15 Years Ago In Byte

The early days of voice synthesis

In an overview article on voice synthesis, authors Kathryn Fons and Tim Gargagliano said *the man-to-machine interface will be one of the biggest challenges facing the industry in the 1980s. It's still a challenge. Several editors have contracted repetitive stress injuries.*

Speaker-independent, unconstrained, continuous voice dictation continues to elude the desktop, though the latest voice-dictation systems approach that holy grail.

And here's a look at those early days of voice synthesis (Features, Feb. 1981, page 164). You decide: Have we come a long way, Baby?.....or not:

Articulate Automata: An Overview of Voice Synthesis -- The area of computer technology that stands to gain most from speech synthesis is the man-to-machine interface

by Kathryn Fons and Tim Gargagliano

The time has arrived for computers to begin speaking for themselves! We discussed some basic techniques for using the TRS-80 Voice Synthesizer in the October 1979 BYTE ("The TRS-80 Speaks," page 113). Response from readers showed many were interested in a more detailed look at voice synthesis. The information presented here is concerned with the basic theory of voice synthesis and the basic procedures involved in constructing a vocabulary. The type of synthesis we focus on is *electronic phoneme synthesis*. A *phoneme* is a basic unit of sound from which speech can be constructed.

Voice-Synthesis Technology

During the past two decades, almost every aspect of computer technology has progressed through several generations of advancement. A relatively recent addition to this list is speech synthesis. The area of computer technology which would seem to gain most from speech synthesis is the man-to-machine interface. This is an area which remains in need of a great deal of development. Today, computers play a role in almost everyone's life, yet we rely on a group of specialists to control the computers. If computer technology is to continue to advance, there will be a strong need for the inexperienced user to communicate directly with the computer. It seems obvious that the man-to-machine interface will be one of the biggest challenges facing this industry in the 1980s.

Another problem confronting computer users is visual confusion and/or saturation. This can occur after watching a video monitor or scanning a printout for hours at a time. Part of this problem can be eliminated by including a nonvisual output channel in the computer system. The obvious choice is voice, since most people normally communicate verbally. In a number of situations, the serial nature of voice output is more desirable than parallel data from a printout or video screen.

A number of applications are already using voice synthesis. Among these are telephone order-entry systems, telephone access systems, reading machines and terminals for the blind, communicators for the verbally

impaired, and computerized dispatching.

Physiology of Speech

The production of speech in the human vocal system begins with a source of acoustical excitation to drive the vocal tract. There are two kinds of excitation: periodic and random. The first type of excitation is a pulse train caused by the vocal folds blowing apart and collapsing under lung pressure (see figure 1). The pulse train is rich in harmonic content due to its sharp wave shape. The second type of excitation is noise (*frication*) caused by air passing over the *articulators* (tongue, cheeks, lips, teeth, etc.) with the vocal folds open.

Phonemes containing periodic excitation are called *voiced phonemes* (e.g., the vowel */a/*). Phonemes containing only frication are said to be *unvoiced* (e.g., the consonant */f/*). It is also possible for a voiced phoneme to contain frication (e.g., the consonant */z/*).

The human vocal tract is formed from resonant cavities including the mouth and nasal cavities which respond to input excitation by filterting the input. At any given time, placement of the articulators determines the frequency response of the vocal tract. Generating speech from the input excitation involves sequentially varying the frequency response of the resonant cavities in the vocal tract. This is done by movement of the articulators. The vocal tract is a fairly complex time-variant filter network.

Speech is composed of several bands of frequencies called *formants* (see figure 2). Each formant varies in postion, amplitude, and quality with respect to time. A static sound, such as a continuous vowel, is produced by moving the air through the vocal tract and over the articulators, which are appropriately positioned to create that sound. During the production of a word, the articulators are constantly moving from one phoneme position to another. This sequencing of the articulator movements is one reason why each sound in the sequence influences every other sound around it. Note that the change in articulator positions does not occur in a single-step fashion, but rather in a continuous movement from one target position toward another. The frequency response of the vocal tract is in flux between the target of the last phoneme and the current phoneme. The acoustical changes that occur during the transition are referred to as *dynamic articulations*. They are important to the production of intelligible speech -- human or synthetic. Without dynamic articulation, speech becomes choppy and often unintelligible.

