

9765

Technic



---

Brugsanvisning

pp. 3-12

---

User instructions

pp. 13-22

---

Mode d'emploi

pp. 23-32

---

Gebruiksaanwijzing

pp. 33-42

---

Istruzioni per l'uso

pp. 43-52

---

Bruksanvisning

pp. 53-62

---

Gebrauchshinweise

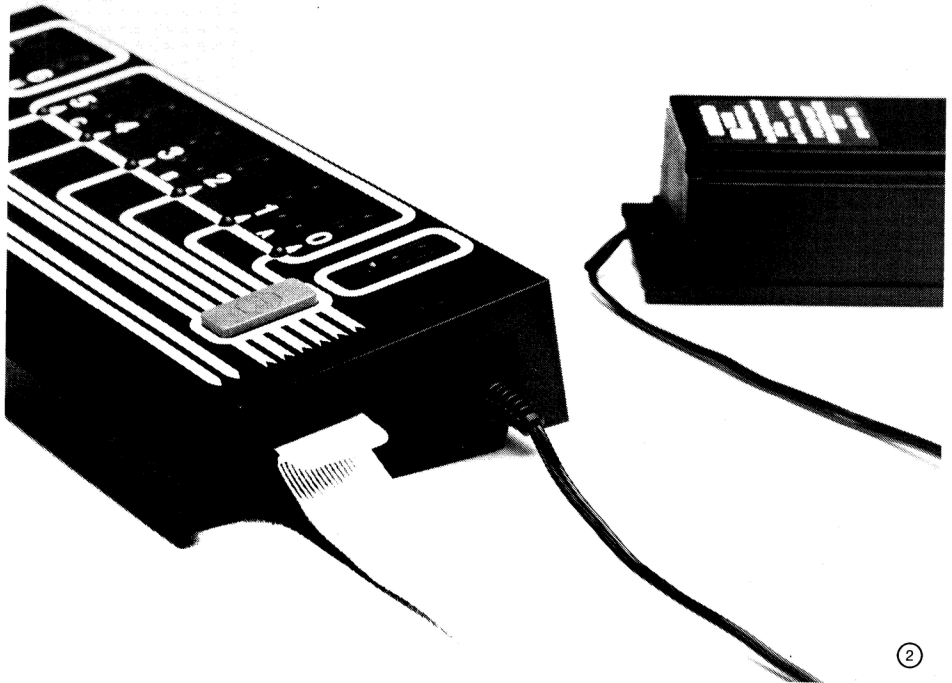
pp. 63-72

---





①



②

# Brugsanvisning – C 64 kabel

Dette LEGO® Technic Control sæt indeholder et computer-kabel og en brugsanvisning.

For at sikre en korrekt anvendelse gennemgås i det følgende:

1. Tilslutning til computer
2. Programmering af computeren til styring af LEGO Interface A

## Fejlsøgning

Såfremt der opstår problemer med interface og modeller se afsnittet vedrørende »Fejlsøgning«.

## Tilslutning til computer

Tilslutning til computeren foretages på følgende måde:

1. Sluk computeren (for at undgå eventuel beskadigelse)
2. Tilslut kablet til computerens user-port (jvf. ill. 1)
3. Tilslut kablet til interface boksen (jvf. ill. 2)
4. Tilslut transformatorledningen til interface boksen (jvf. ill. 2)
5. Tænd transformatoren
6. Tænd computeren

*Kontrollér at den røde stopknap er ude.*

## Programmering af computeren til styring af LEGO Interface A

I det følgende gennemgås:

1. Initialisering af forbindelsen til LEGO Interface A
2. Programmering af udgangene 0-5
3. Aflæsning af indgangene 6 og 7

Indtast programlinierne i bilag A (side 73-74)

REM-sætningerne i BASIC-programmet behøver ikke at blive indtastet, de har ingen funktionel betydning.

Gem eventuelt programmet på diskette eller bånd. Lav om muligt en udskrift og kontrollér for eventuelle indtastningsfejl.

Programmet består af følgende 5 procedurer/subrutiner:

1. init
2. biton
3. bitoff
4. getbit
5. wait

Proceduren init initialiserer forbindelsen til interfacet. Den kræver ingen parametre. Proceduren skal være i 1. linie af alle programmer.

Proceduren biton tænder en udgang på interfacet. Udgangsnummeret overføres som parameter. I BASIC-programmet hedder den variabel hvormed parameteren overføres NUM%. Parameteren skal ligge mellem 0 og 5.

Proceduren bitoff slukker en udgang på interfacet. For udgangsnummeret gælder det samme som ved biton.

Funktionen/subrutinen getbit læser status på en indgang. Indgangsnummeret overføres som parameter. I BASIC-programmet hedder den variabel hvormed parameteren overføres NUM%. Status returneres i variabelen Y%. Indgangsnummeret skal være 6 eller 7.

Proceduren wait venter et antal sekunder. Antallet overføres som parameter. I BASIC-programmet overføres parameteren ved hjælp af variabelen TIM%.

Den overførte parameter skal være positiv og mindre end 32768.

### Anvendelsen af programmet fremgår af følgende:

Initialiseringen af forbindelsen til LEGO Interface A foretages ved at indtaste følgende eksempel:

<b>Eksempel 1A, CBM 64 BASIC</b>	<b>Kommentarer/forklaring</b>
10 GOSUB 10000: REM INIT 20 END	Initialiserer ind- og udgange

<b>Eksempel 1B, COMAL 80</b>	<b>Kommentarer/forklaring</b>
0010 init 0020 END	Initialiserer ind- og udgange

Indtast »RUN« og bemærk at lyset på udgangene 0-5 slukkes.

Computeren er nu klar til at anvende interfaceet.

Ved eventuelle fejlmeddelelser kontrollér det indtastede program mod bilag A.

### Programmering af udgangene 0-5

Indtast følgende eksempel:

<b>Eksempel 2A, CBM 64 BASIC</b>	<b>Kommentarer/forklaring</b>
10 GOSUB 10000: REM INIT 20 NUM%=3	} Initialiserer ind- og udgange } Tænder udgang 3
30 GOSUB 11000: REM BITON 40 TIM%=4	
50 GOSUB 14000: REM WAIT 60 NUM%=4	} Venter 4 sekunder } Tænder udgang 4
70 GOSUB 11000: REM BITON 80 TIM%=2	
90 GOSUB 14000: REM WAIT 100 NUM%=3	} Venter 2 sekunder } Slukker udgang 3
110 GOSUB 12000: REM BITOFF 120 TIM%=2	
130 GOSUB 14000: REM WAIT 140 NUM%=4	} Venter 2 sekunder } Slukker udgang 4
150 GOSUB 12000: REM BITOFF 160 END	

<b>Eksempel 2B, COMAL 80</b>	<b>Kommentarer/forklaring</b>
0010 init	Initialiserer ind- og udgange
0020 biton (3)	Tænder udgang 3
0030 wait (4)	Venter 4 sekunder
0040 biton (4)	Tænder udgang 4
0050 wait (2)	Venter 2 sekunder
0060 bitoff (3)	Slukker udgang 3
0070 wait (2)	Venter 2 sekunder
0080 bitoff (4)	Slukker udgang 4
0090 END	

Hvis der opstår fejl så kontrollér det indtastede eks. og procedureerne/subrutinerne i bilag A.

Bemærk at »biton«/»bitoff« kan tænde/slukke en enkelt udgang uden at påvirke status på de andre. Prøv at tænde en udgang, vente, tænde en anden udgang, vente, slukke førstnævnte udgang, vente og slukke sidstnævnte udgang.

### Aflæsning af indgangene 6 og 7

Indgangene 6 og 7 kan bruges til at modtage signaler fra LEGO optosensoren. Tilslut en optosensor til indgang 6.

Bevæg et gult LEGO element forbi sensoråbningen på optosensoren. Bemærk at den grønne lampe ved indgangen blinker i takt med bevægelsen. Når den grønne lampe lyser, aflæses værdien 1 af computeren; når lampen er slukket, aflæses værdien 0.

Vi vil nu lave et program, der udskriver status 100 gange fra indgang 6.

Indtast følgende eksempel 3:

#### Eksempel 3A, CBM 64 BASIC

```
10 GOSUB 10000: REM INIT
20 NUM%=6
30 NO%=100
40 FOR X=1 TO NO%
50 GOSUB 13000: REM GETBIT
60 PRINT Y%
70 NEXT X
80 END
```

#### Kommentarer/forklaring

Initialiserer ind- og udgange  
Indgangens nummer  
Antal gennemløb

Hent status på indgangen  
Udskriv status

#### Eksempel 3B, COMAL 80

```
0010 init
0020 antal:=100
0030 FOR x:=1 TO antal DO
0040   PRINT getbit (6)

0050 ENDFOR x
0060 END
```

#### Kommentarer/forklaring

Initialiserer ind- og udgange  
Antal gennemløb

Udskriver status på  
indgang 6

Hvis der opstår fejl så kontrollér det indtastede eks. og procedureerne/subrutinerne i bilag A.

Kør programmet og husk at aktivere sensoren.

Prøv at tilslutte en sensor til indgang 7 og udskrive status.

Byg en model med motor, optosensor og tælleskive, f.eks. et pariserhjul. Tilsut motoren til udgang 1 og optosensoren til indgang 6.

*Tælleskivens grove inddeling skal vende ind mod sensoråbningen.*

I det følgende eksempel bruges optosensoren til at tælle med.

*Programmet tæller 24 skift på indgang 6.*

I eksempel 4 kan BASIC-programmet kun tælle op til 3 skift i sekundet. Hvis man skal tælle hurtigere bliver man nødt til at bruge et underprogram, som er skrevet i maskinkode.

**Eksempel 4A, CBM 64 BASIC**

```

10 GOSUB 10000: REM INIT
20 NO%=24
30 NUM%=6
40 GOSUB 13000: REM GETBIT
50 QST%=Y%
60 NUM%=1
70 GOSUB 11000: REM BITON
80 NUM%=6
90 FOR QI=1 TO NO%
100 GOSUB 13000: REM GETBIT
110 IF Y%=QST% THEN GOTO 100
120 QST%=Y%
130 NEXT QI
140 NUM%=1
150 GOSUB 12000: REM BITOFF
160 END

```

**Kommentarer/forklaring**

```

Initialiserer ind- og udgange
Antal skift
} Henter og gemmer status
  på indgang 6
} Tænder udgang 1
  Indgangens nummer
} Venter indtil indgangen
  skifter status
  Gem ny status
} Slukker udgang 1

```

**Eksempel 4B, COMAL 80**

```

0010 init
0020 biton (1)
0030 no:=24
0040 FOR qi:=1 TO no DO
0050   qst:=getbit (6)
0060   REPEAT
0070     UNTIL getbit(6)<>qst
0080   ENDFOR qi
0090 bitoff(1)
0100 END

```

**Kommentarer/forklaring**

```

Initialiserer ind- og udgange
Tænder udgang 1
Antal skift
} Status på indgang 6 gemmes
  } Vent indtil indgang 6
  } skifter status
} Slukker udgang 1

```

Hvis der opstår fejl så kontrollér det indtastede eks. og procedurerne/subrutinerne i bilag A.

Lav et program hvor optosensoren tilsluttes indgang 7 og skal tælle 12 skift.

BASIC-programmet kan laves lidt hurtigere ved at springe parameterkontrollen over. Så skal linienumrene i GOSUB-kommandoen laves om til:

```

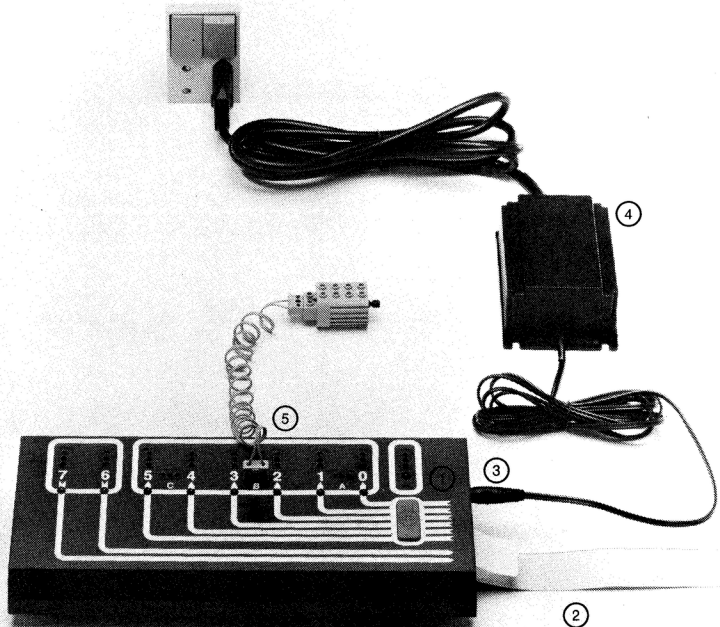
BITON:   GOSUB 11020
BITOFF:  GOSUB 12020
GETBIT:  GOSUB 13020
WAIT:    GOSUB 14020

```

# Fejlsøgning

## Hurtig fejlsøgningsvejledning

Ved problemer med interface og modeller vil nedenstående korte check-liste ofte være nok til at finde fejlen.



1. Den røde stopknap på interfacet skal være ude.
2. Kablet til computeren skal være monteret korrekt (i interface og computer) – se »Tilslutning til computer«.
3. Transformatorens stik skal være monteret korrekt i interfacet.
4. Transformatorens stik skal være tilsluttet net (den røde lampe over stopknappen skal lyse).
5. Fungerer udgangene 0-5? (prøv med lyssten).

Kontrollér endvidere, at modellen er korrekt tilsluttet, og at der ikke er brud på ledningerne.

## Udvidet fejlsøgningsvejledning

En mere udførlig checkliste følger i det nedenstående.

Denne checkliste omfatter:

1. LEGO Interface A og nettilslutning
  - 1.1. Strøm fra transformator
  - 1.2. Fastspændingsudgang
  - 1.3. Indgange
2. LEGO Interface A, kommunikation og funktion
  - 2.1. Opstart
  - 2.2. Udgange
  - 2.3. Indgange

### CHECK PROCEDURER:

#### 1.0. Check for visuelle defekter

##### 1.1. Strøm fra transformator

BEMÆRK: Der må ikke være forbundet noget til fastspænding.

UDFØR	TEST	HVIS JA	HVIS NEJ
1. Tilslut transformator til net			
2. Tilslut transformator til Interface A	2a. Lyser rød lysdiode ved konstant spændingsudgangen over stopknappen?	Transformator OK	Undersøg om der er netspænding til rådighed
			Afprøv transformator med andet interface
			Afprøv interface med anden transformator

##### 1.2. Fastspændingsudgang

UDFØR	TEST	HVIS JA	HVIS NEJ
1. Tilslut 1 stk. lyssten til fastspændingsudgangen	1a. Lyser lyssten?	Fastspændingsudgang OK	Afprøv fastspændingsudgang med anden lyssten
			Afprøv med anden LEGO ledning

### 1.3. Indgange

Indgangene 6 og 7 testes af hver for sig.

