K5 software correlator users manual

Hiroshi Imai[*1], Yasuhiro Koyama[*2], and Tetsuro Kondo[*2]

[*1] Joint Institute for VLBI in Europe, Kagoshima University
[*2] Kashima Space Research Center, NICT, Japan
(28 April 2005)

0. Introduction

This document is the users manual describing how to use the K5 Software Correlator package and how to apply it to the fringe detection and clock parameter search in VLBI observations. This package has been developed and maintained by the Radio Astronomy Applications Group in Kashima Space Research Center (KSRC), National Institute of Information and Communications Technology (NICT), Japan. After agreement with the NICT, a user institute is issued a payment-free license and is accepted to use this package only in the academic research.

Joint Institute for VLBI in Europe (JIVE) obtained the license in July 2003 after collaborating with KSRC in developing the software package in order to customize it for usage in international VLBI observations including European VLBI Network (EVN) observations. The first success in fringe detection with this package was made in March 2003 on basis of the EVN-NICT test observation using the Mark-V and K-5 backends (observation code: N03L1, figure 1). The package consists of parts reading a schedule file that is either SKED DRUDGE or SCHED VEX formatted, converting the Mark-V data format into the K-5 format, making data correlation, and displaying fringe-search results. Although the original users manual is written mainly in Japanese, this document interprets the original manual into English for international usage based on actual experiences in using the package at JIVE. The manual should be put in the following web site, http://www2.nict.go.jp/ka/radioastro/IPVLBI/m5k5_e.html, and will be maintained by either KSRC or JIVE members corresponding for any updating the package.

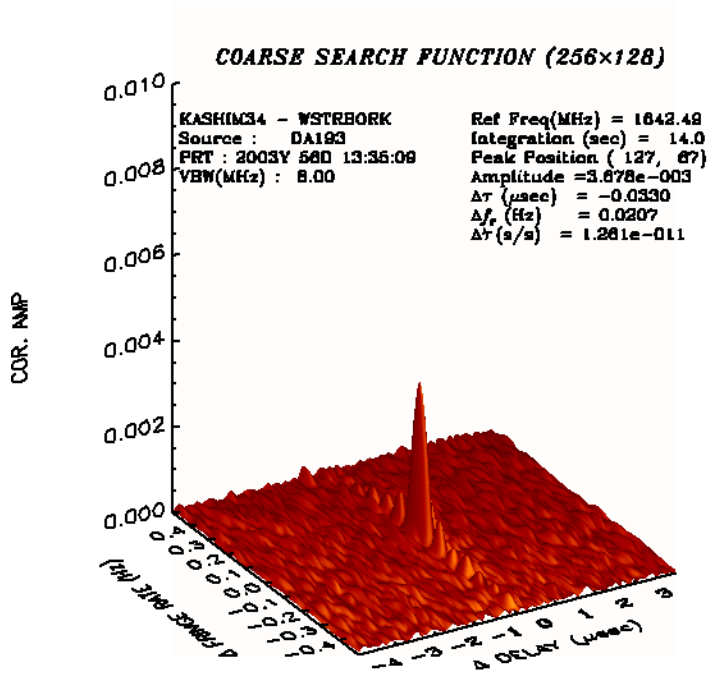

Figure 1: Fringe detected from the baseline between Kashima 34m station and WSRT using the software correlator. The observation was made on March 2003 (N03L1). The

detail of the observation operation and the data correlation is described in the web page, http://www.evlbi.org/tog/softw_corr/Mk5K5_fringe.html.

1. Installation of the package

1.1. Necessary environment

Linux
PGPLOT (cpgplot.h should be valid)

1.2. Unpacking

Make a directory for the package, move to the directory, then extract the package as follows.

gunzip -c ipvlbi_corYYYYMMDD.tar.gz [RETURN]
tar xvf ipvlbi_corYYYYMMDD.tar [RETURN]
or
tar xvzf ipvlbi_ corYYYYMMDD.tar.gz [RETURN]

Here YYYY, MM, DD are the date (year, month, day respectively) when the package was released.

1.3. Set environments in "makefile" and "Makefile"

src/makefile, mark5/makefile, ./makefile,
apri/Makefile, corr/Makefile, network/Makefile, sdelay/Makefile
for cpgplot.h: add the directory containing the library as a following example,
CFLAGS = -I../src -I/asp3/pgplot ,
here the directory /asp3/pgplot contains cpgplot.h .

1.4. Start the installer

Type "source install_cor.sh".

The shell script for JIVE environment at first copies the makefiles in the directory $CORHOME/jive.makefiles, where $CORHOME is the directory in which the package is installed. It compiles the necessary modules.

1.5. Making an archived package

Type "source archive_cor" .

1.6. Environment variables

User-defined environment variables are accepted in the following modules.

apri/apri_calc, corr/fx_cor, corr/fx_cor_all, sdelay/sdelay, mark5/m5tok5Rnp

The modules themselves define some environment variables. By typing as follows,

<<< module name>>> env

the defined variables and current values are displayed.

example: typing

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
((null)), program deflt is (/XTERM)

2. Contents and items essential for the software correlation

$CORHOME: home directory of the software correlator package
$CORHOME/ipvlbi/apri/apri_calc: program for calculating a-priori delays.
$CORHOME/ipvlbi/corr/fx_cor: program for fringe search.
$CORHOME/ipvlbi/corr/fx_cor_all: program for full data correlation.
$CORHOME/ipvlbi/sdelay/sdelay: program for delay-rate estimation and display.

