K5 Software Correlator Manual

T.KONDO/NICT

revised on March 6, 2017

Contents

1 Installation of the package 4
   1.1 Installation from an archive file ............................................. 4
   1.2 Installation step by step ....................................................... 4
   1.3 Environment variables .......................................................... 5
   1.4 Directory structure after installation ....................................... 6
   1.5 Set up your environment ....................................................... 7

2 Software programs list 8
   2.1 A-priori calculation ............................................................... 8
   2.2 Correlation processing ............................................................ 8
   2.3 Data check ................................................................. 8
   2.4 Data Format Conversion ....................................................... 8

3 A-priori calculation software 10
   3.1 apri_calc ................................................................. 10
      3.1.1 How to execute ............................................................. 10
      3.1.2 Examples (monitoring schedule) ......................................... 14
      3.1.3 Examples (in case of K5/VSSP) .......................................... 17
      3.1.4 Example (in case of VDIF Format Data) ................................ 21
      3.1.5 Example (in case of Mark-5B format Data) ........................... 22
      3.1.6 Example (in case of VSSP format for X station and VDIF format Y station) 23
   3.2 skdchk ................................................................. 24
      3.2.1 How to execute ............................................................. 24
      3.2.2 Examples ................................................................. 24

4 Correlation software 26
   4.1 fx_cor, fx_cor_new, cor, cor_new ............................................ 26
      4.1.1 How to execute ............................................................. 26
      4.1.2 Example: “cor” ............................................................. 30
      4.1.3 Example: “fx_cor_new” (Mark5B data) ................................ 32
   4.2 fx_cor_all, fx_cor_all_new, cor_all, cor_all_new ................................ 35
      4.2.1 How to execute ............................................................. 35
   4.3 sdelay ................................................................. 37
      4.3.1 How to execute ............................................................. 37
      4.3.2 Example ................................................................. 38
      4.3.3 Output file format .......................................................... 45
   4.4 cor_mon ................................................................. 46
      4.4.1 How to execute ............................................................. 46
      4.4.2 Example ................................................................. 46
5 Data check utilities
5.1 oscillo .................................................. 48
   5.1.1 How to execute .................................. 48
   5.1.2 Example ......................................... 48
5.2 speana ................................................. 48
   5.2.1 How to execute .................................. 48
   5.2.2 Example ......................................... 49
5.3 gspeana ................................................ 50
   5.3.1 How to execute .................................. 50
   5.3.2 Example ......................................... 51
5.4 datachk ............................................... 52
   5.4.1 How to execute .................................. 52
   5.4.2 Summary file .................................... 53
   5.4.3 Error log file ................................... 53
5.5 vdiffcheck ............................................. 54
   5.5.1 How to execute .................................. 54
   5.5.2 Example ......................................... 54
5.6 m5check ............................................... 55
   5.6.1 How to execute .................................. 55
   5.6.2 Examples ....................................... 55

6 Data format conversion ................................. 57
6.1 k5tom5b ............................................... 57
   6.1.1 How to execute .................................. 57
   6.1.2 Example ......................................... 57
6.2 k5tom5 ................................................ 58
   6.2.1 How to execute .................................. 58
   6.2.2 Examples ....................................... 59
6.3 k5tovdif ................................................ 62
   6.3.1 How to execute .................................. 62
6.4 ads2k5 ................................................ 63
   6.4.1 How to execute .................................. 63
6.5 m5btok5 ............................................... 64
   6.5.1 How to execute .................................. 64
6.6 m5tok5 ................................................ 65
   6.6.1 How to execute .................................. 65
6.7 vdiff2k5 ............................................... 67
   6.7.1 How to execute .................................. 67
6.8 vdiff2m5b .............................................. 68
   6.8.1 How to execute .................................. 68

7 Actual correlation processing .......................... 69
7.1 Fringe search ........................................... 69
7.2 Processing all scans ................................... 74
7.3 Correlation processing for a variety of data formats (VDIF, Mark5B, ADS, OCTAD) ............... 74

8 Update history ........................................... 75

A A-priori file format ..................................... 76

B K5 software correlator output format (Format 7) ...... 84

C KSP correlator output format including extension .... 86
   C.1 Correlation data format: header record (HD) .......... 87
   C.2 Correlation data file format (in case of CRSMODE=“F”):
       correlation data by unit (UD) .......................... 89
   C.3 Correlation data file format (in case that CRSMODE is not “F”):
       correlation data by unit (UD) .......................... 92
### D VLBI data format

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>D.1 Type of data format</td>
<td>94</td>
</tr>
<tr>
<td>D.2 K5/VSSP, VSSP32 and VSSP64 format</td>
<td>94</td>
</tr>
<tr>
<td>D.2.1 Data structure</td>
<td>94</td>
</tr>
<tr>
<td>D.2.2 VSSP header format</td>
<td>95</td>
</tr>
<tr>
<td>D.2.3 VSSP32 and VSSP64 header format (general specifications)</td>
<td>95</td>
</tr>
<tr>
<td>D.2.4 Predefined and reserved format numbers</td>
<td>97</td>
</tr>
<tr>
<td>D.2.5 VSSP data block</td>
<td>101</td>
</tr>
<tr>
<td>D.2.6 Data block format for Format #21 and #22 (extended format)</td>
<td>101</td>
</tr>
<tr>
<td>D.3 Mark5B data format</td>
<td>102</td>
</tr>
<tr>
<td>D.3.1 Data structure</td>
<td>102</td>
</tr>
<tr>
<td>D.3.2 Mark5B header format</td>
<td>102</td>
</tr>
<tr>
<td>D.3.3 Mark5B data block</td>
<td>102</td>
</tr>
<tr>
<td>D.4 VDIF data format</td>
<td>104</td>
</tr>
<tr>
<td>D.4.1 Data structure</td>
<td>104</td>
</tr>
<tr>
<td>D.4.2 VDI header format</td>
<td>104</td>
</tr>
<tr>
<td>D.4.3 VDIF data block</td>
<td>104</td>
</tr>
</tbody>
</table>

### E How to install PGPLOT

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>E.1 by “apt-get” (for Ubuntu 14.04)</td>
<td>107</td>
</tr>
<tr>
<td>E.2 step by step</td>
<td>107</td>
</tr>
</tbody>
</table>

### F How to install FFTW

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>109</td>
</tr>
</tbody>
</table>
1 Installation of the package

K5 software correlator was developed to correlate VLBI data with VSSP format observed by a K5/VSSP sampler. But it can now process any kind of data format, such as, Mark-5B, VDIF, ADS, OCTAD besides VSSP format. Software correlator is written in C, so that it can run on most of Linux distributions. PCPLOT or GNUPlot is used as a graphics package. Therefore PGPLOT or GNUPlot should be installed in advance. Some software programs, e.g., “fx_cor” and “fx_cor_new”, support the use of FFTW package which improves throughput significantly. If you want to use FFTW, FFTW3 package should be installed in advance. Please see Appendices E and F for how to install these packages.

1.1 Installation from an archive file

Put an archive file “ipvlbi_corYYYYMMDD.tar.gz” on an adequate directory, where YYYYMMDD is archive date (year, month, day) such as 20170119, then decompress as follows.

```
tar xzvf ipvlbi_corYYYYMMDD.tar.gz
```

As the directory named “ipvlbiYYYYMMDD” is created under the current directory, change the working directory to “ipvlbiYYYYMMDD” as follows.

```
cd ipvlbiYYYYMMDD
```

Execute “make” as follows.

```
make T=COR [F=FFTW] [G=GNUPlot]
```

Where F=FFTW --- use FFTW package for FFT (default is FFT included in the package)

G=GNUPlot --- use GNUPlot for graphics instead of PGPLOT (default is PGPLOT)

High-speed FFT becomes possible by using the “F=FFTW” option when FFTW package has been already installed.

When re-make, execute

```
make clean
```

then execute

```
make T=COR [F=FFTW] [G=GNUPlot]
```

1.2 Installation step by step

How to install the package step by step is as follows where $HOME is the directory where an archive file is decompressed.

```
cd $HOME/ipvlbiYYYYMMDD/apri
make
make install
cd $HOME/ipvlbiYYYYMMDD/corr
make F=FFTW <= in case of using FFTW package
make install
cd $HOME/ipvlbiYYYYMMDD/sdelay
make
make install
cd $HOME/ipvlbiYYYYMMDD/src
make S=NONE install <= no use of a sampler *1
cd $HOME/ipvlbiYYYYMMDD/mark5
make
make install
cd $HOME/ipvlbiYYYYMMDD/vdif
make
make install
```

*1. Note that how to make at “src” directory differs from other directories (option “install” is used simultaneously with option ”S=NONE”)

When re-make, issue “make clean” at each subdirectory, then issue “make” at each subdirectory.
1.3 Environment variables

User-defined environment variables are accepted in the most of utility programs. The programs themselves define some environment variables. By typing as follows,

```bash
program_name env
```

the defined variables and current values are displayed.

example: typing

```bash
fx_cor env
```

The following messages are displayed.

**Environment variables**

- **K5COUT** --- default directory for correlation data out
  - (null), program deflt is (../cout/)
- **K5APRIDIR** --- default directory for apriori file
  - (null), program deflt is (../corrapri/)
- **PGDISP** --- default PGPLOT display device when selected so
  - (/XSERVE)
1.4 Directory structure after installation

$HOME/ipvlbiYYYYMMDD/

+readme_e.txt instruction (this file)
+readme.txt instruction in Japanese
+archive_cor shell script file to make an archive file
+pgplot_install.txt how to install PGLOT (in Japanese)
+fftw_install.txt how to install FFTW (in Japanese)
+makefile make file to install K5 utilities
+apri/ directory for a-priori calculation
  | +*.c source file
  | +*.h include file
  | +makefile makefile for this directory
+bin/ directory for executable files
+corr/ directory for software correlator
  | +*.c source file
  | +*.h include file
  | +makefile makefile for this directory
+corrapri/ directory for a-priori files
  | +ape_sample.txt sample a-priori file
  | +apelist_sample.txt sample a-priori file list
+cout/ directory for correlator outputs
  | +coutNNNN.txt output from XF-type correlator
  | +couttNNNN.txt output from FX-type correlator
+log/ directory for log output
+mark5/ directory for Mark5<->K5 format conversion
  | +*.c source file
  | +*.h include file
  | +makefile makefile for this directory
  | +m5tok5info.txt sample information file to convert Mrak5 to K5
  | +k5tom5info.txt sample information file to convert K5 to Mrak5
  | +vlbainfo.txt sample information file to convert VLBA to Mrak5
+sdelay/ directory for coarse search software
  | +*.c source file
  | +*.h include file
  | +makefile makefile for this directory
+sked/ directory for schedule files
  | +sample.skd sample schedule file (SKED format)
  | +sample.vex sample schedule file (VEX format)
+src/ directory for observation software
  | +*.c source file
  | +*.h include file
  | +makefile makefile for this directory
+vdif/ directory for VDIF<->K5 format conversion
  | +*.c source file
  | +*.h include file
  | +makefile makefile for this directory
+man/ directory for man page
+doc/ directory for documents
1.5 **Set up your environment**

1. In case that installed directory is, e.g., `$HOME/K5/ipvlbiXXXXXXXX` make symbolic link as follows.

   ```bash
   ln -s $HOME/K5/ipvlbiXXXXXXXX $HOME/ipvlbi
   ```

   if link has been already existed, issue command with “f” option as follows.

   ```bash
   ln -sf $HOME/K5/ipvlbiXXXXXXXX $HOME/ipvlbi
   ```

2. then add following description in your resource file (e.g., `.bashrc`).

   ```bash
   export PATH=$PATH:$HOME/ipvlbi/bin  
   setenv PATH $PATH:$HOME/ipvlbi/bin
   ```

   Now you can execute all K5 programs by merely re-linking a symbolic link when new version is installed.

3. Archives with version after 2010-02-08 includes man page files. Therefore you can use K5 software man page by adding following description in your resource file (e.g., `.bashrc`).

   ```bash
   export MANPATH=$HOME/ipvlbi/man:$MANPATH  
   setenv MANPATH $HOME/ipvlbi/man:$MANPATH
   ```

   e.g., you can issue “man cor” to know how to execute it. “man k5cor” tells about all K5 software correlation programs and utilities with brief instructions.
## 2 Software programs list

As for software programs listed here except for “sdelay”, you can get how to run by just executing a program itself without any option. You can get version information by executing a program with the “--version” option.

### 2.1 A-priori calculation

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>apri_calc</td>
<td>a-priori parameter calculation and generate an a-priori file (both stan-...</td>
</tr>
<tr>
<td>skdchk</td>
<td>check schedule file and display disk size required for K5/VSSP observa-...</td>
</tr>
</tbody>
</table>

Note: Last column shows the directory of source code.

### 2.2 Correlation processing

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fx_cor</td>
<td>FX-type software correlator for K5/VSSP format data</td>
</tr>
<tr>
<td>fx_cor_all</td>
<td>“fx_cor” for two or more scan data</td>
</tr>
<tr>
<td>fx_cor_new</td>
<td>FX-type software correlator supporting a variety of data formats</td>
</tr>
<tr>
<td>fx_cor_all_new</td>
<td>“fx_cor_new” for two or more scan data</td>
</tr>
<tr>
<td>cor</td>
<td>XF-type software correlator dedicated to 1 bit sampling data for K5/VSSP format data</td>
</tr>
<tr>
<td>cor_all</td>
<td>“cor” for two or more scan data</td>
</tr>
<tr>
<td>cor_new</td>
<td>XF-type software correlator dedicated to 1 bit sampling data supporting a variety of data formats</td>
</tr>
<tr>
<td>cor_all_new</td>
<td>“cor_new” for two or more scan data</td>
</tr>
<tr>
<td>sdelay</td>
<td>coarse fringe search. get residual dela and delay rate from correlated data</td>
</tr>
<tr>
<td>cor_mon</td>
<td>display correlation function dynamically (support PGPLOT only)</td>
</tr>
</tbody>
</table>

Note: Last column shows the directory of source code.

### 2.3 Data check

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>oscillo</td>
<td>display sampled data dynamically for K5/VSSP data (support PGPLOT only)</td>
</tr>
<tr>
<td>speana</td>
<td>display spectrum for K5/VSSP data</td>
</tr>
<tr>
<td>speana_n</td>
<td>display spectrum dynamically for K5/VSSP data (support PGPLOT only)</td>
</tr>
<tr>
<td>g_speana</td>
<td>display spectrum supporting a variety of data formats (K5/VSSP, VDIF, ...</td>
</tr>
<tr>
<td>datachk</td>
<td>check K5/VSSP format data</td>
</tr>
<tr>
<td>vdifcheck</td>
<td>check VDIF format data</td>
</tr>
<tr>
<td>m5check</td>
<td>check Mark-5 format data</td>
</tr>
</tbody>
</table>

Note: Last column shows the directory of source code.

### 2.4 Data Format Conversion

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>k5tom5b</td>
<td>convert K5 format to Mark5B format</td>
</tr>
<tr>
<td>k5tom5</td>
<td>convert K5 format to Mark5 format</td>
</tr>
<tr>
<td>k5tovdif</td>
<td>convert K5 format to Mark5B format</td>
</tr>
<tr>
<td>ads2k5</td>
<td>convert ADS3000 (DBBC mode) format to K5 format</td>
</tr>
<tr>
<td>m5btok5</td>
<td>convert Mark5B format to K5 format</td>
</tr>
<tr>
<td>m5tok5</td>
<td>convert Mark5 format to K5 format</td>
</tr>
</tbody>
</table>

Note: Last column shows the directory of source code.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Source Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>vdif2k5</td>
<td>convert VDIF format to K5 format</td>
<td>··vdif</td>
</tr>
<tr>
<td>vdif2m5b</td>
<td>convert VDIF format to Mark5B format</td>
<td>··vdif</td>
</tr>
</tbody>
</table>

Note: Last column shows the directory of source code.
3 A-priori calculation software

3.1 apri_calc

“apri_calc” calculates a priori values and generate an a-priori file necessary for cross correlation processing. Both SKED and VEX format files are supported as a schedule file.

3.1.1 How to execute

Type 1

apri_calc skdfile [options]

where  skdfile  
where option order: when parameter is omitted, it will be asked it later

-apedir apriori_file_out_directory
  
-baseid baseline_id
  
-coffset clock_offset
  
-crate clock_rate
  
-cepoch epoch
  
-g group
  
-ch channel
  
-start start_obs
  
-stop stop_obs
  
-xdir xdir
  
-ydir ydir
  
-ut1 ut1
  
-woblx woblx
  
-wobby wobby
  
-type1 | -type2
  
-type naming_type

-apedir apriori_file_out_directory
  
-baseid baseline_id
  
-coffset clock_offset
  
-crate clock_rate
  
-cepoch epoch
  
-g group
  
-ch channel
  
-start start_obs
  
-stop stop_obs
  
-xdir xdir
  
-ydir ydir
  
-ut1 ut1
  
-woblx woblx
  
-wobby wobby
  
-type1 : sidDDDNNNN.dat (default)
  -type2 : sidDDDDHHMMSSG.dat
  
(Note: this option is to keep compatibility with the program with the old version. Use “-type” option as follows)

- type naming_type
  
-type1 : sidDDDNNNN.dat (default for SKED)
  -1 : Type -1 sidDDDNNNN.#ch.dat
  -type2 : sidDDDDHHMMSSG.dat
  
(Note: this option is to keep compatibility with the program with the old version. Use “type” option as follows)

- type naming_type
  
-type1 : sidDDDNNNN.dat (default for SKED)
  -1 : Type -1 sidDDDNNNN.#ch.dat
  -2 : Type -2 sidDDDDHHMMSSG.#ch.dat
  -3 : Type 3 expid_sidG.scanid_YYYYDDDDHHMMSS.k5
  
(compliant with the naming rule of e-VLBI)
  
4 : unused
  
5 : Type 5 expid_scan#.stcode.k5a(-d) (default for VEX)
  
6 : Type 6 expid_sid_scanid.k5a(-d)
-typeX naming_type – set K5 observation file naming rule for X station
-typeY naming_type – set K5 observation file naming rule for Y station
-subnet | -nosubnet – subnet mode control. Default is subnet on (-subnet)
-skey satellite_key – set satellite key(s) for "NOZOMI" mode. If there are keys more
than one, use " " like -skey "NOZ HYB HEO"
-xoff x_clock_offset – set clock offset (sec) of X station to UTC. Positive value meanas X
clock tic earlier than UTC clock tic.
-lsb – set sideband LSB compulsory in case of SKED file
-shift time – shift start time of 1st scan to given time, where \( i{i;}\text{time}\) is
either YYYY/DDD-HH:MM:SS or YYYY/MM/DD-HH:MM:SS or YYYYDDDDHHMMSS.
-format VDIF|MK5|M5B|OCTAD|ADS|[VSSP] [sampling_cond]
   – set data format and sampling information of both station as the
   same format
-formX VDIF|MK5|M5B|OCTAD|ADS|[VSSP] [sampling_cond]
   – set data format and sampling information for X station (default
   is VSSP)
-formY VDIF|MK5|M5B|OCTAD|ADS|[VSSP] [sampling_cond]
   – set data format and sampling information for Y station (default
   is VSSP)
   where \( sampling\_cond \) is sampling information for MK5, M5B, OC-
TAD, ADS formats as follows.
   \( xM[Hz]nB[IT]mC[H] \) (any order)
   \( x \) – sampling frequency (MHz)
   \( n \) – # of AD bits
   \( m \) – # of channels

==== following options are valid for Ver.2016-10-12 and later ====
-source star_name – set extracting scans by a star name from a schedule file.
-m[onit] – set monitor mode

Type 2

apri_calc skdfile [apedir [baseid coffset raffset frqgr nob1 nob2 xdir ydir ut1c wobbx wobby
[naming_type [subnet [xoff [source]]]]]

where skdfile – schedule file name (either VEX or SKED format)
   if "-" is accompanied with skdfile like "-skdfile", it means monitor
   only the contents of schedule file. option "-monit" as for Ver.2016-
   10-12 and later, option "-monit" gives the same function.
apedir – a-priori file out directory. Default is given by environmental vari-
able K5APRIOUT. If this is not defined program default will be
applied.

if following parameters are omitted, they will be asked interactively
baseid – baseline ID (A2 or A4)
coffset – clock offset (sec)
raffset – clock rate offset (s/s)
frqgr – frequency group# corresponding to observation PC (1-4)
nob1 – start scan#
nob2 – stop scan# (nob1=0, nob2=0 for all scans)
xdir – data file directory for X station
ydir – data file directory for Y station
ut1c – UT1-UTC (sec)
wobbx – WobbX (arcsec)
wobby – WobbX (arcsec)
naming_type – K5 observation file naming rule
   1: Type 1 sidDDDDNNNN.dat (default for SKED)
-1: Type -1 sidDDDNNNN.#ch.dat  
2: Type 2 sidDDDHHHMSSG.dat  
-2: Type -2 sidDDDHHHMSSG.#ch.dat  
3: Type 3 expid_sidG_scanid_YYYYYDDDHHHMSS.k5 (compliant with e-VLBI)  
4: unused  
5: Type 5 expid_scan#.stcode.k5a(-d) (default for VEX)  
6: Type 6 expid_sid_scanid.k5a(-d)  

see Note 1 for details  

subnet  – subnet mode control 0: no-subnet, 1: subnet on (default)  
xcoff  – X clock offset to UTC (sec)  

==== following option is valid for Ver.2016-10-12 and later ====  
source  – set extracting scans by a star name from a schedule file

Monitoring environmental variables

check environmental variables used in “apri_calc”  
apri_calc env

K5SKED  -- default directory for a schedule file  
K5APRIDIR  -- default directory for an a-priori file  
K5NAMING  -- default out file naming rule

Note 1: naming rule of observation data file

Type 1 or −1 (I or −I)

XDDDNNNN.[#ch.].dat  
where  
X  – station ID (1 letter)  
DDD  – total day of the 1st scan  
NNNN  – scan# (4 digits)  
#ch  – # of channels (1|4) (in case of Type -1)

Type 2 or −2 (II or −II)

sidDDDHHHMSSG.[#ch.].dat  
where  
sid  – station ID (1 or 2 letters)  
DDDHHHMSS  – date and time of scan  
DDD  – year (3 digits)  
HH  – hour (2 digits)  
MM  – minute (2 digits)  
SS  – second (2 digits)  
G  – frequency group ID (a|b|c|d) or null  
#ch  – # of channels (1|4) (in case of Type -2)

Type 3 (compliant with the naming rule of e-VLBI)

expid_sidG_scanid_YYYYYDDDHHHMSS.k5

where  
expid  – experiment code  
sid  – station ID (2 lower-case letters)  
G  – terminal (PC) ID (1|2|3|4) or null  
set by “freqg” parameter
scanid – scan ID for VEX file

VEX ファイル使用時はスキャン ID そのまま
ddd-hhmm for SKED file. When multiple scans are included in the same minute, a,b,c,d,... are added for the 2nd scan and later.

YYYYDDDDHHMMSS – scan start time
.k5 – file descriptor for K5 format

Type 4 unused
Type 5 default for VEX file (to keep old version of apri_calc)
expid_scan#.stcode.k5a(-d)

where
expid – experiment code
scan# – scan#
stcode – station ID
k5 – descriptor for K5 format
a–d – frequency group (or null)

Note 2: Naming rule of created a-priori file

Type I
apeDDDNNNNXYG.txt

where
‘ape’ – fixed letters
DDD – total day of 1st scan
NNNN – scan# (4 digits)
XY – baseline ID (2 or 4 letters)
G – sampling host PC ID (a,b,c,d) or null corresponding to frequency group ID 1-4 or 0

Type II
apeDDDHMMSSXYG.txt

where
‘ape’ – fixed letters
DDDDHMMSS – date and time of scan
DDD – year (3 digits)
HH – hour (2 digits)
MM – minute (2 digits)
SS – second (2 digits)
XY – baseline ID (1 or 2 letters)
G – sampling host PC ID (a,b,c,d) or null corresponding to frequency group ID 1-4 or 0

Type III
ape.expid_sid1sid2.scanid_YYYYYDDDDHHMMSS.txt

where
‘ape’ – fixed letters
expid – experiment code
sid1 – station ID of X station (2 lower-case letters)
sid2 – station ID of Y station (2 lower-case letters)
scanid – scan ID for VEX file
Note 3: Clock parameters and epoch

Clock offset $c_{\text{offset}}$ is given as

$$c_{\text{offset}} = c_0 + \Delta \tau$$

where $c_0$ is clock offset used at “apri_calc” and $\Delta \tau$ is residual delay obtained by “sdelay”. Clock rate $c_{\text{rate}}$ is given as

$$c_{\text{rate}} = \frac{c_2 - c_1}{t_2 - t_1}$$

where $c_1$ is clock offset obtained for a scan at time $t_1$ and $c_2$ is clock offset obtained for a scan at time $t_2$, where $t_1$ and $t_2$ are PRT (processing reference time) given by a unit of second. If an experiment session extends on the next day, take the day difference into consideration. If clock rate is not 0, clock epoch should be set as $t_1$.

