FORS

Documentation of the
Data Reduktion Software

Chapter IPOL

VLT Instrument Consortium
27–August–1998

Document Number:	draft
Issue Number:	1.0
Issue Date:	27–August–1998

Prepared by:	Thomas Szeifert, Otmar Stahl, Sabine Möhler, Landessternwarte Heidelberg

Distribution List		
Recipient	Institute / Company	No. of Copies

Document Change Record			
Issue	Date	Sections affected	Reasons / Remarks
1.0	27–August–1998	first issue	
1.1	7–September–1998	`POLCOR/IPOL` and more	

Contents

Chapter 3

Imaging polarimetry

3.1 Introduction

Note

*The default values for any parameters, input data, etc. are given in (***bold font***) and keywords are marked by* `KEYWORD`.

The IPOL context is written for the reduction of imaging polarimetry data obtained with the slitlet mask.
It is assumed that the basic correction for bias, dark, and overscan are performed with standard MIDAS commands. There are no special IPOL averaging commands.
We assume that the detector is a CCD with the two images divided along columns.
Each command offers the choice between a first try and application to all science exposures of one object. We recommend that you use the "First try" option until you are satisfied with the results of your work (and our software) and then apply the parameters you found to all frames.
Since for each polarization measurement at least 4 science frames are necessary we decided to use a table system to keep track of all the data. This can be achieved by using the Data Organizer context (DO) and the command `SORT/IPOL`, which will put together all frames that have the same values (for given descriptors) as a reference frame. We assume that such a table exists when you start using the IPOL context. We will refer to it by `IPOL_TAB` (**ipol**). In case you create the table manually use the command `COPY/IPOL` to copy the angles of the retarder plates to the column `:RET_ANG` of the `IPOL_TAB` (**ipol**) table. An example for such a master table is given in Table 3.1.
In order to allow for automization all auxiliary frames/tables created by IPOL commands will get names derived from the input files. Depending on the stage of the reduction the new name will be generated by adding a prefix or a suffix: All commands used on two-dimensional frames ahead of `OFFSET/IPOL` (and the command itself) will add prefixes, while all commands after `OFFSET/IPOL` will add suffixes. The background of this rule is the fact that it should be possible for the derivation of the polarization parameters to sort together all images of one object (there may be several objects on each science frame). Thus, `OFFSET/IPOL` adds prefixes that describe the ray type (Ordinary/Extra-ordinary) and the number of the "ordinary" slitlet within the frame (01, 03, ...). To determine polarization parameters all images with prefix O01 and E01 (originating from different science exposures but being produced by the same object) will be sorted together (`XSORT/IPOL`, the same will be done

SC	FF_OUT	RET_ANG
fors0042.bdf	ff1.bdf	0.0
fors0043.bdf	ff2.bdf	22.5
fors0044.bdf	ff3.bdf	45.0
fors0045.bdf	ff4.bdf	67.5

Table 3.1: Example for an IPOL master table

Pre-/Suffix (keyword)	command	input file	example for input	output
FF_ (PRE_FFSC)	FLAT/IPOL	science frame	object0001.bdf	FF_object0001.bdf
POS (SUF_SKY)	POINT/IPOL	science frame	O01object0001.bdf	POSO01object0001.tbl
SKY (SUF_SKY)	MARK/IPOL	science frame	O01object0001.bdf	O01object0001SKY.tbl
SKY (SUF_SKY)	SKYFIT/IPOL	science frame	O01object0001.bdf	O01object0001SKY.bdf
SKY (SUF_SUB)	SKYFIT/IPOL	science frame	O01object0001.bdf	O01object0001SUB.bdf
C (SUF_CON)	SEE/IPOL	sky-subtracted science frame	O01object0001SUB.bdf	O01object0001SUBC.bdf
FLU (PRE_FLUX)	INTEGRATE-/IPOL	science frame	O01object0001.bdf	FLUO01object0001.tbl

Table 3.2: Examples for the name generation by IPOL commands

for O03 and E03, etc. Therefore the commands used after `OFFSET/IPOL` must use suffixes to generate the new names of images. Examples for name generation by the commands are given in Table 3.1.
`SKY/IPOL` also generates error frames, which are marked by the suffix ERR.

3.2 Location of slitlets and determination of offsets

The very first step after correcting bias, dark, and overscan is to find the edges of the slitlets. This is done by the command `LOCATE/IPOL`. This command locates the slitlets in a flat-field frame by searching for the maximum (normalized) gradient in a trace perpendicular to the direction of dispersion. Position and width of the trace are given by `SCAN_POS` (**0**) and `XBIN` (**20**). `FLATLIM(1)` (**0.15**) gives the minimum normalized gradient that must be exceeded, after median filtering the scan with a median of width `FLATLIM(2)` (**4**) and discarding scan values below `FLATLIM(3)` (**0**). The results are stored in the tables SLITL#, where # is the position angle of the retarder plate (e.g. SLITL2250.tbl), and the number of detected slitlets is written to `NSLIT` (**0**). The programs allow at most 100 slitlets. The table names are stored in the column `SLIT_TAB` (**:SLIT_TAB**) in `IPOL_TAB` (**ipol**). If the algorithm does not find any slitlets the chosen threshold (`FLATLIM(1)` (**0.15**)) may either be too high (above the intensity of the flats in the center of the frame) or too low (below bias value). Also the width (`FLATLIM(2)` (**4**)) may be chosen too high or too small. Typical values are between 0.1 and 0.2 and 3 and 5, respectively.
You can choose at this step whether you want to use the same flat-field for all position angles of the retarder plates or flats taken with the same position of the retarder plate (Same flat or Individual flats). Your choice will be written to the header of `IPOL_TAB` (descriptor: TBLGETFF).

3.3 Flat-fielding

The flat-field correction for all science frames in `IPOL_TAB` (**ipol**) is performed by `FLAT/IPOL`, which takes the flat fields from the column `FF_OUT` (**:FF_OUT**), the science frames from `SC` (**:SC**) and stores the results in the column `SC_FLAT` (**:SC_FF**). You may choose between using the same flat for all angles of the retarder plate or individual ones. The names of the flat-fielded science frames are created by adding the prefix `PRE_FFSC` (**FF_**) to the names of the input frames (e.g. FF_fors0042.bdf). The contents of `SC` is changed to :SC_FF.

