Averaging and recording of digital deep-level transient spectroscopy transient signals

P. V. Kolev, M. J. Deen, and N. Alberding

Citation: Review of Scientific Instruments **69**, 2464 (1998); doi: 10.1063/1.1148975
View online: http://dx.doi.org/10.1063/1.1148975
View Table of Contents: http://scitation.aip.org/content/aip/journal/rsi/69/6?ver=pdfcov
Published by the AIP Publishing

Averaging and recording of digital deep-level transient spectroscopy transient signals

P. V. Kolev[a] and M. J. Deen
School of Engineering Science, Simon Fraser University, Burnaby, B.C. V5A 1S6, Canada

N. Alberding
Department of Physics, Simon Fraser University, Burnaby, B.C. V5A 1S6, Canada

(Received 12 May 1997; accepted for publication 6 March 1998)

A new approach to the signal processing in digital deep-level transient spectroscopy (DLTS) systems is introduced. The problems of signal recovery from noise and efficient data storage are addressed separately from the transient signal analysis. As a result of this approach, an improved digital averaging scheme for DLTS signal recovery from noise and transient data storage is proposed. We have shown that the combined action of two complementary digital averaging techniques can improve the DLTS digital signal processing. Pseudologarithmic time averaging is efficient in reducing the number of processed data points and improving the signal-to-noise ratio for the high frequency noise components and in the transient tail. We demonstrate that the normalized errors in the magnitude measurements introduced by this type of averaging remain below 1% for pure logarithmic and below 0.1% provided that the logarithmic averaging intervals are further divided into five or more equal parts. Continuous time averaging is well suited for improving overall signal-to-noise ratio, for continuous display and processing of data, and it is more efficient in using the computer resources. The described combination of hardware and software tools for the implementation of these techniques supplies continuously fresh data and does not require any synchronization with the main computer program. Compared to other digital DLTS systems, this new approach offers an improved short delay time resolution, improved signal-to-noise ratio, and more efficient data storage. At the same time it offers real-time observation of essentially noise-free transients and, like analog systems, a real-time display of the DLTS scan. The real-time display means that the displayed transient and DLTS scan can be updated on the computer screen after each pulse even for a very large number of averaged pulses, and the result can be predicted well before the acquisition process reaches this number. The proposed techniques allow one to combine the powerful transient analysis of the digital DLTS methods with the sensitivity and the convenience of the analog methods. The described averaging and data reduction techniques are intended primarily for DLTS data processing, but the same principles can be useful for many other physical experiments involving transient data analysis. © *1998 American Institute of Physics.* [S0034-6748(98)00106-3]

I. INTRODUCTION

Since the introduction of deep-level transient spectroscopy (DLTS) in 1974 by Lang,[1] the method has been refined many times with new techniques for improving the sensitivity and accuracy, for adapting DLTS to the specific properties of new types of samples, and for simplifying measurement and data analysis procedures. At present, DLTS is comprised of a large group of different measurement techniques applied to a variety of test devices and materials, and is considered to be among the most accurate and reliable tools for investigating the properties of electrically active point defects in semiconductors.[2,3]

Because of the great variety of DLTS techniques and numerous possibilities for their combination, any attempt at a detailed classification easily becomes a formidable task. Nevertheless, in all these techniques there are several common features identifying the technique as of DLTS type. First, it is a time domain measurement of a relaxation process, most often the emission of charges trapped in electrically active centers. Second, the process is considered to be thermally activated and the validity of the Shockley–Read–Hall theory[4] is assumed. Third, there is periodic alternation between filling of the traps with charge (filling pulse) and emission of the trapped charge (emission pulse). This last feature is used for synchronized measurement and integration—a basic method used for signal recovery from noise in electronic instrumentation.[5] The increased sensitivity, allowing detection of a signal immersed in noise, is the major advantage of DLTS over the previously used thermally stimulated current, or thermally stimulated capacitance methods for the study of deep levels in semiconductors.

[a]Electronic mail: kolev@bigfoot.com

II. TECHNICAL OVERVIEW OF THE DLTS EXPERIMENT

The DLTS experiment can be regarded as a sequence of several steps. The first step is to detect the change of the trap occupancy with electrons as a measurable change in some electrical parameter of the test structure. In classical DLTS, this is the small-signal high-frequency capacitance. The next steps are synchronous detection and averaging of this signal in order to improve the signal-to-noise ratio (SNR). There are two main approaches at this stage of the experiment—use of analog or digital signal processing methods.

A. Analog methods

The noisy transient signal can be measured and integrated over many pulses using analog instruments such as a boxcar averager,[1] a lock-in amplifier,[6] or an exponential correlator.[7] In the classical setup, two boxcar channels are used to measure the signal at two different times in the transient, and these times define a time constant window of the instrumentation.[1] The averaging is performed by the boxcar channels and the difference versus the sample temperature is recorded, giving a DLTS spectrum. When the emission time constant of a deep level coincides with that of the instrumentation while scanning the temperature of the sample, the output signal indicates this coincidence with a peak in the spectrum. To apply the Arrhenius relationship to determine the energy level and the capture cross section of a trap, several temperature scans are needed using different settings of the time constant window of the instrumentation. Similarly, it is necessary to change the pulse frequency when using a lock-in amplifier, or the reference time constant when an exponential correlator is used, and to repeat the temperature scan in order to obtain enough data for the Arrhenius plot technique. In all these cases, the averaging technique simultaneously achieves two different goals—increasing the SNR and analyzing the transient parameters.

One serious disadvantage of the analog techniques is the need to perform more than one temperature scan. This practical difficulty can be eliminated, for example, by using more complex (and more expensive) instrumentation, or by analyzing the shape of the DLTS spectrum[8,9] instead of just finding the peak temperature. Other disadvantages of the analog methods are their inherent limitations in resolving closely spaced energy levels of defects, and difficulties associated with the analysis of nonexponential transients. Nevertheless, the analog methods are simple to implement, they produce an analog signal in real time during the experiment, and plotting this signal versus the temperature gives spectrumlike curves which are available for immediate interpretation.

B. Digital methods

While the analog methods produce an integrated response to the transient signal, digital methods record the whole transient as a set of data points. The noisy signal is first digitized and then processed using various digital techniques.[10–20] The transient parameter analysis can take place during the experiment, or data can be stored for later analysis. In common with the analog methods described above, the transient can be digitally correlated with a boxcar, lock-in, or exponentially decaying function, and used to build a set of DLTS scans.[10] Alternatively, the transient can be analyzed using spectral analysis DLTS,[11,12] Fourier transform analysis,[13–15] nonlinear least square fitting,[16] the modulation function method,[17] the method of moments,[18,19] the correlation method of linear predictive modeling,[20] or other digital methods.[21] Since the whole transient is recorded, it is necessary to perform just one experimental temperature scan, which greatly reduces the time needed to perform the experiment. This is a major advantage over the analog methods, in addition to the extensive possibilities for analysis.

However, digital methods have disadvantages as well. Because the whole set of data points representing the DLTS transient is recorded for each temperature, the total number of stored data points can be very large compared to that recorded using analog methods. Consider for example a digitizer fast enough to study transients with time constants below 0.5 ms. Commonly available 12-bit analog-to-digital converters (ADC) with conversion time below 10 μs, which interface directly to a PC, are suitable for that purpose and their cost is below $10. If operated at rate of 33.3 kHz, the time spacing between the data points would be 30 μs and, for simultaneous observation of slow transients with time constants in the tens of milliseconds range, there will be several thousand samples for one transient recording. For example, if the emission pulse is 90 ms, there will be 3000 samples for each transient recording. If the transient is recorded in one degree intervals over a temperature range 50–350 K, there will be a total of 900 000 samples, which occupy 1.8 Mbytes of disk space (each 12-bit sample occupies two bytes).

