Fitting PACS ramps with analytical models. Part III
The IMEC model

Martin Groenewegen (ICC KUL)

DOCUMENT CHANGE RECORD

Version	Date	Changes	Remarks
Draft 0	22-September-2004	–	New document.
Draft 1	17-March-2005	Updated after discussion with P.M.	
Draft 2	27-May-2005	Finalised	Uploaded to document tree.

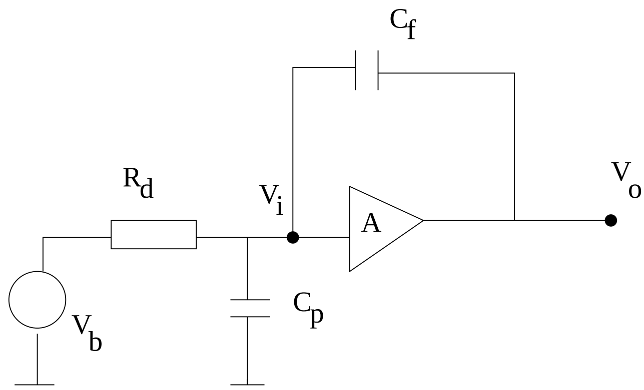

Figure 1: Schematical representation of the circuit. Feedback capacitance C_{f}, Parasitic capacitance C_{p}, Amplifier A, Resistor R_{d} (a proxy for the detector), output voltage V_0, bias voltage V_{b}.

Reference Documents

RD1 – Fitting PACS ramps with analytical models. Part I, PICC-KL-TN-007, Draft 1, 18-02-2004
RD2 – Fitting PACS ramps with analytical models. Part II, PICC-KL-TN-009, Draft 1, 7-06-2004
RD3 – Report on amplifier gain and equivalent input voltage noise density measurement, PACS-IM-RP-770, 15-07-2004

1. Introduction

This technical note is a follow-up to RD1 and RD2 and describes the third attempt to fit Spectroscopic PACS data with an analytical model within the framework of the PCSS/IA.

Using an analytical model to describe PACS ramps could be useful for OBSW data compression or for the on-ground data reduction. Also it could be useful in numerical simulations of ramps, since the ramps are generally not straight lines.

This document accompanies the Python scripts PyRamp4Model.py and PyRamp6Model.py that, respectively, implement the mathematical model described below, and plots the output generated. In addition the file (input4.list) which contains the data files analysed is given. All these file are available under CVS at the Leuven site.

2. Model

RD1 and RD2 were based on the model by Albrecht Poglitsch. Here, a model proposed by IMEC is considered.

Consider the circuit displayed in Figure 1.

Consider the impedance at node V_{i}. Following Millman's theorem one has (in LaPlace coordinate s):

$$V_{\mathrm{i}} = \frac{V_0 s C_{\mathrm{f}} + 0 s C_{\mathrm{p}} + V_{\mathrm{b}}/R_{\mathrm{d}}}{s C_{\mathrm{f}} + s C_{\mathrm{p}} + 1/R_{\mathrm{d}}} \tag{1}$$

The response behaviour of the amplifier is approximated by a time-constant τ_{c}, and the gain A.

$$V_0 = -\frac{A}{1 + s\tau_{\mathrm{c}}} V_{\mathrm{i}} \tag{2}$$

Substituting Equation (1) in (2), and solving for V_0, one obtains

$$V_0 = (-A\ V_{\mathrm{b}}) / \left(1 + R_{\mathrm{d}}\ s(AC_{\mathrm{f}} + C_{\mathrm{p}} + C_{\mathrm{f}}) + s(1 + (C_{\mathrm{p}} + C_{\mathrm{f}})R_{\mathrm{d}}\ s)\tau_{\mathrm{c}}\right) \tag{3}$$

Grouping the terms in s and s^2 in the denominator, and defining

$$\omega = 1.0/\sqrt{(C_\mathrm{f} + C_\mathrm{p})\ R_\mathrm{d}\ \tau_\mathrm{c}} \tag{4}$$

and

$$\zeta = \omega \times \frac{1}{2} \left(((1.0 + A)\ C_\mathrm{f} + C_\mathrm{p})\ R_\mathrm{d} + \tau_\mathrm{c}\right) \tag{5}$$

one arrives at:

$$V_0 = (-A\ V_\mathrm{b})\ \frac{1}{\left(1 + \frac{s^2}{\omega^2} + \frac{2s\zeta}{\omega}\right)} \tag{6}$$

At this point, as an ad-hoc refinement to the model, a parameter (τ) is introduced, to replace the "1" in the nominator in Eq (6) by $(1 - s\tau)$. The inverse LaPlace transformation of

$$(-A\ V_\mathrm{b})\ \frac{(1 - s\tau)}{\left(1 + \frac{s^2}{\omega^2} + \frac{2s\zeta}{\omega}\right)} \tag{7}$$

and the functional form sought after is:

$$V(t) = V(0) + (A\ V_\mathrm{b})\,(-1.0 + \exp(-t\ \zeta\ \omega) \times (\cosh(t\ \omega\ d) + (\zeta + \omega\ \tau)/d \times \sinh(t\ \omega\ d))) \tag{8}$$

with

$$d = \sqrt{\zeta^2 - 1}$$

At this point 2 mathematical implementations of this physical model are introduced. As in RD1, the mathematical implementation includes the provision to avoid negative values, by coding e.g. A as $(\exp p_1)$, etc.

