

# Manual for Blick Software Suite 1.3

Version 7, 20 Apr 2017

Alexander Cede



## **Contents**

Do	Document Change Record  Acronyms and Abbreviations		
A			
1	Intr	roduction	8
	1.1	About this Manual	8
	1.2	Overview	8
2	Soft	tware Installation	10
_	2.1	Computer Requirements	10
	2.2	Blick Software Suite Installation	10
	2.3	Blick Software Suite Directory Structure	11
	2.4	Python Installation	12
	2.5	·	14
•	Dl: a	le Coffenana Cuita Daganintian	1.
3		k Software Suite Description	<b>16</b>
	3.1	BlickO - Instrument Operation	16
		3.1.1 BlickO Startup	17
			18
			19
			20
	3.2	3.1.5 BlickO operations	20
	3.2	3.2.1 BlickF Start and Stop	22
		3.2.2 BlickF File Handling task	22
		3.2.2 BlickF The Handing task	23
	3.3	BlickP - Data Processing	23
	3.3	Blickr - Data Flocessing	<b>4</b>
4		k Software Suite Operations Setup	25
	4.1	Routines	25
		4.1.1 BlickO Standard Routine Library	25
		4.1.2 Routine Syntax	29
		4.1.3 Routine Command DESCRIPTION	32
		4.1.4 Routine Command GETSCRIPT	32
		4.1.5 Routine Command COMMAND	32
		4.1.6 Routine Command DURATION	33
		4.1.7 Routine Command SET FILTERWHEELS	33
		4.1.8 Routine Command SET POINTING	34
		4.1.9 Routine Command SET SPECTROMETER	35
		4.1.10 Routine Command MEASURE	36
		4.1.11 Routine Command CHECK INTENSITY	37
		4.1.12 Routine Command PROCESSINFO	37
		4.1.13 Routine Commands START LOOP and STOP LOOP	38
	4.2	Schedules	39

5	Inpu	at Output Data Specifications 44
	5.1	Meta Data
	5.2	Locations File
		5.2.1 Format
		5.2.2 Adding/Updating a Locations File Entry
		5.2.3 Rules for Constructing a Locations File Entry
	5.3	Instrument Operation File
	5.4	BlickO General Configuration Files
	5.5	BlickO Specific Configuration File
	5.6	BlickF Configuration File
	5.7	L0 File
	3.1	5.7.1 Full L0 File
		5.7.2 Partial L0 File
	5.8	BlickO Status File / Status Line
	5.9	C
		5.9.1 Section [paths]
		5.9.2 Section [retrieval]
	<b>5</b> 10	5.9.3 Section [retrieval] examples
	5.10	Processing Setups File
		5.10.1 Table "Trace Gases"
		5.10.2 Tables "s-codes" and "qs-codes"
		5.10.3 Tables "f-codes" and "qf-codes"
		5.10.4 Tables "r-codes" and "qr-codes"
		5.10.5 Adding new Entries to the Processing Setups File
		Instrument Calibration File
		L1 File
	5.13	L2 Files
		5.13.1 L2Fit Files
		5.13.2 L2Tot and L2Trop Files
		Log Files
	5.15	Alignment Files
6	_	orithms Theoretical Basis 103
	6.1	Alignment Algorithm
		6.1.1 Theoretical Background
		6.1.2 Practical Implementation
		6.1.3 Multiple FOVs
	6.2	Signal to Noise Ratio Optimization
		6.2.1 Theoretical Background
		6.2.2 Single Scan Dark Correction
		6.2.3 Multiple Scans Dark Correction
		6.2.4 Dark Map
		6.2.5 Optimized SNR
		6.2.6 Blind Pixels
	6.3	L1 Algorithm - Data Correction
		6.3.1 Dark Correction
		6.3.2 Non-Linearity Correction
		633 Latency Correction 124

6.3.4	Flat Field Correction	126
6.3.5	Conversion to Count Rates	127
6.3.6	Temperature Correction	127
6.3.7	Stray Light Correction	128
6.3.8	Sensitivity correction	128
6.3.9	Conversion to (Ir)radiances	129
6.3.10	Wavelength Correction	130
L2Fit A	Algorithm - Spectral Fitting	131
6.4.1	Lambert-Beer's law	131
6.4.2	Convolution	131
6.4.3	Slant Optical Depth	132
6.4.4	Basic Fitting Equation	135
6.4.5	Terms of Basic Fitting Equation	136
6.4.6	Least Squares Minimization	137
6.4.7	Linear Fitting	138
6.4.8	Non-linear Fitting	140
L2 Alg	gorithms	141
6.5.1	Direct Algorithm	141
6.5.2	Five Angles O2O2-Ratio Algorithm	141
erences		146
	6.3.5 6.3.6 6.3.7 6.3.8 6.3.9 6.3.10 L2Fit 4 6.4.1 6.4.2 6.4.3 6.4.4 6.4.5 6.4.6 6.4.7 6.4.8 L2 Alg	6.3.5 Conversion to Count Rates 6.3.6 Temperature Correction 6.3.7 Stray Light Correction 6.3.8 Sensitivity correction 6.3.9 Conversion to (Ir)radiances 6.3.10 Wavelength Correction L2Fit Algorithm - Spectral Fitting 6.4.1 Lambert-Beer's law 6.4.2 Convolution 6.4.3 Slant Optical Depth 6.4.4 Basic Fitting Equation 6.4.5 Terms of Basic Fitting Equation 6.4.6 Least Squares Minimization 6.4.7 Linear Fitting 6.4.8 Non-linear Fitting L2 Algorithms 6.5.1 Direct Algorithm 6.5.2 Five Angles O2O2-Ratio Algorithm



# **Document Change Record**

Issue	Date	Section	Modifications
1	21 Dec 2014	All	First version
2	25 Mar 2015	All	Started expanding L2Fit algorithm, several minor modifications
3	31 Jul 2015	All	Added several sections and applied corrections suggested by Thorsten Fehr and Stefano Casadio
4	30 Jan 2016	All	Expanded BlickF description, IO section and algorithm section; introduced s-, f-, and r-codes
5	7 Aug 2016	All	Completed description of L0, L1, L2 files and Processing Setup
6	15 Jan 2017	All	All sections filled; first complete version
7	20 Apr 2017	IO	More description in IO section (calibration file, BlickP config-file)



## **Acronyms and Abbreviations**

2D Two dimensional
LuftBlick LuftBlick OG
AC Alternating Current
AD Analog-to-digital
AMF Air mass factor

ASCII American Standard Code for Information Interchange

BC Bright counts

Blick file-transfer and operation-monitoring software

BlickO Blick operating software
BlickP Blick processing software
BlickSFA Blick spectral fitting algorithm
CCD Charge-Coupled Device

CMOS Complementary Metal-Oxide–Semiconductor

DC Dark counts
DN Digital number
DN Digital number

DOAS Differential Optical Absorption Spectroscopy

DQ Data quality
DQ0 Data quality 0
DQ1 Data quality 1
DQ2 Data quality 2
DQF Data quality flag

ESA European Space Agency

FEL lamp ANSI standard 1000 watt quartz halogen lamp

**FOV** Field of View Fitting Setup Entry **FSE** FTP File Transfer Protocol **FWHM** Full width at half maximum Global Positioning System **GPS GUI** Graphical User Interface **HDF** Hierarchical Data Format **ICF** Instrument Calibration File **IOF Instrument Operation File** IP address Internet protocol address

ISO International Organization for Standardization

JPEG Joint Photographic Experts Group

L0 Level 0 L1 Level 1 L2 Level 2

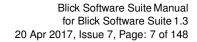
L2Fit Level 2 spectral fitting results file L2H Level 2 files other than L2Fit L2Tot Level 2 total columns file

L2Trop Level 2 tropospheric information file

NASA National Aeronautics and Space Administration

OD Optical Depth

ODR Optimized Dark Ratio





Pandora Pandora spectrometer system

PAZ pointing azimuth

PDF Portable Document Format
PRNU Pixel Response Non Uniformity

PZA Pointing zenith angle
RAM Random-access memory
rms Root Mean Square
ROE Readout electronics

RTC Radiative Transfer Calculations SkyFOV Field of view for sky observations

SNR Signal to noise ratio

SunFOV Field of view for direct sun observations

SZA Solar Zenith Angle
USB Universal Serial Bus
UT Universal Time

VZA Viewing Zenith Angle



## 1 Introduction

## 1.1 About this Manual

This manual is an official document of

LuftBlick OG Kreith 39a 6162 Mutters Austria

It describes the Blick Software Suite, which operates the Pandora spectrometer system (Pandora) and other similar instrumentation. It can be downloaded as PDF from http://pandonia.net/docs/manuals\_reports\_notes/. Note that hardware related information about Pandora such as instrument installation and maintenance is not described in this manual. For this refer to the "SciGlob Pandora Installation Manual" [2], which can be downloaded from the same site.

The Blick Software Suite runs under the following open source license:

"Only under the following conditions you are free to share, copy, distribute, transmit the work or adapt it for your purpose:

- You may not use this work or modified versions of it for commercial purposes.
- If you alter, transform, or build upon this work, you may distribute the resulting work only under the same license to this one."

For questions contact LuftBlick OG (LuftBlick) under office@luftblick.at.

## 1.2 Overview

The Pandora project was initiated in 2006 at the Atmospheric Chemistry and Dynamics Laboratory, Code 614, at NASA Goddard Space Flight Center under principal investigator Jay Herman. NASA has continuously funded Pandora-related projects ever since. Since 2011 Pandora is manufactured by SciGlob Instruments and Services, Maryland, USA (see http://sciglob.com/). Starting in 2013, ESA started to fund Pandora-related projects under principal investigator Alexander Cede. LuftBlick is responsible for Pandora-related software development since 2010 with funding from NASA and ESA projects. This manual describes the Blick Software Suite version 1.3.

The main part of the software and the entire manual are written by Alexander Cede. The other team members, who have contributed to this manual, are:

- Nader Abuhassan, main responsible for instrument hardware
- Martin Tiefengraber, instrument and algorithm research
- · Matt Kowalewski, calibration and hardware
- Christian Retscher, special software support, data processing
- Elena Spinei, algorithm development
- Moritz Müller, operation, calibration and instrument research



- Daniel Santana, network operator, calibration and instrument research
- Christian Posch, software and network support
- Axel Kreuter, algorithm development

There are three separate pieces of software described in this manual:



#### **BlickO**

The Blick operating software reads the Instrument Operation File (IOF), connects to the instrument hardware, changes filterwheel positions, moves the tracker, operates the camera, controls the temperature box, takes spectroscopic measurements, saves the raw signals, called Level 0 (L0) data, and provides a graphical user interface (GUI) for the user to display current data and operate Pandora.



#### **BlickF**

The Blick file-transfer and operation-monitoring software serves two purposes. One task is to automatically push files from a local directory to a remote directory at a user selected frequency. The other task is to monitor BlickO and restart it if needed.



#### BlickP

The Blick processing software reads the IOF and Instrument Calibration File (ICF), reads the L0 data and converts them into Level 1 (L1) data, which are corrected signals, i.e. all instrumental corrections applied on the L0 data. It also creates different types of Level 2 (L2) data (slant column amounts, vertical column amount, surface concentrations, etc.).

The Blick Software Suite is written in the open source programming language Python (http://python. org/). The Windows setup files and other necessary files are freely available from http://pandonia. net/docs/.



## 2 Software Installation

## 2.1 Computer Requirements

Although Python is a platform independent programming language, BlickO has to run on Windows NT or later for most Pandora systems, since most spectrometers used are currently controlled through a manufacturer supplied Windows DLL. BlickF and BlickP are platform independent. No minimum requirements for RAM or processor speed are needed. In order to run a Pandora, the operating computer needs to have one USB connection for each spectrometer, one USB or serial connection to control the head sensor and one USB or serial connection for the temperature control.

## 2.2 Blick Software Suite Installation

If the user decides to run the Blick Software Suite on a Windows machine and does not intend to make modifications to the source code, it is not necessary to install the Python programming language. The Blick Software Suite can be installed from a setup file and run from the executable files (e.g. for BlickO the file /bin/BlickO.exe). The starting slash / stands for the installation directory, e.g. C:/Blick/. If the user wants to use another operating system (e.g. Linux) or make modifications to the source code, he first needs to install the Python programming language on his computer. Section 2.4 describes how to install Python with all the modules needed to run the Blick Software Suite. It is also useful to make some modifications on the computer running BlickO. In section 2.5 we give a list of recommendations for this purpose.

Here we explain how to install the Blick Software Suite on a Windows machine. On another platform the whole directory structure as described in section 2.3 has to be copied (except for /bin/). We recommend installing the Blick Software Suite before connecting the USB or serial interfaces of a Pandora spectrometer system, since the necessary drivers are installed during the setup process (see also *Abuhassan et al.* [2]).

Execute the setup file Blick-OPF-1.3.\*\_setup.exe. Choose the directory for the installation (we recommend C:/Blick/). The chosen directory should NOT have write protection. When finished, icons for BlickO, BlickF, and BlickP will be created on the desktop. To run any of the programs just double-click the respective icon, or directly the executable file (/bin/BlickO.exe, /bin/BlickF.exe and /bin/BlickP.exe respectively).

If the Python programming language is installed on the computer, BlickO, BlickP, or BlickF can also be started by double-clicking the Python files blick\_osmain.pyw, blickp.py, or blick\_filepush.py respectively, which are all located in the source code directory /src/. Finally, one could also use a Python editor (e.g. the one listed in section 2.4) to load one of the above files, and then run or debug it. Only when source code modifications are made and the software is being debugged do we recommend using the last editor method. For operational use, this method has often caused the computer to "hang" after a while, since the editors usually have multiple additional threads running, which can cause loss of control.

When starting BlickO, a dialog appears saying "Choose an instrument operation file". For the first use of BlickO, you can choose Pandora 0, which is a virtual Pandora unit used for testing purposes, or you can click "Cancel" and then copy the IOF of the Pandora unit you want to run into directory /data/operationfiles/ (see section 5.3).

To upgrade to a newer Blick Software Suite version, just execute the new setup file and choose the same software directory (allow it to overwrite). The source code, executables, routines, etc. will then be updated, while all data files such as alignment files, L0 files, etc. will be untouched. If the setup process stops, because a previously installed software version is write-protected, we recommend stopping the setup process, then uninstalling the previous version (execute file /unins000.exe), and then executing the setup file again.

To connect to the Pandora unit after successful installation of the Blick Software Suite, use the BlickO GUI "Connection buttons" as described in section 3.1.



## 2.3 Blick Software Suite Directory Structure

This section describes the directory structure of the Blick Software Suite, which is automatically created when installing from a setup file as described in section 2. In this case, the files /unins000.dat, and /unins000.exe are also created in the installation directory. These files are used if the software is uninstalled or if a newer version is installed.

Below is a complete list of the directories used in the Blick Software Suite. A starting slash / stands for the installation directory, which can be chosen by the user during the setup process (e.g. C:/Blick/).

#### /src

This directory contains the Python source code files (blick\_\*.py or blick\_\*.pyw) and the compiled Python files (blick\_\*.pyc).

#### /lib

This directory contains subdirectories /oslib, /pslib and /fslib, which contain a variety of files needed to run BlickO, BlickP or BlickF respectively.

## /lib/routines

This directory contains "routine files" (XY.rout, where XY is any two letter or one letter plus one number combination). A "routine" is a sequence of commands for a Pandora unit (e.g. direct sun observations SO). For more details see sections 3.1 and 4.1.

#### /lib/schedules

This directory contains "schedule files" (\* . sked). A "schedule" is a sequence of routines to be executed over the course of the day. For more details see sections 3.1 and 4.2.

## /log

This directory contains subdirectories /oslog, /pslog and /fslog, which will be filled by BlickO, BlickP or BlickF respectively with "error-log-files" and "info-log-files" (\*\_errorlog.txt and \*\_infolog.txt). Each time an error occurs, an error-log entry is added to the error-log-file. The info-log-file is filled with information like "Started schedule XY", "Connected tracker", etc.

## /data/tmp, /data/L0, /data/L1, /data/L2

These directories contain contain temporary data files and L0, L1 and L2 data files respectively. Raw data files produced by BlickO go initially into directory /data/tmp and are then pushed to directory /data/t0 once they are finished. Care should be taken when opening/reading these temporary files in directory /data/tmp, since they are currently in use and are permanently written on.

## /data/operationfiles

Instrument operation files (\*\_of.txt) contain the hardware information for a specific Pandora unit (section 5.3). In order to run BlickO or BlickP, the IOF must be present.

## /data/calibrationfiles

Instrument calibration files (\*\_cf.txt) contain the calibration data for a specific Pandora unit (section 5.11). In order to run BlickP the correct ICF must be present.

## /data/alignments

This directory has the "alignment files" (\*\_alignments.txt or



\*\_alignments\_YYYYMMDDTHHMMSSZ.txt), which contain results of the "alignment routines". These routines are sun (or moon) scanning sequences that allow determining the exact pointing of the instrument towards the sun or moon. Some scanning routines also produce pictures, which are also stored in this directory. More details can be found in section 5.15.

## /data/diagnostic

This directory will be filled with diagnostic data (or figures) of a Pandora unit, e.g. the temperature evolution over a day, or the count rate at a specific wavelength.

## /config

This directory contains the "configuration files". These files list parameters, which are configured by the user to run BlickO, BlickP or BlickF, e.g. the size of the GUI window or the last location selected. For BlickO these settings can be changed during software operation. At the end of each BlickO session, the configuration file is updated. The configuration files are described in sections 5.4, 5.5, 5.6, and 5.9.

#### /doc

This directory contains documentation, e.g. this manual itself.

#### /bin

This directory contains the executable files BlickO.exe, BlickP.exe, and BlickF.exe and many other files (mostly .dll and .pyd files). It is not needed if the software is run directly from the Python file in the source code.

## 2.4 Python Installation

Python consists of a basic software package and additional modules for specific purposes. These step-by-step instructions show how to download and install all modules needed in the Blick Software Suite. The files listed for the download are the Windows versions currently used in the Blick Software Suite. Other platforms require different files, which should also be found in the same internet links. There is a good chance, that newer versions for each module are available. These newer versions should in principle also work fine, but it is not guaranteed. In general we recommend staying away from versions with subversions number 0 or 1 (e.g. version 3.5.1), since we have previously encountered simple bugs in these versions, which are typically quickly corrected in sub-version number 2. It is best to stay with the original version of Python until newer versions are tested. Note that it does not matter whether the Python modules are installed before or after the Blick Software Suite installation.

#### Program language "Python"

https://www.python.org/downloads/release/python-2710/ Save <python-2.7.10.msi> (from Windows x86 MSI installer) Execute msi-file, use default directory C:/Python27/

## Module for numerical functions "NumPy"

http://sourceforge.net/projects/numpy/files/NumPy/1.7.1/ Save <numpy-1.7.1-win32-superpack-python2.7.exe> Execute msi-file, use default directory C:/Python27/Lib/Site-packages/

## Module for mathematical functions "SciPy"

http://sourceforge.net/projects/scipy/files/scipy/0.12.0/



Save <scipy-0.12.0-win32-superpack-python2.7.exe> Execute exe-file, use default directory C:/Python27/Lib/Site-packages/

#### Module for GUI functions "wxpython"

http://sourceforge.net/projects/wxpython/files/wxPython/2.8.12.1/ Save <wxPython2.8-win32-unicode-2.8.12.1-py27.exe> Execute file, use default directory C:/Python27/Lib/Site-packages/

#### Module for serial interface "pyserial"

Unpack <pyserial-2.6.tar.gz> to get <pyserial-2.6.tar>

Unpack <pyserial-2.6.tar>

In a command window change to the directory of the unpacked .../pyserial-2.6/

Type "C:/Python27/python.exe setup.py install"

Type "exit" to quit the command window

#### Module to use HDF "h5py"

https://pypi.python.org/pypi/h5py/2.3.1 Save <h5py-2.1.3.win32-py2.7.msi> Execute file, use default directory C:/Python27/Lib/Site-packages/

## Module for Java access "JPype"

(only needed for instruments with "Spectrometer read out type -> OcOpt1", see section 5.3)

http://sourceforge.net/projects/jpype/files/JPype/0.5.4/

Save <JPype-0.5.4.2.win32-py2.7.exe>

Execute file, use default directory C:/Python27/Lib/Site-packages/

(This module also needs the Java Development Kit, which can be obtained from http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1637583.htmlforwindows<jdk-7u5-windows-i586-p.exe>)

## Imaging Library "PIL"

http://www.pythonware.com/products/pil/ Save <PIL-1.1.7.win32-py2.7.exe>

Execute file, use default directory C:/Python27/Lib/Site-packages/

#### Camera module "OpenCV"

(only needed for instruments with "Camera read module -> OpenCV", see section 5.3)

http://opencv.org/downloads.html

Save < OpenCV-2.4.6.0.exe>

Execute file and extract files into a temporary directory

Copy file <.../opencv/build/python/2.7/cv2.pyd> in directory C:/Python27/Lib/Site-packages/

## Camera module "VideoCapture"

(only needed for instruments with "Camera read module -> DirectX", see section 5.3)

http://videocapture.sourceforge.net/

Save <VideoCapture-0.9-4.zip> and unzip it

Copy content of /Python27/DLLs/ in directory C:/Python27/DLLs/



```
Copy content of /Python27/Lib/ in directory C:/Python27/Lib/
In file C:/Python27/Lib/VideoCapture.py replace lines
self.normalfont = ImageFont.load_path('helvetica-10.pil')
self.boldfont = ImageFont.load_path('helvB08.pil')
with these lines
self.normalfont = []
self.boldfont = []
```

#### An optional free editor

http://sourceforge.net/projects/spe/
Save <SPE-0.8.2.a-wx2.6.1.0-py24.exe>
Execute file, use default directory C:/Python27/Lib/Site-packages/
As a shortcut you can pull "SPE.pyw" from C:/Python27/Lib/Site-packages/ onto the desktop

## 2.5 Recommended Computer Settings

This section contains recommendations, which we have found very useful, when running a Pandora spectrometer system. Although they are for Windows operating systems only, it might be useful to read them for other platform installations too in order to understand the basic concept behind them. Basically, these settings keep routine computer operating system operations from interfering with Pandora operations.

#### **Antivirus**

Choose an antivirus program that does not ask you too frequently for restarts of your computer.

#### Firewall and updates

- 1) Turn on Windows Firewall (Control Panel Windows Firewall Turn Windows Firewall on or off)
- 2) Download all windows updates. Repeat the updating as often as needed until no updates are found.
- 3) Set automatic updates: Control Panel Windows Update Change Settings Check "Download them but let me choose ...". Do Windows updates manually, preferably during night.

#### **Power options**

Go to Control Panel – Power Options - Change Plan Settings - Change advanced power settings - set options to...

- ... Hard disk: Turn off hard disk after: Never
- ... Sleep: Sleep after: Never, Hibernate after: Never

monitor after 15 minutes, never turn off hard disk or do standby

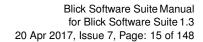
- ... Power buttons and lid: Lid close action: Do Nothing.
- ... Display: Dim display after: 5 min, Turn off display after: 10 min

#### Internet time

In order to increase the update interval frequency (one week by default) do the following:

- 1) Use regedit.exe by writing "regedit" in Start Run or where it says "Start Search"
- 2) Go to HKEY\_LOCAL\_MACHINE/SYSTEM/CurrentControlSet/Services/W32Time/TimeProviders/NtpClient
- 3) Double click on "SpecialPollInterval" in the right hand panel default is 604800 (7 days) in seconds (Decimal). Change it to 40000 (<12 Hours).

Note: the computer time zone does not have to be set to UT. It can be the preferred local time of the operator.





## Task manager

Go to Control Panel – Administrative Tools - Task Scheduler - Task scheduler library Remove unnecessary tasks, especially those that say under "Triggers": "After triggered, repeat every XX:XX for a duration of YY." or even "...repeat indefinitely..." (e.g. Adobe Flash Player Updater)



## 3 Blick Software Suite Description

## 3.1 BlickO - Instrument Operation

## 3.1.1 BlickO Startup

When BlickO is started, the user is first asked to select an IOF in a file selection dialog (figure 1). If a different IOF is selected then at the previous use of BlickO, a warning message will appear in order to avoid that users unintentionally select a wrong IOF. Another warning message might appear, in the case the selected instrument is already in use. Note that this warning message can also appear in the case BlickO was not closed properly in the previous use of the software.

Then BlickO checks internet connections, power status etc. of the operating computer, which can take a few seconds. If the previous use of BlickO was at another IP address than the current one or has been more than a full day ago, another warning message appear to remind the user, that he might be at a different location now.

Note: BlickO also writes to and reads from the so-called "BlickO-BlickF Messenger File" /log/Blick\_message.bin in order to communicate with BlickF. The meaning of this file is described in section 3.2.

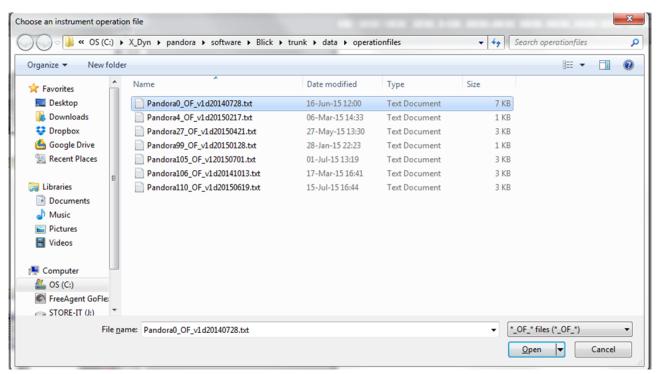


Figure 1: File dialog to select an IOF.



## 3.1.2 GUI Overview

Figure 2 shows the GUI screen as it appears after starting BlickO and selecting an IOF, in this case for Pandora 0. The different sections of the BlickO GUI (called GUI-sections) are marked by red numbers on figure 2 and are listed here:

- 1. Instrument + location label
- 2. Label with date, time, and solar/lunar angles
- 3. Connection status labels
- 4. Connection buttons for all interfaces
- 5. Auxiliary data display
- 6. Display of sun-search results (at software startup just contains an image of the sun)
- 7. Action logger
- 8. Figure showing current measurements for all spectrometers
- 9. Routine control, load schedule button, and reset button
- 10. Start-stop button
- 11. Spectrometer settings: repetition control, integration time control, and cycles control
- 12. Filterwheel settings: filterwheel 1 selection and filterwheel 2 selection
- 13. Tracker settings: zenith angle control, azimuth control, track-sun button, and track-moon button
- 14. Save figure button, save comment button, and exit button

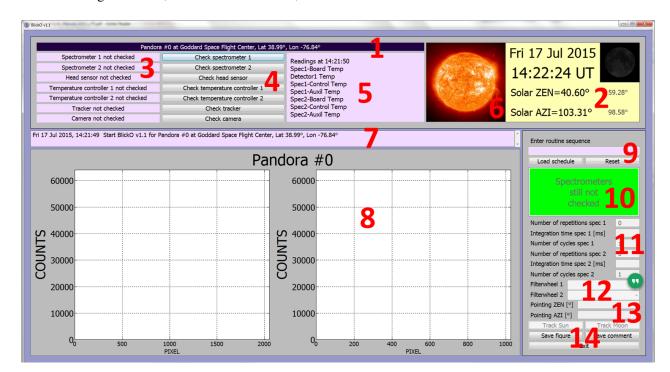


Figure 2: GUI of BlickO in real mode



## 3.1.3 Simulation mode

The purple background in figure 2 indicates that the "Instrument operation mode" for BlickO is set to "real mode", which means the connected hardware will "really" be accessed through the connection ports. BlickO can also be run in a "simulation mode", in which case the GUI would have a red background and the hardware connections will just be simulated (figure 3). The instrument operation mode is set in the BlickO specific configuration file (section 5.5). A value of 1 means real mode with the instrument not in use, 2 means real mode with the instrument in use, and 0 means simulation mode. So in order to change BlickO from real to simulation mode (simulation to real mode), the configuration file has to be manually edited and the line with "Instrument operation mode" has to be changed from 1 to 0 (0 to 1) before starting BlickO. In real mode, BlickO automatically changes the instrument operation mode to 2 at start-up and then back to 1 at program exit. This is to prevent the user from unintentionally trying to start BlickO on the same instrument twice.

All functions of BlickO can be tested in simulation mode. For users who want to modify the source code, it is recommended to test their modified code in the simulation mode first before proceeding to the real mode.

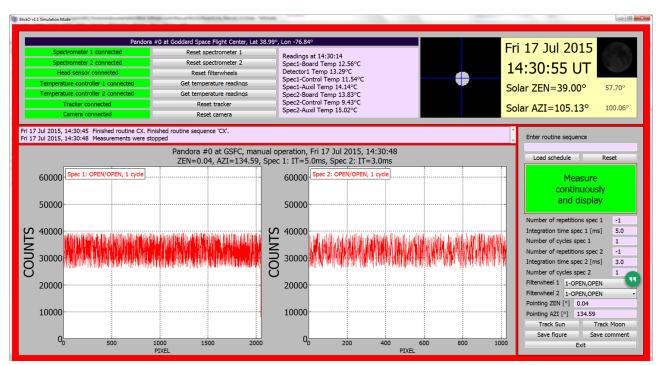


Figure 3: GUI of BlickO in simulation mode



## 3.1.4 GUI sections

GUI-section 1 shows the instrument name and the name, latitude and longitude of the currently selected location.

GUI-section 2 shows the current date and time (UT) and the calculated solar and lunar angles based on the time and the instrument location coordinates. During daytime (between sunrise and sunset), an image of the moon is shown at the top right corner of GUI-section 2. During nighttime an image of the sun is shown.

BlickO reads the current time from the computer operating system. Hence if the time on the computer is not correct, then the solar/lunar angles will also not be correct, which causes pointing problems for the instrument. Therefore, we recommend the computer time be kept correct within  $\pm 10$  s, preferably within  $\pm 1$  s by periodic updating through the internet (see section 2.5). The computer running BlickO does not necessarily have to be in UT time-zone. As long as the computer-time and the time-zone match, BlickO will work fine.

At BlickO start, GUI-section 6 shows an image of the sun (daytime) or the moon (nighttime). Once a successful target search has been done, either be scanning the sun or the moon (see section 6.1) or by taking a picture with the camera, the result of this last target search are displayed.

Once the hardware is connected and powered on, the connection buttons (GUI-section 4) are used to establish connection with the hardware pieces (see section 3.1.5). While the connection is being established, the respective connection status label (GUI-section 3) turns yellow. After a successful connection it turns green (figure 3). Otherwise an error message will appear and the label turns red.

GUI-section 5 displays the last readings of Pandora's auxiliary data (currently only temperatures). The action logger (GUI-section 7) is permanently updated and displays the last action started by BlickO (e.g. "connected tracker" or "start tracking sun").

The figure (GUI-section 8) will display spectral measurements (in counts versus pixel number or wavelength) once they are taken. In the case there are more than one spectrometer in the Pandora unit, the panels can be arranged horizontally (as in figures 2 and 3) or vertically or the two spectra can be plotted in the same panel. This can be changed in the BlickO specific configuration file (section 5.5). There are some mouse-click operations, which can be done on the figure:

- A right-click on the figure toggles the y-scale between "full range" (for measured spectra this is from 0 to the unit's effective saturation limit) and "optimized range" (from the minimum to the maximum of the measured counts with a little margin at both sides).
- The user can also zoom into the figure. This is done by a left-click of the mouse on a point of the figure, then dragging the mouse while the left button is pressed, and then releasing the mouse.
- If the figure has a legend, then a double-right-click on the figure moves the legend to different places on the figure (e.g. top left corner, top right corner, etc.).
- A double-left-click on a figure panel toggles between showing all figure panels or just one figure panel.
- A triple-right-click on a figure panel toggles between showing and not showing grid lines on this panel.

Pressing the "Exit" button (GUI-section 14) exits BlickO, after asking the user for confirmation. This is identical to pressing the "exit cross" on the top-right corner of the window. Pressing the "Save figure" button (GUI-section 14) opens a dialog allowing the user to choose a figure name and then saves the figure currently displayed in GUI-section 8. Pressing the "Save comment" button (GUI-section 14) opens a dialog allowing the user to enter a text line, which will then be saved in the L0 data file (section 5.7).

Pressing the "Track sun" button (GUI-section 13) will point the Pandora head sensor towards the sun, if the tracker is connected and if the sun is above the horizon. In the same way, pressing the "Track moon" button (GUI-section 13) will point the Pandora head sensor towards the moon, if the tracker is connected and if the moon is above the horizon. Entering a zenith angle or azimuth in the zenith angle control or azimuth control



box (GUI-section 13) will point the head sensor to the entered direction. The azimuth is counted clockwise starting at north. So 0° is north and 90° is east.

The combo-boxes filterwheel 1 and filterwheel 2 (GUI-section 12) can be used to select a specific filter.

Entering numbers in the spectrometer controls (GUI-section 11) will change the respective spectrometer setting. The integration time must be in ms between the minimum and maximum allowed values (listed in the IOF, section 5.3). Number of cycles is the number of spectra to be averaged before being displayed on the figure. E.g. 10 cycles means that BlickO reads 10 spectra from the spectrometer, builds average and standard deviation, and then displays the result in the figure. Number of repetitions says how many times a full set of cycles is measured, displayed, and saved in the L0 data file. It can also be set to 0 in which case no repetitions are measured, or set to -1, in which case an unlimited number of repetitions are measured, the data are displayed, but not saved.

The Start-Stop-button (GUI-section 10) is used to start and stop Pandora actions. In case the routine control (GUI-section 9) is empty, the system is said to be in "manual mode". Then the Start-Stop-button simply starts or stops spectrometer measurements using the settings in the spectrometer controls (GUI-section 11). Since Start-Stop is a non-blocking function, any action can be stopped at any time. Spectra saved in the manual mode are marked by a "\*\*" in the first column of the L0 data file (more in section 5.7).

The routine control (GUI-section 9) is used to enter a sequence of one or more two-character code routines (for more about routines see section 4.1). Pressing the "Load schedule" button (GUI-section 9) opens a dialog, which allows the user to select a schedule file (section 4.2).

The "Reset" button (GUI-section 9) simply deletes any text in the routine control, and sets the spectrometer controls to -1 repetition, minimum integration time, and 1 cycle respectively. This brings the system back to "manual mode".

## 3.1.5 BlickO operations

Connect the Pandora system by pressing the connection buttons one after another. "Check spectrometer X" (X is the spectrometer number) is a blocking function, i.e. no other action can be performed on BlickO while the spectrometer connection is established, and takes a few seconds only. "Check head sensor" and "Check temperature controller X" are non-blocking functions, i.e. other actions can be performed on BlickO while the connection is established, and also take only a few seconds each. "Check tracker" is a non-blocking function and takes up to 2 min, since it includes a full tracker reset. "Check camera" is a blocking function and takes a few seconds only.

If no connection to an interface could be established, the user has to trouble-shoot. These are the ad-hoc procedures to be taken. If a procedure is not successful, then the user should proceed to the next point.

- 1. Make another attempt to connect by pressing the connection button again.
- 2. Exit and restart BlickO and try to connect again.
- 3. Exit BlickO and check, whether the respective device is listed in the computer's device manager. If it is not listed, then disconnect and reconnect the respective USB cable until it is listed. Then restart BlickO and try to connect again.
- 4. Exit BlickO, restart the computer and then restart BlickO and try to connect again.
- 5. Turn off the external power to the optical head or the entire system and then restart.

When text is entered in the routine control, the system is said to be in "routine mode". For example, writing "SOSU" in the control and pressing the Start-Stop button will cause it to first execute routine SO and directly afterwards routine SU.



If a positive integer <100 is written at the end of a routine sequence, the sequence is repeated that many times. E.g. "SOSU3" means the combination SO and then SU is repeated 3 times, i.e. it is the same as writing "SOSUSOSUSOSU". One is also allowed to use single () brackets in the routine sequence string. In this case the strings inside the brackets are "resolved" first and then combined to one long sequence. So e.g. "(SOSU3)SB" means first 3 repetitions of the combination SO plus SU are executed followed by routine SB.

A special command for the routine control is "?". In this case a dialog appears with a short description of each routine in directory /lib/routines/ (section 4.1).

When pressing the "Load schedule" button (GUI-section 9), a file dialog opens, where the user can choose a schedule (figure 4). After a schedule file is selected, it will be displayed in the routine control after the characters "->". In this case the system is said to be in "schedule mode". After pressing the Start-Stop-button, BlickO will execute the routines listed in the selected schedule.

The user can also combine a sequence with a schedule. E.g. if the string in the routine control is "SOSU -> sun.pans", then BlickO will first execute "SOSU" and then start schedule "sun.pans".

While a change of the instrument is only possible at the start of BlickO, a change of the location can be done typing "CL" in the routine control (GUI-section 9) and then pressing enter or the Start-Stop-Button (GUI-section 10). A dialog appears allowing to select a different location. If the desired location is not listed, then the locations file has to be edited and the desired location has to be added (see section 5.2). Then BlickO has to be restarted and the location can be selected as described above.

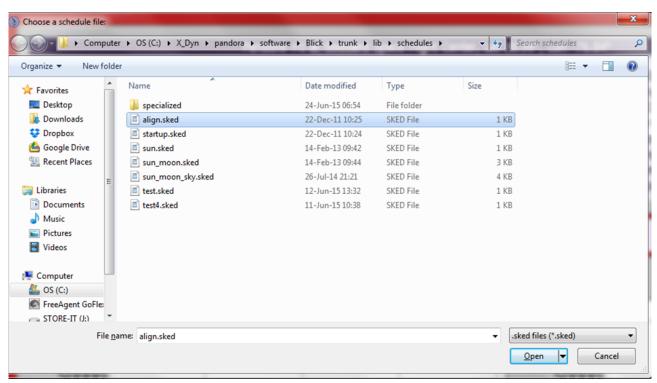


Figure 4: File dialog to select a schedule.



## 3.2 BlickF - File Handling and Operation Monitoring

The Blick file-transfer and operation-monitoring software is a command line program serving two purposes. The 'File Handling task' is used to automatically push files from a local directory to one or two remote servers (section 3.2.2). The 'Operation Monitoring task' is used to monitor BlickO and restart it if needed (section 3.2.3).

## 3.2.1 BlickF Start and Stop

When installing BlickO, BlickF will automatically be installed as well. All the necessary files are in directories /bin/ and /lib/fslib/. The BlickF configuration is given in the text-file /lib/fslib/BlickF\_config.txt. Each of the keywords in this file has to be manually edited by the user (see section 5.6).

To execute BlickF, double-click the executable file /bin/BlickF.exe or the BlickF-icon on the desktop. When BlickF is started, a command prompt window is opened (see figure 5). At first BlickF reads the BlickF Configuration File (section 5.6). If this fails, an error message is displayed in the command window. Otherwise, BlickF goes into a periodic loop at a frequency determined by the parameter POLLING\_RATE in the configuration file, typically set to 10 s. In this period BlickF will check whether it needs to do a File Handling task and/or a Operation Monitoring task and writes a corresponding line in the command window. At the end of each command line it adds the 'OMessage' and the 'FMessage' (see 3.2.3) in the form (OF=31), where the first number is the OMessage (in this case 3) and the second number the FMessage (in this case 1).

```
BlickF load modules
Sat 30 Jan 2016, 18:08:16 BlickF initialization
Sat 30 Jan 2016, 18:08:16 Deleting old partial files
Sat 30 Jan 2016, 18:08:16 Start monitoring loop
Sat 30 Jan 2016, 18:08:16 No action taken (OF=01)
Sat 30 Jan 2016, 18:08:26 No action taken (OF=01)
Sat 30 Jan 2016, 18:08:36 No action taken (OF=01)
```

Figure 5: BlickF screenshot.

Both BlickO and BlickF write to and read from the so-called "BlickO-BlickF Messenger File" /log/Blick\_message.bin, which is needed to perform the Operation Monitoring task (see section 3.2.3).

IMPORTANT: To make a clean exit for BlickF, the user shall press 'q' in the command windows at any time and not just close the command prompt window!

## 3.2.2 BlickF File Handling task

BlickF compares the file names in the local directory (given by parameter DIR\_LOCAL in the configuration file, usually /data/L0/), with the file names listed in the "already-copied-files-file" (given by parameter FILES\_COPIED in the configuration file) and pushes all files that have not already been copied in the remote directory. It will add the names of the newly copied files to the content of the already-copied-files-file.

The first file transfer will start at BlickF startup. Then a new file transfer will be done whenever new data are available.



At the very first use of BlickF, Putty's FTP client will ask the user for confirmation that the host can be trusted. Then a "y" has to be entered at the question "Store key in cache?".

The comparison of the file names in the local directory with those in the already-copied-files-file takes longer the more files are in the local directory. Therefore we recommend to "clean" the local directory at least once per year, e.g. by moving in January all L0 files from the previous year into a separate directory outside of the local directory.

## 3.2.3 BlickF Operation Monitoring task

This task is controlled by the BlickO-BlickF Messenger file /log/Blick\_message.txt. This is an ascii file containing exactly 2 characters representing a hexadecimal number, i.e. between 0 and f, and therefore has a size of exactly 2 bytes. The first character is called "OMessage" and is used by BlickO, the second number is called "FMessage" and is used by BlickF. OMessage and FMessage indicate, in which status BlickO and BlickF are. Tables 1 and 2 list the meanings of both O+F messages.

BlickF performs different actions depending on the value of OMessage as decribed in table 3. Note that OMessages 4, 5, and 7 will only ever happen, if parameter 'Do autostart' in the BlickO General Configuration File is set to 1 (see section 5.4). Furthermore the BlickF action for OMessages 5 or 7, i.e. restarting the computer, will only happen if parameter 'DO\_RESTART' in the BlickF configuration file (section 5.6) is set to 1. Otherwise BlickF will just do a BlickO restart.

FMessage Description

0 BlickF is not running

1 BlickF is running and no BlickO restart will be initiated

2 BlickF is running and will initiate a BlickO restart

3 BlickF has done a computer restart and will initiate a BlickO restart

4 BlickF has initiated to kill BlickO

Table 1: FMessage description

Table 2.	OMessage	description
Table 2.	OMICSSage	ucscription

OMessage	Description
0	BlickO is not running
1	BlickO is performing autostart
2	BlickO is running, but not in schedule mode, and L0 data access is free
3	BlickO is running in schedule mode and L0 data access is free
4	BlickO has been closed and should be restarted
5	BlickO has been closed and the whole computer should be restarted
6	BlickO is still working, but in internal recovery mode
7	BlickO has closed itself since it had an error during autostart, although the computer was previously rebooted
12=c	BlickO is running, but not in schedule mode, and L0 data access is blocked
13=d	BlickO is running in schedule mode and L0 data access is blocked



Table 3: BlickF operation monitoring actions

OMessage	BlickF action
0	Nothing
1,6	At the first occurrence of OMessage 1 or 6, BlickF takes the time. If OMessage stays at this value for more than 5 min, then BlickF writes an automated email to the network operator.
2,3	BlickF remembers the time when new files have arrived in the local L0 data directory. If no new partial file(s) arrive(s) within one minute after the arrival due date (determined by parameter "File push time [min]" in the BlickO General Configuration File (section 5.4), then the messenger file is checked again. If OMessage is not 4, 5, or 6, then BlickF kills BlickO and restarts it.
4	BlickF restarts BlickO.
5,7	BlickF sets FMessage to 2, restarts the computer and then restarts BlickO.
12=c,13=d	BlickF starts a side loop in 0.1 s intervals and once OMessage changes to 2 or 3 it does the actions described. If the OMessage never changes, then BlickF writes an automated email to the network operator.

## 3.3 BlickP - Data Processing

BlickP is a console application, which processes L0 data up to all higher data levels L1, L2Fit and L2. It needs two configuration files to operate, named /config/logging.ini and /config/BlickP.ini respectively.

The first file configures the logging system. A working logging system is provided with the Blick Software Suite distribution. The user does in general not need to edit this file. Advanced users, who may want to customize logging, are referred to the Python Standard libraries documentation https://docs.python.org/2/library/logging.config.html#configuration-file-format.

The second configuration file specifies the parameters needed for processing. This file must be edited by the used and is described in section 5.9.

When starting BlickP it expects to find both configuration files in the correct directories and starts the data processing. Each processing step is displayed in the command window and also added to a BlickP log-file (see section 5.14).



## 4 Blick Software Suite Operations Setup

## 4.1 Routines

The concept of "routines", i.e., sequences of commands to be executed by the Pandora system, is taken from the way Brewer spectrometers are operated [15]. Each routine is identified by two characters, a combination of two letters (e.g. SO) or one letter followed by one number (e.g. W1). The commands for a routine with identifier XY are written in the text file /lib/routines/XY.rout. At startup, BlickO reads through all routines in the directory /lib/routines/ and converts the content of these text files into Python code. This Python code is then executed when the routine is called, i.e. entered in the routine control (GUI-section 9).

## 4.1.1 BlickO Standard Routine Library

This section describes the BlickO standard routine library, which is stored in directory /lib/routines/during installation of BlickO. A more detailed description of each routine can be obtained entering "?" in the routine control (GUI-section 9). Most routines are part of a logical group characterized by the first letter of the routine identifier.