SC	SC_EX	RET_ANG
fors0042.bdf	O01fors0042.bdf	0.00
fors0042.bdf	E01fors0042.bdf	0.00
fors0042.bdf	O03fors0042.bdf	0.00
fors0042.bdf	E03fors0042.bdf	0.00
⋮	⋮	⋮
fors0043.bdf	O01fors0043.bdf	22.50
fors0043.bdf	E01fors0043.bdf	22.50
fors0043.bdf	O03fors0043.bdf	22.50
fors0043.bdf	E03fors0043.bdf	22.50

Table 3.3: Example for an extracted IPOL table

3.4 Alignment and extraction of sub-frames

In order to combine the exposures taken with different angles of the retarder plates the offsets between ordinary and extra-ordinary rays as well as offsets between frames must be known. In the case of *many point sources and/or extended sources* they are derived with the command `OFFSET/IPOL`, which asks for interactive definition of point sources in the "ordinary" slitlets of one reference frames, using `CENTER/GAUSS`. From the estimated offsets `OFF` (**0,0**) between the two rays the program derives the "extra-ordinary" positions of the point sources, again using `CENTER/GAUSS` at the derived positions. Then `ALIGN/IMAGE` is used to get the correct transformation between ordinary and extra-ordinary images, allowing for offsets, rotation and scale changes. This transformation is applied by `REBIN/ROTATE`, transforming the extra-ordinary slitlets to the coordinate system of the respective ordinary slitlets. For all other frames the offsets between rays and between frames are determined using the table with positions of the reference frame. All slitlets are extracted and transformed to the coordinate system of the ordinary rays of the reference frame. The names of the extracted frames are generated by adding the prefix O (ordinary) or E (extra-ordinary), followed by the number of the respective ordinary slitlet, to the name of the input frame (e.g. O01fors0042, E01fors0042, O03fors0042, . . .). The names of the extracted frames are stored in a table named `IPOL_TAB`EX (**ipol**)EX. An example for such an extracted table is given in table 3.4.
In the case of *few point sources*, like standard stars, the commands `POINT/IPOL` and `INTEGRATE/IPOL` should be used (see Section 3.6).

3.5 Selection of sky regions and sky subtraction

The definition of background fields necessary for fitting the sky background is performed by `MARK/IPOL`, which takes the subframes extracted from the reference frame (stored in table `IPOL_TAB`EX (**ipolEX**)) and allows interactive definition of sky fields with the cursor rectangle. The sky regions are stored in tables, whose name is formed by adding the suffix `SUF_SKY` (**SKY**) to the name of the reference subframe from which they are derived (e.g. O01fors0042SKY.tbl, O03fors0042SKY.tbl). `MARK/IPOL` is intended for use *only* with extracted and aligned subframes. For the treatment of few point sources without extracting and aligning the subframes see section 3.6.
The actual sky fit and subtraction is then performed by the command `SKY/IPOL`, which makes use of `FIT/FLAT-_SKY`. The names of the output sky frames are created by adding the suffix `SUF_SKY` (**SKY**) to the names of the input frames (e.g. O01fors0042SKY.bdf, E01fors0042SKY.bdf, etc.). The names of the sky-subtracted frames are created by adding the suffix `SUB_SKY` (**SUB**) to the names of the input frames (e.g. O01fors0042SUB.bdf, E01fors0042SUB.bdf, etc.) and are stored in the column `SC_SKY` (**:SC_SKY**) of `IPOL_TAB`EX (**ipolEX**). `SKY/IPOL` also computes error frames, the names of which (e.g. O01fors0042SUBERR.bdf, E01fors0042-SUBERR.bdf, etc.) are stored in `IPOL_TAB`EX (**ipol**)EX, too.

CON_IMA	RET_ANG
O01fors0042SUBC	0.0
E01fors0042SUBC	0.0
O01fors0044SUBC	45.0
E01fors0044SUBC	45.0
O01fors0043SUBC	22.5
E01fors0043SUBC	22.5
O01fors0045SUBC	67.5
E01fors0045SUBC	67.5

Table 3.4: Example for a sorted extracted IPOL table

If you want to determine the polarization parameters two-dimensionally you use now the command `SEE/IPOL` which will create the same (worst) seeing on all sub-frames. The names of the convolved frames are created by adding the suffix `SUF_CON` (**C**) to the names of the input frames (e.g. O01fors0042SUBC.bdf, E01fors0042-SUBC.bdf, etc.) and are stored in the column `CONIMA` (**:CON_IMA**). `XSORT/IPOL` then sorts the extracted frames according to their values for the angle of the retarder plate, and separately for each slitlet. The names of the frames are stored in a table `IPOL_TAB`# (**ipol**)#, where # marks the number of the slitlet with the ordinary rays (e.g. `ipol01.tbl, ipol03.tbl`, etc.). For an example of such a table see Table 3.5.

3.6 Selection and integration of few point sources

3.6.1 Selection of sources

In case that you want to derive the polarization parameters of few point sources you can use the command `POINT/IPOL` to mark the positions of these objects. It will allow to interactively define the positions using `CENTER/GAUSS` on the "ordinary" subframes extracted from the reference frame (mode `A` of the command). The positions are stored in tables the names of which are created by adding the prefix POS to the names of the input reference frames (e.g. POSO01fors0042.bdf. POSO03fors0042.bdf). the names of the tables will be stored in the column `POSOBJ` (**:POSOBJ**) of the table `IPOL_TAB`EX (**ipolEX**).
You can use this command also to mark positions of stars without extracting and aligning the subframes (mode 'O' for Original frames), provided you know the offsets between the two rays sufficiently well *and there are no offsets between frames taken at different angles of the retarder plate*. In addition the point sources should be rather isolated since the sky background will be determined within the integration aperture by the command `INTEGRATE/STAR`. In this case the definition will be performed on the "ordinary" slitlets of the first science frame stored in column `SC` (**:SC**) of `IPOL_TAB` (**ipol**). The centering of the point sources in the extra-ordinary slitlets and on all othe frames is performed by using the interactively defined positions and the offsets between the two rays. The positions are stored in tables, separately for each frame and also for different rays (e.g. POSfors0042O.tbl, POSfors0042E.tbl, etc.). The names of the tables are written to columns `POSO` (**:POS_O**) and `POSE` (**:POS_E**) of the table `IPOL_TAB` (**ipol**).