1 Ramp4Model.java

with parameters

$V(0) \Rightarrow \exp(p_0)$
$A' = (A\ V_\mathrm{b}) \Rightarrow \exp(p_1)$
$\zeta \Rightarrow \exp(p_2)$
$\omega \Rightarrow \exp(p_3)$
$\tau \Rightarrow \exp(p_4)$

2 Ramp6Model.java

with parameters

$V(0) \Rightarrow \exp(p_0)$
$A \Rightarrow \exp(p_1)$
$C_\mathrm{f} \Rightarrow \exp(p_2)$
$C_\mathrm{p} \Rightarrow \exp(p_3)$
$R_\mathrm{d} \Rightarrow \exp(p_4)$
$\tau_\mathrm{c} \Rightarrow \exp(p_5)$
$\tau \Rightarrow \exp(p_7)$
$V_\mathrm{b} \Rightarrow \exp(p_7)$

3. Initial guesses

Since one might assume under certain circumstances that the parasitic capacitance (and obviously the integrating capacitance and bias voltage) are known, some parameters could be fixed in the fitting.

Can initial guesses for the other parameters be determined for a given $V(t)$?

p_0 trivially follows from $V(0)$.

The time derivative of $V(t)$ is:

$$\frac{dV(t)}{dt} = A' \ \exp(-t\ \zeta\ \omega) \times \{(-\zeta\ \omega)\ [\cosh(t\ \omega\ d) + (\zeta + \omega\ \tau)/d \times \sinh(t\ \omega\ d)] \tag{9}$$

$$+(d\ \omega)\ [\sinh(t\ \omega\ d) + (\zeta + \omega\ \tau)/d \times \cosh(t\ \zeta\ d)]\} \tag{10}$$

Putting $\frac{dV(t)}{dt} = 0$ gives the location of the maximum of the "bump", which can be determined easily from the data.

The result is:

$$\tanh(t\ \omega\ d) = (-\omega\ \tau)/\left(d - \zeta/d \times (\zeta + \omega\ \tau)\right) \tag{11}$$

For small t, $\tanh(x) \approx x$, and using the definition of d, one gets an estimate for τ:

$$\tau = t_{\max}/(1 - t_{\max}\ \zeta\ \omega) \tag{12}$$

where $t_{\max}$ = (readout_where_the_maximum_occurs)/256.

The second interesting regime is when $\exp(x) >> \exp(-x)$, and hence $\cosh x \approx \sinh x = \frac{1}{2}\exp(x)$. In that case:

$$\frac{dV(t)}{dt} \approx \frac{1}{2}\ A'\ \omega\ \exp(-t\ \omega\ (\zeta - d)) \times [d + \omega\ \tau - \zeta\ (\zeta + \omega\ \tau)/d] \tag{13}$$

Consider the regime where the exponential may still be considered to be close to unity. For example, consider specifically,

$$t_{\rm eval} = 0.2/(\omega\ (\zeta - d)), \tag{14}$$

then, also simplifying/approximating the term within "[....]" to $\frac{-1}{\zeta}$, gives:

$$\frac{dV(t)}{dt}_{|t=t_{\rm eval}} = -\frac{1}{2}\ A'\ \omega\ 0.82/\zeta = -0.82\ \frac{(AV_{\rm b})}{((1.0 + A)\ C_{\rm f} + C_{\rm p})\ R_{\rm d} + \tau_{\rm c}} \tag{15}$$

A shorter time for $t_{\rm eval}$ is not feasible as one has to consider the initial assumption $\exp(x) >> \exp(-x)$ which implies roughly $t_{\min} \gtrsim 2.0/(\omega\ d)$.

From the data itself, the time derivative can be estimated at $t = t_{\rm eval} > t_{\min}$, and hence an initial guess for $R_{\rm p}$ may be derived.

In the limit of infinite gain Equation (15) becomes

$$\frac{dV(t)}{dt} = -(0.82)\ \frac{V_{\rm b}}{C_{\rm f}\ R_{\rm d}} \tag{16}$$

4. IA/Python

The necessary files are located at the Leuven CVS server at / develop/ pcss/ herschel/ pacs/ scripts/ obswscripts

The related files are input4.list, PyRamp4Model.py, Ramp4Model.java, PyRamp6Model.py, Ramp6Model.java, RaiseAtEnd.py, EdgeDetection.py, ADCSaturation.py, DynamicRange.py, Histogram.java and MyComplex.java[1].

Ramp4Model.java and Ramp6Model.java are the actual implementation of the mathematical model as an extension of the NonLinear java class. Compile it with "javac Ramp4Model.java". Histogram.java allows you to make histograms.

MyComplex.java is a, bug-corrected, general tool for handling of functions with complex numbers taken from the World Wide Web. It is explicitly used to calculate the $\cosh$ and $\sinh$ functions. Compile it with "javac MyComplex.java" and make sure it is in your CLASSPATH[2].

input4.list (the name is arbitrary) contains the pathnames to the file(s) you want to analyse, and in this case tha names of the files considered in the present study.

PyRamp4Model.py and PyRamp6Model.py do the actual fitting. There are loops over the file, the module, the detector and the ramps (loop-index i, j, k, l, respectively). In JIDE one can run the script as is, or execute it line-by-line and set the appropriate i, j, k, l-index manually (and skip the **FOR**-loops) to select a particular ramp.

The output of PyRampXModel.py are 3 files, "output_of_rampXmodel.dat", "failed_of_rampX.dat", "edge_of_rampX.dat" and "weird_of_rampX.dat" (X = 4 or 6). The first file lists the input and output parameters of the fitting. The second file list the indices l, k, j, i of those ramps where the fitting failed (i.e. A java exception occurred) and at the end some overall statistical information, the third file lists the indices i, j, k, l and filenames of those ramps where an "edge" was found earlier than 10 readouts from the end, and the last file lists the indices i, j, k, l and filenames of those ramps with a very high (arbitrarily chosen set at 50) reduced χ^2.

In the present version all Modules (except j-index= 0 and 1), and all Channels/Detectors (except k-index= 0, 1 and 17) are fitted of all files in the input list, but (still) for reasons of computational speed, only the first 10 ramps. The results below are based on fitting about 32 000 ramps in total.

