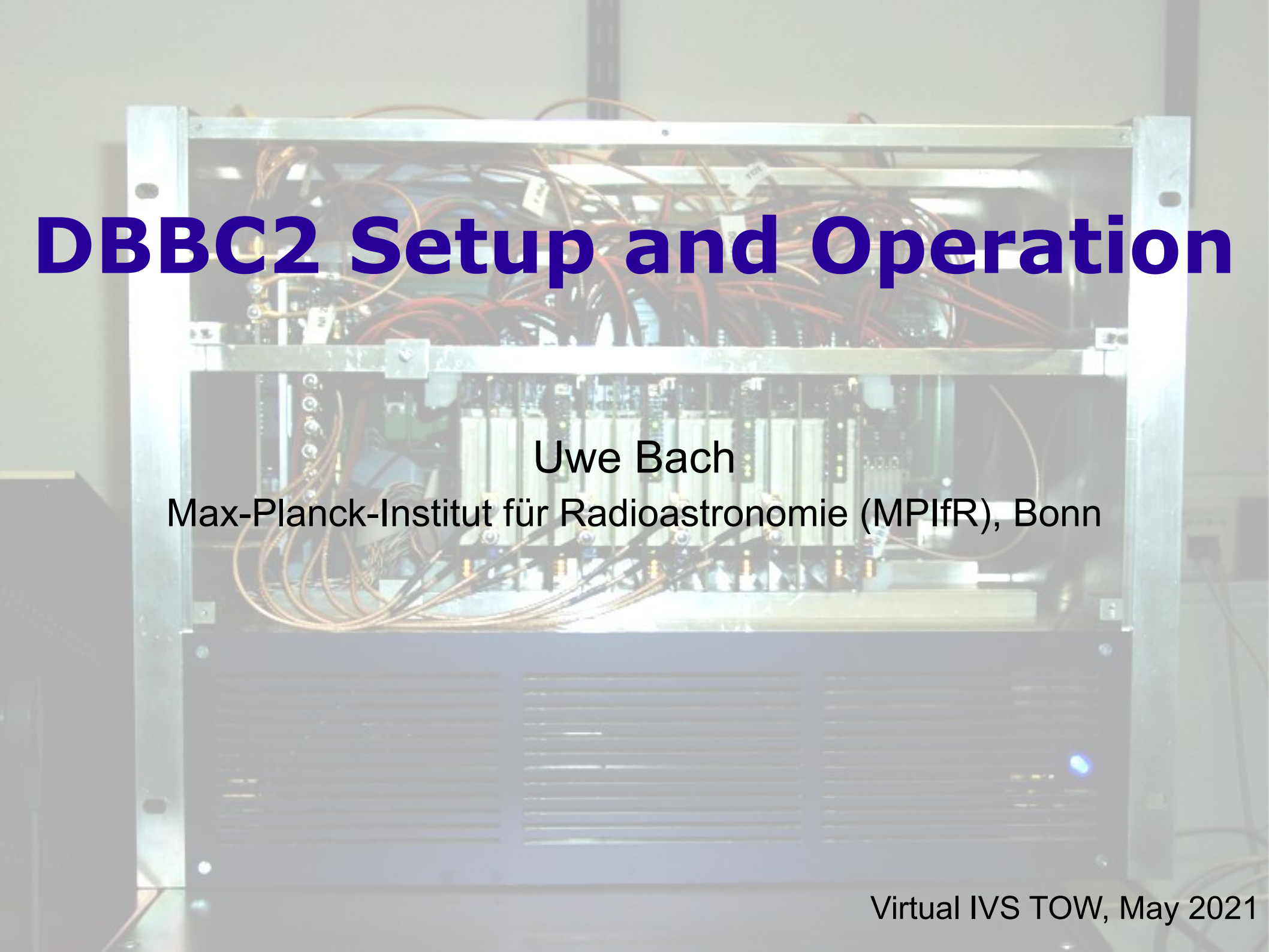


DBBC2 Setup and Operation

Uwe Bach
Max-Planck-Institut für Radioastronomie (MPIfR), Bonn

Virtual IVS TOW, May 2021

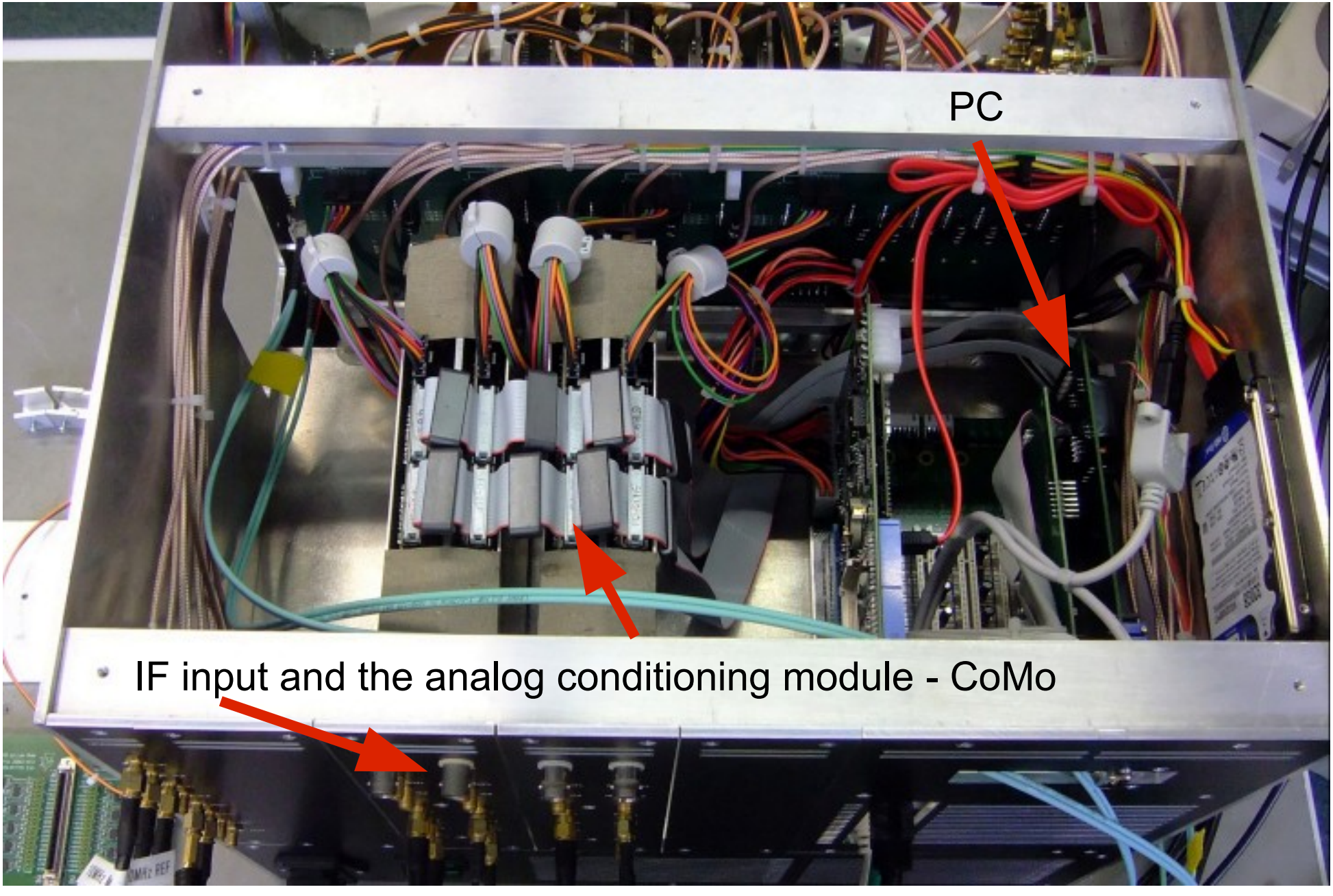
Content

- DBBC2 hardware characteristics
  - A tour around the DBBC2
  - Component description
- Installation of a DBBC2
- DBBC2 software
  - Poly-phase Filter Bank (PFB)
  - Digital Down Conversion (DDC)
- Basic testing
- Field System integration
- VLBI operation

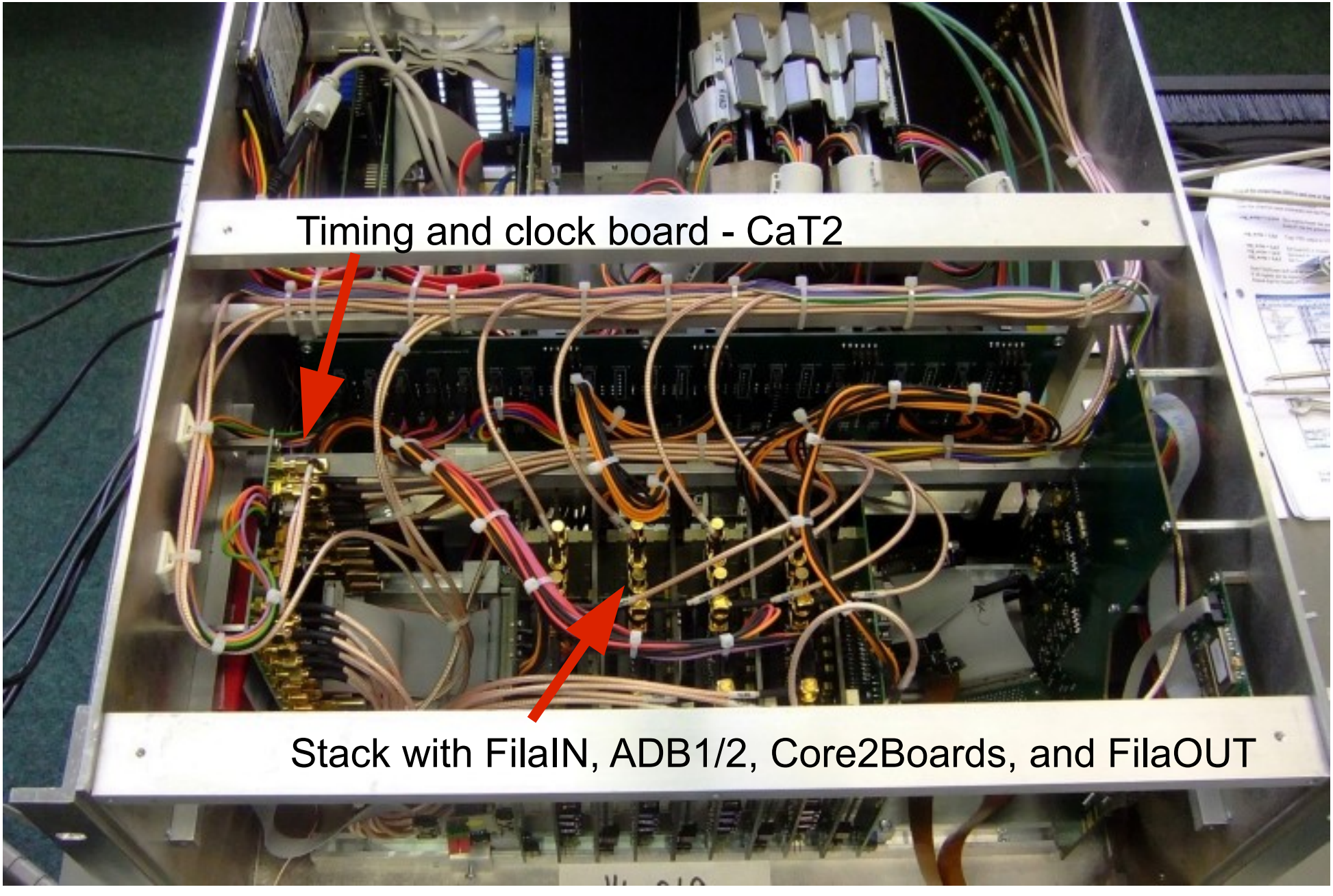
DBBC2 Outside (front view)

