SCHEDULES FOR PULSAR OBSERVATIONS WITH THE SARDINIA RADIO TELESCOPE

Release 0.4

Author: Alessandro Corongiu
Release date: 05/08/2019

Release notes

Issue	Release date	What's new
01	12/09/2018	Document creation as a plain extraction of relevant sections from a more general pourpose document.
02	06/12/2018	Contents revision and organisation
03	30/05/2019	Keyword Receiver modified after removal of "Maccaferri" system. Previous setup files will have problems whenever the L-Band is needed, otherwise they're ok.
04	05/08/2019	Minor revision

Document presentation

This document illustrates the structure and syntax of schedules for pulsar observation at the Sardinia Radio Telescope. A tool is also described for helping the user to build the so called setup file, whose concept and structure will be detailed in a dedicated section.

Before reading this document, the reader is strongly encouraged to have already been introduced to the Sardinia Radio Telescope instrumentation, with particular attention to all devices involved in pulsar observations, as well as to the related control softwares. All these concepts are considered already known by the reader and are not illustrated in this document.

Current version presentation

This is the first major release of this document, obtained through a deep revision and reorganization of the previous one.

Index

1 - Settings concepts

Settings

Settings are statements of the form

```
[keyword] = [value(s)]
```

where `keyword` is a string that identifies a feature, and `parameter(s)` is a string containing the value, a number or a word, that describe how that feature has to be tuned. Examples:

```
SamplingTime = 125
ActiveSurface = PARABOLIC
```

Most features are completely set by specifying a single value, but some of them require values for more than one parameter. In this case, `[value(s)]` is a sequence of comma separated single values:

```
Receiver = P-BAND,LINEAR,WIDE
```

In such settings, the meaning and syntax of the first parameter is always the same. The meaning of the next ones may vary, but it's always related to the value of the first (or a previous one). Let's consider the following example:

```
Receiver = P-BAND,LINEAR,WIDE
Receiver = C-BAND,5950.0,730.0
```

The first row sets the `P-BAND` receiver, whose tunable parameters are the polarization type (`LINEAR`) and the frequency band (`WIDE`, see appendix Y for its meaning). The second row sets the `C-BAND` receiver, whose tunable parameters are the width of the frequency band to be acquired (`730.0MHz`) and the local oscillator frequency (`5950.0MHz`).

The number of parameters may also vary for a given setting:

```
Source = B0329+54
Source = J1910-5959A,J2000,19:11:42.75562,-59:58:26.9029
```

Same syntax in schedules and setup files

The syntax for all settings is exactly the same in both setup and schedule files.

Settings groups

Settings can be divided in groups, each of them related to a specific aspect of the observations. These groups are:

1) `project` settings, i.e. informations that identify and describe the scientific project and the observers
2) `observation` settings, i.e. settings related to the execution of the single observation

3) `antenna` settings, i.e. parameters for setting up and pointing the antenna
4) `backend` settings, i.e. parameters for setting up the data acquisition.

Settings in the last group need to be further divided in sub-groups. The first of them, `generic backend` settings, includes settings whose meaning is (nearly) the same for all backends; the others are backend specific settings, i.e. settings related to the hardware and/or the software of each data acquisition device. Because of each backend's peculiarities, a setting belonging to the `generic backend` sub-group may or may not directly/explicitly be set, or even not settable since a backend may not have a given functionality.

The global group of settings required by a specific backend contains those generic backend settings that can directly/explitiсly be set, and the backend's specific settings.

Here below are enumerated all settings, accordingly to the group they belong and introduced by the keyword that identifies them.

Project settings:

1) `ObserverName`: name(s) of the observer(s)
2) `ProjectCode`: code assigned by the telescope TAC to the scientific project
3) `ProjectName`: name/title/short description of the scientific project

Observation settings:

1) `ObsLength`: duration of the observation
2) `ObsStartHour`: UTC start time of the observation
3) `ObsStopHour`: UTC stop time of the observation
4) `Setup`: file containing all settings
5) `WaitBeforeStart`: time delay before the start of the data acquisition

Antenna settings:

1) `ActiveSurface`: shape and adaptive behaviour of the primary reflector
2) `AzSector`: azimuth sector for pointing the antenna
3) `CalSource`: switches ON/OFF the calibration signal source
4) `Receiver`: receiver, and related settings
5) `Source`: celestial object to be pointed

Backend settings: generic backend

1) `Bandwidth`: width in MHz of the backend's frequency band
2) `BitsPerSample`: number of bits to represent each data sample
3) `DedispersionMode`: coherent or incoherent signal dedispersion
4) `DispersionMeasure`: dispersion measure value for coherent dedispersion
5) `Frequency`: sky frequency that falls at the center of the backend's frequency band
6) `Mode`: data acquisition mode
7) `NumberOfChannels`: number of frequency channels
8) `NumberOfPols`: number of polarizations to be acquired
9) `ProfileBins`: number of bins to represent the pulsar profile
10) `SamplingTime`: signal's sampling time

11) SubintTime: pulsar folding mode subintegration time

Backend settings: Pulsar Digital Filterbank (DFB) custom settings

1) ChannelRange: channel range to be acquired
2) ConfigFile: DFB configuration file name
3) CyclePeriod: DFB cycle period
4) MaxFileLength: search mode maximum file length
5) SamplesSubint: number of samples in search mode data blocks
6) TimeConstant: time to calculate the average bandpass
7) WriteFile: data written/not written on files

Backend settings: ROACH backend custom settings

1) AdvancedOptions: dspsr (folding) or digifil (search) not standard/common options

Settings' device indication

For each antenna and backend settings, the device to which they are related must also be indicated. This is done by prepending to the setting the device name, followed by the ">" character:

```
[device name] > [setting]
```

Example:

```
DFB > ConfigFile = pdfb4_1024_512_2048
```

Device names are ANTENNA, for the antenna, DFB for the Pulsar Digital Filterbank, ROACH for the ROACH backend.

Device enabling

The backends fleet needed by different observations may differ, in number and/or elements. This information is coded by the following statements:

```
[device name] > ENABLE
[device name] > DISABLE
```

These statements respectively enable and disable the data acquisition by a given backend. Because seadas is structured for piggy back data acquisitions (a not yet available functionality) it's required to explicitly enable the use of the antenna with the statement:

```
ANTENNA > ENABLE
```

2 - Files creation and editing

Both schedule and setup files are plain ASCII files: they can be created, written and edited by using common plain text editors (vi, nano, emacs, etc.).

They do not need any special hidden character, e.g. tabs, for giving them the requested format.

Comments are also allowed. The character to be used is the hash '#'. In any line, whatever is placed at the right of this character is considered a comment.

Both schedules and setup files must be placed in a specific directory, namely:

`/home/corr/scheds/[Project code]`

or any subfolder of it. `[Project code]` is the code assigned to the project.

3 - Schedule files structure

A schedule is organised in rows, each of them containing all necessary settings to execute the observation of a single object. Therefore, it appears as follows:

```
[Settings to observe object #1]
[Settings to observe object #2]
...
...
[Settings to observe object #N]
```

Settings can be also put in another file, in this document referred to as *setup* file, with indication in the schedule line of the name and path of such file. The setting `WaitBeforeStart` is the only one that cannot be put in a setup file, since it is related to the execution of the schedule. The presence of setting `Setup` in a setup file will be discussed in section 4.

If a given setting is expressed in both the setup file and schedule line, the value indicated in the schedule line always overrides the one in the setup file. Such a policy has a practical impact in both the preparation and the management of the observing session. The observer can prepare only one setup file containing the most common value for each setting in the session, while in the schedule line those settings only appear for which a custom value is required for a given observation. The most common and obvious example of a setting that changes from one observation to another is the name of the source (and/or its coordinates). In such a case, a schedule has the following appearance:

```
[name/coords of object #1];[common setup file];[custom settings for obj #1]
[name/coords of object #2];[common setup file];[custom settings for obj #2]
...
[name/coords of object #N];[common setup file];[custom settings for obj #N]
```

In the example above, the bold face words indicate the common setting to all observation, namely the same setup file.

