Microprocessor-based digital correlator[a)]

John C. Thomas,[b)] Y. T. Lum, and David Kennett

Department of Physics, University of Waikato, Private Bag, Hamilton, New Zealand

(Received 24 February 1983; accepted for publication 7 June 1983)

We describe the design, construction, and operation of a low-cost, microprocessor-based digital correlator. The device has 128 channels, operates in either the single clipping or single scaling mode, and allows selection of the sample interval with 2-digit precision over the range 100 ns to 9.9 s. The device can be operated in the autocorrelate or cross-correlate mode and may easily be expanded to more than 128 correlation channels.

PACS numbers: 06.50.Dc

INTRODUCTION

The technique of photon correlation spectroscopy (PCS) of scattered laser light is well established and enjoys widespread application, particularly in the study of the solution properties of macromolecules and other particles in the size range 10–1000 nm.[1] Figure 1 is a schematic diagram of our laser light scattering system. This is a typical arrangement for PCS experiments. The light source is a Spectra-Physics 164-06 Ar^+ ion laser with single-frequency intracavity étalon. The output aperture of the laser is reduced to ensure operation in the TEM_{00} transverse mode, and output powers of up to a couple of hundred milliwatts at 488 nm are typically used. The laser beam is focused into the cylindrical scattering cell C by lens L_1 (50-cm focal length) and the light scattered through an angle θ is imaged onto the EMI9863B/100 photomultiplier tube by lens L_2 (10-cm focal length). Pinholes S_1 and S_2 (300-μm diam) define precisely the coherence properties of the scattered light and limit the solid angle over which the light is collected.[2] The photomultiplier output pulses are converted to TTL level (5 V, 35 ns FWHM) by the amplifier discriminator[3] and the photocount autocorrelation function is computed by the digital correlator. The correlator is interfaced to a laboratory computer which performs curve fitting on the autocorrelation data. The commercial autocorrelators used for data collection in these experiments cost upwards of several thousand dollars and this cost is often a major barrier to the acquisition of a PCS facility.

A number of workers[4–6] have presented design and construction details for digital correlators. Of these, the Chen device[4] is the most successful in that it has been constructed and used in a number of laboratories with satisfactory results. One of us (J.C.T.) has used the Chen device and recognized that it had a number of characteristics which made it less than ideal. For instance, the need to provide a multichannel analyzer for data accumulation greatly increases the true cost of the device and the relatively small (16 count) channel storage capacity of the device means that the input count rate has to be kept low to prevent the channels from overflowing between read cycles. With this in mind we set out to build a correlator which does not have these limitations but which preserves the attractive features of the Chen device, i.e., (i) it is relatively cheap and easy to construct, (ii) it has a large number of correlation channels so that multiple decay components can be detected, and (iii) it is fast enough to be useful for PCS studies of biological macromolecules.

The need to provide a large counter storage capacity (at least 20 bits) for each correlation channel increases the cost of the device and makes necessary a complex logic network so that the counter contents can be read and displayed in real time. We have simplified greatly this problem by utilizing a microprocessor and cheap random access memory (RAM) in our correlator.

Recently, single-board, stand-alone microprocessor evaluation and development kits have become available at a moderate cost. These kits consist of a microprocessor unit (MPU), a small amount of RAM, a read only memory (ROM)–resident monitor (for program operation and development), a parallel interface, a cassette interface, a hexadecimal keypad, seven-segment display, and space for additional RAM and EPROM (erasable programmable read only memory). These rudimentary systems have the potential to save an enormous amount of hardware development and they increase dramatically the flexibility and intelligence of a particular device. For the present application an 8-bit microprocessor is the ideal choice because then it is only necessary to provide 8 bits of intermediate storage for each correlator channel yet the 8-bit capacity is sufficient to ensure that a reasonable input count rate can be accepted without causing the counters to overflow. We have built our correlator around the Motorola MEK6802D5 microprocessor development kit. However, our correlator design does not rely on

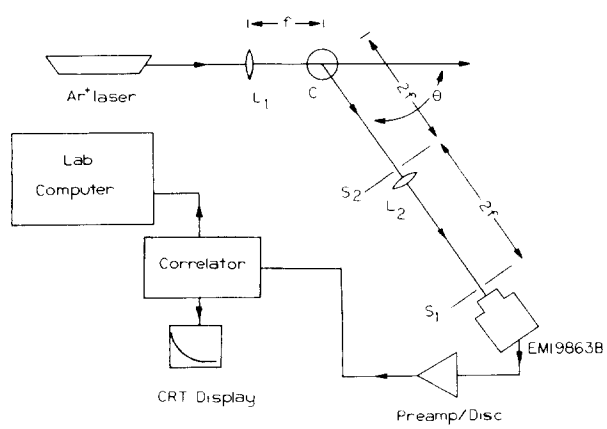

FIG. 1. Apparatus for photon correlation spectroscopy.

 0034-6748/83/101346-10$01.30

a specific microprocessor and, as only very basic control signals are required, a different microprocessor system could easily be used.

In the succeeding sections we discuss one-bit quantization schemes (clipping and scaling), the detailed design and construction of the correlator, and finally we demonstrate the performance of our device with experimental PCS data.

I. ONE-BIT QUANTIZATION SCHEMES

In a PCS experiment one ideally measures the true (or full) photocount (intensity) autocorrelation function

$$G^{(2)}(\tau) = \sum_{i=1}^{N} n(t_i)n(t_i + \tau), \tag{1}$$

where $n(t_i)$ is the number of photons detected in sample interval i centered at time t_i, N is the number of sample intervals (of width T) over which the correlation function is determined, and τ is the time delay.

If one of the elements in the correlation product can be reduced to a 1-bit quantization level, simplified circuitry and high-speed operation result because multiplication is replaced by an AND-gating operation.[7] Single clipping and single scaling are two easy-to-implement schemes which can achieve the required 1-bit quantization.

A. Single clipping

In single-clipped correlation a clipped count $n_k(t_i)$ is derived from the original photocount signal by the "clipping" operation

$$n_k(t_i) = 1 \quad \text{if } n(t_i) > k, \tag{2a}$$

$$n_k(t_i) = 0 \quad \text{if } n(t_i) \leqslant k, \tag{2b}$$

where k is a preset integer clipping level. We then measure the single-clipped correlation function

$$G_k^{(2)}(\tau) = \sum_{i=1}^{N} n(t_i)n_k(t_i + \tau). \tag{3}$$

When the scattered electric field exhibits Gaussian statistics (the majority of cases in laser light scattering), single clipping does not distort the measured correlation function.[8] Moreover, under appropriate experimental conditions single-clipped correlation will provide a measure of the actual intensity autocorrelation function with little loss in efficiency or accuracy.