#### Almucantar routines A\*

AO	Almucantar with open hole
AU	Almucantar with U340 (U340 is a UV bandpass filter with maximum transmission at 340nm)

## Change system settings routines C\*

	Summer of second seconds 1 commer o		
CA	Change maximum pointing azimuth adjustment and decide whether the alignment history should be frozen		
СВ	Change the beep mode		
CE	Change positioning system parameters		
CK	Change the camera mode		
CL	Change the location		
CN	Change tracker zenith and azimuth parking positions		
СР	Change the partial file update time		
СТ	Change the temperature controller parameters for this session		
CX	Change the figure drawing parameters		

## Diagnostic routines D\*

	DC	Reads and displays the counts of routine SO in the current level 0 file at one pixel
Ī	DM	Reads and displays the counts of routine MO in the current level 0 file at one pixel
	DT	Reads and displays the temperature readings in the current level 0 file

#### Elevation scan routines E\*



EL	Detailed sky scan at fixed azimuth with U340
EK	Detailed sky scan at fixed azimuth with open hole
ЕО	Quick sky scan (5 zenith angles) at fixed azimuth with open hole hole
EU	Quick sky scan (5 zenith angles) at fixed azimuth with open U340

## Find routines F\*

	With the state of
FI	Find Initialization, moves to sun and opens a dialog, which allows the user to move the tracker manu-
	ally
FJ	Like FI but using the moon
FS	Find Sun, long sun search, saves final figure and averaged data, but not spectral data
FA	=FS, but also saving the spectral data
FD	=FS, but with diffuser in the optical path
FN	=FA, but without doing a reset in the case the difference is to large
FM	=FS, but searching moon instead of the sun
FV	Quick sun search
FQ	Quick sun search (system-optimized)
FF	Search routine with all "search options" listed

## **Help routines H\***

HE	Displays routine descriptions (same as "?")
HS	Displays a dialog to allow low level serial communication

## Camera routines K\*

KC	Calibrates camera	
KD	Displays camera images	
KK	Opens camera configuration dialog	
KS	Camera supported sun search	
KM	Camera supported moon search	
KP	Saves the last image taken by the camera in JPEG format	

## Loading routines L\*

	8		
LF Allows the user to display a previously stored figure on the figure panel		Allows the user to display a previously stored figure on the figure panel	
	LR	Reloads all the measurements routines	
	LW	Checks light source and temperature stability	

## **Direct Moon routines M\***



MO	Direct-Moon with open hole	
MU	Direct-Moon with U340	
MD	Direct moon measurements during daytime with open hole	
MT	Direct moon measurements during daytime with U340	

## Principal plane routines P\*

PO	Standard principal solar plane routine at open hole
PU	Standard principal solar plane routine with U340

## **Reset routines R\***

RF	Reset filterwheel(s)	
RO	Restart BlickO and possibly also the whole computer	
RP	Power reset	
RS	Reset spectrometer(s)	
RT	Reset tracker	

## **Direct Sun routines S\***

SO	Direct-Sun with open hole	
SU	Direct-Sun with U340	
SB	Direct-Sun with BP300 (BP300 is a UVB bandpass filter with maximum transmission at 300nm)	
SQ	Quick direct-Sun with open hole	

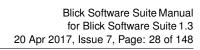
## Target, tracker or test routines T\*

ТО	Target routine using open hole	
TU	Target routine using U340	
T0	Moves the tracker to the zenith	
TD	Moves the tracker to point down (i.e. maximum allowed zenith angle) and waits for 5 secs	
TR	Moves tracker to the horizon in all four directions and to the zenith in between	
TP	Parks the tracker or wakes it up from parking	
TT	Test routine checking spectrometer(s), head sensor and tracker	

## Waiting routines W\*

W0	Wait for 2 <sup>0</sup> =1 minute	
W1	Wait for 2 <sup>1</sup> =2 minutes	
W2	Wait for $2^2$ =4 minutes	
	etc. until	
W9	Wait for 2 <sup>9</sup> =512 minutes	

## Zenith sky routines Z\*





ZO	ZO Zenith-Sky with open hole	
ZU	Zenith-Sky with U340	
ZB	Zenith-Sky with BP300	



## 4.1.2 Routine Syntax

The user can also modify routines or create new routines. In this case the source code of BlickO does not have to be changed. The only action needed is to save the (modified or new) file /lib/routines/XY.panr and then execute routine 'LR'. Note that BlickO does not check, whether the newly created routine makes sense or is potentially even damaging the instrument. E.g. if the user tells the instrument to move to the sun and measure without any attenuation filter, then the detector will be extremely over-illuminated, which produces useless data and could even harm the instrument if this is done frequently.

The routine files use a proper syntax, which is explained in this section. Errors in the syntax may be immediately detected by BlickO, in which case an error message will be displayed at the program start. If the error is not detected at that initial stage, it will in general be detected once the routine is called.

When BlickO translates the Blick syntax to Python commands, it looks for lines that start with a so-called routine command. Only those lines will be analyzed. All other lines can be considered comment lines. All possible routine commands are listed in the left column of table 4.

Some routine commands are 'self-standing' (those with no entries in column 'Related routine keywords'). The other ones are followed by the string "->" and then a series of routine keywords with associated settings, separated by ";" in the form

ROUTINE COMMAND -> KEYWORD1=VALUE1; KEYWORD2=VALUE2; ...

The right column of table 4 lists all possible keywords for a routine command. Keywords that appear in table 5, but are not listed in the command line, stay at their default setting as listed in column 3 of table 5.

Routine command	Related routine keywords
DESCRIPTION	
GETSCRIPT	
COMMAND	
DURATION	LENGTH, TIMEMODE
SET FILTERWHEELS	FUNCFILT, FW1, FW2
SET POINTING	DELTA, AZI, ZEN, AZIMODE, ZENMODE
SET SPECTROMETER	IT*, NCYCLES*, NREPETITIONS*, DURATION
MEASURE	DISPLAY, SAVE, SATCHECK, DARKRATIO
CHECK INTENSITY	ADJUSTIT*, ADJUSTND, %SATURATION*, DARKESTIMATION*, ITLIMIT*
PROCESSINFO	TYPE, DISTANCE
START LOOP	XIJ
STOP LOOP	

Table 4: Routine commands and related keywords

Those routine keywords in table 4 marked with an asterisk can also be "spectrometer specific". In that case they have an "\_X" added, where X is the spectrometer number. E.g. for the integration time the keyword can be IT, IT\_1, IT\_2 or IT\_3. More details about this are given below.

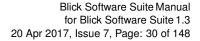




Table 5 gives an overview of all routine keywords, their possible settings, and their default settings. Every keyword except for XIJ itself can also be set to the value XIJ, if it is used in a loop (see description of command START LOOP). Settings FUNCFILT, FW1 and FW2 can be set to "Any valid filter name". The valid filter names are listed in table 6. Column 'Func' says whether the filter is considered a so-called 'Functional filter', which means it is not just use for attenuation or polarization (more in section 5.11). A detailed description of each routine command is given in the following sections.



Table 5: Routine keywords and possible settings

Routine keyword	Possible setting	<b>Default setting</b>
LENGTH	USERINPUT, float≥0	0
TIMEMODE	TOTAL, ADDED	ADDED
IT	CURRENT, CHANGE, MIN, MAX, $\min(IT) \le \text{float} \le \max(IT), x + \text{float} \ge 0, XXX$	CURRENT
NCYCLES	CURRENT, integer≥1, XXX	CURRENT
NREPETITIONS	CURRENT, integer≥0, x+integer≥0, MAX1, XXX	CURRENT
DURATION	AUTO, float≥0	AUTO
DISPLAY	NO, MEAN, MEAN+STD, CURRENT	MEAN
SAVE	NO, STDERR, STDERRTOLINE	NO
SATCHECK	NO, YES, WARN	YES
DARKRATIO	OPTIMIZED, 0≤float≤0.5	0
ADJUSTIT	FROMCURRENT, FROMMIN, XXX	FROMCURRENT
ADJUSTND	NO, FROMCURRENT, FROMMAX, FROMLASTDS, NO-MAX	NO
%SATURATION	-100≤float≤10000, SATPERC, XXX	80
DARKESTIMATION	NEGLECT, THEORY, XXX	THEORY
ITLIMIT	ITMAX, min(IT) \( \lefta \) float \( \lefta \) max(IT), XXX	ITMAX
DELTA	MIDDLE, CONT, CAMERA, float≥0	MIDDLE
AZI	RESET, POWER, CURRENT, DAZI,-360 ≤ float ≤ 360	CURRENT
ZEN	CURRENT, DZEN, PARK, MAX, -100≤float≤100	CURRENT
AZIMODE	ABS, RELSUN, RELMOON, ANGSUN, ANGMOON	ABS
ZENMODE	ABS, RELSUN, RELMOON	ABS
FUNCFILT	Any valid filter name, XXX	XXX
FW1	RESET, CURRENT, MOVE, 1≤integer≤9, any valid filter name	CURRENT
FW2	RESET, CURRENT, MOVE, 1≤integer≤9, any valid filter name	CURRENT
ТҮРЕ	ONLYL1, NOL1, SUN, MOON, SKY, TARGET, PROFILE, ALMUCANTAR, LAMP, SPECIAL	ONLYL1
DISTANCE	NO, float≥0	NO
XIJ	→ see description	0



Table	6.	Δ1	owed	filt.	er n	amee
Table	<b>()</b> .	AII	OWEG	11111	E1 112	111168

Filter type	Filter name	Func	Description
Open hole	OPEN	YES	No filter in place
Opaque blank	OPAQUE	NO	Non-transmissive blank
U340 filter	U340	YES	Colored glass filter with maximum transmission at 340 nm
BP300 filter	BP300	YES	Interference filter with maximum transmission at 300 nm
Diffuser	DIFF	YES	Grounded quartz diffuser
U340-Diffuser sandwich	U340+DIFF	YES	Combination of U340 filter and grounded quartz diffuser
BP300-Diffuser sandwich	BP300+DIFF	YES	Combination of BP300 filter and grounded quartz diffuser
Neutral density filter	NDx	NO	Reflective neutral density filter with attenuation x, $0.01 \le x \le 4.00$ with maximum 2 digits
Polarization filter	POLx	NO	Polymer linear polarizer paced in polarization angle x, $0 \le x \le 359$ , integer
Generic filter	FILTERx	NO	Generic filter name for test purposes, $1 \le x \le 9$ , integer

## 4.1.3 Routine Command DESCRIPTION

The text after "->" will be a line in the routine description, which is listed when the user enters "?" in the routine control (GUI-section 9). The routine file can have multiple lines with routine command DESCRIPTION. Each one will be a separate line in the displayed routine description. Here e.g. is a 3-line description:

```
DESCRIPTION -> Scans at fixed azimuth with open hole in filterwheel 1; DESCRIPTION -> measures at selected zenith angles for 20 secs DESCRIPTION -> at each position including dark
```

## 4.1.4 Routine Command GETSCRIPT

The text after "->" must be a special function in module src/blick\_routinereader.py. The only existing functions of this type at this time are function "SunSearchScript" and function "DiagnoseScript". The former is used for the "Find routines" FS, FA, etc. and the latter is used for the "Diagnistic routines" DC, DT, etc. E.g. this is the use of routine command GETSCRIPT in the sun search routine FS.

```
GETSCRIPT -> self.SunSearchScript(savfig=[0,-1])
```

## 4.1.5 Routine Command COMMAND

The text after "->" must be directly a command in Python. The command will be called in function "ExecuteCommands()" inside the main module /src/blick\_osmain.py. Except for simple calls, this routine command should only be used by a programmer familiar with the Python language. Below is an example for a simple command, which makes the computer beep for 500 ms at a frequency of 200 Hz.



COMMAND -> xfus.xbeep(200,500)

## 4.1.6 Routine Command DURATION

Routine command DURATION gives an estimation of the duration for a routine. It is in most cases not needed, since BlickO automatically calculates the approximate time a routine will take. This duration-estimation of BlickO can be found looking at the schedule-protocol file created when a schedule is started (see section 4.2).

Routine command DURATION should only be used either if the routine needs user input or if the user notices that the duration of the routine is significantly different from what BlickO estimates.

In the first case (user input needed) the keyword LENGTH should be set to USERINPUT and keyword TIMEMODE is not needed. This tells BlickO that at some stage the routine will stop and wait for keyboard input from the user. So the duration of the routine is "user-dependent". Routines with LENGTH=USERINPUT are not allowed in a schedule (section 4.2).

In the second case (the time the routine takes is significantly different from what BlickO estimates) the keyword LENGTH should be set to a number in seconds. BlickO will then take this value to estimate the routine duration, if keyword TIMEMODE=TOTAL, or it will add this number to its own estimation of the duration, if keyword TIMEMODE=ADDED. So e.g. if BlickO estimates the duration of a routine to 60 s, but the user notices it usually takes 80 s, he can use routine command DURATION in one of these two ways:

```
DURATION -> LENGTH=80; TIMEMODE=TOTAL
DURATION -> LENGTH=20; TIMEMODE=ADDED
```

There should only be one routine command DURATION in a routine file. The knowledge of the routine duration is not needed if BlickO is in manual mode or in routine mode. It is only needed for schedule operation (section 4.2).

## 4.1.7 Routine Command SET FILTERWHEELS

This command positions the filterwheels. Keyword FW1 refers to filterwheel 1, FW2 to filterwheel 2. RESET means the filterwheel is reset. CURRENT means the filterwheel position is not changed. MOVE means the filterwheel position is changed to the position listed in element 0 (filterwheel 1) or 1 (filterwheel 2) of global variable self.sys\_status[5]. An integer between 1 and 9 moves the filterwheel to this position. Keywords FW1 and FW2 can also be set to filterwheel position given in the IOF. E.g. the command line below puts filterwheel 1 in open position.

```
SET FILTERWHEELS -> FW1=OPEN
```

Another option for command SET FILTERWHEELS is keyword FUNCFILT. If FUNCFILT is set to any value other than XXX, it overrules the settings for FW1 and FW2. In this case the software looks for a solution in the filterwheel settings, which best matches the desired filter. E.g. this command

```
SET FILTERWHEELS -> FUNCFILT=U340
```

looks for the U340 filter in both filterwheels. If it finds it in any filterwheel, it will set this filterwheel to the position U340 and the other one to OPEN.

All settings for keyword SET FILTERWHEELS can also be combined by a ",", if they are used for Pandora systems with more than one spectrometer. E.g. this command



SET FILTERWHEELS -> FW1=U340, OPEN; FW2=4

will set filterwheel 2 to position 4 and filterwheel 1 in such a position, that spectrometer 1 is at U340 and spectrometer 2 is at OPEN. This is of course only possible if this combination exists.

Setting FUNCFILT can and should preferably be used for situations with more than one spectrometer. E.g. the command

```
SET FILTERWHEELS -> FUNCFILT=U340, OPEN
```

will find the best choice of the filterwheels, so that spectrometer 1 measures at U340 and spectrometer 2 at OPEN. Note that in all usages of FUNCFILT the spectrometer with lower index (usually spectrometer 1) has priority. This means e.g. that if the filterwheel settings are such that it is impossible that both spectrometers measure at U340, then this command

```
SET FILTERWHEELS -> FUNCFILT=U340, U340
```

will place the filterwheels so that spectrometer 1 measures at U340 and spectrometer 2 at something else. If the setting of FUNCFILT does not include a ",", then BlickO assumes that the setting should be obtained for all connected spectrometers. This means that the following two command lines are identical:

```
SET FILTERWHEELS -> FUNCFILT=U340,U340

SET FILTERWHEELS -> FUNCFILT=U340
```

## 4.1.8 Routine Command SET POINTING

This command is used to control the tracker. If keyword AZI=RESET or POWER, then a tracker reset or power reset is initiated and no other keyword is needed. For all other cases the tracker is pointed to azimuth and zenith angle positions as set in keywords AZI and ZEN.

Keywords AZIMODE and ZENMODE indicate whether the settings of AZI and ZEN are absolute or relative numbers. If keyword ZENMODE=ABS, then the setting of ZEN is considered an absolute position in degrees. If keyword ZENMODE=RELSUN, then the setting of ZEN is considered a zenith angle offset relative to the solar zenith angle in degrees. E.g. if the (refraction corrected) solar zenith angle is 46° and ZENMODE=RELSUN and ZEN=-1.5, then the tracker will point to zenith angle 46°-1.5°=44.5°. If keyword ZENMODE=RELMOON, then the zenith angle offset is relative to the lunar zenith angle.

ABS, RELSUN, and RELMOON have the same meaning for keyword AZIMODE. In addition AZIMODE can be set to ANGSUN or ANGMOON. In this case the offset is not in azimuth, but in scattering angle, i.e. the azimuth difference is divided by the sine of the solar (or lunar) zenith angle. E.g. if the (refraction corrected) solar zenith angle is  $30^{\circ}$ , the solar azimuth  $105^{\circ}$ , AZIMODE=RELSUN, and AZI=2, then the tracker will point to azimuth  $105^{\circ}+2^{\circ}=107^{\circ}$ . Differently, if AZIMODE=ANGSUN, the tracker will point to azimuth  $105^{\circ}+2^{\circ}/\sin(30^{\circ})=109^{\circ}$ .

Keyword ZEN can also be PARK or MAX. PARK means the tracker points to the "parking position", which is to the horizon in the east (azimuth 90°, zenith angle 90°). MAX means the tracker points to the maximum zenith angle in the range of the tracker (defined in the IOF entry "Maximum tracker pointing zenith angle [deg]"; see section 5.3).

Another option for AZI and ZEN is DAZI and DZEN respectively. In this case the settings for AZI and ZEN are given by the global variables self.dazi and self.dzen respectively. These variables are



initialized with setting 0, but could be modified by the user with routine command COMMAND. E.g. the following sequence would point the tracker to zenith angle 1° higher than the solar zenith angle (i.e. pointing closer to the horizon).

```
COMMAND -> self.dzen=1
SET POINTING -> ZEN=DZEN; ZENMODE=RELSUN
```

Keyword DELTA decides when the tracking is applied and if it is a one-time movement or a continuous tracking. If the setting of DELTA is a positive number, then a one-time tracking is done pointing the tracker to the desired position using the calculated solar (or lunar) angles this many seconds from now. E.g. when ZEN=0, ZENMODE=RELSUN, and DELTA=30 then the tracking moves to the zenith angle, where the sun will be 30 s from now.

If DELTA=MIDDLE then a one-time tracking is done pointing the tracker to the desired position using the calculated solar (or lunar) angles for the time in the middle of the next spectrometer measurement set. E.g. when ZEN=0, ZENMODE=RELSUN, DELTA=MIDDLE and the duration of the next spectrometer measurements is 20 s, then the tracking moves to the zenith angle, where the sun will be 10 s from now.

DELTA=CONT means the tracking is updated continuously. The update interval is given in element 4 of global variable op.tperiods with default setting 2 s. E.g. the command line below does continuous sun tracking.

```
SET POINTING -> DELTA=CONT; AZI=0; ZEN=0; AZIMODE=RELSUN; ZENMODE=RELSUN
```

DELTA=CAMERA means the camera is in automatic mode and the tracking is solely guided by the camera. In this case all other keywords in SET POINTING have no effect. This situation is usually started calling routine CK and selecting "Automatic" in the dialog. Note that once the tracking is guided by the camera, any subsequent call to SET POINTING has no effect.

## 4.1.9 Routine Command SET SPECTROMETER

This command is used to prepare the spectrometers for measurements. As mentioned above, several keywords of this command can be set individually for each spectrometer, e.g. IT\_1, IT\_2 etc. The general rule is that if a keyword referring to all spectrometers is given, then it overrules the individual keywords. E.g. if IT is set to any value other than XXX, keywords IT\_1, IT\_2 are not used. Keyword IT (or IT\_1, IT\_2, etc.), which stands for integration time, can be set to...

- CURRENT: the integration time is not changed.
- CHANGE: the integration time is changed to the value listed in element 0 of global variable self.sys\_status[7].
- MIN: the integration time is set to the allowed minimum as given in the IOF.
- MAX: the integration time is set to the allowed maximum as given in the IOF.
- Any number between the allowed minimum and maximum integration time: the integration time is set to this value (in units of ms).
- "x" plus a number: This sets the integration time to that many times the current integration time (as long as it is not above the allowed maximum). So e.g. if the integration time is 10 ms, and keyword IT=x2.5, then the integration time will be set to 25 ms.



Keyword NREPETITIONS determines the number of repetitions used for the next spectrometer measurement. It can be CURRENT, a positive integer or zero, MAX1, or an "x" plus an integer. In the last case the number of repetitions is set to that many times the current number of repetitions. So e.g. if the current number of repetitions is 2, and keyword NREPETITIONS=x3, then the next number of repetitions will be 6. MAX1 means the number of repetitions are set to 1 at the maximum. I.e. they stay 0 if they are already 0 and become 1 otherwise.

Keyword NCYCLES determines the number of cycles used for the next spectrometer measurement. It can be CURRENT or a positive integer.

Keyword DURATION can be used to adjust the number of cycles so that the total duration of a measurement is fixed. The default setting DURATION=AUTO means the number of cycles is set by keyword NCYCLES. If DURATION is a positive number, then the number of cycles is adjusted, so that the measurement last this number of seconds. E.g. the following line means that the number of cycles used will be 20 s / 50 ms = 400 and the input NCYCLES=100 is ignored.

```
SET SPECTROMETER -> IT=50; NCYCLES=100; NREPETITIONS=1; DURATION=20
```

A variation of this example sets spectrometer 1 to measure 400 cycles and spectrometer 2 to 200 cycles.

SET SPECTROMETER -> IT\_1=50; IT\_2=100; NREPETITIONS=1; DURATION=20

#### 4.1.10 Routine Command MEASURE

This command is used to measure spectra. Keyword DISPLAY decides how the data are drawn on the figure (GUI-section 8). Setting NO means the data are not drawn, MEAN means the average of all measurement cycles is drawn, and MEAN+STD means the average plus the standard deviation over all cycles is drawn, if the number of cycles exceeds one (otherwise just the average will be drawn). CURRENT means the value in variable self.figdrawmode is used.

Keyword SAVE decides whether the measured data are saved in the level 0 data file. Setting NO means they are not saved, STDERR means the average and the standard error of the measurements is saved, and STDERRTOLINE means the average and the root mean square deviation to a fitted straight line through the measurements is saved. Typically

Keyword SATCHECK decides whether the measurements are checked for saturation. Setting NO means the data are not checked for saturation. YES means the data are checked for saturation and the routine is stopped if saturation has been detected. WARN is the same as YES, but a warning message is displayed at the end of the routine, which means no further routine is executed.

Keyword DARKRATIO decides what fraction of the total measuring time is used for dark measurements. If it is set to 0, then no dark count is measured. If it is a number between 0 and 0.5, then this fraction of the cycles is used for dark measurements. So if e.g. 2000 measurement cycles are to be taken and DARKRATIO=0.1, then first 1800 cycles are measured, then the OPAQUE filter is placed in the filterwheels and 200 cycles are measured at the same integration time. At the end the filterwheels is set back to their initial positions. DARKRATIO=OPTIMIZED means the value of entry "Optimized dark ratio" in the IOF is used (see section 6.2). The simplest call is MEASURE without setting any keyword.

**MEASURE** 

In this case the default settings are used, i.e. the average of the measured data is displayed, the data are not saved to the L0 file, the data are checked for saturation, and no dark count is measured.



#### 4.1.11 Routine Command CHECK INTENSITY

This command is used to check the strength of the light input and adjust the filterwheel settings and the integration time accordingly. Also for this command several keywords can be set individually for each spectrometer, e.g. ADJUSTIT\_1, ADJUSTIT\_2 etc. As for command SET SPECTROMETER the general rule is that if a keyword referring to all spectrometers is given, then it overrules the individual keywords. E.g. if ADJUSTIT is set to any value other than XXX, keywords ADJUSTIT\_1, ADJUSTIT\_2 are not used.

Keyword ADJUSTIT decides what integration time should be used for the intensity-check measurements. FROMCURRENT means the current integration time is used. FROMMIN means the minimum integration time as given in the IOF is used.

Keyword ADJUSTND decides if and how the neutral density filters in the filterwheels are adjusted (this keyword has no meaning if the Pandora system does not have neutral density filters). NO means the neutral densities are not adjusted. FROMCURRENT means the neutral densities are adjusted, starting with the current neutral density. FROMMAX means the neutral densities are adjusted, starting with the highest neutral density in the system. FROMLASTDS means the neutral densities are adjusted, starting with the neutral density that was used during the last direct sun measurement, where the same filterwheel positions were used. NOMAX means the filterwheels are positioned to have maximum attenuation, but are not adjusted afterwards.

Keyword %SATURATION decides to what percentage of the saturation value the intensity will be adjusted. Negative numbers mean the intensity is adjusted to this percentage of the "nominal saturation limit" (taking first the absolute value of the number, i.e. -80 means 80%). The nominal saturation limit is simply 2<sup>nbits</sup>-1, with nbits being the number of bits of the instrument's AD converter as given in the IOF (see section 5.3). Positive numbers below 100 mean the intensity is adjusted to this percentage of the "effective saturation limit". The effective saturation limit is the nominal saturation limit times the value of entry "Effective saturation limit [%]" from the IOF. Positive numbers between 1000 and 10000 mean the following: in the case the effective saturation limit exists, then the intensity is adjusted to this number modulo 100 as a percentage of the effective saturation limit. Otherwise (the effective saturation limit does not exist), then the intensity is adjusted to this number div 100 as a percentage of the nominal saturation limit. If keyword %SATURATION is set to SATPERC, then the value in global variable self.satperc is used as saturation limit.

Keyword DARKESTIMATION decides how the dark count should be accounted for when determining the best integration time. NEGLECT means the influence of the dark count is ignored. THEORY means the dark behavior, as given in entries "Dark offset [counts]" and "Dark slope [counts/s]" of the IOF, is taken into account.

Keyword ITLIMIT decides what the maximum allowed integration time can be after the intensity check. ITMAX means the maximum allowed integration time as given in the IOF is used. A number between the allowed minimum and maximum integration time means this setting is used (in units of ms).

Note that if the number of repetitions for any spectrometer is set to zero, then no intensity check is done for this spectrometer.

As an example, the line below calls for an intensity check starting at minimum integration time, the neutral density used at the last direct sun measurement, trying to adjust to 80% of the instrument's effective saturation limit, taking the dark behavior into account, and not limiting the adjusted integration time (i.e. up to the instrument's maximum allowed integration time).

CHECK INTENSITY -> ADJUSTIT=FROMMIN; ADJUSTND=FROMLASTDS; %SATURATION=80; DARKESTIMATION=THEORY; ITLIMIT=ITMAX

#### 4.1.12 Routine Command PROCESSINFO

This command is needed for the processing software BlickP. It tells BlickP how the data measured by this routine should be analyzed. Keyword TYPE defines the measurement type. The options for TYPE are shown in table 7. Column index is an integer representing a certain TYPE. This index is also a column in the L0 data,



called "Data processing type index" (see section 5.7).

Table 7: Processing type

ТҮРЕ	Index	Description	
ONLYL1	0	BlickP converts the L0 data of this routine to L1 data, but does not produce data levels higher than L1.	
NOL1	1	BlickP ignores this routine.	
SUN	2	Direct sun data. BlickP produces L1 and L2 data for this routine.	
MOON	3	Direct moon data. BlickP produces L1 and L2 data for this routine.	
SKY	4	Not further specified sky radiance data. BlickP produces L1 and L2 data for this routine.	
TARGET	5	"Target routines", where measurements from a reflecting surface are made. BlickP produces L1 and L2 data for this routine.	
PROFILE	6	Routines, which allow retrieving profile information (e.g. sky scans at different viewing zenith angles for fixed azimuth). BlickP produces L1 and L2 data for this routine.	
ALMUCANTAR	7	"Almucantar routines", where sky radiance is measured at a fixed zenith angle for different azimuths. BlickP produces L1 and L2 data for this routine.	
LAMP	8	Lamp measurements or more generally for any measurement, where the input does not originate from the sun. BlickP produces L1 and L2 data, but does not retrieve a wavelength shift even if requested.	
SPECIAL	9	Not further specified "specialized" routine. BlickP produces L1 and L2 data for this routine.	

Keyword DISTANCE is only used if keyword TYPE=TARGET. Otherwise it is set to NO. The setting of DISTANCE gives the distance in m of the reflecting object, of which the measurements are made. While command PROCESSINFO with keyword TYPE should only be used once in a routine, with keyword DISTANCE it can be used as many times as needed. E.g. the lines below mean that a target routine takes measurements of an object at a distance of 0.2 m, and then of an object at 98 m from the head sensor.

```
PROCESSINFO -> TYPE=TARGET; DISTANCE=0.2 ...
PROCESSINFO -> DISTANCE=98
```

### 4.1.13 Routine Commands START LOOP and STOP LOOP

These routine commands indicate the start and stop of a loop. The number of times the loop is run is defined by keyword XIJ in command START LOOP. It can either be comma-separated list of numbers like

```
START LOOP \rightarrow XIJ=0,1,2,3,4
```

or a Python expression inside brackets after the word MAKE like



```
START LOOP -> XIJ=MAKE(range(5))
```

These two examples of command START LOOP have actually the same effect. In both cases all commands from the line with START LOOP until the line with command STOP LOOP are repeated 5 times. The variable XIJ takes the value 0 in the 1st run, 1 in the 2nd run, etc. XIJ can also be used for any keyword of the routine commands inside the loop. Here an example:

```
START LOOP -> XIJ=10,30,100

SET SPECTROMETER -> IT=XIJ; NCYCLES=10; NREPETITIONS=1

MEASURE

STOP LOOP
```

This command sequence takes spectra with 10 cycles, first at integration time 10 ms, then at 30 ms, and finally at 100 ms.

Note that MAKE is usually used when the number of loops is depending of the length of a variable. E.g.

```
START LOOP -> XIJ=MAKE(range(len(self.hh)))
```

means that as many loops are done as there are elements in global variable self.hh. Since the BlickO interpreter of the routine syntax translates the content of all routines at program startup and self.hh possibly has a different length at program execution than at startup, BlickO cannot determine how long the routine including MAKE will take. Therefore, if a routine includes MAKE and the routine is used in schedule mode, routine command DURATION must also be used in the same routine with TIMEMODE=TOTAL (see 4.1.6). Otherwise an error message will appear at program startup saying "... Routine ... includes 'MAKE', but does not include keyword 'DURATION' with subkeyword 'TOTAL'. This is not allowed in a schedule. Execution is stopped." Hence when this error message appears one can simply replace the MAKE in the START LOOP section with directly numbers (as in the first example of this section), or, if this is not possible, estimate how long this routine will take and add routine command DURATION with TIMEMODE=TOTAL.

### 4.2 Schedules

Also the concept of "schedules", i.e. a daily sequence of routines to be executed by the Pandora system, is taken from the way Brewer spectrometers are operated [15]. Schedules are stored in text files /lib/schedules/XXX.sked, where XXX is the name of the schedule, which can be of any length. Some standard schedules are automatically included during the software installation and are explained at the end of this section. To call a schedule, the "Load schedule" button (GUI-section 9) must be pressed, then a schedule selected, and finally the Start-Stop-button pressed. The first thing BlickO does is to read through the schedule commands and convert them into its own internal format.

The schedule files use a proper syntax, which is explained in this section. Errors in the syntax will in general be detected by BlickO, in which case an error message will be displayed.

Comment lines in the schedule files have to start with "#", otherwise they will be interpreted. Not counting the comment lines, a schedule file consists of several line-pairs, i.e. always an even number of lines. The second line of each line-pair, the "routine-line", contains a routine sequence just as it could also be written in the routine control (GUI-section 9). The first line of each line-pair, the "time-line", indicates the time of the routine sequence and its importance. This is an example of such a line-pair in the schedule file.



We will refer to this example in the detailed description of the syntax. The "routine-line" is "FS(SO4)", which by itself entered in the routine control would cause the execution of routine FS followed by 4 times routine SO. The first line of the line-pair, the time-line, consists of 4 or 5 space separated expressions:

- 1. Time stamp (in our example "SOLARZEN75AM-0:05.2")
- 2. Reference routine ("FS")
- 3. Reference time of reference routine ("e")
- 4. Relative importance ("2")
- 5. Repetition information ("x5")

The 1st part of the time-line, the time stamp, is an expression, which defines a specific point of time. These are the options:

- "HH:MM": This is the time in hours and minutes (UT). The minutes can have decimals. E.g. "16:40.5" means UT 16:40:30.
- "SOLARNOON": This is the time of local solar noon, i.e. the time of smallest solar zenith angle.
- "LUNARNOON": This is the time of local lunar noon, i.e. the time of smallest lunar zenith angle (which of course can also be during the night).
- "SOLARMIDNIGHT": This is the time of local solar midnight, i.e. the time of largest solar zenith angle.
- "LUNARMIDNIGHT": This is the time of local lunar midnight, i.e. the time of largest lunar zenith angle.
- "SOLARAZIxxx": This is the time corresponding to solar azimuth xxx (-180 \( \) xxx \( \) 180).
- "LUNARAZIxxx": This is the time corresponding to lunar azimuth xxx (-180 \le xxx \le 180).
- "SOLARZENxxAM": This is the time corresponding to solar zenith angle xx before local solar noon (-90 \le xx \le 90).
- "SOLARZENxxPM": This is the time corresponding to solar zenith angle xx after local solar noon (-90≤xx≤90).
- "LUNARZENxxAM": This is the time corresponding to lunar zenith angle xx before local lunar noon (-90 \le xx \le 90).
- "LUNARZENxxPM": This is the time corresponding to lunar zenith angle xx after local lunar noon (-90\le xx\le 90).
- "THEN": This is the time immediately after the end of the previously executed routine in the schedule.

Note that if a (solar or lunar) zenith angle does not occur at this day and latitude, then this entry of the schedule will be skipped.

Any of the time stamps listed above can have an optional additional "-HH:MM" or "+HH:MM". In this case the time is corrected for this value. So "SOLARZEN75AM-0:05.2" means 5 min and 12 sec before solar zenith angle 75° in the morning.

The 2nd part of the time-line is the reference routine. It must be set to the 2-character-identifier of one of the routines listed in the routine-line. The time-stamp will be applied to the first occurrence of this reference routine. In our example the time "SOLARZEN75AM-0:05.2" refers to routine "FS". If the time-stamp is "THEN", the reference routine has no importance.



The 3rd part of the time-line is the reference time of the reference routine. It can be a "b", an "m", or an "e" meaning the beginning, the middle, or the end of the reference routine. In our example "SOLARZEN75AM-0:05.2 FS e" means the routine sequence "FS(SO4)" has to be started so that routine "FS" is finished 5 min and 12 sec before solar zenith angle 75° in the morning. If the time-stamp is "THEN", the reference time of the reference routine has no importance.

The 4th part of the time-line is the relative importance. This is an integer and is only needed for the case, where two line-pairs in the schedule file "compete" for the same time. If this happens, then the line-pair of higher relative importance (=higher number) is executed, while the other one is skipped. Here an example:

```
SOLARZEN75AM FS e 2 x5
FS(SO4)
10:30 SO b 10
SOSU3
```

It is possible that for a given day and latitude the solar zenith angle of 75° in the morning is just around 10:30 UT. Then BlickO would look at the relative importance of each line-pair and based on this it will execute the "SOSU3" at 10:30 UT (relative importance is 10) and will skip the "FO(SO4)" (relative importance is 2).

The 5th part of the time-line is the repetition information (optional). If it is not given, the routine line "FS(SO4)" is executed exactly once. If it is an "x" followed by an integer number, the routine line is repeated that many times. So in our example "x5" means the routine sequence "FS(SO4)" is repeated 5 times. The repetition information can also be a time stamp of the same format as the 1st part of the time-line. In this case the routine sequence is repeated until this time is reached. E.g.

```
SOLARZEN75AM FS e 2 SOLARZEN60PM FS(SO4)
```

This means that "FS(SO4)" is started solar zenith angle 75° in the morning and repeated until the solar zenith angle is 60° in the afternoon. Note that the number of repetitions will be reduced if another routine sequence of higher relative importance is scheduled to start.

If BlickO can successfully read through the schedule file, it will first create a "protocol file" of name /lib/schedules/XYZ.prot. This is a text file with the "translated" schedule for the day and the location. E.g. a schedule called "simple" is repeating "FS(SO4)" from sunrise to sunset:

```
#simple schedule
SOLARZEN90AM FS b 2 SOLARZEN90PM
FS(SO4)
```

This is the protocol file for this schedule for 17 Dec 2014, at Goddard Space Flight Center (it is truncated in the middle):

```
Predicted Pandora #0 measurements based on schedule 'simple' for day 2014-12-17 at

Goddard Space Flight Center, USA, Lat 38.99°, Long -76.84°, 90m a.s.l.

File created on Wed 17 Dec 2014, 20:08:52 UT

Time stamp 'SOLARZEN90AM FS b 2 SOLARZEN90PM' Command 'FS(SO4)'

FS 12:25:20-12:26:10 (SolarZEN 90.04-89.90) Wed 17 Dec 2014
```



```
SO 12:26:10-12:27:01 (SolarZEN 89.90-89.76) Wed 17 Dec 2014
SO 12:27:01-12:27:52 (SolarZEN 89.76-89.62) Wed 17 Dec 2014
SO 12:27:52-12:28:43 (SolarZEN 89.62-89.48) Wed 17 Dec 2014
SO 12:28:43-12:29:34 (SolarZEN 89.48-89.34) Wed 17 Dec 2014
FS 12:29:34-12:30:24 (SolarZEN 89.34-89.20) Wed 17 Dec 2014
SO 12:30:24-12:31:15 (SolarZEN 89.20-89.06) Wed 17 Dec 2014
SO 12:31:15-12:32:06 (SolarZEN 89.06-88.92) Wed 17 Dec 2014
SO 12:32:06-12:32:57 (SolarZEN 88.92-88.78) Wed 17 Dec 2014
SO 12:32:57-12:33:48 (SolarZEN 88.78-88.64) Wed 17 Dec 2014
FS 12:33:48-12:34:38 (SolarZEN 88.64-88.50) Wed 17 Dec 2014
SO 17:48:47-17:49:38 (SolarZEN 63.26-63.29) Wed 17 Dec 2014
SO 17:49:38-17:50:29 (SolarZEN 63.29-63.33) Wed 17 Dec 2014
SO 17:50:29-17:51:20 (SolarZEN 63.33-63.36) Wed 17 Dec 2014
FS 17:51:20-17:52:10 (SolarZEN 63.36-63.40) Wed 17 Dec 2014
SO 21:34:52-21:35:43 (SolarZEN 88.84-88.98) Wed 17 Dec 2014
FS 21:35:43-21:36:33 (SolarZEN 88.98-89.12) Wed 17 Dec 2014
SO 21:36:33-21:37:24 (SolarZEN 89.12-89.26) Wed 17 Dec 2014
SO 21:37:24-21:38:15 (SolarZEN 89.26-89.40) Wed 17 Dec 2014
SO 21:38:15-21:39:06 (SolarZEN 89.40-89.54) Wed 17 Dec 2014
SO 21:39:06-21:39:57 (SolarZEN 89.54-89.68) Wed 17 Dec 2014
FS 21:39:57-21:40:47 (SolarZEN 89.68-89.82) Wed 17 Dec 2014
SO 21:40:47-21:41:38 (SolarZEN 89.82-89.96) Wed 17 Dec 2014
SO 21:41:38-21:42:29 (SolarZEN 89.96-90.11) Wed 17 Dec 2014
-> last routine probably not executed due to time restriction
Estimated number of occurrences for each routine
FS 132
SO 525
```

After the protocol file has been created, BlickO starts the schedule for the day. At the end of the day it translates the schedule file again for the next day, overwrites the protocol file with the next day's information, and starts the schedule for the next day. This continues until the user presses "Stop".

Whenever there is a break for more than 20 min in the schedule (e.g. during the night), the tracker will go in parking position, i.e. pointing to the horizon on the East (azimuth  $90^{\circ}$  and zenith angle  $90^{\circ}$ ).

The "standard schedules" listed in table 8 are automatically included during the software installation. Schedules #3 to #5 can be used as templates to produce customized schedules for the needs of the user. A typical procedure after instrument installation is to start with schedule #1 (startup) for about 30 min (during conditions with visible sun), then #2 (align) for about one day (preferably mostly with visible sun, see also section 6.1), and then a customized version of schedule templates #3 to #5. Note that the schedule templates are optimized for regular Pandoras (Pandora-1S) and Pandora-2S respectively. More details about each standard schedule can be found in the schedule-files themselves.



### Table 8: BlickO standard schedule

#	Schedule name	Description	
1	startup	Schedule used check tracking performance	
2	align	Schedule used for instrument alignment	
3	sun-1S, sun-2S	Schedule with direct sun observations	
4	sun_moon-1S, sun_moon-2S	Schedule with direct sun and moon observations	
5	sun_moon_sky-1S, sun_moon_sky-2S	Schedule with direct sun, direct moon, and sky observations	



# 5 Input Output Data Specifications

This section describes all files that are used as input or created as output by the Blick Software Suite. Table 9 gives an overview of all files. Column 'IO' says whether this file is an input (I) or output (O) or both (IO). Column 'SW' says which specific software is using this file. 'O' stands for BlickO, 'F' for BlickF and 'P' for BlickP. Column 'Type' says which format the file is written in. This can be ASCII, HDF, or binary. Column 'Sect' lists the section of the manual, where this file type is described.

Most input and output data include meta data information in the header. The meta data are described in section 5.1.

Table 9: Input Output Files Overview

				Tuble 7. Input Gutput Tiles Gverview						
Ю	SW	Type	Sect	Content						
I	О	HDF	5.2	List of locations						
I	O,P	ASCII	5.3	Hardware information for a Pandora unit						
IO	O,F	binary	3.2.3	Hexadecimal number to exchange messages between BlickO and BlickF						
IO	О	ASCII	5.4	General BlickO configuration information						
IO	О	ASCII	5.5	BlickO configuration information specific for one Pandora unit						
I	F	ASCII	5.6	BlickF configuration information						
О	0	ASCII	5.7	L0 data						
IO	O,F	ASCII	5.8	Information about the current instrument status						
I	P	ASCII	5.9	BlickP configuration information						
О	P	ASCII	5.12	L1 data						
I	P	HDF	5.10	File with configuration options for data processing						
I	P	ASCII	5.11	Calibration results for a Pandora unit						
О	P	ASCII	5.13	L2 data						
О	O,F,P	ASCII	5.14	Logger lines						
Ю	О	ASCII	5.15	Sun/Moon search results						
	I I I I I I I I I I I I I I I I I I I	I O O,F IO O,F IO O O O,F I P O P I P O P O O,F,P	I O HDF I O,P ASCII IO O,F binary IO O ASCII IO O ASCII I F ASCII O O ASCII I P ASCII I P ASCII I P HDF I P ASCII I P ASCII I P ASCII I P ASCII I D ASCII	I         O         HDF         5.2           I         O,P         ASCII         5.3           IO         O,F         binary         3.2.3           IO         O         ASCII         5.4           IO         O         ASCII         5.5           I         F         ASCII         5.6           O         O         ASCII         5.7           IO         O,F         ASCII         5.8           I         P         ASCII         5.9           O         P         ASCII         5.12           I         P         HDF         5.10           I         P         ASCII         5.11           O         P         ASCII         5.13           O         O,F,P         ASCII         5.14						



### 5.1 Meta Data

Table 10 lists all the meta data used in the Blick Software Suite. Not all meta data are used in each file type. Column 'Used by' indicates, which files include the respective meta data information. The acronyms used for this column are:

IOF Instrument operation file

ICF Instrument calibration file

L0 Level 0 file (raw signals)

L1 Level 1 file (corrected signals)

L2Fit Level 2 spectral fitting results file

L2H All other Level 2 files, i.e. L2Tot (Total columns file) and L2Trop (Tropospheric information file)

All All of the above files

Table 10: Meta Data Overview

Name	Used by	Description
File name	All	Name of file
File generation date	All	File generation date as ISO 8601 string
Data description	All	Description of file type, e.g "Instrument operation file"
Data file version	All but L0	typically a compination of characters and numbers, e.g. "c2p1s4"
Data originator name	IOF, ICF	Name of person that created the file
Local principal investigator	All but ICF	Name of responsible person at the network location
Network principal investigator	L1, L2Fit, L2	Name of the responsible person for the Pandonia network
Instrument type	All	"Pandora"
Instrument number	All	integer>0, e.g. 0 for Pandora 0
Spectrometer number	All but IOF	1 for spectrometer 1 and 2 for spectrometer 2
Operation software version used	L0	BlickO version used
Processing software version used	L1, L2Fit, L2	BlickP version used
Instrument operation file used	All but IOF, L2	Name of IOF used to run the instrument or to create the ICF
Calibration session number	ICF	integer>0 indicating which calibration session was used to produce the ICF
Calibration data validity starting date	ICF	Same as in ICF name
Instrument calibration file used	L1, L2Fit	Name of the ICF used to process data



Name	Used by	Description	
Level 0 file used	L1	Name of L0 file on which the L1 file is based on	
Level 1 file used	L2Fit	Name of L1 file on which the L2Fit file is based on	
Full location name	All but IOF, ICF	See section 5.2	
Short location name	All but IOF, ICF	See section 5.2	
Country of location	All but IOF, ICF	See section 5.2	
Location latitude [deg]	All but IOF, ICF	See section 5.2	
Location longitude [deg]	All but IOF, ICF	See section 5.2	
Location altitude [m]	All but IOF, ICF	See section 5.2	
Local noon date	L0, L1, L2Fit	Date of local noon at the selected location	
Data start time	L2	Time of first data entry in file	
Data end time	L2	Time of last data entry in file	
Notes on s-code (L1 configuration)	L1, L2Fit, L2	Lists data corrections, which are requested by the s-code, but not possible for this instrument	
Data caveats	All but IOF, L0	Caveats such as "Based on preliminary calibration"	
First pixel inside fitting window	L2Fit	Starts counting at 1 (not 0)!	
Last pixel inside fitting window	L2Fit Starts counting at 1 (not 0)!		

## 5.2 Locations File

### **5.2.1** Format

The locations file is a HDF file called <code>/lib/oslib/Blick\_locations.h5</code>. It is provided in the software distribution and can also be downloaded under <code>http://pandonia.net/docs/software/Blick\_Locations.h5</code>. The file includes one table called "Locations". Every row of the table contains the information of one instrument location (see figure 6).



