Digital Pulse Processor Using A Moving Average Technique

Valentin Jordanov and Glenn F. Knoll
The University of Michigan, Dept. of Nuclear Engineering, Ann Arbor, MI. 48109

Abstract

A digital pulse processor with improved differential linearity and reduced dead time has been designed. The circuit uses an 8-bit flash ADC running at 36 MHz and continually sampling the signal from the preamplifier or shaping amplifier. The digitized signal is then processed by a digital moving averager. A digital peak detector is used for measuring the amplitude of the shaped pulses. A novel, threshold-free circuit has been designed that combines both the moving average and peak detection functions. The circuit also provides a timing signal with an uncertainty of one sampling period. The number of the averaged samples (equivalent to the shaping time constant) is digitally controlled.

I. INTRODUCTION

Interest has recently grown in substituting purely digital methods for the analog techniques traditionally used in processing pulses from radiation detectors. Many of these efforts are intended to improve the pulse throughput rate while maintaining good energy resolution. Some are based on the analysis of digitized wave forms employing an off-line computer [1], and others use specialized on-line processors with relatively long processing times on the order of tenths of microseconds [2]. Other designs have been described [3,4] that are based on real-time processing on a faster time scale. However, these examples require the provision of a second, threshold sensitive, channel to generate the necessary control signal. As a result, these systems are more complex and subject to instabilities at low threshold levels [3]. This paper describes a simplified approach to real-time digital processing that is based on a combination of digital filtering and peak detection.

II. PULSE PROCESSING TECHNIQUE

One of the widely used techniques for pulse filtering at high counting rates is the gated integrator (GI) [5]. An alternative approach to obtain the area under the detector pulse is to use a transversal filter with rectangular weighting function [6,7]. We will call this particular case of transversal filter a moving integrator (MI). The principle of this filter is illustrated in Fig.1. The filter output signal is obtained by integrating the input signal over a fixed interval of time. The upper limit of integration is equal to the time moment at which the output signal is evaluated.

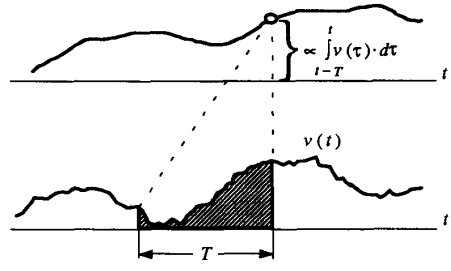

Fig.1 Principle of the moving integrator

In a common spectroscopy system, where a threshold controlled ADC is employed to digitize the filtered signal, the gated integrator method offers higher throughput rate relative to the transversal filter approach. This comparison is illustrated in Fig.2 which shows a case of two close but not overlapping input pulses *1* and *2* processed by GI (a) and MI (b). The areas under the pulses are *A1* and *A2* respectively.

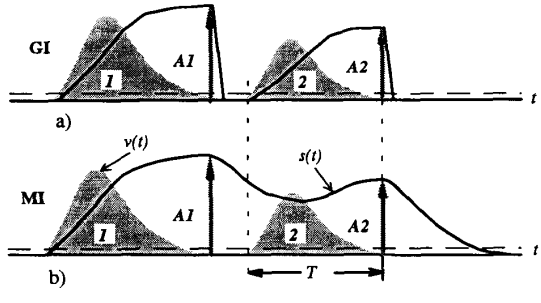

Fig.2 Comparison between a) gated integrator and b) moving integrator.

In this example both MI and GI filters are characterized by an integration time *T*, equal to the width of the input pulse. Under these circumstances the MI shaped pulse is twice as long as the unshaped pulse, while the increase in GI dead time is small owing to the finite integrator reset time. As a result the MI filtered pulses overlap. However, the maximum amplitudes *A1* and *A2* (Fig.2b) still will correctly represent the area under the pulses *1* and *2*. If the shaped signal shown in Fig.2b is processed by a traditional, threshold based, ADC the second peak (*A2*) will be lost,

although it is a correct measure of the area of pulse 2. To eliminate this shortcoming we propose to use a threshold free measuring system based on peak detection and utilizing a moving average technique.

