



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF
Issue: 01
Revision: 03
Date: 06/05/2020

- 1 / 19 -

SOLAR ORBITER



RPW Operation Centre

BIAS Calibration Software (BICAS) User Manual

ROC-PRO-SFT-SUM-00080-IRF
Iss.01, Rev.03

Prepared by:	Function:	Signature:	Date
Erik P G Johansson	Research engineer, BIAS data management		06/05/2020
Verified by:	Function:	Signature:	Date
			Dd/mm/yyyy
Approved by:	Function:	Signature:	Date
Yuri Khotyaintsev	BIAS lead Co-I		Dd/mm/yyyy
For application:	Function:	Signature:	Date
			Dd/mm/yyyy

CLASSIFICATION

PUBLIC



RESTRICTED



Laboratoire d'Études Spatiales et d'Instrumentation en Astrophysique

CNRS-Observatoire de PARIS
Section de MEUDON – LESIA
5, place Jules Janssen
92195 Meudon Cedex – France



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 2 / 19 -

Change Record

Issue	Rev.	Date	Authors	Modifications
1	0	26/04/2017	E. Johansson	First draft
1	1	16/01/2019	E. Johansson	Miscellaneous updates. Added RODP datasets.
1	2	20/09/2019	E. Johansson	Catch up with state of s/w, various improvements
1	3	06/05/2020	E. Johansson	Updated for BICAS 2.0.0. Removed references to two pipelines (ROC-SGSE removed).

Acronym List

Acronym	Definition
BICAS	BIAS Calibration Software
CLI	Command-Line Interface
ICD	Interface Control Document
IDE	Integrated Development Environment
IRF	Swedish Institute of Space Physics (swe. Institutet för rymdfysik)
OS	Operating System
PDID	Process Data ID
PDV	Process Data Variable
RCS	RPW Calibration Software
RCT	RPW Calibration Table
ROC	RPW Operation Centre
RPW	Radio and Plasma Waves instrument
SUM	Software User Manual
TBW	To be written
TC	Telecommand(s)



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 3 / 19 -

Table of Contents

1	General.....	5
1.1	Scope of the document	5
1.2	Applicable documents	5
1.3	Reference documents	5
2	Conventions	6
3	Description of the software	6
3.1	Purpose of the software	6
3.2	Operations environment	6
3.3	Software design overview.....	6
3.3.1	<i>Introduction</i>	<i>6</i>
3.3.2	<i>Relationship with lfu-matlab git repo.....</i>	<i>6</i>
3.3.3	<i>Structure of the source code and its execution</i>	<i>7</i>
3.4	External interfaces.....	8
3.5	Software operating modes.....	8
3.5.1	<i>Normal operating mode.....</i>	<i>8</i>
3.5.2	<i>MATLAB operating mode.....</i>	<i>8</i>
3.6	Description of the software data.....	9
3.6.1	<i>Software input data</i>	<i>9</i>
3.6.2	<i>Software output data</i>	<i>9</i>
3.6.3	<i>Software installation and configuration files</i>	<i>10</i>
3.6.4	<i>Software internal data files</i>	<i>10</i>
3.6.5	<i>Software testing data files</i>	<i>11</i>
4	Reference manual	11
4.1	Software configuration requirements	11
4.1.1	<i>General.....</i>	<i>11</i>
4.1.2	<i>Hardware configuration requirements.....</i>	<i>11</i>
4.1.3	<i>Software configuration requirements</i>	<i>11</i>
4.2	Operations manual	12
4.2.1	<i>Set-up and initialisation</i>	<i>12</i>
4.2.2	<i>Getting started</i>	<i>12</i>
4.2.3	<i>Normal operations.....</i>	<i>12</i>
4.2.4	<i>Normal termination.....</i>	<i>12</i>
4.2.5	<i>Error conditions.....</i>	<i>12</i>
4.2.6	<i>Recover runs</i>	<i>12</i>
4.2.7	<i>Help method</i>	<i>13</i>
4.2.8	<i>Commands and operations</i>	<i>13</i>
4.2.9	<i>Error messages</i>	<i>13</i>
4.2.10	<i>Software testing operations.....</i>	<i>14</i>
4.2.11	<i>Software upgrading operations.....</i>	<i>14</i>
4.2.12	<i>Software uninstalling operations.....</i>	<i>14</i>
4.3	Configuration	14
4.3.1	<i>Settings (and constants).....</i>	<i>14</i>
4.3.2	<i>Additional hard-coded "error-safe constants".....</i>	<i>15</i>
4.4	Make BICAS generate the RCS (S/W) descriptor.....	15
5	Tutorial.....	16
5.1	Introduction.....	16
5.2	Getting started.....	16