Last example also introduces the semicolon as field separator, i.e. the character that separates two different settings. Each schedule line appears therefore as follows:

```
[setting #1] ; [setting #2] ;...; [setting #N];
```

4 - Setup files structure

Setup files are organised in rows, each of them containing a single setting. Consequently, they appear as follows:

```
[setting #1]
[setting #2]
...
...
...
[setting #N]
```

No termination line character is required.

All possible settings have been listed in section 1. If they are put in a single file, such a file would have a considerable number of lines, maybe too much to be easy managed with a common text editor and/or quickly read. In order to avoid too long files, setup files can be hierarchically organised: a *main* file containing the session and observations settings, and *device* setup files, containing settings for a specific device. An example may better clarify the concept. Consider the setup file here below:

```
ProjectCode = 22-19
ProjectName = My fantastic project
ObserverName = Me myself and I
ANTENNA > ENABLE
ANTENNA > Receiver = LP-DUAL,LINEAR,NARROW,LINEAR,WIDE,2316.000
ANTENNA > ActiveSurface = RECEIVER-DEFAULT
ANTENNA > CalSource = OFF
ROACH > ENABLED
ROACH > Mode = SEARCH
ROACH > Frequency = 336.000
ROACH > Bandwidth = 80.000
ROACH > NumberOfChannels = 1280
ROACH > NumberOfPols = 1
ROACH > BitsPerSample = 1
ROACH > SamplingTime = 125.000
ROACH > DedispersionMode = COHERENT
ROACH > DispersionMeasure = 37.456
ROACH > AdvancedOptions = none
DFB > ENABLED
DFB > Mode = SEARCH
DFB > Frequency = 1548.000
DFB > NumberOfPols = 1
DFB > BitsPerSample = 1
DFB > SamplingTime = 125.000
DFB > WriteFile = NO
DFB > SamplesSubint = 4096
DFB > TimeConstant = 1
DFB > ChannelRange = all
DFB > MaxFileLength = 3600
DFB > ConfigFile = srch_1024_2048
DFB > CyclePeriod = 10.000
```

If splitted, its contents become:

```
ProjectCode = 22-19
ProjectName = My fantastic project
ObserverName = Me myself and I
ANTENNA > ENABLE
```

```
ANTENNA > Setup = 22-19/antenna-example.stp
ROACH > ENABLED
ROACH > Setup = 22-19/roach-example.stp
DFB > ENABLED
DFB > Setup = 22-19/dfb-example.stp
```

while ANTENNA, ROACH and DFB settings are moved to files `antenna-example.stp`, `roach-example.stp` and `dfb-example.stp` respectively. We recall that setup files must be placed in the same directory of the schedule file, i.e. `/home/corr/scheds/[project code]`, where `[project code]` is 22-19 in this example, or any subfolder of it. Their contents are reported here below.

file `antenna-example.stp`:

```
Receiver = LP-DUAL,LINEAR,NARROW,LINEAR,NARROW,2316.000
ActiveSurface = RECEIVER-DEFAULT
CalSource = OFF
```

file `roach-example.stp`:

```
Mode = SEARCH
Frequency = 336.000
Bandwidth = 80.000
NumberOfChannels = 1280
NumberOfPols = 1
BitsPerSample = 1
SamplingTime = 125.000
DedispersionMode = COHERENT
DispersionMeasure = 37.456
AdvancedOptions = none
```

file `dfb-example.stp`:

```
Mode = SEARCH
Frequency = 1548.000
NumberOfPols = 1
BitsPerSample = 1
SamplingTime = 125.000
WriteFile = NO
SamplesSubint = 4096
TimeConstant = 1
ChannelRange = all
MaxFileLength = 3600
ConfigFile = srch_1024_2048
CyclePeriod = 10.000
```

These examples also show that devices' settings do not need the `[device] >` preamble, if they are declared in devices setup files. The following declarations are both correct:

```
[keyword] = [value(s)]
[device name] > [keyword] = [value(s)]
```

The user is totally free to choose either organisation for the setup file(s) of his/her observations. If unsure, the user should use the form:

```
[device name] > [keyword] = [value(s)]
```

a form that is always legal in any setup file.

5 - Setup file templates

In this section templates for setup files are provided. In order to cover all possible receiver/backend(s) combinations, separate templates are provided for the main and devices setup files. All statements will have the `[device name] > [setting]` syntax in all cases for which this form is allowed, so that they can be copied and pasted in both a single global file or in separated devices' dedicated files.

Main setup file templates

1) ROACH backend enabled, DFB backend disabled

```
ProjectCode = 22-19
ProjectName = My fantastic project
ObserverName = Me myself and I
ANTENNA > ENABLE
ANTENNA > Setup = 22-19/antenna-example.stp
ROACH > ENABLED
ROACH > Setup = 22-19/roach-example.stp
DFB > DISABLED
```

2) ROACH backend disabled, DFB backend enabled

```
ProjectCode = 22-19
ProjectName = My fantastic project
ObserverName = Me myself and I
ANTENNA > ENABLE
ANTENNA > Setup = 22-19/antenna-example.stp
ROACH > DISABLED
DFB > ENABLED
DFB > Setup = 22-19/dfb-example.stp
```

3) ROACH backend enabled, DFB backend enabled

```
ProjectCode = 22-19
ProjectName = My fantastic project
ObserverName = Me myself and I
ANTENNA > ENABLE
ANTENNA > Setup = 22-19/antenna-example.stp
ROACH > ENABLED
ROACH > Setup = 22-19/roach-example.stp
DFB > ENABLED
DFB > Setup = 22-19/dfb-example.stp
```

ANTENNA setup file templates

Note. Settings for keywords `Source` and `AzSector` are not included in the following templates, because it's strongly recommended to use them in only in schedules. Despite being settings for the `ANTENNA` device, they can be declared in schedules without the "`ANTENNA >`" preamble.

1) P-BAND receiver

```
Receiver = P-BAND,LINEAR,NARROW
ActiveSurface = PARABOLIC
CalSource = OFF
```

2) L-BAND receiver

```
Receiver = L-BAND,LINEAR,WIDE,2316.000
ActiveSurface = PARABOLIC
CalSource = OFF
```

3) LP-DUAL receiver

```
Receiver = LP-DUAL,LINEAR,NARROW,LINEAR,NARROW,2316.000
ActiveSurface = RECEIVER-DEFAULT
CalSource = OFF
```

4) C-BAND receiver

```
Receiver = C-BAND,5750.0,730.000
ActiveSurface = SHAPED
CalSource = OFF
```

5) K-BAND receiver

```
Receiver = K-BAND,18000.0,730.000
ActiveSurface = SHAPED
CalSource = OFF
```

DFB setup file templates

Notes

1) In this backend the bandwidth (keyword `Bandwidth`), the number of channels (keyword `NumberOfChannels`) and number of profile bins (keyword `ProfileBins`) are not directly set, since these informations are in the DFB configuration file, hence they're through the keyword `ConfigFile`.
2) The DFB cannot perform coherent dedispersions. Therefore, keywords `DedispersionMode` and `DispersionMeasure` are not available for this backend.

