Chapter 4

The Mechanism of Signal Averager

Introduction

NMR imaging is rapidly becoming a major diagnostic tool due to its many advantages, such as nonharzardous nature, capability of obtaining anatomical cross-sectional images in any direction, and high tissue discrimination capability. However, it has some minor disadvantages, such as its long data acquisition time due to spin-lattice relaxation time and low signal-to-noise ratio. Signal Averaging is a selective solution to overcoming the problem of low signal-to-noise ratio.

Signal averager is an instrumentation that extract signal waveform from noisy backgrounds. Signal averager has broad applicability in many scientific and engineering disciplines. Physicists and chemists use averaging to study the interaction of atomic particles by measuring selective energy absorption of spinning nuclei and electrons while they are being subjected to strong magnetic fields. Biologists average noisy biological signals to detect the animal's response to repetitive stimuli.

Signal Averager Configurations

Averaging has long been recognized as a powerful technique for recovering repetitive signals obscured by noise, especially when the frequency spectrum of the noise overlaps that of the signal. To utilize this technique, noiseless reference triggers that synchronized to the signal of interest are required. Triggers are applied to generate repetitive signal-acquisition sweeps at the trigger rate, provided each sweep is completed before application of the next trigger pulse. Sweeps repeat until the preselected number of sweeps are reached. During each sweep, newly acquired data updates prior stored data by performing the selected averaging algorithm. A block diagram of signal averager is illustrated in Fig. 4-1. The operations of each elements are discussed in the following.

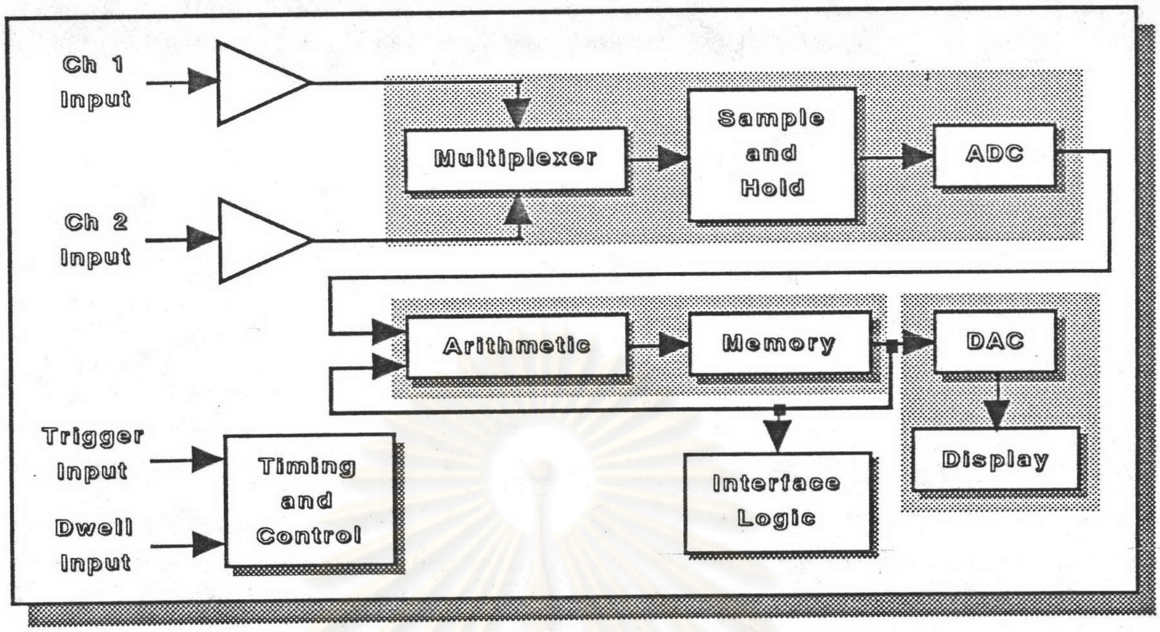

Fig. 4-1 Block diagram of signal averager.

