# ARIEL

# ARIEL Performance Model

| | | Date |
|---|---|---|
| Prepared by: | A. Papageorgiou,<br>E. Pascale,<br>S. Sarkar | 9 Feb 2017 |
| Reviewed by: | | |
| Approved & Released by: | | |

## Document Change Details

| Issue | Date | Page | Description Of Change | Comment |
|---|---|---|---|---|
| 1.0 | 25/05/16 | | First Draft | |
| 2.0 | 09/02/17 | | * Added “Exosim Design” section<br>* Includes Latest Version of ExoSim | |

# DISTRIBUTION LIST

| ARIEL Payload Consortium | | | | External | |
|---|---|---|---|---|---|
| Co-PIs / Science Team Coordinators | | Study Engineering Team Working Group Leads | | European Space Agency | |
| | Giovanna Tinetti | | Paul Eccleston | | Goren Pilbratt |
| | Giusi Micela | | Kevin Middleton | | Ludovic Puig |
| | Jean-Philippe Beaulieu | | Emanule Pace | | Astrid Heske |
| | Paul Hartogh | | Gianluca Morgante | | Ralf Kohley |
| | Enzo Pascale | | Tom Hunt | | |
| | Ignasi Ribas | | Vania Da Deppo | | |
| | Hans Ulrik Nørgaard-Nielsen | | Pino Malaguti | | |
| | Micheal Min | | Jerome Amiaux | | |
| | Mirek Rataj | | Pep Colome | | |
| | Bart Vandenbussche | | Jean-Louis Augures | | |
| | Manuel Gudel | | Etienne Renotte | | |
| | David Luz | | Martin Frericks | | |
| | Tom Ray | | | | |
| | | Other Engineering Team | | | |
| | | | *As necessary for doc* | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

# Table of Contents



**Acronyms**

| | | |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

# 1 Introduction

## 1.1 Scope

This document presents a concise description of **ExoSim**, a numerical end-to-end simulator of transit spectroscopy, which is used as the ARIEL Performance Model. The aim of this document is threefold: 1) Provide a description of the Design and Capabilities of **ExoSim**. 2) Provide instructions on how to install **ExoSim** and 3) Provide instructions on how to simulate ARIEL performance, using **ExoSim.**

## 1.2 Applicable Documents

| | |
|---|---|
| AD1 | ARIEL_CRDF_PL_AN_001, Performance_Analysis_Report, Issue: 1.5 |
| | |

## 1.3 Reference documents

| | |
|---|---|
| | |

## 1.4 Biblioraphy

| | |
|---|---|
| 1 | Sarkar, S., Papageorgiou A., Pascale E., *"ExoSim: elucidating noise in exoplanet transit spectroscopy observations using a generic numerical end-to-end simulator"*. In preparation, 2017 |
| 2 | Sarkar et al, *"Exploring the potential of the ExoSim simulator for transit spectroscopy noise estimation"*, 2016, SPIE, 9904 |

# 2 ExoSim: Design and Capabilities

This section focuses on a high-level description of the design and capabilities of **ExoSim**, while for the more detailed description of the algorithms employed by **Exosim**, the reader is referred to the refereed paper [B1, B2].

**ExoSim** is a generic, numerical end-to-end simulator of transit spectroscopy intended as open-access software. It permits the simulation of a time-resolved spectroscopic observation in either primary transit or secondary eclipse. The observational parameters can be adjusted, and the telescope and instrument parameters changed in a simple manner to simulate a variety of existing or proposed instruments. **ExoSim** is a tool to explore a variety of signal and noise issues that occur in, and might bias, transit spectroscopy observations, including the effects of the instrument systematics, correlated noise sources, and stellar variability. The simulations are fast, which allows **ExoSim** to be used for Monte Carlo simulations of such observations.

## 2.1 Design Overview

Being an end-to-end simulator, **`ExoSim`** has to be able to simulate the path of the signal fro the transiting Star-Planet system, through the telescope/instrument optics and onto the detectors.

