Communication aids for the vocally handicapped using voice synthesis technology, an LCD text display and a single-chip microcomputer

by

Clas I. Rolander

A Thesis Submitted to the

Graduate Faculty in Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE

Major: Biomedical Engineering

Iowa State University
Ames, Iowa

1984

TABLE OF CONTENTS

LIST OF FIGURES

DEDICATION

To Marlene

INTRODUCTION

In the United States alone, the number of people needing communication aids because of vocal disabilities exceeds 1 million, implying that the number world-wide may be ten times as high. The causes include disabilities from birth as well as impairment from head injuries and degenerative diseases. Communication problems addressed in this study are those that result from loss of muscular control over the speech mechanism, not the loss of speech caused by brain dysfunction. Individuals with cerebral palsy are the largest, most visible non-vocal group. Multiple sclerosis and amyotrophic lateral sclerosis are among the most common etiologies developed at an adult age.

Worldwide there are a few dozen companies manufacturing some 50 different types of non-vocal communication devices, most of them aimed for the severely motor-handicapped. For less severely handicapped patients that still have sufficient dexterity left to operate a keyboard, the number of available devices is surprisingly small, only a fraction of the total number of products, despite the fact that such an instrument requires much less complicated user interface.

Almost all existing communication aids suffer from two major limitations: an extremely low communication rate and a very high price in relation to its level of sophistication. Single-switch devices typically show a speech rate of one word per minute, and even for products aimed at patients with most motorical functions intact, communication rates exceeding five words per minute are uncommon. Even though the devices themselves cannot carry all the blame for slow communication, since the rate depends on the user's skill, the rates that are achieved are still in sharp contrast to normal speech rates, which range from 100 to 200 words per minute.

The devices are not only inefficient but also expensive. The rapidly decreasing cost of electronics, lowering the prices for instance of advanced personal computers, is little reflected in the expenses for electronically based handicap aids. Furthermore, the latest "state-of-the-art" in electronics may not be as much exploited in such aids as it could be. Communication systems on the market that are somewhat sophisticated cost from about $1,000 to over $10,000. A comment by Dr. Yale Jay Lubkin, President of Franklin Industries, Ltd. (which does research and development in speech technology), about a device that speaks prestored messages in a fairly intelligible voice, expresses the limitation the high prices impose:

> "The difference such a device makes to an intelligent, but voiceless person, can only be imagined by those who are more fortunate. The HC 110 cost $2,695." (30)

This study was done to gain some insight into the technology (including voice recognition) underlying communication aids for the vocally disabled, and to present a fairly inexpensive demonstration unit, named the "Communicaider," that speaks prestored short messages selected from different sets of vocabularies or writes them on a small text-display.

LITERATURE AND MARKET REVIEW

The areas of microcomputers, speech technology and display devices are undergoing rapid development, which will enhance applications like handicap aids for the vocally impaired. This chapter will give an update on the current state of technology in these fields in addition to an explanation of some of the underlying theory and methods, especially for speech technology. References to some existing products are made and an overview of applications for handicapped is included.

Single-chip Microcomputers

Only a few years ago, several different types of integrated circuits were needed to assemble even a small-sized computer. Beside the processor chip itself, components for functions like clock signal generation, timing, counting and interrupt handling as well as program and data storage had to be present on one or several PC-boards. These requirements increased computer size and costs, created design and layout problems, raised the electrical power consumption and reduced the reliability of the systems.

Today, it is possible to integrate most of these functions on the same chip. By the manufacturing of these single-chip computers in low-power NMOS or CMOS technology, battery operation is made possible, which is a key criterion for portable instrumentation, especially in the medical field.

There are now more than a dozen manufacturers of single-chip microcomputers and many more vendors that use them in different applications (Nicholson (42)). The main difference between various types of single-chip computers is the size of the built-in Read Only Memory (ROM) and Read Write Memory (RWM, commonly called RAM, Random Access Memory) areas. A few manufacturers have also put analog interface circuitry like A/D and D/A converters on the chips, thus making them extremely useful in control applications.

The ROM-area of the chip is used for the customer's program, which is normally stored there at the time of chip manufacturing. However, a few single-chip microcomputers, like the INS-8073 from National Semiconductor and the Z8671 from Zilog, are offered with the ROM-area populated with a BASIC interpreter (Zilog (70),(71)). This necessitates external memory for program storage, but eliminates the need for assembler programming and a development system. Program development for these two computers is easily made in the high-level language BASIC. A drawback is the somewhat limited instruction set of this on-chip BASIC, but the instructions available are normally adequate for relatively simple applications like dedicated control. Furthermore, as with all interpreters, the execution speed of the user's program is also fairly slow. However, for most biomedical applications with not too rapidly varying parameters, the speed of these computers should not be a limitation. Of these two computers, the Z8 offers more hardware features, while the 8073 provides a better BASIC. Appendix B includes a brief description and comparison of the Z8671 and INS-8073, and lists some 20 interesting single-board computers built around these chips.

