ING LA PALMA TECHNICAL NOTE NO. 136

Arc maps for the He, Ne, Hg, ThAr, and Cd lamps when used with WYFFOS/Red+4

Version 1.0

Ronny Errmann (University Jena and ING)
Lilian Domínguez Palmero (ING)

December 2015

Contents

Introduction

The following document is a compilation of the arc lamp maps for AF2+WYFFOS, the multi-object, wide-field, fiber spectrograph at the Prime focus of the 4.2m William Herschel Telescope (WHT).

The document contains a table with the information of the representative setups. The data is ordered from lowest to highest resolution and from the bluest to the reddest wavelength. After the table the document continues with the arc maps for all setups in the order of the table. Appendix 1 contains the list with all lines labeled in the plots. In Appendix 2 and 3 the source code of the two scripts used for creating the plots: First the *idl* script to create the plots and second the *python* script, which processes the output of the WYFFOS reduction pipeline[1] for the *idl* script. Additionally, the latter one creates the tables shown in this document.

The data was obtained in May, October, and November 2015 with the new CCD (Red+4) after the upgrades in the spectrograph, intended to enhance the throughput at blue spectra ranges. For more information about the Red+4 CCD check the AF2 webpage[2]. For all setups a CCD binning of 2x2 was used. The reduction and wavelength calibration was done with the instrument pipeline version 3.

The entries in Table 1 show the information of all used setups. The first three columns give the grating (1), the central wavelength (2) to which WYFFOS was set, and the wavelength range (3). Column (4) gives the dispersion of the grating for the used wavelength. The dispersion values were either copied directly from the AF2+WYFFOS webpage[3] or calculated from the values available on the webpage. The following two columns show the used arc lamp (5) and exposure time (6). Column (7) gives the number of the usable lines. The first number shows all non-blended lines which are visible in the subplots and have enough signal-to-noise to be used for wavelength calibration. The second number shows only these lines which have at least 10% of the flux of the brightest arc-line. If a neutral density filter was used, the value is given in the last column (8).

In the atlas plots the median flux of all fibers in [e^-/px] (pixel-step-in-wavelength) against the wavelength in [Å] is plotted. As each fiber has a different throughput the fibers were scaled and a 2σ clipping was applied before combining them. To determine the scaling factor, the scaling ratio against the median flux for each pixel was determined. These factors were combined by using the weighted average with the median flux as weights. In the plots the flux is normalized to one second exposure time and collapsed to one pixel. λ_{cen} gives the central wavelength as set up in WYFFOS. The plotted wavelength range is the same as in Table 1. Each of the four subplots shows approximately one quarter of the wavelength range of the top plot.

The gratings used in WYFFOS are the same as used in ISIS[4], except for the H1800V grating, which was designed for IDS[5]. The H1800V grating is smaller in size than WYFFOS/ISIS gratings, hence some vignetting in the spectral range occurs.

NOTES:

- The Hg and Cd lamps show strong warming effects. During the first minutes the relative intensity of the lines may change drastically, after ∼2 to ∼5 min it will have stabilized. The spectra in this document were obtained at least 2 to 3 min after the Hg or Cd lamp was switched on but still for some spectra the lamps might not have been stabilized.
- The Hg and Cd lamps contain some Ar gas, hence their arc map contains Ar lines together with the Cd or Hg.

[1] http://www.ing.iac.es/astronomy/instruments/af2/reduction.html
[2] http://www.ing.iac.es/astronomy/instruments/af2/detector.html
[3] http://www.ing.iac.es/astronomy/instruments/af2
[4] http://www.ing.iac.es/astronomy/instruments/isis/
[5] http://www.ing.iac.es/astronomy/instruments/ids/

- The plots without sufficient arc lines are plotted for homogeneity and to assist the decision for optimum arc lamp. See [6] for further notice.

Detailed lamp properties

He lamp from SpectroLamps

`http://www.spectrolamps.com.au/spectral_lamps.html`

The Helium lamp is a stable, high quality source of discrete spectral lines. It consists of a simple borosilicate discharge tube. It uses a high-current electrode and a thin coil of tungsten wire is wound around a pellet of thoria which imparts excellent electron-emissive properties to this electrode, minimising its operating temperature and allowing it to handle a higher current density.

Ne lamp from the Oriel brand, now sold by Newport

`http://search.newport.com/?q=*&x2=sku&q2=6032`

Model 6032

The 6032 spectral calibration lamp produces narrow, intense lines from the excitation of Neon gas. It is a pencil lamp, called like this because of its size and shape. It is made of double bore quartz tubing with two electrodes at one end sealed into a phenolic handle.

Hg(Ar) lamp from the Oriel brand, now sold by Newport

`http://search.newport.com/?q=*&x2=sku&q2=6035`

Model 6035

The 6035 spectral calibration lamp produces narrow, intense lines from the excitation of Argon gas and Mercury vapor. The 6035 Hg(Ar) Lamp is insensitive to temperature. It requires a two-minute warm-up for the mercury vapor to dominate the discharge, then 30 minutes for complete stabilization. The average intensity is remarkably constant and reproducible after the thermal conditions stabilize. It is a pencil lamp, called like this because of its size and shape. It is made of double bore quartz tubing with two electrodes at one end sealed into a phenolic handle.

ThAr lamp from Heraeus

`http://www.heraeus-noblelight.com/en/productsandsolutions/opticalanalysis/hollow_cathode_lamps.aspx`

The Thorium-Argon lamp is a hollow cathode lamp. It consists of a cathode made from the element of interest, Thorium in this case, an anode and an inert filler gas contained in a glass envelope, e.g. Argon. In addition various mica discs, ceramic sheaths and glass shields assist in alignment and insulation.

Cd lamp from UVP

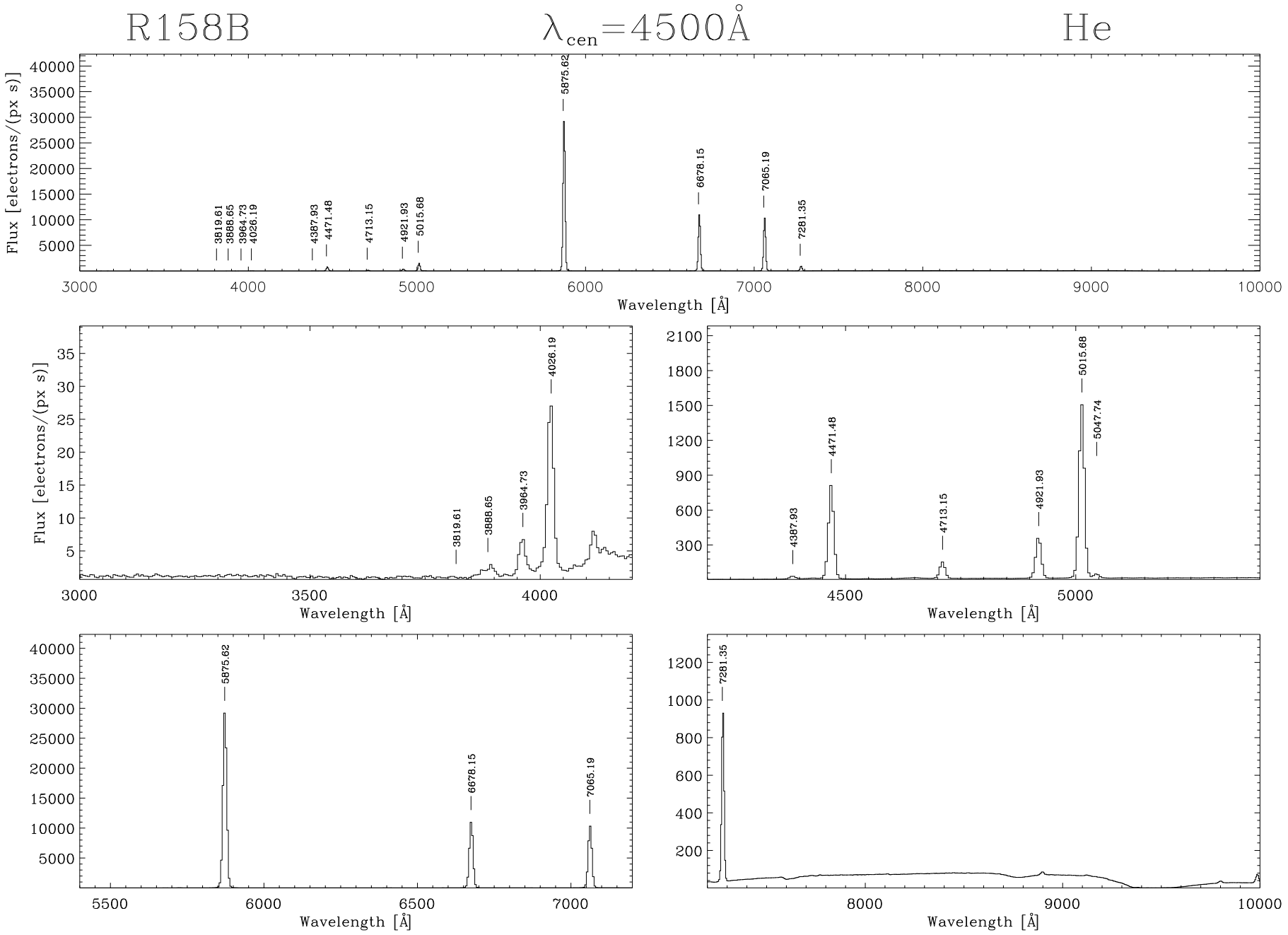
`http://uvp.com/zinccadmium.html`

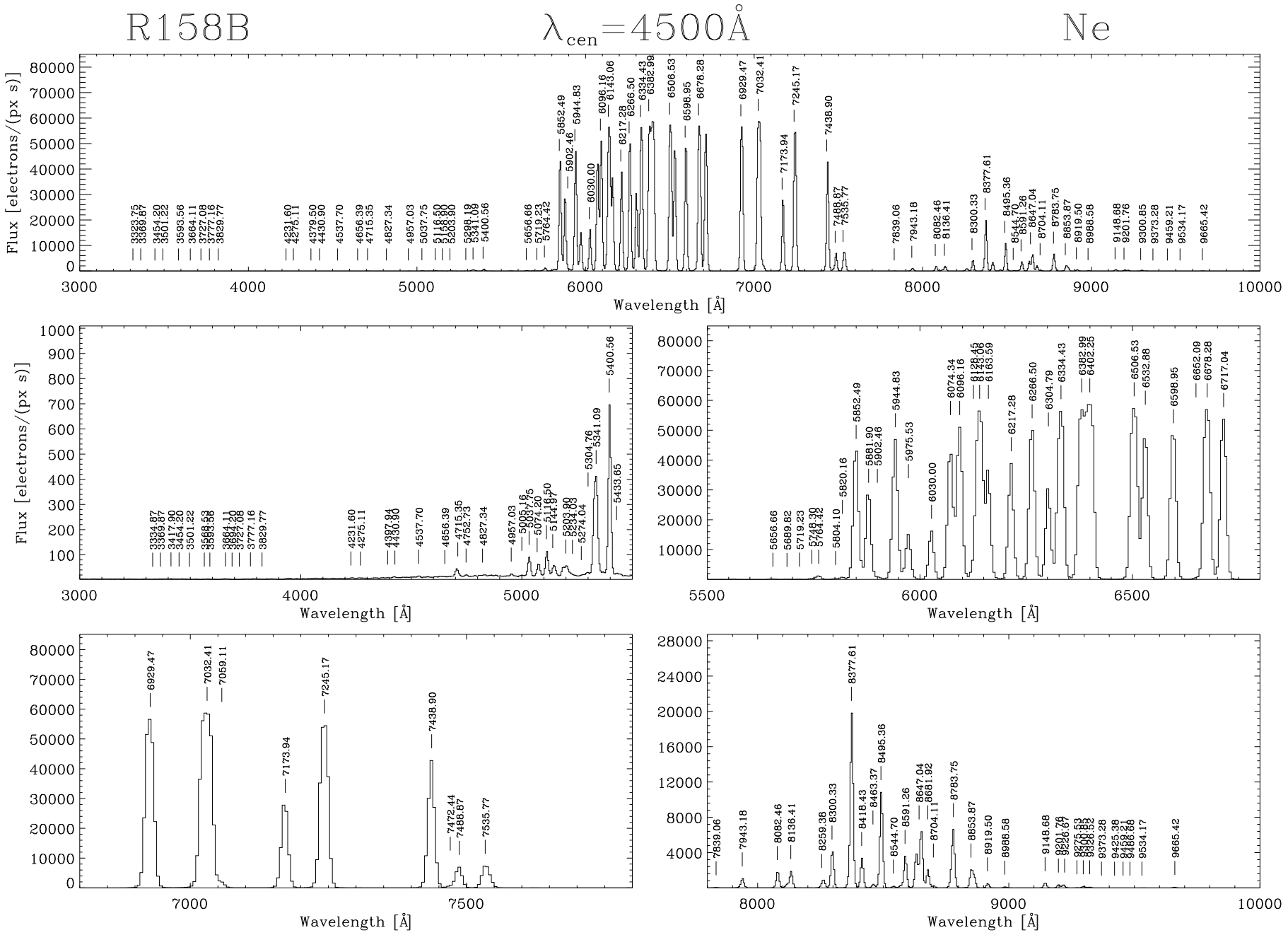
The Cadmium is a Pen-Ray Lamp, which are low pressure, cold cathode UV lamps made of double bore quartz tubing. This design allows the electrodes to be positioned at one end of the lamp providing easy access into small apertures.

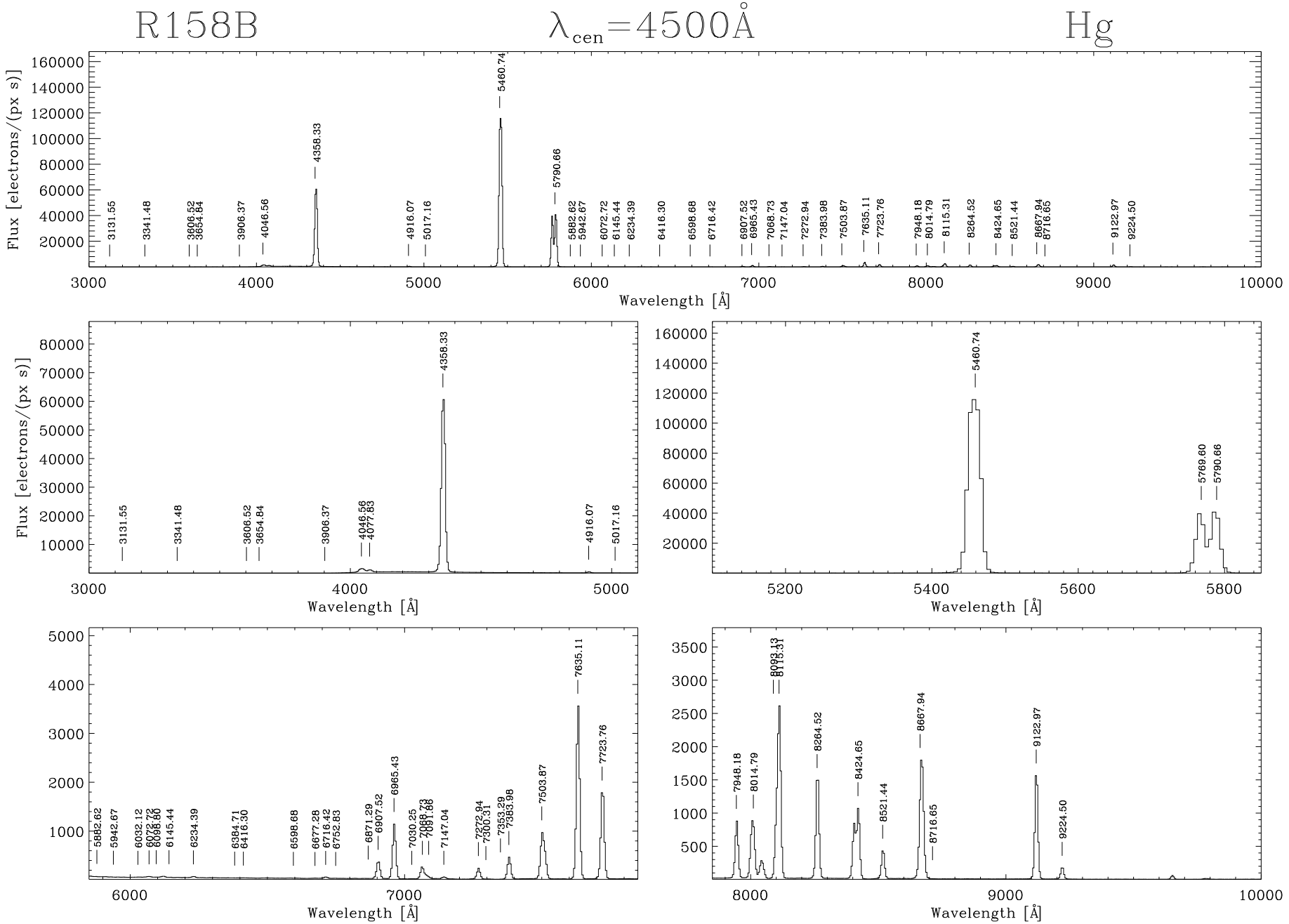
[6] http://www.ing.iac.es/astronomy/instruments/af2/calibration.html

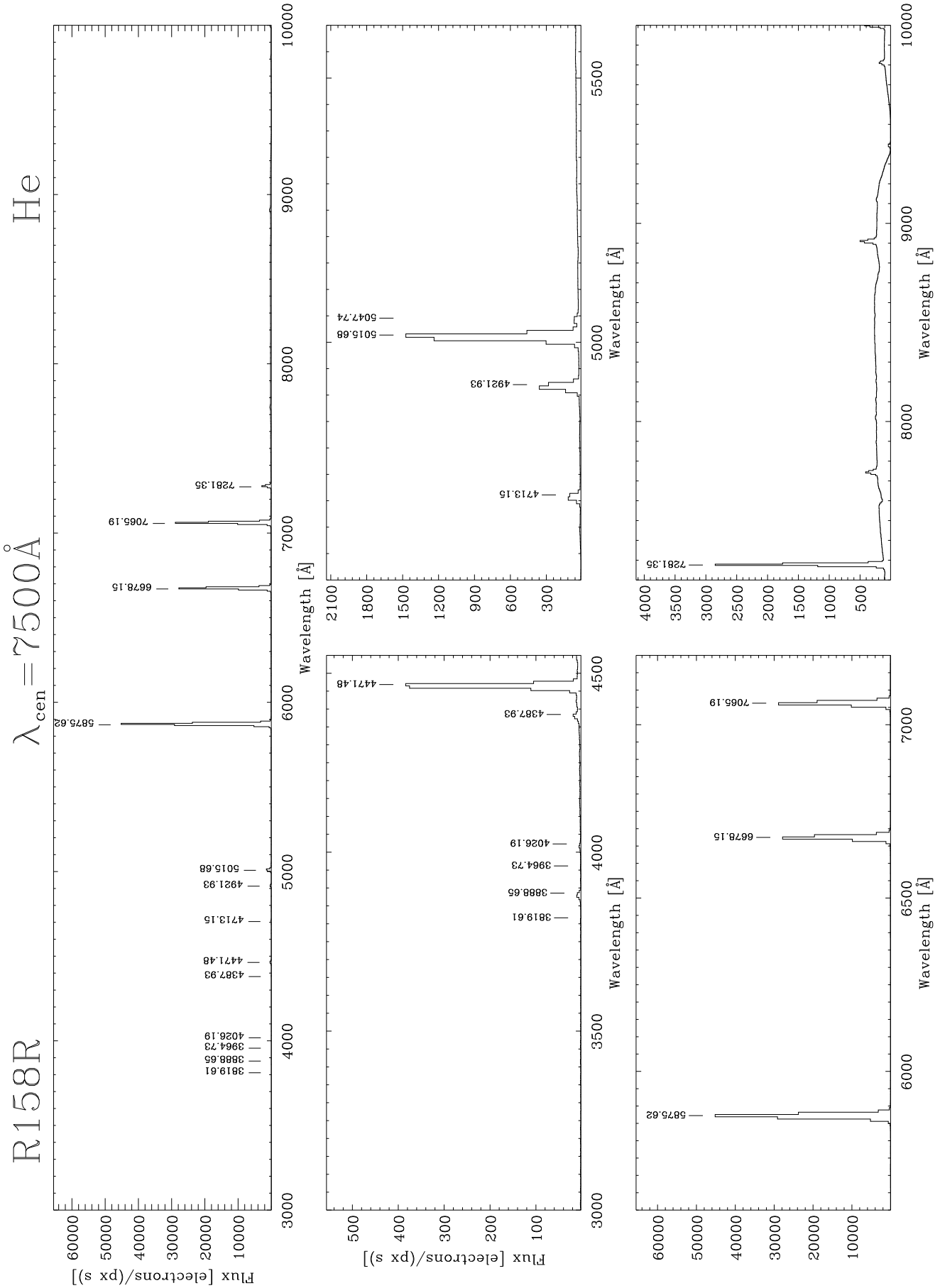
Grating	λ_{cen} **Å**	$\lambda_{min} - \lambda_{max}$ **Å**	**Dispersion Å/px**	**Lamp**	**Exp.time s**	**Lines**	**ND filter**
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
R158B	4500	3000 – 10000	6.4	He	3	9 /8	
				Ne	2	31 /25	
				Hg	1	18 /15	
R158R	7500	3000 – 10000	6.4	He	2	8 /7	
				Ne	1	30 /23	ND.5
				Hg	1	23 /17	ND.5
R300B	5300	3000 – 8730	3.6	He	2	9 /8	
				Ne	2	46 /34	
				Hg	1	21 /14	
R316R	7500	4418 – 10000	3.4	He	2	5 /4	
				Ne	1	47 /32	ND.5
				Hg	1	17 /12	ND.5
R600B	5000	3351 – 6682	1.8	He	5	9 /6	
				Ne	2	25 /21	
				Hg	1	7 /6	ND.5
				ThAr	60	72 /60	
				Cd	10	4 /4	
R600R	6500	4834 – 8177	1.8	He	3	6 /6	
				Ne	1	32 /27	ND.5
				Hg	1	12 /12	ND.5
				ThAr	5	39 /29	
				Cd	1	12 /11	
	8000	6371 – 9668	1.8	He	3	3 /3	
				Ne	1	36 /23	ND.5
				Hg	3	19 /13	
				ThAr	2	30 /17	
				Cd	20	21 /14	
R1200B	4000	3177 – 4815	0.86	He	120	7 /5	
				Ne	120	2 /2	
				Hg	2	5 /3	
				ThAr	120	47 /43	
				Cd	30	15 /13	
	4500	3678 – 5298	0.84	He	60	9 /6	
				Ne	120	11 /10	
				Hg	1	4 /3	
				ThAr	60	66 /48	
				Cd	30	3 /3	
	5000	4193 – 5804	0.81	He	30	6 /6	
				Ne	60	28 /23	
				Hg	1	4 /4	
				ThAr	120	67 /54	
				Cd	30	20 /18	
R1200R	5500	4665 – 6292	0.79	He	15	5 /4	
				Ne	2	21 /18	
				Hg	3	4 /4	
				ThAr	120	64 /47	
				Cd	10	3 /3	
	6000	5169 – 6746	0.76	He	15	2 /2	ND.5
				Ne	2	25 /23	ND.5
				Hg	3	5 /5	

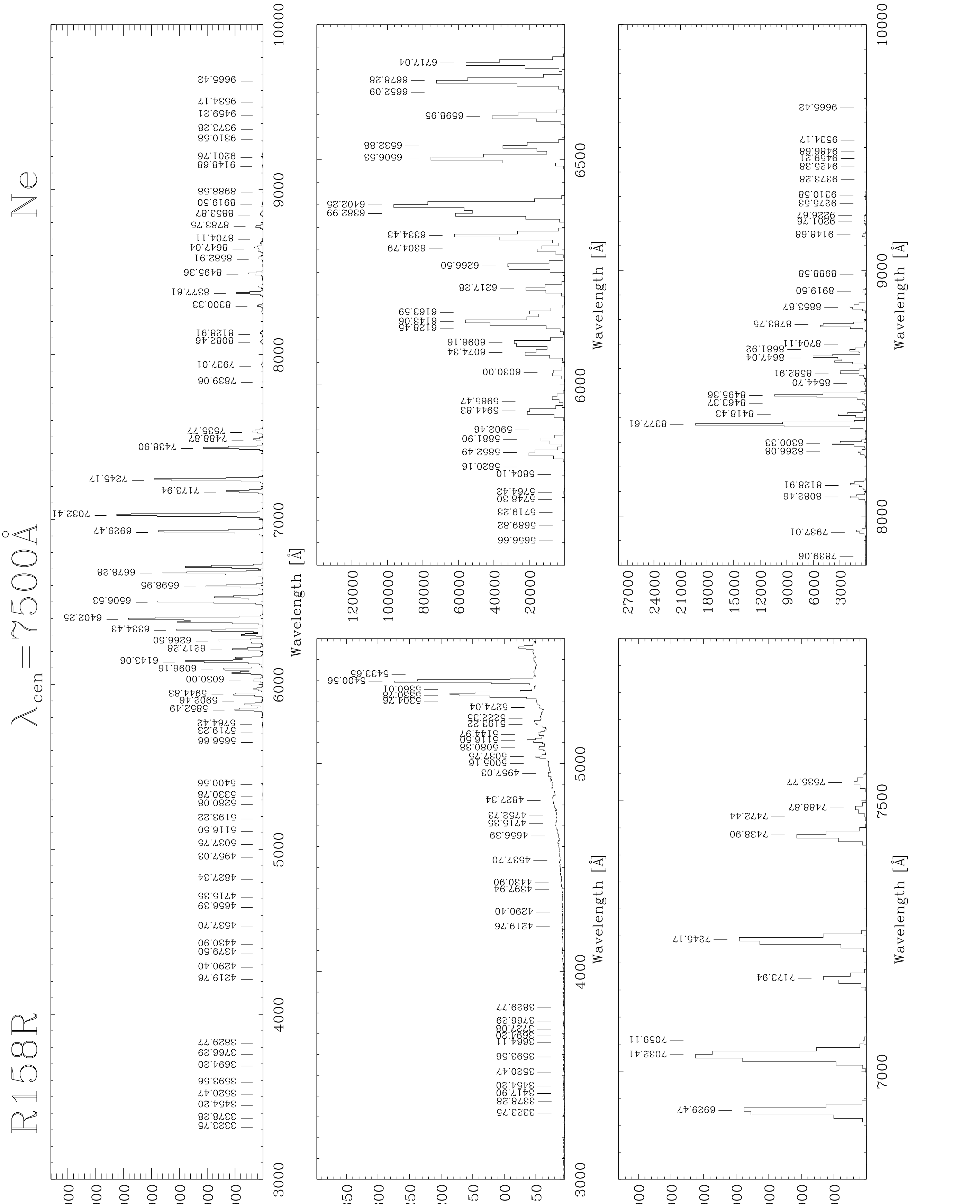
Grating	λ_{cen} **Å**	$\lambda_{\text{min}} - \lambda_{\text{max}}$ **Å**	**Dispersion Å/px**	**Lamp**	**Exp.time s**	**Lines**	**ND filter**
				ThAr	10	51 /43	
				Cd	30	26 /23	
	6500	5681 − 7266	0.74	He	5	3 /3	
				Ne	1	28 /23	
				Hg	10	13 /10	
				ThAr	30	57 /39	
				Cd	30	4 /3	
	7000	6195 − 7748	0.71	He	10	3 /3	
				Ne	1	21 /19	
				Hg	10	18 /10	
				ThAr	5	29 /19	
				Cd	10	11 /10	
	7500	6711 − 8229	0.69	He	5	2 /2	
				Ne	1	15 /13	ND.5
				Hg	10	18 /14	
				ThAr	5	24 /13	
				Cd	10	17 /14	
	8000	7208 − 8708	0.66	He	40	1 /1	
				Ne	1	24 /16	
				Hg	5	17 /16	
				ThAr	3	18 /15	
				Cd	5	16 /15	
	8500	7724 − 9184	0.64	He	80	0 /0	
				Ne	1	31 /14	
				Hg	1	11 /10	
				ThAr	5	11 /11	
				Cd	5	12 /11	
	9350	8601 − 10000	0.59	He	120	0 /0	
				Ne	5	20 /14	
				Hg	2	3 /3	
				ThAr	10	8 /5	
				Cd	30	7 /6	
H1800V	5120	4604 − 5579	0.44	He	60	4 /4	
				Ne	60	30 /23	
				ThAr	30	56 /52	
H2400B	4000	3645 − 4343	0.38	He	120	3 /2	
				Ne	120	0 /0	
				Hg	4	5 /5	
				ThAr	300	35 /35	
				Cd	300	9 /9	
	4500	4144 − 4833	0.38	He	60	2 /2	
				Ne	120	6 /5	
				Hg	1	0 /0	
				ThAr	120	32 /28	
				Cd	60	14 /13	

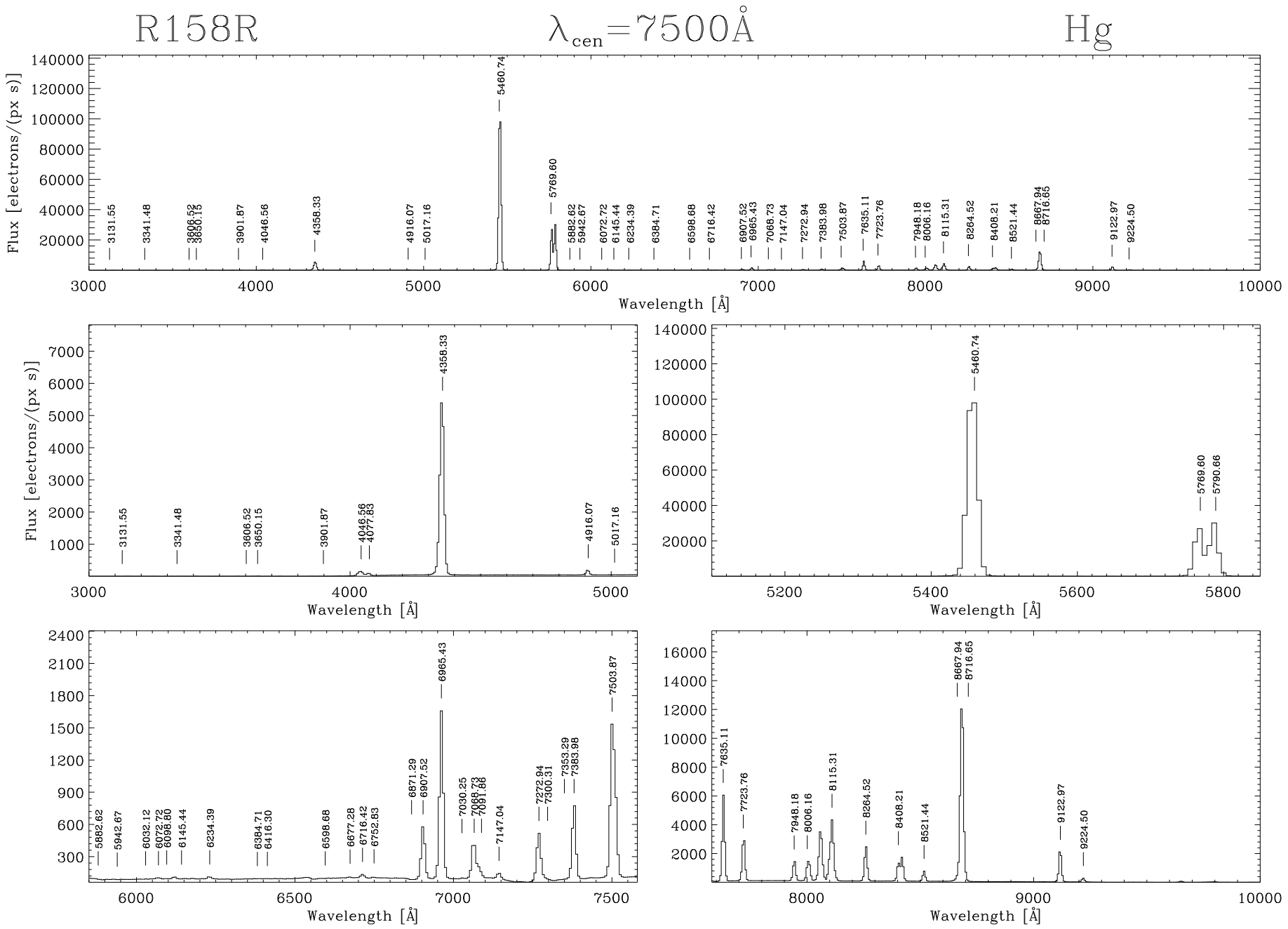

R158B
λcen=4500Å
He
Flux [electrons/(px s)]
Wavelength [Å]
3819.61
3888.65
3964.73
4026.19
4387.93
4471.48
4713.15
4921.93
5015.68
5875.62
6678.15
7065.19
7281.35
5047.74

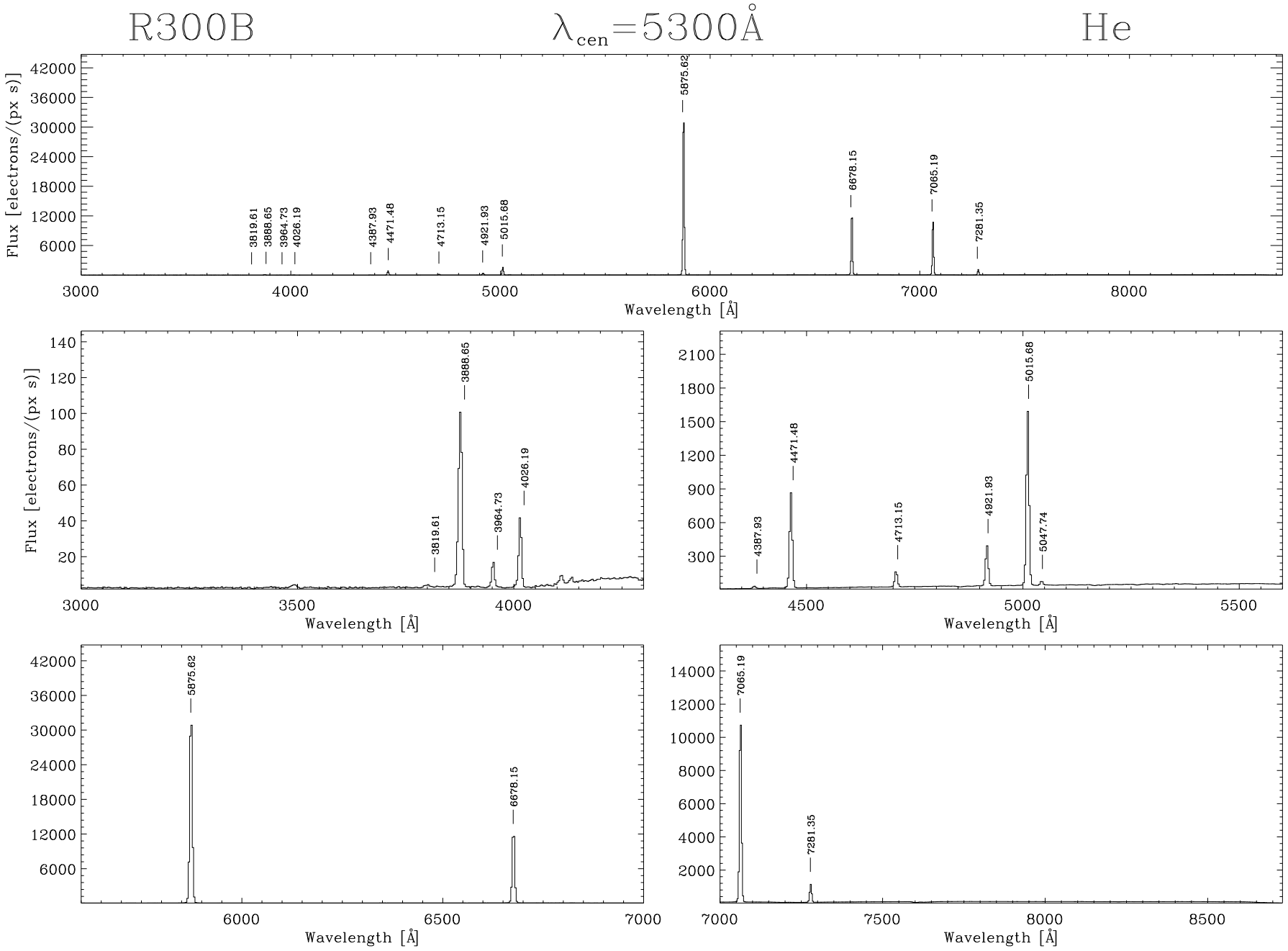

R158B
λcen=4500Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
3323.75
3369.87
3454.20
3501.22
3593.56
3664.11
3727.09
3777.16
3829.77
4231.60
4275.11
4379.50
4430.90
4537.70
4656.39
4715.35
4827.34
4957.03
5037.75
5118.50
5158.90
5203.90
5298.19
5341.09
5400.56
5656.66
5719.23
5764.42
5852.49
5902.46
5944.83
6030.00
6096.16
6143.06
6217.28
6266.50
6334.43
6382.99
6506.53
6598.95
6678.28
6929.47
7032.41
7173.94
7245.17
7438.90
7472.44
7488.87
7535.77
7839.06
7943.18
8082.46
8136.41
8300.33
8377.61
8495.36
8544.70
8591.26
8647.04
8704.11
8783.75
8853.87
8919.50
8988.58
9148.68
9201.76
9300.85
9373.28
9459.21
9534.17
9665.42
5433.65
5304.76
5274.04
5233.03
5203.90
5174.97
5144.97
5118.50
5078.20
5008.75
4957.03
4752.73
4397.94
4430.90
5804.10
5820.16
5881.90
5902.46
5975.53
6030.00
6074.34
6096.16
6128.45
6143.06
6163.59
6217.28
6266.50
6304.79
6334.43
6382.99
6402.25
6506.53
6532.88
6598.95
6652.09
6678.28
6717.04
7059.11
7173.94
7245.17
7438.90
7472.44
7488.87
7535.77
8259.38
8300.33
8377.61
8418.43
8463.37
8495.36
8591.26
8647.04
8681.92
8704.11
8783.75
8853.87
8919.50
8988.58
9148.68
9373.28
9534.17
9665.42

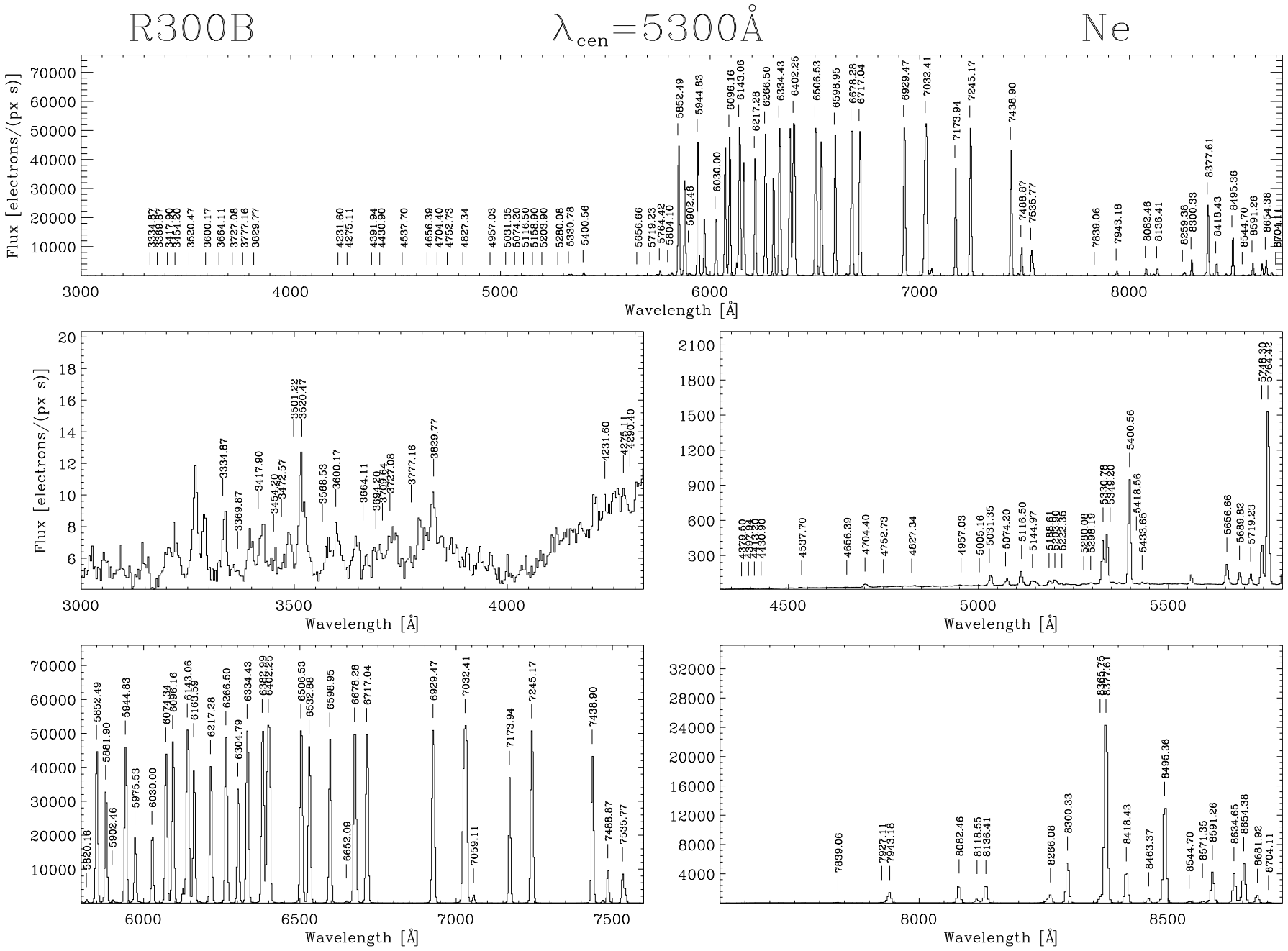

R158B
λcen=4500Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
3131.55
3341.48
3606.52
3654.84
3906.37
4046.56
4077.83
4358.33
4916.07
5017.16
5460.74
5769.60
5790.66
5882.62
5942.67
6032.12
6072.72
6098.60
6145.44
6234.39
6384.71
6416.30
6598.68
6677.28
6716.42
6752.83
6871.29
6907.52
6965.43
7030.25
7088.73
7091.88
7147.04
7272.94
7300.31
7353.29
7383.98
7503.87
7635.11
7723.76
7948.18
8014.79
8093.13
8115.31
8264.52
8424.65
8521.44
8667.94
8716.65
9122.97
9224.50

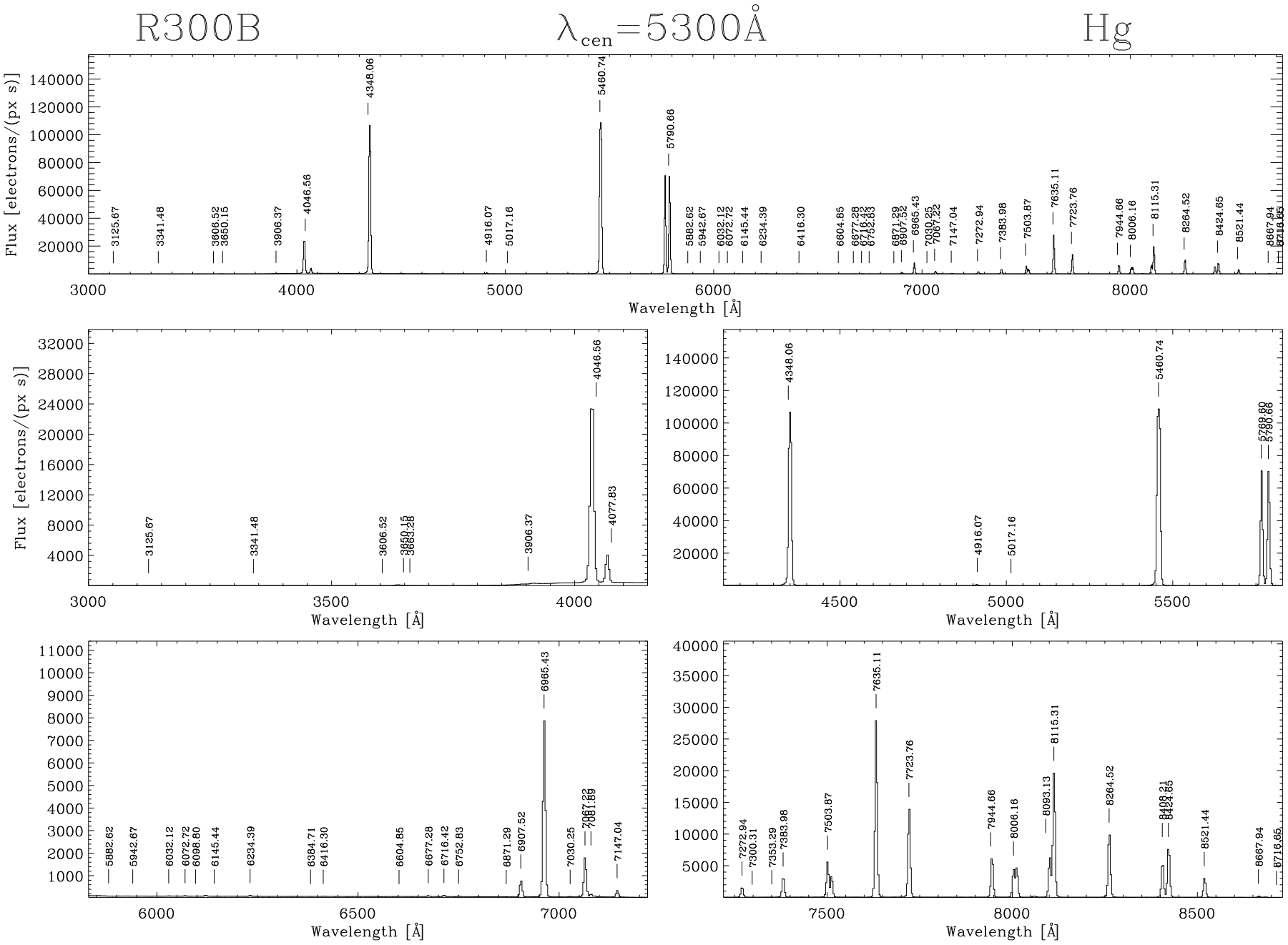

R158R
λ_{cen}=7500Å
He
Flux [electrons/(px s)]
Wavelength [Å]
5875.62
6678.15
7065.19
7281.35
5015.68
4921.93
4713.15
4471.48
4387.93
4026.19
3964.73
3888.65
3819.61
5047.74

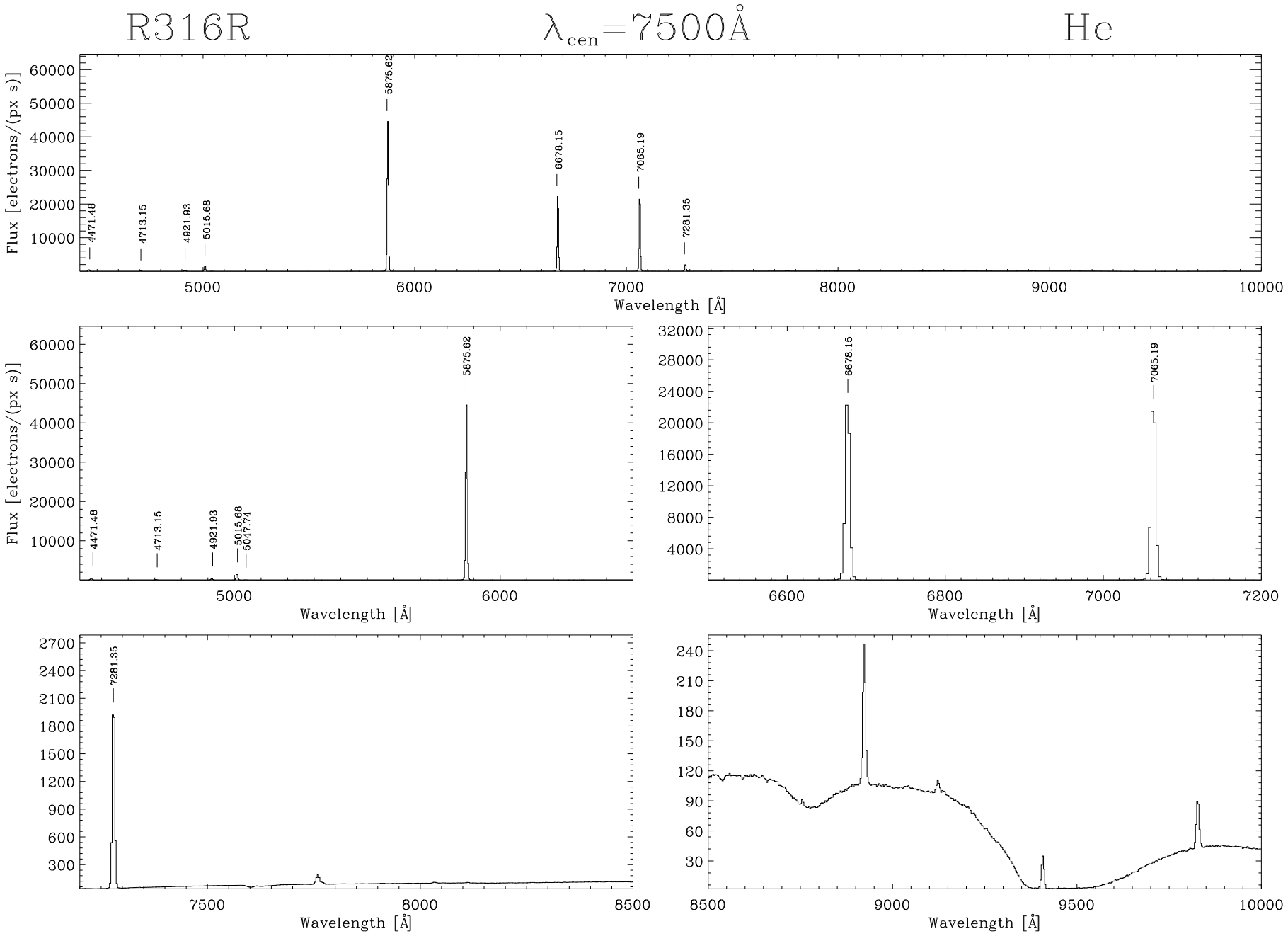

R158R
λcen=7500Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]

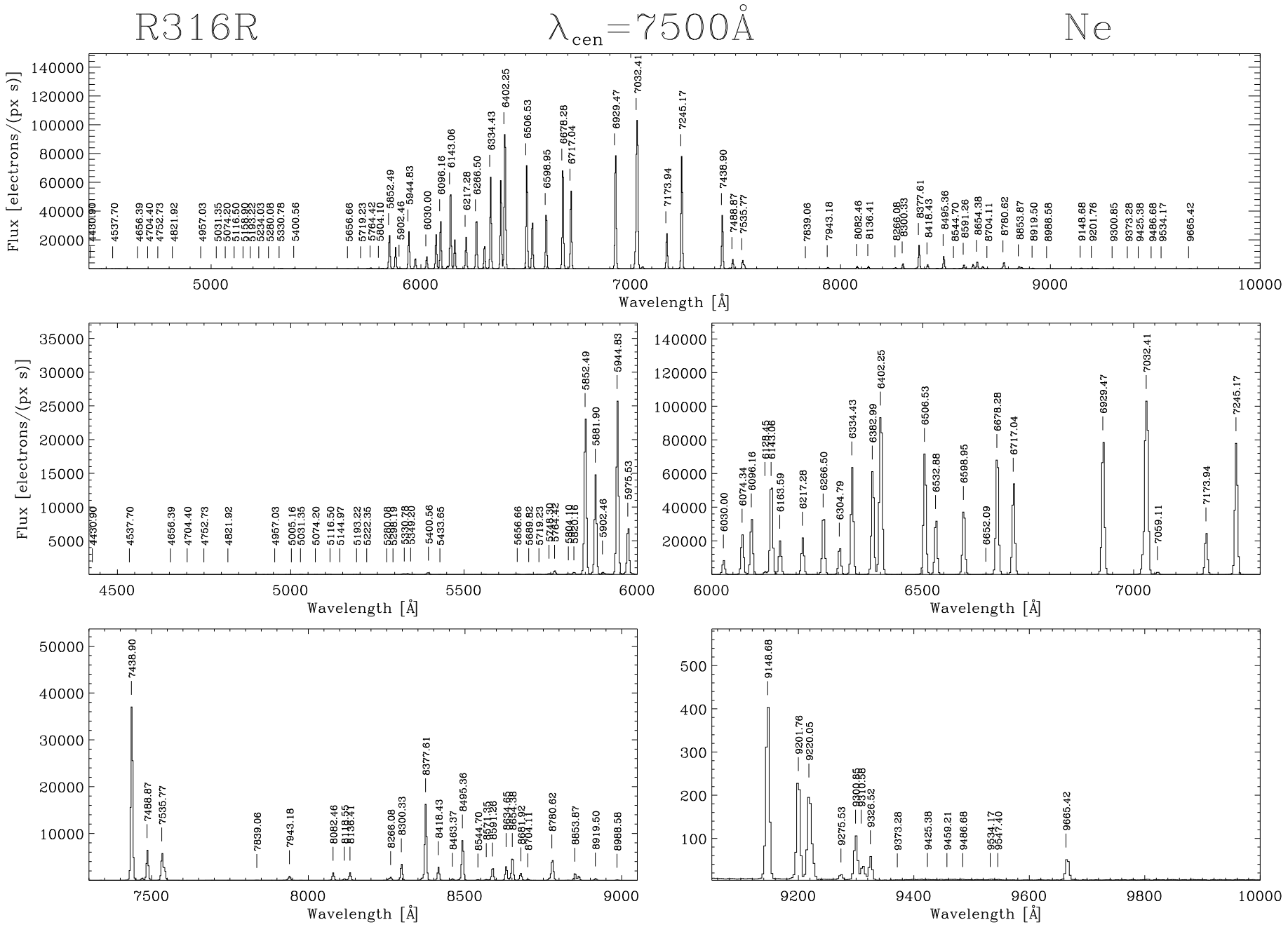

R158R
λcen=7500Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
3131.55
3341.48
3606.52
3650.15
3901.87
4046.56
4358.33
4916.07
5017.16
5460.74
5769.60
5790.66
5882.62
5942.67
6072.72
6145.44
6234.39
6384.71
6598.68
6716.42
6907.52
6965.43
7068.73
7147.04
7272.94
7383.98
7503.87
7635.11
7723.76
7948.18
8006.16
8115.31
8264.52
8408.21
8521.44
8667.94
8716.65
9122.97
9224.50

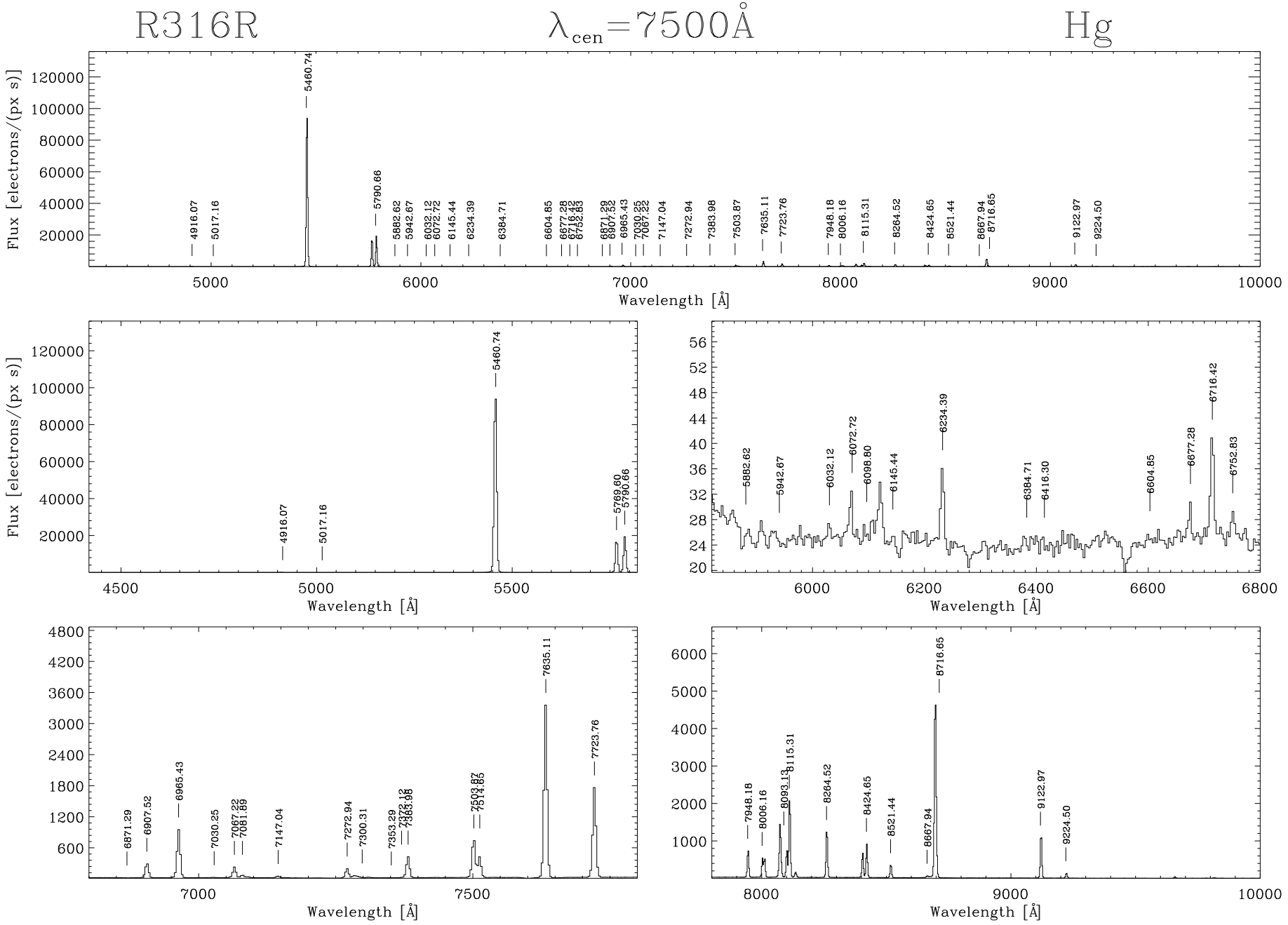

R300B
λcen=5300Å
He
Flux [electrons/(px s)]
Wavelength [Å]
3819.61
3888.65
3964.73
4026.19
4387.93
4471.48
4713.15
4921.93
5015.68
5875.62
6678.15
7065.19
7281.35
5047.74