DBBC2 Outside (rear view)

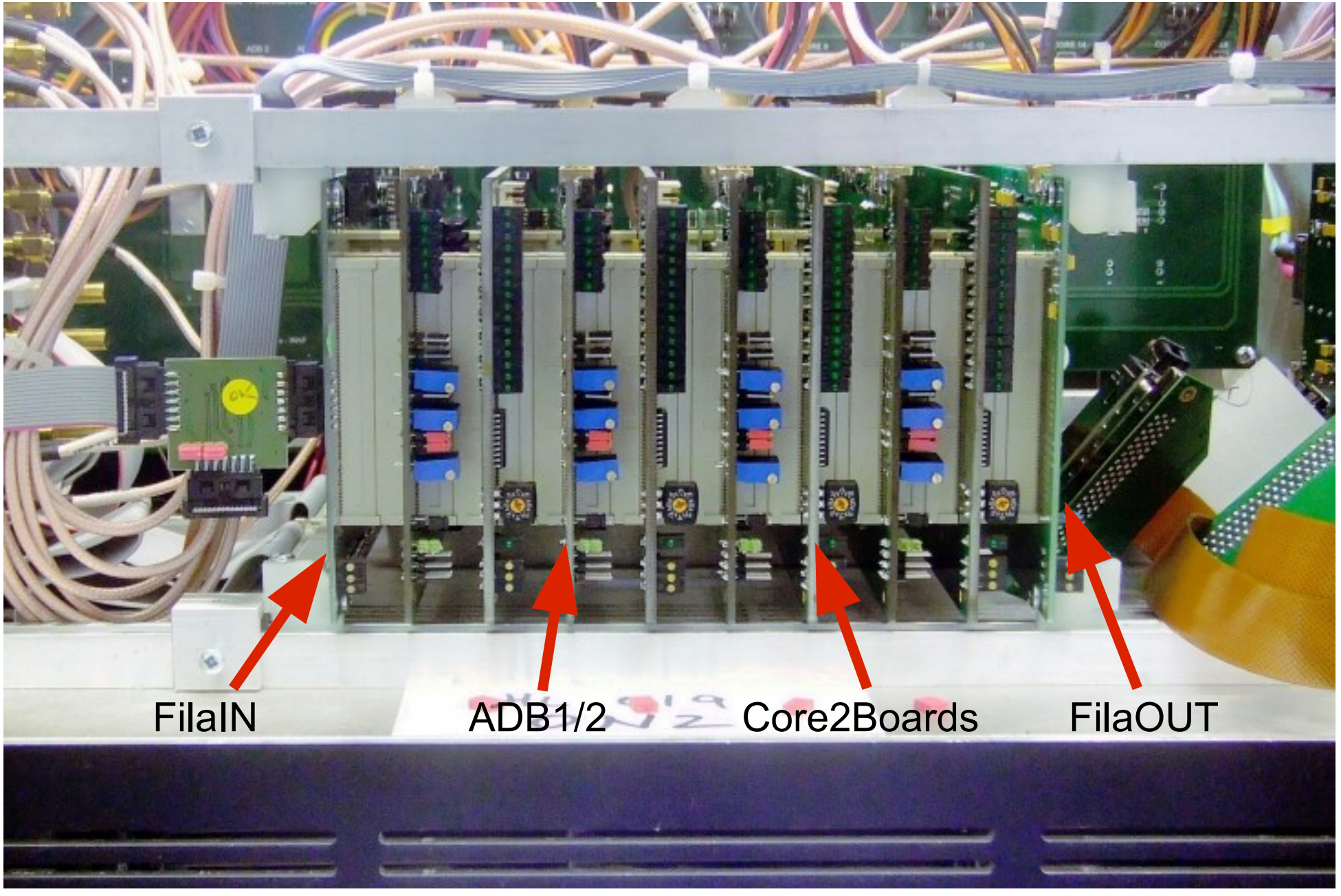
DBBC Inside



DBBC Inside


Timing and clock board - CaT2
Stack with FilaIN, ADB1/2, Core2Boards, and FilaOUT

DBBC Inside



Component description

1. Analog Conditioning Module – CoMo
   RF inputs up to 3.5 GHz; selectable Nyqvist zone filters 1-512, 512-1024, 1024-1536, 1536-2048; 31.5 dB programmable attenuation

2. Analog-Digital Converter (ADB1, 2, 3) 8-bit data stream

3. Data Processing (Core2Board)
   Several personalities available:
   - Digital down conversion (DDC) 1 Core2 = 4 BBCs
   - Poly-phase Filter Bank (PFB) 1 Core2 = 16 Poly-phase filters
   2 VSI 32 channel output

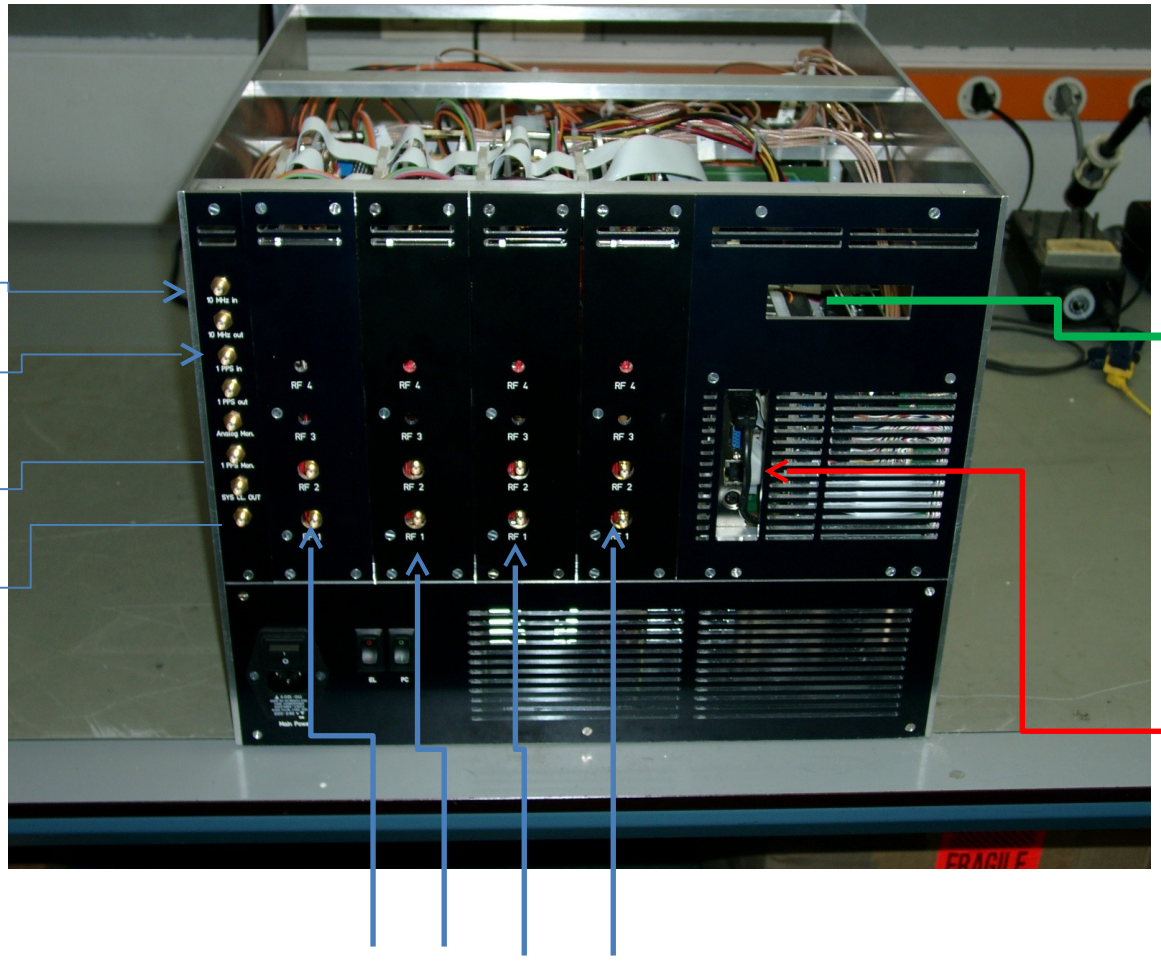
4. Connection and Service
   (FiLaIN/OUT – FiLa10G FILA10G-4)

5. Timing and Clock (CaT2 – Clock and Timing)

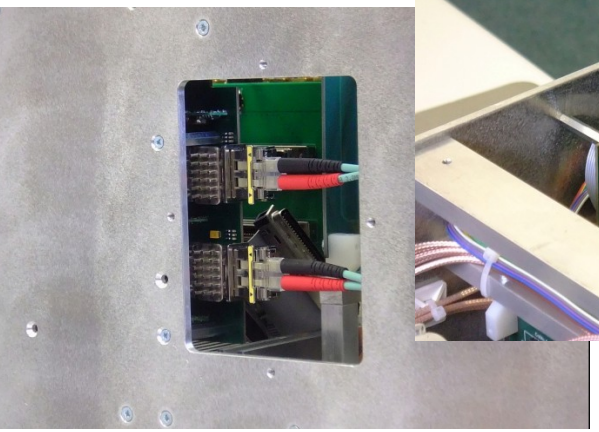
6. Computer Control (PCSet)