The problem, however, is not so much the required disk space, but rather the difficulties in processing this large amount of data, since many of the digital methods involve iterative numerical calculations. The need to transfer 6 kbytes of data in just 90 ms is one more obstacle when the system is based on IBM PC and the processor operates in real mode, known also as MS-DOS mode. Another disadvantage of the digital methods, as compared to analog methods, is the decreased SNR for long delay times. Assuming constant noise, as the transient progresses, the signal decay leads to decreased SNR at the input of the signal processing apparatus. The analog methods have various ways of dealing with this problem. The boxcar technique enables one to compensate for the decreased SNR by increasing the sampling aperture of the channel which is recording the second time delay point.[18,22] In the lock-in technique, an integral of the transient is processed and the high SNR at the beginning of the transient is averaged with the low SNR at the end. The correlator reference function, which multiplies the noisy signal transient before integration, also decays along with the signal, thus minimizing the influence of the low SNR at the end of the transient. However, in a digital system, the sampling aperture of the ADC remains constant, and so the SNR decrease in the tail of the transient cannot be compensated for.

In general, both analog and digital DLTS signal processing techniques are mainly focused on the transient parameter

extraction and on the ability of the method to resolve closely spaced energy levels of defects. Averaging to improve the SNR is considered important, but is a secondary task. While in the analog techniques these two tasks are merged, in the digital techniques, the transient analysis is separate and usually performed *after* some simple digital averaging is done. Most often, this is the multiple time averaging of a selected number of successive transient recordings[10–12,14,16,18] and this number is usually in the range 100–300. This technique has three disadvantages. First, there is a need to allocate a memory buffer large enough to accommodate the selected number of transients. In the above example it takes more than 300 kbytes, or more than the half of the conventional memory of an IBM PC. Second, after averaging, the buffer is cleared for processing of the new transient, and the old information is lost, thus making it more difficult to maintain high SNR. Third, the progress of the summed transient is not monitored as it accumulates in the allocated buffer and, if anything goes wrong and some adjustment and a restart are needed, one is not aware of the problem until the full sequence of the selected number of transients has been accumulated.

In our approach[23] we consider averaging of the noisy DLTS signal as a *separate* task from the extraction of the transient parameters. We present two different, but complementary averaging techniques offering an increased time resolution for short delay times, improved signal-to-noise ratio for long delay times, and more efficient data storage. In addition, the transient and the DLTS scan are available in real time for continuous observation and data analysis after the acquisition of each single transient.

III. PSEUDOLOGARITHMIC AVERAGING

A. Theory

For detailed observation of an exponentially decaying signal, short sampling intervals are needed at the beginning of the transient, while much longer intervals are sufficient at the end. At the same time, high SNR of the processing instrumentation is much more important at the end of the transient, where the signal is weak, as compared to the beginning of the transient. One way to partly to fulfill these requirements is to use a high sampling rate and short emission pulse at high temperatures, and to lower the rate and increase the emission pulse width at low temperatures;[14,16] and this is easily done by using a programmable digital scope as a digitizer. Although this approach doesn't allow the simultaneous monitoring of both fast and slow transients, it is still possible to optimize the time resolution setting of the measurement system for a given time constant of interest.

A much better way to satisfy the time spacing requirements is to use sampling with a logarithmic time base. This approach is not new in DLTS experiments. Morimoto *et al.* described selecting from the measured 512 data points a set of 195 data points having almost logarithmic time distribution.[11–13] The data points in Ref. 24 also appear to have a logarithmic time spacing. To our knowledge, the best published work on this subject is described in Ref. 10 where is used a high, but constant ADC sampling rate, all sampling results are transferred to the computer memory, and pseudologarithmically spaced data points are selected by specially written software. The pseudologarithmic storage scheme in Ref. 10 uses a logarithmic with base 2 because of the ease of unsigned integer division by 2^n in binary arithmetic.[25] This simplicity is equally advantageous in hardware logic circuits and is used in our scheme as well. Another, probably less obvious advantage is the fact that the function 2^n is more closer to $\exp(n)$ than the commonly used 10^n. In our system, the sampling rate is 100 kHz and therefore, the sampling intervals Δt are spaced 10 μs apart. If the data is averaged over time intervals t_n, with each one twice as long as the preceding one, the second averaging interval is $2\Delta t$, and the nth averaging interval starts at

$$t_n=(2^n-1)\Delta t. \tag{1}$$

If, for convenience, we number the intervals starting from zero, which is a commonly used standard in programming and digital electronics, for $2^n \gg 1$ the index n can be expressed as

$$n \approx \log_2(t_n/\Delta t). \tag{2}$$

This represents a logarithmic dependence of the interval index on the interval time length, since the sampling interval Δt is fixed by the constant sampling rate of the ADC.

Unfortunately, the pseudologarithmic storage scheme proposed in Ref. 10 has several disadvantages. First, it is still necessary to allocate very large memory buffers, as the data points are selected after the full set of transient sampling points is recorded into the computer memory. Sampling each 10 μs, there are 131 072 sampling points which are stored into a 256 kilobyte buffer. From this large set of data, only 768 data points are selected and used, while the remaining 130 304 points are discarded. Second, more buffers are needed for the multiple time averaging of successive transients as described above, and the result is available for observation *after* the selected number of transients are averaged. Third and most important, there is a substantial decrease in the SNR, especially in the tail of long transients, where the data point intervals are in the millisecond range, because the data points are selected from just 10 μs long sampling intervals. In addition, recording of 768 data points for each degree in the 50–350 K range still requires almost 0.5 Mb disk space and direct memory access (DMA) transfer into memory blocks larger than 64 kbytes is complicated, since it requires continuous initialization of the DMA controller for crossing the memory page boundary.[26]

The main difference between our pseudologarithmic averaging scheme and that described in Ref. 10 is the method of obtaining the data points. In Ref. 10, the data points are *selected* from the large set of sampling points, but in our system, the data points are *averages of the sampling points* inside the averaging intervals, which increase in a pseudologarithmic manner.[27] In this way, *all measured samples are used.* Although our electronic circuit implementation is different (Fig. 1), the averaging principle (Fig. 2) is the same as in Ref. 27. The logarithmic time spacing algorithm described above gives only about three data points per decade. To increase this number, the length of the averaging

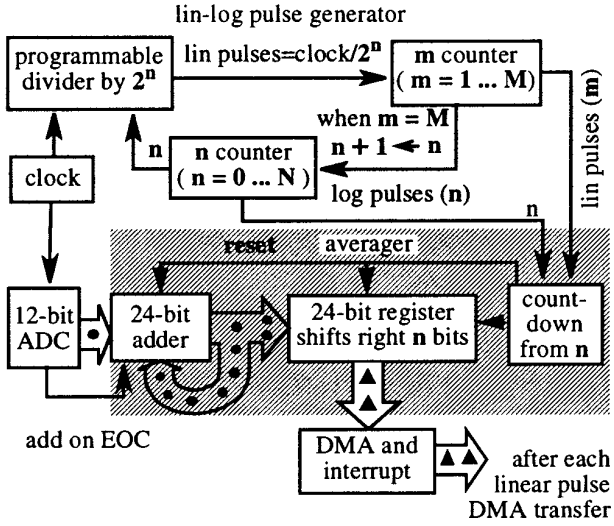

FIG. 1. Block diagram representation of pseudologarithmic time base averaging circuit.