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF
Issue: 01
Revision: 03
Date: 06/05/2020

5.3 Using the software on a typical task 16

6 Appendices 16

6.1 Troubleshooting & know issues 16

6.1.1 *Known limitations, caveats* 16

7 List of TBC/TBD/TBWs 17

8 Distribution list..... 19

List of Figures

No table of figures entries found.

List of Tables

Table 1. Software input datasets. 9

Table 2. Software output datasets. 9

Table 3. RCTs required by BICAS and their current default naming schemes..... 11



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 5 / 19 -

1 GENERAL

1.1 Scope of the document

This user manual primarily (1) seeks to describe information practically useful for the installation and running of BICAS s/w including basic errors, and secondarily (2) offer some insights into its inner workings and source code structure to make debugging and modifications easier, including with regards to the calibration. Much of the basic user interface and functioning is specified and described the RCS ICD (AD1) and this document will therefore mostly concern itself with functionality *not* specified in the RCS ICD. To the extent that the software is not complete, and/or not update to date with the RCS ICD, it will generally reflect the current state of the software and not any planned future state.

The document does *not* describe how data is calibrated (see RD1), or the format of input and output datasets.

1.2 Applicable documents

This document responds to the requirements of the documents listed in the following table:

Mark	Reference/Iss/Rev	Title of the document	Authors	Date
AD1	ROC-PRO-PIP-ICD-00037/1/2 (draft)	RPW Calibration Software Interface Control Document	M.Duarte, X.Bonnin	11/07/2019 (draft)
AD2	ROC-PRO-DAT-NTT-00006-LES/1/0	RPW Data Products	X.Bonnin	23/12/2016
AD3	ROC-TST-GSE-NTT-00017-LES	Data format and metadata definition for the ROC-SGSE data	X. Bonnin	14/10/2016

1.3 Reference documents

This document is based on the documents listed in the following table:

Mark	Reference/Iss/Rev	Title of the document	Authors	Date
RD1	RPW-SYS-MEB-BIA-SPC-00088-IRF	Technical Specifications of the BIAS Calibration Software	E. Johansson	20/09/2019
RD2				
RD3				
RD4				
RD5				



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 6 / 19 -

2 CONVENTIONS

This document uses a special font when referring to computer variables (e.g. CDF global attributes), filenames, directory paths, system commands, and CLI arguments, e.g.

```
bicas --help.
```

Paths are often implicitly specified relative to the `src/` subdirectory, but sometimes relative to BICAS' root directory. MATLAB files, containing MATLAB code, is typically referred to using only their unique filenames.

3 DESCRIPTION OF THE SOFTWARE

3.1 Purpose of the software

BICAS is the BIAS team's contribution to the calibration of RPW instrument data. It is designed to

- read bias currents in TM (TC) units and derive calibrated bias currents for every antenna.
- read electric data that has passed through the BIAS subsystem and been digitized by the LFR and TDS analyzers (TDS-LFM), i.e. voltages in TM units, and derive the corresponding calibrated data corresponding to
 - single voltage values at the probes, and
 - differential voltage values between the probes.

All input and output data is provided in the form of a subset of the datasets defined in AD2.

3.2 Operations environment

BICAS does not require any particular software interfaces, more than basic requirements on installation and configuration described in Section 4.1.3.

3.3 Software design overview

3.3.1 Introduction

BICAS fundamentally consists of source code written in the MATLAB scripting language and interpreted by the proprietary MathWorks MATLAB software, which in turn is launched from a Unix/Linux bash script. Since all source code (supplied by IRF) is written in interpreted languages, no source code ever needs to be pre-compiled.