**`ExoSim`** has been designed with three requirements in mind: It should be *Generic, Dynamic* and *Fast*.

1. Generic, in the sense that it should be able to *easily* simulate different system designs (different optics designs, photometer or spectrometer, grating or prism, different detector designs) making it efficient in comparing competing designs.
2. Dynamic, so that it can simulate many time-depended effects such as telescope pointing jitter or stellar variability(e.g. star spots), effects that are difficult or not simulated at all by static analytical or semi-analytical models.
3. Fast, in the sense that it should be able to complete a simulation in a matter of seconds or a few minutes, at worst, making it possible to perform Monte Carlo simulations.

Taking the above requirements into account, **`ExoSim`** can simulate different system designs, by providing components (focusing elements, mirrors, dispersion elements, detectors) which are extensively parametric – these components can be easily added or removed as elements of an XML structure, which is the main Input Configuration File to **`ExoSim`**. Along with the instrument description parameters, the Input Configuration File, also provides parameters to describe the astronomical source to be simulated, as well as other parameters like pointing jitter model, detector read-out modes (detector resets, non-destructive read-outs, read-out timings) and finally, providing convenience switches that can be used to turn individual effects (noise components) on and off.

With the simulation defined in the Input Configuration File, **`ExoSim`** produces a dynamic End-to-End simulation - from astronomical scene, through to astronomical foreground, through Telescope/Instrument and finally as the expected output (FITS files) to be used by the astronomer – through a series of modules. The overall structure of the **`ExoSim`** modules, can be seen in Figure1, while in the following section there is a description of the functionality and abilities of each individual module.

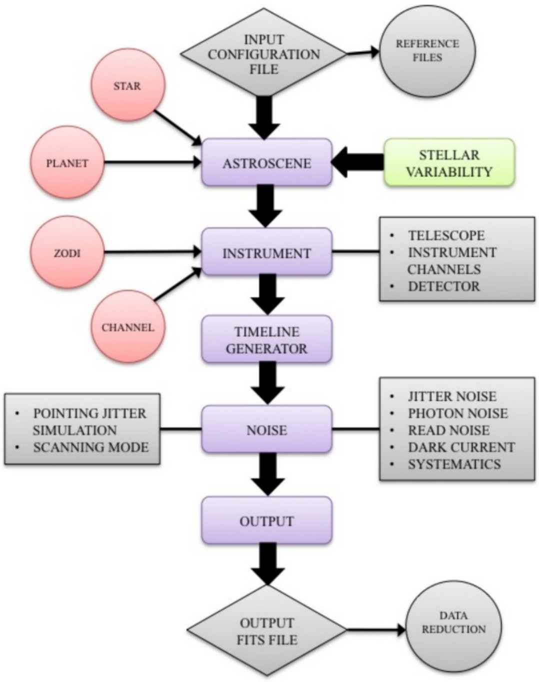


*Figure 1: ExoSim Design*

## 2.2 Modules Description

### 2.2.1 Astroscene Module

This module is responsible for initialising the extrasolar planetary system and computes wavelength-dependent light-curves, according to the relevant parameters provided in the Configuration file. Options in setting up the Astroscene include stellar SED (Black Body or pre-calculated Model), transit and planetary parameters (obtained from the Open Exoplanet Catalogue), limb darkening coefficients and whether the simulated transit is primary or secondary.

In addition, **ExoSim**, is able to include Stellar Variability in the simulations, either in the form of stellar pulsations (provided as a pre-calculated wavelength and time depended input) or in the form of start spots. In the case of star spots, **ExoSim** can internally calculate the transit light curves, given as input a list of spot positions, sizes and temperatures.

### 2.2.2 Instrument Module

This module is  responsible for calculating  and propagating the emission and transmission properties of all optical elements, the calculation of the PSF and the calculation of the pixel and wavelength coordinates for each channel.  The wavelength coordinates calculation can be calculated for both a prism or a diffraction grating, both of which can be parametrically defined in the Configuration file.

### 2.2.3 Time-line Generator Module

This module is responsible  for calculating the total number of frames of the simulation, assigning physical timestamps to them and identifying which of them correspond to events such as Detector Resets or Non-Destructive Readouts (NDR). To achieve this, the module calculates the total  time of an observation, based on the transit parameters, in combination with readout related parameters (number of NDRs per exposure, reset duration) define in the Configuration file.

