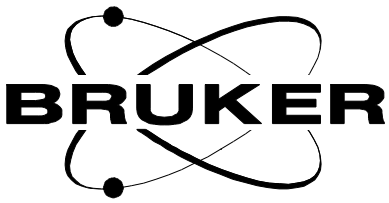


AQR 4 Phase Modulator

2 CH and 3 CH Version Technical manual

Version 005

BRUKER

The information in this manual may be altered without notice.

This manual was written by

Uwe Döttling

Rheinstetten, Germany

P/N: Z31225
DWG-Nr: 896005

Contents

Contents

Index

R

S

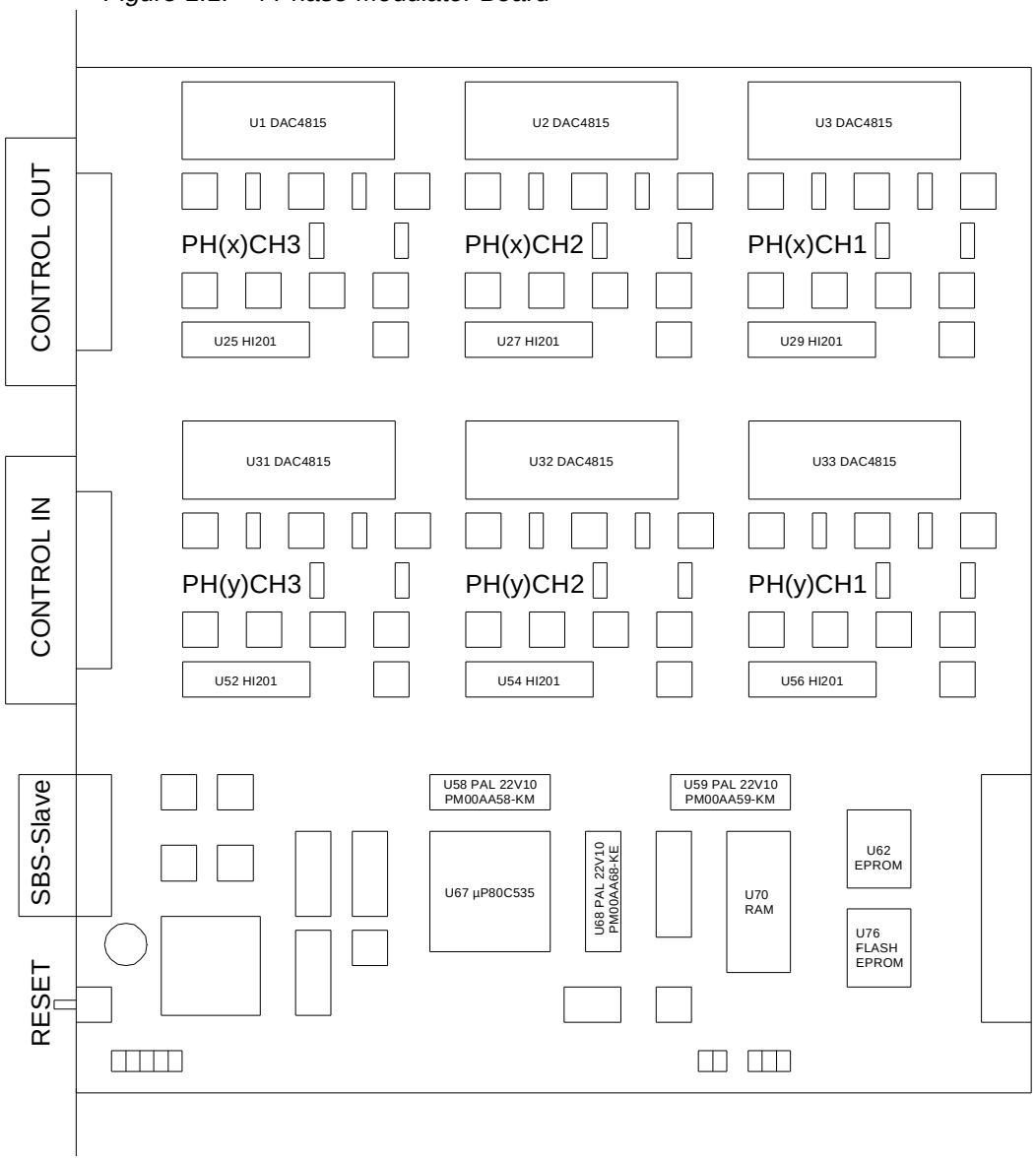
W

Overview

1

The AQR 4 Phase Modulator is designed to set phase and amplitude of the 3 channels of the SE451. Controll of all internal parameters and communication to the spectrometer is done via the HPCU (High Power Control Unit). Parameter changes can either be initiated by the HPCU keyboard or by spectrometer computer. The 4 Phase Modulator is linked by an ***SBS-Bus*** Interface to the HPCU. This enables future expansion to a second 4 Phase Modulator controlled by the same HPCU. The standard configuration is equipped for 2 channels (P/N H5649), but optionally it can be fitted for all three channels (P/N H5485).

Figure 1.1. 4 Phase Modulator Board



Hardware 2

General 2.1

The Hardware of the 4 Phase Modulator can be divided into two functional sections; the digital section, to communicate with the host system and control of all the internal parameters; and the analog section consisting of the output stages for the amplitude- and phase control.

Digitalsection 2.2

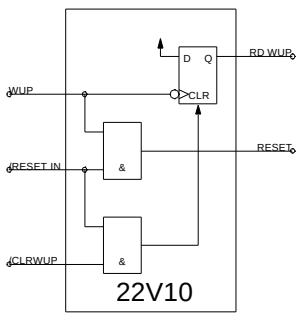
CPU 2.2.1

The CPU is the microcontroller 80C535 from Siemens (U67). For memory a OTProm 27265 (32 kByte) or 27512 (64 kByte) (U62) is used to store the ***boot and download firmware***; a Flash Eprom 28F256 (32 kByte) (U76) is used to store the ***application specific firmware*** and a RAM 62256 (32 kByte) (U70) to store parameters. The use of Flash Eproms enables downloading of new firmware even by the customer, without opening the unit. The microcontroller runs with a frequency of 12 MHz.

Reset and Wake Up signal 2.2.2

Generation of the reset signal is done inside the Pal 22V10 (U68) (see also Pal Listings).

Figure 2.1. Reset Logic



Both inputs /RESET IN and WUP (Wake Up) can generate a reset signal for the microcontroller. After a reset, the reset source can be read by the output RD-WUP

(read Wake Up). After reset by WUP the D-Flip-Flop inside the pal is set and the output RD-WUP is high. The Flip-Flop can be cleared by a signal from the microcontroller (CLRWUP) or by a reset activated by the reset controller TL7705 (U74). This enables a High at RD-WUP only after a Wake Up (Wake Up is a signal from ***SBS-BUS***, used to reactivate the unit after Sleep Mode).

Decoder *2.2.3*

The remainder of the pal logic is used to decode the Memory section. The memory map is dependant on the signal PGMODE (Program Mode) at Pin P1.4 of 80C535.

Table 2.1. Memory Decoder

PGMode	Eprom	Flash Eprom	Ram
0	not mapped	Code 0x0000 - 0xFFFF	Data 0x0000 - 0x7FFFF
1	Code 0x0000 - 0x7FFF	Data 0x0000 - 0x7FFF	not mapped

DAC Controling *2.2.4*

Control of the 6 DAC´s is not performed by address databus, but with 2 of the microcontroller ports. The data lines for the DAC´s are driven by Port 4 and the control lines by Port 5. To select the DAC´s, 3 port lines are decoded by a 3 to 8 multiplexer 74LS138 (U66). The lines WR (Write), LE (Latch Enable), A0, A1, A2 (Address 0, 1, 2) are driven by the remaining 5 port pins of Port5. The DAC input CLR (Clear) is driven directly from the reset signal to clear the Dac after a hardware reset (0x8000 = 0 volts).

I^2C E^2Prom *2.2.5*

A seriall E^2Prom X24C16 (U75) with I^2C bus-structure is used to store all BBIS (**B**ruker **B**oard **I**nformation **S**ystem) data and important system parameters (Setup, last configuration, etc.).

SBS-Bus Interface *2.2.6*

For communication with a host system, an SBS-Bus Interface is implemented. The interface is galvanically isolated. The supply for the interface drivers can be either a 9 volt supply voltage from the SBS Bus for a galvanic isolated interface (J5: connect pin 2 with pin 3, J6: no connection), or the boards internal supply (J5: connect pin 1 with pin 2, J6: pin1 to pin 2 connected) for a non isolated interface (mostly used during test or when interfaced to a PC using an RS232 to RS485 converter). The SBS Bus address can be selected by Jumper J4 (jumper set = address 1, jumper not set = address 0 = default setting). To use only one 4 Phase Modulator in a system, use address 0. To terminate the interface the jumpers J1, J2, J3 must be set in the last unit in the SBS Bus net.