The Electronic Equivalent of the Vocal Tract

An electronic analog of the human vocal tract can be constructed using filters, oscillators, and noise-source modules (see photo 2). Control of these modules is complicated, and requires measuring the static and dynamic parameters of human speech.

The study of speech parameters requires some complex instruments. Speech is most frequently considered in terms of frequency composition, rather than waveforms measured as a function of time. Therefore, analysis of speech is typically carried out in the frequency domain. This requires instruments that are able to measure and plot frequency, amplitude, and time in various relationships. A spectrum-analyzer scope can display a picture of amplitude versus frequency for an instant in time (see photo 3). This provides accurate measurement of energy distribution among the frequencies of a static sound.

Another type of spectrum analyzer used in the study of speech is a voiceprint machine. This device provides a picture of amplitude versus frequency versus time which is collapsed into two dimensions (see photo 4). This type of printout allows us to study the dynamic characteristics of speech, such as phoneme duration and dynamic articulations. Notice how the frequencies continuously move during the transition from one phoneme to the next.

With these instruments, measurements can be made of the center frequencies of formants, their amplitudes, and their bandwidth. These measurements are the basis for designing the filter networks used in an electronic vocal tract. A model of a voice synthesizer in its simplest form is shown in figure 3. Depending on the desired speech quality, a varying number of parameters must be controlled. The number of bits stored for each parameter depends on the needed range and quantization tolerance of each parameter. To control this type of synthesizer, parametric data must be updated every 5 to 25 ms. The update frequency must be high enough to capture the parametric movements during phoneme transitions. While this synthesizer model can provide much flexibility, it does so at the expense of a high bit-rate/storage requirement and complex vocabulary generation.

The Votrax Phoneme Synthesizer

A phoneme synthesizer can be modeled by adding a parametric control generator and a dynamic-articulation control unit. A model for a Votrax phoneme synthesizer with several options is shown in figure 4. Rather than have the user update all the parameters of a phoneme several times during its production, the synthesizer automatically does it using an interal algorithm. Because the Votrax phoneme synthesizer is implemented totally in hardware, there is no requirement for an external computer/memory to generate phonemes.

A high-quality phoneme synthesizer (with many internal parameters) is no more complex for the user to control than a minimal unit because both utilize the same phoneme call-out procedure. A command word is used to signal phoneme production. The command word for a phoneme includes phoneme-select data and optional pitch, rate, and amplitude data. Typically, there are sixty-four phonemes produced, each requiring a 6-bit command word.

A simple digital controller or microcomputer is all that is needed for vocabulary retrieval. In the phoneme synthesizer we have modeled here, the duration of each phoneme is controlled by an internal timer. At the end of an interval, the timer output momentarily goes low, requesting the interface to send the next phoneme command word. This phoneme request signal can be used to generate an interrupt request to a microprocessor or clock a command word out of a FIFO (first-in/first-out) bugger, an interface, or ROM (read-only memory). See figure 5.

Several types of Votrax synthesizers are available. A recent addition to this family is the SC01, the first *single-chip* phoneme synthesizer; it represents a significant breakthrough in speech-synthesis technology. Contained in a 22-pin dual-inline package, this low-power CMOS (complementary metal-oxide semiconductor) synthesizer can be easily used on a printed-circuit board. Latched parallel inputs permit direct connection to a microcomputer data bus. A master clock input on the SC01 permits a variety of voice effects and highly textured sound effects to be generated.

Phonetic Programming

There are a few specific speech rules that dictate how phonemes are sequenced for intelligible speech output. Pronunciation guidelines and symbols, established by the IPA (International Phonetic Association), are often used to identify the phonemes and the altered or adapted units of sound (called *allophones*). These are used because the standard alphabetic characters may have more than one sound associated with a single symbol. Using phonetic guidelines, phonemes and/or allophones are combined to form the symbol sequence that represents the spoken word in a language. The written symbology, however, does not always directly translate into the sounds available in a phoneme synthesizer. Thus, a sequence of the synthetic phonemes constructed from the phonetic guidelines might produce an awkward, if not unintelligible, pronunciation of the word being translated. The pronunciation guidelines from any phonetic symbol system (IPA, *Webster's Dictionary, Thorndike's Dictionary*) can be used to establish a *basic* synthesized phoneme sequence, but *listening* is the final step used to determine the selections for a refined phoneme sequence (see the table

"Programming Phoneme Voice Synthesizers").

