DEVELOPERS NOTES - VERSION 1_0 01 October 2011

1.1 control data (c)

Parameter	Descriptor
c.rbname	Unique name of Reduction block RB
c.pdir	Project directory path and name
c.fiberid	999:extract spectra of all valid fibers ###:extract spectra for single fiber no. ###
c.dfiber1	Fiber number of first fiber lost in gap between two halves of CCD
c.dfiber2	Fiber number of second lost fiber (0 if no second fiber lost)
c.skyoset	0:A median sky spectrum is found from the sky fibers in the science file. 1:Sky spectra of individual fibers are found from offset sky observations.
c.skytype	0:No sky subtraction is made 1:The sky spectrum is subtracted from the target spectrum. 2:The median sky spectrum is scaled prior to subtraction. 3:The median sky spectrum is scaled and to output masked over sky line
c.wlmin	Lower cut-off for common wavelength range
c.wlmax	Upper cut -off for common wavelength range
c.fibrattn	0:Fiber attenuations are set uniformly to 1 1:Attenuations calculated from the mean line counts in the offset image. 2:Attenuations are read file fiber_attn.txt in directory control_standards/. 3:Attenuations are read file fiber_attn.txt in directory project log directory.
c.profile0:	A top hat profile is used producing non optimal extraction 1:The spatial profile is found from the flat image for optimal extraction.
c.flattype	0:Output spectra are as defined by extraction process (target/flat) 1:Output spectra are multiplied by the flat lamp profile
c.nscatter	0:No compensation is made for scattered light 1:Background level due to scattered light is subtracted before extraction.
c.wltype	0:Output spectra on a log wavelength scale 1:Output spectra on a linear wavelength scale
c.xbin	Size x bin of CCD image in raw data file
c.ybin	Size y bin of CCD image in raw data file
c.csize	Plot character sizing – scale relative to standard size
c.maskcut	Dark current cut-off level for a mask calculation
c.maskmax	Upper limit cut off values for mask calculation
c.masklev	Fraction of pixels masked when autoranging mask cut off level
c.circlev	Number of profiles of circ file image used for line tracing
c.circsmth	Smoothing applied to circ file profiles for line tracing
c.circzero	Offset of zero line (as percent of the peak) for line tracing
c.lwide	Parameter setting width of CCD trace across line profile
c.ysmooth	Smoothing parameter used to remove pixel to pixel noise
c.boxwide	Half-width of line in x pixels for top hat extraction
c.staritn	Number of iterations of mask data for optimal extraction
c.nwl	Number of steps in common wavelegth range
c.flatsmth	Number of pixel smoothing flat spectrum to produce smooth flat
c.pdata	Subdirectory containing raw data files
c.pint	Subdirectory containing intermediate pipeline data files
c.pspec	Subdirectory containing output spectra
c.plog	Subdirectory containing log files
c.ptemp	Subdirectory containing temporary output files
c.biaserr	Warning level for high change in bias e.g. 6 x RON
c.flaterr	Warning level for high change in flat signal e.g. 4 x Poison noise
c.sighi	Warning level for high signal as fraction of full scale
c.siglo	Warning level for low signal as fraction of full scale
c.extsigma	Maximum error (in sigma) for unmasked pixel in optimal extraction
c.skymin	Warning level for low sky signal in scaled sky subtraction
c.skymax	Level of sky line that is clipped for masked sky subtraction

c.msigmax	Max error criteria used for medain of two signals
c.ccdron	CCD read out noise (e-)
c.ccdgain	CCD gain (count/e-)
c.ypixel	Number of CCD pixels in y direction
c.ymax	Maximum y-pixel on CCD image for acceptable signal
c.ccdzero	Zero level for binary CCD image
c.noscan	Width of overscan region used in bias correction
c.oscan1	x pixel positions of start of overscan region on LHS gap
c.oscan2	x pixel positions of start of overscan region on LHS image
c.oscan3	x pixel positions of start of overscan region on RHS gap
c.oscan4	x pixel positions of start of overscan region on RHS image
c.circcutl	Addition number of pixels masked to LHS of gap for line tracing
c.circcutr	Addition number of pixels masked to RHS of gap for line tracing
c.nfiber	Number of fibers in AF2/WYFFOS set up
c.nfiducl	Number of fiducials in AF2/WYFFOS set up
c.rcircle	Minimum radial position of fibers on plate+D36 in circle set up
c.dfiber3	Spare dead fiber pointers
c.dfiber4	Spare dead fiber pointers
c.fiberdip	Scaling parameter for overall geometric y-offset of fibers
c.fiberoff	Scaling parameter for local geometric y-offset of fibers
c.pixelsiz	Size of CCD pixel
c.arcorder	Order of polynomial describing wavelength per pixel
c.arcwide	Width of arc spectra used to fit Gaussian profile to peak
c.arcrefc	Pixel position of arcline used to deternine y-offset of fibers
c.arcrefw	Pixel range of arcline used to deternine y-offset of fibers
c.arcscale	Multiplier used to scale plots of arc spectra
c.atlfile	Name of atlas file referenced by Atlas procedure
c.natl	Number of atlas lines displayed in atlas plot
c.atlscale	Multiplier used to scale plots of atlas line magnitudes
c.atlflag	Working variable: showing status of Atlas procedure
c.atlaline	Working variable: showing line being edited in Atlas procedure
c.nbias	Number of bias files (1-21)
c.dbias(0)	Bias file numbers
c.dbias(1)	""
c.dbias(2)	""
c.dbias(3)	""
c.dbias(4)	""
c.dbias(5)	""
c.dbias(6)	""
c.dbias(7)	""
c.nflat	Number of flat files (1-11)
c.dflat(0)	Flat file numbers
c.dflat(1)	""
c.dflat(2)	""
c.nmask	Number of dark files (1-5)
c.dmask(0)	dark file numbers
c.ncirc	Number of circ files (0 or 1)
c.dcirc(0)	Circ file number
c.narc	Number of arc files (1 or 2)
c.darc (0)	Arc file number
c.nstar	Number of target files (1 to 11)
c.dstar(0)	star file numbers
c.dstar(1)	""
c.dstar(2)	""
c.noset	Number of offset sky files (0 to 11)
c.doset(0)	offset sky file numbers
c.doset(1)	""

c.narcline	No. of lines in arcline table
c.calwl(0)	Central wavelength of arcline
c.pixc(0)	Center pixel of arcline in arc lamp spectra
c.pixw(0)	Pixel range for arcline in arc lamp spectra
c.flag(0)	0:arcline not used / 1:arcline used for wavelength calibration
c.calwl(1)	""
c.pixc(1)	""
c.pixw(1)	""
c.flag(1)	""
c.calwl(2)	""
c.pixc(2)	""
c.pixw(2)	""
c.flag(2)	""
c.calwl(3)	""
c.pixc(3)	""
c.pixw(3)	""
c.flag(3)	""
c.calwl(4)	""
c.pixc(4)	""
c.pixw(4)	""
c.flag(4)	""

Additional parameters used set in pipeline procedures

c.log	Full name of log file
c.nparam	Number of single valued control parameters
c.flag1	flag holding table lengths in GUI
c.flag2	flag holding table lengths in GUI
c.flag3	Not used at present
c.flag4	Not used at present
c.atl(0)	Current value lower wavelength cut for atlas
c.atl(1)	Current value upper wavelength cut for atlas
c.atl(2)	Current value lower wavelength cut for atlas
c.atl(3)	Current value maximum y offset for arcspectra
c.atlwl(50)	Current list of atlas line wavelengths
c.psize	Maximum size of window in pixels
c.black	value defining colour
c.red	value defining colour
c.blue	value defining colour
c.white	value defining colour

1.2 General observational data

Structure h	Descriptor
h.run	Run number
h.obstype	Type of observation, e.g. TARGET
h.wyffos	Name of instrument configuration
h.cenwave	Approx. central wavelength (Angstroms)
h.disperse=	Nominal dispersion (Angstroms/mm)
h.linesmm	No. of grating rulings per mm
h.lamp	WYFFOS Comparison Lamps
h.wyfconf	WYFFOS Configuration (Reflection or Trans.)
h.wyfgrat	Name of grating
h.wyforder=	Wavelength order (Echelle grating)
h.confra	Field RA as defined in CONFIGURE
h.confdec	Field Dec as defined in CONFIGURE
h.field	Field name as defined in CONFIGURE
h.ndead	Total number of fibres disabled
h.nfiber	Total number of fibres in this module
h.nfiduc	Total number of fiducial positions
h.ra	RA being tracked
h.dec	DEC being tracked
h.equinox	Equinox of coordinates
h.jd	Julian Date at start of observation
h.start	Local sidereal time at start of observation
h.xbin	Binning factor in x axis
h.ybin	Binning factor in y axis
h.exptime	[s] Exposure time
h.darktime	[s] Integration time

1.3 Fiber observational data (f)

Structure f(#)	Descriptor
f(#).stat	1:valid fiber 2:fiducial fiber 3:dead fiber (recordered in AF2/WYFFOS header) 4:dead fiber located in gap between halves of CCD 5:invalid fiber is case of single fiber extraction
f(#).line	Line number on CCD image (see Circ module)
f(#).type	P:fiber allocated to star target S:fiber alocated to tsy taget C:fiber set in Circle set up X: unused fiber, parked or dead
f(#).name	Object name (from file CONFIGURE)
f(#).mag	Object magnitude (from file CONFIGURE)
f(#).ra	Fibre RA
f(#).dec	Fibre Dec
f(#).x	Fibre x position (microns)
f(#).y	Fibre y position (microns)
f(#).gain	Fiber relative transmission (see Circ module)
f(#).sky	Scaling of median sky subtraction (see Star module)

2.1 Offset image file

File name: c1234567.fit (structure also used for offset images in optimal extraction procedure)

Header: structures c,h,f

Image structure 1 page, 2 dimensions, floating point array

Dimensions nline x nstrip by c.ymax where:
nline is number of lines (traces) on the CCD image produced when fibers are in circle set up
nstrip is ntrace+4 and ntrace is the width of a strip of the CCD image encompassing the line (trace) on the raw CCD image. ntrace is set dynamically in procedure file_offset as (1+c.lwide/c.xbin)

The following example is for x-binning of 2 and clwide =24 giving ntrace=13 and nstrp=16

Column 0: x-pixel position of the maximum in line 1 on the CCD trace at each y-pixel position
Column 1: median signal between lines on the CCD on the LHS of the trace at each y-pixel
Column 2 to ntrace is a section of the CCD image encompassing line 1 at each y-pixel position
Column ntrace+3: median signal between lines on the CCD on the RHS of the trace as each y-pixel

This format is repeated for each of the lines across the CCD image

2.2 Intermediate spectra file

File name a1234567.fit, s1234567.fit, t1234567.fit

Header: structures c,h,f

Image structure 1 page, 2 dimensions, floating point array

Dimensions 8 x nline by c.ymax
where nline is the number of traces identified from the circ set up image
image is created by procedure extr_file.pro
The column containing wavelength data is filled later in the pipeline

Column 0: wavelength value at each y-pixel position
Column 1: optimally extracted target spectra magnitude at y-pixel position
Column 2: optimally extracted target spectra variance at y-pixel position
Column 3: optimally extracted flat spectra magnitude at y-pixel position
Column 4: optimally extracted flat spectra variance at y-pixel position
Column 5: target / flat spectrum magnitude at y-pixel position
Column 6: target / flat spectrum variance at y-pixel position
Column 7: Mask for target / flat spectrum

This format is repeated for each of the lines. Columns relating to lines not allocated to a sky or star type target (i.e. type ‘X’) are set to 0.