3. Structure of the working space

```
$CORHOME/ipvlbi/
    |
    +-- archive_cor ..... shell script for creating archived program package
    +--- ipvlbi_cor[yyyymmdd].tar.gz .....program package archived by archive_cor
    |          ((yyyy, mm, dd indicate the release date (year, month and day)
    |
    +--- readme_cor.txt ..... users manual for the package (in Japanese)
    +-- pgplot_install.txt ..... instruction for installing PGPLOT (in Japanese)
    +-- install_cor.sh ..... shell script for install the package
    +-- nz0294.skd, nz0295.skd ..... example of a schedule file (SCHED output)
    |
    +- mark5/..... directory for analyzing VEX files and
    |    |              converting from Mark-5 format to K-5 format
    |    +-- m5tok5 ..... Mark-5 - K-5 converter
    |    |          (for 1-bit sampling in NICT-USA projects)
    |    +-- m5tok5R .... Mark-5 - K-5 converter
    |    |                                (compatible to EVN Mark-5, 2bits/sample data)
    |    +-- m5tok5np ..... Mark-5 - K-5 converter
    |    |     (for data with no parity bit, or Mark-5A mode)
    |    +-- m5tok5Rnp .... Mark-5 - K-5 converter
    |    |  (compatible to EVN Mark-5 data with no parity bit, or Mark-5A mode,
    |    |    2bits/sample data))
    |    +-- m5check .... Checking the Mark-5 format data
    |    +-- vex_ana_sample ..... checking the information in a VEX file
    |    +-- *.c, *.h ..... utility programs and include files
    |    +-- Makefile ..... makefile for the modules
    |    +-- cp2bin.sh ..... shell script for copying execution files
    |                              to the "bin" directory ../bin
    +- apri/ --
    |    +-- apri_calc
    |    +-- *.c, *.h ..... utility programs and include files
    |    +-- Makefile ..... makefile for the modules
    |    +-- cp2bin.sh ..... shell script for copying execution files
    |                              to the "bin" directory ../bin
    +- corr/ --
    |    +-- fx_cor
```

```
|       +-- fx_cor_all
|       +-- corrapri/    ..... directory for "apri_calc" outputs
|       |         + apeDDDNNNNXYG.txt    .....    a-priori parameter file
|       +-- *.c, *.h ..... utility programs and include files
|       +-- Makefile ..... makefile for the modules
|       +-- cp2bin.sh ..... shell script for copying execution files
|                                    to the "bin" directory ../bin
+-- cout/ --    ..... directory for "fx_cor" and fx_cor_all" outputs
|     |
|     +dumy.txt ..... dummy for creating this directory
|     |           (unnecessary unless backup files should be created by archive_cor.)
|     +-- coutNNNN.txt
+- sdelay/ --
|       +-- sdelay
|       +-- *.c, *.h ..... utility programs and include files
|       +-- Makefile ..... makefile for the modules
|       +-- cp2bin.sh ..... shell script for copying execution files
|                                    to the "bin" directory ../bin
+-- src/ ..... directory containing utility source programs
|       +-- *.c, *.h ..... utility programs and include files
|       +-- Makefile ..... makefile for the modules, but only for machines with the
|       |                           sampler board installed
|       +-- Makefile2 ..... makefile for the modules for machines without
|       |                             sampler board installed
|       +-- cp2bin.sh ..... shell script for copying execution files
|       |                           to the "bin" directory ../bin
|       +cp2bin2.sh ..... shell script for copying execution files
|       |                           to the "bin" directory ../bin, but for machines without
|       |                             sampler board installed.
|       +sample.skd ..... sample schedule file
```

4. Preparation for data correlation

In this manual, one base band channel (BBC) corresponds to a base band filter, supplying one of IF channels on of left- or right-circular polarization. Each of BBCs is specified in each of lines in the section $FREQ in VEX and FRUDGE files.

In the correlation process, you need not only Mark-5 and/or K-5 formatted data but also a schedule file that has either VEX or DRUDGE format. It is convenient to put them in the same directory you created. As shown in Sect. 1.7, you can set several environment variables to set several short cuts to, e.g. your data directory.

5. Data conversion from Mark-5 to K-5 format

5.1. Series of conversion modules

The module m5tok5R is used (m5tok5 is used for NICT-USA experiments having 1-bit sampling).

5.1.1. m5check: Sync-pattern check in a Mark-5 formatted file.

usage:

m5check m5file [mode]

where

m5file: filename of the Mark-5 formatted file.

mode: data format mode

0 (default): 32bit-word (32 tracks), NRZM, with parity.

1: 16bit-word(16 tracks), NRZL, without parity.

Note: You can run in mode=1 at first, then you can recognize the correct mode by finding the

sync-pattern (if you find it, mode=1 is correct).

Note: Mark-5 format is NRZM (non return to zero-mark) encoded, while K-5 format is NRZL (non return to zero-level) encoded. Mark-5A and K-5 formats have no parity bit.

example:

```
------------------------------------------------------------
> m5check G:/ipvlbi/data/evlbi3/288-173
**************************************************
*       Mark-5 data structure analysis                   *
*          Ver 1.10      2003-06-27    by    T.KONDO/NICT        *
*                                                                 *
*         NRZM, with parity, 32bit-word mode                *
**************************************************

Mark 5 Data File : G:/ipvlbi/data/evlbi3/288-1730
BIT#    HEADER    AUX    SYNC      TIME FIELD    CRC    Y/DDD HH:MM:SS.SSS FMHz
   00    --- sync not detected ---
   01    --- sync not detected ---
   02 00310031 04010007 ffffffff 22881730 00110b28    2/288 17:30:00.110    4.0
   03    --- sync not detected ---
   04 00310031 06030007 ffffffff 22881730 001106e6    2/288 17:30:00.110    4.0
   05    --- sync not detected ---
   06 00310031 08050007 ffffffff 22881730 0011093f    2/288 17:30:00.110    4.0
   07    --- sync not detected ---
   08 00310031 10070007 ffffffff 22881730 0011061c    2/288 17:30:00.110    4.0
   09    --- sync not detected ---
   10 00310031 12090007 ffffffff 22881730 00110e21    2/288 17:30:00.110    4.0
   11    --- sync not detected ---
   12 00310031 140b0007 ffffffff 22881730 0011032d    2/288 17:30:00.110    4.0
   13    --- sync not detected ---
   14 00310031 160d0007 ffffffff 22881730 00110db2    2/288 17:30:00.110    4.0
   15    --- sync not detected ---
   16 00310031 18000007 ffffffff 22881730 001103b6    2/288 17:30:00.110    4.0
   17    --- sync not detected ---
   18 00310031 20020007 ffffffff 22881730 00110a85    2/288 17:30:00.110    4.0
   19    --- sync not detected ---
   20 00310031 22040007 ffffffff 22881730 0011041a    2/288 17:30:00.110    4.0
   21    --- sync not detected ---
   22 00310031 24060007 ffffffff 22881730 00110916    2/288 17:30:00.110    4.0
   23    --- sync not detected ---
   24 00310031 26080007 ffffffff 22881730 0011012b    2/288 17:30:00.110    4.0
   25    --- sync not detected ---
   26 00310031 280a0007 ffffffff 22881730 00110da3    2/288 17:30:00.110    4.0
   27    --- sync not detected ---
   28 00310031 300c0007 ffffffff 22881730 001101d1    2/288 17:30:00.110    4.0
   29    --- sync not detected ---
   30    --- sync not detected ---
   31    --- sync not detected ---

1st header time (Y/DDD HH:MM:SS.SSS) : 2/288 17:30:00.110

bit pos    00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15
Validity    0    0    1    0    1    0    1    0    1    0    1    0    1    0    1    0
Track          0    0    4    0    6    0    8    0 10    0 12    0 14    0 16    0

bit pos    16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Validity    1    0    1    0    1    0    1    0    1    0    1    0    1    0    0    0
```

```
Track       18    0 20    0 22    0 24    0 26    0 28    0 30    0    0    0
>
------------------------------------------------------------
```

