ACQUISITION HARDWARE

Scientists are always seeking better methods to record and store the growing amounts of data generated in experiments. In electrophysiological experiments, the data are most often in the form of voltage waveforms whose magnitudes vary with time. Data in this form are appropriate for displaying on an oscilloscope or chart recorder, but entirely inappropriate for storing on a computer disk. Since computers can only store discrete numbers, a process of "analog-to-digital" conversion must be undertaken to convert the analog data into a compatible format for the computer.

Fundamentals of Data Conversion

The analog-to-digital (A/D) conversion process can be illustrated by recording temperature during the course of a day, using pencil and paper. One way to do this is to look at the mercury level in a glass thermometer every 30 min and write the temperature in a column on a piece of paper. The result is a sequence of numbers, recorded at uniform time intervals, which describes the variations in the temperature during the day. This process illustrates the execution of an analog-to-digital conversion and the creation of an array of data that could be typed into the computer and stored on disk.

Time	Digitized Temperature °C
6:00 AM	10
6:30 AM	11
7:00 AM	11
7:30 AM	12
8:00 AM	13
8:30 AM	13
9:00 AM	14
9:30 AM	14
10:00 AM	14
10:30 AM	15
11:00 AM	15
11:30 AM	15
12:00 PM	17
12:30 PM	18
1:00 PM	19
1:30 PM	19
2:00 PM	19
2:30 PM	18
3:00 PM	16
3:30 PM	15

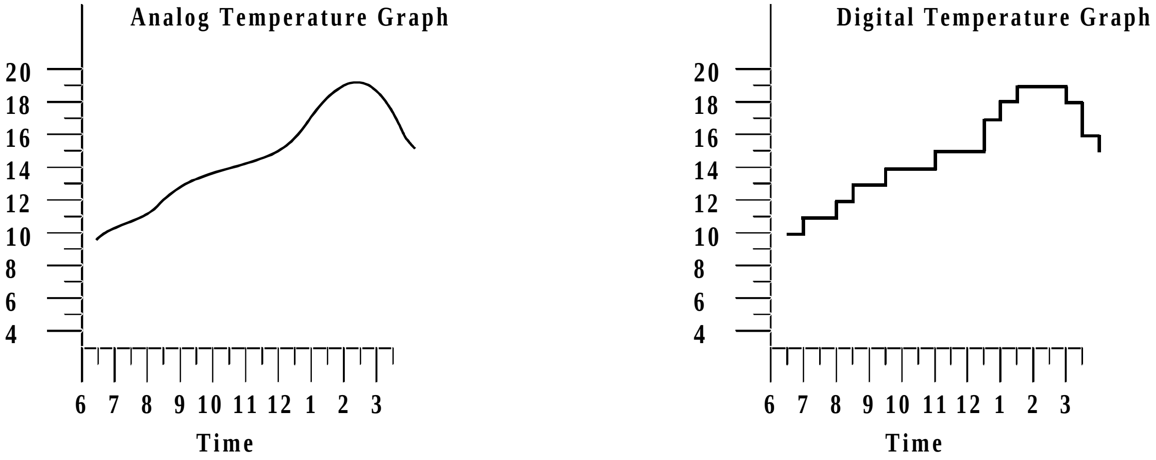

Figure 9-1. Analog-to-Digital Conversion

This figure illustrates the analog and the digital representations of the temperature data presented in the above table.

A much easier way to accomplish the same task would be to use a data acquisition board in the computer to directly record the temperature every 30 minutes. In this case the thermometer must report the temperature in a form that the analog-to-digital converter (ADC) can read. Nearly all ADCs require that the data be presented as a voltage. To do so, one uses a different type of thermometer, such as a temperature-dependent resistor in an appropriate circuit, that generates an analog voltage proportional to the analog temperature. The computer is instructed to initiate an analog-to-digital conversion every 30 minutes and store the results on disk.

Sometimes the computer is required to generate an analog waveform that will be proportional to a list of numbers that are held in memory or on disk. In this case, a digital-to-analog (D/A) converter (DAC) is used. The principles of operation of a DAC are similar to those of an ADC, but the operation is the reverse.

