

K5/VSI システムの手引き

木村守孝

2009年07月01日 改訂版 1.0

目次

1	K5/VSI で VSI 信号を取扱うための PC 構成	1
2	CentOS 5.x のインストール	3
2.1	/etc/grub.conf の編集	10
2.2	VSI2000DIM ボード用デバイスドライバのインストール	11
2.3	各種アプリケーションのインストール	12
2.4	High Point 製の Rocket Raid 2340 の認識と XFS ファイルシステムのフォーマット	13
3	ソフトウェア関連器による VERA バックアップ関連処理	14
3.1	処理の流れ	14
3.2	関連スケジュールファイルの生成	15
3.3	観測ファイルの生成	16
3.4	GICO3 用遅延情報ファイルの生成	17
3.5	ソフトウェア関連処理の実行	18
3.6	関連出力ファイルのバンチング処理	20
3.7	複数の関連出力ファイルを結合する	21
3.8	関連出力ファイルを CODA 型式へ変換	22
4	付録	23
4.1	ソフトの操作方法	23
4.1.1	monitor2000 (旧実行名:vsi2000_monitor)	23
4.1.2	vsi2000_time	24
4.1.3	vsi2000_rec	24
4.1.4	capture2000	25
4.2	関連スケジュールの作成	27
4.2.1	異なるターミナル間での関連処理スケジュールファイル	27
4.2.2	偏波観測のための関連処理スケジュールファイル	28
4.2.3	VERA-1 モードと ADS1000[512Msps/2bit/01ch] の混合関連処理	30
4.3	GICO3 の内部動作	31
4.4	各ファイルの命名規則	33
4.5	遅延情報ファイル構造	33
4.6	GICO3 の関連処理でデータの並び	34
4.7	記録・関連スケジュール [構造]	35
4.8	記録・関連スケジュール [サンプル]	36

1 K5/VSIでVSI信号を取扱うためのPC構成

VSI信号をVSI2000DIMボードを使用して内蔵のメモリに取り込むためには、最低限このボードが装着できる空きのPCI-Xソケットがマザーボード上にひとつ必要です。さらに、VSI信号を長時間記録する必要がある場合にはRAIDやSSD等の高速な記録装置が別途必要となります。これらの条件を満たす場合でもラックマウントサーバーでは限られたスペースにPCIボードを挿入するため、ライザーカードを使用するものがありますが、PCI信号のタイミングマージンが小さくなるため、ライザーカード経由での接続はお控え下さい。下記にNICTで推称しているシステム構成とOSを記載します。

表 1: NICTで推称するK5/VSIの構成

構成要素	メーカー等	型番等
マザーボード	Rio Works	HDAMA/HDAM-Express(販売終了)
	ASUS	P5WDG2 Pro/P5E WS
	SuperMicro	X7DBE (現行K5/VSIで使用)
メモリ		4GB
RAIDカード	High Point	Rocket Raid 1820/1820A/2340(現行K5/VSIで使用)
	Areca	ARC-1260
OS		CentOS 5.2 for x86-64

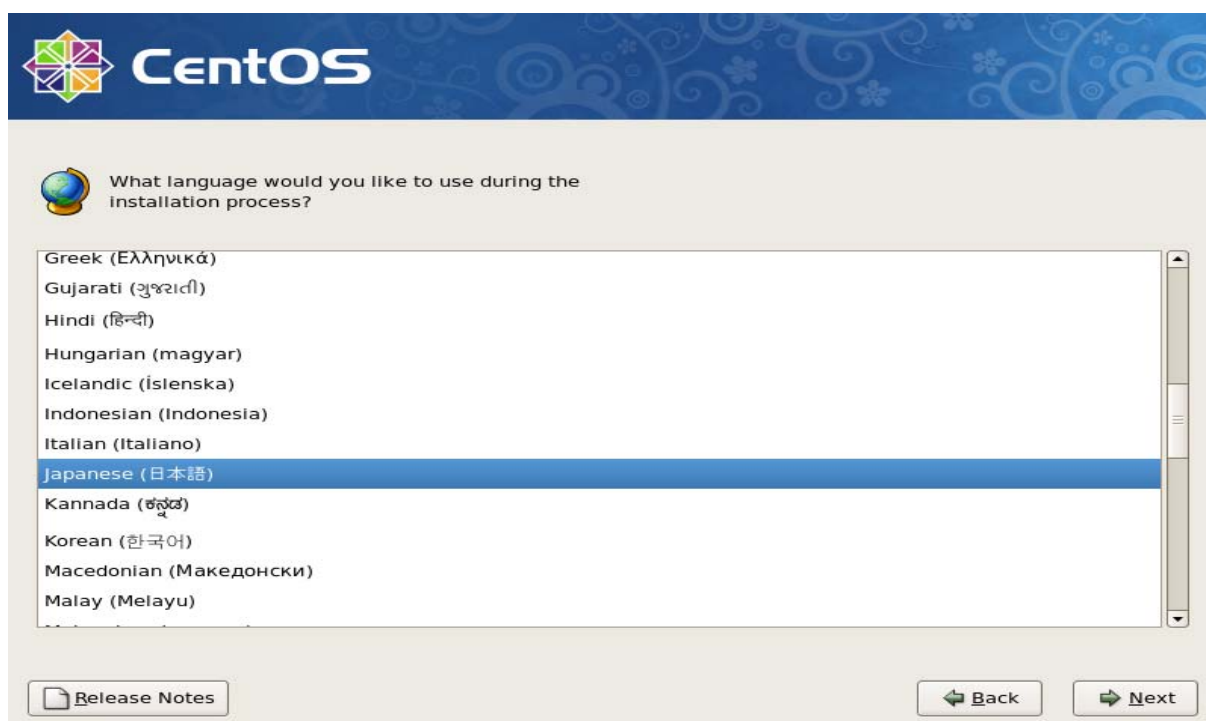
2 CentOS 5.x のインストール

CentOS のインストール自体は通常の手順通りに行なえば問題はありません。参考までに NICT でのインストール設定等を紹介します。

初期画面 Next



インストール時の言語選択 Japanese(日本語) Next



キーボードの選択 日本語 Next



インストール インストール (I) 次



パーティション設定 カスタムレイアウト 次



インストールには、ハードドライブのパーティション設定が必要です。デフォルトでは、ほとんどのユーザーに適しているパーティション設定のレイアウトが選択されます。デフォルトのレイアウトを選択するか、自分で作成することができます。

カスタムレイアウトを作成します。

暗号化されたシステム

このインストールに使用するドライブを選択してください(S)。

hde 238473 MB ASI ARAID99 1010 Rev 1.2

+ 高度なストレージ設定 (A)

パーティションレイアウトの再確認と変更(V)

リリースノート(R) 戻る(B) 次(N)

パーティション設定 ルート=16GB swap=4GB 残り=home 次



ドライブ /dev/hde (238473 MB) (モデル: ASI ARAID99 1010 Rev 1.2)

hde1	hde3
160024	218469 MB

新規(W) 編集(E) 削除(D) リセット(S) RAID(A) LVM(L)

デバイス	マウントポイント/ RAID/ボリューム	タイプ	フォーマットする	容量 (MB)	開始	終了
ハードドライブ						
/dev/hde						
/dev/hde1	/	ext3	✓	16002	1	2040
/dev/hde2		swap	✓	4000	2041	2550
/dev/hde3	/home	ext3	✓	218469	2551	30401

RAID デバイス/LVM ボリュームグループメンバーを非表示にする(G)

リリースノート(R) 戻る(B) 次(N)

ブートローダ デフォルト 次



GRUB ブートローダは、/dev/hde 上にインストールされます。

ブートローダはインストールされません。

他のオペレーティングシステムがブートできるようにブートローダを設定できます。これにより、一覧からブートするオペレーティングシステムを選択できるようになります。自動的に認識されない他のオペレーティングシステムを追加するには、「追加」をクリックします。デフォルトでブートするオペレーティングシステムを変更するには、目的のオペレーティングシステムで「デフォルト」を選択します。

デフォルト	ラベル	デバイス
<input checked="" type="checkbox"/>	CentOS-5.3	/dev/hde1

追加(A) 編集(E) 削除(D)

ブートローダパスワードによってカーネルに渡されるオプションをユーザが変更してしまうのを防ぎます。より高度なシステムセキュリティを確保するために、パスワードを設定することを推奨します。

ブートローダパスワードを使用(U)

高度なブートローダオプションの設定(O)

ネットワーク 適宜設定 次



ネットワークデバイス

起動時にアクティブ	デバイス	IPv4/ネットマスク	IPv6/プレフィックス
<input checked="" type="checkbox"/>	eth0	DHCP	Auto
<input type="checkbox"/>	eth1	DHCP	Auto

編集(E)

ホスト名

ホスト名を設定:

DHCP経由で自動設定(A)

手動設定(M) (例、 host.domain.com)

その他の設定

ゲートウェイ(G):

1 番目の DNS(P):

2 番目の DNS(S):

地域 デフォルト 次



マップ内で地域を選んでクリックしてください:



アジア/東京

システムクロックで UTC を使用 (S)

リリースノート(R) 戻る(B) 次(N)

ルートアカウント 適宜設定 次



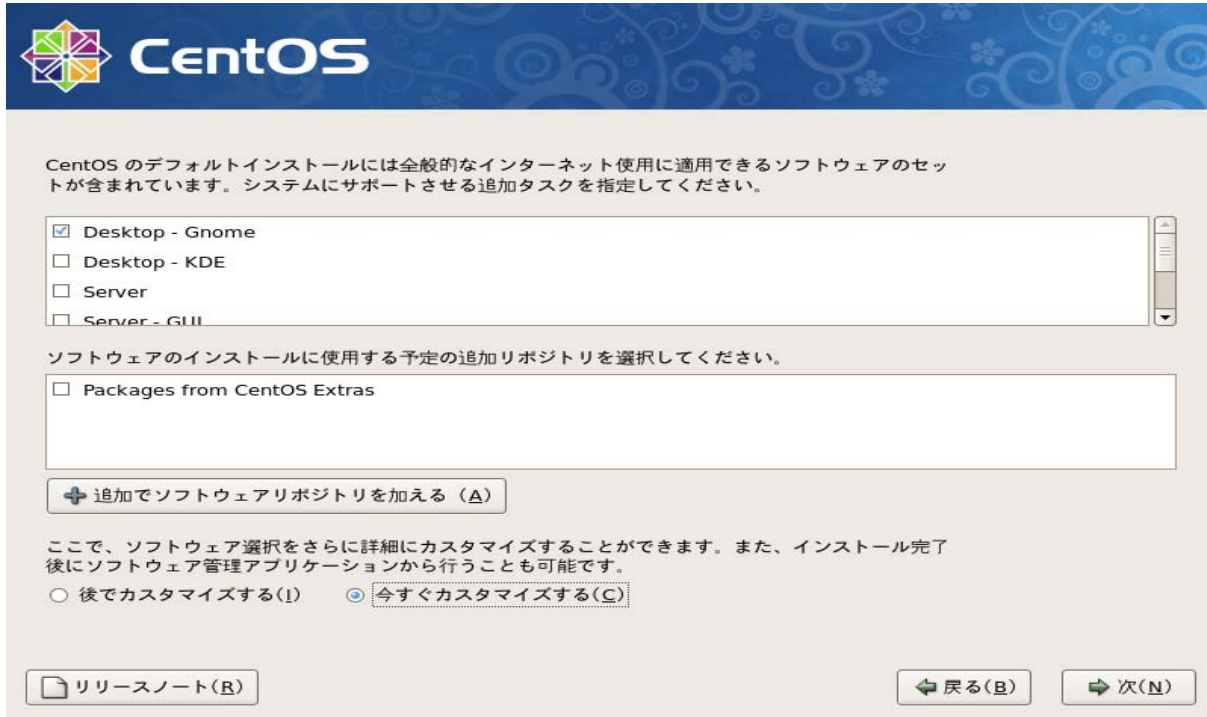
 root アカウントはシステムの管理に使用します。 root ユーザーのパスワードを入力してください。