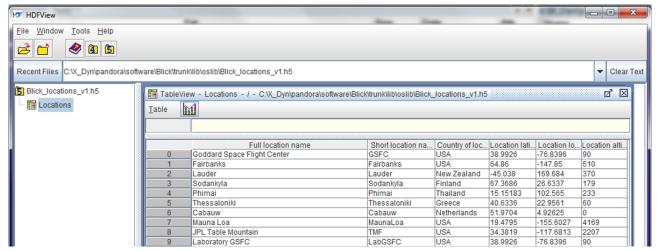


Figure 6: Screen short of location file opened with HDF viewer

For each location the following information is given:

- 1. Location number
- 2. Full location name
- 3. Short location name (used in file output, only alphanumeric!)
- 4. Country of location
- 5. Location latitude in degrees [-90°, 90°] (Minus=South of equator, Plus=North of equator)
- 6. Location longitude in degrees [-180°, 180°] (Minus=West of Greenwich, Plus=East of Greenwich)
- 7. Location altitude above sea level in meter

## 5.2.2 Adding/Updating a Locations File Entry

In principle one can edit the locations file locally. However it is strongly recommended to do it in the way as described in this section to avoid duplicated use of names or coordinates etc. An email should be written to office@luftblick.at with subject "New Pandonia Location Request". This email should include the following information:

- The full location information (all 7 entries) keeping in mind the rules listed in the next section 5.2.3.
- Name, address and contact information (email and phone number) of the Pandonia network station manager and operator (which can be the same or different persons).
- Email address to which Pandonia network correspondence should be sent.

You will receive an email as soon as the request has been resolved by LuftBlick. Then you can download the updated locations file from http://pandonia.net/docs/ and copy it in directory /lib/oslib/.

## 5.2.3 Rules for Constructing a Locations File Entry

These rules/guidelines to follow when choosing a location name:

#### **Short location name**

• No spaces are allowed.



- Maximum 20 characters.
- Use alpha-numeric characters from ISO 646/US ASCII. Example: St. Louis is used as StLouis
- Use CamelCase and if applicable and use exact writing of abbreviated names. Examples: Chesapeake-Bay, GSFC, FourCornersNM
- Use name, which is most representative, unique, and specific to the location.
- Use name of research institution of actual instrument location in abbreviated form if existing. Do not create own abbreviations. Example(s): GSFC, UMBC, UAF
- Use name of closest city, area, or most commonly used identification of the location. Examples: Fairbanks vs. UAF, JPL vs. TableMountain vs. TMF.
- In case of potential ambiguities, the station name picked first is described with priority. To resolve ambiguities, later station names must carry a state or country name. For US and Canada, the state name has preference over the usage of country name. Add country name after state name if not otherwise possible. Example: Essex (Maryland) is used first, so station name is Essex. In case Essex (Massachusetts) is used it must read EssexMA. If Essex United Kingdom is used the name reads EssexUK.
- If state or country names are used, they must follow the ISO 3166-1 alpha 2 standard.
- If more than one instrument is in the vicinity of a specific location and MORE then 100 m apart, add an integer number increment to the end of the station name starting with 2. Example: Two instruments are at GSFC, one at Building 33, one at building 21. Use GSFC (first name has preference) and GSFC2 for second instrument location.
- A moving station shall carry the location name of its platform (ship, plane, car), its base, or the greater geographic area, whatever fits best, but NOT different station names for each new acquisition setup. Examples: ChesapeakeBay, DC3.

### **Full location name**

The full location name is limited to 80 characters, can be free text, must relate to the short name, and should enhance the short name's description. The level of given details should provide uniqueness of the description. Example: GSFC -> Goddard Space Flight Center or GSFC -> Goddard Space Flight Center, Greenbelt. Do not use GSFC -> Greenbelt. The latter might be used for a separate location.

### **Country of location**

Use the name as it appears in http://www.un.org/en/members/. Examples: Use 'United States of America' and 'Republic of Korea'. Do not use 'USA' and 'South Korea'.

### Latitude, Longitude, and Altitude

Use GPS coordinates of actual image location to four digits after the comma. For altitude, use digital elevation model/map or merged barometric and GPS altitude if available.

## 5.3 Instrument Operation File

The IOF is in directory /data/operationfiles and is called PandoraX\_OF\_vVdYYYYMMDD.txt, where X is the instrument number, V the version number and YYYYMMDD the operation data validity starting date. A new version number is given, when entries in the IOF are changed, but the validity starting date stays the same. It is an ASCII-text file containing hardware information for a Pandora unit and is needed to run the instrument with BlickO as well as processing data with BlickP. Note that there is no operation data validity ending date in the file name. The validity simply ends with the starting date of the next IOF.



Each line of the IOF gives either metadata or an instrument property and its setting, separated by the string "->". Some of the instrument properties are mandatory, some are optional. E.g. if no filterwheel properties are given (they are optional), then BlickO will still start, but assumes that this specific Pandora unit does not have a head sensor. Table 11 lists all possible IOF entries and their allowed settings, which are not already listed in the meta data table 10.

Rows 1 to 8 are mandatory entries, all other ones are optional. Rows 1 to 14 refer to the spectrometer. Rows 15 to 22 refer to the head sensor. Rows 23 to 36 refer to the tracker, the shadowband, or the field of view (FOV). Rows 37 to 44 refer to the temperature controller. Row 45 refers to additional existing sensors (temperature, humidity, etc.). Rows 46 to 59 refer to the camera.

Rows 20 and 21 are actually several entries (maximum 9 per filterwheel). Valid filterwheel settings for either filterwheel are given in table 6.

The entries for the spectrometer (rows 1-14), temperature controller (rows 37-44), and FOV (rows 27, 28 and 32 to 36) can also be followed by a " 2" or a " 3" (e.g. "Spectrometer read out type 2"), which means the Pandora unit has not just one, but 2 or 3 spectrometers (or temperature controllers). Note that for all values that can be lists, a space is used as separator.

Entry "Dispersion polynomial" (row 13) lists the coefficients of the dispersion polynomial starting with the highest order. To obtain the nominal air-wavelength centers of the Pandora unit, the coefficients have to be applied on the "scaled pixels". The scaled pixels pixs are given by equation 1:

$$pixs = 3.46 \cdot \left(\frac{pix}{npix} - 0.5\right) \tag{1}$$

npix is the number of pixels (first element of entry 3 of the IOF) and pix are the regular pixel numbers starting with 1 for the first pixel (i.e. 1, 2, 3, ..., npix).



Table 11: Entries of IOF

#	Entry	Description / Allowed settings
1	Spectrometer read out type	"Ava1", "Hama1", "OcOpt1", or "JETI1"
2	Spectrometer unit ID	ID-string unique for spectrometer
3	Number of pixels	one or two elements list of integers>0, the mandatory first element gives the total number of pixels, the optional second element number gives the number of pixels read in a separate function call (usually blind pixels)
4	A/D converter number of bits	integer>0
5	Raw data discriminator factor	float>0, the detector counts are divided by this number by the manufacturer-provided data read function
6	Minimum integration time [ms]	float>0, this is at or slightly above the minimum time given by the spectrometer manufacturer
7	Maximum integration time [ms]	float>0, usually set to 4 s
8	Integration time resolution [ms]	float>0
9	Dark offset [counts]	float, average dark offset over all pixels at standard temperature
10	Dark slope [counts/s]	float, average dark slope over all pixels at standard temperature
11	Optimized dark ratio	One or two floats with 0 <float (equation="" 0.5,="" 57);="" 58),="" 60)<="" <="" a="" and="" based="" between="" bright="" case="" dark="" detector="" first="" for="" given="" is="" map="" measurements="" no="" number="" on="" optimized="" optional="" properties="" second="" split="" td="" the=""></float>
12	Limiting pixels of reference FOV	2-elements-list of integers, is >0 and <number of="" pixels<="" td=""></number>
13	Dispersion polynomial	list of floats
14	Effective saturation limit [%]	0 <float 100<="" \le="" td=""></float>
15	Head sensor-tracker connection type	"RS232"
16	Head sensor-tracker connection baudrate	integer>0
17	Head sensor-tracker ID	ID-string
18	Head sensor type	"SciGlobHSN1", "SciGlobHSGB1"
19	Head sensor port number	integer between 0 and 256; 0 means the port is not known
	Troub content port number	and BlickO scans through all ports
20	Filterwheel 1, position 1 to 9	and BlickO scans through all ports  Valid filterwheel setting
20	•	*



24         Positioning system type         "", "Novatel"           25         Tracker resolution [degrees per step]         float>0           26         Maximum tracker pointing zenith angle [deg]         float>0           27         FWHM, for reference sky FOV         3 floats>0           28         FWHM, for reference sun FOV         3 floats>0           29         Shadowband collidegrees per step]         float>0           30         Shadowband offset-to-radius ratio         0 <floate-0.9< td="">           31         FOV shape angular grid [deg]; first, last, and step         3 floats&gt;0           32         Shape of zenith scan sky FOV         list of floats           33         Shape of azimuth scan sun FOV         list of floats           34         Shape of azimuth scan sun FOV         list of floats           35         Shape of azimuth scan sun FOV         list of floats           36         Sky FOV cutoff angle         float&gt;0           37         Temperature controller connection type         "RS232"           38         Temperature controller connection baudrate         integer&gt;           40         Temperature controller port number         list of floats           41         Temperature controller proportional bandwidth [degc]         0.5≤float≤10      &lt;</floate-0.9<>	23	Tracker type	"Directed Perceptions"
26         Maximum tracker pointing zenith angle [deg]         float>0           27         FWHM, for reference sky FOV         3 floats>0           28         FWHM, for reference sun FOV         3 floats>0           29         Shadowband resolution [degrees per step]         float>0           30         Shadowband offset-to-radius ratio         0 <float<0.9< td="">           31         FOV shape angular grid [deg]; first, last, and step         3 floats&gt;0           32         Shape of zenith scan sky FOV         list of floats           33         Shape of zenith scan sun FOV         list of floats           34         Shape of azimuth scan sun FOV         list of floats           35         Shape of azimuth scan sun FOV         list of floats           36         Sky FOV cutoff angle         float&gt;0           37         Temperature controller connection type         "RS232"           38         Temperature controller connection baudrate         integer&gt;0           40         Temperature controller type         "SciGlobTC1", "TETech1", "TETech2"           41         Temperature controller port number         integer between 0 and 256; 0 means the port in ot known and BlickO scans through all ports           42         Temperature controller proportional bandwidth [degC]         0.5≤float≤100</float<0.9<>	24	Positioning system type	"", "Novatel"
27         FWHM, for reference sky FOV         3 floats>0           28         FWHM, for reference sun FOV         3 floats>0           29         Shadowband resolution [degrees per step]         float>0           30         Shadowband offset-to-radius ratio         0 <float<0.9< td="">           31         FOV shape angular grid [deg]; first, last, and step         3 floats&gt;0           32         Shape of zenith scan sky FOV         list of floats           33         Shape of zenith scan sun FOV         list of floats           34         Shape of zenith scan sun FOV         list of floats           35         Shape of zenith scan sun FOV         list of floats           36         Sky FOV cutoff angle         float&gt;0           37         Temperature controller connection type         "RS232"           38         Temperature controller connection baudrate         integer&gt;0           39         Temperature controller type         "SciGlobTC1", "TETech1", "TETech2"           40         Temperature controller port number         "integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports           42         Temperature controller set temperature [degC]         -10≤float≤10           43         Temperature controller integral gain         0≤float≤10           44</float<0.9<>	25	Tracker resolution [degrees per step]	float>0
28       FWHM, for reference sun FOV       3 floats>0         29       Shadowband resolution [degrees per step]       float>0         30       Shadowband offset-to-radius ratio       0 <float<0.9< td="">         31       FOV shape angular grid [deg]; first, last, and step       3 floats&gt;0         32       Shape of zenith scan sky FOV       list of floats         34       Shape of zenith scan sun FOV       list of floats         35       Shape of azimuth scan sun FOV       list of floats         36       Sky FOV cutoff angle       float&gt;0         37       Temperature controller connection type       "RS232"         38       Temperature controller connection baudrate       integer&gt;0         40       Temperature controller type       "SciGlobTC1", "TETech1", "TETech2"         41       Temperature controller port number       "integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports         42       Temperature controller set temperature [degC]       -10≤float≤10         43       Temperature controller proportional bandwidth [degC]       0.5≤float≤10         44       Temperature controller integral gain       0≤float≤10         45       Auxiliary sensor indices       11 (=Temperature at detector)         12 (=Temperature at detector)       12 (=Temperatur</float<0.9<>	26	Maximum tracker pointing zenith angle [deg]	float>0
29       Shadowband resolution [degrees per step]       float>0         30       Shadowband offset-to-radius ratio       0 <float<0.9< td="">         31       FOV shape angular grid [deg]; first, last, and step       3 floats&gt;0         32       Shape of zenith scan sky FOV       list of floats         33       Shape of azimuth scan sun FOV       list of floats         35       Shape of azimuth scan sun FOV       list of floats         36       Sky FOV cutoff angle       float&gt;0         37       Temperature controller connection type       "RS232"         38       Temperature controller connection baudrate       integer&gt;0         39       Temperature controller type       "SciGlobTC1", "TETech1", "TETech2"         40       Temperature controller port number       "SciGlobTC1", "TETech1", "TETech2"         41       Temperature controller port number       "snot known and BlickO scans through all ports         42       Temperature controller set temperature [degC]       -10≤float≤35         43       Temperature controller proportional bandwidth [degC]       0.5≤float≤100         44       Temperature controller integral gain       0≤float≤10         45       Auxiliary sensor indices       11 (=Temperature at detector)         12 (=Temperature at electronics board)       13 (=Spectrometer contr</float<0.9<>	27	FWHM, for reference sky FOV	3 floats>0
30       Shadowband offset-to-radius ratio       0 <float<0.9< td="">         31       FOV shape angular grid [deg]; first, last, and step       3 floats&gt;0         32       Shape of zenith scan sky FOV       list of floats         33       Shape of azimuth scan sun FOV       list of floats         34       Shape of azimuth scan sun FOV       list of floats         35       Shape of azimuth scan sun FOV       list of floats         36       Sky FOV cutoff angle       float&gt;0         37       Temperature controller connection type       "RS232"         38       Temperature controller connection baudrate       integer&gt;0         39       Temperature controller type       "SciGlobTC1", "TETech1", "TETech2"         40       Temperature controller port number       "integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports         41       Temperature controller set temperature [degC]       -10≤float≤35         43       Temperature controller proportional bandwidth [degC]       0.5≤float≤100         44       Temperature controller integral gain       0<float< td="">         45       Auxiliary sensor indices       11 (=Temperature at detector)         45       Auxiliary sensor indices       11 (=Temperature at detector)         46       Camera read module       "</float<></float<0.9<>	28	FWHM, for reference sun FOV	3 floats>0
31       FOV shape angular grid [deg]; first, last, and step       3 floats>0         32       Shape of zenith scan sky FOV       list of floats         33       Shape of azimuth scan sky FOV       list of floats         34       Shape of zenith scan sun FOV       list of floats         35       Shape of azimuth scan sun FOV       list of floats         36       Sky FOV cutoff angle       float>0         37       Temperature controller connection type       "RS232"         38       Temperature controller connection baudrate       integer>0         39       Temperature controller ID       ID-string         40       Temperature controller type       "SciGlobTC1", "TETech1", "TETech2"         41       Temperature controller port number       integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports         42       Temperature controller set temperature [degC]       -10≤float≤35         43       Temperature controller proportional bandwidth [degC]       0.5≤float≤10         44       Temperature controller integral gain       0≤float≤10         45       Auxiliary sensor indices       11 (=Temperature at detector)         46       Camera read module       "DirectX", "OpenCV"         47       Camera ID       ID-string	29	Shadowband resolution [degrees per step]	float>0
32 Shape of zenith scan sky FOV list of floats 33 Shape of azimuth scan sky FOV list of floats 34 Shape of zenith scan sun FOV list of floats 35 Shape of azimuth scan sun FOV list of floats 36 Sky FOV cutoff angle float>0 37 Temperature controller connection type "RS232" 38 Temperature controller connection baudrate integer>0 39 Temperature controller ID liD-string 40 Temperature controller type "SciGlobTC1", "TETech1", "TETech2" integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports 41 Temperature controller set temperature [degC] -10≤float≤35 43 Temperature controller proportional bandwidth [degC] 0.5≤float≤100 44 Temperature controller integral gain 0≤float≤10 45 Auxiliary sensor indices 11 (=Temperature at detector) 12 (=Temperature at detector) 12 (=Temperature at detector) 13 (=Spectrometer control temperature) 14 (=Auxiliary spectrometer temperature) 14 (=Auxiliary spectrometer temperature) 15 (=Temperature at detector) 16 (=Temperature at detector) 17 (=Temperature at detector) 18 (=Temperature at detector) 19 (=Temperature at detector) 19 (=Temperature at detector) 10 (=Temperature	30	Shadowband offset-to-radius ratio	0 <float<0.9< td=""></float<0.9<>
33       Shape of azimuth scan sky FOV       list of floats         34       Shape of zenith scan sun FOV       list of floats         35       Shape of azimuth scan sun FOV       list of floats         36       Sky FOV cutoff angle       float>0         37       Temperature controller connection type       "RS232"         38       Temperature controller connection baudrate       integer>0         39       Temperature controller ID       ID-string         40       Temperature controller type       "SciGlobTC1", "TETech1", "TETech2"         41       Temperature controller port number       integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports         42       Temperature controller set temperature [degC]       -10≤float≤35         43       Temperature controller proportional bandwidth [degC]       0.5≤float≤100         44       Temperature controller integral gain       0≤float≤10         45       Auxiliary sensor indices       11 (=Temperature at electronics board)         43       (=Spectrometer control temperature)         44       (=Auxiliary spectrometer temperature)         45       Auxiliary spectrometer temperature)         46       Camera read module       "DirectX", "OpenCV"         47       Camera itsolution [pixels]	31	FOV shape angular grid [deg]; first, last, and step	3 floats>0
34       Shape of zenith scan sun FOV       list of floats         35       Shape of azimuth scan sun FOV       list of floats         36       Sky FOV cutoff angle       float>0         37       Temperature controller connection type       "RS232"         38       Temperature controller connection baudrate       integer>0         39       Temperature controller ID       ID-string         40       Temperature controller type       "SciGlobTC1", "TETech1", "TETech2"         41       Temperature controller port number       integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports         42       Temperature controller set temperature [degC]       -10≤float≤35         43       Temperature controller proportional bandwidth [degC]       0.5≤float≤100         44       Temperature controller integral gain       0≤float≤10         45       Auxiliary sensor indices       11 (=Temperature at electronics board)         43       (=Temperature at electronics board)       13 (=Spectrometer control temperature)         46       Camera read module       "DirectX", "OpenCV"         47       Camera ID       ID-string         48       Camera read module       "DirectX", "OpenCV"         49       Camera gain (sun and moon)       two integers>0 <td>32</td> <td>Shape of zenith scan sky FOV</td> <td>list of floats</td>	32	Shape of zenith scan sky FOV	list of floats
35     Shape of azimuth scan sun FOV     list of floats       36     Sky FOV cutoff angle     float>0       37     Temperature controller connection type     "RS232"       38     Temperature controller connection baudrate     integer>0       39     Temperature controller ID     ID-string       40     Temperature controller type     "SciGlobTC1", "TETech1", "TETech2"       41     Temperature controller port number     integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports       42     Temperature controller set temperature [degC]     -10≤float≤35       43     Temperature controller proportional bandwidth [degC]     0.5≤float≤100       44     Temperature controller integral gain     0≤float≤10       45     Auxiliary sensor indices     11 (=Temperature at electronics board)       43     (=Temperature at electronics board)     13 (=Spectrometer control temperature)       44     Temperature at module     "DirectX", "OpenCV"       47     Camera ID     ID-string       48     Camera read module     "DirectX", "OpenCV"       49     Camera again (sun and moon)     two integers>0       49     Camera again (sun and moon)     two floats       50     Camera exposure time (sun and moon)     two floats       51     Camera effective center [pixels]     <	33	Shape of azimuth scan sky FOV	list of floats
36       Sky FOV cutoff angle       float>0         37       Temperature controller connection type       "RS232"         38       Temperature controller connection baudrate       integer>0         39       Temperature controller ID       ID-string         40       Temperature controller type       "SciGlobTC1", "TETech1", "TETech2"         41       Temperature controller port number       integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports         42       Temperature controller set temperature [degC]       -10≤float≤35         43       Temperature controller proportional bandwidth [degC]       0.5≤float≤100         44       Temperature controller integral gain       0≤float≤10         45       Auxiliary sensor indices       11 (=Temperature at electronics board)         43       (=Temperature at electronics board)       13 (=Spectrometer control temperature)         44       (=Auxiliary spectrometer temperature)         46       Camera read module       "DirectX", "OpenCV"         47       Camera ID       ID-string         48       Camera resolution [pixels]       two integers>0         49       Camera again (sun and moon)       two floats         50       Camera exposure time (sun and moon)       two floats>0	34	Shape of zenith scan sun FOV	list of floats
Temperature controller connection baudrate integer>0 Temperature controller ID ID-string Temperature controller set temperature [degC] -10≤float≤10  Temperature controller proportional bandwidth [degC] 0.5≤float≤100  Temperature controller integral gain 0≤float≤10  Auxiliary sensor indices 11 (=Temperature at detector) 12 (=Temperature at electronics board) 13 (=Spectrometer control temperature) 14 (=Auxiliary spectrometer temperature) 14 (=Auxiliary spectrometer temperature) 15 Camera gain (sun and moon) two floats Camera exposure time (sun and moon) two floats>0 camera effective center [pixels] float>0 camera effective solar radius [pixels] float>0 camera effective camera effective solar radius [pixels] float>0 camera	35	Shape of azimuth scan sun FOV	list of floats
Temperature controller type "SciGlobTC1", "TETech1", "TETech2"  Temperature controller type "SciGlobTC1", "TETech1", "TETech2"  Temperature controller port number integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports  Temperature controller set temperature [degC] -10≤float≤35  Temperature controller proportional bandwidth [degC] 0.5≤float≤100  Temperature controller integral gain 0≤float≤10  Auxiliary sensor indices 11 (=Temperature at detector) 12 (=Temperature at electronics board) 13 (=Spectrometer control temperature) 14 (=Auxiliary spectrometer temperature) 14 (=Auxiliary spectrometer temperature) 15 (= Temperature at electronics board) 15 (= Temperature at electronics board) 15 (= Temperature at electronics board) 16 (= Temperature at electronics board) 17 (= Temperature at electronics board) 18 (= Temperature at electronics board) 19 (= Temperature at electronics boar	36	Sky FOV cutoff angle	float>0
Temperature controller ID  Temperature controller type  Temperature controller type  Temperature controller port number  Temperature controller port number  Temperature controller port number  Temperature controller port number  Temperature controller set temperature [degC]  Temperature controller set temperature [degC]  Temperature controller proportional bandwidth [degC]  Temperature controller integral gain  Temperature controller integral gain  Temperature controller integral gain  Temperature controller integral gain  Temperature controller proportional bandwidth [degC]  Temperature controller integral gain  Temperature controller proportional bandwidth [degC]  Temperature controller integral gain  Temperature controller proportional bandwidth [degC]  Temperature controller proportional bandwidth [degC]  Temperature at detector)  12 (=Temperature at detector)  12 (=Temperature at electronics board)  13 (=Spectrometer control temperature)  The camera read module  TorrectX'', "OpenCV''  Tocamera ID  Toperature controller port number  Topefloat≤10  Temperature controller port number  Temperature controller port number  Temperature controller port number  Temperature controller proportional bandwidth [degC]  Temper	37	Temperature controller connection type	"RS232"
40 Temperature controller type "SciGlobTC1", "TETech1", "TETech2"  41 Temperature controller port number integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports  42 Temperature controller set temperature [degC] -10≤float≤35  43 Temperature controller proportional bandwidth [degC] 0.5≤float≤100  44 Temperature controller integral gain 0≤float≤10  45 Auxiliary sensor indices 11 (=Temperature at detector) 12 (=Temperature at electronics board) 13 (=Spectrometer control temperature) 14 (=Auxiliary spectrometer temperature)  46 Camera read module "DirectX", "OpenCV"  47 Camera ID ID-string 48 Camera resolution [pixels] two integers>0  49 Camera gain (sun and moon) two floats  50 Camera exposure time (sun and moon) two floats  51 Camera tracking update interval [ms] float>0  52 Camera effective center [pixels] two floats>0  53 Camera effective solar radius [pixels] float>0	38	Temperature controller connection baudrate	integer>0
Temperature controller port number integer between 0 and 256; 0 means the port is not known and BlickO scans through all ports  Temperature controller set temperature [degC] -10≤float≤35  Temperature controller proportional bandwidth [degC] 0.5≤float≤100  Temperature controller integral gain 0≤float≤10  Auxiliary sensor indices 11 (=Temperature at detector) 12 (=Temperature at electronics board) 13 (=Spectrometer control temperature) 14 (=Auxiliary spectrometer temperature)  Camera read module "DirectX", "OpenCV"  Camera ID ID-string two integers>0  Camera gain (sun and moon) two floats  Camera exposure time (sun and moon) two floats  Camera tracking update interval [ms] float>0  Camera effective center [pixels] two floats>0  Camera effective solar radius [pixels] float>0	39	Temperature controller ID	ID-string
is not known and BlickO scans through all ports  42 Temperature controller set temperature [degC] -10≤float≤35  43 Temperature controller proportional bandwidth [degC] 0.5≤float≤100  44 Temperature controller integral gain 0≤float≤10  45 Auxiliary sensor indices 11 (=Temperature at detector) 12 (=Temperature at electronics board) 13 (=Spectrometer control temperature) 14 (=Auxiliary spectrometer temperature) 14 (=Auxiliary spectrometer temperature) 14 (=Camera ID ID-string  48 Camera resolution [pixels] two integers>0  49 Camera gain (sun and moon) two floats  50 Camera exposure time (sun and moon) two floats  51 Camera effective center [pixels] two floats>0  52 Camera effective center [pixels] float>0  53 Camera effective solar radius [pixels] float>0	40	Temperature controller type	"SciGlobTC1", "TETech1", "TETech2"
43 Temperature controller proportional bandwidth [degC] 0.5≤float≤100  44 Temperature controller integral gain 0≤float≤10  45 Auxiliary sensor indices 11 (=Temperature at detector) 12 (=Temperature at electronics board) 13 (=Spectrometer control temperature) 14 (=Auxiliary spectrometer temperature) 14 (=Auxiliary spectrometer temperature) 14 (=Auxiliary spectrometer temperature) 18 Camera ID ID-string 19 two integers>0  48 Camera resolution [pixels] two integers>0  49 Camera gain (sun and moon) two floats  50 Camera exposure time (sun and moon) two floats  51 Camera tracking update interval [ms] float>0  52 Camera effective center [pixels] two floats>0  53 Camera effective solar radius [pixels] float>0	41	Temperature controller port number	is not known and BlickO scans through all
44 Temperature controller integral gain  45 Auxiliary sensor indices  11 (=Temperature at detector)  12 (=Temperature at electronics board)  13 (=Spectrometer control temperature)  14 (=Auxiliary spectrometer temperature)  46 Camera read module  "DirectX", "OpenCV"  47 Camera ID  ID-string  48 Camera resolution [pixels]  49 Camera gain (sun and moon)  50 Camera exposure time (sun and moon)  51 Camera tracking update interval [ms]  52 Camera effective center [pixels]  53 Camera effective solar radius [pixels]  10 ≤float≤10  11 (=Temperature at detector)  12 (=Temperature at detector)  12 (=Temperature at detector)  12 (=Temperature at detector)  12 (=Temperature at detector)  13 (=Spectrometer control temperature)  14 (=Auxiliary spectrometer temperature)  14 (=Auxiliary spectrometer temperature)  15 two integers>0  16 two floats  17 Camera effective center [pixels]  18 float>0  19 Camera effective solar radius [pixels]	42	Temperature controller set temperature [degC]	-10≤float≤35
Auxiliary sensor indices  11 (=Temperature at detector) 12 (=Temperature at electronics board) 13 (=Spectrometer control temperature) 14 (=Auxiliary spectrometer temperature)  46 Camera read module  "DirectX", "OpenCV"  47 Camera ID  ID-string  48 Camera resolution [pixels]  49 Camera gain (sun and moon)  two floats  50 Camera exposure time (sun and moon)  51 Camera tracking update interval [ms]  52 Camera effective center [pixels]  53 Camera effective solar radius [pixels]  float>0  float>0	43	Temperature controller proportional bandwidth [degC]	0.5≤float≤100
12 (=Temperature at electronics board) 13 (=Spectrometer control temperature) 14 (=Auxiliary spectrometer temperature) 46 Camera read module  "DirectX", "OpenCV"  47 Camera ID  ID-string  48 Camera resolution [pixels] 49 Camera gain (sun and moon)  two floats 50 Camera exposure time (sun and moon)  51 Camera tracking update interval [ms]  52 Camera effective center [pixels]  53 Camera effective solar radius [pixels]  16 (at>0)	44	Temperature controller integral gain	0≤float≤10
13 (=Spectrometer control temperature) 14 (=Auxiliary spectrometer temperature) 46 Camera read module  "DirectX", "OpenCV"  47 Camera ID  ID-string  48 Camera resolution [pixels]  two integers>0  49 Camera gain (sun and moon)  two floats  50 Camera exposure time (sun and moon)  51 Camera tracking update interval [ms]  52 Camera effective center [pixels]  53 Camera effective solar radius [pixels]  float>0  float>0	45	Auxiliary sensor indices	11 (=Temperature at detector)
14 (=Auxiliary spectrometer temperature)  46 Camera read module "DirectX", "OpenCV"  47 Camera ID ID-string  48 Camera resolution [pixels] two integers>0  49 Camera gain (sun and moon) two floats  50 Camera exposure time (sun and moon) two floats  51 Camera tracking update interval [ms] float>0  52 Camera effective center [pixels] two floats>0  53 Camera effective solar radius [pixels] float>0			12 (=Temperature at electronics board)
46 Camera read module "DirectX", "OpenCV"  47 Camera ID ID-string  48 Camera resolution [pixels] two integers>0  49 Camera gain (sun and moon) two floats  50 Camera exposure time (sun and moon) two floats  51 Camera tracking update interval [ms] float>0  52 Camera effective center [pixels] two floats>0  53 Camera effective solar radius [pixels] float>0			13 (=Spectrometer control temperature)
47 Camera ID  48 Camera resolution [pixels] two integers>0  49 Camera gain (sun and moon) two floats  50 Camera exposure time (sun and moon) two floats  51 Camera tracking update interval [ms] float>0  52 Camera effective center [pixels] two floats>0  53 Camera effective solar radius [pixels] float>0			14 (=Auxiliary spectrometer temperature)
48 Camera resolution [pixels] two integers>0 49 Camera gain (sun and moon) two floats 50 Camera exposure time (sun and moon) two floats 51 Camera tracking update interval [ms] float>0 52 Camera effective center [pixels] two floats>0 53 Camera effective solar radius [pixels] float>0	46	Camera read module	"DirectX", "OpenCV"
49 Camera gain (sun and moon) two floats 50 Camera exposure time (sun and moon) two floats 51 Camera tracking update interval [ms] float>0 52 Camera effective center [pixels] two floats>0 53 Camera effective solar radius [pixels] float>0	47	Camera ID	ID-string
50 Camera exposure time (sun and moon) two floats  51 Camera tracking update interval [ms] float>0  52 Camera effective center [pixels] two floats>0  53 Camera effective solar radius [pixels] float>0	48	Camera resolution [pixels]	two integers>0
51 Camera tracking update interval [ms] float>0 52 Camera effective center [pixels] two floats>0 53 Camera effective solar radius [pixels] float>0	49	Camera gain (sun and moon)	two floats
52 Camera effective center [pixels] two floats>0 53 Camera effective solar radius [pixels] float>0	50	Camera exposure time (sun and moon)	two floats
53 Camera effective solar radius [pixels] float>0	51	Camera tracking update interval [ms]	float>0
	52	Camera effective center [pixels]	two floats>0
54 Camera pixels per degree float>0	53	Camera effective solar radius [pixels]	float>0
	54	Camera pixels per degree	float>0



55	Camera rms tolerances (sun and moon)	two floats>0
56	Camera optical FOV full angle [deg]	float
57	Camera rotation angle [deg]	float
58	Camera angular tolerance [deg]	float>0
59	Camera frame rate [1/s]	float>0

## 5.4 BlickO General Configuration Files

The BlickO General Configuration File is in directory /config and is called BlickO\_config.txt. It is an ASCII-text file with four lines described in table 12. While parameters 1, 2 and 4 should in general not be changed be the user, parameters 3 'Do autostart' and 5 'File push time [min]' have to be set by the user. Only if the user does not want to use the autostart feature at all, 'Do autostart' should be set to 0 (see section 3.2.3). The exact meaning of parameter 'File push time [min]' is explained in section 5.7.

The procedure to add BlickF into the startup programs is different for each operating system, also changes from one Windows version to another. We recommend to search on the web for "add startup program XXX", where XXX is the operating system you are using.

# Parameter Description

1 Last operation file name The name of the operation file used in the last execution of BlickO (see section 5.3)

2 Last status file name The name of the status file used in the last execution of BlickO (see section 5.8)

3 Do autostart 0=don't use autostart, 1=use autostart if applicable

4 Latest schedule file name The name of the schedule file used in the last execution of BlickO (see section 4.2)

5 Partial file update time [min] -1=no partial files are made by BlickO, >0 is partial file creation interval in minutes

Table 12: Entries of BlickO General Configuration File

## 5.5 BlickO Specific Configuration File

The BlickO Specific Configuration File is in directory /config and is called PandoraX\_config.txt, where X is the instrument number. It is an ASCII-text file listing some parameters of the current or previous use of BlickO. At the end of each BlickO-session, the BlickO Configuration File is updated. The parameters of the BlickO Configuration File are described in table 13. All parameters can be edited by the user or in some cases by a C\*-routine in BlickO as indicated in last column of the table. A C\*-routine is a "Change system settings routine" (see section 4.1).



Table 13: Entries of BlickO Specific Configuration File

Parameter	Description	C*
Selected location (short name)	The short name of the selected location from the locations file (see section 5.2)	CL
Instrument usage mode	0=simulation mode, 1=real mode with the instrument not in use, 2=real mode with the instrument in use (for more details see section 3.1)	
Last window coordinates [pixels]	Four integer numbers with the screen coordinates of the BlickO-window. These numbers change whenever the BlickO-window is resized.	
Beep mode	0=do not beep at error, 1= beep once at error, 2= beep continuously at error	СВ
Data output mode	Currently not used	
X-axis parameter	0=pixels, 1=wavelength	CX
Show author information	0=do not show author information, 1=show author information (this is only 1 for the very first use of BlickO)	
Custom cursor	If this entry is given, it must be the name of a cursor (i.e. a file with extension "cur") in directory /lib/oslib/. In that case the given cursor is used by BlickO instead of the standard cursors, which are a white arrow and an hourglass. The given cursor rd_cur.cur is often useful, when the computer screen has to be watched in the field.	
Camera data mode	0=do not save camera data, 1=save camera images and data, 2=save camera data only, 3=save camera images only	
Freeze alignment history	0=alignment history is not frozen, 1=alignment history is frozen	CA
Figure draw mode	0=one figure panel is drawn; 1=one figure panel for each spectrometer, panels arranged horizontally, separate axis for each panel; 2=one figure panel for each spectrometer, panels arranged vertically, separate axis for each panel; 3=one figure panel for each spectrometer, panels arranged in rectangle, separate axis for each panel 4=one figure panel for each spectrometer, panels arranged horizontally, same axis for all panels; 5=one figure panel for each spectrometer, panels arranged vertically, same axis for all panels; 6=one figure panel for each spectrometer, panels arranged in rectangle, same axis for all panels	CX
Show grid lines	0=default figure has no grid lines, 1=default figure has grid lines	CX



Parameter	Description	C*
IP authenticating server	Name of server to be used for IP address authentication, if empty then Pandonia-server is used	
Tracker parking zenith angle and azimuth	Two integer numbers with parking zenith angle and azimuth respectively	CN
Positioning system tracking update interval [ms]	This number defines the repetition rate of the tracking using the positioning system	СЕ
Maximum pointing azimuth adjustment [deg]	This number sets the maximum azimuth motion allowed due to the tracker z-axis offset, 180 means no limitation (see section 6.1)	CE

## 5.6 BlickF Configuration File

The BlickF Configuration File is in directory /config and is called BlickF\_config.txt. It is an ASCII-text file setting the parameters needed for the file handling.

Each line of the BlickF Configuration File gives a configuration parameter and its setting, separated by the string "->" (table 14). Note that the local directory DIR\_LOCAL is relative to the directory of the program execution, which is /bin/ in the case the program is run as an executable and /src/ in the case the program is run by clicking on the Python sourcecode file. Therefore use ./, ../ etc. to address directories at the same or higher level, e.g. ../data/L0/ to use the raw data directory. The remote directories are relative to the remote directory after the login to the remote server. The files LOG\_FILE and FILES\_COPIED are assumed to be in directory /log/fslog/. All entries are strings unless the description says otherwise.

All parameters starting with 'AUX' are optional (i.e. if they do not exist, BlickF will still work).

Name	Description	Default value
PUSH_METHOD	File push method; -1=no pushing, 0=Python sftp for all operating systems, 1=Python sftp for Windows, otherwise scp, 2=putty for Windows, otherwise Python sftp, 3=putty for Windows, otherwise scp	3
SERVER_REMOTE	Fully qualified host and domain name	lb1.pandonia.net
PORT_REMOTE	Remote port for file push connection; leave this blank for default value	
USER_REMOTE	Remote user name; if left empty a default user will be used	
PASSWD_REMOTE	Remote password for user USER_REMOTE; if left empty a default password will be used	
DIR_REMOTE	Remote directory, where the data will be uploaded, if left empty no data upload will be done	./monitoring/

Table 14: Entries of BlickF Configuration File



Name	Description	Default value
AUX_PUSH_METHOD	Like PUSH_METHOD, but for auxiliary server	-1
AUX_SERVER_REMOTE	Like SERVER_METHOD, but for auxiliary server	lb1.pandonia.net
AUX_PORT_REMOTE	Like PORT_METHOD, but for auxiliary server	
AUX_USER_REMOTE	Remote user name for the auxiliary file push	
AUX_PASSWD_REMOTE	Remote password for user AUX_USER_REMOTE	
AUX_DIR_REMOTE	Like DIR_REMOTE, but for auxiliary server	./monitoring/
DIR_LOCAL	Local directory, from which data will be copied to remote location	/data/L0/
PUSH_TIMES	Time interval during which file push is allowed, leave empty for no restrictions. Format e.g. "07:00-08:00, 10:00-10:10", times in UTC	
LOG_FILE	BlickF actions log file name; lists actions done and errors occurred	BlickF_log.txt
FILES_COPIED	Lists all files already copied; if empty or removed BlickF will sync all available files	BlickF_alreadycopied.txt
MAX_FILES_COPY	Number of files which should be copied in one session; if left empty, all files will be copied, integer>0	10
ZIP	Zip files before copy (0=no, 1=yes); if set to 0 all new files will be transmitted the way they are; if set to 1 all new files will be zipped before transmitting them (unless they are already zip-files)	0
MESSAGE_FILE	BlickO messenger file name	./log/Blick_message.bin
POLLING_RATE	BlickF monitoring rate in seconds, float>0	10
DO_RESTART	Allow instrument host computer restart (0=no, 1=yes)	0
EMAIL_ADDRESS	Email address(es) of operator(s) to be informed when there are issues (comma separated, empty means no email is sent)	ops@pandonia.net



### 5.7 L0 File

BlickO writes the L0 data first in directory /data/tmp and then pushes them in directory /data/L0. There are 'full L0 files' and 'partial L0 files', which are described int he next sections.

#### 5.7.1 Full L0 File

The full L0 file is a daily ASCII-text files with a header, which includes meta data (see section 5.1), a description of the data columns, and the L0 data. It is called PandoraXsY\_LLL\_YYYYMMDD\_L0.txt. X is the instrument number, Y the spectrometer number, LLL the Short Location Name (see section 5.2) and YYYYMMDD is the UT-date for the time of the local noon. A new line is written to the file each time a spectrometer measurement is saved or a comment is added. For a spectrometer with 2048 pixels one data line in the L0 file occupies approximately 17 kB. This file is moved in the L0-directory after local midnight, when the L0 file of the new day is started. If sun, sky, and moon measurements are done, this file can easily contain several thousands of data lines and therefore occupy up to 100 MB.

Table 15 lists the maximum possible data columns in the L0 file. The real number of columns varies from instrument to instrument. E.g. if an instrument does not have a filterwheel 2, then there is no column "Position of filterwheel #2". The meaning of each column is described in the header after the meta data. Temperature data can be followed by the spectrometer number, e.g. "Temperature at electronics board 2" refers to spectrometer 2 (see table 15).

Below the header, there are only two types of L0 lines. "Data lines", which have exactly the number of columns listed in the header, and "Comment lines", which have only 5 columns. The first 4 columns are as for data lines, and the 5th column is a comment starting with "#". These comments can be info- or errorlogs (see section 5.14), results from sun search routines, etc. An example of a L0 file header is shown in figure 7.

Table 15: Columns in L0 file

Column name	Remark
Two letter code of measurement routine (** for manual operation)	See section 4.1
UT date and time for beginning of measurement, yyyymmddThhmmssZ (ISO 8601)	
Routine count (1 for the first routine of the day, 2 for the second, etc.)	
Repetition count (1 for the first set in the routine, 2 for the second, etc.)	
Total duration of measurement set in seconds (=# if the line is a comment line)	
Latitude at the beginning of the measurement [deg], negative=South of equator, positive=North of equator, -999=no latitude retrieved	
Longitude at the beginning of the measurement [deg], negative=West of Greenwich, positive=East of Greenwich, -999=no longitude retrieved	
Altitude a.s.l. at the beginning of the measurement [m], -999=no altitude retrieved	
West-East inclination angle at the beginning of the measurement [deg], -999=angle not retrieved	
North-South inclination angle at the beginning of the measurement [deg], -999=angle not retrieved	
Rotation angle at the beginning of the measurement [deg], -999=angle not retrieved	



Column name	Remark
Integration time [ms]	
Number of cycles	
Saturation index: positive integer is the number of saturated cycles included in the data, negative integer is the number of cycles skipped due to saturation	
Position of filterwheel #1, 0=filterwheel not used, 1-9 are valid positions	
Position of filterwheel #2, 0=filterwheel not used, 1-9 are valid positions	
Pointing zenith angle in degree, absolute or relative (see next column), 999=tracker not used	
Zenith pointing mode: zenith angle is 0=absolute, 1=relative to sun, 2=relative to moon	
Pointing azimuth in degree, increases clockwise, absolute (0=north) or relative (see next column), 999=tracker not used	
Azimuth pointing mode: like zenith angle mode but also fixed scattering angles relative to sun (3) or moon (4)	
Mean over camera offsets [deg], -9=camera not in automatic mode	
Maximum of camera offsets [deg], -9=camera not in automatic mode	
Data processing type index, -9=manual operation	See table 7
Target distance [m], -1=not pointed on target	See section 4.1.12
Temperature at electronics board X [°C], 999=no temperature signal	X=1, 2 or empty
Spectrometer control temperature X [°C], 999=no temperature signal	X=1, 2 or empty
Auxiliary spectrometer temperature X [°C], 999=no temperature signal	X=1, 2 or empty
Scale factor for data (to obtain unscaled data and uncertainty divide then by this number)	
Uncertainty indicator: uncertainty is 0=not given, 1=standard deviation, 2=rms to a fitted straight line	
Mean over all cycles of raw counts for each pixel	
Uncertainty of raw counts for each pixel divided by the square root of the number of cycles	



```
File name: Pandora0s1 IBK 20170117 L0.txt
File generation date: 20170117T101522Z
Data description: Level 0 file (raw signals)
Local principal investigator: Martin Tiefengraber
Instrument type: Pandora
Instrument number: 0
Spectrometer number: 1
Operation software version used: BlickO v1.2.8
Instrument operation file used: Pandora0 OF v1d20160916.txt
Full location name: Medical University Innsbruck
Short location name: IBK
Country of location: Austria
Location latitude [deg]: 47.2643
Location longitude [deg]: 11.3852
Location altitude [m]: 616
Local noon date: 20170117
Column 1: Two letter code of measurement routine (** for manual operation)
Column 2: UT date and time for beginning of measurement, yyyymmddThhmmssZ (ISO 8601)
Column 3: Routine count (1 for the first routine of the day, 2 for the second, etc.)
Column 4: Repetition count (1 for the first set in the routine, 2 for the second, etc.)
Column 5: Total duration of measurement set in seconds (=# if the line is a comment line)
Column 6: Latitude at the beginning of the measurement [deg], negative=South of equator, positi
Column 7: Longitude at the beginning of the measurement [deg], negative=West of Greenwich, posi
Column 8: Altitude a.s.l. at the beginning of the measurement [m], -999=no altitude retrieved
Column 9: West-East inclination angle at the beginning of the measurement [deg], -999=angle not
Column 10: North-South inclination angle at the beginning of the measurement [deg], -999=angle
Column 11: Rotation angle at the beginning of the measurement [deg], -999=angle not retrieved
Column 12: Integration time [ms]
Column 13: Number of cycles
Column 14: Saturation index: positive integer is the number of saturated cycles included in the
Column 15: Position of filterwheel #1, 0=filterwheel not used, 1-9 are valid positions
Column 16: Position of filterwheel #2, 0=filterwheel not used, 1-9 are valid positions
Column 17: Pointing zenith angle in degree, absolute or relative (see next column), 999=tracker
Column 18: Zenith pointing mode: zenith angle is... 0=absolute, 1=relative to sun, 2=relative t
Column 19: Pointing azimuth in degree, increases clockwise, absolute (0=north) or relative (see
Column 20: Azimuth pointing mode: like zenith angle mode but also fixed scattering angles relat
Column 21: Mean over camera offsets [deg], -9=camera not in automatic mode
Column 22: Maximum of camera offsets [deg], -9=camera not in automatic mode
Column 23: Data processing type index, -9=manual operation
Column 24: Target distance [m], -1=not pointed on target
Column 25: Temperature at detector 1 [°C], 999=no temperature signal
Column 26: Temperature at electronics board 1 [°C], 999=no temperature signal
Column 27: Spectrometer control temperature 1 [°C], 999=no temperature signal
Column 28: Auxiliary spectrometer temperature 1 [°C], 999=no temperature signal
Column 29: Temperature at electronics board 2 [°C], 999=no temperature signal
Column 30: Spectrometer control temperature 2 [°C], 999=no temperature signal
Column 31: Auxiliary spectrometer temperature 2 [°C], 999=no temperature signal
Column 32: Scale factor for data (to obtain unscaled data and uncertainty divide then by this r
Column 33: Uncertainty indicator: uncertainty is... 0=not given, 1=standard deviation, 2=rms to
Columns 34-2085: Mean over all cycles of raw counts for each pixel
Columns 2086-4137: Uncertainty of raw counts for each pixel divided by the square root of the r
```

Figure 7: Header for L0 file from Pandora 0, spectrometer 1

### 5.7.2 Partial L0 File

If the parameter "File push time [min]" in the BlickO General Configuration File (section 5.4) is set to -1, then no partial files are created. Otherwise, i.e. if the file push time is a positive number, partial files are created and



pushed by BlickO in the L0-directory at the interval set by the file push time. The partial L0 files are called PandoraXsY\_LLL\_YYYYMMDD\_L0\_partZ.txt, where Z is an increasing number starting at 0 for the first partial L0 file of the day. Once pushed in the L0-directory, each partial L0 file has a "Status line" added at the end, which gives information about the current instrument status (see section 5.8). Not counting this status line, the combination of all partial L0 files equals the full L0 file (section 5.7.1). This means that only the first partial L0 file (Z=0) has a header. If the L0 data are pushed in the L0 directory every 10 min, then the partial files contains typically 20 to 50 measurements, i.e. occupies 340 to 850 kB. This means both the copy process and a possible transmission to a server can be done in a relatively short time. Partial files are only pushed if the instrument is in "schedule mode" (see section 3.1.5) and no routine is currently running. Therefore partial L0 files always have complete routines, i.e. the measurements of one routine cannot be in different partial files. Therefore the effective file push time is in general somewhat larger then the nominal file push time set in the BlickO General Configuration File. Note that when the instrument is in schedule mode, but not taking data (e.g. during night), partial files are still created and pushed. They do not include data lines, only the status line. However this is done at a maximum rate of 1 min, even if the file push time was below 1 min.

