Documentation and Software for Products of Infax GmbH

Martin Hepperle, June 2017

1. Table of Contents

2. Context

This document describes and contains documents and software for expansion boards for Hewlett-Packard computer systems produced by the former company Infax GmbH[1].

These plug-in boards were for HP 9000-200 and 9000-300 computers with DIO slots. They date from approximately the period between 1980 and 1990. Most of the boards were designed for data acquisition and control of industrial testing equipment.

The German company Infax was founded in 1977 as a subsidiary of the North-American company Infotek. The name Infotek was already taken in Germany, so a new name had to be used. The Infax Computer Products GmbH was located in Frankfurt/Main and distributed Infotek products. They also developed their own range of extension boards for HP 9000 computers with DIO interfaces. Some of these boards were designed by the HuDe GmbH in Erkelenz. Besides I/O cards the range also included memory expansion boards which often exceeded the capacity of HP boards. Such high performance memory boards offered between 4 and 32 MB RAM and were available at a cost between 4’000 and 50’000 DM – those were the days.

Legal Note:

The former head of the INFAX company, Theo Stevens, generously provided disks of the software drivers required for INFAX cards as well as printed documents describing their usage. Stevens’ STCS company still supports customers who operate legacy HP equipment. Nevertheless the information and software is still under copyright and may only be used for archival and research purposes. Commercial usage is prohibited.

June 2017, Martin Hepperle

[1] *GmbH* is the German abbreviation for „Gesellschaft mit beschränkter Haftung“ similar to a „Ltd.“ in England.

3. Software

This section summarizes the content of some floppy disk images with CSUBs and BASIC Software for some INFAX Boards.

The disk images were copied from 3.5" floppy disks. They contain CSUBs with sources and test/demo programs for HP-BASIC 3, 4, 5 and, 6 The CATalogue listings below were created with the HPDIR software (developed and provided by A. Kückes).

Note that many of the TEXT files are Pascal / Assembler source code files.

The following disk images are included in this archive:

- DA.LIF digital/analogue converter card
- DU200.LIF digital clock with battery backup card
- FG200.LIF frequency generator card
- FZ200.LIF frequency counter card
- IN200.LIF digital input card
- OUT200.LIF digital output card
- REL200.LIF relais card
- DEMO.LIF |
- CMCards.LIF |
- CMCards-B4.LIF | some BASIC programs
- CARDS300.LIF | for DA300/301 cards

3.1. DA.LIF

3.5” disk, Tracks 0-76, Label:

[CAT listing] which differs from content listed below

```
VOLUME LABEL: DA
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

da            TEXT        76      256        12     1-Mar-00 15:12
gda           TEXT         8      256        88     1-Mar-00 18:42
pda           TEXT        76      256        96
datest        TEXT         8      256       172
bda           TEXT         8      256       245     1-Mar-00  7:52
bda5          TEXT         8      256       253     1-Mar-00  7:53
dainit        PROG        75      256       261     1-Mar-00  2:34
GDA_5         PROG        75      256       336    23-Nov-87 17:46
MAKEDISK      PROG         2      256       426
da            ASC        122      256       444     1-Mar-00 15:22
danotes       TEXT        16      256       566     1-Mar-00 15:59
da200         TEXT        48      256       582     1-Mar-00 18:12
DA_Demo5      PROG        89      256       630    23-Nov-87 17:44
GDA_6         PROG        74      256       719    16-Sep-91 19:54
gda5          TEXT         8      256       795
DA_Demo4      PROG        89      256       803
GDA_Demo4     PROG        90      256       892
pda           CODE        22      256       982
DA_6          PROG        73      256      1079    16-Sep-91 19:53

360960 of 627712 bytes free.
```

3.2. DU200.LIF

3.5” disk, Tracks 0-66, Label:

DU200 (3)	
DU_4	DU_3
DU_Demo4	DU_Demo3
du.TEXT	DU_5__6/.1
Bdu.TEXT	+Demo
MAKEDISK	

```
VOLUME LABEL: DU
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

bdu           TEXT         8      256        12     1-Mar-00 17:27
du            TEXT        32      256        20     1-Mar-00 17:50
DU_Demo4      PROG        27      256        52
DU_4          PROG        23      256        79     1-Mar-00 17:52
DU_Demo3      PROG        27      256       102
DU_3          PROG        23      256       129
MAKEDISK      PROG         2      256       152
du200         TEXT        24      256       154     1-Mar-00 18:07
TIME2_1       PROG        27      256       178
bdu2_1        TEXT         8      256       205
DU_5          PROG        23      256       213     1-Mar-00  8:18
bdu5          TEXT         8      256       236     1-Mar-00  8:18
Rev_A2        ASCII        1      256       244    15-Sep-91 16:56
DU_5__6       PROG        41      256       245    15-Sep-91 16:56
DUDemo5__6    PROG        48      256       286    15-Sep-91 16:56
SourceCode    ASCII        1      256       334    15-Sep-91 16:57
du_5__6       PROG        54      256       335    15-Sep-91 16:57
DU_6          PROG        23      256       389    16-Sep-91 14:50
bdu6          TEXT         8      256       412    16-Sep-91 14:50

179200 of 283648 bytes free.
```

3.3. FG200.LIF

3.5” disk, Tracks 0-66, Label:

FG200	(5)
FG_4	FC_3
FG_Demo4	FG_Demo3
fg.TEXT	
FG.TEXT	
bfc.TEXT	
MAKEDISK	

```
VOLUME LABEL: FG
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

FG_Demo4      PROG       103      256        12     1-Mar-00 20:52
FG_4          PROG        94      256       115     1-Mar-00 20:52
bfg           TEXT         8      256       209     1-Mar-00 20:52
fg            TEXT        20      256       217     1-Mar-00 20:52
FG            TEXT        36      256       237     1-Mar-00 20:52
FG_Demo3      PROG       103      256       273
FG_3          PROG        94      256       376
```

```
MAKEDISK      PROG          2      256        470
fg200         TEXT         24      256        472     1-Mar-00 18:11
FG_5          PROG         92      256        496     1-Mar-00  9:02
bfg5          TEXT          8      256        588     1-Mar-00  9:02
bfg6          TEXT          8      256        596    15-Sep-91 19:04
FG_6          PROG         92      256        604    15-Sep-91 19:05

108544 of 283648 bytes free.
```

3.4. FZ200.LIF

3.5” disk, Tracks 0-66, Label:

FZ200	(4)
FS_4	FZ_3
FZ_Demo4	FZ_Demo3
FZ.TEXT	
2Hz.TEXT	
6Hz.TEXT	
MAKEDISK	

```
VOLUME LABEL: FZ
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

MAKEDISK      PROG          2      256         12
FZ_Demo4      PROG         72      256         14
FZ_4          PROG         59      256         86
rtfz          TEXT         16      256        145     1-Mar-00  9:29
FZ            TEXT         40      256        161     1-Mar-00  9:30
bfz           TEXT          8      256        201     1-Mar-00  9:30
gfz           TEXT          8      256        209     1-Mar-00  9:30
FZ_Demo3      PROG         72      256        217
FZ_3          PROG         59      256        289
fz200         TEXT         40      256        348     1-Mar-00 18:07
fz_2_1        PROG         22      256        388
FZ_2_1        PROG         13      256        410
FZ_Demo2_1    PROG         21      256        423
FZ_5          PROG         58      256        444     1-Mar-00  9:17
gfz5          TEXT          8      256        502     1-Mar-00  9:17
bfz5          TEXT          8      256        510     1-Mar-00  9:17
fz21          PROG         38      256        518     1-Mar-00 13:39
FZ_Demo       PROG         22      256        556
bfz6          TEXT          8      256        578    15-Sep-91 19:20
rtfza         TEXT         24      256        596     1-Mar-00 14:07
b4fz          TEXT          8      256        620     1-Mar-00 14:08
b5fz          TEXT          8      256        628     1-Mar-00 14:08
FZ_Demo5a     PROG         73      256        636     1-Mar-00 14:08
FZ_Demo4a     PROG         73      256        709     1-Mar-00 14:09
FZ_Demo3a     PROG         73      256        782
FZ_6          PROG         58      256        855    15-Sep-91 19:19
b6fz          TEXT          8      256        913    15-Sep-91 19:20
fzall         PROG         39      256        921    29-Mar-93  7:36
fzalldemo     PROG         41      256        960    30-Mar-93  0:42
FZalldemo     PROG         42      256       1001    19-Apr-93 12:24

22272 of 283648 bytes free.
```

3.5. IN 200.LIF

*** data error on track 0 (nevertheless content seems to be o.k. – to be validated)

3.5” disk, Tracks 0-66, Label:

IN200	(6)
IN_4	IN_3
IN_Demo4	IN_Demo3
bin.TEXT	
in.TEXT	
MAKEDISK	

```
VOLUME LABEL: IN
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

bin           TEXT         8      256        12     1-Mar-00 16:13
in            TEXT        12      256        20
IN_3          PROG         5      256        32
MAKEDISK      PROG         2      256        37
IN_Demo4      PROG         7      256        39
IN_4          PROG         5      256        46
IN_Demo3      PROG         7      256        51
in200         TEXT        24      256        58     1-Mar-00 18:08
Bin           TEXT         8      256        82     1-Mar-00  2:15
inx           TEXT        20      256        90     1-Mar-00  8:04
binx          TEXT         8      256       110     1-Mar-00  8:05
IN_5          PROG         5      256       118     1-Mar-00  8:10
bin5          TEXT         8      256       123     1-Mar-00  8:10
INX_Demo4     PROG        10      256       131     1-Mar-00  8:12
IN_6          PROG         5      256       141    15-Sep-91 19:08
bin6          TEXT         8      256       146    15-Sep-91 19:08

247296 of 283648 bytes free.
```

3.6. OUT200.LIF

3.5” disk, Tracks 0-66, Label:

OUT200	(7)
OUT_4	OUT_3
OUT_Demo4	OUT_Demo3
bout.TEXT	
out.TEXT	
MAKEDISK	

```
VOLUME LABEL: OUT
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

bout          TEXT         8      256        12     1-Mar-00 16:12
out           TEXT        20      256        20
OUT_4         PROG         6      256        40
MAKEDISK      PROG         2      256        46
OUT_Demo4     PROG        10      256        48
OUT_3         PROG         6      256        58
OUT_Demo3     PROG        10      256        64
out200        TEXT        28      256        74     1-Mar-00 18:11
pout          TEXT        16      256       102     1-Mar-00  3:29
poutex        TEXT         8      256       118     1-Mar-00  3:38
pout          CODE        16      256       126     1-Mar-00  3:29
poutex        CODE        14      256       142     1-Mar-00  3:31
OUT_5         PROG         6      256       156     1-Mar-00  7:47
```

```
bout5          TEXT          8      256        162     1-Mar-00  7:48
OUT_6          PROG          6      256        170    15-Sep-91 18:55

241664 of 283648 bytes free.
```

3.7. REL200.LIF

3.5” disk, Tracks 0-66, Label:

REL200	(8)
REL_4	REL_3
REL_Demo4	REL_Demo3
rel.TEXT	
brel.TEXT	
MAKEDISK	

```
VOLUME LABEL: REL
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

rel           TEXT        20      256        12     1-Apr-00 11:30
brel          TEXT         8      256        32     1-Mar-00 16:11
REL_5         PROG         6      256        40     1-Mar-00  7:54
brel5         TEXT         8      256        46     1-Mar-00  7:55
MAKEDISK      PROG         2      256        56
REL_3         PROG         6      256        58
REL_Demo3     PROG         9      256        64
rel200        TEXT        28      256        73     1-Mar-00 18:11
REL_4         PROG         6      256       101
REL_Demo4     PROG         9      256       107
RELCP         HP-UX        8      256       116     1-Aug-91 19:39
RELC          HP-UX        7      256       124     1-Aug-91 19:39
RELDOC        HP-UX        8      256       131     1-Aug-91 19:39
RELMAKE       HP-UX        2      256       139     1-Aug-91 19:39
REL200        HP-UX       12      256       141     1-Aug-91 19:39
REL_6         PROG         6      256       153    15-Sep-91 19:31
brel6         TEXT         8      256       159    15-Sep-91 19:31

244480 of 283648 bytes free.
```

3.8. DEMO.LIF

3.5” disk, Tracks 0-96, Label:

DEMOPROGRAMME	HP200
*** HUDE - SOFT ***	

```
VOLUME LABEL: B9826
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

hp200_uhr     PROG        24      256        16
hp200_rel     PROG        21      256        40
hp200_da      PROG        21      256        61
hp200_in      PROG        18      256        82
hp200_out     PROG        21      256       100
hp200_gen     PROG        25      256       147
hp200_freq    PROG        18      256       172
pulse_trai    PROG        83      256       190
hudef         PROG        71      256       327
optout_cha    PROG        39      256       398
```

```
relais         PROG         42       256        438
save           ASCII        12       256        480     1-Sep-91 18:15
frequency      PROG         57       256        537

153344 of 282624 bytes free.
```

Notes: The volume label indicates a HP 9826 computer. I used the program “hp200_in” to test my IN 200/300 card.

3.9. CMCards.LIF

3.5” disk, Tracks 0-66, Label:

CSUB's INFAX Control&Measurement Cards 21. Mrz 1987

```
VOLUME LABEL: IOCSUB
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

hp200_uhr      PROG         24       256         12
hp200_rel      PROG         21       256         36
hp200_da       PROG         21       256         57
hp200_in       PROG         18       256         78
hp200_out      PROG         21       256         96
hp200_gen      PROG         25       256        117
hp200_freq     PROG         74       256        142
DU             TEXT         32       256        216
bdu            TEXT          8       256        248
TIME           PROG         23       256        256
ClockDemo      PROG         27       256        279
DA             TEXT         24       256        306
bda            TEXT          8       256        330
DA             PROG         41       256        338
DA_Demo        PROG         76       256        379
FZ             TEXT         40       256        455     1-Mar-00 21:37
bfz            TEXT          8       256        495     1-Mar-00 21:41
rtfz           TEXT         12       256        503     1-Mar-00 21:47
retfz          TEXT         12       256        515     1-Mar-00 21:48
FZ             PROG         61       256        527
FZ_Demo        PROG         79       256        588
FG             TEXT         24       256        667     1-Mar-00 21:58
fg             TEXT         20       256        691     1-Mar-00 21:59
bfg            TEXT          8       256        711     1-Mar-00 22:00
FG             PROG         86       256        719
FG_Demo        PROG         94       256        805
OPTO           TEXT         20       256        899     1-Mar-00 22:10
opto           TEXT         12       256        919     1-Mar-00 22:12
OPTO           PROG         44       256        931
REL            PROG         43       256        975
bopto          TEXT          8       256       1018
CLOCKDEMO      ASCII        17       256       1026

19712 of 283648 bytes free.
```

3.10. CMCards-B4.LIF

3.5” disk, Tracks 0-66, Label:

CSUB's (1)

```
VOLUME LABEL: IOCSUB
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

hp200_uhr     PROG        24      256        12
hp200_rel     PROG        21      256        36
hp200_da      PROG        21      256        57
hp200_in      PROG        18      256        78
hp200_out     PROG        21      256        96
hp200_gen     PROG        25      256       117
hp200_freq    PROG        74      256       142
HP200_REL     PROG        18      256       216

3328 of 283648 bytes free.
```

3.11. CARDS300.LIF

3.5” disk, Tracks 0-66, Label:

CARDS
mc
da300
da301

```
VOLUME LABEL: B9826
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS     DATE     TIME

CARDS         PROG        25      256        32    21-Jun-88 17:23
mc            PROG       126      256        57    20-Feb-90 15:58
da300         PROG        74      256       183    22-Feb-90 17:19
da301         PROG        73      256       257    22-Feb-90 11:42

183296 of 283648 bytes free.
```

Note: The volume label indicates a HP 9826 computer.