Despite the complexity of single-chip microcomputers, their price is commonly well below $50 apiece (1984), and future dramatic reductions in price can be anticipated.

Voice Technology

Voice technology is defined as the equipment and methods needed for one or two-way verbal communication with machines. It is divided into the areas of voice synthesis and voice recognition/verification. These dicsiplines, however, have much in common concerning the mathematical models of human speech generation and interpretation. In addition to the underlying physiology of speech and hearing, these different abstract models and their mechanical and electronic implementations have been thoroughly presented by Carter (6), Kuecken (26) and Witten (68) as well as, in a shorter form, by Lerner (27), Schalk et al. (50) and Flanagan (16).

History of voice technology

The first successful and well-documented attempts to produce human speech in an artificial way were made in Russia in the 1770s by Kristian Gottlieb Kratzenstein. He developed a set of acoustic resonators that synthesized the vowel sounds. A few years later, Viennese scientist Wolfgang von Kempelen built and demonstrated an elaborate "talking" machine that generated connected utterances. It used a bellows to provide air to a reed, which in turn excited a single hand-varied resonator that produced voice sounds. Consonants, including nasal sounds, were simulated by four constricted passages controlled by the fingers of the other hand.

A century later Alexander Graham Bell and his brother Melville, impressed by the work of von Kempelen, decided to develop their own speaking device. They constructed a model of the vocal tract, which parts where actuated by keyboard controlled levers. The device could reportedly say vowels and nasals and could even be manipulated to produce a few simple utterances.

In the late 1930s, Bell Laboratories scientist Homer Dudley developed one of the first electrical synthesizers to produce connected speech. Called the Voice Operation Demonstrator (VODER), the device comprised electrical networks selected by keyboard strokes. Resonance produced by the VODER resembled those of individual sounds, and although it took operators a year to become experts at duplicating speech through the machine, it is reported that the VODER produced intelligible speech (Schalk et al. (50)).

Market projections

The market for voice technology products of today is growing very rapidly. In an optimistic estimation done by International Resource Development, a market-research organization, the market volume is predicted to exceed $4 billion by 1992 (Lubkin (31)). The growth is, however, much dependent on the fate of the voice-activated typewriter, which IBM apparently does not have under development as previously thought. Taking this fact under consideration, Venture Development Corp. in Wellesley, Mass., another market organization, expects the value of sold voice synthesis technology equipment to exceed 200 million dollars by 1986, which still corresponds to an annual increase of over 60%.

Similarly, the market projection for voice recognition equipment is 260 million dollars by 1986.

Board-level products (compared to stand-alone devices) will constitute a more solid long-term market for both speech synthesis and recognition equipment, according to a report from Strategic Inc., San Jose, Calif. That report also cites high cost and low quality/reliability as factors retarding the potential market.

To the author's knowledge, today more than 100 different American and Western European companies compete in the speech technology market. Some applications on speech synthesis can be seen in Figure 1 (note especially handicap aids).