UDFØR	TEST	HVIS JA	HVIS NEJ
1. Tilslut optosensor til indgang			
2. Bevæg et gult LEGO element forbi sensoråbningen på optosensoren	2a. Blinker grøn lysdiode ved indgangen i takt med bevægelsen?	Indgang og optosensor OK	Afprøv med anden LEGO ledning
			Afprøv indgang med anden optosensor
			Afprøv indgang med lyssten (udfør 3 mfl.)
3. Fjern optosensor			
4. Tilslut lyssten til indgang (for at sikre at lyssten er OK afprøves denne først med konstant spændingsudg.)	4a. Lyser grøn lysdiode ved indgangen, når lyssten er tilsluttet?	Fortsæt testen med 4b	Indgang er defekt
	4b. Er grøn lysdiode slukket, når intet er tilsluttet?	Indgang er OK	Indgang er defekt

## 2.1. Opstart

UDFØR	TEST	HVIS JA	HVIS NEJ
1. Opstil, forbind og tilslut computerinstallationen i henhold til fabrikan- tens anvisninger			
2. Sørg for at compu- teren er slukket			
3. Tilslut LEGO In- terface A til com- puter og transfor- mator. Stopknap skal væ- re ude			
4. Tilslut transforma- tor til netspænding			
5. Tænd for computer- installationen og transformator	5a. Lyser samtlige lysdioder ved ud- gange?	Lysdioder på udgan- gene er OK	Undersøg om net- spænding er til råd- ighed
			Undersøg om stop- knap på interface er ude
			Undersøg om alle til- slutninger er korrekt udført
			Afprøv med andet interface
			Afprøv med andet interfacekabel
			Afprøv med anden computerinstallation
6. Indtast: POKE 56579,63 POKE 56577,0	6a. Slukkes samtlige udgange?	Gå videre til afsnit 2.2: udgange	Afprøv med andet interfacekabel
			Afprøv med andet interface
			Afprøv med anden computerinstallation

## 2.2. Udgange

UDFØR	TEST	HVIS JA	HVIS NEJ
1. Gennemfør 2.1. opstart procedure			
2. Indtast: POKE 56577,41	2a. Lyser lysdioderne ved udgang 0, 3 og 5?	Kommunikation til udgangene OK	Afprøv med andet interfacekabel
Indtast: POKE 56577,22	Lyser lysdioderne ved udgang 1, 2 og 4?		Afprøv med andet interface
			Afprøv med anden computerinstallation
3. Tilslut LEGO 4,5 V DC motor til udgang A, B og C. Udgangene A, B, C testes hver for sig			
4. Indtast: POKE 56577,42	4a. Starter motoren?	Fortsæt testen	Afprøv med anden LEGO ledning
			Afprøv med anden motor
			Udgang defekt
5. Indtast: POKE 56577,21	5a. Skifter motoren omdrejningsretning?	Udgangene OK	Udgang defekt
6. Indtast: POKE 56577,0 Motoren standser			

### 2.3. Indgange

UDFØR	TEST	HVIS JA	HVIS NEJ
1. Gennemfør 2.1. opstart procedure			
2. Gennemfør 1.3. test af indgange			
3. Tilslut en gul lyssten til indgang 6 og en gul lyssten til indgang 7 Indtast: PRINT PEEK (56577) AND 192	3a. Udskrives tallet 192 på skærmen?	Gå videre i testen	Afprøv indgang med anden LEGO ledning
			Afprøv med andet interfacekabel
			Afprøv indgang med andet interface
			Afprøv med anden computerinstallation
4. Fjern lysstenene fra indgang 6 og 7 Indtast: PRINT PEEK (56577) AND 192	4a. Udskrives tallet 0 på skærmen?	Kabel og indgange OK	Afprøv med andet interfacekabel
			Afprøv indgang med andet interface
			Afprøv med anden computerinstallation

# User instructions – C 64 Cable

This LEGO® Technic Control set contains a computer cable and a set of user instructions.

To ensure correct application a description is given of the following:

1. Connection to computer
2. Programming of computer for control of LEGO Interface A

## Trouble shooting

Should a problem arise with the interface and models, please consult section on 'Trouble shooting'.

## Connection to computer

The cable is connected to the computer as follows:

1. Switch off computer (to avoid possible damage)
2. Connect cable to user-port of computer (cf. ill. 1)
3. Connect cable to interface box (cf. ill. 2)
4. Connect transformer lead to interface box (cf. ill. 2)
5. Switch on transformer
6. Switch on computer

*Check that the red stop button is not depressed.*

## Programming of computer for control of LEGO Interface A

This section describes the following:

1. Initializing of connection to LEGO Interface A
2. Programming of outputs 0-5
3. Reading of inputs 6 and 7

Enter program lines in appendix A (p. 73 to 74)

The REM sentences in the BASIC program need not be entered as they have no functional significance.

If required, save the program on a diskette or tape. If possible, make a print-out and check for possible data input errors.

The program consists of the following 5 procedures/subroutines:

1. init
2. biton
3. bitoff
4. getbit
5. wait

The procedure init initializes the connection to the interface and requires no parameters.

This procedure must be entered on the first line of all programs.

The procedure biton switches on an output on the interface. The output number is transmitted as a parameter. In the BASIC program the variable transmitting the parameter is named NUM%. This parameter must be between 0 and 5.

The procedure bitoff switches off an output on the interface.

For the output number the same applies as with the biton procedure.

The function/subroutine getbit reads the status of an input. The input number is transmitted as a parameter. In the BASIC program the variable transmitting the parameter is named NUM%. The status is returned in the variable Y%. The input number must be 6 or 7.

The procedure wait involves a wait of a number of seconds. This number is transmitted as a parameter. In the BASIC program the parameter is transmitted by means of the variable TIM%. The parameter transmitted must be positive and smaller than 32768.

**Program application**

Initializing of connection to LEGO Interface A is performed by entering the following example:

**Example 1A, CBM 64 BASIC**

```
10 GOSUB 10000: REM INIT
20 END
```

**Comments/explanation**

Initializes inputs and outputs

**Example 1B, COMAL 80**

```
0010 init
0020 END
```

**Comments/explanation**

Initializes inputs and outputs

Enter 'RUN' and note that light on outputs 0-5 extinguishes. The computer is now ready to use the interface. In the event of error messages, check the program entered against appendix A.

**Programming of outputs 0-5**

Enter following example:

**Example 2A, CBM 64 BASIC**

```
10 GOSUB 10000: REM INIT
20 NUM%=3
30 GOSUB 11000: REM BITON
40 TIM%=4
50 GOSUB 14000: REM WAIT
60 NUM%=4
70 GOSUB 11000: REM BITON
80 TIM%=2
90 GOSUB 14000: REM WAIT
100 NUM%=3
110 GOSUB 12000: REM BITOFF
120 TIM%=2
130 GOSUB 14000: REM WAIT
140 NUM%=4
150 GOSUB 12000: REM BITOFF
160 END
```

**Comments/explanation**

Initializes inputs and outputs  
} Switches on output 3  
} Waits 4 seconds  
} Switches on output 4  
} Wait 2 seconds  
} Switches off output 3  
} Waits 2 seconds  
} Switches off output 4

**Example 2B, COMAL 80**

0010 init

0020 biton (3)

0030 wait (4)

0040 biton (4)

0050 wait (2)

0060 bitoff (3)

0070 wait (2)

0080 bitoff (4)

0090 END

**Comments/explanation**

Initializes inputs and outputs

Switches on output 3

Waits 4 seconds

Switches on output 4

Waits 2 seconds

Switches off output 3

Waits 2 seconds

Switches off output 4

In the event of error, check the example entered and the procedures/subroutines in appendix A.

Note that "biton"/"bitoff" can switch on/switch off an individual output without affecting the status of others. Try switching on an output, wait, switch on another output, wait, switch off the first output, wait, and switch off the latter output.

**Reading of inputs 6 and 7**

Inputs 6 and 7 can be used for receiving signals from the LEGO optosensor. Attach an optosensor to input 6. Move a yellow LEGO block past the sensor opening on the optosensor. Note that the green lamp at the input flashes in step with the movement. When the green lamp is lit, the computer reads the value 1; when the lamp is extinguished, the value 0 is read.

We will now make a program which will print out 100 times the status of input 6.

Enter the following example 3:

**Example 3A, CBM 64 BASIC**

10 GOSUB 10000: REM INIT

20 NUM%=6

30 NO%=100

40 FOR X=1 TO NO%

50 GOSUB 13000: REM GETBIT

60 PRINT Y%

70 NEXT X

80 END

**Comments/explanation**

Initializes inputs and outputs

Input number

Number of runs

Get input status

Print status

**Example 3B, COMAL 80**

0010 init

0020 number:=100

0030 FOR x:=1 TO number DO

0040 PRINT getbit (6)

0050 ENDFOR x

0060 END

**Comments/explanation**

Initializes inputs and outputs

Number of runs

Prints status of input 6

In the event of error, check the example entered and the procedures/subroutines in appendix A.

Run the program and remember to activate the sensor.

Try to attach a sensor to input 7 and print status.

Construct a model incorporating a motor, optosensor, and counting disc, e.g. a ferris wheel. Connect the motor to output 1 and the optosensor to input 6. *The rough gradation of the counting disc must face towards the sensor opening.*

In the following example the optosensor is used for counting.

*The program counts 24 shifts on input 6.*

In example 4 the BASIC program can only count up to 3 shifts per second. Counting faster will require use of a subprogram written in machine code.

#### Example 4A, CBM 64 BASIC

```
10 GOSUB 10000: REM INIT
20 NO%=24
30 NUM%=6
40 GOSUB 13000: REM GETBIT
50 QST%=Y%
60 NUM%=1
70 GOSUB 11000: REM BITON
80 NUM%=6
90 FOR QI=1 TO NO%
100 GOSUB 13000: REM GETBIT
110 IF Y%=QST% THEN GOTO 100
120 QST%=Y%
130 NEXT QI
140 NUM%=1
150 GOSUB 12000: REM BITOFF
160 END
```

#### Comments/explanation

```
Initializes inputs and out-
puts
Number of shifts
} Gets and saves status in in-
  put 6
} Switches on output 1
Input number
} Waits until input
  changes status
  Save new status
} Switches off output 1
```

#### Example 4B, COMAL 80

```
0010 init
0020 biton (1)
0030 no:=24
0040 FOR qi:=1 TO no DO
0050   qst:=getbit (6)
0060   REPEAT
0070     UNTIL getbit(6)<>qst
0080 ENDFOR qi
0090 bitoff(1)
0100 END
```

#### Comments/explanation

```
Initializes inputs and out-
puts
Switches on output 1
Number of shifts
Status of input 6 is saved
} Waits until input 6
  changes status
Switches off output 1
```

In the event of error, check the example entered and the procedures/subroutines in appendix A.

Construct a program in which the optosensor is connected to input 7 and is required to count 12 shifts.

The BASIC program can be made somewhat quicker by jumping the parametre check.

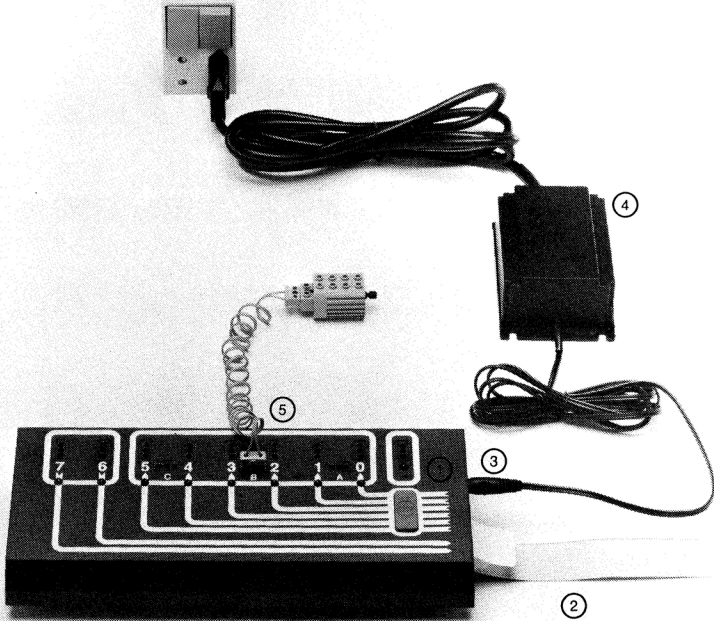
The line numbers in the GOSUB command must then be changed to:

```
BITON:   GOSUB 11020
BITOFF:  GOSUB 12020
GETBIT:  GOSUB 13020
WAIT:    GOSUB 14020
```

# Trouble shooting

## Quick guide to trouble shooting

In case of problems with interface and models, the brief check list below will often be sufficient to pin-point the problem.



1. The red stop button on the interface must be out.
2. The cable to the computer must be fitted correctly (in both interface and computer), see 'Connection to computer'.
3. The transformer plug must be correctly connected to the interface.
4. The transformer must be connected to the mains (red lamp above stop button must be on).
5. Are outputs 0-5 working? (try it with lighting brick).

Check also that the model is correctly connected and that there is no break in the wires.

## Extended guide to trouble shooting

A more detailed check list is provided in the following pages.

This check list covers the following:

1. LEGO Interface A and mains connection
  - 1.1. Power from transformer
  - 1.2. Constant voltage output
  - 1.3. Inputs
2. LEGO Interface A, communication and function
  - 2.1. Startup
  - 2.2. Outputs
  - 2.3. Inputs

### CHECK PROCEDURES:

#### 1.0. Check for visual defects

##### 1.1. Power from transformer

NB: Nothing must be connected to constant voltage output.

ACTION	TEST	IF YES	IF NO
1. Connect transformer to mains			
2. Connect transformer to Interface A	2a. Is the red LED at the constant voltage output above the stop button switched on?	Transformer OK	Check whether mains power is connected <hr/> Test the transformer with another interface <hr/> Test interface with another transformer

##### 1.2. Constant voltage output

ACTION	TEST	IF YES	IF NO
1. Connect 1 lighting brick to the constant voltage output	1a. Is lighting brick switched on?	Constant voltage output OK	Test constant voltage output with another lighting brick <hr/> Test with another LEGO wire

### 1.3. Inputs

Inputs 6 and 7 should be tested separately.