4. Instruction Sheets

This section contains the text documents which were used to create the instruction sheets for the individual cards. They can be found on the individual disks and are reprinted here for convenience. A partial set of scanned printouts of a complete manual in German is appended and overlaps with these text documents. The text documents do not contain figures which were copied and pasted into the final templates for printing of the paper documents.

4.1. DIGITAL-TO-ANALOG CONVERTERS DA201/202

```
DIGITAL-TO-ANALOG CONVERTERS DA201/202
--------------------------------------


The DA20X converter card occupies one slot in a HP9000 Series
200/300 computer.

It is used to output discrete voltages in the range 0..10 V or
-10..+10 V. The range is toggled by positioning the jumpers on the
card.

There are two versions of the card available

1) DA201 1 analog output channel
2) DA202 2 analog output channels




WARNING

When inserting or removing a card the computer has to be
powered down. If this rule is not observed, the computer
or the card or both may be damaged.


Do not touch the gold-plated contacts on the card. The
integrated circuits on the card are sensitive to
electrostatic discharge. You should, therefore, hold
the card only by the back-plate.



Each interface card has to have a separate address, the select
code, which is shared by no other card in the computer. The preset
factory select code is 20 for this card. It does not need to be
changed except to avoid having two or more cards share the same
address.

When the computer is switched on, all the occupied select codes
will be listed on the screen with the short designation, in case of
standard HP cards. If a special-purpose, non-HP card, such as
DA20X, is present it will be described as either ID15 or ID16. In
order to find out which Infax cards are in the computer you should
run program "INFAXCARDS" available on your software distribution
floppy disc.

The address of the card can be changed by using switches 1/4 of
the DIP switch:

Switches
Address   1     2     3     4
-------------------------------
08       on    on    off   on
```

10	off	on	off	on
12	on	off	off	on
14	off	off	off	on
16	on	on	on	off
18	off	on	on	off
20	on	off	on	off
22	off	off	on	off
24	on	on	off	off
26	off	on	off	off
28	on	off	off	off
30	off	off	off	off

1.0 INTRODUCTION

Digital-to-analog converters DA201 and DA202 have been designed for HP9000 desktop computers Series 200/300.

The converter cards are fully compatible to these computers and can be installed without any tools.

They can be set to operate in the following modes:

- output of a discrete voltage
- output of a sequence of discrete voltages
- periodic output of a sequence of discrete voltages

The DA201 has one analog output channel and the DA202 has two analog output channels. The output voltages are in the range of 0..10 V with a resolution of 12 bits. By repositioning jumpers on the card this range can be changed to -10..+10 V.

When using these cards in BASIC the above mentioned operating modes can be accessed by calling CSUB routines which make programming easy.

The CSUB routines have the following headings:

Operating mode: output of a discrete voltage

```
CSUB Dasingle(INTEGER Sc,Ch,Value)
```

Operating mode: output of a sequence of discrete voltages

```
CSUB Damulti(INTEGER Sc,Ch,Array(*),
             OPTIONAL INTEGER Delay,Converted)
```

Operating mode: periodic output of a sequence of discrete voltages

```
CSUB Generate(INTEGER Sc,Ch,Function(*),
              OPTIONAL INTEGER Delay,Converted)
```

2.0 OUTPUTTING A DISCRETE VOLTAGE

In order to output a discrete voltage the CSUB routine Dasingle has to be called. The variable Value, which is passed to the subroutine has to be in the range -2048..+2047. The output voltage will then be in the range 0..10 V or -10..+10 V depending on the position of jumpers.

```
INTEGER Sc       8..30, even addresses only
INTEGER Ch       1..2
INTEGER Value    -2048..+2047
```

```
Example: Dasingle(20,1,1000)

In the above example the DA card at select code 20 will output a
discrete voltage of 1000 units through channel 1. The 1000 units
correspond to 7.4432 V in the 0..10 V range and to 4.8864 V in the
-10..+10 V range according to the following formulas:

     Units+2048
V=10*----------          (0..10 V range)
        4095


     Units+2048
V=20*---------- - 10     (-10..+10 V)
        4095


3.0 OUTPUTTING A SEQUENCE OF DISCRETE VOLTAGES

In order to output a sequence of discrete voltages the CSUB
routine Damulti has to be called. This routine makes possible
output of a sequence of discrete voltages at a high speed of upto
90,000 discrete voltages per second.

 INTEGER Sc       8..30, even addresses only
 INTEGER Ch       1..2
 INTEGER Array(*) -2048..+2047

Example: Damulti(20,1,Sine(*))

In the above example the DA card at select code 20 will output
through channel 1 discrete voltages corresponding to values in the
array Sine(*).


There are two optional parameters to this routine. 'Delay', the
first, can be used to increase the time between consecutive
voltages by Delay*dt us, where 'dt' is a time constant which depends
on the hardware being used, and is in the range 0.4..1.25 us.
'Delay' is an integer in the range 0..32767.

In order to be able to achieve the maximum speed of the DA card,
the array of integer values which are to be output as discrete
voltages has to be prepared in a format somewhat different from
internal HP format. The second optional parameter, 'Converted',
tells the CSUB whether the values in the array have been prepared
for output or not. If its value is non-zero, it means the array has
already been prepared and can be output immediately as discrete
voltages at highest speed. If the value of 'Converted' is zero, or
if it is not passed (since it is optional) the CSUB will first
prepare the array for output (convert it) and then only output it
as discrete voltages. In this case, after output, the CSUB will
require an additional time to restore the original values to the
array.

Since the time spent in preparing data for output and restoring
the original values is not negligible the user can perform the
necessary work in advance so that there is no delay in output. To
do that the following CSUB is used:

      CSUB Conv2da(INTEGER Original(*),OPTIONAL Converted(*))

If only one array is passed it will also be used to receive the
converted values. If the data so converted was required to be used
in their original form, the following inverse CSUB can be used:
```

```
       CSUB Conv2ad(INTEGER Converted(*),OPTIONAL Original(*))
```

which restores the original values to the array.

4.0 PERIODIC OUTPUT OF A SEQUENCE OF DISCRETE VOLTAGES

A variation of the routine outputting a sequence of discrete voltages is CSUB Generate, which is used to repeatedly output a function analytically defined in the array Function(*). This array has to be dimensioned so that the total size of elements is a power of 2 in the range between 1 and 2^22 (i.e. 1, 2, 4, 8, etc.). For example:

```
10 INTEGER Dac,Sine(0:15)
15 Dac=20                  ! address of the card
20 DATA 0,784,1448,1892,2047,1892,784,0,-784,-1892,-2047,-1892,-784
30 READ Sine(*)            ! read the array
40 ON DELAY 100 GOTO 60
50 Generate(Dac,1,Sine(*))
60 END
```

In the above example the 16-element function array Sine contains values of a sinusoid with amplitude 2047. This function will be generated repeatedly for a period of 100 seconds and then the enabled DELAY event will stop it.

This CSUB adds significant flexibility to the DA card. The period of the function can be controlled in three ways: by the length of the array, by the number of times the function is repeated in the array and by varying the amount of 'Delay'. The longer the array is the longer will also the function's period be. The function's period can be reduced by repeating the function a whole number of times in the array. And finally the higher the value of 'Delay' the longer the function's period will be.

Another feature of this CSUB is that all the KBD-originating events remain effective even inside the routine. Since periodic output means repeating the same sequence of discrete voltages over and over again, the routine can theoretically loop around the same function indefinitely. In order to impose some limit any time-related event can be enabled to stop the routine and, in addition, touching any key will force the routine to be exited.

5.0 POSITIONING OF JUMPERS AND NULL- AND GAIN-ADJUSTMENT

The output voltage of each channel can be separately adjusted to either 0..10 V or -10..+10 V range by positioning the appropriate jumpers as illustrated in the following schematic diagram:

Range 0..10 V
Range -10..+10 V
[figure missing]

In order to adjust the null- and gain- settings of the card you should configure the card in 0..10 V range by means of jumpers and then adjust the corresponding null- and gain-potis while at the same time checking the output voltage by means of a Digital Voltmeter. The reference voltage can be output by CSUB routine Dasingle. The 0 V is output when the 'Value' parameter is -2048 and the +10 V is obtained if 'Value' is +2047.

Output of a 0 V at channel 1 of a DA card at select code 20 :

```
Dasingle(20,1,-2048)
```

Output of +10 V at channel 1 of a DA card at select code 20:

```
Dasingle(20,1,2047)
```

6.0 TECHNICAL DATA

Resolution	12 bits
Value range	-2048..+2047
Voltage ranges	0..10 V
	-10..10 V
max. Current	5 mA
Unit voltage	4.88 mV in 0..10 V range
	2.44 mV in -10..10 V range
Stability	8 ppm
Settling time	< 20 us
Trimmer adjustments	amplitude and zero

7.0 ERRORS

ERROR 388: Attempting to access a DA card at a vacant select code

There will be no effect to attempting to access a DA card at a select code occupied by a card other than DA.

The range of values supplied to any of the above routines has to be -2048..2047 because all other values are essentially undefined and the response of the card is unpredictable.

4.2. DIGITAL CLOCK DU200

DIGITAL CLOCK DU200

The DU200 opto-coupler card occupies one slot in a HP9000 Series 200/300 computer.

The card is fully compatible to these computers and can be installed without any tools.

WARNING

When inserting or removing a card the computer has to be powered down. If this rule is not observed, the computer or the card or both may be damaged.

Do not touch the gold-plated contacts on the card. The integrated circuits on the card are sensitive to electrostatic discharge. You should, therefore, hold the card only by the back-plate.

Each interface card has to have a separate address, the select code, which is shared by not other card in the computer. The preset factory select code is 18 for this card. It does not need to be changed except to avoid having two or more cards share the same address.

When the computer is switched on, all the occupied select codes will be listed on the screen with the short designation, in case of standard HP cards. If a special-purpose, non-HP card, such as DU200, is present it will be described as either ID15 or ID16. In order to find out which Infax cards are in the computer you should run program "INFAXCARDS" available on your software distribution floppy disc.

The address of the card can be changed by using switches 1/4 of the DIP switch:

Switches Address	1	2	3	4
08	on	on	off	on
10	off	on	off	on
12	on	off	off	on
14	off	off	off	on
16	on	on	on	off
18	off	on	on	off
20	on	off	on	off
22	off	off	on	off
24	on	on	off	off
26	off	on	off	off
28	on	off	off	off
30	off	off	off	off

1.0 OVERVIEW

The DU200 digital clock functions independently of the main power supply and will therefore continue to work even when the

computer is switched off. The built-in battery which is recharged from the computer's main power supply has enough power to make it tick for about a year without recharging.

The clock can be used to inquire:

- the actual time of day
- the actual date

It can therefore be set to

- a new time of day
- a new date
- a new weekday

All the functions of the digital clock are accessed through CSUB routines which makes programming easy. The CSUB routines are used to set up individual operating modes of the card and to pass input and output parameters to and from the card.

2.0 ACTUAL TIME OF DAY

```
CSUB Time(INTEGER Sc,Time$)
```

The CSUB routine Time passes in the string variable Time$ the system time of the DU200 card in the format "hh:mm:ss". The integer variable Sc has the value of the hardware select code of the card (factory setting: 18).

3.0 SETTING THE TIME OF DAY

```
CSUB Set_time(INTEGER Sc,New_time$)
```

The CSUB routine Set_time sets the DU200 clock to the time of day contained in the string variable New_time$ in the format "hh:mm:ss".

4.0 ACTUAL DATE

```
CSUB Date(INTEGER Sc,Date$)
```

The CSUB routine Date passes in the string variable Date$ the current date of the DU200. The format is "dd Mmm yyyy Ddd" where dd=day-date (1..31), Mmm=month (Jan..Dec), yyyy=year (for example 1987) and Ddd=weekday (Mon..Sun).

5.0 SETTING THE DATE

```
CSUB Set_date(INTEGER Sc,New_date$)
```

The CSUB routine Set_date sets the date of the DU200 card to the value passed in the string variable New_date$ in the format "dd Mmm yyy" as explained above.

6.0 SETTING THE DAY OF WEEK

```
CSUB Set_wk_day(INTEGER Sc,Day$)
```

This CSUB routine allows in addition setting of the day of week. The day of the week is output when reading the date. To set the day of the week the string variable Day$ has to contain the shortened name of the day in the form "Ddd" (Mon..Sun).

7.0 ERRORS

ERROR 388: Attempting to access a DU card at a vacant select code

There will be no effect to attempting to access a DU card at a select code occupied by a card other than DU.

4.3. FREQUENCY GENERATOR CARD FG200

FREQUENCY GENERATOR CARD FG200

The FG200 frequency generator card occupies one slot in a HP9000 Series 200/300 computer.

It is used to generate a pulse or a train of pulses.

WARNING

When inserting or removing a card the computer has to be powered down. If this rule is not observed, the computer or the card or both may be damaged.

Do not touch the gold-plated contacts on the card. The integrated circuits on the card are sensitive to electrostatic discharge. You should, therefore, hold the card only by the back-plate.

Each interface card has to have a separate address, the select code, which is shared by no other card in the computer. The preset factory select code is 28 for this card. It does not need to be changed except to avoid having two or more cards share the same address.

When the computer is switched on, all the occupied select codes will be listed on the screen with the short designation, in case of standard HP cards. If a special-purpose, non-HP card, such as FG200, is present it will be described as either ID15 or ID16. In order to find out which Infax cards are in the computer you should run program "INFAXCARDS" available on your software distribution floppy disc.

The address of the card can be changed by using switches 1/4 of the DIP switch:

Address	Switches 1	2	3	4
08	on	on	off	on
10	off	on	off	on
12	on	off	off	on
14	off	off	off	on
16	on	on	on	off
18	off	on	on	off
20	on	off	on	off
22	off	off	on	off
24	on	on	off	off
26	off	on	off	off
28	on	off	off	off
30	off	off	off	off

1.0 OVERVIEW

The FG200 frequency generator was specially designed for HP9000

desktop computers Series 200/300. The card is fully compatible to these computers and can easily be installed in an empty I/O slot of the computer without any tools.

The FG200 interface card can be operated in the following modes:

- generation of a pulse train
- generation of a single pulse

The card generates pulses through two independent channels. Each of the channels can be accessed through corresponding BNC connector on the card's back-plate.

The output of both channels comply with standard TTL logic. The output frequency can be varied in the range 1.53 Hz .. 25 kHz (corresponding to the period range of 40 us .. 655.35 ms, variable in steps of 10 us).

Both of the above mentioned modes are accessed through CSUB routines which makes programming of the card easy. The CSUB routines are used to set up the operating mode and to pass the required parameters to the card.

The CSUB routines have the following headings:

Operating mode: generation of a pulse train

```
CSUB Pulse_train(INTEGER Sc,Command$,OPTIONAL Status$)
```

Operating mode: generation of a pulse

```
CSUB Impulse(INTEGER Sc,Command$,OPTIONAL Status$)
```

2.0 GENERATION OF A PULSE TRAIN

To generate a pulse train with required period length the CSUB routine Pulse_train has to be called from the program.

INTEGER Sc 8..30, even addresses only

Command$ "channel 1 width 0.04"
                  2   .
                          655.35

Status$ is an optional string variable which returns the error message in case the command string was not correctly formulated.