Multiplexers Unit

When more than one quantity has to be acquired by analog-to-digital conversion, it is necessary either to time-division multiplex the analog inputs to a single ADC, or to provide an ADC for each input and combine the converter outputs by digital multiplexing. Multiplexer circuits may be used when the required conversion rate is not high, and thus one ADC can serve to convert several analog signals into digital form. The circuit of an analog multiplexer may be regarded as a single-pole multiposition switch in which the common is connected to the input of the next section.

Sample-and-Hold Unit

A sample-and-hold unit is a device having a signal input, an output, and a control input. It has two steady-state operating modes: *sample* in which it acquires the input signal as rapidly as possible and tracks it until commanded to *hold*, at which time it retains the last value of input signal that it had at the time the control signal called for a mode change (Fig. 4-2). Sample-and-hold usually have unity gain and are noninverting. Sample-and-hold unit is particularly useful where fast changing signals must be multiplexed in data acquisition systems or where momentary signals must be captured and held.

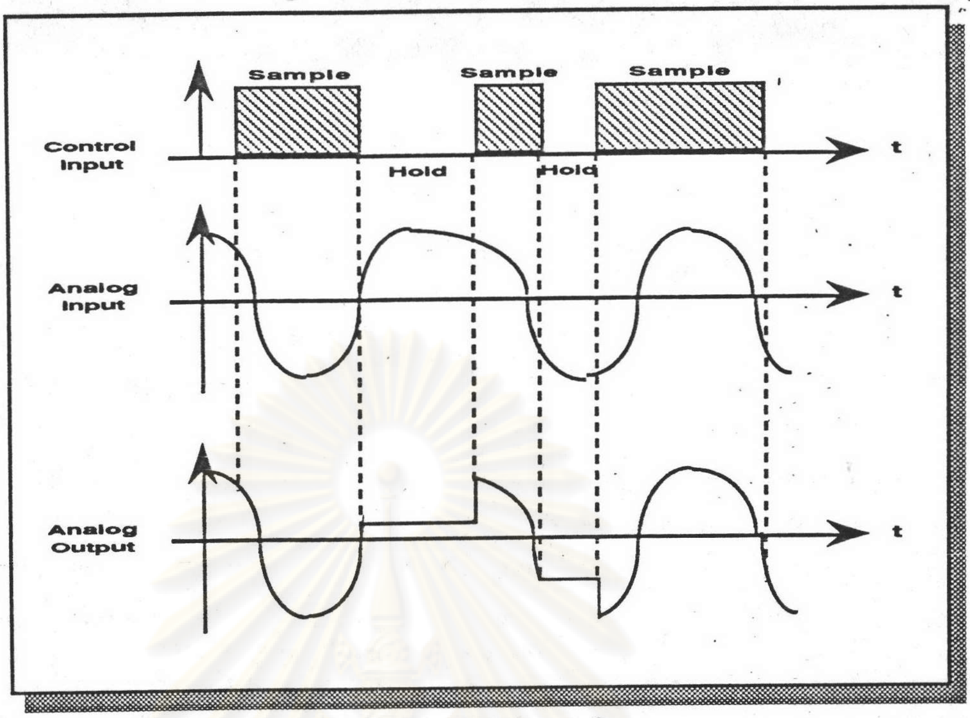

Fig. 4-2 Typical sample/hold waveform

The sample-and-hold maintains the input to A/D converter constant during the conversion interval; meanwhile, the multiplexer is seeking the next channel to be converted, either randomly or sequentially. As soon as conversion is completed, the sample-and-hold samples the newly established input, and the cycle is repeated.

Analog-to-Digital Convertor (ADC) Unit

In ADC, the digital output number depends on the ratio of the quantized input to the full-scale reference. If the reference is allowed to change in response to a second analog input, the digital output will be propotional to the ratio of the analog signal to the reference signal. ADC is used to convert the analog input signal of interest to provide a representative digital output. This digital information can be further recorded in memory or processed by arithmatic operations.