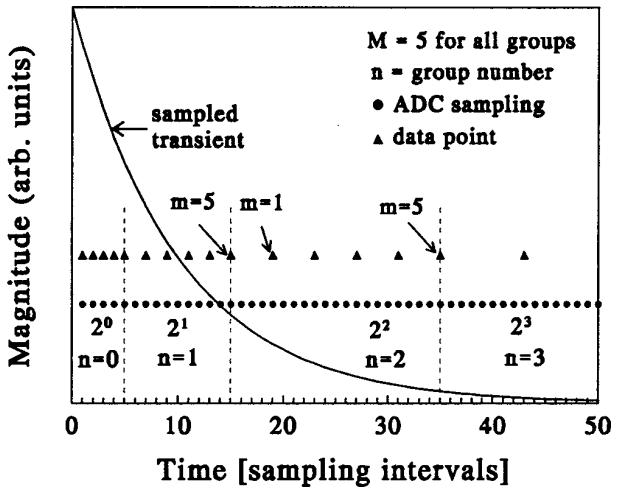

FIG. 2. Pseudologarithmic time averaging scheme. After each fifth data point the number of averaged samples 2^n doubles as n increases by 1.

interval in the pseudologarithmic averaging scheme is doubled after a preselected number M of averaging operations, thus producing a linearly spaced train of M averaging intervals. In this way, the pure logarithmic time spacing is mixed with linear time spacing, and hence the name *pseudologarithmic*. The sampling interval remains the same, 10 μs in our system, and the result is a sequence containing $(N+1)$ groups of averaging intervals m_n. Inside the group n, each interval m_n has the same length, but it is twice as long as an interval from the preceding group m_{n-1}, or half of the time of an interval from the succeeding group m_{n+1}. Since the maximum number M of averaging intervals m_n inside each group is the same for all the groups, the time lengths of the groups change in the same way. Therefore, we have $(N+1)$ groups each one containing M equal averaging intervals m_n, but with the time length of the groups increasing as 2^n.

B. Errors analysis

At a first glance, this scheme for averaging of the sampling points may be understood as an approximation of the exponentially decaying function with a set of straight lines connecting the obtained data points (piecewise linear approximation). Indeed, the obtained set of data points does not contain explicitly any information about the time intervals between the data points—this information remains hidden in the summing results from which the data points were obtained. However, the data points themselves were obtained by summing of *the true experimental signal*, sampled each Δt, and dividing the result by the number of the samples in the averaged interval. For a given group with an index n, the largest error is always expected to occur in the first of the averaging intervals m_n with $m=1$, where the signal varies more rapidly as compared to its variation during the subsequent intervals from the same group with $m>1$. Therefore, we need to consider only the worst case when the averaging is done from $ts_{n,M}=\Delta tM(2^n-1)$ to $te_{n,M}=\Delta tM(2^n-1)+\Delta t2^n$. The result of the averaging process is assigned to the middle of the averaging interval $tm_{n,M}=\Delta tM(2^n-1)+\Delta t2^{n-1}$. Thus, the error function $\mathrm{Err}(\tau)$ can be obtained by subtracting the magnitude obtained by averaging of the exponentially decaying function in this interval from the exact magnitude in the midpoint $tm_{n,M}$:

$$\mathrm{Err}(\tau)=C\,\exp\left[-\frac{tm_{n,M}}{\tau}\right]-\frac{C}{2^n}\exp\left[-\frac{ts_{n,M}}{\tau}\right]\times\sum_{i=1}^{2^n}\exp\left[-\frac{\Delta t}{\tau}\,i\right], \tag{3}$$

where C is the transient magnitude at the beginning of the transient. When $\mathrm{Err}(\tau)$ is plotted versus τ for a given set of Δt and M, this function exhibits one sharp maximum, dominating at short time constants for small n at some τ_{m+}. At long time constants, this maximum is compensated for by another sharp minimum which begins to dominate for large n at some $\tau_{m-}<\tau_{m+}$. Away from these peaks, the error function is close to zero. We find these particular τ_{m+} and τ_{m-} by setting the first derivative versus τ equal to zero and solving numerically with convenient choice of the initial guess value for τ:

$$\frac{-C\Delta t}{\tau^2\left[\exp\left(\frac{\Delta t}{\tau}\right)-1\right]^2}\left[(\gamma_{n,M}-1)\exp\left(-\frac{te_{n,M}}{\tau}\right)-(\gamma_{n,M}-2^{-n}-1)\exp\left(-\frac{te_{n,M}+\Delta t}{\tau}\right)\cdots+(\gamma_{n,M}-2^{-n})\times\exp\left(-\frac{ts_{n,M}+\Delta t}{\tau}\right)-\gamma_{n,M}\exp\left(-\frac{ts_{n,M}}{\tau}\right)\right]-\frac{Ctm_{n,M}}{\tau^2}\exp\left(-\frac{tm_{n,M}}{\tau}\right)=0, \tag{4}$$

where $\gamma_{n,M}$ is $M(2^{-n}-1)$.

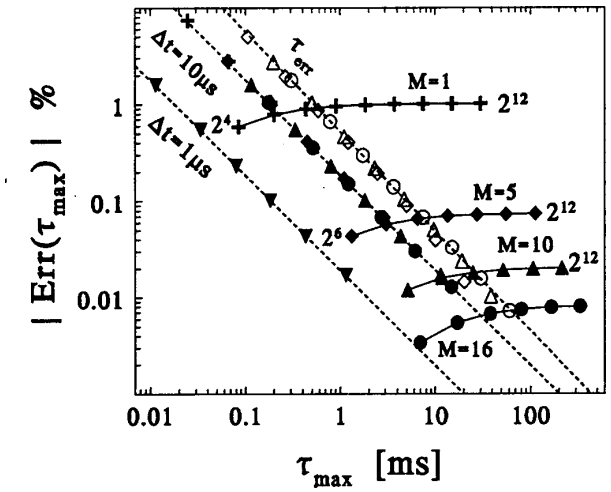

FIG. 3. Normalized absolute magnitude and time constant errors plotted vs the time constant τ_{max} at which these errors occur. The straight lines indicate independence from the group number n or number of equal averaging intervals inside the group M and dependence only on the sampling rate. Inverted triangles indicate sampling with 1 MHz rate and $M=10$. The right line gives the errors in determining the time constant sampling with 100 kHz.

Next, we replace the obtained values for τ_{m+} and τ_{m-} in Eq. (3) which gives the maximal errors $\mathrm{Err}(\tau_{m+})$ and $\mathrm{Err}(\tau_{m-})$ for using the averaged values in the m_n interval with $m=1$. This function was normalized versus the initial magnitude C and plotted against τ_{m+} and τ_{m-} in Fig. 3 with parameters n and M for two values of Δt. There are two distinct reasons for the errors in this pseudologarithmic scheme.

First, the limited speed of the ADC gives an error which is strictly dependent on the sampling interval Δt, in our case 10 μs, and does not depend on the number of the averaged samples 2^n or the number of the equal intervals inside the group M. This gives the straight line which demonstrates increased error for short time constants. Obviously, this error is not inherent to the pseudologarithmic averaging, and can be decreased only by using faster ADC. The second type of error is more pronounced for longer time constants and depends much more on M than on 2^n. Figure 3 demonstrates that for time constants longer than 1 ms the errors are below 0.1% if the groups are divided into five or more averaging intervals. Calculations with a ten-times faster ADC sampling rate show a ten-time reduction of the errors for fast time constants.