3.6.2 Integration of sources

The command `INTEGRATE/IPOL` then performs an integration of the flux of the objects marked by `POINT/IPOL` and writes the results to tables whose names are generated by adding the prefix `PRE_FLUX` (**FLU**) to the names of the input frames. The names of the tables are stored in columns `FLUXO` (**:FLUXO**) and `FLUXE` (**:FLUXE**) of table `IPOL_TAB` (**ipol**) (original frames) resp. `IPOL_TAB`EX (**ipolEX**) (aligned frames). They are also collected into one master table `IPOL_TAB`FLUX (**ipol**)FLUX, where the column names are constructed from the ray-type (O/E) and the angles of the retarder plate. Flux columns are marked by appendix F, error column by appendix ERR.

Again you have to indicate whether the command shall be executed on aligned (mode 'A') or original (mode 'O') frames. !For aligned frames you should use sky-subtracted frames, since the integration !here will not provide any background correction. In the case of the reduction of the aligned and extracted images the Midas command `INTEGRATE/APERTURE` is used by `INTEGRATE/IPOL` to inetrate the flux of point sources. Because this command doesn't subtarct the background the sky has to be subtracted before integration of the source can be done.
For non-aligned original images a special integration routine `MAGNITUDE/IPOL` is called by `INTEGRATE/IPOL` which provides two methods of flux integration, point spread function fitting and aperture photometry. In this case the sky will be automatically subtracted before the source is integrated. `MAGNITUDE/IPOL` is first modelling the point spread function of the point sources with three types of functions[1]:

$$f(x,y) = \frac{F}{\left(1+\left(\frac{x-x_0}{Ax}\right)^2+\left(\frac{y-y_0}{Ay}\right)^2+2A_{xy}\frac{(x-x_0)(y-y_0)}{A_xA_y}\right)^\beta} \tag{3.1}$$

$$\boxed{\texttt{PSF}}=1 \qquad \beta = variable \tag{3.2}$$

$$\boxed{\texttt{PSF}}=2 \qquad \beta = 4 \tag{3.3}$$

$$f(x,y) = \frac{F}{arg^{\beta_1}+arg^{\beta_2}} \tag{3.4}$$

$$arg = \left(1+\left(\frac{x-x_0}{Ax}\right)^2+\left(\frac{y-y_0}{Ay}\right)^2+2A_{xy}\frac{(x-x_0)(y-y_0)}{A_xA_y}\right) \tag{3.5}$$

$$\boxed{\texttt{PSF}}=3 \qquad \beta_1 = 2 \tag{3.6}$$

$$\beta_2 = 7 \tag{3.7}$$

It is possible to combine the PSF-fitting with aperture photometry in the sence that within the inner circle of apertures the raw flux is used in the integraton, in the second ring around this circle the PSF-fit is integrated.

3.7 Polarimetry

Depending on the type of objects observed – point sources or extended objects – the polarimetry is done in a different way:
For point sources, the fluxes of the objects are integrated and the integrated fluxes are, for ordinary and extraordinary ray and for all angles of the retarder plate, stored in one table. For extended objects, all images of the slit mask are aligned to one common grid and treated completely two-dimensionally.
The principle method of the calculation of the Stokes parameter from the normalized flux differences is given in chapter SPOL. A minimum of four (two) observations at distinct position angles of the wave retarder plates are needed to calculate the linear (circular) polarization. Please note that as well as in the case of spectropolarimetric observations the four observations at plates angles (0°, 22.5°, 45°, 67.5°— half wave plate) for linear polarimetry are required to eliminate flat-field biases in the first order. In the case of circular polarimetric meassurements two observations (-45°and 45°— quarter wave plate) are required to eliminate both: the unproper flat fielding and the Stokes parameter cross talks due to the cromatism of the quarter wave plate, both in the first order approximation.
The computation of the Stokes parameters (I, Q, U (V)) and their errors is done with the commands `ISTOKES/TAB` or `ISTOKES/2D` for extracted or 2D-data, respectively. `ISTOKES/TAB` requires as input only a table with the tabulated flux in the ordinary and extraordinary beam, and writes the output (Stokes parameters and linear polarization) as new columns to the same table. `ISTOKES/2D` requires as input all aligned frames and writes the Stokes parameters (and their errors) in the same format as new frames. Both programs provide in the case of linear polarimetry the possibility to add the a constant zero angle during the calculation of the Stokes parameters Q and U. It is also possible to flip the rotation direction, if it is not given correctly.

[1]R. Racine 1996 — PASP 108, 699

In the case of extended image data, the linear polarization frames (angle and degree) are computed with the command `ILINPOL/2D`. The output frames can be plotted (or overplotted to e.g. a contour plot or a displayed image) with the commands `PLOT/IPOL` or `OVERPLOT/IPOL`, respectively. In any case for the extracted sources as well as for the 2D-images the degree of linear polarization is corrected by the offset which is expected from the non-linear distribution of meassurements (See again chapter SPOL for explanation of the method).
In the case of linear imaging polarimetry with broad band filters an additional source of depolarization occurs, due to the colour dependance of the half wave plate zero angle in the extended wave length range of the filter. From the (wave length dependend) tabulated values of the zero angle the intrument depolarization and the intrument depolarization rotated by the paralactic angle and the filter curve one can estimate the correction values by the command `POLCOR/IPOL` The zero angle ε_θ^i, the depolarisation due to the retarder plate cromatism $depol^i$ and the instrumental polarisation $inspol^i_{QU}$ in U and Q for a given filter curve $F^i(\lambda)$ can be expressed by the equations:

$$\varepsilon_\theta^i = \frac{\int F^i(\lambda)\varepsilon_\theta(\lambda)d\lambda}{\int F^i(\lambda)d\lambda} \tag{3.8}$$

$$depol^i = \frac{\int F^i(\lambda)depol(\lambda)d\lambda}{\int F^i(\lambda)d\lambda} \times \frac{\int cos(4(\varepsilon_\theta(\lambda)-\varepsilon_\theta^i))d\lambda}{\int F^i(\lambda)d\lambda} \tag{3.9}$$

$$inspol^i_{QU} = \frac{\int F^i(\lambda)inspol_{QU}(\lambda)d\lambda}{\int F^i(\lambda)d\lambda} \tag{3.10}$$