A. Pulse Integration

The moving integrator output $s(t)$ can be expressed as a convolution of rectangular function and the input signal $v(t)$ [5]. The convolution integral has the form

$$s(t)=\int_{t-T}^{t} v(\tau)\cdot d\tau \tag{1}$$

where T is the width of the rectangular function.
Equation (1) can be rewritten as a difference of two integrals

$$s(t)=\int_{0}^{t} v(\tau)\cdot d\tau-\int_{0}^{t-T} v(\tau)\cdot d\tau\ . \tag{2}$$

When the following condition is met

$$v(t)=0 \quad \text{for} \quad -T\le t<0 \tag{3}$$

then equation (2) can be expressed as

$$s(t)=\int_{0}^{t} v(\tau)\cdot d\tau-\int_{-T}^{t-T} v(\tau)\cdot d\tau\ . \tag{4}$$

After changing the variable in the second integral of equation (4), equation (1) is reduced to

$$s(t)=\int_{0}^{t}[v(\tau)-v(\tau-T)]\cdot d\tau \tag{5}$$

Clearly, if $v(t)$ is a single pulse with duration equal or less than the integration time T then the maximum value of $s(t)$ represents the total area under the pulse. If T is shorter than the pulse width, then the peak value in $s(t)$ corresponds to the maximum area under the pulse enclosed in a time interval of width T.

The function $s(t)$, described by equation (5) has an extremum if $ds(t)/dt=0$ or $v(t)-v(t-T)=0$. Therefore, $s(t)$ reaches its maximum value (peak) when the signal $v(t)-v(t-T)$ crosses zero and changes its sign from positive to negative. Similarly, $s(t)$ has a minimum (valley) when $v(t)-v(t-T)$ crosses zero in the opposite direction.

B. Moving Average

In the discrete-time case the equation (5) can be approximated by a sum of equally spaced samples of the signal $v(t)$. The filter output corresponding to the n-th sample can be written as

$$s[n]=\sum_{i=0}^{n}\{v[i]-v[i-k]\} \tag{6}$$

where the numbers in the brackets denote the number of the sample and the constant k is the number of the samples corresponding to the integration time window. In other words, if t_c is the sampling time then the integration time is $T=t_c\cdot k$. Equation (6) has been normalized to t_c thus providing the same measuring units for $s[n]$ and $v[n]$.

The sum $s[n]$ reaches its maximum value at n_p if

$$\left.\begin{matrix} s[n_p]-s[n_p-k]\ge 0 \\ s[n_p+1]-s[n_p+1-k]<0 \end{matrix}\right\} \tag{7}$$

Similarly, the condition for minimum of $s[n]$ at n_v is given by

$$\left.\begin{matrix} s[n_v]-s[n_v-k]\le 0 \\ s[n_v+1]-s[n_v+1-k]>0 \end{matrix}\right\} \tag{8}$$

Hence, in order to apply the MI technique and to detect the peaks in the shaped signal, a difference between prompt and delayed input signal should be carried out. This difference is accumulated continually. In the common case of digital signal processing, the averaging is performed over the number of samples which have been accumulated in the time window. In our particular case the accumulator sum is normalized to the range of the ADC.

C. Pile-up Effects

The pile-up of two or more pulses results in an erroneous estimation of the measured pulse height and loss of some of the events.

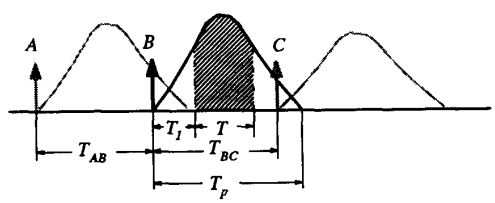

Fig.3 Pile-up of input pulses.