For the purposes of this article, all phonetic sequences are presented utilizing the Votrax Phonetic Symbol System. This system is used because it utilizes characters that are found on a standard computer terminal, as well as those needed for translation.

Phonemes

The sixty-four synthetic phonemes produced by a Votrax speech synthesizer are used here as the base synthetic-phoneme reference. The phonetic symbols representing these sounds and example words are listed in table 1. There are twenty-five different consonant sounds, thirty-six basic vowel and vowel-allophone sounds, and two pause phonemes. The sixty-fourth phoneme is called a *zero-decode command phoneme*. It emits no sound, but can be used a s a short interruption. When you select the appropriate synthetic sounds and place them in a specific sequence, the speech synthesizer can produce any word in the English language (as well as many other languages).

Vocabulary Storage

Vocabulary storage requirements are dependent on the number of words in the vocabulary and the number of bits in a phoneme-command word. For example, a vocabulary of 100 words using a 6- to 8-bit command word to represent each phoneme will require 600 bytes of storage. A 1000-word vocabulary will require 6000 bytes of storage. A 12-bit command word will require 900 to 1200 bytes for a 100-word vocabulary and 9000 to 12,000 bytes for a 1000-word vocabulary (depending on the packing techniques).

When using a phoneme synthesizer with a 6-bit command word and a high-level computer language that allows literal strings to be assigned to a variable, vocabulary storage can be embedded within the program statements by using ASCII strings. This is because a 6-bit command word has only sixty-four possible commands, where there are at least 64 printable ASCII characters. A word or phrase is assigned to a string variable immediately before being sent to a speech-output routine. This routine pulls characters out of the string variable one at a time and sends them to the synthesizer. This technique is suitable for small vocabulary requirements. With large vocabularies, there tends to be word duplication because the storage unit is a sentence or phrase.

A technique better suited for handling large word bases is the assignment of the phoneme string for a single word to a subscripted string variable. This avoids the word duplication experiences by the previous technique and saves memory (provided that the language stores character strings with no wasted space). To generate a sentence using this technique, a sequence of variable subscript numbers is passed to a routine which calls up the indicated variables. Phoneme strings are then removed from the variable and sent to the synthesizer.

For permanent vocabularies stored in ROM (read-only memory) or loaded into programmable memory from a disk file, a word-address look-up scheme works well. This is done by generating a table of words stored sequentially in a portion of the memory. You then produce a look-up table whose entries point to a word in the word-storage table. The number of the look-up-table-entry corresponds to the number assigned to the word (e.g., the fifth entry in the look-up table will point to the fifth word in the word table). These tables can be generated easily (see listing 1). Sentences are called out in the same fashion as the previous scheme.

The assembler scheme works well with any size phoneme-command word, since it does not care how many bits are used to represent a phoneme. However, the driver program must know whether to pull 1, 1-1/2, or 2 bytes per phoneme. Listing 2 shows a driver program in BASIC to access the vocabulary in listing 1. Note that the end of a word is detected by the starting address of the adjacent word in the table.

Applications

In the field of computer technology alone, there is tremendous potential for the use of speech output. Through voice synthesis, applications can expand into areas formerly closed. There are areas where a person must interact with a computer, but where visual output is inappropriate, unavailable, or ineffective.

Currently, a blind person who wishes to use a computer must rely on a sighted person to relay information from a video display or printer. To eliminate this dependency, a terminal for the blind can be built to incorporate voice synthesis. Several such terminals are beginning to appear on the market.

Another situation where speech output is desirable is a warehousing/dispatching system. It is not often cost-effective to place terminals around a large warehouse to list pending tasks. A better method is speech output from a computer connected to a radio link, which dispatches a worker carrying a pocket receiver/transmitter. Similar systems are in use or being developed today.

Another area where computers are presently ineffective is in interfacing with the nonreading population. Such is the case when the users are preschool children or nonreading adults. They are the prime candidates for using CAI (computer-aided instruction) as a supplement to their education. Applications such as computerized testing and evaluation of children would invite advancements in the educational field if a speech-output channel was used.