2.3 output spectra file

File name o1234567.fit, p1234567.fit, q1234567.fit, u1234567.fit

Header: structures c,h,f

Image structure 2 page, 2 dimensions, floating point array
Dimensions c.nfliber+2 by c.nwl for multifiber extraction
Dimensions 3 by c.nwl for single fiber extraction
image is created by procedure star_make.pro
Example given below is for for c.nfiber=160

Multifiber extraction
Page 0
Column 0: wavelength value at each point of common wavelength scale
y-pixel number in case of file type u (generated in auto mode)
Column 1: fiber 1 - sky subtracted spectra magnitude on common wavelength scale
Column 2: fiber 2 - sky subtracted spectra magnitude on common wavelength scale
….
Column 160: fiber 160 - sky subtracted spectra magnitude on common wavelength scale
Column 161: median sky spectra magnitude on common wavelength scale

Page 1
Column 0: wavelength value at each point of common wavelength scale
y-pixel number in case of file type u (generated in auto mode)
Column 1: fiber 1 - sky subtracted spectra error on common wavelength scale
Column 2: fiber 2 - sky subtracted spectra error on common wavelength scale
….
Column 160: fiber 160 - sky subtracted spectra error on common wavelength scale
Column 161: median sky spectra error on common wavelength scale

Single fiber extraction
Page 0
Column 0: wavelength value at each point of common wavelength scale
y-pixel number in case of file type u (generated in auto mode)
Column 1: fiber c.fiberid - sky subtracted spectra magnitude on common wavelength scale
Column 2: median sky spectra magnitude on common wavelength scale

Page 1
Column 0: wavelength value at each point of common wavelength scale
y-pixel number in case of file type u (generated in auto mode)
Column 1: fiber c.fiberid - sky subtracted spectra error on common wavelength scale
Column 2: median sky spectra error on common wavelength scale

3.1 PROCEDURE ARC VERSION 1_0
Procedure Arc
Function Check input file type, binning and fiber allocations are correct
Create intermediate file of optimally extracted spectra
Calibrate wavelength against pixel position for active fibers

Options Arc,/NOEXTRACT - skips optimal extraction stage
Arc,/ATLAS - runs procedure Atlas to create/edit arcline data

Input Raw data image(s) of the arc lamp (r1234567.fit)
Output Arcfile containing extracted arc spectra & wavelength per pixel (a1234567.fit)

Report to log RUNNING PROCEDURE:Arc
For file number 1234567
New Arc file created:a1234567
Lamp Ne max delta for fit 0.### pixels (see file arc.txt)
Common wavelength range ####.# to #####.# A
Arcline wavelength mean_delta_fit rms_delta_fit
1 ####.### #.### #.### (repeat for each arcline)

Report files Image of five plots showing stages of the calibration process (Arc.gif)
Table of wavelength limits and deviation of calibration lines per fiber (Arc.txt)
Error FATAL ERROR: incorrect file type, file type is xxx
message ERROR: specified minimum wavelength greater than actual maximum
ERROR: specified maximum wavelength greater than actual maximum

Procedures called;
read_control - Returns control data in structure c
read_data - Reads pipeline header (c,h,f) and binary image of raw data file
read_file - Reads pipeline header (c,h,f) and image of pipeline fits file
extr_file - Optimal extract target spectra to produced intermediate spectra
atlas - Run GUI to edit arcline data in control file
arc_offset - Get relative offset of individual arc spectra in pixels
arc_cal - Add wavelength calibration to data array of arc spectra
write_file - Writes pipeline header (c,h,f) and data array to fits file
printl - Prints text message to log file

Procedure
Read control data in structure c
print list of files to analyse to screen and log file
read header of first arc file in list
Check file type
get intermediate spectra for specified arc file(s)
Calling extr_file
Input: arcfile number
number of iterations for optimal extraction routine (0)
Output: header information relating to arcfile image (h,f)
array (spectra) containing intermediate spectra vs pixel number
save intermediate spectra to file (a1234567.fit)

Read header and intermediate spectra of first arc spectra
In case when two arcfiles are specified read data for second
update parameter h.lamp in header data
print note to screen and log file
combine two arcfiles as specified in user manual
first part (up to pixel c.afilecut taken from arcfile 1
remainder taken from arcfile 2
save intermediate spectra of combined arcfile to file (a1234567.fit)
Calling Atlas (if using option Arc,/Atlas)
Function:Run GUI to edit arcline data in control file

Select the line the arc spectra used to calculate y-offsets
Select the Atlas file name and common wavelength range
Select the wavelength and pixel position of calibration lines
calibrate arc lines
Read control data in structure c
set plot window used by following subroutines
Calling arc_offset
Input: arcfile header and intermediate spectra (c,h,f,spectra)
Output: yoff - relative offset of individual arc spectra in pixels found
by comparing position of a selected line in the spectra
Calling arc_cal
Input: arcfile header and intermediate arc spectra (c,h,f,spectra)
yoff - relative offset of individual arc spectra
Output: wavelength calibration added data array of arc spectra (spectra)
using polynomial fit of wavelength vs pixel number
array of offsets between calibration data and polynomial fits (del)
limits of common wavelength range found from spectra (wlmin,wlmax)
Save Image of plots of calibration process to log file arc.gif
Output arcfile and log data
save intermediate spectra to file (s1234567.fit)
print summary data to log file and screen
save table max/min wavelength and fit data (del)to file arc.txt
Check specified wavelength range
get mean and rms fit of polynomial for each calibration line
Print fit statistics for calibration lines used to screen and log file

3.2 PROCEDURE ARC_CAL VERSION 1_0
Procedure arc_cal,c,h,f,spectra,yoff,del,wlmin,wlmax

Function For each fiber fit a polynomial of wavelength vs pixel number
using the header table of pixel range and wavelengths for selected arclines
use polymonial to fill the wavelength per pixel in the intermediate arc spectra

Input: arcfile header and intermediate arc spectra (c,h,f,spectra)
yoff - relative offset of individual arc spectra
Output: wavelength calibration added data array of arc spectra (spectra)
using polynomial fit of wavelength vs pixel number
array of offsets between calibration data and polynomial fits (del)
limits of common wavelegth range found from spectra (wlmin,wlmax)

Error FATAL ERROR insufficient arclines flagged for spectified order of fit
message

Procedures called;
printl - Prints text message to log file

Procedure
check number of arclines flagged is OK
set up arrays used in calculation
array yval holds the centerline of each calibration line for each fiber
set y axis range for plots
shift spectra by y offset
plot overlay of offset spectra
lines indicating central range of plot
Plot axes for overlay of arc spectra
add line markers bracketing arc lines used for calibration
plot arc spectra
Get position of calibration lines
by fitting a Gaussian profile to sections of the spectra bracketing each line
set array of pixels used for Gaussian fit
set axes for plot showing overlay of peak profiles
loop for all valid fiber numbers
select offset spectra as 1D array
loop for each calibration line
find position of peak in arc spectra corresponding to calbration line
plot section of arc spectra used to identify the peak
fit Gaussian to get best fit to line centre using IDL procedure GAUSSFIT
this gives message: Program caused arithmetic error: Floating underflow
this message is OK in this context and does not indicate an error
save centreline data in array yval
select flagged arclines to used for polynomial fit
fit a polynomial function to line centre data for each fiber
and use result to get the wavelength value for each pixel
save value of delta between polynomial fit and actual line centre
save the wavelength value for each pixel in the intermediate spectra array
Get maximum common wavelength range
set up plot showing overlay of wavelength calibrated spectra
plot markers showing arclines used for calibration
plot calibrated spectra
annotate plot wit values of rms fit for each calibration line

3.3 PROCEDURE ARC_OFFSET VERSION 1_0

procedure arc_offset,c,h,f,spectra,yoff

Function plots an overlay of arc spectra using y-offsets calculated from header data
Estimates the y-offsets form the position of the peak in a selected spectral line
plots the estimated y offset as a function of fiber number

Option procedure arc_offset,c,h,f,spectra,yoff,/window works as above
but defines a new plot window for graphical output (used with Atlas procedure)

Input: arcfile header and intermediate spectra (c,h,f,spectra)
Output: yoff - relative offset of individual arc spectra in pixels found
by comparing position of a selected line in the spectra

Procedures called;
read_control - Returns control data in structure c
printl - Prints text message to log file

Procedure
Read control data in structure c
Set plot window (only for option /window)
set y axis range for plots
rough estimate of estimate y-offsets using header data
estimate maximum offset across CCD from header dispersion
get nominal fiber position counter across CCD image
translate this to a nominal distance from the centerline
calculate the yoffset as a function of this distance squared
add a term to take account or the 1/2/3 stagger of fibers
set offset of inactive fibers to negative value
Fill array holding offset spectra for all active fibers
identify fiber with lowest y offset
if extracting single fiber replace this with single fiber no.-1
plot composite of offset arc spectra
plot offset so that arc spectra near centre of CCD start at zero
add markers identifying arcline used for offset calculation to plot
Calculate y offset per fiber using reference peak
check line markers are in range and not in single fiber mode
find pixel number of peak in spectra within specified range
make sure that all y offset of all active fibers is > 0
set negative value for inactive fibers
plot calculated values of y-offset against fiber number
Repeat initial plot (for option /window)
this is done in order to leave the plot of overlay arc spectra
as the active plot window which can then be accessed by the
graphical user interface to select a new line for the offset calculation

3.4 PROCEDURE BIAS VERSION 1_0
Procedure Bias
Function Check the input file type is correct
Create a median bias file

Input Raw data bias files - series of raw data files (r1234567.fit)
Output Biasfile (b1234567.fit) containing the median bias image

Reports to log 'RUNNING PROCEDURE:Bias'
New Bias file created:b1234567
Saving copy of Bias.gif

Report files Copy of median bias image (bias.gif)

Error FATAL ERROR: readout binning does not match control data
message FATAL ERROR: incorrect file type, file type is xxx
WARNING: high noise in bias images ### adu

Procedures called;
read_control - Returns control data in structure c
read_data - Reads pipeline header (c,h,f) and binary image of raw data file
write_file - Writes pipeline header (c,h,f) and data array to fits file
printl - Prints text message to log file

Procedure
Read control data in structure c
read header and image of first bias file in list
Check file binning and observation type
creat median bias file
define array to hold stack of bias images
load bias images into stack array
define outarr as median of stack array
check noise level of bias images
calculate rms level of bias signal relative to median
compare value to alarm level c.biaserrc.ccdronc.ccdgain
save median image to bias file (b1234567.fit)
Display TV image of bias file image
get constant to scale image to screen
display TV image on screen
save copy of screen image to log file Bias.gif

3.5 PROCEDURE CIRC VERSION 1_0

Procedure Circ

Function Check input file type and binning are correct
Trace centreline of active fibers across raw data image
Creates an offset image of the Circ file
Set values of relative fiber gains in file header

Options Circ,/AUTO identifies fibers located in gap and overwrites values in control file

Input Raw data file for a uniformly illuminated image with fibers in circle set up (r1234567.fit)

Output Circfile containing offset image and centerline data (c1234567.fit)

Report to log RUNNING PROCEDURE:Circ
lines identified with # fiber(s) in central gap, number(s) ## and
Fiber gains set to 1 / calculated from mean line counts / read from file xxx
New circ file created: c1234567.fit

Report files Table of pixel positions of peaks in image profile for each active fiber

Error WARNING :no circ file specified - using first flat file

message FATAL ERROR: incorrect file type, file type is xxx
FATAL ERROR: incorrect width of CCD line profile, c.lwide/c.xbin must be even
FATAL ERROR: cannot select single fiber if Circfile is also a target file';
ERROR: saturation of circ image (>99% full scale)
WARNING: high circ image, ## of full scale
WARNING: low circ image, ## of full scale
FATAL ERROR: error in header data, incorrect fibers identified with gap
FATAL ERROR: file fiber_attn.txt not found
FATAL ERROR: list of fiber gains does not hold values for all valid fibers

Procedures called;

read_control - Returns control data in structure c
read_data - Reads pipeline header (c,h,f) and binary image of raw data file
file_debias - Reads pipeline header (c,h,f) and debiased image from data file
circ_make - Trace pixel positions of line centres on CCD image
write_file - Writes pipeline header (c,h,f) and data array to fits file
write_control- Write control data to current control file (control_data.txt)
printl - Prints text message to log file