ACTION	TEST	IF YES	IF NO
1. Connect optosensor to input			
2. Move a yellow LEGO block past the sensor opening on the optosensor	2a. Does green LED at input flash in step with movement?	Input and optosensor OK	Test with another LEGO wire
			Test input with another optosensor
			Test input with lighting brick (proceed with no. 3 onwards)
3. Remove optosensor			
4. Connect lighting brick to input (to ensure that lighting brick is OK, test this first on constant voltage output)	4a. Does green LED at input light up when lighting brick is connected?	Continue test with 4b	Input is defective
	4b. Is green LED off when nothing is connected?	Input is OK	Input is defective

## 2.1. Startup

ACTION	TEST	IF YES	IF NO
1. Assemble and connect up the computer installation in accordance with the manufacturer's instructions			
2. Ensure that computer is switched off			
3. Connect LEGO Interface A to computer and transformer. Stop button must be out			
4. Connect transformer to mains			
5. Switch on computer installation and transformer	5a. Are all output LEDs switched on?	Output LEDs are OK	Check whether mains power is connected
			Check whether stop button on interface is out
			Check whether all connections are correct
			Test with another interface
			Test with another interface cable
			Test with another computer installation
6. Enter: POKE 56579,63 POKE 56577,0	6a. Are all outputs switched off?	Proceed to section 2.2: Outputs	Test with another interface cable
			Test with another interface
			Test with another computer installation

## 2.2. Outputs

ACTION	TEST	IF YES	IF NO
1. Implement 2.1 starting procedure			
2. Enter: POKE 56577,41	2a. Are LEDs at outputs 0, 3, and 5 switched on?	Communication to outputs OK	Test with another interface cable
Enter: POKE 56577,22	Are LEDs at outputs 1, 2, and 4 switched on?		Test with another interface
			Test with another computer installation
3. Connect LEGO 4.5 V DC motor to outputs A, B, and C. Test outputs A, B, and C separately			
4. Enter: POKE 56577,42	4a. Does motor start?	Continue test	Test with another LEGO wire
			Test with another motor
			Output defective
5. Enter: POKE 56577,21	5a. Does motor change direction of rotation?	Outputs OK	Output defective
6. Enter: POKE 56577,0 Motor stops			

### 2.3. Inputs

ACTION	TEST	IF YES	IF NO
1. Implement 2.1. starting procedure			
2. Implement 1.3. testing of inputs			
3. Connect a yellow lighting brick to input 6 and a yellow lighting brick to input 7  Enter: PRINT PEEK (56577) AND 192	3a. Does figure 192 appear on display?	Proceed to next step in test	Test input with another LEGO wire Test with another interface cable Test input with another interface Test with another computer installation
4. Remove lighting bricks from inputs 6 and 7 Enter: PRINT PEEK (56577) AND 192	4a. Does figure 0 appear on display?	Cable and inputs OK	Test with another interface cable Test input with another interface Test with another computer installation

# Mode d'emploi câble pour Commodore 64

Le présent équipement de Technic Control LEGO® comprend un câble d'ordinateur et un mode d'emploi.

Pour s'assurer d'un emploi adéquat vérifier ce qui suit:

1. La connexion avec l'ordinateur
2. La programmation de l'ordinateur pour la commande de l'interface A LEGO

## Dépistage des erreurs

S'il y a des problèmes avec l'interface et les modèles voir le paragraphe relatif au "dépistage des erreurs".

## Connexion avec l'ordinateur

La connexion de l'ordinateur s'effectue de la façon suivante:

1. Eteindre l'ordinateur (afin d'éviter un éventuel endommagement)
2. Connecter le câble au port d'entrée-sortie de l'ordinateur (cf. ill. 1)
3. Connecter le câble à la boîte de l'interface (cf. ill. 2)
4. Connecter le fil du transformateur à la boîte de l'interface (cf. ill. 2)
5. Allumer le transformateur
6. Allumer l'ordinateur

*Contrôler que le bouton rouge d'arrêt n'est pas enfoncé.*

## Programmation de l'ordinateur pour la commande de l'interface A LEGO

Nous examinons maintenant les points suivants:

1. Initialisation de la liaison à l'interface A LEGO
2. Programmation des sorties 0-5
3. Relevé des entrées 6 et 7

Entrer les lignes de programme de la pièce jointe A (p. 73-74)

Les phrases REM dans le programme BASIC ne doivent pas nécessairement être entrées car elles n'ont aucune signification sur le plan fonctionnel.

Conservez éventuellement le programme sur disquette ou bande. Faites si possible une sortie et vérifiez s'il y a des erreurs d'entrée.

Le programme comprend les 5 procédures/sous-routines suivantes:

1. init
2. biton
3. bitoff
4. getbit
5. wait

La procédure init initialise la liaison à l'interface. Elle n'exige aucun paramètre. La procédure doit se trouver en première ligne de tous les programmes.

La procédure biton allume une sortie sur l'interface. Le numéro de sortie est transféré comme paramètre. Dans le programme BASIC la variable avec laquelle on transfère le paramètre s'appelle NUM%. Le paramètre doit se trouver entre 0 et 5.

La procédure bitoff éteint une sortie sur l'interface. Pour le numéro de sortie voir la procédure biton.

La fonction/la sous-routine getbit relève le statut sur une entrée. Le numéro d'entrée est transféré comme paramètre. Dans le programme BASIC, la variable avec laquelle on transfère le paramètre s'appelle NUM%. Le statut peut être retourné dans la variable Y%. Le numéro d'entrée doit se trouver entre 6 et 7.

La procédure wait attend un certain nombre de secondes. Ce nombre est transféré comme paramètre. Dans le programme BASIC, le paramètre est transféré à l'aide de la variable TIM%. Le paramètre transféré doit être positif et inférieur à 32768.

**L'utilisation du programme ressort de ce qui suit:**

L'initialisation de la liaison à l'Interface A LEGO se fait en entrant l'exemple suivant:

<b>Exemple 1A, CBM 64 BASIC</b>	<b>Commentaires/explication</b>
10 GOSUB 10000: REM INIT 20 END	Initialise les entrées et les sorties

<b>Exemple 1B, COMAL 80</b>	<b>Commentaires/explication</b>
0010 init 0020 END	Initialise les entrées et les sorties

Entrer "RUN" et remarquer que la lumière s'éteint sur les sorties 0-5.

Maintenant, l'ordinateur est prêt à utiliser l'interface.

Lors d'éventuelles indications d'erreurs, contrôler le programme entré sur base de la pièce jointe A.

**Programmation des sorties 0-5**

Entrer l'exemple suivant:

<b>Exemple 2A, CBM 64 BASIC</b>	<b>Commentaires/explication</b>
10 GOSUB 10000: REM INIT	Initialise les entrées et les sorties
20 NUM%=3	} Allume la sortie 3
30 GOSUB 11000: REM BITON	
40 TIM%=4	} Attend 4 secondes
50 GOSUB 14000: REM WAIT	
60 NUM%=4	} Allume la sortie 4
70 GOSUB 11000: REM BITON	
80 TIM%=2	} Attend 2 secondes
90 GOSUB 14000: REM WAIT	
100 NUM%=3	} Eteint la sortie 3
110 GOSUB 12000: REM BITOFF	
120 TIM%=2	} Attend 2 secondes
130 GOSUB 14000: REM WAIT	
140 NUM%=4	} Eteint la sortie 4
150 GOSUB 12000: REM BITOFF	
160 END	

Exemple 2B, COMAL 80	Commentaires/explication
0010 init	Initialise les entrées et les sorties
0020 biton (3)	Allume la sortie 3
0030 wait (4)	Attend 4 secondes
0040 biton (4)	Allume la sortie 4
0050 wait (2)	Attend 2 secondes
0060 bitoff (3)	Eteint la sortie 3
0070 wait (2)	Attend 2 secondes
0080 bitoff (4)	Eteint la sortie 4
0090 END	

S'il y a des erreurs, contrôler l'exemple entré et les procédures/sous-routines de la pièce jointe A.

Remarquer que "biton"/"bitoff" peuvent allumer/éteindre une seule sortie sans influencer le statut des autres. Essayer d'allumer une sortie, d'attendre, d'allumer une autre sortie, d'attendre, d'éteindre la première sortie, d'attendre et d'éteindre la dernière sortie.

### Relevé des entrées 6 et 7

Les entrées 6 et 7 s'utilisent pour recevoir les signaux de l'optodétecteur LEGO. Connecter un optodétecteur à l'entrée 6.

Faites mouvoir un élément LEGO jaune devant l'ouverture de détection de l'optodétecteur. Remarquer que la lampe verte près de l'entrée clignote en concordance avec le mouvement. Lorsque la lampe verte brille, on relève la valeur 1 de l'ordinateur; lorsque la lampe est éteinte, on relève la valeur 0.

Nous allons maintenant faire un programme qui sort le statut 100 fois de l'entrée 6.

Entrer l'exemple 3 suivant:

Exemple 3A, CBM 64 BASIC	Commentaires/explication
10 GOSUB 10000: REM INIT	Initialise les entrées et les sorties
20 NUM%=6	Numéro d'entrée
30 NO%=100	Nombre de passages
40 FOR X=1 TO NO%	
50 GOSUB 13000: REM GETBIT	Cherche le statut sur l'entrée
60 PRINT Y%	Sort le statut
70 NEXT X	
80 END	

Exemple 3B, COMAL 80	Commentaires/explication
0010 init	Initialise les entrées et les sorties
0020 nombre:=100	Nombre de passages
0030 FOR x:=1 TO nombre DO	
0040 PRINT getbit (6)	Sort le statut sur l'entrée 6
0050 ENDFOR x	
0060 END	

S'il y a erreur, contrôler l'exemple entré et les procédures/sous-routines de la pièce jointe A.

Exécutez le programme et souvenez-vous d'activer le détecteur.

Essayez de connecter un détecteur à l'entrée 7 et de sortir le statut.

Construisez un modèle avec moteur, optodétecteur et cadran compteur, p.ex. une grande roue. Connectez le moteur à la sortie 1 et l'optodétecteur à l'entrée 6.

*La graduation grossière du cadran compteur doit se tourner vers l'ouverture du détecteur.*

Dans l'exemple suivant, on emploie l'optodétecteur pour compter.

*Le programme compte 24 changements sur l'entrée 6.*

Dans l'exemple 4, le programme BASIC peut compter jusqu'à 3 changements par seconde. Si l'on veut compter plus rapidement, on sera obligé d'utiliser un sous-programme écrit en code de machine.

#### Exemple 4A, CBM 64 BASIC

	Commentaires/explication
10 GOSUB 10000: REM INIT	Initialise les entrées et les sorties
20 NO%=24	
30 NUM%=6	} Nombre de changements
40 GOSUB 13000: REM GETBIT	
50 QST%=Y%	} Cherche et garde le statut sur l'entrée 6
60 NUM%=1	
70 GOSUB 11000: REM BITON	} Allume la sortie 1
80 NUM%=6	
90 FOR QI=1 TO NO%	} Numéro d'entrée
100 GOSUB 13000: REM GETBIT	
110 IF Y%=QST% THEN GOTO 100	
120 QST%=Y%	
130 NEXT QI	} Attend jusqu'à ce que l'entrée change de statut
140 NUM%=1	
150 GOSUB 12000: REM BITOFF	} Gardez le nouveau statut
160 END	
	} Eteint la sortie 1

#### Exemple 4B, COMAL 80

	Commentaires/explication
0010 init	Initialise les entrées et les sorties
0020 biton (1)	
0030 no:=24	Allume la sortie 1
0040 FOR qi:=1 TO no DO	
0050 qst:=getbit (6)	} Nombre de changements
0060 REPEAT	
0070 UNTIL getbit(6)<>qst	} Le statut sur l'entrée 6 est gardé
0080 ENDFOR qi	
0090 bitoff(1)	} Attend jusqu'à ce que l'entrée 6 change de statut
0100 END	
	Eteint la sortie 1

S'il y a des erreurs, contrôler l'exemple entré et les procédures/sous-routines de la pièce jointe A.

Faites un programme où l'optodétecteur est connecté à l'entrée 7 et doit compter 12 changements.

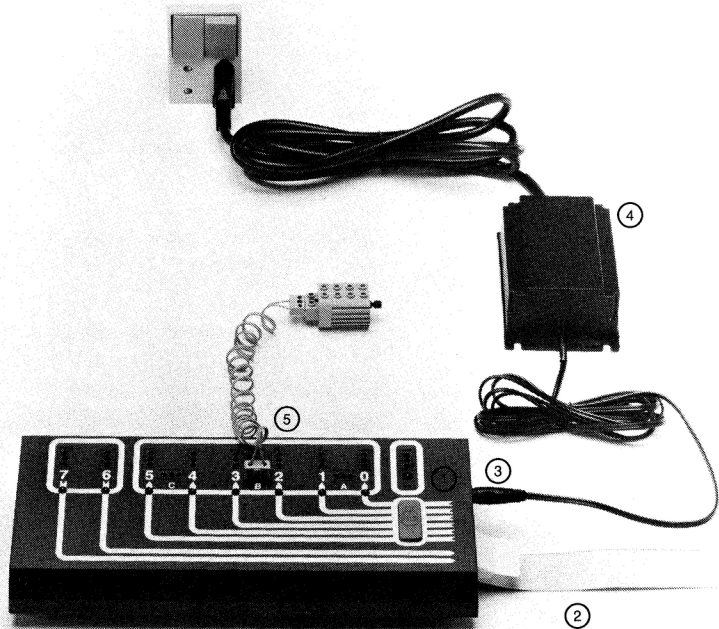
Le programme BASIC peut être rendu un peu plus rapide si l'on saute le contrôle de paramètres. Dans ce cas on doit changer les numéros de ligne dans la commande GOSUB en:

BITON: GOSUB 11020  
BITOFF: GOSUB 12020  
GETBIT: GOSUB 13020  
WAIT: GOSUB 14020

## Dépistage des erreurs

Mode d'emploi pour un dépistage rapide des erreurs

S'il y a des problèmes avec l'interface ou les modèles, la courte liste de vérification ci-dessous sera souvent suffisante pour trouver l'erreur.



1. Le bouton d'arrêt rouge sur l'interface ne doit pas être enfoncé.
2. Le câble menant à l'ordinateur doit être monté de façon correcte (dans l'interface et l'ordinateur), cf. "Connection avec l'ordinateur".
3. La fiche du transformateur doit être montée de façon correcte dans l'interface.
4. Le transformateur doit être sous tension (la lampe rouge au-dessus du bouton d'arrêt doit être allumée).
5. Vérifier que les sorties 0-5 fonctionnent bien (essayer avec une brique lumineuse).

Contrôler en outre que le modèle est connecté de façon correcte et qu'il n'y a pas de ruptures de câble.

## Mode d'emploi pour un dépistage plus approfondi des erreurs

Nous vous donnons ci-après une liste de vérification plus détaillée.

Cette liste comprend:

1. Interface A LEGO et mise sous tension
  - 1.1. Courant venant du transformateur
  - 1.2. Sortie de tension constante
  - 1.3. Entrées
2. Interface A LEGO, communication et fonction
  - 2.1. Mise en marche
  - 2.2. Sorties
  - 2.3. Entrées

### PROCEDURES DE VERIFICATION:

#### 1.0. Vérification des erreurs visuelles

##### 1.1. Courant venant du transformateur

REMARQUE: Rien ne doit être relié à la tension constante.

A EFFECTUER	TEST	SI OUI	SI NON
1. Mettre le transformateur sous tension			
2. Connecter le transformateur à l'Interface A	2a. La diode lumineuse rouge brille-t-elle à la sortie de tension constante au-dessus du bouton d'arrêt?	Transformateur OK	Vérifier s'il y a de la tension disponible
			Essayer le transformateur avec un autre interface
			Essayer l'interface avec un autre transformateur

##### 1.2. Sortie de tension constante

A EFFECTUER	TEST	SI OUI	SI NON
1. Connecter une brique lumineuse à la sortie de tension constante	1a. La brique lumineuse brille-t-elle?	Sortie de tension constante OK	Essayer la sortie de tension constante avec une autre brique lumineuse
			Essayer avec un autre câble LEGO

### 1.3. Entrées

Tester les entrées 6 et 7 séparément.