In our processor the pulse amplitude corresponding to event B in Fig.3 will be correctly measured if the time interval between events A and B satisfies the condition

$$T_{AB} > T_p - T_1 \tag{9}$$

and the time interval between B and C is

$$T_{BC} > T_1 + T, \tag{10}$$

where T_p is the input pulse duration and T is the integration time of the pulse processor.

Let P_{AB} and P_{BC} be the probabilities that (9) and (10) are satisfied. Since the events A, B and C appear independent in time, the probability that the amplitude of the shaped pulse B is free of pile-up is given by

$$P_{pf} = P_{AB} \cdot P_{BC} \tag{11}$$

For a given detector incident rate r, the probabilities P_{AB} and P_{BC} could be obtained from time interval distribution [8]

$$P_{AB} = e^{-(T_p - T_1)\cdot r} \tag{12}$$

and

$$P_{BC} = e^{-(T+T_1)\cdot r} \tag{13}$$

The substitution of (12) and (13) into (11) leads to

$$P_{pf} = e^{-(T+T_p)\cdot r} \tag{14}$$

The rate at which "pile-up free" events will be recorded is given then by

$$r_{pf} = r \cdot e^{-(T+T_p)\cdot r} \tag{15}$$

Equation (15) shows that the proposed pulse processor does not add more pile-ups to the original signal when T_p=T. Furthermore, when T<T_p then the pile-up free recording rate is higher than the pile-up free rate of the pulses coming from the detector. It is also clear that the GI will suffer from more pile-up losses in comparison with the proposed processor because of finite time needed for resetting the GI.

III. Realization of the Pulse Processor

A block diagram of the circuit used to investigate our approach is shown in Fig.4. The circuit is built in pipeline fashion. That is, the data is transferred in and out for each of the units by the predetermined transition of the system clock (*CLK*). As a result there is at least one clock cycle delay between input and output data for a given unit.

The circuit is initialized by activating the *RESET* signal. Upon reset, the accumulator, input register and data pipeline are cleared, thus the necessary condition (3) is fulfilled.

The voltage pulse from the preamplifier or fast shaping amplifier is first digitized in a 36 MHz flash ADC (AD9048). The ADC and the pulse processor are tied to a common clock. The output of the ADC is connected to the data input of an edge triggered register. The code from the input register is applied both to one input of a digital subtraction unit and to a digital pipeline with programmable depth [9]. The output of the pipeline is then applied to the second input of the subtraction unit. Through this operation, the delayed signal is subtracted from the prompt ADC code. The result (including algebraic sign) is added to the content of a digital accumulator. The output of the accumulator represents the equivalent of the filtered signal. The accumulator content is then scaled down to the range of the ADC and is applied to the output FIFO register. It is not necessary to scale the data at this point. It could be done later, after the data has been stored. In such a case, however, more memory will be required in order to store the pulse height data.