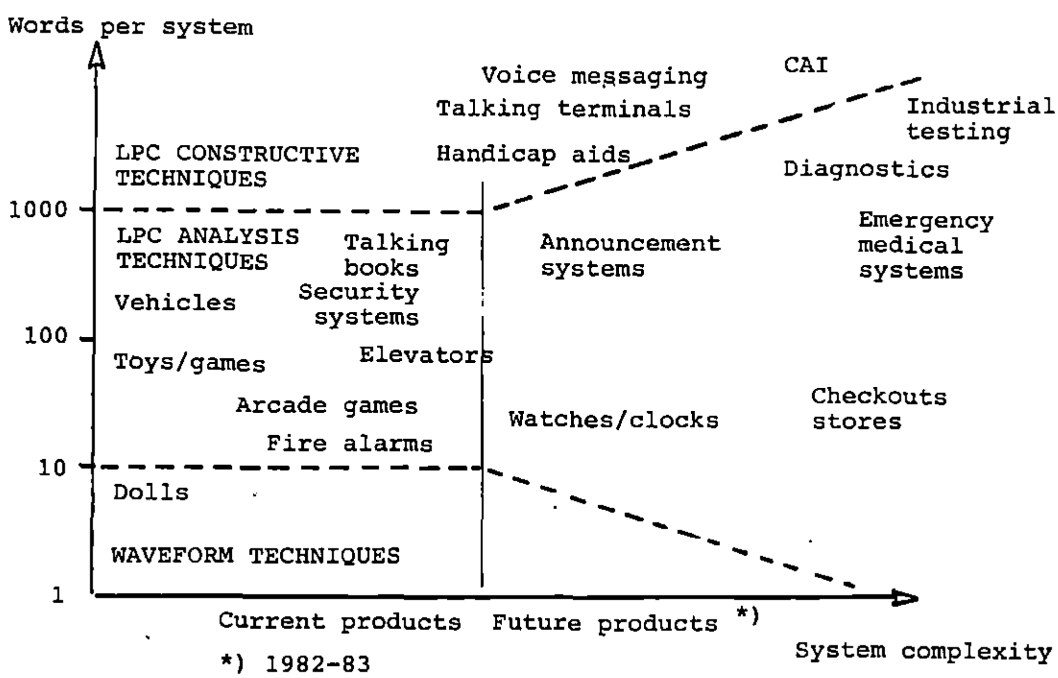

Figure 1. Speech output applications categorized by vocabulary/memory requirements, which translate directly into data rates

Voice synthesis

The process of speech output can most easily be described as in Figure 2. As indicated there, synthesized speech is accomplished in two different ways, either by choosing from a predefined and stored vocabulary or by stringing the smallest building-blocks of speech, the pnonemes or allophones, together to produce virtually any message (Allan (1)).

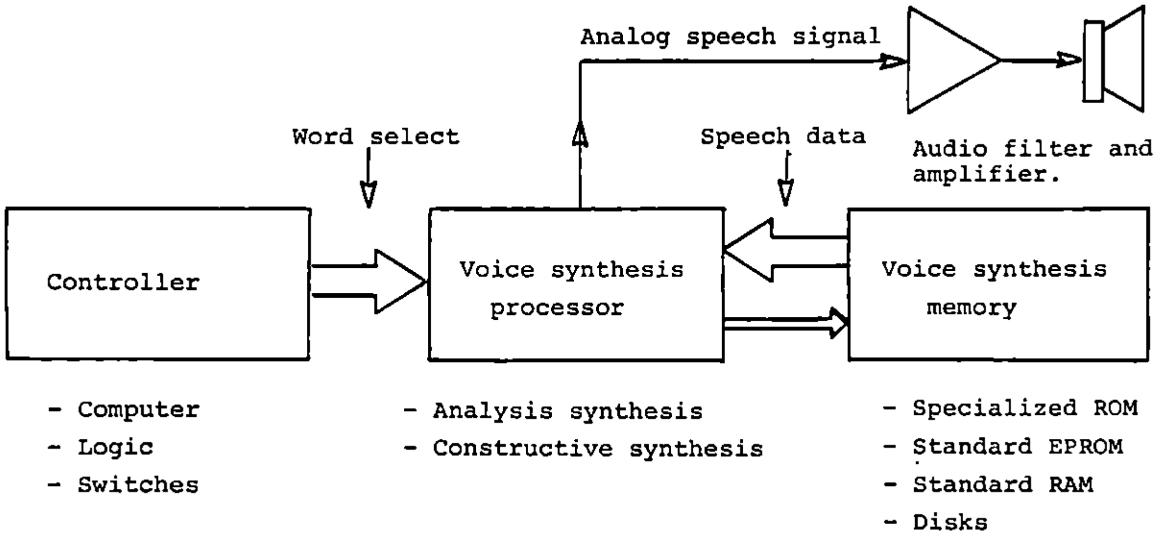

Figure 2 Speech output process

With the first method, called analysis synthesis, the controlling computer program becomes relatively simple. Only the first memory location for the desired word or phrase (normally found in a look-up table) has to be known and sent to the synthesizer electronics. This device then by itself fetches the codes for the whole phrase, thus also liberating the computer for other tasks while the utterance is synthesized. To this date, analysis synthesis techniques have provided the most natural-sounding speech. For a general-purpose application their obvious disadvantage is in the limited number of predefined utterances available. For a specific application, however, it is often possible to cover its area well by choosing from a carefully selected library of phrases.