Root パスワード(P):

確認(C):

リリースノート(R) 戻る(B) 次(N)

追加タスク 今すぐカスタマイズ 次



CentOS のデフォルトインストールには全般的なインターネット使用に適用できるソフトウェアのセットが含まれています。システムをサポートさせる追加タスクを指定してください。

- Desktop - Gnome
- Desktop - KDE
- Server
- Server - GUI

ソフトウェアのインストールに使用する予定の追加リポジトリを選択してください。

- Packages from CentOS Extras

+ 追加でソフトウェアリポジトリを加える (A)

ここで、ソフトウェア選択をさらに詳細にカスタマイズすることができます。また、インストール完了後にソフトウェア管理アプリケーションから行うことも可能です。

後でカスタマイズする (L) 今すぐカスタマイズする (C)

リリースノート (R) 戻る (B) 次 (N)

仮想化/クラスタリング/クラスタストレージ以外は全て選択 次



デスクトップ環境
アプリケーション
開発
サーバー
ベースシステム
仮想化
クラスタリング
クラスタストレージ

Java

- OpenFabrics Enterprise ディストリビューション
- X Window System
- システムツール
- ダイアルアップネットワークサポート
- ベース
- レガシーなソフトウェアのサポート

java プログラミング言語で書かれたプログラムの実行サポート

オプションパッケージ (O)

リリースノート (R) 戻る (B) 次 (N)

インストール開始 次



インストール完了後、ユーザーの登録とセキュリティポリシーの設定等を行なえば OS のインストールは完了です。

2.1 /etc/grub.confの編集

VSI2000DIMボードを使ってVSIデータをPCへ転送するためには、PC上の物理メモリ空間で連続した領域をOSが起動する以前に確保する必要があります。Linuxでは起動オプション "mem="を設定することにより、kernelが使用するメモリサイズを制限することができます。例えば4GBのメモリがある場合には、オプション "mem=2048M"を設定することでアドレス空間が0x80000000-0xFFFFFFFFの2GBの領域はkernelから利用されなくなり、VSI2000DIMボードがこのメモリを内部バッファとして利用することができます¹。下記に起動オプションに "mem=1920M"²を追加したLinuxを2番目に追加し、自動的に2番目から起動されるようにdefault=1³とした "/etc/grub.conf"を以下に記載します。なお、ここでの設定はOSを再起動することにより有効になります。

```
***** /etc/grub.conf ここから *****

# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE: You do not have a /boot partition. This means that
#           all kernel and initrd paths are relative to /, eg.
#           root (hd0,0)
#           kernel /boot/vmlinuz-version ro root=/dev/hde1
#           initrd /boot/initrd-version.img
#boot=/dev/sda
default=1      #0から1に変更
timeout=5
splashimage=(hd0,0)/boot/grub/splash.xpm.gz
hiddenmenu
title CentOS (2.6.18-92.el5)
    root (hd0,0)
    kernel /boot/vmlinuz-2.6.18-92.el5 ro root=LABEL=/ rhgb quiet
    initrd /boot/initrd-2.6.18-92.el5.img
#2番目の起動カーネルに mem=1920Mを追加
title CentOS (K5VSI)
    root (hd0,0)
    kernel /boot/vmlinuz-2.6.18-92.el5 ro root=LABEL=/ rhgb quiet mem=1920M
    initrd /boot/initrd-2.6.18-92.el5.img

***** /etc/grub.conf ここまで *****
```

¹kernelが高位アドレスにPCIのメモリ空間をマッピングする事があるので、実際には0xFFFFFFFFまでは使用できません

²緩衝領域として2048Mより少なめの値を設定しています

³ここではゼロから数えます

2.2 VSI2000DIM ボード用デバイスドライバのインストール

K5/VSI用に配付されている VSI2000DIM 用のデバイスドライバを以下のように rpm コマンドを使いインストールします。

```
rpm -ivh --force vsi2000_driver-2.4.x-x.x86_64.rpm
```

正常にドライバがインストールされた後に、mknod コマンドで以下のようにしてデバイスファイル”/dev/k5vsi”を作成します

```
/sbin/mknod -m 666 /dev/k5vsi c 250 0
```

この後ドライバをカーネルにロードしますが、ここでは1秒分のバッファ長やバッファ数、バッファオフセットアドレス等を指定することができます。前述の/etc/grub.confで”mem=1920M”を指定し、1024Mbitのバッファを8枚分使用する時には以下のような設定をします。

```
/sbin/insmod /lib/modules/2.6.18-92.el5/kernel/drivers/misc/k5vsi.ko Offset=0x80000000 Length=0x08000000 Number=8
```

ドライバが正常にカーネルにロードされた状態で 32MHz の VSI 信号を入力すると、VSI2000DIM ボード上の LED が定期的に点滅し、ボードが正常に動作していることが確認できます。この方法では毎回起動後に上記の設定をしなくてはならないため、正常にドライバの動作が確認出来た場合には以下のように/etc/rc.localに追加します。

```
***** /etc/rc.local ここから *****

#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

touch /var/lock/subsys/local

/bin/mknod -m 666 /dev/k5vsi c 250 0
/sbin/insmod /lib/modules/2.6.18-92.el5/kernel/drivers/misc/k5vsi.ko Offset=0x80000000 Length=0x08000000 Number=8

***** /etc/rc.local ここまで *****
```

上記の設定ではバッファサイズが 0x08000000 バイトであるため、1024Mbps の VSI 信号のみを取扱うことが出来ます。もし 2048Mbps の VSI 信号も取扱う場合には Length の値を 0x10000000 として、Number の値を半分の 4 とします。ここでの指定では 0xBFFFFFFF までのメモリしか使用していませんが、より高位のメモリを使用する場合には、他の PCI デバイスが使用する空間と競合が起きないように/sbin/lspci -vvv などの情報を確認して注意深く設定する必要があります。

2.3 各種アプリケーションのインストール

VSI2000-DIM用に配付されている最新のアプリケーションソフトをRPMコマンドを使いインストールします

```
rpm -ivh --force monitor2000-2.4.x-x.x86_64.rpm
rpm -ivh --force capture2000-2.4.x-x.x86_64.rpm
rpm -ivh --force vsi2000_tool-2.4.x-x.x86_64.rpm
rpm -ivh --force mtkfx-1.0.x-x.x86_64.rpm
```

monitor2000 と capture2000 は GUI 上で動作する単体のアプリケーションであり、信号の監視や記録を実行します。また、vsi2000_tool と mtkfx_tool は以下のような複数のアプリケーションから構成されています。mtkfx 以外のアプリケーションの多くはデバイスドライバを通じて VSI2000DIM にアクセスするため、デバイスドライバと同じバージョンのものを使用して下さい⁴。

表 2: 各パッケージに格納されているソフトウェア

パッケージ名	含まれるソフトウェア
vsi2000_tool	vsi2000_time、vsi2000_rec、gico3_corr、fringe、vsi2000_sender ⁵ 、vsi2000_getter ⁶
mtkfx	log2xml、tau2geo、bunch、join、cor2dat、dat2cor

⁴現在の最新版は 2.4.x となっています

2.4 High Point 製の Rocket Raid 2340 の認識と XFS ファイルシステムのフォーマット

PC に電源を投入した直後、RAID の BIOS 画面が表示されている時に CTRL-H を押し、表示画面にしたがって RAID-0 型式のアレイを作製します⁷。推称環境である CentOS-5.2(64bit) を使用している場合には、K5/VSI 用に配付しているパッケージを以下の rpm コマンドでインストールすることによりドライバソフトのコンパイル作業を省略出来ます。

```
rpm -ihv rr2340-x.x.x-x.x86_64.rpm
```

最新のドライバを使用したい場合には、製造メーカーのホームページ”<http://www.highpoint-tech.com/>” からソースコードをダウンロードして、ソースコード内の Readme 等を参考にインストールして下さい。ドライバのインストール後に OS を再起動すると、RAID が正常に認識されている場合には、dmesg コマンドで以下のようなメッセージを確認することが出来ます。ここでは 1TB のディスク 16 台から構成される約 16TB の RAID が sdb として検出されています。

```
rr2340:RocketRAID 2340 controller driver v1.5.08.0710 (Apr 7 2009 16:08:11)
SCSI device sdb: 31253856256 512-byte hdwr sectors (16001974 MB)
```

CentOS で標準のファイルシステムである ext3 は大きなファイルを取扱う事には不向きなので、K5/VSI ではデータの記録用ファイルシステムとして XFS の使用を前提としています。この XFS は CentOS の標準パッケージではないので、K5/VSI 用に配付しているパッケージを以下の rpm コマンドでインストールします。

```
rpm -ihv kmod-xfs-0.4-1.2.6.18_92.el5.x86_64.rpm
rpm -ihv xfsprogs-2.9.4-1.el5.centos.x86_64.rpm
rpm -ihv xfsprogs-devel-2.9.4-1.el5.centos.x86_64.rpm
```

XFS 用のフォーマットコマンド”mkfx.xfs” で下記の一連の操作で RAID のフォーマットとマウントを行ないます。

```
mkfx.xfs /dev/sdb #sdb として RAID が見えていた場合
mkdir /mnt/raid
mount -t xfs /dev/sdb /mnt/raid
```

このままでは起動時に毎回 mount コマンドを実行する必要があるため、正常に RAID への読み書きが実行出来ることを確認した段階で、/etc/fstab に以下の一行を追加することにより自動でマウント出来る用になります。/dev/sdb /mnt/raid xfs defaults 0 0

⁷詳細は High Point 社の RR2340 用ユーザーマニュアルをご覧ください

3 ソフトウェア相関器による VERA バックアップ相関処理

3.1 処理の流れ

ソフトウェア相関器で VERA 観測の相関処理を行う場合には、一旦磁気テープからハードディスク上に観測ファイルとして観測データのコピーを行います。相関器はこれらの観測ファイルと遅延情報ファイルと相関スケジュールファイルを使用して一連の相関処理を実行します。相関処理に必要なとされる遅延情報ファイルと相関スケジュールファイルは三鷹 FX で使用される遅延情報ファイルと観測局で生成される観測ログファイルから変換ソフトにより生成します。一連の相関処理後に得られる相関出力結果は GICO3 形式で出力されるため、幾つかの変換ソフトを利用した後に、VERA 用の解析系で使用される CODA ファイル形式に変換されます。以下に一連の処理の流れと各処理で使用されるコマンド名を列挙します。