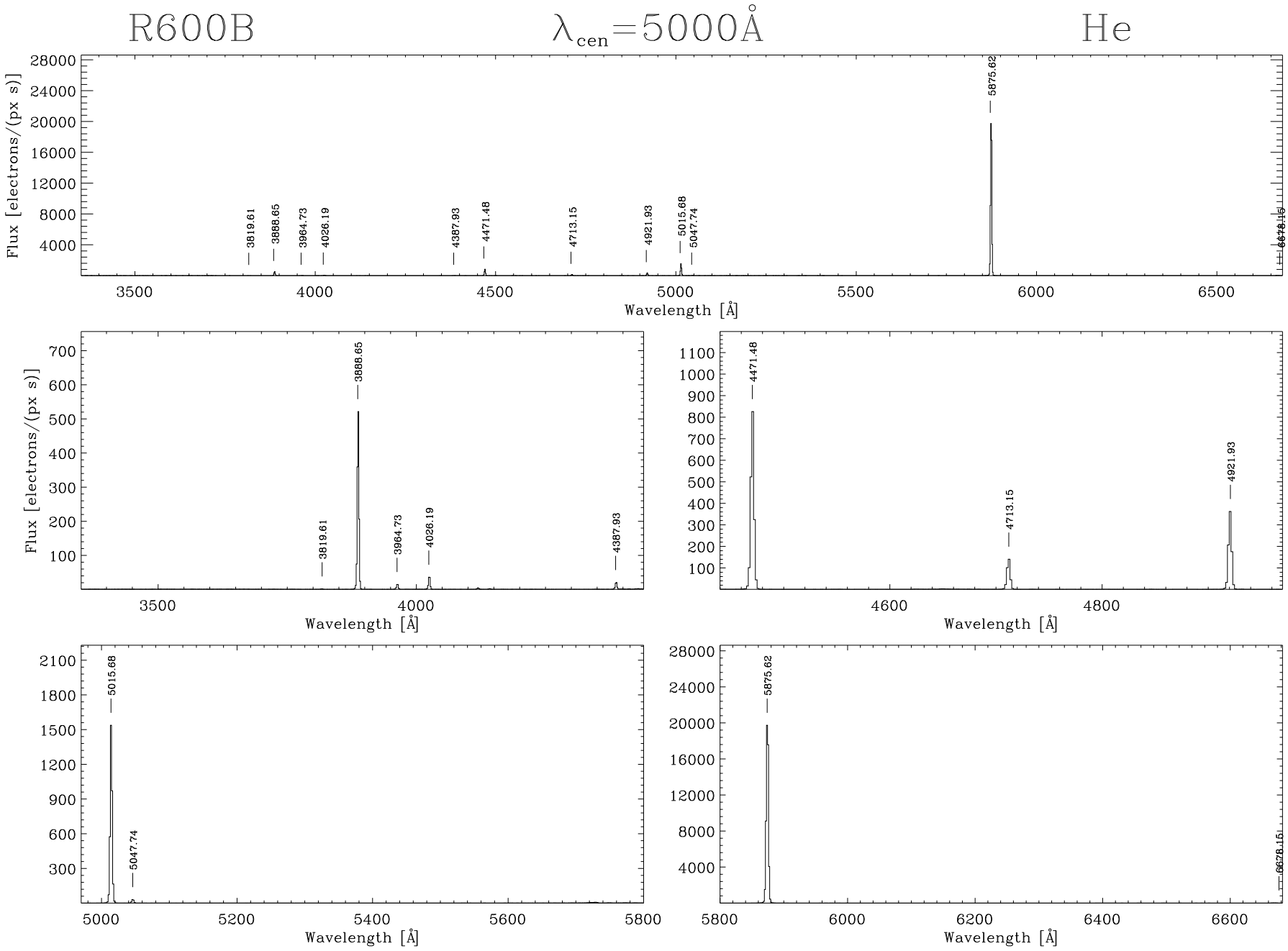

R300B
λcen=5300Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
3334.87
3369.87
3417.90
3454.20
3520.47
3600.17
3664.11
3727.08
3777.16
3829.77
4231.60
4275.11
4391.94
4430.90
4537.70
4656.39
4704.40
4752.73
4827.34
4957.03
5031.35
5074.20
5116.50
5158.90
5203.90
5280.08
5330.78
5400.56
5656.66
5719.23
5764.42
5804.10
5852.49
5902.46
5944.83
6030.00
6096.16
6143.06
6217.28
6266.50
6334.43
6402.25
6506.53
6598.95
6678.28
6717.04
6929.47
7032.41
7173.94
7245.17
7438.90
7488.87
7535.77
7839.06
7943.18
8082.46
8136.41
8259.38
8300.33
8377.61
8418.43
8495.36
8544.70
8591.26
8654.38
8704.11

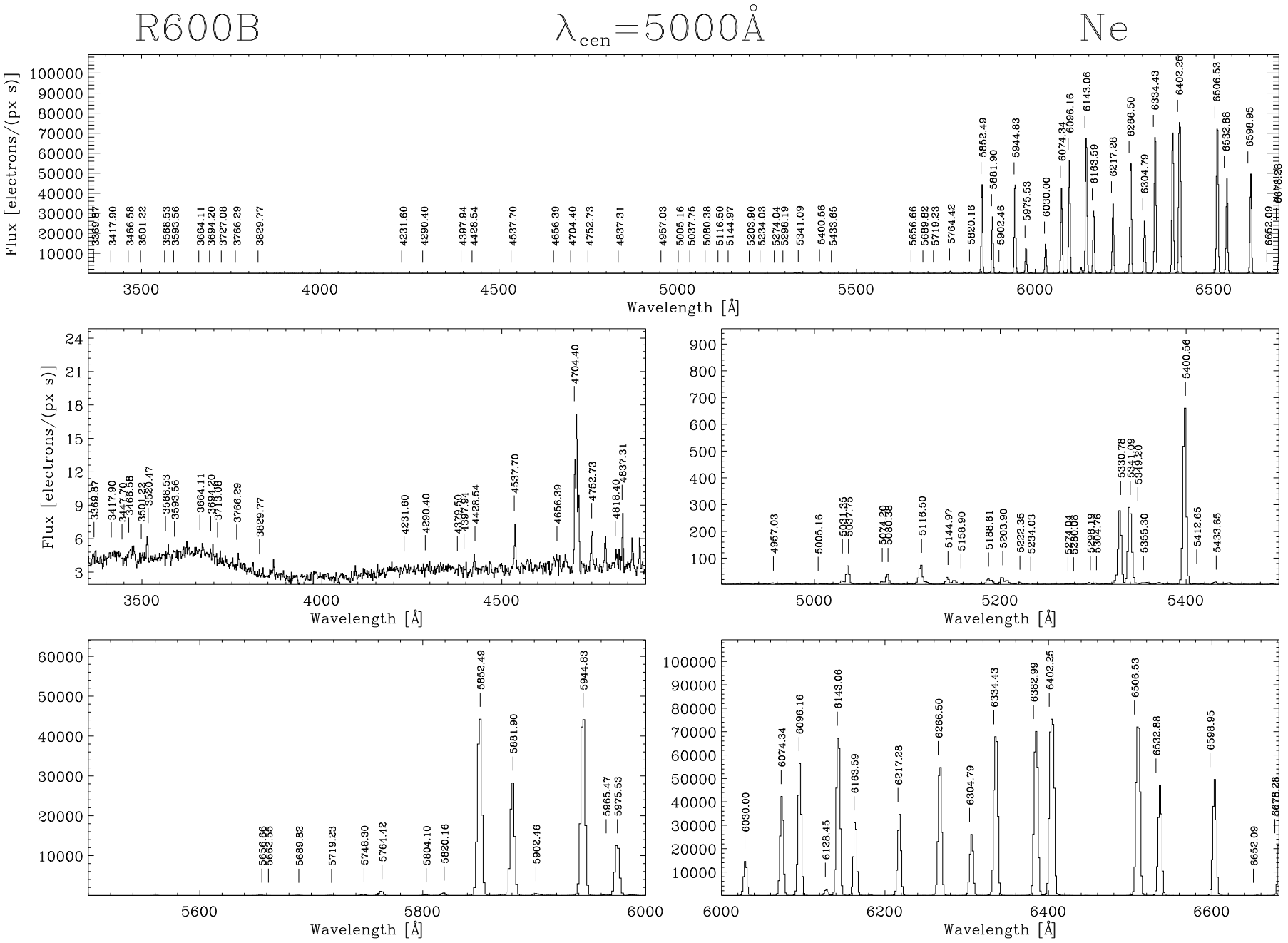

R300B
λcen=5300Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
3125.67
3341.48
3606.52
3650.15
3906.37
4046.56
4348.06
4916.07
5017.16
5460.74
5790.66
5882.62
5942.67
6032.12
6072.72
6145.44
6234.39
6416.30
6604.85
6677.28
6716.42
6752.83
6871.29
6907.52
6965.43
7030.25
7087.22
7147.04
7272.94
7383.98
7503.87
7635.11
7723.76
7944.66
8006.16
8115.31
8264.52
8424.65
8521.44
8667.94
4077.83
5769.60
5790.66
7300.31
7353.29
8093.13
8408.21
8424.65

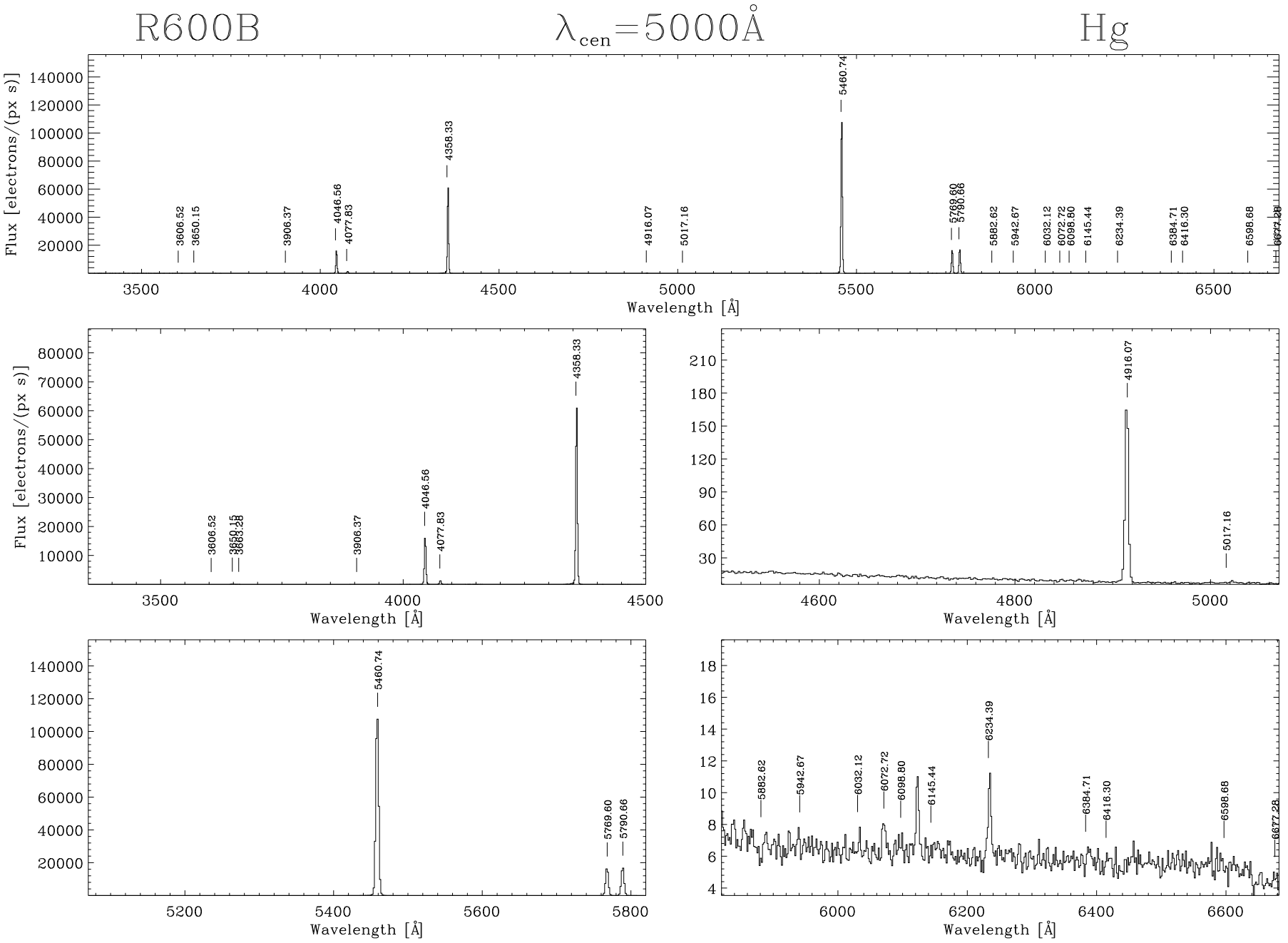

R316R
λcen=7500Å
He
Flux [electrons/(px s)]
Wavelength [Å]
4471.48
4713.15
4921.93
5015.68
5875.62
6678.15
7065.19
7281.35
5047.74

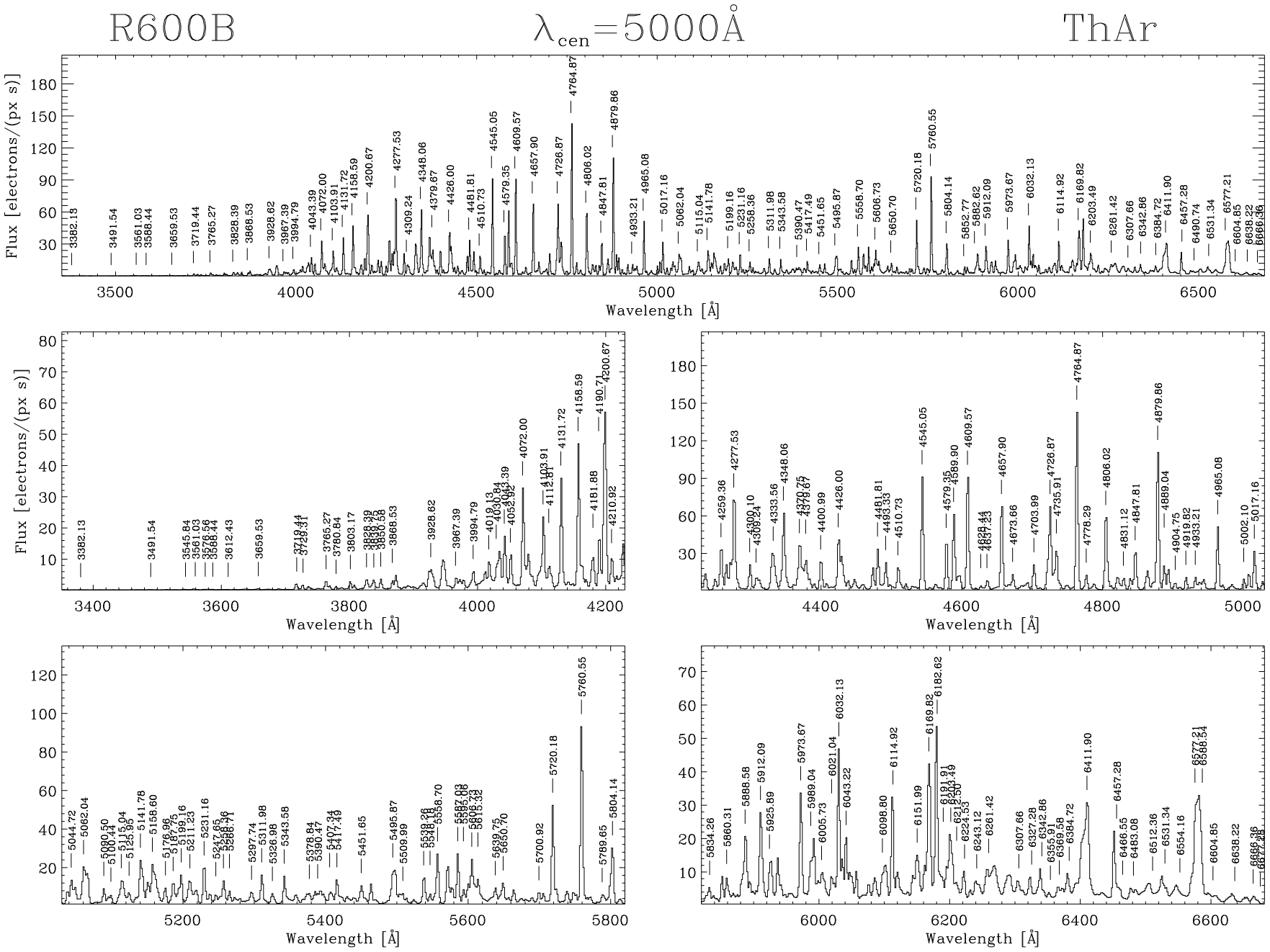

R316R
λcen=7500Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]

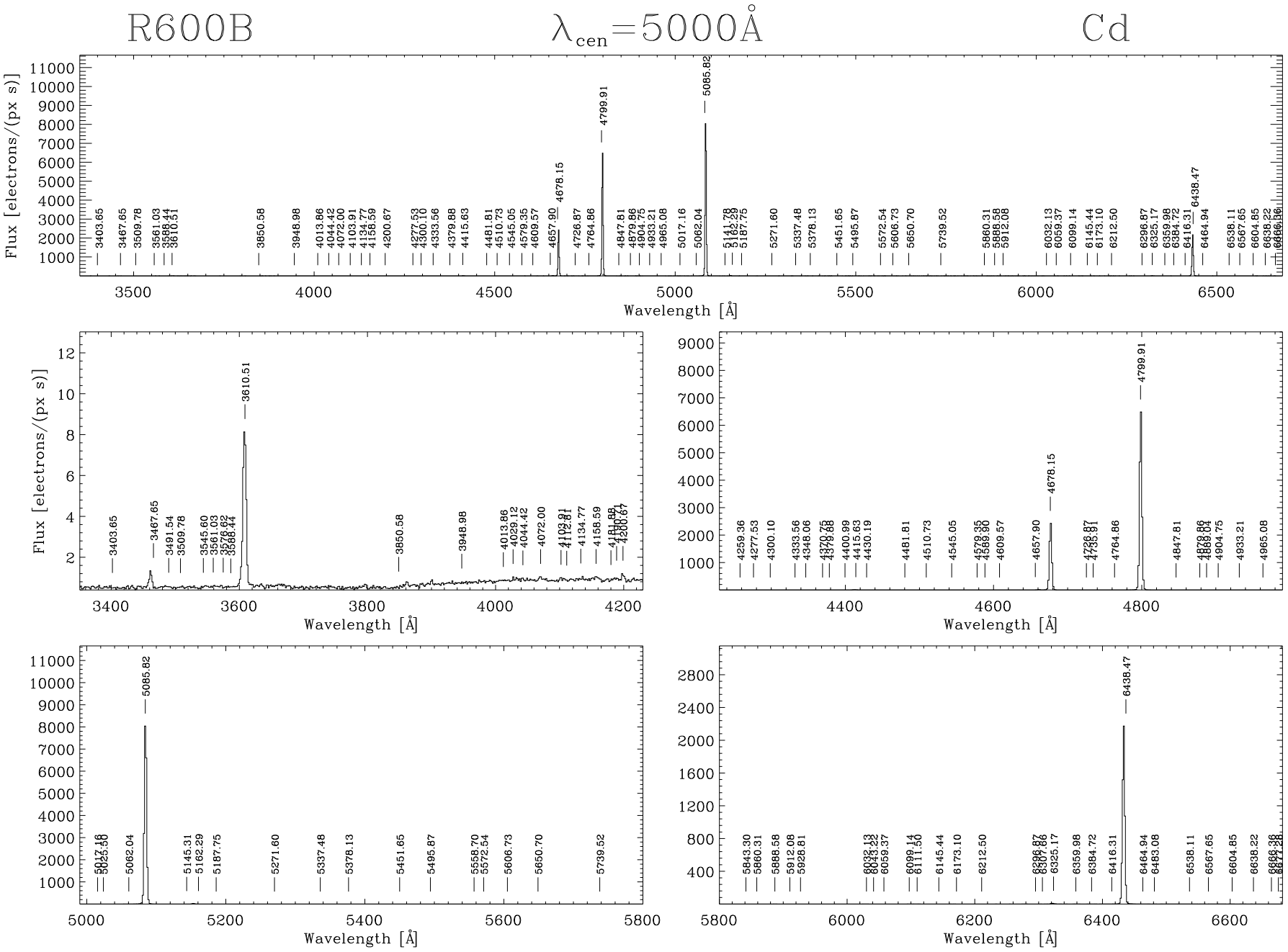

R316R
λcen=7500Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
4916.07
5017.16
5460.74
5769.60
5790.66
5882.62
5942.67
6032.12
6072.72
6098.80
6145.44
6234.39
6384.71
6416.30
6604.85
6677.28
6716.42
6752.83
6871.29
6907.52
6965.43
7030.25
7067.22
7081.89
7147.04
7272.94
7300.31
7353.29
7383.98
7503.87
7514.85
7635.11
7723.76
7948.18
8006.16
8093.13
8115.31
8264.52
8424.65
8521.44
8667.94
8716.65
9122.97
9224.50

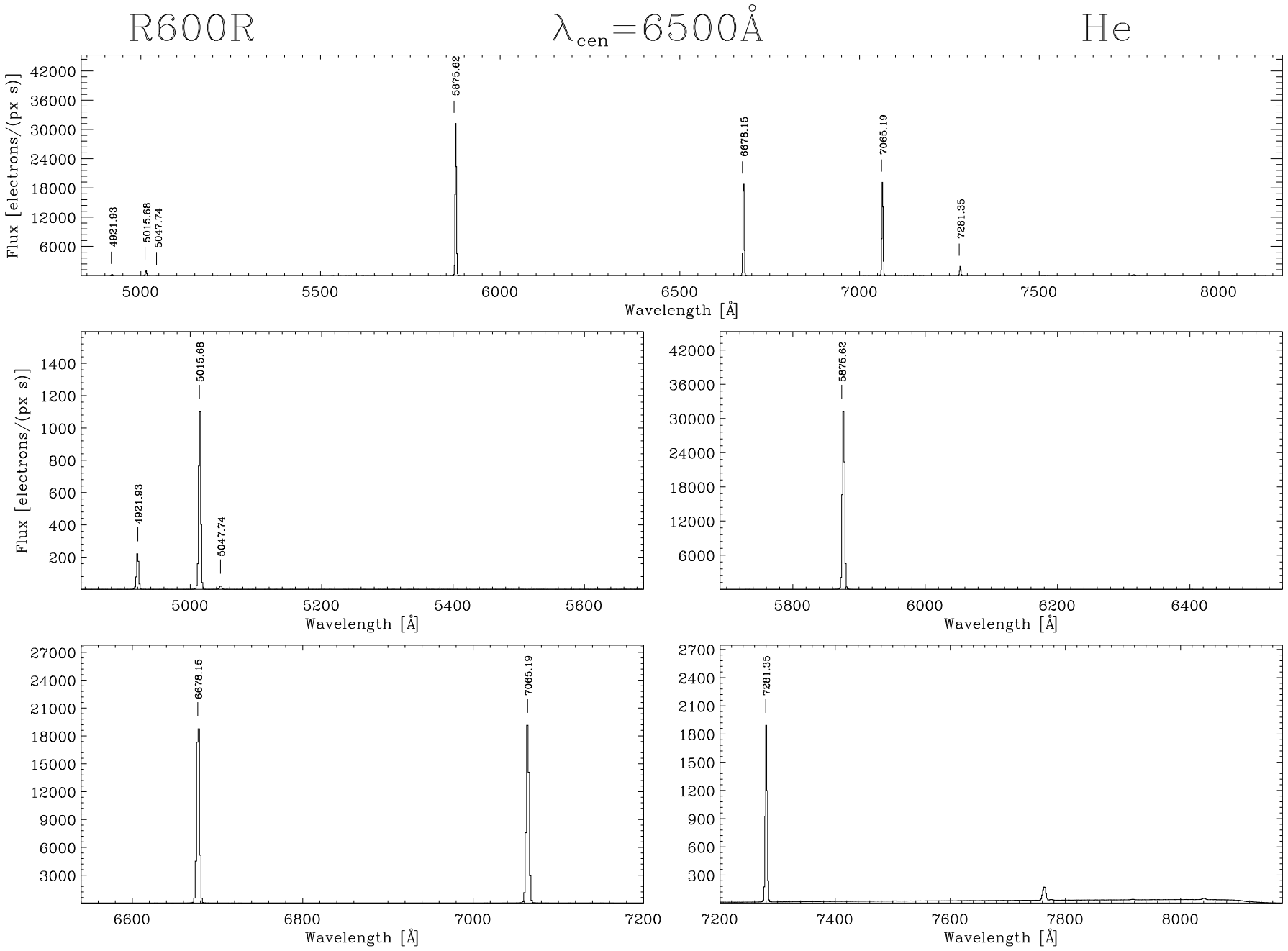

R600B
λcen=5000Å
He
Flux [electrons/(px s)]
Wavelength [Å]
3819.61
3888.65
3964.73
4026.19
4387.93
4471.48
4713.15
4921.93
5015.68
5047.74
5875.62
6678.15

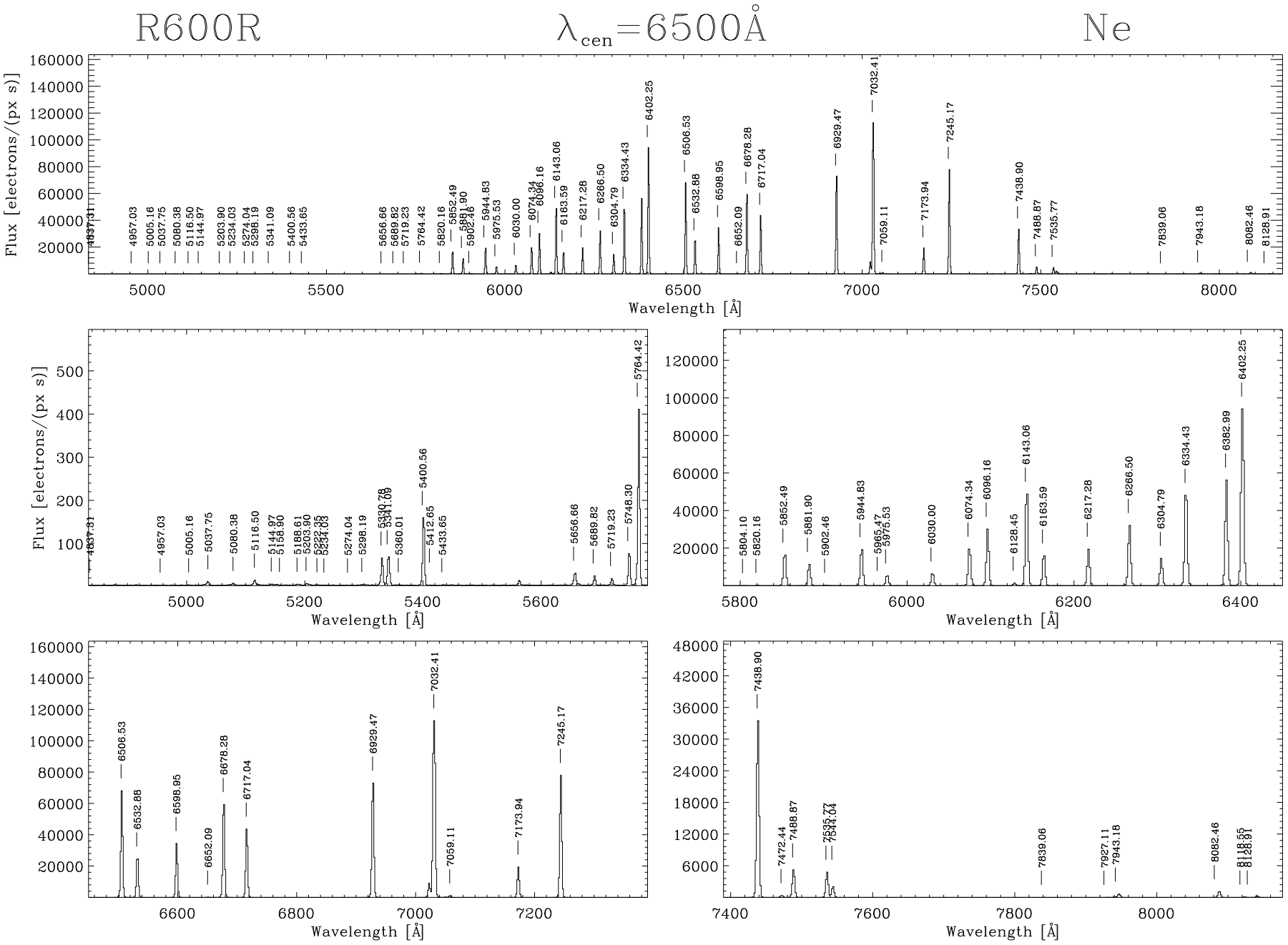

R600B
λcen=5000Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
3369.87
3417.90
3466.58
3501.22
3568.53
3593.56
3664.11
3694.20
3727.08
3766.29
3829.77
4231.60
4290.40
4397.94
4428.54
4537.70
4656.39
4704.40
4752.73
4837.31
4957.03
5005.16
5037.75
5080.38
5116.50
5144.97
5203.90
5234.03
5274.04
5298.19
5341.09
5400.56
5433.65
5656.66
5689.82
5719.23
5764.42
5820.16
5852.49
5881.90
5902.46
5944.83
5975.53
6030.00
6074.34
6096.16
6143.06
6163.59
6217.28
6266.50
6304.79
6334.43
6402.25
6506.53
6532.88
6598.95
6652.09
6678.28
5748.30
5804.10
5965.47
6128.45
6330.78
6341.09
6349.20
5355.30
5412.65
5031.95
5060.38
5174.04
5188.61
5222.35
5158.90
6382.99

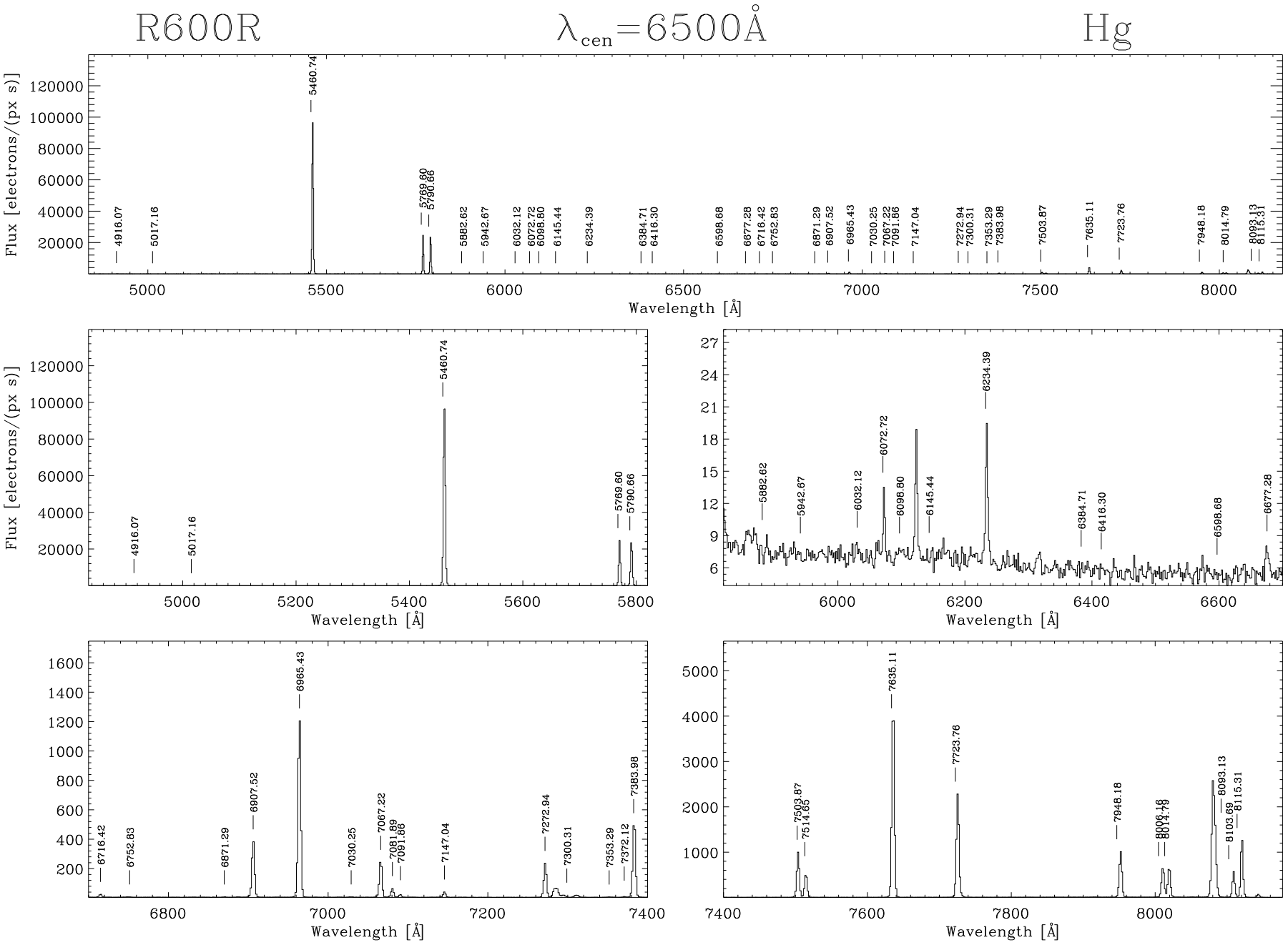

R600B
λcen=5000Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
3606.52
3650.15
3906.37
4046.56
4077.83
4358.33
4916.07
5017.16
5460.74
5769.60
5790.66
5882.62
5942.67
6032.12
6072.72
6098.80
6145.44
6234.39
6384.71
6416.30
6598.68
6677.28

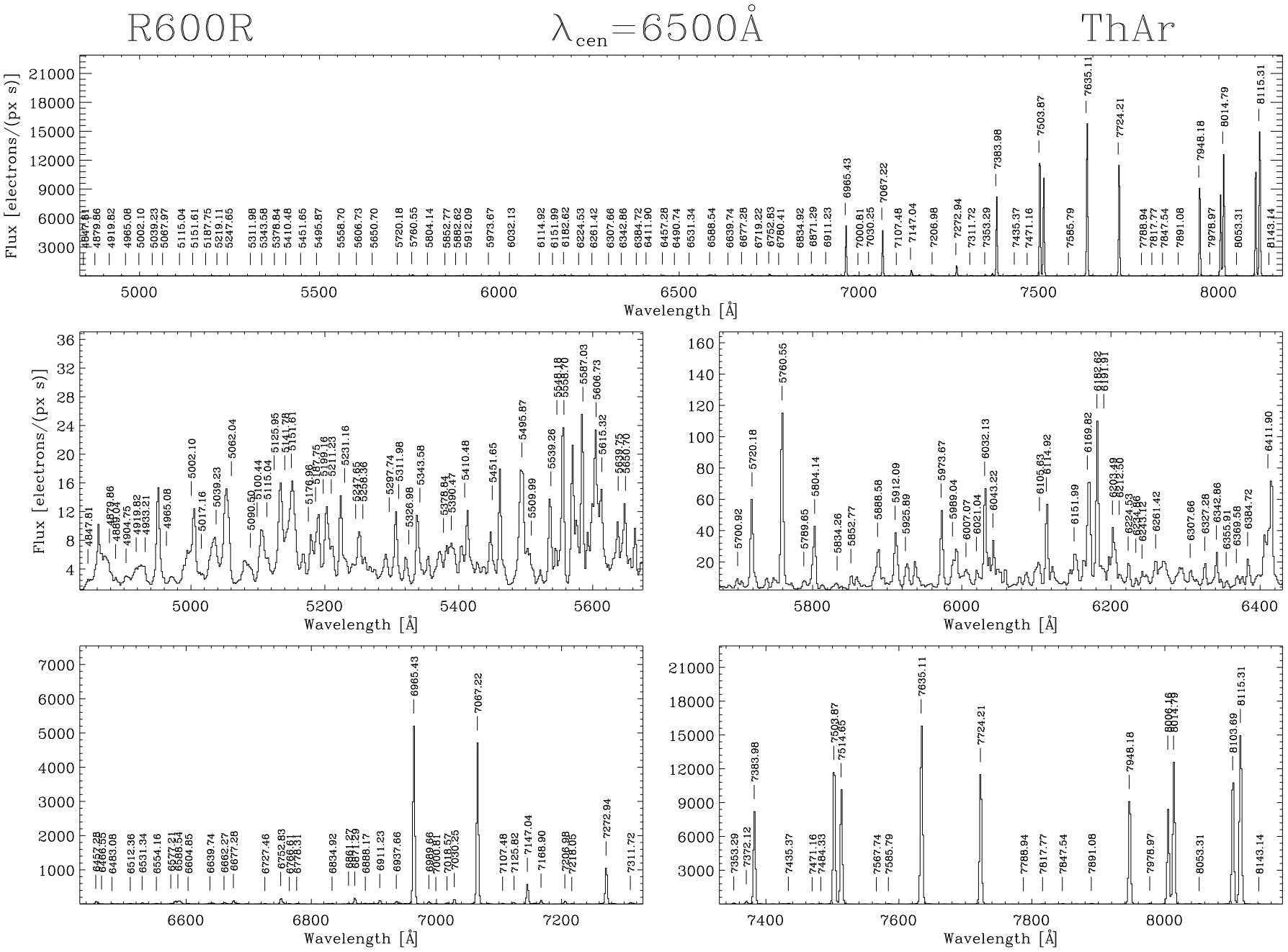

R600B
λcen=5000Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]

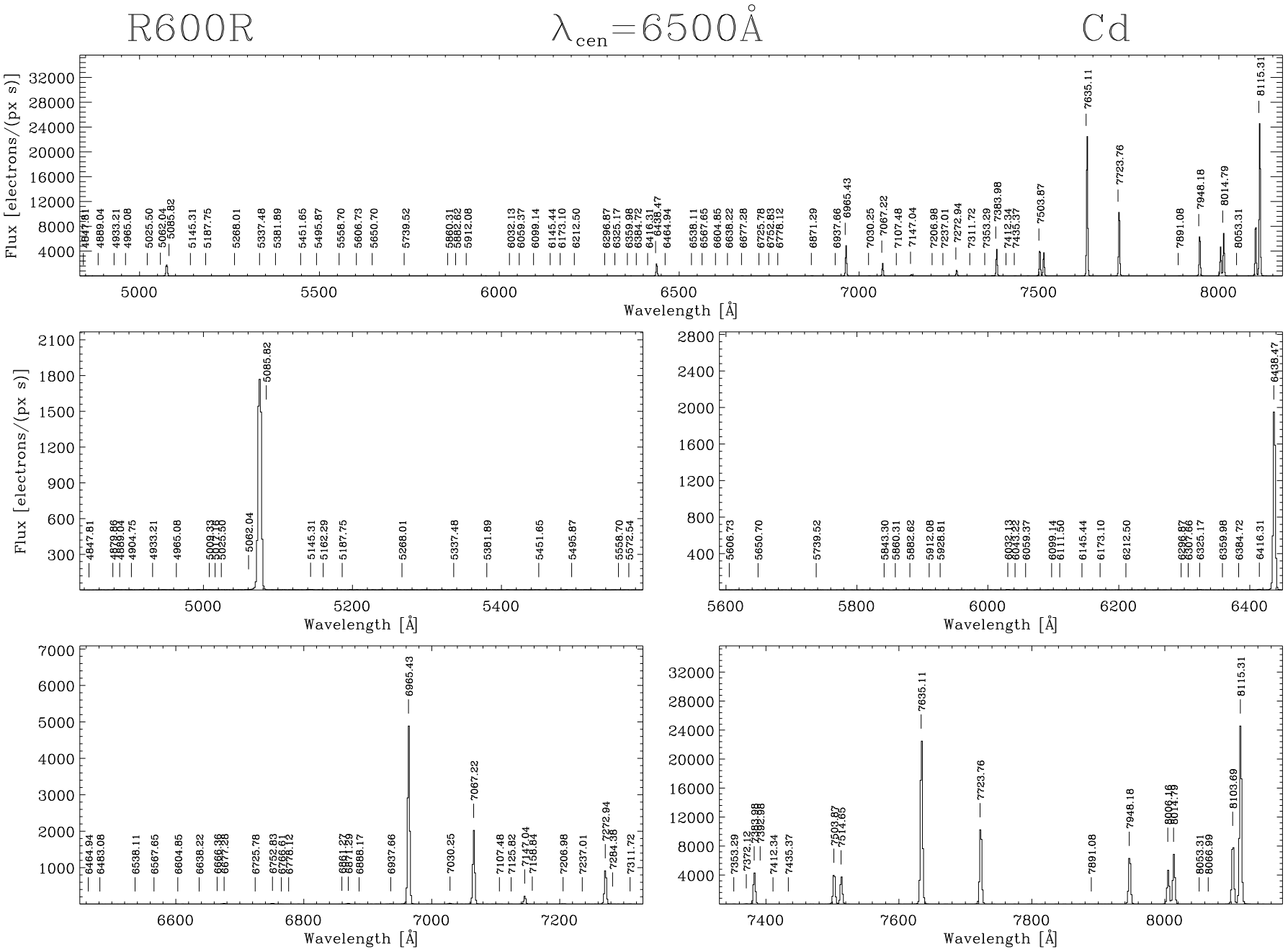

R600B
λcen=5000Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
3403.65
3467.65
3509.78
3561.03
3568.44
3610.51
3850.58
3948.98
4013.86
4044.42
4072.00
4103.91
4134.77
4158.59
4200.67
4277.53
4300.10
4333.56
4379.88
4415.63
4481.81
4510.73
4545.05
4579.35
4609.57
4657.90
4678.15
4726.87
4764.86
4847.81
4879.86
4904.75
4933.21
4965.08
5017.16
5062.04
5085.82
5141.78
5162.29
5187.75
5271.60
5337.48
5378.13
5451.65
5495.87
5572.54
5606.73
5650.70
5739.52
5860.31
5888.58
5912.08
6032.13
6059.37
6099.14
6145.44
6173.10
6212.50
6296.87
6325.17
6359.98
6384.72
6416.31
6438.47
6464.94
6538.11
6567.65
6604.85
6638.22

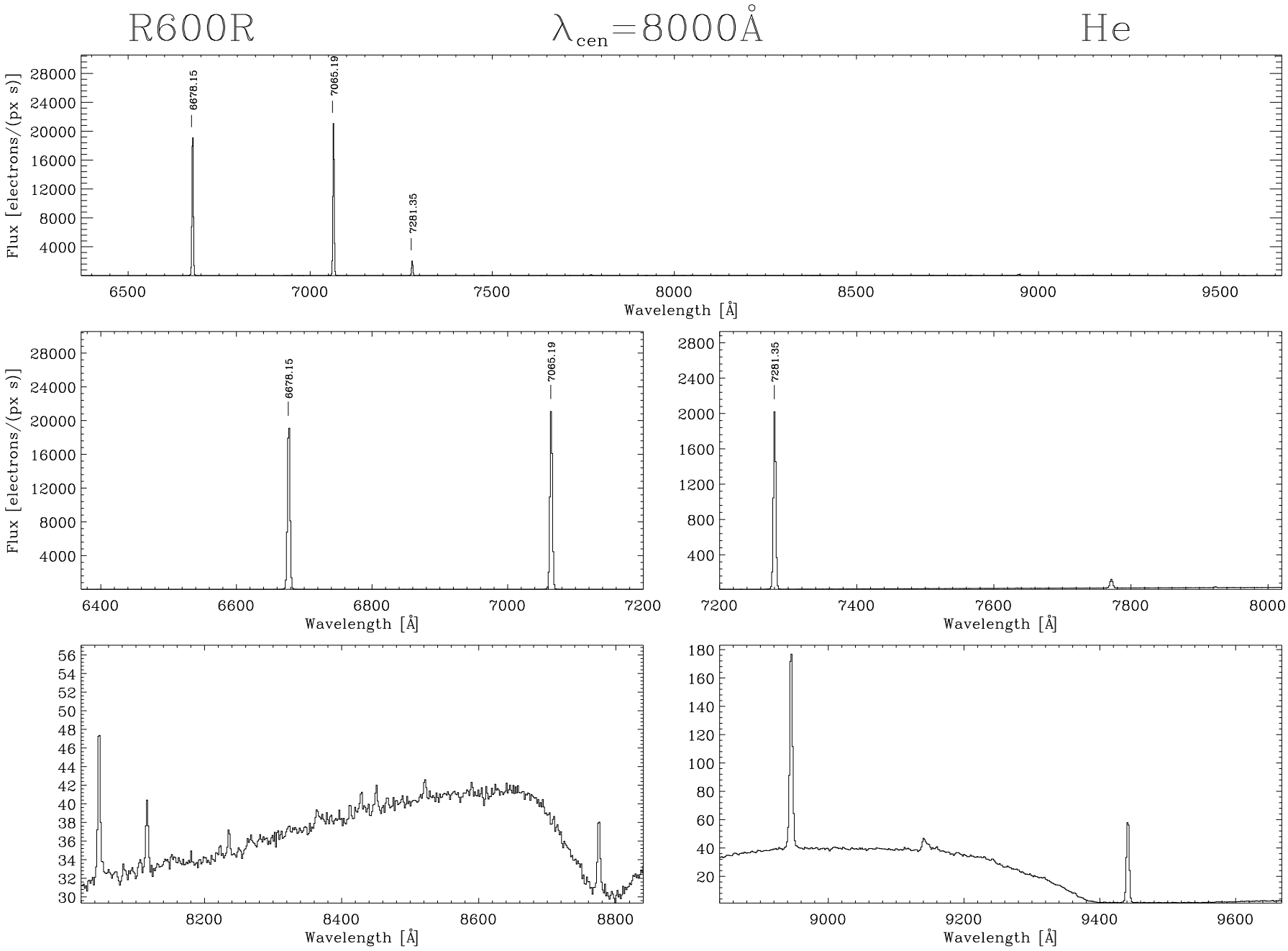

R600R
λcen=6500Å
He
Flux [electrons/(px s)]
Wavelength [Å]
4921.93
5015.68
5047.74
5875.62
6678.15
7065.19
7281.35

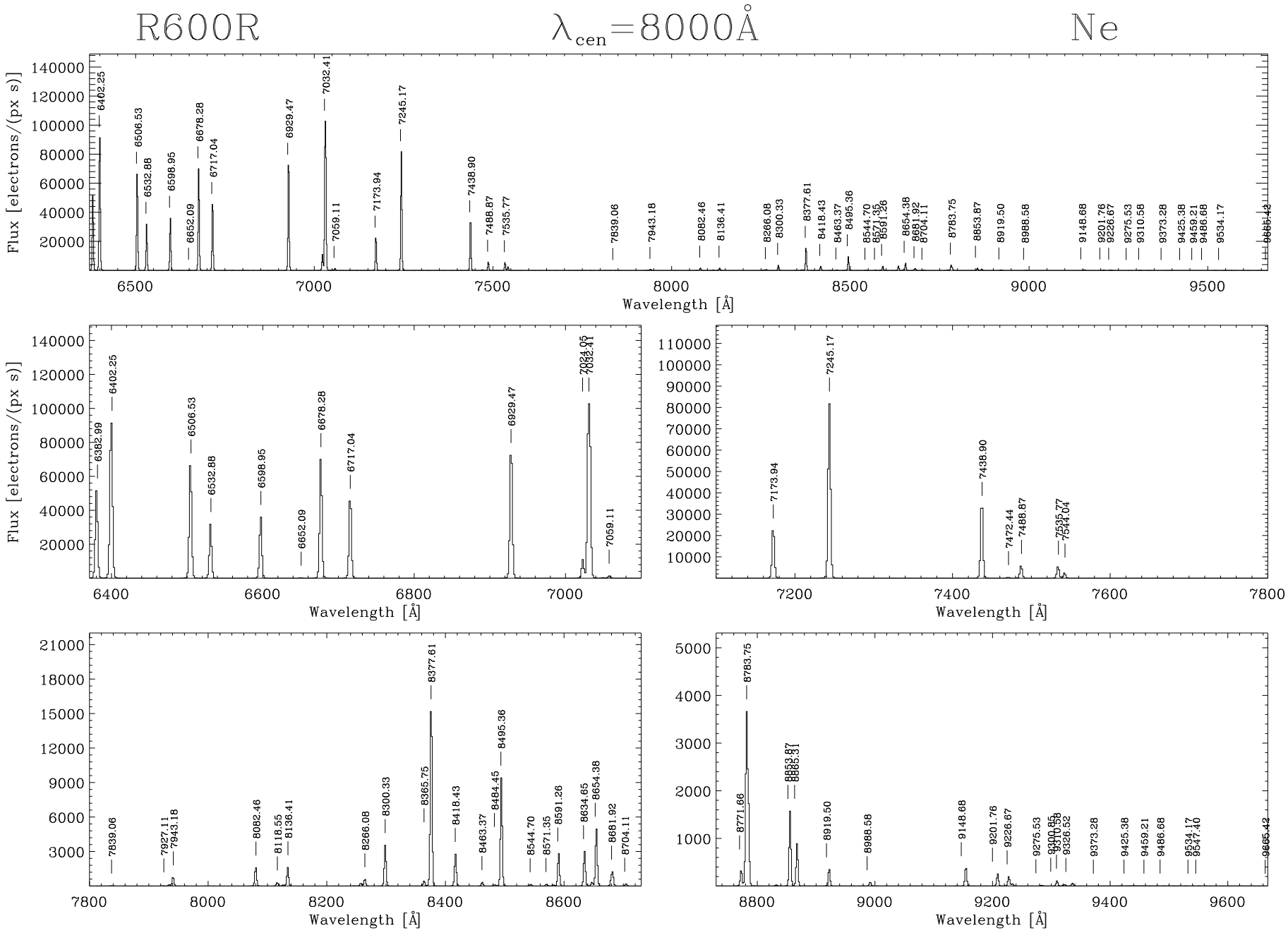

R600R
λcen=6500Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
4837.81
4957.03
5005.16
5037.75
5080.38
5116.50
5144.97
5203.90
5234.03
5274.04
5298.19
5341.09
5400.56
5433.65
5656.66
5689.82
5719.23
5764.42
5820.16
5852.49
5881.90
5902.46
5944.83
5975.53
6030.00
6074.34
6096.16
6143.06
6163.59
6217.28
6266.50
6304.79
6334.43
6402.25
6506.53
6532.88
6598.95
6652.09
6678.28
6717.04
6929.47
7032.41
7059.11
7173.94
7245.17
7438.90
7488.87
7535.77
7839.06
7943.18
8082.46
8128.91
5330.78
5341.09
5360.01
5412.65
5748.30
5804.10
6128.45
6382.99

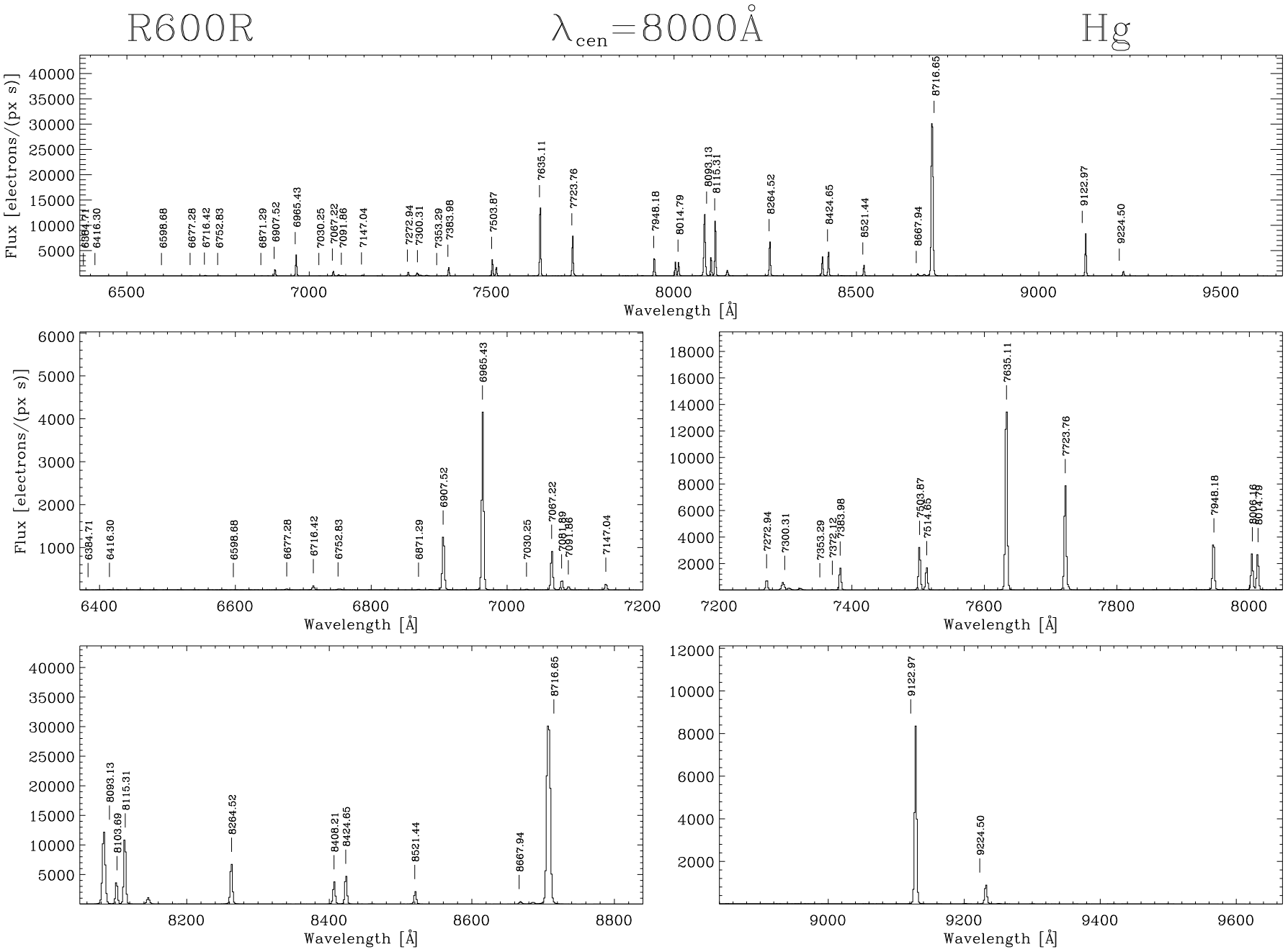

R600R
λcen=6500Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
4916.07
5017.16
5460.74
5769.60
5790.66
5882.62
5942.67
6032.12
6072.72
6098.80
6145.44
6234.39
6384.71
6416.30
6598.68
6677.28
6716.42
6752.83
6871.29
6907.52
6965.43
7030.25
7067.22
7081.89
7091.86
7147.04
7272.94
7300.31
7353.29
7372.12
7383.98
7503.87
7514.65
7635.11
7723.76
7948.18
8006.18
8014.79
8093.13
8103.69
8115.31

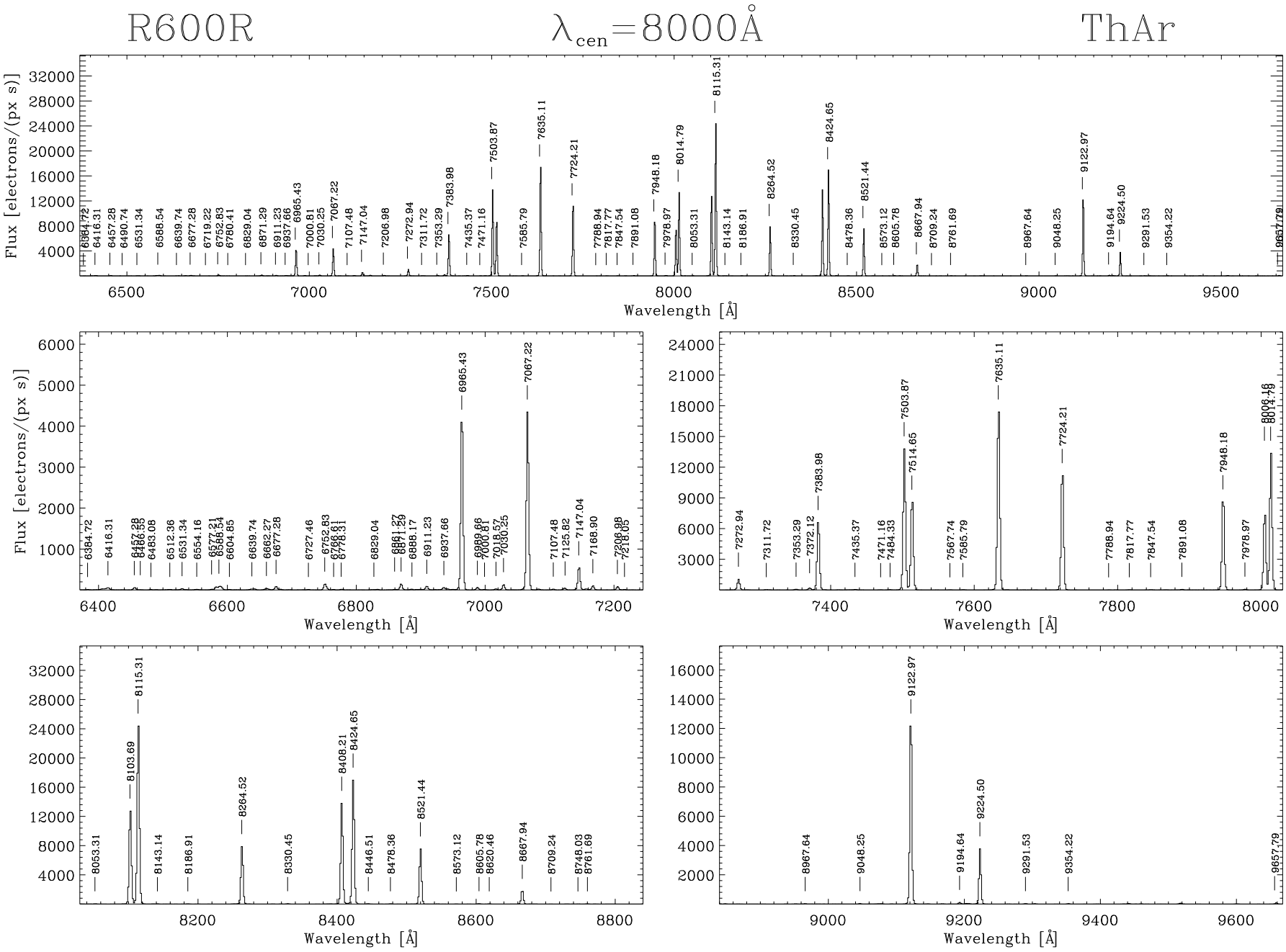

R600R
λcen=6500Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]