All data to characterize the instrument is read from Midas tables. In the case of the chromatic zero angle the program search for columns `:lambda` and `:epstheta`, in the case for the instrument depolarization for the columns `:lambda` and `:depol` and in the case of the instrumental spurious polarization the columns `:lambda`, `:qinspol` and `:uinspol`. If the paralactic angle is not given in the command line the program will call the `COMPUTE/AZIMUTH` command to transform the paralactic coordinates to azimuthal coordinates. Image data as well as data of extracted sources read from a Midas table can be corrected.

3.8 Data Structures

3.8.1 Tables

MIDAS tables are used to store most the information created and/or used by the IPOL context. A short description of the tables is given below; a detailed description of their contents can be found in Tables 3.6 and 3.7. Of some tables only one example is given, although several ones are created. For a detailed description refer to the descriptions of the cited commands. We give only those table columns which are relevant for the reduction. Some of the **MIDAS** commands used create additional columns, which are not used by the IPOL context.

$\boxed{\texttt{ipol\_tab}}$ is created by `SORT/IPOL` and contains all raw frames.

SLITL#.tbl is created by `LOCATE/IPOL` and contains the limits of the slitlets.

REFfors0042O01 is created by `OFFSET/IPOL` and contains the positions of the reference objects.

$\boxed{\texttt{ipol\_tab}}$**EX** is created by `OFFSET/IPOL` and used throughout most of the reduction. It contains the extracted and aligned frames.

O01fors0042SKY is created by `MARK/IPOL` and contains the positions of the sky regions marked for background fitting.

POSO01fors0042 is created by `POINT/IPOL` (mode 'A' and contains the positions of the point sources in the first subframe marked for integration.

POSfors0042O is created by `POINT/IPOL` (mode 'O' and contains the positions of the point sources in the "ordinary" slitlets marked for integration.

Context IPOL	
INITIALIZE/IPOL	[name]
SAVE/IPOL	name
SHOW/IPOL	[param]
HELP/IPOL	[keyword]
SET/IPOL	keyword=value
COPY/IPOL	[ipol_tab] [column]
FLAT/IPOL	[ipol_tab] [sc] [ff_out] [sc_flat] ffloc try
ILINPOL/2D	Stokes-Q Stokes-U pol-degree pol-angle
INTEGRATE/IPOL	mode [ipol_tab] [sc] radius ron,gain try
ISTOKES/2D	ipol_tab mode
ISTOKES/TAB	ipol_tab mode
LOCATE/IPOL	[ipol_tab] [ff_col] [flat_lim] [scan_pos] [xbin] [ff_loc]
MARK/IPOL	[ex_tab] [sc_ex]
OFFSET/IPOL	[ipol_tab] [sc] [slit_tab] [off] try
OVERPLOT/IPOL	pol-degree pol-angle
PLOT/IPOL	x-axis y-axis sc_x,sc_y,off_x,off_y x-label y-label pol-scale
POINT/IPOL	[mode] [ipol_tab] [sc] [slit_tab] [off] try
SEE/IPOL	[ex_tab] [sc_sky]
SKY/IPOL	[ex_tab] [sc_ex] [sc_sky] [sky_par] ron,gain try
SORT/IPOL	ipol_ass ref_ima [ipol_tab] des_list
XSORT/IPOL	[ipol_tab] [conima]
TUTORIAL/IPOL	

Table 3.5: Commands of the context IPOL

$\boxed{\texttt{ipol\_tab}}$**#** is created by `XSORT/IPOL` and contains the extracted, aligned and sky-subtracted images of slitlet #.

FLUO01fors0042SUB is created by `INTEGRATE/IPOL` (mode 'A' and contains the fluxes of the point sources in the first sky-subtracted "ordinary" subframe of fors0042.bdf.

FLUO01fors0042SUBERR is created by `INTEGRATE/IPOL` (mode 'A' and contains the errors of the point sources in the first sky-subtracted "ordinary" subframe of fors0042.bdf.

FLUfors0042O is created by `INTEGRATE/IPOL` (mode 'O' and contains the net fluxes and errors of the point sources in the "ordinary" slitlets of fors0042.bdf.

$\boxed{\texttt{ipol\_tab}}$**FLUX** is created by `INTEGRATE/IPOL` and contains the net flux and error of the point sources marked by `POINT/IPOL` of all frames.

Table `ipol_tab`		
Label	**Unit**	**Description**
SC	–	name of science frame
FF_OUT	–	name of averaged flat-field
SC_FF	–	name of flat-field corrected science frame
RET_ANG	–	name of angle of retarder plate
SLIT_TAB	–	name of slitlet table
POS_E	–	name of position table for extra-ordinary slitlets
POS_O	–	name of position table for ordinary slitlets
FLUXE	–	name of flux table for extra-ordinary slitlets
FLUXO	–	name of flux table for ordinary slitlets

Table SLITL#		
Label	**Unit**	**Description**
SLIT	–	sequential number of slitlet
YSTART	PIXEL	first row of slitlet
YEND	PIXEL	last row of slitlet
RAY_TYP		ray type (**o**rdinary / **e**xtra-ordinary

Table `ipol_tab`EX		
Label	**Unit**	**Description**
SC	–	name of science frame
SC_EX	–	name of extracted and aligned sub-frame
RET_ANG	–	angle of retarder plate
SKYTAB	–	name of tables with sky position
SC_SKY	–	name of sky-subtracted science frame
SC_SKYERR	–	name of error frame of sky-subtracted science frames
POSOBJ	–	name of position table of point sources
FLUXE	–	name of flux table of point sources in “extra-ordinary” sub-frames
FLUXO	–	name of flux table of point sources in “ordinary” sub-frames
SEE_X	pixel	seeing FWHM in x
SEE_Y	pixel	seeing FWHM in y
SIG_X	pixel	FWHM of convolution gaussian in x
SIG_Y	pixel	FWHM of convolution gaussian in y
CON_IMA	–	name of convolved science frame