Analogsection 2.3

DA-Converter 2.3.1

If totally populated, there are 6 Quad 12Bit DAC´s DAC 4815AP (U1, U2, U3, U31, U32, U33) from Burr Brown. Every DAC output is driven by an OP AD 846 to a specific level. Two Dac outputs are used to generate the Receiver Phase outputs (SIN and COS).

Analog switch 2.3.2

For each of the 3 channels of the 4 Phase Modulator there are the following outputs:

- Amplitude +Y (-150mV to -500mV) and Amplitude -Y(150mV to 500mV) - Phase +X (0V) and Phase -X (-150mV to +150mV) and - Amplitude +X (150mV to 500mV) and Amplitude -X(-150mV to -500mV) - Phase +Y (-150V to +150mV) and Phase -Y (-150mV to +150mV)

Phase +X is 0 volts in every channel (referencephase). 4 outputs of each channel build one functional block. They are switched with analog switches (HI201HS) to a common output, in that way, that amplitude and phase from the selected quadrant are switched to the outputs (for example: Amplitude +Y and Phase +Y or Amplitude -X and Phase -X). The analog switches are controlled by the spectrometer. Decoding of the pulses is done by two Pal´s 22V10 (U58 and U59) (see pal listing). Control of the analog switches can also be done by the microcontroller. Therefore the pin TESTEN (Test enable) must be driven low by the microcontroller, which is now able to select the analog switches using the pins TEST1 and TEST2. This is only necessary for test purposes.

Software 3

General 3.1

The software of the 4 Phase Modulator consists of two independant modules; the boot firmware and the application firmware.

3.1. Boot firmware 3.2

The boot firmware controls the reset sequence and downloading of new firmware releases. After initialization, it checks if there is a valid application firmware stored on the Flash Eprom. If all tests are good, it switches to the application program by driving pin PGMODE P1.4 to low. This causes the memory decoder now to use the Flash Eprom as the code memory and the Ram as a data memory. The application program can now run. While the boot program is running, the memory decoder maps the OTProm as code memory and the Flash Eprom as Data memory (only data memory can be written by microcontroller). As Ram, only the internal memory of the microcontroller is used.

3.2. Application firmware 3.3

All application specific functions for the 4 Phase Modulator are controlled by the application program. After a software reset, which is performed after switching from the boot program, the program tests if the reset was initiated by a Wake Up signal or by the reset controller by reading the output RDWUP from Pal 22V10 (U68). Depending on the reset source the hardware will be initiated. After a hardware reset, all DAC´s are loaded with the last value , stored in I^2C E^2Prom. If no reasonable value is stored, maybe after first power on or a defective E^2Prom, all DAC outputs are cleared to 0 volts. After reset by Wake Up, the DAC outputs are not modified, they hold on the value loaded before sleep modus. After initialization of the whole hardware the program goes to the main loop. The main loop handles the following program parts:

- Comand interpreter for all SBS Bus commands.
- Update of E^2Prom (every new output value will be stored on E^2Prom after a timeout of 10 seconds).
- Dac output setting.
- Watchdog timer refresh.
- Test routines if enabled.

Connector Pinout

4

Control Out 4.1

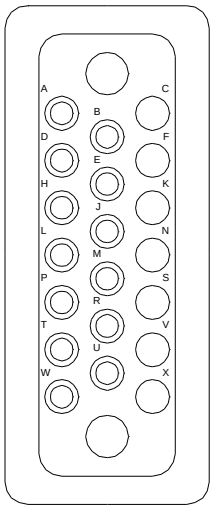


Table 4.1. Control Out Burndy 20 Pin RM

Pin No.	new Name	old Name	Connected to
A	-	-	-
B	-	-	-
C	-	-	-
D	-	-	-
E	-	-	-
F	-	-	-
H	PH(x)CH1	FX current x	TP 9
J	PH(x)CH2	FH current x	TP 7
K	-	-	-
L	PH(y)CH1	FX current y	TP 16
M	PH(y)CH2	FH current y	TP 15
N	-	-	-
P	SIN	SIN	TP 10
R	-	-	-
S	-	-	-
T	COS	COS	TP 8
U	PH(x)CH3	FY current x	TP 6
V	-	-	-
W	PH(y)CH3	FY current y	TP 14
X	-	-	-

Control In *4.2*

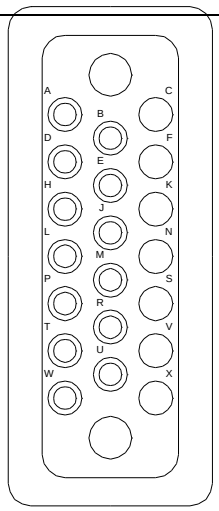


Table 4.2. Control In Burndy 20 Pin RM

Pin No.	new Name	old Name	connected to
A	PH1CH1	PHI90FX	TP 25
B	PH2CH1	PHI180FX	TP 26
C	-	-	-
D	-	-	-
E	-	-	-
F	-	-	-
H	PH1CH2	PHI90FH	TP 27
J	PH2CH2	PHI180FH	TP 28
K	-	-	-
L	-	-	-
M	-	-	-
N	-	-	-
P	PH1CH3	PHI90FY	TP 23
R	PH2CH3	PHI180FY	TP 24
S	-	-	-
T	-	-	-
U	-	-	-
V	-	-	-
W	-	-	-
X	-	-	-

SBS Slave *4.3*

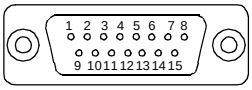


Table 4.3. SBS Slave 15 Pin Mini-D Male

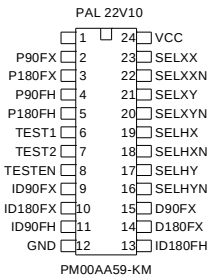
Pin No.	Pin Name
1	Shield
2	RxD+
3	WUP
4	TxD+
5	-
6	GND
7	GNd
8	GND
9	RxD-
10	-
11	TxD-
12	-
13	VRS +12V
14	VRS +12V
15	VRS +12V

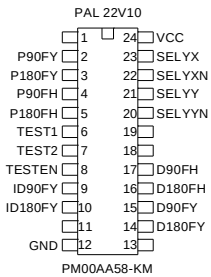
PAL Listings

A

***IDENTIFICATION PAL U58 and U59**

```
 Company:            Bruker Analytische Meßtechnik
                     D-7512 Rheinstetten 4/Karlsruhe
                     Silberstreifen
 Designer:           Bernd Jenissen  Abt. I MSL
 Date:               07.01.93

 Pal:
   Part name:        PM00AA58-KM + PM00AA59-KM
   Part number:      H5430
   Revision:         AA

 Assembly:
   Part name:        4 PHASE MODULATOR
   Part number:      H5263
   Revision:         B
   Pal location:     IC 58 + IC 59

 File name:          MODMUVBN.DCB

*Comment
 Inputmultiplexer for 4 Phase-Modulator

*X-NAMES
 P90FX    ,  ! Inputsignal Channel 1 90 degree
 P180FX   ,  ! Inputsignal Channel 1 180 degree
 P90FH    ,  ! Inputsignal Channel 2 90 degree
 P180FH   ,  ! Inputsignal Channel 2 180 degree
 P90FY    ,  ! Inputsignal Channel 3 90 degree
 P180FY   ,  ! Inputsignal Channel 3 180 degree
 ID90FX   ,  ! Inputsignal Channel 1 90 degree, delayed
 ID180FX  ,  ! Inputsignal Channel 1 180 degree, delayed
 ID90FH   ,  ! Inputsignal Channel 2 90 degree, delayed
 ID180FH  ,  ! Inputsignal Channel 2 180 degree, delayed
 ID90FY   ,  ! Inputsignal Channel 2 90 degree, delayed
 ID180FY  ,  ! Inputsignal Channel 2 180 degree, delayed
 TEST1    ,  ! Testsignal Channel 1+2 90 degree
 TEST2    ,  ! Testsignal Channel 1+2 180 degree
 TESTEN   ;  ! Test enabled from CPU

*Y-NAMES
 D90FX    ,  ! Output. f. delayed Inputsignal Channel 1 90 degree
 D180FX   ,  ! Output. f. delayed Inputsignal Channel 1 180 degree
 D90FH    ,  ! Output. f. delayed Inputsignal Channel 2 90 degree
 D180FH   ,  ! Output. f. delayed Inputsignal Channel 2 180 degree
 D90FY    ,  ! Output. f. delayed Inputsignal Channel 2 90 degree
 D180FY   ,  ! Output. f. delayed Inputsignal Channel 2 180 degree
 SELXX    ,  ! Output Channel 1 X
 SELXXN   ,  ! Output Channel 1 X -
 SELXY    ,  ! Output Channel 1 Y
 SELXYN   ,  ! Output Channel 1 Y -
 SELHX    ,  ! Output Channel 2 X
 SELHXN   ,  ! Output Channel 2 X -
 SELHY    ,  ! Output Channel 2 Y
 SELHYN   ,  ! Output Channel 2 Y -
 SELYX    ,  ! Output Channel 3 X
 SELYXN   ,  ! Output Channel 3 X -
 SELYY    ,  ! Output Channel 3 Y
 SELYYN   ;  ! Output Channel 3 Y -
```