3.1.2 Examples (monitoring schedule)

Ex.1 Checking environmental variables used by “apri_calc”

```bash
$ apri_calc env
apri_calc (Ver. 2016-10-12)
Environment variables

K5SKED --- default directory for schedule file
   (null), program deflt is (./)
K5APRIDIR --- default directory for apriori file
   (null), program deflt is (.corrapri/)
K5NAMING --- default K5 file naming rule
   1|-1|2|-2|3|5
   (null), program deflt is (1)
K5SATKEY --- default satellite keyword of 1ch obs
   (null), program deflt is (HYB NOZ GEO)
```

Ex.2 Checking a schedule file

```bash
$ apri_calc -k10189.skd (or apri_calc k10189.skd -monit)
apri_calc (Ver. 2016-10-12)

------------------------ RUN CONDITION ------------------------
K5 file naming type is Type 1 : sidDDDNNNN.dat (SKED deflt)

SkdMonit: *********** Schedule file monitor ***********
SkdMonit: Schedule file = ./k10189.skd
SkdMonit: Expcode = K10189
SkdMonit: Total Scan # = 2203
SkdMonit: Total Star # = 12
SkdMonit: 1st Scan = 2010/07/07 20:05:00 3C84
SkdMonit: Last Scan = 2010/07/11 00:00:21 3C84
SkdMonit: Station ID Table
SkdMonit: G --- KOGANEI
SkdMonit: R --- KASHIM11
SkdMonit: Star Table
SkdMonit: NAME1 NAME2 RA(deg) DEC(deg) EPOCH
SkdMonit: 1 0212+735 $ 34.378389 73.825728 2000.000000
SkdMonit: 3 1921-293 $ 291.212733 -29.241700 2000.000000
SkdMonit: 4 2134+004 2134+00 324.160776 0.698393 2000.000000
SkdMonit: 5 2145+067 $ 327.022744 6.960723 2000.000000
SkdMonit: 7 1253-055 3C279 194.046527 -5.789312 2000.000000
SkdMonit: 8 1641+399 3C345 250.745042 39.810276 2000.000000
```

...
Ex.3 Checking a schedule file (set radio source name)

```bash
$ apri_calc -k10189.skd -source 3C273B
```

apri_calc (Ver. 2016-10-12)

------------- RUN CONDITION ----------------
K5 file naming type is Type 1 : sidDDDNNNN.dat (SKED deflt)
Pickup Source Name : 3C273B
------------- Schedule file monitor ********
Schedule file = ./k10189.skd
Expcode = K10189
Total Scan # = 2203
Total Star # = 12

---------- Station ID Table -----------
G --- KOGANEI
R --- KASHIM11
---------- Star Table ----------------
NAME1 NAME2 R.A.(deg) DEC(deg) EPOCH
1 0212+735 $ 34.378389 73.825728 2000.000000
2 0727-115 $ 112.579635 -11.686833 2000.000000
3 1921-293 $ 291.212733 -29.241700 2000.000000
4 2134+004 2134+00 324.160776 0.698393 2000.000000
5 2145+067 $ 327.022744 6.960723 2000.000000
6 1226+023 3C273B 187.277916 2.051288 2000.000000
7 1253-055 3C279 194.046527 -5.789312 2000.000000
8 1641+399 3C345 250.745042 39.810276 2000.000000
9 2223-052 3C446 336.446914 -4.950386 2000.000000
10 2251+158 3C454.3 343.490616 16.148211 2000.000000
11 0316+413 3C84 49.950667 41.511695 2000.000000
12 0923+392 4C39.25 141.762558 39.039126 2000.000000

---------- Frequency (MHz) Table ---------
Gr# 1 7700.99 U 7710.99 U 7720.99 U 7850.99 U
Gr# 2 8090.99 U 8290.99 U 8490.99 U 8550.99 U
Gr# 3 8570.99 U 8580.99 U 2210.99 U 2220.99 U
Gr# 4 2240.99 U 2290.99 U 2330.99 U 2340.99 U

---------- PCAL Freq (kHz) Table --------
Gr# 1 4010.0 4010.0 4010.0 4010.0
Gr# 2 4010.0 4010.0 4010.0 4010.0
Gr# 3 4010.0 4010.0 4010.0 4010.0
Gr# 4 4010.0 4010.0 4010.0 4010.0

---------- PICKUP SCAN TABLE -----------
SCAN# SOURCE YYYY/DDD HH:MM:SS DURA STATION_IDS
84 3C273B 2010/189 02:56:33 30 G R
87 3C273B 2010/189 03:01:02 30 G R
90 3C273B 2010/189 03:06:44 30 G R
1791 3C273B 2010/191 10:58:40 30 G R
1795 3C273B 2010/191 11:09:26 30 G R
1797 3C273B 2010/191 11:14:24 30 G R
1799 3C273B 2010/191 11:20:11 30 G R

...
This is a useful to pick up strong source scans.

Ex.4 Checking a schedule file (set baseline and source name)

```
$ apri_calc -eg094a.vex -baseid KsSh -source 3C454.3
   <= baseline is selected by option 'baseid' (2 letters for SKED file and
    4 letters for VEX). star name is set by option 'source'

apri_calc (Ver. 2016-10-12)
=====================================================================
K5 file naming type is Type 1 : sidDDNNNN.dat (SKED deflt)
Pickup Source Name : 3C454.3  <= radio source name
Pickup Baseline ID : KsSh    <= baseline ID
=====================================================================
search_site: No CLOCK info for ATCA included in the VEX FILE.
search_site: So all 0 for clock information was set.
search_site: No CLOCK info for PARKES included in the VEX FILE.
search_site: So all 0 for clock information was set.
....
search_site: No CLOCK info for IRBENE included in the VEX FILE.
search_site: So all 0 for clock information was set.
SkdMonit: *********** Schedule file monitor **********
SkdMonit: Schedule file = ./eg094a.vex
SkdMonit: Expcode = eg094a
SkdMonit: Total Scan # = 74
SkdMonit: Total Star # = 4
SkdMonit: 1st Scan = 2016/09/20 12:57:00 2223-052
SkdMonit: Last Scan = 2016/09/20 23:47:10 2215+020
SkdMonit: ---------- Station ID Table ----------
SkdMonit: At --- ATCA
SkdMonit: Pa --- PARKES
SkdMonit: Mp --- MOPRA
SkdMonit: Hb --- HOB_DBBC
SkdMonit: Cd --- CDDBBC
SkdMonit: Ti --- DSS43LBA
SkdMonit: Ks --- KASHIM34
SkdMonit: Ws --- WARK12M
SkdMonit: T6 --- TIANMA65
SkdMonit: Ur --- URUMQI
SkdMonit: Sh --- SHANGHAI
SkdMonit: Bd --- BADARY
....
SkdMonit: Mc --- MEDICINA
SkdMonit: O8 --- ONSALAB85
SkdMonit: Tr --- TORUN
SkdMonit: Hh --- HART
SkdMonit: Ir --- IRBENE
SkdMonit: (Note that 1 char station ID is not defined in a VEX file)
SkdMonit: --------------------------------------
SkdMonit: ------- Star Table -------------------
SkdMonit: NAME1 NAME2 R.A.(deg) DEC(deg) EPOCH
SkdMonit: 1 FAKERA 180.000000 85.000000 2000.000000
SkdMonit: 2 2215+020 334.450991 2.336309 2000.000000
SkdMonit: 3 2223-052 336.446914 -4.950386 2000.000000
SkdMonit: 4 3C454.3 343.490616 16.148211 2000.000000
SkdMonit: --------------------------------------
SkdMonit: ------- Frequency (MHz) Table --------
SkdMonit: Gr# 1 1668.00 L 1668.00 U 1668.00 L 1668.00 U
SkdMonit: --------------------------------------
SkdMonit: ------- PCAL Freq (kHz) Table ---------
SkdMonit: Gr# 1 2000.0 2000.0 2000.0 2000.0
SkdMonit: --------------------------------------
SkdMonit: --------- PICKUP SCAN TABLE ---------
SkdMonit: SCAN# SOURCE YYYY/DDD HH:MM:SS DURA STATION_IDS
SkdMonit: 14 3C454.3 2016/264 14:41:30 480 At Pa Mp Ho Cd Ti Ks Ws T6 Ur Sh Bd
SkdMonit: 29 3C454.3 2016/264 17:01:00 480 At Mp Ho Cd Ti Ks T6 Ur Sh Bd Sv Zc
```
3.1.3 Examples (in case of K5/VSSP)

Ex.1 non-interactive operation

Following example shows non-interactive operation in case that schedule file is “k10189.skd” in the “../cor-
rapri” directory, baseline ID=“RG”, scan#=-8, clock offset=8.0μsec, and the directory of a-priori output
is “./”.

```
$ apri_calc k10189.skd -baseid RG -apedir ./ -start 8 -stop 8 -coffset 8.0e-6
```

Ex.2 Interactive operation

```
$ apri_calc k10189.skd -apedir apeout <= execute with "-apedir" option to set
a-priori output directory to "apeout"
```
Output directory : apeout <= if this directory does'nt exits, program will stop with warning message.

*************** Schedule File Information **********************
File name --- ./k10189.skd
File type --- SKED
Exp. code --- K10189
# of stations --- 2
  G R
# of stars --- 12
# of scans --- 2203
1st Scan : 2010/07/07 20:05:00 3C84
Last Scan : 2010/07/11 00:00:21 3C84

************************************************************

------ STATION ID TABLE --------
G --- KOGANEI
R --- KASHIM11
--------------------------------
Enter Station ID for X station ----> R <= select station ID for X station
Enter Station ID for Y station ----> G <= select station ID for Y station
Selected Baseline is R-G
Sampling Information is as follows
from schedule file from operator
KASHIM11 : 32MHz 1bit 16CH 0MHz 0bit 0CH
KOGANEI : 32MHz 1bit 16CH 0MHz 0bit 0CH
Enter data directory for X station (KASHIM11) ---> ./ <= set X station data directory
Enter data directory for Y station (KOGANEI) ---> ./ <= set Y station data directory
Data directory for X station (KASHIM11) : ./
Data directory for Y station (KOGANEI) : ./
Frequency group and frequencies
Gr# 1 : 7700.99MHz U 7710.99MHz U 7720.99MHz U 7850.99MHz U
Gr# 2 : 8090.99MHz U 8290.99MHz U 8490.99MHz U 8550.99MHz U
Gr# 3 : 8570.99MHz U 8580.99MHz U 2210.99MHz U 2220.99MHz U
Gr# 4 : 2240.99MHz U 2290.99MHz U 2330.99MHz U 2340.99MHz U
Enter Frequency Group# ----> 2 <= select frequency group
Frequency Group# : 2
Gr# '2' is selected
PCAL freq (kHz) : 4010.0 4010.0 4010.0 4010.0
Enter Clock Offset (sec) -----> 0.0 <= set clock offset
(reflect clock offset after fringe search)
Enter Clock Rate (s/s) -----> 0.0 <= if non zero value is set, epoch will be asked as follows.

Enter Clock Epoch (YYYY/DDD-HH:MM:SS|YYYY/MM/DD-HH:MM:SS|YYYYDDDHHMMSS)
  0 for each PRT (old style)
  ---> 2010/07/07-20:05:00 <= enter epoch. If you enter 0, epoch is regarded as each PRT (processing reference time).
Clock offset and rate : 0 0 <= here clock rate is reflected
Clock Epoch : 0/000 00:00:00 <= here epoch is reflected
X Clock offset against UTC : 0.000000 <= this parameter can not be set in case of interactive operation
Enter UT1-UTC (sec) -----> 0.0 <= if precise value is not clear, 0 is OK.
Enter Wobb X (arcsec) -----> 0.0 <= if precise value is not clear, 0 is OK.
Enter Wobb Y (arcsec) -----> 0.0 <= if precise value is not clear, 0 is OK.

ut1,wobbx,wobby : 0.000000 0.000000 0.000000

# of scans --- 2203
Enter Start Scan number ----> 8 <= start scan#
Enter Stop Scan number ----> 8 <= stop scan#
Scan range : 8 - 8

8 2010189001135 4C39.25 0 0 0

Apriori File ( apeout/ape1880008RGb.txt ) created
Total # of a-priori files created is 1

$
$ apri_calc k10189.skd -source 3C273B  

apri_calc (Ver. 2016-10-12)

====================== RUN CONDITION ========================
K5 file naming type is Type 1 : sidDDDNNNN.dat (SKED deflt)
Sub-net mode ON : PRT is set according to each scan length
Pickup Source Name : 3C273B
Output directory : ../corrapri/

************************ Schedule File Information ****************
File name --- ./k10189.skd
File type --- SKED
Exp. code --- K10189
# of stations --- 2
  G  R
# of stars --- 12
# of scans --- 2203
1st Scan : 2010/07/07 20:05:00 3C84
Last Scan : 2010/07/11 00:00:21 3C84

----- STATION ID TABLE ------
G --- KOGANEI
R --- KASHIM11

Enter Station ID for X station ----> R
Enter Station ID for Y station ----> G
Selected Baseline is R-G

Sampling Information is as follows
from schedule file from operator
KASHIM11 : 32MHz 1bit 16CH 0MHz 0bit 0CH
KOGANEI : 32MHz 1bit 16CH 0MHz 0bit 0CH

Enter data directory for X station (KASHIM11) ---> ./
Enter data directory for Y station (KOGANEI) ---> ./

Data directory for X station (KASHIM11) : ./
Data directory for Y station (KOGANEI) : ./

Frequency group and frequencies
Gr# 1 : 7700.99MHz U 7710.99MHz U 7720.99MHz U 7850.99MHz U 8550.99MHz U 8650.99MHz U 8750.99MHz U 8850.99MHz U
Gr# 2 : 8090.99MHz U 8290.99MHz U 8490.99MHz U 8650.99MHz U 8750.99MHz U 8950.99MHz U 9150.99MHz U 9250.99MHz U
Gr# 3 : 9650.99MHz U 9850.99MHz U 2210.99MHz U 2220.99MHz U 2240.99MHz U 2260.99MHz U 2280.99MHz U 2300.99MHz U
Gr# 4 : 2240.99MHz U 2290.99MHz U 2330.99MHz U 2340.99MHz U 2350.99MHz U 2370.99MHz U 2390.99MHz U 2410.99MHz U

Enter Frequency Group# ----> 1
Frequency Group# : 1
Gr# 1 is selected
PCAL freq (kHz) : 4010.0 4010.0 4010.0 4010.0
Enter Clock Offset (sec) ----> 0
Enter Clock Rate (s/s) ----> 0
Clock offset and rate : 0 0
Clock Epoch : 0/000 00:00:00
X Clock offset against UTC : 0.000000
Enter UT1-UTC (sec) ----> 0
Enter Wobb X (arcsec) ----> 0
Enter Wobb Y (arcsec) ----> 0
ut1,wobbx,wobby : 0.000000 0.000000 0.000000

# of scans --- 2203
Enter Start Scan number ----> 0  
Enter Stop Scan number ----> 0  
Scan range : 1 - 2203

84 2010189025648 3C273B 0 0 0  
87 2010189030117 3C273B 0 0 0  
90 2010189030659 3C273B 0 0 0

Apriori File (../corrapri/ape1880084RGa.txt) created
Apriori File (../corrapri/ape1880087RGa.txt) created

............
Apriori File (.../corrapri/ape1881797RGa.txt ) created
1799 2010191112026 3C273B 0 0 0

Apriori File (.../corrapri/ape1881799RGa.txt ) created
Total # of a-priori files created is 168

$ Ex.4 Non-interactive operation of Ex.3

$ apri_calc k10189.skd ../corrapri RG 0 0 1 0 0 . / . / 0 0 0 1 0 0 0 3C273B
apri_calc (Ver. 2016-10-12)

====================== RUN CONDITION ========================
K5 file naming type is Type 1 : sidDDDNNNN.dat (SKED deflt)
Sub-net mode OFF: PRT is set based on the longest scan
Pickup Source Name : 3C273B
Output directory : ../corrapri
 ==============================================================

*************** Schedule File Information ********************
File name --- ./k10189.skd
File type --- SKED
Exp. code --- K10189
# of stations --- 2
  G R
# of stars --- 12
# of scans --- 2203
1st Scan : 2010/07/07 20:05:00 3C84
Last Scan : 2010/07/11 00:00:21 3C84

----------------------------------------------------------
------ STATION ID TABLE --------
G --- KOGANEI
R --- KASHIM11
--------------------------------
Station ID for X station : R
Station ID for Y station : G
Selected Baseline is R-G
Sampling Information is as follows
from schedule file from operator
KASHIM11 : 32MHz 1bit 16CH 0MHz Obit 0CH
KOGANEI : 32MHz 1bit 16CH 0MHz Obit 0CH

Data directory for X station (KASHIM11) : ./
Data directory for Y station (KOGANEI) : ./
Frequency group and frequencies
Gr# 1 : 7700.99MHz U 7710.99MHz U 7720.99MHz U 7850.99MHz U
Gr# 2 : 8090.99MHz U 8290.99MHz U 8490.99MHz U 8550.99MHz U
Gr# 3 : 8570.99MHz U 8580.99MHz U 2210.99MHz U 2220.99MHz U
Gr# 4 : 2240.99MHz U 2290.99MHz U 2330.99MHz U 2340.99MHz U
Frequency Group# : 1
Gr#`1 is selected
PCAL freq (kHz) : 4010.0 4010.0 4010.0 4010.0
Clock Epoch : 0/000 00:00:00
X Clock offset against UTC : 0.000000
ut1,wobbx,wobby : 0.000000 0.000000 0.000000
# of scans --- 2203
Scan range : 1 - 2203
84 2010189025648 3C273B 0 0 0

Apriori File (.../corrapri/ape1880084RGa.txt ) created
87 2010189030117 3C273B 0 0 0

---------------
Apriori File (.../corrapri/ape1881797RGa.txt ) created
1799 2010191112026 3C273B 0 0 0
Apriori File (.../corrapri/ape1881799RGa.txt ) created
Total # of a-priori files created is 168

20
A-priori files are created under the "apeout" directory. If a-priori output directory is not specified, a-priori files are created under "/corrapri/" directory.

3.1.4 Example (in case of VDIF Format Data)

$ apri_calc ..sked/ks15002.skd -format VDIF

apri_calc (Ver. 2016-10-12)

RUN CONDITION

K5 file naming type is Type 1 : sidDDDNNNN.dat (SKED deflt)
Sub-net mode ON : PRT is set according to each scan length
X station data format : VDIF
Y station data format : VDIF
Output directory : ../corrapri/

Schedule File Information

File name --- ../src/ks15002.skd
File type --- SKED
Exp. code --- KS15002
# of stations --- 5
# of stars --- 16
# of scans --- 289
1st Scan : 2015/01/02 02:00:00 3C345
Last Scan : 2015/01/03 02:02:43 0059+581

------ STATION ID TABLE --------

| G --- KOGANEI |
| R --- KASHIM11 |
| Y --- TATEYAMA |
| B --- MIURA |
| O --- KASHIM34 |

Enter Station ID for X station ----> R
Enter Station ID for Y station ----> G
Selected Baseline is R-G

Enter data directory for X station (KASHIM11) --->
Enter data directory for Y station (KOGANEI) --->

Frequency group and frequencies

Gr# 1 : 7864.99MHz U 7874.99MHz U 7884.99MHz U 8014.99MHz U
Gr# 2 : 8114.99MHz U 8244.99MHz U 8504.99MHz U 8544.99MHz U
Gr# 3 : 8564.99MHz U 8574.99MHz U 2214.99MHz U 2224.99MHz U
Gr# 4 : 2234.99MHz U 2264.99MHz U 2294.99MHz U 2304.99MHz U

Enter Frequency Group# (0 for all) ----> 0
All frequency mode is selected!

Enter Start Scan number ----> 1
Enter Stop Scan number ----> 2
Scan range : 1 - 2

1 2015002020045 3C345 0 0 0
Apriori File ( ../corrapri/ape0020001RG.txt ) created
 2 2015002020520 3C454.3 0 0 0
Apriori File ( ../corrapri/ape0020002RG.txt ) created
Total # of a-priori files created is 2

Edit an a-priori file by a text editor if necessary.

3.1.5 Example (in case of Mark-5B format Data)

$ apri_calc ..sked/ks15002.skd -format M5B < - - set data format "Mark-5B"
  
by "-format M5B" option

--- apri_calc (Ver. 2016-10-12) 

=================================================================
K5 file naming type is Type 1 : sidDDDDNNNN.dat (SKED deflt)
Sub-net mode ON : PRT is set according to each scan length
X station data format : Mark-5B <= specified data format for X station
Y station data format : Mark-5B <= specified data format for Y station
=================================================================

--------------------------- Schedule File Information ---------------------------
File name --- ../src/ks15002.skd
File type --- SKED
Exp. code --- KS15002
# of stations --- 5
# of stars --- 16
# of scans --- 289
1st Scan : 2015/01/02 02:00:00 3C345
Last Scan : 2015/01/03 02:02:43 0059+581

------ STATION ID TABLE ------

G --- KOGANEI
R --- KASHIM11
Y --- TATEYAMA
B --- MIURA
O --- KASHIM34
--------------------------------
Enter Station ID for X station ----> R
Enter Station ID for Y station ----> G
Selected Baseline is R-G

Sampling Information is as follows <= display sampling information
from schedule file from operator
KASHIM11 :  16MHz 1bit 16CH 0MHz 0bit 0CH
KOGANEI :  16MHz 1bit 16CH 0MHz 0bit 0CH

Enter data directory for X station (KASHIM11) ---
   ./ <= set data directory for X station
Enter data directory for Y station (KOGANEI) ---
   ./ <= set data directory for Y station

Data directory for X station (KASHIM11) : ./
Data directory for Y station (KOGANEI) : ./

Frequency group and frequencies
Gr# 1 : 7864.99MHz U 7874.99MHz U 7884.99MHz U 8014.99MHz U
Gr# 2 : 8244.99MHz U 8254.99MHz U 8504.99MHz U 8544.99MHz U
Gr# 3 : 8564.99MHz U 8574.99MHz U 2214.99MHz U 2224.99MHz U
Gr# 4 : 2234.99MHz U 2264.99MHz U 2294.99MHz U 2304.99MHz U

Enter Frequency Group# (0 for all) ---
   0 <= select all frequencies
All frequency mode is selected!

Enter Clock Offset (sec) ---
Enter Clock Rate (a/s) ---
Clock offset and rate : 0 0
Clock Epoch : 0/000 00:00:00
X Clock offset against UTC : 0.000000
Enter UT1-UTC (sec) ---
Enter Wobb X (arcsec) ---
Enter Wobb Y (arcsec) ---

ut1,wobbx,wobby : 0.000000 0.000000 0.000000

# of scans --- 289
Enter Start Scan number ----> 1
Enter Stop Scan number -----> 2
Scan range : 1 - 2
1 2015002020045 3C345 0 0 0
Apriori File ( ../corrapri/apriori0020001RG.txt ) created
2 2015002020520 3C454.3 0 0 0
Apriori File ( ../corrapri/apriori0020002RG.txt ) created
Total # of a-priori files created is 2

Edit an a-priori file by a text editor if necessary.

3.1.6 Example (in case of VSSP format for X station and VDIF format Y station)

$ apri_calc ..sked/ks15002.skd -formY VDIF
<= Y station data format is specified by "-formY" option to VDIF.
X station is default format (VSSP)

apri_calc (Ver. 2016-10-12)

============= RUN CONDITION ==============
K5 file naming type is Type 1 : sidDDDNNNN.dat (SKED deflt)
Sub-net mode ON : PRT is set according to each scan length
Y station data format : VDIF

****** Schedule File Information ******
File name ---- ../src/ks15002.skd
File type --- SKED
Exp. code --- KS15002
# of stations --- 5
G R Y B O
# of stars --- 16
# of scans --- 289
1st Scan : 2015/01/02 02:00:00 3C345
Last Scan : 2015/01/03 02:02:43 0059+581

------- STATION ID TABLE -------
G --- KOGANEI
R --- KASHIM11
Y --- TATEYAMA
B --- MIURA
O --- KASHIM34

Enter Station ID for X station ----> R
Enter Station ID for Y station ----> G
Selected Baseline is R-G

Sampling Information is as follows
from schedule file from operator
KASHIM11 : 16MHz 1bit 16CH 0MHz 0bit 0CH
KOGANEI : 16MHz 1bit 16CH 0MHz 0bit 0CH

Enter data directory for X station (KASHIM11)
--> ./

Enter data directory for Y station (KOGANEI)
--> ./

Frequency group and frequencies
Gr# 1 : 7864.99MHz U 7874.99MHz U 7884.99MHz U 8014.99MHz U
Gr# 2 : 8114.99MHz U 8244.99MHz U 8504.99MHz U 8544.99MHz U
Gr# 3 : 8564.99MHz U 8574.99MHz U 2214.99MHz U 2224.99MHz U
Gr# 4 : 2234.99MHz U 2264.99MHz U 2294.99MHz U 2304.99MHz U

Enter Frequency Group# ----> 1

Enter Clock Offset (sec) ----> 0
Enter Clock Rate (a/s) ----> 0
Clock offset and rate : 0 0
Clock Epoch : 0/000 00:00:00
X Clock offset against UTC : 0.000000
Enter UT1-UTC (sec) ----> 0
Enter Wobb X (arcsec) ----> 0
Enter Wobb Y (arcsec) ----> 0

ut1,wobbx,wobby : 0.000000 0.000000 0.000000

# of scans --- 289
Enter Start Scan number ----> 1
Enter Stop Scan number ----> 2
Scan range : 1 - 2

<table>
<thead>
<tr>
<th>Scan</th>
<th>Start</th>
<th>End</th>
<th>Station</th>
<th>RA (deg)</th>
<th>Dec (deg)</th>
<th>Tstart</th>
<th>Tstop</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2015002020045</td>
<td>3C345</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2015002020520</td>
<td>3C454.3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Apriori File ( ../corrapri/ape0020001RG.txt ) created
Apriori File ( ../corrapri/ape0020002RG.txt ) created
Total # of a-priori files created is 2

Edit an a-priori file by a text editor if necessary.

3.2 skdchk

"skdchk" reads a schedule file (SKED or VEX format) and display station, source, scan information and compute the size of necessary disk space.

3.2.1 How to execute

skdchk sked_file [-NOEARLY]
where sked_file – schedule file name
-NOEARLY – ignore “tape early start” parameter

3.2.2 Examples

```
$ skdchk jd0306.skd
skdchk Ver 2.31 2016-10-12
Schedule file is jd0306.skd

*** SCHEDULE FILE (jd0306.skd) INFORMATION ***
Experiment code : JD0306
Number of stations : 8
  1 T TSUKUB32 -3957408.751200 3310229.346600 3737494.836000
  2 A AIRA -3530219.322300 4118797.541900 3346015.905900
  3 C CHICHI10 -4490618.469200 3483908.166600 2884899.205700
  4 J SINTOTU3 -3642141.844800 2861496.642500 4370361.717900
  5 R KASHIM11 -3997505.701700 3276878.404550 3724240.703140
  6 Y GIFU11 -3787123.360830 3564181.693760 3680274.907440
  7 H TOMAKO11 -3680586.301730 2917515.745560 4309897.652680
  8 K YAMAGU32 -3502535.908490 3909590.219310 3566374.002980

Number of stars : 114
(only 20 stars are listed here)
  1 0003-066 $ 1.557887 -6.393149 2000.000000
  2 0014+813 $ 4.285312 81.585593 2000.000000
  4 0059+581 $ 15.690677 58.403994 2000.000000
  5 0104-408 $ 16.687950 -40.572211 2000.000000
  6 0106+013 $ 17.161546 1.583421 2000.000000
  7 0111+021 $ 18.429771 2.371477 2000.000000
  8 0119+115 $ 20.423313 11.831477 2000.000000
  9 0119+041 $ 20.486924 4.373537 2000.000000
 10 0133+476 $ 24.244145 47.858083 2000.000000
 11 0201+113 $ 30.944404 11.579280 2000.000000
 12 0202+149 $ 31.210058 15.236401 2000.000000
 13 0208-512 $ 32.692502 -51.017192 2000.000000
 14 0229+131 $ 37.941225 13.381866 2000.000000
 15 0235+164 $ 39.662209 16.616465 2000.000000
 16 0316+413 3C84 49.950667 41.511695 2000.000000
 17 0336-019 CTA26 54.878907 -1.776612 2000.000000
 18 0355+508 NRAO150 59.873947 50.963934 2000.000000
 19 0402-362 $ 60.973958 -36.803866 2000.000000
 20 0405-385 $ 61.745981 -38.441123 2000.000000
```

Number of scans : 209
First 5 scans are as follows:
1 CTA26 3/197 02:00:00 310
2 1803+784 3/197 02:06:10 784
3 4C39.25 3/197 02:19:50 100
4 0727-115 3/197 02:23:10 180
5 0537-441 3/197 02:26:40 190

Last 5 scans are as follows:
205 1044+719 3/198 01:22:20 430
206 0552+398 3/198 01:33:30 160
207 0202+149 3/198 01:37:00 784
208 0133+476 3/198 01:50:30 550
209 0537-441 3/198 02:00:40 180

Early Start Parameter (sec) : 0

Maximum Disk requirements
Total observation time (sec) = 76038
Disk requirements
32Mbps : 304.152 GBytes
64Mbps : 608.304 GBytes
128Mbps : 1216.608 GBytes
256Mbps : 2433.216 GBytes

Disk Requirements by Station (GBytes)

<table>
<thead>
<tr>
<th>sec</th>
<th>TSUKUB32</th>
<th>AIRA</th>
<th>CHICH10</th>
<th>SINTUTUS</th>
<th>KASHIM11</th>
<th>GIFU11</th>
<th>TOMAKO11</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sec</td>
<td>46018</td>
<td>65066</td>
<td>65976</td>
<td>37808</td>
<td>76038</td>
<td>76038</td>
</tr>
<tr>
<td></td>
<td>#scans</td>
<td>200</td>
<td>186</td>
<td>189</td>
<td>169</td>
<td>209</td>
<td>209</td>
</tr>
<tr>
<td>32Mbps</td>
<td>184.1</td>
<td>260.3</td>
<td>263.9</td>
<td>151.2</td>
<td>304.2</td>
<td>304.2</td>
<td>304.2</td>
</tr>
<tr>
<td>64Mbps</td>
<td>368.1</td>
<td>520.5</td>
<td>527.8</td>
<td>302.5</td>
<td>608.3</td>
<td>608.3</td>
<td>608.3</td>
</tr>
<tr>
<td>128Mbps</td>
<td>736.3</td>
<td>1041.1</td>
<td>1055.6</td>
<td>604.9</td>
<td>1216.6</td>
<td>1216.6</td>
<td>1216.6</td>
</tr>
<tr>
<td>256Mbps</td>
<td>1472.6</td>
<td>2082.1</td>
<td>2111.2</td>
<td>1209.9</td>
<td>2433.2</td>
<td>2433.2</td>
<td>2433.2</td>
</tr>
</tbody>
</table>

Disk Requirements by Station (GBytes)

<table>
<thead>
<tr>
<th>sec</th>
<th>YAMAGU32</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sec</td>
<td>29012</td>
</tr>
<tr>
<td></td>
<td>#scans</td>
<td>78</td>
</tr>
<tr>
<td>32Mbps</td>
<td>116.0</td>
<td></td>
</tr>
<tr>
<td>64Mbps</td>
<td>232.1</td>
<td></td>
</tr>
<tr>
<td>128Mbps</td>
<td>464.2</td>
<td></td>
</tr>
<tr>
<td>256Mbps</td>
<td>928.4</td>
<td></td>
</tr>
</tbody>
</table>
4 Correlation software

4.1 fx_cor, fx_cor_new, cor, cor_new

“fx_cor” and “fx_cor_new” are FX-type software correlator and carry out cross correlation processing using an apriori file. “cor” and “cor_new” are XF-type software correlator dedicated to 1-bit AD data. Correlated data are output to a file with a “cout” format. Auto-correlation or cross correlation with zero apriori value are also available.