Synthetic speech applications are not limited to merely the computer peripherals mentioned. When used with a small, dedicated microcomputer or digital controller, a stand-alone device can be produced. Such is the case with a reading machine for the blind.

A second type of stand-alone speech system is a communicator for the verbally impaired. A battery-operated microcomputer system and a speech synthesizer can provide a voice for individuals stricken with neurological or physical disorders which impair the human speech mechanism (see photos 5 and 6).

Other applications for voice synthesis are in the area of entertainment electronics. Talking card games, chess games, and video games are beginning to use voice synthesis. Many of these applications are made possible by LSI (large-scale integration) circuits such as the Votrax SC01 single-chip voice synthesizer.

The interface of man-to-machine will provide a challenge for the 1980s. Speech synthesis will play an important role in the future of computer technology.

Editors' Note: *One of the first voice-synthesis products for consumers was Texas Instruments Speak & Spell, which uses a ten-stage lattice filter to simulate the human vocal tract. In the fall of 1980, as part of the continuing trend toward integrating voice synthesis into everyday products, MB Electronics (a subsidiary of Milton-Bradley) introduced an electronic game called "Milton." The game is controlled by a Texas Instruments TMS-1000-series 4-bit microprocessor and utilizes a custom voice-synthesis integrated circuit designed by MB engineers....***SM**

Programming Phoneme Voice Synthesizers

```
There are a number of steps involved in programming a voice synthesizer.
Initially, you will probably have to frequently refer to Table 1, which
lists symbols and example words which represent sounds.

-- Select the words to be programmed.
```

-- **Speak** the words **out loud.**

-- **Select** the appropriate phonetic symbols to represent the sounds in the words. The number of pohnetic symbols you use should equal the number of sounds counted when the words are spoken.

-- **Enter** the phoneme sequence into the synthesizer and listen to the speech output. Check the synthesizer's pronunciation for the appropriate duration (see Table 5) of each syllable and rhythm of each word. The accent (or stress) placed on each word or syllable will help define the duration parameter.

-- **Select** the longer-duration vowel phoneme for the accented syllable and the shorter-duration vowel phoneme for the unaccented syllable. Reenter the program and listen to it again.

-- **Adjust** the program as many times as needed to achieve the desired pronunciation. This can be done by selecting different vowel-phoneme durations for the stressed vowel so that the durational relationship between the syllables sounds correct (see table 3). You can also adjust the sound by inserting a transition allophone between main vowels and consonants to achieve smooth pronunciation (see table 2 and table 3).

A few examples are:

WORD	INITIAL PROGRAM	REFINED PROGRAM
move	M-U-V	M-U1-U1-V
family	F-AE-M-L-E1	F-AE1-EH3-M-L-Y
harvest	H-AH-R-V-I1-S-T	H-AH1-UH3-R-V-I3-S-T

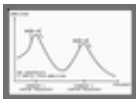
Figure 1

illustration_link (26 Kbytes) 

Periodic excitation of the human vocal tract starts with the vocal folds repeatedly opening and closing (1a), regulating air flow from the lungs.

This results is a pulse train of air (1b) which passes through the resonant cavities of the mouth and nasal passages.

Figure 2

illustration_link (16 Kbytes) 

*Speech is composed of several bands of frequencies known as **formants**. Shown is a generalized formant*

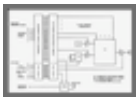
envelope for the first two formants.

Figure 3

illustration_link (32 Kbytes) 

A parametric speech synthesizer. The number of bits stored for each parameter depends on the needed range and the quantization tolerance of each parameter. In order to control this type of synthesizer, parametric data must be updated every 5 to 25 ms.

Figure 4

illustration_link (49 Kbytes) 

A basic Votrax voice synthesizer. A phoneme command word is presented to the unit on the positive edge of the phoneme-request signal. The parametric control generator greatly reduces the synthesizer data consumption by calling out N parameters from only 6 bits. The dynamic-articulation controller generates continuous parametric transitions at phoneme boundaries.