Example:

```
Pulse_train(28,"channel 1 width 0.04",Status$)
```

In the above example the frequency generator card at select code 28 is set up to output a pulse train with period length of 0.04 ms corresponding to 25 kHz.

3.0 GENERATION OF A PULSE

To generate a pulse of required length the CSUB routine Impulse has to be called.

INTEGER Sc 8..30, even addresses only

```
Command$  "channel 1 width   0.04"
                    2          .
                            655.35

Status$   is an optional string variable which contains an error
          message in case the command string was not correctly
          formulated.


Example:

       Impulse(28,"channel 2 width 1.16",Status$)

The frequency generator card at select code 28 is set up to
output a pulse of length 1.16 ms at its channel 2.



4.0 ERRORS

ERROR 388: Attempting to access an FG card at a vacant select
code

There will be no effect to attempting to access an FG card at a
select code occupied by a card other than FG.
```

4.4. FREQUENCY COUNTER CARD FZ200

FREQUENCY COUNTER CARD FZ200

The FZ200 frequency counter card occupies one slot in a HP9000 Series 200/300 computer.

It is used for various frequency related measurements.

WARNING

When inserting or removing a card the computer has to be powered down. If this rule is not observed, the computer or the card or both may be damaged.

Do not touch the gold-plated contacts on the card. The integrated circuits on the card are sensitive to electrostatic discharge. You should, therefore, hold the card only by the back-plate.

Each interface card has to have a separate address, the select code, which is shared by no other card in the computer. The preset factory select code is 30 for this card. It does not need to be changed except to avoid having two or more cards share the same address.

When the computer is switched on, all the occupied select codes will be listed on the screen with the short designation, in case of standard HP cards. If a special-purpose, non-HP card, such as FZ200, is present it will be described as either ID15 or ID16. In order to find out which Infax cards are in the computer you should run program "INFAXCARDS" available on your software distribution floppy disc.

The address of the card can be changed by using switches 1/4 of the DIP switch:

Address	Switches 1	2	3	4
08	on	on	off	on
10	off	on	off	on
12	on	off	off	on
14	off	off	off	on
16	on	on	on	off
18	off	on	on	off
20	on	off	on	off
22	off	off	on	off
24	on	on	off	off
26	off	on	off	off
28	on	off	off	off
30	off	off	off	off

1.0 OVERVIEW

The frequency counter card FZ200 is specially designed for

HP9000 desktop computers Series 200/300. The card is fully compatible to these computers and can easily be installed in an empty I/O slot without any tools.

The FZ200 card can be used to perform the following measurements:

- frequency measurement
- frequency ratio of two signals
- pulse counting
- measurement of interval time
- measurement of period length
- pulse counting in terms of r.p.m.

The card has 4 input channels. One pair of channels handles TTL signals (channels 1a and 2a) and another pair (1b and 2b) has an adjustable threshold to measure non-TTL signals.

Input A can be used for TTL signals up to 10 MHz. Input B has an additional adjustable trigger amplifier, and can handle inputs from 50 mV to 15 V. The maximum frequency in this mode is 2 MHz.

The inputs 2a and 2b are only used when measuring frequency ratio or interval timing.

All of the above listed operating modes of the card are supported by a set of CSUB routines which make programming and implementation of the card in existing software very easy.

By calling these CSUB routines the operating mode is set and parameters are passed to and from the card. The parameters include a command string and a return variable which contains the result of the measurement.

The CSUB routines have the following general heading:

```
CSUB Function(INTEGER Sc,REAL Retvar,OPTIONAL Command$,Status$)
```

where Function can be any of the following:

- Frequency
- Frq_ratio
- Count_rst
- Interval
- Period
- R_p_m

INTEGER Sc 8..30, even addresses only

REAL Retvar the variable which contains the result of measurement

Command$ the string variable which contains command string for the card

Status$ the string variable which returns the error message indicating the correctness of the command string. A string of dashes indicates a correct command while other letters indicate which part of the command was incorrectly formulated.

2.0 FREQUENCY MEASUREMENT

The frequency is measured with 8 digits of precision and the accuracy depends on the gate time in the command string which has

```
the following form

Command$="channel 1a gate 0.01"
                  1b      0.1
                                1
                               10

       CSUB Frequency(INTEGER Sc,REAL Khz,OPTIONAL Command$,Status$)



3.0 FREQUENCY RATIO OF TWO SIGNALS

The measurement is performed on a pair of channels. It can be
averaged over several periods. If for example a signal with
frequency of 5 MHz is passed through channel 1 and another signal
of 2.5 MHz is passed through channel 2 the results would be:

averaged over    1 period = Hz_ratio = 2.
                10                     2.0
               100                     2.00
              1000                     2.000

The command string has the following form:

Command$="channel 1a channel 2a    1 periods"
                  1b         2b   10
                                 100
                                1000

       CSUB Frq_ratio(INTEGER Sc,REAL Ratio,OPTIONAL Command$,Status$)



4.0 PULSE COUNTING

When using this operating mode pulses or periods are counted.
The trigger point can be chosen to be on negative or positive edge
of the signal. After completion the return variable contains the
number of pulses or periods counted.

The command string has the following form:

Command$="channel 1a trig neg"
                  1b      pos

       CSUB Counter_rst(INTEGER Sc,REAL Count, OPTIONAL Command$,Status$)



5.0 INTERVAL TIMING

This operating mode makes possible timing between selected edge
of channel 1 and the selected edge of channel 2. The timing can be
averaged over several periods. If the same signal is passed through
both channels the interval between two edges can also be measured
(half-period).

The command string has the following form:

Command$="channel 1a trig neg channel 2a tring neg    1 periods"
                  1b      pos         2b       pos   10
                                                    100
                                                   1000

       CSUB Interval(INTEGER Sc,REAL Ms,OPTIONAL Command$,Status$)
```

6.0 PERIOD TIMING

The period timing is also measured with 8 digits of precision.
The timed result can be averaged over several periods. The command
string has the following form:

```
Command$="channel 1a trig neg    1 periods"
                  1b      pos   10
                               100
                              1000

       CSUB Period(INTEGER Sc,REAL Ms,OPTIONAL Command$,Status$)
```

7.0 R.P.M. MEASUREMENT

The r.p.m. measurement is made by means of period measurement
and the result is expressed in terms of rotations per minute. The
result can also be averaged over several periods.

The command string has the following form:

```
Command$="channel 1a trig neg    1 periods"
                  1b      pos   10
                               100
                              1000

       CSUB R_p_m(INTEGER Sc,REAL Rotations,OPTIONAL Command$,Status$)
```

8.0 READING OF THE FREQUENCY CARD

The CSUB routine Read_fz_card allows repeated reading of the
card which was previously set-up by another routine like, for
example, Interval or R_p_m.

The card maintains its operating mode until reset by another
mode so that repeated readings are possible by using the routine
Read_fz_card.

This simplifies programming and also allows to read the running
total of pulses in the pulse-counting mode.

This CSUB routine has the following form:

```
       CSUB Read_fz_card(INTEGER Sc,REAL Reading)
```

9.0 TECHNICAL DATA

```
   Input group A: input frequency upto  10 MHz for TTL signals
Schmitt-Trigger input

Input group B: input frequency upto   2 MHz
Trigger point setable between 0 and 5 V by means
                  of a potentiometer.

Input range between 50 mV and 15 V

All 4 input connectors are of BNC type
```

10.0 ERRORS

ERROR 388: Attempting to access an FZ card at a vacant select code

There will be no effect to attempting to access an FZ card at a select code occupied by a card other than FZ.

4.5. INPUT OPTO-COUPLER CARD IN200

INPUT OPTO-COUPLER CARD IN200

The IN200 opto-coupler card occupies one slot in a HP9000 Series 200/300 computer.

It is used to galvanically separate currents at different potentials.

WARNING

When inserting or removing a card the computer has to be powered down. If this rule is not observed, the computer or the card or both may be damaged.

Do not touch the gold-plated contacts on the card. The integrated circuits on the card are sensitive to electrostatic discharge. You should, therefore, hold the card only by the back-plate.

Each interface card has to have a separate address, the select code, which is shared by no other card in the computer. The preset factory select code is 22 for this card. It does not need to be changed except to avoid having two or more cards share the same address.

(When the computer is switched on, all the occupied select codes will be listed on the screen with the short designation, in case of standard HP cards. If a special-purpose, non-HP card, such as IN200, is present it will be described as either ID15 or ID16. In order to find out which Infax cards are in the computer you should run program "INFAXCARDS" available on your software distribution floppy disc.

The address of the card can be changed by using switches 1/4 of the DIP switch:

Switches Address	1	2	3	4
08	on	on	off	on
10	off	on	off	on
12	on	off	off	on
14	off	off	off	on
16	on	on	on	off
18	off	on	on	off
20	on	off	on	off
22	off	off	on	off
24	on	on	off	off
26	off	on	off	off
28	on	off	off	off
30	off	off	off	off

1.0 OVERVIEW

(The opto-electronic coupler card IN200 is specially designed for HP9000 desktop computers Series 200/300. The card has 16 input

channels and can easily be installed in an empty I/O slot of the computer without any tools.

By galvanically separating the circuits it can be used to monitor currents at different potentials without any harm to the host computer.

The IN200 interface card can be operated in the following modes:

- testing the status of a selected channel
- testing the status of all channels simultaneously

Each of these operating modes is accessed through a CSUB routine which makes programming of the card easy. The CSUB routines set up the required operating mode and pass required parameters to and from the card.

The CSUB routines have the following headings:

Operating mode: testing of a selected channel

```
CSUB In_channel(INTEGER Sc,Channel,Active)
```

Operating mode: testing all channels

```
CSUB In_word(INTEGER Sc,Active$)
```

2.0 TESTING OF A SELECTED CHANNEL

For example:

```
In_channel(22,4,Status)
```

The IN200 card at select code 22 is accessed and the status of its channel 4 is tested. If there is current passing the variable Status will have value 1 otherwise it will be zero.

Voltage on = 1
Voltage off = 0

3.0 TESTING ALL CHANNELS

For example:

```
In_word(22,Status$)
```

The IN200 card at select code 22 is accessed and all of its channels are tested. The string variable Status$ will contain a string of zero's and one's depending upon whether the corresponding channel is active (1) or inactive (0).

```
Status$="0000010010100011"
```

I.e. the channels 6,9,11,15 and 16 have active status (1) and all other channels are inactive (0).

4.0 TECHNICAL DATA

16 input channels, optically insulated IL74 or equivalent

Input voltage low,0	0..1 V

```
                      high,1                    5..12 V
       Internal resistance                      220 Ohm
       Imax                                     50 mA
       In/output insulation                     min 1200 V
       In/output capacitance                    max 3.0 pF

Pin-out of the 25-pin connector on the card:

Channel   Pin   Channel   Pin
-----------------------------
1          7      9       19
2         21     10       20
3          6     11       18
4          8     12       13
5          9     13       11
6         10     14       24
7         22     15       25
8         23     16       12

Opto-coupler emitor at pin 5 and 17.

Ground HP computer at pin 16.



7.0 ERRORS

ERROR 388: Attempting to access an IN card at a vacant select
code

There will be no effect to attempting to access an IN card at a
select code occupied by a card other than IN.
```

4.6. OUTPUT OPTO-COUPLER CARD OUT200

OUTPUT OPTO-COUPLER CARD OUT200

The OUT200 opto-coupler card occupies one slot in a HP9000 Series 200/300 computer.

It is used to galvanically separate currents at different potentials.

WARNING

When inserting or removing a card the computer has to be powered down. If this rule is not observed, the computer or the card or both may be damaged.

Do not touch the gold-plated contacts on the card. The integrated circuits on the card are sensitive to electrostatic discharge. You should, therefore, hold the card only by the back-plate.

Each interface card has to have a separate address, the select code, which is shared by no other card in the computer. The preset factory select code is 24 for this card. It does not need to be changed except to avoid having two or more cards share the same address.

When the computer is switched on, all the occupied select codes will be listed on the screen with the short designation, in case of standard HP cards. If a special-purpose, non-HP card, such as OUT200, is present it will be described as either ID15 or ID16. In order to find out which Infax cards are in the computer you should run program "INFAXCARDS" available on your software distribution floppy disc.

The address of the card can be changed by using switches 1/4 of the DIP switch:

Address	Switches 1	2	3	4
08	on	on	off	on
10	off	on	off	on
12	on	off	off	on
14	off	off	off	on
16	on	on	on	off
18	off	on	on	off
20	on	off	on	off
22	off	off	on	off
24	on	on	off	off
26	off	on	off	off
28	on	off	off	off
30	off	off	off	off

1.0 OVERVIEW

The opto-electronic coupler card OUT200 is specially designed for HP9000 desktop computers Series 200/300. The card has 16 output channels and can easily be installed in an empty I/O slot of the

computer without any tools.

By galvanically separating the circuits it can be used to
switch currents at different potentials without any harm to the
host computer.

The OUT200 interface card can be operated in the following
modes:

- switching a selected channel
- switching all channels simultaneously
- verification of the status of all channels

Each of these operating modes is accessed through a CSUB routine
which makes programming of the card easy. The CSUB routines set up
the required operating mode and pass required parameters to and
from the card.

The CSUB routines have the following headings:

Operating mode: switching a selected channel

```
      CSUB Out_channel(INTEGER Sc,Channel,Active)
```

Operating mode: switching all channels

```
      CSUB Out_word(INTEGER Sc,Active$)
```

Operating mode: verifying the status of all channels

```
      CSUB Out_test(INTEGER Sc,Status$)
```

2.0 SWITCHING A SELECTED CHANNEL

For example:

```
      Out_channel(24,4,Status)
```

The OUT200 card at select code 24 is accessed and its channel 4
is set to active state, i.e. the circuit at channel 4 is closed.

```
     Circuit closed = 1
     Circuit opened = 0
```

3.0 SWITCHING ALL CHANNELS

For example:

```
      Out_word(24,"0110100000000000")
```

The OUT200 card at select code 24 is accessed and its channels
are set to active (1) or inactive (0) state according to whether
the corresponding letter in the command string is a '1' or a '0'.
In this case channels 2, 3, and 5 are set active and all other
channels are set inactive.

4.0 VERIFYING THE STATUS OF ALL CHANNELS

For example:

```
      Out_test(24,Status$)
```

The status of all channels of the OUT200 card at select code 24 will be reflected in the string variable Status$ by returning a '1' at the corresponding letter position if the channel is active or a '0' if that channel is inactive.

Status$="0000010010100011"

I.e. the channels 6,9,11,15 and 16 have active status (1) and all other channels are inactive (0).

This CSUB reflects only the cumulative effect of previous uses of the other two routines (Out_channel and Out_word). It will, therefore return the tru status of all channels only if used after all of the channels have been 'touched' by the other two routines. To avoid confusion, an initial invocation of Out_channel to initialize the card is recommended.

5.0 TECHNICAL DATA

16 input channels, optically insulated IL74 or equivalent

Umax Collector/Emitor	30 V
Imax	50 mA
In/output insulation	min 1500 V
In/output capacitance	max 3.0 pF

Pin-out of the 25-pin connector on the card:

Channel	Pin	Channel	Pin
1	7	9	20
2	18	10	19
3	8	11	21
4	6	12	1
5	5	13	3
6	4	14	15
7	17	15	14
8	16	16	2

Opto-coupler emitor at pin 9 and 22.

Ground HP computer at pin 23.