### 5.8 BlickO Status File / Status Line

A status line is added to each partial L0 file, once the partial file is pushed from the temporary directory in the L0 directory. Hence even if no data were measured in the time period of a partial L0 file, then the file includes at least the status line. The status line is used to give information about the current instrument status. The line is a sequence of strings in square brackets. Inside each square bracket there is a status line keyword and a corresponding value (table 16). This is an example of the beginning of a status line:

[STATUSLINE 20141218T102443Z] [MODE MANUAL] [LOCATION GSFC] [WARN 0] ...



Table 16: Entries of Status Line

Status keyword	Description		
CURRFILEDATE	Current date as used in the L0 file; 8-character string "YYYYMMDD"		
PARTIALNUMBER	Number of this partial file; integer≥0		
CURRTIME	Current time; 16-characters string ISO8601 format "YYYYMMDDTHH-MMSSZ"		
MODE	Current measurement mode; "MANUAL" for manual operation, "ROUT-SEQU(xxx)" if routine sequence "xxx" is running or "SCHEDULE(xxx)" if schedule "xxx" is running)		
LOCATION	Current short location name (see section 5.2.1); string		
SOLARPOS	Solar angles at the time when status line was sent; zenith angle and azimuth as floats separated by a semicolon		
LUNARPOS	Lunar angles at the time when status line was sent; zenith angle and azimuth as floats separated by a semicolon		
HOURS2SUNRISE	Hours to (since) sunrise (positive numbers during night, negative numbers during day), e.g. 8.5 means it will last 8 hours and 30 min to sunrise; float		
NWARN	Number of warnings so far; integer≥0		
NTRACKRESET	Number of tracker resets needed so far; integer≥0		
LASTROUT	Name of last measurement routine that was started; 2-character string with routine name, or "" if no routine was started yet, or "wait" if schedule is waiting to be resumed (e.g. during night)		
TIMELASTROUT	Time when last routine was started; 16-characters string ISO8601 format "YYYYMMDDTHHMMSSZ", or "NONE" if no routine was started yet		
NFINDSUN	Number of sun-searches so; four semicolon-separated integers $\geq 0$ with number of long sun searches, quick sun searches, camera sun searches, and moon searches respectively		
NFINDSUNOK	Number of successful sun-searches so far; four semicolon-separated integers≥0 with number of successful long sun searches, quick sun searches, camera sun searches, and moon searches respectively		
TIMELASTFINDSUN	Time when last sun-search had started; four semicolon-separated ISO8601-strings with time of last long sun search, quick sun search, camera sun search, and moon search respectively; "NONE" if no sun search has started yet		
TIMELASTFINDSUNOK	Time when last successful sun-search had finished; four semicolon-separated ISO8601-strings with time of last successful long sun search, quick sun search, camera sun search, and moon search respectively; "NONE" if no successful sun search yet		
TIMELASTDS	Times when last direct sun and direct moon measurement was finished; two semicolon-separated ISO8601-strings; "NONE" means no direct measurement has started yet		
SZALASTDS	SZA during last direct sun and direct moon measurement; two semicolon- separated floats; -1 means no direct measurement has started yet		
FILTLASTDS	Filter used by spectrometer 1 during last direct sun and direct moon measurement; two semicolon-separated strings with filter position, e.g. "6-ND4,8-U340+DIFF"; an empty string means no direct measurement has started yet		
ITLASTDS	Integration time [ms] used by spectrometer 1 during last direct sun and direct moon measurement; two semicolon-separated floats; -1 means no direct measurement has started yet		



Status keyword	Description	
FIRSTTEMP	First effective temperature of the day [°C]; float, -99 means no temperature could be read yet	
LASTTEMP	Last measured effective temperature [°C]; float, -99 means last temperature could not be read	
TIMELASTTEMP	Time of last attempt for a temperature reading; ISO8601-string	
SETTEMP	Set temperature of the temperature controller for spectrometer 1 [°C]; float	
LOSIZE	Total size of full L0 file for spectrometer 1 so far [MB]; float	
IPINT	Internal IP address; string with IP-address, or "ERR1" if the IP-address could not be read for the first time, or "ERR" if the IP-address could not be read more than once	
IPEXT	External IP address; string with IP-address, or "ERR1" if the IP-address could not be read for the first time, or "ERR" if the IP-address could not be read more than once	
LASTIP	Time of last attempt to read the IP address; ISO8601-string	
LASTPOWER	Last retrieved power status; "ACPOWER" if the system is running on AC-power), or "BATTxx" if the system is running on battery power, where "xx" is the percentage of battery power left, or "ERR1" if the power status could not be read for the first time, or "ERR" if the power status could not be read more than once	
TIMELASTPOWER	Time of last attempt to read the power status; ISO8601-string	
WINERR	Result of last checking of window error log file; string with the windows error message, or "" if there is no error log file (i.e. all is OK), or "ERR1" if the windows error log file could not be accessed for the first time, or "ERR" if the windows error log file could not be accessed more than once	
TIMEWINERR	Time of last checking of the window error log file; ISO8601-string	
PUSHTIME	File push period [min]; float, which is identical to parameter "Partial file update time [min]" of the BlickO General Configuration File (see section 5.4)	

## 5.9 BlickP Configuration File

The BlickP configuration file is an ASCII file called /config/BlickP.ini. This file specifies the parameters needed for the data processing, i.e. which instrument, what time periods, what data levels, etc. It is written as a standard python INI-file (see https://docs.python.org/2/library/configparser.html#module-ConfigParse) and includes two sections, paths and retrieval, which will be described in the following sections. A template configuration file is automatically placed in the config-directory during software installation.

## 5.9.1 Section [paths]

This section specifies, where the different files needed for the processing or produced during the processing are located. Below is the path-section for the configuration file, which comes with the software installation. If the user stays with the standard directory structure of the Blick Software Suite, he never needs to edit this section.

[paths]
10\_directory = /data/L0



```
11_directory = /data/L1
12fit_directory = /data/L2Fit
12_directory = /data/L2
operation_directory = /data/operationfiles
calibration_directory = /data/calibrationfiles
processing_setups = /lib/pslib/Blick_ProcessingSetups.h5
```

- 10\_directory is the path, where the L0 data (section 5.7) should be located.
- 11\_directory, 12fit\_directory and 12\_directory are the paths, where the processed L1, L2Fit and L2 data (sections 5.12 and 5.13) will be stored.
- operation\_directory and calibration\_directory are the paths, where the instrument operation files and instrument calibration files (sections 5.3 and 5.11) should be located.
- processing\_setups is the name of the processing setups file (section 5.10).

### 5.9.2 Section [retrieval]

The retrieval section defines, which instruments are processed, what time periods, what data levels, etc. There is a total of six settings in this section, of which three are mandatory and three are optional. The settings are explained in table 17. Examples are given in the next section 5.9.3.

Table 17: Settings of section [retrieval]

Setting	Description
instruments	Space separated list of instrument specifiers (here we assume there are N specifiers). Each instrument specifier has the form XsY/L, where XsY is the instrument number including the spectrometer number and L is the short location name (section 5.2).
rcodes	Space separated list of N (!) specifiers, i.e. the same number as 'instruments'. Each specifier itself can be a semicolon separated string with rcodes (see section 5.10.4) or the word 'NA' if no rcodes should be processed for the respective instrument specifier. By default existing L1, L2Fit and L2 data are used. If the user wants to reprocess existing data, each rcode can be followed by a 'reprocessing specifier' after a colon (see table 18). Note that if this entire setting is missing, then no rcodes are processed.
fcodes	This is the same structure as rcodes, but refers to additional fcodes (see section 5.10.3) to be processed, even if they are not needed by the rcodes.
scodes	This is the same structure as rcodes, but refers to additional scodes (see section 5.10.2) to be processed, even if they are not needed by the rcodes or fcodes.
start_date	Space separated list of N (!) dates in the format YYYY/MM/DD, which define the start of the processing for the respective instrument specifier. Missing L0 files to be processed issue a warning, but the processing continues anyway.
end_date	Space separated list of N (!) dates in the format YYYY/MM/DD, which define the end of the processing for the respective instrument specifier. Instead of a date it can also be the word 'today', which means data are processed until the current day for this instrument specifier. Note that if this entire setting is missing, the end date is set to the current day for all instrument specifiers.



TD 11 10	T 7 1 1 1	•	· 1
Table IX	Valid i	renrocessing	1dentitiers
Table 10.	vanu i	reprocessing	identificis.

Name	Description
	If no identifier is set, BlickP uses existing L1, L2Fit and L2 data and does not recalculate those. In the case of L2 data this means it looks at the data series in the existing file and only adds the data lines, which are new.
from_12fit	If this identifier is set, BlickP uses existing L1 and L2Fit data and does not recalculate them. It does however completely rewrite the L2 file. It can be used for rcodes only.
from_11	If this identifier is set, BlickP uses existing L1 data and does not recalculate them. It does however completely rewrite the L2Fit and L2 files. It can be used for rcodes and fcodes.
from_10	If this identifier is set, BlickP does not use any existing data files and rewrites all L1, L2Fit and L2 files, even if they already exist. It can be used for rcodes, fcodes and scodes.

### 5.9.3 Section [retrieval] examples

The example below processes data from Pandora 110s1 at IBK, from 9 May 2016 until today. It processes rcodes out0 and nvs0 keeping all existing data, i.e. does not rewrite any existing file.

```
[retrieval]
instruments = 110s1/IBK
rcodes = out0;nvs0
start date = 2016/05/09
```

The second example below processes data for three instruments, for 110s1 at IBK from 9 May 2016 until 9 June 2016, for 110s2 at IBK from 1 May 2016 until the current day and for 32s1 at GSFC from 15 April 2016 until the current day.

For Pandora 110s1 at IBK and Pandora 32s1 at GSFC it processes rcodes out0 and nvs0, keeping all existing data. For 110s1 at IBK it also processes scode mca0, keeping all existing data. For 110s2 at IBK it processes rcode nvs0, keeps all existing L1 data, but reprocesses and rewrites L2Fit and L2 data.

Note that in this example the entire line 'fcodes ...' could be omitted without changing the processing.

```
[retrieval]
instruments = 110s1/IBK
                             110s2/IBK
                                            32s1/GSFC
            = out0; nvs0
                            nvs0:from_l1
                                            out0; nvs0
fcodes
                  NA
                                NA
                                              NA
scodes
                 mca0
                                NA
                                              NA
                             2016/05/01
                                           2016/04/15
            = 2016/05/09
start_date
end_date
            = 2016/06/09
                                today
                                              today
```

## 5.10 Processing Setups File

The processing setups file is a HDF file called /lib/pslib/Blick\_ProcessingSetups.h5. It is provided in the software distribution and can also be downloaded under http://pandonia.net/docs/software/Blick\_ProcessingSetups.h5. The file includes seven tables called "Trace Gases", "scodes", "qs-codes", "f-codes", "qf-codes", "r-codes" and "qr-codes". The tables "q\*-codes" include data quality (DQ) parameters corresponding to the "\*-codes". Each code consists of 4 characters, e.g. '1234'. For all the code-tables the idea is to add codes over time (i.e. lines to the tables), but never remove any, so that a specific



code has always the same parameters associated with it. Since the s-codes, f-codes and r-codes also appear in the filenames, characters like '\_', '/', '.', '\' etc. are not allowed. For simplicity we recommend staying with only letters (small or capital, which is not distinguished by the software!) and numbers for all the codes. There is a certain (not mandatory) syntax we are using for the codes, which is described in the respective sections.

### 5.10.1 Table "Trace Gases"

Table "Trace Gases" lists the short name, chemical symbol, full name, standard column amount and information about a typical profile of each trace gas included in the Blick Software Suite. The "Standard amount" is just a rough estimation of the order of magnitude for each gas abundance in the atmosphere and is used to calculate typical optical depths (OD, see section 6.4.3). The "standard unit" for a gas is molc/cm<sup>2</sup> unless the standard amount of the gas is also given in parenthesis in another unit in table 19, in which case the unit in parenthesis is the standard unit. I.e. the standard unit is e.g. DU for ozone, cm for water vapor, and molc/cm<sup>2</sup> for glyoxal.

From the profile information the effective height  $h_{EFFj}$  and effective temperature  $T_{EFFj}$  for extinction process j (or trace gas j) are calculated (also listed in table 19). If a new trace gas is added to the Blick Software Suite, then this table has to be extended in order to use it.

The effective height h<sub>EFFi</sub> is given by:

$$h_{EFFj} = \frac{\int\limits_{SURF}^{ToA} n_{j}(h) \cdot h \cdot dh}{\int\limits_{SURF}^{ToA} n_{j}(h) \cdot dh}$$

$$(2)$$

The integral runs from the surface SURF to the top of the atmosphere ToA along the vertical path h.  $n_j(h)$  is the particle density of the extinction process j at height h. The effective temperature is given by:

$$T_{EFFj} = \frac{\int\limits_{SURF}^{ToA} n_{j}(h) \cdot T_{AIR}(h) \cdot dh}{\int\limits_{SURF}^{ToA} n_{j}(h) \cdot dh}$$

$$(3)$$

 $T_{AIR}(h)$  is the air temperature at height h.



Table 19: Table "Trace Gases" in the fitting setups file

Short name	Symbol	Full name Standard amount [molc/cm <sup>2</sup> ]		h <sub>EFF</sub> [km]	T <sub>EFF</sub> [K]
О3	O <sub>3</sub>	Ozone	8.0603e18 (=300 DU)	20.4	225.0
NO2	NO <sub>2</sub>	Nitrogen dioxide	1.3434e16 (=0.5 DU)	7.2	254.5
O2O2	O <sub>2</sub> O <sub>2</sub>	Oxygen dimer	1.2691e43 molc <sup>2</sup> /cm <sup>5</sup> (=1 std atmosphere)	4.2	262.0
SO2	SO <sub>2</sub>	Sulfur dioxide	1.3434e16 (=0.5 DU)	4.0	259.2
НСНО	НСНО	Formaldehyde	1.3434e16 (=0.5 DU)	4.3	256.9
H2O	H <sub>2</sub> O	Water vapor	3.3428e22 (=1 cm precipitable water)	1.8	273.1
BrO	BrO	Bromine oxide	4e13	22.4	221.0
ClO	ClO	Chloride oxide	1e14	22.4	221.0
OClO	OCIO	Chloride dioxide	1e13	22.4	221.0
GLY	СНОСНО	Glyoxal	6e14	4.3	256.9
IO	IO	Iodine monoxide	2e12	22.4	221.0
ОН	ОН	Hydroxyl radical	5e13	22.4	221.0
O2	O <sub>2</sub>	Oxygen	4.5103e24 (=20.95 % of 1 std atmosphere)	7.3	249.9
CO2	CO <sub>2</sub>	Carbon dioxide	8.1825e21 (=380 ppm of 1 std atmosphere)	7.3	249.9
NO3	NO <sub>3</sub>	Nitrate	2e13	3.4	262.9
CH4	CH <sub>4</sub>	Methane	4.0302e19 (=1500 DU)	22.4	221.0
HONO	HONO	Nitrous acid	2.6868e15 (=0.1 DU)	4.3	256.9



### 5.10.2 Tables "s-codes" and "qs-codes"

Each line of table "s-codes" is a so-called "L1 configuration", that is used by the Blick Software Suite to produce L1 data (see sections 3.3 and 6.3). Each L1 configuration consists of a series of parameters, which make up one row of the L1 configuration table (see table 20). The first parameter of a L1 configuration is a unique 4-characters-string (the s-code), which is part of all L1 file names (see section 5.12). If any L1 configuration parameters changes, a new line is added to the table and a new s-code is given.

For the standard s-codes we use this syntax:

- First digit: this refers to the dark correction method. Letter 'i' stands for the immediate dark method with blind pixels subtracted, letter 'j' for the immediate dark method without blind pixels subtracted, letter 'm' for the dark map method with blind pixels subtracted, and letter 'n' for the dark map method without blind pixels subtracted (see sections 6.2 and 6.3.1).
- Second digit: this refers to the stray light correction method. 'n' means no stray light correction, 's' simple stray light correction, 'm' the (uncorrected) matrix method, and 'c' the corrected matrix method (see section 6.3.7).
- Third digit: this refers to application of the sensitivity correction. Letter 'r' means no sensitivity is applied, i.e. the L1 data units are counts per second. Letter 'a' means that the sensitivity is applied, i.e. the L1 data are (ir)radiances in the case the instrument has been absolutely calibrated (see section 6.3.8).
- Forth digit: This has no special meaning and often just has a number between 0 and 9.



Table 20: Columns of table "s-codes" in the Processing Setups File. The last column shows in parenthesis as an example the value of this parameters for s-code 'mca0'.

Column name	Format	Description [Value for s-code mca0]	
s-code	S4	s-code [mca0]	
s-name	S100	s-code descriptive name [All corrections applied]	
Subtract blind	S3	Are blind pixels subtracted from bright and dark counts? YES/NO [YES]	
Dark method	S5	Dark correction method, 'NO'=no dark correction applied, 'MEAS'=measured dark count is subtracted, 'MAP'=a dark estimation based on the measurements and the dark map is subtracted [MAP]	
Non-lin corr	S3	Non-linearity correction applied? YES/NO [YES]	
Latency corr	S3	Latency correction applied? YES/NO [YES]	
Flat-field corr	S3	Flat field correction? YES/NO [YES]	
Make count rates	S3	Conversion to count rates? YES/NO [YES]	
Temp corr	S3	Temperature correction applied? YES/NO [YES]	
Stray corr method	S6	Stray light correction method, 'NO'=no stray light correction, 'SIM-PLE'=simple method, 'MM'=Uncorrected matrix method, 'COR-RMM'=Corrected matrix method [CORRMM]	
Sensitivity corr	S3	Instrument sensitivity included? YES/NO [YES]	
Wavelength corr	S3	Wavelength correction? YES/NO [YES]	
qs-code	S4	qs-code used to determine the level 1 data quality (see table 21) [st00]	
Creation date	S11	Date when this s-code was added [6-Oct-2015]	
Author info	S50	Information about author of s-code [Alexander Cede]	

Entry "qs-code" indicates, which set of parameters is used to determine the data quality flag (DQF) for the L1 data. Each line of table qs-codes lists a quality setup (see table 21). The first entry of a quality setup is a unique 4-characters-string (the qs-code). If any quality entry changes, a new quality setup line is added and a new qs-code is given.

Except for the first two and last two entries of the qs-code, all entries are so-called "Level 1 data quality limits" for a specific quality indicator (e.g. the retrieved wavelength shift). If the value of the indicator exceeds the limit, the data quality is reduced. Many of the entries in the table are comma separated strings of two numbers (e.g. "WLshift limits"). The first number is the limit to change from data quality 0 (DQ0) to data quality 1 (DQ1) and the second number to change from DQ1 to data quality 2 (DQ2). This is the meaning of the three data qualities:

- DQ0 "High quality": no data quality indicator exceeds the DQ1 limit and therefore there are no indications that the data might not be of the highest possible quality. Those data can be used with high confidence.
- DQ1 "Medium quality": at least one data quality indicator exceeds the DQ1 limit and therefore the quality of the data might be reduced. Depending on the application, the user should decide whether to use these data. Note that the reduced quality can origin from instrumental sources (e.g. too large wavelength shift) or atmospheric sources (e.g. clouds increasing the uncertainty in direct sun measurements).



• DQ2 "Low quality": at least one data quality indicator exceeds the DQ2 limit and therefore the quality of the data is strongly reduced. For most purposes, the user should not use these data. As for DQ1, the low quality can origin from instrumental or atmospheric sources.

The syntax for the standard qs-codes is simply the combination of 'st' plus a number between 00 and 99. Table 21: Columns of table "qs-codes" in the processing setups file. The last column shows in parenthesis as an example the value of this parameters for qs-code 'st00'.

Column name	Format	<b>Description</b> [Value for quality setup st00]
qs-code	S4	qs-code [st00]
qs-name	S100	qs-code descriptive name [Original standard L1 correction quality limits]
Saturation allowed	S7	Are saturated pixels allowed? YES/NO; NO means the quality is reduced if saturation occurs [NO,NO]
dc cycles needed	S11	Number of measured dark cycles needed; e.g. 2 means the quality is reduced if less than 2 dark cycles were measured [2,1]
Eff temp diff limits	S11	Effective temperature difference limits in °; e.g. 3.0 means the quality is reduced if the effective temperature differs more than 3.0° from the reference temperature [3.,10.]
nsig dc difference	S11	Limits for number of sigmas in dark count difference; e.g. 5.0 means the quality is reduced if the dark count minus 5 sigmas is larger than the bright count plus 5 sigmas for at least one pixel [5.,100.]
Dark BG fitting needed	S7	Is successful dark background fitting needed? YES/NO; YES means the quality is reduced if the dark background fitting was not successful [YES,NO]
Stray light level limits	S11	Limits for estimated average residual stray light level in percent; e.g. 2.0 means the quality is reduced if the estimated residuals stray light level is above 2 %; this is only 'active' if the matrix stray light correction method was done [2,,10.]
WLchange retrieval needed	S7	Is successful wavelength change retrieval needed? YES/NO; YES means the quality is reduced if the wavelength change retrieval was not successful [YES,NO]
WLshift limits	S11	Wavelength shift limits in nm; e.g. 0.03 means the quality is reduced if the retrieved wavelength shift differs more than 0.03 nm from the nominal wavelength [0.03,0.1]
Pred WLshift limits	S11	Limits for wavelength shift difference from predicted shift in nm; e.g. 0.03 means the quality is reduced if the retrieved wavelength shift differs more than 0.03 nm from the expected shift based on the temperature [0.03,0.1]
Creation date	S11	Date when this qs-code was added [25-Jul-2016]
Author info	S50	Information about author of qs-code [Alexander Cede]



### 5.10.3 Tables "f-codes" and "qf-codes"

Each line of table "f-codes" is a so-called "fitting setup", that is used by the Blick Software Suite to produce L2Fit data (see sections 5.13.1 and 6.4). Each fitting setup consists of a series of parameters, which make up one row of the f-codes table (see table 22). The first parameter of a fitting setup is a unique 4-characters-string (the "f-code"), which is part of all L2Fit file names. If any fitting parameter changes, a new fitting setup line is added and a new f-code is given.

For the standard f-codes we use this syntax:

- First digit: this is the first letter of the name of the to primary gas in the fitting.
- Second digit: this letter refers to the approximate wavelength region of the fitting windows and can be 'u' for ultraviolet, 'v' for visible and 'n' for near infrared.
- Third digit: this letter refers to the reference used in the spectral fitting (see table 23). 't' stands for the theoretical extraterrestrial reference ("ExtRef" in table 23), 's' for the synthetic reference ("SyntXXX"), 'n' or 'f' for the spectrum of the same routine at the nearest or farest target distance ("MeasNear", "MeasFar"), 'l' or 'h' for the spectrum of the same routine at the lowest or highest pointing zenith angle ("MeasLow", "MeasHigh"), and 'e' for an external reference spectrum (e.g. from another routine, "Ref\_XXX").
- Forth digit: this has no special meaning and often just has a number between 0 and 9.



Table 22: Columns of table "f-codes" in the processing setups file. The last column shows in parenthesis as an example the value of this parameters for f-code 'out0'.

Column name	Format	Description [Value for fitting setup out0]	
f-code	S4	f-code [out0]	
f-name	S100	f-code descriptive name [Original ozone setup using extraterrestrial reference]	
Reference	S50	Reference spectrum used; for options see table 23 [ExtRef]	
WL-starts	S5	Comma separated string with starting wavelengths of the fitting window [nm]. They must be monotonically increasing. [310]	
WL-ends	S50	Comma separated string with ending wavelengths of the fitting window [nm]. They must be monotonically increasing, the same length as the starting wavelengths, and each element must be larger than the corresponding element of the starting wavelengths. The final fitting window is then composed of all the partial fitting windows determined by the wavelength starts and ends. [330]	
npol	int32	Order of fitting polynomial (see section 6.4) [4]	
noffs	int32	Order of offset polynomial (see section 6.4) [0]	
nwlc	int32	Order of wavelength change polynomial (see section 6.4) [1]	
nresc	int32	Order of resolution change polynomial (see section 6.4) [-1]	
Fitted gases	S50	Comma separated string with short names of gases to be fitted (see table 24) [O3,NO2,SO2,HCHO]	
Gas sources	S500	Comma separated string with cross section source for each gas in column "Fitted Gases" (see table 24). Since only one source per gas is listed in a given ICF, the processing can only be done, if the selected ICF matches the gas sources listed in this entry. [Daumont4TGOME, Vandaele, Vandaele, MellMoort2000]	
Gas OD meths	S50	Comma separated string with optical depth fitting method used (between 0 and 3) for each gas in column "Fitted Gases" (see section 6.4.3). In this version only one optical depth fitting method per gas is given in the ICF. Therefore the processing can only be done, if the selected ICF matches the optical depth fitting methods listed in this entry. This may change in a future software version. [1,1,1,1]	
Gas temps	S50	Comma separated string with effective temperatures for each gas in column "Fitted Gases", which should be used in the fitting. This is either a value in Kelvin or "CLIM" for climatology (option "CLIM" is not implemented yet in the current version). If the cross sections of a specific gas are only given for one temperature in the ICF, this entry has no effect for this gas. [225,254.5,259.2,256.9]	
Fitted temps	S50	Comma separated string with temperature treatment for each gas in column "Fitted Gases". NO=temperature not fitted, YES=temperature is fitted. [NO,NO,NO,NO]	



Column name	Format	Description [Value for fitting setup 1]	
Ring	S3	Is the Ring spectrum fitted (see section 6.4)? YES/NO [NO]	
Mol scatt	S3	Optical depth fitting method used for molecular scattering optical depth to be subtracted before the spectral fitting (see section 6.4)? NO means molecular scattering is not subtracted, 0, 1, 2, or 3 are the fitting methods [1]	
Linear fit	S3	Is linear fitting forced (see section 6.4)? YES/NO [NO]	
Uncertainty	S3	Is the uncertainty of the data used in the fitting (see section 6.4)? YES/NO [YES]	
s-code	S4	s-code for the L1 data that are used in the spectral fitting [mca1]	
Diffuse correction	S14	How are measurements on sun or moon corrected for the diffuser radiance entering the instrument's FOV? NO=no diffuse correction is applied and spectral fitting is done on all measurements. CALC=the diffuse fraction is calculated and subtracted from the data and spectral fitting is done on all measurements (this option is not implemented yet in the current version). OFFMEAS and OFFMEAS_WLCORR are for routines, where measurements slightly off the target (sun or moon) were made in addition to the measurements on target. OFFMEAS=subtract the 'off-target' data from the 'on-target' data without wavelength correction. Spectral fitting is only done on the corrected on-target measurement. OFFMEAS_WLCORR=the off-target spectra are wavelength corrected towards the on-target spectrum before the subtraction and spectral fitting is only done on the corrected on-target measurement. If there are more than one off-target measurement, the data will be interpolated in time to the on-target measurement before subtraction [NO]	
Time interpolation	S10	This is for routines, where spectra are measured symmetrically around a central measurement, to which the others are interpolated. NO=no time interpolation is done and spectral fitting is done on all measurements. YES=spectra with the same pointing angles are interpolated in time towards the central measurement without wavelength corrected and interpolated in time towards the central measurement. Unless NO is selected, the number of measurements in the routine must be odd, with exactly two measurements for each set of pointing angles, one before and one after the central measurement. [NO]	
Pixels to use	S6	Are all regular pixels used in the fitting or are hot/warm pixels not included? ALL=all regular pixels, NOHOT=all regular pixels except for hot pixels, NOWARM=all regular pixels except for warm and hot pixels. Note that even if NOHOT or NOWARM is selected, the fitting residuals are given for all regular pixels inside the fitting window limits with pixels not used in the fitting set to -9e99. [ALL]	
qf-code	S4	qf-code used to determine the L2Fit data quality (see table 25) [out0]	
Creation date	S11	Date when this f-code was added [6-Oct-2015]	
Author info	S50	Information about author of f-code [Alexander Cede]	



Table 23: Options for "Reference"

Reference	Description
ExtRef	"ExtRef" refers to an extraterrestrial spectrum from another source (i.e. not measured by the Pandora unit), convoluted with the Pandora filter function; BlickP uses for this purpose a high resolution extraterrestrial spectrum from 270 nm to 1000 nm merged from different sources in a similar way as described in <i>Bernhard et al.</i> [4]; the sources are: #1: <i>Kurucz</i> [16] spectrum normalized to <i>Thuillier et al.</i> [30], #2: SUSIM/Atlas 3 spectrum [33], #3: <i>Gueymard</i> [11] spectrum; the merged spectrum is #2 for $\lambda \leq 299.8$ nm; #1 corrected with smoothed #2 (triangular 0.7 nm FWHM slit) for 299.8 nm< $\lambda \leq 338.5$ nm; #1 corrected with smoothed #2 (triangular 4 nm FWHM slit) for 338.5 nm< $\lambda \leq 419.9$ nm; #1 corrected with smoothed #3 (triangular 20 nm FWHM slit) for 419.9 nm< $\lambda \leq 946.5$ nm; and #1 for $\lambda > 946.5$ nm
SyntXXX	A synthetic spectrum, which is usually the average over several spectra measured by the Pandora unit and corrected for the estimated total OD included in it; XXX is the functional filter, for which is was made (e.g. "SyntOPEN", "SyntU340")
MeasLow or MeasHigh	A single measured spectrum of the same routine at the lowest or highest pointing zenith angle
MeasNear or MeasFar	A single measured spectrum of the same routine at the nearest or farest target distance
Ref_XXX	A reference from an external file, where XXX is the filename (e.g. "C:/refspec1.txt"). Unless the extension of this file is 'ddf', it should be an ascii file with only numbers inside and 2, 3 or 4 data columns. The first column are the wavelengths in nm, the second column the reference spectrum, the optional third column the instrumental and input uncertainty of the reference spectrum, and the optional forth column the instrumental uncertainty of the reference spectrum.



Table 24: Options for "Gas Sources"

Gas	Name	Description
ОЗ	BassPaurGOME	Bass and Paur [3] for $\lambda \le 332.69$ nm combined with Burrows et al. [8] for $\lambda > 332.69$ nm
О3	GOME	Burrows et al. [8]
О3	Daumont	Malicet et al. [17]
ОЗ	IBassPaur	Bass and Paur [3] taken from http://igaco-o3.fmi.fi/ACSO/cross_sections.html
ОЗ	IDaumont5T	Malicet et al. [17] taken from http://igaco-o3.fmi.fi/ACSO/cross_sections.html
О3	IDaumont4T	IDaumont5T without using 273 K
О3	IVoigtH	Voigt et al. [34] for 1000 mbar
О3	IVoigtL	Voigt et al. [34] for 100 mbar
О3	IGOME	Burrows et al. [8] taken from http://igaco-o3.fmi.fi/ACSO/cross_sections.html
О3	ISCIAMACHY	Bogumil et al. [6] taken from http://igaco-o3.fmi.fi/ACSO/cross_sections.html
О3	Daumont4TGOME	IDaumont4T for $\lambda \le 522$ nm plus IGOME for $\lambda > 522$ nm
NO2	Vandaele	Vandaele et al. [31]
O2O2	HermansNewnham	Hermans et al. [12] for $\lambda \le 454.42$ nm combined with Newnham and Ballard [19] for $\lambda > 454.42$ nm
O2O2	Hermans	Hermans et al. [12]
O2O2	HermansOrig	Hermans et al. [12] taken from http://www.oma.be/BIRA-IASB/Scientific/Topics/Lower/LaboBase/Laboratory.html



Gas	Name	Description
SO2	Vandaele	Vandaele et al. [32]
НСНО	Meller	Meller and Moortgat [18]
H2O	HITRAN	HITRAN 2004 database Rothman et al. [23]
H2O	H2007	HITRAN 2008 database Rothman et al. [24]
BrO	Wilmouth	Wilmouth et al. [37]
BrO	Fleischmann	Fleischmann et al. [9]
ClO	Sander	Sander and Friedl [26]
OCIO	Wahner	Wahner et al. [36]
СНОСНО	Volkhamer	Volkamer et al. [35]
IO	Spietz	Spietz et al. [28]
ОН	HITRAN	HITRAN 2004 database Rothman et al. [23]
O2	HITRAN	HITRAN 2004 database Rothman et al. [23]
CO2	HITRAN	HITRAN 2004 database Rothman et al. [23]
CO2	H2007	HITRAN 2008 database Rothman et al. [24]
NO3	Sander	Sander [25]
CH4	H2007	HITRAN 2008 database Rothman et al. [24]
HONO	Bongartz	Bongartz et al. [7]
HONO	Stutz	Stutz et al. [29]

Entry "qf-code" indicates, which set of parameters is used to determine the DQ for the L2Fit data. The principle is the same as for the qs-codes. The parameters are listed in table 25. The syntax for the standard qf-codes is usually the same as for the f-code, for which it is primarily used.

Table 25: Columns of table "qf-codes" in the processing setups file. The last column shows in parenthesis as an example the value of this parameters for qf-code 'out0'.

Column name	Format	Description [Value for quality setup out0]
qf-cumber	S4	qf-code [out0]
qf-Name	S100	qf-code descriptive name [Standard ozone fitting quality limits]
Fitting result	S5	Limits for fitting result index (see table 32) [0,3]
wrms limits	S15	Normalized weighted rms limits [1e-2,2e-2]
WL shift limits	S11	Wavelength shift limits in nm; differently from the quality parameter in the qs-code, this refers to the wavelength shift as retrieved by the spectral fitting [0.2,0.5]
Creation date	S11	Date when this qf-code was added [26-Jul-2016]
Author info	S50	Information about author of qf-code [Alexander Cede]



# 5.10.4 Tables "r-codes" and "qr-codes"

Each line of table "r-codes" is a so-called "retrieval setup", that is used by the Blick Software Suite to produce L2 data such as L2Tot and L2Trop (see sections 5.13.2 and ?? to ??). Each retrieval setup consists of a series of parameters, which make up one row of the r-codes table (see table 26). The first parameter of a fitting setup is a unique 4-characters-string (the "r-code"), which is part of all L2 file names other than L2Fit. If any retrieval parameter changes, a new retrieval setup line is added and a new r-code is given. The syntax for the standard qr-codes is usually the same as for the primary f-code used in the r-code.

Table 26: Columns of table "r-codes" in the processing setups file. The last column shows in parenthesis as an example the value of this parameters for r-code 'out0'.

Column name	Format	Description [Value for retrieval setup out0]
r-code	S4	r-code [out0]
r-Name	S100	r-code descriptive name [Standard direct ozone and SO2 retrieval]
L2 type	S6	Level 2 data type produced: L2Tot or L2Trop [L2Tot]
Algorithm type	S50	Algorithm type used: 'Direct', 'Target', 'TropCol', or 'Surface' [Direct]
Process types	S50	Comma separated string of process types, for which retrieval is applied (see table 7) [SUN,MOON]
Filter types	S50	Comma separated string of functional filters, for which retrieval is applied [U340]
Output gases	S50	Comma separated string of gases to be written in output file [O3,SO2]
qr-codes	S30	Comma separated string of qr-codes used for each output gas (see 27); same length as "Output gases" [out0,sut0]
f-codes	S30	Comma separated string of f-codes used in the retrieval [out0]
Creation date	S11	Date when this r-code was added [20-May-2016]
Author info	S50	Information about author of r-code [Alexander Cede]

Entry "qr-codes" indicates, which set of parameters is used to determine the DQ for the L2 data for each output gas. The principle is the same as for the qs-codes and qf-codes. The parameters are listed in table 27. The syntax for the standard qr-codes is usually the same as for the r-code, for which it is primarily used.



Table 27: Columns of table "qr-codes" in the processing setups file. The last column shows in parenthesis as an example the value of this parameters for qr-code 'out0'.

Column name	Format	Description [Value for quality setup out0]
qr-code	S4	qr-code [out0]
qr-Name	S100	qr-code descriptive name [Standard ozone data quality limits]
Uncertainty limits	S15	Uncertainty limits in the standard unit of the gas (see table 19); negative value means this parameter is not used [3,5]
SNR limits	S15	Signal to noise limits; negative value means this parameter is not used [-1,-1]
AMF limits	S11	AMF limits; negative value means this parameter is not used [5,7]
Creation date	S11	Date when this qr-code was added [27-Jul-2016]
Author info	S50	Information about author of qr-code [Alexander Cede]



### 5.10.5 Adding new Entries to the Processing Setups File

If the user wants to add new s-, f-, q\*-, or r-codes to the tables in the Processing Setups File, e.g. for test purposes, he can just do this on his own machine and process the L2 data locally. In order to add new codes to the official Pandonia Processing Setups File, an email has to be written to office@luftblick.at with subject "New Pandonia Processing Setup Request". This email should include the following information:

- The full set of parameters needed for the new setup.
- Name and contact information (email and phone number) of the person or group that created the fitting setup.
- A short description of why the new processing setup should be used.

You will receive an email as soon as the request has been resolved by LuftBlick. Then you can download the updated processing setups file from http://pandonia.net/docs/ and copy it in directory /lib/pslib/.



# 5.11 Instrument Calibration File

directory The **ICF** in /data/calibrationfiles and is called PandoraXsY\_CF\_vVdYYYMMDD.txt, where X is the instrument number, Y is the spectrometer number, V the version number and YYYYMMDD the calibration data validity starting date. The version is used for the case that new calibration parameters have been obtained and should be applied to data from the past. In that case one can leave the same calibration data validity starting date and increase the version number, so that the processing software uses the new calibration file, when data are reprocessed. It is an ASCII-text file containing the calibration results for a Pandora unit obtained from laboratory and field calibrations. Note that as for the IOF, there is no calibration data validity ending date in the file name. The validity simply ends with the starting date of the next ICF.

Together with the IOF, this file is needed to run BlickP (see section 3.3). Each line of the ICF gives either metadata or a calibration result and its setting, separated by the string '->'. Not every calibration result listed below will necessarily be given in the ICF. A missing entry can mean that:

- This calibration result has no entries. E.g. if the instrument has no dead pixels, the calibration entry "Dead pixels" will not be present.
- This calibration result simply does not apply to the unit. E.g. if the instrument has no filterwheel 2, there will be no entry "Transmission types for filterwheel 2".
- This calibration result has not been determined.

Several calibration results are coefficients of polynomials. The coefficients are always listed in descending order (i.e. the last number is the constant term). The polynomials are usually evaluated on "scaled" values for numerical reasons. The scaling method is given in equation 4.

$$xs = 3.46 \cdot \left(\frac{x - x_{MIN}}{x_{MAX} - x_{MIN}} - 0.5\right)$$
 (4)

x is an unscaled data-point, xs is a scaled data-point and  $x_{MIN}$  and  $x_{MAX}$  are estimates of the minimum and the maximum of the distribution of the unscaled data. So the xs are distributed approximately between -1.73 and +1.73. The values for  $x_{MIN}$  and  $x_{MAX}$  used to obtain the scaled data are listed in the detailed description of each calibration result below. E.g. the scaled pixels (equation 1) use  $x_{MIN}$ =0 and  $x_{MAX}$ =npix.

Several calibration results give the rms of measured data around a fitted line. The rms is defined in equation 5.

$$rms = \sqrt{\frac{\sum\limits_{i=1}^{n}\left(x_{i} - x_{FITi}\right)^{2}}{n - n_{FIT}}}$$
 (5)

The  $x_i$  are the measured data (i=1 to n), the  $x_{FITi}$  are the fitted data and  $n_{FIT}$  is the number of fitting parameters used (e.g.  $n_{FIT}$ =3 for a fitted 2nd order polynomial).

Details for every line of the ICF, which is not already given in the meta data table 10, are listed below with the name of the calibration result in bold, larger letters. Sometimes more than one calibration result is explained in one single paragraph.

Indices of dead pixels; Indices of warm pixels; Indices of hot pixels; Indices of blind pixels; Indices of oversampled pixels



Dead pixels always return the same value, most often this is zero. Blind pixels are behind covered areas of the sensor and are therefore not exposed to incoming light. Hence they measure only dark signal (no stray light either). Oversampled pixels are read more than once. They basically only measure the dark offset (no dark slope either). All pixels that are not dead, blind, or oversampled are called regular pixels. Warm (hot) pixels have dark gain (increase of the dark count with integration time), which is 5 (10) standard deviations outside the distribution of all regular pixels. The pixel numbers start at 1 (not 0), so 1 is the 1st pixel, 2 the 2nd pixel, etc.

#### Radiometric effective temperature sensor index

This is that index from the "Auxiliary sensor indices" (see table 11), which is considered most representative for the instrument's radiometric temperature sensitivity. If this ICF-line is not given, the first index of "Auxiliary sensor indices" is used.

#### Dark variance power fit coefficients

The dark variance VD is estimated in equation 6.

$$VD = V_0 + V_1 \cdot IT^{V_2} \tag{6}$$

IT is the integration time [s] and  $V_0$ ,  $V_1$ , and  $V_2$  are coefficients, which are listed in this entry in the order  $V_0$ ,  $V_1$ , and  $V_2$ .

#### Dark background fitting settings

This are two integers determining how the dark count background should be extracted from the measured dark count (see 6.2.4). The first number if the order of the polynomial used for the fitting. The second number is the index of the pixel, at which the 'hockey stick' extension of the dark background is reduced to approximately zero.

## Dark fine structure slope, parameter A [1e-4\*counts/ms]

This is the fitting parameter A of the dark fine structure slope (called  $A_{SLi}$  in equation 65) for each regular pixel in units of 1e-4 counts/ms.

# Dark fine structure slope, parameter B [1e-4/K]

This is the fitting parameter B of the dark fine structure slope (called  $B_{SLi}$  in equation 65) for each regular pixel in units of 1e-4 per K.

# Dark fine structure intercept, parameter A [1e-4\*counts]

This is the fitting parameter A of the dark fine structure intercept (called  $A_{ICi}$  in equation 65) for each regular pixel in units of 1e-4 counts.

# Dark fine structure intercept, parameter B [1e-4/K]

This is the fitting parameter B of the dark fine structure intercept (called  $B_{ICi}$  in equation 65) for each regular pixel in units of 1e-4 per K.

# Dark map reference temperature [degC]

This is that value of the radiometric effective temperature, which is used in the dark fine structure map.

# Linearity parameters

These are the parameters for the "linearity correction function". Dividing the measured counts by this function gives the counts, which would have been obtained if the instrument had a linear response. The first three



numbers are amplitude, decay constant and power term of an exponential function. The remaining parameters are coefficients of a polynomial. The function is evaluated at the dark corrected counts divided by the nominal saturation count 2<sup>nbits</sup>-1; nbits=number of bits.

#### Radiometric reference temperature [degC]

This is that value of the radiometric effective temperature to which the radiometric temperature correction is referenced to.

#### **Temperature correction polynomial**

This polynomial is evaluated at the scaled pixels (equation 1). The obtained values are the instrument's radiometric temperature sensitivity in %/K. Equation 7 shows how to obtain the temperature corrected counts  $C_{CORR}$ .

$$C_{CORR} = C \cdot \frac{100}{100 + (T_{RE} - T_{RR}) \cdot k}$$
 (7)

C are the counts before correction,  $T_{RE}$  is the radiometric effective temperature (measured and stored in the data files),  $T_{RR}$  is the radiometric reference temperature, and k is the radiometric temperature sensitivity (i.e. the evaluated polynomial).