5.1.2. Data conversion in m5tok5

Note: Read this sub-section to process with m5tok5R, m5tok5np and m5tok5Rnp.

m5tok5 is customized for 1-bit sampling format data to improve the speed for format conversion. This module accepts only the Mark-5 data having 32 tracks, NRZM and parity.

```
usage (part 1):
m5tok5 mk5_filename [options]
   where
      mk5_filename:    filename of the Mark-5 formatted data.
      It should be specified before any option described below.
      options:
         -c [channel]:    BBC channel number processed [1-16].
               Specified when only one BBC channel is processed.
                     Without specifying, all of 16 BBC channels are processed.
         -g [group]:    BBC group number processed [1-4].
               Each of the groups consists of 4 BBC channels. This option is
                     ignored if -c option is specified. If both of -c and -g options are
                     skipped, all of 16 BBC channels are processed.
         -i [info_file]:    filename of the conversion information file.
              The information file is created based on the information in the schedule file
              and consists of,
                the table of correspondence between tracks and BBC channels,
                the table of correspondence between bit positions and BBC channels,
                the table of correspondence between groups and BBC channels.
                Unless the information file is specified, the default file (m5tok5info.txt)
                   is selected.
            Note 1: By adding a minus "-" in front of the filename without blanks,
                                                                  a new information file is created.
            Note 2: "-i make" automatically creates the file named "m5tok5info.txt".
               To do so, a VEX file should be provided.
         -o [k5name]:    filename of the output K-5 formatted file.
               If skipped, the filename is automatically determined
               (see "The rule of given filenames for K-5 formatted file" described below).

         -d [k5dir]:    directory in which the output K-5 formatted file is created.
               If skipped, the directory is set to the same as that in which
               the used Mark-5 formatted data is contained.
         -s [start_set]: time in seconds for processing from the time at the
             beginning of the file. The default is set to zero second.
         -p [period]: time rage of data to be processed in seconds.
               The default is to process the whole time range of the scan in the file.

usage (part 2):
   m5tok5 mk5_filename k5_filename bit1 [bit2 bit3 bit4]
      where
         mk5_filename: filename of the Mark-5 formatted data.
         k5_filename: filename of the K-5 formatted data.
         bit1, bit2, bit3, bit4:
            bit position picked up for processing [0-31] in the Mark-5 formatted data.
            If bit2, bit3 and bit4 are skipped, 1-ch mode is assumed.

            Note: Each pair of correspondences between the bit positions and
```

BBC channels can be obtained by the method described in sub-section 5.2.
EVN users do not need to concern about them.

The rule of given filenames for K-5 formatted file:
For example an original Mark-5 formatted file named "MK5" leads to make the following filenames according to the selected BBC and group number.
in 4-ch mode:
MK5.k5a ---- group #1 in info_file.
MK5.k5b ---- group #2 in info_file.
MK5.k5c ---- group #3 in info_file
MK5.k5d ---- group #4 in info_file
in 1-ch mode:
MK5.k5-NN ---- NN corresponds to one BBC ch# (1-16).

Example: An EVN data with 8 BBCs, group #1 consists of BBC1, 2, 3, 4, while group #2 consists of BBC5, 6, 7, 8.