Table 3.6: Tables created and/or used by the IPOL context

Table REFfors0042O01		
Label	**Unit**	**Description**
XCEN	WORLD COORD	x-position of reference object
YCEN	WORLD COORD	y-position of reference object

Table O01fors0042SKY		
Label	**Unit**	**Description**
Xstart	World Coord	x-start of sky region
Ystart	World Coord	y-start of sky region
Xend	World Coord	x-end of sky region
Yend	World Coord	y-end of sky region
Xstartpix	Frame Pixels	x-start of sky region
Ystartpix	Frame Pixels	y-start of sky region
Xendpix	Frame Pixels	x-end of sky region
Yendpix	Frame Pixels	y-end of sky region

Tables POSO01fors0042 - POSfors0042O		
Label	**Unit**	**Description**
X_COORD	WORLD COORD	x-coordinate of point source
Y_COORD	WORLD COORD	y-coordinate of point source

Table ipol_tab#		
Label	**Unit**	**Description**
CON_IMA	–	name of sky-subtracted convolved subframe
RET_ANG	–	angle of retarder plate

Table 3.7: Tables created and/or used by the IPOL context (cont'd)

Table FLUO01fors0042SUB - FLUO01fors0042SUBERR		
Label	**Unit**	**Description**
X_COORD	WORLD COORD	x-position of aperture
Y_COORD	WORLD COORD	y-position of aperture
RADIUS	USER UNIT	radius of aperture
FLUX	–	integrated flux

Table FLUfors0042O		
Label	**Unit**	**Description**
X_COORD	WORLD COORD	x-position of aperture
Y_COORD	WORLD COORD	y-position of aperture
FLUX	–	integrated flux of source
BACKGROUND	–	background estimate
DIAMETER	WORLD COORD	diameter of aperture
FLUXSUB	–	net flux
FLUXERR	–	flux error

Table ipol_tab FLUX		
Label	**Unit**	**Description**
O0000F	–	net flux of “ordinary” sources for angle 0
O0000ERR	–	flux error of “ordinary” sources for angle 0
E0000F	–	net flux of “extra-ordinary” sources for angle 0
E0000ERR	–	flux error of “extra-ordinary” sources for angle 0
O2250F	–	net flux of “ordinary” sources for angle 22.5
⋮	⋮	⋮
O4500F	–	net flux of “ordinary” sources for angle 45
⋮	⋮	⋮
O6750F	–	net flux of “ordinary” sources for angle 67.5
⋮	⋮	⋮

Table 3.8: Tables created and/or used by the IPOL context (cont’d)

Table 3.9: Keywords used in context IPOL

Name	Default value	Contents
Keywords for flat-field related commands		
FLATLIM/R/1/3	0.15,4,0	detection threshold for slitlet location
NSLIT/I/1/1	0	total number of slitlets
FF_LOC/C/1/1	S	same flat or individual flats for slitlet detection
Keywords for file administration		
N_SC/I/1/1	0	total number of science frames
REF_RET/R/1/1	0	retarder plate angle of reference frame
SC_STRT/I/1/16	0	first rows in table SPOL_TAB for science frames
SC_END/I/1/16	0	last rows in table SPOL_TAB for science frames
ZERO_ANG/R/1/1	0.	zero angle position of retarder plate
Keywords for table and frame names		
IPOL_TAB/C/1/60	ipol	table with science frames and all calibration frames and tables
Keywords for columns storing tables/frames		
CONIMA/C/1/16	:CON_IMA	column with convolved sc frames
FF/C/1/16	:FF	column with flat fields
FF_OUT/C/1/16	:FF_OUT	column with averaged flat fields
FLUXE/C/1/16	:FLUXE	column with 'extra-ordinary' flux tables
FLUXO/C/1/16	:FLUXO	column with 'ordinary' flux tables
POSE/C/1/16	:POS_E	column for position tables (Extra. ray)
POSO/C/1/16	:POS_O	column for position tables (Ordin. ray)
POSOBJ/C/1/16	:POSOBJ	column for position tables (Ordin. ray)
SC/C/1/16	:SC	column with raw science frames
SC_EX/C/1/16	:SC_EX	column with extracted science frames
SC_FLAT/C/1/16	:SC_FF	col. with flat-fielded science frames
SC_SKY/C/1/16	:SC_SKY	column with sky-subtracted sc frames
SKY/C/1/16	:SKY	column with sky frames
SKYTAB/C/1/16	:SKYTAB	column with sky tables
SLIT_TAB/C/1/16	:SLIT_TAB	column with slitlet tables

Table 3.10: Keywords used in context IPOL (cont'd)

Name	Default value	Contents
Keywords for parameters		
INT_MODE/C/1/1	O	integration mode (Aligned/Original)
OFF/I/1/2	0,0	offset between rays (x,y)
SCAN_POS/D/1/1	0	center for scan (locate/ipol) in world coordinates
SKYPAR/I/1/6	1,1	parameters for sky fit
XBIN/I/1/1	20	Binning in X for slitlet detection
DIM_G/I/1/2	21,21	dimensions of convolution gaussian
Keywords for deriving names		
PRE_FFSC/C/1/3	FF_	prefix for flat-fielded science frames
PRE_FLUX/C/1/3	FLU	prefix for tables created by integ/ipol
SUF_CON/C/1/1	C	suffix for convolved sc frames
SUF_SKY/C/1/3	SKY	suffix for sky frames/tables with sky regions
SUF_SUB/C/1/3	SUB	suffix for sky-subtracted frames
Keywords for CCD parameters		
CCDPAR/C/1/60		read-out-noise/conversion factor of CCD

3.9 IPOL Cookbook - A typical session

Note

*The default values for any parameters, input data, etc. are given in (***bold font***) and keywords are marked by* `KEYWORD`.