Quantization Error

The ADC converter generates binary numbers that have finite resolution. That is, a small range of analog values will all produce the same binary number after conversion. Returning to the temperature-measurement example above, the hypothetical ADC rounded off all of the temperature values to the nearest degree Celsius reading. That is, only numbers between 0 and 99°C in steps of 1°C were allowed. Clearly, the temperature changes continuously between each 30-min reading, not in 1°C jumps. The range of analog values that are recorded as the same digital number is the quantization error.

In an ADC, the total measurement range (*e.g.*, 0 - 100°C) is divided into a fixed number of possible values. The number of values is a power of two, often referred to as the number of "bits." Commonly, these values are:

8 bit = 2^8 = 256 values
12 bit = 2^{12} = 4,096 values
16 bit = 2^{16} = 65,536 values

To illustrate the impact on the resolution of using an 8-bit, 12-bit or 16-bit ADC, consider the temperature-measurement example where the electronic thermometer circuit generates an analog output from -10 V to +10 V for temperatures in the range -100°C to +100°C. In this case, the resolutions are:

8 bit = 78.4 mV = 0.784°C
12 bit = 4.88 mV = 0.0488°C
16 bit = 0.305 mV = 0.00305°C

On the surface of it, it would seem desirable to use a 16-bit ADC system. However, 16-bit systems are generally significantly more expensive, or at an equivalent price, significantly slower. Furthermore, the extra resolution is not always realizable. In some systems, there is too much noise introduced by the circuits inside the computer. This noise can be made negligible compared with the resolution of a 12-bit ADC, but it rarely can be made less than the resolution of a 16-bit ADC. In fact, some so-called 16-bit systems have so much noise that they only offer about the same usable resolution as a 12-bit system. Nevertheless, manufacturers usually market these as 16-bit systems because if they are used with substantial averaging, the noise can be reduced and the benefits of the higher resolution can be realized.

In the majority of physiological applications, the resolution of a 12-bit ADC is sufficient. It allows measurements at better than 0.025% resolution of the full range. For example, consider a voltage-clamp amplifier whose output is 100 mV/nA, spanning ±10 V (±100 nA). Current can be measured with a 12-bit analog-to-digital converter in 0.0488 nA increments.

If the ADC is preceded by a programmable-gain amplifier, signals that occupy only a small range, *e.g.*, ±1 V, can be amplified before conversion so that the full resolution of the ADC can be utilized.

The resolution of the standard 12-bit ADC with an input range of ±10 V is 4.88 mV. Although this resolution does not represent a difficult number for a computer-based system to use, some researchers prefer a round number such as 5.00 mV. This can easily be achieved by setting the span of the ADC to ±10.24 V instead of ±10 V. An increasing number of ADC systems are now being designed with the ±10.24 V span (*e.g.*, the Axon Instruments Digidata 1200 data acquisition system).

Choosing the Sampling Rate

There is a well-known theorem known as the *sampling theorem* or the *Nyquist theorem* stating that data should be sampled at a frequency equal to twice the bandwidth of the signal or faster in order to prevent an artifactual increase in the noise, due to a phenomenon known as aliasing (see **Chapter 12**), and guarantee that the analog signal can be reconstructed unambiguously from the digital samples.

In practice, it is common to sample at a rate significantly faster than the minimum rate specified by the sampling theorem. This is known as *oversampling*. Exactly how much oversampling should be used depends upon the type of experiment.

For experiments where the data are analyzed in the frequency domain (*e.g.*, noise analysis, impedance analysis), it is common to oversample only modestly. The main concern is to prevent aliasing. An anti-aliasing filter is introduced between the signal source and the analog-to-digital converter to control the bandwidth of the data. The factor of twice the analog bandwidth required by the sampling theorem is only applicable if the anti-aliasing filter is ideal, *i.e.*, the gain is unity in the pass band and it abruptly changes to zero in the stop band. Ideal filters cannot be realized, although they can be closely approximated. For frequency-domain analysis it is common to use sharp cutoff filters such as Butterworth or Chebyshev realizations. Sampling is typically performed at 2.5 times the filter bandwidth. For example, if the data are filtered at 10 kHz they should be sampled at about 25 kHz. Slower sampling rates are unacceptable. Faster sampling rates are acceptable, but offer little advantage and increase the storage and analysis requirements.