B. Single scaling

In situations where the scattered light statistics are either unknown or known to be non-Gaussian, the use of single clipping may distort the correlation function. Jakeman *et al.*[9] have demonstrated that under particular conditions single-scaled correlation gives an excellent measure of the full photocount autocorrelation function regardless of the signal statistics. In this technique a 1-bit signal $n_s(t_i)$ is derived by scaling the incoming photocount signal by a factor s which is chosen high enough that the probability of obtaining more than one scaled count per sample interval is negligible compared with the probability of obtaining one. Thus we measure the single-scaled correlation function

$$G_s^{(2)}(\tau) = \sum_{i=1}^{N} n(t_i)n_s(t_i + \tau). \tag{4}$$

C. Clipped multichannel digital autocorrelator

Our correlator design utilizes the 1-bit cross-correlation approach and we can use either single clipping or single scaling to get the required 1-bit representation. Figure 2 shows schematically the correlator front end of a 16-channel element (one board) of our device. The shift register holds the 1-bit clipped (or scaled) signal which is simply a 1-bit record of the photocount signal for the previous 16 sample periods. The shift register contents gate the incoming photocounts into the counters of the corresponding correlator channels during the current sample interval. At the end of the sample period the new clipped count is clocked into the shift register and the existing values are all pushed one place to the right, (i.e., one more time delay) and the process repeats itself. Thus the products $n(0)n_k(T)$, $n(0)n_k(2T)$, $n(0)n_k(3T)$, ... are accu-

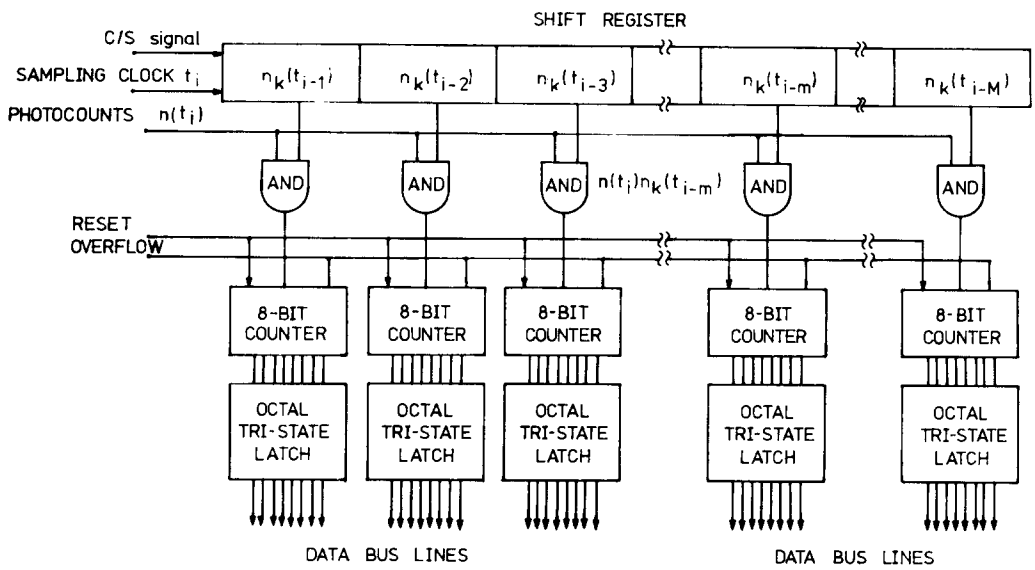

FIG. 2. Single-clipped digital correlator.

mulated in the channels 1, 2, 3, ... This gating arrangement is common to all single-clipped correlators (see, e.g., Chen *et al.*[4]). The essential and innovative difference with our design is that on each channel we have an 8-bit counter which acts as an intermediate store for the accumulating correlation product terms and each of the counters is connected to the microprocessor data bus via an addressable 8-bit tristate latch so that the counter contents can be read, accumulated in RAM, and the counter reset.

II. DIGITAL CORRELATOR HARDWARE

A block diagram of the correlator is given in Fig. 3. The device consists of the MEK6802D5 single-board microprocessor system (with RAM/EPROM expansion board), a control board, and (for 128 channels) eight correlation boards. Accumulation of the correlation data and overall control of the correlator are achieved via the MC6802 microprocessor under control of software which is briefly outlined below.

During normal operation the correlated photocount data are accumulated in the 8-bit counter of each correlation channel. The counters are read and cleared sequentially throughout the measurement period. This read–reset process is under the control of the microprocessor system clock (894.8 kHz) and is thus independent of and asynchronous with the correlation operation. The contents of each channel's counter are accumulated in RAM and, with 3 bytes of storage per channel, the count capacity is 2^{24} (16×10^6) per channel. At the end of each read cycle (every channel is read once) an updated correlation function is displayed on the oscilloscope screen. When a measurement is terminated the accumulated correlation data may be transferred to a laboratory computer or other device via the MC6821 PIA (peripheral interface adapter) of the MEK6802D5. Fully buffered versions of the correlator sample clock (SCLK), the photocount signal (SIG), the clipped/scaled signal (C/S), and the run/halt signal (GATE) are provided on the correlator front panel for use with external counters if it is desired to normalize the correlation data prior to curve fitting.

A. Correlator system memory map

The required address decoding and additional EPROM and RAM locations necessary for the correlator operating system are provided on a RAM/EPROM pc board which is mounted onto the MEK6802D5 pc board in piggyback fashion. It can be seen from the system memory map shown in Fig. 4 that this board provides 16K of EPROM (8×2516 devices) and 12K of RAM (24×2114LP devices). This memory, when added to the 2K EPROM and 5K RAM (1K optional user plus 4K on the wire-wrap test area) already on the MEK6802D5 board, grossly exceeds the minimum requirements (6K RAM and 6K EPROM is ample) but is relatively cheap to implement and will accommodate future expansion of the device. As shown in Fig. 4 the actual correlator is located at addresses \$0000–\$007F (correlation channels 1–128) and \$0080–\$00FF (control board at location \$0080).