“fx_cor” and “cor” are dedicated to K5/VSSP format data, but “fx_cor_new” and “cor_new” can process a variety of data formats, such as Mark5B, VDIF, ADS, OCTAD besides K5/VSSP.

In case of processing geodetic VLBI data with 1-bit sampling with 32 lags, “cor” or “cor_new” are faster than “fx_cor” or “fx_cor_new”.

4.1.1 How to execute

Type 1: normal correlation processing

“fx_cor” can be “fx_cor_new”, “cor”, or “cor_new”.

\[ \text{fx_cor afile [options]} \]

where

- **afile** – a-priori file name
- **options** (any order)
  - **-integ integration_time** – set total integration period (sec) default is scan length described in a schedule file in case of negative value, absolute value means integration exceeding scheduled scan length.
  - **-coffset clock_offset** – set clock offset (sec) applied for fringe search purpose. Positive value means Y clock tic earlier than X clock tic. Default is 0.0
  - **-crate clock_rate** – set clock rate (s/s) difference between X and Y station clock applied for fringe search purpose. Default is 0.0
  - **-soffset start_offset** – set start time offset (integer second)
  - **-t1pp t1pp** – PP (parameter period) (sec) for correlation processing. Default is BT1PP in "cor_head.h" (usually 1.0) see Note 1 for available t1pp less than 1 sec.
  - **-pp_nosync** – turns off the synchronization mode of PP (parameter period) (default is that PP synchronize to second tic)
  - **-lag delsize** – set the size of delay lag window by a power of 2, e.g., 16, 32, 64, 128, 256, 512, 1024, 2048, ..... Default is DELAYSIZE in “cor_head.h” (usually 32)
  - **-pmode pmode** – set plot device mode
    - 0: XWINDOW (env PGDISP can change PGPLOT display device) and PostScript file (pgplot.ps or gnuplot.ps) out (default)
    - 1: PostScript file (pgplot.ps or gnuplot.ps) out only
    - 2: XWINDOW only
    - 1: no graphic output
  - **-comment “any comment”** – set comment appeared in a correlation function plot
  - **-nopcal** – suppress PCAL (phase calibration) signal detection
  - **-ch1 ch1Y** – define Y channel for X CH# 1 (default 1)
  - **-ch2 ch2Y** – define Y channel for X CH# 2 (default 2)
  - **-ch3 ch3Y** – define Y channel for X CH# 3 (default 3)
  - **-ch4 ch4Y** – define Y channel for X CH# 4 (default 4)

Note: in case of “fx_cor_new” and “cor_new”, options upto -ch16 ch16Y are available.

- **orule naming_rule**
- set naming rule of output file name
  0 : use program default ("cout.txt" for “fx_cor" and “fx_cor_new”,
      "coutt.txt" for “cor" and “cor_new")
  1 : coutNNNN.txt for “fx_cor" and “fx_cor_new"
      couttNNNN.txt for “cor" and “cor_new" (default)
      where NNNN : serial number (4 digits)
  2 : coutEXP_CODE/coutYYDDDNNNNXYG.txt for “fx_cor"
      and “fx_cor_new”
      couttEXP_CODE/coutYYDDDNNNNXYG.txt for “cor" and “cor_new"
      where EXP_CODE – experiment code
      YY – year (2 digits)
      DDD – total day (3 digits)
      NNNN – scan# (4 digits)
      XY – baseline ID (2 or 4 letters)
      G – frequency group (a|b|c|d) or null
  3 : coutYYDDDNNNNXYG.txt for “fx_cor" and “fx_cor_new"
      couttYYDDDNNNNXYG.txt for “cor" and “cor_new"
      where YY – year (2 digits)
      DDD – total day (3 digits)
      NNNN – scan# (4 digits)
      XY – baseline ID (2 or 4 letters)
      G – frequency group (a|b|c|d) or null

-o:dir outdir – set output file directory.
  System default is environmental variable K5COUT. If K5COUT is
  not defined, program default defined by COUTDFLT in “cor_head.h"
  is used (usually “./cout")

-frstep frstep – set fringe phase calculation step in samples
  0 – automatic calculation
  1 – every 1 sample
  8 – every 8 samples (default)
  N – every N samples (1000/N should be integer)
  default is FRSTEP_FX_COR (in “cor_head.h”)

-frauto – set frstep=0 (automatic calculation mode)

-rfoffset roffset – set RF frequency difference between Y and X (RFy-RFx)(Hz)

-co:ut cout_file – set output file name compulsory. If this parameter is set, naming rule
  and out directory are ignored.

-call pcalf – set PCAL frequency of all channels compulsory (kHz)

-f1 pcalf1 – set PCAL frequency of CH# 1 compulsory (kHz)
-f2 pcalf2 – set PCAL frequency of CH# 2 compulsory (kHz)
-f3 pcalf3 – set PCAL frequency of CH# 3 compulsory (kHz)
-f4 pcalf4 – set PCAL frequency of CH# 4 compulsory (kHz)

Note: in case of “fx_cor_new" and "cor_new", options upto '-f16 pcalf16' are available

-1ch x_ch [y_ch] – set 1 channel processing mode
  x_ch – CH# of X station
  y_ch – CH# of Y station

===> belows are options for “fx_cor" and “fx_cor_new" ===>

-modefr modefr – set fringe stopping approximation mode
  0 – no approximation (exact calculation)
  9 – 9 level approximation (default)
  2 – 2 level approximation
  3 – 3 level approximation

-hanning – use Hanning lag window

-bpf flow:flow[:fact][,flow:flow[:fact][,flow:flow[:fact][,.....]]]
  – set BPF (bandpass filter) parameters by lower and upper cut-off fre-
    quencies (upto 20 sets)

27
- **flow** – lower cut-off frequency in a baseband (MHz)
- **fhigh** – upper cut-off frequency in a baseband (MHz)
- **fact** – amplitude coefficient (0.0 – 1.0) default is 1.0

```
-bpf2 fc:bw:[fact][,fc:bw:[fact][,fc:bw:[fact][,.....]]]
```

- set BPF (bandpass filter) parameters by center frequency and bandwidth (upto 20 sets)
  - **fc** – center frequency of BPF in a baseband (MHz)
  - **bw** – bandwidth (MHz). It can be omitted in case of the same value with the center frequency.
  - **fact** – amplitude coefficient (0.0 – 1.0) default is 1.0

- **fres**

  - set frequency resolution in case of BPF processing. Default is automatic set up.

Note 1: Valid values as **t1pp** are as follows.

<table>
<thead>
<tr>
<th>sampling frequency</th>
<th>PP period (sec) 0.01</th>
<th>0.02</th>
<th>0.04</th>
<th>0.05</th>
<th>0.1</th>
<th>0.2</th>
<th>0.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>40kHz</td>
<td>×</td>
<td>OK</td>
<td>OK</td>
<td>×</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>100kHz</td>
<td>×</td>
<td>×</td>
<td>OK</td>
<td>×</td>
<td>×</td>
<td>OK</td>
<td>×</td>
</tr>
<tr>
<td>200kHz</td>
<td>×</td>
<td>OK</td>
<td>OK</td>
<td>×</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>500kHz</td>
<td>×</td>
<td>×</td>
<td>OK</td>
<td>×</td>
<td>×</td>
<td>OK</td>
<td>×</td>
</tr>
<tr>
<td>1MHz</td>
<td>×</td>
<td>OK</td>
<td>OK</td>
<td>×</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>2MHz</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>4MHz</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>8MHz</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>16MHz</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>32MHz</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>64MHz</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
</tbody>
</table>

**Type 2: old style**

```
fx_cor aflie [seki: bun soffset coffset roffset t1pp smode pp_mode delsize] zoom pmode comment
```

where

- **afile** – a-priori file name
- **seki: bun** – total integration period (sec)
  - Default or 0 is scan length described in a schedule file
  - In case of negative value, absolute value means integration exceeding scheduled scan length.
- **soffset** – start time offset (integer second)
- **coffset** – clock offset (sec) applied for fringe search purpose. Positive value
  - means Y clock tic earlier than X clock tic. Default is 0.0
- **roffset** – clock rate (s/s) difference between X and Y station clock applied for fringe search purpose. Default is 0.0
- **t1pp** – PP (parameter period) (sec) for correlation processing. Default is BT1PP in "cor:head.h" (usually 1.0)
  - See Note 1 above for available **t1pp** less than 1 sec.
- **smode** – delay search range (no more use)
  - 0: Maximum (200000 samples)
  - 1: Middle (10000 samples)
  - 2: Minimum (1000 or 2000 samples)
  - Default is 2 but no more use!
- **pp_mode** – synchronization mode of PP
  - 0: PP synchronize to second tic (default)
  - 1: PP does not synchronize to second tic
- **delsize** – size of delay lag window (a power of 2)
  - e.g., 16, 32, 64, 128, 256, 512, 1024, 2048, ....
0 means DELAYSIZE in “cor_head.h” (usually 32)

$zoom$  
  - time axis zoom ratio in graphics (integer)
  0: full scale (default) as same as 1
  -VE: Max position is automatically centered. VE is a zoom factor.

$pmode$  
  - plot device selection
  0 : XTERM (environmental variable PGDISP can change PGPLOT display device) and PostScript file (pgplot.ps or gnuplot.ps) out (default)
  1 : PostScript file (pgplot.ps or gnuplot.ps) out only
  2 : XWINDOW only
  -1 : no graphic output

$comment$  
  - comment appeared in a correlation function plot

Type 3: auto-correlation mode

`fx_cor k5file1 [k5file2] [options]`

where

k5file1  
  - data file name

k5file2  
  - 2nd data file name. If directory is omitted in k5file2, directory is considered to be the same as that of k5file1. If 2nd data file name is omitted, k5file1 is used for both X and Y data file. Note: A-priori values are all set to zero. Frequencies are all set to zero.

options  
  - options given in Type 1 is available to use

This mode corresponds to use a-priori file as follows.

** This is Apriori file for auto correlation
**

$EXPCODE$
APE_ZERO

$STATION1$
STATION1 k5file1

$STATION2$
STATION2 k5file2

$FREQUENCY$
0.0 U
0.0 U
0.0 U
0.0 U

$PCAL_FREQ$
0.0
0.0
0.0

$CLOCK$
OFST= 0.0
RATE= 0.0

$SOURCE$
APE_ZERO

$START$
0000000000000

$STOP$
0000000000000

$APRIORI$
PR=0000000000000
TAU0= 0.0
TAU1= 0.0
TAU2= 0.0
TAU3= 0.0

$END
Monitoring environmental variables

Environmental variables used in software correlator are as follows.

- K5COUT – default directory for correlation data out
- K5APRIRDIR – default directory for apriori file
- RGDISP – PGPLOT display device name

They can be checked by

```
fx_cor env
```

4.1.2 Example: “cor”

Use default value except for a-priori file.

```
$ cor ./apeXY10.txt  <=>  ‘apeXY10.txt’ is an a-priori file located in the
current directory
```

************** run parameters (Ver 2014.8.13 or later)***************

```
afile = ./apeXY10.txt
Total Integ Period (sec) = 0.000000 (0.0 means integrated as scheduled)
T1PP (sec) = 1.000000 PP_sync_mode = 0 (PP sync to 1sec tic)
Lag Window Size = 32 Search_mode = 2
Start Offset (sec) = 0
Clock Offset (sec) = 0.000000e+00 Clock Rate(s/s) = 0.000000e+00
PCAL Detection : ON
Graphic Out Mode = 0 (PostScript Out + DISPLAY) tzoom = 1
comment = (null)
loop_param = 0 (for regular processing)
ch assign = (1 - 1) (2 - 2) (3 - 3) (4 - 4)
*************************************************************************
```

ApeMonit: APRIORI file is NEW VERSION
ApeMonit: EXPCODE JD0306
ApeMonit: OBS_NUMBER 8
ApeMonit: KASHIM11 /home/kondo/data/testspeed/Xk5data.10.dat
ApeMonit: XYZ -3997505.701700 3276878.404550 3724240.703140
ApeMonit: TOMAKO11 /home/kondo/data/testspeed/Yk5data.10.dat
ApeMonit: XYZ -3680586.301730 2917515.745560 4300987.652680
ApeMonit: BASEID RH
ApeMonit: PRT 2003 197 2 41 10
ApeMonit: START 2003 197 2 41 5
ApeMonit: STOP 2003 197 2 41 15
ApeMonit: Frequency Table
ApeMonit: Ch1 8209990000.000000 U
ApeMonit: Ch2 8219990000.000000 U
ApeMonit: Ch3 8249990000.000000 U
ApeMonit: Ch4 8309990000.000000 U
ApeMonit: PCAL Frequency Table
ApeMonit: Ch1 10000.000000
ApeMonit: Ch2 10000.000000
ApeMonit: Ch3 10000.000000
ApeMonit: Ch4 10000.000000
ApeMonit: Frequency Group# 1
ApeMonit: Aprioris
ApeMonit: Tau0 1.540623e-04
ApeMonit: Tau1dot 1.153642e-07
ApeMonit: Tau2dot -1.169617e-12
ApeMonit: Tau3dot -6.131427e-16
ApeMonit: Clock offset 2.485000e-06
ApeMonit: Clock rate 0.000000e+00
ApeMonit: X Clock offset 0.000000e+00
ApeMonit: UT1-UTC (sec) 0.000000
ApeMonit: X-WOBB (asec) 0.000000
ApeMonit: Y-WOBB (asec) 0.000000
ApeMonit: Star 3C273B
ApeMonit: RA 12 29 6.69973194
ApeMonit: DEC 2 3 8.59818480
ApeMonit: EPOCH 2000.0
ApeMonit: Directory ../cout already existed!
corr_engine: Version 2011-03-22
corr_engine: maxpp,idsize,irsiz,chsize,smode 2048 32 2048 4 2
corr_engine: X Data File is /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: Y Data File is /home/kondo/data/testspeed/Yk5data.10.dat
checkheader: Header (K5/VSSP) Sync Detected!!
checkheader: File : /home/kondo/data/testspeed/Xk5data.10.dat
checkheader: A/D(bits) 1 Chs 4 SFreq(kHz) 8000 Time 02:41:05 sec 9665
corr_engine: PP# 1 data saved
corr_engine: Time elapsed for 1PP processing is 1.027780 sec
corr_engine: X data time (BOPP) : 9665.000000
corr_engine: processed data (1.0/10.0)
checkheader: Header (K5/VSSP) Sync Detected!!
checkheader: File : /home/kondo/data/testspeed/Yk5data.10.dat
corr_engine: PP# 2 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
temporary corr out file (.agxDGypp.tmp) Opened!
corr_engine: File : /home/kondo/data/testspeed/Yk5data.10.dat
corr_engine: PP# 9 data saved
corr_engine: Time elapsed for 1PP processing is 1.027780 sec
corr_engine: X data time (BOPP) : 9673.000000
corr_engine: processed data (9.0/10.0)
checkheader: Header (K5/VSSP) Sync Detected!!
checkheader: File : /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: PP# 1 data saved
corr_engine: Time elapsed for 1PP processing is 1.026757 sec
corr_engine: X data time (BOPP) : 9673.000000
corr_engine: processed data (9.0/10.0)
checkheader: Header (K5/VSSP) Sync Detected!!
checkheader: File : /home/kondo/data/testspeed/Yk5data.10.dat
corr_engine: PP# 1 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
temporary corr out file (.agxDGypp.tmp) Opened!
corr_engine: File : /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: PP# 9 data saved
corr_engine: Time elapsed for 1PP processing is 1.026757 sec
corr_engine: X data time (BOPP) : 9673.000000
corr_engine: processed data (9.0/10.0)
checkheader: Header (K5/VSSP) Sync Detected!!
checkheader: File : /home/kondo/data/testspeed/Yk5data.10.dat
corr_engine: PP# 1 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
temporary corr out file (.agxDGypp.tmp) Opened!
corr_engine: File : /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: PP# 9 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
temporary corr out file (.agxDGypp.tmp) Opened!
corr_engine: File : /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: PP# 9 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
temporary corr out file (.agxDGypp.tmp) Opened!
corr_engine: File : /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: PP# 9 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
temporary corr out file (.agxDGypp.tmp) Opened!
corr_engine: File : /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: PP# 9 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
temporary corr out file (.agxDGypp.tmp) Opened!
corr_engine: File : /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: PP# 9 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
temporary corr out file (.agxDGypp.tmp) Opened!
corr_engine: File : /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: PP# 9 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
temporary corr out file (.agxDGypp.tmp) Opened!
corr_engine: File : /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: PP# 9 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
temporary corr out file (.agxDGypp.tmp) Opened!
corr_engine: File : /home/kondo/data/testspeed/Xk5data.10.dat
corr_engine: PP# 9 data saved
chkdif_flag = 0
checkheader: File EOF! (/home/kondo/data/testspeed/Yk5data.10.dat)
corr_engine: Atamadashi finished!
temp_info_out_file (.PNRo93hd.tmp) Opened!
Fig. 4.1 is displayed after processing.

4.1.3 Example: “fx_cor_new” (Mark5B data)

Default values are used except for a-priori file.

```
$ fx_cor_new ./ape220205919RG.m5b.txt <= use a-priori file prepared for Mark5B
```

************** run parameters (Ver 2014.8.13 or later)**********************

```
afile = ./ape220205919RG.m5b.txt
Total Integ Period (sec) = 0.000000 (0.0 means integrated as scheduled)
T1PP (sec) = 1.000000 PP_sync mode = 0 (PP sync to 1sec tic)
Lag Window Size = 32  Search_mode = 2
Start Offset (sec) = 0
Clock Offset (sec) = 0.000000e+00  Clock Rate(s/s) = 0.000000e+00
PCAL Detection = ON
Graphic Out Mode = 0 (PostScript Out + DISPLAY)  tzoom = 1
comment = (null)
loop_param = 0 (for regular processing)
ch assign = (1 - 1) (2 - 2) (3 - 3) (4 - 4)
```

BPF parameters

```
modebpf = 0
```

```
Frequency resolution (MHz) (0 mean AUTO) : 0.000000
```

ApeMonit: APRIORI file is NEW VERSION

```
ApeMonit: EXPCODE K10216
```

...........................

```
engine_datafile_open: # of channels to be processed is 16
xros_engine: << runmode (runmode) = 1
xros_engine: << smode (smode) = 2
xros_engine: << # of samples in a unit (usampl) = 1000
```

Figure 4.1. Examples of correlation function plot after "fx_cor" and "cor" processing.
**xros_engine:**

- **<< lag size (idsize) =** 32
- **<< FFT size for processing =** 32
- **<< Lag window type =** Box
- **<< PP period in sec (tipp) =** 1.0
- **<< # of usampl in a PP (nspp) =** 32000
- **<< # of bytes in a usampl (numb) =** 2000
- **<< # of bytes in a usampl per CH =** 125
- **<< RF offset (Hz) =** 0.000000
- **<< Fringe stopping mode =** 0 (base band)
- **<< Fringe phase calc mode (modefr) =** 9 (9 level approx)
- **<< Fringe phase calc step (frstep) =** 8 sample(s)
- **<< Pcal detection mode X station =** 1
- **<< Pcal detection mode Y station =** 1
- **<< # of channels to be processed =** 16
- **<< X data file format =** Mark-5B 32MHz 1BIT 16CH
- **<< Y data file format =** Mark-5B 32MHz 1BIT 16CH
- **<< Channel Assignment =** (1 - 1) (2 - 2) (3 - 3) (4 - 4) (5 - 5) (6 - 6) (7 - 7) (8 - 8) (9 - 9) (10 - 10) (11 - 11) (12 - 12) (13 - 13) (14 - 14) (15 - 15) (16 - 16)

**Postscript out file ==> pgplot.ps**

```
CH#  FREQ(MHz)  MAX AMP  RESIDUAL DELAY (sec)
--------------------------------------------------------------
  1  7700.99   0.000834  -4.24985e-11
  2  7710.99   0.000935   1.00162e-09
  3  7720.99   0.000748   3.14188e-09
  4  7850.99   0.000892   1.04695e-09
  5  8090.99   0.000854   4.31266e-09
  6  8290.99   0.000832   1.02798e-08
  7  8490.99   0.000883   7.45903e-09
  8  8550.99   0.000833   1.06133e-08
  9  8570.99   0.000830   1.49527e-08
 10  8580.99   0.000757   1.01071e-08
 11  8700.99   0.001171   1.84406e-08
 12  8730.99   0.001119   1.64464e-08
 13  8720.99   0.001125   1.6231e-08
 14  8790.99   0.001223   2.18096e-08
 15  8830.99   0.001218   1.84975e-08
 16  8840.99   0.001376   1.57303e-08
--------------------------------------------------------------
```

**COUT File is ./cout/cout0005.txt**

Time elapsed (sec) for one obs process is 333.220641

Fig.4.2 is displayed after processing.

33
Figure 4.2. Plot of correlation function after “fx_cor_new” processing.
4.2 fx_cor_all, fx_cor_all_new, cor_all, cor_all_new

“fx_cor_all” is “fx_cor” for two or more scan data using a list file that contains a number of a-priori file names. “fx_cor_all_new” is “fx_cor_new” for two or more scan data. So do “cor_all” and “cor_all_new”.

An example of how to create a list file is as follows.

An example:
At the directory where a-priori files are located, issue

```
ls -1 ape*.txt > apelist294OU.txt
```

4.2.1 How to execute

“fx_cor_all” can be “fx_cor_all_new” or “cor_all_new” in bellow.

```
fx_cor_all pfile [options]
```

where

- **pfile** – list file name
- **options** (any order)
  - `integ integration_time`
    - total integration period (sec)
    - default is scan length described in a schedule file
    - in case of negative value, absolute value means integration exceeding scheduled scan length.
  - `coffset clock_offset`
    - set clock offset (sec)
    - Positive value means Y clock tic earlier than X clock tic. Default is 0.0
  - `crate clock_rate`
    - set clock rate (s/s) difference between X and Y station clock.
    - Default is 0.0.
  - `soffset start_offset`
    - set start time offset (integer second)
    - Default is 0.
  - `t1pp t1pp`
    - PP (parameter period) (sec) for correlation processing. Default is BT1PP in "cor_head.h" (usually 1.0)
  - `pp_nosync`
    - turns off the synchronization mode of PP (parameter period) (default is that PP synchronize to second tic)
  - `lag delsize`
    - set the size of delay lag window by a power of 2, e.g., 16, 32, 64, 128, 256, 512, 1024, 2048, ....
    - Default is DELAYSIZE in “cor_head.h” (usually 32)
  - `pmode pmode`
    - set plot device mode
      - 0: XWINDOW (env PGDISPLAY can change PGPLOT display device)
      - and PostScript file (pgplot.ps or gnuplot.ps) out (default)
      - 1: PostScript file (pgplot.ps or gnuplot.ps) out only
      - 2: XWINDOW only
      - -1: no graphic output
  - `comment “any comment”`
    - set comment appeared in a correlation function plot
  - `nopcal`
    - suppress PCAL (phase calibration) signal detection
  - `ch1 ch1Y`
    - define Y channel for X CH# 1 (default 1)
  - `ch2 ch2Y`
    - define Y channel for X CH# 2 (default 2)
  - `ch3 ch3Y`
    - define Y channel for X CH# 3 (default 3)
  - `ch4 ch4Y`
    - define Y channel for X CH# 4 (default 4)

Note: in case of “fx_cor_new” and “cor_new”, options upto `ch16 ch16Y` are available

- **orule naming_rule**
  - set naming rule of output file name
    - 0 : use program default ("cout.txt" for “fx_cor” and “fx_cor_new”,
      "coutt.txt" for “cor” and “cor_new”)
    - 1 : coutNNNN.txt for “fx_cor” and “fx_cor_new”
      couttNNNN.txt for “cor” and “cor_new” (default)
where NNNN : serial number (4 digits)
2 : cout_EXP_CODE/coutYYDDDNNNNXYG.txt for “fx_cor”
    and “fx_cor_new”
coutt_EXP_CODE/coutYYDDDNNNNXYG.txt for “cor”
    and “cor_new”
where EXP_CODE – experiment code
    YY – year (2 digits)
    DDD – total day (3 digits)
    NNNN – scan# (4 digits)
    XY – baseline ID (2 or 4 letters)
    G – frequency group (a|b|c|d) or null
3 : coutYYDDDNNNNXYG.txt for “fx_cor” and “fx_cor_new”
couttYYDDDNNNNXYG.txt for “cor” and “cor_new”
where YY – year (2 digits)
    DDD – total day (3 digits)
    NNNN – scan# (4 digits)
    XY – baseline ID (2 or 4 letters)
    G – frequency group (a|b|c|d) or null
-odir outdir – set output file directory.
    System default is environmental variable K5COUT. If K5COUT
    is not defined, program default defined by COUTDFLT in
    “cor_head.h” is used (usually “./cout”)
-frstep frstep – set fringe phase calculation step in samples
    0 – automatic calculation
    1 – every 1 sample
    8 – every 8 samples (default)
    N – every N samples (1000/N should be integer)
    default is FRSTEP_FX_COR (in “cor_head.h”)
-frauto – set frstep=0 (automatic calculation mode)
-rfoffset rf_offset – set RF frequency difference between Y and X (RFy-RFx)(Hz)
=== belows are options for “fx_cor_all” and “fx_cor_new_all” ====
-modefr modefr – set fringe stopping approximation mode
    0 – no approximation (exact calculation)
    9 – 9 level approximation (default)
    2 – 2 level approximation
    3 – 3 level approximation
-hanning – use Hanning lag window
-hamming – use Hamming lag window
4.3 sdelay

“sdelay” carries out coarse search of residual delay and delay rate from correlation data.