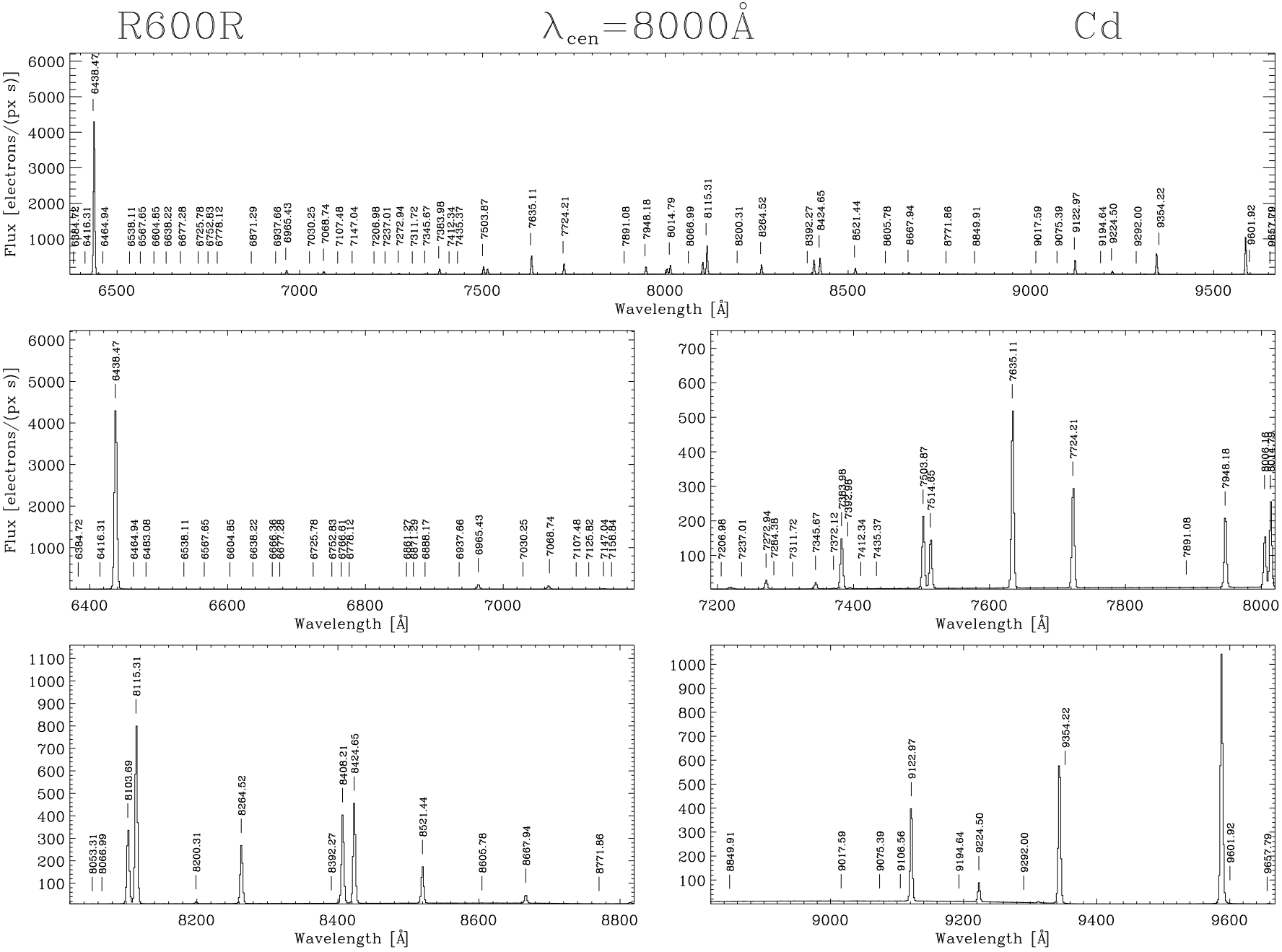

R600R
λcen=6500Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
4847.81
4889.04
4933.21
4965.08
5025.50
5062.04
5085.82
5145.31
5187.75
5268.01
5337.48
5381.89
5451.65
5495.87
5558.70
5606.73
5650.70
5739.52
5860.31
5882.62
5912.08
6032.13
6059.37
6099.14
6145.44
6173.10
6212.50
6296.87
6325.17
6359.98
6384.72
6416.31
6438.47
6464.94
6538.11
6567.65
6604.85
6638.22
6677.28
6725.78
6752.83
6778.12
6871.29
6937.66
6965.43
7030.25
7067.22
7107.48
7147.04
7206.98
7237.01
7272.94
7311.72
7353.29
7383.98
7412.34
7435.37
7503.87
7635.11
7723.76
7891.08
7948.18
8014.79
8053.31
8115.31

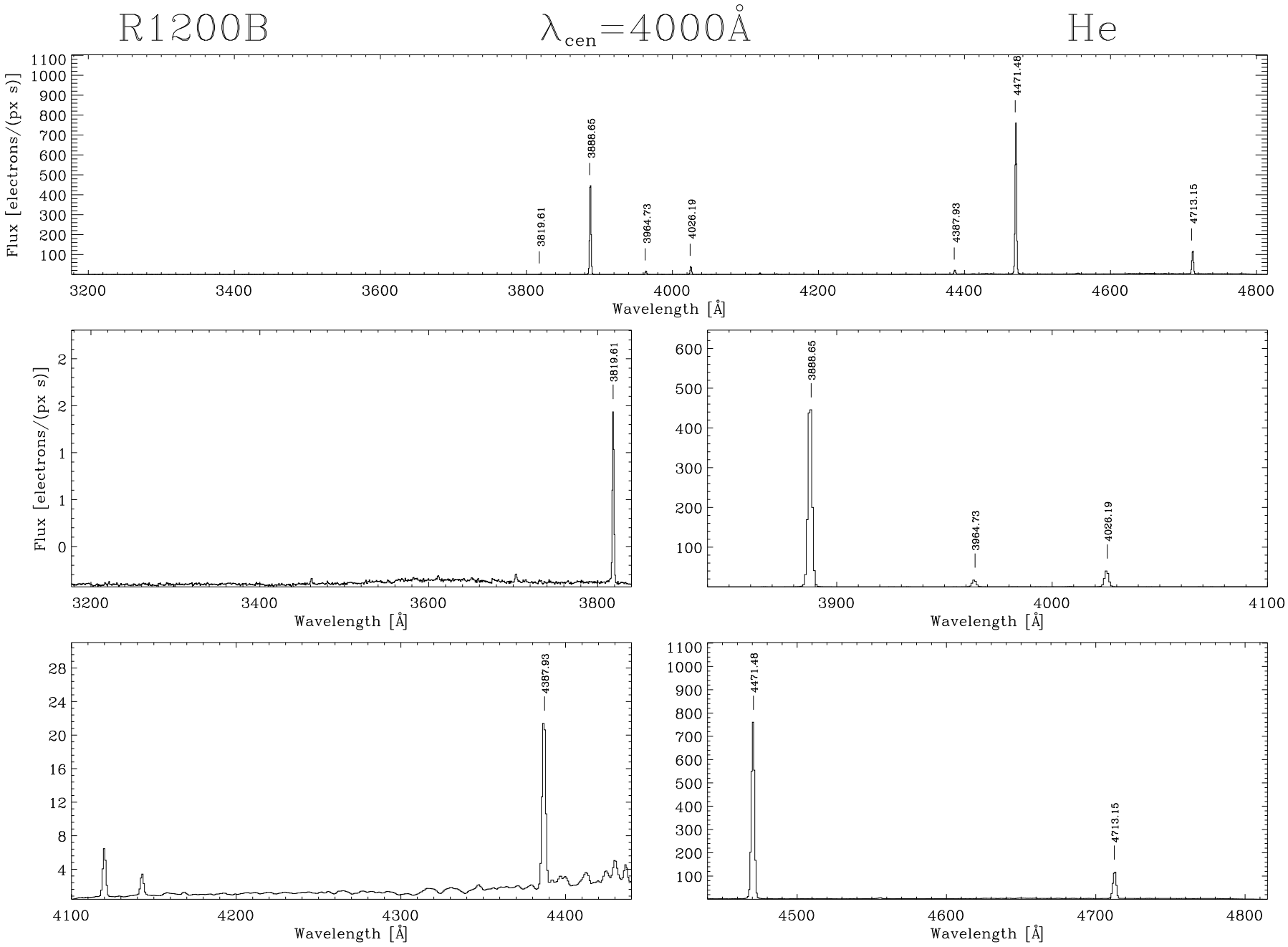

R600R
λcen=8000Å
He
Flux [electrons/(px s)]
Wavelength [Å]
6678.15
7065.19
7281.35

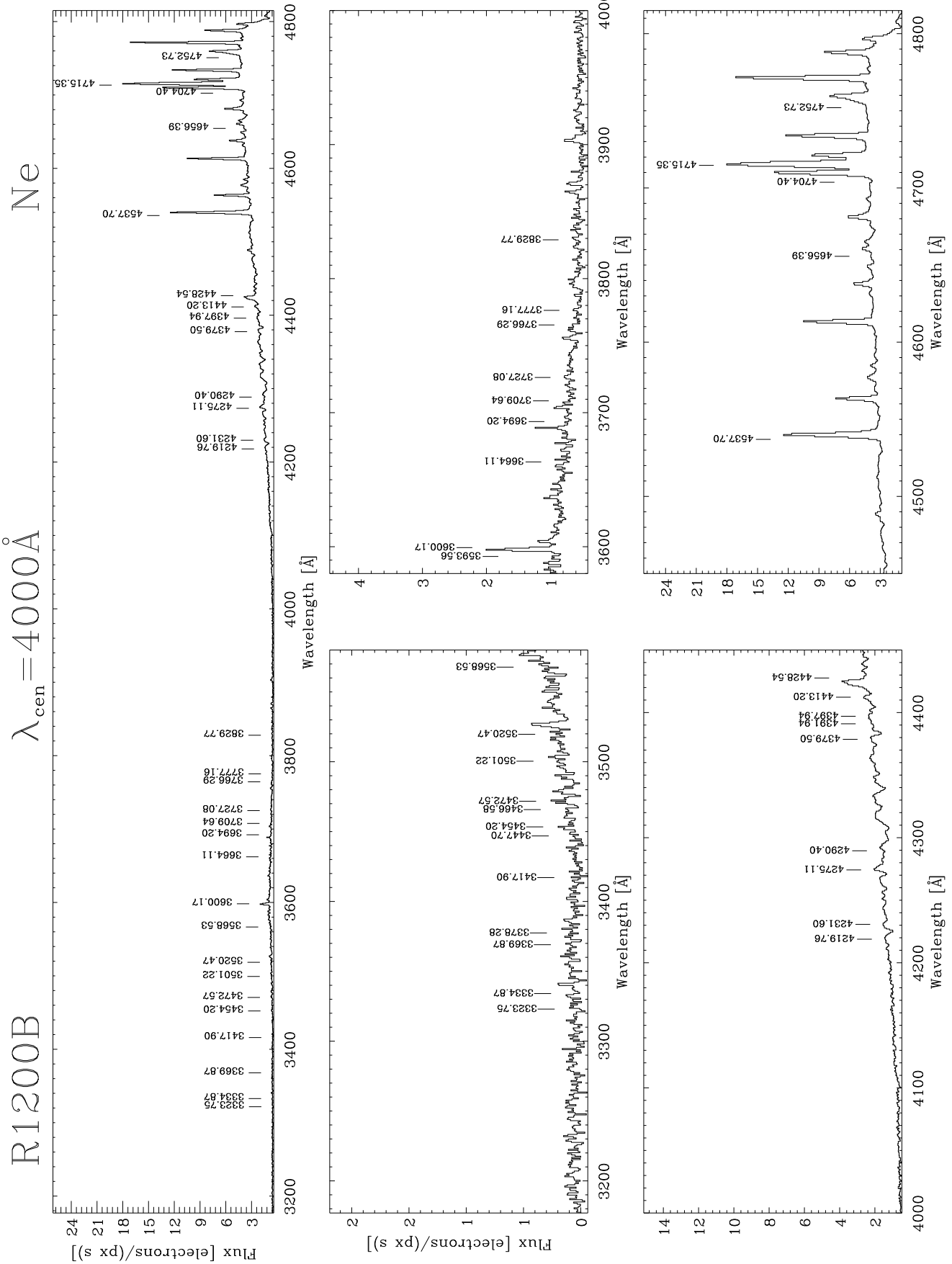

R600R
λcen=8000Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
6402.25
6506.53
6532.88
6598.95
6652.09
6678.28
6717.04
6929.47
7032.41
7059.11
7173.94
7245.17
7438.90
7488.87
7535.77
7839.06
7943.18
8082.46
8136.41
8266.08
8300.33
8377.61
8418.43
8463.37
8495.36
8544.70
8591.26
8654.38
8681.92
8704.11
8783.75
8853.87
8919.50
8988.58
9148.68
9201.76
9226.67
9275.53
9310.58
9373.28
9425.38
9459.21
9486.68
9534.17
9665.42
6382.99
7024.05
7472.44
7544.04
8118.55
8571.35
8634.65
8771.66
8865.31
9326.52
9547.40

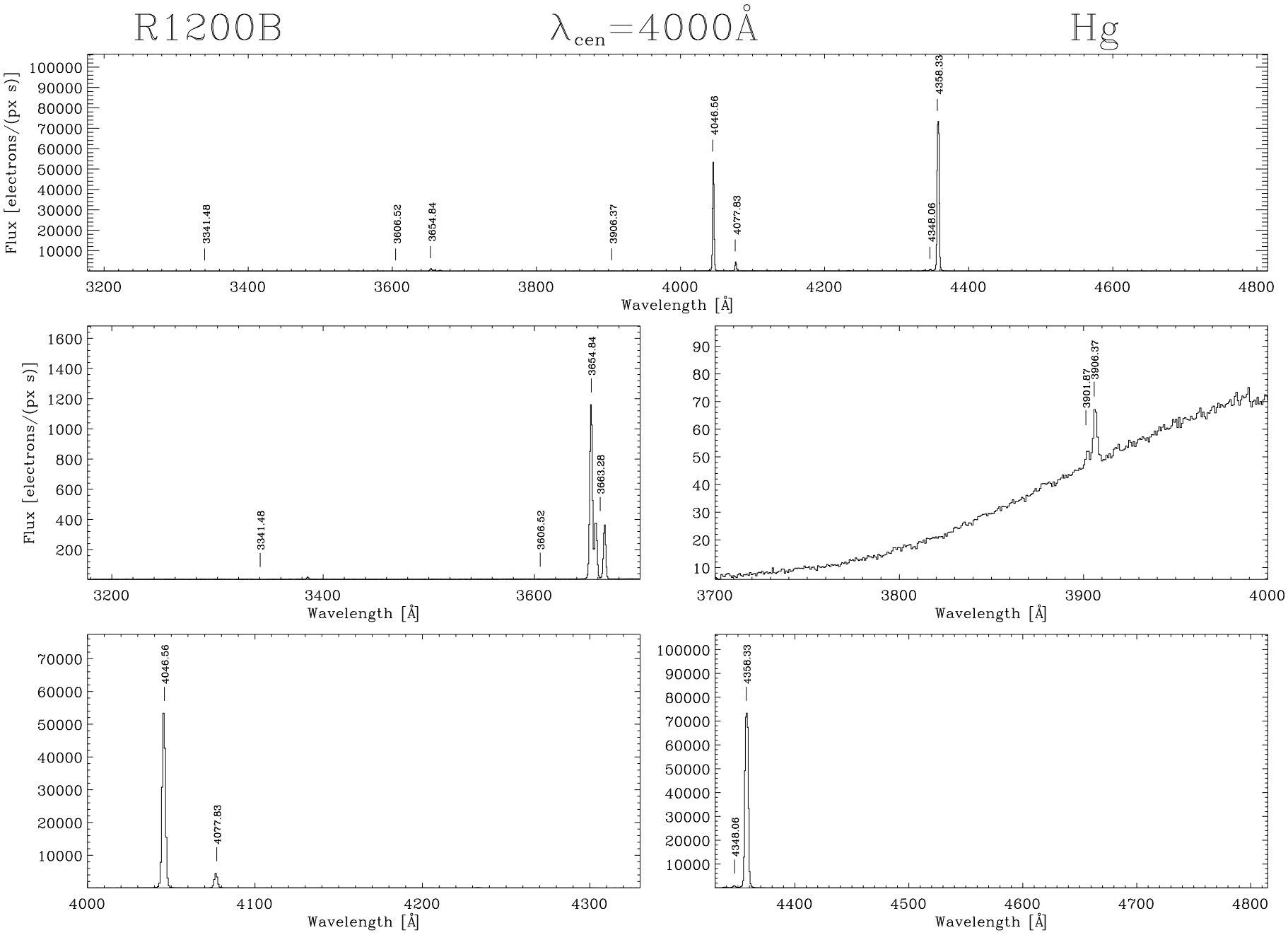

R600R
λcen=8000Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
6384.71
6416.30
6598.68
6677.28
6716.42
6752.83
6871.29
6907.52
6965.43
7030.25
7067.22
7091.86
7147.04
7272.94
7300.31
7353.29
7383.98
7503.87
7635.11
7723.76
7948.18
8014.79
8093.13
8115.31
8264.52
8424.65
8521.44
8667.94
8716.65
9122.97
9224.50

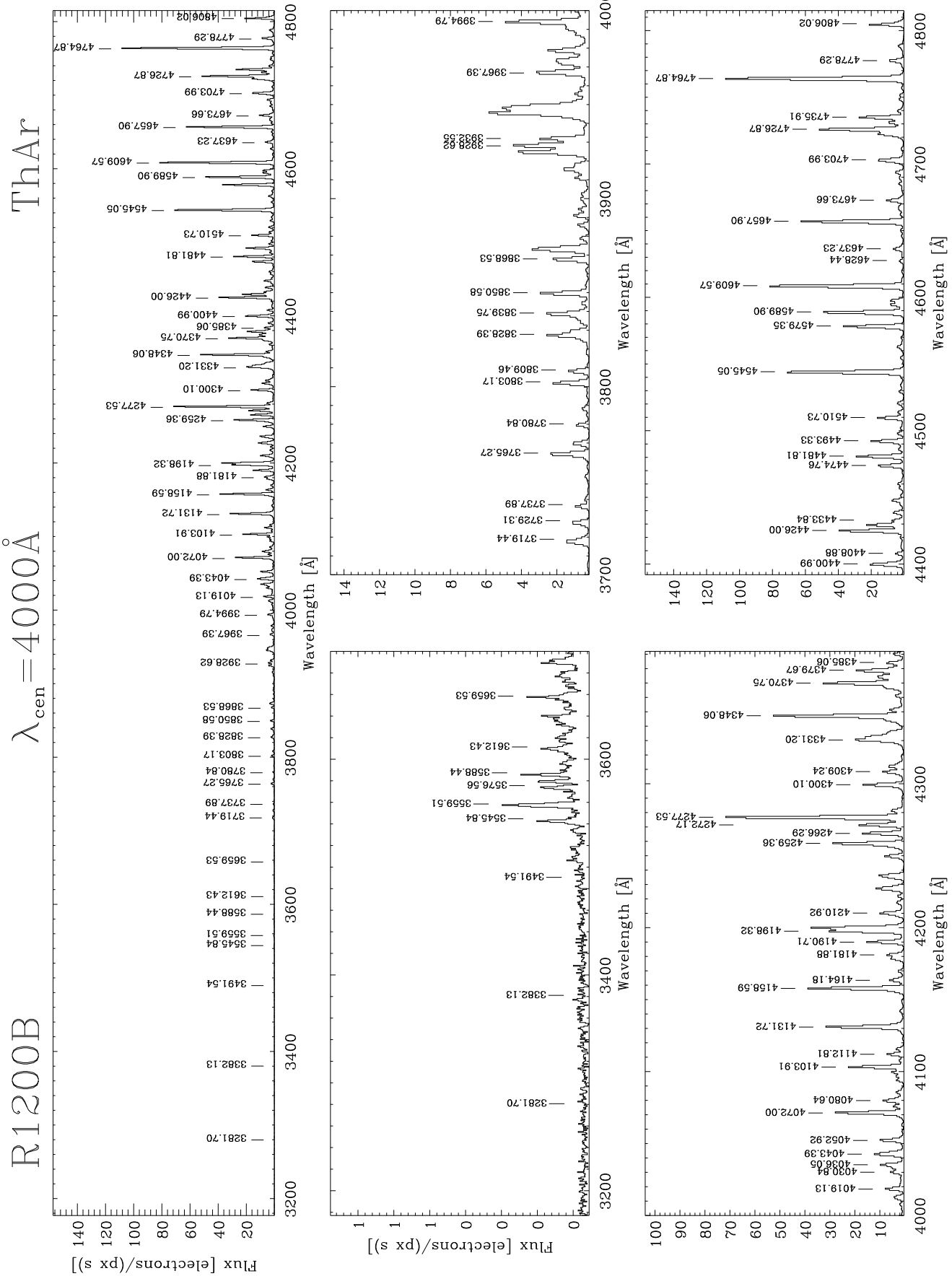

R600R
λcen=8000Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
6384.72
6416.31
6457.28
6490.74
6531.34
6588.54
6639.74
6677.28
6719.22
6752.83
6780.41
6829.04
6871.29
6911.23
6937.66
6965.43
7000.81
7030.25
7067.22
7107.48
7147.04
7206.98
7272.94
7311.72
7353.29
7383.98
7435.37
7471.16
7503.87
7585.79
7635.11
7724.21
7788.94
7817.77
7847.54
7891.08
7948.18
7978.97
8014.79
8053.31
8115.31
8143.14
8186.91
8264.52
8330.45
8424.65
8478.36
8521.44
8573.12
8605.78
8667.94
8709.24
8761.69
8967.64
9048.25
9122.97
9194.64
9224.50
9291.53
9354.22
9657.79
6384.72
6416.31
6457.28
6465.88
6483.08
6512.36
6531.34
6554.16
6577.21
6588.54
6604.85
6639.74
6662.27
6677.28
6727.46
6752.83
6766.61
6778.31
6829.04
6861.27
6888.17
6911.23
6937.66
6965.43
7067.22
7107.48
7125.82
7147.04
7168.90
7206.98
7218.05
7272.94
7311.72
7353.29
7372.12
7383.98
7435.37
7471.16
7484.33
7503.87
7514.65
7567.74
7585.79
7635.11
7724.21
7788.94
7817.77
7847.54
7891.08
7948.18
7978.97
8053.31
8103.69
8115.31
8143.14
8186.91
8264.52
8330.45
8408.21
8424.65
8446.51
8478.36
8521.44
8573.12
8605.78
8620.46
8667.94
8709.24
8748.03
8761.69
8967.64
9048.25
9122.97
9194.64
9224.50
9291.53
9354.22
9657.79

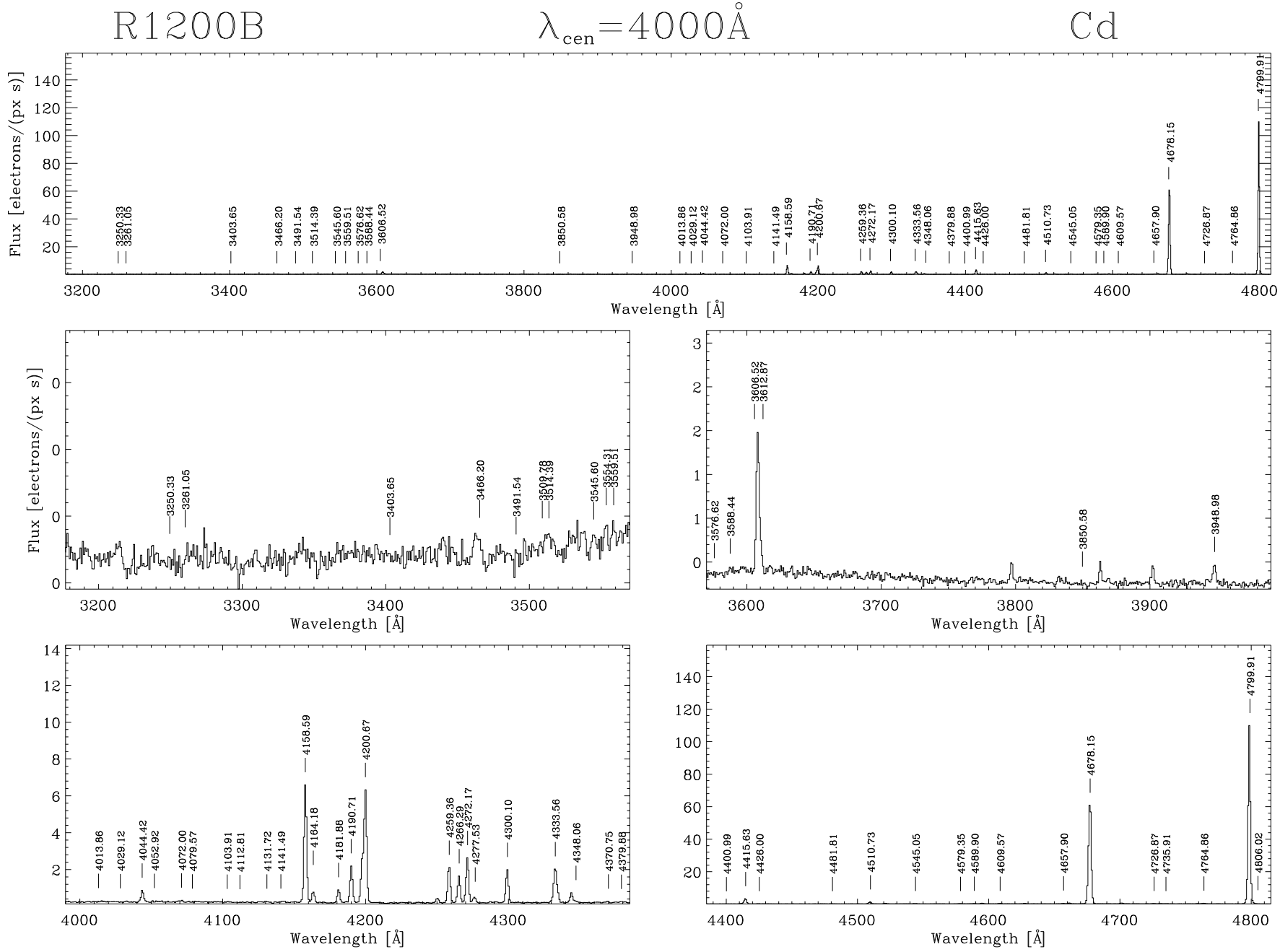

R600R
λcen=8000Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
6384.72
6416.31
6438.47
6484.94
6538.11
6567.65
6604.85
6638.22
6677.28
6725.78
6752.83
6778.12
6871.29
6937.66
6965.43
7030.25
7068.74
7107.48
7147.04
7206.98
7237.01
7272.94
7311.72
7345.67
7383.98
7412.34
7435.37
7503.87
7635.11
7724.21
7891.08
7948.18
8014.79
8066.99
8115.31
8200.31
8264.52
8392.27
8424.65
8521.44
8605.78
8667.94
8771.86
8849.91
9017.59
9075.39
9122.97
9194.64
9224.50
9292.00
9354.22
9601.92
9657.79
6384.72
6416.31
6438.47
6464.94
6483.08
6538.11
6567.65
6604.85
6638.22
6666.36
6677.28
6725.78
6752.83
6766.61
6778.12
6881.27
6888.17
6937.66
6965.43
7030.25
7068.74
7107.48
7125.82
7147.04
7158.84
7206.98
7237.01
7272.94
7284.38
7311.72
7345.67
7372.12
7383.98
7392.98
7412.34
7435.37
7503.87
7514.65
7635.11
7724.21
7891.08
7948.18
8006.16
8014.78
8053.31
8066.99
8103.69
8115.31
8200.31
8264.52
8392.27
8408.21
8424.65
8521.44
8605.78
8667.94
8771.86
8849.91
9017.59
9075.39
9106.56
9122.97
9194.64
9224.50
9292.00
9354.22
9601.92
9657.79

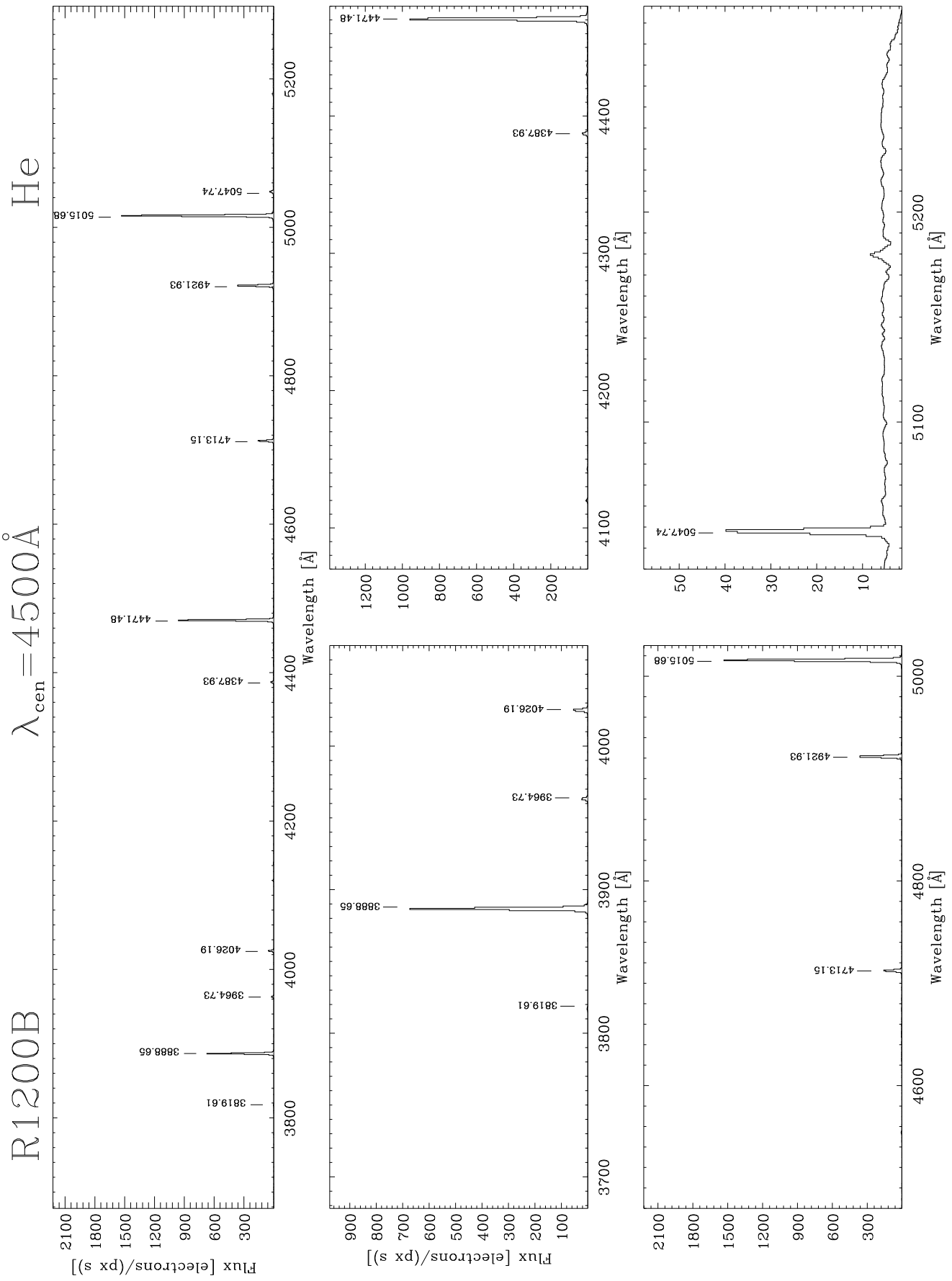

R1200B
λcen=4000Å
He
Flux [electrons/(px s)]
Wavelength [Å]
3819.61
3888.65
3964.73
4026.19
4387.93
4471.48
4713.15

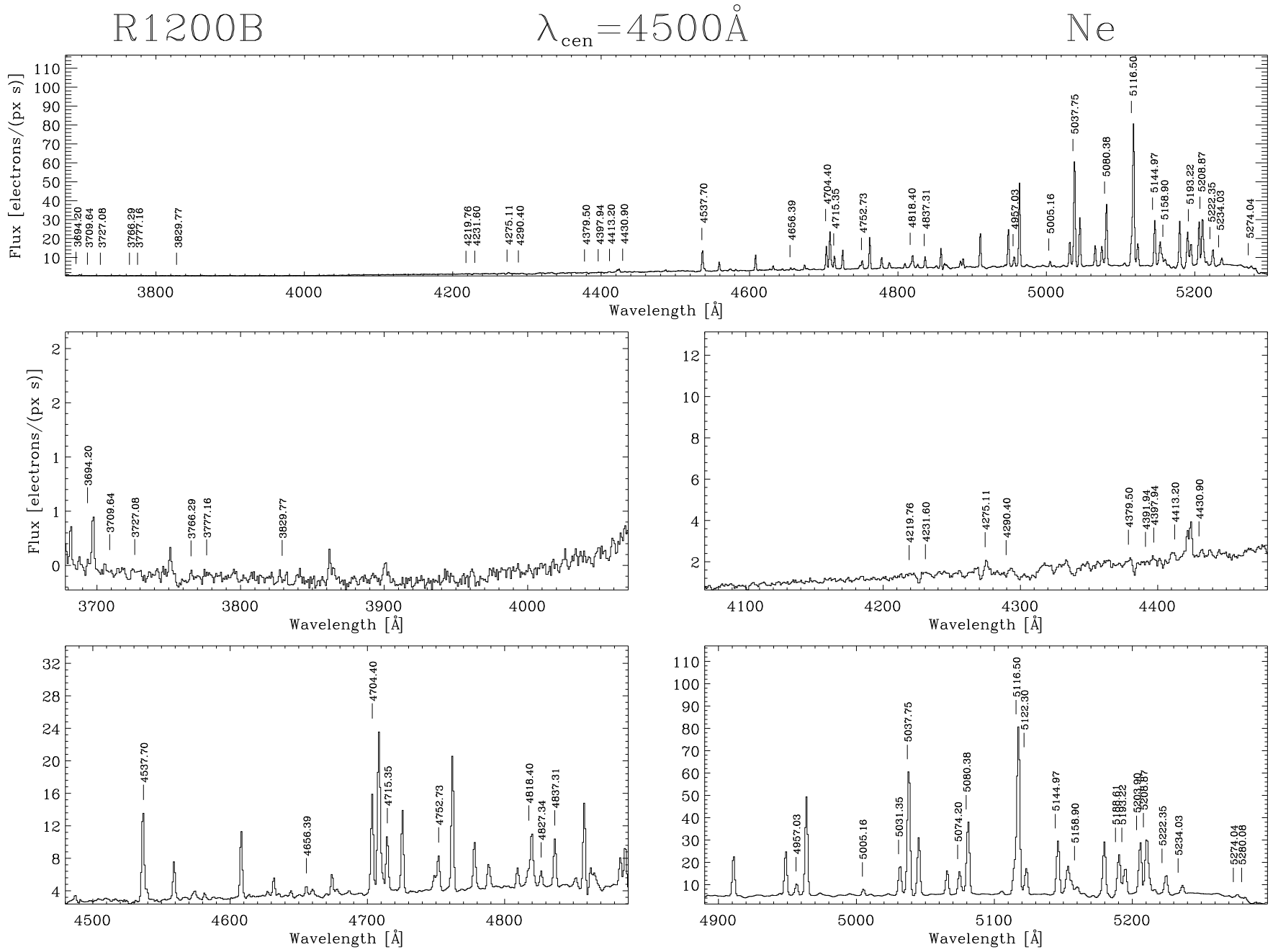

R1200B
λ_{cen}=4000Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
3323.75
3334.87
3369.87
3417.90
3454.20
3472.57
3501.22
3520.47
3568.53
3600.17
3664.11
3694.20
3709.64
3727.08
3766.29
3777.16
3829.77
4219.76
4231.60
4275.11
4290.40
4379.50
4397.94
4413.20
4428.54
4537.70
4656.39
4704.40
4715.35
4752.73

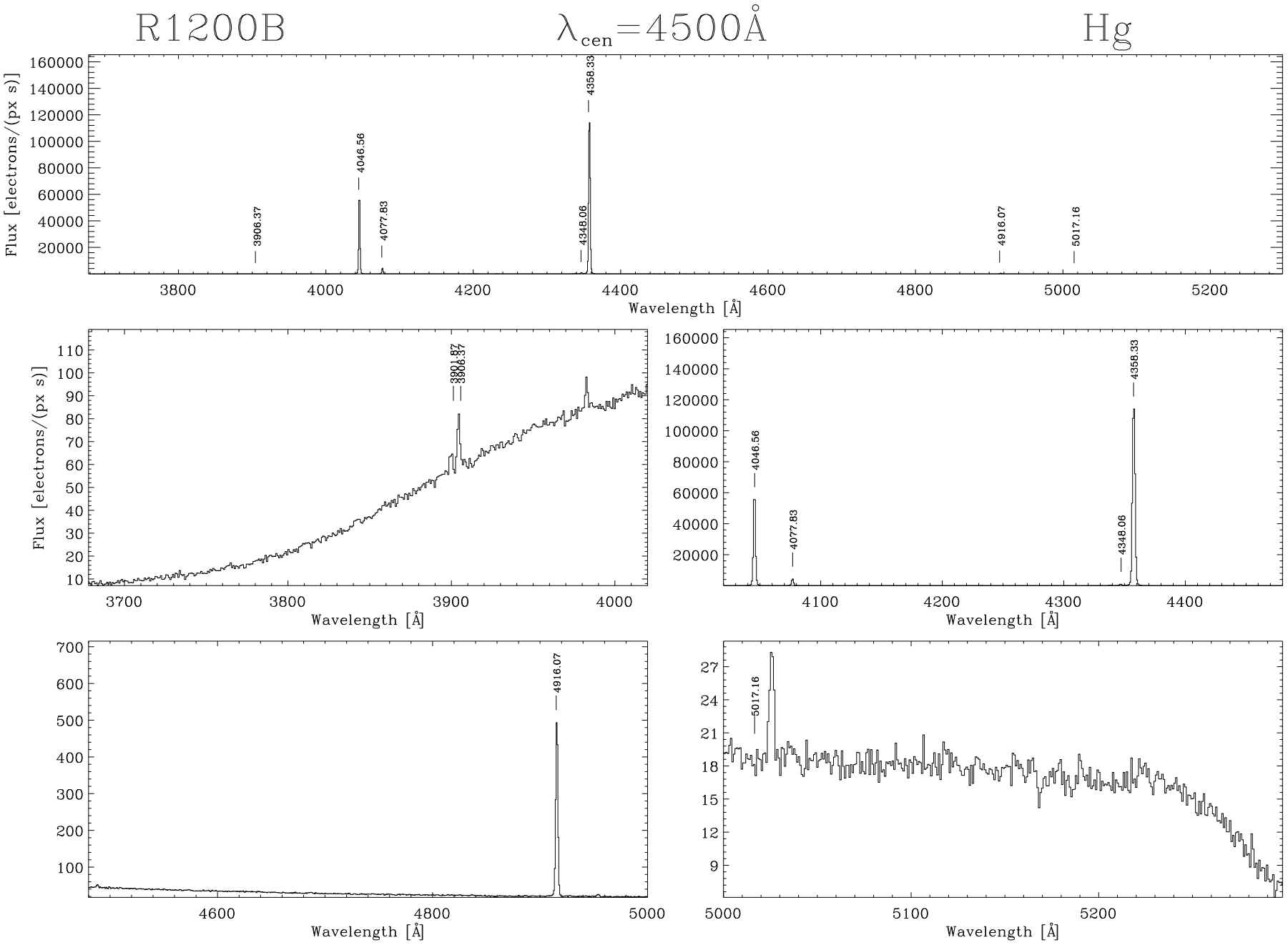

R1200B
λcen=4000Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
3341.48
3606.52
3654.84
3906.37
4046.56
4077.83
4348.06
4358.33
3663.28
3901.87

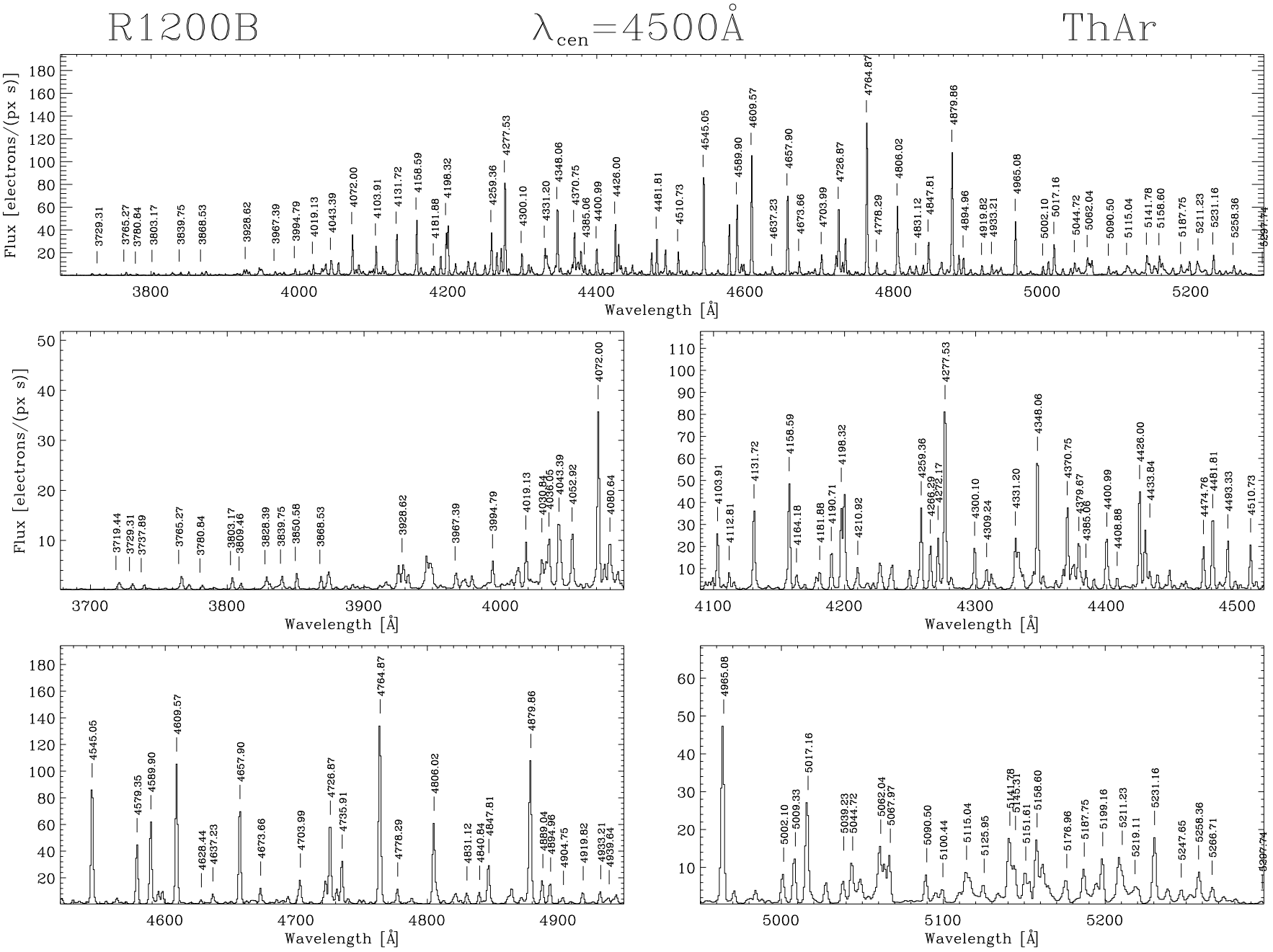

R1200B
λ_{cen}=4000Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
3281.70
3382.13
3491.54
3545.84
3559.51
3588.44
3612.43
3659.53
3719.44
3737.89
3765.27
3780.84
3803.17
3828.39
3850.58
3868.53
3928.62
3967.39
3994.79
4019.13
4043.39
4072.00
4103.91
4131.72
4158.59
4181.88
4198.32
4259.36
4277.53
4300.10
4331.20
4348.06
4370.75
4385.06
4400.99
4426.00
4481.81
4510.73
4545.05
4589.90
4609.57
4637.23
4657.90
4673.66
4703.99
4726.87
4764.87
4778.29
4806.02

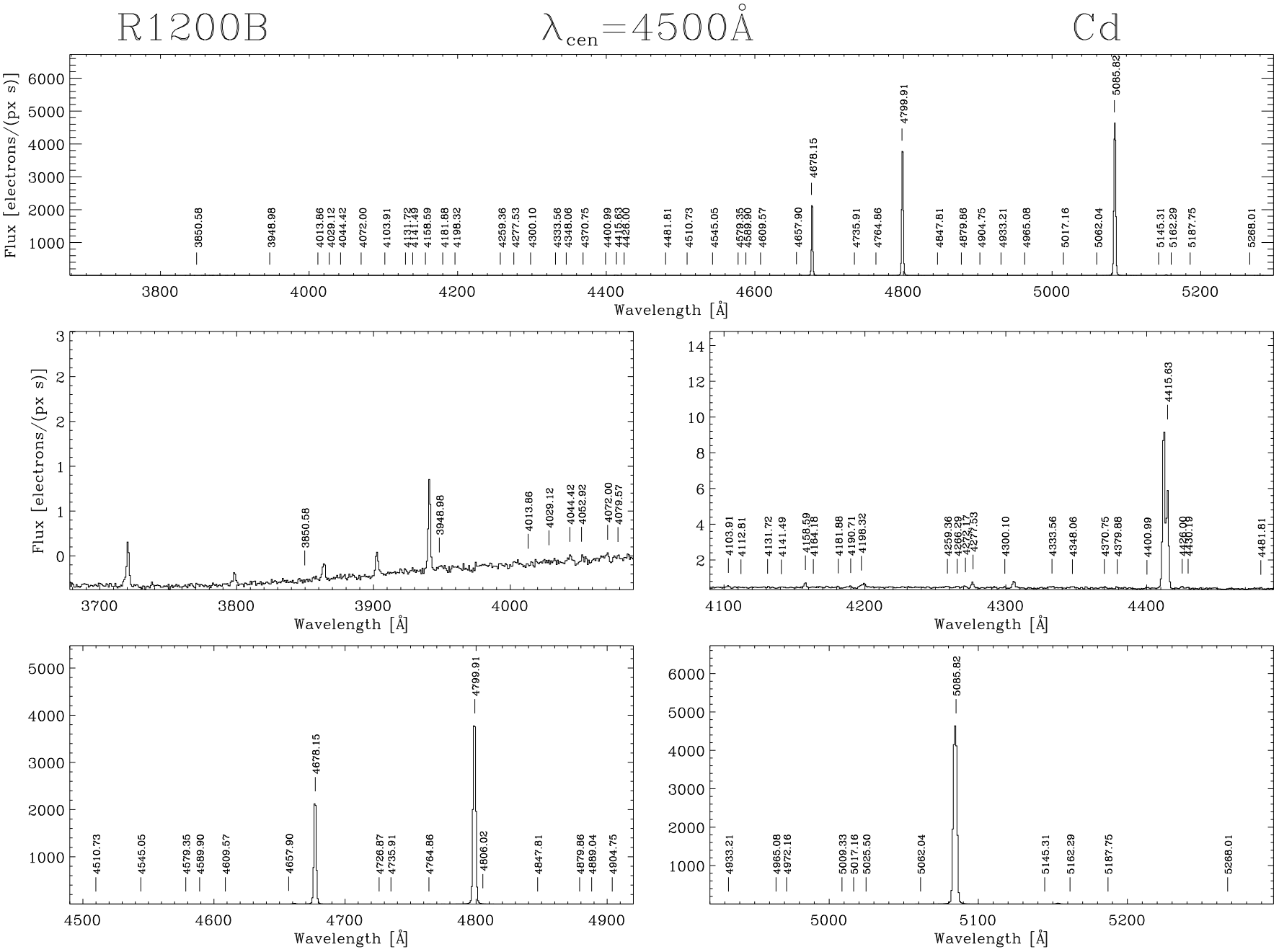

R1200B
λcen=4000Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
3250.33
3261.05
3403.65
3466.20
3491.54
3514.39
3545.60
3559.51
3576.62
3588.44
3606.52
3850.58
3948.98
4013.86
4029.12
4044.42
4072.00
4103.91
4141.49
4158.59
4180.71
4200.67
4259.36
4272.17
4300.10
4333.56
4348.06
4379.88
4400.99
4415.63
4426.00
4481.81
4510.73
4545.05
4579.35
4589.90
4609.57
4657.90
4678.15
4726.87
4764.86
4799.91

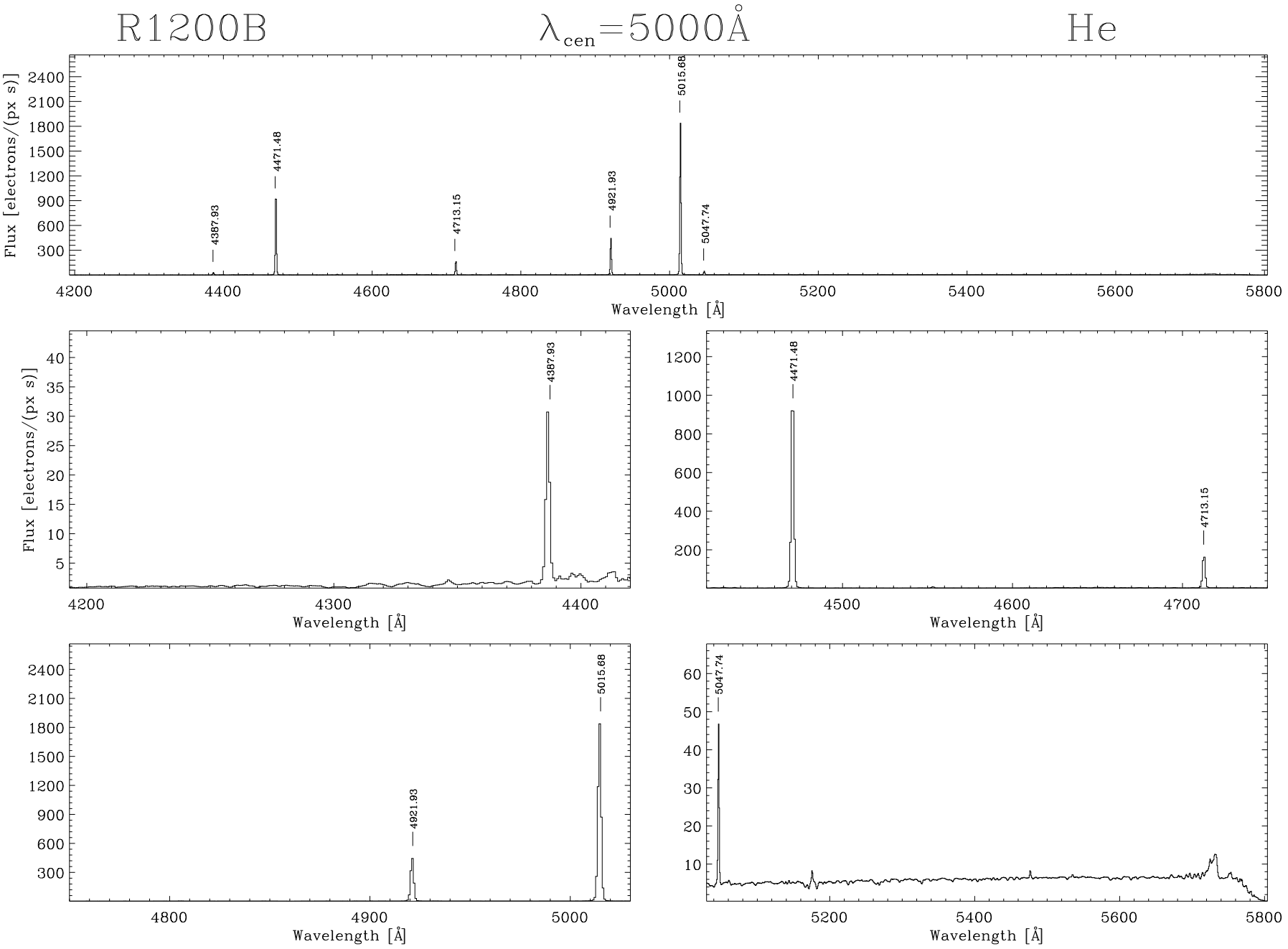

R1200B
λ_{cen}=4500Å
He
Flux [electrons/(px s)]
Wavelength [Å]
3819.61
3888.65
3964.73
4026.19
4387.93
4471.48
4713.15
4921.93
5015.68
5047.74

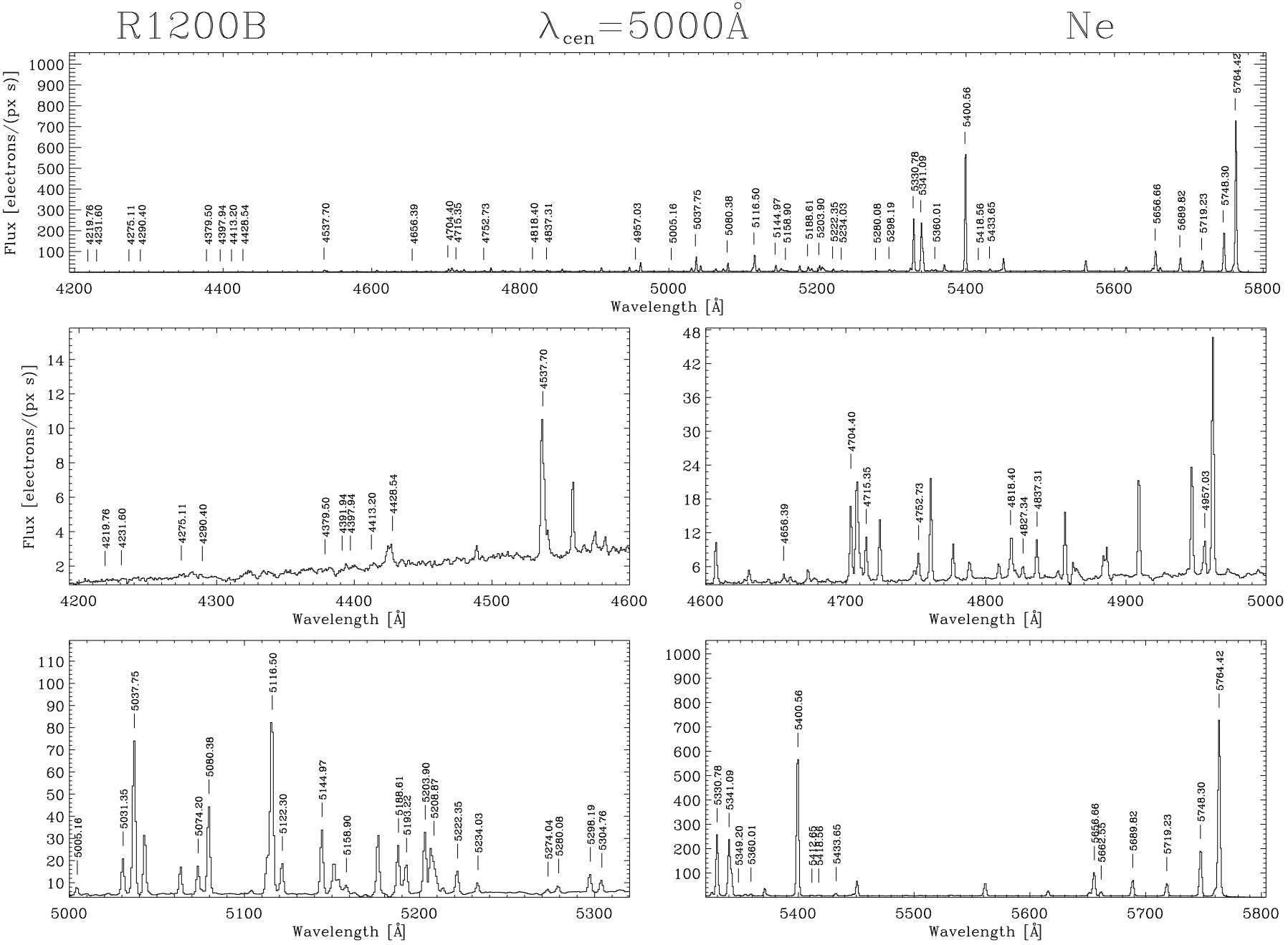

R1200B
λcen=4500Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
3694.20
3709.64
3727.08
3766.29
3777.16
3829.77
4219.76
4231.60
4275.11
4290.40
4379.50
4397.94
4413.20
4430.90
4537.70
4656.39
4704.40
4715.35
4752.73
4818.40
4837.31
4957.03
5005.16
5037.75
5080.38
5116.50
5144.97
5158.90
5193.22
5208.87
5222.35
5234.03
5274.04
4827.34
5031.35
5074.20
5122.30
5280.08