Two auxiliary Python scripts have been written to deal with saturation. The first type of saturation is of the AC-to-DC convertor. ADCSaturation.py looks for the first occurance in the ramp of a specific numerical value (-32767 in this case). All non-destructive readouts (NDRs) after which this value occurs are not considered. The second script, DynamicRange.py, takes an input value (the dynamic range in Volt) and look if there are NDRs (and if so, returns the first occurance) where the voltage drops below the maximum value minus the input value. If so, all NDRs after the dynamical range is exceeded are ignored. In the present version a value of 1.8 Volt has been used. This seems to be sufficient, but there are also clear cases where the dynamical range is larger (1.9 V), and so fewer NDRs have been used in those fits than could have been. However, to optimise this requires a more detailed investigation of the dynamical range over capacitance and other parameters.

EdgeDetection.py uses a [-2 0 2] Sobel filter to detect "edges".

Only ramps where the difference between the maximum and minimum of a ramp exceeded 0.035 Volt (to eliminate some erratic ramps.....)

The first and last NDR are always ignored (as described above, more NDRs at the end of a ramp are ignored if AC-to-DC saturation occurs or if the dynamical range is exceeded).

C_{f} was fixed at 3680/ 1330/ 520/ 280 fF for the nominal values of 3000/ 1000/ 300/ 100 fF, as determined in Heidelberg tests (the true capacitance values have never been determined for the capacitors used in the MPE April 2003 tests).

[1] Ramp5Model was an earlier, incorrect, extension of Ramp4Model. It was decided to keep the numbering scheme.

[2] This java class was introduced when IA had no $\cosh$ and $\sinh$ functionality defined. This is the case now, but this class provides other interesting arithmetic on Complex Numbers.

Table 1: Summary of fit results with all parameters of Ramp4Model left free. The last 6 entries are the median values over all ramps fitted.

T (K)	BB (K)	bias (mV)	t_{int} (sec)	C (pF)	#files	#ramps fitted	$V(0)$	A'	ζ	ω	τ	#calls	χ^2_{red}
1.8	33	60	1	3	1	536	3.06	2.60	21.17	2.07	0.021	19	1.63
1.8	33	60	1	1	1	590	2.87	1.50	10.38	3.92	0.060	17	1.26
1.8	33	60	1	0.3	1	436	2.44	2.22	9.71	6.49	0.141	16	13.4
1.8	33	60	1	0.1	1	388	1.89	3.72	6.88	5.87	0.233	15	18.8
1.8	33	50	1	3	1	426	3.06	1.30	18.52	2.17	0.018	28	1.64
1.8	33	50	1	1	1	590	2.87	1.29	11.15	3.64	0.055	17	1.12
1.8	33	50	1	0.3	1	600	2.45	1.49	9.16	6.11	0.134	17	4.89
1.8	33	50	1	0.1	1	491	1.90	3.55	9.01	5.63	0.274	16	22.4
1.8	33	40	1	3	1	302	3.06	0.75	22.62	2.84	0.034	30	1.62
1.8	33	40	1	1	1	586	2.88	0.99	12.54	3.35	0.056	17	1.18
1.8	33	40	1	0.3	1	600	2.46	1.00	8.76	5.79	0.126	17	3.07
1.8	33	40	1	0.1	1	586	1.94	2.79	10.12	5.03	0.271	16	11.0
1.8	33	30	1	3	1	50	3.05	0.58	20.62	3.12	0.042	21	1.83
1.8	33	30	1	1	1	536	2.88	0.71	16.61	2.98	0.050	18	1.26
1.8	33	30	1	0.3	1	590	2.47	0.74	10.17	5.06	0.131	18	2.39
1.8	33	30	1	0.1	1	600	1.96	2.55	11.91	3.56	0.282	17	8.00
1.8	50	60	1	3	1	511	3.05	2.42	9.77	2.83	0.019	16	1.53
1.8	50	60	1	1	1	599	2.83	4.28	12.91	5.49	0.065	16	4.65
1.8	50	60	1	0.3	1	411	2.37	9.22	11.30	6.32	0.134	17	33.2
1.8	50	60	1	0.1	0								
1.8	50	50	1	3	1	543	3.06	2.67	14.41	2.24	0.019	17	1.52
1.8	50	50	1	1	1	590	2.85	3.77	12.11	4.57	0.065	16	2.94
1.8	50	50	1	0.3	1	231	2.33	6.62	11.94	7.18	0.153	17	35.3
1.8	50	50	1	0.1	1	188	1.91	4.21	5.70	7.10	0.193	16	36.5
1.8	50	40	1	3	1	493	3.06	2.46	17.06	2.09	0.018	18	1.53
1.8	50	40	1	1	1	590	2.86	1.66	10.48	4.22	0.061	17	1.34
1.8	50	40	1	0.3	1	326	2.44	2.69	10.24	6.35	0.146	16	22.0
1.8	50	40	1	0.1	1	377	1.88	3.75	6.66	6.84	0.236	16	29.3
1.8	50	30	1	3	1	470	3.06	1.61	23.22	2.13	0.026	19	1.65
1.8	50	30	1	1	1	590	2.87	1.16	10.88	4.10	0.061	17	1.29
1.8	50	30	1	0.3	1	600	2.44	1.45	8.93	6.10	0.142	17	7.21
1.8	50	30	1	0.1	1	404	1.87	3.25	9.67	6.98	0.277	16	33.5

6. Results for Ramp4Model

C_{p} was initially fixed at 100 pF, and R_p to 3.7GΩ, based on RD2, in order to estimate ω and ζ. Analysis of the fitresults will show that better initial guesses are obtained with significantly smaller values of C_{p} (effectively tending to zero), and values of R_p of 100 GΩ (also see Section 7).

