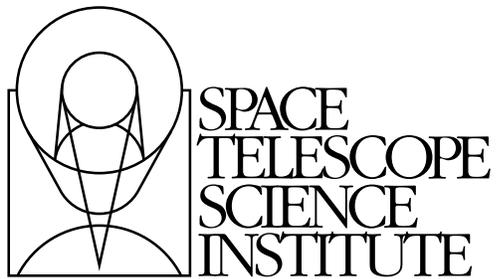

Version 2.0
January 2013

COS Data Handbook



Space Telescope Science Institute
3700 San Martin Drive
Baltimore, Maryland 21218
help@stsci.edu

User Support

For prompt answers to any question, please contact the STScI Help Desk.

- **E-mail:** help@stsci.edu
- **Phone:** (410) 338-1082
(800) 544-8125 (U.S., toll free)

World Wide Web

Information and other resources are available on the COS World Wide Web site:

- **URL:** <http://www.stsci.edu/hst/cos>

COS Revision History

Version	Date	Editor
2.0	January 2013	Derck Massa, Brian York and Svea Hernandez
1.0	March 2009	Brittany Shaw, Derck Massa, and Mary Elizabeth Kaiser

Authorship

This document is written and maintained by the COS/STIS Team in the Instruments Division of STScI with the assistance of associates in the Operations and Engineering Division. Contributions to the current edition were made by T. Ake, A. Aloisi, P. Hodge, M. E. Kaiser, C. Keyes, D. Massa, C. Oliveira, R. Osten, D. Sahnou, B. Shaw, D. Soderblom, and B. York.

In publications, refer to this document as: Massa, D. and York, B. et al. 2013, "COS Data Handbook", Version 2.0, (Baltimore: STScI).

Send comments or corrections to:
Space Telescope Science Institute
3700 San Martin Drive
Baltimore, Maryland 21218
E-mail: help@stsci.edu



Table of Contents

List of Figures	vii
List of Tables	ix
Introduction	xi
Chapter 1: COS Overview	1
1.1 Instrument Capabilities and Design.....	1
1.2 COS Physical Configuration	8
1.2.1 The COS Detectors	10
1.3 Basic Instrument Operations	15
1.3.1 Target Acquisitions	15
1.3.2 Routine Wavecals.....	16
1.3.3 Typical COS Observing Sequence.....	17
1.4 COS Coordinate System.....	17
Chapter 2: COS Data Files	19
2.1 Overview	19
2.2 COS File Names	21
2.3 COS File Structures	24
2.3.1 COS FITS Table Extension Files	24
2.3.2 COS FITS Image Extension Files	24
2.4 COS Data Products	25
2.4.1 Uncalibrated Science Data Files	26
2.4.2 Intermediate Science Data Files	28
2.4.3 Final Science Data Files (and Product Files)	32
2.4.4 Auxiliary Data Files	34
2.5 Data Storage Requirements	48

2.6 Headers, Keywords, and Relationship to Phase II.....	50
2.7 Error and Data Quality Array.....	62
2.7.1 Error Array	62
2.7.2 Data Quality Flags.....	62
Chapter 3: COS Calibration	65
3.1 Raw Data Compilation.....	65
3.2 Pipeline Processing Overview	66
3.3 Calcos : Structure and Data Flow	67
3.4 Descriptions of Spectroscopic Calibration Steps	73
3.4.1 Initialization	73
3.4.2 BRSTCORR: Search for and Flag Bursts	74
3.4.3 BADTCORR: Bad Time Intervals	75
3.4.4 PHACORR: Pulse Height Filter.....	76
3.4.5 WALKCORR: Y-Walk Correction.....	77
3.4.6 RANDCORR: Add Pseudo-Random Numbers to Pixel Coordinates.....	78
3.4.7 TEMPCORR: Temperature-Dependent Distortion Correction.....	79
3.4.8 GEOCORR and IGEOCORR: Geometric Distortion Correction	79
3.4.9 DOPPCORR: Correct for Doppler Shift.....	81
3.4.10 DEADCORR: Nonlinearity Correction	82
3.4.11 FLATCORR: Flat-field Correction.....	84
3.4.12 WAVECORR: Wavecal Correction.....	85
3.4.13 DQICORR: Initialize Data Quality File.....	88
3.4.14 STATFLAG: Report Simple Statistics.....	90
3.4.15 X1DCORR: Locate and Extract 1-D Spectrum	90
3.4.16 BACKCORR: 1D Spectral Background Subtraction.....	94
3.4.17 FLUXCORR/TDSCORR: Conversion to Flux.....	96
3.4.18 HELCORR: Correction to Heliocentric Reference Frame.....	96
3.4.19 Finalization (making the sum files)	97
3.5 Descriptions of Imaging Calibration Steps.....	98
3.6 Customizing COS Data Calibration	98
3.6.1 Mechanics of Tailored Recalibration	99
3.6.2 Using GO Wavecals.....	106

3.7 Reference Files	106
3.7.1 BRSTTAB: Burst Parameters Table	106
3.7.2 BADTTAB: Bad Time Interval Table	107
3.7.3 PHATAB: Pulse Height Discrimination Table	107
3.7.4 PHAFILE: Pulse Height Discrimination File	108
3.7.5 BRFTAB: Baseline Reference Frame Table	108
3.7.6 WALKTAB: Y Walk Correction Table	109
3.7.7 GEOFILE: Geometric Correction File	110
3.7.8 DEADTAB: Deadtime Table	111
3.7.9 FLATFILE: Flat-field File	111
3.7.10 BPIXTAB: Bad Pixel Table	111
3.7.11 LAMPTAB: Template Calibration Lamp Spectra Table	112
3.7.12 WCPTAB: Wavecal Parameter Table	112
3.7.13 DISPTAB: Dispersion Coefficient Table	113
3.7.14 XTRACTAB: 1-D Spectral Extraction Table	114
3.7.15 FLUXTAB: Photometric Throughput Table	115
3.7.16 TDSTAB: Time Dependent Sensitivity Table	116
3.7.17 GSAGTAB: Gain Sag Table	117
3.7.18 HVTAB: High Voltage Table	117
3.7.19 SPWCSTAB: Spectroscopic WCS Parameters Table	118
Chapter 4: COS Error Sources	120
4.1 Overview	120
4.2 Error Sources Associated with Pipeline Processing Steps	120
4.2.1 FUV Dark Count Rate	121
4.2.2 Flat Fields	123
4.2.3 Gain Sag	124
4.2.4 FUV XDL Thermal Drifts	125
4.3 Factors Limiting Flux and Wavelength Accuracy	125
4.3.1 Flux Accuracy	125
4.3.2 Wavelength and Spectral Resolution Accuracies	127

Chapter 5: COS Data Analysis	129
5.1 Data Reduction and Analysis Applications	129
5.1.1 COS-Specific STSDAS Tasks	130
5.1.2 FITS Table Tasks	130
5.1.3 General Spectral Display and Analysis Tasks.....	133
5.2 Evaluating Target Acquisitions and Guiding.....	135
5.2.1 Types of Target Acquisitions	135
5.2.2 Guiding Errors for Single-Guide-Star Mode.....	138
5.3 Working with Extracted Spectra	139
5.3.1 Working With x1d Files in IDL.....	139
5.3.2 Working With x1d Files in Python	139
5.3.3 Working With x1d Files in IRAF/PyRAF	139
5.3.4 Redoing Spectral Extraction	143
5.3.5 Splicing Extracted Spectra	145
5.4 Working with TIME-TAG Data.....	146
5.4.1 Displaying TIME-TAG Data in DS9	147
5.4.2 Filtering Time-Tag Data	149
Appendix A: COS New Spectra Position	152
A.1 COS New Lifetime Position.....	152
Appendix B: COS High Voltage	154
B.1 COS High Voltage History	154

List of Figures

Figure 1.1: Grating Offset Positions (FPPOS).....	7
Figure 1.2: The COS Optical Path and the Locations of the Mechanisms.	8
Figure 1.3: The COS FUV Optical Path.	9
Figure 1.4: The COS NUV Optical Path.	10
Figure 1.5: The FUV XDL Detector.	12
Figure 1.6: COS FUV Stim Pulse.....	12
Figure 1.7: Example of a COS FUV Spectrum.	13
Figure 1.8: Example of a COS FUV Pulse Height Distribution.....	14
Figure 1.9: Example of a COS NUV Spectrum.....	15
Figure 1.10: COS Coordinate Systems.....	18
Figure 2.1: FITS Image Extension File for COS.....	25
Figure 2.2: Overlay of FUV ACCUM Subarrays on FUV TIME-TAG Data.....	26
Figure 2.3: FITS File Format for Raw and corrected TIME-TAG Tables.....	27
Figure 2.4: FITS Array Extension File for COS.....	28
Figure 2.5: FITS File Format for Lampflash Table.....	30
Figure 2.6: FITS File Format for 1-D Extracted Spectrum Table.....	33
Figure 2.7: Sample Association Table 19v221010_asn.....	35
Figure 2.8: COS Support File.....	38
Figure 2.9: FITS File Format for ACQ/SEARCH and ACQ/PEAKD Data.	45
Figure 2.10: FITS File Format for ACQ/IMAGE Data.....	46
Figure 2.11: FITS File Format for jitter Data.....	47
Figure 2.12: FITS File Format for jif Data.	48
Figure 3.1: FUV TIME-TAG Spectroscopic Pipeline Flow Chart.....	69
Figure 3.2: FUV ACCUM Spectroscopic Pipeline Flow Chart.....	70
Figure 3.3: NUV TIME-TAG Spectroscopic Pipeline Flow Chart.....	71
Figure 3.4: NUV ACCUM Spectroscopic Pipeline Flow Chart.....	72
Figure 3.5: NUV Image Pipeline Flow Chart.....	73

Figure 3.6: A Map of the FUV Geometric Correction	80
Figure 3.7: FUV and NUV Deadtime.....	83
Figure 3.8: Flat-field Images of the NUV MAMA Detector.	85
Figure 3.9: The FUV Flat Field.	89
Figure 3.10: FUV Extraction Regions	94
Figure 3.11: NUV Extraction Regions	94
Figure 4.1: Dark Rates	121
Figure 4.2: FUVA Dark	122
Figure 4.3: FUVB Dark	123
Figure 4.4: Gain Sag Effects.....	125
Figure 5.1: Example of an ACQ/SEARCH exposure.....	136
Figure 5.2: Example of an image using the PSA and MIRRORB.....	137
Figure 5.3: BOA and MIRRORA (left), and the BOA and MIRRORB (right).	137

List of Tables

Table 1.1: COS Spectroscopic Modes	2
Table 1.2: Wavelength Ranges for FUV Gratings for FPPOS = 3	3
Table 1.3: Wavelength Ranges for NUV Gratings	4
Table 1.4: COS Detector Characteristics	11
Table 2.1: Data Types and File Naming Conventions	22
Table 2.2: Columns of a Raw TIME-TAG Data Table	27
Table 2.3: Columns of a COS corrtag Table	29
Table 2.4: Columns of a COS Lampflash Table	31
Table 2.5: Columns of a COS Extracted Spectrum Table	33
Table 2.6: Member Types in COS Associations	34
Table 2.7: ACQ/IMAGE Header Keywords.	39
Table 2.8: ACQ/SEARCH Header Keywords.	40
Table 2.9: ACQ/PEAKXD Header Keywords.	41
Table 2.10: ACQ/PEAKD Header Keywords.	43
Table 2.11: Columns of an ACQ/SEARCH or ACQ/PEAKD Table	44
Table 2.12: Columns of a jitter Table	47
Table 2.13: COS Pipeline Data Volumes per Exposure	49
Table 2.14: COS Pipeline Data Volumes per Calibrated Exposure	49
Table 2.15: Selected Header Keywords and Relationship to Phase II Parameters	51
Table 2.16: Spectroscopic Calibration Switch Keywords	60
Table 2.17: Imaging Calibration Switch Keywords	61
Table 2.18: Reference File Keywords	61
Table 2.19: COS Data Quality Flags	64
Table 3.1: Variables used in 1-D Spectral Extraction	91
Table 3.2: Arguments for Running calcos in PyRAF	104
Table 3.3: Arguments for Running calcos in Python:	105

Table 3.4: Command-line Options for Running calcos in Unix/Linux/Mac:	105
Table 3.5: BRSTTAB Table Contents	107
Table 3.6: BADTTAB Table Content	107
Table 3.7: PHATAB Table Contents	108
Table 3.8: BRFTAB Table Contents	109
Table 3.9: WALKTAB Table Contents	110
Table 3.10: BPIXTAB Table Content	112
Table 3.11: LAMPTAB Table Contents	112
Table 3.12: WCPTAB Table Contents	113
Table 3.13: DISPTAB Table Format	114
Table 3.14: XTRACTAB Table Format	115
Table 3.15: FLUXTAB Table Format	116
Table 3.16: TDSTAB Table Format	116
Table 3.17: GSAGTAB Table Format	117
Table 3.18: HVTAB Table Format	118
Table 3.19: SPWCSTAB Table Format	119
Table 5.1: COS-Specific Tasks	130
Table 5.2: Spectral Analysis Tasks	133
Table 5.3: Useful IRAF Tasks for Reducing TIME-TAG Data	147

Introduction

How to Use this Handbook

This handbook is designed to help users manipulate, process, and analyze data from the Cosmic Origins Spectrograph (COS) which was installed on the Hubble Space Telescope (HST) during the 2009 servicing mission (SM4). It is designed for users familiar with HST data but new to COS.

The current edition of the COS Data Handbook was completed in Jul. 2012. It is presented as an independent and self-contained document, referred to as the “COS Data Handbook”.

For detailed information on the capabilities of the instrument, and how to plan observations, users should refer to the *COS Instrument Handbook*. For further information and timely updates, users should consult the COS Web page (<http://www.stsci.edu/hst/cos>), especially the [Document Archive link](#). In particular, the [STScI Analysis Newsletters](#) (STANs) highlight changes in code and calibration procedures and provide other instrument-related news. The [Instrument Science Reports](#) (ISRs) present in-depth characterizations of the instrument and detailed explanations of calibration code and procedures.

Handbook Structure

The *COS Data Handbook* is organized in five chapters, which discuss the following topics:

- [Chapter 1: COS Overview](#) provides a brief overview of the instrument and its operational capabilities.
- [Chapter 2: COS Data Files](#) describes the contents of COS data files, the meanings of selected header keywords, and the relationship of the data products to the original Phase II proposal.
- [Chapter 3: COS Calibration](#) describes how the calibration pipeline processes observations, the content of COS reference files used during calibration and how to run the calibration pipeline locally.

- [Chapter 4: COS Error Sources](#) describes the sources of uncertainty and limiting accuracies of COS data. COS observers should read this chapter to acquaint themselves with the limitations of the data that may remain after pipeline calibration.
- [Chapter 5: COS Data Analysis](#) describes certain **IRAF/PyRAF/STSDAS** tasks, and other software packages useful for optimizing data products and analyzing the data. In particular, it discusses software tools that can be applied to specific types of data and data formats. It describes how to analyze target acquisitions and guide star tracking. It provides descriptions of different kinds of data and gives detailed instructions on how to work with them; specifically: extracted spectra, and TIME-TAG data.

There are some important pieces of general information about *HST* data, the *HST* Archive, and the **IRAF** and **STSDAS** analysis software that are not specific to the COS, and which are therefore not discussed in the COS specific section. Users are referred to a companion volume, *Introduction to HST Data Handbooks*. In particular, Chapter 1, Chapter 2 and Chapter 3 of the *Introduction to HST Data Handbooks* describe how to retrieve and read *HST* data, *HST* file formats, and the basics of the **STSDAS** software package. Chapter 4 offers an **IRAF** primer. Chapter 5 describes *HST* file name conventions and exposure “associations”. Chapter 6 describes *HST* Observation Logs. Additional help with *HST* data is always available via email to the STScI Help Desk at help@stsci.edu.

Because COS is a relatively new instrument, our characterizations of it are still changing. Consequently, readers are advised to consult the COS Web pages (<http://www.stsci.edu/hst/cos/>) for the latest updates.

Typographic Conventions

To help you understand the material in this Data Handbook, we will use a few consistent typographic conventions.

Visual Cues

The following typographic cues are used:

- **bold** words identify an **STSDAS**, **IRAF**, or **PyRAF** task or package name.
- typewriter-like words identify a file name, system command, or response that is typed or displayed.
- *italic* type indicates a new term, an important point, a mathematical variable, or a task parameter.
- SMALL CAPS identifies a header keyword.
- ALL CAPS identifies a table column.

Comments

Occasional side comments point out three types of information, each identified by an icon in the left margin.



Warning: You could corrupt data, produce incorrect results, or create some other kind of severe problem.



Heads Up: Here is something that is often done incorrectly or that is not obvious.



Tip: No problems...just another way to do something or a suggestion that might make your life easier.



Information especially likely to be updated on the COS Web site is indicated by this symbol.

COS Overview

In this chapter...

1.1 Instrument Capabilities and Design / 1
1.2 COS Physical Configuration / 8
1.3 Basic Instrument Operations / 15
1.4 COS Coordinate System / 17

1.1 Instrument Capabilities and Design

The [Cosmic Origins Spectrograph](#) (COS) is an *HST* fourth generation spectrometer, designed to enhance the spectroscopic capabilities of *HST* at ultraviolet (UV) wavelengths. COS was built by Ball Aerospace Corporation to the specifications of Dr. James Green, the Principal Investigator (PI), at the University of Colorado at Boulder in conjunction with the COS Instrument Definition Team (IDT). Designed to primarily observe faint point sources, COS is optimized for maximum throughput, and provides moderate and low resolution spectroscopy in the UV and limited imaging in the NUV.

COS is a slitless spectrograph that employs two circular 2.5 arcsec diameter science apertures, the Primary Science Aperture (PSA) and the Bright Object Aperture (BOA). The PSA is an open aperture and the BOA contains a neutral density filter to attenuate the flux of bright objects. COS also contains two calibration apertures, the Wavelength Calibration Aperture (WCA) and the Flat-Field Calibration Aperture (FCA). Light from external sources does not reach these apertures. Instead they are illuminated by internal calibration lamps. The FCA is not available for observers, but the WCA can be used by observers to obtain wavelength calibration spectra. The WCA can be illuminated by one of two Pt-Ne wavelength calibration lamps. Similarly, the FCA can be illuminated by one of two deuterium flat-field calibration lamps; however, this is restricted to observatory calibration programs.

The instrument has two channels: a far-ultraviolet (FUV) channel that is sensitive across the 900-2150 Å wavelength range and a near-ultraviolet (NUV) channel that provides wavelength coverage from 1650-3200 Å. The COS optical design achieves

its high performance, particularly in the FUV, by minimizing the number of reflections in the optical path and the use of large format detectors which maximize the wavelength coverage per exposure. Each channel has its own photon-counting detector and a selection of gratings (Table 1.1). The NUV channel also has a mirror that can be used in two modes for imaging. The FUV channel uses a single reflection system where a high-efficiency, first-order, aspheric holographic grating completely corrects the beam in the dispersion direction but has low spatial resolution perpendicular to dispersion. *Only one channel may be used at a time.*

Table 1.1: COS Spectroscopic Modes

Grating	Normal wavelength range (Å) ¹	Bandpass per exposure (Å)	Resolving Power $R = \lambda/\Delta\lambda$	Dispersion (mÅ pixel ⁻¹)
FUV Channel				
G130M	900 ² – 1450	300	16,000 – 21,000	9.97
G160M	1405 – 1775	370	16,000 – 21,000	12.23
G140L	900 – 2150 ³	>1100	1,500 – 4,000	80.3
NUV Channel				
G185M	1700 – 2100	3 × 35	16,000 – 20,000	34
G225M	2100 – 2500	3 × 35	20,000 – 24,000	34
G285M	2500 – 3200	3 × 41	20,000 – 24,000	40
G230L	1700 – 3200	(1 or 2) × 400	2,100 – 3,200	390

1. Normal wavelength ranges are for the primary (default) central wavelength setting.

2. Modes covering wavelengths below 1130 Å are expected to have a much reduced resolving power of about 2000.

3. Second order contamination possible longward of 2150 Å.

FUV Spectroscopy

The FUV channel employs a large format cross delay line (XDL) detector consisting of two 16384 x 1024 pixel segments, referred to as FUV segments A and B. The segments are separated by a physical gap of 9 mm, which makes it impossible to obtain a continuous spectrum across the two segments with a single setting. The supported central wavelength positions were selected to enable full wavelength coverage of the gap. Table 1.2 shows the wavelength ranges of both segments for all possible FUV grating and central wavelength combinations.

Table 1.2: Wavelength Ranges for FUV Gratings for FPPOS = 3

Grating	Central wavelength setting (Å)	Recorded wavelengths	
		Segment B	Segment A
G130M	1055 ¹	900 - 1041	1055 - 1197
	1096 ¹	940 - 1081	1096 - 1237
	1222 ¹	1068 - 1208	1223 - 1364
	1291	1132 - 1274	1291 - 1433
	1300	1141 - 1283	1300 - 1442
	1309	1150 - 1292	1309 - 1451
	1318	1159 - 1301	1318 - 1460
	1327	1168 - 1310	1327 - 1469
G160M	1577	1382 - 1556	1577 - 1752
	1589	1394 - 1568	1589 - 1764
	1600	1405 - 1579	1600 - 1775
	1611	1416 - 1590	1611 - 1786
	1623	1428 - 1602	1623 - 1798
G140L ²	1105	N/A ³	1105 - 2253
	1230 ⁴	<300 - 1095	1230 - 2378
	1280	<500 - 1165	1280 - 2405

1. These are new modes that are currently being characterized, so these numbers are preliminary. 1055 and 1096 central wavelength settings have been offered starting from Cycle 19 and 1222 central wavelength mode starting from Cycle 20.
2. It is not yet clear how much of the G140L segment B short wavelength (< 900 Å) ranges will be available due to uncertainties in the HST OTA throughput. This is currently being investigated.
3. The G140L grating and 1105 central wavelength setting moves the zero-order image onto segment B. Therefore, only segment A is available for this setting.
4. Beginning in Cycle 18, the G140L 1230 setting was replaced by the 1280 setting.

NUV Spectroscopy

To retain efficiency utilizing the square format of the NUV detector, three mirrors simultaneously image three, fully aberration-corrected, spectra onto a single 1024 x 1024 Multi-Anode Micro-channel Array (MAMA) detector. Consequently, three

separate regions of the spectrum are imaged onto the detector. These spectral regions, referred to as stripes A, B, and C, each span the physical length of the detector in the dispersion direction - but are not contiguous in wavelength space. The allowable grating positions were defined with two objectives: the capability of obtaining full spectral coverage over the NUV bandpass and maximizing scientific return with a minimum number of grating positions. As a result, several of the supported central wavelength positions were selected to maximize the number of diagnostic lines on the detector in a single exposure. [Table 1.3](#) shows the wavelength ranges of the three stripes for all possible NUV grating and central wavelength combinations

Table 1.3: Wavelength Ranges for NUV Gratings

Grating	Central wavelength setting (Å)	Recorded wavelengths		
		Stripe A	Stripe B	Stripe C
G185M	1786	1670 – 1705	1769 – 1804	1868 – 1903
	1817	1701 – 1736	1800 – 1835	1899 – 1934
	1835	1719 – 1754	1818 – 1853	1916 – 1951
	1850	1734 – 1769	1833 – 1868	1931 – 1966
	1864	1748 – 1783	1847 – 1882	1945 – 1980
	1882	1766 – 1801	1865 – 1900	1964 – 1999
	1890	1774 – 1809	1872 – 1907	1971 – 2006
	1900	1783 – 1818	1882 – 1917	1981 – 2016
	1913	1796 – 1831	1895 – 1930	1993 – 2028
	1921	1804 – 1839	1903 – 1938	2002 – 2037
	1941	1825 – 1860	1924 – 1959	2023 – 2058
	1953	1837 – 1872	1936 – 1971	2034 – 2069
	1971	1854 – 1889	1953 – 1988	2052 – 2087
	1986	1870 – 1905	1969 – 2004	2068 – 2103
	2010	1894 – 1929	1993 – 2028	2092 – 2127

Grating	Central wavelength setting (Å)	Recorded wavelengths		
		Stripe A	Stripe B	Stripe C
G225M	2186	2070 – 2105	2169 – 2204	2268 – 2303
	2217	2101 – 2136	2200 – 2235	2299 – 2334
	2233	2117 – 2152	2215 – 2250	2314 – 2349
	2250	2134 – 2169	2233 – 2268	2332 – 2367
	2268	2152 – 2187	2251 – 2286	2350 – 2385
	2283	2167 – 2202	2266 – 2301	2364 – 2399
	2306	2190 – 2225	2288 – 2323	2387 – 2422
	2325	2208 – 2243	2307 – 2342	2406 – 2441
	2339	2223 – 2258	2322 – 2357	2421 – 2456
	2357	2241 – 2276	2340 – 2375	2439 – 2474
	2373	2256 – 2291	2355 – 2390	2454 – 2489
	2390	2274 – 2309	2373 – 2408	2472 – 2507
	2410	2294 – 2329	2393 – 2428	2492 – 2527
G285M	2617	2480 – 2521	2596 – 2637	2711 – 2752
	2637	2500 – 2541	2616 – 2657	2731 – 2772
	2657	2520 – 2561	2636 – 2677	2751 – 2792
	2676	2539 – 2580	2655 – 2696	2770 – 2811
	2695	2558 – 2599	2674 – 2715	2789 – 2830
	2709	2572 – 2613	2688 – 2729	2803 – 2844
	2719	2582 – 2623	2698 – 2739	2813 – 2854
	2739	2602 – 2643	2718 – 2763	2837 – 2878
	2850	2714 – 2755	2829 – 2870	2945 – 2986
	2952	2815 – 2856	2931 – 2972	3046 – 3087
	2979	2842 – 2883	2958 – 2999	3073 – 3114
	2996	2859 – 2900	2975 – 3016	3090 – 3131
	3018	2881 – 2922	2997 – 3038	3112 – 3153
	3035	2898 – 2939	3014 – 3055	3129 – 3170
	3057	2920 – 2961	3036 – 3077	3151 – 3192
	3074	2937 – 2978	3053 – 3094	3168 – 3209
3094	2957 – 2998	3073 – 3114	3188 – 3229	

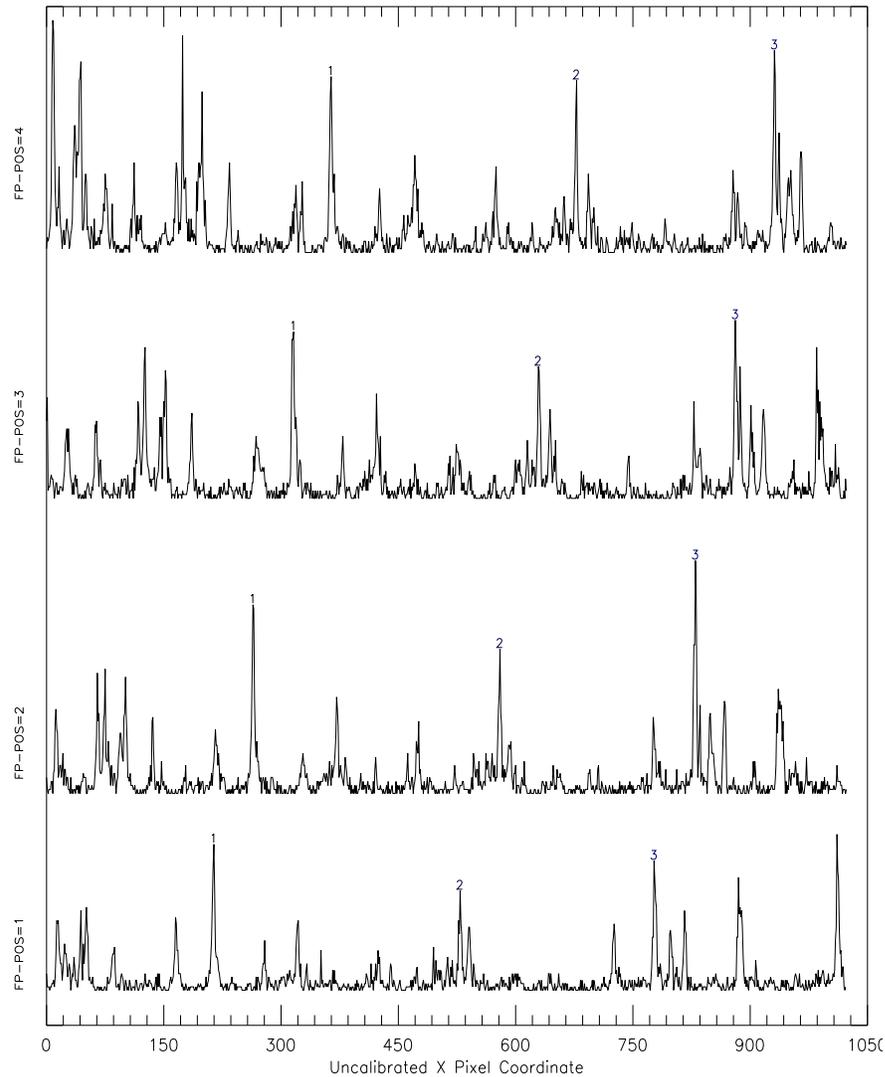
Grating	Central wavelength setting (Å)	Recorded wavelengths		
		Stripe A	Stripe B	Stripe C
G230L	2635	1334 – 1733 ¹	2435 – 2834	1768 – 1967²
	2950	1650 – 2050	2750 – 3150	1900 – 2100²
	3000	1700 – 2100	2800 – 3200	1950 – 2150²
	3360	2059 – 2458	3161 – 3560 ³	2164 – 2361²

1. The wavelengths listed for central wavelength 2635 Å in stripe A are for completeness only and also in case a bright emission line falls onto the detector. Note that the NUV detector's sensitivity at these wavelengths is extremely low. To obtain a low-resolution spectrum at wavelengths below about 1700 Å we recommend G140L and the FUV channel.
2. The values in shaded cells are wavelength ranges as seen in second-order light. In these cases the achieved dispersion is twice that for first-order mode. Note, however, that some first order light may contaminate the spectrum depending on the SED of the target.
3. The spectrum longward of 3400 Å may be contaminated by second order spectrum, depending on the SED of the target.

Grating Offset Positions (FPPOS)

For each NUV and FUV central wavelength setting there are four grating offset positions (FPPOS=1-4) available to move the spectrum slightly in the dispersion direction. This allows the spectrum to fall on different areas of the detector to minimize the effects of small scale fixed pattern noise in the detector. [Figure 1.1](#) shows the shifts in uncalibrated x pixel coordinates of the stripe B spectra for all four FPPOS positions.

Figure 1.1: Grating Offset Positions (FPPOS)



This figure shows spectra obtained at all four $FPPOS$ positions using the G185M grating with a central wavelength setting of 1850. The individual plots show the collapsed counts from the stripe B spectra versus the uncalibrated x pixel coordinates. Note that the three features marked 1, 2, and 3, shift slightly for each $FPPOS$ position.

NUV Imaging

COS imaging may only be done with the NUV channel and the spectral coverage includes the entire NUV bandpass from ~ 1650 - 3200 Å. This mode utilizes a flat mirror with two available mirror settings, MIRRORA and MIRRORB. The first setting uses a primary reflection off the mirror surface, and the second setting provides an attenuated reflection. MIRRORB and/or the BOA may be used to obtain images of brighter objects, but MIRRORB produces a secondary image and the BOA produces an image with coma that degrades the spatial resolution (Figure 5.2 and Figure 5.3).

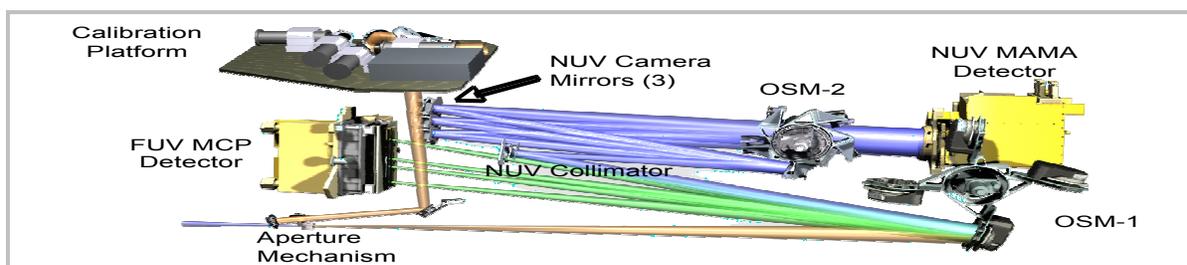
While the spatial resolution of COS NUV MIRRORA (Section 1.2) images can be quite good, the field of view is very small. Furthermore, because the optics image the sky onto the detector – not the aperture – the image includes some light from sources out to a radius of about 2 arcsec. However, only point sources within about 0.5 arcsec of the aperture center have essentially all their light imaged, and so the photometric interpretation of a COS image can be inherently complex.

Data Collection Modes

COS has two modes of data collection, TIME-TAG and ACCUM, and only one mode can be used for a given observation. In TIME-TAG mode the position, time, and for FUV, pulse height of each detected photon are tabulated into an events list, while in ACCUM mode the photon events are integrated onboard into an image. TIME-TAG data have a time resolution of 32 ms, and can be screened as a function of time during the post-observation pipeline processing to modify temporal sampling and exclude poor quality data. COS is optimized to perform in TIME-TAG mode, although ACCUM mode is fully supported in the pipeline processing. ACCUM mode should be used primarily for UV bright targets that can not be observed in TIME-TAG mode due to high count rates. Users should note that FUV data taken in ACCUM mode use sub-arrays since the 18MB of onboard memory cannot hold a complete FUV image (containing both detector segments). ACCUM mode omits only the wavecal region and unused detector space, therefore the FUV ACCUM subarrays contain all of any external spectrum. The FUV ACCUM subarrays, whose sizes are 16384 x 128, are shown in Figure 2.2.

1.2 COS Physical Configuration

Figure 1.2: The COS Optical Path and the Locations of the Mechanisms.



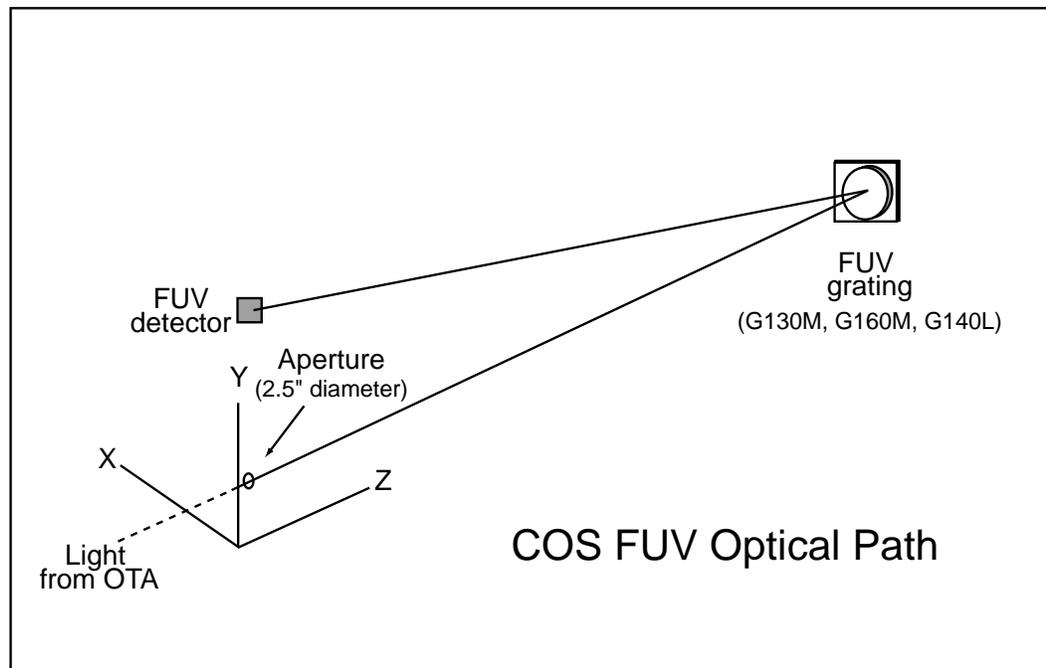
Scaled with all elements shown in their correct relative locations.

The COS optical design includes an external shutter, two science apertures, two calibration apertures, two Optics Select Mechanisms (OSM1 and OSM2), and separate NUV and FUV detectors. COS also has an independent calibration lamp assembly containing two Pt-Ne and two deuterium lamps, which can illuminate the

detectors with an emission line or a continuum spectrum, respectively. The COS optical design and elements are displayed in [Figure 1.2](#).

External light enters the aperture mechanism through either the PSA or the BOA and illuminates OSM1, which contains the three FUV gratings and a mirror. Each grating can be set to one of several positions, to obtain different wavelength ranges. The positioning of the OSM1 mechanism is not precisely repeatable, and this can cause small, but significant, variations in how the spectrum or image is projected onto the detector. This non-repeatability can be corrected in post-observation data processing using separate or concurrent (TAGFLASH) calibration lamp exposures (wavecals). The FUV gratings correct for aberration in the dispersion direction only, and disperse the incoming light onto the FUV XDL detector. The COS FUV channel optical path is illustrated in [Figure 1.3](#)

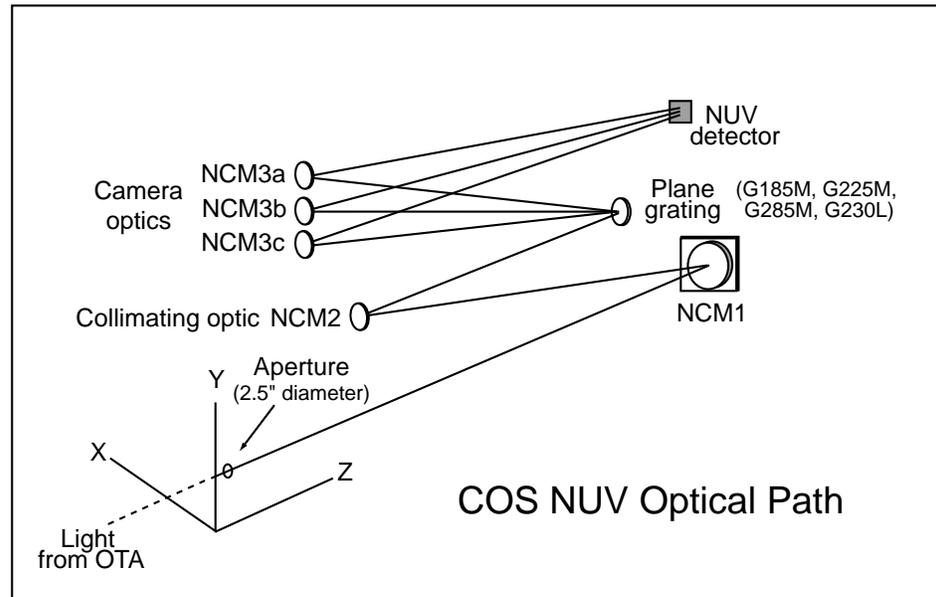
Figure 1.3: The COS FUV Optical Path.



If the OSM1 is set to the mirror position, the incoming light is directed to a collimating mirror, and then to OSM2, which contains a mirror for imaging and the four NUV gratings. Each grating offers multiple positions. As is the case with OSM1, the positioning of OSM2 does not repeat exactly, and the data need to be corrected in post-observation data processing via either separate or concurrent wavecals. If a grating is in place on OSM2, the dispersed light is imaged onto the NUV detector by three separate parallel camera mirrors (NCM3a, b, c). This results in three spectra, or stripes, covering different wavelength ranges. Full wavelength coverage may be obtained through multiple observations with different grating positions. Alternatively, if the plane mirror is in place on OSM2, the undispersed light is sent to the middle camera mirror (NCM3b) and then imaged onto the NUV detector. The plane mirror on OSM2 may be used in either of two settings, designated as MIRRORA and

MIRRORB. The MIRRORA setting employs a direct reflection from the plane mirror. For the MIRRORB setting, the plane mirror is slightly offset to provide primary reflection off the front surface of its coating and hence an attenuation factor of approximately 25 compared to the MIRRORA setting. The COS NUV channel optical path is illustrated in [Figure 1.4](#)

Figure 1.4: The COS NUV Optical Path.



A series of beam-splitters and fold mirrors direct light from the calibration lamp assembly (see [Figure 1.2](#)), through either the WCA or FCA and into the optical path. The calibration lamp assembly can provide continuum illumination with its deuterium lamps and emission line illumination with its Pt-Ne lamps to both the NUV and FUV detectors. The Pt-Ne lamps may be operated during science exposures in order to produce concurrent wavelength calibrations (TAGFLASH mode).

1.2.1 The COS Detectors

COS uses two detectors, a FUV XDL and a NUV MAMA. [Table 1.4](#) gives an overview of their characteristics.

Table 1.4: COS Detector Characteristics

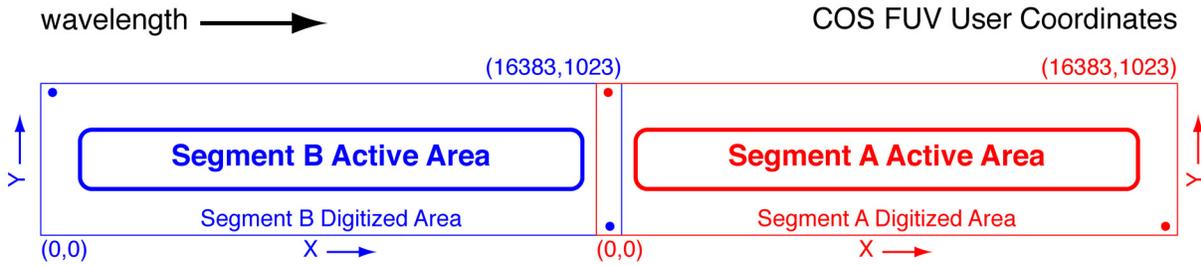
Detector Characteristic	FUV XDL	NUV MAMA
Photocathode	CsI (opaque)	Cs ₂ Te (semi-transparent)
Window	None	MgF ₂ (re-entrant)
Wavelength range	900 – 2050 Å	1650 – 3200 Å
Active area	85 × 10 mm ¹	25.6 × 25.6 mm
Pixel format (full detector)	16384 × 1024 ¹	1024 × 1024
Image size recorded per spectrum	16384 × 128 (ACCUM) ¹ 16384 × 1024 (TIME-TAG) ¹	1024 × 1024
Pixel size	6 × 24 μm 0.023 × 0.092 arcsec	25 × 25 μm 0.025 × 0.025 arcsec
Spectral resolution element size (= “resel”)	6 × 10 pix	3 × 3 pix
Plate scale: Along dispersion (per resel)	0.13 arcsec	0.075 arcsec
Plate scale: Cross dispersion (per resel)	0.92 arcsec	0.075 arcsec
Plate scale: Imaging (per resel)	N/A	0.075 arcsec
Quantum efficiency	~26% at 1335 Å ~12% at 1560 Å	~10% at 2200 Å ~8% at 2800 Å
Dark count rate ²	1.25 cnt s ⁻¹ cm ⁻² 1.80 × 10 ⁻⁶ cnt s ⁻¹ pix ⁻¹ 1.1 × 10 ⁻⁴ cnt s ⁻¹ resel ⁻¹	117 cnt s ⁻¹ cm ⁻² 7.3 × 10 ⁻⁴ cnt s ⁻¹ pix ⁻¹ 6.6 × 10 ⁻³ cnt s ⁻¹ resel ⁻¹

1. Sizes given are for an individual FUV segment.
2. NUV dark rate is time dependent.

FUV Channel

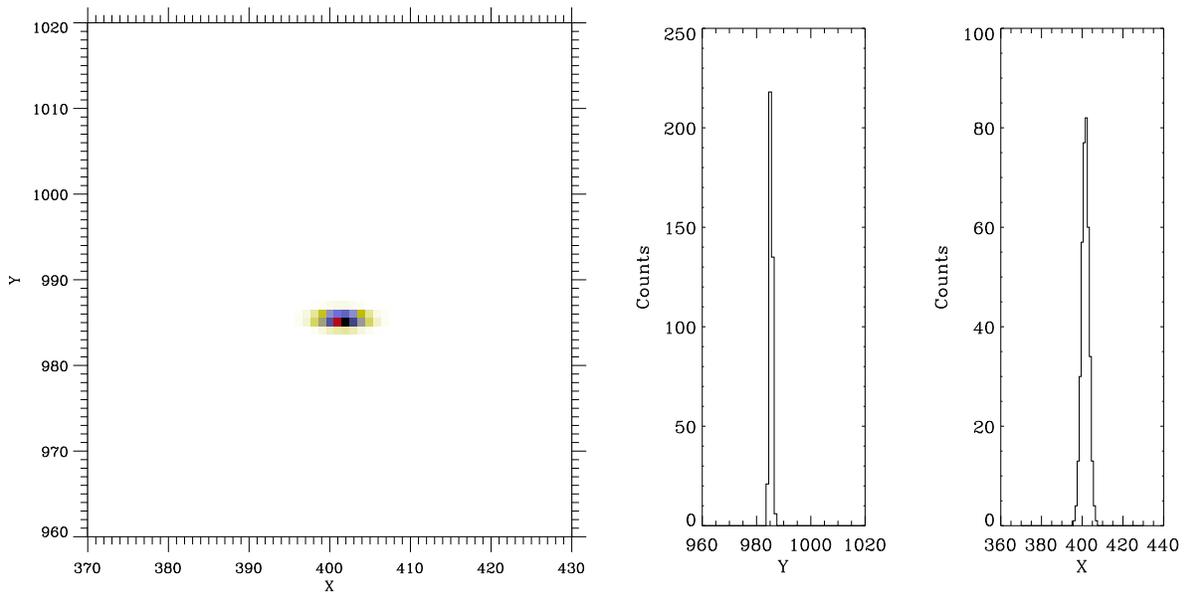
The FUV channel uses a large-format, windowless solar-blind cross delay line (XDL) detector. This is a two-segment photon-counting detector with microchannel plates feeding a XDL anode. The data are digitized to a 16384 × 1024 pixel format for each segment; however the active area is only 14200 × 540 for Segment A (FUVA) and 14150 × 400 for Segment B (FUVB). Because there are no physical pixels, fiducial electronic pulses are recorded at specific times throughout an observation to permit alignment of data to a standard reference frame. These electronic pulses are referred to as “stim pulses”. Figure 1.5 schematically shows the COS FUV XDL segments with the locations of the active areas and stim pulses. When active, the stim pulses emulate counts located near the edges of the anode, beyond the illuminated portions of the detector. A zoomed-in image of one of the FUV stim pulses on segment B is shown in Figure 1.6. An example of an FUV external science spectrum taken with Segment B is shown in Figure 1.7, with a simultaneous wavelength calibration spectrum.