example 1: making an info_file file.

```
------------------------------------------------
> m5tok5 G:/ipvlbi/data/evlbi3/288-1730 -i make
mk5tok5R running under Information File create mode
info file (m5tok5info.txt) will be created (updated).
Enter VEX file name -> G:/ipvlbi/data/evlbi3/2994.ovex
SITES (Station ID) defined are
   1 -- Kb
   2 -- Wf
Select Station by number      ----> 2

Detailed site information
     site definition :    WESTFORD
     site name             :    WESTFORD
     site ID               :    Wf
     site position      : 1492206.600000 -4458130.507000 4296015.532000
     site clock
          validity epoch : 2002 288 0 0 0
          clock epoch        : 2002 288 0 37 0
          clock offset      : 8.500000e-006
          clock rate         : 0.000000e+000
mode is 1230188

TRACK and FREQUENCY information for WESTFORD
     BARREL ROLL : off
     FREQDEF = CDP-SX-SX01      TRACKDEF = C1-SX01
     adbit= 1 sample_rate= 4000000.000000

     bb Tr      AD fo    chan          RF(Hz)               S          VBW(Hz)
      1 18    sign 1 &Ch01 8210990000.000000 U 2000000.000000
      2    4    sign 1 &Ch02 8210990000.000000 U 2000000.000000
      3 20    sign 1 &Ch03 8210990000.000000 U 2000000.000000
      4    6    sign 1 &Ch04 8210990000.000000 U 2000000.000000
      5 22    sign 1 &Ch05 8210990000.000000 U 2000000.000000
      6    8    sign 1 &Ch06 8210990000.000000 U 2000000.000000
      7 24    sign 1 &Ch07 8210990000.000000 U 2000000.000000
      8 10    sign 1 &Ch08 8210990000.000000 U 2000000.000000
      9 26    sign 1 &Ch09 8210990000.000000 U 2000000.000000
     10 12    sign 1 &Ch10 8210990000.000000 U 2000000.000000
     11 28    sign 1 &Ch11 8210990000.000000 U 2000000.000000
```

```
     12 14    sign 1 &Ch12 8210990000.000000 U 2000000.000000
     13 30    sign 1 &Ch13 8210990000.000000 U 2000000.000000
     14 16    sign 1 &Ch14 8210990000.000000 U 2000000.000000
Mark 5 Data File : G:/ipvlbi/data/evlbi3/288-1730
BIT#    HEADER      AUX      SYNC      TIME FIELD    CRC    Y/DDD HH:MM:SS.SSS FMHz
   00    --- sync not detected ---
   01    --- sync not detected ---
   02 00310031 04010007 ffffffff 22881730 00110b28    2/288 17:30:00.110    4.0
   03    --- sync not detected ---
   04 00310031 06030007 ffffffff 22881730 001106e6    2/288 17:30:00.110    4.0
   05    --- sync not detected ---
   06 00310031 08050007 ffffffff 22881730 0011093f    2/288 17:30:00.110    4.0
   07    --- sync not detected ---
   08 00310031 10070007 ffffffff 22881730 0011061c    2/288 17:30:00.110    4.0
   09    --- sync not detected ---
   10 00310031 12090007 ffffffff 22881730 00110e21    2/288 17:30:00.110    4.0
   11    --- sync not detected ---
   12 00310031 140b0007 ffffffff 22881730 0011032d    2/288 17:30:00.110    4.0
   13    --- sync not detected ---
   14 00310031 160d0007 ffffffff 22881730 00110db2    2/288 17:30:00.110    4.0
   15    --- sync not detected ---
   16 00310031 18000007 ffffffff 22881730 001103b6    2/288 17:30:00.110    4.0
   17    --- sync not detected ---
   18 00310031 20020007 ffffffff 22881730 00110a85    2/288 17:30:00.110    4.0
   19    --- sync not detected ---
   20 00310031 22040007 ffffffff 22881730 0011041a    2/288 17:30:00.110    4.0
   21    --- sync not detected ---
   22 00310031 24060007 ffffffff 22881730 00110916    2/288 17:30:00.110    4.0
   23    --- sync not detected ---
   24 00310031 26080007 ffffffff 22881730 0011012b    2/288 17:30:00.110    4.0
   25    --- sync not detected ---
   26 00310031 280a0007 ffffffff 22881730 00110da3    2/288 17:30:00.110    4.0
   27    --- sync not detected ---
   28 00310031 300c0007 ffffffff 22881730 001101d1    2/288 17:30:00.110    4.0
   29    --- sync not detected ---
   30    --- sync not detected ---
   31    --- sync not detected ---

Information file (m5tok5info.txt) created!!
>
          the contents in info_file (m5tok5info.txt):
*** mk5tok5 information file created by make_mk5tok5_info_file 1.00
*** station : WESTFORD Wf        mode : CDP-SX-SX
***
$CHANNEL;    * channel-track info block
adbit    = 1 ;    *      A/D resolution
sample = 4000000.000000 ;    *      Sampling frequency
fanout = 1 ;    *      Fanout
** ch = ch# : sign_tr# : mag_tr# :    sign_tr# : mag_tr# : .....
     ch = 01 : 18 ;
     ch = 02 : 04 ;
     ch = 03 : 20 ;
     ch = 04 : 06 ;
     ch = 05 : 22 ;
     ch = 06 : 08 ;
     ch = 07 : 24 ;
     ch = 08 : 10 ;
```

```
    ch = 09 : 26 ;
    ch = 10 : 12 ;
    ch = 11 : 28 ;
    ch = 12 : 14 ;
    ch = 13 : 30 ;
    ch = 14 : 16 ;
$BITPOS;      * bit position versus track information
** bitpos = nit_pos# : track #
    bitpos = 00 : -- ;
    bitpos = 01 : -- ;
    bitpos = 02 : 04 ;
    bitpos = 03 : -- ;
    bitpos = 04 : 06 ;
    bitpos = 05 : -- ;
    bitpos = 06 : 08 ;
    bitpos = 07 : -- ;
    bitpos = 08 : 10 ;
    bitpos = 09 : -- ;
    bitpos = 10 : 12 ;
    bitpos = 11 : -- ;
    bitpos = 12 : 14 ;
    bitpos = 13 : -- ;
    bitpos = 14 : 16 ;
    bitpos = 15 : -- ;
    bitpos = 16 : 18 ;
    bitpos = 17 : -- ;
    bitpos = 18 : 20 ;
    bitpos = 19 : -- ;
    bitpos = 20 : 22 ;
    bitpos = 21 : -- ;
    bitpos = 22 : 24 ;
    bitpos = 23 : -- ;
    bitpos = 24 : 26 ;
    bitpos = 25 : -- ;
    bitpos = 26 : 28 ;
    bitpos = 27 : -- ;
    bitpos = 28 : 30 ;
    bitpos = 29 : -- ;
    bitpos = 30 : -- ;
    bitpos = 31 : -- ;
$GROUP;      * group # versus channel # table
** group = group# : ch 1st : ch 2nd : ch 3rd : ch 4th;
    group = 1 :     1 :     2 :     3 :     4 ;
    group = 2 :     5 :     6 :     7 :     8 ;
    group = 3 :     9 :    10 :     9 :    10 ;
    group = 4 :    11 :    12 :    13 :    14 ;
-------------------------------------------------
```

example2: conversion of all groups with 4-ch mode using the info_file file created in example 1. The converted file is created in the directory in which the Mark-5 formatted file exists.

```
> m5tok5 G:/ipvlbi/data/evlbi3/288-1730
```

example3: same as example 2, but the converted file is created in the directory G:/ipvlbi/data/evlbi3/k5 .

```
> m5tok5 G:/ipvlbi/data/evlbi3/288-1730    -d G:/ipvlbi/data/evlbi3/k5
```

example4: conversion by specifying a bit position in 1-ch mode. A bit position #1 is specified. The converted file is created as G:/ipvlbi/data/evlbi3/k5/k5.dat .

> m5tok5 G:/ipvlbi/data/evlbi3/288-1730 G:/ipvlbi/data/evlbi3/k5/k5.dat 1

example5: conversion by specifying bit positions in 4-ch mode. The converted file is created as k5.dat in the current directory. Bit positions #1, #2, #3 and #4 are specified.