3.3.2 Relationship with Irfu-matlab git repo

Internally at IRF, BICAS is developed as a part of a larger git repository of software, named "irfu-matlab". "irfu-matlab" is focused on MATLAB code and is primarily used internally at the IRF and by former IRF coworkers. BICAS uses the remainder of irfu-matlab as an internal, but separate, library of generic functions. Every new official version of BICAS is meant to be released as a commit on the "master" branch of irfu-matlab's git repository, but not every new commit on irfu-matlab's master branch corresponds to a new BICAS version. Note that



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 7 / 19 -

the irfu-matlab git repository is distinct from the git repository nominally used to deliver BICAS to the ROC.

The reading and writing of CDF files is done via the “NASA CDF patch for MATLAB”, freely available from NASA GSFC and already included in irfu-matlab and hence in BICAS.

3.3.3 Structure of the source code and its execution

The (MATLAB) source code consists of a few main files which use the remainder of the code as library functions. MATLAB code files are recognized via their filename suffix “.m”. Below describes some of the most important source code, but far from all.

`roc/bicas` is a bash script which launches MATLAB and instructs it to execute the MATLAB function `+bicas/main.m`. This wrapper script (`roc/bicas`) only handles a few basic tasks:

- (1) launching the correct installation of MATLAB to run the BICAS MATLAB code,
- (2) extracting and saving log messages from MATLAB’s stdout and stderr to file, and
- (3) handling errors associated with above tasks.

It passes all CLI arguments on to `main.m` and receives an exit code from it. It only interprets (1) the CLI argument `--config` that specifies the configuration file in order to determine which MATLAB installation to use, and (2) `--log` which specifies the log file (see AD1).

`+bicas/main.m` performs basic initialization and checks, interprets CLI arguments and their syntax, and converts them into data structures that is more natural and easy for code to work with and act on. For processing datasets, the function calls `execute_sw_mode.m` to do the actual processing associated with a s/w mode. Its interface is very analogous to the bash wrapper script `roc/bicas`.

`+bicas/execute_sw_mode.m` reads the required inputs datasets (CDF files) into analogous MATLAB data structures in RAM and does some checks on the contents before passing it on for processing.

`+bicas/swmode_defs.m` (roughly speaking) centralizes hard-coded metadata associated with every s/w mode: input and output datasets, master CDF version, which function in `proc.m` that should be used to actually convert the data in the input datasets into data for the output datasets. s/w mode-related data in the s/w descriptor and s/w mode-related constraints on argument indirectly comes from here.

`+bicas/RCT.m` contains code for reading and interpreting supported RCTs.

`+bicas/calib.m` contains all calibration-related code and little else. It is designed so that it can be used (called) separately from the rest of BICAS in order to test, debug and inspect the calibration.

`+bicas/create_default_SETTINGS.m` defines the settings key-value pairs that exist, and their default values. It also tends to contain comments on the meaning of these settings. See Section 4.3.1.



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 8 / 19 -

3.4 External interfaces

BICAS presently contains no user interfaces besides those specified in the RCS ICD (AD1), i.e.

- The command-line arguments (though slightly amended; See Section 4.2.8),
- the config file,
- the log file,
- the standard/error output
- exit (error) code, and
- the input and output datasets.

3.5 Software operating modes

3.5.1 Normal operating mode

The normal operating mode is to run BICAS by launching the bash script `roc/bicas`.

3.5.2 MATLAB operating mode

Since BICAS is fundamentally implemented in MATLAB, and the bash wrapper script that launches it passes on all the CLI arguments to the corresponding MATLAB script, BICAS can also be run from the MATLAB command-line, and hence the MATLAB GUI, with almost identical functionality and user interface. This is useful since (1) it can likely be run on platforms other than Linux(!) (not tested), (2) it eliminates the overhead of launching and quitting MATLAB (a few seconds), and (3) it enables the user to use the MATLAB IDE's debugging features (breakpoints, step-by-step execution, inspecting internal variables etc.). This mode of operation is therefore very useful for debugging and development. The other primary differences between this mode of operation and launching BICAS from the bash wrapper script are: (1) The user has to use MATLAB's syntax for submitting arguments, and (2) log messages cannot be saved to a log file using the official `--log <path>` arguments, and are instead intermingled with error messages (stderr), standard output (stdout), and various MATLAB text output although all these categories of messages can be visually distinguished, partly due to using BICAS' own internal standardized use of prefixes.