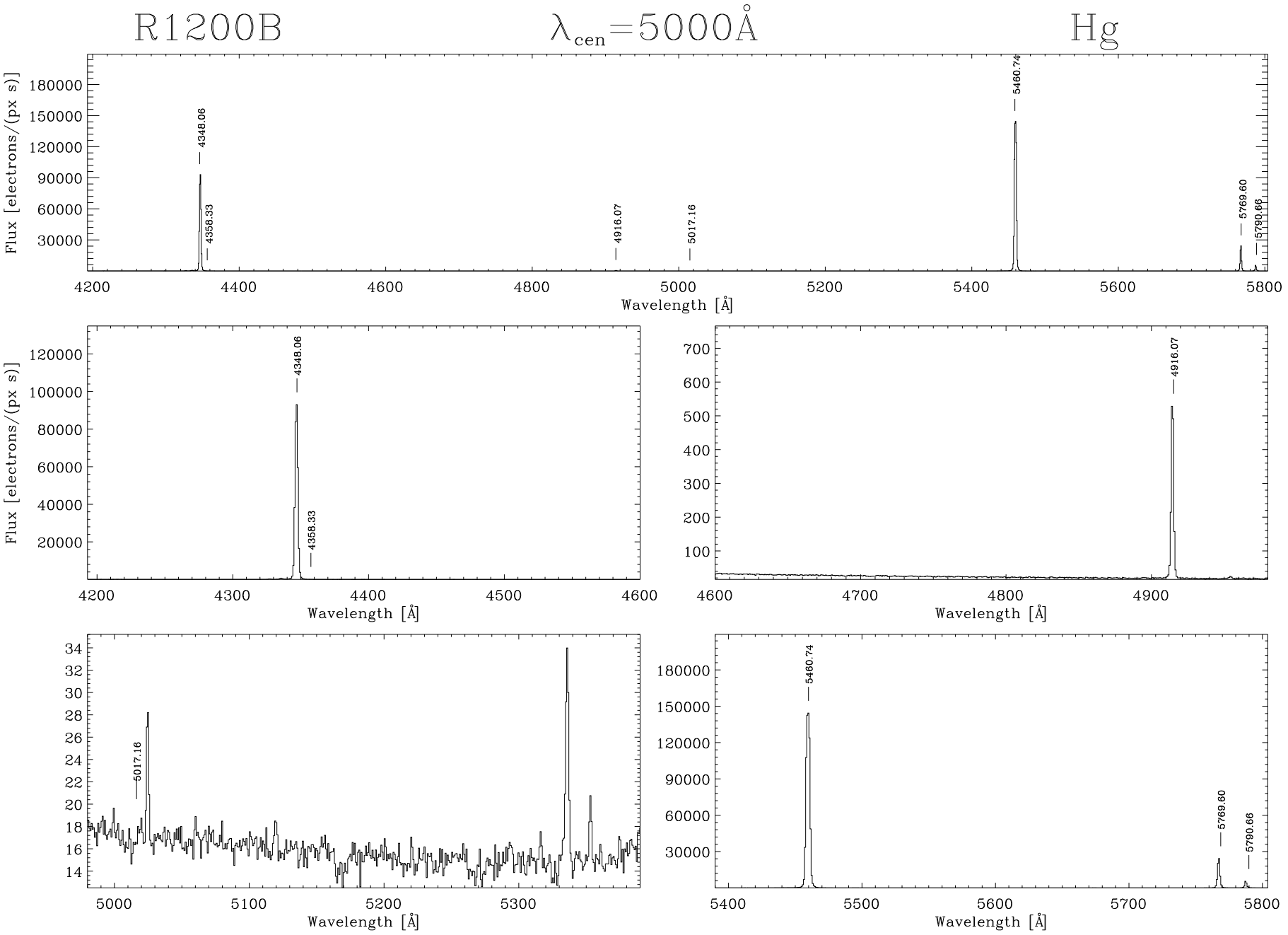

R1200B
λcen=4500Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
3906.37
4046.56
4077.83
4348.06
4358.33
4916.07
5017.16

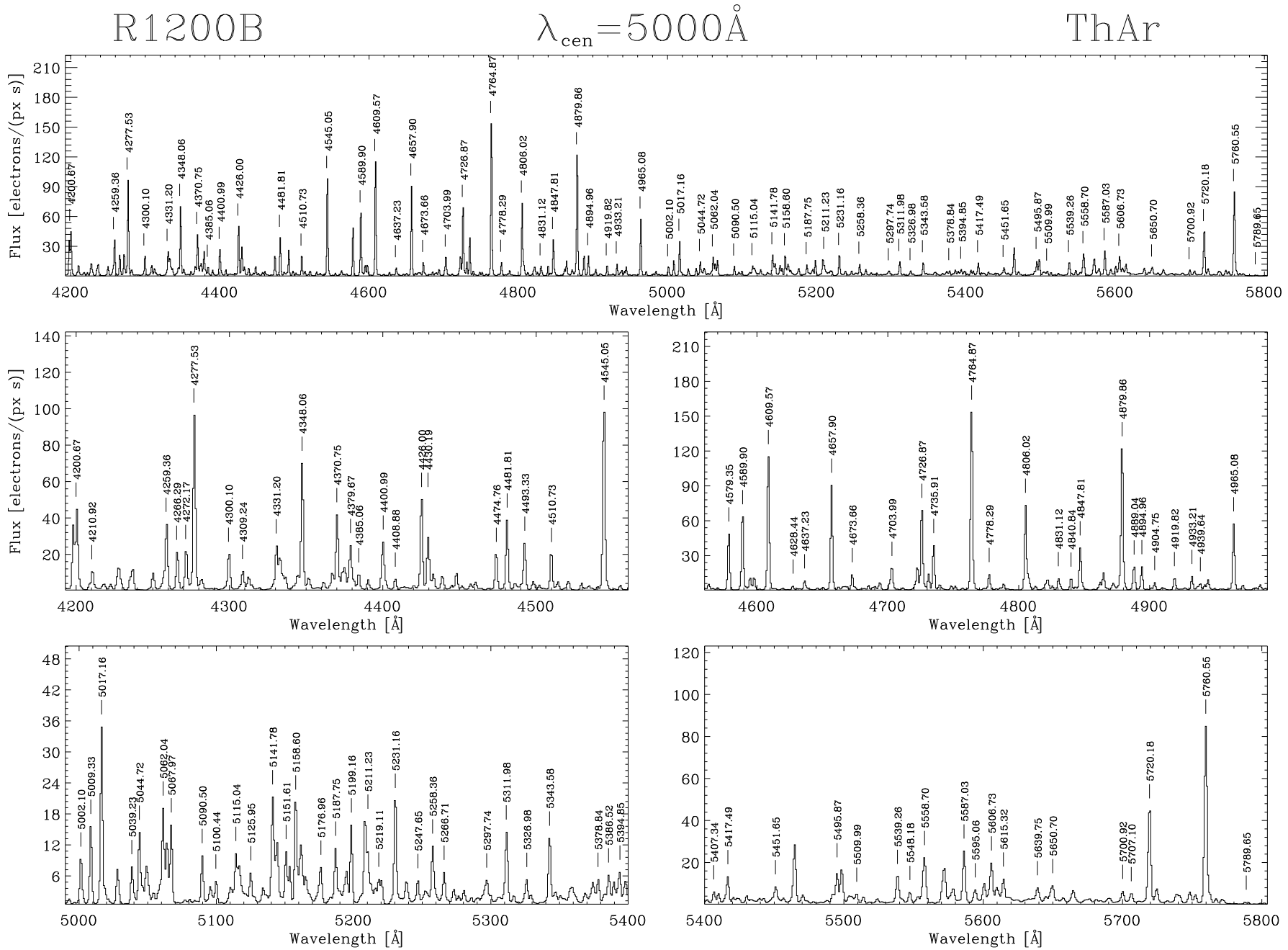

R1200B
λcen=4500Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
3729.31
3765.27
3780.84
3803.17
3839.75
3868.53
3928.62
3967.39
3994.79
4019.13
4043.39
4072.00
4103.91
4131.72
4158.59
4181.88
4198.32
4259.36
4277.53
4300.10
4331.20
4348.06
4370.75
4385.06
4400.99
4426.00
4481.81
4510.73
4545.05
4589.90
4609.57
4637.23
4657.90
4673.66
4703.99
4726.87
4764.87
4778.29
4806.02
4831.12
4847.81
4879.86
4894.96
4919.82
4933.21
4965.08
5002.10
5017.16
5044.72
5062.04
5090.50
5115.04
5141.78
5158.60
5187.75
5211.23
5231.16
5258.36
5291.74
3719.44
3737.89
3809.46
3828.39
3850.58
4030.84
4036.05
4052.92
4080.64
4112.81
4164.18
4190.71
4210.92
4266.29
4272.17
4309.24
4379.67
4408.88
4433.84
4474.76
4493.33
4510.73
4628.44
4840.84
4889.04
4904.75
4939.64
5039.23
5067.97
5100.44
5125.95
5145.31
5151.61
5176.96
5199.16
5219.11
5247.65
5266.71
5297.74

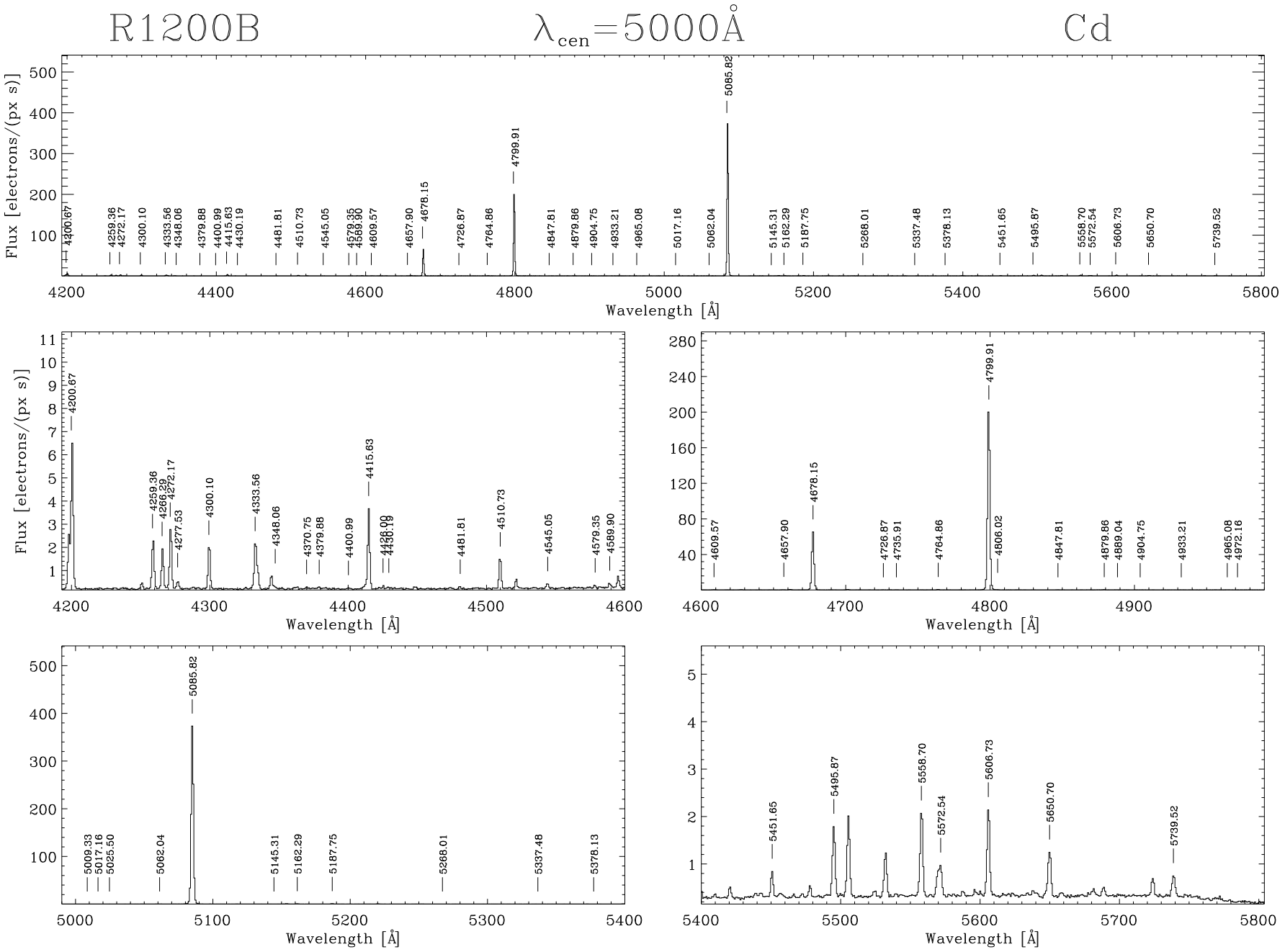

R1200B
λcen=4500Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
3850.58
3948.98
4013.86
4029.12
4044.42
4072.00
4103.91
4158.59
4181.88
4198.32
4259.36
4277.53
4300.10
4333.56
4348.06
4370.75
4481.81
4510.73
4545.05
4657.90
4678.15
4735.91
4764.86
4799.91
4847.81
4879.86
4904.75
4933.21
4965.08
5017.16
5062.04
5085.82
5145.31
5162.29
5187.75
5268.01
4052.92
4079.57
4103.91
4112.81
4131.72
4141.49
4158.59
4164.18
4181.88
4190.71
4198.32
4259.36
4266.29
4272.17
4277.53
4300.10
4333.56
4348.06
4370.75
4379.88
4400.99
4415.63
4481.81
4510.73
4545.05
4579.35
4589.90
4609.57
4657.90
4678.15
4728.87
4735.91
4764.86
4799.91
4806.02
4847.81
4879.86
4889.04
4904.75
4933.21
4965.08
4972.16
5009.33
5017.16
5025.50
5062.04
5085.82
5145.31
5162.29
5187.75
5268.01

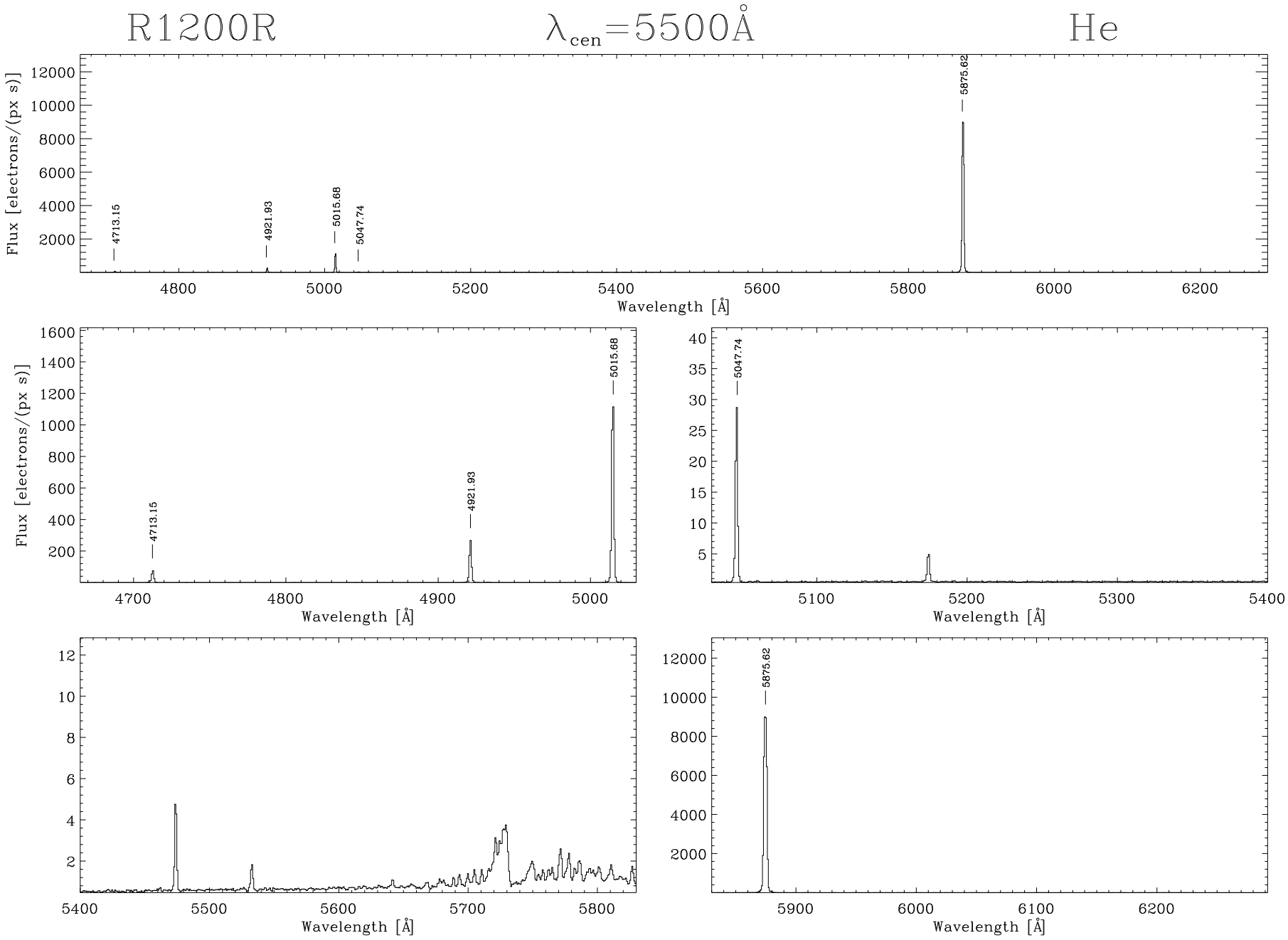

R1200B
λcen=5000Å
He
Flux [electrons/(px s)]
Wavelength [Å]
4387.93
4471.48
4713.15
4921.93
5015.68
5047.74

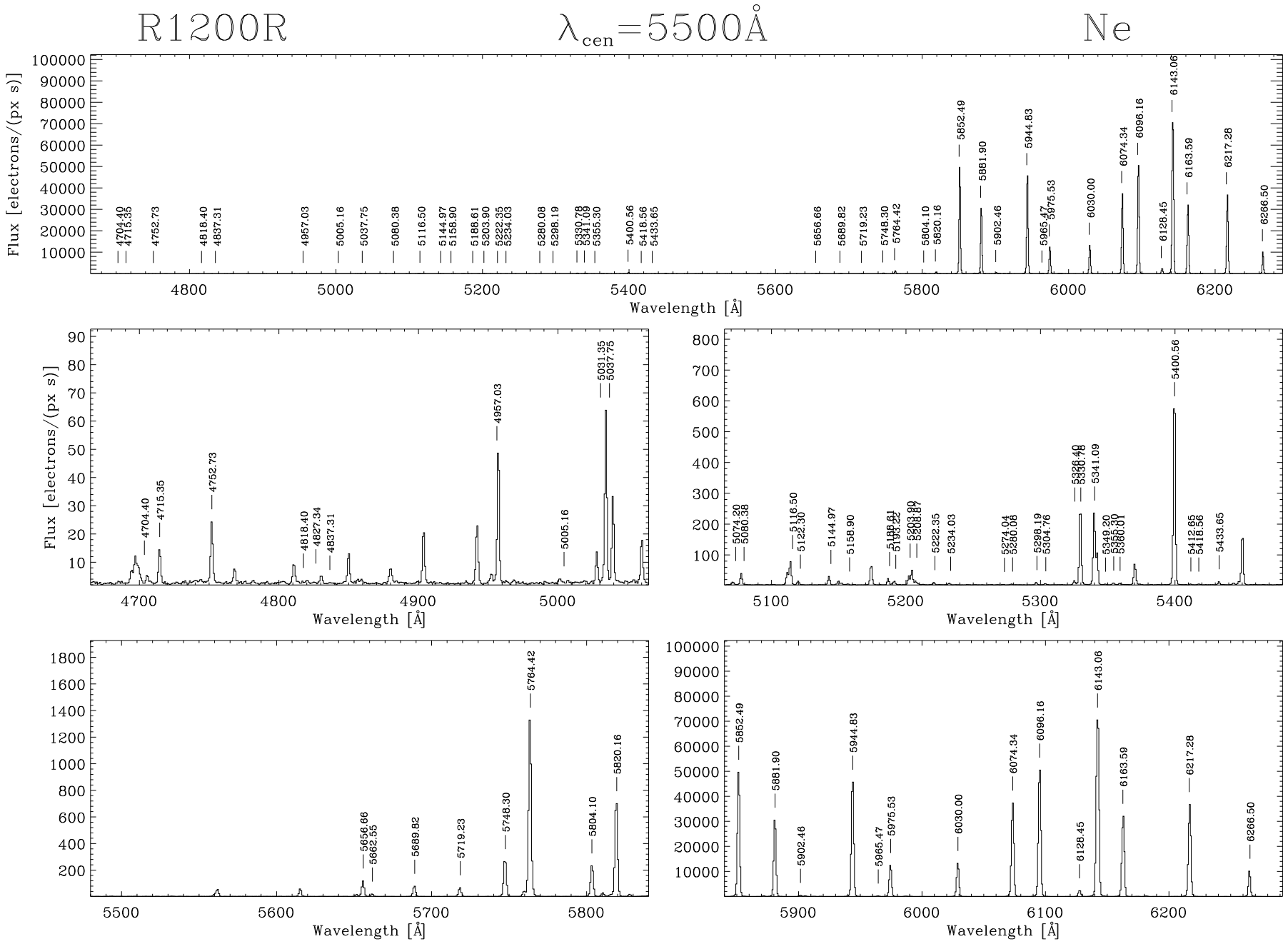

R1200B
λcen=5000Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
4219.76
4231.60
4275.11
4290.40
4379.50
4397.94
4413.20
4428.54
4537.70
4656.39
4704.40
4715.35
4752.73
4818.40
4837.31
4957.03
5005.16
5037.75
5080.38
5116.50
5144.97
5158.90
5188.61
5203.90
5222.35
5234.03
5280.08
5298.19
5330.78
5341.09
5360.01
5400.56
5418.56
5433.65
5656.66
5689.82
5719.23
5748.30
5764.42
4827.34
5031.35
5074.20
5122.30
5193.22
5208.87
5274.04
5304.76
5349.20
5412.65
5662.55

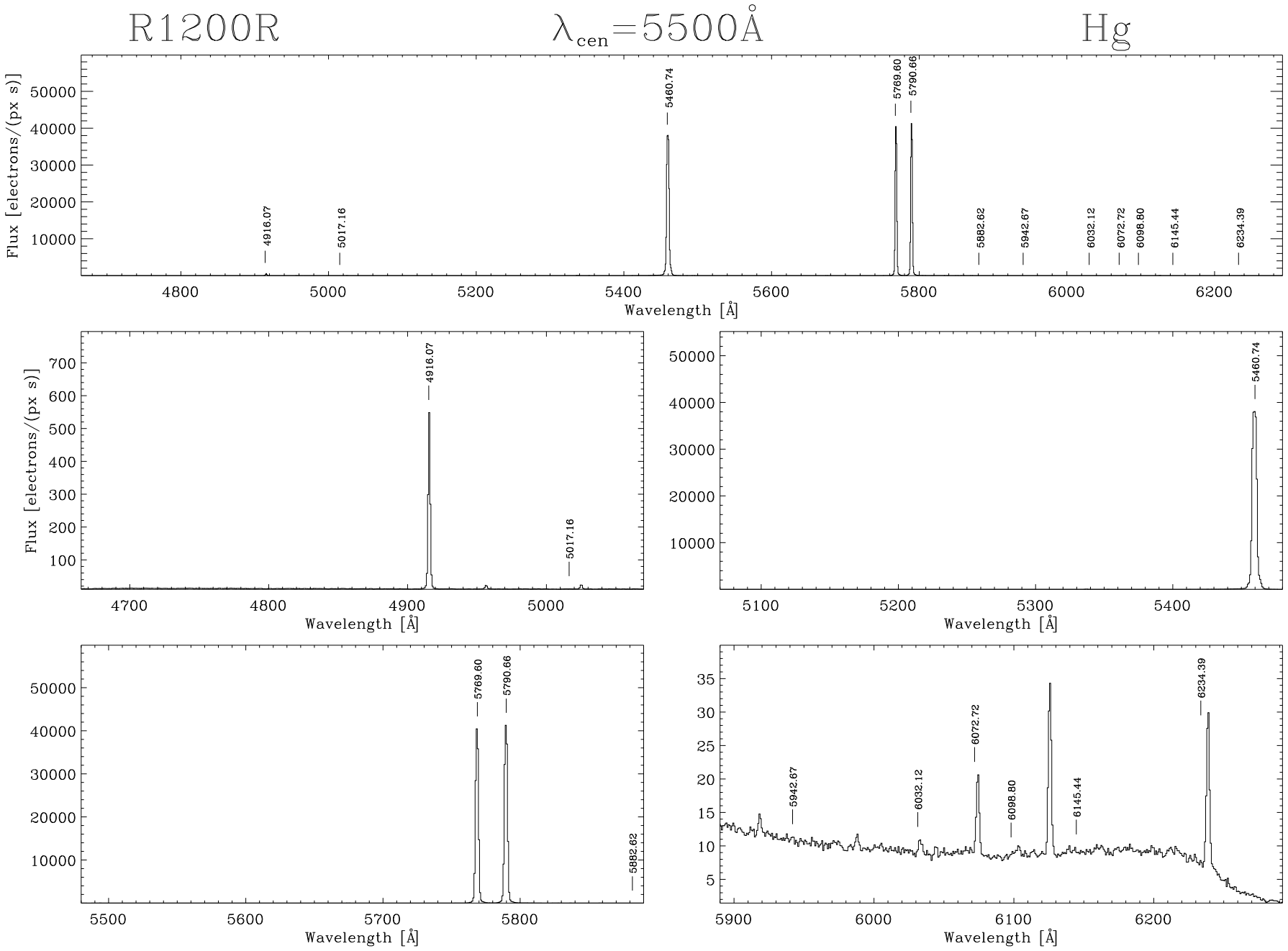

R1200B
λcen=5000Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
4348.06
4358.33
4916.07
5017.16
5460.74
5769.60
5790.66

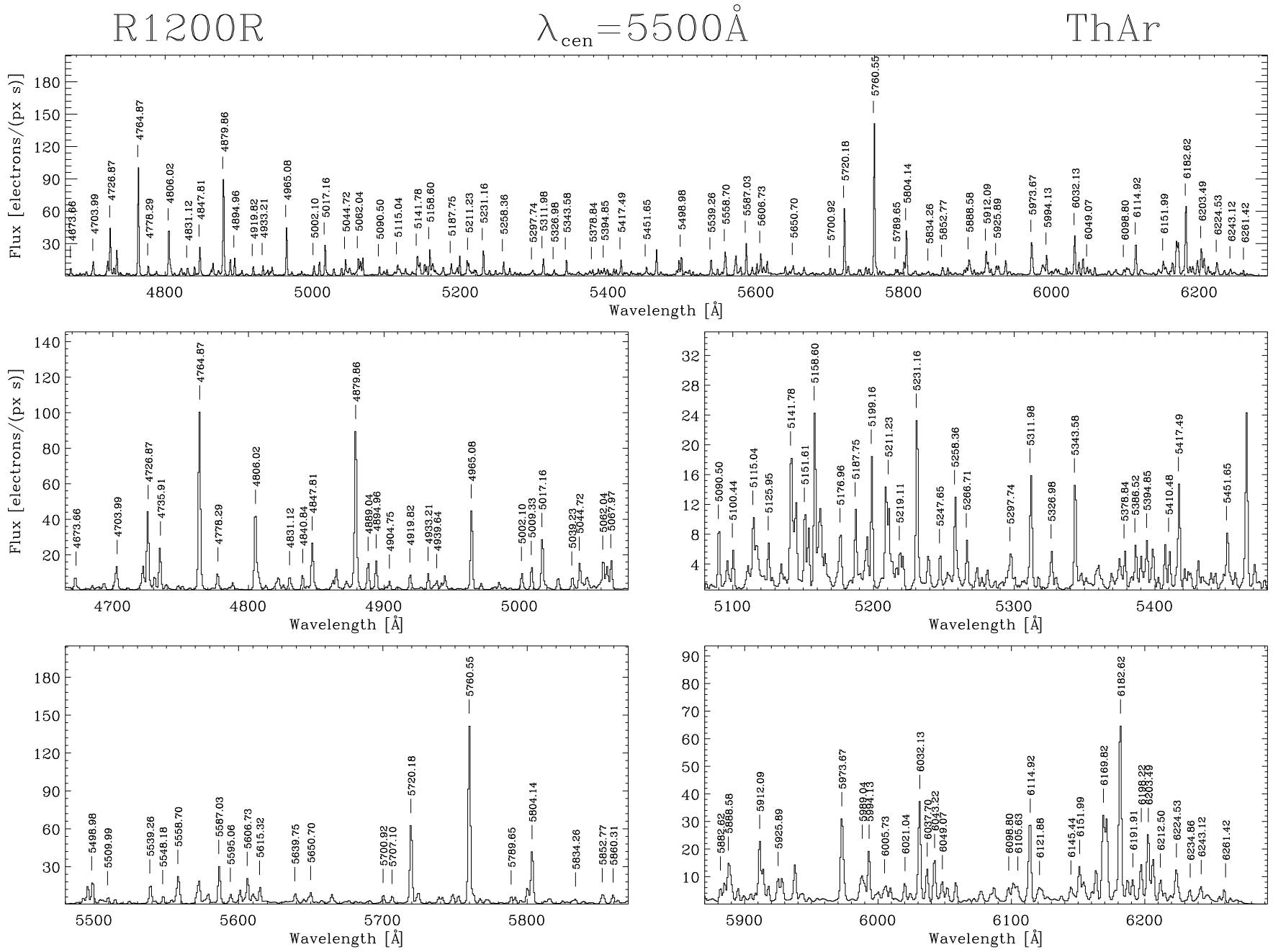

R1200B
λcen=5000Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
4200.67
4259.36
4277.53
4300.10
4331.20
4348.06
4370.75
4385.06
4400.99
4426.00
4481.81
4510.73
4545.05
4589.90
4609.57
4637.23
4657.90
4673.66
4703.99
4726.87
4764.87
4778.29
4806.02
4831.12
4847.81
4879.86
4894.96
4919.82
4933.21
4965.08
5002.10
5017.16
5044.72
5062.04
5090.50
5115.04
5141.78
5158.60
5187.75
5211.23
5231.16
5258.36
5297.74
5311.98
5326.98
5343.58
5378.84
5394.85
5417.49
5451.65
5495.87
5509.99
5539.26
5558.70
5587.03
5606.73
5650.70
5700.92
5720.18
5760.55
5789.65
4210.92
4266.29
4272.17
4309.24
4379.67
4385.06
4400.99
4408.88
4426.00
4430.19
4474.76
4493.33
4579.35
4589.90
4628.44
4637.23
4673.66
4703.99
4726.87
4735.91
4778.29
4806.02
4831.12
4840.84
4847.81
4879.86
4889.04
4894.96
4904.75
4919.82
4933.21
4939.64
4965.08
5002.10
5009.33
5017.16
5039.23
5044.72
5062.04
5067.97
5090.50
5100.44
5115.04
5125.95
5141.78
5151.61
5158.60
5176.96
5187.75
5199.16
5211.23
5219.11
5231.16
5247.65
5258.36
5266.71
5297.74
5311.98
5326.98
5343.58
5378.84
5386.52
5394.85
5407.34
5417.49
5451.65
5495.87
5509.99
5539.26
5548.18
5558.70
5587.03
5595.06
5606.73
5615.32
5639.75
5650.70
5700.92
5707.10
5720.18
5760.55
5789.65

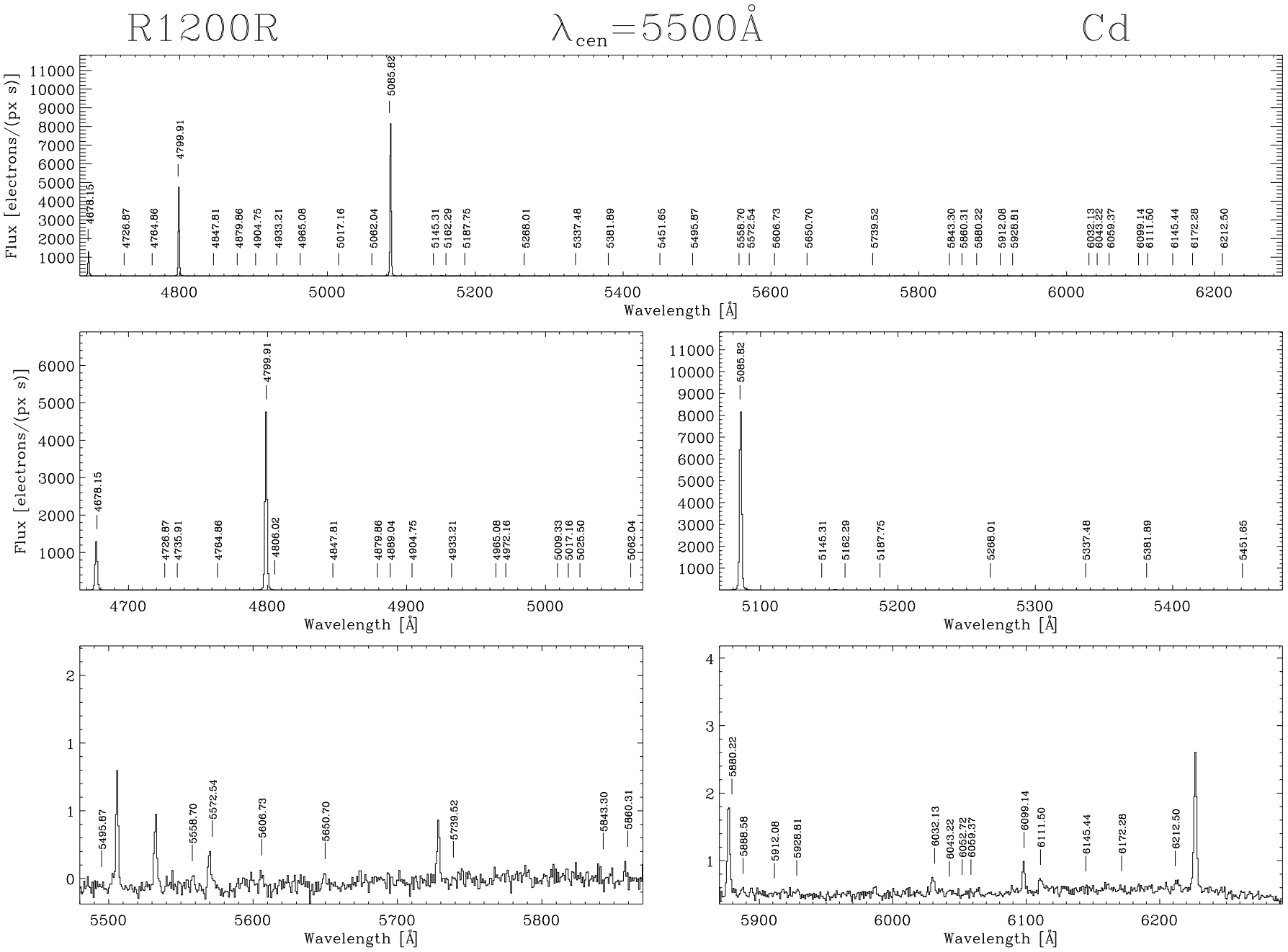

R1200B
λcen=5000Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
4200.67
4259.36
4272.17
4300.10
4333.56
4348.06
4379.88
4400.99
4415.63
4430.19
4481.81
4510.73
4545.05
4579.35
4589.90
4609.57
4657.90
4678.15
4726.87
4764.86
4799.91
4847.81
4879.86
4904.75
4933.21
4965.08
5017.16
5062.04
5085.82
5145.31
5162.29
5187.75
5268.01
5337.48
5378.13
5451.65
5495.87
5558.70
5572.54
5606.73
5650.70
5739.52

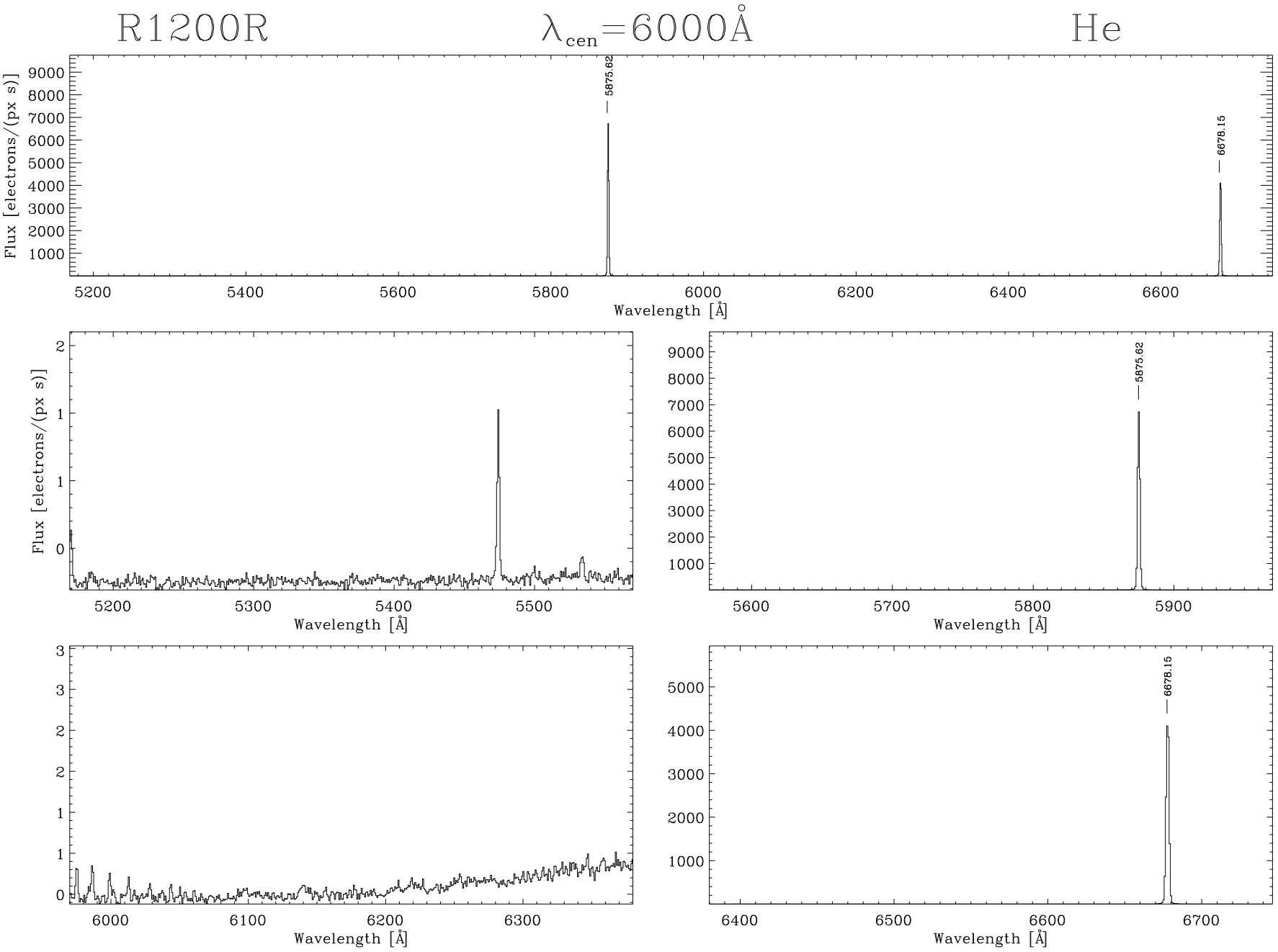

R1200R
λcen=5500Å
He
Flux [electrons/(px s)]
Wavelength [Å]
4713.15
4921.93
5015.68
5047.74
5875.62

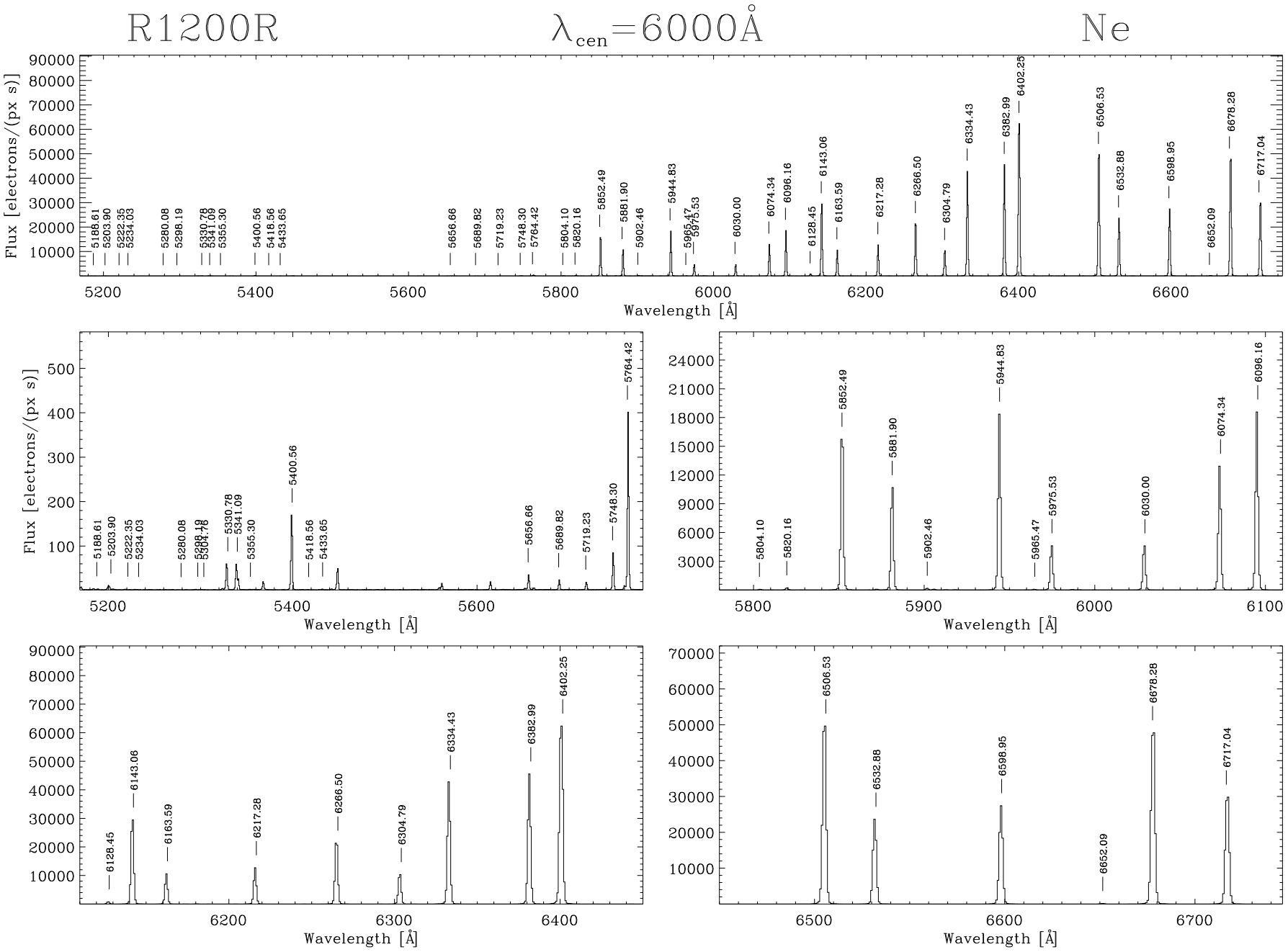

R1200R
λcen=5500Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
4704.40
4715.35
4752.73
4818.40
4837.31
4957.03
5005.16
5037.75
5080.38
5116.50
5144.97
5158.90
5188.61
5203.90
5222.35
5234.03
5280.08
5298.19
5330.78
5341.09
5355.30
5400.56
5418.56
5433.65
5656.66
5689.82
5719.23
5748.30
5764.42
5804.10
5820.16
5852.49
5881.90
5902.46
5944.83
5965.47
5975.53
6030.00
6074.34
6096.16
6128.45
6143.06
6163.59
6217.28
6266.50

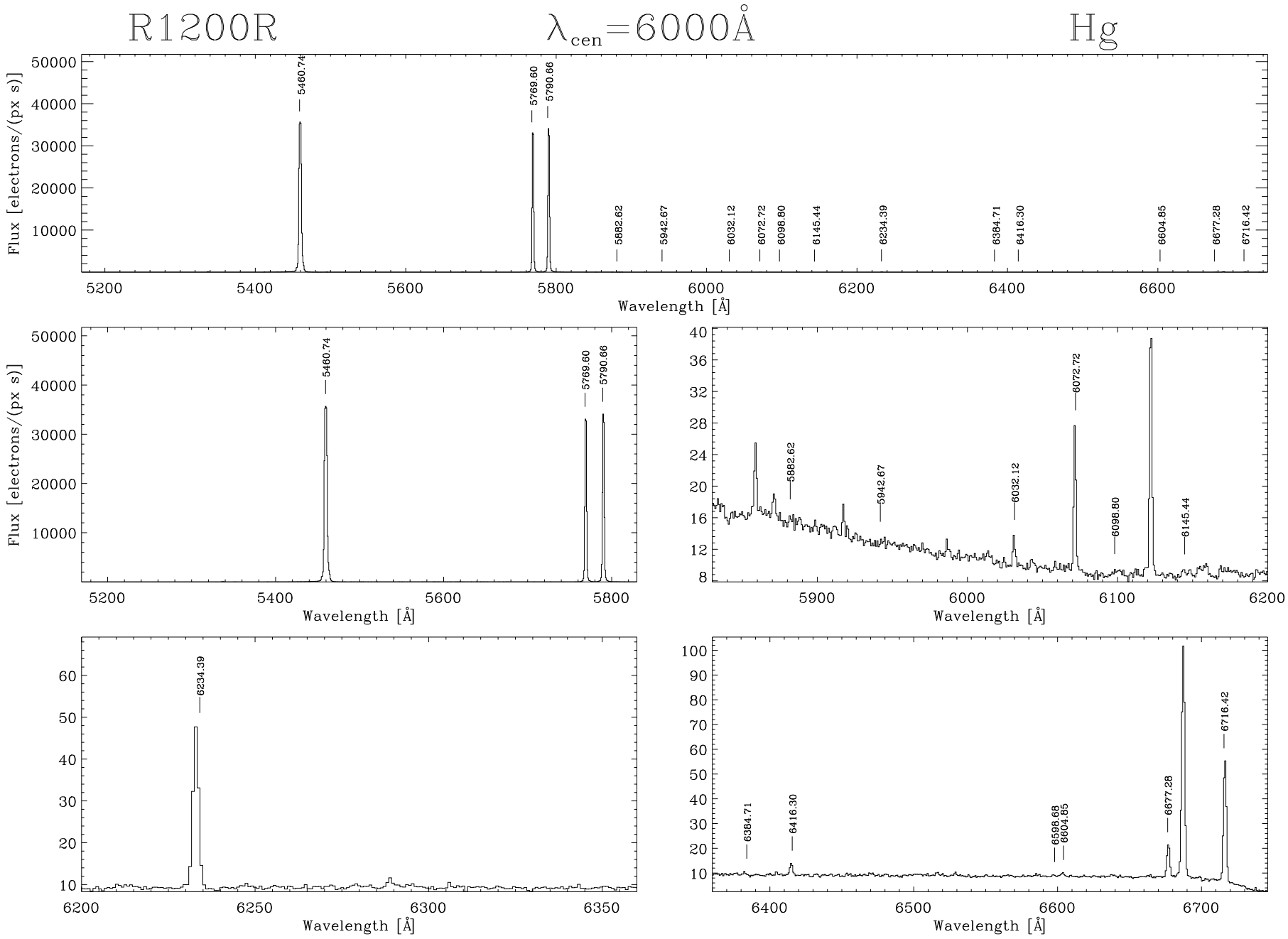

R1200R
λcen=5500Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
4916.07
5017.16
5460.74
5769.60
5790.66
5882.62
5942.67
6032.12
6072.72
6098.80
6145.44
6234.39

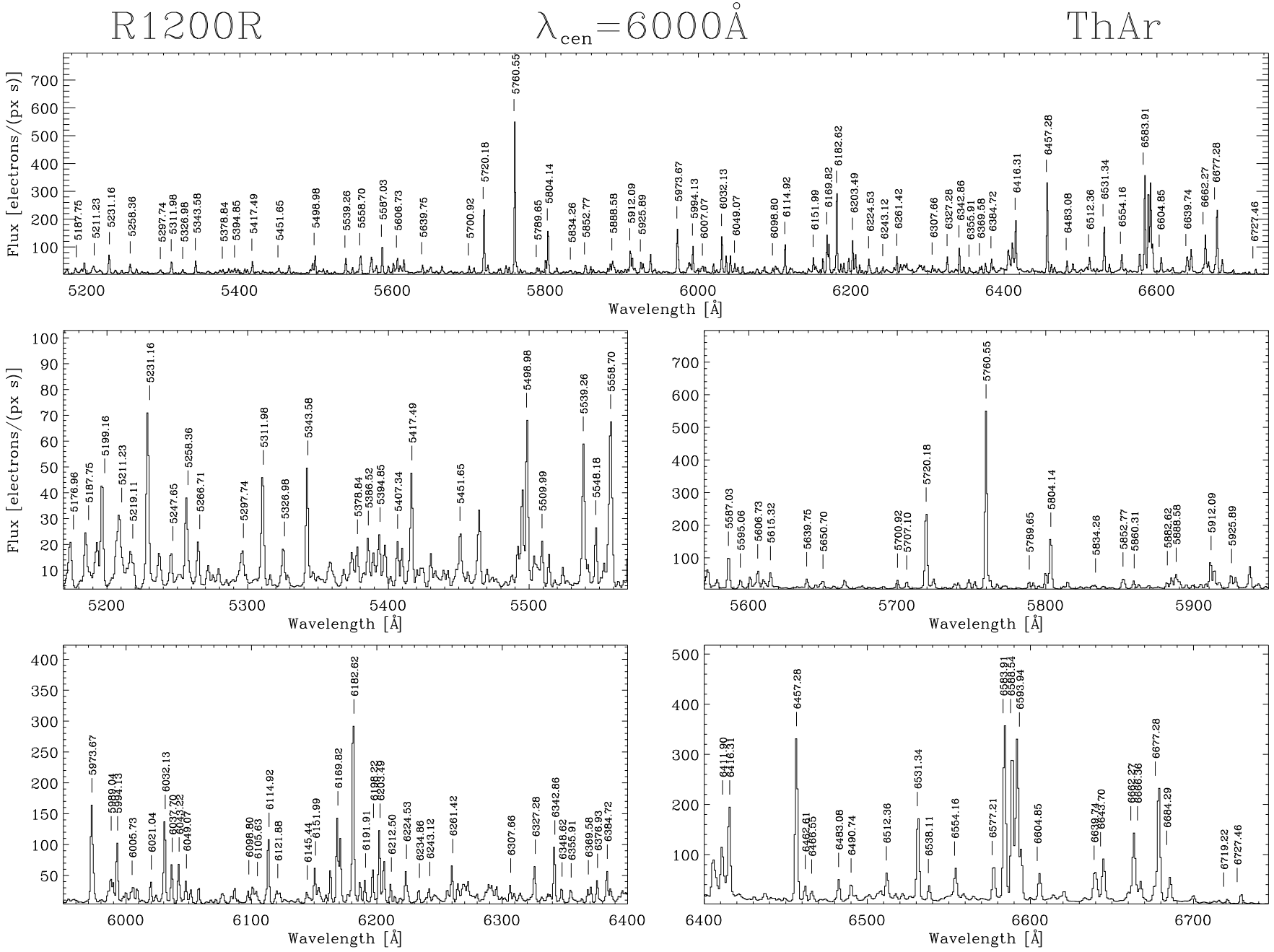

R1200R
λcen=5500Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
4673.66
4703.99
4726.87
4735.91
4764.87
4778.29
4806.02
4831.12
4840.84
4847.81
4879.86
4889.04
4894.96
4904.75
4919.82
4933.21
4939.64
4965.08
5002.10
5009.33
5017.16
5039.23
5044.72
5062.04
5067.97
5090.50
5100.44
5115.04
5125.95
5141.78
5151.61
5158.60
5176.96
5187.75
5199.16
5211.23
5219.11
5231.16
5247.65
5258.36
5286.71
5297.74
5311.98
5326.98
5343.58
5378.84
5386.52
5394.85
5410.48
5417.49
5451.65
5498.98
5509.99
5539.26
5548.18
5558.70
5587.03
5595.06
5606.73
5615.32
5639.75
5650.70
5700.92
5707.10
5720.18
5760.55
5789.65
5804.14
5834.26
5852.77
5860.31
5882.62
5888.58
5912.09
5925.89
5973.67
5989.04
5994.13
6005.73
6021.04
6032.13
6037.70
6043.22
6049.07
6098.80
6105.63
6114.92
6121.88
6145.44
6151.99
6169.82
6182.62
6191.91
6203.49
6212.50
6224.53
6234.86
6243.12
6261.42

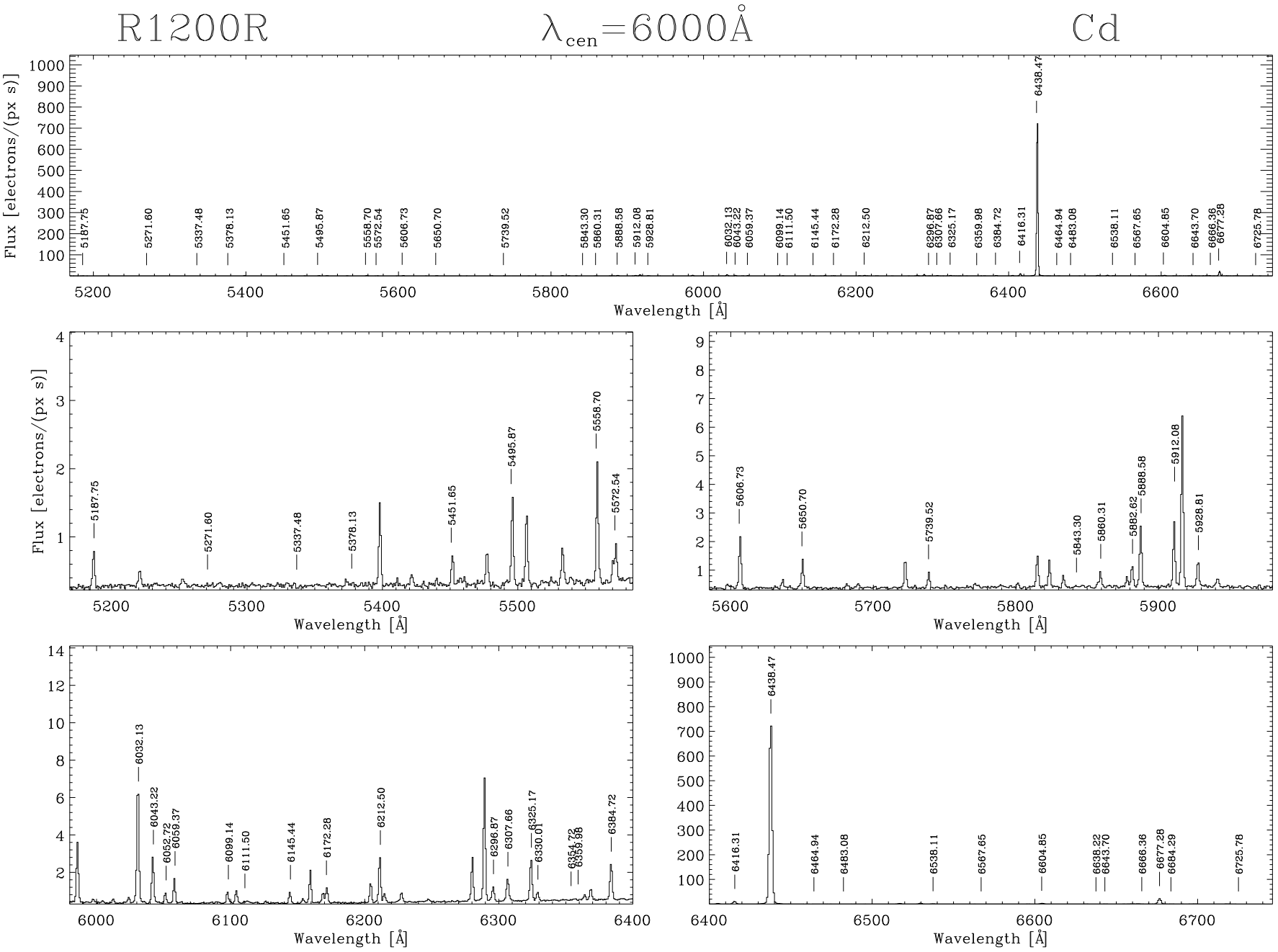

R1200R
λcen=5500Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
4678.15
4726.87
4764.86
4799.91
4847.81
4879.86
4904.75
4933.21
4965.08
5017.16
5062.04
5085.82
5145.31
5162.29
5187.75
5268.01
5337.48
5381.89
5451.65
5495.87
5558.70
5572.54
5606.73
5650.70
5739.52
5843.30
5860.31
5880.22
5912.08
5928.81
6032.13
6043.22
6059.37
6099.14
6111.50
6145.44
6172.28
6212.50
4735.91
4806.02
4889.04
4972.16
5009.33
5025.50
5888.58
6052.72