For most commands two options exist: **F**irst try or **A**ll frames. Use the **F** option until you are satisfied with the results and then extend this solution to all frames with **A**

3.9.1 Starting the whole thing

Before you start the IPOL context you should have done the following preparations:

- average your bias frames
- average your dark frames
- average all flat-fields of the same setup
- correct the object frames for bias, overscan, and dark

In addition collect all frames you need in one table, using the Data Organizer package. Then you may activate the IPOL context with

Midas . . . > `SET/CONTEXT ipol`

If you want to restore the parameters of a former session, that have been saved with

Midas . . . > `SAVE/IPOL previous`

type

Midas . . . > `INIT/IPOL previous`

If you want to restore the default values (as is automatically done by `SET/CONTEXT IPOL` type

Midas . . . > `INIT/IPOL`

A demonstration of the package can now be obtained with

Midas . . . > `TUTORIAL/IPOL`

which runs interactively and also allows to look at selected parts of the package. In order to create all necessary test data you should run it once completely.
We tried to implement reasonable defaults whenever possible so you should at first try to use these until you got somewhat accustomed to the commands. The first step of the IPOL reduction consists of sorting all frames of one object (or one field) together. All later routines assume that the frames you want to reduce are collected in one table, together with the respective calibration images. In addition they should always be available in multiples of four with the correct angles of the retarder plates for linear polarization measurements (i.e. 0, 22.5, 45, 67.5; then 90, 112.5, 135, 157.5; etc. or offsets of this size from a reference angle, define by `ZERO_ANG` (**0**)). For linear polarization frames with angles 0 and 45 (or offsets that size with respect to a reference angle) should be present.
The sorting is performed by

Midas . . . > `SORT/IPOL ipol_ass fors0042 ipol des_tab`

An example for such a master table (after the whole reduction has been prformed) can be found in Table 3.9.1.

Table : ipol

Sequence	SC	FF_OUT	RET_ANG	SLIT_TAB	SC_FF
1	fors0042.bdf	flat1.bdf	0.0	SLITL0000.tbl	FF_fors0042
2	fors0043.bdf	flat2.bdf	22.5		FF_fors0043
3	fors0044.bdf	flat3.bdf	45.0		FF_fors0044
4	fors0045.bdf	flat4.bdf	67.5		FF_fors0045
⋮	⋮	⋮	⋮	⋮	⋮
Sequence	POS_O	POS_E	FLUX_O	FLUX_E	
1	POSfors0042O	POSfors0042E	FLUfors0042O	FLUfors0042E	
2	POSfors0043O	POSfors0043E	FLUfors0043O	FLUfors0043E	
3	POSfors0044O	POSfors0044E	FLUfors0044O	FLUfors0044E	
4	POSfors0045O	POSfors0045E	FLUfors0045O	FLUfors0045E	
⋮	⋮	⋮	⋮	⋮	

Table 3.11: Typical IPOL table. The columns :SC and :FF_OUT contain input frames (and are taken from the input table created by the Data Organizer), the columns :SLIT_TAB, :SC_FF, :POS_O, :POS_E, :FLUXO, :FLUXE are created by IPOL commands

3.9.2 Locating slitlets and flat-field correction

Locating slitlets

The next step is the location of the slitlets with

Midas ... > `LOCATE/IPOL ? ?  ?  ?  S`

This command will take the flats from column `FF_OUT` (**:FF_OUT**) of the table ipol and look for the slitlets in a trace averaged along the dispersion direction. The flag `S` means that you want to use the same flat for all science frames (in contrast to `I`ndividual flats). The parameters for the slitlet search, like center and width of scan and threshold parameters are taken from the keywords `SCAN_POS` (**0** = center of frame along x-axis) and `FLATLIM` (**0.15,4,0**). The limits of the slitlets will be stored in tables SLITL#, where # is the resp. position angle of the retarder plate. The names of the tables will be stored in the column `SLIT_TAB` (**:SLIT_TAB**) of ipol. The number of detected slitlets is written to `NSLIT` (**0**). This command will also store the ray type (`O`rdinary/`E`xtra-ordinary) of each slitlet in a column, assigning `O` always to the slitlets with odd numbers and `E` to those with even numbers. An example for a slitlet table is given below:

:SLIT	sequential number of slitlet
:YSTART	first row of slitlet (world coordinates)
:YEND	last row of slitlet (world coordinates)
:RAY_TYP	type of rays (Ordinary/Extra-orinary)

If you are not satisfied with the result have a look at your flat field and try with new parameters

Midas ... > `SET/IPOL flatlim=threshold,width,limit scan_pos=xpos xbin=width`

Flat-field correction

The flat-field correction can be performed by

Midas ... > `FLAT/IPOL ? ?  ?  ?  S A`

which will take the first flat-field stored in column `FF_OUT` (**:FF_OUT**) of `IPOL_TAB`.tbl (**ipol**).tbl (because of the flag S for Same flat for all frames) and didvide all (flag A for All frames) science frames stored in column

Table : ipolEX

Sequence	SC_FF	SC_EX	RET_ANG	SKY_TAB
1	FF_fors0042.bdf	O01fors0042.bdf	0.0	O01fors0042SKY
2	FF_fors0042.bdf	E01fors0042.bdf	0.0	
3	FF_fors0042.bdf	O03fors0042.bdf	0.0	O03fors0042SKY
4	FF_fors0042.bdf	E03fors0042.bdf	0.0	
5	FF_fors0043.bdf	O01fors0043.bdf	22.5	
⋮	⋮	⋮	⋮	

Sequence	SC_SKY	SC_SKYERR	POSOBJ
1	O01fors0042SUB	O01fors0042SUBERR	POSO01fors0042
2	E01fors0042SUB	E01fors0042SUBERR	
3	O03fors0042SUB	O03fors0042SUBERR	POSO03fors0042
4	E03fors0042SUB	E03fors0042SUBERR	
5	O01fors0043SUB	O01fors0043SUBERR	
⋮	⋮	⋮	

Sequence	FLUX_O	FLUX_E
1	FLUO01fors0042SUB	
2		FLUE01fors0042SUB
3	FLUO03fors0042SUB	
4		FLUE03fors0042SUB
5	FLUO01fors0043SUB	
⋮	⋮	⋮

`SC` (**:SC**) by it. The names of the results are stored in column `SC_FLAT` (**:SC_FF**) and are created by adding the prefix `PRE_FFSC` (**FF_**) to the names of the input science frames. Also the contents of the keyword `SC` are changed to **:SC_FF**.