To call BICAS from within MATLAB, simply launch `main.m` from the MATLAB command-line with the same arguments that would be used from the OS, but using MATLAB's syntax, e.g.

```
bicas.main('--version', '--config', '~/bicas.conf')
```

. The prefix `"bicas."` is MATLAB syntax to denote that `main.m` is part of the MATLAB code "package" named `"bicas"`, which is defined by the content of the subdirectory `+bicas/`.

The flag `--log` (specified in the RCS ICD, AD1) is permitted but will have no effect since only the bash launcher script interprets and uses that information. Instead, BICAS additionally supports the option

```
--log-matlab <path>
```

which specifies a log file generated by the MATLAB code only.



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF
Issue: 01
Revision: 03
Date: 06/05/2020

- 9 / 19 -

The value returned by `main.m` is identical to the exit code that would be returned to the OS when launching BICAS from the bash launch script.

Running BICAS from the MATLAB command-line requires one to correctly initialize MATLAB's internal path variable for finding MATLAB source code files, by adding

- BICAS' `src/` directory, and
- BICAS' `lib/irfu-matlab/` directory.

The necessary MATLAB functionality is exposed to the user via MATLAB functions `path` and `addpath`.

3.6 Description of the software data

3.6.1 Software input data

Input data name/ DATASET_ID	Description	Required in the following mode(s)
SOLO_L1R_RPW-LFR-SBM1-CWF-E	Uncalibrated electric fields	LFR-SBM1-CWF-E
SOLO_L1R_RPW-LFR-SBM2-CWF-E	Uncalibrated electric fields	LFR-SBM2-CWF-E
SOLO_L1R_RPW-LFR-SURV-CWF-E	Uncalibrated electric fields	LFR-SURV-CWF-E
SOLO_L1R_RPW-LFR-SURV-SWF-E	Uncalibrated electric fields	LFR-SURV-SWF-E
SOLO_L1R_RPW-TDS-LFM-CWF-E	Uncalibrated electric fields	TDS-LFM-CWF-E
SOLO_L1R_RPW-TDS-LFM-RSWF-E	Uncalibrated electric fields	TDS-LFM-RSWF-E
SOLO_HK_RPW-BIA	BIAS HK	All modes
SOLO_L1_RPW-BIA-CURRENT	Uncalibrated bias currents (TC)	All modes

Table 1. Software input datasets.

A detailed description of the datasets can be found in TBW.

3.6.2 Software output data

Output data name/ DATASET_ID	Description	Product of the following mode(s)
SOLO_L2_RPW-LFR-SBM1-CWF-E	Calibrated E field & bias current	LFR-SBM1-CWF-E
SOLO_L2_RPW-LFR-SBM2-CWF-E	Calibrated E field & bias current	LFR-SBM2-CWF-E
SOLO_L2_RPW-LFR-SURV-CWF-E	Calibrated E field & bias current	LFR-SURV-CWF-E
SOLO_L2_RPW-LFR-SURV-SWF-E	Calibrated E field & bias current	LFR-SURV-SWF-E
SOLO_L2_RPW-TDS-LFM-CWF-E	Calibrated E field & bias current	TDS-LFM-CWF-E
SOLO_L2_RPW-TDS-LFM-RSWF-E	Calibrated E field & bias current	TDS-LFM-RSWF-E

Table 2. Software output datasets.

A detailed description of the datasets can be found in TBW.



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 10 / 19 -

3.6.3 Software installation and configuration files

3.6.3.1 Configuration file

The SUM shall identify the files that are required to install and configure the software.

If references to a more detailed description of these files do exist, they shall be provided. If there is no reference, a full description shall be given in the present document.

BICAS uses a configuration file, by default `config/bicas.conf`. The choice of configuration file can be overridden with the `--config` CLI argument. The content of the config file relates to the settings actually used by BICAS as described in Section 4.3.

3.6.3.2 Exact MATLAB installation to use

The exact MATLAB executable to use must be specified in BICAS' config file before BICAS can be run. This feature is useful if the user has multiple MATLAB installations on the same machine. Example excerpt from BICAS config file.

```
MATLAB_COMMAND = "/usr/local/MATLAB/R2019b/bin/matlab"
```

3.6.3.3 Specify CDF leap second table

The OS should preferably set the environment variable `CDF_LEAPSECONDSTABLE` before invoking BICAS, see Section 4.1.3

3.6.4 Software internal data files

3.6.4.1 Master CDF files

BICAS requires dataset master CDF files located in a directory as specified by an environment variable `ROC_RCS_CAL_PATH` as defined in AD1. Only master CDF files corresponding to output datasets to be produced are necessary.