B. Control board

The control board is designed to generate a 1-bit signal by performing either clipping or scaling on the input photocount signal, to generate timing signals and sample interval timing from a 20-MHz system clock, to provide readouts and display of the system status, and to provide an analog output for real-time oscilloscope display of the correlation data.

1. Correlator sample clock

The basic clock for the correlator is a 74LS320 crystal-controlled oscillator/clock driver running at 20 MHz. The output of this circuit is programmably divided to give a sample interval in the range 100 ns–9.9 s. This sample interval is selected with two-digit resolution (e.g., 1.3 μs).

Programmable binary counters scale down the basic clock by a preset factor to set the mantissa of the sample interval (1–99). A series of six decade counters then scales the

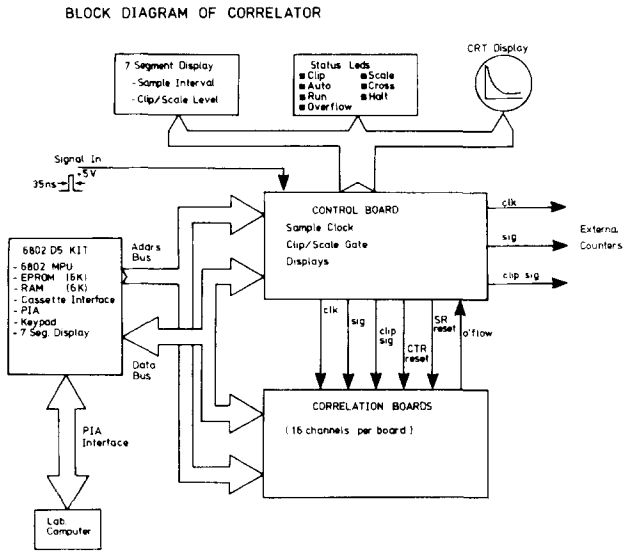

FIG. 3. Block diagram of the microprocessor-based correlator.

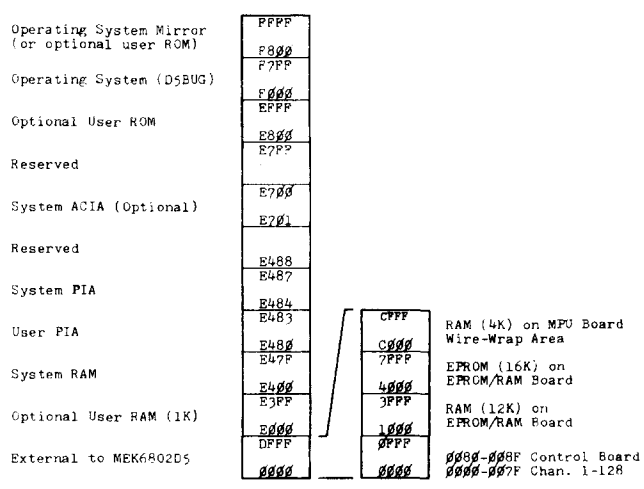

FIG. 4. Memory map modification for the digital correlator.

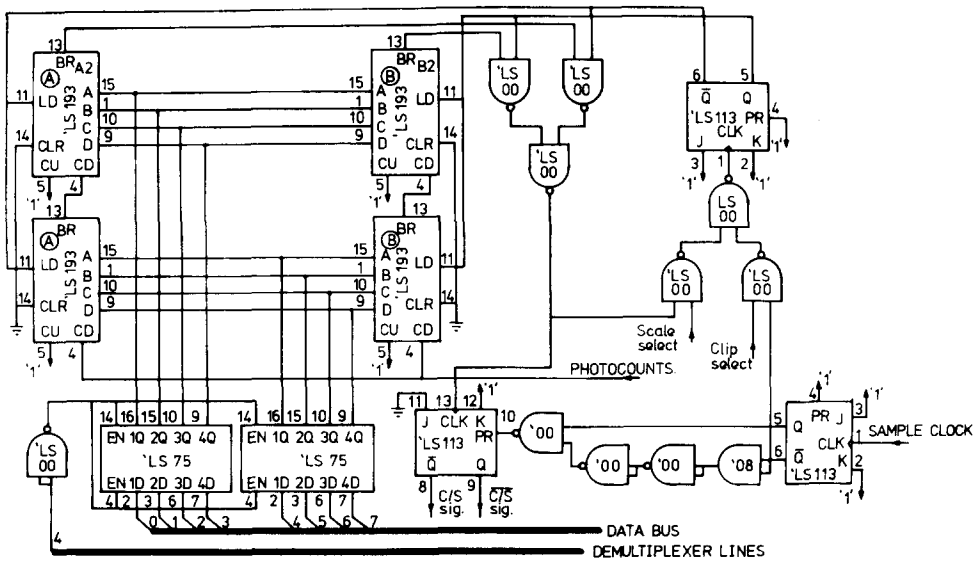

FIG. 5. Clip/scale circuit.

clock by the chosen submultiple of ten (0–6). We presently limit the minimum clock period to 100 ns, but if necessary this can be reduced to 50 ns. The sample clock is fed to the clip/scale circuitry on the control board and is buffered through a 74S140 line driver to the shift register clock inputs on each of the correlation boards and to a front-panel BNC socket.

2. Clip/scale circuitry

The 1-bit C/S signal is derived from the input photocount signal via the clip/scale circuitry shown in Fig. 5. This scheme is similar to that used by Chen *et al.*[4] in that to minimize dead time we have two banks of counters which count in alternate counting periods, but in the present case we have implemented the single scaling function as well as single clipping.

A preset count (1–99) is loaded from two 74LS75 latches into one bank of counters while the other bank is counting. The counters simply count down the incoming photocount pulses from the preset level until an underflow occurs and this generates a BR output. This BR output (after passing through a gating scheme to eliminate spurious borrows generated when the other counter bank is being reset) is stored in a J–K flip-flop and becomes the C/S count. At the end of the current sample interval this C/S count is clocked into the 74LS164 shift register on the first correlation board and the flip-flop is preset and ready to store the C/S count from the succeeding sample interval.