- 観測ログから相関スケジュールファイルを生成する (コマンド名 : log2xml)
- 相関スケジュールファイルを利用して観測テープから観測ファイルにコピーを行なう (コマンド名 : capture2000)
- 三鷹 FX 用遅延情報ファイルから GICO3 用遅延情報ファイルを生成する (コマンド名 : tau2geo)
- 相関スケジュールファイルを利用してソフトウェア相関処理を行なう (コマンド名 : gico3_corr)
- 個々の相関処理結果ファイルに対してバンチング処理を行なう (コマンド名 : bunch)
- 個々のバンチング済相関出力ファイルに対して基線単位で全スキャンを結合する (コマンド名 : join)
- 基線毎の相関出力ファイルを CODA 型式に変換する (コマンド名 : cor2dat)

3.2 関連スケジュールファイルの生成

観測ログファイルから関連スケジュールファイルを生成する為には変換ソフト `log2xml` を使用します。このソフトは下記のように複数の観測ログを指定することにより、多基線用の関連スケジュールを標準出力にしますのでリダイレクト等してスケジュールファイルを生成して下さい。

例 : `log2xml MIZNAO20.log IRIKI.log OGASA20.log ISHIGAKI.log > schedule.xml`

VERA バックアップ関連処理では相関出力速度を三鷹 FX の 0.8 秒と厳密に一致させるため、オプションに `-pp=5` を追加して相関出力速度を 5Hz に設定、相関終了後に 4 個単位で再積分をすることで擬似的に 0.8 秒積分相当の処理結果を得ます。デフォルトの FFT 数は全ストリーム共通で 2048 点となっていますので、変更する場合にはオプションに `-fft=points` を追加します。さらに、ストリーム毎に FFT 数を変えて相関処理する必要がある場合には、生成されたスケジュールファイルを直接編集することで対応が可能です。VERA の観測ログ中に西暦情報は存在しないため、この変換ソフトでは実験コード名の 2 文字目と三文字目を西暦の下 2 桁と仮定し、それに 2000 を足したものを西暦としています。実験コード名が想定されたフォーマットでない場合にはオプション `-year=YYYY` で直接西暦情報を指定することが出来ます。

```
##### コマンドヘルプ #####
名前
    log2xml - VERA 観測ログから GIC03 用スケジュールを生成します
書式
    log2xml VERA 観測ログ VERA 観測ログ... --fft=int --pp=int --year=int
    [--help] [--version]
オプション
    --fft          FFT 数を指定します
    --pp           相関出力速度 [Hz] を指定します
    --year         年号を直接指定します
    --help        この画面を表示します
    --version     バージョンとコンパイルされた日時を表示します
#####
```

3.3 観測ファイルの生成

観測テープから観測ファイルを生成するためには capture2000 と VERA のテープ再生ソフトを同時に使用します。capture2000 のファイル生成の順番はスケジュールファイルにより決まっていますので、VERA のテープ再生ソフトはスケジュールファイル内に記述のある schedule⇒capture の順番でテープを再生する必要があります。テープの再生時刻は TCP 通信により VERA の制御系から得ていますので、使用する IP アドレスとポート番号を capture2000 の起動オプション-VERA で指定します。観測ファイルを記録するディレクトリは-raw-file オプションで指定することで、通常”/mnt/raid”と指定します。ソフトの起動後、メニューからファイル → オープンを選択して関連スケジュールファイルを読み込むと、GUI 上のスケジュール画面にスケジュールが表示されます。VERA 再生系からのテープコピーには再生時刻と K5/VSI システムを同期させるために、同期制御の中から VERA 制御系を選択状態にします⁸。また、テープコピーでは再生時刻が時系列順とはならない場合があるため、記録モードのコピーモードを選択状態にします。記録モード中の”自動時刻”と”非同期”はデバッグ用ですのでかならず非選択状態とします。最後に VERA のテープ再生ソフトをスタートさせた後に、capture2000 の記録開始ボタンを押してファイルにコピーを開始します。スケジュールの途中からコピーを再開する場合には、GUI の記録スケジュール中の左側のチェックリストを非選択にすればそのコピーは無視され途中からテープコピーが続行されます。

```
##### コマンドヘルプ #####
名前
    capture2000 - スケジュールに従ってファイルへ記録します
書式
    capture2000 --schedule=string --raw-file=string
                --VERA=string [--help] [--version]
オプション
    --schedule   スケジュールファイルを指定します
    --raw-file   記録先のディレクトリ名を指定します
    --VERA       テープ時刻に同期させます [string=address:port]
    --debug      デバッグ情報を表示します
    --help       この画面を表示します
    --version    バージョンとコンパイルされた日時を表示します
#####
```

⁸三秒間にわたり時系が安定しており、再生時刻がスケジュールファイル内の再生時刻内であれば時刻の再設定をします

3.4 GICO3 用遅延情報ファイルの生成

GICO3 の相関処理では遅延情報を内部で計算することも出来ますが、VERA 用の相関処理では三鷹 FX 用の遅延情報ファイルから tau2geo を使用して変換される GICO3 用の遅延情報ファイルを使用します。この変換は各観測局毎に全てのチャンネルで行ないます。この変換ソフトで指定するラベル名は log2xml が出力するスケジュールファイルに合わせ”CH01”のようにチャンネル番号をラベル名とします。例として、三鷹 FX 用の遅延情報ファイルが”MIZNAO20.A.taug“、”MIZNAO20.B.taug“として与えられた場合は、以下のコマンドにより、GICO3 用の遅延情報ファイル”MIZNAO20_YYYYDDDDHHMMSS_CH01.geo”、”MIZNAO20_YYYYDDDDHHMMSS_CH02.geo” が生成されます。

```
tau2geo -input=MIZNAO20.A.taug -station=MIZNAO20 -label=CH01
tau2geo -input=MIZNAO20.B.taug -station=MIZNAO20 -label=CH02
```

コマンド実行後、MIZNAO20_YYYYDDDDHHMMSS_CH01.geo 等がカレントディレクトリに生成されますので、後で紹介する gico3_corr の実行時に指定される遅延情報ファイル用のディレクトリにファイルを移動します。⁹

コマンドヘルプ

名前

tau2geo - 三鷹 FX 形式の遅延情報ファイルを GICO3 形式へ変換します

書式

```
tau2geo --input=string --station=string --label=string
<--help> <--version>
```

オプション

- input 三鷹 FX 形式の遅延情報ファイルを指定します
- station 遅延量の計算に使用した観測局名を指定します
- label 遅延量の計算に使用した天体名又はラベル名を指定します
- verbose 詳細な情報を表示します
- help この画面を表示します
- version このソフトのバージョンとコンパイルされた日時を表示します

#####

⁹-station=./geo-file/MIZNAO20 等とするとディレクトリ名 geo-file に直接遅延情報ファイルが生成されます

3.5 ソフトウェア関連処理の実行

ソフトウェア関連器 `gico3_corr` では `log2xml` で生成した関連スケジュールファイルと、`capture2000` で取得した全局の観測データファイルと `tau2geo` で変換した全局/全チャンネルの遅延情報ファイルを使用します。カレントディレクトリ上に関連スケジュールファイル”./schedule.xml” と GICO3 用の遅延情報ファイルを格納したディレクトリ”./geo-file/” と 関連結果が保存されるディレクトリ”./cor-file” を用意した後、以下のコマンドで関連処理を実行します。デフォルトの設定ではフリンジ回転補正を FFT の後に行なっていますが、`-complex` オプションを指定することにより、FFT の前段でフリンジ回転処理を行なうことが出来ます。関連処理の内部動作の付録を参照して下さい。フリンジ検出の確認のために一部のデータの関連処理を実行する場合には、`-epoch` と `-length` で関連処理開始時刻と、関連処理時間を設定します。関連処理が完了すると”./cor-file” 内に各基線毎のファイルが生成されていますので、`fringe` を使用して 2 次元フリンジサーチを行い、フリンジが検出されているかを確認します。

```
gico3_corr --schedule=schedule.xml --raw-file=/mnt/raid --geo-file=./geo-file --cor-file=./cor-file
```

```
##### コマンドヘルプ #####
```

名前

```
gico3_corr - スケジュールファイルに従い関連処理を実行します
```

書式

```
gico3_corr --schedule=string --raw-file=string
--geo-file=string --cor-file=string --multi=int
--epoch='YYYY/DDD HH:MM:SS' --length=int
--complex --command=string [--help] [--version]
```

オプション

```
--schedule      GICO3 用の関連スケジュールファイルを指定します
--raw-file      観測ファイルの格納ディレクトリ名を指定します
--geo-file      遅延ファイルの格納ディレクトリ名を指定します
--geo-file      遅延ファイルの格納ディレクトリ名を指定します
--epoch         [epoch:epoch+length] 間の関連処理のみを実行します
--length        [epoch:epoch+length] 間の関連処理のみを実行します
--multi         関連処理で使用するスレッド数を指定します
--boost         関連処理を間引きして実行速度を見掛け上向上させます
--complex       FFT 前にフリンジ回転処理をおこないます
--command       1 秒毎に指定したスクリプトファイルを起動します
--verbose       関連処理中に詳細を表示します
--help         この画面を表示します
--version       ソフトのバージョンとコンパイルされた日時を表示します
```

```
#####
```

```
fringe -in=cor-file/MIZNAO20_IRIKI_2003328013400_CH01.cor -length=40 -plot
```

```
##### コマンドヘルプ #####
```

名前

fringe - GIC03 形式の相関出力ファイルのフリンジサーチ結果を標準出力へ表示します

書式

```
fringe [--input=file] [--skip=pp] [--length=pp] [--continue] [--plot]
      [--demo=file-name] [--help] [--version]
```

オプション

--input	GIC03 形式の相関出力ファイルを指定します
--skip	処理開始時刻を先頭からの PP 数で指定します
--length	処理する時間長 [PP 単位] を指定します
--continue	length 毎にフリンジサーチ処理を繰り返す
--plot	処理結果をポストスクリプトファイルへ出力します
--rough	ポストスクリプトファイルの質を落とし生成を高速化します
--demo	デモソフト用の出力を指定されたファイルに出力します (未実装)
--help	この画面を表示します
--version	このソフトのバージョンとコンパイルされた日時を表示します

```
#####
```

3.6 相関出力ファイルのバンチング処理

三鷹 FX 相関器にあるバンチング相当の機能は GICO3 では相関処理後にコマンド bunch により実行されます。この bunch は時間領域での積分と周波数領域での積分を同時に行うことができます。バンチング処理のフォーマットは時間積分の後に','と三鷹 FX のバンチング用フォーマットを追加したもので、オプション-pattern='4,0000-1023,16,16' 等と指定します。バンチング処理されたデータはオプション-output で指定されたファイルに出力されます。オプション-output を指定しない場合は、-input で指定されたファイル名のラベル部にバンチングパターンを追加したものを出力ファイル名とします。なお、GICO3 による相関処理での FFT 点数は可変なので、最終的に三鷹 FX と同じ周波数点数を得るためには、三鷹 FX とは異なるパターンを指定する必要がある場合があります。

例 1：水沢-水沢基線の開始時刻"2003/328 01:34:00" の CH01 を周波数領域 0-1023 までを 16 分の 1 に積分する
bunch -input=MIZNAO20_MIZNAO20_2003328013400_CH01.cor -pattern="1,0000-1023,16,16"

