

---

# Observing With The Green Bank Telescope

---



by GBT Scientific Staff

February 14, 2017

Version 4.0

This guide provides essential information for the preparation of  
Scheduling Blocks for observations with the Green Bank Telescope.

## Authors, Contributors, and Editors

This Guide is the product of many of the GBT Scientific Staff, with major contributions from:

Dana Balsler

Jim Braatz

David Frayer

Frank Ghigo

Amanda Kepley

Adam Kobelski

Karen O'Neil

Glen Langston

Ryan Lynch

Ron Maddalena

Brian Mason

Toney Minter

Dan Perera

Richard Prestage

Scott Ransom

# Contents

<b>1</b>	<b>How To Use This Manual</b>	<b>1</b>
<b>2</b>	<b>The GBT Observing Process</b>	<b>3</b>
2.1	Overview Of The Green Bank Telescope . . . . .	3
2.1.1	Main Features of the GBT . . . . .	4
2.1.2	National Radio Quiet Zone . . . . .	5
2.1.3	Front Ends . . . . .	6
2.1.3.1	Prime focus receivers . . . . .	6
2.1.3.2	Gregorian focus receivers . . . . .	7
2.1.3.3	The MUSTANG Receiver . . . . .	7
2.1.4	Backends . . . . .	7
2.1.4.1	Digital Continuum Receiver (DCR) . . . . .	7
2.1.4.2	Caltech Continuum Backend (CCB) . . . . .	7
2.1.4.3	VEGAS . . . . .	7
2.1.4.4	GUPPI . . . . .	8
2.1.4.5	“VLBI” . . . . .	8
2.1.4.6	Radar . . . . .	8
2.1.5	Polarization Measurements . . . . .	8
2.2	The GBT Observing Process . . . . .	9
<b>3</b>	<b>Introduction to the Dynamic Scheduling System</b>	<b>11</b>
3.1	Overview of the DSS . . . . .	11
3.2	DSS Terminology . . . . .	11
3.3	Controlling the Scheduling of a Project . . . . .	12
3.4	Canonical Target Positions . . . . .	13
3.5	Contact Information and Project Notes . . . . .	13
3.6	The DSS Software . . . . .	14
3.6.1	The DSS Home Page . . . . .	14
3.6.2	The DSS Project Page . . . . .	15

3.7	Responsibilities . . . . .	16
3.8	Remote Observing . . . . .	17
3.9	The Daily Schedule . . . . .	17
3.10	Backup Projects . . . . .	17
3.11	Session Types . . . . .	17
3.12	Projects that can Tolerate Degraded Weather . . . . .	18
3.13	Other DSS Control Parameters . . . . .	18
<b>4</b>	<b>Introduction To Astrid</b>	<b>19</b>
4.1	What Is Astrid? . . . . .	19
4.2	How To Start Astrid . . . . .	20
4.2.1	Astrid Modes . . . . .	20
4.3	Astrid GUI Composition . . . . .	21
4.3.1	Resizing Astrid Display Areas . . . . .	21
4.3.2	Application . . . . .	21
4.3.3	Application Component Tabs . . . . .	21
4.3.4	Drop-down Menus . . . . .	22
4.3.5	Toolbar . . . . .	22
4.3.6	Logs . . . . .	22
4.3.7	Command Console . . . . .	22
4.3.8	State . . . . .	23
4.3.9	Observation Control Buttons . . . . .	23
4.4	The Observation Management Tab . . . . .	24
4.4.1	The Edit Subtab . . . . .	24
4.4.1.1	Project Name and List of Scheduling Blocks . . . . .	24
4.4.1.2	Editor . . . . .	25
4.4.1.3	Adding and Editing Scheduling Blocks in the Database . . . . .	26
4.4.1.4	Validator . . . . .	26
4.4.1.5	The Observing Log . . . . .	26
4.4.2	The Run Subtab . . . . .	27
4.4.2.1	Header Information Area . . . . .	27
4.4.2.2	Submitting An SB to the Run Queue . . . . .	28
4.4.2.3	The Run Queue and Session History . . . . .	28
4.4.2.4	The Observing Log . . . . .	28
4.5	The Data Display Tab . . . . .	28
4.6	The GbtStatus Tab . . . . .	29
4.6.1	General Status . . . . .	29

4.6.2	Telescope Status . . . . .	30
4.6.3	Scan and Source Status . . . . .	31
4.6.4	Configuration Status . . . . .	32
4.6.5	Weather Status . . . . .	32
4.6.6	Time Delay Status . . . . .	32
4.6.7	VEGAS Status . . . . .	32
4.6.8	IF Status . . . . .	33
<b>5</b>	<b>Near–Real–Time Data Displays and CLEO Utilities</b>	<b>35</b>
5.1	The Astrid Data Display Tab . . . . .	35
5.1.1	Working Online . . . . .	35
5.1.2	Working Offline . . . . .	35
5.1.3	Pointing and Focus Data Display . . . . .	36
5.1.3.1	Fitting Acceptance Options . . . . .	37
5.1.3.2	Data Processing Options . . . . .	37
5.1.3.3	Heuristics Options . . . . .	38
5.1.3.4	Send Corrections . . . . .	38
5.1.4	OOF Data Display . . . . .	39
5.1.4.1	AutoOOF Solutions . . . . .	40
5.1.4.2	AutoOOF Raw Data . . . . .	40
5.1.4.3	Selecting the Zernike order to fit . . . . .	41
5.1.4.4	Sending a Solution to the Active Surface . . . . .	42
5.1.4.5	OOF Processing Options . . . . .	42
5.1.5	Continuum Data Display . . . . .	43
5.1.6	Spectral Data Display . . . . .	43
5.1.7	The Data Display Plotting Panel Toolbar . . . . .	44
5.1.8	Use of Plotting Capabilities . . . . .	45
5.2	The CLEO Utilities . . . . .	46
5.2.1	Starting CLEO . . . . .	46
5.2.2	Talk and Draw . . . . .	47
5.2.3	Scheduler and Skyview . . . . .	47
5.2.4	Status . . . . .	48
5.2.5	Messages . . . . .	48
5.3	VEGAS Monitoring Tools . . . . .	49
5.3.1	The VEGAS Data Display . . . . .	49
5.3.2	The VEGAS CLEO screen . . . . .	50
5.3.3	VEGAS Data Monitor . . . . .	51

<b>6</b>	<b>Scheduling Blocks</b>	<b>53</b>
6.1	Making A Scheduling Block . . . . .	54
6.1.1	Components of a Scheduling Block . . . . .	54
6.2	Configuration of the GBT IF System . . . . .	54
6.2.1	Overview . . . . .	54
6.2.2	Defining and Executing A Configuration . . . . .	54
6.2.3	Basic Configuration Syntax . . . . .	55
6.2.4	Example Configurations . . . . .	56
	Continuum Observations . . . . .	56
	Spectral Line, Frequency Switching Observations . . . . .	57
	Multiple Spectral Lines, Total Power Observations . . . . .	58
	Multiple Spectral Lines, Multi-beam, Total Power Observations . . . . .	59
	Multiple Spectral Lines, KFPA Observations . . . . .	60
	Advanced Use of the Restfreq Keyword . . . . .	61
6.2.5	Configuration Keywords . . . . .	63
6.2.5.1	Keywords That Must Always Be Present . . . . .	63
6.2.5.2	Keywords With Default Values . . . . .	66
6.2.5.3	Backend and Receiver Dependent Keywords . . . . .	67
6.2.5.4	Expert Keywords . . . . .	69
6.2.6	Resetting The Configuration . . . . .	70
6.3	Catalogs . . . . .	71
6.3.1	Getting Your Catalog Into Astrid . . . . .	71
6.3.2	The Format of the Catalog . . . . .	72
	Catalog Header Keywords . . . . .	72
6.3.2.1	SPHERICAL format Examples . . . . .	74
6.3.3	Standard Catalogs . . . . .	76
6.3.4	Catalog Functions . . . . .	77
	c.keys() . . . . .	77
	c['sourcename']['keyword'] . . . . .	77
6.3.5	EPHEMERIS : Tables for moving objects . . . . .	78
6.3.5.1	Example Ephemeris Catalogs . . . . .	79
6.3.5.2	Comets . . . . .	80
6.3.6	NNTLE : Tracking Earth satellites . . . . .	84
6.4	Scan Types . . . . .	85
6.4.1	Utility Scans . . . . .	85
6.4.1.1	AutoPeakFocus . . . . .	87
6.4.1.2	AutoPeak . . . . .	89

6.4.1.3	AutoFocus	89
6.4.1.4	AutoOOF	89
6.4.1.5	Focus	91
6.4.1.6	Peak	91
6.4.1.7	Tip	92
6.4.1.8	Slew	93
6.4.1.9	Balance	93
6.4.1.10	BalanceOnOff	94
6.4.2	Observing Scans	94
6.4.2.1	Track	95
6.4.2.2	OnOff	96
6.4.2.3	OffOn	96
6.4.2.4	OnOffSameHA	97
6.4.2.5	Nod	97
6.4.2.6	SubBeamNod	98
6.4.3	Mapping Scans	99
6.4.3.1	RALongMap	100
6.4.3.2	RALongMapWithReference	101
6.4.3.3	DecLatMap	102
6.4.3.4	DecLatMapWithReference	103
6.4.3.5	PointMap	103
6.4.3.6	PointMapWithReference	105
6.4.3.7	Daisy	105
6.5	Utility Functions	108
6.5.1	Annotation	108
6.5.2	Break	108
6.5.3	Comment	109
6.5.4	GetUTC	109
6.5.5	GetLST	109
6.5.6	Now	110
6.5.7	WaitFor	110
6.5.8	ChangeAttenuation	111
6.6	Scheduling Block Objects	112
6.6.1	Location Object	112
6.6.2	Offset Object	113
6.6.3	Horizon Object	114
6.6.4	Time Object	115

6.7	Example Scheduling Blocks . . . . .	116
6.7.1	Frequency Switched Observations Looping Through a List of Sources . . . . .	116
6.7.2	Position Switched Observations Repeatedly Observing the Same Source . . . . .	117
6.7.3	Position Switched Observations of Several Sources and Using the Horizon Object . . . . .	118
6.7.4	Frequency Switched On-The-Fly Mapping . . . . .	119
6.8	What Makes a Good Scheduling Block . . . . .	120
<b>7</b>	<b>Observing Tactics and Recommendations</b>	<b>121</b>
7.1	Active Surface (AS) Strategies . . . . .	121
7.2	AutoOOF Strategy . . . . .	121
7.2.1	How long does the solution remain valid? . . . . .	122
7.3	Balancing Strategies . . . . .	123
7.3.1	Balancing VEGAS . . . . .	124
7.4	Calibration Strategies . . . . .	124
7.5	Pointing and Focusing Strategies . . . . .	125
7.6	Spectral Configurations . . . . .	126
7.7	Mapping Strategies . . . . .	126
7.8	High Frequency Observing Strategies . . . . .	126
7.9	Observing Strategies For Strong Continuum Sources . . . . .	127
<b>8</b>	<b>VEGAS</b>	<b>129</b>
8.1	Overview . . . . .	129
8.2	Data Rates . . . . .	130
8.3	IF Configuration . . . . .	131
8.4	Blanking . . . . .	131
8.5	Monitoring VEGAS observations . . . . .	134
8.6	The Online Filler and Filling VEGAS data using SDFITS . . . . .	134
8.6.1	The Online Filler . . . . .	134
8.6.2	Filling Offline . . . . .	135
8.7	Instrumental Features and their Cure . . . . .	135
8.7.1	The Spike . . . . .	136
8.7.2	The Spurs . . . . .	136
8.8	Known Bugs and Features . . . . .	137
8.8.1	vegasdisplay is not updating, or is running very slowly . . . . .	137
8.8.2	Data is not filling . . . . .	137
8.8.3	There is a “square wave” and/or divot in my VEGASDM display . . . . .	137



<b>9 Pulsar Observing with GUPPI</b>	<b>139</b>
9.1 Summary . . . . .	139
9.2 Details of GUPPI Configuration . . . . .	140
9.3 Balancing and Checking Levels . . . . .	142
9.4 Taking Data . . . . .	145
9.5 Monitoring GUPPI During an Observation . . . . .	146
9.5.1 Monitoring Incoherent Data Taking . . . . .	146
9.5.2 Monitoring Coherent Data Taking . . . . .	146
9.6 Processing Your Data . . . . .	148
9.7 Tips and Tricks . . . . .	148
9.8 Warnings . . . . .	149
9.9 More Information . . . . .	149
<b>10 K-band (18-27.5 GHz) Focal Plane Array (KFPA)</b>	<b>151</b>
10.1 Configuration . . . . .	152
10.1.1 Beam selection with VEGAS banks . . . . .	152
10.1.2 Instantaneous Bandwidth . . . . .	152
10.2 Calibration . . . . .	152
10.2.1 Realigning the Noise Diodes . . . . .	153
<b>11 The 4mm (68-92 GHz) Receiver</b>	<b>155</b>
11.1 Overview . . . . .	155
11.2 Configuration . . . . .	155
11.3 Observing . . . . .	156
11.3.1 CalSeq . . . . .	157
11.3.2 Pointing and Focus . . . . .	158
11.3.3 AutoOOF Thermal Corrections . . . . .	159
11.4 Calibration and Data Reduction . . . . .	159
11.5 Web Documentation . . . . .	161
<b>12 The CalTech Continuum Backend (CCB)</b>	<b>163</b>
12.1 Observing with the CCB . . . . .	163
12.1.1 Configuration . . . . .	164
12.1.2 Pointing & Focus . . . . .	165
12.1.3 Observing Modes & Scheduling Blocks . . . . .	165
12.1.4 Calibration . . . . .	166
12.1.5 Online Data Analysis . . . . .	166
12.2 Performance . . . . .	170
12.3 Differences Between the CCB/Ka System and other GBT Systems . . . . .	170

<b>13 VLBI Observing using the GBT</b>	<b>171</b>
13.1 Proposals . . . . .	171
13.2 VLBA-compatible recording . . . . .	171
13.3 Schedule Preparation . . . . .	172
13.4 Special considerations when using the GBT . . . . .	172
13.5 Available Receivers and Bands . . . . .	173
13.6 Include Pointing and Focus Checks . . . . .	174
13.6.1 3mm Receiver (68-92 GHz) calibration . . . . .	174
13.7 Weather Considerations . . . . .	175
13.8 Telescope Move times and limits . . . . .	175
13.9 High Frequency (40-90 GHz) active surface considerations . . . . .	175
13.10GBT Coordinates . . . . .	176
<b>14 Solar System Radar with the GBT</b>	<b>177</b>
14.1 Introduction . . . . .	177
14.2 Data Acquisition Backends . . . . .	177
14.3 GBT Scheduling Blocks . . . . .	178
14.4 Tracking moving objects . . . . .	179
<b>15 Radio Frequency Interference</b>	<b>181</b>
Mitigation of known RFI signals . . . . .	181
<b>16 Weather Effects on Observations</b>	<b>183</b>
16.1 Winds . . . . .	183
16.2 Time of Day . . . . .	184
16.3 Atmospheric Opacities . . . . .	185
16.4 GBT Weather Restrictions . . . . .	187
16.4.1 Winds . . . . .	187
16.4.2 Snow . . . . .	187
16.4.3 Ice . . . . .	187
16.4.4 Temperature . . . . .	187
16.4.5 Feed Blowers . . . . .	187
<b>17 Computing</b>	<b>189</b>
17.0.1 Accounts . . . . .	189
17.0.2 Computing Facilities . . . . .	189
17.0.3 Remote Access and VNC . . . . .	189
<b>18 Remote Observing With The GBT</b>	<b>191</b>
18.1 Remote Observing Guidelines for Approved Projects . . . . .	191

<b>19 Planning Your Observations And Travel</b>	<b>193</b>
19.1 Preparing for Your Observations . . . . .	193
19.2 Travel Support . . . . .	193
19.3 Trains, Planes and Automobiles . . . . .	194
19.4 Housing . . . . .	194
19.5 Getting To Green Bank . . . . .	194
19.5.1 Where is Green Bank? . . . . .	194
19.5.2 Directions to Green Bank . . . . .	194
19.5.2.1 Beware of GPS!! . . . . .	194
19.5.2.2 Pittsburgh to Green Bank . . . . .	194
19.5.2.3 Washington Dulles or National to Green Bank . . . . .	196
19.5.2.4 Charlottesville to Green Bank . . . . .	196
19.5.2.5 Roanoke to Green Bank . . . . .	196
19.5.3 Once You Are in Green Bank . . . . .	196
<b>20 After Your Observations</b>	<b>199</b>
20.1 Taking your data home . . . . .	199
20.2 Installing GBTIDL . . . . .	199
20.3 Keep Your Contact Person Informed . . . . .	199
20.4 Press Releases and News-worthy Items . . . . .	199
20.5 Publishing Your Results . . . . .	200
<b>A GBT IF System</b>	<b>201</b>
A.1 From the Receiver to the IF Rack . . . . .	202
A.2 From the Converter Rack to the Backend . . . . .	203
A.3 KFPA Combined IF . . . . .	203
<b>B Introduction to Spectral Windows</b>	<b>205</b>
B.1 Array Receiver Spectral Windows . . . . .	206
<b>C Usage of vlow and vhigh</b>	<b>207</b>
<b>D Advanced Functions</b>	<b>209</b>
D.1 General Functions . . . . .	209
D.1.1 GetValue() . . . . .	209
D.1.2 SetValues() . . . . .	210
D.1.3 DefineScan() . . . . .	210
D.1.4 GetCurrentLocation() . . . . .	211
D.1.5 SetSourceVelocity() . . . . .	211
D.2 Specialty Scan Types . . . . .	212
D.2.1 Spider . . . . .	212
D.2.2 Z17 . . . . .	213

<b>E</b>	<b>Advanced Use of the Balance() Command</b>	<b>215</b>
<b>F</b>	<b>Backup Projects</b>	<b>217</b>
<b>G</b>	<b>DSS Control Parameters</b>	<b>219</b>
<b>H</b>	<b>VEGAS Reference</b>	<b>223</b>
	H.0.1 VEGAS ADC (MMCM, OGP and INL) Calibration . . . . .	223
	H.0.2 IF Levels and Balancing . . . . .	223
	H.0.2.1 vegas_status . . . . .	224
	H.1 Baseline Performance . . . . .	224
	H.2 VEGAS and Deep Integrations . . . . .	227
	H.3 The Spurs . . . . .	227

# List of Figures

2.1	The parent parabola and off-axis design of the GBT . . . . .	3
2.2	National Radio Quiet Zone . . . . .	5
3.1	A sample DSS home page . . . . .	14
3.2	A sample DSS project page . . . . .	15
4.1	Astrid splash screen . . . . .	20
4.2	Astrid startup pop-up window . . . . .	20
4.3	Components of the Astrid GUI . . . . .	21
4.4	Astrid Observation Management/Edit Subtab . . . . .	24
4.5	Selecting your project using the drop-down menu . . . . .	25
4.6	Astrid Observation Management/Run Subtab . . . . .	27
4.7	Astrid Status Tab (top) . . . . .	29
4.8	Astrid Status Tab (bottom) . . . . .	31
5.1	The Pointing subtab of the Astrid Data Display . . . . .	36
5.2	The Focus subtab of the Astrid Data Display . . . . .	36
5.3	Pointing and Focus acceptance pop-up . . . . .	37
5.4	Pointing and Focus change fitting pop-up . . . . .	37
5.5	Pointing and Focus heuristics pop-up . . . . .	38
5.6	Send Corrections pop-up . . . . .	38
5.7	The OOF subtab of the Astrid Data Display . . . . .	39
5.8	Comparison of AutoOOF solutions . . . . .	40
5.9	Comparison of AutoOOF raw data . . . . .	41
5.10	AutoOOF fitted beam maps indicating a bad solution . . . . .	42
5.11	OOF Processing Options . . . . .	42
5.12	The Continuum subtab of the Astrid Data Display . . . . .	43
5.13	The Spectral Line subtab of the Astrid Data Display . . . . .	44
5.14	The Cleo Launcher . . . . .	46

5.15	CLEO Talk and Draw . . . . .	46
5.16	CLEO Scheduler & Skyview . . . . .	47
5.17	CLEO Status . . . . .	48
5.18	CLEO Messages . . . . .	48
5.19	The VEGAS real-time data display . . . . .	49
6.1	The JPL Horizons website . . . . .	80
6.2	Selecting quantities to generate an ephemeris. . . . .	81
6.3	RALongMap() GBT Antenna Trajectory . . . . .	101
6.4	DecLatMap() GBT Antenna Trajectory . . . . .	102
6.5	PointMap() GBT Antenna Trajectory . . . . .	104
6.6	Daisy map trajectory . . . . .	106
6.7	Closed Daisy() GBT Antenna Trajectory split into multiple scans . . . . .	107
8.1	VEGAS setups . . . . .	131
8.2	VEGAS spurs . . . . .	136
9.1	The GUPPI ADC histogram display . . . . .	143
9.2	The GUPPI monitor display . . . . .	144
9.3	The GUPPI status display . . . . .	147
10.1	Orientation of the KFPA feeds on the GBT . . . . .	151
10.2	KFPA azimuth pointing scans before and after noise diode realignment . . . . .	153
12.1	Data from a CCB, beamswitched OTF-NOD . . . . .	166
12.2	CCB data from an OTF-NOD observation of a bright source . . . . .	168
12.3	CCB OTF-NOD data on a bright source under marginal conditions . . . . .	168
12.4	CCB OTF-NOD measurement of a weak (mJy-level) source under good conditions . . . . .	169
12.5	The same weak-source data with the individual integrations binned into 0.5 second bins . . . . .	169
16.1	Wind speed statistics . . . . .	183
16.2	Night-time for the GBT . . . . .	184
16.3	Typical system temperatures . . . . .	185
16.4	Opacity statistics . . . . .	186
19.1	Directions to Green Bank . . . . .	195
19.2	Green Bank Site Map . . . . .	197
A.1	GBT IF system routing . . . . .	201
A.2	GBT IF system flow chart . . . . .	202

A.3	KFPA IF system chart . . . . .	203
D.1	Spider() GBT Antenna Trajectory . . . . .	213
D.2	Z17() GBT Antenna Trajectory . . . . .	213
H.1	The VEGAS shared memory screen . . . . .	225
H.2	VEGAS bandpass for the four families of VEGAS modes . . . . .	226
H.3	A deep integration with VEGAS . . . . .	228
H.4	Data from Figure 8.2 after the high pass filter . . . . .	229





# List of Tables

2.1	GBT Telescope Specifications . . . . .	4
2.2	GBT receiver properties . . . . .	6
4.1	Astrid online and offline modes . . . . .	20
6.1	GBT receivers and frequencies . . . . .	63
6.2	GBT backends . . . . .	63
6.3	Allowed bandwidths . . . . .	64
6.4	Available catalogs . . . . .	76
6.5	Utility Scan Types . . . . .	86
6.6	Peak and Focus recommendations . . . . .	86
6.7	Utility Scan Types . . . . .	94
6.8	Mapping Scan Types . . . . .	99
7.1	Observing wind default limits and Point/Focus strategies . . . . .	125
8.1	VEGAS Modes . . . . .	130
8.2	Minimum recommended switching periods for VEGAS observations using a noise diode. . . . .	132
8.3	Minimum recommended switching periods for VEGAS observations not using a noise diode. . . . .	133
10.1	Nominal beam offsets of the KFPA . . . . .	151
11.1	4mm Channel Definitions . . . . .	160
11.2	Effective Cold Load Temperature for W-band . . . . .	161
12.1	CCB Port labels and the astronomical quantities they measure . . . . .	163
13.1	VLBA bands and GBT receivers . . . . .	173
13.2	GBT pointing and focus checks with VLBA observations . . . . .	174
14.1	Radar data acquisition backends . . . . .	177
H.1	$\nu_{V,c}$ for each VEGAS mode . . . . .	228



# List of Scripts

6.1	A simple Scheduling Block . . . . .	53
6.2	Basic configuration syntax . . . . .	55
6.3	Loading a configuration with <code>execfile()</code> . . . . .	55
6.4	Explicitly defining the configuration in a Scheduling Block . . . . .	55
6.5	An example continuum configuration . . . . .	56
6.6	An example frequency switched, spectral line configuration . . . . .	57
6.7	An example total power, spectral line configuration . . . . .	58
6.8	An example total power, spectral line configuration for a multi-beam receiver . . . . .	59
6.9	An example total power, spectral line configuration for the KFPA . . . . .	60
6.10	An example showing advanced use of the <code>restfreq</code> keyword . . . . .	61
6.11	A simple catalog . . . . .	71
6.12	Loading catalogs into Scheduling Blocks . . . . .	72
6.13	Catalog header keywords . . . . .	72
6.14	Catalog equinox header keyword . . . . .	73
6.15	A simple spherical format catalog . . . . .	74
6.16	Catalog default header keywords and columns . . . . .	74
6.17	Catalog velocity column . . . . .	75
6.18	Catalog using Galactic coordinates . . . . .	75
6.19	Catalog user-defined keywords . . . . .	75
6.20	Loading standards catalogs into Astrid . . . . .	76
6.21	Catalog <code>c.keys()</code> function . . . . .	77
6.22	Using a catalog function to retrieve the declination of a source . . . . .	77
6.23	Catalog user-defined columns . . . . .	78
6.24	Retrieving information from Catalog user-defined columns . . . . .	78
6.25	A simple ephemeris catalog . . . . .	79
6.26	A ephemeris catalog with coordinate rates . . . . .	79
6.27	A ephemeris catalog for tracking a satellite . . . . .	79
6.28	JPL ephemeris file . . . . .	82
6.29	<code>jpl2astrid</code> usage . . . . .	83

6.30 Astrid catalog generated by jpl2astrid . . . . .	83
6.31 An example NNTLE format Catalog . . . . .	84
6.32 An example NNTLE format Catalog using the “FILE” keyword . . . . .	84
6.33 An example NNTLE format Catalog using the “URL” keyword . . . . .	84
6.34 AutoPeakFocus examples . . . . .	88
6.35 AutoOOF example usage . . . . .	90
6.36 AutoOOF example using DCR with the Ka-band 26–40 GHz receiver. . . . .	90
6.37 AutoOOF example using MUSTANG. . . . .	90
6.38 Focus() example. . . . .	91
6.39 Peak() example. . . . .	92
6.40 Tip() example. . . . .	92
6.41 Slew() example. . . . .	93
6.42 Balance() example. . . . .	93
6.43 BalanceOnOff() example. . . . .	94
6.44 Track() example using startTime=Horizon(). . . . .	95
6.45 Track() example using a Horizon object in startTime and stopTime. . . . .	95
6.46 Examples of the Track() scan function. . . . .	96
6.47 OnOff() example. . . . .	96
6.48 OffOn() example. . . . .	96
6.49 OnOffSameHA() example. . . . .	97
6.50 Nod() example. . . . .	97
6.51 SubBeamNod() example. . . . .	98
6.52 RALongMap() example. . . . .	100
6.53 RALongMapWithReference() example. . . . .	101
6.54 DecLatMap() example. . . . .	102
6.55 DecLatMapWithReference() example. . . . .	103
6.56 PointMap() example. . . . .	104
6.57 PointMapWithReference() example. . . . .	105
6.58 Daisy() example. . . . .	107
6.59 Annotation() example. . . . .	108
6.60 Break() example. . . . .	108
6.61 Comment() example. . . . .	109
6.62 GetUTC() example. . . . .	109
6.63 GetUTC() example. . . . .	109
6.64 Now() example. . . . .	110
6.65 WaitFor() example. . . . .	110
6.66 ChangeAttenuation() example. . . . .	111

6.67	Specifying Location Objects. . . . .	112
6.68	Adding Offset Objects. . . . .	113
6.69	Horizon Objects. . . . .	114
6.70	Time Objects. . . . .	115
6.71	The catalog used with SB examples . . . . .	116
6.72	SB example 1 – Frequency switched observations looping through a list of sources. . . . .	116
6.73	SB example 2 – Position switched observations repeatedly observing the same source. . . . .	117
6.74	SB example 3 – Position switched observations of several sources using Horizon(). . . . .	118
6.75	SB example 4 – Frequency switched On-The-Fly mapping. . . . .	119
10.1	KFPA calibration SB. . . . .	153
13.1	pointing and focusing with the GBT . . . . .	174
14.1	Example SB for radar observations . . . . .	178
14.2	Example ephemeris file for an asteroid . . . . .	179
B.1	Multi-beam restfreq syntax example 1 . . . . .	206
B.2	Multi-beam restfreq syntax example 2 . . . . .	206
B.3	Multi-beam restfreq syntax example 3 . . . . .	206
B.4	Multi-beam restfreq syntax example 4 . . . . .	206
B.5	Multi-beam restfreq syntax example 5 . . . . .	206
B.6	Multi-beam restfreq syntax example 6 . . . . .	206
D.1	GetValue() example. . . . .	209
D.2	SetValues() example. . . . .	210
D.3	DefineScan() example. . . . .	210
D.4	GetCurrentLocation() example. . . . .	211
D.5	SetSourceVelocity() example. . . . .	211
D.6	Spider() example. . . . .	212
D.7	Z17() example. . . . .	213
E.1	Advanced Balance() example. . . . .	215



# Chapter 1

## How To Use This Manual

This document provides the necessary information to be able to perform successful observations with the Green Bank Telescope (GBT).

- In Chapter 2 we briefly outline the features of the GBT and the general observing process.
- in Chapter 3 you are introduced to the Dynamic Scheduling System.
- In Chapters 4 and 5 we provide an introduction to the Astrid observing interface and other observing applications.
- In Chapter 6 we provide example Scheduling Blocks (SBs) that can be used in Astrid. We also provide detailed descriptions of the contents of SBs.
- In Chapter 7 we provide information on the strategies that should be used and advanced techniques for observing with the GBT.
- Later chapters give basics of observing with various specific instruments: Chapter 8 for spectral line observing with VEGAS, Chapter 9 for pulsar observing with GUPPI, Chapter 10 for observing with the KFPA, Chapter 11 for observing with the 4mm (68-92 GHz) receiver, Chapter 12 for continuum observing with the CCB, Chapter 13 for VLBI observing with the GBT, and Chapter 14 for solar system radar with the GBT.
- In Chapter 15 we provide the locations of where to find more information about RFI.
- In Chapter 16 is a discussion of the effect of weather conditions on observing.
- In Chapters 17 and 18 we provide information on computing and remote observing.
- In Chapter 19 we provide information on what happens before your observations and directions on getting to Green Bank.
- In Chapter 20 we provide information on how to take your data home with you and where to obtain the GBT spectral line data reduction package, GBTIDL.
- Additional information and special topics are covered in the Appendices.

New users should read Chapters 2, 3, 4, 5, 6, 7, and 17 in their entirety. They should also read the remaining Chapters as needed.





## Chapter 2

# The GBT Observing Process

## 2.1 Overview Of The Green Bank Telescope

The 100 meter Green Bank Telescope (GBT) is intended to address a very broad range of astronomical problems at radio wavelengths and consequently has an unusual and unique design. Unlike conventional telescopes that have feed legs projecting over the middle of the surface, the GBT's aperture is unblocked so that incoming radiation meets the surface directly. This increases the useful area of the telescope and reduces reflection and diffraction, which ordinarily complicate a telescope's pattern of response to the sky. To keep the aperture unblocked, the design incorporates an off-axis feed arm that cradles the dish and projects upward at one edge. This requires that the figure of the telescope surface be asymmetrical. To make a projected circular aperture 100 meters in diameter, the dish is actually a 100 by 110 meter section of a conventional, rotationally symmetric 208 meter figure, beginning four meters outward from the vertex of the hypothetical parent structure (see Figure 2.1). The GBT's lack of circular symmetry greatly increases the complexity of its design and construction.

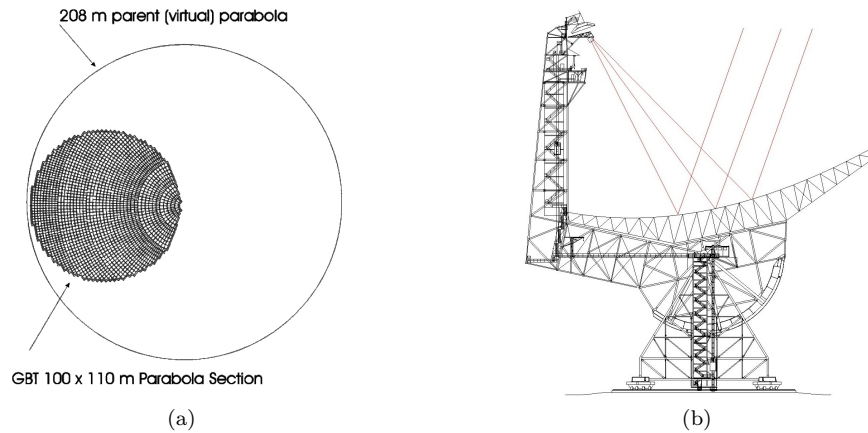


Figure 2.1: The parent parabola (2.1a) and off-axis design (2.1b) of the GBT.

To maintain precise surface figures and pointing accuracy at high frequencies the telescope is equipped with a complex Active Surface (AS). At higher frequencies gravity distorts the surface figure of the telescope to unacceptable levels. Temperature variations and wind can also deform the figure of the dish. To compensate for these distortions, the surface of the GBT is “active” i.e. it is made up of 2008 independent panels and each of these panels are mounted on actuators at the corners, which can raise and lower the panels to adjust the shape of the dish's surface.

### 2.1.1 Main Features of the GBT

- **Fully steerable antenna:**  $+5^\circ$  to  $+90^\circ$  elevation range ( $-46.5^\circ$  to  $+90^\circ$  declination); 85% coverage of the celestial sphere. Note that observing at elevations  $>86^\circ$  (or  $80^\circ$  during extremely cold weather) may fail due to the high azimuth rates required.
- **Unblocked aperture:** Reduces sidelobes, RFI, and spectral standing waves.
- **Active surface:** Compensates for gravitational and thermal distortions.
- **Frequency coverage of 100 MHz to 115+ GHz:** 3 orders of magnitude of frequency coverage for maximum scientific flexibility.
- **Location in the National Radio Quiet Zone:** Comparatively low RFI environment (See Figure 2.2).
- **Dynamic Scheduling:** Matching scientific programs to the required weather conditions.

Table 2.1: GBT Telescope Specifications.

Location	Green Bank, West Virginia, USA
Coordinates	Longitude: $79^\circ 50' 23.406''$ West (NAD83) Latitude: $38^\circ 25' 59.236''$ North (NAD83) Track Elevation: 807.43 m (NAVD88)
Optics	110 m $\times$ 100 m unblocked section of a 208 m parent paraboloid Offaxis feed arm
Telescope Diameter	100 m (effective)
Available Foci	Prime and Gregorian f/D (prime) = 0.29 (referred to 208 m parent parabola) f/D (prime) = 0.6 (referred to 100 m effective parabola) f/D (Gregorian) = 1.9 (referred to 100 m effective aperture)
Receiver mounts	Prime: Retractable boom with Focus-Rotation Mount Gregorian: Rotating turret with 8 receiver bays
Subreflector	8-m reflector with Stewart Platform (6 degrees of freedom)
Main reflector	2004 actuated panels (2209 actuators) Average intra-panel RMS 68 $\mu\text{m}$
FWHM Beamwidth	Gregorian Feed: $\sim 12.60' / f_{\text{GHz}}$ Prime Focus: $\sim 13.01' / f_{\text{GHz}}$
Elevation Limits	Lower limit: $5^\circ$ Upper limit: $90^\circ$
Declination Range	Lower limit: $-46.5^\circ$ Upper limit: $90^\circ$
Slew Rates	Azimuth: $35.2^\circ / \text{min}$ Elevation: $17.6^\circ / \text{min}$
Surface RMS	Passive surface: 450 $\mu\text{m}$ at $45^\circ$ elevation, worse elsewhere Active surface: $\sim 250 \mu\text{m}$ , under benign night-time conditions
Tracking accuracy ( $\sigma_{tr}$ )	$\sigma_{tr}^2 = \sigma_0^2 + (s/3.5)^4$ $\sigma_0 = \text{night:}1.32'', \text{day:}2.19''; s = \text{wind speed (ms}^{-1}\text{)}$
Pointing accuracy	$5''$ blind pointing accuracy

## 2.1.2 National Radio Quiet Zone

The National Radio Quiet Zone (NRQZ) was established by the Federal Communications Commission (FCC) and by the Interdepartmental Radio Advisory Committee (IRAC) on November 19, 1958 to minimize possible harmful interference to the National Radio Astronomy Observatory (NRAO) in Green Bank, WV and the radio receiving facilities for the United States Navy in Sugar Grove, WV. The NRQZ is bounded by North American Datum of 1983 (NAD83) meridians of longitude at 78d 29m 59.0s W and 80d 29m 59.2s W and latitudes of 37d 30m 0.4s N and 39d 15m 0.4s N, and encloses a land area of approximately 13,000 square miles near the state border between Virginia and West Virginia.

- Further information on the NRQZ can be obtained at
  - <https://science.nrao.edu/facilities/gbt/interference-protection/nrqz/>
- Information on the West Virginia State Code Chapter 37A “Radio Astronomy Zoning Act” (WVRAZ) can be found at
  - <http://www.legis.state.wv.us/WVcode/code.cfm?chap=37a&art=1>

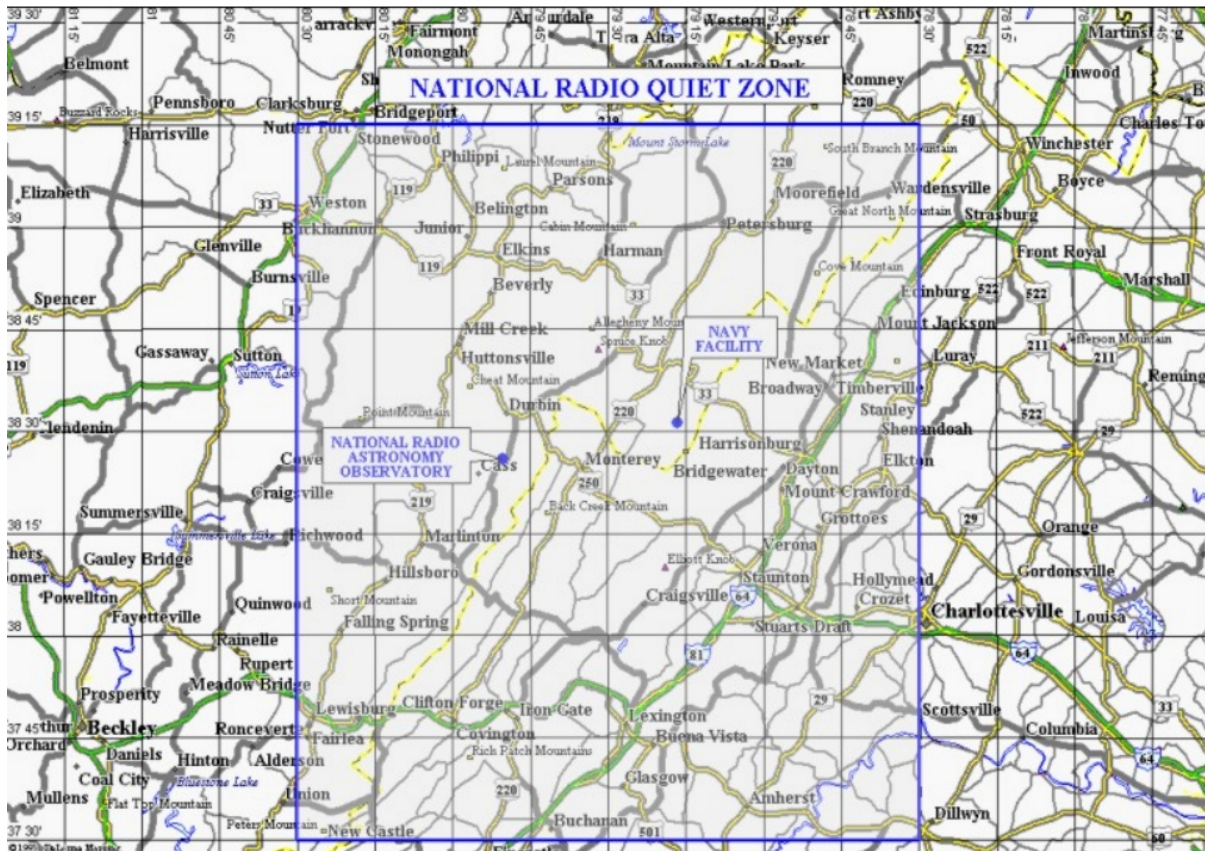


Figure 2.2: The National Radio Quiet Zone.

### 2.1.3 Front Ends

The GBT receivers cover several frequency bands from 0.290 - 50 GHz and 70 - 100 GHz. Table 6.1 lists the properties of the Prime Focus receivers and the Gregorian Focus receivers. System temperatures are derived from lab measurements or from expected receiver performance given reasonable assumptions about spillover and atmospheric contributions.

The GBT Proposer’s Guide (<https://science.nrao.edu/facilities/gbt/proposing/GBTpg.pdf>) provides more information on the GBT receivers.

Table 2.2: Properties of the Prime Focus and Gregorian Focus Receivers.

Name		$\nu$ Range (GHz)	Polarization	Beams	Polns/ Beam	$T_{\text{rec}}$ (K)	$T_{\text{sys}}$ (K)
— Prime Focus Receivers —							
PF1	Rcvr_342	0.290-0.395	Lin/Circ	1	2	12	46
PF1	Rcvr_450	0.385-0.520	Lin/Circ	1	2	22	43
PF1	Rcvr_600	0.510-0.690	Lin/Circ	1	2	12	22
PF1	Rcvr_800	0.680-0.920	Lin/Circ	1	2	21	29
PF2	Rcvr_1070	0.910-1.230	Lin/Circ	1	2	10	17
— Gregorian Focus Receivers —							
L-band	Rcvr1_2	1.15-1.73	Lin/Circ	1	2	6	20
S-band	Rcvr2_3	1.73-2.60	Lin/Circ	1	2	8-12	22
C-band	Rcvr4_6	3.95-8.0	Lin/Circ	1	2	5	18
X-band	Rcvr8_10	8.00-10.1	Circ	1	2	13	27
Ku-band	Rcvr12_18	12.0-15.4	Circ	2	2	14	30
KFPA	RcvrArray18_26	18.0-26.5	Circ	7	2	15-25	30-45
Ka-band	Rcvr26_40 (MM-F1)	26.0-31.0	Lin	2	1	20	35
Ka-band	Rcvr26_40 (MM-F2)	30.5-37.0	Lin	2	1	20	30
Ka-band	Rcvr26_40 (MM-F3)	36.0-39.5	Lin	2	1	20	45
Q-band	Rcvr40_52	38.2-49.8	Circ	2	2	40-70	67-134
W-band	Rcvr68_92 (FL1)	67-74	Lin/Circ	2	2	50	160
W-band	Rcvr68_92 (FL2)	73-80	Lin/Circ	2	2	50	120
W-band	Rcvr68_92 (FL3)	79-86	Lin/Circ	2	2	50	100
W-band	Rcvr68_92 (FL4)	85-92	Lin/Circ	2	2	60	110

#### 2.1.3.1 Prime focus receivers

The Prime focus receivers are mounted in a Focus Rotation Mount (FRM) on a retractable boom. The boom is moved to the prime focus position when prime focus receivers are to be used, and retracted when using Gregorian receivers. The FRM holds one receiver box at a time. Currently there are two receiver boxes, PF1 and PF2. A change from PF1 to PF2 receivers requires a box change, taking about 4 hours and done only during scheduled maintenance days.

The PF1 (0.29 - 0.92 GHz) receiver is divided into 4 frequency bands within the same receiver box. The receivers are cooled Field Effect Transistor (FET) amplifiers. The feeds for the lower three bands are short-backfire dipoles, and the feed for the fourth (680-920MHz) is a corrugated feed horn with an Ortho-Mode Transducer (OMT) polarization splitter. A feed change, required to switch between bands, takes 4 hours and must occur on a maintenance day. The PF2 (0.920 - 1.23 GHz) receiver uses a cooled FET and a corrugated feed horn with the OMT.

### 2.1.3.2 Gregorian focus receivers

The Gregorian receivers are mounted in a rotating turret in a receiver room located at the Gregorian Focus of the telescope. The turret has 8 portals for receiver boxes. Up to 8 receivers can be kept cold and active at all times. Changing between any two Gregorian receivers that are installed in the turret takes about 60-90 seconds.

### 2.1.3.3 The MUSTANG Receiver

The Multiplexed SQUID TES Array at Ninety GHz (MUSTANG) is a multi-pixel bolometer array observing at 80-100 GHz mounted at the Gregorian focus. It is both a receiver and the associated backend. The The Multiplexed SQUID TES Array at Ninety GHz (MUSTANG) receiver must be used at elevations above 30°, due to the design of the cryogenics.

## 2.1.4 Backends

The GBT has two continuum backends: the The Digital Continuum Receiver (DCR) and the CCB. The spectral line backend is VEGAS. Pulsar observations can be done with GUPPI. There is a single dish mode for the Very Long Baseline Array (VLBA) backend that is available for high time-resolution observations. Planetary radar uses a specialized backend.

For more information on GBT backends, please see the “GBT Proposer’s Guide” which is available at <https://science.nrao.edu/facilities/gbt/proposing/GBTpg.pdf>.

### 2.1.4.1 Digital Continuum Receiver (DCR)

The DCR is the GBT’s general purpose continuum backend. It is used both for utility observations such as pointing, focus, and beam-map calibrations, as well such as for continuum astronomical observations including point-source on/off, extended source mapping, etc.

### 2.1.4.2 Caltech Continuum Backend (CCB)

The CCB is a sensitive, wideband backend designed exclusively for use with the GBT Ka-band (26-40 GHz) receiver. It provides a carefully optimized Radio Frequency (RF) (not an Intermediate Frequency (IF)) detector circuits and the capability to beam-switch the receiver rapidly to suppress instrumental gain fluctuations. There are 16 input ports (only 8 can be used at present with the Ka-band receiver), hard-wired to the receiver’s 2 feeds  $\times$  2 polarizations  $\times$  4 frequency sub-bands (26-29.5 , 29.5-33.0; 33.0-36.5; and 36.5-40 GHz). The CCB allows the left and right noise-diodes to be controlled individually to allow for differential or total power calibration. Unlike other GBT backends, the noise-diodes are either on or off for an entire integration (there is no concept of “phase within an integration”). The minimum practical integration period is 5 milliseconds; integration periods longer than 0.1 seconds are not recommended. The maximum practical beam-switching rate is about 4 kHz, limited by the needed 250 $\mu$ s beam-switch blanking time. Switching slower than 1 kHz is not recommended.

### 2.1.4.3 VEGAS

The VErSatile GBT Astronomical Spectrometer (VEGAS) is the spectral line backend for the GBT. It consists of eight independent dual polarization spectrometers (banks) that can be configured in any one of 29 modes and can be used with any receiver except MUSTANG. It provides up to 64 spectral windows as well as wide bandwidths (1000-1500 MHz). See chapter 8 for detailed information on VEGAS.

#### 2.1.4.4 GUPPI

The Green Bank Ultimate Pulsar Processing Instrument (GUPPI) has one hardware mode and many software modes. GUPPI can be used with any receiver with the exception of MUSTANG. Only one polarization would be available for the Ka-band (26-40 GHz) receiver. GUPPI uses 8-bit sampling to dramatically improve dynamic range and RFI resistance. Currently GUPPI can use bandwidths of 100, 200 and 800 MHz with 2 polarizations and full stokes parameters. The minimum integration time is  $40.96\mu s$  using 2048 channels and an 800 MHz bandwidth. See the introduction to GUPPI in Chapter 9.

#### 2.1.4.5 “VLBI”

The GBT supports Very Long Baseline (VLB) observations with a Mark5 VLBA recorder. This recorder can also be used in a “single-dish” mode to make high time-resolution observations. See Chapter 13 for more information.

#### 2.1.4.6 Radar

Planetary radar observations are supported by the Portable Fast Sampler (PFS) and JPL radar backends. See Chapter 14 for more information.

### 2.1.5 Polarization Measurements

Measurement of Polarization and Stokes parameters is possible using VEGAS and GUPPI. This is an “expert user” mode: users should contact their GBT support person or the GBT helpdesk. For an introduction to polarization observations, see “A Heuristic Introduction to Radioastronomical Polarization”, by C. Heiles, ASP Conference Series Vol 278, 2002.

## 2.2 The GBT Observing Process

The following list summarizes the general flow of how GBT observing proceeds. By the time you are reading this document you should have already been through several of the steps.

### Step 1 - Contact your GBT “friend”

Before you observe, you need to prepare for your observations (see Chapter 19) and set up your computing account (see Chapter 17). You will be assigned a scientific contact person (GBT “friend”) whom you should contact well in advance of your observing to determine optimum dates for a visit and ensure the telescope and hardware will be available for the project while you are on site. Your “friend” will help you develop appropriate observing tactics for your proposal (see Chapter 7). They will also help you with any technical questions e.g., dealing with RFI (see Chapter 15), etc. At this time you should review your project page(s) in the Dynamic Scheduling System (DSS) (see Chapter 3) and develop your Scheduling Blocks (SBs) (see Chapter 6).

### Step 2 - Make travel arrangements

If you are an experienced GBT observer, you can observe remotely (see Chapter 18). If you are new to the GBT, you must plan to travel to Green Bank (see Chapter 19) and spend at least a week and preferably two weeks at the site to ensure appropriate weather conditions for the observations (see Chapter 16). You should arrive in Green Bank at least one business day before your observations. This will allow you to meet with the contact scientist and also with the scientific staff person who will be “on call” during your observations (these might be different people). After hands-on experience with observing you will qualify for remote observing.

### Step 3 - Set blackout dates in the DSS

If there are periods of time or dates when you cannot observe, you should indicate these as “blackout dates” in the Dynamic Scheduling System (DSS) web page <https://dss.gb.nrao.edu>. Those visiting Green Bank should use blackout dates to mark the periods of their travel before and after their stay to ensure they are scheduled only when available and ready.

### Step 4 - Wait for scheduling notification

Unless you are running a project which must be fixed to a certain date and time, your observations will be dynamically scheduled. See Chapter 3 for details on how dynamical scheduling is done with the GBT. When your project is scheduled you will receive an e-mail notification indicating the exact time the observing session will start. Notifications go to the project PI and all others designated as observers on the project. Thus you should have prepared your scripts and be ready to observe with 24-36 hours notice.

### Step 5 - 30 minutes before your observation session

If you are present in Green Bank, go to the control room and log into one of the computers. Bring up any programs that you need so that you are prepared when your observation time begins.

If you are observing remotely (see Chapter 18) you should contact the GBT operator through *talk and draw* (see 5.2.2) or call on 304-456-2341 or 304-456-2346. In case of a site phone or power outage, the direct line to the control room is 304-456-3203. You should give the operator your contact information (phone numbers, emails) so that they can contact you during the observations if necessary. You will also need to let the operator know what computer you will be using during your observations. At this time you will begin to open a Virtual Network Computer (VNC) session that you will use for the remote observations. Starting this early will allow for any problems encountered while preparing to observe remotely to be solved before the observations are to begin.

- You can find information about GBT remote observing policies at
  - <https://science.nrao.edu/facilities/gbt/observing/policies>
- You can find information about opening a VNC session at
  - <https://science.nrao.edu/facilities/gbt/observing/remote-observing-with-the-gbt>.

**Step 6 - Operator responsibilities**

The operator on duty will handle several tasks for you at the beginning of your observations. They will “put you in the gateway” (give you security access) so that you can control the GBT. They will also get the correct receiver into the focus position of the GBT, get the antenna motor drives ready for movement, place the correct pointing models into the system, and set the GBT’s Active Surface (AS) into the proper state. The operator is there to take care of all safety issues concerning the GBT.

**Step 7 - Begin observations**

Now you are ready to observe. You will use the Astrid (see Chapter 4) to perform your observations by submitting an SB (see Chapter 6). The steps you will take in observing are generally:

- A** Configure the hardware to the desired states. The parameters used to determine these states are known as the “configuration” and the act of setting these states is known as “configuring”.
- B** Slew to your source.
- C** Balance the Intermediate Frequency system (IF system). In this step you command the system to automatically adjust amplifier and attenuator settings in the IF system to ensure that all components operate within their linear regime.
- D** Execute your observations using one of the SB Scan Types (see Chapter 6).
- E** If problems develop let the operator know and they will either help solve the issue or contact the on-call support scientist for assistance.

**Step 8 - Immediately after observations**

Once you are done observing you should close Astrid, log out of the computer you were using, and leave the control room. Or, if observing remotely, **properly close your VNC session** using the `vncserver -kill` command.

**Step 9 - Data reduction**

During and after your observing run you will reduce your data. You will generally use GBTIDL for data reduction of spectral line data. This can be done either at Green Bank or your home institution. Continuum reduction support is available for the CCB (see Chapter 12). Otherwise only rudimentary continuum data reduction support is available for the GBT at this time, and you should contact your GBT “friend” for more information. A pulsar data reduction package, PRESTO, is available from Scott Ransom.

Use one of the data reduction machines, not the workstations used for running the observations. Refer to <http://www.gb.nrao.edu/pubcomputing/data-reduction.shtml> for a list of data reduction machines at Green Bank.

Note that observers wishing to reduce VEGAS data will need access to the “lustre” file system. Refer to <http://www.gb.nrao.edu/pubcomputing/public.shtml> for a list of lustre clients.

**Step 10 - Transfer your data**

Once you are done you will want to transfer your data to your home institution (see Chapter 20).

**Step 11 - Keep in touch**

Finally you will want to write your Nobel Prize winning paper. The NRAO can help you with your page charges (see Chapter 20). You should also notify your scientific contact person of your paper to help the Observatory keep track of how successful all observing projects have been.



## Chapter 3

# Introduction to the Dynamic Scheduling System

This chapter gives an introduction to the Dynamic Scheduling System (DSS) for the Robert C. Byrd Green Bank Telescope (GBT). The GBT has been scheduled with the DSS since October 1, 2009. Observers can access the DSS through this site: <https://dss.gb.nrao.edu>

### 3.1 Overview of the DSS

The primary goal of the Green Bank Telescope Dynamic Scheduling System (DSS) is to improve the efficiency of GBT observations by matching the observing schedule to predicted weather conditions while allowing each observer to retain interactive control of the telescope. Each day the DSS will examine the weather forecast, equipment availability, observer availability, and other factors, and set an observing schedule for the 24-hour period beginning the next day. Observers will therefore get about 24-48 hours notice before their project will observe. Observers will have the opportunity to suspend their observing program, set blackout dates indicating when they are unavailable for observing, and back out of current observations if they find the observing conditions are not suitable to their science goals.

The DSS readily accommodates remote observing, but by being on site in Green Bank observers increase their likelihood of being scheduled during the period of their visit. Visits to Green Bank should be arranged in advance with the project's "Friend", and observers should ideally spend one to two weeks in Green Bank to give enough opportunity for their project to get scheduled at least once. **Projects observing at high frequencies (20 GHz and higher) typically require staying in Green Bank for two weeks or longer.**

### 3.2 DSS Terminology

The process of scheduling GBT observations begins with the preparation of the proposal using the NRAO Proposal Submission Tool (PST). Proposals accepted by the NRAO Time Allocation Committee become GBT projects that appear in the DSS system and are identified by an assigned project ID (e.g., GBT09A-001).

Projects are divided into sessions, which have associated parameters that define how the observation should be scheduled. These parameters include sky position, time allocated, observing frequency, and minimum and maximum durations preferred for a single, contiguous block. Sessions for monitoring

observations have additional parameters describing how often to repeat the observation. The project investigators initially define the session parameters in the proposal, but the parameters may be modified by request to the helpdesk (helpdesk-gb@nrao.edu). Observers can see the most critical session parameters on the DSS web pages.

Completing the observations for a session may require scheduling multiple segments. Each contiguous block of scheduled time is called a *telescope period*.

As telescope periods are completed, the project and associated sessions will be billed for the time. If any time is lost to weather or an equipment failure, the observer may consult with the telescope scheduler (via the helpdesk) and request that the project not be billed for the lost time.

### 3.3 Controlling the Scheduling of a Project

Users can access their DSS account by logging in to the system at <https://dss.gb.nrao.edu>. The DSS username and password are the same as those used for NRAO Interactive Services (i.e., the Proposal Submission Tool).

From the DSS web site, users can view and manage the scheduling information for their projects. In order for a project session to enter the pool of sessions eligible for scheduling, the user is responsible for ensuring that the session is enabled in the DSS, and that a qualified observer is available to perform the observation. Sessions and observers are enabled for observing simply by clicking a check box in the DSS project page (See figure 3.2). Users can control when their project is scheduled by enabling or disabling individual sessions.

Note that astronomers intending to observe remotely must be trained and approved by GB staff before the project can be authorized and made eligible for scheduling.

Observers can enter personal blackout dates. Blackouts can be entered either as one time events (e.g., May 1, 20:00 to May 4, 05:00 UT) or as repeating events (e.g., every Monday from 15:30 to 17:30 ET). If all observers for a given project are blacked out at a given time, that project will not get scheduled. If at least one observer is not blacked out, the project is eligible for scheduling. The default time zone used for entering blackouts is set on the *Preferences* tab, which is linked at the top of every DSS web page. Observers can also override the default by selecting a time zone when making a blackout entry. Observers with more than one project will find that they need to enter blackout dates only once, and the dates will be applied to all their projects. **Those visiting Green Bank to observe should use blackout dates to mark the periods of their travel before and after the run to ensure they are scheduled only when available and ready on-site.**

*Guidelines for the use of blackouts:* While blackout dates give observers control of the scheduling process, efficient GBT operation requires that not too much time be blacked out or disabled. It is especially important that projects with large observing allocations not have too much time unavailable for scheduling because of blackouts. As a guideline, projects with more than 20 hours of allocated observing should limit time that cannot be scheduled to no more than 20% of the total eligible observing time over the course of a semester. If a project cannot meet this guideline, the PI is encouraged to increase observing opportunities by enlisting additional observers who are qualified for remote observing. Projects that require observers to visit Green Bank for training are excluded from this guideline until the observers are trained for remote observing.

*Caution Regarding Blackouts:* If a project has only one observer, that observer should be particularly conscientious of blackouts. It can be easy for an observer to inadvertently hamper observing opportunities by setting blackout dates too freely, particularly repeating blackouts. Repeating blackouts should be used with care. Targets with low declinations, such as the Galactic Center, have tightly constrained observing opportunities to begin with, so observers on such projects should be particularly careful with blackouts that would further limit their observing opportunities. Consider, as an example, a project that has a session with a 4-hour minimum duration to observe the Galactic Center. If the observer has

a repeating 1-hour blackout date that intersects the window, the entire session becomes ineligible each time the blackout intersects the 4-hour window. The Green Bank Observatory is not responsible for lost observing opportunities due to excessive blackouts.

When entering blackouts, keep in mind, too, that projects do expire, so it is in the interest of the observer to keep the projects eligible for scheduling as much as possible.

## 3.4 Canonical Target Positions

The DSS keeps track of a project's scheduling requirements via the session parameters, which can be viewed on the project page. The PI should check that session parameters properly reflect the needs of the project. The project Friend assigned by NRAO can also offer advice on optimizing session parameters, where appropriate. In some cases, a session's target position may be representative of a group of objects clustered on the sky. As the project progresses and some of these targets are observed, this representative position may need to be updated. In this case, the PI should send an email request to the DSS helpdesk.

The DSS can automatically update the sky coordinates of common, fast-moving solar system objects, including comets. The position is updated each day prior to scheduling. On the project page under *Project Sessions*, an asterisk next to the coordinates indicates that the position for that session is automatically updated in this manner.

Many observers find it helpful to use a sky-plotting tool to help plan their observations and keep track of target locations on the sky. The Control Library for Engineers and Operators (CLEO) Scheduler & Skyview tool, which runs on Linux systems in Green Bank and can be run remotely through VNC, is one such tool that allows a GBT user to plot target locations on the sky for any date and time. This application can read target coordinates from a standard astrid catalog file. Observers will find this tool handy for identifying the time of day a project may get scheduled, as well as helping to plan observations in detail after they are scheduled. To run the program, type `cleo_scheduler` from the command line. See § 5.2.3 for further details on the CLEO Scheduler & Skyview application.

## 3.5 Contact Information and Project Notes

Observers can specify how they should be contacted, prior to and during their observations. It is critical to keep contact information current. Each observer can provide *dynamic contact* information in a free-format text box. Here the observer should provide any contact information not available through the person's (static) NRAO contact information, which is also listed on the page. Observers can also specify the order in which they should be contacted by GBT operations, in the event of any schedule changes or in case there is need to contact the observer for any reason prior to the scheduled start time. Specify the order by clicking the arrow icons next to the list of team members, on the DSS project page.

Finally, observers can record *Project Notes* on the DSS project web page. Project notes provide observers a place to store and share observing instructions. The notes are visible to all project team members as well as the GBT operations staff and schedulers. Observers who need to share instructions or other information with the GBT operator prior to the start of an observation can provide these instructions in the project notes area. Project notes are not intended to be a log for observations, but rather a place to store brief instructions or news that should be shared among observers and the GBT operator.

## 3.6 The DSS Software

### 3.6.1 The DSS Home Page

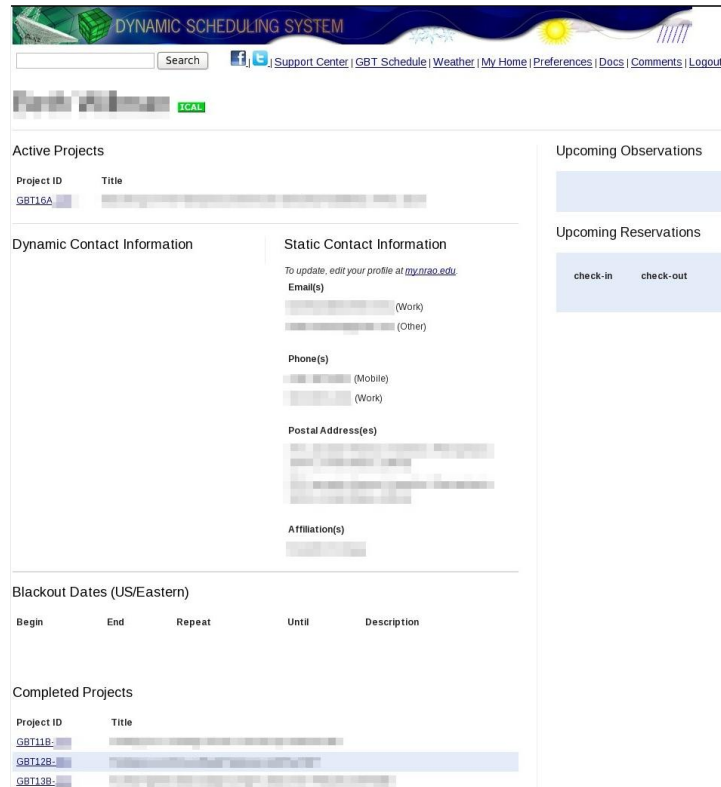


Figure 3.1: A sample DSS home page.

Upon logging in to the DSS system, users arrive at their DSS home page (Figure 3.1) where they see a list of active projects on which they appear as co-investigator. From the DSS home page, users can:

- Access the project page for each of their affiliated projects
- See a list of upcoming observations
- See a list of upcoming Green Bank room reservations
- See their *static* contact information, as entered in the NRAO services system <http://my.nrao.edu>.
- Set *dynamic* contact information
- Set blackout dates
- Follow a link to the current GBT fixed schedule
- Follow a link to the weather forecasts page
- follow a link to the NRAO support center
- Set the default time zone via the Preferences link
- Access DSS documentation
- Establish an iCalendar subscription. Instructions for using iCalendar are available by hovering the mouse cursor over the iCal icon on the DSS Home Page.

### 3.6.2 The DSS Project Page

By selecting a project ID, observers are presented with the project page, where they can:

**Project Sessions**

Name	Coordinates	Freq	Rcvr	Time billed	Min/Max Dur.	Type	Gr	Enabled	Other Parameters
GBT16A	RA: 02:00:00.0 Dec: 30:00:00.0	21.0	KFPA	19.75 / 20.0	0.25 - 5.0	spectral line	A	<input checked="" type="checkbox"/>	Xi: 1.2
GBT16A	RA: 18:00:00.0 Dec: 30:00:00.0	21.0	KFPA	23.25 / 29.25	3.0 - 6.0	spectral line	B	<input checked="" type="checkbox"/>	Xi: 1.2
GBT16A	RA: 10:00:00.0 Dec: 30:00:00.0	21.0	KFPA	50.0 / 50.0	1.0 - 5.0	spectral line	B	<input checked="" type="checkbox"/>	Xi: 1.2

**Project Calendar (US/Eastern)**

April 2016

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	31	1	2
	19:00 Observing GBT16A -03 5:00 Observing GBT16A -02	20:00 Observing GBT16A -03	8:30 Observing GBT16A -02			
3	4	5	6	7	8	9
	23:00 Observing GBT16A -03 3:00 Observing GBT16A -02 15:00 blackout	19:00 Observing GBT16A -03 3:00 Observing GBT16A -02	17:00 Observing GBT16A -03	11:00 blackout	4:00 Observing GBT16A -02	
10	11	12	13	14	15	16
blackout	19:30 Observing GBT16A -03	20:00 Observing GBT16A -03				
17	18	19	20	21	22	23
	8:00 blackout					
24	25	26	27	28	29	30
1	2	3	4	5	6	7

**Team Members**

Call Order	Name	Email(s)	PI	Contact	Remote	Observer(s)
▲ ▼					Yes	<input checked="" type="checkbox"/>
▲ ▼				Yes	Yes	<input checked="" type="checkbox"/>
			Yes			<input type="checkbox"/>
						<input type="checkbox"/>
						<input type="checkbox"/>

**Project Friends**

Name	Email(s)	Required
Daniel Perera	dperera@nrao.edu	

**Observer Blackouts (US/Eastern)**

Observer	Begin	End	Repeat	Until	Description

**Telescope Periods Completed (US/Eastern)**

Session	Start Time	Duration	Billed Hours
GBT16A -01	2016-02-04 14:00	3:30	3:30

**Project Notes (edit)**

**Project Abstract**

**Project Disposition**

- Inspect session parameters
- Enable or disable individual session
- View total allocated and billed time
- See a project calendar
- View scheduling alerts
- View receiver availability
- View upcoming reservations
- View upcoming observations
- Specify observers from the project team, and set the order they should be contacted by GBT operations
- See a list of blackout dates for all observers on the project
- See a list of completed telescope periods
- Store and share project notes
- View your abstract and disposition

Figure 3.2: A sample DSS project page.

The project calendar gives observers an idea when their project is eligible for scheduling. Regardless of the weather, there will be times when a project is not eligible for scheduling, for example because of no receiver availability, observer blackouts, fixed telescope maintenance periods, and other fixed projects appearing on the GBT schedule. Times not eligible for scheduling will be grayed out on the project calendar.

The project calendar helps with planning in a number of ways. However, it is important to understand that a session’s eligibility is based on ever-changing constraints, and can change from *not eligible* to *eligible* at any time. Therefore, if observers wish to take a break from observing based on the calendar outlook, they should either disable all sessions until they are ready to resume with the observing, or enter blackout dates to cover the period they do not wish to observe.

The project page includes a panel with project team members listed. Using a checkbox, team members can select or deselect those identified as observers. They can also rearrange the order observers are listed. The top observer in the list is expected to observe the next scheduled session. If there is a change in schedule, this person will be called first.

## 3.7 Responsibilities

Each project has a Principle Investigator (PI) and, optionally, a list of additional investigators. An investigator is eligible to be an observer for a given project if that person is qualified for remote observing or is on site in Green Bank.

It is essential that one of the observers for a scheduled project contact GBT operations at least 30 minutes prior to the start of the observation. Observers can contact the GBT operator by telephone (304-456-2341), by the CLEO chat program “Talk and Draw” (for qualified remote observers), or by showing up in the GBT control room. If the GBT operator has not been contacted before the session’s start time, the operator will phone observers in the order they are listed on their project web page.

- The PI is responsible for:
  - Managing the project
  - Identifying all associated observers
  - Working with project team members and the GBT project Friend to ensure that SBs are properly and promptly prepared.
  - Enabling each session by clicking the “enable” button on the project’s web page. Sessions should be enabled only if they will be ready for observing in the next 24 hours.
  - Ensuring that all associated observers have provided contact information, including a current telephone number and an email address for each observer.
  - Ensuring that a project’s scheduling information is current. This includes checking the hours remaining on the project and ensuring that the session parameters are up-to-date and accurate.
  - Ensuring that each scheduled telescope period has an observer who is available at least 30 minutes before the session is scheduled to begin.
- Observers are responsible for:
  - Ensuring that the DSS project web page has their current contact information. For remote observers, this includes entering telephone numbers where they can be reached at the time of observation.
  - Contacting GBT operations 30 minutes prior to the start time of an observation.
  - Attending to observations during a scheduled telescope period. The PI is responsible for “no-shows” and the ensuing reduction in their allotted time.
  - Notifying GBT operations if they find conditions unsuitable for their session.

## 3.8 Remote Observing

To use the GBT remotely, observers must first be trained and certified by Green Bank staff. In general, astronomers must observe at least once in Green Bank before being certified for remote observing. **Please note that students should be trained on site by GBT staff, not off site by others.** Experienced observers, when using instruments or observing modes unfamiliar to them, should plan to visit Green Bank if they require assistance.

Contact your project “friend” or the DSS helpdesk ([helpdesk-dss@gb.nrao.edu](mailto:helpdesk-dss@gb.nrao.edu)) if you believe the DSS does not have you listed properly as a qualified remote observer.

See <https://science.nrao.edu/facilities/gbt/observing/remote-observing-with-the-gbt> and Chapter 18 and for more information on remote observing.

## 3.9 The Daily Schedule

Each day between about 7:00 and 12:00 PM ET the telescope schedule is fixed for the 24-hour period beginning 8:00 AM ET the next day. For example, by 12:00 PM Monday, the observing schedule is fixed for the period 8:00 AM Tuesday through 8:00 AM Wednesday. Each morning this daily schedule is published and can be viewed on the DSS web site by anyone. Those with projects on the 24-hour fixed schedule will be notified by email.

Observers must ensure that their blackout dates and “session enabled” flags are up to date each day by about 5:00 AM ET. Changes made after this time may not be reflected in the upcoming day’s schedule.

It is possible that weather conditions may change after a schedule is published, compromising the observing efficiency for some scheduled telescope periods. The observer or GBT staff may then decide to cancel a telescope period and substitute an alternate “backup” observation in its place. Note that the observer may decide that the weather conditions are too poor even after beginning the observation. Equipment failure can also lead to cancellations. If GBT staff must change the 24-hour schedule for these reasons, affected observers will be notified immediately by email or telephone.

## 3.10 Backup Projects

When a scheduled telescope period is cancelled, a backup project will be scheduled on short notice. By volunteering as a backup project, observers improve their project’s chances of getting observing time. Backup projects can come in two categories: observer-run and operator-run. There are several requirements that must be met before a project can be considered for backup status. Please refer to Appendix F for further details.

## 3.11 Session Types

There are four types of sessions defined for astronomy projects: open, windowed, elective, and fixed. Open sessions have no major constraints on when they can be scheduled, beyond the functional requirements that an observer is available, the source is above the horizon, and the weather is suitable. Most sessions fall into this category and provide the most flexibility in the DSS. At the other extreme are fixed sessions that have no flexibility and are prescheduled at a particular date/time; that is, their telescope periods have already been defined.

The other two types are windowed and elective sessions, which have some constraints but are not fixed on the schedule. The most common examples are monitoring and VLBI sessions, where the science demands that an object must be observed at defined intervals or times.

Windowed sessions are defined by a cadence that may be either periodic or irregular. For example, an observer may require observing a target once per month for five months, with each observation having a tolerance of plus or minus 3 days. In this example, the window size is 7 days.

Currently, windowed sessions are scheduled in the following way. The cadence information from the proposal is used to preschedule all windowed sessions whereby all of the telescope periods are temporarily fixed in what are called default periods. The user is given the window template (e.g, 8-14 January; 8-14 February; 8-14 March; 8-14 April; and 8-14 May). Within a windowed period, a windowed session will be considered like an open session. Near the end of each window range is a default period. If the session has not been selected by the time the default period arrives, the session will be scheduled in the default period. The default period may be moved manually to a later time slot within the window if the human scheduler notices a problem with the original default period. When the windowed period is scheduled, the observer will be informed 24-48 hours in advance, just like an open session. The only difference is that the observer will be provided with the window template for planning purposes.

Elective sessions are a restrictive form of windowed sessions. Here, rather than having a range of days on which the project session can be scheduled, there is a list of possible days. As with windows the list of possible days, or *opportunities*, has a default period on which the session will be scheduled if it has not run in advance of that date.

## 3.12 Projects that can Tolerate Degraded Weather

The DSS is designed to schedule projects in weather that is appropriate for the frequency being observed. Some projects can tolerate lesser weather conditions than the DSS would assign by default. For example, consider a project at K-band that observes many targets, each for a short duration, say 10 seconds. The observing time for this project is dominated by overheads in slewing from one position to the next, so marginal K-band weather might be acceptable. The observing team may prefer not to wait for very good K-band weather, which is rare and would delay their scheduling.

To enable more aggressive scheduling, the observer should send an email to the DSS helpdesk requesting that the project be considered for scheduling in lesser weather conditions. The DSS support team can enter a session-specific factor ( $\xi$ ) that effectively elevates the score for this session in marginal opacity conditions. The  $\xi$  parameter is tunable so the observer can request that the project be scheduled very aggressively, or modestly so. The factor only affects scoring related to atmospheric opacity, so high frequency projects that are sensitive to high winds will still not get scheduled when the forecasted winds preclude accurate pointing.

The DSS support team will help observers decide if their project can tolerate lesser weather. Note that this capability will not be used to accelerate scheduling of projects that truly do benefit from the most appropriate weather.