#### Signal shift to radiometric effective temperature [min]

This number is an estimation for the time difference between the change in the signal and the change in the radiometric effective temperature. E.g. 0.5 means the signal "reacts" to temperature changes with a delay of 30 s.

#### Gain [counts per electron]

The detector counts per one electron "accumulated" in the pixel (see also section 6.2.

#### **Integration time correction [ms]**

This is a correction in ms added to the nominal integration time to obtain the best estimation of the "true" integration time. E.g. 0.1 means that if the instrument is set to measure 4 ms, it actually integrates for 4.1 ms.

# Pixel response non uniformity [ppm]

For each pixel the difference of the pixel's response relative to the "average" response of the pixels around it. E.g. a value of 11000 means the value at this pixel is 1.1% higher than the average over the surrounding pixels.

# Slit function fitting method

The fitting method used for the slit function (see table 28). Note that the Blick Software Suite differentiates between the "slit function" (or slit scatter function) and the "filter function". The former is the signal as it appears when the detector is illuminated by a single wavelength, the latter is the function describing how the instrument filters incoming light as if it was a filter instrument. The filter function is determined by flipping the slit function around the center wavelength.

Parameters  $A_0$ , the air-wavelength of the pixel-center, and  $A_1$ , the height of the slit function at the center, are common to each fitting method and therefore not listed in table 28. Formulas for some of the slit functions are given in equations 8 to 12, where  $\lambda$  is the wavelength and  $S(\lambda)$  is the slit function. Typical fitting methods used are "Symmetric modified Lorentzian" and "Symmetric modified Gaussian".



Table 28: Slit function fitting methods

Name	Parameters
Symmetric triangle	A <sub>2</sub> : half of base width [nm]
Asymmetric triangle	A <sub>2</sub> : from center to left edge at base [nm]
	A <sub>3</sub> : from center to right edge at base [nm]
Symmetric trapezoid	A <sub>2</sub> : half of base width [nm]
	A <sub>3</sub> : half of top width [nm]
Trapezoid asymmetric on bottom	A <sub>2</sub> : half of base width left [nm]
	A <sub>3</sub> : half of base width right [nm]
	A <sub>4</sub> : half of top width [nm]
Symmetric double-trapezoid	A <sub>2</sub> : half of base width [nm]
	A <sub>3</sub> : half of top width [nm]
	A <sub>4</sub> : height of flexion point
	A <sub>5</sub> : half of middle width [nm]
Double-trapezoid both sides free	A <sub>2</sub> : half of base width left [nm]
	A <sub>3</sub> : half of base width right [nm]
	A <sub>4</sub> : half of top width [nm]
	A <sub>5</sub> : height of left flexion point
	A <sub>6</sub> : height of right flexion point
	A <sub>7</sub> : half of left middle width [nm]
	A <sub>8</sub> : half of right middle width [nm]
Double-trapezoid left side free	A <sub>2</sub> : half of base width left [nm]
	A <sub>3</sub> : half of base width right [nm]
	A <sub>4</sub> : half of top width [nm]
	A <sub>5</sub> : height of left flexion point
	A <sub>6</sub> : half of left middle width [nm]
Double-trapezoid right side free	A <sub>2</sub> : half of base width left [nm]
	A <sub>3</sub> : half of base width right [nm]
	A <sub>4</sub> : half of top width [nm]
	A <sub>5</sub> : height of right flexion point
	A <sub>6</sub> : half of right middle width [nm]
Symmetric modified Lorentzian	A <sub>2</sub> : half width [nm]
(Equation 8 with $A_6=1$ )	A <sub>3</sub> : steepness
Asymmetric modified Lorentzian	A <sub>2</sub> : half width [nm]
(Equation 9)	A <sub>3</sub> : steepness
	A <sub>4</sub> : asymmetry



Name	Parameters
Symmetric modified Gaussian	A <sub>2</sub> : half width [nm]
(Equation 10 with $A_6=1$ )	A <sub>3</sub> : steepness
Symmetric modified double Lorentzian	A <sub>2</sub> : half width of mode 1 [nm]
(Equation 8)	A <sub>3</sub> : steepness of mode 1
	A <sub>4</sub> : half width of mode 2 [nm]
	A <sub>5</sub> : steepness of mode 2
	A <sub>6</sub> : fractional strength of mode 1
Symmetric modified double Gaussian	A <sub>2</sub> : half width of mode 1 [nm]
(Equation 10)	A <sub>3</sub> : steepness of mode 1
	A <sub>4</sub> : half width of mode 2 [nm]
	A <sub>5</sub> : steepness of mode 2
	A <sub>6</sub> : fractional strength of mode 1
Symmetric modified Lorentz-Gauss combo	A <sub>2</sub> : half width of Lorentz mode [nm]
(Equation 11)	A <sub>3</sub> : steepness of Lorentz mode
	A <sub>4</sub> : half width of Gauss mode [nm]
	A <sub>5</sub> : steepness of Gauss mode
	A <sub>6</sub> : fractional strength of Lorentz mode
Truncated symmetric modified Gaussian	A <sub>2</sub> : reduced half width [nm]
(Equation 12)	A <sub>3</sub> : steepness
	A <sub>4</sub> : cut-off position

$$S(\lambda) = A_1 \cdot \left\{ A_6 \cdot \left[ 1 + \left| \frac{\lambda - A_0}{A_2} \right|^{A_3} \right]^{-1} + (1 - A_6) \cdot \left[ 1 + \left| \frac{\lambda - A_0}{A_4} \right|^{A_5} \right]^{-1} \right\}$$
(8)

$$S(\lambda) = A_1 \cdot \frac{1 + A_4 \cdot \frac{\lambda - A_0}{A_2}}{1 + \left| \frac{\lambda - A_0}{A_2} \right|^{A_3}}$$
 (9)

$$S(\lambda) = A_1 \cdot \left\{ A_6 \cdot exp \left[ -\left| \frac{\lambda - A_0}{A_2} \right|^{A_3} \right] + (1 - A_6) \cdot exp \left[ -\left| \frac{\lambda - A_0}{A_4} \right|^{A_5} \right] \right\}$$
 (10)

$$S(\lambda) = A_1 \cdot \left\{ A_6 \cdot \left[ 1 + \left| \frac{\lambda - A_0}{A_2} \right|^{A_3} \right]^{-1} + (1 - A_6) \cdot exp \left[ -\left| \frac{\lambda - A_0}{A_4} \right|^{A_5} \right] \right\}$$
 (11)

$$S(\lambda) = A_1 + A_4 \cdot \left[ exp\left( -\left| \frac{\lambda - A_0}{A_2} \right|^{A_3} \right) - 1 \right]$$
 (12)



#### **Dispersion polynomial**

This polynomial is evaluated at the scaled pixels. The evaluated data are the air-wavelengths (i.e. parameter  $A_0$  in equations 8 to 12) for the center of each pixel. This ICF-line is identical to the one with the same name in entry the IOF (see table 11).

#### Dispersion rms [nm]

This is the rms of the dispersion based on the laboratory calibration (something like a "best case rms").

#### **Resolution polynomial**

This polynomial is evaluated at the air-wavelengths for the center of each pixel in  $\mu$ m. The evaluated data are the FWHM of the slit function for each pixel.

#### **Resolution rms [nm]**

This is the rms of the resolution based on the laboratory calibration.

#### Slit function parameter $A_x$ polynomial

This polynomial is evaluated at the air-wavelengths for the center of each pixel in  $\mu$ m. The evaluated data are the slit function parameter  $A_x$  (x=2, 3, ...) for each pixel (see table 28).

#### Slit function parameter $A_x$ rms

This is the rms of slit function parameter  $A_x$  based on the laboratory calibration.

#### Wavelength effective temperature sensor index

This is that index from the "Auxiliary sensor indices" (see table 11), which is considered most representative for the instrument's wavelength temperature sensitivity. If this ICF-line is not given, the first index of "Auxiliary sensor indices" is used.

# Wavelength reference temperature [degC]

This is the value of the wavelength effective temperature, for which the wavelength parameters (dispersion, slit function) were determined.

# Dispersion temperature dependence

This is a list of values resulting from a linear fit of the wavelength temperature sensitivity k [nm/K] in the scaled pixels. If the last element of the list equals 0, then k equals the first element of the list and is constant for all pixels. If the last element of the list equals 1, then k is linearly dependent on the scaled pixels with slope and intercept being the first and second element of the list respectively. For a given k, one can estimate the "best" wavelength k of a pixel-center using equation 13.

$$\lambda = \lambda_{\text{NOM}} - k \cdot (T_{\text{WE}} - T_{\text{WR}}) \tag{13}$$

 $\lambda_{NOM}$  is the nominal center-wavelength (from the dispersion),  $T_{WE}$  is the wavelength effective temperature,  $T_{WR}$  is the wavelength reference temperature, and k is the wavelength temperature sensitivity.

#### **Resolution temperature dependence**

This is analogous to "Dispersion temperature dependence", but refers to the resolution (FWHM of slit function). Equation 13 applies when replacing  $\lambda$  with the resolution.

#### **Core slit function cut-off**

The slit function is considered to be composed of a core-part around the center and stray light in the wings.



Only the core-part of the slit function is used when convoluting spectra and cross sections. The core part ends (left and right of the center), where the slit function (normalized to 1 in the center) goes below this cut-off value.

#### Stray light correction methods for filterwheel 1 (or 2)

This entry lists the stray light correction method for each position of filterwheel 1 (or 2). In the case of a unit with two filterwheels, the final method applied is the higher number of the method for filterwheels 1 and 2. E.g. if filterwheel 1 is in position 3 and for this position there is stray light correction method 3, and filterwheel 2 is in position 4 for this position there is stray light correction method 1, then the stray light correction method applied will be method 3. The possible stray light methods are:

Method 0 "No stray light correction": data are unchanged.

Method 1 "Simple method": the average over all pixels with no light input is subtracted (see section 6.3.7). Pixels with no light input are those below 290 nm. If the unit has no pixels below 290 nm, pixels with no light input are those below 295 nm. If the unit has no pixels below 295 nm, nothing is subtracted, and then this method is like method 0. For this method 1 the exact knowledge of the point spread function for each pixel is not needed.

Methods 2 and 3 "Matrix methods": these methods are based on the exact knowledge of the instrument's slit function, typically obtained from laser measurements. The stray light corrected data are obtained applying a "stray light correction matrix" on the uncorrected data (see section 6.3.7). For Method 2 "Uncorrected matrix method": no additional correction to the stray light correction matrix is applied. For Method 3 "Corrected matrix method": the simple method 1 is applied after the stray light correction matrix is applied. This method is typically used when light contribution outside the unit's spectral range is expected.

# Dynamic nodes of slit function [nm]; Polynomial orders for dynamic nodes; Polynomials for dynamic nodes; Static nodes of slit function [nm]; Polynomial orders for static nodes; Polynomials for static nodes

These data are used to reconstruct the stray light correction matrix of the instrument.

#### Sensitivity types

This entry has as many integer numbers as there are filter-combinations in the system. For instruments with one filterwheel they are nine numbers 'belonging' to positions 1 to 9 respectively. For instruments with two filterwheels they are 81 numbers. The first number refers to filterwheel 2 in position 1 and filterwheel 1 in position 1, the second to filterwheel 2 in position 1 and filterwheel 1 in position 2, etc. The meaning of the sensitivity type is the following:

- 0 means that the sensitivity is a nominal 1 for each pixel.
- The last two digits indicate, which of the sensitivities given in entries "Sensitivity x" or "Sensitivity polynomial x" is valid for this filter-combination.
- Positive values indicate that the sensitivity is given directly for each pixel inside the valid wavelength range, while negative values indicate that the sensitivity is given as a polynomial.
- Absolute values below 100 mean that the unit of the L1 data after applying the sensitivity is counts/s, while absolute values above 100 mean that the unit of the L1 data after applying the sensitivity is W/m<sup>2</sup>/nm.



E.g. a sensitivity type of -108 means that the sensitivity for this filter-combination is listed in entry "Sensitivity polynomial 8" and the L1 units after applying the sensitivity are W/m<sup>2</sup>/nm.

# Wavelength minima for sensitivities [nm]; Wavelength maxima for sensitivities [nm]

These entries list the minimum (maximum) wavelength for which the sensitivities listed in entries "Sensitivity x" or "Sensitivity polynomial x" are given. L1 data outside these limits will be given values of "0".

#### **Scale factors for sensitivities**

This entry lists the scale factors for the data given in entries "Sensitivity x". To obtain the unscaled sensitivity, the values have to be divided by this number. The scale factors have a nominal 1 for entries "Sensitivity polynomial x".

#### Sensitivity polynomial x

These are the coefficients of the sensitivity polynomial. The polynomial is evaluated at scaled wavelengths ( $x_{MIN}$ =entry "Wavelength minima for sensitivities",  $x_{MAX}$ =entry "Wavelength maxima for sensitivities").

#### Sensitivity x

This is the scaled spectral sensitivity as an integer for each pixel inside the (nominal) wavelength limits set by entries "Wavelength minima for sensitivities" and "Wavelength minima for sensitivities".

#### **Spectral data sources**

These space separated strings list the conditions and literature sources for the convoluted values. The 1st string gives the atmospheric pressure used in the convolution of the cross sections, e.g. "p=1013.25hPa". The 2nd string gives the extraterrestrial spectrum used, e.g. "F0:XThoullier" (see table 23). The next strings give the cross section source and the reference temperature used, e.g. "O3:Daumont(225.0K)" (see table 24). The last string gives the extraterrestrial spectrum used to calculate the Ring cross sections, e.g. "RING:XThoullier" (see table 23).

#### Spectra wavelength grid; first, last, and step [nm]

This is the air-wavelength grid used for the "Reference spectrum", "Standard spectrum", and "Ring spectrum".

# Spectra scale [mW/m2/nm]

This is the scale factor for the "Reference spectrum" and "Standard spectrum". To obtain the unscaled spectra in mW/m2/nm, they have to be divided by this number.

# Reference spectrum

This is the scaled convoluted extraterrestrial spectrum using the data source, wavelength grid, and scale as given in the ICF-entries above (see also table 23).

#### Standard spectrum

This is the scaled convoluted "standard" spectrum using the data source, wavelength grid, and scale as given in the ICF entries above. Here the standard spectrum is a direct sun spectrum for solar zenith angle  $SZA=70^{\circ}$  and the standard amount of the absorbers given in table 19.

# Ring spectrum scale

This is the scale factor for the "Ring spectrum". To obtain the unscaled Ring spectrum, it has to be divided by this number. The so-called Ring effect arises in the atmosphere due to inelastic scattering processes,



mainly Rotational Raman Scattering by molecular O<sub>2</sub> and N<sub>2</sub> [10]. Roughly speaking, it manifests itself by a broadening of the solar and atmospheric spectral features present in measured spectra.

#### Ring spectrum

This is the scaled convoluted Ring spectrum using the data source, wavelength grid, and scale as given in the ICF entries above. The convoluted Ring spectrum is calculated using equation 83, just using the high resolution Ring spectrum instead of  $F_0(\lambda)$ . The possible sources for the high resolution Ring spectrum are listed in table 24 (name starts with "RING").

#### Optical depth wavelength grid; first, last, and step [nm]

This is the air-wavelength grid used for the OD-fitting parameters A, B, and C (see section 6.4.3).

#### **Temperature scale [K]**

This is the temperature scale factor (see section 6.4.3).

#### **Optical depth scale**

This is the scale factor for OD-fitting parameters A, B, and C (see section 6.4.3). To obtain the unscaled value of the parameter, it has to be divided by this number. E.g. to obtain the convoluted OD for the standard column amount of a linear absorber (B=C=0) at its reference temperature, just use OD-fitting parameter A divided by the scale factor.

#### GAS: OD-parameter Ax (or Bx, Cx), constant (or linear, quadratic) term

This is the constant (or linear, quadratic) term in temperature of the scaled convoluted OD-fitting parameter Ax (or Bx, Cx) (see section 6.4.3) for absorber with name GAS (see table 24) using the data source, wavelength grid, and scale as given in the ICF-entries above. x refers to the OD-method and can be between 0 and 3 for parameter A, 2 or 3 for parameter B and is always 3 for parameter C. Note that for wavelengths below 280 nm these entries are set to zero.

#### **Molecular scattering: OD-parameter Ax (or Bx, Cx)**

This is the scaled convoluted OD-fitting parameter Ax (or Bx, Cx) for molecular (Rayleigh and Raman) scattering using the wavelength grid and scale as given in the ICF-entries above (see section 6.4.5).

# Standard total optical depth

This is the sum over all individual optical depths for standard conditions.

# Relative reference spectrum change with resolution

This is the change of the logarithm (or relative change) of ICF-entry "Reference spectrum" for a 1% increase in the instrument's resolution, scaled by ICF-entry "Optical depth scale".

# Standard total optical depth change with resolution

This is the change of the ICF-entry "Standard total optical depth" for a 1% increase in the instrument's resolution, scaled by ICF-entry "Optical depth scale".

# Reference wavelength for wavelength change retrieval [nm]

This is the reference wavelength  $\lambda_{REF}$  used in the wavelength change polynomial. The wavelength change polynomial is retrieved by comparing the measured spectrum to ICF-entry "Standard spectrum". The wavelength change  $\Delta \lambda_i$  for pixel i is given by equation 14:



$$\Delta \lambda_{i} = \sum_{k=0}^{n_{\text{WLC}}} c_{k} \cdot (\lambda_{i\text{NOM}} - \lambda_{\text{REF}})^{k}$$
(14)

 $\lambda_{iNOM}$  is the nominal center-wavelength for pixel i (as it comes from the dispersion polynomial) and the  $c_k$  are the coefficients of the wavelength change polynomial.  $c_0$  is also called (wavelength) shift and  $c_1$  is also called (wavelength) squeeze. The best estimation of the "true" wavelengths for the measurement is given by equation 15:

$$\lambda_{\text{iBEST}} = \lambda_{\text{iNOM}} + \Delta \lambda_{\text{i}} \tag{15}$$

#### Wavelength change retrieval parameters for FUNCFILT

This entry lists five parameters needed to retrieve the wavelength change polynomial (see equation 14) for a measurement using "functional filter" FUNCFILT. A functional filter is basically any filter listed in entries 19 or 20 of the IOF (see table 11), which is not a neutral density filter (e.g. U340, BP300). Note that OPEN is also considered a functional filter. The second of the five parameters is the order of the wavelength change polynomial (nWLC in equation 14).

# Synthetic reference spectrum for FUNCFILT wavelength grid; first, last, and step [nm]

This is the air-wavelength grid used for the "Synthetic reference spectrum for FUNCFILT".

#### Scale factor of synthetic reference spectrum for FUNCFILT

This is the scale factor for the "Synthetic reference spectrum for FUNCFILT". To obtain the unscaled spectrum, it has to be divided by this number.

#### Synthetic reference spectrum for FUNCFILT

This is the scaled synthetic spectrum for functional filter FUNCFILT. The synthetic spectrum is usually a combination of several measured spectra. Depending on the calibration status of the instrument, this spectrum might be in units of corrected count rates or  $mW/m^2/nm$ .

#### 5.12 L1 File

BlickP converts the L0 data into L1 data by applying the all instrumental corrections to the raw data (see section 6.3). L1 files are daily ASCII-text files with a header, which includes meta data (see section 5.1), a description of the data columns, and the L1 data. Note that only "Data lines" from the L0 file are converted into L1 data. "Comment lines" with information, warnings, or errors are ignored (see 5.7). Also, L0 data with a "Data processing type index" of -9 (manual operation) or 1 (processing type "NOL1", see section 4.1.12) are ignored by BlickP.

L1 files are called PandoraXsY\_LLL\_YYYYMMDD\_L1\_sssssccpP.txt. X, Y, LLL, YYYYMMDD are as for the L0 file (section 5.7). The segment "sSSSscCpP" is the so-called "Data file version". SSSS is the "s-code" or "L1 configuration", that was used for the processing (see section 5.10.2). C is the version number of the ICF used for the processing. It corresponds to the number after "v" in the ICF name (see section 5.11). P is the main version number and first subversion number of BlickP used for the processing, combined with a dash instead of a dot. E.g. for BlickP version 1.2.5, P equals 1-2. Obviously if L1 data are processed with different data file versions, different L1 data files are produced, i.e. more than one L1 file can exist for a given



L0 file. L1 files occupy around three quarters of the space of the corresponding L0 file. The data lines are approximately cut in half, since the dark measurement have been included in the data correction, but the data lines are longer since the instrumental uncertainty is listed in addition to the total uncertainty.

Table 29 lists the maximum possible data columns in the L1 file. As for the L0 file, the real number of columns varies from instrument to instrument. E.g. if an instrument does not have a filterwheel 2, then there is no column "Effective position of filterwheel #2". The meaning of each column is described in the header after the meta data. Temperature data can be followed by the spectrometer number, e.g. "Temperature at electronics board 2" refers to spectrometer 2 (see table 29).

Note that the filterwheel positions are now called 'effective' positions. In the L0 data (table 15) they are the 'nominal' filterwheel positions as set during the measurements. The 'effective' filterwheel position says through which position the light for the specific spectrometer was going. For spectrometer 1 the nominal and effective filterwheel positions are the same. For the other spectrometers the effective position is shifted relative to the nominal position by the corresponding number given listed in IOF entry 'Filter position offsets' (table 11). E.g. if 'Filter position offsets' equals 3 and the nominal filterwheel position is 4, then the effective filterwheel position for spectrometer 2 is 7.

Table 29: Columns in L1 file

Column name	Remark
Two letter code of measurement routine	See section 4.1
UT date and time for beginning of measurement, yyyymmddThhmmssZ (ISO 8601)	
Fractional days since 2000-1-1 UT midnight for beginning of measurement	
Routine count (1 for the first routine of the day, 2 for the second, etc.)	Corresponds to the
	L0 routine count
Repetition count (1 for the first set in the routine, 2 for the second, etc.)	Does not necessar-
	ily correspond to the
	L0 repetition count
Total duration of measurement set in seconds	
Latitude at the beginning of the measurement [deg], negative=South of equator, pos-	
itive=North of equator, -999=no latitude retrieved	
Longitude at the beginning of the measurement [deg], negative=West of Greenwich,	
positive=East of Greenwich, -999=no longitude retrieved	
Altitude a.s.l. at the beginning of the measurement [m], -999=no altitude retrieved	
West-East inclination angle at the beginning of the measurement [deg], -999=angle	
not retrieved	
North-South inclination angle at the beginning of the measurement [deg], -999=angle	
not retrieved	
Rotation angle at the beginning of the measurement [deg], -999=angle not retrieved	



Column name	Remark
Data processing type index	See table 7
Integration time [ms]	
Number of bright count cycles	0 if only dark count
	was measured
Number of dark count cycles	0 if no dark count
	was measured
Saturation index: positive integer is the number of saturated cycles included in the	
data, negative integer is the number of cycles skipped due to saturation	
Effective position of filterwheel #1, 0=filterwheel not used, 1-9 are valid positions	
Effective position of filterwheel #2, 0=filterwheel not used, 1-9 are valid positions	
Pointing zenith angle in degree, absolute or relative (see next column), 999=tracker	
not used	
Zenith pointing mode: zenith angle is 0=absolute, 1=relative to sun, 2=relative to	
moon	
Pointing azimuth in degree, increases clockwise, absolute (0=north) or relative (see	
next column), 999=tracker not used	
Azimuth pointing mode: like zenith angle mode but also fixed scattering angles rela-	
tive to sun (3) or moon (4)	
Mean over camera offsets [deg], -9=camera not in automatic mode	
Maximum of camera offsets [deg], -9=camera not in automatic mode	
Target distance [m], -1=not pointed on target	See section 4.1.12
Sum over 2 <sup>i</sup> with i being a L0 to L1 conversion step, 0=dark correction, 1=non-	See section 6.3
linearity correction, 2=latency correction, 3=flat field correction, 4=conversion to	
count rates, 5=temperature correction, 6=stray light correction, 7=wavelength change	
determination, 8=sensitivity correction, 9=wavelength correction	
Dark correction method: -9=no dark correction done, since it was not requested, -	
2=no dark correction done, since there were no bright measurements, -1=no dark	
correction done, since there was no matching dark measurement, 0=dark correction	
done with measured dark count only, since input darkmeth was MEAS or no dark	
map parameters available, 1=dark correction done using dark fine structure map and	
measured dark count, >1=dark correction done with measured dark count only, since	
dark background fitting gave an error	
Level 1 data quality flag: 0=high quality, 1=medium quality, 2=low quality	
Sum over 2 <sup>i</sup> using those i, for which the corresponding L1 data quality parameter ex-	See table 21
ceeds the DQ1 limit, 0=Saturated data, 1=Too few dark counts measurements, 2=No	
temperature given or effective temperature too different from the reference temper-	
ature, 3=Dark count too high, 4=Unsuccessful dark background fitting, 5=Absolute	
value of estimated average residual stray light level too high, 6=Although attempted,	
no wavelength change could be retrieved, 7=Retrieved wavelength shift too large,	
8=Retrieved wavelength shift differs too much from the shift predicted by the effec-	
tive temperature	
Sum over 2 <sup>i</sup> using those i, for which the corresponding L1 data quality parameter	See table 21
exceeds the DQ2 limit (same parameters as for DQ1)	
Wavelength effective temperature [°], 999=no effective temperature given	



Column name	Remark
Number of pixels, where dark count is higher than bright count, -9=no bright counts	
or dark counts measured	
Number of pixels, where DQ1 sigma ranges of dark count and bright count do not	
overlap, -9=no bright count uncertainty or dark count uncertainty measured	
Number of pixels, where DQ2 sigma ranges of dark count and bright count do not	
overlap, -9=no bright count uncertainty or dark count uncertainty measured	
Index of (not dead, blind, warm, or saturated) pixel with the highest corrected counts	
Uncertainty [%] of data from pixel with the highest counts based on measured bright	See also section 6.2
count uncertainty and measured dark count uncertainty in the case the uncertainty	
indicator is 9 or 10; -9 otherwise	
Uncertainty [%] of data from pixel with the highest counts based on measured bright	See also section 6.2
count uncertainty and theoretical dark count uncertainty in the case the uncertainty	
indicator is >=7; -9 otherwise	
Uncertainty [%] of data from pixel with the highest counts based on theoretical bright	See also section 6.2
count uncertainty and measured dark count uncertainty in the case the uncertainty	
indicator is 4, 9 or 10; -9 otherwise	
Uncertainty [%] of data from pixel with the highest counts based on theoretical bright	See also section 6.2
count uncertainty and theoretical dark count uncertainty in the case the uncertainty	
indicator is 3 or 4 or >=7; -9 otherwise	
Mean over blind and oversampled pixels in the bright counts, -9 if there is no bright	See also section 6.2
count	
Uncertainty of mean over blind and oversampled pixels in the bright counts, -9 if	See also section 6.2
there is no bright count uncertainty	
Mean over blind and oversampled pixels in the dark counts, -9 if there is no dark	See also section 6.2
count	
Uncertainty of mean over blind and oversampled pixels in the dark counts, -9 if there	See also section 6.2
is no dark count uncertainty	
Number of function evaluations used for dark background fitting, -9 if no background	See also section 6.3
fitting was done	
rms of unweighted fitting residuals for dark background fitting, -9 if no background	See also section 6.3
fitting was done	
Normalized rms of weighted fitting residuals for dark background fitting, -9 if no	See also section 6.3
background fitting was done	
Mean difference estimated minus measured dark count, -9 if no background fitting	See also section 6.3
was done	
rms of difference estimated minus measured dark count, -9 if no background fitting	See also section 6.3
was done	
Mean absolute difference between estimated and measured dark count expressed in	See also section 6.3
number of standard deviations, -9 if no background fitting was done	
Retrieved value for dark background fitting parameter X, -9 if no background fitting	X=1 until the value
was done	of ICF entry "Dark
	background fitting
	settings" plus three
Uncertainty of retrieved value for dark background fitting parameter X, -9 if no back-	X=1 until the value
ground fitting was done or no measured uncertainty was given	of ICF entry "Dark
	background fitting
	settings" plus two



Stray light correction method: 0=no stray light correction, 1=simple method, 2=corrected matrix method, 3=modified corrected matrix method, 4=uncorrected matrix method  Estimated average residual stray light level [%] (only valid for stray light correction methods 2 and higher)  Estimated stray light in the signal before correction at X nm [%], -9=no stray light correction, -1=signal after stray light correction negative, -2=estimated stray light negative, -3=signal before stray light correction negative  Number of iterations needed to reach tolerance goal at wavelength change determination  Last cost function at wavelength change determination  Last cost function at wavelength change determination  may of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3  See also section 6.3  See also section 6.3  See also section 6.3
Estimated average residual stray light level [%] (only valid for stray light correction methods 2 and higher)  Estimated stray light in the signal before correction at X nm [%], -9=no stray light correction, -1=signal after stray light correction negative, -2=estimated stray light negative, -3=signal before stray light correction negative  Number of iterations needed to reach tolerance goal at wavelength change determination, -9=no wavelength change determination  Last cost function at wavelength change determination [nm], -9=no wavelength change determination  rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3
Estimated average residual stray light level [%] (only valid for stray light correction methods 2 and higher)  Estimated stray light in the signal before correction at X nm [%], -9=no stray light correction, -1=signal after stray light correction negative, -2=estimated stray light negative, -3=signal before stray light correction negative  Number of iterations needed to reach tolerance goal at wavelength change determination  Last cost function at wavelength change determination  Last cost function at wavelength change determination [nm], -9=no wavelength change determination  rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3
rection methods 2 and higher)  Estimated stray light in the signal before correction at X nm [%], -9=no stray light correction, -1=signal after stray light correction negative, -2=estimated stray light negative, -3=signal before stray light correction negative  Number of iterations needed to reach tolerance goal at wavelength change determination, -9=no wavelength change determination  Last cost function at wavelength change determination [nm], -9=no wavelength change determination  rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3
Estimated stray light in the signal before correction at X nm [%], -9=no stray light correction, -1=signal after stray light correction negative, -2=estimated stray light negative, -3=signal before stray light correction negative  Number of iterations needed to reach tolerance goal at wavelength change determination, -9=no wavelength change determination  Last cost function at wavelength change determination [nm], -9=no wavelength change determination  rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3
light correction, -1=signal after stray light correction negative, -2=estimated stray light negative, -3=signal before stray light correction negative  Number of iterations needed to reach tolerance goal at wavelength change determination, -9=no wavelength change determination  Last cost function at wavelength change determination [nm], -9=no wavelength change determination  rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3  See also section 6.3
stray light negative, -3=signal before stray light correction negative  Number of iterations needed to reach tolerance goal at wavelength change determination, -9=no wavelength change determination  Last cost function at wavelength change determination [nm], -9=no wavelength change determination  rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3  See also section 6.3
Number of iterations needed to reach tolerance goal at wavelength change determination, -9=no wavelength change determination  Last cost function at wavelength change determination [nm], -9=no wavelength change determination  rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3  See also section 6.3
termination, -9=no wavelength change determination  Last cost function at wavelength change determination [nm], -9=no wavelength change determination  rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3
Last cost function at wavelength change determination [nm], -9=no wavelength change determination  rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3  See also section 6.3
length change determination  rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3
rms of final radiometric difference at wavelength change determination, -9=no wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3
wavelength change determination  Number of pixels not included in the wavelength change retrieval, -9=no wave-  See also section 6.3
Number of pixels not included in the wavelength change retrieval, -9=no wave- See also section 6.3
length change determination
Number of pixels not included in the wavelength correction, -9=no wavelength See also section 6.3
correction
Mean wavelength correction applied [%], -9=no wavelength correction See also section 6.3
Standard deviation of wavelength correction applied [%], -9=no wavelength See also section 6.3
correction
Minimum wavelength correction applied [%], -9=no wavelength correction See also section 6.3
Maximum wavelength correction applied [%], -9=no wavelength correction See also section 6.3
Expected wavelength shift based on effective temperature [nm], -9=no effec- See also section 6.3
tive temperature given
Retrieved wavelength change, order X, -9=no wavelength change determina- X=0 until the largest
tion order listed in ICF en-
tries "Wavelength change
retrieval parameters for
FUNCFILT"
Temperature at electronics board X [°C], 999=no temperature signal X=1, 2 or empty
Spectrometer control temperature X [°C], 999=no temperature signal X=1, 2 or empty
Auxiliary spectrometer temperature X [°C], 999=no temperature signal X=1, 2 or empty
Scale factor for data (to obtain unscaled data and uncertainty divide then by
this number)
Uncertainty indicator (see manual for exact meaning)  See table 30
Level 1 data type, data are 1=corrected count rate [s <sup>-1</sup> ], 2=radiance
[W/m <sup>2</sup> /nm/sr], 3=irradiance [W/m <sup>2</sup> /nm]
Level 1 data for each pixel, 0 is NOT a 'true' zero if the corresponding value
in the uncertainty is -2, -3 or -4
Uncertainty of level 1 data for each pixel, -1=no uncertainty was measured
and pixel is inside the range of the filter, -2=no uncertainty was measured
and pixel is outside the range of the filter, -3=uncertainty was measured, but
pixel is outside the range of the filter, -4=saturated data, -9=no uncertainty was
determined since no dark correction was attempted
Instrumental uncertainty of level 1 data for each pixel, -2=pixel is outside the
range of the filter, -4=saturated data



Table 30: Meaning of column 'Uncertainty indicator'

Number	Description
0	No bright count was measured and dark count was measured for one cycle only. Uncertainty is not valid. Instrumental uncertainty is based on theoretical dark count uncertainty.
1	No bright count was measured and dark count was measured with uncertainty based on the measured standard deviation of the data. Uncertainty is based on the measured dark count uncertainty. Instrumental uncertainty is based on the theoretical dark count uncertainty.
2	Bright count was measured for one cycle only and no dark count was measured. Uncertainty is not valid. Instrumental uncertainty is based on the theoretical bright count uncertainty.
3	Both bright and dark count were measured for one cycle only. Uncertainty is not valid. Instrumental uncertainty is based on the theoretical uncertainties of bright and dark count.
4	Bright count was measured for one cycle only and dark count was measured with uncertainty based on the measured standard deviation of the data. Uncertainty is not valid. Instrumental uncertainty is based on the theoretical uncertainties of bright and dark count.
5	Bright count was measured with uncertainty based on the measured standard deviation of the data and no dark count was measured. Uncertainty is based on the measured bright count uncertainty. Instrumental uncertainty is based on the theoretical bright count uncertainty.
6	Bright count was measured with uncertainty based on the measured standard error to a straight line fitted in the data and no dark count was measured. Uncertainty is based on the measured bright count uncertainty. Instrumental uncertainty is based on the theoretical bright count uncertainty.
7	Bright count was measured with uncertainty based on the measured standard deviation of the data and dark count was measured for one cycle only. Uncertainty is based on the measured bright count uncertainty and theoretical dark count uncertainty. Instrumental uncertainty is based on the theoretical uncertainties of bright and dark count.
8	Bright count was measured with uncertainty based on the measured standard error to a straight line fitted in the data and dark count was measured for one cycle only. Uncertainty is based on the measured bright count uncertainty and theoretical dark count uncertainty. Instrumental uncertainty is based on the theoretical uncertainties of bright and dark count.
9	Both bright and dark count were measured with uncertainty based on the measured standard deviation of the data. Uncertainty is based on the measured uncertainties of bright and dark count. Instrumental uncertainty is based on the theoretical uncertainties of bright and dark count.
10	Bright count was measured with uncertainty based on the measured standard error to a straight line fitted in the data and dark count was measured with uncertainty based on the measured standard deviation of the data. Uncertainty is based on the measured uncertainties of bright and dark count. Instrumental uncertainty is based on the theoretical uncertainties of bright and dark count.



# 5.13 L2 Files

#### **5.13.1 L2Fit Files**

BlickP converts the L1 data into L2Fit data by applying the L2Fit algorithm (see section 6.4). L2Fit files are daily ASCII-text files with a header, which includes meta data (see section 5.1), a description of the data columns, and the L2Fit data. L1 data with a "Data processing type index" of 0 (processing type "ONLYL1", see section 4.1.12) are not converted to L2Fit data.

L2Fit files are called PandoraXsY\_LLL\_YYYYMMDD\_L2Fit\_fFFFFCCpP.txt. X, Y, LLL, YYYYMMDD are as for the L0 file (section 5.7). The segment "fFFFFcCpP" is the "Data file version". C and P are as for the L1 file (section 5.12). FFFF is the "f-code" or "Fitting setup", that was used for the spectral fitting (see section 5.10.3). Only those L1 data are used for a specific L2Fit file, which have a "Data processing type index" (see section 4.1.12), that is the one required by the r-code (see 5.10.4). Obviously, if L1 data are processed with different data file versions, different L2Fit data files are produced, i.e. more than one L2Fit file can exist for a given f-code.

L2Fit files usually occupy less space than the corresponding L1 file, since the fitting window includes in general less pixels than exist on the spectrometer. Table 31 lists the maximum possible data columns in the L2Fit file. The real number of columns varies with the instrument, s-code and f-code. The meaning of each column is described in the header after the meta data.



Table 31: Columns in L2Fit file

Column name	Remark
Two letter code of measurement routine	See section 4.1
UT date and time at center of measurement, yyyymmddThhmmssZ (ISO 8601)	
Fractional days since 2000-1-1 UT midnight for center of measurement	
Routine count (1 for the first routine of the day, 2 for the second, etc.)	Corresponds to the L0 rou-
	tine count
Repetition count (1 for the first set in the routine, 2 for the second, etc.)	Corresponds to the L1 rep-
	etition count
Total duration of measurement set in seconds	
Latitude at the beginning of the measurement [deg], negative=South of equator,	
positive=North of equator, -999=no latitude retrieved	
Longitude at the beginning of the measurement [deg], negative=West of Green-	
wich, positive=East of Greenwich, -999=no longitude retrieved	
Altitude a.s.l. at the beginning of the measurement [m], -999=no altitude re-	
trieved	
Data processing type index	See table 7
Solar zenith angle at the center-time of the measurement in degree	
Solar azimuth at the center-time of the measurement in degree, 0=north, in-	
creases clockwise	
Lunar zenith angle at the center-time of the measurement in degree	
Lunar azimuth at the center-time of the measurement in degree, 0=north, in-	
creases clockwise	
Pointing zenith angle in degree, absolute or relative (see next column),	
999=tracker not used	
Zenith pointing mode: zenith angle is 0=absolute, 1=relative to sun, 2=rela-	
tive to moon	
Pointing azimuth in degree, increases clockwise, absolute (0=north) or relative	
(see next column), 999=tracker not used	
Azimuth pointing mode: like zenith angle mode but also fixed scattering angles	
relative to sun (3) or moon (4)	
Fitting result index: 0,1,2=no error or warning, 3=warning, >3=error	See table 32
Number of function evaluations used, 0=linear fitting	
rms of unweighted spectral fitting residuals, -9=fitting not successful	See equation 109
Normalized rms of weighted spectral fitting residuals, -9=fitting not successful	See equation 110
Expected rms based on measured uncertainty only, -9=fitting not successful or	See equation 111
no uncertainty given	9 119
Expected normalized rms based on measured uncertainty only, -9=fitting not	See equation 112
successful or no uncertainty given	0 111
Expected rms based on theoretical instrumental uncertainty only, -9=fitting not	See equation 111
successful or no uncertainty given	0 110
Expected normalized rms based on theoretical instrumental uncertainty only,	See equation 112
-9=fitting not successful or no uncertainty given	



Column name	Remark
GAS slant column amount [GASUNIT], -9e99=fitting not successful	Fitted slant column in unit
	GASUNIT for gas GAS
Uncertainty of GAS slant column amount [GASUNIT], -8=fitting not	Uncertainty of the GAS slant col-
successful, -1=cross section is zero in this wavelength range, -3=spectral	umn
fitting was done, but no uncertainty could be retrieved	
GAS effective temperature [K], -8=fitting not successful, 0=cross sec-	Fitted effective temperature for
tion is zero in this wavelength range or differential optical depth is too	gas GAS
small to retrieve the temperature	
Uncertainty of GAS effective temperature [K], -8=fitting not successful,	Uncertainty of fitted effective
-1=cross section is zero in this wavelength range, -2=differential optical	temperature for gas GAS
depth is too small to retrieve the temperature, -3=spectral fitting was	
done, but no uncertainty could be retrieved	
Direct GAS air mass factor	Direct moon air mass factor
	(AMF) for direct moon data, di-
	rect sun AMF otherwise, see
	equation 115
Climatological station pressure [hPa]	Possibly used to correct for
	molecular scattering OD
Molecular scattering air mass factor used for molecular scattering sub-	
traction before the fitting	
Climatological station temperature [K]	Possibly used in the L2 algorithm
Climatological effective O2O2 height [km]	Possibly used in the L2 algorithm
Fitted Ring spectrum, -9e99=fitting not successful	Ring spectrum slant column
Uncertainty of fitted Ring spectrum, -8=fitting not successful	
Fitting polynomial coefficient, order X	X=0 to maximum 10, see 6.4
Uncertainty of fitting polynomial coefficient, order X, -8=fitting not suc-	
cessful	
Offset polynomial coefficient, order X	X=0 to maximum 5, see 6.4
Uncertainty of offset polynomial coefficient, order X, -8=fitting not suc-	
cessful	
Wavelength change polynomial coefficient, order X	X=0 to maximum 5, see 6.4
Uncertainty of wavelength change polynomial coefficient, order X, -	
8=fitting not successful	
Resolution change polynomial coefficient, order X	X=0 to maximum 5, see 6.4
Uncertainty of resolution change polynomial coefficient, order X, -	
8=fitting not successful	
Level 2 Fit data quality flag: 0=high quality, 1=medium quality, 2=low	
quality	
Sum over 2 <sup>i</sup> using those i, for which the corresponding data quality pa-	See table 25
rameter exceeds the DQ1 limit, 0=L1 data quality above 0, 1=Spectral	
fitting was not successful, 2=Weighted rms of spectral fitting too large,	
3=Wavelength shift too large	
Sum over 2 <sup>i</sup> using those i, for which the corresponding data quality pa-	
rameter exceeds the DQ2 limit (same parameters as for DQ1)	
Level 1 data quality flag: 0=high quality, 1=medium quality, 2=low	As in L1
quality	



Column name	Remark
Sum over 2 <sup>i</sup> using those i, for which the corresponding L1	As in L1
data quality parameter exceeds the DQ1 limit, 0=Saturated data,	
1=Too few dark counts measurements, 2=No temperature given	
or effective temperature too different from the reference temper-	
ature, 3=Dark count too high, 4=Unsuccessful dark background	
fitting, 5=Absolute value of estimated average residual stray light	
level too high, 6=Although attempted, no wavelength change	
could be retrieved, 7=Retrieved wavelength shift too large, 8=Re-	
trieved wavelength shift differs too much from the shift predicted	
by the effective temperature	
Sum over 2 <sup>i</sup> using those i, for which the corresponding L1 data	As in L1
quality parameter exceeds the DQ2 limit (same parameters as for	
DQ1)	
Wavelength effective temperature [°C], 999=no effective temper-	As in L1
ature given	
Estimated average residual stray light level [%] (only valid for	
stray light correction methods 2 and higher)	
Retrieved wavelength shift from level 1 data [nm], -9=no wave-	
length change determination	
Retrieved wavelength shift from spectral fitting data [nm], -9=no	Note that this is in general 'in addition'
wavelength change fitting	to the wavelength shift from L1 data,
	since the L1 shift was included in the
	wavelength grid for the spectral fitting
Number of dark count cycles	As in L1
Effective position of filterwheel #1, 0=filterwheel not used, 1-9	As in L1
are valid positions	
Effective position of filterwheel #2, 0=filterwheel not used, 1-9	As in L1
are valid positions	
Sum over 2 <sup>i</sup> , 0=spectra were interpolated in time, 1=spectra are	See table 22
corrected for off-target signal	
Integration time [ms]	As in L1
Mean over camera offsets [deg], -9=camera not in automatic	As in L1
mode	
Maximum of camera offsets [deg], -9=camera not in automatic	As in L1
mode	
Unweighted slant column residuals for each pixel inside the fit-	
ting window multiplied by 1e5, 9e5=pixel was not used for fitting	
Normalized weighted slant column residuals for each pixel inside	
the fitting window multiplied by 1e5, 9e5=pixel was not used for	
fitting	



Table 32: Meaning fo fitting error indices

Index	Description
0	Successful fitting using measured uncertainty
1	Successful fitting using theoretical uncertainty
2	Successful fitting without including uncertainty
3	Some gases have not been fitted, since their optical depth was zero in the range of the valid data; otherwise successful fitting
4	The reference data do not fully include the wavelength range of the fitting
5	No fitting done since number of measured data or reference data is smaller than number of fitting parameters
6	No fitting because of rank deficiency in design matrix
7	lstsq raised LinAlgError
8	Negative values were in the covariance matrix
9	Encountered singular covariance matrix in non-linear fitting
>9	Errors in leastsq (see description of scipy.optimize.leastsq function), e.g. 10=maximum number of function evaluations reached

#### 5.13.2 L2Tot and L2Trop Files

BlickP converts the L2Fit data into L2Tot and L2Trop data by applying an L2 algorithm (see section 6.5). L2Tot files contain total vertical column trace gas amounts retrieved from direct sun or moon observations. L2Trop files contain trace gas information in the troposphere such as surface concentration, tropospheric vertical column amount and layer height. These "higher order L2 files" (L2H) are ASCII-text files with a header, which includes meta data (see section 5.1), a description of the data columns, and the data. L2H files are not daily files anymore. They include the whole time series of measurements for a given instrument at a given location.