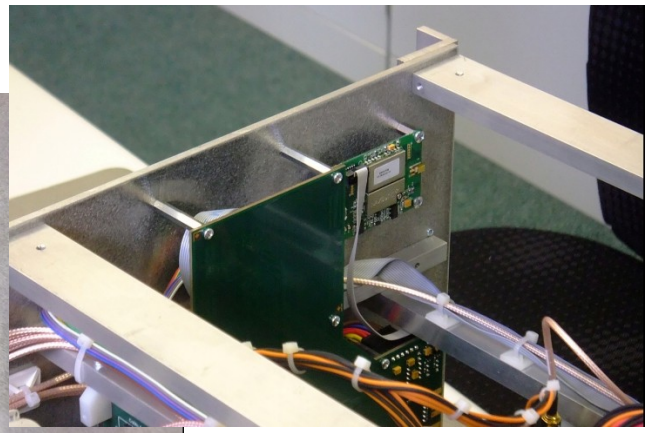
Installation of a DBBC

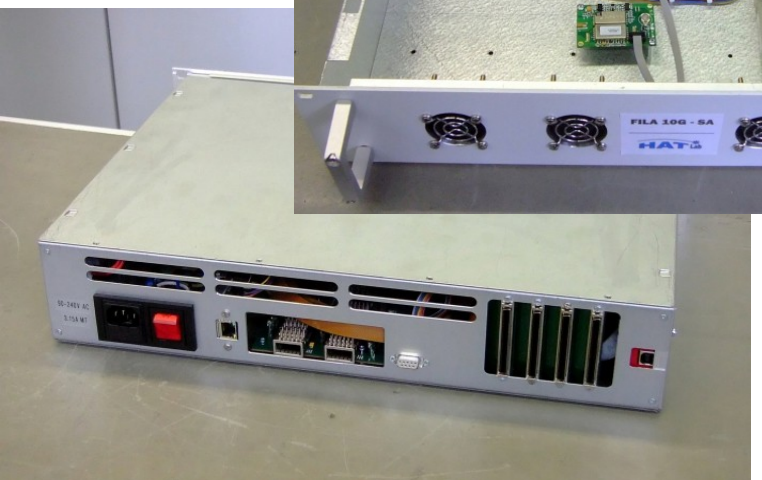
How to connect the DBBC



FiLa10G (SA)

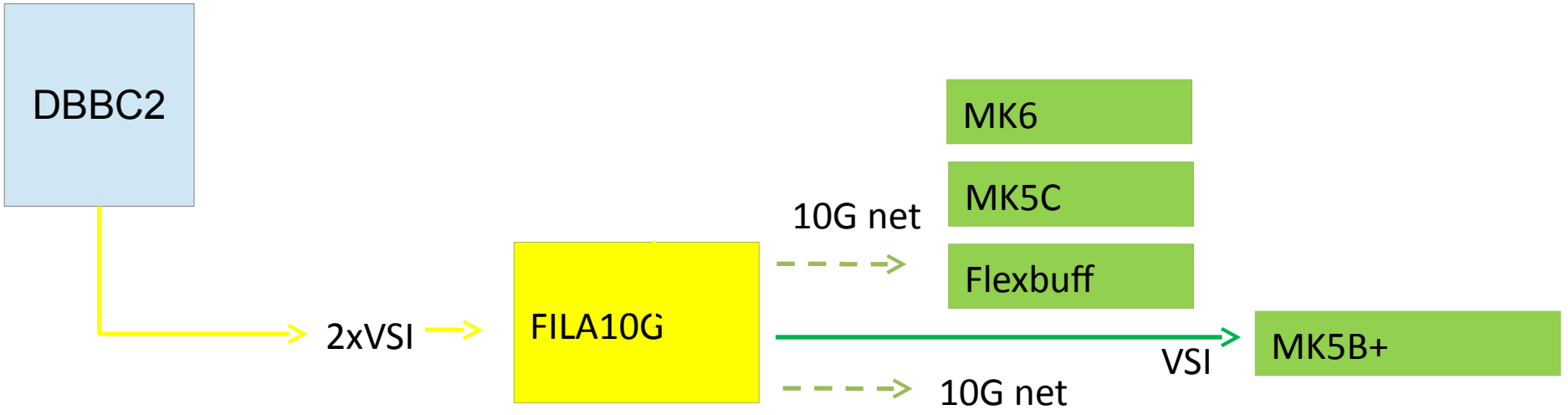






- Two 10G Ethernet ports
- 4 VSI in-/output ports
- Installed inside the DBBC box or as stand-alone
- Format mode: RAW, MK5B or VDIF
- Includes a GPS module for time synchronization
- Serial connection to the DBBC2
- Upload of the firmware is automatically made by the control software (internal FiLa10G) or done with an additional Xilinx JTAG programmer.

Connection example



Observing modes

- DDC: tunable, channel bandwidth between 1 MHz and 64 MHz, U&L, Continuous cal with 80 Hz synchronization, modes: geo, astro, astro2, w-astro, lba, test
- PFB: fixed tuning, channel bandwidth 32/64 MHz, all U or L depending on the Nyquist zone
- DSC: full 4 x 512/1024 MHz, max 8 x 1024 MHz band direct sampling conversion, all U or L depending on the Nyquist zone
- SPECTRA: 4Kch/IF spectrometer, max 32K channels

Each firmware comes with a dedicated set of control software and configuration files.

https://www.hat-lab.cloud

https://www.hat-lab.cloud/downloads-dbbc2-ddc/

Suchen

Hat Lab

Hi, Uwe Bach

HAT Lab

HOME COMPANY PROFILE DBBC BACK-ENDS EVOLUTION ACTIVITY AND PLANS PRODUCTS FUNK HAUS DOWNLOAD LOGIN CONTACTS

There are **5 files**, weighing **36.9 MiB** with **59 hits** in **DBBC2-DDC**.

Displaying **1** to **5** of **5** files.

DBBC2-DDC

DBBC2 DDC v105
» 3.1 MiB - 6 hits - 20 April 2018
DBBC2 DDC v105

DBBC2_DDC_v106_261118.rar
» 9.9 MiB - 5 hits - 26 November 2018
DBBC2_DDC_v106_261118.rar

DBBC2_DDC_v107_beta1.zip
» 6.8 MiB - 16 hits - 20 November 2018
DBBC2_DDC_v107_beta1.zip

DBBC2_DDC_v107_beta2.rar
» 8.4 MiB - 9 hits - 11 January 2019
DBBC2_DDC_v107_beta2.rar

DBBC2_DDC_v107_beta3.rar
» 8.7 MiB - 23 hits - 30 January 2019
DBBC2_DDC_v107_beta3.rar

Software (Windows XP)

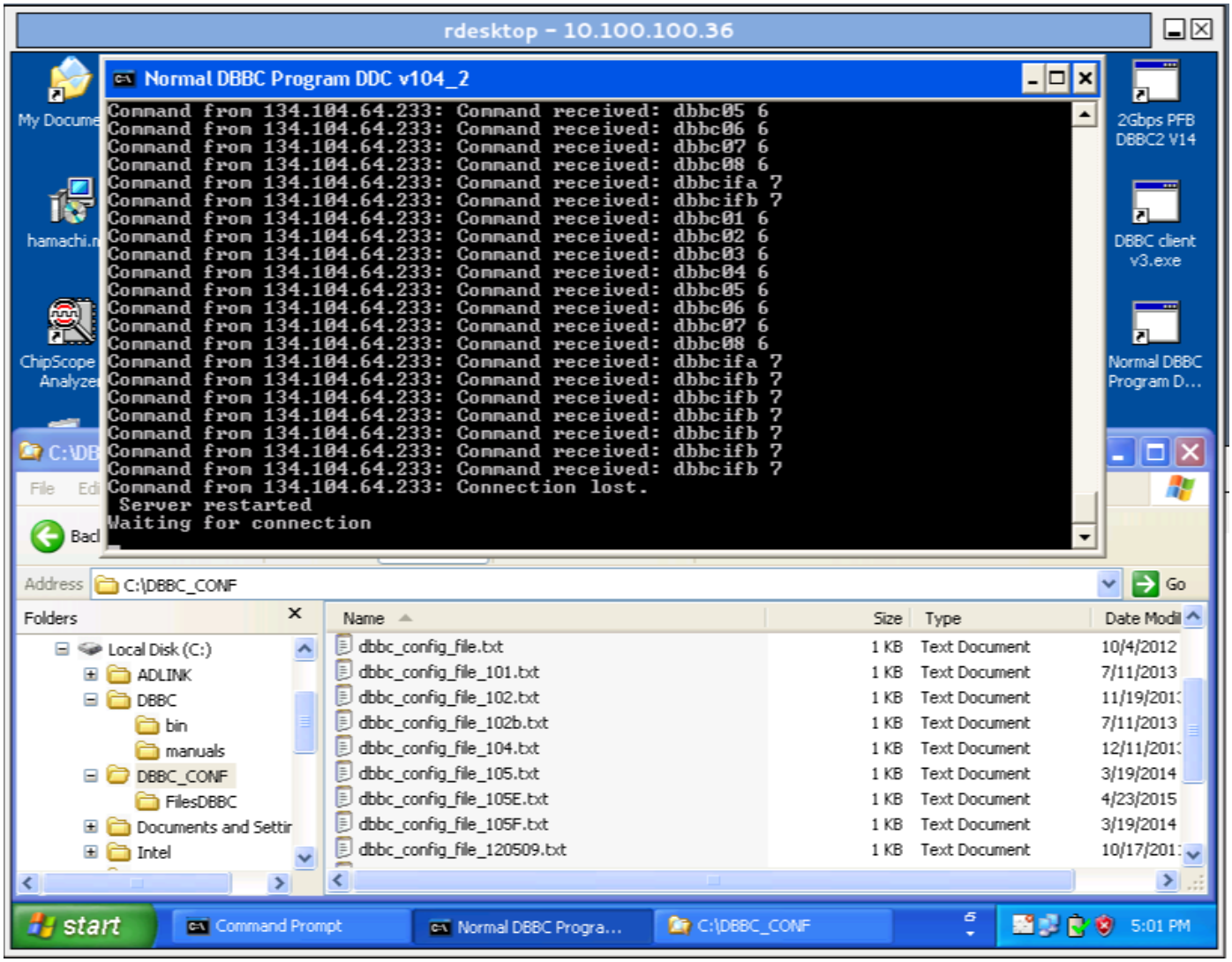
Files Structure:

C:\DBBC\bin
→ control software

C:\DBBC\doc
→ manuals

C:\DBBC_CONF\
→ configuration text files

C:\DBBC_CONF\FilesDBBC
→ firmware



Software

- DDC:

  c:\DBBC\bin\DBBC2 Control DDC v107.exe (server)
  c:\DBBC_conf\dbbc_config_file_107.txt
  c:\DBBC_conf\FilesDBBC\dbbc2_ddc_v107.bit
  c:\DBBC\doc\DBBC2 DDC command set v107.pdf