With the second method, called constructive synthesis, the computer program must control every syllable of the utterance as well as the pitch, amplitude and inflection level of each phoneme. This fact, in addition to the problem of getting continuity at the phoneme boundaries, explains why constructive synthesis results in speech of lower quality than analysis synthesis, even if the phoneme set, 42 in the American-English language, is further broken down into 128 allophones or fractions of syllables (see Appendix E for a list of the phonemes). The codes for the phonemes, as well as the rules for stringing them together considering contextual influences (coarticulation), are stored in the synthesizer chip itself, thus making these speech devices smaller in size than an analysis synthesizer with a somewhat large vocabulary. But the main advantage is that anything can be said; it is only a matter of changing the computer software.

Today hybrid devices also exist, where a vocabulary of preselected words is backed up by constructive speech parts to generate unique words (the "Communicaider" presented in this study is such a device). In most hybrid devices both words and phonemes are LPC-coded (see special section on this topic) and their synthesizers implemented on the same chip (Schalk and Van Meir (49)). Appendix C includes a list of some interesting speech synthesis products.

The results of speech synthesis products are largely evaluated by three measures, according to Flanagan (16):

1. The quality (naturalness and intelligibility) of the artificial voice.
2. The versatility (and fluency) of the message set.
3. The complexity (or cost) of the processor necessary to implement the voice output.

The following sections will go into some detail about different ways to code and reproduce human speech.

Coding and reproduction of speech

Whichever method is used at the time of speech output, the initial coding procedure, which can be carried out by the manufacturer or at the user's site, is principally the same. Somewhat simplified, the steps to transform utterances in human voice into coded, digital form and back to speech again are as follows:

1. The desired words, phrases or phonemes are recorded, normally using a trained speaker. Some filtering and editing may be performed on the recording.

2. The analog recording is broken down into small time-periods called analysis windows, commonly 20 ms each. The information contained in this block of time is called a frame. Often a 30 ms Hanning-window is used additionally to smooth out the transitions betweeen the frames. The frame is sampled and digitized at a rate at least twice the highest frequency component of the speech, according to the sampling theorem. Since, for speech, the highest frequency components with considerable energy (amplitude) are in the 3-4 kHz range, a sampling rate of 8 kHz is often used.

3. The amount of data sampled from the speech signal can be significantly reduced, because human speech contains much redundant information. A variety of mathematical algorithms can be used, and the result is the so-called speech codes (Schalk et al. (50)). See special section on this topic.

4. The codes for the target words, or phonemes, are thereafter carefully "lifted" from the contextually recorded sentences and edited, i.e., the pitch, amplitude, duration and inflection levels are manually fine-tuned, so that the produced speech and the transitions between

the words will sound as natural as possible.

5. The final step in the recording stage is storing the speech codes in Read Only Memories or other memory media. For phoneme synthesizers, the codes are often stored in ROM-area on the same chip.

6. At the time of playback (synthesis), the controlling program (or otherwise generated control data) decides what phrase or phoneme will be uttered. A look-up table, residing in the external ROMs or in the synthesizer chip, gives information on the actual memory location where the first speech code for the desired utterance is stored.

7. The digital voice synthesizer itself fetches the codes and decodes them using an algorithm complementary to the one that was used in coding the speech. See special section on this topic.

8. The digital speech signal is converted to an analog signal, filtered and finally amplified in order to drive a loudspeaker.

Data reduction methods

For both analysis and constructive synthesis, the data reduction and speech coding are done using either of two methods. One of these operates in the time-domain and the other in the frequency-domain. Each of these methods has different bit-rate characteristics and produces speech of varying quality. The quality of synthesized speech is generally split up into four different categories (listed in order of increasing quality), namely synthetic, communication, toll and commentary, as shown in Figure 3 (Parikh et al. (43)).

Time-domain techniques The time-domain method of speech analysis, also called waveform coding, interprets speech as a bit stream that can be approximately reconstructed into the original waveform for a variety of signals. The coding procedure may use the waveform's statistical properties such as bandwidth, amplitude distribution, autocorrelation and power spectral density. Thus, depending on the mathematical algorithms used, waveform coding can be divided into several techniques:

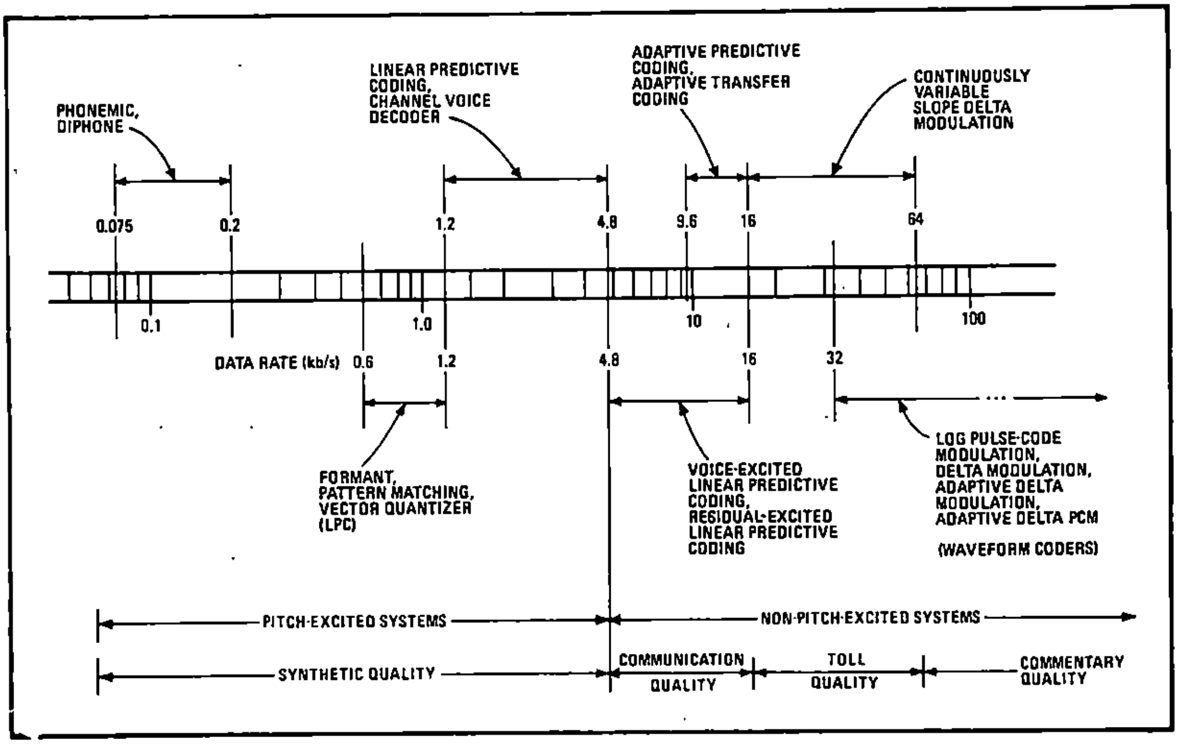

Figure 3 Classification of speech-synthesis systems according to voice quality

1. Pulse-code modulation is a direct digitization of sampled speech after filtering with a low-pass filter. It is the simplest approach, but also the most memory intensive. Typically a data rate (measured in kilobits of coded information per second of produced speech) of 64 to 96 kbits/s is necessary, providing a sampling rate of 8 kHz.

2. Delta modulation, or differential modulation, attempts to achieve some data compression by using differential quantization, i.e., the differences between successive waveform samples are stored rather than the magnitudes of the samples themselves. As this method requires knowledge of previous speech samples, similarities with the predictive methods (described later) exist. A variety of delta modulation techniques have been invented, e.g., Adaptive Delta Modulation, Linear and Logarithmic Delta Modulation and Continuously Variable Slope Delta Modulation. The data rates necessary with these techniques are still high, typically 16-32 kbits/s (Helms and Petersen (24)).

3. The Mozer technique relies on the fact that human speech is often periodic in nature and that the human ear is not particularly sensitive to the phase information of the speech output. With this method, the phase angle of the speech is adjusted to obtain a time-symmetric segment of a waveform for each frame period. Since

half of the resulting waveform then is redundant, only half need to be stored. The data rate required is about 2 kbits/s. The Mozer technique could be combined with one of the other waveform coding procedures. The Digitalker chip from National Semiconductor used in this project is an example of a phase-compression synthesizer (Wickersham et al. (65)), and a picture of how this technique works is shown in Figure 4.