Timing and Control Unit

Triggers applied to the averager timing unit generate repetitive signal-acquisition sweeps at the trigger rate, provided each sweep is completed before application of the next trigger pulse. Sweeps repeat until the preselected number of sweeps are reached. During each sweep, ADC of sequential signal (+noise) samples occurs at preselected regular intervals called dwell times. An internal controller manages sucessively digitized samples by register into sequential memory addresses. A number of samples in a complete sweep is limited by the length of memory, with a time duration equal to dwell time x memory length. Newly acquired data updates prior stored data in accordance with the selected averaging algorithm. Since the synchronization of trigger signal and signal of interest, acquired data in each address is always from the same preassigned waveform segment. Consequently the stored signal reinforces itself with each sweep whereas random noise and unrelated extraneous signals tend to self cancel. The signal-to-noise ratio tends to improve, as sweeps progress.

<u>Arithmetic Unit</u>

An Arithmetic unit is a digital circuit that perform arithmetic calculations. The circuit is designed according to averaging algorithms. Newly acquired data and prior stored data are the arguments of the operation.

Three averaging algorithms are often used: linear summation averaging, normalized averaging and exponential averaging. A large improvement in signal to noise or the greatest accuracy can be achieved by using linear summation averaging. However, this mode is somewhat less convenient to use than normalized averaging which can stop the averager as soon as enough noise has been stripped away. The third mode, exponential averaging, is useful when the signal of interest is itself changing during the experiment.

1. <u>Linear Summation Averaging</u>

Algorithm:

$$A_s = \sum_{k=1}^{s} I_k$$

S	=	number of elapsed sweeps
A_S	=	average after S sweeps
I_k	=	Kth sample value

Accumulated signals grow linearly from stable baseline with each successive sweep S. Since all data samples are weighted equally, occasional noise bursts or signal irregularities are not given disproportionate emphasis during the averaging interval. Consequently, accumulated results are not interfered by unusual or uncontrollable occurrences. Because processed signals are not subject to computational round-off errors, the full capacity of the memory width can provide faithful waveform reproduction even of extremely weak signals.

2. Normalized Averaging

Algorithm:

$$A_s = A_{s-1} + \frac{I_s - A_{s-1}}{2^J}$$

A_s	=	average after S sweeps
A_{s-1}	=	average after S-1 sweeps
I_s	=	Sth input sample
J	=	Positive integer, slaved to elapsed sweeps

In normalized averaging, accumulated data are always displayed in full scale, even during the first sweep, giving immediate sensitivity calibration. The signal remains constant as noise removes with succeeding sweeps. Averaging can be stopped when satisfied with the resulting output signal-to-noise ratio, or when the number of elapsed sweeps reaches the preset number of sweeps.

3. Exponential Averaging

Algorithm:

$$A_s = A_{s-1} + \frac{I_s - A_{s-1}}{2^n}$$

A_s	=	average after S sweeps
A_{s-1}	=	average after S-1 sweeps
I_s	=	Sth input sample
2^n	=	Weighting function

Exponential averaging is useful for tracking changing waveform trends, because the oldest data is discarded. Like Summation averaging, the accumulated signal builds from the baseline. However, in this mode the signal grows exponentially with each sweep.

Memory Unit

Memory unit is used to store the result of operation from arithmatic unit. It is composed of a number of memory segments arranged in contiguous line. Each segment can be access by first assigned its address. An amount of memory segments refered as memory length determines the number of samples in a complete sweep. Consequently, the memory width limits the maximum number of sweeps especially in linear summation averaging since accumulated signals grow linearly with each successive sweep.

Digital-to-Analog Convertor (DAC) Unit

In DAC, the output voltage or current depends on the reference value chosen to determine full scale output. If the reference varies in response to analog signal, the output is propotional to the product of the digital number and the analog input. The product's polarity depends on the analog signal polarity, and the digital coding and conversion relationship.

The conversion of digital values to proportional analog values is a necessary task in order that results of digital computations can be used and easily understood in the analog world. For signal averager, DAC is used in order to convert the stored data in memory into analog form. Thus, the averaged signal can be presented directly to a display such as an CRT.

Interface Logic Unit

Signal averager can be interfaced with a main computer via interface logic unit. Using main computer as a central controller, it is possible to initiate, record, process and analyze the averaged data. By altering the parameters to test hypotheses, the results of many such experiments can be easily stored on permanent media on the main computer for later analysis.