For the transient analysis, it is attractive to use a boxcar function obtained simply by subtracting two data points with time delays t_1 and t_2. Since each data point represents an average with time width W, it is interesting to evaluate the error in defining the rate window. We follow the procedure outlined in Ref. 22 with a slight modification. Instead of integral averages, we use summing averages in order to model accurately the actual function of our circuit and to account for the limited rate of the ADC. Furthermore, we subtract only the first subintervals in two adjacent groups where the maximal error is expected to occur. The normalized output $S(\tau)$ is then given by

$$S(\tau)=\frac{C}{2^n}\exp\left[-\frac{tm_{n,M}}{\tau}\right]\sum_{i=1}^{2^n}\exp\left[-\frac{\Delta t}{\tau}i\right]-\frac{C}{2^{n+1}}\times\exp\left[-\frac{tm_{n+1,M}}{\tau}\right]\sum_{i=1}^{2^{n+1}}\exp\left[-\frac{\Delta t}{\tau}i\right]. \quad (5)$$

We find the rate window by differentiating with respect to τ and setting the result equal to zero. The solution can be found numerically to give τ_{max}. We calculated τ_{max} for variations of n and M and compared the values with these obtained using the Lang's expression using the midpoints of the averaged intervals, which is

$$\tau'_{max}=\frac{tm_{n+1,M}-tm_{n,M}}{\ln(tm_{n+1,M}/tm_{n,M})}. \quad (6)$$

The normalized error in determining $\tau_{err}=(\tau'_{max}-\tau_{max})/\tau_{max}$ is also shown in Fig. 3. One should mention that because of the logarithmic dependence from τ on the Arrhenius plot, errors for τ in the range of 1% can be ignored.[22]

C. Implementation

There are two possible ways to implement the averaging scheme described above. It might be possible to use the same hardware as described in Ref. 10. Instead of selecting only 768 data points from the whole set of sampled points, fast software could be used to change the length of the averaging interval, perform the actual averaging of the sampled points inside the interval, and store the corresponding data points. There are two potential problems with this approach. First, there may be some difficulties with the processing during the first three groups, when there is no averaging or only two or four samples are averaged per data point. In real mode of the processor, fast data transfer can only be done with a DMA. Since the DMA process actually ''steals'' cycles from the PC processor, an intensive DMA transfer can considerably slow down the program execution.[27] Thus, the DMA transfer can cause timing conflict with the real-time averaging routines. Therefore, we give preference to the hardware implementation shown in Fig. 1.

In our circuit, one can distinguish four functional blocks: an ADC, a pseudologarithmic pulse generator, an averager, and an interfacing block. The ADC operates continuously at a fixed rate of 100 kHz. The pseudologarithmic pulse generator is constructed from programmable counters and shift registers. It incorporates a programmable divider of the clock frequency which is set at the beginning of the transient to divide by 2^0, since for the first group $n=0$. Of course, in practice this means no division at all, since $2^0=1$. When the counter of linear pulses m reaches predetermined number M, it is cleared to start counting again, and this produces a logarithmic pulse n, thus changing the setup of the programmable divider to 2^1. This doubles the time between each one of the next M pulses. When the linear pulse counter reaches M again, n increments by 1, and this increase sets up the divider to divide the clock pulses by 2^2. This sequence is repeated until the logarithmic pulse counter reaches the software-programmed number of groups N.

The averager is made by combining an accumulator, shift register, and a programmable down counter. The end-of-conversion signal from the ADC triggers an adding operation which adds the result from the current conversion to the stored sum of the previous conversions. This accumulation of conversion results is repeated over the whole length of the averaging interval. When the new linear pulse arrives, it stores the accumulated result into the shift register, which must be large enough to accommodate the whole sum. Then the accumulator is cleared for storing the new sum while the result in the register is divided by 2^n by being shifted right n bits by the down counter. Since the down counter is programmed by the same logarithmic pulse counter n, which is programming the clock frequency divider to divide by 2^n, the accumulated results are always properly divided.

After the averaging operation is over, an interfacing circuit signal requests DMA controller of the PC to transfer the averaged result to the corresponding memory location. After the last result from the last octave is transferred to the PC memory, the logarithmic pulse counter is cleared and ready for the next transient. Simultaneously, the interfacing circuit produces a signal which triggers a hardware interrupt routine for performing the second averaging technique described later in this work. The operation of the whole circuit is synchronized by a 16 MHz system clock, and the ADC sampling frequency is obtained by dividing the system clock frequency by 160.

The maximum allowed numbers for N and M depend on the hardware circuit. In our system, $N_{\max}=12$ and $M_{\max}=16$ which gives a total of 208 pulses (13×16, since N can be 0). This is enough to store transients up to 1.3 s long, with 10 μs and 20 μs resolution, and with 32 data points in the first 0.5 ms. In addition, we have a programmable delay of up to 160 μs in order to compensate for possible slow capacitance meter response. The intervals in the last group are 40.96 ms long, each one averaging 4096 samples. Therefore, the expected SNR improvement is $\sim\sqrt{4096}$ or 64 times.[5] However, it refers to points *inside* the averaging interval and not for the whole transient which implies that the SNR improvement will be less for low-frequency noise. If we use the averaging scheme proposed in Ref. 10, then each data point from the last group (the second half of the transient), would be just one selected ADC sampling result out of 4096 ADC samples. In our averaging scheme that same data point represents an average of 4096 ADC samples. That gives a major advantage of our averaging technique for long delay times over the previously used technique. Another advantage is the storage efficiency. Because of the significant SNR improvement, even long transients can be recorded with just 208 points and the disk storage space for a DLTS scan in the range 50–350 K occupies less than 125 kbytes. Other advantages are the reduced size of the required buffers, in our system we allocate just two 420-byte buffers, and the large amount of available time for the microprocessor to process and display the results while the measurement is in progress.

D. Demonstration

In order to demonstrate the benefits of using the proposed pseudologarithmic averaging scheme, we recorded a real DLTS signal when our system was operated in constant-capacitance (cc) mode. We used an analog memory feedback[28] which eliminates the distortion of the voltage transient introduced by the integration of the voltage step from the filling to the emission pulse, and provides a stable zero baseline.[29] This feedback was used in combination with a Boonton 7200 capacitance meter. The measured sample is a p-Si metal-oxide-semiconductor capacitor with insufficient annealing after the ion implantation for correction of the threshold voltage. More details can be found elsewhere.[28] The traces in Fig. 4 were recorded using DMA transfer and a commercially available data acquisition board. Trace (a) appears in the same way as the signal seen on the oscilloscope screen, and trace (b) was obtained with multiple time averaging of 32 consecutive transients. This trace gives an idea of the expected improvement in SNR using a simple, multiple time averaging technique. It should be noted that real-time observation of these traces on the computer screen was impossible. Also, we demonstrate our system mainly with a relatively weak DLTS signal and with a presence of 60 Hz electromagnetic interference in order to better demonstrate the complementary action of both averaging techniques presented in this work. For the same reason, we intentionally recorded transients longer than normally used in DLTS. In Fig. 5 are shown four traces. The trace denoted with $k=0$ was obtained using only pseudologarithmic averaging. At first glance, the advantage of using this technique alone is not obvious. Although the noise magnitude inside the trace is significantly reduced, the shape is substantially different from the expected exponentially decaying transient. The reason for this unsatisfactory result is because the pseudologarithmic averaging technique is used *inside* the averaging intervals and therefore, cannot suppress low frequency noise components. However, note the transition to longer averaging intervals around 320 ms and its effect on the curve shape. In order for the above digital averaging circuit to operate properly, it must be assured that essentially all the averaged transients are inside the limits of the ADC. We use a digital-to-analog converter to display the averaged data points on