Procedure

Read control data in structure c
clear open windows
Check if dedicated circ file is being used
read header and debiased image of first circ file in list
check file type
check parameter c.lwide is compatible with x binning
create median circ image (if required)
define arrays to hold stack of circ images
read header and debiased circ image
scale image to its mean value before taking median
define image as median of stack array and rescale
check specified circ file is OK to use for case singe fiber
check circ signal level
measure peak level in (ystep) cuts through the flat image
cuts averaged over several pixels in wavelength direction
express peak value as a fraction of full scale
Error if normalised peak is under/over specified value
mask edges of centre sectionv
Option only used in case of a bad alignment of traces
where a line falls partially in gap at centre of ccd
identify line centres

Calling circ_make
Input: median circfile header and image (c,h,f,image)
Output: ypix - array of y pixel values used for image cuts
        peak - array of x-pixel value of line centres for image cuts
        num - Number of lines identified in circ image for each cut
        gapfiber - Array identifying fibers in gap (max of 4)
Check if fibers in gap agrees with header data
if using auto mode (circ,/auto) then overwrite control file
parameters c.dfober# with new numbers for fibers in gap
print summary information to screen and log file
showing number of lines and fibers in gap
save table of line centre data to file circ.txt
creat array to hold line centre data at all y values
interpolate x pixel values of line centres as function of y pixel
save results to designated columns of array offset
Calling file_offset
Input: circfile header and image (c,h,f,image)
        array offset of x pixel values of line centres as function of y
Output: array output - contining offset data (see user manual) plus a
        copy of the original circ image shifted along the x-axis
        so that the centres of each line fall at a constant x value

add value of fiber gains to header data (parameter f.gain)
Evaluate approximate offset of lines in the x direction
used to estimate fiber gains for option c.fibrattn=1 (see user manual)
get nominal fiber position counter across CCD image
translate this to a nominal distance from the centerline
calculate the yoffset as a function of this distance squared
using maximum offset across CCD from header dispersion
add a term to take account or the 1/2/3 stagger of fibers
If fibrattn=0 then f.gain=1
If fibrattn=1 then calculate fiber gains by averaging traces of
circ file over fixed line width and length corrected for y offsets
If fibrattn=2 then f.gain is read from file in standards directory
If fibrattn=3 then f.gain is read from file in project log directory
open specified file and read fiber attenuations f.gain
check all active fibers have a value for gain
save image containing line centre data to circ file (c1234567.fit)

3.6 PROCEDURE CIRC_MAKE VERSION 1_0
Procedure circ_make,c,h,f,image,ypix,peak,num,gapfiber

Function Count the number of line (traces) on the CCD image (image)
Identify the fiber number associated with each line
Identify the fiber numbers where the line is lost in
the gap between the two haves of the CCD

Input: median circfile header and image (c,h,f,image)
Output: ypix - array of y pixel values used for image cuts
peak - array of x-pixel value of line centres for image cuts
num - Number of lines identified in circ image for each cut
gapfiber - Array identifying fibers in gap (max of 4)

Report to log Saving copy of Circ.gif
Report files Plot of image profiles showing the fiber number for each line (Circ.gif)

Error FATAL ERROR: unable to identify peaks correctly in image profile
message FATAL ERROR: variable number of peaks found in image profile

Procedures called;
printl - Prints text message to log file

Procedure
set graphics window
set up plot and output arrays
set y-pixel value for intermediate cuts
set y-pixel value for upper and lower cuts
create and analyse a smoothed profile at each cut level
Select a strip accross the CCD image several y-pixels wide
average and smooth this strip to give a line profile (y)
Remove the low frequency component of the line profile
Normalise profile to its mean value
plot current image profile
Fill array holding positive zero crossing values
Fill array holding negative zero crossing values
Check equal numbers of positive and negative crossings are detected
Use +ve and -ve crossing values to fil array of x-pixel positions of peaks
print number of lines identified in current cut
check same number of lines found for each cut
count no. lines on each side of gap
set array of all valid fibers
f.stat type 1 is a valid fiber & type 4 is a valid fiber but in gap
f.type X is a fiber with invalid or parked target (i.e. no trace)
identify fibers showing lines on rhs (i.e. not in gap)
identify fibers showing lines on lhs (i.e. not in gap
capture fiber numbers of fibers in gap (maximum 4)
identify fiber numbers of lines (traces) on CCD image
create array of fiber numbers corresponding to lines left to right across image
replot image profile with text indicating fiber number of each line
add fiber numbers to plot of mid y profile
Save plot to log file

3.7 PROCEDURE EXT_FILE VERSION 1_0
Procedure extr_file,nfile,nit,h,f,spectra
Function Optimal extraction of spectra for a targets in a star file

Input stafile number (1234567), number of repeat iterations of optimal extraction process
Output header data (h,f) and intermediate spectra array for starfiles (s1234567.fit)

Report to log Extracting file r1234567 (Repeated for each target)
FLAT Iteration 0 masked points = ### (etc.)
TARGET Iteration 0 masked points = ### (etc.)
Error ERROR: Saturation of extracted image (>99% full scale)
message WARNING: Flat signal zero over part of spectra - a total of ## bad pixels masked
WARNING: high extracted image, ## of full scale

Procedures called;
read_control - Returns control data in structure c
file_debias - Reads pipeline header (c,h,f) and debiased image from data file
file_offset - offset image to centre of each line at a uniform x value
printl - Prints text message to log file

Procedure
Read control data in structure c
Get input and associated files
read circfile header (hc,fc) to get line No. and attenuation for each each fiber
read image (offset) which gives offset of lines on CCD image
get number of lines (traces) on CCD image
read the starfile header (h,f) and debiased image
offset image (star) with centres of each line at a constant x value
read the intermediate mask image (if required)
offset image (mask) with centres of each line at a constant x value
read the median flat file image
offset image (flat) with centres of each line at a constant x value
check star signal level
measure peak level in (ystep) cuts through the flat image
cuts averaged over several pixels in wavelength direction
express peak value as a fraction of full scale
Error if normalised peak is saturated or specified value
Copy circfile values of line No. and attenuation to starfile header
Scale starfile and flat by CCD gain to give output in e- per pixel
set up variance arrays for star and flat
Set up extraction profile
get width of strip encompassing a line (trace) on the CCD
get profile in case of a top hat profile (non standard)
get profile for optimal extraction
smooth flat profile in the wavelength direction
offset profile (prob) with centres of each line at a constant x value
set target type as a function of line (trace) number
Subtract scattered light (iff required)
The offset image contains the median signal level between traces
These levels are liniarly interpolated over the width of the trace
to estimate the background level of scattered light for an image
define array of scattered light for star, flat and probability image
subtract scattered light array from star, flat and probability image
set miminun value of these arrays to zero
normalise probability profile
set up probabilty mask where pmask=1 if probability profile is valid
to extract the spectra at a given y-pixel position of a given spectra
Check and normalise probability profile at all valid line positions
sum probability profile across width of trace
for y-pixel positions where sum is zero set probability mask to zero

Print log message if any pixels are masked for bad probabilty profile
Extract spectra
Calling extr_make
Input: Flat file header data (c,hf) flat image (flat) and variance (varf)
      mask (mask),profile (prob) target types (type) and No. iterations (nit)
Output: line (extracted) spectra of flat (lspecf) and variance (vspecf)
Calling extr_make
Input: Flat file header data (c,h) flat image (star) and variance (vars)
      mask (mask),profile (prob) target types (type) and No. iterations (nit)
Output: line (extracted) spectra of star (lspecs) and variance (vspecs)
Load extracted spectra of flat and star into intermediate spectra array
each line (trace) is represented by 8 columns (0 to 7 etc) of the array
column 0 is zero (later it will hold wavelength vs pixel No.)
columns 1 and 2 hold the magnitude an variance of the star spectra
columns 3 and 4 hold the magnitude an variance of the flat spectra
get mean magnitude of (unmasked) flat spectra
valid pixel positions of star spectrum
these are unmasked points with a good extraction profile
columns 5 and 6 hold the magnitude an variance of the normalise spectra
which is (star spectrum/flat spectrum) x mean flat signal

3.8 PROCEDURE EXTR_MAKE VERSION 1_0
Procedure extr_make,c,h,image,var,mask,prob,type,nit,lspec,vspec,dspec

Function optimally extract line spectrum from the measured trace on
the CCD image See algorithm in Fig15 of user manual

Input: extract file header data (c,f) and image (image) and variance (var)
mask (mask),profile (prob) target types (type) and No. iterations (nit)

Procedures called;
printl - Prints text message to log file

Procedure
Set up
set up arrays for extracted spectrum, variance
set up working array holding denominator of extraction formula
set up array holding predicted line profile
set up array of pixels to be masked - initial value = input mask
extract spectrum
Define terms required for optimal extraction
Block pixels of output spectra where more than 50% of input is masked
increment total number of blocked points in all extracted spectra
evaluate magnitude and variance arrays of extracted spectra
print image type and number of masked points to log file

3.9 PROCEDURE FILE_DEBIAS VERSION 1_0
Procedure file_debias,nfile,c,h,f,image
Function read raw data file, dbias the image using intermediate
bias image b1234567.fit and compensate for dc offset

Input: control data (c) and file number (nfile)
Output: file header data (,c,h,f) and debiased image (image)

;Error FATAL ERROR: readout binning does not match control data
message ERROR: central wavelength not compatible with target file r1234567
FATAL ERROR: allocation of active fibers incompatible with target file r1234567

Procedures called;
read_data - Reads pipeline header (c,h,f) and binary image of raw data file
read_file - Reads pipeline header (c,h,f) and image of pipeline fits file
printl - Prints text message to log file

Procedure
read intermediate bias file image (biasim)
set up image array to hold dc bias
read header and image of data file r1234567.fit
convert image to floating point and subtract bias image
check x and y binning is compatible with control data
set counter to identify input file type, 1 is true 0 is false
check central wavelength compatible with target image
for file types flat,arc,star,offset sky
check fibers flagged as active compatible with target image
for file types star and offset sky
for file types flat and arc
report error if incompatible
Correct for dc bias using overscan regions
get median dc offset using overscan regions of LHS of CCD
get median dc offset using overscan regions of RHS of CCD
correct debiased image for any dc offset found

3.10 PROCEDURE FILE_OFFSET VERSION 1_0
Procedure file_offset,c,h,image,offset,output

Function Creates an offset version of the debiased image in the form
required by the optimal extraction process i.e. with the peaks in
the lines (traces) on the CCD image at consistent x pixel values.
the format of the output array is the same as the offset image
the format is described in section 2.1 of the programmers guide

Input: header data (c,h) and debiased image (image)
array containing x pixel values of line centres on CCD as function of y (offset)
Output: array (output) containing x values of line centres together with strip images
for each line of the debiased image offset to a show peaks at constant x values
The output array also contains the median signal level between lines on the CCD.