A total of 32 000 ramps has been fitted. Figure 1 shows selected fits for all relevant settings with an integration time of 1 second. Plotted are Volts versus read-out number. On the left the BB-temperature is 33 K, on the right 50 K. From top to bottom (nominal) capacitance values of 3, 1, 0.3, 0.1 pF. The bias voltage changes per page/figure from 60, 50, 40, 30 mV. The red points are the fits to the thick black points. Smaller black points are read-outs not considered (last read-out, saturation, etc). At the top of the plots are listed: the file name, the indices i_j_k_l, indicating the file number in the full list of files analysed, Module, Detector, Ramp, and the reduced Chi-square. The reduced Chi-square value is based on an (arbitrary) assigned error of 1 mV at every read-out.

The blue dots represent the residual **TEN** * (observed-fitted), shifted to the mean voltage, indicated by the blue horizontal line. The blue line is a Gaussian fit to a 9-bin histogram of these residuals evaluated at the centre of the bins.

Table 1 summarises the fit results. This "calibration table" shows that, although the "bump" is fitted well (see Figure 1), fits with significantly lower chi-square are only obtained with higher capacitance values (1 and 3 pF), where the "bump" is intrinsically less pronounced. This is also visible from inspection of the residuals in Figure 1. For the higher capacitance values the noise appears to be Gaussian distributed around the fit, while for lower capacitance values there are systematic deviations of the residuals.

In Figure 2 a comparison is made for certain setting (BB-temperature of 33 K, a bias voltage of 50 mV) how the AlPog and IMEC model compare for the four capacitance values. The ramps within the dataset are randonly chosen, and the trends are independent of this or of the particular set-up. Note that the fit to the AlPog model was done in its final configuration described in RD2, i.e. with 6 or 7 of the 10 model parameters fixed, depending on the "bump". In all cases the IMEC model gives the lower reduced chi-squared. The difference is largest for the lower capacitance values, but as described earlier, the IMEC model does perform poorly in an absolute sense. The biggest gain is in the 1 pF model where systematic effects seen in the AlPog model are largely (but not completely) removed.

7. Results for Ramp6Model

In this implementation the bias voltage, feedback capacitance, parasitic capacitance are fixed ($C_{\rm p}$ at 4pF). Equation (16) was used to estimate $R_{\rm d}$, where (dV/dt) was estimated as 256 times the median value of the values (V(i+1)-V(i)).

Table 2 lists the fit results. With all these parameters left free some unexpected trends ans results may be observed: (1) The gain is low (tens rather than hundreds), and (2) a correlation between τ_c and feedback capacitance (at least for $c_{\rm f} \gtrsim 0.3$ pF).

As noted in the previous section, the "bump" is more pronounced for lower capacitor values, resulting in a peak value of the ramps at later read-outs, leading to a larger value for τ.

8. Conclusions and future work

The IMEC model does perform better than the AlPog model described in RD1 and RD2 in terms of a lower reduced chi-square. This is largely due to an improved description of the "bump".

Nevertheless, for the 0.1 and 0.3 pF capacitance values the fits are poor in an absolute sense, and the residuals show systematic trends.

For capacitances of 1 and 3 pF very good fits can be obtained with essentially no systematic effects and a Gaussian distribution of the residuals.

A calibration table is provided with initial estimates for the 4 free parameters (the fifth parameter of the model is the voltage at the first-readout which is trivial) as function of bias and capacitance. Alternatively, a numerical recipe is presented how estimates for 2 of these parameters can be obtained from the data itself, while for the remaining 2 estimates can be obtained assuming generic values for the parasitic capacitance and resistor value.

An important caveat is that these results were obtained on data taken in 2003 with CREs that are no longer the nominal ones. Part of the analysis should be repeated on the latest CRE datasets, also taking into account some of the latest measured results obtained by IMEC on e.g. the gain and $\tau_{\rm c}$ (see e.g. RD3). However the software tools to do so now exist.

Table 2: Summary of fit results for Ramp6Model. The last 8 entries are the median values over all ramps fitted.