## 3.13 Other DSS Control Parameters

A list of the most relevant parameters can be found in Appendix G. There are a number of additional controls and parameters that can be used within the DSS system which are fully described in DSS project note 10 ([https://safe.nrao.edu/wiki/pub/GB/Dynamic/DynamicProjectNotes/dspn10\\_6\\_final.pdf](https://safe.nrao.edu/wiki/pub/GB/Dynamic/DynamicProjectNotes/dspn10_6_final.pdf)). Any changes to these parameters must be requested by contacting the GBT scheduler via the helpdesk ([helpdesk-gb@nrao.edu](mailto:helpdesk-gb@nrao.edu))



# Chapter 4

## Introduction To Astrid

### 4.1 What Is Astrid?

Astrid is a single, unified workspace that incorporates a suite of applications that can be used with the GBT. Astrid provides a single interface from which the observer can create, execute and monitor observations with the GBT. Some of the features of Astrid are:

- Executes Scheduling Blocks (SBs) to perform astronomical observations.
- Provides a real time display of GBT data
- Provides the status of the GBT.
- Provides an area to edit SBs. They may be edited offline and saved before observing.
- Allows a second observer to monitor observations in progress.

Astrid brings together many applications into a single, unified Graphical User Interface (GUI). Of particular note, Astrid provides a single point of contact to all of the Monitor and Control (M&C) software by interpreting the Python code and function in SBs. The GBT M&C systems can roughly be thought of as a group of programs - one for each hardware device - and a master program, the Scan Coordinator. The GUI places each application into its own tab window. Applications available in Astrid are:

#### **Observation Management**

Astrid interfaces with the Observing Management Application in order to execute SBs. The Astrid Edit Subtab (see § 4.4.1) provides a windows-like text editor that features syntax highlighting for Python code and allows SBs to be edited, validated, copied, and saved. SBs may be queued and executed via the Astrid Run Subtab (see § 4.4.2).

#### **Data Display**

Astrid provides a real time data display by connecting to The GBT Fits Monitor (GFM). This allows the automatic processing of pointing and focus scans that can immediately update the GBT M&C system with the determined corrections. GFM can show raw, uncalibrated continuum data as a function of time (see Chapter 5).

#### **GBT Status**

Astrid provides a screen that displays information on the real time status of the GBT. This provides meta-information such as the Local Sidereal Time (LST), Coordinated Universal Time (UTC), observer, project ID, information on the antenna such as current position, and information on the current scan and IF setup (see § 4.6).

## 4.2 How To Start Astrid

To start Astrid, type `astrid` from the command line on any Linux computer in Green Bank. The first thing you will see is the Astrid “splash screen” which is shown in Figure 4.1. The Astrid GUI should appear on-screen after 10-20 seconds (see Figure 4.3).

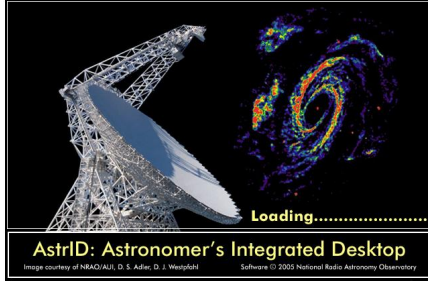


Figure 4.1: The Astrid splash screen.

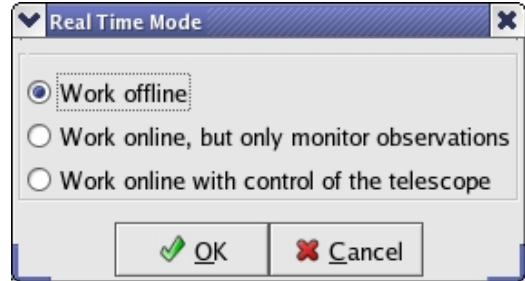


Figure 4.2: Astrid startup pop-up window.

### 4.2.1 Astrid Modes

On startup, Astrid will automatically ask what mode to operate in via the pop-up window shown in Figure 4.2. Once an initial mode has been set it may be changed at any time by selecting `Real Time Mode...` from the `File` drop-down menu (see § 4.3.4).

**Note** that observers should use `File→Real Time Mode...` to relinquish control of the telescope immediately after their scheduled observing session.

Table 4.1: Astrid mode features

Mode	Edit & Validate Syntax	Validate Configuration	Submit SBs	Observing Logs	Data Display
<b>Offline</b>	✓	Simulated			Historical <sup>(1)</sup>
<b>Online (monitor)</b>	✓	Simulated		✓	Real-time
<b>Online (control)</b>	✓	Real <sup>(2)</sup>	✓ <sup>(3)</sup>	✓	Real-time

(1) Previously acquired data should always be viewed ‘offline’.

(2) Requested configurations are validated with respect to the actual `dev_health.conf` file rather than the simulated ‘ideal’ universal cabling file.

(3) Only permitted when you are ‘in the gateway’ (the GBT operator has given you security access).

The features available for each mode are listed in table 4.1. Users should select the most appropriate mode for their purposes:

- **Work offline:** Primarily used to create, edit and validate SBs. It is also the preferred method to look at previously obtained data in the Data Display since online modes will continually refresh the display window with near-real-time data.
- **Work online, but only monitor observations:** May be used to view what is happening in the Astrid observing logs and Data Display for the current observations. You will not be able to submit SBs or affect observing in any manner.
- **Work online with control of the telescope:** Used to perform observations with the GBT by allowing the user to submit SBs. Log information and real-time data displays are also available in this mode. **Note that working online requires the GBT operator to “put you in the gateway” (give you security access).**

## 4.3 Astrid GUI Composition

The Astrid GUI layout consists of several components shown in Figure 4.3 and described in the following section.

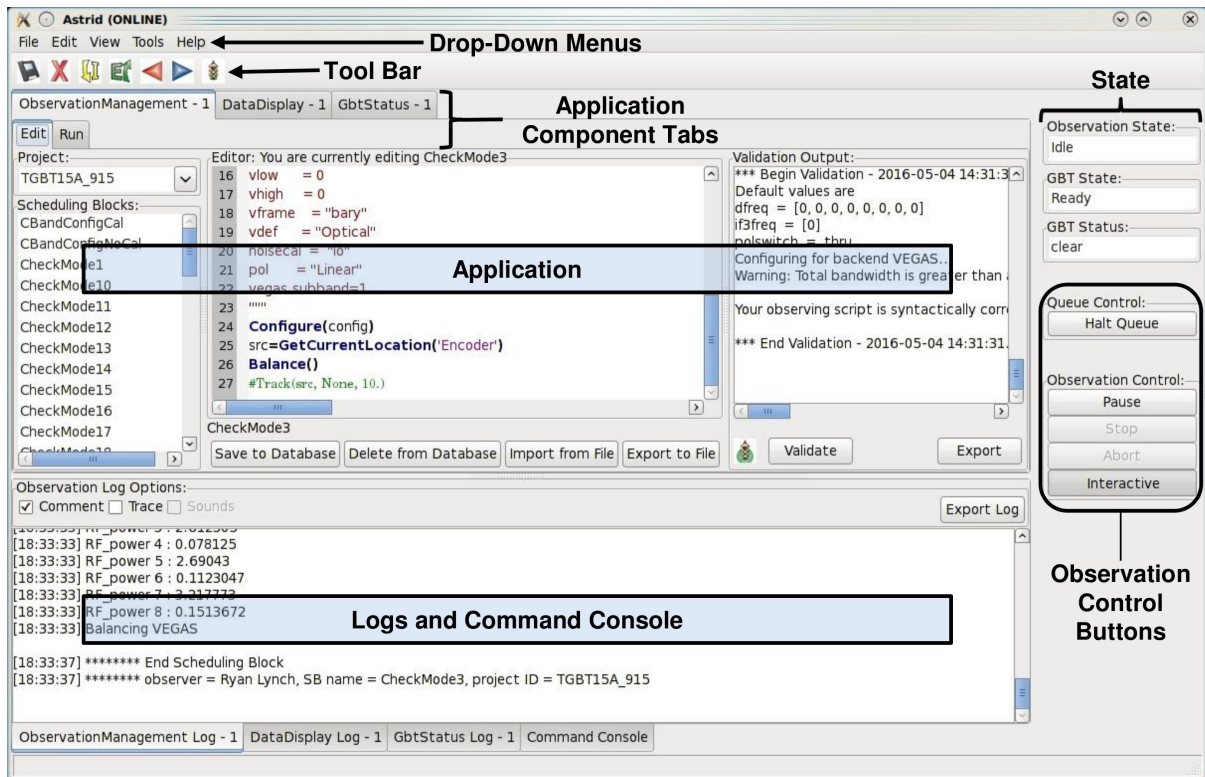


Figure 4.3: The different components on the Astrid GUI.

### 4.3.1 Resizing Astrid Display Areas

It is possible to resize some of the display areas within Astrid. If you put the mouse over the bar separating two display areas you will get a double-arringed cursor. If you then hold down the left-mouse button you can use the mouse to move the border and resize the display areas.

### 4.3.2 Application

This comprises the majority of the space within the Astrid GUI. This shows the contents of the Application selected by the application component tabs.

### 4.3.3 Application Component Tabs

The application component tabs are located under the Drop-down menus and the Toolbar. The top level of tabs allow users to switch between the three main Astrid applications: Observation Management, Data Display, and GBT Status. Below these are a set of subtabs that vary for each application component tab.

### 4.3.4 Drop-down Menus

In the top, left hand side of the Astrid GUI you will find the drop-down menus. The contents of the drop-down menus change according to which Application is currently being displayed on the Astrid GUI. We will not discuss all of the options under the drop-down menus in this document but we will provide some highlights.

**File** - **New Window** Launch applications within the Astrid GUI or in an independent GUI.  
- **Close Window** Close the currently displayed application in the Astrid GUI.  
- **Real Time Mode...** Change between the operational modes of Astrid (see § 4.2.1).

**Edit** Standard “Windows” undo, redo, cut and paste options.

**View** Display or hide the Toolbar or view Astrid in Full Screen mode.

**Tools** Only active for the Data Display Application. You may use checkboxes to select various tooltips such as `info`, `pan`, and `zoom`. You can also change the “Fitting Heuristics” used during the reduction of Pointing and Focus Observations by selecting `Options...` (see § 5.1.3.3).

**Help** Bring up documentation for some but not all Applications.

### 4.3.5 Toolbar

The Toolbar is located just under the Drop-down Menus near the top of the Astrid GUI. The contents of the Toolbar change depending on which Application is being displayed in the Astrid GUI. The Toolbar options are a subset of commonly used options from the Drop-down Menus. When you leave the mouse situated over one of the Toolbar buttons for a few seconds a “pop-up” will appear that tells you what action the Toolbar button will invoke.

### 4.3.6 Logs

The Log Window is located in the lower portion of the Astrid GUI underneath the Application display area. Clicking on the log tabs at the very bottom of the GUI will display log information for the Observation Management, Data Display, or GBT Status applications. Viewing a specific log will also change the application window to display the matching application.

The contents of the Observation Management application Log may be saved to an external file via the `Export Log` button. Note that closing or restarting Astrid will clear the Observation Management Log. If you wish to retrieve an unsaved observing log, please contact your GBT “Friend”.

### 4.3.7 Command Console

The Command Console is a Python shell that imports the “Configuration Tool” and “Balance” Application Programming Interfaces (APIs). Both APIs will only interact with the M&C systems if the user has been granted security access and is operating Astrid from the “Work online with control of the telescope” mode (see § 4.2.1).

Observers may find the Command Console useful as a stand alone Python Shell. However, the “configuration tool” and “Balance” APIs are only intended for use by GBT staff and expert users. Note that internal Astrid commands such as those listed in Chapter 6 are not available for use without first importing all necessary Astrid modules.

### 4.3.8 State

Three indications of state are located in the upper right corner of the Astrid GUI.

**Observation State** indicates **Astrid's state**.

If Astrid is not communicating with the M&C system (such as in its “offline” mode) then you will see “Not Connected”. If Astrid is communicating with the M&C system and there isn't an SB being executed then you will see “Idle” and if an SB is running (or has been paused) then you will see “SB Executing” (“SB Paused”).

**GBT State** indicates the **M&C system state**.






If the M&C system is not working properly you will see “Not In Service” or “Not Connected.” “Unknown” indicates that the M&C system is working but does not know the state of any of the hardware devices. You will see the state be “Ready” when the GBT is not doing anything. It will be “Activating” or “Committed” when the GBT is preparing to perform an observation, etc. While taking data during a scan the state will be “Running”. At the end of a scan you will see the state become “Stopping.” If the scan is ended for any abnormal reason the state will be “Aborting.”

**GBT Status** indicates the **error state of the M&C system**.

If the M&C system is not communicating properly with the hardware the status can be “Unknown” or “Not Connected.” If the status is “Clear”, “Info”, or “Notice” then there are no significant problems with the GBT. If “Warning” then it is worth asking the Operator what the problem is, but it may not affect observation quality. If the status is “Error” then there is potentially something wrong that may need attention. If the status is “Fault” or “Fatal” then something has definitely gone wrong with the observations.

### 4.3.9 Observation Control Buttons

The Observation Control Buttons are located in the lower-right of the Astrid GUI. These buttons give the observer control of the GBT during the execution of an SB and have the following functions:

	If this button is not activated then the SBs in the Run queue will continue to be executed in order. If this button is activated it will finish the current SB but will not allow the next SB in the Run Queue to execute until the button is returned to its default “off” state.
	Stop the execution of the current SB when the next line of code is encountered.
	Stop the current scan at the end of the next integration time. This is a nice, gentle way to stop a scan.
	Stop the current scan immediately. This may lead to corrupted data.
	When selected, will cause Astrid to automatically answer any pop-up query. Astrid will always choose what it deems to be the safest answer. This is useful when you have to leave the control for an extended period of time (such as when you go to the cafeteria to eat, etc.).

## 4.4 The Observation Management Tab

The Observation Management Application consists of two sub-GUIs, the Edit Subtab and the Run Subtab (see Figures 4.4 and 4.6). In the Edit Subtab you can create, load, save, and edit SBs. You can also Validate that the syntax is correct. The Run Subtab is where you will execute GBT observations.

### 4.4.1 The Edit Subtab

The Edit Subtab has five major areas: a list of Project Names, SBs that have been saved into the Astrid database for that project, an editor, a Validation area, and a log summarizing the observations. This is shown in Figure 4.4. Chapter 6 covers the contents and creation of SBs.

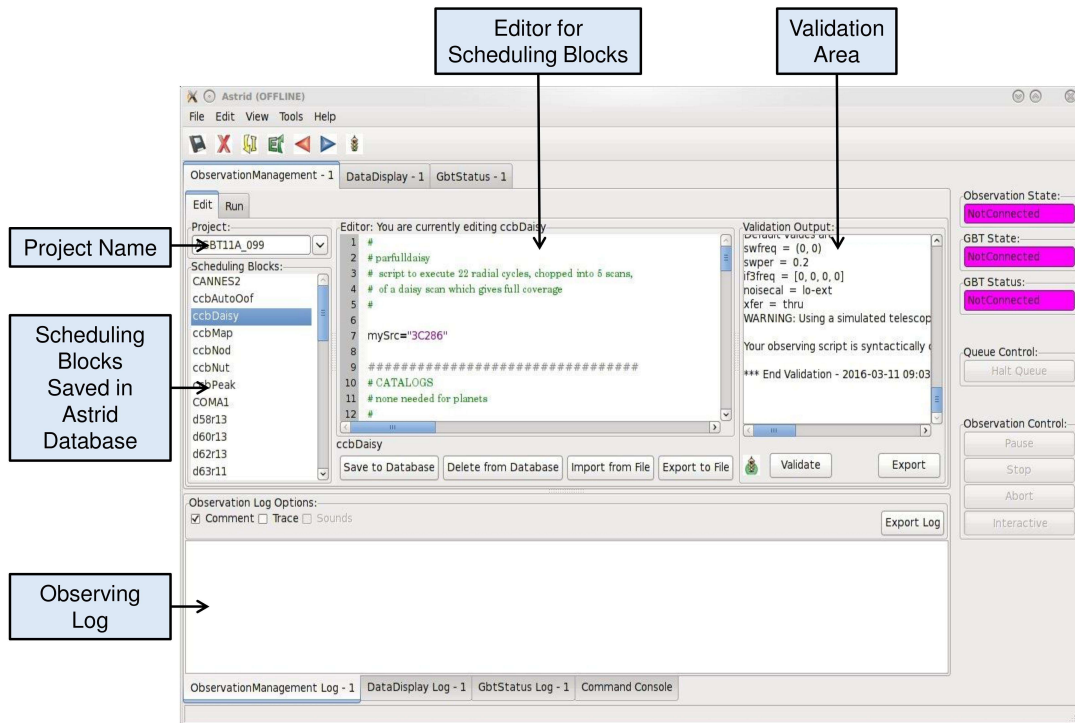


Figure 4.4: The Astrid Observation Management/Edit Subtab.

#### 4.4.1.1 Project Name and List of Scheduling Blocks

To access scheduling blocks associated with your project, you will need to enter your Project Name in the “project” window located in the upper left part of the Edit Subtab. Your Project Name is the code that your GBT proposal was given with the prefix “AGBT”, e.g., AGBT16A\_001. To enter a Project Name you may either type it in directly, or use the drop-down arrows to navigate to your project through a project hierarchy as shown in Figure 4.5

After doing this you will see in the window labeled “Scheduling Blocks” a list of SBs, if any, that have been previously saved into the Astrid database. All of the saved SBs for a given project will show up in the “Scheduling Blocks” section of the Edit Subtab. If an SB has been Validated (i.e. it is syntactically correct) then it will appear in bold-face type. This means that it can be executed. If the script has been saved but is syntactically incorrect it will appear in lighter-face type.

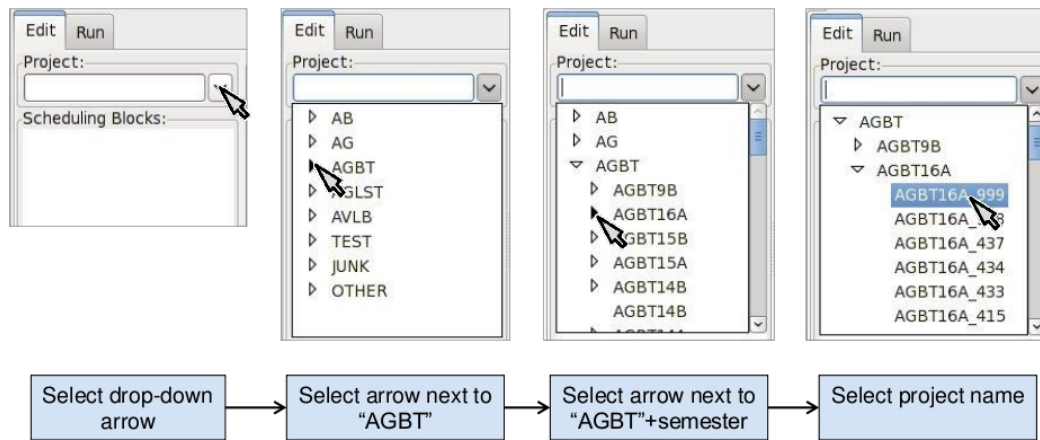


Figure 4.5: Selecting your project using the drop-down menu.

#### 4.4.1.2 Editor

You can use the Editor to create or modify an SB within Astrid. Standard Windows functions like Ctrl-X (to cut selected text), Ctrl-C (to copy selected text), and Ctrl-V (to paste selected text) can be used within the editor. The editor lists the line number on the left hand side of the window and marks Python code as follows:

- **Green highlighted text** - Commented characters
- **Black highlighted text** - Standard Python commands/syntax
- **Purple highlighted text** - Strings
- **Magenta highlighted text** - Triple quoted strings (used in Python to enclose strings that span multiple lines)
- **Dark blue highlighted text** - Python functions
- $\ominus/\oplus$  - Marks the start of an indented block of Python code such as an **if** statement or **for** loop. Clicking on  $\ominus$  will collapse the indented code block and change the symbol to  $\oplus$ . Likewise, clicking on  $\oplus$  will expand a previously collapsed code block.

The editor also has four operational buttons:

- |                      |   |
|----------------------|---|
| Save to Database     | - This button will check the validation of the current SB and then save it to the Astrid database. A pop-up window will notify you if the SB did not pass Validation. A second pop-up window will allow you to set the name that the SB will be saved under in the Astrid database. |
| Delete from Database | - This button will delete the currently selected SB from the Astrid database.   |
| Import from File     | - This button will allow you to load an SB from a file on disk.   |
| Export to File       | - This button will allow you to save the edited SB displayed in the editor to a file on a disk. This does not save the SB into the Astrid database.   |

The first time you select either of the **Import from File** or **Export to File** buttons you will have a pop-up window that lets you select the default directory to use. After selecting the default directory you will get a second pop-up window that shows the contents of the default directory so that you can select or set the disk file name to load from or export to.

### 4.4.1.3 Adding and Editing Scheduling Blocks in the Database

We will first describe how to add an SB to the “Scheduling Block” list (i.e. database) and then we will describe how to manipulate and edit SBs in the list.

#### Saving a Scheduling Block to the Database

If you have already created an SB outside of Astrid, you should go to the Edit Subtab in Astrid and then use `Import from File` to load your SB into the Editor. Otherwise you can just create your SB in the Editor. To save the SB into the Astrid database you just need to hit `Save to Database`. This will run a validation check (see § 4.4.1.4) on your SB and then a pop-up window will appear which allows you to specify the name which you would like to use in the list for your SB.

#### Selecting a Scheduling Block

If you perform a single click on any SB in the “Scheduling Block” list, the contents of the selected SB will appear in the Editor. The selected SB will be highlighted with a blue background.

#### Mouse-button Actions on the selected Scheduling Block

If you perform a right mouse button click on the selected SB a pop-up window will appear that will let you rename, create a copy or save the SB to the Astrid database. You can also delete the SB from the Astrid database. You may also rename the SB if you perform a left mouse button double click on the script name in the list.

### 4.4.1.4 Validator

The Validation area is where you can check that the currently selected SB is syntactically correct. This does not check for run-time errors and thus, does not guarantee that the script will do exactly what you want it to do. For example, it can not check that you have the correct coordinates for your source. You will also see error messages, notices and warnings from the Validation in this area.

The Validator will attempt to verify that you are using a legal configuration. When run in Astrid’s offline mode, the Validator can only compare your requested configuration with a simulated “ideal” model of the telescope hardware. To perform a full configuration check against the true hardware state of the telescope (modelled by the `dev_health.conf` file), you must be running Astrid from the “Work online with control of the telescope” mode.

Before an SB can be run within Astrid it first must pass Validation. To Validate a script without saving it you can just hit `Validate`. An SB automatically undergoes a validation check when you hit `Save to Database` in the editor. Any messages, etc. from the validation will appear in the “Validation Output” test area. You can export these messages to a file on disk by hitting `Export` in the validation area.

The state of an SB’s validation is shown by the stop-light. If the script has never been validated or has been changed since the last validation the stop-light will have the yellow light on. If the SB fails validation the stop-light will turn red, while it will turn green if the SB passes validation.

**Note:** `for` loops with many repeats can take an extended amount of time to validate since the Validator will go through each step in the loop. Also be careful of infinite loops in the validation process. Use of time functions such as `Now()` (see Chapter 6) always return “None” in the validation.

### 4.4.1.5 The Observing Log

The observing log is always visible at the bottom of the Observation Management Tab. It shows information from the execution of SBs in either of the Astrid online modes. The observing log can be saved to a disk file by hitting the `Export` button that is just above the top right corner of the log display area. Note that closing Astrid will clear the observing log. If you wish to retrieve unsaved observing log information, please contact your GBT “friend”.



### 4.4.2 The Run Subtab

The Run Subtab is shown in Figure 4.6. Here you will queue up SBs to perform the various observations that you desire to make. The Run Subtab has five components. Across the top of the Run Subtab you enter information that will be put into the headers associated with the observations. On the left is a list of SBs that you can execute. On the right are the “Run Queue” which holds SBs that are to be executed in the future, and the “Session History” which shows which SBs have previously been executed. At the bottom is the “Observing Log”.

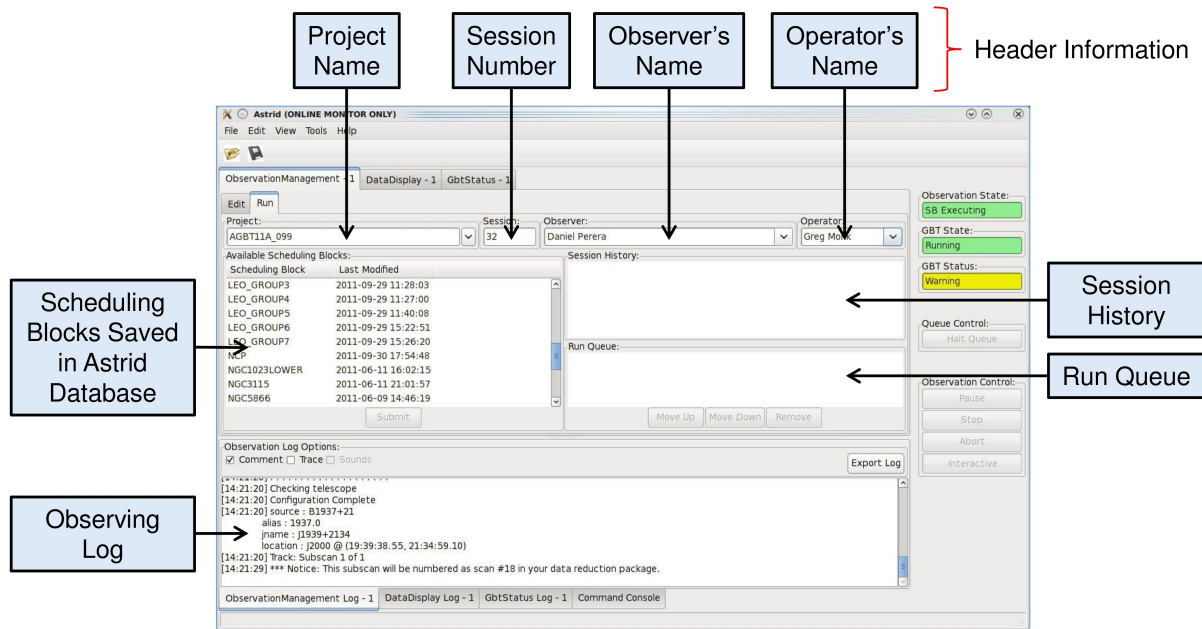


Figure 4.6: The Astrid Observation Management/Run Subtab.

#### 4.4.2.1 Header Information Area

The following fields must have entries before an SB can be executed:

**Project:** Just as in the Edit Subtab you use the drop-down menu to select your Project Name. If your project is not listed, ask your GBT “friend” or the telescope Operator to add it to the database.

**Session:** A session is a contiguous amount of time (a block of time) for which the project is scheduled to be on the telescope. Each time a project begins observing for a new block of time it should have a new session number. The session number is usually determined by Astrid and automatically entered. However, there are cases (such as Astrid crashing) where the session number could become incorrect. You can type in the correct session number if needed. **Note that a “Session” in Astrid is equivalent to an “observing period” in the lingo of the Dynamic Scheduling System (DSS). “Session” has a different meaning in the DSS.**

**Observer’s Name:** This is a drop-down list where you choose the observer’s name. Only the PIs on a project are guaranteed to have their name in this list. If your name is not listed, ask your GBT “friend” or the telescope operator to add it.

**Operator’s Name:** This is a drop-down list from which you pick the current operator’s name at the beginning of your observations.

### 4.4.2.2 Submitting An SB to the Run Queue

In order to execute an SB you must:

- Step 1.** Select the Observation Management Tab.
- Step 2.** Select the Run Subtab.
- Step 3.** Make sure that the header information fields all have entries.
- Step 4.** Select the SB you wish to execute from the list of available SBs.
- Step 5.** Hit the  button below the list of SBs.

Your SB is then automatically then sent to the Run Queue. Note that double-clicking on an SB is the same as selecting the SB and then hitting  .

### 4.4.2.3 The Run Queue and Session History

When an SB is submitted for execution it is first sent to the Run Queue. This contains a list of submitted SBs that will be sequentially executed in the future.

When an SB begins execution it is moved to the Session History list. So the Session History list contains the currently executing SB on the first line and all previously executed SBs that have been run while the current instance of Astrid has been running on subsequent lines.

If there are not any SB in the Run Queue when a new SB is submitted for execution it may appear that the SB just shows up in the Session History. However it has indeed gone through the Run Queue - albeit very quickly.

### 4.4.2.4 The Observing Log

The observing log is always visible at the bottom of the Observation Management Tab. It shows information from the execution of SBs. The observing log can be saved to a disk file by hitting the  button that is just above the top right corner of the log display area. Note that closing Astrid will clear the observing log. If you wish to retrieve unsaved observing log information, please contact your GBT “friend”.

## 4.5 The Data Display Tab

The Data Display Tab provides a near-real time display of your GBT data and is discussed in Chapter 5.

## 4.6 The GbtStatus Tab

The GbtStatus Tab displays various GBT specific parameters, sampled values and computed values. Special care was taken to promote its use for remote observing. An Example of how the GBT Status Display appears in Astrid is shown in Figure 4.7 and 4.8.

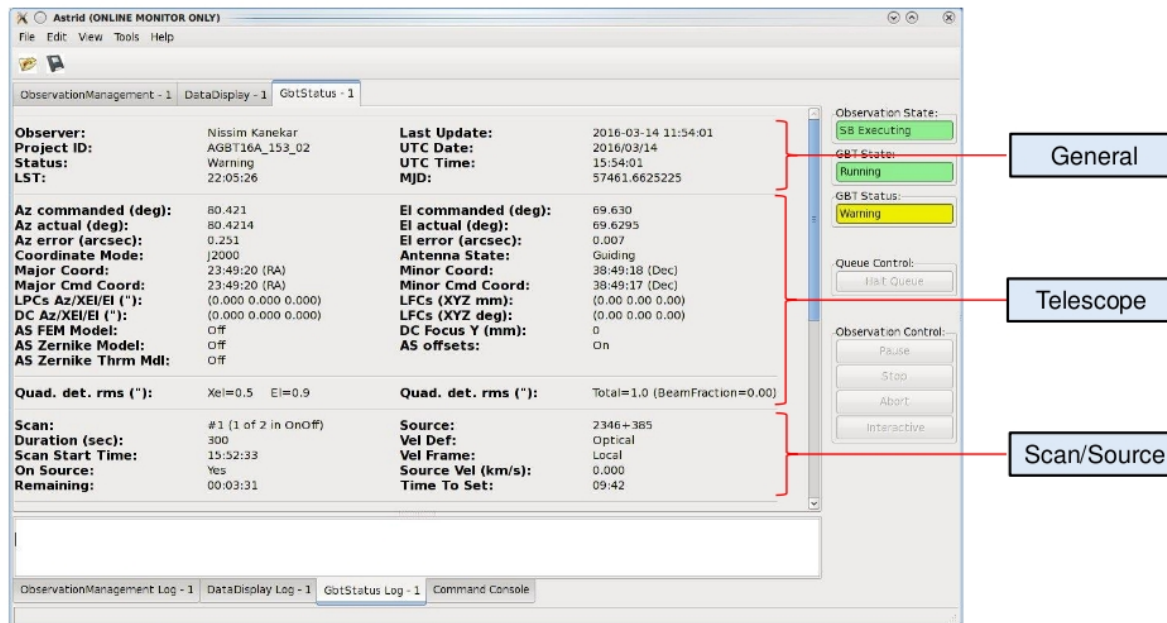


Figure 4.7: The top portion of the Astrid GbtStatus Tab. To see the rest of the status screen you will need to use the scroll bar.

The default status screen displays all of the currently supported items of the gbtstatus program grouped into various sections. These are:

### 4.6.1 General Status

**Observer:** The observer name.

**Project ID:** The data directory of the FITS files. This is your Project Name with the session as a suffix. For example, the Project ID for session 02 of AGBT16A\_001 would be AGBT16A\_001.02 (See § 4.4.1.1).

**Status:** The status of the GBT. See § 4.3.8

**LST:** The Local Sidereal Time (LST) of the last update.

**Last Update:** The local time when the database was last updated.

**UTC Date:** The Coordinated Universal Time (UTC) date of the last update.

**UTC Time:** The UTC time of the last update.

**MJD:** The Modified Julian Date (MJD) of the last update.

## 4.6.2 Telescope Status

**Az commanded:** The commanded azimuth position of the telescope in degrees.

**Az actual:** The actual azimuth position of the telescope in degrees.

**Az error:** The difference between the commanded and the actual azimuth position of the telescope in arc-seconds. This value does not contain a  $\cos(\text{el})$  correction.

**El commanded:** The commanded el position of the telescope in degrees.

**El actual:** The actual elevation position of the telescope in degrees.

**El error:** The difference between the commanded and the actual elevation position of the telescope in arc-seconds.

**Coordinate Mode:** The coordinate mode used to represent a particular location on the sky. See § 6.6.1

**Major and Minor Coord:** The telescope position in the current Coordinate Mode.

**Major and Minor Cmd Coord:** The telescope position in the current commanded Coordinate Mode.

**Antenna State:** If the antenna software is not running the state will be “Disconnected.” If the antenna software is running but with its control of the antenna turned off then the state is “Dormant.” If the antenna is not moving then the state will be “Stopped.” If the antenna is moving and data are being taken then the state is “Guiding” and if data are not being taken the state is “Tracking.” If the antenna is moving to a new commanded position the state is “Slewing.”

**LPCs Az/XEl/El:** The Local Pointing Correction (LPC) offsets in arc-seconds.

**DC Az/XEl/El:** The Dynamic Corrections values in arc-seconds. The GBT has temperature sensors attached at various points on the backup structure and the feed-arm. These are used in a dynamic model for how the GBT flexes with changing temperatures. This model is used to correct for pointing and focus changes that occur from this flexing.

**LFCs (XYZ mm):** The Local Focus Corrections (LFCs) for the offset focus position in millimeters. This value is determined from a Focus observation (see Chapter 6).

**LFCs (XYZ deg):** The subreflector tilt offset in degrees.

**DC Focus Y (mm):** The Dynamic Corrections Y subreflector offset in millimeters.

**AS FEM Model:** The state of the Finite Element Model (FEM) correction for the Active Surface (AS). The FEM predicts how the surface changes due to gravitational flexure versus the elevation angle.

**AS Zernike Model:** The state of the AS Zernike model correction model. The Zernike model is a set of Zernike polynomial coefficients determined from Out-Of-Focus holography that improve the shape of the AS versus the elevation angle.

**AS Zernike Thrm Model:** The state of the FEM correction for the AS. The FEM predicts how the surface changes due to thermal flexure.

**AS Offsets:** The state of the AS zero offsets. The zero offsets are the default positions for the AS. This should always be “On” if the AS is being used.

**Quad. det. rms:** The quadrant detector is used to detect and correct for wind-induced pointing errors. rms values in arc-seconds are reported in elevation and cross-elevation. Total rms is also given as a fraction of the beam.

### 4.6.3 Scan and Source Status

**Scan:** A scan is a command within an SB used to collect observational data. The field here is derived from the scan number and PROCNAME, PROCsize and PROCSEQN keywords from the GBT Observing (GO) FITS file.

**Duration:** The scan length in seconds.

**Scan Start Time:** If scan has started it is the UTC scan start time - if the scan has not started, then it is the countdown until the start of scan.

**On Source:** “Yes” or displays a countdown until the antenna is on source.

**Remaining:** The time remaining in the scan.

**Source:** The source name.

**Vel Def:** The velocity definition specifies which mathematical equation is used to convert between frequency and velocity. See Equations 6.1, 6.2, and 6.3

**Vel Frame:** The velocity frame or inertial reference frame. See the “vframe” keyword in § 6.2.5

**Source Vel:** The source velocity ( $\text{km s}^{-1}$ ).

**Time To Set:** The time till the current source sets.

The screenshot shows the Astrid ONLINE MONITOR ONLY window with the GbtStatus tab selected. The window is divided into several sections:

- Receiver:** Rcvr\_800, Linear, Cal State: On, Sw Period (sec): 1
- Obs Freq (MHz):** 699.0, Rest Freq (MHz): 699.000, Center Freq (MHz): 1080.000, Frequency State: Total Power
- Pyrometer(W/m2):** -126.7, IR Cloud cover(%): 6 (optically clear), Temperature (c): 15.4051, Wind Velocity (m/s): 2.6690, Dew Point (c): 11.7137, Wind Direction (deg): 234.5410, Humidity (%): 78.6375, Pressure (mbars): 919.6042, GBT Weather Source: Weather\_Station\_2
- RT phase delay (ps):** 823.9795, TAC-GBT\_VLBA dt (ns): 886, Site1Hz-TAC dt (ns): -278, Site1Hz-GBRtn dt (ns): 28933
- VEGAS Power Levels:** Bank A, -19.50|-19.88, 10, 950, Off, Total Intensity, 1, 2, Internal
- VEGAS:** PF0: 0.864, IF1: 2.666, CM1: 3.257, VEGAS-J1: -19.504, PF1: 0.898, IF3: 2.583, CM5: 3.525, VEGAS-J2: -19.858, PF0: 0.864, IF1: 2.666, DCR:A\_1: 13.083, TSys1: 43.665, backendIF: 0.699, PF1: 0.898, IF3: 2.583, DCR:A\_3: 12.370, TSys3: 44.171, backendIF: 0.699
- Observation State:** SB Executing, Running, Warming
- Queue Control:** Halt, Queue
- Observation Control:** Pause, Stop, Abort, Interactive

Red lines connect the following sections to labels on the right:

- Configuration: Observation State, GBT State, GBT Status
- Weather: IR Cloud cover(%), Wind Velocity (m/s), Wind Direction (deg), Pressure (mbars)
- Time Delays: RT phase delay (ps), Site1Hz-TAC dt (ns), TAC-GBT\_VLBA dt (ns), Site1Hz-GBRtn dt (ns)
- VEGAS: VEGAS Power Levels, VEGAS parameters
- IF: backendIF

Figure 4.8: The top portion of the Astrid GbtStatus Tab. To see the rest of the status screen you will need to use the scroll bar.

### 4.6.4 Configuration Status

**Receiver:** The receiver being used.

**Polarity:** The receiver polarity.

**Cal State:** ‘ON’ if the noise diode is firing during the scan.

**Sw Period:** The period in seconds over which the full switching cycle occurs. This is determined by the user in their configuration (see § 6.2).

**Obs Freq:** The observed spectral line frequency in the local frame (MHz).

**Rest Freq:** The spectral line frequency in the rest frame (MHz).

**Center Freq:** The center IF frequency set by the Local Oscillator (LO) in MHz. See Appendix B for further details.

**Frequency State:** The switching type. Either total power or frequency switching.

### 4.6.5 Weather Status

A real-time readout from one of the GBT weather stations providing information on temperature, pressure, humidity, dew point, wind direction and velocity. In addition, the pyrgeometer measures the net near-IR irradiance of the sky to give an approximate indication of cloud cover.

### 4.6.6 Time Delay Status

**RT phase delay:** This is the time delay between the timing center in the GBT equipment room and the GBT receiver room, in picoseconds, modulo 2000 ps. It is measured by comparing the phase of the 500 MHz reference signal sent to the receiver room with a copy of the signal returned to the timing center.

**Site1Hz-TAC dt:** Time difference between the Site1Hz (a one pulse per second signal that is locked to the hydrogen maser time standard) and a pulse from the GPS receiver (“TAC”)

**TAC-GBT\_VLBA dt:** Time difference between the GPS receiver and the VLBA back end timing module.

**Site1Hz-GBTRtn dt:** Time delay between the Site 1Hz and a copy of the 1 Hz returned from GBT receiver room. It is twice the delay of the fiber cables. The value is about 28933 ns which means the time delay between the equipment room the the receiver room is about 14466 ns.

### 4.6.7 VEGAS Status

**VEGAS:** The VEGAS Bank (spectrometer with letter designation  $A \rightarrow H$ ) selected in the scan coordinator.

**Power Levels:** The power levels at the inputs to the VEGAS Analog to Digital Converter (ADC) cards. There are two ADCs per bank, one for each polarization. The VEGAS balance API sets these values to approximately -20dBm by default.

**Mode Name:** Each VEGAS Bank can be configured in one of 29 modes (see Table 8.1).

**FilterBW:** The bandwidth (MHz) of the digital filter implemented in the Field-Programmable Gate Array (FPGA). Note that these values do not correspond to the bandwidths listed in Table 8.1.

**Noise:** The state of the noise source which can be either “On” or “Off”.

**Polarization:** Users may specify which spectral product to record (See the “vegas.vpol” keyword in § 6.2.5). vegas.vpol=“self” records “Total Intensity” products, “cross” records “Full Stokes” parameters, “self1” records the polarization inputs from the first ADC only, and “self2” records the polarization inputs from the second ADC only.

**Subbands:** Each VEGAS bank can select between single (subbands=1) and multiple (subbands=8) spectral windows when using VEGAS modes with a 23.44 MHz bandwidth.

**IntTime:** The VEGAS integration (dump) time in seconds.

**Switching:** Determines whether switching is controlled by VEGAS (“Internal”) or another source (“External”).

### 4.6.8 IF Status

The Intermediate Frequency path (IF path) firstplural in use are always displayed in the last section of the GBT status screen. An example screen is shown in Figure 4.8. Each line represents the IF path for a single polarization path from the IF Rack to the backend. Each line contains only the devices in use for the listed path. A path may include a subset of the devices and values listed below.

**IF#:** The # displayed is the number corresponding to the IF Rack switch in use. The value displayed is the RF power in Volts detected by the IF Rack.

**CM#:** The # displayed is the number corresponding to the Converter Module in use. The value displayed is the RF power in Volts coming out of the Converter Module after the Second LO (LO2) and Third LO (LO3) mixers and before the Converter Module filters.

**CF#:** The # displayed is the number corresponding to the Analog Filter in use. The value displayed is the RF power in Volts coming out of the Analog Filter Rack after all filters have been applied (used with 100 MHz Converters).

**SG#:** The # displayed is the number corresponding to the Analog Filter in use. The value displayed is the RF power in Volts coming out of the Analog Filter Rack after all filters have been applied (used with 1.6 GHz Samplers).

**VEGAS-J#:** The # displayed is the number corresponding to the port of VEGAS in use. The value displayed is the power level in dBFS. For best performance, it should be approximately -20 dBFS.

**Radar-Port#:** The # displayed is the number corresponding to the port of the Radar in use.

**DCR-Port#:** The # displayed is the bank and number corresponding to the port of the DCR in use. The value displayed is the total power in raw counts.

**TSys#:** The # displayed is the number corresponding DCR port in use. The value displayed is the system temperature as reported by the DCR (should be considered a loose approximation).

**backendIF:** The value displayed is the frequency of the Doppler track rest frequency as seen by the backend, in GHz.





## Chapter 5

# Near–Real–Time Data Displays and CLEO Utilities

## 5.1 The Astrid Data Display Tab

The Data Display Tab provides a real time display of your GBT data so that you can check that you are getting valid data. The Data Display is actually running an application called The GBT Fits Monitor (GFM). This application provides sub-scan-based display and analysis of GBT data, either in real-time as the data is being collected, or in an offline mode where it can be used to simply step through the sub-scans from an observation. Users are encouraged to run GFM offline for reanalyzing data during observations. A separate GFM application can be launched from the Linux prompt via the `gfm` command or Astrid could be switched to offline-mode.

### 5.1.1 Working Online

If you are using either of Astrid’s “online” modes (see § 4.2.1) and have selected the “DataDisplay” tab, then the data display will update as new data are obtained. Continuum and Spectral Line data are only updated when these displays are being viewed. Pointing and Focus data are always automatically updated whether or not their displays are being shown or not. Due to this feature, clicking on previous observations while Pointing and Focus scans are in progress can confuse GFM and should be avoided. The list of scans will always automatically update.

### 5.1.2 Working Offline

You can look at data that have already been taken with the GBT by running Astrid in its “offline” mode. To view data in this mode you need to follow these steps:

- Step 1.** Change the Astrid mode to “offline” (see § 4.2.1).
- Step 2.** Select **File**→**Open** from the drop-down menu in the Data Display Tab.
- Step 3.** Select a project ID from the list of project directories in `/home/gbtdata/`.
- Step 4.** Double-click `ScanLog.fits` to access the data.
- Step 5.** It may take several seconds to a few minutes to access all of your scans depending on the amount of data to load. The process is complete when you see a list of scans displayed sequentially on the left hand side of the GFM display.
- Step 6.** Click on a scan in the scan list window to process it.

### 5.1.3 Pointing and Focus Data Display

Pointing scans (from Peak, AutoPeak and AutoPeakFocus – see below) will appear under the Pointing Tab. If working “Online”, the data display will automatically process the pointing scans. **Note that clicking on previous scans while Pointing and Focus scans are in progress may interfere with automatic processing.** It will calibrate the data, remove a baseline and fit a Gaussian to the data. After the two azimuth scans it will then automatically update the GBT M&C system with the new azimuth pointing offset values that it determined. It will then automatically update the elevation pointing offset after the two elevation scans, unless certain criteria are not met (see § 5.1.3.1). A sample of the Data Display Application after a pointing is shown in Figure 5.1.

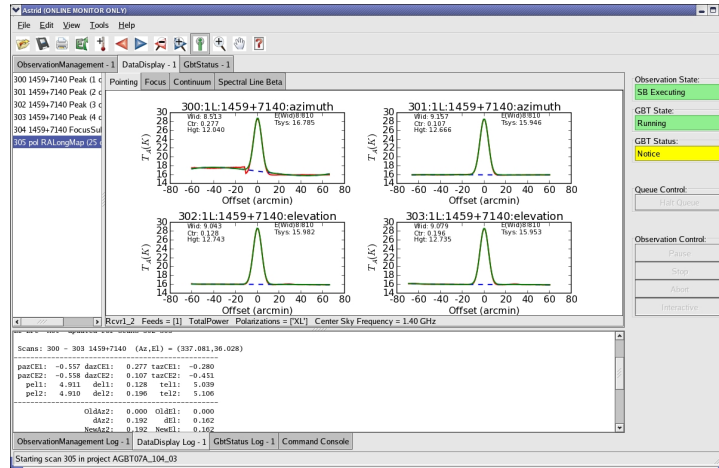


Figure 5.1: The Pointing subtab of the Astrid Data Display.

The focus scan data will appear under the Focus Tab; see Figure 5.2. Again, if “Online” the data will be processed automatically. They will be calibrated, have a baseline removed and a Gaussian will be fit to the data. The focus offset will automatically be sent to the M&C system.

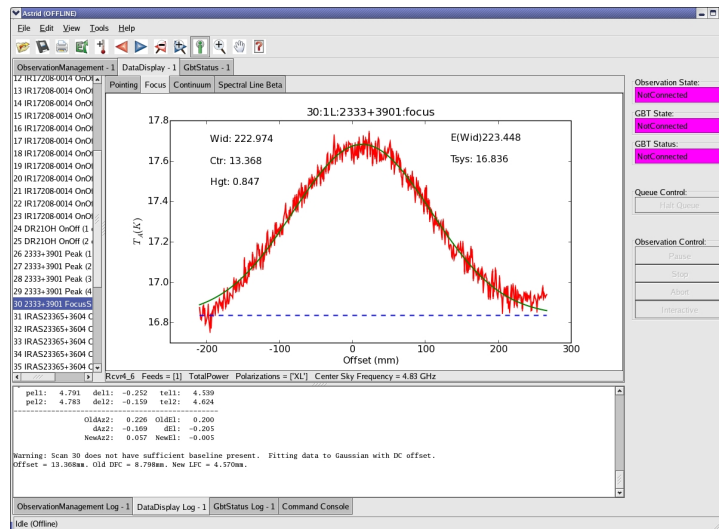


Figure 5.2: The Focus subtab of the Astrid Data Display.

The details of pointing and focus observations are described in § 6.4.1.

### 5.1.3.1 Fitting Acceptance Options

GFM has several levels of determining whether or not the pointing and focus solutions will be updated in the M&C system. The expected Full Width at Half the Maximum (FWHM) of the Gaussian fitted to the observed pointing data as the GBT slews across the source should be  $\sim 748/\nu_{\text{GHz}}$  arc-seconds where  $\nu_{\text{GHz}}$  is the observing frequency in GHz.

For a focus scan the resulting data should approximate a Gaussian with a FWHM of  $1080 \nu_{\text{GHz}}$ , in mm. The default behavior is to assume that a pointing fit is bad if the FWHM differ from the expected value by more than 30% or if the pointing correction is more than twice the FWHM in magnitude. The default for a bad focus scan is if the FWHM is more than 30% from the expected value. Users may change fitting acceptance criteria by:

**Step 1.** Select the Pointing or Focus Subtab in the Data Display.

**Step 2.** Select **Tools**→**Options...** from the drop-down menu.

**Step 3.** Select the new mode in the “Fitting Acceptance Criteria” tab of the pop-up window.

**NOTE:** Options must be set independently for both Pointing and Focus **before** each type of observation in order to take effect.

GFM recognizes the fitting acceptance criteria shown in Figure 5.3 only when Astrid is in one of its online modes. The default setting is to “Automatically accept good fits, automatically reject bad fits”. Users may also choose to never apply corrections or interactively accept bad and/or good fits. There is also an option to “Accept all automatically” which can be very dangerous and should only be used by experts.

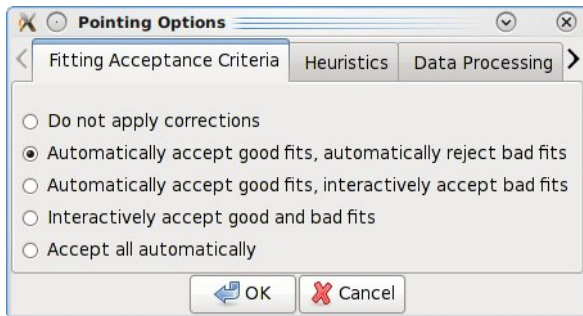


Figure 5.3: The pop-up menu to change the pointing and focus fitting acceptance criteria.

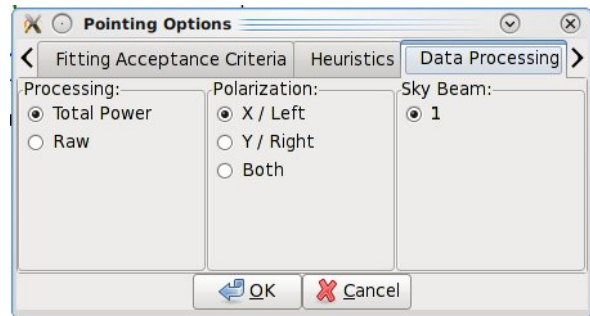


Figure 5.4: The pop-up menu to change the polarization and calibration used in pointing and focus fitting.

### 5.1.3.2 Data Processing Options

The user may change the data processing strategy, beams, and/or polarizations used by GFM in reducing pointing or focus scans. This is not needed typically since the software picks the proper default settings under normal conditions. However, for example, if the X polarization channel is faulty for some reason, one can use the Y channel instead. This can be done by:

**Step 1.** Select the Pointing or Focus Subtab in the Data Display.

**Step 2.** Select **Tools**→**Options...** from the drop-down menu.

**Step 3.** Make new data processing selections in the Data Processing Tab of the pop-up window (see Figure 5.4).

**NOTE:** Options must be set independently for both Pointing and Focus **before** each type of observation in order to take effect.

### 5.1.3.3 Heuristics Options

Heuristics is a generic term used at the GBT to quantify the “goodness of fit” of the pointing and focus data reduction solutions. Based on the known properties of the GBT, parts of the solution, such as the beam-width in pointing data, should have certain values within measurement errors. The Heuristics define how large these errors can be. The user may change the Heuristics by:

**Step 1.** Select the Pointing or Focus Subtab in the Data Display.

**Step 2.** Select **Tools**→**Options...** from the drop-down menu.

**Step 3.** Select the new mode in the Heuristics tab of the pop-up window (see Figure 5.5).

**NOTE:** Options must be set independently for both Pointing and Focus **before** each type of observation in order to take effect.

GFM allows the observer to switch between “standard”, “relaxed”, and “user-defined” heuristics. The “standard” and “relaxed” heuristic values are predefined and cannot be changed by the user. Under normal observing conditions the observer should expect to use the “standard” values. Under marginal weather conditions and/or high frequency observations “relaxed” heuristics may be appropriate. The “user-defined” heuristic values should only be used by experts. If you wish to use “user-defined” heuristics then you should contact your GBT support scientist. The default mode is “standard”.

The “standard” heuristics expect that the fitted Gaussians have a FWHM within 30% of the expected values and that the pointing solution is within twice the FWHM of the nominal location of the source. For the “relaxed” heuristics this becomes within 50% of the expected FWHM of the Gaussian fits and three times the FWHM for the pointing correction.

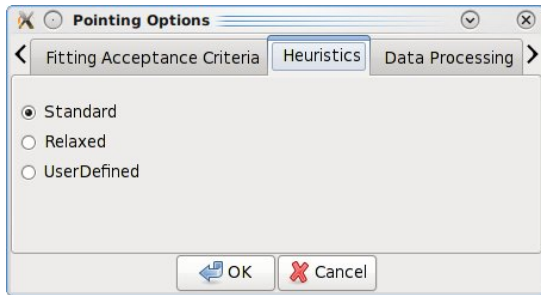


Figure 5.5: The pop-up menu to change the pointing and focus fitting heuristics.

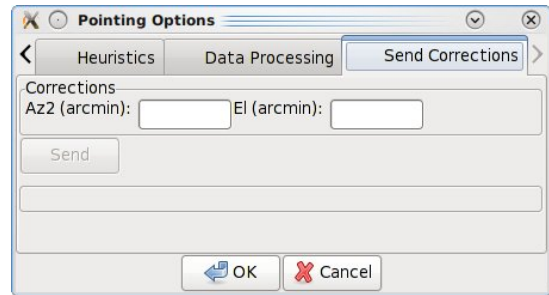


Figure 5.6: The pop-up menu to manually send pointing corrections to the telescope.

### 5.1.3.4 Send Corrections

For most observations, GFM processing produces good fits, and the solutions are automatically sent to the telescope using the default settings. However, at high frequencies (especially W-band 68-92 GHz Receiver), fits may fail, and the user may want to manually send the corrections to the telescope. The user may tell the operator to enter a solution, or they can send the corrections themselves using the Send Corrections tab. Note that corrections show up instantly within the CLEO status window (see § 5.2), but do not take effect until the start of the next scan. This can be done by:

**Step 1.** Select the Pointing or Focus Subtab in the Data Display.

**Step 2.** Select **Tools**→**Options...** from the drop-down menu.

**Step 3.** Select the Send Corrections Tab in the pop-up window (if not visible use arrow button on the right, the Send Corrections tab is farthest to the right)

**Step 4.** Enter the corrections in the text box, and click **Send** to send the solutions to the telescope. (see Figure 5.6).

### 5.1.4 OOF Data Display

Out-Of-Focus holography (OOF) is a technique for measuring large-scale errors in the shape of the reflecting surface by mapping a strong point source both in and out of focus. The procedure derives surface corrections which can be sent to the active surface controller to correct surface errors. The procedure is recommended for high-frequency observing at frequencies of 30 GHz and higher.

The AutoOOF procedure will obtain three On-The-Fly (OTF) maps, each taken at a different focus position. Processing will begin automatically upon completion of the third map, the status of which can be viewed in the progress bar under “AutoOOF Processing Status” on the right-hand-side of the screen. Once complete, the result will be displayed in the OOF subtab of the Astrid Data Display (see Figure 5.7).

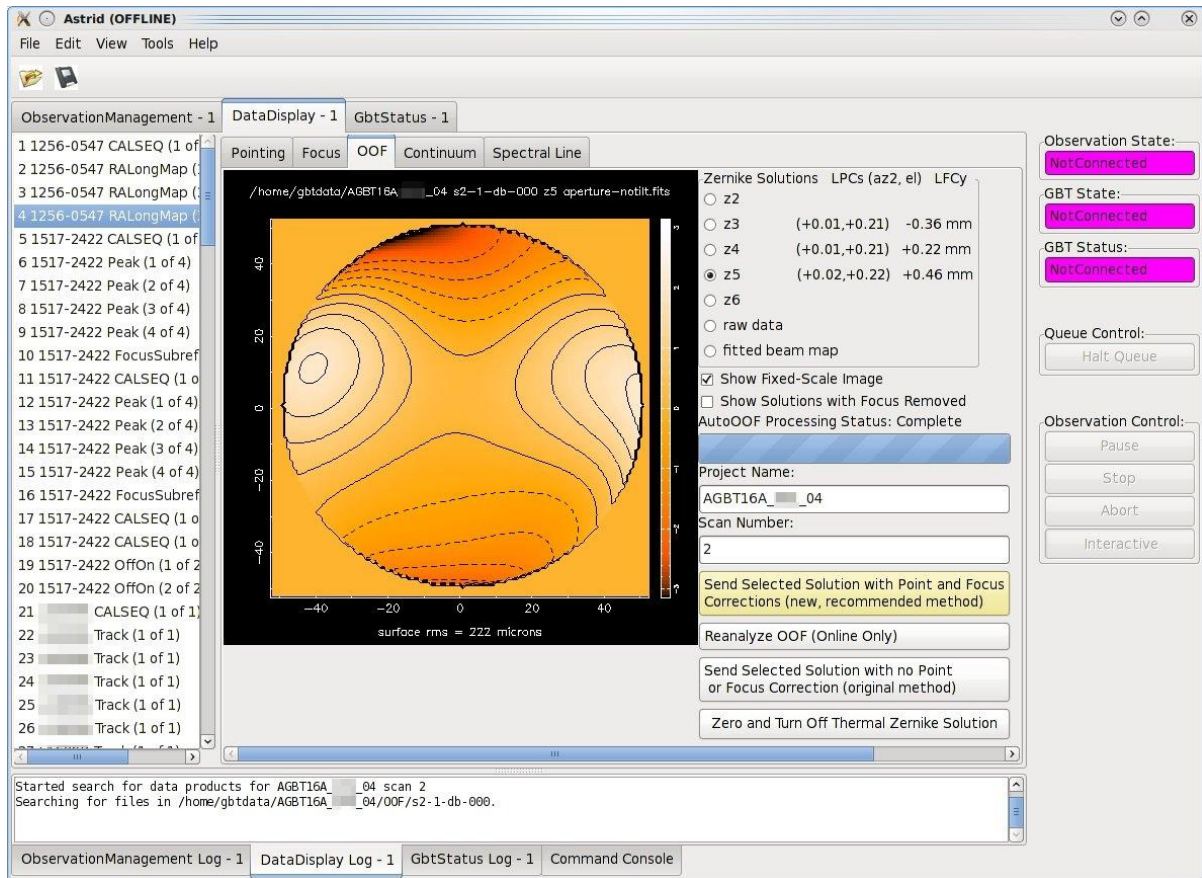


Figure 5.7: The OOF subtab of the Astrid Data Display.

Once processing is complete, the default solution displayed in Astrid is the fifth-order Zernike fit (z5). The most aggressive fit is z6, while z3 is less aggressive. Solutions may be selected and viewed via the radio buttons in the upper-right section of the screen. Derived Local Pointing Corrections (LPCs) in arcminutes, and Local Focus Corrections (LFCy) in millimeters are displayed to the right of each radio button. Raw AutoOOF data at each focus position can be viewed as a timestream and map by selecting the “raw data” radio button. The “fitted beam map” radio button will display fitted beam map images and reduced  $\chi^2$  values for the three highest orders of Zernike fits (z3, z4, and z5 by default).

Solutions must be chosen by the observer and manually sent to the active surface. Therefore, it is essential that the Zernike fits and raw AutoOOF data are examined carefully before deciding upon a solution. Steps for validating and discerning appropriate solutions be found in the following sections.

### 5.1.4.1 AutoOOF Solutions

Figure 5.8 shows examples acceptable and unacceptable OOF solutions. Good solutions have the following characteristics:

- Broad features of less than  $\pm 1.5$  radians of phase in early to mid-morning to a few radians in the afternoon. Note that you may uncheck “Show Fixed Scale Image” to view the full data range in the color bar.
- Surface rms residuals  $< 400 \mu\text{m}$ .

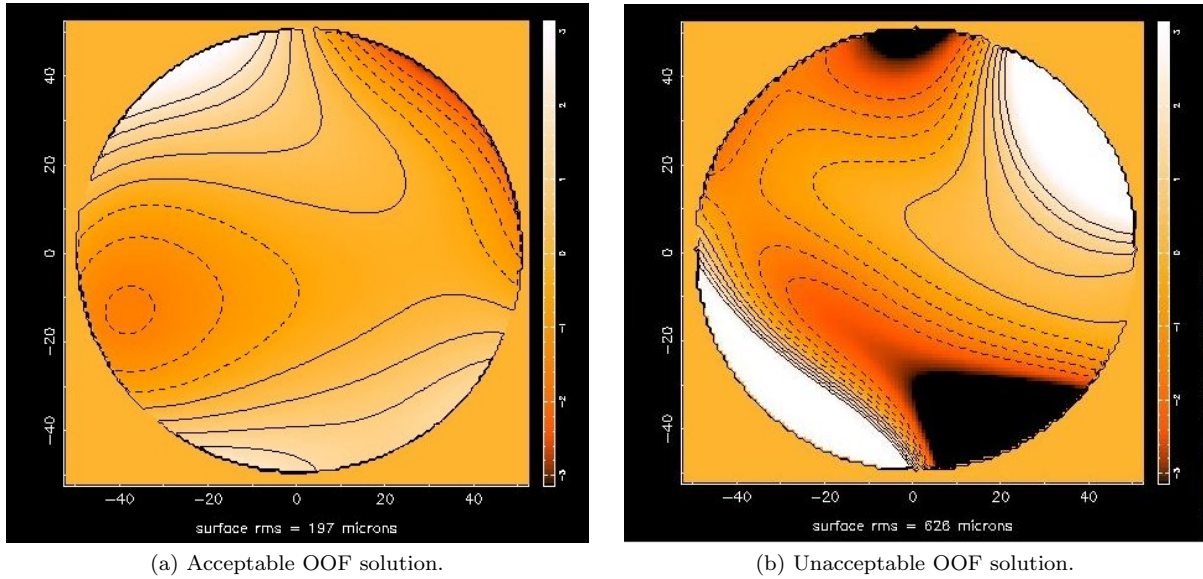


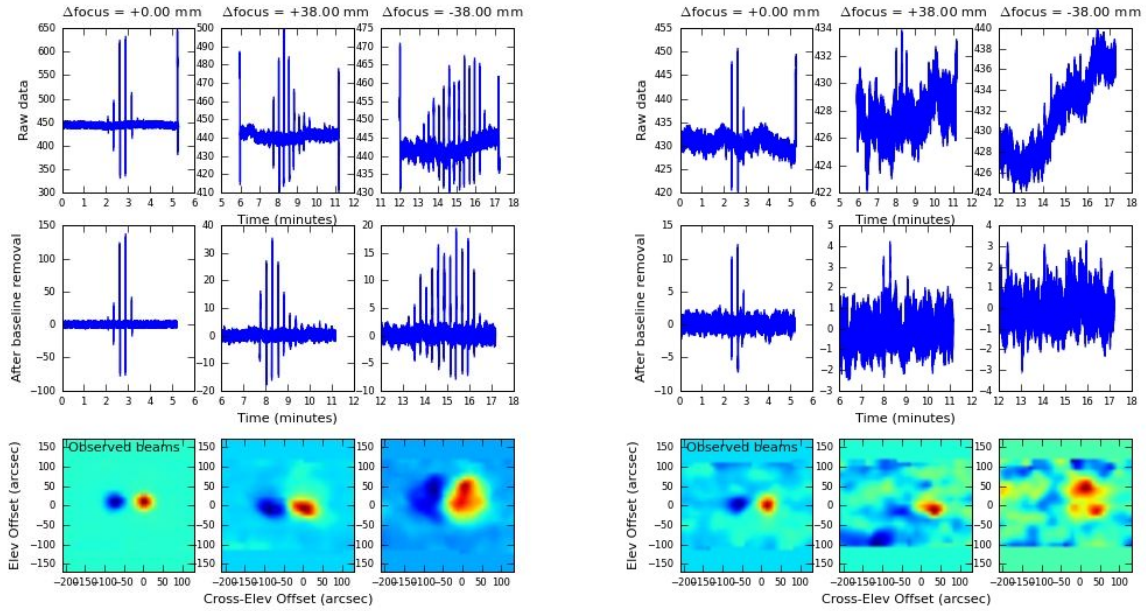
Figure 5.8: Figure 5.8a shows broad features ( $\pm 1.5$  radians of phase) with a surface rms of  $197 \mu\text{m}$ . Figure 5.8b shows steep contour lines ( $\pm 15$  radians of phase) and a surface rms of  $626 \mu\text{m}$ . This is likely the result of poor quality raw data and should not be used.

### 5.1.4.2 AutoOOF Raw Data

Although an OOF solution may appear to be reasonable (e.g., Figure 5.8a) it may also be invalid if it was derived from a bad set of raw data. Sending such a solution to the active surface could degrade performance. Therefore, observers should always check the quality of the raw AutoOOF data in order to determine whether their derived solutions are valid. For a set of raw data to be considered valid, it should show the following characteristics:

- Clear detections of the source in the raw data timestream at all focus positions.
- Symmetrical left/right positive/negative pattern in all three raw data images.
- Smooth features in all three raw data images. Sharp edges or stripes indicate hardware/software glitches or excessive winds.

The AutoOOF raw data can be viewed by selecting the “raw data” radio button in the upper-right section of the OOF Subtab of the Data Display. Each column represents one focus position. The top row is the raw timestream data from the receiver, the second row has the baselines removed, and the bottom row shows the corresponding beam maps. See Figure 5.9 for a comparison of acceptable and unacceptable raw AutoOOF data.



(a) A plot of the raw OOF data on a fairly clean Ka– band/CGB dataset. (b) A plot of raw OOF data on a source which is too faint.

Figure 5.9: A comparison of acceptable and unacceptable AutoOOF raw data.

### 5.1.4.3 Selecting the Zernike order to fit

By default, AutoOOF will halt processing after the fifth-order Zernike ( $z_5$ ) solution has been computed. The  $z_5$  solution is suitable for most conditions and is generally what observers should expect to use. A more aggressive sixth-order ( $z_6$ ) fit may also be derived at the cost of a few additional minutes of processing time. This is usually unnecessary and should only be done on bright calibrators under favorable weather conditions. See § 5.1.4.5 for information on how to change the maximum order of fit to process.

Occasionally, it may be necessary to occasionally drop to a lower order of fit if the following features are seen:

- **Large excursions** over a significant area of the dish edge in the OOF solution.
- **Regularly spaced features** around the circumference of the dish at higher order fits in the OOF solution.
- **Anomalous values in the pointing/focus LPC/LFCs** for one particular solution, or a significant jump in LPCs above a certain Zernike fit order. For example, if the focus (LPCy) values for the  $z_3$ – $z_4$  solutions are around  $-3\text{mm}$ , then abruptly jump to  $+10\text{mm}$  for the  $z_5$  solution, then it would be prudent to assume that some or all of the solutions may be invalid. It may be possible to determine which solutions are valid by examining the fitted beam maps for obvious artifacts or deviations from the observed beams (see Figure 5.10).

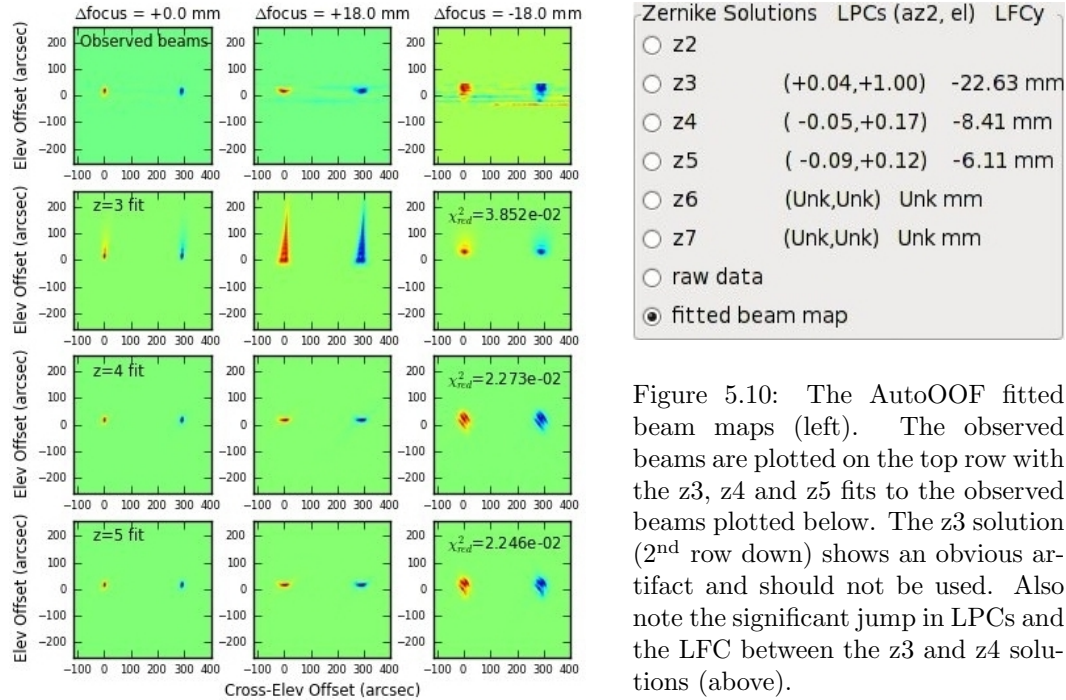


Figure 5.10: The AutoOOF fitted beam maps (left). The observed beams are plotted on the top row with the z3, z4 and z5 fits to the observed beams plotted below. The z3 solution (2<sup>nd</sup> row down) shows an obvious artifact and should not be used. Also note the significant jump in LPCs and the LFC between the z3 and z4 solutions (above).

#### 5.1.4.4 Sending a Solution to the Active Surface

When you are ready to accept the solution being displayed it will need to be manually sent to the active surface. It is recommended that when sending the solutions, you use the yellow button labeled **Send Selected Solution with Point and Focus Corrections**. If you use this option, you do not have to perform a Peak or Focus after an AutoOOF. It is still good practice to Peak and Focus at the beginning of your observing session unless you are using the W-band 68–92 GHz receiver (see Chapter 11). Subsequent pointing and focus corrections may be computed via AutoOOF.

Many high frequency observers will perform Peak scans immediately following an AutoOOF to verify the surface solution (see § 7.2). If the solution is satisfactory the LPCs and LFC from Peak/Focus scans should agree with values from the OOF solution, there should be no significant sidelobes visible in the peak scans, and Peak scans should also yield the expected beam FWHM. If in doubt, you may disable OOF corrections by pressing **Zero and Turn Off Thermal Zernike Solution** in order to compare Peak scans with and without OOF corrections.

#### 5.1.4.5 OOF Processing Options

Deriving the sixth-order Zernike (z6) solution will require a few additional minutes of processing time and for the user to manually change the maximum order of fit to process in the following way:

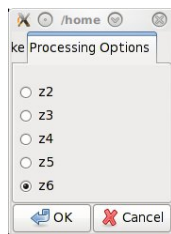


Figure 5.11: OOF Processing Options.

**Step 1.** Select the OOF Subtab of the Data Display.

**Step 2.** Select **Tools**→**Options...** from the drop-down menu.

**Step 3.** Select the maximum order of fit to process from the “Processing Options” tab of the pop-up window (Figure 5.11).

**NOTE:** All changes must be made **before submitting the SB** containing the **AutoOOF()** function in order to take effect. You may also repeat processing after making any changes by pressing **Reanalyze OOF (Online Only)**.



### 5.1.5 Continuum Data Display

Continuum data taken with the GBT that are not part of pointing and focus scans will show up in plots under the Continuum Tab (see Figure 5.12). This will show the uncalibrated continuum data as a function of time only.

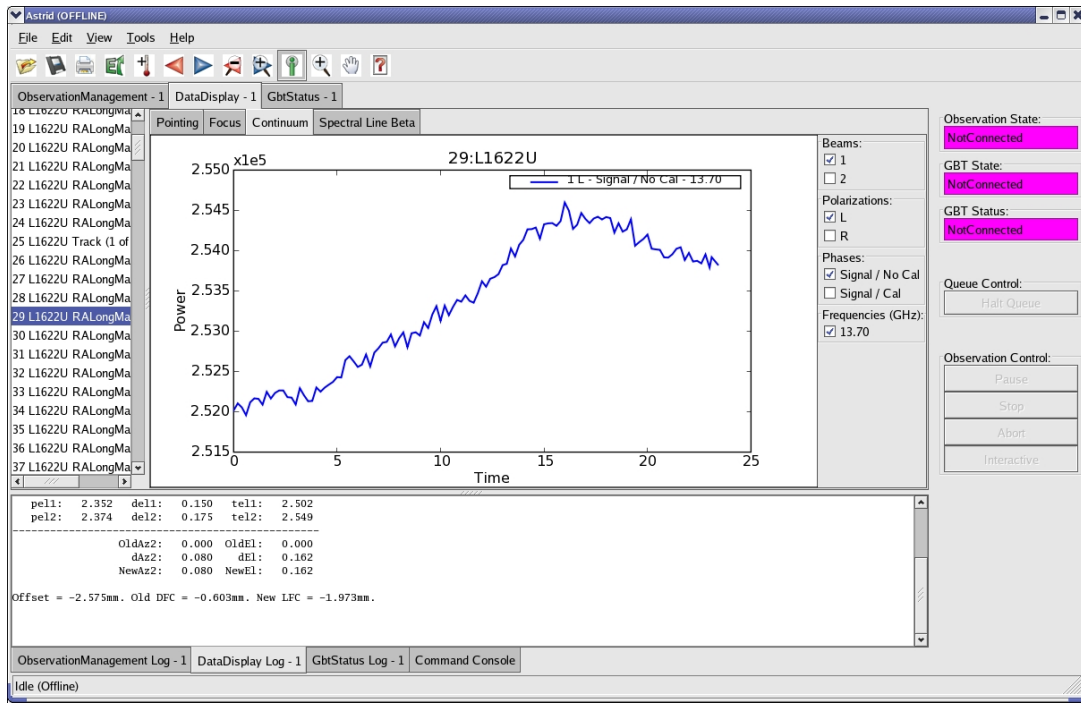


Figure 5.12: The Continuum subtab of the Astrid Data Display.