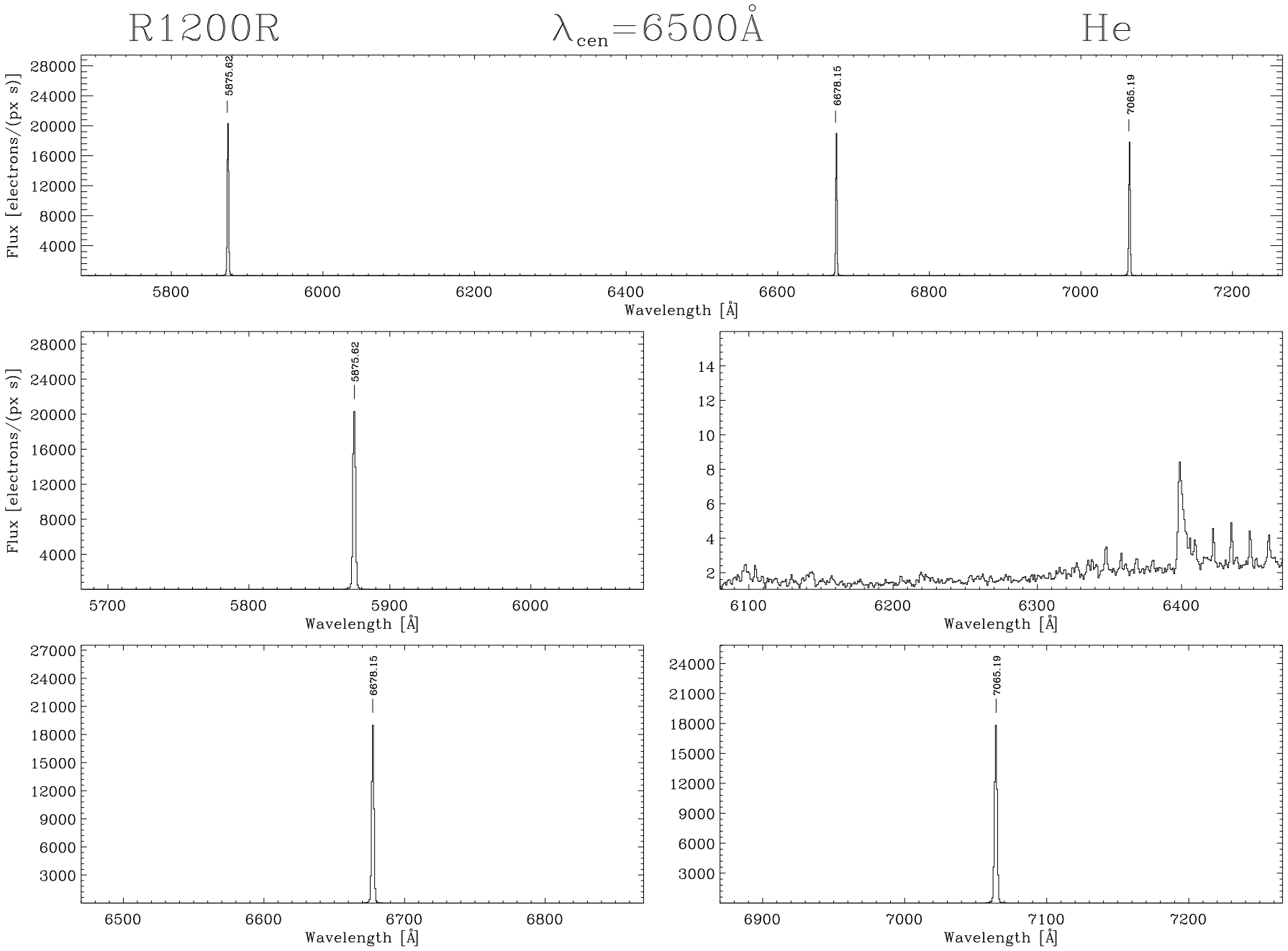

R1200R
λcen=6000Å
He
Flux [electrons/(px s)]
Wavelength [Å]
5875.62
6678.15

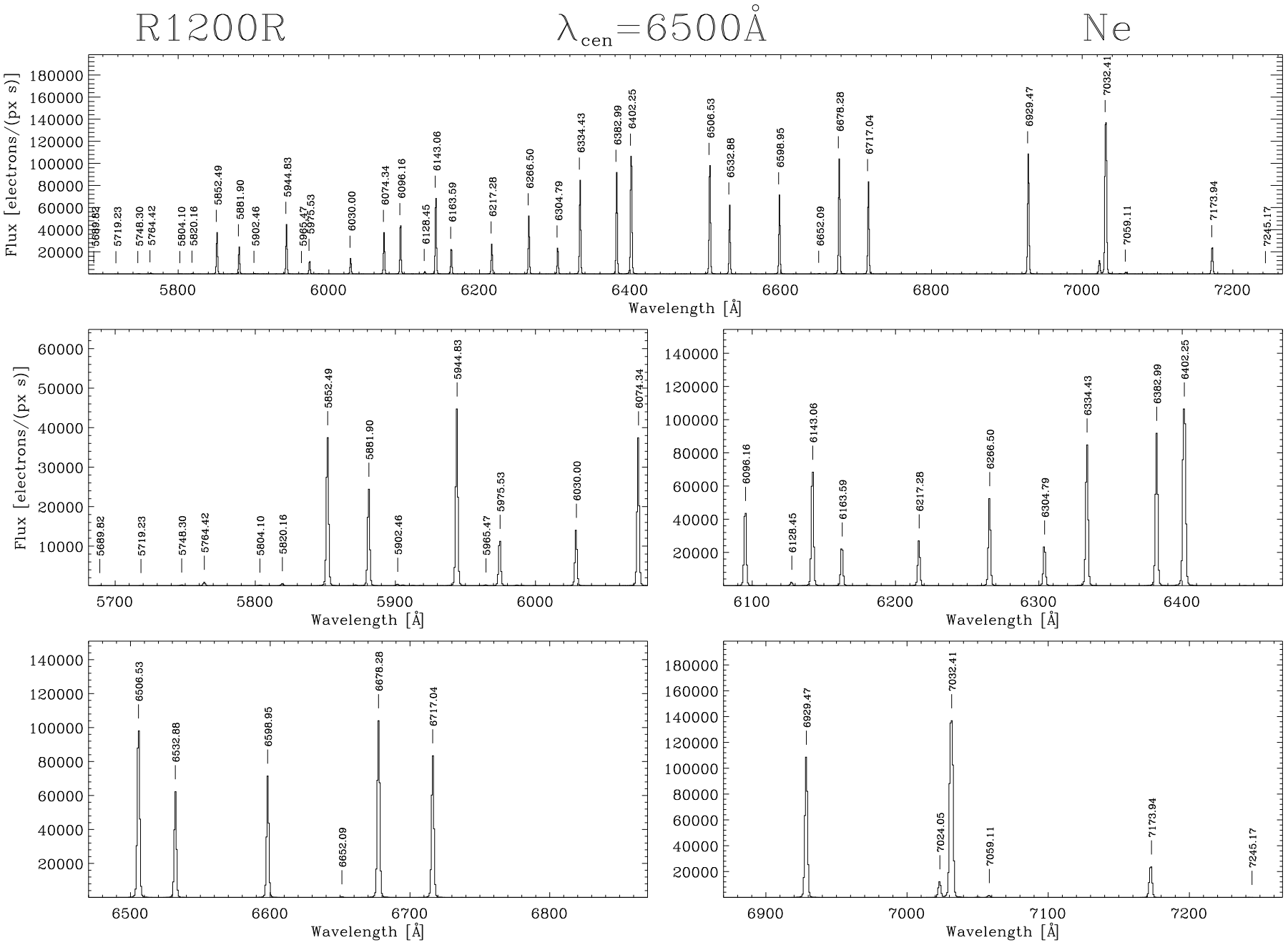

R1200R
λcen=6000Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
5188.61
5203.90
5222.35
5234.03
5280.08
5298.19
5330.78
5341.09
5355.30
5400.56
5418.56
5433.65
5656.66
5689.82
5719.23
5748.30
5764.42
5804.10
5820.16
5852.49
5881.90
5902.46
5944.83
5965.47
5975.53
6030.00
6074.34
6096.16
6128.45
6143.06
6163.59
6217.28
6266.50
6286.50
6304.79
6334.43
6382.99
6402.25
6506.53
6532.88
6598.95
6652.09
6678.28
6717.04

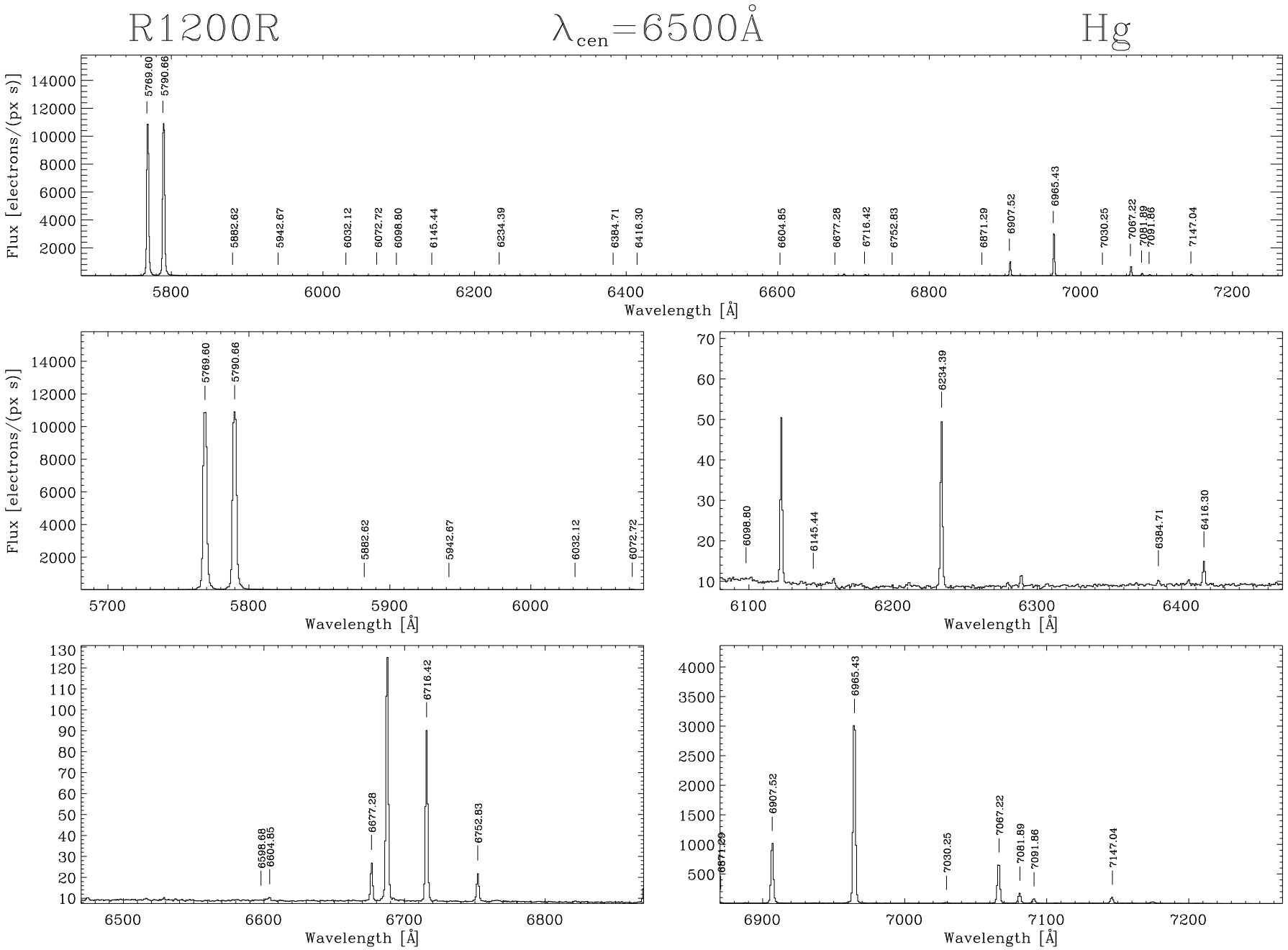

R1200R
λcen=6000Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
5460.74
5769.60
5790.66
5882.62
5942.67
6032.12
6072.72
6098.80
6145.44
6234.39
6384.71
6416.30
6604.85
6677.28
6716.42
6598.68

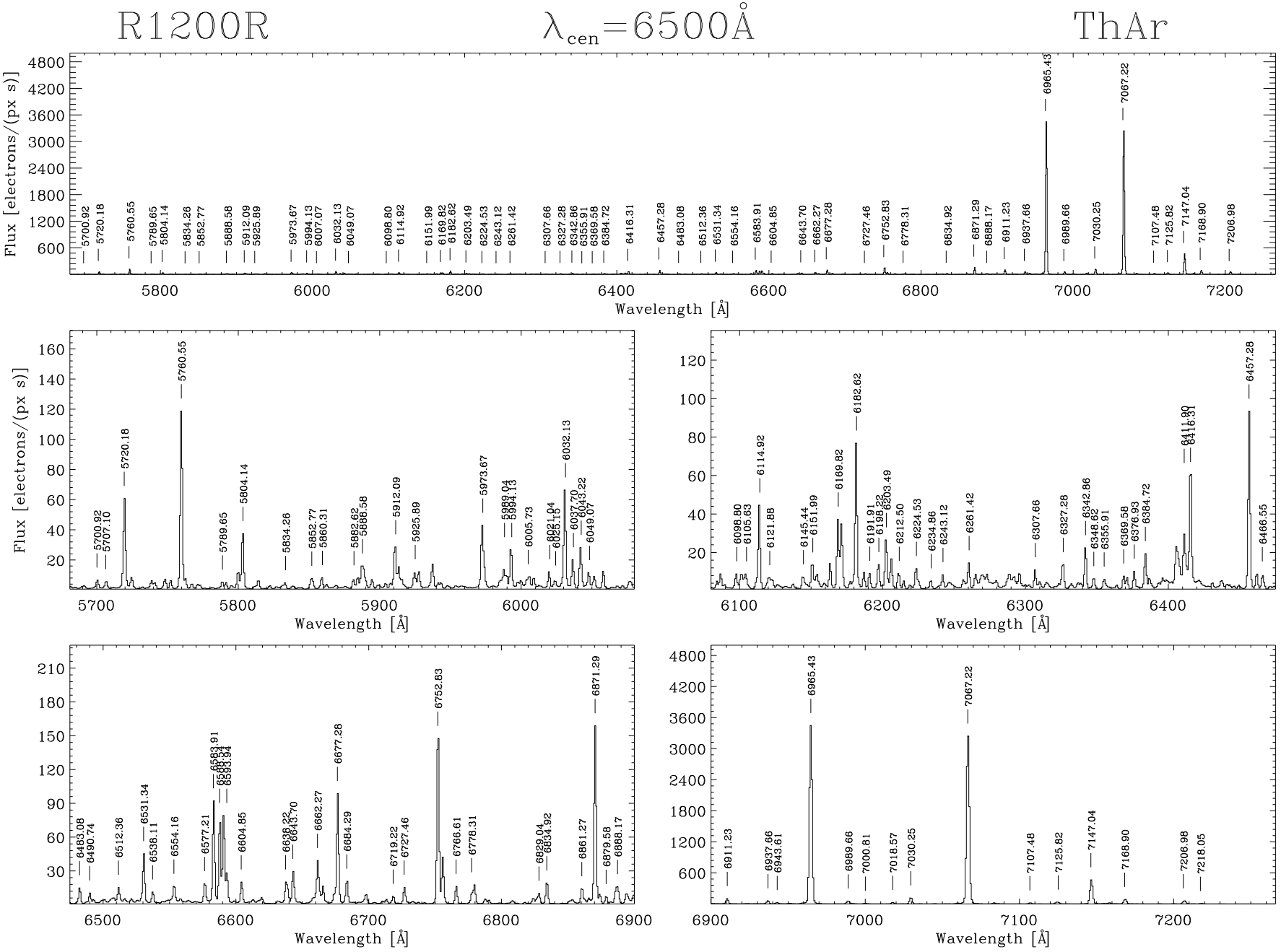

R1200R
λcen=6000Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
5187.75
5211.23
5231.16
5258.36
5297.74
5311.98
5326.98
5343.58
5378.84
5394.85
5417.49
5451.65
5498.98
5539.26
5558.70
5587.03
5606.73
5639.75
5700.92
5720.18
5760.55
5789.65
5804.14
5834.26
5852.77
5888.58
5912.09
5925.89
5973.67
5994.13
6007.07
6032.13
6049.07
6098.80
6114.92
6151.99
6169.82
6182.62
6203.49
6224.53
6243.12
6261.42
6307.66
6327.28
6342.86
6355.91
6369.58
6384.72
6416.31
6457.28
6483.08
6512.36
6531.34
6554.16
6583.91
6604.85
6639.74
6662.27
6677.28
6727.46
5176.96
5199.16
5219.11
5247.65
5266.71
5386.52
5407.34
5509.99
5548.18
5595.06
5606.73
5615.32
5650.70
5707.10
5860.31
5882.62
5886.58
6005.73
6021.04
6037.70
6043.22
6105.63
6121.88
6145.44
6188.22
6191.91
6203.49
6212.50
6234.86
6348.62
6376.93
6411.80
6462.61
6468.53
6490.74
6538.11
6577.21
6583.91
6588.54
6593.94
6643.70
6684.29
6719.22

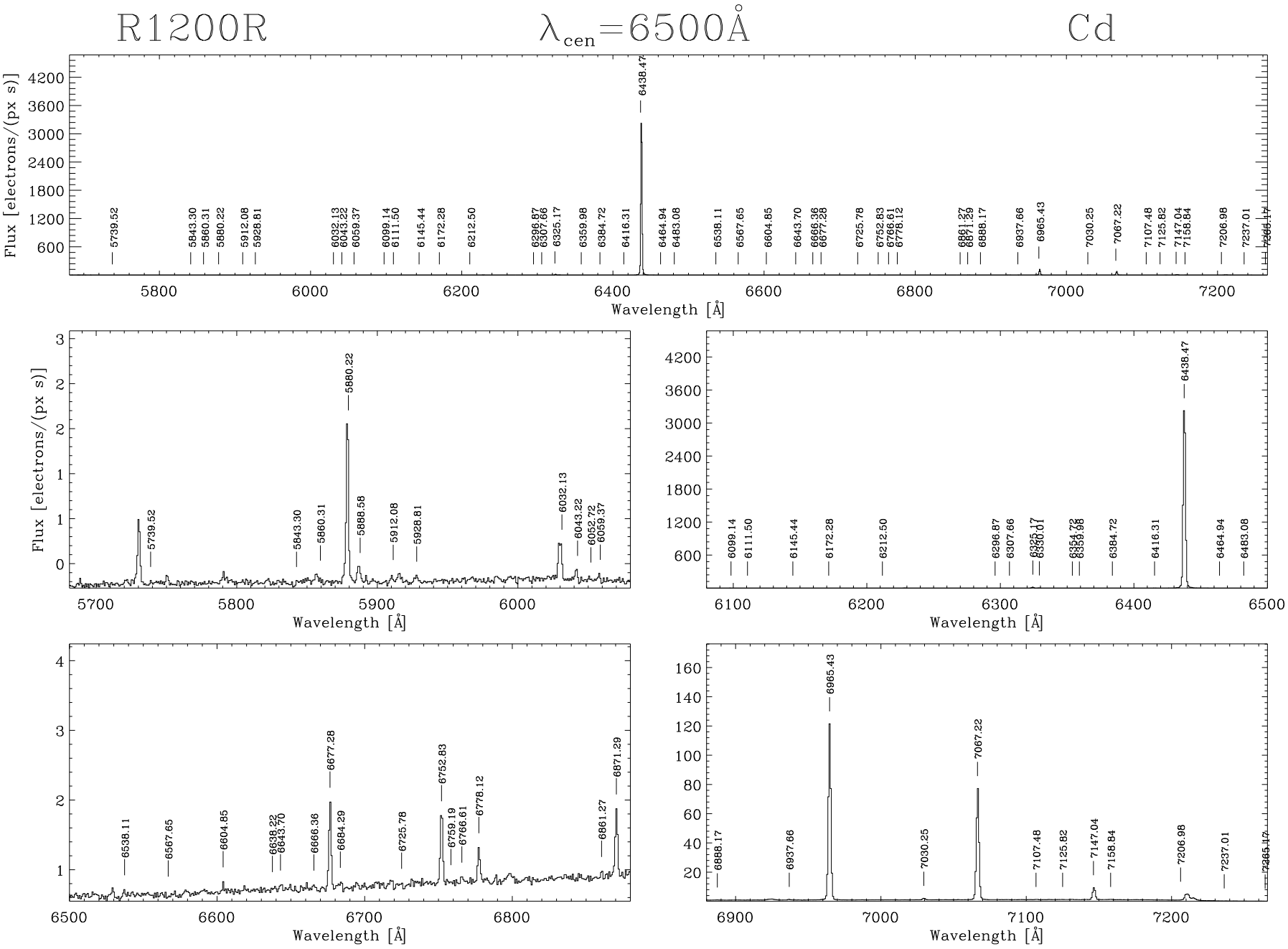

R1200R
λcen=6000Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
5187.75
5271.60
5337.48
5378.13
5451.65
5495.87
5558.70
5572.54
5606.73
5650.70
5739.52
5843.30
5860.31
5888.58
5912.08
5928.81
6032.13
6043.22
6059.37
6099.14
6111.50
6145.44
6172.28
6212.50
6296.87
6307.66
6325.17
6359.98
6384.72
6416.31
6438.47
6464.94
6483.08
6538.11
6567.65
6604.85
6643.70
6666.36
6677.28
6725.78

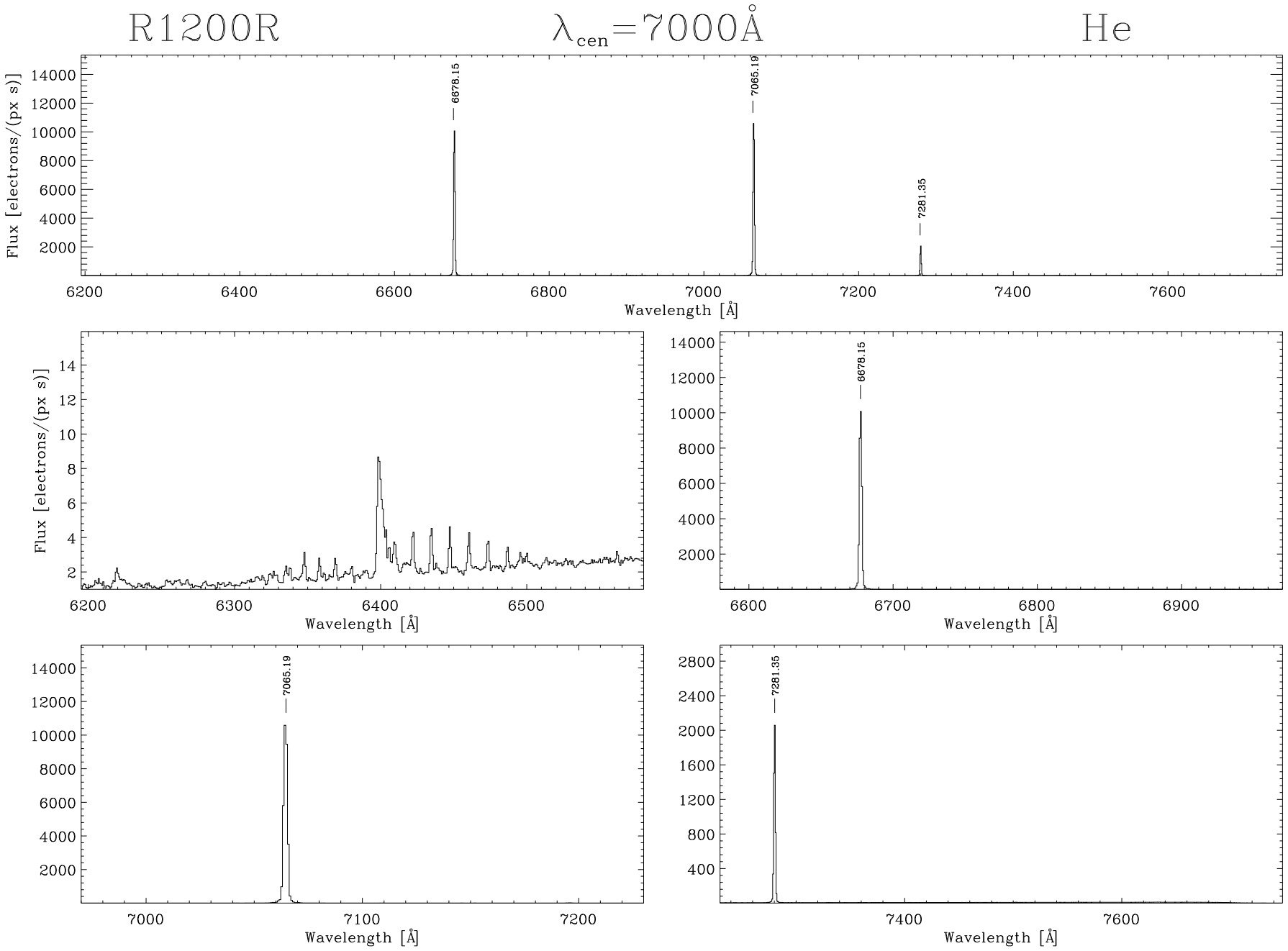

R1200R
λcen=6500Å
He
Flux [electrons/(px s)]
Wavelength [Å]
5875.62
6678.15
7065.19

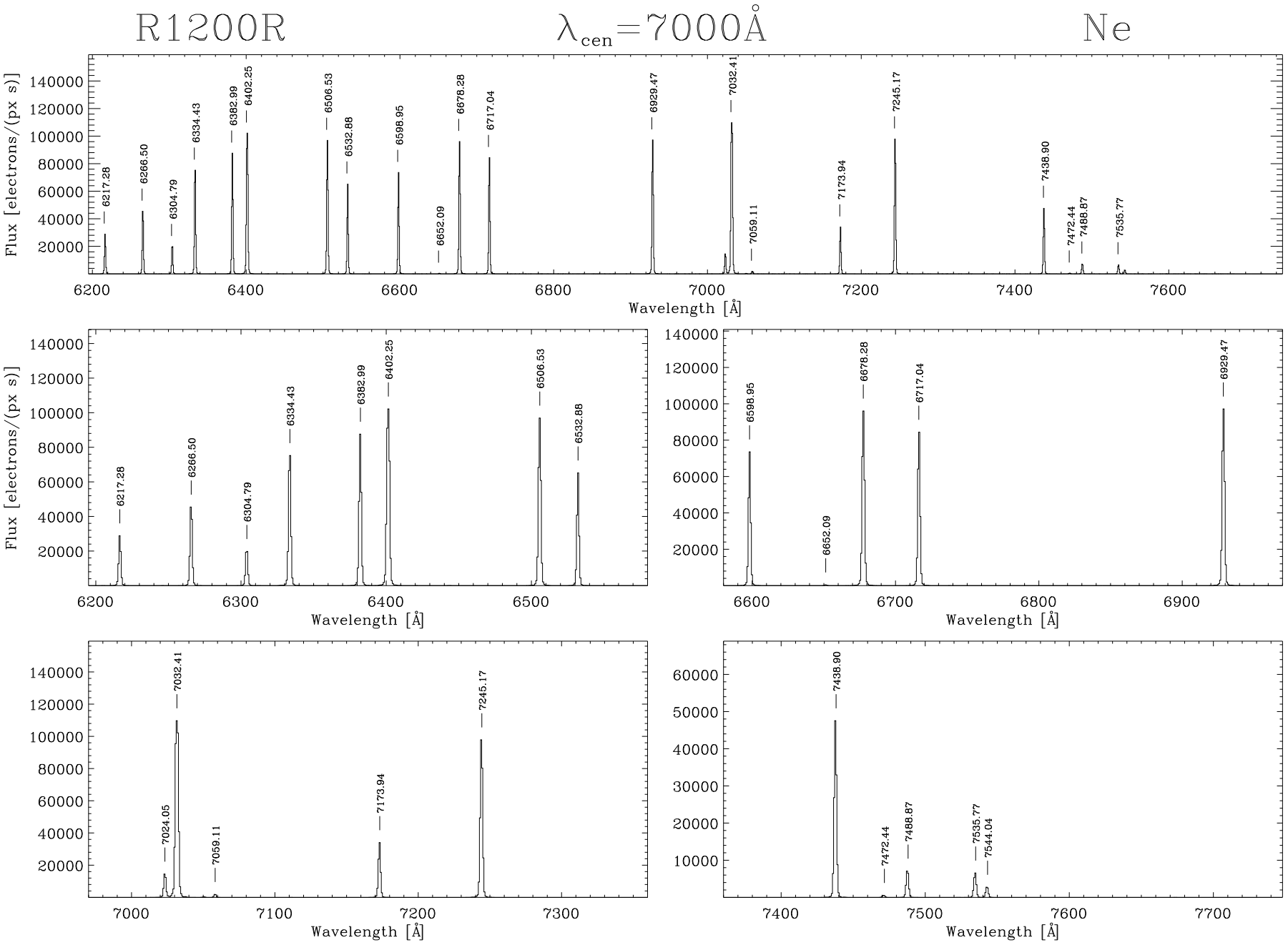

R1200R
λcen=6500Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
5689.82
5719.23
5748.30
5764.42
5804.10
5820.16
5852.49
5881.90
5902.46
5944.83
5965.47
5975.53
6030.00
6074.34
6096.16
6128.45
6143.06
6163.59
6217.28
6266.50
6304.79
6334.43
6382.99
6402.25
6506.53
6532.88
6598.95
6652.09
6678.28
6717.04
6929.47
7032.41
7059.11
7173.94
7245.17
7024.05

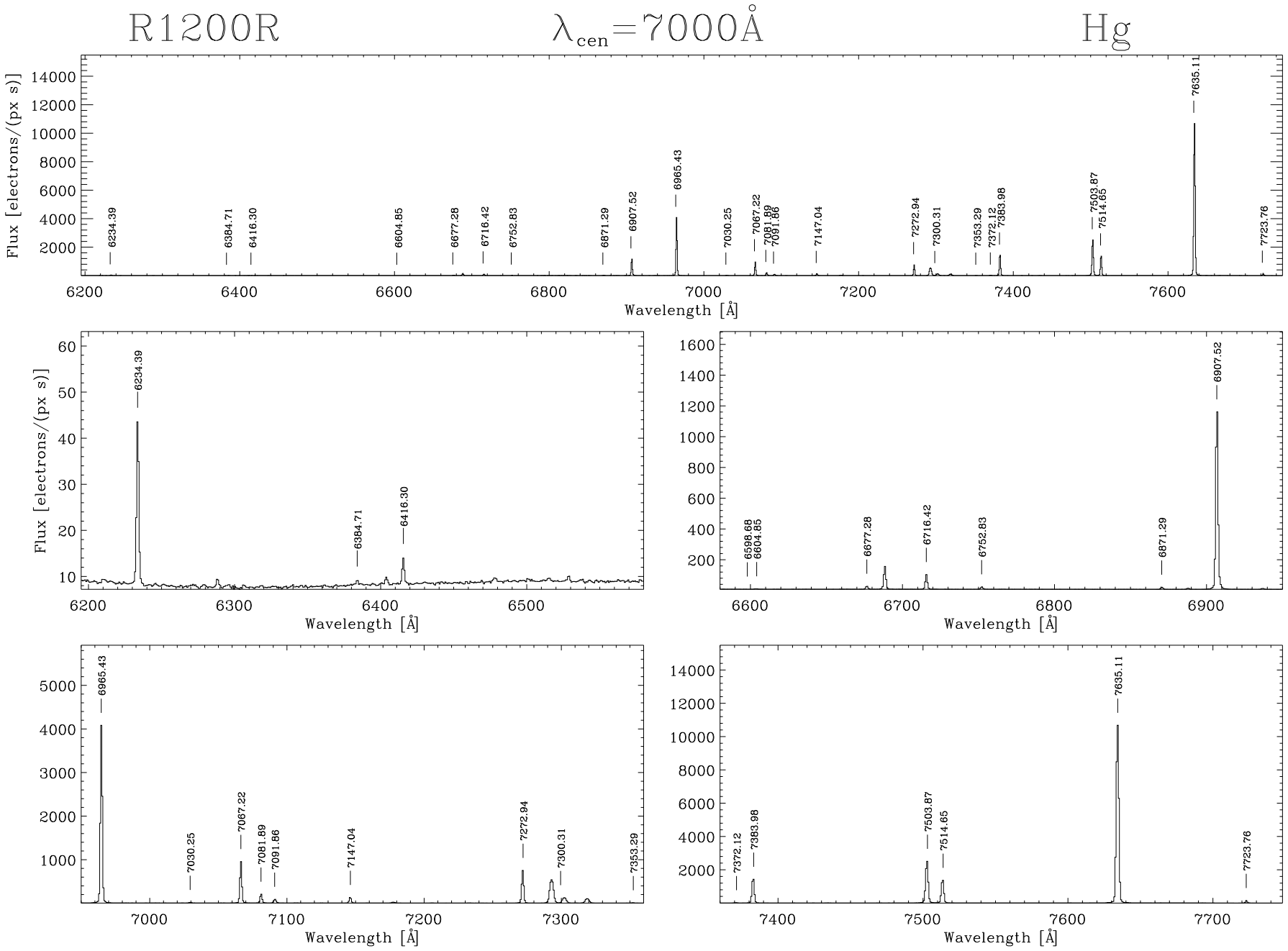

R1200R
λcen=6500Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
5769.60
5790.66
5882.62
5942.67
6032.12
6072.72
6098.80
6145.44
6234.39
6384.71
6416.30
6604.85
6677.28
6716.42
6752.83
6871.29
6907.52
6965.43
7030.25
7067.22
7081.89
7091.86
7147.04

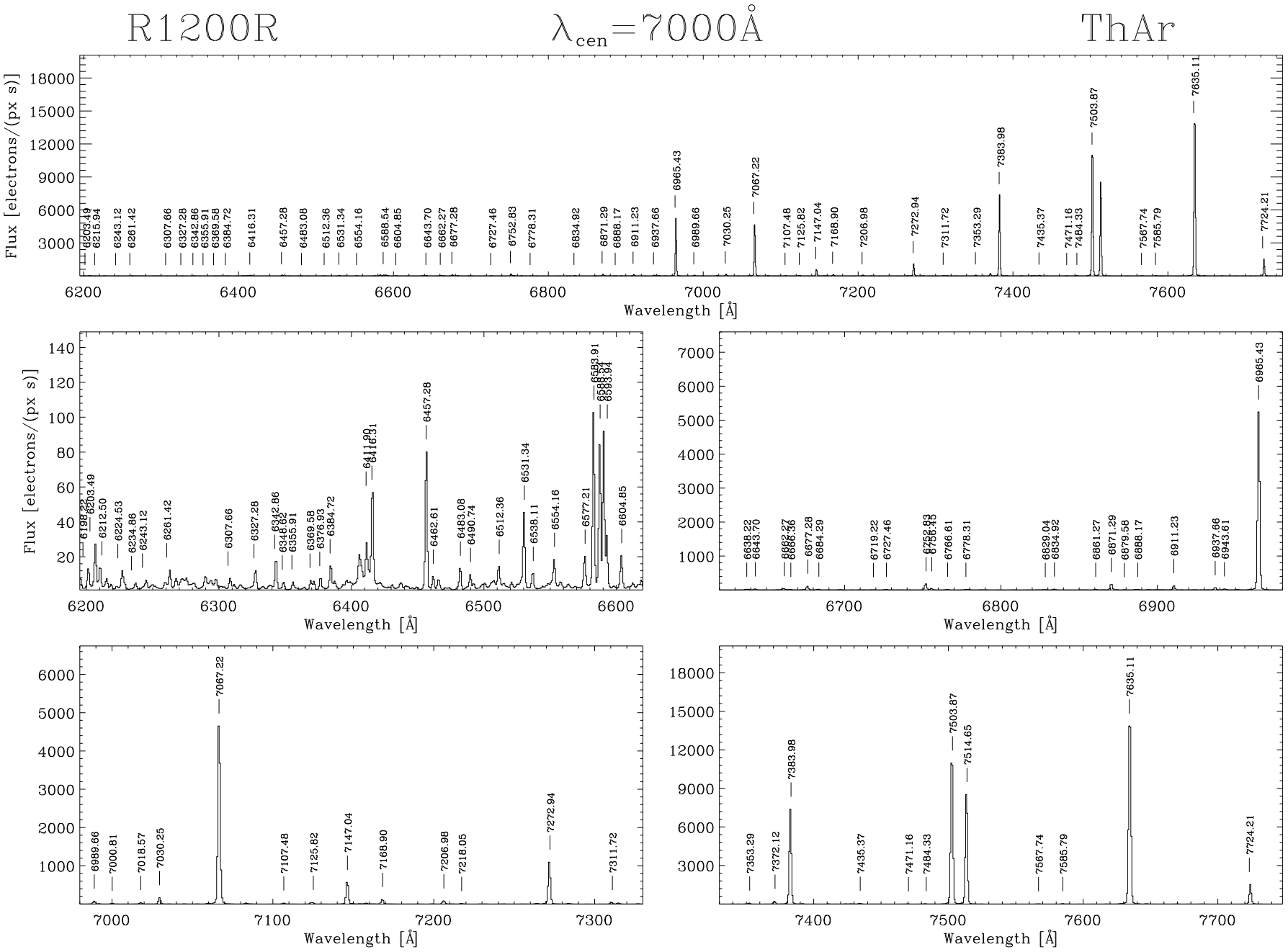

R1200R
λcen=6500Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
5700.92
5720.18
5760.55
5789.65
5804.14
5834.26
5852.77
5888.58
5912.09
5925.89
5973.67
5994.13
6007.07
6032.13
6049.07
6098.80
6114.92
6151.99
6169.82
6182.62
6203.49
6224.53
6243.12
6261.42
6307.66
6327.28
6342.86
6355.91
6369.58
6384.72
6416.31
6457.28
6483.08
6512.36
6531.34
6554.16
6583.91
6604.85
6643.70
6662.27
6677.28
6727.46
6752.83
6778.31
6834.92
6871.29
6888.17
6911.23
6937.66
6965.43
6989.66
7030.25
7067.22
7107.48
7125.82
7147.04
7168.90
7206.98
6098.80
6105.63
6114.92
6121.88
6145.44
6151.99
6169.82
6182.62
6191.91
6198.22
6203.49
6212.50
6224.53
6234.86
6243.12
6261.42
6307.66
6327.28
6342.86
6348.62
6355.91
6369.58
6376.93
6384.72
6411.80
6416.31
6457.28
6466.55
6483.08
6490.74
6512.36
6531.34
6538.11
6554.16
6577.21
6583.91
6588.64
6593.94
6604.85
6638.22
6643.70
6662.27
6677.28
6684.29
6719.22
6727.46
6752.83
6766.61
6778.31
6829.04
6834.92
6861.27
6871.29
6879.58
6888.17
6911.23
6937.66
6943.61
6965.43
6989.66
7000.81
7018.57
7030.25
7067.22
7107.48
7125.82
7147.04
7168.90
7206.98
7218.05

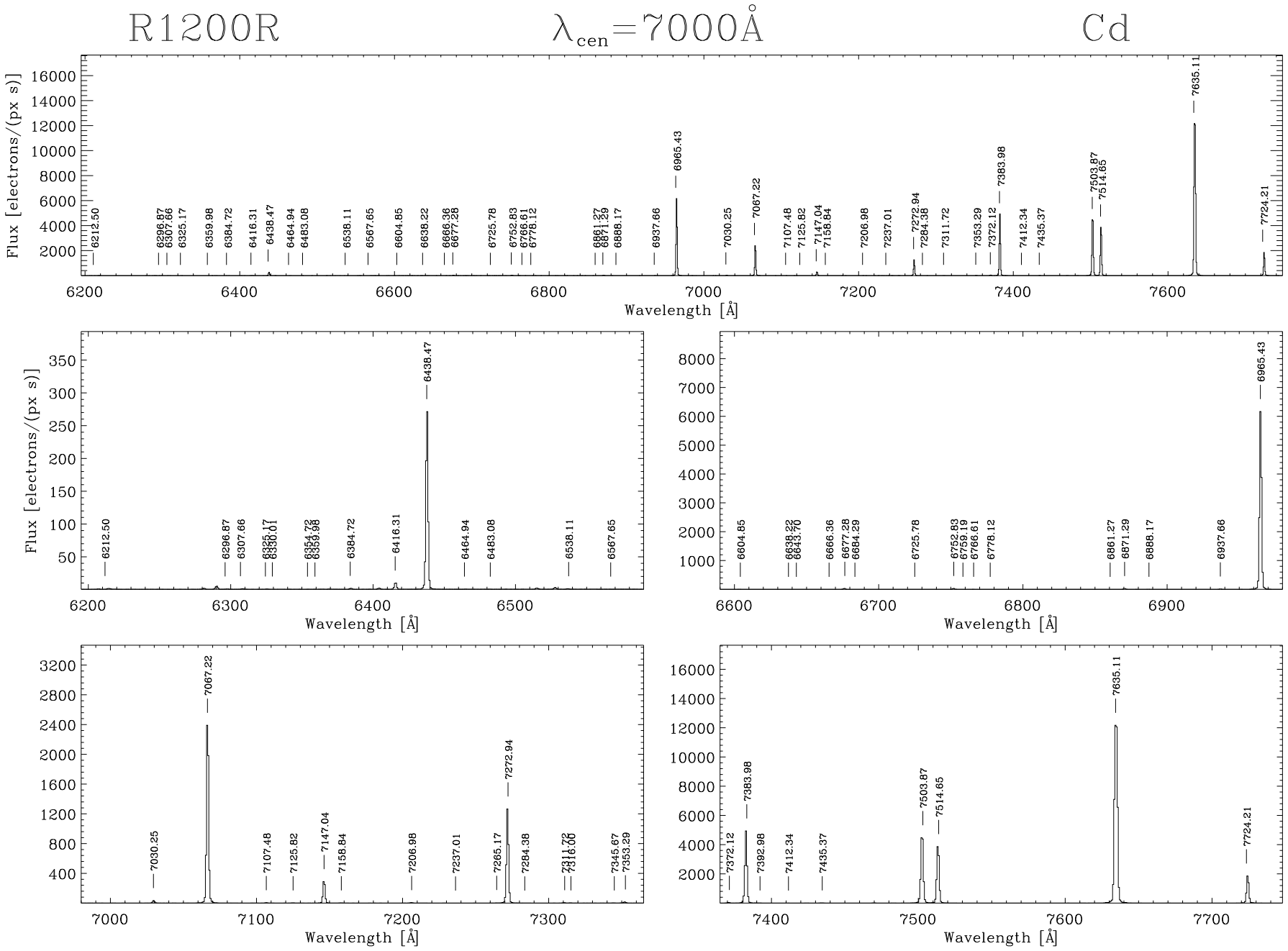

R1200R
λcen=6500Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
5739.52
5843.30
5860.31
5880.22
5912.08
5928.81
6032.13
6043.22
6059.37
6099.14
6111.50
6145.44
6172.28
6212.50
6296.87
6307.66
6325.17
6359.98
6384.72
6416.31
6438.47
6464.94
6483.08
6538.11
6567.65
6604.85
6643.70
6666.36
6677.28
6725.78
6752.83
6766.61
6778.12
6861.27
6871.29
6888.17
6937.66
6965.43
7030.25
7067.22
7107.48
7125.82
7147.04
7158.84
7206.98
7237.01
6052.72
6638.22
6759.19

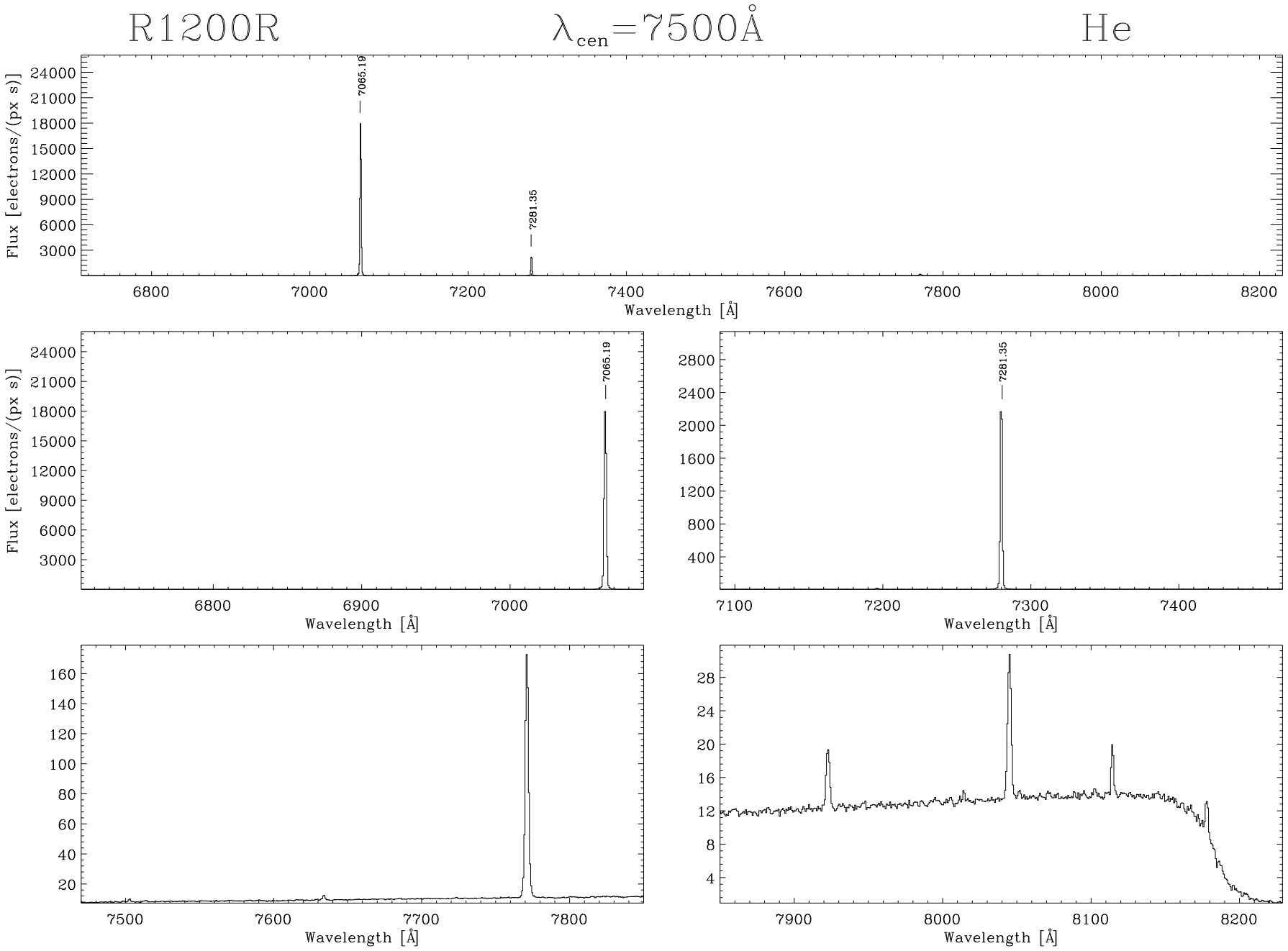

R1200R
λcen=7000Å
He
Flux [electrons/(px s)]
Wavelength [Å]
6678.15
7065.19
7281.35

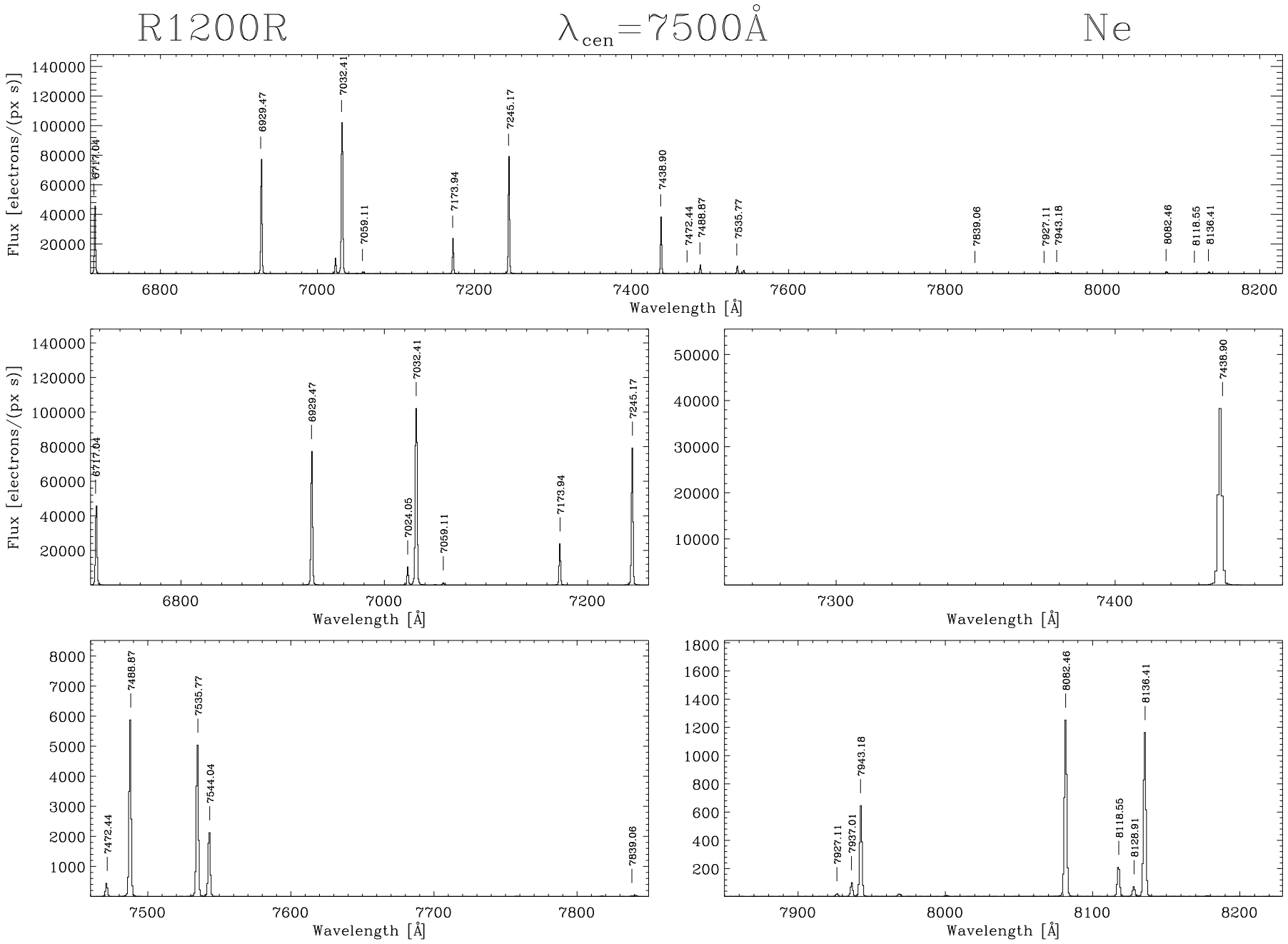

R1200R
λcen=7000Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
6217.28
6266.50
6304.79
6334.43
6382.99
6402.25
6506.53
6532.88
6598.95
6652.09
6678.28
6717.04
6929.47
7032.41
7059.11
7173.94
7245.17
7438.90
7472.44
7488.87
7535.77
7024.05
7544.04

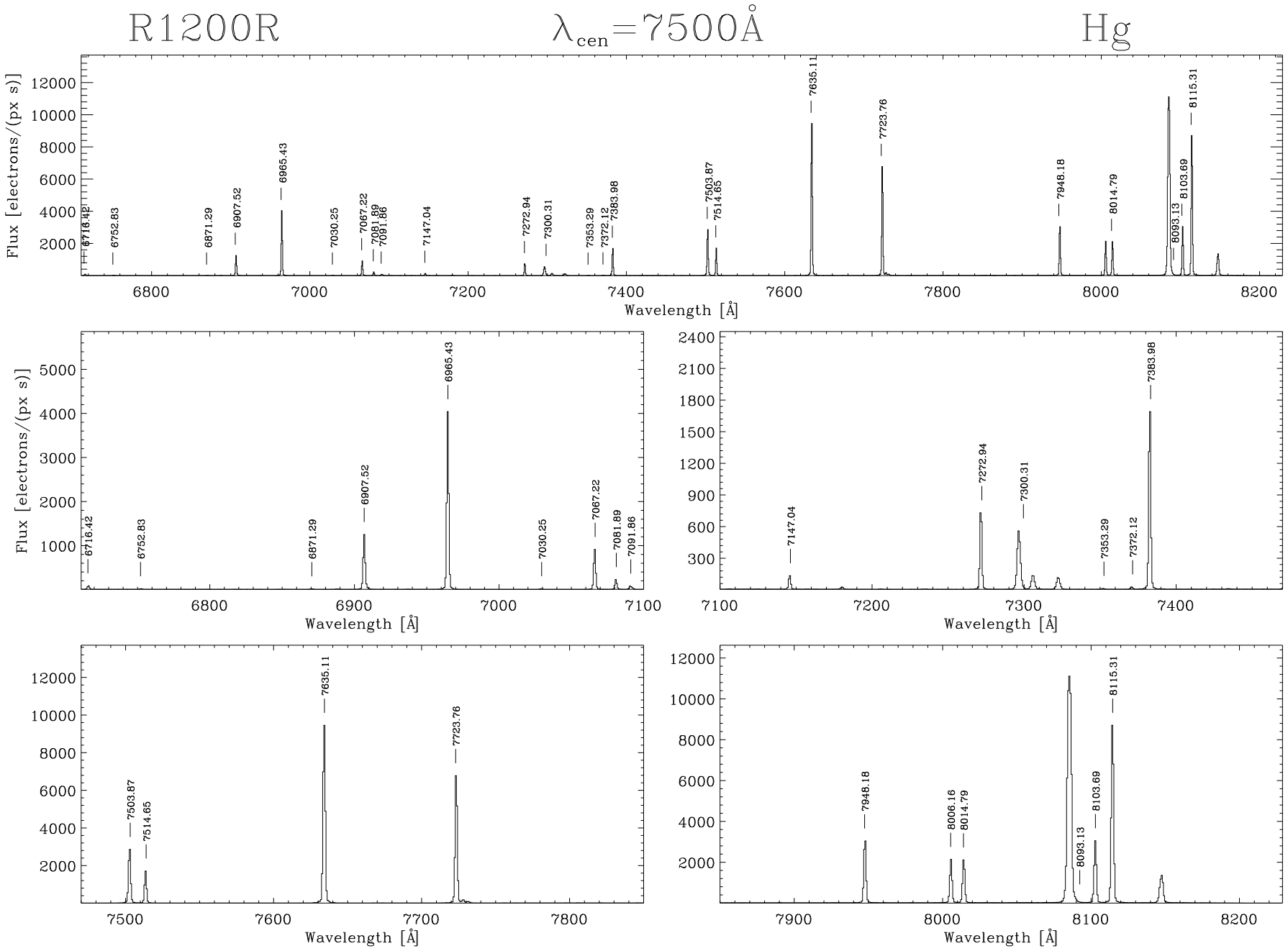

R1200R
λcen=7000Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
6234.39
6384.71
6416.30
6604.85
6677.28
6716.42
6752.83
6871.29
6907.52
6965.43
7030.25
7067.22
7081.89
7091.86
7147.04
7272.94
7300.31
7353.29
7372.12
7383.98
7503.87
7514.65
7635.11
7723.76
6598.68

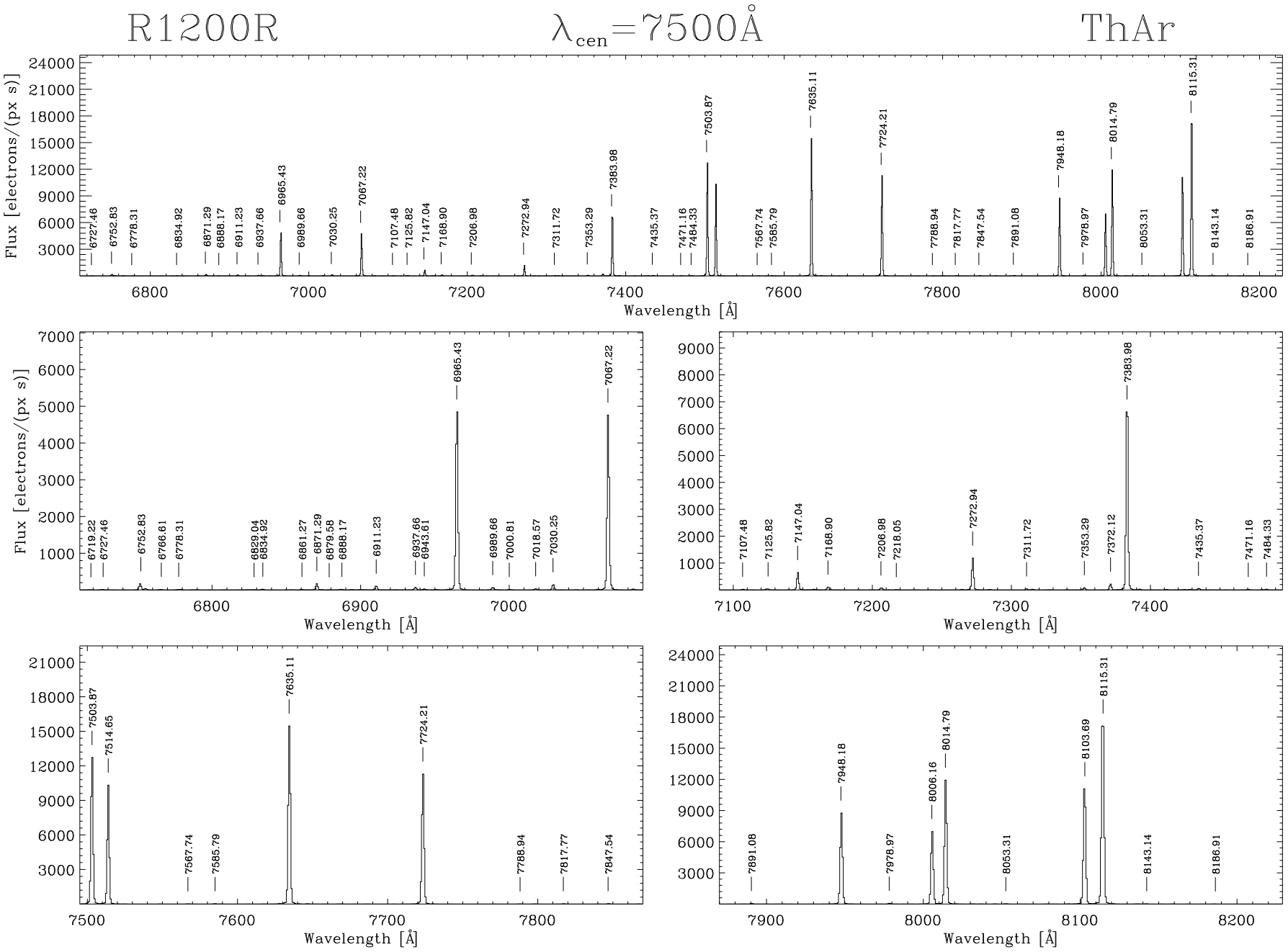

R1200R
λcen=7000Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]

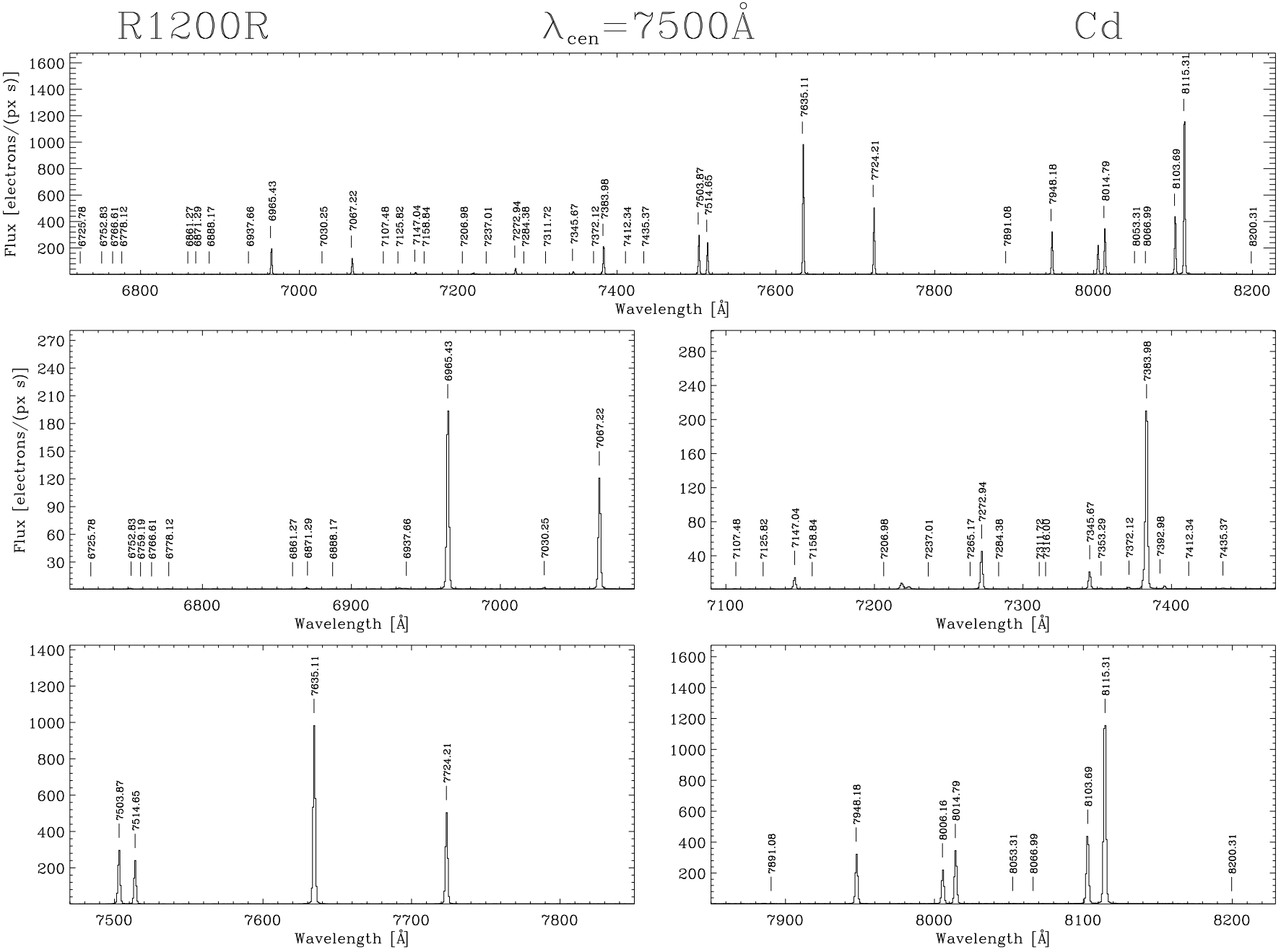

R1200R
λcen=7000Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
6212.50
6296.87
6307.66
6325.17
6359.98
6384.72
6416.31
6438.47
6464.94
6483.08
6538.11
6567.65
6604.85
6638.22
6666.36
6677.28
6725.78
6752.83
6766.61
6778.12
6861.27
6871.29
6888.17
6937.66
6965.43
7030.25
7067.22
7107.48
7125.82
7147.04
7158.84
7206.98
7237.01
7272.84
7284.38
7311.72
7353.29
7372.12
7383.98
7412.34
7435.37
7503.87
7514.65
7635.11
7724.21

