

# Ansteuerung externer Hardware über den Parallelport

von Klaus Schreiber (Friedrich Leopold Woeste Gymnasium in Hemer)

Über eine 25-polige Sub-D-Buchse und ein entsprechendes Kabel wird die Verbindung zum Parallelport des Rechners hergestellt. Als **Ausgangsleitungen** werden die Datenleitungen  $D_0$  bis  $D_7$  (vgl. 2.) verwendet.

Bei Ausgabe einer "1" leuchtet die zugehörige Diode. Wenn man die Farben der LEDs geeignet wählt, lassen sich auf einfache Weise verschiedenartige Ampelsteuerungen realisieren (vgl. 5.).

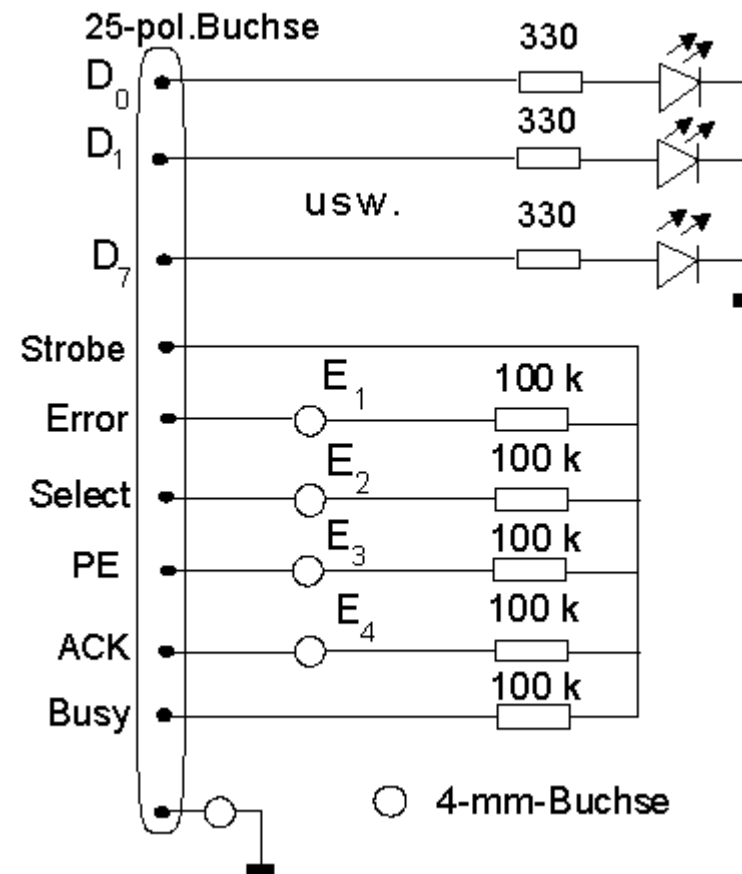


Abb. 1 Interface 1

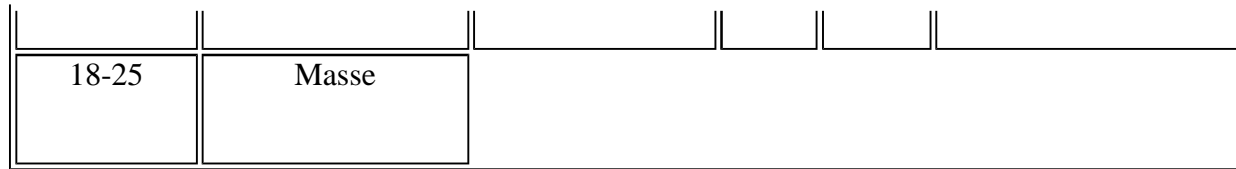
Die **Eingabeleitungen  $E_1$  bis  $E_4$**  werden mit Hilfe des Strobessignals auf "1" gesetzt. Beim Einlesevorgang werden eine bis vier dieser Leitungen über einen hinreichend niederohmigen Widerstand (es reicht das Durchschalten der C-E-Strecke eines Transistors oder das Beleuchten eines lichtempfindlichen Widerstandes) mit Masse verbunden; das zugehörige Bit nimmt dann den Wert "0" an. Zur Herstellung der Schaltverbindungen dienen 4-mm-Buchsen.