For experiments where the data are analyzed in the time domain (*e.g.*, pulse analysis, I-V curves), greater oversampling is required because reconstructing the analog signal requires not only an ideal anti-aliasing filter but also an ideal reconstruction filter. The simplest and most common reconstruction filter is to join each sample by a straight line. Other techniques, such as cubic-spline interpolation, can be used, but due to their much more demanding computational requirements they are used infrequently.

The reconstruction problem is discussed in the *Sampling Rate* section in **Chapter 12**. There is no commonly accepted rule regarding the sampling rate of data for time-domain analysis. In general, five times the data bandwidth is a common sampling rate, and ten times is considered good. Sampling at 20 times is excessive and is rarely used.

Converter Buzzwords

The terms in this section are defined for a D/A converter. With simple reorganization, these definitions can be re-cast for an A/D converter. Only the more esoteric terms have been defined in this section. It is assumed that the reader already understands conventional terms such as "voltage offset."

Gain Accuracy

Gain accuracy is the closeness with which the output of the D/A converter corresponds to the specified full-scale value when the digital input to the D/A converter is the maximum value.

Linearity Error

Linearity error is the maximum deviation of the D/A output from a straight line drawn between the minimum and the maximum output.

Differential Nonlinearity

When the digital control word changes by a minimal step, *i.e.,* one least-significant bit (LSB), the output value should change by 2^{-n} of the full-scale value, where n is the number of bits in the converter. Any deviation from this ideal change is called the differential nonlinearity error. It is expressed in multiples of LSBs.

Least Significant Bit (LSB)

LSB is the value of the smallest possible digital increment.

Monotonicity

Monotonic behavior means that for every increase in the digital input to the D/A converter there will be an increase in the analog output. This requires that the differential nonlinearity error be less than 1 LSB.

Data Formats

Understanding the terms used to describe data formats can enhance ones appreciation of the D/A conversion process. Following are a few of those terms.

Offset Binary

Offset binary is the simplest binary numbering scheme. In this scheme the smallest value is all zeros (*i.e.,* 0000 0000) and the largest number is all ones (*i.e.,* 1111 1111). For a D/A converter wired for a bipolar output (*e.g.,* ±10 V), analog zero occurs when the most significant bit (MSB) changes to one (*i.e.,* 1000 0000).

2's Complement

This is the binary numbering scheme used by IBM-compatible computers and many other computers. In this scheme the most significant bit (MSB) is inverted so that the smallest value is 1000 0000 and the largest number is 0111 1111. For a D/A converter wired for a bipolar output (*e.g.,* ±10 V), analog zero occurs when all the bits are zeros (*i.e.,* 0000 0000).

Big Endian vs. Little Endian

Each 16-bit word consists of two 8-bit bytes. In Intel 80x86 microprocessors (*i.e.,* the ones used in IBM-compatible personal computers), the least significant byte is stored first and the most significant byte is stored second. This is referred to as "little endian." In Motorola 68000 microprocessors (*i.e.,* the ones used in Macintosh computers) and most RISC microprocessors (*i.e.,* the ones used in most "workstations"), the most significant byte is stored first and the least significant byte is stored second. This is referred to as "big endian." This simple inconsistency makes data transfer between different computers tricky.

Deglitched DAC Outputs

When the input to a digital-to-analog converter changes value, the analog output ideally moves rapidly and monotonically towards its new value. While usually this is the case, for certain changes in value a transient is superimposed on the changing output. This transient is known as a "glitch."

Glitches occur when several bits in the control word change. Each bit is connected to a set of internal digital logic and analog switches. There is variability in the response time of each part of the circuit, and therefore the response to the new control word does not happen at the same instant for each bit. In addition, charge is injected from the digital logic circuitry into the analog switches. Glitches are worst when a large number of the bits of the digital input to the D/A are changed. For example, when the analog output is required to go from negative to positive, the digital input bits from 1111 1111 1111 to 0000 0000 0000. That is, every single bit changes. This change generates the worst glitch.