With the frames synchronized to the transit time-line, the module then calculates the transit light-curve for all wavelengths, while taking in account other light-curve dependent effects such as occulted spots or general stellar pulsations.

### 2.2.4 Noise Module

With the light-curves calculated, the Noise module proceeds to apply instrumental effects. Each frame has the incident spectrum convolved with the system's PSF. If Spatial and/or Spectral jitter is enabled for the simulation, for each frame the PSF is shifted in the image plane, according to the Jitter model provided/described in the Configuration file.

Additional instrumental effects are then added, effects like Detector Readout Noise, Shot Noise, Dark current, pixel Quantum Efficiency distribution throughout the array.

### 2.2.5 Output Module

This module is responsible to creating the final output of **`ExoSim`**, one FITS file for each channel, with each FITS file containing a series of frames corresponding to all the NDRs of the detector array. During the observation. In addition, the FITS files contain all meta-data of all of the observation parameters used for the simulation – all the information required in order to test a data reduction pipeline is enclosed within the FITS files.

# 3 ExoSim Set-up

## 3.1 ExoSim Source Code

**`ExoSim`** is written in Python and the source code used for ARIEL Performance verification is included in the following archive file **`ExoSim_Ariel_20170209.tgz`** which accompanies this document. For reference purposes, the digits in the file name correspond to the date the code was committed.

## 3.2 Requirements

**`ExoSim`** is written in Python 2.7 and has been designed and tested only for Linux. Due to OS similarities, **`ExoSim`** has been known to function properly in Mac OSX.

**`ExoSim`** requires several python packages (e.g. **`Numpy`**, **`Scipy`**, python-quantities), which are usually installed by default (or easily installed) in most systems. If not present in your system, install as necessary. In addition to these common packages, **`ExoSim`** makes usage of scipy.weave, which means that a C++ compiler is also required.

Finally, **ExoSim** requires several science-specific python packages that will have to be downloaded from their public git repositories.

- PyTransit: Tools for exoplanet transit light curve modelling with Python
- ExoData: A python interface to the **OpenExoplanetCatalogue**
- OpenExoplanetCatalogue: A database of all discovered extra-solar planets.

### 3.3 Environment set-up

With all required packages installed/downloaded, add **ExoSim**, **ExoData** and **PyTransit** to **$PYTHONPATH**

```
PYTHONPATH=$PYTHONPATH:/path/to/ExoSim/
PYTHONPATH=$PYTHONPATH:/path/to/ExoData/
PYTHONPATH=$PYTHONPATH:/path/to/git/PyTransit/build/lib.linux-x86_64-2.7/
export PYTHONPATH
```

# 4 Running ExoSim

With **$PYTHONPATH** properly setup, ExoSim can then be run from anywhere in the system, as follows:

```
python runexosim.py exosim_ariel_mcr.xml
```

as long as long as an Exosim telescope description xml file is present in the directory. The Simulation Parameters File used for the Ariel Performance Model is called **exosim_ariel_mcr.xml** and is included within the ExoSim source code.

The output of ExoSim are fits files (one for each detector simulated) which are saved in the directory specified by the user in the xml file (**<ExoSimOutputPath>** parameter).

# 5 Modifying the Simulation Parameters

The Simulation Configuration File is an xml file with entries specifying parameters of the simulation like the Astroscene, Telescope Optics and Detectors. In-depth explanation of these parameters is considered beyond the scope of this document and it is assumed here that the descriptions contained in the xml file (also included here in the Appendix) are self-explanatory.

Telescope and Detector parameters in the are specific to the Ariel Model investigated and it is recommended that they are not altered. On the other hand, parameters within **<astroscene>**, **<timeline>** or **<noise>** can be easily changed to investigate different observation scenarios. Bellow is a list of parameters that are often changed:

| parameter | Typical values |
|---|---|
| astroscene/planet | Example values: 'gj1214b' faint star, '55 Cnc e' bright star, or any other planet that is contained in the Open Exoplanet Catalaog |
| astroscene/planetCR | Example values: Contrast Ratios corresponding to the planet used. Example contrast ratios included in ExoSim. Examples: "__path__/data/planetary/transmission/gj1214_specmodel.dat" or "__path__/data/planetary/emission/55cnc_ariel_emission2.txt" |
| timeline/exposure_time | Set the exposure time, in seconds. This should be changed in in accordance to the brightness of te star selected. For example, in the case of "55 Cnc e" exposure time should be small. Typical values: 88sec for "gj1214b" and 2sec for "55 Cnc b" **Limitation:** at the moment ExoSim can go down to 1, 2 seconds exposure time depending available memory. |
| timeline/frame_rate | Simulation time resolution in (Hz). Typical values: 2 for "gj1214b" and 100 for short exposure (55 Cnc e) |
| noise | This group contains different noise components (Jitter, Shot, Readout) that can be turned on and off |

# 6 Appendix

The following xml structure (included as a separate file along with the **`ExoSim`** source code) reflects the latest ARIEL design, in terms of optics, detectors, dispersion elements, pointing performance, channels etc. It is included here as a quick reference of the simulation parameters used for the ARIEL performance modelling.



```xml
<root>
  <title> ARIEL Goal Instrument Model - European detectors</title>
  <common>
    <logbinres val="1000" units="" comment="Resoving power for common spectral binnin. A linear binning wavelength integral is
estimated as wl_min/R"/>
    <wl_min val="0.45" units="micron" comment="Shortest wavelength in simularion used for binning input spectra"/>
    <wl_max val="10" units="micron" comment="Shortest wavelength in simularion used for binning input spectra"/>
    <ExoSimOutputPath val="~/ExoSimOutput" comment="output directory to store Sims" />
    <ConfigPath val = "__path__" />
  </common>

  <aocs>
    <PointingModel val="__path__/data/ariel/pointing_model_psd.csv" comment="Jitter PSD in units of deg**2/Hz" />
    <pointing_rms val="4.0e-5" units="deg" comment="RMS of desired jitter" />
    <pointing_scan_throw val="-8" units="arcsec" comment="Positive value enables scan-mode simulation. The value defines the throw of
the scan, in arcsec. The period of the scan is defined by the exposure time." />
  </aocs>

  <astroscene>
    <transit_is_primary val = "True"/>
    <apply_phase_curve val = "False" comment="If True, phase curve effects are added to the light curve"/>
    <use_planck_spectrum val = "False" comment="If True, use a Planck function for the stllar sed, otherwise use Phoenix"/>
    <planet val = 'gj1214b' />
    <planetCR val="__path__/data/planetary/transmission/gj1214_specmodel.dat" comment="planet/star contrast ratio"/>
    <OpenExoplanetCatalogue val = "~/git/open_exoplanet_catalogue/systems/" comment="Path to openexoplanet catalogue. Null string
downloads from URL."/>
    <StarSEDPath  val="__path__/data/star" />
    <StarLimbDarkening val="__path__/data/ldc_coeffs.dat" comment="Path to Model Limb Darkening coefficients table"/>
  </astroscene>

  <noise>
    <EnableSpatialJitter val="False" />
    <EnableSpectralJitter val="False" />
    <EnableShotNoise val="False" />
    <EnableReadoutNoise val="False" />
  </noise>

  <timeline>
    <exposure_time val="20.0" units="s" comment="time for one exposure containg set of NDRs" />
    <multiaccum val="2" units="" comment="number of NDRs per exposure" />
    <frame_rate val="20" units="1/s" comment="frame rate in SPS" />
    <nGND val='1' units="" comment="duration of reset event in units of 1/frame_rate" />
    <nNDR0 val='1' units="" comment="Integration time of first NDR in units of 1/frame_rate" />
    <nRST val = '1' units="" comment="Time lapse between last NDR and reset in units of 1/frame_rate" />
    <before_transit val="0.5" units="" comment="fraction of T14 for pre-transit observation" />
    <after_transit val="0.5"  units="" comment="fraction of T14 for post-transit observation" />
  </timeline>

<common_optics>