Wenn man leistungstärkere Verbraucher wie z.B. die Spulen von Schrittmotoren ansteuern will, so muß man die Ausgangssignale zur Ansteuerung von Transistoren nutzen, die genügend hohe Ströme schalten können. Dann benötigt man aber auch eine zusätzliche Spannungsquelle, deren Minuspol mit der Rechnermasse zu verbinden ist.

## 2. Pinbelegung und Register der parallelen Schnittstelle

Pin-Nr.	Bezeichnung	Register	Bit	Wert	Bemerkung
2	D <sub>0</sub>		0	1	
3	D <sub>1</sub>	Datenregister	1	2	
4	D <sub>2</sub>		2	4	Leitungen sind
5	D <sub>3</sub>	Adresse:	3	8	kurzschlußfest
6	D <sub>4</sub>	378 <sub>hex</sub> = 888 <sub>dez</sub>	4	16	
7	D <sub>5</sub>		5	32	

8	D <sub>6</sub>	WRITE ONLY	6	64	
9	D <sub>7</sub>		7	128	
15	Error	Statusregister	3	8	Die Bits 0 bis 2
13	Select	Adresse:	4	16	können nicht
12	Paper End	379 <sub>hex</sub> = 889 <sub>dez</sub>	5	32	beeinflußt
10	Acknowledge		6	64	werden
11	Busy	READ ONLY	wird im Interface nicht benutzt*)		
1	Strobe	Kontrollregister	0	1	üblicherweise "1"
14	Auto Feed	Adresse:	1	2	diese Leitungen
16	Init	37A <sub>hex</sub> = 890 <sub>dez</sub>	2	4	werden nicht
17	Select In	READ / WRITE	3	8	benutzt



\*) Der Wert dieses Bits wird vom Rechner automatisch invertiert, aufgrund der Interfaceschaltung hat es also den Wert 0.

### 3. Aus- und Eingabe (in Qbasic)

#### a) Ausgabebefehl

***OUT 888, A***

Hierbei ist A eine dezimale Ganzzahl zwischen 0 und 255; in binärer Darstellung hat A demnach 8 Stellen. Es sind diejenigen Datenleitungen auf 5 V gelegt (die zugehörige Leuchtdiode leuchtet also), deren zugehörige Binärstelle eine "1" aufweist, z.B. leuchten bei  $A = 131 = (128 + 2 + 1)$  nur die Dioden an  $D_0$ ,  $D_1$ ,  $D_7$  (vgl. Abschnitt 1, Abb. 1).

#### b) Einlesebefehl

Da die Bits 3 bis 7 des Statusregisters aufgrund der Interfaceschaltung auf "1" gesetzt sind, ist der Registerinhalt nach Start des Rechners davon abhängig, ob die Bits 0 bis 2 intern gesetzt sind oder nicht. Um ein eindeutiges Ergebnis beim Auslesen zu erhalten, ist es sinnvoll, diese Bits zu maskieren. Das kann z.B. auf folgende Weise geschehen:

$$A = (120 - (INP(889) \text{ AND } 120)) / 8 (*)$$

Die Variable A nimmt einen Wert an, der durch den Zustand der Steuerleitungen (vgl. Abschnitt 2) gegeben ist:

$$A = (120 - (x * 2^6 + x * 2^5 + x * 2^4 + x * 2^3)) / 8 ;$$

dabei gilt  $x = 0$ , wenn die zugehörige Leitung mit Masse verbunden, ein Schalter z.B. also geschlossen ist, bzw.  $x = 1$ , wenn die zugehörige Leitung auf 5 V bleibt, ein Schalter z.B. also geöffnet ist.

Durch die Subtraktion von 120 und die anschließende Division durch 8 bekommt die Variable A genau den Wert, der dem Index bzw. der Summe der Indices derjenigen Eingangsleitungen entspricht, die mit Masse verbunden sind: die zugehörigen Schalter sind geschlossen.