4.3.1 How to execute

Type 1: help

sdelay HELP]

Type 2: interactive operation

sdelay [PP] [options]

where PP – to make each PP output. See Type 4 for options.

Type 3: non-interactive operation

sdelay coutfile [PP] [options]

where coutfile – K5 software correlator output file. See Type 4 for options.

Type 4: general operation

sdelay [options]

where options (any order)

-v – verbose
-cout coutfile – set correlator output file (cout file) name
-sdir coutdir – set search start cout file directory to read
-cdir coutdir – set cout file directory to process all
    The difference between -sdir and cdir is as follows,
    -sdir : set search start directory
    -cdir : set cout directory to process all files
-ppout – make output file of each PP data by every channel besides normal
    sdelay out file.
-nosingle – force output to multiple files (default: all results are output in a
    single file)
-odir outdir – sdelay output directory
-pploot device – set PGPLOT device (/NULL for suppression)
-ps – set graphics out to PostScript (pgplot.ps or gnuplot.ps)
-2nd [t2dot] – search up to 2nd order rate and set search range from −t2dot to
    +t2dot
    (default for t2dot is 1−13s/s²)
-fringe – plot fringe phase and amplitude
-pcal – plot PCAL phase and amplitude
-vspeplot – plot video cross spectrum and output spectrum data to “vspe-
    out.txt” file
-no3d – suppress plot of coarse search function
-noplot – suppress all graphics plots
-integ tinteg – force integration time to tinteg sec. (default is all available period)
-vanveleck – make Van Vleck correction using \( r = \sin(r_c \pi/2) \) where \( r \) is true
    correlation coefficient, and \( r_c \) is coefficient for 1-bit sampling data.
    This option will be neglected in case of multi-bit AD data.
-out ofile – output 2D array data to a text file besides normal sdelay out
-classic – classic axis style (no tic marks) for 3D plots
-tzoom tzoom – delay axis zoom-up factor (tzoom ≥ 1.0)
-tshift tshift – delay axis shift value when tzoom is set (sec) (set new center value
    for delay axis).
-bpf flow:fhigh[:fact][,flow:fhigh[:fact][,flow:fhigh[:fact][,......]]]
  - set BPF by lower and higher cut-off frequencies (upto 20 sets)
  flow        - lower cut-off frequency in a baseband (MHz)
  fhigh       - higher cut-off frequency in a baseband (MHz)
  fact        - amplitude factor (0.0–1.0), default is 1.0

-bpf2 fc:bw[:fact][,fc:bw[:fact][,fc:bw[:fact][,......]]]
  - set BPF parameters by center frequency and bandwidth (upto 20 sets)
  fc          - center frequency in a baseband (MHz)
  bw          - pass bandwidth (MHz). It can be omitted for 2nd and later BPF if it is the same bandwidth with the 1st one.
  fact        - amplitude factor (0.0–1.0), default is 1.0

-fres fres   - set frequency resolution (MHz) (default is automatic)

-line[mode]   - set line spectrum mode. 1st FFT size is set as same as lag size

-hanning     - apply Hanning filter on correlation function

-lag[size] lag - set new lag size to reduce lag size (should be a power of 2). Note: This option cannot be used with ‘-hanning’ and/or ‘-vanvleck’ options

-sub[panels] 1|4|9|16 - set # of subpanels in a main panel compulsory (default is automatic) (Note: only PGPLOT support this parameter)

-nodel[ay correction] - no delay correction in integrated video spectrum

-obs[out]    - display observed results and total fringe phase (default is no display)

4.3.2 Example

$ sdelay

SDELAY Ver. 2016-08-12
# of cout files found under ..\cout\ ----------- 107
# of cout directories found under ..\cout\ ---- 51
1 --- Go to File selection
2 --- Go to further directory selection
Enter your selection --> 2          <= select 2
Software correlator out directories are as follows
1 cout
  2 cout021550R
  3 cout021550Rold
  4 cout021550U
  5 cout021550U2
  6 cout021550Uold
        ......
  33 coutCs7200
  34 coutD0301
  35 coutGIFU
  36 coutGSI
  37 coutJD0306
  38 coutJD0609
        ......
  48 cout_tid062020G
  49 cout_tid062020Y_8sec
  50 cout_tid062020G
  51 cout_tid062020Y
Select directory --> 36         <= select 36
Selected directory is ..\cout\coutGSI\
Software correlator out files are as follows
1 cout040970001ACa.txt
  2 cout040970001ACb.txt
  3 cout040970001ACc.txt
  4 cout040970001ACd.txt
  5 cout040970001TAc.txt
  6 cout040970001Tab.txt
        ......
  28 cout042420001TVd.txt
  29 cout042420002TVa.txt
  30 cout042420002TVb.txt
  31 cout042420002TVc.txt
  32 cout042420002TVd.txt
Select File (0 means all) --> 5        <= select 5

38
coutt040970001TAa.txt is selected!
sdelay: output file is .\sdelayout.txt
sdelay: correlation data file is ..\cout\coutGSI\coutt040970001TAa.txt
fx_cor_out_hd_read: Data File format 7

******************** SDELAY (Ver. 2016-08-12) SUMMARY OUT PUT ************
COUT : ..\cout\coutGSI\coutt040970001TAa.txt
X DATA : /vncpc1/ad2/JD0404/T097020000a.dat
Y DATA : /vncpc1/ad4/JD0404/A097020000a.dat
BASELINE : TSUKUB32 - AIRA
SOURCE : 3C454.3 SAMPLING : 1 bit 8 MHz
PRT : 2004/097 02:00:50 Tinteg(s) : 99.0
LAG SIZE : 32
CLOCK : offset -1.085e-006(s) rate -4.542e-014(s/s)
EOP : ut1-utc -0.439965(s)
: x-wobb -0.140730(asec)
: y-wobb 0.333260(asec)

Note: No amplitude correction is made.

CH# FREQUENCY AMP MAX POSITION RESIDUAL
(MHz) ( 64x 128) Delay(usec) Rate(ps/s) SNR
---------------------------------------------------------------
1 8209.99 U 2.757e-003 ( 33, 65) 0.000 0.016 77.6
2 8219.99 U 2.822e-003 ( 33, 65) -0.012 0.019 79.4
3 8249.99 U 2.732e-003 ( 33, 65) -0.013 -0.038 76.9
4 8309.99 U 2.627e-003 ( 33, 65) 0.014 0.028 73.9

Note: No amplitude correction is made.

CH# PCAL FREQ(kHz) X-Amp X-Phase Y-Amp Y-Phase
---------------------------------------------------------------
1 10.00 0.027 132.8 0.062 45.5
2 10.00 0.028 -89.4 0.061 -4.1
3 10.00 0.028 -45.4 0.061 -115.5
4 10.00 0.028 161.1 0.060 155.1

A coarse search function shown in Fig.4.3 will be displayed. Figs.4.4, 4.5, and 4.6 show PGPLOT graphics plots in case of options “-fringe” (phase and amplitude by PP), “-pcal” (PCAL phase and amplitude), and “-vspeplot” (video spectrum), respectively.
Figure 4.3. An example of graphics plots (coarse search function).

Figure 4.4. An example of sdelay graphics in case of “fringe” option (fringe phase and amplitude by PP).
Figure 4.5. An example of sdelay graphics in case of “-pcal” option (PCAL phase and amplitude by PP).

Figure 4.6. An example of sdelay graphics in case of “-vspeplot” option (video spectrum).

An example of text file (vspeout.txt9 output in case of “-vspeplot” option.

$FORMAT Ver. 2016-12-19
$OBS
Figs. 4.7, 4.8, 4.9, and 4.10 examples of GNUPLOT graphics.
Figure 4.8. An example of sdelay graphics in case of “-fringe” option (fringe phase and amplitude by PP)(GNUPlot)

Figure 4.9. An example of sdelay graphics in case of “-pcal” option (PCAL phase and amplitude by PP)(GNUPlot).
Figure 4.10. An example of sdelay graphics in case of “-espeplot” option (video spectrum)(GNULOT).
4.3.3 Output file format

Default output file

file name — “sdelayout.txt” or the name ‘cout’ in correlator output file name substituted by ‘sdel’.
Example: correlator output file name = coutt0205.txt ===> generated sdelayout output file = sdel0205.txt

Contents of file are as follows.

$OBS
$BASELINE KASHIM11 TOMAKO11
$PRT 2003 197 2 41 10
$SOURCE 3C273B
$SAMPLING(Hz) 8e+06
$VIDEO_Bw(Hz) 4e+06
$A/D(bits) 1
$NUMBER of PP 9
$SEKIBUN(s) 9.000000
$APRIORI (TAU(s) TAU1dot(s/s) TAU2dot TAU3dot)
0.00015406231 1.1536424e-07 1.1696165e-12 -6.1314267e-16
$CLOCK (offset(s),rate(s/s))
2.485e-06 0
$RESULTS Freq(MHz) AMP Res_Delay(s) Err Res_Rate(s/s) Err Res_2dot(s/s^2) Res_Phase(deg)
8209.99 0.00189916 1.96904e-09 4.40734e-13 4.7298e-13 0 66.52
8219.99 0.00189337 -1.03363e-08 3.30101e-12 6.3016e-13 0 96.38
8249.99 0.00188501 -9.87243e-09 8.73712e-09 4.40734e-13 4.7298e-13 0 114.36
8309.99 0.00188501 -9.87243e-09 8.6713e-09 5.59579e-13 4.6088e-13 0 114.79
$OBSERVED Freq(MHz) Observed_Delay(s) Observed_Rate(s/s) Total_Phase(deg)
8209.99 0.000154064279043935 1.15364680734089e-07 134.03
8219.99 0.000154051973685111 1.15367541008296e-07 342.32
8249.99 0.000154078532405484 1.15364794919007e-07 240.92
8309.99 0.00015405243756551 1.15364839579305e-07 322.84

Output file by PP

file name — “cout” in correlator output file name is substituted by ‘sdel’ + ‘.CH1’–’.CH4’.
Example: correlator output = coutt0205.txt ===> generated sdel output = sdel0205.txt.CH1, ..., sdel0205.txt.CH4

Contents of file are as follows.

$FORMAT Ver. 2017-02-24
$OBS
$BASELINE KASHIM11 TOMAKO11
$PRT 2003 197 2 41 10
$PRT (sec in day) 9670.000000
$SOURCE 3C273B
$SAMPLING(Hz) 8e+06
$VIDEO_Bw(Hz) 4e+06
$A/D(bits) 1
$NUMBER of PP 9
$SEKIBUN(s) 9.000000
$APRIORI (TAU(s),TAU1dot(s/s),TAU2dot,TAU3dot)
0.00015406231 1.1536424e-07 -1.1696165e-12 -6.1314267e-16
$CLOCK (offset(s),rate(s/s))
2.485e-06 0
$RESULTS freq(MHz), amp, residual_delay(s), err, residual_rate(s/s), err, residual_rate(s/s^2)
8209.99 0.00185916 1.96904e-09 8.73712e-09 4.40734e-13 4.7298e-13 0
8219.99 0.001486231 1.1536424e-07 -1.1696165e-12 -6.1314267e-16
$AVERAGED X-PCAL AMP and PHASE(deg), Y-PCAL AMP and PHASE(deg)
0.0688 70.362 0.1595 111.713
$Total PP number and PP period in sec 9 1.000000
$REFERENCE FREQUENCY(MHz) 8209.990000
$ EACH PP DATA
$ Date MOPP_Time tau(sec) amp phs(deg) xpamp xpphas(deg)
ypamp ypphas(deg)
2003/07/16 02:41:05.50 1.5354512615e-04 0.0013034785 85.446 0.0682 71.240
0.1594 111.211
2003/07/16 02:41:06.50 1.5362094550e-04 0.0014538306 88.053 0.0692 73.339
0.1600 111.025
2003/07/16 02:41:07.50 1.5377556369e-04 0.0020580706 50.928 0.0678 71.786
0.1590 114.300
--- continue upto # of PPs

45
## Output of 2D array data

**file name** — name specified by ‘-out ofile’ option

Contents of file are as follows.

```plaintext
TSUKUB32 - WETTZELL
CH#:1  2344.99MHz U  1bit 16MHz sampling
Source : 4C39.25, T integ(sec)=54.0
Amp = 0.004531, SNR = 133.2 (no amp correction)
Delay Res (sec) : -2.631e-008  Rate Res(s/s) :  8.603e-013

**** 2D DATA START ****
128 64  <= array size  line (rate direction) X column (delay direction)
6.12758e-005 4.88404e-005 6.13282e-005 ... <= line#1 64 point data
5.81745e-005 6.01026e-005 5.68238e-005 ... <= line#2 64 point data

....
3.73513e-005 9.13274e-006 4.24058e-005 ... <= line#128 64 point data

**** 2D DATA END ****

TSUKUB32 - WETTZELL
CH#:2  2352.99MHz U  1bit 16MHz sampling
Source : 4C39.25, T integ(sec)=54.0
Amp = 0.004311, SNR = 126.7 (no amp correction)
Delay Res (sec) : -2.184e-008  Rate Res(s/s) :  1.112e-012

**** 2D DATA START ****
128 64

.... <= start of CH#2 data

....
```

4.4 `cor_mon`

“`cor_mon`” displays correlation function dynamically

(Note: only PGPLOT is supported.)

### 4.4.1 How to execute

```plaintext
`cor_mon file_name [options]`
```

where

- `file_name` — correlator output file name (either cout format or KSP format)
- `-z[oom] zoom` — zooming factor in delay direction
- `-a[max] amppmax` — set maximum amplitude (default is 0.001)
- `-d[range] t1 t2` — set delay range from `t1` to `t2` (sec)
- `-h[alt]` — stop display by PP (default is continuous display)
- `-s[msec] sleep_msec` — suspend time (msec) between plots in case of continuous display (default is 200 msec)
- `-ch ch#1,ch#2[ch#3[,]ch#4[[,]...]]` — set display channel (example: `-ch 1, 4, 3`) (default is all channels)

4.4.2 Example

Fig.4.11 shows an example of 16 channel correlation functions.
Figure 4.11. An example of correlation function displayed by “cormon”. Red lines show a real part and blue lines an imaginary part.
Data check utilities

5.1 oscillo

"oscillo" displays time series of sampling data dynamically (dedicated to K5/VSSP format and PGPLOT)

5.1.1 How to execute

oscillo file_name [options]

where

- file_name - sampling data file name with K5/VSSP format
- t[span] tspan - time axis span (sec)
- h[alt] - stop display by a sweep (default is continuous display)
- s[msec] sleep_msec - suspend time (msec) between plots in case of continuous display (default is 0)

5.1.2 Example

Fig.5.1 shows an example of display in case of 64MHz×2bit×4ch sampling data.

![Figure 5.1. Time series plots displayed by “oscillo” for 64MHz×2bit×4ch sampling data.](image)

5.2 speana

“speana” displays spectrum of sampling data (dedicated to K5/VSSP format)

5.2.1 How to execute

speana data_file [options]
where  \textit{data\_file}  
\texttt{-m\[ode\] mode}  
\texttt{-pmode pmode}  
\texttt{-se[kibun] sekibun}  
\texttt{-ti[integration] sekibun}  
\texttt{-i[integration] sekibun}  
\texttt{-co[mment] comment}  
\texttt{-so[ffset] soffset}  
\texttt{-to[ffset] soffset}  
\texttt{-of[ffset] soffset}  
\texttt{-f1[khz] f1khz}  
\texttt{-f2[khz] f2khz}  
\texttt{-min[dbm] mindbm}  
\texttt{-max[dbm] maxdbm}  
\texttt{-nops}  

- sampling data file name with K5/VSSP format
- axis mode
  \texttt{mode} :  
  \begin{itemize}
  \item \texttt{YX}  
    \begin{itemize}
    \item \texttt{X} (frequency) axis mode  
      \begin{itemize}
      \item \texttt{0}: log scale  
      \item \texttt{1}: linear scale (default)  
      \end{itemize}
    \item \texttt{Y} (amplitude) axis mode  
      \begin{itemize}
      \item \texttt{0}: log scale (default)  
      \item \texttt{1}: linear scale (common for all channels)  
      \item \texttt{2}: linear scale (independent for channels)  
      \end{itemize}
    \end{itemize}
  \item \texttt{−value}: auto-correlation plot centered at \texttt{value}
  \item \texttt{−1}: auto-correlation (full range)
  \end{itemize}
- set plot device mode
  \begin{itemize}
  \item \texttt{0}: XWINDOW and PostScript file (pgplot.ps or gnuplot.ps) (default)
  \item \texttt{1}: PostScript file (pgplot.ps or gnuplot.ps) out only
  \item \texttt{2}: XWINDOW only
  \end{itemize}
- total integration period (sec)
- same as `-se` option
- same as `-se` option
- comment display in a plot
- start time offset (default is 0.0)
- same as `-so` option
- same as `-so` option
- lower frequency of frequency axis (kHz) (default is 0.0)
- higher frequency of frequency axis (kHz) (default is a video frequency)
- minimum value of amplitude axis (dBm) (default is automatic)
- maximum value of amplitude axis (dBm) (default is automatic)
- suppress Postscript file out (same as “−p 2”)

5.2.2 Example

Fig.5.2 shows an example of spectrum displayed by “speana” for 64MHz×2bit×4ch sampling data.
Figure 5.2. An example of spectrum plot displayed by “speana” for 64MHz×2bit×4ch sampling data

5.3 g_speana

“g_speana” displays spectrum of sampling data with a variety of data formats, such as, Mark-5B, VDIF, ADS, OCTAD besides K5/VSSP.

5.3.1 How to execute

g_speana data_file [options]

data_file – sampling data file name with a variety of data formats (Mark-5B, VDIF, ADS, OCTAD, K5/VSSP)

options (any order)
- m[ode] mode – axis mode
  mode :
  X: X (frequency) axis mode
    =0: log scale
    =1: linear scale (default)
  Y: Y (amplitude) axis mode
    =0: log scale (default)
    =1: linear scale (common for all channels)
    =2: linear scale (independent for channels)
  -value: auto-correlation plot centered at value
    -1: auto-correlation (full range)

- pmode pmode – set plot device mode
  0: XWINDOW and PostScript file (pgplot.ps or gnuplot.ps) (default)
  1: PostScript file (pgplot.ps or gnuplot.ps) out only
  2: XWINDOW only

- s[e]kibun sekibun – total integration period (sec)
- t[ime] integration sekibun – same as ‘-se’ option
- i[n]tegration sekibun – same as ‘-se’ option
- c[o]mment comment – comment display in a plot

50
An example of spectrum of sampling data display with Mark5B format.

$ g_speana cdasall2.m5b -m5b 
  
use '-m5b' to specify Mark5B format data.
use '-vdif' for VDIF format data.

$ g_speana Ver. 2016-06-17 compiled with FFTW3.0
  compiled for PGPLOT

Data File is cdasall2.m5b
File cdasall2.m5b opened (638349270 bytes)
Time(UT) MJD 226 16:57:31
fname : cdasall2.m5b
data file start time : Time(UT) MJD 226 16:57:31
data pick-up start time : Time(UT) MJD 226 16:57:31
sfkHz : 64000
adbit : 2
numch : 16
plot numbch : 16
plot ch# : 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
sekibun : 1.000000
soffset : 0.000000

CH# 1: Maximum data (dB, dBm) is -16.716162 -7.716162
CH# 2: Maximum data (dB, dBm) is -34.588219 -25.588217
CH# 3: Maximum data (dB, dBm) is -39.548599 -30.548597
CH# 4: Maximum data (dB, dBm) is -35.545822 -26.545824
CH# 5: Maximum data (dB, dBm) is -41.150509 -32.150509
CH# 6: Maximum data (dB, dBm) is -41.239742 -32.239742
CH# 7: Maximum data (dB, dBm) is -42.367371 -33.367371
CH# 8: Maximum data (dB, dBm) is -36.048561 -27.048559
CH# 9: Maximum data (dB, dBm) is -42.393009 -33.393009
CH# 10: Maximum data (dB, dBm) is -37.771706 -28.771704
All CH: Maximum and Minimum data (dBm) are -7.716162 -49.829594

Time elapsed for processing is 50.265196 sec
Created PostScript file => pgplot.ps
Type <RETURN> for next page: 

$
Fig.5.3 shows an example of spectrum displayed by “g_speana” for Mark5B format data (last 4 channels).

Figure 5.3. An example of spectrum displayed by g_speana for Mark5B format data (last 4 channels).

5.4 datachk

“datachk” checks K5/VSSP or K5/VSSP32 format data and outputs results to log files. Data are checked as follows. Bit-slip or bit-make is checked by counting the number of bits between adjacent headers of data. If bit-slip or bit-make is occurred, it is recorded in a log file. “datachk” can also check the percentage of positive sign of analog signals and can monitor the occurrence distribution of AD data by sampled value.

If AUX MISALIGN (misaligned AUX field) is found by this check, it can be recovered by using “aux_recov” program.

5.4.1 How to execute

datachk file_name [mode [logfile [errlog [keepmode]]]] [option]

where

file_name – sampling data file name with K5/VSSP format (default is “tds.data”)
mode – statistics display mode
  0 : no display (default)
      display first and last frame information and errored frames only
  1 : display percentage of ‘+’ sign data in analogue data
  2 : same as mode=0, but display all frames
  3 : display occurrence distribution by AD value by 1 second
  4 : display occurrence distribution of AD value for all data
logfile – log file name for summary output in case of mode=0. file name starting with ‘-’ means append mode open (default is no output)
errlog – log file name to contain errors detected in data file for mode=0. This file will not be created in case of no error (default is no output) file name starting with ‘-’ means append mode open
### Summary file

A summary file is generated when “datachk” is executed with `logfile` option. An example is as follows.

```plaintext
# File Name:
D:\IPVLBI\data\test02.dat
# FMT  A/D CH f(kHz) LPF(MHz):
VS32 1 1 32000 16
# Start and Last Time:
2006/318 23:20:28 84028
2006/318 23:25:27 84327
# Duration:
300
# Byte offset of 1st header:
0
# STATISTICS total bad discon discon_with_bitslip aux_sep EFLG:
300 1 0 0 147 0
# BIT SLIP:
1 26432
```

### Error log file

An error log file is generated when “datachk” is executed with `errlog` option. An example is as follows.

In case of `keepmode`=0

```plaintext
# Errorred Data File Name:
test02.dat
# FMT  A/D CH f(kHz) LPF(MHz):
VS32 1 1 32000 16
```

In case of `keepmode`=1 (rename mode)

```plaintext
# Errorred Data File Name:
test02.dat
# FMT  A/D CH f(kHz) LPF(MHz):
VS32 1 1 32000 16
# Renamed to:
test02.dat.0006.err
```

In case of `keepmode`=2 (copy mode)

```plaintext
# Errorred Data File Name:
test02.dat
# FMT  A/D CH f(kHz) LPF(MHz):
VS32 1 1 32000 16
# Copied to:
test02.dat.0007.err
```
5.5 vdifcheck

“vdifcheck” checks contents of VDIF format data.

5.5.1 How to execute

vdifcheck vdif_file [options]

where   

where

vdif_file – VDIF format data file name to check
options (any order)
-f[ormat] 0|1|2|3|99 – set display format
  0 : automatic according to VDIF format and edv# (default)
  1 : legacy format
  2 : same as mode=0, but edv# is not considered
  3 : set edv# compulsory as that of KASHIMA
  99 : for debug
-d[mode] 0|1|2 – set display mode
  0 : display first and last frame only (default)
  1 : display every frame
  2 : display frame on the second besides first and last frames
-n[frame] nmax – set # of frames to read
-l[egacy] – set header length 16 bytes compulsory
-s[kip] n – skip first n bytes data
-k[ashima][byteframe] – set header format to Kashima format compulsory and set frame
                      length bytesframe-bytes compulsory

5.5.2 Example

$ vdifcheck kas34_2014112035830.3sec.vdif

*****************************************************************************
* VDIF data check *
* Ver 1.11 2014-06-12 by T.KONDO/NICT *
*****************************************************************************
Data File : kas34_2014112035830.3sec.vdif
1st header information is as follows

[Header Information]
VDIF header (raw) : 00928E66 1C000000 000000A4 00000000
extended header (raw) : 01800800 ACABFEED 6873616B 00616D69
Invalid flag = 0 : Legacy flag = 0
Sec from ref epoch = 9604710 : Ref epoch = 28 (2014/04/22 03:58:30)
Frame # = 0 : VDIF Ver# = 0 : # Chs(log2) = 0
Frame length in 8 byte unit = 164 (= 1312 bytes)
Data type = 0 : #bits/sample-1 = 0 : Thread ID = 0
Station ID (A2) = (=0 in number)
EDV : 1
Extended Data W1 W2 W3 W4 : 800800 ACABFEED 6873616B 00616D69
--- in case of NICT Extended Format ---
  uflag = 1 : srate = 2048 : Sync block = ACABFEED
  DAS/station name (A8) : kashima

===========================================================================

Sampling

<table>
<thead>
<tr>
<th>HH:MM:SS</th>
<th>FRAME#</th>
<th>I</th>
<th>L</th>
<th>VER</th>
<th>CHS</th>
<th>#Bytes</th>
<th>TID</th>
<th>SID (MHz)</th>
<th>SYNC</th>
<th>DAS/Stat</th>
</tr>
</thead>
<tbody>
<tr>
<td>03:58:30</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1312</td>
<td>0</td>
<td>2048</td>
<td>ACABFEED</td>
<td>kashima</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>03:58:32</td>
<td>199999</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1312</td>
<td>0</td>
<td>2048</td>
<td>ACABFEED</td>
<td>kashima</td>
</tr>
</tbody>
</table>

===========================================================================

============================================================================

Summary

Sampling
5.6 m5check

“m5check” checks the contents of Mark5 format data.