Switching between the two counter banks is achieved by having the load inputs (LD) of the two pairs of counters connected to complementary outputs of a toggle flip-flop. Thus while the LD input of one pair of counters is high and the counters are counting, the LD input of the other pair is low, counting is inhibited, and the counters are loaded with the appropriate preset count. In the single clip mode the sample clock switches between the two pairs of counters (by triggering the toggle flip-flop) while in the single scale mode the BR output of the counters does the switching. Selection of either clipping or scaling is done via the microprocessor. The clip/scale circuitry (and the clocking of the shift registers on the correlation boards) is synchronized to the rising edge of the sample clock.

3. Digital-to-analog interface

Two 8-bit digital-to-analog (D/A) converters (DAC0808) provide a real-time display of the correlation function during an experiment. A maximum-ratio/self-scaled display is achieved by examining the correlation data and displaying the most significant data-containing 8 bits.

The analog display circuitry is shown schematically in Fig. 6. For the X axis two 74LS75s are used as an 8-bit latch to hold the channel number and the D/A converter simply converts this to an analog voltage. The Y axis (correlation data) circuitry is made more complicated by the need to provide a scaled display. In this case we use four 74LS75s as a 16-bit latch to hold the topmost 16 bits of the 24 bits of correlation data, then a series of eight 74LS151 8-of-1 data multiplexers 'windows' out of the most significant 8 bits for input to the D/A converter. This multiplexing scheme is controlled by a further 74LS75 latch which holds a 3-bit

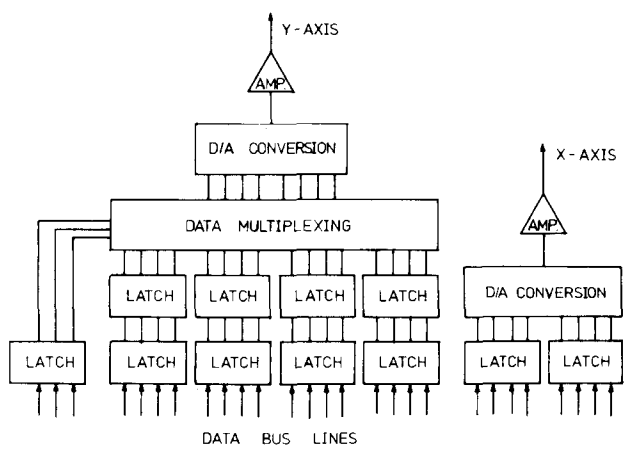

FIG. 6. D/A interface schematic.

display code determined by the highest bit set in the data of the first correlation channel. The 3-bit code is determined by examining the correlation data in channel No. 1 at the beginning of each display period and this code is used to display the subsequent 127 channels of data, i.e., the same 8 bits of data are output for all the correlation channels and the data are scaled by the amount required for channel No. 1 to fit on the screen.

The analog voltage output of the D/A converters is amplified by an inverting amplifier to give an output in the range 0 to 2.3 V. A TTL level signal which can be used for blanking the oscilloscope screen during flyback and the X and Y axis signals are available through BNC connectors on the rear of the correlator.

4. Status indicators, seven-segment displays, and overflow indicator

The operating status and experimental parameters (sample interval, clip/scale level) are displayed on the correlator front panel on a series of high-intensity light-emitting diodes (LED's) and seven-segment displays. The correlator operating status (CLIP or SCALE, RUN or HALT, AUTO or CROSS) is microprocessor controlled and the control data bits for these functions are stored in data latches. The outputs of the latches control the appropriate circuitry and buffered versions of these outputs provide drive for the appropriate status LED's. An overflow LED warns when overflow occurs in any of the binary counters on the correlation boards. When overflow is detected either the input count rate must be decreased or the clip/scale level must be increased.

C. Correlation boards

A schematic diagram of the correlation boards is shown in Fig. 7. Each correlation board consists of 16 channels, each with a fast 8-bit counter and control signals to transfer data from the counter to the MC6802 for accumulation in RAM. The modular construction and bus configuration of the correlator mean that additional channels may be easily added.

Two 74LS164 8-bit serial-in, parallel-out shift registers on each board provide 16 channels of storage for the C/S signal. The shift registers are cleared under software control at the beginning of each fresh data accumulation period. A 74LS154 4-16 demultiplexer with the lower four address bits A_0–A_3 and a strobe or card select (STR) as inputs allows each of the 16 channels on a board to be addressed uniquely so that the channels can be read and reset in turn by the microprocessor. The $\overline{E\Phi}$ clock synchronizes data transfer to the microprocessor clock.

1. Correlation and data transfer

The new C/S count from the control board is clocked into the first shift register (on the first correlation board) on the rising edge of the sample clock and the output of the second shift register on the board is fed into the first shift register on the next board and so on for the six subsequent boards. Thus the values of the C/S signal for the 128 previous sample intervals which are stored in the 128-bit shift register are all clocked one place to the right. The parallel outputs of the shift registers gate the incoming photocounts

