Behavior Research Methods, Instruments, & Computers
1990, 22 (6), 550-559

The Haskins Laboratories' pulse code modulation (PCM) system

D. H. WHALEN, E. R. WILEY, PHILIP E. RUBIN, and FRANKLIN S. COOPER
Haskins Laboratories, New Haven, Connecticut

The pulse code modulation (PCM) method of digitizing analog signals has become a standard both in digital audio and in speech research, the focus of this paper. The solutions to some problems encountered in earlier systems at Haskins Laboratories are outlined, along with general properties of A/D conversion. Specialized features of the current Haskins Laboratories system, which has also been installed at more than a dozen other laboratories, are also detailed: the Nyquist filter response, the high-frequency preemphasis filter characteristics, the dynamic range, the timing resolution for single- and (synchronized) dual-channel signals, and the form of the digitized speech files (header information, data, and label structure). While the solutions adopted in this system are not intended to be considered a standard, the design principles involved are of interest to users and creators of other PCM systems.

The pulse code modulation (PCM) system of digitizing analog waveforms, in which amplitude samples are taken at frequent, regular intervals, can accurately represent continuously varying signals as binary digital numbers (see Goodall, 1947). In the years since its introduction, PCM has become the standard technique for the digital sampling of analog signals for research purposes (in preference to such alternatives as delta modulation or predictive coding of various sorts; see Heute, 1988). PCM systems are now available for almost any computer, and the recording industry's digital CDs have surpassed analog formats in sales.

Although PCM systems are now commonplace, this has not always been the case. When Haskins Laboratories needed an interactive, multichannel system in the mid-1960s, such systems simply were not available. A design was devised, and implemented in an unconventional way, to meet the needs of our researchers. Much of our speech research at that time was concerned with perceptual responses to different words or syllables arriving at the two ears simultaneously or with small temporal offsets. Stimulus tapes for such experiments could be made by tape splicing (a separate tape for each ear) and rerecording the signals onto a dual-track tape, but the method was both error-prone and laborious. Moreover, each change in stimulus condition—different pairings of the overlapping words, differences in relative onset time or in relative level—required doing the whole job over again. Hence, the design objective was to store all the stimulus words in the computer, then convert them back to analog form and bring them out in real time to a listener or, in the usual case, to a dual-track recorder in whatever combination of stimuli, offsets, and levels the experimenter might choose.

The system that resulted is still in use, but its very singularity makes it mostly of historical interest. Certain aspects of that system, however, are incorporated into newer systems based on current, commercially available hardware. These newer systems are in place at Haskins Laboratories and at more than a dozen other sites in the United States and abroad. These will be described in detail so that current and future users of Haskins-based systems can have easy reference to them and so that designers of other systems can see the reasoning that went into the choices made. The basic principles of A/D conversion will be outlined along the way.

EARLY PROBLEMS AND SOLUTIONS

In 1964, when the earliest PCM system at Haskins Laboratories was begun, the challenge for our designers was simply to create a system where none could be bought. Although PCM was common in telephony, there were no commercial systems available for programmable computers. We therefore designed a system to be interfaced with a Honeywell DDP24 computer (and later on a DDP224) with 8K of memory. Although only brief stretches of speech could be digitized directly into core memory, double buffering allowed the system to deal with continuous speech input; that is, the incoming digital stream was stored alternately in one of two buffer areas of memory while earlier samples were being read out from the other buffer and written to digital tape. For output, 2.8 sec of speech could be called up at will directly from core memory. Longer sequences could be compiled onto

The writing of this article was supported by NIH Contract N01-HD-5-2910 to Haskins Laboratories. We thank Michael D'Angelo, Vincent Gulisano, Mark Tiede, Ignatius G. Mattingly, Patrick W. Nye, Tom Carrell, David B. Pisoni, and two anonymous reviewers for helpful comments. We also thank Leonard Szubowicz for the time and care spent designing and implementing the original version of the Haskins PCM software. Correspondence should be addressed to D. H. Whalen, Haskins Laboratories, 270 Crown St., New Haven, CT 06511.

digital tape and then read off from the tape in near-real time. For two-channel synchronized output, the samples stored on the tape alternated between the two speech channels. Later, faster disks became available, so that long, one-channel sequences could be output without going to tape. The same might have happened for the two-channel output, except that technology passed this system by, and it disappeared when the DDP 224 was liquidated for its gold content in 1982.

The next challenge was to meet the growing demands of an increasing research field by adding more channels that could access a set of common disks, avoiding both the recording on digital tape and the limitation to one user at a time. The result was a multichannel PCM system, which was designed by Leonard Szubowicz, Rod McGuire, and E. R. Wiley and implemented with the collaboration of Richard Sharkany. It consists (it is still in use) of four output channels and two input channels, controlling direct memory access (DMA) boards and filled continuously in first in/first out (FIFO) circuits. Memory is dynamically allocated to each active channel; the amount is trimmed back as other requests come in or is expanded as other channels become inactive. The advantage of this memory management is that large memory areas make the rare FIFO shutdown (i.e., data did not arrive in time) even rarer. The advantage of FIFO organization is that buffers can be filled with less concern for time-critical disk accesses. A drawback is that the system does not know exactly where in the output it is, since only the DMA has that information, so that the controlling computer cannot receive an exact reading of how far the sequence has gone.