5.6.1 How to execute

m5check m5file [mode]

or

m5check m5file [options]

where

m5file - Mark-5 data file name to check
mode - set format to check
0 : with parity and 8|16|32|64 bits/word
1 : without parity and 8|16|32|64 bits/word
2 : display old style
when mode is neglected, all modes are checked

options (any order)
-a[ll] - all data are checked (only for Mark5B format data)
-old - execute with old style
-f - omit checking sync block. if one of options below is set, this option
is neglected
-t[rack] ntrack - set # of tracks (8|16|32|64). Default is 32
-vbla - set VLBA format (default is Mark-5 format)
-p[arity] - set with parity mode (default is without parity)
-nrzm - set NRZM mode (default is non-NRZM mode)
-s[kip] samples - set # of samples to be skipped at first

5.6.2 Examples

An example of checking Mark5B format data.

kondo@io:~/chkdata/m5b$ m5check JPddcall.m5b
***********************************************************
* Mark-5 data structure analysis *
* Ver 1.82 2012-11-26 by T.KONDO/NICT *
***********************************************************
1st STEP: Checking Mark5B format
Data File : JPddcall.m5b
Now analizing the data file is Mark5B or not ....
OK this is Mark 5B Format data
TIME CODE CRC-
<table>
<thead>
<tr>
<th>FRAME#</th>
<th>SYNC</th>
<th>UUUUTFFF</th>
<th>MJDssssss.ssss</th>
<th>16</th>
<th>DATA#1</th>
<th>DATA#2</th>
<th>Mbps</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000</td>
<td>ABADDEED</td>
<td>F00F0000</td>
<td>22661050</td>
<td>0000A474</td>
<td>5CB997F7</td>
<td>538A10F3</td>
<td>N.A.</td>
</tr>
<tr>
<td>00001</td>
<td>ABADDEED</td>
<td>F00F0001</td>
<td>22661050</td>
<td>0000A474</td>
<td>5B5B6BB7</td>
<td>994A5166</td>
<td>N.A.</td>
</tr>
<tr>
<td>00002</td>
<td>ABADDEED</td>
<td>F00F0002</td>
<td>22661050</td>
<td>0012471</td>
<td>8975153B</td>
<td>055BC52A</td>
<td>N.A.</td>
</tr>
<tr>
<td>00003</td>
<td>ABADDEED</td>
<td>F00F0003</td>
<td>22661050</td>
<td>0012471</td>
<td>7DAD8195</td>
<td>1744AD2E</td>
<td>N.A.</td>
</tr>
<tr>
<td>00004</td>
<td>ABADDEED</td>
<td>F00F0004</td>
<td>22661050</td>
<td>0002247B</td>
<td>E0E5EE59</td>
<td>19526019</td>
<td>N.A.</td>
</tr>
<tr>
<td>00005</td>
<td>ABADDEED</td>
<td>F00F0005</td>
<td>22661050</td>
<td>0002247B</td>
<td>185A2A95</td>
<td>82166A5D</td>
<td>N.A.</td>
</tr>
<tr>
<td>00006</td>
<td>ABADDEED</td>
<td>F00F0006</td>
<td>22661050</td>
<td>0002247B</td>
<td>ADC29CD9</td>
<td>4941DA55</td>
<td>N.A.</td>
</tr>
<tr>
<td>00007</td>
<td>ABADDEED</td>
<td>F00F0007</td>
<td>22661050</td>
<td>0003A47E</td>
<td>0994A026</td>
<td>2BADDA00</td>
<td>N.A.</td>
</tr>
<tr>
<td>00008</td>
<td>ABADDEED</td>
<td>F00F0008</td>
<td>22661050</td>
<td>0003A47E</td>
<td>48A66BF7</td>
<td>E94525D5</td>
<td>N.A.</td>
</tr>
<tr>
<td>00009</td>
<td>ABADDEED</td>
<td>F00F0009</td>
<td>22661050</td>
<td>0004246F</td>
<td>F40B9359</td>
<td>A7D41C19</td>
<td>N.A.</td>
</tr>
</tbody>
</table>

**************** SUMMARY of DATA FORMAT ANALYSIS****************

File Name : JPddcall.m5b
Data Format : Mark-5B
File Size : 534277415 bytes
Total # of Frames (estimated from file size): 53342.393670
1st Header Information
   Time (MJD HH:MM:SS.SSSS) : 226 16:57:30.0000
   Frame # (in a second) : 0
   User Specified (16 bits) : F00F
   TVG data flag : 0
Data Rate (Mbps) : 2048.0
Frames/sec : 25600
Data Length (sec): 2.083687

******************************************************************************

$
6 Data format conversion

6.1 k5tom5b

“k5tom5b” convert K5 format data to Mark5B format data

6.1.1 How to execute

k5tom5b k5name1 [k5name2 k5name3 k5name4 [m5name]] [options]

where
k5name1 – K5 file name for CH#01-04 or CH#01
k5name2 – K5 file name for CH#05-08 or CH#02
k5name3 – K5 file name for CH#09-12 or CH#03
k5name4 – K5 file name for CH#13-16 or CH#04

if K5 file names after the 2nd are not accompanied by directory name, the directory of 1st K5 file is adopted for these files

m5name – Mark5 file name to be created

options (any order)
-o m5name – Mark5 file name to be created
-d m5dir – directory for Mark5 file out (default : mark5 directory)
-s offset – set offset time from data head (sec) (default : 0 )
-p period – data period to convert (sec) (default : all)
-e2bit n – 2bit data encode mode
    n=1 : offset binary 0 1 2 3
    n=2 : twisted offset binary 0 2 1 3
    n=3 : signed binary 1 0 2 3
        (default is 2)
-nocheck – no header error check (adopted for 2nd and later headers)
-ch1 nn – set K5 channel corresponding to Mark5 ch #1
         nn is K5 channel number (1-16)
-ch2 nn – set K5 channel corresponding to Mark5 ch #2
-.................
-ch16 nn – set K5 channel corresponding to Mark5 ch #16
-chall n1:n2: ...:n16 – set K5 channels corresponding to Mark5 all channels
    n1 – K5 channel number (1-16) corresponding to Mark5 ch 1
    n2 – K5 channel number (1-16) corresponding to Mark5 ch 2
    .................
    n16 – K5 channel number (1-16) corresponding to Mark5 ch 16

[Environmental variables]

k5tom5b env

M5DIR – default directory for Mark5 data
M5VEX – default directory for Mark5 vex file

6.1.2 Example

$ k5tom5b ./tds.k5a tds.k5b tds.k5c tds.k5d data.m5bA data.m5b

******************************************************************************
* K5/VSSP to Mark-5B Data Format Converter *
* k5tom5b (Ver 1.70 2016-08-15) by T.KONDO/NICT *

57
Second Length : 30.000000
Second Length : 30.000000
Second Length : 30.000000
Second Length : 30.000000

K5 file(s)
FILE1 CH# 1- 4 : ./tds.k5a
   (A/D(bits) 1 CHs 4 SFreq(kHz) 16000 Time 23:59:54 sec 86394) VSSP32
FILE2 CH# 5- 8 : ./tds.k5b
   (A/D(bits) 1 CHs 4 SFreq(kHz) 16000 Time 23:59:54 sec 86394) VSSP32
FILE3 CH# 9-12 : ./tds.k5c
   (A/D(bits) 1 CHs 4 SFreq(kHz) 16000 Time 23:59:54 sec 86394) VSSP32
FILE4 CH# 13-16 : ./tds.k5d
   (A/D(bits) 1 CHs 4 SFreq(kHz) 16000 Time 23:59:54 sec 86394) VSSP32

Mark5B file (created) : data.m5b
Observation Date : 3digit MJD = 589
   : Year = 2016
   : Month = 7
   : Day = 20
   : Total Day = 202
Conversion length (sec) : 30

Mark5 file (data.m5b) has been created.
Time elapsed (sec) for One obs process is 52.000000

6.2 k5tom5

“k5tom5” convert K5 format data to Mark5 format data

6.2.1 How to execute

k5tom5 jjj yyyy mm dd k5name1 [k5name2 k5name3 k5name4 [m5name]] [options]
or
k5tom5 -i -infofile make -v vexfile [options]
    .... for information file creation mode

where
jjj yyyy mm dd
   - observation date in three ways as follows
     jjj : lower 3digits of MJD
or
yyyy mm dd : 4 digit year, 2 digit month, 2 digit day of month
*** K5/VSSP to Mark-V Data Format Converter ***
* k5tom5 (Ver 1.21 2005-09-17) by T.KONDO/NICT *
************************************************************************
-------- RUN CONDITION --------

Example 1: conversion using default information file

set date of observation by year and total day and use default information file “k5tom5info.txt”

k5tom5 20031015 02880001.k5a 02880001.k5b 02880001.k5c 02880001.k5d m5test.dat

--- Environmental variables ---

k5tom5 env

M5DIR – default directory for Mark5 data
M5VEX – default directory for Mark5 vex file
Example 2: Create an information file

create conversion information file using the VEX file "/home/vlbi/mark5/gg057c.vex". Set station ID “Ks”.

```bash
k5tom5 -i -/home/vlbi/mark5/k5tom5info.txt -v /home/vlbi/mark5/gg057c.vex -sid Ks
K5tom5 running under Information File create mode
info file (/home/vlbi/mark5/k5tom5info.txt) will be created (updated).
VEX file name --- /home/vlbi/mark5/gg057c.vex
SITES (Station ID) defined are
ID SITE NAME
----------------
Pt VLBA_PT
Kp VLBA_KP
La VLBA_LA
Br VLBA_BR
Fd VLBA_FD
Nl VLBA_NL
Ov VLBA_OV
60
```

Mark5 file (m5test.dat2) has been created.

Time elapsed (sec) for One obs process is 44.000000
Selected Station ID = Ks

search_site: No CLOCK info for KASHIM34 included in the VEX FILE.
search_site: So all 0 for clock information was set.

Detailed site information
  site definition : KASHIM34
  site name : KASHIM34
  site ID : Ks
  site position : -3997649.222000 3276690.753000 3724278.823000
  site clock
    validity epoch : 0 0000
    clock epoch : 0 0000
    clock offset : 0.000000\text{$\text{e+000}$}
    clock rate : 0.000000\text{$\text{e+000}$}

Scan # for mode get is 1
mode is huysS
Mode was taken from Scan #1 as huysS

TRACK and FREQUENCY information for KASHIM34
BARREL ROLL : off
FREQDEF = 2034.99MHz8x16MHz TRACKDEF = MKIV.8Ch2bit1to4
adbit= 2 sample_rate= 32000000.000000

<table>
<thead>
<tr>
<th>bb</th>
<th>HS</th>
<th>Tr</th>
<th>AD</th>
<th>fo</th>
<th>chan</th>
<th>bbcid</th>
<th>BBC#</th>
<th>RF(Hz)</th>
<th>S</th>
<th>VBW(Hz)</th>
<th>PASS</th>
</tr>
</thead>
</table>
| 1  | 2  | sign 1 &CH01 &BBC01 1 2034990000.0 L 16000000.0
| 2  | 4  | sign 2 &CH01 &BBC01 1 2034990000.0 L 16000000.0
| 3  | 6  | sign 3 &CH01 &BBC01 1 2034990000.0 L 16000000.0
| 4  | 8  | sign 4 &CH01 &BBC01 1 2034990000.0 L 16000000.0
| 5  | 10 | mag 1 &CH01 &BBC01 1 2034990000.0 L 16000000.0
| 6  | 12 | mag 2 &CH01 &BBC01 1 2034990000.0 L 16000000.0
| 7  | 14 | mag 3 &CH01 &BBC01 1 2034990000.0 L 16000000.0
| 8  | 16 | mag 4 &CH01 &BBC01 1 2034990000.0 L 16000000.0
| 9  | 18 | sign 1 &CH02 &BBC02 2 2034990000.0 L 16000000.0
| 10 | 20 | sign 2 &CH02 &BBC02 2 2034990000.0 L 16000000.0
| 11 | 22 | sign 3 &CH02 &BBC02 2 2034990000.0 L 16000000.0
| 12 | 24 | sign 4 &CH02 &BBC02 2 2034990000.0 L 16000000.0
| 13 | 26 | mag 1 &CH02 &BBC02 2 2034990000.0 L 16000000.0
| 14 | 28 | mag 2 &CH02 &BBC02 2 2034990000.0 L 16000000.0
| 15 | 30 | mag 3 &CH02 &BBC02 2 2034990000.0 L 16000000.0
| 16 | 32 | mag 4 &CH02 &BBC02 2 2034990000.0 L 16000000.0
| 17 | 3 | sign 1 &CH03 &BBC01 1 2034990000.0 U 16000000.0
| 18 | 5  | sign 2 &CH03 &BBC01 1 2034990000.0 U 16000000.0
| 19 | 7  | sign 3 &CH03 &BBC01 1 2034990000.0 U 16000000.0
| 20 | 9  | sign 4 &CH03 &BBC01 1 2034990000.0 U 16000000.0
| 21 | 11 | mag 1 &CH03 &BBC01 1 2034990000.0 U 16000000.0
| 22 | 13 | mag 2 &CH03 &BBC01 1 2034990000.0 U 16000000.0
| 23 | 15 | mag 3 &CH03 &BBC01 1 2034990000.0 U 16000000.0
| 24 | 17 | mag 4 &CH03 &BBC01 1 2034990000.0 U 16000000.0
| 25 | 19 | sign 1 &CH04 &BBC02 2 2034990000.0 U 16000000.0
| 26 | 21 | sign 2 &CH04 &BBC02 2 2034990000.0 U 16000000.0
| 27 | 23 | sign 3 &CH04 &BBC02 2 2034990000.0 U 16000000.0
| 28 | 25 | sign 4 &CH04 &BBC02 2 2034990000.0 U 16000000.0
| 29 | 27 | mag 1 &CH04 &BBC02 2 2034990000.0 U 16000000.0
| 30 | 29 | mag 2 &CH04 &BBC02 2 2034990000.0 U 16000000.0
| 31 | 31 | mag 3 &CH04 &BBC02 2 2034990000.0 U 16000000.0
| 32 | 33 | mag 4 &CH04 &BBC02 2 2034990000.0 U 16000000.0
| 33 | 2  | sign 1 &CH05 &BBC03 3 2139990000.0 L 16000000.0
| 34 | 4  | sign 2 &CH05 &BBC03 3 2139990000.0 L 16000000.0
| 35 | 6  | sign 3 &CH05 &BBC03 3 2139990000.0 L 16000000.0
| 36 | 8  | sign 4 &CH05 &BBC03 3 2139990000.0 L 16000000.0
| 37 | 10 | mag 1 &CH05 &BBC03 3 2139990000.0 L 16000000.0
| 38 | 12 | mag 2 &CH05 &BBC03 3 2139990000.0 L 16000000.0
| 39 | 14 | mag 3 &CH05 &BBC03 3 2139990000.0 L 16000000.0
| 40 | 16 | mag 4 &CH05 &BBC03 3 2139990000.0 L 16000000.0
| 41 | 18 | sign 1 &CH06 &BBC03 3 2139990000.0 U 16000000.0
| 42 | 20 | sign 2 &CH06 &BBC03 3 2139990000.0 U 16000000.0
| 43 | 22 | sign 3 &CH06 &BBC03 3 2139990000.0 U 16000000.0
| 44 | 24 | sign 4 &CH06 &BBC03 3 2139990000.0 U 16000000.0
| 45 | 26 | mag 1 &CH06 &BBC03 3 2139990000.0 U 16000000.0
| 46 | 28 | mag 2 &CH06 &BBC03 3 2139990000.0 U 16000000.0

61
default Mark-V data format is as follows

data encode -- NRZL without parity

data format -- Mark-IV

#_of_track = 64

Information file (/home/vlbi/mark5/k5tom5info.txt) created!!

6.3 k5tovdif

“k5tovdif” converts K5 format data to VDIF format data

6.3.1 How to execute

k5tovdif k5name1 [k5name2 k5name3 k5name4 [vdifname]] [options]

where

k5name1  – K5 file name for CH#01-04 or CH#01
k5name2  – K5 file name for CH#05-08 or CH#02
k5name3  – K5 file name for CH#09-12 or CH#03
k5name4  – K5 file name for CH#13-16 or CH#04

if K5 file names after the 2nd are not accompanied by directory name, the directory of 1st K5 file is adopted for these files

vdifname  – VDIF file name to be created

options (any order)

-o vdifname  – VDIF file name to be created (default: extension of 1st K5 name is replaced by ‘.vdif’)
-dir vdir  – directory for VDIF file out (default: K5 directory)
-so soffset  – set offset time from data head (sec) (default: 0)
-p period  – data period to convert (sec) (default: all)
-das das_name  – set station name or compulsoly set station name (in case of VSSP32 station or PC name got from it’s AUX data)
-sid sid  – set station id (A2) (in case of VSSP32 sid got from it’s AUX data)
-date jjj/yyyymmdd/yyyyyddd  – set date in case of VSSP (not VSSP32) format as follows

jjj : lower 3digits of MJD

yyyymmdd : 4 digit year, 2 digit month, 2 digit day of month

yyyyyddd : 4 digit year, 3 digit day of year

-e2bit n  – 2bit data encode mode

n=1 : offset binary 0 1 2 3

n=2 : twisted offset binary 0 2 1 3

n=3 : signed binary 1 0 2 3

(default is 2)

-nocheck  – no header error check (adopted for 2nd and later headers)
-fr[amelength] frame_bytes
– set frame length without header (in byte) (default 1280)
-\texttt{num[frame\_sec]} \texttt{nframe}
   – set # of frames per second (default: $2048 \times 10^6/(1280) = 200000$)
-\texttt{ch1} \texttt{nn}
   – set K5 channel corresponding to VDIF ch #1
      \texttt{nn} is K5 channel number (1-16)
-\texttt{ch2} \texttt{nn}
   – set K5 channel corresponding to VDIF ch #2
-\texttt{ch16} \texttt{nn}
   – set K5 channel corresponding to VDIF ch #16
-\texttt{chall} \texttt{n1:n2: \cdots :n16}
   – set K5 channels corresponding to VDIF all channels
      \texttt{n1} – K5 channel number (1-16) corresponding to VDIF ch 1
      \texttt{n2} – K5 channel number (1-16) corresponding to VDIF ch 2
      \texttt{n16} – K5 channel number (1-16) corresponding to VDIF ch 16

\begin{itemize}
\item \textbf{[Environmental variables]}
\end{itemize}
\begin{verbatim}
k5tovdif env
VDIFDIR – default directory for VDIF data
\end{verbatim}

\section*{6.4 \texttt{ads2k5}}
“\texttt{ads2k5}” converts ADS3000+(DBBC mode) format to K5 format data

\subsection*{6.4.1 How to execute}
\begin{verbatim}
\texttt{ads2k5} \texttt{adsname[options]}
\end{verbatim}
where
\begin{itemize}
\item \texttt{adsname} – ADS3000-DBBC mode data file name
\item \texttt{options} (any order)
\item \texttt{-c} \texttt{channel}
   – pickup channel (1-16) for 1ch conversion mode (same as ‘-1ch option’.
      default 4x4ch mode)
\item \texttt{-u} \texttt{unit}
   – pickup unit (1-4) for 4 ch conversion mode (-c option is stronger than
      this option) (default 4x4ch mode)
\item \texttt{-o} \texttt{k5name}
   – K5 file name to be created (default : see below)
\item \texttt{-d} \texttt{k5dir}
   – directory for k5 file out
\item \texttt{-fsampl} \texttt{fmhz}
   – channel sampling frequency (MHz) (deflt=32)
\item \texttt{-adbit} \texttt{adbit}
   – # of AD conversion bit (deflt=1)
\item \texttt{-1ch} \texttt{ch}
   – set K5 data to 1ch mode and set pick up ADS3000-DBBC ch (starting
      at 1)
\item \texttt{-4ch} \texttt{ch1} \texttt{ch2} \texttt{ch3} \texttt{ch4}
   – set K5 data to 4 ch mode and set pick up channels from ADS3000-
      DBBC (default: convert to 4x4ch K5 files)
\item \texttt{-2bit} \texttt{mode}
   – select 2bit AD decode table
      1: offset binary 0,1,2,3 (no conversion: default)
      2: Mark V standard 0,2,1,3
      3: signed integer 1,0,2,3
\item \texttt{-t} \texttt{YYYYDDDHHMMSS} | \texttt{YYYY/DDD-HH:MM:SS} or \texttt{HHMMSS|HH:MM:SS} or \texttt{YYYYDDD|YYYY/DDD}
   – set start date and time of data. If this is omitted, start time is taken
      from ADS filename.
\item \texttt{-s} \texttt{soffset}
   – set offset time from data head (sec) (default : 0 )
\end{itemize}
-p period – data period to convert (sec) (default : all)
-vssp – force K5 format to VSSP (default is VSSP32)
-monit – monitor run parameters
-adsbit adsbit – ADS AD bits (2|4|8) (deflt=2).
    if adbit>=4, adsbit = adbit
-adsnumch adsnumch – ADS # of channels (1|8) (deflt=8)

[Naming rule for K5 file created]

(ADS : ADS3000 file name or name given by ‘-o’ option)

4x4ch mode
    ADS.k5a ---- for group#1 (ch01-04)
    ADS.k5b ---- for group#2 (ch05-08)
    ADS.k5c ---- for group#3 (ch09-12)
    ADS.k5d ---- for group#4 (ch13-16)
4ch mode
    ADS.k5[a|b|c|d]
1ch mode
    ADS.k5-NN ---- where NN is channel number 01-16

[Environmental variables]

ads2k5 env

    K5.CH_TABLE – operator defined CH allocation table
    ADSDIR – default directory for ADS data
    K5DIR – default directory for K5 file out

Note:
If ADS3000 file name satisfies following condition, start time and sampling frequency are properly estimated
from the file name.

1. File name has a following structure.
   xxx...xxxx_YYYYYDDDHHMMSS.raw
   where xxx...xxxx — any characters YYYYDDDHHMMSS – start time (year,day of year, hour, minute, second)
2. “xxx...xxxx” includes a sampling information block like “16MHz” or “8Mps” (don’t care about upper-case
   or lower-case of characters for MHz or Mps).
3. This information block should be precede by a non-number ASCII character or start of the file name, e.g.,
   ....dbbc_16MHz_... or ...xyz.32MSPS.xxxxxx

6.5 m5btok5

“m5btok5” converts Mark5B format data to K5 format data

6.5.1 How to execute

m5btok5 m5bname [options]
where  

where \( m5bname \) – Mark5B data file name 

`options` (any order)

- `-o k5name` – K5 file name to be created (default: see below)
- `-d k5dir` – directory for k5 file out (default: Mark5 directory)
- `-bs nbbitsream` – set Mark5B # of bit-stream (1,2,4,8,16,32)
- `-fs qmpl fmhz` – channel sampling frequency (MHz) (default=4)
- `-adbit adbit` – # of AD conversion bit (default=1)
- `-1ch ch` – set K5 data to 1ch mode and set pick up Mark5B ch (starting at 1)
- `-4ch ch1 ch2 ch3 ch4` – set K5 data to 4ch mode and set pick up channels from Mark5B data
- `-2bit mode` – select 2bit AD decode table
  1: offset binary 0,1,2,3 (no conversion)
  2: Mark V standard 0,2,1,3 (default)
  3: signed integer 1,0,2,3

[Naming rule for K5 file created]

(MK5: original mark5 name)

4ch mode

MK5.k5a ---- for group#1 (ch01-04)
MK5.k5b ---- for group#2 (ch05-08)
MK5.k5c ---- for group#3 (ch09-12)
MK5.k5d ---- for group#4 (ch13-16)

1ch mode

MK5.k5-NN ---- where NN is channel number 01-16

[Environmental variables]

```
  m5btok5 env
```

  M5DIR – default directory for Mark5 data
  K5DIR – default directory for K5 file out

6.6 m5tok5

“m5tok5” converts Mark5B format data to K5 format data

6.6.1 How to execute

Mode 1

```
m5tok5 m5name [options]
```

where  

where \( m5name \) – Mark5 data file name 

`options` (any order)

- `-c channel` – pickup channel (1-16) for 1ch conversion mode (default: all group and 4ch mode)
- `-g group` – pickup group (1-4) for 4ch conversion mode (omitted when ‘-c’ option is set) (default: all group)
-i infofile – information file name that contains
  track vs channel table
  bit position vs channel table
  group# vs channel table
  (default “m5tok5info.txt”)
  ‘-i infofile’ means that create information file named infofile
  ‘-i make’ means that create default information file
-o k5name – K5 file name to be created (default : see below)
-d k5dir – directory for k5 file out (default Mrak5 directory)
-s soffset – set offset time from data head (sec) (default : 0 )
-p period – data period to convert (sec) (default : all)
-r – reverse track order when fanout=2 or more
-v vex_file – VEX file name to be read when “information file creation mode” is set
-sid stat_id – station ID when “information file creation mode” is set
-scan scan# – scan# for MODE get when “information file creation mode” is set
  (default : 1)
-subp subpass – set subpass at information file creation mode (default is “A”)
-odd – 32bit (4byte) shift in reading Mark5 data (this works only for 64 track mode)
-monit – run condition monitor ON

Mode 2

m5tok5 m5name k5file bit1 [bit2 bit3 .. bitN] [options2]

where
  m5name – Mark5 data file name
  k5file – K-5 file name to be created
  bit1 – 1st pick up bit position (0- max track#)
  bit2 – 2nd pick up bit position (0- max track#)
  ....
  bitN – Nth pick up bit position (0- max track#)

options2 (any order)
  -track ntrack – # of Mark5 track (8,16,32,64) (deflt=32)
  -fsampl fnhz – channel sampling frequency (MHz) (deflt=4)
  -adbit adbit – # of AD conversion bit (deflt=1)
  -nch nunch – # of channels in K5 format (1 or 4) (deflt=4)
  -parity – with parity data
  -nparity – non-parity data (default)
  -vlba – set VLBA mode (default mark IV mode)
  -s soffset – set offset time from data head (sec) (default : 0 )
  -p period – data period to convert (sec) (default : all)
  -odd – 32bit (4byte) shift in reading Mark5 data (this works only for 64 track mode)

[Naming rule for K5 file created]

(MK5 : original mark5 name)

4ch mode
  MK5.k5a ---- for group#1 (ch01-04)
MK5.k5b ---- for group#2 (ch05-08)
MK5.k5c ---- for group#3 (ch09-12)
MK5.k5d ---- for group#4 (ch13-16)
1ch mode
MK5.k5-NN ---- where NN is channel number 01-16

[Environmental variables]

m5tok5 env

M5DIR – default directory for Mark5 data
M5VEX – default directory for VEX file

6.7 vdif2k5

“vdif2k5” converts VDIF format data to K5 format data

6.7.1 How to execute

vdif2k5 vdiffile [options]

where vdiffile – VDIF data file name

options (any order)
-o k5name – K5 file name to be created (default: see below)
-d k5dir – directory for k5 file out (default VDIF directory)
-fsampl fmhz – channel sampling frequency (MHz) (default=4)
-adbit abit – # of AD conversion bit (default=1)
-1ch ch – set K5 data to 1ch mode and set pick up VDIF ch (starting at 1)
-4ch ch1 ch2 ch3 ch4 – set K5 data to 4 ch mode and set pick up channels from VDIF data
-2bit mode – select 2bit AD decode table
1: offset binary 0,1,2,3 (no conversion) (default)
2: Mark V standard 0,2,1,3
3: signed integer 1,0,2,3
-s soffset – set offset time from data head (sec) (default: 0)
in case of negative value: absolute value means skip over bytes

[Naming rule for K5 file created]

(VDIF : original VDIF name)

4ch mode
VDIF.k5a ---- for group#1 (ch01-04)
VDIF.k5b ---- for group#2 (ch05-08)
VDIF.k5c ---- for group#3 (ch09-12)
VDIF.k5d ---- for group#4 (ch13-16)
1ch mode (from multi channel VDIF)
MK5.k5-NN ---- where NN is channel number 01-16
1ch mode (from 1ch VDIF)
VDIF.k5

67
6.8 **vdif2m5b**

“vdif2m5b” converts VDIF format data to Mark5B format data

### 6.8.1 How to execute

vdif2m5b *vdiffile [options]*

where  

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-o m5bname</code></td>
<td>Mark5B file name to be created (default: see below)</td>
</tr>
<tr>
<td><code>-d m5dir</code></td>
<td>Directory for Mark5B file out (default: VDIF directory)</td>
</tr>
<tr>
<td><code>-fsampl fMhz</code></td>
<td>Channel sampling frequency (MHz) (default=4)</td>
</tr>
</tbody>
</table>
| `-2bit mode` | Select 2bit AD decode table  
1: Offset binary 0,1,2,3 (no conversion)  
2: Mark V standard 0,2,1,3 (default)  
3: Signed integer 1,0,2,3 |
| `-s sofset` | Set offset time from data head (sec) (default: 0) |
| `-p period` | Data period to convert (sec) (default: all) |
| `-type1`    | Outfile naming Type 1 (replace file extension to “m5b”) |
| `-pid thread` | Set pickup thread ID compulsory |
| `-ch1 nn`   | Set VDIF channel corresponding to Mark5B ch #1  
|             | `nn` is VDIF channel number (1-16) |
| `-ch2 nn`   | Set VDIF channel corresponding to Mark5B ch #2  
|             | `nn` is VDIF channel number (1-16) |
| `...`       | Set VDIF channels corresponding to Mark5B all channels  
|             | `n1:n2:...:n16` |
| `-chall n1:n2:...:n16` | Set VDIF channels corresponding to Mark5B all channels  
|             | `n1` - VDIF channel number (1-16) corresponding to Mark5B ch 1  
|             | `n2` - VDIF channel number (1-16) corresponding to Mark5B ch 2  
|             | `...` |
|             | `n16` - VDIF channel number (1-16) corresponding to Mark5B ch 16 |

[Naming rule for Mark5B file created]

(VDIF.vdif: original VDIF name)

- Type 0  VDIF.vdif.m5b (default)
- Type 1  VDIF.m5b
Actual correlation processing

Correlation processing is divided into two steps as follows.