6.0 ERRORS

ERROR 388: Attempting to access an OUT card at a vacant select code

There will be no effect to attempting to access an OUT card at a select code occupied by a card other than OUT.

4.7. RELAIS CARD REL200

```
RELAIS CARD REL200
------------------


The REL200 relais card occupies one slot in a HP9000 Series
200/300 computer.

It is used to switch loads upto 25 W with 50 Veff. It is
equipped with 16 single relais.



WARNING

When inserting or removing a card the computer has to be
powered down. If this rule is not observed, the computer
or the card or both may be damaged.


Do not touch the gold-plated contacts on the card. The
   integrated circuits on the card are sensitive to
electrostatic discharge. You should, therefore, hold
the card only by the back-plate.



Each interface card has to have a separate address, the select
code, which is shared by no other card in the computer. The preset
factory select code is 26 for this card. It does not need to be
changed except to avoid having two or more cards share the same
address.

When the computer is switched on, all the occupied select codes
will be listed on the screen with the short designation, in case of
standard HP cards. If a special-purpose, non-HP card, such as
REL200, is present it will be described as either ID15 or ID16. In
order to find out which Infax cards are in the computer you should
run program "INFAXCARDS" available on your software distribution
floppy disc.

The address of the card can be changed by using switches 1/4 of
the DIP switch:

Switches
Address   1     2     3     4
--------------------------------
08        on    on    off   on
10        off   on    off   on
12        on    off   off   on
14        off   off   off   on
16        on    on    on    off
18        off   on    on    off
20        on    off   on    off
22        off   off   on    off
24        on    on    off   off
26        off   on    off   off
28        on    off   off   off
30        off   off   off   off


1.0 OVERVIEW

The opto-electronic coupler card REL200 is specially designed
for HP9000 desktop computers Series 200/300. The card has 16 output
channels and can easily be installed in an empty I/O slot of the
```

computer without any tools.

The REL200 interface card can be operated in the following
modes:

- switching a selected relais
- switching all relais simultaneously
- verification of the status of all relais

Each of these operating modes is accessed through a CSUB routine
which makes programming of the card easy. The CSUB routines set up
the required operating mode and pass required parameters to and
from the card.

The CSUB routines have the following headings:

Operating mode: switching a selected channel

```
      CSUB Rel_channel(INTEGER Sc,Channel,Active)
```

Operating mode: switching all channels

```
      CSUB Rel_word(INTEGER Sc,Active$)
```

Operating mode: verifyint the status of all channels

```
      CSUB Rel_test(INTEGER Sc,Status$)
```

2.0 SWITCHING A SELECTED RELAIS

For example:

```
      Rel_channel(26,4,1)
```

The REL200 card at select code 26 is accessed and its channel 4
is set to active state, i.e. the circuit at channel 4 is closed.

```
       Circuit closed = 1
       Circuit opened = 0
```

3.0 SWITCHING ALL CHANNELS

For example:

```
      Rel_word(26,"0110100000000000")
```

The REL200 card at select code 26 is accessed and its relais are
set to active (1) or inactive (0) state according to whether the
corresponding letter in the command string is a '1' or a '0'. In
this case channels 2, 3, and 5 are set active and all other
channels are set inactive.

4.0 VERIFYING THE STATUS OF ALL RELAIS

For example:

```
      Rel_test(26,Status$)
```

The status of all relais of the REL200 card at select code 26
will be reflected in the string variable Status$ by returning a '1'

at the corresponding letter position if the relais is active or a '0' if that relais is inactive.

```
Status$="0000010010100011"
```

I.e. the relais 6,9,11,15 and 16 have active status (1) and all other relais are inactive (0).

This CSUB reflects only the cumulative effect of previos uses of the other two routines (Rel_channel and Rel_word). It will therefore return the true status of all relais only if used after all of the relais have been 'touched' by other two routines. To avoid confusion, an initial invocation of Rel_word to initialize the card is recommended.

5.0 TECHNICAL DATA

16 relais

Imax	500 mA
Umax	50 V
Pmax	25 W

Pin-out of the 50-pin connector on the card:

Relais	Pin	Relais	Pin
S 01	50	D 09	15
A	48	A	14
D	49	S	16
S 02	47	D 10	12
A	45	A	11
D	46	S	13
S 03	41	D 11	5
A	39	A	6
D	40	S	4
S 04	44	D 12	2
A	42	A	3
D	43	S	1
S 05	38	D 13	30
A	36	A	31
D	37	S	29
S 06	25	D 14	27
A	23	A	28
D	24	S	26
S 07	22	D 15	8
A	20	A	9
D	21	S	7
S 08	19	D 16	33
A	17	A	34
D	18	S	32
----	10	----	35

S = switch A = active (Close) D = inactive (Open)

6.0 ERRORS

ERROR 388: Attempting to access an REL card at a vacant select code

There will be no effect to attempting to access an REL card at a select code occupied by a card other than REL.

5. Catalogue Pages

The next pages contain pages from a catalogue of the Infax GmbH.

Interfacekarten

Bei Ihren Meß- und Steueraufgaben ist eine optimale Schnittstelle eine Voraussetzung für gute Ergebnisse. Da die Aufgaben von sehr unterschiedlicher Art sein können, sind auch die Anforderungen an die benötigten Schnittstellen sehr unterschiedlich. Daher bieten wir Ihnen mit unserem Interfacekarten-Programm ein weitreichendes Spektrum in dem Sie die, für Sie optimale Interfacekarte finden.

AD 200	Intelligente 16-Kanal-Analog/Digital-Wandlerkarte mit Onboard-Mikroprozessor, 12 Bit Auflösung, 16 Einzelkanäle oder 8 Differenzkanäle, kaskadierbar; bis zu 200.000 Messungen pro Sekunde mit DMA. Inklusive Betriebsanleitung.

Zubehör für AD 200:

CA 200	Flachbandkabel-Interface für AD 200, bestehend aus: 50-poligem Stecker mit 1,60 m Flachbandkabel. Komplett mit Abdeckplatte.
CA 201	Meßinstrument-Interface für AD 200, bestehend aus: 4 BNC Anschlüssen und einem 37-poligen Cannon-Stecker (female) auf Abdeckplatte montiert. Inklusive Installationsanweisung.
CA 202	Industrie Interface, bestehend aus: 44-poliger Klemmleiste zum Anschluß aller IN- und OUTPUT-Signale; auf Abdeckplatte montiert. Inklusive Installationsanweisung.
CA 300	Adaptergehäuse zum Anschluß von 16 Meßkanälen über BNC-Buchsen. Die Steuersignale der AD 200 Karte (TRIGGER-IN, TRIGGER-OUT, SYNCH und READY) sind ebenfalls als BNC-Buchsen ausgelegt. Der Anschluß zur AD 200 Karte ist ausgeführt wie bei dem CA 200 Interface.
AD 300	Intelligente 32-Kanal-Analog/Digital-Wandlerkarte mit Onboard-Mikroprozessor, 12 Bit Auflösung, 32 Einzel- oder 16 Differenzeingänge, 4 Kanäle simultan meßber; 64 KByte Onboard-FIFO-Buffer, programmierbares Triggerlevel; digitaler Ausgabekanal (8 Bit), bis zu 200.000 Meßwerte pro Sekunde. Nur mit Industrie-Interface entsprechend CA 202 lieferbar. Inklusive Betriebsanleitung.
DA 201	1-Kanal-Digital/Analog-Wandlerkarte. 12 Bit Auflösung. Inklusive Driver-Software und Betriebsanleitung.
DAOPT.010	DA-Driver-Software für BASIC 3.0/4.0/5.0
OPT.003	Software auf 3,5" SS Diskette
OPT.005	Software auf 5,25" Diskette.
DA 202	2-Kanal-Digital/Analog-Wandlerkarte. 12 Bit Auflösung. Inklusive Driver-Software und Betriebsanleitung.
DAOPT.010	DA-Driver-Software für BASIC 3.0/4.0/5.0
OPT.003	Software auf 3,5" SS Diskette
OPT.005	Software auf 5,25" Diskette
DA 300	Intelligente 8-Kanal-Digital/Analog-Wandlerkarte mit Onboard-Mikroprozessor, 64 KByte Buffer, externer Trigger, Ausgangsspannungen im Bereich von +/– 5 V oder +/– 10 V; Datendurchsatz 2 MHz. Inklusive Betriebsanleitung.
DU 200	Digitaluhr für HP 9000 Serie 200 liefert Jahr, Monat, Tag, Wochentag, Stunde, Minute und Sekunde; batteriegepuffert mit automatischer Aufladung; quarzgesteuert. Inklusive Driver-Software und Bedienungsanleitung.
DUOPT.010	DU-Driver-Software für BASIC 3.0/4.0/5.0
OPT.003	Software auf 3,5" SS Diskette
OPT.005	Software auf 5,25" Diskette

FG 200	2-Kanal-Pulsgenerator-Einschubkarte. Pulsbreite von 0,04 msec–655 msec programmierbar. Inklusive Driver-Software und Bedienungsanleitung.
FGOPT.010	FG-Driver-Software für BASIC 3.0/4.0/5.0
OPT.003	Software auf 3,5" SS Diskette
OPT.005	Software auf 5,25" Diskette
FZ 200	Frequenzzähler-Einschubkarte. Auflösung 8 Digits bis maximal 10 MHz. Inklusive Driver-Software und Bedienungsanleitung.
FZOPT.010	FZ-Driver-Software für BASIC 3.0/4.0/5.0
OPT.003	Software auf 3,5" SS Diskette
OPT.005	Software auf 5,25" Diskette
IN 200	16-Kanal-Optokoppler-Eingangskarte. Isolation 1500 V min. Inklusive Driver-Software und Bedienungsanleitung.
INOPT.010	IN-Driver-Software für BASIC 3.0/4.0/5.0
OPT.003	Software auf 3,5" SS Diskette
OPT.005	Software auf 5,25" Diskette
OUT 200	16-Kanal-Optokoppler-Ausgangskarte. Isolation 1500 V min. Inklusive 25-poligem Anschlußstecker, Driver-Software und Bedienungsanleitung.
OUTOPT.010	OUT-Driver-Software für BASIC 3.0/4.0/5.0
OPT.003	Software auf 3,5" SS Diskette
OPT.005	Software auf 5,25" Diskette
REL 200	16-Kanal-Relais-Ausgangskarte. Ein Wechsler-Kontakt pro Kanal. Inklusive 25-poligem Anschlußstecker, Driver-Software und Bedienungsanleitung.
RELOPT.010	REL-Driver-Software für BASIC 3.0/4.0/5.0
OPT.003	Software auf 3,5" SS Diskette
OPT.005	Software auf 5,25" Diskette

Einzeldisketten mit Driver-Software für Interface-Karten

DA 010	DA-Driver-Software für BASIC 3.0/4.0/5.0
DU 010	DU-Driver-Software für BASIC 3.0/4.0/5.0
FG 010	FG-Driver-Software für BASIC 3.0/4.0/5.0
FZ 010	FZ-Driver-Software für BASIC 3.0/4.0/5.0
IN 010	IN Driver-Software für BASIC 3.0/4.0/5.0
OUT 010	OUT-Driver-Software für BASIC 3.0/4.0/5.0
REL 010	REL-Driver-Software für BASIC 3.0/4.0/5.0

Bei den Einzeldisketten können Sie zwischen zwei Optionen wählen:

OPT.003	Software auf 3,5" SS Diskette
OPT.005	Software auf 5,25" Diskette

6. Original Instruction Manual Pages

The next pages contain pages scanned from original paper copies provided by Infax GmbH resp. its follow up company STCS GmbH.

1.0 ALLGEMEINES

Die hier beschriebenen Erweiterungskarten :

- OPTOKOPPLER-EINGANGSKARTE
- OPTOKOPPLER-AUSGANGSKARTE
- RELAIS -KARTE
- FREQUENZ/IMPULSGENERATOR
- FREQUENZZÄHLERKARTE
- D/A-WANDLERKARTE
- DIGITALUHR

werden direkt in einen der Hewlett-Packard Tischrechner der Serie 200/300 eingesteckt.

Mit Hilfe der Optokoppler-Karten ist eine optisch isolierte Ein- bzw. Ausgabe mit externen Geräten möglich.

Die Relaiskarte ermöglicht ein potentialfreies Schalten kleinerer und mittlerer Lasten.

Die Freqeunz- und Impulsgeneratorkarte ermöglicht die Ausgabe von unterschiedlichen Rechteckfrequenzen, sowie in der Zeit variable Ausgangsimpulse auf zwei separaten Kanälen.

Die Frequenzzählerkarte ermöglicht die Betriebsarten :
Frequenzmessung,Drehzahlmessung,Frequenzverhältnis,Impulszähler
Intervallzeitmessung,Periodendauermessung.

D/A-Wandler mit ein oder zwei Kanälen zur Ausgabe im Bereich 0...10.24 Volt (2.50mV) oder -10...+10.24 Volt (5.00mV).

Digitaluhr mit Batteriebufferung und automatischer Nachladung.
Jahr,Monat,Tag,Stunde,Minute,Sekunde,Wochentag.

Die zum Betrieb der Karten notwendige Software gehört zum Lieferumfang der jeweiligen Karte. Die Einschubkarten werden durch in Assembler geschriebene CSUB-Programme auf dem hp-Rechner unterstützt.Zu jeder Karte finden Sie auf der mitgelieferten Diskette zwei Programme. Z.B. für die Relaiskarte sind dies das separate CSUB 'RELAIS' und ein von uns erstelltes Programm 'HP200_REL'.

> **ACHTUNG**
> BEIM EINSETZEN ODER HERAUSNEHMEN DER KARTE IST DER RECHNER AUSZUSCHALTEN. ANDERENFALLS SIND BESCHÄDIGUNGEN DES RECHNERS UND DER KARTE ZU ERWARTEN.

Vermeiden Sie eine Berührung der vergoldeten Steckerleiste.Dies könnte zu Kontaktschwierigkeiten führen.

Beachten Sie, daß alle integrierten Schaltungen empfindlich gegen statische Aufladungen sind.

Nehmen Sie die Einschubkarte daher immer an der Metallplatte.

1.1 SELECT - SCHALTER

An jedem Rechner-Interface muß eine Adresse eingestellt werden. Dies ist notwendig, damit die Software des Rechners die vorhandenen Karten erkennen und entsprechend behandeln kann.

Alle im Rechner eingesteckten Einschubkarten MÜßEN auf eine Adresse eingestellt werden, die von keiner anderen Karte belegt wird.

Beachten Sie, daß die Adressen 1 bis 7 durch den Rechner selbst benötigt werden. Diese Adressen dürfen somit nicht eingestellt werden.

Bei Auslieferung der Karten sind die folgenden Adresse voreingestellt.

Adresse 18 bei Digitaluhr
Adresse 20 bei D/A-Wandlerkarten
Adresse 22 bei Optokoppler-Eingangskarte
Adresse 24 bei Optokoppler-Ausgangskarte
Adresse 26 bei Relaiskarten
Adresse 28 bei Frequenz/Periodenkarte
Adresse 30 bei Frequenzzählerkarte

Eine Änderung dieser Voreinstellungen ist nur dann erforderlich, wenn sich bereits eine Karte mit dieser Adresse in Ihrem Rechner befindet.

Wenn Sie Ihren Rechner vor dem ersten Einsetzen der Karte einschalten, so zeigt Ihnen der Rechner im Bildschirm an, welche Adressen belegt sind. Drücken Sie nach dem Einschalten direkt die 'Space-Taste' des hp so bleibt die Bildschirmliste stehen.

Eine Änderung dieser Adressen erfolgt an den DIP-Schaltern 1-4.