Figure 1.5: The FUV XDL Detector.

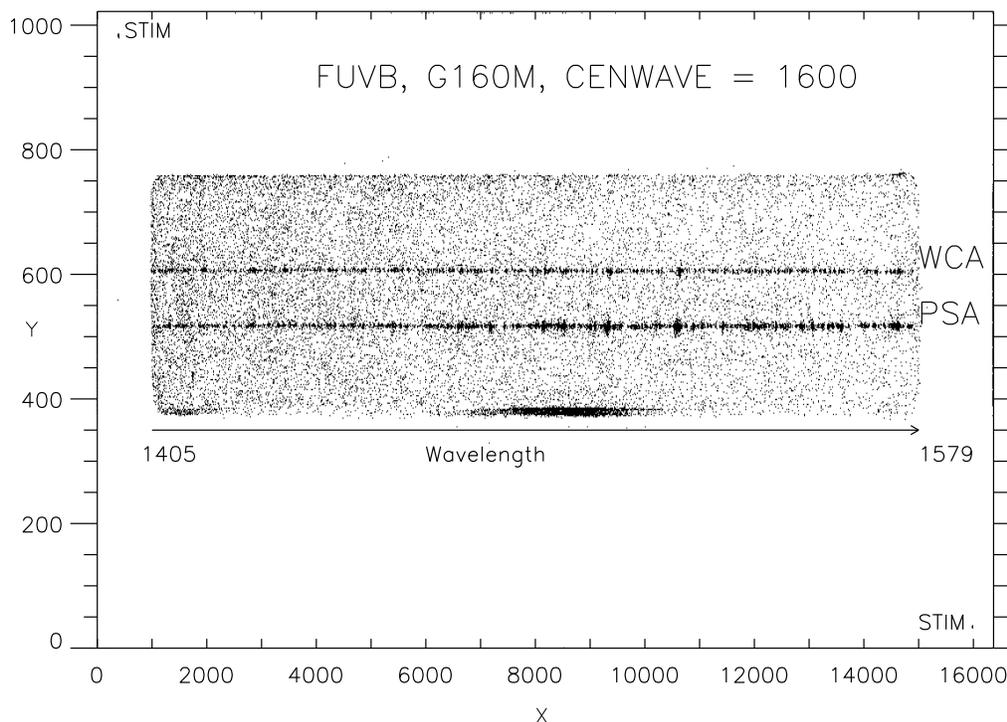


Drawn to scale. The slight curvature at the corners of the active areas is also present on the flight detectors. The red and blue dots show the approximate locations of the stim pulses. The numbers in parentheses are the pixel coordinates at the corners of the segment's digitized area.

Figure 1.6: COS FUV Stim Pulse



Left: A portion of an image in the FUV detector with a typical stim pulse is shown. Right: A histogram of the stim pulse profile in the x and y direction. The electronic stim pulses are used to remove thermal distortions and to map the XDL detector elements to a standard reference frame.

Figure 1.7: Example of a COS FUV Spectrum.

Wavelength calibration spectra for FUV segment B with G160M at 1600 obtained during ground testing. The upper spectrum is from the internal wavelength calibration lamp obtained through the WCA. The lower spectrum is from an external lamp obtained through the PSA. The bright streak at the bottom is due to an area of enhanced background on the detector segment. Note that the size of the active area is somewhat less than the overall digitized area, and that the Y axis has been stretched. The STIMs are also visible in the upper left and lower right corners.

With each recorded event on the XDL detector, the total charge in the associated electron cloud incident on the anode is recorded. For FUV TIME-TAG data this pulse height amplitude (PHA) is sent to the ground along with the position of the event and can be used during data analysis to identify non-photon events. For FUV XDL ACCUM mode data, only an integrated pulse height distribution (a histogram of the PHA data) for the entire segment is available, see [Figure 1.8](#).

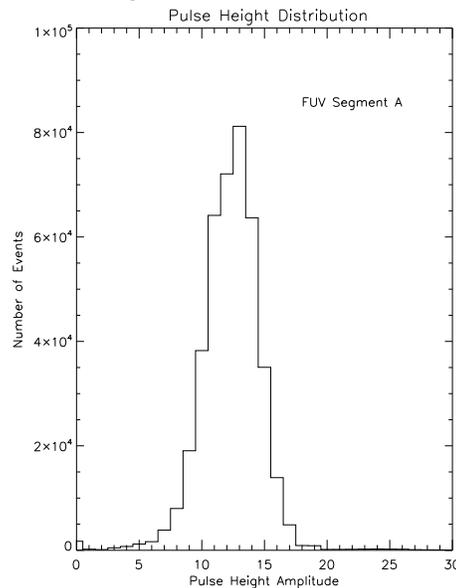
A photon landing on an FUV detector segment creates an event (a cascade of electrons) at the backside of the detector which is characterized by a pulse height amplitude (PHA) that is detected by the electronics. The detector electronics distinguishes between real and electronic noise events by the value of the PHA, with noise events having low PHAs and real events large PHAs. However, as a portion of the detector is exposed to more and more light, the PHAs that it produces become smaller, an effect called “gain sag”. Gain sag results in two effects: *the mis-registration of the event positions* and *localized sensitivity loss*.

Mis-registration of event positions as a function of PHA is termed “walk”. Walk has been identified in both the dispersion (X) and cross-dispersion (Y) directions.

Currently, a simple Y-walk correction is made by the COS calibration pipeline (see [Section 3.4.5](#)), and work has begun on a better correction for both X and Y.

Localized sensitivity loss occurs when the PHAs for some pixels become too small to be distinguished from background events, causing events to be missed or filtered out. This results in a localized region of low sensitivity. Eventually, the PHAs of all of the pixels in a region become so small that photons landing on that location no longer create events with non-zero PHAs. In that case, no events are registered and the region is termed a “dead spot”. When this occurs, it is necessary to either increase the high voltage applied to the detector (which increases the PHAs of all the pixels), or to move the aperture so that the science spectra land on a different portion of the detector (which has not been exposed to as much light). The COS FUV detectors have already experienced localized gain sag on regions of the FUVB and FUVA detectors exposed to the bright Ly α airglow line when the G130M and G140L are used respectively. As a result, the detector high voltage was increased on FUVB on 10 March 2011 and on FUVA on March 26 2012. Furthermore, beginning on July 23 2012 the default location for the science spectra and target acquisitions on the detector was moved to a different position, also known as “new lifetime position”. For more information on the new lifetime position consult [Appendix A.1](#).

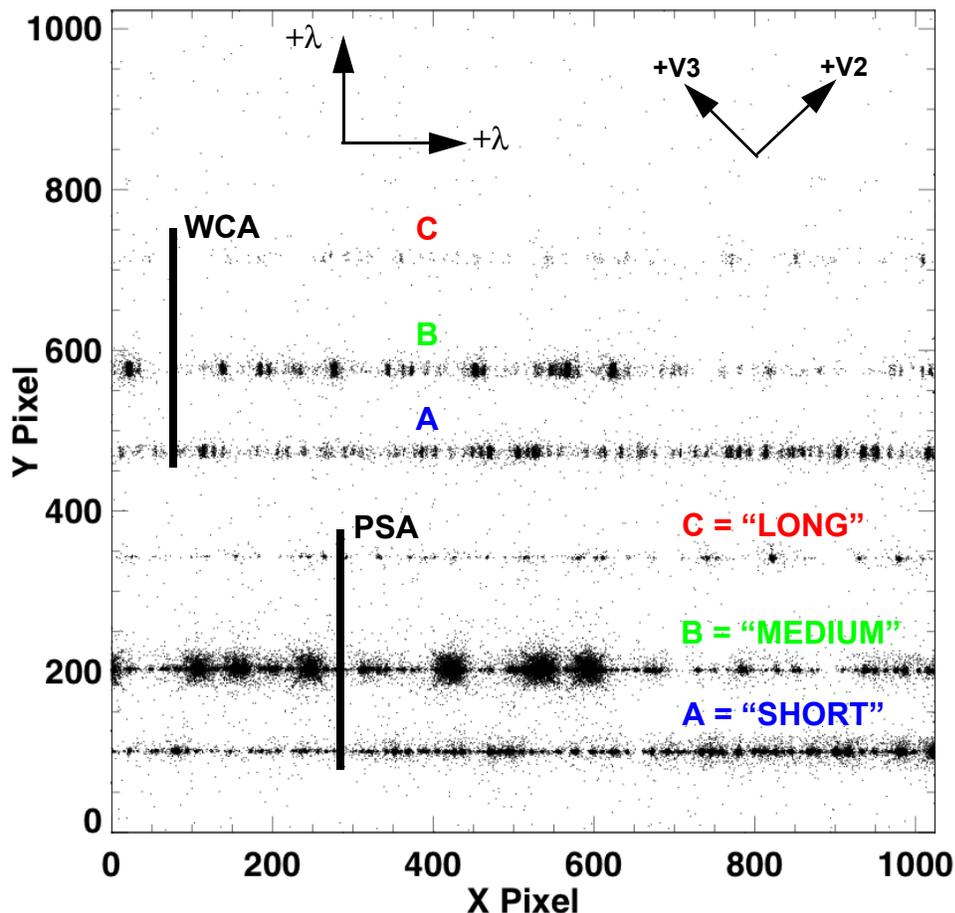
Figure 1.8: Example of a COS FUV Pulse Height Distribution



NUV Channel

The NUV channel uses a 1024 x 1024 pixel Multi-Anode Micro-channel Array (MAMA) detector. This has a semi-transparent cesium telluride photocathode on a magnesium fluoride window, which allows detection of photons with wavelengths from 1150 to 3200 Å. The NUV MAMA provides no pulse-height information, but may be used in both ACCUM and TIME-TAG mode. The NUV channel creates three spectrum stripes on the MAMA detector, resulting in three separate stripes for the science data and three for wavelength calibration data as shown in [Figure 1.9](#).

Figure 1.9: Example of a COS NUV Spectrum.



Wavelength calibration spectra obtained from the internal source through the WCA (upper three stripes) and an external source through the PSA (lower three stripes). The stripes are designated A, B, and C, in going from bottom to top for each source. Wavelength increases from left to right in each stripe and from bottom to top (hence the SHORT, MEDIUM, and LONG designations).

1.3 Basic Instrument Operations

1.3.1 Target Acquisitions

The details of acquiring objects with COS are described in Chapter 7 of the *COS Instrument Handbook*. In brief, the COS flight software provides several methods for acquiring and centering a target in the aperture in both imaging and dispersed light modes. The simplest and fastest method uses the ACQ/IMAGE command to obtain a direct NUV image of the target field and then moves the telescope to the centroid of the measured light. This is the preferred method, but the target coordinates must be

accurate enough to ensure that it falls within the aperture after the initial pointing of the telescope. With less accurate coordinates, a spiral search (ACQ/SEARCH) should be performed with either detector prior to other acquisition methods to ensure the target will fall within the aperture. The other COS acquisition methods (ACQ/PEAKXD and ACQ/PEAKD) use dispersed light from the target, and can also be performed with either detector.

1.3.2 Routine Wavecals

Routine wavelength calibration exposures, or wavecals, are needed by the COS calibration pipeline, **calcos**, to compensate for the effects of OSM drifts. All wavelength calibration exposures are taken in TIME-TAG mode. They may be obtained in either the TAGFLASH mode, where FLASH=YES for TIME-TAG science observations, or in separate wavelength calibration exposures that are either automatic or user-specified.

For TAGFLASH exposures, the wavecal lamp is turned on briefly at the start of an externally targeted exposure, and again at predefined intervals throughout the exposure. In this mode, photons from the external science target and the internal wavelength calibration source are recorded simultaneously on different portions of the detector; see Figures 1.7 and 1.9.

For TIME-TAG exposures not done in TAGFLASH mode, a separate wavecal exposure will be automatically performed (AUTO wavecal) for each set of external spectrographic science exposures using the same spectral element, central wavelength, and FPPOS value. These automatic wavecals are performed after the first such science exposure and after each subsequent science exposure if more than 40 minutes of visibility time has elapsed since the previous wavecal and the same spectrograph set-up has been in use over that time.

Observers also have the ability to insert additional wavecals by specifying TARGET=WAVE (GO wavecal). These exposures will use the same calibration lamp configurations and exposure times as the automatic wavecals. The only way to tell the difference between GO and automatic wavecal data is to look at the MEMTYPE header keyword, which will be discussed later in Table 2.6 of the “Association Tables (ASN)” Section.

1.3.3 Typical COS Observing Sequence

For most observations, the following sequence of events occurs:

- Acquire the object using ACQ/IMAGE with the NUV detector. This may be preceded by an ACQ/SEARCH if needed to scan a larger area of sky. If the target is bright enough, the ACQ/PEAKXD, ACQ/PEAKD sequence can be used.
- Obtain a spectrum in TIME-TAG mode using TAGFLASH mode so that the data can be corrected for any OSM drifts, and with different FPPOS positions to enhance the signal-to-noise.
- Obtain more spectra during additional orbits as needed to achieve a desired signal-to-noise.

The typical COS observing sequence depends greatly on the type of observation specified. Typical COS observations use TIME-TAG mode and the PSA, with simultaneous wavelength calibrations taken via TAGFLASH. Multiple exposures are often used to cover the FUV detector gap, or to produce full wavelength coverage from the NUV wavelength stripes.

1.4 COS Coordinate System

References to multiple coordinate systems appear in the headers of COS data. These are tied to the OTA frame, the User frame, and the POS-TARG frame. The following is a brief explanation of how these systems (shown in [Figure 1.10](#)) are related, and a more thorough explanation can be found in the [Phase II Instructions](#).

The three coordinate systems of interest are the:

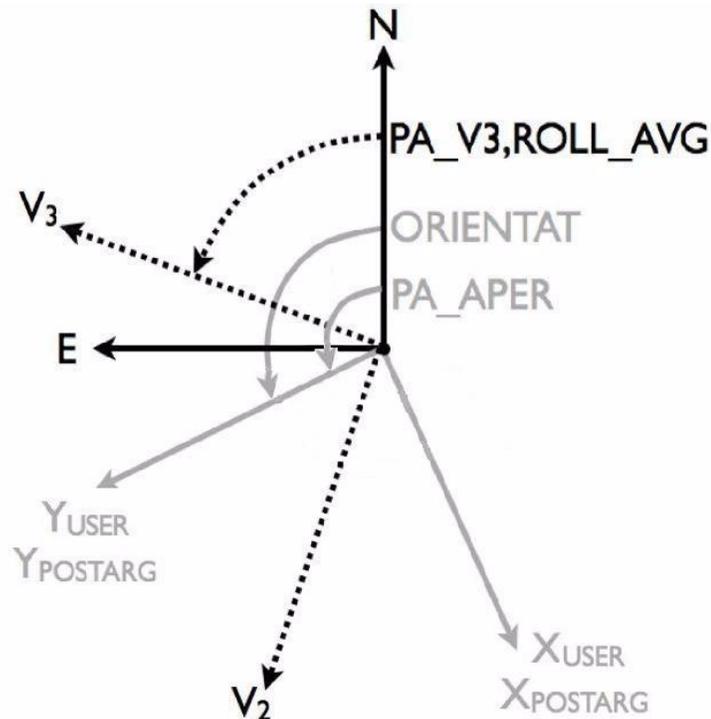
- OTA or “V” Frame (V_1, V_2, V_3): The common coordinate system for Scientific Instruments and the FGSs. It is a distortion-free frame whose metric is arc seconds.
- User (or **IRAF**) Frame ($X_{\text{user}}, Y_{\text{user}}$): The frame associated with a pipeline science image. It is aligned with the detector.
- POS-TARG Frame ($X_{\text{POSTARG}}, Y_{\text{POSTARG}}$): This is a distortion-free frame with units of arc seconds. Its origin coincides with the science aperture and its axes are closely aligned with the user frame.

The angles associated with these frames that appear in the headers of COS data files are:

- PA_V3: The position angle of the V_3 axis; the angle from North, towards East, to V_3 , measured at the center of the *HST* focal plane (in the `spt` header).
- ROLL_AVG: The average angle from North towards East to V_3 , measured at the position of the COS field in the *HST* focal plane (in the `jit` header, computed).
- PA_APER: The angle from North through East to Y_{POSTARG} measured at the aperture reference (in the science header).
- ORIENTAT: The angle from North through East to Y_{USER} measured at the aperture reference (in science header). For COS, PA_APER and ORIENTAT are equal, i.e., $Y_{\text{POSTARG}} = Y_{\text{USER}}$. Note that this is not the same angle as the ORIENT specified in Phase II, which gives the position angle of the $U3$ axis, where $U3 = -V3$.

Refer to [ISR TEL2008-02](#) for a complete discussion of the COS reference frame geometry.

Figure 1.10: COS Coordinate Systems



COS Data Files

In this chapter...

2.1 Overview / 19
2.2 COS File Names / 21
2.3 COS File Structures / 24
2.4 COS Data Products / 25
2.5 Data Storage Requirements / 48
2.6 Headers, Keywords, and Relationship to Phase II / 50
2.7 Error and Data Quality Array / 62

2.1 Overview

Raw COS telescope data are processed through the STScI **OPUS** pipeline. The **OPUS** pipeline first processes the data through Generic Conversion, where the data bits from individual exposures are unpacked and combined into files containing raw, uncalibrated data. Next, the data are processed through the COS calibration pipeline, **calcos**, which performs image and spectroscopic reduction to produce output files that can be used directly for scientific analysis (see [Chapter 3](#) for a more detailed description of the COS calibration pipeline). Finally, the data are ingested into the Hubble Data Archive (HDA) through the Data Archive and Distribution System (DADS). This system populates a database containing header keywords which is accessible to users via the Multimission Archive at STScI (MAST). The data (both calibrated and uncalibrated) are then available for distribution by MAST to the user.

When COS data are requested from the HDA, they go through “On The Fly Reprocessing” (OTFR) which provides the best calibrated products by reprocessing

the raw telemetry files “on-the-fly” each time data are requested. OTFR reprocessing uses the latest software versions and reference files available. The re-processed data are then distributed to the requestor.

The calibration reference files (e.g. flat fields, bad pixel tables) are also available from the *HST* Data Archive for users to download. Since reference files are frequently updated, OTFR may use different reference files depending on the date of reprocessing. In the event of an updated reference file or calibration software, users may re-calibrate their data in one of two ways. Once the updated reference files are released, the preferred method is for the user to re-retrieve the data from the HDA and let OTFR recalibrate the data with the default settings. Alternatively, the user can reprocess the data at home through **calcos** using the most recent reference files and software code (see “Run Calcos” in [Section 3.6.1](#)). *The second option will not include any changes in the data due to Generic Conversion updates, but will allow a customized calibration through the use of modified reference files or keyword switches.* Also, the user will need to manually edit the header keywords stating which reference files should be used by **calcos** ([Table 2.18](#), and [Section 3.6.1](#)).

Once you have retrieved your data, you will need to understand:

- The naming conventions and file suffixes of the individual files ([Section 2.2](#)).
- The basic format in which the COS data are stored ([Section 2.3](#)).
- The structure and content of the individual files ([Section 2.4](#)).
- The size of the COS data files ([Section 2.5](#)).
- How to use the header keywords to identify the principal parameters of an observation and how to determine the calibration processing steps that were performed on a dataset ([Section 2.6](#)).
- The meanings of the error and data quality arrays, which are propagated through the pipeline for each COS science observation ([Section 2.7](#)).

2.2 COS File Names

The naming convention for COS files is `rootname_*.fits`, where `rootname` follows the `ippsoot` naming convention (see Chapter 5 of the *Introduction to HST Data Handbooks*), and `*` is a three to nine character file suffix. The suffix identifies the type of data within the file. All FUV data files with the exception of the lampflash, `x1d` and `x1dsum` files will have an additional suffix of `_a` or `_b` (e.g. `rootname_*_[a,b].fits`) to denote the detector segment. However, if `segment=A` is specified in the Phase II proposal there will be no corresponding `_b` files and vice versa. The FUV lampflash, `x1d` and `x1dsum` files will always be segment combined and therefore will not have the additional suffix.

Table 2.1 lists the file suffixes for the COS data files and indicates which files are produced by the different types of observations. Depending on the type of observation, and the path it has taken through the calibration pipeline (see calibration flow charts; Figure 3.1-Figure 3.5), there will be an appropriate subset of these files in a given dataset. Note, the format of some of the COS files can be different depending on the observing mode; see Section 2.3 for more details.



COS data utilize a modified naming convention from other HST instruments. In, particular COS FUV files can have TWO suffixes. The first suffix identifies the filetype and the second suffix if present identifies the FUV detector segment. For the remainder of this document the use of “suffix” will refer to the first suffix which identifies the filetype and will always include filetypes with the additional FUV segment suffix if they exist.

Table 2.1: Data Types and File Naming Conventions

Long Suffix	Data Format	Spectroscopic				Imaging		Related Page No.	Contents
		FUV		NUV		NUV			
		TIME_TAG	ACCUM	TIME_TAG	ACCUM	TIME_TAG	ACCUM		
<i>Uncalibrated Science Data</i>									
rawtag	table			•		•		27	Raw NUV TIME-TAG events list
rawtag_a, rawtag_b	table	•						27	Raw FUV TIME-TAG events list
rawaccum	image				•		•	26	Raw NUV ACCUM image
rawaccum_a, rawaccum_b	image		•					26	Raw FUV ACCUM image
rawacq	table or image						•	38	Raw acquisition file
pha_a, pha_b	image		•					28	Pulse height distribution
<i>Uncalibrated Support Data</i>									
asn	table	•	•	•	•	•	•	34	Association file
jit	table	•	•	•	•	•	•	46	Spacecraft pointing data averaged over 3 s intervals
jif	image	•	•	•	•	•	•	48	2-D histogram of the _jit file
spt	image	•	•	•	•	•	•	36	Support, planning and telemetry information
trl	table	•	•	•	•	•	•	35	Trailer file with a historical record of generic conversion processing
<i>Intermediate Data Products</i>									
trl	table	•	•	•	•	•	•	35	The raw trailer file is updated with a historical record and errors log of calibration pipeline processing ¹

Long Suffix	Data Format	Spectroscopic				Imaging		Related Page No.	Contents
		FUV		NUV		NUV			
		TIME-TAG	ACCUM	TIME_TAG	ACCUM	TIME_TAG	ACCUM		
corrtag ²	table			•				28	NUV TIME-TAG events list with calibrated values
corrtag_a ² , corrtag_b ²	table	•						28	FUV TIME-TAG events list with calibrated values
flt	image			•	•	•	•	32	NUV flat-fielded science image
flt_a, flt_b	image	•	•					32	FUV flat-fielded science image
counts	image			•	•	•	•	31	NUV not flat-fielded science image
counts_a, counts_b	image	•	•					31	FUV not flat-fielded science image
lampflash	table	• ³		• ³				30	1-D extracted TAGFLASH (FLASH=yes) spectra
x1d	table	•	•	•	•			32	1-D extracted spectra for a single exposure
x1dsum<n> ⁴	table	•	•	•	•			32	Averaged 1-D extracted spectra for multiple exposures with the same grating, central wavelength, aperture and FPPOS=<n>
<i>Final Data Products</i>									
fltsum	image					•	•	33	Summed flat-fielded image (imaging only). <i>Final calibrated association product for all COS imaging datasets</i>
x1dsum	table	•	•	•	•			32	Final combined 1-D extracted spectra for multiple exposures with the same grating, central wavelength and aperture combining all FPPOS. <i>Final calibrated association product for all COS spectroscopic datasets.</i>

1. Only updated during processing and ingestion by the HDA. When reprocessing data in a user's home environment the `tr1` file will not be updated. Instead reprocessing will generate an ASCII `tra` file.
2. For ACCUM data the time stamps in the first extension are set to the median value in the `corrtag` files; each count in the `rawaccum` file becomes an event. See section 2.4.2.
3. Only for TIME-TAG with FLASH=yes (TAGFLASH mode)
4. <n> can be 1,2,3,4 and denotes the FPPOS number.

2.3 COS File Structures

All COS data products are Multi-Extension FITS (MEF) format files and begin with a primary data unit which includes only a header with no data extension. The **catfits** task in **STSDAS** can be used to list the complete set of extensions and their data formats for the COS data files. For more information on working with MEF format files please refer to Chapter 2 of the *Introduction to HST Data Handbooks*.

2.3.1 COS FITS Table Extension Files

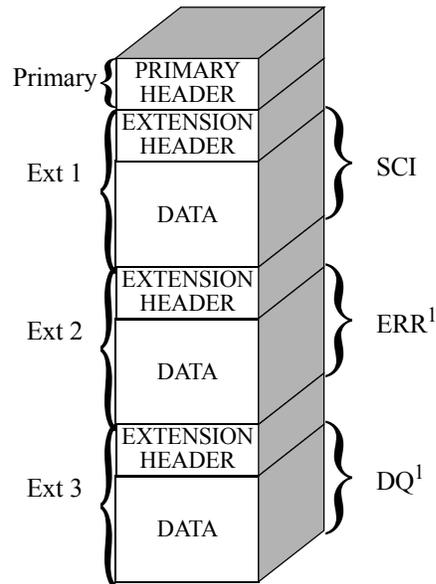
Tabular COS information, such as extracted one-dimensional spectra or the TIME-TAG mode event series, are stored as FITS binary tables. The tables can be accessed directly in the **PyRAF/IRAF/STSDAS** environment using tasks in the **tables.ttools** package as described in Chapters 2 and 3 of the *Introduction to HST Data Handbooks* of this document, or with other standard FITS tools.

2.3.2 COS FITS Image Extension Files

COS images and two-dimensional spectroscopic data are stored in FITS image extension files, which can be directly manipulated, without conversion, in the **PyRAF/IRAF/STSDAS** environment. Accessing images in the FITS image extension files in **IRAF** follows a simple convention explained in detail in Chapter 2 of the *Introduction to HST Data Handbooks*. [Figure 2.1](#) illustrates the structure of a COS FITS image extension file, which contains:

- A primary header that stores keyword information describing the global properties of the exposure in the file (e.g., the target name, target coordinates, exposure type, optical element, aperture, detector, calibration switches, reference files used).
- A set of image extensions, each containing header keywords with information specific to the given exposure (e.g., exposure time, world coordinate system) and a data array.

Figure 2.1: FITS Image Extension File for COS



1. Not all COS image extension files will contain the ERR and DQ extensions.

The following file types are stored in FITS image extension files with the particular format shown in [Figure 2.1](#): `rawaccum`, `flt`, `counts`, `pha` and `rawacq`¹. Each COS readout can generate one FITS image SCI extension or three FITS image extensions (SCI, ERR, and DQ) as explained below:

- The first extension type, SCI, stores the science values.
- The second extension type, ERR, contains the statistical errors, which are propagated through the calibration process. It is unpopulated in raw data files.
- The third extension type, DQ, stores the data quality values, which flag suspect pixels in the corresponding SCI data.

The error arrays and data quality values are described in more detail in [Section 2.7](#). The value of the `XTENSION` keyword in the extension header identifies the type of data the extension contains; the value of this keyword may be determined using the IRAF `tables` tasks `catfits` or `thedit`.

2.4 COS Data Products

The following sections discuss the COS raw science data files, intermediate calibration products, final calibration products, and auxiliary data files. Uncalibrated

¹ Only ACQ/IMAGE files use the exact format shown in [Figure 2.1](#). For more details on acquisition file formats see “Acquisition Files (RAWACQ)” in [Section 2.4.4](#).

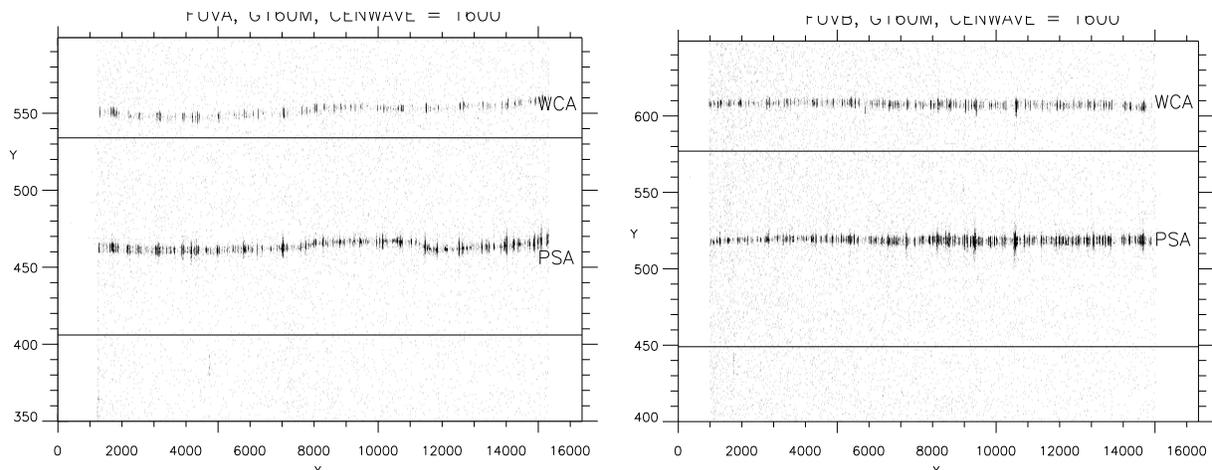
science data include all raw science data generated during Generic Conversion that have not been processed through the calibration pipeline. These raw files are the input files to the **calcos** pipeline, usually as part of an association (see “Association Tables (ASN)”). The result of the pipeline is both individual calibrated exposure files and, when appropriate, a final combined product file.

2.4.1 Uncalibrated Science Data Files

Raw ACCUM Images (rawaccum)

For ACCUM data, the raw files contain a set of images, as shown in [Figure 2.1](#), and have filenames with the suffix `rawaccum` for NUV data, or `rawaccum_a` and `rawaccum_b` for the two segments of the FUV detector. The `SCI` extension contains an image of the total accumulated counts during an exposure. For NUV data the `ERR` and `DQ` extensions have only a header with no data. For FUV data the `ERR` extension has only a header with no data, and the `DQ` extension is populated with data quality information only for pixels that are outside the subarray boundaries (defined below). The `DQ` extensions will be populated in the `flt` files, after calibration pipeline processing. Even though FUV `rawaccum_a [b]` data are 16384 x 1024 images, only portions of them contain actual data. These portions are called subarrays. Typically, three subarrays are used for each segment of an FUV ACCUM image. Two are centered on the STIM positions and the third is a stripe 128 pixels wide which is centered on the wavecal spectrum of the object. [Figure 2.2](#) shows these spectral region subarrays superimposed on two FUV `rawtag` images. As [Figure 2.2](#) shows, the spectrum falls outside of the subarray. Consequently, wavecals must be taken separately for ACCUM data.

Figure 2.2: Overlay of FUV ACCUM Subarrays on FUV TIME-TAG Data



The above figures show FUV TAGFLASH data for both segments with the corresponding ACCUM subarrays noted by the dark lines. The data plotted here are the raw event locations prior to calibration processing. The distortion in the data, particularly for segment A, is very noticeable and discussed further in [Section 3.4.8](#).

Raw TIME-TAG Events Lists (rawtag)

Raw events tables contain the locations and arrival times of individual photon events collected in TIME-TAG mode. These files have the suffix `rawtag` for NUV or `rawtag_a[b]` for the two FUV segments. Figure 2.3 shows the format of a `rawtag` table. The first extension contains the events list, in which each row of the table corresponds to a single event in the data stream and the columns of the table contain scalar quantities that describe the event. The second extension contains the good time intervals (GTI) table, where an uninterrupted period of time is considered as one good time interval. Interruptions in the data taking due to memory overflow could result in more than one GTI. Table 2.2 shows the columns of a `rawtag` table.

Figure 2.3: FITS File Format for Raw and corrected TIME-TAG Tables

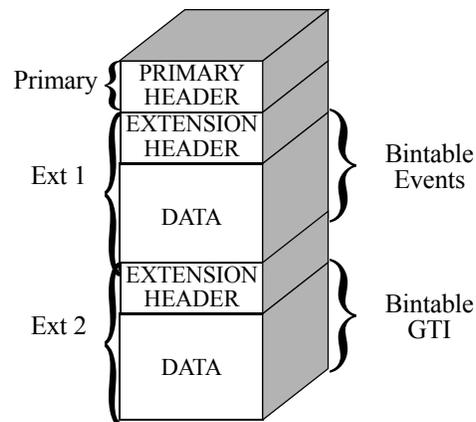


Table 2.2: Columns of a Raw TIME-TAG Data Table

Extension 1			
Column Name	Units	Data Type	Description
TIME	sec	float	Elapsed time in seconds since the exposure start time
RAWX	pixel	integer	Pixel coordinate along the dispersion axis
RAWY	pixel	integer	Pixel coordinate along the cross-dispersion axis
PHA ¹		byte	Pulse height amplitude (0-31)
Extension 2			
Column Name	Units	Data Type	Description
START	sec	float	Start good time interval since exposure start
STOP	sec	float	End good time interval

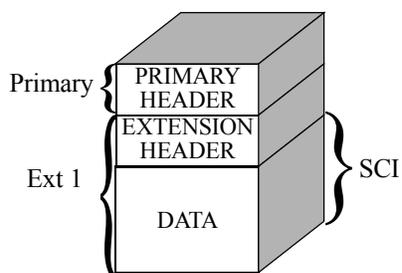
1. The PHA column is present in the NUV data *only* for symmetry with the FUV data columns. For NUV data the values in this column are set to 0, since no pulse height amplitudes are available.

For more information on working with TIME-TAG data see [Section 5.4](#)

Pulse Height Amplitude Files (pha)

For FUV ACCUM data only, a 7 bit pulse height amplitude histogram is accumulated in the detector electronics onboard. This information is placed in a file with the suffix *pha*. The pulse-height histogram files contain a primary header with no data and a single FITS image SCI extension containing a histogram of the pulse-height distribution during the exposure. The pulse height amplitude files do not contain an ERR or DQ extension, as shown in [Figure 2.4](#). The pulse height distribution is an image array of length 128, corresponding to the number of photons with values from 0 to 127, corresponding to the pulse heights of 0-31 available in TIME-TAG data.

Figure 2.4: FITS Array Extension File for COS



2.4.2 Intermediate Science Data Files

Corrected Events Lists (corrtag)

The COS pipeline produces corrected TIME-TAG events lists and stores them in binary tables with suffix *corrtag*. These files have a main header and three extensions: a corrected events list extension, a good time interval extension, and a time line table extension, with a format similar to the one shown in [Figure 2.3](#). The first extension of the *corrtag* file is the events table (see [Table 2.3](#)) which includes X and Y event locations that have been corrected for distortion, doppler shift, and offsets due to OSM motions in both the dispersion and cross-dispersion directions. It also includes wavelengths associated with events that occur within the active area of the detectors and a data quality (DQ) flag for each event (see, [Table 2.19](#)). The second extension gives the start and stop times of the good time intervals (as in the *rawtag* file), and the third extension is the time line table. The time line table includes second by second values for spacecraft position, solar and target altitude above the horizon, and count rates for the most prominent airglow lines and the background. These observed rates might include counts from other external sources in addition to the ones from the airglow line itself. The data in this extension can be useful for reprocessing TIME-TAG data to exclude, for example, daytime data using the Python tool *timefilter*, described in [Chapter 5](#), which is also available as an IRAF task.

For ACCUM data, the *corrtag* files are somewhat different. All of the time stamps in the first extension are set to the median value of the observation. Each count in the *rawaccum* file becomes an event so, for example, a pixel in the *rawaccum* that had 100 counts, would have 100 entries in the *corrtag* file. The RAWX, XCORR and XDOPP entries are all the same for NUV data, but can be different for FUV. In

addition, RAWY and YCORR entries will have the same values. However, XFULL and YFULL can be different. In the timeline extension, the SHIFT1, airglow and DARKRATE entries are fixed, but all others are time dependent.

Table 2.3: Columns of a COS `corrtag` Table

Extension 1			
Column Name	Units	Data Type	Description
TIME	sec	float	Elapsed time in seconds since the exposure start time
RAWX	pixel	integer	Pixel coordinate along dispersion axis (same as in rawtag file)
RAWY	pixel	integer	Pixel coordinate along cross-dispersion axis (same as in rawtag file)
XCORR ¹	pixel	float	RAWX corrected for distortion ¹
XDOPP	pixel	float	XCORR corrected for Doppler shift and for FUV only distortion
YCORR ¹	pixel	float	RAWY corrected for distortion ¹
XFULL	pixel	float	XDOPP corrected for offset in the dispersion direction, based on the wavecal spectrum
YFULL	pixel	float	YCORR corrected for offset in the cross-dispersion direction, based on the wavecal spectrum
WAVELENGTH	Angstrom	float	Only events in the active area are assigned wavelengths
EPSILON		float	Event weight based on flat field and deadtime
DQ		integer	Data quality flag
PHA ²		byte	Pulse height amplitude
Extension 2			
START	sec	float	Start good time interval since exposure start
STOP	sec	float	End good time interval
Extension 3			
TIME	sec	float	Time in 1 sec intervals from first entry
LONGITUDE	degrees	float	Earth based longitude
LATITUDE	degrees	float	Earth based latitude
SUN_ALT	degrees	float	Altitude of the sun above the geometric horizon
SUN_ZD	degrees	float	Angle between HST and the Sun, seen from the center of Earth
TARGET_ALT	degrees	float	Altitude of the target above the geometric horizon
RADIAL_VEL	km/s	float	Instantaneous HST radial velocity toward the target
SHIFT1	pixels	float	Instantaneous dispersion direction shift (stripe B for NUV)
LY_ALPHA	counts/s	float	Total counts/sec in a box across the aperture at Ly alpha
OI_1304	counts/s	float	Total counts/sec in a box across the aperture at OI 1304
OI_1356	counts/s	float	Total counts/sec in a box across the aperture at OI 1356
DARKRATE	counts/s	float	Counts/sec/pixel averaged over both dark regions

1. The XCORR and YCORR columns are present in the NUV data only for symmetry with FUV data. Currently no distortion correction is applied to NUV data, so for NUV data the XCORR and YCORR columns are identical to the RAWX and RAWY columns.
2. The PHA column is present in the NUV data *only* for symmetry with the FUV data columns. For NUV data this column is set to a default value of 0, since no pulse height amplitudes are available for NUV.

Lampflash Files (lampflash)

For TAGFLASH data, calcos produces an events list with suffix lampflash, that contains the extracted wavecal lamp flashes. Each row in the events list corresponds to a different segment or stripe and flash number (the first flash is number 1, the second is number 2, etc.). The lampflash files have the format shown in Figure 2.5. The contents of the columns in a lampflash events list are listed in Table 2.4. Columns TIME, LAMP_ON, and LAMP_OFF have the same temporal zero point as the TIME column of the rawtag and corrtag tables and the same unit (seconds). The shifts contained in the SHIFT_DISP and SHIFT_XDISP columns of the lampflash table are applied to the XDOPP and YCORR columns of the corrtag file to produce the X[Y]FULL entries. When multiple TAGFLASHES are present, the shifts are interpolated in time for events occurring between the first and last flashes. Events occurring before the first flash are shifted by a value extrapolated using the slope defined by the first two flashes; events beyond the last flash are given the shift determined by the last flash. As a result, the difference between the X[Y]FULL and X[Y]CORR entries in the corrtag file can be a function of time.

Figure 2.5: FITS File Format for Lampflash Table

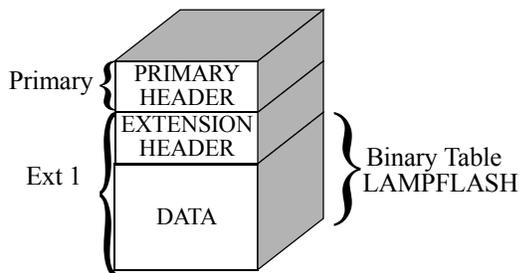


Table 2.4: Columns of a COS Lampflash Table

Column Name	Units	Data Type	Description
SEGMENT		String	FUV segment(s) or NUV stripe(s) corresponding to the extracted tagflash wavecal
TIME	sec	float	Median time of each flash
EXPTIME	sec	float	Duration of each flash in seconds
LAMP_ON	sec	float	Lamp turn on time for each flash, counting from start of exposure
LAMP_OFF	sec	float	Lamp turn off time for each flash, counting from start of exposure
NELEM		integer	Length of the WAVELENGTH, GROSS, NET, BACKGROUND, DQ, DQ_WGT, and ERROR arrays
WAVELENGTH	Å	double[nelem]	Wavelengths of each extracted tagflash wavecal spectrum(s)
GROSS	counts s ⁻¹	float[nelem]	Gross count rate of each extracted tagflash wavecal spectrum(s)
NET	counts s ⁻¹	float[nelem]	Net count rate of each extracted tagflash wavecal spectrum(s)
BACKGROUND	counts s ⁻¹	float[nelem]	Background count rate of each extracted tagflash wavecal spectrum(s)
SHIFT_DISP	pixel	float[nelem]	Dispersion direction shift(s) determined by comparing each tagflash wavecal with a wavecal template
SHIFT_XDISP	pixel	float[nelem]	Cross-dispersion direction shift(s) determined by comparing each tagflash wavecal with a wavecal template
CHI_SQUARE		float	Chi square of comparison between tagflash wavecal and wavecal template
N_DEG_FREEDOM		integer	Number of degrees of freedom in chi square comparison
SPEC_FOUND		string[nelem]	T (true) or F (false), if each tagflash wavecal spectrum was found or not

Counts Files (counts)

The counts images are an intermediate calibrated output product for both imaging and spectroscopic data with suffix `counts`. These files contain three extensions (SCI, ERR, and DQ) as shown in Figure 2.1. These files are constructed by summing up the events from each pixel using the XFULL and YFULL coordinates. The data are in units of counts per pixel. For FUV data the images are 16384 columns in the x (dispersion) direction by 1024 rows in the y (cross-dispersion) direction. The NUV images are 1274 columns in the x direction by 1024 rows in the cross-dispersion direction for spectroscopic data, and 1024 x 1024 for data obtained in imaging mode. The NUV spectroscopic files have more pixels in the dispersion direction than the actual NUV detector. This is necessary because Doppler shifts and shifts due to OSM

motions can cause wavelengths at one time during an exposure to fall outside of those obtained at another time during the exposure. As a result, the format has to be expanded to accommodate the shifts that occur during an exposure. The FUV images are not extended since the active area is less than the size of the detector, so these effects can be incorporated into the images without the need to extend them. The FUV data are also corrected for Y-walk and geometric distortions.

Flat-Fielded Image Files (f1t)

For *spectroscopic* data a flat-fielded image is an intermediate calibrated data file. These files have a suffix `f1t`, and contain three extensions (SCI, ERR, and DQ) as shown in [Figure 2.1](#). These files are constructed by summing up the values in the EPSILON column for each pixel using the XFULL and YFULL coordinates. The data are in units of the count rate. For FUV data the images are 16384 x 1024, and, like the `counts` images, the NUV images are 1274 x 1024 for spectroscopic data and 1024 x 1024 for data obtained in imaging mode. The `f1t` images are corrected for deadtime effects. The NUV images are corrected for all flat-field effects and the FUV data are currently only corrected for the XDL grid wire shadows.

2.4.3 Final Science Data Files (and Product Files)

The initial input files to **calcos** are the association tables with suffix `asn`. These files provide the calibration pipeline with information about how the data files are associated. In general, only exposures taken in sequence with the same spectral element, central wavelength (if applicable), and aperture at any FPPOS will be associated. However, the user can create associations that include data taken with different central wavelengths, and **calcos** version 2.18.5 and later will calibrate such associations. For more information on COS association files see the “Association Tables (ASN)” portion of [Section 2.4.4](#).

Processing of each individual exposure in the association produces a final calibrated result named with exposure rootname and suffix `x1d` (spectroscopy) or `f1t` (imaging).

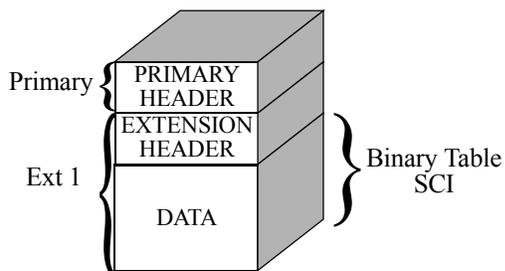
Next, for each FPPOS position `<n>` (where `<n>`=1,2,3, or 4), if there are *multiple spectroscopic* exposures in the association that use the same FPPOS position, **calcos** will combine them into a file named with the association rootname and suffix `x1dsum<n>`, where `<n>` is the integer FPPOS value.

Lastly, a final association product file is produced with association rootname and suffix `x1dsum` (spectroscopy) or `f1tsum` (imaging) by combining *all science exposures* in the association.

One-Dimensional Extracted Spectra (x1d, x1dsum)

The COS pipeline produces extracted one-dimensional spectra and stores them in binary tables with suffix `x1d`, `x1dsum<n>`, or `x1dsum`. [Figure 2.6](#) shows the format of the 1-D extracted spectra table.