例 2：カレントディレクトリにある相関出力ファイルを時間領域で 4 分の 1、周波数領域で 16 分の 1 に積分する
find . -name '*.cor' -exec bunch -pattern='4,0000-1023,16,16' -input {} \;

コマンドヘルプ

名前

bunch - GICO3 の相関出力ファイルを時間・チャンネル積分します

書式

bunch --input=string --output=string --pattern='int,int-int,int,int'
[--help] [--version]

オプション

- input 入力用の GICO3 型式の相関出力ファイル名を指定します
- output 出力用の GICO3 型式の相関出力ファイル名を指定します
- pattern バンチングパターンを指定します 例'1,0000-1023,16,16'
- help この画面を表示します
- version このソフトのバージョンとコンパイルされた日時を表示します

#####

3.7 複数の関連出力ファイルを結合する

GICO3 の関連出力ファイルは各スキャン/各基線/各チャンネル毎に出力されていますが、CODA 形式の関連出力ファイルはスキャン毎に分割されてはならず、ひとつの出力ファイル中に全スキャンが時系列順にならんでいます。そのため、GICO3 の関連出力を CODA 形式へ変換する場合は、一旦全スキャンを join コマンドを使用してひとつに結合させた GICO3 形式の関連出力ファイルの生成する必要があります。この結合では指定されたファイルの順番で結合されるので、時刻が古いものほど先に指定する必要があります¹⁰。このソフトは最初に指定された関連出力ファイルに 2 番目以降の関連出力ファイルのセクター部を追加するように設計されているため、異なる FFT 数を持つ関連出力ファイルの結合は不可能であり、また 2 番目以降の天体情報などは失われます。

例 1 : 水沢-水沢基線のチャンネル CH01 を MIZNAO20_MIZNAO20_YYYYDDDHHMMSS_CH01.cor に結合する
join -output=MIZNAO20_MIZNAO20_YYYYDDDHHMMSS_CH01.cor MIZNAO20_MIZNAO20-*_CH01.cor

```
##### コマンドヘルプ #####
名前
    join - 複数の GICO3 用関連出力ファイルをひとつのファイルに結合します
書式
    join  --output=string FILE-1 FILE-2 [--help] [--version] ...
引数
    FILE-1 ...      結合する GICO3 用関連出力ファイルを複数指定します
オプション
    --output        結合後の GICO3 形式の関連出力ファイルを指定します
    --help          この画面を表示します
    --version       このソフトのバージョンとコンパイルされた日時を表示します
#####
```

¹⁰ 関連出力ファイルの時刻部分を '*' 等で指定した場合、自動的に時系列順に並べられます

3.8 相関出力ファイルを CODA 型式へ変換

全スキャンのデータをひとつに結合した GICO3 型式の相関処理ファイルを cor2dat コマンドを使用して CODA 型式に変換します。この変換時にオプション `--output` を指定しない場合には入力ファイルの拡張子を ".dat" にしたものへ出力を行いません。

例 1: 水沢-水沢基線の全スキャン分の CH01 を CODA 型式に変換する

```
cor2dat --in=MIZNA020_MIZNA020_YYYYDDDDHHMMSS_CH01.cor
```

例 2: ./result 以下にある全スキャン分の相関出力ファイルを変換する

```
find ./result -name '*YYYYDDDDHHMMSS*.cor' -exec cor2dat --input {} \;
```

```
##### コマンドヘルプ #####
```

名前

```
cor2dat - GICO3 型式の相関出力ファイルを CODA 型式の相関出力ファイルへ変換します
```

書式

```
cor2dat --input=string --output=string [--help] [--version]
```

オプション

```
--input      GICO3 型式の相関結果ファイルを指定します
--output     CODA 型式の相関出力ファイルを指定します
--help      この画面を表示します
--version    このソフトのバージョンとコンパイルされた日時を表示します
```

```
#####
```

4 付録

4.1 ソフトの操作方法

4.1.1 monitor2000 (旧実行名: vsi2000_monitor)

VSI2000DIM のモニタソフトである monitor2000 は現在のボードの状態と P-DATA の内容、そしてバッファの内容をリアルタイムで表示します¹¹。このソフトは他の記録ソフトとの同時起動ができますので、観測中は VSI 信号の確認用に立ち上げ状態にしておいて下さい。

Physical Memory	Epoch@Speed	Physical Memory Contents	TVG Check
0x80000000-0x87a11fff	2003/004 01:50:24@1024Mbps	0xffffffff-0x00000000-0x03000401-0x32180000 ...	0x00000000 +000.0000006 [%]
0x88000000-0x8fa11fff	2003/004 01:50:25@1024Mbps	0xffffffff-0x00000000-0x03000401-0x32190000 ...	0x00000000 +000.0000008 [%]
0x90000000-0x97a11fff	2003/004 01:50:26@1024Mbps	0xffffffff-0x00000000-0x03000401-0x321a0000 ...	0x00000000 +000.0000008 [%]
0x98000000-0x9fa11fff	2003/004 01:50:19@1024Mbps	0xffffffff-0x00000000-0x03000401-0x32130000 ...	0x00000000 +000.0000008 [%]
0xa0000000-0xa7a11fff	2003/004 01:50:20@1024Mbps	0xffffffff-0x00000000-0x03000401-0x32140000 ...	0x00000000 +000.0000006 [%]
0xa8000000-0xafa11fff	2003/004 01:50:21@1024Mbps	0xffffffff-0x00000000-0x03000401-0x32150000 ...	0x00000000 +000.0000008 [%]
0xb0000000-0xb7a11fff	2003/004 01:50:22@1024Mbps	0xffffffff-0x00000000-0x03000401-0x32160000 ...	0x00000000 +000.0000008 [%]
0xb8000000-0xbfa11fff	2003/004 01:50:23@1024Mbps	0xffffffff-0x00000000-0x03000401-0x32170000 ...	0x00000000 +000.0000010 [%]

Annotations: ① points to Epoch@Speed, ② to TVG Check, ③ to Physical Memory, ④ to Epoch@Speed, ⑤ to Physical Memory Contents, ⑥ to TVG Check.

- 1: VSI2000DIM の現在時刻と動作速度が表示されます (ここに表示される時刻がすべての時間の基本となります)
- 2: 1 秒前の P-DATA の内容が表示されます (P-DATA が無い場合は何も表示されません)
- 3: 各バッファが使用するメモリ空間を表示します
- 4: 各バッファに記録された観測時刻とその時のデータ速度を表示します
- 5: 各バッファに記録された VSI データの最初の 4 ワードと最後の 1 ワードを表示します
- 6: TVG 信号との一致度を表示します

```
##### コマンドヘルプ #####
名前
    vsi2000_monitor - K5VSI の現在状況を表示します
書式
    vsi2000_monitor --noTVG [--help] [--version]
オプション
    --noTVG      TVG チェック機能を無効にします
    --help      この画面を表示します
    --version    バージョンとコンパイルされた日時を表示します
#####
```

¹¹このソフトは /dev/mem にアクセスするため root での起動が必要です

4.1.2 vsi2000_time

vsi2000_time コマンドを使用することで、K5/VSI のデバイス時刻を指定時刻に設定することが出来ます。

```
##### コマンドヘルプ #####
```

名前

```
vsi2000_time - K5VSI のデバイス時刻を設定します
```

書式

```
vsi2000_time --set='YYYY/DDD HH:MM:SS' --ads1000  
--now --show=int <--help> <--version>
```

オプション

```
--set          デバイス時刻を 'YYYY/DDD HH:MM:SS' に設定します  
--ads1000      デバイス時刻を ADS1000 の P-DATA に同期させます  
--now          デバイス時刻を計算機時刻に設定します  
--show        デバイス時刻を n 秒間表示します  
--help        この画面を表示します  
--version     バージョンとコンパイルされた日時を表示します
```

```
#####
```

4.1.3 vsi2000_rec

vsi2000_rec コマンドを使用することで、指定した時刻から VSI 信号をディスク上に記録することが出来ます。この時ディスク上に記録されるファイル名はオプション-file で指定した文字列に記録開始時の時刻と拡張子”_YYYYDDDDHHMMSS.raw”が追加されます。

```
##### コマンドヘルプ #####
```

名前

```
vsi2000_rec - VSI 信号をファイルへ記録します
```

書式

```
vsi2000_rec --epoch='YYYY/DDD HH:MM:SS' --length=int  
--file=string <--help> <--version>
```

オプション

```
--epoch      指定時刻 'YYYY/DDD HH:MM:SS' から記録を開始します  
--length     記録時間長 [秒] を指定します  
--file       記録ファイル名を設定します  
--verbose    記録中に詳細を表示します  
--help      この画面を表示します  
--version   このソフトのバージョンとコンパイルされた日時を表示します
```

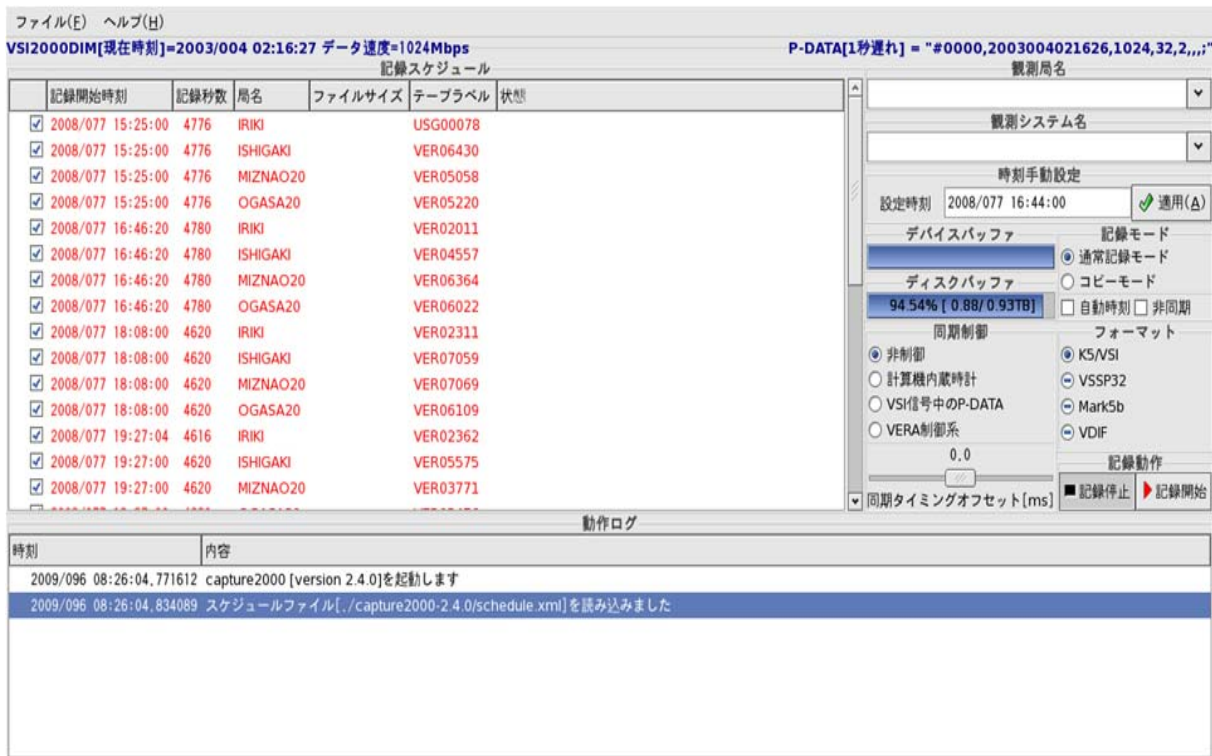
```
#####
```


4.1.4 capture2000

数多くの記録を実行する場合には、スケジュールファイルを XML 形式で記述することで、GUI ベースのスケジュール記録ソフト capture2000¹²が使用できます。このソフトはスケジュールファイル中の複数の schedule⇒capture で記述された記録を逐次的に行ないます。以下に記録のみを目的としたスケジュールファイルの例を記載します¹³。

```
<?xml version='1.0' encoding='UTF-8' ?>
<schedule>
  <station key='R'><name>IRIKI</name> ... </station>
  <capture><epoch>2008/077 15:25:00</epoch><length>4776</length><station>IRIKI</station><label>USG00078</label></capture>
  <capture><epoch>2008/077 16:46:20</epoch><length>4780</length><station>IRIKI</station><label>VER02011</label></capture>
</schedule>
```