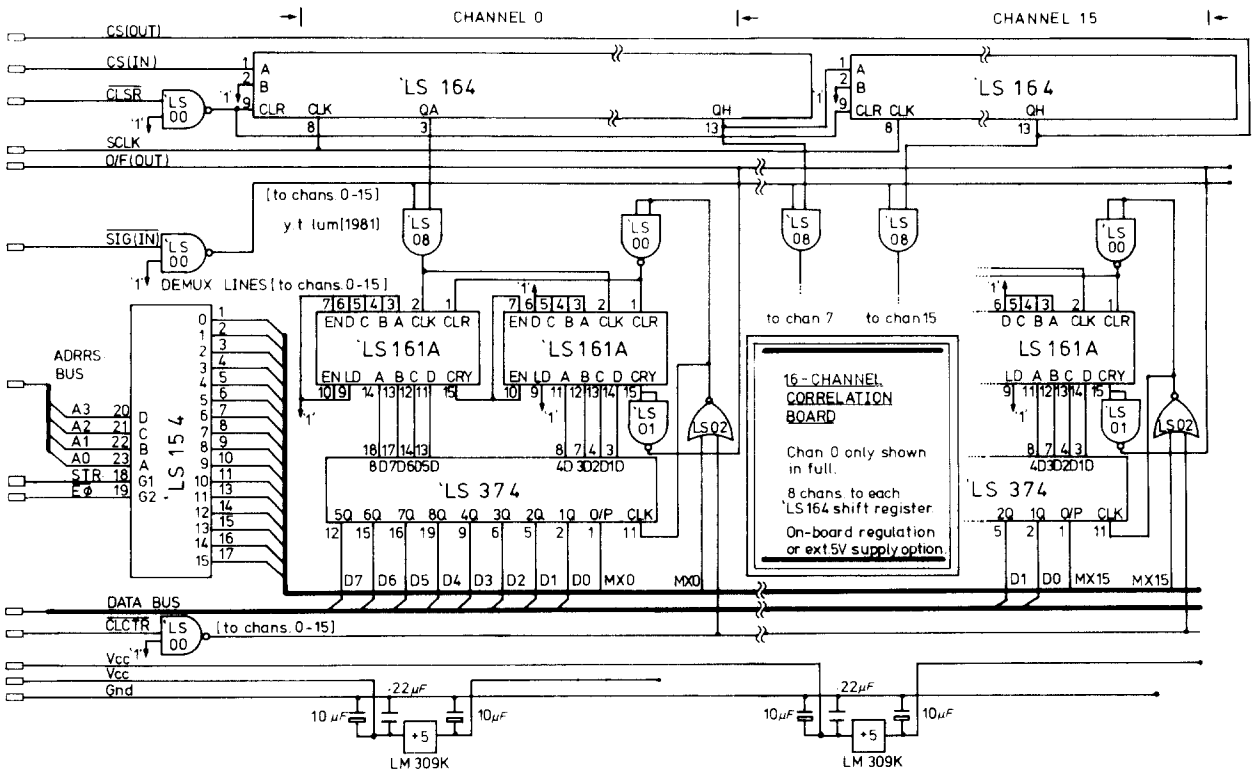

FIG. 7. Correlation board schematic.

into an 8-bit counter on each correlation channel. Thus the 1-bit by n-bit correlation product is accumulated for each correlation channel. The 8-bit counters consist of two 74LS161A 4-bit synchronous binary counters cascaded to perform 8-bit synchronous counting.

When a particular channel is addressed the following sequence of events occurs: (i) the outputs of the 74LS374 tristate latch are enabled onto the data bus; (ii) the contents of the 8-bit counter are loaded into the latch; (iii) the 8-bit counter is reset and begins a fresh count period; (iv) the microprocessor gets the count off the data bus and adds this to the contents of the RAM location corresponding to the correlator channel being read.

The whole process is synchronized to the VMA and $E\Phi$ control signals of the MC6802 and is quite straightforward, although in practice there are two time delays (one associated with the tristate latch and one associated with the counter resetting) that need to be considered.

A minimum data setup time of 20 ns must be satisfied for the 74LS374 latch. If the data at the latch input change during this time, the data at the output will be uncertain. It was not possible to measure directly the manifestation of this effect in the correlation data. However, the net result will be to add noise to the data. It is important to realize that this process will not systematically distort the correlation data. In any event, since this 20-ns setup time applies only when a channel is read (i.e., once every 7.8 ms), the fraction of the total counts arriving in this time is of the order 20 ns/7.8 ms or 2.6×10^{-6}. This is clearly a very small fraction and, given that the 20-ns setup time is actually narrower than the pulse width of the incoming photopulses (35 ns), the setup time is expected to be violated extremely infrequently.

When each channel is read by the microprocessor, a dead time of 28 ns occurs while the counters are reset. Following the same argument used above we conclude that in practice this dead time has an insignificant effect on the correlation data because it also only occurs when a channel is read. This means that the fraction of missed counts due to the dead time is at most 28 ns/7.8 ms or 3.6×10^{-6}, which is clearly not significant. Furthermore, even if this was larger it would not systematically distort the correlation data as each channel experiences the same dead time and will thereby miss the same fraction of the total counts for that channel.

III. SOFTWARE SYSTEM

One of the advantages of a microprocessor-based system is that the correlator can be operated entirely via the front-panel keypad. For ease of operation our software allows data to be entered via the (hex) keypad as though the keypad was coded in decimal, i.e., in this case an entry of "10" is interpreted as 10 (decimal) and not 16 (decimal). We have written user functions to carry out clip/scale level select, sample interval select, single clipping select, single scaling select, autocorrelate select, cross-correlate select, and data output select. The software is relatively device specific so we will only outline the operating program and discuss the software considerations that need to be made for efficient operation of the correlator. A complete software listing is available on request.

A. Overall program description

Figure 8 is a flow chart of the correlator software (stored in EPROM) operation. At the beginning of the program the user function pointer is initialized to the user function table and the software flags and 384 bytes of RAM (to store the accumulating correlation data) are cleared for each