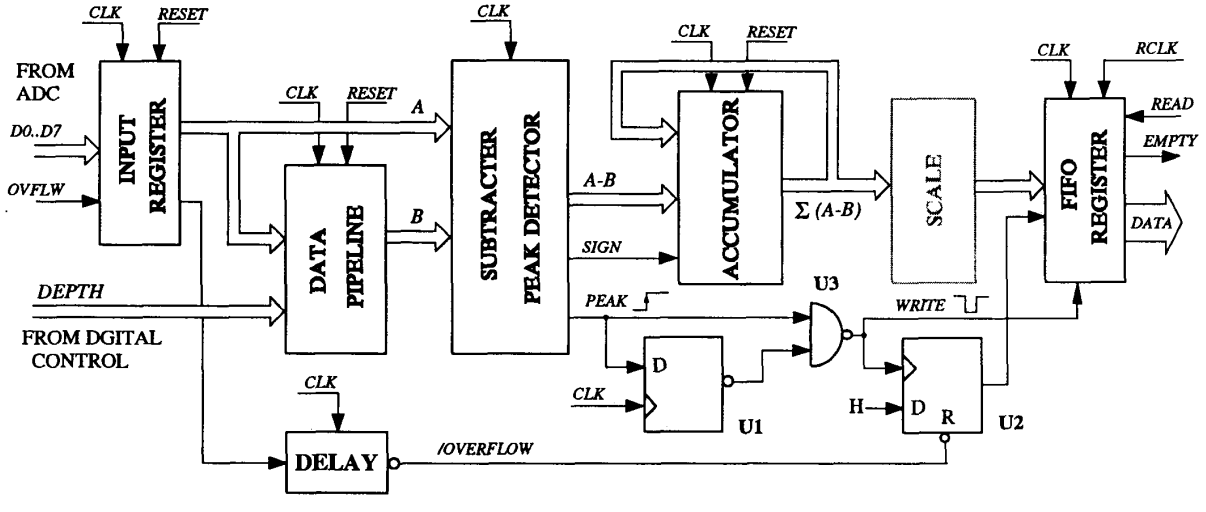

Fig. 4 Block diagram of the digital pulse processor

The peaks in the filtered signal are detected by the LOW to HIGH transition of the *PEAK* signal. A detailed diagram of the subtraction and peak detection unit is shown in Fig.5.

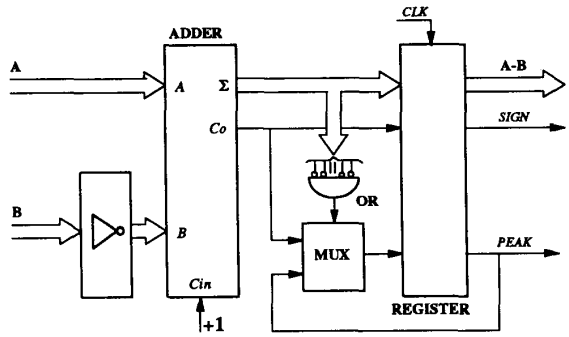

Fig.5 Block diagram of the subtraction and peak detection unit.

The prompt signal from the detector is introduced at one of the inputs of full binary adder. The delayed signal *B* is inverted and then applied to the other input of the adder. The carry in input of the adder is permanently kept high (addition of one). Hence, the binary code *B* is negated and the corresponding twos complimentary code is added to the binary code *A*. The sum output of the adder represents the twos compliment code of the difference *A-B*. The sign signal of the difference *A-B* is represented by the logic state of the carry out (*Co*). In order to avoid false peak detection, measures have been taken to prevent *PEAK* signal change, when the difference *A-B* is equal to zero, by holding its value to that in place in the previous iteration (clock cycle). For this purpose a simple circuit consisting of an OR gate and multiplexer (MUX) is used. If the result *A-B* is not equal to zero then the output of the MUX and the signal *PEAK* follow the changes in *Co*. If *A-B* is equal to zero, the circuit causes the signal peak to follow its previous state.

When the signal *PEAK* changes from LOW to HIGH, the content of the accumulator has reached a maximum. This transition of the signal *PEAK* is used to generate (Fig.4, D-trigger U1, NAND gate U3) one cycle wide pulse *WRITE* which enables a write operation into the output FIFO register, thus storing the peak value of the pulse. In order to improve the noise immunity, the LOW level at the inverting output of U1 could be extended with a time interval slightly shorter than the integration time interval *T*.

The ADC may also supply overflow information to the processor by means of an one bit line. If such signal is not available then it could be derived from the maximum possible ADC code. When the overflow signal has been generated between two adjacent detected peaks, then the second peak will be classified as an overflow event. The overflow signal is derived using the D-trigger U2. After detecting and storing a peak value, U2 is set to HIGH. If there is an overflow in the time interval to the next peak detection, U2 will be reset and will indicate an overflow. Otherwise, the next peak will be recorded as a normal event. The DELAY unit in the overflow signal line is used to avoid a shift between overflow and peak detection signals.

The FIFO register thus will be filled with data corresponding to the amplitude of the detected peaks. The register is read by an external MCA circuit. If the processing time of this circuit is sufficiently short then all of the detected events (including those corresponding to the noise) will be stored; otherwise some events may be lost. In the second case a digital threshold circuit might be used, following the FIFO register. If the amplitude exceeds the threshold, the value is recorded in a multichannel memory; if not, it is discarded. After the completion of the MCA cycle or after discarding the value, the next data from FIFO is read.