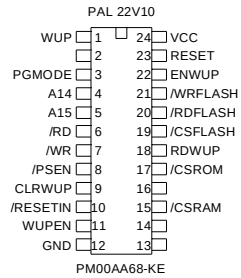




```
*FUNCTION-TABLE

! delay Inputsignals

 $P90FX:D90FX ;
    0  :  0   ;
  REST :  1   ;

 $P180FX:D180FX ;
    0   :  0    ;
  REST  :  1    ;

 $P90FH:D90FH ;
    0  :  0   ;
  REST :  1   ;

 $P180FH:D180FH ;
    0   :  0    ;
  REST  :  1    ;

 $P90FY:D90FY ;
    0  :  0   ;
  REST :  1   ;

 $P180FY:D180FY ;
    0   :  0    ;
  REST  :  1    ;

 ! Final Function

$TESTEN,TEST2,TEST1,P180FX,P90FX,ID180FX,ID90FX:SELXX,SELXXN,SELXY,SELXYN;
   0   ,  0  ,  0  ,  -   ,  -  ,   -   ,  -   :  0  ,   1  ,  1  ,  1    ;
   0   ,  0  ,  1  ,  -   ,  -  ,   -   ,  -   :  1  ,   1  ,  0  ,  1    ;
   0   ,  1  ,  0  ,  -   ,  -  ,   -   ,  -   :  1  ,   0  ,  1  ,  1    ;
   0   ,  1  ,  1  ,  -   ,  -  ,   -   ,  -   :  1  ,   1  ,  1  ,  0    ;
   1   ,  -  ,  -  ,  0   ,  0  ,   0   ,  0   :  0  ,   1  ,  1  ,  1    ;
   1   ,  -  ,  -  ,  0   ,  1  ,   0   ,  1   :  1  ,   1  ,  0  ,  1    ;
   1   ,  -  ,  -  ,  1   ,  0  ,   1   ,  0   :  1  ,   0  ,  1  ,  1    ;
   1   ,  -  ,  -  ,  1   ,  1  ,   1   ,  1   :  1  ,   1  ,  1  ,  0    ;
   REST                                        :  1  ,   1  ,  1  ,  1    ;

$TESTEN,TEST2,TEST1,P180FH,P90FH,ID180FH,ID90FH:SELHX,SELHXN,SELHY,SELHYN ;
   0   ,  0  ,  0  ,  -   ,  -  ,   -   ,  -   :  0  ,   1  ,  1  ,  1    ;
   0   ,  0  ,  1  ,  -   ,  -  ,   -   ,  -   :  1  ,   1  ,  0  ,  1    ;
   0   ,  1  ,  0  ,  -   ,  -  ,   -   ,  -   :  1  ,   0  ,  1  ,  1    ;
   0   ,  1  ,  1  ,  -   ,  -  ,   -   ,  -   :  1  ,   1  ,  1  ,  0    ;
   1   ,  -  ,  -  ,  0   ,  0  ,   0   ,  0   :  0  ,   1  ,  1  ,  1    ;
   1   ,  -  ,  -  ,  0   ,  1  ,   0   ,  1   :  1  ,   1  ,  0  ,  1    ;
   1   ,  -  ,  -  ,  1   ,  0  ,   1   ,  0   :  1  ,   0  ,  1  ,  1    ;
   1   ,  -  ,  -  ,  1   ,  1  ,   1   ,  1   :  1  ,   1  ,  1  ,  0    ;
   REST                                        :  1  ,   1  ,  1  ,  1    ;


$TESTEN,TEST2,TEST1,P180FY,P90FY,ID180FY,ID90FY:SELYX,SELYXN,SELYY,SELYYN ;
   0   ,  0  ,  0  ,  -   ,  -  ,   -   ,  -   :  0  ,   1  ,  1  ,  1    ;
   0   ,  0  ,  1  ,  -   ,  -  ,   -   ,  -   :  1  ,   1  ,  0  ,  1    ;
   0   ,  1  ,  0  ,  -   ,  -  ,   -   ,  -   :  1  ,   0  ,  1  ,  1    ;
   0   ,  1  ,  1  ,  -   ,  -  ,   -   ,  -   :  1  ,   1  ,  1  ,  0    ;
   1   ,  -  ,  -  ,  0   ,  0  ,   0   ,  0   :  0  ,   1  ,  1  ,  1    ;
   1   ,  -  ,  -  ,  0   ,  1  ,   0   ,  1   :  1  ,   1  ,  0  ,  1    ;
   1   ,  -  ,  -  ,  1   ,  0  ,   1   ,  0   :  1  ,   0  ,  1  ,  1    ;
   1   ,  -  ,  -  ,  1   ,  1  ,   1   ,  1   :  1  ,   1  ,  1  ,  0    ;
   REST                                        :  1  ,   1  ,  1  ,  1    ;

*RUN-CONTROL
 LISTING  = SYMBOLTABLE, NETTABLE ;
 OPTIMIZE = P-TERMS ;

*END
```

```
*IDENTIFICATION
  LOG/iC Partitioner 3.4-08   92/11/30  11:18:45

@DEVICE = PM00AA59 ;

  *PLD
   TYPE =  AMPAL22V10;

  *PINS
   P90FX  =  2 ,
   P180FX =  3 ,
   P90FH  =  4 ,
   P180FH =  5 ,
   ID90FX =  9 ,
   ID180FX= 10 ,
   ID90FH = 11 ,
   ID180FH= 13 ,
   TEST1  =  6 ,
   TEST2  =  7 ,
   TESTEN =  8 ,
   D90FX  = 15 ,
   D180FX = 14 ,
   SELXX  = 23 ,
   SELXXN = 22 ,
   SELXY  = 21 ,
   SELXYN = 20 ,
   SELHX  = 19 ,
   SELHXN = 18 ,
   SELHY  = 17 ,
   SELHYN = 16 ;

@ENDDEVICE = PM00AA59 ;


@DEVICE = PM00AA58 ;

  *PLD
   TYPE =  AMPAL22V10;

  *PINS
   P90FH   =  4,
   P180FH  =  5,
   P90FY   =  2,
   P180FY  =  3,
   ID90FY  =  9,
   ID180FY = 10,
   TEST1  =   6,
   TEST2  =   7,
   TESTEN  =  8,
   D90FH  =  17,
   D180FH  = 16,
   D90FY  =  15,
   D180FY  = 14,
   SELYX  =  23,
   SELYXN  = 22,
   SELYY  =  21,
   SELYYN  = 20;

@ENDDEVICE = PM00AA58;

*RUN-CONTROL

  Progformat=Jedec;

  Listing=Equations,Pinout;

  TESTVECTOR = GENERATE;

*END
```

```
*IDENTIFICATION      PAL U68

 Company:            BRUKER ELEKTRONIK GmbH
                     D-7512 Rheinstetten 4/Karlsruhe
                     Akazienweg 2

 Designer:           Uwe Döttling

 Date:               02.06.92

 PAL:
   Part name:        PM00AA68-KE
   Part number:      H5430
   Revision:         AA

 Assembly:
   Part name:        4 PHASE MODULATOR
   Part number:      H5263
   Revision:         B
   Pal location:     IC 68

 File name:          4PHDECA.DCB

*COMMENT

 THIS PAL WORKS AS AN ADDRESS DECODER AND MEMORY MAPPER.
 THE RESET LOGIC IS ALSO DONE BY THE PAL.

*X-NAMES

 WR          ;LOW ACTIVE CPU WRITE SIGNAL
 RD          ;LOW ACTIVE CPU READ SIGNAL
 PSEN        ;LOW => CODE MEMORY, HIGH => DATA MEMORY
 A[15..14]   ;HIGH ACTIVE CPU ADDRESSES
 PGMODE      ;HIGH ACTIVE MODE BIT FOR MEMORY MAPPING AND FLASH WRITE
 WUPEN       ;HIGH ACTIVE MODE BIT FOR WAKE UP ENABLE
 RESETIN     ;LOW ACTIVE RESET INPUT FROM TL7705
 WUP         ;LOW ACTIVE WAKE UP INPUT FROM 74LS123
 CLRWUP      ;RESET D-FLIP FLOP SET BY WUP

*Y-NAMES

 CSRAM       ;LOW ACTIVE CHIP SELECT FOR RAM
 CSROM       ;LOW ACTIVE CHIP SELECT FOR BOOT EPROM
 CSFLASH     ;LOW ACTIVE CHIP SELECT FOR FLASH EPROM
 RDFLASH     ;LOW ACTIVE READ SIGNAL EITHER /RD OR /PSEN FOR FLASH EPROM
 WRFLASH     ;LOW ACTIVE WRITE SIGNAL FOR FLASH EPROM
 ENWUP       ;HIGH ACTIVE ENABLES WAKE UP FROM RS485
 RESET       ;LOW ACTIVE RESET OUT TO 80535
 RDWUP       ;READ D-FLIP FLOP ACTIVATED BY WUP
 TEMP        ;TEMPORAR OUTPUT TO GENERATE RESET TERM FOR ASYNCHRON RESET
             ;OF ALL D-FLIP FLOPS

*FUNKTION-TABLE

 $ WUPEN : ENWUP  ;JUST AN INVERTER
     0   :   1    ;WAKE UP ENABLED
     1   :   0    ;WAKE UP DISABLED

 $ (A[15..14]),PGMODE,PSEN:CSROM,CSFLASH,CSRAM;CHIP SELECTS

    0H..03H   ,   1  , -  :  0  ,   0   ,  1  ;EPROM+FLASH IN DOWN MODE
    0H..01H   ,   0  , -  :  1  ,   0   ,  0  ;FLASH+RAM IN NORMAL MODE
    2H..03H   ,   0  , -  :  1  ,   0   ,  1  ;FLASH IN NORMAL MODE
    REST                  :  1  ,   1   ,  1  ;ELSE NOTHING

 $ (A[15..14]),PGMODE,RD,WR,PSEN:RDFLASH,WRFLASH;

    0H..03H   ,  1   , 1, 0, 1  :   1   , 0     ;FLASH WR IN DATA LOC.
    0H..03H   ,  1   , 0, 1, 1  :   0   , 1     ;FLASH RD IN DATA LOC.
    0H..03H   ,  0   , 1, 1, 0  :   0   , 1     ;FLASH RD IN CODE LOC.
    REST                        :   1   , 1     ;
```



```
 $  WUP, RESETIN :   RESET     ;RESET OUTPUT FOR 80535
     0 ,    1    :     1       ;RESET SOURCE IS EITHER RESETIN OR WUPIN
   REST          :     0       ;

*BOOLEAN-EQUATIONS

 TEMP     = /RESETIN + /CLRWUP ;TEMPORARY TERM TO CLEAR D-FLIP FLOPS
 RDWUP.RS = TEMP               ;CLEAR ALL FLIP FLOPS ASYNCHRONUS
 RDWUP   := VCC                ;LOW -> HIGH AT CLOCK BRINGS RDWUP HIGH

*SPECIAL-FUNCTIONS

 RDWUP.INV = NO                ;OUTPUT RDWUP NOT INVERTED
 RDWUP.REG = YES               ;AND OUTPUT RDWUP IS USING REGISTERS

*RUN-CONTROL

 LISTING = PLOT, EQUATIONS, PINOUT, SYMBOLTABLE;
 OPTIMIZE = P-TERMS;
 PROGFORMAT = JEDEC;
 TESTVECTORS = GENERATE;

*PLD

 TYPE = PAL22V10;
 CHECKSUM = COMPUTE;

*PINS

  WUP       = 1,
  PGMODE    = 3,
  A14       = 4,
  A15       = 5,
  RD        = 6,
  WR        = 7,
  PSEN      = 8,
  CLRWUP    = 9,
  RESETIN   = 10,
  WUPEN     = 11,
  TEMP      = 14,
  CSRAM     = 15,
  DACWR     = 16,
  CSROM     = 17,
  RDWUP     = 18,
  CSFLASH   = 19,
  RDFLASH   = 20,
  WRFLASH   = 21,
  ENWUP     = 22,
  RESET     = 23;

*END
```