  <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="70" units="K" comment="M1"/>
  <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="70" units="K" comment="M2"/>
  <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="70" units="K" comment="M3"/>
  <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="70" units="K" comment="M4"/>
  <TelescopeEffectiveDiameter val="0.9" units="m" />
</common_optics>

<channel name="AIRS CH0" comment="AIRS CH0" type='spectrometer'>
  <is_spec val = "True" comment="if True, channel is a spectrometer, otherwise a photometer (no spectral dispersion)"/>
  <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d1_te_v1.csv"  val="60" units="K" comment="D1"/>
  <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d2_re_v1.csv"  val="60" units="K" comment="D2"/>
  <optical_surface type="lens" transmission="__path__/data/ariel_mcr/l_generic_te_v1.csv"  val="60" units="K" comment="CH0L1"/>
  <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M1"/>
  <optical_surface type="prism" transmission="__path__/data/ariel_mcr/p_airs_ch0_te_v1.csv" val="60" units="K" comment="CH0PRISM"/>
  <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M2"/>
```

```
  <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M3"/>
  <optical_surface type="lens" transmission="__path__/data/ariel_mcr/l_generic_te_v1.csv"  val="60" units="K" comment="CH0L2"/>
  <optical_surface type="correction" transmission="__path__/data/ariel_mcr/Ch0_correction.csv"  val="60" units="K"
comment="CH0corr"/>

  <slit_width val="17" units="" comment="Size of slit image in units of detector pixels"/>
  <dispersion path="__path__/data/ariel_mcr/CH0_prism_dispersion_angle_v1.csv" val="3840" units="micron" comment="Dispersion law. Val
is the reference pixel, i.e. the centre of the array. If Linear dispersion is defined, dispersion takes precedence"/>

  <array_geometry val="64, 512" units = "" comment="pixel count"/>
  <wfno val="13.2" units="" comment="Image space f/#" />
  <osf val="3" units="" comment="Oversample each detector pixel by this amount. Need to be changed" />
  <pix_offs val="1" units = "" comment="Detector pixel center offset" />
  <qe val="__path__/data/ariel/qe_v0.csv" comment="" />
  <qe_rms_matrix_file val = "None" comment="[Default 'None'] Path to CSV file containing matrix of QE of pixels."/>
  <plate_scale val="6.11e-5" units="deg" comment="In degrees per pixel" />
  <detector_pixel>
        <pixel_size val="15" units="micron" />
        <Idc val="30" units="1/s" comment="Detector dark current"/>
        <sigma_ro val="16.0" units="" comment="Detector redout noise in e-rms"/>
        <pixel_diffusion_length val="1.7" units="micron" />
  </detector_pixel>
</channel>


<channel name="AIRS CH1" comment="AIRS CH1" type='spectrometer'>
    <is_spec val = "True" comment="if True, channel is a spectrometer, otherwise a photometer (no spectral dispersion)"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d1_te_v1.csv"  val="60" units="K" comment="D1"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d2_te_v1.csv"  val="60" units="K" comment="D2"/>
    <optical_surface type="lens" transmission="__path__/data/ariel_mcr/l_generic_te_v1.csv"  val="60" units="K" comment="CH1L1"/>
    <optical_surface type="prism" transmission="__path__/data/ariel_mcr/p_airs_ch1_te_v1.csv" val="60" units="K" comment="CH1PRISM"/>
    <optical_surface type="lens" transmission="__path__/data/ariel_mcr/l_generic_te_v1.csv"  val="60" units="K" comment="CH1L2"/>
    <optical_surface type="filter" transmission="__path__/data/ariel_mcr/mwir_hp_te_v1.csv"  val="60" units="K" comment="CH1L2"/>
    <optical_surface type="correction" transmission="__path__/data/ariel_mcr/Ch1_correction.csv"  val="60" units="K"
comment="CH1corr"/>

    <slit_width val="13" units="" comment="Size of slit image in units of detector pixels"/>
    <dispersion path="__path__/data/ariel_mcr/CH1_prism_dispersion_angle_v1.csv" val="960" units="micron" comment="Dispersion law.
Val is the reference pixel, i.e. the centre of the array"/>
    <array_geometry val="64, 128" units= "" comment="pixel count, original 64 x 128"/>

    <wfno val="6.36" units="" transmission="The image space f/#" />