> m5tok5 G:/ipvlbi/data/evlbi3/288-1730 k5.dat 1 2 3 4

5.1.3. Data conversion in m5tok5np

This module is used for conversion of data in 1-bit sampling and customized for higher-speed processing. A Mark-5 file should have 16 tracks, be NRZL encoded without parity bits.

usage: same as that in m5tok5 (see sub-section 5.1.2).
info_file in default: m5tok5npinfo.txt

5.1.4. Data conversion in m5tok5R

This module is used for general conversion from Mark-5 to K-5 formats. A Mark-5 file should have 16 o 32 tracks, be NRZL encoded with parity bits.

usage: same as that in m5tok5 (see sub-section 5.1.2).
info_file in default: m5tok5Rinfo.txt

5.1.5. Data conversion in m5tok5Rnp

This module is used for general conversion from Mark-5 to K-5 formats. A Mark-5 file should have 16 or 32 tracks, be NRZL encoded without parity bits.

Note: Since May 2003 session, the EVN data should be applied by this module.

usage: same as that in m5tok5 (see sub-section 5.1.2).
info_file in default: m5tok5npinfo.txt (same as the case in mk5tok5np)

Since 1 August 2003, the following options are also valid only in this module and m5tok5np.

-v [vex_file]: VEX file used to create the Mark-V data.
-sid [stat_id]: Station ID, one or two alphabetical characters.

Example:

```
-------------------------------------------------
jop13{jive}102: m5tok5Rnp d03c1_2.mc -v d03c1.vix -sid Mc -i make
*********************************************************
*       Mark-5 data conversion                          *
*          Ver 1.40     2003-08-01    by    T.KONDO/NICT        *
*********************************************************
 ---------- RUN CONDITION -----------
    mk5name     : ./d03c1_2.mc
    k5name      :
    outdir      : ./
    infofile    : m5tok5npinfo.txt
    channel     : 0
    group       : 0
    soffset     : 0
    period      : 0
    vex_file    : d03c1.vix
    station_id: Mc
```

```
 ---------------------------------------
mk5tok5R running under Information File create mode
info file (m5tok5npinfo.txt) will be created (updated).

Enter VEX file name -> d03c1.vix
VEX file name --- ./d03c1.vix
SITES (Station ID) defined are
   1 -- Ef
   2 -- Mc
   3 -- Wb
   4 -- Nt
   5 -- Jb
   6 -- On
Select Station by number      ----> Mc

Detailed site information
     site definition :    MEDICINA
     site name            :    MEDICINA
     site ID              :    Mc
     site position      : 4461369.868300 919596.955500 4449559.254900
     site clock
          validity epoch : 2003 153 7 30 0
          clock epoch        : 2003 153 9 12 30
          clock offset      : -8.191700e-05
          clock rate         : 3.610000e-13
mode is -1075037092

TRACK and FREQUENCY information for MEDICINA
     BARREL ROLL : off
     FREQDEF = 4982.49MHz8x8MHz      TRACKDEF = MKIV.8Ch2bit1to2
     adbit= 2 sample_rate= 16000000.000000
     bb Tr      AD fo    chan         RF(Hz)              S         VBW(Hz)
      1    2    sign 1 &CH01 4982490000.000000 L 8000000.000000
      2    4    sign 2 &CH01 4982490000.000000 L 8000000.000000
      3    6     mag 1 &CH01 4982490000.000000 L 8000000.000000
      4    8     mag 2 &CH01 4982490000.000000 L 8000000.000000
      5 10    sign 1 &CH02 4982490000.000000 L 8000000.000000
      6 12    sign 2 &CH02 4982490000.000000 L 8000000.000000
      7 14     mag 1 &CH02 4982490000.000000 L 8000000.000000
      8 16     mag 2 &CH02 4982490000.000000 L 8000000.000000
      9 18    sign 1 &CH03 4982490000.000000 U 8000000.000000
     10 20    sign 2 &CH03 4982490000.000000 U 8000000.000000
     11 22     mag 1 &CH03 4982490000.000000 U 8000000.000000
     12 24     mag 2 &CH03 4982490000.000000 U 8000000.000000
     13 26    sign 1 &CH04 4982490000.000000 U 8000000.000000
     14 28    sign 2 &CH04 4982490000.000000 U 8000000.000000
     15 30     mag 1 &CH04 4982490000.000000 U 8000000.000000
     16 32     mag 2 &CH04 4982490000.000000 U 8000000.000000
     17    3   sign 1 &CH05 4998490000.000000 L 8000000.000000
     18    5   sign 2 &CH05 4998490000.000000 L 8000000.000000
     19    7    mag 1 &CH05 4998490000.000000 L 8000000.000000
     20    9    mag 2 &CH05 4998490000.000000 L 8000000.000000
     21 11    sign 1 &CH06 4998490000.000000 L 8000000.000000
     22 13    sign 2 &CH06 4998490000.000000 L 8000000.000000
     23 15     mag 1 &CH06 4998490000.000000 L 8000000.000000
     24 17     mag 2 &CH06 4998490000.000000 L 8000000.000000
     25 19    sign 1 &CH07 4998490000.000000 U 8000000.000000
```

```
    26 21    sign 2 &CH07 4998490000.000000 U 8000000.000000
    27 23      mag 1 &CH07 4998490000.000000 U 8000000.000000
    28 25      mag 2 &CH07 4998490000.000000 U 8000000.000000
    29 27    sign 1 &CH08 4998490000.000000 U 8000000.000000
    30 29    sign 2 &CH08 4998490000.000000 U 8000000.000000
    31 31      mag 1 &CH08 4998490000.000000 U 8000000.000000
    32 33      mag 2 &CH08 4998490000.000000 U 8000000.000000
Mark 5 Data File : ./d03c1_2.mc
Now analizing the data .....
BIT#  HEADER   AUX     SYNC     TIME FIELD   CRC    Y/DDD HH:MM:SS.SSS FMHz
   00 03190319 02110026 ffffffff 31531010 09677466   3/153 10:10:09.677   8.0
   01 03190319 03130026 ffffffff 31531010 096775fe   3/153 10:10:09.677   8.0
   02 03190319 04510026 ffffffff 31531010 096771cb   3/153 10:10:09.677   8.0
   03 03190319 05530026 ffffffff 31531010 09677053   3/153 10:10:09.677   8.0
   04 03190319 06310026 ffffffff 31531010 09677623   3/153 10:10:09.677   8.0
   05 03190319 07330026 ffffffff 31531010 096777bb   3/153 10:10:09.677   8.0
   06 03190319 08710026 ffffffff 31531010 0967720a   3/153 10:10:09.677   8.0
   07 03190319 09730026 ffffffff 31531010 09677392   3/153 10:10:09.677   8.0
   08 03190319 10100026 ffffffff 31531010 09677ddf   3/153 10:10:09.677   8.0
   09 03190319 11120026 ffffffff 31531010 09677c47   3/153 10:10:09.677   8.0
   10 03190319 12500026 ffffffff 31531010 096778b0   3/153 10:10:09.677   8.0
   11 03190319 13520026 ffffffff 31531010 09677928   3/153 10:10:09.677   8.0
   12 03190319 14300026 ffffffff 31531010 09677f9a   3/153 10:10:09.677   8.0
   13 03190319 15320026 ffffffff 31531010 09677e02   3/153 10:10:09.677   8.0
   14 03190319 16700026 ffffffff 31531010 09677af5   3/153 10:10:09.677   8.0
   15 03190319 17720026 ffffffff 31531010 09677b6d   3/153 10:10:09.677   8.0
   16 03190319 18010026 ffffffff 31531010 09677bcf   3/153 10:10:09.677   8.0
   17 03190319 19030026 ffffffff 31531010 09677a57   3/153 10:10:09.677   8.0
   18 03190319 20410026 ffffffff 31531010 09677a5d   3/153 10:10:09.677   8.0
   19 03190319 21430026 ffffffff 31531010 09677bc5   3/153 10:10:09.677   8.0
   20 03190319 22210026 ffffffff 31531010 09677db5   3/153 10:10:09.677   8.0
   21 03190319 23230026 ffffffff 31531010 09677c2d   3/153 10:10:09.677   8.0
   22 03190319 24610026 ffffffff 31531010 09677818   3/153 10:10:09.677   8.0
   23 03190319 25630026 ffffffff 31531010 09677980   3/153 10:10:09.677   8.0
   24 03190319 26000026 ffffffff 31531010 09677520   3/153 10:10:09.677   8.0
   25 03190319 27020026 ffffffff 31531010 096774b8   3/153 10:10:09.677   8.0
   26 03190319 28400026 ffffffff 31531010 09677109   3/153 10:10:09.677   8.0
   27 03190319 29420026 ffffffff 31531010 09677091   3/153 10:10:09.677   8.0
   28 03190319 30200026 ffffffff 31531010 0967740c   3/153 10:10:09.677   8.0
   29 03190319 31220026 ffffffff 31531010 09677594   3/153 10:10:09.677   8.0
   30 03190319 32600026 ffffffff 31531010 09677163   3/153 10:10:09.677   8.0
   31 03190319 33620026 ffffffff 31531010 096770fb   3/153 10:10:09.677   8.0
Information file (m5tok5npinfo.txt) created!!
------------------------------------------------
```