Glitches can be prevented by including a sample-and-hold circuit at the output of the D/A converter. The sample-and-hold is normally in the "sample" mode so that the output of the D/A converter feeds straight through. However, for one or two microseconds, perhaps less, after the digital input to the D/A converter is updated, "hold" mode is selected so that the glitch, if present, is blocked.

In modern integrated-circuit D/A converters, efforts are made in the design of the chip to match the propagation and activation delays of each bit in the circuit and to minimize the charge injection so that glitches are small in the first place. It is thus becoming increasingly less common for a deglitching circuit to be included. On the other hand, the high-level of integration in modern D/A converters sometimes introduces another problem, called *feedthrough noise*. This problem occurs when the digital latches that contain the word for the D/A converter are integrated into the same chip. It may happen that because of the proximity of the digital circuits to the D/A converter, digital noise couples into the D/A converter circuit. The coupled signal manifests itself as a pulse on the output of the D/A converter circuit each time the digital word is updated. Strangely, this feedthrough noise often appears even if the D/A value is being held constant. This is because, for simplicity, most software that simultaneously performs A/D and D/A conversions updates the D/A word once per A/D sample, even if the D/A value is not changing. The best solution is to eliminate the feedthrough noise at its source by using separate integrated circuits for the D/A converter latches. This is the approach used in the Digidata 1200.

DMA, Memory Buffered, I/O Driven

It is essential for the acquired data to be transferred from the ADC board to the computer's disk for permanent storage. The simplest way to achieve this is under the direct control of the microprocessor. Since this technique makes heavy use of the input/output ports of the microprocessor, it is often referred to as "I/O driven." A crystal-controlled clock on the ADC board initiates A/D conversions at regular intervals. After each conversion is completed, a status bit on the board is set high to indicate that the conversion is complete. The microprocessor reads the converted value from the ADC board and moves it to a location in memory. If necessary, the microprocessor moves a new D/A value to the board, then waits for the next converted word. When an array of converted data has been filled up, the microprocessor stops the conversions, displays the data and saves it to disk. On IBM-compatible personal computers, it is not possible to continue storing A/D conversions during the disk write.

In many experiments, continuous gap-free acquisition to disk is required. To achieve this, more sophisticated techniques to handle the incoming data are required, such as direct memory access (DMA) or memory buffering.

DMA acquisition requires the presence of a DMA controller circuit either on the computer mother-board or on the ADC board. The purpose of the DMA controller is to move arrays of data between two locations without involving the microprocessor. To use the DMA controller, the microprocessor sets it up in advance by telling it how many data values to move, where to get them and where to put them. Once the process begins, the microprocessor is free to perform other tasks. To achieve gap-free acquisition to disk, Axon Instruments' AxoTape and FETCHEX (one of the programs in pCLAMP) software uses the Digidata 1200 ADC board under DMA control. While the DMA controller is loading acquired data into the first of two memory buffers, the microprocessor simultaneously displays the data. When the first memory buffer is full, the DMA starts filling the second buffer without a pause. Once this commences, the microprocessor takes the opportunity to write the contents of the first array to disk. When the transfer to disk is completed, the microprocessor resumes real-time display of the data that is filling the second memory buffer. When the second memory buffer is full, the DMA starts filling the first buffer again without pausing. This cycle continues indefinitely.

Memory buffering is another way to achieve gap-free acquisition. In this case, the ADC board contains its own memory split into two halves. While one half is being filled with freshly converted data, the microprocessor is free to display the data from the other half and write it to disk. This technique is used by Axon Instruments' AxoData software to provide gap-free acquisition with the Instrutech ITC-16 data-acquisition hardware.

An advantage of memory buffering compared with DMA data transfer is that it can potentially support extremely fast data conversion rates for fixed-length acquisitions. The on-board memory is usually high-speed static memory and it can be filled at a rate sampling of a few megahertz. On the other hand, most 12-bit A/D converters operate at rates much slower than 1 MHz, and the DMA controller on most PCs can handle the data flow with ease. A disadvantage of the memory buffering technique is that it is more expensive because of the need to supply memory and control circuitry on the ADC board.