Figure 2.6: FITS File Format for 1-D Extracted Spectrum Table



These COS extracted spectra tables can be 1 to 3-Dimensional, with one row for each unique segment or stripe. For FUV data there are two rows which correspond to segments A and B distinguished by “FUVA” and “FUVB” in the SEGMENT column respectively. For NUV data there are three rows, “NUVA”, “NUVB” and “NUVC” corresponding to stripes A, B, and C respectively. Each table column can contain either a scalar value or an array of values, such as WAVELENGTH or FLUX. For example, NELEM will contain a scalar number, while the WAVELENGTH column will contain an array of wavelengths. Table 2.5 shows the contents of the different columns in an extracted spectrum table. A discussion of the data in COS extracted spectra is provided in Section 3.4.15.

Table 2.5: Columns of a COS Extracted Spectrum Table

Column Name	Units	Data Type	Description
SEGMENT		string	FUV segments or NUV stripe names
EXPTIME	sec	float	Corrected exposure times for each segment
NELEM		integer	Length of the array fields, such as the WAVELENGTH and GROSS arrays
WAVELENGTH	Å	double[nelem]	Wavelengths corresponding to fluxes
FLUX	$\text{erg s}^{-1} \text{cm}^{-2} \text{Å}^{-1}$	float[nlam]	Flux calibrated NET spectrum
ERROR	$\text{erg s}^{-1} \text{cm}^{-2} \text{Å}^{-1}$	float[nlam]	Internal error estimate
GROSS	counts s^{-1}	float[nlam]	Gross extracted spectrum count rate
NET	counts s^{-1}	float[nelem]	Difference of GROSS and BACKGROUND arrays
BACKGROUND	counts s^{-1}	float[nelem]	Background count rate
GCOUNTS	counts	float[nlam]	Gross counts
DQ_WGT		float[nelem]	Data quality weight
DQ		integer[nelem]	Logical OR of data quality flags in extraction region

Flat-Fielded Image Files (f1t, f1tsum)

For NUV imaging observations, the f1t and f1tsum images are the final data products, with the latter being a simple sum of the individuals when several exposures are processed together. They are fully linearized and flat-field corrected images. Unlike the f1t files produced for the spectroscopic data (which are intermediate data

products with a format of 1274 x 1024, see [Section 2.4.2](#)), the formats of the `flt` and `fltsum` files for imaging data are 1024 x 1024, since Doppler and OSM motions are not applied.

2.4.4 Auxiliary Data Files

Association Tables (ASN)

An association file is created for all COS observation sets, and has the suffix `asn` (e.g., `19v221010_asn.fits`). This file holds a single binary table extension, which can be displayed with the IRAF tasks `tprint` or `tread`.

`Calcos` calibrates raw data from multiple science exposures and any contemporaneously obtained line lamp calibration exposures through the pipeline as an associated unit. Each individual science exposure in an *association* is fully calibrated in the process. See Chapter 5 of *Introduction to HST Data Handbooks* for a general explanation of *HST* data associations. The information within an association table shows how a set of exposures are related, and informs the COS calibration pipeline how to process the data.

An example association table is shown in [Figure 2.7](#). Note that all related COS exposures will be listed in an association table, with the exception of acquisitions, darks, and flats. It is possible to have an association which contains only one exposure. The association file lists the rootnames of the associated exposures as well as their membership role in the association. The exposures listed in an association table directly correspond to individual raw FITS files. For example, the association table can describe how wavecal exposures are linked to science exposures. [Table 2.6](#) summarizes the different exposure membership types (`MEMTYPES`) used for COS association tables.

Table 2.6: Member Types in COS Associations.

<code>MEMTYPE</code>	Description
<code>EXP-AWAVE</code>	Input automatic wavelength calibration exposure
<code>EXP-FP</code>	Input science exposure
<code>EXP-GWAVE</code>	Input GO wavelength calibration exposure
<code>PROD-FP</code>	Output science product

[Figure 2.7](#) illustrates the contents of the association table for a sequence of spectroscopic exposures for four FPPOS positions.

Figure 2.7: Sample Association Table l9v221010_asn

To display the association table for l9v221010_asn.fits:

```
cl> tprint l9v221010_asn.fits
# row MEMNAME MEMTYPE MEMPRSNT
1 L9V221EUQ EXP-FP yes
2 L9V221EWQ EXP-AWAVE yes
3 L9V221EYQ EXP-FP yes
4 L9V221F0Q EXP-AWAVE yes
5 L9V221F2Q EXP-FP yes
6 L9V221F4Q EXP-AWAVE yes
7 L9V221F6Q EXP-FP yes
8 L9V221F8Q EXP-AWAVE yes
9 L9V221010 PROD-FP no
```

In the above figure, MEMTYPE describes the exposure membership type or role in the association. The column MEMPRSNT lists whether the member is present or not. A user could choose to change the association file to not include a member during processing by changing the MEMPRSNT to 'no'.

The association table above lists the names of the eight associated exposures (four external and four calibration) that are calibrated and combined to create the various association products which will have a rootname of l9v221010. This particular association is created from a single TIME-TAG spectroscopic APT specification with FPPOS=ALL and FLASH=NO specified in the Phase II file, which leads to both a science exposure and automatic wavecal exposure taken at each FPPOS location. For example, the first entry in the table, l9v221euq, is the rootname of a single external science exposure taken with FPPOS=1. This exposure corresponds to the following rawtag files: l9v221euq_rawtag_a.fits, l9v221euq_rawtag_b.fits. The memtype of this exposure is EXP-FP which shows that it is an external exposure. The second entry in the table has a memtype of EXP-AWAVE. This denotes that the corresponding rawtag exposures, l9v221ewq_rawtag_a.fits and l9v221ewq_rawtag_b.fits, are wavecal exposures that will be used by the pipeline for wavelength calibration. Similar files correspond to the remaining three pairs of entries in the association file for data taken with the remaining three FPPOS positions. The pipeline will calibrate the members of an association as a unit, producing the calibrated data products for each individual exposure as well as the final combined association data product. For this particular association, the pipeline will produce a final combined association product, l9v221010_x1dsum.fits, which contains the final FPPOS combined, calibrated spectrum.

Trailer Files (TRL)

When COS data are processed through OTFR in the HDA, the output messages from generic conversion and the different calibration steps are stored in a FITS

ASCII table known as the trailer file, with suffix `tr1`. Each time the archive processes data before retrieval, the old trailer file is erased and a new one created using the results of the most recent processing performed. The archive will produce a trailer file for each individual exposure and association product. Association product trailer files contain the appended information from all the exposures in the association, in order of processing. The order of processing is the same as the order of exposures in the association table, with the exception of AUTO or GO wavecalls which are always processed first.

In the trailer files from the HDA, the output messages from generic conversion appear first in the file. This section contains information relevant to the selection of the best reference files and the population of some of the header keywords. The second part of this file contains information from **calcos** processing. Each task in the **calcos** pipeline creates messages during processing which describe the progress of the calibration, and appear in the order in which each step was performed. These messages are quite relevant to understanding how the data were calibrated, and in some of the cases, to determining the accuracy of the products.



It is highly recommended to always examine the trailer files.

In this last section of the `_tr1` file, the **calcos** steps are indicated by their module name. The **calcos** messages provide information on the input and output files for each step, the corrections performed, information regarding the reference files used, and in the case of FUV data, messages about the location of the stims, or shift correction applied to the data. **Calcos** also gives warnings when the appropriate correction to the data could not be applied. For more detailed information on the calibration steps and structure of **calcos**, please refer to [Chapter 3](#).

Calcos Trailer Files (TRA)

When **calcos** is run in a user's home environment, **calcos** redirects the output of its steps to the STDOUT and an ASCII file with name `rootname.tra`. Note, the level of detail included in the output messages can be modified when running **calcos** (see "Run Calcos" on page 104). So, when run on a personal machine, **calcos** will *not* overwrite the `tr1` file but rather will direct the output to STDOUT and an ASCII `tra` file. The `tra` file is formatted like the `tr1` file but with two exceptions: the `tra` file will not contain the output messages from generic conversion, and the `tra` file is not converted to FITS format. Therefore, one must look at both the trailer (generic conversion messages) and `tra` (calibration messages) file when running **calcos** from a home environment. Each time **calcos** is run on a file, the STDOUT messages will be appended to the `tra` file if it already exists. Also, when running **calcos** on a personal machine there will be no `tra` created for the association products. Instead, the **calcos** messages for association products will be sent only to STDOUT.

Support Files (SPT)

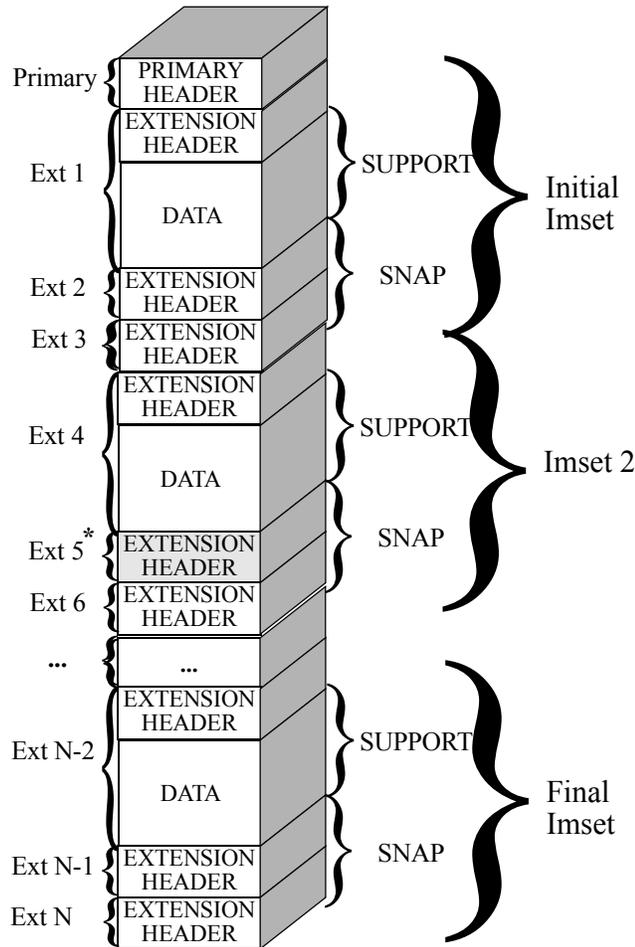
The support files contain information about the observation and engineering data from the instrument and spacecraft that were recorded at the time of the observation. A

COS support file contains a primary header and at least three FITS image extensions. The first extension contains a header with the proposal information and an (16-bit) image array containing the data which populate the `spt` header keyword values. The image array element values are used by conversion software to populate the header keywords. Following the support extension, the COS `spt` files contain two engineering snapshot extensions. These extensions contain a readout of several instrument and telescope parameters from telemetry data at different times during the course of an exposure. The very first snapshot extension will always contain telemetry information from the beginning of an exposure. Depending on the length of the exposure, the support file may also contain one or several “imsets” which include a support extension and two snap extensions. These intermediate imsets will have only their second snapshot extension populated with telemetry data taken during the course of an exposure, while the first snapshot will be populated with default values. The very last imset of an `spt` file will have all three extensions (1 support and 2 snaps) populated with telemetry values at the completion of the exposure. [Figure 2.8](#) depicts the structure of an N extension COS support file.

With several snapshots of COS telemetry values, one may track the instrument status periodically throughout an exposure. For a schematic listing of the `spt` headers with detailed information about the `spt` header keywords, See:

<http://stdatu.stsci.edu/keyword/cgi-bin/kdct-header.cgi?i=COS&s=20.1&db=Operational>

Figure 2.8: COS Support File



* Extension 5, is not populated, and therefore all header keywords in this extension will be set to a default. Every other snapshot extension from extension 5 through N-4, will also not be populated.

COS support file with N extensions. The initial imset contains telemetry values at the start of the exposure. Extensions 3 through (N-3) contain imsets with telemetry values at intermediate times during the exposure. Note that the first snap extensions in these intermediate imsets are NOT populated. The final imset includes extensions N-2 through N and contains telemetry values at the end of the exposure. Both snap extensions are populated for the final imset.

Acquisition Files (RAWACQ)

All COS acquisition exposures will produce a single raw data file with suffix rawacq. Almost all COS spectroscopic science exposures are preceded by an acquisition sequence or exposure to center the target in the aperture. Keywords in the header of COS science data identify the exposure names of relevant acquisition exposures in each visit. In addition, there are several other useful keywords in the COS acquisition exposures that describe the acquisition parameters used, as well as the calculated centroid positions and slew offsets. Table 2.7 lists all the relevant acquisition keywords.

Table 2.7: ACQ/IMAGE Header Keywords.

Keyword Name	Description
ACQSNAME ¹	Rootname of first acquisition search exposure
ACQINAME ¹	Rootname of first acquisition image exposure
PEAKXNAM ¹	Rootname of first cross-dispersion peakup exposure
PEAKDNAM ¹	Rootname of first dispersion peakup exposure
ACQ_NUM ¹	Total number of exposures in acquisition sequence
LAMPSTAT	Status of Wavecal lamp exposure (LTAIMCAL)
LAMPTIME	Lamp exposure integration time (s)
LAMPXCR	Measured lamp x centroid (pixels)
LAMPYCR	Measured lamp y centroid (pixels)
LAMPEVNT	Number of events in the lamp exposure
LAMPCNTR	Lamp Centering method
LMPSUBX1	X co-ordinate of the top left of the lamp subarray (pixels)
LMPSUBX2	X co-ordinate of the bottom right of the lamp subarray (pixels)
LMPSUBY1	Y co-ordinate of the top left of the lamp subarray (pixels)
LMPSUBY2	Y co-ordinate of the bottom right of the lamp subarray (pixels)
ACQSTAT	Status of the acquisition exposure (LTAIMAGE)
TARGETIME	Acquisition exposure integration time (s)
ACQCENTX	Measured target centroid in X direction (pixels)
ACQCENTY	Measured target centroid in Y direction (pixels)
WCA2SCIX	WCA to science Aperture X offset (pixels)
WCA2SCIY	WCA to Science aperture Y offset (pixels)
ACQPREFX	Nominal target X centroid (pixels)
ACQPREFY	Computed center of the target in the Y direction (pixels)
ACQSLEWX	Commanded slew in the X direction (arcseconds)
ACQSLEWY	Commanded slew in the Y direction (arcseconds)
TRGSUBX1	X co-ordinate of the top left of the target subarray (pixels)
TRGSUBX2	X co-ordinate of the bottom right of the target subarray (pixels)
TRGSUBY1	Y co-ordinate of the top left of the target subarray (pixels)
TRGSUBY2	Y co-ordinate of the bottom right of the target subarray (pixels)

1. These keywords are also found in the COS science headers in addition to being in the acquisition headers.

Table 2.8: ACQ/SEARCH Header Keywords.

Keyword Name	Description
ACQSNAME ¹	Rootname of first acquisition search exposure
ACQINAME ¹	Rootname of first acquisition image exposure
PEAKXNAM ¹	Rootname of first cross-dispersion peakup exposure
PEAKDNAM ¹	Rootname of first dispersion peakup exposure
ACQ_NUM ¹	Total number of exposures in acquisition sequence
ACQSTAT	Status of target exposure
TARGETIME	Integration time per dwell (s)
CENTER	Centering method used by the search
ACQFLOOR	Threshold Floor used (for FLUX-WEIGHT-FLOOR centering method)
SCANSIZE	Number of dwells per side of the square pattern
ACQNPOS	Total number of dwells
STEPSSIZE	Scan step size between dwells (arcsec)
ENDSLEWX	Commanded X-direction slew from the final dwell point (arcsec)
ENDSLEWY	Commanded Y-direction slew from the final dwell point (arcsec)
ACQSLEWX	Commanded X slew from the centre of the search pattern (arcsec)
ACQSLEWY	Commanded Y slew from the centre of the search pattern (arcsec)
SEGMENT ²	Segment used
TRGSUBX1 ³	X co-ordinate of the bottom right of the target subarray (pixels)
TRGSUBX2 ³	X co-ordinate of the bottom right of the target subarray (pixels)
TRGSUBY1 ³	Y co-ordinate of the top left of the target subarray (pixels)
TRGSUBY2 ³	Y co-ordinate of the bottom right of the target subarray (pixels)
TRGAS1X1 ²	X co-ordinate of the top left of the first A segment target subarray (pixels)
TRGAS1X2 ²	X co-ordinate of the bottom right of the first A segment target subarray (pixels)
TRGAS1Y1 ²	Y co-ordinate of the top left of the first A segment target subarray (pixels)
TRGAS1Y2 ²	Y co-ordinate of the bottom right of the first A segment target subarray (pixels)
TRGBS1X1 ²	X co-ordinate of the top left of the first B segment target subarray (pixels)
TRGBS1X2 ²	X co-ordinate of the bottom right of the first B segment target subarray (pixels)
TRGBS1Y1 ²	Y co-ordinate of the top left of the first B segment target subarray (pixels)
TRGBS1Y2 ²	Y co-ordinate of the bottom right of the first B segment target subarray (pixels)
TRGAS2X1 ²	X co-ordinate of the top left of the second A segment target subarray (pixels)
TRGAS2X2 ²	X co-ordinate of the bottom right of the second A segment target subarray (pixels)
TRGAS2Y1 ²	Y co-ordinate of the top left of the second A segment target subarray (pixels)
TRGAS2Y2 ²	Y co-ordinate of the bottom right of the second A segment target subarray (pixels)
TRGBS2X1 ²	X co-ordinate of the top left of the second B segment target subarray (pixels)
TRGBS2X2 ²	X co-ordinate of the bottom right of the second B segment target subarray (pixels)
TRGBS2Y1 ²	Y co-ordinate of the top left of the second B segment target subarray (pixels)
TRGBS2Y2 ²	Y co-ordinate of the bottom right of the second B segment target subarray (pixels)

1. These keywords are also found in the COS science headers in addition to being in the acquisition headers.

2. FUV only

3. NUV only

Table 2.9: ACQ/PEAKXD Header Keywords.

Keyword Name	Description
ACQSNAME ¹	Rootname of first acquisition search exposure
ACQINAME ¹	Rootname of first acquisition image exposure
PEAKXNAM ¹	Rootname of first cross-dispersion peakup exposure
PEAKDNAM ¹	Rootname of first dispersion peakup exposure
ACQ_NUM ¹	Total number of exposures in acquisition sequence
LAMPSTAT	Status of lamp exposure (LTACAL)
LAMPTIME	Integration time of lamp exposure (s)
LAMPYCR	Measured lamp Y centroid (pixels)
LAMPEVNT	Number of events in lamp exposure
LAMPENTR	Lamp Centering Method
STRIPE ²	Stripe used
LMPSUBX1 ²	X co-ordinate of the top left of the lamp subarray (pixels)
LMPSUBX2 ²	X co-ordinate of the bottom right of the lamp subarray (pixels)
LMPSUBY1 ²	Y co-ordinate of the top left of the lamp subarray (pixels)
LMPSUBY2 ²	Y co-ordinate of the bottom right of the lamp subarray (pixels)
LMPASBX1 ³	X co-ordinate of the top left of the A segment lamp subarray (pixels)
LMPASBX2 ³	X co-ordinate of the bottom right of the A segment lamp subarray (pixels)
LMPASBY1 ³	Y co-ordinate of the top left of the A segment lamp subarray (pixels)
LMPASBY2 ³	Y co-ordinate of the bottom right of the A segment lamp subarray (pixels)
LMPBSBX1 ³	X co-ordinate of the top left of the B segment lamp subarray (pixels)
LMPBSBX2 ³	X co-ordinate of the bottom right of the B segment lamp subarray (pixels)
LMPBSBY1 ³	Y co-ordinate of the top left of the B segment lamp subarray (pixels)
LMPBSBY2 ³	Y co-ordinate of the bottom right of the B segment lamp subarray (pixels)
ACQSTAT	Status of target exposure (LTAPKXD)
TARGETIME	Acquisition exposure integration time (s)
ACQMEASY	Measured target centroid in the Y direction (pixels)
ACQPREFY	Nominal target Y centroid (pixels)
ACQSLEWY	Commanded slew in the Y direction (arcsec)
TARGEVNT	Number of events in the acquisition exposure
STRIPE ²	Stripe used
SEGMENT ³	Segment used
FBASLOPE ³	Slope used for Segment B->A Mapping
FBAINTER ³	Intercept used for Segment B->A Mapping
FBASCALE ³	Scaling factor used for Segment B -> A Mapping

Table 2.9: ACQ/PEAKXD Header Keywords. (Cont'd)

Keyword Name	Description
TRGSUBX1 ²	X co-ordinate of the bottom right of the target subarray (pixels)
TRGSUBX2 ²	X co-ordinate of the bottom right of the target subarray (pixels)
TRGSUBY1 ²	Y co-ordinate of the top left of the target subarray (pixels)
TRGSUBY2 ²	Y co-ordinate of the bottom right of the target subarray (pixels)
TRGAS1X1 ³	X co-ordinate of the top left of the first A segment target subarray (pixels)
TRGAS1X2 ³	X co-ordinate of the bottom right of the first A segment target subarray (pixels)
TRGAS1Y1 ³	Y co-ordinate of the top left of the first A segment target subarray (pixels)
TRGAS1Y2 ³	Y co-ordinate of the bottom right of the first A segment target subarray (pixels)
TRGBS1X1 ³	X co-ordinate of the top left of the first B segment target subarray (pixels)
TRGBS1X2 ³	X co-ordinate of the bottom right of the first B segment target subarray (pixels)
TRGBS1Y1 ³	Y co-ordinate of the top left of the first B segment target subarray (pixels)
TRGBS1Y2 ³	Y co-ordinate of the bottom right of the first B segment target subarray (pixels)
TRGAS2X1 ³	X co-ordinate of the top left of the second A segment target subarray (pixels)
TRGAS2X2 ³	X co-ordinate of the bottom right of the second A segment target subarray (pixels)
TRGAS2Y1 ³	Y co-ordinate of the top left of the second A segment target subarray (pixels)
TRGAS2Y2 ³	Y co-ordinate of the bottom right of the second A segment target subarray (pixels)
TRGBS2X1 ³	X co-ordinate of the top left of the second B segment target subarray (pixels)
TRGBS2X2 ³	X co-ordinate of the bottom right of the second B segment target subarray (pixels)
TRGBS2Y1 ³	Y co-ordinate of the top left of the second B segment target subarray (pixels)
TRGBS2Y2 ³	Y co-ordinate of the bottom right of the second B segment target subarray (pixels)

1. These keywords are also found in the COS science headers in addition to being in the acquisition headers.

2. NUV only

3. FUV only

Table 2.10: ACQ/PEAKD Header Keywords.

Keyword Name	Description
ACQSNAME ¹	Rootname of first acquisition search exposure
ACQINAME ¹	Rootname of first acquisition image exposure
PEAKXNAM ¹	Rootname of first cross-dispersion pickup exposure
PEAKDNAM ¹	Rootname of first dispersion pickup exposure
ACQ_NUM ¹	Total number of exposures in acquisition sequence
ACQSTAT	Status of acquisition (LTAPKD)
TARGETIME	Acquisition exposure integration time (s)
CENTER	Centering method used
ACQFLOOR	Threshold floor value
ACQNPOS	Number of dwells in the acquisition
STEPSIZE	Distance between dwells (arcsec)
ACQMEASX	Measured target X position (pixels)
ACQPREFIX	Nominal target X position (pixels)
ENDSLEWX	Commanded X-direction slew from final dwell point (arcsec)
ACQSLEWX	Commanded X-direction slew from the nominal ACQ centre (arcsec)
SEGMENT ²	Segment used
TRGSUBX1 ³	X co-ordinate of the bottom right of the target subarray (pixels)
TRGSUBX2 ³	X co-ordinate of the bottom right of the target subarray (pixels)
TRGSUBY1 ³	Y co-ordinate of the top left of the target subarray (pixels)
TRGSUBY2 ³	Y co-ordinate of the bottom right of the target subarray (pixels)
TRGAS1X1 ²	X co-ordinate of the top left of the first A segment target subarray (pixels)
TRGAS1X2 ²	X co-ordinate of the bottom right of the first A segment target subarray (pixels)
TRGAS1Y1 ²	Y co-ordinate of the top left of the first A segment target subarray (pixels)
TRGAS1Y2 ²	Y co-ordinate of the bottom right of the first A segment target subarray (pixels)
TRGBS1X1 ²	X co-ordinate of the top left of the first B segment target subarray (pixels)
TRGBS1X2 ²	X co-ordinate of the bottom right of the first B segment target subarray (pixels)
TRGBS1Y1 ²	Y co-ordinate of the top left of the first B segment target subarray (pixels)
TRGBS1Y2 ²	Y co-ordinate of the bottom right of the first B segment target subarray (pixels)

Table 2.10: ACQ/PEAKD Header Keywords. (Cont'd)

Keyword Name	Description
TRGAS2X1 ²	X co-ordinate of the top left of the second A segment target subarray (pixels)
TRGAS2X2 ²	X co-ordinate of the bottom right of the second A segment target subarray (pixels)
TRGAS2Y1 ²	Y co-ordinate of the top left of the second A segment target subarray (pixels)
TRGAS2Y2 ²	Y co-ordinate of the bottom right of the second A segment target subarray (pixels)
TRGBS2X1 ²	X co-ordinate of the top left of the second B segment target subarray (pixels)
TRGBS2X2 ²	X co-ordinate of the bottom right of the second B segment target subarray (pixels)
TRGBS2Y1 ²	Y co-ordinate of the top left of the second B segment target subarray (pixels)
TRGBS2Y2 ²	Y co-ordinate of the bottom right of the second B segment target subarray (pixels)

1. These keywords are also found in the COS science headers in addition to being in the acquisition headers.

2. FUV only

3. NUV only

PEAKD and SEARCH Acquisitions:

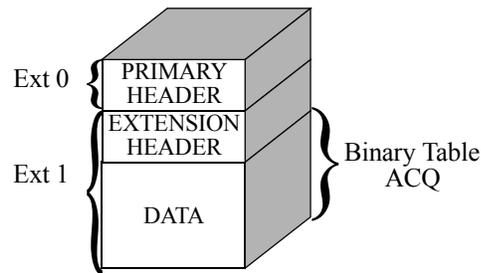
Acquisition peakups in the dispersion direction (ACQ/PEAKD) and acquisition spiral searches (ACQ/SEARCH) both use the flux from exposures taken at different dwell points to center the target. For more information on these types of COS acquisitions see Sections 7.6.4 and 7.6.2 respectively of the *COS Instrument Handbook*. Data for these acquisitions contain one binary table extension which describes the acquisition search pattern dwell point locations and counts as shown in Table 2.11 and Figure 2.9

Table 2.11: Columns of an ACQ/SEARCH or ACQ/PEAKD Table

Column Name	Units	Description
DWELL_POINT		Dwell point number in search pattern
DISP_OFFSET ¹	arcsec	Offset in dispersion direction from the initial target pointing
XDISP_OFFSET	arcsec	Offset in the cross-dispersion direction from the initial target pointing
COUNTS	counts	Raw counts value at dwell point

1. This column is only present in ACQ/SEARCH tables

Figure 2.9: FITS File Format for ACQ/SEARCH and ACQ/PEAKD Data.

***PEAKXD Acquisition:***

Acquisition peakups in the cross-dispersion direction (ACQ/PEAKXD) use a TIME-TAG spectrum to center the target in the cross-dispersion direction. For more information on the ACQ/PEAKXD algorithm see Section 7.6.3 of the *COS Instrument Handbook*. An ACQ/PEAKXD exposure includes only a primary header and extension header. There are no data downlinked for this type of acquisition.

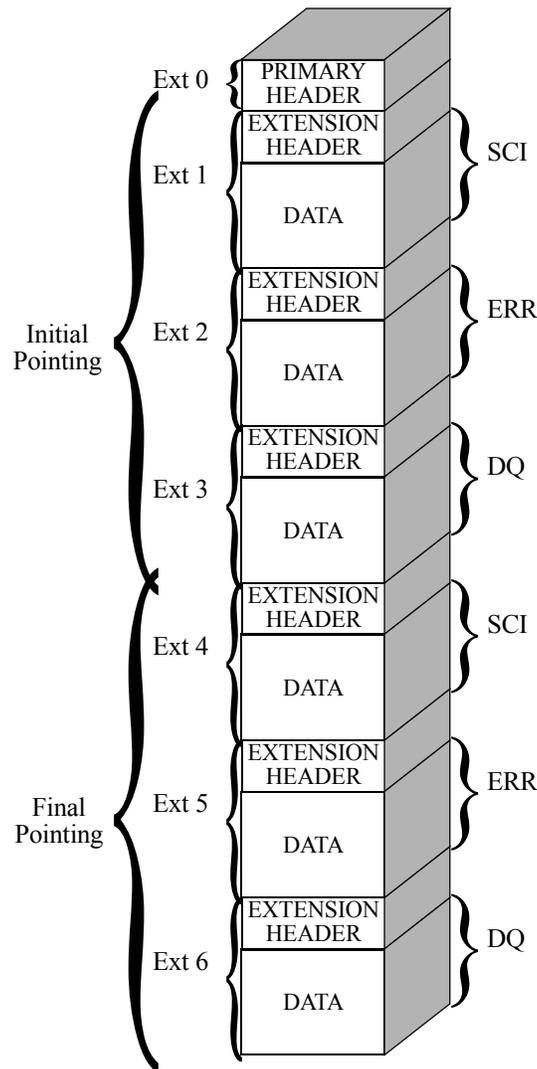
IMAGE Acquisition:

Acquisition images (ACQ/IMAGE) use a NUV image to center the target in the aperture. For more information on the ACQ/IMAGE algorithm see Section 7.5 of the *COS Instrument Handbook*. An ACQ/IMAGE exposure produces a raw data file containing two science image extensions corresponding to the initial and final pointing:

- [SCI,1] is an image of the initial target pointing.
- [SCI,2] is a confirmation image after the acquisition procedure has been performed.

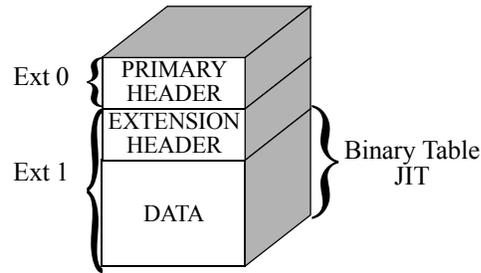
See [Figure 2.10](#) for the FITS file format for ACQ/IMAGE data.

Figure 2.10: FITS File Format for ACQ/IMAGE Data



Jitter Files (jit)

The COS jitter files include engineering data that describe the performance of the Pointing Control System (PCS) including the Fine Guidance Sensors that are used to control the vehicle pointing. The jitter files report on PCS engineering data during the duration of the observation. The support files contain information about the observation and engineering data from the instrument and spacecraft that were recorded at the time of the observation. COS jitter files utilize the file format shown in [Figure 2.11](#) for all science observations, excluding acquisitions.

Figure 2.11: FITS File Format for JITTER Data.

The jitter tables contain PCS data for each three second interval during the observation, as listed in [Table](#) . For more information on jitter files refer to Chapter 6 of *Introduction to HST Data Handbooks*.

Table 2.12: Columns of a jitter Table.

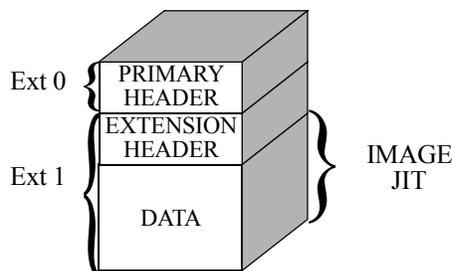
Column Name	Data Type	Units	Description
SECONDS	float	seconds	'Seconds' three second intervals from start
V2_DOM	float	arcsec	Dominant FGS V2 Coordinate
V3_DOM	float	arcsec	Dominant FGS V3 Coordinate
V2_ROLL	float	arcsec	Roll FGS V2 Coordinate
V3_ROLL	float	arcsec	Roll FGS V3 Coordinate
SI_V2_AVG	float	arcsec	Mean jitter in V2 over 3 seconds
SI_V2_RMS	float	arcsec	Peak jitter in V2 over 3 seconds
SI_V2_P2P	float	arcsec	RMS jitter in V2 over 3 seconds
SI_V3_AVG	float	arcsec	Mean jitter in V3 over 3 seconds
SI_V3_RMS	float	arcsec	Peak jitter in V3 over 3 seconds
SI_V3_P2P	float	arcsec	RMS jitter in V3 over 3 seconds
RA	double	degrees	Right Ascension of aperture reference
DEC	double	degrees	Declination of aperture reference
ROLL	doublet	degree	Position angle between North and +V3
LIMBANG	float	degree	Position angle between V1 axis and Earth limb
TERMANG	float	degree	Angle between V1 axis and terminator
LOS_ZENITH	float	degree	Angle between HST Zenith and target
LATITUDE	float	degree	HST subpoint latitude
LONGITUDE	float	degree	HST subpoint longitude
MAG_V1	float	Gauss	Magnetic field along V1
MAG_V2	float	Gauss	Magnetic field along V2
MAG_V3	float	Gauss	Magnetic field along V3

Table 2.12: Columns of a jitter Table. (Cont'd)

Column Name	Data Type	Units	Description
BRIGHTLIMB	integer		Earth limb of LimbAng is bright (1 or 0) t
FGS_FLAGS	float		FGS status flags
DAYNIGHT	string		Observation taken during the day (0) or night (1)
RECENTER	string		Recentering status flag, event in progress =1
TAKEDATA	string		Vehicle guiding status, nominal GS tracking =1
SLEWFLAG	string		Vehicle slewing status, slewing =1

2-D Spacecraft Pointing Histogram (jif)

The COS jif files are a 2-D histogram of the corresponding jit file (See “Jitter Files (jit)”) and have the file format shown in [Figure 2.12](#) for all science observations excluding acquisitions.

Figure 2.12: FITS File Format for jif Data.

2.5 Data Storage Requirements

Users are reminded to consider the large size of `counts` and `flt` files when allocating disk space for storing and reprocessing COS data. Additionally, `corrtag` files with a large number of events can be quite large. These images serve as intermediate or final calibration products from the pipeline and have the file sizes given in Megabytes in [Table 2.13](#). *Note, that these sizes are per exposure, and an associated observation set may have several exposures.*

Table 2.13: COS Pipeline Data Volumes per Exposure

File Type	FUVA	FUVB	Total FUV	Total NUV	Calibrated File
rawtag	9 bytes per photon	9 bytes per photon	9 bytes per photon (18MB per BUFFER-TIME)	8 bytes per photon (16 MB per BUFFER-TIME)	
corrtag	35 bytes per photon	35 bytes per photon	35 bytes per photon (70 MB per BUFFER-TIME)	26 bytes per photon (52 MB per BUFFER-TIME)	•
rawaccum	64MB	64MB	128MB	2MB	
flt	160MB	160MB	320MB	10MB	•
x1d	0.5MB ¹	0.5MB ¹	1MB ²	<1MB	•
fltsum	N/A	N/A	N/A	10MB	•
counts	160MB	160MB	320MB	10MB	•
x1dsum	0.5MB	0.5MB	1MB ²	<1MB	•
lampflash	N/A	N/A	<1MB	<1MB	•

1. Values pertain to x1d_a or x1d_b files only. These files are temporary output products from **calcos** processing

2. Values are in addition to amounts given for each segment.

Similarly, users are reminded of the large cumulative size of calibrated COS spectroscopic datasets. Table 2.14 provides volume estimates for calibrated COS datasets.

Table 2.14: COS Pipeline Data Volumes per Calibrated Exposure

Detector	FUV		NUV	
Observation Mode	TIME-TAG	ACCUM	TIME_TAG	ACCUM
Pipeline-processed volume per exposure	650 MB + 44 bytes per photon	775 MB + 36 bytes per photon	25-35 MB + 34 bytes per photon	25-35 bytes
Standard calibrated files ¹	325 MB + 35 bytes per photon ²	325 MB	15-25 MB + 36 bytes per photon ³	15-25 MB

1. Minimum volume delivery option over the internet

2. Approximately 70 MB per BUFFER-TIME

3. Approximately 52 MB per BUFFER-TIME

2.6 Headers, Keywords, and Relationship to Phase II

As with previous *HST* instruments, the FITS header keywords in COS data files store important information characterizing the observations and telemetry received during the observations, and describe the post-observation processing of your dataset. Each keyword follows FITS conventions and is no longer than eight characters. Values of keywords can be integer, real (floating-point), boolean, and character strings. Several keywords are *HST* and COS specific. Knowledge of the keywords and where to find them is an important first step in understanding your data. By examining your file headers, using either **catfits**, **imhead**, **hselect**, **thselect** or **hedit**, in **STSDAS** you will find detailed information about your data including:

- Target name, coordinates, proposal ID, and other proposal level information.
- Observation and exposure time information such as observation start and duration.
- Instrument configuration information such as detector, grating, central wavelength setting, and aperture.
- Readout definition parameters such as subarray parameters.
- Exposure-specific information such as more detailed timing, world coordinate system information, and Fine Guidance Sensor identification.
- Calibration information such as the calibration switches and reference files used by the pipeline and parameters derived from the calibration, such as image statistics and wavelength shifts.

The keywords relevant for one COS data type will not necessarily be relevant to another. Accordingly, you will find that the header on a particular file type contains a unique combination of keywords appropriate for that type of observation. Long definitions for the keywords can also be accessed from the following Web page, which provides detailed explanations of the contents and algorithm for populating the keywords. This site also provides sample headers for different COS file types:

<http://stdatu.stsci.edu/keyword/>.

Keywords that deal with a particular topic, such as the instrument configuration, are grouped together logically throughout the headers. [Table 2.15](#) lists a useful subset of these groups of keywords, indicates the name of the grouping, and where applicable, shows their relationship to the corresponding information from the Phase II proposal.

[Table 2.16](#) summarizes the possible calibration switch keywords, and indicates whether they are present for a particular observation; it also indicates the reference file keyword corresponding to the particular calibration step. A calibration switch keyword is initially populated with values of OMIT, PERFORM or N/A in the raw uncalibrated science data. After each calibration step is executed in the COS calibration pipeline, **calcos** will set the keyword switch to COMPLETE.

Table 2.15: Selected Header Keywords and Relationship to Phase II Parameters

Header Keyword	Phase II Equivalent	Description
General File Information (Primary Header)		
FILENAME		Name of the file
FILETYPE		Type of data found in the file (SCI, ACQ, SPT, ASN_TABLE)
NEXTEND		Number of extensions in the file.
DATE		Date file was created
Program Information (Primary Header)		
PROPOSID		4 or 5 digit program number.
PR_INV_L	PI Last Name	Last name of principal investigator
PR_INV_F	PI First Name	First name of principal investigator
PR_INV_M	PI Middle Initial	Middle name initial of principal investigator
LINENUM	Visit_Number, Exposure_Number	Indicates the visit and exposure number from the Phase II proposal: Visit_Number, Exposure_Number.
Target Information (Primary Header)		
TARGNAME	TargetName	Name of target.
RA_TARG	RA	Right ascension of the target (deg) (J2000).
DEC_TARG	DEC	Declination of the target (deg) (J2000).
POSTARG1	POSTARG	Postarg in axis 1 direction.
POSTARG2	POSTARG	Postarg in axis 2 direction.

Table 2.15: Selected Header Keywords and Relationship to Phase II Parameters (Cont'd)

Header Keyword	Phase II Equivalent	Description
Science Instrument Configuration (Primary Header)		
OBSTYPE		Observation type (IMAGING or SPECTROSCOPIC).
OBSMODE	Opmode	Operating mode (ACCUM, TIME-TAG).
EXPTYPE	Opmode	Exposure type (EXTERNAL/SCI, WAVECAL, PHA, DARK, FLAT, ACQ/IMAGE, ACQ/SEARCH, ACQ/PEAKD, ACQ/PEAKXD, ENG DIAG, or MEM DUMP).
DETECTOR	Config	Detector in use (NUV or FUV).
SEGMENT	SEGMENT	FUV detector segment in use (FUVA, FUVB, BOTH, or N/A).
DETECTHV		FUV detector high voltage state (NomAB, NomA, NomB, Off, Low).
SUBARRAY		Data from a subarray (T) or full frame (F).
LAMPUSED		Lamp status, NONE or name of lamp which is on (P1, D1, P2, or D2)
LAMPSET		Spectral calibration lamp current value (milliamps).
LIFE_ADJ		Detector Life time adjustment position. (1-5)
OPT_ELEM	SpElement	Optical element in use (grating or mirror name).
CENWAVE	Wavelength	Central wavelength for grating settings.
APERTURE	Aperture	Aperture name.
PROPAPER	Aperture	Proposed aperture name.
APER_FOV		Aperture field of view description in mm.
FPPOS	FPPOS	Grating offset index (1-4) for spectrum dithers (FPPOS).
TAGFLASH	FLASH	Type of flashed exposures in TIME-TAG (NONE, AUTO, or UNIFORMLY-SPACED).
EXTENDED	Extended	Is the target extended (Yes or No).
NRPTEXP	NumberOfIterations	Number of repeat exposures in dataset: DEFAULT = 1.
EXP_NUM		Exposure number for repeated observations.
SHUTTER		External shutter position (OPEN or CLOSED).
Engineering Parameters (Primary Header)		
FPOFFSET		FP offset from nominal, in motor steps.
DEVENTA		Digital event counter, FUV segment A (counts s ⁻¹).
DEVENTB		Digital event counter, FUV segment B (counts s ⁻¹).
FEVENTA		Fast event counter, FUV segment A (counts s ⁻¹).
FEVENTB		Fast event counter, FUV segment B (counts s ⁻¹).
MEVENTS		NUV MAMA event counter (counts s ⁻¹).

Table 2.15: Selected Header Keywords and Relationship to Phase II Parameters (Cont'd)

Header Keyword	Phase II Equivalent	Description
Target Acquisition Dataset Identifiers (Primary Header)		
ACQSNAME		Rootname of first acquisition search exposure.
ACQINAME		Rootname of first acquisition image exposure.
PEAKXNAM		Rootname of first x-dispersion pickup exposure.
PEAKDNAM		Rootname of first dispersion pickup exposure.
ACQ_NUM		Total number of exposures in acquisition sequence.
Archive Search Keywords (Primary Header)		
BANDWID		Bandwidth of the data.
SPECRES		Approximate resolving power at central wavelength.
CENTRWV		Central wavelength of the data.
MINWAVE		Minimum wavelength in spectrum.
MAXWAVE		Maximum Wavelength in spectrum.
PLATESC		Plate scale (arcsec/pixel).
Association Keywords (Primary Header)		
ASN_ID		Unique identifier assigned to association.
ASN_MTYT		Role of the member in the association.
ASN_TAB		Name of the association table.
Exposure Information (in Extension header 1 or greater)		
DATE-OBS		UT date of start of observation (yyyy-mm-dd).
TIME-OBS		UT time of start of observation (hh:mm:ss).
EXPTIME		Corrected exposure time (seconds). For FUV exposures, the largest of EXPTIMEA and EXPTIMEB
EXPTIMEA		Corrected FUV Segment A exposure time (seconds).
EXPTIMEB		Corrected FUV Segment B exposure time (seconds).
RAWTIME		Exposure time of an individual raw exposure (seconds).
EXPSTART		Exposure start time (Modified Julian Date).
EXPEND		Exposure end time (Modified Julian Date).
EXPSTRTJ		Exposure start time (Julian Date).
EXPENDJ		Exposure end time (Julian Date).
PLANTIME	TimePerExposure	Planned exposure time (seconds).

Table 2.15: Selected Header Keywords and Relationship to Phase II Parameters (Cont'd)

Header Keyword	Phase II Equivalent	Description
NINTERPT		Number of exposure interrupts.
ORIENTAT		Position angle of image y-axis (degrees).
SUNANGLE		Angle between sun and V1 axis.
MOONANGL		Angle between moon and V1 axis.
SUN_ALT		Altitude of the sun above Earth's limb.
FGSLOCK		Commanded FGS lock (Fine, Coarse, Gyros, Unknown).
GYROMODE		Number of gyros scheduled for observation.
REFFRAME		Guide star catalog version.
Aperture Information (in Extension header 1 or greater)		
RA_APER		RA of reference aperture center.
DEC_APER		Declination of reference aperture center.
PA_APER		Position angle of reference aperture center.
SHIFT1A		Wavecal shift determined for FUV segment A or NUV stripe A in dispersion direction (pixels).
SHIFT1B		Wavecal shift determined for FUV segment B or NUV stripe B in dispersion direction (pixels).
SHIFT1C		Wavecal shift determined for NUV stripe C in dispersion direction (pixels).
SHIFT2A		Offset in cross-dispersion direction for FUV segment A or NUV stripe A (pixels).
SHIFT2B		Offset in cross-dispersion direction for FUV segment B or NUV stripe B (pixels).
SHIFT2C		Offset in cross-dispersion direction for NUV stripe C (pixels).
SP_LOC_A		Cross-dispersion location at which calcos extracted the FUV segment A or NUV stripe A spectrum.
SP_LOC_B		Cross-dispersion location at which calcos extracted the FUV segment B or NUV stripe B spectrum.
SP_LOC_C		Cross-dispersion location at which calcos extracted the NUV stripe C spectrum.
SP_NOM_A		Nominal location of the spectral extraction region for FUV segment A or NUV stripe A based on the wavecal aperture location.
SP_NOM_B		Nominal location of the spectral extraction region for FUV segment B or NUV stripe B based on the wavecal aperture location.
SP_NOM_C		Nominal location of the spectral extraction region for NUV stripe C based on the wavecal aperture location.

