rename function in order to delete the "copy of " in front of the disk's name. When you have made the copy, store the fischertechnik COMPUTING diskette in a safe place inaccessible to the natural enemies of diskettes such as sand, heat, cats or magnetic fields.

Work with the copy and only use the original fischertechnik COMPUTING diskette to make another copy if necessary. Additional copies are especially useful if you want to change the programs while retaining the contents of the original diskette. You are restricted to making copies for your own use.

Some fischertechnik programs will write data to a diskette. While you are at it, you should format some additional diskettes.

If you have not already connected the interface to your computer as described in the previous section, you should do so now. Turn the computer on while the system's workbench is inserted.

After the system has booted insert the fischertechnik COMPUTING diskette in one of the disk drives. If your computer is equipped

with only one disk drive, you have to remove the workbench. Open the disk window by double clicking the diskette icon. You should see in the window:

A drawer with the fischertechnik programs; it carries the name COMPUTING, ROBOT, PLOTTER, EXPERIMENTAL etc. and contains the sample programs of the fischertechnik COMPUTING models.

A drawer with the fischertechnik source texts which are programs for experienced computer users, realizing own ideas using the fischertechnik COMPUTING interface. Normally you need not to open this drawer. A book with the title "Read Me" containing additional information and last minute news concerning the fischertechnik COMPUTING software and models. You should browse through that book. Just give it a double click or use the open function in the menu bar.

A trashcan like the one of the workbench. To see the trashcan, you must enlarge the window. This is a safety measure avoiding that you erase files by putting them into the trashcan.

A living room with the caption "Install".

You should now convert the copy of your fischertechnik COMPUTING diskette into a fully operational fischertechnik COMPUTING workbench. For this purpose keep the sys-



tem's workbench and the system Extras diskette at your reach. Start the batch program with the "Install" icon. The program will prompt you for inserting the system workbench, the Extras diskette, or the fischertechnik COMPUTING diskette according the procedure of installing. If your computer is equipped with only one disk drive this will happen more frequently than if it is equipped with two disk drives. Proceed just according the instructions given by the program. The complete procedure of installation will take about ten minutes.

**Important notice:** The Extras diskette of your Amiga may not carry the name "Extras". For example, the Extras diskette in germany contains german textes and therefore is labelled "ExtrasD". So check for the name of your Extras diskette. If necessary use the text editor on the workbench and modify the batch program "install". Replace the name "Extras" by what is applicable in your case on every occurence. Quit the text editor saving the "install" file and try the installation once more.

After the installation procedure terminated successfully, on the (copy of) your fischertechnik COMPUTING disk will be assembled all programs you need for experimenting. This includes parts of the

system's workbench and the AmigaBASIC interpreter. Insert now the fischertechnik diskette in drive DF0: and boot the computer. This may be effected by pressing simultaneously the keys <CTRL><left Amiga><right Amiga>. When using the fischertechnik COMPUTING diskette next time you just insert it in drive DF0: before powering up your Amiga. Please note: when you intend to do experiments using fischertechnik COMPUTING you never should boot with another workbench and switch to the fischertechnik later on and call in programs. The startup sequence of the fischertechnik diskette copies some programs and data files to the RAM disk during booting. Those files would be missing, if the computer system would not have been started using the fischertechnik worbench.

Another note for users of international versions of the Amiga: During the startup sequence there is no special installation of an international keyboard driver. So the default mode is the US ASCII keyboard. In case you need a keyboard driver for your version of Amiga just have a look to the file startup-sequence of your system's workbench. At the very end of the file you will find a command like

**setmap x**

where x stands for the keyboard table (e.g. f=french). Copy the setmap command together with the appropriate keyboard table into the c directory of the fischertechnik diskette. Include the appropriate command in the fischertechnik startup-sequence using a text editor, e.g.:

**setmap f**

We recommend to insert the above command line close to the end of the file, just before the command

**loadWb**

After having booted the system with the fischertechnik workbench open once more the diskette window. Now there should appear the AmigaBASIC interpreter, too. Open the fischertechnik drawer (COMPUTING, ROBOT, PLOTTER, EXPERIMENTAL, etc.). Call in the program FISCHER by double clicking it. The program FISCHER will be loaded. Together with the program the AmigaBASIC interpreter is loaded as well. When the program starts it will give you a report on the hardware and software of your fischertechnik COMPUTING kit.

Next load the driver program DRIVER into the computer. For this purpose either type in the dialog window of AmigaBASIC

**RUN"DRIVER"**

or close the previous window and double

click the icon of "DRIVER".

The program "DRIVER" stands for the interface driver. It installs everything you need for data exchange with the interface. The driver program is also contained in all other sample programs on the disk. Just have a look to the driver in the list window of AmigaBASIC: One of the first lines contains the program text:

```
LIBRARY "ram:fischer.library"
```

or something similar.

This line is the key for the data transfer with the interface. It refers to a library with input and output commands located on the RAM disk. It was copied there during the startup procedure. The use of the RAM disk brings a lot of advantages: It is all the time available and features short access times.

The library "fischer.library" contains the above-mentioned detail knowledge about the input/output devices of the computer. By the use of the fischer library the control of the interface is simplified to a great extent. We would like to note that the fischer libraries on the different fischertechnik COMPUTING diskettes differ from each other. According the topic of the construction kit it contains specialized commands: robotic commands for the Training Robot, or data taking commands for the EXPERI-

MENTAL kit etc. Therefore you must not exchange the libraries.