T (K)	BB (K)	bias (mV)	C (pF)	#files	#ramps fitted	$V(0)$	A	R_p	τ_c	τ	σ	#calls	χ^2_{red}
1.8	33	60	3	1	476	3.06	49.2	1.33 e11	0.33	0.023	0.00125	12	1.63
1.8	33	60	1	1	590	2.87	25.1	1.21 e11	0.096	0.060	0.00111	13	1.28
1.8	33	60	0.3	1	555	2.44	37.4	1.25 e11	0.043	0.141	0.00438	48	13.6
1.8	33	60	0.1	1	385	1.89	62.3	1.17 e11	0.059	0.233	0.00473	20	19.0
1.8	33	50	3	1	386	3.06	37.2	1.72 e11	0.238	0.014	0.00128	12	1.66
1.8	33	50	1	1	588	2.87	26.0	1.54 e11	0.100	0.055	0.00106	13	1.14
1.8	33	50	0.3	1	600	2.45	29.8	1.36 e11	0.045	0.134	0.00235	26	4.96
1.8	33	50	0.1	1	488	1.90	71.2	1.46 e11	0.057	0.274	0.00491	21	22.6
1.8	33	40	3	1	297	3.06	34.4	2.07 e11	0.132	0.025	0.00126	18	1.68
1.8	33	40	1	1	531	2.88	24.9	2.16 e11	0.086	0.056	0.00108	13	1.19
1.8	33	40	0.3	1	597	2.46	24.9	1.73 e11	0.041	0.126	0.00178	22	3.11
1.8	33	40	0.1	1	582	1.94	70.5	1.74 e11	0.061	0.271	0.00323	21	11.2
1.8	33	30	3	1	59	3.05	26.1	1.80 e11	0.136	0.022	0.00130	22	1.86
1.8	33	30	1	1	422	2.88	24.5	2.63 e11	0.087	0.045	0.00111	16	1.26
1.8	33	30	0.3	1	564	2.47	24.6	2.42 e11	0.040	0.131	0.00156	25	2.39
1.8	33	30	0.1	1	598	1.96	87.3	2.64 e11	0.079	0.282	0.00274	21	8.11
1.8	50	60	3	1	504	3.05	41.3	3.80 e10	0.352	0.020	0.00124	13	1.54
1.8	50	60	1	1	598	2.83	71.6	4.06 e10	0.129	0.065	0.00231	16	4.70
1.8	50	60	0.3	1	406	2.37	154.	4.78 e10	0.127	0.133	0.00642	22	33.6
1.8	50	60	0.1	0									
1.8	50	50	3	1	510	3.06	57.6	5.45 e10	0.458	0.019	0.00123	12	1.53
1.8	50	50	1	1	590	2.85	75.7	5.47 e10	0.171	0.065	0.00183	15	2.97
1.8	50	50	0.3	1	225	2.33	131.	4.09 e10	0.100	0.153	0.00757	21	35.5
1.8	50	50	0.1	1	185	1.91	84.2	5.19 e10	0.087	0.193	0.00739	18	36.6
1.8	50	40	3	1	458	3.06	67.3	7.31 e10	0.505	0.019	0.00124	12	1.57
1.8	50	40	1	1	590	2.86	41.5	6.84 e10	0.145	0.061	0.00117	14	1.36
1.8	50	40	0.3	1	326	2.44	68.4	7.90 e10	0.070	0.146	0.00595	21	22.3
1.8	50	40	0.1	1	370	1.88	99.0	6.55 e10	0.083	0.237	0.00609	19	29.5
1.8	50	30	3	1	452	3.06	69.1	1.04 e11	0.388	0.021	0.00128	12	1.67
1.8	50	30	1	1	590	2.87	38.8	9.48 e10	0.125	0.061	0.00112	13	1.31
1.8	50	30	0.3	1	600	2.44	48.4	8.61 e10	0.058	0.142	0.00279	21	7.35
1.8	50	30	0.1	1	392	1.87	109.	9.06 e10	0.065	0.277	0.00584	37	33.7

Acknowledgements

Patrick Merken (IMEC) is warmly thanked for his involvment.