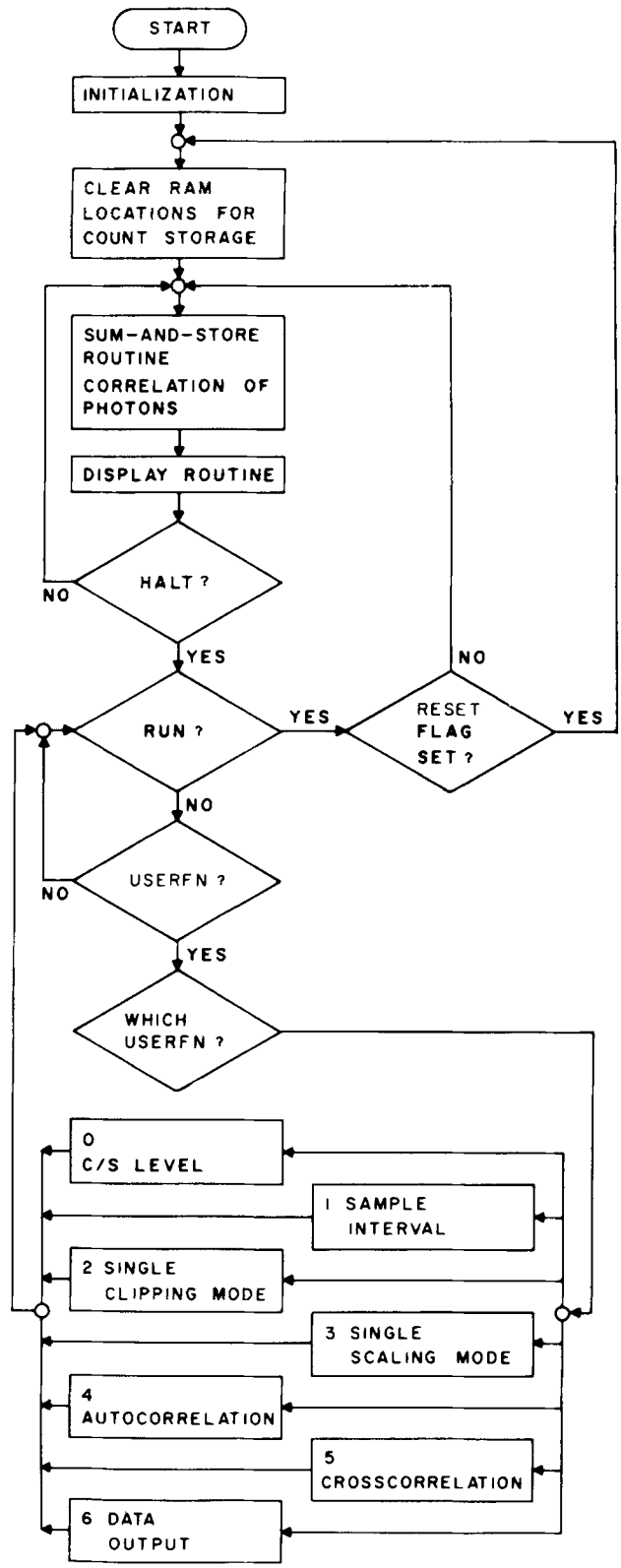

FIG. 8. Flow chart for overall program operation.

experiment. The program continually loops through the sum-and-store and display routines collecting and displaying data until it is halted through the keypad. After halting the correlator, the user may either invoke a user function to alter the correlator operation (by entering an appropriate hexadecimal digit) or resume data collection.

1. Sum-and-store routine

To streamline the sum-and-store routine we have written it in in-line code for the whole 128 channels. Furthermore the actual correlator hardware is located in the lowest memory addresses so that it can be addressed via direct addressing. The assembly code to read, sum, and store one correlation channel is

```
STRT  LDA   A D $00     3 Read channel
      ADD   A E $CC00   4 Add to
      STA   A E $CC00   5 low byte.
      BCC       .+9     4 Next channel.
      LDX     E $CC01   5 Add
      INX               4 the
      STX     E $CC02   6 carry.
      LDA   A D $01     3
```

This code is simply repeated 128 times incrementing the correlator channel address and the RAM destination addresses each time. For the case depicted here data are loaded from the binary counter of channel No. 1 (address $00) into the A register and then added to the low byte of RAM assigned to that channel (address $CC00). The result is stored back in the low-byte RAM location and the carry flag is checked. If the carry flag is set, the contents of the high- and middle-byte RAM locations ($CC01, $CC02) are loaded into the index register (a 16-bit register) which is then incremented and stored back in RAM. If the carry flag is not set, a branch to the next channel occurs and so on throughout the 128 correlation channels. In the worst-case situation, when there is a carry from the low byte every time, this routine takes 31 machine cycles per channel to execute.

2. Display routine

The assembly routine which implements the correlation data display is also written in-line code and is listed in Table I.

This routine starts out by examining the high byte of channel No. 1 to determine the required scaling, (i.e., which eight of the 16 topmost data bits will be selected for display by the multiplexing circuitry). The high byte is first compared with 1 and if it is $\leqslant 1$, the high seven bits of the middle byte and the first bit of the high byte are the eight bits displayed. If the high byte is > 1, the contents of the A register are shifted left (ASL A) until the carry bit is set. The B register is loaded with 8 and each time an ASL A is performed it is decremented; thus we identify the highest data-containing bit of the high byte. The contents of the B register are stored in the data select latch (SELAT) on the control board and this latch then sets the 8-bit display window.

The data for channel No. 1 are displayed by loading the high and middle bytes of the correlation data into the 16-bit

TABLE I. Data display routine.

```
*Display Data
      CLR B                          2
      LDA A E  $CC01                 4 Get Hi byte
      CMP A #  $1                    2 >1?
      BLS      DISPY                 4
      LDA B #  $8                    2
BITST ASL A                          2 Roll data to
      DEC B                          2 find 1st
      BCC      BITST                 4 non-zero bit
DISPY STA B D  SELAT                 4 Stow in sel.  latch
      LDA A E  SIGNL                 4 Disable
      ADD A #  6                     2
      STA A D  RHAC                  4 blanking
*
      LDA A E  $CC01                 4 Get Hi byte,
      STA A D  $8A                   4 stow in latch.
      LDA A E  $CC02                 4 Get Mid byte,
      STA A D  $8B                   4 stow in latch.
      LDA B #  $0                    2 Channel # x2,
      STA B D  $8C                   4 stow in latch.
      LDA A E  $CC04                 4 Get Hi byte
          .
      (127 channels)
          .
      LDA B E  SIGNL                 4
      ADD B #  $4                    2
      STA B D  RHAC                  4 Blank display
      JMP    E INTRP                 3 Check Keys.
          .
INTRP TST    E KEYFG                 6 Key pending ?
      BNE      OUT                   4 Yes
BACK  JMP    E STRT                  3 Go get more data
OUT   JSR    E RDKEY                 9 Read Key.
```

Y-axis data latch on the control board (addresses $8A, $8B) and loading the channel count into the 8-bit X-axis data latch (address $8C). This process is repeated for the subsequent 127 channels.

In the worst case the display sequence takes 2923 machine cycles and since the sum-and-store sequence requires $128 \times 31 = 3968$ machine cycles for the worst case, the display is refreshed at least once every 6891 machine cycles or once every 7.8 ms. This means that the display is refreshed at a rate of at least 130 Hz, which is well above the refresh rate at which display flicker would be obvious.

B. User functions

The user functions control the correlator operating mode simply by setting or clearing particular bits in the status latches on the control board. The only functions needing additional input information are the routine which sets the clip/scale level and the routine which sets the sample interval.

Selection of the clip/scale level is done by keying in a particular value in the range 0–99. To select the sample interval (in μs) three digits are entered through the keypad. The first two digits are the mantissa and the third is the exponent of the sample interval. For example, an interval of 1.2 ms is entered as "123," i.e., 1.2×10^3 μs. When the data for the

clip/scale level and the sample interval are entered, they are checked for validity and if they are outside the allowed ranges default values are set.