What is described in this manual is a nucleus of commands as is available in every fischertechnik library. To test the library start the driver program. After completion of the program the Amiga will display its ready prompt again.

There is no apparent change on your computer. However, by linking the library your Amiga now has a number of new commands available which were not previously a part of BASIC in this form. These commands are tailored exactly to your Amiga and the fischertechnik COMPUTING interface. They will use the CALL command and the IN function. The driver program itself is written in BASIC and generates the parameters of the machine language program calls as BASIC variables.

The only BASIC commands you need to be able to use are the following ones:

The motor output M1 is controlled with the following commands:

<b>CALL M1(MON)</b> <b>(turn on)</b>	<b>CALL M1(MOFF)</b> <b>(turn off)</b>
<b>CALL M1(MCW)</b> <b>(clockwise rotation)</b>	<b>CALL M1(MCCW)</b> <b>(counterclockwise rotation)</b>

The command parameters designate the

motor, motor 1 in this case, and the mode of operation.

The same command is used with the parameters M2, M3 and M4 to control the other three outputs. Please note that turning a motor on (MON) is equivalent with clockwise rotation (MCW).

The 10 inputs of the interface are read with the aid of the IN function. The function

**IN(E1)**

is 1 if + 5 volts apply at input E1 of the interface. Otherwise IN(E1) will output 0. Similarly, IN(E2)...IN(E8) will display the status of the other digital inputs.

The analog inputs EX and EY are each connected to + 5V via a sensor, e.g. a potentiometer (4.7 kΩ). The functions

**IN(EX)**

**IN(EY)**

will then produce a value within the range between 20 and 220 according to the position of the potentiometer. The actual limits may be slightly different.

Analog inputs are mostly used for measuring positions. If, for example, a robot arm like the one in the fischertechnik COMPUTING robotics kit is driven by a motor, and potentiometer EX is adjusted synchronously with the motion of the arm, then the program, by repeatedly calling the function



## IN(EX)

can trace the movement of the robot exactly.

If no sensor is connected to inputs EX or EY, the effect of 100 k $\Omega$  resistors contained in the interface circuit will be perceptible. They are parallel to the external sensor and avoid "hanging" of the interface. The analog value will range between 4000 and 5000 in that case. We recommend, however, never to leave the analog inputs without connection. In case you don't use them, just make a short between the analog input and the reference voltage, +5 V; this corresponds to a resistance of 0  $\Omega$ .

The last of the new commands is

### CALL INIT

This command is used to switch the interface to a well-defined initial state. The command can also be used to turn off all motor channels simultaneously.

Enough introductory remarks, though. Connect a fischertechnik motor to M1 using the twenty-wire, color-coded ribbon cable. Use the yellow and the orange cables in the upper half of the ribbon cable.

Type:

### CALL M1(MCW)

The motor will briefly start up and then stop again. Congratulations! You've just learned

how to control the most complicated fischertechnik models with your Amiga. You have not been amongst the lucky ones? An error message appeared at your screen instead? Just start the driver program again. To our best knowledge there is a bug in the actual AmigaBASIC version. During the first start the interpreter does not install all links to the library. Entering commands in the direct mode will work only after several starts of the driver program. This bug, however, does not appear in the run mode of programs; so you have not to worry about it when using the sample programs or programs written by yourself.

Let's consider again the motor. It started and briefly later stopped again. Probably you would like to find out why the motor has stopped. Didn't you turn it on? Isn't there a separate command for turning it off, as described above? Well - the motor was stopped, but the value stored in the interface says that it is still running. The interface itself has "gone to sleep". This always happens when another command is not received within half a second and is programmed this way for safety reasons. Let's say you are testing a new program. The probability that there is a mistake hidden away somewhere in the program borders

on certainty. The program run is interrupted with a message such as

### Syntax error

### Ok

The motor, which was turned on a short time ago, however, continues running and is about to demolish that beautiful model you worked so hard to build. You would have to run to the power supply unit and turn the power off.

How good it is to know, then, that the motor will stop by itself. Even if you have interrupted programm execution by activating the stop function (open the menu bar using the right mouse button and select stop in the run menu). And if the programm is resumed, the first command "wakes up" the interface, and none of the motors has been forgotten. The procedure can go on as if nothing had ever happened.

There is a reason why the interface is turned off with a delay. Due to calculations, there will always be delays between input and output commands to the interface. These delays must be bridged. You can tell by looking at the light emitting diode (LED) on the interface whether the interface is being activated by input or output commands. The LED not only indicates voltage but also shows if the interface is operative.

Next we will take a brief look at the input

commands. Connect a pushbutton switch between E1 (the brown wire on the lower edge) and +5V (the red wire in the center of the ribbon cable). Type in the following command (and do not desparate if you obtain the above mentioned error message):

#### **PRINT IN(E1)**

Depending on whether or not the pushbutton switch between E1 and +5V was depressed when the return key on the computer was pressed, the screen displays a 1 or a 0.

Since the motor is still connected to the output from before, it will start up once more. Input commands will also reactivate the outputs of the interface! Connect a 4.7 kΩ potentiometer like the one in the fischertechnik COMPUTING robotics kit between EX and +5V. Adjust the slider to a medium position and type in

#### **PRINT IN(EX)**

The number which is displayed on the screen must be in the range between 20 and 220. You will regularly need the driver program which you were using just now. Every program written to control models via the fischertechnik COMPUTING interface begins with this routine which installs the new commands. The driver program is already

contained in all the sample programs on the diskette. All programs that you write yourself must start with the driver program and continue on subsequent lines.