Table 2.15: Selected Header Keywords and Relationship to Phase II Parameters (Cont'd)

Header Keyword	Phase II Equivalent	Description
SP_OFF_A		Offset from SP_NOM_A at which the spectrum was found.
SP_OFF_B		Offset from SP_NOM_B at which the spectrum was found.
SP_OFF_C		Offset from SP_NOM_C at which the spectrum was found.
SP_SLP_A		Slope of FUV segment A or NUV stripe A spectrum
SP_SLP_B		Slope of FUV segment B or NUV stripe B spectrum
SP_SLP_C		Slope of NUV stripe C spectrum
SP_HGT_A		Height in pixels of the FUV Segment A spectral extraction region.
SP_HGT_B		Height in pixels of the FUV Segment B spectral extraction region.
DPIXEL1A		Fractional part of pixel coordinate for FUV segment A or NUV stripe A (pixels). Average binning error.
DPIXEL1B		Fractional part of pixel coordinate for FUV segment B or NUV stripe B (pixels). Average binning error.
DPIXEL1C		Fractional part of pixel coordinate for NUV stripe C (pixels). Average binning error.
TIME-TAG Parameters (in Extension header 1 or greater)		
BUFFTME	BUFFER-TIME	Onboard memory half-buffer-fill time.
OVERFLOW		Number of science data overflows.
NBADEVNT		Total number of events deleted in screening (NUV).
NBADEVTA		Total number of events deleted in screening (FUV, segment A).
NBADEVTB		Total number of events deleted in screening (FUV, segment A).
NEVENTS		Total number of events in raw data (NUV).
NEVENTSA		Total number of events in raw data (FUV, segment A).
NEVENTSB		Total number of events in raw data (FUV, segment B).
FUV TIME-TAG Parameters (in Extension header 1 or greater)		
TBRST_A		Time lost to bursts on FUV segment A (seconds).
TBRST_B		Time lost to bursts on FUV segment B (seconds).
TBADT_A		Time lost to BADTCORR screening on FUV segment A (sec).
TBADT_B		Time lost to BADTCORR screening on FUV segment B (sec).
NPHA_A		Number of events lost due to pulse height screening on segment A.
NPHA_B		Number of events lost due to pulse height screening on segment B.
NBRST_A		Number of events lost due to burst screening on segment A.
NBRST_B		Number of events lost due to burst screening on segment B.
NBADT_A		Number of events flagged by BADTCORR for segment A.
NBADT_B		Number of events flagged by BADTCORR for segment B.
NOUT_A		Number of events outside the active area for segment A.
NOUT_B		Number of events outside the active area for segment B.

Table 2.15: Selected Header Keywords and Relationship to Phase II Parameters (Cont'd)

Header Keyword	Phase II Equivalent	Description
NUV TIME-TAG Parameters (in Extension header 1 or greater)		
NEADT		Number of events flagged by BADTCORR.
TBADT		Time lost to BADTCORR screening (sec).
TAFGFLASH Parameters (in Extension header 1 or greater)		
NUMFLASH		Integer number of flashes.
LMPDUR _i		Duration of flash <i>i</i> , seconds.
LMP_ON _i		Lamp turn-on time for flash <i>i</i> , seconds since EXPSTART.
LMPOFF _i		Lamp turn-off time for flash <i>i</i> , seconds since EXPSTART.
LMPMED _i		Median time of flash <i>i</i> , seconds since EXPSTART.
Velocity Reference Frame Conversion (in Extension header 1 or greater)		
V_HELIO		Geometric to heliocentric velocity.
V_LSRSTD		Heliocentric to standard solar LSR.
Doppler Correction Parameters (in Extension header 1 or greater)		
ORBITPER		Orbital period used onboard for Doppler correction.
DOPPER		Doppler shift period (seconds).
DOPPMAGV		Doppler shift magnitude (Km/sec).
DOPPON		Doppler correction flag (T or F).
DOPPZERO		Commanded time of zero Doppler shift (MJD).
Instrument Doppler Correction Parameters (in Extension header 1 or greater)		
ORBTPERT		Orbital period used onboard for Doppler correction.
DOPMAGT		Doppler shift magnitude (low-res pixels).
DOPPONT		Doppler correction flag (T or F).
DOPPZEROT		Commanded time of zero Doppler shift (MJD).
Global Count Parameters (in Extension header 1 or greater)		
GLOBRATE		Global count rate (NUV).
GLOBRT_A		Global count rate (FUV, segment A).
GLOBRT_B		Global count rate (FUV, segment B).
GLOBLIM		Was global linearity level exceeded (NUV).
GLOBLIMA		Was global linearity level exceeded for FUV segment A.
GLOBLIMB		Was global linearity level exceeded for FUV segment B.

Table 2.15: Selected Header Keywords and Relationship to Phase II Parameters (Cont'd)

Header Keyword	Phase II Equivalent	Description
Subarray Readout Parameters (in Extension header 1 or greater)		
NSUBARRY ¹		Number of subarrays (1-8)
CORNER [N] X		Subarray N axis1 corner pt in unbinned detector pixels.
CORNER [N] Y		Subarray N axis2 size in unbinned detector pixels.
SIZE [N] Y		Subarray N axis1 corner pt in unbinned detector pixels.
SIZE [N] X		Subarray N axis2 size in unbinned detector pixels.
Stim Pulse Parameters (in Extension header 1 or greater; for FUV data only)		
STIMRATE		Approximate STIM pulse injection rate.
STIMA_LX		Segment A Left STIM pulse X centroid in raw data.
STIMA_LY		Segment A Left STIM pulse Y centroid in raw data.
STIMA_RX		Segment A Right STIM pulse X centroid in raw data.
STIMA_RY		Segment A Right STIM pulse Y centroid in raw data.
STIMB_LX		Segment B Left STIM pulse X centroid in raw data.
STIMB_LY		Segment B Left STIM pulse Y centroid in raw data.
STIMB_RX		Segment B Right STIM pulse X centroid in raw data.
STIMB_RY		Segment B Right STIM pulse Y centroid in raw data.
STIMA0LX		Reference location of Segment A Left STIM pulse X coordinate.
STIMA0LY		Reference location of Segment A Left STIM pulse Y coordinate.
STIMA0RX		Reference location of Segment A Right STIM pulse X coordinate.
STIMA0RY		Reference location of Segment A Right STIM pulse Y coordinate.
STIMB0LX		Reference location of Segment B Left STIM pulse X coordinate.
STIMB0LY		Reference location of Segment B Left STIM pulse Y coordinate.
STIMB0RX		Reference location of Segment B Right STIM pulse X coordinate.
STIMB0RY		Reference location of Segment B Right STIM pulse Y coordinate.
STIMASLX		RMS width of Segment A Left STIM pulse X coordinate.
STIMASLY		RMS width of Segment A Left STIM pulse Y coordinate.
STIMASRX		RMS width of Segment A Right STIM pulse X coordinate.
STIMASRY		RMS width of Segment A Right STIM pulse Y coordinate.
STIMBSLX		RMS width of Segment B Left STIM pulse X coordinate.
STIMBSLY		RMS width of Segment B Left STIM pulse Y coordinate.
STIMBSRX		RMS width of Segment B Right STIM pulse X coordinate.
STIMBSRY		RMS width of Segment B Right STIM pulse Y coordinate.
Pulse Height Parameters (in Extension header 1 or greater for FUV data only)		
PHALOWRA		Pulse height screening lower limit for segment A.
PHALOWRB		Pulse height screening lower limit for segment B.
PHAUPPRA		Pulse height screening upper limit for segment A.
PHAUPPRB		Pulse height screening upper limit for segment B.

Table 2.15: Selected Header Keywords and Relationship to Phase II Parameters (Cont'd)

Header Keyword	Phase II Equivalent	Description
Image Statistics and Data Quality Flags (in Extension header 1 or greater)		
NGOODPIX		Number of good pixels (NUV).
NGOOD_A		Number of good pixels (FUV, segment A).
NGOOD_B		Number of good pixels (FUV, segment B).
SDQFLAGS		Serious data quality flags. (Can be modified as a calcos parameter see Section 3.4.19)
GOODMAX		Maximum value of good pixels (NUV).
GOODMAXA		Maximum value of good pixels (FUV, segment A).
GOODMAXB		Maximum value of good pixels (FUV, segment B).
GOODMEAN		Mean value of good pixels (NUV).
GOODMN_A		Mean value of good pixels (FUV, segment A).
GOODMN_B		Mean value of good pixels (FUV, segment B).
SOFTERRS		Number of soft error pixels (DQF=1).
Deadtime Correction Keywords (in Extension header 1 or greater)		
DEADRT		Count rate used for the NUV dead time correction (cps)
DEADRT_A		Count rate used in the FUV Segment A dead time correction (cps)
DEADRT_B		Count rate used in the FUV Segment B dead time correction (cps)
DEADMT		NUV Deadtime correction method (DATA, DEVENTS, or MEVENTS)
DEADMT_A		FUVA Deadtime correction method (DATA, DEVENTS, or MEVENTS)
DEADMT_B		FUVB Deadtime correction method (DATA, DEVENTS, or MEVENTS)
TIME-TAG Events Orientation Keywords (in Extension header 1 or greater)²		
TCTYP2		Axis type for dimension 1.
TCTYP3		Axis type for dimension 2.
TCRVL2		Sky coordinates of 1st axis.
TCRVL3		Sky coordinate of 2nd axis.
TCDLT2		Axis 1 degrees per pixels.
TCDLT3		Axis 2 degrees per pixels.
TCRPX2		Axis 1 pixel of tangent plane direction.
TCRPX3		Axis 2 pixel of tangent plane direction.
TALEN2		Length of axis 1.
TALEN3		Length of axis 2.
TC2_2		Partial of first axis coordinate with respect to x.
TC2_3		Partial of first axis coordinate with respect to y.
TC3_2		Partial of second axis coordinate with respect to x.
TC3_3		Partial of second axis coordinate with respect to y.
TCUNI2		Units of first coordinate value.
TCUNI3		Units of second coordinate value.

Table 2.15: Selected Header Keywords and Relationship to Phase II Parameters (Cont'd)

Header Keyword	Phase II Equivalent	Description
World Coordinate System and Related Parameters (in Extension header 1 or greater)		
WCSAXES		Number of World Coordinate System axes.
CRPIX1		x-coordinate of reference pixel.
CRPIX2		y-coordinate of reference pixel.
CRVAL1		First axis value at reference pixel.
CRVAL2		Second axis value at reference pixel.
CTYPE1		The coordinate type for the first axis.
CYTYPE2		The coordinate type for the second axis.
CD1_1		Partial of first axis coordinate with respect to x.
CD1_2		Partial of first axis coordinate with respect to y.
CD2_1		Partial of second axis coordinate with respect to x.
CD2_2		Partial of second axis coordinate with respect to y.
CUNIT1		Units of first coordinate value.
CUNIT2		Units of second coordinate value.
LTV1		Offset in X to subsection start.
LTV2		Offset in Y to subsection start.
LTM1_1		Reciprocal of sampling rate in X.
LTM2_2		Reciprocal of sampling rate in Y.

1. For FUV data subarrays 1-3 refer to segment A, and subarrays 4-7 refer to segment B.
2. The values for these keywords are currently deleted from the output files except for NUV Imaging.

Table 2.16: Spectroscopic Calibration Switch Keywords

EXPTYPE	EXTERNAL/SCI EXTERNAL/CAL				WAVECAL		DARK		FLAT	
	FUV		NUV		FUV	NUV	FUV	NUV	FUV	NUV
OBSMODE	TIME-TAG	ACCUM	TIME-TAG	ACCM	TIME-TAG	TIME-TAG	TIME-TAG	TIME-TAG	TIME-TAG	TIME-TAG

Module

BRSTCORR	Omit ¹	N/A	N/A	N/A	Omit ¹	N/A	Omit ¹	N/A	Omit ¹	N/A
BADTCORR	Perform	N/A	Perform	N/A	Perform	Perform	Perform	Perform	Perform	Perform
PHACORR	Perform	Perform	N/A	N/A	Perform	N/A	Perform	N/A	Perform	N/A
RANDCORR	Perform	Perform	N/A	N/A	Perform	N/A	Perform	N/A	Perform	N/A
RANDSEED	-1	-1	N/A	N/A	-1	N/A	-1	N/A	-1	N/A
TEMPCORR	Perform	Perform	N/A	N/A	Perform	N/A	Perform	N/A	Perform	N/A
GEOCORR	Perform	Perform	N/A	N/A	Perform	N/A	Perform	N/A	Perform	N/A
IGEOCORR	Perform	Perform	N/A	N/A	Perform	N/A	Perform	N/A	Perform	N/A
DOPPCORR	Perform	Perform	Perform	Perform	N/A	N/A	N/A	N/A	N/A	N/A
DEADCORR	Perform	Perform	Perform	Perform	Perform	Perform	Perform	Perform	Perform	Perform
FLATCORR	Omit ²	Omit	Perform	Perform	Omit	Perform	Omit	Omit	Omit	Omit
DQICORR	Perform	Perform	Perform	Perform	Perform	Perform	Perform	Perform	Perform	Perform
WAVECORR	Perform	Perform	Perform	Perform	Perform	Perform	Omit	Omit	Omit	Omit
X1DCORR	Perform	Perform	Perform	Perform	Perform	Perform	Omit	Omit	Omit	Omit
BACKCORR	Perform	Perform	Perform	Perform	Perform	Perform	Omit	Omit	Omit	Omit
FLUXCORR	Perform	Perform	Perform	Perform	N/A	N/A	Omit	Omit	Omit	Omit
TDSCORR	Perform	Perform	Perform	Perform	N/A	N/A	Omit	Omit	Omit	Omit
HELCORR	Perform	Perform	Perform	Perform	N/A	N/A	Omit	Omit	Omit	Omit
STATFLAG	T	T	T	T	T	T	T	T	T	T

1. BRSTCORR is set to Omit until further information about FUV bursts has been determined on-orbit.

2. FLA.

Table 2.17: Imaging Calibration Switch Keywords

EXPTYPE	EXTERNAL/SCI EXTERNAL/CAL		WAVECAL	DARK	FLAT	ACQ/IMAGE
DETECTOR	NUV		NUV	NUV	NUV	NUV
OBSMODE	TIME-TAG	ACCUM	TIME-TAG	TIME-TAG	TIME-TAG	ACCUM
Modules						
BADTCORR	Perform	N/A	Perform	Perform	Perform	N/A
FLATCORR	Perform	Perform	Perform	Omit	Omit	Perform
DEADCORR	Perform	Perform	Perform	Perform	Perform	Perform
DQICORR	Perform	Perform	Perform	Perform	Perform	Perform
PHOTCORR	Perform	Perform	Perform	Omit	Omit	Perform
TDSCORR	Perform	Perform	Perform	Omit	Omit	Perform
STAFLAG	T	T	T	T	T	T

Table 2.18: Reference File Keywords

Reference File	Description
HVTAB	High voltage level table
BRSTTAB	Burst parameter table
BRFTAB	Baseline reference frame reference table
BADTTAB	Bad time interval reference table
PHATAB	Pulse height discrimination reference table
GEOFILE	Geometric distortion table
WALKTAB	Y walk correction table
DEADTAB	Deadtime reference file
FLATFILE	Pixel to pixel flat-field reference file
LAMPTAB	Template calibration lamp spectra table
WCPTAB	Wavecal parameters table
DISPTAB	Dispersion coefficient reference table
BPIXTAB	Bad pixel table
GSAGTAB	Gain sag table
XTRACTAB	1-D Spectral extraction information table
FLXTAB	Photometric throughput table
TDSTAB	Time-dependent sensitivity correction table
SPWCSTAB	Spectroscopic World Co-ordinate System table

2.7 Error and Data Quality Array

The COS pipeline propagates both statistical errors and data quality flags throughout the calibration process. These are then combined from both the science data and the reference file data to produce data quality information in the calibrated data.

2.7.1 Error Array

The error array contains an estimate of the statistical error at each pixel. In the raw file, the error array is empty, and in the calibrated files the error array is generated from Poisson statistics of the gross counts and propagation of the errors to account for those introduced by correcting the observed data for flat-field and dead-time effects.

2.7.2 Data Quality Flags

Every photon event in a COS corrtag file has a Data Quality (DQ) flag (Table 2.19). Each flagged condition sets a specific bit in the 16-bit DQ word, thus allowing each event during an exposure to be flagged with multiple conditions using the bitwise logical OR operation. DQ flags can be divided into four types:

1. Spatial flags mark events which fall on a detector region which may be questionable. The BPIXTAB reference file marks the corners of each region on the detector which falls into each of these categories. Separate BPIXTAB files are used for the FUV and NUV detectors. These regions were determined by visual inspection of a set of science data files. For FUV data, the GSAGTAB is applied along with the BPIXTAB. The GSAGTAB is used to flag regions that are severely gain sagged

The DQICORR step of **calcos** maps these spatial regions to the individual photon events, and the **x1dcorr** module uses these flags and the value of SDQFLAGS to create the DQ and DQ_WGT arrays, and ultimately which events to include in the final (**x1dsum**) spectrum (Section 3.4.15). For more information on DQFLAGS and SDQFLAGS consult http://www.stsci.edu/hst/cos/pipeline/cos_dq_flags

The spatial flags include:

- Detector shadows (4) include the locations of the grid wires for the FUV detector, and the vignetted region on the NUV detector.
- Poorly calibrated regions (8) these include areas near the edge of the detector which may be suspect.
- Very low response regions (16) are areas on the detector where the response presents a >80% depression.
- Background feature (32) are regions on the detector where the background count rate has been observed to be higher than the surrounding region and/or unstable.

- The Pixel outside the active area (128) flag marks regions outside of the calibrated region of the detector.
- Low response regions (1024) are areas on the detector where the response presents a >50% depression.
- Low PHA feature (4096) are regions in which unusual features have been identified in long background exposures. These features may have an effect on very low count rate observations.
- Gain-Sag hole (8192) are regions on the FUV detector where the gain is low enough that the calibration may be affected.

2. Temporal flags mark photons that occur during time spans in which the data quality is suspect. Events flagged in this way will be removed from the data products, and the exposure time will be adjusted accordingly. Two types of temporal flags are used:

- FUV event bursts (64), which are flagged by the BRSTCORR module of **calcos**. As of this writing, no bursts have been seen on orbit, so the BRSTCORR step has been set to OMIT by default. If bursts are seen at some point, it is likely that the parameters in the BRSTTAB reference table will have to be adjusted before using BRSTCORR.
- Other Bad Time Intervals (2048) can be defined in the BADTTAB reference file, for time ranges that are known to be problematic. At present, STScI has not defined any bad time intervals, but users running **calcos** on their own may wish to define their own intervals in order to exclude times with high background, etc.

3. Event flags are set by **calcos** if a photon event falls outside defined thresholds. Currently, only the FUV Pulse Height flag (512) falls into this category. All FUV events with pulse heights falling outside the range specified in the PHATAB reference file will have this flag set, and the data will be excluded by the DQICORR module. This flag does not apply to NUV data.

4. Lost Data flags occur if data are missing for some reason, such as errors in transmitting the data from the instrument to the ground. Data marked with these flags is always excluded from the final products. There are two flags in this category:

- Reed-Solomon errors (1)
- Fill Data (256)

Screening for temporal and event flags is done by turning calibration switches on or off, or by altering reference files. Once a photon has been determined to have a bad temporal or event flag, it will never appear in a final data product unless the modules which screen for it are turned off or the reference files which define them are changed. On the other hand, the screening for the spatial flags can be easily altered by changing the SDQFLAGS keyword in the header of the raw data file.

The DQ extension of raw ACCUM files will be filled only when there are missing (data lost) or dubious (software error) data. If no such errors exist, initialization will produce an empty data quality extension whose header has NAXIS=0. These flags are

set and used during the course of calibration, and may likewise be interpreted and used by downstream analysis applications. See [Section 3.4.13](#) for more information on the data quality initialization calibration module.

Table 2.19: COS Data Quality Flags

FLAG Value	Bit Setting	Quality Condition	Type	FUV/NUV
	0000 0000 0000 0000	No anomalies	N/A	Both
1	0000 0000 0000 0001	Reed-Solomon error.	Lost data	Both
2	0000 0000 0000 0010	Not used	N/A	N/A
4	0000 0000 0000 0100	Detector shadow	Spatial	Both
8	0000 0000 0000 1000	Poorly calibrated region (including detector edge)	Spatial	Both
16	0000 0000 0001 0000	Very low response region (>80% depression)	Spatial	Both
32	0000 0000 0010 0000	Background feature	Spatial	FUV
64	0000 0000 0100 0000	Burst	Temporal	FUV
128	0000 0000 1000 0000	Pixel out-of-bounds	Spatial	Both
256	0000 0001 0000 0000	Fill data.	Lost data	Both
512	0000 0010 0000 0000	Pulse Height out of bounds	Event	FUV
1024	0000 0100 0000 0000	Low response region (>50% depression)	Spatial	Both
2048	0000 1000 0000 0000	Bad time interval	Temporal	Both
4096	0001 0000 0000 0000	Low PHA feature	Spatial	Both
8192	0010 0000 0000 0000	Gain-Sag Hole	Spatial	FUV
16384	0100 0000 0000 0000	Not used	N/A	N/A

COS Calibration

In this chapter...

3.1 Raw Data Compilation / 65
3.2 Pipeline Processing Overview / 66
3.3 Calcos: Structure and Data Flow / 67
3.4 Descriptions of Spectroscopic Calibration Steps / 73
3.5 Descriptions of Imaging Calibration Steps / 98
3.6 Customizing COS Data Calibration / 98
3.7 Reference Files / 106

3.1 Raw Data Compilation

The basic input to **calcos** is raw science data files. This section provides a brief overview of how these files are generated from raw spacecraft telemetry.

Telemetry containing COS science data is downlinked from the *HST* satellite through a Tracking and Delay Relay Satellite System (TDRSS) satellite to a ground station in White Sands, NM. From there it is sent to Goddard Space Flight Center where the data capture facility packet processor (PACOR) collects the downlinked science data and places them into telemetry “pod files”. These pod files are then transmitted to STScI where they are saved to a permanent storage medium. The STScI ingest pipeline, OPUS, then unpacks the data, populates keyword values extracted from the telemetry stream, reformats the data, and repackages them into raw, uncalibrated, but scientifically interpretable data files.

The raw files are then processed by the **calcos** software to produce a variety of calibrated data files. The results of these procedures are used to populate the databases that form the searchable archive catalog at STScI describing the individual instrument exposures. At this point, the raw and calibrated data files generated from the pod files are normally discarded. Only the pod files and the information placed in the archive databases are preserved. Each time a user requests data from the Hubble Data Archive via the “On The Fly Reprocessing” (OTFR) system, the raw files are regenerated from

the original pod files, and then recalibrated with the latest reference files. A more detailed description of the OTFR system as it applies to COS and other *HST* instruments can be found in Swade et al, available on-line at

<http://www.adass.org/adass/proceedings/adass00/P2-36/>.

Although the OTFR system uses pod files for input, **calcos** uses the raw files (together with their association files) as its initial input. Figure 1.7 shows a raw image of one segment of the COS FUV XDL detector.

3.2 Pipeline Processing Overview

The calibration pipeline, **calcos**, has been developed by the Space Telescope Science Institute (STScI) to support the calibration of *HST*/COS data. Although the COS pipeline benefits from the design heritage of previous *HST* instruments and of the Far Ultraviolet Spectroscopic Explorer (FUSE), the **calcos** modules are tailored specifically to the COS instrument and based on data reduction algorithms defined by the COS Instrument Definition Team (IDT). As with other *HST* pipelines, **calcos** uses an association table (the `_asn` files) to specify the data files to be included, and employs header keywords to specify the calibration steps to be performed and the reference files to be used. **Calcos** is written in Python, which enables the pipeline and users to take advantage of an extremely productive, open-source, easy-to-read scripting language, with many libraries for data reduction and analysis. **Calcos** is in the **stsci_python** package, which is available for download from STScI:

http://www.stsci.edu/institute/software_hardware/pyraf/current/download

Calcos is designed with a common underlying structure for processing FUV and NUV channels which, respectively, use a cross delay line (XDL) and a Multi Anode Microchannel Array (MAMA) detector. The **calcos** calibration pipeline includes pulse-height filtering and geometric correction for the FUV channel, and flat-field, deadtime, and Doppler correction for both channels. It includes methods for obtaining an accurate wavelength calibration by using the onboard spectral line lamps. A background subtracted spectrum is produced and the instrument sensitivity is applied to create the final flux calibrated spectrum.

There are two basic types of raw data files: TIME-TAG photon lists and ACCUM images of the detector. In general, **calcos** must convert these into one dimensional calibrated flux and wavelength arrays. **Calcos** must be able to perform different types of calibration processes to accommodate the different input types.

The level of calibration performed depends upon the data type.

- Acquisition mode exposures (ACQ/SEARCH, ACQ/PEAKXD, and ACQ/PEAKD) are not calibrated by **calcos**, with the exception of ACQ/IMAGE. Only the raw data from these modes are provided.

- All other science data, including NUV imaging data (ACQ/IMAGE), are completely calibrated. This includes geometric and thermal correction for the FUV data, flat fielding, linearity corrections and pulse height filtering. The spectroscopic data are also flux calibrated and corrected for time dependence in the instrumental sensitivity. The data flow and calibration modules for processing the data are described in detail in sections 3.3 and 3.4.
- Raw data taken in TIME-TAG mode are event lists (rawtag binary tables). The basic calibration is done on the tabular data, producing a calibrated (corrtag) events table. The events are then accumulated into a calibrated image (flt) by **calcos**.
- Raw data taken in ACCUM mode (_rawaccum) are binned to an image array onboard the spacecraft.
- For spectral data, **calcos** extracts a spectrum from the flat-fielded image, computes associated wavelengths, and converts the count rates to flux densities, yielding a one-dimensional, background subtracted spectrum. For FUV data there will normally be two spectra, one from segment A and one from segment B. For NUV data there will normally be three spectra, one for each spectral “stripe”.
- When multiple exposures comprise an observation, these are combined into a single, summed spectrum.

See Chapter 2 for the naming conventions of the various input, temporary, and output calibrated files.

3.3 Calcos: Structure and Data Flow

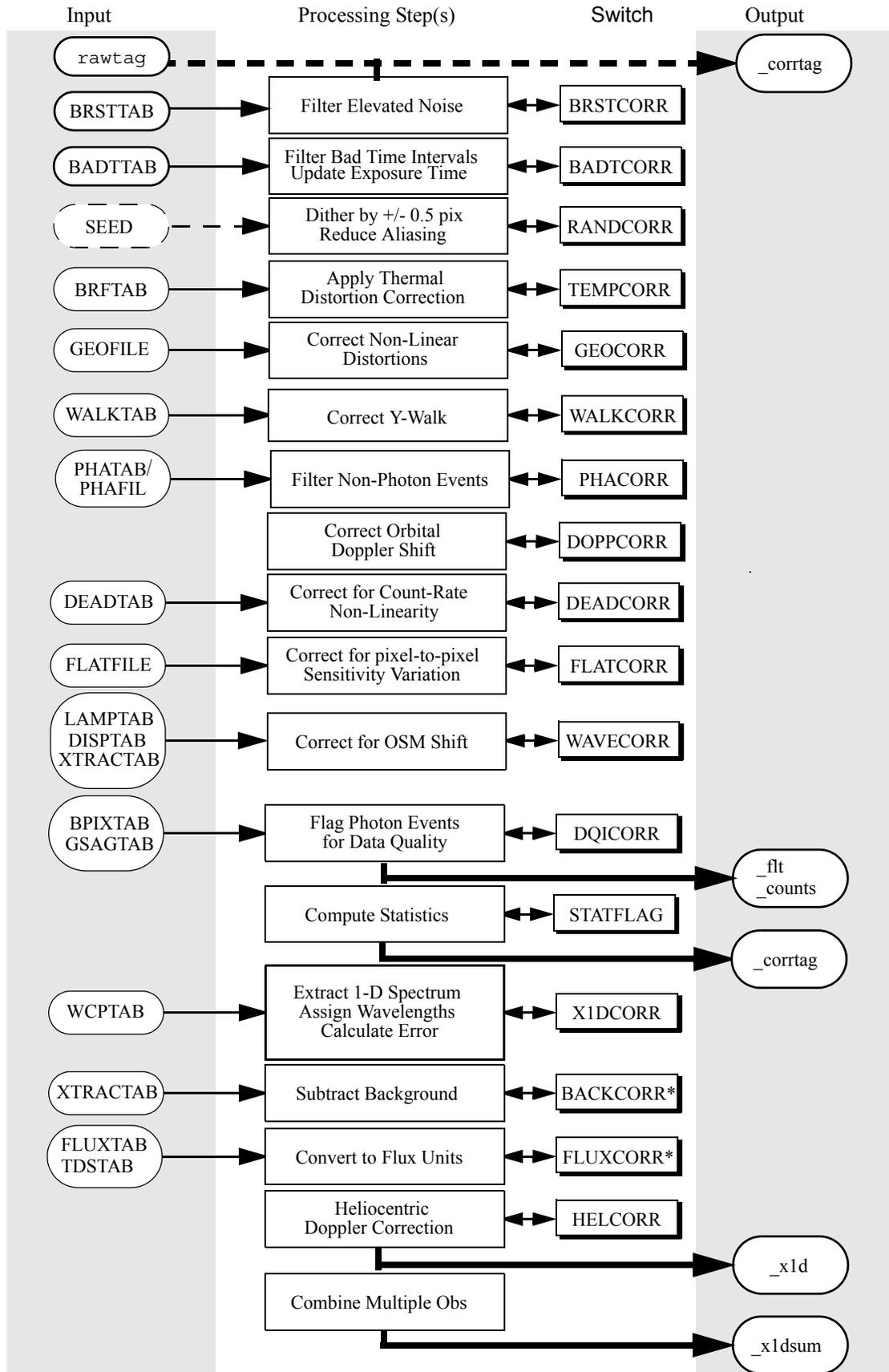
Calcos is comprised of three main components that provide calibration of the COS data. This structure incorporates modules that (1) correct the data for instrument effects (e.g. noise, thermal drifts, geometric distortions, pixel-to-pixel variations in sensitivity), (2) generate an exposure-specific wavelength-calibrated scale, and (3) extract and produce the final (one-dimensional) flux-calibrated (summed) spectrum for the entire observation. Both COS FUV and NUV TIME-TAG event lists and ACCUM images are fully calibrated by **calcos**. Target acquisition exposures are not calibrated by **calcos**, except for ACQ/IMAGE, although the raw data from these observations are available through the data archive.

As with *HST* calibration pipelines for other instruments, the choice of which operations are performed during calibration is controlled by calibration switches, which are stored in the primary FITS header. OPUS sets the switches that are appropriate for a given data type to PERFORM for steps to be carried out by **calcos**, and then **calcos** changes them to COMPLETE in the calibrated files. When OPUS creates the raw data files, it also populates the headers with the names of the appropriate reference files for each calibration operation. Any calibration step may

require zero, one, or more calibration reference files. Exactly how the data are processed depends on whether they are FUV TIME-TAG or ACCUM spectra, NUV TIME-TAG or ACCUM spectra, or NUV images. The names of the keywords containing the switches and reference file names were introduced in [Table 2.16](#), and their roles in the data reduction and the calibration steps are described in the following sections.

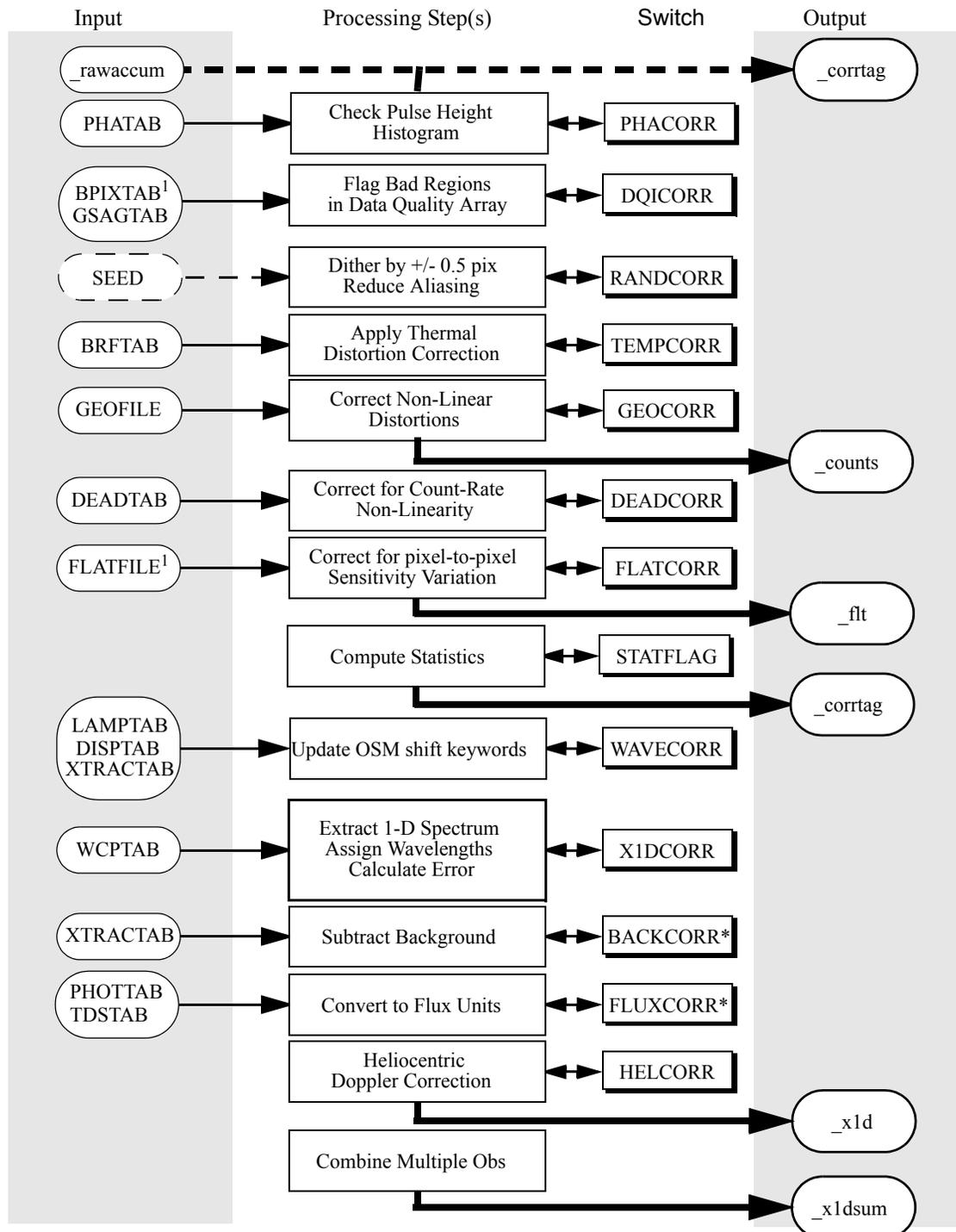
[Figure 3.1](#) - [Figure 3.5](#) show how a single raw file moves through the pipeline for FUV TIME-TAG, FUV ACCUM, NUV TIME-TAG and NUV ACCUM spectroscopic data, and for NUV images. Each Figure shows, from left to right, the input files, the processing steps performed by each module, and the output files. Note that in some instances, output files are created and then subsequently modified. In these cases, the file is shown at the end of a dashed arrow at the point it is created and again by a solid arrow at the point where it is finalized. Steps that apply only when spectra are extracted are marked with an * in [Figure 3.1](#) through [Figure 3.4](#). For ACCUM data, Doppler corrections are done onboard. Consequently, for these spectra certain reference files are transformed to the coordinate system of the data, rather than the other way around. We note on [Figure 3.2](#) and [Figure 3.4](#) when this is done.

Figure 3.1: FUV TIME-TAG Spectroscopic Pipeline Flow Chart.



* These steps are only implemented if X1DCORR=PERFORM.

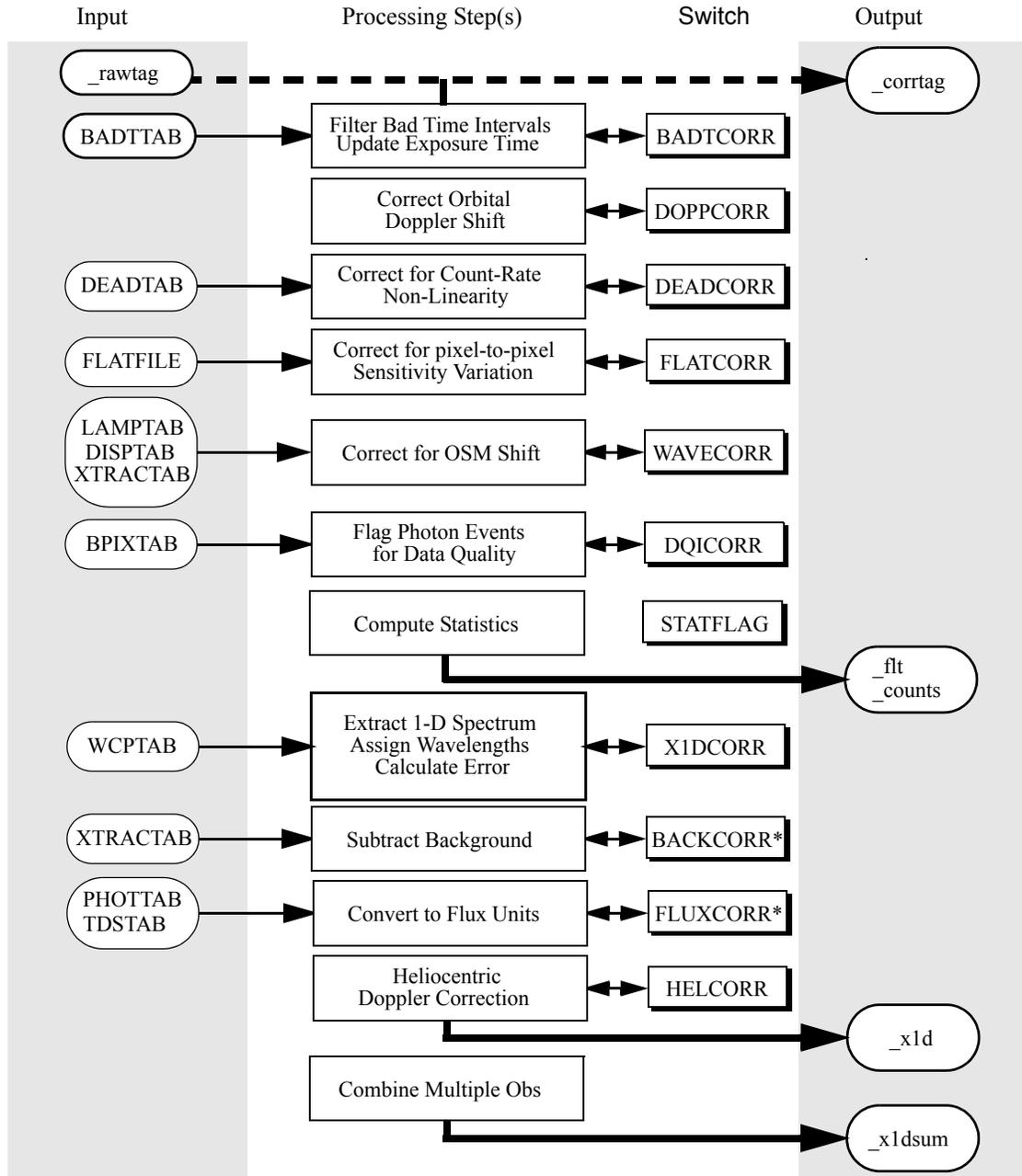
Figure 3.2: FUV ACCUM Spectroscopic Pipeline Flow Chart.



* These steps are only implemented if `X1DCORR=PERFORM`.

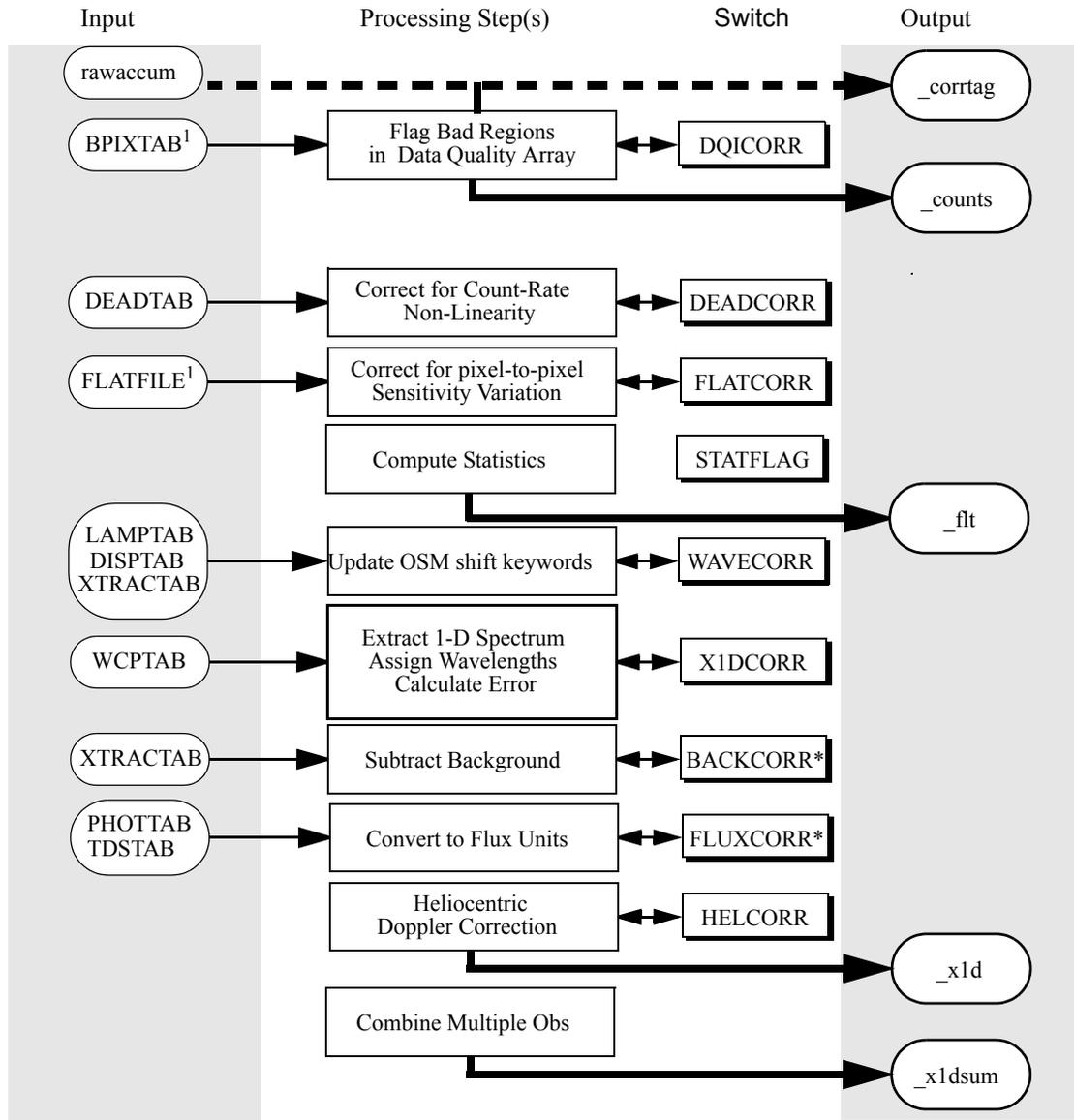
¹ Reference files that are transformed to the doppler corrected coordinate system of the data before being applied

Figure 3.3: NUV TIME-TAG Spectroscopic Pipeline Flow Chart



* These steps are only implemented if `X1DCORR=PERFORM`.

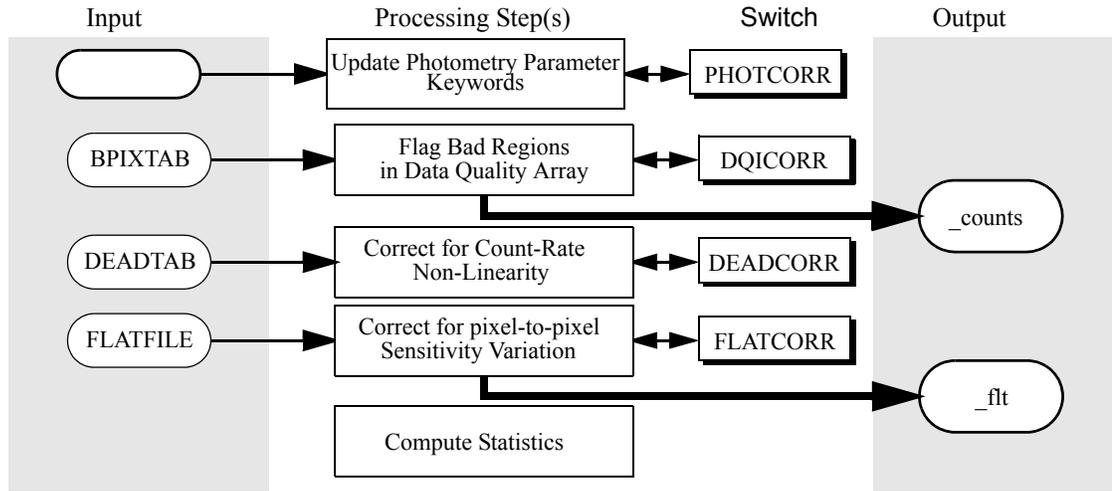
Figure 3.4: NUV ACCUM Spectroscopic Pipeline Flow Chart



* These steps are only implemented if X1DCORR=PERFORM.

¹ Reference files that are transformed to the doppler corrected coordinate system of the data before being applied.

Figure 3.5: NUV Image Pipeline Flow Chart



3.4 Descriptions of Spectroscopic Calibration Steps

This section provides a more detailed description of the calibration processing steps and algorithms applied by **calcos**, including the switches, reference file inputs, science file inputs, and the output products. Setting the calibration switch to **PERFORM** enables the execution of the corresponding pipeline calibration task.