スケジュールファイルを用意した後、capture2000 を起動し、上段のメニューバーのファイルからスケジュールファイルを選択します。ファイル選択後、通常の記録観測時には適切な測局名と観測システム名を GUI 右上の場所で選択します。テープコピー時にはここは空欄のままとします。通常の記録観測時には同期制御を ADS1000 互換機の場合は P-DATA、ntp で時計が制御されている場合は計算機内蔵時計を選択します。どちらも使用出来ない場合には、ここで非制御を選択し、時刻手動設定でマニュアルで時刻を合わせます¹⁴。VERA からのテープコピーではここで VERA 制御系を選択します。VERA 制御系を使用する場合には、TCP 通信を行なう必要がありますので、起動時にオプション VERA を指定してアドレスとポート番号を設定する必要があります。同期処理を VERA 制御系にした場合には、VSI 中の 1PPS 信号と TCP 経由で得られる時計の繰り上がりタイミングのずれを補正するため、同期タイミングオフセットを適宜指定することが出来ます¹⁵。記録モードは通常の観測時には通常記録モードを選択し、テープコピーの場合にはコピーモードを選択します。記録モードの下段の自動時刻と非同期はデバッグ用コマンドで通常は非選択状態で使用します¹⁶全ての設定後、記録動作の記録開始を押して記録動作を開始します。



\clearpage

¹²このソフトは /dev/mem にアクセスするため root での起動が必要です
¹³label 情報は capture2000-2.4.x では使用していませんが適当な文字列を指定する必要があります
¹⁴正午に時計を合わせる場合には”西暦/通日 12:00:00”と代入して、12:00:00+0.5 秒あたりで適用ボタンを押します
¹⁵VSI 信号の 1PPS を検出したのち、250+オフセット [ミリ秒] 後に VERA 制御系に時計を读出しに行きます
¹⁶自動時刻は各スキャン開始時に時計を開始 5 秒前に再設定します。非同期は K5/VSI の時刻を無視して記録を行ないます

```
##### コマンドヘルプ #####
名前
    capture2000 - スケジュールに従ってファイルへ記録します
書式
    capture2000 --schedule=string --raw-file=string
                --VERA=string [--help] [--version]
オプション
    --schedule   スケジュールファイルを起動時に指定します
    --raw-file   記録先のディレクトリ名を指定します
    --VERA       テープ時刻に同期させます [string=address:port]
    --debug      デバッグ情報を表示します
    --help       この画面を表示します
    --version    バージョンとコンパイルされた日時を表示します
#####
```

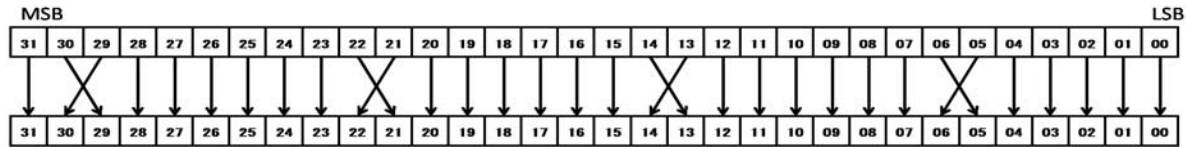
4.2 関連スケジュールの作成

4.2.1 異なるターミナル間での関連処理スケジュールファイル

GICO3 での関連処理は shuffle 機能により 32 ビット内の並びを自由に設定することが出来ます。この並び変えは変換前の 00 から 31 までのビットが変換後どのビットに対応するかを schedule⇒terminal⇒shuffle に指定します。GICO3 で正常に関連処理するためには変換後のビット並びは付録にあるデータの並びに合っていることが必要です。例えば 32 ビットの中で第 05 ビットと第 06 ビット、第 13 ビットと第 14 ビット、第 21 ビットと第 22 ビット、第 29 ビットと第 30 ビットを入れ替えて関連処理を行いたい場合には、下記のように 32 要素を設定します。

```
<shuffle key="A">  
31,29,30,28,27,26,25,24,23,21,22,20,19,18,17,16,15,13,14,12,11,10,09,08,07,05,06,04,03,02,01,00  
</shuffle>
```

記録データのビット並び



関連処理で使用するデータのビット並び

多ビットのデータをどのような数値にデコードするかは schedule⇒terminal⇒level で設定します。2 ビット量子化の場合に二進数で 00 から 11 までの 4 段階を -1.5, -0.5, +0.5, +1.5 として解釈させたい場合に以下のように記述します¹⁷。

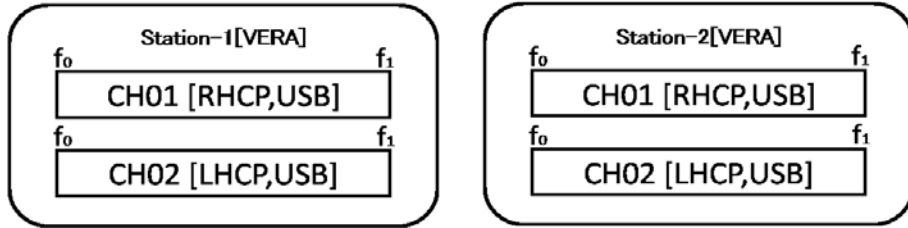
```
<level>-1.5,-0.5,+0.5,+1.5</level>
```

観測局毎に USB か LSB の違いがある場合には、schedule⇒process⇒stream⇒frequency に USB での観測周波数を記述して、schedule⇒process⇒stream⇒special⇒sideband に USB か LSB を記述することで混合関連処理が可能です。

¹⁷VERA 系の場合には -1.5, -0.5, +0.5, +1.5、ADSx000 系の場合には -1.5, +0.5, -0.5, +1.5 とする必要があります

4.2.2 偏波観測のための相関処理スケジュールファイル

相関処理で全ての偏波の組み合わせを計算する場合には、一つの観測局を仮想的に二局相当とみなした相関スケジュールファイルを作成します。ここでは、VERA-1 モードで Station1, Station2 の 2 局で 2 偏波観測した場合での相関スケジュールファイルの作成例を示します。



はじめに観測局を倍にするために局名に"-R"、"-L"を付け、それぞれのkeyを大文字と小文字で区別した合計4個のschedule⇒stationを記述します。

```
<station key="A"><name>Station1-R</name><terminal>VERA-01</terminal></station>
<station key="a"><name>Station1-L</name><terminal>VERA-01</terminal></station>
<station key="B"><name>Station2-R</name><terminal>VERA-01</terminal></station>
<station key="b"><name>Station2-L</name><terminal>VERA-01</terminal></station>
```

GICO3での相関処理ではschedule⇒station⇒nameで記述される観測局名から始まる観測ファイルを使用するため、デフォルトのままではこれらのファイルが存在しないため相関処理が失敗します。そこで相関処理に使用する観測ファイルをschedule⇒process⇒station⇒fileで記述します。

```
<station key="A"><file>/mnt/raid/Station1_2003328013400.raw</file></station>
<station key="B"><file>/mnt/raid/Station2_2003328013400.raw</file></station>
<station key="a"><file>/mnt/raid/Station1_2003328013400.raw</file></station>
<station key="b"><file>/mnt/raid/Station2_2003328013400.raw</file></station>
```

それぞれの観測局で適切なチャンネル番号を使用するために、schedule⇒process⇒stream⇒specialにチャンネル番号1と2を記述します。

```
<stream>
  <special key="A"><channel>01</channel></special>
  <special key="B"><channel>01</channel></special>
  <special key="a"><channel>02</channel></special>
  <special key="b"><channel>02</channel></special>
</stream>
```

以上の記述以外に汎用的な記述を追加し、実際に GICO3 で動作させるスケジュールファイルは以下のようになります。この例では一つのスキャン分しか記述されていませんが、複数の schedule⇒process を記述すれば数多くのスキャンに対応できます

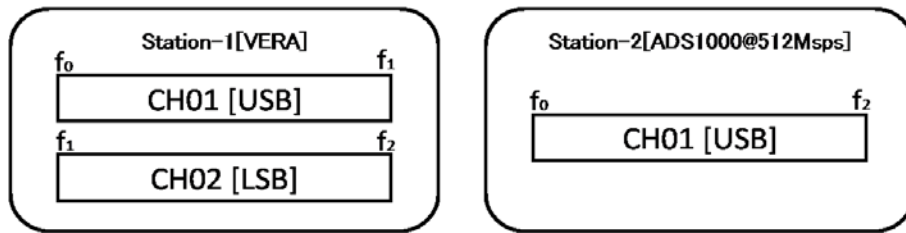
```
<?xml version="1.0" encoding="UTF-8" ?>
<schedule>
  <terminal name="VERA-01">
    <speed> 256000000</speed><channel> 2</channel><bit>2</bit><level>-1.5,-0.5,+0.5,+1.5</level>
  </terminal>
  <station key="A"><name>Station1-R</name><terminal>VERA-01</terminal></station>
  <station key="a"><name>Station1-L</name><terminal>VERA-01</terminal></station>
  <station key="B"><name>Station2-R</name><terminal>VERA-01</terminal></station>
  <station key="b"><name>Station2-L</name><terminal>VERA-01</terminal></station>

  <source name="CH01"><ra>00h00m00.000000</ra><dec>+00d00'00.00000</dec></source>

  <process>
    <start>2003/328 01:34:00</start><length>3960</length><object>Multi</object>
    <stations>AaBb</stations>
    <station key="A"><file>/mnt/raid/Station1_2003328013400.raw</file></station>
    <station key="B"><file>/mnt/raid/Station2_2003328013400.raw</file></station>
    <station key="a"><file>/mnt/raid/Station1_2003328013400.raw</file></station>
    <station key="b"><file>/mnt/raid/Station2_2003328013400.raw</file></station>
    <stream>
      <label>CH01</label><source>CH01</source><frequency>2222000000</frequency>
      <channel>01</channel><fft>2048</fft><output>1</output>
      <special key="A"><channel>01</channel></special>
      <special key="B"><channel>01</channel></special>
      <special key="a"><channel>02</channel></special>
      <special key="b"><channel>02</channel></special>
    </stream>
  </process>
</schedule>
```