In order to be able to observe the potentiometer more easily, we will now write the first fischertechnik COMPUTING program in this manner. The driver program is loaded into the computer. Move the mouse pointer into the list window, press the left mouse button, move the cursor to the end of the program and type in:

**read\_in:**

**PRINT IN (EX)**

**GOTO read\_in**

Then give the command

**RUN**

There may be a brief delay while the fischer.library and the title picture are being loaded. And then it will begin. All at once the screen will be filled with numbers which will continuously scroll up. If you then take the potentiometer in your hand and move the slider, you will be able to observe the change in the numbers. Turn the potentiometer from one stop to the other. The numbers displayed should be between 20 and 220. Activate the stop function to interrupt the program.

For those who are interested in more detailed

information about these processes and who do not merely wish to use the programs included on the diskette, we will provide some detailed information here. The function of the driver program is to install the link to the fischertechnik library. The library is contained on the disk as fischer.library and fischer.bmap in the directory fischer/libs. On booting the computer it will be copied to the RAM disk according the commands contained in the file startup-sequence. The parameters INIT, M1, M2, M3, M4, and IN (and eventually some more names) are jump addresses to the library. Besides the library itself, the standardized parameters E1, E2, E3, E4, E5, E6, E7, E8, EX, EY, MCW, MCCW, MON and MOFF are set. Also all variables are defined as integer variables (DEFLNG).

The following limitations therefore apply when you write your own BASIC programs: Just as reserved BASIC key words such as PRINT or STOP cannot be used as variables, the use of the above-mentioned parameters as variables is prohibited. However, the limitation does not apply to variables of another type (strings, real), so that

**MOFF\$, M1!, E1!...**

may be used.



Those interested in further details, e.g. in the electronic processes in the interface or adaptations to other programming languages, should refer to the description of the interface operation and the source code.

The last function of the driver program is to turn on the interface and turn off all connected loads. This is caused by issuing the following command:

### CALL INIT

This switches the interface to the ready state for a subsequent user program. Additionally we have supplied the routine for loading the headline picture, which you may like to adopt in your own programs. All sample programs on the diskette are built according to this pattern. We suggest you study them for hints and ideas to use in designing your own programs.

```

REM *****
REM *** Interface-Driver ***
REM *** for ***
REM *** Commodore AMIGA ***
REM *****
REM
DECLARE FUNCTION In& LIBRARY
LIBRARY "ram:libs/fischer.library"
DEFINING a-z
Ex=160
Ey=144
REM
EA=255
REM Using IN(EA) all digital input
REM lines may read simultaneously.
E8=128
E7=64
E6=32
E5=16
E4=8
E3=4
E2=2
E1=1
Mccw=85
Mccw=170
Moff=255
Mon=85
REM Control of the motors is effected
    by
REM CALL M1(Direction) or CALL
    MOTOR(1..4,direction)
CALL Init
REM *****
REM *** Screen display ***

```

```

REM *****
SCREEN 1,640,256,2,2
WINDOW2," DRIVER
    PROGRAM",,0,1
PALETTE 0,208/256,208/256,208/256
    'light grey
PALETTE 1,0,0,0 'black
PALETTE 2,0,128/256,224/256 'blue
PALETTE 3,240/256,0,0 'red
REM
REM Read in title as BOBs
REM
OPEN "TITLE" FOR INPUT AS 1
OBJECT.SHAPE 1,INPUT$(6506,1)
OBJECT.SHAPE 2,INPUT$(1790,1)
OBJECT.SHAPE 3,INPUT$(794,1)
CLOSE 1
OBJECT.X 1,34
OBJECT.Y 1,10
OBJECT.X 2,145
OBJECT.Y 2,48
OBJECT.X 3,129
OBJECT.Y 3,233
OBJECT.ON
OBJECT.CLOSE

```

## The Diagnostic Program

If you have constructed a fischertechnik computing model, you may find that things do not work as you had expected. No wonder considering the great number of lines running between the model and the interface. If only one pushbutton switch were incorrectly wired, this could produce the most astounding results. Matters become twice as complicated when you write your own programs. Where do you start looking for errors? Is it a hardware problem? Is the software to blame?

In order to help you test the hardware clearly and conveniently, we have developed the diagnostic program. It is included on every fischertechnik diskette as the DIAGNOST file. Always load this program for testing a model. It allows you to monitor all the inputs and to verify whether their behavior corresponds to your expectations. The motor control outputs are selected using the mouse. A box is assigned to each motor. In the respective box the symbol for motor stop is in the middle. The symbols on the right cause operation in clockwise mode, the symbols on the left cause operation in counter-clockwise mode. Additionally you may choose between two running conditions. The dotted symbol controls motor operation only as long as the mouse button

is pressed. This option may be used for fine positioning. The solid symbol controls continuous running. You may select it if e.g. a magnet should be switched on and stay on, even when you shift to another motor for operating a robot axis.