Procedures called; None

Procedure
get number of lines seen on CCD image (from size set for offset array)
set width of strip on CCD image encompassing a line (trace)
set incremental step along output image for each line
set width in x pixels of image averaged to find median signal between lines
initialise output array (output) and two working arrays
fill output array with offset strip images showing peaks at constant x values
evaluate median signal between lines (traces) on the CCD image
write median value (over c.yavg/c.ybin pixels) to output array

3.11 PROCEDURE FLAT VERSION 1_0
Procedure Flat
Function Check the input file type, binning and fiber allocation are correct
Create a median flat image

Input Raw data flat files - series of raw data files (r1234567.fit)
Output Flatfile (f1234567.fit) containing median flat image

Reports to log RUNNING PROCEDURE:Flat
New Flat file created:f1234567 median of # file
Saving copy of Flat.gif
Report files Copy of median flat image (flat.gif)
Error FATAL ERROR: incorrect file type, file type is xxx
message WARNING: high noise in flat files, ### adu
ERROR: Saturation of flat image (>99% full scale)
WARNING: High flat image, ## of full scale
WARNING: Low flat image, ## of full scale

Procedures called;
read_control - Returns control data in structure c
read_data - Reads pipeline header (c,h,f) and binary image of raw data file
file_debias - Reads pipeline header (c,h,f) and data array
write_file - Writes pipeline header (c,h,f) and data array to fits file
printl - Prints text message to log file

Procedure
Read control data in structure c
clear open windows
read header and debiased image of first flat file in list
Check file type
creat median bias file
define arrays to hold stack of flat images and output
load flat images into stack array
read header and debiased flat image
scale image to its mean value before taking median
define outarr as median of stack array and rescale
save median image to flat file (f1234567.fit)
check noise level of flat images
calculate rms level of flat signal relative to median
check flat signal level
measure peak level in (ystep) cuts through the flat image
cuts averaged over several pixels in wavelength direction
express peak value as a fraction of full scale
Error if normalised peak is under/over specified value
Display TV image of flat file image
get constant to scale image to screen
display TV image on screen
save copy of screen image to log file Flat.gif

3.12 PROCEDURE FLAT_SCALE VERSION 1_0
Procedure flat_scale,c,fwl,fmag,wl,fscale

Function Create smoothed profile of the input flat spectrum
normalised to its mean value (unit value returned if parameter c.flatscale=0)
Input: control data (c), wavelength (fwl) and magnitude (fmag) of intermediate flat spectrum
common wavelegth scale (wl)
Output: flat spectrum profile (fscale) i.e. smoothed version of flat spectrum
normalised to its average value on common wavelength scale
unit value returned if parameter c.flatscale=0

Procedures called: None

Procedure
creat unit value spectrum on common wavelength base
return in no flat spectrum profile is required
create 1 dimensional versions of input spectra and wavelength
Interpolate over masked sections
set first and last point of column holding mask data to 1
Set arrays to hold pixel No., magnitude and variance of unmasked points
initialise counter of number of pixels containing unmasked point
fill arrays containing pixel No., mag and val for unmasked point
increment counter if original spectra contains unmasked point
truncate arrays of wavelength and magnitude at max No. of unmasked points
interpolate data for unmasked points over all wavelength values
Normalise interpolated flat spectrum to its mean value
Smooth normalised spectrum to give flat lamp profile
degree of smoothing set by parameter c.flatsmth
transform flat lamp profile onto common wavelength base

3.13 PROCEDURE LOADRB VERSION 1_0

Procedure Loadob
Function Selects an RB control file from a network directory
Option Loads a copy as the active control file
Resets the log file of the selected reduction block

Input Control file in project directory
Output Active control file in working directory

Report to log Loading control_file: C:/xxx/xxx.txt

read_control - Returns control data in structure c
printl - Prints text message to log file

Procedure
pick new control file to load
copy over new control file
Create new log file using new reduction block name

3.14 PROCEDURE MASK VERSION 1_0

Procedure Mask

Function Check the input file type is correct
In case of multiple dark files determine the minimum dark current per pixel
Create mask of pixels that show a dark current above the specified cut off level

Options Mask,/AUTO - Mask a fraction of pixels rather than a fixed dark current per pixel

Input Raw data dark files series of raw data files (r1234567.fit)

Output Maskfile (m1234567.fit) containing mask image

Reports to log RUNNING PROCEDURE:Mask
Masking #.##% of pixels with > ##.## adu/hour
New mask image created:m1234567 using # files
Saving copy of Mask.gif

Report files Copy of mask image (mask.gif)
Table of fraction of masked pixels versus cut off level (mask.txt)

Error WARNING: no dark file specified

message FATAL ERROR: incorrect file type, file type is xxx
ERROR: mask cut greater than specified maximum level

Procedures called;

read_control - Returns control data in structure c

file_debias - Reads pipeline header (c,h,f) and data array

write_file - Writes pipeline header (c,h,f) and data array to fits file

printl - Prints text message to log file

Procedure

Read control data in structure c

cancel operation if zero mask files are specified

read header and debiased image of first dark file in list

Check file type

Get mask array

define arrays to hold stack of dark images and output

load flat images into stack array

read header and debiased dark image

scale input to give dark current per hour

Get minimum of dark current from stack array

Get histogram of fraction of pixels vs dark current

define dark current levels in pixel count per hour

define array to hold histogram values

Set maximum dark current for auto mode (mask,/auto)

set variable maskcut to give specified fraction of masked pixels

error if cut off level required for auto mode is too high

Set maximum dark current for standard mode

set variable maskcut to specified value

Generate mask image (1/0)

get percentage of masked pixels

save mask mask to mask file (m1234567.fit)

print summary information to screen and log file

save table of histogram values to log file

log/log plot on histogram on screen

Display TV image of bias file image

get constant to scale image to screen

display TV image on screen

save copy of screen image to log file mask.gif

3.15 PROCEDURE MEDAN VERSION 1_0

Procedure Medan
Function Evaluate median values of extracted spectra, plots and saves results
Option Median,/AUTO - plots data against pixel No. (no wavelength calibration)

Input Output sky subtracted spectra files (p1234567.fit) for each observation
Note Starfile (s1234567.fit) used in case of /AUTO mode
Output Median spectra output file (q1234567.fit) over common wavelength range
Median file (u1234567.fit) containing spectra versus pixel no. (for auto mode)
Plots showing individual spectra and resulting median spectra for each active fiber

Report to log RUNNING PROCEDURE:Medan
Medan of files: , p1234567,(etc.) total exposure time #### sec
Median spectra saved to file q1234567
Saving plots to directory c:/####/temp/

Procedures called;
read_control - Returns control data in structure c
read_file - Reads pipeline header (c,h,f) and image of pipeline fits file
medan_make - Get median spectra on a common wavelegth base
write_file - Writes pipeline header (c,h,f) and data array to fits file
printl - Prints text message to log file

Procedure
Read control data in structure c
read spectra on common wavelength base
set observation type 'p' for output type spectra of star
Read output spectra of first of the starfiles (page 1 mag/ page 2 err)
set up arrays to hold stack of output spectra and median spectra
copy output spectra of first starfile to stack array
initialise counter of total exposure time
copy output spectra of additional starfiles to stack array
Read output spectra of the starfiles (page 1 mag/ page 2 err)
set wavelength scale for median spectra
read raw extracted spectra
set observation type 's' for intermediate spectra of star
initialise counter of total exposure time
set up arrays to hold stack of output spectra and median spectra
copy intermediate spectra for starfiles to stack array
Read intermediate spectra of the starfiles (s1234567.fit))
index pixel number in column zero of current page of stack array
interpolate intermediate spectra over masked sections
set first and last point of column holding mask data to 1
Set arrays to hold pixel No., magnitude and variance of unmasked points
initialise counter of number of pixels containing unmasked point
fill arrays containing pixel No., mag and val for unmasked point
increment counter if original spectra contains unmasked point
truncate arrays of pixel no., mag and var at max No. of unmasked points
interpolate data for unmasked points over all pixel values
taking sqrt of variance to give rms error
Resizes stack arrays (mags,errs) in case of single target data
row 0 wavelength, row 1 fiber spectrum, row 2 blank
Resizes median arrays (mags,errs) in case of single target data
print list of files analysed and exposure time to screen/log file
Evaluate median for all valid targets
set column number in mags/errs arrays to be processed to get median
Calling median_make
Input: Observation type- 'p' for output spectra, 's' for intermediate
Observation header data (h,f)
Stacked array of spectra on common wavelegth base (mags & errs)

fiber number (i+1)
column number (na) of spectra for which median is found
Output: array containing median magnitudes (magm) and errors (errm) of
spectra with designated column (na) filled
Screen shows plot of individual and median spectra for fiber i+1
Save plot of individual and median spectra for fiber number ###
plot saved to file 1234567M###.gif in directory /temp
Evaluate median for individual median skys (one per observation)
check that a median sky spectra will have been created
Calling median_make
see notes above
plot saved to file 1234567Msky.gif in directory /temp
Save output spectra to file

3.16 PROCEDURE MEDAN_MAKE VERSION 1_0
Procedure Medan version 1_0
Procedure medan_make,mode,c,f,mags,errs,nf,na,mag,err
Function Evaluate the median spectra of a individual spectra for the selected fiber
Plot the individual and median spectra to screen .

Input: Observation type (mode)- 'p' for output sprectra, 'o' for offset sky, 's' for intermediate
Observation header data (h,f)
Stacked array of spectra on common wavelength base (mags & errs)
fiber number (i+1)
column number (na) of spectra for which median is found
Output: array containing median magnitudes (magm) and errors (errm) of
spectra with designated column (na) filled
Screen shows plot of individual and median spectra for fiber i+1

Procedures called;None

Procedure
get number of targets and file list for target or offset sky
set graphics window
one or two columns depending on the number of observations (nstar)
two columns if 4 or more observations
set up arrays for median spectra (for one fiber)
loop for individual fibers
set sale factor for spectra equal to mean value
set title for plot
set x axis title for plot
plot individual spectra and error
treat trivial case of single observation
get median of a set of observation each normalised to their mean
treat special case of 2 observations as mean corrected for outlying data
rescale output spectra to give absolute values in e-/pixel
set title for plot
plot median spectra and error

3.17 PROCEDURE OFFSET VERSION 1_0
Procedure Offset
Function Optimal extraction of set of offset sky observations
Interpolates spectra onto a common wavelength scale
Evaluate median values of offset sky spectra from repeated frames

Options Offset,/NOEXTRACT - skips optimal extraction stage

Input Raw data images of offset sky
Output File(s) containing intermediate offset sky spectra(t1234567.fit)
File of median offset sky spectra over common wavelength range (o1234567.fit)

Report to log RUNNING PROCEDURE:Offset
For file number(s), 1234567, 1234567, 1234567
Add wavelength calibration from file: a1234567
Offset file created: t1234567
Evaluating median offset sky spectra for target fibers
Median offset sky spectra saved to file o1234567
Saving plots to directory C:/CepOB3_data/temp/

Error FATAL ERROR: no offset sky files specified
Message FATAL ERROR: no target fibers found for sky offset

Procedures called;
read_control - Returns control data in structure c
read_file - Reads pipeline header (c,h,f) and image of pipeline fits file
extr_file - Optimal extract target spectra to produced intermediate spectra
star_make - Get spectra on a common wavelegth base
medan_make - Get median spectra on a common wavelegth base
write_file - Writes pipeline header (c,h,f) and data array to fits file
printl - Prints text message to log file

procedure
Read control data in structure c
Return if no offset sky called or necessary files missing
print list of files to analyse to screen and log file
get intermediate spectra for offset sky file
Calling extr_file
Input: offset sky file number
number of iterations for optimal extraction routine (c.staritn)
Output: header information relating to offset sky image (h,f)
array (spectra) containing intermediate spectra vs pixel number
check file contains at least on valid target
name all valid targets (type P) as "Offset_sky"
designate all other fibers (i.e. not type P) as invalid (type X)
add wavelength calibration data to files of intermediate spectra
make vector (type) containing object type for each line of image
read the intermediate spectra array of the current arcfile
copy wavelength calibration data to the offset sky spectra array
save intermediate spectra for each offset sky to file (t1234567.fit)

Recast on common wavelength range and determine median spectra
Set graphic window
set up arrays to hold stack of output spectra and median spectra
initialise counter of total exposure time
get output spectra onto a common base for each offset sky file
Read header and intermediate spectra for each starfile in turn
set run number for header data of output file
Calling star_make
Input: header data (c,h,f) and array of intermediate spectra (spectra)

file number (1234567) of offset sky file being processed
Output: arrays of magnitude (mag) and error (err) of offset sky spectra
spectra on a common wavelength base (see section 2.3 of developer notes
row 0 wavelength, row 1 to 160 fiber spectra, row 161 blank