### 5.1.6 Spectral Data Display

The Spectral Line Display was a tool originally designed for browsing the previous GBT Spectrometer spectral line data. Although this tool can provide limited VEGAS information, there is a separate, more flexible web-based display intended for VEGAS. See § 5.3.1 for more details.

When viewing data online, the most recent integration is plotted automatically. Individual integrations may be selected and viewed offline. See Figure 5.13 for an example of the spectral line data display. The spectra displayed are raw data and no calibration has been applied to them. As spectra are plotted, information about each plot is printed in the console window. Each line is color coded to match the color of that spectrum in the plotting window. In addition, some of the information for the very first spectra are used to annotate the plot. The plot title is parsed as project\_name:scan\_number:integration\_number. For offline usage, the desired integration can be selected either using the up/down arrows, or by typing in a value in the edit box.

All user interaction for this plugin occurs in the right-hand side options panel. The check boxes allow selection of spectra to plot via astronomical variables: Beams, Polarizations, IF Numbers, and Phases. The options panel also includes three buttons and a radio box for plot viewing. The “Views” radio box offers options for plotting the bandpass vs. Channels or Sky Frequency. The **Keep Zoom** toggle button will maintain the current zoom, even as new spectra are plotted. Using the unzoom command (mouse right-click, or via the tool bar) will return the plot to its original scale. The **Overlay** toggle button can be used to overplot spectra from different integrations or scans. Finally, the **Clear** button erases the plot.

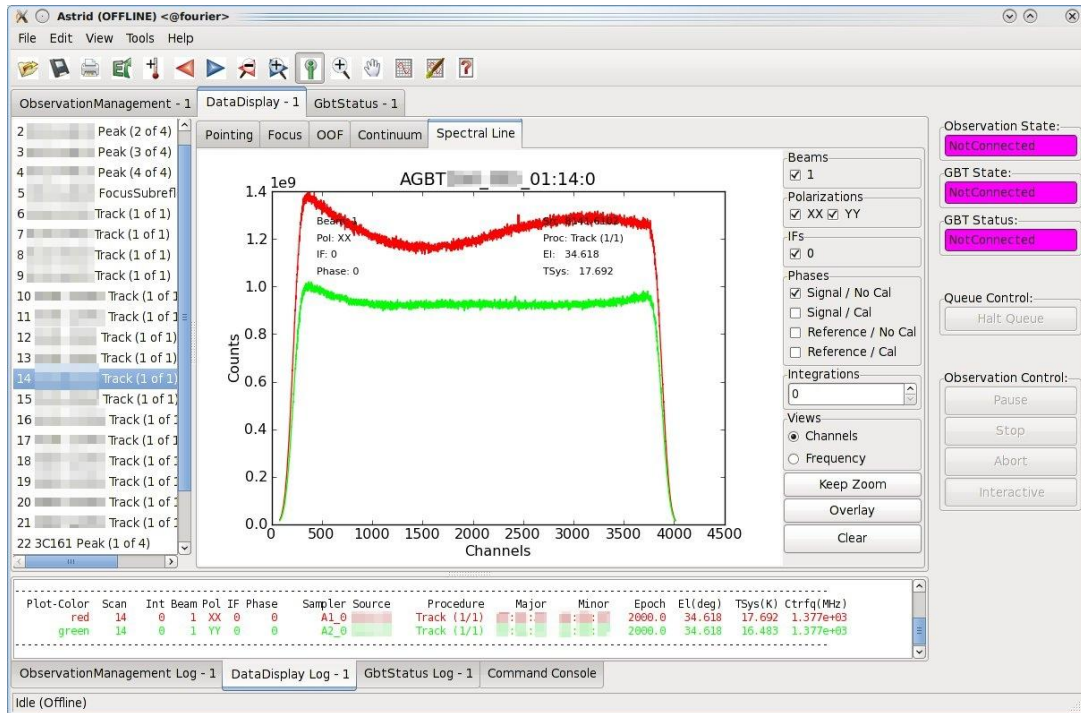


Figure 5.13: The Spectral Line subtab of the Astrid Data Display.

### 5.1.7 The Data Display Plotting Panel Toolbar

The plotting panel toolbar allows user interactions with plots in the display window and is located near the top of the Astrid Screen. The following features are available:



**Open:** Allows the user to open a previously saved session. This has the same functionality as file → open described in § 5.1.2.



**Save:** Allows the user to save output from the data display log as a text file.



**Print (DO NOT USE):** Please use the “export” function instead.



**Export:** Allows the user to save the figure displayed in the plotting panel to a file. The name must have an extension of either .png, .ps or .eps.



**Unfreeze:** Not applicable to Astrid general use. Unfreezes the processing of commands via the command line and intended for use in conjunction with the “freeze” command.



**Undo:** Undoes your last command.



**Redo:** Redoes your last command.



**Unzoom:** Undoes a previously executed zoom.



**Rezoom:** Redoes a previously executed zoom.



**Info Tool:** Selecting the info tool allows the user to use the mouse pointer to focus in/out among the available subplots (e.g., peak scans). Left-clicking the mouse brings a subplot into focus (hiding the other subplots). Right-clicking the mouse on the focused plot will show all subplots. If there is only one subplot, the info tool simply displays the mouse xy coordinates.



**Zoom Tool:** Selecting the zoom tool allows the user to use the mouse pointer for zooming in on a particular area of the plot. Left-clicking the mouse will zoom in. Right-clicking the mouse will zoom out.



**Pan Tool:** Allows the user to use the mouse pointer to pan around the selected subplot. Left-clicking the mouse and holding the left button down will pan around the subplot. Right-clicking restores the original view.



**Grid Tool:** Turns on the plot grid.



**Plot Edit Tool:** Allows the user to edit plot labels, colors, and title. Clicking on "Advanced Options" brings up an additional dialog which contains options for transparency, legend placement, and ordering of plots. Colors may be entered as hex codes or selected by clicking on the colored button to the right of the text field. Plots can only be reordered within their subplot - i.e., Y1 lines will always be below Y2 lines. Legend location can be specified with simple strings (e.g., "upper right") or coordinates 0-1 along the plot edges. If a string is chosen it will be used in place of any coordinates.



**User Manual:** Displays the DEAP user manual.

## 5.1.8 Use of Plotting Capabilities

A User Manual is available at <http://deap.sourceforge.net/help/index.html> that describes all the plotting functionality available in GFM. There is also a plotting Tutorial that illustrates the plotting capabilities by example which is available at <http://deap.sourceforge.net/tutorial/index.html>.

## 5.2 The CLEO Utilities

The Control Library for Engineers and Operators (CLEO) system provides a large number of utilities for monitoring and controlling the GBT hardware systems. Some of these are quite useful for observers, although most are intended for expert users and GBT staff.

Useful help messages pop up when you hover the mouse over any CLEO widget for a few seconds. Documentation is also available on the following web pages, but is somewhat out of date, so its best to consult your GBT “friend” for details.

- <http://www.gb.nrao.edu/~rmaddale/GBT/CLEOManual/index.html>
- <http://www.gb.nrao.edu/~rmaddale/GBT/CLEOManual/tableofcontents.html>

The following section describes just a few CLEO utilities that are useful for observers.

### 5.2.1 Starting CLEO

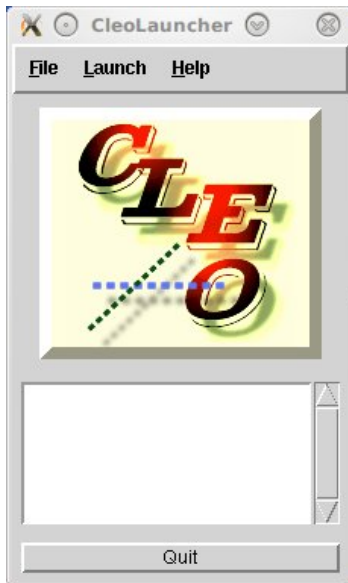


Figure 5.14: The Cleo Launcher.

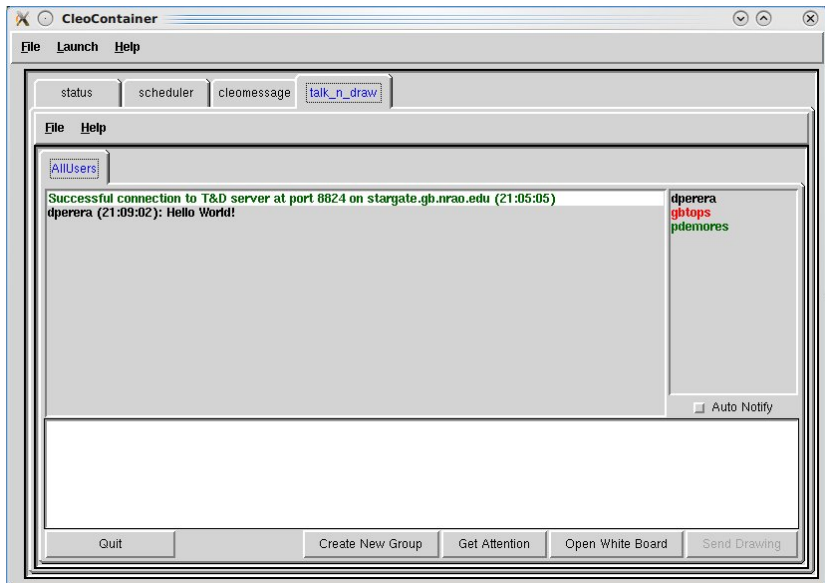


Figure 5.15: The CLEO Talk and Draw application launched as a tab of the Cleo Container.

To start CLEO, log in to any Linux workstation in Green Bank, open a terminal window, and type `cleo`. A “Cleo Launcher” window will appear (see Figure 5.14). Click on the **Launch** menu to get a list of programs that can be run. Remote observers working via VNC may prefer to use “Cleo Container” which launches and displays CLEO applications as tabs within a single window. Cleo Container can be opened from the terminal by typing `cleo cleocontainer` or from the Cleo Launcher via **Launch** → **Cleo Container...**

## 5.2.2 Talk and Draw

**Launch** → **Observer Tools** → **Talk and Draw**

Launching “Talk and Draw” will open a window that allows communication with all other users of the same application including the GBT operator (see Figure 5.15). Messages typed in the white text box near the bottom of the screen will become visible to other users after pressing the “Enter” key.

Users may also create private groups via the **Create New Group** button and inviting other users to join. The private session will be accessible through a new tab next to the default “AllUsers” tab. Users may then send messages within their newly created group or to “AllUsers” (including the operator) by selecting the relevant tab and entering a message.

## 5.2.3 Scheduler and Skyview

**Launch** → **Observer Tools** → **Scheduler & Skyview...**

This displays a plot of the sky in Az/El coordinates as viewed from Green Bank as shown in Figure 5.16. One can import a catalog of source positions to be displayed, or display one of the lists of standard calibration sources. By default it displays solar system objects. For example, to display sources listed in the standard Astrid “xband\_pointing” catalog press

- **Catalog...** → **Add/Select/DeSelect Catalogs...** → **xband\_pointing** → **Apply** → **OK**

If one selects **Schedule** (button at upper right), one may enter a date and time and display the sky for that time. It shows the corresponding LST, and moving the cursor on the plot displays the RA/Dec and Az/EL under the cursor. This is very useful for planning observations. There is also a **Real Time** option in which the location of objects and the direction the GBT is pointed are displayed for the current time.

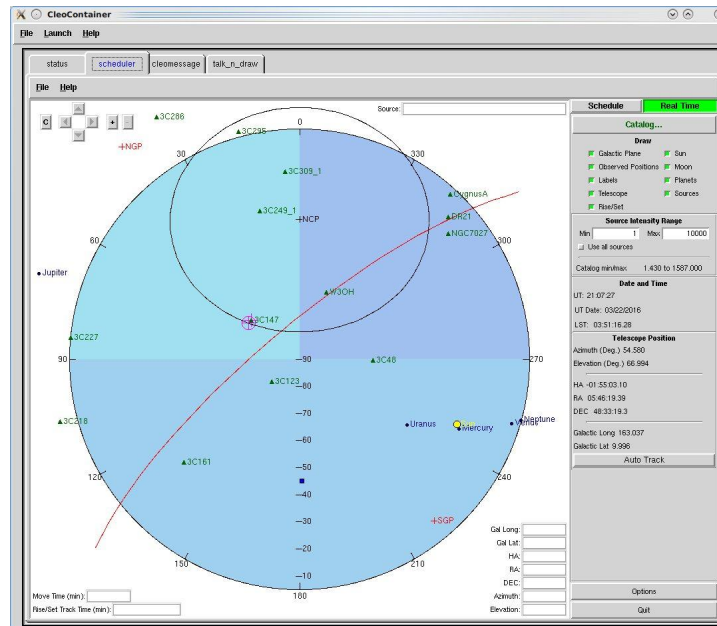


Figure 5.16: The CLEO Scheduler & Skyview application.

## 5.2.4 Status

**Launch** → **Status...**

This displays the status of many GBT systems all on one screen as shown in Figure 5.17. While very useful, it is not recommended for use remotely because it is a heavy user of computing resources. For remote observing, it is recommended to use the Astrid GbtStatus display (See Section 4.6).

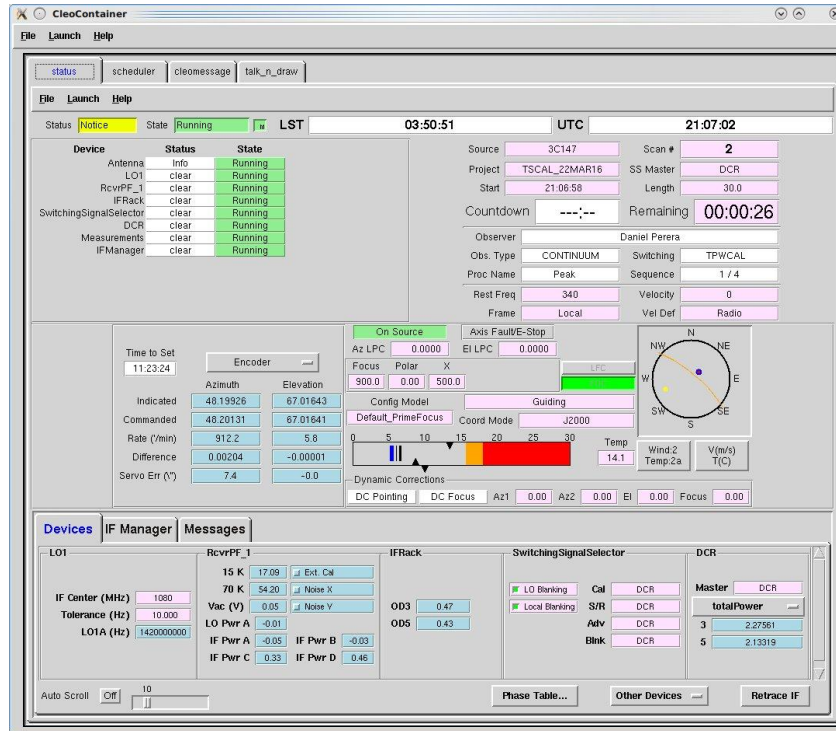


Figure 5.17: The CLEO Status application.

## 5.2.5 Messages

**Launch** → **Messages...**

This shows all system status messages as shown in Figure 5.18. It's often useful to identify problems that might arise with any of the GBT devices.

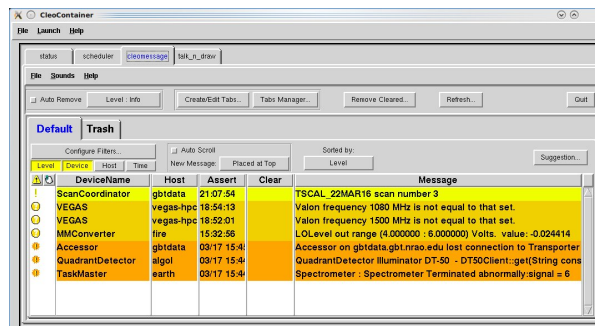


Figure 5.18: The CLEO Messages application.

## 5.3 VEGAS Monitoring Tools

### 5.3.1 The VEGAS Data Display

Observers are urged to use the real-time data display any time they are observing with VEGAS to monitor data quality in real time. The VEGAS real-time data display may be accessed by pointing a web browser to <http://vegasdisplay.gb.nrao.edu>

(This URL is only accessible to machines inside the Green Bank observing network)

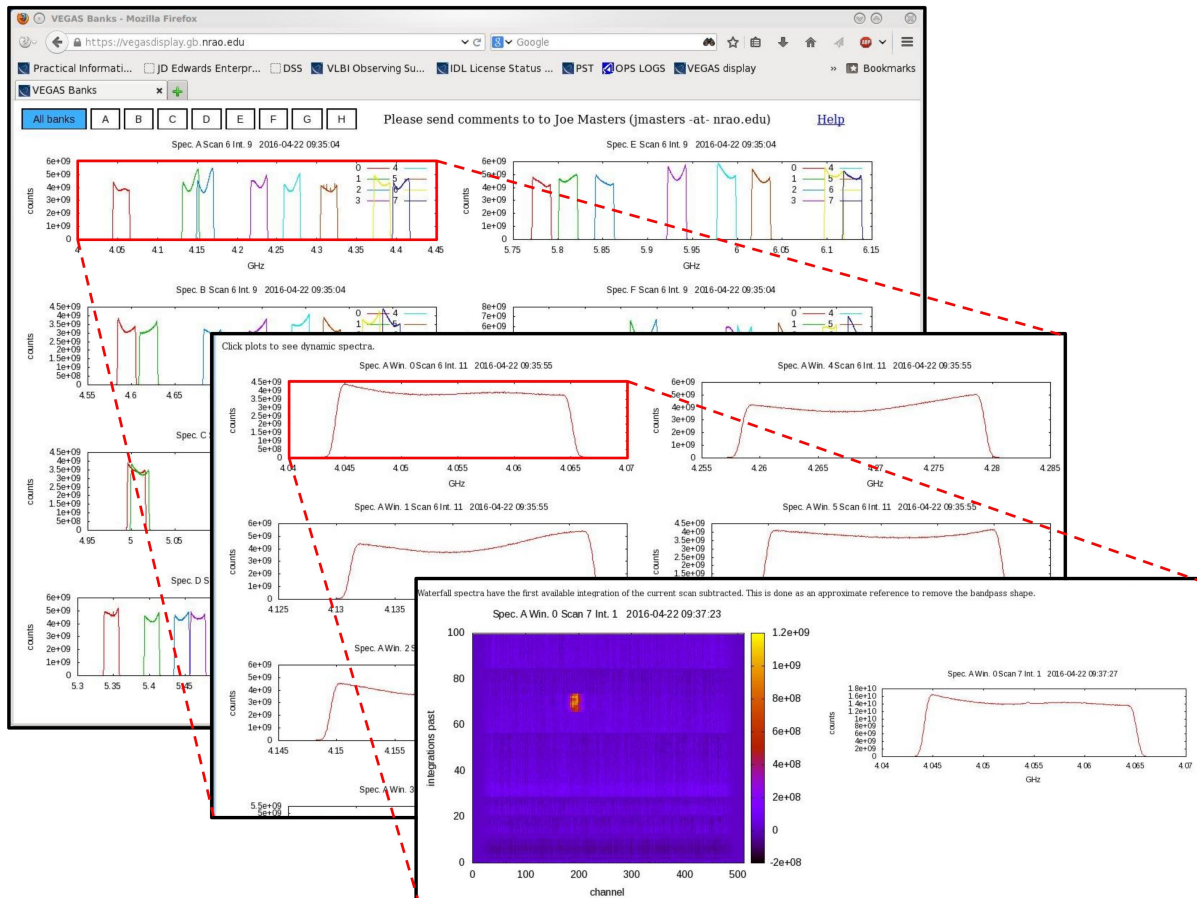


Figure 5.19: The VEGAS real-time data display. The first level displays data for each VEGAS bank, the second level displays data for each subband of a selected bank, and the third level displays a false color “waterfall” plot for a single selected subband.

This will bring up a display as shown in the top-left box of Figure 5.19 where each plot represents a bank of VEGAS. Subbands (if available) are color coded within each bank.

- Clicking on the plot for a VEGAS bank or the boxes labelled “A” → “H” near the top of the screen will display total power in counts vs. frequency for each subband of the selected bank. A maximum of eight subbands will be displayed in the plots labelled “Window 0→7”. If using a single band mode of VEGAS, then only “Window 0” will display a bandpass.
  - Clicking on a subband plot will display a false color “waterfall” plot, building up integration by integration. The right hand plot of this new screen will also display the bandpass of the latest integration. Note that the waterfall plot will always rebin data to 512 channels.

### 5.3.2 The VEGAS CLEO screen

This can be launched via **Launch** → **Backends** → **VEGAS...** from the CLEO launcher (see § 5.2). An example of the VEGAS CLEO screen is shown in Figure 5.3.2.



Figure 5.20: The VEGAS CLEO Screen

The VEGAS CLEO screen follows standard CLEO conventions, and is fairly self-explanatory. As for all backend screens, IF system information for a selected bank can be displayed by clicking on the blue square to the right of the Bank label.

The VEGAS CLEO screen can be used to launch the VEGAS Data Monitor (described in the next section) by clicking on the **VEGAS Power Monitor...** button.



### 5.3.3 VEGAS Data Monitor

The VEGAS Data Monitor (VEGASDM) provides a real-time display of the current total power level as measured by the VEGAS ADCs, as well as a histogram of the distribution of ADC counts. VEGASDM may be launched by:

```
% source /home/gbt/gbt.bash (or .csh)
% VEGASDM
```

or by clicking the `VEGAS Power Monitor...` button from the VEGAS CLEO screen. The VEGASDM display will appear as in Figure 5.21

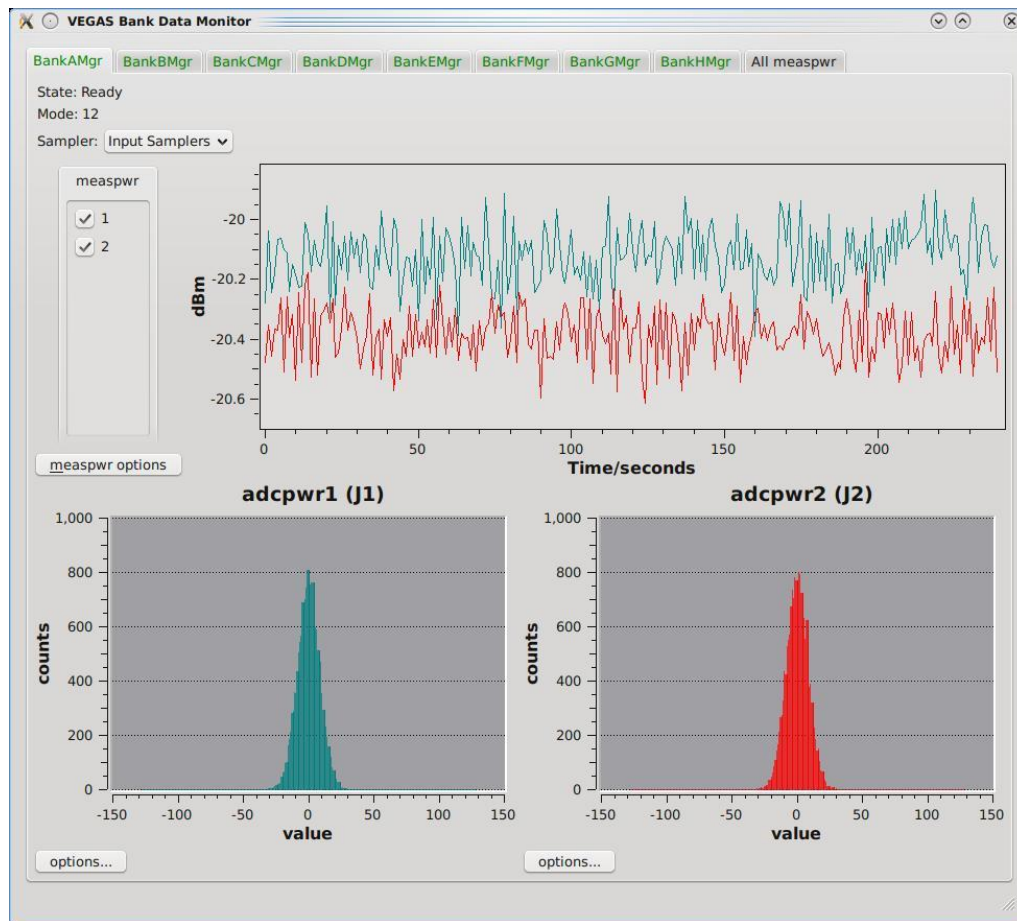


Figure 5.21: The VEGAS Data Monitor screen

VEGASDM has nine tabs, one for each Bank, and one overview tab. If a Bank is active, the tab label will be green, otherwise it will be red. Each Bank tab shows whether the Bank is in the running state, and what mode it is in. The upper plot shows the total power from each polarization as a function of time, while the lower two plots show the distribution of ADC counts for each polarization. If VEGAS is balanced correctly, the ADC counts should be approximately Gaussian, centered around zero, with a full-width half maximum of approximately 25–50 counts. If the ADC counts are very centrally peaked, there is not enough power going into VEGAS, while if the ADC counts have peaks at  $\pm 127$  counts, VEGAS is being over-driven.

The final tab of VEGASDM gives an overview plot of the total power for all eight banks on a single screen.



# Chapter 6

## Scheduling Blocks

At the GBT, we use Scheduling Blocks (SBs) to perform astronomical observations. The SB can contain information for configuring the telescope, balancing the IF system, and other commands to “tweak” the telescope system (observing directives) along with the commands (scan types) to collect observational data. Astrid interprets SBs via Python. Thus SBs should follow Python syntax rules (such as indentation for loops) and can also contain or make use of any Python commands. Here is an example of a simple SB:

```
#load the configurations file
execfile('/mypath/myconfigurations.txt')

# load catalogs file
Catalog('/mypath/mycatalog.cat')

# configure the GBT
Configure(myconfig)

# slew to the source
Slew('B0329+54')

# balance the IF system
Balance()

# now observe the source for ten minutes
Track('B0329+54', None, 600)
```

Script 6.1: A simple SB

- **execfile** loads definitions for configuring the GBT’s receivers, IF system and backends for the observations. This is described in § 6.2.
- **Catalog** loads a catalog containing information (such as positions and radial velocity, etc.) on the sources to observe. This is described in § 6.3.
- **Configure** runs the configuration defined in `myconfigurations.txt` to select the receiver and backend and set switches and frequencies. This is described in § 6.2.
- **Slew** moves the telescope to the desired source (see § 6.4).
- **Balance** balances the power levels in the IF system and backend so that they should be in their linear regime (see § 6.5).
- **Track** performs and acquires data for the desired observation. Track and other pre-defined scans are described in § 6.4.

## 6.1 Making A Scheduling Block

**Scheduling Blocks must be created well prior to your telescope time. We suggest that you review SBs with your project’s contact support scientist.**

SBs can be written using Astrid’s “Observation Management” Edit subtab (see § 4.4.1), which contains a simple text editor reminiscent of Notepad (MS Windows), or you can choose to write your SB outside of Astrid and use the “Observation Management” Import facility in Astrid to upload it into the database; see § 4.4.1.2 for details.

For the database, you should choose a descriptive name for your SB, such as “map\_G11.0” or “pointfocus”, which will remind you of the science you are trying to accomplish by running that block. Names such as “test” or “turtle.p” are not descriptive and should be avoided. The name you choose can be up to 96 characters long, and can contain white spaces, so you may have an SB name that consists of a few words (such as “K-band frequency-switched spectroscopy”). You do not need to add a suffix to your SB name (\*.sb or \*.py).

### 6.1.1 Components of a Scheduling Block

A typical Scheduling Block will include:

- A) A configuration of the system (see § 6.2).
- B) Specification of the sources via a catalog (see § 6.3).
- C) A slew to the source and then balancing the power levels, and maybe other commands (see § 6.5).
- D) Observational scan type commands (see § 6.4).

In the following sections we discuss each of these components.

## 6.2 Configuration of the GBT IF System

### 6.2.1 Overview

The routing of signals through the GBT system is controlled by many electronic switches which eliminate the need to physically change cables by hand. The GBT’s electronically configurable IF system allows many, and more complicated paths for the signals to co-exist at all times. Configuring the GBT IF system can usually be accomplished in under one minute.

### 6.2.2 Defining and Executing A Configuration

Configurations are defined as sets of keyword–value pairs within a single string variable. To execute a configuration, this variable is passed as an argument into the **Configure()** command in an SB. Configurations may be defined in two ways:

1. The configuration definition may reside on a text file external to the SB. It can then loaded into the SB via the **execfile()** command (see script 6.3).
2. The configuration may be explicitly defined within the SB (see script 6.4).

We usually recommend that configuration definitions reside on text files external to the SB. This allows for configurations to be changed on the file without the need to re-validate and re-save the SB (see § 4.4.1.4). It also allows for simple SBs without clutter. Note that you should always use **execfile** rather than **import**, as **import** will not reread the file and miss any changes that you may have made.

Explicitly defining configurations within SBs allows users to easily edit and view their configuration from Astrid. If this method is chosen, users **must** re-validate and re-save the SB if any changes are made. Examples of both methods can be seen in scripts 6.3 and 6.4.

If using multiple configurations, it is recommended that you define them all in one text file and load them into the SB via a single **execfile** command. You may then use **Configure()** to execute each configuration as necessary.

```
#the configuration file syntax.py

myconfiguration = '''
# This is a comment (ignored by software)
primarykeyword1 = your primarykeyword value
primarykeyword2 = your primarykeyword value
. . .
. . .
primarykeywordN = your primarykeyword value
'''
```

Script 6.2: A text file containing an example of basic configuration syntax.

```
# An SB to configure only

# Configuration is defined
# in an external file.

execfile('mypath/syntax.py')
Configure(myconfiguration)
```

Script 6.3: An SB using **execfile()** to load the contents of Script 6.2 into Astrid. “myconfiguration” can then be used as an argument to **Configure()** to configure the system.

```
# An SB to configure only - Configuration is defined here

myconfiguration = '''
# This is a comment (ignored by software)
primarykeyword1 = your primarykeyword value
primarykeyword2 = your primarykeyword value
. . .
. . .
primarykeywordN = your primarykeyword value
'''

Configure(myconfiguration)
```

Script 6.4: An SB that explicitly defines all configuration keywords within triple quotes as the parameter “myconfiguration”. This parameter is then used as an argument to **Configure()** in order to configure the system. This SB will perform exactly the same function as the text file and SB shown in Scripts 6.2 and 6.3.

### 6.2.3 Basic Configuration Syntax

An example of the basic configuration syntax is shown in scripts 6.2 and 6.4. Configurations are passed into **Configure()** as a string argument. For each configuration, all keywords and values (§ 6.2.5) exist as line separated keyword=value pairs, all enclosed within a single set of triple-quotes.

## 6.2.4 Example Configurations

The best way to learn about how to define and perform configurations is through examples. Keywords available for use in a configuration definition will be discussed in § 6.2.5 and all examples have been placed in the directory,

```
/home/astro-util/projects/GBTog_examples/configs/
```

### Continuum Observations

```
# configuration definition for continuum observations

continuum_config=''
receiver = 'Rcvr1_2'
beam     = '1'
obstype  = 'Continuum'
backend  = 'DCR'
nwin     = 1
restfreq = 1400
bandwidth = 80
swmode   = 'tp'
swtype   = 'none'
swper    = 0.2
tint     = 0.2
vframe   = 'topo'
vdef     = 'Radio'
noisecal = 'lo'
pol      = 'Linear'
'''
```

Script 6.5: An example continuum configuration.

The above configuration definition (script 6.5) has been given the name ‘continuum\_config’ and can be used for pointing and focusing observations or for continuum mapping. We have configured for the following:

- Total power, continuum observations [obstype=‘Continuum’; swmode=‘tp’; swtype=‘none’].
- The single beam L-band (1 to 2 GHz) receiver [receiver=‘Rcvr1.2’; beam=‘1’]
- The DCR as the backend detector [backend=‘DCR’]
- Take data using a single band centered on 1400 MHz with a 80 MHz bandwidth [nwin=1; restfreq=1400; bandwidth=80]
- Go through a full switching cycle in 0.2 seconds [swper=0.2]
- Record data with the DCR every 0.2 seconds [tint=0.2]
- Disable doppler tracking for continuum observations [vframe=‘topo’; vdef=‘Radio’].
- Use a low-power noise diode [noisecal=‘lo’]
- Linear polarization [pol=‘Linear’]

## Spectral Line, Frequency Switching Observations

```
# configuration definition for spectral line observations
# using frequency switching

fs_config=''
receiver = 'Rcvr1_2'
obstype  = 'Spectroscopy'
backend   = 'VEGAS'
restfreq  = 1420
bandwidth = 23.44
nchan     = 65536
vegas.subband = 1
swmode    = 'sp'
swtype    = 'fsw'
swper     = 2.0
swfreq    = 0, -5.0
tint      = 10
vframe    = 'lsrk'
vdef      = 'Radio'
noisecal  = 'lo'
pol       = 'Linear'
'''
```

Script 6.6: An example frequency switched, spectral line configuration.

The above example (script 6.6) will configure for the following:

- Frequency switching, spectral line observations [obstype='Spectroscopy'; swmode='sp'; swtype='fsw'].
- The single beam L-band (1 to 2 GHz) receiver [receiver='Rcvr1\_2']. Note that not specifying 'beam' defaults to [beam='1'].
- VEGAS as the backend detector using linear polarization without cross-polarization products [backend='VEGAS'; pol='Linear'].
- Take data using a single band using VEGAS mode 11 (see Table 8.1) defined by a 23.44 MHz bandwidth, 65536 channels, and one band per spectrometer. [bandwidth=23.44; nchan=65536; vegas.subband=1], centered on 1420 MHz [restfreq=1420].
- Go through a full switching cycle in 2 seconds [swper=2.0]. Over one cycle, the frequency switching states will be centered on the line, and then be shifted by -5 MHz [swfreq=0,-5.0].
- Record data with VEGAS every 10 seconds [tint=10]
- Doppler track the spectral line with the rest frequency 1420 MHz in the commonly used Local Standard of Rest velocity with the radio definition of Doppler tracking [vframe='lsrk'; vdef='Radio'].
- Use a low-power noise diode [noisecal='lo']

## Multiple Spectral Lines, Total Power Observations

```
# configuration definition for multiple spectral line observations
# using total power switching
tp_config=''
receiver = 'Rcvr8_10'
obstype = 'Spectroscopy'
backend = 'VEGAS'
restfreq = 9816.867, 9487.824, 9173.323, 8872.571,
           9820.9, 9821.5, 9822.6, 9823.4, 9824.6
dopplertrackfreq = 8873.1
bandwidth = 23.44
nchan = 8192
swmode = 'tp'
swtype = 'none'
swper = 1.0
tint = 30
vframe = 'lsrk'
vdef = 'Radio'
noisecal = 'lo'
pol = 'Circular'
''
```

Script 6.7: An example total power, spectral line configuration.

The above example (script 6.7) will configure for the following:

- Total power, spectral line observations [obstype='Spectroscopy'; swmode='tp'; swtype='none'].
- The single beam X-band (8 to 10 GHz) receiver. [receiver='Rcvr8\_10']. Note that not specifying 'beam' defaults to [beam='1'].
- VEGAS as the backend detector using circular polarization without cross-polarization products [backend='VEGAS'; pol='Circular'].
- Mode 21 of VEGAS (see Table 8.1). This mode is defined by a bandwidth of 23.44 Mhz, 8192 spectral channels in the eight subband mode of VEGAS [bandwidth=23.44; nchan=8192]. Note that not specifying 'vegas.subband' for a bandwidth of 23.44 MHz will default to vegas.subband=8.
- 9 spectral windows, each of which centered on one of the 9 frequencies (in MHz) listed under restfreq [restfreq=9816.867, 9487.824, 9173.323, ....].
- Go through a full switching cycle in 1 second [swper=1.0] and record data with VEGAS every 30 seconds [tint=30].
- Doppler track the spectral line with the rest frequency 8873.1 MHz [dopplertrackfreq=8873.1] in the commonly used Local Standard of Rest velocity [vframe='lsrk'] with the radio definition of Doppler tracking [vdef='Radio'].
- Use a low-power noise diode [noisecal='lo'].



## Multiple Spectral Lines, Multi-beam, Total Power Observations

```
# configuration definition for spectral line observations
# using a multi-beam receiver
tp_config_multi_beam = '''
receiver = 'Rcvr40_52'
beam     = '1,2'
obstype  = 'Spectroscopy'
backend  = 'VEGAS'
restfreq = 44580, 43751, 45410, 46250
deltafreq = 0,100,0,0
bandwidth = 1500
nchan    = 16384
swmode   = 'tp'
swtype   = 'none'
swper    = 1.0
tint     = 10
vframe   = 'lsrk'
vdef     = 'Radio'
noisecal = 'lo'
pol      = 'Circular'
'''
```

Script 6.8: An example total power, spectral line configuration for a multi-beam receiver.

The above example (script 6.8) will configure for the following:

- Total power, spectral line observations [obstype='Spectroscopy'; swmode='tp'; swtype='none'].
- The dual beam Q-band (40 to 52 GHz) receiver using both beams [receiver='Rcvr40\_52'; beam='1,2'].
- VEGAS as the backend detector using circular polarization without cross-polarization products [backend='VEGAS'; pol='Circular'].
- Mode 2 of VEGAS (see Table 8.1). This mode is defined by a bandwidth of 1500 Mhz with 16384 spectral channels [bandwidth=1500; nchan=16384].
- 4 spectral windows, each of which centered on one of the 4 frequencies (in MHz) listed under restfreq [restfreq=44580, 43751, 45410, 46250].
- Shift the window centered on 43751 MHz by 100 MHz in the local (topocentric) frame. Thus, this window will now be centered on 43851 MHz [deltafreq=0,100,0,0]. deltafreq should be defined in the same manner as restfreq: This example uses 4 comma separated values.
- Go through a full switching cycle in 1 second [swper=1.0] and record data with VEGAS every 10 seconds [tint=10].
- Doppler track the spectral line with the rest frequency 44580 MHz (default is the first specified rest frequency) in the commonly used Local Standard of Rest velocity [vframe='lsrk'] with the radio definition of Doppler tracking [vdef='Radio'].
- Use a low-power noise diode [noisecal='lo'].

## Multiple Spectral Lines, KFPA Observations

```
#configuration definition for spectral line observations using the KFPA
kfpa_config=''
receiver = 'RcvrArray18_26'
beam     = 'all'
obstype  = 'Spectroscopy'
backend  = 'VEGAS'
restfreq = {24600:'1,2,3,4', 23900:'5,6,7', 25500 : '-1',
            'DopplerTrackFreq': 24700}
deltafreq = {24600:-100, 23900:0, 25500:0}
bandwidth = 187.5
nchan     = 32768
swmode    = 'tp'
swtype    = 'none'
swper     = 1.0
tint      = 30
vframe    = 'lsrk'
vdef      = 'Radio'
noisecal  = 'lo'
pol       = 'Circular'
vegas.vpol= 'cross'
'''
```

Script 6.9: An example total power, spectral line configuration for the KFPA.

The above example (script 6.9) will configure for the following:

- Total power, spectral line observations [obstype='Spectroscopy'; swmode='tp'; swtype='none'].
- The KFPA (18 to 26 GHz) receiver using all 7 beams [receiver='RcvrArray18\_26'; beam='all'].
- VEGAS as the backend detector with circular cross-polarization products [backend='VEGAS'; vegas.vpol='cross'; pol='Circular'].
- Mode 4 of VEGAS (see Table 8.1). This mode is defined by a bandwidth of 187.5 Mhz with 32768 spectral channels [bandwidth=187.5; nchan=32768].
- 3 spectral windows centered on 24600, 23900, and 25500 MHz. Data will be recorded for beams 1→4 using the first window (24600 MHz) while beams 5→7 will use the second window (23900 MHz). An additional IF path will be routed from beam 1 to the window centered on 25500 MHz. This is known as the “7+1” mode of the KFPA (see Chapter 10) [restfreq=24600:'1,2,3,4',23900:'5,6,7',25500:'-1','DopplerTrackFreq': 24700].
  - Note that doppler tracking the center (24700 MHz) of the full frequency range (25500-23900+bandwidth) is necessary in this example. The maximum frequency separation limitation of the KFPA is 1.8 GHz when using multiple beams (see Chapter 10). The Radio definition of doppler tracking has been used in the Local Standard of Rest Velocity [vframe='lsrk'; vdef=Radio]
- Shift the window centered on 24600 MHz by -100 MHz in the local (topocentric) frame. Thus, this window will now be centered on 24500 MHz [deltafreq=24600:-100, 23900:0, 25500:0]. deltafreq should be defined using the same syntax as restfreq: This example uses Python dictionary syntax.
- Go through a full switching cycle in 1 second [swper=1.0] and record data with VEGAS every 30 seconds [tint=30].
- Use a low-power noise diode [noisecal='lo'].

## Advanced Use of the Restfreq Keyword

```
# Example for the advanced usage of restfreq
adv_restfreq_config=''
receiver = 'Rcvr12_18'
beam = '1,2'
obstype = 'Spectroscopy'
backend = 'VEGAS'
swmode = 'tp'
swtype = 'none'
swper = 1.0
tint = 10
vframe = 'lsrk'
vdef = 'Radio'
noisecal = 'lo'
pol = 'Circular'
bandwidth = 23.44
nchan = 32768
dopplertrackfreq = 13500.0
restfreq = [
  {'restfreq':14000,'bank':'A','bandwidth':1500,'nchan':1024,'beam':'1'},
  {'restfreq':14000,'bandwidth':1500,'nchan':1024,'beam':'2'},
  {'restfreq':13000,'bandwidth':187.5,'nchan':32768,'beam':'1'},
  {'restfreq':13100,'bandwidth':187.5,'nchan':32768,'beam':'2',
    'vpol':'cross','deltafreq':1},
  {'restfreq':13200,'bank':'C','bandwidth':23.44,'res':0.7,'beam':'1',
    'subband':8},
  {'restfreq':13300,'bank':'C','bandwidth':23.44,'res':0.7,'beam':'1',
    'subband':8},
  {'restfreq':13400,'bank':'C','bandwidth':23.44,'res':0.7,'beam':'1',
    'subband':8},
  {'restfreq':13400,'bandwidth':23.44,'res':0.7,'beam':'2',
    'subband':1},
  {'restfreq':13500,'bandwidth':100,'nchan':32768,'beam':'1'},
  {'restfreq':13500,'bandwidth':100,'nchan':32768,'beam':'2'}]
'''
```

Script 6.10: An example showing advanced use of the restfreq keyword.

The above example (Script 6.10) uses the advanced restfreq syntax (an array of Python dictionary terms) to more precisely configure the GBT system. Note that this is an example of usage only and it is not recommended that users attempt to manually route beams to specific VEGAS banks.

When using the advanced restfreq syntax, it is important to be aware of the following details in the main configuration block:

- Key values specified in the restfreq dictionary term override key-values pairs in the main configuration. If no values for a key have been specified, a default value will be used if available.
- **bandwidth** and **nchan** must always be specified in the main configuration block outside of restfreq. This is required for the configuration to pass validation, even if such values are redundant [bandwidth=23.44; nchan=32768].
- **dopplertrackfreq** must be set by the user [dopplertrackfreq=13500.0] since there is no default doppler tracking frequency for the advanced restfreq syntax.

The following points give details on the usage of the advanced restfreq syntax in this example:

- Multiple rest frequencies (or windows centered on a rest frequency) are input as an array of Python dictionary terms. The `'restfreq'` dictionary key is the minimum required entry for each dictionary term and specifies the center of each window. Each bank may also be configured with different resolution, bandwidth, and number of spectral windows. However, the integration time, switching period and frequency switch must be the same for all banks.
- Each window may be routed to a specific bank (VEGAS spectrometer) with the `'bank'` dictionary key (see the first window of this example). By omitting `'bank'`, the system will attempt to route windows to available banks automatically (recommended). Note that certain restrictions exist when routing multi-beam receivers to VEGAS banks. See § 8.3 and § 10.1 for further information.
- The `'beam'` dictionary key specifies which beam is used for the window. Omitting `'beam'` defaults to beam 1 [`'beam':'1'`].
- VEGAS modes are set for a window by defining valid combinations of bandwidth and resolution, and the number of sub-bands if using a 23.44 MHz bandwidth (see Table 8.1). If these values are not defined as dictionary keys, then values defined in the main configuration block or default values will be used. It is worth noting the following points in this example:
  - Bank C has been split into 3 subbands and uses VEGAS mode 23 defined by 23.44 MHz bandwidth, 8 subbands, and 0.7 kHz resolution [`'bank':'C', 'bandwidth':23.44, 'res':0.7, 'subband':8`]. The 3 windows are centered on 13200, 13300, and 13400 MHz. **Note that all sub-bands within a single bank must use identical VEGAS settings apart from the center frequency and offset.**
  - A second window has been centered around 13400 MHz using a bandwidth of 23.44 MHz with 0.7 kHz resolution. However, this window is configured to use beam 2 and mode 10 of VEGAS with a single sub-bank [`'restfreq':13400,'bandwidth':23.44,'res':0.7,'beam':'2','subband':1`]
  - The window centered at 13100 MHz gives an example of the other dictionary keys available. This window has been shifted +1 MHz in the local frame [`'deltafreq':1`] to be centered on 13101 MHz. Data will be recorded with full Stokes polarization products [`'vpol':'cross'`]. All other windows will record data with total intensity polarization products [`vegas.vpol='self'` (the default setting)]

## 6.2.5 Configuration Keywords

### 6.2.5.1 Keywords That Must Always Be Present

The following keywords do not have default values and must be present in all configuration definitions.

**receiver** This keyword specifies the name of the GBT receiver to be used. The names and frequency ranges of the receivers can be found in Table 6.1. The value of the receiver keyword is a string and should therefore be placed within quotes when used.

Table 6.1: GBT receivers and their nominal frequency ranges.

Name	Frequency Range (GHz)	Notes
Rcvr_342	.290-.395	PF1 feed
Rcvr_450	.385-.520	PF1 feed
Rcvr_600	.510-.690	PF1 feed
Rcvr_800	.680-.920	PF1 feed
Rcvr_1070	.910-1.23	PF2
Rcvr1.2	1.15-1.73	L-band
Rcvr2.3	1.73-2.60	S-band
Rcvr4.6	3.95-7.8	C-band
Rcvr8_10	8.00-10.0	X-band
Rcvr12_18	12.0-15.4	Ku-band
RcvrArray18_26	18.0-26.5	KFPA 7-beam focal plane array
Rcvr26_40	26-31, 30.5-37, 36-40	Ka-band
Rcvr40_52	40.5-47.0	Q-band
Rcvr68_92	67-74, 73-80, 79-86, 85-93	W-band
Rcvr_PAR	80-100	MUSTANG Bolometer Array
NoiseSource	N/A	

**backend** This keyword specifies the name of the backend (data acquisition system) to be used. The value for this keyword is a string. Valid backends are listed in Table 6.2.

Table 6.2: GBT backends.

Name	Notes
DCR	The Digital Continuum Receiver directly from the IF Rack. One frequency available.
DCR_AF	The Digital Continuum Receiver from the Analog Filter Rack Four/two frequencies maximum for single/dual beam receivers.
VEGAS	Spectral line backend with up to 524288 channels and 64 frequencies with various bandwidths.
VLBA_DAR	Very Long Baseline Array Data Acquisition Recorder.
Radar	For bi-static radar observations. Private backend.
CCB	CalTech Continuum Backend
GUPPI	Green Bank "Ultimate" Pulsar Processor.

**obstype** This keyword specifies the type of observing to be performed. The allowed values are one of the following strings: ‘Continuum’, ‘Spectroscopy’, ‘Pulsar’, ‘Radar’, ‘VLBI’.

**bandwidth** This keyword gives the bandwidth in MHz to be used by the specified backend. The value of the keyword should be a float. Possible values depend on the receiver and backend that are chosen (see Table 6.3 and Table 8.1).

Table 6.3: Allowable bandwidths for backends.

Backend	Receiver	Possible Bandwidths (MHz)
VEGAS	Any	1500, 1000, 187.5, 100, 23.44, 16.9, 11.72
DCR, VLBI, or Radar	Prime Focus	20, 40, 80, 240
DCR, VLBI, or Radar	Rcvr1_2, Rcvr4_6, Rcvr8_10, Rcvr12_18	20, 80, 320, 1280
DCR, VLBI, or Radar	Rcvr2_3, RcvrArray18_26, Rcvr40_52	80, 320, 1280
DCR_AF	Any	12.5, 50, 200, 800
CCB	Rcvr26_40	600
GUPPI	Any	100, 200, 800

**restfreq** This keyword specified the rest frequencies for spectral line observations or the center frequencies for continuum observations. There are three available syntaxes for restfreq:

### 1. Simple

```
restfreq = 1420, 1661, 1667
deltafreq = 0, 5, 0
```

The above example sets 3 rest frequencies and offsets the second window (1661 MHz) by +5 MHz in the local (topocentric) frame using deltafreq. Rest frequencies may be specified as a list of comma separated float values (MHz). This syntax should be used when all beams (including single beam receivers) are configured to observe the same rest frequencies and VEGAS does not need to use an advanced configuration (see “Advanced” below). Note that:

- **deltafreq** can also be specified using the same syntax as restfreq, a single global offset, or omitted to use the default value of zero.
- If **dopplertrackfreq** is not set in the main configuration block then the first rest frequency listed using this syntax will be doppler tracked by default.

### 2. Multi-beam

```
restfreq={24000:'1,2,3,4',23400:'5,6',25000:'7',
          'dopplertrackfreq':24200}
#deltafreq must be specified with this syntax - even when zero
deltafreq = {24000:0, 23400:0, 25000:0}
```

The above example specifies a rest frequency of 24000 MHz for beams 1–4, 23400 MHz for beams 5 and 6, and 25000 MHz for beam 7. Different feeds of multi-beam receivers may be tuned to different rest frequencies. Rest frequencies and delta frequencies are input as Python dictionaries. Further information on this syntax and example can be found in Appendix B. Note that:

- **deltafreq** must **always** be specified as a separate Python dictionary, even when zero.
- **dopplertrackfreq** must **always** be specified in the restfreq Python dictionary.

### 3. Advanced

```
bandwidth = 23.44
nchan      = 32768
dopplertrackfreq = 1420.0
restfreq = [{'restfreq':1420.0},
             {'restfreq':1420.0, 'deltafreq':-20.0},
             {'restfreq':1667.0, 'bandwidth':11.72, 'nchan':65536}]
```

The above example will configure VEGAS to use 3 rest frequencies. The first two windows are centered on 1420 MHz with mode 23 of VEGAS using bandwidth=23.44 and nchan=32768 from the main configuration block (8 subbands are selected by default for bandwidth=23.44). However, `deltafreq` has been used as a dictionary key to offset the second window by -20 MHz in the local topocentric frame. A third window is centered on 1667 MHz with mode 16 of VEGAS using the ‘bandwidth’ and ‘nchan’ dictionary keys to override values from the main configuration block.

This syntax may be used to more precisely configure VEGAS observations and specifies `restfreq` as an array of Python dictionaries. See script 6.10 for a more detailed example of this syntax. Note that:

- Available dictionary keys are: ‘restfreq’, ‘bandwidth’, ‘nchan’, ‘res’, ‘deltafreq’, ‘tint’, ‘vpol’, ‘bank’, ‘beam’, and ‘subband’.
  - ‘restfreq’ takes a float value in MHz and is the only required key for each dictionary term.
  - ‘res’ is the spectral resolution (kHz) and can be used as an alternative to the ‘nchan’ restfreq dictionary key or the ‘nchan’ keyword in the main configuration block to select the VEGAS mode. Allowed values are floats and listed in Table 8.1.
  - ‘bank’ specifies which VEGAS bank to use. Allowed values are the string letters ‘A’→‘H’. The default is to let the configuration tool select which bank should be used (recommended).
  - All other keys have the same meaning as the standard configuration keywords.
- Key-value pairs specified in the dictionary override configuration keywords specified in the main configuration block which in turn override any default values.
- `dopplertrackfreq` must **always** be set in the main configuration block.
- `deltafreq` may still be specified as a single global offset in the main configuration block or omitted to use the default value of zero.
- `nchan` must **always** be set in the main configuration block, even if that value is overridden by ‘nchan’ in the restfreq dictionary.

### 6.2.5.2 Keywords With Default Values

**swmode** This keyword specifies the switching mode to be used for the observations. This keyword's values are given as a string. The switching schemes are

- **'tp'**: (Total Power With Cal) - The noise diode is periodically turned on and off for equal amounts of time. **(Default value)**
- **'tp\_nocal'**: (Total Power Without Cal) - The noise diode is turned off for the entire scan.
- **'sp'**: (Switched Power With Cal) - The noise diode is periodically turned on and off for equal amounts of time while another component is in a signal state and then again in a reference state. This is used in frequency switching where the signal state is one frequency and the reference state is another frequency. Similarly beam switching and polarization switching change the beams or polarizations so that their signals are sent down two different IF paths.
- **'sp\_nocal'**: (Switched Power Without Cal) - The noise diode is turned off while another component is switched between a signal and reference state.

**swtype** This keyword is only used when `swmode='sp'` or `swmode='sp_nocal'`, and specifies the type of switching to be performed. This keyword's values are 'none', 'fsw' (frequency switching), 'bsw' (beam switching) and 'psw' (polarization switching). **Default values are 'fsw' for all receivers except receiver='Rcvr26\_40'. The default for receiver='Rcvr26\_40' is swtype='bsw'.**

**swper** This keyword defines the period in seconds over which the full switching cycle occurs. See Table ?? for recommended minimum switching periods for each VEGAS mode. The value is a float. **Default values are 0.2 for obstype='continuum', 0.04 for obstype='pulsar', and 1.0 for any other value for the obstype keyword.**

**swfreq** This keyword defines the frequency offsets used in frequency switching (`swtype='fsw'`). The value consists of two comma separated floats which are the pair of frequencies in MHz. The best values for `swfreq` are  $\text{bandwidth}/2^n$  where  $n$  is an integer so that the frequency switch will be an integer number of channels giving less artifacts in data reduction. **Default values are `swfreq=-0.25*Bandwidth, +0.25*Bandwidth` for `swtype='fsw'`, and `swfreq=0,0` otherwise.**

**tint** This keyword specifies the backend's integration (dump) time. The value is a float with units of seconds. See Table 8.1 for minimum integration times with VEGAS. **Default values are 10.0 for obstype='continuum', tint=swper for obstype='spectroscopy' and 30.0 of any other value for the obstype keyword.**

**beam** This keyword specifies which beams are to be used for observations with multi-beam receivers. The keyword value is a string of comma separated integers. For example `beam='2'` would record data for the second beam and `beam='3,7'` would record data for beams 3 and 7. When using the KFPA, `beam='all'` can be used to record data from all seven beams. This 'beam' configuration keyword has a different meaning to the 'beamName' in observing scans which usually specifies a tracking beam, not which beams to record data for. **The default value is '1'.**

**Note:** The Ka-band dual-beam single-polarization 26–40 GHz receiver ('Rcvr26\_40') has a unique architecture among GBT receivers. The two beams are routed through the IF system as if they are two polarizations of a single beam. Therefore, the only allowable setting for 'beam' with 'Rcvr26\_40' is `beam='1'`.

**nwin** This keyword specifies the number of frequency windows that will be observed for backends other than VEGAS. The value for this keyword is an integer with a maximum value that is backend and receiver dependent, see § 2.1.4. The number of values given for the `restfreq` keyword must be the same as `nwin`. **The default value is 1.**

**Note:** 'nwin' does not need to be specified for VEGAS configurations.



**deltafreq** This keyword specifies offsets in MHz for each spectral window so that the restfreq is not centered in the middle of the spectral window. ‘deltafreq’ can be specified as a single float offset which will be applied across all windows or in the same manner as ‘restfreq’. For examples of using deltafreq with different types of restfreq syntax, see Scripts 6.8, 6.9, and 6.10. More details on deltafreq can be found in Appendix B. **The default value is 0.0.**

**vframe** This keyword specifies the velocity frame (the inertial reference frame). The keyword value is a string. Allowed values are ‘topo’ (topocentric, i.e. Earth’s surface), ‘bary’ (Barycenter of solar system), ‘lsrk’ (Local Standard of Rest kinematical definition, i.e. typical LSR definition), ‘lsrd’ (Local Standard of Rest dynamical definition – rarely used), ‘galac’ (center of galaxy), ‘cmb’ (relative to Cosmic Microwave Background). **The default value is ‘topo’.**

**vdef** This keyword specifies which mathematical equation (i.e. definition) is used to convert between frequency and velocity. The keyword value is a string. Allowed values are ‘Optical’, ‘Radio’, ‘Relativistic’. **The default value is ‘Radio’.**

$$v_{\text{radio}} = c \left[ 1 - \frac{\nu}{\nu_o} \right] \quad (6.1)$$

$$v_{\text{optical}} = c \left[ \frac{\nu_o}{\nu} - 1 \right] \quad (6.2)$$

$$v_{\text{relativistic}} = c \left[ \frac{\nu_o^2 - \nu^2}{\nu_o^2 + \nu^2} \right] \quad (6.3)$$

### 6.2.5.3 Backend and Receiver Dependent Keywords

Some configuration keywords depend on which backends and receivers are being used. Some observations may require one of these keywords while for other observations none may be needed.

**nchan** An integer. This keyword is used to determine the number of spectral channels that VEGAS will provide. Available values are listed in Table 8.1.

- The following string values designed for use with the now obsolete GBT spectrometer may still be used: ‘low’, ‘medium-low’, ‘medium’, ‘medium-high’, and ‘high’. These string values may be used to distinguish between up to 5-levels of resolution for a given bandwidth. For example, mode 18 of VEGAS could be set by setting bandwidth=11.72 and nchan=262144 or nchan=‘medium-high’.

**dopplertrackfreq** A float specifying the rest frequency in MHz used to compute the velocity for doppler tracking. When using the simple restfreq syntax, the default is the first listed restfreq value.

**pol** Each of the prime focus receivers, L–band, S–band and C–band receivers have a hybrid that can output either linear or circular polarization. This keyword specifies whether linear or circular polarization is desired for these receivers. The keyword value is a string. Allowed values are ‘Linear’ and ‘Circular’. **The default value is ‘Circular’ for the VLBI and Radar backends, and ‘Linear’ otherwise.**

**noisecal** All receivers below 12 GHz have two noise diodes for calibration signals – one with an equivalent brightness temperature at roughly one tenth the system temperature (‘lo’ value) and one nearly equal to the system temperature (‘hi’ value). This keyword is a string which specifies which noise diode is to be used. Allowed values<sup>1</sup> are ‘lo’, ‘hi’ and ‘off’. **The default value is ‘lo’ except for the Radar backend for which the default values is ‘off’.**

<sup>1</sup>There are expert values of ‘on-mcb’, ‘on-ext’, ‘lo-mcb’, ‘hi-mcb’, ‘lo-ext’ and ‘hi-ext’ whose use is beyond the scope of this document. Please contact a support person about the use of these values.

For the Ka-band (26-40 GHz) receiver there are three additional choices. These are ‘L’, ‘R’, or ‘LR’. The Ka-band receiver has two ‘lo’ noise diodes, one for each polarization for each of the two beams. The ‘L’, ‘R’, and ‘LR’ options specify which of these noise diodes are to be used.

**notchfilter** There is a notch filter covering roughly 1200–1310 MHz in the L-band receiver that filters out an FAA radar signal. This keyword determines if this notch filter is in place and used by the system or is removed from the receiver’s RF path. The keyword value is a string with allowed values of ‘In’ or ‘Out’. **The default value is ‘In’.**

**vegas.vpol** Keyword to specify which spectral product to record in the FITS file. It assumes the following values:

- ‘self’: Record the total intensity polarization products. **(Default value)**
- ‘cross’: Record the full Stokes polarization products.
- ‘self1’: Record the polarization from the first Analog to Digital Converter (ADC) card only. There are two ADCs per VEGAS bank, one for each polarization.
- ‘self2’: Record the polarization from the second ADC only.

**vegas.subband** Keyword used by config tool to select between 23.44 MHz VEGAS modes with single and multiple spectral windows (see Table 8.1). It assumes values 1 or 8. **The default value is 8.**

**guppi.obsmode** GUPPI-specific keyword (see Chapter 9). Controls both the dedispersion and observing mode. All data are written in format. Allowed values are

- ‘search’: Incoherent search-mode, i.e. spectra are rapidly written to disk.
- ‘fold’: Incoherent fold-mode, i.e. spectra are folded on-line using a pulsar ephemeris, with sub-integrations written to disk at a rate controlled by the **guppi.fold.dumptime** keyword. *Note: incoherent fold-mode is effectively deprecated in favor of coherent fold-mode.*
- ‘cal’: Incoherent cal-mode, i.e. data are folded at a constant 25 Hz frequency; used in conjunction with the pulsed noise diodes for each receiver.
- ‘coherent\_search’: Coherent search-mode, i.e. spectra are coherently dedispersed at a DM specified using the **guppi.dm** keyword before being channelized, accumulated, and written to disk.
- ‘coherent\_fold’: Coherent fold-mode, i.e. spectra are coherently dedispersed at the DM specified in the ephemeris file before being written to disk. Since the DM is read from the ephemeris file, the **guppi.dm** keyword is not needed.
- ‘coherent\_cal’: Coherent cal-mode, i.e. data are taken as in coherent fold mode, although no dedispersion is applied since the noise diode is not dispersed.

**guppi.polnmode** controls whether Full Stoke’s or total intensity data are recorded. Allowed values are ‘full\_stokes’ and ‘total\_intensity’, though total intensity can only be used in incoherent search-mode.

**guppi.numchan** sets the number of spectral channels. Allowed values are **any power-of-two between 64 and 4096**, though care must be taken not to exceed the maximum data rate.

**guppi.outbits** controls the number of bits used for output values. The only allowed value is **8**.

**guppi.scale** controls the internal scaling so that the output data is properly scaled for 8-bit values. This value is typically chosen from experience with the observing set-up. Contact your project friend for advice on which value to use.

**guppi.datadisk** controls which RAID data are written to in incoherent modes. Allowed values are ‘data1’ or ‘data2’. In coherent modes data are written to eight HPC machines and this keyword is not used. It will go in a subdirectory called **/guppi.datadisk/observname/projectID/date/**. Since the data will be owned by the “monctrl” computer account, you will not be able to remove it – that means Scott Ransom will bug you mercilessly until you process your data!

**guppi.fold\_parfile** specifies the path to the ephemeris (parfile) used for either incoherent or coherent fold-modes. *The parfile must exist and be visible from beef, and readable by tempo.*

**guppi.dm** controls the DM used for coherent search-mode. It is not used by any other modes.

**guppi.fold\_bins** controls the number of phase bins used for either incoherent or coherent fold- or cal-modes. Enough bins should be used to fully resolve fine profile structure. Typical values are 256 in incoherent modes (which are not typically used anymore) and 2048 in coherent modes.

**guppi.fold\_dumptime** controls the length of a sub-integration in either incoherent or coherent fold- or cal-modes. The value is specified in seconds, with 10 s being typical. It must be shorter than the total scan length.

**vlbi.phasecal** This expert keyword turns the VLBI phase calcs on or off. The phase calcs can run at 1 MHz ('M1') or 5 MHz ('M5'). The keyword value is a string. Allowed values are 'off', 'M1' or 'M5'.

**broadband** This keyword is used to activate the “broadband” 7.5 GHz maximum instantaneous mode of the KFPA by setting `broadband=1`. This may only be used with single beam configurations using either beam 1 or beam 2. The default is “off” (`broadband=0`).

#### 6.2.5.4 Expert Keywords

These keywords should only be used by very experienced observers who have expert knowledge of how a given backend works or in how the GBT IF system works.

**vlow** and **vhigh** These keywords specify the minimum and maximum velocity to be observed from a group of sources. The value is a float and is in  $\text{km s}^{-1}$  for velocities. See Appendix C for more details on the use of `vlow` and `vhigh`. The use of `vlow` and `vhigh` is not recommended for frequencies where there can be large amounts of RFI. **The default value is 0.0.**

**iftarget** This keyword specifies the target voltage level to use when balancing the IF Rack. The keyword value is a float. The nominal range of the IF Rack is 0.0–10.0 and the linear range is 0.1–5.0.

**xfer** This expert keyword sets the beam switch for the Ku-band, K-band and Ka-band receivers. The keyword is a string. Allowed values are 'ext', 'thru', or 'cross'. The default values are 'ext' when `swtype='bsw'` and 'thru' otherwise.

**polswitch** This expert keyword sets the polarization switch for the L-band and X-band receivers. The keyword value is a string. Allowed values are 'ext', 'thru', and 'cross'. The default value is 'ext' if `swtype='psw'` and 'thru' otherwise.

**ifbw** This expert keyword sets the minimum IF bandwidth to be used in filters within the receiver and in the IF Rack. The keyword value is float with units of MHz.

**if0freq** This expert keyword is used to set the center frequency of the IF after the mixing the RF signal with the first LO. The keyword value is a float with units of MHz.

**lo1bfreq** This expert keyword is used to set the frequency of the synthesizer used for the alternative The First LO (LO1), LO1B. This keyword is only to be used with the Ka-band receiver. The keyword value is a float with units of MHz.

**lo2freq** This expert keyword is used to set the frequency values of the eight LO2 synthesizers within the Converter Rack. The keyword values are a comma separated list of floats with units of MHz.

**if3freq** This expert keyword is used to set the IF input frequency of the backend. The keyword value is a comma separated list of floats with units of MHz.

### 6.2.6 Resetting The Configuration

The configuration tool in Astrid remembers all the keyword values defined during a session. This feature occasionally results in Astrid being unable to validate an otherwise correct configuration because of previously set values or hardware being configured improperly. To reset the configuration parameters to their default state, you can issue the **ResetConfig()** command in a script before another **Configure()**. This command will reset the configuration tool values to their defaults.

## 6.3 Catalogs

The Source Catalog system in Astrid provides a convenient way for the user to specify a list of sources to be observed, as well as a way to refer to standard catalogs of objects. At a minimum for each source there must be a name and a location (Ra/Dec or Glat/Glon, etc). Other parameters may be set, such as radial velocity. An example of a simple Catalog is:

```
# My source list
format=spherical
coordmode=J2000
HEAD = NAME      RA      DEC
Object1  09:56:16.98  +49:16:25.5
Object2  10:56:16.98  +50:16:25.5
Object3  11:56:16.98  +51:16:25.5
Object4  12:56:16.98  +52:16:25.5
```

Script 6.11: A simple catalog.

There are three formats of catalogs:

- **SPHERICAL** A fixed position in one of our standard coordinate systems, e.g., RA/DEC, AZ/EL, GLON/GLAT, etc.
- **EPHEMERIS** A table of positions for moving sources (comets, asteroids, satellites, etc.)
- **NNTLE NASA/NORAD** Two-Line Element (TLE) sets for earth satellites.

In addition, the following solar system bodies may be referred to by name: “Sun”, “Moon”, “Mercury”, “Venus”, “Mars”, “Jupiter”, “Saturn”, “Uranus”, “Neptune”, and “Pluto”. These names are case-insensitive and may be given to any Scan Type function (see § 6.4). No catalog needs to be invoked for the system to understand these names.

To use the catalog system, observers invoke the **Catalog()** command within SBs and pass names of the desired objects to any of the scan functions (§ 6.4). All sources named in all the catalogs that have been invoked are available within an SB. If the same name appears in two or more catalogs, the name from the most recently invoked catalog will prevail. Name comparisons are case-insensitive, hence “b2322+16” and “B2322+16” are equivalent.

### 6.3.1 Getting Your Catalog Into Astrid

Although one can include any number of Catalogs in an SB, the standard practice is to put all the Catalogs into separate files that are then brought into the SB via multiple calls to the **Catalog()** command. This: a) keeps SBs simple and without clutter; and b) allows changes to be made to a Catalog without having to validate and re-save the SB.

The best way to learn about how to bring Catalogs into an SB is through an example. Let’s suppose that there are two Catalogs that you need for your observations. These two catalogs are in the following files:

```
/home/astro-util/projects/GBTog_examples/cats/sources.cat
/home/astro-util/projects/GBTog_examples/cats/pointing.cat
```

These catalogs may be loaded into the SB as shown in the following example:

```
#first load the catalog with the flux calibrators
cata=Catalog('/home/astro-util/projects/GBTog_examples/cats/sources.cat')

#now load the catalog with the pointing source list
catb=Catalog('/home/astro-util/projects/GBTog_examples/cats/pointing.cat')

#Objects defined in loaded catalogs may now be used in scan functions
#Object1 is in source.cat and 0006-0004 is in pointing.cat
Track('Object1',None, 60)
Slew('0006-0004')
```

Script 6.12: Loading two catalogs into an SB, then using objects defined in those catalogs as inputs to scan functions

All sources from all catalogs are available and referenced by name within the scope of the SB, with the exception that for duplicate source names only the last entry of that name will be recognized. After loading a Catalog any scan function may be run by giving it the source name as shown above (Script 6.12).

## 6.3.2 The Format of the Catalog

A Catalog typically has two sections: a header section followed by a table of information for all the sources. The header section consists of “KEYWORD = VALUE” pairs. The “KEYWORD = VALUE” pairs tell the Scheduling Block interpreter how to read the information in the table section of the Catalog. Once a keyword value is given, its value will persist until re-set or the end of the Catalog is reached. The keywords are case-insensitive. The values for a keyword must not contain any embedded blanks (except source names in NNTLE format).

A Catalog can contain comments with the beginning of a comment being denoted by the hash symbol, “#”. All information on a line after the hash symbol is considered to be part of the comment. After the header, each source in the Catalog occupies a single line. You should not use the hash symbol in source names.

## Catalog Header Keywords

Catalog Header Keywords are used to define how the catalog entries should be read. The keywords and their values are case insensitive. The example shown in Script 6.13 will be used to describe some of the Catalog Header Keywords. Unless mentioned otherwise, the following keywords should be listed as column headings under “HEAD”:

```
# My source list
format=spherical
HEAD = NAME COORDMODE RA DEC RESTFREQ VELDEF VEL type
Object1 J2000 09:56:16.9 +49:16:25 1420.405 VRAD-LSR -25.3 HII
Src_A J2000 10:56:16.9 +50:16:25 1665.401 VOPT-BAR 100.9 Gal
```

Script 6.13: An example catalog using additional header keywords

**FORMAT** This tells the type of catalog and **must be the first line in any catalog**. Possible values are “spherical”, “ephemeris” and “nmtle”. For the SPHERICAL format, the first line would contain “FORMAT=SPHERICAL”. This is the default format, hence the “FORMAT=SPHERICAL” may be omitted.

**HEAD** This gives the header for tabular data, and consists of a list of any keywords. This should appear as the last line in the header before lines giving information about the sources in the catalog. You can also create your own header keyword, such as the “type” column in the above example. The default header is “HEAD = NAME RA DEC VELOCITY”. In the above example we have added more entries than the default. We have also created a new keyword named “type”.

**NAME** The source name is any string up to 32 characters long. The name should not contain any embedded blanks or hashes.

**COORDMODE** The default is J2000. Possible values are: J2000, B1950, JMEAN (mean coordinate of date given by EQUINOX), GAPPT (geocentric apparent coordinates of date), GALACTIC, HADEC, AZEL, ENCODER. In the above example we put the COORDMODE keyword in the HEAD line since we have sources whose positions are given in different coordinate modes (J2000 and B1950). This keyword may be given as either a header keyword or column heading under “HEAD”.

**VEL** or **VELOCITY** The radial velocity in km/sec. The Default is to use any previous setting or 0.0 if there is none.

**VELDEF** Velocity definition in the FITS convention, e.g. “VOPT-BAR”, “VRAD-LSR”, etc. (see <https://safe.nrao.edu/wiki/bin/view/GB/Data/VelDefFits>). The default is the velocity definition or reference frame that was previously set. In the above example we put the VELDEF keyword in the HEAD line since we have sources whose velocity definitions are different. This keyword may be given as either a header keyword or column heading under “HEAD”. This value will also override the velocity definition in the configuration (see § 6.2.5).

**RESTFREQ** The rest frequency, in MHz. The default is to use the previous setting. Again we put the RESTFREQ keyword in the HEAD line since we are defining two different spectral line rest frequencies for each source. Note that this is an “expert” keyword as one has to be aware of any conflicts with the hardware configuration. This keyword may be given as either a header keyword or column heading under “HEAD”.

**RA, HA, DEC, AZ, EL, GLON, GLAT** A pair of coordinates must be given: RA/DEC, HA/DEC, AZ/EL, or GLON/GLAT. Angle formats may be either in sexagesimal with colons (e.g. dd:mm:ss.ss) or in decimal format. RA and HA are always in hours regardless of decimal or sexagesimal notation, while all other coordinates use degrees or arc in both formats.

**EQUINOX** Used if the Coordmode is “JMEAN”. The value is a float (e.g. 2006 December 1, 12:00 UT would be 2006.919178082192). This keyword may be given as either a header keyword or column heading under “HEAD”.

```
# My source list
format=spherical
coordmode=jmean
equinox=2007.123456
HEAD = NAME      RA      DEC
Object2  10:56:16.98  +50:16:25.5
```

Script 6.14: An example catalog using the equinox header keyword.

- Additional keywords used when the Ephemeris format is active are (see § 6.3.5 for examples):

**DATE** The UTC date, either “2005-06-23” or “2005-Jun-23” form. This keyword may be given as either a header keyword or column heading under “HEAD”.

**UTC** The UTC time in the form “hh:mm:ss”.

**DRA, DHA, DDEC, DAZ, DEL, DLON, DLAT** The coordinate rate keywords given in arc-seconds per hour.

**DVEL** The radial velocity rate in km/sec/hour.

- Additional keywords used by the NNTLE format are (see § 6.3.6 on NNTLE format below for examples):

**FILE** For use in NNTLE format only. This keyword value may refer to a file or a URL containing a 2-line element set. This keyword may not be listed as a column name under “HEAD”.