    <osf val="3" units="" comment="Oversample each detector pixel by this amount. Need to be changed" />
    <pix_offs val="1" units = "" comment="Detector pixel center offset" />
    <qe val="__path__/data/ariel/qe_v0.csv" comment="" />
    <qe_rms_matrix_file val = "None" comment="[Default 'None'] Path to CSV file containing matrix of QE of pixels."/>
    <plate_scale val="1.23e-4" units="deg" comment="In degrees per pixel" />
    <detector_pixel>
        <pixel_size val="15" units="micron"  />
        <Idc val="50" units="1/s" comment="Detector dark current"/>
        <sigma_ro val="16" units="" comment="Detector redout noise in e-rms"/>
        <pixel_diffusion_length val="1.7" units="micron" />
    </detector_pixel>
</channel>


<channel name="NIR Spec" comment="NIR Spec - see tech note for simulation of WFE usinf broadened PSF" type='spectrometer'>
    <is_spec val = "True" comment="if True, channel is a spectrometer, otherwise a photometer (no spectral dispersion)"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d1_re_v1.csv"  val="60" units="K" comment="D1"/>
    <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M1"/>
    <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M1"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d3_te_v1.csv"  val="60" units="K" comment="D3"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d5_te_v1.csv"  val="60" units="K" comment="D5"/>
    <optical_surface type="prism" transmission="__path__/data/ariel_mcr/p_airs_ch0_te_v1.csv" val="60" units="K"
comment="CH0PRISM"/>
    <optical_surface type="correction" transmission="__path__/data/ariel_mcr/NIRSpec_correction.csv"  val="60" units="K"
comment="NIRSpeccorr"/>

    <slit_width val="13" units="" comment="Size of slit image in units of detector pixels"/>
    <dispersion path="__path__/data/ariel_mcr/NIRSpec_prism_dispersion_angle_v1_R=50.csv" val="480" units="micron"
comment="Dispersion law. This version for R=50. Val is the reference pixel, i.e. the centre of the array"/>
    <array_geometry val="64, 160" units = "" comment="pixel count"/>
    <wfno val="38.68" units="" comment="Image space f/# doubled due to WFE, natural number is half this" />
    <osf val="3" units="" comment="Oversample each detector pixel by this amount. Need to be changed" />
    <pix_offs val="1" units = "" comment="Detector pixel center offset" />
    <qe val="__path__/data/ariel/qe_v0.csv" comment="" />
```

```
    <qe_rms_matrix_file val = "None" comment="[Default 'None'] Path to CSV file containing matrix of QE of pixels."/>
    <plate_scale val="4.84782e-05"units="deg" comment="In degrees per pixel" />
    <detector_pixel>
        <pixel_size val="18" units="micron" />
        <Idc val="1" units="1/s" comment="Detector dark current"/>
        <sigma_ro val="16.0" units="" comment="Detector redout noise in e-rms"/>
        <pixel_diffusion_length val="1.7" units="micron" />
    </detector_pixel>
</channel>

<channel name="FGS 2" comment="FGS 2 - see tech note for using simulated PSF with WFE" type='photometer'>
    <is_spec val = "True" comment="if True, channel is a spectrometer, otherwise a photometer (no spectral dispersion)"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d1_re_v1.csv"  val="60" units="K" comment="D1"/>
    <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M1"/>
    <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M1"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d3_te_v1.csv"  val="60" units="K" comment="D3"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d5_re_v1.csv"  val="60" units="K" comment="D5"/>
    <optical_surface type="correction" transmission="__path__/data/ariel_mcr/FGS_Red_correction.csv"  val="60" units="K"
comment="FGSRedcorr"/>

    <array_geometry val="80, 80" units = "" comment="pixel count"/>
    <wfno val="31.30" units="" comment="Image space f/#" />
    <osf val="3" units="" comment="Oversample each detector pixel by this amount. Need to be changed" />
    <pix_offs val="1" units = "" comment="Detector pixel center offset" />
    <qe val="__path__/data/ariel/qe_v0.csv" comment="" />
    <qe_rms_matrix_file val = "None" comment="[Default 'None'] Path to CSV file containing matrix of QE of pixels."/>