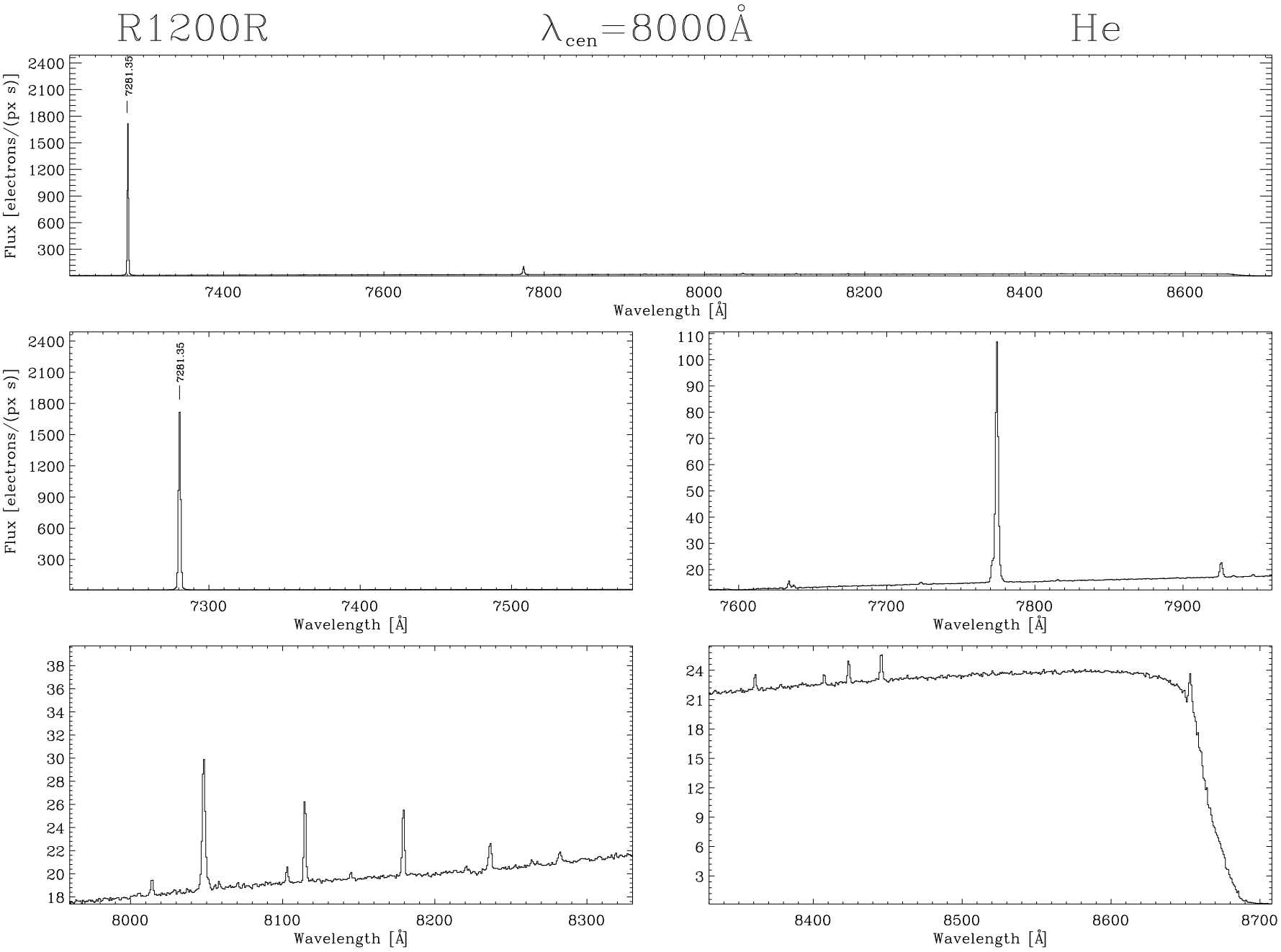

R1200R
λcen=7500Å
He
Flux [electrons/(px s)]
Wavelength [Å]
7065.19
7281.35

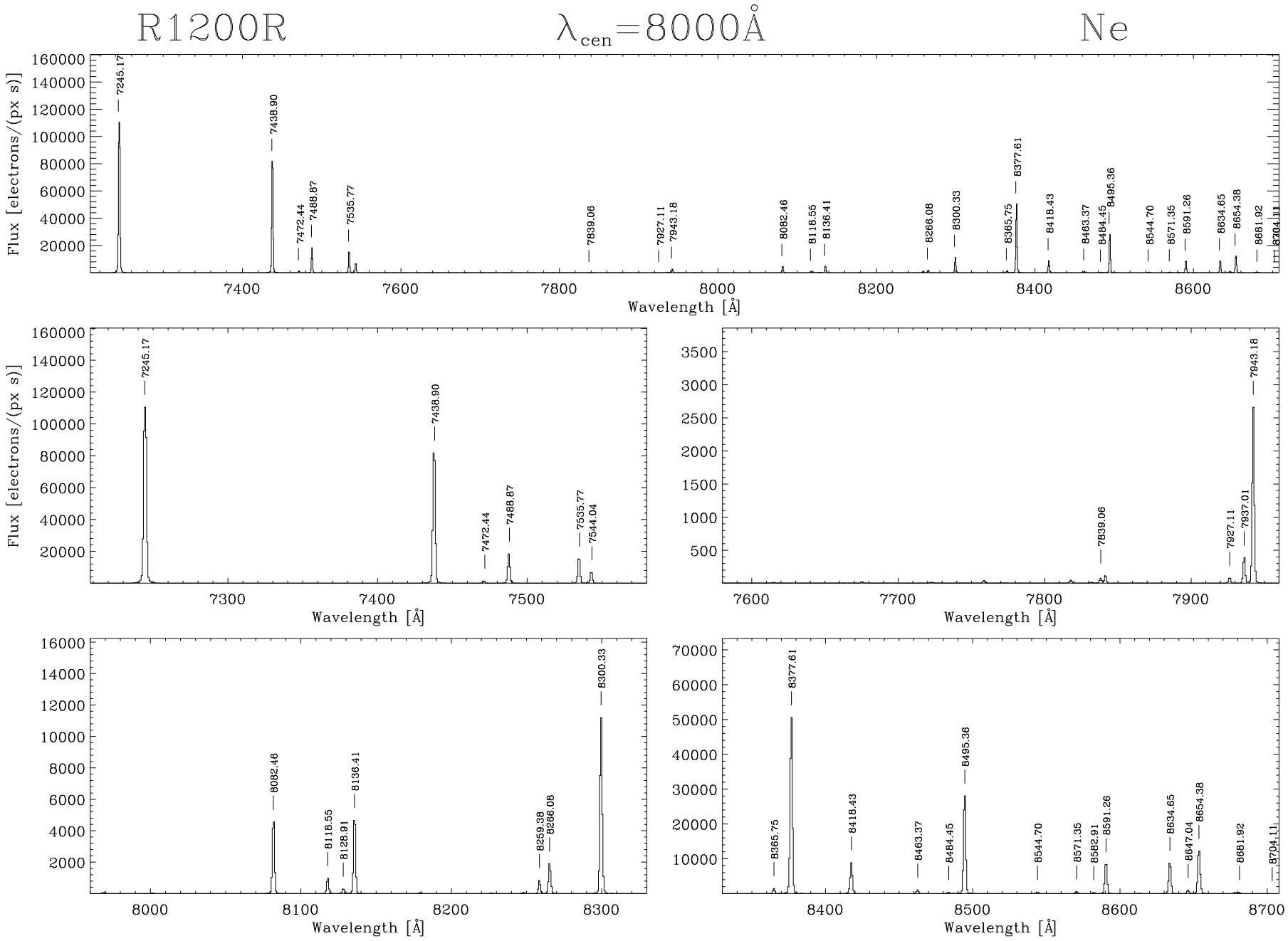

R1200R
λcen=7500Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
6717.04
6929.47
7032.41
7059.11
7173.94
7245.17
7438.90
7472.44
7488.87
7535.77
7839.06
7927.11
7943.18
8082.46
8118.55
8136.41
7024.05
7544.04
7937.01
8128.91

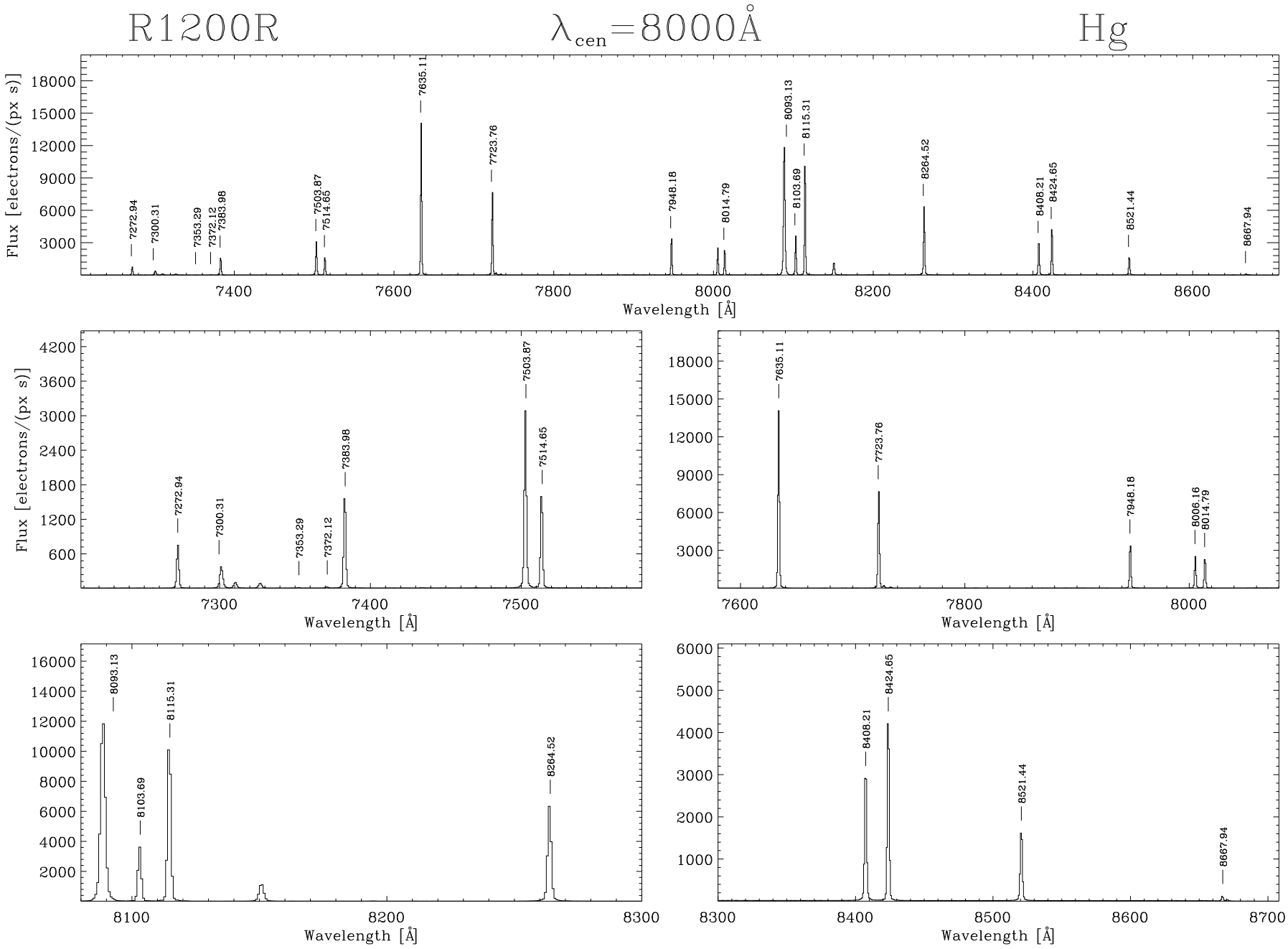

R1200R
λcen=7500Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
6716.42
6752.83
6871.29
6907.52
6965.43
7030.25
7067.22
7081.89
7091.86
7147.04
7272.94
7300.31
7353.29
7372.12
7383.98
7503.87
7514.65
7635.11
7723.76
7948.18
8006.16
8014.79
8093.13
8103.69
8115.31

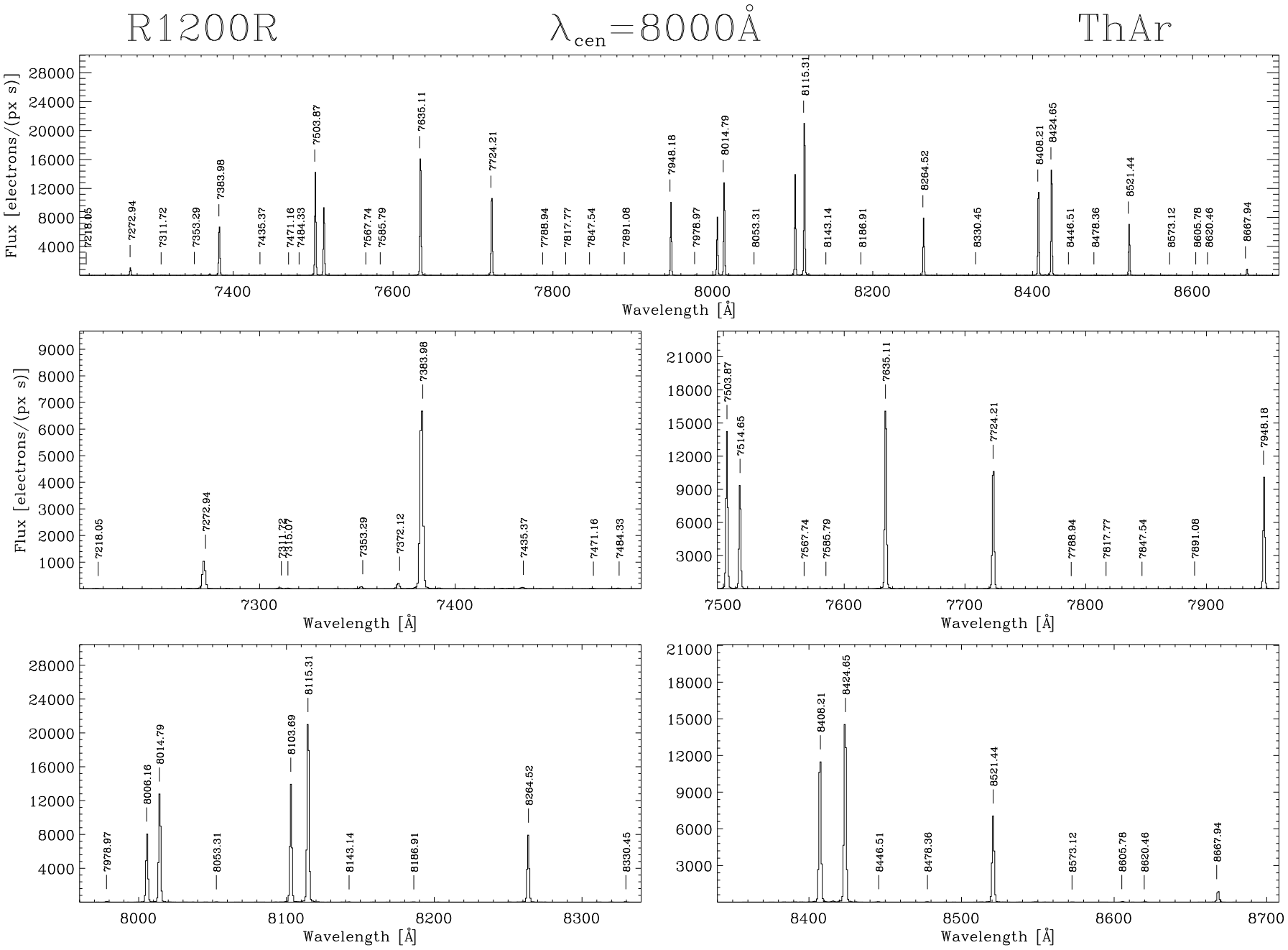

R1200R
λcen=7500Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
6727.46
6752.83
6778.31
6834.92
6871.29
6888.17
6911.23
6937.66
6965.43
6989.66
7030.25
7087.22
7107.48
7125.82
7147.04
7168.90
7206.98
7272.94
7311.72
7353.29
7383.98
7435.37
7471.16
7484.33
7503.87
7567.74
7585.79
7635.11
7724.21
7788.94
7817.77
7847.54
7891.08
7948.18
7978.97
8014.79
8053.31
8115.31
8143.14
8186.91
6719.22
6766.61
6829.04
6854.92
6861.27
6879.58
6943.61
7000.81
7018.57
7372.12
7514.65

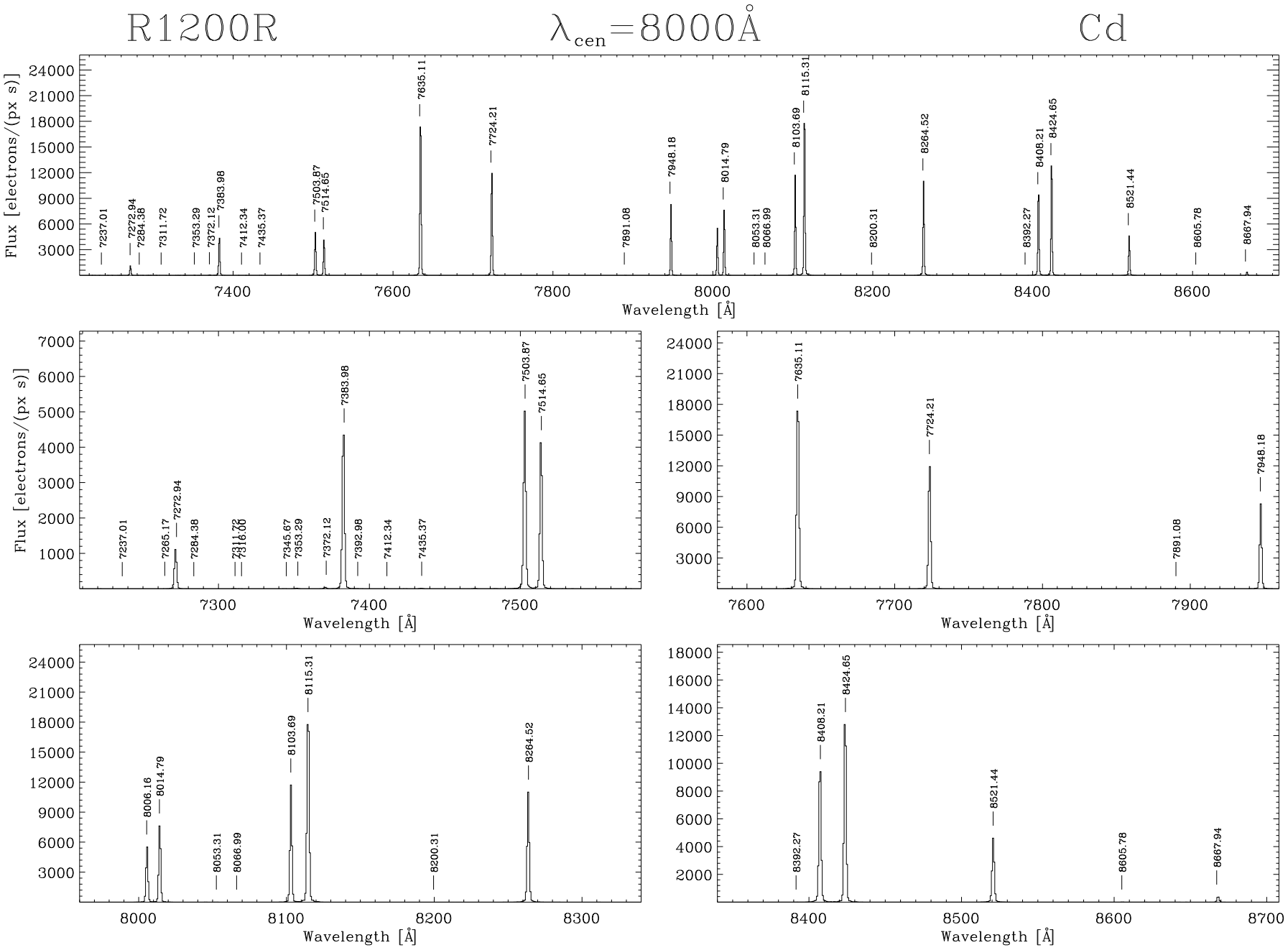

R1200R
λcen=7500Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
6725.78
6752.83
6759.19
6766.61
6778.12
6861.27
6871.29
6888.17
6937.66
6965.43
7030.25
7067.22
7107.48
7125.82
7147.04
7158.84
7206.98
7237.01
7265.17
7272.94
7284.38
7311.72
7316.00
7345.67
7353.29
7372.12
7383.98
7392.98
7412.34
7435.37
7503.87
7514.65
7635.11
7724.21
7891.08
7948.18
8006.16
8014.79
8053.31
8066.99
8103.69
8115.31
8200.31

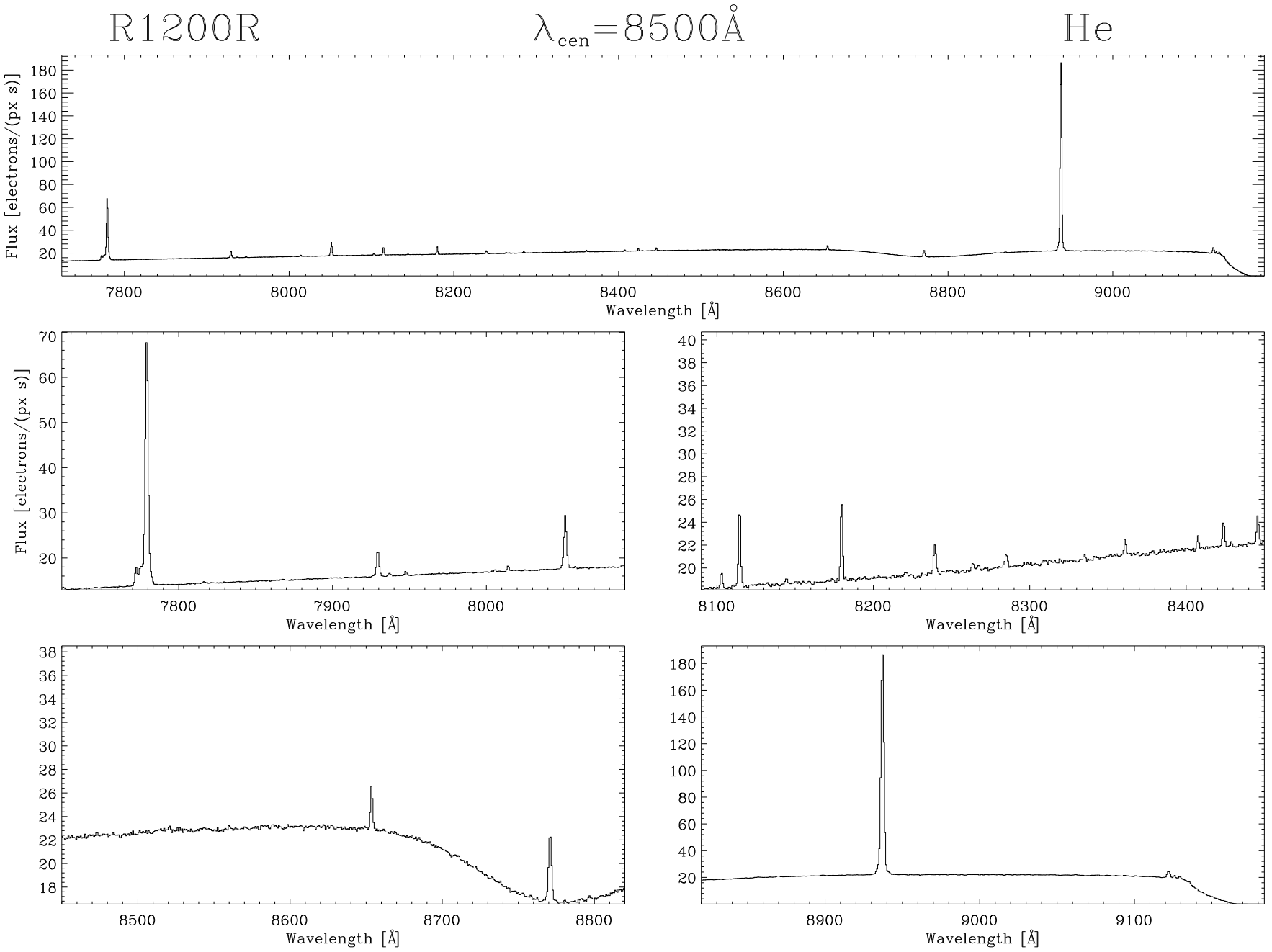

R1200R
λcen=8000Å
He
Flux [electrons/(px s)]
7281.35
Wavelength [Å]

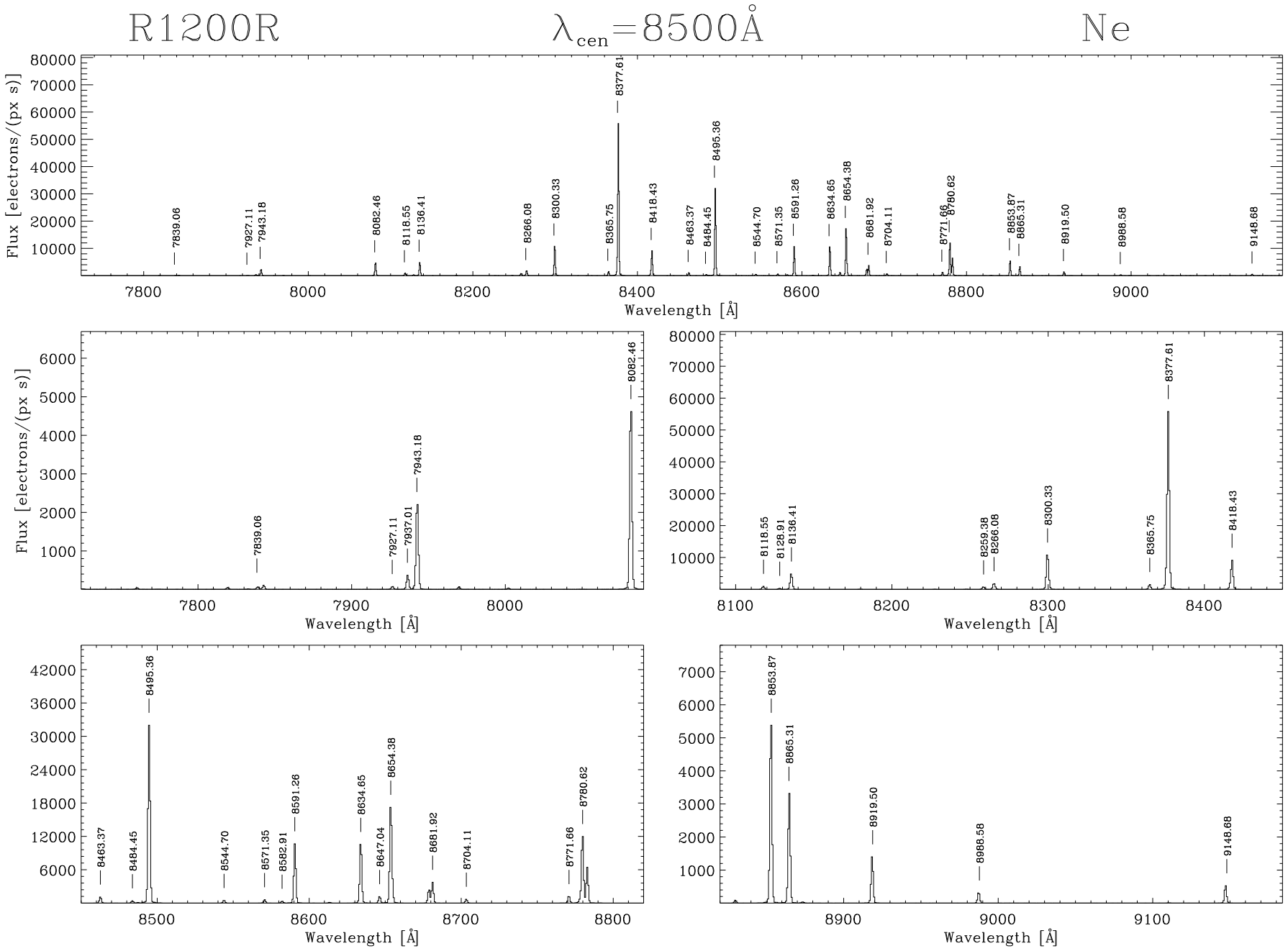

R1200R
λcen=8000Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
7245.17
7438.90
7472.44
7488.87
7535.77
7839.06
7927.11
7943.18
8082.46
8118.55
8136.41
8266.08
8300.33
8365.75
8377.61
8418.43
8463.37
8484.45
8495.36
8544.70
8571.35
8591.26
8634.65
8654.38
8681.92
8704.11
7544.04
7937.01
8128.91
8259.38

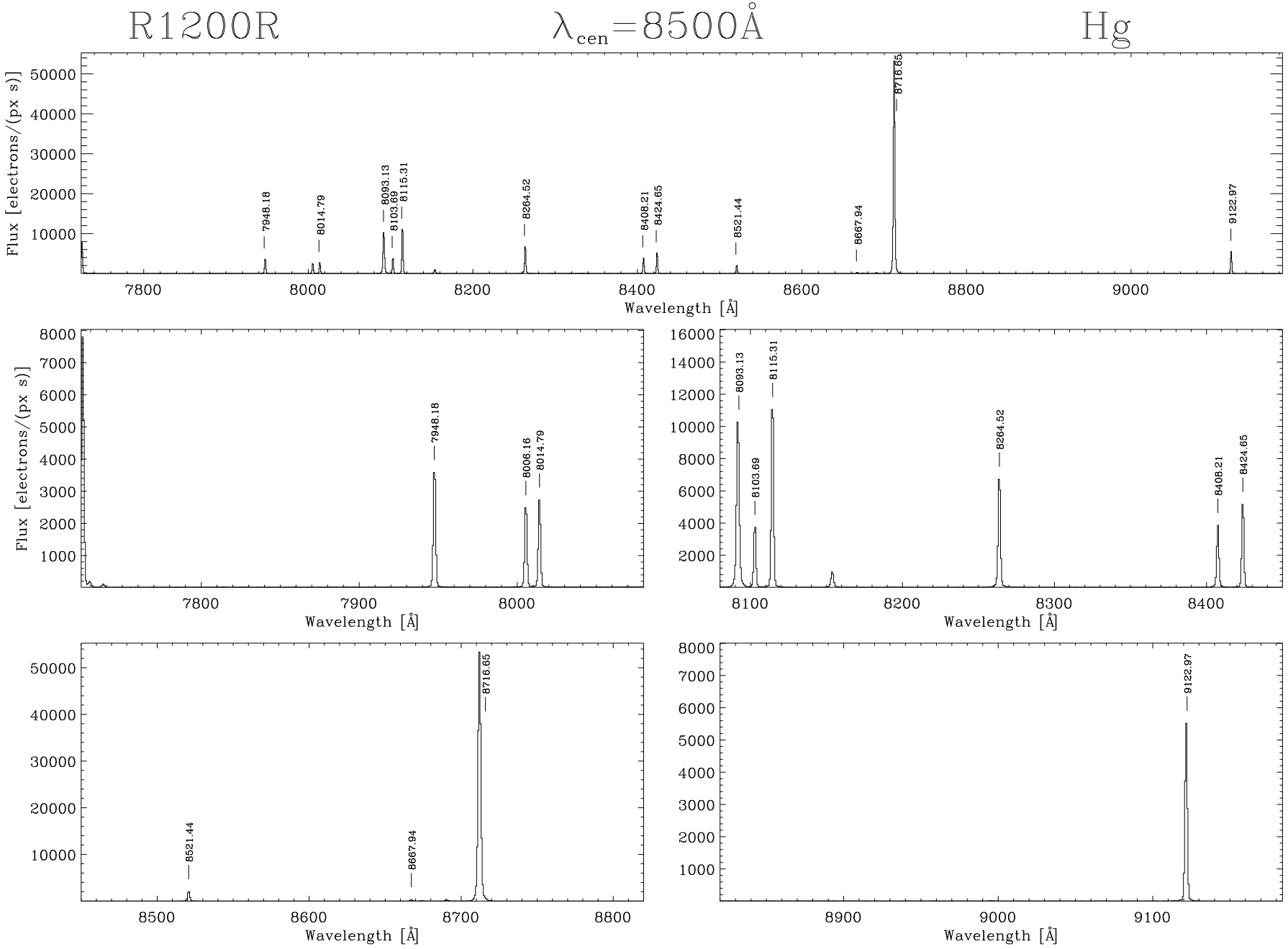

R1200R
λcen=8000Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
7272.94
7300.31
7353.29
7372.12
7383.98
7503.87
7514.65
7635.11
7723.76
7948.18
8014.79
8093.13
8103.69
8115.31
8264.52
8408.21
8424.65
8521.44
8667.94
8006.16

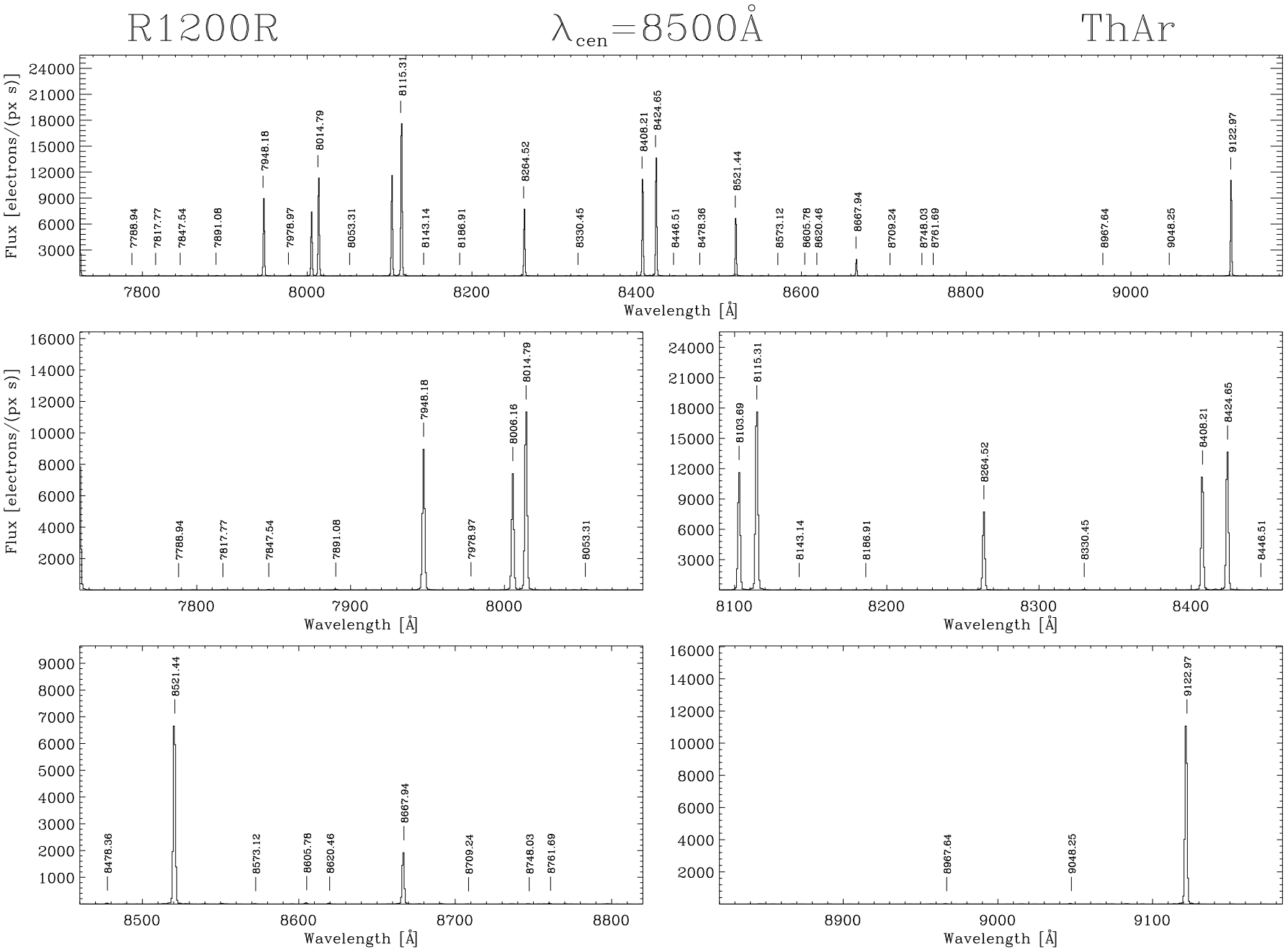

R1200R
λcen=8000Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
7218.05
7272.94
7311.72
7353.29
7383.98
7435.37
7471.16
7484.33
7503.87
7567.74
7585.79
7635.11
7724.21
7788.94
7817.77
7847.54
7891.08
7948.18
7978.97
8014.79
8053.31
8115.31
8143.14
8186.91
8264.52
8330.45
8408.21
8424.65
8446.51
8478.36
8521.44
8573.12
8605.78
8620.46
8667.94
7372.12
7514.65
8006.16
8103.69

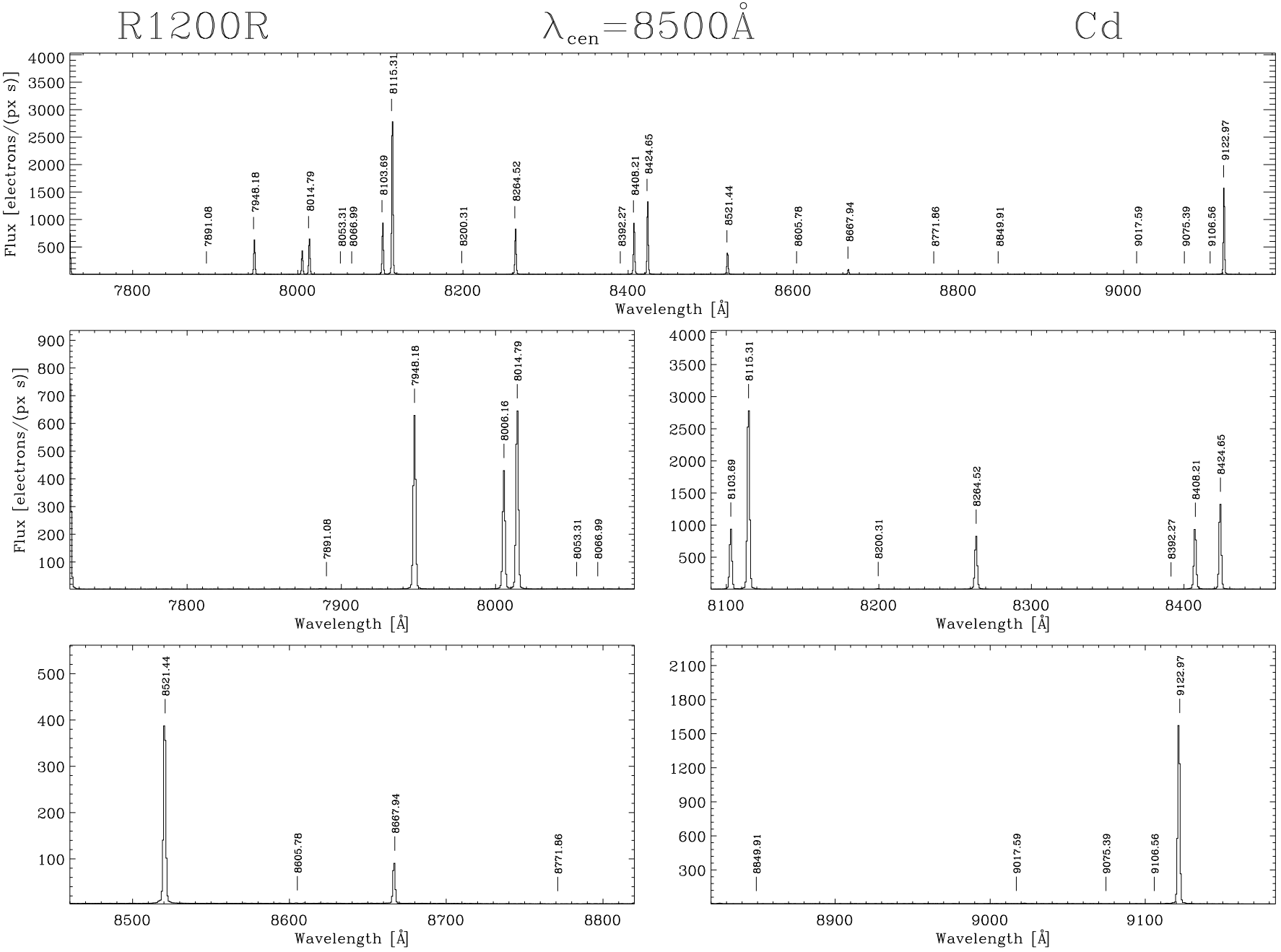

R1200R
λcen=8000Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
7237.01
7272.94
7284.38
7311.72
7353.29
7372.12
7383.98
7412.34
7435.37
7503.87
7514.65
7635.11
7724.21
7891.08
7948.18
8014.79
8053.31
8066.99
8103.69
8115.31
8200.31
8264.52
8392.27
8408.21
8424.65
8521.44
8605.78
8667.94
7265.17
7316.00
7345.67
8006.16

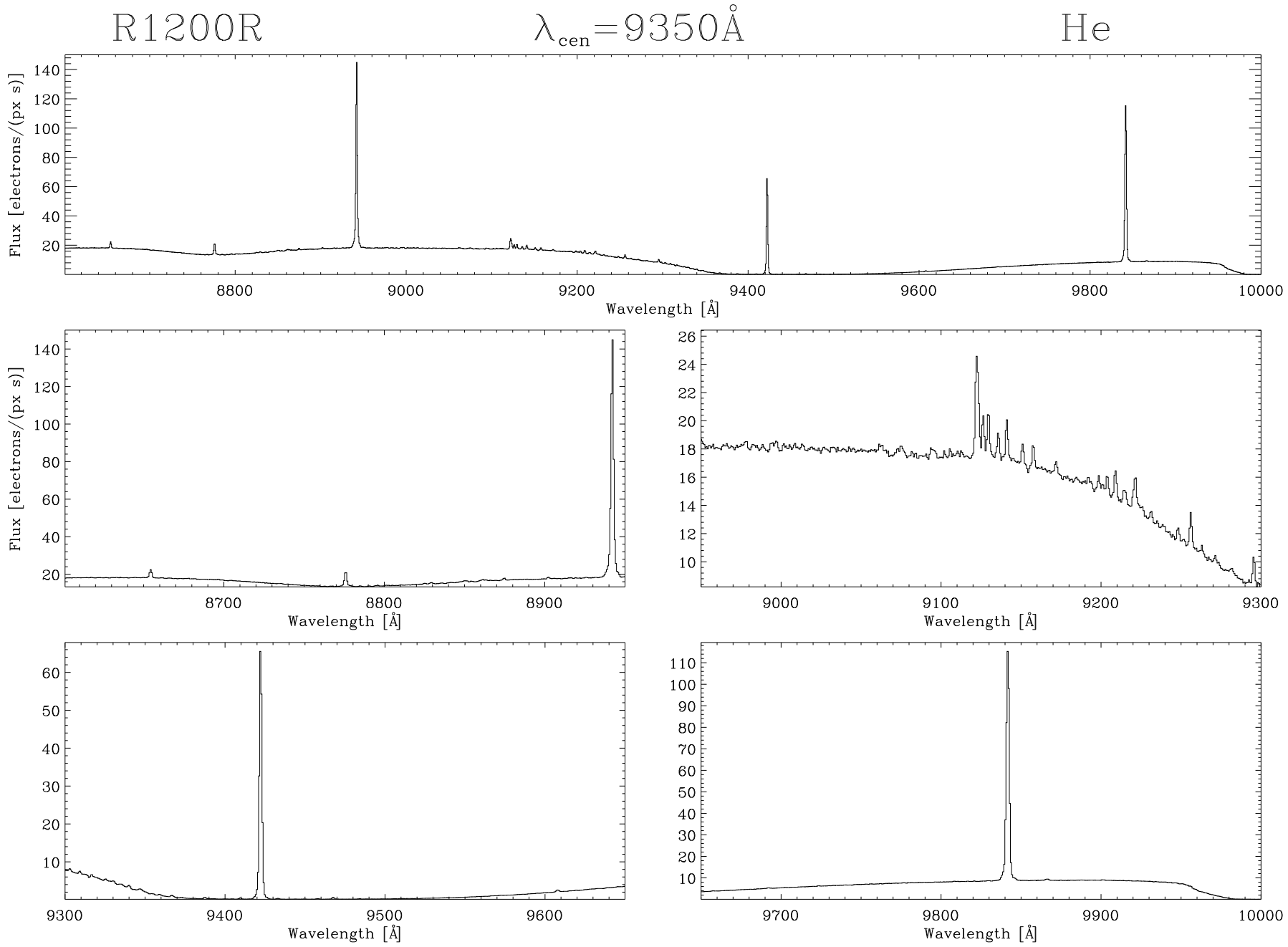

R1200R
λcen=8500Å
He
Flux [electrons/(px s)]
Wavelength [Å]

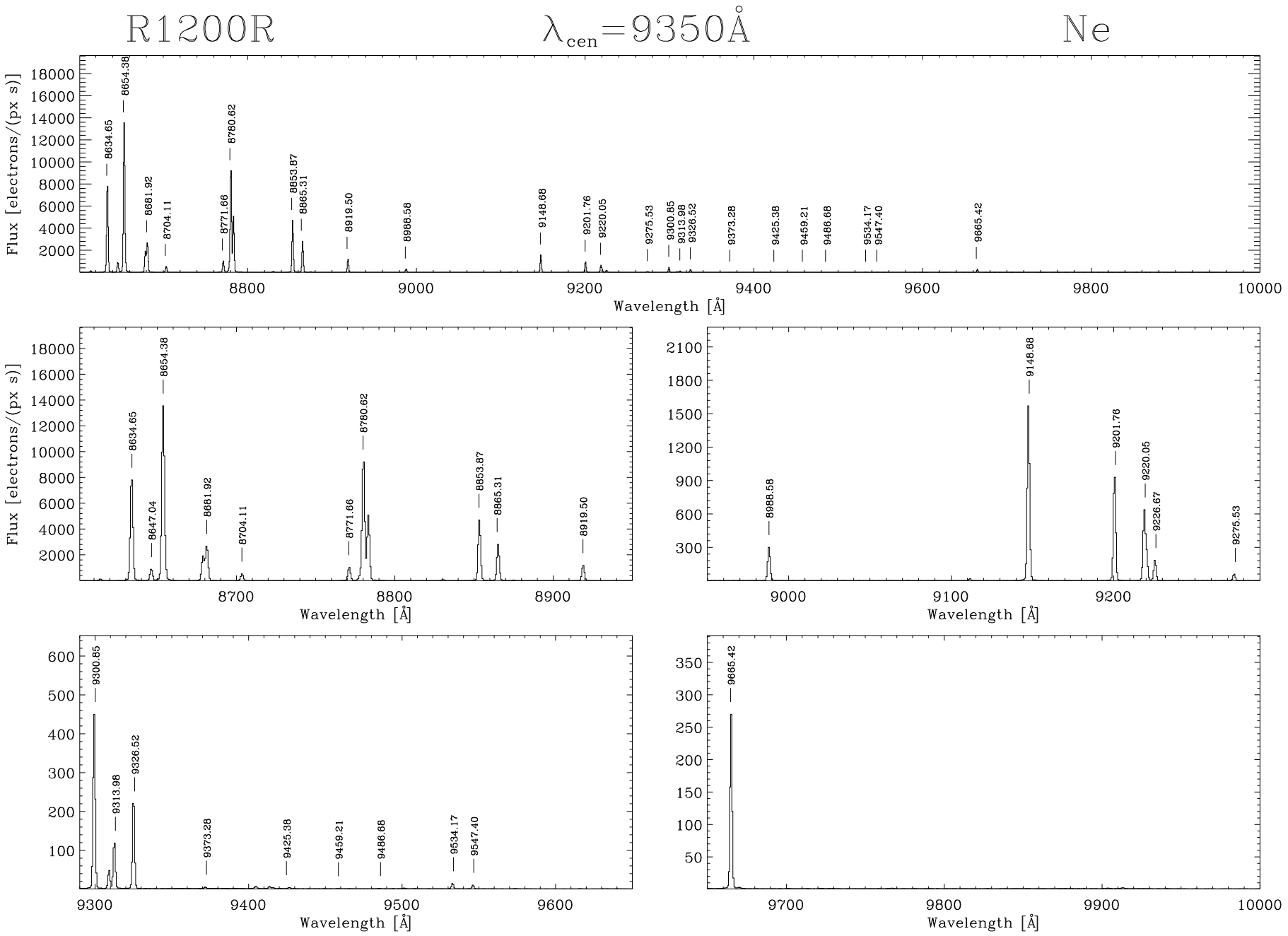

R1200R
λcen=8500Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
7839.06
7927.11
7943.18
8082.46
8118.55
8136.41
8266.08
8300.33
8365.75
8377.61
8418.43
8463.37
8484.45
8495.36
8544.70
8571.35
8591.26
8634.65
8654.38
8681.92
8704.11
8771.66
8780.62
8853.87
8865.31
8919.50
8988.58
9148.68
7937.01
8128.91
8259.38
8582.91
8647.04

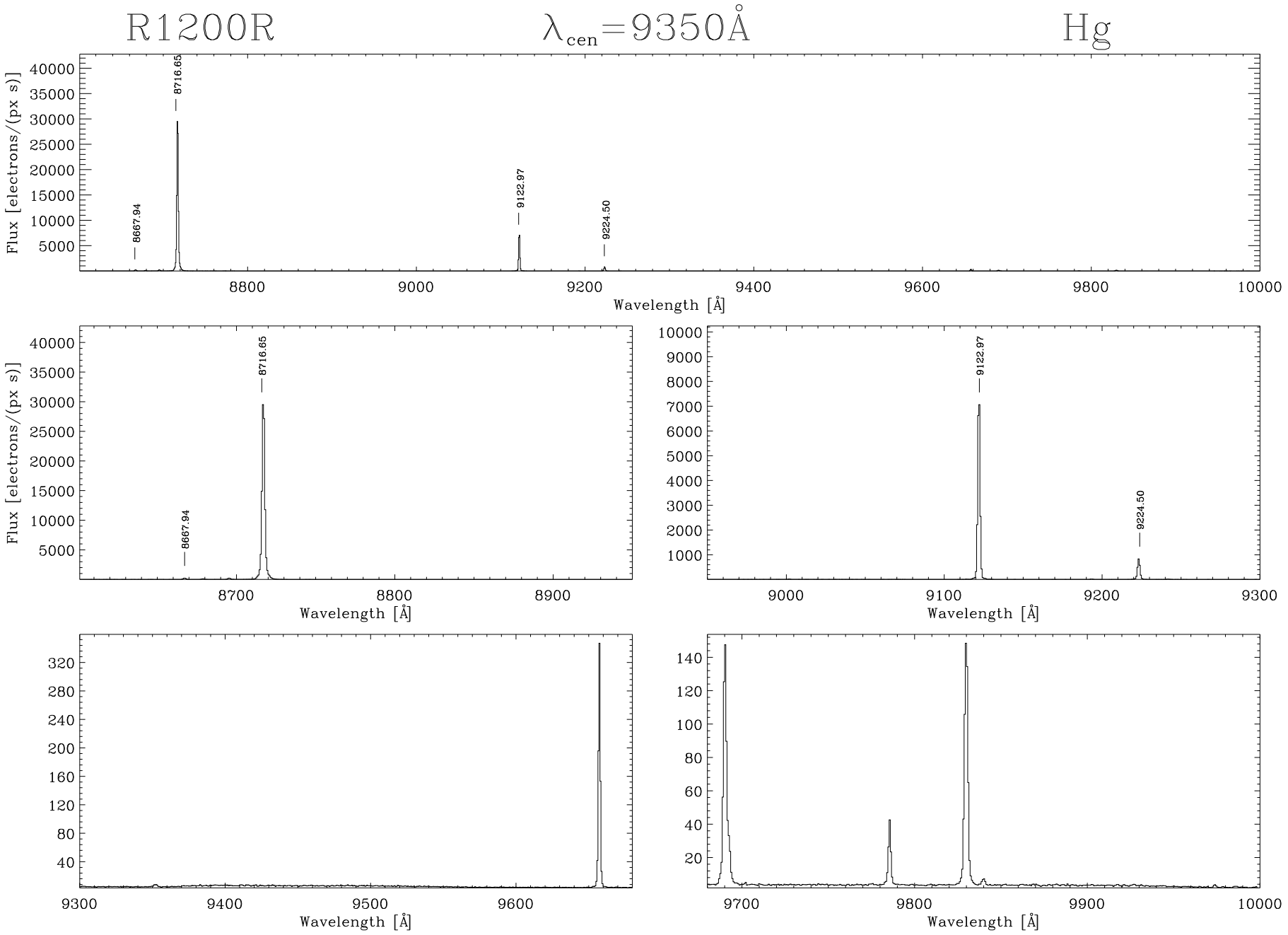

R1200R
λcen=8500Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
7948.18
8014.79
8093.13
8103.69
8115.31
8264.52
8408.21
8424.65
8521.44
8667.94
8716.65
9122.97
8006.16

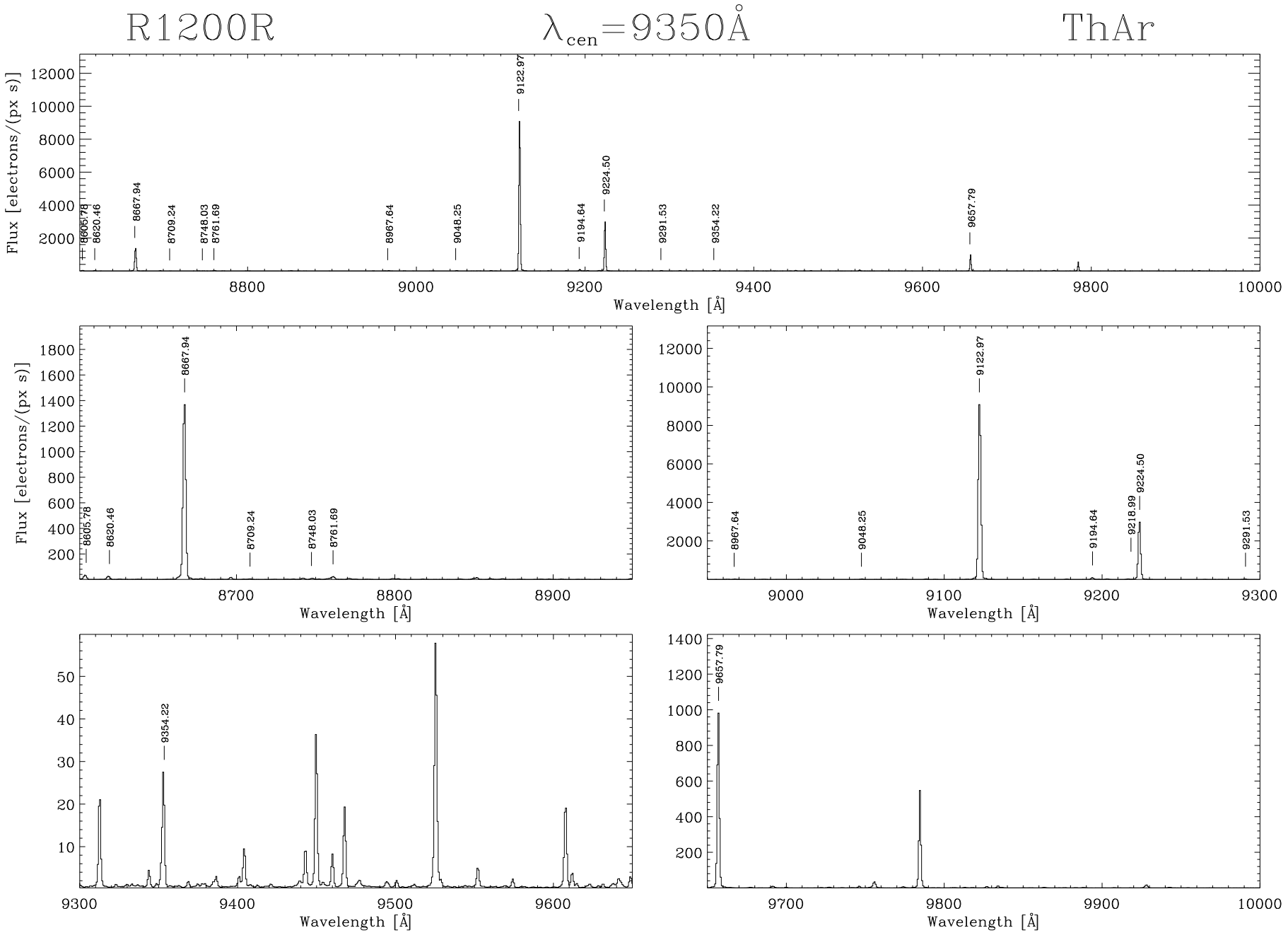

R1200R
λcen=8500Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
7788.94
7817.77
7847.54
7891.08
7948.18
7978.97
8014.79
8053.31
8115.31
8143.14
8186.91
8264.52
8330.45
8408.21
8424.65
8446.51
8478.36
8521.44
8573.12
8605.78
8620.46
8667.94
8709.24
8748.03
8761.69
8967.64
9048.25
9122.97
8006.16
8103.69