**USERADVEL** For use in the NNTLE format only. If this is set to 1, then the radial velocity tracking will be performed. Otherwise, if this is set to 0 or is missing then radial velocity tracking will not be performed. This keyword may not be listed as a column name under “HEAD”.

### 6.3.2.1 SPHERICAL format Examples

–Here is an example of a simple catalog.

```
# My source list
format=spherical
coordmode=J2000
HEAD = NAME      RA          DEC
Object1  09:56:16.98  +49:16:25.5
Object2  10:56:16.98  +50:16:25.5
Object3  11:56:16.98  +51:16:25.5
Object4  12:56:16.98  +52:16:25.5
```

Script 6.15: A simple catalog.

–Because all the keyword values use the defaults, the following is equivalent:

```
# My source list
Object1  09:56:16.98  +49:16:25.5
Object2  10:56:16.98  +50:16:25.5
Object3  11:56:16.98  +51:16:25.5
Object4  12:56:16.98  +52:16:25.5
```

Script 6.16: A simple catalog using default header keywords and columns.



–Here is an example catalog that specifies the radial velocities of the sources.

```
# My source list with radial velocities
format=spherical
coordmode = B1950
head = name      ra      dec      velocity
Object1 09:56:16.98 +49:16:25.5 27.23
Object2 08:56:16.98 +48:16:25.5 28.24
Object3 07:56:16.98 +47:16:25.5 29.25
Object4 06:56:16.98 +45:16:25.5 30.26
```

Script 6.17: A catalog using the “velocity” column to specify radial velocity in km/sec.

–Here is an example Catalog where one may omit the “format=” line, but not the “coordmode=” line.

```
# A list of HII regions
coordmode=Galactic
head= NAME GLON      GLAT      vel      restfreq
G350+.07  350.107  +0.079  42.2235  9816.867
G351+.17  351.613   0.172 -15.553  9487.824
G352-.17  352.393  -0.176 -52.227  9173.323
G352-.36  353.4219 -0.3690 22.335  9487.824
```

Script 6.18: A catalog using the “Galactic” coordinate mode and omitting “format=”.

**Warning:** setting the velocity or rest frequency in a catalog only changes the values in the LO1 manager. If either value is changed by a large amount, the receiver selection or bandpass filters or the frequency spacing between spectral windows may change. Thus one should re-configure the IF system for a large change in velocity or frequency. The user should be wary of how much the velocity or rest frequency can change for a particular configuration.

–Finally we show an example Catalog with user-defined keywords. The user may create custom keywords (or equivalently column headings). These are available within an SB, but are otherwise ignored.

```
# a list of pointing references
format=spherical
coordmode=j2000
head= name      ra      dec      BMIN BMAX  S20  S6
0011-1434 00:11:40.40 -14:34:04.7 15 45 0.17 0.20
0012-3321 00:12:17.96 -33:21:57.8 15 180 0.85 0.18
0012+6551 00:12:37.80 +65:51:10.5 15 360 1.20 0.55
0012+2702 00:12:38.14 +27:02:40.7 15 180 0.60 0.21
0012+3353 00:12:47.3826 +33:53:38.459 0 45 0.08 0.08
0012-3954 00:12:59.9080 -39:54:25.836 0 45 0.49 1.5
```

Script 6.19: A catalog with user-defined keywords

### 6.3.3 Standard Catalogs

Several “standard” catalogs listed in Table 6.4 are available for use within the Green Bank computing system. They are all ASCII files in the directory `/home/astro-util/astridcats`.

Note that for convenience, these standard catalogs may be referred to within Astrid simply by name, without the “.cat” extension. e.g.:

```
c = Catalog(kaband_pointing)
```

Script 6.20: Loading standard catalogs into an SB

Table 6.4: The following Catalogs are present as of April 2016. The flux densities of pointing calibrators vary by up to a factor of two on time scales of years at frequencies higher than 8 GHz, so the pointing calibrators will never be good flux calibrators. The main reason for updating their flux densities is to make sure the observer gets a strong-enough pointing calibrator. For genuine flux-density calibration, we recommend observers use the flux densities of 3C123, 3C286, and 3C295 (See Perly and Butler, 2013).

Catalog	Description
fluxcal	Calibrators with well-determined flux densities. U. S. Government Printing Office (Usgpo) 2006, The Astronomical Almanac for the year 2006, Washington: U.S. Government Printing Office (USGPO), 2006, U.S. Naval Observatory (USNO), Royal Greenwich Observatory (RGO).
pointing	Condon’s master pointing catalog for the GBT. <a href="https://safe.nrao.edu/wiki/bin/view/GB/PTCS/PointingFocusCatalog">https://safe.nrao.edu/wiki/bin/view/GB/PTCS/PointingFocusCatalog</a>
pf_pointing	Extracted from pointing catalog for the 50 cm band (0.6GHz).
lband_pointing	Extracted from pointing catalog for the 21 cm band (1.4GHz).
sband_pointing	Extracted from pointing catalog for the 10 cm band (3GHz).
cband_pointing	Extracted from pointing catalog for the 6 cm band (6GHz).
xband_pointing	Extracted from pointing catalog for the 3.5 cm band (9GHz).
kuband_pointing	Extracted from pointing catalog for the 2 cm band (14GHz).
kband_pointing	Extracted from pointing catalog for the 1.5 cm band (20GHz).
kaband_pointing	Extracted from pointing catalog for the 9 mm band (32GHz).
qband_pointing	Extracted from pointing catalog for the 7 mm band (43GHz).
wband_pointing	Extracted from pointing catalog for the 3.5mm band (86GHz).
mustang_pointing	Extracted from pointing catalog for the 3.3mm band (90GHz).
HI_strong	Galaxies with strong HI lines, extract from Rich Fisher’s database. <a href="http://www.gb.nrao.edu/~rfisher/GalaxySurvey/galaxy_survey.html">http://www.gb.nrao.edu/~rfisher/GalaxySurvey/galaxy_survey.html</a>
pulsars.all	All 1533 pulsars in the ATNF database as of 26 Aug 2005.
pulsars.all_GBT	All 1054 pulsars visible from Green Bank.
pulsars.brightest_GBT	The brightest pulsars, visible from Green Bank.
pulsars.bright_MSPs_GBT	Bright millisecond pulsars visible from Green Bank.

The GBT pointing catalog has been updated several times to include better positions and more recent flux densities. These changes are described in the PTCS project notes posted at <https://safe.nrao.edu/wiki/bin/view/GB/PTCS/ProjectNotes>

- PTCS/PN/58 introduces PCALS4.1 and “gold standard” pointing calibrators for use at higher frequencies
- PTCS/PN/66 introduces PCALS4.4, a catalog upgrade incorporating high-frequency flux densities from WMAP5 and accurate positions from the VLBA calibrator surveys through VCS6
- PTCS/PN/72 introduces PCALS4.5 with high-frequency flux densities updated by WMAP7, the Planck “Early Release Compact Source Catalog”, and the Australia Telescope AT20G survey.
- PCALS 4.7 adds new 30, 44, 70, and 100 GHz flux densities from the final Planck Catalogue of Compact Sources (Planck Collaboration, 2013 ( arXiv:1303.5088) and 20 GHz fluxes from Righini, S., Carretti, E., Ricci, R. et al. 2012 MNRAS, 426, 2107 “A 20 GHz bright sample for  $\delta > +72^\circ$ : I. Catalogue”. There is no PTCS/PN describing this release.

### 6.3.4 Catalog Functions

Two useful catalog functions are available.

#### c.keys()

Acts like a Python function that returns a list of all the source names in the Catalog loaded into the variable ‘c’ [i.e. via `c=Catalog('mycatalog')`]. The value returned can be used in the SB to automatically loop through all the sources in a catalog. Here is an example of how to do this:

```
c = Catalog(HI_strong)
sourcenames = c.keys()
for s in sourcenames :
    Nod(s, '1', '2', 120)
```

Script 6.21: Using c.keys() to create an array of source names in the SB

#### c['sourcename']['keyword']

Returns the value of the keyword for the named source in the Catalog loaded into the variable ‘c’. This function can be used to pass information in the Catalog on to the SB (e.g. specifying different map sizes for different sources/directions).

The `c['sourcename']['keyword']` function can be used to get information out of the “keyword” column of the Catalog for use within the SB. In the following example we get the source’s Declinations and only observe those sources above  $20^\circ$  Declination (note that the coordinates are always returned in degrees):

```
c = Catalog(lband_pointing)
sourcenames = c.keys()
for s in sourcenames :
    print c[s]['dec']
    if c[s]['dec'] > 20 :
        Nod(s, '1', '2', 120)
```

Script 6.22: An example SB that retrieves the declination of each source in the catalog and prints it to screen. If the declination of the source is above  $20^\circ$  then the SB will proceed to perform a `Nod()` scan.

The `c[“sourcename”][“keyword”]` function can also be used to execute more complicated observing strategies. In the following example we have many sources to observe and we desire a different amount of total integration time for each source. To accomplish this we add two new columns to the Catalog. We will call these columns “sourcetime” and “status”. A few lines of the Catalog (lets call it `mycatalog.cat`) would look like:

```
head= name ra dec velocity sourcetime status
SrcA  00:01:02 -03:04:05 -22.0 300 done
SrcB  06:07:08 +10:11:12 +56.3 120 waiting
```

Script 6.23: An example catalog with user-defined columns.

The SB would look like:

```
c = Catalog('mycatalog.cat')
sourcenames = c.keys()
for s in sourcenames :
    if c[s]['status'] == 'waiting' :
        dwelltime = float(c[s]['sourcetime'])
        Track(s, None, dwelltime)
```

Script 6.24: An example SB that retrieves information for each source from user-defined columns in the catalog shown in Script 6.23.

**Note that `c[“sourcename”][“keyword”]` will return a string value.** Thus, we convert “dwell-time” in Script 6.24 to a float value in order to use it as a suitable time argument in the **Track** scan function.

### 6.3.5 EPHEMERIS : Tables for moving objects

A Catalog can also be used as an Ephemeris for the position of a moving object, such as a comet or asteroid. To make the Catalog into an Ephemeris the first non-comment line of the Catalog must contain:

```
FORMAT = EPHEMERIS
```

The header of the Catalog for an Ephemeris can also contain the `NAME`, `COORDMODE`, `VELDEF` and `HEAD` keywords. The “data lines” in the Catalog must contain at least the date, the time, and a pair of coordinates for an Ephemeris. Optional parameters are coordinate rates, radial velocity and radial velocity rate. User-defined parameters may also be added.

The dates and times are required to be in UTC. The dates and times can be specified in any legal python form, for example: a) ‘YYYY-MM-DD hh:mm:ss’ where MM is month number (e.g August = 09); or b) ‘YYYY-MMM-DD hh:mm:ss’ where MMM is the abbreviated month name such as Jan, Feb, etc.

The ephemeris table should contain enough entries to cover a period longer than that required by a particular observing session with sufficient time resolution for the expected motion with respect to the telescope’s beam size. The observing system selects the portion of the table needed for the current scan start time and duration.

### 6.3.5.1 Example Ephemeris Catalogs

```

FORMAT      = EPHEMERIS
NAME        = MyMovingObject
COORDMODE   = J2000
VELDEF      = VRAD-LSR
#-----
2004-07-16 00:10:00 09:56:16.98 +49:16:25.5 27.234234
2004-07-16 00:20:00 09:56:17.76 +49:16:36.2 27.456345
2004-07-16 00:30:00 09:56:18.55 +49:16:46.9 27.568233
2004-07-16 00:40:00 09:56:19.32 +49:16:57.6 27.623423
2004-07-16 00:50:00 09:56:20.10 +49:17:08.3 27.723456
#-----

```

Script 6.25: A simple ephemeris catalog. Note that the “HEAD=” line has been omitted because the default is “DATE UTC RA DEC VEL”

```

FORMAT = EPHEMERIS
VELDEF = VRAD-TOP
COORDMODE = J2000
HEAD = date utc ra dec dra ddec vel
# 1: soln ref.= JPL#153
NAME = 2008CM
2015-Dec-30 05:00 09:00:46.65 +13:13:58.0 -1045.1405 -1344.9600 4.8196
2015-Dec-30 05:10 09:00:35.06 +13:10:13.7 -1044.8328 -1344.5100 4.8595
2015-Dec-30 05:20 09:00:23.47 +13:06:29.5 -1044.4856 -1344.0600 4.8997
2015-Dec-30 05:30 09:00:11.88 +13:02:45.3 -1044.0885 -1343.6000 4.9405
2015-Dec-30 05:40 09:00:00.30 +12:59:01.2 -1043.6313 -1343.1400 4.9815
2015-Dec-30 05:50 08:59:48.72 +12:55:17.2 -1043.1244 -1342.6900 5.0230
2015-Dec-30 06:00 08:59:37.15 +12:51:33.3 -1042.5679 -1342.2200 5.0648

```

Script 6.26: A more complicated ephemeris catalog for a comet that specifies the coordinate rates.

```

# PRN14 tracking table (angles in degrees)
# visible 01:30 to 3:00 UT

format      = ephemeris
name        = PRN14
coordmode   = azel
head=date   utc          az          el
#-----
2004-05-16 01:30:06 103.1822  43.0174
2004-05-16 01:30:14 103.2464  42.9721
2004-05-16 01:30:22 103.3105  42.9268
2004-05-16 01:30:30 103.3745  42.8814
#-----

```

Script 6.27: A ephemeris catalog for tracking a satellite.

### 6.3.5.2 Comets

Tracking a comet which does not track at the sidereal rate will require the use of an external file generated from the NASA JPL Horizons website which holds a database of all the orbital parameters of all major and minor bodies in the solar system. First you must download the ephemeris file for your object of interest from the website: <http://ssd.jpl.nasa.gov/horizons.cgi>. Then you will have to convert the file into the CATALOG format for astrid.

When you go to <http://ssd.jpl.nasa.gov/horizons.cgi> you should see something like what is shown in Figure 6.1.

The screenshot shows the JPL Horizons website interface. At the top, there is a NASA logo and the text "Jet Propulsion Laboratory California Institute of Technology". To the right, there are links for "+ View the NASA Portal" and "+ Near-Earth Object (NEO) Project", along with a "Search JPL" search bar. Below this is a navigation menu with tabs for "JPL HOME", "EARTH", "SOLAR SYSTEM", "STARS & GALAXIES", and "TECHNOLOGY". The main content area features a banner for "Solar System Dynamics" with a background image of the solar system. Below the banner is another navigation menu with tabs for "BODIES", "ORBITS", "EPHEMERIDES", "TOOLS", "PHYSICAL DATA", "DISCOVERY", "FAQ", and "SITE MAP".

**HORIZONS Web-Interface**

This tool provides a web-based *limited* interface to JPL's HORIZONS system which can be used to generate ephemerides for solar-system bodies. Full access to HORIZONS features is available via the primary telnet interface. HORIZONS system news shows recent changes and improvements. A web-interface tutorial is available to assist new users.

**Current Settings**

Ephemeris Type [change] : **OBSERVER**  
 Target Body [change] : **Mars** [499]  
 Observer Location [change] : **Geocentric** [500]  
 Time Span [change] : Start=**2012-05-09**, Stop=**2012-06-08**, Step=**1 d**  
 Table Settings [change] : *defaults*  
 Display/Output [change] : *default* (formatted HTML)

**Special Options:**

- set default ephemeris settings (preserves only the selected target body and ephemeris type)
- reset *all* settings to their defaults (caution: all previously stored/selected settings will be lost)
- show "batch-file" data (for use by the E-mail interface)

At the bottom of the page, there is a footer with navigation links for "ABOUT SSD", "CREDITS/AWARDS", "PRIVACY/COPYRIGHT", "GLOSSARY", and "LINKS". It also includes the "FIRST GOV" logo, the text "2012-May-09 18:14 UT (server date/time)", the NASA logo, and contact information: "Site Manager: Donald K. Yeomans" and "Webmaster: Alan B. Chamberlin".

Figure 6.1: The JPL Horizons website

Your entries in the “Current Settings” should be:

**Ephemeris Type:** OBSERVER

**Target Body:** SELECT YOUR OBJECT

Clicking on the blue [change] link will open a form to search for the object of interest.

**Observer Location:** Green Bank (GBT) [-9] (radar) ( 280 09' 36.7"E, 38 25' 59.1"N, 873.10 m)

To set the location to Green Bank, first click [change], then select “Observatories”, click “Display List”, and select “Green Bank (GBT) [-9] (radar)”.

**Time Span:** CHOOSE YOUR RANGE

The ephemeris table should contain enough entries to cover a period longer than that required by a particular observing session. The observing system selects the portion of the table needed for the current scan start time and duration. If the position of the comet is changing rapidly, you should select a “step” range of 5 mins or shorter. If the comet is further out in the solar system and is not moving as rapidly with respect to the sidereal rate, a “step” range of 10-15 mins may be adequate to track the comet. Consult your observatory friend if you are unsure of the step range you should choose.

**Table Setting:** QUANTITIES=1,3,20

Figure 6.2 shows the quantities that should be selected through the web interface to properly generate an ephemeris for tracking a comet. NOTE: The dates and times are required to be in UTC. The dates and times can be specified in any legal python form, for example: a) 'YYYY-MM-DD hh:mm:ss' where MM is month number (e.g August = 09); or b) 'YYYY-MMM-DD hh:mm:ss' where MMM is the abbreviated month name such as Jan, Feb, etc. (see below)

**Table Settings**

Select observer quantities from table below:  
[ switch to manual-entry list-of-numbers form ]

Use Settings Below Cancel

Optionally preset observer quantities selection using one of the following:  
planets satellites small-bodies default all none

1. <input checked="" type="checkbox"/> Astrometric RA & DEC	15. <input type="checkbox"/> Sun sub-long & sub-lat	29. <input type="checkbox"/> Constellation ID
* 2. <input type="checkbox"/> Apparent RA & DEC	16. <input type="checkbox"/> Sub Sun Pos. Ang & Dis	30. <input type="checkbox"/> Delta-T (CT - UT)
3. <input checked="" type="checkbox"/> Rates; RA & DEC	17. <input type="checkbox"/> N. Pole Pos. Ang & Dis	* 31. <input type="checkbox"/> Obs eclip. lon & lat
* 4. <input type="checkbox"/> Apparent AZ & EL	18. <input type="checkbox"/> Helio eclip. lon & lat	32. <input type="checkbox"/> North pole RA & DEC
5. <input type="checkbox"/> Rates; AZ & EL	19. <input type="checkbox"/> Helio range & rng rate	33. <input type="checkbox"/> Galactic latitude
6. <input type="checkbox"/> Sat. X & Y, pos. ang	20. <input checked="" type="checkbox"/> Obsrv range & rng rate	34. <input type="checkbox"/> Local app. SOLAR time
7. <input type="checkbox"/> Local app. sid. time	21. <input type="checkbox"/> One-Way Light-Time	35. <input type="checkbox"/> Earth->Site lt-time
8. <input type="checkbox"/> Airmass	22. <input type="checkbox"/> Speed wrt Sun & obsrvr	> 36. <input type="checkbox"/> RA & DEC uncertainty
9. <input type="checkbox"/> Vis mag. & Surf Brt	23. <input type="checkbox"/> Sun-Obsrvr-Target angl	> 37. <input type="checkbox"/> POS error ellipse
10. <input type="checkbox"/> Illuminated fraction	24. <input type="checkbox"/> Sun-Target-Obsrvr angl	> 38. <input type="checkbox"/> POS uncertainty (RSS)
11. <input type="checkbox"/> Defect of illumin.	25. <input type="checkbox"/> Targ-Obsrv-Moon/Illum%	> 39. <input type="checkbox"/> Range & Rng-rate sig.
12. <input type="checkbox"/> Sat. angle separ/vis	26. <input type="checkbox"/> Obsr-Primary-Targ angl	> 40. <input type="checkbox"/> Doppler/delay sigmas
13. <input type="checkbox"/> Target angular diam.	27. <input type="checkbox"/> Pos. Ang;radius & -vel	
14. <input type="checkbox"/> Obs sub-lng & sub-lat	28. <input type="checkbox"/> Orbit plane angle	

Notes:  
\* affected by apparent position estimation (atmospheric refraction model, see below)  
> requires object orbit covariance

Figure 6.2: Selecting quantities to generate an ephemeris.

**Display/Output:** download/save

After clicking “Generate Ephemeris”, you should save the file to a directory in your area in Green Bank. The ephemeris file will begin with a large amount of header information followed by lines containing the date, time and pairs of coordinates as shown in Script 6.28. Optional parameters are coordinate rates, geocentric distance and geocentric radial velocity.

```

*****
JPL/HORIZONS                      103P/Hartley 2                      2016-Apr-05 07:53:21
Rec #:900870 (+COV)   Soln.date: 2014-Nov-10_13:49:39   # obs: 6506 (1986-2013)

FK5/J2000.0 helio. ecliptic osc. elements (au, days, deg., period=Julian yrs):

EPOCH= 2456981.5 ! 2014-Nov-20.0000000 (TDB)   RMSW= n.a.
EC= .6937804720128784   QR= 1.064195154203179   TP= 2457863.8230300969
OM= 219.7487450736021   W= 181.3222857728302   IN= 13.6042724340803
A= 3.475268743304754   MA= 225.7701522441   ADIST= 5.886342332406328
PER= 6.4787437937458   N= .152132314   ANGMOM= .023095142
DAN= 5.88279   DDN= 1.06431   L= 41.0339546
B= -.310996   MOID= .0720049   TP= 2017-Apr-20.3230300969

Comet physical (GM= km^3/s^2; RAD= km):
GM= n.a.   RAD= .800
M1= 14.7   M2= n.a.   k1= 8.   k2= n.a.   PHCOF= n.a.

COMET comments
1: soln ref.= JPL#252, data arc: 1986-03-15 to 2013-05-14
2: k1=8.;
*****

Ephemeris / WWW_USER Tue Apr 5 07:53:22 2016 Pasadena, USA / Horizons
*****
Target body name: 103P/Hartley 2   {source: JPL#252}
Center body name: Earth (399)   {source: DE431}
Center-site name: Green Bank (GBT)
*****
Start time : A.D. 2016-Apr-05 00:00:00.0000 UT
Stop time : A.D. 2016-Apr-06 00:00:00.0000 UT
Step-size : 5 minutes
*****
Target pole/equ : No model available
Target radii : 0.8 km
Center geodetic : 280.160200,38.4330940,0.8760930 {E-lon(deg),Lat(deg),Alt(km)}
Center cylindric: 280.160200,5003.37558,3943.7589 {E-lon(deg),Dxy(km),Dz(km)}
Center pole/equ : High-precision EOP model {East-longitude +}
Center radii : 6378.1 x 6378.1 x 6356.8 km {Equator, meridian, pole}
Target primary : Sun
Vis. interferer: MOON (R_eq= 1737.400) km {source: DE431}
Rel. light bend : Sun, EARTH {source: DE431}
Rel. lght bnd GM: 1.3271E+11, 3.9860E+05 km^3/s^2
Small-body perts: Yes {source: SB431-BIG16}
Atmos refraction: NO (AIRLESS)
RA format : HMS
Time format : CAL
RTS-only print : NO
EOP file : eop.160404.p160626
EOP coverage : DATA-BASED 1962-JAN-20 TO 2016-APR-04. PREDICTS-> 2016-JUN-25
Units conversion: 1 au= 149597870.700 km, c= 299792.458 km/s, 1 day= 86400.0 s
Table cut-offs 1: Elevation (-90.0deg=NO),Airmass (>38.000=NO), Daylight (NO)
Table cut-offs 2: Solar Elongation ( 0.0,180.0=NO ),Local Hour Angle( 0.0=NO )
*****
Initial FK5/J2000.0 heliocentric ecliptic osculating elements (au, days, deg.):
EPOCH= 2456981.5 ! 2014-Nov-20.0000000 (TDB)   RMSW= n.a.
EC= .6937804720128784   QR= 1.064195154203179   TP= 2457863.8230300969
OM= 219.7487450736021   W= 181.3222857728302   IN= 13.6042724340803
Comet physical (GM= km^3/s^2; RAD= km):
GM= n.a.   RAD= .800
M1= 14.7   M2= n.a.   k1= 8.   k2= n.a.   PHCOF= n.a.
*****
Date__(UT)__HR:MN   R.A.__(ICRF/J2000.0)_DEC dRA*cosD d(DEC)/dt   delta   deldot
*****
$$$$$
2016-Apr-05 00:00 C   18 27 00.30 -13 25 32.9 12.40954   9.786338 3.50004632213965 -36.1740593
2016-Apr-05 00:05 C   18 27 00.37 -13 25 32.0 12.40419   9.784071 3.49997377062988 -36.1812496
.....
.....
.....

```

Script 6.28: A JPL ephemeris file



Now that you have your ephemeris, it needs to be converted to a form that Astrid can read. You can do this by running the Python script `jpl2astrid` from any directory in your area on the Green Bank computer system. If you just type “`jpl2astrid`”, and give it no arguments, it lists instructions, like this:

```
Usage: jpl2astrid cometfilename.txt [vel]
  If 'vel' is blank, do not write the radial velocities.
  If 'vel' is non-blank, do write the radial velocities.
  output will have '.astrid' extension.
  Include in Astrid with, e.g. Catalog(fullpath/cometfile.astrid )

  If '-h' or '-help' instead of 1st argument, print help message

Access the JPL Horizons web interface: http://ssd.jpl.nasa.gov/horizons.cgi
Set up Horizons web-interface as follows
ephemeris type:  OBSERVER
target body:    [select the object]
Observer Location:  Green Bank (GBT) [select from list of observatories]
Time Span:      [put in desired values]
Table Settings:  QUANTITIES=1,3,20
                 i.e., (1)Astrometric RA&Dec, (3)rates RA&Dec, and (20)Range and range rate
Display/Output:  plain text

Use the web browser file menu to save the output file as (for example) cometfilename.txt
```

Script 6.29: `jpl2astrid` usage

If you give it a file name, say by typing “`jpl2astrid jplephemfile.txt`”, it produces another file in the form for Astrid Catalogs. You should verify that the first non-comment line of the resulting catalog file contains:

```
FORMAT = EPHEMERIS
```

You now have a valid catalog file that Astrid will be able to use. When you load the catalog into Astrid, make sure you have the correct path and that the name of the comet is exactly what is in the `.astrid` catalog file in “quotations”. The catalog file should look something like this:

```
FORMAT = EPHEMERIS
VELDEF = VRAD-TOP
COORDMODE = J2000
HEAD = date utc ra dec dra ddec
# 1: soln ref.= JPL#252
NAME = 103P/Hartley
2016-Apr-05 00:00 18:27:00.30 -13:25:32.9 12.7582 9.7863
2016-Apr-05 00:05 18:27:00.37 -13:25:32.0 12.7527 9.7841
2016-Apr-05 00:10 18:27:00.44 -13:25:31.2 12.7469 9.7818
2016-Apr-05 00:15 18:27:00.52 -13:25:30.4 12.7409 9.7796
```

Script 6.30: An Astrid ephemeris catalog for generated by running `jpl2astrid` on Script 6.28] Astrid catalog generated by `jpl2astrid`] An Astrid ephemeris catalog for generated by running `jpl2astrid` on Script 6.28

**Note:** You may wish to edit the “`NAME =`” line to rename the object. The name of the object will be used in SBs as an argument to scan functions such as **Track**.

### 6.3.6 NNTLE : Tracking Earth satellites

“NNTLE” stands for NASA/NORAD Two-Line Elements. This refers to a standard NASA format for orbital elements for Earth satellites (see e.g. <http://ghrc.msfc.nasa.gov/orbit/tleformat.html> or <http://www.amsat.org/amsat/keps/formats.html>). The first non-comment line of the Catalog must contain:

```
FORMAT = NNTLE
```

If the FILE keyword is used then one should only give the name of the object in the Catalog as the elements of the orbit are retrieved from the file or URL. Note that the full path name of the file must be given, and the file must have world read permission.

The remainder of the non-comment lines contain the names for one or more satellites and their orbital elements in the NASA/NORAD Two-Line Element format.

An example of a valid file is as follows (data taken from the AMSAT URL listed above):

```
FORMAT = NNTLE
USERADVEL = 1      # optional keyword
#
OSCAR10
1 14129U           88230.56274695 0.00000042           10000-3 0 3478
2 14129 27.2218 308.9614 6028281 329.3891 6.4794 2.05877164 10960
GPS-0008
1 14189U           88230.24001475 0.00000013           0 5423
2 14189 63.0801 108.8864 0128028 212.9347 146.3600 2.00555575 37348
```

Script 6.31: An example of a valid NNTLE format Catalog file

When implementing an NNTLE catalog, the scantype function will pass the 3 lines to a program that will calculate positions for the antenna, given the scan start time and duration. The source name is the string that appears on the first of the three lines, and that is what one would pass to the scan function.

It may also be convenient to use TLEs on a file or website as shown in Scripts 6.32 and 6.33.

```
FORMAT      = NNTLE
USERADVEL   = 0
FILE        = /home/astro-util/projects/GBTog_examples/gps-ops.txt
Name        = 'GPS BIIR-2'
```

Script 6.32: An example NNTLE format Catalog file using the “FILE” keyword

```
FORMAT      = NNTLE
USERADVEL   = 0
URL         = http://www.celestrak.com/NORAD/elements/gps-ops.txt
Name        = 'GPS BIIR-2'
```

Script 6.33: An example NNTLE format Catalog file using the “URL” keyword

The first set of orbital elements whose name matches the name listed in the file will be used for calculating the satellite position. Note that the generation of tracks for satellites is based on “pyephem”, an implementation of xephem in Python.

## 6.4 Scan Types

A Scan is a pattern of antenna motions that when used together yield a useful scientific dataset. This section describes the various scan types that are available for use within GBT SBs. Each scan type consists of one or more scans, which are the individual components of the antenna’s motion on the sky. The scan types listed below are the functions within your SB where data will be obtained with the GBT.

Please note that the syntax for all Scan Types is case-sensitive. Location, Offset, Horizon, and Time objects are defined in § 6.6 while Catalogs are defined in § 6.3. Seldom used scan types are discussed in Appendix D.2. Nearly all scans use the following parameters:

### Location

Most Scan commands require a “location” parameter. This may be either a Location object (see § 6.6 and § 6.6.1, or it may be the name of a radio source given in a Catalog (see § 6.3).

### beamName

Most Scan commands use a “beamName” parameter. This should not be confused with the **beam** keyword in Configurations (see § 6.2.5). This indicates the “tracking beam” i.e., the beam that is pointed at the specified location. It may have values ‘1’, ‘2’, ‘3’, up to the maximum beam number for the specified receiver. The beam numbers and their relative locations depend on the receiver. A value of ‘C’ means the center of the receiver box, where there may or may not be a feed. Syntax such as ‘MR12’ means halfway between beams 1 and 2. These are used for subreflector nodding, (see “SubBeamNod” in § 6.4.2).

Note that the Ka–band (26–40 GHz) receiver **can** use beamName=‘1’, ‘2’, ‘C’ or ‘MR12’ even though the **beam** keyword in the configuration only allows beam=‘1’. **beamName** refers to the tracking beam of a scan function and not the **beam** configuration in the IF system. The IF system labels the two single-polarization beams of the Ka–band receiver as two separate polarizations of a single beam. Therefore, the only allowable setting for **beam** in the configuration (not beamName in a scan function) is beam=‘1’ (see § 6.2.5).

### scanDuration

A float value specifying the length of a scan in seconds. Note that scan type procedure may consist of a series of subscans. In these cases “scanDuration” refers to the length of each subscan, not the time required to complete the full scan type procedure. For example, the **OnOff** scan type, typically used for position switching observations, will perform a pair of subscans: one ‘on’-source, and one ‘off’-source. If scanDuration is set to 60.0, then the full procedure will require 2 minutes to complete (60 seconds for the ‘on’ scan and then 60 seconds for the ‘off’ scan).

### 6.4.1 Utility Scans

Utility scans generally describe procedures that are used to calibrate some aspect of the system such as pointing, focus, or power levels. Nearly every observing session will require the use of one or more of the utility scans described in the following section and listed in table 6.5. Some utility scans may change the configuration of the system, so it is important to be aware of the following points:

#### – Configure immediately after any Auto\* procedure

**AutoPeakFocus**, **AutoPeak**, **AutoFocus** and **AutoOOF** automatically execute their own default continuum configurations unless “configure=False” has been supplied as an optional argument (only recommended for expert users). **Configure** should be used after each **Auto\*** procedure to reconfigure for the desired observing mode.

#### – Configure at the start of each observing session

**Auto\*** procedures use whatever receiver is currently loaded in the M&C system. A **Configure** should always be executed at the start of each session to ensure the correct receiver has been loaded.

Table 6.5: Utility Scan Types available for the GBT.

Scan Type	Observing Type	Description
AutoPeakFocus	Continuum	Selects and observes a nearby calibration source and updates the pointing and focus corrections.
AutoPeak	Continuum	Selects and observes a nearby calibration source and updates the pointing corrections.
AutoFocus	Continuum	Selects and observes a nearby calibration source and updates the focus correction.
AutoOOF	Continuum	Selects and observes a nearby calibration source with different focus settings to create an out-of-focus holography map to update the surface.
Focus	Continuum	Performs a focus observation.
Peak	Continuum, Line	Performs a pointing or cross observation.
Tip	Continuum, Line	Performs an observation to derive $T_{\text{sys}}$ vs. elevation.
Slew	Continuum, Line, Pulsar	Slews the telescope to the specified source or Location.
Balance	Continuum, Line, Pulsar	Balances the IF system so that each device is operating in its linear response regime
BalanceOnOff	Continuum, Line, Pulsar	Move from a source to a reference position and balance the IF system at the mid-point of the two power levels.

Table 6.6: Recommended (default) values for performing peak and focus observations.

Receiver	$\nu$ (MHz)	$\Delta\nu$ (MHz)	Beam -refBeam	Peak			Focus			Notes
				Beam FWHM	Length (')	Time (sec)	Focus FWHM	Length (mm)	Time (sec)	
Rcvr_342	340	20	1	36'	180	30	3.2m	—	—	A,C
Rcvr_450	415	20	1	30'	180	30	2.6m	—	—	A,C
Rcvr_600	680	20	1	18'	90	15	1.6m	—	—	A,C
Rcvr_800	770	20	1	16'	80	15	1.4m	—	—	A,C
Rcvr_1070	970	20	1	13'	65	15	1.1m	—	—	A,C
Rcvr1_2	1400	80	1	8.8'	130	30	76cm	480	60	B,D,E
Rcvr2_3	2000	80	1	6.2'	90	30	54cm	480	60	B,D,E
Rcvr4_6	5000	80	1	2.5'	40	30	22cm	480	60	B,D,E
Rcvr8_10	9000	80	1	1.4'	16	24	12cm	480	60	B,D,E
Rcvr12_18	14000	320	1-2	53"	18	30	76mm	320	60	B,D,F
RcvrArray18_26	25000	800	3-7	30"	9	30	43mm	240	60	B,D,F
Rcvr26_40	32000	320	1-2	23"	8	24	32mm	180	60	B,D,F
Rcvr40_52	43000	320	1-2	17"	6	30	25mm	120	60	B,D,F
Rcvr68_92	77000	320	1-2	10"	3	30	14mm	100	60	B,D

**A** Prime Focus: Peak Lengths are chosen to be 5 x FWHM with a scan time of 15 seconds to have good sampling across the beam.

**B** Gregorian Focus: Peak Rates are chosen to give 2 seconds across the FWHM, Peak Times to give a scan time of 30 seconds (to allow vibrations to settle).

**C** Prime Focus: Axial focus measurements are not recommended for prime focus receivers since the gain changes only slightly over the entire focus range.

**D** Gregorian Focus: The optimal focus length is 2 x FWHM, but to allow for varying baselines we currently recommend  $\sim 3$  x focus FWHM, plus 40mm at each end to allow for the fact that focus measurement is done with respect to focus tracking curve, not last offset. The Focus Rate is then chosen to give a 60sec scan time. This is a trade-off between completing the focus scan quickly, and allowing any potential scan-start anomalies to die away.

**E** Focus rates and lengths are conservative limits set by subreflector hardware (the absolute maximum would be 600mm/min and 600mm).

**F** Multi-beam receivers use a larger peak length to accommodate the beam separation in azimuth.

### 6.4.1.1 AutoPeakFocus

The intent of this scan type is to automatically peak and focus the antenna for the current location on the sky and with the current receiver. Therefore it should not require any user input. However, by setting any of the optional arguments the user may partially or fully override the search and/or procedural steps as described below.

**AutoPeakFocus()** should not be used with Prime Focus receivers. The prime focus receivers have pre-determined focus positions and there is not enough travel in the feed to move them significantly out of focus.

**Configure immediately after an AutoPeakFocus.** **AutoPeakFocus()** will execute its own default continuum configuration unless “configure=False” is supplied as an optional argument (not recommended). A **Configure** should be used after each AutoPeakFocus use to reconfigure for the desired observing mode.

#### SYNTAX:

**AutoPeakFocus**(source, location, frequency, flux, radius, balance, configure, beamName, gold)

**source** A string. It specifies the name of a particular source in the pointing catalog or in a user-defined Catalog. The default is None. Specifying a source bypasses the search process. Please note that NVSS source names are used in the pointing catalog. If the name is not located in the pointing catalog then all the user-specified catalogs previously defined in the Scheduling Block are searched. If the name is not in the pointing catalog or in the user defined catalog(s) then the procedure fails.

**location** A Catalog source name or Location object (see § 6.6.1). It specifies the center of the search radius. The default is the antenna’s current beam location on the sky. Planets and other moving objects may **not** be used.

**frequency** A float. It specifies the observing frequency in MHz. The default is the rest frequency used by the standard continuum configurations, or the current configuration value if “configure=False” (see Table 6.6).

**flux** A float. It specifies the minimum acceptable calibration flux in Jy at the observing frequency. The default is 20 times the continuum point-source sensitivity.

**radius** A float. The routine selects the closest calibrator within the radius (in degrees) having the minimum acceptable flux. The default radius is 10 degrees. If no calibrator is found within the radius, the search is continued out to 180 degrees and if a qualified calibrator is found the user is given the option of using it [default], aborting the scan, or continuing the scheduling block without running this procedure.

**balance** A Boolean. Controls whether after slewing to the calibrator the routine balances the power along the IF path and again to set the power levels just before collecting data. Allowed values are True or False. The default is True.

**configure** A Boolean. This argument causes the scan type to configure the telescope for continuum observing for the specified receiver. The default is True. **Note: because AutoPeakFocus() is self-configuring, one must re-configure the GBT IF path for your normal observing after the pointing and focus observations are done, unless the configure parameter is set to False.** Also be aware that setting configure to False means the observer must ensure the DCR is properly configured and included in the Scan Coordinator, as the AutoPeakFocus() procedures will not check the configuration of the GBT.

**beamName** A string. It specifies which receiver beam will be the center of the cross-scan. beamName can be ‘C’, ‘1’, ‘2’, ‘3’, ‘4’, etc, up to ‘7’ for the KFPFA receiver. The default value is the recommended value for the receiver. If you configure for one beam, and point with another (using the beamName parameter) you can have very, very bad data. Make sure that if you choose “configure=False” and “beamName” that the two are compatible.

**refBeam** A string. It specifies which receiver beam will be the reference beam for subtracting sky contribution to the pointing observations. The name strings are the same as for the **beamName** argument. Two beams used for pointing should be at the same elevation, ie. `beamName='7'`, `refBeam='3'` or `beamName='6'`, `refBeam='4'` for the KFFPA.

**gold** A Boolean. If True then only “Gold standard sources” (i.e. sources suitable for pointing at high frequencies) will be used by `AutoPeakFocus()`. This parameter is ignored if the “source” parameter is specified.

`AutoPeakFocus` will use the default scanning rates and lengths listed in Table 6.6. The sequence of events done by `AutoPeakFocus()` in full automatic mode, i.e, with no arguments are:

1. Get recommended beam, antenna/subreflector motions, and duration for peak and focus scans.
2. Get current receiver from the M&C system.
3. Get current antenna beam location from the control system.
4. Configure for continuum observations with the current receiver.
5. Run a balance (see § 6.4.1.9) to obtain accurate system temperature readings from the DCR.
6. Select a source using computed minimum flux, observing frequency, location, and search radius. If no pointing source is found within the specified radius, then provide the observer the option to use a more distant source (default), and if none found either aborting (second default) or continuing the scheduling block.
7. Slew to source.
8. Run a balance to set scan power levels.
9. Run a scan using Peak
10. Run a scan using Focus.

#### USAGE:

Script 6.34 gives examples demonstrating the expected use of `AutoPeakFocus`:

```
#Configure for correct receiver at start of session...
execfile('/home/astro-util/projects/GBTog/configs/tp_config.py')
Configure(tp_config)

#Default (fully automatic)
AutoPeakFocus()
#point and focus on 3C286
AutoPeakFocus('3C286')
# find a pointing source near ra=16:30:00 dec=47:23:00
AutoPeakFocus(location=Location('J2000', '16:30:00', '47:23:00'))

#AutoPeakFocus has executed its own configuration
#Reconfigure for science observations
Configure(tp_config)
```

Script 6.34: Examples demonstrating the expected use of `AutoPeakFocus`.

### 6.4.1.2 AutoPeak

**AutoPeak()** is the same as **AutoPeakFocus()** except that it does not perform a focus scan.

**Configure immediately after an AutoPeak.** **AutoPeak()** will execute its own default continuum configuration unless “configure=False” is supplied as an optional argument (not recommended).

**SYNTAX:**

**AutoPeak**(source, location, frequency, flux, radius, balance, configure, beamName, gold)

**Parameter descriptions:** See **AutoPeakFocus()**

**USAGE:** See **AutoPeakFocus()**

### 6.4.1.3 AutoFocus

**AutoFocus()** is the same as **AutoPeakFocus()** except that it does not perform pointing scans.

**AutoFocus()** **should not be used with Prime Focus receivers.** The prime focus receivers have pre-determined focus positions and there is not enough travel in the feed to move these receivers significantly out of focus.

**Configure immediately after an AutoFocus.** **AutoFocus()** will execute its own default continuum configuration unless “configure=False” is supplied as an optional argument (not recommended).

**SYNTAX:**

**AutoFocus**(source, location, frequency, flux, radius, balance, configure, beamName, gold)

**Parameter descriptions:** See **AutoPeakFocus()**

**USAGE:** See **AutoPeakFocus()**

### 6.4.1.4 AutoOOF

“OOF” (Out-Of-Focus holography) is a technique for measuring large-scale errors in the shape of the reflecting surface by mapping a strong point source both in and out of focus. The procedure derives surface corrections which can be sent to the active surface controller to correct surface errors. The procedure is recommended for high-frequency observing at frequencies of 26 GHz and higher. Recommended strategies for using AutoOOF can be found in § 7.2.

**AutoOOF()** **should only be used for observations above 26 GHz.** Receiver choices are limited to ‘Rcvr26\_40’, ‘Rcvr40\_52’, ‘Rcvr68\_92’, and ‘Rcvr\_PAR’ (MUSTANG).

**Configure immediately after an AutoOOF.** **AutoOOF()** will execute its own default continuum configuration unless “configure=False” is supplied as an optional argument (not recommended).

**SYNTAX:**

**AutoOOF**(source, location, frequency, flux, radius, balance, configure, beamName, gold)

**Parameter descriptions:** **AutoOOF** uses the same parameters as **AutoPeakFocus** with only a few minor changes:

- **receiver** choices are limited to ‘Rcvr26\_40’, ‘Rcvr40\_52’, ‘Rcvr68\_92’, and ‘Rcvr\_PAR’
- **nseq** is an optional parameter for use with ‘Rcvr\_PAR’. It is used to specify the number of OTF maps made with **AutoOOF** and may take values of 3 or 5.

**USAGE:**

**AutoOOF** is used in a similar manner to **AutoPeakFocus**. The command normally does not require any arguments, although it is prudent to specify a source with a flux density of at least 3-4 Jy. If you don't know the name of a bright nearby calibrator, you may alternatively specify a flux density cutoff, but beware that the flux density database is not kept current. Both methods are shown in Script 6.35

```
#Configure for correct receiver at start of session...
execfile('/home/astro-util/projects/GBTog/configs/wconfig.py')
Configure(wband_config)

#Specifying a source for AutoOOF
AutoOOF('2253+1608')

#Let AutoOOF find a source > 3 Jy near the specified location
AutoOOF(location=Location('J2000','16:30:00','47:23:00'),flux=3.0)
Break('Examine solutions and send them to the active surface')

#AutoOOF has executed its own configuration
#Reconfigure for science observations
Configure(wband_config)
```

Script 6.35: Examples demonstrating the expected use of AutoOOF.

**Ka-band (Rcvr26\_40):**

If the current receiver is Rcvr26\_40 (Ka-band 26-40 GHz), then **AutoOOF** will automatically configure for the CCB using the second highest frequency channel (34.25 GHz) because it provides significantly better receiver temperature than the highest frequency band.

In case the preferred backend (CCB) is not available, the DCR can be used instead. In order to do this, you must configure the DCR prior to calling **AutoOOF**. As an example, we provide a DCR configuration used by GB.PTCS shown in Script 6.36

```
#Configure for the Ka receiver and DCR backend...
execfile('/home/groups/ptcs/obs/turtle/configs.py')
Configure(kaband)

AutoOOF(source='2253+1608',configure=False)
```

Script 6.36: AutoOOF example demonstrating the use of the Ka-band 26-40 GHz receiver with the non-default DCR backend.

**MUSTANG (Rcvr\_PAR):**

MUSTANG must be set up and tuned before **AutoOOF** can be used as shown in Script 6.37.

```
#Configure for MUSTANG...
Configure('/users/bmason/mustangPub/sb/mustangfull.conf')
AutoOOF(source='1159+2914')
```

Script 6.37: AutoOOF example using MUSTANG.



### 6.4.1.5 Focus

The Focus scan type moves the subreflector or prime focus receiver (depending on the receiver in use) through the axis aligned with the beam. Its primary use is to determine focus positions for use in subsequent scans and is used almost exclusively with continuum observing.

**SYNTAX:** `Focus`( location, start, focusLength, scanDuration, beamName)

**location** A Catalog source name or Location object. It specifies the source upon which to do the scan.

**start** A float. It specifies the starting position of the subreflector (in mm) for the Focus scan. See Table 6.6 for the recommended value for each receiver.

**focusLength** A float. It specifies the ending position of the subreflector relative to the starting location (also in mm). See Table 6.6 for the recommended value for each receiver.

**scanDuration** A float. It specifies the length of each scan in seconds. See Table 6.6 for the recommended value for each receiver.

**beamName** A string. It specifies the receiver beam to use for the scan. beamName can be 'C', '1', '2', '3', '4' or any valid combination for the receiver you are using such as 'MR12'. The default for each receiver is listed in Table 6.6. If you configure for one beam, and focus with another (using the beamName parameter) you can have very, very bad data. Make sure that you configure with the same beam with which you Focus!!!

**USAGE:** The only required parameter for `Focus()` is location. In the following example a focus of the subreflector is performed from -200 to +200mm at 400mm/min using beam 1:

```
#Focus using default settings
Focus ('0137+3309')

#Focus from -200 to +200mm at 400mm/minute with beam 1
Focus ('0137+3309', -200.0, 400.0, 60.0, '1')
```

Script 6.38: Focus() example.

### 6.4.1.6 Peak

The Peak scan type sweeps through the specified sky location in the four cardinal directions. Its primary use is to determine pointing corrections for use in subsequent scans. Note that the hLength, vLength and scanDuration should be overridden as a unit since together they determine the rate.

**SYNTAX:** `Peak`( location, hLength, vLength, scanDuration, beamName)

**location** A Catalog source name or Location object. It specifies the source upon which to do the scan.

**hLength** An Offset object. It specifies the horizontal distance used for the Peak. **hLength** values may be negative. The default value is the recommended value for the receiver (see Table 6.6).

**vLength** An Offset object. It specifies the vertical distance used for the Peak. **vLength** values may be negative. The default value is the recommended value for the receiver (see Table 6.6).

**scanDuration** A float. It specifies the length of each scan in seconds. The default value is the recommended value for the receiver (see Table 6.6).

**beamName** A string. It specifies the receiver beam to use for the scan. beamName can be 'C', '1', '2', '3', '4' or any valid combination for the receiver you are using such as 'MR12'. The default for each receiver is listed in Table 6.6. If you configure for one beam, and peak with another (using the beamName parameter) you can have very, very bad data. Make sure that you configure with the same beam with which you Peak!!!

**USAGE:** The only required parameter for `Peak()` is location. The following example does a Peak in encoder coordinates with 90 minute lengths and a 30 second scan duration using beam 1.

```
#Peak using default settings
Peak('0137+3309')

#Peak using encoder coordinates with scans of 90' length in 30 sec
Peak('0137+3309', Offset('Encoder', '00:90:00', 0),
      Offset('Encoder', 0, '00:90:00'), 30, '1')
```

Script 6.39: Peak() example.

### 6.4.1.7 Tip

The Tip scan moves the beam on the sky from one elevation to another elevation while taking data and maintaining a constant azimuth.

**SYNTAX:** `Tip`( location, endOffset, scanDuration, beamName, startTime, stopTime)

**location** A Catalog source name or Location object. It specifies the start location of the tip scan. The Location must be in AzEl or encoder coordinates.

**endOffset** An Offset object. It specifies the beam's final position for the scan, relative to the location specified in the first parameter. The Offset also must be in AzEl or encoder coordinates.

**scanDuration** A float. It specifies the length of each scan in seconds.

**beamName** A string. It specifies the receiver beam to use for the scan. beamName can be 'C' (center), '1', '2', '3', '4' or any valid combination for the receiver you are using such as 'MR12' (i.e., track halfway between beams 1 and 2). The default value for beamName is '1'.

**startTime** A time string with the following format: 'hh:mm:ss'. It allows the observer to specify a start time for the Tip.

**stopTime** A time string with the following format: 'hh:mm:ss'. It allows the observer to specify a stop time for the Tip.

**USAGE:** Scan timing may be specified by either a scanDuration, a stopTime, a startTime plus stopTime, or a startTime plus scanDuration. The following example tips the GBT from 6° in elevation to 80° in elevation over a period of five minutes using beam 1:

```
#Tip from 6 to 80 degrees elevation over 5 minutes with beam 1
Tip(Location('AzEl', 1.5, 6.0),
     Offset('AzEl', 0.0, 74.0), 300.0, '1')
```

Script 6.40: Tip() example.

### 6.4.1.8 Slew

Slew moves the telescope beam to point to a specified location on the sky without collecting any data. Note that once `Slew()` is complete, the location will continue to be tracked at a sidereal rate until a new command is issued.

**SYNTAX:** `Slew( location, offset, beamName)`

**location** A Catalog source name or Location object. It specifies the source to which the telescope should slew. The default is the current location in “J2000” coordinate mode.

**offset** An Offset object. It moves the beam to an optional offset position that is specified relative to the location specified in the location parameter value. The default is None. See § 6.6 for information on Offset objects.

**beamName** A string. It specifies the receiver beam to use for the scan. beamName can be ‘C’, ‘1’, ‘2’, ‘3’, ‘4’ or any valid combination for the receiver you are using such as ‘MR12’. The default is ‘1’.

**USAGE:** Slew does the following based on the arguments provided:

1. If only a location is given the antenna slews to the indicated position.
2. If a location and offset are given, the antenna slews to the indicated position plus the offset.
3. If only an offset is given, the antenna slews to the current location plus the specified offset.

The following example slews to 3C 48 using the center of all the receiver’s beams:

```
Slew('3C48', beamName='C') #Slew to 3c48 using the center of all beams
```

Script 6.41: Slew() example.

### 6.4.1.9 Balance

The `Balance()` command is used to balance the electronic signal throughout the GBT IF system so that each device is operating in its linear response regime. `Balance()` will work for any device with attenuators and for a particular backend. Individual devices can be balanced, such as the Prime Focus receivers, the IF system, the DCR, GUPPI and VEGAS (The Gregorian receivers lack attenuators and do not need to be balanced). If the argument to `Balance()` is blank (recommended usage), then all devices for the current state of the IF system will be balanced.

**RECOMMENDED SYNTAX:** `Balance()`

**ADVANCED SYNTAX:** `Balance( 'DeviceName', {'DeviceKeyword':Value} )`

**Parameter descriptions** See Appendix E for details on the advanced use of `Balance()`.

**USAGE:** Without any arguments, the `Balance()` command uses the last executed configuration to decide what hardware will be balanced. Strategies for balancing the hardware in the GBT IF system are discussed in § 7.3. The following script gives a simple example showing the expected use of `Balance()`:

```
execfile('/home/astro-util/projects/GBTog/configs/tp_config.py')
Configure(tp_config) #Execute the desired configuration
Slew('3C286') #Slew so that you may balance 'on-source'
Balance() #Balance the IF and devices for your configuration
```

Script 6.42: Balance() example.

### 6.4.1.10 BalanceOnOff

When there is a large difference in power received by the GBT between two positions on the sky, it is advantageous to balance the IF system power levels to be at the mid-point of the two power levels. Typically this is needed when the “source position” is a strong continuum source. This scan type has been created to handle this scenario; one should consider using it when the system temperature on and off source differ by a factor of two or more.

**BalanceOnOff()** slews to the source position and then balances the IF system. It then determines the power levels that are observed in the IF Rack. Then the telescope is slewed to the off position and the power levels are determined again. The change in the power levels is then used to determine attenuator settings that put the balance near the mid-point of the observed power range. Note that the balance is determined only to within  $\pm 0.5$  dB owing to the integer settings of the IF Rack attenuators.

**SYNTAX:** **BalanceOnOff**( location, offset, beamName)

**location** A Catalog source name or Location object. It specifies the source to which the telescope should slew. The default is the current location in “J2000” coordinate mode.

**offset** An Offset object. It moves the beam to an optional offset position that is specified relative to the location specified in the location parameter value. The default is None. See § 6.6 for information on Offset objects.

**beamName** A string. It specifies the receiver beam to use for the scan. beamName can be ‘C’, ‘1’, ‘2’, ‘3’, ‘4’ or any valid combination for the receiver you are using such as ‘MR12’. The default is ‘1’.

**USAGE:** The following example balances on 3C 48 and remeasures  $1^\circ$  off:

```
execfile('/home/astro-util/projects/GBTog/configs/tp_config.py')
Configure(tp_config) #Execute the desired configuration
BalanceOnOff('3C48', Offset('J2000', 1.0, 0.0))
```

Script 6.43: BalanceOnOff() example.

## 6.4.2 Observing Scans

Observing scan types will acquire scientific datasets by performing one or more scans at specific locations on the sky. Available GBT observing scans are listed in Table 6.7 and are described in the following section.

Table 6.7: Observing Scan Types available for the GBT.

Scan Type	Observing Type	Description
Track	Continuum, Line, Pulsar	Takes data at a single position or while moving with constant velocity.
OnOff	Continuum, Line	Observe a source and then a reference position.
OffOn	Continuum, Line	Observe a reference position and then a source.
OffOnSameHA	Continuum, Line	Observe a source and then a reference position using the same hour angle as the source observations.
Nod	Continuum, Line	Observe a source with one beam and then with another beam.
SubBeamNod	Continuum, Line	Moves the subreflector alternately between two beams.

### 6.4.2.1 Track

The Track scan type follows a sky location while taking data.

#### SYNTAX:

**Track**( location, endOffset, scanDuration, beamName, startTime, stopTime, fixedOffset )

**location** A Catalog source name or Location object. It specifies the source which is to be tracked.

**endOffset** An Offset object (see § 6.6 for information on Offset objects).

Supplying an endOffset object with a value other than **None** will track the telescope across the sky at constant velocity. The scan will start at the specified **location** and end at (**location+endOffset**) after **scanDuration** seconds. If you wish to only track a single location rather than slew the telescope between two points, use **None** for this parameter.

**scanDuration** A float. This specifies the length of the scan in seconds.

**beamName** A string. It specifies the receiver beam to use for the scan. beamName can be 'C', '1', '2', '3', '4' or any valid combination for the receiver you are using such as 'MR12'. The default value for beamName is '1'.

**startTime** A time object. This specifies when the scan begins. If **startTime** is in the past then the scan starts as soon as possible with a message sent to the scan log. If (**startTime+scanDuration**) is in the past, then the scan is skipped with a message to the observation log. The value may be:

- **A time object** Note, if startTime is more than ten minutes in the future then a message is sent to the observation log. See § 6.6 for information on time objects.
- **A Horizon object** The following script implicitly calculates the **startTime** using Horizon():

```
Track('VirgoA', None, 120.0, startTime=Horizon())
```

Script 6.44: Track() example using startTime=Horizon(). If the source never rises then the scan is skipped and if the source never sets then the scan is started immediately. In either case a message is sent to the observation log. See § 6.6 for information on Horizon objects.

**stopTime** A time object (see § 6.6 for information on time objects). This specifies when the scan completes. If **stopTime** is in the past then the scan is skipped with a message to the observation log. The value may also be:

- **A Horizon Object** When a Horizon object is used, the stop time is implicitly computed. The following lines in an SB would track VirgoA from rise to set using a horizon of 20°:

```
horizon = Horizon(20.0)
Track('VirgoA', None, 120.0, startTime=horizon, stopTime=horizon)
```

Script 6.45: Track() example using a Horizon object in startTime and stopTime. If the source never sets, then the scan stop time is set to 12 hours from the current time. See § 6.6 for information on Horizon objects.

**fixedOffset** An Offset object (see § 6.6 for information on Offset objects). **Track** follows the sky location plus this fixed Offset. The **fixedOffset** may be in a different coordinate mode than the **location**. If an **endOffset** is also specified, **Track** starts at (**location+fixedOffset**), and ends at (**location+fixedOffset+endOffset**). The **fixedOffset** and **endOffset** must be both of the same coordinate mode, but may be of a different mode than the **location**. The **fixedOffset** parameter may be omitted.

**USAGE:** `location` and `endOffset` are required parameters. Scan timing must be specified by either a `scanDuration`, a `stopTime`, a `startTime` plus `stopTime`, or a `startTime` plus `scanDuration`. Examples of `Track` are shown in Script 6.46:

```
#Example 1 - track 3C48 for 60 seconds using the center beam
Track('3C48', None, 60.0)

#Example 2 - track a position offset by 1 degree in elevation
Track('3C48', None, 60.0, fixedOffset=Offset('AzEl', 0.0, 1.0) )

#Example 3 - scan across the source from -1 to +1 degrees in azimuth
Track('3C48', Offset('AzEl', 2.0, 0.0), 60.0,
      fixedOffset=Offset('AzEl', -1.0, 0.0))
```

Script 6.46: Examples of the `Track()` scan function.

### 6.4.2.2 OnOff

The `OnOff` scan type performs two scans. The first scan is on source, and the second scan is at an offset from the source location used in the first scan.

**SYNTAX:** `OnOff( location, referenceOffset, scanDuration, beamName )`

**location** A Catalog source name or Location object. It specifies the source upon which to do the “On” scan.

**referenceOffset** An Offset object. It specifies the location of the “Off” scan relative to the location specified by the first parameter.

**scanDuration** A float. It specifies the length of each scan in seconds.

**beamName** A string. It specifies the receiver beam to use for both scans. `beamName` can be ‘C’, ‘1’, ‘2’, ‘3’, ‘4’ or any valid combination for the receiver you are using such as ‘MR12’. The default value for `beamName` is ‘1’.

**USAGE:** The following example does an `OnOff` scan with reference offsets of 1 degree of arc in Right Ascension and 1 degree of arc in Declination and a 60 second scan duration (120 seconds total), using beam 1:

```
OnOff('0137+3309', Offset('J2000', 1.0, 1.0, cosv=False), 60, '1')
```

Script 6.47: `OnOff()` example.

### 6.4.2.3 OffOn

The `OffOn` scan type is the same as the `OnOff` scan except that the first scan is offset from the source location.

**SYNTAX:** `OffOn( location, referenceOffset, scanDuration, beamName )`

**Parameter descriptions:** See `OnOff`

**USAGE:** The following example does an `OffOn` scan with reference offsets of 1 degree of arc in Right Ascension and 1 degree of arc in Declination and a 60 second scan duration (120 seconds total), using beam 1:

```
OffOn('0137+3309', Offset('J2000', 1.0, 1.0, cosv=False), 60, '1')
```

Script 6.48: `OffOn()` example.

### 6.4.2.4 OnOffSameHA

The OnOffSameHA scan type performs two scans. The first scan is on the source, and the second scan follows the same HA track used in the first scan.

**SYNTAX:** `OnOffSameHA( location, scanDuration, beamName )`

**location** A Catalog source name or Location object. It specifies the source upon which to do the On scan.

**beamName** A string. It specifies the receiver beam to use for both scans. beamName can be 'C', '1', '2', '3', '4' or any valid combination for the receiver you are using such as 'MR12'. The default value for beamName is '1'.

**scanDuration** A float. It specifies the length of each scan in seconds.

**USAGE:** The following example does an **OnOffSameHA** scan with a 60 second scan duration (120 seconds total), using beam 1:

```
OnOffSameHA('0137+3309', 60, '1')
```

Script 6.49: OnOffSameHA() example.

### 6.4.2.5 Nod

The Nod procedure does two scans on the same sky location with different beams. **Nod should only be used with multi-beam receivers.**

**SYNTAX:** `Nod( location, beamName1, beamName2, scanDuration )`

**location** A Catalog source name or Location object. It specifies the source upon which to do the Nod.

**beamName1** A string. It specifies the receiver beam to use for the first scan. beamName1 can be 'C', '1', '2' or any valid combination for the receiver you are using such as 'MR12'.

**beamName2** A string. It specifies the receiver beam to use for the second scan. beamName2 can be 'C', '1', '2' or any valid combination for the receiver you are using such as 'MR12'.

**scanDuration** A float. It specifies the length of each scan in seconds.

**USAGE:** The following example does a **Nod** between beams 3 and 7 with a 60 second scan duration (120 seconds total).

```
Nod('1011-2610', '3', '7', 60.0)
```

Script 6.50: Nod() example.

### 6.4.2.6 SubBeamNod

For multi-beam receivers **SubBeamNod** causes the subreflector to tilt about its axis between two feeds at the given periodicity. The primary mirror is centered on the midpoint between the two beams. The beam selections are extracted from the scan’s beamName, i.e., ‘MR12’. The “first” beam (‘1’) performs the first integration. The periodicity is specified in seconds (float) per nod (half-cycle). A nod is limited to a minimum of 4.4 seconds for a half cycle. **Note:** For KPFA observations, it is preferred that observers use the **Nod** directive for superior baseline performance.

**SYNTAX:** **SubBeamNod**( location, scanDuration, beamName, nodLength, nodUnit )

**location** A Catalog source name or Location object. It specifies the source upon which to do the nod.

**scanDuration** A float. It specifies the length of each subscan in seconds.

**beamName** A string. It specifies the receiver beam pair to use for nodding. beamName can be ‘MR12’.

**nodLength** Type depends on value of **nodUnit**: integer for ‘integrations’, and float or integer for ‘seconds’. It specifies the half-cycle time which is the time spent in one position plus move time to the second position.

**nodUnit** A string, either ‘integrations’ or ‘seconds’. The default is ‘seconds’.

**USAGE:** The following examples both do a **SubBeamNod** between beams 1 and 2. The first uses the default ‘seconds’ **nodUnit** and the second sets the **nodLength** in units of the primary backend’s integration time (see **tint** in § 6.2.5).

```
#Example 1 - nodLength units in seconds (default)
SubBeamNod('3C48', scanDuration=60.0, beamName='MR12',
           nodLength=4.4826624)

#Example 2 - nodLength units are "tint" as set in the configuration
SubBeamNod('3C48', scanDuration=60.0, beamName='MR12',
           nodLength=3, nodUnit='integrations')
```

Script 6.51: SubBeamNod() example.

If the backend’s actual integration time is obtainable then a warning is issued if the alignment between the integration times and the nod times shift over the duration of the scan by more than 10% of the nod time. A warning is issued in any case if the backend’s actual integration time is not obtainable. Attempting to use integrations as the unit when the integration time cannot be obtained from the selected backend will cause a failure.

The scan will end at the end of the scanDuration (once the current integration is complete) regardless of the phase of the nod cycle. When the subreflector is moving the entire integration during which this occurs is flagged. It takes about 0.5 seconds for the subreflector to move between beams plus additional time to settle on source (total time is ~ 2 seconds for Rcvr68\_92 and ~ 1 second for all other receivers).

For example, if we had previously configured for Rcvr26\_40 and an integration time of 1.5 seconds (tint=1.5 in the configuration), example 2 in script 6.51 would blank one out of every three integrations in a half-cycle (nodLength=3) while the subreflector was moving between beams. If nodLength=5, then only one in five integrations would be blanked.

The antenna uses the average position of the two beams for tracking the target, and SDFITS reports the positions of the beams relative to the tracking position. Although the SDFITS header position will not match the target position, **SubBeamNod** successfully nods between the two beams during the scan. Control of the subreflector may be done with any scan type using the submotion class. This should only be done by expert observers. Those observers interested in using this class should contact their GBT “Friend”.



### 6.4.3 Mapping Scans

Mapping scan types will record data over specified areas of the sky. The GBT mapping procedures are described in the following section and listed in table 6.8.

- **On-The-Fly (OTF) mapping:** Data are recorded while the telescope slews across a region of the sky using a specified trajectory (also known as raster scanning). OTF mapping is described in Mangum, Emerson, and Greisen (2007, A&A 474, 679). This is more efficient than point mapping, and may also minimize changes in the system and atmosphere by slewing rapidly across the sky.
- **Point mapping:** A region of the sky is divided into a grid of discrete positions. Data will then be recorded at each of these locations for a specified amount of time. This method is more simple than OTF mapping and may be suitable when data are required for a few specific locations.

Most GBT mapping procedures have versions that allow for periodic reference observations. These may be used to correct for the instrumental bandpass shape in total power observations during data reduction. For OTF mapping, some observers may prefer to use the edge pixels of the map as a reference position if they are suitably “off-source”.

The GBT mapping calculator is a useful tool for planning mapping observations. It may be used to provide Astrid commands and parameters for many of the mapping scan types. The mapping calculator can be found at <http://www.gb.nrao.edu/~rmaddale/GBT/GBTMappingCalculator.html>.

Table 6.8: Mapping Scan Types available for the GBT.

Scan Type	Observing Type	Description
RALongMap	Continuum, Line	Make an OTF raster map by moving along the major axis of the coordinate system.
RALongMapWithReference	Continuum, Line	Make an OTF raster map by moving along the major axis of the coordinate system and making periodic reference observations.
DecLatMap	Continuum, Line	Make an OTF raster map by moving along the minor axis of the coordinate system.
DecLatMapWithReference	Continuum, Line	Make an OTF raster map by moving along the minor axis of the coordinate system and making periodic reference observations.
PointMap	Continuum, Line, Pulsar	Make a map using individual pointings.
PointMapWithReference	Continuum, Line, Pulsar	Make a map using individual pointings with periodic reference observations.
Daisy	Continuum, Line	Make an OTF map in the form of daisy petals.

### 6.4.3.1 RALongMap

A Right Ascension/Longitude (RALong) map performs an OTF raster scan centered on a sky location. Scans are performed along the major axis of the selected coordinate system. The starting point of the map is defined as  $(-hLength/2, -vLength/2)$  from the specified center location.

**SYNTAX:** `RALongMap( location, hLength, vLength, vDelta, scanDuration, beamName, unidirectional, start, stop )`

**location** A Catalog source name or Location object. It specifies the center of the map.

**hLength** An Offset object. It specifies the horizontal width of the map (i.e., the extent in the longitude-like direction). **hLength** values may be negative.

**vLength** An Offset object. It specifies the vertical height of the map (i.e., the extent in the latitude-like direction). **vLength** values may be negative.

**vDelta** An Offset object. It specifies the distance between map rows. **vDelta** values must be positive.

**scanDuration** A float. It specifies the length of each scan in seconds.

- **Note:** Observers should limit **scanDuration** so that no more than 2 scans (or accelerations) are performed per minute. Overhead is  $\sim 20$  seconds per scan.

**beamName** A string. It specifies the receiver beam to use for the scan. `beamName` can be 'C', '1', '2', '3', '4' or any valid combination for the receiver you are using such as 'MR12'. Default is '1'.

**unidirectional** A Boolean. It specifies whether the map is unidirectional (True) or boustrophedonically<sup>2</sup> (False). Default is False.

**start** An integer. It specifies the starting row for the map. The default value for start is 1. This is useful for doing parts of a map at different times. For example, if map has 42 rows, one can do rows 1-12 by setting "start=1, stop=12", and later finishing the map using "start=13, stop=42".

**stop** An integer. It specifies the stopping row for the map. The default value for stop is None, which means "go to the end".

#### USAGE:

Script 6.52 produces a map with 41 rows each 120' long, using a row spacing of 3' and scan rate of 20'/min with beam 1 (default). A plot showing the actual trajectory of the antenna on the sky when script 6.52 was executed is shown in figure 6.3. Note that the blue dots in figure 6.3 mark timestamps of the data sampled along the trajectory.

Observers should ensure that they are sampling sufficiently in the scanning direction when using OTF mapping. In this example data were recorded every 5 seconds (`tint=5.0` in the configuration). This results in one sample every 1.67' in the scanning direction using the above scan rate of 20'/min. This is suitable for observations at 1420 MHz, where the FWHM of the beam is 8.8'.

```
RALongMap ('NGC4258', # center of the map
  Offset ('J2000', 2.0, 0.0, cosv=True), # 120' width
  Offset ('J2000', 0.0, 2.0, cosv=True), # 120' height
  Offset ('J2000', 0.0, 0.05, cosv=True), # 3' row spacing
  360.0) # 6 minutes per row
```

Script 6.52: `RALongMap()` example.

<sup>2</sup>from the Greek meaning "as the ox plows" i.e. back and forth

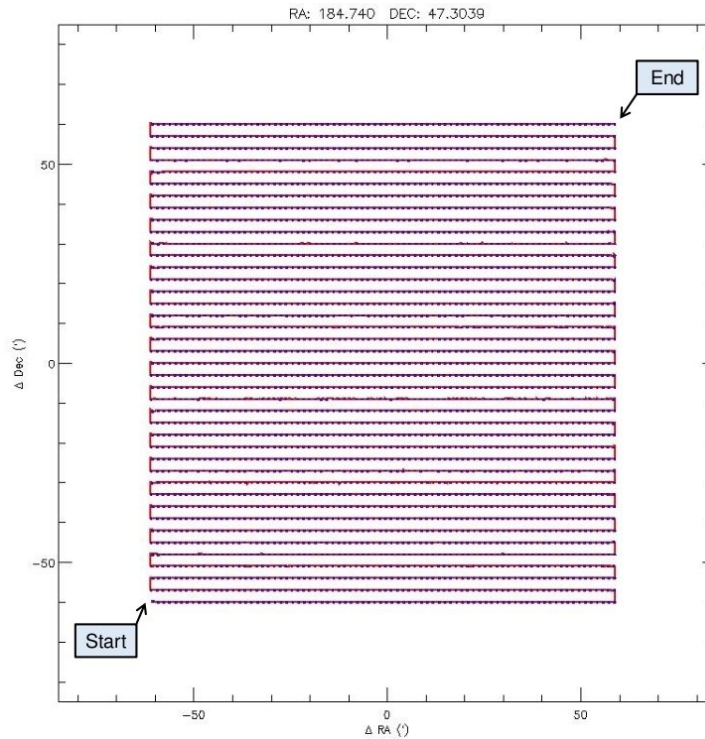


Figure 6.3: The actual GBT antenna trajectory (red) generated by executing script 6.52. Blue dots mark timestamps of sampled data. (sampling frequency is set via `tint` in the configuration).

### 6.4.3.2 RALongMapWithReference

**RALongMap** with periodic reference observations.

**SYNTAX:** `RALongMapWithReference( location, hLength, vLength, vDelta, referenceOffset, referenceInterval, scanDuration, beamName, unidirectional, start, stop )`

**Parameter Descriptions** See **RALongMap**. The following additional parameters are used to define the periodic reference observations:

- **referenceOffset:** An Offset object. It specifies the position of the reference source on the sky relative to the **Location** specified by the first input parameter.
- **referenceInterval:** An integer. It specifies when to do a reference scan in terms of map rows. For example, setting `referenceInterval=4` will periodically perform one scan on the reference source followed by 4 mapping scans.

**USAGE:** Script 6.53 produces a map with 6 rows each 60' long, using a row spacing of 6' and scan rate of 720'/min. A reference position will be observed once before every 3 rows. The sequence of scans will be: reference → rows 1-3 → reference → rows 4-6.

```
RALongMapWithReference ('CygA',           # center of map
  Offset ('J2000', 2.0, 0.0, cosv=True),  # 120' width
  Offset ('J2000', 0.0, 0.5, cosv=True),  # 60' height
  Offset ('J2000', 0.0, 0.1, cosv=True),  # 6' row spacing
  Offset ('J2000', 2.0, 0.0, cosv=True),  # 2 degree ref offset in RA
  3, 10.0)                                # ref before every 3 rows, 10 second scan duration
```

Script 6.53: `RALongMapWithReference()` example.

### 6.4.3.3 DecLatMap

A Declination/Latitude map performs an OTF raster scan centered on a sky location. Scans are performed in declination, latitude, or elevation coordinates depending on the desired coordinate system. The starting point of the map is defined as  $(-hLength/2, -vLength/2)$  from the specified center location.

**SYNTAX:** `DecLatMap( location, hLength, vLength, hDelta, scanDuration, beamName, unidirectional, start, stop )`

**Parameter Descriptions** See `RALongMap` with the following difference:

- **hDelta:** An Offset object. Similar to **vDelta** in `RALongMap`. It specifies the horizontal distance between map columns. **hDelta** values must be positive.

**USAGE:** Script 6.54 produces a map with 41 columns each 120' tall, using a column spacing of 3' and scan rate of 20'/min with beam 1 (default). A plot showing the actual trajectory of the antenna on the sky when script 6.54 was executed is shown in figure 6.4. Note that the blue dots in figure 6.4 mark timestamps of the data sampled along the trajectory.

```
DecLatMap ('NGC4258',           # center of the map
  Offset ('J2000', 2.0, 0.0, cosv=True), # 120' width
  Offset ('J2000', 0.0, 2.0, cosv=True), # 120' height
  Offset ('J2000', 0.05, 0.0, cosv=True), # 3' column spacing
  360.0)                                # 6 minutes per column
```

Script 6.54: DecLatMap() example.