Although the speech waveform is the primary signal of interest at this laboratory, other analog signals, such as the output of transducers measuring the speech articulators and muscles (electromyographic [EMG] signals), are also used. Many such signals are more restricted in the frequency domain and, thus, can be represented adequately at slower sampling rates. The lower the rate, the less disk space is used. Even for speech, some purposes are well served by the 10-kHz rate, whereas others need the information between 5 and 10 kHz, which is preserved at the 20-kHz rate. Each of the six channels can be used at a 10- or 20-kHz sampling rate. One input channel and one output channel also support the rates of 100, 200, 500, 1,000, 2,000, 4,000, 5,000, 8,000, and 16,000 samples per second. If necessary, these two channels can be connected to an external clock that can be run at any rate up to 50 kHz.

When the system was designed, computer memory was quite limited, so the simplest, memory-intensive solutions to real-time output were not available. To obtain a large throughput from a small system, our design undid the major advance in computation, von Neumann's use of data and instructions in the same area. Given the small address area of our platform, the PDP 11/04, there was very little room to write a program and extremely little left over for data. To overcome this limitation, additional memory was attached, even though the processor could not access it. However, the DMAs could, and the program was capable of telling them how to do so. In this way, an adequate amount of memory was available to sustain a throughput of about 40,000 data samples per second, divided among up to four channels.

A continuing concern was the synchronization of any two PCM channels. This was accomplished by setting any two channels to wait for the same clock. When the clock is started, the two channels begin at exactly the same time. The primary goal of this feature was the easy creation of stimulus tapes for dichotic listening procedures (Cooper & Mattingly, 1969). It also allowed the simultaneous input of two analog channels (e.g., speech and laryngograph). Furthermore, an output and input channel could also be synchronized, allowing for such features as resampling a file with different characteristics (e.g., sampling rate).

Sometimes, it is convenient to have an arbitrary audio signal that marks the passage of a certain portion of the main signal. An example is the use of a tone to start a clock for a timed response from a subject. Although this could be accomplished by having a second, synchronized channel outputting such a "mark tone," the lack of variation in the signal allows for a simpler solution—and one that would allow mark tones to accompany two-channel output. Each output channel is thus associated with a mark-tone channel, which allows the output of an unvarying audio signal (a 1-kHz tone, in this case) without any increase in processing load. Whenever a sample is output that has the second highest bit set, a 1-kHz tone, 4 msec in duration, is simultaneously output on the mark-tone channel. This tone can be recorded along with the main signal, allowing (for example) the synchronization of the main signal with other devices (e.g., a reaction timer). Since the mark tone is essentially part of the data stream, it does not impose any further load on the system: The second highest bit is part of the 16-bit word that is stored in the computer, but not part of the 12 bits of data. Thus, mark tones can be freely intermixed with either or both channels of synchronized output.

While the PCM system just described is still in use at Haskins Laboratories, it is no longer the only system in use there. Input and output (A/D and D/A) boards from Data Translation, Inc., have been added to several VAXstations (from Digital Equipment Corp., or DEC) and made compatible with the file and data formats from the older system. Such features as the file format, the synchronization of channels, and the characteristics of the filters have been maintained. So, while the convenient features of the old system can be included in the new systems, these systems, unlike the original, can be duplicated at other laboratories.

COMPUTER ENVIRONMENTS

The main Haskins PCM system, with its four output channels and two input channels, consists of a PDP 11/04 (DEC) that shares disks with a VAX 11/780 (DEC) via

a Local Area VAX Cluster. These disks contain the computer files that store the digitized samples of the PCM system. The VAX and the 11/04 communicate via two 16-bit parallel programmable I/O interfaces. Control parameters, such as disk addresses and start or stop signals, are passed from the VAX to the 11/04, and status words are passed back to the VAX. When input or output is being performed, the 11/04 has priority on the disks, allowing it the best chance of completing its time-sensitive tasks. For both input and output, the disk files must be contiguous, rather than being spread across several segments as an ordinary file would be. If the file were not contiguous, computing an address for a file extension and repositioning the heads would often take longer than the amount of time used to output the data obtained on the previous disk access.

The newer systems use Data Translation A/D and D/A boards installed in MicroVAXs or VAXstations. In contrast with the older system, the PCM data must pass through the main CPU. This requires the process performing the input or output to be set to real-time priority, but does not automatically exclude other jobs from running on the computer. Having only the PCM job, however, reduces the chance that the data cannot be read off the disk within the time allowed. Also unlike the older system, the new systems support only a single user. And, although there are two output boards on most of the new systems, they both demand the same CPU resources, so only one signal, or two synchronized signals, can be processed at a time.

DYNAMIC RANGE

Dynamic range is the ratio of the maximum to minimum amplitude difference in the signal that can be accurately represented. Thus, the primary limitation on this is the number of bits of resolution used for representing the data. The Haskins PCM format for data consists of 12 bits of digitization, which can represent 4,096 distinct values. These are stored in 2-byte (16-bit) words, with the upper 4 bits (the ones not used for data), containing output control information. Sixteen-bit systems are quite common and form the basis of digital audio systems. Eight-bit systems, which can represent 256 distinct values, are used in many personal computers, but they do not have adequate resolution for many research purposes. The coding itself is simply a binary representation of the quantized voltage. Most systems, including the Haskins one, avoid having a sign bit by adding a dc offset half as large as the dynamic range. For a 12-bit system, this means that the original representations of −10 V to +10 V as −2,048 to 2,047 will be stored machine-internally as values ranging from 0 to 4,095. (Thus, the dynamic range is, more accurately, −10 V to +9.995 V, since one value of the coding scheme must be used for zero, leaving the range one value off center; for the rest of this paper, the value +10 V will be used, even though 9.995 V is meant.) In the Haskins system, each value is represented as a 16-bit number.