To switch off all motors simultaneously there exists a big emergency-off-box. With a quit box you may select the end of the program.

```
REM *****
REM *** Interface-driver ***
REM *** for ***
REM *** Commodore AMIGA ***
REM *****
REM
DECLARE FUNCTION IN& LIBRARY
LIBRARY "fischer.library"
DEFLNG a-z
Ex=160
Ey=144
EA=255
E8=128
E7=64
E6=32
E5=16
E4=8
E3=4
E2=2
E1=1
Mcw=85
Mccw=170
Moff=255
Mon=85
CALL Init
REM
REM *****
REM *** screen display ***
REM *****
SCREEN 1,640,256,3,2
WINDOW 2,"
GNOSTIC",,0,1
PALETTE 0,208/256,208/256,208/256
PALETTE 1,0,0,0
PALETTE 2,0,128/256,224/256
```



```

PALETTE 3,240/256,0,0
PALETTE 6,48/256,48/256,48/256
PALETTE 7,80/256,80/256,80/256
PALETTE 4,128/256,128/256,128/256
PALETTE 5,160/256,160/256,160/256
OPEN "TITLE" FOR INPUT AS 1
OBJECT.SHAPE 1,INPUT$(6506,1)
OBJECT.SHAPE 2,INPUT$(1790,1)
OBJECT.SHAPE 3,INPUT$(794,1)
CLOSE 1
OBJECT.X 1,34
OBJECT.Y 1,10
OBJECT.X 2,145
OBJECT.Y 2,48
OBJECT.X 3,129
OBJECT.Y 3,233
OBJECT.ON
OBJECT.CLOSE
REM*****
REM* *
REM* fischertechnik computing *
REM* *
REM* Diagnostic *
REM* *
REM*Copyright(c)fischerwerke 1987*
REM* *
REM*****
REM
REM Function:
REM
REM Using this program the correct
REM functioning of the model may be
REM tested.
REM All input lines are displayed.
REM All output lines are controlled

```

```

REM using the mouse.
REM
REM *****
REM *** Initialize Control ***
REM *****
REM
ON BREAK GOSUB END
BREAK ON
GOSUB SPRITES
GOSUB SCREEN_DISPLAY
REM all motors off
REM stat of 4 motors
DIM Sta(4),X(4)
FOR i=1 TO 4
    LET Sta(i)=Moff
NEXT i
REM Start with motor 1
LET MOT=1
REM
REM *****
REM *** Display input lines ***
REM *****
REM
MAIN_LOOP:
    LOCATE 15,10:PRINT IN(E1)
    LOCATE 15,13:PRINT IN(E2)
    LOCATE 15,16:PRINT IN(E3)
    LOCATE 15,19:PRINT IN(E4)
    LOCATE 15,22:PRINT IN(E5)
    LOCATE 15,25:PRINT IN(E6)
    LOCATE 15,28:PRINT IN(E7)
    LOCATE 15,31:PRINT IN(E8)
    LOCATE 15,36:PRINT USING "####";
        IN(Ex)
    LOCATE 15,41:PRINT USING "####";

```

```

IN(Ey)
REM *****
REM *** Control of output lines ***
REM *****
REM
REM Mouse control
k=MOUSE(0)
X=MOUSE(1)
Y=MOUSE(2)
REM Selected motor
ZU=Sta(MOT)
IF k THEN
    IF Y>140 THEN
        IF Y>180 AND X<316 THEN
            MOT=2
        END IF
        IF Y>180 AND X>328 THEN
            LET MOT=4
        END IF
        IF Y<172 AND X<316 THEN
            LET MOT=1
        END IF
        IF Y<172 AND X>328 THEN
            LET MOT=3
        END IF
    END IF
    IF ABS(X-188)<=20 OR ABS(X-452)
    <=20 THEN
        LET Sta(MOT)=Moff
        LET X(MOT)=0
        LET ZU=Moff
    END IF
    IF ABS(X-92)<=20 OR ABS(X-356)
    <=20 THEN
        LET Sta(MOT)=Mccw
        LET X(MOT)=96
    END IF

```

```

LET ZU=Mccw
END IF
IF ABS(X-288)<=20 OR ABS(X-548)
<=20 THEN
LET Sta(MOT)=Mcw
LET X(MOT)=-96
LET ZU=Mcw
END IF
IF ABS(X-140)<=20 OR ABS(X-404)
<=20 THEN
LET ZU=Mccw
END IF
IF ABS(X-236)<=20 OR ABS(X-500)
<=20 THEN
LET ZU=Mcw
END IF
ELSE
IF X>376 THEN
IF ABS(Y-124)<=16 THEN
CALL Init
FOR i=1 TO 4
LET Sta(i)=Moff
LET X(i)=0
NEXT i
LET ZU=Moff
END IF
IF ABS(Y-100)<=16 THEN END
END IF
END IF
END IF
REM Output control of selected motor
CALL MOTOR(MOT,ZU)
OBJECT.X 1,181-X(1)
OBJECT.X 2,181-X(2)
OBJECT.X 3,444-X(3)