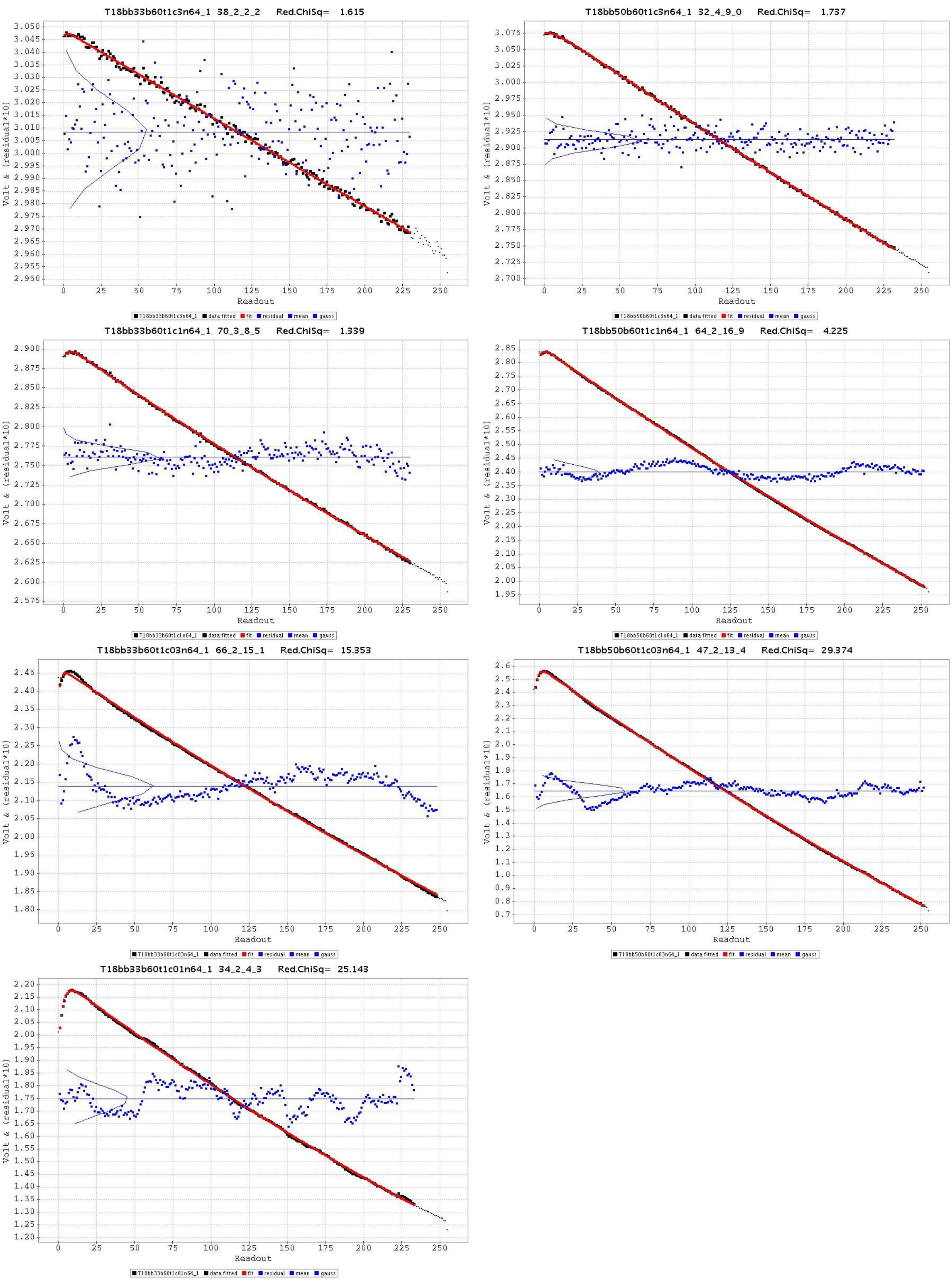

Figure 2: Selected fits for ramps of 1 second with Ramp4Model. Plotted are Volts versus read-out number. On the left with BB-temperature 33 K, on the right 50 K. From top to bottom (nominal) capacitance values of 3, 1, 0.3, 0.1 pF. The bias voltage changes per page/figure from 60, 50, 40, 30 mV. The red points are the fits to the thick black points. Smaller black points are read-outs not considered in the fitting (last read-out, saturation, etc). The blue dots represent the residual, **TEN** * (observed-fitted), shifted to the mean voltage, indicated by the blue horizontal line. The other blue line is a Gaussian fit to a 9-bin histogram of these residuals. At the top of the plots are listed: the file name, the indices i_j_k_l, indicating the file number in the full list of files analysed, Module, Detector, Ramp, and the reduced Chi-square.