5.2. Step-by-step recipe of data conversion

An input Mark-5 formatted files are assigned here as "MK5".

a) Checking the status of your Mark-5 formatted file.
Run m5check and find out whether the data in the file have parity bits.
If sync is found in the processing "m5check MK5", the data have parity bits.
If sync is found in the processing "m5check MK5 1", the data do not have parity bits.

b) Making a table of correspondence between track numbers and BBCs.

> m5tok5 MK5 -i make (the case for data with parity bits)

> m5tok5np MK5 -i make (the case for data without parity bits)

The module asks the VEX filename. After inputting the filename, the module creates the table.

c) Run format converter.
In the case for data with parity bits,

> m5tok5 MK5 -d outdir
> m5tok5R MK5 -d outdir (for non 1-bit sampling data)

where outdir is the directory in which the output file is created.

In the case for data without parity bits,

> m5tok5np MK5 -d outdir
> m5tok5Rnp MK5 -d outdir (for non 1-bit sampling data)

d) Repeat (b) and (c) (also (a) if necessary) for individual stations.
This has to be done separately for stations that have different track setups; otherwise run it only once.

5.3. Contents of correlation output

The detail of the correlation output is listed in the different document, which is available in the following web page,

http://www2.nict.go.jp/ka/radioastro/IPVLBI/fx_cor_output.pdf .

6. Data correlation

The data correlation procedure is a sequence of preparation of an a-priori parameter file (apri_calc), fringe (clock parameter) search (fx_cor), full baseline-based correlation (fx_cor_all), and coarse search (residual delay and rate estimation) (sdelay).

6.1. Preparation of an a-priori parameter file

work directory: $HOME/apri/ (previously $HOME/ipvlbi/apri/)
used module: apri_calc
The program apri_calc is valid for a SKED DRUDGE and VEX files.
usage:
apri_calc skdfile [apedir [baseid coffset roffset frqgr xdir ydir nobs1 [nobs2]]]
where
skdfile: schedule file.
(Note: If adding a minus "-" in front of the filename (without any space after the minus), the module only monitors the contents of the schedule file.)
apedir: directory for the a-priori calculation output . (default= ../corrapri/)
baseid: baseline ID described in two characters.
coffset: clock delay offset [s] (usually set in a VEX file).
roffset: clock rate offset [s/s] (usually set in a VEX file).
frqgr: frequency group (1-4), corresponding to PC. No default.
A frequency group corresponds to a group of base-band channels. In the case of EVN data with BBC, for example, the group 1 consists of BBC#1-4, the group 2 consists of BBC#5-8.
nobs1: start scan number for processing.
nobs2: end scan number for processing.
(Note: If nobs1=nobs2=0, all possible scans are selected.)
xdir: data directory for x-station.
ydir: data directory for y-station.

(Note: For parameters baseid, coffset, roffset, frqgr, nobs1, nobs2, xdir, and ydir, an interactive input is requested when without any input.)

File name of the output: "apeDDDNNNNXYG.txt"
ape: fixed head characters.
DDD: day of the year at 1-st scan.
NNNN: scan number (4-digit number).
XY: baseline ID (2 characters, ID described in the schedule file).
G: Frequency group ID.