```

```

OBJECT.X 4,444-X(4)
GOTO MAIN_LOOP
SCREEN_DISPLAY:
pi = 3.141593
LINE(64,92)-(360,132),1,b
LOCATE 13,10:PRINT" E1 E2 E3 E4
E5 E6 E7 E8 Ex Ey";
LINE(376,92)-(576,108),1,b
LOCATE 13,55
PRINT"End of program"
LINE(376,116)-(576,132),1,b
LOCATE 16,53:PRINT"Emergency
Stop"
LINE(64,182)-(316,214),1,b
LINE(64,142)-(316,174),1,b
LINE(328,182)-(576,214),1,b
LINE(328,142)-(576,174),1,b
REM *****
CIRCLE(356,198),20,1,1.25*pi,.9*pi,.5
CIRCLE(356,158),20,1,1.25*pi,.9*pi,.5
CIRCLE(92,198),20,1,1.25*pi,.9*pi,.5
CIRCLE(92,158),20,1,1.25*pi,.9*pi,.5
PATTERN 3
CIRCLE(404,198),20,5,1.25*pi,.9*pi,.5
CIRCLE(404,158),20,5,1.25*pi,.9*pi,.5
CIRCLE(140,198),20,5,1.25*pi,.9*pi,.5
CIRCLE(140,158),20,5,1.25*pi,.9*pi,.5
REM*****
CIRCLE(452,198),20,1,,.5
CIRCLE(452,158),20,1,,.5
CIRCLE(188,198),20,1,,.5
CIRCLE(188,158),20,1,,.5
PATTERN 3
CIRCLE(500,198),20,5,.1*pi,1.75*pi,.5
CIRCLE(500,158),20,5,.1*pi,1.75*pi,.5

```

```

CIRCLE(236,198),20,5,.1*pi,1.75*pi,.5
CIRCLE(236,158),20,5,.1*pi,1.75*pi,.5
REM*****
CIRCLE(548,198),20,1,.1*pi,1.75*pi,.5
CIRCLE(548,158),20,1,.1*pi,1.75*pi,.5
CIRCLE(284,198),20,1,.1*pi,1.75*pi,.5
CIRCLE(284,158),20,1,.1*pi,1.75*pi,.5
REM *****
OBJECT.X 1,181:OBJECT.Y 1,155
OBJECT.X 3,444:OBJECT.Y 3,155
OBJECT.X 2,181:OBJECT.Y 2,195
OBJECT.X 4,444:OBJECT.Y 4,195
FOR i=0 TO 1:PUT (69+i*48,154),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (69+i*48,194),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (333+i*48,154),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (333+i*48,194),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (250+i*48,154),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (250+i*48,194),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (514+i*48,154),
PF%,PSET:NEXT
FOR i=0 TO 1:PUT (514+i*48,194),
PF%,PSET:NEXT
OBJECT.ON
RETURN
SPRITES:
RESTORE m1
FOR i=0 TO 12:READ a%:
header$=header$+MKIS(a%):NEXT
FOR i=0 TO 8 :READ a%:

```



```

en$=en$+MKI$(a%):NEXT
FOR i=0 TO 6 :READ a%:
  mo1$=mo1$+MKI$(a%):NEXT
FOR i=0 TO 6 :READ a%:
  mo2$=mo2$+MKI$(a%):NEXT
FOR i=0 TO 6 :READ a%:
  mo3$=mo3$+MKI$(a%):NEXT
FOR i=0 TO 6 :READ a%:
  mo4$=mo4$+MKI$(a%):NEXT
RESTORE pointer
FOR i=0 TO 6 :READ PF$(i):NEXT
m1$=header$+mo1$+en$
m2$=header$+mo2$+en$
m3$=header$+mo3$+en$
m4$=header$+mo4$+en$
OBJECT.SHAPE 1,m1$
OBJECT.SHAPE 3,m3$
OBJECT.SHAPE 2,m2$
OBJECT.SHAPE 4,m4$
RETURN
m1:
DATA 0,0,0,0,0,2,0,16,0,8,24,3,0
DATA 0,0,0,0,0,0,0,0
DATA -32232,-14792,-4584,-488,
-10728,-14824,-14722 :REM M1
DATA -32196,-14746,-4602,-484,
-10704,-14746,-14722 :REM M2
DATA -32196,-14746,-4602,-484,
-10746,-14746,-14788 :REM M3
DATA -32228,-14788,-4500,-308,
-10498,-14836,-14818 :REM M4
pointer:
DATA 12,4,2
DATA 32640,16128,7680,3072
END:

```

```

CALL Init
WINDOW CLOSE 2
SCREEN CLOSE 1
LIBRARY CLOSE
OBJECT.CLOSE
END

```

## Amiga BASIC

The fischertechnik COMPUTING programs included on the supplied diskette are mostly written in Amiga BASIC. The programs documented in the fischertechnik COMPUTING programming instructions, however, have been written using the BASIC language of another computer. The difference in the program listings are considerable, especially because Amiga BASIC does not use line numbers as a mean for structuring the programs. Instead a number of efficient structure commands are available, which have been used optimally in the present software. The BASIC-typical GOTO has been avoided in almost every case.

Also the possibilities to manipulate the screen display are much wider than in a "classical" BASIC. In addition it is possible to refer to the routines of the operating system AmigaDOS with its support of mouse and windows.

For those reasons the conception of the Amiga software of the fischertechnik COMPUTING kits is not modelled on the original, but takes advantages of the above-mentioned possibilities. The proper sequence of controlling the models is still following the printed programs, however. Here again we confront the most important differing commands:

fischertechnik  
COMPUTING  
Programming  
Instructions

Amiga  
BASIC

**SYS M1, CW**  
**PRINT USR(E1)**  
**SYS INIT**  
**PRINT CHRS (147)**  
**GET A\$**

**CALL M1(MCW)**  
**PRINT IN(E1)**  
**CALL INIT**  
**CLS**  
**A\$=INKEY\$**

Cursor control is simplified in Amiga BASIC, since the LOCATE command is available. Inverse text lines cannot be created using the PRINT command, but require the use of the COLOR command in each case. On the other hand, some PRINT commands are simplified by the use of the PRINT command with the format specification (PRINT USING).