3.6.4.2 RPW Calibration Tables (RCTs)

BICAS requires one BIAS RCT, and may, depending on processing mode, require one LFR RCT, and two TDS RCTs, see Table 3.

Note: The filenaming conventions recognized by BICAS can be set via BICAS' SETTINGS mechanism, see Section 4.3.

Note: BICAS nominally (i.e. by default) retrieves the exact filenames of non-BIAS RCTs from the CDF global attribute CALIBRATION TABLE and thus does not (by default) require any filenaming convention as in Table 3.

RCT filenaming convention	Comments
<code>SOLO_CAL_RPW_BIAS_Vyyyymmddhhmm.cdf</code>	File naming convention is always used to identify BIAS RCTs.
<code>SOLO_CAL_RCT-LFR-BIAS_Vyyyymmddhhmmdd.cdf</code>	File naming convention is only used if BICAS runs with disabled use of CDF global attribute CALIBRATION_TABLE (not default). This can be specified using BICAS' settings mechanism, see Section 4.3.1.
<code>SOLO_CAL_RCT-TDS-LFM-RSWF-E_Vyyyymmdd.cdf</code>	



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 11 / 19 -

SOLO_CAL_RCT-TDS-LFM-CWF-
E_Vyyyymmdd.cdf

Note: The LFR and TDS RCTs supplied to the BIAS team do not seem to conform to the official file naming convention. The file naming described here correspond to the actual LFR & TDS RCTs.

Table 3. RCTs required by BICAS and their current default naming schemes.

3.6.5 Software testing data files

BICAS contains some MATLAB code for somewhat informal automated test code that covers some well-defined functions which only constitute a very small part of BICAS. The code is found in the files named *___ATEST.m. They have to be invoked manually from within MATLAB.

4 REFERENCE MANUAL

4.1 Software configuration requirements

4.1.1 General

4.1.2 Hardware configuration requirements

BICAS itself does not presently have any particular hardware requirements. Instead, the primary hardware requirements are expected to be driven by (1) MATLAB's requirements to run, and (2) the sizes of the datasets. BICAS, i.e. irfu-matlab excluding BICAS' data files and excluding the irfu-matlab git repository data, requires 101 MiB of storage space (30/04/2020). BICAS has been developed and tested on a Linux PC with 32 GiB of RAM but it likely works with less.

4.1.3 Software configuration requirements

Running BICAS the nominal way as specified in the ICD (AD1) requires

- Linux operating system (?)
- the bash shell
- MathWorks MATLAB version R2019b

BICAS has been developed and tested under Linux Ubuntu 18.04, 64-bit, and "GNU bash, version 4.4.20(1)-release (x86_64-pc-linux-gnu)". The exact Linux and bash version should be unimportant.

BICAS can *likely* (untested) be run using MATLAB versions other than R2019b too since new versions of MATLAB are typically backwards compatible, and *possibly* also using slightly older MATLAB versions, but neither of these two options is guaranteed to work. In both of these cases, one has to manually alter BICAS' hard-coded MATLAB version check. See also Section 3.5.2 for running BICAS using only MATLAB (on any OS).



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 12 / 19 -

Although BICAS may or may not be delivered with the irfu-matlab git repository data (a `.git/` subdirectory), it does *not* require the git repository data, and it does *not* require git to be installed.

4.1.3.1 Required modification of config file and MATLAB command

As a bare minimum, the user needs to call BICAS in such a way that it uses a config file which specifies the MATLAB command/executable (e.g. a path) to be used. Assign the settings key `MATLAB_COMMAND`. See the default `config/bicas.conf` file.

4.1.3.2 CDF leap second table

BICAS needs a leap second table to function due to its use of the “NASA CDF patch for MATLAB”. If (1) there is an OS environment variable `CDF_LEAPSECONDSTABLE` that specifies a valid path to a valid leap second table (preferred), then that will be used. If it does not, BICAS will (2) use the file `CDFLeapSeconds.txt` which is included in BICAS (via irfu-matlab). If both are available and the latter (2) seems more up-to-date than the former (1), then (1), `CDF_LEAPSECONDSTABLE`, will still be used but a warning will be printed to the log.