L2H files are called PandoraXsY\_LLL\_L2H\_rRRRRPP.txt. X, Y, and LLL are as for the L0 file (section 5.7). H can be "Tot" or "Trop". The segment "rRRRRPP" is the "Data file version". P is as for the L1 file (section 5.12). RRRR is the "r-code" or "Retrieval setup", that was used for the retrieval (see section 5.10.4). Note that the data file version does not include information on the ICF version used. Since the ICF version is "allowed" to vary over time, it is listed as a column in the data themselves (see table 33).

The size of the L2H files depends on the length of the data base at a station, but they are usually much smaller than L0, L1, or L2Fit files, since no spectral information is included. Table 33 lists the maximum possible data columns in the L2H file. The real number of columns varies with the instrument and L2H type (see column "Remark"). The meaning of each column is described in the header after the meta data.



Table 33: Columns in L2H file

Column name	Remark
UT date and time for center of measurement, yyyymmddThhmmssZ	For retrievals including more
(ISO 8601)	than one spectrum, the "Center
	of measurement" is the center-
	time of the spectrum, to which all
	other are interpolated
Fractional days since 2000-1-1 UT midnight for center of measurement	
Total duration of measurement in seconds	From start of first spectra to end
	of last spectra taken
Latitude at the beginning of the measurement [deg], negative=South of equator, positive=North of equator, -999=no latitude retrieved	
Longitude at the beginning of the measurement [deg], negative=West of	
Greenwich, positive=East of Greenwich, -999=no longitude retrieved	
Altitude a.s.l. at the beginning of the measurement [m], -999=no altitude	
retrieved	
Solar zenith angle for center of measurement in degree	
Solar azimuth for center of measurement in degree, 0=north, increases	
clockwise	
Lunar zenith angle for center of measurement in degree	L2Tot only
Lunar azimuth for center of measurement in degree, 0=north, increases	L2Tot only
clockwise	
Pointing zenith angle for center of measurement in degree	L2Trop only
Pointing azimuth for center of measurement in degree, 0=north, in-	L2Trop only
creases clockwise	
GAS total vertical column amount [GASUNIT], -9e99=retrieval not	Retrieved total vertical column
successful	amount for gas GAS in unit
	GASUNIT, L2Tot only
Uncertainty of GAS total vertical column amount [GASUNIT], -	L2Tot only
8=retrieval not successful, -1=cross section is zero in this wavelength	
range, -3=spectral fitting was done, but no uncertainty could be retrieved	
GAS effective temperature [K], -8=retrieval not successful, 0=cross sec-	Retrieved effective temperature
tion is zero in this wavelength range or differential optical depth is too	for gas GAS, L2Tot only
small to retrieve the temperature	
Uncertainty of GAS effective temperature [K], -8=retrieval not success-	L2Tot only
ful, -1=cross section is zero in this wavelength range, -2=differential	
optical depth is too small to retrieve the temperature, -3=spectral fitting	
was done, but no uncertainty could be retrieved	
GAS surface concentration [ppb], -9e99=retrieval not successful	Retrieved surface concentration for gas GAS in ppb, L2Trop only
Uncertainty of GAS surface concentration [ppb], -8=retrieval not successful	L2Trop only
GAS tropospheric vertical column amount [GASUNIT], -9e99=retrieval	Retrieved tropospheric vertical
not successful	column amount for gas GAS in
not buccossiui	unit GASUNIT, L2Trop only
Uncertainty of GAS tropospheric vertical column amount [GASUNIT],	L2Trop only
-8=retrieval not successful	· r · J



Column name	Remark
GAS surface concentration index, integer >0 (see manual), -8=retrieval	See section 6.5.2; L2Trop only
not successful	
GAS heterogeneity flag, 0=well mixed conditions, 1=heterogeneous	See section 6.5.2; L2Trop only
conditions, -8=retrieval not successful	
L2 data quality flag for GAS: 0=high quality, 1=medium quality, 2=low	
quality	
Sum over 2 <sup>i</sup> using those i, for which the corresponding L2 data quality	See table 27
parameter for GAS exceeds the DQ1 limit, 0=L2Fit data quality above	
0, 1=Uncertainty too high, 2=Signal to noise ratio too low, 3=Air mass	
factor too large	
Sum over 2 <sup>i</sup> using those i, for which the corresponding L2 data quality	See table 27
parameter for GAS exceeds the DQ2 limit (same parameters as for DQ1)	
Direct GAS air mass factor	As in L2Fit
Fitting result index: 1,2=no error or warning, >2=error or warning	For L2Trop this is the maximum
	over all sets
rms of unweighted spectral fitting residuals, -9=fitting not successful	For L2Trop this is the mean, min-
	imum and maximum over all sets
Normalized rms of weighted spectral fitting residuals, -9=fitting not suc-	For L2Trop this is the mean, min-
cessful	imum and maximum over all sets
Expected rms based on measured uncertainty only, -9=fitting not suc-	L2Tot only, see equation 111
cessful or no uncertainty given	
Expected normalized rms based on measured uncertainty only, -	L2Tot only
9=fitting not successful or no uncertainty given	
Climatological station pressure [mbar]	
Climatological station temperature [K]	L2Trop only
Climatological effective O2O2 height [km]	L2Trop only
Data processing type index	See table 7
Calibration file version used	The version of the ICF used for
	this retrieval
Level 2 Fit data quality flag: 0=high quality, 1=medium quality, 2=low	For L2Trop this is the minimum
quality	and maximum over all sets
Sum over 2i using those i, for which the corresponding data quality pa-	As in L2Fit, for L2Trop this is the
rameter exceeds the DQ1 limit, 0=L1 data quality above 0, 1=Spectral	maximum over all sets
fitting was not successful, 2=Weighted rms of spectral fitting too large,	
3=Wavelength shift too large	
Sum over 2 <sup>i</sup> using those i, for which the corresponding data quality pa-	For L2Trop this is the maximum
rameter exceeds the DQ2 limit (same parameters as for DQ1)	over all sets
Level 1 data quality flag: 0=high quality, 1=medium quality, 2=low	As in L1, for L2Trop this is the
quality	minimum and maximum over all
	sets



Column name	Remark
Sum over 2 <sup>i</sup> using those i, for which the corresponding L1	As in L1, for L2Trop this is the maxi-
data quality parameter exceeds the DQ1 limit, 0=Saturated data,	mum over all sets
1=Too few dark counts measurements, 2=No temperature given	
or effective temperature too different from the reference temper-	
ature, 3=Dark count too high, 4=Unsuccessful dark background	
fitting, 5=Absolute value of estimated average residual stray light	
level too high, 6=Although attempted, no wavelength change	
could be retrieved, 7=Retrieved wavelength shift too large, 8=Re-	
trieved wavelength shift differs too much from the shift predicted	
by the effective temperature	
Sum over 2 <sup>i</sup> using those i, for which the corresponding L1 data	As in L1, for L2Trop this is the maxi-
quality parameter exceeds the DQ2 limit (same parameters as for	mum over all sets
DQ1)	
Wavelength effective temperature [°C], 999=no effective temper-	As in L1, for L2Trop this is the mean
ature given	over all sets
Estimated average residual stray light level [%] (only valid for	As in L1, for L2Trop this is the mean,
stray light correction methods 2 and higher)	minimum and maximum over all sets
Retrieved wavelength shift from level 1 data [nm], -9=no wave-	As in L1, for L2Trop this is the mean,
length change determination	minimum and maximum over all sets
Retrieved wavelength shift from spectral fitting [nm], -9=no	As in L2Fit, for L2Trop this is the mean,
wavelength change fitting	minimum and maximum over all sets
Integration time [ms]	As in L1, for L2Trop this is the mini-
	mum and maximum over all sets
Number of dark count cycles	As in L1, for L2Trop this is the mean,
	minimum and maximum over all sets
Effective position of filterwheel #1, 0=filterwheel not used, 1-9	As in L1, for L2Trop this is the mini-
are valid positions	mum and maximum over all sets
Effective position of filterwheel #2, 0=filterwheel not used, 1-9	As in L1, for L2Trop this is the mini-
are valid positions	mum and maximum over all sets

# 5.14 Log Files

BlickO, BlickP, and BlickF create log-files, which report information, warnings and errors. Those files are in directories /log/oslog, /log/pslog and /log/fslog respectively.

#### **BlickO log files**

They are called PandoraX\_LLL\_YYYYMMDD\_NNNlog.txt. X is the instrument number, LLL the Short Location Name (see section 5.2), YYYYMMDD is the UT-date for the time of the local noon and NNN can be "info", "warning", or "error" for the info-file, warning-file, and error-file respectively. The info-file includes information, warnings and errors. The warning-file includes warnings and errors. The error-file only includes errors. Hence the warning-file is a subset of the info-file and the error-file a subset of the warning-file. Each line in the log-file looks like this example:

```
[Thu 18 Dec 2014, 10:24:47.722] [INFO] [Connected spectrometer 1]
```

It starts with the UT-time, then the error-level, which can be "INFO, "WARNING" or "ERROR" and finally the



error message. Note that most log-entries also added to the L0-files, especially all warnings and errors.

#### BlickF log files

The actual BlickF log-file is called BlickF\_log.txt. Once the log file has reached 5000 lines, it is renamed to BlickF\_logYYYYMMDDThhmmssZ.txt, where the time of the last entry in the file is added to the filename. The actual log file is then started with zero lines. The BlickF log-files include all the messages written by BlickF on the command window (see figure 5), except the lines saying 'No action taken', for example:

```
Sat 16 Jan 2016, 08:45:20, Initiated new data pushing to remote server; 2 files still to be pushed (OF=01)
```

#### BlickP log files

The actual BlickP log-file is called BlickP.log. Once the log file has reached 5000 lines, it is renamed to BlickP\_YYYYMMDDThhmmssZ.log, where the time of the last entry in the file is added to the filename. The actual log file is then started with zero lines. The BlickP log-files include all the messages written by BlickP on the command window. Below are some sample lines. Each line starts with the time, follwed by INFO, WARNING, or ERROR, then in parenthesis the module, from which it is called, and finally the log-message.

```
[2017-01-12 13:25:55,237] INFO(blickp): BlickP starting up [2017-01-12 13:25:55,239] INFO(blickp): Reading configuration [2017-01-12 13:25:55,278] INFO(blickp): Processing day: 2016/05/10 [2017-01-12 13:25:55,918] INFO(blickp.l2fitprocessor): Processing L1-File: Pandoral10s1_IBK_20160510_L1_sNWLCc3p1-2.txt with F-Code OORI
```

# 5.15 Alignment Files

BlickO creates so-called alignment files in directory /data/alignments. These files are needed to determine the position of the Pandora unit at the current location in order to ensure correct pointing of the system. For more details on the alignment see section 6.1.

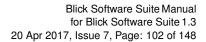
The alignment file is called PandoraX\_LLL\_alignments.txt. X is the instrument number and LLL is the Short Location Name (see section 5.2). It is an ASCII-text file with a header, which includes meta data and a description of the data columns, and the "alignment lines". A new line is written each time a successful sun search has been performed.

Each alignment line has  $4*n_{SPEC}+4$  lines, where  $n_{SPEC}$  is the number of spectrometers in the Pandora system. An alignment line for a system with two spectrometers looks like this example:

```
20141218T112919Z FS 65.72 196.23 65.78 196.25 0.027 1362.33 65.74 196.16 0.027 1362.33
```

After the UT-time (column 1) and the name of the routine used (column 2), there are the "true" solar (or lunar) zenith angle (column 3) and azimuth (column 4). Those true angles are calculated based on time and location using the formulation of *Jensen et al.* [13].

Then there are groups of 4 columns, one for each spectrometer. The first two columns in the group are the "apparent" solar (or lunar) zenith angle and azimuth respectively for this spectrometer. These are the angles, in which the instrument observed the sun (or moon). The 3rd column in the group is the rms of the field of view





fitting for this alignment line for this spectrometer. The 4th column in the group finally is the weighting factor for this alignment line for this spectrometer.

Note that only a maximum of 5000 alignment lines are stored in the alignment file. If this number is exceeded then two things are happening:

- 1) The alignment file is reduced to the last 5000 entries.
- 2) The earlier alignments are stored in a "historic alignment file", which is called  $PandoraX\_LLL\_alignments\_YYYYMMDDTHHMMSSZ.txt$ , where YYYYMMDDTHHMMSSZ is the UT-time of the file creation in ISO 8601 format.



# 6 Algorithms Theoretical Basis

# 6.1 Alignment Algorithm

This section describes the theoretical background (section 6.1.1) and the practical implementation (section 6.1.2) of the Pandora alignment. It also addresses the problem of different FOVs for sun and sky observations and for Pandora units with more than one spectrometer (section 6.1.3).

#### 6.1.1 Theoretical Background

After proper installation of Pandora (at a Northern Hemisphere location) and after a tracker reset, the entrance window of the head sensor should look to the zenith and the head sensor should be at the East of the shaft holding the tracker as shown in figure 8.

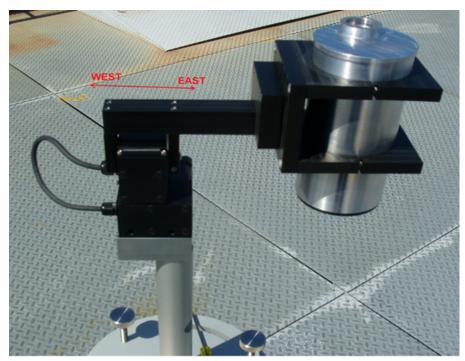


Figure 8: Head sensor position after tracker reset

If we assume that everything is perfectly aligned, i.e. the tracker base plate is perfectly leveled and rotated in the right direction, the tracker's zenith and azimuth motions are perfectly perpendicular to each other, and the optical axis inside the head sensor is perfectly parallel to the axis of the zenith motion, then Pandora is in the so-called "horizontal reference frame" (figure 9). The x-axis goes from South (negative x) to North (positive x), the y-axis goes from East (negative y) to West (positive y), and the z-axis goes from Nadir (negative z) to Zenith (positive z).

To describe a point P in the reference frame (the arrow in figure 9, which represents the optical axis of the head sensor), we use Cartesian or spherical coordinates. In the Cartesian system P is characterized by (x,y,z), where x, y, and z are the projections of P to the x-, y-, and z-axis respectively. In the spherical system P is described by  $(\rho,\theta,\varphi)$ , where  $\rho$  is the distance from the origin,  $\theta$  is the zenith angle (between  $0^{\circ}$  and  $180^{\circ}$ ), measured from the positive z-axis, and  $\varphi$  is the azimuth (between  $0^{\circ}$  and  $360^{\circ}$ ), measured from the positive x-axis to the projection of P on the x-y-plane, clockwise when looking from the zenith to the origin. Note that this definition of  $\varphi$  is different from the standard spherical coordinate convention, where  $\varphi$  is taken counterclockwise. So in our definition  $\varphi$ = $0^{\circ}$  is North,  $\varphi$ = $90^{\circ}$  East,  $\varphi$ = $180^{\circ}$  South, and  $\varphi$ = $270^{\circ}$  West.



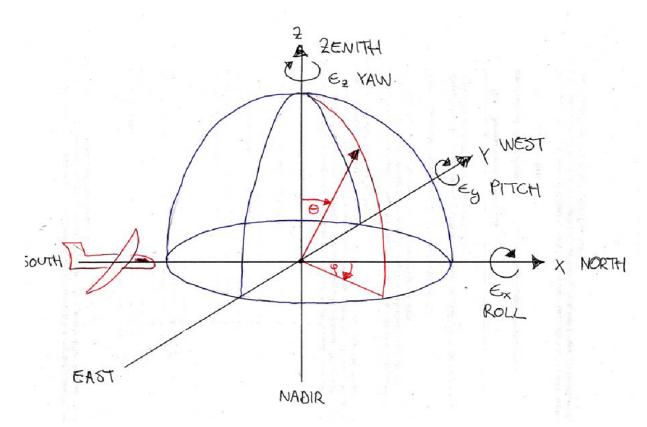


Figure 9: Horizontal frame

Using this definition, the conversion from spherical to Cartesian coordinates is given in equation 16.

$$x = \rho \cdot \sin \theta \cdot \cos \varphi$$

$$y = -\rho \cdot \sin \theta \cdot \sin \varphi$$

$$z = \rho \cdot \cos \theta$$
(16)

The conversion from Cartesian to spherical coordinates is given in equation 17.

$$\rho = \sqrt{x^2 + y^2 + z^2}$$

$$\theta = \arccos\left(\frac{z}{\sqrt{x^2 + y^2 + z^2}}\right)$$

$$\varphi = \begin{cases} \arccos\left(\frac{x}{\sqrt{x^2 + y^2}}\right) & \text{if } y \le 0 \\ 2\pi - \arccos\left(\frac{x}{\sqrt{x^2 + y^2}}\right) & \text{if } y > 0 \end{cases}$$
(17)

Since in this context we always deal with "directions" rather than "points",  $\rho$  equals 1 in all situations and all vectors P introduced are unit vectors.

In practice it is very difficult to have a perfectly aligned instrument. The base plate of the tracker is in general misaligned, i.e. rotated relative to the horizontal reference frame. To describe this we introduce the



"tracker reference system", where the x-y-plane is the plane defined by the tracker azimuth motion and the z-axis is simply perpendicular to this plane.

Assume we have the coordinates of a point  $P_T$  in the tracker frame. In order to obtain the coordinates of the same point in the horizontal frame  $P_H$ , we have to apply a full rotation on vector  $P_T$ . A full rotation can be described as a sequence of 3 rotations around the x-, y-, and z-axis for angles  $\epsilon_x$ ,  $\epsilon_y$  and  $\epsilon_z$  (see figure 9). A positive angle corresponds to a clockwise rotation of the respective axis, when looking from the positive axis towards the origin. Note that in the horizontal reference frame  $\epsilon_x$ ,  $\epsilon_y$  and  $\epsilon_z$  correspond to roll, pitch, and yaw of an airplane flying along the x-axis to increasing values (from South to North, see figure 9). Equation 18 gives the relationship of  $P_T$  and  $P_H$ :

$$P_H = R_z(\epsilon_z) \cdot R_y(\epsilon_y) \cdot R_x(\epsilon_x) \cdot P_T = R(\epsilon_x, \epsilon_y, \epsilon_z) \cdot P_T$$
(18)

 $P_H$  and  $P_T$  are the 3x1-elements vectors with the Cartesian coordinates of the point.  $R_x$ ,  $R_y$ , and  $R_z$  are 3x3-elements rotation matrices defined in equation 19.

$$R_{x}(\epsilon_{x}) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\epsilon_{x} & sin\epsilon_{x} \\ 0 & -sin\epsilon_{x} & cos\epsilon_{x} \end{pmatrix}$$

$$R_{y}(\epsilon_{y}) = \begin{pmatrix} cos\epsilon_{y} & 0 & -sin\epsilon_{y} \\ 0 & 1 & 0 \\ sin\epsilon_{y} & 0 & cos\epsilon_{y} \end{pmatrix}$$

$$R_{z}(\epsilon_{z}) = \begin{pmatrix} cos\epsilon_{z} & sin\epsilon_{z} & 0 \\ -sin\epsilon_{z} & cos\epsilon_{z} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$
(19)

R is simply the product of the 3 matrices (in this order!). So given a point in the tracker frame  $P_T$  and knowing the leveling angles  $\epsilon_x$ ,  $\epsilon_y$  and  $\epsilon_z$  we can calculate the coordinates of this point in the horizontal frame  $P_H$ . For the opposite, i.e. getting  $P_T$  when having  $P_H$ , we have to invert equation 18.

$$P_T = R^{-1}(\epsilon_x, \epsilon_y, \epsilon_z) \cdot P_H = R_x(-\epsilon_x) \cdot R_y(-\epsilon_y) \cdot R_z(-\epsilon_z) \cdot P_H$$
(20)

Hence if the optical axis inside the head sensor is perfectly parallel to the axis of the zenith motion or, in other words, after a tracker reset, the instrument's optical axis equals the z-axis of the tracker frame (i.e. the head sensor is not slightly tilted inside its mounting bracket; see figure 8), then we can fully describe the alignment of the instrument by the 3 angles  $\epsilon_x$ ,  $\epsilon_y$  and  $\epsilon_z$ .

However in practice the optical axis after a tracker reset is not exactly parallel to the z-axis. It is in general rotated relative to the z-axis. Here we assume the direction of the optical axis after the tracker reset is characterized by zenith angle  $\vartheta$  and azimuth  $\phi$ . Then the position of the optical axis after a zenith motion of  $\theta$  and an azimuth motion of  $\varphi$  in the tracker reference frame is given by equation 21.

$$P_T = R_z(\varphi) \cdot R_u(-\theta) \cdot R_z(\phi) \cdot R_u(-\theta) \cdot e_z = R(0, -\theta, \varphi) \cdot R(0, -\theta, \phi) \cdot e_z = R(0, -\theta, \varphi) \cdot P_{0AXT}$$
 (21)

 $e_z$  is the 3x1-elements z-axis-unit-vector (0,0,1). Applying  $R(0,-\vartheta,\phi)$  on  $e_z$  gives the 3x1-elements vector  $P_{0AXT}$ , the direction of the optical axis after the tracker reset. Then applying  $R(0,-\theta,\varphi)$  on  $P_{0AXT}$  gives  $P_T$ , the direction of the optical axis after having moved the zenith motor for  $\theta$  and the azimuth motor for  $\varphi$ . So given the zenith and azimuth motion angles  $\theta$  and  $\varphi$  and knowing the position of the optical axis after the tracker reset relative to the z-axes  $\vartheta$  and  $\varphi$  we can calculate the coordinates of the optical axis in the tracker frame  $P_T$ . For



the opposite, i.e. getting  $\theta$  and  $\varphi$  when having  $P_T$ , we have to invert equation 21, which is not so easy, but can be done

Combining equations 18 and 21, we obtain  $P_H$  as a function of  $\theta$ ,  $\varphi$ ,  $\epsilon_x$ ,  $\epsilon_y$ ,  $\epsilon_z$ ,  $\vartheta$  and  $\varphi$ .

$$P_H = R(\epsilon_x, \epsilon_y, \epsilon_z) \cdot R(0, -\theta, \varphi) \cdot R(0, -\theta, \phi) \cdot e_z$$
(22)

So once we have the full set of leveling angles  $\epsilon_x$ ,  $\epsilon_y$ ,  $\epsilon_z$ ,  $\vartheta$  and  $\phi$ , we can calculate the "true" pointing in the horizontal reference frame  $P_H$  for each pair of tracker motion angles  $\theta$  and  $\varphi$ . And, using the inversion of equation 22, we can calculate the motion angles  $\theta$  and  $\varphi$  of the tracker needed to point the instrument to the absolute position  $P_H$ .

# 6.1.2 Practical Implementation

The Pandora software executes so-called "sun-searches" in a certain frequency over the day (usually every 15 min during the initial days after instrument installation and less frequent afterwards). During a sun-search the instrument moves to the "true" solar position based on astronomical calculations and the geographical coordinates of the location [13], then it scans the sky in zenith direction ("up-down") and in azimuth direction ("left-right") and looks for the position of maximum signal. This position is saved as the "apparent" solar position (see section 5.15). An example for such a sun search is shown in figure 10.

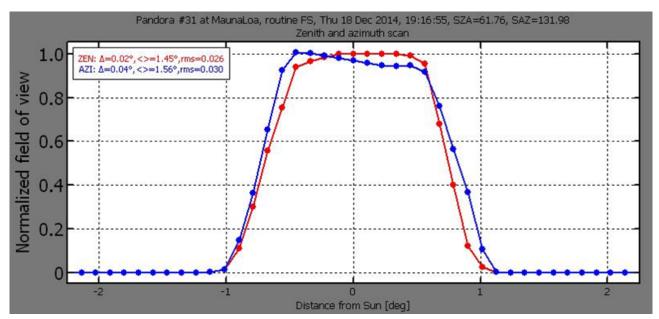


Figure 10: Sun search for Pandora #31 at Mauna Loa on 18 Dec 2014. Red line is the sun scan in zenith direction ("up-down") and blue line in azimuth direction ("left-right"). In this case the apparent angles differ from the true angles by  $0.02^{\circ}$  in zenith and  $0.04^{\circ}$  in azimuth.

After a successful sun-search we can calculate angles  $(\theta_H, \varphi_H)$ , which correspond to point  $P_H$ , the position of the sun in the horizontal reference frame, and we have measured angles  $(\theta_T, \varphi_T)$ , the tracker motion angles of the maximum signal, which correspond to  $\theta$  and  $\varphi$  in equation 22. Now we try to solve equation 22 with given  $\theta_H$ ,  $\varphi_H$ ,  $\theta_T$  and  $\varphi_T$  to obtain the leveling angles. The technique BlickO uses is based on a linearization of equation 22 and iterations to find the solution. Equation 23 shows the first order approximations of the rotation matrices from equation 19 at position  $\epsilon$ :



$$R_{x}(\epsilon + \Delta \epsilon) \approx \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\epsilon & sin\epsilon \\ 0 & -sin\epsilon & cos\epsilon \end{pmatrix} + \Delta \epsilon \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & -sin\epsilon & -cos\epsilon \\ 0 & cos\epsilon & -sin\epsilon \end{pmatrix} = R_{x}(\epsilon) + \Delta \epsilon \cdot V_{x}(\epsilon)$$

$$R_{y}(\epsilon + \Delta \epsilon) \approx \begin{pmatrix} cos\epsilon & 0 & -sin\epsilon \\ 0 & 1 & 0 \\ sin\epsilon & 0 & cos\epsilon \end{pmatrix} + \Delta \epsilon \cdot \begin{pmatrix} -sin\epsilon & 0 & cos\epsilon \\ 0 & 1 & 0 \\ -cos\epsilon & 0 & -sin\epsilon \end{pmatrix} = R_{y}(\epsilon) + \Delta \epsilon \cdot V_{y}(\epsilon)$$

$$R_{z}(\epsilon + \Delta \epsilon) \approx \begin{pmatrix} cos\epsilon & sin\epsilon & 0 \\ -sin\epsilon & cos\epsilon & 0 \\ 0 & 0 & 1 \end{pmatrix} + \Delta \epsilon \cdot \begin{pmatrix} -sin\epsilon & -cos\epsilon & 0 \\ cos\epsilon & -sin\epsilon & 0 \\ 0 & 0 & 1 \end{pmatrix} = R_{z}(\epsilon) + \Delta \epsilon \cdot V_{z}(\epsilon)$$

$$(23)$$

The 3-3-elements matrices  $V_x$ ,  $V_y$ , and  $V_z$  are the first derivatives of the rotation matrices in  $\epsilon$ . Using equation 23 we can linearize equation 22 at position  $\epsilon_x$ ,  $\epsilon_y$ ,  $\epsilon_z$ ,  $\vartheta$  and  $\phi$ .

$$P_{H} \approx R_{z}(\epsilon_{z}) \cdot R_{y}(\epsilon_{y}) \cdot R_{x}(\epsilon_{x}) \cdot R(0, -\theta, \varphi) \cdot R_{z}(\phi) \cdot R_{y}(-\vartheta) \cdot e_{z}$$

$$+ \Delta \epsilon_{x} \cdot R_{z}(\epsilon_{z}) \cdot R_{y}(\epsilon_{y}) \cdot V_{x}(\epsilon_{x}) \cdot R(0, -\theta, \varphi) \cdot R_{z}(\phi) \cdot R_{y}(-\vartheta) \cdot e_{z}$$

$$+ \Delta \epsilon_{y} \cdot R_{z}(\epsilon_{z}) \cdot V_{y}(\epsilon_{y}) \cdot R_{x}(\epsilon_{x}) \cdot R(0, -\theta, \varphi) \cdot R_{z}(\phi) \cdot R_{y}(-\vartheta) \cdot e_{z}$$

$$+ \Delta \epsilon_{z} \cdot V_{z}(\epsilon_{z}) \cdot R_{y}(\epsilon_{y}) \cdot R_{x}(\epsilon_{x}) \cdot R(0, -\theta, \varphi) \cdot R_{z}(\phi) \cdot R_{y}(-\vartheta) \cdot e_{z}$$

$$+ \Delta \vartheta \cdot R_{z}(\epsilon_{z}) \cdot R_{y}(\epsilon_{y}) \cdot R_{x}(\epsilon_{x}) \cdot R(0, -\theta, \varphi) \cdot R_{z}(\phi) \cdot V_{y}(-\vartheta) \cdot e_{z}$$

$$+ \Delta \varphi \cdot R_{z}(\epsilon_{z}) \cdot R_{y}(\epsilon_{y}) \cdot R_{x}(\epsilon_{x}) \cdot R(0, -\theta, \varphi) \cdot V_{z}(\phi) \cdot R_{y}(-\vartheta) \cdot e_{z}$$

$$+ \Delta \varphi \cdot R_{z}(\epsilon_{z}) \cdot R_{y}(\epsilon_{y}) \cdot R_{x}(\epsilon_{x}) \cdot R(0, -\theta, \varphi) \cdot V_{z}(\phi) \cdot R_{y}(-\vartheta) \cdot e_{z}$$

$$P_{H} \approx P + \Delta \epsilon_{x} \cdot C_{x} + \Delta \epsilon_{y} \cdot C_{y} + \Delta \epsilon_{z} \cdot C_{z} + \Delta \vartheta \cdot C_{\vartheta} + \Delta \phi \cdot C_{\varphi}$$

All  $C_x$ ,  $C_y$ , etc. in equation 24 are 3x1-element vectors. Rearranging 24 we obtain a system of linear equations:

$$P_{H} - P = \Delta P = \begin{pmatrix} C_{xx} & C_{yx} & C_{zx} & C_{\vartheta x} & C_{\varphi x} \\ C_{xy} & C_{yy} & C_{zy} & C_{\vartheta y} & C_{\varphi y} \\ C_{xz} & C_{yz} & C_{zz} & C_{\vartheta z} & C_{\varphi z} \end{pmatrix} \cdot \begin{pmatrix} \Delta \epsilon_{x} \\ \Delta \epsilon_{y} \\ \Delta \epsilon_{z} \\ \Delta \vartheta \\ \Delta \phi \end{pmatrix} = C \cdot \Delta \epsilon$$
 (25)

Equation 25 cannot be solved in a unique way, since the rank of matrix C is 2, which means there are only 2 independent columns in it. Or in other words we can make many possible choices for the leveling angles that fulfill equation 25. This makes sense, since we are trying to get 5 pieces of information out of equation 25, while putting only 2 pieces of information in. Nevertheless if we combine the results of several sun searches we can add them up and obtain equation 26.

$$\Delta P = \begin{pmatrix} \Delta P_1 \\ \Delta P_2 \\ \dots \\ \Delta P_N \end{pmatrix} = \begin{pmatrix} C_1 \\ C_2 \\ \dots \\ C_N \end{pmatrix} \cdot \Delta \epsilon = C \cdot \Delta \epsilon$$
 (26)

N is the number of sun searches used,  $\Delta P$  is a 3Nx1-elements vector, and C is a 3Nx5-elements matrix. For N>2 equation 26 is an overdetermined system of linear equations, which we can solve in a least squares sense with the pseudo-inverse matrix.



$$\Delta \epsilon = [\mathbf{C}^T \cdot \mathbf{C}]^{-1} \cdot \mathbf{C}^T \cdot \Delta \mathbf{P} = pinv(\mathbf{C}) \cdot \Delta \mathbf{P}$$
(27)

This is how BlickO determines the leveling of the instrument in an iterative process:

- 1. Read the results of the successful sun searches from the alignments file (see section 5.15).
- 2. Take a first guess for the leveling angles. At the very beginning the first guess is  $\epsilon$ =(0,0,0,0,0). When a previous determination of the leveling had been done, then this is used as first guess.
- 3. Calculate  $\Delta P$  and C using  $\epsilon$  and the sun search results as in equation 26.
- 4. Calculate the "cost-function", which is the sum-of-squares over  $\Delta P$ .
- 5. Get  $\Delta \epsilon$  from equation 27.
- 6. Get the next guess for the leveling angles  $\epsilon = \epsilon + \Delta \epsilon$ .
- 7. Repeat steps 3 to 6 until the cost function calculated in step 4 changes below a certain tolerance (in BlickO set to 1e-5).

This method usually reaches a solution within a few (<10) iterations and lasts less than 0.2 s.

The more sun search results are available, the better. However it is even more important that the sun searches are spread over a large range of zenith angles and azimuths. I.e. 3 sun search, one from the morning, one from noon and one from evening, work better than 10 sun searches measured in a row. Therefore we recommend to run an instrument, which has been installed at a new location, for about one day in schedule "align", which focuses on sun searches (see section 4.2). In principle one could also use moon searches to improve the instrument alignment, but we do not do this for two reasons:

- 1. The accuracy of the formulas to calculate the lunar position is 0.14° and comparable to our tracking accuracy, while for the solar position it is 0.02° [13].
- 2. When the moon is not entirely full, the moon search gives a small offset, since the illuminated area is not centered around the center of the moon.

A special case is when only one sun search is available (as in equation 25). Then BlickO puts some constraints on the leveling angles. It sets  $\theta$ =0 and  $\phi$ =0 and fixes the relationship between  $\epsilon_x$  and  $\epsilon_y$ .

$$\frac{\epsilon_x}{\epsilon_y} = tan(\varphi_H) \quad \text{with} \quad \left| \frac{\epsilon_x}{\epsilon_y} \right| < 1000$$
 (28)

So e.g. if the solar azimuth is  $180^{\circ}$  (=South), then  $\epsilon_x$  is set to zero, or if the solar azimuth is  $225^{\circ}$  (=South-West), then  $\epsilon_x = \epsilon_y$ . With these constraints 25 reduces to 29, which can be solved with the iterative method described above.

$$\Delta P = \begin{pmatrix} C_{xx} + tan(\varphi_H) \cdot C_{yx} & C_{zx} \\ C_{xy} + tan(\varphi_H) \cdot C_{yy} & C_{zy} \\ C_{xz} + tan(\varphi_H) \cdot C_{yz} & C_{zz} \end{pmatrix} \cdot \begin{pmatrix} \Delta \epsilon_y \\ \Delta \epsilon_z \end{pmatrix}$$
(29)

# 6.1.3 Multiple FOVs

The theoretical concept and practical implementation described in sections 6.1.1 and 6.1.2 work well if there is only "one" FOV. However Pandora has actually two separate FOVs for each spectrometer, one for direct sun observations (SunFOV) and one for sky observations (SkyFOV).



The FWHM of the SunFOV is about 2.5° and is determined by the diameter of the aperture on top of the collimator, the diameter of the aperture behind the diffuser, and the distance of these two apertures (see figure 11). The FWHM of the SkyFOV is about 1.5° and is determined by the diameter of the fiber and the distance between lens and fiber (see figure 11).

If the centers of the aperture on top of the collimator, of the aperture behind the diffuser, of the lens, and of the fiber are not perfectly aligned along a straight line, then the centers of the SkyFOV and the SunFOV will not be the same. Since they are also not necessarily the same for different spectrometers, we actually have four different pointing positions in the sky for Pandora-2S systems: a SunFOV and SkyFOV for spectrometer 1 and the same for spectrometer 2.

The tolerance for co-aligning the different FOVs in the manufacturing process is 0.3°. This means that the centers of the FOVs can vary by as much as this angle.

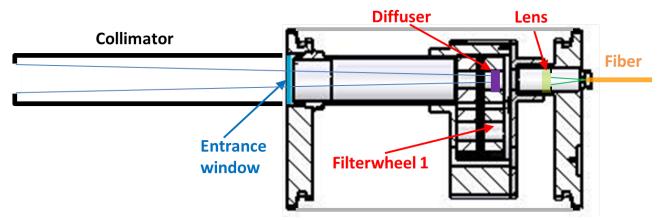


Figure 11: Schematics of the Pandora head sensor. The SunFOV is determined by the aperture on top of the collimator and the aperture behind the diffuser (thin blue lines). The SkyFOV is determined by the focal length of the lens and the diameter of the fiber (thin green lines).

This issue of different FOVs is solved in BlickO in the following way: first the best leveling is determined separately for each FOV using sun searches with different filter settings. Then an average SkyFOV is determined using the SkyFOVs from all spectrometers. Then we determine the angular difference for each individual SkyFOV from the average value. The pointing of the instrument is always done using the leveling to optimize the pointing of the average SkyFOV, but when the data are reported, each individual SkyFOV is used to calculate the pointing position. Here an example: if a Pandora-2S unit is commanded to point to a certain position in the sky, e.g. pointing zenith angle (PZA) 80.0° and pointing azimuth (PAZ) 180.0°, then the true pointing angles (and the numbers written in the raw data file) for spectrometer 1 and 2 will be slightly different, e.g. PZA=80.11°, PAZ=179.97° for spectrometer 1 and PZA=79.94°, PAZ=180.20° for spectrometer 2.



# 6.2 Signal to Noise Ratio Optimization

This section describes what technique is applied by BlickO to maximize the signal to noise ratio (SNR) for the Pandora measurements.

The first correction from L0 to L1 is the dark correction (see section 6.3.1), where an estimation for the dark counts (DC) is subtracted from the measured bright counts (BC). Pandora can measure DC by setting the filterwheels in a position to block the light input. The question of when and how often DC, i.e. measurements with no light input, should be measured with a photometer to obtain the highest SNR for the final L1 data is not trivial. We distinguish two situations:

- "Immediate dark measurements": immediately after the BC have been measured, DC measurements are taken at the same integration time and are used to correct the BC.
- "Dark map": at a more distant time from the BC measurements, many DC measurements over the full range of possible integration times and temperatures are made and analyzed to create a so-called "Dark map". This map either describes the full dark measurements or parts of it (e.g. the dark count fine structure, see below) and is used either by itself or in combination with immediate dark measurements to correct the BC.

The advantage of the immediate dark measurements is that the conditions (temperature, state of the instrument, etc.) are basically the same as for BC measurements. The disadvantage is that in order to optimize the SNR of the L1 data, the system needs to spend quite some time doing DC measurements, which could be better used to take more BC data.

For the dark mapping it is the opposite. One spends much less time in doing DC measurements (instead they are done at a time the instrument would be idle anyway), but runs the risk that the dark map is not entirely representative for the BC data, since the conditions (temperature, state of the instrument, etc.) might have changed between the time the dark map was created and the time of the BC measurements. Nevertheless this possible issue can be reduced by combining the dark map with some measured DC.

# 6.2.1 Theoretical Background

Since "true" noise is uncorrelated from pixel to pixel, all equations given in this section are for one pixel and photons from one wavelength only. The number of electrons accumulated in a pixel is given by

$$\begin{aligned} N_{e} &= N_{eT} + N_{e\gamma} \\ &= N_{eT} + N_{\gamma} \cdot QE \\ &= I_{eT} \cdot \Delta t + I_{\gamma} \cdot QE \cdot \Delta t \end{aligned} \tag{30}$$

 $\Delta t$  Exposure time (or integration time) [s]

 $N_e$  Total number of electrons accumulated within  $\Delta t$  [1]

 $N_{eT}$  Number of thermal electrons accumulated within  $\Delta t$  [1]

 $N_{e\gamma}$  Number of electrons produced by incident photons within  $\Delta t$  [1]

 $N_{\gamma}$  Number of incident photons within  $\Delta t$  [1]

QE Detector quantum efficiency for this wavelength, number of electrons per number of photons [1]

 $I_{eT}$  Mean rate of thermal electrons during  $\Delta t$ , electrons per second [1/s]

 $I_{\gamma}$  Mean rate of incident photons during  $\Delta t$ , photons per second [1/s]



The value in square brackets is a possible unit for each parameter. The amplifier in the read out electronics (ROE) converts the accumulated charge in the output voltage. Here we assume a linear response of the amplifier.

$$V = N_e \cdot q_e \cdot \frac{A}{C_0} + OFFSET_V$$

$$= N_e \cdot GAIN_V + OFFSET_V$$
(31)

V Output voltage [V]

q<sub>e</sub> Electron charge [C]

C<sub>0</sub> Output node capacitance [F=C/V]

A Amplifier voltage gain [1]

OFFSET<sub>V</sub> Electrical offset added by the amplifier [V]

GAIN<sub>V</sub> Voltage gain per electron [V]

The analog-digital converter converts the output voltage in digital counts. Here we assume a linear response of the AD-converter with zero offset (a non-zero offset would not change the principle).

$$C = V \cdot F$$

$$= N_e \cdot GAIN_V \cdot F + OFFSET_V \cdot F$$

$$= N_e \cdot GAIN + OFFSET$$
(32)

C Raw counts [DN]

F Analog-digital conversion factor [DN/V]

OFFSET (Count) offset (=OFFSETV·F) [DN]

GAIN (Count) gain (=GAINV·F) per electron [DN]

A single dark measurement DC<sub>i</sub> and a single bright measurement BC<sub>i</sub> are given by:

$$\begin{split} DC_i &= OFFSET_i + GAIN_i \cdot N_{eTi} \\ &= OFFSET_i + GAIN_i \cdot I_{eTi} \cdot \Delta t_i \\ BC_i &= OFFSET_i + GAIN_i \cdot [N_{eTi} + N_{e\gamma i}] \\ &= OFFSET_i + GAIN_i \cdot [I_{eTi} + I_{\gamma i} \cdot QE] \cdot \Delta t_i \end{split} \tag{33}$$

where i is the measurement index. In the following paragraphs each of the parameters in equation 33 is discussed separately:

#### $\Delta \mathbf{t}$

We assume the uncertainty in the exposure time  $\Delta t$  to be negligible and set the variance ( $\sigma^2$ ) to zero. There might be a systematic error (bias) in  $\Delta t$  (see section 6.3.5), but we do not assume this bias changing over time.

$$\sigma_{\Delta t}^2 = 0 \tag{34}$$



#### **OFFSET**

The offset is determined by the read out electronics, varies in general with temperature, may have a long-term drift due to changes in the electronics, and may even depend on  $N_e$ . Note that if it depends on  $N_e$ , then the parameter OFFSET<sub>i</sub> in equation 33 is different for DC and BC and a systematic error is introduced, if the measured dark counts are used to apply "dark correction" on the bright counts. This case is discussed in section 6.2.6. Here we limit ourselves to the statistical uncertainty of the offset over a short time period (<1 min). The (short-term) uncertainty in the measured offset is called read noise or electronic noise (see e.g.  $http://en.wikipedia.org/wiki/Electronic_noise$ ).

$$\sigma_{\text{OFFSET}}^2(\text{SHORT-TERM}) = \sigma_{\text{READ}}^2$$
 (35)

#### **GAIN**

The gain is determined by the read out electronics, may vary with temperature and may also have a long-term drift due to changes in the electronics. Over a short time period (<1 min) we consider the variation of GAIN negligible.

$$\sigma_{\text{GAIN}}^2(\text{SHORT-TERM}) = 0$$
 (36)

### **QE**

The quantum efficiency depends on the wavelength and may have a long-term drift due to changes in the detector. Over a short time period (<1 min) we consider the variation of QE negligible.

$$\sigma_{\text{OE}}^2(\text{SHORT-TERM}) = 0$$
 (37)

## $N_{eT}$

The number of thermal electrons is the product of  $I_{eT}$  times  $\Delta t$ . The (short-term) uncertainty in the measured number of thermal electrons follows a Poisson distribution (see e.g. http://www.photomet.com/pm\_solutions/library\_encyclopedia/library\_enc\_signal.php). The variance of a Poisson distributed variable with mean value N equals N (see e.g. http://en.wikipedia.org/wiki/Poisson\_distribution). For the (short-term) uncertainty of  $N_e$  follows

$$\sigma_{\text{NeT}}^2(\text{SHORT-TERM}) = N_{\text{eT}}$$
 (38)

 $\mathbf{I}_{\gamma}$ 

The mean rate of incident photons during  $\Delta t$  depends on the light source. For a "stable lamp" input over a short time period (<1 min) we may assume no drift and only statistical noise in  $I_{\gamma}$ .

$$\sigma_{\text{I}\gamma}^2(\text{LAMP,SHORT-TERM}) = N_{\text{I}\gamma\text{STAT}}$$
 (39)

For solar radiation the input is in general changing, even over a short time period, even for perfectly stable atmospheric conditions, since the solar angles are changing. With variable atmospheric conditions the rate of incident photons changes even more over the duration of the measurement time. Hence the variation of  $I_{\gamma}$  is not a purely statistical uncertainty. If several identical instruments would measure the solar radiation at the same time, they all would see the same systematic variation in  $I_{\gamma}$ .

$$\sigma_{\text{I}\gamma}^2(\text{SUN,SHORT-TERM}) = N_{\text{I}\gamma\text{STAT}} + N_{\text{I}\gamma\text{SYS}}$$
 (40)

 $N_{e\gamma}$ 

The number of electrons produced by incident photons is the product of  $I_{\gamma}$  times QE times  $\Delta t$ . The (short-term) uncertainty in the measured  $N_{e\gamma}$  also follows a Poisson distribution (see e.g. http://en.wikipedia.



org/wiki/Shot\_noise). The total (short-term) uncertainty of  $N_{e\gamma}$  is therefore a combination of the photon noise and the input rate variation.

$$\begin{split} \sigma_{\text{Ne}\gamma}^2(\text{SHORT-TERM}) &= N_{\text{e}\gamma} + \sigma_{\text{I}\gamma}^2 \cdot \text{QE}^2 \cdot \Delta t^2 \\ &= N_{\text{e}\gamma} + \sigma_{\text{eINPUT}}^2 \end{split} \tag{41}$$

 $\sigma_{\text{eINPUT}}$  is the uncertainty of the electrons produced by incident photons due to the input-rate variation. We may neglect  $\sigma_{\text{eINPUT}}$  for a short-term stable input such as a lamp on a very stable power supply.