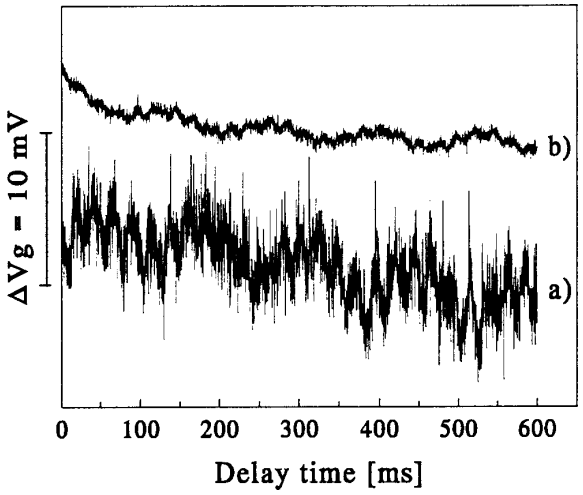

FIG. 4. (a) Original signal sampled each 0.2 ms with 12-bit resolution; (b) same signal after multiple time averaging of 32 consecutive transients.

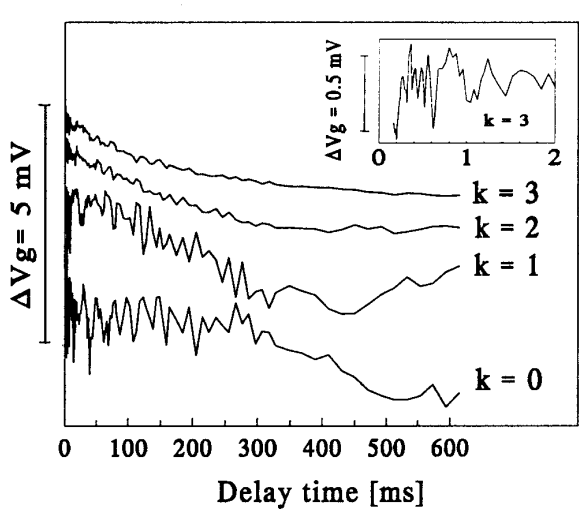

FIG. 5. The signal of Fig. 4 trace (a) after pseudologarithmic time averaging alone for $k=0$. For $k>0$, pseudologarithmic time averaging is combined with continuous time averaging with time constants 2^k. The inset shows good time resolution during the first two ms with $k=3$.

the oscilloscope screen, which helps us to detect ADC input overloading.

IV. CONTINUOUS TIME AVERAGING

A. Theory

The pseudologarithmic averaging proposed above is a very efficient technique for reducing the number of data points and for SNR improvement at relatively high frequencies and large delay times. However, it needs to be complemented with another averaging technique which can suppress low frequency noise components and improve the SNR at the beginning of the transient. We found that continuous time averaging[5] is a very convenient technique for these purposes. In addition, it has the advantage of allowing continuous transient display after each pulse and the size of the allocated memory buffer is independent of the number of averaged transients. The continuous time averaging mode is similar to the running average formed by a low-pass filter. In this mode, the result of the *last* n_t transients is available at any time for display and for other data processing. Unfortunately, it is not convenient to apply the low-pass filter directly for averaging of multiple transients, because the data points that this filter averages are consecutive in time. In contrast, averaging of multiple transients processes the whole set of transient data points for each transient in parallel, i.e., each data point is averaged with corresponding data points from the other transients which are at the same delay from the start of the transient. In this way the data points to be averaged are separated by one or more pulse periods and are not consecutive in time. Therefore, we need a method to adapt the low-pass filter function to be applied separately for each data point of the transient.

For a simple asymmetrical first order low-pass filter consisting of only a resistor R and capacitor C, the voltage increment ν_0 is[5]

$$d\nu_0/dt=(\nu_{\text{in}}-\nu_0)/T_c, \tag{7}$$

where $T_c=RC$ is the filter time constant. Let us consider ν_{in} digitized at short sample intervals δt. Then in digital form, Eq. (7) becomes

$$\Delta\nu_0/\delta t=(\nu_{\text{in}}-\nu_0)/n_t\delta t, \tag{8}$$

where $n_t\delta t$ is the digital time constant corresponding to the analog time constant RC. Therefore, in order to implement a low-pass filter function on each data point of the transient, we have to replace the simple summing and dividing algorithm with

$$\nu_0\leftarrow\nu_0+(\nu_{\text{in}}-\nu_0)/n_t, \tag{9}$$

which means that the value ν_0 of the data point stored in the computer memory buffer is updated after each new transient, with a fraction of its difference from the corresponding data point ν_{in} from the new transient.

If the starting value stored in the buffer is zero, then in the beginning of the averaging process the difference is large, and the value in the buffer quickly grows as a new value is added to it after each new transient. As the value stored in the buffer approaches that of the new transient, the growth rate decreases and eventually when the stored value is nearly equal to that of the new transient, the growth terminates. After this, the value in the buffer becomes largely stable and it reflects only changes in the incoming transient which are sustained long enough to be comparable to the digital time constant. This evolution of the transient stored in the computer memory buffer is continuously monitored on the computer screen where each data point looks like it was produced by a separate, virtual boxcar channel. This is not a surprise, since a real boxcar channel performs exactly the same averaging operation. Of course, this virtual boxcar channel has some limitations when it is compared to the real one. Each virtual channel has a fixed time delay corresponding to the data point, and aperture corresponding to the length of the pseudologarithmically averaged interval. Fortunately, this aperture is conveniently self-adjusted depending on the time delay for maintaining a high SNR throughout the transient. In addition, new ''channels'' can be created easily, and selecting the averaging time constant $n_t\delta t$ can be fully automated and varied throughout the experiment. Since δt remains constant, only n_t is varied. For convenience, in our system, n_t is represented as 2^k and the actual change of the digital time constant is made by selecting k. Since the SNR increases with the square root of the number of averaged transients it is expected that the SNR will depend linearly on k.

B. Demonstration

The results of the combined action of both techniques (pseudologarithmic and continuous time averaging with variation of k) are shown in Figs. 5 and 6. A high time resolution in the beginning of the transient is seen in the inset in Fig. 5, and a very good SNR is demonstrated in the inset in Fig. 6. The trace with $k=2$ seems almost identical to that with $k=3$, but in fact it contains more noise at low frequencies comparable to the pulse period. Note the difference in the second half of the transient. Also, the trace with $k=4$ in

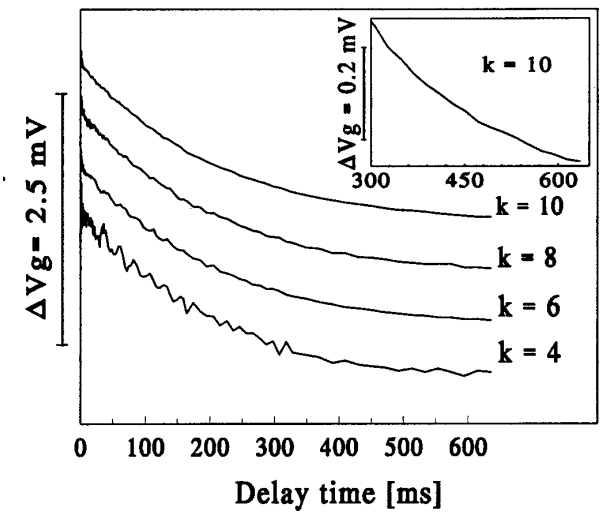

FIG. 6. Combined pseudologarithmic and continuous time averaging with different time constants 2^k. The inset shows very good SNR at the end of the transient with $k=10$. Input signal is seen in Fig. 4.