If analog output waveforms are to be generated simultaneously with the acquisition at maximum speed, a second DMA controller channel is required to transfer the DAC values from the computer memory to the DAC board. Similarly, if synchronous digital outputs are to be generated, either a third DMA channel is required or the digital data must be transferred simultaneously with the DAC data. The latter technique is used in the Digidata 1200.

Interrupts

Much of the activity during experiments occurs at unpredictable times, *e.g.,* when solution exchange is complete, when a seal forms or when a pre-synaptic input is found. It is essential to have a means of interrupting the computer during data acquisition to tell it that a significant event has occurred.

Special lines on the I/O bus of the computer are devoted to carrying "interrupts." Interrupts are generated by the system clock (so that the time of day can be maintained), by the keyboard and by the ADC board. A typical use of the keyboard interrupt is to allow the user to abort the acquisition by pressing the **<Esc>** key. A typical use of the ADC interrupt is to trigger the start of an acquisition. Another is to attach tags to the data during acquisition to indicate that a specific event has occurred.

Timers

Most ADC systems have several timers available for a multitude of tasks. The most essential is the provision of a regular clock signal to initiate each A/D conversion. In most systems, one D/A conversion is performed for each A/D conversion, but this is not required and some systems provide for running the D/A converter at a different clock rate to the A/D converter.

Additional timers are used to implement "gear shifting." This is a technique wherein the acquisition rate is rapidly changed (gear shifted) during acquisition from a low to a high rate or vice versa, without stopping the acquisition.

Uncommitted timers are generally provided for use in frequency measurement, event counting or interval timing. For example, the frequency of nerve spikes can be counted if they are first detected by an event detector that puts out a digital pulse for every spike. An example of event counting is the measurement of the number of photons collected by a photomultiplier tube in a fixed interval. An example of interval timing is measurement of the period of a spinning optical filter wheel so that the optical data collected can be normalized against fluctuations in the rotational speed of the wheel.

Digital I/O

Digital outputs are used during experiments to control external equipment. For example, an oscilloscope can be triggered before the acquisition commences, or a flash lamp or an isolated stimulator could be activated during the acquisition. In other experiments, several solenoids can be sequentially activated before, during and after the acquisition.

Digital inputs are used routinely in industrial control applications for monitoring the state of solenoids and other apparatuses, but they are rarely used in electrophysiological experiments. The main application for digital inputs in electrophysiology is for triggering acquisitions and for indicating that a tag should be attached to the data.

Optical Isolation

Computers provide quite hostile environments for plug-in data acquisition boards. The rapid switching activity of the digital logic circuits causes electronic noise to radiate directly into plug-in boards installed by the user and to be introduced into the ground reference used by the plug-in boards. Even worse, the low-cost switching power supplies used by all computers that Axon Instruments has tested generate significant ground noise with a frequency of about 30 kHz.

Nevertheless, it is usually possible to design 12-bit data acquisition systems that do not suffer from extraneous noise pickup. This is achieved by careful layout and good grounding. Because of their greater resolution, the task is more challenging for 16-bit systems. Grounding and shielding, if applied with great skill and diligence, can sometimes be made to work acceptably for the A/D converter, especially when combined with differential recording. However, it is very difficult to achieve acceptable noise levels on the DAC outputs. The following approach is often invoked for 16-bit systems.

First, move the A/D converter off the plug-in board in the computer to a separate board housed in an external box. This eliminates the problems due to direct pickup of noise generated inside the computer housing. However, it does not eliminate the noise introduced through the system ground. The noise in the system ground can be eliminated by using optical isolation. In this technique, the digital signals to and from the computer are not connected directly. Instead, an optical coupler is interposed in each digital line. A separate power supply is provided for the A/D and D/A side of the optical couplers. This technique allows a low-noise ground to be maintained for the A/D and D/A converters and for the experimental setup. Optical isolation significantly increases the expense and complexity of the data acquisition system, but it is likely to be the only method capable of true 16-bit operation.

Operating Under Multi-Tasking Operating Systems

Data acquisition systems are not naturally compatible with multi-tasking operating systems. As a general rule, data acquisition systems are designed to use all of the resources of the computer (*e.g.,* display and disk) and they use these resources on demand, rather than when the computer makes them available.