Schematics

B

On the following pages you will find the schematics in the following order:

AQR 4 Phase Modulator CPU

AQR 4 Phase Modulator SBS-Bus

AQR 4 Phase Modulator PH(x)CH1

AQR 4 Phase Modulator PH(y)CH1

AQR 4 Phase Modulator PH(x)CH2

AQR 4 Phase Modulator PH(y)CH2

AQR 4 Phase Modulator PH(x)CH3

AQR 4 Phase Modulator PH(y)CH3

AQR 4 Phase Modulator Capacitors

AQR 4 Phase Modulator Power

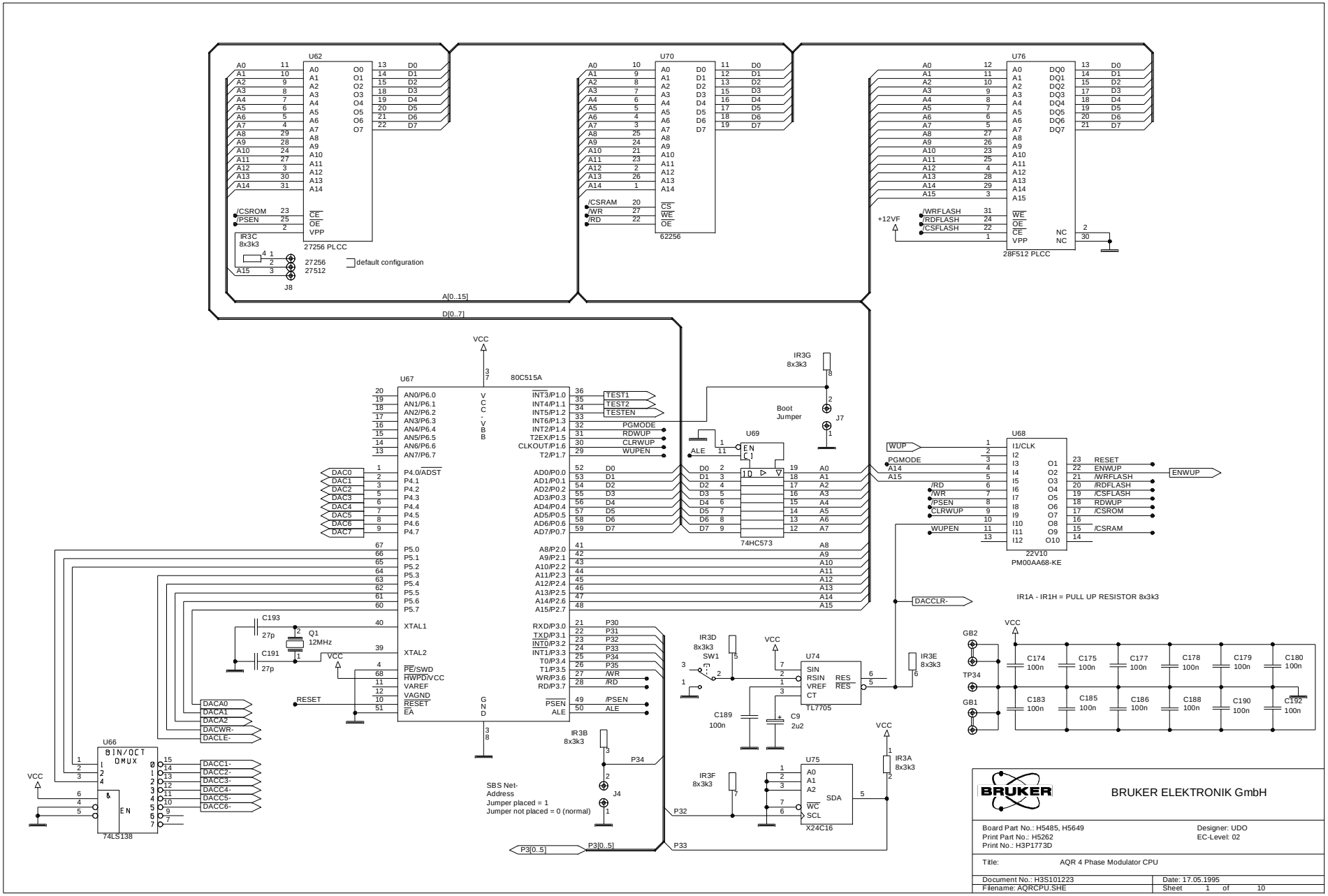
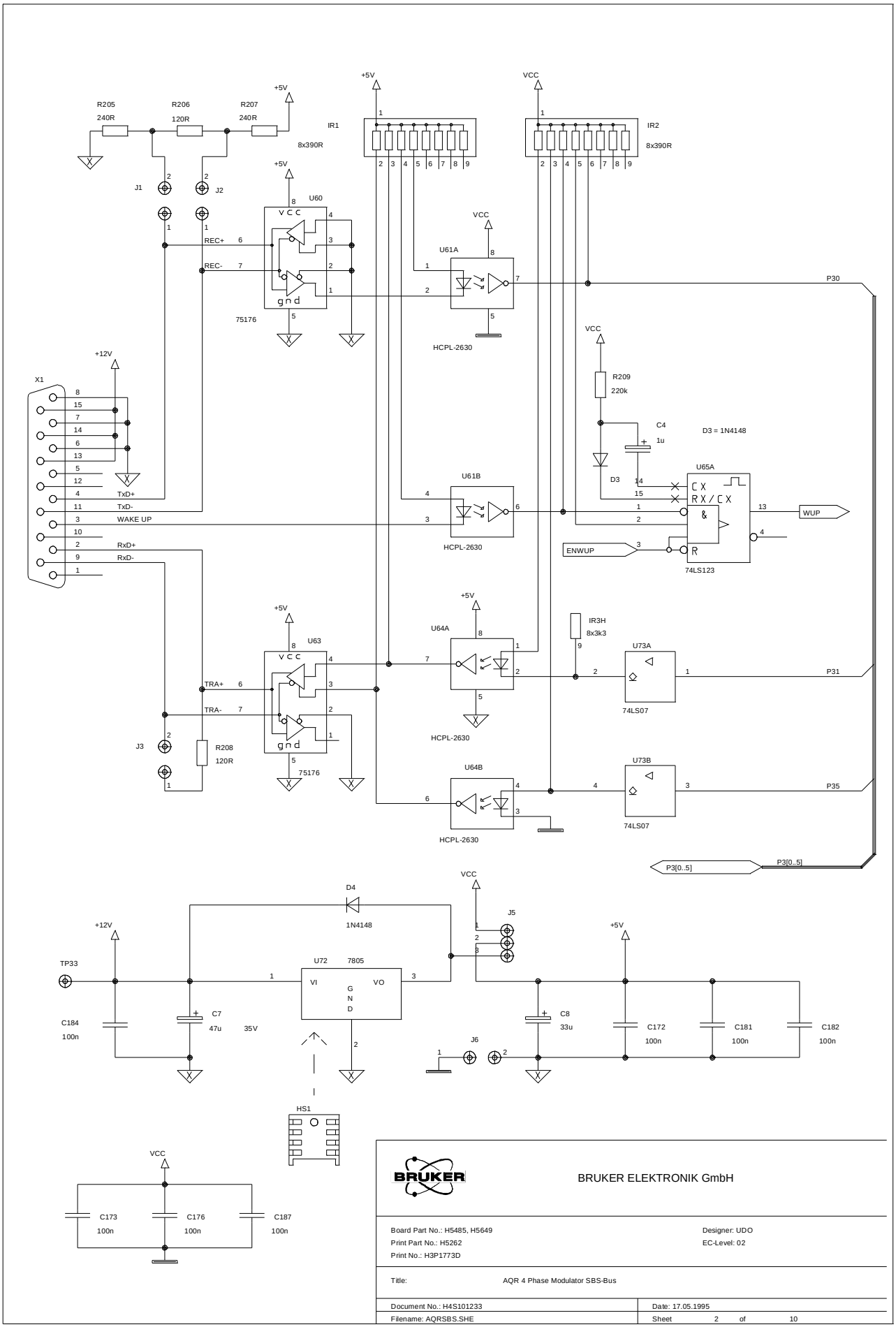