6.2. Quick correlation and fringe search

fx_cor afile [duration soffset coffset roffset t1pp smode pp_mode delsize tzoom pmode comment loop]

where,
afile: file name of a-priori parameters.(default=0: selecting the internal default file name.)
duration: total duration of integration [s]. (negative: the possible maximum duration, 0: the duration described in the schedule file.)
soffset: start offset, should be integer [s]. (0: zero.)
coffset: clock delay offset [s], positive when the Y-station clock is advanced. (default: 0.0 s.)
roffset: clock rate offset [s/s]. (default: 0.0 s/s.)
t1pp: parameter period [s]. it can be set to be shorter than 1 s only when it equals to 1-second divided by integer (e.g. 1/5=0.2 s). (default: 1.0 s.)
smode: number of data points per process, or a size of the delay search window.
0: Maximum (20,000 points), slow processing.
1: Medium (1,000 points).
2 (default): Minimum (1,000 or 2,000 points), fast processing.
pp_mode: synchronization mode.
0 (default): The start of PP is set to an integer in unit of second.
1: No synchronization, or the beginning of the data available for processing. delsize: lag window size in the display (16,32,64,128,256,512,1024,2048,....).
(default: set to DELAYSIZE=128, which can be edited in the source program.)
tzoom: absolute factor of time-axis zooming, should be integer. 0 (default): full scale (=1), negative: set to the window center at the fringe peak.
pmode: device mode in plotting. 0 (default): xterm display plus PostScript output.1: Only PostScript output, 2: Only xterm display, -1: No graphic output.
comment: comment described on the top of the plot.
Do not include any space. If skipped, interactive input is requested (space is valid).
loop: infinite loop parameter.
0 (default): no loop (used for fringe search).
1: infinite loop (only for real-time processing, no display).

Since 24 July 2003, it is possible to optionally set above parameters using the option keys listed as follows. Type only "fx_cor" to see the explanation in detail.

-integ: integration period (sec). Default is scheduled scan period.
If specified explicitly, Quick-look delay windows are displayed.
-coffset: a-priori clock offset (sec) applied for fringe search purpose.
Positive value means Y clock tic earlier than X clock tic. Default is 0.0.
-crate: a-priori clock rate (s/s) difference between X and Y station clock. Default is 0.0
-soffset: start time offset from the scan start (integer sec). Default is 0
-t1pp: unit integration period (parameter period) (sec) for correlation processing.
Default is 1.0
Note: When set < 1.0, 1.0/t1pp should be integer number, e.g., t1pp=0.2, or 0.5 etc.
-smode: search mode selection for processing (for a programmer) internal use.
0: Wide, 1: Mid, 2:Narrow (default 2)

-pp_nosync: turns of the synchronization mode of PP (parameter period) (dfault is that PP synchronize to second tic);
-lag: delay lag window size. The value can be set to 16,32,64,128,256,512, and more. Default is DELAYSIZE(in fx_cor.c). Zero also means the default size.
-pmode: plot mode.
0 : XTERM (env PGDISP can change PGPLOT display device) and PostScript file (pgplot.ps) out (dfault)
1 : PostScript file (pgplot.ps) out only
2 : XTERM only
-1 : No graphic output
-comment: any comment, which will appear in the correlation plot when pmode=0,1,2
-loop: loop parameter for real-time VLBI processing in the future.
0 : no loop (default), 1 : Infinite loop (for real-time use)
-nopcal: turns off the function of the phase calibration signal detection.
Note: EVN clock search should set this option to make the processing faster.

The output file named "cout0000.txt" is created in the directory "cout", which includes the fringe-search results.

Note: When duration of correlated data (-integ) is explicitly specified, delay windows appear as shown in figure 2.

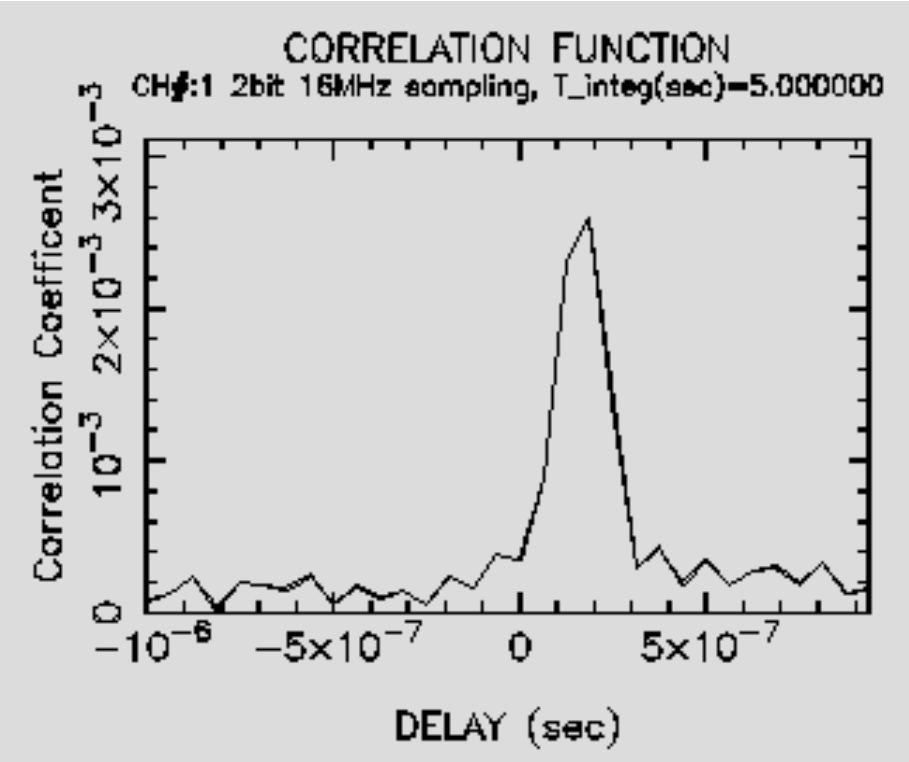

Figure 2: Delay window that appears after executing fx_col.

Note: In the case of correlation between the different BBC (e.g. for cross-hand circular polarization correlation), use the following options.