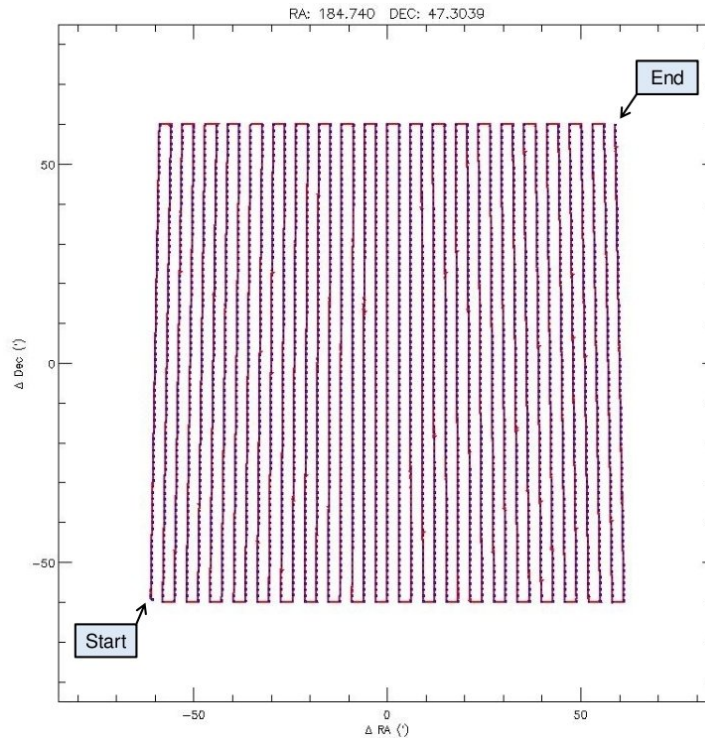


Figure 6.4: The actual GBT antenna trajectory (red) generated by executing script 6.54. Blue dots mark timestamps of sampled data. (sampling frequency is set via `tint` in the configuration).

### 6.4.3.4 DecLatMapWithReference

**DecLatMap** with periodic reference observations.

**SYNTAX:** **DecLatMapWithReference**( location, hLength, vLength, hDelta, referenceOffset, referenceInterval, scanDuration, beamName, unidirectional, start, stop )

**Parameter Descriptions** See **RALongMap** with the following difference:

- **hDelta:** An Offset object. Similar to **vDelta** in **RALongMap**. It specifies the horizontal distance between map columns. **hDelta** values must be positive.

The following parameters are used to define the periodic reference observations:

- **referenceOffset** An Offset object. It specifies the position of the reference source on the sky relative to the **Location** specified by the first input parameter.
- **referenceInterval** An integer. It specifies when to do a reference scan in terms of map columns. For example, setting **referenceInterval**=4 will periodically perform one scan on the reference source followed by 4 mapping scans.

**USAGE:** Script 6.55 produces a map with 6 columns each 60' long, using a column spacing of 6' and scan rate of 720'/min. A reference position will be observed once before every 3 columns. The sequence of scans will be: reference → columns 1-3 → reference → columns 4-6.

```
DecLatMapWithReference ('CygA',           # center of map
  Offset ('J2000', 2.0, 0.0, cosv=True), # 120' width
  Offset ('J2000', 0.0, 0.5, cosv=True), # 60' height
  Offset ('J2000', 0.0, 0.1, cosv=True), # 6' column spacing
  Offset ('J2000', 2.0, 0.0, cosv=True), # 2 degree ref offset in RA
  3, 10.0)                               # ref before every 3 columns, 10 second scan duration
```

Script 6.55: DecLatMapWithReference() example.

### 6.4.3.5 PointMap

A **PointMap**() constructs a map by sitting on fixed positions laid out on a grid. The starting point of the map is defined as  $(-\mathbf{hLength}/2, -\mathbf{vLength}/2)$ .

**SYNTAX:**

**PointMap**( location, hLength, vLength, hDelta, vDelta, scanDuration, beamName, start, stop )

**location** A Catalog source name or Location object. It specifies the center of the map.

**hLength** An Offset object. It specifies the horizontal width of the map. **hLength** values may be negative.

**vLength** An Offset object. It specifies the vertical height of the map. **vLength** values may be negative.

**hDelta** An Offset object. It specifies the horizontal distance between points in the map. **hDelta** values must be positive.

**vDelta** An Offset object. It specifies the vertical distance between points in the map. **vDelta** values must be positive.

**scanDuration** A float. It specifies the length of each scan in seconds.

**beamName** A string. It specifies the receiver beam to use for the scan. **beamName** can be 'C', '1', '2', '3', '4' or any valid combination for the receiver you are using such as 'MR12'. Default is '1'.

**start** An integer. It specifies the starting point for the map. The default value for **start** is 1. Note in **PointMap** this counts points, not stripes.

**stop** An integer. It specifies the stopping point for the map. The default value for **stop** is None, which means "go to the end".

**USAGE:** Script 6.56 produces a 9 point map using a  $3 \times 3$  grid. Points are separated by 10 arc-seconds in RA and 10 arc-seconds in Dec. Each point will be observed for 30 seconds using beam 1 (default). A plot showing the actual trajectory of the antenna on the sky when script 6.56 was executed is shown in figure 6.5. Note that the black crosses mark the average positions of sampled data at each point.

```

PointMap('W75N', # center of map
Offset('J2000', 20.0/3600.0, 0.00, cosv=True), # 20" width
Offset('J2000', 0.00, 20.0/3600.0, cosv=True), # 20" height
Offset('J2000', 10.0/3600.0, 0.00, cosv=True), # 10" horizontal spacing
Offset('J2000', 0.00, 10.0/3600.0, cosv=True), # 10" vertical spacing
30.0) # 30 second scan length

```

Script 6.56: PointMap() example.

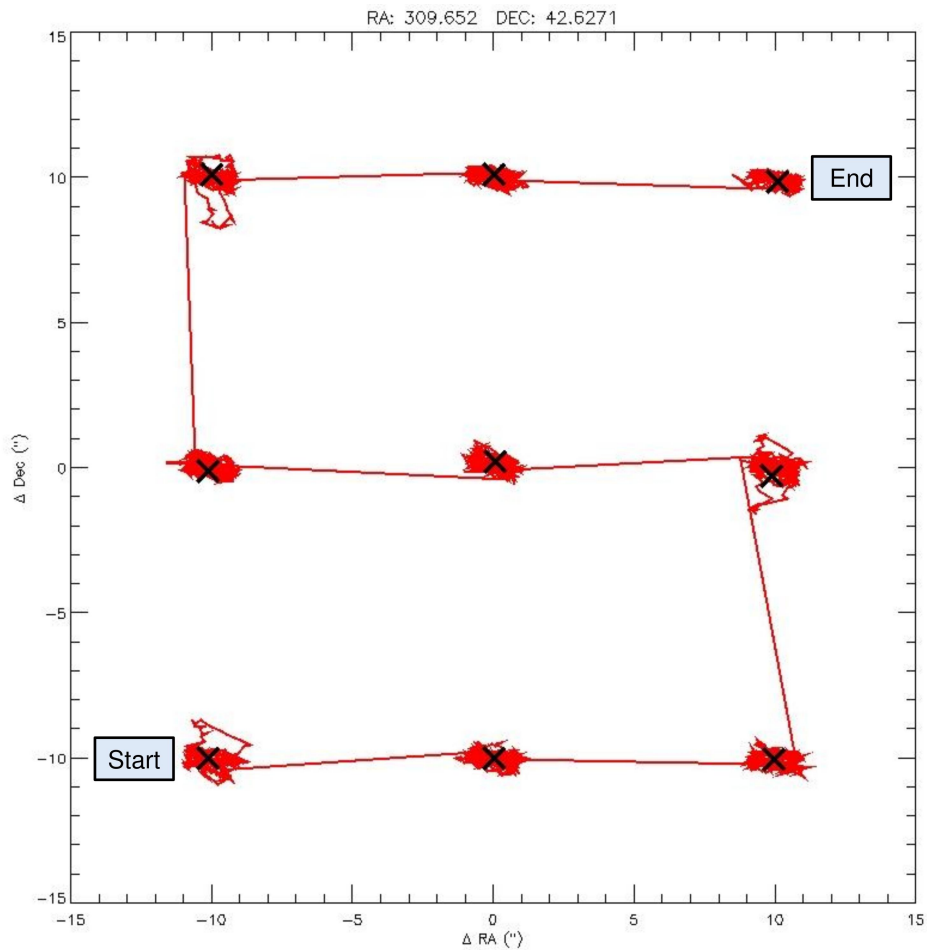


Figure 6.5: A plot of the actual GBT antenna trajectory (red) generated by executing script 6.56. The average positions of data sampled at each point are marked with black crosses.

### 6.4.3.6 PointMapWithReference

**PointMap** with periodic reference observations.

**SYNTAX:** `PointMapWithReference( location, hLength, vLength, hDelta, vDelta, referenceOffset, referenceInterval, scanDuration, beamName, start, stop )`

**Parameter Descriptions** See **PointMap**. The following additional parameters are used to define the periodic reference observations:

- **referenceOffset:** An Offset object. It specifies the position of the reference source on the sky relative to the **Location** specified by the first input parameter.
- **referenceInterval:** An integer. It specifies when to do a reference scan in terms of map points. For example, setting `referenceInterval=4` will periodically perform one scan on the reference source followed by 4 pointed scans.

**USAGE:** Script 6.57 produces a  $4 \times 4$  point map using beam 1 (default). A reference position will be observed before every 2 points. The sequence of scans will be: `reference(r) → points 1 and 2 (P1,2) → r → P3,4 → r → P5,6 → r → P7,8 → r → P9,10 → r → P11,12 → r → P13,14 → r → P15,16.`

```
PointMapWithReference('2023+2223',      # center of map
  Offset('B1950', 1.5, 0.0, cosv=True), # 90' width
  Offset('B1950', 0.0, 1.5, cosv=True), # 90' height
  Offset('B1950', 0.5, 0.0, cosv=True), # 30' horizontal step spacing
  Offset('B1950', 0.0, 5.0, cosv=True), # 30' vertical step spacing
  Offset('J2000', 3.0, 0.0, cosv=True), # 3 degree ref offset in RA
  2, 2.0)                               # ref before every 2 points, 2 second scan duration
```

Script 6.57: `PointMapWithReference()` example.

### 6.4.3.7 Daisy

The Daisy scan type performs an OTF scan around a central point in the form of daisy petals. It is a useful observing mode for focal plane arrays, allowing more integration time in the central field of view.

The **Daisy** scan will produce an approximately closed circular pattern on the sky after 22 radial oscillation periods (see figure 6.6b). For beam-sizes of  $20''$  FWHM or so, the circular area mapped will be fully sampled if the map radius is less than  $6'$ . It is not an especially useful observing mode for general-purpose single-beam mapping, since the largest “hole” in the map is  $\sim 0.3 \times$  the map radius.

Trajectories are generated according to:

$$\Delta \hat{x}(t) = \frac{r_0 \sin(2\pi t/\tau + \phi_1) \cos(2t/\tau + \phi_2)}{\cos(\hat{y}_0)} \quad (6.4)$$

$$\Delta \hat{y}(t) = r_0 \sin(2\pi t/\tau + \phi_1) \sin(2t/\tau + \phi_2) \quad (6.5)$$

$\hat{x}$  and  $\hat{y}$  are then major and minor coordinates of a sperical coordinate system,  $t$  is the time,  $r_0$  is the map radius,  $\tau$  is the radial oscillation period,  $\phi_1$  and  $\phi_2$  are the radial and rotational phases, and  $\hat{y}_0$  is the minor coordinate of the map center.

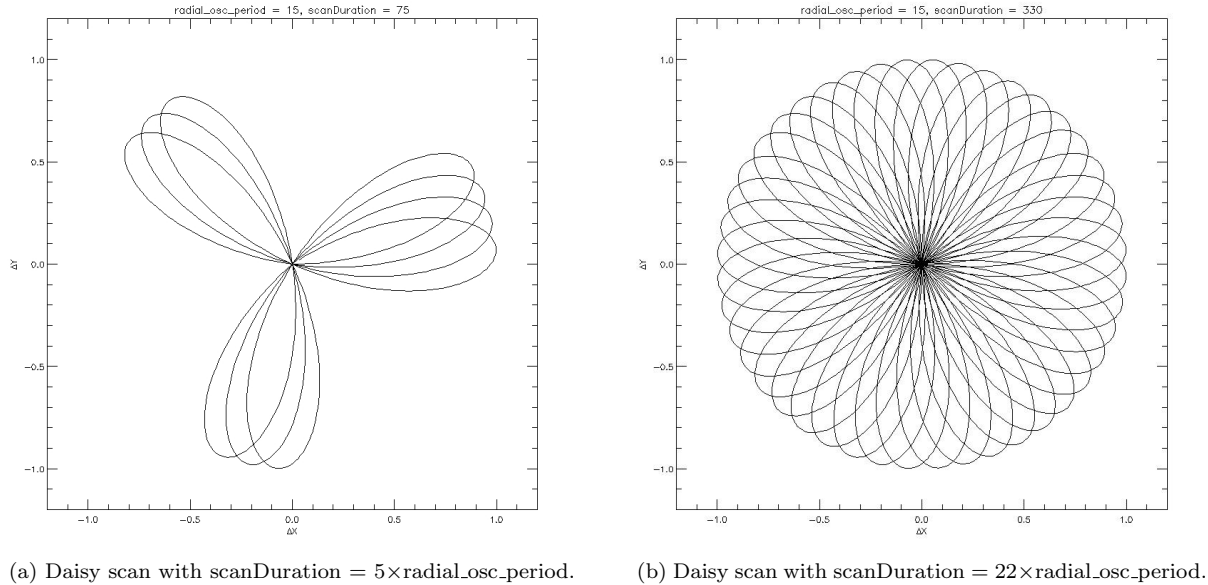


Figure 6.6: Figure 6.6a shows the Daisy trajectory after 5 radial oscillations. Figure 6.6b shows an approximately closed pattern after 22 radial oscillations

**SYNTAX:** `Daisy`( location, map\_radius, radial\_osc\_period, radial\_phase, rotation\_phase, scanDuration, beamName, cos\_v, coordMode, calc\_dt )

**location** A Catalog source name or Location object. It specifies the center of the map.

**map\_radius**  $r_0$  in equations 6.4 and 6.5. A float which specifies the radius of the map’s “daisy petals” in arc-minutes.

**radial\_osc\_period**  $\tau$  in equations 6.4 and 6.5. A float which specifies the period of the radial oscillation in seconds.

–**Note:** not to be less than  $15 \text{ sec} \times \sqrt{r_0/1.5'}$  for radii  $> 1.5'$  and in no case under 15 seconds.

**radial\_phase**  $\phi_1$  in equations 6.4 and 6.5. A float which specifies the radial phase in radians.

**rotation\_phase**  $\phi_2$  in equations 6.4 and 6.5. A float which specifies the rotational phase in radians.

**scanDuration** A float. It specifies the length of the scan in seconds.

**beamName** A string. It specifies the receiver beam to use for both scans. **beamName** can be ‘C’, ‘1’, ‘2’, ‘3’, ‘4’ or any valid combination for the receiver you are using such as ‘MR12’. The default value is ‘1’.

**cos\_v** A Boolean. It specifies whether secant minor corrections (the  $\cos(\hat{y}_0)$  term in equation 6.4) should be used for the major axis of the coordinate system. The default is True.

**coordMode** A string. It specifies the coordinate mode for the radius that generate the map. The default is ‘AzEl’.

**calc\_dt** A float. It specifies time sampling used by the control system to calculate a path. Values should be between 0.1 and 0.5. Calculating many points for a long daisy scan can significantly increase overhead at scan startup. The default is 0.1.



**USAGE:** It takes approximately 22 radial oscillation periods to complete a closed Daisy pattern. However, **radial\_oscillation\_period** is typically set to be in the range of 15–60 seconds depending on the radius being used. As an example, 22 oscillations of 20 seconds would take 440 seconds. If a long trajectory such as this is sent to the antenna manager, intrinsic inefficiencies in the array handling mechanism can significantly increase overhead at the start of a scan. Therefore one should try to keep individual scans to 5 minutes or less.

Script 6.58 will do 22 radial periods over 5 scans lasting 110 seconds each. The **rotation\_phase** and **radial\_phase** arguments are used so that each scan starts where the previous scan finished. This will produce the closed Daisy pattern shown in figure 6.7. The entire SB should take approximately 10 minutes to complete.

```
nosc          = 22.0 # 22 radial oscillations for closed Daisy pattern
map_radius    = 2.8 # arc-minutes
radial_osc_period = 25.0 # seconds
n_scans       = 5 # split 22 oscillations over 5 scans
scanDuration  = nosc * radial_osc_period / n_scans
phi2          = 2.0 * nosc / n_scans
phil          = 3.14159265 * phi2
#NOTE - increment rotation_phase by phi2 each scan
#       - increment radial_phase by phil each scan
for i in range(n_scans):
    Daisy('3C123', map_radius, radial_osc_period, i*phil, i*phi2, scanDuration,
          beamName='1', coordMode='J2000', cos_v=True, calc_dt=0.2)
```

Script 6.58: Daisy() example.

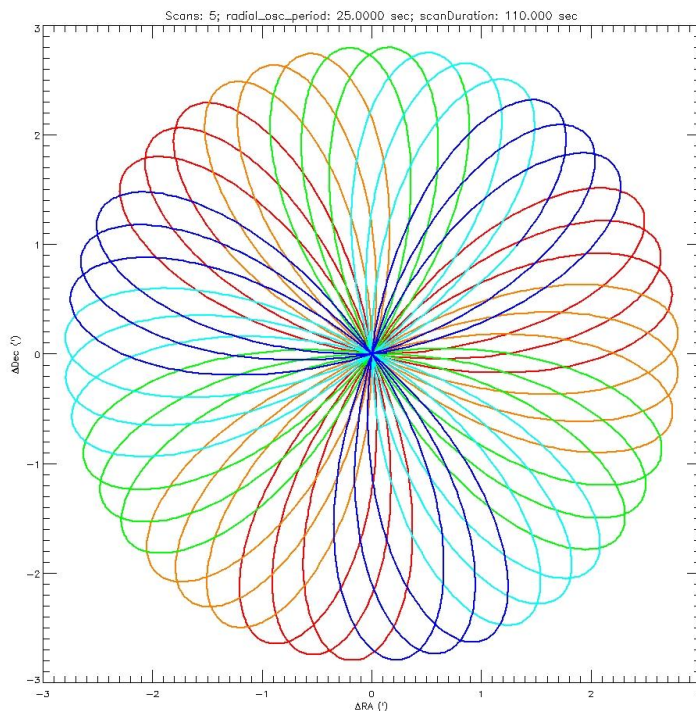


Figure 6.7: A plot of the GBT antenna trajectory executed with script 6.58. Each scan is plotted using a different color.

## 6.5 Utility Functions

Utility functions are used in SBs to control various aspects of the GBT other than data-taking scans. This includes such things as changing power levels, pausing the SB, or waiting for a source to rise. Please note that the syntax for all utility functions is case-sensitive. Advanced utility functions are found in Appendix D.

### 6.5.1 Annotation

The **Annotation()** function allows you to add any keyword and value to the GO FITS file. This could be useful if there is any information you would like to record about your observation for later data processing, or for record keeping. Note that the information in a FITS KEYWORD created via the **Annotation()** function will be ignored by the standard GBT data reduction package GBTIDL.

**SYNTAX:** **Annotation**( KEYWORD, Value )

**KEYWORD** A completely uppercase string of eight characters or less. Do not use any standard FITS keywords.

**Value** A string value for **KEYWORD**.

**USAGE:** An example use of the **Annotation()** function is if you wish to specify what type of source you are observing. Your sources might include H II regions and Planetary Nebulae for example. You could specify each type with

```
Annotation ( 'SRCTYPE', 'HII' )
Annotation ( 'SRCTYPE', 'PNe' )
```

Script 6.59: Annotation() example.

### 6.5.2 Break

The **Break()** function inserts a breakpoint into your SB and gives the observer the choice of continuing or terminating the SB. When a breakpoint is encountered during execution, your SB is paused and a pop-up window is created. The SB remains paused for a set amount of time or until you acknowledge the pop-up window and tell Astrid to continue running your script.

The **Break()** function can take two optional arguments, a message string and a timeout length. Why have a timeout? If an observer walks away from the control room during his or her observing session (e.g. to go to lunch or the bathroom) and a breakpoint is reached, it would be counterproductive to pause the observation indefinitely. This will help save valuable telescope time.

**SYNTAX:** **Break**( message, timeout )

**message** A string. Displayed in the pop-up dialog with a default of “Observation paused”

**timeout** A float. The number of seconds to get user-input before continuing the SB. If you wish for the timeout to last forever then use None. The default is 300 seconds, or 5 minutes.

**USAGE:**

```
Break ( 'This will time out in 5 minutes, the default.' )
Break ( 'This will time out after 10 minutes.', 600 )
Break ( 'This will never time out.', None )
```

Script 6.60: Break() example.

### 6.5.3 Comment

The **Comment()** function allows you to add a comment into the Astrid observing process which will be echoed to the observation log during the observation. What’s the difference between this, and just writing comments with the pound (#) sign in your SB? When you use the pound sign to write your comments, they will not appear in the observation log when your SB is run. Using the **Comment()** function directs your comment to the output in the observation log.

**SYNTAX:** **Comment**( message )

**message** A string. Text to display during the observation.

**USAGE:**

```
# now slew to the source
Comment('Now slewing to 3C 286')
Slew('3C286')
```

Script 6.61: Comment() example.

### 6.5.4 GetUTC

**SYNTAX:** **GetUTC**()

**Return Value:** A float. The current UTC time in decimal hours since midnight.

**WARNING:** If Astrid is in “offline” mode, then **GetUTC()** will return a value of None. Attempting to validate Script 6.62 without checking the return value is not equal to None while “offline” will result in an infinite loop.

**USAGE:** The following example will repeatedly perform **Track** scans until the UTC time is past 12.0 hours.

```
while GetUTC() < 12.0 and GetUTC() != None:
    Track('0353+2234', None, 600.)
```

Script 6.62: GetUTC() example.

### 6.5.5 GetLST

**SYNTAX:** **GetLST**()

**Return Value:** A float. The current Local Sidereal Time in decimal hours.

**WARNING:** If Astrid is in “offline” mode, then **GetLST()** will return a value of None. Attempting to validate Script 6.63 without checking the return value is not equal to None while “offline” will result in an infinite loop.

**USAGE:** The following example will repeatedly perform **Track** scans on the source “1153+1107” until the LST is past 13.5 hours when the source “1712+035” will be observed once.

```
while GetLST() < 13.5 and GetLST() != None:
    Track('1153+1107', None, 600.)
    Track('1712+036', None, 600.)
```

Script 6.63: GetLST() example.

### 6.5.6 Now

#### SYNTAX: Now()

**Return Value:** A UTC time object (see § 6.6.4) containing the UTC time and date.

**WARNING:** If Astrid is in “offline” mode, then **Now()** will return a value of None. Attempting to validate Script 6.64 without checking the return value is not equal to None while “offline” will result in an infinite loop.

**USAGE:** The following example will repeatedly perform **Track** scans on the source “1153+1107” until 09:54:12 UTC on 12 June 2016.

```
while Now() < '2016-06-12 09:54:12' and Now() != None:
    Track('1153+1107', None, 600.)
```

Script 6.64: Now() example.

### 6.5.7 WaitFor

**WaitFor()** pauses the SB until the specified time is reached. The expected wait time is printed in the observation log including a warning if the wait is longer than 10 minutes. **WaitFor()** will immediately return if the specified time has already passed and is within the last 30 minutes. While **WaitFor()** has the SB paused, it does not prevent the user from aborting. However if the user chooses to continue once the abort is detected, then the **WaitFor()** abandons the wait and returns immediately.

#### SYNTAX: WaitFor( Time\_object )

**Time\_object** A valid time object (see § 6.6.4).

- **Note:** If a value of None is used as an argument to **WaitFor()**, the SB will abort with a message to the observation log. This can occur when passing a value from `Horizon().GetRise()` or `Horizon().GetSet()` when such an event may never occur, such as the rise time for a circumpolar source.

**USAGE:** The following example will pause the SB until a Local Sidereal Time of 15:13, then wait for the source “1532\_3421” to rise above 10° elevation, and finally wait for the Sun to set below 5° elevation.

```
#Wait for 15:13 LST
WaitFor('15:13:00 LST')

#Wait until source is above 10 deg elevation
WaitFor(Horizon(10.0).GetRise('1532+3421'))

#Wait for the Sun to set below 5 deg elevation
WaitFor(Horizon(5.0).GetSet('Sun'))
```

Script 6.65: WaitFor() example.

## 6.5.8 ChangeAttenuation

**ChangeAttenuation()** allows the observer to change all the attenuators in the IF Rack or the Converter Rack by the same amount.

**SYNTAX:** **ChangeAttenuation**( devicename, attnchange )

**devicename** A string that can be either 'IFRack' or 'ConverterRack'. This specifies the device in which the attenuators will be changed.

**attnchange** A float. This specifies how much the attenuators should be changed. This value can be either positive or negative.

- **Note:** if any new attenuator setting is less than zero or exceeds the maximum value, 31 for the IF Rack and 31.875 for the Converter Rack, then the attenuator settings is made to be the appropriate limiting value.

**USAGE:** The Following example adds 1 to the attenuation value in the IF rack and subtracts 0.5 from the attenuation value in the converter rack.

```
ChangeAttenuation('IFRack', 1.0)
ChangeAttenuation('ConverterRack', -0.5)
```

Script 6.66: ChangeAttenuation() example.

## 6.6 Scheduling Block Objects

Scheduling Block Objects are Python objects that are used to contain multiple pieces of information within a single variable. These are used with positions (requiring a major and minor axis value along with an epoch), times (requiring the date and the time of day), and for defining a horizon for the minimum elevation below which you would not want to observe.

### 6.6.1 Location Object

A Location object is used to represent a particular location on the sky.

**SYNTAX:** `Location( coordinateMode, value1, value2 )`

**coordinateMode** A string. The following modes are allowed: 'J2000', 'B1950', 'RaDecOfDate', 'HaDec', 'ApparentRaDec', 'Galactic', 'AzEl', and 'Encoder'

**value1, value2** May be a float, or sexagesimal quoted as a string (i.e. 'hh:mm:ss.s'). A location must be specified by these two values, the meanings of which are dependent on the both the chosen coordinate mode and value type of each unit:

- **float values:** will always denote units in degrees of arc, regardless of the coordinate mode.
  - This should not be confused with decimal use in Catalogs (see § 6.3) which denote decimal hours for RA and HA, and degrees of arc for all other angles.
- **sexagesimal value1:** Represent units of time for J2000, B1950, ApparentRaDec, and RaDecOfDate and degrees of arc for HaDec, Galactic, AzEl and Encoder.
- **sexagesimal value2:** Represent degrees of arc.

**USAGE:**

```
# RA is in units of *time*, Dec is in degrees
location = Location('J2000', '16:30:00', '47:15:00')

# Same location - RA is in degrees, Dec is in degrees
location = Location('J2000', 247.5, 47.25)

# Az is in degrees, El is in degrees
location = Location('AzEl', '45:00:00', '72:30:00')
```

Script 6.67: Specifying Location Objects.

## 6.6.2 Offset Object

An Offset is a displacement from the position of a source or from the center position of a map. Offset objects may be added to other offset objects with the same coordinate mode and cosv correction. Offset objects may be added to Location objects with the same coordinate mode. **Note that such addition is not commutative and must be of the form (Location+Offset).** **Offset+Location** will produce a validation error.

**SYNTAX:** `Offset( coordinateMode, value1, value2, cosv )`

**coordinateMode** A string. The following modes are allowed: 'J2000', 'B1950', 'RaDecOfDate', 'HaDec', 'ApparentRaDec', 'Galactic', 'AzEl', and 'Encoder'

**value1, value2** May be a float, or sexagesimal quoted as a string (i.e. 'hh:mm:ss.s'). An offset must be specified by these two values, the meanings of which are dependent on the both the chosen coordinate mode and value type of each unit:

- **float values:** will always denote units in degrees of arc, regardless of the coordinate mode.
  - This should not be confused with decimal use in Catalogs (see § 6.3) which denote decimal hours for RA and HA, and degrees of arc for all other angles.
- **sexagesimal value1:** Represent units of time for J2000, B1950, ApparentRaDec, and RaDecOfDate and degrees of arc for HaDec, Galactic, AzEl and Encoder.
- **sexagesimal value2:** Represent degrees of arc.

**cosv** A Boolean. It specifies whether secant minor corrections in equation 6.7 should be used for the major axis of the coordinate system (i.e.  $h/\cos(v)$  is the offset value in the direction of  $h$ ). The default is True. Since coordinate distances and angular separations are not equivalent for sperical coordinate systems, the following approximations may be used for small separations:

$$\Delta v = v_1 - v_2 \tag{6.6}$$

$$\Delta h = (h_1 - h_2) \cdot \cos(v) \tag{6.7}$$

where  $h$  is the value of the major coordinate axis and  $v$  is the value of the minor coordinate axis. For example, setting `cosv=True` with J2000 coordinate offsets will apply a  $\cos(Dec)$  term from equation 6.7 to make maps appear rectangular if plotted with  $\Delta RA$  vs.  $\Delta Dec$  relative to a central location.

**USAGE:** Script 6.68 gives examples of adding Offset objects to Location and other Offset objects. The resulting coordinates are printed to screen.

```
start_location = Location('J2000', '12:00:00', '45:00:00')
offset1        = Offset('J2000', '00:04:00', '01:00:00', cosv=False)
offset2        = Offset('J2000', 2.0          , 2.0          , cosv=False)
offset3        = offset1 + offset2

loc1 = start_location + offset1 #loc1 (RA,Dec) = (12:04:00, 45:00:00)
loc2 = start_location - offset2 #loc2 (RA,Dec) = (11:52:00, 43:00:00)
loc3 = start_location + offset3 #loc3 (RA,Dec) = (12:12:00, 48:00:00)
print 'RA,Dec of loc3 = (%s,%s)'%(loc3.GetH(),loc3.GetV())
```

Script 6.68: Adding Offset objects to Location objects and other Offset objects.

### 6.6.3 Horizon Object

Observing Scripts allow an observer to specify a definition of the horizon. The user defined horizon can be used to begin an observation when an object “rises” and/or end the observation when it “sets” relative to the specified elevation of the “horizon”. The Horizon object may be used to obtain the initial time that a given source is above the specified horizon (including an approximate atmospheric refraction correction).

#### SYNTAX:

**Horizon**( elevation )

#### FUNCTIONS:

**Horizon**( elevation ).**GetRise**( location )

**Horizon**( elevation ).**GetSet**( location )

**location** A Catalog source name or Location object using a spherical coordinate mode. **Horizon**() will not work with planets and ephemeris tables.

**elevation** A float. The Horizon elevation in degrees. The default is 5.25 (the nominal GBT horizon limit).

**Return Value:** A UTC time object (see § 6.6.4) containing the UTC time and date.

- **GetRise**(source) will return the most recent rise time if the source is currently above the horizon, or the next rise time if the source has not yet risen. **GetRise**(source) will return **None** if the source never rises and the current time if the source never sets.
- **GetSet**(source) will return the next set time of the source. **GetSet**(source) will return **None** if the source never sets and the current time if the source never rises.

**USAGE:** Any Horizon object may be substituted as a start or stop time in scan types, such as **Track**(). Script 6.69 will display the time when VirgoA rises above 20° elevation. Depending on the position of the source at the time of execution, the SB would then either begin a **Track**() scan immediately or wait for VirgoA to rise above 5.25° elevation before beginning the scan. In both cases, the SB would terminate the next time VirgoA sets below 5.25° elevation.

```
print Horizon(20.0).GetRise('VirgoA')

h = Horizon() #default horizon of 5.25 degrees elevation
Track('VirgoA', None, startTime=h, stopTime=h)
```

Script 6.69: Using Horizon Objects.



### 6.6.4 Time Object

The Time Object is primarily used for defining scan start or stop times. The time may be represented as either a sexagesimal string or in a python `mxDateTime` object. You can learn more about `mxDateTime` at <http://www.egenix.com/files/python/mxDateTime.html><sup>3</sup>.

#### SYNTAX:

The Time Object can be expressed in either UTC or LST. The time can be either absolute or relative. An absolute or dated time specifies both the time of day and the date. An absolute time may be represented by either a sexagesimal string, i.e., “yyyy-mm-dd hh:mm:ss” or by a `DateTime` object. Relative or dateless times are specified by the time of day for “today”. “WaitFor” will treat a dateless time that is more than 30 minutes in the past as being in the future, i.e., the next day. Relative times may be represented by either a sexagesimal string, i.e., “hh:mm:ss” or a `DateTimeDelta` object.

For UTC times, the sexagesimal representation may include a “UTC” suffix. Note that `mxDateTime` objects are always UTC. LST time may only be used with relative times and the sexagesimal representation must include a “LST” suffix.

Time Objects can have slightly varying formats and can be created in a few different ways. Some examples are:

“2006-03-22 15:34:10” Absolute time in UTC represented by a string.

`DateTime.TimeDelta(12, 0, 0)` Relative time in UTC as a `mxDateTime` object.

“2006/03/22 15:34:10 UTC” Absolute time in UTC represented by a string.

“22:15:48 LST” Relative time in LST as a string.

`DateTime.DateTime(2006, 1, 21, 3, 45, 0)` Absolute time in UTC as a `mxDateTime` object.

**USAGE:** In this example we will continue to do one minute observations of `srcA` until Feb 12, 2007 at 13:15 UTC when we will then do a ten minute observations of `srcB`.

```

from mx import DateTime

switchTime=DateTime.DateTime(2016,2,12,13,15,0) # Feb 12, 2016, 13:15 UTC

while Now() < switchTime and Now() != None:
    Track(srcA, None, 60)

Track(srcB, None, 600)

```

Script 6.70: Using Time Objects.

<sup>3</sup>Note, one must access the python `DateTime` module directly from an observation script to generate time objects, i.e., using `mx import DateTime`.

## 6.7 Example Scheduling Blocks

For the following SB examples we will use the configuration examples from § 6.2. All configurations, catalogs and scripts are available within the Green Bank computing environment at

`/home/astro-util/projects/GBTog/`

The following catalog (`sources.cat`) will be used for all examples:

```
# My source list with radial velocities
format=spherical
coordmode = B1950
head = name      ra          dec          velocity
Object1 09:56:16.98 +49:16:25.5 27.23
Object2 08:56:16.98 +48:16:25.5 28.24
Object3 07:56:16.98 +47:16:25.5 29.25
Object4 06:56:16.98 +45:16:25.5 30.26
```

Script 6.71: The `sources.cat` catalog used for the SB examples in this section.

### 6.7.1 Frequency Switched Observations Looping Through a List of Sources

In this example we perform frequency switched observations of the HI 21 cm line towards several different sources.

This example is available as `/home/astro-util/projects/GBTog/SBs/example1.py`.

```
# Frequency Switched Observations where we loop through a list of sources

# first we load the configuration file
execfile('/home/astro-util/projects/GBTog/configs/fs_config.py')

# now we load the catalog file
c = Catalog('/home/astro-util/projects/GBTog/cats/sources.cat')

# now we configure the GBT IF system for frequency switch HI observations
Configure(fs_config)

# get the list of sources
sourcenames = c.keys()

# now loop the sources
for src in sourcenames:
    Slew(src)           # Slew to each source
    Balance()          # Balance power levels
    Track(src, None, 600.) # Observe each source for 10 minutes
```

Script 6.72: SB Example 1 – Frequency switched observations looping through a list of sources.

## 6.7.2 Position Switched Observations Repeatedly Observing the Same Source

In this example we perform position switched observations of a single source. We observe the source for two minutes and the off position for two minutes. This is repeated twenty times.

This example is available as `/home/astro-util/projects/GBTog/SBs/example2.py`.

```
# Position Switched Observations to repeatedly observe the same source

# load the configuration file
execfile('/home/astro-util/projects/GBTog/configs/tp_config.py')

# load the catalog file
Catalog('/home/astro-util/projects/GBTog/cats/sources.cat')

# configure the GBT IF system for frequency switch HI observations
Configure(tp_config)

# specify which source we wish to observe
src = 'Object1'

# specify how far away from the source the off position should be
# offset two minutes of time in Right Ascension
myoff=Offset("J2000", "00:02:00", 0.0)

#Slew to the source and then balance the power levels
Slew(src)
Balance()
# now we use a Break() so that we can check the IF system
Break('Check the Balance of the IF system')

# specify how many times to observe the source
numobs = 20

# observe 'on' source for 2 minutes and 'off' source for 2 minutes
# and then repeat
for i in range(numobs):
    OnOff(src, myoff, 120.)
```

Script 6.73: SB Example 2 – Position switched observations repeatedly observing the same source.

### 6.7.3 Position Switched Observations of Several Sources and Using the Horizon Object

In this example we perform position switched observations of three sources. We observe the first source until the second source rises above 20° elevation. Then we observe the second source until it goes below 20° elevation at which point we observe a third source.

This example is available as `/home/astro-util/projects/GBTog/SBs/example3.py`.

```
# Load the configuration file and the catalog file
execfile('/home/astro-util/projects/GBTog/configs/tp_config.py')
Catalog('/home/astro-util/projects/GBTog/cats/sources.cat')

# now we configure the GBT IF system for position switched observations
Configure(tp_config)

# define which sources to observe
srcA = 'Object4'
srcB = 'Object3'
srcC = 'Object1'
myoff=Offset('J2000','00:02:00',0.0) # Off position of 2min time in RA
h=Horizon(20.0) # specify a horizon of 20 degrees elevation

riseSrcB = h.GetRise(srcB) # now get rise and set times of srcB
setSrcB = h.GetSet(srcB)

#print the rise and set times of srcB
risesetstring='20 deg elev. rise = %s and set = %s'%(riseSrcB,setSrcB)
Comment(risesetstring)

# observe srcA until srcB has risen above 20 deg elevation
Slew(srcA)
Balance()
while Now() < riseSrcB and Now() != None:
    OnOff(srcA,myoff,120.)

# now observe srcB until it sets
Slew(srcB)
Balance()
while Now() < setSrcB and Now() != None:
    OnOff(srcB,myoff,120.)

# now observe srcC five times
numobs=5
Slew(srcC)
Balance()
for i in range(numobs):
    OnOff(srcC,myoff,120.)
```

Script 6.74: SB Example 3 – Position switched observations of several sources and using the Horizon object.

### 6.7.4 Frequency Switched On-The-Fly Mapping

In this example we perform frequency switched observations of the HI 21 cm line to map a  $5 \times 5$  degree region of the sky. We use pixels that are  $3'$  in size and have an integration time of 2 seconds per pixel. We do not observe the whole map in this example.

This example is available as `/home/astro-util/projects/GBTog/SBs/example4.py`.

```
# Frequency Switched Observations where we loop through a list of sources

# Load the configuration file
execfile('/home/astro-util/projects/GBTog/configs/fs_config.py')

# Load the catalog file
Catalog('/home/astro-util/projects/GBTog/cats/sources.cat')

# now we configure the GBT IF system for freq switched HI observations
Configure(fs_config)

# now we set the parameters for the map
src      = 'Object2'          # location of the map center
majorSize = Offset('Galactic',5.0,0.0) # 5 degrees in galactic longitude
minorSize = Offset('Galactic',0.0,5.0) # 5 degrees in galactic latitude
rowStep  = Offset('Galactic',0.0,0.05) # 3 arcminutes between map rows

# the time to scan each row
# time = majorSize / rowStep * integration time per pixel
scanTime = 5.0/0.05*2. # 2 seconds per pixel

# Balance power levels
Slew(src)
Balance()
Break('Check power levels')

# only do part of the map here
rowStart = 10
rowStop  = 20

# now observe for the map
RALongMap(src,majorSize,minorSize,rowStep,scanTime,
           start=rowStart,stop=rowStop)
```

Script 6.75: SB Example 3 – Frequency switched On-The-Fly mapping.

## 6.8 What Makes a Good Scheduling Block

Rarely does an observing session exactly follow one's plans. A useful philosophy is to consider the work that would be involved in editing an SB if something were to go wrong during its execution and you wanted to resume its execution where you left off. You should break apart any long scripts into smaller individual scripts to reduce the need for edits.

During your observing, you will make decisions as to how to proceed with the next observations. You should break apart large scripts to increase your flexibility in being able to react to the circumstances that arise during your observing.

We recommend that the following should be avoided within a single SB, as it will make the block too long:

- **Multiple Configurations:** Multiple configurations (peak/focus and science observations should optimally be performed with separate SBs.
- **Changing Receivers:** You should only use a single receiver within an SB.
- **Multiple Maps:** You should perform only a single map within any SB.

# Chapter 7

## Observing Tactics and Recommendations

### 7.1 Active Surface (AS) Strategies

If you are observing at a frequency of 8 GHz or higher then you should use the Active Surface (AS). At frequencies below 8 GHz the AS does not provide any improvements to the efficiency of the GBT. Due to RFI considerations the AS may be turned off for lower frequency observations.

**You do not need to do anything to turn on or off the use of the AS. The GBT telescope operator performs these tasks.**

### 7.2 AutoOOF Strategy

AutoOOF is recommended for observing at frequencies of 26 GHz and higher and only available for use with Rcvr26\_40 (Ka-band), Rcvr40\_52 (Q-band, Rcvr68\_92 (W-band, and Rcvr\_PAR (MUSTANG). For a description of this procedure, refer to Section 6.4.1.4. For the associated data display, see Section 5.1.4. It is important to note the following points when using AutoOOF:

**Choose a bright calibrator:** preferably at least 7 K in the observed band, which is about 4 Jy at Q-band. You should not rely on the catalog flux to be accurate as it is often many years out of date. The ALMA Calibrator Source Catalogue has an extensive record of the flux densities for many of the bright 3mm sources (<https://almascience.eso.org/sc/>). If you are not sure then run a point/focus scan on the calibrator first in order to confirm its strength. Remember, you need to be able to detect the source when the subreflector is  $\pm 5\lambda$  out of focus which typically reduces its peak intensity by a factor of 8.

**Allow approximately 20 minutes for an AutoOOF:** The AutoOOF procedure will obtain three OTF maps, each taken at a different focus position. Each map will require nearly 1 minute of initial M&C overhead and last 316 seconds with Rcvr26\_40, 300 seconds with Rcvr40\_52, and 366 seconds with Rcvr68\_92. Processing will then require 1 minute for a total of 19–22 minutes.

**Use AutoOOF to derive pointing and focus offsets:** The processing is launched automatically upon completion of the third map, and the result is displayed in the OOF plug-in tab of Astrid. It is incumbent upon the user to examine the solutions, and click the button (in the Astrid DataDisplay tab) to send the selected solution to the active surface. It is recommended that when sending the solutions, you use the yellow button in the OOF display tab labeled “Send Selected Solution with Point and Focus Corrections”. By using this method, it is no longer necessary to follow AutoOOF with another AutoPeakFocus.

**Running AutoOOF first:** If you plan to run AutoOOF as the first thing during your observing session, we recommend running an AutoPeakFocus before the AutoOOF for all receivers except Rcvr68\_92 (See next point). Subsequent runs of AutoOOF will not need a pre-point/focus as small errors in these values do not harm the results.

**AutoOOF with the W-band (68–92 GHz) receiver:**

- **Use AutoOOF before AutoPeakFocus:** Blind pointing offsets at W-band (68–92 GHz) frequencies are similar to beam sizes. The source may be missed in the simple Az–El scans used by the Peak procedure. In These cases, AutoOOF should be run before AutoPeakFocus.
- **AutoOOF is not necessary for daytime observations of extended sources:** While beam sizes can vary significantly during the day (10–14”), this has little effect on main-beam efficiency used for the calibration of extended sources. Therefore, extended sources may be observed during the day without the AutoOOF corrections if the science is not impacted by the primary beam variations.

**AutoPeakFocus after AutoOOF:** AutoPeakFocus may be run as a sanity check on the AutoOOF solution. If Peak/Focus scans were performed before AutoOOF, then source amplitude should be greater than what was seen before the surface correction was sent. Additionally, AutoPeakFocus pointing and focus corrections should agree with values derived by AutoOOF.

## 7.2.1 How long does the solution remain valid?

- **Nighttime:** If the corrections are measured at least an hour after sunset, then they should last for the next few hours as the backup structure cools off. This can take many hours if it was a sunny day. At frequencies below 90 GHz, turning off the corrections sometime between midnight and 3AM may improve the surface if a sidelobe begins to appear on bright pointing sources.
- **Daytime:** During the daytime, this is a difficult question to answer, as it depends on how much the pose of the telescope is changing with respect to the Sun, cloud cover changes, etc. The answer can be anything from 1-4 hours. In practice, we suggest running an AutoPeak every 30-40 minutes to look for characteristics outlined below.
- **Periodically Examine Peak Scans:** A new AutoOOF may be necessary if the following characteristics are seen:
  - Significant sidelobes begin to appear.
  - The beam size increases by more than 10%.
  - Source amplitude decreases systematically by 15% or more.

More information on AutoOOF can be found at <https://safe.nrao.edu/wiki/bin/view/GB/PTCS/AutoOOFInstructions>



## 7.3 Balancing Strategies

The GBT IF system has many ways to add gain and/or attenuation in the IF path, depending upon the desired configuration. Before taking data with the GBT, the observer must ensure that all components along the IF path have optimum input power levels; this process is referred to as “balancing”. This will ensure for example that no components saturate and that amplifiers are in the most linear part of their dynamic range. The system automatically adjusts power levels to optimum values when you issue the **Balance()** command in Astrid. The following discussion gives guidelines for when and how often to use the **Balance()** command.

Strategies for balancing the IF power levels depend upon the backend, the observing frequency, the observing tactics, the weather and the objects being observed. The DCR has a dynamic range of about  $10^1$  in its ability to handle changes in the brightness of the sky as seen by the GBT. The sky brightness can change because of continuum emission of a source or a maser line as you move on and off the source. It can also change due to changes in the atmosphere’s contribution to the system temperature as the elevation of the observations change or due to the weather.

Whenever you **Balance()** you almost always change the variable attenuator. Each attenuator setting has a unique bandpass shape. So if you change attenuators then you will likely see changes in the bandpasses and baselines of the raw data.

There are not any set-in-stone rules for when an observer should balance the GBT IF system. However there are some guidelines which will allow you to determine when you should balance the IF system. Here are the guidelines:

1. You should balance the IF system after performing a configuration.
2. You should minimize the number of times you balance when observing.
3. If you know  $T_{\text{sys}} + T_{\text{src}}$  will change by more than a factor of two ( $3 \text{ dB}^2$ ) when you change sources (not between and on and off observation) you should consider balancing.
4. Try to avoid balancing while making maps.
5. Never balance between “signal” and “reference” observations (such as during an on/off observation).
6. If you are observing target sources and calibration sources then try not to balance between observations of the targets and calibrators.

### Note:

- If during your observing you expect to see a change in power levels on the sky that are roughly equivalent to the GBT system temperature, then you should contact your GBT support person to discuss balancing strategies. There are no global solutions and each specific case must be treated independently.
- If the system temperature between “signal” and “reference” observations differ by a factor of two or more, **BalanceOnOff()** (see § 6.4.1.10) should be used in place of **Balance()**. This will balance the IF power levels at the midpoint of the two power levels at each sky position.

---

<sup>1</sup>From about 0.5 to 5 Volts of IF power in the IF Rack.

<sup>2</sup>A change in power from  $P_1$  to  $P_2$  can be represented in dB by  $10 \log_{10}(P_1/P_2)$

### 7.3.1 Balancing VEGAS

There are two aspects to VEGAS balancing that will produce separate error messages, so the user should check the origin carefully:

1. **Adjusting IF power levels upstream of the VEGAS ADC:** Power levels upstream of the VEGAS ADCs are balanced so that the power going into an ADC is at an acceptable level.
  - **Balancing will fail if the input IF power levels to VEGAS is more than  $\pm 2$  dB from the target value of -20 dBFS.** VEGAS has a much higher range than that, but it is extremely rare that the observation/equipment combination should prevent the balancing algorithm from meeting the target. A conservative limit was chosen since a failure normally means that some part of the system is not configured correctly, or that there is hardware failure. If an IF balancing failure occurs, the user (or operator) should look for errors in the IF system and can view the actual IF power levels in the VEGAS CLEO screen (see § 5.3.2). If they are significantly different from -20 dBFS, there is a problem somewhere in the IF chain.
  - **If the power levels are different from -20 dBFS, but close to it, there may not be a real problem.** In some cases, the IF balancing will fail due to an exceptional, but acceptable circumstance; for example, looking at an extremely bright source, or using a spectral window close to the edge of the receiver passband. The IF balancing failure does not cause an abort, and it is often acceptable to continue observing under these circumstances.
 

The useable dynamic range of VEGAS is actually  $> 20$  dB. It is set at a low level by quantization effects, and at high levels by saturation. If the IF power level looks reasonable, the next check is to look at the ADC histogram counts. As long as the histogram looks like a Gaussian distribution, with a FWHM around 20 counts or larger, but with no counts approaching  $\pm 127$ , then the IF level into VEGAS is acceptable (see Figure 5.21). Make sure you monitor the ADC histogram through all phases of your observation (e.g. switching on and off a bright source).
2. **Adjusting the “digital gain” inside the VEGAS processing firmware:** There should be no circumstances (e.g. an FFT overflow) which result in lost precision.
  - **The digital gain should never fail to balance.** It is a property of the firmware design of each mode, not the IF input. A failure of the digital gain balancing indicates a serious problem, and engineering support should be called.

## 7.4 Calibration Strategies

For best flux density calibration of spectra, it is recommended that you should observe continuum flux density calibration sources at least once during an observing session. To do this, do a Peak/Focus on the calibrator, followed by an observation in the same spectral line setup used for the program sources. This will give the relation of flux density to antenna temperature as a function of frequency that can be applied to the program spectra.

Of course, there may be other outstanding reasons to perform calibration observations more often. If you have concerns over how often you should observe a calibrator you should get into contact with your GBT support person.

Flux density histories for many of the bright 3mm point sources can be found in the ALMA Calibrator Source Catalogue (<https://almascience.eso.org/sc/>).

## 7.5 Pointing and Focusing Strategies

How often you need to point and focus the GBT depends on the frequency of your observations, the weather conditions, whether or not it is day or night-time, and the amount of flux error that your experiment can tolerate from pointing and focus errors. See Table 7.1 for guidelines on how often to Point/Focus. Note that spacings between Point/Focus observations may be increased if results appear stable, especially during the night.

Within the DSS, the tracking error  $\sigma_{tr}$  (arc seconds) as a function of wind speed  $s$  ( $ms^{-1}$ ) is given by

$$\sigma_{tr}^2 = \sigma_0^2 + \left(\frac{s}{3.5}\right)^4 \quad (7.1)$$

Where  $\sigma_0 = 1.32''$  at night and  $\sigma_0 = 2.19''$  during the day, and is the tracking and pointing error with no winds. The DSS will only schedule observations if the tracking error is smaller than a specified fraction ( $f < f_{max}$ ) of the beam FWHM ( $\sigma_{beam}$ ) given by

$$f = \frac{\sigma_{tr}}{\sigma_{beam}} = \frac{\sigma_{tr}\nu}{748} \quad (7.2)$$

where  $\nu$  is the observing frequency in GHz. Values for  $f_{max}$  in the DSS are currently set at 0.2 for receivers below 50 GHz, 0.22 for receivers above 50 GHz (Rcvr68\_92) and 0.4 for filled arrays (Rcvr\_PAR). An  $f_{max}$  value of 0.2 assures observers that their flux uncertainty due to tracking errors is no more than 10%, assuming they are observing a point source.

Table 7.1: Observing wind limits using DSS default parameters and suggested time periods between Point and Focus observations.

Receiver	$\nu$ (GHz)	Wind limit ( $ms^{-1}$ )		Recommended Point/Focus spacing	
		Day	Night	Day	Night
Rcvr_342	0.340	73.4	73.4	— Initial Peak only —	
Rcvr_450	0.415	66.5	66.5	— Initial Peak only —	
Rcvr_600	0.680	52.0	52.0	— Initial Peak only —	
Rcvr_800	0.770	48.8	48.8	— Initial Peak only —	
Rcvr_1070	0.970	43.5	43.5	— Initial Peak only —	
Rcvr1_2	1.4	36.2	36.2	— Initial Peak and Focus only —	
Rcvr2_3	2.0	30.3	30.3	— Initial Peak and Focus only —	
Rcvr4_6	5.0	19.1	19.1	Hourly on hot afternoons	Every 2-3 hours
Rcvr8_10	10.0	13.5	13.5	Hourly on hot afternoons	Every 2-3 hours
Rcvr12_18	15.0	11.0	11.0	Hourly	Every 1-2 hours
RcvrArray18_26	25.0	8.3	8.5	Hourly	Every 1-2 hours
Rcvr26_40	32.0	7.1	7.4	Hourly	Hourly
Rcvr40_52	45.0	5.5	6.1	Every 30-60 minutes	Hourly
Rcvr68_92	80.0	—	4.4	Every 30-60 minutes	Every 30-60 minutes
Rcvr_PAR	90.0	5.5	6.1	Every 30-60 minutes	Every 30-60 minutes

Table 7.1 provides wind limits using default DSS parameters. Observers may wish to alter some parameters in the DSS to better suit their observing requirements. For example, pointing may be relaxed for extended sources (i.e. set  $\theta_{src} > 0$  in the DSS), or more tightly constrained (a value of  $f_{max} = 0.14$  in the DSS assures no more than 5% flux uncertainty due to tracking errors). You may request changes to DSS control parameters by contacting your GBT “friend” and emailing helpdesk-gb@nrao.edu.

For further information on DSS control parameters see Appendix G. See DSS Project Note 18.1 ([https://safe.nrao.edu/wiki/pub/GB/Dynamic/DynamicProjectNotes/dss\\_20nov2014.pdf](https://safe.nrao.edu/wiki/pub/GB/Dynamic/DynamicProjectNotes/dss_20nov2014.pdf)) for tracking performance and parameters used in Equation 7.1.

## 7.6 Spectral Configurations

It is good practice for the observer to schedule a short observation towards a strong spectral line source at the beginning of each observing session. The observer should process and check the spectra before proceeding with the observations to confirm the spectral configuration. It is also important to check the system temperatures for each beam. Poor configuration choices for the backend will sometimes be detected as anomalous system temperatures.

## 7.7 Mapping Strategies

The sky should be sampled five times across the beam (i.e., the scan rate and integration times should be set such that there are five integrations recorded in the time the telescope scans across an angle equal to the beam FWHM) and there should be four switching periods per integration (see Mangum, Emerson, and Greisen 2007 A&A, 474, 679 for details). For VEGAS, minimum recommended integration times and switching periods are listed in Table 8.1 and § 8.4. The DCR has a minimum integration time of 100 ms.

The GBT Mapping Calculator is a useful tool for planning mapping observations and may be used to provide Astrid commands and parameters for many of the mapping scan types. The mapping calculator can be found at <http://www.gb.nrao.edu/~rmaddale/GBT/GBTMappingCalculator.html>.

It is important to take account of the overheads involved with mapping scans. For example, it takes approximately 25 seconds to start a **RALongMap** mapping scan. So observers scheduling scans much shorter than 1 minute will lose a large fraction of their observing time to overheads. Using Daisy pattern scans are more efficient for scheduling small maps, as a region can be mapped in a single scan. However there is an additional delay in starting a **Daisy** procedure, as the system computes the antenna trajectories. As a rule of thumb, maps larger than about 10–20 beam FWHM in size should use the **RALongMap** or **DeclatMap** scan types while smaller maps should use **Daisy** scans. For practical reasons, it is often best to keep the scan length under about 15 minutes.

Observers wishing to use the GBT mapping pipeline may periodically wish to include reference scans into their Scheduling Block. A Python script used to perform astronomical observations with the GBTs. Further information on using the pipeline can be found at <https://safe.nrao.edu/wiki/bin/view/GB/Gbtpipeline/PipelineRelease>.

## 7.8 High Frequency Observing Strategies

When observing at frequencies above 10 GHz you should be aware that additional calibration measurements may be necessary. The telescope efficiency can become elevation dependent, atmospheric opacities are important and the opacities can be time variable. You should contact your GBT support person to discuss these issues.

All the GBT high frequency receivers have at least two beams (pixels) on the sky. You should make use of both of these during your observations if possible. For example, if you are doing position switched observations and your source is not extended then you can use the Astrid **Nod()** procedure to observe (see § 6.4.2).

## 7.9 Observing Strategies For Strong Continuum Sources

Spectral line observations of strong continuum sources leads to a great amount of structure (i.e. ripples) in the observed spectra. So observations of strong continuum sources requires careful consideration of the observing setup and the techniques used.

If you are trying to observe weak broad spectral lines (wider than  $\sim 100$  MHz) toward a source with strong continuum emission (more than 1/10th the system temperature), then you should consider using double position switching. This technique is discussed in an Arecibo memo by Tapasi Ghosh and Chris Salter which can be found at <http://www.naic.edu/astro/aotms/performance/2001-02.ps>.

Another issue is finding a proper IF balance that allows both the “on” and “off” source positions to remain in the linear range of the backend being used. This means that one must find the IF balance in both the “on” and “off” position and then split the difference – assuming that the difference in power levels between the “on” and “off” do not exceed the dynamic range of the backend. The **BalanceOnOff()** procedure in Astrid can be used to accomplish this type of balancing (see § 6.4.1).



# Chapter 8

## VEGAS

### 8.1 Overview

The The VErSatile GBT Astronomical Spectrometer (VEGAS) is an FPGA based spectrometer that can be used with any receiver except MUSTANG. It consists of eight independent spectrometers (banks) that can be used simultaneously. Eight-bit samplers and polyphase filter banks are used to digitize and generate the spectra – together they provide superior spectral dynamic range and RFI resistance. For details on the design of VEGAS, please consult <http://www.gb.nrao.edu/vegass/report/URSI2011.pdf>.

Observers can use between one and eight dual-polarization spectrometers (or “banks”) at the same time (see Fig. 8.1). Each bank within VEGAS can be configured with a different spectral resolution, bandwidth, and number of spectral windows (subbands). However, the integration time, switching period, and the frequency switch **must** be the same for all banks. The resolution and bandwidth of all subbands in a single VEGAS bank must be identical, but the center frequencies may be set independently (within limits).

Although the individual banks could be arranged to cover 10 GHz of total bandwidth, the maximum bandwidth is typically limited to 4-6 GHz by filters in the GBT IF system (see your project friend for more information). All banks have the same switching signal (i.e., same switching period, same integration time, same frequency switch), which is controlled by spectrometer bank “A”. Each bank can be configured in one of the 29 modes listed in Table 8.1.

- **Modes 1-19** provide a single subband per bank. Modes 1–3 have the following constraints on useable bandwidth:
  - **Modes 1-2:** Have a useable bandwidth of 1250 MHz within the baseband bandwidth of 1500 MHz. The useable baseband frequency range is 150–1400 MHz.
  - **Mode 3:** Has a useable bandwidth of 800 MHz within the baseband bandwidth of 1080 MHz. The useable baseband frequency range is 150–950 MHz.
- **Modes 20-29** provide up to eight subbands per bank. All subbands must have equal bandwidths and be placed within the total bandwidth processed by that bank:
  - **Modes 20-24:** Have a useable bandwidth of 1250 MHz within the baseband bandwidth of 1500 MHz. The useable baseband frequency range is 150–1400 MHz.
  - **Modes 25-29:** Have a useable bandwidth of 800 MHz within the baseband bandwidth of 1080 MHz. The useable baseband frequency range is 150–950 MHz.

Each mode provides the polarization products XX, YY, and optionally XY, YX necessary for observations of polarized emission without requiring a reduction in the number of channels or sampling speed. VEGAS can also record only a single polarization for single-polarization receivers.

Table 8.1: VEGAS Modes supported by each of the 8 VEGAS spectrometers.

Mode	Bandwidth (MHz)	Channels	Spectral resolution (kHz)	Minimum int. time <sup>b</sup> (ms)	Max data rate <sup>a</sup> at minimum int. time (MBs <sup>-1</sup> )
Single Sub-band modes <sup>c</sup>					
1	1500	1024	1465	1	32
2	1500	16384	92	2	187
3	1080	16384	66	4	130
4	187.5	32768	5.7	11	52
5	187.5	65536	2.9	22	52
6	187.5	131072	1.4	35	69
7	100	32768	3.1	12	51
8	100	65536	1.5	25	51
9	100	131072	0.8	40	69
10	23.44	32768	0.7	16	93
11	23.44	65536	0.4	33	93
12	23.44	131072	0.2	72	75
13	23.44	262144	0.1	134	93
14	23.44	524288	0.04	246	125
15	11.72	32768	0.4	27	93
16	11.72	65536	0.2	55	93
17	11.72	131072	0.1	123	62
18	11.72	262144	0.04	223	93
19	11.72	524288	0.02	447	93
8 Sub-band modes <sup>c</sup>					
20	23.44	4096	5.7	6	12
21	23.44	8192	2.9	12	12
22	23.44	16384	1.4	35	8
23	23.44	32768	0.7	51	12
24	23.44	65536	0.4	97	13
25	16.9	4096	4.1	8	9
26	16.9	8192	2.1	17	9
27	16.9	16384	1.0	47	6
28	16.9	32768	0.51	69	9
29	16.9	65536	0.26	132	10

<sup>a</sup> Maximum data rate is calculated for recording full polarization and all channels at the minimum integration period for one spectrometer. Each spectral value is represented by 4 bytes.

<sup>b</sup> The integration per switching state should be  $\geq$  the minimum integration. For example, if an observation uses 2 switching states, then the minimum integration will be 2 times the value listed in the table.

<sup>c</sup> For modes 20→24 the subbands can be placed within the baseband bandwidth of 1500 MHz (see note *d*) and for modes 25→29 the subbands can be placed within 1000 MHz.

<sup>d</sup> The usable frequency range for baseband bandwidth of 1500 MHz (modes 1 & 2, also for modes 20→24) is 150–1400 MHz and for bandwidth of 1000 MHz (mode 3, also for modes 25→29) it is 150–950 MHz.

## 8.2 Data Rates

The data rate for an individual bank can be calculated using

$$\text{Data Rate (GB/hr)} = 1.34 \times 10^{-5} \cdot \frac{n_{channels} \times n_{spw} \times n_{stokes} \times n_{states}}{t_{int}(\text{seconds})} \quad (8.1)$$

where  $n_{channels}$  is the number of channels per spectral window,  $n_{spw}$  is the number of spectral windows,  $n_{stokes}$  is the number of stokes parameters (2 for dual polarization, 4 for full polarization),  $n_{states}$  is the number of switching states (4 for frequency switching and 2 for total power), and  $t_{int}$  is the integration time. The total data rate for a project can be calculated by adding the data rates for each bank together.



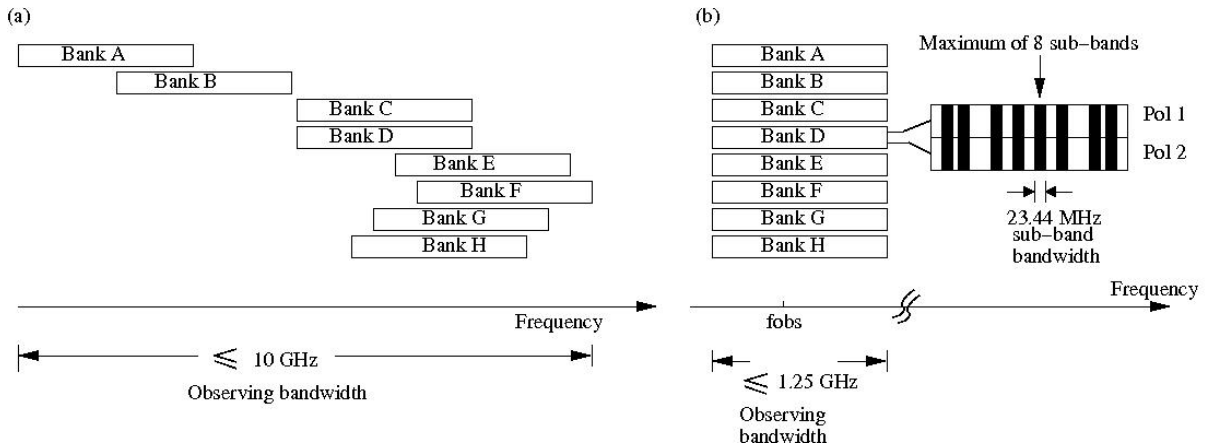


Figure 8.1: Examples of two basic VEGAS setups. Since each bank can be configured independently, these setups can be mixed and matched with the restriction that all banks have the same switching signal. (a) *Wide band, single beam, dual polarization observations*. The 8 banks can be tuned to a different frequencies where each bank produces a single spectral window. (b) *Narrow band, multiple beam, dual polarization mode*. Each bank can provide a maximum of 8 spectral windows (subbands).

## 8.3 IF Configuration

The GBT IF system introduces some constraints on routing signals from the receivers to VEGAS.

- Single beam receivers or multi-beam receiver that has been configured to use a single beam may be routed to any or all of the VEGAS banks A→H. No spectral resolution is gained with VEGAS by only using one beam of a multi-beam receiver.
- Dual-beam configurations allow each beam to be routed to a maximum of 4 VEGAS banks.
- When using 3–4 beams, each beam may be routed to up to a maximum of 2 VEGAS banks.
- When using more than 5 beams, each beam may only be routed to a single VEGAS bank.
- When using all 7 beams of the KFPA, each beam may be routed to a single VEGAS bank with an optional second copy of beam 1 being routed to the remaining VEGAS bank. This is known as the “7+1” mode of the KFPA

## 8.4 Blanking

While the observing system is switching between states (such as switching the noise diodes on or off, switching frequencies, running doppler updates, etc...) the collected data is not valid, and thus must be 'blanked' by VEGAS. VEGAS allows the user to switch states frequently enough that the required blanking time can become a non-negligible percentage of the total observing time. For efficient observing, It is important to choose switching periods that are long enough for the total amount of blanking to be negligible. The amount of blanking per switching signal is dependent on the VEGAS mode used. Conservative values are shown in Tables 8.2 and 8.3 for values with the noise diode turned either on or off. For a more thorough description of the appropriate switching periods for a given amount of blanking, and more accurate estimates of the minimum switching periods we refer the interested reader to [http://library.nrao.edu/public/memos/gbt/GBT\\_288.pdf](http://library.nrao.edu/public/memos/gbt/GBT_288.pdf)

Table 8.2: Minimum recommended switching periods (swper) with VEGAS for observations that use a noise diode.

Mode	Total power (tp)	Frequency switching <sup>a</sup> (sp)		
	Nominal <sup>b</sup> swper (sec)	Nominal <sup>c</sup> swper (sec)	$\nu_{min}$ <sup>d</sup> (GHz) swper=1.52 sec	Mapping <sup>e</sup> swper (sec)
1	0.01	0.4	115.0	0.4
2	0.028	0.4	115.0	0.4
3	0.04	0.4	115.0	0.4
4	0.028	0.4	115.0	0.4
5	0.0559	0.4	115.0	0.4
6	0.1118	0.4318	115.0	1.52
7	0.0524	0.4	115.0	0.4
8	0.1049	0.4249	115.0	1.52
9	0.2097	0.5297	59.6	1.52
10	0.2237	0.5437	54.4	1.52
11	0.4474	0.8948	16.5	1.52
12	0.8948	1.7896		1.7896
13	1.7896	3.5791		3.5791
14	3.5791	7.1583		7.1583
15	0.4474	0.8948	16.5	1.52
16	0.8948	1.7896		1.7896
17	1.7896	3.5791		3.5791
18	3.5791	7.1583		7.1583
19	7.5383	14.3166		14.3166
20	0.028	0.4	115.0	0.4
21	0.0559	0.4	115.0	0.4
22	0.1118	0.4318	115.0	1.52
23	0.2237	0.5437	54.4	1.52
24	0.4474	0.8948	16.5	1.52
25	0.0388	0.4	115.0	0.4
26	0.0777	0.4	115.0	1.52
27	0.1553	0.4753	89.7	1.52
28	0.3107	0.6307	33.8	1.52
29	0.6214	1.2428	8.6	1.52

- <sup>a</sup> When frequency switching, switching periods must always be  $>0.4$  seconds due to the settling time of the LO1.
- <sup>b</sup> Recommended minimum switching period (swper) for total power observations with noise diodes (swtype='tp'). These values will yield less than 10% blanking overall.
- <sup>c</sup> Recommended minimum switching period for frequency switching observations with noise diodes (swtype='sp'). These values will yield less than 10% blanking in the first state of the switching cycle as well as less than 10% blanking overall.
- <sup>d</sup> The minimum recommended switching period is 1.52 seconds **when Doppler tracking** frequencies above  $\nu_{min}$ .
- <sup>e</sup> Recommended minimum switching period (swper) for **Doppler-tracked**, frequency switching observations with noise diodes (swtype='sp'). These values will yield less than 10% blanking in the first state of the switching cycle as well as less than 10% blanking overall. This switching period will result in less than 10% of the data being blanked. These values assume that the maps are sampled at twice Nyquist in the scanning direction and that there are four integrations per switching period. **when Doppler tracking**.

Table 8.3: Minimum recommended switching periods (swper) with VEGAS for observations that do not use a noise diode.

Mode	Total power (tp_nocal)		Frequency switching <sup>a</sup> (sp_nocal)		
	Nominal <sup>b</sup> swper (sec)	Mapping <sup>c</sup> swper (sec)	Nominal <sup>d</sup> swper (sec)	$\nu_{min}$ <sup>e</sup> (GHz) swper=0.76 sec	Mapping <sup>f</sup> swper (sec)
1	0.0005	0.001	0.4	115.0	0.4
2	0.0014	0.0028	0.4	115.0	0.4
3	0.002	0.004	0.4	115.0	0.4
4	0.01	0.0114	0.4	115.0	0.4
5	0.0199	0.0227	0.4	115.0	0.4
6	0.0301	0.0357	0.4	115.0	0.76
7	0.0102	0.0128	0.4	115.0	0.4
8	0.0203	0.0256	0.4	115.0	0.76
9	0.0301	0.0406	0.4	115.0	0.76
10	0.0056	0.0168	0.4	115.0	0.76
11	0.0112	0.0336	0.4474	33.1	0.76
12	0.028	0.0727	0.8948		0.8948
13	0.0447	0.1342	1.7896		1.7896
14	0.0671	0.2461	3.5791		3.5791
15	0.0056	0.028	0.4474	33.1	0.76
16	0.0112	0.0559	0.8948		0.8948
17	0.0336	0.123	1.7896		1.7896
18	0.0447	0.2237	3.5791		3.5791
19	0.0895 or <sup>g</sup> 0.38	0.4474	7.1583		7.1583
20	0.0051	0.0065	0.4	115.0	0.4
21	0.0101	0.0129	0.4	115.0	0.4
22	0.0301	0.0357	0.4	115.0	0.76
23	0.0405	0.0517	0.4	115.0	0.76
24	0.0755	0.0979	0.4474	33.1	0.76
25	0.007	0.009	0.4	115.0	0.4
26	0.0141	0.018	0.4	115.0	0.76
27	0.0398	0.0476	0.4	115.0	0.76
28	0.0544	0.0699	0.4	68.6	0.76
29	0.101	0.132	0.6214	17.1	0.76

- <sup>a</sup> When frequency switching, switching periods must always be >0.4 seconds due to the settling time of the LO1.
- <sup>b</sup> Recommended minimum switching period (swper) for total power observations that do not use noise diodes (swtype='tp\_nocal'). This value is equivalent to the hardware exposure value for VEGAS.
- <sup>c</sup> Recommended minimum switching period (swper) for total power OTF mapping observations that do not use noise diodes (swtype='tp\_nocal') **when Doppler Tracking**. These values will yield less than 10% blanking overall and assume that the maps are sampled at twice Nyquist in the scanning direction and that there are four integrations per switching period.
- <sup>d</sup> Recommended minimum switching period for frequency switching observations that do not make use of noise diodes (swtype='sp\_nocal'). These values will yield less than 10% blanking in the first state of the switching cycle as well as less than 10% blanking overall.
- <sup>e</sup> The minimum recommended switching period is 0.76 seconds **when Doppler tracking** frequencies above  $\nu_{min}$ .
- <sup>f</sup> Recommended minimum switching period (swper) for **Doppler-tracked**, frequency switching OTF mapping observations without noise diodes (swtype='sp\_nocal'). These values will yield less than 10% blanking in the first state of the switching cycle as well as less than 10% blanking overall. This switching period will result in less than 10% of the data being blanked. These values assume that the maps are sampled at twice Nyquist in the scanning direction and that there are four integrations per switching period.
- <sup>g</sup> For mode 19 this value is 0.0895/0.38 seconds for observations below/above 10.3 GHz **when Doppler tracking**.

## 8.5 Monitoring VEGAS observations

The Spectral Line tab in the Astrid Data Display (See § 5.1.6) is not fully capable of displaying VEGAS observations in real time (it will display passbands at the end of a scan, and may be used in offline mode). Rather, there are four monitoring tools that are useful with VEGAS:

- the VEGAS Data Display at <https://vegasdisplay.gb.nrao.edu/> (See § 5.3.1).
- the VEGAS CLEO screen (See § 5.3.2).
- VEGASDM – the VEGAS Data Monitor (See § 5.3.3).
- `vegas_status` – the VEGAS shared memory display (See § H.0.2.1).

The first three items are generally useful while observing with VEGAS and are described in § 5.3, while `vegas_status` is for specialized problem diagnosis and is described in Appendix H.

## 8.6 The Online Filler and Filling VEGAS data using SDFITS

VEGAS writes “Engineering” FITS files. Once a scan is over, the “Filler” reads these files, combines the data with metadata from the Antenna and other FITS files, and produces a single-dish (SDFITS) file. This can be done automatically, by the on-line filler, or manually by the Observer. Due to the significantly higher data rate, and some other features of VEGAS, the filling process requires some oversight by the user.