1) FOLD mode

```
DFB > Mode = FOLD
DFB > Frequency = 256.000
DFB > SubintTime = 10
DFB > WriteFile = YES
DFB > ChannelRange = all
DFB > ConfigFile = pdfb4_1024_512_1024
DFB > CyclePeriod = 10.000
```

2) PSRCAL mode

```
DFB > Mode = PSRCAL
```

```
DFB > Frequency = 1548.000
DFB > SubintTime = 10
DFB > WriteFile = YES
DFB > ChannelRange = all
DFB > ConfigFile = pdfb4_1024_512_2048
DFB > CyclePeriod = 10.000
```

3) SEARCH mode

```
DFB > Mode = SEARCH
DFB > Frequency = 256.000
DFB > NumberOfPols = 1
DFB > BitsPerSample = 2
DFB > SamplingTime = 125.000
DFB > WriteFile = YES
DFB > SamplesSubint = 4096
DFB > TimeConstant = 1
DFB > ChannelRange = all
DFB > MaxFileLength = 0
DFB > ConfigFile = srch_512_1024
DFB > CyclePeriod = 10.000
```

4) SEARCHSET mode

```
DFB > Mode = SEARCHSET
DFB > Frequency = 1548.000
DFB > NumberOfPols = 1
DFB > BitsPerSample = 2
DFB > SamplingTime = 125.000
DFB > SamplesSubint = 4096
DFB > TimeConstant = 1
DFB > ChannelRange = all
DFB > MaxFileLength = 0
DFB > ConfigFile = srch_512_1024
DFB > CyclePeriod = 10.000
```

ROACH setup file templates

Notes

1) Coherent dedispersion is availabe in `FOLD` and `SEARCH` mode only; therefore, keyword `DedispersionMode` is ignored in `BASEBAND` and `PSRCAL` mode.
2) The value for dispersion measure value can be specified by the user in `SEARCH` mode only; therefore, keyword `DispersionMeasure` is available in this mode only.
3) In folding mode, the coherent dedispersion is computed for the value of the dispersion measure indicated in the pulsar's ephemeris; therefore, keyword `DispersionMeasure` is ignored in this mode. If a custom value is required, the user should produce custom ephemeris and put them in `psrcat`.

1) FOLD mode

```
ROACH > Mode = FOLD
ROACH > Frequency = 336.000
ROACH > Bandwidth = 80.000
ROACH > NumberOfChannels = 5120
ROACH > SubintTime = 10
ROACH > ProfileBins = 512
```

```
ROACH > DedispersionMode = INCOHERENT
ROACH > AdvancedOptions = none
```

2) PSRCAL mode

```
ROACH > Mode = PSRCAL
ROACH > Frequency = 336.000
ROACH > Bandwidth = 80.000
ROACH > NumberOfChannels = 5120
ROACH > SubintTime = 10
ROACH > ProfileBins = 512
```

3) SEARCH mode, INCOHERENT dedispersion

```
ROACH > Mode = SEARCH
ROACH > Frequency = 336.000
ROACH > Bandwidth = 80.000
ROACH > NumberOfChannels = 5120
ROACH > NumberOfPols = 4
ROACH > BitsPerSample = 8
ROACH > SamplingTime = 125.000
ROACH > DedispersionMode = INCOHERENT
ROACH > AdvancedOptions = none
```

4) SEARCH mode, COHERENT dedispersion

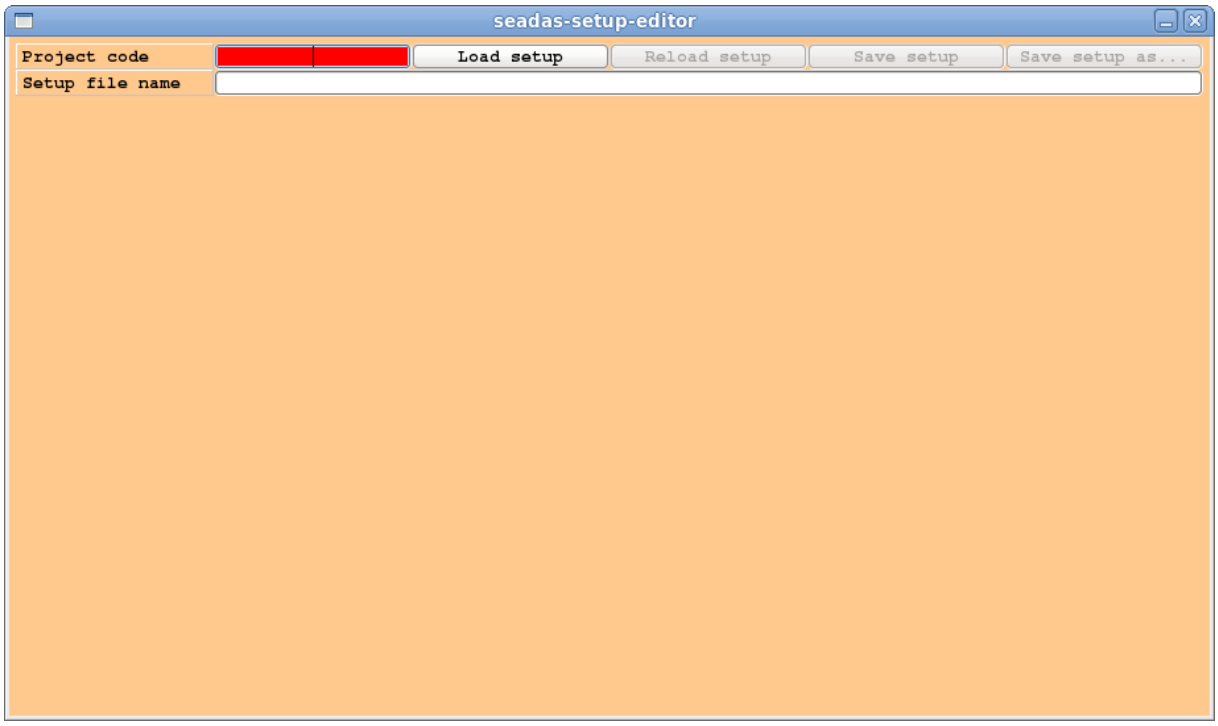
```
ROACH > Mode = SEARCH
ROACH > Frequency = 336.000
ROACH > Bandwidth = 80.000
ROACH > NumberOfChannels = 5120
ROACH > NumberOfPols = 4
ROACH > BitsPerSample = 8
ROACH > SamplingTime = 125.000
ROACH > DedispersionMode = COHERENT
ROACH > DispersionMeasure = 37.456
ROACH > AdvancedOptions = none
```

5) BASEBAND mode

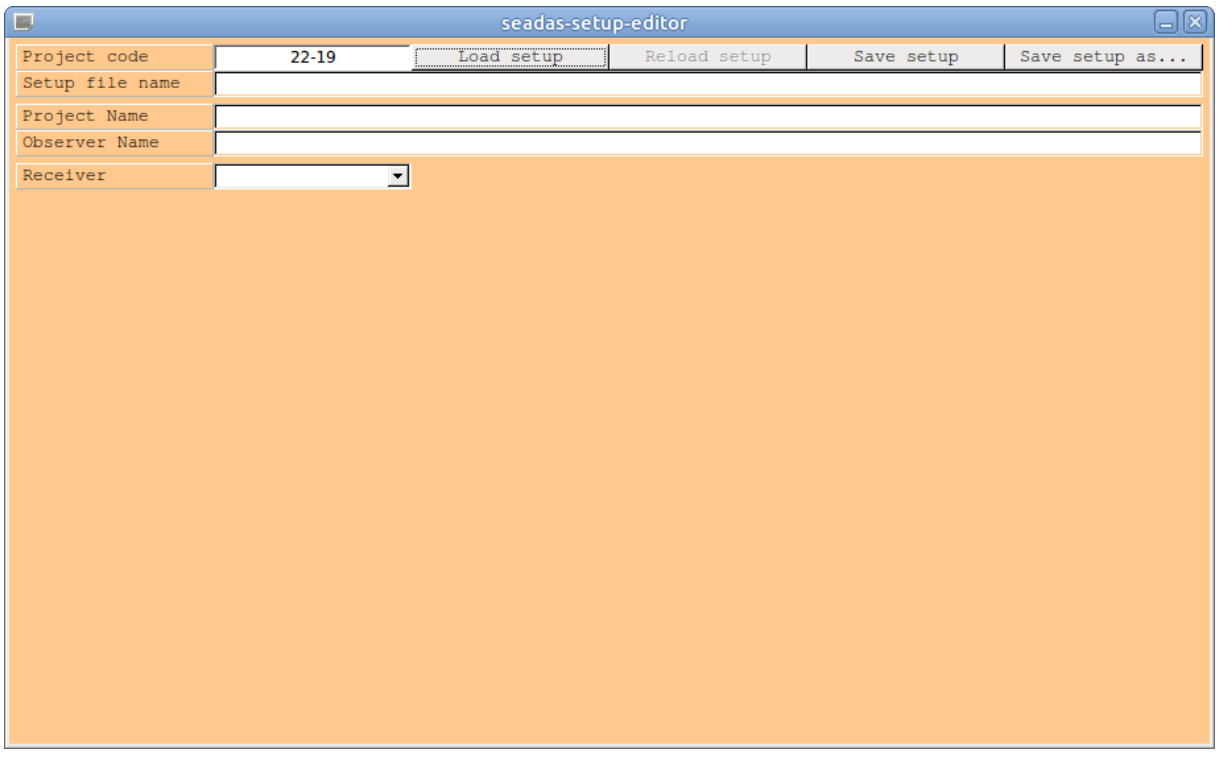
```
ROACH > Mode = BASEBAND
ROACH > Frequency = 336.000
ROACH > Bandwidth = 80.000
```