- PFB:

  c:\DBBC\bin\DBBC2 Control PFB v16_2.exe (server)
  c:\DBBC_conf\dbbc_poly_config_file_16.txt
  c:\DBBC_conf\FilesDBBC\ dbbc2_pfb_v16.bit
  c:\DBBC\doc\DBBC2 PFB command set v16.pdf

DDC configuration file

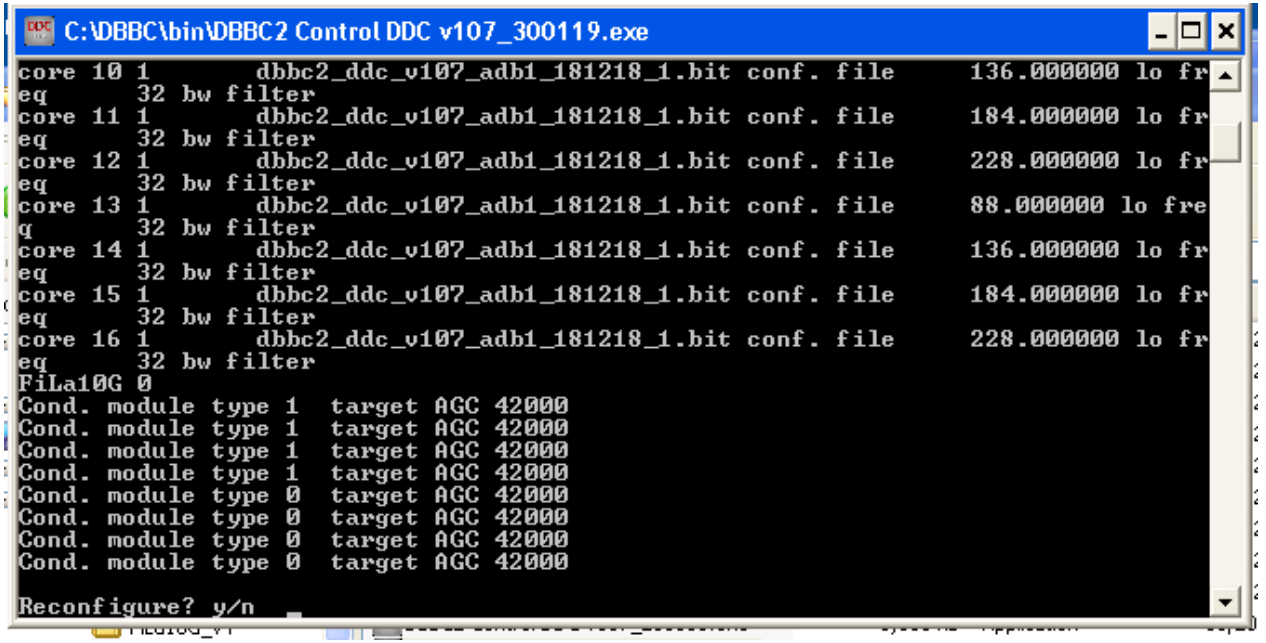
c:\DBBC_conf\dbbc_config_file_v107.txt

Example:

```
1  dbbc2_ddc_v107.bit   597.00   8  ←the first number is indication of ADB1|2, in this case ADB1 is on
1  dbbc2_ddc_v107.bit   682.00   8     IFA  and ADB2 on IFB, ADB1 in IFC, no Core2 for IFD
1  dbbc2_ddc_v107.bit   853.00   8     If no Core2 is inserted in the first and second column put 0.
1  dbbc2_ddc_v107.bit   938.00   8     The second parameter is the firmware file name to be used.
2  dbbc2_ddc_v107.bit   597.00   8     The third and fourth parameters are frequency and bandwidth respectively.
2  dbbc2_ddc_v107.bit   682.00   8
2  dbbc2_ddc_v107.bit   853.00   8
2  dbbc2_ddc_v107.bit   938.00   8
1  dbbc2_ddc_v107.bit   597.00   8
1  dbbc2_ddc_v107.bit   682.00   8
1  dbbc2_ddc_v107.bit   853.00   8
1  dbbc2_ddc_v107.bit   938.00   8
0  dbbc2_ddc_v107.bit   597.00   8     Each Core2 board supports 4 bbcs so if not present 0 has to be inserted in
0  dbbc2_ddc_v107.bit   682.00   8    four lines
0  dbbc2_ddc_v107.bit   853.00   8
0  dbbc2_ddc_v107.bit   938.00   8
1   fila10g_v2_1.bit COM2   ← if installed set 1st version 1 (with ACE), 2nd version (without ACE 2), otherwise 0, ser. port
1 38000                                  ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFA
1 38000                                  ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFB
1 38000                                  ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFC
0 38000                                  ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFD
0 38000                                  ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFE
0 38000                                  ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFF
0 38000                                  ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFG
0 38000                                  ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFH
107 112 90 0                    ←  phase calibration values
CAT2 1024                       ←  CAT1|2 and sampling frequency
PROG 0 3                        ← jtag programmer: 0=xilinx, 1=digilent; prog freq. 3/6 MHz
```

Starting the software

DDC: running **DBBC2 Control DDC v107.exe**



after the Core2 configuration is completed

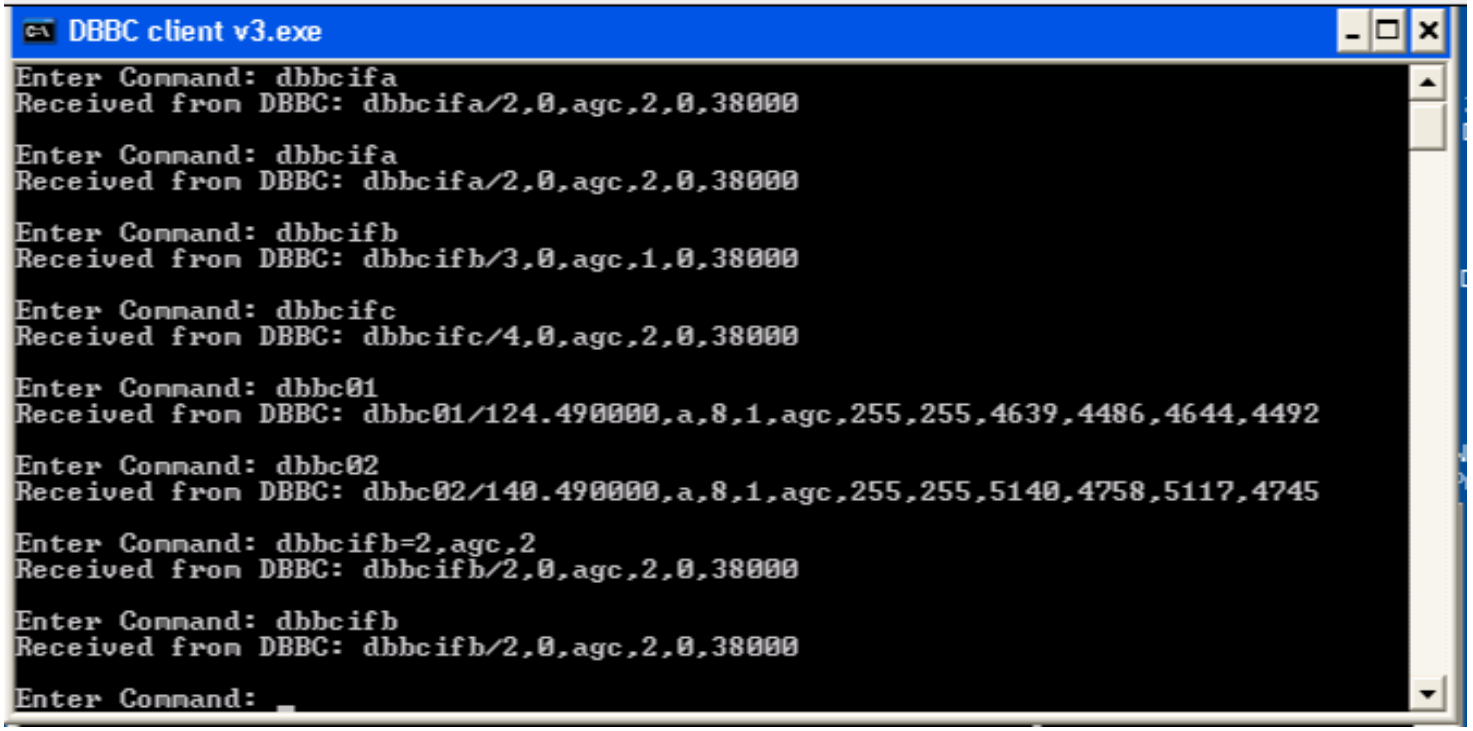
then run a client: **DBBC Client v3.exe or Field System**

DDC Mode Commands and Form Table (see documents)

First tests with the DBBC

> dbbcifa # for query
> dbbcifa=2,agc,2 # to set RF input 2, agc on, IF filter 2 (0-500 MHz)

read out BBCs set different frequencies, ...
> dbbc01 # for query
> dbbc01=596.00,a,16.00 # to set BBC freq=596 MHz, IFA, BBC band width = 16 MHz