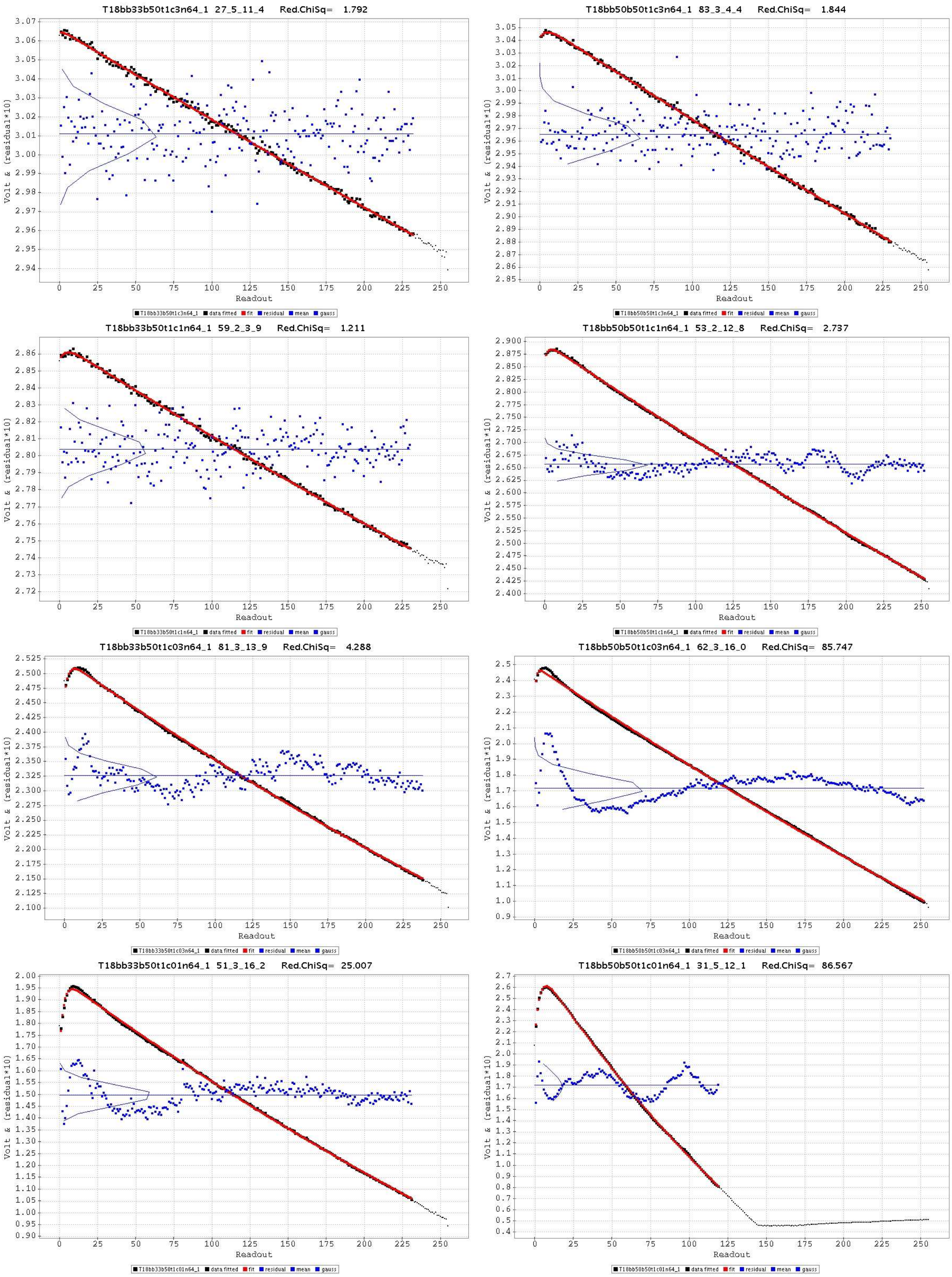

Figure 2: Continued

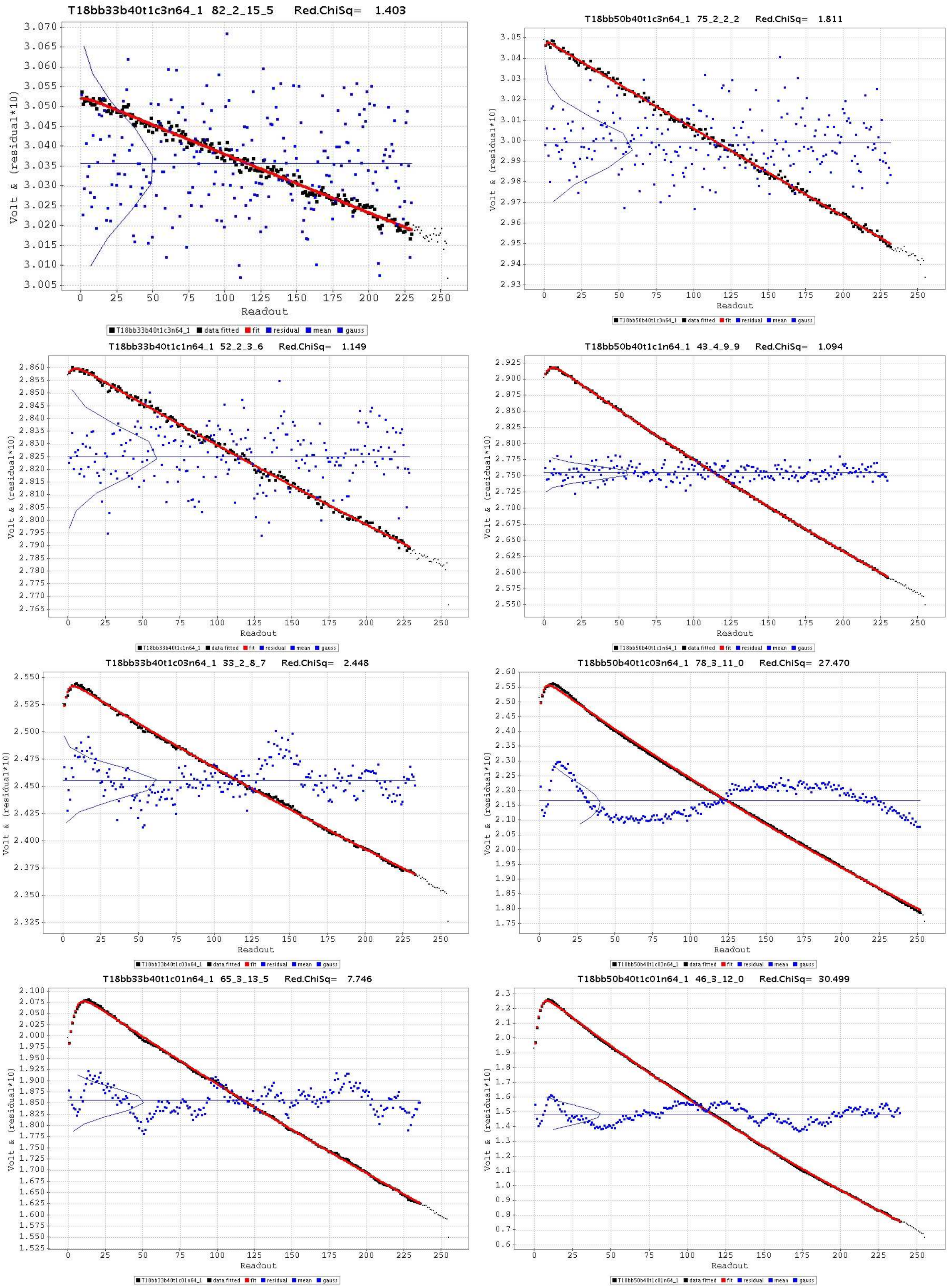

Figure 2: Continued

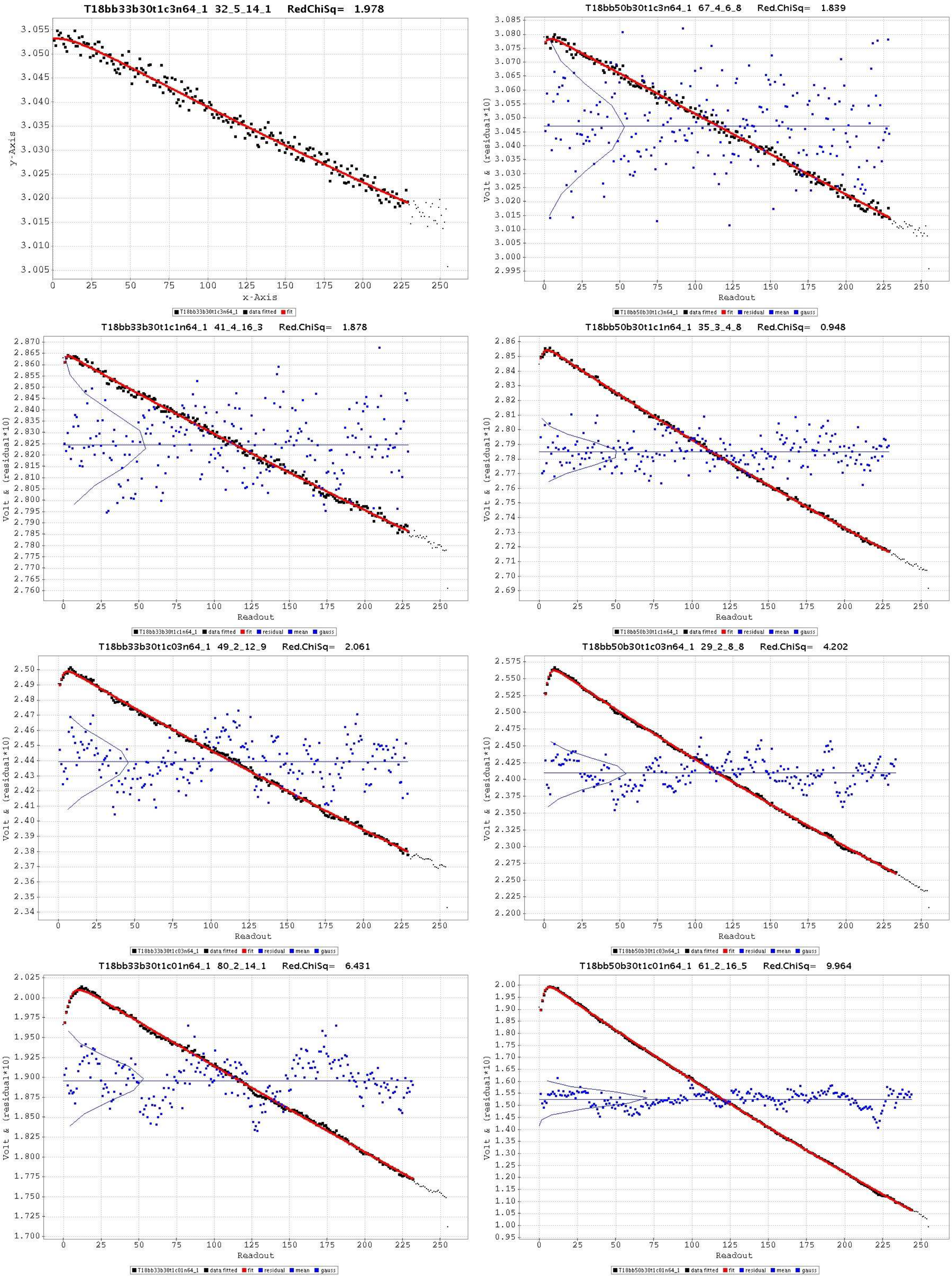

Figure 2: Continued

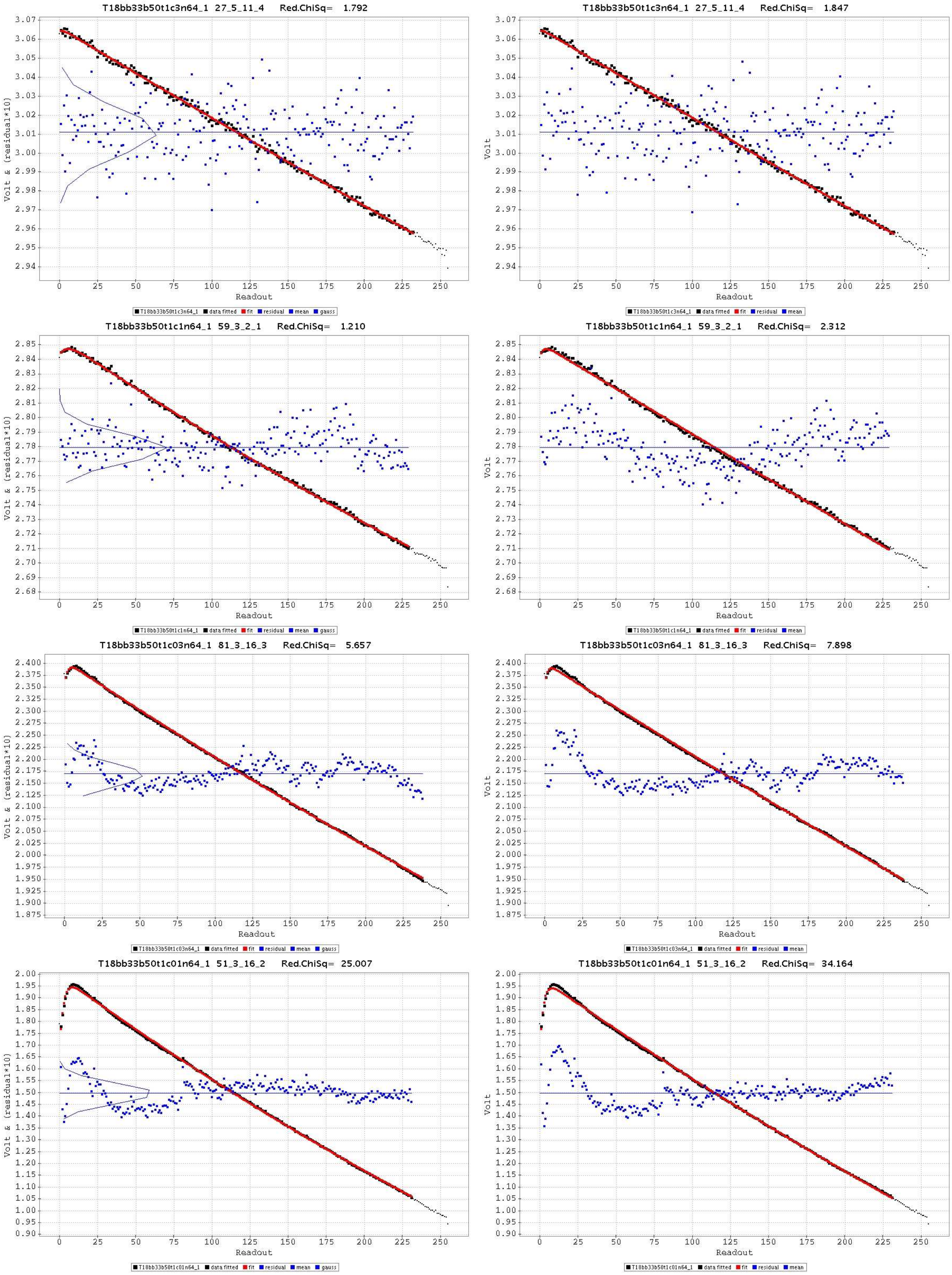

Figure 3: Comparison between IMEC model (left), and AlPog model (right) for (nominal) capacitance values of 3, 1, 0.3, 0.1 pF (top to bottom), a BB-temperature 33 K, and a bias voltage of 50 mV.