Nevertheless, by carefully designing the hardware and the software, some degree of compatibility is achievable. However, data acquisition programs designed to work under multi-tasking operating systems are just becoming available commercially. One such program is AxoData for the Macintosh II computer.

Ideally, the data acquisition hardware should not be I/O-driven since this requires the full-time attention of the computer. However, if the data acquisition software can take control of the computer, I/O-driven hardware can be used. DMA control or local memory buffering are more suited to operation under a multi-tasking operating system since they can be used without taking over control of the computer.

Interrupt handling represents an acute problem for data acquisition in multi-tasking operating systems. Usually, when an interrupt occurs an immediate response is required. If the multi-tasking operating system is busy updating a screen for a word processor, an immediate response will clearly not be forthcoming. Therefore, high performance clearly requires that the data acquisition task runs as the foreground application. Another problem is that intelligent computer buses used in modern computers, such as NuBus on the Macintosh II and MicroChannel on the IBM PS/2, generally have long interrupt latencies of several hundred microseconds or more. One way to minimize the problem of excessive interrupt latency is to ensure that the data acquisition routine has the highest priority and, in many cases, exclusive control over the computer. Another way to minimize excessive interrupt latency is to provide hardware support in the acquisition system. For example, a memory-buffered data acquisition system can be configured by the acquisition software and left in a "primed" state. When it "sees" an external trigger it can generate an interrupt, then commence the acquisition before it even receives a response from the host.

Another significant problem when running data acquisition under multi-tasking operating systems results from the tendency of these systems to use a "windowed" graphical user interface. The operating system manages all drawing to the video screen but the data acquisition software usually prefers to write directly to the video screen for performance reasons. To operate under a windowed interface the data acquisition software has to cooperate in its use of the screen, and the overhead of this cooperation can significantly slow the performance.

Other problems arise when trying to run a DOS program in a DOS window under Microsoft Windows 3.1 in enhanced mode. For example, Microsoft Windows 3.1 does not allow the program to issue instructions directly to the DMA controller. Instead, it traps the instructions from the data acquisition program and passes on only those that it regards as acceptable. Unfortunately, Microsoft Windows 3.1 does not support the command to instruct the DMA controller to automatically keep filling memory buffers without a pause. As a result, the DOS programs AxoTape and FETCHEX, two of Axon Instruments' programs for continuous data acquisition and simultaneous display, are incompatible with Microsoft Windows 3.1 enhanced mode.

Software Support

Nowadays fewer researchers are writing their own data acquisition and analysis software using low-level languages and device drivers. Most researchers are using one of two types of commercially available software. The first type is a rich development environment such as AxoBASIC. This environment allows researchers to design their own software without having to become experts in the low-level control of the data-acquisition hardware. The second type of software is a turn-key package, such as pCLAMP, AxoTape or AxoData. These packages provide sophisticated acquisition and analysis, but only limited ability to customize them. Nevertheless, they are extremely popular because they perform well and are easy to learn.

When choosing hardware and software for a data acquisition system, we recommend that you choose the software first and then purchase the hardware recommended by the software applications that you have selected.

PC, Macintosh and PS/2 Considerations

Three personal computers are commonly used for data acquisition in labs. In decreasing order of use, these are the PC-compatible, the Macintosh II and the PS/2 MicroChannel computers. Workstations from manufacturers such as Sun are also appearing in labs, but they are not frequently used for data acquisition. Many existing acquisition setups are based on the PDP-11 computer, but these are routinely being replaced. Atari computers are also used for lab data acquisition, mainly in Europe.

IBM-compatible personal computers using the industry standard AT bus (also known as Industry Standard Architecture, ISA) are by far the most prevalent data acquisition computers in labs. These computers are inexpensive and there is an enormous range of software available for them.

PS/2 MicroChannel computers potentially offer more functionality in networked environments, but practical benefits compared to the AT bus are not obvious. Since they are generally more expensive than PC compatibles, these computers have not made major inroads into labs.

Macintosh II computers are increasingly common in laboratories. These computers and the data acquisition hardware are significantly more expensive than equivalently powerful PC compatibles, but they have a number of distinct advantages. Data can be exchanged between programs, and sophisticated and easy to use analysis software is available. Many of these benefits are now available to PC users because of the popularity of Microsoft Windows.