Future modifications and updates to **calcos** will be announced in STScI Analysis Newsletters (STANs) and documented at the Web site:

<http://www.stsci.edu/hst/cos/documents/newsletters>

The calibration steps for FUV **TIME-TAG** data in the pipeline processing flow are described next. When present, each sub-section header begins with the switch that activates the module.

3.4.1 Initialization

When the pipeline is initiated, it first opens an association file to determine which files should be processed together. For **TIME-TAG** data, it also creates a **_corrtag** file before anything else is done. The initial contents of this file are simply a copy of the **_rawtag** file, except that new columns have been added to the **_corrtag** file. It is then updated throughout the running of the pipeline.

3.4.2 BRSTCORR: Search for and Flag Bursts

This module flags “event bursts” in the FUV TIME-TAG data for removal.

- Reference file: BRSTTAB, BRFTAB, XTRACTAB
- Input files: FUV_rawtag
- Header keywords updated: TBRST_A, TBRST_B (time affected by bursts in segments A and B), NBRST_A, NBRST_B (number of events flagged as bursts in segments A and B), EXPTIME, EXPTIMEA, EXPTIMEB.
- Updates _corrtag file for TIME-TAG data.

The COS FUV detectors are similar to the FUSE detectors, which experienced sudden, short-duration increases in counts while collecting data. These events, called bursts, led to very large count rates and occurred over the entire detector. Thus far, no bursts have been recorded on-orbit, and the default setting for BRSTCORR is OMIT. Nevertheless, since it is uncertain whether COS will suffer bursts at some point, the BRSTCORR module remains in the pipeline and is available to identify bursts and flag their time intervals should they occur. This module can only be applied to FUV TIME-TAG data.

The first step in the screening process is to determine the count rate over the whole detector, including stim pulses, source, background, and bursts. This rate determines which time interval from the BRSTTAB table to use for screening.

Screening for bursts is then done in two steps. The first step identifies large count rate bursts by calculating the median of the counts in the background regions, defined in the XTRACTAB table, over certain time intervals (DELTA_T or DELTA_T_HIGH for high overall count rate data). Events with count rates larger than MEDIAN_N times the median are flagged as large bursts.

The search for small count rate bursts is done iteratively, up to MAX_ITER. This step uses a boxcar smoothing of the background counts (taking the median within the box) and calculates the difference between the background counts and the running median. The boxcar smoothing is done over a time interval MEDIAN_DT or MEDIAN_DT_HIGH. Elements that have already been flagged as bursts are not included when computing the median. For an event to be flagged as affected by a small burst the difference between the background counts and the running median has to be larger than the following quantities:

1. A minimum burst count value: $BURST_MIN * DELTA_T$ (or $DELTA_T_HIGH$ for large overall count rates),
2. A predetermined number of standard deviations above the background: $STDREJ * \text{square_root}(\text{background counts})$,
3. A predetermined fraction of the source counts: $SOURCE_FRAC * \text{source counts}$.

The source counts value in 3) is the number of events in the source region defined in the XTRACTAB table minus the expected number of background counts within that region.

All events that have been identified as bursts are flagged in the data quality column (DQ in the `corrtag` table) with data quality bit = 64. In addition `calcos` updates the following header keywords to take into account time and events lost to burst screening: `TBRST_A` and `TBRST_B` (time lost to bursts in segments A and B); `NBRST_A`, `NBRST_B` (number of events lost to bursts in segments A and B), `EXPTIME`, `EXPTIMEA` and `EXPTIMEB`. Keywords `EXPTIMEA` and `EXPTIMEB` are the sums of the values in the input `_x1d` files. For `_x1dsum` files the value of the `EXPTIME` keyword is that corresponding to the larger value of `EXPTIMEA` and `EXPTIMEB`. The `EXPTIME` for each segment is contained in a table column in the first extension of the `x1dsum` file.

When running `calcos` a user can specify that the information about bursts be saved into a file. This output file contains four columns, each with one row per time interval (`DELTA_T` or `DELTA_T_HIGH`). Column 1 contains the time (seconds) at the middle of the time interval, column 2 contains the background counts for that time interval, column 3 contains a 1 for time intervals with large bursts and is 0 elsewhere, and column 4 contains a 1 for time intervals with small bursts and is 0 elsewhere.



Note: Although a systematic study has not been performed, as of August 2012, no bursts have been detected.

3.4.3 BADTCORR: Bad Time Intervals

This module flags time intervals in TIME-TAG data that have been identified as bad for some reason.

- Reference file: `BADTTAB`
- Input files: `FUV` and `NUV rawtag`
- Header keywords updated: `EXPTIME`, `EXPTIMEA` and `EXPTIMEB` (for `FUV` data), `NBADT`, or `NBADT_A` and `NBADT_B` (number of events flagged for `NUV` or `FUVA` and `B`, respectively) and `TBADT` or `TBADT_A` and `TBADT_B` (time lost to bad events in `NUV` or `FUVA` and `FUVB`, respectively).
- Updates `corrtag` file for TIME-TAG data.

The `BADTTAB` table lists zero or more intervals of time which will be excluded from the final spectrum for various reasons. This file is currently empty, but it could be populated by the COS team if events occur on orbit which they feel render data collected during specific time intervals not scientifically useful. It is also available for the convenience of the user. For example, the user may wish to eliminate observations obtained in the daytime portion of the orbit to minimize airglow contamination, or they may want to isolate a certain portion of an exposure. In these cases, modifying `BADTTAB` may be the most convenient means to accomplish this. Events in the `rawtag` file having times within any bad time interval in `BADTTAB` are flagged in the DQ column of the `corrtag` table with data quality = 2048. The exposure time is

updated to reflect the sum of the good time intervals. This step applies only to TIME-TAG data.

3.4.4 PHACORR: Pulse Height Filter

This module operates on FUV data and flags events whose pulse heights are outside of nominal ranges.

- Reference file: PHATAB, PHAFILE
- Input files: FUV rawtag, FUV rawaccum
- Header keywords updated: NPHA_A, NPHA_B, PHAUPPRA, PHAUPPRB, PHALOWRA, PHALOWRB

This module works differently for FUV TIME-TAG and ACCUM data. It is not used for NUV data.

For FUV TIME-TAG data, each photon event includes a 5 bit (0 - 31) Pulse Height Amplitude (PHA). The value of the pulse height is a measure of the charge produced by the microchannel plate stack, and can be used to identify events which are likely due to cosmic rays or detector background. The PHATAB reference file lists lower and upper pulse height thresholds expected for valid photon events for each detector segment. The PHACORR module compares each event's pulse height to these thresholds, and if the pulse height is below the Lower Level Threshold (LLT) or above the Upper Level Threshold (ULT), the event is flagged in the DQ column of the corrtag table with a data quality bit of 512. The upper and lower thresholds are also written to the PHALOWRA (PHALOWRB) and PHAUPPRA (PHALOWRB) keywords in the output data files for segment A (B), while the number of events flagged is written to the NPHA_A and NPHA_B keywords.

Default values of the lower (LLT) and upper (ULT) thresholds have been chosen based on the properties of the detector and are implicit in data used when generating other reference files (e.g. FLUXTAB)

With continuing exposure to photons, pulses from the MCPs have smaller amplitudes, a phenomenon known as “gain sag”. As this occurs, the thresholds in the PHATAB will be updated to maximize the number of real events counted. Which PHATAB is used for data collected at a particular time will be handled by the USEAFTER date keyword in the calibration file header.

The PHAFILE reference file is an alternative to the PHATAB, and allows pulse-height limits to be specified on a per-pixel basis rather than a per-segment basis. The PHAFILE has a primary header and four data extensions, consisting of the FUVA PHA lower limits, FUVA PHA upper limits, FUVB PHA lower limits, and FUVB PHA upper limits respectively. The use of a PHAFILE instead of a PHATAB (if both are specified and PHACORR = 'PERFORM', the PHAFILE will take precedence) allows a number of adjustments, including (for example) the use of a lower PHA threshold in gain-sagged regions, thus allowing more background events to be filtered out while still continuing to detect photon events in gain-sagged regions. As of August 2012, no PHAFILE has been produced by the COS team, but in the future one or more

such files may be produced for use with FUV TIME-TAG data. Note that the use of a PHAFILE requires **calcos** 2.14 or later.



Modifying these threshold values could lead to incorrect results in the calibrated products, and should therefore be done with EXTREME caution.

For FUV ACCUM data, pulse height information is not available for individual events. However, a 7 bit (0 - 127) Pulse Height Distribution (PHD) array, containing a histogram of the number of occurrences of each pulse height value over the entire detector segment, is created onboard for each exposure. PHACORR compares the data in this pha file to the values in the PHATAB file. Warnings are issued if the peak of the distribution (modal gain) does not fall between the scaled values of LLT and ULT; or if the average of the distribution (mean gain) does not fall between the MIN_PEAK and MAX_PEAK values in PHATAB. The PHALOWRA and PHAUPPRA, or PHALOWRB and PHAUPPRB keywords are also populated in the output files with the LLT and ULT values from the PHATAB.

3.4.5 WALKCORR: Y-Walk Correction

This module corrects for the tendency of the FUV XDL detector to register events at lower Y-locations as gain sag accumulates.

- Reference file: WALKTAB
- Input files: FUV rawtag
- Header keywords updated: None.
- Updates corrtag file for TIME-TAG data.

The WALKTAB table contains parameters for the polynomial used to correct Y-walk caused by gain sag. Y-walk causes events on the XDL detector to appear to have originated lower on the detector than their actual location. Currently, the Y-walk correction involves an equation of the form:

$$\begin{aligned} dx &= x_1 * (PHA - x_0) + x_2 * (PHA - x_0)^2 \\ dy &= y_1 * (PHA - y_0) + y_2 * (PHA - y_0)^2 \end{aligned}$$

where x_n and y_n are coefficients which are, in turn, calculated from the event X and Y location, and a variable-length series of other coefficients stored in the reference file. For example, for the case with three X coefficients and three Y coefficients, the x_n coefficients are calculated as follows:

$$\begin{aligned} x_0 &= a_{000} + a_{100} * x + a_{010} * y + a_{200} * x^2 + a_{110} * x * y + a_{020} * y^2 \\ x_1 &= a_{001} + a_{101} * x + a_{011} * y + a_{201} * x^2 + a_{111} * x * y + a_{021} * y^2 \\ x_2 &= a_{002} + a_{102} * x + a_{012} * y + a_{202} * x^2 + a_{112} * x * y + a_{022} * y^2 \end{aligned}$$

where a_{mno} is the item at location m,n,o in the coefficient matrix.

The current Y-walk correction is considerably simpler than this, with many of the coefficients equal to zero:

$$\text{FUVA: } dx = 0; dy = (\text{PHA} - 12) * 0.47$$

$$\text{FUVB: } dx = 0; dy = (\text{PHA} - 14) * 0.47$$

However, it is possible that future revisions will result in a more complex correction, and the correction may certainly vary with time and lifetime position.

The WALKTAB table contains the following items for each FUV segment:

- X_0 : subtracted from x before evaluating coefficient polynomials. Set to 8192.
- Y_0 : as X_0 , but for y . Set to 512.
- N_X : Number of terms in X
- N_Y : Number of terms in Y
- N_PHA_COEFF : Number of coefficients in polynomial for PHA
- $XCOEFF$: Array of $N_X * N_Y * N_PHA_COEFF$ coefficients for determining dx
- $YCOEFF$: As $XCOEFF$, but for determining dy .

3.4.6 RANDCORR: Add Pseudo-Random Numbers to Pixel Coordinates

This module adds a random number between -0.5 and +0.5 to each x and y position of a photon detected by the FUV detectors.

- Reference file: none
- Input files: FUV rawtag, FUV rawaccum
- Header keywords updated: RANDSEED
- Updates corrtag file for TIME-TAG data and a virtual corrtag file for ACCUM data.

For FUV TIME-TAG data RANDCORR adds random numbers to the raw coordinates of each event, i.e.:

$$XCORR = RAWX + \Delta x$$

$$YCORR = RAWY + \Delta y$$

Where Δx and Δy are uniformly distributed, pseudo-random numbers in the interval $-0.5 < \Delta x, \Delta y \leq +0.5$.

The result of this operation is to convert the raw, integer pixel values into floating point values so that the counts are smeared over each pixel's area.

For FUV ACCUM data, a pseudo TIME-TAG list of x and y values is created with an entry for each recorded count. Next, a unique Δx and Δy are added to each entry.

If the RANDSEED keyword in the raw data file is set to its default value of -1, the system clock is used to create a seed for the random number generator. This seed value is then written to the RANDSEED keyword in the output files. Alternatively, an integer

seed (other than -1) in the range -2147483648 to +2147483647 can be specified by modifying the `RANDSEED` keyword in the raw data file. Doing so will ensure that identical results will be obtained on multiple runs.

`RANDCORR` is only applied to events in the active area of the detector, as defined in the `BRFTAB`. Stim pulses, for example, do not have this correction applied.

3.4.7 TEMPCORR: Temperature-Dependent Distortion Correction

This module corrects for linear distortions of the FUV detector coordinate system that are caused by changes in the temperature of the detector electronics.

- Reference file: `BRFTAB`
- Input files: `FUV rawtag`, `FUV rawaccum`
- Header keywords updated: none
- Updates `corrtag` file for `TIME-TAG` data and `flt` file for `ACCUM`.

The FUV XDL detector has virtual, not physical, detector elements that are defined by the digitization of an analog signal. The charge packet associated with a photon event is split and transported to opposite sides of the detector where the difference in travel time of the two packets determines the location of the photon event on the detector. Since the properties of both the delay line and the sensing electronics are subject to variations as a function of temperature, apparent shifts and stretches in the detector format can occur.

To measure the magnitude of this effect, electronic pulses (Figure 1.6) are recorded at two reference points in the image (“electronic stims”) at specified time intervals throughout each observation. `TEMPCORR` first determines the locations and separations of the recorded stim positions and then compares them to their expected locations in a standard reference frame (as defined in columns `SX1`, `SY1`, `SX2`, and `SY2` of the `BRFTAB` file). The differences between the observed and reference stim positions are used to construct a linear transformation between the observed and reference frame locations for each event (or pseudo-event in the case of `ACCUM` data). `TEMPCORR` then applies this transformation to the observed events, placing them in the standard reference frame.

3.4.8 GEOCORR and IGEOCORR: Geometric Distortion Correction

This module corrects geometric distortions in the FUV detectors.

- Reference file: `GEOFILE`
- Input files: `FUV rawtag`, `FUV rawaccum`
- Header keywords updated: none
- Updates `corrtag` file for `TIME-TAG` data and `flt` file for `ACCUM`.

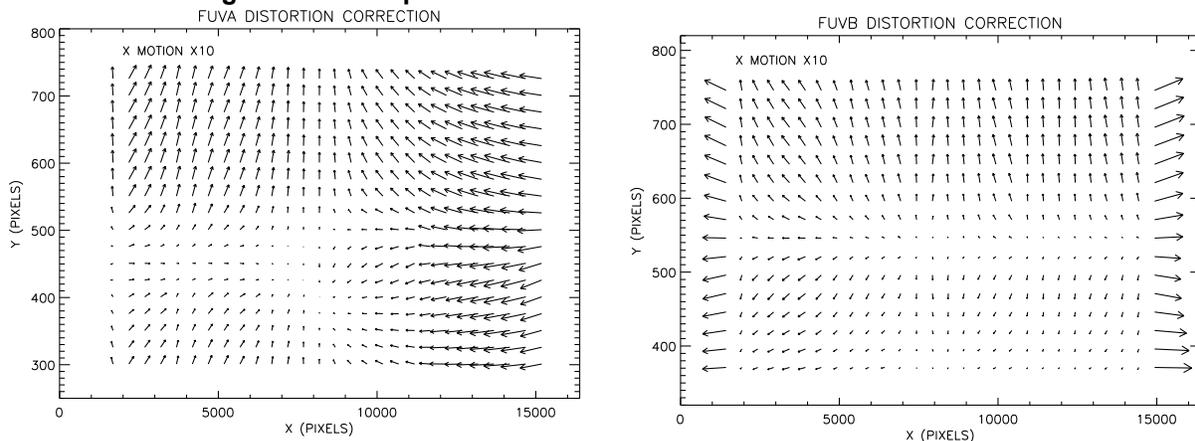
The GEOCORR module corrects for geometric distortions due to differences between the inferred and actual physical sizes of pixels in the FUV array (ground measurements indicated that geometric distortions in the NUV MAMA are negligible). It produces a geometrically corrected detector image with equal sized pixels. This is done by applying the displacements listed in the reference file, GEOFILE, which lists the corrections in x and y for each observed pixel location. The geometric distortion varies across the detector, and the GEOFILE gives the distortion only at the center of each pixel. If IGEOCORR is ‘PERFORM’ (the default), the displacements to correct the distortion at (XCORR, YCORR) will be interpolated to that location, which includes a fractional part (even before geometric correction) due to TEMPCORR and RANDCORR. If IGEOCORR is ‘OMIT’, the correction will be taken at the nearest pixel to (XCORR, YCORR).

GEOFILE was created by using a ray-trace analysis of the COS FUV optical system. A set of wavelength calibration exposures was taken while stepping the aperture mechanism in the cross-dispersion direction to create an image of dispersed line profiles. The ray trace and measured line positions were compared to determine the shift between the measured (uncorrected) and predicted (corrected) line positions (see Figure 3.6).

The distortion corrections are given as images in the GEOFILE in the following order:

- Extension 1 contains an image of the X distortions for the FUVA
- Extension 2 contains an image of the Y distortions for the FUVA
- Extension 3 contains an image of the X distortions for the FUVB
- Extension 4 contains an image of the Y distortions for the FUVB

Figure 3.6: A Map of the FUV Geometric Correction



A map of the FUV geometric correction, scaled by a factor of 10 in the x-direction for illustration purposes, for detector segment A (left) and segment B (right) in user coordinates. The arrow points from the observed to the corrected position.

3.4.9 DOPPCORR: Correct for Doppler Shift

This module corrects for the effect that the orbital motion of *HST* has on the arrival location of a photon in the dispersion direction.

- Reference files: DISPTAB, XTRACTAB
- Input files: FUV/NUV rawtag, FUV/NUV rawaccum
- Header keywords updated: none
- Updates corrtag file for TIME-TAG data

During a given exposure the photons arriving on the FUV and NUV detectors are Doppler shifted due to the orbital motion of *HST*. The orbital velocity of *HST* is 7.5 km/s, so spectral lines in objects located close to the orbital plane of *HST* can be broadened up to 15 km/s, which can be more than a resolution element.

DOPPCORR is the **calcos** routine which corrects for the orbital motion of *HST*. It operates differently on TIME-TAG and ACCUM files:

For TIME-TAG files the raw events table contains the actual detector coordinates of each photon detected, i.e., the photon positions will include the smearing from the orbital motion. In this case DOPPCORR will add an offset to the pixel coordinates (the XCORR column) in the events table to undo the Doppler broadening. The corrected coordinates are written to the column XDOPP in the corrtag file for both FUV and NUV data.

For ACCUM files Doppler correction is applied onboard and is not performed by **calcos**. This means, however, that the pixel coordinates of a spectral feature can differ from where the photon actually hit the detector - a factor which affects the data quality initialization and flat-field correction. Therefore for ACCUM images DOPPCORR shifts the positions of pixels in the bad pixel table to determine the maximum bounds that could be affected. It is also used to convolve the flat-field image by an amount corresponding to the Doppler shift which was computed on orbit. The information for these calculations are contained in the following header keywords:

- DOPPONT: True if Doppler correction was done onboard.
- ORBTPERT: Orbital period of *HST* in seconds.
- DOPMAGT: Magnitude of the Doppler shift in pixels.
- DOPZEROT: Time (in MJD) when the Doppler shift was zero and increasing.

The “T” suffix at the end of each of these keywords indicates that they were derived from the onboard telemetry, whereas the other keywords described below were computed on the ground from the orbital elements of *HST*. The two sets of keywords can differ by a small amount, but they should be nearly identical.

DOPPCORR assumes that the Doppler shifts vary sinusoidally with time according to the orbital movement of *HST*. The following keywords are used to

perform the correction and are obtained from the first extension (EVENTS) in the `_rawtag`:

- `EXPSTART` - start time of the exposure (MJD)
- `DOPZERO` - the time (MJD) when the Doppler shift was zero and increasing (i.e., when *HST* was closest to the target)
- `DOPPMAG` - The number of pixels corresponding to the Doppler shift (used only for shifting the data quality flag arrays and flat fields)
- `ORBITPER` - the orbital period of *HST* in seconds

The data columns used in the correction are `TIME` (elapsed seconds since `EXPSTART`) and `RAWX` (position of photon along dispersion direction). The Doppler correction to be applied is then

$$\text{SHIFT} = -(\text{DOPPMAG}/(c*d))*\lambda(\text{XCORR})*\sin(2*\pi*t/\text{ORBITPER})$$

where c is the speed of light (km/s), d is the dispersion of the grating used in the observation ($\text{\AA}/\text{pixel}$), $\lambda(\text{XCORR})$ is the wavelength at the `XCORR` position being corrected (obtained from the dispersion solution for that grating and aperture in the `DISPTAB` reference file) and t is defined as

$$t = (\text{EXPSTART} - \text{DOPZERO})*86400 + \text{TIME}$$

where the factor of 86400 converts from days to seconds.

3.4.10 DEADCORR: Nonlinearity Correction

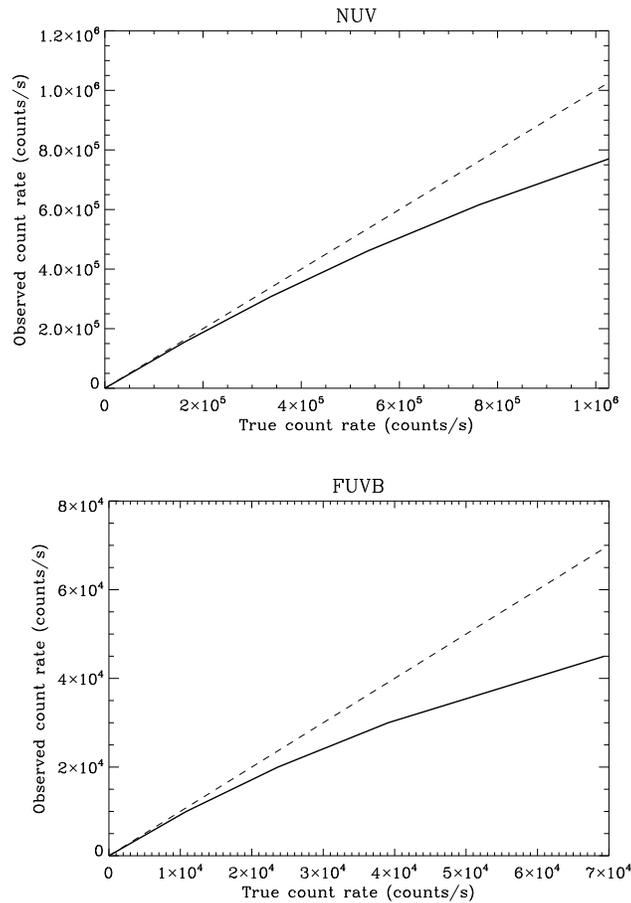
This module corrects for count rate dependent non-linearities in the COS detectors.

- Reference file: `DEADTAB`
- Input files: `FUV/NUV rawtag`, `FUV/NUV rawaccum`, `NUV images`
- Header keywords updated: none
- Updates `corrtag` file for `TIME-TAG` data and `flt` for `ACCUM`.

`DEADCORR` corrects for non-linear photon detection in the COS detector electronics. Both the FUV and NUV detector electronics have a small temporal overhead when counting events. This overhead becomes noticeable when the count rates become large.

The efficiency of the detector's photon counting is calculated as the ratio of the true count rate and the observed count rate. This value is referred to as the *deadtime*. The *deadtime* for each detector is modeled and the reference file `DEADTAB` contains a lookup table of the correction for various count rates. [Figure 3.7](#) shows how the measured count rates deviate from the actual count rates as a function of the actual count rate for the NUV detector, and segment B of the FUV detector (the FUV segment A curve is nearly identical).

Figure 3.7: FUV and NUV Deadtime



The solid curves are the observed count rates versus true count rates for the COS detectors and the dashed lines are for perfect detectors. TOP: The NUV MAMA. BOTTOM: Segment B of the FUV XDL detector (the curve for Segment A is nearly identical). Significant deviations from the true count rates occur at about 15,000 counts per second for the XDL detectors, and at roughly 10 times this rate for the MAMA.

For TIME-TAG data the deadtime correction is computed every 10 seconds. The observed count rate is the number of events within that time interval, and the deadtime factor is determined by interpolation within the values in DEADTAB. The values in the EPSILON column in the _corrtag file for events within that time interval will then be divided by the deadtime factor. For ACCUM data the observed average count rate is taken from a header keyword for the digital event counter. The deadtime factor is then found by interpolation within the DEADTAB, the same as for TIME-TAG data, and the science and error arrays will be divided by the deadtime factor.

3.4.11 FLATCORR: Flat-field Correction

This module corrects for pixel-to-pixel non-uniformities in the COS detectors.

- Reference file: `FLATFILE`
- Input files: `FUV/NUV rawtag`, `FUV/NUV rawaccum`, `NUV images`
- Header keywords updated: none
- Updates `corrtag` file for `TIME-TAG` data and `flt` for `ACCUM`.

The FLATCORR step corrects for high frequency, pixel-to-pixel sensitivity differences across the detector. It uses a flat-field image located in the file specified by the `FLATFILE` header keyword. Figure 3.8 shows an NUV flat. For spectroscopic data, any wavelength dependence of the detector response or remaining low frequency spatial variations are removed by the flux calibration step (FLUXCORR, Section 3.4.17). Flat fielding is performed in geometrically corrected space, and because the pixel-to-pixel variations should be largely wavelength independent, only one reference image is used per detector or detector segment (NUV, FUVA, and FUVB). The flat-field correction is applied differently for `TIME-TAG` and `ACCUM` mode data for both spectroscopic and imaging modes.

For spectroscopic `TIME-TAG` exposures, each photon in the events list is individually corrected. In the `_corrtag` file, the photon weight in the `EPSILON` column is divided by the flat-field value at the event's detector location rounded to the nearest pixel (`XCORR`, `YCORR` for FUV; `RAWX`, `RAWY` for NUV).

For spectroscopic `ACCUM` mode data, photons are summed into an image onboard by the COS electronics. To compensate for the motion of *HST* during the observation, spectroscopic exposures are normally taken with Doppler compensation performed during the accumulation (science header keyword `DOPPONT=TRUE`). During Doppler compensation, photon locations are shifted as the data are received, and the underlying flat field at each imaged pixel is an average of the original pixel position sensitivities. FLATCORR replicates this averaging for the flat-field correction using the same control parameters as those onboard (`DOPPMAGT`, `DOPZEROT`, `ORBTPERT`) if `DOPPCORR=PERFORM` (Section 3.4.9). The convolved flat-field image is divided into the `rawaccum` data, which is then divided by the exposure time to create the `flt` image file.

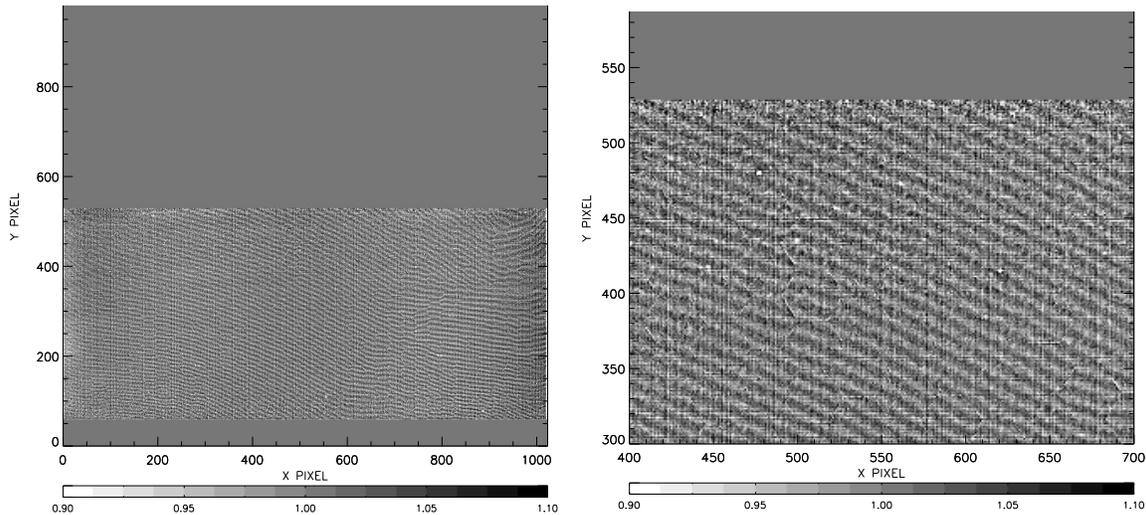
NUV images using the mirrors are not Doppler corrected. In this case, `DOPPCORR=OMIT`, and the input data are divided by the flat field without convolution.

For both the `flt` and `counts` files, error arrays are created based on counting statistics (Section 2.7), but they are not used in further processing.

It was discovered on-orbit, that the NUV suffers some vignetting. This causes a structure in “pixel space”, affecting roughly the first 200 pixels of all three spectral stripes by as much as 20%. The NUV flat field was originally modified to correct for this effect, but variation in the vignetting caused sufficient errors that the vignetting is no longer included in the NUV flat field. Work continues on grating-specific vignetting corrections.

For the FUV channels, the ground flats proved inadequate. Consequently, the current FUV flats correct only for the effects of grid wires in the M-mode gratings. The project is currently investigating a variety of correction schemes to eliminate the flat-field effects in FUV spectra.

Figure 3.8: Flat-field Images of the NUV MAMA Detector.



The image at left shows the full detector, and the one on the right has been enlarged to illustrate structure in the flat-field images. The hex structure associated with the micro-channel plate is visible in both FUV and NUV flat fields.

3.4.12 WAVECORR: Wavecal Correction

For spectroscopic data, this module determines location of the wavelength calibration spectrum on the detector relative to a template, and then applies zero point shifts to align the wavecal and the template.

- Reference files: LAMPTAB, WCPTAB, XTRACTAB, DISPTAB
- Input files: FUV/NUV rawtag, FUV/NUV rawaccum
- Header keywords updated: SHIFT1[A-C], SHIFT2[A-C], LMP_ONi, LMPOFFi, LMPDURi, LMPMEDI.
- Updates corrtag file for TIME-TAG data.
- Creates lampflash file for TAGFLASH data.

The wavecal step of **calcos** determines the shift of the 2-D image on the detector along each axis resulting from thermal motions and drifts within an OSM encoder position. This step applies only to spectroscopic data, TIME-TAG and ACCUM, for both the FUV and NUV detectors. The shifts are determined from one or more contemporaneous wavelength calibration observations of a spectral line lamp (wavecal) which must be obtained without moving the OSM between the science and wavecal exposures.

There are two types of wavecals. For ACCUM data the spectrum of the calibration lamp is contained in an exposure that is separate from that of the science (AUTO or GO wavecals). For TIME-TAG data the wavecals can also be separate exposures, but the default when observing with the PSA aperture is TAGFLASH mode. In the TAGFLASH mode the line lamp is turned on and off (flashed) one or more times during a single science exposure, producing a wavecal spectrum that is offset in the cross-dispersion direction from the science spectrum (See [Figure 1.7](#), and [Figure 1.9](#)). The algorithm used to determine the shifts is the same in either case, but the way that the shift is determined at the time of the observation differs. Thus, we begin by describing how the offsets are found.

Determining the offsets:

For each wavecal, the location of the spectrum in the cross-dispersion direction is determined by collapsing the spectrum along the dispersion direction using the extraction slope defined in the XTRACTAB table (SLOPE). The location of the brightest pixel, after boxcar smoothing, is taken as the spectrum location and that location is compared to the nominal position defined in the XTRACTAB table. The offsets from nominal positions for segments A and B (FUV) or stripes A, B, and C (NUV) are recorded in the lampflash file (which is created at this stage) in the SHIFT_XDISP field. The two FUV segments are processed independently. Cross-dispersion shifts are determined for each NUV stripe and then the average is computed and applied to all three stripes. The sign of the SHIFT_XDISP entry is positive if the spectrum was found at a larger pixel number than the nominal location.

To determine the offsets in the dispersion direction, the wavecal spectrum is collapsed along the cross-dispersion direction and compared to the template wavecal (LAMPTAB) taken with the same grating and central wavelength. For the NUV wavecal spectra for each stripe are determined independently. The positions are determined from a least squares fit to a shifted and scaled version of the template spectrum. The maximum range for shifting the wavecal and template wavecal spectra is defined by the value of XC_RANGE in the WCPTAB table. **Calcos** takes into account the FPPOS of the wavecal spectrum by shifting it by FP_PIXEL_SHIFT (from the column in the LAMPTAB) before fitting it to the template wavecal. The final shift is stored as SHIFT_DISP in the lampflash file and the minimum value of chi squared is stored in the CHI_SQUARE array.

Applying the offsets:

The way the offsets are applied to the spectral data depends on whether the data were obtained with AUTO or GO wavecals or with TAGFLASH wavecals. For AUTO or GO wavecals, the wavecals are obtained at different times than the spectral data and temporal interpolation is done to determine the appropriate shifts. For TAGFLASH data, the wavecals are interspersed with the spectral data, allowing more precise and, consequently, more intricate corrections to be made. In either case, the result is the X[Y] FULL entries in the corrtag file. Because, as we shall see, the corrections can be time dependent, the differences between X[Y] CORR and X[Y] FULL can also be time dependent. This step of the calibration amounts to a time dependent translation of the detector coordinate system to a coordinate system relative to the wavecal spectrum, which is more appropriate for wavelength calibration.

AUTO or GO wavecal

For ACCUM science exposures which are bracketed by AUTO or GO wavecal observations, the shifts determined from the bracketing wavecal exposures are interpolated (linearly) to the middle time of the science observation, and the interpolated values are assigned to the SHIFT1[A-C] and SHIFT2[A-C] keywords in the science data header. If there is just one wavecal observation in a dataset, or if there are more than one but they don't bracket the science observation, the SHIFT1[A-C] and SHIFT2[A-C] keywords are just copied from the nearest wavecal to the science data header.

For non-TAGFLASH TIME-TAG science exposures bracketed by AUTO or GO wavecal observations, the shifts determined from the wavecal observations are interpolated (linearly) so that each event in the corrtag file is shifted according to its arrival time. The SHIFT1[A-C] and SHIFT2[A-C] keywords recorded in the science data header are in this case the averages of the values applied. As in the ACCUM case, if there is only one wavecal observation in a dataset, or if there are more than one but they do not bracket the science observation, the SHIFT1[A-C] and SHIFT2[A-C] keywords are just copied from the nearest wavecal to the science data header.

TAGFLASH DATA

A TAGFLASH wavecal is a lamp exposure that is taken concurrently with a TIME-TAG science exposure, and the photon events for both the wavecal lamp and the science target are mixed together in the same events table. In many respects, TAGFLASH wavecal observations are handled differently from conventional wavecal observations.

The nominal start and stop times for each lamp flash are read from keywords in the corrtag table. The actual start and stop times can differ from the nominal times, so **calcos** determines the actual times (restricted to being within the nominal start-to-stop intervals) by examining the number of photon events within each 0.2-second interval in the wavecal region defined in the XTRACTAB table. A histogram of the count rate is constructed. The histogram is expected to have one peak near zero, corresponding to dark counts, and another at high count rate, due to the lamp illumination. The average count rate when the lamp is on is taken to be the count rate for the second peak of the histogram. The lamp turn-on and turn-off times are taken to be the times when the count rate rises above or sinks below half the lamp-on count rate.

Calcos uses the time of the median photon event within a lamp turn-on and turn-off interval as the time of the flash. The keywords LMP_ONi and LMP_OFFi (i is the flash number) are updated with the actual turn-on and turn-off times, in seconds, since the beginning of the science exposure. The keywords LMPDURi and LMPMEDI are updated with the actual duration and median time of the flash.

As before, the cross dispersion location of each wavecal spectrum is determined by collapsing it along the dispersion direction and comparing it with the template in the XTRACTAB table to produce the SHIFT_XDISP entries in the lampflash file. The wavecal spectrum is then collapsed along the cross-dispersion direction to produce a 1-D spectrum that is fit to the template spectrum to obtain the SHIFT_DISP entries. Typically there will be more than one wavecal flash during a science exposure; so the shifts will be piece-wise linearly interpolated between flashes. The SHIFT1[A-C] and

SHIFT2[A-C] values that are recorded in the science data header are the average of the shift values found from the different flashes.

Additional Functions: WAVECORR also corrects the `_flt` and `_counts` files which result from both ACCUM and TIME-TAG science data for the offsets in the dispersion and cross-dispersion directions. However, since these images are in pixel space they can only be corrected by an integer number of pixels. The `_flt` and `_counts` images are corrected by the nearest integer to SHIFT1[A-C] and SHIFT2[A-C]. DPIXEL1 [A-C] is the average of the difference between XFULL and the nearest integer to XFULL, where XFULL is the column by that name in the `_corrtag` table. This is the average binning error in the dispersion direction when the `_flt` and `_counts` images are created from the `_corrtag` table. DPIXEL1 [A-C] is zero for ACCUM data. This shift is used when computing wavelengths during the X1DCORR step.

3.4.13 DQICORR: Initialize Data Quality File

This module identifies pixels which are suspect in some respect and creates the DQ extension for the `flt` and `counts` images.

- Reference file: FUV/NUV BPIXTAB, FUV GSAGTAB
- Input files: FUV/NUV rawtag, FUV/NUV rawaccum, NUV images
- Header keywords updated: none
- Updates the DQ column of the `corrtag` files and the DQ image array.

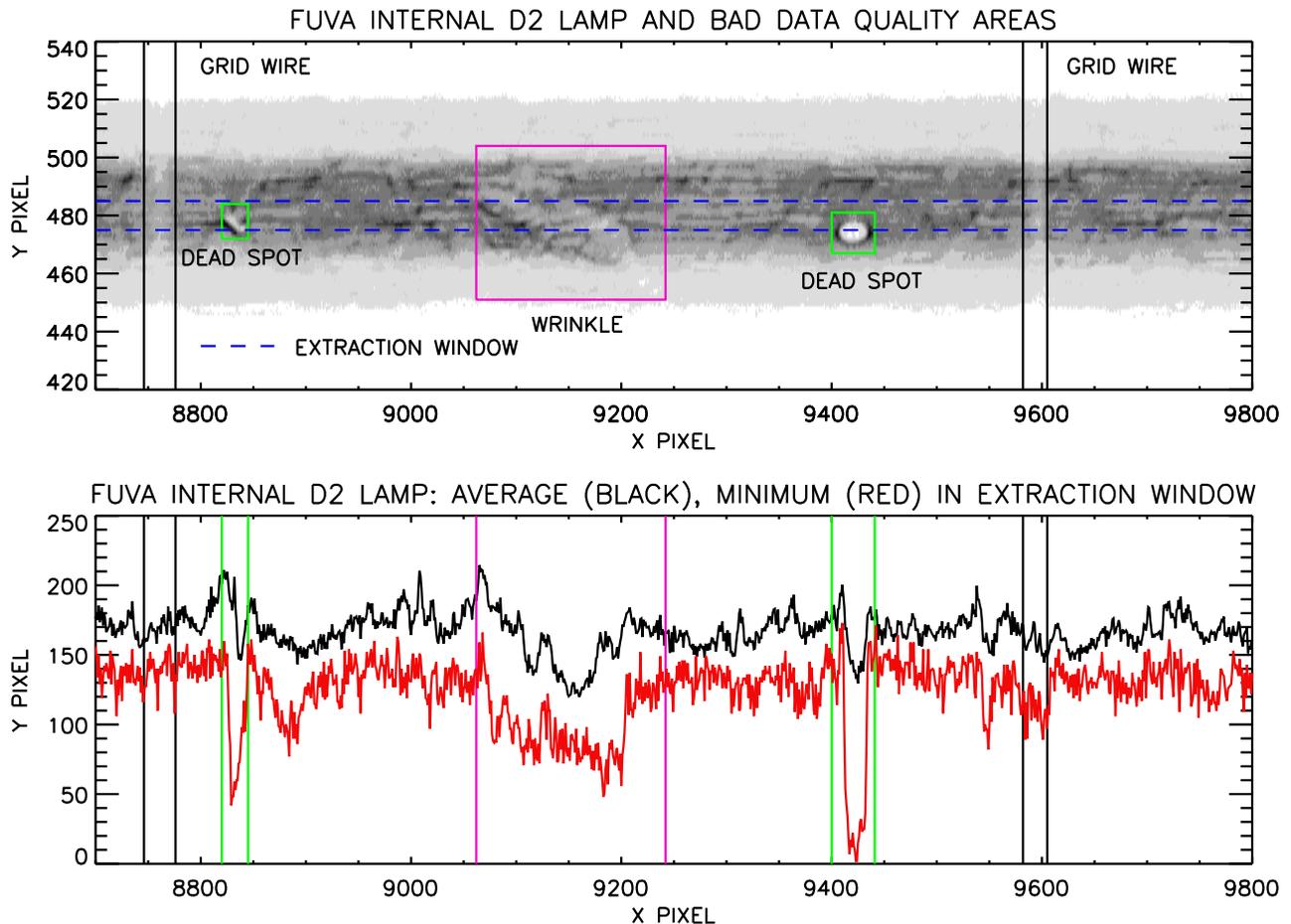
The DQICORR step maps data quality flags for suspect detector pixels listed in the BPIXTAB table to the science data. The COS data quality flags are discussed in [Section 2.7.2](#) and are listed in [Table 2.19](#). [Figure 3.9](#) shows examples of the types of regions isolated by the DQ flags and the effect that they can have on an extracted spectrum. DQICORR proceeds differently for TIME-TAG and ACCUM mode exposures, but the flags in the `flt` and `counts` images are created similarly in preparation for spectral extraction. Consequently, we describe each mode separately.

TIME-TAG: DQICORR compares the XCORR, YCORR pixel location of each event in the `corrtag` file to the rectangular regions flagged in the BPIXTAB table. The value in the DQ column for that event is then updated with the flags of all the regions (if any) that contain that pixel location. When the `flt` and `counts` images are generated from the `corrtag` file, photons which arrived during bad times or bursts are omitted from the image and ERR array. For FUV data, events whose PHAs were flagged as out of bounds are omitted as well. However, data with spatial DQ flags are retained at this stage. Several of these blemishes are illustrated in [Figure 3.9](#)

The third FITS extension of the `flt` and `counts` files is an array of data quality values generated directly from the BPIXTAB table. If DOPPCORR=PERFORM, the BPIXTAB locations are Doppler-smearred and the flags from all neighboring pixels that contribute to the `flt` and `counts` image pixels are combined.

ACCUM: For ACCUM exposures, the `rawaccum` image file will already have a third FITS extension of data quality values if any pixel had been flagged when constructing the raw image. The extension will be a null image if all initial data quality flags are zero. This is usually the case for NUV data, but not FUV. For FUV ACCUM exposures, photons are collected for only part of the detector segment and an initial data quality array is created to mark the pixels outside those subimage boundaries (`flag=128`, outside active area). When `calcos` creates the `flt` and `counts` images, it first converts the `rawaccum` image to a pseudo-time-tag table. In this table, the DQ column is updated with the DQ flags from `BPIXTAB` just as for the TIME-TAG data. In addition, the third extension of the `flt` and `counts` files contains a Doppler-smearred version of the `BPIXTAB` table, but it also includes the initial flag assignments in the `rawaccum` DQ extension.

Figure 3.9: The FUV Flat Field.



An FUV flat field obtained during ground testing illustrates the different kinds of blemishes and regions of lower sensitivity that occur. These regions are flagged in `BPIXTAB` according to the feature type, e.g., a ‘wrinkle’ is a kind of detector flaw and grid wire is an example of a detector shadow.

3.4.14 STATFLAG: Report Simple Statistics

This module computes some statistical measures that provide general information about COS science observations.

- Reference file: XTRACTAB, BRFTAB
- Input files: `flt`, `counts`, `x1d`, `lamptab`
- Header keywords updated: `NGOODPIX`, `GOODMEAN`, `GOODMAX`

STATFLAG enables the reporting of statistics for COS observations. STATFLAG is enabled by default for all science observations and operates on `x1d`, `counts`, and `flt` data products. STATFLAG is intended to provide a very basic statistical characterization of the events and locations on the detectors that are known to be good. The SDQFLAGS header keyword (Serious Data Quality FLAGS), indicates which DQ values (see [Table 2.19](#) for definitions) should be excluded from the statistical calculations. Numerically, the default value of SDQFLAGS for NUV is 184. For FUV, SDQFLAG varies with grating. For FUV/G130M and G160M the default value is 8376, and 8380 for FUV/G140L.



To select an alternative definition of SDQFLAGS, the user should modify the `_rawtag` or `_rawaccum` header and reprocess the file with `calcos`.

STATFLAG reports the following statistics:

- `NGOODPIX`: The number of good pixels or collapsed spectral columns. For the `counts` and `flt` images, this is the number of pixels in the spectral extraction or imaging region. For the `x1d` file, each 'Y' column in the spectral extraction region of the `flt` file is combined to produce the one-dimensional spectrum. The DQ of each column is the logical OR of the DQ flags of the individual pixels. Only collapsed spectral columns that pass the DQ conditions indicated by SDQFLAGS are considered good for purposes of calculating statistics.
- `GOODMEAN`: The mean of the good bins in counts per bin. For the `counts` and `flt` files, a bin is an individual pixel, while for `x1d` files, a bin is a collapsed spectral column.
- `GOODMAX`: The maximum of the good bins in the same units as the mean.