From equations 31 to 41 we can compose the total short-term (<1 min) statistical uncertainties (noise) of BC<sub>i</sub> and DC<sub>i</sub> respectively:

$$\begin{split} \sigma_{\text{DCi}}^2 &= \sigma_{\text{READi}}^2 + \text{GAIN}_i^2 \cdot \text{N}_{\text{eTi}} \\ &= \sigma_{\text{READi}}^2 + \text{GAIN}_i^2 \cdot \text{I}_{\text{eTi}} \cdot \Delta t_i \\ \sigma_{\text{BCi}}^2 &= \sigma_{\text{READi}}^2 + \text{GAIN}_i^2 \cdot \left[ \text{N}_{\text{eTi}} + \text{N}_{\text{e}\gamma i} + \sigma_{\text{eINPUTi}}^2 \right] \\ &= \sigma_{\text{READi}}^2 + \text{GAIN}_i^2 \cdot \left[ \text{I}_{\text{eTi}} + \text{I}_{\gamma i} \cdot \text{QE}_i \right] \cdot \Delta t_i + \sigma_{\text{INPUTi}}^2 \end{split}$$

$$(42)$$

 $\sigma_{INPUT}$  is the product of  $\sigma_{eINPUT}$  times GAIN, i.e. the uncertainty due to input-rate variation expressed in counts. This equation is for single DC or BC measurements. In practice we average several detector readings to reduce the noise. For the next equation we assume having  $n_{BC}$  repetitions of BC-measurements BC<sub>i</sub> and  $n_{DC}$  repetitions of DC-measurements DC<sub>i</sub>, all at the same integration time  $\Delta t$ . The total time of bright measurements  $n_{BC} \cdot \Delta t$  or dark measurements  $n_{DC} \cdot \Delta t$  is less than 1 min, so all equations from the theory for short time periods are valid.

The average over the DC- and BC-measurements is given by

$$\begin{split} DC &= \frac{1}{n_{DC}} \cdot \sum_{i=1}^{n_{DC}} DC_i = OFFSET + GAIN \cdot I_{eT} \cdot \Delta t \\ BC &= \frac{1}{n_{BC}} \cdot \sum_{i=1}^{n_{BC}} BC_i = OFFSET + GAIN \cdot [I_{eT} + I_{\gamma} \cdot QE] \cdot \Delta t \end{split} \tag{43}$$

We have removed index i from all parameters, since the integration time  $\Delta t$  is the same for all scans, OFFSET, GAIN and QE have presumably not changed, and  $I_{eT}$  and  $I_{\gamma}$  are defined as the average values over the  $I_{eTi}$  and  $I_{\gamma i}$ . We can calculate the standard deviation over the DC-measurements as best estimation for  $\sigma_{DCi}$ :

$$\sigma_{\text{DCi}}^2 = \frac{1}{n_{\text{DC}} - 1} \cdot \sum_{i=1}^{n_{\text{DC}}} (\text{DC}_i - \text{DC})$$
 (44)

If we assume an input without drift (e.g. lamp measurements over a short time period), we can calculate the standard deviation over the BC-measurements as best estimation for  $\sigma_{BCi}$  (first line of equation 45). If we assume an input changing linearly in time (e.g. solar radiation over a short time period at perfectly stable atmospheric conditions) we calculate the rms to a fitted straight line through the BC-measurements as best estimation for  $\sigma_{BCi}$  (second line of equation 45). In the latter case the points on the fitted straight line are called BC<sub>iFIT</sub>.



$$\sigma_{BCi}^{2} = \frac{1}{n_{BC} - 1} \cdot \sum_{i=1}^{n_{BC}} (BC_{i} - BC)$$

$$\sigma_{BCi}^{2} = \frac{1}{n_{BC} - 2} \cdot \sum_{i=1}^{n_{BC}} (BC_{i} - BC_{iFIT})$$
(45)

From equations 42 and 43 we can derive the noise in the averaged dark measurements  $\sigma_{DC}$ :

$$\sigma_{\rm DC}^2 = \frac{\sigma_{\rm DCi}^2}{n_{\rm DC}} = \frac{1}{n_{\rm DC}} \cdot \left[ \sigma_{\rm READ}^2 + {\rm GAIN}^2 \cdot {\rm I}_{\rm eT} \cdot \Delta t \right]$$
 (46)

Here we apply that for the standard uncertainty of the mean value (also called standard error), the statistical components in the uncertainties are reduced by the square root of the number of measurements (see e.g. http://en.wikipedia.org/wiki/Standard\_error).

For the uncertainty of the averaged bright measurements  $\sigma_{BC}$  we use the same equations and distinguish two cases. The first line in equation 47 is for the case that  $\sigma_{INPUT}$  is purely statistical and the variance scales with  $1/n_{BC}$  (e.g. a lamp with a white noise flicker). The second line in equation 47 is for the case that  $\sigma_{INPUT}$  is not purely statistical and includes a systematic component, which means it does not scale with  $1/n_{BC}$  (e.g. an atmospheric change while the measurement is taken).

$$\sigma_{BC}^{2}(STAT) = \frac{1}{n_{BC}} \cdot \left[ \sigma_{READ}^{2} + GAIN^{2} \cdot (I_{eT} + I_{\gamma} \cdot QE) \cdot \Delta t + \sigma_{INPUT}^{2} \right] = \frac{\sigma_{BCi}^{2}}{n_{BC}}$$

$$\sigma_{BC}^{2}(SYS) = \frac{1}{n_{BC}} \cdot \left[ \sigma_{READ}^{2} + GAIN^{2} \cdot (I_{eT} + I_{\gamma} \cdot QE) \cdot \Delta t \right] + \sigma_{INPUT}^{2} > \frac{\sigma_{BCi}^{2}}{n_{BC}}$$

$$(47)$$

# 6.2.2 Single Scan Dark Correction

Here we look at the case, where exactly one bright and one dark measurement is given. For the dark corrected counts CC<sub>i</sub> we obtain from equation 33:

$$CC_{i} = BC_{i} - DC_{i} = GAIN \cdot I_{\gamma i} \cdot QE \cdot \Delta t$$
(48)

We assume that  $DC_i$  and  $BC_i$  were measured at the same temperature and exposure time and that all parameters such as OFFSET, GAIN, QE, etc. have not changed. Therefore, we write for the uncertainty of  $CC_i$  based on equation 42:

$$\begin{split} \sigma_{\text{CCi}}^2 &= \sigma_{\text{DCi}}^2 + \sigma_{\text{BCi}}^2 \\ &= 2 \cdot \sigma_{\text{READ}}^2 + \text{GAIN}^2 \cdot [2 \cdot I_{\text{eT}} + I_{\gamma} \cdot \text{QE}] \cdot \Delta t + \sigma_{\text{INPUT}}^2 \\ &= 2 \cdot \sigma_{\text{READ}}^2 + 2 \cdot \text{GAIN}^2 \cdot I_{\text{eT}} \cdot \Delta t + \text{GAIN} \cdot \text{CC}_i + \sigma_{\text{INPUT}}^2 \\ &= 2 \cdot \sigma_{\text{DCi}}^2 + \text{GAIN} \cdot \text{CC}_i + \sigma_{\text{INPUT}}^2 \\ &= \sigma_{\text{INSTR}}^2 + \sigma_{\text{INPUT}}^2 \end{split} \tag{49}$$



In this equation we have split the uncertainty of the corrected counts in the "instrumental noise"  $\sigma_{INSTR}$  and the "input variation"  $\sigma_{INPUT}$ . If we assume no input variation ( $\sigma_{INPUT}$ =0) and furthermore neglect  $I_{eT}$ , which can be done for low exposure times and/or low temperatures, then equation 49 reduces to:

$$\sigma_{\text{CCi}}^2 = 2 \cdot \sigma_{\text{READ}}^2 + \text{GAIN} \cdot \text{CC}_{\text{i}}$$
 (50)

Figure 12 shows the estimated  $\sigma_{CCi}$  from equation 50 for two cases: the "CMOS-system" is characterized by low gain and high read noise (red lines) and is typical for CMOS detectors as used in the first Pandora units, which could only measure in direct sun mode. The "CCD-system" is characterized by high gain and low read noise (blue lines) and is typical for CCD detectors as used in the newer Pandora systems. Note that currently all Pandora systems use a 16-bit AD converter, i.e. the nominal saturation limit is  $2^{16}$ -1=65535 counts, which means a read noise of 20 (40) counts corresponds to 0.03% (0.06%) of saturation.

The solid lines in the figure are the total noise (combination of read noise and photon noise), the dashed lines represent the photons noise only. The total noise is driven by the read noise for low counts and by the photon noise for high counts.

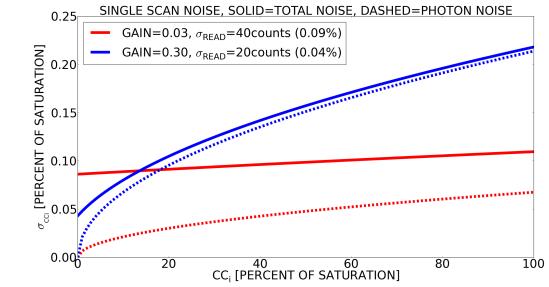


Figure 12: Noise of dark corrected counts for a single bright and dark measurement as a function of the corrected counts, expressed as a fraction of saturation.

The SNR of CC<sub>i</sub> based on equation 50 is given by:

$$SNR_{i} = \frac{CC_{i}}{\sigma_{CCi}} = \frac{CC_{i}}{\sqrt{2 \cdot \sigma_{READ}^{2} + GAIN \cdot CC_{i}}}$$
 (51)

This is shown in figure 13. At 80% of saturation (black dashed vertical line), the SNR for the CMOS-system (red) is about 760:1 and for the CCD-system (blue) about 410:1.



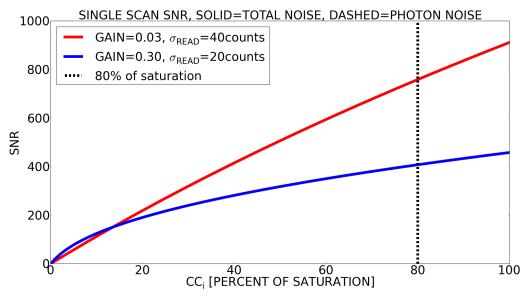


Figure 13: SNR of dark corrected counts for a single bright and dark measurement as a function of the corrected counts, expressed as a fraction of saturation.

## 6.2.3 Multiple Scans Dark Correction

Here we use the same assumptions as in the previous section, but allow multiple measurements of DC<sub>i</sub> and BC<sub>i</sub>. From equation 43 we get for the corrected counts CC:

$$CC = BC - DC = GAIN \cdot I_{\gamma} \cdot QE \cdot \Delta t$$
 (52)

For the uncertainty of CC follows from equations 46 and 47:

$$\begin{split} \sigma_{\text{CC}}^2 &= \sigma_{\text{DC}}^2 + \sigma_{\text{BC}}^2 \\ &= \left(\frac{1}{n_{\text{DC}}} + \frac{1}{n_{\text{BC}}}\right) \cdot \left[\sigma_{\text{READ}}^2 + \text{GAIN}^2 \cdot I_{\text{eT}} \cdot \Delta t\right] + \frac{1}{n_{\text{BC}}} \cdot \left[\text{GAIN}^2 \cdot I_{\gamma} \cdot \text{QE} \cdot \Delta t\right] \\ &= \left(\frac{1}{n_{\text{DC}}} + \frac{1}{n_{\text{BC}}}\right) \cdot \sigma_{\text{DCi}}^2 + \frac{1}{n_{\text{BC}}} \cdot \text{GAIN} \cdot \text{CC} \\ &= \sigma_{\text{INSTR}}^2 \end{split} \tag{53}$$

Here we assume no input variation ( $\sigma_{\text{INPUT}}$ =0), in which case the uncertainty of CC is the instrumental noise only.

We can ask ourselves the question, what the best split between the number of bright and dark measurements is to get the highest SNR in the corrected counts. We can rewrite equation 53 using the "Dark fraction"  $f_{DC}$ , which is the number of dark count repetitions relative to the total number of repetitions (so e.g.  $f_{DC}$ =0.5 means the same number of repetitions for bright and dark counts).

$$\sigma_{\text{INSTR}}^2 = \frac{\sigma_{\text{DCi}}^2}{\mathbf{n} \cdot \mathbf{f}_{\text{DC}}} + \frac{\sigma_{\text{DCi}}^2 + \text{GAIN} \cdot \text{CC}}{\mathbf{n} \cdot (1 - \mathbf{f}_{\text{DC}})}$$
(54)

n is the sum of  $n_{DC}$  and  $n_{BC}$ . Deriving equation 54 for  $f_{DC}$  yields:



$$\frac{\partial \sigma_{\text{INSTR}}^2}{\partial f_{\text{DC}}} = \frac{-\sigma_{\text{DCi}}^2}{n \cdot f_{\text{DC}}^2} + \frac{\sigma_{\text{DCi}}^2 + \text{GAIN} \cdot \text{CC}}{n \cdot (1 - f_{\text{DC}})^2}$$
(55)

For the extreme conditions, we obtain this solution:

$$f_{DCOPT} = \frac{-\sigma_{DCi}^2}{GAIN \cdot CC} \pm \sqrt{\left(\frac{\sigma_{DCi}^2}{GAIN \cdot CC}\right)^2 + \frac{\sigma_{DCi}^2}{GAIN \cdot CC}}$$
(56)

f<sub>DCOPT</sub> must be positive and we call it the "Optimized dark ratio" ODR:

$$f_{DCOPT} = ODR = \sqrt{q^2 - q} - q \tag{57}$$

where q is given by

$$q = \frac{\sigma_{DCi}^2}{GAIN \cdot CC} \ge \frac{\sigma_{READ}^2}{GAIN \cdot CC}$$
 (58)

If we neglect  $I_{eT}$ , which can be done for low exposure times and/or low temperatures, we can replace  $\sigma_{CCi}$  by  $\sigma_{READ}$ . ODR as a function of q is shown in figure 14.

For high values of q (low gain and high read noise systems), ODR approaches 0.5, i.e. to optimize the SNR, one needs to measure bright and dark counts for nearly the same duration. For a typical CMOS-system, the ODR ranges from 0.41 at 80% of saturation to 0.47 at 20% of saturation (red line in figure).

For low values of q (high gain and low read noise systems), ODR approaches 0, i.e. to optimize the SNR one needs to dedicate much more measurement time for BC than for DC. For a typical CCD-system, the ODR ranges from 0.14 at 80% of saturation to 0.23 at 20% of saturation (blue line in figure).

For the IOF-entry "Optimized dark ratio" (see section 5.3) we use the ODR at 50% of saturation, which would be 0.44 for the typical CMOS-system and 0.16 for the typical CCD-system (red and blue dots in figure respectively). Hence for "Immediate dark measurements", the best split between BC and DC is given by the ODR.

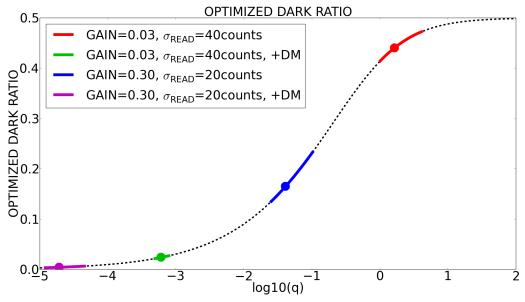


Figure 14: Optimized dark ratio as a function of q from equation 58 or 60 for q-values ranging from 1e-5 to 1e2. The colored lines are for CC-values ranging from 20% to 80% of saturation for typical CMOS- and CCD-systems respectively. The colored dots are for CC-values at 50% of saturation. Red and blue data are for the case of not using a dark map, while green and magenta data assume the use of a dark map.



### 6.2.4 Dark Map

If instead or in addition to the immediate dark measurements, a dark map is used, the noise in the dark estimation is reduced. As described in details in section 6.3.1 we are not creating a full dark map at present. Instead the dark count is split into a "DC background" and "DC fine structure".

Since the background consists of a few fitting parameters only and we assume no noise in the dark fine structure parameters, the noise in DC<sub>E</sub> is reduced to approximately  $\sigma_{DCi}$  divided by square root of the number of pixels  $\sqrt{\text{npix}}$ . Including this reduction, equation 54 becomes

$$\sigma_{\text{INSTR}}^2 = \frac{\sigma_{\text{DCi}}^2}{\mathbf{n} \cdot \mathbf{n}_{\text{PIX}} \cdot \mathbf{f}_{\text{DC}}} + \frac{\sigma_{\text{DCi}}^2 + \text{GAIN} \cdot \text{CC}}{\mathbf{n} \cdot (1 - \mathbf{f}_{\text{DC}})}$$
(59)

and consequently q changes to

$$q = \frac{\sigma_{DCi}^2}{\sigma_{DCi}^2 \cdot [n_{PIX} - 1] + n_{PIX} \cdot GAIN \cdot CC}$$
(60)

Using the dark map the values of q are much smaller causing ODR to drop significantly to 0.03 for low gain and even <0.01 for high gain (see figure 14).

Figure 15 compares the immediate dark method and the dark mapping method. For this example we have used a total number of scans of n=1000. The solid lines show the dark mapping method, where most of the 1000 scans are spent taking BC data. The dashed lines refer to the immediate dark method, where the number of dark measurements is much larger. The improvement of the dark mapping method over the immediate dark method is especially pronounced for the CMOS-case.

Based on this figure we can say that if we believe that the dark map represents all instrumental parameter such as OFFSET, GAIN and QE accurately enough, it is the preferred method for the dark correction.

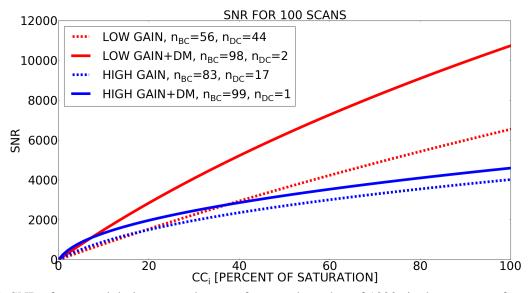


Figure 15: SNR of averaged dark corrected counts for a total number of 1000 single scans as a function of the corrected counts, expressed as fraction of saturation. Solid lines show dark mapping method, where all 1000 scans are spent taking BC data. Dashed lines refer to the immediate dark method, where the 1000 scans are split between DC and BC measurements.



### 6.2.5 Optimized SNR

The optimized SNR as a function of the number of scans is shown in figure 16, where we assume corrected counts at 50% of saturation. For the retrieval of weaker absorbers in the atmosphere using the spectral fitting technique, we consider a SNR of 10000:1 as the desired minimum (black dashed line) to avoid being noise-limited in the retrieval. In order to exceed this limit, we need at least 250 scans for CMOS-systems and 960 scans for CCD-systems. These scan-numbers can be obtained by Pandora for direct sun observations, except for measurements near twilight or with clouds in the solar direction. For sky radiance measurements (only possible with CCD-systems), the integration times are usually 20 to 500 ms (depending on the wavelength range, solar angles, etc.). This means that we would have to acquire spectra for 20 s to 500 s in order to not be noise-limited. While 20 s works from a practical point of view, 500 s is definitely too long and Pandora data are noise limited for some applications in this case.

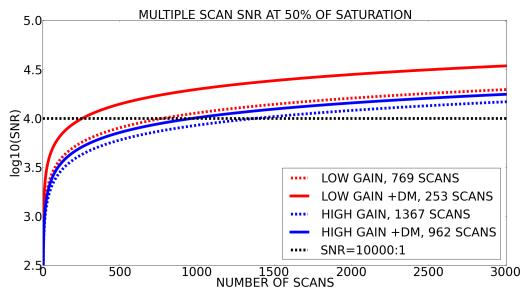


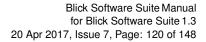
Figure 16: SNR of averaged dark corrected counts at 50% of saturation as a function of the number of scans averaged with (solid) and without (dashed) using a dark map. The black dashed line is for SNR=10000:1. Values listed in the legend give the number of scans needed to exceed an SNR of 10000:1.

### 6.2.6 Blind Pixels

This section is not directly related to the SNR, but discusses an important systematic issue in the dark correction. All previous sections were based on the assumption that the dark offset does not depend on  $N_e$ , i.e. the parameter OFFSET in equation 33 is the same for DC and BC. However, we noticed that this is in general not the case for most Pandora systems. Instead, we observed that OFFSET is reduced when  $N_e$  increases, i.e. BC measurements have systematically a smaller OFFSET than DC measurements.

We were not able to determine the magnitude of this effect in the laboratory in a consistent way, which means we could not develop a technique to correct for it with some mathematical algorithm. We could only determine that the OFFSET bias increases with the total photon energy falling on the detector (or the sum over the  $N_e$  for all pixels) and can be >50 counts (for a 16-bit unit) at high signal levels.

This problem affects both the dark mapping and the immediate dark method. Fortunately it is a small effect that has negligible significance in regular field operation. However, it is a large problem in the characterization of the instrument's spectral stray light. For the stray light characterization, the instrument is illuminated by a laser line and the distribution of the signal over the whole detector is analyzed. We know that the pixels in the far field (>10 nm away from the maximum signal at the laser wavelength) have signals relative to the signal at





the maximum in the order of <3e-5 for "good units" (with low amount of stray light) and >1e-4 for "bad units" (with high amount of stray light). Hence, if the maximum is at 60000 counts, then the far field pixels have <2 counts for good units and >6 counts for bad units. Since the OFFSET bias is in this order of magnitude or even larger, the determination of the stray light level is not possible.

It turns out that most detectors have a feature, which can be used to reduce this problem, the "Blind pixels". Blind pixels are pixels that do not receive any light input even if the detector is illuminated. All newer Pandoras with CCD-detectors have at least 6 such blind pixels, of which at least 3 are read during the data acquisition. The blind pixels allow to estimate OFFSET for the lowest pixels (those adjacent to the blind pixels) the during the BC measurements, which is a big help in the determination of the instrument's stray light in the laboratory.

The blind pixels are included in the dark correction in a way that the average of them is subtracted from both the BC and DC data before the corrected counts CC are build (see also section 6.3.1). This correction assumes that systematic OFFSET change is a constant value, affecting all pixels in the same way. We know that this is not the case, but applying this correction is better than not applying it.



# 6.3 L1 Algorithm - Data Correction

A total of 10 processing steps are applied to the Pandora L0 data to produce L1 data.

- 1. Dark correction
- 2. Non-linearity correction
- 3. Latency correction
- 4. Flat field correction
- 5. Conversion to count rates
- 6. Temperature correction
- 7. Stray light correction
- 8. Filter correction
- 9. Conversion to (ir)radiances
- 10. Wavelength correction

Each step is described in the following sections. The mathematical equations describe how L0 data at pixel i,  $L0_i$ , are converted into L1 data at pixel i,  $L1_i$ . By default BlickP applies all correction steps to the L0 data, but it also allows to only apply a subgroup of correction steps for research purposes. Note that the order of the correction steps cannot be changed since they are not commutative. If no correction steps are applied, then the L1 data equal the L0 data.

$$L1_i = L0_i \tag{61}$$

We will use this equation as a starting point for all equations to come, which means  $L1_i$  "evolves" through the correction steps. In order to make the equations more readable, we will call the L1 data, which have been processed for all previous correction steps,  $L1_i^*$  (at the right side of the equation) and the L1 data with the new correction step added  $L1_i$  (at the left side of the equation).



### 6.3.1 Dark Correction

The theory of the dark count is described in section 6.2. It is the sum of the dark offset (given by an electronic bias of the ROE) and the dark slope (charge produced by thermal electrons). The Pandora dark count depends on the detector it is using. Most instruments have a dark count behavior as shown in figure 17. The dark offset is about 1-2 % (of the saturation value) and the dark slope is about 1-2 % per second at the current operational temperature. For our standard 16-bit AD converter, this corresponds to about 1000 counts dark offset and 1000 counts/s dark slope at a detector temperature of about 25°C.

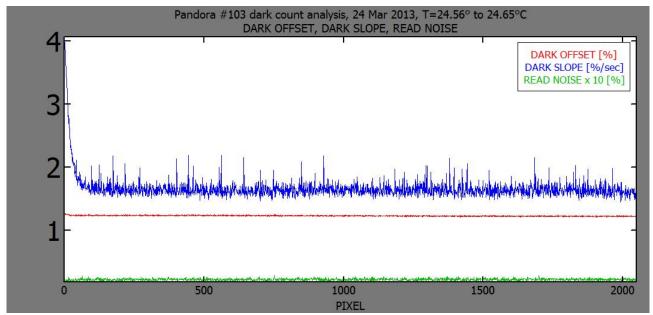


Figure 17: Dark offset (red), dark slope (blue), and read noise (green) of Pandora 103 as a function of pixel. Data are for a set temperature of 20 °C. The electronics board temperature is about 25 °C.

As explained in section 6.2, there are two methods to estimate the dark count. For the "Immediate dark method", only the dark count measured immediately after the regular measurements (with light input) is used to estimate the dark count included in the regular data. In that case the estimated dark count at pixel i,  $DC_{Ei}$ , is identical to the immediate dark measurements  $DC_i$ :

$$DC_{Ei} = DC_i (62)$$

For the "Dark map method", the immediate dark measurements are used in combination with the "dark fine structure map", which has been determined during calibration. In that case the dark count is estimated as follows:

First the dark count fine structure  $DC_{FINEi}$  is calculated:

$$DC_{FINEi} = A_{ICi} \cdot \exp^{B_{ICi} \cdot (T_{RE} - T_{RR})} + A_{SLi} \cdot \exp^{B_{SLi} \cdot (T_{RE} - T_{RR})} \cdot IT$$
(63)

 $T_{RE}$  is the radiometric effective temperature during the bright measurement,  $T_{RR}$  is the dark count reference temperature, IT is the integration time, and  $A_{ICi}$ ,  $B_{ICi}$ ,  $A_{SLi}$ , and  $B_{SLi}$  are the so-called A- and B-parameters for the dark fine structure intercept and slope respectively. All those parameters are determined during calibration.

Then the dark count fine structure is subtracted from the immediate dark measurements and the dark background  $DC_{BGi}$  is obtained by fitting the "dark background function" in the difference  $DC_i$  minus  $DC_{FINEi}$ . The dark background function is the sum of an exponential function and an Nth order polynomial as shown in equation 64:



$$DC_{BGi} = E_0 \cdot \exp^{-E_1 \cdot pixind} + C_0 + C_1 \cdot pixind + \dots + C_{N+1} \cdot pixind^N$$
(64)

pixind are the indices of the regular pixels divided by 1000, i.e. for the first regular pixel pixind=0, for the second pixind=0.001, etc.  $E_0$  and  $E_1$  are the parameters of the exponential part and  $C_0$ ,  $C_1$  are the coefficients of the polynomial. By adding the exponential term, the function is able to follow the "hockey stick" in the dark count (see figure 17). Note that the fitting includes the uncertainty in the measured dark count.

The dark count fine structure is added to the fitted dark background to obtain the estimated dark count  $DC_{Ei}$ :

$$DC_{Ei} = DC_{RGi} + DC_{FINEi}$$
(65)

If the detector has blind pixels, they can also be included in the dark correction by subtracting their mean value from both the regular measurements and the dark measurements. The  $L1_i$  after dark correction are given by:

$$L1_{i} = \left(L1_{i}^{*} - \frac{1}{n_{\text{BLIND}}} \cdot \sum_{j=1}^{n_{\text{BLIND}}} L1_{j}^{*}\right) - \left(DC_{Ei} - \frac{1}{n_{\text{BLIND}}} \cdot \sum_{j=1}^{n_{\text{BLIND}}} DC_{j}\right)$$

$$(66)$$

 $DC_{Ei}$  is from equation 62 or 65 and  $n_{BLIND}$  is the number of blind pixels in the detector (see section 6.2.6). The ' $n_{BLIND}$ -terms' in equation 66 are not included if the detector has no blind pixels or if no 'blind correction' is requested in the data processing.



## 6.3.2 Non-Linearity Correction

Image sensors are in general not linear, i.e. they do not return a doubled signal when they are illuminated by the double the amount of light. The non-linearity of Pandora is typically a few percent for not too low, which can be characterized in the laboratory (figure 18), and is not dependent on the temperature. A typical non-linearity curve is shown in figure 18. One can observe that for very low counts, the non-linearity leaves the range of a few percent correction, i.e. the instrument returns significantly more signal than it should if it was linear. To fit the non-linearity the Blick software uses an equation of this type:

$$NLC_{i} = E_{0} \cdot \exp^{-E_{1} \cdot (L1_{i}^{*})^{E_{2}}} + C_{0} + C_{1} \cdot L1_{i}^{*} + \dots + C_{N+1} \cdot (L1_{i}^{*})^{N}$$
(67)

The order N of the polynomial depends on the detector of the specific Pandora unit, but in general a first or second order polynomial is sufficient. To correct for the non-linearity, the measured counts are divided by the non-linearity. The  $L1_i$  after non-linearity correction are given by:

$$L1_i = \frac{L1_i^*}{NLC_i} \tag{68}$$

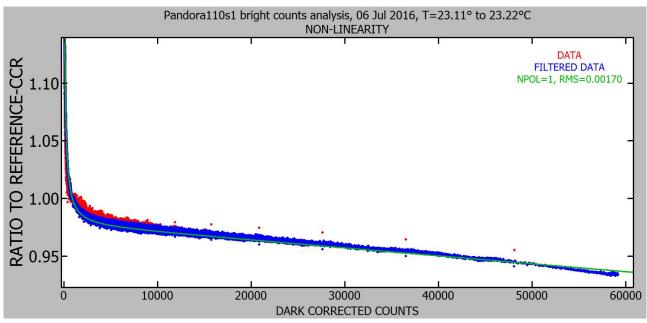


Figure 18: Non-linearity effect for Pandora 110. Data are in red, the data passing the filter criteria in blue, and the fit in the blue data is in green.

# 6.3.3 Latency Correction

Latency effects in the ROE of CCD detectors can cause the readings in a pixel to be influenced by the readings in the previously read pixels. E.g. if there are many subsequent high readings followed by very low readings, then the first low readings are biased high, since a residual charge from the previous readings is still in the ROE capacitor. This is shown in figure 19 for measurements with Pandora 106. For this test the detector was removed from the spectrometer, about a quarter of the pixels on each side were covered and data with only the central pixels illuminated taken (blue line in figure). Then the detector was 'reversed' (rotated by 180°) keeping the illumination constant and the measurements repeated (red line). Hence the only difference between the blue and red line in the figure is that the CCD is read from left to right in one case (blue line) and from right to left in the other case (red line). One can see that for the first pixels read after the high signal



region the values are biased high, therefore blue line exceeds the red line around pixels 1500 and higher, while the red line exceeds the blue line around pixels 550 and lower.

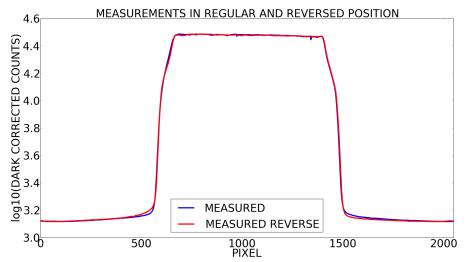


Figure 19: Logarithm of dark corrected counts as a function of the pixel number for measurements, where only the central part of the detector (pixels 600 to 1500) was illuminated and the left and right side blocked. Blue line is for the detector in regular orientation. Red line shows the same illumination on the reversed detector. Reading direction is left to right for the regular orientation (blue line) and right to left for the reversed orientation (red line).

The  $L1_i$  after latency correction are given by:

$$L1_i = L1_i^* - \Delta LAT_i \tag{69}$$

 $\Delta$ LAT<sub>i</sub> is the latency correction vector. If is determined in a recursive way as shown in equation 70.

$$\Delta LAT_1 = 0$$

$$\Delta LAT_{i+1} = \Delta LAT_i \cdot (1 - c_{DECAY}) + L1_i^* \cdot c_{GAIN}$$
(70)

 $c_{DECAY}$  and  $c_{GAIN}$  are the latency decay and latency gain constants respectively. i=1 is the first pixel to be read by the ROE, i=2 the seconds one, etc. For Pandora 106, these constants were determined to  $c_{DECAY}=6.3e-3$  and  $c_{GAIN}=1.8e-5$ . Figure 20 is a zoom of figure 19 in linear scale. The black line are the latency corrected counts.



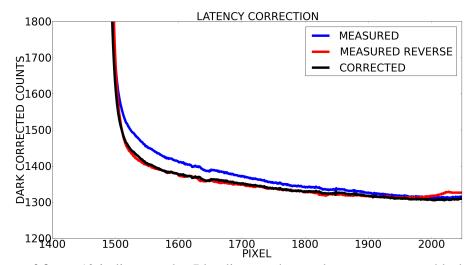


Figure 20: Zoom of figure 19 in linear scale. Blue line are the regular measurements, black line the regular measurements with the latency correction applied.

While the latency correction algorithm is incorporated in the L1 algorithm, it is currently not applied to most instruments, since the actual technique to determine the latency constants  $c_{DECAY}$  and  $c_{GAIN}$  is extremely risky, difficult and time-consuming as the detector has to be removed from the spectrometer.

### 6.3.4 Flat Field Correction

Even when each pixel is illuminated by the same amount of light, they all return slightly different signals. This is called "Pixel Response Non Uniformity", PRNU, and is caused by physical differences in the pixels. The PRNU is determined during the radiometric calibration and is expressed in percent (figure 21). Note that if the lamp signal is not smooth as a function of wavelength, lamp features might falsely be interpreted as PRNU. Therefore only selected lamps can be used for the radiometric calibration.

The PRNU is typically on the order of  $\pm 1$  %. Some detectors used in Pandora are 2D image arrays, where N vertical pixels are read in binned mode. This means that the returned value is in reality the sum over N pixels and therefore the measured PRNU is the "true" PRNU divided by  $\sqrt{N}$ . E.g. Pandora 39 uses a N=16 detector, reducing the measured PRNU from about  $\pm 1$  % to  $\pm 0.25$  % (figure 21). During radiometric calibration we only determine the PRNU for the binned pixels.

The  $L1_i$  after flat field correction are given by:

$$L1_i = \frac{L1_i^*}{PRNU_i} \tag{71}$$

 $PRNU_i$  is the PRNU for pixel i.



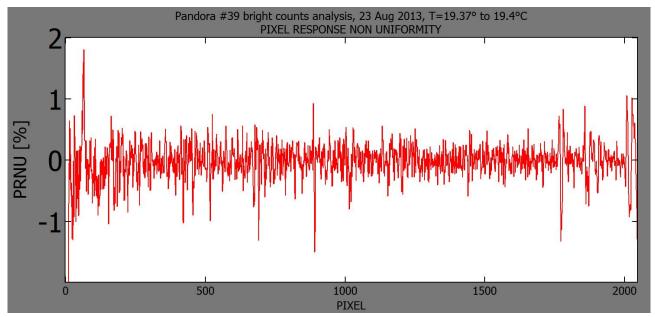


Figure 21: PRNU for Pandora 39 as a function of pixel. Note that the detector for this unit is binning 16 pixels at each scan, which reduces the true PRNU by a factor of 4.

### 6.3.5 Conversion to Count Rates

In this correction step the data are divided by the integration time, which changes their unit from "counts" or "DN" to "counts/s" or "DN/s". For earlier generation Pandoras we noticed that the nominal integration time (the one requested by the user) and the effective integration time (the "true" number) often had a small bias relative to each other. This bias causes very irregular features in the final figure of the linearity calibration (figure 18 in section 6.3.2) and can therefore be detected during calibration. Newer Pandoras show no difference between the nominal and effective integration time.

The  $L1_i$  after the conversion to count rates are given by:

$$L1_i = \frac{L1_i^*}{t - \Delta t} \tag{72}$$

t is the nominal integration time and  $\Delta t$  difference between the nominal and effective integration time. Both are given in seconds resulting a unit of "counts/s" for the L1<sub>i</sub> after this step.

# 6.3.6 Temperature Correction

In order to determine the temperature sensitivity of Pandora, the radiometric calibration is performed in the laboratory at three different spectrometer temperatures. Our experience has shown that a Pandora in good condition does only have very small radiometric temperature sensitivity, which means no temperature correction is applied to the data. In the case a signal change as a function of temperature is noted, the instrument will give false calibration results in the other tests as well, thus leading to an investigation of a hardware problem, e.g. moisture inside the optical bench.

The  $C_{1i}$  after temperature correction are given by:

$$L1_i = \frac{L1_i^*}{TC_i} \tag{73}$$

 $TC_i$  is the temperature correction polynomial evaluated at pixel i.



## 6.3.7 Stray Light Correction

The Pandora data processing software is set up to apply a so-called matrix stray light correction method to the spectra, which is based on measured slit functions [38]. Note that the determination of the slit function outside the region near the target pixel was not possible until spring of 2015 due to a problem with the dark offset (for more details see section 6.2.6). In these cases, BlickP applies a simple stray light correction on the data, which consists of subtracting the average signal below 290 nm from the spectra. The logic being that no solar light reaches the Earth's surface in these wavelengths, thus we can attribute such residual signals to stray light.

When using the simple stray light correction, the  $L1_i$  are given by:

$$L1_i = L1_i^* - MEAN(L1_i^* \quad for \quad \lambda_i < 290nm) \tag{74}$$

For those units with proper stray light characterization, the matrix method is used:

$$\overrightarrow{L1} = \mathbf{M} \cdot \overrightarrow{L1}^* \tag{75}$$

 $\overrightarrow{L1}$  is the (npix,1) vector of the L1 data,  $\mathbf{M}$  is the (npix,npix) stray light correction matrix. npix is the number of pixels on the detector and the dot stands for a matrix multiplication this case.

## 6.3.8 Sensitivity correction

In this step the spectra are converted to "what would have been measured if open hole had been in both filter wheels". The data are divided by the filter transmission, which is determined in the laboratory. Figure 22 shows measured transmissions for the filters in filterwheel 1 of Pandora 103. Note that the filter transmission determined in the laboratory is also affected by stray light. Therefore the transmission values in the short UV are biased high. This can only be improved with a better stray light correction.

The  $L1_i$  after the filter correction are given by:

$$L1_i = \frac{L1_i^*}{TRM_i} \tag{76}$$

 $TRM_i$  is the filter transmission for pixel i.



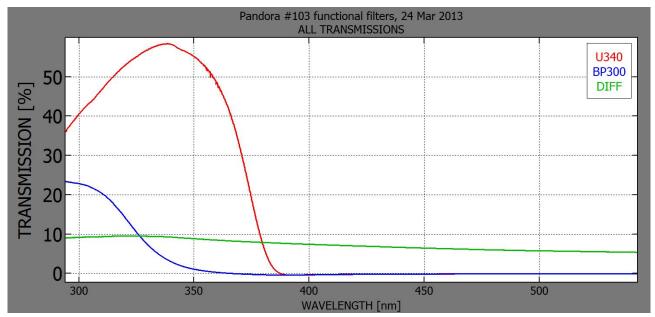


Figure 22: Measured transmissions for filters in Pandora 103 as a function for wavelength. U340 is a colored glass bandpass filter with peak transmission at 340 nm (red line). BP300 is an interference bandpass filter with peak transmission at 300 nm. DIFF is a ground quartz diffuser. Note that the fact that the blue line is increasing to the left of 300 nm is not due to a peak transmission of the BP300 filter shifted to the left of 300 nm. This is caused by the instrument's stray light. That means the measured transmission values in the region around 300 nm are overestimated, in the case only the simple stray light correction method is applied (see section 6.3.7).

# 6.3.9 Conversion to (Ir)radiances

This correction step is not operational at present, as we are still working on the details of the absolute calibration procedure. The calibration technique we plan to apply is similar to the one described in *Kazadzis et al.* [14]. The instrument is illuminated by an absolutely calibrated 1000 W FEL lamp from several distances that underfill Pandora's FOV (i.e. they are far enough away that the FOV does not clip the lamp).

The shape of the absolute sensitivity of Pandora is basically a combination of all optical elements involved, mainly the fiber transmission, grating efficiency and quantum efficiency of the detector. The  $L1_i$  after the conversion to (ir)radiances are given by:

$$L1_i = \frac{L1_i^*}{K_{ABS_i}} \tag{77}$$

 $K_{ABSi}$  is the absolute sensitivity (or calibration function) of the instrument at pixel i. There are actually two different sensitivity functions for sky and sun observations respectively. For sky observations, the function is in units of  $\left[\frac{\text{counts/s}}{\text{mW/m}^2/\text{nm/sr}}\right]$ , while for direct sun data it is in  $\left[\frac{\text{counts/s}}{\text{mW/m}^2/\text{nm}}\right]$ . Therefore the  $L1_i$  after this step are in units of  $\left[\text{mW/m}^2/\text{nm/sr}\right]$  or  $\left[\text{mW/m}^2/\text{nm}\right]$ .

Figure 23 shows the absolute spectral sensitivity of Pandora 30 for direct sun setting, i.e. with a diffuser in the filterwheels. It reaches at  $16 \text{ mW/m}^2/\text{nm}$  at 430 nm. Different lines refer to different alignment methods and different calibration lamps used in the laboratory setup. We still notice differences at the order of  $\pm 5\%$  in the results based on the exact laboratory setup. For this reason we still investigate how we can improve the absolute calibration technique.



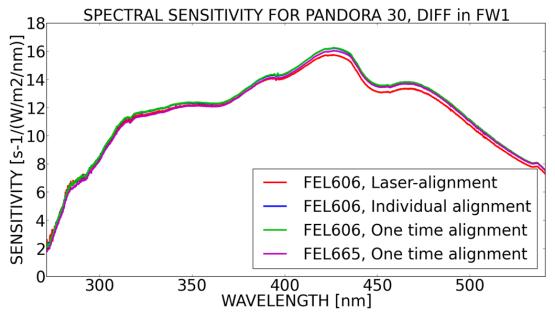


Figure 23: Absolute sensitivity of Pandora 30 for direct sun observations (i.e. with a diffuser in the filterwheels). Different lines refer to different alignment methods and different calibration lamps.

## 6.3.10 Wavelength Correction

An optional last data correction step is the wavelength correction. This is only needed for users which require the L1 data to be on a fixed wavelength grid. It is not necessary for the current operational Pandonia data products. In this step the measured spectra are compared to a theoretical high resolution extraterrestrial spectrum [4] and shifted to the nominal wavelength grid.



# 6.4 L2Fit Algorithm - Spectral Fitting

The L2Fit data are the result of applying the Blick spectral fitting algorithm (BlickSFA) on the L1 data, which is described in this section.

### 6.4.1 Lambert-Beer's law

BlickSFA is based on Lambert-Beer's law.

$$F(\lambda) = F_0(\lambda) \cdot exp\left(-\sum_{j=1}^{n_{\text{EX}}} \tau s_j(\lambda)\right)$$
(78)

 $\lambda$  Wavelength

 $F_0(\lambda)$  Spectrum at wavelength  $\lambda$  reaching the top of the atmosphere; that is the "Extraterrestrial spectrum" adjusted for the Sun-Earth distance

 $F(\lambda)$  Spectrum at wavelength  $\lambda$  reaching the instrument's entrance window

n<sub>EX</sub> Number of extinction processes in the atmosphere

 $\tau s_i(\lambda)$  Slant optical depth at wavelength  $\lambda$  for extinction process j

The slant optical depth  $\tau s_i(\lambda)$  is given by:

$$\tau s_{j}(\lambda) = q s_{j}^{*}(\lambda) \cdot \sigma_{j}(\lambda, T_{j}) 
= q s_{j}(\lambda) \cdot \tau_{STDj}(\lambda, T_{j}) 
= m_{j}(\lambda) \cdot q_{i} \cdot \tau_{STDj}(\lambda, T_{j})$$
(79)

 $qs_j^*(\lambda)$  (Absolute) slant column amount for extinction process j at wavelength  $\lambda$  (units e.g. molecules per cm<sup>2</sup>)

T<sub>i</sub> Effective temperature of extinction process j (see equation 3)

 $\sigma_j(\lambda, T_j)$  Cross section for extinction process j at wavelength  $\lambda$  and effective temperature  $T_j$  (units e.g. cm<sup>2</sup> per molecule)

 $\tau_{\text{STDi}}(\lambda, T_i)$  "Standard vertical optical depth" for extinction process j at wavelength  $\lambda$ 

 $qs_i(\lambda)$  (Relative) slant column amount for extinction process j at wavelength  $\lambda$  (no dimension)

 $m_i(\lambda)$  AMF (or optical mass) for extinction process j at wavelength  $\lambda$ 

q<sub>j</sub> (Relative) vertical column amount extinction process j (no dimension)

For the standard vertical optical depth  $\tau_{STDj}(\lambda,T_j)$ , BlickSFA uses the settings as listed in table 19. Note that introducing  $\tau_{STDj}$  has no other reason than bringing the relative column amounts  $qs_j$  or  $q_j$  into values around 1, which makes the whole algorithm numerically more stable. Hence using even very different values for  $\tau_{STDj}$  would not change the results, as long as they are in a similar order of magnitude.