With a 12-bit system, the theoretical dynamic range is 72.2 dB. This is calculated from the formula $20 \log 2^n$, where n is the number of bits in the system. Conveniently, this reduces to $6.0206n$. Machine-internal noise effectively reduces this by 1 bit, yielding a more realistic estimate of 66.2 dB. By contrast, a 16-bit system has a theoretical range of 96.3 dB, and an 8-bit system has a theoretical range of 48.2 dB.

When digitizing, the system cannot differentiate between signals that reach the upper or lower quantization limits and those that exceed them and thus fall outside the dynamic range. Any signal that exceeds either of the limits will therefore be truncated to the limiting value, resulting in *peak clipping*. Although the clipping of a single sample will have relatively benign consequences, many successive peak-clipped samples will result in an obnoxious noise and an unreliable frequency analysis of the clipped region. The only remedy for peak clipping is to reinput the signal at a lower level.

Any PCM system has inherent limits on the size of differences in the input voltage that can be represented accurately. Analog values that fall within the range of 1 bit will be given a single digital value. The divergence from the original signal due to these limits is called *quantization error*. Since the voltages of −10 V to +10 V are covered by 12 bits in the Haskins system, the quantization error is 4.88 mV (or 0.0244%) for signals using the entire dynamic range. For low-amplitude sounds using less of the dynamic range, the quantization error will be larger in terms of percent.

TIMING RESOLUTION

The frequency at which the system examines the analog signal and codes it into a digital number is the *sampling rate*. This rate imposes a limit on the frequencies within the original signal that can be accurately represented. If there is an input signal that has a frequency higher than half of the sampling rate, its samples will be indistinguishable from those of a lower frequency signal. This shift in apparent frequency is called *aliasing*, and the frequency above which the effect occurs is called the *Nyquist frequency*.

The sampling rate also imposes limits on the accuracy of frequency measurements for some aspects of the speech signal—formants and, most noticeably, the fundamental frequency (F0). For a file sampled at 10 kHz, an F0 of 100 Hz will be limited in accuracy to ±0.5%. This is usually quite acceptable, but there are times when greater accuracy is desirable. For higher F0s, however, the error due to temporal quantization is much larger. For a typical female F0 of 200 Hz, the accuracy is ±1%; for a high (but not exceptional) child's F0 of 500 Hz, it goes to ±2.5%. All of these figures can be cut in half for files sampled at 20 kHz, but even ±1.3% is variable enough to obscure some effects. The most clear-cut instance in which these differences become important is in the measurement of vocal jitter (e.g., Baken, 1987, pp. 166-188)—that is, the difference in F0 between adjacent pitch

periods. Here, the differences add up, because a half sample excluded from one period will be added into the next, increasing apparent jitter, when there may in fact be none. The cost of higher accuracy, in this case, is the larger storage space required. Doubling the sampling rate doubles the amount of disk storage needed.

Another timing relationship is that between two channels that are started at the same time. For synchronized channels in the Haskins system, whether on input or output, the time difference between the two channels is nonexistent. Both channels read the same clock, and, thus, they both start at exactly the time that the clock starts. When digitizing, there is a minuscule amount of *amplitude* decay for the second channel, since the signals will be read off the sample-and-hold circuits after the 20 μsec it takes for the first channel to perform its coding. However, since the decay for these circuits is measured in seconds and the coding occurs at a delay that is considerably less than half of the sampling rate, the reduction in amplitude is truly negligible. The important fact is that the two channels are triggered at exactly the same time, rather than half a sample apart.

The absolute simultaneity of the two channels has been preserved in our more recent systems based on commercially available boards. The input and output boards from Data Translation, Inc. have two channels available on each, but our system ignores the second channel and uses a second board instead. One consequence is that the two channels are completely simultaneous rather than slightly offset, as they are when the two channels of one board are used. A more practical consequence is that the samples from the two files do not have to be interleaved as they are read into memory. This saves a considerable amount of overhead for the system, allowing a much more flexible approach to the capture and presentation of simultaneous signals. Files of any length can be played together in any combination with no more processing time than for a single file.

FILTER CHARACTERISTICS

Every analog signal that is to be digitized and every conversion of a digital signal into an analog one benefits from the use of filters. Unfiltered digital output can produce severe "digitization noise," due to the sharp edges of the pulses that are produced by the digital samples. On input, frequencies that cannot be accurately represented must be filtered out so that they do not contaminate the signal with aliased sounds (see the end of the previous section). (Even if we are not interested in the nature of the signals above the Nyquist frequency, they must be filtered out to avoid contaminating the spectral content below the Nyquist frequency.) Since the limit is called the Nyquist frequency, the filters are called *Nyquist filters*.

A more specialized filter, which aids in the representation and analysis of high-frequency sounds, is the *high-frequency preemphasis filter*.