### 8.6.1 The Online Filler

The online filler will make every attempt to fill the SDFITS file automatically. In this case, a file will be produced in:

```
/home/sdfits/<project>
```

and GBTIDL can connect to it automatically using the “online”, or “offline, <project>” commands. There are some caveats, however.

- Because of the way VEGAS writes it’s data, the filler cannot start filling until the scan has finished. For immediate inspection of data, Observers are urged to use the real-time display (see § 5.3.1).
- For “large” scans, the filler could potentially fall behind the data acquisition process. To avoid this, the filler will skip scans that it cannot keep up with. The rule is:

```
If (integration time / total number of spectra per integration) < 0.00278s
    skip the scan
Except if (integration length >= 0.9s) it will be filled
```

The total number of spectra per integration is the total across all banks. So, for example, 2 banks, 8 subbands, 2 polarizations and 4 switching states (e.g. frequency switching with calibration) will produce  $2*8*2*4 = 128$  spectra, and so if the integration time is  $< 0.356s$  the online filler will not fill that data. The 0.9s limit is because for that integration time the online filler can almost keep up even in the worst

case, and interscan latencies, pauses for pointing and focus scans, and so on will normally allow it time to catch up. The online filler prints a summary in `/home/sdfits/<project>/<project>.log` indicating what scans were filled, had problems and were skipped, and if any data was skipped because the data rate was too fast.

The decision on whether to fill or not is made independently for each bank. For cases where the integration time is close to the limit it's possible that some banks might be filled while others are not filled for the same scan if the number of subbands or the number of polarizations vary across the banks. The summary log file will indicate when this happens.

If observers are concerned about the interpolation across the center channel (see § 8.7.1) they can turn that off in `sdfits` by using the “`-nointerp`” option.

## 8.6.2 Filling Offline

You may wish to (re-) fill your data offline. In this case, you make use the SDFITS filler program in the standard manner. Note however, that the actual VEGAS data is stored to a high-speed (“lustre” file system). For a current list of lustre client machines please see <http://www.gb.nrao.edu/pubcomputing/public.shtml>

If you try to fill data without being logged into a lustre client, the filler will fail with the error message:

```
VEGAS data expected but not found, this workstation is not a lustre client.
For a list of public lustre client workstations see:
http://www.gb.nrao.edu/pubcomputing/public.shtml
```

-In this case, `ssh` to a lustre client (using the domain `.gb.nrao.edu`), and fill your data there.

Filling using `sdfits` directly (instead of the output online `sdfits`) might also be useful if there are a lot of spectra to be processed in GBTIDL simply because it improves the response times in GBTIDL if there are not as many spectra to search through. So if there's a convenient way to divide up the scans, then this sort of syntax works (`sdfits -help` for more details):

```
sdfits -backends=vegas -scans=<scan-list> <PROJECT_SESSION> <OUTPUT_PREFIX>
```

- `scan-list` is a list of comma separated scans to fill using colons to denote ranges e.g., `-scans=1,4:6,10:` would fill scans 1,4,5,6 and all scans from 10 onwards.
- `<PROJECT_SESSION>` is what you'd expect, e.g. “AGBT14A\_252\_04”.
- `<OUTPUT_PREFIX>` is the leading part of the output directory name, e.g. `scan5to25` would result in a directory named `scan5to25.raw.vegas`

## 8.7 Instrumental Features and their Cure

The architecture of the VEGAS hardware, specifically the architecture of the ADC, results in some characteristic features in the VEGAS spectrum. Specifically, these are:

- a strong spurious single-channel wide spike at the exact center of the ADC passband, - the so-called “center spike”.
- weak single-channel wide spurs at various locations in the bandpass - the 32 spurs.

### 8.7.1 The Spike

The center spike is caused by the FPGA clock. By default, the center spike is interpolated over by the SDFITS filler by taking the mean of the adjacent channel on either side of the spike. The center spike is also interpolated-over by the real-time spectrum display. We have chosen to interpolate over this spike as it is omnipresent, and can cause problems for data reduction (such as system temperature calculations). If you are concerned about this process, you may shift your line from the center of the passband using the `deltafreq` keyword in your `astrid` script.

### 8.7.2 The Spurs

When attempting to search for RFI with VEGAS by running a high-pass filter through the data, significantly more spikes/spurs were found than naively expected. These spurs could be found in the same bins in relatively RFI free wavelengths, such as Q-band. The spurs appear at the same location (in bin space) for a given mode and have relatively stable amplitudes. These faint spurs are not always directly visible in the data, but became clear when high-pass filtered, as shown in Figure 8.2. After significant testing, it was determined that these spurs are below the spurious-free dynamic range of -60dBc specified by the manufacturer, and cannot be fully removed. In overly simplistic terms, the spurs are caused by the leaking of the FPGA clock into the four interleaved ADCs. For further information see § H.3.

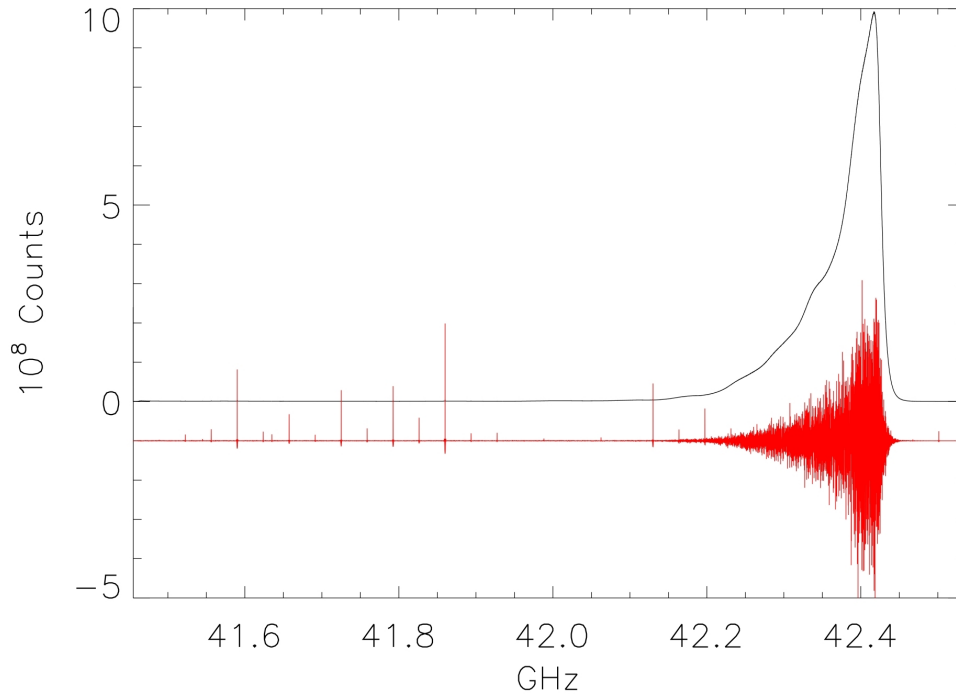


Figure 8.2: Spectra of a noise source centered near 42.4 GHz (black). A high pass filtered version of the data (offset by -1 and scaled by a factor of 100 for visibility) is shown in red, showing the spurs.

These spurs are relatively stable and will remain constant (for a given mode) and the magnitude of the spurs is relatively constant. These features are also quite small by most standards (Spurious Free Dynamic Range no more than -60dBc), but nevertheless can be problematic when looking for faint narrow features. The stability of these features allows them to be removed by standard data practices (such as position and/or frequency switching), but they are an added noise source which can bleed through to the final product. Due to the limited and often negligible effect of these spurs, we do not automatically interpolate across them, but let the user decide how to handle those channels.

## 8.8 Known Bugs and Features

### 8.8.1 vegasdisplay is not updating, or is running very slowly

This occasionally happens. The cure is to ask the Operator to restart the display:

```
TaskMaster gbtdata stop 15
TaskMaster gbtdata start 15
```

### 8.8.2 Data is not filling

The online filler checks for project changes when it is not actively filling a scan. This means that if the previous project was a VEGAS one and it ended on a long scan, the filler may still be filling that project when the VEGAS scan has finished in your project. If you suspect that this is the case, the only solution is to ask the Operator to restart the online filler task.

— All data can still be accessed in GBTIDL by running SDFITS offline (see § 8.6.2).

### 8.8.3 There is a “square wave” and/or divot in my VEGASDM display

The samples which are taken to produce the VEGASDM total power display run asynchronously to the switching signals. Hence, the sampling may occur during the “Cal on” phase at one point in time, and then drift into the “Cal off” phase sometime later. This may produce an apparent square wave in the VEGASDM output, with an amplitude of a few tenths of a dB, and a period of seconds.

Similarly, it is possible for the VEGASDM data to be acquired when the LO is updating (e.g. during a Doppler track). These data are blanked in the true VEGAS spectral data acquisition, but may cause drop-outs in the VEGASDM samples.





## Chapter 9

# Pulsar Observing with GUPPI

The Green Bank Ultimate Pulsar Processing Instrument (GUPPI) is an FPGA + GPU backend that offers many advantages over previous pulsar backends. GUPPI uses 8-bit sampling to dramatically improve upon the dynamic range and RFI resistance of the Spectral Processor. GUPPI can observe in either incoherent or coherent dedispersion modes over bandwidths of 100, 200, or 800 MHz, with 64, 128, 256, 512, 1024, 2048, or 4096 spectral channels. Three pulsar observing modes are available with either dedispersion mode: search, on-line folding, and “cal” (a specialized on-line folding mode). Integration time is highly configurable, subject to the maximum data rate of < 200 MB/s. Typically, all four Stoke’s parameters are recorded, but the option to only record total intensity (Stoke’s I) is available in incoherent search mode. GUPPI can be used with any receiver with the exception of MUSTANG (only one polarization is available when using the Ka-band receiver).

GUPPI is controlled by a machine called beef, which also mounts two disks used to record incoherent mode data. In coherent modes, the total bandwidth is divided into eight parts, with each part being sent to an HPC equipped with an nVidia GPU that performs the coherent dedispersion. These data are then combined in the first stage of post-processing to recover the full bandwidth.

GUPPI can be record baseband data in expert-user modes. If you are interested in this capability contact Scott Ransom (sransom@nrao.edu), Ryan Lynch (rlynch@nrao.edu), or Paul Demorest (pdemores@nrao.edu).

Additional resources can be found at:

<https://safe.nrao.edu/wiki/bin/view/CICADA/GUPPIAstridGuide> and

Detailed information: <https://safe.nrao.edu/wiki/bin/view/CICADA/GUPPISupportGuide>

## 9.1 Summary

Basic control of GUPPI is fully integrated into Astrid. A typical observing script will consist of: 1) loading a catalog of sources, 2) setting up the IF/LO path and GUPPI via `Configure()`, 3) slewing and balancing the system, and 4) taking data, mostly likely using a `Track()` command. Depending on your science goals, you may also wish to take calibration scans, either through an `OnOff()` or separate `Track()` commands. If you are observing at C-band or above then you should also run an `AutoPeakFocus()`. Each of these Astrid functions are described elsewhere in this guide. Here we will focus on GUPPI configuration and tips and strategies specific to pulsar observing with GUPPI.

GUPPI-specific tools are available for checking that the system is properly balanced, monitoring power levels and RFI, and checking that GUPPI is properly configured and successfully taking data.

## 9.2 Details of GUPPI Configuration

Pulsar observers will typically use only certain values for some Astrid configtool keywords. These are described below.

- **obstype** will always be **Pulsar**.
- **backend** will always be **GUPPI**.
- **bandwidth** can be either **100**, **200**, or **800**.
- **ifbw** will be **80** for **100 MHz bandwidth modes** and **0** for **200 or 800 MHz bandwidth modes**.
- **tint** is very flexible. Under the hood, it is controlled by the hardware accumulation length, so that  $t_{\text{int}} = \text{acc\_len} \times n_{\text{chan}} / \text{BW}$ . **acc\_len** can take on values from 4 (down to two in some special cases) up to 1024. Most observers will want to keep their integration times fast enough to resolve fast MSPs, while keeping data rate  $< 200$  MB/s.
- **swmode** will either be **tp** for calibration scans or **tp\_nocal** for pulsar scans.
- **noisecal** should be **lo** for calibration scans (this uses the low-power noise diodes) and **off** for pulsar scans.

The Astrid configtool also supports a number of GUPPI specific keywords, some of which are only actively used in certain observing modes. These keywords and their allowed values are described below.

- **guppi.obsmode** controls both the dedispersion and observing mode. All data are written in PSRFITS format. Allowed values are
  - **search**: Incoherent search-mode, i.e. spectra are rapidly written to disk.
  - **fold**: Incoherent fold-mode, i.e. spectra are folded on-line using a pulsar ephemeris, with sub-integrations written to disk at a rate controlled by the **guppi.fold.dumptime** keyword. *Note: incoherent fold-mode is effectively deprecated in favor of coherent fold-mode.*
  - **cal**: Incoherent cal-mode, i.e. data are folded at a constant 25 Hz frequency; used in conjunction with the pulsed noise diodes for each receiver.
  - **coherent\_search**: Coherent search-mode, i.e. spectra are coherently dedispersed at a DM specified using the **guppi.dm** keyword before being channelized, accumulated, and written to disk.
  - **coherent\_fold**: Coherent fold-mode, i.e. spectra are coherently dedispersed at the DM specified in the ephemeris file before being written to disk. Since the DM is read from the ephemeris file, the **guppi.dm** keyword is not needed.
  - **coherent\_cal**: Coherent cal-mode, i.e. data are taken as in coherent fold mode, although no dedispersion is applied since the noise diode is not dispersed.
- **guppi.polnmode** controls whether Full Stoke's or total intensity data are recorded. Allowed values are **full\_stokes** and **total\_intensity**, though total intensity can only be used in incoherent search-mode.
- **guppi.numchan** sets the number of spectral channels. Allowed values are **any power-of-two between 64 and 4096**, though care must be taken not to exceed the maximum data rate.
- **guppi.outbits** controls the number of bits used for output values. The only allowed value is **8**.

- **guppi.scale** controls the internal scaling so that the output data is properly scaled for 8-bit values. This value is typically chosen from experience with the observing set-up. Contact your project friend for advice on which value to use.
- **guppi.datadisk** controls which RAID data are written to in incoherent modes. Allowed values are **data1** or **data2**. In coherent modes data are written to eight HPC machines and this keyword is not used.
- **guppi.dm** controls the DM used for coherent search-mode. It is not used by any other modes.
- **guppi.fold\_parfile** specifies the path to the ephemeris (parfile) used for either incoherent or coherent fold-modes. *The parfile must exist and be visible from beef, and readable by tempo.*
- **guppi.fold\_bins** controls the number of phase bins used for either incoherent or coherent fold- or cal-modes. Enough bins should be used to fully resolve fine profile structure. Typical values are 256 in incoherent modes (which are not typically used anymore) and 2048 in coherent modes.
- **guppi.fold\_dumptime** controls the length of a sub-integration in either incoherent or coherent fold- or cal-modes. The value is specified in seconds, with 10 s being typical. It must be shorter than the total scan length.

The following is a heavily documented configuration script. As is, it will configure for search-mode at L-band, but it can be easily edited for other observing modes.

```
#
# An example well-documented L-band "search"-mode script
#
config_g=""
obstype = 'Pulsar'
# usually 'Rcvr_342', 'Rcvr_800', 'Rcvr1_2', 'Rcvr2_3', 'Rcvr4_6'
receiver = 'Rcvr1_2'
pol = 'Linear'      # C-band and below are native 'Linear'
restfreq = 1500.0  # in MHz
backend = 'GUPPI'
bandwidth = 800    # in MHz. 100, 200, or 800
ifbw = 0           # 0 for >100MHz BW modes, 80 for 100MHz.
tint = 40.96e-6    # sample time in seconds (very flexible)
swmode = 'tp_nocal' # 'tp' for cal, 'tp_nocal' for no cal
noisecal = 'off'   # if no cal, set to 'off', else 'lo'
# The following are boilerplate until 'guppi' section
# You should probably not change them...
swtype = 'none'
swper = 0.04
swfreq = 0.0, 0.0
nwin = 1
deltafreq = 0
vlow = 0
vhigh = 0
vframe = 'topo'
vdef = 'Radio'
# — GUPPI specific params — #
# obsmode can be 'search', 'fold', 'cal', 'coherent_search', 'coherent_fold',
# or 'coherent_cal'
guppi.obsmode = 'search'
# numchan can be a power-of-two between 64 to 4096
guppi.numchan = 2048
```

```

# polnmode is 'full_stokes' or 'total_intensity'
guppi.polnmode = 'total_intensity'
# scale should be set in first config block and
guppi.scale = 9.0
# Top level disk where data will be written
guppi.datadisk = 'data2' # 'data1' or 'data2'
guppi.outbits = 8        # Currently only 8 is available
guppi.dn = 50           # in pc/cc. Only used in 'coherent_search'
# Folding specific params — not needed for cal or search
# Make sure that the parfile exists!
guppi.fold_parfile = "/users/sransom/parfiles/1713.par"
guppi.fold_bins = 2048  # number of bins in profile
guppi.fold_dumptime = 10 # in sec
"""
Configure(config_g)

```

## 9.3 Balancing and Checking Levels

Before balancing the system you must configure it for your observing set-up as described above. Once that is done, slew to your source of interest using the `Slew()` command, and then issue a `Balance()` command. Note that these can (and should) all be included in the same scheduling block.

Once the system has finished balancing, you should run the `guppi_adc_hist` command on `beef`. As the name implies, this will plot two histograms, one for each polarization channel, of the data values being output by the ADCs. Both histograms should have a roughly Gaussian shape with a FWHM of about 30. The two histograms do not need to be identical but they should be reasonably close. Figure ?? shows a well-balanced example. *Note that `guppi_adc_hist` will only update when there is no scan in progress. If you try to run it while taking data you will only see the results of the last run, which will not be up-to-date.* For this reason, is usually a good idea to put a `Break()` statement in your scheduling block, which will pause Astrid and give you a chance to check the ADC levels. If the ADC histogram looks extremely narrow, then GUPPI is probably not getting signal. If it is very broad then the input power is too high. In both cases, stop the scheduling block and resubmit, so that you configure and balance again. If the problem persists after one or two attempts, ask the operator to contact the on-duty support scientist.

Once the IF/LO system is properly balanced you will need to check the internal GUPPI internal scaling (controlled via the `guppi.scale` parameter). The only way to do this is via the `guppi_monitor` command while data is being taken. This command will plot the bandpass. The mean bandpass level should be around 20–30, and less than 50. Figure ?? shows a well-scaled example. If the level looks too high or too low, you should adjust the `guppi.scale` parameter accordingly (the relationship is linear) and then resubmit your scheduling block (*be sure to save the scheduling block first!*). It is highly recommended that this be done using a short scan in cal-mode. These data will be written to disk but you should only use the scans that are properly scaled for your actual science, so make a note of which scan is which. Once you have determined an appropriate `guppi.scale` value, be sure to update any other configurations accordingly. *DO NOT leave `guppi_monitor` running during your observations, as it uses a lot of resources. DO re-run it from time to time during your session to monitor the RFI environment, and close the program once you are done.*

In practice, appropriate `guppi.scale` parameters have already been determined for most common configurations, and it is a good idea to re-use these values. Contact your project friend for advice on which values to use. This will reduce the `guppi_monitor` step to a sanity check/check of the RFI environment and reduce overhead time.

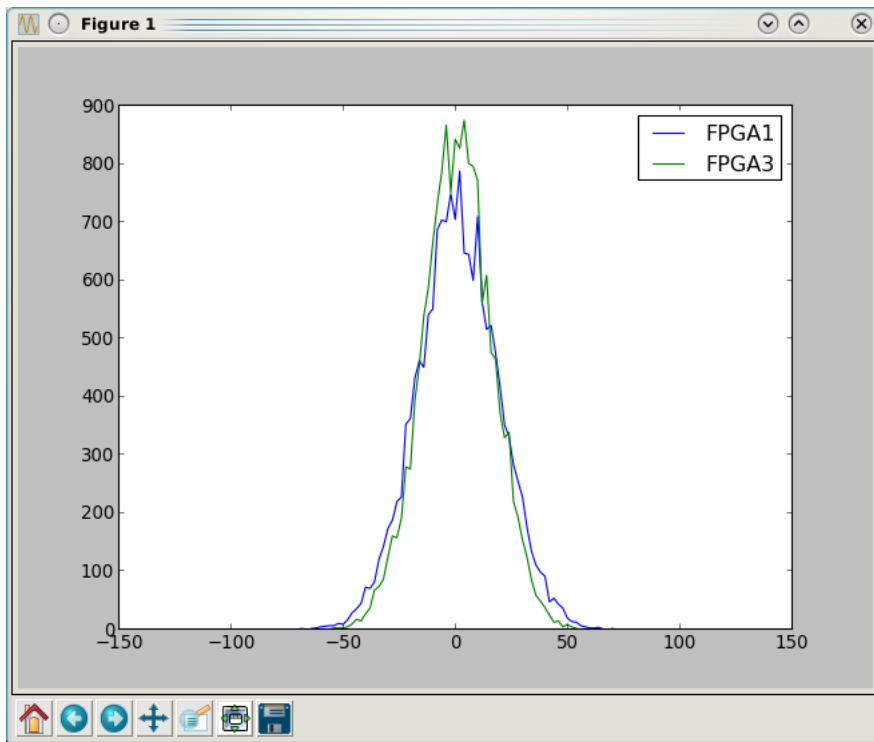


Figure 9.1: The `guppi_adc_hist` screen. The two histograms (one for each polarization channel) should be of roughly the same width and amplitude. Both should be roughly Gaussian in shape and have FWHMs of about 30.

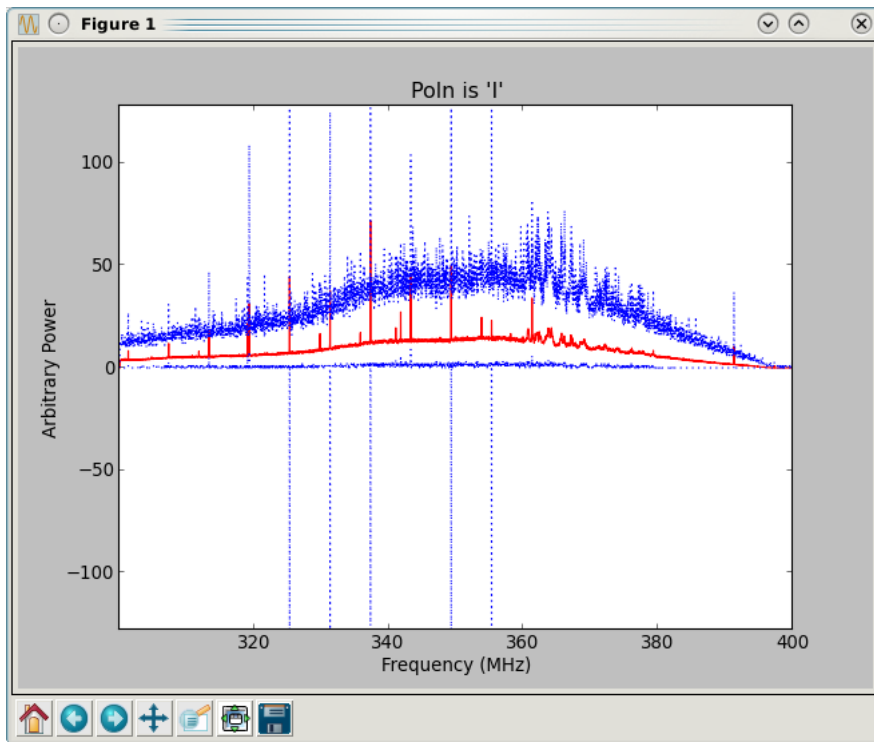


Figure 9.2: The `guppi_monitor` screen. In this example the bandpass at 350 MHz is shown. The red line shows the average and the blue lines show the maximum and minimum over the last second. A mean level of around 20–30 is optimal.

## 9.4 Taking Data

In most cases you will use a `Track()` command to take data. This is most easily accomplished by loading a catalog and specifying the source name in your call to `Track()`, e.g.

```
#
# Slew, balance, then take data...
#
bright_MSPs = Catalog(pulsars_bright_MSPs_GBT)

Configure(config_g)
Slew("B1937+21")
Balance()

# Track is how we take data now.
# Scan duration is in sec. Recommend you
# add 5-sec to account for some delays in the system

Track("B1937+21", endOffset=None, scanDuration=65)
```

As shown in the example above, it is usually a good idea to add 5 seconds to the scan duration when using fold-mode. This ensure that the last sub-integration will be written to disk.

If you wish to take drift scan data or observe during maintenance time, you can pass the current encoder position to track instead of an astronomical object, e.g.

```
#
# Take Drift-scan data
#
Configure(config_g)
Balance()
loc = GetCurrentLocation("Encoder")
Track(loc, endOffset=None, scanDuration=20000)
```

When observing flux calibrators it can be useful to use the `OnOff()` function, e.g.

```
#
# Slew, balance, then take data...
#
fluxcal = Catalog(fluxcal)

Configure(config_g)
Slew("3C286")
Balance()

# Track is how we take data now.
# Scan duration is in sec. Recommend you
# add 5-sec to account for some delays in the system

Track("3C286", Offset('J2000', 1.0, 0.0, cosv=True), scanDuration=65)
```

Consult Chapter 6 for other useful observing tricks, such as `Horizon()` objects and using `startTime` or `stopTime` to control scan duration.

If you need to stop a scan early, you can use `Stop` or `Abort` in Astrid.

## 9.5 Monitoring GUPPI During an Observation

When you first start an observing session it is a good idea to have several terminals open on `beef`, and to set-up your shell for the GUPPI environment by using either `source /opt/64bit/guppi/guppi_daq/guppi.bash` or `source /opt/64bit/guppi/guppi_daq/guppi.csh` as appropriate. Once this is done you can use `guppi_adc_hist` and `guppi_monitor` as described above, as well as other tools described here.

### 9.5.1 Monitoring Incoherent Data Taking

To check the status shared memory, use `guppi_status`. This will display the current GUPPI configuration. Check that the observing mode, source, integration time, parfile, etc. are all correct. When data is being taken you will see the `NETSTAT` key turn to `Receiving` and the `DISKSTAT` keyword change to either `waiting` or `writing`. Figure 9.3 shows an example of this screen.

You should also follow the data acquisition server log by typing

```
> tail -f /tmp/guppi_daq_server.log
```

This will update when scans start or end, new files are open or closed, and sub-integrations are written to disk. You should also check this log for warnings about large fractions of dropped packets. This can occur when the `beef` disks are under heavy use by another user (which should not usually happen if people are being careful), if there are network issues, etc. If you start to see many warnings about dropped packets, ask the operator to contact the on-duty support scientist. Note that it is common to see a very small number of dropped packets ( $\ll 1\%$ ) at the start of a scan. This is not an issue and can be safely ignored.

You can also check the data that is being written to to disk by going to the output directory. This will be in `/data[12]/<username>/<projectID>/<date>`, where `data[12]` refers to either `/data1` or `/data2` (whichever you specified with `guppi.datadisk`), `username` is your NRAO user name, `project ID` is the GBT project ID as it appears in `astrid` (e.g. `AGBT14A.507`), and `date` is the UTC date in `YYYYMMDD` format.

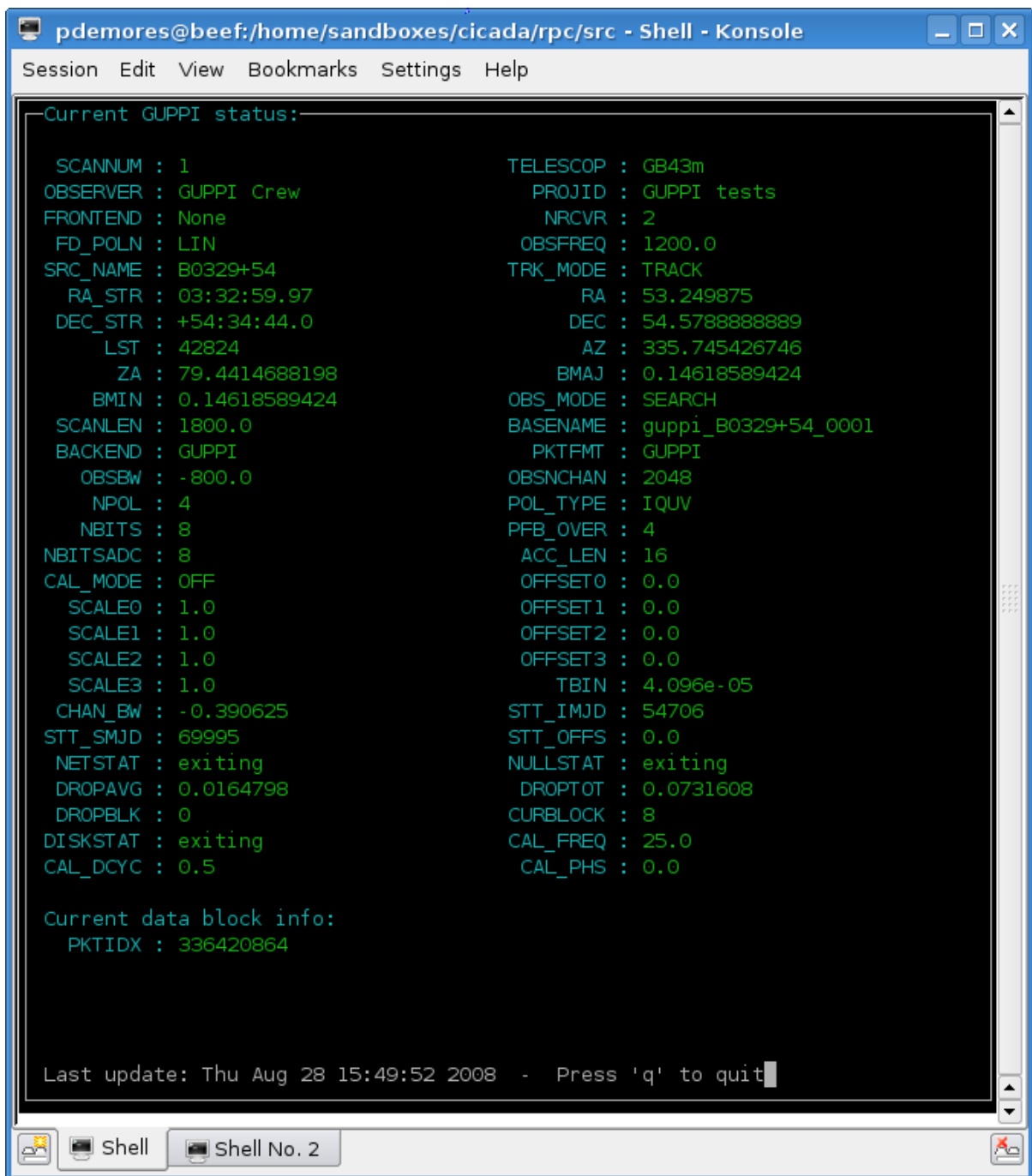
### 9.5.2 Monitoring Coherent Data Taking

In coherent dedispersion mode data are written to one of eight GPU nodes. This means that `guppi_status` will not indicate that data is being written unless it is running on one of the GPU nodes. The `guppi_daq_server.log` file will also not update on `beef`.

Instead, you should use `guppi_gpu_status`. This will show the data taking status on each GPU node and show the last few lines from the server logs on each node. Monitor this screen for messages about packet loss or errors on the various nodes. The most common problems will be when a node is hung in `Stopping` or `Waiting`, even while other nodes are taking data. This usually indicates that the data acquisition software has crashed on that node. You may also see a status of `Unk` (for unknown), which usually means that the GPU node itself crashed. In both cases, ask the operator to contact the on-duty support scientist.

In coherent fold- or cal-mode you can check data using a web browser by navigating to `http://www.gb.nrao.edu/guppi/`. This page shows a summary of the most recent output file written to disk on each GPU node and updates automatically. It can be accessed from anywhere. If it doesn't appear to be updating as expected, the auto-plotting routine may have crashed. You can check data directly





The image shows a terminal window titled "pdemores@beef:/home/sandboxes/cicada/rpc/src - Shell - Konsole". The window contains the following text:

```
Current GUPPI status:

  SCANNUM : 1                      TELESCOP : GB43m
  OBSERVER : GUPPI Crew             PROJID  : GUPPI tests
  FRONTEND : None                   NRCVR   : 2
  FD_POLN  : LIN                    OBSFREQ : 1200.0
  SRC_NAME : B0329+54              TRK_MODE: TRACK
  RA_STR   : 03:32:59.97           RA      : 53.249875
  DEC_STR  : +54:34:44.0           DEC     : 54.5788888889
  LST      : 42824                 AZ      : 335.745426746
  ZA       : 79.4414688198         BMAJ    : 0.14618589424
  BMIN     : 0.14618589424         OBS_MODE: SEARCH
  SCANLEN  : 1800.0               BASENAME: guppi_B0329+54_0001
  BACKEND  : GUPPI                PKTFMT  : GUPPI
  OBSBW    : -800.0               OBSNCHAN: 2048
  NPOL     : 4                    POL_TYPE: IQUV
  NBITS    : 8                    PFB_OVER: 4
  NBITSADC : 8                    ACC_LEN : 16
  CAL_MODE : OFF                  OFFSET0  : 0.0
  SCALE0   : 1.0                  OFFSET1  : 0.0
  SCALE1   : 1.0                  OFFSET2  : 0.0
  SCALE2   : 1.0                  OFFSET3  : 0.0
  SCALE3   : 1.0                  TBIN    : 4.096e-05
  CHAN_Bw  : -0.390625            STT_IMJD: 54706
  STT_SMJD : 69995                STT_OFFS: 0.0
  NETSTAT  : exiting              NULLSTAT: exiting
  DROPAVG  : 0.0164798           DROPTOT : 0.0731608
  DROPBLK  : 0                    CURBLOCK: 8
  DISKSTAT : exiting              CAL_FREQ: 25.0
  CAL_DCYC : 0.5                  CAL_PHS  : 0.0

Current data block info:
  PKTIDX  : 336420864

Last update: Thu Aug 28 15:49:52 2008 - Press 'q' to quit
```

The terminal window also shows a taskbar at the bottom with icons for "Shell" and "Shell No. 2".

Figure 9.3: The GUPPI Status Display screen.

by ssh'ing into the GPU nodes (e.g. `ssh gpu1`) and navigating to `/data/gpu/partial/gpu[1-9]` where `gpu[1-9]` refers to the GPU node of interest. Note that though there are nine GPU nodes, only eight are used at a given time (one is a spare).

## 9.6 Processing Your Data

Data are recorded in the PSRFITS standard, which can be processed by all common pulsar data analysis packages (e.g. PRESTO<sup>1</sup>, SIGPROC<sup>2</sup>, PSRCHIVE<sup>3</sup>, and DSPSR<sup>4</sup>). Incoherent mode data are written to `/data[12]/<username>/<projectID>/<date>`, where `data[12]` refers to either `/data1` or `/data2` (whichever you specified with `guppi.datadisk`), `username` is your NRAO user name, `projectID` is the GBT project ID as it appears in `astrid` (e.g. `AGBT14A_507`), and `date` is the UTC date in `YYYYMMDD` format. Each data disk has a capacity of 7.5 TB, but in practice we usually are only able to keep  $< 3$  TB free on each disk. **Please make arrangements to move large volumes of data off of beef as quickly as possible after your observing session. It can be extremely problematic if the disks get too full. You can expect to be bugged relentlessly if data are not moved off of the beef disks in a timely manner.** Please contact your project friend if you need help managing your data.

Coherent mode data are written to each GPU node and need to be combined to recover the full bandwidth. For fold- and cal-mode data, you can do this yourself using the executable scripts

```
/home/gpu/bin/get_data or
/home/gpu/bin/combine_psrfits
```

These should be run in `/home/scratch` area while you are logged in to beef, or in your user area on `/data1` or `/data2`. But note that you must be logged in to beef. These scripts will `rsync` the data off of each GPU node and put them in temporary sub-directories, and then combine the data. Note that the scripts assume that a `parfile` exists in `$HOME/tzpar/<source>.par`, where `source` is the source name you specified in `Astrid`.

Coherent search-mode data require root privileges to combine, so contact your project friend if you need help with that.

## 9.7 Tips and Tricks

- If you are searching for pulsars or observing a new source, consider observing a well known pulsar as a test source at the start of your session to make sure that things are working properly. A cal-mode can also be used.
- If `guppi_status` shows unexpected values, the system seems to be having trouble balancing, or you experience other issues, ask the operator to do a “conform params” and “prepare” on GUPPI (you can also do this yourself in Cleo if you know how). This simple step can solve lots of problems, especially if you are the first GUPPI project to observe following a maintenance period.
- If you are observing multiple sources with relatively short scan lengths, and the operator needs to take control for a wind-stow or snow-dump, ask if you can let the current scan finish and then use `Pause` to let the operator take control. Once control is released back to you, you can simply `un-pause` and pick up where you left off. But if the operator needs to take control immediately, abort your scan and let them take over.

<sup>1</sup><http://www.cv.nrao.edu/~sransom/presto/>

<sup>2</sup><http://sigproc.sourceforge.net/>

<sup>3</sup><http://psrchive.sourceforge.net/>

<sup>4</sup><http://dspsr.sourceforge.net/>

- The GBT noise diodes are stable over short-to-medium time scales, and a number of continuum flux calibration scans are available for common observing set-ups (this is especially true of 820 MHz and L-band NANOGrav set-ups because NANOGrav observes flux calibrators at least once a month). If your project requires flux calibration, consider contacting your project friend to see if appropriate calibration data already exist.
- Before writing scheduling blocks from scratch, ask your project friend if there are any already available from other projects that might suit your needs. This minimizes the possibility of an incorrect set-up or scheduling block.

## 9.8 Warnings

- Do not run any commands from the GUPPI prompt!
- Do not run `guppi_set_params` from the command line at all! This is all handled by configuring in Astrid now.

## 9.9 More Information

There are several other example configurations which you can copy, load into Astrid, or simply browse in `/users/sransom/astrid`. They are:

- `/users/sransom/astrid/GUPPI_astrid_example.py` The well-documented S-band search-mode example from above
- `/users/sransom/astrid/GUPPI_astrid_820MHz_cal.py` A 200-MHz BW 'cal' scan using the PF1 receiver at 820 MHz
- `/users/sransom/astrid/GUPPI_astrid_820MHz_fold.py` A 200-MHz BW 'fold' scan of a bright MSP using the PF1 receiver at 820 MHz
- `/users/sransom/astrid/GUPPI_astrid_Xband_fastdump.py` A special 256-channel fast-dump mode at X-band for Crab Giant Pulses
- `/users/sransom/astrid/GUPPI_astrid_350MHz_fastdump.py` A way to dump at 81.92us for 100MHz-BW mode data for searching
- `/users/sransom/astrid/GUPPI_astrid_slew+takedata.py` The `Slew()` and `Track()` example from above
- `/users/sransom/astrid/GUPPI_astrid_driftscan.py` The `Track()` for driftscans example from above



# Chapter 10

## K-band (18-27.5 GHz) Focal Plane Array (KFPA)

The K-band Focal Plane Array has seven beams total, each with dual circular polarization. Each beam covers the 18-27.5 GHz frequency range with fixed separations on the sky with nominal offsets listed in Table 10.1. The feeds have cooled polarizers producing circular polarization. The only internal switching mode is frequency switching. The seven feeds are laid out in a hexagon with one central feed as shown in Figure 10.1. The hexagon is oriented such that the central feed is not at the same cross-elevation or the same elevation as any of the other beams. Beam pairs (3,7) and (4,6) are at equal elevations and are appropriate choices for nodding and peak/focus observations.

Unlike other receivers, the KFPA uses variable noise diodes for each beam (~10% of the system temperature) that can change, so it is very important for observers to calibrate their data every session. The maximum instantaneous bandwidth for the receiver is currently 1.8 GHz. An experimental “broad-band” mode allowing 7.5 GHz of instantaneous bandwidth is also available for single beam observations using either beam 1 or 2.

Figure 10.1: Orientation of the KFPA feeds on the GBT

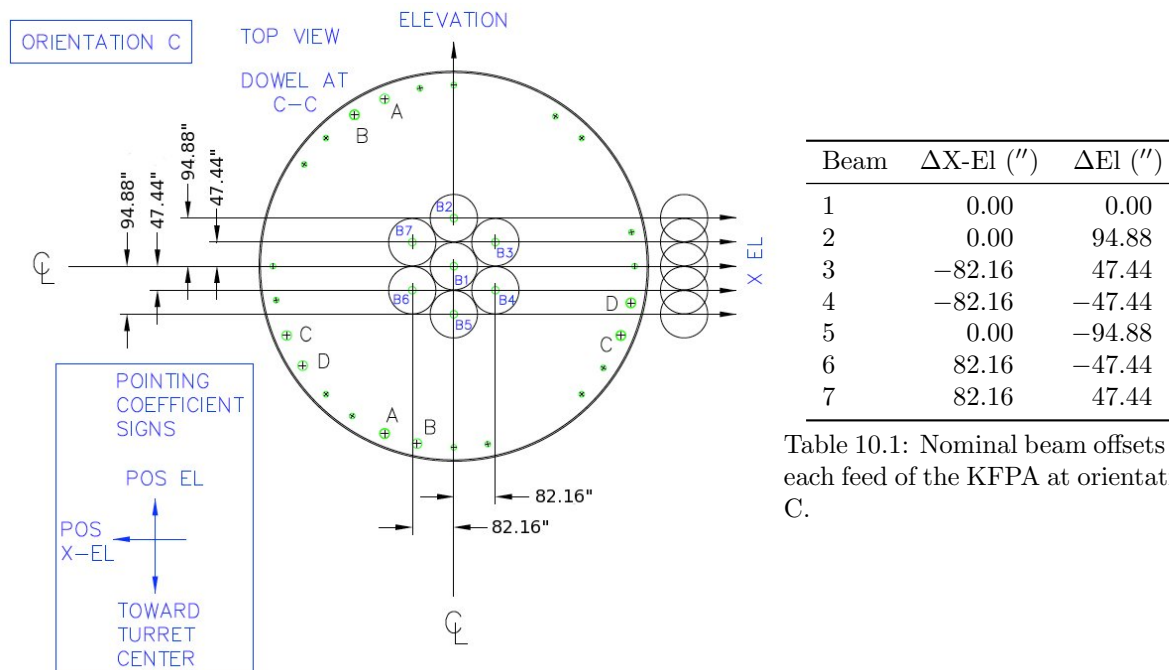


Table 10.1: Nominal beam offsets for each feed of the KFPA at orientation C.

## 10.1 Configuration

### 10.1.1 Beam selection with VEGAS banks

Observers can use all 7 KFPA beams simultaneously, or select a only subset of them. To maximize versatility and observing efficiency when mapping, most observers will want to use the full set of 7 beams paired with the observations with the VEGAS backend.

An equal number of beams must be routed to Converter Rack Modules A and B, imposing the following constraints on mapping beams to VEGAS banks:

- A single beam may be routed to any or all of the VEGAS banks A→H.
- Dual-beam configurations allow each beam to be routed to a maximum of 4 VEGAS banks.
- When using 3–4 beams, each beam may be routed to up to a maximum of 2 VEGAS banks.
- When using more than 5 beams, each beam may only be routed to a single VEGAS bank.
- When using all 7 beams, each beam may be routed to a single VEGAS bank with an optional second copy of beam 1 being routed to the remaining VEGAS bank. This is known as the “7+1” mode of the KFPA.

An example configuring for the 7+1 mode of the KFPA is given in Script 6.9.

### 10.1.2 Instantaneous Bandwidth

- **Narrowband mode** is the default setting and must be used with any KFPA multi-beam configuration. In this mode all frequencies must have a maximum frequency separation of 1.8 GHz and lie within 900 MHz of the doppler tracking frequency.
- **Broadband mode** mode allows the system to process up to 7.5 GHz of bandwidth. This mode is only available for single beam observations using either beam 1 or beam 2 and is achieved by setting `broadband=1` in the configuration (see § 6.2.5).

## 10.2 Calibration

The KFPA receiver has variable noise-diodes that can change so it is important that users calibrate their data for every session. The diodes have effective temperatures that can drift slowly over time or change after a thermal cycle of the cryostat.

We currently recommend that users calibrate their data by carrying out **Nod/OnOff** observations on a known flux calibrator using the same configuration that they would use for their science observations. An example SB for performing this type of calibration using all 7 beams of the KFPA is shown in Script 10.1.

It is also possible to use the sky at different elevations to “flat-field” the relative gains of the beams and then absolute calibrate the data with an **OnOff/Nod** observation of one reference beam using a known flux calibrator. Calibration using resolved sources such as planets or the Moon adds the requirement for an accurate temperature model of the source plus a model for coupling the GBT beam to the planet disk. Observers wishing to use such methods should consult the “Friend” of their project to devise the best strategy for calibration.

```

execfile('/home/astro-util/projects/GBTog/configs/kfpa_config.py')
Catalog(fluxcal)
Catalog(kband_pointing)
src = '3C286' # Do not use extended 3C sources

Configure(kfpa_config) # Configure for KFPA receiver
AutoPeakFocus(src) # Automatically slews, balances,
# and configures for continuum.

Configure(kfpa_config) # Reconfigure for VEGAS + KFPA using
Slew(src) # the same configuration you would use
Balance() # for your science observations....

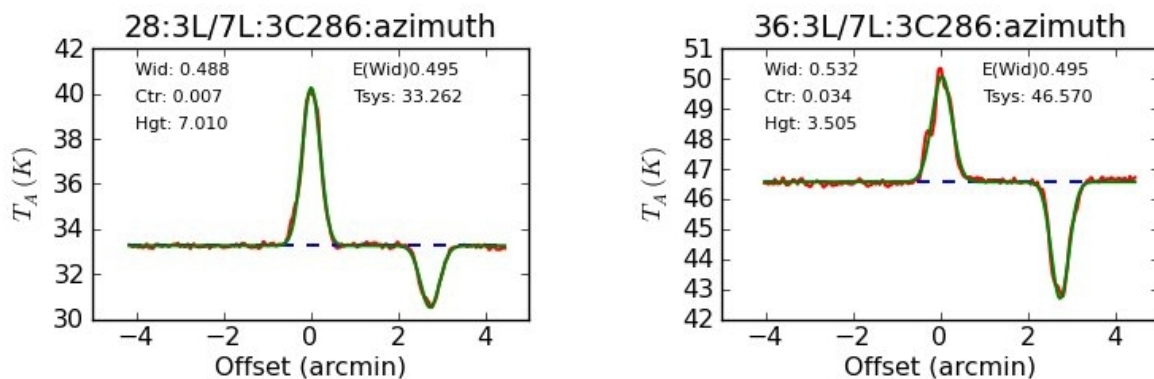
Nod(src, '3', '7', scanDuration=30.0) # This combination covers all
Nod(src, '2', '6', scanDuration=30.0) # 7 beams - edit as necessary
Nod(src, '4', '1', scanDuration=30.0) # for other beam configurations.
Nod(src, '1', '5', scanDuration=30.0)

```

Script 10.1: An example SB used to calibrate KFPA noise diodes for all 7 beams.

### 10.2.1 Realigning the Noise Diodes

Sometimes the effective temperature of the noise diodes can suddenly change by significant amounts due to firmware configuration glitches that may occur after resetting the manager, updates to receiver parameters, and power failures. A large change the effective temperature of a noise diode could result in unusual  $T_{sys}$  values or a large difference between the amplitude of beams in azimuth Peak scans as shown in Figure 10.2a.



(a) KFPA azimuth pointing scan where noise diode temperatures have drifted. The amplitude of beam 3 is approximately 3 times that of beam 7.

(b) KFPA azimuth pointing scan after realigning the noise diodes. Beam amplitudes are approximately equal.

Figure 10.2: KFPA azimuth pointing scans before and after noise diode realignment

If you suspect that the noise diode temperatures have drifted and need to be aligned, let the GBT operator know to contact the on-duty support scientist who can examine the data and realign the noise diodes if necessary. Figure 10.2b shows a repeat of the scan shown in Figure 10.2a after the noise diodes were realigned. Instructions for realigning the noise diodes can be found at <https://safe.nrao.edu/wiki/bin/view/Kbandfpa/OperationsResources>





# Chapter 11

## The 4mm (68-92 GHz) Receiver

### 11.1 Overview

The 4 mm receiver (“W-band”) is a dual-beam, dual-polarization receiver which covers the frequency range of approximately 67-93 GHz. The performance degrades significantly outside of this range. After the amplifier upgrade in the fall of 2012, typical system temperatures are under 100 K over much of the band (Figure 11.1). A key difference between the 4 mm receiver and other GBT receivers is that there are no noise-diodes for the 4 mm receiver. This impacts the observing and calibration techniques for the receiver. Users need to take a calibration sequence whenever the configuration changes or whenever the IF system is balanced for any data that needs to be calibrated. The receiver has an optical table with an ambient and cold load that is used for calibration (Figure 11.2) . The optical table can also convert linear polarization into circular polarization using a 1/4-wave plate in front of one of the beams for VLBI observations. The two beams are separation by 286'' in the cross-elevation direction on the sky (i.e., along azimuth).

In this chapter, we present information for carrying out W-band observations. We concentrate on items specific to W-band, and assume the user is familiar with the other chapters of the observing guide. Contact your support astronomer if you have questions.

### 11.2 Configuration

The 4mm Receiver uses the standard config-tool software that automatically configures the GBT IF system based on user input (e.g., frequency and bandwidth). Example w-band configuration files are given in `/home/astro-util/projects/4mm/`. The 4mm system is broken into into four separate bands:

- FL1: 67-74 GHz
- FL2: 73-80 GHz
- FL3: 79-86 GHz
- FL4: 85-93 GHz,

You can only use one of these bands at a time (i.e., you cannot simultaneously observe lines in more than one band). The millimeter down-converter filters of the system limits the instantaneous bandwidth to 4 GHz for 73-93 GHz (filters FL2, FL3,FL4), while up to 6 GHz of total bandwidth is available for 67-74 GHz (filter FL1).

The configuration items specific to the 4mm receiver are the following:

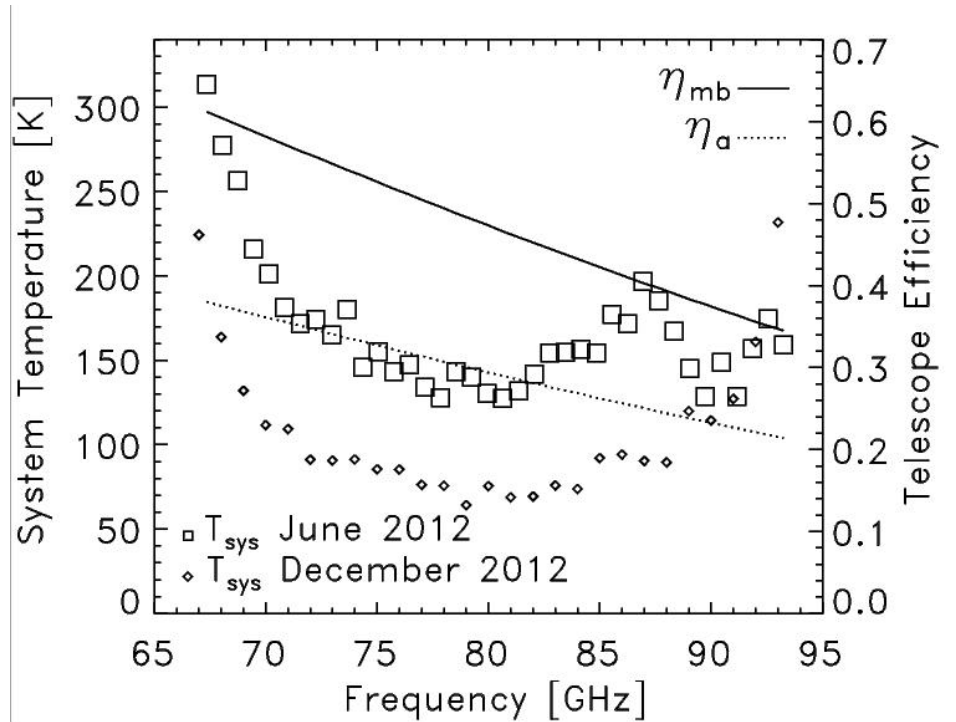


Figure 11.1: The derived main-beam ( $\eta_{mb}$ ) and aperture ( $\eta_a$ ) efficiencies from 2012 are given by the solid and dotted line respectively. These estimated efficiencies are based on measurements made after an AutoOOF, and the  $\eta_a$  values are slightly less than expected (measurements correspond to surface errors of  $280\mu\text{m}$  instead of the advertised  $240\mu\text{m}$  errors for the dish). The squares and diamonds show the measured system temperatures for the original amplifiers (2012 June) and the upgraded amplifiers (2012 December) respectively. Both sets of data were collected under similar conditions with  $\tau_o \simeq 0.2$  at 86 GHz.

- receiver = 'Rcvr68.92' (name of the receiver)
- beam = 'B12', 'B1', or 'B2' (dual beam receiver)
- swmode = "tp\_nocal" or "sp\_nocal". There are no noise diodes with this receiver.
- polarization = "linear" or "circular" (Default is linear). If user selects circular, then the 1/4-wave plate is placed in front of the chosen beam. There is only one 1/4-wave plate, so users can have circular polarization for only one of the beams.

## 11.3 Observing

In general, observations should be carried out during the night under stable thermal conditions to maximize the observing efficiency for targets smaller or similar in size to the beam ( $\sim 10''$ ). During the daytime, the effective point-source aperture efficiency decreases significantly since the beam shape increases in size. Depending on the science goals, successful daytime observations are possible for extended sources.

- Start the project with an AutoOOF (unless observing extended sources during the day). This sets the active surface, including the thermal corrections, as well as getting initial pointing and focus corrections.

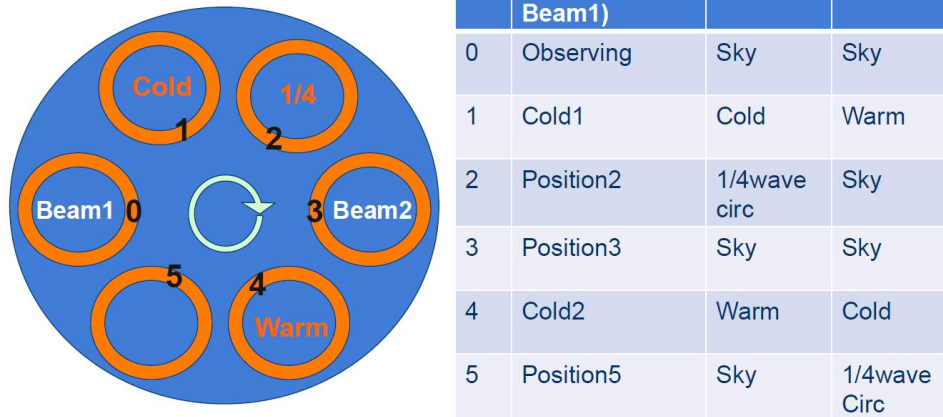


Figure 11.2: Diagram showing the positions of the 4mm Calibration wheel. The wheel is rotated above the cryostat, and the location of the beams are separated by 180 degrees on the wheel. In the Observing position, both beams see the sky. In the Cold1 position, beam-1 sees the cold load and beam-2 sees the ambient load, while for the Cold2 position, beam-2 sees the cold load and beam-1 sees the ambient load. The 1/4-wave plate can be placed over only one of the beams at a time.

- After configuration and balancing, check the RF power levels in the IF rack to confirm that power is going through the channels and that the power levels are not saturated ( $< 10$ ). Beam-1 uses channels 1 and 3, and beam-2 uses channels 5 and 7. To avoid possible saturation on the warm load, the power levels on the sky need to be at  $< 6$ . Although reasonable looking data may be detected downstream, observers should avoid non-linear calibration effects when the IF power saturates (RF Power = 10), and special care should be taken in calibration of such data. The target power level for w-band is 1.5, and the software adjusts the attenuation to reach this level. If the channel(s) are saturated with maximum attenuation (31dB), users can decrease the bandwidth of the observations (e.g., avoid using “All Pass” or choose a smaller filter to decrease the power levels). If the channel(s) do not have enough power with the no attenuation (0dB), users can increase the filter bandwidth or choose “All Pass” to increase the power levels. In general, manual changes of the default configuration values for the IF rack should not be needed for most configurations.
- Users must run the CalSeq procedure to calibrate the data (see § 11.3.1). During the calibration sequence, users can watch the movement of the calibration wheel from the Rcvr68\_92 cleo page (Figure 11.3).

### 11.3.1 CalSeq

The CalSeq procedure is used to calibrate w-band data. This procedure should be run after every configuration and balance. This is needed to convert instrumental voltages into antenna temperatures. The syntax for the Calseq command is the following:

**CalSeq**(type, scanDuration, [location, beamName, fixedOffset, tablePositionList, dwellFraction-List]), where the arguments in [ ] are optional.

- type: string keyword to indicate type of calibration scan: manual, auto, autocirc

- "manual" – A separate scan will be done for each table position. The user can input a list of calibration table wheel positions with the tablePositionList argument.
  - "auto" – default dwell fraction =(0.33, 0.33, 0.34) and default three positions = (Observing, Cold1, Cold2). The user can specify a list of positions and dwell times with the tablePositionList and dwellFractionList arguments.
  - "autocirc" – dwell fraction =(0.25, 0.25, 0.25, 0.25) and four positions = (Observing, [Position2 for beamName='1' or Position5 for beamName='2' for use with circular VLBI observations], Cold1, Cold2).
- scanDuration: scan exposure time, in seconds. For manual mode, each specified position will be observed for the scan exposure time (i.e., separate scans for each position). For auto modes, the total scan exposure time will be divided between positions based on the dwell fractions (i.e., one scan for all positions).
  - location: a Catalog source name or Location object; default is None (use current location).
  - beamName: Beam name associated with pointing location. This argument is a string. Default beam is '1'.
  - fixedOffset: offset sky position (used in cases when observing a bright source and want to measure the system temperature of the sky off-source). This argument should be an Offset object. Default sky offset is 0.
  - tablePositionList: user-specified, variable-length ordered list of cal table positions for the manual or auto modes. The default sequence is ['Observing', 'Cold1', 'Cold2'].
  - dwellFractionList: user-specified, ordered list of dwell fractions associated with the tablePositionList for use only with the auto mode. By using auto mode with tablePositionList and dwellFractionList, expert users can control the wheel in any order of positions and dwell fractions. This input not needed for autocirc or manual modes is ignored in these modes if given.

Example usage of the CalSeq command:

- CalSeq("auto",40.0). This command can be used for most observations, which uses the default tablePositionList=['Observing', 'Cold1', 'Cold2'].
- CalSeq("auto",40.0,fixedOffset=Offset("J2000", "00:00:00", "00:02:00")). This command can be used for bright objects where one wants a system temperature measurement on blank sky. In this example, the offset is 2' to the north. If observing a large object, one can increase the offset size to move off-source for the blank-sky measurement.
- CalSeq("manual",10.0,tablePositionList=['Position2', 'Observing', 'Cold1', 'Cold2']). This is an example command for calibration of VLBI observations with beam-1 circular polarization. We can only observe the cold and ambient loads with linear polarization. The calibration from linear to circular requires observations of the same sky with both linear and circular polarization (Observing and Position2, respectively, in this example).

### 11.3.2 Pointing and Focus

Blind pointing at the start of the observing run may not be successful since the blind pointing errors can be similar to the beam size, and the source may be missed in the simple Az-El scans used by the Peak procedure. Initial pointing offsets can be found with the AutoOOF procedure, or users may want to point on Jupiter or another large source as needed. Pointing and focus for W-band requires special attention, and users should not blindly accept the default solutions provided by the software system. Users can enter solutions manually as needed as discussed in Section 5.1.3.4.

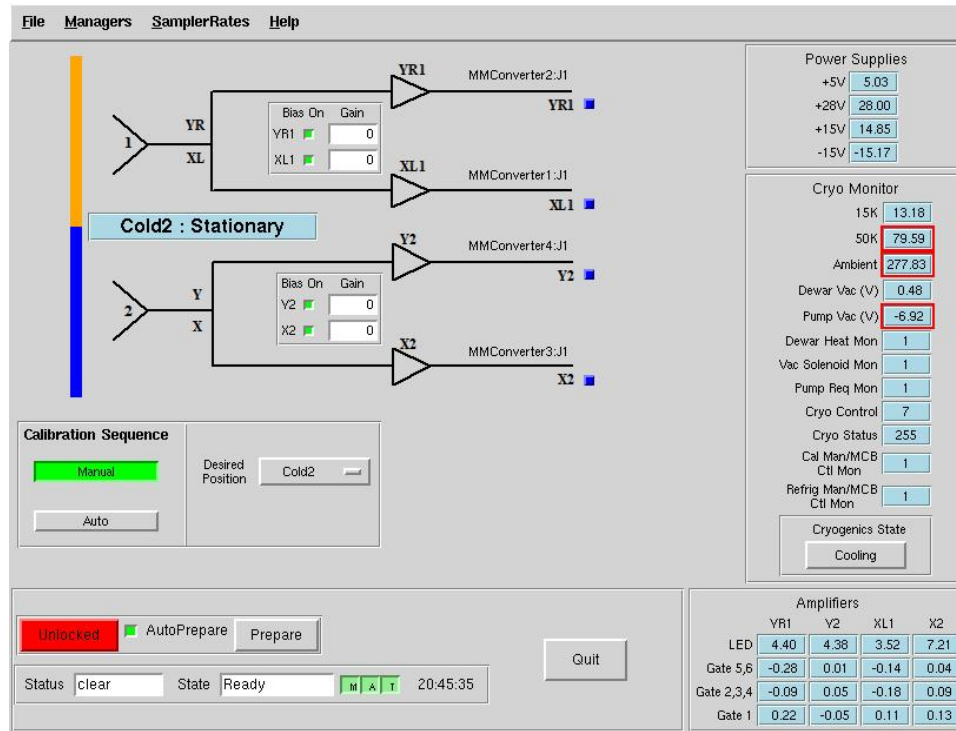


Figure 11.3: The 4mm Receiver CLEO window. Users can manual move the wheel by using the “Desired Position” button. The temperature of the ambient load used for calibration is given by the “Ambient” temperature sensor value shown at the right (277.83 K in this example).

### 11.3.3 AutoOOF Thermal Corrections

Optimal point-source observations should be carried out with regular AutoOOF measurements (every 2-4 hours) during the nighttime when the thermal stability of the dish is best. Based on commissioning tests with the 4mm receiver, the AutoOOF corrections improves the point-source aperture efficiency by 30–50%. Application of these corrections during the day are typically not practical for the 4mm receiver given the thermal environment of the dish is generally not sufficiently stable. During the day, the measured beam sizes can vary significantly (e.g., 10–14”), but the beam shape typically remains well-behaved (fairly symmetric and Gaussian). Although the variation of beam size has a direct impact on the point-source aperture efficiency ( $\eta_a$ ), it has less of an impact on the effective main-beam efficiency ( $\eta_{mb}$ ) used for the calibration of extended sources. For example during the commissioning of the instrument, we measured a beam size of 10.8” at 77 GHz and derived  $\eta_a = 31\%$  and  $\eta_{mb} = 50\%$  in good nighttime conditions with the application of the thermal AutoOOF corrections. Without the AutoOOF corrections, the beam-size increased to 12.5” and the aperture efficiency decreased to 23%, but the main-beam efficiency remained nearly constant at about 50% ( $\eta_{mb} \propto \theta_{mb}^2 \eta_a$ ). Therefore, extended sources may be observed during the day without the AutoOOF corrections if the science is not impacted by the primary beam variations.

## 11.4 Calibration and Data Reduction

For accurate calibration, users are recommended to run a CalSeq before and after each set of source data. Users are also recommended to take a short nod observation on their pointing & focus source every hour to track the relative efficiency of the system. If users only want absolute calibration done to within 20–30%, then the default calibration from the cold&ambient loads may be used. For more

Table 11.1: 4mm Channel Definitions

Channel	Polarization	Beam
ch1	beam1 (fdnum=0)	X or L (plnum=0)
ch3	beam1 (fdnum=0)	Y or R (plnum=1)
ch5/6	beam2 (fdnum=1)	X or L (plnum=0)
ch7	beam2 (fdnum=1)	Y or R (plnum=1)

**Table Notes:** The GBT IF channel numbers 1,3,5,7 and their corresponding beam and polarization definitions. The parameters fdnum and plnum are GBTIDL keywords.

accurate absolute flux calibration, a source of known flux density should be used. The ALMA Calibrator Source Catalogue has an extensive record of the flux density histories for many of the bright 3mm point sources (<https://almascience.eso.org/sc/>). By using ALMA flux density values as a function of time, ~10% absolute calibration uncertainties can be obtained for W-band data.

The standard GBTIDL scripts (getps, getnod, getfs) do not work since these assume a noise diode for calibration. W-band scripts for the reduction of spectral line data can be found at /home/astro-util/projects/4mm/PRO. For continuum (DCR) reduction, contact your support scientist. Users can use the calseq.pro within GBTIDL to derive the gains for each of the channels. After deriving the gains, users can and reduce the spectra-line data, for example, using wonoff\_gain.pro.

The formulae for calibrating data are given below. The observed antenna temperature is

$$T_A = T_{\text{sys}}(V_{ON} - V_{OFF})/V_{OFF}, \quad (11.1)$$

where  $V_{ON}$  and  $V_{OFF}$  are the observed voltages of the ON and OFF scans. The system temperature ( $T_{\text{sys}}$ ) is given by

$$T_{\text{sys}} = g V_{OFF}, \quad (11.2)$$

and the gain ( $g$ ) is

$$g = [(T_{\text{amb}} - T_{\text{cold}})/(V_{\text{amb}} - V_{\text{cold}})], \quad (11.3)$$

where  $T_{\text{amb}}$  and  $T_{\text{cold}}$  are the temperatures of the ambient and cold loads and  $V_{\text{amb}}$  and  $V_{\text{cold}}$  are the observed voltages of the ambient and cold load scans. The ambient load is given on the receiver CLEO page and within the receiver FITS files. The cold load is measured in the lab (Table 18.2). The main-beam temperature  $T_{mb}$  is related to the observed antenna temperature  $T_a$  by

$$T_{mb} = T_A \exp(\tau_o A)/\eta_{mb}, \quad (11.4)$$

where  $\tau_o$  is the zenith opacity and  $A$  is the airmass. Assuming a Gaussian beam, the main-beam efficiency is given by

$$\eta_{mb} = 0.8899\eta_a(\theta_{FWHM}D/\lambda)^2, \quad (11.5)$$

where  $\theta_{FWHM}$  is the full-width half-maximum beam size in radians,  $D$  is the diameter of the GBT (100m), and  $\lambda$  is the observed wavelength. For the GBT, the effective aperture efficiency is

$$\eta_a = 0.3516T_A \exp(\tau_o A)/S_\nu, \quad (11.6)$$

where  $S_\nu$  is the flux density of a known calibration source.

The opacity as a function of time can be obtained by the weather database derived for Green Bank using the getatmos.pro script. Except for periods of rapidly changing weather conditions, the predicted opacities are accurate to within  $\Delta\tau_o \simeq 0.01$  based on historical measurements.

Table 11.2: Effective Cold Load Temperature for W-band

Date Range	Average Temperature
2010 May – 2012 April	53 K
2012 May – 2012 June	56 K
2012 October – 2015 August	50 K
2015 September –	54 K

**Table Notes:** The estimated effective cold load temperature measured in the lab. The temperature is estimated to better than  $\pm 5$  K. Note, that a 10% error in the cold load measurement corresponds to only about a 2.5% error in the measured  $T_a$  or  $\eta_a$ .

## 11.5 Web Documentation

- 4mm Web Page: <http://www.gb.nrao.edu/4mm/>
- 4mm Project Book: <http://www.gb.nrao.edu/4mm/ProjectBook/>
- 4mm Wiki: <https://safe.nrao.edu/wiki/bin/view/GB/Gbt4mmRx>
- 4mm Status and Commissioning Wiki which provides latest info on performance and notes for users: <https://safe.nrao.edu/wiki/bin/view/GB/Gbt4mmRxCommissioning>





## Chapter 12

# The CalTech Continuum Backend (CCB)

The Caltech Continuum Backend (CCB) is a dedicated continuum backend for the GBT Ka-band receiver, built in collaboration with A.C.S. Readhead’s radio astronomy instrumentation group at Caltech and commissioned on the GBT in 2006. The driving consideration behind its design is to provide fast electronic beam switching in order to suppress the electronic gain fluctuations which usually limit the sensitivity of continuum measurements with single dish radio receivers. To further improve stability, it is a *direct detection* system: there are no mixers before the conversion from RF to detected power. The Ka-band receiver provides *eight* simultaneous, directly detected channels of RF power levels to the CCB: one for each feed, times four frequency channels (26-29.5 GHz; 29.5-33 GHz; 33-36.5 GHz; and 36.5-40 GHz). Astronomical information and labels for these 8 channels (or “ports” in GBT parlance) is summarized in Table 12.

The following sections outline the process of observing with, and analyzing the data from, the CCB. Much of the information in this chapter is also maintained at

`/users/bmason/ccbPub/README.txt`

which is convenient, for instance, for cutting and pasting data analysis commands. Template scheduling blocks are also in this directory.

## 12.1 Observing with the CCB

Table 12.1: CCB Port labels and the astronomical quantities they measure.

Port	Beam	Polarization	Frequency
9	1	Y	38.25
10	1	Y	34.75
11	1	Y	31.25
12	1	Y	27.75
13	2	X	38.25
14	2	X	34.75
15	2	X	31.25
16	2	X	27.75

### 12.1.1 Configuration

Configuration of the CCB is straightforward, and for most purposes the only two configurations needed are provided in the two configuration files

```
/users/bmason/ccbPub/ccb.conf
```

and

```
/users/bmason/ccbPub/ccbBothCalsLong.conf.
```

These differ only in the duration of the integrations: the former configures for 5 millisecond integrations, which is useful for estimating the scatter in the samples to obtain meaningful  $\chi^2$  values in the analysis of science data; the later configures for 25 millisecond integrations, which is useful in peak/focus observations to speed up processing of the data (see Chapter 5, section 5.1.3). `ccb.conf` is reproduced and explained below.

The following keywords tell ASTRID to expect beamswitched continuum observations with Ka and the CCB:

```
receiver='Rcvr26_40'
beam='B12'
obstype='Continuum'
backend='CCB'
nwin=4
restfreq=27000, 32000, 35000, 38000
deltafreq=0, 0, 0, 0
bandwidth=600, 600, 600, 600
swmode='sp'
swtype='bsw'
pol='Circular'
vdef='Radio'
frame='topo'
```

They do not have any practical effect on the actual instrument configuration but are necessary to set up internal variables and ensure the recorded FITS files are accurate.

```
ccb.cal_off_integs=20
ccb.XL_on_integs=2
ccb.both_on_integs=2
ccb.YR_on_integs=2
ccb.bswfreq=4
tint=0.005
```

The meaning of these keywords is as follows:

- The first four specify the cal firing pattern.
- `ccb.bswfreq` specifies the beam switching frequency in kHz. 4 kHz is standard; the “> 10% blanking” warning which results is also standard and may be safely ignored.
- `tint` is the integration time in seconds.

## 12.1.2 Pointing & Focus

The online processing of pointing and focus data is handled by GFM (which runs within the Astrid Data Display window) similarly as for other GBT receivers and the DCR. A few comments:

- because the Ka-band receiver currently only has one polarization per beam, GFM will by default issue some complaints which can be ignored. These can be eliminated by choosing “Y/Right” polarization in the Astrid Data Display window (see Chapter 5, section 5.1.3.2) under Tools → Options → Data Processing.
- in the same menu ( Tools → Options → Data Processing), choosing “31.25 GHz” as the frequency to process, instead of the default 38.25 GHz, can improve robustness of the result.
- The results shown in the Astrid Display are in *raw counts*, not Kelvin or Janskys.
- Choosing “Relaxed” heuristics is also often helpful.

There is a template pointing and focus SB for the CCB in `/users/bmason/ccbPub` called `ccbPeak.turtle`. This scheduling block does a focus scan, four peak scans, and a symmetric nod (for accurate photometry to monitor the telescope gain).

## 12.1.3 Observing Modes & Scheduling Blocks

Science projects with the CCB typically fall into two categories: mapping, and point source photometry. The majority of CCB science is the latter, since this is what, by design, it does best. Template scheduling blocks for both are in `/users/bmason/ccbPub`.

**Observers and support scientists are strongly encouraged to use these template scheduling blocks as the basis for their CCB observing scripts and make only the changes that are required! Relatively innocuous changes can make the data difficult or impossible to calibrate with existing analysis software.**

The basic template SBs are:

- `ccbObsCycle.turtle`: perform photometry on a list of sources.
- `ccbRaLongMap.turtle`: perform a standard RALongMap on a source (see 6.4.3).
- `ccbMap.turtle`, `ccbMosaicMap.turtle`: make maps using longer, single-scan, custom raster maps. Your staff friend will help tailor these to your project’s needs, should you choose this approach.

Point source photometry is accomplished with an OTF variant of the symmetric NOD procedure described in 6.4.2. This procedure, which we refer to as the OTF-NOD, alternately places the beam in each of the two beams of the Ka-band receiver in a B1/B2/B2/B1 pattern. This sequence cancels means and gradients in the atmospheric or receiver emission with time. Plotting the beamswitched data from this sequence produces a sawtooth pattern shown in Figure 12.1; this is discussed more in § 12.1.5. Each NOD is 70 seconds long (10 seconds in each phase, with a 10 second slew between beams and an initial 10 second acquire time).

Note: OTF-NOD is not one of the standard scan types; it is implemented in the scripts mentioned here (e.g., “`ccbObsCycle.turtle`”).