4.2.3 VERA-1モードとADS1000[512Msps/2bit/01ch]の混合相関処理

下図のような周波数配置でVERA-1モードとADS1000[512Msps/2bit/01ch]の混合相関をする場合には、基本的な設定はVERA-1モードと同じで2チャンネル分の相関処理スケジュールファイルを作成します。



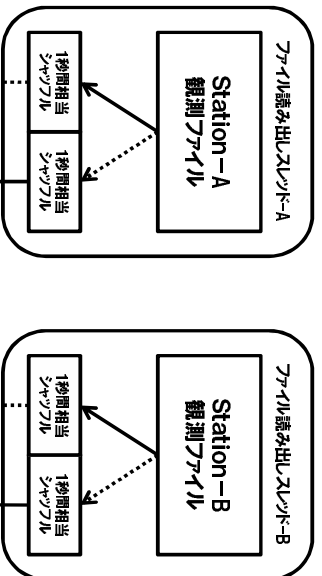
VERA側の第一チャンネルとADS1000側の第一チャンネルは観測周波数が同じなので、正しくterminalが定義されていれば、通常のスケジュールのままに相関処理が可能です。VERA側の第二チャンネルとADS1000側の第一チャンネルはチャンネル番号と観測周波数が異なるので、`schedule⇒process⇒stream⇒special⇒channel`、`schedule⇒process⇒stream⇒special⇒offset`、でチャンネル番号と周波数オフセットを記述する必要があります。相関処理ではVERA局では第二チャンネル、ADS1000局では第一チャンネルを使用するため、channelに2と1をそれぞれ記述します。このスケジュールファイルでは512Mspsのデータを2048点でFFTする設定になっており、その場合には128MHzは512個相当のオフセット量となるため、ADS1000局側のoffsetに+512を記述します。

```
<?xml version="1.0" encoding="UTF-8" ?>
<schedule>
  <terminal name="VERA-01">
    <speed> 256000000</speed><channel> 2</channel><bit>2</bit><level>-1.5,-0.5,+0.5,+1.5</level>
  </terminal>
  <terminal name="ADS1000">
    <speed> 512000000</speed><channel> 1</channel><bit>2</bit><level>-1.5,+0.5,-0.5,+1.5</level>
  </terminal>
  <station key="A"><name>Station-1</name><terminal>VERA-01</terminal></station>
  <station key="B"><name>Station-2</name><terminal>ADS1000</terminal></station>
  <source name="CH01"><ra>00h00m00.000000</ra><dec>+00d00'00.000000</dec></source>
  <source name="CH02"><ra>00h00m00.000000</ra><dec>+00d00'00.000000</dec></source>
  <process>
    <start>2003/328 01:34:00</start><length>3960</length><object>Multi</object>
    <stations>AB</stations>
    <stream>
      <label>CH01</label><source>CH01</source><frequency>22000000000</frequency>
      <channel>01</channel><fft>2048</fft><output>1</output>
    </stream>
    <stream>
      <label>CH02</label><source>CH02</source><frequency>22128000000</frequency>
      <channel>02</channel><fft>2048</fft><output>1</output>
      <special key="A"><channel>02</channel><offset>+0000</offset><sideband>LSB</sideband></special>
      <special key="B"><channel>01</channel><offset>+0512</offset><sideband>USB</sideband></special>
    </stream>
  </process>
</schedule>
```

4.3 GICO3の内部動作

FX型の多基線相関器として設計されているGICO3の内部動作について簡単に説明します。ここでは単純化のため2局分のデータを相関処理する場合について、下図にその内部処理を示します。ソフトウェア相関器の処理は内部で1秒単位で実行されており、相関処理に必要な観測データを準備する複数の読み出しスレッドと、実際の演算を実行する複数のスレッドが並列に動作しています。読み出しスレッドは演算を実行中に次に必要になる1秒分のデータをファイルから読み出ししており、必要があればここで読み出されたデータのシャッフルも行います。ソフトウェア相関処理では1秒間の処理をストリームとよばれる単位に分解して処理を行います。1個のストリームは1個のチャンネル番号を記述することができます。さらに、各ストリームは相関処理スケジュールの `schedule⇒process⇒stream⇒output` で記述される相関出力速度 ($n[\text{Hz}]$) 毎に細分化され、プロセッサのコア数と等しい m 個のスレッドからなるGICO3演算コアにより分散処理が行われます。GICO3演算コア内部では1つのスレッドが $1/mn$ 秒相当の処理を独立して行い、1FFT区間に相当する時間単位で処理が進められます。GICO3演算コアははじめに観測データから指定されたチャンネル番号を選択して単精度型の浮動小数点に変換します。この時使用されるチャンネル番号は `schedule⇒process⇒stream⇒channel` に記述されているものを使用しますが、観測局毎に異なる選択をする場合には `schedule⇒process⇒stream⇒special⇒channel` で個別に記述します。GICO3での相関処理はすべてUSBで計算されるため、LSBの場合にはここでUSBに変換されます。GICO3での相関処理では処理速度を向上させるため実数FFTが使用されますが、基線長が長く観測周波数が高い場合には問題を生じる場合がありますので、相関処理の引数に `-complex` が指定されている場合には、FFTの前段でフリッジ回転補正が行われた後、複素FFTが使用されます。FFTでは全局同じFFT数の変換が実行されますが、異なるサンプリング速度間の混合相関処理の場合には、もっともサンプリング速度が大きい観測局のデータを `schedule⇒process⇒stream⇒fft` で記述されたFFT数でフーリエ変換を行い、それよりサンプリング速度が小さい観測局のデータは周波数分解能が等しくなるように小さいFFT数でフーリエ変換が行われます。フーリエ変換後にそれぞれの周波数分解能の単位で `schedule⇒process⇒stream⇒special⇒offset` で信号をシフトさせた後、群遅延相当の位相傾斜の補正と引数 `complex` が指定されていない場合はここでフリッジ回転補正が実行されます。サンプリング速度の違いや周波数シフトにより観測信号の無い周波数成分にはゼロが代入された後、全組み合わせでの相関が行われ1FFT区間の処理が完了します。全てのGICO演算コアで $1/mn$ 秒分の処理が完了した後に全スレッド分の処理結果が一つに結合され、最終的に1つのファイルとして出力されます。

スキューンの開始時刻から終了時刻まで1秒単位でルーブ開始



スキューン毎のルーブ開始

相関出力毎 [Hz] のルーブ開始

GIC03演算

Aの自己相関

A&Bの相互相関

Bの自己相関

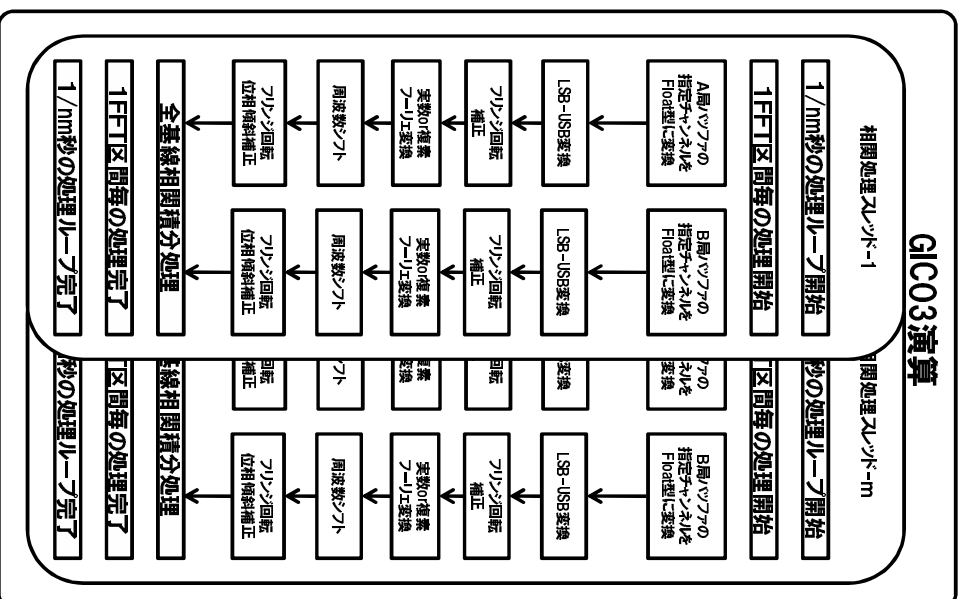
相関出力フレイムに追加記録

相関出力毎 [Hz] のルーブ完了

スキューン毎のルーブ完了

スキューンの開始時刻から終了時刻まで1秒単位でルーブ終了

GIC03演算



GIC03 の内部動作

4.4 各ファイルの命名規則

capture2000 及び gico3_corr 等で取扱う各種ファイルの名前は以下の命名規則により決定されています。スケジュールファイル中に個々のファイル名を明示的に記述していなければ、暗黙的にこの規則が適応されます。但し、VERA バックアップ関連処理では途中の処理を簡便に行なうために、処理の過程において一部拡張型を使用する事があります。

表 3: 各ファイルの命名規則 (基本型)

ファイルの種類	命名規則	例
観測データファイル	局名+"_"+"開始時刻 [YYYYDDDDHHMMSS]+".raw"	IRIKI_2003328013400.raw
遅延情報ファイル	局名+"_"+"開始時刻 [YYYYDDDDHHMMSS]+ラベル名+".geo"	IRIKI_2003328013400_CH01.geo
遅延情報ファイル	局名+"_"+"開始時刻 [YYYYDDDDHHMMSS]+天体名+".geo"	IRIKI_2003328013400_3C345.geo
相関結果ファイル	局名+"_"+"局名+"_"+"開始時刻 [YYYYDDDDHHMMSS]+ "_"+"ラベル名"+".cor"	IRIKI_IRIKI_2003328013400_CH01.cor

表 4: 各ファイルの命名規則 (拡張型)

ファイルの種類	命名規則	例
遅延情報ファイル	局名+"_"+"YYYYDDDDHHMMSS"+ラベル名+".geo"	IRIKI_YYYYDDDDHHMMSS_CH01.geo
遅延情報ファイル	局名+"_"+"YYYYDDDDHHMMSS"+天体名+".geo"	IRIKI_YYYYDDDDHHMMSS_3C345.geo
相関結果ファイル	局名+"_"+"局名+"_"+"YYYYDDDDHHMMSS"+ "_"+"ラベル名"+".cor"	IRIKI_IRIKI_YYYYDDDDHHMMSS_CH01.cor