In creating a PCM file, the combination of filters to be used is specified in the program, and that combination is stored in the header of the new file. For outputting a PCM file, the program determines the appropriate filters based on information in the file header. Once these are selected, they cannot be changed. Resetting the filters usually results in an audible click, which would be unacceptable in the midst of an output.

Nyquist Filters

The filters that Haskins systems use to eliminate frequencies above the Nyquist frequency are hardware filters with the response shown in Figure 1. Components below 4.8 kHz (or 9.6 kHz for the 20-kHz system) emerge with only minor reduction in amplitude, whereas those above

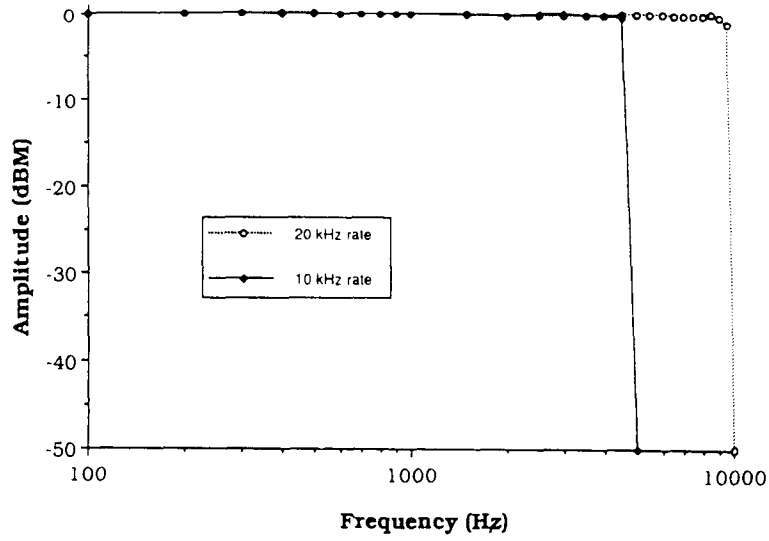

Figure 1. Resultant amplitude of 0-dBm test signals of differing frequencies after passing through the Nyquist filter. Measurements shown are for one system, but similar results obtain for other Haskins systems.

are severely attenuated. At 5 kHz (or 10 kHz), the attenuation is, at a maximum, approximately 50 dB. Most filters are described in terms of the number of decibels per octave that the attenuation attains. Since the attenuation here is accomplished in much less than an octave, it is misleading to describe this cutoff in a decibel/octave formula. Stated in those terms, these filters have a 1,200 db/octave attenuation, which is over 16 times larger than the entire dynamic range. Since it is theoretically impossible to attenuate a signal more than the dynamic range allows, this number is impossibly large. Instead, the filters should be described as sharply tuned and as reaching the 3-dB attenuation level at 4.8 kHz (or 9.6 kHz). In any event, the sounds above the Nyquist frequency have virtually no chance of affecting the signal any more than the background noise does.

High-Frequency Preemphasis Filters

For signals such as speech that are primarily driven by low-frequency sources, the high-frequency components generally have lower amplitude than do the low-frequency ones. Of course, high-frequency signals of a given amplitude, being more intense, will sound louder than low-frequency signals of the same amplitude, so that, in a sense, the high-frequency signals are more perceptually salient than their amplitude would suggest. Nonetheless, early researchers found that the high frequencies, especially of speech, were difficult to measure or even detect when input at their natural level. To rectify this situation, a hardware filter was selected that could boost the high frequencies (before digitization) by a reliable and known amount. A complementary filter could then reduce their amplitudes by the same amount when the digitized signal was played out. There is a slight gain in accuracy of the digitization, since the quantization error will be a smaller proportion for a signal that uses more of the dynamic range. For the /f/ noise presented in Figure 3, for example, the quantization error is about 0.488% for the nonpreemphasized signal, whereas it is about 0.029% for the preemphasized signal. Although this difference is sizable, the improvement in quality may not be very noticeable to the naked ear (however, see Whalen, 1984, for a demonstration of perceptual effects of differences that are not consciously detectable).

Figure 2 shows the preemphasis function used with the 20-kHz sampling rate. The response is fairly linear up to 1 kHz, then rises exponentially, shown as a straight line in Figure 2, where frequency is represented in a log scale. On output, a filter with exactly the reverse characteristics is used. Thus, if the amplitude value is read as a decrement, this figure can be used to represent the deemphasis filter as well. The same filter is actually used for the 10-kHz rate, but since the Nyquist filter (which in this case functions as an antidigitization noise or "antiimaging" filter) follows it, there will be nothing left above 5 kHz.

Ideally, the preemphasis filter should equalize the long-term speech spectrum so that the maximum use of the dynamic range is achieved for each frequency region. Clearly, no one filter shape can serve this function, since different speakers, and even the same speaker at different times, will generate different long-term spectra. The shape of the preemphasis function is a compromise based on the sorts of long-term spectra encountered in the early research. The function is not based on properties of the human auditory system, though it bears a superficial resemblance to the ear's increase in sensitivity between 1500 and 4500 Hz (e.g., Robinson & Dadson, 1956).