Fig. 6, compared to the trace with $k=3$ in Fig. 5, appears more noisy because the scale of the vertical axes are different. The noise fluctuations in the inset in Fig. 6 can be estimated to be in the range of 10 to 30 $\mu V_{p\text{-}p}$. Compared with the noise magnitude as seen in Fig. 4(a), which is about 10 $mV_{p\text{-}p}$, the SNR improvement is substantial.

The 12-bit resolution of the ADC (or that of the pseudologarithmic circuit) is always much better than the vertical resolution of the computer screen; therefore, digitization errors are not visible. In addition, the 12-bit resolution is effectively boosted by the averaging process to 16-bit resolution, since the computer operates with 16-bit ''words''.[5,10] The continuous time averaging routine[30] is activated by a hardware interrupt immediately after the transfer of the last pseudologarithmically averaged data point of each transient. For each data point of the received transient, the subprogram shifts left four bits, which is equivalent to multiplying by 16. Next, it subtracts the corresponding 16-bit value, stored in the reference memory buffer, then shifts right k bits and finally adds the result to the old buffer value to be stored as a new value. This is exactly the low-pass filter algorithm implied in Eq. (9). Because the averaging is performed over 16-bit digits, the digital time constant can be as large as $2^{15}-1$ or 32 767, which means the maximum increment or decrement to the 16-bit value stored in the buffer is just one bit. However, in practice this is a very inconvenient choice. Consider, for example, a relatively fast transient recorded during a 20-ms-long emission pulse period. There will be just 50 averaging steps per second, and to change the stored value e times will require 32 767 s/50 or about 11 min. To reach the new value within 1% error, one has to wait almost 1 h. Besides, a 20-ms-long emission pulse period does not allow the pseudologarithmic averaging technique to be used efficiently.

V. APPLICATIONS

A. Long transients

The averaging techniques suggested above are not limited only to the DLTS measurements. The main goals achieved by these techniques are substantial SNR improvement and the efficient reduction of the number of data points. In DLTS, these techniques should be considered as preprocessing steps followed by the actual transient parameter analysis, which can use all digital processing methods listed earlier.[10–20] Furthermore, because of the reduced number of data points and increased SNR, it can be expected that these techniques will give better results.

Alternatively, the weight functions well known in the analog methods, such as boxcar, lock-in amplifier, and exponential correlator, can also be used for the transient analysis as demonstrated in Ref. 10. When signal-to-noise ratio is not of concern, the boxcar function is particularly attractive since it gives higher resolution of the peaks. In our system, we rely mostly on the use of the boxcar function with time delays $t_2=2t_1$ by simply subtracting two selected data points with the same index m but from different groups n and $n+1$. The corresponding DLTS scan is displayed in an inset on the computer screen during the thermal scan, overlapping the transient display. The time delays can be varied and the resulting DLTS scans can be seen in real time. Thus both the transient signal and the DLTS scan are updated on the computer monitor after each pulse like on an oscilloscope screen. Since the data points for the Arrhenius plots can be obtained during the experiment, the analysis of a given trap can start as soon as the scan of the temperature interval corresponding to the measurable trap time constants is over.

Figure 7 shows several transients recorded for each degree of temperature change. The magnitudes of the traces were adjusted to fit in the same chart and are not to scale. This adjustment was made in order to demonstrate the benefits of the very high SNR which allows for making obvious the observed changes in the time constant. These results sug-

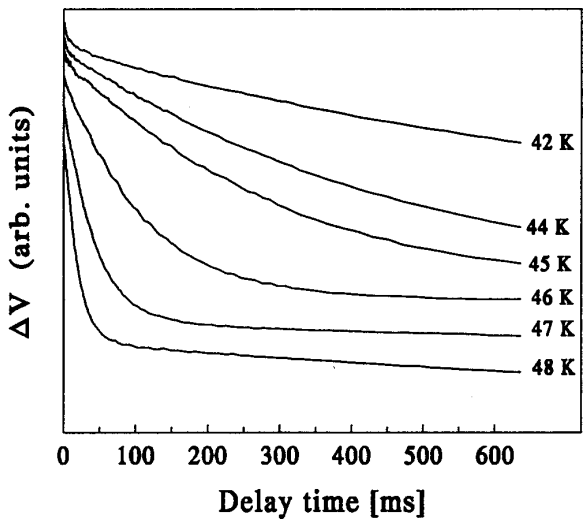

FIG. 7. Temperature variation of the transient time constant as observed during measurements. The curves demonstrate the need for better temperature resolution at low temperatures. Magnitude is not to scale.

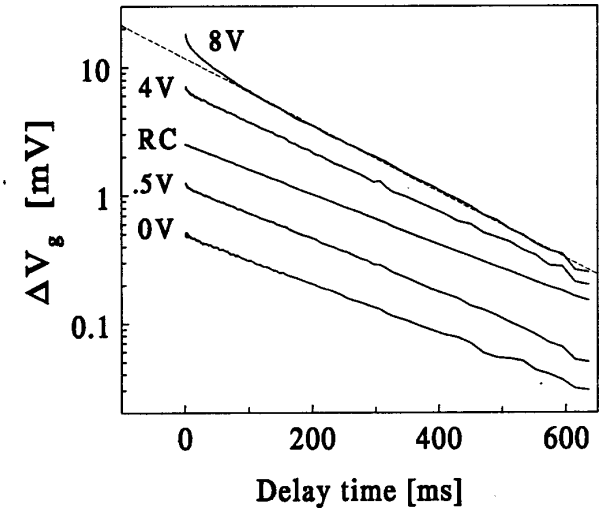

FIG. 8. Time constant variation at fixed temperature of 47 K and variable reverse bias 0–12 V. The time constants calculated from the slope of the trace vary from 200 ms at zero bias to 158 ms at 4 V. Note the departure from the exponential behavior at 8 V bias during the first 50 ms. For comparison is shown a transient obtained by differentiating a square wave with an *RC* high-pass filter.

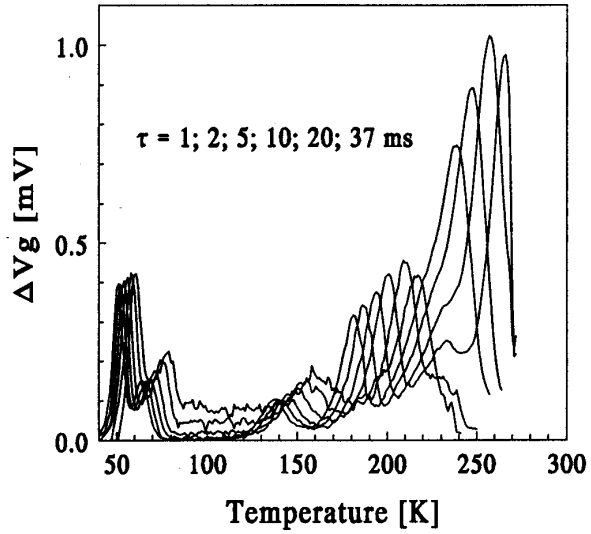

FIG. 9. Series of six boxcar DLTS spectra with channel delay ratio $t_2 = 2t_1$ obtained from one thermal scan. The change in the background level seen at 100 K is for $\tau = 1$ ms and $\tau = 2$ ms. The change in the magnitude for longer time constants is probably due to temperature variation of the capture cross section.

gest the option of using isothermal analysis DLTS[31,32] which can minimize the effects of thermal dependencies of the capture cross section and transient magnitude. For this purpose, we need to determine the emission rate without relying on its temperature dependence, for example, by displaying the transient on a logarithmic scale.