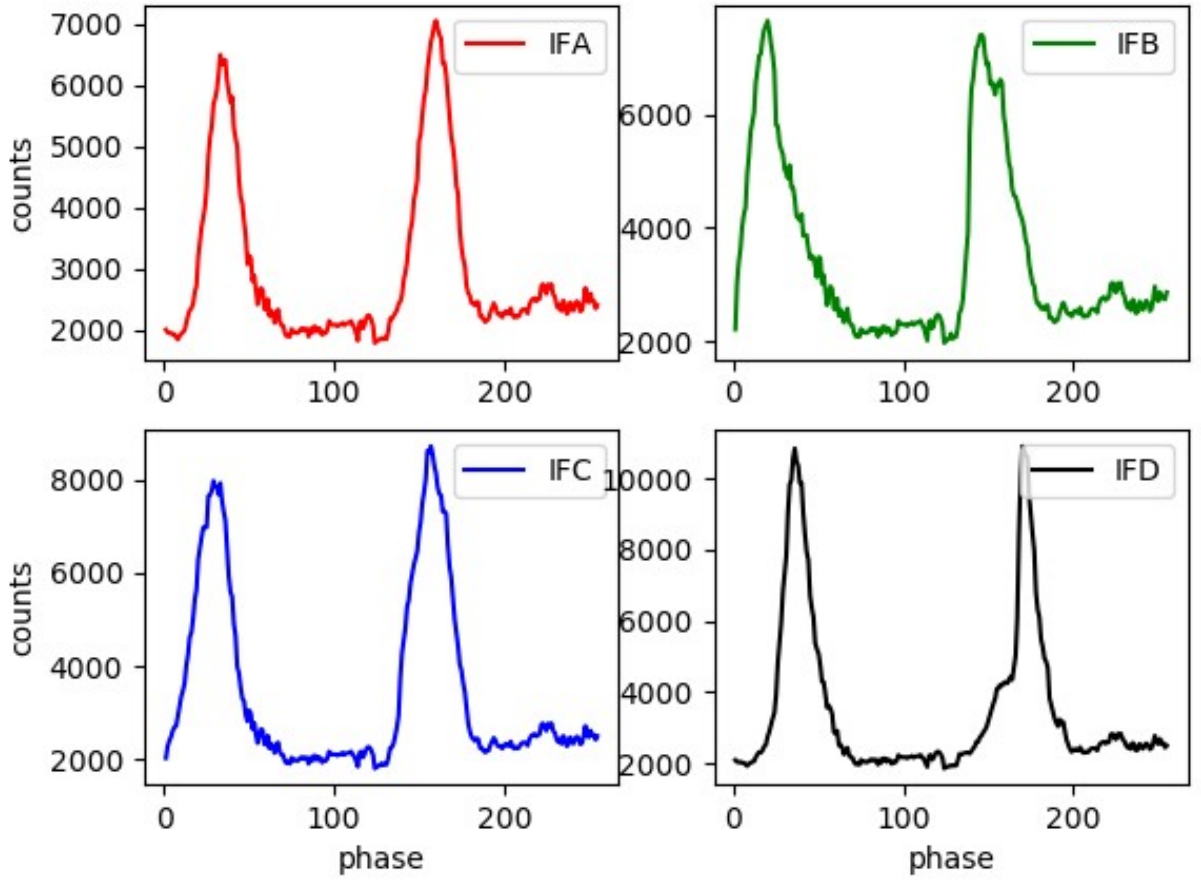
Calibration of the DBBC

Calibration or phase optimization is required at the system installation and has to be repeated after a hardware modification in the stack, transportation, or a new firmware. Periodically as a general check.

- Connect a synthesizer tuned to 764 MHz to all Ifs or a broad band IF signal (not too strong).
- Load the firmware to test.
- Point all dbbcifa,b,c,d to this input and set AGC to manual, e.g.
  - dbbcifa=1,40,1 # adjusted to about 10000 counts
  - In DDC mode: turn off AGC for BBCs: dbbcgain=all,20,20
- Run the DBBC command: calibration=all
- … wait

Description at:
https://deki.mpifr-bonn.mpg.de/GMVA/GMVA_HOWTO/DBBC2_calibration

Calibration of the DBBC



…

252	106	3959	16276	10431
253	135	5588	17455	10729
254	161	5276	18712	11039
255				

minM1 00050 ele1 **107** minM2 00050 ele2 **79** minM3 00049 ele3 **92** minM4 00051 ele4 **124**

….

60	270437	872	261803	16988
61	285347	653	205494	12851
62	289611	395	169170	10302
63	301585	352	144859	7090
64	309365	169	111552	3386
65	317749	102	95884	2313
66	322930	79	79745	1817
67	339064	67	54644	1305
68	332014	57	37490	881
69	338031	55	28940	526
70	324313	54	22799	296
71	320547	52	17611	223
72	310049	51	10504	187
73	276350	51	6440	148
74	260401	51	4751	106
75	251864	51	3334	84
76	204246	51	2061	76
77	169837	51	1407	60
78	149612	51	1155	56
79	97942	51	361	54
80	74886	51	228	53
81	55966	50	130	53
82	46097	51	113	53
83	28929	51	80	53
84	21030	53	69	52
85	7957	55	59	52
86	5530	55	51	52
87	2958	57	51	52
88	2078	61	50	52
89	1368	80	50	52
90	734	79	50	52
91	247	117	50	52

…

DDC configuration file

c:\DBBC_conf\dbbc_config_file_v107.txt

Example:

1 dbbc2_ddc_v107.bit 597.00 8 ←the first number is indication of ADB1|2, in this case ADB1 is on
1 dbbc2_ddc_v107.bit 682.00 8 IFA and ADB2 on IFB, ADB1 in IFC, no Core2 for IFD
1 dbbc2_ddc_v107.bit 853.00 8 If no Core2 is inserted in the first and second column put 0.
1 dbbc2_ddc_v107.bit 938.00 8 The second parameter is the firmware file name to be used.
2 dbbc2_ddc_v107.bit 597.00 8 The third and fourth parameters are frequency and bandwidth respectively.
2 dbbc2_ddc_v107.bit 682.00 8
2 dbbc2_ddc_v107.bit 853.00 8
2 dbbc2_ddc_v107.bit 938.00 8
1 dbbc2_ddc_v107.bit 597.00 8
1 dbbc2_ddc_v107.bit 682.00 8
1 dbbc2_ddc_v107.bit 853.00 8
1 dbbc2_ddc_v107.bit 938.00 8
0 dbbc2_ddc_v107.bit 597.00 8 Each Core2 board supports 4 bbcs so if not present 0 has to be inserted in
0 dbbc2_ddc_v107.bit 682.00 8 four lines
0 dbbc2_ddc_v107.bit 853.00 8
0 dbbc2_ddc_v107.bit 938.00 8
1 fila10g_v2_1.bit COM2 ← if installed set 1st version 1 (with ACE), 2nd version (without ACE 2), otherwise 0, ser. port
1 38000 ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFA
1 38000 ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFB
1 38000 ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFC
1 38000 ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFD
0 38000 ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFE
0 38000 ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFF
0 38000 ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFG
0 38000 ← no unica=0 unica3=1, unica4=2, initial CoMos target values for IFH
100 100 100 100 ← **phase calibration values**
CAT2 1024 ← CAT1|2 and sampling frequency
PROG 0 3 ← jtag programmer: 0=xilinx, 1=digilent; prog freq. 3/6 MHz

Test recordings

- Test recordings are good to control the correct sampling (bit statistics), band pass shape, and pcal tones
- The Mark5B comes with a set of programs that allow to check the bit statistics (bstate), do auto- or cross correlations (vlbi2), and extract phase cal (bpcal).
- More power full are the mark5access programs:
  m5bstate, m5pcal, m5spec, m5timeseries, …
  Available from the EVN TOG wiki pages
  https://deki.mpifr-bonn.mpg.de/Working_Groups/EVN_TOG/DBBC/DBBC_Test_Procedures
- jive5ab allows to stream data directly on a local disk, which avoids to record on diskpacks and use disk2file for small tests.

Test recordings

oper@eff-mark5c-1:~$ m5spec

m5spec ver. 1.3.1 Walter Brisken, Chris Phillips 20120508

A Mark5 spectrometer. Can use VLBA, Mark3/4, and Mark5B formats using the mark5access library.

Usage : m5spec <infile> <dataformat> <nchan> <nint> <outfile> [<offset>]

  <infile> is the name of the input file

  <dataformat> should be of the form: <FORMAT>-<Mbps>-<nchan>-<nbit>,
e.g.:
    VLBA1_2-256-8-2
    MKIV1_4-128-2-1
    Mark5B-512-16-2
    VDIF_1000-64-1-2 (here 1000 is payload size in bytes)

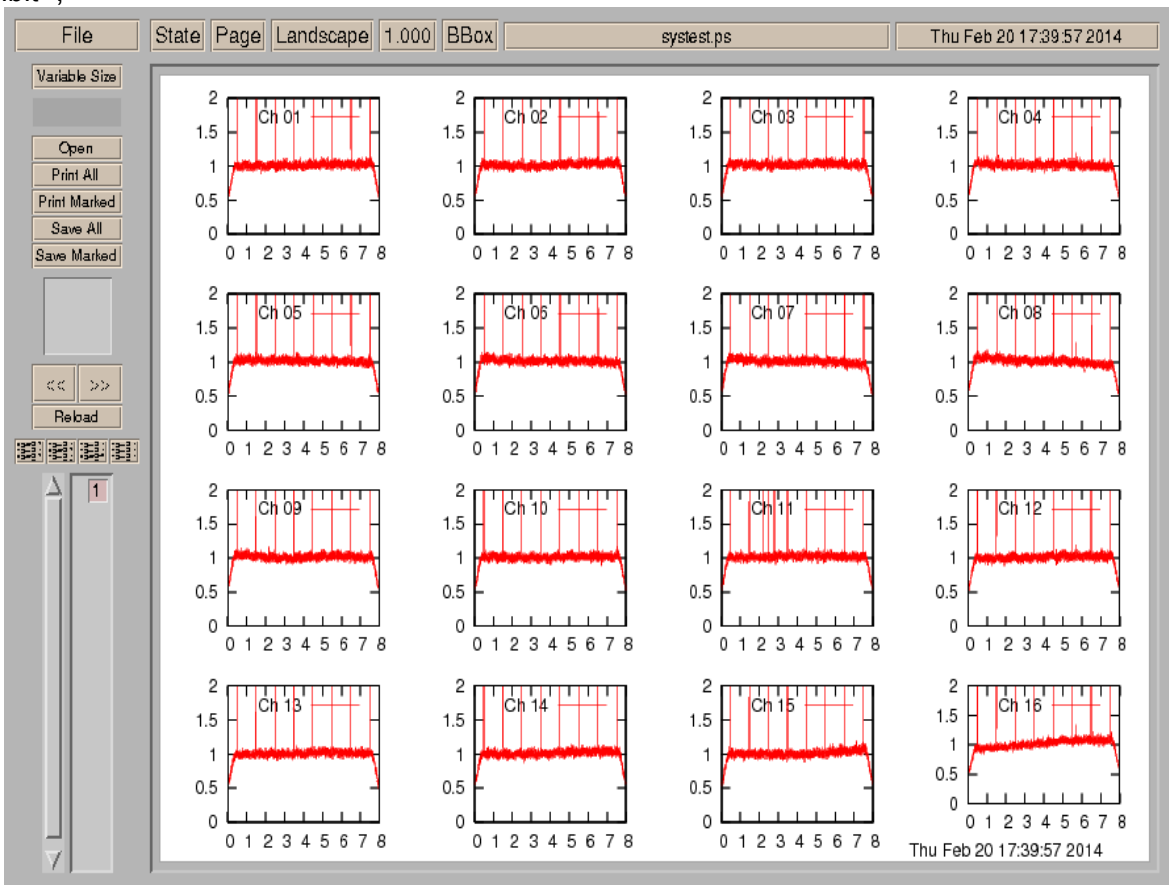
  <nchan> is the number of channels to make per IF