3.4.15 X1DCORR: Locate and Extract 1-D Spectrum

This module extracts a one dimensional spectrum from the image of the spectrum on the detector.

- Reference files: XTRACTAB, DISPTAB

- Input files: `flt`, `counts`
- Header keywords updated: `SP_LOC_[ABC]`, `SP_OFF_[ABC]`, `SP_NOM_[ABC]`, `SP_SLP_[ABC]`, `SP_HGT_[ABC]`
- Creates `x1d` files

A 1-D spectrum and its error array are extracted from the `flt` and `counts` images by summing the counts in the cross-dispersion direction within a band centered on the spectrum. The data are not resampled in the dispersion direction. Wavelengths are assigned by evaluating a polynomial function (dispersion relation) of the pixel coordinates. Background is subtracted (see `BACKCORR`) to get the net count rate, and the absolute flux is computed from the net count rate (see `FLUXCORR`).

This section provides the details of the spectral extraction process and the construction of the arrays which populate the `x1d` files. [Table 3.1](#) lists these arrays along with others that are used to calculate them. The following discusses how each array is calculated in detail. The summed `_x1dsum[n]` files are described in [Section 3.4.19](#).

Table 3.1: Variables used in 1-D Spectral Extraction

Variable	Description
<code>e[i]</code>	Effective count rate, extracted from <code>_flt</code>
<code>GROSS[i]</code>	Gross count rate, extracted from <code>_counts</code>
<code>BACK-GROUND[i]</code>	Smoothed background count rate, extracted from <code>_counts</code>
<code>eps[i]</code>	<code>e[i] / GROSS[i]</code>
<code>NET[i]</code>	Net count rate = <code>eps[i] (GROSS[i] - BACKGROUND[i])</code>
<code>ERROR[i]</code>	Error estimate for net count rate
<code>FLUX[i]</code>	Calibrated flux
<code>WAVE-LENGTH[i]</code>	Wavelength scale in Angstroms.
<code>DQ_WGT[i]</code>	Weights array
<code>DQ</code>	Bitwise OR of the DQ in the extraction region
<code>snr_ff</code>	The value of keyword <code>SNR_FF</code> from the flat-field reference image
<code>extr_height</code>	The number of pixels in the cross-dispersion direction that are added together for each pixel of the spectrum
<code>bkg_extr_heigh</code>	The number of pixels in the cross-dispersion direction in each of the two background regions
<code>bkg_smooth</code>	The number of pixels in the dispersion direction for boxcar smoothing the background data
<code>bkg_norm</code>	Float (<code>extr_height</code>) / (2.0 float (<code>bkg_extr_heigh</code>))

Table 3.1 defines the variables used in the 1-D spectral extraction. Variables beginning with a capital letter are saved in the output `x1d` file. An “[i]” represents array element `i` in the dispersion direction.

SEGMENT: A string array listing the segments/stripes, contained in the file.

NELEM: An array listing the number of elements in the extracted arrays.

EXPTIME: An array of the exposure times used for each segment (these can differ for FUV data).

GROSS: The GROSS count rate spectrum is obtained from the `counts` file. The extraction is performed over a parallelogram, whose shorter edges are parallel to the image boundaries and longer (slightly sloping) edges are parallel to the spectrum (see, [Figure 3.10](#)). The columns within the parallelogram are summed in the cross-dispersion direction to obtain each element of the GROSS spectrum. Note that in some cases the spectral lines are obviously tilted, i.e., not aligned with the columns, but the sum ignores any such tilt. The location of the parallelogram in the cross-dispersion direction is taken from column `B_SPEC` in the `XTRACTAB`.

Extraction routines of this sort are referred to as “boxcar” extractions because they do not weight the elements of the spectrum in the cross-dispersion direction.

GCOUNTS: This is simply the number of gross counts, or `GROSS` times `EXPTIME`.

BACKGROUND: Two background regions are sampled on the `counts` array to obtain a mean background count rate spectrum. For the FUV data, these are above and below the spectrum, and for the NUV data they are above stripe C and below stripe A (see, [Figure 3.10](#) and [Figure 3.11](#)). The background regions are extracted in the same way as the spectrum. The values in the two background regions are added, boxcar smoothed in the dispersion direction, and scaled by the sizes of their extraction regions before being subtracted from the science spectrum. Details of the background extractions are given in [Section 3.4.16](#).

NET: The NET spectrum is the difference between the GROSS spectrum and a properly scaled BACKGROUND spectrum multiplied by an array which accounts for flat-field and dead-time effects. This array is $\text{eps}[i] = e[i]/\text{GROSS}[i]$, where $e[i]$ is an element in an array extracted from the `_flt` file in exactly the same way as the GROSS spectrum is extracted from the `counts` file. Consequently, this factor corrects the NET spectrum for flat-field and dead-time effects.

DQ: The DQ array in the `x1d` file is the bitwise OR of the members of the DQ array, contained in the third FITS extension of the `counts` file, for all of the points in the `counts` image that contribute to an element of the GROSS spectrum. Consequently, if anything is flagged within the extraction region, it is reflected in the `x1d` DQ array.

DQ_WGT: The `DQ_WGT` array has one point for each extracted point in the spectrum. It is 0 or 1 depending on whether the DQ for a given point is allowed according to the header keyword, `SDQFLAGS`. The `SDQFLAG` value depends on the configuration of the instrument. For FUV/G130M and G160M the `SDQFLAG` default value is 8376. For FUV/G140L the value is 8380, and for all NUV modes, 184. These `SDQFLAG` values set the `DQ_WGTS` to 0 for events that are near the edge of the detector, dead spots, hot spots or outside of the subarray (see, [Table 2.19](#)). Otherwise, `DQ_WGTS` = 1. The `DQ_WGTS` array is used to construct the `x1dsum` file discussed in [Section 3.4.19](#).

ERROR: The ERROR array is calculated from a combination of variables needed to track the detected counts and the different scale factors which multiply them. The raw

ERROR array involves elements from both the `_flt` and the `_counts` files. It is calculated as follows:

$$\text{term1}[i] = (\text{NET}[i] * \text{exptime} / (\text{extr_height} * \text{snr_ff}))^2$$

$$\text{term2}[i] = \text{eps}[i]^2 * \text{exptime} * (\text{GROSS}[i] + \text{BACKGROUND}[i] * (\text{bkg_norm} / \text{bkg_smooth}))$$

$$\text{ERROR}[i] = (1 + \sqrt{\text{term1}[i] + \text{term2}[i] + 0.75}) / \text{exptime}$$

The ERROR array contained in the `_x1d` file differs from this one only in the sense that it has the absolute flux calibration applied (see [Section 3.4.17](#)).

WAVELENGTH: As part of the spectral extraction, `calcos` assigns wavelengths to pixels in the extracted spectra using dispersion coefficients from the reference table `DISPTAB`. For each grating, central wavelength, and aperture, the `DISPTAB` table contains the dispersion solution with respect to the template spectral lamp table that was used in the `WAVECORR` step. The dispersion solution has the following form:

$$\text{WAVELENGTH}[i] = A_0 + A_1 x[i] + A_2 x[i]^2 + A_3 x[i]^3$$

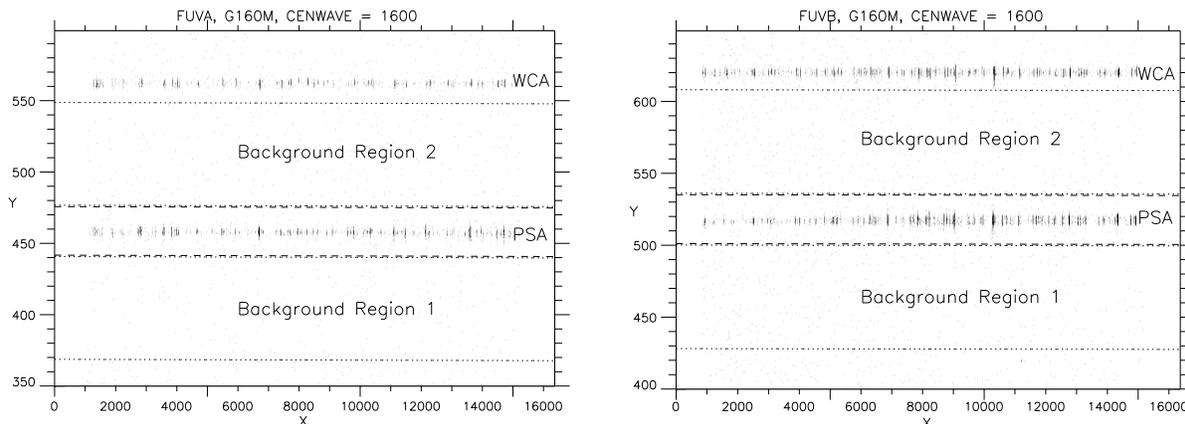
where `WAVELENGTH[i]` is the wavelength in Angstroms, `x[i]` is the pixel coordinate in the dispersion direction, and A_i are the dispersion coefficients. Offsets due to an OSM shift determined from `WAVECORR` ([Section 3.4.12](#)) are corrected for by applying a linear offset after all other corrections have been made. For all modes, small residual offsets occur because of thermal drifts and drifts in the positioning of the OSM. The precision of the OSM positioning and the impact of small offsets in the spectra are discussed further in [Chapter 4: COS Error Sources](#)

FLUX: The FLUX array in the `x1d` file is the NET spectrum corrected by the appropriate, time dependent sensitivity curve. The details of this process are discussed in [Section 3.4.17](#).

The SP_KEYWORDS: This set of keywords provides useful information on the location of the spectrum in the cross dispersion direction and the location where the spectrum is extracted. The actual location of the spectrum is found from the `_flt` file through a two step process. First, the image of the active area is collapsed along the extraction window to produce a mean cross-dispersion profile. Second, a quadratic is fit to full-width-half-maximum pixels (but minimum of 5) point region centered on the maximum of the profile. The difference between this value and the expected location, `SP_NOM_A[B]`, is given as `SP_OFF_A[B]`.

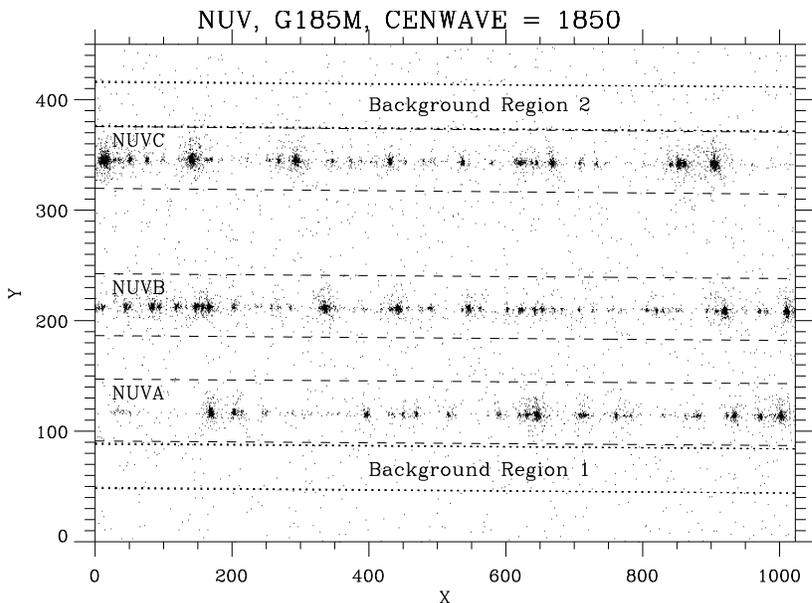
The actual location where the spectrum is extracted is given by `SP_LOC_A[B]`. For normal pipeline extractions, `SP_LOC_A[B] = SP_NOM_A[B]`, and `SP_OFF_A[B]` is listed for informational purposes only. However, it is possible to override these values and extract a spectrum at the `SP_OFF` location or any other by using the stand alone version of `x1dcorr` discussed in [Section 5.1.1](#)

Figure 3.10: FUV Extraction Regions



Portions of the undistorted images of the FUV detector segments (compare to Figure 2.2) illustrating the regions used to extract the spectrum and define the background. The dashed lines indicate the spectral extraction window, and the dotted lines define the background extraction region.

Figure 3.11: NUV Extraction Regions



Portion of the NUV detector showing spectral extraction regions used for the three non-contiguous PSA spectra (dashed lines) and for the background (dotted lines).

3.4.16 BACKCORR: 1D Spectral Background Subtraction

This module determines the background contribution to the extracted spectrum and subtracts it.

- Reference file: XTRACTAB
- Input files: flt, counts

- Header keywords updated: none
- Updates the `x1d` file

The BACKCORR module computes the number of counts in the background regions, scales them by the ratio of sizes of the spectral extraction region to background regions, and then subtracts that value from the extracted spectrum at each wavelength. There are two background regions defined. For FUV data, there is one above and one below the object spectrum (see [Figure 3.10](#)). For the NUV spectra, the two regions are above and below the three stripes (see [Figure 3.11](#)). Each background region is a parallelogram with the same slope used to define the object extraction region, but with different y-intercepts. The parameters of the background extraction region in the FUV are:

- HEIGHT: The full height (along the cross-dispersion) of the object extraction region in pixels
- BHEIGHT: The full height (along the cross-dispersion) of the background region in pixels
- BWIDTH: The full width (along the dispersion) of the box-car average performed on the background.
- B_BKG1: y-intercept of first background region
- B_BKG2: y-intercept of second background region

The centers of background regions 1 and 2 in the cross-dispersion (Y) direction follow a linear function in the dispersion (X) direction according to the function

$$Y = mX + b$$

where m is the slope of the background (keyword SLOPE), and b is the Y-intercept of the background region (B_BKG1 or B_BKG2). At the i -th pixel along the dispersion direction (X) the background is computed by first summing all of the counts in pixels in the cross-dispersion within \pm (BHEIGHT/2) of the central Y pixel of the background box. All DQ spatially related DQ flags are ignored (note that counts which occur during bad time intervals or which have out of bounds PHAs, never make it to the `counts` file). Once this is done for all X pixels, the result is averaged over \pm BWIDTH/2 pixels along the dispersion direction. This gives a local average background (with known anomalous pixels such as dead spots or strong hot spots excluded). The background regions below and above the object spectrum are both computed in this way, and then they are summed and divided by 2 to yield an average background rate. This average is then scaled to the number of pixels in the object extraction box by multiplying it by the factor “HEIGHT / BHEIGHT”. The result is the background count rate BK[i] in [Table 3.1](#), which is written to the BACKGROUND column in the `x1d` file. The background-subtracted count rate (corrected for flat field and dead time) is written to the NET column in the `x1d` table.

3.4.17 FLUXCORR/TDSCORR: Conversion to Flux

This module converts extracted spectrum into physical units, and allows for time dependencies in the conversion.

- Reference files: FLUXTAB, TDSTAB
- Input file: x1d
- Header keywords updated: none

If FLUXCORR=PERFORM, FLUXCORR divides the NET and ERROR columns by the appropriate sensitivity curve read from the FLUXTAB reference table, which converts them to flux units ($\text{erg cm}^{-2} \text{s}^{-1} \text{\AA}^{-1}$). The NET divided by the sensitivity is written to the FLUX column of the x1d file, while the ERROR column is modified in-place.

The sensitivity curves read from the reference files are interpolated to the observed wavelengths before division. The flux calibration is only appropriate for point sources and has an option to accommodate time-dependent sensitivity corrections.

If TDSCORR=PERFORM, then the module TDSCORR will correct for temporal changes in the instrumental sensitivity relative to the reference time given by REF_TIME keyword in the FITS header of TDSTAB. TDSTAB provides the slopes and intercepts needed to construct a *relative* sensitivity curve. The curve for the epoch of the observation is determined by piecewise linear interpolation in time using the slopes and intercepts closest to the time of the observation. The sensitivity may be discontinuous at an endpoint of a time interval. Different piecewise linear functions may be specified for each of the wavelengths listed in the table. This process results in a relative sensitivity at the epoch of the observation, at the wavelengths given in the reference table. Interpolation between these wavelengths to the observed wavelength array is also accomplished by piecewise linear interpolation.

3.4.18 HELCORR: Correction to Heliocentric Reference Frame

This module converts the observed wavelengths to Heliocentric wavelengths.

- Reference file: none
- Input files: rawtag, x1d
- Header keywords updated: V_HELIO

In addition to the Doppler smearing from *HST* orbital motion, the photons acquired during an observation are also Doppler shifted due to the orbital motion of the Earth around the Sun ($V \sim 29.8 \text{ km/s}$). The sign and magnitude of the Doppler shift depend on the time of the observation as well as the coordinates of the target (i.e., the position of the target relative to the orbital plane of the Earth).

The HELCORR module in **calcos** transforms wavelengths of a spectrum to the heliocentric reference frame. It is applied to the extracted 1D spectrum during the operation of X1DCORR, by utilizing the keyword V_HELIO, which is the contribution of the Earth's velocity around the Sun to the radial velocity of the target

(increasing distance is positive), in km/s. It is computed by **calcos** and written to the science data header of the output `corrtag` file before spectral extraction is performed.

The shift at each wavelength is

$$\lambda_{\text{Helio}} = \lambda_{\text{Obs}} [1 - (V_{\text{Helio}}/c)]$$

where λ_{Helio} is the corrected wavelength (Å), c is the speed of light in km/s and λ_{Obs} is the wavelength before the Heliocentric correction.

The velocity vector of the Earth is computed in the J2000 equatorial coordinate system, using derivatives of low precision formulae for the Sun's coordinates in the *Astronomical Almanac*. The algorithm does not include Earth-Moon motion, Sun-barycenter motion, or the Earth-Sun light-time correction.

3.4.19 Finalization (making the `sum` files)

The final data products for spectroscopic and NUV imaging are different and are discussed separately.

SPECTROSCOPY: Once the processing is complete, an `x1d` file is written for each spectroscopic exposure. This file includes spectra from both segments A and B for the FUV detector, and from all three stripes for the NUV detector (see, [Section 2.4.3](#)). In addition, one or more `x1dsum` files are created. This is done even if only one spectrum was obtained.

The `x1dsum` files differ from the `x1d` files in one important respect. When an `x1dsum` file is created the `DQ_WGT` array ([Section 3.4.19](#)) is used to determine whether a point is good or bad. When only a single file contributes to the `x1dsum` file, if `DQ_WGT = 0` for a pixel, then the counts, net and flux arrays for that point are set to zero. If the `x1dsum` or `x1dsum<n>` (for FPPOS observations) includes several `x1d` files, then, for each point in the spectrum, only those files with a `DQ_WGT = 1` at that point are included (weighted by the individual exposure times), and the `DQ_WGT` array in the `x1dsum` file is updated to reflect the number of individual spectra which contributed to the point. If the updated value of `DQ_WGT` for a particular point is 0, then the value of the spectrum at that point is set to 0 in the `x1dsum` file.

NUV IMAGING: The end product for NUV imaging is an `fltsum` file. Like the `x1dsum` files, an `fltsum` file is created even if only one exposure is processed. However, since no shifting is performed for imaging observations (see, [Figure 3.5](#)), the `fltsum` file is a simple exposure time weighted mean of the individual `flt` files (and it is identical to the `flt` file if only one exposure contributed to it). The DQ flags image of the `fltsum` file and, for that matter, all of the individual `flt` images, are identical. This is because the only data which make it into an `flt` or `counts` image are free of temporal or event flags (see, [Section 2.7](#)). Consequently, in the absence of shifting, all of their spatial flags should be identical.

3.5 Descriptions of Imaging Calibration Steps

The processing of image data is depicted in [Figure 3.5](#). It is an abbreviated version of the pipeline that only involves those steps which identify bad data and linearize the initial counts. No absolute flux calibration is performed and no background is identified or subtracted.

The final data products for image data are the `flt` and `fltsun` files described in [Section 2.4.2](#). Although the data are not calibrated, a crude calibration of imaging data can be performed using the total count rate from the `_flt` file and one of the two keywords provided in the count rate extension header. The header keyword `PHOTFLAM` is appropriate for a source spectrum that is flat and featureless across the MAMA detector band when measured in units of power/area/wavelength and `PHOTFNU` is appropriate for a source that is flat in power/area/Hz units. The values provided for the `PHOTFLAM` and `PHOTFNU` keywords depend on the specific combination of mirrors and apertures used in the observation.

3.6 Customizing COS Data Calibration

Sometimes the pipeline calibration performed shortly after the data were obtained from the telescope is not the best possible calibration for your science program. There are a number of reasons why it may be desirable to recalibrate your data. The most likely reasons include:

- More appropriate reference files have become available since the data were originally obtained.
- Some steps need to be repeated with different input parameters. For example, you may wish to re-perform the 1-D spectral extraction with a smaller height in order to minimize the background, or you may wish to cut a `TIME-TAG` exposure into sub-exposures, in order to study time variability.

The simplest way to recalibrate your data with the most appropriate reference files is to request the data from the archive again. However, to tailor the calibration to your individual preferences, it may be beneficial to run `calcos` yourself on your local machine, or to use tasks that improve the reference files or allow customized treatment of the data. `Calcos` can be imported and executed when running `PyRAF` or `Python`.



Be sure you are using the latest versions of the `calcos` and `STSDAS` software, `COS` calibration files, and raw data files (which list the latest reference files in their headers). `STSDAS` release information can be found at www.stsci.edu/resources/software_hardware/stsdas.

Calcos contains provisions for recalibrating raw data. Users can specify the pipeline processing steps to be performed and select the associated reference files. However, **calcos** was not designed to run its various modules independently, i.e. it is not re-entrant. The pipeline flow is modified by setting calibration switches or reference file names and then rerunning the entire pipeline. The calibration switches in the headers of the calibrated data files will reflect the operations performed on the calibrated data and the reference files used.

3.6.1 Mechanics of Tailored Recalibration

If you chose to recalibrate your COS data on your local machine, there is a certain amount of set up required for **calcos** to run properly. The operations mentioned in the checklist below will be described in detail in the following subsections:

- Set up a directory with the required reference files.
- Determine which reference files are needed and retrieve them from the Archive.
- Set the environment variable `lref` to point to your reference file directory. *Note:* you must do this before starting a **PyRAF** session!
- Update the input data file headers (including reference file names). In **IRAF**, this would be done using **thedit**.
- Update the input association files if changing files to be included.
- Set the calibration switches in the headers of the raw data files to perform the needed steps. The default calibration switches are listed in table 2.16.
- Run **calcos**.

Set up the Directory Structure for Running **calcos**

Before running **calcos**, you will need to define an environment variable to indicate the location of the directory containing the needed calibration reference files. The names of the calibration files are preceded with the logical path name “`lref$`” in the COS science headers. Ordinarily you would define this directory in a **PyRAF** session to be, for example, “`/data/vega3/cos/lref/`” using the **set** command:

```
cl> set lref "/data/vega3/cos/lref/" # Won't work!
```

Note the trailing slash (/). However, **calcos** and all of its modules are actually foreign tasks and as such do not access **PyRAF** environment variables. Therefore, *before invoking the cl*, you will need to define an environment variable from the host command line (see below) that is appropriate to your host machine. For Unix/Linux/Mac systems, the appropriate command for the example above is:

```
% setenv lref /data/vega3/cos/cal_ref/
```

Note that an alternative to using the `lref$` variable is specifying the full pathnames to the reference files in the science headers.



When running calcos or any of its modules, you must define environment variables (such as `lref$`) before starting the `cl`. It is not possible to define them within IRAF using the `set` command, nor is it possible to define them with an escape to the host level, such as:

```
!setenv lref /data/vega3/cos/lref/
```

Retrieve Reference Files

To recalibrate your data, you will need to retrieve the reference files used by the different calibration steps to be performed. The names of the reference files to be used during calibration must be specified in the primary header of the input files, under the section “CALIBRATION REFERENCE FILES.” Note that the data headers will be populated already with the names of the reference files used during pipeline calibration at STScI.

Chapter 1 of the *Introduction to HST Data Handbooks* describes how to retrieve data and reference files via the World Wide Web. To retrieve the best reference files via MAST, check “Best Reference Files” in the “Reference Files” section of the Retrieval Options form.

The COS reference files are all in FITS format, and can be in either IMAGE or BINTABLE extensions. The names of these files along with descriptions of their contents are given in [Section 3.7](#). The rootname of a reference file is based on the time that the file was delivered to the Calibration Data Base System (CDBS).

Edit the Calibration Header Keywords

To edit file headers in preparation for recalibration, use the **STSDAS** task `thedit`. The `thedit` task takes several input parameters: the name(s) of the raw data files to be edited, the header field to be edited, and the new value of the header field. It can be used to change the values of any calibration switches, reference files or tables to the values you wish to use for recalibrating your data. To edit the calibration keyword values:

1. Run the `thedit` task, specifying a list of files in which you want to change calibration keyword values. You may specify more than one file (using wildcards, for example) to be updated. For example, you could change the flat reference file to be used for all COS raw science files in the current directory using the command:

```
ct> thedit *raw*.fits[0] flatfile 'lref$n9n201821_flat.fits' up+
```

Similarly, to turn off the FUV burst calibration switch use the command:

```
ct> thedit *raw*.fits[0] brstcorr 'OMIT' up+
```



*If you are changing keywords that reside in the FITS primary header unit with **hedit** or **thedit**, be sure to explicitly specify the primary header by appending “[0]” to the FITS file name.*

Edit the Input Association File

Users may find it necessary to edit the input association file for **calcos**. Reasons for editing an association file might include the use of a different wavecal or to remove a compromised exposure from an association. One way to update an association file is to use the **STSDAS** task, **tedit**. For example, use the **PyRAF** task **tprint** to first look at the contents of association table, `l9v221010_asn.fits`.

```
-->tprint l9v221010_asn.fits prparam=no prdata=yes
# MEMNAME          MEMTYPE          MEMPRSNT
  L9V221EUQ        EXP-FP           yes
  L9V221EWQ        EXP-AWAVE        yes
  L9V221EYQ        EXP-FP           yes
  L9V221F0Q        EXP-AWAVE        yes
  L9V221F2Q        EXP-FP           yes
  L9V221F4Q        EXP-AWAVE        yes
  L9V221F6Q        EXP-FP           yes
  L9V221F8Q        EXP-AWAVE        yes
  L9V221010        PROD-FP          yes
```

To quickly see basic exposure information for each exposure listed in the association use the **thselect** command:

```
--> thselect 19v221*raw*fits[0] \
filename,detector,aperture,opt_elem,cenwave,exptype,obsmode,fppos 'yes'
```

19v221euq_rawtag_a.fits	FUV	PSA	G130M	1309	EXTERNAL/SCI	TIME-TAG	1
19v221euq_rawtag_b.fits	FUV	PSA	G130M	1309	EXTERNAL/SCI	TIME-TAG	1
19v221ewq_rawtag_a.fits	FUV	WCA	G130M	1309	WAVECAL	TIME-TAG	1
19v221ewq_rawtag_b.fits	FUV	WCA	G130M	1309	WAVECAL	TIME-TAG	1
19v221eyq_rawtag_a.fits	FUV	PSA	G130M	1309	EXTERNAL/SCI	TIME-TAG	2
19v221eyq_rawtag_b.fits	FUV	PSA	G130M	1309	EXTERNAL/SCI	TIME-TAG	2
19v221f0q_rawtag_a.fits	FUV	WCA	G130M	1309	WAVECAL	TIME-TAG	2
19v221f0q_rawtag_b.fits	FUV	WCA	G130M	1309	WAVECAL	TIME-TAG	2
19v221f2q_rawtag_a.fits	FUV	PSA	G130M	1309	EXTERNAL/SCI	TIME-TAG	3
19v221f2q_rawtag_b.fits	FUV	PSA	G130M	1309	EXTERNAL/SCI	TIME-TAG	3
19v221f4q_rawtag_a.fits	FUV	WCA	G130M	1309	WAVECAL	TIME-TAG	3
19v221f4q_rawtag_b.fits	FUV	WCA	G130M	1309	WAVECAL	TIME-TAG	3
19v221f6q_rawtag_a.fits	FUV	PSA	G130M	1309	EXTERNAL/SCI	TIME-TAG	4
19v221f6q_rawtag_b.fits	FUV	PSA	G130M	1309	EXTERNAL/SCI	TIME-TAG	4

To remove a member from association, `l9v221010_asn.fits`, and to use a different wavecal file, use the following commands in **PyRAF**.

```
cl> tedit l9v221010_asn.fits
```

#	row	MEMNAME	MEMTYPE	MEMPRSNT
	1	L9V221EUQ	EXP-FP	yes
	2	L9V221EWQ	EXP-AWAVE	yes
	3	L9V221EYQ	EXP-FP	yes
	4	L9V221F0Q	EXP-AWAVE	yes
	5	L9V221F2Q	EXP-FP	yes
	6	L9V221F4Q	EXP-AWAVE	yes
	7	L9V221F6Q	EXP-FP	yes
	8	L9V221F8Q	EXP-AWAVE	yes
	9	L9V221010	PROD-FP	yes

Change the MEMNAME of row 8 to L9V221FSQ

Change the MEMPRSNT of rows 1 and 2 to no

```
cl> tprint l9v221010_asn.fits
```

#	MEMNAME	MEMTYPE	MEMPRSNT
	L9V221EUQ	EXP-FP	no
	L9V221EWQ	EXP-AWAVE	no
	L9V221EYQ	EXP-FP	yes
	L9V221F0Q	EXP-AWAVE	yes
	L9V221F2Q	EXP-FP	yes
	L9V221F4Q	EXP-AWAVE	yes
	L9V221F6Q	EXP-FP	yes
	L9V221FSQ	EXP-AWAVE	yes
	L9V221010	PROD-FP	yes

Finally, reprocess the data by running **calcos** on the updated association file. See the following section for syntaxes of running **calcos**.

Run Calcos

Users may choose any of three ways to run **calcos** using Python, **PyRAF**, or from the Unix/Linux/Mac command line. The input arguments and examples for each case are as follows:

1. To run **calcos** in **Pyraf** from the **hstcos** package:

```
--> stsdas.hst_calib.hstcos
--> calcos filename_asn.fits outdir=new
```

Table 3.2: Arguments for Running calcos in PyRAF

Argument	Values	Default	Description
input	filename		Association table (<i>asn</i>) or individual raw file (<i>rawtag</i> , <i>rawaccum</i>) or <i>corrtag</i> file to be processed
verbosity	0-2	1	Verbose output from calcos
savetmp	yes or no	no	Save temporary files: <i>x1d_a</i> , <i>x1d_b</i> , <i>lampflash_a</i> , and <i>lampflash_b</i>
outdir	directory name		The name of the output directory. If blank, current directory.
find_target	yes or no	no	Have calcos find the spectrum location and centre the extraction box on that location
cutoff	-1, 0, positive floating point	-1	If specified, this value will be used as cutoff of standard deviation for <i>find_target</i>
shift_file	filename		File containing wavecal shifts (will override shifts calculated by calcos)
stimfile	filename		If specified, the stim positions will be written to (or appended to) this text file
livefile	filename		If specified, the livetime factors will be written to (or appended to) this text file
burstfile	filename		If specified, burst information will be written to (or appended to) this text file

2. To run **calcos** in Python:

```
>>> import calcos
>>> calcos.calcos('filename_asn.fits', verbosity=2, \
outdir="new")
```

Table 3.3: Arguments for Running calcos in Python:

Argument	Values	Default	Description
asntable	'filename'	' '	Association table (asn) or individual raw file (rawtag, rawaccum) to be processed
outdir	directory name	None	The name of the output directory
verbosity	0, 1, 2	1	0=quiet, 1=verbose, 2=very verbose
find_target	True or False	False	Have calcos find the spectrum location and centre the extraction box on that location
shift_file	'filename'		File containing wavecal shifts (will override shifts calculated by calcos)
save_temp_files	True or False	False	Save temporary files: x1d_a, x1d_b, lampflash_a, and lampflash_b
stimfile	"filename"		If specified, the stim positions will be written to (or appended to) this text file
livetimefile	"filename"		If specified, the livetime factors will be written to (or appended to) this text file
burstfile	"filename"		If specified, burst information will be written to (or appended to) this text file

3. To run **calcos** from the Unix/Linux/Mac command line:

```
% calcos -o new --stim stim.txt filename_asn.fits
```

Table 3.4: Command-line Options for Running calcos in Unix/Linux/Mac:

Option	Description
--version	print the version number and exit
-r	print the full version string and exit
-q	Quiet
-v	Very verbose
-s	save temporary file
-o outdir	Output directory
--find yes	Have calcos find Y location of spectrum
--find no	Extract spectrum at default location
--find cutoff	Find Y location if sigma <= cutoff
--shift filename	File to specify shift values
--stim filename	Append stim locations to filename
--live filename	Append livetime factors to filename
--burst filename	Append burst information to filename

To redirect the **calcos** STDOUT to a file use the following command:

```
% calcos -v -o new filename_asn.fits > log.txt
```

While users are recommended to run **calcos** on association files, it is possible to run **calcos** with a single raw or corrtag file as the input. In this mode, **calcos** will always automatically process both segment files for FUV data if they both exist. For example if `rootname_rawtag_a.fits` is the input for **calcos**, then `rootname_rawtag_b.fits` will automatically be processed. The data from both segments will be calibrated and combined to create the final product, `rootname_x1d.fits`

3.6.2 Using GO Wavecals

Through the use of associations, **calcos** also contains a provision to select wavecals other than the default for calibration of the science exposures. To use an exposure other than or in addition to the default wavecal, the user can add a row to the association table. The rootname (case insensitive) should be given in the MEMNAME column, the string EXP-GWAVE in the MEMTYPE column, and the value in the boolean MEMPRSNT column set to true (e.g. yes if you use the **IRAF** **tedit** task). Make sure that the WAVECORR keyword in the primary header of the raw science file is set to PERFORM, and then run **calcos** as normal. Note GO Wavecals can only be used with non TAGFLASH data.

3.7 Reference Files

This section contains a description of the COS reference files. See [Figure 3.1](#) - [Figure 3.5](#) for which modules use these files and [Section 3.4](#) for explanations of how their contents are applied by those modules.

3.7.1 BRSTTAB: Burst Parameters Table

- File Suffix: `_burst`

The BRSTTAB file provides the parameters needed to identify bursts. It consists of a primary header extension and a binary table extension with the columns listed in [Table 3.5](#). Details of the burst rejection routine are given in [Section 3.4.2](#).

Table 3.5: BRSTTAB Table Contents

Column Name	Data Type	Description
SEGMENT	String	Segment name, FUVA or FUVB
MEDIAN_N	Double	Factor above the median count rate for a time interval to be identified as a burst
DELTA_T	Double	Normal sampling time for large burst detection (s)
DELTA_T_HIGH	Double	High count rate sampling time for large burst detection (s)
MEDIAN_DT	Double	Time interval used to search for localized bursts (s)
BURST_MIN	Double	Minimum threshold rate for small bursts (counts/s)
STDREJ	Double	Number of standard deviations above background noise for small bursts
SOURCE_FRAC	Double	Minimum factor small bursts must be above source counts.
MAX_ITER	Long	The maximum number of iterations used to re-evaluate the median to detect a localized burst
HIGH_RATE	Double	Total count rate threshold to use DELTA_T_HIGH instead of DELTA_T (counts/s)

3.7.2 BADTTAB: Bad Time Interval Table

- File Suffix: `_badt`

The BADTTAB reference file lists the start and end times of known bad time intervals. It is used by the BADTCORR calibration module to flag events in TIME-TAG events lists which occur during a bad time interval. In later processing the flagged events will be removed from the final calibrated data, and the exposure time header keyword, EXPTIME, updated. The bad time interval table consists of segment, start, and end columns (see, [Table 3.6](#)). The segments columns can be populated with either FUVA, FUVB or ANY. The start and end columns are in Modified Julian Date.

Table 3.6: BADTTAB Table Content

Column Name	Data Type	Description
SEGMENT	String	Detector segment, FUVA, FUVB or ANY
START	Double	Bad time interval start time in MJD
END	Double	Bad time interval end time in MJD

3.7.3 PHATAB: Pulse Height Discrimination Table

- File Suffix: `_pha`

The PHATAB reference file is only valid for FUV data, and is applied during the PHACORR step of `calcos` to filter non-photon events. The file consists of two extensions, the first being the primary header, and the second a binary table (see [Table](#)

3.7). The table lists the lower and upper thresholds for valid individual pulse heights in TIME-TAG mode. In TIME-TAG mode, each detector event has an associated pulse-height of 5 bits with values ranging from 0 to 31, The table also gives the minimum and maximum values for the location of the mean value of the pulse height distribution used in ACCUM mode. In ACCUM mode, a pulse height distribution histogram is generated for the whole exposure and downloaded as part of the science data file. The histogram includes all the digitized events for each segment independently of the currently defined subarrays. Note in ACCUM mode the pulse height is a 7 bit number with values ranging from 0 to 127.

Table 3.7: PHATAB Table Contents

Column Name	Data Type	Description
SEGMENT	String	Segment name, FUV A or FUV B
LLT	Long	Lower limit threshold (TIME-TAG)
ULT	Long	Upper limit threshold (TIME-TAG)
MIN_PEAK	Float	Lower limit for location of mean (ACCUM)
MAX_PEAK	Float	Upper limit for location of mean (ACCUM)

3.7.4 PHAFILE: Pulse Height Discrimination File

- File Suffix: `_phf`

This file is only used for FUV data, and is a 2D equivalent to the PHATAB. The PHAFILE is used by the PHACORR calibration module to filter non-photon events. If both a PHATAB and PHAFILE are available, the PHAFILE will be used.

Each pulse height discrimination reference file contains four IMAGE extensions. There are two for each segment, containing the lower and upper PHA limits for each pixel. At a given (X,Y) location in the uncorrected COS data, the value at that location gives the lowest and highest (respectively) pulse height that will be treated as a valid photon event at that detector location.

3.7.5 BRFTAB: Baseline Reference Frame Table

- File Suffix: `_brf`

The BRFTAB reference file is only applicable to FUV data and is used during pipeline processing in the TEMPCORR module to apply the thermal distortion correction. The FUV detector does not have physical pixels like a CCD. Instead, the x and y positions of detected photon events are obtained from analog electronics, which are susceptible to thermal changes. Electronic stim pulses are normally commanded during integration and are used as physical position reference points. To return the FUV data to a known physical space, the BRFTAB defines the stim positions.

The BRFTAB file consists of a primary header extension and a binary table extension. The table lists the stim locations, stim search regions, and the active detector areas (Table 3.8).

Table 3.8: BRFTAB Table Contents

Column Name	Data Type	Description
SEGMENT	String	Segment name, FUVA or FUVB
SX1	Double	X pixel coordinate (zero indexed) of stim1 ¹
SY2	Double	Y pixel coordinate (zero indexed) of stim1
SX2	Double	X pixel coordinate (zero indexed) of stim2 ²
SY2	Double	Y pixel coordinate (zero indexed) of stim2
XWidth	Long	Half width of search region for stims
YWidth	Long	Half height of search region for stims
A_Left	Long	X pixel of left side of active region
A_Right	Long	X pixel of right side of active region
A_Low	Long	Y pixel of lower side of active region
A_High	Long	Y pixel of upper side of active region

1. Stim 1 is located in the upper left corner
2. Stim 2 is located in the lower right corner

3.7.6 WALKTAB: Y Walk Correction Table

- File Suffix: `_walk`

The WALKTAB reference file is only applicable to FUV data and is used during pipeline processing in the WALKCORR module to correct the effects of Y walk. The COS FUV XDL detector is subject to gain sag, where as physical locations on the detector accumulate photon events, the pulse height of the electron cloud generated by the event becomes smaller, and the Y co-ordinates of the event are mis-registered towards the bottom of the detector (i.e. a decrease in apparent Y coordinate). These effects are time-variable, and depend on event pulse height.

The current correction employed is a simple linear correction to registered Y location based on event pulse height, but the WALKTAB has the ability to correct both X and Y location based on arbitrary polynomials taking into account X location, Y location, and pulse height.

The WALKTAB file consists of a primary header extension and a binary table extension. The table lists the coefficients of the polynomials in X and Y (Table 3.9). In order to determine how the coefficients will be used, see the WALKCORR section (Section 3.4.5).

Table 3.9: WALKTAB Table Contents

Column Name	Data Type	Description
SEGMENT	String	Segment name, FUVA or FUVB
X0	Double	Zero point in XCORR for polynomials (i.e. subtracted from XCORR before evaluating polynomials)
Y0	Double	Zero point in YCORR for polynomials
N_X	Integer	Number of terms in X
N_Y	Integer	Number of terms in Y
N_PHA_COEFF	Integer	Number of coefficients in polynomial for PHA
XCOEFF	Double Array	Array of coefficients for determining the change to XCORR
YCOEFF	Double Array	Array of coefficients for determining the change to YCORR

3.7.7 GEOFILE: Geometric Correction File

- File Suffix: `_geo`

This file is only used for FUV data. The `GEOFILE` is used by the `GEOCORR` calibration module to perform the geometric correction. From the nature and construction of the XDL detectors, the physical sizes of the pixels vary across the detector. The geometric distortion maps are used to correct for this variation and to transform the data into a constant physical pixel size early in the data reduction calibration process. After the thermal correction has been applied and the detector digital span and position are adjusted to their reference values, as defined in the reference table, the geometric correction can be applied. This implies that all the files used to determine the geometric correction were initially thermally-corrected.

Each geometric correction reference file contains four `IMAGE` extensions. There are two for each segment, and for each segment, there is one for each axis. At a given (X,Y) location in the uncorrected COS data, the value at that location (corrected for binning and offset) in the geometric correction image gives the distortion to be subtracted from the X or Y coordinates. The order of the extensions are: 1 = X coordinate for FUVA, 2 = Y coordinate for FUVA, 3 = X coordinate for FUVB and 4 = Y coordinate for FUVB.

3.7.8 DEADTAB: Deadtime Table

- File Suffix: `_dead`

The DEADTAB reference file is used in the DQICORR: Initialize Data Quality File module, to obtain the true number of events received compared to the number of events counted by the detector electronics.

There is one DEADTAB reference file for the NUV and FUV detectors. Each consists of a primary header extension and a binary table extension which contains the LIVETIME values for a given observed count rate (OBS_RATE) and segment. The livetime is defined as:

$$\text{livetime} = \text{observed rate} / \text{true rate}$$

and can be used to calculate the true count rate.

3.7.9 FLATFILE: Flat-field File

- File Suffix: `_flat`

FLATFILE provides a flat-field image which is used by the pipeline to remove the pixel-to-pixel variations in the detector. The FUV FLATFILE consists of a primary header and two 14000 x 400 IMAGE extensions, one for each segment. The NUV FLATFILE consists of a primary header and a 1024 x 1024 IMAGE extension.

Currently, the FUV flat-field reference file only corrects the effect of grid wire shadows, and it only corrects G130M and G160M data. There is no flat-field reference file to correct the data taken with G140L.

The NUV flat-field is a combination of internal and external deuterium flat field lamp exposures from thermal-vacuum testing which illuminate the portion of the detector where spectra fall. The data cover the following pixel region of the detector: x (dispersion): 0 to 1023, and y (cross-dispersion): 495 to 964. The rest of the detector, where flat field data are not available, has a value of 1.0. The bottom four and top three rows of the detector do not fit well with the rest of the detector and they are flagged in the data quality table.

3.7.10 BPIXTAB: Bad Pixel Table

- File Suffix: `_bpix`

The data quality initialization table identifies rectangular regions on the detectors that are known to be less than optimal. The feature type describes the type of detector blemish enclosed within the bounding box and dq is the quality value assigned to all events detected within the box. The regions were identified by visual inspection of the combined flat field data for each detector (and segment). The BPIXTAB files consist of a primary header and a binary table extension which consists of the columns listed in [Table 3.10](#).

Table 3.10: BPIXTAB Table Content

Column Name	Data Type	Description
SEGMENT	String	Segment name, FUVA, FUVB, or ANY for NUV
LX	Long	X coordinate of lower left corner of region
LY	Long	Y coordinate of lower left corner of region
DX	Long	Width of region in X
DY	Long	Width of region in Y
DQ	Long	Data quality value to assign to current region
TYPE	String	Comment regarding current region

In the BPIXTAB table, the DQ field may be a logical OR due to several different values, each associated with a unique issue (see [Table 2.19](#)).

3.7.11 LAMPTAB: Template Calibration Lamp Spectra Table

- File Suffix: `_lamp`

The LAMPTAB files consist of a primary header extension and a binary table extension which contains an extracted 1-D spectrum from the internal PtNe calibration lamp through the WCA aperture, for each grating, central wavelength, and FPPOS setting. It is used in the **calcos** pipeline to determine the pixel offset of the observed data. The structure of the template calibration lamp spectra table is shown in [Table 3.11](#). The stepper motor offsets range from -2 to +1 and correspond to FPPOS settings of 1 to 4.

Table 3.11: LAMPTAB Table Contents

Column Name	Data Type	Description
SEGMENT	String	Segment: FUVA, FUVB, NUVA, NUVB, NUVC
OPT_ELEM	String	Grating name
CENWAVE	Long	Central wavelength (Angstrom)
FPOFFSET	Integer	Array of stepper motor offsets
FP_PIXEL_SHIFT	Double	Offset in pixels from FPOFFSET=0
INTENSITY	Float	Wavecal spectrum array

3.7.12 WCPTAB: Wavecal Parameter Table

- File Suffix: `_wcp`

The WCPTAB file contains information relevant for the wavecal pipeline processing. It consists of primary header extension and a binary table extension which

is described in [Table 3.12](#). `XC_RANGE` is the maximum pixel offset to use when doing a cross correlation between the observed data and the template wavecal. That is, the observed spectrum should be shifted relative to the template by a number of pixels, ranging from `-XC_RANGE` to `+XC_RANGE` inclusive. `XD_RANGE` is half the search range for finding the spectrum in the cross dispersion direction. The search range is from `B_SPEC - XD_RANGE` to `B_SPEC + XD_RANGE` inclusive, where `B_SPEC` is the nominal location of the spectrum from the `XTRACTAB` table discussed below. `BOX` is the width of the boxcar filter for smoothing the cross-dispersion profile. `RESWIDTH` is the number of pixels per resolution element, and is assigned a value of 6.0 for the FUV detectors and 3.0 for the NUV detector.