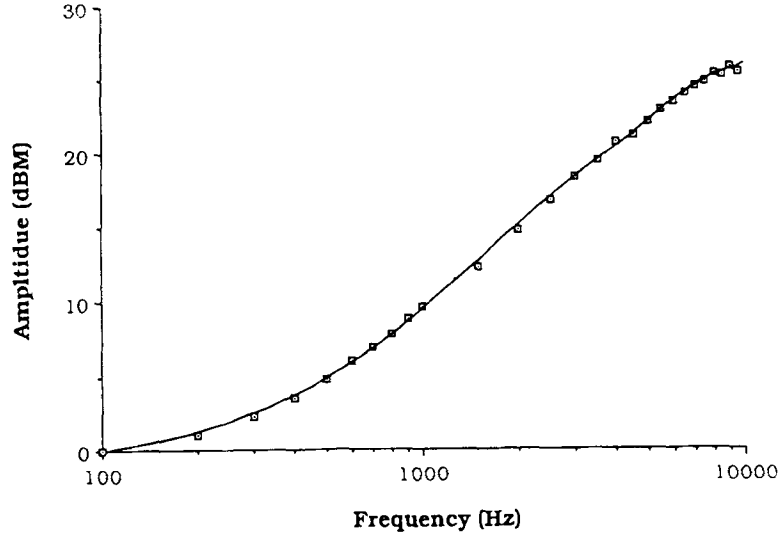

Figure 2. Resultant amplitude of 0-dBm test signals of differing frequencies after passing through the high-frequency preemphasis and Nyquist filters. Symbols represent measurements for one system; the line is a fitted polynomial. Because of the Nyquist filter, the output level drops steeply at 10 kHz (not shown).

There is also some resemblance to the historically later Dolby noise-reduction systems. Dolby systems have become standard in the recording industry, but there are good reasons not to use them as part of a PCM system. Although the Dolby system greatly increases the separation of low-intensity, high-frequency signals from the noise encountered on playback from audio tape, it would be inappropriate to use it as a front end to a digitizer, since digitized signals are not subject to media noise. (Even for signals that are simply recorded on audio tape for later digitization with a PCM system, Dolby noise reduction may be inappropriate. The net effects of the Dolby filters may be benign in terms of intelligibility, but finer acoustic measurements [e.g., the bandwidths of formants that happen to lie at the edge of one of the four Dolby bands] may be affected. In addition, having the tape noise at a constant level makes it easier to take into account when comparing the amplitude of speech sounds. Reducing the tape noise for high-frequency sounds would reduce their amplitude, compared with low-frequency sounds that included the noise.) Similarly, there are digital techniques (e.g., first-differencing) that can have similar effects without requiring the hardware filters. However, such digital filters are neither sharp enough nor linear enough for many of the measurements that are made in the speech field. So, for consistency and reproducibility, the hardware filter approach has the most benefits. This system does have the drawback that the PCM representation of these signals cannot be played back faithfully on other systems unless the other systems have the same filter. (They can be played back without the deemphasis filter, and the speech is usually quite recognizable, just distorted by the additional amplitude in the high frequencies.) For many purposes, such representations are adequate.

Figure 3 shows the effect of this preemphasis filtering system. In the top panel is the waveform of the word *fast*, with the high frequencies preemphasized. The characteristically weak /f/ fricative noise is easy to discern in the first 100 msec. In the bottom panel, exactly the same signal (input synchronously on the second input channel) is shown in its nonpreemphasized version. The onset of the /f/ noise is very difficult to discern at this level of resolution. The middle panel of Figure 3 shows the result of magnifying the display of the bottom panel by a factor of 3. The shape of the fricative noise is now somewhat clearer, though the gradualness of the beginning of the noise is still somewhat hard to make out, but the vocalic segment (/æ/) is now (visually) peak-clipped. Along with the fricative noise, the low-frequency, dc air-flow noise can also be seen. Such information is useful for recognizing less than optimal recordings, but it is not part of the speech signal. With preemphasis, the shape of both the fricative noise and the vocalic segment are evident, and there is no need to use separate magnifications to make