4.5 遅延情報ファイル構造

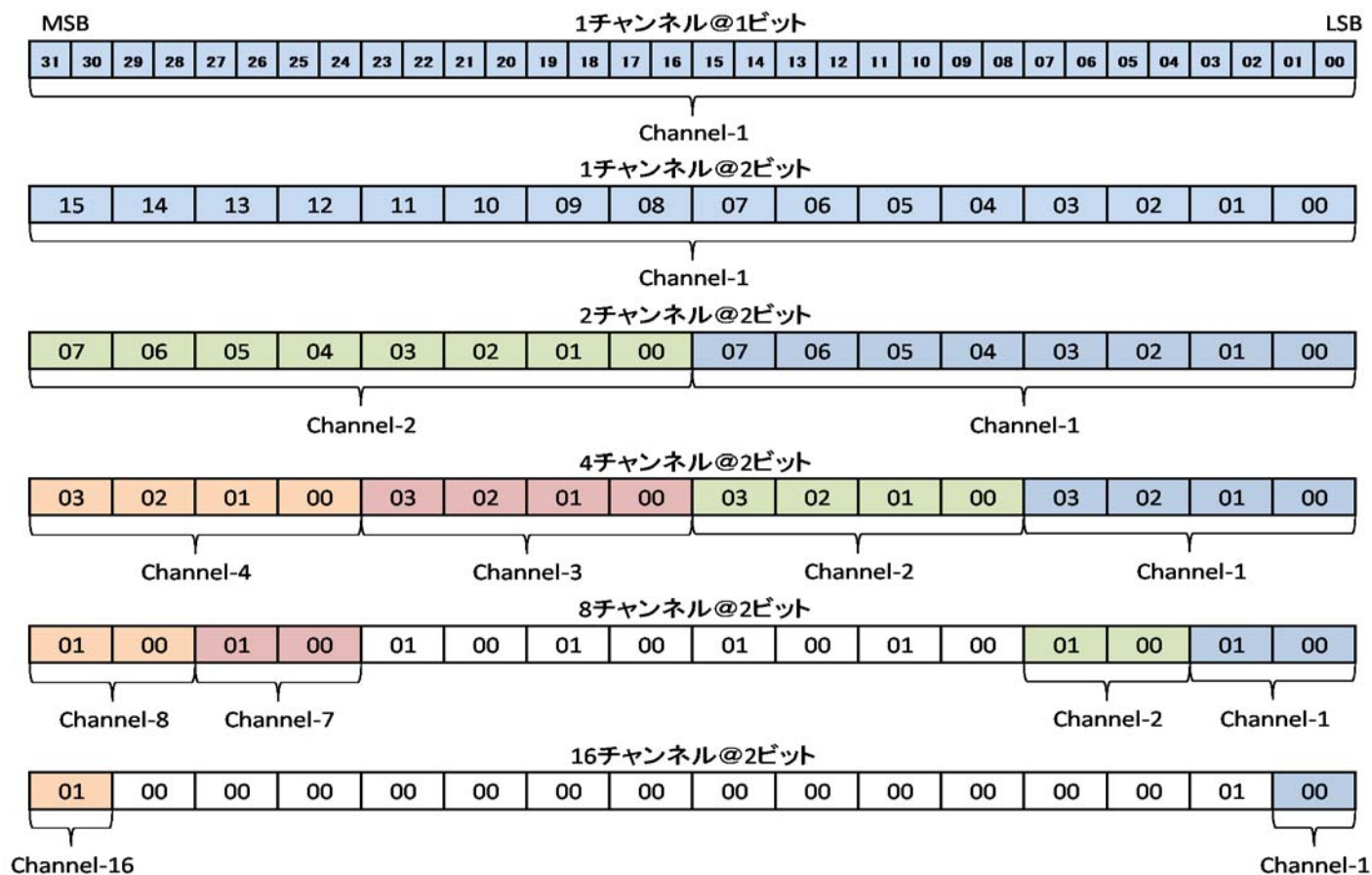
GICO3 で使用される遅延情報ファイルは時刻情報と遅延情報から構成されており、時刻情報は'1970/001 00:00:00'からの経過時刻の秒数を格納する 4 バイト長の *time_t* 型とナノ秒を格納する 4 バイト長の整数型で構成されています。ここでの遅延量は 4 次の微分係数までを含む倍精度浮動小数点型として格納されています。

表 5: 遅延情報ファイルのフォーマット (リトルエンディアン型/IEEE 754 型準拠)

オフセット	00	01	02	03	04	05	06	07
0x00000000	時刻 [秒 : time_t]				時刻 [ナノ秒 : int]			
0x00000008	遅延量 0 階微分 [double]							
0x00000010	遅延量 1 階微分 [double]							
0x00000018	遅延量 2 階微分 [double]							
0x00000020	遅延量 3 階微分 [double]							
0x00000028	遅延量 4 階微分 [double]							
0x00000030	時刻 [秒 : time_t]				時刻 [ナノ秒 : int]			
0x00000038	遅延量 0 階微分 [double]							
0x00000040	遅延量 1 階微分 [double]							
0x00000048	遅延量 2 階微分 [double]							
0x00000050	遅延量 3 階微分 [double]							
0x00000058	遅延量 4 階微分 [double]							
.....	.							

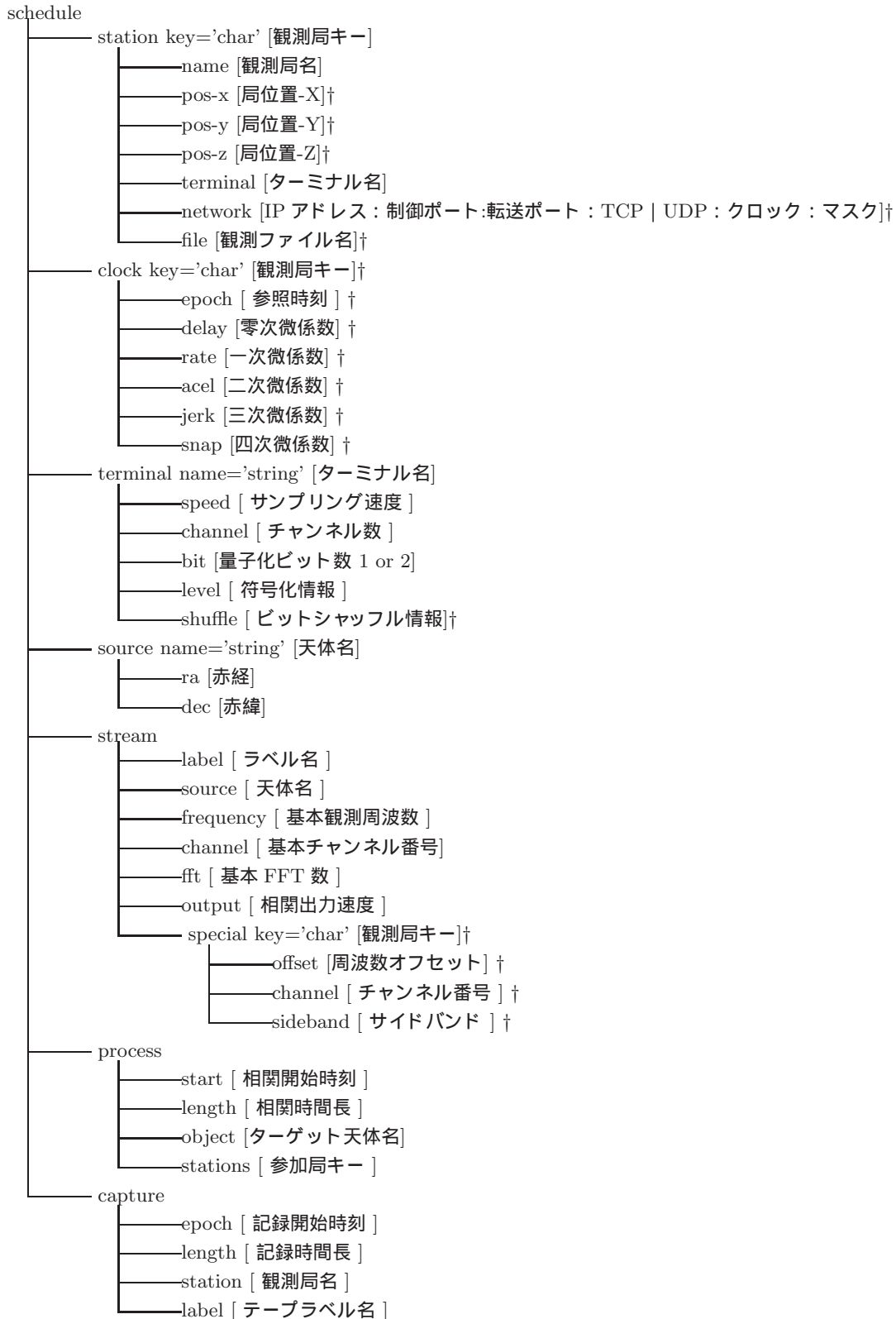
4.6 GICO3の相関処理でデータの並び

多チャンネルのデータを処理する場合には、観測ファイル中の各サンプルの並びは下図のようになっている必要があります。この並びとは異なる場合には shuffle 機能を使用してこの並びに一致させる必要があります。なお、各サンプルがどの数値に変換されるかは、schedule⇒terminal⇒levelにより決定されます。



4.7 記録・関連スケジュール [構造]

capture2000 と gico3_corr のスケジュールファイルは以下のような構造を持つ共通ファイルフォーマットを採用しています。ここで † 付きのものは省略可能となっており、省略された場合は未使用又はデフォルト値又はゼロが代入されます。また、process 内に station,clock,terminal,source,stream を記述することにより、process 毎に異なる条件で相関処理が可能です。



4.8 記録・相関スケジュール [サンプル]

```
<?xml version='1.0' encoding='UTF-8' ?>
<schedule>
  <terminal name='VERA'><speed> 32000000</speed><channel>02</channel><bit>2</bit><level>-1.5,-0.5,+0.5,+1.5</level></terminal>
  <station key='R'>
    <name>IRIKI </name><pos-x>-3521719.568700</pos-x><pos-y>+4132174.752800</pos-y><pos-z>+3336994.325500</pos-z><terminal>VERA</terminal>
  </station>
  <station key='I'>
    <name>ISHIGAKI</name><pos-x>-3263994.648300</pos-x><pos-y>+4808056.355800</pos-y><pos-z>+2619949.394400</pos-z><terminal>VERA</terminal>
  </station>
  <station key='M'>
    <name>MIZNA020</name><pos-x>-3857241.855200</pos-x><pos-y>+3108784.850900</pos-y><pos-z>+4003900.585800</pos-z><terminal>VERA</terminal>
  </station>
  <station key='G'>
    <name>OGASA20 </name><pos-x>-4491068.894000</pos-x><pos-y>+3481544.829500</pos-y><pos-z>+2887399.622700</pos-z><terminal>VERA</terminal>
  </station>
  <clock key='R'><epoch>2008/077 15:10:00</epoch><delay>+4.9000e-06</delay><rate>+2.0900e-12</rate></clock>
  <clock key='I'><epoch>2008/077 15:10:00</epoch><delay>+4.6000e-07</delay><rate>+1.7000e-12</rate></clock>
  <clock key='M'><epoch>2008/077 15:10:00</epoch><delay>-5.0700e-06</delay><rate>+0.0000e+00</rate></clock>
  <clock key='G'><epoch>2008/077 15:10:00</epoch><delay>+1.1080e-05</delay><rate>+2.7700e-12</rate></clock>
  <capture><epoch>2008/077 15:25:00</epoch><length>4776</length><station> IRIKI</station><label>USG00078</label></capture>
  <capture><epoch>2008/077 15:25:00</epoch><length>4776</length><station>ISHIGAKI</station><label>VER06430</label></capture>
  <capture><epoch>2008/077 15:25:00</epoch><length>4776</length><station>MIZNA020</station><label>VER05058</label></capture>
  <capture><epoch>2008/077 15:25:00</epoch><length>4776</length><station> OGASA20</station><label>VER05220</label></capture>
  <process>
    <source name='DUM'><ra>00h00m00.000000</ra><dec>+90d00'00.000000 </dec></source>
    <source name='1641+399'><ra>16h42m58.809950</ra><dec>+39d48'36.993990 </dec></source>
    <epoch>2008/077 15:25:00</epoch><length>0060</length><object>Multi</object><stations>RIMG</stations>
    <stream>
      <label>CH01</label><source> DUM</source><frequency>+22226000000</frequency><channel>01</channel>
      <fft>002048</fft><output>001</output>
    </stream>
    <stream>
      <label>CH02</label><source>1641+399</source><frequency>+22002000000</frequency><channel>02</channel>
      <fft>002048</fft><output>001</output>
    </stream>
    <station key='R'><file> IRIKI_2008077152500.raw</file></station>
    <station key='I'><file>ISHIGAKI_2008077152500.raw</file></station>
    <station key='M'><file>MIZNA020_2008077152500.raw</file></station>
    <station key='G'><file> OGASA20_2008077152500.raw</file></station>
  </process>
</schedule>
```