ADRESSE	SCHALTER 1	2	3	4
8	on	on	off	on
10	off	on	off	on
12	on	off	off	on
14	off	off	off	on
16	on	on	on	off
18	off	on	on	off
20	on	off	on	off
22	off	off	on	off
24	on	on	off	off
26	off	on	off	off
28	on	off	off	off
30	off	off	off	off

2.0 PROGRAMMIERUNG DER OPTO-KOPPLER-AUSGÄNGE

NAME DES CSUB -PROGRAMMES >>>> OPTO_OUT
NAME DES BASIC-PROGRAMMES >>>> HP200_OUT

FORM DES AUFRUFES :
Opto_out (Select_code,Kanal,Ein_aus)

Select_code - INTEGERVARIABLE
Interfaceadresse der Karte. 24 ist voreingestellt.

Kanal - INTEGERVARIABLE
Bezeichnet den für die Ausgabe gewünschten Kanal 1 bis 16. Wird der Wert 0 übergeben, so werden alle Kanäle gleichzeitig zurückgesetzt.

Ein_aus - INTEGERVARIABLE
Beim Aufruf des CSUB bedeutet 0=Kanal Reset und 1=Kanal Setzen. Bei der Rückkehr hat das CSUB in diese Variable den Status aller Ausgänge geladen. Mit der BIT-Funktion des Rechners kann jeder Ausgangsstatus ermittelt werden.

BEISPIEL 1.... Laden und starten des BASIC-Programmes.

```
LOAD "HP200_OUT"
RUN
```

BEISPIEL 2.... Laden und Handhabung des CSUB.

```
LOAD "OPTO_OUT"
EDIT
```

Im Bildschirm finden Sie nur das CSUB-Programm. Drücken Sie nun bitte die Taste 'Zeile +' des Rechners und geben Sie die nachfolgenden Zeilen ein.

```
1  INTEGER Select_code,Kanal,Ein_aus
2  Select_code=24
3  Kanal=0
4  Opto_out (Select_code,Kanal,Ein_aus)
5  Kanal=1
6  Ein_aus=1
7  Opto_out (Select_code,Kanal,Ein_aus)
8  END
10 CSUB Opto_out (Select_code,Kanal,Ein_aus)
```

Die notwendigen Variablen werden zu INTEGER erklärt. Kanal=0 und anschließender Aufruf des CSUB setzt alle Kanäle zurück. Zum Schluß wird der Kanal 1 gesetzt und das Programm beendet.

```
10     !
20     !    P R O G R A M M
30     !    O P T O _ O U T      16 KANAL     HP 200/300
40     !
50     !                                              1987
60     !
70     !
80     !VERSION 2.0         DATE: 29.06.87  <HU>
90     !
100    !RE-STORE"HP200_OUT"
110    !
120            OUTPUT KBD;" #SCRATCH KEY  ( E";
130            INTEGER Select_code,Kanal,Ein_aus
140            Select_code=24                              ! OPTO_OUT-KARTE
150            Opto_out(Select_code,Kanal,Ein_aus)
160           !
170           !
180 Lesen:     !   SETZEN DER AUSGAENGE
190           !
200           !
210            PRINT TABXY(4,2),"P R O G R A M M          'OPTO_OUT'     1 6 - K A
N A L
220
230            PRINT TABXY(31,8),CHR$(129);" STATUS DER AUSGAENGE ";CHR$(128)
240            PRINT TABXY(10,10),"--1---2---3---4---5---6---7---8---9--10--11--1
2--13--14--15--16"
250            PRINT TABXY(9,11),RPT$("-",65)
260            PRINT TABXY(9,13),RPT$("-",65)
270            PRINT TABXY(13,15),"1 = AUSGANG AKTIV                    0 = AUSGANG
 INAKTIV"
280            PEN 1
290            GRAPHICS ON
300            GCLEAR
310            VIEWPORT 0,128,0,100
320            FRAME
330            VIEWPORT 1,127,1,99
340            FRAME
350            VIEWPORT 1,127,85,99
360            FRAME
370            VIEWPORT 1,127,84,99
380            FRAME
390            !
400            Ein_aus=0
410 Set:       PRINT TABXY(12,12),"";
420            IF Set_reset=0 THEN
430                 Ein_aus=1
440            ELSE
450                 Ein_aus=0
460            END IF
470            Set_reset=Ein_aus
480            !
490            FOR Kanal=1 TO 16
500                 Ein_aus=Set_reset
510                 Opto_out(Select_code,Kanal,Ein_aus)
520                 DISP "STATUS IN 'Ein_aus'  = ";Ein_aus
530                 WAIT .08
540                 IF Set_reset=0 THEN
550                      PRINT "0   ";
560                 ELSE
570                      PRINT "1   ";
580                 END IF
590            NEXT Kanal
600            GOTO Set
610           !
620           !
630            END !
640           !
650           !
660            CSUB Opto_out(INTEGER Select_code,Kanal,Ein_aus)
```

```
10      !
20      !    P R O G R A M M
30      !    O P T O I N P U T     16-KANAL  HP 200/300
40      !
50      !                                          1987
60      !
70      !
80      !VERSION 2.0         DATE: 29.06.87  <HU>
90      !
100     !RE-STORE"HP200_IN"
110     !
120             OUTPUT 2 USING "#,K";" K"
130             INTEGER Select_code,Opto_in
140             Select_code=22                          ! OPTO-IN-KARTE
150           !
160           !
170 Lesen:    !  LESEN DER EINGAENGE
180           !
190           !
200             PRINT TABXY(4,2),"P R O G R A M M         'OPTO_IN'     1 6 - K A
N A L
210               !
220             PRINT TABXY(31,8),CHR$(129);" STATUS DER EINGAENGE ";CHR$(128)
230             PRINT TABXY(10,10),"--1---2---3---4---5---6---7---8---9--10--11--1
2--13--14--15--16"
240             PRINT TABXY(9,11),RPT$("-",65)
250             PRINT TABXY(9,13),RPT$("-",65)
260             PRINT TABXY(13,15),"0 = EINGANG AKTIV                   1 = EINGANG
 INAKTIV"
270             PEN 1
280             GRAPHICS ON
290             GCLEAR
300             VIEWPORT 0,128,0,100
310             FRAME
320             VIEWPORT 1,127,1,99
330             FRAME
340             VIEWPORT 1,127,85,99
350             FRAME
360             VIEWPORT 1,127,84,99
370             FRAME
380             !
390 Read:       Opto_in(Select_code,Opto_in)
400             !
410             PRINT TABXY(11,12)," ";
420             FOR Kanal=0 TO 15
430                  IF BIT(Opto_in,Kanal)=1 THEN
440                       PRINT "1   ";
450                  ELSE
460                       PRINT "0   ";
470                  END IF
480             NEXT Kanal
490             GOTO Read
500           !
510           !
520             END !
530           !
540           !
550             CSUB Opto_in(INTEGER Select_code,Opto_in)
```

4.0 PROGRAMMIERUNG DER RELAISKARTE

NAME DES CSUB -PROGRAMMES >>>> RELAIS
NAME DES BASIC-PROGRAMMES >>>> HP200_REL

FORM DES AUFRUFES :
Relais (Select_code,Relais,Ein_aus)

Select_code - INTEGERVARIABLE
Interfaceadresse der Karte. 26 ist voreingestellt.

Relais - INTEGERVARIABLE
Bezeichnet das für die Ausgabe gewünschten Relais 1 bis 16. Wird der Wert 0 übergeben, so werden alle Relais gleichzeitig zurückgesetzt.

Ein_aus - INTEGERVARIABLE
Beim Aufruf des CSUB bedeutet 0=Relais Reset und 1=Relais Setzen. Bei der Rückkehr hat das CSUB in diese Variable den Status aller Relais geladen. Mit der BIT-Funktion des Rechners kann jeder Relaisstatus ermittelt werden.

BEISPIEL 1.... Laden und starten des BASIC-Programmes.

```
LOAD "HP200_REL"
RUN
```

BEISPIEL 2.... Laden und Handhabung des CSUB.

```
LOAD "RELAIS"
EDIT
```

Im Bildschirm finden Sie nur das CSUB-Programm. Drücken Sie nun bitte die Taste 'Zeile +' des Rechners und geben Sie die nachfolgenden Zeilen ein.

```
1  INTEGER Select_code,Relais,Ein_aus
2  Select_code=26
3  Relais=0
4  Opto_out (Select_code,Relais,Ein_aus)
5  Relais=4
6  Ein_aus=1
7  Opto_out (Select_code,Relais,Ein_aus)
8  END
10 CSUB Relais (Select_code,Kanal,Ein_aus)
```

Die notwendigen Variablen werden zu INTEGER erklärt. Relais=0 und anschließender Aufruf des CSUB setzt alle Relais zurück. Zum Schluß wird das Relais 4 gesetzt und das Programm beendet.

16 Relais pro Einschubkarte mit Wechslerkontakten.

Spannung an Kontakten	:	max. 50 V
Strom an Kontakten	:	max. .5 A
Schaltleistung	:	max 25 W

Pinbelegung des 50. poligen Steckers

****** RELAIS KARTENANSCHLUESSE ******			
RELAIS	PIN-NUMMER	RELAIS	PIN-NUMMER
R 9	15	S 1	50
A 9	14	A 1	48
S 9	16	R 1	49
R 10	12	S 2	47
A 10	11	A 2	45
S 10	13	R 2	46
R 11	5	S 3	41
A 11	6	A 3	39
S 11	4	R 3	40
R 12	2	S 4	44
A 12	3	A 4	42
S 12	1	R 4	43
R 13	30	S 5	38
A 13	31	A 5	36
S 13	29	R 5	37
R 14	27	S 6	25
A 14	28	A 6	23
S 14	26	R 6	24
R 15	8	S 7	22
A 15	9	A 7	20
S 15	7	R 7	21
R 16	33	S 8	19
A 16	34	A 8	17
S 16	32	R 8	18
----	10	----	35

```
10     !
20     !    P R O G R A M M
30     !    R E L A I S   1 6 - K A N A L   HP 200/300
40     !
50     !                                          1987
60     !
70     !
80     !
90     !    VERSION 2.0        DATE: 29.06.87  <HU>
100    !
110    !    RE-STORE"HP200_REL"
120    !
130             OUTPUT KBD;" #SCRATCH KEY ( E";
140             INTEGER Select_code,Relais,Ein_aus
150             Select_code=26                          ! RELAISKARTE
160             Relais(Select_code,Relais,Ein_aus)
170           !
180           !
190 Lesen:    !   SETZEN DER AUSGAENGE
200           !
210           !
220             PRINT TABXY(4,2),"P R O G R A M M         R E L A I S   1 6 - K A
N A L
230               !
240             PRINT TABXY(31,8),CHR$(129);" STATUS DER AUSGAENGE ";CHR$(128)
250             PRINT TABXY(10,10),"--1---2---3---4---5---6---7---8---9--10--11--1
2--13--14--15--16"
260             PRINT TABXY(9,11),RPT$("-",65)
270             PRINT TABXY(9,13),RPT$("-",65)
280             PRINT TABXY(13,15),"1 = AUSGANG AKTIV                    0 = AUSGANG
 INAKTIV"
290             PEN 1
300             GRAPHICS ON
310             GCLEAR
320             VIEWPORT 0,128,0,100
330             FRAME
340             VIEWPORT 1,127,1,99
350             FRAME
360             VIEWPORT 1,127,85,99
370             FRAME
380             VIEWPORT 1,127,84,99
390             FRAME
400             !
410             Ein_aus=0
420 Set:        PRINT TABXY(12,12),"";
430             IF Ein_aus=0 THEN
440                  Ein_aus=1
450             ELSE
460                  Ein_aus=0
470             END IF
480             Set_reset=Ein_aus
490             !
500             FOR Relais=1 TO 16
510                  Ein_aus=Set_reset
520                  Relais(Select_code,Relais,Ein_aus)
530                  DISP "STATUS IN 'Ein_aus'  = ";Ein_aus
540                  WAIT .08
550                  IF Set_reset=0 THEN
560                       PRINT "0   ";
570                  ELSE
580                       PRINT "1   ";
590                  END IF
600             NEXT Relais
610             GOTO Set
620           !
630           !
640           END !
650           !
660           CSUB Relais(INTEGER Select_code,Relais,Ein_aus)
```

5.0 PROGRAMMIERUNG DES FREQUENZ- UND IMPULSGENERATORS

NAME DES CSUB -PROGRAMMES >>>> FREQ_GEN
NAME DES BASIC-PROGRAMMES >>>> HP200_GEN

FORM DES AUFRUFES :
Freq_gen (Select_code,Mode$,Low,High)

Select_code - INTEGERVARIABLE
Interfaceadresse der Karte. 28 ist voreingestellt.

Mode$ - STRINGVARIABLE
Die gewünschte Betriebsart wird mittels der Stringvariablen Mode$ übergeben. Folgende Möglichkeiten sind vorhanden :

```
Mode$="F1"    ---  Betrieb  Kanal 1  als Frequenzgenerator
Mode$="F2"    ---  Betrieb  Kanal 2  als Frequenzgenerator
Mode$="P1"    ---  Betrieb  Kanal 1  als Impulsgenerator
Mode$="P2"    ---  Betrieb  Kanal 2  als Impulsgenerator
Mode$="E1"    ---  Ausgabe  Kanal 1  beenden
Mode$="E2"    ---  Ausgabe  Kanal 2  beenden
```

Low und High - INTEGERVARIABLE
Die letztlich möglichen Werte zum Laden der Karte im Bereich 4...65 530 werden als Low- und Highbyte an das CSUB übergeben. Die Ermittlung der Low- und Highwerte entnehmen Sie bitte dem beigefügten Programm.

BEISPIEL 1.... Laden und starten des BASIC-Programmes.

```
LOAD "HP200_GEN"
RUN
```

BEISPIEL 2.... Laden und Handhabung des CSUB.

```
LOAD "FREQ_GEN"
EDIT
```

Im Bildschirm finden Sie nur das CSUB-Programm. Drücken Sie nun bitte die Taste 'Zeile +' des Rechners und geben Sie die nachfolgenden Zeilen ein.

```
1  INTEGER Select_code,Low,High)
2  Select_code=28
3  Mode$="F1"
4  Frequenz=20
5  Wert=1/Frequenz/.00001
6  High=Wert DIV 256
7  Low=Wert-High*256.0
8  Freq_gen (Select_code,Mode$,Low,High)
9  END
10 CSUB Freq_gen (Select_code,Mode$,Low,High)
```

Die notwendigen Variablen werden zu INTEGER erklärt. Es wird eine Frequenz von 20 Hz an kanal 1 ausgegeben.

5.1 TECHNISCHE DATEN

Es stehen zwei unabhängige Kanäle pro Karte mit jeweils zwei Betriebsarten zur Verfügung.

1) FREQUENZGENERATOR	Die Periodendauer einer Ausgabefrequenz kann für jeden der beiden Kanäle separat eingestellt werden.
2) IMPULSGENERATOR	Von jedem Kanal kann ein einmaliger in der Zeitdauer variabler Impuls ausgegeben werden.

An der Frontplatte befinden sich zwei BNC-Buchsen für die jeweiligen Kanäle.

Die Ausgänge entsprechen Standart-TTL Pegeln.

Die Ausgangsfrequenz kann im Bereich von 1.53 Hz bis 25 KHz variiert werden. Entsprechend 40µs bis 655.35 ms Periodendauer.
Die kleinstmögliche Schrittweite entspricht 10µs.