  <nint> is the number of FFT frames to spectrometize

  <outfile> is the name of the output file

  <offset> is number of bytes into file to start decoding

The following options are supported
    -dbbc   Assume dBBC polarisation order (all Rcp then all Lcp)
    -nopol   Do not compute cross pol terms
    -help   This list



Field System integration

- The DBBC2 is fully integrated into the Field System:
  - It supports both PFB and DDC firmware.
  - Continuous calibration in DDC mode.
  - With and without Fila10G.
  - Allows synchronization to internal GPS or NTP on FS-PC.

```
EFLSBERG equipment: Rack=DBBC_DDC/Fil   Recorder=FlexBuff
| Select rack           | Select Rec 1 | Select Rec 2 | Start|
|   1=none              |   1=none     | * 1=none     | * 1  |
|   2=Mark3A            |   2=unused   |   2=unused   |   2  |
|   3=VLBA              |   3=Mark3A   |   3=Mark3A   |      |
|   4=VLBAG             |   4=VLBA     |   4=VLBA     |      |
|   5=VLBA/8            |   5=VLBA4    |   5=VLBA4    |      |
|   6=VLBA4/8           |   6=Mark4    |   6=Mark4    |      |
|   7=Mark4             |   7=S2       |   7=S2       |      |
|   8=VLBA4             |   8=K4-1     |              |      |
|   9=K4-1              |   9=K4-2     |              |      |
|  10=K4-2              |  10=Mark5A   |              |      |
|  11=K4-1/K3           |  11=Mk5APigW |              |      |
|  12=K4-2/K3           |  12=Mark5P   |              |      |
|  13=K4-1/M4           |  13=K5       |              |      |
|  14=K4-2/M4           |  14=Mark5B   |              |      |
|  15=LBA               |  15=Mark5C   |              |      |
|  16=Mark5             | *16=FlexBuff |              |      |
|  17=VLBA5             |              |              |      |
|  18=DBBC_DDC          |              |              |      |
| *19=DBBC_DDC/Fila10g  |              |              |      |
|  20=DBBC_PFB          |              |              |      |
|  21=DBBC_PFB/Fila10g  |              |              |      |
|  22=VLBAC             |              |              |      |
|  23=CDAS              |              |              |      |
 Press <ret> or type 0 for no change.  Else <rack><rec1><rec2><start>
CAUTION! Be sure the schedule works with your choices!
```

Field System integration

- Notes on DBBC2 integration are available in /usr2/fs/misc/dbbc.txt
- There are the typical control-files that need to be adapted for a new backend and one special for the DBBC IP address:
  - *dbbad.ctl* hold the DBBC IP address
  - *equip.ctl* for the FS
  - *skedf.ctl* for DRUDG
  - Some more in point.prc, station.prc, and .Xresources

→ Once this is done the FS should be ready to DRUDG and observe DBBC schedules.

Field System integration

Applications Places — Tue Apr 30, 9:03 AM — VLBI Operator

Field System Log

```
08:51:03&bread/if=bbc15,bbc15
08:51:03&bread/if=bbc16,bbc16
08:51:03&bread/if=pfb,if=core1\,pfb1
08:51:03&bread/if=pfb,if=core2\,pfb2
08:51:03&bread/if=pfb,if=core3\,pfb3
08:51:03&bread/if=pfb,if=core4\,pfb4
08:51:03/bbc01/ 866.490000,a, 8, 1,agc,136,133,11955,11963,11385,11393
08:51:03/bbc02/ 882.490000,a, 8, 1,agc,127,131,11812,11908,11241,11328
08:51:03/bbc03/ 898.490000,a, 8, 1,agc,127,124,11917,11830,11317,11256
08:51:03/bbc04/ 914.490000,a, 8, 1,agc,133,131,12130,12121,11525,11520
08:51:03/bbc09/ 866.490000,c, 8, 1,agc,219,219,10428,10938, 9932,10411
08:51:03/bbc10/ 882.490000,c, 8, 1,agc,219,219,11029,10552,10508,10059
08:51:03/bbc11/ 898.490000,c, 8, 1,agc,218,219,11835,10946,11274,10431
08:51:03/bbc12/ 914.490000,c, 8, 1,agc,219,217,10620,11806,10099,11233
08:51:15;caltsys
08:51:15&caltsys/onsource
08:51:15&caltsys/if=cont_cal,tpicd=tsys,caltsys_man
08:51:15&caltsys/if=cont_cal,,ifagc
08:51:15&caltsys/if=cont_cal,,if=ddc\,bbc_gain=all\\\,agc
08:51:15#antcn#Antenna TRACKING
08:51:15#antcn# RFcentre wanted:  4850 Antenna:  4850 synth1,2= 613.000000    0.000000
08:51:15/onsource/TRACKING
08:51:27/tpi/1l,11399.2,1u,11377.2,2l,11337.0,2u,11246.8,3l,11260.2,3u,11331.2,4l,11537.2
08:51:27/tpi/4u,11536.5,ia,1838.33
08:51:27/tpi/9l,11316.2,9u,11220.5,al,11290.0,au,11320.8,bl,11233.2,bu,11298.0,cl,11264.2
08:51:27/tpi/cu,11267.5,ic,1787.60
08:51:27/tpical/1l,11976.0,1u,11960.8,2l,11921.2,2u,11820.0,3l,11842.8,3u,11914.2,4l,12124.0
08:51:27/tpical/4u,12129.0
08:51:27/tpical/9l,11888.8,9u,11788.2,al,11855.5,au,11886.2,bl,11793.2,bu,11867.2,cl,11837.2
08:51:27/tpical/cu,11835.8
08:51:27/caltemp/1l,1.836,1u,1.840,2l,1.841,2u,1.837,3l,1.838,3u,1.846,4l,1.854,4u,1.890
08:51:27/caltemp/9l,1.817,9u,1.812,al,1.806,au,1.797,bl,1.793,bu,1.801,cl,1.817,cu,1.833
08:51:27/tsys/1l,36.3,1u,35.9,2l,35.7,2u,36.0,3l,35.5,3u,35.9,4l,36.4,4u,36.8
08:51:27/tsys/9l,35.9,9u,35.8,al,36.0,au,36.0,bl,36.0,bu,35.7,cl,35.7,cu,36.3
```

Mark6-VLBI-Monitor

```
r  V0.997
alized value $mk6state in string eq at /usr2/oper/bin/zl
 line 537.    :  laeuft                          |  OK
c zu 1pps     :                                  |  OK
t korrekt     :                                  |  OK
det Daten     :  e-VLBI stoppedessenen!          |  OK
--------------  Mark6-System  ---------------------------
              :  laeuft                          |  OK
.....................................            |  0 %
                                                 |  e-VLBI
ler:
el:                                              |  OK
IES:                                             |  ULO OK
sr2/log/em135bef.log
-------------------------------------------------------
 30.Apr. 2019     Day 120                 09:03:40
-------------------------------------------------------
it:  q (15 sec Verzoegerung)     Feb 2016  by AK/DG/UB
```

System Temperatur

Tsys	0.0 (IFA)		(IFB)
	0.0 (IFC)		(IFD)
BBC	Freq	Ts-U	Ts-L
01	866.49	35.9	36.3
02	882.49	36.0	35.7
03	898.49	35.9	35.5
04	914.49	36.8	36.4
05			
06			
07			
08			
09	866.49	35.8	35.9
10	882.49	36.0	36.0
11	898.49	35.7	36.0
12	914.49	36.3	35.7
13			
14			
15			
16			

System Status

```
EFLSBERG        2019.120.09:03:48  UT          TEMP     9.5C 3c48        TRACKING
MODE RATE               09:06:11  NEXT        HUMID  83.17% RA  01h37m41.3s
                    SCHED=none     LOG=station PRES  984.8mb DEC 33d09m   (2000)
                       TSYS: IFA IFB IFC IFD  CABLE 0.000000s AZ  125.8  EL  65.5
                             0   0   0   0

NO CHECK: rx
```

ERRORS

Operator Input

```
>proc=n19c1ef
>setup01
>3c48
>iread
>bread
>caltsys
>
```

DBBC

Field System integration

Applications Places Tue Apr 30, 9:05 AM VLBI Operator

Field System Log

gv: /tmp/systest.ps (on Mark6-4040)

File | State | Page | Landscape | 1.000 | BBox | /tmp/systest.ps | Tue Apr 30 09:05:33 2019

Variable Size | Open | Print All | Print Marked | Save All | Save Marked | << | >> | Reload

RCP 1l | RCP 1u | RCP 2l | RCP 2u
LCP 5l | LCP 5u | LCP 6l | LCP 6u
RCP 3l | RCP 3u | RCP 4l | RCP 4u
LCP 7l | LCP 7u | LCP 8l | LCP 8u

Warning: Cannot convert string "-efont-biwidth-medium-r-normal--16-*-*-*-*-*-iso10646-1,-gnu-unifont-medium-r-normal--16-*-*-*-*-*-iso10646-1,-adobe-helvetica-medium-r-normal--14-*-*-*-*-*-*-*,-jis-fixed-medium-r-*--16-*-*-*-*-*-jisx0208.1983-0,-*-*-medium-r-*--16-*-*-*-*-*-*-*,*" to type FontSet

I-Server (Wichtig nicht schliessen! r wiederholt den letzten Befehl)

Mark6-VLBI-Monitor

```
r  V0.997
alized value $mk6state in string eq at /usr2/oper/bin/zl
 line 537.     :  laeuft                          |  OK
c zu 1pps      :                                  |  OK
t korrekt      :                                  |  OK
det Daten      :  e-VLBI stoppedessenen!          |  OK
-------------  Mark6-System  ----------------------------
               :  laeuft                          |  OK
.....................................             |  0 %
                                                  |  e-VLBI
ler:
el:                                               |  OK
IES:                                              |  ULO OK
sr2/log/em135bef.log
-------------------------------------------------------------
 30.Apr. 2019      Day 120                  09:05:46
-------------------------------------------------------------
it:  q (15 sec Verzoegerung)      Feb 2016  by AK/DG/UB
```

System Temperatur

Tsys	0.0 (IFA)		(IFB)
	0.0 (IFC)		(IFD)
BBC	Freq	Ts-U	Ts-L
01	866.49	35.9	36.3
02	882.49	36.0	35.7
03	898.49	35.9	35.5
04	914.49	36.8	36.4
05			
06			
07			
08			
09	866.49	35.8	35.9
10	882.49	36.0	36.0
11	898.49	35.7	36.0
12	914.49	36.3	35.7
13			
14			
15			
16			