Das folgende Beispiel dient zur Erläuterung der Funktionsweise:

Man schließt den Kollektor eines Transistors an die Steuerleitung "Select" (13) und einen Schalter an die Steuerleitung "Error" (15) an; der Emitter bzw. der zweite Anschluß des Schalters sind mit Masse verbunden. Die möglichen Ergebnisse des Befehls (\*) finden sich in der nachstehenden Tabelle:

Transistor	Schalter	Wert von A
gesperrt	geöffnet	$A = (120 - (64+32+16+8))/8 = 0$
gesperrt	geschlossen	$A = (120 - (64 + 32 + 16 + \mathbf{0}))/8 = 1$
durchgeschaltet	geöffnet	$A = (120 - (64 + 32 + \mathbf{0} + 8))/8 = 2$
durchgeschaltet	geschlossen	$A = (120 - (64 + 32 + \mathbf{0} + \mathbf{0}))/8 = 3$

Man kann das Interface natürlich auch mittels einer anderen Programmiersprache, z.B. PASCAL, ansteuern.

#### 4. Interface 2

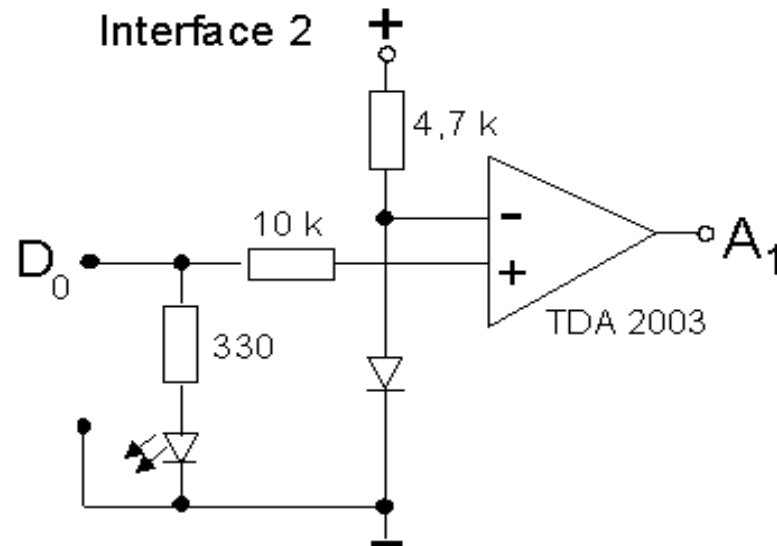


Abb. 2

Im Unterschied zu Interface 1 steuern in dieser Version die vier Datenleitungen ( $D_0$  bis  $D_3$ ) Leistungsoperationsverstärker, deren Ausgänge auf den Pluspol einer Fremdquelle (zugehöriges Bit hat den Wert 1) bzw. auf Masse (zugehöriges Bit hat den Wert 0) gezogen werden.

Der Spannungsteiler am Eingang sorgt dafür, daß der OPV für Fremdspannungen bis 18 V sauber schaltet.

Die vier nicht verwendeten Ausgabelleitungen ( $D_4$  bis  $D_7$ ) können natürlich bei Bedarf zusätzlich verwendet werden, z.B. um einen Schalttransistor anzusteuern. Zu diesem Zweck wird die Basis des Transistors über einen Vorwiderstand mit der gewählten Datenleitung verbunden, der Emitter liegt an Masse und am Kollektor wird die zu schaltende Last angeschlossen.

In der Abb. 2 ist die Schaltung nur für eine der vier Ausgabelleitungen dargestellt, die drei nicht gezeichneten sind völlig identisch. Die Schaltung der Eingabeleitungen entspricht derjenigen für das Interface 1 (vgl. Abschnitt 1, Abb. 1).

Die Ein- bzw. Ausgaberoutinen entsprechen den für das Interface 1 erläuterten (vgl. Abschnitt 3).

## 5. Experimentiervorschläge

### a) Einführende Versuche mit Interface 1

- **Lauflicht:** Man läßt die Zahlen 20, 21, ... , 27 in einer Schleife ausgeben. Als Variante kann man die Reihenfolge durch eine Zufallszahl steuern lassen.
- **Ampel:** Kreuzung zweier Straßen, Straße mit Fußgängerüberweg ohne Anforderung, Blinkampel für verkehrsarme Zeiten (als Umschalter kann z.B. ein Fotowiderstand an einem der Eingänge dienen), Fußgängerüberweg mit Anforderung (Taster an einem oder zwei der Eingänge, Verzögerung der Auslösung durch Programmierung einer Warteschleife), Hauptstraße / Nebenstraße mit Anforderungskontakt (Reedschalter an einem der Eingänge)