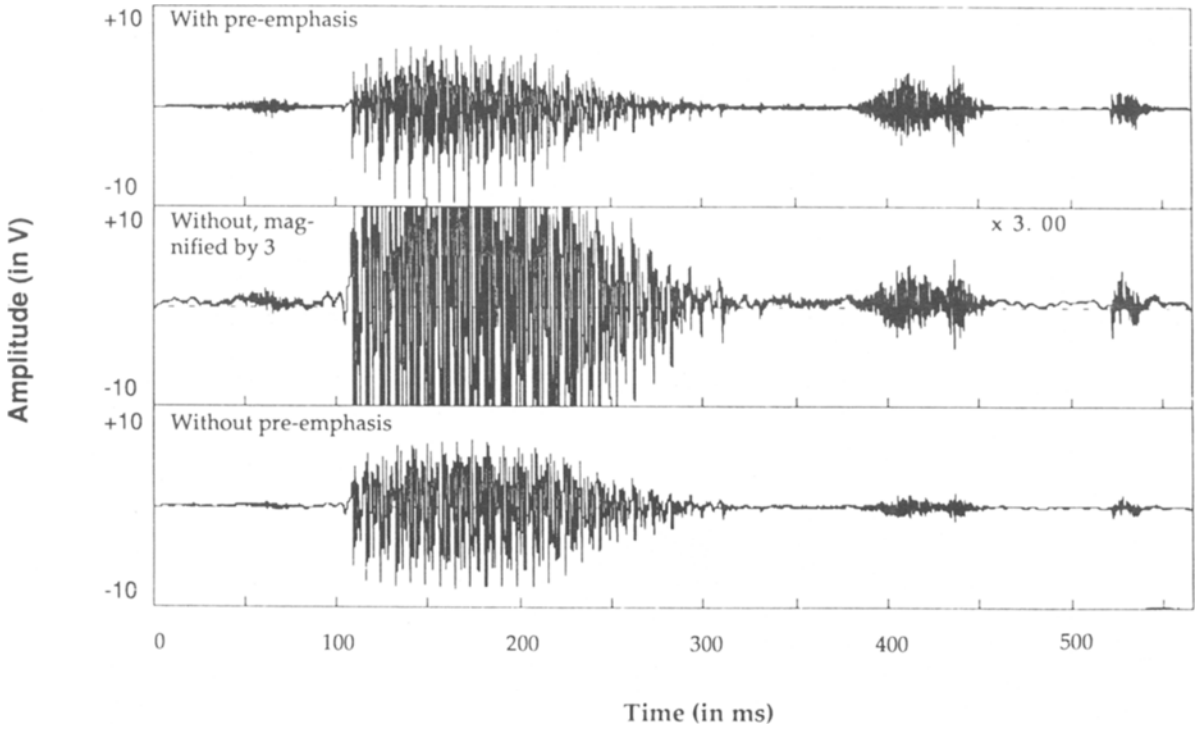

Figure 3. Waveforms of the word *fast* under two sampling and two display conditions. The top and bottom panels represent the syllable with and without preemphasis, respectively, at original amplitude. The middle panel is the nonpreemphasized signal magnified by a factor of 3.

them so. While the /f/ noise could have tolerated much greater preemphasis, the /s/ noise (around 375–450 msec), which also contains high frequencies, could not.

Preemphasis is not without its cost in other regards, however. Although the frequency analysis of the high frequencies is more accurate, the amplitude values of those frequencies relative to low-frequency components are inflated. While the amount of change is predictable, it is not terribly convenient for humans looking at the display to calculate. When many comparisons of, say, the amplitude of F4 with that of F1 are to be made, preemphasis is definitely a drawback. If F5 is in question, however, it may be that the structure of the formant itself is not discernible without the preemphasis, so that the translation of the amplitude is a necessary evil. Such comparisons are relatively rare, however, and most researchers take advantage of the greater resolution in preemphasized digitization.

One other cost deserves mention, since it has already caused a certain amount of confusion in the literature (Fowler, Whalen, & Cooper, 1988; Howell, 1988; Tuller & Fowler, 1981). In Tuller and Fowler's (1981) study, the amplitude of various speech signals was equated without the complete destruction of the speech information by a technique called *infinite peak clipping* (Licklider & Pollack, 1948). For each sample of the signal, positive values are amplified to the maximum level and negative values to the minimum. The result is an irritatingly noisy, though usually recognizable, utterance. If the original file was preemphasized, however, it would normally go through the deemphasis filter. When output through the deemphasis filter, the high frequencies are lowered in amplitude, so that signals with different frequency components would again have different amplitudes, despite the infinite peak clipping. If the deemphasis filter is avoided (which can be done by changing the PCM file header), the intended result is obtained even for preemphasized files. (The preemphasis filter rarely changes the sign of a sample, though it can happen when an intense high-frequency sound occurs with a simple low-frequency sound.)

Another technique, which results in a sound called *signal-correlated noise* (Schroeder, 1968), interacts with the preemphasis function. Signal-correlated noise retains the amplitude contour of the source sound but has a flat spectrum. The samples of approximately half of the digitized source have their signs changed at random, while the magnitude remains the same. The overall energy remains the same, since the same amount of deviation from the baseline is present. But, since the direction the wave takes is randomly related to its original direction, the spectrum of the signal is flat. For a preemphasized original signal, however, the spectrum of the signal-correlated noise is flat only machine-internally. If the noise passes through the deemphasis filter, the high frequencies will fall off by the amount specified in Figure 2. This does not restore any of the spectral structure of the original; however, the spectrum is not perfectly flat. Avoiding the deemphasis filter will not salvage the noise, since that would maintain the flat spectrum but change the amplitude contour. For sounds from which signal-correlated noise stimuli will be created, a nonpreemphasized original is preferable. Alternatively, a brief description of the deviation from a flat spectrum (the high-frequency rolloff) is necessary.

HASKINS PCM FILE FORMATS

The information in this section is quite detailed and will be of interest primarily to users of the Haskins system. The kinds of information included, though, may be of interest to users of other PCM systems. The format of digitized files takes advantage of the special features of the Haskins PCM hardware (such as mark tones) and of in-house programs (such as the labels of the waveform editor WENDY). For third-party software, modifications are required. For example, the ILS package of Signal Technology, Inc. is a large set of programs for doing signal analysis. By default, these programs expect a header format in PCM files that contains some of the same information as Haskins headers but puts them in different locations. The input and output routines have been changed so that ILS can put its information at an otherwise unused part of the header, leaving the rest in the Haskins format. Another alternative that is employed by some newer Haskins programs is to translate from one header format to the other and create two versions of a file if needed.