System Status

```
EFLSBERG        2019.120.09:05:50  UT          TEMP     9.6C 3c48        TRACKING
MODE RATE              09:10:30    NEXT        HUMID   79.90% RA  01h37m41.3s
              SCHED=none           LOG=station PRES   984.7mb DEC 33d09m   (2000)
                TSYS: IFA IFB IFC IFD  CABLE 0.000000s AZ  126.6  EL  65.8
                       0   0   0   0
NO CHECK: rx
```

ERRORS

Operator Input

```
>setup01
>3c48
>iread
>bread
>caltsys
>mk5=record=test_ef_no0430
>mk5=record=on:test_ef_no0430
>mk5=record?
>mk5=record?
>mk5=record=off
```

Field System integration

Applications Places Tue Apr 30, 9:05 AM VLBI Operator

Field System Log

```
  sample rate = 16000000 Hz
  offset = 0
  framebytes = 8032 bytes
  datasize = 8000 bytes
  sample granularity = 1
  frame granularity = 1
  gframens = 125000
  payload offset = 32
  read position = 0
  data window size = 1048576 bytes

Ch    --      -     +     ++        --      -      +     ++     gfact
 0  138368  262632  262243  136757      17.3   32.8   32.8   17.1   1.04
 1  137356  263934  262360  136350      17.2   33.0   32.8   17.0   1.04
 2  140120  261046  260263  138571      17.5   32.6   32.5   17.3   1.03
 3  152449  249598  268300  129653      19.1   31.2   33.5   16.2   1.02
 4  138858  263584  261868  135690      17.4   32.9   32.7   17.0   1.04
 5  138883  263806  261438  135873      17.4   33.0   32.7   17.0   1.04
 6  138185  263866  261590  136359      17.3   33.0   32.7   17.0   1.04
 7  139228  263236  256358  141178      17.4   32.9   32.0   17.6   1.03
 8  135818  262091  263365  138726      17.0   32.8   32.9   17.3   1.04
 9  136270  261450  263638  138642      17.0   32.7   33.0   17.3   1.04
10  135398  262259  262153  140190      16.9   32.8   32.8   17.5   1.04
11  104176  293127  250359  152338      13.0   36.6   31.3   19.0   1.09
12  135139  261104  264012  139745      16.9   32.6   33.0   17.5   1.04
13  135031  261317  263550  140102      16.9   32.7   32.9   17.5   1.04
14  135647  260682  263579  140092      17.0   32.6   32.9   17.5   1.04
15  119476  277567  263587  139370      14.9   34.7   32.9   17.4   1.08
800000 / 800000 samples unpacked
8192000 / 8192000 samples unpacked
Warning: Missing charsets in String to FontSet conversion
Warning: Cannot convert string "-efont-biwidth-medium-r-normal--16-*-*-*-*-*-iso10646-1,-gnu-unifont
-medium-r-normal--16-*-*-*-*-*-iso10646-1,-adobe-helvetica-medium-r-normal--14-*-*-*-*-*-*-*,-jis-fi
xed-medium-r-*--16-*-*-*-*-*-jisx0208.1983-0,-*-*-medium-r-*--16-*-*-*-*-*-*-*,*" to type FontSet
```

BI-Server (Wichtig nicht schliessen! r wiederholt den letzten Befehl)

Mark6-VLBI-Monitor

```
r  V0.997
alized value $mk6state in string eq at /usr2/oper/bin/zl
 line 537.    :  laeuft                          |  OK
c zu 1pps     :                                  |  OK
t korrekt     :                                  |  OK
det Daten     :  e-VLBI stoppedessenen!          |  OK
-------------  Mark6-System  ----------------------
              :  laeuft                          |  OK
.....................................................|  0 %
                                                 |  e-VLBI
ler:
el:                                              |  OK
IES:                                                ULO OK
sr2/log/em135bef.log
-------------------------------------------------------
 30.Apr. 2019     Day 120                  09:05:54
-------------------------------------------------------
it:  q (15 sec Verzoegerung)     Feb 2016  by AK/DG/UB
```

System Temperatur

Tsys	0.0	(IFA)		(IFB)
	0.0	(IFC)		(IFD)
BBC	Freq	Ts-U	Ts-L	
01	866.49	35.9	36.3	
02	882.49	36.0	35.7	
03	898.49	35.9	35.5	
04	914.49	36.8	36.4	
05				
06				
07				
08				
09	866.49	35.8	35.9	
10	882.49	36.0	36.0	
11	898.49	35.7	36.0	
12	914.49	36.3	35.7	
13				
14				
15				
16				

System Status

```
EFLSBERG        2019.120.09:05:59  UT          TEMP     9.6C 3c48        TRACKING
MODE RATE                09:10:30  NEXT        HUMID  80.15% RA  01h37m41.3s
                    SCHED=none     LOG=station PRES  984.7mb DEC 33d09m   (2000)
                       TSYS: IFA IFB IFC IFD  CABLE 0.000000s AZ  126.7  EL  65.8
                             0   0   0   0

NO CHECK: rx
```

ERRORS

Operator Input

```
>setup01
>3c48
>iread
>bread
>caltsys
>mk5=record=test_ef_no0430
>mk5=record=on:test_ef_no0430
>mk5=record?
>mk5=record?
>mk5=record=off
```

DBBC2 resources

- DBBC2 software, firmware and documents:
  http://www.hat-lab.com (until 2018)
  https://www.hat-lab.cloud
- DBBC2 installation, testing, and operational notes:
  https://deki.mpifr-bonn.mpg.de/Working_Groups/EVN_TOG
  https://deki.mpifr-bonn.mpg.de/GMVA

DBBC3 Operations
TOW 2021

Sven Dornbusch
Max-Planck-Institut für Radioastronomie

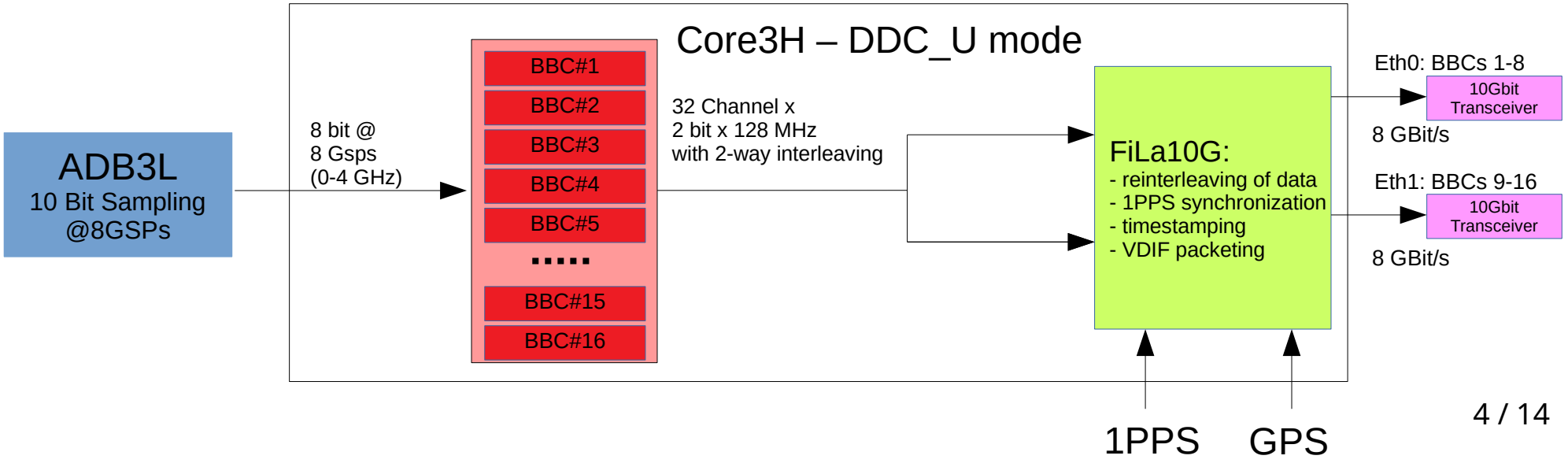
Contents

Observation Modes

- **DSC** (Direct Sampling Conversion) full 4GHz Bandwidth/IF
- **OCT** (Octopus), provides single (OCT_S) or double (OCT_D) 32-tap FIR-Filter/IF
  Available Bandpass-Filters:
  512 MHz BW: 0-512, 512-1024, ..., 3584-4096
  1024 MHz BW: 0-1024, 1024-2048, ..., 3072-4096
  2048 MHz BW: 0-2048, 2048-4096
- **DDC** (Digital Tunable Downconversion): provides up to 16 BBCs/IF with fully tunable frequency, selectable BW of 2, 4, 8, 16, 32, 64 and 128 MHz
- Each observation mode has its own control software/firmware/ set of configuration files.
- Latest versions can be downloaded at https://www.hat-lab.cloud/

DDC Mode

- **D**igital Tunable **D**own**c**onversion
- **DDC_L (legacy)**: selectable bandwidth of 2, 4, 8, 16 and 32 MHz.
  Up to 8 BBCs/IF → max. 64 BBCs per System
  not supported any more, replaced by DDC_U.
- **DDC_V (VGOS)**: 32 MHz filters with broader bandshape optimized for VGOS observations
  8 BBCs/IF → max. 64 BBCs per System
- **DDC_U (unified)**: selectable bandwidth of 2, 4, 8, 16, 32, 64 and 128 MHz.
  16 BBCs/IF → max. 128 BBCs per System