Resize mag and err files in case of single target data
row 0 wavelength, row 1 fiber spectrum, row 2 blank
Copy output spectra for single offset sky observation
increment total exposure time
set values of header data for output file
Get median offset sky spectra for target fibers
set column number in mags/errs arrays to be processed to get median
Calling medan_make
Input: observation type 'o' used for offset sky observation
Input: Observation header data (h,f)
Stacked array of spectra on common wavelength base (mags & errs)
fiber number (i+1)
column number (na) of spectra for which median is found
Output: array containing median magnitudes (magm) and errors (errm) of
offset sky spectra with designated column (na) filled
Screen shows plot of individual and median spectra for fiber i+1
save plot to file
Save plot of individual and median spectra for fiber number ###
plot saved to file 1234567M###.gif in directory /temp
Save output spectra to fits file with page 1 for mag and page 2 for err
Messages to screen and logfile showing module actions done

3.18 PROCEDURE PIPELINE VERSION 1_0

Procedure Pipeline

Function Runs the pipeline modules to extract spectra using project calibration data
Modules run Flat > Circ > Arc > Offset > Star > Medan

Options Pipeline,/full runs all modules generating bias and mask files and arcline data
Bias > Mask > Flat > Circ > Arc (with Atlas) > Offset > Star > Medan
Pipeline,/auto runs modules in auto mode
Use existing Bias and Mask files, no sky subtraction or wavelength calibration
Pipeline,/auto,/full runs all modules in auto mode
Generates new Bias and Mask files, no sky subtraction or wavelength calibration

Input Raw data files + intermediate Bias and Mask if available

Output Standard output for each module run

Report to log RUNNING PROCEDURE:Pipeline
Using control file: xxx
Setting auto mode options: profile:1, fibbrattn:1, flattype:0, nscatter:1 (if applicable)
Then one of the following:
Running all modules in Auto mode
Using existing bias and mask files, running remaining modules in Auto mode
Running all modules
Using existing bias and mask files
Followed by standard reports for each pipeline model

Error message ERROR: target file header different no. of fibers (#) and/or fiducials (#)
Overwriting data in control file to match target data
ERROR: target file header shows different readout binning (X:#, Y:#)
Overwriting data in control file to match target data
FATAL ERROR: control file shows incompatible sky type
Use skytype:0 or set offset sky for single fiber extraction
ERROR: intermediate Bias file not found - running Bias
FATAL ERROR: designated Bias file shows incompatible binning
ERROR: intermediate Mask file not found - running Mask
FATAL ERROR: designated Mask file shows incompatible binning

;Procedures called;
read_control - Returns control data in structure c
read_data - Reads pipeline header (c,h,f) and binary image of raw data file
read_file - Reads pipeline header (c,h,f) and image of pipeline fits file
Bias - Create a median bias file
Mask - Create mask of pixels for dark current above cut off level
Flat - Create a median flat image
Circ - Trace centreline of active fibers across raw data image
Arc - Calibrate wavelength against pixel position for active fibers
Oset - Optimal extraction of set of offset sky observations
Star - Optimal extraction of spectra for a set of star files
Median - Evaluate median values of extracted spectra
write_file - Writes pipeline header (c,h,f) and data array to fits file
write_control- Write control data to current control file (control_data.txt)
printl - Prints text message to log file

Procedure
Read control data in structure c
Create new log file using current reduction block name
read target header parameters (structure h and f)
check number of fibers and number of fiducials
on error update control file with star data file values
check x and y binning
on error update control file with star data file values

If running in auto mode then set pipeline options to default values
Check sky subtraction option is compatible with other options
Check bias and mask file
check Bias file can be read
if Bias file not found then run Bias
check binning bias file is compatible with star data file values
If mask file is required then check it can be read
if Mask file not found then run Mask
check binning mask file is compatible with star data file values
run pipeline modules in sequence according to the options selected
Actions for running Pipeline,/full,/auto
Actions for running Pipeline,/auto
Actions for running Pipeline,/full
Actions for running Pipeline (with no options selected)

3.19 PROCEDURE PRINTL VERSION 1_0
Procedure:prinl,logfile,text
Function:Write a line of text to the log file

Input :full name of log file including path and extension (logfile)
string of text to be added to control file (text)

note full logfile name is held in control parameter c.log

Procedures called: None

Procedure

3.20 PROCEDURE READ_CONTROL VERSION 1_0
Procedure:read_control,c
Function:Read control data set (c) from ASCII file named
control_data.txt located in the pipeline directory
Option: read_control,c,file=xxx read control data from a file xxx
where xxx is the file name including path

Input :None
Output :control data in structure (c)

Procedures called;None

local function used to set consistent path separator(/)

procedure
set structure for control data structure
descriptors of control data are insection 1.1 of developer notes
Open control_data file
read single parameter items of control data
note parameters are read in the order they are found in
therefore the contol file mist contain a list of all parameters
in the correct order with no blank lines
any comments following a semi-colon are not read
read options
read general setting
read subdirectories
read limits
read CCD/fiber properties
read arcline parameters
Read Bias data file nos
Read Flat data file nos
Read Mask data file nos
Read Circ data file nos
Read Arc data file nos
Read Star data file nos
Read offset sky data file nos
Read table of arcline data
Ensure that all file paths specify a consistent separator
set total numbers for different types of control parameters
these values are used to set table size in the GUI
set additional control parameters used in I/O procedures
set plot window parameters
this is required to reset IDL settings after procedure ATV
has been run since can leave things in a strange state
set parameter controlling scaling of plot window on screen
c.psize sets the maximum pixel size used
according to the screen size (with a maximum of 1000)
set colours used for pipeline plot

3.21 PROCEDURE READ_DATA VERSION 1_0
procedure read_data,nfile,c,h,f,image
Function read specified fits data file with AF2/WYFFOS header (r1234567.fit)
where the file directory path is specified by the control data
Option read_file,nfile,c,h,f,image,path=xxx reads file from path xxx

Input: Data file ile name including path but without .fit extension
Output: File header data (c,h,f), array of specified page (image)

Module set fiber status in header data (f.stat) as follow;
1 Fiber OK to allocate to target
2 Fiducial fiber
3 Fiber flagged as dead in data file header
4 Fiber OK to allocate but trace is located in CCD gap so no line
5 Fiber OK but not analysd since pipeline operating in single target mode

Module set fiber types in header data (f.type) as follows;
X Unallocated fiber (parked, fiducial or dead)
P Fiber allocated to star target
S Fiber allocated to sky target (including all fibers named PLACE_XY)
C Active fibers (status 1 or 4) positioned in circle set up

Error FATAL ERROR: file xxx not found
message FATAL ERROR: image size does not match control data
WARNING: Data file header shows different number of dead fibers, ##, flagged

Procedures called;
read_control - Returns control data in structure c
printl - Prints text message to log file

Procedure
Read control data in structure c
check that the specified file exists
read fits file header (head) and page 0 (blank) array (image)
read filts file page 1 containing a the binnary data array (image)
Check size of image compatible with control data
clip off lower (blue) mend of spectra (if required)
set maximum length of spectra (in pixels). This clips red end of spectra
set structure for general header parameter (h)
descriptors of header data (h) are section 1.2 of developer notes
read values of parameters (h) from fits file header
set structure for fiber specific header parameters (f)
descriptors of header data (f) are in section 1.3 of developer notes
define values of fiber parameters (f) from fits file header
define fiber name (f.name)
read values of fiber parameters (f) from fits file header
read target type from fits file header setting type=X if none given
read target name and magnitude from fits file header
read target RA and Dec from fits file header
read x and y position of fiber on plate
set fiber gain to 1 and sky scaling parameter to zero
these parameters are generally reset later by pipeline procedures
designate fibers placed at plate position x,y as sky fibers
check No. fibers flagged dead (status 3) is same as fits header
step type of any dead fibers to X
set status of stat=1 fibers flagged as gap fibers in control data to 4
get number lines on CCD image if all fibers used (i.e. circle set up)
allocate line numbers to fibers in reverse order across CCD image
Indentify fibers inside plate radius corresponding to circle set up
if all active fibers are in circle set up then set fiber header data

set type as c, name as circle, RA and Dec as zero and magnitude as 999
Set fiber parameter for case of single fiber extraction
check if file type is and arc,star or offset sky if it is then
 set fiber status of all active but non-selected fibers to type 5

3.22 PROCEDURE READ_FILE VERSION 1_0
procedure read_file,dummy,c,h,f,image
Function read fits file with pipeline header (page 0 read by default)
Option read_file,dummy,c,h,f,image,page=# specifies page number
Input: Full file name including path but without .fit extension
Output: File header data (c,h,f), array of specified page (image)

Procedures called;None

local function used to set consistent path separator(/)

procedure
check file is available to read
read file header (head) and array (image)
read selected page if specified in call to procedure
set structure for control data structure
this is duplicated here so that pipeline files can be read
even if no local control_data.txt file is avaiable
descriptors of control data are in section 1.1 of developers notes
Read key control parameters needed for this procedure
number of single value control parameters (not arrays)
number of fibers and parameters
set structure for general header parameter (h)
descriptors of header data (h) are in section 1.2 of developers notes
read values of parameters (h) from fits file header
set structure for fiber specific header parameters (f)
descriptors of header data (f) are in section 1.3 of developers notes
read values of fiber parameters (f) from fits file header
read (single valued) of control data from fits file header
Ensure that all file paths specify a consistent separator
read input file data for bias files from fits file header
read input file data for flat files from fits file header
read input file data for dark files from fits file header
read input file data for circ files from fits file header
read input file data for arc files from fits file header
read input file data for star files from fits file header
read input file data for offset sky files from fits file header
Read arcline data from fits file header

3.23 PROCEDURE SKY_SUB VERSION 1_0
Procedure sky_sub,nf,c,f,wl,starspec,starerr,skymed,skyerr

Function Subtract sky spectrum from star spectrum.
c.skytype=0:No sky spectrum is subtracted
c.skytype=1:linear subtraction of sky spectrum
c.skytype=2:sky spectrum scaled for 'optimum' subtraction
c.skytype=2:sky spectrum scaled and output masked over large sky lines

Input: fiber number counter (i) and header data (c,f)
common wavelegth scale (wl), star spectrum and its error
sky spectrum and its error
Output: sky subtracted star spectrum and error
header data updated to show fraction of sky spectrum subtracted
type of sky subtraction used set by parameter c.skytype

Error WARNING poor convergence of sky subtraction for fiber ###, scale parameter set to 1
Message

Procedures called:
printl - Prints text message to log file

Procedure
for skytype 0 set header f.sky=0 and return (with no subraction)
hold unmodified version of starspec in starspco
linear subtraction of sky spectrum
add error in sky spectrum in quadrature
for skytype 1 set header f.sky=1 and return
optimum sky subtraction
take ac component of sky spectrum
take ac component of original star spectrum
get zero offset convolution of these two spectra
linear subtraction of sky spectrum
take ac component of sky subtracted spectrum
get zero offset convolution of with ac sky spectrum
set f.sky to achieve to a zero convolution with ac sky spectrum
subtract this optimum fraction of the sky spectrum
reject poor convergence
take ac component of sky subtracted spectrum
get zero offset convolution of with ac sky spectrum
check for convergence
In case of poor convergence set scaling parameter f.sky=1
for skytype 2 return
Mask sky lines above level specified in control data
Mask sky lines
define masking vector set to zero when sky signal exceed cut off level
broaden masked sections to allow for any misalignment at line edges
mask sky subtracted spectrum
Interpolate masked data
set first and last point of mask to 1
Set arrays to hold pixel No., magnitude and variance of unmasked points
initialise counter of number of pixels containing unmasked point
fill arrays containing pixel No., mag and val for unmasked point
interpolate data for unmasked points over all wavelength values