### b) Weitere Versuche mit Interface 1

- **Windrichtungsanzeige:** Ausgewertet wird das Signal einer Gruppe von 3 Reflexlichtschranken, die relativ zu einer Winkelcodierscheibe rotieren.
- **Strichcodeleser:** Ausgewertet wird das Signal eines über eine Strichcodemarkierung geführten Lesekopfes (Reflexlichtschranke).
- **Matrixanzeige:** 35 Leuchtdioden werden in einer 7 x 5 Matrix angeordnet. Zur Auswahl von einer der 7 Zeilen benötigt man 3 Bit, die restlichen 5 Bit dienen zur Ansteuerung der 5 in einer Zeile befindlichen Dioden. Indem man die Zeilen mittels einer Programmschleife ständig nacheinander ansteuert, leuchtet das gewünschte Muster der Anzeige auf. Eine Ansteuerschaltung für die Diodenmatrix findet man im Heft "Prozeßdatenverarbeitung".
- **Schrittmotor:** Die Ansteuerung von Schrittmotoren läßt sich auf verschiedene Weisen realisieren: 4 Bit steuern jede Spule einzeln, man verwendet eine Ansteuerschaltung, die ein Taktsignal und ein Steuersignal für Rechts- bzw. Linksdrehung erfordert.

### c) Versuche mit Interface 2

- **Schrittmotormodell:** 4 gleiche Spulen sind jeweils um 90° versetzt angeordnet; als Rotor dient eine Magnetnadel. In der bipolaren Anordnung werden je zwei Spulen (die sich gegenüberstehen) in Reihe geschaltet und mit den Ausgangsleitungen  $D_0 / D_1$  bzw.  $D_2 / D_3$  verbunden. Mit dieser Anordnung lassen sich Voll- und Halbschrittbetrieb nachvollziehen.
- **Aufzugmodell:** Eine seilgeführte Blende wird durch einen Getriebemotor nach oben bzw. unten bewegt. Dabei passiert sie 3 Lichtschranken (K, E, I) ; zusätzlich ist noch ein Taster vorhanden. Diese Bauteile werden mit den 4 Eingangsleitungen verbunden; der Motor wird z.B. zwischen  $D_0 / D_1$  angeschlossen.
- **D-/A-Wandler:** Die 4 Ausgänge  $D_0$  bis  $D_3$  werden an ein Netzwerk aus entsprechend bemessenen Widerständen angeschlossen. Damit läßt sich das Prinzip der gewichteten Ströme demonstrieren.



- A-/D-Wandler: Das beschriebene D-/A-Wandler-Modell läßt sich zu einem Modellversuch für die A-/D-Wandlung ausbauen, wenn man als Vergleichsverstärker einen Operationsverstärker nimmt, dessen Ausgangssignal über eine der Eingangsleitungen in den Rechner eingelesen wird.
- Servomotor: Das D-/A-Wandler-Modell gibt in diesem Fall dazu eine Spannung vor, die zu einer bestimmten Motorstellung führt. Der Spannungsvergleich wird wieder mit einem Operationsverstärker durchgeführt.

## 6. Beispielprogramme (in Qbasic) zu den Experimenten

### a) Lauflicht

#### Programm: Schaltung und Bemerkungen

```
CLS                                Interface1 ohne weitere Schaltung,
FOR I = 0 TO 7                    auch zum Testen geeignet
    OUT 888, 2^I
    SLEEP 3                       Die 8 LEDs leuchten nacheinander
NEXT I                            für jeweils ca. 3 Sekunden.
END
```

### b) Fußgängerüberwegampel mit Anforderungskontakt

#### Programm: Schaltung und Bemerkungen

```
CLS: A=0                          Interface 1
DO                                Bitbelegung:
    WHILE A<>1                    D0 bis D2 : Autofahrer rot - gelb - grün
        GOSUB Einlesen           D3 und D5 : Fußgänger rot - grün
    WEND                         E1 : Tasterverbindung zur Masse (Schließer)

    OUT 888, 12 : SLEEP 4
    OUT 888, 10 : SLEEP 2
    OUT 888, 9 : SLEEP 2
    OUT 888, 33 : SLEEP 10
    OUT 888, 9 : SLEEP 2
```