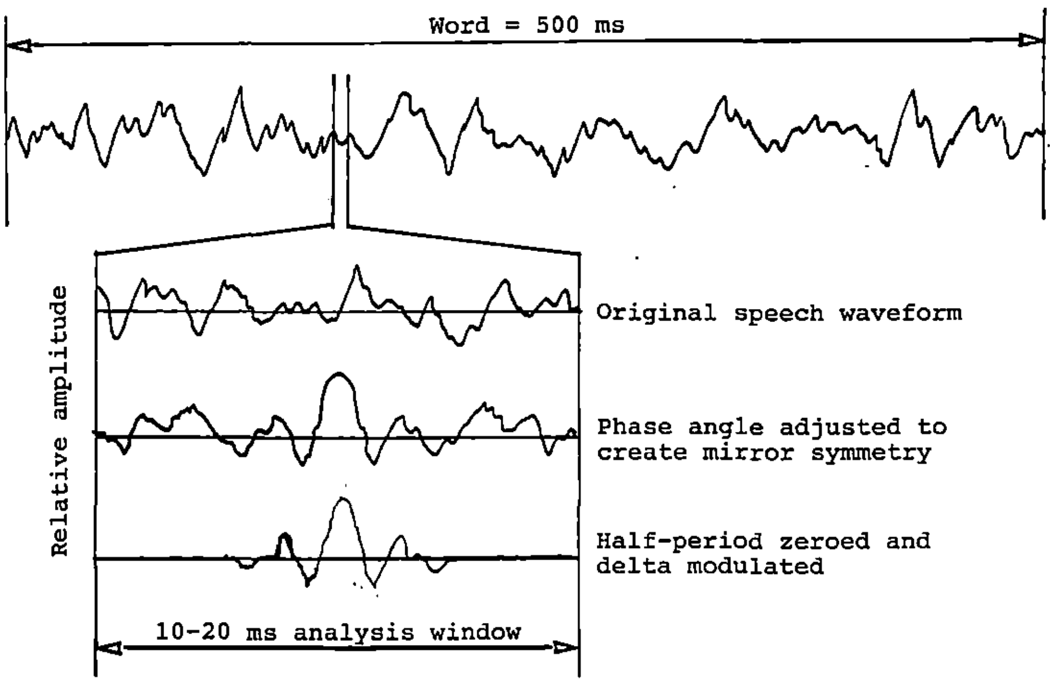

Figure 4 Waveform compression by phase-angle adjustment

The time-domain techniques result in synthesized speech of at least toll quality, but they rely on high data rates thus requiring large storage capabilities of the computer. About 50 bits of information, somewhat varying depending on the model and technique used, is necessary to describe the frequency, amplitude and voicing of speech at any instant of time. Using a sampling frequency of 8 kHz, the initial data rate comes to 400 kbits/s. For a synthesized voice to sound natural, it is necessary to update the speech data at least 40-50 times per second. This gives a minimum data rate of 2-2.5 kbits/s for waveform coded speech synthesizers. The commercially available products today mostly use rates in the 3-50 kbits/s range, which corresponds to a 10 to 100-fold data compression.

Frequency-domain techniques The method of analyzing speech by operating in the frequency-domain, also called parametric coding, extracts certain parameters from the waveform and codes them as digital bits. The parameters are variables in an electronic model of the human vocal tract, implemented in hardware or software or a combination of both. This model assumes that the sound has either a voiced (a periodic sound that originates from air passing through the vocal cords causing them to vibrate periodically) or an unvoiced source (a sound produced by turbulent air flow that bypasses the vocal cords). This, however, is not the case for nasal sounds and voiced fricatives that combine the voiced and unvoiced sources (Parikh et al. (43)). The electronic implementations of this model include a white-noise source (a psuedo random-number generator) to emulate background unvoiced sources, a periodic pulse generator to emulate voiced sources, frication and nasal resonators in the form of digital filters and, finally, output gain stages. When speech is, reconstructed the waveform may not appear to be similar to the original, but it will sound similar.

The frequency-domain methods require prior knowledge of how the signal was generated, because they develop a time-invariant linear model of a sound for a short interval of time. However, speech has non-linear characteristics because the vocal cords, the glottis, are coupled to the vocal tract. These and earlier mentioned discrepancies in the model of voice generation result in a speech-quality that is inferior to the quality of waveform coded speech, but the advantage is the much lower data rates needed, about one-tenth of the rates used for pulse-code modulated speech.

Parametric coding consists of a series of actions in addition to the recording and digitizing stage. These are the following:

1. Speech-parameter extraction
2. Speech-parameter editing
3. Quantization
4. Repeat framing
5. Parameter formatting

Regarding the parameter extraction and coding as well as the final synthesis, parametric coding can be further divided into two closely related techniques, formant synthesis and linear predictvie coding (Parikh et al. (43)). They both are based on linear predictive analysis, which in turn relies on the fact that the degree of change in speech-formant frequencies can be closely predicted based on previous speech samples. The method uses linear equations to make such predictions. Somewhat different sets of parameters are produced by the algorithms depending upon the synthesis technique to be used, but the sets all include excitation source parameters (voiced/unvoiced source, pitch and relative energy) and filter parameters for the mathematical model of human speech formation. The spectral parameters for both linear predictive coders and formant synthesizers are extracted most easily using autocorrelation, because that algorithm always provides a stable transfer function. Figure 5 shows the linear predictive analysis procedure schematically, followed by a description of the two most important parametric techniques.