A EFFECTUER	TEST	SI OUI	SI NON
1. Connecter l'optodétecteur à l'entrée			
2. Faire mouvoir un élément LEGO jaune devant l'entrée du détecteur sur l'optodétecteur	2a. La diode lumineuse verte brille-t-elle à l'entrée en concordance avec le mouvement?	Entrée et optodétecteur OK	Essayer avec un autre câble LEGO
			Essayer l'entrée avec un autre optodétecteur
			Essayer l'entrée avec une brique lumineuse (Effectuer le No. 3 et autres)
3. Enlever l'optodétecteur			
4. Connecter la brique lumineuse à l'entrée (afin d'assurer que la brique lumineuse est OK on essayera celle-ci d'abord avec une sortie de tension constante)	4a. La diode lumineuse verte brille-t-elle à l'entrée lorsque la brique lumineuse est connectée?	Continuer le test par 4b	L'entrée est défectueuse
	4b. La diode lumineuse verte est-elle éteinte lorsque rien n'est connecté?	L'entrée est OK	L'entrée est défectueuse

## 2.1. Mise en marche

A EFFECTUER	TEST	SI OUI	SI NON
1. Monter, relier et connecter l'installation d'ordinateur conformément aux instructions du fabricant			
2. Veiller à ce que l'ordinateur soit éteint			
3. Connecter l'Interface A LEGO à l'ordinateur et au transformateur. La bouton d'arrêt ne doit pas être enfoncé			
4. Mettre le transformateur sous tension			
5. Allumer l'installation d'ordinateur et de transformateur	5a. Les diodes lumineuses brillent-elles toutes aux sorties?	Les diodes lumineuses des sorties sont OK	Vérifier s'il y a de la tension disponible
			Vérifier si le bouton d'arrêt sur l'interface n'est pas enfoncé
			Vérifier si toutes les connexions ont été correctement effectuées
			Essayer avec un autre interface
			Essayer avec un autre câble d'interface
			Essayer avec une autre installation d'ordinateur
6. Entrer: POKE 56579,63 POKE 56577,0	6a. Toutes les sorties s'éteignent-elles?	Passer au paragraphe 2.2.: Sorties	Essayer avec un autre câble d'interface
			Essayer avec un autre interface
			Essayer avec une autre installation d'ordinateur

## 2.2. Sorties

A EFFECTUER	TEST	SI OUI	SI NON
1. Effectuer la procédure de mise en marche 2.1.			
2. Entrer: POKE 56577,41	2a. Les diodes lumineuses brillent-elles aux sorties 0, 3 et 5?	La communication aux sorties est OK	Essayer avec un autre câble d'interface
Entrer: POKE 56577,22	Les diodes lumineuses brillent-elles aux sorties 1, 2 et 4?		Essayer avec un autre interface
			Essayer avec une autre installation d'ordinateur
3. Connecter le moteur CC LEGO 4,5 V aux sorties A, B et C. Les sorties A, B et C se testent séparément			
4. Entrer: POKE 56577,42	4a. Le moteur se met-il en marche?	Continuer le test	Essayer avec un autre câble LEGO
			Essayer avec un autre moteur
			La sortie est défectueuse
5. Entrer: POKE 56577,21	5a. Le moteur change-t-il de direction de rotation?	Les sorties sont OK	La sortie est défectueuse
6. Entrer: POKE 56577,0 Le moteur s'arrête			

### 2.3. Entrées

A EFFECTUER	TEST	SI OUI	SI NON
1. Effectuer la procédure de mise en marche 2.1.			
2. Effectuer le test 1.3. des entrées			
3. Connecter une brique lumineuse jaune à l'entrée 6 et une brique lumineuse jaune à l'entrée 7 Entrer: PRINT PEEK (56577) AND 192	3a. Le chiffre 192 apparaît-il sur l'écran?	Continuer le test	Essayer l'entrée avec un autre câble LE-GO
			Essayer avec un autre câble d'interface
			Essayer l'entrée avec un autre interface
			Essayer avec une autre installation d'ordinateur
4. Enlever les briques lumineuses des entrées 6 et 7 Entrer: PRINT PEEK (56577) AND 192	4a. Le chiffre 0 apparaît-il sur l'écran?	Le câble et les entrées sont OK	Essayer avec un autre câble d'interface
			Essayer l'entrée avec un autre interface
			Essayer avec une autre installation d'ordinateur

# Gebruiksaanwijzing Commodore 64/128 kabel

Deze set LEGO<sup>®</sup> kabel voor Commodore 64/128 bevat een computer-kabel en een gebruiksaanwijzing.

In de volgende punten wordt uitgelegd hoe je het set op de juiste manier gebruikt:

1. Aansluiten op de computer
2. Programmeren van LEGO Interface A

## Foutenwijzer

Voor eventuele problemen met de interface en/of de modellen, zie het hoofdstuk "Foutenwijzer".

## Aansluiten op de computer

Wat je moet doen en in welke volgorde wordt hieronder beschreven:

1. Zet de computer uit (dit om eventuele beschadiging te voorkomen)
2. Verbind de kabel met de user-port van de computer (vergelijk illustratie 1)
3. Verbind de kabel met de interface box (vergelijk illustratie 2)
4. Verbind het transformatorstoer met de interface box (vergelijk illustratie 2)
5. Zet de transformator aan
6. Zet de computer aan

*De rode stopknop moet uit (d.w.z. stopknop omhoog!) zijn*

## Programmeren van de computer voor het besturen van de LEGO Interface A

De volgende punten worden achtereenvolgens behandeld:

1. Het gebruiksklaarmaken van de LEGO Interface A
2. Het programmeren van de uitgangen 0-5
3. Het aflezen van de ingangen 6 en 7

Toets de programmaregels in zoals aangegeven in bijlage A (p. 73-74)

De REM-zinnen in het programma BASIC hoeven niet te worden ingetoetst, ze hebben geen functionele waarde.

Leg het programma eventueel vast op diskette of band. Als je een printer hebt maak dan een uitdraai en controleer of er geen typefouten inzitten.

Het programma bestaat uit de volgende 5 procedures/subprogramma's:

1. init
2. biton
3. bitoff
4. getbit
5. wait

De procedure init maakt de verbinding met de interface gebruiksklaar. Hierbij zijn geen parameters nodig. De procedure moet in de eerste regel van alle programma's staan.

De procedure biton schakelt een uitgang van de interface in. Het nummer van de uitgang wordt als parameter ingevoerd. De variabele waarmee de parameter wordt ingevoerd heet in het programma BASIC NUM%. De parameter moet tussen 0 en 5 liggen.

De procedure bitoff schakelt een uitgang van de interface uit. Voor het nummer van de uitgang geldt hetzelfde als onder biton.

De functie getbit leest de status af van een ingang. Het nummer van de ingang wordt ingevoerd als parameter. De variabele waarmee de parameter wordt ingevoerd heet in het programma BASIC NUM%. De status wordt weergegeven als de variabele Y%. Het nummer van de ingang moet 6 of 7 zijn.

De procedure wait wacht een aantal seconden. Het aantal wordt ingevoerd als parameter.

In het programma BASIC wordt de parameter ingevoerd met behulp van de variabele TIM%. De overgedragen parameter moet positief en kleiner dan 32768 zijn.

## Hoe het programma moet worden gebruikt, wordt hieronder uitgelegd:

Om LEGO Interface A klaar te maken voor gebruik moet het volgende voorbeeld worden ingetoetst:

Voorbeeld 1A, CBM 64 BASIC	Kommentaar/uitleg
10 GOSUB 10000: REM INIT 20 END	Maakt de in- en uitgangen gebruiksklaar
Voorbeeld 1B, COMAL 80	Kommentaar/uitleg
0010 init 0020 END	Maakt de in- en uitgangen gebruiksklaar

Toets "RUN" in, je zult zien dat het licht boven de uitgangen 0-5 uitgaat.

De computer kan nu gebruik maken van de interface.

Als er een foutmelding verschijnt, controleer dan of je precies hebt gedaan wat er in bijlage A staat.

## Het programmeren van de uitgangen 0-5

Toets het volgende voorbeeld in:

Voorbeeld 2A, CBM 64 BASIC	Kommentaar/uitleg
10 GOSUB 10000: REM INIT 20 NUM%=3 30 GOSUB 11000: REM BITON 40 TIM%=4 50 GOSUB 14000: REM WAIT 60 NUM%=4 70 GOSUB 11000: REM BITON 80 TIM%=2 90 GOSUB 14000: REM WAIT 100 NUM%=3 110 GOSUB 12000: REM BITOFF 120 TIM%=2 130 GOSUB 14000: REM WAIT 140 NUM%=4 150 GOSUB 12000: REM BITOFF 160 END	Maakt de in- en uitgangen gebruiksklaar } Schakelt uitgang 3 in } Wacht 4 seconden } Schakelt uitgang 4 in } Wacht 2 seconden } Schakelt uitgang 3 uit } Wacht 2 seconden } Schakelt uitgang 4 uit
Voorbeeld 2B, COMAL 80	Kommentaar/uitleg
0010 init 0020 biton (3) 0030 wait (4) 0040 biton (4) 0050 wait (2) 0060 bitoff (3) 0070 wait (2) 0080 bitoff (4) 0090 END	Maakt de in- en uitgangen gebruiksklaar Schakelt uitgang 3 in Wacht 4 seconden Schakelt uitgang 4 in Wacht 2 seconden Schakelt uitgang 3 uit Wacht 2 seconden Schakelt uitgang 4 uit

Als er een foutmelding verschijnt, controleer dan het ingetoetste voorbeeld en de procedures/subprogramma's in bijlage A.  
Denk eraan dat "biton"/"bitoff" de uitgangen onafhankelijk van elkaar kan in-/uitschakelen. Schakel als proef een uitgang in, wacht, schakel een andere uitgang in, wacht, schakel de eerste uitgang uit, wacht en schakel de laatste uitgang uit.

### Aflezen van de ingangen 6 en 7

De ingangen 6 en 7 kunnen gebruikt worden voor de ontvangst van de LEGO optosensoren. Sluit een optosensor aan op ingang 6.

Beweeg een gele LEGO steen langs de sensoropening op de optosensor. Zoals je ziet knippert het groene lampje bij de ingang op de maat van de beweging. Als het groene lampje brandt, zie je als waarde "1" staan op de computer; als het lampje uit is zie je als waarde "0" staan.

We gaan nu een programma maken, dat 100 keer de status van ingang 6 print.

Toets het volgende voorbeeld in:

#### Voorbeeld 3A, CBM 64 BASIC

```
10 GOSUB 10000: REM INIT
20 NUM%=6
30 NO%=100
40 FOR X=1 TO NO%
50 GOSUB 13000: REM GETBIT
60 PRINT Y%
70 NEXT X
80 END
```

#### Kommentaar/uitleg

Maakt de in- en uitgangen gebruiksklaar  
Het nummer van de ingang  
Aantal passages  
  
Haalt status van de ingang  
Print status

#### Voorbeeld 3B, COMAL 80

```
0010 init
0020 aantal:=100
0030 FOR x:=1 TO aantal DO
0040 PRINT getbit (6)
0050 ENDFOR x
0060 END
```

#### Kommentaar/uitleg

Maakt de in- en uitgangen gebruiksklaar  
Aantal passages  
  
Print status van ingang 6

Als er een foutmelding verschijnt, controleer dan het ingetoetste voorbeeld en de procedures/subprogramma's in bijlage A.

Voer het programma uit en vergeet niet de sensor in te schakelen.

Probeer een optosensor aan te sluiten op ingang 7 en de status uit te printen.

Bouw een model met motor, optosensor en telschijf, bijvoorbeeld een reuzenrad. Sluit de motor aan op uitgang 1 en de optosensor op ingang 6.

*De grove gradering van de telschijf moet richting sensoropening worden geplaatst.*

In het volgende voorbeeld wordt de optosensor gebruikt om mee te tellen.

*Het programma telt 24 omwentelingen bij ingang 6.*

In voorbeeld 4 kan het programma BASIC naar 3 omwentelingen per seconde tellen.

Indien er sneller moet worden geteld, zal er gebruik moeten worden gemaakt van een subprogramma, dat in machinecode is geschreven.

**Voorbeeld 4A, CBM 64 BASIC**

```

10 GOSUB 10000: REM INIT
20 NO%=24
30 NUM%=6
40 GOSUB 13000: REM GETBIT
50 QST%=Y%
60 NUM%=1
70 GOSUB 11000: REM BITON
80 NUM%=6
90 FOR QI=1 TO NO%
100 GOSUB 13000: REM GETBIT
110 IF Y%=QST% THEN GOTO 100
120 QST%=Y%
130 NEXT QI
140 NUM%=1
150 GOSUB 12000: REM BITOFF
160 END

```

**Kommentaar/uitleg**

Maakt de in- en uitgangen  
gebruiksklaar  
Aantal omwentelingen  
} Haalt en bewaart status  
van ingang 6

} Schakelt uitgang 1 in  
Nummer van de ingang

} Wacht tot de ingang van  
status verandert  
Bewaart nieuwe status

} Schakelt uitgang 1 uit

**Voorbeeld 4B, COMAL 80**

```

0010 init
0020 biton (1)
0030 no:=24
0040 FOR qi:=1 TO no DO
0050   qst:=getbit (6)
0060   REPEAT
0070     UNTIL getbit(6)<>qst
0080   ENDFOR qi
0090 bitoff(1)
0100 END

```

**Kommentaar/uitleg**

Maakt de in- en uitgangen  
gebruiksklaar  
Schakelt uitgang 1 in  
Aantal omwentelingen

Status van ingang 6 wordt  
bewaard  
} Wacht tot ingang 6 van sta-  
tus verandert  
Schakelt uitgang 1 uit

Als er een foutmelding verschijnt, controleer dan het ingetoetste voorbeeld en de procedures en subprogramma's in bijlage A.

Probeer een programma te maken, waarbij de optosensor wordt aangesloten op ingang 7 en vervolgens 12 omwentelingen moet tellen.

Het programma BASIC kan wat sneller worden gemaakt, door de parametercontrole over te slaan. In dat geval moeten de regelnummers in het GOSUB-kommando worden veranderd in:

```

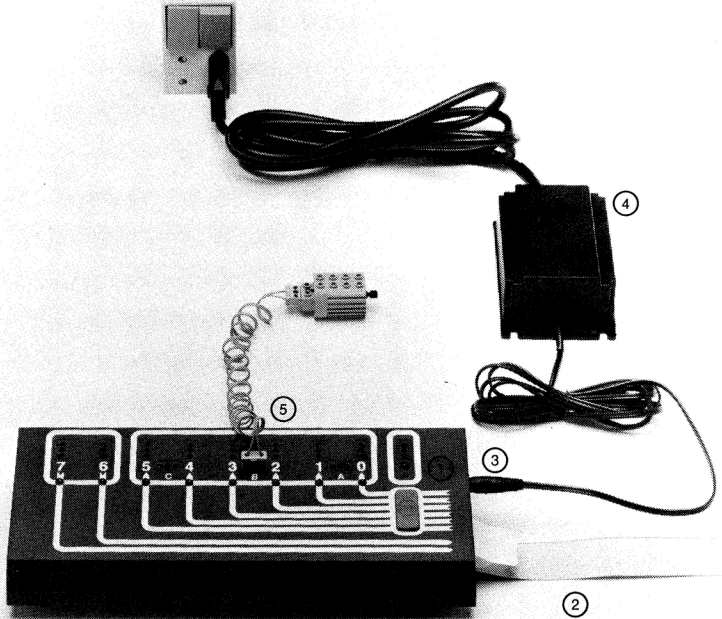
BITON:   GOSUB 11020
BITOFF:  GOSUB 12020
GETBIT:  GOSUB 13020
WAIT:    GOSUB 14020

```

# Foutenwijzer

## Beknopte foutenwijzer

Meestal is onderstaande, beknopte controlelijst voldoende voor het oplossen van eventuele problemen met de interface en/of de modellen.