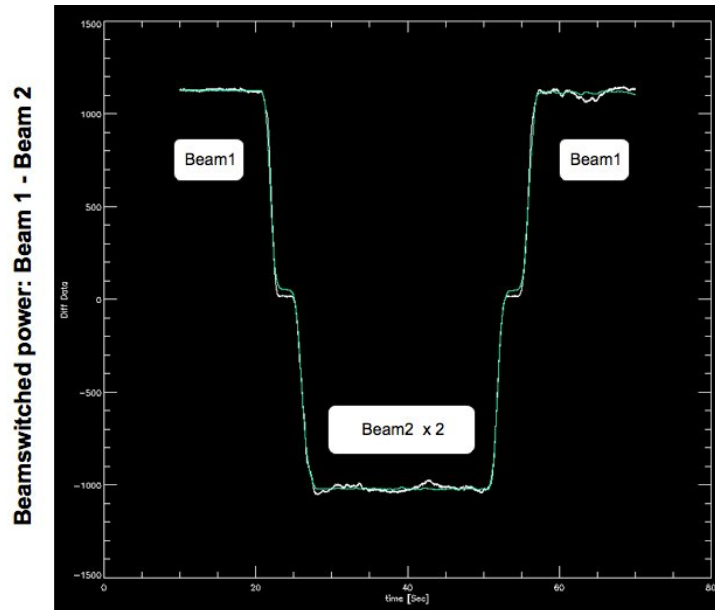


Figure 12.1: Data from a CCB, beamswitched OTF-NOD, showing data and model versus time through one B1/B2/B2/B1 scan. The white line is the CCB beamswitched data and the green line is the fit for source amplitude using the known source and telescope (as a function of time) positions.

### 12.1.4 Calibration

If at all possible, be sure to do a peak and focus, and perform photometry (an OTF-NOD, as implemented in `ccbObsCycle.turtle` or `ccbPeak.turtle`) on one of the following three primary (flux) calibrators: 3c48, 3c147, or 3c286. This will allow your data to be accurately calibrated (our calibration scale is ultimately referenced to the WMAP 30 GHz measurements of the planets). If this is not possible the calibration can be transferred from another telescope period (observing session) within a few days of the session in question.

### 12.1.5 Online Data Analysis

It is important to assess data quality during your observing session. There are a set of custom IDL routines for analyzing CCB data; if you use the observing procedures and config files described here, your data should be readily calibratable and analyzable by them. To use the IDL code, start IDL by typing (from the GB UNIX command line)

```
/users/bmason/ccbPub/ccbidl
```

Here is an example data reduction session that provides a quick look at your data:

```
; set up global variables
; don't write files or plots to disk...
proj='AGBT06A_049_09'
setccbpipelineopts , gbtproj=proj , ccbwritefiles=0, \$
  gbtdatapath='/home/archive/science-data/tape-0016/'
; to use postprocessing scripts , set ccbwritefiles=1

; a good color table for the plots:
loadct,12
```

```

; create an array indexing scan numbers
; to file name
indexscans , si

; summarize the project
summarizeproject

; read a nod observation from scan 12
readccbotfnod , si [12] , q
; fit the data, binning integrations to 0.5sec bins
fitccbotfnod , q , qfit , bin=0.5
; the resulting plot shows the differenced
; data (white) and the fit to the data (green)
; for each of 16 CCB ports. (the first 8 are blank)

; look at the next nod that just came in
; this time calibrate to antenna temperature
; before plotting
; First you need to derive a calibration , which
; requires a scan with both calcs firing independently.
; /dogain tells the code to solve for the calibration;
; the results are stored in calibdat , which we can
; pass into subsequent invocations of the calibration.
indexscans , si
readccbotfnod , si [13] , q
calibtokelvin , q , /dogain , calibdat=calibdat
fitccbotfnod , q

; the scan index si must be updated to read in scans
; collected after it was first created
indexscans , si
readccbotfnod , si [14] , q
; and calibrate to kelvin using the information
; we just derived
calibtokelvin , q , calibdat=calibdat
; fit/plot
fitccbotfnod , q

; et cetera...

```

Example OTF-NOD data for bright sources (under good and poor conditions) and a weak source (under good conditions) are shown in Figures 12.2 through 12.5

Mapping data can also be imaged using the IDL tools:

```

; make a map from scans 7–10 using port 11 data
; (note the port must be specified; valid ports are
; 9–16)
img=makedcrrccbmap ([7,8,9,10] , /isccb , port=11)
; replot the map
plotmap , img , /int
; make a png copy of it
grabpng , 'mymap.png'
; save the map in standard FITS format—

```

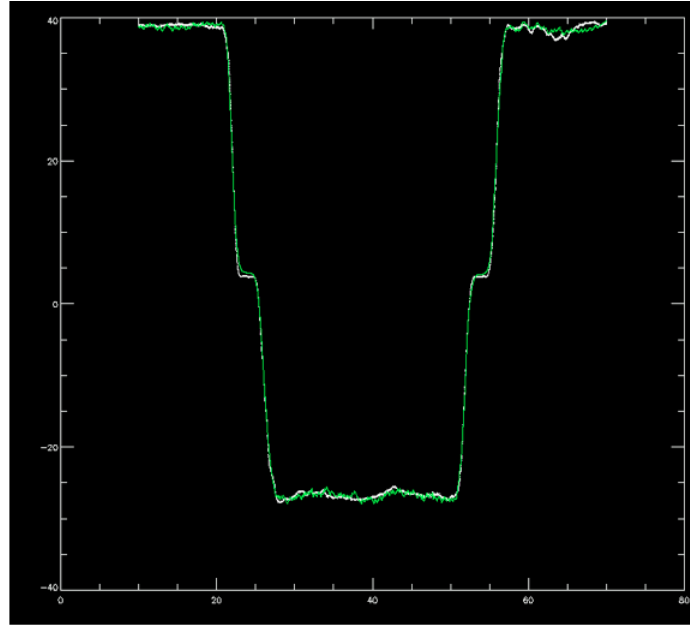


Figure 12.2: CCB data from an OTF-NOD observation of a bright source, showing data and model versus time through one B1/B2/B2/B1 scan. The white line is the CCB beamswitched data and the green line is the fit for source amplitude using the known source and telescope (as a function of time) positions. The close agreement between the data and the fit indicate that neither fluctuations in atmospheric emission nor pointing fluctuations (typically due to the wind on these timescales) are problems in this data.

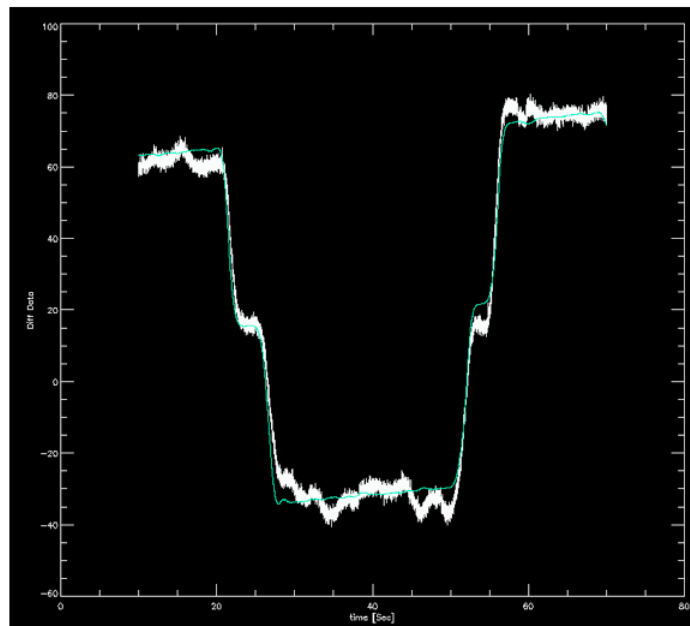


Figure 12.3: CCB OTF-NOD data on a bright source under marginal conditions. The differences between the data and the model are clearly larger in this case.

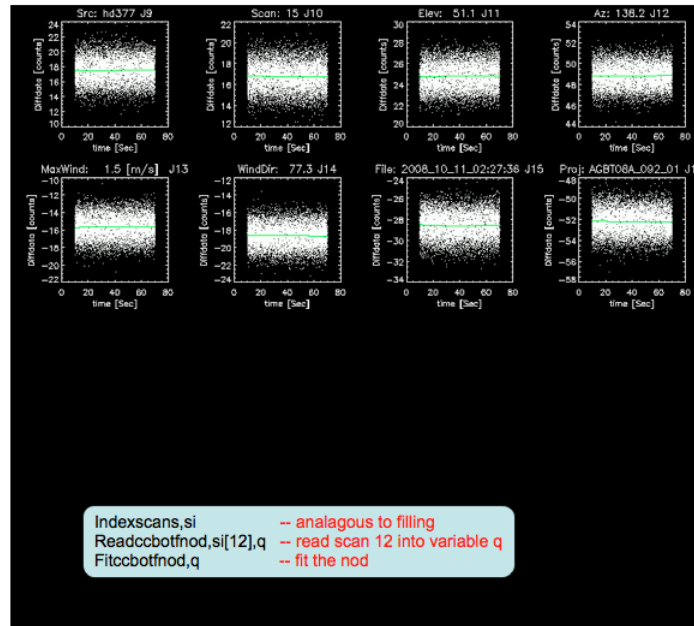


Figure 12.4: CCB OTF-NOD measurement of a weak (mJy-level) source under good conditions. The IDL commands used to obtain this plot are shown inset.

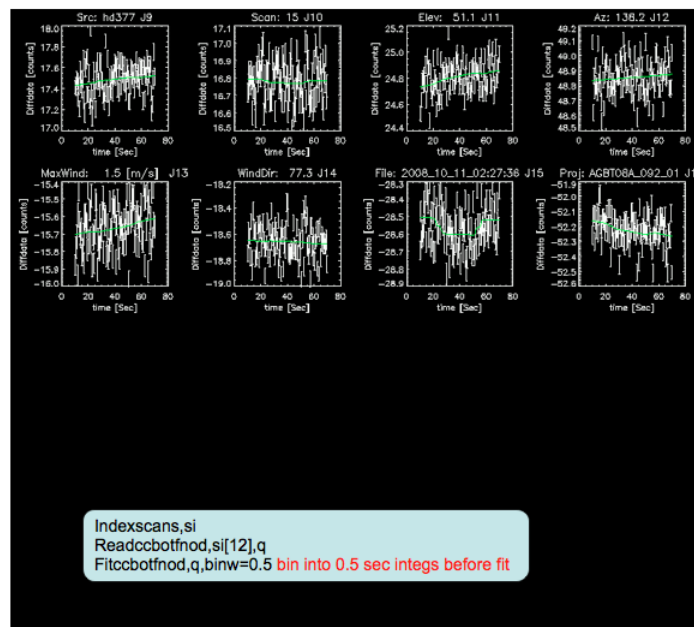


Figure 12.5: The same weak-source data, this time with the individual integrations binned into 0.5 second bins (using `fitccbotfnod`'s `binwidth` optional argument in seconds) so the thermal-noise scatter doesn't dominate the automatically chosen  $y$ -axis scale. This better shows any gradients or low-level fluctuations in the beamswitched data (due, for instance, to imperfect photometric conditions). In this data they are not significant.

```
saveimg ,img , 'mymap.fits '
```

This will be a *beamswitched* map. The beamswitching can be removed by an EKH<sup>1</sup> deconvolution algorithm also implemented in the code. Your GBT friend will help you with this, if needed.

## 12.2 Performance

Recent tests under excellent conditions show a sensitivity of  $150\ \mu\text{Jy}$  (RMS) for the most sensitive single channel (34 GHz), or  $100\ \mu\text{Jy}$  (RMS) for all channels combined together. These are the RMS of fully-calibrated, 70-second OTF-NODs on a very weak source. Typical “reasonable-weather” conditions are a factor of two worse. Improvements to the receiver made since these data were acquired may result in better sensitivity for the 2010/2011 season.

## 12.3 Differences Between the CCB/Ka System and other GBT Systems

There are a few differences between the CCB/Ka system and other GBT receiver/backend systems which users familiar with the GBT will want to bear in mind.

- Because it is a direct detection system, the GBT IF system does not enter into observing.
- The Ka/CCB gains are engineered to be stable (10% - 20% over months), so no variable attenuators are in the signal chain. Consequently there is no Balance () step.
- To optimize the RF balance (for spectral baseline and continuum stability), the OMT's have been removed from the Ka band receiver. It is therefore sensitive to *one linear polarization per feed*. The two feeds are sensitive to orthogonal linear polarizations (X and Y).
- Feed orientation is  $45^\circ$  from the Elevation/cross-Elevation axes. All other receivers have feed separations that are parallel to the Elevation or cross-Elevation axes (except for the KFPA).
- There are two cal diodes (one for each feed), and they are separately controlled (*i.e.*, it is possible to turn one on and not the other). Cals are ON or OFF for an entire integration; they are not pulsed ON and OFF within a single integration.

---

<sup>1</sup>Emerson, Klein, Haslam 1979 A&A **76**,92.



# Chapter 13

## VLBI Observing using the GBT

### 13.1 Proposals

The GBT has a VLBA-compatible data acquisition system. Proposals requesting GBT participation in VLBA or global VLBI observations should be submitted to the VLBA only, not to the GBT.

Proposals requesting the GBT participation in a Very Long Baseline (VLB) experiment that includes no other NRAO telescopes should be submitted to the VLBA as well as to the GBT and other agencies as appropriate, such as the European VLBI Network (EVN).

References for VLBA proposals: <https://science.nrao.edu/facilities/vlba/proposing>

General information about the VLBA: <https://science.nrao.edu/facilities/vlba/>

### 13.2 VLBA-compatible recording

The data acquisition system is similar to those at the VLBA stations: two Roach Digital Backend (RDBE) units and a Mark5C recorder are in use, allowing wide-band recording up to 2 Gbits/sec. Two modes are available, “PFB” mode provides 16 32-MHz channels and a total recording rate of 2 Gbits/sec. The “DDC” mode allows up to 4 channels of bandwidth 1 to 128 MHz. With two RDBEs available, up to 8 Digital Down Converter (DDC) channels may be used.

The SCHED default frequency setups should be correct for writing schedules for the new system.

The old data acquisition system with the DAR rack and Mark5A recorder has been retired. No proposals should request it.

## 13.3 Schedule Preparation

Scheduling is done through the VLBA analysts in Socorro.

- Schedules are prepared with the SCHED program.  
(refer to: <http://www.aoc.nrao.edu/software/sched/index.html>)
- The GBT uses the standard VLBA schedule files (\*.key and \*.vex files).
- The user needs to prepare a .key file for SCHED and send it to the VLBA analysts.
- Use **GBT\_VLBA** as the station name, except for cold weather in which case use **GBT\_COLD**. Refer to pointing and weather sections, below.
- In general, use **GBT\_COLD** during the months of December, January, and February.

The schedules, either .vex or .key files, are processed by the VLBA analysts to produce schedule scripts for each VLBA telescope, including one for the GBT. These scripts are interpreted at the GBT by a process called “RunVLBI” which generates the configuration and pointing commands for the GBT. The same script runs in the VLBA backend to drive the recording and backend frequency setup. The GBT telescope operator runs these experiments. The user does not need to know anything about GBT-specific script details, i.e, the Astrid configurations, catalogs, and scheduling blocks.

There are, however, several GBT-specific details which the user needs to take into consideration when designing the observing schedule. These are described in the next few sections.

## 13.4 Special considerations when using the GBT

- Allow about 30 minutes setup time at the beginning of a session before VLBI recording begins, except for the 3mm and 7mm receivers for which one hour setup time should be allowed.
- Changing between Gregorian receivers requires rotating the turret. The telescope operator initiates this rotation. At least 5 minutes should be allowed in the schedule to change from one Gregorian receiver to another.
- Changing between Gregorian and prime focus requires about 10 minutes; that is the time required to extend or retract the prime focus boom. Changing from one prime focus receiver to another requires about 4 hours, because one feed must be physically removed and replaced with another.
- The prime focus receivers include 50 cm and 90 cm bands; whereas L-band and all higher frequencies ( $\nu > 1.2$  GHz) use the Gregorian focus.
- One needs to include enough pointing/focus updates – see below.
- There are some weather-related restrictions – see below.

## 13.5 Available Receivers and Bands

The receivers and frequency bands are listed in table 13.1. Note that some bands are available on the GBT but not on the VLBA. Note also the time it takes to change receivers, as described above. For more information, consult

- The GBT proposers guide, chapter 4, for antenna and receiver performance.
- Gain curves, see <https://safe.nrao.edu/wiki/bin/view/GB/Observing/GainPerformance>

Table 13.1: VLBA bands and GBT receivers.

VLBA Band	VLBA Frequency Range (GHz)	GBT Frequency Range (GHz)	GBT Receiver	Net Side-band	Primary Beam FWHM	Est. SEFD (Jy)	Typical Tsys (K)
90 cm	0.312-0.342	0.290-0.395	Rcvr_342 (PF1)	lower	36'	25	20-70
—	—	0.385-0.520	Rcvr_450 (PF1)	lower	27'	22	20-50
50 cm	0.596-0.626	0.510-0.690	Rcvr_600 (PF1)	lower	21'	12	20-35
—	—	0.680-0.920	Rcvr_800 (PF1)	lower	15'	15	18-25
—	—	0.910-1.230	Rcvr_1070 (PF2)	lower	12'	10	18-22
18/21 cm	1.35-1.75	1.1-1.8	Rcvr1_2	lower	9'	10	15-18
13 cm	2.15-2.35	1.68-2.60	Rcvr2_3	lower	5.8'	12	18
6 cm	3.9-7.9	3.95-8.0	Rcvr4_6	lower	2.5'	10	23
4 cm	8.0-8.8	7.9-10.1	Rcvr8_10	lower	1.4'	15	27
2 cm	12.0-15.4	11.8-18.0	Rcvr12_18	upper	54''	20	27
1 cm	21.7-24.1	18.0-27.5	RcvrArray18_26	lower	32''	25	40
—	—	26.0-40.0	Rcvr26_40	upper	22''	20-40	40
7 mm	41.0-45.0	40.0-50.0	Rcvr40_52	upper	16''	60	80
3 mm	80.0-90.0	68-92	Rcvr68_92	upper	10''	100	110

### Notes:

- Receivers with “PF1” or “PF2” are at the prime focus; the others are at the Gregorian focus.
- Rcvr26\_40 has linear polarization only; 2 beams but one polarization state per beam; all other receivers can receive dual circular polarizations.
- Pulse Cal (or phase cal) is injected in receivers of 2 cm wavelength and longer; pulse cal is injected in the 7mm receiver after the first mix; other receivers have no pulse cal injection.
- The 3mm receiver (Rcvr68.92) has no noise cal or pulse cal injection. See the section below for how calibration is done.

## 13.6 Include Pointing and Focus Checks

It is recommended to allow for pointing and focus touch-ups when observing at the higher frequencies. Recommendations are listed in table 13.2.

Table 13.2: GBT pointing and focus checks with VLBA observations.

Frequency Band	Interval between pointing scans
4–10 GHz	4–5 hours
12–18 GHz	3–4 hours
18–26 GHz	1.5–2 hours
40–90 GHz	30–60 minutes

### Notes:

- The observer should select a strong continuum source (flux density  $> 0.5$  Jy, or  $> 1.0$  Jy for  $\nu > 20$  GHz).
- Allow about 6 minutes for the pointing/focus check, except for the 3mm receiver for which you should allow 8 minutes in order to include the temperature calibration.
- For observing at frequencies below 5 GHz, include one pointing scan at the beginning of the session.
- The telescope operator will usually do a point/focus scan at the beginning of an observing session, during the startup time.

To include a point/focus scan in your schedule, put commands into your `.key` file similar to the following:

```
comment='GBT pointing scan'
peak=1
stations=gbt_vlba
source= 'J0920+4441' dwell=06:00 vlbmode='VA' norecord /
nopeak
```

Script 13.1: Example additions to the `.key` file for GBT point and focus observations

It is important to specify only the GBT (`stations=gbt_vlba` or `stations=gbt_cold`) when putting in `peak=1`. Otherwise it may do a reference pointing for the whole VLBA, and if the pointing source is under about 5 Jy, it can produce bad results. Refer to the SCHED manual for details of schedule preparation at <http://www.aoc.nrao.edu/software/sched/index.html>

### 13.6.1 3mm Receiver (68-92 GHz) calibration

System Temperature ( $T_{sys}$ ) calibration with this receiver uses a calibration wheel that can place hot and cold loads in front of the feed. There is no noise injection as happens with the other receivers. A “cal sequence” procedure is done before and after each peak/focus to provide a  $T_{sys}$  measurement. A cal sequence is inserted automatically with the peak/focus; the user does not have to specify it explicitly. A cal sequence takes about one minute, and will happen before and after a peak/focus. The user should use a dwell time of 8 minutes for the pointing scan, and that will include the cal sequences. Pointing Sources for high frequency observing should be strong, i.e., stronger than 3 Jy if possible.

## 13.7 Weather Considerations

At the higher frequencies, windy conditions can degrade the pointing. Refer to recommended wind limits for observing at <https://safe.nrao.edu/wiki/bin/view/GB/PTCS/PointingFocusGeneralStrategy>

- For sustained winds of  $> 35$  MPH or gusts  $> 40$  MPH, the telescope is stowed for safety.
- Ambient temperature  $< 17\text{F}$  ( $-8.3^\circ\text{C}$ ) : the maximum azimuth slew rate is reduced to  $18^\circ/\text{min}$ .
- Ambient temperature  $< -10\text{F}$  ( $-23^\circ\text{C}$ ) : the antenna is shut down.

If your project will run in December, January, or February you should use the lower azimuth slew rate of  $18^\circ/\text{min}$  when making the schedule. This is accomplished by using `stations=gbt_cold` in your `.key` file, instead of `stations=gbt_vlba`.

## 13.8 Telescope Move times and limits

### Move Limits:

- Elevation:  $5^\circ \rightarrow 90^\circ$
- Azimuth:  $-90^\circ \rightarrow +450^\circ$ , i.e,  $180^\circ \pm 270^\circ$

### Calculating time to change sources:

- **Maximum Azimuth slew rate:**  $36^\circ/\text{min}$  ( $17^\circ/\text{min}$  at low temperature)
- **Maximum Elevation slew rate:**  $18^\circ/\text{min}$
- **Acceleration:**  $0.05^\circ\text{sec}^{-2}$
- **Overhead:** 20 seconds to settle
- Allow a minimum of 30 seconds for a source change, even for short moves.

## 13.9 High Frequency (40-90 GHz) active surface considerations

When using the 40-50 or 68-92 GHz receivers, one should tune up the active surface by doing an “AutoOOF” procedure. This is so-called “Out of focus holography” in which a strong point source is observed both in and out of focus, and large-scale deviations of the surface can be derived. The surface corrections are applied to the active surface model. This improves the aperture efficiency by a factor of 2 at 86 GHz. One should do an AutoOOF, which takes about 30 minutes, at the beginning of any high-frequency observing. The user does not have to specify this in the observing file; the operator or telescope friend will do an AutoOOF calibration prior to starting the observing, during the setup.

When observing with the 68-92 GHz receiver, one should repeat the AutoOOF about every 3-4 hours. This means that the user should allow a 30 minute gap in the schedule about every 3-4 hours. The user does not have to specify anything about an autoOOF in the schedule; just allow the 30 minute gap. The operator or telescope friend will do the calibration.

## 13.10 GBT Coordinates

The geodetic position for the GBT (as of Jan 2000), based on a local survey referred to a standard NGS survey marker on the Green Bank site in the NAD83 system is

- longitude =  $79^{\circ}50'23.406''W$
- latitude =  $38^{\circ}25'59.236''N$
- Height of Track: NAVD88 height: 807.43 m (wrt ellipsoid: 776.34 m)
- Height of elevation axle: NAVD88 height: 855.65 m (wrt ellipsoid: 824.55 m)

The surveyed height refers to the top of the azimuth track. The phase center (intersection of azimuth and elevation axes) is 48.22m above the top of the azimuth track. The average geoid height = -31.10m with respect to the ellipsoid. The estimate uncertainty is 0.04".

The Earth-centered International Terrestrial Reference Frame (ITRF) coordinates for the phase center of the GBT were derived from a "TIES" run with the GBT and 20-meter telescopes in December 2002. Geodetic solution for the ITRF coordinates may be found through the web site:

<http://gemini.gsfc.nasa.gov/solutions/>

The solution as of Oct 2007 is:

$$\begin{aligned}x &= 882589.638 \text{ meters} \\y &= -4924872.319 \text{ meters} \\z &= 3943729.355 \text{ meters}\end{aligned}$$

Based on the ITRF solution, the best NAD83 geodetic position is:

$$\begin{aligned}\text{Latitude} &= 38^{\circ}25'59.266''N \text{ (} 38.433129^{\circ}N \text{)} \\ \text{Longitude} &= 79^{\circ}50'23.423''W \text{ (} 79.839840^{\circ}W \text{)} \\ \text{Height above the ellipsoid} &= 824.36 \text{ m} \\ \text{Height above the geoid} &= 855.46 \text{ m}\end{aligned}$$

# Chapter 14

## Solar System Radar with the GBT

### 14.1 Introduction

The GBT participates in radar observations of near-Earth asteroids and comets, as well as Lunar and planetary mapping and rotation studies. These are done in collaboration with the Arecibo Observatory which can transmit at 2380 MHz (S-band) or 430 MHz (P-band), and with JPL/Goldstone at 8560 MHz (X-band) or 7190 MHz (C-band).

Anyone wishing to do radar studies should collaborate with Scientists at either Arecibo or NASA/JPL to plan the experiment. Proposals should be submitted to NRAO-Green Bank as well as to Arecibo or JPL. Opportunities for radar observations can arise on short notice, in which case proposers can make use of “DTT” (director’s discretionary time) proposals if the normal proposal process is not timely enough. Use the NRAO proposal tool to submit all proposals, and indicate the proposal is for “DTT”; these proposals will be reviewed immediately without waiting for the usual review process.

### 14.2 Data Acquisition Backends

There are two data acquisition backends:

1. The Portable Fast Sampler (PFS) does either 2 or 4-bit sampling. Installed in 2001. Consult Jean-Luc Margot or Bruce Campbell if you want to use it.
2. JPL system, installed in 2014, does 8-bit sampling.

There is a possibility of a third option, using the VEGAS spectrometer. This is being developed and may be available at a later time. At present, the best choice is the JPL system which can be configured flexibly under computer control for a wide choice of bandwidths and sampling rates. The old PFS is configured by hand by changing cables and filters to its inputs. Sample rates and bandwidths are listed in table 14.1 and descriptions on the usage of both backends can be found in the following wiki pages:

**PFS system:** <https://safe.nrao.edu/wiki/bin/view/GB/Observing/RadarObserverAdvice>

**JPL system:** <https://safe.nrao.edu/wiki/bin/view/GB/Observing/JPLRadar>

Table 14.1: Radar data acquisition backends.

Backend	Sample rates	Bandwidths
PFS	5,10 MHz	0.5–10.7 MHz
JPL	6.25–160 MHz	1.3–60 MHz
VEGAS	undefined as yet	up to 1 Ghz

## 14.3 GBT Scheduling Blocks

The two backends, PFS and JPL, are connected in parallel, so the following configuration works for either one. It should be noted that the data recording is not controlled through the GBT user interface (Astrid). The SB tracks the object, but the user has to run the data acquisition process independently.

Here is an example script for 8560 MHz observations.

```
# Astrid setup script for X-band planetary radar
Catalog('/home/astro-util/GBTog/cats/asteroidephemexample.astrid')
obj = '1999JV6'

Xsetup = '''
receiver = 'Rcvr8_10' # select receiver
obstype  = 'Radar'    # select observing type
backend   = 'Radar'   # select type of backend
nwin      = 1
restfreq  = 8560      # observing frequency
bandwidth = 80
swmode    = 'tp_nocal'
swper     = 1.0
tint      = 1.0
vframe    = 'topo'
vdef      = 'Radio'
noisecal  = 'lo'
pol       = 'Circular'
'''

Configure(Xsetup)
Slew(obj)
AutoPeakFocus()
Break('Check peak')

Configure(Xsetup) # need to configure after the AutoPeakFocus
SetValues('ScanCoordinator',{'subsystemSelect,DCR':1}) # add DCR
Slew(obj)
Balance() # adjust power levels

# when tracking the object, adjust power levels in the back end.
Track(obj, None, 3600) # track object for one hour
Track(obj, None, stopTime='2016-01-09 07:00:00') # Track until UT stop time
```

Script 14.1: Example SB for radar observations.

The ephemeris file referred to in the **Catalog**() command, above, gives the coordinates for the object, as described in the next section. The object name, in this case “1999JV6” is defined in the file referred to in the **Catalog** command. One may omit the **Focus** part of the **AutoPeakFocus** command when observing at the lower frequencies, i.e, 2380 MHz or 430 MHz.

Refer to Chapter 6 for more information on GBT configurations and SBs.



## 14.4 Tracking moving objects

Here is an example of an ephemeris file for an asteroid. Refer to § 6.3.5 for description of the “Ephemeris format”.

```
# ephemeris format example for Astrid
FORMAT = EPHEMERIS
VELDEF = VRAD-TOP
COORDMODE = J2000
HEAD = date utc ra dec dra ddec
# 1: soln ref.= JPL#178
NAME = 1999JV6
2016-Jan-09 04:00 07:15:34.38 -23:41:33.7 -317.1984 1123.6330
2016-Jan-09 04:01 07:15:34.02 -23:41:15.0 -317.2251 1123.6110
2016-Jan-09 04:02 07:15:33.67 -23:40:56.3 -317.2518 1123.5900
2016-Jan-09 04:03 07:15:33.31 -23:40:37.6 -317.2763 1123.5680
2016-Jan-09 04:04 07:15:32.96 -23:40:18.8 -317.3008 1123.5460
2016-Jan-09 04:05 07:15:32.60 -23:40:00.1 -317.3231 1123.5250
2016-Jan-09 04:06 07:15:32.25 -23:39:41.4 -317.3454 1123.5030
2016-Jan-09 04:07 07:15:31.90 -23:39:22.7 -317.3667 1123.4820
2016-Jan-09 04:08 07:15:31.54 -23:39:04.0 -317.3868 1123.4610
# etcetera ...
```

Script 14.2: Example ephemeris file for an asteroid.

Consult § 6.3.5.2 for a description of obtaining ephemeris data from the NASA/JPL “Horizons” website and converting it for use with Astrid. Here is a brief description of the process:

- Access the JPL Horizons web interface: <http://ssd.jpl.nasa.gov/horizons.cgi>
- Set up Horizons web-interface as follows:
  - ephemeris type:** OBSERVER
  - target body:** [select the object]
  - Observer Location:** Green Bank (GBT) [select from list of observatories]
  - Time Span:** [put in desired values]
  - Table Settings:** QUANTITIES=1,3,20
    - (1) Astrometric RA&Dec, (3) rates in RA&Dec, and (20) Range and range rate
  - Display/Output:** plain text
- Click “Generate Ephemeris”
- Use the web browser file menu to save the output file as (for example) cometfilename.txt
- Run the program `jpl2astrid cometfilename.txt`  
 A new file with an `.astrid` extension will be created. An example of such a file is shown in Script 14.2.

The resulting `.astrid` file is used as an argument to the Astrid `Catalog()` command.

If you wish to track the velocity, use:

- `jpl2astrid cometfilename.txt vel`

This will put the velocity in the `.astrid` file. This option is usually not necessary because the relative velocity of the object is compensated by the transmitter, i.e., the transmitted frequency at Arecibo or Goldstone is programmed to result in a constant frequency received at Green Bank.

**Note:** the coordinate rates, columns 5 and 6 in the above example, as given by the Horizons listing, are:

- $dRA * \cos D$
- $d(DEC)/dt$

In converting to the `.astrid` result, `jpl2astrid` divides the RA rate by cosine(Declination) so that it is the rate in the RA, rather than in  $RA * \cos(Dec)$ . The units in both coordinates are arcseconds per hour.

The `jpl2astrid` program often does not fill in the object's name correctly. One should edit the NAME in the `.astrid` file to be something meaningful, and one should make sure the object name in the SB matches that in the ephemeris table.

## Chapter 15

# Radio Frequency Interference

Radio Frequency Interference (RFI) can be a significant problem for some observations. The most up to date information on the RFI environment at the GBT can be found at:

- <http://www.gb.nrao.edu/IPG/>

Useful resources, referenced from the above web page include a list of known sources of RFI:

- <https://safe.nrao.edu/wiki/bin/view/GB/Projects/RFIReportsTable>

and plots of RFI monitoring data:

- <http://www.gb.nrao.edu/IPG/rfiarchivepage.html>

Every observer should check for known RFI around their observing frequencies. If you suspect that this could have a significant impact on your observations you should contact your scientific support person to decide on an appropriate course of action.

### Mitigation of known RFI signals

In some cases, it is possible to turn off a known RFI source. For example, there is an amateur transponder at about 432 MHz, which we can request be shut down. If there are known RFI signals, the user should discuss them with the scientific support person. Given enough advance warning (days to weeks), we may be able to have them shut down during the observing.



# Chapter 16

## Weather Effects on Observations

The weather affects observations in three ways: winds affect the telescope pointing, differential heating and cooling affect the telescope pointing and efficiency, and atmospheric opacity affect the received signal and the system temperature.

### 16.1 Winds

Winds can set the feed arm into motion. The current recommendations for wind limits can be found in § 7.5 (specifically in Table 7.1). The fraction of time when wind speeds are low is illustrated in Figure 16.1 which shows the cumulative percentages when wind speeds are below a certain value. (Figure 16.1 is from Ries, PTCS project Note 68.1) The Dynamic Scheduling System (DSS), see Chapter 3) uses forecasted wind speeds when it determines what projects are suitable for scheduling, one should rarely see any negative impacts from winds.

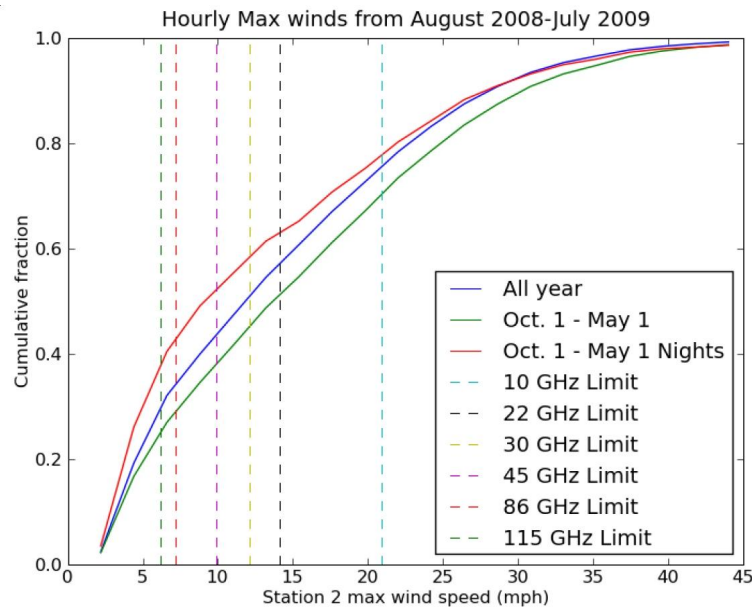


Figure 16.1: The cumulative fraction when wind speeds are below a certain value. Data from the year August 2008 to July 2009 are shown in blue; green shows winter data, and red shows winter nights.

## 16.2 Time of Day

Differential heating and cooling of the telescope alters the surface of the telescope, resulting in degradation of telescope efficiencies, and ‘bends’ the telescope, resulting in pointing changes. At high frequencies, these effects are important. The current recommendations are that, for best work, observing above 40 GHz should only be done at night, from 3 hours after sunset to 2 hours after sunrise. At 40 GHz and above it is recommended to use `AutoOOF()` (see 6.4.1.4 at the start of an observing session. Use `AutoOOF` for daytime observing at 27 GHz or higher

Low frequency observers may want to consider night time observing for two reasons. RFI is usually lower at night; and, in some cases, the sun has a slight negative impact on baseline shapes. By default, we assume that daytime observing will be acceptable for all observations below about 16 GHz.

Figure 16.2 depicts the range of UT, EST, and LST for our definition of “night-time” observing.

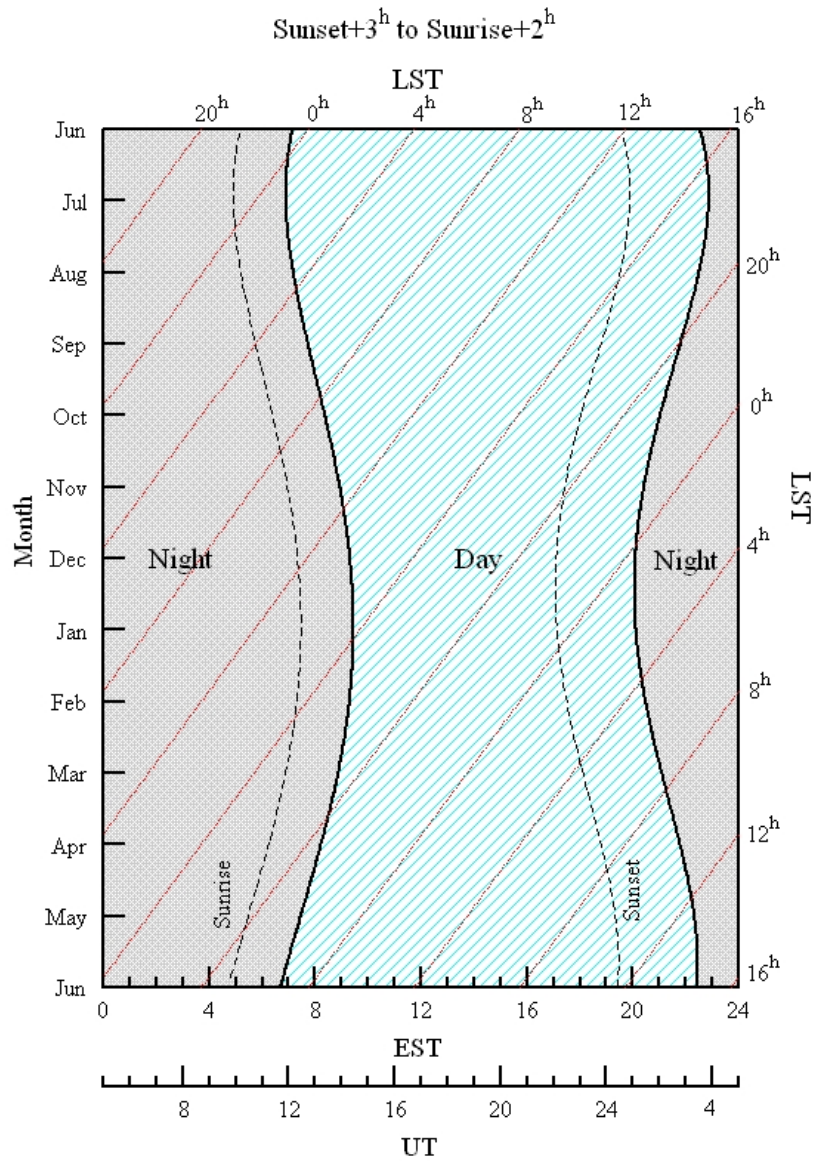


Figure 16.2: The range of UT, EST, and LST used in the GBT definition for “night-time” observing.

## 16.3 Atmospheric Opacities

The frequency range covered by the GBT extends from low frequencies where the opacity is relatively low ( 0.008 nepers) to high frequencies where opacity is very high ( $> 1$  nepers). Atmospheric opacity hits observing twice – it attenuates the astronomical signal and it increases the system temperature, and thus the noise in the observation, due to atmospheric emission.

Figure 16.4 shows opacities, atmospheric contributions to the system temperature and number of air masses<sup>1</sup> the astronomical signal must pass through vs. elevation under three typical weather conditions as calculated using the method described on the GBT “High Frequency Weather Forecasts” web page (<http://www.gb.nrao.edu/~rmaddale/Weather/index.html>). Typical total system temperatures are shown in Figure 16.3.

The opacities shown in Figure 16.4 are for planning purposes only and observers should not use them at high frequencies for calibrating data. Instead, one should use the actual opacities and the air mass from the bottom of Figure 16.4 to approximate the amount of attenuation a signal will experience at the expected elevation of the observation. The signal is attenuated by:

$$\exp^{-\tau A} \quad (16.1)$$

where  $\tau$  is the opacity and  $A$  is the total number of air masses. Since opacity is very weather dependent, please consult with a local support staff on how best to determine opacities for your observing run.

During the cold months, high frequency observers can expect to be observing with opacities that are at or below the average (50 percentile) winter conditions for Green Bank. Thus, high frequency observers can anticipate that the typical weather conditions under which they will observe will be best represented by the top 25 percentile conditions. In contrast, low-frequency, winter observers should expect they will observe under conditions that are worse than the 50 percentile and more like those of the 75 percentile conditions.

During the warm season (June through September), high-frequency observing is much less productive and we almost exclusively schedule low frequency observing. During these months, low frequency observers can plan on observing under the average, 50 percentile conditions.

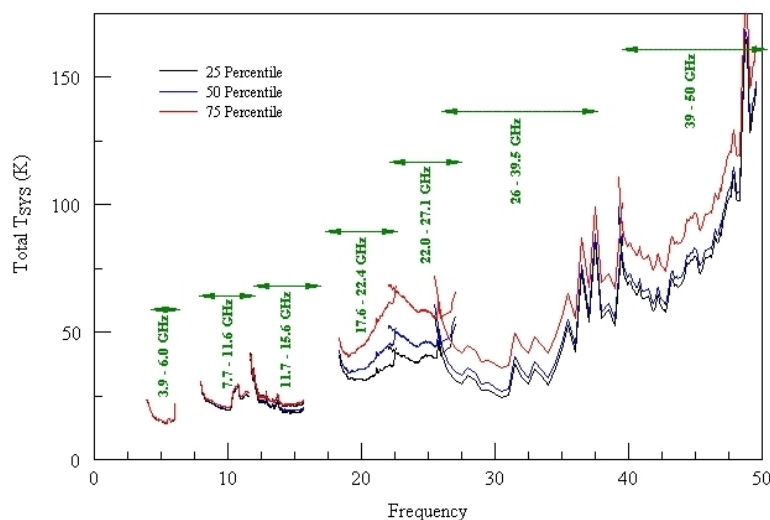


Figure 16.3: The zenith system temperatures for typical weather conditions.

<sup>1</sup>The airmass curve in Figure 16.4 is a better approximation than the  $\csc(\text{elevation})$  approximation which is only correct above about  $20^\circ$  elevation.

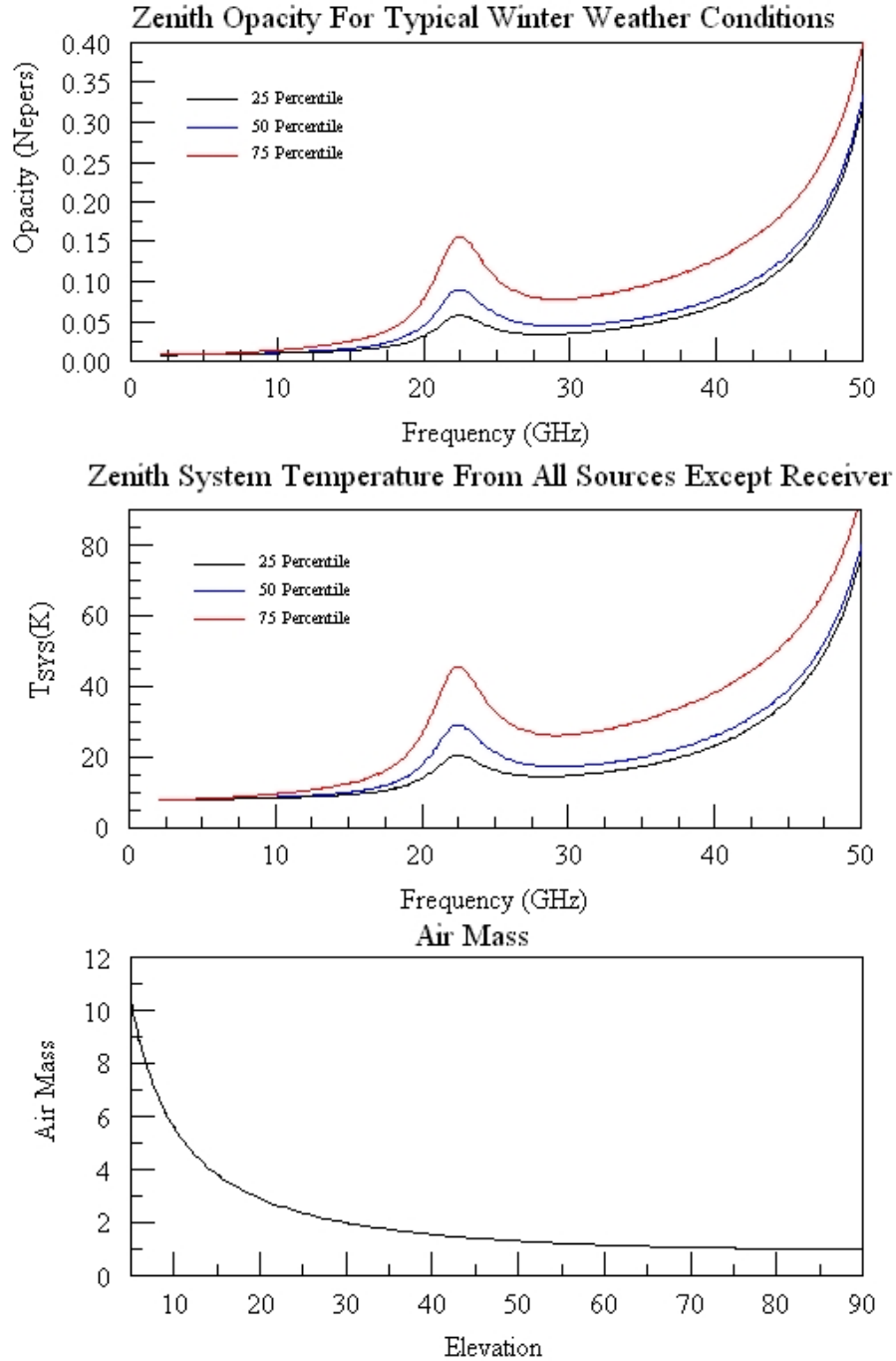


Figure 16.4: The top panel shows opacities under three typical weather conditions. The black, blue, and red curves represent the opacity under the best 25, 50, and 75 percentile weather conditions. (The 'average' opacity over the winter months is best described by the 50 percentile graph.) The middle panel is an estimate of the contribution to the system temperature at the zenith from the atmosphere, spillover, and cosmic microwave background. The bottom panel shows the number of air masses the astronomical signal must pass through as a function of elevation.



## 16.4 GBT Weather Restrictions

During weather conditions that pose a risk for the safety of the GBT, the GBT operators will cease all observations and take the appropriate action to ensure the safety of the GBT. The operator is fully responsible for the safety of the GBT and their judgement is final. The operators decisions should not be questioned by the observer.

### 16.4.1 Winds

The following guidelines exist for periods of high winds. If the average wind speed exceeds 35 MPH ( $15.6 \text{ ms}^{-1}$ ) over a one minute period, the operator will stop antenna motion. If wind gusts exceed 40 mph ( $17.9 \text{ ms}^{-1}$ ), or if winds are expected to exceed 40 mph for a period of time, the operator will move the antenna into the survival position. Only after the wind speeds have been below these criteria for 15 minutes will observations be allowed to resume.

Safety measures for high winds will take precedence over those for snow and ice.

### 16.4.2 Snow

If snow is sticking to any of the GBT structure, the operator will move the GBT to the “snow-dump” position. The decision to halt and resume observations is solely the responsibility of the GBT operator.

If dry snow appears to be accumulating, the operator may periodically interrupt operations to dump snow, and then resume observations.

### 16.4.3 Ice

If ice is accumulating on any part of the GBT structure, the operator will move the GBT to the survival position. The decision to halt and resume observations is solely the responsibility of the GBT operator.

### 16.4.4 Temperature

When the air temperature drops to  $16^\circ$  Fahrenheit ( $-8.9^\circ\text{C}$ ), the Azimuth slew rate of the GBT will be reduced to half of its normal rate. (This is due to the changing properties of the grease used in the Azimuth drive bearings.) Half rate speed ( $18^\circ/\text{min}$  instead of  $36^\circ/\text{min}$ ) will be utilized until the temperature returns above  $17^\circ$  Fahrenheit ( $-8.3^\circ\text{C}$ ). When the temperature drops below  $-10^\circ$  Fahrenheit ( $-23.3^\circ\text{C}$ ) observations will cease until the temperature is above  $0^\circ$  Fahrenheit ( $-17.8^\circ\text{C}$ ) and the operator has determined that the Azimuth drive motors are ready for use.

### 16.4.5 Feed Blowers

The feed blowers blow warm air over the radomes of the feeds to prevent condensation and frost. Although beneficial for most receivers, they produce vibrations that contaminate the MUSTANG data. Thus, users of MUSTANG can request that the operator turn off the feed blower at the start of their observing session. One hour before the end of a MUSTANG observing session, the operator will decide whether or not the blower needs to be turned back on in order to ensure the feeds for all receivers are in good shape for the next observer. The operators use the criteria that the blowers will be turned back on for the last hour if either: 1) the dew point is within  $5^\circ$  Fahrenheit of the air temperature, or (2) the air temperature went from above to below freezing anytime during the MUSTANG run.



# Chapter 17

## Computing

### 17.0.1 Accounts

New observers should obtain a Green Bank Linux computing account as soon as possible. Before receiving your account you will have to go through a screening process. This screening will automatically happen for visiting observers making a room reservation in the BOS system. You can also begin the screening process by filling in the form at <https://info.nrao.edu/oas/dpscompliance>.

Accounts may be requested from [helpdesk-gb@nrao.edu](mailto:helpdesk-gb@nrao.edu). Any problems with connecting a personal computer to our network should also be referred to the helpdesk. Please note that accounts will be archived after approximately two years of inactivity but can be reactivated upon request.

### 17.0.2 Computing Facilities

Workstations with access to both Windows and Linux servers are available for visitors at the Observer Periphery (Room 105) in the Jansky Lab. Laptop connections are provided there and in several locations around the Observatory, including some rooms in the residence hall. Additional workstations can be found in the residence hall lounge and Jansky Lab library.

### 17.0.3 Remote Access and VNC

You will need to access the Green Bank Linux system remotely for script preparation, remote observations, and data reduction.

Scheduling blocks (scripts that perform astronomical observations) should be prepared and uploaded into Astrid (see Chapter 4) well in advance of any observations or first time visits. In order to run Astrid all observers must ensure they can connect to our network via VNC. Detailed instructions for setting up VNC on Linux, Mac OS X, and Windows can be found at <https://science.nrao.edu/facilities/gbt/observing/remote-observing-with-the-gbt>.

When starting a new VNC session please remember that the *stargate* host, along with the machines *titania* and *ariel* are reserved exclusively for remote observations. For all other purposes please use *prospero* (<ssh.gb.nrao.edu>) as the host, and one of the data reduction machines listed in <http://www.gb.nrao.edu/pubcomputing/data-reduction.shtml>.

Observers wishing to reduce VEGAS data will need access to the “lustre” file system. Refer to <http://www.gb.nrao.edu/pubcomputing/public.shtml> for a list of lustre clients.



# Chapter 18

## Remote Observing With The GBT

### 18.1 Remote Observing Guidelines for Approved Projects

Permission to observe remotely must be explicitly granted by the Head of Science Operations (currently Toney Minter) at least two weeks prior to the observing run. Permission will be granted based on the appropriateness of the project and the demonstrated experience of the observer. If you wish to become authorized for remote observations, please contact your assigned GBT “Friend”. Guidelines for approved remote observing projects are as follows.

- Consult with your assigned GBT “Friend” at least two weeks prior to observing time.
- Ensure your contact details in the DSS (<https://dss.gb.nrao.edu>) are current.
- Prepare all Astrid SBs in advance.
- Contact the telescope operator 30 minutes before the start of your observing session. You may use the CLEO `Talk` and `Draw` application (see § 5.2.2) or call via telephone. The contact number is 304-456-2346 or the Operator’s direct line at 304-456-2341. x2346 connects to a speaker phone and is the preferred number to use. In case of a power outage, the direct line to the control room is 304-456-3203. Provide the operator with all the appropriate contact information.
- Launch VNC network displays of observing applications. Suggested VNC setup procedures are provided below.
- Start up your observing applications immediately after contacting the operator. These usually include
  - **CLEO** Open at least the `Talk` and `Draw` windows, then any other CLEO application you need (see § 5.2).
  - **Astrid** The GBT observing interface (see Chapter 4).
  - **GBTIDL** The GBT data reduction package (see <http://http://gbtidl.nrao.edu/>).
- When your observing time starts and the operator gives you security access (puts you in the “gateway”), you should put Astrid into “online” mode, load your configuration and start observing.
- At the end of your observations, put Astrid into “offline” mode and remember to properly close your VNC session using the `vncserver -kill` command.

Please consult <https://safe.nrao.edu/wiki/bin/view/GB/Observing/GbtObservingPolicies> and <https://science.nrao.edu/facilities/gbt/observing/remote-observing-with-the-gbt> for information on observing policies, remote observing and VNC setup instructions.



## Chapter 19

# Planning Your Observations And Travel

## 19.1 Preparing for Your Observations

After your proposal has been accepted you will be notified of how much observing time you will receive on the GBT. You will also be notified of who will be your scientific contact person (friend). You should contact your scientific support person well in advance of your observations to help you develop observing strategies and your SBs.

We require that new observers (or experienced observers doing new projects outside their previous realm of experience) come to Green Bank for their initial observations. Advisers are also required to accompany their students for their first trip to Green Bank.

- All policies are found at <https://safe.nrao.edu/wiki/bin/view/GB/Observing/GbtObservingPolicies>.
- Room reservations can be made at <https://bos.nrao.edu/reservations>.
- Call +1 (304)-456-2227 for further information on reservations and planning your travel.

Contact your GBT friend well in advance of the observations to determine the optimum dates for your visit and ensure that the telescope and hardware will be available for the project.

You should plan on being in Green Bank at least one full business day before your observations begin. This will allow you to meet with your scientific support person as well as the staff support person who will be on-call during your observations.

If your observations are dynamically scheduled and dependent on weather conditions, you should plan to spend at least a week, and preferably two weeks, to increase the likelihood that appropriate conditions for the observations will occur during your visit.

## 19.2 Travel Support

Some travel support for observing and data reduction is available for U.S. investigators on successful proposals. More information on the travel support that NRAO provides can be found at [http://www.nrao.edu/admin/do/nonemployee\\_observing\\_travel.shtml](http://www.nrao.edu/admin/do/nonemployee_observing_travel.shtml).

## 19.3 Trains, Planes and Automobiles

In principle, observers may use a number of area airports for their travel to Green Bank. These include Washington Dulles, Pittsburgh, Charlottesville, Roanoke (VA), Charleston (WV), Clarksburg (WV) or Lewisburg (WV). In addition, limited AMTRACK train service is available to Charlottesville and White Sulphur Springs, WV. Rental cars are available at most of the airports. If you do not have your own transportation, the Observatory can also send a driver for pickup at any of these airports or stations.

## 19.4 Housing

The residence hall is available for astronomers while observing or reducing data after completion of their observations. Single rooms (2 beds) are available for \$55.00 (+tax) a day single occupancy or \$69.00 (+tax) per day per room double occupancy. Students attending a degree conferring college or university and coming to Green Bank to use the telescope will pay single room rate \$47.00 (+tax) or \$59.00 (+tax) per day double occupancy. In addition, there are four one-bedroom apartments with equipped kitchens at \$83.00 (+tax) per day. Cribs, high chairs, and fold-up beds are also available. Costs of lodging at observatory facilities can be waived on request in advance and on approval of the Site Director.

## 19.5 Getting To Green Bank

### 19.5.1 Where is Green Bank?

A map showing the location of Green Bank relative to major, nearby towns and cities is shown in Figure 19.1. Simplified directions are also shown in Figure 19.1.

Green Bank is located in Pocahontas County, WV, very close to the Virginia border and at about the mid-point of the full extent of the Virginia–West Virginia border.

### 19.5.2 Directions to Green Bank

The following are directions to Green Bank from airports in Pittsburgh, Pa, Washington, DC, Charlottesville, Va and Roanoke, Va. The duration of the drive from either Pittsburgh or Washington is four to five hours. The duration of the drive from Charlottesville and Roanoke is about 2–1/2 hours.

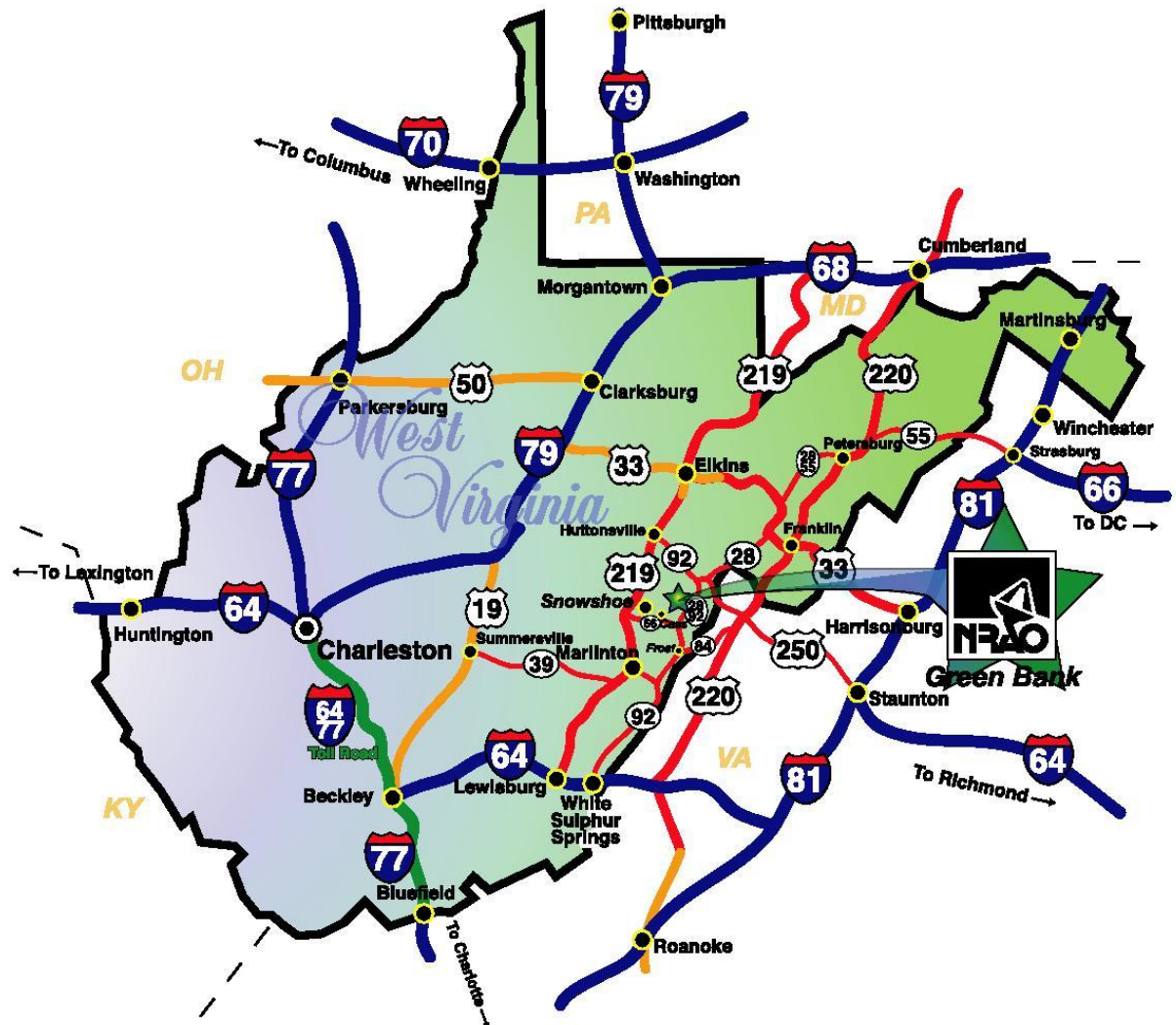
#### 19.5.2.1 Beware of GPS!!

GPS systems, Mapquest, and other such automated route finding systems are notoriously unreliable within 50 miles of the Observatory. Some roads that are recommended by these systems are passable only with 4-wheel drive vehicles. Do not turn onto unpaved roads !

#### 19.5.2.2 Pittsburgh to Green Bank

From the Greater Pittsburgh International Airport, go east on route 60 to US 22/30. Follow 22/30 east to Interstate 79. Take I-79 south through Clarksburg, WV to the US 33 exit (exit 99) near Buckhannon, WV. Go east on US 33 through Buckhannon to Elkins, WV. Turn south on US 250/219 to go to Huttonsville. In Huttonsville, take US 250 (route 92) southeast to Bartow. Follow route 92/28 south to Green Bank.





### Directions to the National Radio Astronomy Observatory Green Bank, West Virginia

**From Charleston, WV:**  
Via I-64 East, exit at White Sulphur Springs (Exit 175) and take Rt. 92 North to Green Bank.

**From Roanoke, VA:**  
Via I-64 West, exit at White Sulphur Springs (Exit 181) and take Rt. 92 North to Green Bank.

**From Pittsburgh:**  
Via I-79 South, exit at Weston/Buckhannon (Exit 99), and travel US 33 East to Elkins, then take Rt. 92 South to Green Bank.

**From Washington DC:**  
Via I-66 West to I-81 South.  
**Option 1:** Take I-81 South for ~3 miles exit at Strasburg (exit 296) to Route 55. Stay on Route 55 through Wardensville, WV, Moorefield, and Petersburg, then Rt. 28 South to Green Bank.

**From Washington DC:**  
Via I-66 West to I-81 South.  
**Option 2:** Take I-81 South to Harrisonburg (exit 247) take the truck bypass around Harrisonburg to US 33 West. Take US 33 West through Franklin, then take Rt. 28 South at Judy Gap to Green Bank.

**From Charlottesville/Richmond:**  
Via I-64 West to Staunton, then I-81 North for ~3 miles to Exit 225 (Woodrow Wilson Parkway). Take the Parkway around Staunton to US 250 West. Stay on US 250 West to Monterey, VA. At Monterey take Rt. 220 South ~3.5 miles to Rt. 84 West. Take 84 West to Frost, WV. At Frost, take Rt. 92 North to Green Bank.

Figure 19.1: Direction to Green Bank.

### 19.5.2.3 Washington Dulles or National to Green Bank

From the Washington Dulles International Airport, go south on route 28 to Interstate 66.

From the Washington National Airport, take US 1/Va 110/US 50 to Interstate 66 (ask at airport for exact details).

Take I-66 west to I-81. From here follow I-81 south to the exit for route 55 near Strasburg, VA. Go west on 55 to Moorefield, WV. Turn south on US 220/55, and drive to Petersburg, WV. Go south on route 28/55 to Seneca Rocks. Continue south on 28 through Judy Gap to Bartow. Follow route 92/28 south to Green Bank.

### 19.5.2.4 Charlottesville to Green Bank

From the Charlottesville-Albemarle Airport, go east (straight) on Airport Road to US 29. Take 29 south to US 250, and follow 250 west to Interstate 64. Go west on I-64 to Interstate 81 near Staunton, VA. After traveling north on I-81 for about two miles, take the Woodrow Wilson Parkway exit. Go west on the parkway to US 250, and follow 250 west to Monterey, VA. In Monterey, turn south on US 220, and shortly thereafter, veer west on route 84 to go to Frost, WV. Follow route 92/28 north to Green Bank.

### 19.5.2.5 Roanoke to Green Bank

From the Roanoke Airport go left (south) on Valley View Drive/Airport road and then almost immediately turn right onto Hershberger Road. Go about 1/2 mile and then take I-581 north to I-81 north. Go two exits on I-81 north to US 220, the Daleville/Troutville exit. Take US 200 north toward (and through) Fincastle until you reach I-64. Take I-64 west to White Sulphur Springs, WV. Take the first White Sulphur Springs exit and turn right. After about 1/2 mile turn right onto route 92 north and this will take you to Green Bank.

## 19.5.3 Once You Are in Green Bank

The entrance to the observatory is about one-half mile north of “downtown” Green Bank on the west side of route 92/28 (see Figure 19.2). Look for the Jansky and Reber antennas. They are located on either side of the entrance.

To pick up your room keys and site keycards you will need to come to the Jansky Lab. The Jansky lab is the second building on the left as you enter the site. Go in the “atrium” main entrance. If after working hours, use the telephone on the left to call the GBT operator at 2341. The operator will then “buzz you in”. If during normal working hours, just walk in. The room packets and keys will be found near the beginning of the hallway on your right as you enter the Jansky Lab.

The residence hall is the first building on the right as one enters the observatory. Adequate parking is provided on the west side of the residence hall. Enter the residence hall through the double glass doors on the west side of the building. The observer’s lounge is located on the second floor of the residence hall, directly above the entrance.

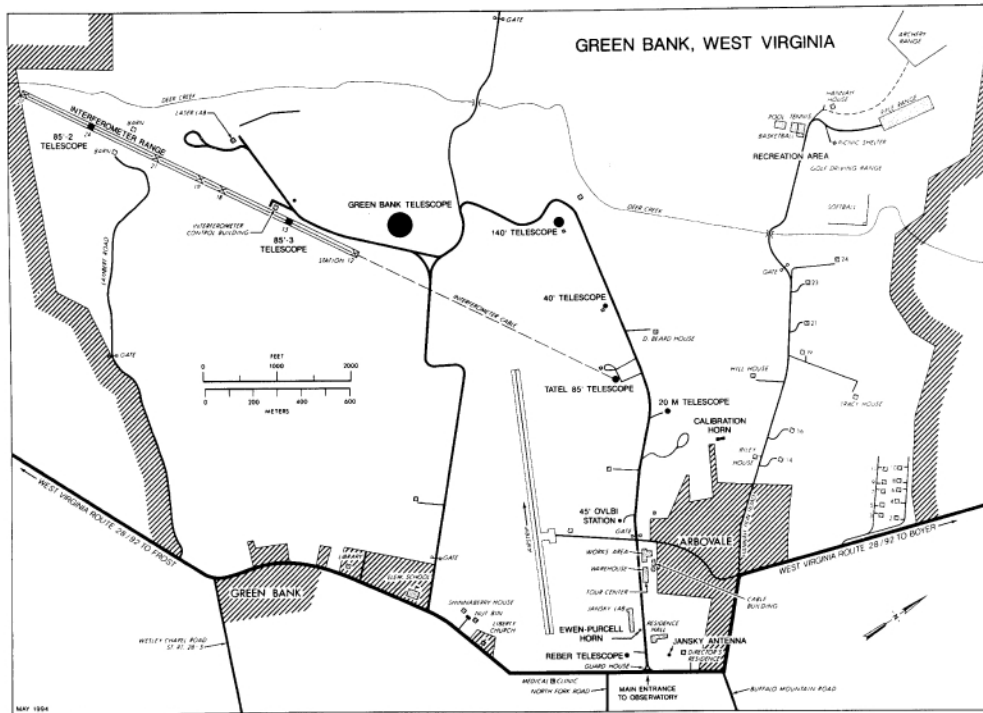


Figure 19.2: NRAO, Green Bank site map.



## Chapter 20

# After Your Observations

## 20.1 Taking your data home

All data must be transferred over the internet to your home institution. Several methods are available for this such as the `scp` utility shown below. If you wish to transfer more than a few hundred GigaBytes, please contact the computing department at [helpdesk-gb@nrao.edu](mailto:helpdesk-gb@nrao.edu) for assistance. For more general questions or advice, please contact your GBT “friend”.

```
[you@yourmachine ~]$ scp you@GBTmachine:data.fits datacopy.fits
```

## 20.2 Installing GBTIDL

Most spectral line observers will use GBTIDL to reduce their data. If you have an IDL licence at your home institution then you can obtain a copy of GBTIDL, the user guide and installation instructions from <http://gbitdl.nrao.edu>. Otherwise, feel free to continue using your GBT Linux account for data reduction.

## 20.3 Keep Your Contact Person Informed

Don't hesitate to ask your scientific contact person if you are having trouble reducing your data, or if you have questions about your data. It does not matter how long its has been since you observed, your contact person will be happy to help you.

## 20.4 Press Releases and News-worthy Items

News-worthy items should be discussed with the NRAO press officer. The press officer can help write an NRAO press release or a press release from your home institution. For more information see page 30 of the January 2007 version of the NRAO Newsletter (<http://www.nrao.edu/news/newsletters/nraonews110.pdf>).

## 20.5 Publishing Your Results

Finally you should publish your results. The NRAO will help with page charges for the publication of the results from your observations. Please see [http://www.nrao.edu/library/page\\_charges.shtml](http://www.nrao.edu/library/page_charges.shtml) for more details. **Please inform your scientific contact person of any publication resulting from your observations.**

# Appendix A

## GBT IF System

In this appendix we provide a general outline of the GBT IF system. Figures A.1 and A.2 give a simplified overview of the GBT IF path and will guide our discussion. We will not cover MUSTANG as it is a direct detection system. Note that during each frequency mix, each polarization pair is mixed with a signal from the same synthesizer. All synthesizers are locked to our H-maser frequency standard. All synthesizers are locked to our H-maser frequency standard.

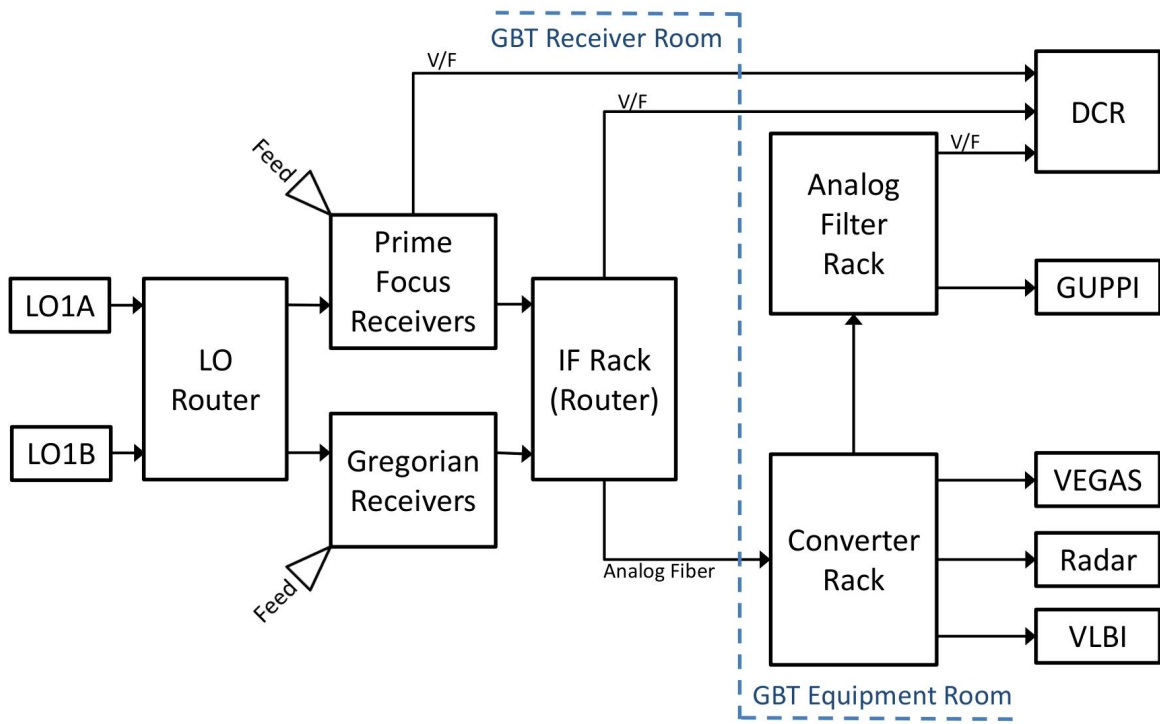


Figure A.1: A simplified flow diagram of the GBT IF system routing.

## Simplified GBT LO/IF system

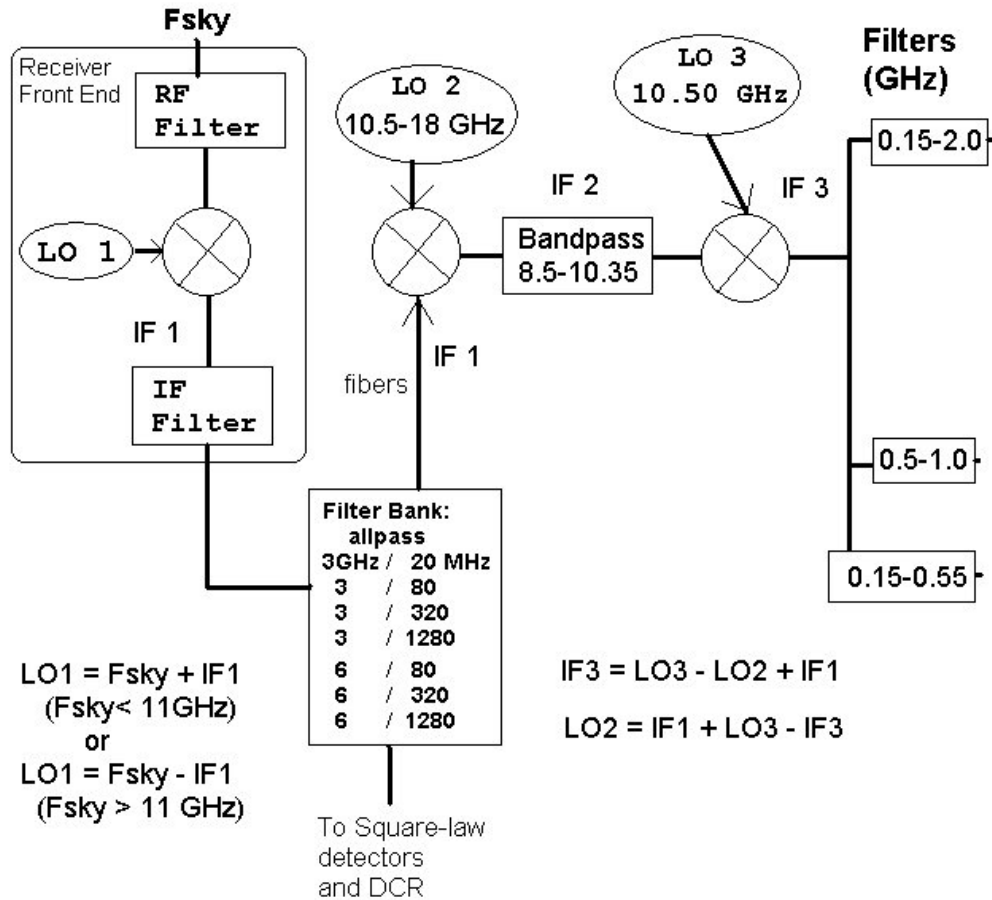


Figure A.2: A simplified flow chart of the GBT IF system.