Ein einzelner Ausgangsimpuls kann ebenfalls im Bereich von 40µs bis 655.35 ms gewählt werden.
Die kleinstmögliche Schrittweite beträgt ebenfalls 10µs.

```
10      !
20      !  P R O G A M M
30      !  F R E Q U E N Z G E N E R A T O R  HP 200/300
40      !
50      !
60      !
70      !VERSION 2.0         DATE: 29.06.87  <HU>
80      !
90      !RE-STORE"HP200_GEN"
100     !
110             OUTPUT KBD;" # (SCRATCH KEY E";
120             INTEGER Select_code,Low,High
130             Select_code=28                                  ! FREQUENZ_GENERATOR
140            !
150             PRINT TABXY(4,2),"P R O G R A M M              'FREQUENZ_GENERATOR-KART
E'
160               !
170             PRINT TABXY(16,9),CHR$(129);" BITTE WAEHLEN SIE DIE FUNKTION MIT D
EN KEY-TASTEN ";CHR$(128)
180             PRINT TABXY(9,11),RPT$("-",65)
190             PRINT TABXY(9,13),RPT$("-",65)
200             PEN 1
210             GRAPHICS ON
220             GCLEAR
230             VIEWPORT 0,128,0,100
240             FRAME
250             VIEWPORT 1,127,1,99
260             FRAME
270             VIEWPORT 1,127,85,99
280             FRAME
290             VIEWPORT 1,127,84,99
300             FRAME
310             !
320             ON KEY 1 LABEL "FREQUENZ 1" GOTO Freq_1
330             ON KEY 2 LABEL "IMPULS   1" GOTO Peri_1
340             ON KEY 3 LABEL "ENDE     1" GOTO Ende_1
350             ON KEY 5 LABEL "FREQUENZ 2" GOTO Freq_2
360             ON KEY 6 LABEL "IMPULS   2" GOTO Peri_2
370             ON KEY 7 LABEL "ENDE     2" GOTO Ende_2
380             !
390     Warte:  GOTO Warte
400             !
410     Ausgabe: Freq_gen(Select_code,Mode$,Low,High)
420             GOTO Warte
430             !
440     Freq_1: GOSUB F_zeit
450             Mode$="F1"
460             GOTO Ausgabe
470             !
480     Freq_2: GOSUB F_zeit
490             Mode$="F2"
500             GOTO Ausgabe
510             !
520     Peri_1: GOSUB P_zeit
530             Mode$="P1"
540             GOTO Ausgabe
550             !
560     Peri_2: GOSUB P_zeit
570             Mode$="P2"
580
```

```
590             GOTO Ausgabe
600             !
610 Ende_1:     Mode$="E1"
620             PRINT TABXY(20,12)," AUSGABE AN AUSGANG 1 BEENDET                    "
630             GOTO Ausgabe
640             !
650 Ende_2:     Mode$="E2"
660             PRINT TABXY(20,12)," AUSGABE AN AUSGANG 2 BEENDET                    "
670             GOTO Ausgabe
680             !
690 F_zeit:     INPUT "BITTE GEBEN SIE DIE FREQUENZ IN [Hz] EIN  (  >1.53Hz .... <
=25 000Hz  )."
,Freq
700             IF Freq<1.53 OR Freq>25000 THEN
710                  DISP "FALSCHE EINGABE"
720                  BEEP
730                  WAIT 2
740                  GOTO F_zeit
750             END IF
760             Wert=1/Freq/.00001
770             High=Wert DIV 256
780             Low=Wert-High*256.0
790             PRINT TABXY(20,12)," AUSGABE FREQUENZ = ";1/((Low+High*256.0)*.000
01);"  [Hz]                "
800             RETURN
810             !
820 P_zeit:     INPUT "BITTE GEBEN SIE DIE IMPULSDAUER [ms]  EIN (  >=0.04ms ....
<=655.35ms   )."
,Periode
830             IF Periode<.04 OR Periode>655.35 THEN
840                  DISP "FALSCHE EINGABE"
850                  BEEP
860                  WAIT 2
870                  GOTO P_zeit
880             END IF
890             Wert=Periode/.01
900             High=Wert DIV 256
910             Low=Wert-High*256.0
920             PRINT TABXY(20,12)," IMPULSLAENGE IST = ";(Low+High*256.0)*.01;"
[ms]                 "
930             RETURN
940             !
950             !
960             END !
970             !
980             !
990             CSUB Freq_gen(INTEGER Select_code,Mode$,INTEGER Low,High)
```

6.0 FREQUENZZÄHLERKARTE

Die Frequenzzählerkarte besitzt insgesamt 4 Eingänge. Ein Eingangspaar für TTL-Signale (1A und 2A), sowie ein Eingangspaar mit Eingangskomparatoren (1B und 2B). Der Eingang A kann für TTL-Signale bis 10 MHz benutzt werden. Der Eingang B besitzt hingegen einen Triggerverstärker der mittels des Potentiometers auf der Frontplatte eingestellt werden kann. An diesem Eingang können Signale von ca. 50 mV bis 15 V verarbeitet werden, wobei Frequenzen bis 2 Mhz gemessen werden können.Die Eingänge 2A und 2B sind lediglich notwendig für Frequenzverhältnis- sowie Intervallzeitmessungen.

6.1 TECHNISCHE DATEN DER FREQUENZZÄHLERKARTE

Eingänge A	:	für TTL-Signale. Schmitt - Trigger - Eingänge
Eingangsfrequenz	:	10 MHz maximal
Eingänge B	:	Triggerpunkt einstellbar von 0 - 5 V mittels Potentiometer an Frontplatte. Eingangsbereich von 50 mV bis 15 V.
Eingangsfrequenz	:	2 MHz maximal

6.2 ANSCHLUSSPLAN DER FREQUENZZÄHLERKARTE

****** FREQUENZZÄHLER - EINGÄNGE ******			
Kanal 1 A Kanal 1 B	BNC BNC	Kanal 2 A Kanal 2 B	BNC BNC

6.3 PROGRAMMIERUNG DER FREQUENZZÄHLERKARTE

NAME DES CSUB -PROGRAMMES >>>> FREQ_Z
NAME DES BASIC-PROGRAMMES >>>> HP200_FREQ

FORM DES AUFRUFES :

```
Freq_z (Select_code,Mode$,Flanke$,Eingang$,
          T_mm"24",Result$)
```

Select_code - INTEGERVARIABLE
Interfaceadresse der Karte. 30 ist voreingestellt.

Mode$ - STRINGVARIABLE
Die gewünschte Betriebsart wird mittels der Stringvariablen Mode$ übergeben. Folgende Möglichkeiten sind vorhanden :

```
Mode$="FREQUENZ"
Mode$="PERIODE"
Mode$="CRESET"                  Reset Counter zu Null
Mode$="COUNTER"                 Aktuellen Zählerstand lesen
Mode$="FREQUENZVERHÄLTNIS"      (Ratio)
Mode$="INTERVALLZEIT"
Mode$="DREHZAHL"
```

Flanke$ - STRINGVARIABLE
Festlegung, ob eine Messung auf der positiven oder negativen Flanke erfolgen soll. Eingang B ist nur für Intervallzeit und Frequenzverhältnis von Interresse.

```
Flanke$="PP" -Eingang 1 ist positiv . Eingang 2 ist positiv
Flanke$="PN" -Eingang 1 ist positiv . Eingang 2 ist negativ
Flanke$="NP" -Eingang 1 ist negativ . Eingang 2 ist positiv
Flanke$="NN" -Eingang 1 ist negativ . Eingang 2 ist negativ
```

Eingang$ - STRINGVARIABLE
Festlegung, welches Eingangspaar selektiert werden soll.

```
Eingang$="A" -TTL Eingänge sind selektiert
Eingang$="B" -Komparatoreingang mit einstellbarer Trigger-
              schwelle
```

T_m$ - STRINGVARIABLE
Einstellung der Torzeit für die Frequenzmessung (10,100,1000, 10000ms).
Einstellung der Mittelungsrate bei allen anderen Betriebsarten mit Ausnahme des Counters (1,10,100,1000).

ACHTUNG
Die Eingabe von "10ms" oder "10M" ist richtig.
Die Eingabe von "10" oder "100" etc. ist falsch.
Dem numerischen Wert muss ein Buchstabe nachgestellt werden.

Result$ - STRINGVARIABLE
Das Messergebnis wird vom CSUB dem String 'Result$ zugeordnet.

6.4 BETRIEBSARTEN DER FREQUENZZÄHLERKARTE

FREQUENZMESSUNG : Einstellen der Variablen

Die Frequenzmessung erfolgt 8 stellig. Die Auflösung ist abhängig von der verwendeten Messzeit.

```
T_m$ = "10ms"     =   123456.78  kHz
T_m$ = "100ms"    =   12345.678  kHz
T_m$ = "1000ms"   =   1234.5678  kHz
T_m$ = "10000ms"  =   123.45678  kHz

Mode$="FREQUENZ"
Flanke$="PP"      ) Triggerung auf positive Flanke
Flanke$="NN"      ) Triggerung auf negative Flanke
Eingang$="A"      ) TTL-Eingänge selektiert
Eingang$="B"      ) Komparatoreingänge selektiert
```

PERIODENMESSUNG : Einstellen der Variablen

Die Ausgabe der Periodendauer eines Messsignales erfolgt ebenfalls 8 stellig. Das Messergebnis kann auch als Mittelwert über mehrere Perioden angefordert werden. Bei einer Periodendauer von z.B. 2 ms erfolgt die Ausgabe wie folgt :

```
T_m$ = "1"    =   über      1 Periode  / 2000.0    µs
T_m$ = "10"   =   über     10 Perioden / 2000.00   µs
T_m$ = "100"  =   über    100 Perioden / 2000.000  µs
T_m$ = "1000"=    über   1000 Perioden / 2000.0000 µs

Mode$="PERIODE"
Flanke$="PP"      ) Triggerung auf positive Flanke
Flanke$="NN"      ) Triggerung auf negative Flanke
Eingang$="A"      ) TTL-Eingänge selektiert
Eingang$="B"      ) Komparatoreingänge selektiert
```

IMPULSZÄHLER : Einstellen der Variablen

Bei der Ausgabe dieses Befehles wird die Zählerbetriebsart der Karte selektiert. Die Zählkapazität beträgt 8 Stellen.

```
T_m$ = "1"    =   ohne Bedeutung auch 10,100,1000,10000
                  zulässig.

Mode$="CRESET"    ) Setzen des Counters auf Null
Mode$="COUNTER"   ) Lesen des aktuellen Counterstandes
Flanke$="PP"      ) Triggerung auf positive Flanke
Flanke$="NN"      ) Triggerung auf negative Flanke
Eingang$="A"      ) TTL-Eingänge selektiert
Eingang$="B"      ) Komparatoreingänge selektiert
```

VERHÄLTNISMESSUNG: Einstellen der Variablen

Es wird das Frequenzverhältnis (Ratio) fKanal1 / fKanal2 ermittelt. Eine Mittelung über mehrere Perioden kann wieder selektiert werden. Wird z.B. an Eingang 1 eine Frequenz von 5 KHz und an Eingang 2 eine Frequenz von 2.5 KHz angelegt, so ergibt sich die folgende Ausgabe :

```
T_m$ ="1"     = über    1 Periode  an f1 /    2.
T_m$ ="10"    = über   10 Perioden an f1 /    2.0
T_m$ ="100"   = über  100 Perioden an f1 /    2.00
T_m$ ="1000"  = über 1000 Perioden an f1 /    2.000
```

```
Mode$="RATIO"
Flanke$="PP"     > Eingang 1=positiv   Eingang 2=positiv
Flanke$="PN"     > Eingang 1=positiv   Eingang 2=negativ
Flanke$="NP"     > Eingang 1=Negativ   Eingang 2=positiv
Flanke$="NN"     > Eingang 1=negativ   Eingang 2=negativ
Eingang$="A"     > TTL-Eingänge selektiert
Eingang$="B"     > Komparatoreingänge selektiert
```

INTERVALLZEIT : Einstellen der Variablen

Es wird die Zeit von der Flanke an Kanal 1 bis zur Flanke an Kanal 2 gemessen. Eine automatische Mittelung kann wieder gewählt werden. Wird dasselbe Signal an beide Eingänge angelegt, so kann mittels :

Flanke$="PN" **> Eingang 1=positiv Eingang 2=negativ**

der positive Teil eines periodischen Signales und mit

Flanke$="NP" **> Eingang 1=negativ Eingang 2=positiv**

der negative Teil gemessen werden.

```
T_m$ = "1"    =  über     1 Intervall  / 2000.0    µs
T_m$ = "10"   =  über    10 Intervalle / 2000.00   µs
T_m$ = "10"   =  über    10 Intervalle / 2000.00   µs
T_m$ = "1000" =  über  1000 Intervalle / 2000.0000 µs
```

```
Mode$="INTERVALL"
Flanke$="PP"     > Triggerung auf positive Flanke
Flanke$="NN"     > Triggerung auf negative Flanke
Eingang$="A"     > TTL-Eingänge selektiert
Eingang$="B"     > Komparatoreingänge selektiert
```

DREHZAHLMESSUNG : Einstellen der Variablen

Die Messung der Drehzahl wird mit Hilfe der Periodenmessung durchgeführt. Die Umrechnung zu einer Drehzahl erfolgt durch den Rechner. Eine automatische Mittelung kann gewählt werden.

```
T_m$ = "1"     =   über     1 Umdrehung
T_m$ = "10"    =   über    10 Umdrehungen
T_m$ = "100"   =   über   100 Umdrehungen
T_m$ = "1000"  =   über  1000 Umdrehungen

Mode$="DREHZAHL"
Flanke$="PP"     ) Triggerung auf positive Flanke
Flanke$="NN"     ) Triggerung auf negative Flanke
Eingang$="A"     ) TTL-Eingänge selektiert
Eingang$="B"     ) Komparatoreingänge selektiert
```

UMRECHNUNG PERIODE ZU DREHZAHL

```
Wert=VAL(Result$)
Drehzahl=60000000/Wert
```

ACHTUNG

Die zeitliche Dauer einer Messung ist bei den Betriebsarten :
Periodendauer, Frequenzverhältnis, Drehzahl und Intervallzeit von der Anzahl der gewählten Mittelungen und der Frequenz selbst abhängig.
Die Messzeit kann unter Umständen sehr lange sein. Wird eine dieser Betriebsarten angewendet, ohne das ein Messsignal anliegt, so 'hängt' das Programm, da die Karte kein Signal Messung beendet liefern kann.