1. De rode stopknop op de interface moet uit (d.w.z. stopknop omhoog!) zijn.
2. De computerkabel moet op de juiste manier zijn aangesloten (zowel op de interface als op de computer) zie "Aansluiten op de computer".
3. De transformatorstekker moet op de juiste manier op de interface zijn aangesloten.
4. De transformator moet aangesloten zijn op het lichtnet (de rode lamp boven de stopknop moet branden).
5. Werken de uitgangen 0-5? (Kontroleer met behulp van een lichtsteen).

Kontroleer bovendien of het model goed is aangesloten en of de snoeren onbeschadigt zijn.

## **Uitgebreide foutenwijzer**

Hieronder volgt een wat uitgebreidere controlelijst.

De controlelijst omvat:

1. LEGO Interface A en aansluiting op het lichtnet
  - 1.1. Stroom van een transformator
  - 1.2. Uitgangen met konstante stroom
  - 1.3. Ingangen
2. LEGO Interface A, Kommunikatie en functie
  - 2.1. Opstart
  - 2.2. Uitgangen
  - 2.3. Ingangen

### **CONTROLE PROCEDURES:**

#### **1.0. Controleer of er niets beschadigd is**

##### **1.1. Stroom van de transformator**

LET OP: Er mag niets aangesloten zijn op de uitgang met konstante stroom.

DOE	TEST	ZO JA	ZO NEE
1. Sluit de transformator aan op het lichtnet			
2. Sluit de transformator aan op Interface A	2a. Brandt het rode lampje boven de stopknop bij de konstante spanningsuitgang?	Transformator OK	Kontroleer of er stroom staat op het stopkontakt Test de transformator met een andere interface Test de interface met een andere transformator

##### **1.2. Uitgang met konstante stroom**

DOE	TEST	ZO JA	ZO NEE
1. Sluit een LEGO lichtsteen aan op de uitgang met konstante stroom	1a. Brandt de lichtsteen?	De uitgang met konstante stroom is OK	Test de uitgang met konstante stroom met een andere lichtsteen Probeer een ander LEGO snoer

### 1.3. Ingangen

De test moet voor elk van de ingangen 6 en 7 afzonderlijk worden uitgevoerd.

DOE	TEST	ZO JA	ZO NEE
1. Sluit de optosensor aan op de ingang			
2. Beweeg een gele LEGO steen langs de sensoropening op de sensor	2a. Knippert het groene lampje bij de ingang op de maat van de beweging?	Ingang en optosensor zijn OK	Probeer een ander LEGO snoer
			Test de ingang met een andere optosensor
			Test de ingang met een lichtsteen (zie 3)
3. Verwijder de optosensor			
4. Sluit een lichtsteen aan op de ingang (om er zeker van te zijn dat de lichtsteen OK is, test je hem eerst, op de uitgang met konstante stroom)	4a. Brandt het groene lampje als er een lichtsteen is aangesloten?	Vervolg de test met 4b	Ingang is defect
	4b. Is het groene lampje uit als er geen lichtsteen op aangesloten is?	Ingang is OK	Ingang is defect

## 2.1. Opstarten

DOE	TEST	ZO JA	ZO NEE
1. Volg de aanwijzingen van de fabrikant bij het aansluiten van de computer			
2. Let op dat de computer uit is			
3. Sluit de LEGO Interface A op de transformator aan en vervolgens op de computer. De stopknop moet uit zijn			
4. Sluit de transformator aan op het lichtnet			
5. Zet de transformator en de computer aan  Attentie: Eerst randapparatuur aansluiten; dan de computer!	5a. Branden alle lampjes bij de uitgangen?	Lampjes van de uitgangen zijn OK	Kontroleer of er spanning opstaat
			Kontroleer of de stopknop op de interface uit is
			Kontroleer of alles goed is aangesloten
			Probeer een andere interface
			Probeer een andere interfacekabel
6. Toets in: POKE 56579,63 POKE 56577,0	6a. Gaan alle uitgangen uit?	Ga door naar hoofdstuk 2.2: uitgangen	Probeer een andere interfacekabel
			Probeer een andere interface
			Probeer een andere computerinstallatie

## 2.2. Uitgangen

DOE	TEST	ZO JA	ZO NEE
1. Voer de opstart procedure van 2.1. uit			
2. Toets in: POKE 56577,41	2a. Branden de lampjes van uitgang 0, 3 en 5?	Kommunikatie met de uitgangen is OK	Probeer een andere interfacekabel
Toets in: POKE 56577,22	Branden de lampjes van uitgang 1, 2 en 4?		Probeer een andere interface
			Probeer een andere computerinstallatie
3. Sluit de LEGO 4,5 V DC motor aan op de uitgang A, B en C. De uitgangen A, B en C worden afzonderlijk getest			
4. Toets in: POKE 56577,42	4a. Start de motor?	Vervolg de test	Probeer een ander LEGO snoer
			Probeer een andere motor
			Uitgang is defect
5. Toets in: POKE 56577,21	5a. Verandert de draairichting van de motor?	De uitgangen zijn OK	Uitgang is defect
6. Toets in: POKE 56577,0 De motor stopt			

### 2.3. Ingangen

DOE	TEST	ZO JA	ZO NEE
1. Voer de opstart procedure 2.1. uit			
2. Voer de test van de ingangen 1.3. uit			
3. Sluit een gele lichtsteen aan op ingang 6 en een gele lichtsteen op ingang 7  Toets in: PRINT PEEK (56577) AND 192	3a. Verschijnt het getal 192 op het scherm?	Vervolg de test	Test de ingang met een ander LEGO snoer
			Probeer een andere interfacekabel
			Test de ingang met een andere interface
			Probeer een andere computerinstallatie
4. Verwijder de lichtstenen van ingang 6 en 7  Toets in: PRINT PEEK (56577) AND 192	4a. Verschijnt het getal 0 op het scherm?	Kabelingangen zijn OK	Probeer een andere interfacekabel
			Test de ingang met een andere interface
			Probeer een andere computerinstallatie

# Istruzioni per l'uso Cavo Commodore 64

Questa confezione LEGO® Technic Control contiene un cavo di collegamento al computer e le istruzioni per l'uso.

Per essere sicuri che l'userete correttamente vi spieghiamo le operazioni da eseguire:

1. Collegamento al computer
2. Programmazione del computer per il comando dell'Interfaccia A LEGO

## Ricerca errori

Se sorgessero dei problemi con l'interfaccia ed i modellini leggete il capitolo "Ricerca errori".

## Collegamento al computer

Il collegamento al computer si esegue nel modo seguente:

1. Spegnete il computer (per evitare che venga danneggiato)
2. Collegate il cavo al computer (vedi fig. 1)
3. Collegate il cavo alla cassetta dell'interfaccia (vedi fig. 2)
4. Collegate il filo del trasformatore alla cassetta dell'interfaccia (vedi fig. 2)
5. Accendete il trasformatore
6. Accendete il computer

*Controllate che il bottone rosso di arresto non sia premuto.*

## Programmazione del computer per il comando dell'Interfaccia A LEGO

Qui di seguito vi spieghiamo:

1. Come inizializzare il collegamento all'Interfaccia A LEGO
2. Come programmare gli exit 0-5
3. Come leggere gli entry 6 e 7

Impostate le linee di programma dell'allegato A (p. 73-74)

Non è necessario impostare le frasi REM del programma in BASIC dato che non sono da eseguire.

Mettete eventualmente il programma in memoria su dischetto o nastro. Se possibile stampatene una copia e controllate se avete eventualmente impostato dati sbagliati.

Il programma consiste nelle 5 procedure/subroutine seguenti:

1. init
2. biton
3. bitoff
4. getbit
5. wait

La procedura init inizializza il collegamento all'interfaccia. Non richiede alcun parametro. Bisogna scrivere questa procedura nella prima linea di tutti i programmi.

La procedura biton accende un exit sull'interfaccia. Il numero dell'exit viene usato come parametro. Nel programma in BASIC la variabile con cui viene passato il parametro si chiama NUM%. Il parametro deve essere compreso tra 0 e 5.

La procedura bitoff spegne un exit sull'interfaccia. Per il numero dell'exit vale lo stesso che per il biton.

La funzione/subroutine getbit legge lo stato su un entry. Il numero dell'entry viene usato come parametro. Nel programma in BASIC la variabile con cui viene passato il parametro si chiama NUM%. Lo stato viene indicato tramite la variabile Y%. Il numero dell'entry deve essere 6 oppure 7.

La procedura wait attende un dato numero di secondi. Il numero dei secondi viene indicato come parametro. Nel programma in BASIC il parametro viene indicato per mezzo della variabile TIM%. Il parametro indicato deve essere positivo e minore di 32768.

### Qui di seguito vi spieghiamo l'uso del programma:

Inizializzazione del collegamento all'Interfaccia A LEGO viene fatta impostando il seguente esempio:

<b>Esempio 1A, CBM 64 BASIC</b>	<b>Commenti/spiegazioni</b>
10 GOSUB 10000: REM INIT 20 END	Inizializza entry e exit

<b>Esempio 1B, COMAL 80</b>	<b>Commenti/spiegazioni</b>
0010 init 0020 END	Inizializza entry e exit

Impostate "RUN" e notate che la luce degli exit 0-5 si spegne. Ora il computer è pronto per usare l'interfaccia. Se compaiono eventuali messaggi di errore controllate il programma impostato confrontandolo con l'allegato A.

### Programmazione degli exit 0-5

Impostate il seguente esempio:

<b>Esempio 2A, CBM 64 BASIC</b>	<b>Commenti/spiegazioni</b>
10 GOSUB 10000: REM INIT	Inizializza input e output
20 NUM%=3	} Accende l'output 3
30 GOSUB 11000: REM BITON	
40 TIM%=4	} Attende 4 secondi
50 GOSUB 14000: REM WAIT	
60 NUM%=4	} Accende l'output 4
70 GOSUB 11000: REM BITON	
80 TIM%=2	} Attende 2 secondi
90 GOSUB 14000: REM WAIT	
100 NUM%=3	} Spegne l'output 3
110 GOSUB 12000: REM BITOFF	
120 TIM%=2	} Attende 2 secondi
130 GOSUB 14000: REM WAIT	
140 NUM%=4	} Spegne l'output 4
150 GOSUB 12000: REM BITOFF	
160 END	

<b>Esempio 2B, COMAL 80</b>	<b>Commenti/spiegazioni</b>
0010 init	Inizializza input e output
0020 biton (3)	Accende l'output 3
0030 wait (4)	Attende 4 secondi
0040 biton (4)	Accende l'output 4
0050 wait (2)	Attende 2 secondi
0060 bitoff (3)	Spegne l'output 3
0070 wait (2)	Attende 2 secondi
0080 bitoff (4)	Spegne l'output 4
0090 END	

Se si dovessero verificare degli errori controllate i dati impostati con le procedure/sub-routine dell'allegato A.  
 Notate che "biton"/"bitoff" possono accendere/spegnere un singolo exit senza influenzare lo stato degli altri. Provate ad accendere un exit, attendere, accendere un altro exit, attendere, spegnere il primo exit, attendere e spegnere l'altro exit.

**Lettura degli entry 6 e 7**

Gli entry 6 e 7 vengono usati per ricevere segnali dall'optosensor LEGO (rilevatore ottico). Collegate un optosensor all'entry 6.  
 Fate passare un mattoncino giallo LEGO davanti all'apertura dell'optosensor. Notate che la spia luminosa verde dell'entry lampeggia allo stesso ritmo di movimento del mattoncino. Quando la spia verde è accesa il computer legge il valore 1; quando la spia è spenta legge il valore 0.  
 Adesso proviamo a fare un programma che scriva 100 volte lo stato dall'entry 6.

Impostate il seguente esempio 3:

Esempio 3A, CBM 64 BASIC	Commenti/spiegazioni
<pre>10 GOSUB 10000: REM INIT 20 NUM%=6 30 NO%=100 40 FOR X=1 TO NO% 50 GOSUB 13000: REM GETBIT 60 PRINT Y% 70 NEXT X 80 END</pre>	<p>Inizializza entry e exit            Numero dell'entry            Numero ripetizioni</p> <p>Prelevare lo stato sull'entry            Scrivere lo stato</p>

Esempio 3B, COMAL 80	Commenti/spiegazioni
<pre>0010 init 0020 numero:=100 0030 FOR x:=1 TO numero DO 0040   PRINT getbit (6) 0050 ENDFOR x 0060 END</pre>	<p>Inizializza entry e exit            Numero ripetizioni</p> <p>Scrive lo stato sull'entry 6</p>

Se si dovessero verificare degli errori controllate i dati impostati con le procedure/sub-routine dell'allegato A.

Eseguite il programma e ricordate di attivare l'optosensor. Provate a collegare un optosensor all'entry 7 ed a scrivere lo stato.

Costruite un modello con motore, optosensor e disco contatore, per esempio una grande ruota. Collegate il motore all'exit 1 e l'optosensor all'entry 6.

*Il lato del disco contatore con la suddivisione grossa deve essere volto verso l'apertura dell'optosensor.*

Nell'esempio seguente l'optosensor viene usato per contare.

*Il programma conta 24 variazioni dell'entry 6.*

Nell'esempio 4 il programma in BASIC conta solo fino a 3 variazioni al secondo. Se si vuole contare più in fretta bisogna usare una subroutine scritta in linguaggio macchina.

**Esempio 4A, CBM 64 BASIC****Commenti/spiegazioni**

```

10 GOSUB 10000: REM INIT
20 NO%=24
30 NUM%=6
40 GOSUB 13000: REM GETBIT
50 QST%=Y%
60 NUM%=1
70 GOSUB 11000: REM BITON
80 NUM%=6
90 FOR QI=1 TO NO%
100 GOSUB 13000: REM GETBIT
110 IF Y%=QST% THEN GOTO 100
120 QST%=Y%
130 NEXT QI
140 NUM%=1
150 GOSUB 12000: REM BITOFF
160 END

```

```

Inizializza entry e exit
Numero variazioni
} Preleva e memorizza lo
} stato sull'entry 6

} Accende l'exit 1
Numero dell'entry

} Attende finché l'entry non
} cambia stato
Memorizzare il nuovo stato

} Spegne l'exit 1

```

**Esempio 4B, COMAL 80****Commenti/spiegazioni**

```

0010 init
0020 biton (1)
0030 no:=24
0040 FOR qi:=1 TO no DO
0050   qst:=getbit (6)
0060   REPEAT
0070     UNTIL getbit(6)<<>qst
0080 ENDFOR qi
0090 bitoff(1)
0100 END

```

```

Inizializza entry e exit
Accende l'exit 1
Numero variazioni

Si memorizza stato entry 6
} Attendere finché l'entry 6
} non cambia stato

Spegne l'exit 1

```

Se si dovessero verificare degli errori controllate i dati impostati con le procedure/sub-routine dell'allegato A.