表 6: 付録 相関結果ファイルの旧フォーマット (Version 1.0)

Address	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f	
Header Region																	
0x000-0x00f	Magic Word [0x3ea2f983]			Header Version [0x01000000]			Software Version [int]			Sampling Speed [int]							
0x010-0x01f	Observed Sky Frequency [double]			Station-1 Name [char × 16]			FFT-Point [int]			Number of Sector [int]							
0x020-0x02f	Station-1 Position-X [double]			Station-1 Position-Y [double]			Free-Area [default : 0x00000000]										
0x030-0x03f	Station-1 Position-Z [double]			Station-2 Name [char × 16]			Station-2 Position-X [double]			Station-2 Position-Y [double]							
0x040-0x04f	Station-2 Position-Z [double]			Source Name [char × 16]			Free-Area [default : 0x00000000]										
0x050-0x05f	Source Position-Ra [double]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]							
0x060-0x06f	Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]							
0x070-0x07f	Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]							
0x080-0x08f	Correlation Start Sec [time-t]			Correlation Start NanoSec [int]			Correlation Stop Sec [time-t]			Correlation Stop NanoSec [int]							
0x090-0x09f	Station-1 Model Epoch Sec [time-t]			Station-1 Model Epoch NanoSec [int]			Station-1 Model Rate [double]			Station-1 Model Jerk [double]							
0x0a0-0x0af	Station-2 Model Epoch Sec [time-t]			Station-2 Model Epoch NanoSec [int]			Station-2 Model Rate [double]			Station-2 Model Jerk [double]							
0x0b0-0x0bf	Effective Integration Length [float]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]			Real[0] [float]			Imag[0] [float]							
0x0c0-0x0cf	Real[0] [float]			Imag[0] [float]			Real[1] [float]			Imag[1] [float]							
0x0d0-0x0df	Real[1] [float]			Imag[1] [float]			Real[2] [float]			Imag[2] [float]							
0x0e0-0x0ef	Real[2] [float]			Imag[2] [float]			Real[3] [float]			Imag[3] [float]							
0x0f0-0x0ff	Real[3] [float]			Imag[3] [float]			Real[4] [float]			Imag[4] [float]							
First Sector																	
0x100-0x10f	Correlation Start Sec [time-t]			Correlation Start NanoSec [int]			Correlation Stop Sec [time-t]			Correlation Stop NanoSec [int]							
0x110-0x11f	Station-1 Model Epoch Sec [time-t]			Station-1 Model Epoch NanoSec [int]			Station-1 Model Rate [double]			Station-1 Model Jerk [double]							
0x120-0x12f	Station-2 Model Epoch Sec [time-t]			Station-2 Model Epoch NanoSec [int]			Station-2 Model Rate [double]			Station-2 Model Jerk [double]							
0x130-0x13f	Effective Integration Length [float]			Free-Area [default : 0x00000000-0x00000000-0x00000000-0x00000000]			Real[0] [float]			Imag[0] [float]							
0x140-0x14f	Real[0] [float]			Imag[0] [float]			Real[1] [float]			Imag[1] [float]							
0x150-0x15f	Real[1] [float]			Imag[1] [float]			Real[2] [float]			Imag[2] [float]							
0x160-0x16f	Real[2] [float]			Imag[2] [float]			Real[3] [float]			Imag[3] [float]							
0x170-0x17f	Real[3] [float]			Imag[3] [float]			Real[4] [float]			Imag[4] [float]							
0x180-0x18f	Real[4] [float]			Imag[4] [float]			Real[5] [float]			Imag[5] [float]							
.....							
.....							
.....							
.....							
.....							
Second Sector																	
.....	Correlation Start Sec [time-t]			Correlation Start NanoSec [int]			Correlation Stop Sec [time-t]			Correlation Stop NanoSec [int]							
.....	Station-1 Geometry-Sec [time-t]			Station-1 Geometry-NanoSec [int]			Station-1 Geometry-Delay [double]			Station-1 Geometry-Delay [double]							
.....	Station-2 Geometry-Sec [time-t]			Station-2 Geometry-NanoSec [int]			Station-2 Geometry-Delay [double]			Station-2 Geometry-Delay [double]							
.....	Station-3 Geometry-Sec [time-t]			Station-3 Geometry-NanoSec [int]			Station-3 Geometry-Delay [double]			Station-3 Geometry-Delay [double]							
.....	Station-4 Geometry-Sec [time-t]			Station-4 Geometry-NanoSec [int]			Station-4 Geometry-Delay [double]			Station-4 Geometry-Delay [double]							
.....	Station-5 Geometry-Sec [time-t]			Station-5 Geometry-NanoSec [int]			Station-5 Geometry-Delay [double]			Station-5 Geometry-Delay [double]							

表 7: 付録 相関結果ファイルの現フォーマット (Version 1.1)

Address	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
Header Region																
0x000-0x00f	Magic Word [0x3ea2f983]			Header Version [0x01010000]			Software Version [int]			Sampling Speed [int]						
0x010-0x01f	Observed Sky Frequency [double]			Station-1 Name [char × 16]			FFT-Point [int]			Number of Sector [int]						
0x020-0x02f	Station-1 Position-X [double]			Station-1 Position-Y [double]			Free-Area [default : 0x00000000]									
0x030-0x03f	Station-1 Position-Z [double]			Station-2 Name [char × 16]			Station-2 Position-X [double]			Station-2 Position-Y [double]						
0x040-0x04f	Station-2 Position-Z [double]			Source Name [char × 16]			Station-1 Clock Epoch NanoSec [int]			Station-1 Clock Delay [double]						
0x050-0x05f	Station-1 Clock Rate [double]			Station-1 Clock Jerk [double]			Station-1 Clock Accel [double]			Station-2 Clock Epoch NanoSec [int]						
0x060-0x06f	Station-2 Clock Rate [double]			Station-2 Clock Jerk [double]			Station-2 Clock Accel [double]			Station-2 Clock Delay [double]						
0x070-0x07f	Station-1 Clock Jerk [double]			Station-2 Clock Jerk [double]			Correlation Start NanoSec [int]			Correlation Stop NanoSec [int]						
0x080-0x08f	Station-2 Clock Jerk [double]			Station-1 Model Rate [double]			Station-1 Model Jerk [double]			Station-1 Model Accel [double]						
0x090-0x09f	Station-2 Model Rate [double]			Station-2 Model Jerk [double]			Station-2 Model Accel [double]			Station-1 Model Delay [double]						
0x0a0-0x0af	Station-1 Model Rate [double]			Station-1 Model Jerk [double]			Station-1 Model Accel [double]			Station-2 Model Delay [double]						
0x0b0-0x0bf	Station-2 Model Rate [double]			Station-2 Model Jerk [double]			Station-2 Model Accel [double]			Station-1 Model Snap [double]						
0x0c0-0x0cf	Station-1 Model Rate [double]			Station-1 Model Jerk [double]			Station-1 Model Accel [double]			Station-2 Model Snap [double]						
0x0d0-0x0df	Station-2 Model Rate [double]			Station-2 Model Jerk [double]			Station-2 Model Accel [double]			Station-1 Model Delay [double]						
0x0e0-0x0ef	Station-1 Model Rate [double]			Station-1 Model Jerk [double]			Station-1 Model Accel [double]			Station-2 Model Delay [double]						
0x0f0-0x0ff	Station-2 Model Rate [double]			Station-2 Model Jerk [double]			Station-2 Model Accel [double]			Station-1 Model Snap [double]						
First Sector																
0x100-0x10f	Correlation Start Sec [time_t]			Correlation Start NanoSec [int]			Correlation Stop Sec [time_t]			Correlation Stop NanoSec [int]						
0x110-0x11f	Station-1 Model Epoch Sec [time_t]			Station-1 Model Epoch NanoSec [int]			Station-1 Model Delay [double]			Station-1 Model Accel [double]						
0x120-0x12f	Station-1 Model Rate [double]			Station-1 Model Jerk [double]			Station-1 Model Snap [double]			Station-2 Model Delay [double]						
0x130-0x13f	Station-2 Model Rate [double]			Station-2 Model Jerk [double]			Station-2 Model Accel [double]			Station-2 Model Delay [double]						
0x140-0x14f	Station-1 Model Rate [double]			Station-1 Model Jerk [double]			Station-1 Model Accel [double]			Station-2 Model Delay [double]						
0x150-0x15f	Station-2 Model Rate [double]			Station-2 Model Jerk [double]			Station-2 Model Accel [double]			Station-1 Model Delay [double]						
0x160-0x16f	Station-1 Model Rate [double]			Station-1 Model Jerk [double]			Station-1 Model Accel [double]			Station-2 Model Delay [double]						
0x170-0x17f	Effective Integration Length [float]			Free-Area [default : 0x00000000-0x00000000-0x00000000]			Real[0] [float]			Imag[0] [float]						
0x180-0x18f	Real[1] [float]			Imag[1] [float]			Real[2-1] [float]			Imag[2-1] [float]						
.....	Real[2-2] [float]			Imag[2-2] [float]			Real[2-3] [float]			Imag[2-3] [float]						
.....	Real[2-4] [float]			Imag[2-4] [float]			Real[2-5] [float]			Imag[2-5] [float]						
.....	Real[2-6] [float]			Imag[2-6] [float]			Real[2-7] [float]			Imag[2-7] [float]						
.....	Real[2-8] [float]			Imag[2-8] [float]			Real[2-9] [float]			Imag[2-9] [float]						
Second Sector																
.....	Correlation Start Sec [time_t]			Correlation Start NanoSec [int]			Correlation Stop Sec [time_t]			Correlation Stop NanoSec [int]						
.....	Station-1 Geometry-Sec [time_t]			Station-1 Geometry-NanoSec [int]			Station-1 Geometry-Delay [double]			Station-1 Geometry-Accel [double]						
.....	Station-2 Geometry-Sec [time_t]			Station-2 Geometry-NanoSec [int]			Station-2 Geometry-Delay [double]			Station-2 Geometry-Accel [double]						
.....	Station-3 Geometry-Sec [time_t]			Station-3 Geometry-NanoSec [int]			Station-3 Geometry-Delay [double]			Station-3 Geometry-Accel [double]						
.....	Station-4 Geometry-Sec [time_t]			Station-4 Geometry-NanoSec [int]			Station-4 Geometry-Delay [double]			Station-4 Geometry-Accel [double]						