Figure B.1. AQR 4 Phase Modulator CPU

Figure B.2. AQR 4 Phase Modulator SBS-Bus



BRUKER ELEKTRONIK GmbH

Board Part No.: H5485, H5649
Print Part No.: H5262
Print No.: H3P1773D

Designer: UDO
EC-Level: 02

Title: AQR 4 Phase Modulator SBS-Bus

Document No.: H4S101233
Filename: AQRSBS.SHE

Date: 17.05.1995
Sheet 2 of 10

Figure B.3. AQR 4 Phase Modulator PH(x)CH1

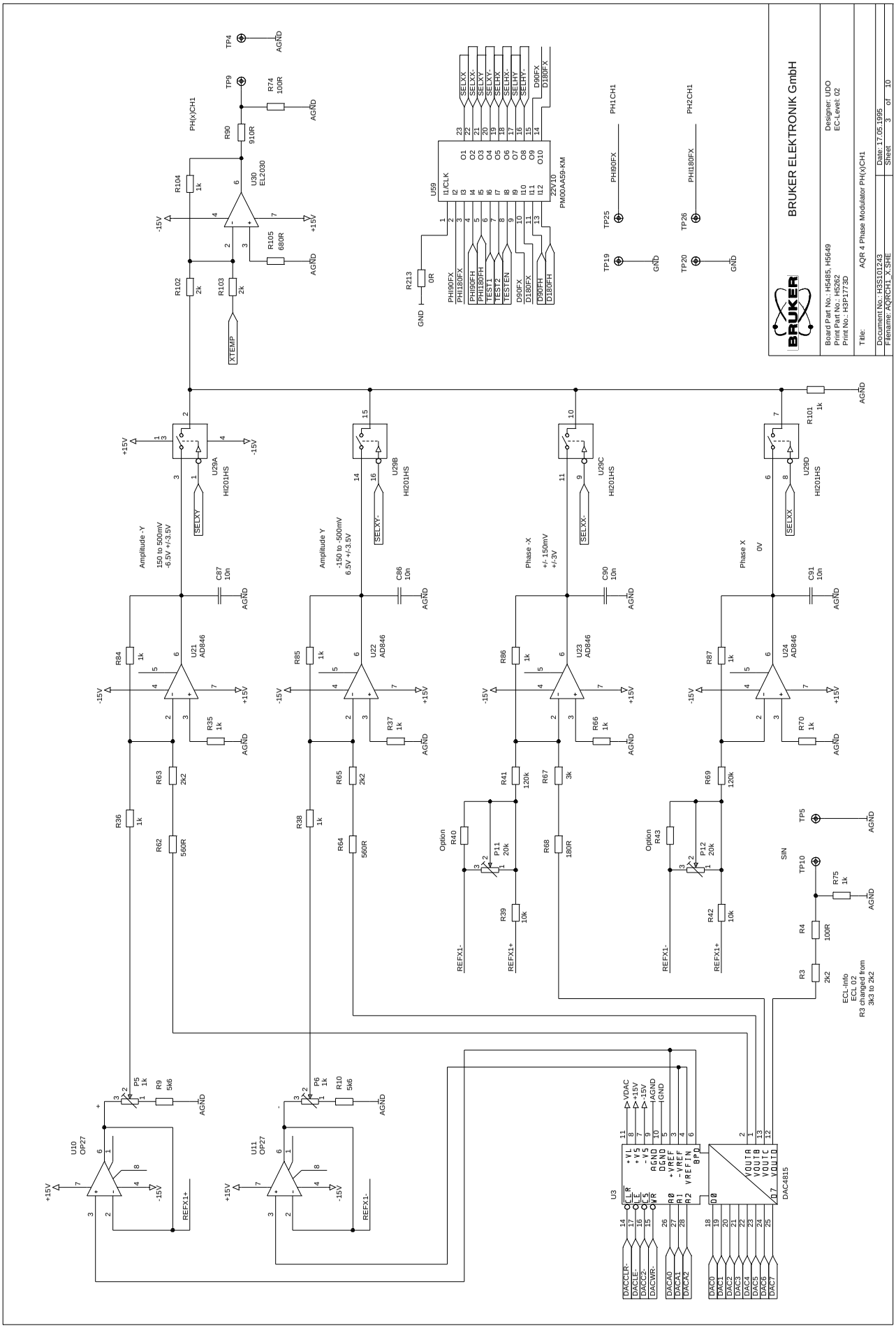


Figure B.4. AQR 4 Phase Modulator PH(y)CH1

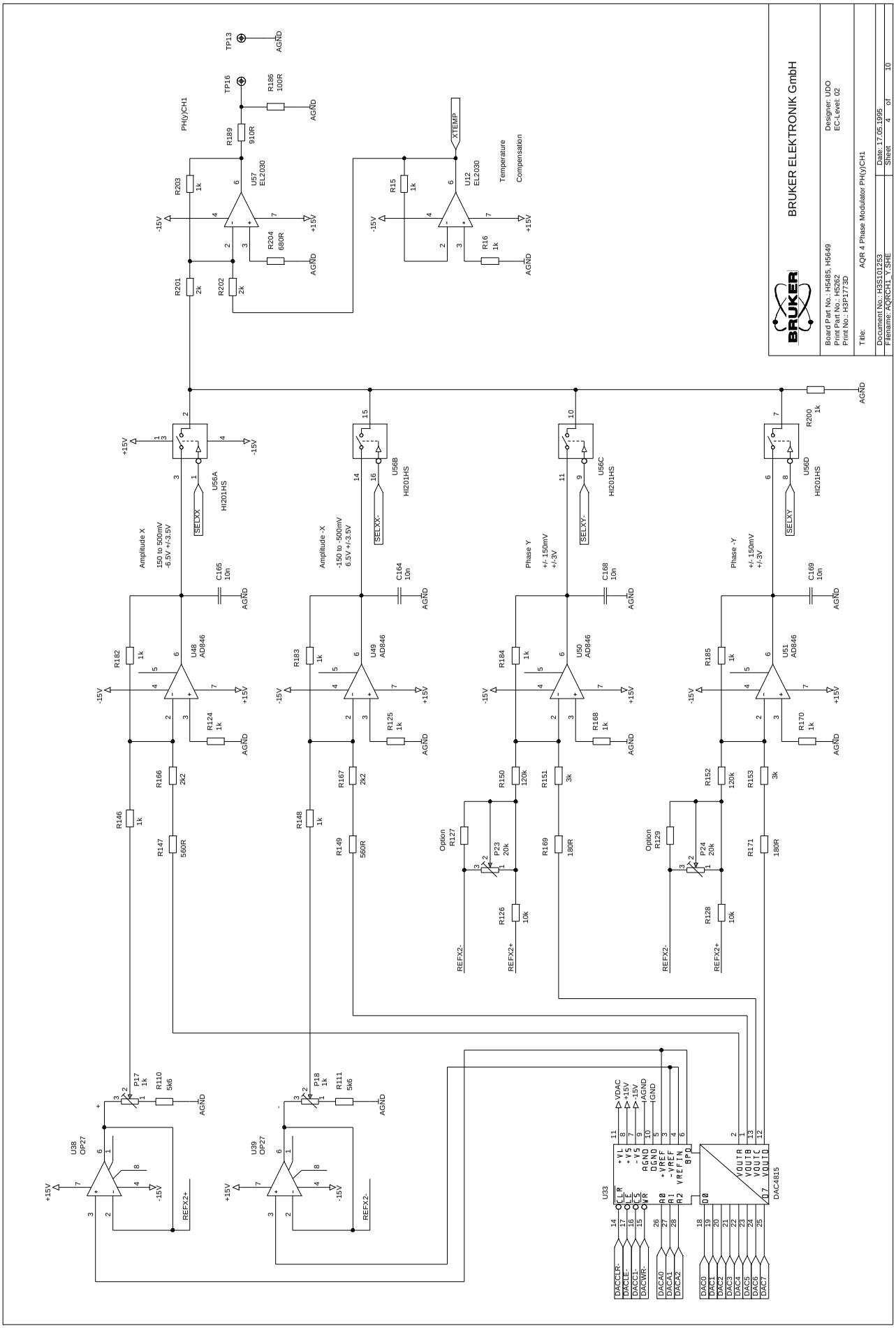


Figure B.5. AQR 4 Phase Modulator PH(x)CH2

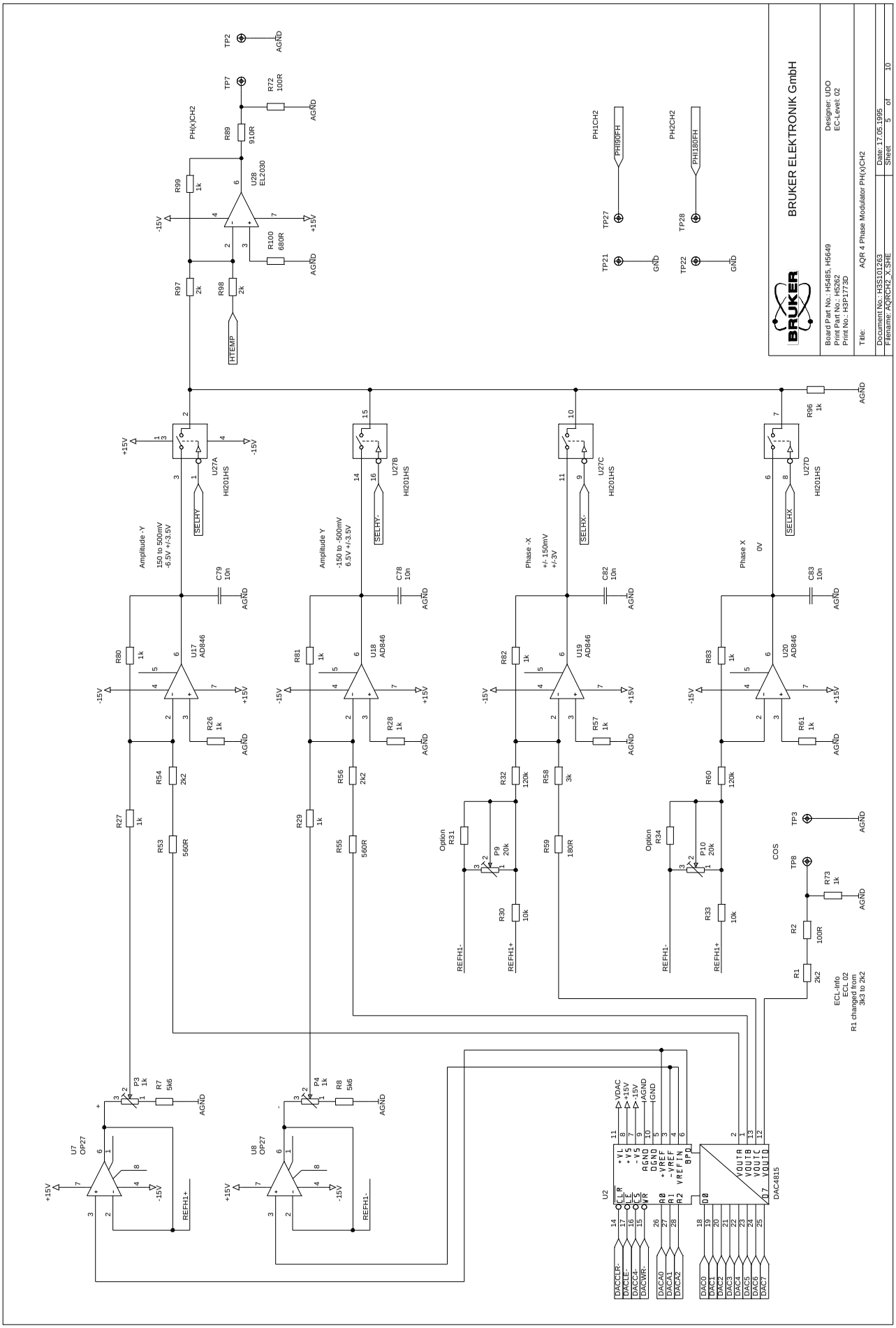


Figure B.6. AQR 4 Phase Modulator PH(y)CH2