#### 6.4.2 Convolution

Real instruments do not measure  $F(\lambda)$ , since their slit function extends over a certain bandpass. In the case absolute calibration has been applied, the L1 value at pixel i is given by a convolution of  $F(\lambda)$  with the slit function  $S_i(\lambda)$  at the center wavelength  $\lambda_i$  of the pixel.



$$L1_{i} = F_{i} = \int_{\Delta \lambda_{i}} F(\lambda) \cdot S_{i}(\lambda_{i} - \lambda) \cdot d\lambda$$
(80)

 $\Delta \lambda_i$  is the instrument bandpass at pixel i and  $F_i$  is the high-resolution spectra convoluted at pixel i. Applying the same convolution over the entire equation 78 yields:

$$F_{i} = \int_{\Delta \lambda_{i}} F_{0}(\lambda) \cdot exp\left(-\sum_{j=1}^{n_{\text{EX}}} \tau s_{j}(\lambda)\right) \cdot S_{i}(\lambda_{i} - \lambda) \cdot d\lambda$$
(81)

Equation 81 is the fundamental equation for another type of trace gas retrieval algorithm, often called "Direct fitting algorithm". In such algorithm all unknowns in equation 81 (the  $m_j(\lambda)$ ,  $q_j$ , and  $T_j$ ) are varied in an iterative process until the right and left side of the equation agree within a certain tolerance for all pixels i. These types of algorithms are very time consuming, since the numerical convolution is applied at each iteration step and even nowadays they are not practical for near real time data processing. Therefore the Blick Software Suite uses the following approximation of equation 81:

$$F_{i} = F_{0i} \cdot exp\left(-\sum_{j=1}^{n_{\text{EX}}} \tau s_{ji}\right)$$
(82)

 $F_0(\lambda)$  is the convoluted top-of-atmosphere spectrum given by equation 83 and  $\tau s_{ji}$  is the slant optical depth of absorber j for slant column  $qs_i$  and effective temperature  $T_i$  at pixel i, which is described in the next section.

$$F_{0i} = \int_{\Delta \lambda_i} F_0(\lambda) \cdot S_i(\lambda_i - \lambda) \cdot d\lambda$$
 (83)

## 6.4.3 Slant Optical Depth

BlickSFA's method to calculate the slant optical depth for each trace gas j listed in FSE "Fitted Gases" with slant column  $qs_j$  and effective temperature  $T_j$  at pixel i is done in a two step process.

The first step is to calculate the convoluted the slant optical depth of extinction process j at wavelength  $\lambda_i$  and different relative slant columns  $qs_j$ ,  $\tau s_j(\lambda_i, m_k)$ .

$$\tau s_{j}(\lambda_{i},qs_{j}) = -ln \left[ \frac{\int\limits_{\Delta\lambda} F_{0}(\lambda) \cdot exp\left(-qs_{j} \cdot \tau_{STDj}(\lambda,T_{j})\right) \cdot S_{i}(\lambda_{i}-\lambda) \cdot d\lambda}{\int\limits_{\Delta\lambda} F_{0}(\lambda) \cdot S_{i}(\lambda_{i}-\lambda) \cdot d\lambda} \right] \tag{84}$$

All variables are as in equations 79 and 81. Equation 84 is calculated for qs<sub>i</sub> values ranging from 1 to 8.

The second step consists in expressing the  $\tau s_{ji}$  as a function of  $qs_j$  using so-called OD-fitting parameters  $A_{ii}$ ,  $B_{ii}$ , and  $C_{ii}$ .

$$\begin{split} \tau s_{ji} &= A_j(\lambda_i, T_j) \cdot q s_j^{1+B_j(\lambda_i, T_j) + C_j(\lambda_i, T_j) \cdot ln(qs_j)} \\ &= A_{ji} \cdot q s_j^{1+B_{ji} + C_{ji} \cdot ln(qs_j)} \end{split} \tag{85}$$

Equation 85 is a more general approach than what is usually used in classical Differential Optical Absorption Spectroscopy (DOAS) [21], where only linear absorbers are included. The BlickSFA is also capable to include



stronger extinction processes, for which the relationship between the slant column amount and the slant OD is not linear. We call these species "non-linear absorbers".

Which OD fitting method is used for absorber j by the BlickSFA is decided by FSE "Gas OD Meths". There are four options:

<b>OD Fitting Method</b>	Description
OD Fitting Method 0	Here no OD fitting is done and $\tau s_{ji}(\lambda_i,qs_j=1)$ from equation 84 is used as an approximation of $\tau s_{ji}$ . For this method OD-fitting parameters B and C result zero.
OD Fitting Method 1	A linear fit forced through the point $(0,0)$ is done in the $\tau s_j(\lambda_i,qs_j)$ calculated at $qs_j$ ranging from 1 to 8. Also for this method OD-fitting parameters B and C result zero.
OD Fitting Method 2	A linear fit is done for $ln[\tau s_j(\lambda_i,qs_j)]$ calculated at $qs_j$ ranging from 1 to 8 as a function of $ln[qs_j]$ . For this method OD-fitting parameter B differs from zero and C is zero.
OD Fitting Method 3	A quadratic fit is done for $ln[\tau s_j(\lambda_i,qs_j)]$ calculated at $qs_j$ ranging from 1 to 8 as a function of $ln[qs_j]$ . For this method OD-fitting parameters B and C differ from zero.

The technique of using non-linear relationships between the slant column and the slant OD has already been applied for filter instruments, where OD fitting method 2 was implemented [27]. Method 3 is an extension of method 2 as a novel approach of BlickSFA.

The different OD-fitting methods are illustrated in figure 24. OD fitting method 0 is the dark blue line, method 1 the green line, method 2 the magenta line and method 3 the light blue line. For this very strong water vapor line at 823 nm, any other OD fitting method than 3 would result in huge errors.



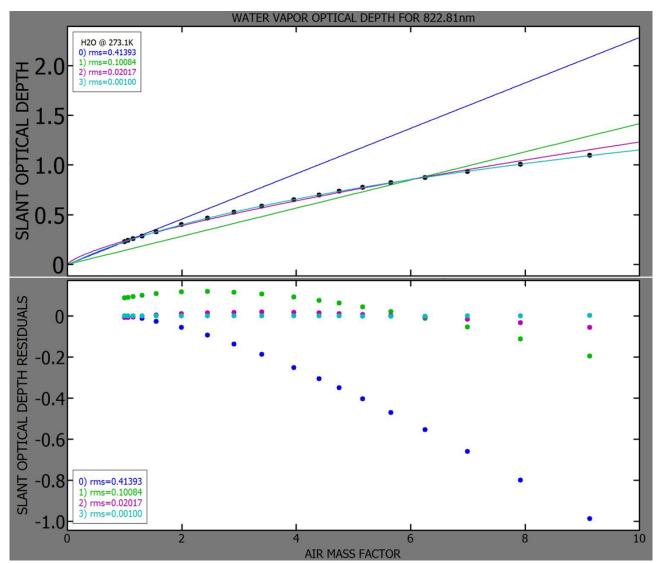


Figure 24: Top: Slant optical depth as a function of the AMF for a vertical column amount of 1 cm precipitable water calculated for AMFs from 1 to 9.3 (black dots) at a strong water vapor line (822.81 nm). Colored lines are different fits in the black dots using 1 (dark blue and green), 2 (magenta) or 3 (light blue) fitting parameters. Bottom: Difference between the black dots and the fitted lines from the top panel. The rms of the difference is given in the figure legend.

OD-fitting parameters A, B, and C from equation 85 are in general a function of the effective gas temperature  $T_i$ . The BlickSFA allows a temperature dependence up to order 2.

$$A_{ji} = A_{jCi} + A_{jLi} \cdot \frac{T_j - T_{jREF}}{T_{SCALE}} + A_{jQi} \cdot \left(\frac{T_j - T_{jREF}}{T_{SCALE}}\right)^2$$
(86)

 $A_{jCi}$ ,  $A_{jLi}$ , and  $A_{jQi}$  are the constant, linear, and quadratic terms of OD-fitting parameter A in temperature,  $T_j$  is the effective gas temperature for gas j,  $T_{jREF}$  is the reference temperature of gas j as given in table 24 and  $T_{SCALE}$  is a temperature scale factor given in the ICF (usually 50 K). The same equation 86 is used for OD-fitting parameters B and C.



## 6.4.4 Basic Fitting Equation

Taking the logarithm of equation 82 and rearranging we obtain:

$$lnF_{0i} - lnF_i = \sum_{i=1}^{n_{EX}} \tau s_{ji}$$
(87)

The following extinction processes are included in the right side of the equation:

- Molecular absorption by gases (see table 19)
- Rayleigh and Raman scattering by air molecules
- Mie scattering and absorption by aerosols

In addition there are instrumental effects, which can alter equation 87:

- Stray light can alter lnF<sub>i</sub>, if is not fully corrected for in the L1 algorithm.
- If the instrument's dispersion or slit function is different than what the calibration assumes, then all expressions change, since the convolution is wrong.
- If the instrument is not absolutely calibrated, the spectral sensitivity (see e.g. figure 23) has to be added as an additive term to the equation.

The BlickSFA basic fitting equation takes all the effects mentioned above into account and modifies equation 87 accordingly. Details to each term are given in the next section 6.4.5.

$$lnF_{0i} - ln\left(F_{i} - P_{OFFSi}\right) - \tau s_{FIXi} = \sum_{j=1}^{n_{GAS}} \tau s_{ji}(qs_{j}, T_{j}) + \tau_{RINGi} \cdot qs_{RING} + P_{SMOi} + P_{RESCi}$$
(88)

i Index for pixel inside the limits of the fitting window, i=1 to n; the center-wavelength of pixel i is  $\lambda_i$ 

 $F_{0i}$  Reference spectrum at pixel i used in the fitting (see table 23)

F<sub>i</sub> L1 data for pixel i

P<sub>OFFSi</sub> Offset polynomial evaluated at pixel i

 $\tau$ s<sub>FIXi</sub> "Known" slant optical depth at pixel i

j Index for atmospheric absorber, j=1 to n<sub>GAS</sub>

 $\tau s_{ji}$  Slant optical depth of absorber j for slant column  $qs_j$  and effective temperature  $T_j$  at pixel i (see section 6.4.3)

qs<sub>RING</sub> "Equivalent" slant column amount of the Ring effect

 $\tau_{RINGi}$  Standard vertical optical depth of the Ring effect at pixel i

P<sub>SMOi</sub> Smooth-part polynomial evaluated at pixel i

P<sub>RESCi</sub> Resolution change polynomial evaluated at pixel i



## 6.4.5 Terms of Basic Fitting Equation

#### $\tau$ SFIXi

The "known" slant optical depth  $\tau s_{FIXi}$  is initially set to 0. If FSE "Mol Scatt" is set to "YES", then the estimated molecular (Rayleigh and Raman) scattering extinction slant optical depth  $\tau s'_{SCAi}$  for the given conditions is added to  $\tau s_{FIXi}$ .  $\tau s'_{SCAi}$  is calculated by:

$$\tau s'_{SCAi} = \frac{p}{p_{STAN}} \cdot \tau s_{SCAi}$$
 (89)

p is the estimated air pressure at the measurement site (calculated using the station altitude),  $p_{STAN}$  is the standard pressure (=1013.25hPa), and  $\tau s_{SCAi}$  is the standard molecular optical depth for pixel i. BlickSFA uses molecular scattering cross sections based on the work of *Bodhaine et al.* [5] and *Owens* [20] for 1 standard atmosphere of dry air molecules (2.1533e25 molc/cm<sup>2</sup>) containing 400 ppm of  $CO_2$ . The effective height (see equation 2) is estimated to  $h_{SCAEFF}$ =6.2 km.

If FSE "Fixed Gases" is not empty, then the "known" slant optical depths for all gases listed in FSE "Fixed Gases" are also added to  $\tau s_{FIXi}$ . Those values are taken from L2Fit data from other fitting setups, of which the numbers are listed in FSE "Fixed Numbers".

#### Poffsi

The offset polynomial  $P_{OFFSi}$  is used to reduce effects of uncorrected stray light in the data. The assumption is that the way stray light alters the measurements can be approximated by a polynomial. While this assumption is clearly not true, we still see some minor improvement in the spectral fitting results adding such polynomial.

$$P_{\text{OFFS}i} = \overline{F} \cdot \sum_{k=0}^{\text{noffs}} p_{\text{OFFS}k} \cdot \lambda_i^k$$
(90)

 $\overline{F}$  is the mean over the  $F_i$ , the  $p_{OFFSk}$  are the coefficients of the offset polynomial, and the  $\lambda_i$  are the scaled wavelength-centers for each pixel i (scaling with equation 4 and  $x_{MIN}$  and  $x_{MAX}$  being the minimum and maximum nominal wavelength of the instrument's total wavelength range).

#### auRINGi

If FSE "Ring" is set to "YES", the Ring spectrum [10] is also fitted by the BlickSFA. The standard vertical optical depth of the Ring effect  $\tau_{RINGi}$  is given in the ICF.

### P<sub>SMOi</sub>

The purpose of the smooth-part polynomial  $P_{SMOi}$  is to "absorb" all smooth structures in equation 88, e.g. Rayleigh scattering if it has not been included in  $\tau s_{FIXi}$ , or the instrument spectral sensitivity for the case the unit is not absolutely calibrated.

$$P_{\text{SMOi}} = \sum_{k=0}^{\text{nsmo}} p_{\text{SMOk}} \cdot \lambda_i^k$$
(91)

The  $p_{SMOk}$  are the coefficients of the smooth-part polynomial and the  $\lambda_i$  are as in equation 90.

### P<sub>RESCi</sub>

The resolution change polynomial P<sub>RESCi</sub> is given by

$$P_{\text{RESCi}} = \Delta_{\text{RESCi}} \cdot \sum_{k=0}^{\text{nresc}} p_{\text{RESCk}} \cdot \lambda_i^k$$
 (92)



The  $p_{RESCk}$  are the coefficients of the resolution change polynomial and the  $\lambda_i$  are as in equation 90.  $\Delta_{RESCi}$  is the estimated change of equation 88 for a 1 % change in the instrument's resolution. It is calculated by

$$\Delta_{\text{RESCi}} = m_{i} \cdot \Delta \tau_{\text{TOTi}} - \Delta ln F_{0i} \tag{93}$$

 $\Delta lnF_{0i}$  is the relative change (or absolute change of the logarithm) of  $F_{0i}$  for a 1 % increase in the instrument's resolution.

$$\Delta lnF_0(\lambda_i) = lnF_{0RESC}(\lambda_i) - lnF_0(\lambda_i)$$
(94)

 $lnF_{0RESC}$  is the logarithm of the reference spectrum for a 1 % increased resolution and is given in the ICF.  $\Delta \tau_{TOTi}$  is the absolute change of the standard total optical depth for a 1 % increase in the instrument's resolution.

$$\Delta \tau_{\text{TOT}}(\lambda_i) = \tau_{\text{TOTRESC}}(\lambda_i) - \tau_{\text{TOT}}(\lambda_i)$$
(95)

 $\tau_{\text{TOTi}}$  and  $\tau_{\text{TOTRESCi}}$  are the standard total optical depth change and the standard total optical depth for a 1 % increased resolution respectively and are given in the ICF.

## 6.4.6 Least Squares Minimization

BlickSFA is executed for a given fitting setup as defined in the 'f-codes'-table of the processing setups file (section 5.10). It retrieves  $n_{PAR}$  output-variables from n measured data points, which are the L1 data for all pixels inside the fitting window limits. These limits are defined by fitting setup entries FSE "WL-start" and "WL-end" (for all FSEs see table 22). BlickSFA requires  $n_{PAR} \le n$ . The total number of output-parameters is given by:

$$n_{PAR} = n_{GAS} + n_{TEMP} + n_{RING} + (n_{PAR} + 1) + (n_{OFFS} + 1) + (n_{WLC} + 1) + (n_{RESC} + 1)$$
(96)

 $n_{GAS}$  Number of absorbers, for which slant columns are fitted; this is the length of FSE "Fitted Gases";  $n_{GAS} \ge 0$ 

 $n_{TEMP}$  Number of absorbers, for which effective gas-temperatures are fitted; this is the length of FSE "Fitted Temps";  $0 \le n_{TEMP} \le n_{GAS}$ 

n<sub>RING</sub> =1 if the Ring effect is fitted (FSE "Ring" equals YES); =0 if the Ring effect is NOT fitted (FSE "Ring" equals YES)

n<sub>SMO</sub> Order of the smooth-part polynomial; from FSE "npol"

n<sub>OFFS</sub> Order of the offset polynomial; from FSE "noffs"

n<sub>WLC</sub> Order of the wavelength change polynomial; from FSE "nwlc"

n<sub>RESC</sub> Order of the resolution change polynomial; from FSE "nresc"

A value of -1 for  $n_{SMO}$ ,  $n_{OFFS}$ ,  $n_{WLC}$  or  $n_{RESC}$  means the respective polynomial is not used in the fitting process. The residuals  $\xi_i$  are the differences between the left side and the right side of the basic fitting equation.

$$\xi_{i} = lnF_{0i} - ln\left(F_{i} - P_{OFFSi}\right) - \tau s_{FIXi} - \sum_{i=1}^{n_{GAS}} \tau s_{ji} - \tau_{RINGi} \cdot qs_{RING} - P_{SMOi} - P_{RESCi}$$

$$(97)$$

The BlickSFA minimizes the residuals in a least squares sense.

$$\sum_{i=1}^{n} \left(\frac{\xi_{i}}{\sigma_{i}}\right)^{2} \to Minimum \tag{98}$$

 $\sigma_i$  is the noise in the measurements. If FSE "Uncertainty" is set to "NO", then all elements of  $\sigma_i$  are set to zero, i.e. the noise is not included. Otherwise  $\sigma_i$  in a combination of the noise in the L1 measurements  $\sigma_{MEASi}$  and in the reference  $\sigma_{0i}$ .



$$\sigma_{\rm i} = \sqrt{\sigma_{\rm MEASi}^2 + \sigma_{\rm 0i}^2} \tag{99}$$

 $\sigma_{\text{MEASi}}$  is given in the L1 data and  $\sigma_{0i}$  in the ICF. Note that  $\sigma_{0i}$ =0 if FSE "Reference" equals "ExtRef". Equation 99 assumes that the measurements and reference noise are uncorrelated, which is not always be the case, e.g. in the trivial case where the measurements and the reference are the same spectra.

## 6.4.7 Linear Fitting

BlickSFA consists of a linear and a non-linear part. First, equation 88 is linearized and solved. Second, if necessary, the results of the linear fitting are used as first guesses for a non-linear fitting process. The least squares fitting in the BlickSFA follows the description of *Press et al.* [22], chapter 15. The linearized form of equation 88 is:

$$lnF_{0i} - ln\left(F_{i} - P_{OFFSi}\right) - \tau s_{FIXi} = \sum_{j=1}^{n_{GAS}} A_{jCi} \cdot qs_{j} + \sum_{j=1}^{n_{TEMP}} A_{jLi} \cdot \Delta T_{j} \cdot qs_{j} + \tau_{RINGi} \cdot qs_{RING} + P_{SMOi} + P_{RESCi}$$
(100)

 $A_{jCi}$  and  $A_{jLi}$  are the constant and linear term in temperature of OD-fitting parameter A (see equation 86) and  $\Delta T_j$  is the scaled temperature difference to the reference temperature.

$$\Delta T_{j} = \frac{T_{j} - T_{jREF}}{T_{SCALE}} \tag{101}$$

All variables in 101 are as in equation 86. So the linearized version of the fundamental equation assumes  $P_{OFFSi}=0$  (i.e.  $n_{OFFS}=-1$ ), no wavelength change ( $n_{WLC}=-1$ ), a linear dependence of the slant optical depth on the slant column amount (OD-fitting parameters B and C equal 0), and at most a linear dependence of OD-fitting parameter A on the temperature ( $A_Q=0$ ). The number of output-parameters  $n_{PARLIN}$  in the linearized case is:

$$n_{PARLIN} = n_{GAS} + n_{TEMP} + n_{RING} + (n_{SMO} + 1) + (n_{RESC} + 1)$$
 (102)

Equation 100 can be written in matrix form

$$\overrightarrow{Y} = \mathbf{M} \cdot \overrightarrow{X} \tag{103}$$

 $\overrightarrow{Y}$  is a (n,1)-vector. Its elements  $Y_i$  are given by

$$Y_{i} = lnF_{0i} - lnF_{i} - \tau s_{FIXi}$$

$$\tag{104}$$

 $\overrightarrow{X}$  is a  $(n_{PARLIN}, 1)$ -vector

$$\overrightarrow{X} = \left[qs_1, ..., qs_{nGAS}, \Delta T_1 \cdot qs_1, ..., \Delta T_{nTEMP} \cdot qs_{nTEMP}, qs_{RING}, p_{SMO0}, ..., p_{SMOnSMO}, p_{RESC0}, ..., p_{RESCnRESC}\right]^T \tag{105}$$

 $\mathbf{M}$  is a  $(n,n_{PARLIN})$ -matrix. One row of  $\mathbf{M}$  is given by

$$\overrightarrow{\mathbf{M}_{i:}} = \left[ \mathbf{A}_{1\text{Ci}}, ..., \mathbf{A}_{n\text{GASCi}}, \mathbf{A}_{1\text{Li}}, ..., \mathbf{A}_{n\text{TEMPLi}}, \tau_{\text{RINGi}}, 1, \lambda_{i}, \lambda_{i}^{2}, ..., \lambda_{i}^{n\text{SMO}}, \Delta_{\text{RESCi}}, \Delta_{\text{RESCi}} \cdot \lambda_{i}, ..., \Delta_{\text{RESCi}} \cdot \lambda_{i}^{n\text{RESC}} \right] \tag{106}$$

Each element of vector  $\overrightarrow{\mathbf{Y}}$  and each column of matrix  $\mathbf{M}$  are divided by  $\sigma_i$  to include the uncertainty in the fitting.  $\mathbf{M}^*$  is called the design matrix of the fitting problem [22].



$$Y^*_{i} = \frac{Y_{i}}{\sigma_{i}} \qquad \overrightarrow{M^*_{i:}} = \frac{\overrightarrow{M_{i:}}}{\sigma_{i}}$$
 (107)

BlickSFA solves for  $\overrightarrow{Y}$  in equation 108 using the linalg.lstsq-routine of Python's numpy module (see section 2.4).

$$\overrightarrow{Y^*} = \mathbf{M}^* \cdot \overrightarrow{X} \tag{108}$$

BlickSFA returns two versions of the root mean square of the spectral fitting residuals. rms (equation 109) is called "rms of unweighted spectral fitting residuals", while wrms (equation 110) is called "Normalized rms of weighted spectral fitting residuals".

$$rms = \sqrt{\frac{\sum_{i=1}^{n} \xi_i^2}{n - n_{FIT}}}$$
 (109)

$$wrms = \sqrt{\frac{\sum_{i=1}^{n} \left(\frac{\xi_{i}}{\sigma_{i}}\right)^{2}}{\sum_{i=1}^{n} \left(\frac{1}{\sigma_{i}}\right)^{2}} \cdot \frac{n}{n - n_{FIT}}}$$
(110)

In both equations the residuals  $\xi_i$  are calculated inserting the output parameters retrieved in the spectral fitting process in equation 97 setting  $n_{FIT}=n_{PARLIN}$ . If the measurement uncertainty was not included in BlickSFA, rms equals wrms. Otherwise (i.e. the measurement uncertainty was included) rms is not a very meaningful parameter. So wrms should be used to estimate the quality of the spectral fitting process.

In addition to rms and wrms, BlickSFA also calculates the 'expected (weighted) rms based on the measured uncertainty only', which are called rmsu and wrmsu and are also in the L2Fit output (see table 31). They can be used to estimate of how much of the rms can be explained by instrumental and input noise in the data. In the same way

$$rmsu = \sqrt{\frac{\sum\limits_{i=1}^{n} \sigma_{i}^{2}}{n - n_{FIT}}}$$
 (111)

$$wrmsu = \sqrt{\frac{n}{\sum\limits_{i=1}^{n} \left(\frac{1}{\sigma_i}\right)^2} \cdot \frac{n}{n - n_{FIT}}}$$
 (112)

To obtain the effective gas temperature-difference  $\Delta T_j$ , BlickSFA divides the output parameter  $\Delta T_j \cdot qs_j$  by the corresponding output parameter  $qs_j$ .

BlickSFA also calculates the uncertainty  $\sigma_{Xj}$  for each element of X. The covariance matrix C is given as the inverse of the matrix product of  $M^*$  transposed times  $M^*$ .

$$\mathbf{C} = inv\left(\mathbf{M}^{*T} \cdot \mathbf{M}^{*}\right) \tag{113}$$

The main diagonal elements of C are the variances of the  $X_i$ .

$$\sigma_{Xj}^2 = C_{ji} \tag{114}$$



Note that the  $\sigma_{Xj}$  are not the accuracy of the output parameters. They are the statistical uncertainty or noise of the output parameters based on the uncertainty of the measurements. If the measurement uncertainty  $\sigma_i$  was NOT included in the spectral fitting process, the  $\sigma_{Xj}$  are meaningless.

If no non-linear fitting is needed ( $n_{OFFS}$ =-1,  $n_{WLC}$ =-1, OD parameters B, C and AQ equal 0) or if FSE "Linear Fit" is set to "NO", then the BlickSFA stops here. Otherwise the non-linear spectral fitting is applied.

### 6.4.8 Non-linear Fitting

The non-linear fitting minimizes the residuals in equation 97 in an iterative way using the optimize leastsq-routine of Python's scipy module (see section 2.4). The parameters retrieved from the linear fitting are used as first guesses. The first guesses of all remaining output parameters (i.e. those not included in the linear fitting) are set to 0. The number of function evaluations is limited to 1000. Equations 109 and 110 also apply to the non-linear fitting with  $n_{FIT}=n_{FIT}$ . While the non-linear fitting is a very fast process (<0.1 s per fit), the non-linear fitting can take up to several seconds per fit.



# 6.5 L2 Algorithms

The Blick Software Suite includes some L2 algorithms, which produce atmospheric data such as total or tropospheric vertical column amounts and surface concentrations of trace gases. All L2 algorithms in the Blick Software Suite have several characteristics in common:

- They are not using any external data, i.e. only measurements from Pandora.
- Only L2Fit data from one single routine (i.e. one measurement sequence) are used. This means they can be applied in real time directly after the measurement sequence is finished.
- They are not performing radiative transfer calculations (RTC) and are consequently very fast.
- They do not require any special software license to run and the source code is provided with the standard installation of the Blick Software Suite.

We envision that, at a later stage, there will also be additional algorithms developed for Pandonia. If any new algorithm fulfills all the criteria listed above, it might be added to the Blick Software Suite. Otherwise it will be considered an "offline algorithm". Offline algorithms may be written in a language other than Python, may require licenses, may include RTC, etc. We believe that e.g. for a very sophisticated ozone profile algorithm, one will likely need to analyze all the measurements from an entire day together as a group and also do RTC, therefore this will probably be an offline algorithm.

The existing L2 algorithms are described in the sections below.

## 6.5.1 Direct Algorithm

This algorithm is used for direct sun and direct moon measurements and produces total vertical column amounts. The L2Fit data return the slant column amount  $q_{ij}$  for absorber j. To convert to the vertical column amount  $q_{ij}$ , the slant columns are divided by the direct sun (or moon) AMF for extinction process j,  $m_{DIRj}$ . It is estimated by

$$m_{DIRj}(ZA) = sec\left(arcsin\left[\left(\frac{R}{R + h_{EFFj}}\right) \cdot sin(ZA^*)\right]\right) \tag{115}$$

R is the distance from the center of the Earth to the measurement location (about 6370 km, refined based on the location's latitude),  $ZA^*$  is the apparent solar (or lunar) zenith angle (i.e. the geometrical ZA corrected for refraction), and  $h_{EFF_i}$  is the effective height for the extinction process j taken from table 19.

# 6.5.2 Five Angles O2O2-Ratio Algorithm

This algorithm has been developed in collaboration with Elena Spinei from NASA/Goddard Space Flight Center. It estimates the surface concentration (vmr) and the tropospheric column (qt) of a trace gas based on sky measurements at 5 viewing zenith angles (VZA) at a fixed azimuth. The 5 VZAs are 0°, 60°, 75°, one angle above 87° (the "largest angle") and one angle 1° smaller than the largest angle. The largest angle should be as large as possible, but not having an object other than the sky in the FOV. Since the Pandora Sky-FOV is about 1.5° full width, we recommend staying 1° away from the horizon. I.e. at a flat place, the lowest angle would be 89°. When the horizon is elevated (e.g. trees or nearby mountains in the view) then the largest angle might be <89° and if the instrument is located on a hill it might be even possible to go above 89°.

Although not necessarily required by the algorithm, the measurements should be done in "V-shape", i.e. all angles are measured twice around a central angle. The typical sequence (assuming the largest angle is 89°) is 0°, 60°, 75°, 88°, 89°, 88°, 75°, 60°, 0°. The V-shape allows to interpolate all measurements to the central



measurement and therefore avoids systematic errors at larger SZAs. The central measurement is used as a reference in the spectral fitting. Besides the slant column amounts qs of the trace gas, for which vmr and qt should be determined, the algorithm also needs the slant columns of the oxigen dimer qsO4 for all the angles, since it is based on the assumption of similar AMFs for the trace gas and  $O_2O_2$ . Here we will use indices 0 to 4 for the angles, where index 0 corresponds to largest VZA, 1 to the second largest VZA, 2 to VZA=75°, 3 to 60° and 4 to 0°. Since the largest VZA is used as a reference, qs<sub>0</sub> and qsO4<sub>0</sub> equal 0 by definition. Here is a step-by-step description of the algorithm:

#### **Step 1: Extract Viewing Angles**

In the case more than just the required 5 angles were measured, the data are reduced to the 5 angles only in this step.

#### Step 2: Interpolate qs to central measurement

If V-shape was measured, the fitted slant columns at each VZA are interpolated in time to the central measurement. After this step there are 8 effective data inputs left, which are qs<sub>1</sub> to qs<sub>4</sub> and qsO<sub>4</sub> to qsO<sub>4</sub>.

### Step 3: Extrapolate to VZA=90°

Since we actually would like to have the largest VZA at  $90^{\circ}$ , but in general cannot measure this, the following extrapolations is done to get the estimated differential slant column between VZA= $90^{\circ}$  and VZA= $0^{\circ}$  qsh in the case the largest viewing angle VZA<sub>0</sub> was below  $90^{\circ}$ :

$$qsh = qs_1 \cdot \frac{VZA_0 - 90}{VZA_0 - VZA_1} - qs_4 \tag{116}$$

This step is also done for using the same method to obtain qshO4.

#### Step 4: Modifications to account for the heterogeneous case

RTCs have shown that tropospheric AMFs of  $O_2O_2$  within the lowest 100 m are ranging between 0.5 and 3 when considering scenarios with different SZAs, low to high AODs, different aerosol phase functions etc. Therefore we are using an empirical correction on the  $O_2O_2$  slant columns to adjust for the case of not equal mixing conditions:

$$qshO4c = \frac{qshO4}{qsO4_2 - qsO4_4} \cdot nO4 \cdot heffO4$$
 (117)

The estimation in equation 117 is sensitive to clouds, especially low clouds when multiple scattering within the cloud or up-down the cloud changes the photon path differently for the trace gas and  $O_2O_2$  due to difference in profiles. heffO4 in equation 117 is the estimated effective  $O_2O_2$  height calculated with equation 2. nO4 is the estimated  $O_2O_2$  surface concentration, calculated by:

$$nO4 = \left(\frac{P_{surf} \cdot f_{O2}}{T_{surf} \cdot k_{Boltz}}\right)^{2}$$
 (118)

 $P_{surf}$  is the surface pressure,  $f_{O2}$  the  $O_2$  mixing ratio,  $T_{surf}$  is the surface temperature, and  $k_{Boltz}$  the Boltzmann constant. Blick Software Suite uses  $f_{O2}$ =0.20946 and calculates  $P_{surf}$ ,  $T_{surf}$  and heffO4 from a climatology [1] based on the season, latitude and height a.s.l. of the location. The values for sea level are shown in figure 25.



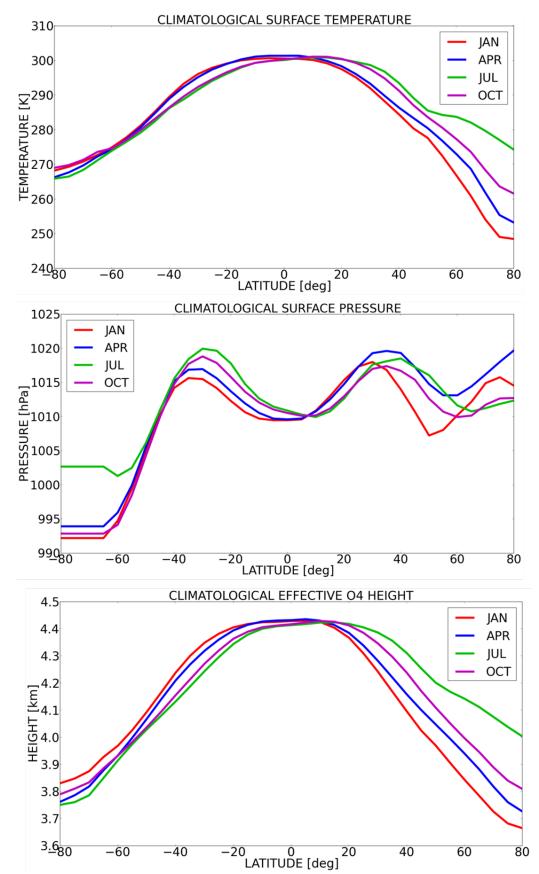


Figure 25: Climatological COS [1] surface temperature (top), surface pressure (middle) and effective  $O_2O_2$  height (bottom) at sea level\* for the months of January, April, July and October as a function of latitude. \*For latitudes below 60 deg South the values are actually not at sea level, but at the average height of the surface in Antarctica.



The same correction as in equation 117 is also applied to qsO4<sub>4</sub>.

$$qs4O4c = \frac{qsO4_4}{qsO4_2 - qsO4_4} \cdot nO4 \cdot heffO4$$
 (119)

#### **Step 5: Get the surface concentration**

First four different estimations of the surface concentration are calculated, called vmr1 to vmr4.

$$vmr1 = \frac{qs_4 \cdot P_{surf} \cdot f_{O2}}{qsO4_4 \cdot T_{surf} \cdot k_{Boltz}}$$

$$vmr2 = \frac{qsh \cdot P_{surf} \cdot f_{O2}}{qshO4 \cdot T_{surf} \cdot k_{Boltz}}$$

$$vmr3 = \frac{-qs_4 \cdot P_{surf} \cdot f_{O2}}{qs4O4c \cdot T_{surf} \cdot k_{Boltz}}$$

$$vmr4 = \frac{qsh \cdot P_{surf} \cdot f_{O2}}{qshO4c_4 \cdot T_{surf} \cdot k_{Boltz}}$$

$$(120)$$

For the final best guess of the surface concentration the largest value among vmr1 to vmr4 is picked. The index of the largest vmr (between 1 and 4) is called the "surface concentration index". As a rough estimation of the uncertainty uvmr of the final vrm, the algorithm uses simply the standard deviation over the 4 values vmr1 to vmr4. Note that this is a different approach for the uncertainty than used in all other uncertainties, since it is not the propagated instrumental and atmospheric noise. Finally the so-called "heterogeneity flag" is calculated. This flag equals 0, if the ratio vmr/uvmr is above 5 and OR if the surface concentration index is 1 or 2. In this case it is assumed that there are rather well mixed conditions. Otherwise the flag equals 1 indicating more heterogeneous conditions near the surface.

### Step 6: Get air mass factors for the tropospheric column

For the tropospheric column amount qt the important VZAs are 60° and 70°. The best guess AMF results from a modified geometrical approach. Two more AMFs, AMF1 and AMF2 are also calculated to get an uncertainty estimation of qt.

$$AMF = \frac{qsO4_2 - qsO4_3}{nO4 \cdot heffO4} + 2$$

$$AMF1 = \frac{qsO4_2 - qsO4_3}{nO4 \cdot heffO4} + \frac{qs_2 - qs_4}{(qs_3 - qs_4) \cdot 2}$$

$$AMF2 = 1$$
(121)

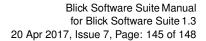
#### **Step 7: Get the tropospheric column**

Three different estimations of the tropospheric column are calculated, called qt, qt1 and qt2.

$$qt = \frac{qs_2 - qs_4}{AMF}$$

$$qt1 = \frac{qs_2 - qs_4}{AMF1}$$

$$qt2 = \frac{qs_3 - qs_4}{AMF2}$$
(122)





For the final best guess of the tropospheric columns qt is taken. As a rough estimation of the uncertainty uqt of the final qt, the algorithm uses simply the standard deviation over the 3 values qt, qt1 and qt2. As for the surface concentration this is a different approach for the uncertainty than used in all other uncertainties, since it is not the propagated instrumental and atmospheric noise.



## 7 References

- [1] Cospar international reference atmosphere: 1986 (0 km to 120 km). URL http://ccmc.gsfc.nasa.gov/modelweb/atmos/cospar1.html.
- [2] N. Abuhassan, A. Cede, and M. Kowalewski. *Pandora spectrometer system Installation and Maintenance Manual Version 4*, 2014. URL http://sciglob.com.
- [3] A. M. Bass and R. J. Paur. The ultraviolet cross-sections of ozone. I. The measurements. II Results and temperature dependence. In C. S. Zerefos and A. Ghazi, editors, *Atmospheric ozone; Proceedings of the Quadrennial Ozone Symposium, Halkidiki, Greece, September 3-7, 1984 (A86-48601 24-46).*, pages 606–616, 1985.
- [4] G. Bernhard, C. R. Booth, and J. C. Ehramjian. Version 2 data of the National Science Foundation's ultraviolet radiation monitoring network: South Pole. *Journal of Geophysical Research (Atmospheres)*, 109(D21), 2004.
- [5] B. A. Bodhaine, N. B. Wood, E. G. Dutton, and J. R. Slusser. On Rayleigh Optical Depth Calculations. *Journal of Atmospheric and Oceanic Technology*, 16:1854, November 1999. doi: 10.1175/1520-0426(1999)016<1854:ORODC>2.0.CO;2.
- [6] Konstanze Bogumil, J Orphal, T Homann, S Voigt, P Spietz, OC Fleischmann, A Vogel, M Hartmann, H Kromminga, H Bovensmann, et al. Measurements of molecular absorption spectra with the sciamachy pre-flight model: instrument characterization and reference data for atmospheric remote-sensing in the 230–2380 nm region. *Journal of Photochemistry and Photobiology A: Chemistry*, 157(2):167–184, 2003.
- [7] A Bongartz, J Kames, F Welter, and U Schurath. Near-uv absorption cross sections and trans/cis equilibrium of nitrous acid. *The Journal of Physical Chemistry*, 95(3):1076–1082, 1991.
- [8] JP Burrows, A Richter, A Dehn, B Deters, S Himmelmann, S Voigt, and J Orphal. Atmospheric remote-sensing reference data from gome-2. temperature-dependent absorption cross sections of o<sub>3</sub> in the 231–794nm range. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 61(4):509–517, 1999.
- [9] Oliver C Fleischmann, Matthias Hartmann, John P Burrows, and Johannes Orphal. New ultraviolet absorption cross-sections of bro at atmospheric temperatures measured by time-windowing fourier transform spectroscopy. *Journal of Photochemistry and Photobiology A: Chemistry*, 168(1):117–132, 2004.
- [10] J. F. Grainger and J. Ring. Anomalous Fraunhofer line profiles. *Nature*, 193:762, 1962.
- [11] Christian A Gueymard. The sun's total and spectral irradiance for solar energy applications and solar radiation models. *Solar energy*, 76(4):423–453, 2004.
- [12] C Hermans, AC Vandaele, S Fally, M Carleer, R Colin, B Coquart, A Jenouvrier, and M-F Merienne. Absorption cross-section of the collision-induced bands of oxygen from the uv to the nir. In *Weakly interacting molecular pairs: unconventional absorbers of radiation in the atmosphere*, pages 193–202. Springer, 2003.
- [13] Henrik Wann Jensen, Frédo Durand, Julie Dorsey, Michael M Stark, Peter Shirley, and Simon Premože. A physically-based night sky model. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 399–408. ACM, 2001.



- [14] Stelios Kazadzis, Alkiviadis Bais, Natalia Kouremeti, Evangelos Gerasopoulos, Katerina Garane, Mario Blumthaler, Barbara Schallhart, and Alexander Cede. Direct spectral measurements with a brewer spectroradiometer: absolute calibration and aerosol optical depth retrieval. *Applied optics*, 44(9):1681–1690, 2005.
- [15] JB Kerr, CT McElroy, DI Wardle, RA Olafson, and WFJ Evans. The automated brewer spectrophotometer. In *Atmospheric Ozone*, pages 396–401. Springer, 1985.
- [16] Robert L Kurucz. New atlases for solar flux, irradiance, central intensity, and limb intensity. *Memorie della Societa Astronomica Italiana Supplementi*, 8:189, 2005.
- [17] J Malicet, D Daumont, J Charbonnier, C Parisse, A Chakir, and J Brion. Ozone uv spectroscopy. ii. absorption cross-sections and temperature dependence. *Journal of atmospheric chemistry*, 21(3):263–273, 1995.
- [18] Richard Meller and Geert K Moortgat. Temperature dependence of the absorption cross sections of formaldehyde between 223 and 323 k in the wavelength range 225–375 nm. *Journal of Geophysical Research: Atmospheres* (1984–2012), 105(D6):7089–7101, 2000.
- [19] David A Newnham and John Ballard. Visible absorption cross sections and integrated absorption intensities of molecular oxygen (o2 and o4). *Journal of Geophysical Research: Atmospheres* (1984–2012), 103 (D22):28801–28815, 1998.
- [20] James C Owens. Optical refractive index of air: dependence on pressure, temperature and composition. *Applied Optics*, 6(1):51–59, 1967.
- [21] Ulrich Platt and Jochen Stutz. Differential absorption spectroscopy. Springer, 2008.
- [22] William H Press, Saul A Teukolsky, William T Vetterling, and Brian P Flannery. Numerical recipes in c: the art of scientific computing, 1992. *Cité en*, page 92, 1992.
- [23] Laurence S Rothman, D Jacquemart, Aetal Barbe, D Chris Benner, M Birk, LR Brown, MR Carleer, C Chackerian Jr, K Chance, LH et al Coudert, et al. The< i> hitran</i> 2004 molecular spectroscopic database. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 96(2):139–204, 2005.
- [24] Laurence S Rothman, Iouli E Gordon, Alain Barbe, D Chris Benner, Peter F Bernath, Manfred Birk, Vincent Boudon, Linda R Brown, Alain Campargue, J-P Champion, et al. The< i> hitran</i> 2008 molecular spectroscopic database. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 110(9): 533–572, 2009.
- [25] Stanley P Sander. Temperature dependence of the nitrogen trioxide absorption spectrum. *The Journal of Physical Chemistry*, 90(17):4135–4142, 1986.
- [26] Stanley P Sander and Randall R Friedl. Kinetics and product studies of the reaction chlorine monoxide+bromine monoxide using flash photolysis-ultraviolet absorption. *The Journal of Physical Chemistry*, 93 (12):4764–4771, 1989.
- [27] Beat Schmid, Joseph J Michalsky, Donald W Slater, James C Barnard, Rangasayi N Halthore, James C Liljegren, Brent N Holben, Thomas F Eck, John M Livingston, Philip B Russell, et al. Comparison of columnar water-vapor measurements from solar transmittance methods. *Applied Optics*, 40(12):1886–1896, 2001.



- [28] Peter Spietz, Juan Carlos Gomez Martin, and John P Burrows. Spectroscopic studies of the i2/o3 photochemistry: Part 2. improved spectra of iodine oxides and analysis of the io absorption spectrum. *Journal of Photochemistry and Photobiology A: Chemistry*, 176(1):50–67, 2005.
- [29] J Stutz, ES Kim, U Platt, P Bruno, C Perrino, and A Febo. Uv-visible absorption cross sections of nitrous acid. *Journal of Geophysical Research: Atmospheres* (1984–2012), 105(D11):14585–14592, 2000.
- [30] G Thuillier, L Floyd, TN Woods, R Cebula, E Hilsenrath, M Hersé, and D Labs. Solar irradiance reference spectra for two solar active levels. *Advances in Space Research*, 34(2):256–261, 2004.
- [31] A. C. Vandaele, C. Hermans, P. C. Simon, M. Carleer, R. Colin, S. Fally, M. F. Mérienne, A. Jenouvrier, and B. Coquart. Measurements of the NO<sub>2</sub> absorption cross-section from 42,000 cm<sup>-1</sup> to 10,000 cm<sup>-1</sup> (238-1000 nm) at 220 K and 294 K. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 59: 171–184, May 1998. doi: 10.1016/S0022-4073(97)00168-4.
- [32] Ann Carine Vandaele, Paul C Simon, Jean Michel Guilmot, Michel Carleer, and Réginald Colin. So2 absorption cross section measurement in the uv using a fourier transform spectrometer. *Journal of Geophysical Research: Atmospheres* (1984–2012), 99(D12):25599–25605, 1994.
- [33] Michael E VanHoosier. Solar ultraviolet spectral irradiance data with increased wavelength and irradiance accuracy. In *SPIE's 1996 International Symposium on Optical Science, Engineering, and Instrumentation*, pages 57–64. International Society for Optics and Photonics, 1996.
- [34] S Voigt, J Orphal, K Bogumil, and JP Burrows. The temperature dependence (203–293 k) of the absorption cross sections of o< sub> 3</sub> in the 230–850 nm region measured by fourier-transform spectroscopy. *Journal of Photochemistry and Photobiology A: Chemistry*, 143(1):1–9, 2001.
- [35] Rainer Volkamer, Peter Spietz, John Burrows, and Ulrich Platt. High-resolution absorption cross-section of glyoxal in the uv–vis and ir spectral ranges. *Journal of Photochemistry and Photobiology A: Chemistry*, 172(1):35–46, 2005.
- [36] Andreas Wahner, Geoffrey S Tyndall, and ANDA R Ravishankara. Absorption cross sections for symmetric chlorine dioxide as a function of temperature in the wavelength range 240-480nm. *Journal of Physical Chemistry*, 91(11):2734–2738, 1987.
- [37] David M Wilmouth, Thomas F Hanisco, Neil M Donahue, and James G Anderson. Fourier transform ultraviolet spectroscopy of the a  $2\pi 3/2$ ? x  $2\pi 3/2$  transition of bro. *The Journal of Physical Chemistry A*, 103(45):8935-8945, 1999.
- [38] Y. Zong, S. W. Brown, B. C. Johnson, K. R. Lykke, and Y. Ohno. Simple spectral stray light correction method for array spectroradiometers. *Applied Optics*, 45(6):1111–1119, 2006.