6 - Setup files editor

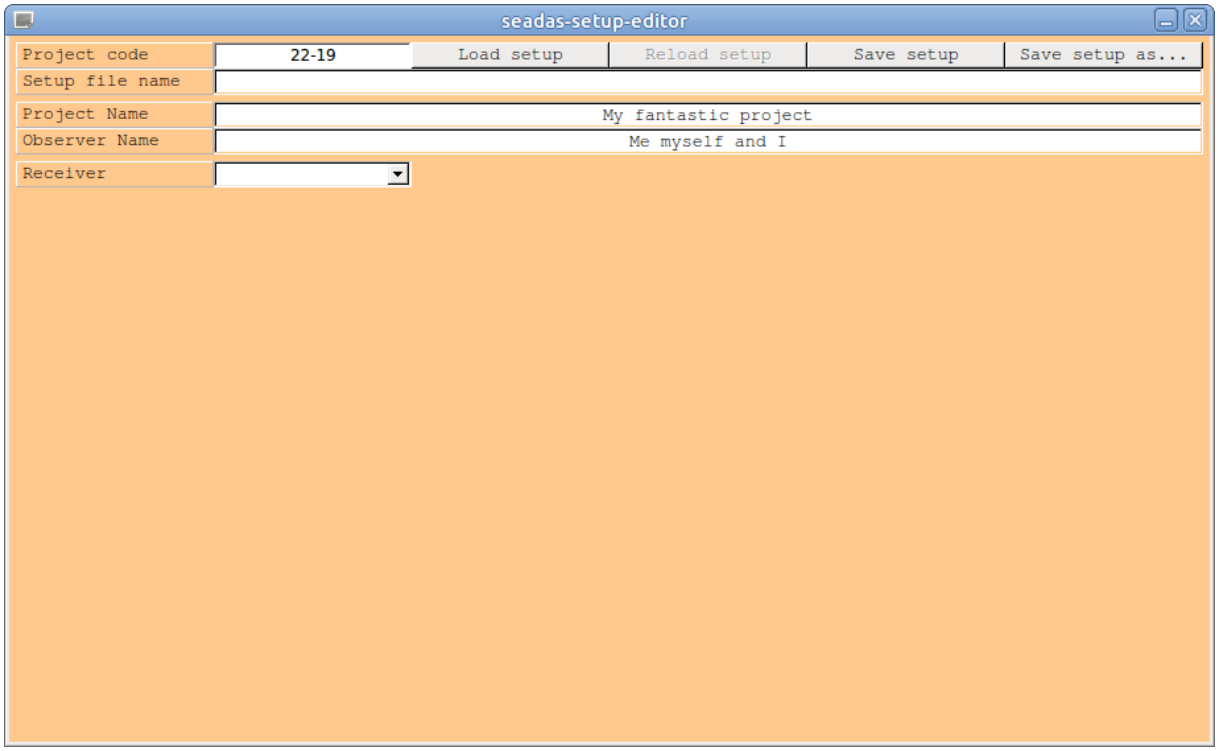
There is a much easier way for creating a setup file. A tool named *seadas-setup-editor* allows to create a setup file from scratch, and modify an already existing one. In the same computer where seadas runs (currently viewer01), open a terminal and type at prompt the command *seadas-setup-editor*. The following window pops up:



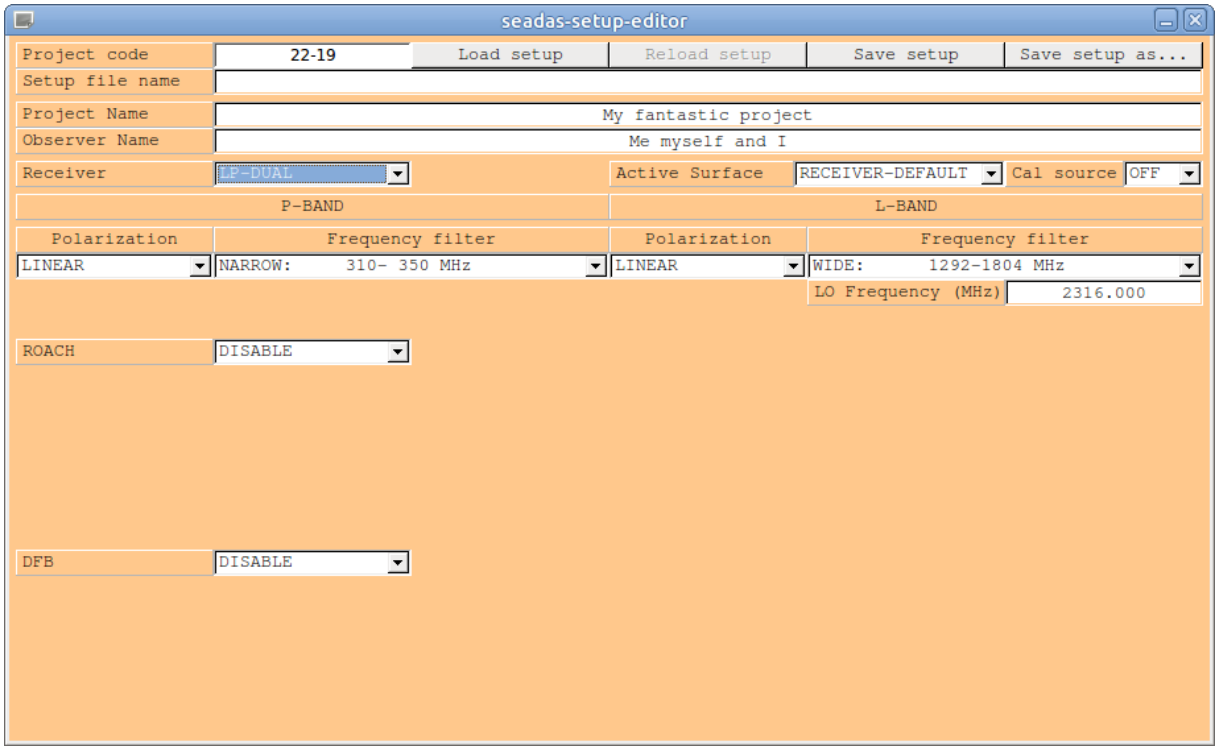
The only active field is the *Project code* one. Type there your code, the press return. Fields appear for entering the project and observer names, and the combo box for enabling the antenna and entering the related parameters:



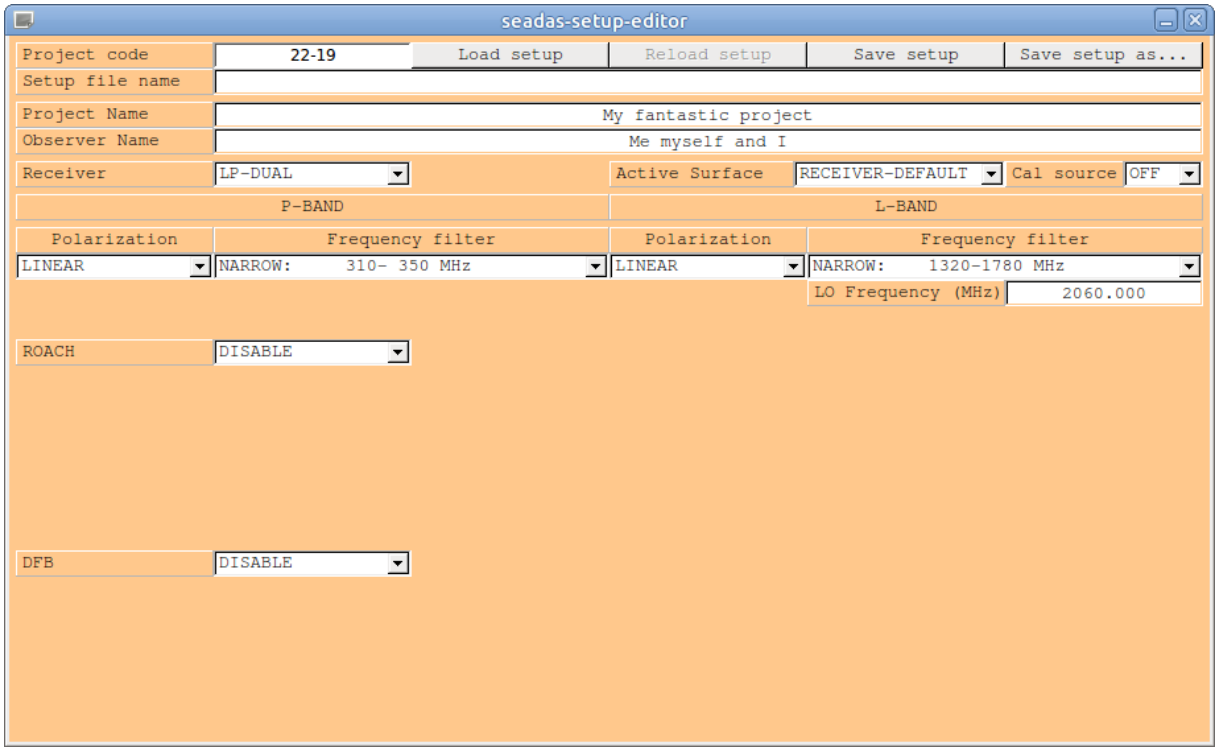
Fill the Project and Observer name fields:



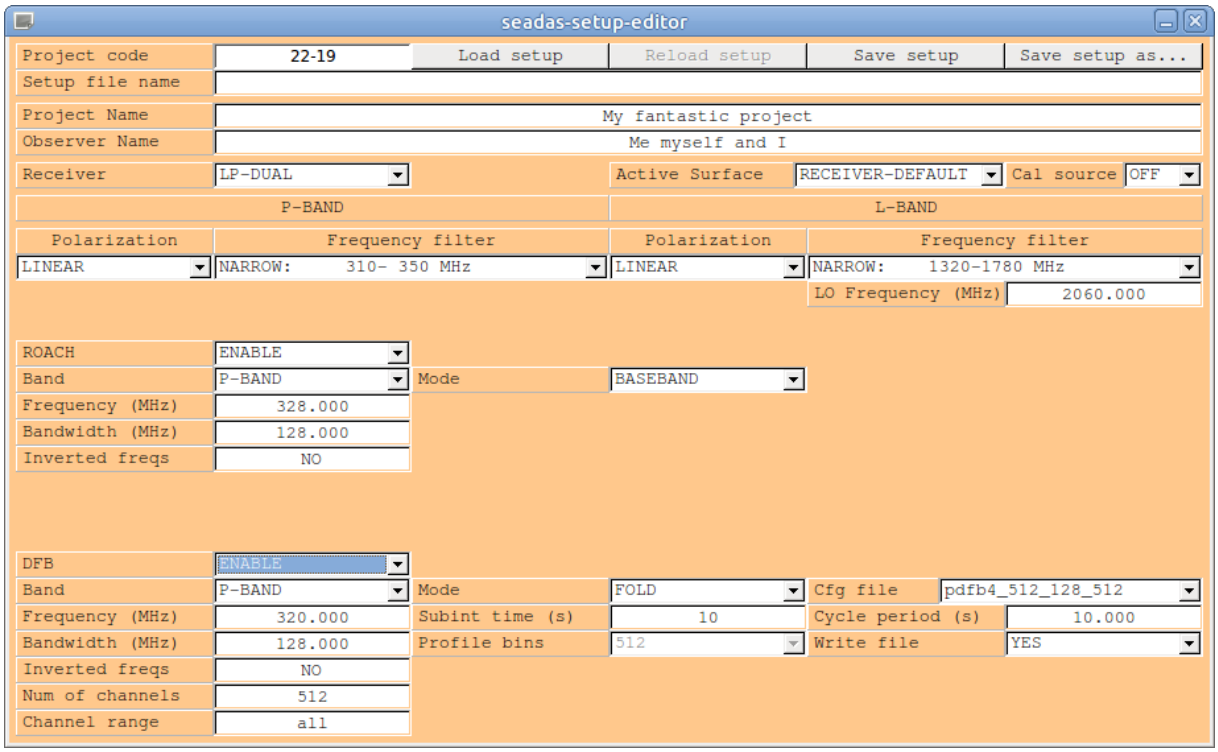
Now select the requested receiver through the *Receiver* combo box:



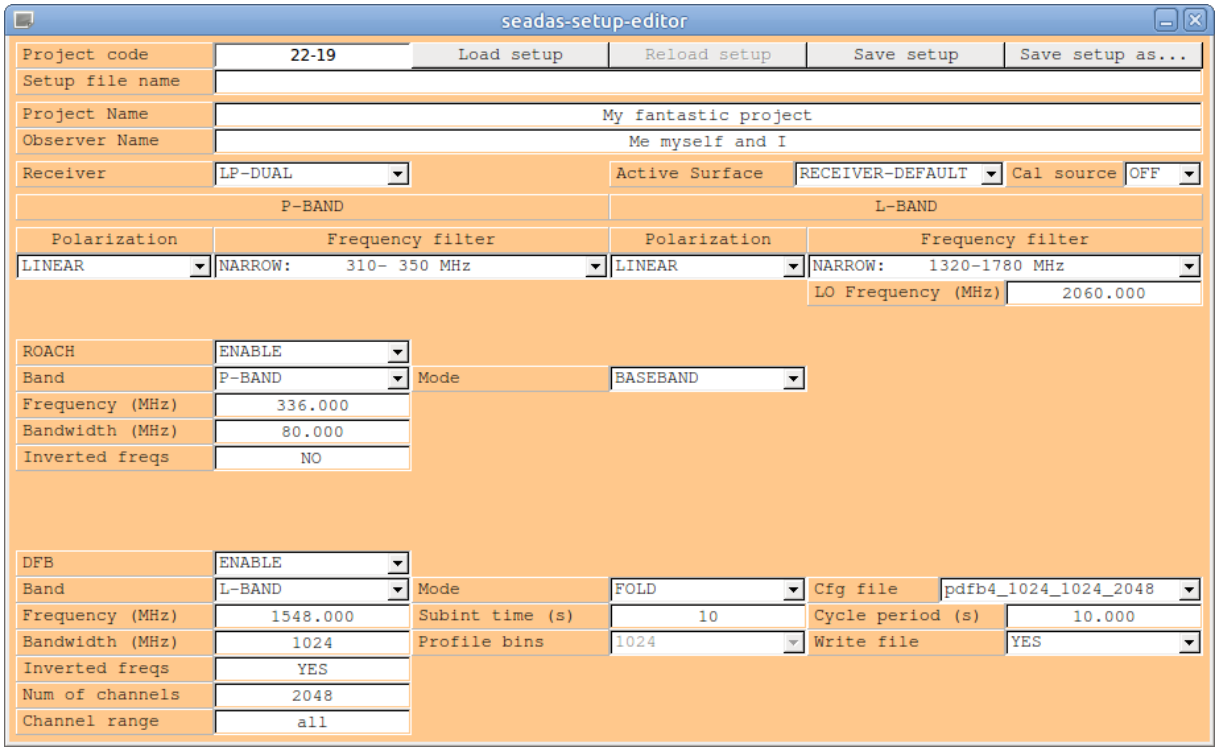
Now select the requested settings for the selected receiver:



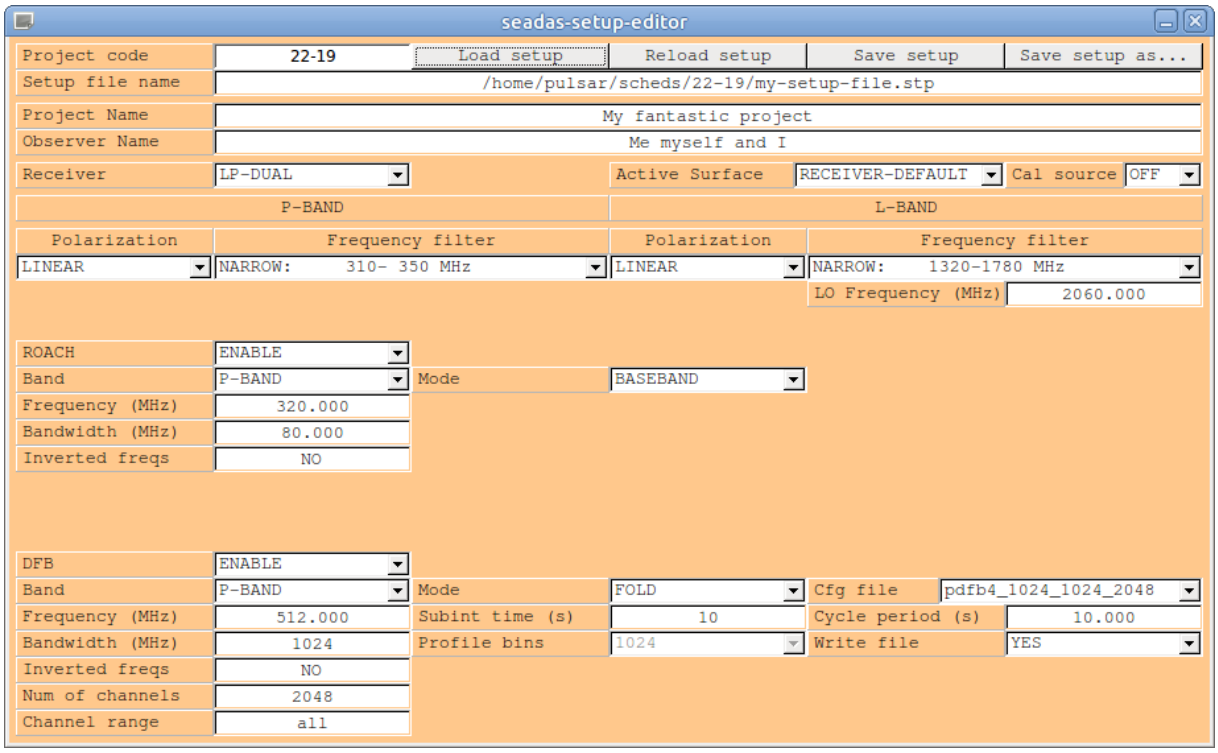
Enable the requested backends by selecting the item *ENABLE* in the combo box indicated by their names:



Now select the various setting for each backend:



Once done, click on the button *Save setup* and your setup file is saved in the directory */home/corr/setup/[project code]* (22-19 in this example). The chosen name for the setup file is displayed in the *Setup file name* field:



and the resulting setup file is this:

```
#########################################################################
# Setup file created for project 22-19 on 2019 ago 05 at UTC 15:14:00 #
#########################################################################
ProjectCode = 22-19
ProjectName = My fantastic project
ObserverName = Me myself and I
ANTENNA > ENABLE
ANTENNA > Receiver = LP-DUAL,LINEAR,NARROW,LINEAR,NARROW,2060.000
ANTENNA > ActiveSurface = RECEIVER-DEFAULT
ANTENNA > CalSource = OFF
ROACH > ENABLE
ROACH > Mode = BASEBAND
ROACH > Frequency = 336.000
ROACH > Bandwidth = 80.000
ROACH > InvertedFreqs = NO
DFB > ENABLE
DFB > Mode = FOLD
DFB > Frequency = 1548.000
DFB > InvertedFreqs = YES
DFB > SubintTime = 10
DFB > ChannelRange = all
DFB > ConfigFile = pdfb4_1024_1024_2048
DFB > CyclePeriod = 10.000
DFB > WriteFile = YES
#########################################################################
#                           end of setup file                           #
#########################################################################
```

7 - Settings analytic list

This section presents the syntax for each setting, identified by the related keyword.

Project settings

`ObserverName`: name(s) of the observer(s)

Syntax:

1) `ObserverName = [observer name(s)]`
`[observer name(s)]`: a string with the name(s) of the observer(s)

`ProjectCode`: code assigned by the telescope TAC to the scientific project

Syntax:

1) `ProjectCode = [project code]`
`[project code]`: the project code assigned by the TAC

`ProjectName`: name/title/short description of the scientific project

Syntax:

1) `ProjectName = [project name]`
`[project name]`: a string with the name of the project

Observation settings

`ObsLength`: duration of the observation

Syntax:

1) `ObsLength = [obs. length]`
`[obs. length]`: the duration of the observation in seconds (integer). Value zero means parameter unset.

`ObsStartHour`: UTC start time of the observation

Syntax:

1) `ObsStartHour = [obs. start hour]`
`[obs. start hour]`: the UTC start time of the observation (format hh:mm:ss). Value zero means parameter unset.

`ObsStopHour`: UTC stop time of the observation

Syntax:

1) `ObsStopHour = [obs. stop hour]`
`[obs. stop hour]`: the UTC stop time of the observation (format hh:mm:ss). Value zero means parameter unset.

`Setup`: name of the setup file

Syntax:

1) `Setup = [projectcode]/[file name]`
2) `[device id] > Setup = [projectcode]/[file name]`
`[device id]`: device for which the setup file contains settings. Values are ANTENNA, DFB and ROACH for indicating the homonymous device.
`[projectcode]`: the project code assigned by the TAC
`[file name]`: name of the setup file

`WaitBeforeStart`: time delay before the start of the data acquisition

Syntax:

1) `WaitBeforeStart = [delay]`
`[delay]`: the time delay in seconds between the beginning of the observation's setup procedures and the beginning of the data acquisition (integer).

Antenna settings

Keywords marked with the (*) sign can be mentioned in the MAIN setup file without the `ANTENNA >` preamble.

`ActiveSurface`: shape and adaptive behaviour of the primary reflector

Syntax:

1) `ActiveSurface = [Shape and adaptive behaviour]`
`[shape and adaptive behaviour]`: the shape and the adaptive behaviour of the primary mirror.
Allowed values are:
- `PARABOLIC` : parabolic shape, continuously adapted
- `PARABOLIC-FIXED` : parabolic shape, fixed to 45 degrees elevation optimal shape
- `SHAPED` : shaped shape, continuously adapted
- `SHAPED-FIXED` : shaped shape, fixed to 45 degrees elevation optimal shape
- `RECEIVER-DEFAULT` : Parabolic for primary focus receivers, Shaped for all other receivers

`AzSector`: azimuth sector for pointing the antenna

Syntax:

1) `AzSector = [azimuth sector]`
`[azimuth sector]`: the azimuth sector for pointing the antenna. Values `NEUTRAL`, `CW`, `CCW`

`CalSource`: switches ON/OFF the calibration signal source

Syntax:

1) `CalSource = ON`
2) `CalSource = OFF`
`ON` : the cal source is activated
`OFF` : the cal source is deactivated

`Receiver`: receiver, and related settings

Syntax:

1) `Receiver = P-BAND,[P-BAND pol],[P-BAND freq filter]`
2) `Receiver = L-BAND,[L-BAND pol],[L-BAND freq filter],[LO freq]`
3) `Receiver = LP-DUAL,[P-BAND pol],[P-BAND freq filter],[L-BAND pol],[L-BAND freq filter],[LO freq]`
4) `Receiver = C-BAND,[LO freq],[bandwidth]`
5) `Receiver = K-BAND,[LO freq],[bandwidth]`
`[P-BAND pol]`: polarization to acquire from the `P-BAND` (`LINEAR`/`CIRCULAR`)
`[P-BAND freq filter]`: the frequency filter for the `P-BAND`. Values: `OFF` = all available band; `NARROW` = 310MHz-350MHz; `WIDE` = 305MHz-410MHz.
`[L-BAND pol]`: polarization to acquire from the `L-BAND` (`LINEAR`/`CIRCULAR`)
[L-BAND freq filter]: the frequency filter for the L-BAND. Values: `OFF` = all available band; `NARROW` = 1320MHz-1780MHz; `WIDE` = 1292MHz-1804MHz; `VLBI-LOW` = 1350MHz-1450MHz; `VLBI-HIGH` = 1625MHz-1715MHz
`[LO freq]`: frequency of the local oscillator (= to the acquired band lower frequency value)
`[bandwidth]`: TotalPower bandwidth in MHz (300,730,1250,2000).