3.24 PROCEDURE STAR VERSION 1_0
Procedure Star
Function Optimal extraction of spectra for a set of star files
Interpolates spectra onto a common wavelength scale
Sky subtraction using either median value of sky fibers or offset sky spectra

Options Star,/NOEXTRACT - skips optimal extraction stage
Star,/PLOT - saves plots of individual spectra for each fiber
Star,/AUTO - Optimal extraction vs pixel number (i.e. no wavelength calibration)

Input Raw data images of designated targets(r.1234567.fit)
Output Starfiles (s1234567.fit) containing intermediate target spectra
Specfiles (p1234567.fit) sky subtracted spectra over common wavelength range
ASCII tables (p1234567.txt) fiber no, line, RA, Dec, Mag, Mean, SNR, sky lev,name

Report to log RUNNING PROCEDURE:Star
For file number(s), 1234567, 1234567, 1234567
Add wavelength calibration from file: a1234567
Star files created: s1234567 (repeat for each target)
Spectra saved to file p1234567 (repeat for each target)
Error FATAL ERROR: readout binning does not match control data
message FATAL ERROR: control file shows incompatible sky type
Cannot have skytype 0, and skyoset 0 for since fiber extraction

Procedures called;
read_control - Returns control data in structure c
read_file - Reads pipeline header (c,h,f) and image of pipeline fits file
extr_file - Optimal extract target spectra to produced intermediate spectra
star_make - Get sky subtracted spectra on a common wavelegth base
write_file - Writes pipeline header (c,h,f) and data array to fits file
printl - Prints text message to log file

Procedure
Read control data in structure c
print list of files to analyse to screen and log file
get intermediate spectra for specified star file(s)
Calling extr_file
Input: starfile number
number of iterations for optimal extraction routine (c.staritn)
Output: header information relating to starfile image (h,f)
array (spectra) containing intermediate spectra vs pixel number
add wavelength calibration data to file of intermediate spectra
make vector (type) containing object type for each line of image
read the intermediate spectra array of the current arcfile
copy wavelength calibration data to the intermediate star spectra array
save intermediate spectra for each starfile to file (s1234567.fit)
finish if running in auto mode (star,/auto)
else continue to subtract sky and recast on common wavelength range
check Sky options are compatible with other settings
Set graphic window
Read control data in structure c
get output spectra onto a common base for each star file
Read header and intermediate spectra for a file
get sky subtracted spectra on common wl base
Calling star_make
Input: header data (c,h,f) and array of intermediate spectra (spectra)
file number (1234567) of starfile being processed
Output: arrays of magnitude (mag) and error (err) of sky subtracted
spectra on a common wavelegth base (see section 2.3 of developer notes
row 0 wavelength, row 1 to 160 fiber spectra, row 161 median sky

save intermediate spectra for each starfile to file (s1234567.fit)
Resize mag and err files in case of single target data
row 0 wavelength, row 1 fiber spectrum, row 2 blank
Save output spectra to fits file with page 1 for mag and page 2 for err

3.25 PROCEDURE STAR_MAKE VERSION 1_0
Procedure star_make,c,h,f,file,spectra,mag,err

Function Generate sky subtracted spectra on a common wavelength base
Scale spectra to flat profile (if specified in control file)

Option star_make,c,h,f,file,spectra,mag,err,/plot
saves individual fiber plots of spectra for each observation

Input: header data (c,h,f) and array of intermediate spectra (spectra)
file number (1234567) of starfile being processed
Output: arrays of magnitude (mag) and error (err) of sky subtracted
spectra on a common wavelength base (see section 2.3 of developer notes
row 0 wavelength, row 1 to 160 fiber spectra, row 161 median sky

Report to log Reading offset sky from o1234567 (in case of offset sky)
Average sky fiber signal ##.# +/- ##.# e-/pixel

Error FATAL ERROR: wavelength calibration data not present
message FATAL ERROR: offset sky file o1234567 not found (if offset sky specified)
WARNING: sky signal is low for scaled sky subtraction
WARNING: Parameter c.skymax is low, less than 3 sigma of median sky signal.'

Procedures called;
read_file - Reads pipeline header (c,h,f) and image of pipeline fits file
sky_sub - Subtract sky spectrum from a single fiber target spectrum
flat_scale - Get flat lamp profile for a single fiber
printl - Prints text message to log file

Procedure
set startype=1 for starfile otherwise zero
set up arrays for output spectra
count number targets of each type (sky,star or circle set up)
Check wavelength calibration data present
Set common wavelength scale - log scale type(0) or linear scale type (1)
fill column 1 of output spectra arrays with wavelength data
set up arrays for sky spectra
Read in median offset sky
only do if current file is a star type and offset sky set in control file
check offset sky file exists
read offset sky file
Copy offset sky spectra to sky arrays scaling for exposure time
treat case of single fiber extraction
Evaluate median sky spectrum
only do if no. sky targets >0 and offset sky not set in control file
set up arrays to stack all sky spectra
Interpolate intermediate sky spectra onto a common wavelegth base
save results sequentially to stack arrays (skymed and skyvar)
scale sky spectra by fiber attenuation and normalised to mean
increment counter of number of sky spectra
truncate stack arrays to contain only sky spectra
evaluate median sky and error (root of variance)
copy median sky to all sky spectra
print message to log
output median sky spectra
set RH column of output spectra arrays equal to median sky spectra
plot median sky spectra
Evaluate output spectra for star (or offset sky) targets
check level of median sky spectra compatible with type of sky subtraction
check cut off level for sky lines is >> typical sky signal level

loop through all valid targets
interpolate intermediate spectra over masked sections
set first and last point of column holding mask data to 1
Set arrays to hold pixel No., magnitude and variance of unmasked points
interpolate data for unmasked points over all wavelength values
skip interpolation in case of arc lamp spectra - a non standard action
Interpolate intermediate spectra onto a common wavelength base
taking sqrt of variance to give rms error
Remove bad data values
cut spectra at maximum credible value
Subtract median sky from star spectrum (but not offset sky)
Calling sky_sub
Input: fiber number counter (i) and header data (c,f)
        common wavelength scale (wl), star spectrum and its error
        sky spectrum and its error
Output: sky subtracted star spectrum and error
        type of sky subtraction used set by parameter c.skytype
Calling flat_scale
Input: control data (c), wavelength and magnitude of intermediate flat spectrum
        common wavelength scale (wl)
Output: flat spectrum profile i.e. smoothed version of flat spectrum
         normalised to its average value on common wavelength scale
         unit value returned if parameter c.flatscale=0
Scale spectra to smoothed flat spectrum
plot target spectrum
if option /plot then save plot to file 1234567Msky.gif in /temp
write summary text file (p1234567.txt) see section 2.3 of developer notes

3.26 PROCEDURE VIEW VERSION 1_0
Procedure View
Function View and save file headers and median output spectra in different formats
saves text copies of the file headers for input and output data files

Options View,xxx,xxx returns a two data arrays containing output spectra magnitude and errors
View,/plot - plots the output spectra for active fibers and saves copies to file
View,/ps - creates PostScript plots of spectra of active fibers and saves to file
View,/text - saves the output spectra for active fibers to individual ASCII files
View,/attenuation - Saves a table and plot of fiber attenuations to log/ directory

Input Target data file and files of intermediate and output spectra
Output Plots and/or text files for individual spectra and/or a table of fiber attenuations

Report to log RUNNING PROCEDURE:View
Saving copy of AF2/WYFFOS header for file: r1234567
Saving copy of pipeline header for file: s1234567

Error FATAL ERROR: file not found:xxx
message

Procedures called;
read_control - Returns control data in structure c
read_file - Reads pipeline header (c,h,f) and image of pipeline fits file
View_plot - Plots the output spectra for active fibers and saves copies to file
View_text - saves the output spectra for active fibers to individual ASCII files
View_attn - Saves a table and plot of fiber attenuations to log/ directory
printl - Prints text message to log file

Procedure
returns arrays of magnitude(mag) and error (err) of median output spectra
the format of these arrays is shown in section 2.3 of the developer notes
Read control data in structure c
save copy of raw data header
open raw data file of the first star (target) file
if file exists then read header array (head)
save header array (head) to file named r1234567.txt
save copy of output header
open intermediate data file of the first star (target) file
if file exists then read header array (head)
save header array (head) to file named s1234567.txt
View median data
execute options
Calling view_plot (option view,/plot)
Input: Header data (c,h,f) and median output spectra (mag,err)
Function:Plots the output spectra for active fibers and saves copies to file
Calling view_plot (option view,/ps)
Input: Header data (c,h,f) and median output spectra (mag,err)
Function:Creates PostScript plots of spectra of active fibers and saves to file
Calling view_plot (option view,/plot)
Input: Header data (c,h,f) and median output spectra (mag,err)
Function:Saves the output spectra for active fibers to individual ASCII files
Calling view_plot (option view,/plot)
Input: Header data (c,h,f) and median output spectra (mag,err)
Function:Saves a table and plot of fiber attenuations to log/ directoory

3.27 PROCEDURE VIEW_ATTN VERSION 1_0

Function Determine attenuation factor for individual fibers scaled to the average for active fibers
The input file must a flat (lamp or twilight sky) recorded in circle set up,
analysed with c.fiberatn=0 (to give f.gain=1) to show uncompensated signal levels

Input: Header data (c,h,f) and text file associated with intermediate spectra (s.1234567.txt)

Report to log:Evaluating fiber attenuation using source file s1234567.txt
Standard deviation in attenuations ##.#%
Saving plot of fiber attenuations to file fiber_attn.gif
Saving table of fiber attenuations to file fiber_attn.txt
Report file :Table of fiber attenuations scaled to the average value for active fibers

Error FATAL ERROR invalid fiber set up, targets must be in circle setup for attenuations
ERROR: fiber gains invalid, fiber attenuation should be unity in source file'

Procedures called:
printl - Prints text message to log file

Procedure
check fiber configuration is OK
check no correction made for fiber attenuation in source file
set up arrays for
nominal angle of fiber in circle set up (mag) used to check harmonic content
mean signal level read from file s1234567.txt
flag set to 1 for valid fiber numbers
maen signal level normalised to average for all valid fibers
Read star file data
identify valid fibers in circle set up
read fiber number
evaluate nominal angle (assuming 160 fibers in a circle)
read mean signal level per fiber
normalise mean signal level per fiber to average value of active fibers
Interpolate missing fibers in circle for harmonic analysis
harmonic analysis
plot fiber attenuation as function of fiber number
title includes note of first component harmonic
Save table of fiber attenuations to text file

3.28 PROCEDURE VIEW_PLOT VERSION 1_0
Procedure View_plot,mode,c,h,f,mag,err
Option View_plot,mode,c,h,f,mag,err,/ps

Function Plots the output spectra for active fibers and saves copies to gif file
Option /ps saves plots to postscript files (.ps)

Input: Header data (c,h,f) and median output spectra (mag,err)

Report to log:Saving gif plot of median of ### sky fibers to c:/xxx/xxx
Saving gif plot(s) of xxx spectra to c:/xxx/xxx

Report to log:Saving ps plots of median of ### sky fibers to c:/xxx/xxx
option /ps Saving ps plot(s) of xxx spectra to c:/xxx/xxx

Procedures called:
printl - Prints text message to log file

Procedure
count number of sky, star and circle set up spectra
Set window for gif plot
Set output parameter for PostScript plot
A copy of plots is sent to the nominated output file
Note the dimensions of the plot are fixed (15cm x 6cm) see below
Plot median sky spectrum
set file name for PostScript output
plot spectrum and associated error
Save copy of plot to gif file (if not PostScript plot)
Plot star spectra
loop through all star targets
create file name for plot and output file
set file name for PostScript output
plot spectrum and associated error
close PostScript output
Save copy of plot to gif file (if not PostScript plot)
Plot individual star and sky files S
create file name for plot and output file
set file name for PostScript output
plot spectrum and associated error
close PostScript output
Save copy of plot to gif file (if not PostScript plot)
Plot circle set up spectra
loop through valid spectra
set file name for PostScript output
plot spectrum and associated error
close PostScript output
Save copy of plot to gif file (if not PostScript plot)
reset device after ps plot