The headlines of the screen displays are not printed, but loaded from disk (the ram disk in this case) as a picture. Study the programs on the fischertechnik COMPUTING diskette to understand the programming in Amiga BASIC.

## Control of two Interfaces

If in the course of great projects the number of Interface inputs or outputs becomes insufficient you may connect two interfaces to one single computer. The procedure is as follows: the first interface is connected to the computer as usual. The second interface is connected to a pin connector in the first interface.

The connector of the fischertechnik COMPUTING interface is a 20 pole connector consisting of two rows of ten pins each. You may plug to this connector another fischertechnik COMPUTING interface (order number 30 566) directly without using any adapter.

The combination of two interface allows the control of as much as eight motors und 16 digital inputs. The analog channels of the second interface may not be used, however; there remain the two analog channels of the first interface.

The control of both interfaces requires a special driver program. Ask your fischertechnik representative for more information.



## Using the fischertechnik Electromechanics and Electronic Kits

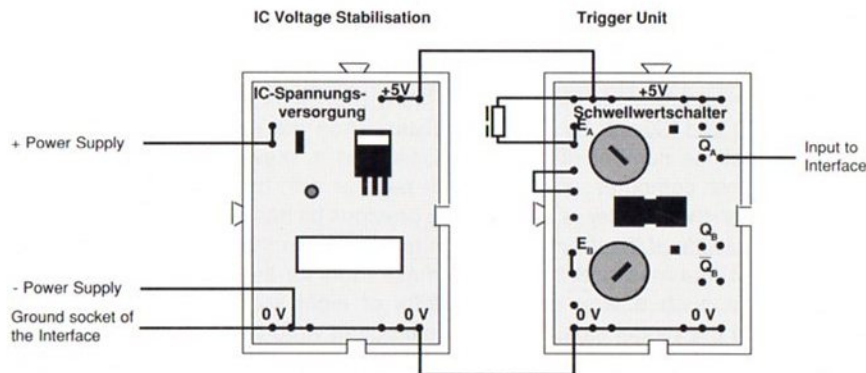
The fischertechnik computing interface is compatible with the components and electronic devices of the above-mentioned kits. Other types of pushbutton switches and switches may be used in place of the mini pushbuttons used with the fischertechnik computing models, e.g. the large pushbutton or the reversing switch contained in the fischertechnik **Electromechanics** kit, but also the reed contact or the switching contact of a relay (e.g. from the **ec** kits). Be careful when using pushbuttons and switches which you have constructed yourself from articulated elements and springs - they could cause contact bounce. In such cases, we recommend having the program check the input several times and consider the result valid only if the same value appears twice in a row.

The analog inputs of the interface can be connected to sensors with an output resistance between 0 and 5 k $\Omega$ . The potentiometers from the fischertechnik computing kit may be used of course. Other devices, such as the photoresistor from the fischertechnik **Electronics** kit, may be used as well.

The motor outputs of the interface feature high load capacity. Not only the mini mo-

tors, but also the S-motor and the 6V-motor can be controlled via the interface, whereby one lamp may be connected in parallel as a function indicator. Besides motors, the electromagnet and the relay **RBII** are also suitable. Signals from the electronic modules with integrated circuits of the TTL family (e.g. trigger unit from the fischertechnik

**Electronics** kit) may also be fed to the interface inputs. However, the common point of reference must be established by connecting the ground circuit of the electronic module to the ground jack of the interface. The figure shows the design of a light barrier. The trigger unit is used to adjust the reaction threshold of the light barrier.



## Operation of the Interface and the Interface Driver Program

If you use the fischertechnik computing software or write programs yourself according to the notes in the previous sections, most likely you will not need the information that follows. If, however, you intend to write the programs in a language other than BASIC, would like to speed them up through complex procedures in machine language, wish to extend the functions of the interface or simply want to glimpse behind the scenes, then the following information will most certainly be helpful. In this case, however, you should have a basic knowledge of machine language and digital electronics, since this is about the "bits and pieces".

The fischertechnik interface handles a number of tasks which we would like to discuss with the aid of the block diagram. On the left side you see the signals from and to the computer. Note how little they have in common with outputs M1 to M4 and inputs E1 through E8 and EX and EY. The reason for this is that the number of data lines available at the computer port is significantly lower than the number of lines required on the model side of the interface. This limited number of data lines must therefore be employed in such a way as to control all signals on the model side. The concept employed is that of multiple use of

the data lines with the aid of shift registers. In this way, for example, only three data lines are required for controlling the output. A parallel connection scheme would have required eight data lines. Let's take a closer look at the output at connections M1 to M4. The data lines required are designated DATA OUT, CLOCK and LOAD OUT. If there is an output, the data for all four motors are transmitted in each case, i.e. a whole byte (a byte because each of the four motors requires two bits for controlling the direction of rotation). The motor outputs to which the signal does not apply are thus once again supplied with the current state which is buffered in the computer as an output word. For output, the bits of the output word are sequentially (with the most significant bit first) fed to the DATA OUT line. When the signal at the CLOCK output goes from low to high, the bit is transferred to a shift register. Then the next bit at DATA OUT follows, and is likewise transferred to the shift register with the next CLOCK pulse. The previous bit has been shifted one position to the right in the shift register in order to make room for the subsequent bit. After a total of eight such data transfers, the whole output word has been transferred of the shift register. The bit first transferred

has been shifted all the way to the right in the course of the data transfer. Thus far, the activity in the shift register has not had any effect on its outputs. The output amplifiers are not controlled directly by the shift register, but rather via an storage register which is integrated in the shift register chip. Only when the LOAD OUT output goes from low to high are the data transferred to the storage register. The timing of the signals is shown in the pulse diagram.