IV. DEVICE EVALUATION

A. Maximum input count rate

The data throughput rate of the correlator is determined by the speed of the read, sum, store, and display process and on the number of channels implemented. We require a large number of correlation channels so that multiple decay components can be detected during data analysis, but the throughput rate of the device will decrease in direct proportion to the number of channels (because it takes longer to read more channels). A trade-off between the number of channels implemented and the maximum input count rate must be made. We regard 128 correlation channels as a reasonable working number and now proceed to calculate the maximum input count rate for a 128-channel device operating in the single-clipped mode.

The 8-bit counters on the correlator channels must not overflow, i.e., we require $\leqslant 255$ coincident counts per correlator cycle time T_{cyc} (the time to read, sum, store, and display 128 channels). The maximum mean number of coincident counts per sample interval T is[4]

$$2\langle n\rangle\langle n_k\rangle = 2\langle n\rangle\left(\frac{\langle n\rangle}{1+\langle n\rangle}\right)^{1+k}. \tag{5}$$

The maximum number of coincident counts per cycle time is thus

$$2\langle n\rangle\langle n_k\rangle = 255T/T_{cyc}, \tag{6}$$

and the maximum mean count per sample interval for each value of k is found by solving

$$2\langle n\rangle\left(\frac{\langle n\rangle}{1+\langle n\rangle}\right)^{1+k} - \frac{255T}{T_{cyc}} = 0. \tag{7}$$

The maximum input count rate C_{max} is then

$$C_{max} = \langle n\rangle/T. \tag{8}$$

We showed above that for our device in the worst case $T_{cyc} = 7.8$ ms. Using this value of T_{cyc} we have calculated the maximum input count rate as a function of clip level and sample interval. These results are summarized on the graph shown in Fig. 9. From the graph we see that the maximum allowable input count rate decreases from 108×10^6 cps at 100-ns sample interval and a clip level of 99 to an asymptotic value of 16×10^3 cps at longer sample intervals and lower clip levels. This asymptotic value is the true throughput rate of the device in that it represents the number of coincident counts actually processed by the device. In practice the count rates that prevail in PCS experiments are usually much less than two or three hundred thousand so that the fundamental 16×10^3 cps does not really limit the utility of our device.

B. Correlation functions

During the construction of this device numerous tests were made to check that the correlator channels were counting correctly and that the contents of the counters in each channel were being faithfully read and accumulated in RAM. The simplest and most reliable way to confirm the device's overall operation is with a real dynamic light scattering experiment.

Using the experimental arrangement shown in Fig. 1, we have performed a series of PCS experiments on polystyrene spheres. In this case the unnormalized correlation function has the form

$$G_k^{(2)}(\tau) = A + B\exp(-2\Gamma\tau), \tag{9}$$

where A and B are constants. The exponential decay constant Γ is related to the translational diffusion coefficient D of the scatterers by

$$\Gamma = K^2D. \tag{10}$$

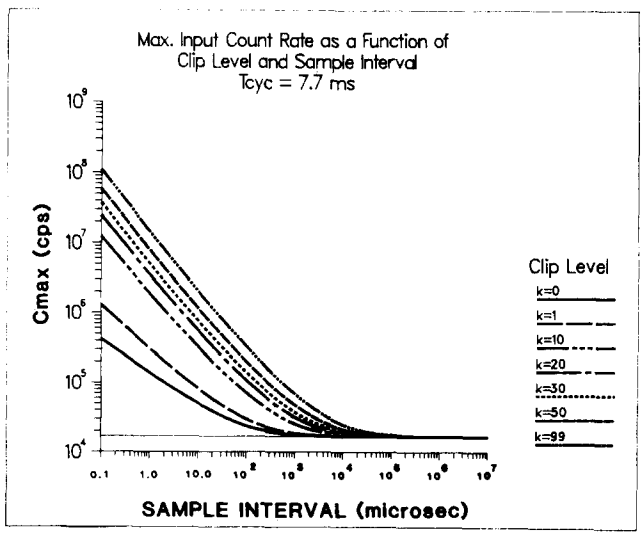

FIG. 9. Maximum input count rate as a function of clip level and sample interval.

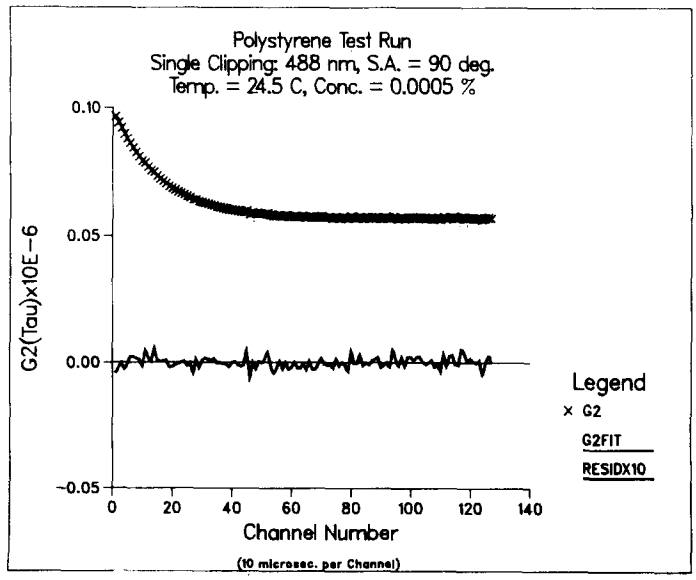

FIG. 10. Single-clipped correlation function and best-fit single exponential plus base line for homodyne detection. Sample interval = 10 μs. Data collection time = 10 s.

TABLE II. PCS measurements on 91-nm polystyrene spheres.

Clipping/ scaling	Sample int. (μs)	Temp. (°C)	Γ (s^{-1})	D (10^{-12} m^2 s^{-1})	Diameter (nm)
Clipping at 0	10	24.5	3149	5.337	90.6 ± 1.0
Clipping at 0	5	24.5	3240	5.491	88.2 ± 4.0
Scaling at 8	10	22.8	2968	5.030	90.6 ± 0.5
Scaling at 5	5	24.5	3152	5.342	91.9 ± 0.6
Clipping at 0 (cross-correlation)	10	22.8	2898	4.912	94.2 ± 2.0

Here

$$K = \frac{4\pi n \sin \theta/2}{\lambda_0} \tag{11}$$

is the magnitude of the scattering vector, θ is the scattering angle, n is the solvent refractive index, and λ_0 is the vacuum wavelength of the incident light. We determine the radius R of the scatterers from the Stokes–Einstein relationship and D, i.e.,

$$D = k_B T/6\pi\eta R. \tag{12}$$

Here k_B is the Boltzmann constant, T is the absolute temperature, and η is the solvent viscosity.

