Section and #Bytes	offset		datatype	function	dict key for vk4extract dicts
Header		0	String	VK4_	
	12	4	String	dll_version	
		8	String	file_type(always 0)	
Offset Table					use extract_offsets()
		12	uint32	measurement conditions	'meas_conds'
		16	uint32	RGB data	'color_peak'
		20	uint32	RGB + light data	'color_light'
		24	uint32	light/intensity offsets (3)	'light'
		28	uint32		
		32	uint32		
		36	uint32	height offsets (3)	'height'
		40	uint32		
		44	uint32		
		48	uint32	RGB thumbnail	'clr_peak_thumb'
		52	uint32	RGB + light thumbnail	'clr_thumb''
		56	uint32	light thumbnail	'light_thumb'
		60	uint32	height thumbnail	'height_thumb'
		64	uint32	assembly data/conditions	'assembly_info'
		68	uint32	line measure	'line_measure'
		72	uint32	line thickness	'line_thickness'
		76	uint32	string data	'string_data'
		80	uint32	reserved	'reserved'
Meas Conditions (setting off +)					use extract_measurement_conditions()
	0	84	uint32	size	'size'
	4	88	uint32	year	'year'
	8	92	uint32	month	'month'
	12	96	uint32	day	'day'
	16	100	uint32	hour	'hour'
	20	104	uint32	minute	'minute'
	24	108	uint32	second	'second'
	28	112	int32	difference from UTC (min)	'diff_from_UTC'
	32	116	uint32	image attributes	'img_attributes'
	36	120	uint32	user interface mode	'user_interface_mode'
	40	124	uint32	color composite mode	'color_composite_mode'
	44	128	uint32	image layer number	'img_layer_number'
	48	132	uint32	run mode	'run_mode'
	52	136	uint32	peak mode	'peak_mode'
	56	140	uint32	sharpening level	'sharpening_level'
	60	144	uint32	speed	'speed'
	64	148	uint32	distance	'distance'
	68	152	uint32	pitch	'pitch'
	72	156	uint32	optical zoom	'optical_zoom'
	76	160	uint32	number of lines	'number_of_lines'
	80	164	uint32	line0 position	'line0_position'
	84	168	uint32	Reserved1 (3)	'Reserved_1'
	88	172	uint32		
	92	176	uint32		
	96	180	uint32	lens magnification	'lens_magnification'
	100	184	uint32	pmt gain mode	'PMT_gain_mode'
	104	188	uint32	pmt gain	'PMT_gain'
	108	192	uint32	pmt offset	'PMT_offset'
	112	196	uint32	nd filter	'ND_fliter'
	116	200	uint32	reserved2	'Reserved_2'
	120	204	uint32	persist count	'persist_count'
	124	208	uint32	shutter speed mode	'shutter_speed_mode'
	128	212	uint32	shutter speed	'shutter_speed'
	132	216	uint32	white balance mode	'white_balance_mode'
	136	220	uint32	white balance red	'white_balance_red'
	140	224	uint32	white balance blue	'white_balance_blue'
	144	228	uint32	camera gain	'camera_gain'
	148	232	uint32	plane compensation	'plane_compensation'
	152	236	uint32	xy length unit	'xy_length_unit'
	156	240	uint32	z length unit	'z_length_unit'
	160	244	uint32	xy decimal place	'xy_decimal_place'
	164	248	uint32	z decimal place	'z_decimal_place'
	168	252	uint32	x length per pixel	'x_length_per_pixel'
	172	256	uint32	y length per pixel	'y_length_per_pixel'
	176	260	uint32	z length per digit	'z_length_per_digit'
	180	264	uint32	reserved3 (5)	'Reserved_3'
	184	268	uint32		

188	272	uint32		
192	276	uint32		
196	280	uint32		
200	284	uint32	light filter type	'light_filter_type'
204	288	uint32	reserved4	'Reserved_4'
208	292	uint32	gamma reverse	'gamma_reverse'
212	296	uint32	gamma	'gamma'
216	300	uint32	gamma correction offset	'gamma_correction_offset'
220	304	uint32	ccd bw offset	'CCD_BW_offset'
224	308	uint32	numerical aperture	'num_aperture'
228	312	uint32	head type	'head_type'
232	316	uint32	pmt gain2	'PMT_gain_2'
236	320	uint32	omit color image	'omit_color_img'
240	324	uint32	lens id	'lens_ID'
244	328	uint32	light lut mode	'light_lut_mode'
248	332	uint32	light lut in0	'light_lut_in0'
252	336	uint32	light lut out0	'light_lut_out0'
256	340	uint32	light lut in1	'light_lut_in1'
260	344	uint32	light lut 0ut1	'light_lut_out1'
264	348	uint32	light lut in2	'light_lut_in2'
268	352	uint32	light lut out2	'light_lut_out2'
272	356	uint32	light lut in3	'light_lut_in3'
276	360	uint32	light lut out3	'light_lut_out3'
280	364	uint32	light lut in4	'light_lut_in4'
284	368	uint32	light lut out4	'light_lut_out4'
288	372	uint32	upper position	'upper_position'
292	376	uint32	lower position	'lower_position'
296	380	uint32	light effective bit depth	'light_effective_bit_depth'
300	384	uint32	height effective bit depth	'height_effective_bit_depth'