Whether the data are fed to the power amplifiers, however, depends on the enabling control of the memory module. The enabling circuit is controlled by a monostable multivibrator. This circuit generates an enabling signal with a duration of half a second if there is a pulse on the CLOCK line. We may assume that the power amplifiers receive a signal first since the data were just transferred with the aid of the CLOCK line. If no more data are transmitted within the next half second, however, the monostable multivibrator will flip back to the stable state and the enabling signal is removed. The monostable multivibrator, by the way, can be retriggered, i.e. the time of half a second is always calculated from the time of the last CLOCK pulse.

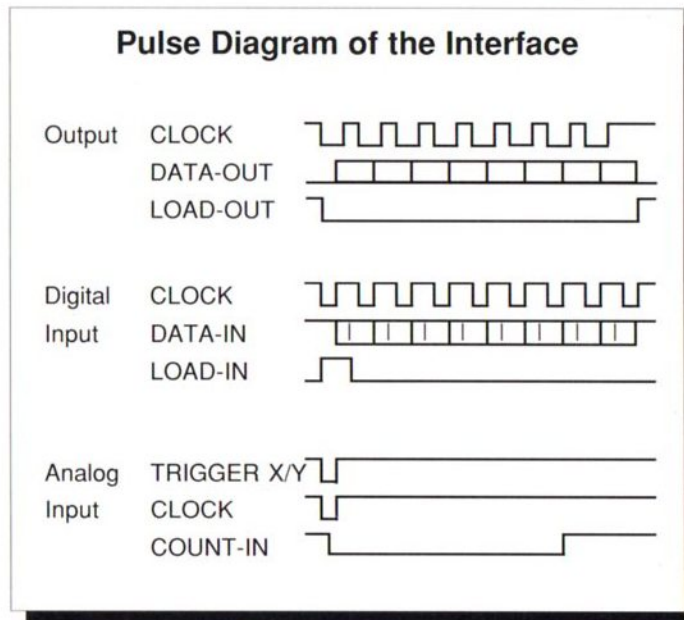
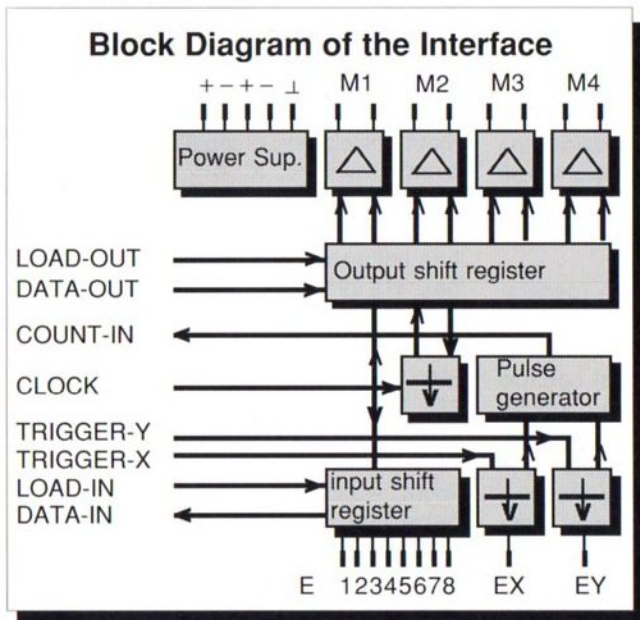
We will proceed with the transfer of the



digital signals to inputs E1 through E8. Basically the input is a reversal of the output process described above. The output signal LOAD IN causes the transfer of the data applying at the inputs to the input shift register. This always involves all eight inputs, even though only one of them is to be interrogated. When applying to the shift

register, each pulse of the CLOCK line will cause the transfer of one bit on the input line DATA IN, the bit from E8 first and the one from E1 last. By testing this line, the computer can "collect" the bits and reassemble them into a data word. The desired bit is subsequently filtered out and transferred to the BASIC program.

Since the same CLOCK line is used for data transmission as for output, the digital input will also activate the monostable multivibrator, which controls the enabling signal for the output data. Malfunctioning of the output shift register caused by the multiple function of the CLOCK line is not to be expected since the current output data are



not contained in the output shift register, but in the storage register. The former is controlled by the CLOCK pulses, unlike the latter, which only reacts to the LOAD OUT signal.