## A.1 From the Receiver to the IF Rack

The frequency that is observed is given by  $F_{sky}$ . Within the receiver the detected signal at  $F_{sky}$  is mixed with the LO1 signal. The LO1 frequency is derived from a synthesizer and can vary in time when Doppler tracking the velocity of a spectral line. The result of the mixing of  $F_{sky}$  and LO1 is the IF frequency, IF 1. The typical IF 1 center frequencies are 1080, 3000 and 6000 MHz. Filters limit the bandwidth in the receivers both before and after the LO1 mix. There are also filters in the IF Rack that limit the bandwidth. The resulting allowed bandwidths are 20, 80, 320, 1280 MHz and “All Pass” (i.e. no filtering other than the response of the receiver).

Before the IF Rack each signal is split into two (single beam receivers only) copies of the original signal. Each signal in the IF Rack is detected and then sent to the DCR (as used during pointing and focus observations). Each signal is also sent as an analog signal over optical fiber to the Jansky Lab to the Converter Rack.



## A.2 From the Converter Rack to the Backend

When the signal reaches the Converter Rack it is split into four separate copies. This allows up to eight different copies of the received signal for single beam receivers and four copies of each received signal for dual beam receivers.

In the Converter Rack the signal is mixed with the LO2 signal. Each copy of the signal can be mixed with a different LO2 since there are eight different LO2 synthesizers. The resultant signals are then sent through a filter to make sure it has a bandpass of no more than 1.85 GHz. A final mix with a fixed frequency of 10.5 GHz then gets the signal within the input band-passes of the backends. There is a final set of filters that ensures the signal has the correct bandwidth for the backend.

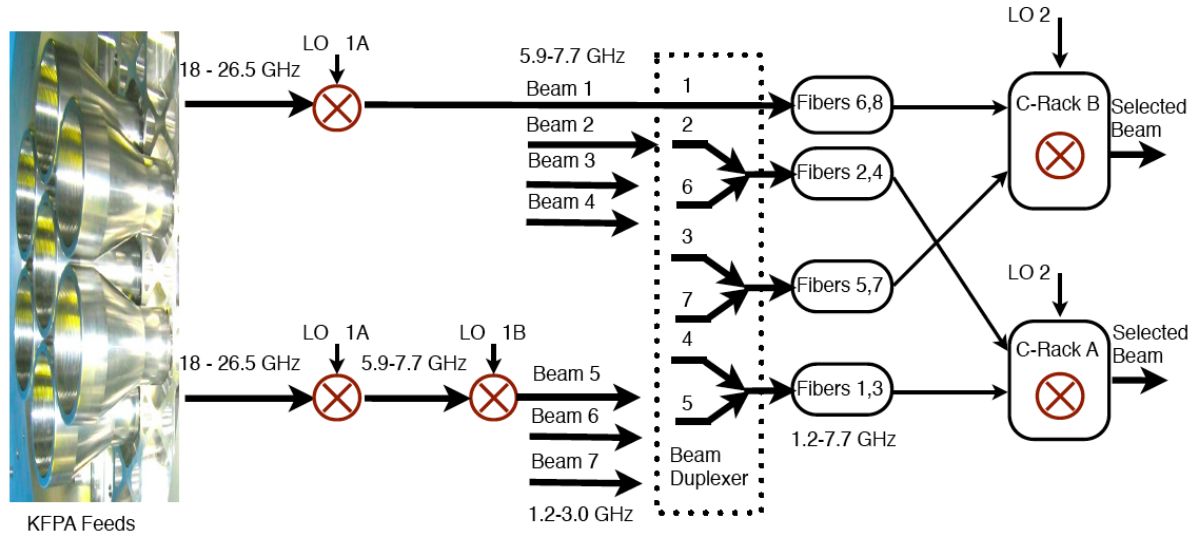


Figure A.3: A simplified KFDA diagram showing the combination of beams onto fiber modems and their selection in the Converter Rack Modules A and B.

## A.3 KFDA Combined IF

The KFDA receiver, with 7 beams, is the first GBT receiver with more IF signals than there are Optical Fibers from the GBT to the Jansky Lab. In order to bring the IF signals to the control room, pairs of signals from different beams were duplexed on single fibers. The signal combination was accomplished by an analog addition of the IFs of pairs of beams. Beams 1,2,3 and 4 have IF signals centers at 6800 MHz. The IF signals from beams 5, 6 and 7 are down converted to 2100 MHz center frequency. Beam 2 is paired with 6, beam 3 with 7 and beam 4 with 5 (See Figure A.3).

At the Converter Rack one of the two beams is selected by appropriately setting the converter rack LO frequency. Beams 2, 4, 5 and 6 are routed to Converter Rack A and beams 1, 3 and 7 to Converter Rack B. This constrains certain multi-beam observing modes, as is described in Chapter 10.



## Appendix B

# Introduction to Spectral Windows

Several simultaneous frequency bands may be specified with a list of rest frequencies and offsets (keywords **restfreq**, **deltafreq**, see § 6.2.5). If using a backend other than VEGAS, the **nwin** (number of spectral windows) keyword will also need to be specified. Each spectral window includes both polarizations. i.e., if you specify one window, you get two IF systems routed to the back end device, one for each polarization; if you specify two windows, you get 4 glsplIF, and so forth.

The configuration software tries to put the midpoint of the total frequency range spanned by all windows at the center of the nominal IF1 band so as to use the narrowest IF bandpass filters that will pass the desired range of frequencies. In some uncommon cases this is not possible, so the IF bandwidth must be increased to pass the desired range of frequencies. For prime focus receivers, the total IF bandwidth is 240 MHz; for the Gregorian receivers, up to 4 GHz is possible, depending on the receiver.

The user specifies the rest frequencies (**restfreq** keyword) and may also specify a range of radial velocities (**vlow** and **vhigh** keywords, see Appendix C). The various IF filters are set to include the required range of frequencies in the local frame required by the radial velocity range. The configuration software predicts the local frequency for each spectral window based on the rest frequencies and the radial velocity. During observing the tracking LO will correctly track the doppler tracking frequency set by the **dopplertrackfreq** keyword. If **dopplertrackfreq** is not provided, the default value will be the first spectral window specified by the **restfreq** keyword will be used (if not using the advanced **restfreq** syntax). Because there is only one tracking LO, the other spectral windows are set up with frequency offsets in the local frame with respect to the doppler tracking frequency. When observing at a variety of high velocities, one should run a configuration for each change of velocity (i.e., do not rely on just changing the velocity in the LO1 manager), and one should set **vlow=vhigh**.

Note that the **deltafreq** keyword gives frequency offsets that are applied in the local (or topocentric) frame i.e., it is applied as an offset in the IF system. For example, if  $V_{frame}$  is velocity of the reference frame,  $V$  is source velocity in that frame,  $\nu_{rest}$  is the rest frequency of the line and we use the Radio definition of velocity then the topocentric frequency will be

$$\nu_{topo} = \nu_{rest} \left( 1 - \frac{(V + V_{frame})}{c} \right) + \text{deltafreq} \quad (\text{B.1})$$

Finally note that the expert user may specify any of the IF system conversion frequencies and total IF system bandwidth, overriding the calculations done by the configuration software (**ifbw**, **if0freq**, **lo1bfreq**, **lo2freq**, and **if3freq** keywords). This option may be needed in some peculiar cases. Of course one needs a good knowledge of the IF system to make use of this option.

## B.1 Array Receiver Spectral Windows

Array Receivers can be configured with a variety of spectral windows. The `configtool`, part of Astrid, sets up these spectral windows, and a new syntax was required to specify more complex configurations. Each feed has the potential to be tuned to a different rest frequency. For the KFPFA receiver, a special “all” beam mode is defined which uses all 7 beams, plus one beam tuned to a second, different spectral window. This stretches the syntax of the `configtool` `restfreq` and `deltafreq` keywords. In order to support these modes within the `configtool`, expanded values and interpretations of `nwin`, `deltafreq` and `restfreq` were implemented.

The syntax uses a python dictionary for the `restfreq` and `deltafreq` keyword values for KFPFA configurations. The `restfreq` dictionary maps beams and frequencies of the spectral windows. The delta frequency is a map of `deltafreq` to `restfreq`. The list of values syntax continues to be supported for simpler modes. When the dictionary is used to specify the rest frequencies, this dictionary must contain a key named ‘DopplerTrackFreq’. The value assigned to this key is the rest frequency that will be used by the LO as the Doppler tracking frequency.

The following examples show of how to specify `configtool` frequency settings:

- **Example 1:** Requests that beams 1,2,3 and 4 have a rest frequency of 24000 MHz, that beams 5,6,7 have a rest frequency of 23400 MHz and the 2nd beam 1 IF band has a rest frequency of 25000 MHz. There are no delta frequencies used in this observation. For non zero delta frequencies, the `deltafreq` values should be specified in the same manor as the `restfreq`.

```
beam = 'all'
restfreq = {24000: '1,2,3,4', 23400: '5,6,7', 25000: '-1',
            'DopplerTrackFreq': 24200}
deltafreq = {24000: 0, 23400: 0, 25000: 0}
```

- **Example 2:** For simple configurations the syntax for the existing receivers would also be supported. This results in the routing of 4 beams, 2 polarizations with each tuned to a rest frequency of 24000 MHz.

```
beam = '1,2,3,4'
restfreq = 24000
```

- **Example 3:** Comparison of two `configtool` inputs where `restfreq` is a list, and input with the dictionary syntax. The two configurations are equal.

```
beam = '1,2'
restfreq = 23706.3, 24139.417
deltafreq = 0, 0
```

```
beam = '1,2'
restfreq = {23706.3: '1,2', 24139.417: '1,2',
            'DopplerTrackFreq': 23706.3 }
deltafreq = {23706.3: 0, 24139.417: 0}
```

- **Example 4:** 8 different rest frequencies specified.

```
restfreq = {23706.3 : '1', 24139.417: '2', 24139.417: '3', 24706.3: '4',
            24149.4: '5', 24122: '6', 23899: '7', 24876.1: '-1', 'DopplerTrackFreq': 24876}
```

- **Example 5:** A configuration that specifies delta frequencies.

```
beam = 'all'
restfreq = {24000: '1,2,3,4', 23400: '5,6,7',
            25500: '-1', 'DopplerTrackFreq': 24876}
deltafreq = {24000: 0, 23400: -500, 25500 : 0}
```

## Appendix C

# Usage of **vlow** and **vhigh**

The configuration keywords **vlow** and **vhigh** give the range of velocities of all sources to be observed. This information is used to set various filters in the system that will simultaneously cover the required range of velocity. Setting the velocity for each specific source is done later in the SB. For galactic sources where the range of velocities is rather small it is usually best to set both **vlow** and **vhigh** to zero.

When strong RFI is present it is best not to use **vlow** and **vhigh**. The use of **vlow** and **vhigh** can cause the GBT IF system to have a larger IF bandwidth than is necessary for a single source. This can let parts of the IF system be unnecessarily affected by RFI. The observers might need to reconfigure after each source if the change in velocity is larger than the bandwidth of a filter.

An example of how **vlow** and **vhigh** can be used is as follows. Suppose that you are looking for water masers in extragalactic AGN. Furthermore, let's say that you are looking at 100 candidates with velocities from  $1000 \text{ km s}^{-1}$  to  $40000 \text{ km s}^{-1}$ . Then you would set **vlow**=1000.0 and **vhigh**=40000.0 and will not change the IF configuration when you change sources.

Note that if **vdef**="Red" (i.e., redshift), then you must give the redshift parameter "z" as the values for "vlow" and "vhigh" instead of velocity.

Your scientific contact person can help you decide if you should use **vlow** and **vhigh**.



# Appendix D

## Advanced Functions

There are a few advanced functions that once can use in a Scheduling Block (SB).

### D.1 General Functions

#### D.1.1 GetValue()

The **GetValue()** function can be used to retrieve any parameter or sampler value within the Monitor and Control (M&C) system.

**SYNTAX:** value = **GetValue**( 'manager', 'parameter,parameter\_field' )

**value** A string. If you need the return value to be another data type such as an integer or float, please consult your favorite Python manual to find out how to use conversion operators.

**manager** A manager in the M&C system such as 'scanCoordinator'

**parameter** The name of the parameter or sampler value to retrieve.

**parameter\_field** The name of the parameter or sampler field value to retrieve.

**USAGE:** Please consult with your scientific contact person.

```
current_source = GetValue('ScanCoordinator','source')
current_El_lpc = GetValue('Antenna','localPointingOffsets,elOffset')
```

Script D.1: GetValue() example.

## D.1.2 SetValues()

The **SetValues()** function can be used to directly set any of the parameters within the M&C system. As a result, it is used to support complex configurations and expert observations. Please note that **SetValues()** does not always issue a “prepare” on the M&C Manager containing the parameter. If you wish to do a “prepare”, you can also use **SetValues()** to do that as well.

**SYNTAX:** **SetValues**( ‘manager’, {‘parameter,parameter\_field: value’} )

**manager** The manager containing the parameter in the M&C system.

**parameter** The name of the parameter.

**parameter\_field** The name of the parameter field.

**value** The actual value to set. Data types depend on the parameter.

**USAGE:** Please consult with your scientific contact person.

```
lfcValues = {
  'local_focus_correction,Y': -7.469,           # in mm
  'localPointingOffsets,azOffset2': 9.8902e-06, # in radians
  'localPointingOffsets,elOffset': 7.27221e-05} # in radians

SetValues('Antenna', lfcValues)
SetValues('Antenna', {'state': 'prepare'})
```

Script D.2: SetValues() example.

## D.1.3 DefineScan()

If you have written your own scan type using the Python language, the **DefineScan()** function is used to load your new scan type into the current SB. Once loaded, it can be referred to by name, just like any other scan type.

**SYNTAX:** **DefineScan**( scanName , filepath )

**scanName** A string specifying a name for the scan.

**filepath** A string specifying the full filepath to the scan.

**USAGE:** The following example defines and then executes a scan used primarily with MUSTANG observations (see § ??).

```
DefineScan('boxtraj', '/users/bmason/gbt-dev/scanning/ptcsTraj/boxtraj.py')
boxtraj(mySrc, x0=x0, y0=y0, taux=taux, tauy=tauy,
        scanDuration=scandur, dx=dx , dy=dy)
```

Script D.3: DefineScan() example.



## D.1.4 GetCurrentLocation()

Given a coordinate mode, **GetCurrentLocation()** returns a Location object

**SYNTAX:** value = **GetCurrentLocation**( coordinateMode )

**value** A Location object (see § 6.6.1) containing the coordinates of the currently selected receiver beam's position on the sky (as selected in the most recent scan type).

**coordinateMode** A string specifying the coordinate mode of the Location object in the return value. Available coordinate modes are 'J2000', 'B1950', 'RaDecOfDate', 'HaDec', 'ApparentRaDec', 'Galactic', 'AzEl', and 'Encoder'.

**USAGE:** The following example prints the current coordinates in azimuth and elevation. Note that GetH() and GetV() return float values for the major and minor axis coordinates of Location and Offset objects.

```
location = GetCurrentLocation('AzEl')
print 'Az = %s, El = %s'%(location.GetH(), location.GetV())
```

Script D.4: GetCurrentLocation() example.

## D.1.5 SetSourceVelocity()

The **SetSourceVelocity()** function sets the LO1 source velocity directly, in units of km/s.

**SYNTAX:** value = **SetSourceVelocity**( velocity )

**velocity** The source velocity in km/s.

**USAGE:** If you include the velocities in your catalog (see § refsec:catalogs) then you do not need to use this function.

```
SetSourceVelocity(10.5)
```

Script D.5: SetSourceVelocity() example.

## D.2 Specialty Scan Types

### D.2.1 Spider

**Spider()** executes the specified number of slices of duration **scanDuration** through the specified location. Each slice is of length  $2*\text{startOffset}$ . The argument **startOffset** also specifies the angle of the initial slice. The user may specify unidirectional or bidirectional subscons of length **calDuration** and when to run calibration subscons relative to each slice, i.e., at ‘begin’, ‘end’, or ‘both’.

#### SYNTAX:

**Spider**( location, startOffset, scanDuration, slices, beamName, unidirectional, calDuration )

**location** A Catalog source name or Location object. It specifies the source which is to be tracked.

**startOffset** An Offset object. It specifies the 1/2 length of the subscons and the angle from location of the initial subscon. For example, If **startOffset** = **Offset**( ‘AzEl’, ‘00:40:00’,‘00:00:00’,cosv=True ) then the first leg of the scan would start at +40’ in azimuth (from the location) and would complete at -40’ in AZ. If instead you used **startOffset** = **Offset**( ‘AzEl’, ‘00:40:00’,‘00:40:00’,cosv=True) the first leg would start at AZ=+40’, EL=+40’, and would go to the opposite (AZ=-40’, EL=-40’).

**scanDuration** A float. It specifies the length of the subscons in seconds.

**slices** An integer. It specifies the number of subscons through location. The default is 4 (making a spider shape – i.e eight legs).

**beamName** A string. It specifies the receiver beam to use for the scan. beamName can be ‘C’, ‘1’, ‘2’, ‘3’, ‘4’ or any valid combination for the receiver you are using such as ‘MR12’. The default is ‘1’.

**unidirectional** A Boolean. It specifies whether each slice is scanned once in one direction or twice in both directions. The default is True (one direction).

**cal** A string. It specifies the order of calibration subscons, i.e., at the beginning of the slice subscon (‘begin’), at the end of the slice subscon (‘end’), or both (‘both’). The default is ‘both’.

**calDuration** A float. It specifies the length of the calibration subscons in seconds. The default is 10.0.

#### USAGE:

Script D.6 generates subscons through 3C 286 starting the first leg 40’ from the source’s “right”. A plot showing the actual trajectory on the sky when the script was executed is shown in figure D.1. Black crosses mark timestamps of data sampled along the red trajectory.

```
Spider('3C286', Offset('AzEl', '00:40:00', 0.0, cosv=True), 80)
```

Script D.6: Spider() example.

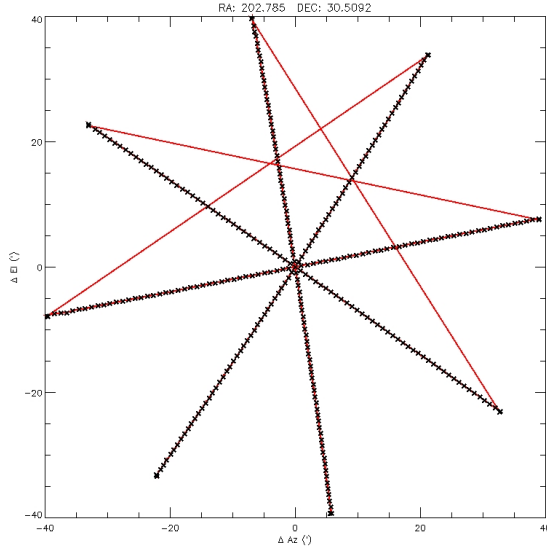


Figure D.1: The actual Spider() trajectory (red) on the sky generated by executing script D.6. Crosses mark timestamps of sampled data. (sampling period is set via `tint` in the configuration).

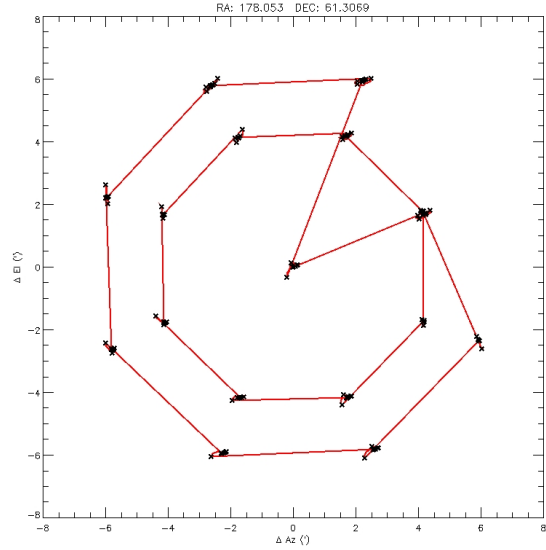


Figure D.2: The actual Z17() trajectory (red) on the sky generated by executing script D.7. Crosses mark timestamps of sampled data at each point.

## D.2.2 Z17

**Z17()** executes two circles of point subscans around location at  $45^\circ$  intervals. The first circle with a radius of `startOffset` and the second circle at a radius of  $\sqrt{2} \cdot \text{startOffset}$ . The initial subscan is at the angle specified by the `startOffset`. After circling twice, the procedure executes a subscan on location. The entire set of 17 subscans each of length `scanDuration`, is sandwiched between two cal subscans of lengths `calDuration` which consist of equal parts calibration noise signal on and off.

### SYNTAX:

**Z17**( location, startOffset, scanDuration, beamName, calDuration )

**location** A Catalog source name or Location object. It specifies the source which is to be tracked.

**startOffset** An Offset object. It specifies the angle from location of the initial subscan as well as the radius of the inner circle.

**scanDuration** A float. It specifies the length of the subscans in seconds.

**beamName** A string. It specifies the receiver beam to use for the scan. beamName can be 'C', '1', '2', '3', '4' or any valid combination for the receiver you are using such as 'MR12'. The is '1'.

**calDuration** A float. It specifies the length of the calibration subscans in seconds. The default is 10.0.

**USAGE:** Script D.7 generates subscan points around G135.1+54.4 starting the first circle at the source's "right". A plot showing the actual trajectory on the sky when the script was executed is shown in figure D.2. Black crosses mark timestamps of data sampled along the red trajectory.

```
Z17('G135.1+54.4', Offset("AzE1", "00:04:30", "00:00:00", cosv=True), 10)
```

Script D.7: Z17() example.



## Appendix E

# Advanced Use of the Balance() Command

You can specify which devices are to be balanced. This overrides the default behavior of `Balance()` and should only be used when absolutely necessary.

**SYNTAX:** `Balance( device, { option : value } )`

**device** A string that may take the following values: 'IFRack', 'DCR\_AF', 'GUPPI', 'vlbi', 'VEGAS', 'RcvrPF\_1', and 'RcvrPF\_2'.

**{option : value}** An optional parameter to the `Balance()` function can be a Python dictionary containing one or more of the balancing options listed below. Items which are not in the dictionary are assigned their default values and non-applicable options are ignored.

**option** is string used to control the balancing. The allowed **value** types depend on the option:

- **'target\_level'** : The target balancing level for the specified **device**. **value** is a float. Default values vary by **device**.
- **'port'** : Used to specify which ports to balance. **value** is an integer list (e.g., [1,2,9,10]). VEGAS ports are in the range of 1–16. The default is to balance all ports.
- **'sample\_time'** : Only applicable when balancing the prime focus receivers. The prime focus balance API will try and balance the receiver over a period of **sample\_time** seconds. This will be repeated a maximum of 6 times or until the power level is within 20% of the **target\_level**. **value** is an integer between 1 and 41 seconds. The default is 2.
- **'cal'** : Turns the noise diodes on or off when balancing the prime focus receivers. **value** can be either 'on' or 'off'. The default is 'on'.

**USAGE:**

```
Balance('RcvrPF_1', {'sample_time':5, 'cal':'on'})
Balance('VEGAS', {'target_level':-20})
```

Script E.1: Advanced Balance() example.



# Appendix F

## Backup Projects

When a scheduled telescope period is cancelled, a backup project will fill the time. Backup projects can come in two categories: observer-run and operator-run.

**Observer-run** backup projects are those for which observers have volunteered to be called on short notice. The notice could be as little as 15 minutes, although the GBT staff will attempt to make the lead time as long as possible. Backup project observers should be ready to take control of the telescope at any time of the day or night, consistent with their observing program and blackout dates. These call-outs are expected to be rare. By volunteering as a backup project, observers improve their project's chances of getting observing time. Note that identifying a project as a backup does not penalize that project during the normal scheduling procedure. The project will compete for regular scheduling on an equal footing with all other projects, but the PI is agreeing to make the project available as a backup in addition to regular scheduling.

**Operator-run** projects contain observing scripts that may be run by the GBT operator, without need for direction from project team members. The observational strategy must be simple. The following criteria must be met in order to be considered for being run by the GBT operators.

1. The project must use only VEGAS or the DCR. Other backends do not currently have standard near real time displays which the operators can use to make sure that the data quality appears acceptable.
2. Minimal calibration requirements, e.g. a single pointing/focus calibration at the beginning of the run. If the observation requires more calibration than a single pointing/focus or simple repetition of a pointing/focus script at regular intervals then it will not qualify as an operator-run candidate.
3. Minimal changes in observing mode.
4. Use of only one receiver.
5. No scientist intervention required. An operator can be expected to determine if a point/focus measurement is reliable but cannot be asked to judge the quality of astronomical data. The operator also cannot be asked to judge which source would be best to observe at any given time. If there is any doubt whether an observation will produce reliable "blind" results then this project is not suitable as an operator-run candidate.
6. The Astrid scripts should be as basic as possible.
7. The operator will not edit scripts. The PI must keep all scripts up to date. Unfortunately, for mapping observations this means that the maps will start over from the beginning if there is a problem encountered while running a script since the same script will be restarted.

8. The operators will not reduce any of the data in gbtidl or using any other data reduction package.
9. The project will be charged for the observing time even if the data quality is not acceptable.
10. At least one telescope period for the project must have been successfully run by the observer.
11. The PI must provide very explicit and concise instructions for the operators to follow and which scripts to run. These must also include an example of what all the data should look like in the Astrid Data Display tab and VEGAS display monitor. Instructions may be stored in the “Project Notes” on the DSS web page.

These requirements bias operator-run projects to low frequency observations, but high frequency projects can be considered as well. There is no intention to implement “service observing” by GBT scientific staff. Green Bank scientific staff will not be on hand to check operator-run projects.

Getting a project approved as an operator-run backup requires consent from the GBT Friend and the GBT DSS staff. To identify your project as a backup project of either sort, inform your GBT Friend.



# Appendix G

## DSS Control Parameters

- **Required Minimum Duration:** Minimum time for scheduling a session
- **Required Maximum Duration:** Maximum time for scheduling a session
- **Time between sessions:** Time which must elapse between the end of one scheduled session period and the start of the next, typically used to allow the observer to sleep or reduce data.
- **Minimum effective system temperature ( $\xi$ ):** Some observers may wish to have their project scheduled even if the weather is not ideal. For example, projects that use very short integration times are dominated by overheads, not the radiometer equation, and so they may wish to get a scoring boost in order to allow observing under a wider range of weather conditions. To implement this desire, we allow observers to modify the minimum effective system temperature,  $T'_{sys}e^{\tau'}$ . This value usually has a default set by the DSS, and depends on observing frequency. Here,  $T_{sys}$  is the atmospheric system temperature and  $\tau$  is the opacity. Recall the atmospheric observing efficiency is  $\eta_{atm} = \left(\frac{T'_{sys}e^{\tau'}}{T_{sys}e^{\tau}}\right)^2$ . (The prime denotes the minimum value). The minimum effective system temperature can be scaled by a factor  $\xi$  to either improve or degrade the atmospheric conditions for a particular session. The default is  $\xi = 1.0$ . The user needs to use caution when modifying this parameter, especially because it will have a different effect at different frequencies. For example, doubling the minimum effective system temperature will have a much larger effect at Ku-band than at K-band.

The parameter  $\xi$  enters into the scoring in two ways. It affects the computation of overall observing efficiency,  $\eta_{total} = \eta_{atm} \times \eta_{tracking} \times \eta_{surface}$  and it also enters by modifying the threshold, minimum value.

- **Tracking Error Threshold, source size, and Tracking Efficiency:** To control the effect of the expected tracking error on scheduling a session, the observer should be able to modify either of two values:
  - $f_{max}$ : the tracking error limit in units of HPBW (default 0.22 for Rcvr68\_92, 0.4 for Rcvr\_PAR and 0.2 for all other receivers)
  - $\theta_{src}$ : the nominal source size in units of arc seconds (default 0.0)

The tracking error is called  $f$ , and the tracking error limit is called  $f_{max}$ . If  $f > f_{max}$  the observation is too inefficient and does not get scheduled. Keep in mind that the tracking error  $f$  comes into play not only in regard to the limit, but also in the scoring equation. The value of  $f$  is ultimately a function of wind speed and observing frequency:

$$f = \frac{\sigma}{\theta_b}$$

where

$$\left(\frac{\sigma}{\text{arcsec}}\right) \simeq \sqrt{\sigma_0^2 + \left(\frac{|v|}{3.5 \text{ m s}^{-1}}\right)^4} \quad (\text{DSPN 18.1 equation 1})$$

and

$$\left(\frac{\theta_b}{\text{arcsec}}\right) \simeq \frac{748}{\nu}$$

with  $\theta_b$  being the HPBW,  $\sigma_0$  being the rms tracking error in the absence of wind, equal to 1.32'' at night and 2.19'' during the day, and  $\nu$  being the observing frequency in GHz.

A value  $f_{max} = 0.2$  assures observers that their flux uncertainty due to tracking errors will be no more than 10%, assuming they are observing a point source ( $\theta_{src} = 0.0$ ). Observers who wish to do better than 10% may decide to specify a smaller value of  $f_{max}$ . For example  $f_{max} = 0.14$  assures no more than 5% flux uncertainty due to tracking errors.

Some observers may wish to relax the tracking restrictions because their source is extended, not point-like. So the most natural way for them to ease the constraint is to specify a source size:  $\theta_{src}$ . The default value is  $\theta_{src} = 0.0$  arc seconds. If the user specifies  $\theta_{src}$  then the tracking error  $f$  should be calculated as follows:  $f = \frac{\sigma}{\theta_{obs}}$  where  $\theta_{obs} = \sqrt{\theta_{src}^2 + \theta_b^2}$ . This new value for the tracking error must then be used in calculating the tracking efficiency,  $\eta_{tr}$ . So changing the source size impacts the scoring equation through both the tracking efficiency and the tracking error limit.

So to summarize, most observers will not need to modify  $f_{max}$  or  $\theta_{src}$ . If they do, it would be most sensible to modify only one or the other. If observing point sources, the observer may wish to change  $f_{max}$  to tighten or loosen the requirements. If observing extended sources, the observer should stick with  $f_{max} = 0.2$  and change the value of  $\theta_{src}$ .

Specifying both  $f_{max}$  and  $\theta_{src}$  need not be forbidden by the software, but it is probably not the best approach.

- **Irradiance:** Section 3.4.5 of DSPN5 describes the concept of irradiance, an important concept for continuum observations above 2 GHz. Most continuum observations prefer the default values of irradiance (300 W M<sup>-2</sup>), there are times when the value should be tunable, and any value of irradiance may be used for the DSS.
- **Elevation Limit:** This parameter allows the observer to modify the frequency dependent hard elevation limit, and instead set it to any value (in degrees). When using this parameter, the hour angle scoring factor will be set to zero when the source for a session is less than the minimum allowed elevation and set to one otherwise.
- **Solar Avoidance:** The angle by which the project must avoid the sun. The default here is 0.
- **Time of Day:** Time of day restrictions for this session. Options are: Any Time of Day (default); RFI (8pm - 8am); and PTCS (sunset - sunrise+2hours).
- **Transit:** The central coordinates of this session must pass through transit, with at least 25% of the session on either side of the transit window.
- **LST Exclude/Include:** This allows the session to exclude/include LST ranges when scheduling. More than one range can be give, but they must be listed sequentially.
- **Keyhole Limit:** Boolean to set a maximum elevation, specified by the sessions primary (first in list) receiver. When set to true, sessions not requiring Mustang will not be scheduled when their source will be above 80 degrees in elevation during the duration of the telescope period; Mustang observations will not be scheduled when the source will be above 78 degrees in elevation during the duration of the telescope period.
- **Good Atmospheric Stability (GAS):** The atmospheric stability limit,  $\ell_{st}$ , is a factor in the scoring algorithm (see DSPN5). It is used only for continuum observations which are sensitive to atmospheric fluctuations. Currently, a forecast downward irradiance,  $I_{down}$ , threshold value of

300 W/mi<sup>2</sup>, is used to derive  $\ell_{st}$ . However, a different metric has been developed for the 90 GHz Bolometer array, MUSTANG, that uses the atmospheric system temperature (including hydrosols) at the target position elevation. GAS is used to set the value of  $\ell_{st}$  for MUSTANG only and is ignored for all other receivers, as follows:

For MUSTANG **only** derive the atmospheric stability limit by first calculating the zenith atmospheric system temperature at 90 GHz. (This includes hydrosols, the default in the CLEO command line interface, and is described in equation 7 in DSPN5.) The atmospheric system temperature is the last term on the right hand side of the equation,  $T_{sys}^{atm}(El = 90^\circ) = T_k(1 - e^{(EL=90^\circ)})$ . Then derive  $T_{sys}^{atm}(El = 90^\circ)/Sin(El)$ , the the low opacity atmospheric system temperature. For MUSTANG, assume a frequency of 90 GHz. When GAS is TRUE ("good") then  $T_{sys}^{atm}(El = 90^\circ)/Sin(El) < 35K$  then  $\ell_{st}=1$ ; otherwise  $\ell_{st}=0$ . If GAS is FALSE ("usable") then if  $T_{sys}^{atm}(El = 90^\circ)/Sin(El) < 50K$  then  $\ell_{st}=1$ ; otherwise  $\ell_{st}=0$ .

For all other receivers derive the atmospheric stability limit as usual: if  $I_{down} \geq 300$  W/mi<sup>2</sup> then  $\ell_{st}=1$ ; otherwise  $\ell_{st}=0$ .



# Appendix H

## VEGAS Reference

### H.0.1 VEGAS ADC (MMCM, OGP and INL) Calibration

*An understanding of VEGAS ADC calibration is not necessary in order to be able to observe with the instrument. It is included here as background information only; those who are not interested in it may safely skip to section n. The precise method of calibrating the ADCs is still evolving; this description is correct as of December 2014*

Each VEGAS Bank contains two Analog to Digital Converter (ADC) cards, one for each polarization. The ADC cards used are National Semiconductor (now Texas Instruments) ADC083000s, which perform 8-bit sampling at 3 Gigasamples per second (Gs/s). The ADC083000 has a 1:4 demultiplexer (data is output on four busses at a quarter of the ADC sampling rate). The outputs are interleaved to provide output words at the full conversion rate. Use of these ADCs require a number of calibration steps.

“Mixed-mode Clock Management (MMCM)”. The ADC clock is driven by the FPGA on the Roach II card. The “MMCM phase calibration”, consists of calibrating the FPGA clock phase relative to the ADC inputs to avoid glitches when the FPGA captures the data samples. The MMCM calibration depends on the mode (FPGA clock frequency and BOF file) in use.

MMCM calibration is performed automatically by the VEGAS Manager, whenever a new mode is selected. This calibration step is stable and routine.

“Offset, Gain and Phase (OGP)”. Each core of the ADC may have a separate Offset, Gain and Phase. Currently, these are calibrated off-line, by injecting a test-tone, and adjusting the offset, gain and phase of the ADC counts to match the injected tone as closely as possible. Once the OGP calibration has been performed, the optimum values are stored in the VEGAS config file, and downloaded by the Manager to the firmware whenever the mode changes. The OGP values also depend on the mode (the FPGA clock frequency) in use.

“Integral non-linearity (INL)”. INL is a measure of the deviation of each individual 8-bit value from a straight line through the input to output transfer function. The deviation of any given value from a straight line is measured from the center of that value. INL calibration is also done off-line. It is independent of VEGAS Mode, so the appropriate values are stored in the config file, and simply downloaded to the hardware when the manager is turned “ON”.

### H.0.2 IF Levels and Balancing

Once the IF system system from the receiver to the IF rack is balanced, the VEGAS balance API then adjusts the attenuators in the converter rack to balance the power level at the inputs to the VEGAS ADCs to be (by default) -20dBm.

For VEGAS modes higher than three, digital filters are implemented in the FPGA. The filtered output is requantized to 8 bits; a digital gain adjustment is implemented in the FPGA to scale the digital filter output values so that there will be no overflow after requantization.

Both of these steps are performed when the Astrid `Balance()` directive is issued. A new keyword has been added to the `Balance()` python dictionary (see Appendix D) which can be used to change the target input level to VEGAS, for example:

```
Balance('VEGAS', 'target_level':-20)
```

will set the input target level to -20 dBm for both the ADC and requantizer balance.

The IF system must be able to handle the large bandwidths all the way to VEGAS, which then splits the signal into the more narrow 1.25 GHz bandwidths of the individual VEGAS banks. This can be problematic, as part of balancing the entire bandwidth is based on the signal received by a single VEGAS bank. After the correct attenuation is set at the converter module for one bank, the system can potentially require amplification for other portions of the bandpass. This can cause the system gain to transition to the non-linear regime, where changes in the input signal do not linearly change the recorded output. We have found that using when using a 3.5 GHz bandwidth, the system was able to properly configure across all frequencies. We were not able to sufficiently test using larger than 3.5 GHz of total observed bandwidth across the system. It is recommended that users hoping to push the bandwidth limit should watch the attenuation levels in the converter module, and if necessary discuss the optimum settings with their support scientist.

### H.0.2.1 vegas\_status

*This section is only relevant for advanced diagnosis of VEGAS problems. It may be skipped by the average Observer.*

VEGAS utilizes eight high performance computers (HPCs), one for each Bank. Each HPC contains a Graphics Processor Unit (GPU) which is used to perform much of the digital processing for modes 4 and above. These HPCs are also responsible for collating metadata, and writing this plus the actual spectra to disk. These machines are called VEGAS-HPC1 through VEGAS-HPC8.

VEGAS meta-data is stored in shared memory on these machines, and it is possible to monitor the status of this meta-data as follows:

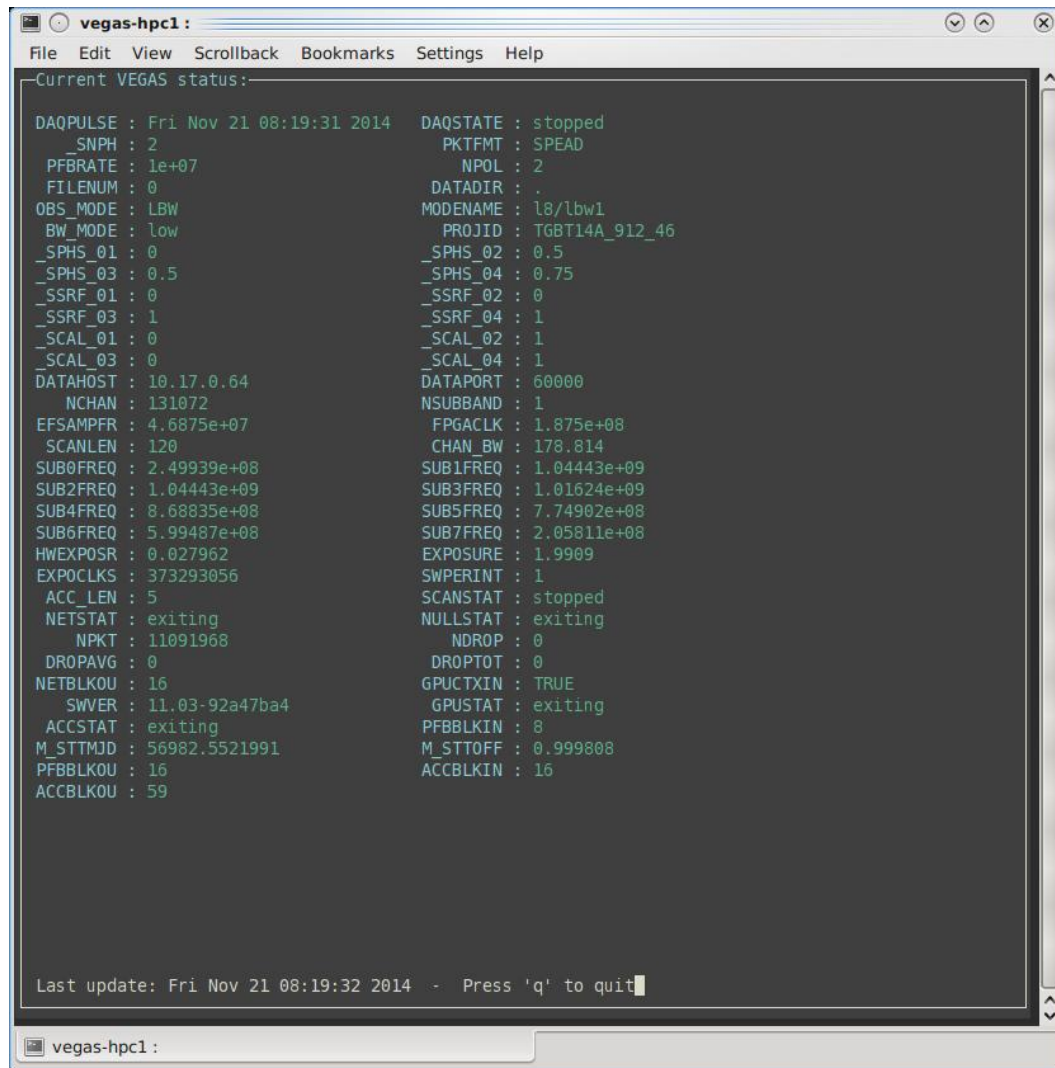
```
ssh into the machine of interest, then
% source /home/gbt/gbt.bash (or .csh)
% vegas_status
```

This will bring up a terminal status display, as shown in Figure H.1.

Observer's should not normally need to use this tool, but it is possible on occasion that by running it they might be able to give support scientists / engineers some useful information in the case of problems.

## H.1 Baseline Performance

Gain variations across the large bandpasses desired in radio astronomy require significant efforts to balance and stabilize. These variations create baselines that are not perfectly flat and can potentially vary with time. Thorough data calibration helps to mitigate these effects, but calibration can only be effective if the baselines are stable. With this in mind, developing the VEGAS infrastructure has focused on the temporal and spectral stability of baselines. It has served as an interesting engineering problem to create an IF system that is capable and stable for all receivers. In this section, we will discuss the status of the baselines as recorded by VEGAS.



```

vegashpc1:
File Edit View Scrollback Bookmarks Settings Help
Current VEGAS status:
DAQPULSE : Fri Nov 21 08:19:31 2014   DAQSTATE : stopped
  _SNPH : 2                           PKTFMT : SPEAD
PFBRATE  : 1e+07                       NPOL  : 2
FILENUM  : 0                           DATADIR : .
OBS_MODE : LBW                         MODENAME : l8/lbw1
BW_MODE  : low                          PROJID : TGBT14A_912_46
_SPHS_01 : 0                           _SPHS_02 : 0.5
_SPHS_03 : 0.5                         _SPHS_04 : 0.75
_SSRF_01 : 0                           _SSRF_02 : 0
_SSRF_03 : 1                           _SSRF_04 : 1
_SCAL_01 : 0                           _SCAL_02 : 1
_SCAL_03 : 0                           _SCAL_04 : 1
DATAHOST : 10.17.0.64                  DATAPORT : 60000
NCHAN    : 131072                      NSUBBAND : 1
EFSAMPFR : 4.6875e+07                  FPGACLK : 1.875e+08
SCANLEN  : 120                         CHAN_BW  : 178.814
SUB0FREQ : 2.49939e+08                  SUB1FREQ : 1.04443e+09
SUB2FREQ : 1.04443e+09                  SUB3FREQ : 1.01624e+09
SUB4FREQ : 8.68835e+08                  SUB5FREQ : 7.74902e+08
SUB6FREQ : 5.99487e+08                  SUB7FREQ : 2.05811e+08
HWEXPOSR : 0.027962                    EXPOSURE : 1.9909
EXPCLKS  : 373293056                   SWPERINT : 1
ACC_LEN  : 5                            SCANSTAT : stopped
NETSTAT  : exiting                      NULLSTAT : exiting
NPKT     : 11091968                     NDRROP   : 0
DROPAVG : 0                             DROPTOT : 0
NETBLKOU : 16                           GPUCTXIN : TRUE
SWVER    : 11.03-92a47ba4               GPUSTAT  : exiting
ACCSTAT  : exiting                      PFBBLKIN : 8
M_STTMJD : 56982.5521991                M_STTOFF : 0.999808
PFBBLKOU : 16                           ACCBLKIN : 16
ACCBKOU  : 59

Last update: Fri Nov 21 08:19:32 2014 - Press 'q' to quit
vegashpc1:

```

Figure H.1: The VEGAS shared memory screen

The baseline shapes come from two main causes; variations due to the receiver and IF system, and shapes due to digital filters within VEGAS. Figure H.2 shows the baselines types expected from VEGAS. While not flat, all of these baseline shapes have been found to be stable out to at least 30 minutes, and are easily removed with standard calibration techniques

When looking at Figure H.2, the ripples in the mode 4-9 data (lower left) stand out. This pattern is caused by the Finite Impulse Response (FIR) filter used in the digital processing of low bandwidth modes. This pattern is stable, and as long as the amplification within the IF system is kept in the linear regime (see Section H.0.2), this pattern will not effect the final calibrated data. Modes 10-29 show a similar pattern, with just a single ripple.

Secondary ripples (10MHz wide) are visible in the mode 3 bandpass shown in the upper right of Figure H.2. These ripples are due to a standing wave in the cabling between the Converter Rack and VEGAS. Placing attenuation on the cables at the converter rack minimizes these standing waves while leaving the signal strength and signal-to-noise ratio of the signal intact. The ripples in the mode 2 image (upper left) have a lower amplitude, since this data was taken with the attenuation in place.

**In the next version of this guide, we will replace the mode 3 figure with one taken after**

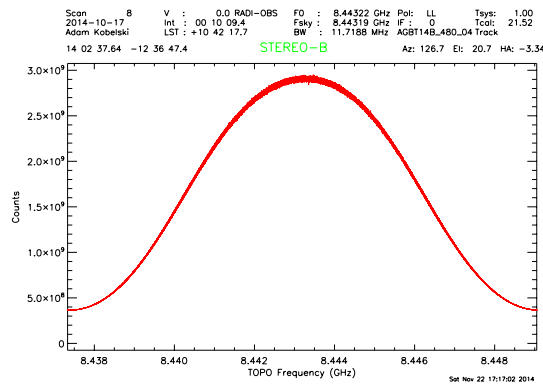
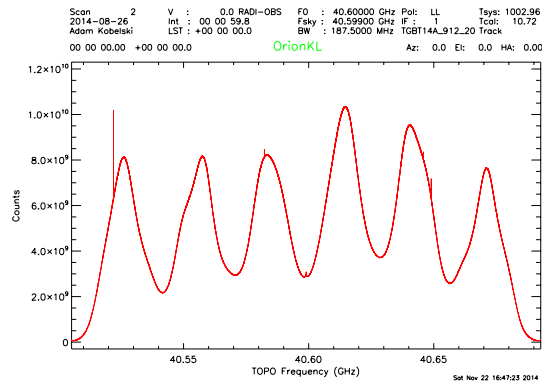
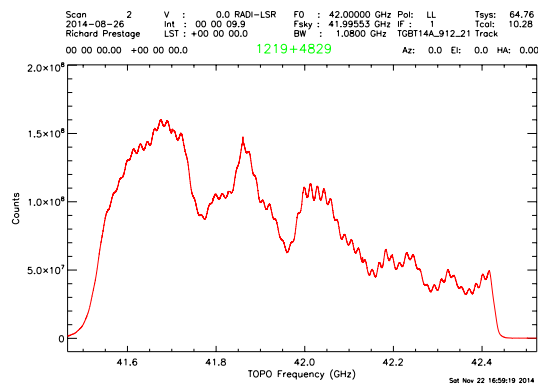
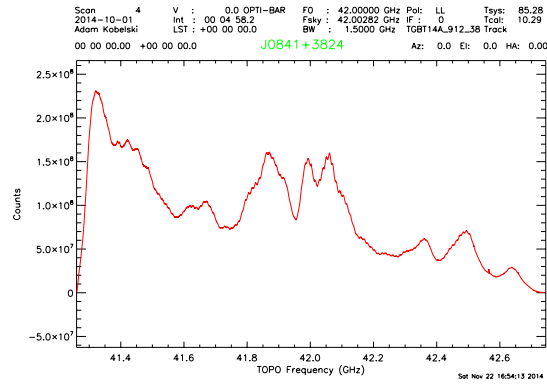


Figure H.2: VEGAS bandpass for the four families of VEGAS modes. The upper left represents the 1500 MHz bandwidth of modes 1 and 2, the upper right is mode 3, the lower left shows the bandpass for mode



the attenuators were in place.

As mentioned previously, the stability of the baseline is more important than the overall baseline shape. The thermal stability of the system greatly impacts the temporal stability of the baselines. We have attempted to isolate all cables and VEGAS itself to minimize these impacts, and are currently in the process of further improving the thermal isolation of the system. The calibration of the digital systems within VEGAS can also affect the temporal stability of the baselines; the software/digital team has worked diligently to successfully track down and alleviate these issues.

Removal of all baseline shapes is not feasible, so it is important to account for these shapes in data calibration. This is most often accomplished by using position switching (obtaining data when pointing on and off source) and/or frequency switching (shifting the central frequency such that the desired spectral lines appear at different locations within the bandpass shape). If the shape of the baseline is still of concern after these types of calibration, it is recommended to look into vectorized baseline removal (such as described in Winkel, Kraus, & Bach (2012) <http://adsabs.harvard.edu/abs/2012A%26A...540A.140W>).

## H.2 VEGAS and Deep Integrations

At the time of writing (December 2014), we have not extensively tested the performance of VEGAS for deep integrations.

However, some science observations have been performed with VEGAS at Q-band, looking for high-redshifted CO lines. Figure H.2 shows the result of four and a half hours of effective integration time using the “sub beam nod” observing technique (D. Frayer, private communication). The previous vector calibration, as applied for example to the GBT spectrometer yields bad baselines with VEGAS. However, a simple scalar  $T_{sys}$  calibration works well, when applying an appropriate high-pass filter to remove large-scale smooth frequency structures. The sub beam nod observations were taken with a nod period of 6 seconds, and 0.2 second integration times to minimize the blanking. The high frequency ripples then present in VEGAS data, and the “spikes” (see § 8.7.1) calibrate away nicely. This spectrum consists of 4 overlapping Mode 1 spectra; no special processing was required to “stitch” the spectra together; overlapping channels were simply dropped. After the above processing was performed, the scalar processing yielded a nice flat baseline, and noise that is consistent with the radiometer equation.

## H.3 The Spurs

When attempting to search for RFI with VEGAS by running a high-pass filter through the data, significantly more spikes/spurs were found than naively expected. These spurs could be found in the same bins in relatively RFI free wavelengths, such as Q-band. The spurs appear at the same location (in bin space) for a given mode and have relatively stable amplitudes. These faint spurs are not always directly visible in the data, but became clear when high-pass filtered, as shown in Figure 8.2. After significant testing, it was determined that these spurs are below the spurious-free dynamic range of -60dBc specified by the manufacturer, and cannot be fully removed. In overly simplistic terms, the spurs are caused by the leaking of the FPGA clock into the four interleaved ADCs.

In overly simplistic terms, the spurs are caused by the leaking of the FPGA clock into the four interleaved ADCs. This causes a rail of thin spikes/spurs to be inserted into the VEGAS IF spectrum at frequencies:

$$\nu_{\text{spurs}} = \frac{i * f_s}{64}, \quad (\text{H.1})$$

where  $0 \leq i \leq 64$  and  $f_s$  is the sample rate of the ADC. Since the bandwidth of the FFT can only be half of  $f_s$ ,  $i$  actually runs from  $0 \leq i \leq 32$ , meaning there will never be more than 33 spurs in a given data

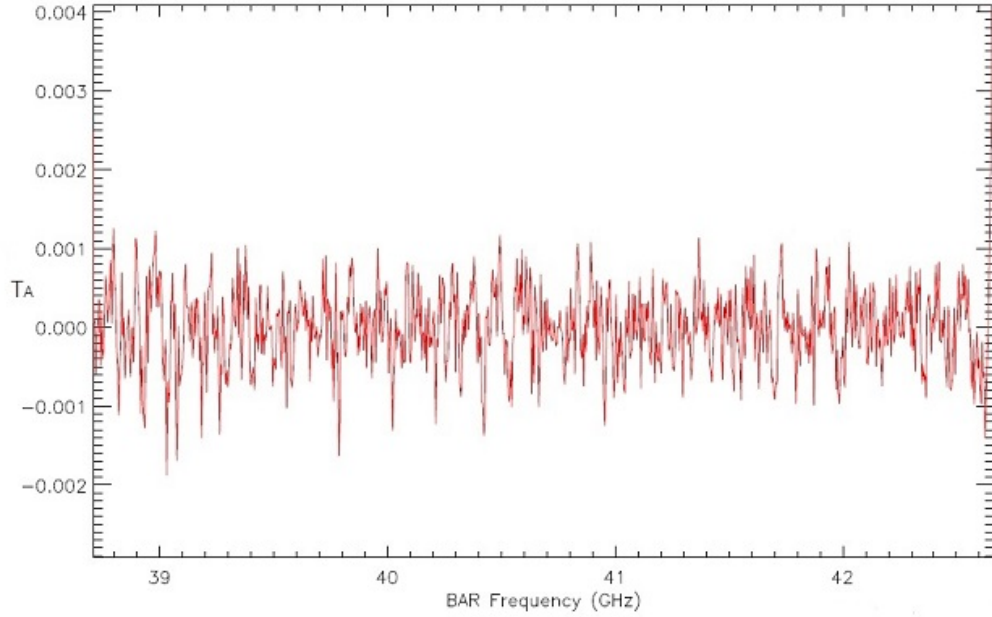


Figure H.3: A deep integration with VEGAS

set<sup>1</sup>. Note that these frequencies are relative to the intermediate frequency input into VEGAS, so they should be offset by  $\nu_{V,c}$  (the centered VEGAS specific intermediate frequency) and  $\nu_{s,c}$ , the center sky frequency, such that the spurs will occur at sky frequencies

$$\nu_{\text{spurs,sky}} = \nu_{s,c} + (\nu_{\text{spurs}} - \nu_{V,c}), \quad (\text{H.2})$$

as shown in Figure H.4.

It is important to keep in mind that  $\nu_{s,c} = \mathcal{D}(\nu_{s,r})$ ; the central sky frequency may be shifted by a function  $\mathcal{D}$  from the requested sky frequency,  $\nu_{s,r}$  (e.g. via doppler tracking). The value of  $\nu_{V,c}$  is dependent on which mode is used, and is not as straightforward when using the low bandwidth (higher resolution) modes 11-29. For modes 4-29, the FFT of the raw time series from the ADCs is not performed on the FPGA but on external GPUs, which further filter the bandwidth (allowing higher resolution), and thus all 33 spurs are not present in these modes. This filtering in the GPU modes also causes  $\nu_{V,c}$  to not simply equal to  $f_s/4$  in these cases as it is in modes 1-3. Additionally, for modes 4-29, the signal is decimated after being read by the ADCs, and requires an additional scaling to be included into Equation H.2. Table H.1 lists  $\nu_{V,c}$  for the various modes. The exact formula for these modes is still being worked out, and will be included as soon as possible.

Table H.1:  $\nu_{V,c}$  for each VEGAS mode

modes	$\nu_{V,c}$
1,2	750
3	540
4-6	750
7-9	400
10-29	TBD

<sup>1</sup>The  $33 \leq i \leq 64$  spurs alias back to the  $0 \leq i \leq 32$  spurs when creating the power spectrum

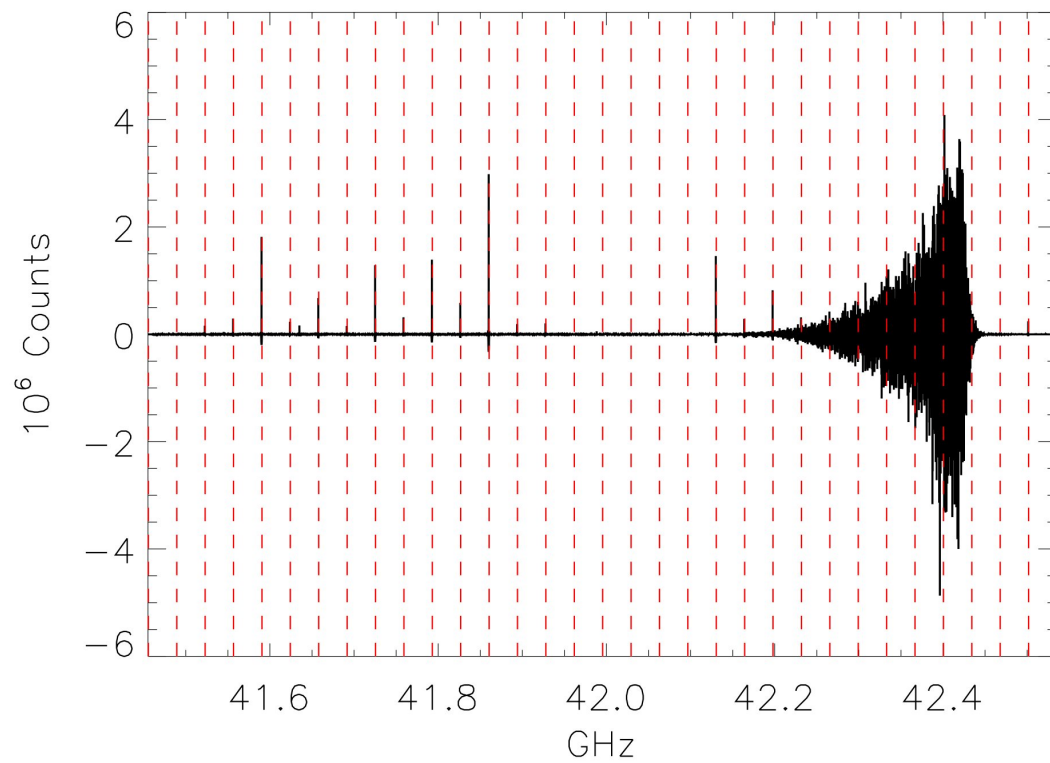


Figure H.4: Data from Figure 8.2 after the high pass filter in black, with the vertical red lines denoting the locations of the spurs as dictated by Equation H.2

These spurs are relatively stable:  $\nu_{\text{spurs}}$  will remain constant (for a given mode) and the magnitude of the spurs is relatively constant. These features are also quite small by most standards (Spurious Free Dynamic Range no more than -60dBc), but nevertheless can be problematic when looking for faint narrow features. The stability of these features allows them to be removed by standard data practices (such as position and/or frequency switching), but they are an added noise source which can bleed through to the final product. Due to the limited and often negligible effect of these spurs, we do not automatically interpolate across them, but let the user decide how to handle those channels. A tool for locating these spurs within GBTIDL is forthcoming.



# Glossary

- A** The number of air masses along the line of sight. One air mass is defined as the total atmospheric column when looking at the zenith. 185
- ADC** Analog to Digital Converter. A card used to convert an analog signal into a quantized digital signal. Each VEGAS Bank contains two ADC cards, one for each polarization. 32, 51, 68, 124, 135, 136, 223, 224, 227
- Analog Filter Rack** A rack in the GBT IF system that provides contains filters to provide GUPPI and the DCR with signals of the proper bandwidth. 33, 63
- API** Application Programming Interface. A set of routines, protocols and tools that can be used when building software and applications for a specific system. 22, 32, 215, 223
- AS** Active Surface. The surface panels on the GBT whose corner heights can be adjusted to form the best possible parabolic surface. 3, 10, 30, 121
- Astrid** Astronomer’s Integrated Desktop. The software tool used for executing observations with the GBT. 1, 10, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 31, 35, 36, 37, 39, 43, 44, 47, 53, 54, 55, 69, 71, 83, 99, 108, 109, 110, 123, 126, 127, 134, 165, 172, 178, 179, 189, 191, 205, 224
- baseline** Baseline is a generic term usually taken to mean the instrumental plus continuum bandpass shape in an observed spectrum, or changes in the background level in a continuum observation. 36, 85, 123, 184
- beam switching** The Ka-band (26-40 GHz) receiver is the only receiver that can perform beam switching. The switching can route the inputs of each feed to one of two “first amplifiers” which allows the short time-scale gain fluctuations to be removed from the data. This type of switching is only recommended for continuum observations. Total power mode is recommended for Ka-band dual-beam nodded observations using VEGAS as the backend. 66
- beam-width** The FWHM of the Gaussian response to the sky, the beam, of the GBT. 38
- C-band** A region of the electromagnetic spectrum covering 4–8 GHz. 6, 63, 67, 177
- CCB** Caltech Continuum Backend. A wideband continuum backend designed for use with the GBT Ka-band receiver. 1, 7, 10, 41, 90
- CLEO** Control Library for Engineers and Operators. A suite of utilities for monitoring and controlling the GBT hardware systems. 13, 16, 38, 46, 47, 48, 50, 51, 124, 134, 191
- Converter Rack** A rack in the GBT IF system that receives the signal from the optical fibers (sent from the IF Rack), mixes the IF signal with LO2 and LO3 references, and then distributes the IF signal to the various backends. 69, 202, 203
- DCR** The Digital Continuum Receiver. A continuum backend designed for use with any of the GBT receivers. 7, 33, 56, 87, 88, 90, 93, 123, 126, 165, 202

- DDC** Digital Down Converter. Converts a digitized real IF signal to a complex baseband signal. 171
- DSS** Dynamic Scheduling System. The DSS examines the weather forecast, equipment availability, observer availability, and other factors in order to generate an observing schedule. 9, 11, 12, 13, 14, 16, 17, 18, 27, 125, 183, 191
- Dynamic Corrections** A system that uses temperature sensors located on the backup structure of the GBT to correct for deformations in the surface, and deformations that change the pointing and focus of the GBT. 30
- EVN** European VLBI Network. A collaboration of the major radio astronomical institutes in Europe, Asia and South Africa. 171
- FAA** Federal Aviation Administration. The U.S. Government agency that oversees and regulates the airline industry in the U.S. 68
- FEM** Finite Element Model. This is a model for how the GBT support structure changes shape due to gravitational forces at different elevation angles. 30
- FET** Field Effect Transistor. A type of amplifier used in the receivers. 6
- FPGA** Field-Programmable Gate Array. An integrated circuit designed to be programmed in the field after manufacture. 32, 129, 135, 136, 223, 227
- frequency switching** A calibration method that obtains blank sky information while keeping the telescope pointed at the object of interest. The central frequency is shifted such that the desired spectral lines appear at different locations within the bandpass shape. 32, 57, 66, 132, 133
- FRM** Focus Rotation Mount. A mount that holds the Prime Focus Receivers which allows the receivers to be moved and rotated relative to the focal point. The FRM has three degrees of freedom, Z-axis radial focus, Y-axis translation (in the direction of the dish plane of symmetry), and rotation. 6
- FWHM** Full Width at Half the Maximum. Used as a measure for the width of a Gaussian. 37, 38, 42, 85, 100, 105, 125, 126
- GBT** Green Bank Telescope. 1, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 19, 20, 22, 23, 24, 26, 27, 28, 29, 30, 32, 33, 35, 36, 37, 38, 43, 46, 47, 48, 53, 54, 61, 63, 66, 67, 69, 76, 84, 85, 87, 92, 93, 94, 98, 99, 108, 114, 121, 123, 124, 125, 126, 129, 131, 151, 153, 165, 171, 172, 173, 174, 176, 177, 178, 181, 184, 185, 191, 193, 199, 201, 207
- GBTIDL** Green Bank Telescope Interactive Data Language. The GBT data reduction package written in IDL for analyzing GBT spectral line data. 1, 10, 108, 135, 137, 199
- GFM** GBT Fits Monitor: The software program that provides a real time display for GBT data. 19, 35, 37, 38, 45, 165
- GO** GBT Observing. 30, 108
- GUI** Graphical User Interface. 19, 20, 21, 22, 23, 24
- GUPPI** The Green Bank Ultimate Pulsar Processing Instrument. An FPGA + GPU backend for use with GBT pulsar observations. 1, 7, 8, 68
- IDL** The Interactive Data Language program of ITT Visual Information Solutions. 232
- IF** Intermediate Frequency. A frequency to which the Radio Frequency is shifted as an intermediate step before detection in the backend. Obtained from mixing the RF signal with an LO signal. 7, 19, 32, 43, 69, 123, 124, 127, 202, 203, 205, 206, 207, 224

- IF Rack** A rack in the GBT IF system where the IF signal is distributed onto optical fibers and sent from the GBT receiver room to the GBT equipment room where the backends are located. A signal may also be sent directly to the DCR. 33, 63, 69, 94, 123, 202
- IF path** Intermediate Frequency path. The actual signal path between the receiver and the backend through the IF system. 33, 60, 66, 87, 123, 201
- IF system** Intermediate Frequency system. A general name for all the electronics between the receiver and the backend. These electronics typically operate using an Intermediate Frequency (IF). 10, 50, 53, 54, 66, 69, 75, 85, 93, 94, 123, 124, 129, 131, 201, 205, 223
- ITRF** International Terrestrial Reference Frame. A world spatial reference system co-rotating with the Earth in its diurnal motion in space. 176
- JD** Julian Date. A continuous count of days since the beginning of the Julian period (12h Jan 1, 4713 BC). 233
- K-band** A region of the electromagnetic spectrum covering 18–26 GHz. 18, 69
- Ka-band** A region of the electromagnetic spectrum covering 26–40 GHz. 6, 7, 41, 63, 66, 67, 69, 85, 90, 121, 165
- KFPA** The K-band Focal Plane Array receiver covering 18-26.5 GHz. 1, 6, 60, 63, 66, 69, 87, 98, 131, 151, 152, 153, 170, 205, 206
- Ku-band** A region of the electromagnetic spectrum from 12–18 GHz. 6, 63, 69
- L-band** A region of the electromagnetic spectrum covering 1–2 GHz. 6, 56, 57, 63, 67, 68, 69, 172
- LFC** Local Focus Correction. Corrections for the general telescope focus model that are measured by the observer. 30, 41, 42
- LO** Local Oscillator. A generator of a stable, constant frequency, radio signal used as a reference for determining which radio frequency to observe. 32, 69, 137, 205, 206
- LO1** The first LO in the GBT IF system. This LO is used to convert the RF signal detected by the receiver into the IF sent through the electronics to the backend. This is also the LO used for Doppler tracking. 69, 75, 132, 133, 202, 205, 211
- LO2** Second LO. The second LO in the GBT IF system. This is actually a set of eight different LOs that can be used to observe up to eight different spectral windows at the same time. 33, 69, 203
- LO3** Third LO. The third LO in the GBT IF system which operates at a fixed frequency of 10.5 MHz. 33
- LPC** Local Pointing Correction. Corrections for the general telescope pointing model that are measured by the observer. 30, 41, 42
- LST** Local Sidereal Time. A time scale based on the Earth's rate of rotation measured relative to the fixed stars rather than the Sun. 19, 29, 47
- M&C** Monitor and Control. The group of software programs which control the hardware devices which comprise the GBT. 19, 22, 23, 36, 37, 85, 88, 121, 209, 210
- MJD** Modified Julian Date.  $MJD = \text{Julian Date (JD)} - 2400000.5$ . 29
- MUSTANG** The MULTiplexed SQUID TES Array at Ninety GHz 80-100GHz bolometer receiver. 7, 63, 90, 121, 129, 201, 210

- NAD83** North American Datum of 1983. An earth-centered model for the Earth’s surface based on the Geodetic Reference System of 1980. The size and shape of the earth was determined through measurements made by satellites and other sophisticated electronic equipment; the measurements accurately represent the earth to within two meters. 4, 5
- NAVD88** The North American Vertical Datum of 1988. 4
- noise diode** A device with a known effective temperature that is coupled to the telescope system to give a measure of system temperature ( $T_{sys}$ ). When the telescope is pointed on blank sky, the noise diode is turned on and then off to determine the off-source system temperature. This device is also referred to as the “Cal”. 32, 56, 57, 58, 59, 60, 66, 131, 132, 133
- NRAO** National Radio Astronomy Observatory. The organization that operates the GBT, VLA, VLBA and the North American part of ALMA. 5, 10, 11, 12, 13, 14, 171, 177, 193, 199
- NRQZ** National Radio Quiet Zone. An area ( $\sim 34,000$  km<sup>2</sup>) around the GBT set up by the U.S. government to provide protection from RFI. 5
- OMT** Ortho-Mode Transducer. This is part of the receiver that takes the input from the wave-guide and separates the two polarizations to go to separate detectors. 6
- OOF** Out-Of-Focus holography. A technique for measuring large-scale errors in the shape of the reflecting surface by mapping a strong point source both in and out of focus. 39, 40, 41, 42
- OTF** On-The-Fly. On-The-Fly mapping scans take data while the telescope pointing moves between two points on the sky. This move is usually done in a linear fashion with constant slewing speed with respect to the sky. 39, 89, 99, 100, 102, 105, 121, 133, 165
- P–band** A region of the electromagnetic spectrum covering 300–1000 MHz. Also known as the Ultra High Frequency (UHF) band in the U.S. (Sometimes P–band is considered to be a narrow region around 408 MHz, while A–band is the region around 600 MHz). 177
- PF1** The first of two prime focus receivers for the GBT. This receiver has four different bands: 290–395, 385–520, 510–690 and 680–920 MHz. 6, 63
- PF2** The second of two prime focus receivers for the GBT. This receiver covers 901–1230 MHz. 6, 63
- PFS** A radar data acquisition backend. 8, 177, 178
- PI** Principle Investigator. 12, 13, 16, 27
- polarization switching** This is only available for the L and X-band receivers. During an observation and at a rate of about once per second, the polarization of the observation is switched between two orthogonal linear polarizations or the two circular polarizations. This switching method is used almost exclusively for Zeeman measurements. 66
- position switching** A calibration method that involves observing an object of interest period of time, and then moving the telescope to a blank sky region to obtain the blank sky observations necessary for baseline subtraction. Nodding is a form of position switching. Position switching is done via an observing routine and is not setup in hardware unlike other switching schemes. 85
- PRESTO** PulsAR Exploration and Search TOolkit: A software package used to analyze pulsar observations. 10
- PROCNAME** A GO FITS file keyword that contains the name of the Scan Type used in Astrid to obtain the data. 30
- PROCSEQN** A GO FITS file keyword that contains the current number of scans done of the total scans given by PROCSIZE in a given Scan Type. 30



- PROCSIZE** A GO FITS file keyword that contains the number of scans that are to be run as part of the Scan Type given by PROCNAME. 30
- Q–band** A region of the electromagnetic spectrum from 40–50 GHz.. 6, 59, 63, 121, 136, 227
- RDBE** A Roach Digital Backend, where ROACH is the core board containing a large FPGA. 171
- RF** Radio Frequency. The frequency of the incoming radiation detected by the GBT. 7, 33, 68, 69
- RFI** Radio Frequency Interference. Light pollution at radio wavelengths. 1, 4, 7, 9, 69, 121, 129, 136, 181, 184, 207, 227
- S–band** A region of the electromagnetic spectrum covering 2–4 GHz. 6, 63, 67, 177
- SB** Scheduling Block. A Python script used to perform astronomical observations with the GBT. 1, 9, 10, 16, 19, 20, 23, 24, 25, 26, 27, 28, 30, 42, 53, 54, 55, 71, 72, 75, 76, 77, 78, 83, 84, 87, 95, 107, 108, 109, 110, 112, 114, 116, 120, 126, 152, 153, 165, 178, 180, 191, 193, 207, 209, 210
- $\tau$  The opacity of the atmosphere. 185
- $T_{\text{src}}$  The equivalent blackbody temperature brightness from the astronomical source. 123
- $T_{\text{sys}}$  The total equivalent blackbody temperature brightness that the GBT sees. Depending on usage, it may or may not include  $T_{\text{src}}$ . 6, 85, 123
- $T_{\text{rec}}$  The equivalent blackbody temperature brightness that the GBT receiver contributes to the detected signal. 6
- TLE** Two-Line Element. 71, 84
- total power** Spectral-line observing typically requires differencing “signal” and “reference” observations so as to remove the instrumental bandpass shape. In total power observing, the reference observations are either separate scans (as acquired with, for example, Astrid’s OnOff or OffOn observing directives), as separate integrations in an on-the-fly observations (for example, as edge pixels in a map, or as separate integrations in some types of subreflector nodding observations. “Switched Power”, the alternative to “Total Power”, provides faster switching between signal and reference observations but, in some cases, worse baseline shapes. 32, 56, 58, 59, 60, 99, 132, 133
- UTC** Coordinated Universal Time. The mean solar time at 0° longitude. 19, 29, 31
- VEGAS** The GBT spectral line backend. 1, 7, 8, 32, 33, 43, 49, 50, 51, 57, 58, 59, 60, 61, 62, 64, 65, 66, 67, 68, 93, 124, 126, 129, 152, 177, 205, 215, 231
- $v_{\text{relativistic}}$  The velocity of a source using the relativistic definition of the velocity–frequency relationship. 67
- $v_{\text{optical}}$  The velocity of a source using the optical approximation of the velocity–frequency relationship. 67
- $v_{\text{radio}}$  The velocity of a source using the radio approximation of the velocity–frequency relationship. 67
- VLB** Very Long Baseline: A general acronym for VLBI or VLBA. 8, 171
- VLBA** Very Long Baseline Array: An interferometer run by the NRAO. 7, 8, 32, 171, 172, 173, 174
- VLBI** Very Long Baseline Interferometer: The use of unconnected telescopes to form an effective telescope with the size of the separation between the elements of the inteferometer. 1, 17, 67, 69, 171, 172

**VNC** Virtual Network Computer. A GUI based system that is platform independent that allows you to view the screen of one computer on a second computer. This is very useful for remote observing. 9, 10, 13, 46, 189, 191

**W-band** A region of the electromagnetic spectrum covering 75–111 GHz. 6, 38, 42, 63, 121

**X-band** A region of the electromagnetic spectrum covering 8–12 GHz. 6, 58, 63, 69, 177