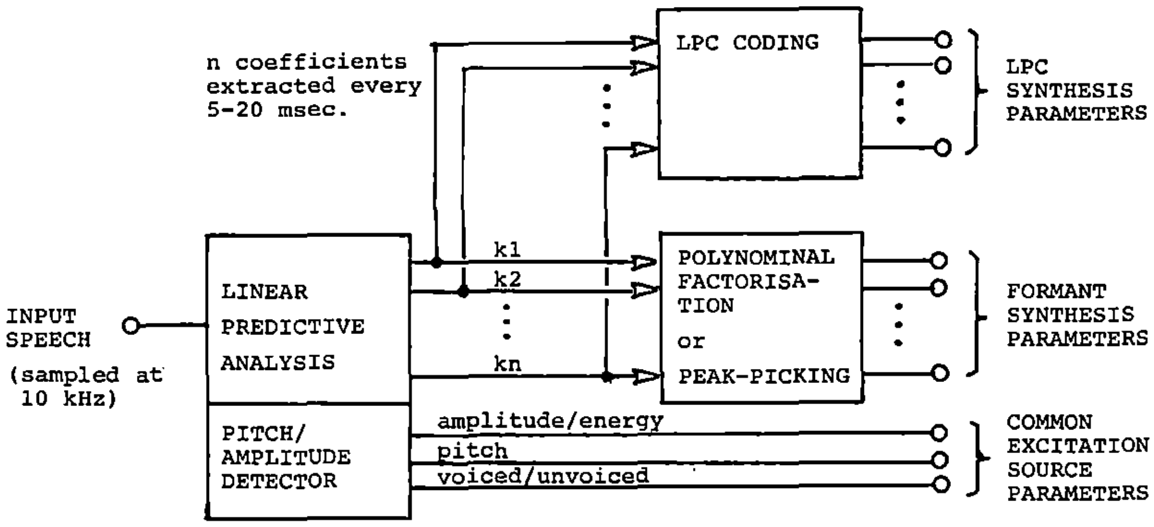

Figure 5 Linear predictive analysis and parameter calculation

1. Formant-synthesis (also sometimes referred to as phoneme synthesis) is based upon the formants, the resonances of the vocal tract. The parameters needed for formant synthesis are extracted, normally at 5-10 ms intervals, using some form of linear predictive analysis described above, often called formant filtering. When the sound later is reproduced, the formant synthesizer, as well as the LPC synthesizer, uses electronic sources for voiced/unvoiced sounds. But a formant synthesizer implements the models of the glottis, vocal tract and lip radiation as sections of a second-order time-varying all-pole multiresonance filter. The main advantage of this method, often used in phoneme or allophone synthesizers, is the low data rate required, typically a few hundred bits per second or less. However, the produced voice sounds rather robotic-like (Bruckert et al. (4); Parikh et al. (43)). The SC-01 phoneme synthesizer chip from Votrax used in this project is an example of a formant synthesizer.

2. Linear Predictive Coding (LPC) (Flanagan (16); Secrest et al. (52); Helms and Petersen (24); Parikh et al. (43)) extracts the so-called k-parameters (from which the reflection coefficients for the 10-pole timevarying, digital filter are calculated) from the analog signal. The parameters are usuallly obtained for a 20-ms analysis-window period, which is short enough to capture most dynamic components of the voice. Spectral parameters are calculated for each frame and during calculation it is assumed that the speech properties are invariant for each frame length (linearity).

In the linear-predictive-coding method, the different anatomical vocal structures are modeled as a unit lattice filter, which has N poles. The value of N depends on the bandwidth of the speech signal and is often set to 10 or 12, corresponding to a bandwidth of 4 or 5 kHz. Current LPC-chips use a single-stage filter to execute in pipeline the 200,000 additions and 200,000 multiplications per second required to produce human speech. This method implements a 10-stage lattice filter and does the work of 20 adders and 20 multipliers (Schalk et al. (50); Parikh et al. (43)). This shows a special advantage of LPC-chips: their high level of integration. Figure 6 shows a schematic for speech generation with parametric coding techniques.