In Fig. 8 are shown several transients at the same temperature of 47 K, but with different settings of the reference capacitor corresponding to different reverse steady-state bias voltages. For comparison, we recorded also a true exponential transient with the magnitude and the time constant adjusted to be close to the real DLTS signal. This true exponential transient was obtained by differentiating a square wave with an appropriate RC filter. The slope of the traces gives time constants decreasing from 200 to 158 ms. The decreased transient time constants, corresponding to low values of the reference capacitor or higher reverse bias demonstrate the effect of field-enhanced emission.[33] It is important to note here that the change in the slope, and, respectively, in the transient time constant, can be easily monitored at a fixed temperature, and this allows for the adjustment of the measurement setup during the experiment. For example, by adjusting the voltage and monitoring the slope of the trace, we can avoid the conditions of field-enhanced emission, and at the same time, obtain a large signal which improves the sensitivity of the measurement.

B. DLTS spectra

In the measurements in Figs. 5–8, the pseudologarithmic settings were $N = 11$ and $M = 16$, the temperature variation was limited between 40 and 50 K, and all transients were associated with just one trap. Indeed, with less than two transient recordings per second and the digital time constant set to 2^9, about 600 s were needed to obtain good averaging results. Clearly, these settings are not very convenient for a full-range thermal scan, and this setup is better suited for isothermal measurements when a significant noise reduction is essential. When the signal level is high compared to the noise, we set the digital time constant to 2^4 through 2^6 and then we can store transients up to 1.3 s long for full range of thermal scan. For scanning temperature measurements of noisy transients, we use a setup of $N = 8$ and $M = 10$ which is enough to record a 50 ms long transients with just 90 points. With about 20 continuous time averages per second, this setup allows us to use digital time constants up to 2^{10}, depending on the temperature scanning rate. For example, if the setup is 2^{10}, then to change the values of the stored transient e times takes less than a minute. Recording the transient each degree, and with temperature scanning rate of 0.8 K/min or less, there is still enough time for averaging, since the time constant of the transient varies far less than e times for each degree of temperature change. This is true especially at temperatures above 100 K where the DLTS signal varies slowly with the temperature. Obviously, the scanning rate of the temperature has to be conveniently selected for a given averaging time constant, but this is a simple requirement for using any analog system using a boxcar averager or lock-in amplifier.

Using this setup and the simple boxcar technique described above, a series of six DLTS spectra were obtained (Fig. 9). There are five well-defined peaks present. The shape variation of the large peak at the highest temperature suggests the existence of a sixth peak which is dominated by the large peak. The peak appears more on the curves corresponding to a short time constant setup and this is a rather strange result because it is well known that the DLTS peak resolution improves for setups of long time constant windows.[10] However, further analysis shows, that this false peak can be associated with the signal overloading, which affects the first half of the stored transient more than the second half.

In Fig. 10 are shown the Arrhenius plots obtained using the same data file (Fig. 9). As outlined in Ref. 2, obtaining

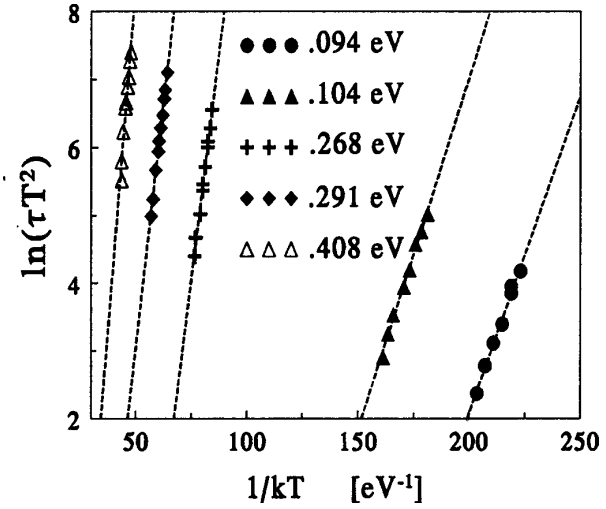

FIG. 10. Arrhenius plots obtained from the scan data file used in Fig. 8. The energy level positions of the hole traps are in eV above the valence band as indicated. Note the good linearity of the data points.

the energy level positions and capture cross sections is only the first step toward impurity or defect identification. We limit our work here to demonstration of the averaging techniques. The deep level analysis is the subject of further research. The trace in Fig. 8 with lowest influence of the field-enhanced emission is fitted well by an exponentially decaying signal with a time constant of about 200 ms. In the rightmost Arrhenius plot in Fig. 10 for the trap around 50 K, this time constant corresponds to 47.6 K. This comes to support once more the need to record more than one transient for each degree at low temperatures.

Using setup of $t_1 = 10.3$ ms and $t_2 = 20.4$ ms for the rate window, we have found the peak temperatures of the traps appearing in Fig. 9. The recorded transients at these temperatures are shown in Fig. 11. Signal overloading of the transient recorded at 250 K is seen at the beginning of the trace for delay times less than 5 ms. This is the reason for the false peak appearance on the low-temperature side of the large

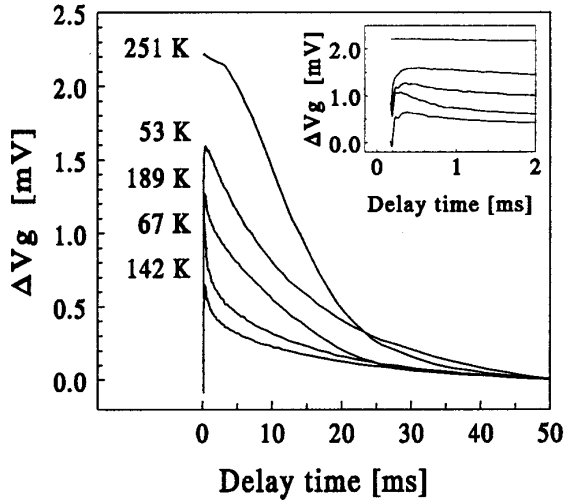

FIG. 11. Series of transients at the peak temperatures for setup $t_1 = 10.3$ ms and $t_2 = 20.4$ ms of the traps in Fig. 8 (same data file). Details of the first two ms are shown in the inset. Note the overloading at 251 K.