Fate un programma dove l'optosensor viene collegato all'entry 7 e deve contare 12 variazioni.

Il programma in BASIC può essere fatto un po' più in fretta

saltando il parametro di controllo. In questo caso la

numerazione delle linee del comando GOSUB deve essere cambiata

in questo modo:

```

BITON:    GOSUB 11020
BITOFF:   GOSUB 12020
GETBIT:   GOSUB 13020
WAIT:     GOSUB 14020

```



## Istruzioni più ampie di ricerca errori

Qui di seguito vi diamo una lista di controllo più ampia.

Questa lista comprende:

1. Interfaccia A LEGO e collegamento con la presa di corrente

1.1. Corrente elettrica dal trasformatore

1.2. Exit costante

1.3. Entry

2. Interfaccia A LEGO, comunicazione e funzione

2.1. Avvio

2.2. Exit

2.3. Entry

## PROCEDURE DI CONTROLLO:

### 1.0. Controllo dei difetti visibili

#### 1.1. Corrente elettrica dal trasformatore

ATTENZIONE: Non deve essere collegato niente all'exit costante

FARE	CONTROLLARE	IN CASO AFFERMATIVO	IN CASO NEGATIVO
1. Collegate il trasformatore alla corrente			
2. Collegate il trasformatore all'Interfaccia A LEGO	2a. Si accende la spia luminosa dell'exit costante sopra il bottone di arresto?	Il trasformatore è in ordine	Verificate che ci sia la corrente Provate il trasformatore con un'altra interfaccia Provate l'interfaccia con un altro trasformatore

#### 1.2. Output costante

FARE	CONTROLLARE	IN CASO AFFERMATIVO	IN CASO NEGATIVO
1. Collegate 1 mattoncino luminoso all'output costante	1a. Si illumina?	L'output costante è in ordine	Provate l'output con un altro mattoncino luminoso Provato con un altro cavo LEGO

### 1.3. Entry

Gli entry 6 e 7 devono essere controllati uno alla volta.

FARE	CONTROLLARE	INCASOAFFERMATIVO	IN CASO NEGATIVO
1. Collegate l'optosensor all'entry			
2. Fate passare un mattoncino giallo LEGO davanti all'apertura dell'optosensor	2a. La spia luminosa verde dell'entry lampeggia allo stesso ritmo del movimento?	L'entry e l'optosensor sono in ordine	Provate con un altro filo LEGO
			Provate l'entry con un altro optosensor
			Provato l'entry con il mattoncino luminoso (eseguite il punto 3 ed altri)
3. Togliete l'optosensor			
4. Collegate un mattoncino luminoso all'entry (per essere sicuri che il mattoncino è in ordine controllatelo prima con l'exit costante)	4a. Si accende la spia verde dell'entry quando il mattoncino luminoso è collegato?	Continuate il controllo con 4b	L'entry è difettoso
	4b. La spia verde è spenta quando non è collegato niente?	L'entry è in ordine	L'entry è difettoso

## 2.1. Avvio

FARE	CONTROLLARE	INCASO Affermativo	IN CASO NEGATIVO
1. Installate e collegate il computer seguendo le istruzioni del fabbricante			
2. Assicuratevi che il computer sia spento			
3. Collegate l'Interfaccia A LEGO al computer e al trasformatore. Il bottone di arresto non deve essere premuto			
4. Collegate il trasformatore alla corrente			
5. Accendete il computer e il trasformatore	5a. Tutte le spie luminose degli exit sono accese?	Le spie luminose degli exit sono in ordine	Controllate se la c'è la tensione
			Controllate che il bottone di arresto dell'interfaccia non sia premuto
			Controllate che tutti gli allacciamenti siano corretti
			Provate con un'altra interfaccia
			Provate con un altro cavo per interfaccia
6. Impostate: POKE 56579,63 POKE 56577,0	6a. Si spengono tutti gli exit?	Proseguite col punto 2.2.: exit	Provate con un altro cavo per interfaccia
			Provate con un'altra interfaccia
			Provate con un altro computer

## 2.2. Exit

FARE	CONTROLLARE	INCASOAFFERMATIVO	IN CASO NEGATIVO
1. Eseguite la procedura di avvio 2.1.			
2. Impostate: POKE 56577,41	2a. Le spie luminose degli exit 0, 3 e 5 sono accese?	La comunicazione con gli exit è in ordine	Provate con un altro cavo per interfaccia
Impostate: POKE 56577,22	Le spie luminose degli exit 1, 2 e 4 sono accese?		Provate con un'altra interfaccia
			Provate con un altro computer
3. Collegate il motore LEGO DC 4,5 V agli exit A, B, C. Gli exit A, B, C devono essere controllati una alla volta			
4. Impostate: POKE 56577,42	4a. Si avvia il motore?	Continuate il controllo	Provate con un altro filo LEGO
			Provate con un altro motore
			L'exit è difettoso
5. Impostate: POKE 56577,21	5a. Il motore inverte il senso di marcia?	Gli exit sono in ordine	L'exit è difettoso
6. Impostate: POKE 56577,0 Il motore si arresta			

### 2.3. Entry

FARE	CONTROLLARE	INCASO AFFERMATIVO	IN CASO NEGATIVO
1. Eseguite la procedura di avvio 2.1.			
2. Eseguite il controllo 3.1. degli entry			
3. Collegate un mattoncino giallo luminoso all'entry 6 ed un mattoncino giallo luminoso all'entry 7  Impostate: PRINT PEEK (56577) AND 192	3a. Appare il numero 192 sullo schermo?	Proseguite il controllo	Provate l'entry con un altro filo LEGO
			Provate con un altro cavo per interfaccia
			Provate l'entry con un'altra interfaccia
			Provate con un altro computer
4. Togliete i mattoncini luminosi dagli entry 6 e 7  Impostate: PRINT PEEK (56577) AND 192	4a. Appare il numero 0 sullo schermo?	Il cavo e gli entry sono in ordine	Provate con un altro cavo per interfaccia
			Provate l'entry con un'altra interfaccia
			Provate con un altro computer

# Bruksanvisning Commodore 64 kabel

Detta LEGO® Technic Control set innehåller en datorkabel och en bruksanvisning. För att säkerställa korrekt användning genomgås nedan:

1. Anslutning till dator
2. Programmering av datorn för styrning av LEGO Interface A

## Felsökning

Skulle problem uppstå rörande interface och modeller, se avsnittet med "Felsökning".

## Anslutning till dator

Anslutning till datorn görs på följande sätt:

1. Stäng datorn (för att undvika eventuella skador)
2. Anslut kabeln till datorns användar-port (jfr. ill. 1)
3. Anslut kabeln till interface-boxen (jfr. ill. 2)
4. Anslut transformatorledningen till interface-boxen (jfr. ill. 2)
5. Anslut transformatorn till nätet.
6. Tänd datorn

*Kontrollera, att den röda stoppknappen är ute.*

## Programmering av datorn för styrning av LEGO Interface A

I det följande genomgås:

1. Initialisering av förbindelsen till LEGO Interface A
2. Programmering av utgångarna 0-5
3. Avläsning av ingångarna 6 och 7

Mata in programraderna i bilaga A (p. 73-74).

Man behöver inte mata in REM-meningarna i BASIC-programmet, de har ingen funktionell betydelse.

Lagra eventuellt programmet på flexskiva eller band. Gör om möjligt en utskrift och kontrollera ev. inmatningsfel. Programmet består av följande 5 procedurer/subrutiner:

1. init
2. biton
3. bitoff
4. getbit
5. wait

Proceduren init initialiserar förbindelsen till interfacet. Den kräver inte några parametrar.

Proceduren biton tänder en utgång på interfacet. Utgångsnumret överförs som parameter.

Proceduren biton tänder en utgång på interfacet. Utgångsnumret överförs som parameter.

I BASIC-programmet heter den variabel, som överför parametern NUM%. Parametern skall ligga mellan 0 och 5.

Proceduren bitoff släcker en utgång på interfacet. För utgångsnumret gäller det samma, som vid biton.

Funktionen/subrutinen getbit läser status på en ingång.

Ingångsnumret överförs som parameter. I BASIC-programmet heter den variabel, som överför parametern NUM%. Status returneras i variabeln Y%. Ingångsnumret skall vara 6 eller 7.

Proceduren wait väntar ett antal sekunder. Antalet överförs som parameter. I BASIC-programmet överförs parametern med hjälp av variabeln TIM%. Den överförda parametern skall vara positiv och mindre än 32768.

### Programmets användning framgår av följande:

Initialisering av förbindelsen till LEGO Interface A görs genom att följande exempel matas in:

Exempel 1A, CBM 64 BASIC	Kommentarer/förklaring
10 GOSUB 10000: REM INIT 20 END	Initialiserar in- och utgångar

Exempel 1B, COMAL 80	Kommentarer/förklaring
0010 init 0020 END	Initialiserar in- och utgångar

Mata in "RUN" och lägg märke till att ljuset på utgångarna 0-5 släcks. Datorn är nu klar att använda interfacet.

Vid eventuella felmeddelanden kontrolleras det inmatade programmet i bilaga A.

### Programmering av utgångarna 0-5

Mata in följande exempel

Exempel 2A, CBM 64 BASIC	Kommentarer/förklaring
10 GOSUB 10000: REM INIT 20 NUM%=3 30 GOSUB 11000: REM BITON 40 TIM%=4 50 GOSUB 14000: REM WAIT 60 NUM%=4 70 GOSUB 11000: REM BITON 80 TIM%=2 90 GOSUB 14000: REM WAIT 100 NUM%=3 110 GOSUB 12000: REM BITOFF 120 TIM%=2 130 GOSUB 14000: REM WAIT 140 NUM%=4 150 GOSUB 12000: REM BITOFF 160 END	Initialiserar in- och utgångar } Tänder utgång 3 } Väntar 4 sekunder } Tänder utgång 4 } Väntar 2 sekunder } Släcker utgång 3 } Väntar 2 sekunder } Släcker utgång 4

Exempel 2B, COMAL 80	Kommentarer/förklaring
0010 init 0020 biton (3) 0030 wait (4) 0040 biton (4) 0050 wait (2) 0060 bitoff (3) 0070 wait (2) 0080 bitoff (4) 0090 END	Initialiserar in- och utgångar Tänder utgång3 Väntar 4 sekunder Tänder utgång 4 Väntar 2 sekunder Släcker utgång 3 Väntar 2 sekunder Släcker utgång 4

Om fel uppstår, kontrolleras det inmatade exemplet och procedurerna/subrutinerna i bilaga A.

Lägg märke till att ”biton”/”bitoff” kan tända/släcka en enstaka utgång, utan att påverka status på de andra. Prova med att tända en utgång, vänta, tända en annan utgång, vänta, släcka den första utgången, vänta, släcka den sista utgången.

### Avläsning av ingångarna 6 och 7

Ingångarna 6 och 7 kan användas till att motta signaler från LEGO optosensor (optisk avsökare). Anslut en sensor till ingång 6.

För ett gult LEGO element förbi sensorns öppning. Lägg märke till, att den gröna lampan vid ingången blinkar i takt med rörelsen. När den gröna lampan lyser, avläser datorn värdet 1; när lampan är släckt är värdet 0.

Vi skall nu göra ett program, som skriver ut status 100 gånger från ingång 6.

Mata in exempel 3:

#### Exempel 3A, CBM 64 BASIC

```
10 GOSUB 10000: REM INIT
20 NUM%=6
30 NO%=100
40 FOR X=1 TO NO%
50 GOSUB 13000: REM GETBIT
60 PRINT Y%
70 NEXT X
80 END
```

#### Kommentarer/förklaring

Initialiserar in- och utgångar  
Ingångens nummer  
Antal växlingar  
  
Hämta status på ingången  
Skriv ut status

#### Exempel 3B, COMAL 80

```
0010 init
0020 antal:=100
0030 FOR x:=1 TO antal DO
0040 PRINT getbit (6)
0050 ENDFOR x
0060 END
```

#### Kommentarer/förklaring

Initialiserar in- och utgångar  
Antal växlingar  
  
Skriver ut status på  
ingång 6

Om fel uppstår, kontrolleras det inmatade exemplet och procedurerna/subrutinerna i bilaga A.

Kör programmet och kom ihåg att aktivera sensorn.

Prova med att ansluta en sensor till ingång 7 och skriva ut status.

Bygg en modell med motor, optosensor och räkneskiva, t.ex. ett pariserhjul. Anslut motor till utgång 1 och sensorn till ingång 6.

*Räkneskivans grova indelning skall vända in mot sensoröppningen.*

I följande exempel används sensorn till att räkna med.

*Programmet räknar 24 växlingar på ingång 6.*

I exempel 4 kan BASIC-programmet endast räkna upp till 3 växlingar per sekund. Skall man räkna fortare, är man tvungen att använda ett underprogram, som är skrivet i maskinkod.

Exempel 4A, CBM 64 BASIC	Kommentarer/förklaring
10 GOSUB 10000: REM INIT	Initialiserar in- och utgångar
20 NO%=24	
30 NUM%=6	} Hämtar och lagrar status på ingång 6
40 GOSUB 13000: REM GETBIT	
50 QST%=Y%	} Tänder utgång 1
60 NUM%=1	
70 GOSUB 11000: REM BITON	} Ingångens nummer
80 NUM%=6	
90 FOR QI=1 TO NO%	} Väntar till ingången byter status
100 GOSUB 13000: REM GETBIT	
110 IF Y%=QST% THEN GOTO 100	} Lagra ny status
120 QST%=Y%	
130 NEXT QI	} Släcker utgång 1
140 NUM%=1	
150 GOSUB 12000: REM BITOFF	
160 END	

Exempel 4B, COMAL 80	Kommentarer/förklaring
0010 init	Initialiserar in- och utgångar
0020 biton (1)	
0030 no:=24	} Tänder utgång 1
0040 FOR qi:=1 TO no DO	
0050 qst:=getbit (6)	} Status på ingång 6 lagras
0060 REPEAT	
0070 UNTIL getbit(6)<>qst	} Vänta till ingång 6 byter status
0080 ENDFOR qi	
0090 bitoff(1)	} Släcker utgång 1
0100 END	

Om fel uppstår, kontrolleras det inmatade exemplet och procedurerna/subrutinerna i bilaga A.

Gör ett program, där optosensorn ansluts till ingång 7 och skall räkna 12 växlingar.