`Source`: celestial object to be pointed

Syntax:

1) `Source = [source name]`
2) `Source = [source name],[coordinate system],[longitude],[latitude]`
`[source name]`: the name of the source
`[coordinate system]`: the frame for espressing the source coordinates: `J2000`, `GALACTIC`, `AZ/EL`
`[Longitude]`: right ascension (hh:mm:ss.ddd), galactic longitude or azimuth (ddd.dddd), accordingly to the selected frame
`[latitude]`: declination (+/-dd:mm:ss.zzz), galactic longitude or azimuth (+/-ddd.dddd), accordingly to the selected frame

Note. Syntax #1 can be used for catalogue (psrcat) pulsars only. J2000 catalogue

coordinates are used.

Backend settings: generic backend

`Bandwidth`: width in MHz of the backend's frequency band

Syntax:

1) `Bandwidth = [bw]`
`[bw]`: the width in MHz of the backend's frequency band (Float).

`BitsPerSample`: number of bits to represent each data sample

Syntax:

1) `BitsPerSample = [Nbits]`
`[nbits]`: the number of bits to represent each data sample (1,2,4,8)

`DedispersionMode`: coherent or incoherent signal dedispersion

Syntax:

1) `DedispersionMode = INCOHERENT`
2) `DedispersionMode = COHERENT`
`INCOHERENT` = incoherent dedispersion
`COHERENT` = coherent dedispersion

`DispersionMeasure`: dispersion measure value for coherent dedispersion

Syntax:

1) `DispersionMeasure = [dm]`
`[dm]`: dispersion measure value

`Frequency`: sky frequency that falls at the center of the backend's frequency band

Syntax:

1) `Frequency = [freq]`
`[freq]`: the value in MHz of the sky frequency that falls at the center of the backend's band (float).

`Mode`: data acquisition mode

Syntax:

1) `Mode = [mode]`
[mode]: the data acquisition mode.
Allowed values are:
- `BASEBAND` : baseband recording (ROACH)
- `FOLD` : pulsar folding mode (DFB,ROACH)

- PSRCAL : pulsar calibration source observation (DFB,ROACH)
- SEARCH : pulsar search mode (DFB,ROACH)
- SEARCHSET : internal level adjustment for pulsar search mode (DFB)

NumberOfChannels: number of frequency channels

Syntax:

1) NumberOfChannels = [nchan]
[nchan]: number of frequency channels (int)

NumberOfPols: number of polarizations to be acquired

Syntax:

1) NumberOfPols = [npol]
[npol]: the number of polarizations to be recorded. Values are 1 = total intensity, 2 = left and right intensity only, 4 = full Stokes

ProfileBins: number of bins to represent the pulsar profile

Syntax:

1) ProfileBins = [nbin]
[nbin]: The number of phase bins in pulse profile (integer power of 2)

SamplingTime: signal's sampling time

Syntax:

1) SamplingTime = [tsamp]
[tsamp]: the search mode sampling time in microseconds (int).

SubintTime: pulsar folding mode subintegration time

Syntax:

1) SubintTime = [tsub]
[tsub]: the folding mode subintegration's time in seconds (int).

Backend settings: Pulsar Digital Filterbank (DFB) custom settings

`ChannelRange`: channel range to be acquired

Syntax:

1) `ChannelRange = [ch1]-[ch2]`
2) `ChannelRange = all`
`[ch1]`: the first frequency channel o be acquired
`[ch2]`: the last frequency channel o be acquired
`0-0` : all frequency channels are acquired
`ALL` : all frequency channels are acquired

`ConfigFile`: DFB configuration file name

Syntax:

1) `ConfigFile = [cfg file]`
`[cfg file]`: the configuration file for the DFB (string).
Formats:
`srch_[bw]_[nch]` : search mode configuration
`pdfb4_[nbin]_[bw]_[nch]`: folding mode configuration
Fields description:
`[nbin]`: number of phase bins for representing the pulse profile
`[bw]`: bandwidth in MHz
`[nch]`: number of channels

Available configurations:

```
pdfb4_1024_1024_1024
pdfb4_1024_1024_1024_new
pdfb4_1024_1024_2048
pdfb4_1024_1024_256
pdfb4_1024_1024_512
pdfb4_1024_128_1024
pdfb4_1024_128_2048
pdfb4_1024_128_512
pdfb4_1024_256_1024
pdfb4_1024_256_2048
pdfb4_1024_256_256
pdfb4_1024_256_512
pdfb4_1024_512_1024
pdfb4_1024_512_2048
pdfb4_1024_512_512
pdfb4_1024_64_1024
pdfb4_1024_64_2048
pdfb4_1024_64_512
pdfb4_128_64_1024
pdfb4_128_64_512
pdfb4_2048_1024_1024
pdfb4_2048_1024_128
pdfb4_2048_1024_2048
pdfb4_2048_1024_256
pdfb4_2048_1024_512
pdfb4_2048_256_1024
pdfb4_2048_256_2048
pdfb4_2048_256_512
pdfb4_2048_512_1024
```

```
pdfb4_2048_512_2048
pdfb4_2048_512_512
pdfb4_256_1024_1024
pdfb4_256_1024_2048
pdfb4_256_1024_512
pdfb4_256_256_1024
pdfb4_256_256_2048
pdfb4_256_256_512
pdfb4_256_64_1024
pdfb4_256_64_512
pdfb4_512_1024_1024
pdfb4_512_1024_2048
pdfb4_512_1024_512
pdfb4_512_128_1024
pdfb4_512_128_2048
pdfb4_512_128_512
pdfb4_512_16_2048
pdfb4_512_256_1024
pdfb4_512_256_2048
pdfb4_512_256_512
pdfb4_512_32_2048
pdfb4_512_512_1024
pdfb4_512_512_2048
pdfb4_512_512_512
pdfb4_512_64_1024
pdfb4_512_64_2048
pdfb4_512_64_512
pdfb4_512_8_2048
srch_1024_1024
srch_1024_512
srch_256_1024
srch_256_512
srch_512_1024
srch_512_128
srch_512_256
srch_512_512
```

`CyclePeriod`: DFB cycle period

Syntax:

1) `CyclePeriod = [cycle period]`
`[cycle period]`: the DFB cycle period in seconds (range [2,30]).

`MaxFileLength`: search mode maximum file length

Syntax:

1) `MaxFileLength = [max length]`
`[max length]`: The maximum file length in seconds for search mode files produced by the DFB (int). Value 0 (zero digit) means unlimited length.

`SamplesSubint`: number of samples in search mode data blocks

Syntax:

1) `SamplesSubint = [nsamp]`
`[nsamp]`: the number of samples for each DFB search mode data block (int power of 2)

`TimeConstant`: time to calculate the average bandpass

Syntax:

1) `TimeConstant = [tconst]`
`[tconst]`: the time to calculate the average bandpass in seconds (int >= 1).