-ch1 ch1Y defines Y channel for X ch 1 (default 1), 0 for delete
-ch2 ch2Y defines Y channel for X ch 2 (default 2), 0 for delete
-ch3 ch3Y defines Y channel for X ch 3 (default 3), 0 for delete
-ch4 ch4V defines Y channel for X ch 4 (default 4), 0 for delete

If you want to correlate X:ch1 - Y:ch2, X:ch2-Y:ch1, set parameters as follows.

fx_cor afile -ch1 2 -ch2 1

6.3. Full data correlation

a) Move to the directory "corrapri".
b) Prepare a list of a-priori files by typing as follows,
ls -1 apeDDD*XY*.txt > apelistDDDXY.txt ,
where DDD is the day of the year, XY is the baseline code processed. Note that "-1" is minus one.
c) Move to the directory corr.
d) Make backup for the directory "cout" in the directory where the K-5 format data exist.
e) Create the new "cout" directory.
f) Run fx_cor_all as follows,

fx_cor_all [file name of the list of a-priori files] .

6.4. Course search of a residual delay and rate

a) Move the created directory "cout" to a new directory entitled "coutDDDXX", where DDD is the day of the year and XX is the baseline code.

b) Run sdelay.

The module sdelay asks
b.1) a directory (ID) specified above, so input the directory name.
b.2) data processed, so type "0" that means "all data".
b.3) file name of the output (cout####), so choose the file name recommended by the module.

To change the output device, type as follows before executing sdelay.

setenv PGDISP /xterm (using XTERM)
setenv PGDISP /ps (creating PostScript file)
setenv PGDISP /xw (using XWINDOW)

Note: A displayed correlation coefficient is not corrected for e.g. Van Vleck correction.

6.5. Search residual delay and delay rate

The estimated residual delay and rate are displayed in the output of the module “sdelay” (see figure 3). A positive value in the delay means that the Y-station clock is delayed with respected to the X-station. Left-right and front-behind axes in the horizon of the 3-D diagram indicate the delay and delay-rate axes, respectively.

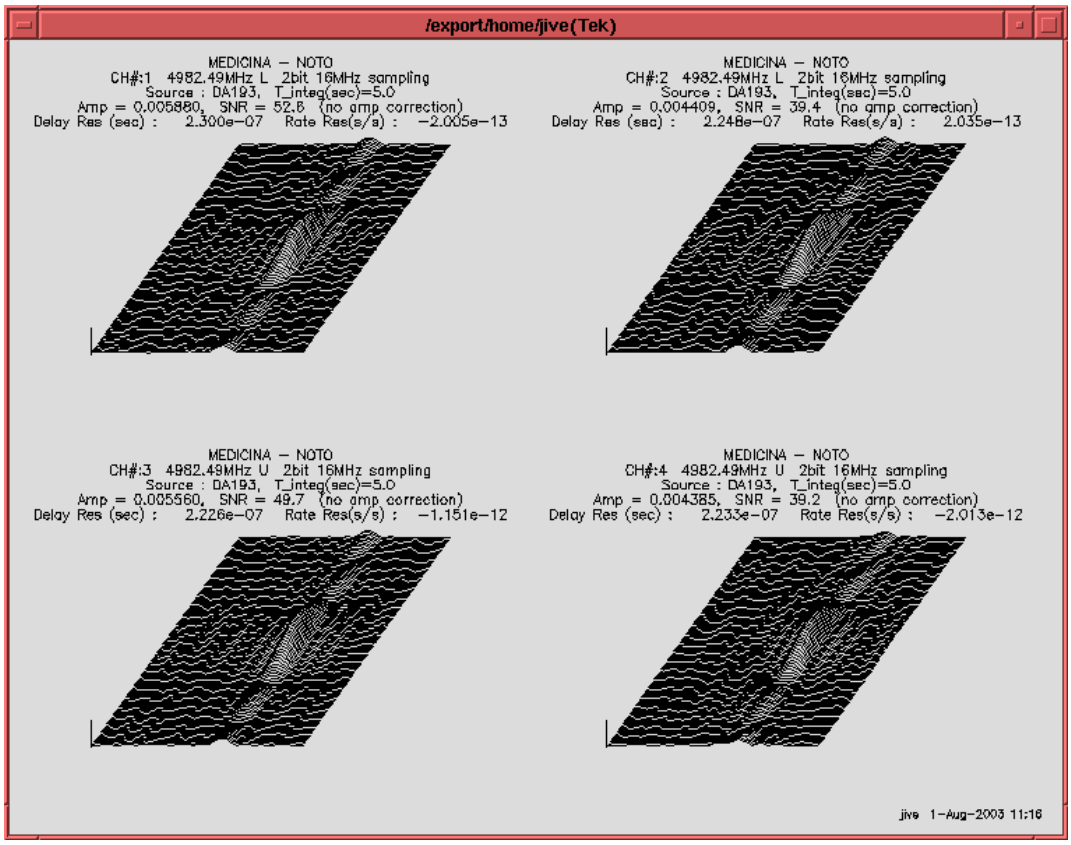

Figure 3: Fringe search windows obtained with the software correlator. These are the output of the module “sdelay”. The data used here was obtained with the Mark-V A backend in the Medicina--Noto baseline in the EVN experiment D03C1 made in May 2003.

Note: If a fringe is located very close to the window edge, a false delay and rate solution might be obtained. In this case, recorrelation is recommended to put the fringe away from the window edge. To do so, increase in a lag number or shift a delay and a rate (by setting a-priori delay and rate residuals) in the a-priori parameter calculation or in data correlation.

6.6. Strategy for fringe search

a) In fringe search, set t1pp=0.2 for a bright quasar and run fx_cor for ~10 s.
b) Check on the display whether a fringe is detected at the center of the search window.
c) Calculate a residual delay (and rate) by running "sdelay cout.txt". Check the fringe if SNR > 7.
d) Input the residual delay and rate as clock parameters, then run apri_calc.
e) Repeat the procedure (a) until the fringe appears at the center of the search window.

Note 1: Without phase-cal tone signals in the data, the processing to find the signals can be skipped to make the whole processing faster. This option improve the CPU speed by about 40%. For this option, type as follows.

```
fx_cor afile -nopcal
or
fx_cor_all afile -nopcal
```