3.9.3 Extraction and alignment of subframes

In order to allow a two-dimensional determination of polarization parameters all images of one slitlets must be transformed to a common coordinate system. This is achieved with

Midas ... > `SET/IPOL off = xoff,yoff,`

which sets the offsets between ordinary and extra-ordinary rays. Then use

Midas ... > `OFFSET/IPOL ? ?  ?  ?  A`

which takes the frames stored in column `SC` (**:SC**) of `IPOL_TAB` (**ipol**), the slitlet tables from column `SLIT_TAB` (**:SLIT_TAB**) and the offsets from `OFF` (**0,0**). It then takes the first image stored in `SC` as reference image and allows interactive definition of point sources in the slitlets containing ordinary rays. The position in the "extra-ordinary" slitlets and on all other frames stored in `SC` is then performed by re-centering the objects, taking into account the offsets between the rays stored in `OFF`. The offsets between frames are assumed to be small (< **??** pixels).
The subframes (= slitlets) are then extracted and transformed to the coordinate system of the respective "ordinary" slitlet in the reference frame. The names of the output frames are stored in column `SC_EX` (**:SC_EX**) of a new table `IPOL_TAB`EX (**ipol**)EX and created by adding the ray-type (O/E) and the slitlet number of the corresponding "ordinary" slitlet (01, 03, 05, ...) as prefix to the name of the input frame (e.g. O01fors0042.bdf, E01fors0042.bdf, etc.). A typical extracted IPOL table (after the complete reduction) can be found in Table 3.9.3.
For the alignment the commands `ALIGN/IMAGE` and `REBIN/ROTATE` are used.

3.9.4 Sky definition and subtraction

Sky background regions can be marked with

Midas ... > `MARK/IPOL`

which takes the reference slitlets and allows the interactive definition of sky regions with a cursor rectangle. The regions are stored in tables, whose names are craeted by adding the suffix `SUF_SKY` (**SKY**) to the names of the input frames (e.g. O01fors0042SKY.tbl, O03fors0042SKY.bdf, etc.) and are stored in column `SKYTAB` (**:SKYTAB**) of table `IPOL_TAB`EX (**ipol**)EX.
The actual sky fit and subtraction is done by

Midas ... > `SKY/IPOL ? ? ? ? read-out-noise,gain A`

which takes the input images from column `SC_EX` (**:SC_EX**) (e.g. E01fors0043.bdf) of `IPOL_TAB`EX (**ipol**)EX and fits the sky background according to the regions defined in the corresponding sky table (e.g. O01fors0042-SKY.bdf), using `FIT/FLAT_SKY`. The resulting sky is stored in a frame, whose name is constructed by appending the suffix `SUF_SKY` (**SKY**) to the name of the input frame (e.g. E01fors0043SKY.bdf) and the sky subtracted frame is named by appending the suffix `SUF_SUB` (**SUB**) to the name of the input frame (e.g. E01fors0043SUB.bdf). The names of sky-subtracted frames are stored in column `SC_SKY` (**:SC_SKY**) of `IPOL_TAB`EX (**ipol**)EX. The parameters `read-out-noise` and `gain` are necessary to correctly determine the errors of the sky-subtracted frames, which are stored in frames, whose names are created by appending ERR to the names of the sky-subtracted frames (e.g. E01fors0043SUBERR.bdf) and stored in column `SC_SKY`ERR (**:SC_SKY**)ERR.

3.9.5 Final preparation for computation of polarization

Two-dimensional approach

If you want to determine the polarization parameters two-dimensionally you first have to achieve the same (worst) seeing on all frames with

Midas ... > `SEE/IPOL.`

The names of the convolved frames are constructed by appending the suffix `SUF_CON` (**C**) to the name of the input frame (e.g. E01fors0043SUBC.bdf). The names of convolved frames are stored in column `CONIMA` (**:CON_IMA**) of `IPOL_TAB`EX (**ipol**)EX. Then you sort the images of each "ordinary" slitlet together with

Midas ... > `XSORT/IPOL`

which takes all the sky-subtracted, convolved subframes of one "ordinary" slitlet and stores them in a table `IPOL_TAB`# (**ipol**)#, where # is the number of the slitlet (e.g. ipol01.tbl, ipol03.tbl, etc.). It also sorts them according to the angles of the retarder plate in a way as is necessary for the further treatment.

Treatment of few point-sources

If you want to determine the polarization parameters of a few point sources you have two possibilities, depending on your interest:

Few (foreground) point sources in aligned images This approach should also be used if the point sources are not very isolated. Here you mark the positions with

Midas ... > `POINT/IPOL A ? ? A`

where the first 'A' marks the fact that you work on aligned sub-images. The command will allow interactive definition of point source position on all (second 'A') reference subframes stored in `SC_EX` (**:SC_EX**) of `IPOL_TAB`EX (**ipol**)EX. The marked positions will be stored in tables, whose names are created by adding the prefix `POS` to the names of the reference subframes (e.g. POSO01fors0042.bdf) and are stored in column `POSOBJ` (**:POSOBJ**) of `IPOL_TAB`EX (**ipol**)EX.

With

Midas ... > `INTEGRATE/IPOL A ? ? integr_radius read_out_noise,gain A`

you can then integrate the flux of the point sources on all (last 'A') aligned (first 'A'), sky-subtracted subframes. The program takes the frames stored in `SC_SKY` (**:SC_SKY**) of table `IPOL_TAB`EX (**ipol**)EX to integrate the flux, which is necessary, since the integration (`INTEGRATE/APERTURE`) does not perform any background correction. The error of the flux is determined by integrating on the error frames. The results are stored in tables, whose names are created by adding the prefix `PRE_FLUX` (**FLU**) to the names of the input frames (e.g. FLUO01fors0043SUB.tbl, FLUO03fors0043SUB.tbl, etc.) and are stored in columns `FLUXO` (**:FLUXO**) and `FLUXE` (**:FLUXE**) of `IPOL_TAB`EX (**ipol**)EX. They are also collected into one master table `IPOL_TAB`FLUX (**ipol**)FLUX, where the column names are constructed from the ray-type (O/E) and the angles of the retarder plate. Flux columns are marked by appendix F, error column by appendix ERR (for an example see below).