1. Fringe search: determine clock parameters (offset and rate) by baseline.
2. Processing all scans: process all scan data using clock parameters determined by fringe search.

7.1 Fringe search

Correlation processing for geodetic VLBI data is usually carried out with a small number of lags such as 32 to increase the throughput of processing. To keep a correlation peak at the center of the lag window, clock offset should be determined with an accuracy of better than a few tenths percentage of a sampling period. Fringe search is a correlation processing to find out fringes and to determine a clock offset between two stations.

A-priori delay necessary for correlation processing is calculated from time of observation, station positions, source position, earth orientation parameters, and clock error. It is difficult to obtain an accurate clock error between two stations in advance, so that a clock error measured by GPS at each station is usually used as a temporal value to calculate a-priori delay. If there is no GPS clock measurement, zero is used as a temporal value. In case of geodetic VLBI lasting for 24 hours or more, clock rate is also determined. The concrete method of fringe search and clock parameters determination is as follows.

1. Determination of the scan for fringe search

- without clock rate determination: a scan observed a strong source around the middle of the session
- with clock rate determination: two scans that observed a strong source around the beginning and end of the session

Source information is found in a schedule file. It is also obtained by executing “apri_calc” with a schedule monitoring mode as follows.

```
apri_calc -/home/vlbi/sked/sample.skd
```

or

```
apri_calc /home/vlbi/sked/sample.skd -monit
```

where "/home/vlbi/sked/sample.skd" is a schedule file

then observation information is displayed as follows.

```
apri_calc (Ver. 2016-10-12)
---------------------------------------------- RUN CONDITION ----------------------------------------------
K5 file naming type is Type 1 : sidDDDDNNN.dat (SKED deflt)
---------------------------------------------- Schedule file monitor ****************************
SkdMonit: Schedule file = ipvlbisked/sample.skd
SkdMonit: Expcode = KS07235
SkdMonit: Total Scan # = 593
SkdMonit: Total Star # = 16
SkdMonit: 1st Scan = 2007/08/23 01:15:00 3C84
SkdMonit: Last Scan = 2007/08/24 00:46:20 3C273B
SkdMonit: Station Table ------
SkdMonit: G --- KOGANEI
SkdMonit: R --- KASHIM11
SkdMonit: Y --- TATEYAMA
SkdMonit: Star Table ---------------------
SkdMonit: NAME1 NAME2 R.A.(deg) DEC(deg) EPOCH
SkdMonit: 1 0059+581 $ 15.690677 58.403093 2000.000000
SkdMonit: 2 0316+413 3C84 49.950667 41.511695 2000.000000
SkdMonit: 3 0420-014 $ 65.815836 -1.342518 2000.000000
SkdMonit: 4 0552+398 $ 88.878357 39.813657 2000.000000
SkdMonit: 5 0727-115 $ 112.579635 -11.686833 2000.000000
SkdMonit: 7 1226+023 3C273B 187.277915 2.052389 2000.000000
```
Then select a strong source such as "3C273B", and execute "apri calc" with a schedule monitoring mode with the option “-source” as follows.

```
apri_calc -/home/vlbi/sked/sample.skd -source 3C273B
```

or

```
apri_calc /home/vlbi/sked/sample.skd -monit -source 3C273B
```

A certain baseline can be specified if necessary by using the “-baseid” option as follows.

```
apri_calc /home/vlbi/sked/sample.skd -monit -source 3C273B -baseid RY
```

where "RY" is a combination of Station IDs. Then scan information is displayed as follows.

```
SkdMonit: --------- PICKUP SCAN TABLE ----------
SkdMonit: SCAN# SOURCE YYYY/DDD HH:MM:SS DURA STATION_IDS
SkdMonit:  10 3C273B 2007/235 01:36:03 62 G R Y

......................
SkdMonit: 579 3C273B 2007/236 00:10:11 62 G R Y
SkdMonit: 586 3C273B 2007/236 00:29:04 62 G R Y
SkdMonit: 593 3C273B 2007/236 00:46:20 62 G R Y
```

Column named "SCAN#" shows the scan number.

2. Calculation of a-priori values

Execute “apri_calc” for a scan selected by the previous step as follows.

```
apri_calc /home/vlbi/sked/sample.skd -start 10 -stop 10 -apedir /home/vlbi/corrapi
```

where “-apedir” option sets the directory of a-priori files created. If you want to execute for all scans selected, execute as follows.

```
apri_calc /home/vlbi/sked/sample.skd -source 3C273B -baseid RY -apedir /home/vlbi/corrapi
```

See “How to run apri_calc” (section 3.1) for details. Since a-priori file is a text file, you can edit it easily if necessary (see Appendix A for the format of a-priori file).
3. Correlation Processing

Execute "fx_cor" (or "fx_cor_new" for not K5/VSSP format data) with a large number of lags such as 1024 or more as follows.

```
fx_cor /home/vlbi/corrapri/apesample.txt -lag 1024
```

where “apesample.txt” is a a-priori file created by “apri_calc” (use actual a-priori file name created by the previous step). After processing, correlation amplitude and residual delay are displayed by channel as follows.

<table>
<thead>
<tr>
<th>CH#</th>
<th>FREQ(MHz)</th>
<th>MAX AMP</th>
<th>RESIDUAL DELAY (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8209.99</td>
<td>0.000651</td>
<td>-3.26108e-06</td>
</tr>
<tr>
<td>2</td>
<td>8219.99</td>
<td>0.000443</td>
<td>-3.28622e-06</td>
</tr>
<tr>
<td>3</td>
<td>8249.99</td>
<td>0.000589</td>
<td>-3.23678e-06</td>
</tr>
<tr>
<td>4</td>
<td>8309.99</td>
<td>0.000480</td>
<td>-3.25464e-06</td>
</tr>
</tbody>
</table>

COUT File is ../cout/cout0005.txt

where "COUT File" is a correlation results file. Correlation functions are also plotted as shown in Fig.7.1.

![Figure 7.1. Correlation functions displayed after “fx_cor” (or “cor”) processing.](image)

If clear peak appears in the plots of correlation function, we can determine clock offset. If it is not clear, execute the coarse-search program “sdelay” to search a clear peak as follows.

```
sdelay ../cout/cout0005.txt
```

where “../cout/cout0005.txt” is a COUT File (correlator out file) displayed at the end of “fx_cor” processing. After “sdelay” processing, summary of coarse search is displayed as follows and search functions are plotted (Fig.7.2).
COUT : ../cout/cout0005.txt
X DATA : /home/kondo/data/testspeed/Xk5data.10.dat
Y DATA : /home/kondo/data/testspeed/Yk5data.10.dat
BASELINE : KASHIM11 - TOMAKO11
SOURCE : 3C273B SAMPLING : 1 bit 8 MHz
PRT : 2003/197 02:41:10 Tinteg(s) : 9.0
LAG SIZE : 1024
CLOCK : offset 5.735e-06(s) rate 0.000e+00(s/s)
EOP : ut1-utc 0.000000(s)
: x-wobb 0.000000(asec)
: y-wobb 0.000000(asec)

CH# FREQUENCY AMP MAX POSITION RESIDUAL
(MHz) (2048x 128) Delay(usec) Rate(ps/s) SNR
------------------------------------------------------------------------
1 8209.99 U 1.192e-03 ( 973, 65) -3.251 -0.134 10.1
2 8219.99 U 8.613e-04 ( 973, 68) -3.270 2.482 7.3
3 8249.99 U 1.089e-03 ( 973, 65) -3.239 -0.089 9.2
4 8309.99 U 1.095e-03 ( 973, 66) -3.242 0.943 9.3
------------------------------------------------------------------------
Note: No amplitude correction is made.

CH# PCAL FREQ(kHz) X-Amp X-Phase Y-Amp Y-Phase
-----------------------------------------------------------------
1 10.00 0.109 70.6 0.251 111.9
2 10.00 0.107 -92.2 0.244 -162.7
3 10.00 0.106 136.2 0.247 84.3
4 10.00 0.104 -143.5 0.259 -43.2

Outfile is ./sdelayout.txt

Figure 7.2. Search functions displayed after “sdelay” processing. Peaks are not clear due to a large number of lags. In this case, zoom up using the options “-tzoom” and “-tshift”. Fig.7.3 shows search functions with the options “-tzoom 20 -tshift -3.25e-6”. 
Fringe detection is judged from summary output and plot of search functions as follows.

- peak positions shown at “POSITION” column are the same for all channels
- clear single peak appears in the plot of search function
- SNR is larger than 7

If above conditions are almost satisfied, we can say “fringe is successfully detected”.

4. Determination of clock offset

Clock offset \( c_{offset} \) is given as

\[
    c_{offset} = c_0 + \Delta \tau
\]

where \( c_0 \) is clock offset used at “apri_calc” and \( \Delta \tau \) is residual delay obtained by “s delay”.

5. Determination of clock rate

Clock rate \( c_{rate} \) is given as

\[
    c_{rate} = \frac{c_2 - c_1}{t_2 - t_1}
\]

where \( c_1 \) is clock offset obtained for a scan at the time \( t_1 \) and \( c_2 \) is clock offset obtained for a scan at the time \( t_2 \), and \( t_1 \) and \( t_2 \) are PRT (processing reference time) given by a unit of second. If a session extends on the next day, take the day difference into consideration.

6. Set up clock parameters at “apri_calc”

Clock offset, clock rate, and clock epoch are set up when executing “apri_calc” as follows.

- in case of setting up clock offset only:
  set clock offset only, use default or set 0 for rate and epoch.

- in case of setting up both clock offset and rate:
  set \( c_1 \) for clock offset, \( c_{rate} \) for clock rate, and \( t_1 \) (after converting to year, total day, hour, minute, second) for clock epoch
7.2 Processing all scans

1. Calculation of a-priori values
   Calculate a-priori values for all scans with reflecting clock parameters obtained by fringe search. An example of non-interactive execution is as follows.

   ```
   apri_calc /home/vlbi/sked/jd1606.skd -coffset -3.25e-6 -crate 1.0e-13
             -cepoch 2016/02:41:10 -baseid RY -g 2 -xdir /home/vlbi/data/R
             -ydir /home/vlbi/Y -apedir /home/vlbi/corrapri
   ```

   Conditions used in this example are as follows; clock offset is -3.25 μsec, clock rate is 1.0×10^{-13}(s/s), clock epoch is 2016/02:41:10, baseline is “RY”, data directory for X station data is “/home/vlbi/data/R” and for Y station “/home/vlbi/data/Y”, directory for a-priori files is “/home/vlbi/corrapri”. In case of using clock offset only, omit options “-crate” and “-cepoch”.

2. Generation of the list of a-priori files
   Execute linux command as follows.

   ```
   ls -l /home/vlbi/corappri/ape*RYb.txt > apelistRY.txt
   ```

   where the directory of a-priori files is “/home/vlbi/corappri” and “apelistRY.txt” is a file generated. “ape*RYb.txt” is a filter to select baseline “RY” and frequency code “b” if there are a-priori files for multiple baselines. If you want to select all a-priori files, execute like,

   ```
   ls -l /home/vlbi/corappri/ape*.txt > apelistRY.txt
   ```

   where "apelistRY.txt" can be any another name.

3. Correlation processing for all scans
   As for 1-bit sampling data, execute as follows,

   ```
   cor_all ./apelistRY.txt
   ```

   where "apelistRY.txt" is the name of list file located in the current directory. See "How to run software correlator" (section 4.2) for other options. As for multi-bit sampling data, such as 2-bit AD data, use “fx_cor_all” instead. Use “cor_all_new” or “fx_cor_all_new” for format data other than K5/VSSP format.

7.3 Correlation processing for a variety of data formats (VDIF, Mark5B, ADS, OCTAD)

    When executing “apri_calc”, data format should be specified using options “-format” or “-formX”, “-formY”. (See “How to execute apri_calc” (section 3.1) for details.) Then use “cor_new”, “fx_cor_new”, “cor_all_new”, “fx_cor_all_new” instead of “cor”, “fx_cor”, “cor_all”, “fx_cor_all”.

74
8 Update history

2016/05/29 add “cor_new” and “fx_cor_new”

2017/03/03 translated to English

2017/03/06 add two sections “6 Data format conversion” and “D VLBI data format”
A  A-priori file format

1. A-priori file structure

A-priori file consists of "section" described by letters starting with `$` and parameters followed by the section letter. Table 1 shows the list of sections and their order in an a-priori file. $FORMAT1$ and $FORMAT2$ are new sections to define data format other than K5/VSSP. Any letters after `*` in a line are treated as comments.

Table 1. List of sections

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$EXPCODE$</td>
<td>experiment title</td>
</tr>
<tr>
<td>$OBS_NUMBER$</td>
<td>scan (observation) number</td>
</tr>
<tr>
<td>$STATION1$</td>
<td>station 1 (X) information</td>
</tr>
<tr>
<td>$FORMAT1$</td>
<td>station 1 data format (new)</td>
</tr>
<tr>
<td>$XYZ-STATION1$</td>
<td>station 1 position</td>
</tr>
<tr>
<td>$STATION2$</td>
<td>station 2 (Y) information</td>
</tr>
<tr>
<td>$FORMAT2$</td>
<td>station 2 data format (new)</td>
</tr>
<tr>
<td>$XYZ-STATION2$</td>
<td>station 2 position</td>
</tr>
<tr>
<td>$BASEID$</td>
<td>baseline ID</td>
</tr>
<tr>
<td>$FRQ_GRP(1-4)$</td>
<td>frequency group</td>
</tr>
<tr>
<td>$FREQUENCY$</td>
<td>RF frequency</td>
</tr>
<tr>
<td>$PCAL_FREQ$</td>
<td>PCAL (phase calibration) frequency</td>
</tr>
<tr>
<td>$CLOCK$</td>
<td>clock parameters</td>
</tr>
<tr>
<td>$SOURCE$</td>
<td>radio source name</td>
</tr>
<tr>
<td>$RA$</td>
<td>radio source position (right ascension)</td>
</tr>
<tr>
<td>$DEC$</td>
<td>radio source position (declination)</td>
</tr>
<tr>
<td>$EPOCH$</td>
<td>epock of radio position</td>
</tr>
<tr>
<td>$GHA$</td>
<td>Greenwich hour angle of radio source</td>
</tr>
<tr>
<td>$EOP$</td>
<td>earth orientation parameters</td>
</tr>
<tr>
<td>$START$</td>
<td>scan start time (UTC)</td>
</tr>
<tr>
<td>$STOP$</td>
<td>scan stop time (UTC)</td>
</tr>
<tr>
<td>$APRIORI$</td>
<td>a-prioi values (PRT, delay, delay rate, delay 2 dots, delay 3 dots)</td>
</tr>
<tr>
<td>$END$</td>
<td>end of a-priori file</td>
</tr>
</tbody>
</table>

2. Parameters at each section

```plaintext
$EXPCODE$      section experiment code
    exp_code

$OBS_NUMBER$   section scan (observation) number
    n

$STATION1$     section station 1 (X) information
    station1_name data_file

$FORMAT1$      section station 1 data format (can be omitted for VSSP format)
    data_format [sampling_info]
    data format is VDIF|M5B|OCTAD|ADS
    where
    VDIF – VDIF format
    M5B  – Mark-5B format
    OCTAD – OCTAD format
    ADS   – ADS format
    sampling information is sampling frequency ($m$), # of channels ($n$) and
    AD resolution in bits ($k$), and described as follows.
    $m$MHz $nCH$ $kbit$

$XYZ-STATION1$ section station 1 position
    x y z

$STATION2$     section station 2 (Y) information
    station2_name data_file

$FORMAT2$      section station 2 data format (can be omitted for VSSP format)
    data_format [sampling_info]

$XYZ-STATION2$ section station 2 position
    x y z
```

76
$BASEID
baseline_id
section baseline ID
baseline ID (either 2 letters or 4 letters)

$FRQ_GRP
n
section frequency group
frequency group # (1-4) or 0
0 means all 16CH processing

$FREQUENCY
rf_freq side_band [x-ch [y-ch]]
where rf_freq – RF frequency (Hz), side_band – sideband (U|L)
x_ch – X data CH#, y_ch – Y data CH#

$PCAL_FREQ
pcal_freq
section PCAL (phase calibration) frequency
PCAL frequency (Hz)

$CLOCK
OFST= offset
RATE= rate
XCOF= x_offset
section clock parameters
clock offset (s). Positive value means Y clock tic earlier than X clock tic.
clock rate (s/s)
clock offset (sec) of X station to UTC.
Positive value means X clock tic earlier than UTC clock tic.

$SOURCE
srcnam
section radio source name
radio source name (8 letters)

$RA
hour minute sec
section radio source position (right ascension)
right ascension (hour, minute, second)

$DEC
deg minute sec
section radio source position (declination)
declination (degree, minute, second)

$EPOCH
year
section epoch of radio position
epoch (year)

$GHA
hour minute sec
section Greenwhich hour angle of radio source
hour angle (hour, minute, second)

$EOP
UT1-UTC= ut1utc
X_WOBB = wobbx
Y_WOBB = wobby
section earth orientation parameters
UT1-UTC (s)
Wobbling X (arcsec)
Wobbling Y (arcsec)

$START
yyyydddhhmmss
section scan start time (UTC)
scan start time (UTC) (yyyy – year, ddd – total day, hh – hour,
mm – minute, ss – second)

$STOP
yyyydddhhmmss
section scan stop time (UTC)
scan stop time (UTC) (year, total day, hour, minute, second)

$APRIORI
PRT=yyyydddhhmmss
TAU0= tau
TAU1= tau1
TAU2= tau2
TAU3= tau3
section a-priori values
PRT (processing reference time) (UTC) (year, total day, hour, minute, second)
a-priori delay (s)
a-priori delay rate (s/s)
a-priori delay 2-dots (s/s^2)
a-priori delay 3-dots (s/s^3)

$END
end of a-priori file
3. Examples of a-priori file

Ex.1 in case of K5/VSSP format

** This is Apriori file made by apri_calc Ver. 2016-09-29
** for cor, cor_all, fx_cor, and fx_cor_all
** SUBNET ON: PRT is set according to each scan length
** Clock parameters at run are as follows,
** Clock Offset (s) : 0.000000
** Clock Rate (s/s) : 0.000000
** Clock Epoch : 0000/000 00:00:00
**
**
$EXPCODE <--- section experiment title
KS15002

$OBS_NUMBER <--- section scan (observation) number
1 <--- scan (observation) number

$STATION1 <--- section X station information
KASHIM1 ./R0020001.dat <--- X station name and datafile name

$XYZ-STATION1 <--- section station X position
-3997505.701700 3276878.404550 3724240.703140 <--- X station position (X Y Z)(m)

$STATION2 <--- section Y station information
KOGANEI ./G0020001.dat <--- Y station name and datafile name

$XYZ-STATION2 <--- section station Y position
-3941937.479090 3368150.907990 3702235.288150 <--- X station position (X Y Z)(m)

$BASEID <--- section baseline ID
RG <--- baseline ID (2 letters or 4 letters)

$FRQ_GRP(1-4) <--- section frequency group
1 <--- frequency group # (1-4)

$FREQUENCY <--- section RF frequency
7864990000.0 U <--- RF frequency (Hz) and sideband (U|L) for CH #1
7874990000.0 U <--- RF frequency (Hz) and sideband (U|L) for CH #2
7884990000.0 U <--- RF frequency (Hz) and sideband (U|L) for CH #3
8014990000.0 U <--- RF frequency (Hz) and sideband (U|L) for CH #4

$PCAL_FREQ <--- section PCAL (phase calibration) frequency
10000.0 <--- PCAL frequency (Hz) for CH #1
10000.0 <--- PCAL frequency (Hz) for CH #2
10000.0 <--- PCAL frequency (Hz) for CH #3
10000.0 <--- PCAL frequency (Hz) for CH #4

$CLOCK <--- section clock parameters
OFST= 0.000000 <--- clock offset (s)
RATE= 0.000000 <--- clock rate (s/s)
X_WOBB = 0.000000 <--- clock offset (s) of X station to UTC

$SOURCE <--- section radio source name
3C345 <--- radio source name

$RA <--- section radio source position (right ascension)
16 42 58.80996700 <--- right ascension (hour, minute, second)

$DEC <--- section radio source position (declination)
39 48 36.99406000 <--- declination (degree, minute, second)

$EPOCH <--- section epoch of radio position
2000.0 <--- epoch (year)

$GHA <--- section Greenwich hour angle of radio source
16 3 23.584000 <--- hour angle (hour, minute, second)

$EOP <--- section earth orientation parameters
UT1-UTC= 0.000000
X_WOBB = 0.000000
Y_WOBB = 0.000000

$START <--- section scan start time (UTC)
2015002020000 <--- YYYYDDDDHHMMSS

78
$STOP  
2015002020130  

$APRIORI  
PRT=2015002020045  
TAU0= -8.744597367101878e-05  
TAU1=  7.14765473084870e-13  
TAU2=  9.254412615463209e-17  
$END

Ex.2 in case of VDIF format data

** This is Apriori file made by apri_calc Ver. 2016-09-29  
** for cor, cor_all, fx_cor, and fx_cor_all  
** SUBNET ON: PRT is set according to each scan length  
** Clock parameters at run are as follows,  
** Clock Offset (s) : 0.000000  
** Clock Rate (s/s) : 0.000000  
** Clock Epoch : 0000/000 00:00:00  
**

$EXPCODEK  
KS15002  
$OBS_NUMBER  
1  

$FORMAT1  
KASHIM1 ./R0020001.dat  

$XYZ-STATION1-3997505.70170  

$FORMAT2  
KOGANEI ./G0020001.dat  

$XYZ-STATION2-3941937.47909  

$BASEID  
RG  

$FRQ_GRP(1-4)  
0  

$FREQUENCY * RFfreq U|L <pickup ch# for station1> <pickup ch# for station2>  
7864990000.0 U  
7874990000.0 U  
8014990000.0 U  
8114990000.0 U  
8244990000.0 U  
8504990000.0 U  
8544990000.0 U  
8574990000.0 U  
2214990000.0 U  
2224990000.0 U  
2234990000.0 U  
2264990000.0 U  
2294990000.0 U  
2304990000.0 U  

$PCAL_FREQ  
10000.0  
10000.0  
10000.0  
10000.0  
10000.0  
10000.0
Ex.3 in case of Mark-5B format data

** This is Apriori file made by apri_calc Ver. 2016-09-29
** for cor, cor_all, fx_cor, and fx_cor_all
** SUBNET ON: PRT is set according to each scan length
** Clock parameters at run are as follows,
** Clock Offset (s) : 0.000000
** Clock Rate (s/s) : 0.000000
** Clock Epoch : 0000/000 00:00:00
**
$EXPCODE KS15002
$OBS_NUMBER 1
$STATION1 KASHIM11 ./R0020001.dat
$FORMAT1 M5B 16MHz 16CH 1bit <--- set Mark-5B format and sampling information
$XYZ-STATION1 -3997505.701700 3276878.404550 3724240.703140
$STATION2 KOGANEI ./G0020001.dat
$FORMAT2 M5B 16MHz 16CH 1bit <--- set Mark-5B format and sampling information
$XYZ-STATION2
-5941937.479090 3368150.907990 3702235.288150
$BASEID
RG
$FRQ_GRP(1-4)
0
$FREQUENCY
7864990000.0 U
7874990000.0 U
7884990000.0 U
8014990000.0 U
8114990000.0 U
8144990000.0 U
8544990000.0 U
8564990000.0 U
8574990000.0 U
2214990000.0 U
2224990000.0 U
2234990000.0 U
2304990000.0 U
$PCAL_FREQ
10000.0
10000.0
10000.0
10000.0
10000.0
10000.0
10000.0
10000.0
10000.0
10000.0
10000.0
10000.0
10000.0
$CLOCK
OFST= 0.000000
RATE= 0.000000
XCOF= 0.000000
$SOURCE
30345
$RA
16 42 58.80996700
$DEC
39 48 36.99406000
$EPOCH
2000.0
$CHA
16 3 23.584000
$EOP
UT1-UTC = 0.000000
X_WOBB = 0.000000
Y_WOBB = 0.000000
$START
2015002020000
$STOP
2015002020130
$APRIORI
PBT=20150020200045
TAU0= -9.744597387101878e-05
TAU1= -1.740376052034359e-08
TAU2= 7.147465473084870e-13
TAU3= 9.284412615483209e-17
$END
Ex. 4 in case of VSSP format and VDIF format data

** This is Apriori file made by apri_calc Ver. 2016-09-29
** for cor, cor_all, fx_cor, and fx_cor_all
** SUBNET ON: PRT is set according to each scan length
** Clock parameters at run are as follows,
** Clock Offset (s) : 0.000000
** Clock Rate (s/s) : 0.000000
** Clock Epoch : 0000/000 00:00:00
**
$$EXPCODE$KS15002
$$OBS_NUMBER
1
$$STATION1
KASHIM11 ./R0020001.dat
$$XYZ-STATION1
-3997505.701700 3276878.404550 3724240.703140
$$STATION2
KOGANEI ./G0020001.dat
$$FORMAT2
VDIF <--- set VDIF format for Y station (X station is default data format VSSP)
$$XYZ-STATION2
-3941937.479090 3368150.907990 3702235.288150
$$BASEID
R0
$$FRQ_GRP(1-4)
3
$$FREQUENCY
8564990000.0 U 1 9 <--- RF frequency for X station CH# and Y station CH#
8574990000.0 U 2 10
2214990000.0 U 3 11
2224990000.0 U 4 12
$$PCAL_FREQ
10000.0 10000.0 10000.0 10000.0
$$CLOCK
OFST= 0.000000
RATE= 0.000000
X_OOF= 0.000000
$$SOURCE
3C34S
$$RA
16 42 58.80996700
$$DEC
39 48 36.99406000
$$EPOCH
2000.0
$$GHA
16 3 23.584000
$$EOP
UT1-UTC= 0.000000
X_WOBB = 0.000000
Y_WOBB = 0.000000
$$START
2015002020000
$$STOP
2015002020130
**$APRIORI
PRT=2015002020045
TAU0= -8.744597367101878e-05
TAU1= -1.740376052034359e-08
TAU2= 7.147465473084870e-13
4. A-priori file for special processing

By editing parameter at $FREQUENCY$, we can change the number of processing channels, and/or channel allocation between X and Y stations. When the number of channels is changed at $FREQUENCY$, the number of channels at $PCAL_FREQ$ should be changed to keep the number of channels same.