BEISPIEL 1.... Laden und starten des BASIC-Programmes.

```
LOAD "HP200_FREQ"
RUN
```

BEISPIEL 2.... Laden und Handhabung des CSUB.

```
LOAD "FREQ_EDIT
```

Im Bildschirm finden Sie nur das CSUB-Programm. Drücken Sie nun bitte die Taste 'Zeile +' des Rechners und geben Sie die nachfolgenden Zeilen ein.

```
1  INTEGER Select_code
2  Select_code=30
3  ModeFREQUENZ"
4  Flanke$="PP"
5  Eingang$="A"
6  T_m$="1000ms"
7  Freq_z(Select_code,Mode$,Flanke$,Eingang24
"    ,T_m$,Result$)
8  PRINT Result$
9  END
10 CSUB Freq_z(Select_code,Mode$,Flanke$,
   Eingang$,T_m$,Result$)
```

Die notwendigen Variablen werden zu INTEGER erklärt. Es wird Frequenzmessung mit 1000 ms Torzeit und Triggerung auf positive Flanke selektiert. Die TTL-Eingänge werden eingestellt. Das Ergebnis der Messung wird in Zeile 8 auf dem Bildschirm ausgegeben.

```
10     !
20     !    P R O G R A M M
30     !    F R E Q U E N Z Z A E H L E R   HP 200/300
40     !
50     !                                                1987
60     !
70     !
80     !VERSION 2.0        DATE: 08.07.87  <HU>
90     !
100    !RE-STORE"HP200_FREQ"
110    !
120           OUTPUT KBD;" #SCRATCH KEY ( E";
130           CONTROL CRT,21;0
140           !
150           DIM A$[50]
160           INTEGER Select_code
170           Select_code=30                          ! FREQUENZ-KARTE
180          !
190           !
200           !  AUFBAU GRUNDBILD
210           !
220           !
230           PRINT TABXY(4,2),"P R O G R A M M       'F R E Q U E N Z Z ª H L E
 R'
240             !
250           PRINT TABXY(1,5),"    BETRIEBSART          EINGANG        FLANKE
   MESSZEIT         MITTELUNGEN"
260           PRINT TABXY(1,15),"     MOMENTANWERT         MAXWERT        MINWERT
    MITTELWERT       MESSUNGEN"
270           PEN 1
280           GRAPHICS ON
290           GCLEAR
300           VIEWPORT 0,128,0,100
310           FRAME
320           VIEWPORT 1,127,1,99
330           FRAME
340           VIEWPORT 1,127,85,99
350           FRAME
360           VIEWPORT 1,127,84,99
370           FRAME
380           VIEWPORT 1,127,76,99
390           FRAME
400           VIEWPORT 1,127,75.5,99
410           FRAME
420           VIEWPORT 1,127,49,99
430           FRAME
440           VIEWPORT 1,127,48,99
450           FRAME
460           VIEWPORT 1,127,38,99
470           FRAME
480           VIEWPORT 1,127,37.5,99
490           FRAME
500           VIEWPORT 1,127,25.5,99
510           FRAME
520           VIEWPORT 1,127,25.0,99
530           FRAME
540           VIEWPORT 1,37.5,25.5,84
550           FRAME
560           VIEWPORT 1,37,25.5,84
570           FRAME
580           VIEWPORT 1,56,25.5,84
590           FRAME
600           VIEWPORT 1,56.5,25.5,84
610           FRAME
620           VIEWPORT 1,77.5,25.5,84
630           FRAME
640           VIEWPORT 1,78,25.5,84
650           FRAME
660           VIEWPORT 1,101,25.5,84
670           FRAME
680           VIEWPORT 1,101.5,25.5,84
690           FRAME
700           !
710           !
```

```
720             ! SET DEFAULT WERTE
730             !
740             !
750             Mode$="FREQUENZ"
760             Dimen$="KHz"
770             Mode=1
780             Flanke$="PP"
790             Flanke_1=1
800             Flanke_2=4
810             Eingang$="A"
820             Eingang=1
830             Tor$="1000E"
840             Tor=3
850             Mes$="1E"
860             Mes=1
870             Result$=" "
880             !
890 Neu_mess:   GOSUB Bild
900             BEEP
910 Weiter:     GOSUB Neu
920             DISP
930             !
940 Next:       DISABLE
950             Freq_z(Select_code,Mode$,Flanke$,Eingang$,T_m$,Result$)
960             Wert=VAL(Result$)
970             IF Wert=0 AND Mode$<>"COUNTER" THEN
980                  DISP "ACHTUNG, KEINE EINGANGSFREQUENZ"
990                  BEEP 3500,.01
1000            ELSE
1010                 DISP
1020            END IF
1030            IF Mode$="DREHZAHL" THEN
1040                 IF Wert>0 THEN
1050                      Wert=60000000/Wert            !DA MIKROSEKUNDEN
1060                 ELSE
1070                      GOTO Next
1080                 END IF
1090            END IF
1100            IF Wert>Max THEN Max=Wert
1110            IF Wert<Min THEN Min=Wert
1120            Mean=Mean+Wert
1130            Messungen=Messungen+1
1140            ENABLE
1150            PRINT TABXY(2,18),DROUND(Wert,9);" [";Dimen$;"]         "
1160            PRINT TABXY(23,18),DROUND(Max,9);"       "
1170            PRINT TABXY(37,18),DROUND(Min,9);"       "
1180            PRINT TABXY(51,18),DROUND(Mean/Messungen,9);"      "
1190            PRINT TABXY(75,18),Messungen
1200            GOTO Next
1210            !
1220 Neu:       Max=0                              !RESET F+R NEUE MESSUNG
1230            Min=99999999
1240            Mean=0
1250            Messungen=0
1260            PRINT TABXY(1,18),RPT$(" ",78)
1270            IF Mode$="COUNTER" THEN
1280                 Mode$="CReset"
1290                 Freq_z(Select_code,Mode$,Flanke$,Eingang$,T_m$,Result$)
1300                 Mode$="COUNTER"
1310            END IF
1320            !
1330 Keys:      FOR I=1 TO 10
1340                 ON KEY I LABEL " " GOSUB Dummy
1350            NEXT I
1360            ON KEY 1 LABEL "BETRIEB" GOTO Betrieb_n
1370            ON KEY 2 LABEL "EINGANG" GOTO Eingang_n
1380            ON KEY 3 LABEL "FLANKE_1" GOTO Flanke1_n
1390            ON KEY 4 LABEL "FLANKE_2" GOTO Flanke2_n
1400            IF Mode$="FREQUENZ" THEN
1410                 ON KEY 5 LABEL "MESSZEIT" GOTO Messz_n
1420            ELSE
1430                 IF Mode$<>"COUNTER" THEN ON KEY 6 LABEL "MITTLUNG" GOTO Mitte
l_n
1440            END IF
```

```
1450            ON KEY 8 LABEL "RESET    " GOTO Weiter
1460            RETURN
1470            !
1480 Dummy:     RETURN
1490            !
1500 Betrieb_n:Mode=Mode+1
1510            IF Mode=7 THEN Mode=1
1520            IF Mode=1 THEN
1530                 Mode$="FREQUENZ"
1540                 Dimen$="KHz"
1550            END IF
1560            IF Mode=2 THEN
1570                 Mode$="PERIODE"
1580                 Dimen$="µs"
1590            END IF
1600            IF Mode=3 THEN
1610                 Mode$="COUNTER"
1620                 Dimen$="IMPULSE"
1630            END IF
1640            IF Mode=4 THEN
1650                 Mode$="RATIO"
1660                 Dimen$="f1/f2"
1670            END IF
1680            IF Mode=5 THEN
1690                 Mode$="INTERVALLZEIT"
1700                 Dimen$="µs"
1710            END IF
1720            IF Mode=6 THEN
1730                 Mode$="DREHZAHL"
1740                 Dimen$="U/MIN"
1750            END IF
1760            GOTO Warten
1770               !
1780 Eingang_n:Eingang=Eingang+1
1790            IF Eingang=3 THEN Eingang=1
1800            IF Eingang=1 THEN Eingang$="A"          !TTL
1810            IF Eingang=2 THEN Eingang$="B"          !SCHWELLE
1820            GOTO Warten
1830               !
1840 Flanke1_n:Flanke_1=Flanke_1+1
1850            IF Flanke_1=3 THEN Flanke_1=1
1860            IF Flanke_1=1 THEN Flanke$[1,1]="P"
1870            IF Flanke_1=2 THEN Flanke$[1,1]="N"
1880            GOTO Warten
1890               !
1900 Flanke2_n:Flanke_2=Flanke_2+1
1910            IF Flanke_2=6 THEN Flanke_2=4
1920            IF Flanke_2=4 THEN Flanke$[2,2]="P"
1930            IF Flanke_2=5 THEN Flanke$[2,2]="N"
1940            GOTO Warten
1950               !
1960 Messz_n:  Tor=Tor+1
1970            IF Tor=5 THEN Tor=1
1980            IF Tor=1 THEN Tor$="10E"
1990            IF Tor=2 THEN Tor$="100E"
2000            IF Tor=3 THEN Tor$="1000E"
2010            IF Tor=4 THEN Tor$="10000E"
2020            GOTO Warten
2030               !
2040 Mittel_n:Mes=Mes+1
2050            IF Mes=5 THEN Mes=1
2060            IF Mes=1 THEN Mes$="1E"
2070            IF Mes=2 THEN Mes$="10E"
2080            IF Mes=3 THEN Mes$="100E"
2090            IF Mes=4 THEN Mes$="1000E"
2100            GOTO Warten
2110               !
2120 Warten:    BEEP
2130            DISP "WENN EINSTELLUNGEN BEENDET, BITTE TASTE <WEITER> DR+CKEN !!"
2140            GOSUB Bild
2150            GOSUB Keys
2160            ON KEY 8 LABEL "WEITER" GOTO Weiter
2170 Standby:   GOTO Standby
2180            !
```

```
2190 Bild:     !
2200           ! AKTUELLES BILD ZEICHNEN
2210           !
2220           !
2230           RESTORE Betrieb
2240           Invers=Mode
2250           Invers_2=0
2260           Wohin=1
2270           GOSUB Ausgabe
2280           RESTORE Eingang
2290           Invers=Eingang
2300           Wohin=24
2310           GOSUB Ausgabe
2320           RESTORE Messzeit
2330           Invers=0
2340           IF Mode$="FREQUENZ" THEN
2350                Invers=Tor
2360                T_m$=Tor$
2370           END IF
2380           Wohin=51
2390           GOSUB Ausgabe
2400           RESTORE Mittlung
2410           Invers=0
2420           IF Mode$<>"FREQUENZ" THEN
2430                Invers=Mes
2440                T_m$=Mes$
2450           END IF
2460           IF Mode$="COUNTER" THEN Invers=0
2470           Wohin=68
2480           GOSUB Ausgabe
2490           RESTORE Flanke
2500           Invers=Flanke_1
2510           Invers_2=Flanke_2
2520           Wohin=36
2530           GOSUB Ausgabe
2540           RETURN
2550           !
2560 Ausgabe:  READ Anzahl
2570           FOR I=1 TO Anzahl
2580                IF I=Invers OR I=Invers_2 THEN
2590                     Chr=129
2600                ELSE
2610                     Chr=128
2620                END IF
2630                READ A$
2640                PRINT TABXY(Wohin,I+6),CHR$(Chr);A$;CHR$(128)
2650           NEXT I
2660           RETURN
2670           !
2680 Betrieb: DATA 6," FREQUENZ           "," PERIODE            "," IMPULS-COUN
TER     "," FREQUENZVERHÄLTNIS "," INTERVALLZEIT      "," DREHZAHL           "
2690 Eingang: DATA 2," TTL       "," SCHWELLE "
2700 Flanke:  DATA 5," 1_POSITIV "," 1_NEGATIV "," ","2_POSITIV "," 2_NEGATIV "
2710 Messzeit:DATA 4,"      10 ms ","     100 ms ","  1 000 ms ","10 000 ms "
2720 Mittlung:DATA 4,"       1    ","      10    ","     100    ","  1 000    "
2730          !
2740          !
2750           END !
2760          !
2770          !
2780           CSUB Freq_z(INTEGER Select_code,Mode$,Flanke$,Eingang$,T_m$,Result
$)
```

7.0 PROGRAMMIERUNG DER DIGITALUHR

NAME DES CSUB -PROGRAMMES >>>> UHR
NAME DES BASIC-PROGRAMMES >>>> HP200_UHR

FORM DES AUFRUFES :
Uhr (Select_code,Mode$,Zeit$,Datum$,Tag$)

Select_code - INTEGERVARIABLE
Interfaceadresse der Karte. 18 ist voreingestellt.

Mode$ - STRINGVARIABLE
Die gewünschte Betriebsart wird mittels der Stringvariablen Mode$ selektiert.
Mode$="LESEN" - der Uhr Mode$="Setzen" - der Uhr

Zeit$ - STRINGVARIABLE
Uhrzeit in der Form : SS:MM:SS-Stunde:Minute:Sekunde
Zeit$="17:43:00"
Die Sekunden können beim Setzen nur zu '00' eingestellt werden.

Datum$ - STRINGVARIABLE
Beinhaltet das Datum in der Form :
TT:MM:JJ - Tag,Monat,Jahr
Datum$="30:12:86"

Tag$ - STRINGVARIABLE
Beinhaltet den Wochentag in der Form :
SONNTAG,MONTAG,DIENSTAG,MITTWOCH,DONNERSTAG,FREITAG,SAMSTAG
Tag$="MONTAG"

BEISPIEL 1.... Laden und starten des BASIC-Programmes.

```
LOAD "HP200_UHR"
RUN
```

BEISPIEL 2.... Laden und Handhabung des CSUB.

```
LOAD "UHR"
EDIT
```

Im Bildschirm finden Sie nur das CSUB-Programm. Drücken Sie nun bitte die Taste 'Zeile +' des Rechners und geben Sie die nachfolgenden Zeilen ein.

```
1  INTEGER Select_code
2  Select_code=18
3  ModeLESEN"
4  Uhr(Select_code,Mode$,Zeit$,Datum$,Tag$)
5  PRINT Zeit$,Datum$,Tag$
6  END
10 CSUB Uhr(Select_code,Mode$,Zeit$,Datum$,
   Tag$)
```

Die notwendigen Variablen werden zu INTEGER erklärt. Die Karte wird ausgelesen und Zeit, Datum und Tag angezeigt.

```
10     !
20     !    P R O G R A M M
30     !    D I G I T A L U H R              HP 200/300
40     !
50     !                                          1987
60     !
70     !
80     !VERSION 2.0        DATE: 07.07.87  <HU>
90     !
100    !RE-STORE"HP200_UHR"
110    !
120            OUTPUT KBD;" #SCRATCH KEY ( E";
130            INTEGER Select_code
140            Select_code=18                                  ! UHR-KARTE
150            ON KEY 1 LABEL "SETZEN" GOTO Set_uhr
160           !
170           !
180 Lesen:     !  LESEN DER UHR
190           !
200           !
210            PRINT TABXY(4,2),"P R O G R A M M                        'DIGITALUHR'

220            PRINT TABXY(9,9),"-------UHRZEIT----------DATUM--------------W
OCHENTAG---------"
230            PRINT TABXY(9,11),RPT$("",65)
240            PRINT TABXY(9,13),RPT$("",65)
250            PEN 1
260            GRAPHICS ON
270            GCLEAR
280            VIEWPORT 0,128,0,100
290            FRAME
300            VIEWPORT 1,127,1,99
310            FRAME
320            VIEWPORT 1,127,85,99
330            FRAME
340            VIEWPORT 1,127,84,99
350            FRAME
360            !
370 Read:      Mode$="LESEN"
380            Uhr(Select_code,Mode$,Zeit$,Datum$,Tag$)
390            PRINT TABXY(16,12),Zeit$
400            PRINT TABXY(37,12),Datum$
410            PRINT TABXY(56,12),Tag$;"                "
420            GOTO Read
430           !
440 Set_uhr:   BEEP
450            INPUT "EINGABE DER UHRZEIT z.b. : 16:58:00 >>>>",Zeit$
460            INPUT "EINGABE DES DATUMS  z.b. : 28:12:87 >>>>",Datum$
470            INPUT "EINGABE DES WOCHENTAGES z.B. : FREITAG >>>>",Tag$
480            INPUT "WENN SIE NUN DIE TASTE 'ENTER' DR+CKEN,WIRD DIE UHR GESTELL
T.....",A$
490            Mode$="SETZEN"
500            Uhr(Select_code,Mode$,Zeit$,Datum$,Tag$)
510            GOTO Read
520           !
530            END !
540           !
550           !
560            CSUB Uhr(INTEGER Select_code,Mode$,Zeit$,Datum$,Tag$)
```

```
580             !
590 Warten:     GOTO Warten
600             !
610 Da1:        !  K A N A L  1
620             !
630             Kanal=1
640             INPUT "EINGABE DES WERTES IN [mV] ",Wert
650             Dac_out(Select_code,Kanal,Wert)
660             PRINT TABXY(17,12),"ANALOGWERT AN KANAL 1 ..... ";Wert;" [mV]
    "
670             GOTO Warten
680             !
690 Da2:        !  K A N A L  2
700             !
710             !
720             Kanal=2
730             INPUT "EINGABE DES WERTES IN [mV] ",Wert
740             Dac_out(Select_code,Kanal,Wert)
750             PRINT TABXY(17,14),"ANALOGWERT AN KANAL 2 ..... ";Wert;" [mV]
    "
760             GOTO Warten
770             !
780             END
790             !
800             !
810             CSUB Dac_out(INTEGER Select_code,Kanal,Wert)
```

8.0 D/A-WANDLERKARTE

Die ausgebbare Analogspannung liegt in der Standardeinstellung der D/A-Karte im Bereich +/- 10.24 Volt. Durch das Umstecken einiger Brücken kann der Ausgabebereich auf 0...10.24 Volt geändert werden.