Figure 5

illustration_link (63 Kbytes) 

Interface characteristics. A new phoneme is sent on the positive edge of the phoneme-request signal (5a). A FIFO (first-in/first-out) shift register (5b) provides an elastic buffer by shifting data in at a rate independent from the data being shifted out. Phoneme-request (5c) sets a flip-flop which generates an interrupt request (IRQ) to the microcomputer. When the computer writes the next phoneme command into the latch, the flip-flop is reset.

Table 1

illustration_link (152 Kbytes) 

Phoneme-conversion table. Shown are the Votrax and IPA (International Phonetic Alphabet) phonetic symbols and example words that show the pronunciation of each sound.

Table 2

illustration_link (35 Kbytes) 

Vowel phonemes are categorized here according to their place of production within the human vocal tract. Durational vowel allophones have a number following their symbol which indicates their durational relationship to the base vowel. (the suffix 1 indicates the longest duration; 3 indicates the shortest duration.) The Votrax phonetic symbols are used here.

Table 3

illustration_link (20 Kbytes) 

*Consonant phonemes are listed here according to their voicing quality and grouped according to the manner in which they are produced. Not that **all** vowels are classified as voiced phonemes.*

Table 4

illustration_link (72 Kbytes) 

Since a number of phonetic sequences consistently produce intelligible speech, they can be classified as phonetic pattern rules. The most consistent patterns are shown here. Other phonetic patterns are more flexible, and many specific "sound effects" can be created through experimentation.

Table 5

illustration_link (69 Kbytes) 

In voice synthesis, it is often desirable to lengthen or shorten a vowel or consonant sound at the end of a syllable, word, phrase, or sentence. Shown here are several of the most common "tricks" for creating such effects.

Listing 1

illustration_link (139 Kbytes) 

An example assembly-language program designed to store a permanent vocabulary for voice synthesis in a read-only memory. The program generates a table of words which the user has entered and stores them

sequentially in memory. It then produces a look-up table with entries that point to the corresponding word in the word-storage table.

Listing 2

illustration_link (34 Kbytes) 

A driver program in BASIC which accesses the vocabulary as stored by the program shown in Listing 1. The end of a word is detected by the starting address of the adjacent word in the table.

Photo 1

photo_link (185 Kbytes) 

A selection of voice synthesizers. ***Top left:*** *Votrax ML-1 multilingual synthesizer.* ***Bottom left:*** *phonetic keyboard for controlling a synthesizer without the use of a computer.* ***Right top to bottom:*** *Radio Shack TRS-80 Voice Synthesizer, Votrax VS6 synthesizer, Votrax VSK single-board voice synthesizer. Not shown: Votrax SC01 single-chip voice synthesizer.*

Photo 2

photo_link (224 Kbytes) 

An electronic analog of the human vocal tract using filters, oscillators, and noise-source modules. Control of these circuits requires an understanding of the static and dynamic parameters of human speech.

Photo 3

photo_link (154 Kbytes) 

A spectrum analyzer display of a static phoneme. The X axis is frequency; the Y axis is amplitude.

Photo 4

photo_link (91 Kbytes) 

A voiceprint of the message "hello readers." The X axis is time; the Y axis is frequency. Amplitude is displayed as a function of print density.

Photos 5 and 6

photo_link (221 Kbytes) 

Photo 5 (top): *A communicator for the verbally impaired. The Phonic Mirror HandiVoice HC-110 is a battery-operated speech synthesizer controlled by a microprocessor. The user can select from its 500 word/phrase vocabulary by touching the keypad.*

Photo 6 (bottom): *The Phonic Mirror HandiVoice HC-120 is an advanced version of the voice synthesizer shown in photo 5. It has a 1000 word/phrase vocabulary selected by entering a 3-digit numeric code. Paralyzed users can operate the unit through the use of a paddle switch and a scroll mode.*

Photo 7

photo_link (134 Kbytes) 

Talking typewriters for use by the verbally impaired. The units, which use phonemes, have a virtually unlimited vocabulary.

The authors are both employed by the Votrax Division of Federal Screw Works in Michigan. Kathryn Fons is a speech scientist; Tim Gargagliano is a computer engineer. both have done extensive research in language-processing systems and have worked on the Votrax text-to-speech algorithm. They have a special interest in voice synthesizers in relation to the needs of the handicapped.