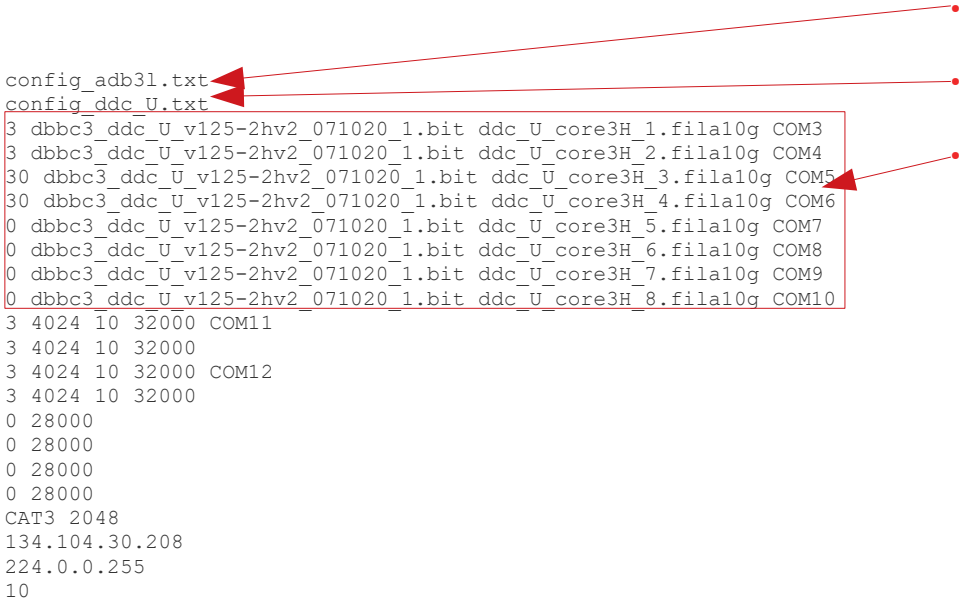
Configuration and Setup for DDC_U

- **Control Software** (in Folder *C:\DBBC\bin*):
  „*DBBC3 Control_DDC_V_v125.exe*“
- **Config Files** (in Folder *C:\DBBC_CONF*):
  - one main config file „*dbbc3_config_file_ddc_V_125.txt*“
  - one config file for the ADB3L-Sampler settings „*config_adb3l.txt*“
  - for each Core3H a separate config file: „*ddc_U_core3H_1.fila10g*“, „*ddc_U_core3H_2.fila10g*“, ...
  - a config file with the BBC settings (frequencies and bandwidth): „*config_ddc_U.txt*“
- **Firmware** (in Folder *C:\DBBC_CONF\FilesDBBC*):
  - one bit-File (FPGA-Firmware): „*dbbc3_ddc_U_v125-2hv2_071020_1.bit*“
- **Documentation** (in Folder *C:\DBBC\manuals*):
  - Command sets for Control Software, ADB3L and Core3H
  - Description of the Setup Procedure
  - Changelogs for new versions

Control Software

- Loads the Firmware for the Core3H
- Starts the GCAT Clock Synthesizer
- Initializes and Synchronizes the ADB3L Samplers, loads settings for offset, gain, and delay.
- Initializes and configures the Core3H
- After initialization is finished, the control software provides a socket connection for communication using a socket client or the Field System.
- Through the control software direct communication with the GCoMo, ADB3L, Core3H and Synthesizer for the Downconversion is possible. A few examples:
  - core3h=1,time (check timestamp of the first Core3H-Board)
  - adb3l=offset=1,0,128 (set offset of the first sampler on the first ADB3L-Board)
  - synth=1,lock (check if the first LO-synthesizer is locked)
- Starting with DDC_U version 125 the control software supports multicast, for continuous monitoring of the system status by multiple clients. This gives out a multicast package every second with all the status information about the DBBC3. The multicast is used by the FS to monitor the DBBC3. A simple python-client is provided with the package as an example to write your own clients.

Main Config-File for DDC_U



Config-file for Sampler settings

Config-file for BBC frequencies and BW

Configuration for Core3H-Boards:

- `3 dbbc3_….020_1.bit ddc_U_core3H_1.fila10g COM3`
  - 3 – board present and signal connected to IF
  - 30 – board present and no signal connected to IF
  - 0 – no board present
- `3 dbbc3_….270218.bit ddc_U_core3H_1.fila10g COM3`
  - Firmware for the Core3H, located in the folder: „C:\DBBC_CONF\FilesDBBC“
- `3 dbbc3_….270218.bit ddc_U_core3H_1.fila10g COM3`
  - Config-File for this Core3H
- `3 dbbc3_….270218.bit ddc_U_core3H_1.fila10g COM3`
  - COM-Port for serial communication with Core3H

Main Config-File for DDC_U (2)

Configuration for GcoMos / Downconversion
- 3 4024 10 32000 COM11
  - 3 – GCoMo with internal synthesizer
  - 2 – GCoMo without internal synthesizer
  - 0 – No GCoMo present in that slot
- 3 4024 10 32000 COM11
  - synthesizer frequency (LO * 0.5)
- 3 4024 10 32000 COM11
  - synthesizer attenuation in dBm
- 3 4024 10 32000 COM11
  - AGC target value
- 3 4024 10 32000 COM11
  - COM-Port for serial com. with synthesizer

GCAT Type and sampler clock frequency

The DBBC3s IP-Address in the network (for multicast)

IP-Address for Multicast-Group

Maximum number of checks for initial phase check routine

ADB3L Config-File

```
bistoff=1
bistoff=2
reset
SDA_on=1,0
SDA_on=1,1
SDA_on=1,2
SDA_on=1,3
SDA_on=2,0
SDA_on=2,1
SDA_on=2,2
SDA_on=2,3
delay=1,0,268
delay=1,1,461
delay=1,2,564
delay=1,3,958
offset=1,0,108
offset=1,1,120
offset=1,2,158
offset=1,3,134
gain=1,0,118
gain=1,1,130
gain=1,2,121
gain=1,3,139
delay=2,0,335
delay=2,1,439
delay=2,2,585
delay=2,3,951
offset=2,0,124
offset=2,1,122
offset=2,2,122
offset=2,3,110
gain=2,0,135
gain=2,1,118
gain=2,2,118
gain=2,3,128
reseth
```

Static part

Calibration settings for each sampler
- format: command=board,sampler,value
- boards numbered from 1 to 8
- samplers numbered from 0 to 3
- range of values depend on the command
    - delay: 0-1023
    - offset: 0-255
    - gain: 0-255

Core3H Config-File for DDC_U mode

```
core3_init
core3_mode pfb
regwrite core3 0 0x10101010
regwrite core3 1 0xBFBFBFBF
regwrite core3 9 1
reboot
inputselect vsi1-2-3-4
vsi_samplerate 128000000
splitmode on
vsi_bitmask 0xFFFFFFFF 0xFFFFFFFF 0xFFFFFFFF 0xFFFFFFFF
reset
vdif_frame 2 16 8192 ct=off
tengbcfg eth0 ip=192.168.1.16 gateway=192.168.1.1 nm=27
tengbcfg eth1 ip=192.168.1.17 gateway=192.168.1.1 nm=27
tengbarp eth0 2 00:60:dd:44:47:60
tengbarp eth1 3 00:60:dd:44:47:61
destination 0 192.168.1.2:46220
destination 1 192.168.1.3:46221
timesync
start vdif
sysstat
```

Static part, don't change any code here

VDIF-Settings:

`vsi_samplerate 128000000`

frequency never changes, decimation needs to be set for different BBC Bandwidths

- vsi_samplerate 128000000 (for 128 MHz)
- vsi_samplerate 128000000 2 (for 64 MHz)
- vsi_samplerate 128000000 4 (for 32 MHz)
- vsi_samplerate 128000000 8 (for 16 MHz)
- vsi_samplerate 128000000 16 (for 8 MHz)
- vsi_samplerate 128000000 32 (for 4 MHz)
- vsi_samplerate 128000000 64 (for 2 MHz)

`vsi_bitmask 0xFFFFFFFF 0xFFFFFFFF …` (optional)

due to 2-way interleaving the bitmasks for the 32 channels (16 BBCs with USB/LSB each) need to be doubled! So from left to right:
first bitmask (BBCs 16-9, LSB+USB each)
second bitmask (BBCs 8-1, LSB+USB each)
third bitmask (copy of first bitmask)
fourth bitmask (copy of second bitmask)
all bitmask need to have the same number of bits!

`vdif_frame 2 16 8192 ct=off`

vdif-configuration for one output stream
both output streams (eth0 and eth1) need to have the same vdif-configuration!
the parameter ct=off is mandatory!

Core3H Config-File for DDC_U mode (2)

```
core3_init
core3_mode pfb
regwrite core3 0 0x10101010
regwrite core3 1 0xBFBFBFBF
regwrite core3 9 1
reboot
inputselect vsi1-2-3-4
vsi_samplerate 128000000
splitmode on
vsi_bitmask 0xFFFFFFFF 0xFFFFFFFF 0xFFFFFFFF 0xFFFFFFFF
reset
vdif_frame 2 16 8192 ct=off
tengbcfg eth0 ip=192.168.1.16 gateway=192.168.1.1 nm=27
tengbcfg eth1 ip=192.168.1.17 gateway=192.168.1.1 nm=27
tengbarp eth0 2 00:60:dd:44:47:60
tengbarp eth1 3 00:60:dd:44:47:61
destination 0 192.168.1.2:46220
destination 1 192.168.1.3:46221
timesync
start vdif
sysstat
```

Ethernet Configuration:

`tengbcfg eth0 ip=...`

IP settings for each output (eth0 and eth1)

with `mac=...` you can assign a source MAC address

`tengbarp eth0 2 00:60:dd:44:47:60`

ARP table for each output

first parameter is the ethernet output to configure

second parameter is the subnet (usually last value of the corresponding destination IP address)

third parameter is the assigned target MAC address

`destination 0 192.168.1.2:46220`

configure destination IP address and port for specified ethernet output (0 or 1)

Static part, don't change any code here

`timesync`

performs time synchronization with GPS

if GPS module is not connected this process will fail!

`start vdif`

starts the vdif ouput stream over ethernet

`sysstat`

prints out system status

Additional Config-Files

- config_ddc_U.txt
  - contains frequency and bandwidth for each BBC:

```
1  700.0 32
2  700.0 32
3  700.0 32
4  700.0 32
5  700.0 32
6  700.0 32
7  700.0 32
8  700.0 32
9  700.0 32
10 700.0 32
11 700.0 32
12 700.0 32
13 700.0 32
14 700.0 32
15 700.0 32
16 700.0 32
...
```

Setup Procedure

1) Make necessary changes to config-files:
   - main config-file: select the IFs that should be used, set the frequencies for the downconversion, …
   - Core3H config-files: configuration of the ethernet ports, correct MAC-Addresses, vdif_frame configuration, …
   - DDC-Config-File: adjust the frequencies and BW for the BBCs

2) Start the control Software:
   - You will be prompted: `"Configure y/n? "`, press „y“
   - The firmware will be loaded and the system initialized
   - This can take some time, around 30 min for a system with full stack (eight IFs).
   - After the initialization and configuration the control software will be ready for a client to connect and starts to send the multicast packages

3) Use python routine the verify proper system status
   Example: `python ./setupDBBC3_DDC_U.py -n 4 134.104.30.223`
   will check system status for a DBBC3 with four boards using the specified IP address of the DBBC3

4) System is ready for observation, FS can connect

DBBC3 Python Package

- Available on github: https://github.com/mpifr-vlbi/dbbc3
- Contains Python Scripts and object oriented Library to monitor and control the DBBC3
  - “low-level“ implementation of most DBBC3 commands
  - „high-level“ validation routines
  - Utility scripts to interact with the DBBC3
- Available Utility Scripts:
  - `dbbc3client.py` – simple client to send commands to the DBBC3
  - `dbbc3ctl.py` – script for performing higher-level checks and tasks
  - `setupDBBC3_DDC_U.py` – validation script for DDC_U mode
- DBBC3Multicast class supports parsing of multicast messages, example script for multicast is also included.