Ex.1 change 16CH data to 6CH data and change CH# of Y station

```
$FREQUENCY
7864990000.0 U 1 6
7874990000.0 U 2 5
7884990000.0 U 3 4
8014990000.0 U 4 3
8114990000.0 U 5 2
8244990000.0 U 6 1

$PCAL_FREQ
10000.0
10000.0
10000.0
10000.0
10000.0
10000.0
```
### B  K5 software correlator output format (Format 7)

<table>
<thead>
<tr>
<th>line#</th>
<th>items</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>&quot;FORMAT7&quot; comments — &quot;FORMAT7&quot; (fixed letters) + comments (comments are program name and fringe rotation parameters) A1~A5 are inserted when filtering processing is carried out by &quot;fx_cor&quot;</td>
</tr>
<tr>
<td>A1</td>
<td>BPF parameters (following shows set M-times BPF)</td>
</tr>
<tr>
<td>A2(1)</td>
<td># flow(MHz)-fhigh(MHz) factor : 1.250000-1.450000 1.000000 (1st BPF parameters)</td>
</tr>
<tr>
<td></td>
<td>repeat M times</td>
</tr>
<tr>
<td>A2(M)</td>
<td># flow(MHz)-fhigh(MHz) factor : 1.650000-1.850000 1.000000 (last (M times) BPF parameters)</td>
</tr>
<tr>
<td>A3</td>
<td># Adopted frequency resolution (MHz) = 0.040000</td>
</tr>
<tr>
<td>A4</td>
<td># Output lag size = 2048</td>
</tr>
<tr>
<td>A5</td>
<td># FFT size for processing = 2048</td>
</tr>
</tbody>
</table>

The line# shown below is that for the case without filtering parameters.

2
<table>
<thead>
<tr>
<th>host name — host PC name</th>
</tr>
</thead>
<tbody>
<tr>
<td>scan# (starting from 1)</td>
</tr>
<tr>
<td>baseline ID</td>
</tr>
<tr>
<td>date and time of correlation processing (year totalday hour minute second month day)</td>
</tr>
<tr>
<td>X station name</td>
</tr>
<tr>
<td>X station position (x,y,z) (m)</td>
</tr>
<tr>
<td>X data file name</td>
</tr>
<tr>
<td>Y station name</td>
</tr>
<tr>
<td>Y station position (x,y,z) (m)</td>
</tr>
<tr>
<td>Y data file name</td>
</tr>
<tr>
<td>radio source name</td>
</tr>
<tr>
<td>radio source right ascension (hour minute second)</td>
</tr>
<tr>
<td>radio source declination (degree minute second)</td>
</tr>
<tr>
<td>epoch of radio source position (year)</td>
</tr>
<tr>
<td>Greenwich apparent sidereal time at PRT (processing reference time) (hour minute second)</td>
</tr>
<tr>
<td>scan start time (year totalday hour minute second)</td>
</tr>
<tr>
<td>scan stop time (year totalday hour minute second)</td>
</tr>
<tr>
<td>PRT (year totalday hour minute second)</td>
</tr>
<tr>
<td>a-priori delay $\tau$ (sec) at PRT</td>
</tr>
<tr>
<td>a-priori delay rate $\dot{\tau}$ (s/s) at PRT</td>
</tr>
<tr>
<td>a-priori delay 2-dots $\ddot{\tau}$ (s/s²) at PRT</td>
</tr>
<tr>
<td>a-priori delay 3-dots $\dddot{\tau}$ (s/s³) at PRT</td>
</tr>
<tr>
<td>clock offset (sec) and, clock error of X station (sec) (Positive value means Y clock tic earlier than X clock tic) (Positive value means X clock tic earlier than UTC clock tic)</td>
</tr>
<tr>
<td>clock rate (s/s)</td>
</tr>
<tr>
<td>UT1-UTC (sec) Wob X (arcsec) Wob Y (arcsec) — earth orientation parameters</td>
</tr>
<tr>
<td># of channels [N]</td>
</tr>
<tr>
<td>CH-1 RF frequency (Hz), PCAL frequency (Hz), sideband (1:USB, 0:LSB)</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>CH-N RF frequency (Hz), PCAL frequency (Hz), sideband (1:USB, 0:LSB)</td>
</tr>
<tr>
<td>sampling frequency (Hz)</td>
</tr>
<tr>
<td>X station AD resolution (1, 2, 4 or 8) Y station AD resolution (can be omitted)</td>
</tr>
<tr>
<td>unit integration period of PP (parameter period) (sec)</td>
</tr>
<tr>
<td>total integration period (sec)</td>
</tr>
<tr>
<td># of lags [L]</td>
</tr>
<tr>
<td>total # of PP [K]</td>
</tr>
<tr>
<td>“PP# 1” — PP#1 start of correlation result</td>
</tr>
<tr>
<td>36+N</td>
</tr>
<tr>
<td>36+N(1+L)</td>
</tr>
<tr>
<td>37+N(1+L)</td>
</tr>
<tr>
<td>38+N(1+L)</td>
</tr>
<tr>
<td>39+N(1+L)</td>
</tr>
<tr>
<td>39+N(2+L)</td>
</tr>
<tr>
<td>40+N(2+L)</td>
</tr>
<tr>
<td>40+N(3+L)</td>
</tr>
</tbody>
</table>

...
C  KSP correlator output format including extension

On the bandwidth processing using "KOMB" software package, COUT type correlation data file which is
the output of K5 software correlator are converted to KSP correlation data at first, then processed by KOMB.
The number of lags of the original KSP format is fixed to 32, however it is required to increase the number of
lags more than 32 these days. Therefore KSP format is extended to support the large number of lags.

Header portion (HD: 512 bytes) of extended format is almost same as that of the original format, but “F”
(meaning FULL) is added as the “CRSMODE”. The size of correlation counter is changed from 24 bits to 32
bits for “F” mode. Furthermore add “LAG” for lag size and “ADBIT” for AD resolution.

Correlation data (CD) of each integration period PP (parameter period) consists of correlation data of each
channel unit (UD) like the original format, but UD can be extended by 256 bytes according to the lag size.
Therefore items in UD were changed largely.

<table>
<thead>
<tr>
<th>HD (512 byes)</th>
<th>CD</th>
<th>CD</th>
<th>...</th>
</tr>
</thead>
</table>

Figure C.1. Record structure of KSP correlation data format. HD: header, CD: correlation data by PP.

<table>
<thead>
<tr>
<th>CD (correlation data by PP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>UD (1unit)</td>
</tr>
</tbody>
</table>

Figure C.2. Record structure in CD by PP. UD: unit (channel) data.

<table>
<thead>
<tr>
<th>Correlation data by UD (channel)</th>
</tr>
</thead>
<tbody>
<tr>
<td>UD#0</td>
</tr>
<tr>
<td>256 bytes</td>
</tr>
</tbody>
</table>

Figure C.3. Record structure in UD (unit data). UD#0 includes time information, and 32-lag correlation data
are contained in UD#1 and after. In case of 64-lag data, it finish in UD#2. In case of 1024-lag data, UD#
continues upto 32.
### C.1 Correlation data format: header record (HD)

#### Table 1. KSP correlation data format: header record (HD: 512bytes)

<table>
<thead>
<tr>
<th>symbol</th>
<th># of bytes</th>
<th>byte position</th>
<th>type</th>
<th>note</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXCODE</td>
<td>10</td>
<td>1</td>
<td>A10</td>
<td>experiment code (A10)</td>
</tr>
<tr>
<td>NOBS</td>
<td>2</td>
<td>11</td>
<td>I*2</td>
<td>scan #</td>
</tr>
<tr>
<td>LFILE</td>
<td>6</td>
<td>13</td>
<td>A6</td>
<td>correlation file name (A6)</td>
</tr>
<tr>
<td>LBASE</td>
<td>2</td>
<td>19</td>
<td>A2</td>
<td>baseline ID (A2)</td>
</tr>
<tr>
<td>NPP</td>
<td>2</td>
<td>21</td>
<td>I*2</td>
<td># of PPs</td>
</tr>
<tr>
<td>NPPSEC</td>
<td>2</td>
<td>23</td>
<td>I*2</td>
<td>period of PP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>unit is sec for FMTFLAG “KSP ” and “K4”</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>unit is 10 msec for FMTFLAG “KSP1”</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>unit is 1 msec for FMTFLAG “KSP2”</td>
</tr>
<tr>
<td>NKOMB</td>
<td>2</td>
<td>25</td>
<td>I*2</td>
<td># of KOMB processings</td>
</tr>
<tr>
<td>KRDATE</td>
<td>8</td>
<td>27</td>
<td>I*2</td>
<td>correlation processing date and time DIM(4) (year, total day, hour, minute)</td>
</tr>
<tr>
<td>KBFILE</td>
<td>6</td>
<td>35</td>
<td>A6</td>
<td>KOMB out file name (set by KOMB)</td>
</tr>
<tr>
<td>SRCNAM</td>
<td>8</td>
<td>41</td>
<td>A8</td>
<td>radio source name (A8)</td>
</tr>
<tr>
<td>SRCRA</td>
<td>4</td>
<td>49</td>
<td>I*2</td>
<td>DIM(2) right ascension of radio source (α) (hour, minute) J2000</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>53</td>
<td>R*8</td>
<td>right ascension of radio source (α) (second) J2000</td>
</tr>
<tr>
<td>SRCDEC</td>
<td>4</td>
<td>61</td>
<td>I*2</td>
<td>DIM(2) declination of radio source (δ) (degree, minute) J2000</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>65</td>
<td>R*8</td>
<td>declination of radio source (δ) (second) J2000</td>
</tr>
<tr>
<td>IPRT</td>
<td>10</td>
<td>73</td>
<td>I*2</td>
<td>DIM(5) PRT (processing reference time) (almost center of scan length)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(year, total day, hour, minute, second)</td>
</tr>
<tr>
<td>STATX</td>
<td>8</td>
<td>83</td>
<td>A8</td>
<td>X station name (A8)</td>
</tr>
<tr>
<td>STATY</td>
<td>8</td>
<td>91</td>
<td>A8</td>
<td>Y station name (A8)</td>
</tr>
<tr>
<td>X,XYZ</td>
<td>24</td>
<td>99</td>
<td>R*8</td>
<td>DIM(3) X station position (X , Y , Z)(m)</td>
</tr>
<tr>
<td>Y,XYZ</td>
<td>24</td>
<td>123</td>
<td>R*8</td>
<td>DIM(3) Y station position (X , Y , Z)(m)</td>
</tr>
<tr>
<td>OSTART</td>
<td>10</td>
<td>147</td>
<td>I*2</td>
<td>DIM(5) scan start time (year, total day, hour, minute, second)</td>
</tr>
<tr>
<td>OSTOP</td>
<td>10</td>
<td>157</td>
<td>I*2</td>
<td>DIM(5) scan stop time (year, total day, hour, minute, second)</td>
</tr>
<tr>
<td>SRCGHA</td>
<td>4</td>
<td>167</td>
<td>I*2</td>
<td>DIM(2) Greenwich hour angle of source at PRT (hour, minute)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Greenwich hour angle of source at PRT (second)</td>
</tr>
<tr>
<td>TSAMPL</td>
<td>4</td>
<td>179</td>
<td>R*4</td>
<td>sampling period (sec)</td>
</tr>
<tr>
<td>VBW</td>
<td>4</td>
<td>183</td>
<td>R*4</td>
<td>video bandwidth (Hz)</td>
</tr>
<tr>
<td>NCH</td>
<td>2</td>
<td>187</td>
<td>I*2</td>
<td># of channel at correlation processing</td>
</tr>
<tr>
<td>ACLKO</td>
<td>4</td>
<td>189</td>
<td>R*4</td>
<td>clock offset (sec) at PRT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>positive value menas Y clock tic earlier than X clock tic</td>
</tr>
<tr>
<td>ACLKR</td>
<td>4</td>
<td>193</td>
<td>R*4</td>
<td>clock rate difference at PRT (s/s)</td>
</tr>
<tr>
<td>DLYINX</td>
<td>4</td>
<td>197</td>
<td>R*4</td>
<td>instrumental delay difference at X band (sec)</td>
</tr>
<tr>
<td>DLYINS</td>
<td>4</td>
<td>201</td>
<td>R*4</td>
<td>instrumental delay difference at S band (sec)</td>
</tr>
<tr>
<td>AXCLKE</td>
<td>4</td>
<td>205</td>
<td>R*4</td>
<td>clock error of X station (sec) at PRT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>positive value menas X clock tic earlier than UTC clock tic</td>
</tr>
<tr>
<td>PI</td>
<td>8</td>
<td>209</td>
<td>R*8</td>
<td>π</td>
</tr>
<tr>
<td>C</td>
<td>8</td>
<td>217</td>
<td>R*8</td>
<td>light speed (m/s)</td>
</tr>
<tr>
<td>FRQTAB</td>
<td>128</td>
<td>225</td>
<td>R*8</td>
<td>DIM(16) RF frequency table(Hz)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>+VE: USB, -VE: LSB</td>
</tr>
<tr>
<td>PCALF</td>
<td>64</td>
<td>353</td>
<td>R*4</td>
<td>DIM(16) PCAL (phase calibration) frequency table (Hz)</td>
</tr>
<tr>
<td>APTAU</td>
<td>32</td>
<td>417</td>
<td>R*8</td>
<td>DIM(4) a-priori values of delay etc.</td>
</tr>
<tr>
<td>SRCH</td>
<td>2</td>
<td>449</td>
<td>I*2</td>
<td>τ(τ)/τ(τ)/(τ) = 1/τ(τ)</td>
</tr>
<tr>
<td>CMODE</td>
<td>2</td>
<td>451</td>
<td>A2</td>
<td>KSP hardware correlator mode</td>
</tr>
<tr>
<td>UINT</td>
<td>2</td>
<td>453</td>
<td>I*2</td>
<td># of lags between units in case of fringe search mode (default is 30)</td>
</tr>
<tr>
<td>CUNIT</td>
<td>2</td>
<td>455</td>
<td>I*2</td>
<td>unit # that includes 0 lag in case of fringe search mode</td>
</tr>
<tr>
<td>CRLDBL</td>
<td>8</td>
<td>457</td>
<td>R*8</td>
<td>8-byte real value set from KSP control work station (unused)</td>
</tr>
<tr>
<td>CRLNG</td>
<td>4</td>
<td>465</td>
<td>I*4</td>
<td>4-byte integer value set from KSP control work station (unused)</td>
</tr>
<tr>
<td>CRLSHT</td>
<td>2</td>
<td>469</td>
<td>I*2</td>
<td>2-byte integer value set from KSP control work station (unused)</td>
</tr>
<tr>
<td>FRGMOD</td>
<td>2</td>
<td>471</td>
<td>A2</td>
<td>fringe stopping mode “CO”: continuous, “EV”: initialize each PP</td>
</tr>
<tr>
<td>CRSMODE</td>
<td>1</td>
<td>473</td>
<td>A1</td>
<td>flag of integration counter resolution for correlation and PCAL detection</td>
</tr>
<tr>
<td>VER</td>
<td>8</td>
<td>474</td>
<td>A8</td>
<td>version of correlator ROM for hardware correlator. in case of CRSMODE=“F”, set “K5-WIDE”</td>
</tr>
<tr>
<td>JXOFST</td>
<td>4</td>
<td>483</td>
<td>I*4</td>
<td>delay offsets of X station data in case hardware correlator (in unit of sample)</td>
</tr>
<tr>
<td>JYOFST</td>
<td>4</td>
<td>487</td>
<td>I*4</td>
<td>delay offsets of Y station data in case hardware correlator (in unit of sample)</td>
</tr>
<tr>
<td>LAG</td>
<td>4</td>
<td>491</td>
<td>I*4</td>
<td># of lags in case of CRSMODE=“F”</td>
</tr>
<tr>
<td>ADBIT</td>
<td>4</td>
<td>495</td>
<td>I*4</td>
<td>AD resolution (bits)</td>
</tr>
<tr>
<td>ADBITY</td>
<td>4</td>
<td>499</td>
<td>I*4</td>
<td>AD resolution of Y station when CORTYPE is set</td>
</tr>
<tr>
<td>CORTYPE</td>
<td>2</td>
<td>503</td>
<td>A2</td>
<td>correlator type “Xi” for XF or “Fx” for FX</td>
</tr>
<tr>
<td>—</td>
<td>4</td>
<td>505</td>
<td>—</td>
<td>unused</td>
</tr>
<tr>
<td>FMTFLAG</td>
<td>4</td>
<td>509</td>
<td>A4</td>
<td>format ID. “KSP”, “K4”, “KSP1”, “KSP2”</td>
</tr>
</tbody>
</table>
C.2 Correlation data file format (in case of CRSMODE=“F”);
correlation data by unit (UD)

Format is the same as the original one in case that CRSMODE is not “F”.

Table 2. KSP correlation data format (UD#0)(1st 256 bytes)

<table>
<thead>
<tr>
<th>symbol</th>
<th># of bytes</th>
<th>byte position</th>
<th>type</th>
<th>note</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMKS</td>
<td>2</td>
<td>1</td>
<td>2BYTE</td>
<td>remarks set by correlator</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>byte #1: KSEL (K value at fringe rotation)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>byte #2: BIT#(LSB=0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7-3: channel # (1-16)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2: delete flag set by KOMB 1:delete</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1-0: unused</td>
</tr>
<tr>
<td>COFLG</td>
<td>1</td>
<td>3</td>
<td>BYTE</td>
<td>correlation flag</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>BIT#(LSB=0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7-6: sign of fringe rotation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10 ... negative rotation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>00 ... sign reversal occurred during</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>01 ... positive rotation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5: mode of fringe rotator</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 ... carried out by hardware</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0 ... carried out by software</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4: fringe stopping reference frequency</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 ... at center of baseband bandwidth</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0 ... at baseband frequency</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3: flag for fractional bit correction</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 ... carried out by hardware</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0 ... carried out by software</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2: PP parameter update flag</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 ... updated</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0 ... not updated</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1-0: unused</td>
</tr>
<tr>
<td>TWESTS</td>
<td>1</td>
<td>4</td>
<td>BYTE</td>
<td>integration status</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>BIT#(LSB=0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7: AVL integration validity flag</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 ... valid</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0 ... invalid</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6-0: unused</td>
</tr>
</tbody>
</table>

**** items below are changed largely ****

| TIMX   | 7          | 5             | 14×4bits | X station time label: YYDDHHMMSSmmm (by hexadecimal) |
|        |            |               |          | offset of X and Y time series (unit of sample)  |
|        |            |               |          | positive for Y station time is ahead  |
| TIMY   | 7          | 12            | 14×4bits | X station time label:YYDDHHMMSSmmm (by hexadecimal) |
|        |            |               |          | fringe rotator address (32 bits) at the end of PP  |
|        |            |               |          | when fringe stopping is carried out, a-priori fringe rotator address at PP.  |
| TMDIFF | 4          | 19            | I*4      | 32bits  |
| FRADD  | 4          | 23            | 32bits   | fractional part of delay in the unit of sample at PP.  |
|        |            |               |          | −32768 ~ 32767 corresponds to −0.5 ~ +0.5.  |
| IFBIT  | 2          | 27            | I*2      | versatile mode (unused)  |
| MODE   | 1          | 29            | BYTE     | BIT#(LSB=0)  |
|        |            |               |          | 7-2: unused  |
|        |            |               |          | 1: 2/1 bit mode  |
1 ... 2-bit correlation (K4 hardware correlator only)
0 ... 1-bit correlation
2: weight mode at 2-bit correlation
1 ... Weight mode
0 ... Binary binary mode

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IPP</td>
<td>2</td>
<td>30</td>
<td>I*2</td>
</tr>
<tr>
<td>PCALD</td>
<td>16</td>
<td>32</td>
<td>I*4</td>
</tr>
<tr>
<td>COUNTP</td>
<td>8</td>
<td>48</td>
<td>I*4</td>
</tr>
<tr>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

Table 3. KSP correlation data format (UD#1)(256 bytes)

<table>
<thead>
<tr>
<th>symbol</th>
<th># of bytes</th>
<th>byte position</th>
<th>type</th>
<th>note</th>
</tr>
</thead>
<tbody>
<tr>
<td>CROSP</td>
<td>4</td>
<td>1</td>
<td>I*4</td>
<td>lag #1 real part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>5</td>
<td>I*4</td>
<td>lag #2 real part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>125</td>
<td>I*4</td>
<td>lag #32 real part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>129</td>
<td>I*4</td>
<td>lag #1 imaginary part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>133</td>
<td>I*4</td>
<td>lag #2 imaginary part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>253</td>
<td>I*4</td>
<td>lag #32 imaginary part of correlation data</td>
</tr>
</tbody>
</table>

When the number of lags is larger than 32, UD records are added by 256 bytes (32 lags) as follows.

Table 4. KSP correlation data format (UD#2)(256 bytes)

<table>
<thead>
<tr>
<th>symbol</th>
<th># of bytes</th>
<th>byte position</th>
<th>type</th>
<th>note</th>
</tr>
</thead>
<tbody>
<tr>
<td>CROSP</td>
<td>4</td>
<td>1</td>
<td>I*4</td>
<td>lag #33 real part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>5</td>
<td>I*4</td>
<td>lag #34 real part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>125</td>
<td>I*4</td>
<td>lag #64 real part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>129</td>
<td>I*4</td>
<td>lag #33 imaginary part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>133</td>
<td>I*4</td>
<td>lag #34 imaginary part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>253</td>
<td>I*4</td>
<td>lag #64 imaginary part of correlation data</td>
</tr>
</tbody>
</table>
Table 5. KSP correlation data format (UD#N)(256 bytes)

<table>
<thead>
<tr>
<th>symbol</th>
<th># of bytes</th>
<th>byte position</th>
<th>type</th>
<th>note</th>
</tr>
</thead>
<tbody>
<tr>
<td>CROSP</td>
<td>4</td>
<td>1</td>
<td>I*4</td>
<td>lag #32 \times (N - 1) + 1 real part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>5</td>
<td>I*4</td>
<td>imaginary part</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>9</td>
<td>I*4</td>
<td>lag #32 \times (N - 1) + 2 real part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>13</td>
<td>I*4</td>
<td>imaginary part</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>249</td>
<td>I*4</td>
<td>lag #32 \times N64 real part of correlation data</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>253</td>
<td>I*4</td>
<td>imaginary part</td>
</tr>
</tbody>
</table>
Table 6. KSP correlation data format (UD)(256 bytes)

<table>
<thead>
<tr>
<th>symbol</th>
<th># of bytes</th>
<th>byte position</th>
<th>type</th>
<th>note</th>
</tr>
</thead>
</table>
| RMKS   | 2          | 1             | 2BYTE | remarks set by correlator
<p>|        |            |               |      | byte #1: KSEL (K value at fringe rotation) |
|        |            |               |      | byte #2: BIT#(LSB=0) |
|        |            |               |      | 7-3: channel # (1-16) |
|        |            |               |      | 2: delete flag set by KOMB 1:delete |
|        |            |               |      | 1-0: unused |
| COFLG  | 1          | 3             | BYTE | correlation flag |
|        |            |               |      | BIT#(LSB=0) |
|        |            |               |      | 7-6: sign of fringe rotation |
|        |            |               |      | 10 ... negative rotation |
|        |            |               |      | 00 ... sign reversal occurred during |
|        |            |               |      | PP |
|        |            |               |      | 01 ... positive rotation |
|        |            |               |      | 5: mode of fringe rotator |
|        |            |               |      | 1 ... carried out by hardware |
|        |            |               |      | 0 ... carried out by software |
|        |            |               |      | 4: fringe stopping reference frequency |
|        |            |               |      | 1 ... at center of baseband bandwidth |
|        |            |               |      | 0 ... at baseband frequency |
|        |            |               |      | 3: flag for fractional bit correction |
|        |            |               |      | 1 ... carried out by hardware |
|        |            |               |      | 0 ... carried out by software |
|        |            |               |      | 2: PP parameter update flag |
|        |            |               |      | 1 ... updated |
|        |            |               |      | 0 ... not updated |
| TWESTS | 1          | 4             | BYTE | integration status |
|        |            |               |      | BIT#(LSB=0) |
|        |            |               |      | 7: AVL integration validity flag |
|        |            |               |      | 1 ... valid |
|        |            |               |      | 0 ... invalid |
| CROSP  | 192        | 5             | I<em>3  | DIM(64) correlation data (reset every PP) |
|        |            |               |      | (real part 3 bytes × 32 lags, imaginary part 3 bytes × 32 lags) |
|        |            |               |      | according to CRSMODE in HD |
|        |            |               |      | “L”: take lower 24 bits from 28-bit counter, “U”: take upper 24 bits |
|        |            |               |      | “H”: take 24 bits from 32-bit counter |
| COUNTP | 8          | 197           | I</em>4  | DIM(2) # of samples for correlation |
|        |            |               |      | (real part 4 bytes, imaginary part 4 bytes) |
| PCALD  | 12         | 205           | I*3  | DIM(4) PCAL detection counter (reset every PP) |
|        |            |               |      | real part (3 bytes) for X station |
|        |            |               |      | imaginary part (3 bytes) for X station |
|        |            |               |      | real part (3 bytes) for Y station |
|        |            |               |      | imaginary part (3 bytes) for Y station |
|        |            |               |      | according to CRSMODE in HD |
|        |            |               |      | “L”: take lower 24 bits from 28-bit counter, “U”: take upper 24 bits |
|        |            |               |      | “H”: take 24 bits from 32-bit counter |</p>
<table>
<thead>
<tr>
<th>Code</th>
<th>Type</th>
<th>Length</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIMX</td>
<td>7</td>
<td>217</td>
<td>X station time label: YYDDDHHMSSmmm (by hexadecimal)</td>
</tr>
<tr>
<td>TIMY</td>
<td>7</td>
<td>224</td>
<td>X station time label: YYDDDHHMSSmmm (by hexadecimal)</td>
</tr>
<tr>
<td>TMDIFF</td>
<td>4</td>
<td>231</td>
<td>offset of X and Y time series (unit of sample)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>positive for Y station time is ahead</td>
</tr>
<tr>
<td>FRADD</td>
<td>4</td>
<td>235</td>
<td>fringe rotator address (32 bits) at the end of PP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>when fringe stopping is carried out, a-priori fringe rotator address at PP</td>
</tr>
<tr>
<td>IFBIT</td>
<td>2</td>
<td>239</td>
<td>fractional part of delay in the unit of sample at PP.</td>
</tr>
<tr>
<td>MODE</td>
<td>1</td>
<td>241</td>
<td>versatile mode (unused)</td>
</tr>
<tr>
<td>IPP</td>
<td>2</td>
<td>242</td>
<td>PP#</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>−32768 ∼ 32767 corresponds to −0.5 ∼ +0.5.</td>
</tr>
</tbody>
</table>

BIT#(LSB=0)

7-2: unused
1: 2/1 bit mode
1 ∙ ∙ ∙ 2-bit correlation (K4 hardware correlator only)
0 ∙ ∙ ∙ 1-bit correlation
2: weight mode at 2-bit correlation
1 ∙ ∙ ∙ Weight mode
0 ∙ ∙ ∙ Binary binary mode

IPP# unused
D VLBI data format

D.1 Type of data format

There are two types of VLBI data. One is that consisting of frame data (Type-1), the other is non-frame data (Type-2). The former is further divided into two types Type-1A and 1B. In case of Type-1A, a part of the data portion is substituted by the header, i.e., sampled data are lost at the header portion, while in case of Type-1B, the header is inserted in the data, so that any sampled data are lost (Fig.D.1). When VLBI technique was realized, data were only the Type-1A format (Mark-III, Mark-IV, K-3). However, Type-1B (VLBA, Mark-5B, VDIF, K5/VSSP, VSSP32/64, ADS/DBBC mode) is nowadays used commonly. Type-2 format (OCTAD, ADS/raw mode) is also used in some samplers.