When applying the offsets found from the wavecals to the science data, it may happen that there was no wavecal at the same OSM position. In this case, the wavecal that was closest in time to the science observation may be used, with a correction for the difference in OSM positions. That correction is based on `STEP_SIZE`, the number of pixels corresponding to one OSM step. There may be a check, however, to guard against using a wavecal that was taken too far away in time from the science observation. If the science observation and wavecal were taken more than `MAX_TIME_DIFF` apart, then the wavecal should not be used for that science observation.

Table 3.12: `WCPTAB` Table Contents

Column Name	Data Type	Description
<code>OPT_ELEM</code>	String	Grating name
<code>XC_RANGE</code>	Long	Maximum Lag (amplitude) for cross correlation
<code>RESWIDTH</code>	Double	Number of pixels per resolution element in the dispersion direction
<code>MAX_TIME_DIFF</code>	Double	Defines ‘close in time’ for wavecals
<code>STEP_SIZE</code>	Long	One step of OSM is this many pixels
<code>XD_RANGE</code>	Long	Amplitude of search range for finding spectrum
<code>BOX</code>	Integer	Width of boxcar smoothing filter

3.7.13 `DISPTAB`: Dispersion Coefficient Table

- File Suffix: `_disp`

There are two `DISPTAB` files with similar formats, one for the NUV, and one for the FUV. They consist of a main header and a binary table in the second HDU. These tables provide the dispersion relations for each segment, aperture, optical element, and central wavelength. Each file has the format given in [Table 3.13](#). The dispersion relation table gives a set of polynomial coefficients for computing wavelength from pixel number (see Oliveira, [COS ISR2010-05](#) and [-06](#) for details).

Each row of the table gives a set of dispersion coefficients. The row to be used is selected on SEGMENT, OPT_ELEM, CENWAVE, and APERTURE.

Table 3.13: DISPTAB Table Format

Column Name	Data Type	Description
SEGMENT	String	Segment name, FUVA, FUVB, NUVA, NUVB, NUVC
OPT_ELEM	String	Grating name
APERTURE	String	Aperture name
CENWAVE	Long	Central wavelength of setting
NELEM	Long	Number of non-zero coefficients in the polynomial
COEFF	Double[4]	Coefficients, up to 4.
D_TV03	Double	Offset from WCA to PSA in TV03 data
D	Double	

For P_X = the zero-indexed Doppler corrected pixel value in the dispersion direction, the associated wavelength for a specific segment, optical element, let

$$P_X' = P_X + (D_TV03 - D),$$

Then the corresponding wavelength in Angstroms is given by:

$$\lambda(P_X) = \text{COEFF}[0] + \text{COEFF}[1]*P_X' + \text{COEFF}[2]*P_X'^2 + \text{COEFF}[3]*P_X'^3$$

3.7.14 XTRACTAB: 1-D Spectral Extraction Table

- File Suffix: `_1dx`

There are two XTRACTAB files with similar formats, one for the NUV and one for the FUV. They consist of a main header and a binary table in the second HDU. These tables provide the information needed to extract the spectrum from a geometrically corrected image of the detector for each optical element and central wavelength. Each file has the format given in [Table 3.14](#).

Table 3.14: XTRACTAB Table Format

Column Name	Data Type	Description
SEGMENT	String	Segment name, FUVA, FUVB, NUVA, NUVB, NUVC
OPT_ELEM	String	Grating name
CENWAVE	Long	Central wavelength setting
APERTURE	String	Aperture name
SLOPE	Double	Slope of the spectral extraction box
B_SPEC	Double	Intercept of the spectrum extraction box
B_BKG1	Double	Intercept of the lower background spectral extraction box
B_BKG2	Double	Intercept of the upper background spectral extraction box
HEIGHT	Long	Height of the spectral extraction window
BHEIGHT	Long	Height of the extraction window for the background
BWIDTH	Long	Width of the boxcar filter used to smooth the backgrounds

The spectral extraction of a source is performed by collapsing the data within a parallelogram of height HEIGHT that is centered on a line whose slope and intercept are given by SLOPE and B_SPEC. Similarly, two background spectra are determined by collapsing the data within a parallelogram of height BHEIGHT centered on the lines defined by SLOPE and B_BKG1 and SLOPE and B_BKG2. The background spectra are then smoothed by a boxcar of width BWIDTH. These are then scaled and subtracted from the source spectrum.

3.7.15 FLUXTAB: Photometric Throughput Table

- File Suffix: `_phot`

There are two FLUXTAB files with similar formats, one for the NUV, and one for the FUV. They consist of a main header and a binary table in the second HDU. These tables provide the information needed to convert from corrected detector counts to flux units of $\text{erg s}^{-1}\text{cm}^{-2}\text{A}^{-1}$ for each segment, optical element, aperture, and central wavelength. Each file has the format given in [Table 3.15](#).

Table 3.15: FLUXTAB Table Format

Column Name	Data Type	Description
SEGMENT	String	Segment Name
OPT_ELEM	String	Name of optical element
CENWAVE	Long	Central wavelength of the setting
APERTURE	String	Name of the aperture
WAVELENGTH	Double	Wavelength array in Angstroms
SENSITIVITY	Float	Sensitivity array

The units of the Sensitivity array are $(\text{count s}^{-1} \text{ pixel}^{-1})/(\text{erg s}^{-1} \text{ cm}^{-2} \text{ Angstrom}^{-1})$. For each segment, optical element, central wavelength setting, and aperture, these files contain arrays of wavelengths and sensitivities which can be interpolated onto the observed wavelength grid. The net counts can then be divided by the sensitivity curves to produce flux calibrated spectra.

3.7.16 TDSTAB: Time Dependent Sensitivity Table

- File Suffix: `_t ds`

There are two such files, one for the FUV and one for the NUV. They are only used for spectroscopic data. The files contain the information necessary to determine the relative sensitivity curve at any given time by interpolating between relative sensitivity curves given at fiducial times which bracket the observation, or else extrapolate the results from the last curve if the observation date is more recent than the last fiducial date. Interpolation data are provided for each segment, optical element, and aperture (see [Table 3.16](#)).

Table 3.16: TDSTAB Table Format

Column Name	Data Type	Description
SEGMENT	String	Segment Name
OPT_ELEM	String	Name of optical element
APERTURE	String	Name of the aperture
NWL	Long	Number of wavelength points
NT	Long	Number of time points
WAVELENGTH	Double[NWL]	Wavelength array in Angstroms
TIME	Double[NT]	Fiducial times in MJD
SLOPE	Double[NWL, NT]	Percent per year
INTERCEPT	Double[NLW, NT]	Ratios of current curve to original curves

For an observation obtained at time T , which lies between $TIME[j]$ and $TIME[j+1]$, the sensitivity curve used to calibrate the spectrum will be corrected by the following factor:

$$(T - REF_TIME) SLOPE[i,j]/(365.25*100) + INTERCEPT[i,j].$$

where REF_TIME is a general reference time given in the header of the FITS extension.

3.7.17 GSAGTAB: Gain Sag Table

- File Suffix: `_gsag`

The gain sag reference table is only applicable for FUV data and it is used along with the bad pixel reference table (`_bpix`) in the DQICORR module. The table provides the locations of rectangular regions for portions of the FUV detector that have very low pulse height amplitude.

After the primary header, each extension of the GSAGTAB is a binary fits table of the gain sagged pixels on the detector at a given voltage. During the pipeline processing, these extensions are selected depending on the SEGMENT and HVLEVEL. Each row in the table gives the location and data quality value for one rectangular region. The DATE column is used to select rows. A row will be used to flag a gain sagged region if the value in the DATE column is less than or equal to the exposure start time. For a description on the columns contained in the binary tables see [Table 3.17](#).

Table 3.17: GSAGTAB Table Format

Column Name	Data Type	Description
DATE	Double	Modified Julian Date at which the PHA in a region dropped so low that the region should be flagged as gain-sagged.
LX	Long	X coordinate of lower left corner of region
LY	Long	Y coordinate of lower left corner of region
DX	Long	Width of region in X
DY	Long	Width of region in Y
DQ	Long	Data quality value assign to current region

3.7.18 HVTAB: High Voltage Table

- File Suffix: `_hv`

The high voltage reference table is only used for FUV data and it populates extension header keywords HVLEVELA and HVLEVELB. In order to optimize the

performance of the instrument, the nominal high voltage level on both FUV segments has been changed since launch. This reference table provides the dates when the FUV high voltage was adjusted and the values (raw counts) used in the command to set the high voltage. The file consists of two extensions, FUVA and FUVB, each of them a binary table listing the date and HVLEVEL. (See [Table 3.18](#)).



Nominal high voltage level on both FUV segments has been changed since launch. For a complete description consult [Appendix B.1](#).

Table 3.18: HVTAB Table Format

Column Name	Data Type	Description
DATE	Double	Modified Julian Date when the high voltage was changed
HVLEVEL#	Long	Commanded (raw) high voltage level, segment A or B

3.7.19 SPWCSTAB: Spectroscopic WCS Parameters Table

- File Suffix: `_spwcs`

The spectroscopic SPWCS table gives the parameters needed to populate the world coordinate keywords in the `_corrtag`, `_counts`, and `_flt` files. There are entries for each SEGMENT, OPT_ELEM, CENWAVE, and APERTURE. The columns (see [Table 3.19](#)) are interpreted as follows. The detector coordinate system has two dimensions. Let the more rapidly varying axis be X and the less rapidly varying axis Y. The world coordinate system has three dimensions, the spectral coordinate, right ascension, and declination. The reference pixel is at approximately the middle of the detector. CTYPE1 can be WAVE to indicate that the wavelength is a linear function of pixel number, or it can be WAVE-GRI to indicate that the wavelengths should be computed by using the grating (“grism”) equation. In either case, the wavelengths are in vacuum. CRVAL1 is the wavelength at the reference pixel. CRPIX1 is the location of the reference pixel in the first axis (X); the location of the reference pixel in the second axis (Y) is gotten separately from the 1-D Extraction Parameters Table (XTRACTAB). CDELTA1 is the dispersion in Angstroms per pixel at the reference pixel. At a single wavelength (nominally the wavelength at the reference pixel), a pixel when projected onto the sky would be approximately a rectangle. CDELTA2 and CDELTA3 are the sizes of that rectangle in the X and Y directions. SPECRES is the spectral resolution; this is only used for updating the archive search keyword of the same name. G is the groove density of the grating, e.g. 3.8E6 grooves per meter for G130M. SPORDER is the spectral order. This will usually be 1, but for G230L, stripe NUVC, SPORDER will be 2. ALPHA is the angle between the normal to the grating and the light that is incident

onto the grating. THETA is the angle between two lines from the grating to the detector, the line to the reference pixel and the line that is perpendicular to the detector. Since the reference pixel is close to the middle of the detector, THETA will probably be close to zero.

Table 3.19: SPWCSTAB Table Format

Column Name	Data Type	Description
SEGMENT	String	Segment Name
OPT_ELEM	String	Name of optical element
CENWAVE	Integer	Central wavelength (Angstroms)
APERTURE	String	NPSA, BOA, WCA
CTYPE1	String	Type of world coordinate on spectral axis
CRPIX1	Double	Reference pixel number for spectral axis (X)
CRVAL1	Double	Wavelength at the reference pixel (Ang)
CDEL1	Double	dispersion at reference pixel (Ang/pixel)
CDEL2	Double	Size of a pixel in dispersion direction (deg/pixel)
CDEL3	Double	Size of a pixel perpendicular to dispersion direction (deg/pixel)
SPECRES	Double	Spectral resolution
G	Double	Groove density of grating (grooves/m)
SPORDER	Integer	Spectral order
ALPHA	Double	Incident angle from aperture onto grating (degrees)
THETA	Double	Angle from reference pixel to base of normal from grating to detector (degrees)

COS Error Sources

In this chapter...

4.1 Overview / 120

4.2 Error Sources Associated with Pipeline Processing Steps / 120

4.3 Factors Limiting Flux and Wavelength Accuracy / 125

4.1 Overview

In this chapter we describe various properties of the COS detectors that affect the final quality of the science data products. Note that several of the effects outlined below are the subjects of ongoing study.



Always check the COS Web pages for the latest [Instrument Science Reports \(ISRs\)](#) which describe the results of various studies intended to better characterize COS.

4.2 Error Sources Associated with Pipeline Processing Steps

In this section, we discuss sources of error that are associated with major steps in the COS calibration pipeline (**calcos**). Note that these steps themselves were already described in [Chapter 3](#) and will not be repeated here; this section will only describe specific issues related to the error budget of the resulting data which were not described before.

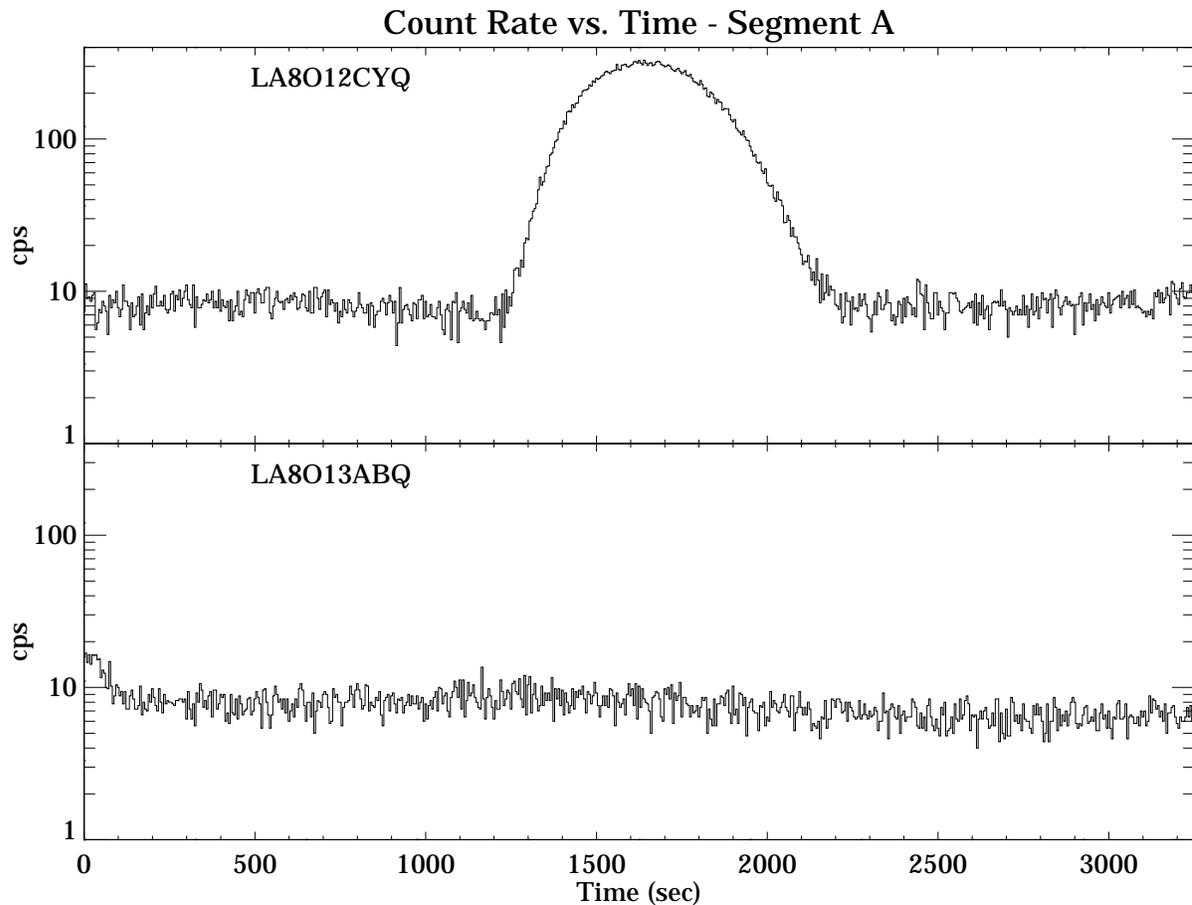
4.2.1 FUV Dark Count Rate

Dark counts arise from a combination of detector effects and external sources. **Calcos** will remove the effects of detector background (which includes dark, scattered light, etc.) in the **BACKCORR** module. This is done after the **X1DCORR** converts the detector image to a 1D extracted spectrum. Here, we discuss the instrumental contribution, since it can be the limiting factor in the error budget for very faint sources.

FUV-XDL Dark Count Rate

The FUV detector dark rates measured on the ground were very low, of order 0.4 counts $\text{cm}^{-2} \text{s}^{-1}$. Typical dark rates on-orbit away from the South Atlantic Anomaly (SAA) during the first year on orbit for both segments are about three times higher, 1.8×10^{-6} counts $\text{pixel}^{-1} \text{s}^{-1}$ (or 1.25 counts $\text{cm}^{-2} \text{s}^{-1}$). This is equivalent to 2.6×10^{-4} counts s^{-1} per resolution element in a spectrum with the default extraction height. These rates have remained stable since SM4.

Figure 4.1: Dark Rates



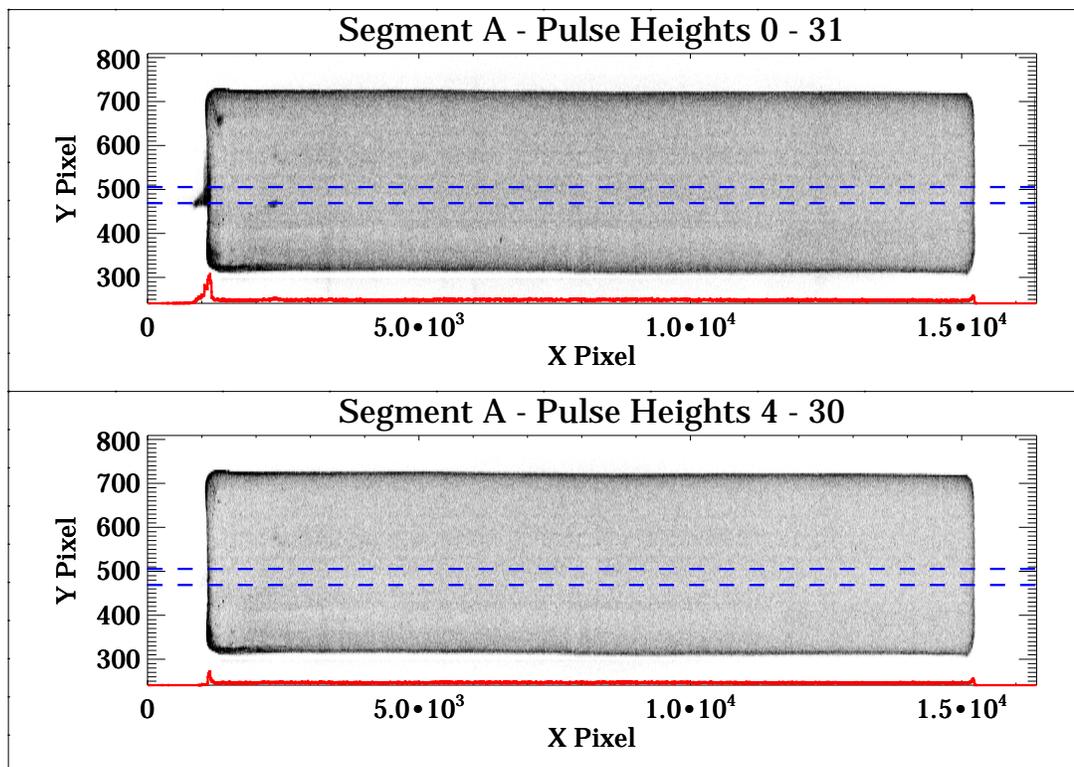
FUV Segment A count rate as a function of time during an orbit which skims the SAA (top), and during one which is further from the SAA (bottom).

When HST passes through the SAA, observations stop and the detector high voltage is lowered in order to prevent damage to the detector. Elevated dark rates of up to 30 times higher than normal (Figure 4.1) have been observed during exposures made when skimming the edge of the SAA. To minimize the observing time with higher background, the SAA model was shifted 6 degrees to the west in May 2010.

The spatial distribution of background counts on Segment A is quite uniform, independent of pulse height thresholding or proximity to the SAA (Figure 4.2). For segment B, however, there are a number of bright spots in the region where the spectra fall when all pulse heights are included. These features disappear when the appropriate pulse height thresholding (used by default in the *calcos* pipeline for TIME-TAG data) is applied, as shown in Figure 4.3.

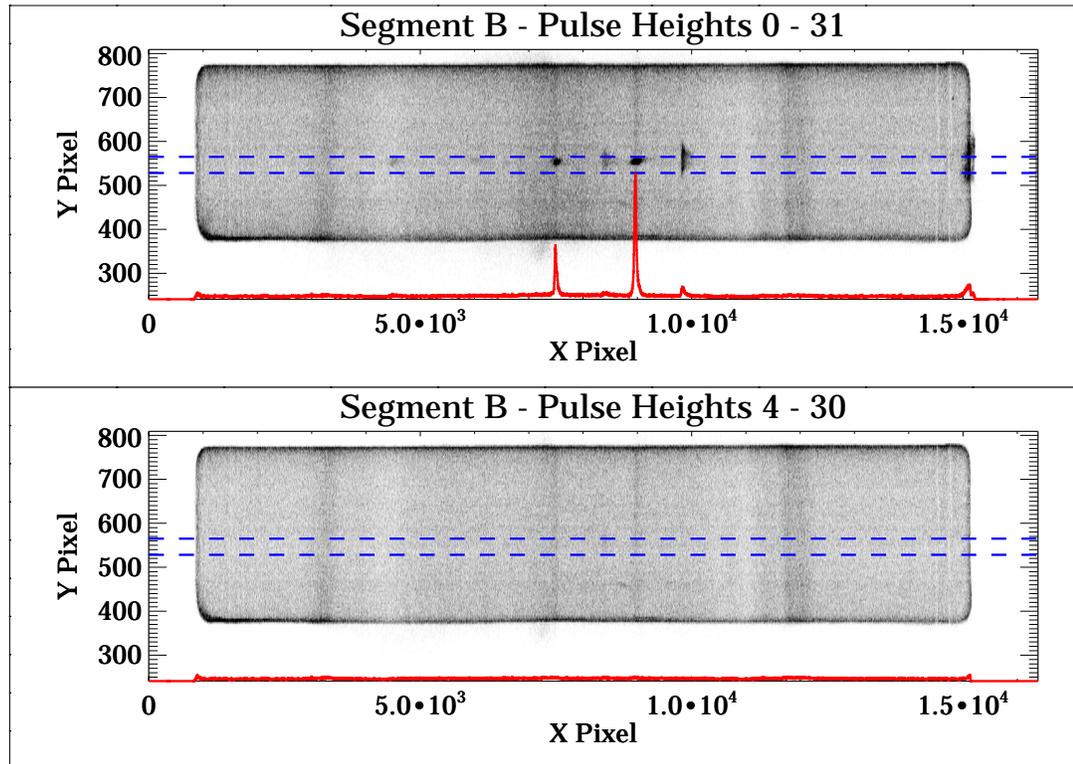
There is an additional complication to FUV dark correction. As the FUV detectors have been exposed to more light, the portion of the detectors where the spectrum falls has become less sensitive. This sensitivity loss even affects the dark count rate. As a result, the background, which is estimated from rarely illuminated regions on either side of the science spectrum, tends to over estimate the dark rate at the location of the spectrum and, therefore, over correct the spectrum. This effect is small, and only affects very faint objects. Nevertheless, one should be aware of it. The team is currently studying a more robust background subtraction, which accounts for this effect.

Figure 4.2: FUVA Dark



Dark rate for FUV Segment A with no pulse height thresholding (top), and with the default thresholding used by *calcos* (bottom). The background is spatially uniform at all pulse heights.

Figure 4.3: FUVB Dark



Dark rate for FUV Segment B with no pulse height thresholding (top), and with the default thresholding used by calcos (bottom). Using the appropriate thresholding minimizes the effects of the extra features near the middle of the segment.

4.2.2 Flat Fields

NUV-MAMA Flat Fields

The STIS MAMA flat fields are dominated by a fixed pattern that is a combination of several effects including “beating” between the micro-channel plates and the anode pixel array and variations in the charge cloud structure at the anode. Similar effects are present in the COS MAMA. Intrinsic pixel-to-pixel variations measured on the ground for the COS NUV-MAMA are 5.2% rms. Analysis of the COS NUV flat-field taken during SMOV by Ake et al. ([COS ISR 2010-03](#)) found that it aligned to within one pixel of the flat field created during ground testing. Consequently, all SMOV and ground data were combined to produce a single flat field reference file for pipeline processing.

The reference file does not correct vignetting, which affects X pixels with values between 0 and 200. The vignetting can eliminate as much as 20% of the flux from X = 0 to 100, and then slowly decrease to 0 between X = 100 and 200. Since the amount of vignetting depends on the angle of illumination, and because the OSM positions do not repeat, simple corrections were inadequate. Due to the low current usage of the NUV channel, a more complex solution has not been pursued.

Studies of the on-orbit S/N achievable indicate that the Poisson limit can be reached for $S/N < 70$ and that a $S/N > 150$ can be achieved by combining high S/N exposures obtained at different FPPOS settings over most of the detector. However, the variable vignetting can introduce large, spatially coherent errors over the first 200 pixels of each stripe of the NUV spectra.

FUV-XDL Flat Fields

The FUV XDL detector has considerable fixed pattern noise. These include dead spots and a honeycomb pattern due to the manufacturing process used to produce the MCP and shadows from the repeller grid wires. A full, two dimensional flat field obtained during internal ground tests did not produce the signal-to-noise needed for a useful flat, and it has been deemed too costly in terms of exposure time and impact on detector lifetime to fully characterize the COS flat field using on-orbit observations.

Nevertheless, some progress has been made. The grid wire shadows, which are the largest single source of fixed pattern noise, are now corrected by the flat fields for the G130M and G160M. However, we currently do not employ a similar grid wire flat for the G140L. Instead, **calcos** flags the grid wire shadows and removes them in the `_x1dsum` files. So if G140L data are obtained at more than one FPPOS setting, a complete spectrum will result which has reduced S/N at the grid wire positions.

Note that even with the correction (for the G130M and G160M) or elimination (for the G140L) of the grid wire shadows, other large amplitude (up to 10%) fixed pattern features remain in the spectra. At present, the best approach to mitigate these is to combine observations obtained at different FPPOS settings. A complete description of the G130M and G160M grid wire flats, and estimates for the achievable S/N for these gratings from normally processed data, are given in [COS ISR 2011-03](#).

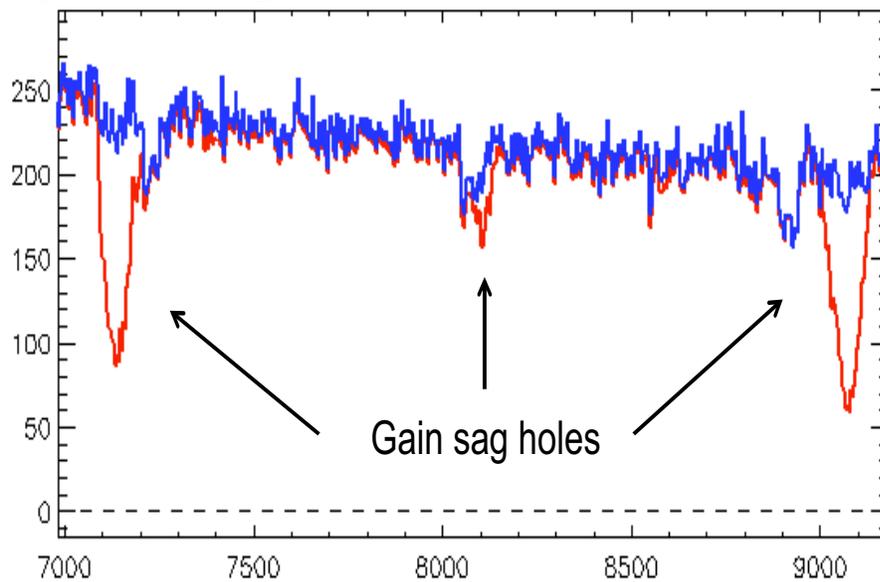
4.2.3 Gain Sag

As described in [Chapter 1](#), the COS XDL FUV detectors experience a loss of sensitivity called gain sag. The COS FUV detectors already experienced localized gain sag in the regions of the detector that are exposed to the bright Ly α airglow line when the G130M is used. These are most serious on the FUVB side near pixels 7150 and 9100. [Figure 4.4](#) shows the effect of changing the lower PHA cutoff from 4 to 2 on the feature centered near pixels 7150 and 9100. With a PHA cutoff of 4, the adjoining pixels are not affected, but the region of gain sag is depressed by nearly 50%. In contrast, with a PHA cutoff of 2, the gain sag regions are depressed by 10%, approximately. As the gain sag deteriorated, changing the PHA threshold no longer worked, and the high voltage had to be raised.



Finally, in July 23, 2012 the COS FUV spectra was shifted to a new lifetime position. For a complete description consult [Appendix A](#).

Figure 4.4: Gain Sag Effects.



This figure shows two versions of the same NET spectrum. The data displayed was taken in December, 2010 with the G160M grating. Red eliminates events with PHAs less than 4 and blue eliminates events with PHAs less than 2.

4.2.4 FUV XDL Thermal Drifts

The XDL centroiding electronics are sensitive to thermal effects. The `TEMPCORR` module of `calcos` measures the location of the stim pulses in order to determine the shift and stretch of the detector format and correct for any changes; `TEMPCORR` applies a linear correction based on the position of these stims. The accuracy of this correction will influence the ability to properly register the flat field corrections and may influence the final error budget. As of this time, no comprehensive study of how well this registration is performing has been carried out, but spot checks indicate that it is working as expected.

4.3 Factors Limiting Flux and Wavelength Accuracy

4.3.1 Flux Accuracy

The accuracy of the absolute flux calibration of COS spectroscopic data is limited by several factors including:

- The presence of fixed pattern noise in the FUV detectors. Although the grid wire shadows are either corrected (G130M, G160M) or removed (G140L), several other artifacts remain, some of them with amplitudes up to 10%.

Because the flux calibration is intended to be a smooth function, it interpolates through such small scale features, which can result in localized errors of plus or minus 5% (see, [COS ISR 2010-02](#) and [COS ISR 2011-03](#)).

- For the G140L settings, the initial wavelength calibrations were not as accurate as they are now. This, coupled with the fact that the instrumental response changes rapidly below about 1200 Angstroms, and that individual FUVB observations may not be properly aligned (see below), results in rather large flux calibration uncertainties (5 - 10%) below 1200 Angstroms for G140L data. A recalibration of this region is planned, and it should alleviate much of the uncertainty. However, the alignment issue will remain.
- **calcos** currently uses a “**boxcar**” extraction procedure. This ignores the instrumental profile perpendicular to the dispersion, and treats all elements within the extraction box equally. To ensure that all of the signal is included, the width of the extraction box is typically larger than the spectrum and, as a result, includes more background signal than necessary. In order to avoid introducing artifacts into the extracted spectrum, the algorithm rejects all data in a column that contains a single bad pixel, thereby throwing away some of the data. On-orbit data have been used to create profiles perpendicular to the dispersion, and the possibility of adopting an optimal extraction algorithm is being investigated.
- The time-dependent sensitivity correction to the FUV flux calibration may not be exact. The FUV sensitivity varies with time, and the rate depends on the grating, segment, and wavelength region considered. Based on our 3 year baseline for COS, the degradation is likely dominated at early times by an out-gassing product and at later times by atomic oxygen in the residual Earth atmosphere at HST altitude. Monthly monitoring of the time-dependent sensitivity captures changes in the sensitivity due to varying atmospheric conditions stemming from variations in the solar cycle. The decline at early times is characterized in [COS ISR 2011-02](#); the complex behavior at early times may add a small error to the calibration of data early in the mission.
- Due to on-board Doppler corrections, a given pixel in ACCUM data will contain data from nearby pixels, which will cause a slight smearing of the fixed pattern noise.
- Because no PHA filtering is done onboard, FUV ACCUMs include events for all PHA values. This has two minor effects. First, background counts are included. However, since objects observed in ACCUM mode are bright, this should not be a practical issue. Second, because the absolute flux calibrations are derived from PHA filtered TIME-TAG data, this can result in small, systematic effects in the flux calibration, but these should be less than a few percent.
- Both of the COS low resolution gratings are affected by order overlaps. For the NUV G230L, wavelengths longer than about 3200Å (which primarily affects the NUVB stripe of CENWAVE 3360), second order light from wave-

lengths longer than 1600\AA can contaminate the result. For the second order G230L spectra (stripe NUVC), first order light from wavelengths at twice the observed wavelength can affect the spectra (see, [COS ISR 2010-01](#)). For the FUV G140L, spectra longward of about 2150\AA can be contaminated by an overlapping second order spectrum (see [COS ISR 2010-02](#)). The exact extent of the contamination depends on the SED of the object being observed.

4.3.2 Wavelength and Spectral Resolution Accuracies

There are several issues that may affect the COS wavelength calibration and spectral resolution, and these are explained in detail in [COS ISRs 2010-05](#), [2010-06](#), [2009-01](#), and [2010-09](#). Some of these issues are outlined here.

- Because the COS optics corrects the HST spherical aberration after light passes through its large aperture, it accepts all of the uncorrected light from the HST Telescope beam. Consequently, its image quality is subject to mid-range polishing errors which create broad wings on the PSF (see, [COS ISR 2009-01](#)). Other spectrographs such as STIS can eliminate the effects of these wings by inserting a small aperture into the beam. Because COS cannot do that, its spectral purity is affected by the wings.
- Small, localized deviations from the dispersion relations determined by a low order polynomial have been reported for FUV XDL data. These deviations most probably result from localized inaccuracies in the geometric correction.
- For the FUVB segment of the G140L CENWAVE = 1280 setting, the wavecal lamp does not have detectable lines. As a result, the wavelength calibration from the FUVA side is applied to the FUVB. However, for some observations, the FUVA is turned off, to avoid an over-bright condition. In these cases, a default wavelength calibration is applied. Note that the wavecal not only affects the wavelength calibration itself, but also the determination of where the PSA or BOA spectrum is expected to be. These same comments apply to FUVB observations obtained with the new G130M CENWAVE = 1055 and 1096 settings that are available for Cycle 19.

- OSM motions, or drifts, can cause the spectrum to shift in the dispersion direction by as much as 2-3 pixels (~ 1 resolution element for NUV, approximately one-half resolution element for FUV) in the first 20 minutes after an OSM is moved. TAGFLASH wavecals correct for these motions to accuracies ≤ 0.5 pixel. However, it is only possible to correct ACCUM data for the mean OSM motion that occurred during the exposure and, in rare circumstances, this may result in a slight degradation in the spectral resolution of ACCUM data.
- The accuracy to which the source is centered in the science aperture along the dispersion direction can result in small displacements in the absolute wavelength scale corresponding to the plate-scales of 0.22 arcsec per FUV pixel and 0.25 arcsec per NUV pixel. Initial measurements for ACQ/IMAGE centering accuracies are of the order of 0.05 arcsec, and accuracies of other types of COS acquisition can be of the order of 0.1 arcsec or more. One can calculate the resulting wavelength accuracy using the plate-scale and dispersion given in [Table 1.4](#) and [Table 1.1](#) respectively.
- As discussed in the *COS Instrument Handbook*, the BOA degrades the target image, resulting in a reduction of the spectral resolution by a factor of three or more.

COS Data Analysis

In this chapter...

5.1 Data Reduction and Analysis Applications / 129
5.2 Evaluating Target Acquisitions and Guiding / 135
5.3 Working with Extracted Spectra / 139
5.4 Working with TIME-TAG Data / 146

5.1 Data Reduction and Analysis Applications

Most of the software tools for operating on COS FITS files are contained in two STSDAS packages:

- **hst_calib.hstcos**: Contains COS specific tasks to calibrate and interpret data, including **calcos**. Many of these tasks are described in [Chapter 3](#). A complete listing is presented in [Section 5.1.1](#).
- **toolbox.ttools**: Contains tasks to read and operate on FITS tables, which is the format used for COS spectral extractions, TIME-TAG data, and most COS reference files. These tasks and their syntax are described in [Section 2.2.2](#) of the *Introduction to HST Data Handbooks*. We give specific examples of their use here in [Section 5.1.2](#).

In addition to the above packages, most basic image manipulation software (e.g., **display**, **daophot**, **imexamine**, **contour**) and spectral analysis software (e.g., **splot**, **tables**, **specfit**, **igi**) available in **PyRAF/IRAF/STSDAS** can be used on COS data, either directly through the **PyRAF** or **IRAF** FITS interface or by converting data to another format. [Chapter 3](#) of the *Introduction to HST Data Handbooks* includes information about how to display COS images and extracted spectra, as well as how and when to convert data formats, and a description of spectral analysis tasks. We present a brief summary of spectral display and analysis tasks in [Section 5.1.3](#).

5.1.1 COS-Specific STSDAS Tasks

In [Chapter 3](#), we gave detailed discussions of the use of the data reduction pipeline **calcos**. This task is contained in the **STSDAS** package **hst_calib.hstcos**. Other tasks useful for reducing and analyzing COS data are contained in this package as well. A complete listing and brief description of these tasks is given in [Table 5.1](#). All of these tasks can be run in **PyRAF**. Consult the on-line help for each task for more information. Some of these tasks will be discussed in greater detail in the remainder of this chapter.

Table 5.1: COS-Specific Tasks

Task	Description
calcos tasks	
calcos	Process COS data through the calibration pipeline.
splittag	Split a corrtag file into multiple sub-files by time interval
x1dcorr	Extract a 1D spectrum beginning with a corrtag file

For the most up-to-date list of COS specific tasks as well as examples please refer to the COS Instrument Web site:

<http://www.stsci.edu/hst/cos>.

5.1.2 FITS Table Tasks

COS spectral extractions, TIME-TAG data, and most COS reference files are stored in FITS tables. (See [Section 2.4.1](#) for a description of the structure of the table extension files for spectral extractions and TIME-TAG data.) Tasks designed to handle this format can be used to examine and extract information from the tables. Here we give specific examples of the use of routines in **ttools** to help you get started. A sample output is given after each command.

To find out what information is given in the columns of a FITS table (the parameters listed here are discussed in depth in [Section 5.4](#)) using the **tlcol** task:

```
--> tlcol lbb920c8q_x1d.fits
# lbb920c8q_x1d.fits[1]
SEGMENT          CH*4          %-4s  ""
EXPTIME          D             %8.3f  s
NELEM            I             %6d   ""
WAVELENGTH       D[16384]     %25.16g angstrom
FLUX             R[16384]     %15.7g "erg /s /cm**2 /angstrom"
ERROR            R[16384]     %15.7g "erg /s /cm**2 /angstrom"
GROSS            R[16384]     %15.7g "count /s"
GCOUNTS         R[16384]     %15.7g count
NET              R[16384]     %15.7g "count /s"
BACKGROUND       R[16384]     %15.7g "count /s"
DQ               S[16384]     %11d  ""
DQ_WGT           R[16384]     %15.7g ""
```

To use **tread** to look at the contents of the table:

```
cl> tread filename_x1d.fits
Column  1      2      3
Label  _SEGMENT_  _EXPTIME_  _NELEM_
      1  FUVA      249.760      16384
```

Note that the number of columns displayed is limited by the width of the window that you are working in. To see more columns, you can start **PyRAF** or **IRAF** in a wider window or populate the task parameter *columns* in **tread** with some of the column names reported by **tlcol**:

```
cl> tread filename_x1d.fits \
columns=' WAVELENGTH, FLUX, ERROR'
Column  1      2      3
Label  _WAVELENGTH_  _FLUX_  _ERROR_
      1  1585.584808800369      0.      0.
```

The **tlcol** output indicates that some of the columns contain arrays of 16384 elements rather than a single value. For those columns, **tread** displays the value of the first element in the array. e.g., the initial wavelength in this x1d extraction is 1585.58

Angstroms. To plot an entire array, or plot one array against another, you can use a routine that operates on the table data directly:

```
cl> sgraph "filename_x1d.fits[1] WAVELENGTH FLUX"
```

You may want to change the parameter values in the supporting parameter sets **axispar**, **dvpar**, and **pltpar** to adjust the plotting. Alternatively, you may find it easier to work with FITS images. To make FITS image files of the arrays:

```
cl> tximage "filename_x1d.fits[1] [c:WAVELENGTH]" filename_x1d_wave
cl> tximage "filename_x1d.fits[1] [c:FLUX]" filename_x1d_flux
```

The images can then be displayed or operated on with standard FITS image handling routines such as python or IDL, e.g.:

```
IDL> filename_x1dwave=readfits('filename_x1d_wave.fits',0)
IDL> filename_x1dflux=readfits('filename_x1d_flux.fits',0)
IDL> plot,filename_x1dwave,filename_x1dflux

or

IDL> filename_x1d=mrdfits('filename_x1d.fits',1)
IDL> plot, filename_x1d.wavelength, filename_x1d.flux
```

Reference file FITS tables generally have many rows, with each row characterizing a specific operating mode, location on the detector, value of a parameter to be used in the reduction, etc.:

```
cl> tread lref$$s7g1700h1_disp.fits
Column 1 2 3 4 5 6
Label SEGMENT OPT_ELEM APERTURE CENWAVE NELEM COEFF
1 FUVA G130M PSA 1291 2 1272.64958352037
2 FUVA G130M WCA 1291 2 1272.21660012193
3 FUVA G130M BOA 1291 2 1272.64958352037
4 FUVB G130M PSA 1291 2 1119.38291962259
5 FUVB G130M WCA 1291 2 1118.98191864602
6 FUVB G130M BOA 1291 2 1119.38291962259
7 FUVA G130M PSA 1300 2 1282.84550637193
8 FUVA G130M WCA 1300 2 1282.41252297349
9 FUVA G130M BOA 1300 2 1282.84550637193
....
```

5.1.3 General Spectral Display and Analysis Tasks

Table 5.2 lists some of the more useful PyRAF/IRAF/STSDAS applications for displaying and analyzing COS spectral data.

Table 5.2: Spectral Analysis Tasks

Task	Input Formats	Purpose
<code>stsdas.analysis.fitting.nfit1d</code>	2-D & 3-D tables, images	General 1-D feature fitting; part of the STSDAS fitting package.
<code>stsdas.graphics.stplot.igi</code>	2-D & 3-D tables, images	General presentation graphics; supports world coordinates.
<code>stsdas.graphics.stplot.sgraph</code>	2-D & 3-D tables, images	General 1-D plotting; supports world coordinates.
<code>stsdas.contrib.spfit.specfit</code>	1-D images, ASCII tables	General 1-D spectrum modelling package.
<code>stsdas.hst_calib.ctools.tomultispec</code>	3-D table	Converts spectra from rows of a 3-D table to an IRAF multispec image.
<code>noao.onedspec.splot</code>	multispec images	General 1-D spectral analysis.

Some of the tasks are intended for browsing data or producing publication quality plots: the **igi** and **sgraph** tasks are described in Chapter 3 of the *Introduction to HST Data Handbooks*.

Specfit is a powerful spectral fitting routine that provides the user with many options for modelling emission and absorption features, continuum, and extinction. The fitting parameters can be linked to each other (e.g., fixed wavelength ratio or fixed flux ratio for two emission lines) and constrained. Different algorithms can be selected to first explore chi-square space, then rapidly achieve convergence in the deepest minimum. The application is fully described in a help file.

Plotting COS Spectral Images (flt)

The FUV data consists of separate files for each of the two FUV detector segments, segment A and segment B. In general, each segment contains one target spectrum and one wavecal. But, for the NUV data there are three disjoint portions of the spectrum present on the image for both the target and the wavecal. The following examples

illustrate the use of the **sgraph** task to plot COS FUV and NUV spectra from the 2-D `flt` files.

```
# target spectrum, FUV segment A:
--> sgraph l61h54cxr_flt_a.fits[1] [* ,451:480]

# wavecal spectrum, FUV segment A:
--> sgraph l61h54czr_flt_a.fits[1] [* ,551:580]

# target spectrum, NUV stripes A, B, C:
--> sgraph l61h57ahr_flt.fits[1] [* ,91:110]
--> sgraph l61h57ahr_flt.fits[1] [* ,196:215]
--> sgraph l61h57ahr_flt.fits[1] [* ,338:357]

# wavecal spectrum, NUV stripes A, B, C:
--> sgraph l61h57ajr_flt.fits[1] [* ,465:484]
--> sgraph l61h57ajr_flt.fits[1] [* ,569:588]
--> sgraph l61h57ajr_flt.fits[1] [* ,710:729]
```

Plotting COS Tabular Spectra (x1d)

COS spectra in tabular format are very similar to STIS spectra. The following examples illustrate the use of the **sgraph** task to plot COS FUV and NUV tabular spectra using a row selector and specifying the columns (e.g., wavelength and flux) to plot.

```
# target spectrum, FUV segments A and B (respectively):
--> sgraph "l61h54cxr_x1d.fits[1] [r:row=1] wavelength flux"
--> sgraph "l61h54cxr_x1d.fits[1] [r:row=2] wavelength flux"

# wavecal spectrum, FUV segments A and B (respectively):
--> sgraph "l61h54czr_x1d.fits[1] [r:row=1] wavelength net"
--> sgraph "l61h54czr_x1d.fits[1] [r:row=2] wavelength net"
```

5.2 Evaluating Target Acquisitions and Guiding

COS target acquisitions and the options available to the observer are fully described in the *COS Instrument Handbook*. If you are examining COS observations that were specified by another observer, please refer to the instrument handbook to understand the options and parameters that may have been used.