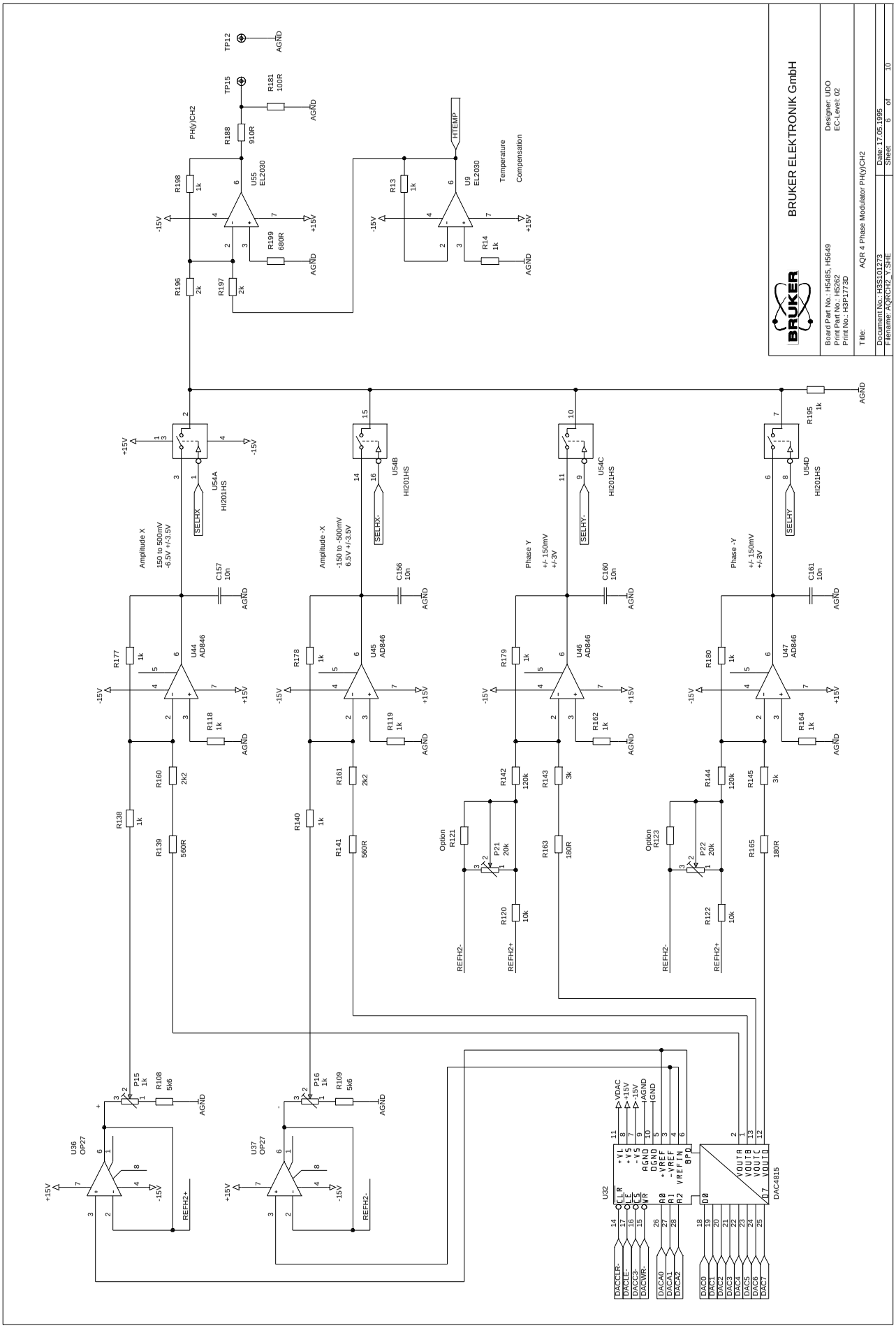


Figure B.7. AQR 4 Phase Modulator PH(x)CH3

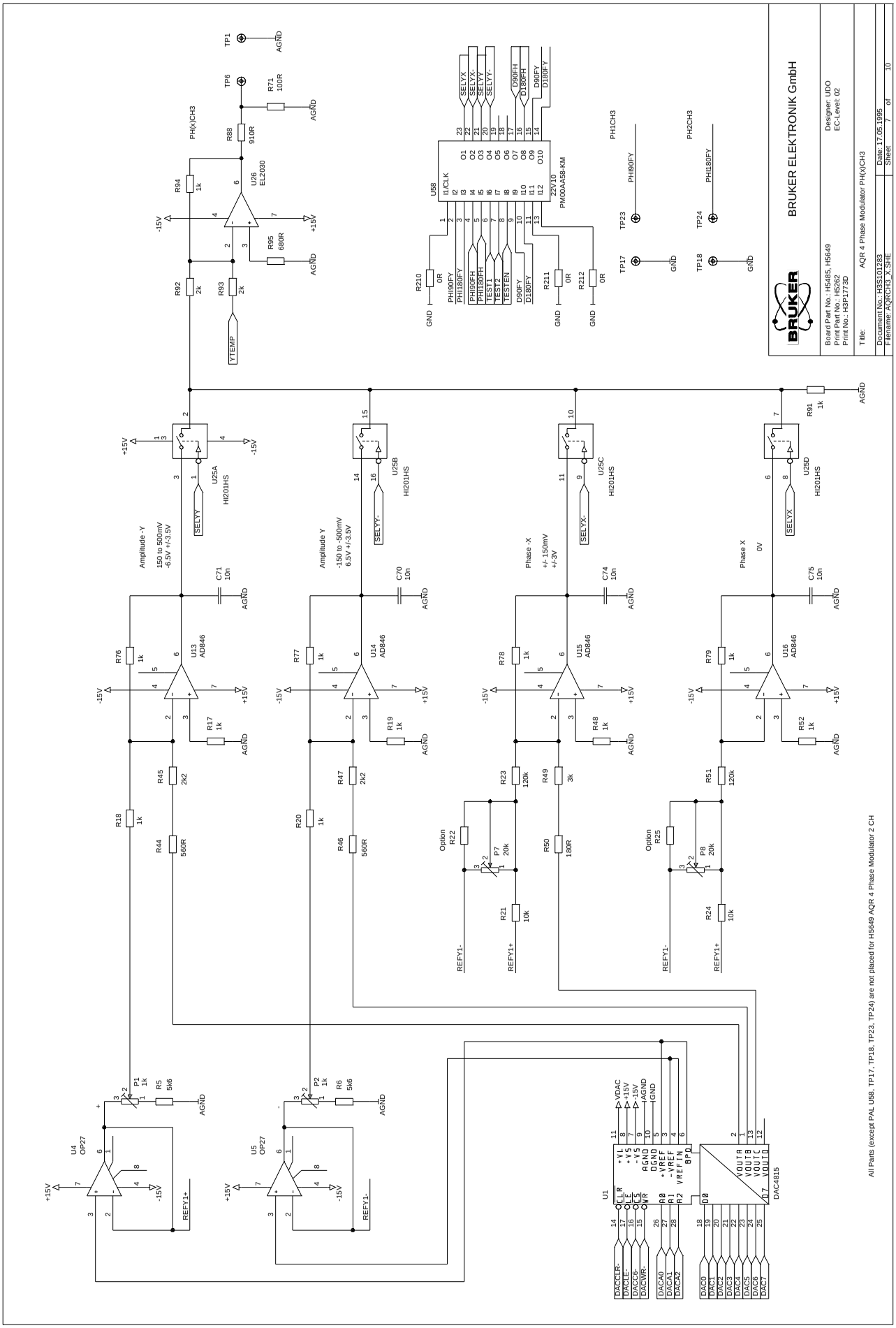


Figure B.8. AQR 4 Phase Modulator PH(y)CH3

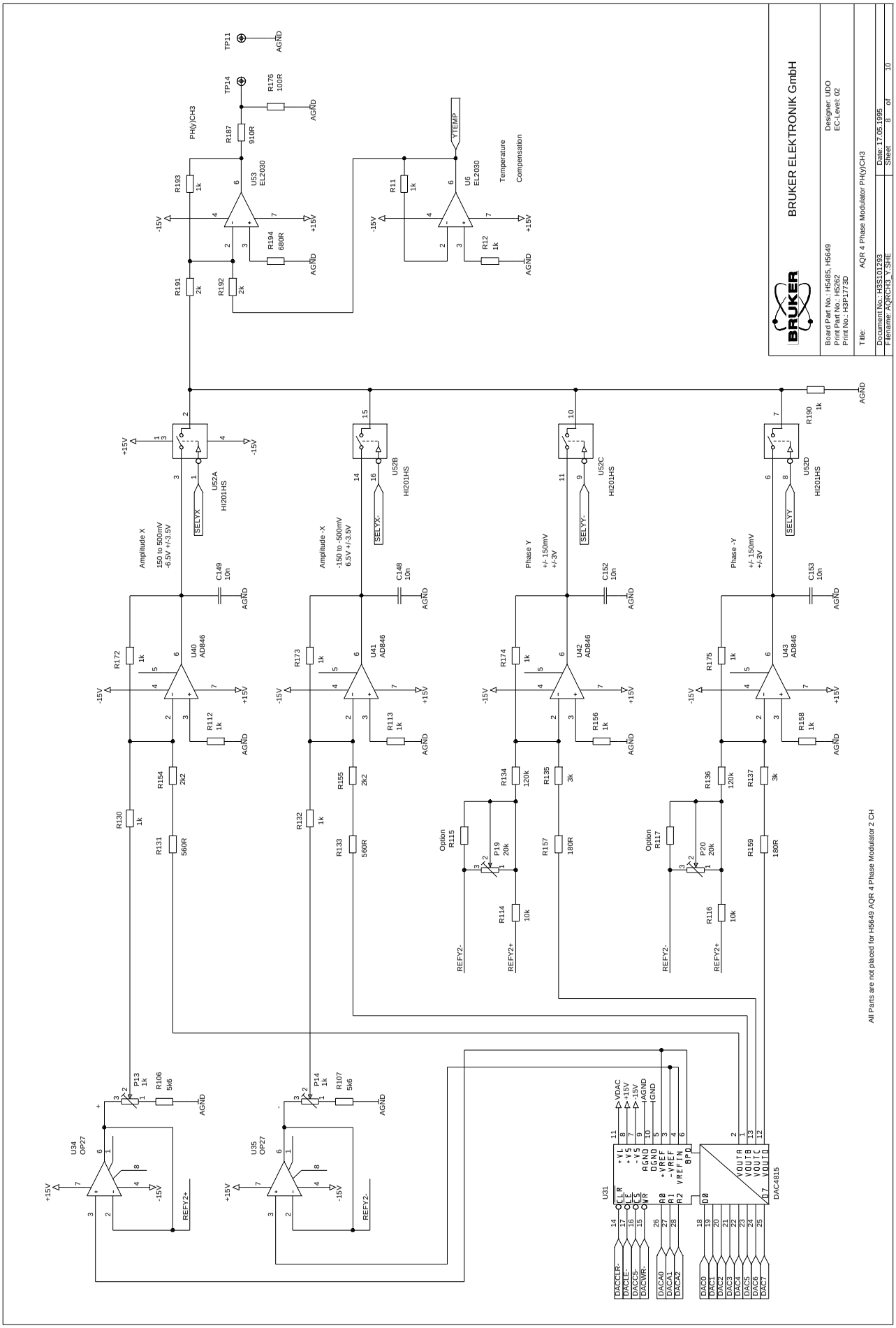
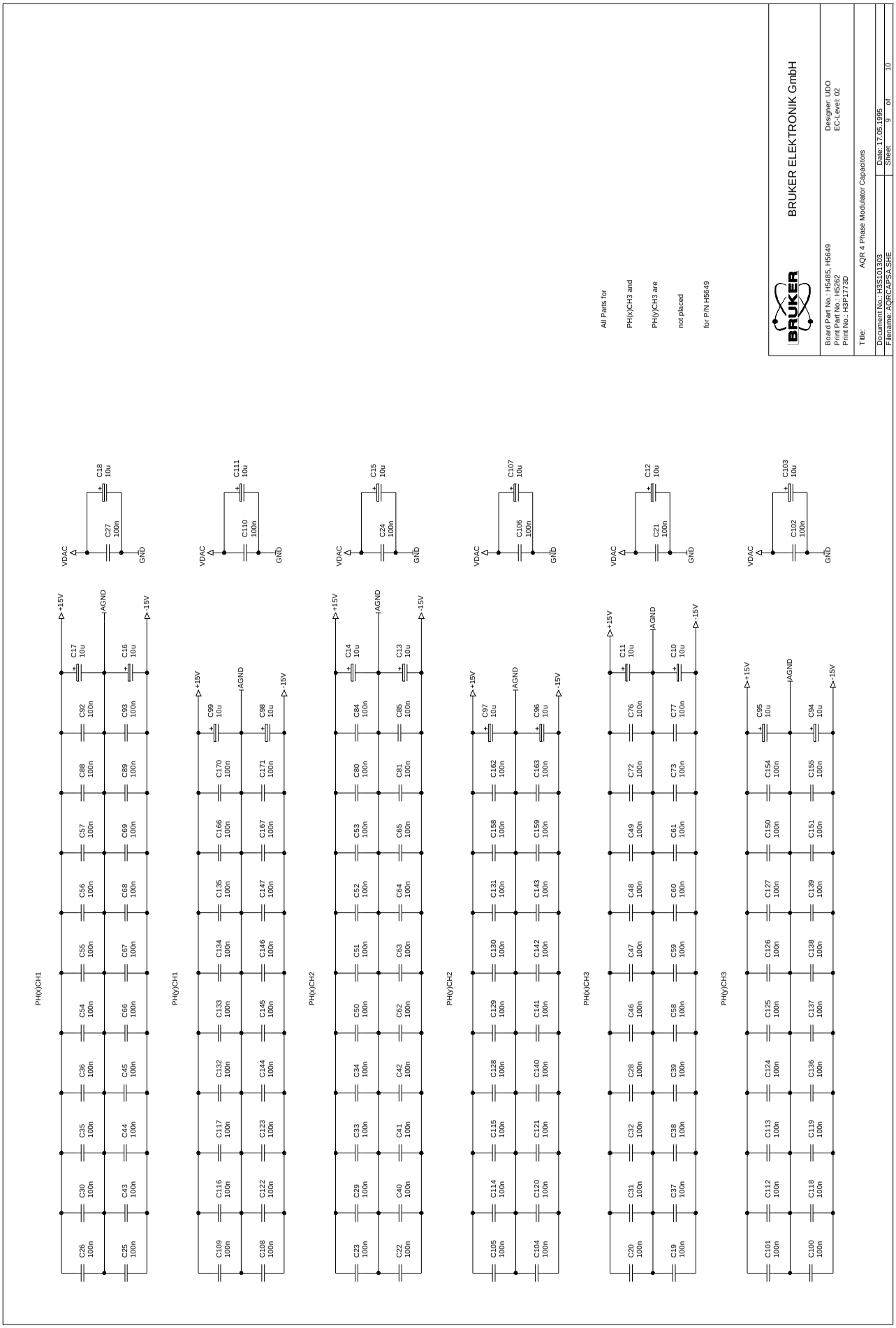
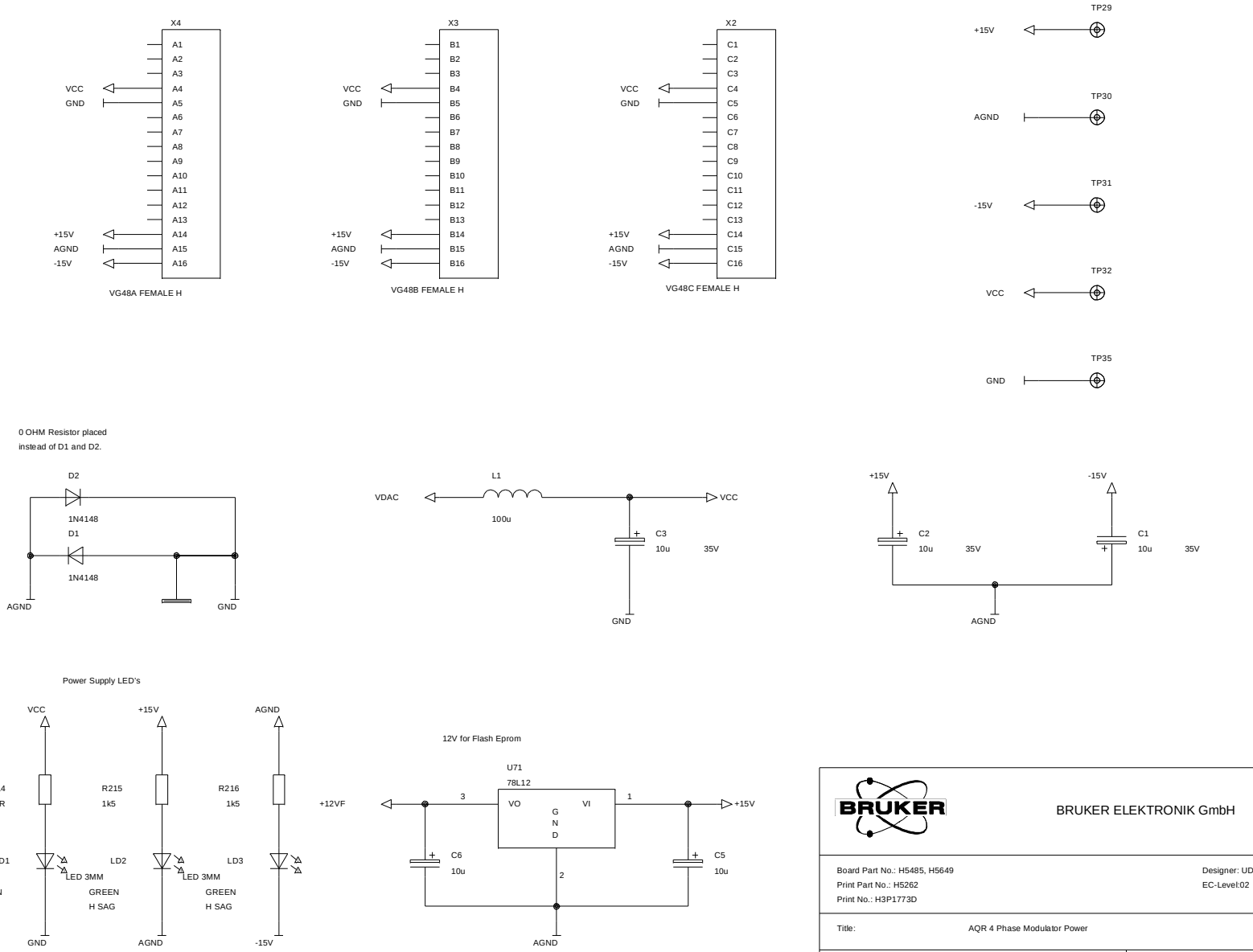


Figure B.9. AQR 4 Phase Modulator Capacitors





BRUKER ELEKTRONIK GmbH

Board Part No.: H5485, H5649	Designer: UDO
Print Part No.: H5262	EC-Level:02
Print No.: H3P1773D	
Title: AQR 4 Phase Modulator Power	
Document No.: H4S101313	Date: 17.05.1995
Filename: AQRPOWER.SHE	Sheet 10 of 10

Figure B.10. AQR 4 Phase Modulator Power

Figures

Figures

Tables

Tables