Zwei Versionen sind lieferbar :

a) 1 analoger Ausgabekanal
b) 2 analoge getrennte Ausgabekanäle

Das mitgelieferte CSUB erkennt automatisch um welche Karte in welchem Ausgabebereich es sich handelt.

Die D/A-Wandler besitzen eine Auflösung von 12 BIT. Beim Einschalten des Rechners nehmen die Analogausgänge Zufallswerte an.

Im Bereich +/- 10.24 Volt kann die negative Ausgangsspannung -9.9 Volt nicht überschreiten.

Dies ist dadurch bedingt, daß der HP200 lediglich +/- 12 Volt als Betriebsspannung für seine Einschubkarten zur Verfügung stellen kann.

8.1 EINSTELLUNG VON ZERO UND GAIN

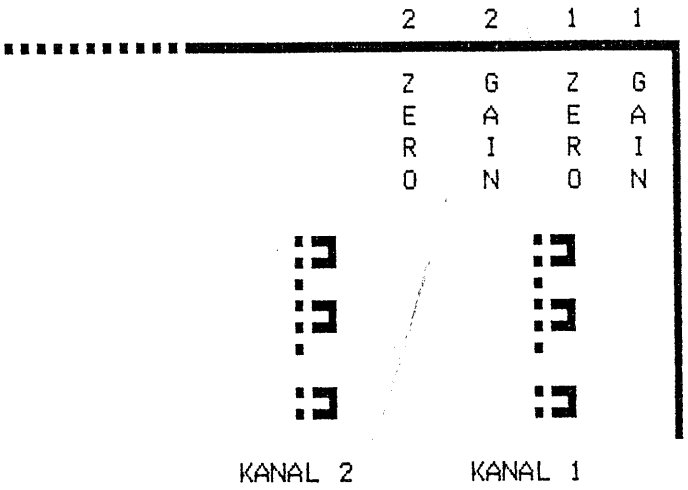
Geben Sie per Programm 0 Volt an den einzustellenden Kanal. Überprüfen Sie den ausgegeben Wert mittels eines Digitalvoltmeters. Stellen Sie den Nullpunkt gegebenenfalls mit dem Poti ZERO (Z) nach. Geben Sie nun per Programm 10 Volt aus. Stellen Sie gegebenenfalls den Ausgang auf 10.000 Volt mit Hilfe des Potis GAIN (G) ein. Wiederholen Sie diesen Abgleich mehrfach.

8.2 JUMPER EINSTELLUNGEN

Die Ausgabespannung kann von +/-10.24 Volt auf 0...10.24 Volt umgestellt werden.

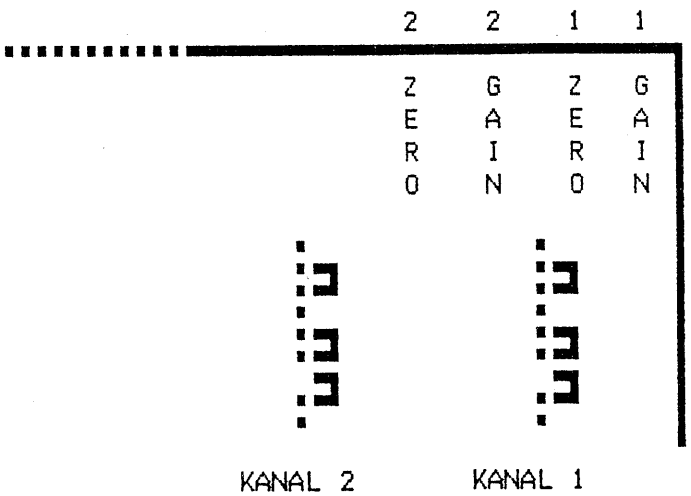
a) Einstellung 0...10.24 Volt

Schalter 8 des 8-reihigen DIL-Schalters der Karte auf 'ON'.



b) Einstellung +/- 10.24 Volt

Schalter 8 des 8-reihigen DIL-Schalters der Karte auf 'OFF'.



8.3 PROGRAMMIERUNG DER D/A-WANDLERKARTE

NAME DES CSUB -PROGRAMMES >>>> DAC_OUT
NAME DES BASIC-PROGRAMMES >>>> HP200_DA

FORM DES AUFRUFES :
Dac_out (Select_code,Kanal,Wert)

Select_code - INTEGERVARIABLE
Interfaceadresse der Karte. 20 ist voreingestellt.

Kanal - INTEGERVARIABLE
Bezeichnet den für die Ausgabe gewünschten Kanal 1 oder 2.

Wert - INTEGERVARIABLE
Der gewünschte Analogwert in Millivolt.

BEISPIEL 1.... Laden und starten des BASIC-Programmes.

```
LOAD "HP200_DA"
RUN
```

BEISPIEL 2.... Laden und Handhabung des CSUB.

```
LOAD "DAC_OUT"
EDIT
```

Im Bildschirm finden Sie nur das CSUB-Programm. Drücken Sie nun bitte die Taste 'Zeile +' des Rechners und geben Sie die nachfolgenden Zeilen ein.

```
1  INTEGER Select_code,Kanal,Wert
2  Select_code=20
3  Kanal=1
4  Wert=2500
4  Dac_out (Select_code,Kanal,Wert)
5  END
10 CSUB Dac_out (Select_code,Kanal,Wert)
```

Die notwendigen Variablen werden zu INTEGER erklärt. Auf dem Analogkanal 1 wird eine Spannung von 2 500 mV ausgebeben.

8.4 TECHNISCHE DATEN D/A-WANDLER

Auflösung	:	12 bit
Ausgangsspannung a)	:	-9.9...0...10.24 Volt
Ausgangsspannung b)	:	0...10.24 Volt
Ausgangsstrom	:	5 mA
Kleinster Schritt a)	:	2.50 mV
Kleinster Schritt b)	:	5.00 mV
Linearität	:	0.01%
Stabilität Gain	:	8ppm / °C F.S.R.
Stabilität Linearität	:	2ppm / °C F.S.R.
Stabilität Offset	:	8ppm / °C F.S.R.
Trimmpotis für	:	GAIN und ZERO

```
10      !
20      !  P R O G R A M M
30      !  D I G I T A L / A N AL O G  WANDLER   HP 200
40      !
50      !                                          1987
60      !
70      !
80      ! RE-STORE"HP200_DA"
90      !
100     !    VERSION 2.0         DATE: JUN.87     <HU>
110     !
120             OUTPUT KBD;" # K (SCRATCH KEY E";
130             INTEGER Select_code,Kanal,Wert
140             Select_code=20                          !DER DAC-KARTE
150             !
160             PRINT TABXY(4,2),"P R O G R A M M    D I G I T A L / A N A L O G -
 WANDLER
170             PRINT TABXY(9,11),RPT$("-",63)
180             PRINT TABXY(9,13),RPT$("-",63)
190             PRINT TABXY(9,15),RPT$("-",63)
200             PEN 1
210             GRAPHICS ON
220             GCLEAR
230             VIEWPORT 0,128,0,100
240             FRAME
250             VIEWPORT 1,127,1,99
260             FRAME
270             VIEWPORT 1,127,85,99
280             FRAME
290             VIEWPORT 1,127,84,99
300             FRAME
310             !
320             PRINT TABXY(17,6),"AUSGAENGE SIND ........ ";
330             Art=READIO(Select_code,3)
340             IF BIT(Art,5)=1 THEN
350                  PRINT "BIPOLAR  +/- 10.24 VOLT"
360             ELSE
370                  PRINT "UNIPOLAR  0..10.24 VOLT"
380             END IF
390             !
400             Kanal=1
410             Wert=0
420             Dac_out(Select_code,Kanal,Wert)
430             PRINT TABXY(17,12),"ANALOGWERT AN KANAL 1 ..... ";Wert;" [mV]
    "
440             !
450             ON KEY 1 LABEL "D/A KANAL 1" GOTO Da1
460             PRINT TABXY(17,7),"KANALZAHL IST  ........ ";
470             Channels=READIO(Select_code,1)
480             IF BIT(Channels,4)=1 THEN
490                  PRINT "1 KANAL-KARTE"
500             ELSE
510                  PRINT "2 KANAL-KARTE"
520                  ON KEY 8 LABEL "D/A KANAL 2" GOTO Da2
530                  Kanal=2
540                  Dac_out(Select_code,Kanal,Wert)
550                  PRINT TABXY(17,14),"ANALOGWERT AN KANAL 2 ..... ";Wert;" [mV
]       "
560             END IF
570             PRINT
```

7. Some Sample Cards

7.1. FZ 200/300

Frequency counter, max. 10 MHz

Sticker with Nº. G189
Four BNC connectors: 1A, 2A, 1B, 2B
Two trimmers K1, K2
Programmable peripheral interface: NEC D8255AD-2
Frequency counter: INTERSIL ICM7226BIPL

DIP Switch:

```
12345678
00001011    1 = ON
---OFF-
```

SW 1-4 correspond to SC 30 (factory default)

7.1.1. IN 200/300

16 digital inputs
16 Optocouplers IL74

DIP Switch:

```
12345678
00101111    1 = ON
---OFF--
```

SW 1-4 correspond to SC 22 (factory default)

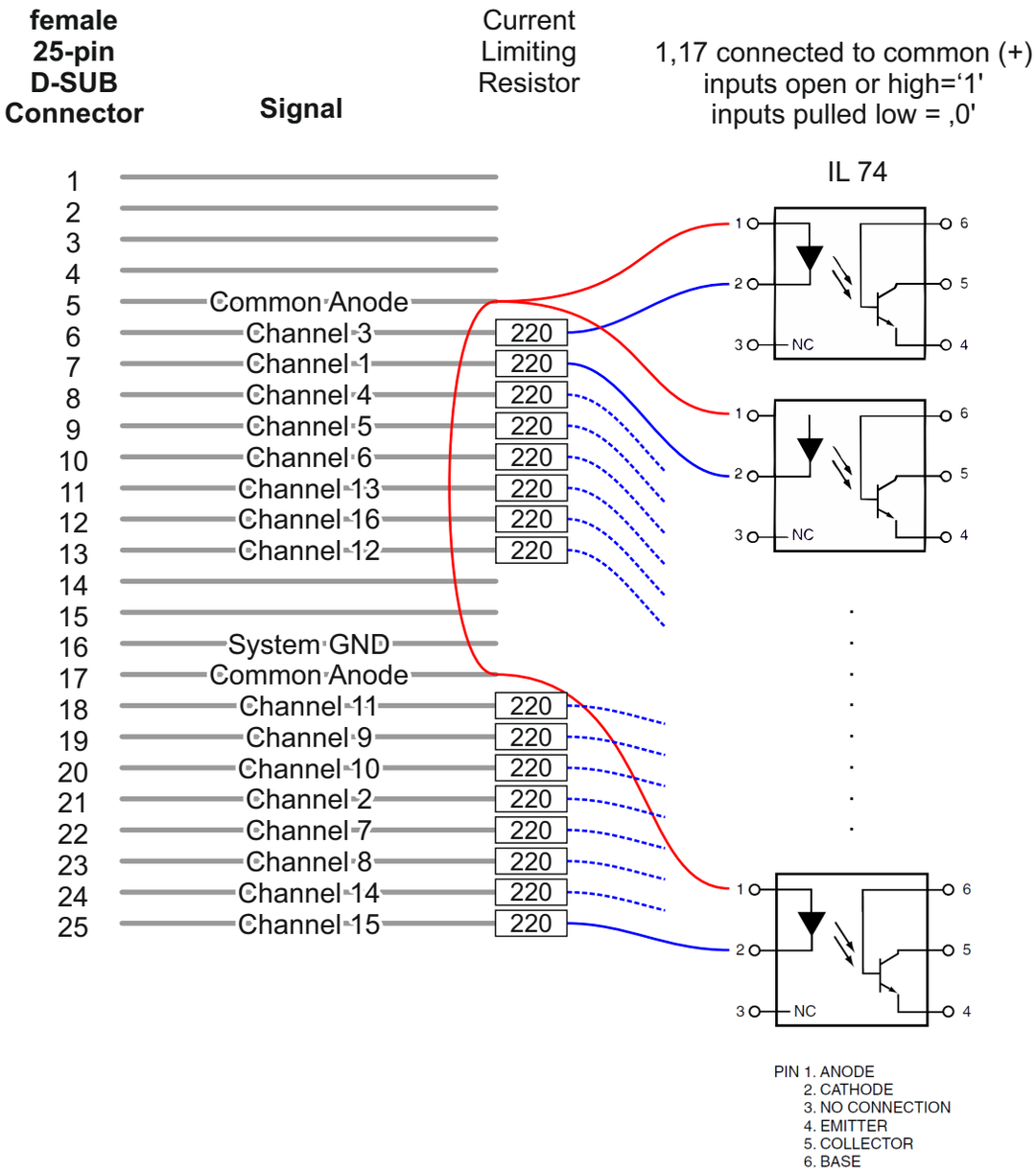
7.1.2. DU 200/300

Battery backed realtime clock
1 trim capacitor
Clock chip: OKI M5832
Backup Battery: Sanyo Cadnica, N-50SB3, 3.6V, 45mAh

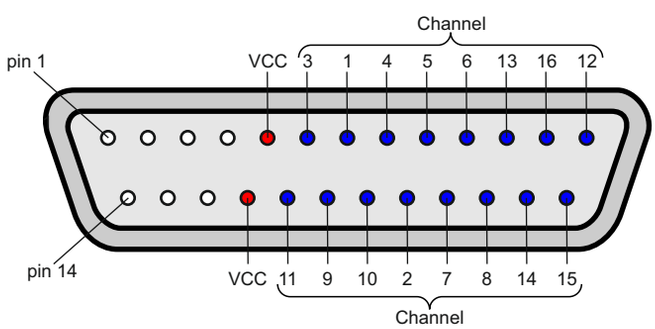
DIP Switch:

```
12345678
01101111    1 = ON
---OFF-
```

SW 1-4 correspond to SC 18 (factory default)



DB25 - male
view on connector pins



INFAX IN 200/300
Digital Input Interface

Martin Hepperle, 2017