BASIC-programmet kan göras lite snabbare genom att man hoppar över parameterkontrollen. I så fall skall radnumren i GOSUB-kommandot göras om till:

```

BITON:   GOSUB 11020
BITOFF:  GOSUB 12020
GETBIT:  GOSUB 13020
WAIT:    GOSUB 14020

```



## Utvidgad felsökningsvägledning

En mera utförlig kontrolllista följer nedan.

Kontrollistan omfattar:

1. LEGO Interface A och nätanslutning
  - 1.1. Ström från transformatorn
  - 1.2. Konstantspänningsutgång
  - 1.3. Ingångar
2. LEGO Interface A, kommunikation och funktion
  - 2.1. Uppstart
  - 2.2. Utgångar
  - 2.3. Ingångar

### KONTROLLPROCEDURER:

#### 1.0. Kontrollera ev. visuella defekter

##### 1.1. Ström från transformatorn

OBS: Ingenting får vara anslutet till konstantspänning.

UTFÖR	TEST	OM JA	OM NEJ
1. Nätanslut transformatorn			
2. Anslut transformatorn till Interface A	2a. Lyser röd ljusdiod vid konstantspänningsutgången över stoppknappen?	Transformator OK	Undersök om det finns nätspänning
			Prova transformatorn med annat interface
			Prova interfacet med annan transformator

##### 1.2. Konstantspänningsutgång

UTFÖR	TEST	OM JA	OM NEJ
1. Anslut 1 st. ljussten till konstantspänningsutgången	1a. Lyser ljusstenen?	Konstantspänningsutgången OK	Prova konstantspänningsutgången med en annan ljussten
			Prova med annan LEGO ledning

### 1.3. Ingångar

Ingångarna 6 och 7 testas var för sig.

UTFÖR	TEST	OM JA	OM NEJ
1. Anslut optosen-sorn till ingången			
2. För ett gult LEGO element förbi sensoröppningen	2a. Blinkar den gröna ljusdioden vid ingången i takt med rörelsen?	Ingången och sensorn er OK	Prova med en annan LEGO ledning
			Prova ingången med en annan optosensor
			Prova ingången med ljussten (Utför 3 m.fl.)
3. Tag bort sensorn			
4. Anslut ljusstenen till ingången (för att säkra att ljusstenen är OK, provas den först med konstantspänningsutgång)	4a. Lyser den gröna ljusdioden vid ingången, när ljusstenen är ansluten?	Fortsätt testen med 4b	Ingången er defekt
	4b. Är den gröna ljusdioden släckt, när ingenting är anslutet?	Ingången er OK	Ingången er defekt

## 2.1. Upstart

UTFÖR	TEST	OM JA	OM NEJ
1. Ställ upp, förbind och anslut datorinstallationen enligt fabrikantens anvisningar			
2. Sörj för att datorn är släckt			
3. Anslut LEGO Interface A till dator och transformator. Stoppknappen skall vara ute			
4. Nätanslut transformatorn			
5. Tänd datorinstallationen och transformatorn	5a. Lyser samtliga ljusdioder vid utgångarna?	Ljusdioderne på utgångarna är OK	Undersök om det finns nätspänning
			Undersök om stoppknappen på interface är ute
			Undersök om alla anslutningar är korrekt utförda
			Prova med annat interface
			Prova med annan interfacekabel
			Prova med annan datorinstallation
6. Mata in: POKE 56579,63 POKE 56577,0	6a. Släcks samtliga utgångar?	Gå vidare till avsnitt 2.2: utgångar	Prova med annan interfacekabel
			Prova med annat interface
			Prova med annan datorinstallation

## 2.2. Utgångar

UTFÖR	TEST	OM JA	OM NEJ
1. Genomför 2.1. uppstartprocedur			
2. Mata in: POKE 56577,41	2a. Lyser ljusdioder- na vid utgång 0, 3 och 5?	Kommunikation till utgångarna OK	Prova med annan in- terfacekabel
Mata in: POKE 56577,22	Lyser ljusdioder- na vid utgång 1, 2 och 4?		Prova med annat in- terface
			Prova med annan datorinstallation
3. Anslut LEGO 4,5 V DC motor till utgång A, B och C Utgångarna A, B, C testas var för sig			
4. Mata in: POKE 56577,42	4a. Startar motorn?	Fortsätt testen	Prova med annan LEGO ledning
			Prova med annan motor
			Utgången är defekt
5. Mata in: POKE 56577,21	5a. Ändrar motorn varvriktning?	Utgångarna är OK	Utgången är defekt
6. Mata in: POKE 56577,0 Motorn stannar			

### 2.3. Ingångar

UTFÖR	TEST	OM JA	OM NEJ
1. Genomför 2.1. uppstartprocedur			
2. Genomför 1.3. test av ingångarna			
3. Anslut en gul ljusten till ingång 6 och en gul ljusten till ingång 7  Mata in: PRINT PEEK (56577) AND 192	3a. Skrivs talet 192 ut på skärmen?	Gå vidare i testen	Prova ingången med en annan LEGO ledning
			Prova med en annan interfacekabel
			Prova ingången med ett annat interface
			Prova med en annan datorinstallation
4. Tag bort ljustenarna från ingång 6 och 7  Mata in: PRINT PEEK (56577) AND 192	4a. Skrivs talet 0 ut på skärmen?	Kabel och ingångar är OK	Prova med en annan interfacekabel
			Prova ingången med ett annat interface
			Prova med en annan datorinstallation

# Gebrauchshinweise

Dieser LEGO® Technic Artikel enthält ein LEGO Interfacekabel zu den Computern Commodore 64 und 128 und Gebrauchshinweise.

Um eine korrekte Verwendung sicherzustellen, wird folgendes erläutert:

- A Anschließen des LEGO Interface A.
- B Hinweise zum direkten Ansprechen des LEGO Interface A in C64-BASIC und in COMAL 80.
- C Fehlersuche: bei Problemen mit dem Interface und den Modellen kann hier nachgesehen werden.

## **Anschließen des LEGO Interface A**

Das Interface wird folgendermaßen angeschlossen:

1. Der Computer wird ausgeschaltet, um mögliche Beschädigung zu vermeiden.
2. Das Interfacekabel wird an den user-port des Computers angeschlossen (vgl. Abb. 1).  
Abb. 1).
3. Das Interfacekabel wird an das Interface angeschlossen (vgl. Abb. 2).
4. Das Netzteil wird an das Interface angeschlossen (vgl. Abb. 2).
5. Das Netzteil wird an das Netz angeschlossen.
6. Der Computer wird eingeschaltet.

*Der rote Stop-Schalter auf dem Interface soll nicht gedrückt sein!*

## **Hinweise zum direkten Ansprechen des LEGO Interface A in C64-BASIC und in COMAL 80**

Vorgehensschritte:

1. Das Herstellen der Datenverbindung vom Computer zum LEGO Interface A
2. Die Programmierung der Ausgänge 0-5
3. Das Abfragen der Eingänge 6 und 7

## **Das Herstellen der Datenverbindung vom Computer zum LEGO Interface A:**

Zunächst müssen die Programmzeilen des Hilfsprogramms, s. Anhang A (Seite 73+74), in den Computer eingegeben werden.

Für den weiteren Gebrauch ist es sinnvoll, das Hilfsprogramm auf einer Diskette oder einer Kassette zu sichern. Ein Ausdruck mit einem Drucker erleichtert die Suche nach eventuellen Eingabefehlern.

Das Hilfsprogramm besteht aus den folgenden fünf Unterprogrammen:

1. init
2. biton
3. bitoff
4. getbit
5. wait

Das Unterprogramm init stellt die Datenverbindung zum Interface her. Hier sind keine Parameter erforderlich. Dieses Unterprogramm muß in der ersten Zeile aller Programme zur Steuerung von Modellen aufgerufen werden.

Das Unterprogramm biton schaltet einen Ausgang auf dem Interface ein. Die Nummer des Ausgangs wird als Parameter übergeben. Im BASIC Programm heißt die Variable zur Übergabe des Parameters NUM%. Der Parameter muß zwischen 0 und 5 liegen.

Das Unterprogramm bitoff schaltet einen Ausgang auf dem Interface aus. Für die Nummer des Ausgangs gilt das gleiche wie bei biton.

Das Unterprogramm getbit liest den Zustand eines Eingangs ab. Die Nummer des Eingangs wird als Parameter übergeben. Sie muß 6 oder 7 sein. Im BASIC Programm heißt

die Variable zur Übergabe dieses Parameters NUM%. Der Zustand des Eingangs wird durch die Variable Y% wiedergegeben.

Das Unterprogramm wait wartet einige Sekunden. Die Zahl - die Wartezeit in Sekunden - wird als Parameter übergeben. Im BASIC Programm heißt die Variable zur Übergabe des Parameters TIM%. Der Parameter muß positiv und kleiner als 32768 sein.

Für ein einwandfreies Funktionieren der C64-BASIC Programme muß sorgfältig zwischen den Programmvariablen und den Variablen zur Interfacesteuerung wie NUM%, Y% und TIM% unterschieden werden.

### Programmbeispiele zum Herstellen der Datenverbindung vom Computer zum LEGO

#### Interface A:

Die Datenverbindung zum LEGO Interface A wird durch das Eingeben des Programmbeispiels in C64-BASIC bzw. COMAL 80 vorbereitet.

Beispiel C64 BASIC	Kommentar/Erklärung
10 GOSUB 10000: REM INIT  20 END	Herstellung der Datenverbindung
Beispiel COMAL 80	Kommentar/Erklärung
0010 init  0020 END	Herstellung der Datenverbindung

Mit der Tastenfolge "RUN" und der Taste <RETURN> werden die Programmbeispiele gestartet. Die Kontrolllampen an den Ausgängen 0-5 erlöschen. Jetzt ist der Computer für die Datenübertragung zum und vom Interface startbereit. Bei etwaigen Fehlern muß das eingegebene Programmbeispiel überprüft und eventuell die eingegebenen Unterprogramme mit dem Anhang A verglichen werden.

#### Die Programmierung der Ausgänge 0-5

Programmbeispiele in C64-BASIC bzw. in COMAL 80:

Beispiel C64 BASIC	Kommentar/Erklärung
10 GOSUB 10000: REM INIT	Herstellung der Datenverbindung
20 NUM%=3	} Schaltet den Ausgang 3 ein
30 GOSUB 11000: REM BITON	
40 TIM%=4	} Wartet 4 Sekunden
50 GOSUB 14000: REM WAIT	
60 NUM%=4	} Schaltet den Ausgang 4 ein
70 GOSUB 11000: REM BITON	
80 TIM%=2	} Wartet 2 Sekunden
90 GOSUB 14000: REM WAIT	
100 NUM%=3	} Schaltet den Ausgang 3 aus
110 GOSUB 12000: REM BITOFF	
120 TIM%=2	} Wartet 2 Sekunden
130 GOSUB 14000: REM WAIT	
140 NUM%=4	} Schaltet den Ausgang 4 aus
150 GOSUB 12000: REM BITOFF	
160 END	

**Beispiel COMAL 80**

```

0010 init
0020 biton (3)
0030 wait (4)
0040 biton (4)
0050 wait (2)
0060 bitoff (3)
0070 wait (2)
0080 bitoff (4)
0090 END

```

**Kommentar/Erklärung**

Herstellung der Datenverbindung  
 Schaltet den Ausgang 3 ein  
 Wartet 4 Sekunden  
 Schaltet den Ausgang 4 ein  
 Wartet 2 Sekunden  
 Schaltet den Ausgang 3 aus  
 Wartet 2 Sekunden  
 Schaltet den Ausgang 4 aus

Bei etwaigen Fehlern muß das eingegebene Programmbeispiel überprüft und eventuell die eingegebenen Unterprogramme mit dem Anhang A verglichen werden.  
 Wichtiger Hinweis: Die Unterprogramme "biton"/"bitoff" können Ausgänge einzeln ein-/ausschalten, ohne den Zustand der anderen Ausgänge zu beeinflussen.

Beispiel einer Anweisungsfolge:

Einen Ausgang einschalten, warten, einen anderen Ausgang einschalten, warten, den ersten Ausgang ausschalten, warten und den zweiten Ausgang ausschalten.

**Das Abfragen der Eingänge 6 und 7**

Mit den Eingängen 6 und 7 können Signale vom LEGO Optosensor aufgenommen werden.

Der Optosensor wird an Eingang 6 angeschlossen. Wenn ein gelber LEGO Stein dicht vor der ovalen Öffnung des Optosensors hin- und herbewegt wird, blinkt die grüne Kontrolllampe am Eingang gleichzeitig mit der Bewegung. Jedesmal, wenn die grüne Kontrolllampe aufleuchtet, zeigt der Computer bei einem geeigneten Programm den Wert 1 an, wenn die grüne Lampe erlischt, wird der Wert 0 angezeigt.

Ein Beispielprogramm in C64-BASIC bzw. COMAL 80, das 100 mal den Zustand des Eingangs 6 anzeigt:

**Beispiel C64 BASIC**

```

10 GOSUB 10000: REM INIT
20 NUM%=6
30 NO%=100
40 FOR X=1 TO NO%
50 GOSUB 13000: REM GETBIT
60 PRINT Y%
70 NEXT X
80 END

```

**Kommentar/Erklärung**

Herstellung der Datenverbindung  
 Nummer des Eingangs  
 Zahl der Durchläufe  
 Ermittelt den Zustand des Eingangs 6  
 Zeigt den Zustand an

**Beispiel COMAL 80**

```

0010 init
0020 zahl:=100
0030 FOR x:=1 TO zahl DO
0040 PRINT getbit (6)
0050 ENDFOR x
0060 END

```

**Kommentar/Erklärung**

Herstellung der Datenverbindung  
 Zahl der Durchläufe  
 Ermittelt und zeigt den Zustand des Eingangs 6 an

Bei etwaigen Fehlern muß das eingegebene Programmbeispiel überprüft und eventuell die eingegebenen Unterprogramme mit dem Anhang A verglichen werden. Während des Programmablaufs muß der Sensor 100 mal z.B. in der oben beschriebenen Weise aktiviert werden.

Ein Beispiel zu einem LEGO Modell:

Vorlage: Ein Modell mit einem Motor, Optosensor und einer Zählscheibe, z.B. ein Riesenrad. Der Motor wird an den Ausgang 1, und der Optosensor an den Eingang 6 angeschlossen.

Die Grobeinteilung der Zählscheibe soll zur ovalen Öffnung des Sensors hin gerichtet sein. Bei dem folgenden Beispiel wird der Optosensor zum Zählen verwendet.

Das Programm zählt 24 Wechsel der Hell-Dunkel-Segmente der Zählscheibe, also 24 Wechsel des Zustandes am Eingang 6.

In den folgenden Programmbeispielen in C64-BASIC bzw. COMAL 80 kann das BASIC Programm nur bis zu 3 Wechsel pro Sekunde zählen. Falls schneller gezählt werden soll, muß ein Unterprogramm benutzt werden, das im Maschinenkode geschrieben ist.