RGB (off +)				use extract_color_data() with ‘peak’
0	732	uint32	image width	'width'
4	736	uint32	image height	'height'
8	740	uint32	bit depth	'bit_depth'
12	744	uint32	compression	'compression'
16	748	uint32	byte size	'data_byte_size'
20	752	const uchar*	RGB data for byte size	'data'

RGB + light (off +)				use extract_color_data() with ‘light’
0	2360048	uint32	image width	'width'
4	2360052	uint32	image height	'height'
8	2360056	uint32	bit depth	'bit_depth'
12	2360060	uint32	compression	'compression'
16	2360064	uint32	byte size	'data_byte_size'
20	2360068	const uchar*	RGB data for byte size	'data'

-

Light Data (off +)				use extract_img_data() with ‘light’
0	4719364	uint32	image width	'width'
4	4719368	uint32	image height	'height'
8	4719372	uint32	bit depth	'bit_depth'
12	4719376	uint32	compression	'compression'
16	4719380	uint32	byte size	'data_byte_size'
20	4719384	uint32	palette range min	'palette_range_min'
24	4719388	uint32	palette range max	'palette_range_max'
28	4719392	uchar[0x300]	palette (lookup table?)	'palette'
796	4720160	const uchar*	data for byte size	'data'

Height Data (off +)				use extract_img_data() with ‘height’
0	6293024	uint32	image width	'width'
4	6293028	uint32	image height	'height'
8	6293032	uint32	bit depth	'bit_depth'
12	6293036	uint32	compression	'compression'
16	6293040	uint32	byte size	'data_byte_size'
20	6293044	uint32	palette range min	'palette_range_min'
24	6293048	uint32	palette range max	'palette_range_max'
28	6293052	uchar[0x300]	palette (lookup table?)	'palette'
796	6293820	const uchar*	data for byte size	'data'

Color Peak Thumb

9439548

Color Thumb

9526004

Light Thumb

9612460

Height Thumb

9698916

Assembly Info (off +)			
0	9785372	uint32	size
4	9785376	uint16	file type
6	9785378	uint16	stage type
8	9785380	unit32	x position
12	9785384	uint32	y postion
16			

String Data (off +)			use extract_string_data()	
	9785388	unint32	Title length	
4	9785392	char*	title for length*2 bytes	'title'
4 + 2*title length	9785388	unint32	Lens name length	
8 + 2*title length	9785392	char*	lens name for length*2 bytes	'lens_name'

End of File

example file: FY09 DE02_Y1_X1.vk4
VK4_
0x09 0x01 0x02 0x01
0x00 0x00 0x00 0x00

	72 bytes
84	
732	
2360048	
4719364	
0	
0	
6293024	
0	
0	
9439548	
9526004	
9612460	
9698916	
9785372	
0	
0	
9785388	
0	
	300 bytes
648	
2016	
12	
9	
13	
44	
5	
-300	
0	
0	
0	
1	
0	
0	
0	
5	
135450	nm
500	nm
10	Float => optical_zoom/10.0
0	
0	
0	
0	
0	
200	Float => lens_mag/10.0
0	
7260	
8101	
1	
0	
0	“Image average frequency”
2	
54	
1	
129	
168	
0	6*camera_gain in dB
0	
1	
1	
0	
0	
694920	pm
694920	pm
100	pm
28356	
21082	

10720	
569	
54299	
0	
0	
0	
45	Float => gamma/100.0
0	Float => gamma_offset/65536.0
908	Float => ccd_bw_offset/100.0
350	Float => num_aperture/1000.0
6	
0	
0	
5	
0	
0	
200	
50	
150	
100	
100	
150	
50	
200	
0	
6037090	nm
6172540	nm
16	
28	

	2359316
1024	
768	
24	
0	
2359296	width*height*(bit depth/8)

	2359316
1024	
768	
24	
0	
2359296	width*height*(bit depth/8)

	1573660
1024	
768	
16	
909198157	MC16
1572864	width*height*(bit depth/8)
65535	2^16
16777216	2^24

	3146524
1024	
768	
32	
842220365	MC32
3145728	width*height*(bit depth/8)
35278	
1173820	

86456 bytes

86456 bytes

86456 bytes

86456 bytes

16? bytes
16
0
1
4294967209
6106
218?
23?

64

Hexdump shows:
FY09 DE02 270 SW
(Stitch 6mm from Weld)
After 2nd Cleaning_Y1_X1
20x Long WD