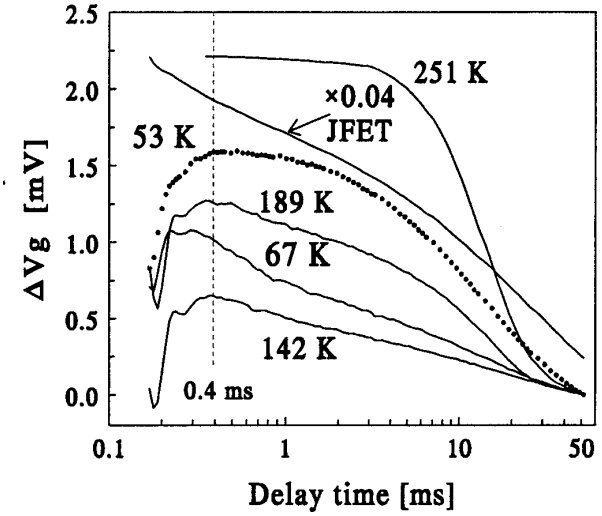

FIG. 12. The data of Fig. 11 plotted on a semilogarithmic scale to display details at the beginning of the transient. The signal before 0.4 ms is distorted because of the instrument recovery after the filling pulse. The dots in the trace at 53 K show the data points evenly spaced on a logarithmic scale. For comparison is shown a CC-DLTS trace obtained from neutron irradiated junction field-effect transistors and reduced 25 times in magnitude.

peak in Fig. 9. Therefore, the observation of the whole transient can easily prevent the incorrect interpretation of the shape variation mentioned above. In the inset are shown details of the same transients during the first 2 ms. This can further improve the rejection of incorrect data for transient analysis. One simple way to check the transients for hard-to-see distortions at the beginning is to display the transient on a logarithmic time scale.

In Fig. 12 is shown the same series of transients as in Fig. 11, but on a logarithmic time scale. Here one considers the data above 0.4 ms, since below this limit the recorded data points reflect the capacitance meter recovery process. The switching time of the feedback circuit is below 1 ms and it does not introduce any significant delay. Another possible source of relatively slow recovery time is the high series resistance of the sample. From the point of view of demonstrating the averaging techniques, it is interesting to note the smooth shape of the curves during the instrument recovery. This is possible using only very high time resolution in the beginning of the transient capture and with a good SNR. One of the curves is presented as a set of data points in order to demonstrate their even spacing on a logarithmic time scale. For comparison, another constant capacitance-DLTS trace recorded with the same setup is also shown. The measured sample was a junction field-effect transistor subjected to neutron irradiation. This trace proves that the feedback and the biasing circuit are not the source for the distortion in the remaining traces.

ACKNOWLEDGMENTS

This work was partially supported by a G.R.E.A.T. scholarship from the Science Council of British Columbia, by NSERC of Canada, and by the Center for Systems Science at SFU.

[1] D. V. Lang, J. Appl. Phys. **45**, 3023 (1974).
[2] J. L. Benton, J. Cryst. Growth **106**, 116 (1990).
[3] D. K. Schroder, *Semiconductor Material and Device Characterization* (Wiley, New York, 1990), Chap. 7.
[4] W. Shockley and T. W. Read, Phys. Rev. **87**, 835 (1952); R. H. Hall, *ibid.* **87**, 387 (1952).
[5] T. H. Wilmshurst, in *Signal Recovery from Noise in Electronic Instrumentation*, 2nd ed. (IOP, Bristol, England, 1990), p. 29.
[6] D. Pons, P. M. Mooney, and J. C. Bourgoin, J. Appl. Phys. **51**, 2038 (1980).
[7] G. L. Miller, J. V. Ramirez, and D. A. H. Robinson, J. Appl. Phys. **46**, 2638 (1975).
[8] H. P. Hjalmarson and G. A. Samara, J. Appl. Phys. **63**, 1801 (1988).
[9] Z. Su and J. W. Farmer, J. Appl. Phys. **68**, 4068 (1990).
[10] W. A. Doolittle and A. Rohatgi, Rev. Sci. Instrum. **63**, 5733 (1992).
[11] J. Morimoto, M. Fudamoto, K. Tahira, T. Kida, S. Kato, and T. Miyakawa, Jpn. J. Appl. Phys., Part 1 **26**, 1634 (1987).
[12] J. Morimoto, M. Fudamoto, S. Tashiro, M. Arai, T. Miyakawa, and R. H. Bube, Jpn. J. Appl. Phys., Part 1 **27**, 2256 (1988).
[13] K. Ikeda and H. Takaoka, Jpn. J. Appl. Phys., Part 1 **21**, 462 (1982).
[14] M. Okuyama, H. Takakura, and Y. Hamakawa, Solid State Electron. **26**, 689 (1983).
[15] S. Weiss and R. Kassing, Solid State Electron. **31**, 1733 (1988).
[16] T. R. Hanak, R. K. Ahrenkiel, D. J. Dunlavy, A. M. Bakry, and M. L. Timmons, J. Appl. Phys. **67**, 4126 (1990).
[17] S. M. Ransom, T. I. Chappell, J. L. Freeouf, and P. D. Kirchner, Mater. Res. Soc. Symp. Proc. **69**, 337 (1986).
[18] P. D. Kirchner, W. J. Schaff, G. N. Maracas, L. F. Eastman, T. I. Chappell, and S. M. Ransom, J. Appl. Phys. **52**, 6462 (1981).
[19] K. Ikossi-Anastasiou and K. P. Roenker, J. Appl. Phys. **61**, 182 (1987).
[20] F. R. Shapiro, S. D. Senturia, and D. Adler, J. Appl. Phys. **55**, 3453 (1984).
[21] P. R. Bevington and D. K. Robinson, *Data Reduction and Error Analysis for the Physical Sciences*, 2nd ed. (McGraw-Hill, New York, 1992). See also C. E. Reid and T. B. Passin, *Signal Processing in C* (Wiley, New York, 1992), part II.
[22] D. C. Day, M. Y. Tsai, B. G. Streetman, and D. V. Lang, J. Appl. Phys. **50**, 5093 (1979).
[23] P. Kolev, M. J. Deen, and N. Alberding, in *Proceedings of the Symposium on Diagnostic Techniques for Semiconductor Materials and Devices*, edited by P. Rai-Choudhury, J. Benton, D. Schroder, and T. Shaffner (Electrochemical Society, Pennington, NJ, 1997), p. 377, PV 97-12.
[24] C. W. Wang, C. H. Wu, and J. L. Bone, J. Appl. Phys. **73**, 760 (1993).
[25] J. N. Fernandez and R. Ashley, *Assembly Language Programming for the 80386* (McGraw-Hill, New York, 1990), p. 182.
[26] J. P. Royer, *Handbook of Software and Hardware Interfacing for IBM PC* (Prentice-Hall, Englewood Cliffs, NJ, 1987), p. 206.
[27] R. H. Austin, K. W. Beeson, S. S. Chan, P. G. Debrunner, R. Downing, L. Eisenstein, H. Frauenfelder, and T. M. Norlund, Rev. Sci. Instrum. **58**, 445 (1976).
[28] P. Kolev, Solid-State Electron. **35**, 387 (1992).
[29] J. J. Shiau, A. L. Fahrenbruch, and R. H. Bube, Solid State Electron. **30**, 513 (1987).
[30] Free for download at http://www.ensc.sfu.ca/ensc/GradStudents/kolev/kolev.html or at http://www.GeoCities.com/SiliconValley/Pines/6839.
[31] H. Okushi and Y. Tokumaru, Jpn. J. Appl. Phys., Part 2 **19**, L335 (1980).
[32] H. Yoshida, H. Niu, and S. Kishino, J. Appl. Phys. **73**, 4457 (1993).
[33] G. Couturier, A. Thabti, and A. S. Barriere, Rev. Phys. Appl. **24**, 234 (1989).