In addition, a timing signal can be derived from the *PEAK* signal that will be accurate to within one clock cycle.

The system is operated under full digital control, including the choice of data pipeline depth. This value can be selected to be any binary number from 1 to 254 samples. In the limit of small values, the circuit functions as a simple peak detector. Larger values for the pipeline depth result in longer averaging periods and consequently a reduced sensitivity of the amplitude measurement to the effects of high frequency noise.

IV. Preliminary Test Results

A prototype of the processor was built using high speed PLD's and fast TTL integrated circuits. The design allows the circuit to operate at clock frequencies up to 65MHz. The prototype operates with 8 bit input data; however the data width is expandable to 12 bits without reducing the sampling frequency.

One of the benefits of using digital integration technique is the improved differential nonlinearity (DNL) [3]. It is important to notice that the reduction of the differential nonlinearity is observed on event basis. That is, the amplitude of each pulse is digitized properly and the mean width of the MCA channels does not depend on the number of recorded events. DNL was estimated using a custom built tail pulse generator. The generated pulses have uniform distributed amplitudes which cover the range of the ADC. The shape of the pulses was made similar to the shape of the pulses from scintillation detector described later. Fig.6 shows the results of differential nonlinearity test. Case a) represents the simple peak detector DNL which is equal to the DNL of the ADC. It is seen that the DNL improves with increasing number of averaging samples *k*. The increase of DNL in case b) for the lower channels can be explained in terms of the finite resolution of ADC. At small amplitudes of the input pulses some of the adjacent samples describing the pulse shape are digitized to the same code. Thus, the effective number of the different digital codes used in the averaging sum are reduced.

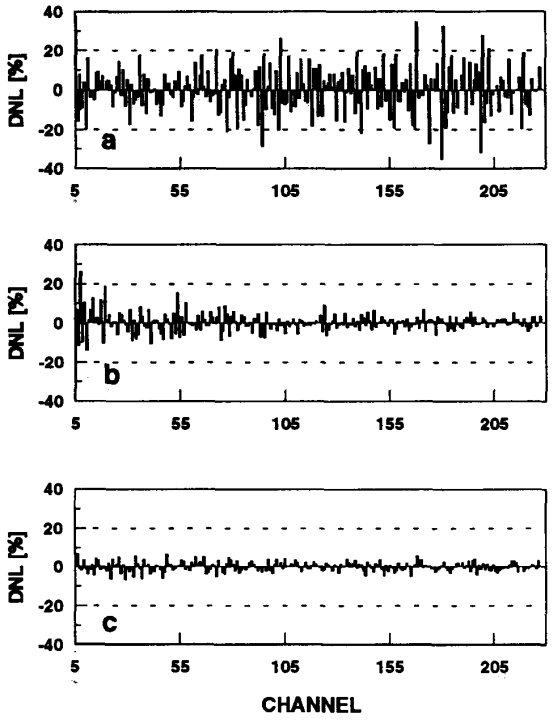

Fig.6 Differential nonlinearity: a) DNL=35% ±1.5%, k=1; b) DNL=27%, ±1.5%, k=9; c) DNL=6% ±1.5%, k=36.

The result of an integral linearity test measurement is shown in Fig.7. The data were obtained by using a calibrated tail pulse generator, and the integral nonlinearity found to be less than 0.5%.

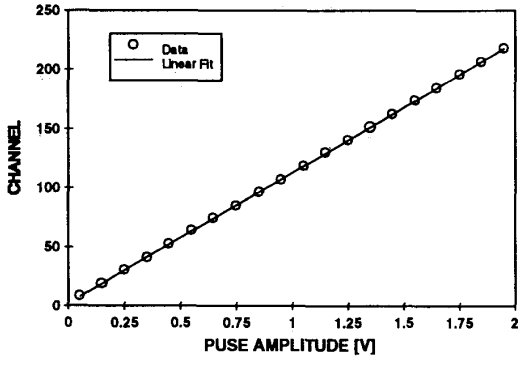

Fig.7 Integral linearity.