Das Programm läuft endlos, bis es mit STRG + PAUSE abgebrochen wird

```
OUT 888, 11 : SLEEP 2
OUT 888, 12 : SLEEP 4
OUT 888, 8 : SLEEP 6
A=0
LOOP
END

Einlesen:
A = (120 -(INP(889) AND 120)) / 8
RETURN
```

### c) Schrittmotor (bipolar)

#### Programm: Schaltung und Bemerkungen

Unterprogramm für Links- Interface 2; Versorgungsspannung ca. 12 V; Drehung: (\*)  
die 2 Spulenpaare werden mit den Ausgängen der Operationsverstärker verbunden:  
Links: Spulenpaar 1 mit  $A_0 / A_1$

```
OUT 888, 1      Spulenpaar 2 mit  $A_2 / A_3$ 
GOSUB Warten(**)
OUT 888, 4      (*) : Die tatsächliche Drehrichtung hängt
GOSUB          Warten natürlich vom Aufbau ab.
OUT 888, 2
GOSUB Warten  (**) : In diesem Unterprogramm muß eine dem
OUT 888, 8      jeweiligen Aufbau und Rechner
GOSUB Warten  entsprechende Zeitverzögerung
RETURN          programmiert werden
```

### d) Steuerung des Aufzugmodells (vgl. S. 6)

#### Programm: Schaltung und Bemerkungen

```

CLS: A=0 : OUT 888,0
Anfang:
WHILE A <> 3
    GOSUB Einlesen
    OUT 888,1
WEND bewegung der Blende führt.
PRINT " Aufzug im 1. Stock"
OUT 888, 0
SLEEP 5
WHILE A <> 5
    OUT 888, 2
    GOSUB Einlesen
WEND
OUT 888, 0
PRINT "Aufzug im Erdgeschoß"
SLEEP 5

WHILE A<> 6
    OUT 888, 2
    GOSUB Einlesen
WEND
OUT 888, 0
PRINT "Aufzug im Keller"
SLEEP 5
WHILE A <> 5
    OUT 888,1
    GOSUB Einlesen
WEND
OUT 888, 0
PRINT "Erdgeschoß"
SLEEP 5

GOTO Anfang

```

## Interface 2

Der Getriebemotor ist mit den Ausgängen der Operationsverstärker 1 und 2 verbunden, und zwar z.B. so, daß das Setzen von Bit 0 zur Aufwärts-, das Setzen von Bit 1 zur Abwärts-

(Wenn beide Bit den gleichen Wert haben, bleibt der Motor stehen.) Die Versorgungsspannung ist auf den Motor abzustimmen.

Die Eingangsleitungen  $E_1$  (unten, K),

$E_2$  (Mitte, E) und  $E_3$  (oben, I) werden durch

Lichtschraken beeinflusst, die durch die Blende unterbrochen werden.

Für die Logik ist zu beachten, daß ein beleuchteter LDR das zugehörige Bit auf "0" setzt, ein nicht beleuchteter LDR das zugehörige Bit auf "1" läßt. Daraus folgt, daß die verdunkelten Lichtschraken an den Werten 3 (oben), 5 (Mitte) und 6 (unten) zu erkennen sind. Falls alle Lichtschraken frei sind, wird die Zahl 7 eingelesen.

## Umkehr der Drehrichtung des Motors

Das Programm läuft endlos, bis es mit STRG + PAUSE abgebrochen wird.

Die Blende fährt zunächst bis zur oberen Lichtschrake, bleibt dort ca. 5 Sekunden, fährt dann abwärts bis zur mittleren Lichtschrake, bleibt

END

Einlesen:

$A = (120 - (\text{INP}(889) \text{ AND } 120)) / 8$

RETURN

dort ca. 5 Sekunden, fährt dann abwärts bis zur unteren Lichtschranke, bleibt dort ca. 5 Sekunden, fährt dann aufwärts bis zur mittleren Lichtschranke usw. Der erreichte Ort wird jeweils auf dem Bildschirm ausgegeben.

### **Literatur:**

[1] H.Schmidt / W. Weber, Messen und Experimentieren mit dem PC, Dümmler 4236, Reihe Computerpraxis Physik, Bonn 1993

[2] Deutsches Institut für Fernstudien (DIFP), Computer im Physikunterricht, Studienbrief 3A: Ankopplung des Computers an Realexperimente, Tübingen 1990

[3] Prozeßdatenverarbeitung in der Reihe Informatik 9/10, LSW