    <plate_scale val="3.60163e-05" units="deg" comment="In degrees per pixel" />
    <detector_pixel>
        <pixel_size val="18" units="micron" />
        <Idc val="1" units="1/s" comment="Detector dark current"/>
        <sigma_ro val="16.0" units="" comment="Detector redout noise in e-rms"/>
        <pixel_diffusion_length val="1.7" units="micron" />
    </detector_pixel>
</channel>


<channel name="FGS 1" comment="FGS 1 - see tech note for using simulated PSF with WFE" type='photometer'>
    <is_spec val = "False" comment="if True, channel is a spectrometer, otherwise a photometer (no spectral dispersion)"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d1_re_v1.csv"  val="60" units="K" comment="D1"/>
    <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M1"/>
    <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M1"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d3_re_v1.csv"  val="60" units="K" comment="D3"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d4_te_v1.csv"  val="60" units="K" comment="D5"/>
    <optical_surface type="filter" transmission="__path__/data/ariel_mcr/fgs_prime_lp_te_v1.csv" val="60" units="K"
comment="CH0PRISM"/>
    <optical_surface type="correction" transmission="__path__/data/ariel_mcr/FGS_Prime_correction.csv" val="70" units="K"
comment="FGSPrimecorr"/>

    <array_geometry val="80, 80" units = "" comment="pixel count"/>
    <wfno val="24.62" units="" comment="Image space f/#" />
    <osf val="3" units="" comment="Oversample each detector pixel by this amount. Need to be changed" />
    <pix_offs val="1" units = "" comment="Detector pixel center offset" />
    <qe val="__path__/data/ariel/qe_v0.csv" comment="" />
    <qe_rms_matrix_file val = "None" comment="[Default 'None'] Path to CSV file containing matrix of QE of pixels."/>
    <plate_scale val="4.57811e-05" units="deg" comment="In degrees per pixel" />
    <detector_pixel>
        <pixel_size val="18" units="micron" />
        <Idc val="1" units="1/s" comment="Detector dark current"/>
        <sigma_ro val="16.0" units="" comment="Detector redout noise in e-rms"/>
        <pixel_diffusion_length val="1.7" units="micron" />
    </detector_pixel>
</channel>


<channel name="VIS Phot" comment="VIS Phot - see tech note for using simulated PSF with WFE" type='photometer'>
    <is_spec val = "False" comment="if True, channel is a spectrometer, otherwise a photometer (no spectral dispersion)"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d1_re_v1.csv"  val="60" units="K" comment="D1"/>
    <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M1"/>
    <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M1"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d3_re_v1.csv"  val="60" units="K" comment="D3"/>
    <optical_surface type="dichroic" transmission="__path__/data/ariel_mcr/d4_re_v1.csv"  val="60" units="K" comment="D4"/>
    <optical_surface type="filter" transmission="__path__/data/ariel_mcr/nir_phot_lp_te_v1.csv" val="60" units="K"
comment="NIRPHOTLP"/>
    <optical_surface type="mirror" transmission="__path__/data/ariel_mcr/Mx_re_v1.csv" val="60" units="K" comment="CH0M1"/>
    <optical_surface type="correction" transmission="__path__/data/ariel_mcr/NIR_Phot_correction.csv"  val="60" units="K"
comment="NIRPhotcorr"/>
```

```
    <array_geometry val="64, 64" units = "" comment="pixel count"/>
    <wfno val="39.46" units="" comment="Image space f/#" />
    <osf val="3" units="" comment="Oversample each detector pixel by this amount. Need to be changed" />
    <pix_offs val="1" units = "" comment="Detector pixel center offset" />
    <qe val="__path__/data/ariel/qe_v0.csv" comment="" />
    <qe_rms_matrix_file val = "None" comment="[Default 'None'] Path to CSV file containing matrix of QE of pixels."/>
    <plate_scale val="2.85639e-05"  units="deg" comment="In degrees per pixel" />
    <detector_pixel>
        <pixel_size val="18" units="micron" />
        <Idc val="1" units="1/s" comment="Detector dark current"/>
        <sigma_ro val="16.0" units="" comment="Detector redout noise in e-rms"/>
        <pixel_diffusion_length val="1.7" units="micron" />
    </detector_pixel>
</channel>


</root>
```