The major limitations of the method are the ADC resolution and the finite sampling frequency. In order to investigate the performance of the processor at different sampling frequencies a series of energy resolution measurements was carried out. The device was tested using a Cs-137 source and NaI(Tl) scintillation detector. An energy resolution of 6.4% at 662 keV was obtained using a conventional quasi-gaussian filter with 1.5μs shaping time constant. The pulses from the detector have 90 ns rise time (10%-90%). The integration time was fixed to $t_c \cdot k=1$ μs and the resolution was measured for different clock frequencies. The results for the energy resolution that was observed using the digital processor are presented in Fig.8.

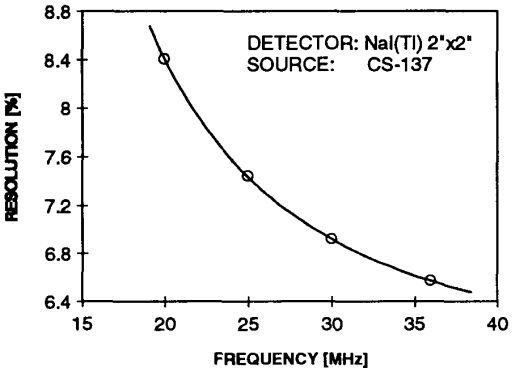

Fig.8 Energy resolution vs. sampling frequency.

As expected the energy resolution of the pulse processor is improved with increasing the sampling frequency because the area under the pulse is estimated using more samples. There is another source of error due to the fact that the processor clock and the pulse arrival time are not synchronized. This error also decreases with increasing sampling frequency. In addition inaccuracy is introduced because of the finite ADC resolution.

V. Conclusion

A real time digital pulse processor using a moving average technique has been designed. A threshold-free approach has been used for decreasing pileup losses. The processor was shown to be effective in reducing differential nonlinearity. The resolution of the processor is limited by the finite ADC resolution and the finite sampling frequency. Increasing the sampling frequency should improve the resolution in pulse height analysis and the timing precision.

V. References

[1] V. Drndarevich, P. Pyge and T. Gozani,, "Digital signal processing for high rate gamma-ray spectroscopy", *Nucl. Instr. and Meth.*, vol. A277, pp. 532-536, 1989.

[2] T. Lakatos, "Adaptive digital signal processing for x-ray spectrometry", *Nucl. Instr. and Meth.*, vol. B47, pp. 307-310, 1990.

[3] F. Hilsenrath, H. D. Voss, and J. C. Bakke, "A single chip pulse processor for nuclear spectroscopy", *IEEE Trans. Nucl. Sci.*, vol. NS-32, pp.145-149, February 1985.

[4] J. S. Karp et al., "Event localization in a continuous scintillation detector using digital processing" *IEEE Trans. Nucl. Sci.*, vol. NS-33, pp.550-555, February 1986.

[5] V. Radeka, "Trapezoidal filtering of signals from large germanium detectors at high rates", *Nucl. Instr. and Meth.* vol. 99, pp. 525-539, 1972.

[6] G. L. Miller and D. A. H. Robinson, "Transversal filters for pulse spectroscopy", *IEEE Trans. Nucl. Sci.*, vol. NS-22, pp. 2022-2032, October 1975.

[7] M. Cisotti et al., "Transversal filter approach of trapezoidal pulse shaping for silicon live targets", *Nucl. Instr. and Meth.*, vol. 159, pp.235-242, 1979.

[8] G. F. Knoll, *Radiation detection and measurement*, 2nd ed., New York, John Wiley and Sons, 1989, pp. 96-98.

[9] V. Jordanov, "Data pipeline has programmable depth", *Electr. Design News*,.vol. 37, pp.200-202, November 12, 1992.