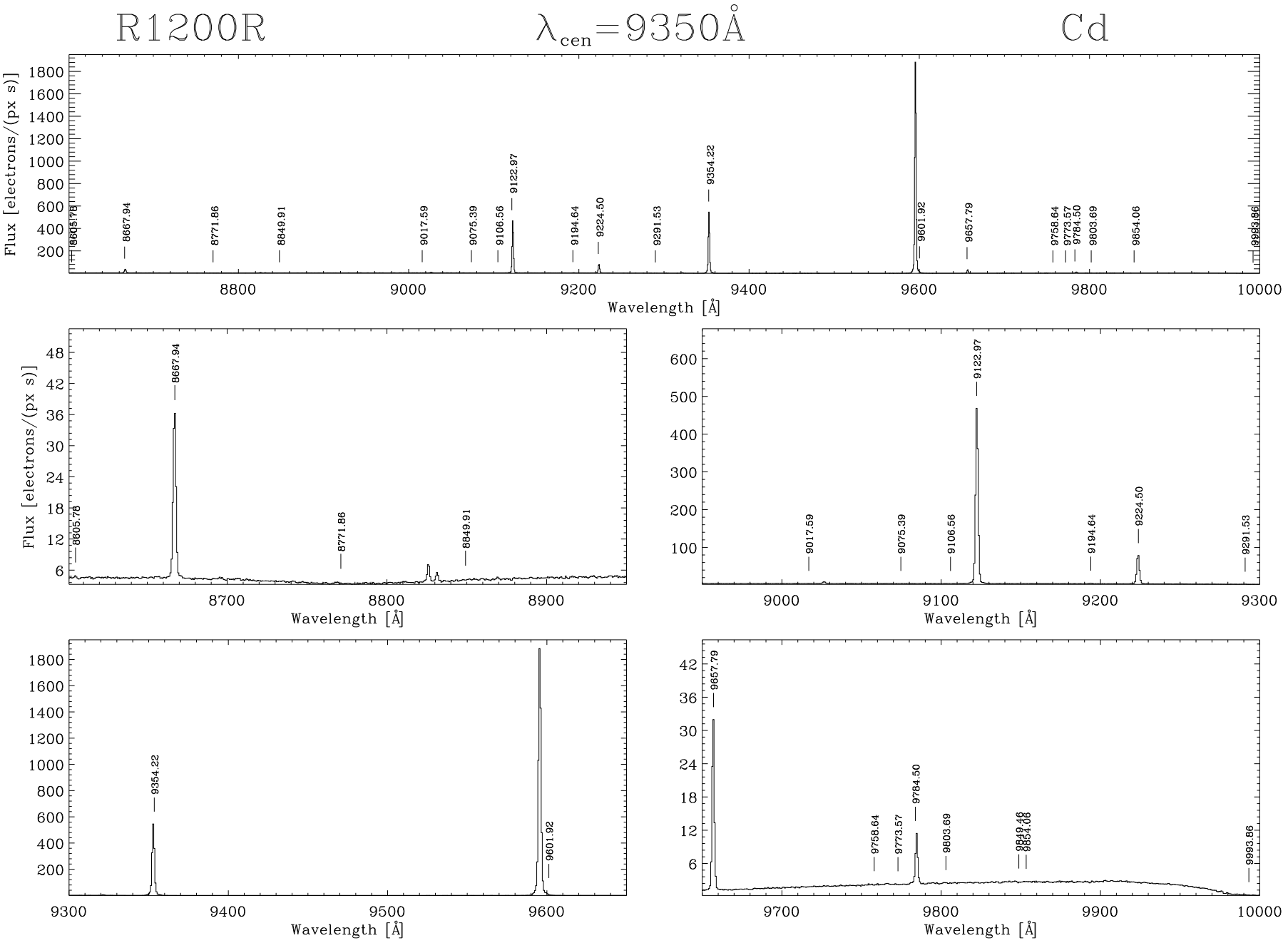

R1200R
λcen=8500Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
7891.08
7948.18
8014.79
8053.31
8066.99
8103.69
8115.31
8200.31
8264.52
8392.27
8408.21
8424.65
8521.44
8605.78
8667.94
8771.86
8849.91
9017.59
9075.39
9106.56
9122.97
8006.16


R1200R
λcen=9350Å
He
Flux [electrons/(px s)]
Wavelength [Å]

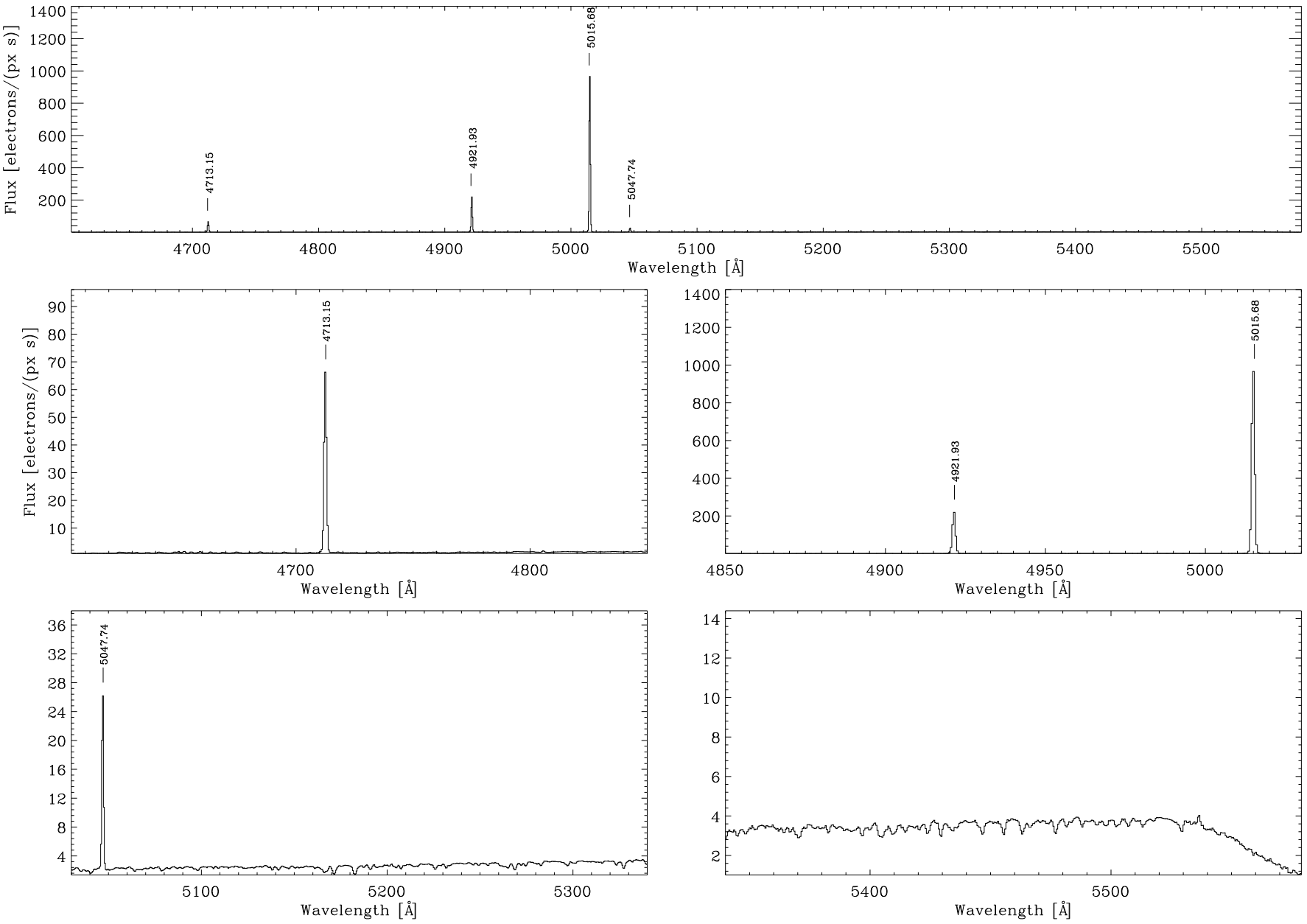

R1200R
λcen=9350Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
8634.65
8654.38
8681.92
8704.11
8771.66
8780.62
8853.87
8865.31
8919.50
8988.58
9148.68
9201.76
9220.05
9275.53
9300.85
9313.98
9326.52
9373.28
9425.38
9459.21
9486.68
9534.17
9547.40
9665.42
8647.04
9236.67

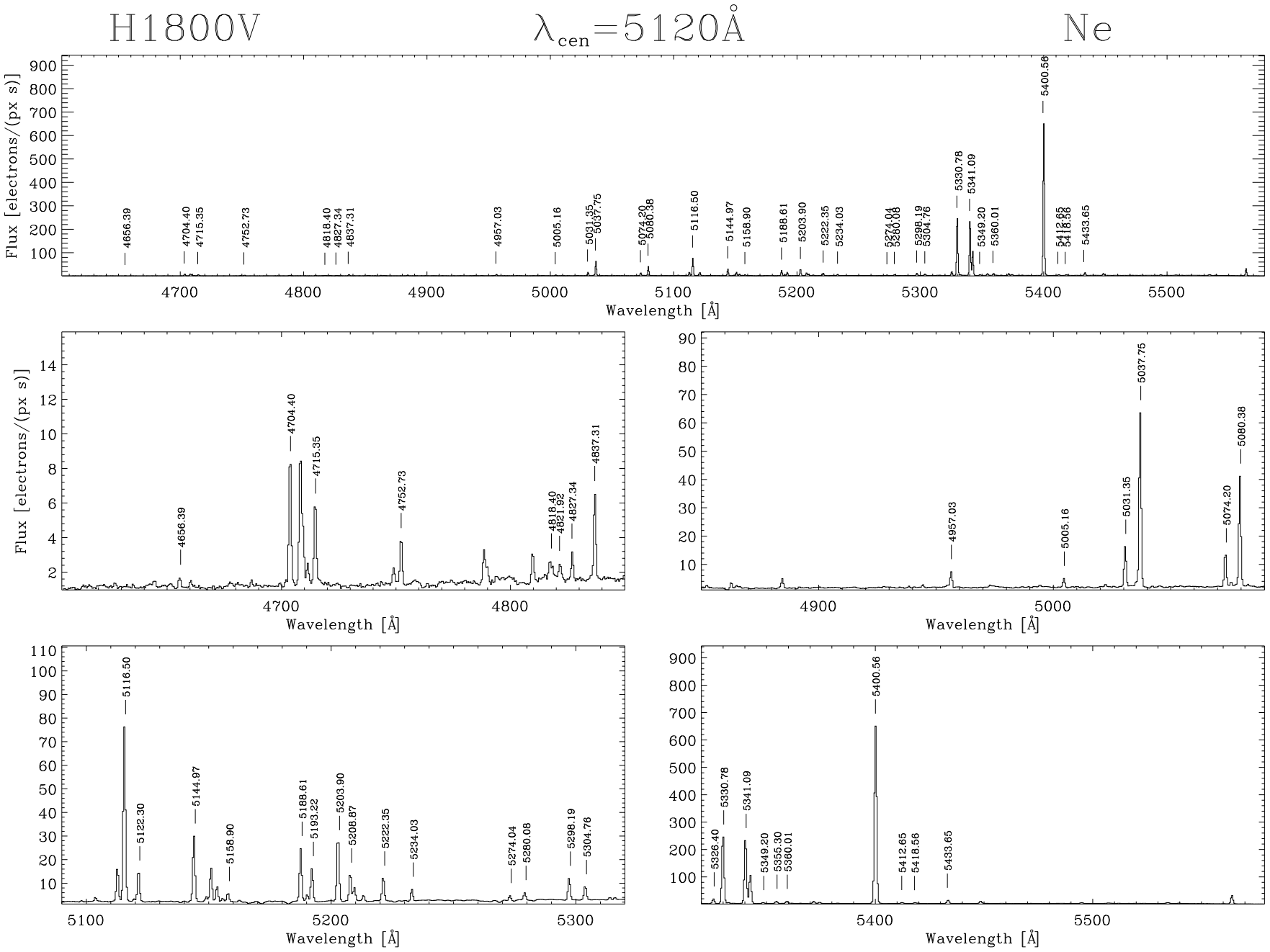

R1200R
λcen=9350Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
8667.94
8716.65
9122.97
9224.50

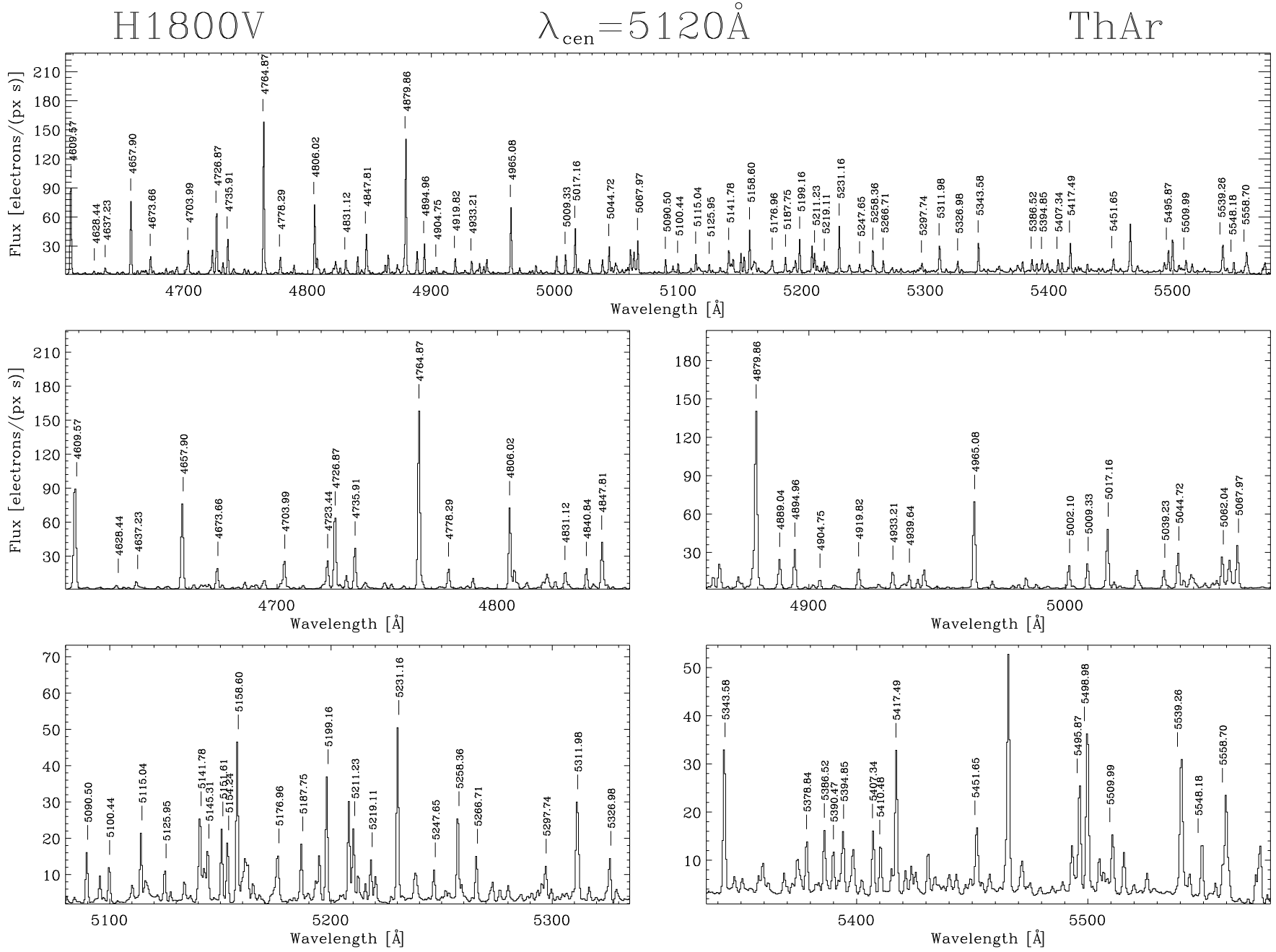

R1200R
λcen=9350Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
8605.78
8620.46
8667.94
8709.24
8748.03
8761.69
8967.64
9048.25
9122.97
9194.64
9224.50
9291.53
9354.22
9657.79
9218.99

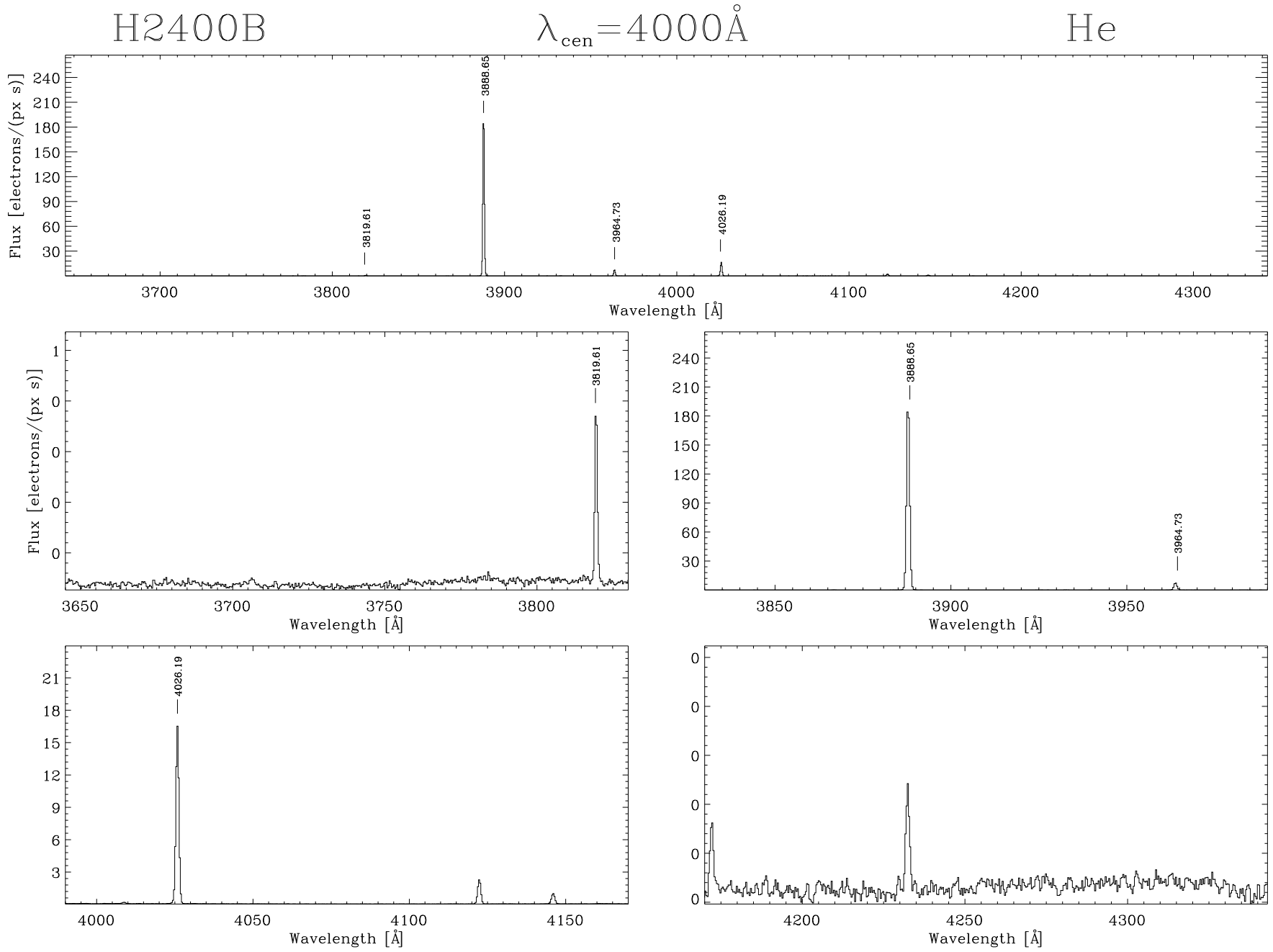

R1200R
λcen=9350Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
8605.78
8667.94
8771.86
8849.91
9017.59
9075.39
9106.56
9122.97
9194.64
9224.50
9291.53
9354.22
9601.92
9657.79
9758.64
9773.57
9784.50
9803.69
9854.06
9993.86

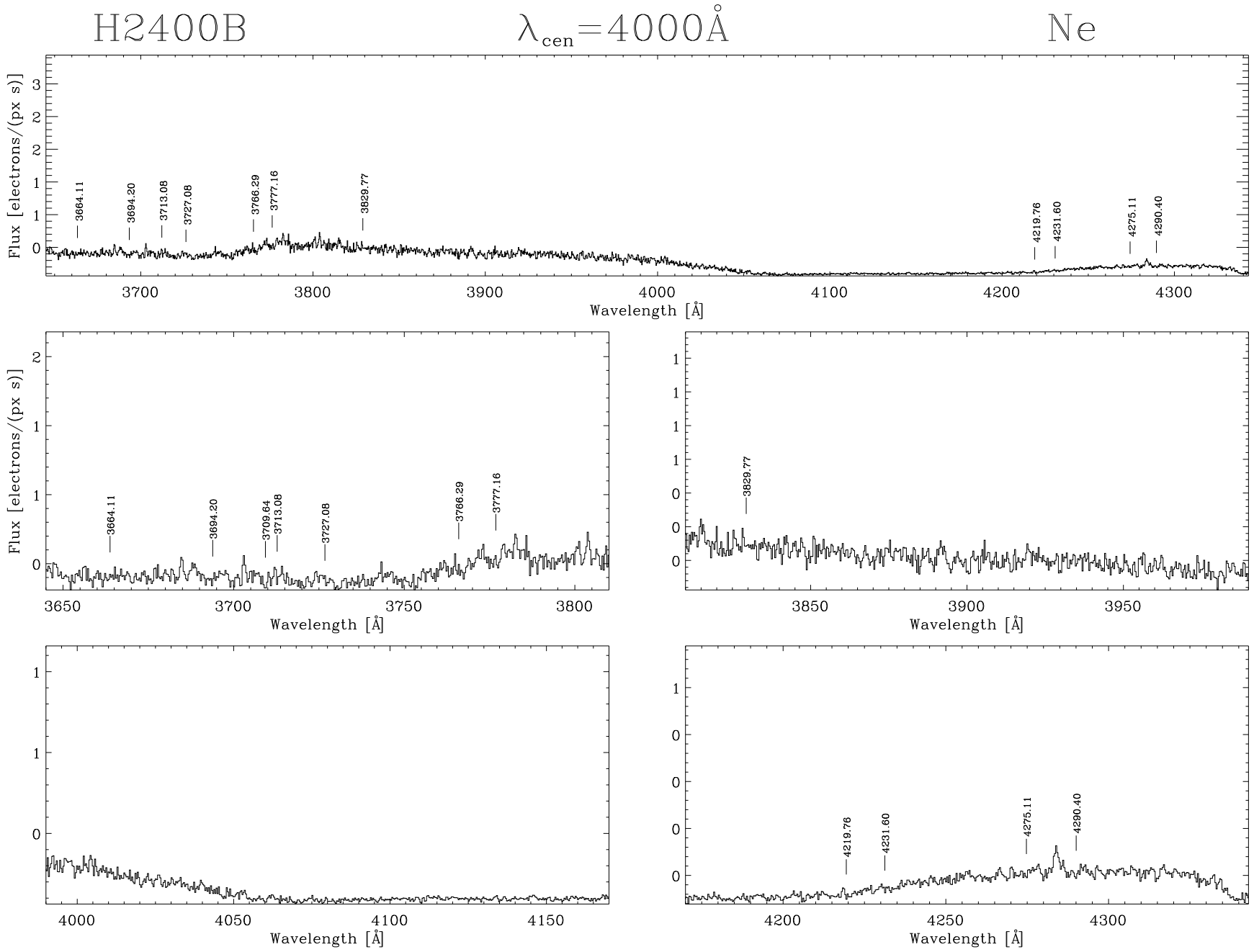
H1800V λ_{cen}=5120Å He

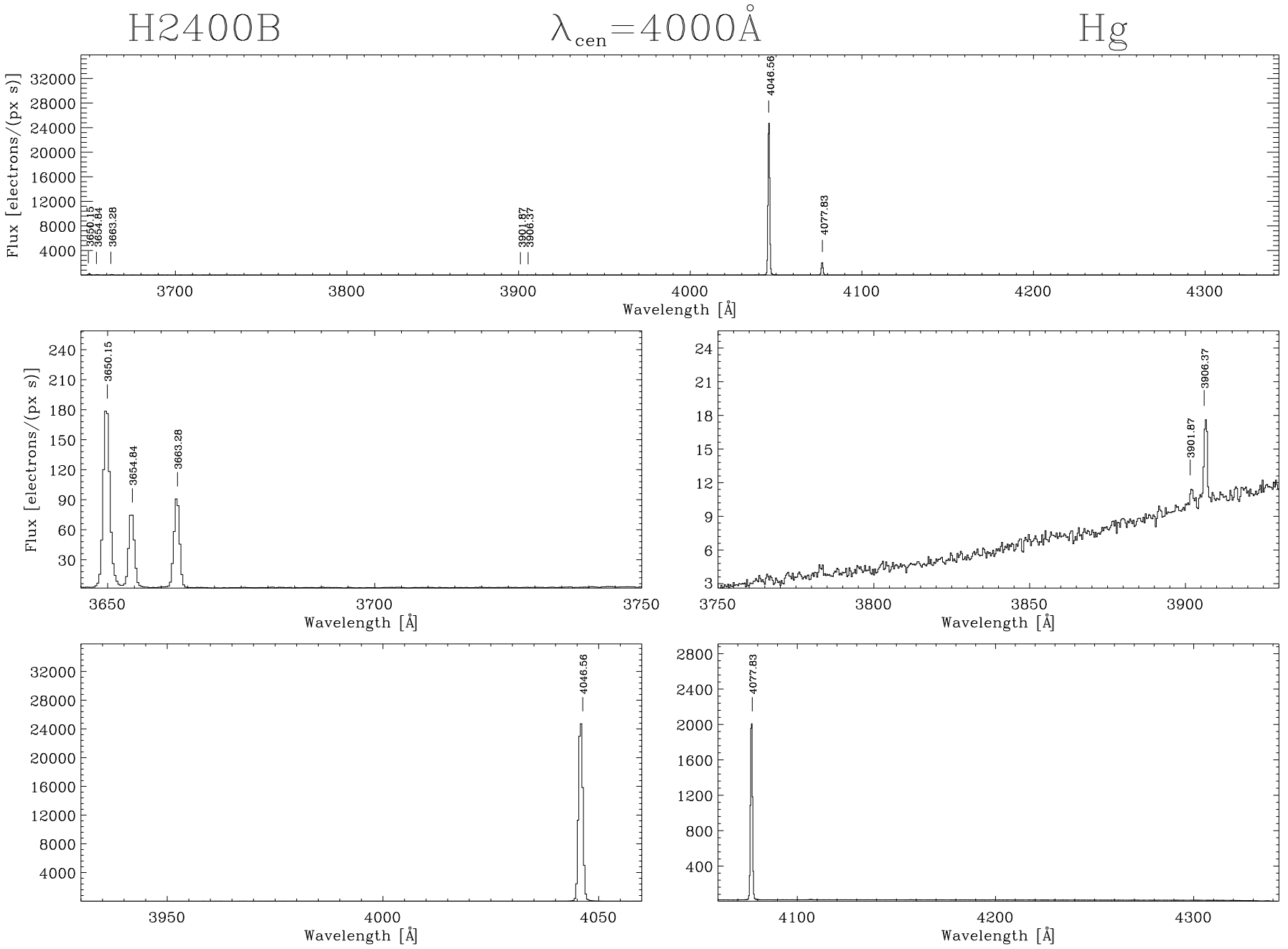


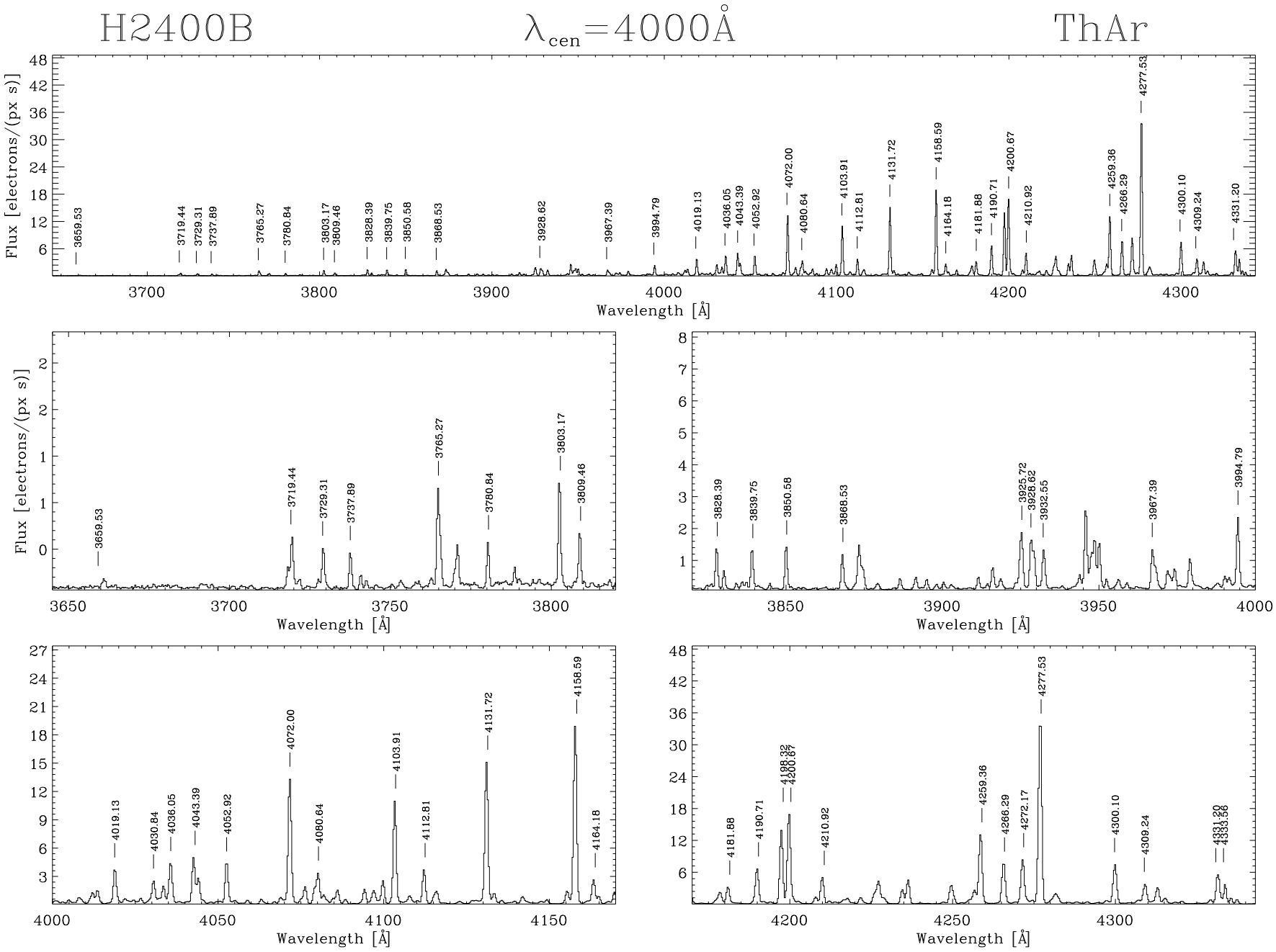

H1800V
λcen=5120Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
4656.39
4704.40
4715.35
4752.73
4818.40
4827.34
4837.31
4957.03
5005.16
5031.35
5037.75
5074.20
5080.38
5116.50
5144.97
5158.90
5188.61
5203.90
5222.35
5234.03
5274.04
5280.08
5298.19
5304.76
5330.78
5341.09
5349.20
5360.01
5400.56
5412.65
5418.56
5433.65
4821.92
5122.30
5193.22
5208.87
5326.40
5355.30

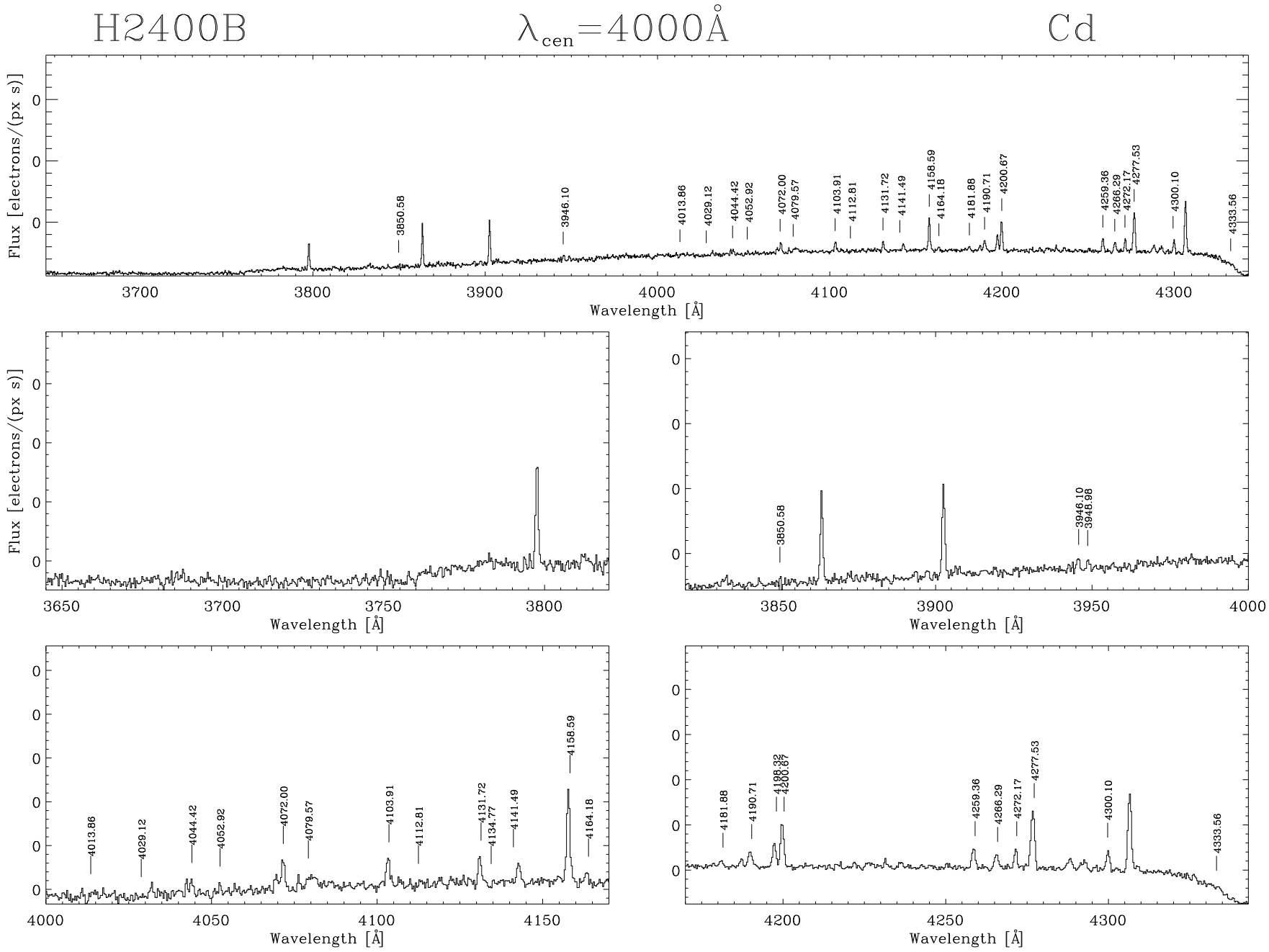

H1800V
λcen=5120Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
4609.57
4628.44
4637.23
4657.90
4673.66
4703.99
4726.87
4735.91
4764.87
4778.29
4806.02
4831.12
4847.81
4879.86
4894.96
4904.75
4919.82
4933.21
4965.08
5009.33
5017.16
5044.72
5067.97
5090.50
5100.44
5115.04
5125.95
5141.78
5158.60
5176.96
5187.75
5199.16
5211.23
5219.11
5231.16
5247.65
5258.36
5266.71
5297.74
5311.98
5326.98
5343.58
5386.52
5394.85
5407.34
5417.49
5451.65
5495.87
5509.99
5539.26
5548.18
5558.70
4723.44
4840.84
4889.04
4939.64
5002.10
5039.23
5062.04
5141.31
5153.61
5154.24
5378.84
5390.47
5410.48
5498.98


H2400B
λcen=4000Å
He
Flux [electrons/(px s)]
Wavelength [Å]
3819.61
3888.65
3964.73
4026.19

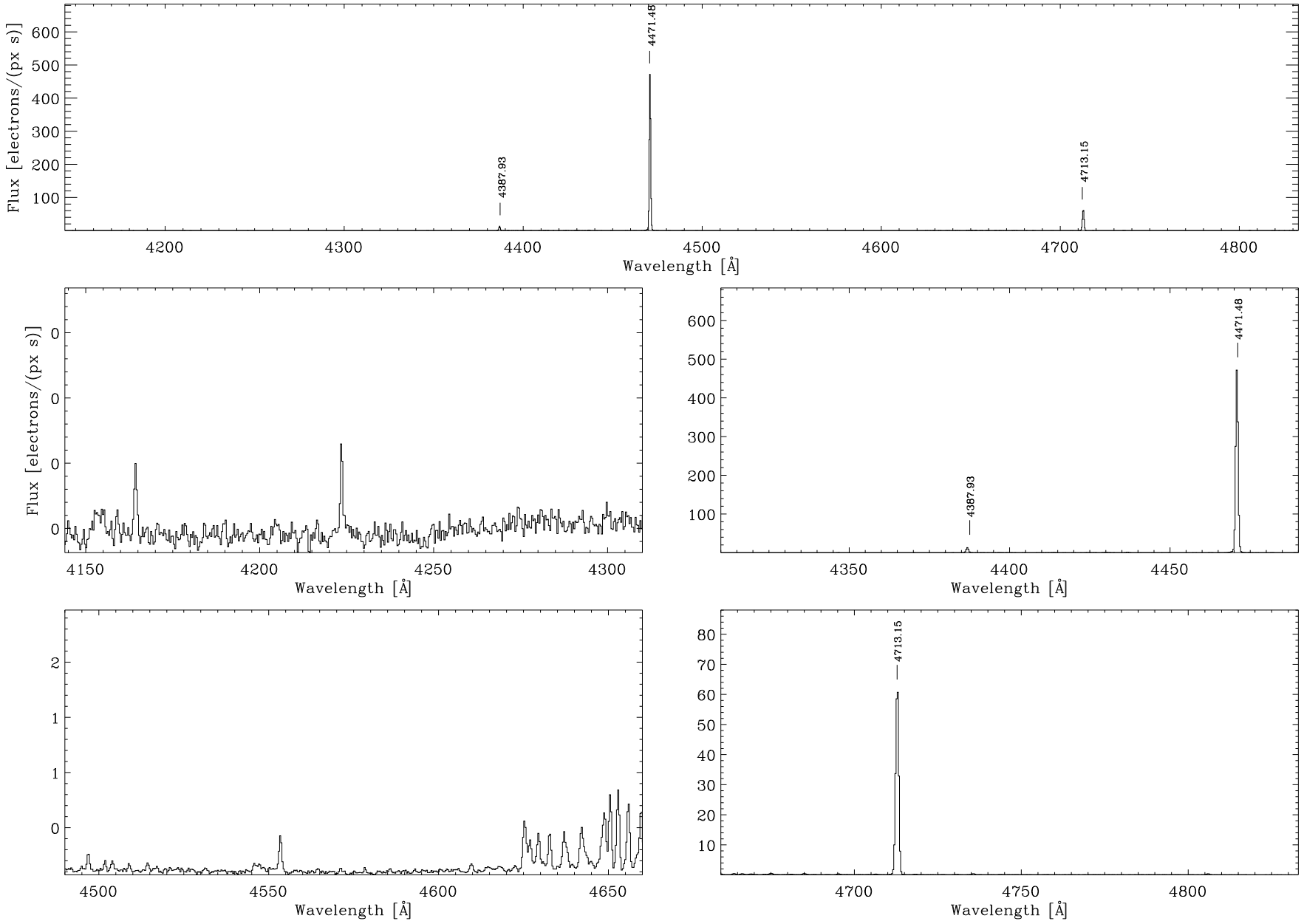

H2400B
λcen=4000Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
3664.11
3694.20
3713.08
3727.08
3766.29
3777.16
3829.77
4219.76
4231.60
4275.11
4290.40
3709.64

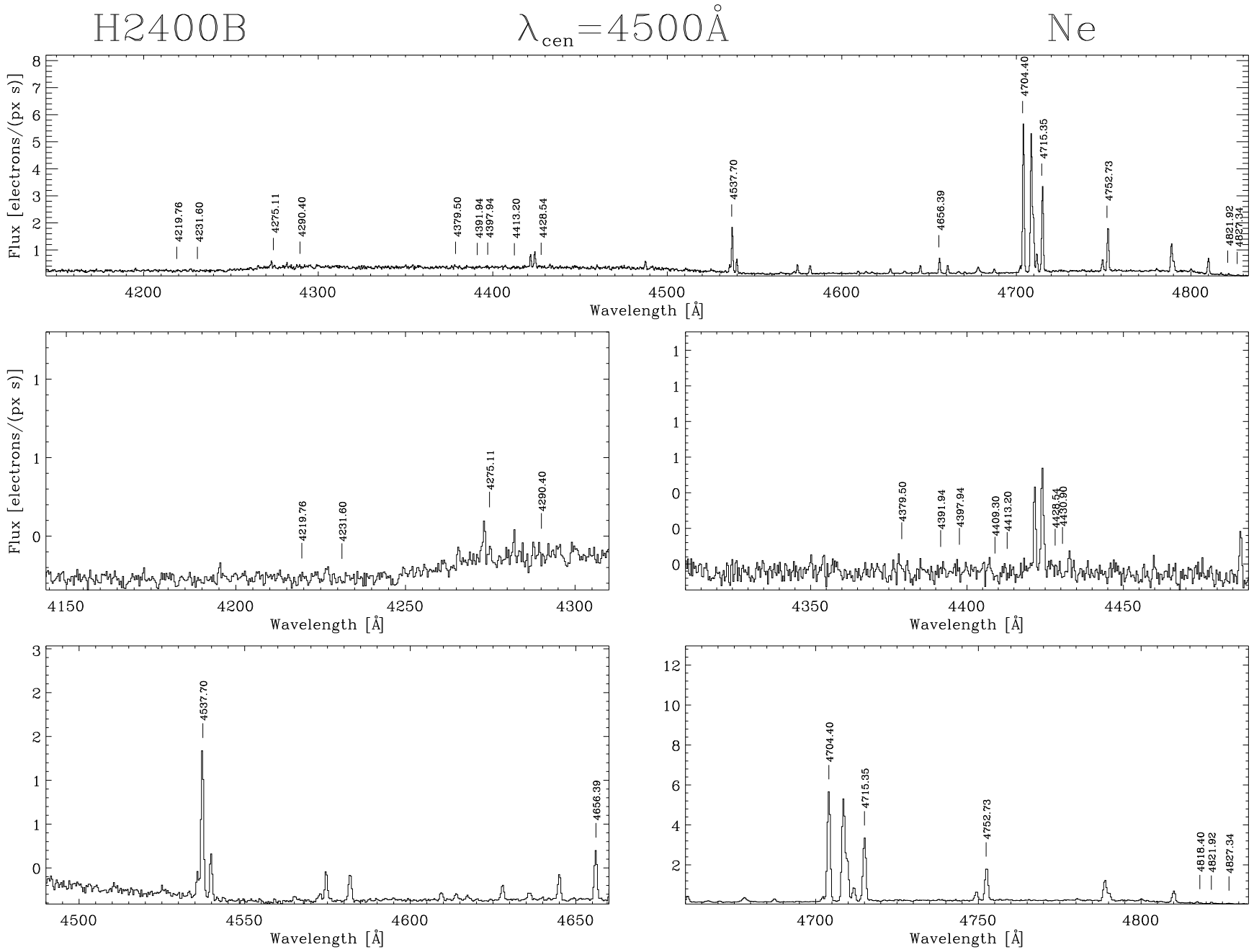

H2400B
λcen=4000Å
Hg
Flux [electrons/(px s)]
Wavelength [Å]
3650.15
3654.84
3663.28
3901.87
3906.37
4046.56
4077.83


H2400B
λcen=4000Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
3659.53
3719.44
3729.31
3737.89
3765.27
3780.84
3803.17
3809.46
3828.39
3839.75
3850.58
3868.53
3928.62
3967.39
3994.79
4019.13
4036.05
4043.39
4052.92
4072.00
4080.64
4103.91
4112.81
4131.72
4158.59
4164.18
4181.88
4190.71
4200.67
4210.92
4259.36
4266.29
4277.53
4300.10
4309.24
4331.20
3925.72
3928.62
3932.55
4030.84
4198.32
4200.87
4272.17
4333.56

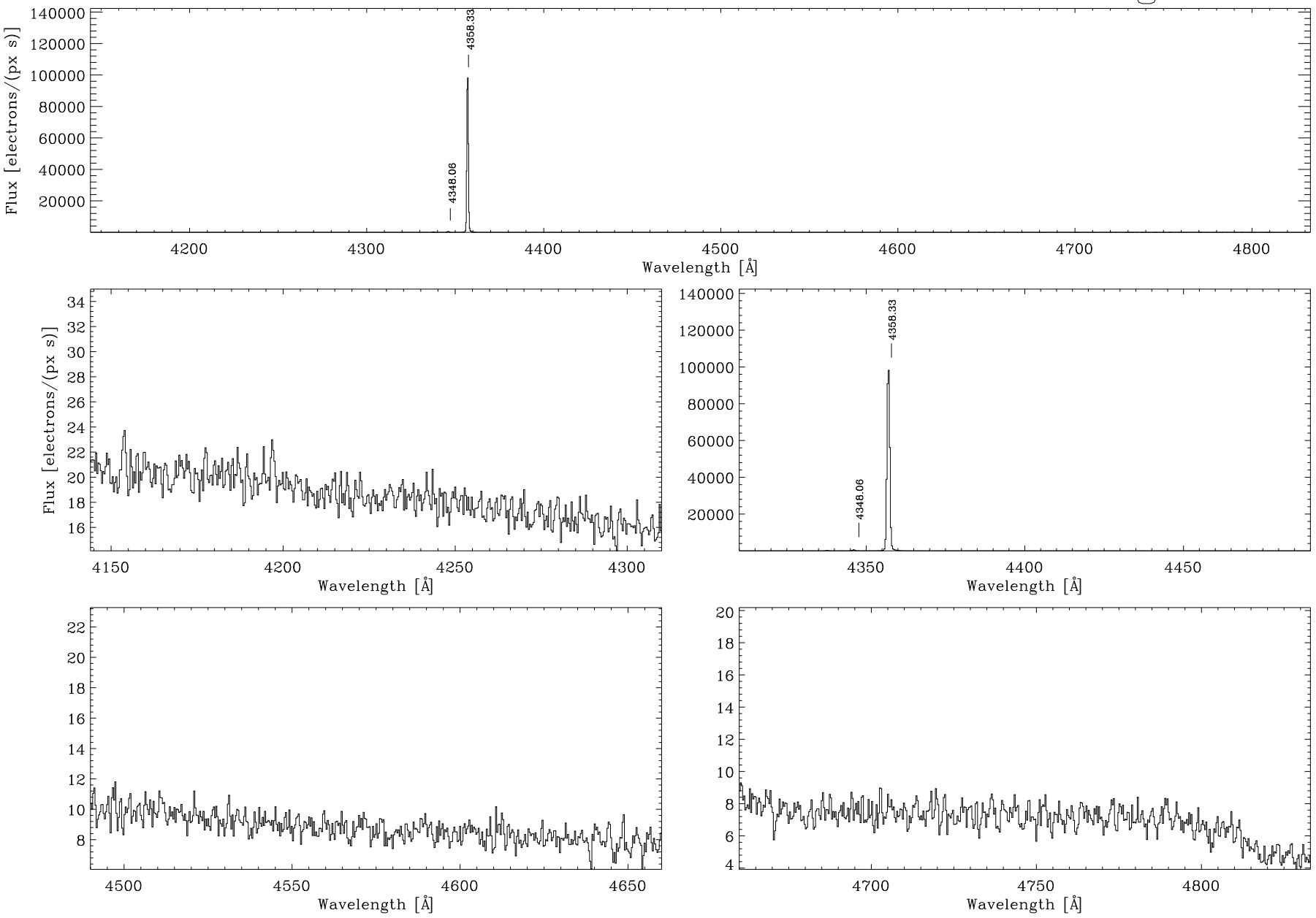

H2400B
λcen=4000Å
Cd
Flux [electrons/(px s)]
Wavelength [Å]
3850.58
3946.10
4013.86
4029.12
4044.42
4052.92
4072.00
4079.57
4103.91
4112.81
4131.72
4141.49
4158.59
4164.18
4181.88
4190.71
4200.67
4259.36
4266.29
4272.17
4277.53
4300.10
4333.56
3948.98
4134.77

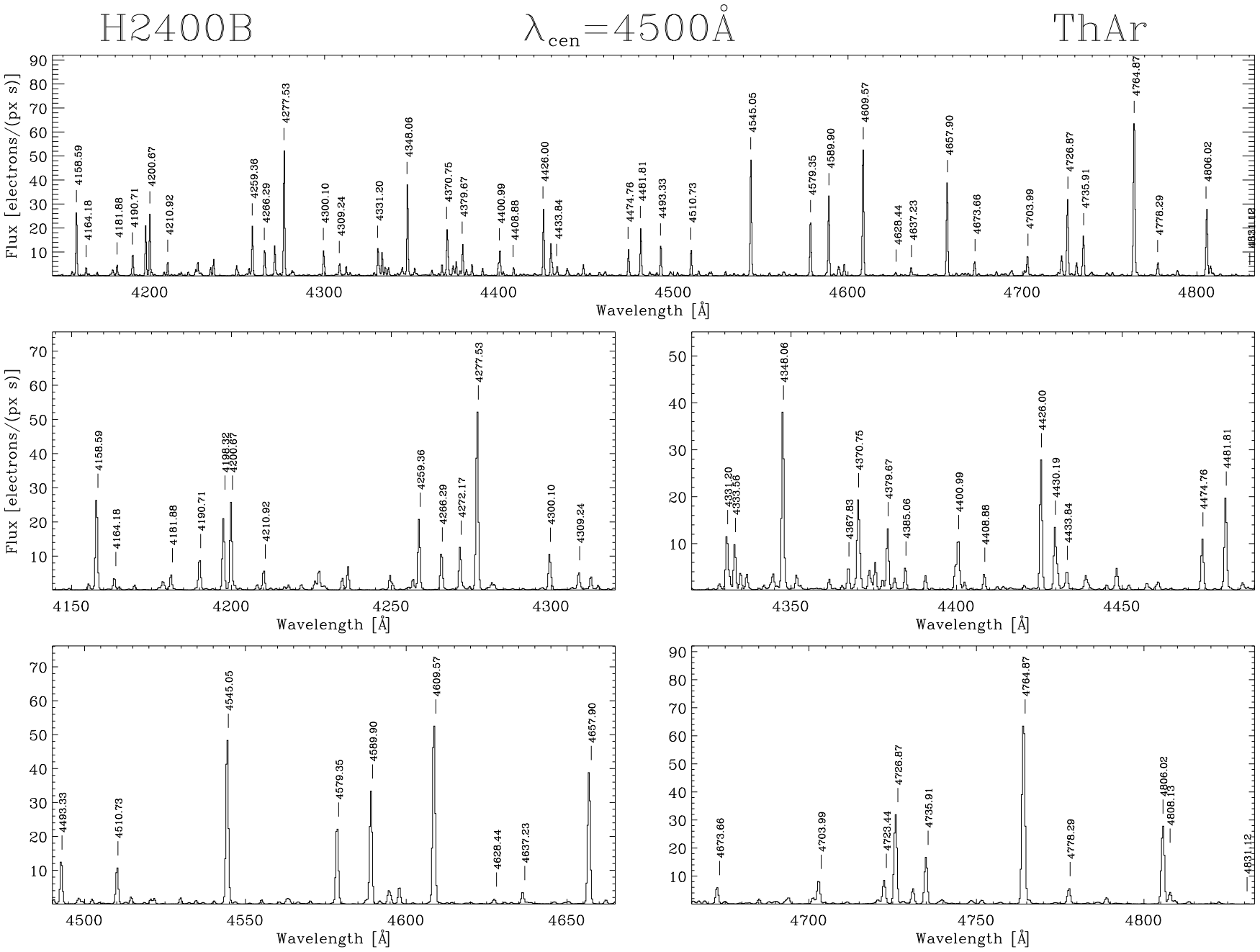
H2400B λ_{cen}=4500Å He



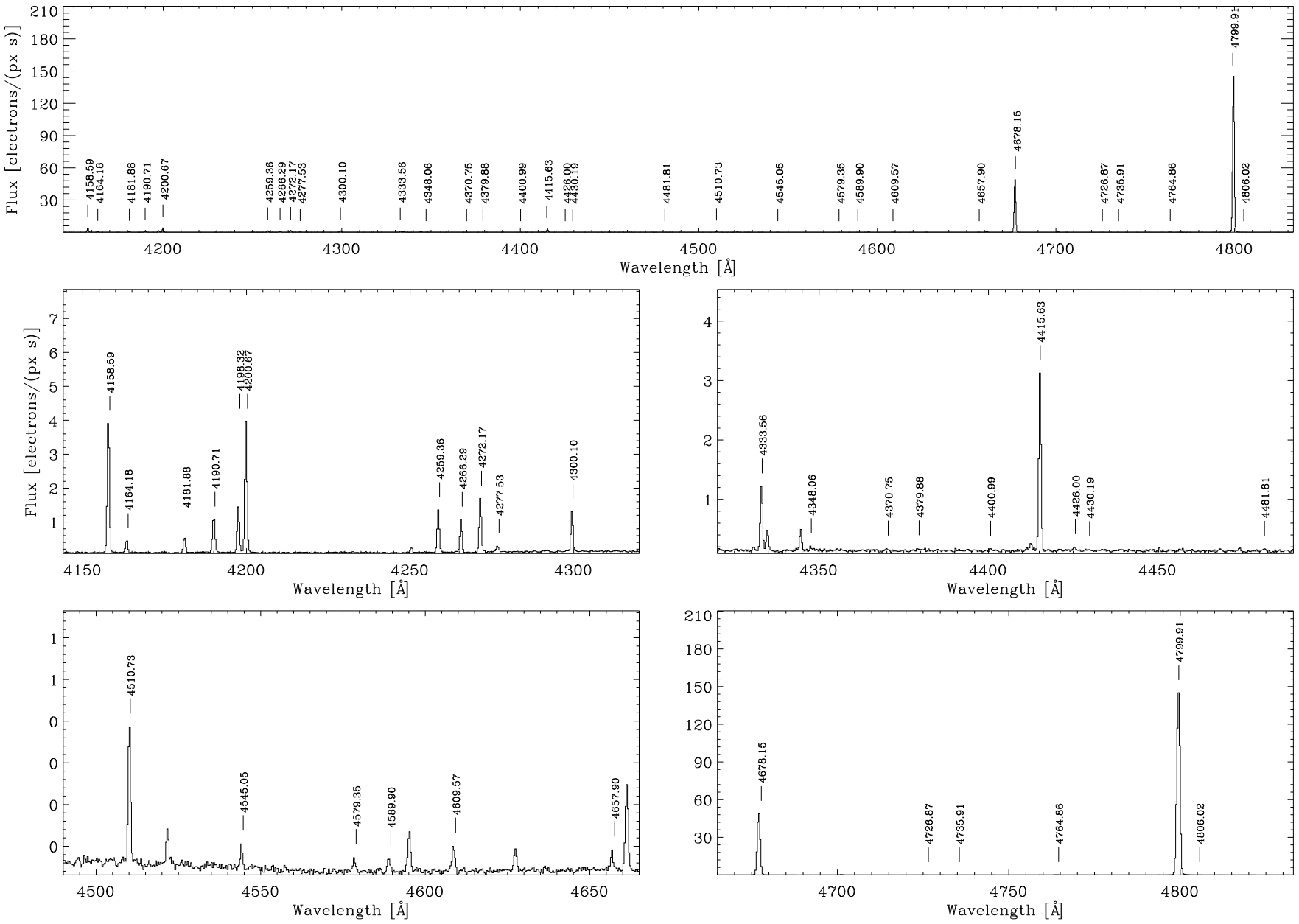

H2400B
λcen=4500Å
Ne
Flux [electrons/(px s)]
Wavelength [Å]
4219.76
4231.60
4275.11
4290.40
4379.50
4391.94
4397.94
4413.20
4428.54
4537.70
4656.39
4704.40
4715.35
4752.73
4821.92
4827.34
4409.30
4430.90
4818.40

H2400B λ_{cen}=4500Å Hg




H2400B
λcen=4500Å
ThAr
Flux [electrons/(px s)]
Wavelength [Å]
4158.59
4164.18
4181.88
4190.71
4200.67
4210.92
4259.36
4266.29
4277.53
4300.10
4309.24
4331.20
4348.06
4370.75
4379.67
4400.99
4408.88
4426.00
4433.84
4474.78
4481.81
4493.33
4510.73
4545.05
4579.35
4589.90
4609.57
4628.44
4637.23
4657.90
4673.66
4703.99
4726.87
4735.91
4764.87
4778.29
4806.02
4831.12
4158.59
4164.18
4181.88
4190.71
4198.32
4200.87
4210.92
4259.36
4266.29
4272.17
4277.53
4300.10
4309.24
4331.20
4333.56
4348.06
4367.83
4370.75
4379.67
4385.06
4400.99
4408.88
4426.00
4430.19
4433.84
4474.76
4481.81
4493.33
4510.73
4545.05
4579.35
4589.90
4609.57
4628.44
4637.23
4657.90
4673.66
4703.99
4723.44
4726.87
4735.91
4764.87
4778.29
4806.02
4808.13
4831.12

H2400B λ_{cen}=4500Å Cd



Appendix 1: Laboratory spectral lines

For Hg and ThAr the lines were taken from the SALT Longslit Line Atlas[7]. The labratory lines for He, Ne, and Cd were extracted from the NIST[8] database. Lines with low relative Intensity were removed from the NIST data.