`WriteFile`: data written/not written on files

Syntax:

1) `WriteFile = YES`
2) `WriteFile = NO`
`YES` : data are written on file
`NO` : no file is opened for writting data.

Backend settings: ROACH backend custom settings

`AdvancedOptions`: dspsr (folding) or digifil (search) not standard/common options

Syntax:

1) `AdvancedOptions = [options]`
`[options]`: a string containing all "`ADVANCED`" options for dspsr (fold mode) and digifil (search mode), as they would be specified in a dspsr/digifil shell command.

The attribute "`BASIC`" indicates dspsr and digifil options that can directly be set by any of the aforementioned backends' keywords, no matter if they are generic or custom ones. In `FOLD` mode they are:

```
-F <N>[:D]  create an N-channel filterbank [with coherent dedispersion]
-b nbin     number of phase bins in folded profile
-L seconds  create integrations of specified duration
```

while in search mode they are:

```
-b bits    number of bits per sample output to file
-F <N>:D]  create an N-channel filterbank (voltages only) [with coherent
dedispersion]
-D dm      set the dispersion measure
-t nsamp   decimate in time
-d npol    1=PP+QQ, 2=PP,QQ, 3=(PP+QQ)^2 4=PP,QQ,PQ,QP
```

The attribute "`ADVANCED`" indicates dspsr and digifil options that imply custom settings for executing the processing calculations or have an impact in the final data file. `FOLD` mode, i.e. dspsr routine, advanced options are:

```
Processor options:
-Z lib      choose the FFT library ('-Z help' for availability)
-dump op    dump time series before performing operation
-order      order data optimally when possible [default:true]

RFI removal options:
-2 code          unpacker options ("2-bit" excision)
-skz             apply spectral kurtosis filterbank RFI zapping
-noskz_too       also produce un-zapped version of output
```

```
-skzm samples    samples to integrate for spectral kurtosis statistics
-skzs stddevs    number of std deviations to use for spectral kurtosis
excisions
-skz_start chan  first channel where signal is expected
-skz_end chan    last channel where signal is expected
-skz_no_fscr     do not use SKDetector Fscrunch feature
-skz_no_tscr     do not use SKDetector Tscrunch feature
-skz_no_ft       do not use SKDetector despeckeler
-sk_fold         fold the SKFilterbank output

Dispersion removal options:
-G nbin              create phase-locked filterbank
-cyclic N            form cyclic spectra with N channels (per input channel)
-cyclicoversample M  use M times as many lags to improve cyclic channel
isolation (4 is recommended)
-D dm                over-ride dispersion measure
-K                   remove inter-channel dispersion delays
-x nfft|minX         over-ride optimal transform length
-R                   apply time-variable narrow-band RFI filter
-pac dbase           pac database for phase-coherent matrix convolution
-fft-bench           use benchmark data to choose optimal FFT length

Detection options:
-d npol       1=PP+QQ, 2=PP,QQ, 3=(PP+QQ)^2 4=PP,QQ,PQ,QP
-n ndim       [experimental] ndim of output when npol=4
-4            compute fourth-order moments

Folding options:
-p phase      reference phase of rising edge of bin zero
-E file       pulsar ephemeris used to generate predictor
-P file       phase predictor used for folding
-w file       phase predictors used for folding.
-X name       additional pulsar to be folded

Time division options:
-s             create single pulse sub-integrations
-turns N       create integrations of specified number of spin periods
-Lepoch MJD    start time of first sub-integration (when -L is used)
-Lmin seconds  minimum integration length output
-y             output partially completed integrations

Output archive options:
-a archive     output archive class name
-e ext         output filename extension
-no_dyn        disable dynamic extensions
-j job         psrsh command run before output
-J a.psh       psrsh script run before output
```

SEARCH mode, i.e. digifil routine, advanced options are:

```
Processor options:
-Z lib             choose the FFT library ('-Z help' for availability)
-dump op           dump time series before performing operation
-c                 keep offset and scale constant
-x nfft            backward FFT length in voltage filterbank
-K                 remove inter-channel dispersion delays
-d npol            1=PP+QQ, 2=PP,QQ, 3=(PP+QQ)^2 4=PP,QQ,PQ,QP
-P ipol            process only a single polarization of input
-I secs            rescale interval in seconds
-s fac             data scale factor to apply
-p                 revert to FPT order
```

If any of the mentioned FOLD/SEARCH options require further arguments not listed here, such arguments can also be indicated in the AdvancedOptions keyword string.

Any option not listed as basic nor as advanced will be ignored.

Appendix A - Calibration source acquisition schedule line

The use of the noise source for calibration purposes requires a dedicated setup that is planned to be fully automated in SEADAS in the near future. Given the current lack of such automatization, the user has to follow one of the following procedures while preparing the schedule. The first one is addressed to the user who wishes to use the PDFB for the data acquisition; conversely, the second one must be followed by the user who is not interested in using the PDFB.

A.1. The user wants to acquire data with the PDFB, and maybe other backends.

The schedule line for the calibration observation is the same as for the on-source folding mode observation, but for the following changes:

`Source = [source name]_R,J2000,[offset RA],[offset DEC]`

Explanation: The source name must be followed by the _R suffix, the coordinates system must be always `J2000`, the telescope must be pointed at about one degree offset position with respect to the source position

`DFB > Mode = PSRCAL`

Explanation: The PDFB must be instructed to operate in PSRCAL mode, in order to command the fast switching of the calibration source.

`ANTENNA > CalSource = ON`

Explanation: The calibration source must be enabled in the antenna control procedures

Example (to be put in one single line in the schedule):

```
Source = B0329+54_R,J2000,03:33:00.000,+55:30:00.000 ; Setup = 22-19/mySetup.stp ,
DFB > Mode = PSRCAL ; ANTENNA > CalSource = ON ; ObsLength = 120
```

If other backends are needed, they all must be instructed to acquire data in `PSRCAL` mode

A.2. The user is not interested to acquire data with the PDFB, but the ROACH system only.

The fast switching on and off of the calibration source is triggered by the PDFB. For this reason this backend should be enabled and some settings should be given to it. As in the previous case, the schedule line for the calibration observation is the same as for the on-source folding mode observation, but for the following changes:

`Source = [source name]_R,J2000,[offset RA],[offset DEC]`

Explanation: The source name must be followed by the _R suffix, the coordinates system must be always J2000, the telescope must be pointed at about one degree offset position with respect to the source position

`ANTENNA > CalSource = ON`

Explanation: The calibration source must be enabled in the antenna control procedures

`ROACH > Mode = PSRCAL`

Explanation: The ROACH system must be informed that the ongoing acquisition is

about the cal source.

`DFB > ENABLED`

Explanation: The PDFB must be enabled so that the noise source can be fast switched on and off

`DFB > Setup = [project code]/PDFBCal.stp`

Explanation: Settings for the PDFB are indicated in a text file called PDFBCal.stp, which must be placed in the directory /home/corr/setup/**[project code]**. The content of this file is the following:

```
Mode = PSRCAL
ConfigFile = pdfb4_512_512_512
Frequency = 1548.000
SubintTime = 10
ChannelRange = all
writeFile = NO
CyclePeriod = 10.000
```

N.B. Values for all keywords but `Mode` are dummy values in this particular case. The one for `ConfigFile` must anyway be any folding mode config file (i.e. any pfb4_BB_BW_CH file).

Example (to be put in one single line in the schedule):

```
Source = B0329+54_R,J2000,03:33:00.000,+55:30:00.000 ; Setup = 22-19/mySetup.stp;
ANTENNA > CalSource = ON ; ROACH > Mode = PSRCAL ; ObsLength = 120; DFB > ENABLED ;
DFB > Setup = PDFBCal.txt
```

N.B. The schedule line for the on-source observation MUST contain the statements:

DFB > DISABLED
ANTENNA > CalSource = OFF

Example (to be put in one single line in the schedule):

```
Source = B0329+54 ; Setup = 22-19/mySetup.stp ; DFB > DISABLED ; ANTENNA >
CalSource = OFF ; ObsLength = 3600
```