In terms of suitability for data acquisition, the IBM compatible personal computer is the easiest to design hardware for and the easiest to control. The bus does not offer special support for multi-tasking, which means that it is relatively simple. Interrupt latencies are very short and DOS allows direct access to the interrupts. The NuBus architecture used in the Macintosh II is much more sophisticated. It arbitrates the use of the bus by multiple intelligent plug-in boards.

The disadvantage of this power is that interrupt latencies are much longer. This, and the sophisticated demands of the operating system, has slowed the development of data acquisition hardware and software for the Macintosh.

Archival Storage (Backup)

It is prudent to have some means of backing up experimental data. There are two distinct approaches to archival storage of data. One is to record the data initially to a laboratory tape recorder. After the experiment, the portions of interest are digitized and transferred to disk. The second is to digitize the data as it occurs and store it on disk. After the experiment, or even after analysis, the data can be saved on backup media.

The former approach is quite common for continuous data, such as single-channel patch-clamp data. Experimenters commonly store the data during the experiment onto a laboratory tape recorder. This tape recorder could be an FM recorder, a VCR recorder or a DAT recorder. The advantage of this technique is that laboratory recorders have huge storage capacities. A VCR system can typically store three hours of data at a 10 kHz bandwidth. If this data were sampled at 50 kHz using a computer-based data acquisition system, three hours of recording would require 1.08 gigabytes of disk storage. While storage capacities of this magnitude are almost achievable using erasable optical disks, these disks and their drives are still quite expensive and not commonly used.

After the experiment has been completed, the data are transferred at leisure to the computer. Two transfer techniques are possible. If the recorder is a VCR or a DAT system the data are internally stored in digital format. It is possible to transfer the digital data direct to the disk. The second technique, which is suitable for all recorders, is to use the computer's ADC system to digitize the analog data replayed from the recorder. In general, the second approach is more flexible. The digitization can be performed at any rate and an analog filter can be used during the transfer to lower the bandwidth of the data. When a direct digital transfer is used, the data must usually be transferred at the internally used sampling rate of about 44 kHz. Some digital transfer systems allow this rate to be decimated in order to reduce the storage requirements. Filtering is not possible unless a real-time digital filter is implemented during the transfer. This is not generally available.

The second approach, direct digitization and storage in the computer, is most common for episodic data, such as a series of fixed-length steps used to determine an I-V curve. In these types of experiments the data storage demands are not nearly as large. For example, if 20 episodes of 1024 samples are recorded at 1s intervals, and then the procedure is repeated every minute, the total storage requirement for a three-hour experiment is 7 megabytes, quite a manageable amount.

The advantages of direct digitization and storage for episodic data are significant. First, the data can be acquired at rates that exceed the capacity of laboratory recorders. For example, the Digidata 1200 can acquire episodes at 330 kHz. Second, trigger information is maintained without jitter. If episodically acquired data are stored on the laboratory recorder, small delays are usually introduced between the signals on different channels. Third, protocol information, such as the command waveforms, can be stored with the data. If the data are only stored on the laboratory recorder, this information is lost. For these reasons, Axon Instruments strongly advises customers not to record episodic data onto tape.

A common variation on these two methods is to record data simultaneously to the laboratory recorder and the computer. This combination is often used when continuous single-channel patch-clamp data are acquired. Recording to the computer might only be activated when the data looks most promising, but all of the data are stored on the laboratory recorder for post-acquisition analysis.

When data are stored directly on the computer's hard disk, a means must be available for backup and archival storage. Since the data are in computer format, it is most common to archive the data on conventional computer backup media. Floppy disks are the most ubiquitous backup media. These are cheap and easily interchanged between computers. However, if there are many megabytes of data to be saved it is tedious to use floppy disks. As mentioned in **Chapter 8**, relatively inexpensive and rapid backup can be achieved using a streaming cartridge tape backup device. These are available from most computer vendors. Backup devices based on the DAT tape format are becoming available and carry the promise of prodigious storage capacity at very reasonable prices.