He

Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion
3819.607	HeI	4026.191	HeI	4713.145	HeI	5047.738	HeI	7065.188	HeI
3888.646	HeI	4387.928	HeI	4921.931	HeI	5875.621	HeI	7281.349	HeI
3964.729	HeI	4471.479	HeI	5015.68	HeI	6678.151	HeI		

Ne

Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion
3323.75	NeII	4537.704	NeI	5355.3	NeI	6532.882	NeI	8544.7	NeI
3334.87	NeII	4656.394	NeI	5360.012	NeI	6598.953	NeI	8571.353	NeI
3369.87	NeI	4704.395	NeI	5400.562	NeI	6652.09	NeI	8582.91	NeI
3378.28	NeII	4715.347	NeI	5412.649	NeI	6678.276	NeI	8591.259	NeI
3417.903	NeI	4752.732	NeI	5418.558	NeI	6717.043	NeI	8634.647	NeI
3447.703	NeI	4818.4	NeI	5433.651	NeI	6929.467	NeI	8647.04	NeI
3454.195	NeI	4821.924	NeI	5656.659	NeI	7024.05	NeI	8654.384	NeI
3466.579	NeI	4827.344	NeI	5662.549	NeI	7032.413	NeI	8681.922	NeI
3472.571	NeI	4837.312	NeI	5689.816	NeI	7059.107	NeI	8704.113	NeI
3501.216	NeI	4957.033	NeI	5719.225	NeI	7173.939	NeI	8771.659	NeI
3520.472	NeI	5005.159	NeI	5748.298	NeI	7245.167	NeI	8780.622	NeI
3568.53	NeII	5031.35	NeI	5764.419	NeI	7438.899	NeI	8783.754	NeI
3593.56	NeI	5037.751	NeI	5804.098	NeI	7472.438	NeI	8853.867	NeI
3600.169	NeI	5074.2	NeI	5820.156	NeI	7488.871	NeI	8865.306	NeI
3664.112	NeII	5080.385	NeI	5852.488	NeI	7535.774	NeI	8919.499	NeI
3694.197	NeII	5116.503	NeI	5881.895	NeI	7544.044	NeI	8988.58	NeI
3709.64	NeII	5122.3	NeI	5902.462	NeI	7839.06	NeI	9148.68	NeI
3713.084	NeII	5144.97	NeI	5944.834	NeI	7927.11	NeI	9201.76	NeI
3727.08	NeII	5158.902	NeI	5965.471	NeI	7937.01	NeI	9220.05	NeI
3766.29	NeII	5188.612	NeI	5975.534	NeI	7943.181	NeI	9226.67	NeI
3777.16	NeII	5193.224	NeI	6029.997	NeI	8082.458	NeI	9275.53	NeI
3829.77	NeII	5203.895	NeI	6074.338	NeI	8118.549	NeI	9300.85	NeI
4219.76	NeII	5208.865	NeI	6096.163	NeI	8128.908	NeI	9310.58	NeI
4231.6	NeII	5222.351	NeI	6128.451	NeI	8136.406	NeI	9313.98	NeI
4275.11	NeI	5234.028	NeI	6143.063	NeI	8259.379	NeI	9326.52	NeI
4290.4	NeII	5274.039	NeI	6163.594	NeI	8266.079	NeI	9373.28	NeI
4379.5	NeII	5280.085	NeI	6217.281	NeI	8300.326	NeI	9425.38	NeI
4391.94	NeII	5298.19	NeI	6266.495	NeI	8365.749	NeI	9459.21	NeI
4397.94	NeII	5304.758	NeI	6304.789	NeI	8377.606	NeI	9486.68	NeI
4409.3	NeII	5326.396	NeI	6334.428	NeI	8418.427	NeI	9534.167	NeI
4413.2	NeII	5330.778	NeI	6382.992	NeI	8463.37	NeI	9547.4	NeI
4428.54	NeII	5341.094	NeI	6402.246	NeI	8484.45	NeI	9665.424	NeI
4430.9	NeII	5349.204	NeI	6506.528	NeI	8495.359	NeI		

Hg

Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion
3125.67	HgI	4916.07	HgI	6598.684	ArI	7272.935	ArI	8103.693	ArII
3131.55	HgI	5017.162	ArII	6604.853	ArI	7300.306		8115.311	ArI
3341.48	HgI	5460.74	HgI	6677.278	ArI	7353.293	ArI	8264.523	ArI
3606.522	ArI	5769.6	HgI	6716.423	HgI	7372.118	ArI	8408.21	ArI
3649.832	ArI	5790.66	HgI	6752.83	ArI	7383.98	ArI	8424.647	ArI
3650.15	HgI	5882.623	ArI	6871.286	ArI	7503.868	ArI	8521.442	ArI
3654.84	HgI	5942.667	ArI	6907.521	HgI	7514.651	ArI	8667.944	ArI

[7] http://pysalt.salt.ac.za/lineatlas/lineatlas.html
[8] http://physics.nist.gov/PhysRefData/ASD/lines_form.html

3663.28	HgI	6032.124	ArI	6965.426	ArI	7635.106	ArI	8716.654	
3901.87	HgI	6072.72	HgI	7030.252	ArI	7723.76	ArI	9122.967	ArI
3906.37	HgI	6098.8	ArI	7067.217	ArI	7944.66	HgII	9224.499	ArI
4046.56	HgI	6145.439	ArI	7068.73	ArI	7948.176	ArI		
4077.83	HgI	6234.393	HgI	7081.894	HgI	8006.157	ArI		
4348.063	ArII	6384.714	ArI	7091.861	HgI	8014.786	ArI		
4358.33	HgI	6416.304	ArI	7147.041	ArI	8093.126			

Cd

Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion
3080.822	CdI	4259.362	ArI	5162.285	ArI	6483.083	ArII	7635.106	ArI
3133.167	CdI	4266.286	ArI	5187.746	ArI	6538.112	ArI	7723.761	ArI
3250.33	CdII	4272.169	ArI	5268.01	CdII	6567.65	CdII	7724.207	ArI
3252.524	CdI	4277.528	ArII	5271.6	CdII	6604.853	ArI	7891.075	ArI
3261.055	CdI	4300.101	ArI	5337.48	CdII	6638.22	ArII	7948.176	ArI
3403.652	CdI	4333.561	ArI	5378.13	CdII	6643.697	ArII	8006.157	ArI
3466.2	CdI	4348.064	ArII	5381.89	CdII	6666.359	ArII	8014.786	ArI
3467.655	CdI	4370.753	ArII	5451.652	ArI	6677.282	ArI	8053.308	ArI
3491.536	ArII	4379.667	ArII	5495.874	ArI	6684.292	ArII	8066.99	CdII
3509.778	ArII	4379.881	ArII	5558.702	ArI	6725.78	CdII	8103.693	ArI
3514.387	ArII	4400.986	ArII	5572.541	ArI	6752.834	ArI	8103.693	ArII
3545.595	ArII	4415.63	CdII	5606.733	ArI	6759.19	CdII	8115.311	ArI
3554.306	ArI	4426.001	ArII	5650.704	ArI	6766.612	ArI	8200.309	CdI
3559.508	ArII	4430.189	ArII	5739.52	ArI	6778.116	CdI	8264.522	ArI
3561.03	ArII	4481.81	ArII	5843.3	CdII	6861.269	ArII	8392.27	ArI
3576.615	ArII	4510.733	ArI	5860.31	ArI	6871.289	ArI	8408.21	ArI
3588.44	ArII	4545.052	ArII	5880.22	CdII	6888.174	ArI	8424.648	ArI
3606.522	ArI	4579.349	ArII	5882.624	ArI	6937.664	ArI	8521.442	ArI
3610.508	CdI	4589.898	ArII	5888.584	ArI	6965.431	ArI	8605.776	ArI
3612.873	CdI	4609.567	ArII	5912.085	ArI	7030.251	ArI	8667.944	ArI
3614.453	CdI	4657.901	ArII	5928.813	ArI	7067.218	ArI	8771.86	ArII
3850.581	ArII	4678.149	CdI	6032.127	ArI	7068.736	ArI	8849.91	ArI
3946.097	ArII	4726.868	ArII	6043.223	ArI	7107.478	ArI	9017.592	ArII
3948.979	ArI	4735.905	ArII	6052.723	ArI	7125.82	ArI	9075.394	ArI
4013.856	ArII	4764.864	ArII	6059.372	ArI	7147.042	ArI	9106.562	ArII
4029.12	CdII	4799.912	CdI	6099.142	CdI	7158.839	ArI	9122.967	ArI
4042.893	ArII	4806.02	ArII	6111.495	CdI	7206.98	ArI	9194.638	ArI
4044.418	ArI	4847.81	ArII	6114.923	ArII	7237.01	CdII	9224.499	ArI
4052.921	ArII	4879.863	ArII	6145.441	ArI	7265.172	ArI	9291.531	ArI
4072.004	ArII	4881.72	CdII	6172.278	ArII	7272.936	ArI	9292.0	CdI
4079.573	ArII	4889.042	ArII	6173.096	ArI	7284.38	CdII	9354.22	ArI
4103.912	ArII	4904.751	ArII	6212.503	ArI	7311.716	ArI	9601.925	ArII
4112.815	ArII	4933.209	ArII	6296.872	ArI	7316.005	ArI	9657.786	ArI
4131.723	ArII	4965.079	ArII	6307.657	ArI	7345.67	CdI	9758.637	ArII
4134.77	CdII	4972.16	ArII	6325.166	CdI	7353.293	ArI	9773.567	ArII
4141.49	CdII	5009.334	ArII	6330.013	CdI	7372.118	ArI	9783.079	ArII
4158.59	ArI	5017.163	ArII	6354.72	CdII	7383.98	ArI	9784.503	ArI
4164.18	ArI	5025.5	CdII	6359.98	CdII	7392.98	ArI	9803.69	ArII
4181.884	ArI	5062.037	ArII	6384.717	ArI	7412.337	ArI	9849.458	ArII
4190.713	ArI	5085.822	CdI	6416.307	ArI	7435.368	ArI	9854.061	ArII
4198.317	ArI	5141.783	ArII	6438.47	CdI	7503.869	ArI	9993.863	ArII
4200.674	ArI	5145.308	ArII	6464.94	CdII	7514.652	ArI		

ThAr

Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion	Line [Å]	Ion
3281.701	ArII	4433.838	ArII	5343.581	ThI	6212.503	ArI	7107.48	ArI
3382.132	ArII	4474.759	ArII	5378.836	ThI	6215.939	ArI	7125.82	ArI
3491.536	ArII	4481.81	ArII	5386.525	ArII	6224.528	ThI	7147.04	ArI
3545.843	ArII	4493.334	ThI	5390.466	ThII	6234.856	ThI	7168.9	ThI

3559.508	ArII	4510.73	ArI	5394.852		6243.12	ArII	7206.98	ArI
3561.03	ArII	4545.052	ArII	5407.345	ArII	6261.418	ThI	7218.054	ThI
3576.557	ThI	4579.349	ThI	5410.475	ArI	6307.657	ArI	7272.94	ArI
3588.44	ArII	4589.898	ArII	5417.486	ThI	6327.278	ThI	7311.716	ArI
3612.427	ThI	4609.567	ArII	5451.652	ArI	6342.86	ThI	7315.067	ThI
3659.529	ArI	4628.441	ArI	5495.874	ArI	6348.625		7353.29	ArI
3719.435	ThI	4637.232	ArII	5498.985		6355.911	ThI	7372.12	ArI
3729.309	ArII	4657.901	ArII	5509.994	ThI	6369.575	ArI	7383.98	ArI
3737.889	ArII	4673.661	ThI	5539.262	ThI	6376.931	ThI	7435.37	ArI
3765.27	ArII	4703.99	ThI	5548.176	ThI	6384.72	ArI	7471.16	ArI
3780.84	ArII	4723.438	ThI	5558.702	ArI	6411.9	ThI	7484.33	ArI
3803.172	ArII	4726.868	ArII	5587.026	ThI	6416.307	ArI	7503.869	ArI
3809.456	ArII	4735.906	ArII	5595.064	ThI	6457.28	ThI	7514.651	ArI
3828.385	ThI	4764.865	ArII	5606.733	ArI	6462.614	ThI	7567.74	ThI
3839.746	ThII	4778.294	ThI	5615.32	ThI	6466.553	ArI	7585.792	ThI
3850.58	ArII	4806.021	ArII	5639.746	ThII	6483.083	ArII	7635.11	ArI
3868.528	ArII	4808.134	ThI	5650.704	ArI	6490.738	ThI	7724.206	ArI
3925.718	ArII	4831.121	ThI	5700.918	ThII	6512.364	ThI	7788.937	ThI
3928.623	ArII	4840.843	ThI	5707.103	ThII	6531.34	ThI	7817.771	ThI
3932.546	ArII	4847.81	ArII	5720.183	ThI	6538.112	ArI	7847.54	ThI
3967.392	ThI	4879.864	ArII	5760.551	ThI	6554.16	ThI	7891.08	ArI
3994.792	ArII	4889.042	ArII	5789.645	ThI	6577.215	ThI	7948.176	ArI
4019.129	ThII	4894.955	ThI	5804.141	ThI	6583.906	ThI	7978.97	ThI
4030.842	ThI	4904.751	ArII	5834.264	ArI	6588.54	ThI	8006.157	ArI
4036.047	ThI	4919.816	ThII	5852.774		6593.94	ThI	8014.786	ArI
4043.39	ThI	4933.208	ArII	5860.31	ArI	6604.85	ArI	8053.31	ArI
4052.921	ArII	4939.642	ThI	5882.624	ArI	6638.22	ArII	8103.693	ArI
4072.005	ArII	4965.08	ArII	5888.584	ArI	6639.74	ArII	8115.311	ArI
4080.636	ArII	5002.097	ThI	5891.451	ThI	6643.7	ArII	8143.139	ThI
4103.912	ArII	5009.334	ArII	5912.09	ArI	6662.27	ThI	8186.914	ThI
4112.815	ArII	5017.163	ArII	5925.893	ThII	6666.359	ArII	8264.523	ArI
4131.724	ArII	5039.23	ThI	5928.812	ArI	6677.28	ArI	8330.45	ThI
4158.591	ArI	5044.719	ThI	5973.665	ThI	6684.29	ArII	8408.21	ArI
4164.18	ArI	5062.037	ArII	5989.044	ThII	6719.219	ArI	8424.648	ArI
4181.884	ArI	5067.974	ThI	5994.129	ThI	6727.459	ThI	8446.509	ThI
4190.713	ArI	5090.495	ArII	6005.725	ArI	6752.833	ArI	8478.36	ThI
4198.317	ArI	5100.442	ArII	6007.072	ThI	6756.453	ThI	8521.44	ArI
4200.675	ArI	5115.044	ThI	6021.036	ThI	6766.611	ArI	8573.122	ThI
4210.923	ThI	5125.95	ThI	6025.15	ArI	6778.313	ThI	8605.78	ArI
4259.36	ArI	5141.783	ArII	6032.127	ArI	6780.413	ThI	8620.46	ArI
4266.286	ArI	5145.308	ArII	6037.698	ThI	6829.036	ThI	8667.94	ArI
4272.169	ArI	5151.612	ThI	6043.223	ArI	6834.925	ThI	8709.236	ThI
4277.53	ArII	5154.243	ThI	6049.075	ArII	6861.269	ArII	8748.033	ThI
4300.1	ArI	5158.604	ThI	6098.803	ArI	6871.29	ArI	8761.686	ArI
4309.239	ArII	5176.961	ThI	6105.635	ArI	6874.754	ThI	8967.641	ThI
4331.199	ArII	5187.746	ArI	6114.92	ArII	6879.583	ArI	9048.252	ThI
4333.561	ArI	5199.164	ThI	6121.879	ArI	6888.174	ArI	9122.967	ArI
4348.064	ArII	5211.23	ThI	6145.441	ArI	6911.23	ThI	9194.638	ArI
4367.831	ArII	5219.11	ThI	6151.993	ThI	6937.66	ArI	9218.986	ArII
4370.75	ArII	5231.16	ThI	6155.239	ArI	6943.61	ThI	9224.498	ArI
4379.666	ArII	5247.654	ThII	6169.822	ThI	6965.43	ArI	9291.532	ArI
4385.057	ArII	5258.36	ThI	6172.28	ArII	6989.66	ThI	9354.218	ArI
4400.99	ArII	5266.71	ThI	6182.622	ThI	7000.806	ThI	9657.786	ArI
4408.882	ThI	5297.743	ThI	6191.906	ThI	7018.57	ThI		
4426.001	ArII	5311.982	ArII	6198.223	ThI	7030.25	ArI		
4430.189	ArII	5326.976	ThI	6203.493	ThI	7067.218	ArI		

Appendix 2: Atlas plotting code

This appendix gives the IDL code for plotting the arc spectra with the labels and ticks on some identified lines. The inputs are two files, one containing the wavelengths and the fluxes, and another one with the wavelengths of the lines to be marked. The output is a formatted pdf file showing five plots: one covers the entire spectrum and the other four zoom in four splits, in order to facilitate the line identification.

```
;*******************************************************
; idl_plot.pro procedure
; July 2015
; Edited for WYFOS: Ronny Errmann
; original Authors: Hassan Fathivavsari, Javier Mendez and Liam Hardy:
;   http://www.ing.iac.es/astronomy/observing/manuals/ps/tech_notes/tn133.pdf
;*******************************************************
; It requires files <grating>_<cenwave>_<arc>_<extension>.txt
; and ticksall_<grating>_<cenwave>_<arc>_<extension>.txt in the
; same directory
;*******************************************************
; <grating>_<cenwave>_<arc>_<extension>.txt is the 1-D ascii file
; containing two columns: wavelength and flux.
;*******************************************************
; ticksall_* contains all the lines to mark. Example:
; 5100      He II
; 5105.54   Hg+Ar
; 5330.78                (text is optional and not used)
;*******************************************************
; Execute as follows:
; IDL> idl_plot,grating,cenwave,arc    ,arctext,exptime,extension ,5 wavelengthes, line
     offset
; the wavelength are: begin, 3x splitting, end
; IDL> idl_plot,'R900V','6500' ,'hgar','Hg+Ar',10      ,'p1234578
     ',3500,5800,6600,8500,10500,0
;*******************************************************
pro idl_plot,gra,cenw,arc,arctext,exptime,extension,x1,x2,x3,x4,x5,offset

only_first_plot=0       ;to create ps files with only the first plot use
     only_first_plot=1
set_plot,'ps'
device, filename=gra + '_' + cenw + '_' + arc + '.ps',/color,bits_per_pixel=8,/
     landscape
;device,XSIZE=29.7-5.6, YSIZE=21.0-4.0, /cm, filename=gra + '_' + cenw + '_' + arc +
     '.ps',/color,bits_per_pixel=8,/landscape    ;for a4
loadct, 0

ticksfile='ticksall_' + gra + '_' + cenw + '_' + arc + '_' + extension + '.txt'
filename=gra + '_' + cenw + '_' + arc + '_' + extension + '.txt'
xran=[fix(x1),fix(x2),fix(x3),fix(x4),fix(x5)]

readcol, ticksfile, ticksdata
n=file_lines(filename)
data=fltarr(2,n)
get_lun,lun
openr,lun,filename
readf,lun,data
free_lun,lun
wave=data(0,*)
flux=data(1,*)/exptime

nf = n_elements(flux)
nt = n_elements(ticksdata)

ytitl='Flux [electrons/(px s)]'
;*************** Subwindow Settings *************
for xk = 0,4*(1-only_first_plot) do begin       ;4 normally, otherwise only first plot
  if xk eq 0 then begin
    !Y.MARGIN=[4,6]      ;distance to paper edges
    !p.multi=[0,1,3]
    xb=xran(0)   ;global min
    xe=xran(4)   ;global max
  endif else begin
    !Y.MARGIN=[4,2]
  endelse
  if xk eq 1 then begin
    !p.multi=[4,2,3]
    xb=xran(0)
    xe=xran(1)   ;first subplot
  endif
  if xk eq 2 then begin
    !p.multi=[3,2,3]
    xb=xran(1)
```

```
69        xe=xran(2);second subplot
70        ytitl=''
71      endif
72      if xk eq 3 then begin
73        !p.multi=[2,2,3]
74        xb=xran(2)
75        xe=xran(3);third subplot
76      endif
77      if xk eq 4 then begin
78        !p.multi=[1,2,3]
79        xb=xran(3)
80        xe=xran(4)   ;fourth subplot
81        ytitl=''
82      endif
83
84      charsize=0.5
85      xycharsize=1.5
86
87      if only_first_plot eq 1 then begin
88        !p.multi=0
89         !Y.MARGIN=[4,3]
90        DEVICE, YSIZE=7.7, /cm
91        ytitl='Flux_[electrons/px]'
92        charsize=0.6
93        xycharsize=1.0
94      endif
95
96    ;******************** Ticks Mark ******************************
97      dx = xe - xb            ;wavelength range
98      ;xx = 0.003666141 * dx          ;minimum distance of two lines in plot
99      if xk eq 0 then begin
100       xx = 0.006 * dx               ;plotting offset
101       if strmatch(arc,'thar') EQ 1 then xxn = 0.008 * dx else xxn = 0.006 * dx ;for ThAr
              wider sep till next line
102     end else begin
103       xx = 0.01 * dx        ;plotting offset
104       xxn = 0.01 * dx       ;minimum distance of the text of two lines
105     end
106
107     t = fltarr(nt,3)        ;empty nt entry array
108     for k=0,nt-1 do begin
109       if (ticksdata(k) lt xb) or (ticksdata(k) gt xe) then continue        ;only in range
110       msb = value_locate(wave, ticksdata(k)+offset)
111       if msb eq -1 then continue            ;if outside of the flux range
112       t(k,0)=total(flux(max([msb-2,0]):min([msb+2,nf-1])))          ;flux arround the line
113       t(k,1)=ticksdata(k)
114       t(k,2)=max(flux(max([msb-5,0]):min([msb+5,nf-1])))
115     endfor
116     tsort=reverse(sort(t(*,0)))     ;tsort contains indexe starting with the highest flux
117     ticks = fltarr(nt)
118     m = 1
119     ticks(0)=t(tsort(0),1)          ;wavelength with highest flux
120     maxi=t(tsort(0),2)              ;highest flux
121     for k=1,nt-1 do begin           ;adding the other wavelengthes, starting with second
          highest flux and only, if no line close to it is already added
122       if t(tsort(k),1) lt 1 then break     ;stop, if no more catalog data is available
123       add=1
124       for l=0,m-1 do begin                 ;search the already added lines
125         if abs(t(tsort(k),1)-ticks(l)) LE xxn then begin  ;found a close line
126           add=0
127           break             ; -> don't use
128         endif
129       endfor
130       if add eq 1 then begin        ;add line
131         ticks(m) = t(tsort(k),1)   ;add the line to the ticks
132         maxi=max([maxi,t(tsort(k),2)])
133         m = m + 1
134       endif
135     endfor
136     ticks = ticks(sort(ticks(0:m-1)))       ;sort by wavelength and ignore the zeros
137     print,ticks
138
139   ;*************** Borders ************************
140     msb0=value_locate(wave, xb+1)
141     msb1=value_locate(wave, xe-1)
142     print, msb0
143     print, msb1
144     maxi=max([maxi,max(flux(msb0:msb1))/1.4])       ;if highest line has no tick it
          creates unused space
145     mini=min(flux(msb0:msb1))
146     std=stddev(flux(msb0:msb1))
147     ;print,std,mini,maxi,maxi-mini
```

```
148     if (maxi-mini lt 15*std) and (maxi-mini lt 10) then maxi=mini+min([10,15*std])
149
150   ;*************** Ticks ************************
151     xtickint=1000
152     if xe - xb lt 2000 then xtickint=500
153     if xe - xb lt 1000 then xtickint=200
154     if xe - xb lt 500 then xtickint=100
155     if xe - xb lt 200 then xtickint=50
156     if xe - xb lt 100 then xtickint=20
157     if xk eq 0 then begin
158       xtickint=1000
159       if xe - xb lt 4000 then xtickint=500
160       if xe - xb lt 2000 then xtickint=200
161       if xe - xb lt 1000 then xtickint=100
162       if xe - xb lt 400 then xtickint=50
163     endif
164
165     ytickint=(maxi-mini)*1.45/8             ;unrounded
166     rounded=10^round(alog10(ytickint)-0.3)
167     ytickint = round(ytickint/(rounded+0.0))*rounded       ;rounded
168     ;ytickint = 30000000
169
170   ;*************** Plot *************************
171     plot, wave, flux, psym=10, xrange=[xb, xe],/xstyle, yrange=[mini, maxi*1.45],/ystyle
          , thick=0.65*(1+only_first_plot), ytickinterval=ytickint, ytickformat='(I8)',
          yticklen=0.01, yminor=5, ycharsize=xycharsize, ytitle=ytitl, xtickinterval=
          xtickint, xticklen=0.02, xminor=10, xcharsize=xycharsize, xtitle='Wavelength
          [!6!sA!r!u!9 %!6!n]'
172
173     if ticks(0) gt 1 then begin    ;only if catalog data available
174       for i=0,m-1 do begin
175         msb = value_locate(wave, ticks(i)+offset) -0
176         y = max(flux(max([msb-5,0]):min([msb+5,nf-1])))    ;maximum flux around line to
              plot tick marks right
177         plots, [ticks(i)-0.2*xx+offset,ticks(i)-0.2*xx+offset], [y+0.07*maxi, y+0.15*
              maxi], thick=0.8      ;tick marks
178         xyouts,ticks(i)+0.5*xx+offset,y+0.08*maxi,ticks(i), charsize=charsize,
              alignment=0.2 ,orientation=90, charthick=0.8 ,/data        ;tick text
179       endfor
180     endif
181
182   endfor  ;subplots
183
184   if only_first_plot eq 1 then begin
185   xyouts,[2000, 10000, 20000], [7000, 7000, 7000], [gra,'!7k!X!Icen!N='+cenw+'!6!sA!r!u
        !9 %!6!n',arctext], CHARSIZE = 2, /DEVICE  ;Title text
186   endif else begin
187   xyouts,[2000, 10000, 20000], [17000, 17000, 17000], [gra,'!7k!X!Icen!N='+cenw+'!6!sA!r
        !u!9 %!6!n',arctext], CHARSIZE = 2, /DEVICE        ;Title text
188   endelse
189
190   device, /close
191
192   if only_first_plot ne 1 then begin
193     spawn, 'rm -f out.pdf'
194     spawn, 'ps2pdf13 ' + gra + '_' + cenw + '_' + arc + '.ps out.pdf'
195     ;spawn, 'ps2pdf13 -sPAPERSIZE=a4 ' + gra + '_' + cenw + '_' + arc + '.ps out.pdf'
              ;leaves a too big border on the left
196     spawn, 'rm -f '+ gra + '_' + cenw + '_' + arc + '_' + extension + '.pdf'
197     spawn, 'pdftk out.pdf cat 1-endN output ' + gra + '_' + cenw + '_' + arc + '_' +
          extension +'.pdf'     ;rotates the pdf by 0deg
198     spawn, 'mv ' + gra + '_' + cenw + '_' + arc + '.ps ' + gra + '_' + cenw + '_' + arc
          + '_' + extension +'.ps'
199     print,'Output saved to ' + gra + '_' + cenw + '_' + arc + '_' + extension + '.pdf'
200   endif
201
202   end
```

Appendix 3: Preparation for the atlas plotting

This code combines the flux from fibers by running a median on the "iraf_*.fit" files, produced by the WYFFOS pipeline. As each fiber has a different throughput the fibers were scaled and a 2σ clipping was applied before combining them. To determine the scaling factor, the scaling ratio against the median flux for each pixel was determined. These factors were combined by using the weighted average with the median flux as weights.

Additionally files necessary to create this document are also prepared: (1) The catalog with

the line labels for each individual setup is created. (2) Header information are collected and put together in a table (**table.tex**), which can be imported into a *tex* file. (3) Furthermore the lists for all lines plotted in the graphs are written in a text file (**arc_catalogs.txt**). (4) A script to be run in *idl* is created and executed (**run_in_idl.pro**).

```
1  import os
2  import sys
3  import time
4  import numpy
5  import pyfits
6  from operator import itemgetter
7  #copy data into plot folder
8  #find /data/ronny/data_af2_proj/ -name "iraf_*.fit" -exec cp -puv {} . \;
9  #find /obsdata/outgoing/ldp/LAMPS/ -name "iraf_*.fit" -exec cp -puv {} . \;
10 #run: python ../combine_fibers.py iraf_*
11
12 fibers=[1,161]  #[77,85]        #which fiber range should be used to determine the
       median flux for the setup
13 comborders=True          #False does not create the single order files for idl_plot (
       again). Only use it to save computing time
14 sort_out3=False          #True doesn't plot graphs with less than 3 lines
15 sort_out_not_usable=True        #True doesn't plot graphs from the not_usable list
16 use_ignore_files=True   #True means the list "ignore_files" is applied and the files
       in the list are not used
17 borders=True             #True means that the min and max wavelength from dispersions
       is applied to have all lamps of a configuration equal
18 min_flux=1       #if median flux in a fiber is lower, the fiber is not used
19 lampfolder='/home/ronny/Programs/WYFFOS_pipeline_v3.0.3/control_standards/'      #Where
       are the catalog files located?
20 #as ND filter is missing in header create a file 'ndfilter.lst' with the columns
       runnumber and header keyword, e.g. "2282280 WYC_clear1"
21
22 keywords=[]
23 keywords.append('EXPTIME')        #IDS,WYFFOS
24 keywords.append('WYFGRAT')        #WYFFOS
25 keywords.append('CENWAVE')        #IDS,WYFFOS
26 keywords.append('WYFORDER')       #WYFFOS
27 keywords.append('WYFCLAMP')       #WYFFOS
28 keywords.append('LAMP')           #WYFFOS
29
30 dispersions=[]  #grating, cenwave, dispersion, plot borders (dispersion for 2x2
       binning from webpage for Red+4, or interpolated to write into tex table)
31 dispersions.append(['R158B' , '3600', '6.4', '','']) #'3.2'])#
       binning 1x1
32 dispersions.append(['R158B' , '4500', '6.4']) #'3.2'])
33 dispersions.append(['R158B' , '4900', '6.4']) #'3.2'])
34 dispersions.append(['R158R' , '6500', '6.4']) #'3.2'])
35 dispersions.append(['R158R' , '7500', '6.4']) #'3.2'])
36 dispersions.append(['R300B' , '4000', '3.6']) #'1.8'])
37 dispersions.append(['R300B' , '4500', '3.6', '3000','8305']) #'1.8'])
38 dispersions.append(['R300B' , '5300', '3.6']) #'1.8'])
39 dispersions.append(['R316R' , '6500', '3.4']) #'1.7'])
40 dispersions.append(['R316R' , '7500', '3.4', '4418','10000']) #'1.7'])
41 dispersions.append(['R600B' , '3900', '1.8']) #'0.89'])
42 dispersions.append(['R600B' , '4000', '1.8']) #'0.89'])
43 dispersions.append(['R600B' , '5000', '1.8', '3351','6682']) #'0.89'])
44 dispersions.append(['R600R' , '6500', '1.8', '4834','8177']) #'0.89'])
45 dispersions.append(['R600R' , '7000', '1.8']) #'0.89'])
46 dispersions.append(['R600R' , '8000', '1.8', '6371','9668']) #'0.89'])
47 dispersions.append(['R1200B', '4000', '0.86', '3177','4815']) #'0.43'])
48 dispersions.append(['R1200B', '4500', '0.84', '3678','5298']) #'0.42'])
49 dispersions.append(['R1200B', '5000', '0.81', '4193','5804']) #'0.41'])
50 dispersions.append(['R1200B', '5500', '0.79']) #'0.39'])
51 dispersions.append(['R1200R', '5500', '0.79', '4665','6292']) #'0.39'])
52 dispersions.append(['R1200R', '6000', '0.76', '5169','6746']) #'0.38'])
53 dispersions.append(['R1200R', '6500', '0.74', '5681','7266']) #'0.37'])
54 dispersions.append(['R1200R', '7000', '0.71', '6195','7748']) #'0.36'])
55 dispersions.append(['R1200R', '7200', '0.70']) #'0.35'])
56 dispersions.append(['R1200R', '7500', '0.69', '6711','8229']) #'0.34'])
57 dispersions.append(['R1200R', '8000', '0.66', '7208','8708']) #'0.33'])
58 dispersions.append(['R1200R', '8200', '0.65']) #'0.33'])
59 dispersions.append(['R1200R', '8500', '0.64', '7724','9184']) #'0.32'])
60 dispersions.append(['R1200R', '9000', '0.61']) #'0.31'])
61 dispersions.append(['R1200R', '9350', '0.59', '8601','10000']) #'0.30'])
62 dispersions.append(['H1800V', '5120', '0.44']) #'0.22'])
63 dispersions.append(['H1800V', '5500', '0.44']) #'0.22'])
64 dispersions.append(['H2400B', '4000', '0.38', '3645','4343']) #'0.19'])
65 dispersions.append(['H2400B', '4500', '0.38', '4144','4833']) #'0.19'])
66 dispersions.append(['','','']) #if no matching dispersion fill dispersion in tex
       table with empty entry
67
68 replace_cenwave=[]       #the cenwave in the header is not exact, with this list values
```

```
        are replaced
69  replace_cenwave.append(['3499','3500'])
70  replace_cenwave.append(['3501','3500'])
71  replace_cenwave.append(['9004','9000'])
72  replace_cenwave.append(['9005','9000'])
73  replace_cenwave.append(['9301','9300'])
74  replace_cenwave.append(['9351','9350'])
75  replace_cenwave.append(['9352','9350'])
76
77  ignore_files=[]
78  ignore_files.append('p2279722') #'H2400B','4500','cd','Cd','120'          saturated
79  ignore_files.append('p2279719')
80  ignore_files.append('p2271410') #'R600B','4000','he','He','60'
81  ignore_files.append('p1710557') #'R158B','4900','hgar','Hg','2' replace by new
        dedector
82
83  override_splittings=[]  #grat,cenwave,arc, wave1,wave2,wave3,wave4,wave5          #''
        for wave uses auto
84  override_splittings.append(['H2400B','4000','he',  '','3830','','','',''])
85  override_splittings.append(['H2400B','4000','hgar','','3750','3930','4060',''])
86  override_splittings.append(['R1200R','9350','ne',  '','','9290','','',''])
87  override_splittings.append(['R1200R','9350','hgar','','','','9680','',''])
88  override_splittings.append(['R158B','4500','he',  '3000','4200','5400','7200','10000'
        ])
89  override_splittings.append(['R158B','4500','ne',  '3000','5500','6800','7800','10000'
        ])
90  override_splittings.append(['R158B','4500','hgar','3000','5100','5850','7850','10000'
        ])
91  override_splittings.append(['R600R','8000','hgar','','7200','8050','','',''])
92  override_splittings.append(['R600R','8000','thar','','7245','8030','','',''])
93  override_splittings.append(['R600R','8000','cd',  '','7190','8020','8820',''])
94
95  manual_shift=[] #move the lines by how many angstom as small offset occurs
96  manual_shift.append(['ECHELLE-3','8685','hgar','0.7'])
97  manual_shift.append(['ECHELLE-5','5675','ne','-0.4'])
98  manual_shift.append(['ECHELLE-5','5400','ne','-0.5'])
99
100 not_usable=[]           #settings which should not be used for the final plot
101 not_usable.append(['H2400V','4500','ne'])       #no flux
102 not_usable.append(['H2400V','4000','he'])       #bad solution
103
104 fibers=range(fibers[0],fibers[1]+1)
105 names=[]
106 for j in range(1,len(sys.argv)):
107         name=sys.argv[j].replace('.fits','')
108         name=name.replace('.fit','')
109         if name==sys.argv[j]:   #no change -> no fits
110                 continue
111         names.append([name,sys.argv[j]])
112 if comborders==True:    #will make the spectra to text, read them and combine the
        fibers into one text-file
113         for name in names:
114                 data=[]
115                 header_table = pyfits.getheader(name[1])
116                 data_cube = pyfits.getdata(name[1], 0)  #for good files: 0==2 contains
                        flux, 1==3 contains errors?, 4?, 5?, otherwise data_cube.shape
                        ==(2,*) for 0
117                 if len(data_cube)==2:
118                         data.append(data_cube[0])
119                         print "extraction was done for only one fiber",name[0]
120                 else:
121                         print name[0],":\t",
122                         for j in [fibers,range(1,162)]:         #all fibers in case
                                the selected ones contain no data
123                             for i in j:
124                                 keyword='F%3.3i'%i+'TYPE'
125                                 if keyword in header_table.keys():
126                                     if header_table[keyword][0]<>'X':
                                            #X means 0 flux, as fiber
                                            has a problem
127                                         if numpy.median(data_cube[i
                                                -1])>min_flux:       #only
                                                if enough flux
128                                             data.append(data_cube[
                                                    i-1])
129                             if data<>[]:
130                                 break
131                         print "no data in the selected fibers:",j,", try with
                                all fibers","\t\t",
132                 print len(data),"fibers used"
133                 wavelength=[header_table['CRVAL1']]     #prepare wavelength list
134                 step=header_table['CDELT1']
```

```
135                     for i in range(1,len(data[0])):          #create wavelength list
136                             wavelength.append(wavelength[i-1]+step)
137                     data=numpy.array(data)
138
139                     if len(data)>3:          #scale the flux between fibers by determining
                            the median flux and determing the factor
140                             data_sub=[]
141                             for i in range(len(wavelength)):          #for every pixel
142                                     if numpy.median(data[:,i])>min_flux:     #if enough
                                            flux
143                                             data_sub.append(data[:,i])
144                             data_sub=numpy.array(data_sub)
145                             data_sub=numpy.transpose(data_sub)                              #
                                    subarray of data which contains flux
146                             med_flux=[]                                                     #
                                    median flux in each pixel
147                             for i in range(len(data_sub[1,:])):      #for every pixel
148                                     med_flux.append(numpy.median(data_sub[:,i]))     #
                                            determine median flux
149                             med_flux=numpy.array(med_flux)           #median flux
150                             factor=[]                                #scaling factors for
                                    each fiber
151                             for j in range(len(data_sub)):           #for every fiber
152                                     factors=numpy.array(data_sub[j,:]/med_flux)      #
                                            factors for each pixel towards median_flux
153                                     factors=numpy.where(factors>0.33, factors, 10)  #
                                            replace factors below 1/3 by 10
154                                     temp=numpy.where(factors<3)                             #get
                                            indexes where factor between 1/3 and 3 to use only
                                             these ones.
155                                     factor.append(numpy.average(factors[temp[0]],weights=
                                            med_flux[temp[0]]**2))      #get weighted average
                                            of factors for each pixel by weighting them with
                                            the median flux of that pixel
156                             factor/=numpy.median(factor)             #normalize to one by
                                    dividing through median
157                             std_factor=2*numpy.std(factor,ddof=1)    #2 times standard
                                    deviation
158                             i=0
159                             for j in range(len(data))[::-1]:                        #for every
                                    fiber backwards
160                                     if abs(factor[j]-1)<std_factor:          #median(factor
                                            )==1
161                                             data[j,:]/=factor[j]             #if in 2*std,
                                                    then scale data of this fiber
162                                     else:
163                                             data=numpy.delete(data,(j),axis=0)       #
                                                    otherwise delete fiber
164                                             i+=1
165                             if i>0:
166                                     print "deleted",i,"bad fiber(s)"
167                     file=open(name[0].replace('iraf_','')+'.txt','w')
168                     for i in range(len(wavelength)):
169                             file.write('%5.3f'%wavelength[i]+'\t%4.2f'%(numpy.median(data
                                    [:,i]))+'\n')
170                     file.close()
171     ndfilters=[]
172     if os.path.isfile('ndfilter.lst')==True:
173             file=open('ndfilter.lst')
174             for line in file:
175                     line=line.split()
176                     if len(line)<2:
177                             continue
178                     #line[1]=line[1].replace('','')
179                     line[1]=line[1].replace('WYC_clear1','')
180                     line[1]=line[1].replace('WYC_','')
181                     line[1]=line[1].replace('0','')
182                     ndfilters.append(line)
183             file.close()
184     if ndfilters==[]:
185             print "no or empty file: ndfilter.lst"
186
187     result=[]
188     arc_catalogs=[]
189     log_lines=[]
190     for name in names:       #does the important stuff to each file
191             shortname=name[0].replace('iraf_','')
192             if shortname in ignore_files and use_ignore_files==True:
193                     continue
194             spectrum=[]
195             if os.path.isfile(shortname+'.txt')==False:
196                     print shortname+'.txt  does not exist, please set "comborders" to "
```

```
                    True"'
197                 exit(1)
198         file=open(shortname+'.txt','r')
199         for line in file:
200                 line=line.split()
201                 spectrum.append([float(line[0]),float(line[1]),len(spectrum)])
202         file.close()
203         keyresult=[]
204         header_table = pyfits.getheader(name[1])
205         for keyword in keywords:
206                 if keyword in header_table.keys():
207                         temp=str(header_table[keyword])
208                         if temp.find('␣')>-1:
209                                 keyresult.append(header_table[keyword].replace('␣','')
                                        )
210                         else:
211                                 keyresult.append(header_table[keyword])
212                 else:
213                         keyresult.append('')
214         #get data for wavelength in plot
215         spectrum=numpy.array(spectrum)
216         spectrum=sorted(spectrum, key=itemgetter(0))    #to have smallest and highest
                wavelength at beginning and end, respectivly
217         spectrum=numpy.array(spectrum)
218         minwave=spectrum[0,0]
219         maxwave=spectrum[-1][0]
220         exptime='%1.1i'%float(keyresult[0])
221         if exptime=='0':
222                 exptime='1'
223         grat=keyresult[1]
224         if grat=='ECHELLE':
225                 grat=grat+'-'+keyresult[3]
226         cenwave=str(keyresult[2])
227         for line in replace_cenwave:    #replace not even numbers in cenwave
228                 if cenwave==line[0]:
229                         cenwave=line[1]
230                         break
231         arc=keyresult[4]
232         if arc=='':
233                 arc=keyresult[5]
234         if arc=='Hg':
235                 if name[0].find('p18')>-1:      #for p18 the lamp in the header is
                        wrong
236                         arc='ThAr'
237                 else:
238                         arc='HgAr'
239         compwaves=[]
240         file=open(lampfolder+'/'+arc+'.txt','r')
241         for line in file:
242                 if len(line)<5: #empty lines might be a problem
243                         continue
244                 line=line.split()
245                 if float(line[0])>minwave and float(line[0])<maxwave:
246                         temp=[arc,line[0]]
247                         if len(line)>1:
248                                 compwaves.append(line[0]+'\t'+line[1]+'\n')
249                                 temp.append(line[1].replace('␣',''))
250                         else:
251                                 compwaves.append(line[0]+'\n')
252                                 temp.append('')
253                         if temp not in arc_catalogs:
254                                 arc_catalogs.append(temp)
255         file.close()
256         if sort_out3==True:
257                 cont=False
258                 if len(compwaves)<3:    #at least 3 lines should be available for
                        configuration
259                         print 'not␣enough␣lines:',shortname,minwave,maxwave,grat,
                            cenwave,arc
260                         continue
261         if sort_out_not_usable==True:
262                 cont=False
263                 for line in not_usable: #doesn't plot the configurations in not_usable
264                         if line[0]==grat and line[1]==cenwave and (line[2]==arc.lower
                                () or line[2]=='*'):
265                                 print 'not␣used:',shortname,minwave,maxwave,grat,
                                    cenwave,arc
266                                 cont=True
267                 if cont==True:
268                         continue
269         if grat.find("ECHELLE")<>-1:    #doesn't plot ECHELLE as not complete
270                 print 'all␣ECHELLE␣modes␣are␣sorted␣out:',shortname,minwave,maxwave,
```

```
                grat,cenwave,arc
271             continue
272         if compwaves==[]:
273             print 'no template data from '+lampfolder+'/'+arc+'.txt for',
274             compwaves.append("0\n")
275         temp=grat+'_'+cenwave+'_'+arc.lower()+'_'+shortname+'.txt'
276         if os.path.isfile(temp)==True:
277             os.system('rm '+temp)
278         os.system('ln -s '+shortname+'.txt '+temp)       #for idl_plot
279         file=open('ticksall_'+grat+'_'+cenwave+'_'+arc.lower()+'_'+shortname+'.txt','w
                ')                                      #for idl_plot
280         for line in compwaves:
281             file.write(line)
282         file.close()
283         print shortname,minwave,maxwave,grat,cenwave,arc,        #file, min(wavelength)
                , max(wavelength), header-key-result
284         deltawave=(maxwave-minwave)/4
285         wave=['','','','','']
286         wave[1]=str(int(round(minwave+deltawave,-1)))
287         wave[2]=str(int(round(minwave+2*deltawave,-1)))
288         wave[3]=str(int(round(minwave+3*deltawave,-1)))
289         wave[0]=str(int(round(minwave,0)))       #-1 makes problems in the idl_plot, as
                the exact wavelength is necessary
290         wave[4]=str(int(round(maxwave,0)))
291         for line in override_splittings:
292             if line[0]==grat and line[1]==cenwave and line[2]==arc.lower():
293                 for i in range(5):
294                     if line[i+3]<>'':
295                         wave[i]=line[i+3]
296                 print "-> with manual settings",
297         if borders==True:
298             for dispersion in dispersions:
299                 if len(dispersion)>=5: #only dispersions with border
                        information
300                     if dispersion[0]==grat and dispersion[1]==cenwave:
                            #dispersion and cenwave are matching
301                         wave[0]=dispersion[3]
302                         wave[4]=dispersion[4]
303                         break
304         offset='0'      #offset
305         for line in manual_shift:       #check, if manual offset has to be applied
306             if line[0]==grat and line[1]==cenwave and line[2]==arc.lower():
307                 offset=line[3]
308         #spectrum=sorted(spectrum, key=itemgetter(1,2)) #sort by flux and pixel
309         #spectrumori=spectrum
310         for i in range(len(compwaves)):
311             line=compwaves[i].split()
312             compwaves[i]=float(line[0])      #replace ['wavelength',line] by float(
                    wavelength)
313         compwaves.sort()#sort by wavelength
314         dlambda=1.0*(maxwave-minwave)/len(spectrum)
315         if dlambda<1.5:
316             step=[2,4,6,10] #search-range for [maximum, close peak, too high peak
                    too close, not real maximum]
317         else:
318             step=[2,3,5,7]
319         maxflux=[]
320         posis=[]
321         quarter=0
322         for i in range(len(spectrum)): #get max(flux) in each of the four subplots
323             if spectrum[i,0]>int(wave[quarter+1]):
324                 maxflux.append(max(spectrum[posis,1])/100.)
325                 posis=[]
326                 quarter+=1
327                 if quarter>3:
328                     break
329             posis.append(i)
330         if len(maxflux)<4:
331             maxflux.append(max(spectrum[posis,1])/100.)
332         maxflux.append(max(maxflux)/10)
333         if len(maxflux)<>5:
334             exit(100)
335
336         quarter=0
337         usedlines=[]
338         goodlines=[]
339         verygoodlines=[]
340         todelete=[]
341         for compwave in compwaves:
342             text=grat+'_'+cenwave+'_'+arc.lower()+'\t'+shortname+'  '+str(compwave
                )+'  %5.2f'%dlambda+'\t'
343             for i in range(quarter,4):
```

```
344                         if compwave>int(wave[quarter+1]):        #next subplot
345                                 quarter+=1
346                         else:
347                                 break
348                 temp=abs(spectrum[:,0]-compwave)        #distance to compwave
349                 mini=min(temp)          #smallest distance to compwave
350                 posi1=numpy.where(temp==mini)[0]        #index of smallest distance to
                        compwave
351                 posi1=posi1[0]                  #make it number instead of array
352                 if posi1<step[3] or posi1>len(spectrum)-step[3]:        #only central
                        lines, these line is too far to the borders
353                         log_lines.append("6 "+text+'border\t'+str(posi1))
354                         continue
355                 if mini/dlambda<0.3:    #well centered
356                         posi1=[posi1]
357                 elif temp[posi1-1]<temp[posi1+1]:       #lower pixel is also close to
                        catalog line
358                         posi1=[posi1-1,posi1]
359                 else:                           #higher pixel is also close to catalog
                        line
360                         posi1=[posi1,posi1+1]
361                 maxi=max(spectrum[posi1[0]-step[0]:posi1[-1]+step[0]+1,1])      #
                        maximum of column 2 (flux)
362                 if maxi<maxflux[quarter] or maxi<maxflux[4]:             #otherwise the
                        lines are not visible in the subplot
363                         log_lines.append("5 "+text+'brightness\t'+str([maxi,maxflux[
                                quarter],maxflux[4]]))
364                         continue
365                 posi2=numpy.where(spectrum[posi1[0]-step[0]:posi1[-1]+step[0]+1,1]==
                        maxi)[0]                #where is maximum of the line
366                 posi2=posi2[0]+posi1[0]-step[0]                                 #make it
                        number instead of array
367                 text=text+str(spectrum[posi2,0])+'\t'+str([posi1,posi2])+'\t'+str(maxi
                        )+'\t'
368                 if max(spectrum[posi2-step[1]:posi2+step[1]+1,1])>maxi:
                                #another peak too close
369                         log_lines.append("1 "+text+'neighbor1\t'+str(max(spectrum[
                                posi2-step[1]:posi2+step[1]+1,1])))
370                         continue
371                 if max(spectrum[posi2-step[2]:posi2+step[2]+1,1])*0.1>maxi:
                                #too high peak too close
372                         log_lines.append("2 "+text+'neighbor2\t'+str(max(spectrum[
                                posi2-step[2]:posi2+step[2]+1,1])))
373                         continue
374                 posis=range(posi2-step[3],posi2-step[0]+1)+range(posi2+step[0],posi2+
                        step[2]+1)
375                 std=numpy.std(spectrum[posis,1], ddof=1)
376                 med=numpy.median(spectrum[posis,1])
377                 if maxi<med+3.5*std:                    #not significant enough
378                         log_lines.append("3 "+text+'significance\t'+str(numpy.round([
                                med,std],1)))
379                         continue
380                 if posi2 in usedlines:
381                         if goodlines not in todelete:
382                                 todelete.append(goodlines[-1])
383                         log_lines.append("4 "+text+'same line\t'+str(goodlines[-1]))
384                 log_lines.append("0 "+text+'good line\t'+str(numpy.round([med,std],1))
                        )
385                 usedlines.append(posi2)
386                 goodlines.append(compwave)
387                 if maxi>10*maxflux[quarter]:            #if more than 10% of brightest
                        line in quarter
388                         verygoodlines.append(compwave)
389         for i in range(len(goodlines))[::-1]:
390                 if goodlines[i] in todelete:
391                         del goodlines[i]
392         for i in range(len(verygoodlines))[::-1]:
393                 if verygoodlines[i] in todelete:
394                         del verygoodlines[i]
395         file=open('goodlines_'+grat+'_'+cenwave+'_'+arc.lower()+'.txt','w')
396         for line in goodlines:
397                 if line in verygoodlines:
398                         file.write(str(line)+'\t'+str(line)+'\n')
399                 else:
400                         file.write(str(line)+'\n')
401         file.close()
402         #print goodlines
403         ndfilter='-1'
404         for line in ndfilters:          #add, if neutral density filter was used
405                 if line[0].find(shortname)<>-1 or shortname.find(line[0])<>-1:
406                         ndfilter=line[1]
407                         break
```

```
408             if ndfilter=='-1':
409                     print "not in list of ndfilters",
410             print ""
411
412             #line="idl_plot,'"+grat+"','"+cenwave+"','"+arc.lower()+"','"+arc+"','"+
                    exptime+"','"+shortname+"',"+wave[0]+","+wave[1]+","+wave[2]+","+wave
                    [3]+","+wave[4]
413             index=0         #for nice sorting
414             index=+len(grat)*1E6
415             if grat.find('ECHELLE')==0:
416                     index+=(9-int(grat[-1]))*1E4     #start with ECHELLE-6 and end with
                            ECHELLE-3
417             else:
418                     index+=int(grat[1:3])*1E4        #start with low resolutions
419             if grat[-1]=='B':                                #to distinguish between same
                    gratings with B
420                     index+=0*1E4
421             elif grat[-1]=='R':                      #.... and R
422                     index+=1*1E4
423             index+=int(cenwave[:-1])*1E1     #only 3 digits
424             if arc.lower()=='he':
425                     index+=1
426             if arc.lower()=='ne':
427                     index+=2
428             if arc.lower()=='hgar':
429                     index+=3
430             if arc.lower()=='thar':
431                     index+=4
432             if arc.lower()=='cd':
433                     index+=5
434             arctext=arc
435             if arc=='HgAr':
436                     arctext='Hg'
437             #if arc<>'Cd':
438             #       continue
439             line=[index, grat,cenwave,arc.lower(),arctext,exptime,shortname,wave[0],wave
                    [1],wave[2],wave[3],wave[4],offset, str(len(goodlines)), str(len(
                    verygoodlines)), ndfilter]
440             result.append(line)
441
442     result.sort()
443     text=''
444     file=open('run_in_idl.pro','w')
445     #file.write('.RESET_SESSION\n\n')
446     for line in result:
447             text=text+' '+line[1]+"_"+line[2]+"_"+line[3].lower()+"_"+line[6]+'.pdf'
448             plot=True
449             if os.path.isfile(line[1]+"_"+line[2]+"_"+line[3].lower()+"_"+line[6]+'.pdf')
                    ==True:
450                     if os.path.getmtime(line[1]+"_"+line[2]+"_"+line[3].lower()+"_"+line
                            [6]+'.pdf')>os.path.getmtime(line[1]+'_'+line[2]+'_'+line[3].lower
                            ()+'_'+line[6]+'.txt'):
451                             plot=False
452                             print line[1]+"_"+line[2]+"_"+line[3].lower()+"_"+line[6]+'.
                                    pdf',"is newer than",line[1]+'_'+line[2]+'_'+line[3].lower
                                    ()+'_'+line[6]+'.txt',"and therefore not plotted"
453             if plot==True:
454                     file.write("idl_plot,'"+line[1]+"','"+line[2]+"','"+line[3].lower()+"
                            ','"+line[4]+"','"+line[5]+"','"+line[6]+"',"+line[7]+","+line[8]+
                            ","+line[9]+","+line[10]+","+line[11]+","+line[12]+'\n')
455                                             #grat           cenwave         arc
                                                        arctext         exptime
                                                extension       wavelength
                                                                                offset
456     file.write("\nspawn, 'pdftk "+text+" cat output catalog.pdf'\n")
457     file.write("print, 'catalog.pdf erstellt'\n")
458     file.close()
459
460     arc_catalogs.sort()
461     before=''
462     file=open('../arc_catalogs.txt','w')
463     for line in arc_catalogs:
464             if line[0]<>before:
465                     before=line[0]
466                     file.write('\n'+line[0]+'\n')
467             file.write(str(round(float(line[1]),3)).ljust(10)+' &'+line[2].ljust(6)+' &  
                    \\\\ \n' )
468     file.close()
469
470     file=open('logfile_usable_lines','w')
471     for line in log_lines:
472             file.write(line+'\n')
```

```
473  file.close()
474
475  before=-1000
476  file=open('../table.tex','w')
477  file.write('\\begin{center}\n')
478  file.write('\\begin{longtable}{l c c c l r c r}\n')
479  file.write('\\label{tab:allconfigs}\n')
480  file.write('\\bf Grating &\\bf $\lambda_{\mathrm{cen}}$ &\\bf $\lambda_{\mathrm{
         min}}-\lambda_{\mathrm{max}}$ &\\bf Dispersion &\\bf Lamp &\\bf Exp.time &\\bf
         Lines &\\bf ND \\\\\n')
481  file.write('\\\\ &\\bf \\AA &\\bf \\AA &\\bf \\AA/px &\\bf s &\\bf
         filter \\\\\n')
482  file.write('(1) &(2) &(3) &(4) &(5) &(6) &(7) &(8) \\\\\n')
483  file.write('\\hline\n')
484  file.write('\\endfirsthead\n\n')
485  file.write('\\bf Grating &\\bf $\lambda_{\mathrm{cen}}$ &\\bf $\lambda_{\mathrm{
         min}}-\lambda_{\mathrm{max}}$ &\\bf Dispersion &\\bf Lamp &\\bf Exp.time &\\bf
         Lines &\\bf ND \\\\\n')
486  file.write('\\\\ &\\bf \\AA &\\bf \\AA &\\bf \\AA/px &\\bf s &\\bf
         filter \\\\\n')
487  file.write('\\hline\\hline\n')
488  file.write('\\endhead\n\n')
489  #file.write('  \\hline\n')
490  file.write('\\endfoot\n\n')
491  file.write('\\hline\n')
492  file.write('\\endlastfoot\n\n')
493  for line in result:
494          for dispersion in dispersions:
495                  if line[1]==dispersion[0] and line[2]==dispersion[1]:   #dispersion
                         and cenwave are matching
496                          break
497          if line[0]-before>1E4:
498                  file.write('\\hline \\hline\n')
499                  file.write(line[1]+' &'+line[2]+' &$'+line[7]+'-'+line[11]+'$ &'+
                         dispersion[2]+' &'+line[4]+' &'+line[5]+' &'+line[13]+' /'+line
                         [14]+' &'+line[15]+' \\\\\n')
500                          #grat          cenwave          min wave                max
                                 wave          dispersion       arctext          exptime good
                                 lines        ndfilter
501          elif line[0]-before>1E1:
502                  file.write('\\hline\n')
503                  file.write('      &'+line[2]+' &$'+line[7]+'-'+line[11]+'$ &'+
                         dispersion[2]+' &'+line[4]+' &'+line[5]+' &'+line[13]+' /'+line
                         [14]+' &'+line[15]+' \\\\\n')
504                          #no  grat          cenwave min wave         max wave
                                 dispersion       arctext          exptime          good lines
                                        ndfilter
505          else:
506                  file.write('      &       &                &       &'+line[4]+' &'+line
                         [5]+' &'+line[13]+' /'+line[14]+' &'+line[15]+' \\\\\n')
507                  #file.write('       &       & '+line[7]+' & '+line[11]+' &       & '+line
                         [4]+' & '+line[5]+' & '+old+'       \\\\ \n')
508                          #no  gratcenwave min wave         max wave  no dispersion
                                 arctext          exptime
509          before=line[0]
510  file.write('\\end{longtable}\n')
511  file.write('\\end{center}\n')
512  file.close()
513
514  os.system('idl < run_in_idl.pro')
```