These features will be discussed in the order in which they appear in the computer file. The first component of the file is a header block of 512 bytes, which contains information about the characteristics of the data. The next is the data itself, taking up as many 512-byte blocks as are needed to accommodate the number of samples in the file. The final, optional, portion is a section of up to four trailer blocks containing labels of locations within the file. (This label format is in the process of being superseded by separate label files.)

The conventions presented here are not intended as a standard (see Mertus, 1989), since there are many concerns that are not adequately addressed by this format. For example, there is currently a word in the header to indicate the number of bits of resolution (always 12 for current Haskins systems), but this format may not be optimal for a more broadly defined standard. The present discussion is intended to make the information more accessible for laboratories that already use the format and to bring the Haskins conventions to the attention of those devising their own systems.

PCM Headers

The initial portion of each PCM file consists of a header that contains attributes of the sampled data within the file. For some files, especially those from the Haskins Physiological Speech Processing (PSP) system, the header also establishes a correspondence between time and sample position within the file. The first file block of the PCM file (512 bytes on DEC systems) is used, though for speech

files much of it is simply zero-filled. Physiological files contain more information (see below).

PCM Data

The PCM data begin in the first block immediately following the header block. Samples are stored as fixed-length 128-byte records of 64 words and are usually input into contiguous files, though the files do not have to be contiguous for analysis programs that do not do real-time output. To output a section of a sampled data file with the older system, it must be contiguous. The newer systems can read noncontiguous files into memory sufficiently fast to keep the real-time output going.

One 12-bit sample is stored in the low-order bits of each 16-bit word. This 12-bit sample represents a bipolar analog voltage that ranges from endpoints set near −10 V and +10 V. The four high-order bits in each 16-bit word form a control field that is utilized by the audio output system. When samples are read for analysis within the computer, this control field must be cleared before subtracting the midline. That is, if one of the control bits is set, it will appear to the general computer as a legitimate part of a number, even though it would be far outside the dynamic range. Normally, these bits should also be cleared when samples are written out to a PCM file. Programs that generate speech files must truncate the samples to avoid overflow into the control field.

The format of the data word is presented in Table 1.

To conform with the conventions used by the A/D and D/A converters at Haskins Laboratories, the signal voltage levels are encoded digitally in excess-2048 form—that is, −10 V is encoded to 0, 0 V is encoded to 2048, and +10 V is encoded to 4095. Thus, a 16-bit bipolar digital value that ranges from −2048 to 2047 can be obtained by subtracting 2048 from the 12-bit encoded sample value.

Haskins PCM Labels

Labels are used to record the position and, optionally, the range of user-defined portions of the PCM file. Each label consists of a string of alphanumerics (beginning, by convention, with a letter) that is a file-unique name for the label, a location (given in milliseconds from the beginning of the file), a left range and a right range (which can be set in terms of milliseconds in relation to the label), and a code to determine whether or not there is a mark tone.

The length of a single label is 32 bytes. The older style maximum number of labels was 64. (In the older style of programs, labels were stored in trailer block[s] of the PCM file immediately following the data blocks within the file.) If there are old-style labels stored in the file, the number is contained in a field in the header block (Word 7). Many of our own programs currently change automatically from old to new style any time a PCM file is accessed.

Table 1
Format of the Data Word

Bit Position	Description
1-12	Data field
13	If set, data field is an interstimulus interval value
14	If set, something is wrong
15	If set, a mark tone will be generated at that sample
16	If set, something is wrong

Table 2
The Old Format for Labels in a Haskins PCM File

Byte Position	Length of Field	Description
1	4	Label left range
5	4	Label right range
9	4	Label location (time value of label)
13	1	Label mark-tone flag
14	19	Name of label

The old format for labels in a Haskins PCM file is presented in Table 2.

The unit for time representation is 1/20,000th of a second. The scope of a label is defined to extend from its time value minus its left range to its time value plus its right range.

The new format consists of separate ASCII files containing label information coded by keywords, many of which are common but some are specific to one program. This allows for greater flexibility in the number of labels that can be maintained, convenient correction or even creation of labels with a text editor, and compact sharing of labels across several related files (such as physiological measurements of one event that might end up in a dozen different files). The implementation of this system is in progress and, eventually, will be the only one used by Haskins programs.

SUMMARY

The Haskins PCM system is a combination of standard techniques and unique features. Copies have been built with custom-made hardware and, more recently, with commercially available boards. Some salient features are (1) convenient input and output of signals of any length (dependent on the system's disk rather than on the PCM system constraints), (2) exactly simultaneous synchronization of two channels (either two output, two input, or an input and an output) without the need for interleaving the samples, (3) consistent preemphasis of high frequencies for easier analysis and converse deemphasis for accurate reproduction, and (4) the capability of having any number of mark tones associated with a file without any added load on the system. This system has been used in generating the data for dozens of papers over the last 20 years and will continue to be used both at Haskins Laboratories itself and at the growing number of laboratories that are using the system.

REFERENCES

BAKEN, R. J. (1987). *Clinical measurement of speech and voice*. Boston: College-Hill.

COOPER, F. S., & MATTINGLY, I. G. (1969). A computer-controlled PCM system for the investigation of dichotic speech perception. *Journal of the Acoustical Society of America*, **46**, S115(A).