Virtually all COS observations start with one or more acquisition exposures. The purpose of the acquisition is to ensure that the object observed is well centered in the COS aperture being used so as to avoid throughput losses and to produce a reliable wavelength zero point. Examining the acquisition data should allow you to detect significant errors in the centering of the target. Note that target acquisition data are always uncalibrated.

5.2.1 Types of Target Acquisitions

There are two types of COS acquisitions: imaging and dispersed-light. In an imaging acquisition, the COS NUV channel is used to obtain an image of the target in the COS aperture. This image is then analyzed by the COS flight software, the object's centroid is calculated, and the object is centered in the aperture. A dispersed-light acquisition directly analyzes the spectrum of the object being acquired and determines how to center the object so as to maximize throughput. Both types of COS acquisitions are intended to work on point sources or point-like sources.

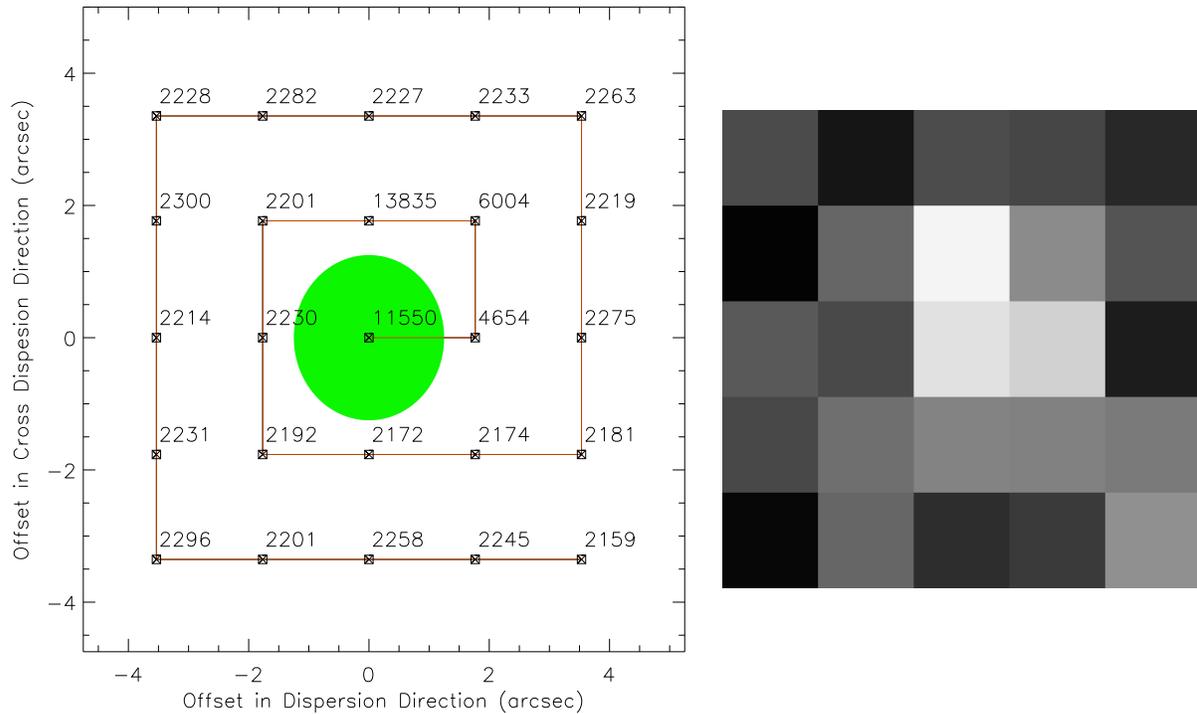
ACQ/SEARCH

A sequence of observations may begin with an ACQ/SEARCH, either in imaging mode or in dispersed-light mode (see the *COS Instrument Handbook* for a full description). The optical element selected will appear in the header: either a grating (and central wavelength) for dispersed light, or a mirror (MIRRA or MIRRORB) for imaging. In either case, the telescope is commanded to move in a square spiral pattern, and at each dwell point an exposure is taken. The STEP-SIZE parameter sets the spacing between dwell points; the default is 1.767 arcsec, the optimum size to ensure that no area of sky is missed. The SCAN-SIZE parameter sets the number of dwells on each side of the square, and the choices are 2, 3, 4, or 5. If an even number of points is used (SCAN-SIZE = 2 or 4), the first point is offset by half the STEP-SIZE in both directions so that the overall pattern remains centered on the initial pointing.

The data from an ACQ/SEARCH exposure consists of a header and a binary table data extension which contains the accumulated counts at each dwell point, see [Table 2.11](#). This array of counts was processed by the flight software to calculate the centroid and the telescope was then commanded to move to that centroid. A *quick verification* that an ACQ/SEARCH exposure was successful would be to find the values of the XDISP_OFFSET and DISP_OFFSET columns of the ACQ/SEARCH data table corresponding to the maximum counts value at a single dwell point. Then,

compare the XDISP_OFFSET and DISP_OFFSET values to the ACQSLEWX and ACQSLEWY header keyword values (see Table 2.7). Similar, the data can be easily plotted for quick visual verification (see Figure 5.1).

Figure 5.1: Example of an ACQ/SEARCH exposure



Left: An example of an ACQ/SEARCH spiral pattern showing the offsets in the dispersion and cross dispersion directions from the initial pointing for each dwell point with the counts at each dwell point overlotted. The green circle in the center represents the science aperture and the initial pointing. **Right:** Linearly scaled image of the counts at each dwell point for the same 5 x 5 ACQ/SEARCH exposure.

Undispersed Light (Imaging) Acquisitions (ACQ/IMAGE)

When the ACQ/IMAGE command is used, two ACCUM exposures in imaging mode are taken for the specified exposure time, using the NUV channel of COS. The first exposure is taken after the initial pointing by *HST* and is used by the flight software to determine the centroid of the object and the amount of pointing change needed to center the object. The second image is taken after the object is centered to confirm that proper centering occurred. Each of the two images uses a sub-array of the size 816×345 on the COS NUV MAMA. The commanded motions of the telescope in x and y are provided in the ACQ/IMAGE header. The `_rawacq` file contains the initial target image as a 1024×1024 array, followed by the confirmation image, another array of the same size.

The appearance of the image of a point source recorded by COS in ACQ/IMAGE mode will depend on the aperture used (PSA or BOA) and the mirror (MIRRORA or

MIRRORB). The best optical quality is achieved with the PSA used with MIRRORA, in which case a diffraction-limited image is created with a tight core. If MIRRORB was used instead to attenuate the source, two images of the source are produced (Figure 5.2). If the BOA was used, a neutral-density filter attenuates the source, but that filter has a slight wedge shape that degrades optical quality. Figure 5.3 shows images of point sources obtained with the BOA using MIRRORA and MIRRORB. Profiles of images taken with various combinations of (PSA, BOA) and (MIRRORA, MIRRORB) are shown in the *COS Instrument Handbook*.

The data produced by ACQ/IMAGE can be used to confirm proper acquisition of an object, by direct comparison of the two images.

Figure 5.2: Example of an image using the PSA and MIRRORB.

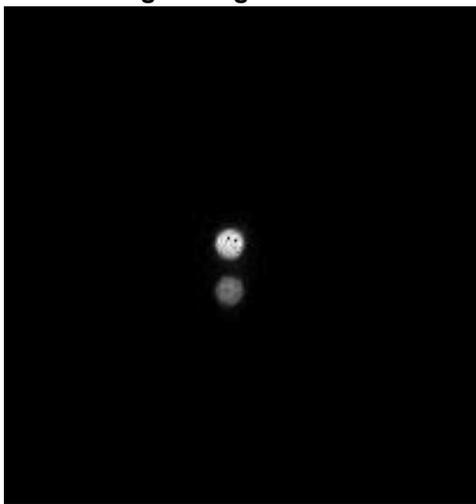
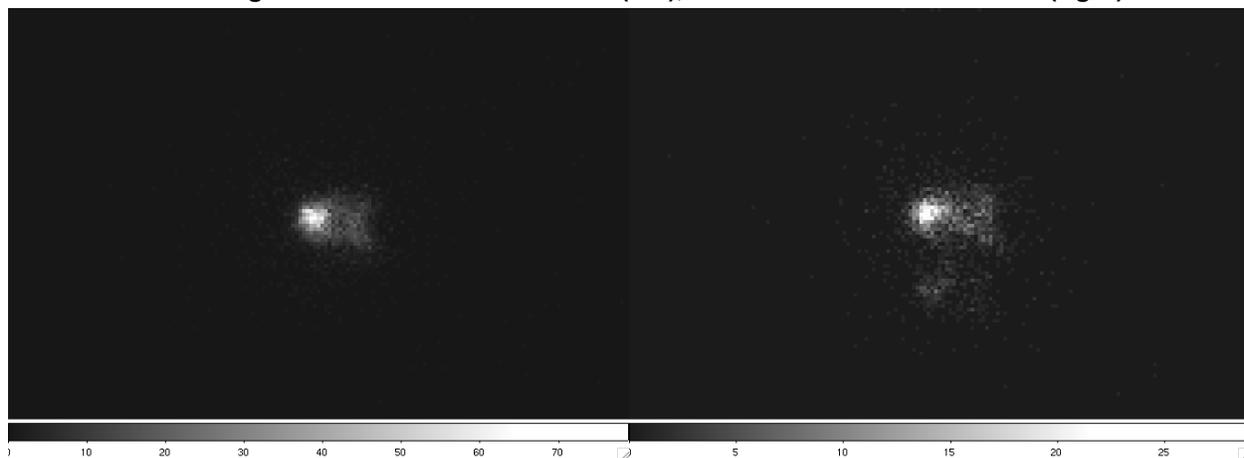


Figure 5.3: BOA and MIRRORA (left), and the BOA and MIRRORB (right).



Example of an image using the BOA and MIRRORA (left), and the BOA and MIRRORB (right)

Dispersed-Light Acquisitions (ACQ/PEAKD and ACQ/PEAKXD)

As noted above, an ACQ/SEARCH exposure can be performed in dispersed light. In that case, the file header will show a grating and central wavelength for the optical element chosen. As for ACQ/IMAGE, any acquisition performed in dispersed light can

use either aperture: the PSA or BOA. In addition to ACQ/SEARCH, two other commands are available to improve the centering of an object in dispersed light: ACQ/PEAKXD and ACQ/PEAKD.

An ACQ/PEAKXD should always precede an ACQ/PEAKD if both were performed. ACQ/PEAKXD centers the spectrum in the cross-dispersion direction by obtaining a short exposure, calculating the centroid, and moving the telescope by that amount. Users will only receive files with headers containing the commanded movement of the telescope for ACQ/PEAKXD exposures. A *quick verification* that an ACQ/PEAKXD exposure was successful would be to compare the ACQSLEWY and (ACQPREFY - ACQMEASY) header keyword values (see [Table 2.7](#)).

ACQ/PEAKD centers the spectrum along the dispersion direction by executing a series of short exposures with the telescope moving the source in a line for a specified number of points (SCAN-SIZE), spaced by STEP-SIZE arcsec (effectively a 1-D ACQ/SEARCH). A centroid is calculated, and the same options available for ACQ/SEARCH are also available for ACQ/PEAKD. Following the centroid calculation, the telescope is moved to center the source, and the counts at each dwell point are recorded in a table, see [Table 2.11](#). Users may compare the offsets associated with the dwell point containing the maximum counts to the telescope slews recorded in the header. A *quick verification* that an ACQ/PEAKD exposure was successful would be to find the value of the DISP_OFFSET column of the ACQ/PEAKD data table corresponding to the maximum counts value at a single dwell point. Then, compare the DISP_OFFSET value to the ACQSLEWX header keyword values (see [Table 2.7](#)). The data can also be easily plotted for a quick visual verification, similar to what is shown in [Figure 5.1](#) for the ACQ/SEARCH example.

5.2.2 Guiding Errors for Single-Guide-Star Mode

Tracking on two guide stars should provide pointing accuracy sufficient to keep targets centered in the COS aperture for several orbits. However, in some cases, observations are made using only a single guide star instead of the usual two. Either the General Observer has consented to this in consultation with the Program Coordinator when two suitable guide stars could not be found, or one Fine Guidance Sensor failed to acquire its guide star during the guide star acquisition/reacquisition. See [Table 2.6](#) for keywords to check for the status of the guide star acquisition. In this situation, the roll of the telescope is under GYRO control, which may allow a slow drift of the target on a circular arc centered on the single guide star. The rate of the drift of the radius of this circle depends on the characteristics of the pointing for any particular observation, but typical values are expected to be in the range of 1.0 to 1.5 milliarcsec/sec (possibly, but very rarely, as large as 5 milliarcsec/sec).

To calculate the approximate magnitude of the drift of the target on the detector, you will need to find the distance of the target from the acquired guide star. The primary header of the observation log file `jif` identifies the acquired guide star (GSD_ID) and gives its right ascension (GSD_RA) and declination (GSD_DEC) in degrees. For example, for a target 10 arcmin from the guide star, a drift of the target around the guide star of 1 milliarcsec/sec during a 1,000 second exposure would cause

the target to move 0.0029 arcsec on the detector. The direction of the motion on the detector can be deduced from header keywords in the science data describing the position angle of the detector (e.g. PA_APER) in combination with the direction perpendicular to the radiant. In many cases, the drift will be a small fraction of a pixel, although in some cases an image exposure may appear smeared.

5.3 Working with Extracted Spectra

Here we discuss ways of customizing the extraction of spectra and modifying reference files.

5.3.1 Working With `x1d` Files in IDL

While STScI does not support IDL tasks, the FITS files generated by the **OPUS** and **calcos** pipelines can be read directly into IDL by using the **mrdfits.pro** task. This task and many other are available as part of a library of astronomical IDL routines available from:

<http://idlastro.gsfc.nasa.gov/>

5.3.2 Working With `x1d` Files in Python

STScI provides support for PyFITS, a module that provides an interface to FITS formatted files in the Python scripting language and **PyRAF**, the Python-based interface to IRAF. It is useful both for interactive data analysis and for writing analysis scripts in Python using FITS files as either input or output. More information, including the PyFITS handbook, can be found at

http://www.stsci.edu/resources/software_hardware/pyfits.

5.3.3 Working With `x1d` Files in IRAF/PyRAF

When calibrating a single spectroscopic exposure, the **calcos** pipeline creates a one-dimensional extracted spectra file, with suffix `x1d` and a filename such as “19v220eqq_x1d.fits”.

COS `x1d` files are MEF format files and their data contents and extension formats are discussed in [Section 2.4.3](#). As with other COS data files, the primary [0] extension will contain only header information, but no data. The extracted spectra are stored in a single [SCI] extension as *multi-dimensional* binary table. A standard FITS table consists of columns and rows forming a two-dimensional grid of cells; however, each of these cells can contain a data array, effectively creating a table of higher dimensionality. Tables containing extracted COS spectra take advantage of this feature and are *three-dimensional*.

Using the “Selectors Syntax” to work with 3-D tables

In order to analyze COS tabular spectral data with **STSDAS** tasks other than **sgraph** and **igi** (which have been appropriately modified to handle three-dimensional data tables), you will need to extract the desired arrays from the three-dimensional table. Two IRAF tasks, named **tximage** and **txtable**, can be used to extract the table-cell arrays. Complementary tasks, named **tiimage** and **titable**, will insert arrays back into table cells. The task **tscopy** will copy rows, columns, and subsets of tables. To specify the arrays which should be extracted from or inserted into the table cells, you will need to use the *selectors* syntax to specify the desired row and column. The general syntax for selecting a particular cell is:

```
intable.fits [extension number] [c:column_selector] [r:row_selector]
or
intable.fits [keyword options] [c:column_selector] [r:row_selector]
```

A *column selector* is a list of column patterns separated by commas. The column pattern is either a column name, a file name containing a list of column names, or a pattern using the **IRAF** pattern matching syntax (type "help system.match" for a description of the **IRAF** pattern matching syntax). To obtain a list of the column names, you can run the **tlcol** task (type "tlcol infile.fits").

A *row selector* can be used to specify a certain row in the table. For example, if you specify:

```
infile.fits [1] [c:WAVELENGTH, FLUX] [r:SEGMENT=FUVA]
```

IRAF will extract data from the table stored in the first extension of `infile.fits`, specifically from the columns labeled `WAVELENGTH` and `FLUX`, and will restrict the extraction to the row containing segment A data. Section 2.3.2 of the *Introduction to HST Data Handbooks* describes the selectors syntax and provides a few examples using the selector syntax to plot COS spectra.

Dumping x1d data to an ASCII File

In **PyRAF**, it possible to dump the arrays of an `x1d` file to an ASCII file by using the tasks **txtable** and **tdump**. For example, to extract the `WAVELENGTH`, `FLUX` and `ERROR` columns of FUV `x1d` file `l9v221fkq_x1d.fits`, first use the **txtable** task to convert the 3-D `x1d` table to a 2-D table:

```
--> txtable l9v221fkq_x1d.fits [1] [r:row=1:2] l9v211fkq
l9v221fkq_x1d.fits [1] [r:row=1:3] row=1 -> l9v211fkq_r0001
l9v221fkq_x1d.fits [1] [r:row=1:3] row=2 -> l9v211fkq_r0002
```

This will create two new 2-D tables, `l9v221fkq_r0001.tab` and `l9v221fkq_r0002.tab`, containing the FUV A and FUV B data respectively.



If using `xtable` to extract rows of 3-D COS tables, always use the row selector syntax of `[r:row=1:50]` to extract all possible rows. Even if there are less than 50 rows (as is always the case), `xtable` will return the correct number of files without error. This can be useful for dealing with the lampflash tables that can have varying number of rows depending on the number of flashes.

Then use the `tdump` task to dump the WAVELENGTH, FLUX and ERROR columns of the 2-D tables into one ASCII file using the following commands:

```
--> tdump l9v221fkq_r0001.tab columns="WAVELENGTH, FLUX, ERROR, DQ_WGT" \
>> l9v221fkqtest.tab

--> tdump l9v221fkq_r0002.tab columns="WAVELENGTH, FLUX, ERROR, DQ_WGT" \
>> l9v221fkqtest.tab
```

Plotting COS x1d Data

Each row of each of the science extensions in an `x1d` file will contain the columns listed in [Table 2.5](#); a similar table, including array dimensions, can be displayed by using the task `tblcol` (see Section 2.3.2 of the *Introduction to HST Data Handbooks*).

When using many IRAF and PyRAF routines with `x1d` files as input, it will be necessary to specify the extension number of the file. For example, to plot flux vs. wavelength in an `x1d` file, using the `sgraph` task, the command needed would be:

```
cl> sgraph "l9v220eqq_x1d.fits[1][r:row=1] wavelength flux"
```

For FUV data, the `x1d` files contain both¹ segments A and B. For example, to plot flux vs. wavelength in an FUV `x1d` file for segment A, using the `sgraph` task, the command needed would be:

```
cl> sgraph "l9v220eqq_x1d.fits[1][r:row=1] wavelength flux"
or
cl> sgraph "l9v220eqq_x1d.fits[1][r:SEGMENT=FUVA] wavelength flux"
```

1. For FUV `x1d` files, both segments A and B will be present as long as the individual raw data from both segments were available at the time of processing. If only one segment was present during processing, then a row selector of `row=1` will point to the data from that segment. Similarly, a row selector of `row=2` will result in an error.

To plot flux vs. wavelength in an FUV `x1d` file for segment B, using the **sgraph** task, the command needed would be:

```
cl> sgraph "l9v220eqq_x1d.fits[1][r:row=2] wavelength flux"
or
cl> sgraph "l9v220eqq_x1d.fits[1][r:SEGMENT=FUVB] wavelength flux"
```

For NUV data, the `x1d` files contain the three stripes A, B, and C. For example, to plot flux vs. wavelength in an NUV `x1d` file for stripe A, using the **sgraph** task, the command needed would be:

```
cl> sgraph "l9v220efq_x1d.fits[1][r:row=1] wavelength flux"
or
cl> sgraph "l9v220efq_x1d.fits[1][r:SEGMENT=NUVA] wavelength flux"
```

To plot flux vs. wavelength in an NUV `x1d` file for stripe B, using the **sgraph** task, the command needed would be:

```
cl> sgraph "l9v220efq_x1d.fits[1][r:row=2] wavelength flux"
or
cl> sgraph "l9v220efq_x1d.fits[1][r:SEGMENT=NUVB] wavelength flux"
```

To plot flux vs. wavelength in an NUV `x1d` file for stripe C, using the **sgraph** task, the command needed would be:

```
cl> sgraph "l9v220efq_x1d.fits[1][r:row=3] wavelength flux"
or
cl> sgraph "l9v220efq_x1d.fits[1][r:SEGMENT=NUVC] wavelength flux"
```

`x1d` files can also be displayed and analyzed using the **splot** routine in the **noao.onedspec** package of **IRAF** or **PyRAF**. This requires converting the `x1d` binary table first into a standard **IRAF** format (or OIF) which can be read by **splot**. The standard format is usually `.imh` format, containing the flux and wavelength information. The most convenient way of extracting the wavelength and flux information from the `x1d` table is to use **tomultispec** in the **stdas.hst_calib.ctools** package. **tomultispec** takes rows from the `x1d` table and converts them into standard **IRAF** form. It fits a dispersion solution to each wavelength in the array and stores the solution in the header of the output **IRAF** spectrum.

If a COS `x1d` table contains spectra from segments A and B of the FUV channel (as would be the case if **calcos** were ran with both `_rawtag_a` and `_rawtag_b` files present), then the spectrum output by **tomultispec** will consist of two spectra, one

from each channel. Running **tomultispec** on an NUV `x1d` file will always produce a multispec file with three separate spectra, one for each stripe.

The conversion can be done using the following command:

```
cl> tomultispec 19v220eqq_x1d.fits 19v220eqq_x1d.imh
```

This command will write the spectra from both channels to a single `.imh` file. However, a single channel can also be written if desired by specifying the row to be written. For example, to write spectra from FUV segment A use the following syntax:

```
cl> tomultispec 19v220eqq_x1d.fits[r:row=1] 19v220eqq_x1d.imh
```

Segment B would be denoted by `row=2`. For NUV data the three stripes are denoted as rows 1, 2, and 3 for Stripes A, B, and C respectively. The `FLUX` column is extracted by default, though other columns such as `GROSS`, `NET`, or `ERROR` could also be extracted.

If running **tomultispec** produces a segmentation fault, bus error, or other equivalent issue, the problem may be solved by setting a value for `IMDIR` in `login.cl` (in particular, setting `IMDIR=HDR$` will generally ensure that **tomultispec** functions as intended).

The **IRAF** spectrum can now be displayed using **splot**:

```
cl> splot 19v220eqq_x1d.imh
```

Within **splot**, individual emission lines can be fit with commands like ‘`k`’ and composite lines can be deblended using ‘`d`’.

5.3.4 Redoing Spectral Extraction

The `x1dcorr` module in **calcos** is designed to extract flux calibrated 1-D spectra from corrected COS event lists (`corrtag` files). This module is called by **calcos** as part of standard pipeline processing; its functioning in that role is described in [Section 3.4.15](#).

Correcting for Shifts Along the Dispersion Direction

Properly aligning the spectrum along the dispersion direction is important not only for obtaining the correct wavelength solution, but also for properly applying the flux calibration. Incorrect registration of the spectrum will result in the wrong sensitivity being applied at each wavelength. This is especially important for low resolution spectra, since at some wavelengths the sensitivity changes rapidly with wavelength.

For `rawtag` exposures the wavecal lamp exposures are taken either concurrently with the science `rawtag` spectra (`TAGFLASH`) or they are acquired as separate `rawtag` spectra (`AUTO` or `GO` wavecals). For all science `rawaccum` exposures the wavecals are acquired as separate `rawtag` exposures.

The wavecal exposures are used by **calcos** to determine the location of both the wavecal image and the corresponding science image on the detector. The location may vary in a non-repeatable manner due to non-repeatability of the COS grating positions. When auto-wavecals are acquired as separate exposures they are taken close in time to the science exposures, with the grating in the same position as during the science exposure.

After processing data through **calcos**, you may decide that you need to shift the spectrum along the dispersion direction to correct offsets in the wavelength calibration. For example, wavelength calibration offsets may occur due to offsets of the target from the center of the PSA aperture (which can occur if the target acquisition was imperfect), or from repositioning of the grating due to thermal flexures. Assuming that **calcos** has been run on the data and a residual wavelength offset has been found in the calibrated spectrum, the offset can be corrected by first calculating the number of pixels corresponding to the offset, then subtracting it from the raw position of coordinates along the dispersion direction. The shift is applied to the RAWX column of the `rawtag` or `rawtag_(a,b)` file. In **IRAF** and **PyRAF** the **tcalc** task from the **ttools** package can be used to apply the shift:

```
c1> tcalc "l9v220eqq_rawtag_a.fits[1][r:RAWY=Y1:Y2]" "RAWX" "RAWX + SHIFT"
```

In the example above, the RAWX positions of the science spectrum in an FUV A `rawtag` file have been moved by "SHIFT" pixels. The shift is only applied to the parts of the science spectrum located between the pixels RAWY=Y1 and RAWY=Y2 along the cross-dispersion direction.

Adjusting the Background Subtraction

For spectra, a background region offset from the extraction region is used to determine the background. You can adjust the default parameters for this background region by first copying the `_1dx` reference file listed under the `XTRACTAB` keyword in the primary header to a local directory, then adjusting the background parameters within the local version of the `_1dx` reference file. Once you have adjusted the parameters to your satisfaction, edit the primary header of the `_rawtag` file (with an **IRAF** task such as **hedit**) to indicate the path to the local version of the `_1dx` file. You can then run **calcos** with the updated background subtraction parameters.

The background parameters available for editing in the `_1dx` file are:

- **SLOPE**: the slope of the line tracing the centers of both the spectrum and background regions
- **B_BKG1**: the y-intercept of the line tracing the center of the background region lying below (smaller Y- coordinate) the science spectrum
- **B_BKG2**: the y-intercept of the line tracing the center of the background region lying above (larger Y- coordinate) the science spectrum

- **BHEIGHT**: the total cross-dispersion height, in pixels, of the background extraction region. The upper and lower edges of the background are defined as $\pm (\text{BHEIGHT}-1)/2$ pixels from the line tracing the center of the background extraction region
- **BWIDTH**: the width of the background extraction region along the dispersion direction
- **HEIGHT**: the total cross-dispersion height, in pixels, of the source extraction region. The upper and lower edges of the source extraction region are defined as $\pm (\text{HEIGHT}-1)/2$ pixels from the line tracing the center of the source spectrum.

The values of these parameters in the local `_1d.x` file can be edited with the **tedit** task in **IRAF/PyRAF**.

5.3.5 Splicing Extracted Spectra

The task **splice** can be applied to combine overlapping extracted COS spectra (e.g. spectra taken with different central wavelengths). It takes into account the error (**ERR**) array as well as the data quality (**DQ**) array. Handling of the **DQ** array is important as it helps **splice** perform the combination properly and avoid bad or noisy data in the output file arising from the large changes in throughput at the edges of the detector.

```
cl> splice obs1_x1d.fits,obs2_x1d.fits output_splice.fits
```

Please refer to the **splice** task help file for more useful information. If a multispec format spectrum is preferred for further analysis, the task **tomultispec** can be run on the output file of the **splice** task.

Running **splice** as mentioned above (rather than transforming individual `x1d` fits tables into multispec format before combining them) has important advantages: it keeps the science data, error, and **DQ** arrays intact allowing for easier error analysis, and it does not have a limitation on the number of segments or wavelengths to include, a problem with the multispec format due to the limit on the size of the FITS header which requires fitting the wavelength scale with a function.

5.4 Working with TIME-TAG Data

COS detectors can be used in ACCUM or TIME-TAG modes, as described in Chapter 5 of the *COS Instrument Handbook*. In TIME-TAG mode, the position and detection time of every photon is recorded in an events list. Detection times are recorded with 32 millisecond precision, although events may be buffered for as long as 32 milliseconds prior to assignment of a detection time.

For TIME-TAG datasets, the *HST* archive returns a raw events list in a file with a `_rawtag` suffix. The `_rawtag` file is a FITS file with two binary table extensions. The first extension contains the events list and the last extension a list of good time intervals, indicating time intervals when events are valid.

An events list in a `_rawtag` file is a FITS binary table extension named EVENTS, containing four columns named TIME, RAWX, RAWY, and PHA. Note only FUV data will include the PHA columns in the `_rawtag` files.

The TIME in the events extension contains the time when each event was recorded, relative to the start time (MJD) of the exposure given in the EXPSTART keyword of the primary FITS header.

In TIME-TAG the RAWX column contains the pixel coordinate along the spectral axis where each event was recorded. Corrections to remove Doppler shifts introduced by the orbital motion of *HST* are applied by `calcos` and placed in the `corrtag` file. The correction depends on optical element and the projected orbital velocity of *HST*, which varies over the course of an observation. In ACCUM mode, this Doppler compensation is applied on orbit during an observation and is included in the RAWX column, but in TIME-TAG mode the uncorrected positions are downlinked and Doppler compensation is applied during ground processing. The RAWY column contains the pixel coordinate along the spatial, or cross-dispersion, axis. No Doppler compensation is applied. The PHA column (for FUV data only) contains the pulse height amplitude for each event as an integer on a 5-bit scale.

After all EVENTS extensions in a `_rawtag` file, there will be one final binary table extension named GTI, containing columns named START and STOP. There will be associated start and stop times for every uninterrupted observing interval during a planned exposure. For most datasets, there will be only one START and one STOP time encompassing all buffer dumps in an exposure. Multiple good time intervals are possible, however - for example, if guide star lock is lost. Times in START and STOP are expressed in seconds since the start time (MJD) of the exposure given in the EXPSTART keyword of the primary FITS header. The exposure start time (JD) is also provided in the EXPSTARTJ keyword of the primary FITS header. The start time is also provided in the In IRAF, good time intervals can be examined using the `tprint` task in the `tables` package:

```
cl> tprint rootname_rawtag.fits[GTI]
```

where rootname must be replaced by the rootname of the `_rawtag` file being examined.

Useful **IRAF** tasks for analyzing and manipulating data taken in the TIME-TAG observing mode are listed in [Table 5.3](#).

Table 5.3: Useful IRAF Tasks for Reducing TIME-TAG Data

Task	Purpose
<code>tpar, tprint</code>	Read the value for table header keywords
<code>sgraph</code>	Display the 1-D spectrum
<code>tomultispec</code>	Convert x1d spectrum into IRAF .imh spectrum
<code>splot</code>	Display an IRAF .imh spectrum
<code>splice</code>	Splice spectra together

5.4.1 Displaying TIME-TAG Data in DS9

To view a TIME-TAG file (`_rawtag_(a,b)` in the FUV, `_rawtag` in the NUV), open **ds9**, then choose ‘open’ from the menu bar at the top. The image will load but, save for a few pixels registering a value of 1, the remaining pixels will be zero.

Once the image is loaded, go to the menu item ‘bin’ and open the pull-down menu from that. From that pulldown menu you can choose the size of the image to view – generally you should make it as big as possible: 8192X8192 pixels for FUV data, 1024X1024 pixels for NUV.

NOTE: for the instructions below, the changes will not take effect until you click on the ‘Apply’ button.

Now choose ‘Binning Parameters’ from the ‘bin’ pulldown menu. This will open a new window with the binning parameters listed. You will notice right away that the Bin Columns are listed as TIME and RAWX. These are what is currently being displayed by **ds9** (which is why the image looks so strange when initially loaded). However, what you really want is RAWX vs. RAWY, so change that in the pulldown menu under Bin Columns.

You can also set the blocking size of the image in the ‘Binning Parameters’ window – just type in ‘2’ in the Block field next to RAWX. By blocking this way along the dispersion direction, you can now see virtually all of the 16384 pixels along the dispersion direction. If you are looking at NUV data, then no additional blocking is needed – just leave the blocksize as 1, but choose the image size as 1024 pixels from the ‘bin’ pulldown menu.

Spectroscopic Data

Next, from the ‘Binning Parameters’ window choose the part of the spectrum to be centered on the middle of the dispersion direction by clicking on the button marked

‘center of data’. Now press ‘Apply’ on the binning parameters window to update the **ds9** display.

The spectrum should now be displayed, with the dispersion direction running from left to right. To better see the data, choose ‘scale’ under the main **ds9** menu bar, and from that pulldown menu choose a square root stretch and min/max range. You can now pan your cursor over the image, while holding the right button down on your cursor, until the contrast looks just right. If you would like to smooth the data a bit (this can be useful for bringing out fainter features and increasing signal to noise along the display), choose the ‘Analysis’ menu item under the main **ds9** menu bar and select ‘smooth parameters’. A dialogue box will open, and from there you can set the number of pixels to smooth. Finally, you can also click on the ‘Color’ item on the **ds9** menu bar and choose ‘invert color map’ to get an inverted color map.

You can also load a `_corrtag_(a,b)` table in **ds9**, but in this case the appropriate columns to display are `XFULL` and `YFULL`. Otherwise, the same **ds9** commands apply as for `_rawtag` files. For both TIME-TAG and ACCUM spectroscopic data the `_flt` and `_counts` spectral images will load as simple 2-D images in **ds9**.

Imaging Data

For both TIME-TAG and ACCUM imaging data The `_flt` and `_counts` images will load as simple 2-D images in **ds9**.

TIME-TAG Animation

You can assign events registered during each time interval to a separate image in **ds9**, thereby creating a sequence of images which can be played as an animation. This can be useful in verifying the occurrence of lamp flashes in TAGFLASH data, in searching for the appearance of bursts in raw data, and so on. To bin the images in time, set up the image as described above – with RAWX and RAWY chosen in the ‘Binning Parameters’ dialogue box. At the bottom of the ‘Binning Parameters’ box is a parameter called ‘Bin 3rd Column’. Set the value of this parameter to TIME. Next, choose the number of bins you would like to divide the event file into under the ‘Depth’ parameter. Setting this value to 10, for example, will create 10 separate images, with the first one showing all events registered during the first ($EXPTIME/10$) seconds, the next one showing all events registered between ($EXPTIME/10$) and ($2*EXPTIME/10$) seconds, the next showing all events registered between ($2*EXPTIME/10$) and ($3*EXPTIME/10$), and so on up to EXPTIME. The ‘Min’ and ‘Max’ parameters let you choose the range of values in time to display – usually this is pre-set to 0 and EXPTIME, and can be left unchanged to bin the entire image as above. Select ‘Apply’ to do the binning.

Note that some time will be required to create the sequence of images, and that binning the events in time in **ds9** is very memory intensive, and that it is easy to make **ds9** crash if EXPTIME is large (for example >1000 seconds) and the number of bins in ‘Depth’ is set to a large value (for example 30). It is best to start with a small value for ‘Depth’ that works, then increase the value if needed.

After the binning is done, a new dialogue box will appear called ‘Data Cube’. Numbered from left to right will be the enumeration of the bins (in the example above from 1 to 10), along with a slider underneath. Click on ‘Play’ in that window to start

the animation – it will play each of the binned images sequentially in the **ds9** window. Again, the spacing between each of the bins will be $(\text{EXPTIME}/10)$ in seconds, or $(\text{EXPTIME}/\text{Nbin})$, where Nbin is the number of bins.

In the animation, it should be possible to see the TAGFLASH spectrum appear and disappear as the sequence progresses. Obviously the sequence will show the flashes only if the keyword TAGFLASH=AUTO or TAGFLASH="UNIFORMLY SPACED" is in the header of the event file.

To exit from the animation, close the ‘Data Cube’ window, and then set the ‘Depth’ parameter in the ‘Binning Parameters’ dialogue box to zero, and click ‘Apply’. That will reset the image in **ds9** to show all of the data again.

It is possible to bin in other parameters as well, such as PHA. The logic is the same as above.

5.4.2 Filtering Time-Tag Data

Filtering Events in the Timeline Extension

As mentioned in [Section 2.4.2](#), all corrtag files processed with **calcos** 2.14 or later contain a timeline extension. The timeline extension can be operated on by the timefilter module to exclude photon events that match user-specified patterns in the time extension. The timefilter module is available as part of STSCI_PYTHON, and requires **calcos** version 14+ to work. The normal use of timefilter is to exclude daytime events in order to minimize the contribution of geocoronal Lyman alpha or O I emission lines to your data. Timefilter will filter events according to a filter string passed to it.

The filter string consists of one or more filter conditions, separated by "and", "or", or "xor" (parentheses are currently unsupported). Each filter condition consists of a column name, a relation, and a cutoff value. Valid column names are "time", "longitude", "latitude", "sun_alt", "target_alt", "radial_vel", "shift1", "ly_alpha", "OI_1304", "OI_1356", and "darkrate" (see [Table 2.3](#) for a description of the columns). Valid relations are '>', '>=', '<', '<=', '==', and '!='. Cutoff values are numerical values. In addition, it is possible to flag events based on one of the 32 SAA model contours with the filter condition "SAA #" where # is a number from 1 to 32. Events which match the filter string will be marked with the DQ flag 2048 (bad time interval), and will be excluded in the creation of flt and x1d files.

Timefilter can either modify an existing CORRTAG file in place, or create a new one, and it can be run in conjunction with splittag (although in that case, it is possible that some output files will contain no valid events at all). The file produced may be extracted with the **x1dcorr** task as usual. It is possible to remove any events filtered with timefilter by running "timefilter.py 'input_file.fits' " reset" followed by "timefilter.py 'input_file.fits' " clear".

The following examples show common uses of Timefilter:

- Take "test_corrtag_a.fits", flag all data taken during orbital day (sun_alt > 0), and save in the file "output_corrtag_a.fits"

```
timefilter.py test_corrtag_a.fits output_corrtag_a.fits 'sun_alt > 0'
```

- Filter "xyz_corrtag_b.fits" in place to remove data with (sun_alt > -10 AND ly_alpha > 2.5) OR taken in the SAA 31 profile

```
timefilter xyz_corrtag_b.fits "" "sun_alt > -10 and ly_alpha > 2.5 or saa 31"
```

- Remove filters from "xyz_corrtag_b.fits"

```
timefilter.py xyz_corrtag_b.fits " reset
timefilter.py xyz_corrtag_b.fits " clear
```

Manipulating TIME-TAG Data for Variability

Users may wish to process only sub intervals of TIME-TAG events, to look for variability in the data. One way to do this would be to divide an exposure up into several sub-exposures before re-processing by using the **splittag** program.

splittag operates on `corrtag` files, so you will need to retrieve the calibrated data (by using `corrtag` files, **calcos** is able to use the existing wavelength fits derived during the calibration process, and as such splitting data with lamp flashes will not result in wavelength calibration information being unavailable).

The **splittag** task is available as part of the STSDAS package within IRAF (**stsdas.hst_calib.hstcos**). It is a useful tool for dividing a COS time-tag exposure (FUV or NUV) into a series of sub-exposures with time intervals specified by the user. The task operates on the **calcos** `corrtag` files, copying rows from a `corrtag` file into one or more output files. The number of files depends on the number of time intervals specified by the user. The resulting `corrtag` sub-exposures can then be run separately through the **x1dcorr** task in **stsdas.hst_calib.hstcos** to extract one-dimensional, flux-calibrated spectra (`*_x1d.fits` files) for each file.

The following keywords are modified when **splittag** copies the time columns to the new `corrtag` files: `EXPTIME`, `EXPEND` and `EXPENDJ`. The keywords `EXPSTART` and `EXPSTRTJ`, on the other hand, are not changed. The `EXPTIME` keyword in each of the new `corrtag` files will be set to the duration of the time interval being extracted, while the modified Julian date and Julian date in `EXPEND` and `EXPENDJ` will be set to the following:

$$\text{EXPEND} = \text{EXPSTART} + t_{\text{end}}(i)$$

$$\text{EXPENDJ} = \text{EXPSTRTJ} + t_{\text{end}}(i)$$

where $t_{\text{end}}(i)$ is the ending time of the i th desired sub-exposure. In addition to the updated keywords, **splittag** also produces updated GTI (good time interval) tables for each of the output `corrtag` files. The GTI intervals are specified relative to the times of the original `corrtag` file, such that the split `corrtag` files will not include events outside the GTI values. The `EXPTIME` keyword written to the `corrtags` affected by the GTI intervals is shortened accordingly.

There are two ways to run the **splittag** task: (1) specify a starting time, an increment, and an ending time, or (2) provide an explicit list of times (not necessarily adjacent to one another). In either case, the output `corrtag` files will have a root

name specified by the user. If no root name is specified, the root name of the input `corrtag` will be used, appended with numbers 1,...N for N exposures.

The parameters input by the user for **splittag** include the following: an input `corrtag` file name, a root name for the output files, the starting time for the first event to be extracted, the time increment to be used in extracting the following intervals, and the ending time of the extraction. If option (1) from above is used, then the starting time and increment are specified, with the remaining parameters left at their indefinite values. This will extract however many `corrtag` files are needed until the ending time of the original exposure is reached. If option (2) is used, then the user can specify explicitly, in the form of start/stop pairs, which intervals are desired. For example, specifying `time_list="0,20,100"` will extract events in the range $0 < t < 20$ seconds and output that to a `corrtag` file, then extract events in the range $20 < t < 100$ seconds and write that to another file, and so on. **splittag** can also read in a text file with the start/stop pairs entered (using the format in the example above). In that case, all the start/stop pairs would be listed in one line in the text file, separated by either commas or spaces. If this option is used (i.e., the `time_list` parameter is set to point to the text file), then the `starttime` and `increment` parameters are ignored.

For example, to split the exposure, `161h9002r_corrtag.fits`, into two equally spaced sub-exposures:

```
cl> splittag 161h9002r_corrtag.fits split increment=INDEF /
time_list="0,60,120"
```

Next, the two sub-exposures should be extracted with **x1dcorr**. To instead split the exposure into 20-second increments, the following command would be used instead:

```
cl> splittag 161h9002r_corrtag.fits split INDEF /
increment=20 INDEF time_list=INDEF
```



APPENDIX A:

COS New Spectra Position

In this appendix. . .

A.1 COS New Lifetime Position / 152

A.1 COS New Lifetime Position

To overcome gain sag effects, the position of spectra on the COS FUV detector was moved on July 23, 2012 to a fresh part of the detector which has not yet experienced significant gain sag. All GO observations will now use this new lifetime position.

This new position is displaced in the cross-dispersion direction by 3.5" (no dispersion direction displacement), corresponding to ~41 pixels, and should, at least for the next two to three years, eliminate the deleterious effects of gain sag on detector throughput and flat fielding.

This change in lifetime position will result in a slight decrease in spectral resolution (~5-10%), but will avoid the low response areas that have resulted from the gain sag and which had been starting to compromise the quality of the data taken at the original position. Small changes to the wings of the line spread function are predicted; new LSF models have been computed and will be made available to users once they are validated by calibration observations at the new lifetime position.

The flux and wavelength calibrations of the new lifetime position are expected to be very similar to those at the original position. Calibration observations will be executed during the Summer and Fall of 2012 and updates to calibration reference files, if needed, will be promptly made. These observations include measurements of the resolution, verification of the wavelength scales, verification of the FUV BOA operations, and flux and flat field calibration observations.

Changes have already been implemented in the calibration pipeline and associated calibration reference files so that data taken at the new lifetime position can be properly calibrated. These changes include new header keywords or modifications to existing ones. [Table A.1](#) lists the keywords that are relevant to the lifetime position, indicates the possible values and provides a brief description on the meaning of these keywords.

Table A.1: Selected Header Keywords relevant to the New Lifetime Position

HEADER KEYWORD	VALUE	DESCRIPTION
LIFE_ADJ	-1, 1, 2	May now be set to unknown position (-1), original position (1) or new position (2).
PROPAPER	PSA/BOA/WCA/FCA	Proposed Aperture
APMPOS	String	Aperture Mechanism Position
APERXPOS	Floating Point	Aperture X Position. The position of the aperture block mechanism in X as recorded by telemetry
APERYPOS	Floating Point	Aperture Y Position. The position of the aperture block mechanism in Y as recorded by telemetry

COS High Voltage

In this appendix. . .

B.1 COS High Voltage History / 154

B.1 COS High Voltage History

The high voltage on the two FUV detector segments has been adjusted numerous times since launch in order to optimize the performance. [Table B.1](#) lists the nominal high voltage values used since COS installation in 2009.

The initial values used were identical to those used during ground testing. After early on-orbit tests showed that the gain of the MCPs was higher than expected on both segments, the voltages were lowered to return the gain to the prelaunch values. As described in [Section 1.2.1](#), exposure to photon events lowered the gain in the spectral region at the original lifetime position, so the voltage was raised in March 2011 (Segment B) and March 2012 (Segment A) to keep the gain high enough to minimize throughput loss.

The voltage was adjusted once more when the spectrum was moved to the second lifetime position in July 2012, since that location has not yet been affected by gain sag. The new voltages are lower than those used in 2009 in order to keep the rate of gain sag as low as possible while ensuring optimal detector performance. The voltage will be adjusted as needed in the future in order to keep the gain in the spectral region at acceptable levels.

In addition to the nominal voltages listed in the table, other values have been used in a number of calibration activities, so some data files in the archive have nonstandard voltage levels. For updates on the FUV high voltage, users should refer to:

http://www.stsci.edu/hst/cos/performance/high_voltage/.

Table B.1: COS FUV Detector Nominal High Voltage History

DATE	HV LEVEL COUNTS (SegA / SegB)	HV LEVEL VOLTS (SegA / SegB)	DESCRIPTION
05/11/2009	178 175	-5293 -5246	Initial on-orbit values
08/12/2009	169 167	-5152 -5120	Lowered both segments to decrease modal gain
03/08/2011	169 175	-5152 -5246	Increased Segment B to compensate for gain sag
03/26/2012	178 175	-5293 -5246	Increased Segment A to compensate for gain sag
07/23/2012	167 163	-5120 -5058	Initial values at Second Lifetime Position