3.29 PROCEDURE VIEW_TEXT VERSION 1_0
Procedure View_text,c,h,f,mag,err
;
Function Saving ASCII copies of individual xxx spectra to /temp directory
file name 1234567M###.txt where xxx is the fiber number

Input: Header data (c,h,f) and median output spectra (mag,err)

Report to log:Saving ASCII copy of median of ### sky fibers to c:/xxx/xxx
Saving ASCII copies of xxx spectra to c:/xxx/xxx

Procedures called:
printl - Prints text message to log file

Procedure
count number of sky, star and circle set up spectra
save median sky spectrum to ASCII file
save individual spectra for multiple target or circle set up
loop through valid spectra
save spectrum and associated error to ASCII file
treat case where single fiber is observed

3.30 PROCEDURE WRITE_CONTROL VERSION 1_0
Procedure:write_control,c
Function:Write control data set (c) to ASCII file named
control_data.txt located in the pipeline directory
Option: read_control,c,file=xxx write control data to file xxx
where xxx is the file name including path

Input :None
Output :control data in structure (c)

Procedures called: None

Procedure
write single parameter items of control data
Read Bias data file nos
Read Flat data file nos
Read Mask data file nos
Read Circ data file nos
Read Arc data file nos
Read Star data file nos
Read offset sky data file nos
Write table of arcline data

3.31 PROCEDURE WRITE_FILE VERSION 1_0
procedure write_file,file,c,h,f,image
Function write fits file with pipeline header (page 0 read by default)
Option write_file,file,c,h,f,image,page=# specifies page number
Input: Full file name including path but without .fit extension
File header data (c,h,f), array of specified page (image)

Procedures called;None

Procedure
create blank fits file header
set file type to accept extensions
load general observation parameters (h) into header
load fiber specific observation parameters (f) into header
load single value control data into header
load non standard parameters into header
load input file data for bias files into header
load input file data for flat files into header
load input file data for mask files into header
load input file data for circ files into header
load input file data for arc files into header
load input file data for star files into header
load input file data for sky offset files into headers
load table of arcline data into header
write fits file
default setting writing image to page 0
option setting writing image to specified page number

4.1 PROCEDURE ATLAS VERSION 1_0
Procedure Atlas
Function GUI that is used to create and edit the arcline data for wavelength calibration
Select the line the arc spectra used to calculate y-offsets
Select the Atlas file name and common wavelength range for the arc spectra
Select the wavelength and pixel position of lines used for wavelength calibration
Calibrate wavelength against pixel position for active fibers

Input Intermediate arc spectra (a1234567.fit)
Contol_data.txt - file containing current control data
Output Contol_data.txt - modified control file with changes made to arcline data

Report to log RUNNING PROCEDURE:Atlas
Overwriting arcline data for control file ### (if selected)
New arcline calibration data saved to control file (if selected)
Quit arcline calibration without changing control file (if selected)

Error FATAL ERROR: Atlas file: xxx not found
message ERROR: Atlas shows no lines in specified wavelength range'

Procedures called;
read_control - Returns control data in structure c
read_file - Reads pipeline header (c,h,f) and image of pipeline fits file
extr_file - Optimal extract target spectra to produced intermediate spectra
arc_offset - Get relative offset of individual arc spectra in pixels
arc_atlas - Plot overlay of arc spectra and plot atlas lines
arc_cal - Add wavelength calibration to data array of arc spectra
arc_gui - Graphical user interface to edit arcline data in control file
write_control- Write control data to current control file (control_data.txt)
printl - Prints text message to log file

Procedure
Read control data structure c from current control file
save copy of input control data to co to reset control data if Quit
read the intermediate spectra array of the current arcfile
Set flags in control data to initialise Atlas procedure
If no atlas file is specified set name as header parameter h.lamp
Overwrite current control file with data in structure c
Calibrate wavelength
The loop below causes a GUI displaying the arcline data. In parallel
procedures are called to generate graphical displays of arc spectra
The case executed is set by control parameter c.atlflag -set in the GUI

Case 0:Select the arcline used to find the y-offsets
Case 1:Select a atlas file, the wavelength range and arclines
Case 2:View wavelength solution using current arcline data
Case 3:Accept the calibration and the update control file
Case 9:Quit procedure (after resetting control data to old values)

variable j holds the number of calls to the loop (maximum 50)
Read control data structure c from current control file
Case 0:Select the arcline used to find the y-offsets
Calling arc_offset
Input: arcfile header and intermediate spectra (c,h,f,spectra)
Output: yoff - relative offset of individual arc spectra in pixels found
by comparing position of a selected line in the spectra
Case 1:Select a atlas file, the wavelength range and arclines
Read control data structure c from current control file
;Calling arc_atlas
Input: arcfile header (c,h,f) and intermediate spectra (spectra)

yoff - relative offset of individual arc spectra in pixels (yoff)
Function:Plot overlay of arc spectra offset by values in yoff
Plot Atlas lines in wavelength limits c.wlmin to c.wlmax
Case 2: View wavelength solution using current arcline data
set plot window
Calling arc_offset
see notes above
Calling arc_cal
Input: arcfile header and intermediate arc spectra (c,h,f,spectra)
yoff - relative offset of individual arc spectra
Output: wavelength calibration added data array of arc spectra (spectra)
using polynomial fit of wavelength vs pixel number
array of offsets between calibration data and polynomial fits (del)
limits of common wavelength range found from spectra (wlmin,wlmax)
Case 3: Accept the calibration and the update control file
Read control data structure c from current control file
Update wavelength limits in control file if outside range
If wavelength limits are changed write a note to screen and log file
set control parameter c.altflag=5 (showing calibration is accepted)
Case 9: Quit procedure (after resetting control data to old values)
save copy of initial control data held in co to control file
Calling arc_gui (if control parameter c.atlflag=0,1,2 or 3)
Input: current value of control data (c)
Function:allows user to unpdate arcline data held current control file
if control parameter c.atlflag=9 then quit procedure
If calibration accepted save fit statistics to log file Atlas.txt
get mean and rms fit of polynomial for each calibration line
print results to screen and to log file

4.2 PROCEDURE ARC_ATLAS VERSION 1_0
Input: arcfile header (c,h,f) and intermediate spectra (spectra)
yoff - relative offset of individual arc spectra in pixels (yoff)
Function:Plot overlay of arc spectra per fiber offset by values of yoff
with markers bracketing the arclines used for wavelength calibration
Plot Atlas spectra in wavelength limits c.wlmin to c.wlmax
Error FATAL ERROR: Atlas file: xxx not found'
message ERROR: Atlas shows no lines in specified wavelength range

Procedures called;
read_control - Returns control data in structure c
printl - Prints text message to log file

Procedure
set plot window
Plot overlay of arc spectra showing positions of calibration lines
shift spectra by offset
plot overlay of arc spectra with markers showing calibration lines
set lines indicating central range of plot
Plot markers on overlay graphic
shaded areas outside central range of plot
plot offset spectra
add markers bracketing calibration lines
Auto set atlas file and wavelength limits
set default name for atlas file
set default limits on wavelength range
if either limit not set in current control file
using central wavelength, dispersion/mm and pixel size in header data
get upper and lower wavelength limits for atlas data
read atlas data file
wavelength calibration line
text ID of calibration line
nominal magnitude of calibration line
select part of atlas data arrays that fall with wavelength range
get array indices equivalent to wavelength limits
flag error if there are no lines in this range
select part of atlas data arrays that fall with wavelength range
limit number of lines to plot to number in wavelength range
limit number of lines to plot to max allowed
atlas for combined lamps contains +ve lines for lamp 1 and -ve for lamp 2
Match scaling of positive & negative magnitudes
select number (c.natl) lines showing highest magnitude
reset order of arrays of Atlas line data in increasing wavelength
fill data arrays used to plot selected atlas lines
set scale for atlas plot
plot of selected atlas lines
copy plot dimensions to control structure (c.atl)

4.3 PROCEDURE ARC_GUI VERSION 1_0
procedure arc_gui,c
Function This procedure provides the graphical user interface for procedure Atlas
to edit parameters of the control data relating to arc line calibration
It is not a standalone procedure, it is called by the parent procedure Atlas,
which also controls the graphics associated the editing process
Control data may be saved to the current control file
It can also overwrite the source control file in the project directory

Message to log:Overwriting arcline data for control file xxx (if actioned)

set of procedures relating to graphical user interface GUI.pro

Local procedures used to control mouse functions
these are used to identify lines on plots of arc/atlas spectra
and to return the pixel position/wavelength at the cursor location

procedure get_yoff,s
Function read the pixel position of a cursor on a plot of overlayed arc spectra
to identify the y pixel position line to be used to estimate the y
offset of spectra (parameter s.arcrefc in te control data)

Input control data (s)
Output control data with parameter s.arcrefc updated

write message on plot area
read x,y value of mouse co-ordinates on plot scale
if x value is in range then update parameter s.arcrefc
procedure get_pix,s
Function read the pixel position of a cursor on a plot of overlayed arc spectra
to identify the y pixel position line of an arcline selected for calibration
the centre positions of these arclines are stored in array s.apixc(#)
whre # is the row counter in the arcline table (parameter c.atlaline)

Input control data (s)
Output control data with parameter s.apixc(#) updated

write message on plot area
If the number of atlas lines is less than the row counter then return
read x,y value of mouse co-ordinates on plot scale. Note the
plotscale refers to the last plot displayed i.e. the wavelength scale of the atlas plot
if x value out of range then return
calculate the equivalent pixel number from the reported wavelength values
set control parameter s.apixc(#) to the measured pixel number
procedure get_wl,s
Function read the wavelength position of a cursor on a plot of atlas lines
to identify the wavelength of an arcline selected for calibration
the wavelengths these arclines are stored in array s.acalwl(#)
where # is the row counter in the arcline table (parameter c.atlaline)

Input control data (s)
Output control data with parameter s.acalwl(#) updated

write message on plot area
If the number of atlas lines is less than the row counter then return
read x,y value of mouse co-ordinates on plot scale
if x value out of range then return
get index of line in array of atlas line wavelengths closest to cursor vale
set control parameter s.acalwl(#) to the corresponding wavelength
Button event procedures