That leaves the analog inputs EX and EY. The potentiometers or other variable resistors are used as the timing element in two additional monostable multivibrator circuits. A low resistance value is converted to a short pulse, a high resistance value to a pulse with a long duration. The pulse itself is triggered by the starting signals TRIGGER-X and TRIGGER-Y (with negative logic), respectively, and then appears on the COUNT IN line. A machine language program determines the pulse duration by means of the number of loops which can be executed during the duration of the pulse. This number is fed back to the BASIC program which calls this function. You can see that there is no direct relationship between the analog value and the resistance. The clock rate of the microprocessor, however, is involved. Nevertheless there is a linear relationship between the number determined in the end and the resistance. If required, this value must be converted into angular degrees or resistance values by means of calibration. At this point, we will briefly review the con-

nection between the interface and the Amiga. As you know from the section about connecting the interface, the computer's parallel printer port is used for this purpose. Out of the eight data lines of this interface, the lower six are used for the output signals discussed above (Also refer to the following table.).

The input lines pose a slight problem in that the parallel printer port has only one input line available applying to all PC's. This is the active signal from the printer, BUSY. There is no conflict, however, if all input lines are combined using an OR circuit. Since the machine language program "knows" which input function it has requested, it has the capability of interpreting the signals on this one input line correctly.

The machine language program (generated from the source files fischer.asm, fischer.def, fischer.lib, and fischer.src in the drawer "sources" of the diskette) controls the processes described above. Concerning the linking of machine language routines and transfer of the values of the parameters between BASIC and the machine language routines refer to the chapter of the CALL-command in the BASIC-manual of your Amiga.

## Connecting the fischertechnik Interface to the Parallel Printer Port

Interface Signal Printer Signal Pin Number

<b>LOAD OUT</b>	<b>Data bit 1</b>	<b>2</b>
<b>LOAD IN</b>	<b>Data bit 2</b>	<b>3</b>
<b>DATA OUT</b>	<b>Data bit 3</b>	<b>4</b>
<b>CLOCK</b>	<b>Data bit 4</b>	<b>5</b>
<b>TRIGGER-X</b>	<b>Data bit 5</b>	<b>6</b>
<b>TRIGGER-Y</b>	<b>Data bit 6</b>	<b>7</b>
<b>DATA/COUNT IN</b>	<b>Busy</b>	<b>11</b>



## Check List

Should you ever have a problem with the fischertechnik computing interface, and it does not work as you expect it to, then check the following points with the diagnostic program:

- The diagnostic program displays 1 for E1 through E8, although no model is connected. – The interface has not been connected to the computer or to the power supply unit.
- One of the inputs E1 through E8 shows the reverse effect from what you would expect when the pushbutton switch is actuated. – Opening and closing functions of the pushbutton switch have been reversed.
- One of the inputs E1 through E8 always shows 0, although it is connected and has been actuated. – Check for a broken or improperly connected wire.
- One of the inputs E1 through E8 always shows 1, even though no model is connected. – The input gate IC 4014 has probably been damaged by overvoltage (electrostatic discharge).
- A motor output does not work. – Check for a broken or improperly connected wire.
- A motor output only works in one direction of rotation. – The power stage of the

motor output is defective.

- The motor rotates slowly or works intermittently. – Either the power supply unit is overloaded by too many motors (use two mot4 power supply units or the more powerful COMPUTING power supply unit) or the power supply unit has not been turned up all the way when using the adjustable output.
- The interrogation of analog input channel requires a perceptible amount of time. – An open analog channel (without any sensor connected to it) has been called in.

In case of defect, send the unit to your dealer or the company representing and importing fischertechnik.

## Technical Data

fischertechnik COMPUTING interface for use with printer ports according the Centronics standard specifications, Part No. 30566. The interface may used, amongst a wide range of other computers, with Commodore's Amiga 500 and Amiga 2000.

### Output

Four outputs for connection of motors, lamps, electromagnets, etc. (M1 to M4).

Polarity of output (direction of rotation) controllable.

Load capacity: 1 A continuous, 1.5 A peak.  
Protection: On a short circuit of the output, an electronic fuse will shut down the output amplifier. The output amplifier will be powered up again on removal of the short circuit. On overloading the output the temperature in the output amplifier will rise. When exceeding the permissible temperature an electronic fuse will shut down the output amplifier. The output amplifier will be powered again, when the temperature has decayed.

Circuitry for monitoring the stream of data:

If no data signals are received from the Amiga Computer, the interface will disable all outputs after 0,5 seconds. The signals are stored in memory.

### Input:

Eight inputs for digital signals (E1 through E8). Due to internal wiring, the connection of electromagnetic devices (pushbutton switches, relay contacts) in positive logic as well as the connection of TTL outputs is possible.

Two inputs for analog signals (EX and EY). Sensors with resistances between 0 and 5 k $\Omega$ , e.g. potentiometers, photoresistors, etc. can be connected.

### Power supply:

Operating voltage: 6 to 10 V unfiltered dc

voltage or 8 V stabilized dc voltage.

The use of the fischertechnik COMPUTING power supply is recommended.

### Connectors:

Interface cable fitting to printer port adaptor connector.

Power supply (twice), two jacks, 2.5 mm (approx. 1/10") diameter.

Ground socket, 2.5 mm (approx. 1/10") diameter.

Modell connector, 20 poles according IEC.  
Connector for additional interface.

### Dimensions:

150 x 90 x 30 mm (approx. 6 inches x 3.5 inches x 1.2 inches),

1200 mm (approx. 47 inches) cable length, transparent lid.



## Verdrahtungsplan der Interface Ein- und Ausgänge · Circuit layout of the Interface Inputs and Outputs