Samples of a 0.005-wt. % solution of polystyrene spheres in water were filtered through 0.45-μm pore-size filters into a clean scattering cell and a number of measurements were made using both single clipping and single scaling. The cross-correlation function was also tested by applying the photocount signal to both the A and B inputs of the correlator. This is expected to give the same result as for autocorrelation with the photocount signal applied to input A only. The incident laser power was about 100 mW at 488 nm and the photocount rate around 30×10^3 cps at a scattering angle of 90°. The experimental correlation data for a series of 10-s runs were recorded and subsequently fitted to a single exponential plus a base line using a locally written nonlinear regression routine to determine the decay constant Γ. The best-fit value of Γ was then used to calculate the diffusion coefficient and the radius of the scatterers using the results outlined above.

Figure 10 shows a typical result of curve fitting to the experimental data. Clearly the residuals between the raw data and the fitted curve are evenly scattered about zero suggesting a good fit to the experimental data. Furthermore the reduced chi-square obtained from the curve fitting was invariably close to 1.0, which is expected if the experimental data are well described by the fitted functional form.[10] The results of these experiments are summarized in Table II. Each result is the mean of five experimental runs. In each case the value for the diameter calculated from the experimental data agrees closely with the nominal value of 91 nm. On the basis of these experiments it is clear that this correlator operates correctly and provides a faithful measurement of the autocorrelation function using either single clipping or single scaling.

V. DISCUSSION AND CONCLUSION

As we mentioned at the outset, a number of other workers have developed low-cost digital correlators, but these all have characteristics which, to varying extents, limit their desirability for PCS experiments. For example, the device of Bohidar *et al.*[5] is very low-cost, but has only 64 correlation channels, does not have provision for real-time display of the correlation data, and the correlation data must be read manually, one channel at a time. This mode of operation is not really compatible with the high data production rate common in PCS experiments. Basano and Ottonello's device[6] has a large number of correlation channels (256), but it is not capable of parallel processing since it works on the principle of measuring the photoelectron arrival time distribution and this is a single-channel process. Thus to get a good signal-to-noise ratio run times will be long. The Chen device[4] suffers principally from the need to provide an expensive multichannel analyzer to accumulate and display the autocorrelation data. The 16-count channel counter capacity also limits the input count rate for this device. Recently, Paul and Nicoli[11] have built a modified Chen correlator wherein they increased the channel capacity to 128 counts and used a microprocessor and RAM to store the correlation data. This device has only 48 correlation channels and the measured correlation function is scaled by 128, which is undesirable at low count rates.

In the foregoing sections we have described the design, construction, and operation of a device which has none of the disadvantages mentioned above and which preserves the more attractive features of the Chen device. Moreover, with the choice of either clipping or scaling our device can handle both Gaussian and non-Gaussian signal statistics. The cross-correlation facility also makes our device suitable for applications such as multidetector experiments. Incorporation of the microprocessor in our correlator design has made the device extremely flexible and easy to use. Simply by changing the software, the correlator can be made to perform more complex modes of operation (e.g., instead of taking one long measurement, take many measurements of short duration). With appropriate interfacing the correlator could also record (and control) experimental parameters such as the sample temperature, scattering angle, etc.

Our correlator is a research quality device that is ideally suited for use in PCS experiments; however, we believe that the low cost of our device makes it especially attractive for applications in areas where cost is a major consideration,

e.g., in undergraduate teaching laboratories. Finally our correlator design clearly illustrates the advantages of using single-board microprocessor kits as the basis of an instrument design. These simple, low-cost devices are cost effective and have tremendous potential for reducing both the design effort and the hardware requirements in an intelligent instrument. The end result in our case is that for around $1500 we have built a device which compares well with commercial devices costing on the order of $10 000.

ACKNOWLEDGMENTS

The authors wish to thank Dr. R. A. Sherlock for helpful discussions throughout this work, A. C. Galbreath for the use of his 6800 cross assembler and advice on programming the 6802, and Kerrie Thomas for assistance with preparation of the manuscript. This work was supported in part by UGC Grant No. 80/111 to J. C. T.

[a] Preliminary accounts of this work have been given at the VIIth International Biophysics Congress, Mexico City, September 1981, and at QELSS1, a Conference on Quasi-Elastic Light Scattering Spectra and Their Interpretations, University of Michigan, 23–25 June 1982.

[b] Current address and address for reprints: Department of Chemistry, University of Washington, Seattle, Washington 98195.

[1] J. M. Schurr, CRC Crit. Rev. Biochem. **4**, 371 (1977).
[2] P. N. Pusey and J. M. Vaughan, in *Dielectric and Related Processes*, edited by M. Davies (The Chemical Society, London, 1975), Vol. 2, pp. 48–105.
[3] H. Guttinger, M. Gautschi, and E. Serrallach, J. Phys. E **9**, 936 (1976).
[4] S. H. Chen, W. B. Veldkamp, and C. C. Lai, Rev. Sci. Instrum. **45**, 1356 (1975).
[5] H. Bohidar, S. Kulkarni, S. Prasad, and S. Chopra, J. Phys. E **13**, 614 (1980).
[6] L. Basano and P. Ottonello, J. Phys. E **14**, 1257 (1981).
[7] E. Jakeman and E. R. Pike, J. Phys. A **2**, 411 (1969).
[8] E. Jakeman, E. R. Pike, and S. Swain, J. Phys. A **4**, 517 (1971).
[9] E. Jakeman, C. J. Oliver, E. R. Pike, and P. N. Pusey, J. Phys. A **5**, L93 (1972).
[10] P. R. Bevington, *Data Reduction and Error Analysis for the Physical Sciences* (McGraw-Hill, New York, 1969).
[11] M. Paul and D. F. Nicoli, Rev. Sci. Instrum. **51**, 1253 (1980).