![Figure D.1. Type of VLBI data structures](image)

Table D.1 shows the comparison of data formats.

<table>
<thead>
<tr>
<th>Format</th>
<th>Frame length (bytes)</th>
<th>Header length (bytes)</th>
<th>Data (bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mark-III/IV, K3</td>
<td>2500</td>
<td>20</td>
<td>2480</td>
</tr>
<tr>
<td>VLBA</td>
<td>2520</td>
<td>20</td>
<td>2500</td>
</tr>
<tr>
<td>Mark-5B</td>
<td>10016</td>
<td>16</td>
<td>10000</td>
</tr>
<tr>
<td>VDIF</td>
<td>variable</td>
<td>32</td>
<td>variable</td>
</tr>
<tr>
<td>VSSP</td>
<td>1 frame/sec</td>
<td>8</td>
<td>variable</td>
</tr>
<tr>
<td>VSSP32/64</td>
<td>1 frame/sec</td>
<td>32</td>
<td>variable</td>
</tr>
</tbody>
</table>

Among these formats, VSSP format (including K5/VSSP32, K5/VSSP64) specified NICT, Mark-5B, and VDIF are described in sub-sections below.

D.2 K5/VSSP, VSSP32 and VSSP64 format

D.2.1 Data structure

K5/VSSP, K5/VSSP32, and K5/VSSP64 sampler output is made up of multiple frames, which consist of a header and a data block (Fig.D.2). A header block of VSSP consists of 8-byte data which contain two sync blocks, time information, and sampling parameters. A header block of VSSP32 usually consists of 32-byte data, of which first 8-byte data are as same as that of VSSP excepting the second sync pattern. A rest of header block contains year, date, and user definable auxiliary field. (When VSSP64 sampler operates with a VSSP32...
mode, the header block is as same as that of VSSP32. When VSSP64 operates as its native mode, the second sync pattern differs.) A header block doesn’t affect a data block, i.e., no data are lost by the insertion of the header block.

The size of a data block can be expressed by \([\text{sampling frequency (Hz)}] \times [\# \text{ of AD bits}] \times [\# \text{ of channels}]\), however actual size is limited by the maximum data rate between a sampler and host PC. In case of VSSP, which adopts PCI-bus interface, 64 Mbits/sec is maximum data transfer speed. As for VSSP32, which adopts USB-2.0 interface, the maximum data speed is 256 Mbits/sec. As for VSSP64, which adopts USB-3.0 interface, the practical maximum data speed is 1024 Mbits/sec.

\[
\begin{array}{|c|c|c|}
\hline
\text{Header (HD)} & \text{Sampled Data (SD)} & \text{Header (HD)} \\
\hline
\vdash \cdots \vdash & 1 \text{ sec} & \vdash \cdots \vdash \\
\hline
\end{array}
\]

\[< \cdots \cdots \cdots >
\]

Figure D.2. Data structure of K5/VSSP, K5/VSSP32, or K5/VSSP64 sampler output.

### D.2.2 VSSP header format

Fig.D.3 shows VSSP header format. It consists of 8 byte data.

| bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
|     |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W0  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|     |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W1  | 2nd Sync Pattern '0x8B' | AD | SFREQ | CH | seconds from 0h UTC (17bits) |
|     |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

<table>
<thead>
<tr>
<th>Sync Pattern</th>
<th>'0xFFFFFFFF'</th>
</tr>
</thead>
<tbody>
<tr>
<td>AD: AD Index</td>
<td>0: 1 bit, 1: 2 bits, 2: 4 bits, 3: 8 bits</td>
</tr>
<tr>
<td>SFREQ: Sampling frequency index (Note: sampling frequencies larger than 32MHz MHz is for format conversion)</td>
<td></td>
</tr>
<tr>
<td>0: 40kHz, 1: 100kHz, 2: 200kHz, 3: 500kHz, 4: 1MHz, 5: 2MHz, 6: 4MHz, 7: 8MHz, 8: 16MHz, 9: 32MHz, 10: 64MHz, 11: 128MHz, 12: 256MHz, 13: 512MHz, 14: 1024MHz, 15: 2048MHz</td>
<td></td>
</tr>
<tr>
<td>CH: number of channels 0: 1 channel, 1: 4 channels</td>
<td></td>
</tr>
</tbody>
</table>

Figure D.3. VSSP header format (8 bytes)

where

- **sync pattern**: 0xFFFFFFFF
- **2nd sync pattern**: 0x8B (0x8C for VSSP32, 0x8D for VSSP64)
- **AD**: number of AD resolution bits (0: 1 bit / 1: 2 bits / 2: 4 bits / 3: 8 bits)
- **SFREQ**: index for sampling frequency (Note: sampling frequencies faster than and equal to 32 MHz are for supporting data converted from other format data) (0: 40kHz / 1: 100kHz / 2: 200kHz / 3: 500kHz 4: 1MHz / 5: 2MHz / 6: 4MHz / 7: 8MHz / 8: 16MHz 9: 32MHz / 10: 64MHz / 11: 128MHz / 12: 256MHz / 13: 512MHz / 14: 1024MHz / 15: 2048MHz)
- **CH**: number of channels used 0: 1 channel 1: 4 channels
- **time**: seconds from 00h00m00s (0~86399)

### D.2.3 VSSP32 and VSSP64 header format (general specifications)

Fig.D.4 shows the specifications of VSSP32 and VSSP64 data header. All current K5 utilities adopt 20 bytes for the size of auxiliary field, so that total size of header is 32 bytes.
### General specifications of VSSP32 and VSSP64 data header.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W0</td>
<td>Sync Pattern: 0xFFFFFFFF</td>
</tr>
<tr>
<td>W1</td>
<td>2nd Sync Pattern: 0x8C for VSSP32, 0x8D for VSSP64 mode</td>
</tr>
<tr>
<td></td>
<td>AD: AD bits Index: 0: 1 bit, 1: 2 bits, 2: 4 bits, 3: 8 bits</td>
</tr>
<tr>
<td></td>
<td>SFREQ: Index for sampling frequency: (Note: maximum is 64MHz for VSSP32, 128MHz for VSSP64).</td>
</tr>
<tr>
<td></td>
<td>sampling frequency larger than 128MHz MHz is for format conversion)</td>
</tr>
<tr>
<td></td>
<td>0: 40kHz, 1: 100kHz, 2: 200kHz, 3: 500kHz, 4: 1 MHz, 5: 2MHz, 6: 4MHz, 7: 8MHz, 8: 16MHz, 9: 32MHz,</td>
</tr>
<tr>
<td></td>
<td>10: 64MHz, 11: 128MHz, 12: 256MHz, 13: 512MHz, 14: 1024MHz, 15: 2048MHz</td>
</tr>
<tr>
<td>W2</td>
<td>major version #</td>
</tr>
<tr>
<td>W3</td>
<td>CH: number of channels: 0: 1 channel, 1: 4 channels</td>
</tr>
<tr>
<td>W4</td>
<td>EF: error flag: 1: error occurred in the just before frame</td>
</tr>
<tr>
<td>W5</td>
<td>CH: 2ch flag for VSSP64 mode: 1: 2ch, 0: follow CH flag</td>
</tr>
</tbody>
</table>

**Figure D.4.** General specifications of VSSP32 and VSSP64 data header.

where
- **sync pattern**: 0xFFFFFFFF
- **2nd sync pattern**: 0x8B (0x8C for VSSP32, 0x8D for VSSP64)
- **AD**: number of AD resolution bits
  - 0: 1 bit / 1: 2 bits / 2: 4 bits / 3: 8 bits
- **SFREQ**: index for sampling frequency
  - (Note: sampling frequencies faster than and equal to 32 MHz are for supporting data converted from other format data)
  - 0: 40kHz / 1: 100kHz / 2: 200kHz / 3: 500kHz / 4: 1MHz / 5: 2MHz / 6: 4MHz / 7: 8MHz / 8: 16MHz / 9: 32MHz / 10: 64MHz / 11: 128MHz / 12: 256MHz / 13: 512MHz / 14: 1024MHz / 15: 2048MHz
- **CH**: number of channels used: 0: 1ch 1: 4ch
- **time**: seconds from 00h00m00s (0~86399)
- **major version #**: major version # of VSSP32 control ROM
- **minor version #**: minor version # of VSSP32 control ROM
- **AUX FIELD size**: size of auxiliary field (bytes)
- **EF**: error flag (set when error occurred in a previous frame)
- **year**: lower 2 digits of year
- **total day**: day of year
- **AUX FIELD Format #**: format # of auxiliary field

User can define the portion of “AUX FIELD” freely except for the first byte of auxiliary field, which is defined as the format number. When user defines a new format, confliction with the format number, that is predefined or reserved, should be avoid.
Table D.4 summarizes predefined and reserved format numbers for auxiliary field.

### Table D.4. Predefined and reserved format numbers

<table>
<thead>
<tr>
<th>Format #</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>for test</td>
</tr>
<tr>
<td>1</td>
<td>for observation (“autoobs” output data format)</td>
</tr>
<tr>
<td>2</td>
<td>for observation (“sampling” output data format)</td>
</tr>
<tr>
<td>30~39</td>
<td>reserved (for ISAS group)</td>
</tr>
<tr>
<td>85</td>
<td>for test</td>
</tr>
<tr>
<td>170</td>
<td>for test</td>
</tr>
<tr>
<td>21</td>
<td>extended format No.1 (to support arbitrary sampling frequency and channels up to 16)</td>
</tr>
<tr>
<td>22</td>
<td>extended format No.2 (to support arbitrary sampling frequency, # of AD bits, and channels)</td>
</tr>
</tbody>
</table>

Predefined format numbers are 0, 1, 2, 85, 170, 21, and 22. Numbers 30~39 are reserved for ISAS group. Format #21 and #22 are defined to support arbitrary sampling frequency and channels up to 16 (arbitrary # of AD bits for #22). This format allows to combine multiple sampler output. It is also possible to convert data obtained by a sampler other than VSSP-type sampler to VSSP32 format data.

**Format #0**

---

![Figure D.5. VSSP32 format #0](image-url)
Format #1

This is “autoobs” output data format.

<table>
<thead>
<tr>
<th>bit</th>
<th>31</th>
<th>30</th>
<th>29</th>
<th>28</th>
<th>27</th>
<th>26</th>
<th>25</th>
<th>24</th>
<th>23</th>
<th>22</th>
<th>21</th>
<th>20</th>
<th>19</th>
<th>18</th>
<th>17</th>
<th>16</th>
<th>15</th>
<th>14</th>
<th>13</th>
<th>12</th>
<th>11</th>
<th>10</th>
<th>9</th>
<th>8</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>W0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure D.6. VSSP32 format #1

Format #2

This is “sampling” output data format.

| bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9  | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|-----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| W0  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W1  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W2  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W3  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W4  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W5  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W6  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W7  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

Figure D.7. VSSP32 format #2
### Format #85

| bit  | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| W0   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W1   | 2nd Sync Pattern: '0x8C' for VSSP64 mode | AD | SFREQ | OH | seconds from 0h UTC (17bits) |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W2   | majorversion # | minorversion # | AUX FIELD size (bytes) | (8 bits) (20) | EF ch | year (2 digits) (6 bits) | total day (9 bits) |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W3   | filler data '0x5555' |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W4   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W5   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W6   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W7   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

#### 2nd Sync Pattern
- '0x8C'
- '0x8D'

#### Figure D.8. VSSP32 format #85

### Format #170

| bit  | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| W0   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W1   | 2nd Sync Pattern: '0x8C' | AD | SFREQ | OH | seconds from 0h UTC (17bits) |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W2   | majorversion # | minorversion # | AUX FIELD size (bytes) | (8 bits) (20) | EF ch | year (2 digits) (6 bits) | total day (9 bits) |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W3   | filler data '0xAAAA' |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W4   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W5   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W6   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W7   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

#### 2nd Sync Pattern
- '0x8C'
- '0x8D'

#### Figure D.9. VSSP32 format #170
Format #21 (extended format No.1)

This format supports arbitrary sampling frequency and channels up to 16. Sampling frequency and # of channels are newly defined in the AUX field.

| bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9  | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|-----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| W0  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W1  | Sync Pattern: '0x8C' for VSSP64 mode | AD | SFREQ | *1 | --- | seconds from Oh UTC (17bits) |
| W2  | major version # | minor version # | AUX FIELD size (bytes) (8 bits) (20) | year (2 digits) (7 bits) | total day (9 bits) |
| W3  | sampling frequency (MHz: 13 bits) *2 | # of CH (2) | LPF frequency (MHz) 0 means through | AUX FIELD Format # (21) |
| W4  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W5  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W6  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W7  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

*1: this is effective only when sampling frequency in AUX field is set to 0.
*2: when 0 is set here, SFREQ becomes valid.
*3: the number of channels is expressed by 2^n where 'n' is set to this field.

Figure D.10. VSSP32 format #21

where # of channels are defined by n, then # of channels is given by 2^n. n is 0,1,2,3,4, so that maximum # of channels is 16. Sampling frequency (MHz) in the AUX field is specified by 13-bit integer.

Format #22 (extended format No.2)

This format supports arbitrary sampling frequency, arbitrary # of channels, and arbitrary # of AD bits. They are newly defined in the AUX field.

| bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9  | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|-----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| W0  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W1  | Sync Pattern: '0x8C' for VSSP64 mode | not used | seconds from Oh UTC (17bits) |
| W2  | major version # | minor version # | AUX FIELD size (bytes) (8 bits) (20) | year (2 digits) (7 bits) | total day (9 bits) |
| W3  | sampling frequency (positive: MHz, negative: kHz) *1 | LPF frequency (MHz) 0 means through | AUX FIELD Format # (22) |
| W4  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W5  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W6  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| W7  |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

*1: 16-bit signed integer

Figure D.11. VSSP32 format #22

100
D.2.5 VSSP data block

Sampled data output from the VSSP or VSSP32 sampler consists of 4-byte (32-bit) unit. Figs.D.12 and D.13 show the relation among bit position, sample number, and channel number in a 32-bit unit.

![Figure D.12. Data block format for 1-ch mode.](image)

![Figure D.13. Data format for 4-ch mode](image)

D.2.6 Data block format for Format #21 and #22 (extended format)

Format #21 (extended format) supports 2, 8, and 16-ch mode besides 1 and 4-ch mode as shown in Figs.D.14, D.15, and D.16. As for 8-ch mode and 8-bit AD, 8 bytes (64 bits) are required for 1 sample data at a time. As for 16-ch mode, 8 bytes (64 bits) are required for 4-bit AD and 16 bytes (128 bits) for 8-bit AD for 1 sample data.

Format #22 (extended format) supports arbitrary sampling frequency, arbitrary # of channels, and arbitrary # of AD bits.

![Figure D.14. Extended data format for 2-ch mode.](image)

![Figure D.15. Extended data format for 8-ch mode.](image)
Figure D.16. Extended data format for 16-ch mode.

The size of data block should be an integer multiple of a 4-byte (32-bit) unit. In case when a fractional figure occurs, a rest of data block is filled by ‘0’. How to define the size of data block is as follows. The size of data sampled in 1 sec can be calculated as

\[ B = F \cdot A \cdot N \] [bits]

where \( F \), \( A \), and \( N \) are sampling frequency (Hz), the number of AD bits, and the number of channels, respectively. If \( B/32 \) has a fractional part (not an integer), the size of data block is defined as follows,

\[ S = (\text{int}(B/32) + 1) \times 32 \] [bits]

when \( \text{int} \) is a function which returns the integer part of a specified number. Then a rest of data block starting from \( B + 1 \) bits is filled out by ‘0’. If \( B/32 \) is an integer, \( S \) is set to be \( B \).

D.3 Mark5B data format

D.3.1 Data structure

Mark5B data consist of multiple frames. A frame of data consists of a header (16 bytes) and a data block (10000 bytes).

<table>
<thead>
<tr>
<th>Header (16 bytes)</th>
<th>Data block (10000 bytes)</th>
<th>Header (16 bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; ------------ frame ------------ &gt;</td>
<td>&lt; ------------ frame ------------ &gt;</td>
<td></td>
</tr>
</tbody>
</table>

Figure D.17. Data structure of Mark5B format data.

D.3.2 Mark5B header format

Fig.D.18 shows Mark5B header format (16 bytes).

D.3.3 Mark5B data block

Data block of Mark5B format consists of 10000-byte data, and it is shown as figures below by using the unit of “bit-stream”, where bit-stream is the number of bits representing sampled data at a time. There are 6 kinds of bit-stream, such as, 2, 4, 8, 16, and 32. It is expressed as bit-stream= nk where \( n \) is the number of channels and \( k \) is the number of AD bits.
Figure D.18. Mark5B header format

Figure D.19. Mark5B data block for 1 bit-stream data. The number in a cell denotes sampling # that lines up in order of a sampling. This format corresponds to 1CH×1-bit AD data.

Figure D.20. Mark5B data block for 2 bit-stream data. This format corresponds to 1CH×2-bit AD or 2CH×1-bit AD data.

Figure D.21. Mark5B data block for 4 bit-stream data. This format corresponds to 4CH×1-bit AD, 2CH×2-bit AD data, and so on.

Figure D.22. Mark5B data block for 8 bit-stream data. This format corresponds to 8CH×1-bit AD, 4CH×2-bit AD data, and so on.

Figure D.23. Mark5B data block for 16 bit-stream data. This format corresponds to 16CH×1-bit AD, 8CH×2-bit AD data, and so on.

Figure D.24. Mark5B data block for 32 bit-stream data. This format corresponds to 16CH×2-bit AD and so on.
D.4 VDIF data format

D.4.1 Data structure

VDIF (VLBI Data Interchange Format) data consist of stream of data frames. Each frame contains a header and a data block. The header consists of 32 bytes; first 16-byte data contain time information, etc., and rest is an extended user block (16 bytes). At present time, two formats (EDV #1 and #2) are defined for the extended user block.

```
<----------------- frame ---------------->  <-----------------                ------------>
```

![Figure D.25. Data structure of VDIF format data.](image)

D.4.2 VDI header format

Common specifications

Fig. D.26 shows VDIF header format (32 bytes).

```
| bit 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| W0    | I  | L  |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W1    |     |    |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W2    |     |    |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W3    |     |    |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W4    |     |    |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W5    |     |    |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W6    |     |    |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W7    |     |    |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
```

- **W0**: I = Invalid Flag; L = Legacy mode: 0: standard 32-byte VDIF header, 1: legacy header-length (16-byte) mode
- **W1**: Un-assigned
- **W2**: Version Number
- **W3**: bits/sample-1
- **W4**: Extended User Data Version Number
- **W5**: Extended User Data
- **W6**: Extended User Data
- **W7**: Extended User Data

![Figure D.26. VDIF header format](image)

EDV#1 specified by NICT

Fig. D.27 shows the extended header format EDV#1 specified by NICT.

EDV#3 for VLBA, VLA, GBT

Fig. D.28 shows the extended header format EDV#3 for VLBA, VLA, GBT.

D.4.3 VDIF data block

Examples of single channel data
| Bit 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9  | 8  | 7  | 6  | 5  | 4  | 3  | 2  | 1  | 0  |
|-------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| W0    | I  | L  |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W1    | Un-assign | Ref Epoch: Half year counter |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W2    | Version Number | log(#chns) |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W3    | C |     | bits/sample-1 |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W4    | Extended User Data Version Number | U |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W5    |     | Sync Pattern: '0xACABFEED' |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W6    |     | DAS/Station Name (8 bytes) |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| W7    |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |

U: units of sampling rate 0: kHz, 1: MHz

Figure D.27. VDIF header format (EDV#1 specified by NICT)

Figure D.29. VDIF data block for 1CH×1-bit AD data. The number in a cell denotes sampling # that lines up in order of a sampling.

Figure D.30. VDIF data block for 1CH×2-bit AD data.

Figure D.31. VDIF data block for complex 1CH×1-bit AD data.

Figure D.32. VDIF data block for complex 1CH×2-bit AD data.

Examples of multiple channel data

Figure D.33. VDIF data block for 16CH×1-bit AD data.
### Table D.28: DIF header format (EDV#3 for VLBA, VLA, GBT)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>Ref Epoch: Number of seconds from reference epoch</td>
</tr>
<tr>
<td>30</td>
<td>Ref Epoch: Half year counter</td>
</tr>
<tr>
<td>29</td>
<td>Data Frame # within second</td>
</tr>
<tr>
<td>28</td>
<td>Data Frame length (units of bytes)</td>
</tr>
<tr>
<td>27</td>
<td>Thread ID for multi-stream (0–1023)</td>
</tr>
<tr>
<td>26</td>
<td>Station ID</td>
</tr>
<tr>
<td>25</td>
<td>Sampling Rate (kHz or MHz)</td>
</tr>
<tr>
<td>24</td>
<td>Sync Pattern: '0xACABFEED'</td>
</tr>
<tr>
<td>23</td>
<td>Extended User Data Version Number</td>
</tr>
<tr>
<td>22</td>
<td>Personality Type: x00: RFBG, x80: DDC Mark5B, x81: DDC complex, x82: DDC VDIF</td>
</tr>
</tbody>
</table>

**U:** units of sampling rate 0: kHz, 1: MHz  
**ESB:** Electronic Side Band (1: upper, 0: lower)  
**Personality Type:** x00: RFBG, x80: DDC Mark5B, x81: DDC complex, x82: DDC VDIF

#### Figure D.28. DIF header format (EDV#3 for VLBA, VLA, GBT)

![Diagram](image1)

**Figure D.34. VDIF data block for 16CH×2-bit AD data.**

<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>Complex 16ch X 2bits/sample</td>
</tr>
<tr>
<td>30</td>
<td>CH8Q</td>
</tr>
<tr>
<td>29</td>
<td>CH16Q</td>
</tr>
<tr>
<td>28</td>
<td>CH8Q</td>
</tr>
<tr>
<td>27</td>
<td>CH16Q</td>
</tr>
<tr>
<td>26</td>
<td>CH8Q</td>
</tr>
<tr>
<td>25</td>
<td>CH16Q</td>
</tr>
<tr>
<td>24</td>
<td>CH8Q</td>
</tr>
<tr>
<td>23</td>
<td>CH16Q</td>
</tr>
</tbody>
</table>

**Figure D.35. VDIF data block for complex 16CH×2-bit AD data.**
E How to install PGPLOT

E.1 by “apt-get” (for Ubuntu 14.04)

If software package is available, we can install PGPLOT by using “apt-get” command as

sudo apt-get install pgplot5

E.2 step by step

1. download
   
download pgplot5.2.tar.gz from ftp://ftp.astro.caltech.edu/pub/pgplot/pgplot5.2.tar.gz
   copy pgplot5.2.tar.gz to /usr/local/src

2. decompress
   
cd /usr/local/src
   gunzip -c pgplot5.2.tar.gz | tar xvf -
   or
   tar xvfz pgplot5.2.tar.gz
   /usr/local/src/pgplot and subdirectories are generated by decompression

3. create the install directory
   
   mkdir /usr/local/pgplot

4. select driver
   
cd /usr/local/pgplot
   cp /usr/local/src/pgplot/drivers.list .
   remove “!” at driver name
   vi drivers.list
   drivers recommended are
   /NULL /PS /VPS /CPS /VCPS /XWINDOW /XSERVE

5. create the makefile
   
cd /usr/local/pgplot
   /usr/local/src/pgplot/makemake /usr/local/src/pgplot linux g77 gcc

6. execute “make”
   
   make
   make cpg
   make clean

   now there are files as follows in the directory
   cpgdemo grexec.f libcpgplot.a pgdemo1 pgxwin_server cpplot.h grfont.dat libpgplot.a pgplot.doc rgb.txt
   drivers.list grpckg1.inc makefile pgplot.inc
   libpgplot.so

7. copy libraries and include file to standard directory
   
   cp libcpgplot.a /usr/lib
   cp libpgplot.a /usr/lib
   cp cpplot.h /usr/include
   cp libpgplot.so /usr/lib

8. execution of demonstration programs
   
   set environmental variables as follows
   
   csh : setenv PGPLOT_DIR /usr/local/pgplot/
   sh : PGPLOT_DIR="/usr/local/pgplot/"; export PGPLOT_DIR
   or
   export PGPLOT_DIR=/usr/local/pgplot/
   As for /XWINDOW or /XSERVE, set as follows
   
   csh : setenv DISPLAY IP_address(or hostname):0.0
execute demo programs
   pgdemo1 (FORTRAN demo)
   or
   cpdgdemo (C demo)

9. How to compile C program
   compile by either Method 1 or Method 2 as follows, where test source code is test.c

   Method 1
   f77 -o test test.c -lcpgplot -lpgplot -L/usr/X11R6/lib -lX11 -lm
   (Note: use Fortran compiler)

   Method 2
   cc -O2 -c -I. test.c
   f77 -o test test.o -lcpgplot -lpgplot -L/usr/X11R6/lib -lX11 -lm
F  How to install FFTW

1. download
   get the latest version of FFTW from http://www.fftw.org/
   (latest version is ftp://ftp.fftw.org/pub/fftw/fftw-3.3.6-pl1.tar.gz as of March 2, 2017)
   copy fftw-3.3.6-pl1.tar.gz to /usr/local/src

2. decompress
   cd /usr/local/src
   tar xvfz fftw-3.X.X.tar.gz

   /usr/local/src/fftw-3.X.X and subdirectories are generated by decompression

3. generate the makefile
   cd /usr/local/src/fftw-3.X.X
   ./configure

4. execute “make”

   make

   /usr/local/src/fftw-3.X.X/.libs will be created

5. set up library and include file to the standard directory

   make install

   by this execution, libfftw3.a and libfftw3.la are copied to /usr/local/lib
   fftw3.h is copied to /usr/local/include

6. How to compile C program

   use “-lfftw3 -lm” at linkage
   and set include file directory and library directory as follows
   -I/usr/local/include
   -L/usr/local/lib

   see http://www.fftw.org/ for details