Beispiel C64 BASIC	Kommentar/Erklärung
10 GOSUB 10000: REM INIT	Herstellung der Datenverbindung Anzahl der Wechsel Ermittelt und speichert den Startzustand des Eingangs 6 Schaltet den Ausgang 1 ein Nummer des Eingangs Wartet, bis der Eingang 6 umgeschaltet wird Zwischenspeicherung des Zustandes Schaltet den Ausgang 1 aus
20 NO%=24	
30 NUM%=6	
40 GOSUB 13000: REM GETBIT	
50 QST%=Y%	
60 NUM%=1	
70 GOSUB 11000: REM BITON	
80 NUM%=6	
90 FOR QI=1 TO NO%	
100 GOSUB 13000: REM GETBIT	
110 IF Y%=QST% THEN GOTO 100	
120 QST%=Y%	
130 NEXT QI	
140 NUM%=1	
150 GOSUB 12000: REM BITOFF	
160 END	

Beispiel COMAL 80	Kommentar/Erklärung
0010 init	Herstellung der Datenverbindung Schaltet den Ausgang 1 ein Anzahl der Wechsel Zwischenspeicherung des Zustandes Wartet, bis Eingang 6 umschaltet Schaltet den Ausgang 1 aus
0020 biton (1)	
0030 no:=24	
0040 FOR qi:=1 TO no DO	
0050   qst:=getbit (6)	
0060   REPEAT	
0070     UNTIL getbit(6)<>qst	
0080 ENDFOR qi	
0090 bitoff(1)	
0100 END	

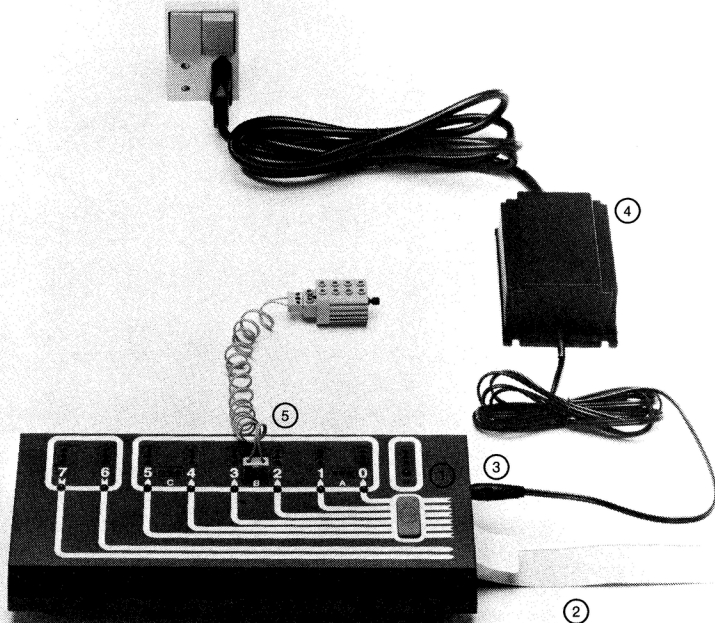
Bei etwaigen Fehlern muß das eingegebene Programmbeispiel überprüft und eventuell die eingegebenen Unterprogramme mit dem Anhang A verglichen werden. Das BASIC Programm kann etwas beschleunigt werden, indem die Parameterkontrolle übersprungen wird. Die Zeilennummern bei den GOSUB-Anweisungen müssen dann jeweils folgendermaßen geändert werden:

BITON: GOSUB 11020  
BITOFF: GOSUB 12020  
GETBIT: GOSUB 13020  
WAIT: GOSUB 14020

## Fehlersuche

Anleitung zur schnellen Fehlersuche

Bei Problemen mit dem Interface und den Modellen wird die folgende kurze Checkliste zum Erkennen von Fehlern in den meisten Fällen ausreichen:



1. Der rote Stop-Schalter auf dem Interface soll nicht gedrückt sein.
2. Das Interfacekabel muß korrekt am Interface und am Computer angebracht sein, vgl. "Anschließen des Interface".
3. Der Stecker des Netzteils muß korrekt am Interface angebracht sein.
4. Das Netzteil muß am Netz angeschlossen sein (die rote Kontrolllampe über dem Stop-Schalter muß leuchten).
5. Funktionieren die Ausgänge 0-5? (Überprüfung mit einem LEGO Leuchtstein)
6. Ist das Modell korrekt angeschlossen?
7. Sind alle Anschlußkabel in Ordnung?

### Erweiterte Anleitung zur Fehlersuche:

Nachfolgend eine ausführliche Checkliste. Sie umfaßt folgende Teile:

1. LEGO Interface A und Netzanschluß
  - 1.1. Test der Spannungsversorgung vom Netzteil
  - 1.2. Test des Permanentausganges des Interface
  - 1.3. Test der Eingänge des Interface
2. LEGO Interface A, Datenübertragung
  - 2.1. Start
  - 2.2. Schalten der Ausgänge
  - 2.3. Abfragen der Eingänge

### CHECKVERFAHREN:

#### 1.0 Untersuchung auf visuelle Defekte

##### 1.1 Test der Spannungsversorgung vom Netzteil

ACHTUNG: Bei diesem Test soll der Permanentausgang des Interface (über dem Stop-Schalter) nicht belegt sein.

AKTION	TEST	WENN JA	WENN NEIN
1. Anschluß des Netzteils an das Interface A			
2. Anschluß des Netzteils an das Netz	2a. Leuchtet die rote Kontrollampe über dem Stop-Schalter?	Das Netzteil ist in Ordnung	Ist das Netzteil mit dem Netz verbunden?
			Test des Netzteils mit einem anderen Interface
			Test des Interface mit einem anderen Netzteil

##### 1.2. Test des Permanentausganges des Interface

ACTION	TEST	WENN JA	WENN NEIN
1. Ein LEGO Leuchtstein wird an den Permanentausgang angeschlossen	1a. Leuchtet der Leuchtstein?	Der Permanentausgang ist in Ordnung	Test des Permanentausganges mit einem anderen Leuchtstein
			Test mit einem anderen LEGO Anschlußkabel

### 1.3. Test der Eingänge des Interface

Testverfahren für die Eingänge 6 und 7. Die Eingänge müssen einzeln getestet werden.

AKTION	TEST	WENN JA	WENN NEIN
1. Der Optosensor wird an den Eingang angeschlossen			
2. Ein gelber LEGO Stein wird dicht vor der ovalen Öffnung des Optosensors hin- und herbewegt	2a. Blinkt die grüne Eingangskontrollampe gleichzeitig mit der Hin- und Herbewegung?	Der Eingang und der Optosensor sind in Ordnung	Test mit einem anderen LEGO Anschlußkabel
			Test des Eingangs mit einem anderen Optosensor
			Test des Eingangs mit einem Leuchtstein (siehe Aktion 4)
3. Entferne den Optosensor			
4. Ein Leuchtstein wird an den Eingang angeschlossen. Der Leuchtstein muß in Ordnung sein. Dieses kann mit Hilfe des Permanentausganges getestet werden	4a. Leuchtet die grüne Eingangskontrollampe beim Anschluß des Leuchtsteins?	Weiter: Aktion 4b	Der Eingang ist defekt
	4b. Erlischt die grüne Eingangskontrollampe beim Entfernen des Leuchtsteins?	Der Eingang ist in Ordnung	Der Eingang ist defekt

## 2.1. Start

AKTION	TEST	WENN JA	WENN NEIN
1. Der Computer wird gemäß den Anleitungen des Herstellers aufgestellt und angeschlossen			
2. Der Computer muß abgeschaltet sein			
3. Das LEGO Interface A wird an den Computer angeschlossen. Das Netzteil wird an das Interface angeschlossen. Der Stop-Schalter auf dem Interface darf nicht gedrückt sein			
4. Das Netzteil wird an das Netz angeschlossen			
5. Der Computer wird eingeschaltet	5a. Leuchten sämtliche roten Kontrolllampen an den Ausgängen?	Die Kontrolllampen an den Ausgängen sind in Ordnung	Ist das Netzteil mit dem Netz verbunden?
			Test, ob der Stop-Schalter am Interface gedrückt ist. Er soll nicht gedrückt sein
			Test, ob alle Anschlüsse korrekt sind
			Wenn vorhanden: Test mit einem anderen Interface
			Test mit einem anderen Interfacekabel
Test mit einem anderen Computer			

6. Eingabe und Ausführung der Anweisungen: POKE 56579,63 POKE 56577,0	6a. Werden alle Ausgänge ausgeschaltet?	Weiter Abschnitt 2.2.: »Schalten der Ausgänge«	Wenn vorhanden: Test mit einem anderen Interfacekabel
			Test mit einem anderen Interface
			Test mit einem anderen Computer

## 2.2. Schalten der Ausgänge

AKTION	TEST	WENN JA	WENN NEIN
1. Startverfahren wie unter Abschnitt 2.1. »Start« beschrieben			
2. Eingabe und Ausführung der Anweisung: POKE 56577,41	2a. Leuchten die Kontrolllampen an den Ausgängen 0, 3 und 5 auf?	Datenübertragung zu den Ausgängen ist in Ordnung	Wenn vorhanden: Test mit einem anderen Interfacekabel
Eingabe und Ausführung der Anweisung: POKE 56577,22	Leuchten die Kontrolllampen an den Ausgängen 1, 2 und 4 auf?		Wenn vorhanden: Test mit einem anderen Interface
			Test mit einem anderen Computer
3. Die Ausgänge A, B, C werden jeder für sich mit einem LEGO 4,5 V DC Motor getestet			
4. Eingabe und Ausführung der Anweisung: POKE 56577,42	4a. Startet der Motor, wenn er mit den Ausgängen A, B oder C verbunden wird?	Weiter: Aktion 5	Test mit einem anderen LEGO Anschlußkabel
			Test mit einem anderen LEGO Motor
			Ausgang defekt
5. Eingabe und Ausführung der Anweisung: POKE 56577,21	5a. Ändert der Motor die Drehrichtung?	Die Ausgänge sind in Ordnung	Ausgang defekt
6. Eingabe und Ausführung der Anweisung: POKE 56577,0 Der Motor wird ausgeschaltet			

### 2.3. Abfrage der Eingänge

AKTION	TEST	WENN JA	WENN NEIN
1. Startverfahren wie unter Abschnitt 2.1. »Start« beschrieben			
2. Test der eingänge wie unter Abschnitt 1.3. »Test der Eingänge des Interface« beschrieben			
<p>3. Ein LEGO Leuchtstein wird an den Eingang 6 und ein LEGO Leuchtstein an den Eingang 7 angeschlossen</p> <p>Eingabe und Ausführung der Anweisung: PRINT PEEK (56577) AND 192</p>	<p>3a. Erscheint die Zahl 192 auf dem Bildschirm?</p>	<p>Weiter: Aktion 4</p>	<p>Test mit anderen LEGO Anschlußkabeln</p> <p>Wenn vorhanden: Test mit einem anderen Interfacekabel</p> <p>Test mit einem anderen Interface</p> <p>Test mit einem anderen Computer</p>
<p>4. Die Leuchtsteine werden von den Eingängen 6 und 7 entfernt</p> <p>Eingabe und Ausführung der Anweisung: PRINT PEEK (56577) AND 192</p>	<p>4a. Erscheint die Zahl 0 auf dem Bildschirm?</p>	<p>Die Anschlußkabel und die Eingänge sind in Ordnung</p>	<p>Wenn vorhanden: Test mit einem anderen Interfacekabel</p> <p>Test mit einem anderen Interface</p> <p>Test mit einem anderen Computer</p>

## COMAL 80

```
8000
8010 // end of user program space
8020 END
8030
8040 PROC init
8050   portb:=56577
8060   POKE 56579,63
8070   POKE portb,0
8080 ENDPROC init
8090
8100 PROC biton(num)
8110   IF num<0 OR num>5 THEN errhand(1)
8120   POKE portb,2↑num BITOR PEEK(portb)
8130 ENDPROC biton
8140
8150 PROC bitoff(num)
8160   IF num<0 OR num>5 THEN errhand(1)
8170   POKE portb,PEEK(portb) BITAND (255-2↑num)
8180 ENDPROC bitoff
8190
8200 FUNC getbit(num)
8210   IF num<>6 AND num<>7 THEN errhand(2)
8220   RETURN (PEEK(portb) BITAND 2↑num)=2↑num
8230 ENDFUNC getbit
8240
8250 PROC wait(tim)
8260   IF tim<0 THEN errhand(3)
8270   qt:=TIME+tim*60
8280   REPEAT
8290     UNTIL TIME>qt
8300 ENDPROC wait
8310
8320 PROC errhand(erc)
8330   IF erc=1 THEN PRINT "outputbits must be between 0 and 5"
8340   IF erc=2 THEN PRINT "inputbits must be 6 or 7"
8350   IF erc=3 THEN PRINT "waittime must be positive"
8360   END
8370 ENDPROC errhand
```

## CBM 64 BASIC

```
1 REM INIT                               LINE 10000
2 REM BITON                               LINE 11000
3 REM BITOFF                              LINE 12000
4 REM GETBIT                              LINE 13000
5 REM WAIT                                LINE 14000
6 REM ERRORHANDLING                      LINE 20000
9940 END
9950 REM
9960 REM-----
```

```

9970 REM INIT
9980 REM-----
9990 REM
10000 P=56577
10010 POKE 56579,63
10020 POKE P,0
10030 RETURN
10950 REM
10960 REM-----
10970 REM BITON    PAR:NUM%
10980 REM-----
10990 REM
11000 IF NUM%>=0 AND NUM%<6 THEN GOTO 11020
11010 ERC=1:GOTO 20000
11020 POKE P,(2↑NUM% OR PEEK(P))
11030 RETURN
11950 REM
11960 REM-----
11970 REM BITOFF   PAR:NUM%
11980 REM-----
11990 REM
12000 IF NUM%>=0 AND NUM%<6 THEN GOTO 12020
12010 ERC=1:GOTO 20000
12020 POKE P,PEEK(P) AND (255-2↑NUM%)
12030 RETURN
12950 REM
12960 REM-----
12970 REM GETBIT   PAR:NUM%
12980 REM-----
12990 REM
13000 IF NUM%=6 OR NUM%=7 THEN GOTO 13020
13010 ERC=2:GOTO 20000
13020 Y%=ABS((PEEK(P) AND 2↑NUM%)=2↑NUM%)
13030 RETURN
13950 REM
13960 REM-----
13970 REM WAIT     PAR:TIM%
13980 REM-----
13990 REM
14000 IF TIM%>=0 THEN GOTO 14020
14010 ERC=3:GOTO 20000
14020 QT=TIME+(TIM%*60)
14030 IF QT>TIME THEN GOTO 14030
14040 RETURN
19950 REM
19960 REM-----
19970 REM ERRORHANDLING
19980 REM-----
19990 REM
20000 PRINT:PRINT"PARAMETER ERROR"
20010 IF ERC=1 THEN PRINT "OUTPUTBITS MUST BE BETWEEN 0 AND 5"
20020 IF ERC=2 THEN PRINT "INPUTBITS MUST BE 6 OR 7"
20030 IF ERC=3 THEN PRINT "WAITTIME MUST BE POSITIVE"
20040 END

```





---

Brugsanvisning

---

User instructions

---

Mode d'emploi

---

Gebruiksaanwijzing

---

Istruzioni per l'uso

---

Bruksanvisning

---

Gebrauchshinweise

---