FOWLER, C. A., WHALEN, D. H., & COOPER, A. M. (1988). Perceived timing is produced timing: A reply to Howell. *Perception & Psychophysics*, **43**, 94-98.

GOODALL, W. M. (1947). Telephony by pulse code modulation. *Bell System Technical Journal*, **26**, 395-409.

HEUTE, U. (1988). Medium-rate speech coding—Trial of a review. *Speech Communication*, **7**, 125-149.

HOWELL, P. (1988). Prediction of the P-center location from the distribution of energy in the amplitude envelope: I. *Perception & Psychophysics*, **43**, 90-93.

LICKLIDER, J. C. R., & POLLACK, I. (1948). Effects of differentiation, integration and infinite peak clipping upon the intelligibility of speech. *Journal of the Acoustical Society of America*, **25**, 375-388.

MERTUS, J. (1989). Standards for PCM files. *Behavior Research Methods, Instruments, & Computers*, **21**, 126-129.

ROBINSON, D. W., & DADSON, R. S. (1956). A redetermination of the equal-loudness relations for pure tones. *British Journal of Applied Physics*, **7**, 166-181.

SCHROEDER, M. R. (1968). Reference signal for signal quality studies. *Journal of the Acoustical Society of America*, **44**, 1735-1736.

TULLER, B., & FOWLER, C. A. (1981). The contribution of amplitude to the perception of isochrony. *Haskins Laboratories Status Report on Speech Research*, **SR65**, 245-250). New Haven, CT: Haskins Laboratories.

WHALEN, D. H. (1984). Subcategorical phonetic mismatches slow phonetic judgments. *Perception & Psychophysics*, **35**, 49-64.

APPENDIX
Information Stored in the Haskins PCM File Headers

Start Position	Number of Words	Description
The seven main header entries occupy the first eight words of the header block. They are:		
1	1	Data Type Indicator: A 1 in this field indicates a sampled data format file that will be recognized as such by Haskins software.
2	2	Sampled Data Size: Double precision integer representation of the size of the file (number of samples). The first word is the low-order part of the count.
4	1	Sampling Rate: Expressed as samples taken per second.
5	1	Attributes: Format of word: Bit 0 is the preemphasis flag. If 0, the data were preemphasized during sampling (the level of higher frequencies was boosted) and should be deemphasized when output. If 1, the data were not preemphasized. Bit 1 is the filtering flag. If 0, the data were filtered during sampling at the Nyquist frequency. If 1, the data were not Nyquist filtered. The remainder of the words (14 bits) are unused.
6	1	Number of Additional Header Blocks: No longer implemented.
7	1	Number of Labels: If greater than zero, then the file contains labels that are stored in the trailer blocks of the file. Each label is 32 bytes long.

Appendix (Continued)

Start Position	Number of Words	Description
The remaining 249 words of the header block code the following:		
8	1	Revision Level: Indicates which version of the arrangement of information in the header is used.
9	2	Virtual Block Number of First Trailer Block: Where old-style labels are kept.
11	1	Number of Trailer Blocks: For old-style labels.
12	1	Data Source: Currently either VAX (1) or unknown (0).
13	1	Number of Bits of Resolution: Only "12" is implemented.
14	1	Source: No longer implemented.
15	50	Filler Words.
PSP (Physiological Signal Processing) information:		
65	1	Datel Hardware Input Mode: 0 = EMG data, which is already filtered and integrated; 1 = speech, which must be at 10 kHz to be synchronized with physiological measurments; 2 = LED (usually movement) data, in which the x and y values each take up a channel; 3 = electropalatagraph data, where each word represents the on/off state of the 63 contact points in the false palate.
66	5	Filler Words.
71	1	PSP Header Version Number.
72	1	Samples per Frame.
73	1	Channel Map: A 16-bit word that serves as a bitmap representation of which of the 16 possible input channels are actually being used.
74	1	Data File Record Size.
75	2	M Calibration Constant: Together with the B constant, this allows the machine units in the file to be interpreted as physical units. The physical value = M*(sample value) + B. So, M is a scaling factor and B is an offset.
77	2	B Calibration Constant.
79	6	Calibration Units (12 characters): A description of the units that result from the application of the calibration constants (e.g., "millimeters").
85	16	Index File Name (32 characters): Name of a file that contains a catalog of the number of samples associated with each octal code (a time marker on the analog tape) for all of the other PCM files that were created in the same input pass as this one. This information allows for the compen-

Appendix (Continued)

Start Position	Number of Words	Description
		sation of minor speed changes in the analog tape system.
101	2	Smoothing Constant: If the file was smoothed (as is usual for EMG signals), this is the size (in milliseconds) of the base of the triangular averaging filter.
103	2	Line-Up Point: Location of an event chosen by the experimenter to coordinate the displays across PCM files. If PSP header version number = 0, then the line-up point is in samples. If PSP header version number > 0, then the line-up point is in 1/20,000th of a second.
105	2	Graphics Scaling: *Y* min.
107	2	Graphics Scaling: *Y* max.
109	20	Filler Words.
129	128	Filler Words.

Note that the last set of filler words of the header may contain the ILS header information if the file has been analyzed with the Haskins-modified version of ILS.

(Manuscript received April 16, 1990;
revision accepted for publication October 30, 1990.)