4.2 Operations manual

4.2.1 Set-up and initialisation

The SUM shall describe any procedures to be performed by the user in order to install software on the equipment, to perform the installation, to configure the software, to delete or overwrite former files or data, and to enter parameters for software operation.

4.2.2 Getting started

The SUM shall include the step-by-step procedures for beginning work, including any options available, and a check-list for problem determination.

4.2.3 Normal operations

The SUM shall identify the normal operations, to be performed by the user, for the use of software, including inputs from other software or hardware that may affect the software’s interface with the system, outputs, diagnostic or error messages or alarms.

4.2.4 Normal termination

The SUM shall describe how the user can cease or interrupt use of the software and how to determine whether normal termination or cessation has occurred.

4.2.5 Error conditions

The SUM shall describe the common error conditions that can occur as a result of executing a function, and how to detect that the error has occurred.

4.2.6 Recover runs

No recovery operation should be needed to re-run BICAS after error, besides possibly deleting corrupt output datasets produced by the s/w before the failure.



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 13 / 19 -

4.2.7 Help method

The SUM shall include the help information about method installation, in terms of the actions to be performed by the user, how to invoke a function, possible errors, how to resolve them and what results to expect.

The command

```
bicas --help
```

, which is specified in the RCS ICD (AD1), prints the possible error codes, the settings that exist and the values which are actually used by the s/w in the particular run. This includes default settings which have been overridden as described in Section 4.3.1.1.

Note that the log contains the exact command line arguments used in the call on a copy-paste-friendly format for both the bash command line and the MATLAB command line. This is useful for calling BICAS again in the exact way in order to reproduce errors or bugs.

4.2.8 Commands and operations

BICAS only receives input from the user via its CLI arguments. It can receive no user input after its invocation, i.e. during its execution. The syntax for the CLI arguments, here referred to as just the “syntax”, is compatible with the RCS ICD (AD1).

4.2.8.1 Extended CLI syntax to modify settings

BICAS supports CLI options for modifying the settings (Section 4.3) in addition to the CLI syntax required by the RCS ICD (AD1), while still being compliant with said ICD. These extra options take the form of an arbitrary number of additional options (three arguments each) on the form

```
--set <key> <value>
```

. <key> refers to exactly which BICAS setting is to be set. The <key> must already be pre-defined in `create_default_SETTINGS.m`. <value> is a string representation of the specified setting's new value. Section 4.3 for how the value relates to other ways of modifying the settings.

4.2.9 Error messages

The SUM shall list all error messages, diagnostic messages, and information messages that can occur while accomplishing any of the user's functions, including the meaning of each message and the action to be taken after each such message.

The command

```
bicas --help
```

prints, among other things, a list of recognized and standardized errors, including short explanations and their exit codes. The explanation will also be printed when the corresponding errors occur.



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 14 / 19 -

4.2.10 Software testing operations

The s/w does not have any testing mode, though the MATLAB operating mode is useful for debugging the software, see Section 3.5.2.

4.2.11 Software upgrading operations

BICAS does not have any facilities for automatic updates. Updating the software should be identical to removing the old version and installing the new one, with the exception of keeping the configuration file and master CDF files.

4.2.12 Software uninstalling operations

BICAS itself does not require any particular uninstallation procedure. The directory and files can simply be deleted.

Uninstalling MATLAB may in principle require some form of uninstallation procedure. See MATLAB documentation for help. The uninstallation procedure for MATLAB R2016a and R2019b on Linux should be to simply delete the MATLAB installation directory.