Arcline data is edited in a 3 stage process
the sequence is controlled by parameter s.atlflag which takes values 0,1,2
Edit pixel number of arcline used to calculate y offsets
return if not at step 0 of editing process
Edit pixel range bracketing arcline used to calculate y offsets
return if not at step 0 of editing process
Use cursor to read new pixel number of arcline used to calculate y offsets
return if not at step 0 of editing process
write control data to file return to parent procedure to refresh plot
edit parameter controlling the y axis scale on plots of arc spectra
return if beyond 1 of editing process
write control data to file return to parent procedure to refresh plot
Move to step 2 of editing process by setting S.atlflag=1
return if not at step 0 of editing process
write control data to file return to parent procedure to refresh plot
edit minimum wavelength of common wavelength scale
return if at step 0 or reset to step 1 (if at a later stage of process)
write control data to file return to parent procedure to refresh plot
edit maximum wavelength of common wavelength scale
return if at step 0 or reset to step 1 (if at a later stage of process)
write control data to file return to parent procedure to refresh plot
set the maximum amd minimum wavelength values to zero
this caused new values to be calculated from header data parameters
the next time procedure arc_atlas is run by the parent procedure Atlas
return if at step 0 or reset to step 1 (if at a later stage of process)
write control data to file return to parent procedure to refresh plot
Edit the number of calibration lines displayed in the arcline table
return if at step 0 or reset to step 1 (if at a later stage of process)
zero elements of arcline data above new table length (if new less than old)
fill elements of arcline data arrays up to new table length (if new > old)
write control data to file return to parent procedure to update GUI
Edit name of Atlas file
return if at step 0 or reset to step 1 (if at a later stage of process)
write control data to file return to parent procedure to refresh plot
edit parameter controlling the y axis scale on plot of atlas data
return if at step 0 or reset to step 1 (if at a later stage of process)
write control data to file return to parent procedure to refresh plot
edit number of calibration lines displayed on plot atlas data
return if at step 0 or reset to step 1 (if at a later stage of process)
write control data to file return to parent procedure to refresh plot
Text editing of items in table of arc line data
return if at step 0 or reset to step 1 (if at a later stage of process)
write control data to file return to parent procedure to refresh plot
Edit parameter (s.atlaline) specifying row of the arcline table is
edited when cursor is used to select new pixel position or wavelength data
return if at step 0 or reset to step 1 (if at a later stage of process)
write control data to file return to parent procedure to refresh plot
Edit pixel position of current arcline in table of arcline data using cursor
return if at step 0 or reset to step 1 (if at a later stage of process)
if at step 1 then call procedure get_pix
Input control data (s)
Output control data with parameter s.apixc(#) updated
write control data to file return to parent procedure to refresh plot
Edit wavelength of current arcline in table of arcline data using cursor
return if at step 0 or reset to step 1 (if at a later stage of process)
if at step 1 then call procedure get_wl
Input control data (s)
Output control data with parameter s.acalwl(#) updated
write control data to file return to parent procedure to refresh plot
Run the wavelength calibration process using the current arcline control data

if at step 0 then return
if at step 1 then set flag to step 2
write control data to file return to parent procedure to refresh plot
Quit areline editing process cancelling all changes to control file data
write control data to file return to parent procedure to quit
atlflag=9 causes Altas to reset the control data cancelling all changes
Overwrite the source control file in the project directory with current data
write control data to the source control file in the project directory
Finish editing process
check at right stage to finish
write control data to file return to parent procedure with altflag set to 3
Parent procedure
copy control data to structure s
set array controlling pixel sizing of GUI
set parameters for table of arcline data
Widget base
ROW A title for stage 1
ROW B Arcline for yoffset
open data for editing if atlflag=0
Line centre text
width of markers text
select line centre from plot using mouse
ROW C Plotscale and accept offset
Arc plot scaling text
Accept stage 1 data text
Row D set wavelength range
open data for editing if atlflag>0
lower limit of wavelength text
upper limit of wavelength text
zero wavelengths so that parent procedure calculates limits from header data
Row E Arcline settings
open data for editing if atlflag>0
atlas file name text
atlas plot scaling text
Row F Atlas settings
open data for editing if atlflag>0
number of arclines text
number of atlas lines text
Edit data in arc line table
open data for editing if atlflag>0
Row G Table of arclines
text editing of te arcline data table
Row H Action buttons to edit accept arcline data
select row number to edit droplist
select line centre from arc spectra plot using mouse
select line wavelength from atlas plot using mouse
view wavelength solution using current data button
Section 4 Accept arcline data
Row I Complete calibration
open buttons for editing if atlflag=2
quit button
overwrite control data in project directory button
overwrite current control data file button
event handler
xmanager blocks command line entry until gui closed

4.4 PROCEDURE GUI VERSION 1_0
procedure gui
Function Manage the ASCII files of control data as follow:
Read and display a set of control data from file
View specified data files using the fits file view utility ATV
Edit the parameters of the control data and save to file
Save the displayed data set to file control_data.txt
Run the pipeline in Auto mode using the displayed data set
Pipeline parameters are listed in Section 1.1 of Developers notes

Procedures called;
read_control - Returns control data in structure c
pipeline,/auto- Run pipeline in auto mode
ATV - View contents of a fits data file using ATV utility
write_control- Write control data to current control file (control_data.txt)
printl - Prints text message to log file

set of procedures relating to graphical user interface GUI.pro

local function used to set consistent path separator(/)
local procedure used to view a data file using ATV utility
input control data (s) and file name (dfile)
check file exists
load header
load image array and set to zero (image)
run ATV using header and image data
local procedure to define bitmap for ATV button
Button event procedures
Select and display a new set of control file
Edit name of current control data set (rbname)
Select new project directory path (pdir)
Set fiber number to extract or set to all (fiberid)
Set offset sky yes or no (skyoset)
Set type of sky subtraction (skytype)
Set number of first fiber in gap between halves of CCD (dfiber1)
Set number of second fiber in gap between halves of CCD (dfiber2)
Edit name of first data file for bias (dbias(0))
Edit number of bias files (nbias)
Edit name of first data file for flat (dflat(0))
Edit number of flat files (nflat)
Edit name of first data file for mask (dmask(0))
Edit number of mask files (nmask)
Edit name of first data file for circ (dcirc(0))
Edit number of circ files (ncirc)
Edit name of first data file for arc (darc(0))
Edit name of second data file for arc (darc(0))
Edit number of arc files (narc)
Edit number of star files (nstar)
Edit number of offset sky files (noset)
edit minimum wavelength (wlmin)
edit maximum wavelength (wlmax)
edit name of atlas file (atlfile)
the following procedures edit the names of the star files
a maximum of 10 file names can be set using the GUI
the following procedures edit the names of offset sky files
a maximum of 5 file names can be set using the GUI
Set the control parameter flag3=2 for the GUI to display advanced data
Set the control parameter flag3=1 so the GUI to display arcline data
Set the control parameter flag3=0 so the GUI closes additional displays
save the displayed control parameter to file (pdir+rbname+'.txt')

load the displayed control data as the active control file
save the displayed control parameter to file (pdir+rbname+'.txt')
also save to control_data.txt in the pipeline directory
also save to control_data.txt in the pipeline directory
create new log file with new RB name (rbname)
run the pipeline in auto mode using the display set of control data
save the displayed control parameter to file (pdir+rbname+'.txt')
also save to control_data.txt in the pipeline directory
create new log file with new RB name (rbname)
run procedure pipeline in auto mode
close the GUI without saving the current set of control data
edit a data item in the LH table of advanced data
edit a data item in the centre table of advanced data
edit a data item in the RH table of advanced data
edit table of general arc parameters
edit table of arc calibration lines
view data file using ATV file viewer
the following 21 event procedures call the procedure atv_view
for a specified dta file name

procedure GUIa
Function :Display graphical user interface
:the basic display comprises 10 rows labelled B to K (i.e. guib, guic etc)
:two further rows L and M are opened to display advanced/arcline data

Input :control data (s)

Options s.flag3=0 basic display
s.flag3=1 basic display + arcline dtat tables
s.flag3=2 basic display + advanced parameter tables

get bitmap to be used for ATV button
set array controlling pixel sizing of GUI
Base widget and title of GUI
ROW B
Template file text and button
rbname text
Project directory text and button
ROW C
fiberID droplist
skyoset droplist
Skytype droplist
1st fiber in gap droplist
2nd fiber in gap droplist
ROW D
TITLES
ROWS E, F, G, H, I, J
Biasfile text and droplist
Flatfile text and droplist
Maskfile text and droplist
Circfile text and droplist
First arcfile text and droplist
Second arcfile text and droplist (displayed if narcline=2)
ROWS E, F, G, H, I
nstar droplist
noffset droplist
wlmin text
wlmax text
atlas text
ROWS E, F, G, H, I

Text boxes for star file names - only the specified number (nstar) are displayed
dstar(0) text
dstar(1) text
dstar(2) text
dstar(3) text
dstar(4) text
ROWS E, F, G, H, I
dstar(5) text
dstar(6) text
dstar(7) text
dstar(8) text
dstar(9) text
ROWS E, F, G, H, I
Text boxes for offset sky file names - only the specified number (nosky) are displayed
doset(0) text
doset(1) text
doset(2) text
doset(3) text
doset(4) text
ROW K
Action buttons
Row L, M
display 3 tables of advanced parameters (if flag3=2)
set length of table using No basic parameters (c.flag1)
copy advanced data into 3 data lists (one for each table)
copy tag names into 3 name lists (one for each table)
widgets to display the 3 tables of advanced data
ROW L, M
display 2 tables of arcline parameters (if flag3=1)
copy general arcline data into data list for table 4
copy tag names into list
copy calibration line data into 4 column data list named arc
set row and column labels
widgets to display the 2 tables of arc data
event handler
procedure GUI
if a version of GUI is already running then do not open second
read the active control file (control_data.txt) from the pipeline directory
run procedure GUIa to dispay the graphical user interface

4.5 PROCEDURE GUO VERSION 1_0
procedure guo
Function View the ASCII files of control data
view fits files of reduced spectra using the ATV utility
run procedure view,/text to save ASCII copies of the median output spectra
run procedure view,/plot to save gif plots of the median output spectra
run procedure view,/ps to save PostScript plots of the median output spectra

Procedures called;
read_control - Returns control data in structure c
ATV - View contents of a output spectra file using ATV utility
View_plot - Plots the output spectra for active fibers and saves copies to file
View_ps - Creates PostScript plots of spectra of active fibers and saves to file
View_text - saves the output spectra for active fibers to individual ASCII files
printl - Prints text message to log file

set of procedures relating to graphical user interface GUO.pro

local function used to set consistent path separator(/)

local procedure used to view a data file using ATV utility
input control data (s) and file name (dfile)
check file exists
load fits file header (head) and array (image)
run ATV using header and image data
local procedure to define bitmap for ATV button
Button event procedures

Select and display a new set of control file
Select new project directory path (pdir)
Set the control parameter flag3=2 for the GUI to display advanced data
Set the control parameter flag3=1 so the GUI to display arcline data
save the displayed control parameter to file (pdir+rbname+'.txt')
run procedure view,/text to save ASCII copies of the median spectra
run procedure view,/plot to save gf plots of the median spectra
run procedure view,/ps to save PostScript plots of the median spectra
close the display
view data file using ATV file viewer
the following 21 event procedures call the procedure atv_view
for a specified output spectra file name
procedure GUOa
Function :Display graphical user interface
:the basic display comprises 10 rows labelled B to K (i.e. guib, guic etc)
:two further rows L and M are opened to display advanced/arcline data

Input :control data (s)

Options s.flag3=0 basic display
s.flag3=1 basic display + arcline dtat tables
s.flag3=2 basic display + advanced parameter tables

get bitmap to be used for ATV button
set array controlling pixel sizing of GUI
Base widget
ROW B
Control file text and button
Project directory text and button
ROW C
No. science text
No. offset text

fiberID text
fibers in gap text
ROW D
skytype text
Skyoset text
Spectrum scaling
Wavelength range text
ROW E
TITLES
ROWS F, G, H, I, J, K
Biasfile and No. text
Flatfile and No. text
Darkfile and No. text
Circfile and No. text
First arcfile and No. text
Second arcfile and No. text
ROWS F, G, H, I, J, K
Fiber attenuation text
Scattered light text
Extraction profile text
Arc lamp atlas text
Wavelength scale (log or linear) text
ROWS F, G, H, I, J
Text boxes for single observation output spectra - only specified number (nstar) are displayed
Star file 0 text
Star file 1 text
Star file 2 text
Star file 3 text
Star file 4 text
ROWS F, G, H, I, J
Star file 5 text
Star file 6 text
Star file 7 text
Star file 8 text
Star file 9 text
ROWS F, G, H, I, J
Median star file
Median offset file
Text box for median offset sky spectra - only shown if offset sky used
Median star in auto mode
Text box for median spectra in auto mode - only shown if file exists
ROW L
Action buttons
Row L, M
display 3 tables of advanced parameters (if flag3=2)
set length of table using No basic parameters (c.flag1)
widgets to display the 3 tables of advanced data
ROW L, M
display 2 tables of arcline parameters (if flag3=1)
copy general arcline data into data list for table 4
widgets to display the 2 tables of arc data
event handler
procedure GUI
if a version of GUI is already running then do not open second
read the active control file (control_data.txt) from the pipeline directory
run procedure GUIa to display the graphical user interface