`:O0000F`	"ordinary" net flux for angle 0
`:O0000ERR`	"ordinary" error for angle 0
`:E0000F`	"ordinary" net flux for angle 0
`:E0000ERR`	"ordinary" error for angle 0
`:O2250F`	"ordinary" net flux for angle 22.5
`:O2250ERR`	"ordinary" error for angle 22.5
`:E2250F`	"ordinary" net flux for angle 22.5
`:E2250ERR`	"ordinary" error for angle 22.5
`:O4500F`	"ordinary" net flux for angle 45
`:O4500ERR`	"ordinary" error for angle 45
`:E4500F`	"ordinary" net flux for angle 45
`:E4500ERR`	"ordinary" error for angle 45
`:O6750F`	"ordinary" net flux for angle 67.5
`:O6750ERR`	"ordinary" error for angle 67.5
`:E6750F`	"ordinary" net flux for angle 67.5
`:E6750ERR`	"ordinary" error for angle 67.5

Few isolated point sources in non-aligned images In this case you do not have to perform the alignment, since you are dealing with few, isolated point sources (e.g. standard stars) and know the offsets between the rays well enough. Then you use

Midas ... > `POINT/IPOL O ? ? A ? ?`

which takes the first science frame stored in column `SC` (**:SC**) and the slitlet tables stored in column `SLIT_TAB` (**:SLIT_TAB**) of `IPOL_TAB` (**ipol**). It allows interactive definition of point sources in the "ordinary" slitlets and uses the offsets from `OFF` (**0,0**) to recenter the objects on the "extra-ordinary" slitlets of the first frame and on all ('A') other frames. The positions are stored in tables, whose names are created by adding the prefix POS to the names of the input frames and appending a suffix indicating the ray type (e.g. POSfors0042O.tbl, POSfors0042E.tbl, etc.). The names of the tables are stored in columns `POSO` (**:POSO**) and `POSE` (**:POSE**) of `IPOL_TAB` (**ipol**).
The integration of the flux, background subtraction and computation of errors is the performed by

Midas ... > `INTEGRATE/IPOL O ? ? radius,step,ref_magnitude read_out_noise,gain A`

which takes the frames stored in `SC` (**:SC**) of `IPOL_TAB` (**ipol**), the position tables stored in columns `POSO` (**POSO**) and `POSE` (**POSE**) of the same table and applies the command `INTEGRATE/STAR`. The results are stored in tables, whose names are created by adding the prefix `PRE_FLUX` (**FLU**) to the names of the input frames and appending a character indodcating the ray type (e.g. FLUfors0043O.tbl, FLUfors0043E.tbl, etc.) and are stored in columns `FLUXO` (**:FLUXO**) and `FLUXE` (**:FLUXE**) of `IPOL_TAB` (**ipol**). They are also collected into one master table `IPOL_TAB`FLUX (**ipol**)FLUX, where the column names are constructed from the ray-type (O/E) and the angles of the retarder plate (e.g. :O0000F, :E2250ERR, etc.). Flux columns are marked by appendix F, error columns by appendix ERR.

3.9.6 Computation of Stokes parameters and polarization

Now the data are ready for the computation of the Stokes parameters and the polarization (degree and angle). For the one-dimensional treatment of point sources, all is done in one single step:

Midas ... > `ISTOKES/TAB ipoltab lin`

computes from the fluxes in columns `:O000F` ... `:O9000F` and their errors in columns `:O0000ERR` ... `:O9000-ERR` the Stokes parameters I, Q and U. The output is written to columns with the name of the corresponding parameter (e.g. :I and :Ierr).
For 2D-objects, the input is a table with the required input images listed in one column .

Midas ... > `ISTOKES/2D IPOL_TAB# LIN`

requires eight aligned frames (ordinary and extra-ordinary rays with four positions of the retarder plate (λ/2))

Midas ... > `ISTOKES/2D IPOLTAB# CIR`

computes the Stokes parameter spectra Stokes I and V. In this case only four aligned frames (ordinary and extra-ordinary rays with two positions of the retarder plate (λ/4)) are needed. For both modes, the error frames are also computed. The characters Q, U, I or V are appended to the name of the input table to form the name of the resulting frames. In addition, the character E is appended for the error frames.
With the command `POLCOR/IPOL` you can then correct the linear Stoekes parameters for the cromatism of the retarder plate, the instrumental depolarization and the instrumental spurious polarisation:

Midas ... > `POLCOR/IPOL ? IPOL_TAB#.tbl epstheta depol inspol filter 0.0`

The chromatism of the halfwave plate, the depolarisatiom and the instrumental polarisation is read from the input tables epstheta depol and inspol and with the filter curve "filter" the constant correction terms are integrated.
Assuming that the `IPOL_TAB` is set to **object**, the command

Midas ... > `ILINPOL/2D object01Q object01U degree angle`

then computes degree and angle of linear polarization and their errors. Output is to the frames degree, degreeE, angle and angleE, respectively. With the command

Midas ... > `PLOT/IPOL degree angle`

these frames are plotted to the currently assigned Midas graphics device and with the graphics options set with the Midas command SET/GRAPH.
The Midas commands

Midas ... > `plot/conto intens.bdf ?  0.3,0.3 100,200,300,400,500,600,700,`
`800,900`

Midas ... > `ove/ipol degree angle`

may be used to overplot the polarization vectors to a contour plot.
Use the commands

Midas ... > `ass/dis d,0`

Midas ... > `load/ima intens`

Midas ... > `ass/gra d,0`

Midas ... > `plot/ax`

Midas ... > `over/ipol degree angle`

to get an overplot in the display channel 0.