4.3 Configuration

4.3.1 Settings (and constants)

BICAS contains a uniform system for managing its own “settings”: various global configuration settings/options, and some constants, in the form of a set of key-value pairs which the user might want to be aware of, or modify, to alter the behaviour of BICAS. These settings include key-value pairs for e.g.:

- the path to the exact MATLAB executable to use (useful if having multiple MATLAB versions installed)
- the granularity of logging
- certain values in the S/W descriptor (e.g. version, author)
- calibration file (RCT) naming conventions
- configuring extra/relaxed checks on input datasets
- optionally overriding certain environment variables read by BICAS, e.g. `ROC_RCS_CAL_PATH`
- choosing between implementations, e.g. to choose between old and new algorithms, experimental algorithms, temporary implementations (“hacks”), unfinished implementations (e.g. calibration).
- Temporarily hide and unhide functionality while still keeping BICAS ICD (AD1) compliant.

etc. The settings keys available can be viewed by calling `bicas --help`. The values actually used by BICAS in any given run can be viewed at the beginning of any BICAS log file.

Note: The settings are implemented as `SETTINGS`, an internal, global MATLAB variable and (the only) instantiation of the class `settings.m`.



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 15 / 19 -

Note: The “settings” is effectively a combination of (1) global settings, which are meant to be modified by a user of the software when necessary, and (2) global constants, which are meant to only be modified by a developer of the software (e.g. version number, author). The dividing line between the two is blurry though.

Note: The BICAS settings is in large part meant to make it possible to handle situations where it is desirable to have the same, non-perfect version of BICAS simultaneously function in different ways but still do so in a benign, practical, and well-documented way that is compliant with the ICD (AD1). Therefore, the set of keys, the exact meaning of their values, and their default values, is not meant to be fixed or maybe not even converge over time, but is expected to evolve and change to smoothly handle different situations over the lifetime of BICAS.

4.3.1.1 Ways in which BICAS obtains settings and their priority order

The settings key values can be obtained from the three different sources below. The value for each particular key is retrieved, in priority order, from the first source where it is found to be set.

1. CLI arguments (Section 4.2.8.1) (highest priority). Later arguments have priority over earlier arguments.
2. The BICAS configuration file
3. Hardcoded default values, set in `create_default_SETTINGS.m` (lowest priority). Note: The complete set of permitted/available settings keys is defined here.

Exception to the above priority order: The key value for the MATLAB executable, key `MATLAB_COMMAND`, has to be set in the BICAS configuration file so that the bash wrapper script can read it and launch the correct MATLAB version.

4.3.2 Additional hard-coded “error-safe constants”

`error_safe_constants.m` contains a smaller number of global constants that must be available early in the execution and in the event of error, e.g. error codes and the required MATLAB version, and are therefore implemented separately. There is no provision for modifying these values without directly altering the source code.

4.4 Make BICAS generate the RCS (S/W) descriptor

BICAS can be made to generate the S/W descriptor on stdout by calling it using the CLI argument

```
--swdescriptor
```

. This is not required by the RCS ICD (AD1). This is useful for (1) generating the default descriptor file rather than manually editing it, (2) making sure that the descriptor file is in sync with BICAS. This is the authoritative list of valid s/w modes and their corresponding input and output datasets. If the tables of input and output datasets in this document, or

ROC-PRO-SFT-SUM-00080-IRF_Iss01_Rev03(BICAS_User_Manual).docx



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 16 / 19 -

`descriptor.json`, are inconsistent with the s/w descriptor generated by the above call, then the s/w descriptor generated by the above call applies.

5 TUTORIAL

5.1 Introduction

The SUM shall describe how to use the software and what the software does, combining tutorials and reference information for both novices and experts.

5.2 Getting started

The SUM shall include a welcoming introduction to the software.

5.3 Using the software on a typical task

The SUM shall describe a typical use case of the software. Graphical pictures and diagrams to demonstrate the actions performed by the user can be used.

6 APPENDICES

6.1 Troubleshooting & know issues

6.1.1 Known limitations, caveats

This section lists known current limitations and caveats. Most of the items are likely less important. Most or all of these limitations represent features that should be implemented eventually. Within each category, entries are sorted from most important to least important.

List last updated 05/05/2020.

- Reading CDF files:
 - BICAS ignores input CDF pad values (not to be confused with fill values) since all or most input master CDF zVariables specify pad value 0 which is a legitimate data value.
 - BICAS might not check explicitly if an input CDF file (depends on setting), or master CDF file appears to be of the right type of dataset (dataset ID) before trying to process the information. Note: Many input datasets (i.e. *not* generated by BICAS) found in the wild have not set the CDF global attributes needed to pass this check.
- Processing:
 - Output datasets may not have bias currents due to how input data is supplied to BICAS. Explanation: Input bias current datasets only contain values for when bias currents are explicitly set via TC, which is very rare (days between each occasion). BICAS can therefore not derive bias currents to calibrated (TC/TM) for timestamps before the first setting of bias current contained in the input bias current dataset. The timestamps in output datasets are determined by the



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF

Issue: 01

Revision: 03

Date: 06/05/2020

- 17 / 19 -

timestamps of input electric field datasets, which thus may not have any known corresponding bias current for extending periods of time (days).

- Calibration offsets are not verified and may have the wrong sign.
- LFR: zVariables `QUALITY_BITMASK`, `QUALITY_FLAG` are set to fill values with the right size when the source dataset has empty zVariables (instead of copying).
- Parasitic capacitance is not accounted for (shortcoming in the calibration of electric fields).
- Writing CDF files:
 - Global attributes:
 - `Generation_date`: Date is set in local time rather than UTC.
 - Some global attributes which should be set, and which are not inherited from the master CDF files, are not set, e.g. `SPECTRAL_RANGE_MIN`, `SPECTRAL_RANGE_MAX`, `TIME_MIN`, `TIME_MAX`.
 - Some global attributes are not set:
 - `CAL_ENTITY_NAME`,
 - `CAL_ENTITY_AFFILIATION`,
 - `CAL_EQUIPMENT`,
 - `CALIBRATION_TABLE`,
 - `CALIBRATION_VERSION`, and possibly more
 - Note: Failure to set global attributes can also be due to missing global attribute values in the parent datasets.
 - Some global attributes may in principle be wrong if using a s/w mode with multiple output datasets (hypothetical since no such s/w mode exists yet).
 - BICAS ignores output CDF pad values (not to be confused with fill values) since all or most output master CDF zVariables specify pad value 0 which is a legitimate data value.
 - BICAS cannot produce CDF files with zVariables with zero records.

7 LIST OF TBC/TBD/TBWs

TBC/TBD/TBW			
Reference/Page/Location	Description	Type	Status



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF
Issue: 01
Revision: 03
Date: 06/05/2020



BIAS Calibration Software (BICAS) User Manual

Ref: ROC-PRO-SFT-SUM-00080-IRF
 Issue: 01
 Revision: 03
 Date: 06/05/2020

8 DISTRIBUTION LIST

<p style="text-align: center;">LISTS</p> <p>See Contents lists in “Baghera Web”: Project’s informations / Project’s actors / RPW_actors.xls and tab with the name of the list or NAMES below</p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px;">Tech_LESIA</td></tr> <tr><td style="padding: 2px;">Tech_MEB</td></tr> <tr><td style="padding: 2px;">Tech_RPW</td></tr> <tr><td style="padding: 2px;">[Lead-]Cols</td></tr> <tr><td style="padding: 2px;">Science-Cols</td></tr> </table>	Tech_LESIA	Tech_MEB	Tech_RPW	[Lead-]Cols	Science-Cols
Tech_LESIA						
Tech_MEB						
Tech_RPW						
[Lead-]Cols						
Science-Cols						

INTERNAL

LESIA CNRS		

LESIA CNRS		

EXTERNAL (To modify if necessary)

		C. FIACHETTI
		C. LAFFAYE
		R.LLORCA-CEJUDO
		E.LOURME
		M-O. MARCHE
		E.GUILHEM
		J.PANH
		B.PONTET
CNES		
		L. BYLANDER
		C.CULLY
		A.ERIKSSON
		SE.JANSSON
		A.VAIVADS
IRFU		
		P. FERGEAU
		G. JANNET
		T.DUDOK de WIT
		M. KRETZSCHMAR
		V. KRASNOSELSKIKH
LPC2E		
		S.BALE
SSL		

		J.BRINEK
		P.HELLINGER
		D.HERCIK
		P.TRAVNICEK
		J.BASE
		J. CHUM
		I. KOLMASOVA
		O.SANTOLIK
		J. SOUCEK
		L.UHLIR
		G.LAKY
		T.OSWALD
		H. OTTACHER
		H. RUCKER
		M.SAMPL
		M. STELLER
		T.CHUST
		A. JEANDET
		P.LEROY
		M.MORLOT
Asi/CSRC		
IAP		
IWF		
LPP		