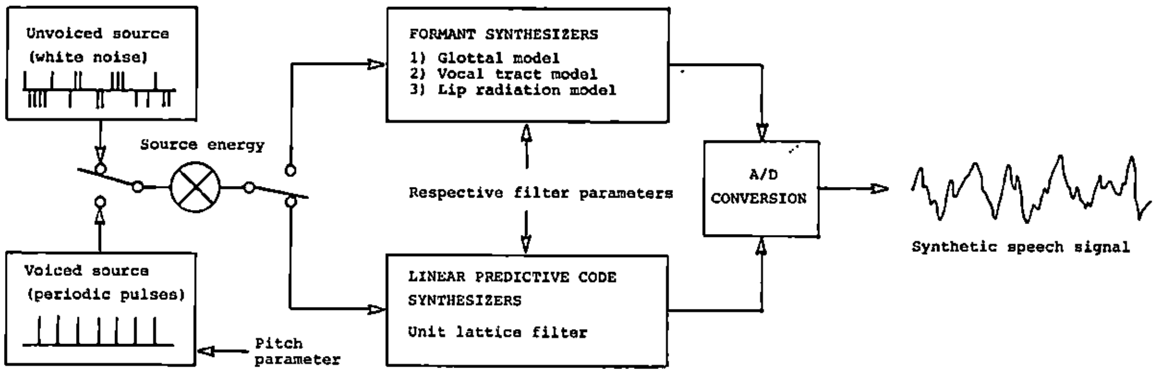

Figure 6 Model of speech generation for parametric coding techniques

Parcor (partial autocorrelation, used mostly by Japanese manufacturers), a close relative of LPC, is statistically similar but is more sensitive to rapidly changing speech transients that might otherwise be smoothed out by LPC algorithms. RELP (Residual Excited Linear Predictive Coder) is an extension of LPC that permits filter excitation to be controlled by the microprocessor, so that the user has the advantage of adaptive filtering and can modify the filter's transfer functions (Allan (1)). LPC analysis and editing can compress the audio input pattern from several hundred kbits/s to an average of 1.2 kbits/s.

The quality of speech is not as high using parametric methods as with waveform coding, but the data rate necessary to produce intelligible speech is much lower, between 0.1 and 2.5 kbits/s. Therefore, this type of synthesizer is most common on the market.

Voice recognition

Artificial understanding of human speech is based upon the complex process of analyzing an unknown voice pattern, matching it against a number of known utterances and finally selecting the closest match or verifying its, or the speaker's, identity (Schalk and Van Meir (49); Scott (51); Dusek et al. (14)).

Depending upon the application, voice recognition is generally divided into the areas of word recognition, word verification (where the computer asks the questions and the user answers them) and speaker verification (where the identity of the speaker is verified). For the two latter cases, the matching process may be comparatively simple, as the number of possibilities is limited. For word recognition, the matching task is difficult as, ultimately, any utterance from any speaker should be understood with high accuracy, preferably in real time. This requires both high computing capability and a large memory bank. On the other hand, word recogniton is the area with the most interesting applications, especially in the biomedical field.

Word recognition The area of word recognition can be divided into two groups, template analysis and feature analysis. These techniques, however, have much in common. The first method is best suited for speaker-dependent systems, whereas the second method simplifies speaker-independent recognition (Dusek et al. (14), Wilson (67)). Template matching systems do not mimic the human auditory process as do feature analyzers, and they are sometimes called signal matchers as opposed to speech recognizers.

Template analysis   Most word recognition systems now produced are based on so-called template analysis. It is performed in the following steps, as shown in Figure 7:

1. Sampling and digitization of the speech waveform, which is done in analysis-windows of commonly 20 ms length smoothed out by a 30 ms Hanning-window. The data within that period are called a frame.

2. Feature extraction from the unknown utterance, using methods and algorithms similar to those used in data reduction and coding. Commonly, so-called formant filtering, extracting the formants of a sound, or linear predictive coding is used for this acoustic analysis, both including autocorrelation calculations.

3. Time registration of the analysis target and calculation of time normalized distances between input speech patterns and reference patterns. This is a step in the process of deciding where one word ends and the next one starts, so-called end-point detection, which is a necessity for recognition of continuous speech.

4. Pattern-similarity measurement, in which the extracted features are matched against those for already known words. This is a frame-by-frame comparison of speech data with reference data, or templates, and consumes the largest part of the processing time. The matching could be performed using basically two methods:

   a. Formant-matching is used if the input speech signal was divided into several frequency regions, using a filterbank, and the amplitudes for different frequency components extracted. These data are analyzed in an all-zero inverse filter, which, somewhat simplified, is the inverse of the reference template data. When the spectral valleys in the inverse template match the spectral peaks in the input signal, a low-energy residual error signal will result. These error or difference parameters are computed using autocorrelation as well as residual-energy algorithms. See Figure 8.

   b. LPC-code matching is based on linear predictive coding described above. Speech recognizers using this technique have the best performance, but also the highest complexity and price. One limitation for LPC as well as formant matching is that the analysis is done with fixed sampling intervals, which tends to smear fine temporal events.

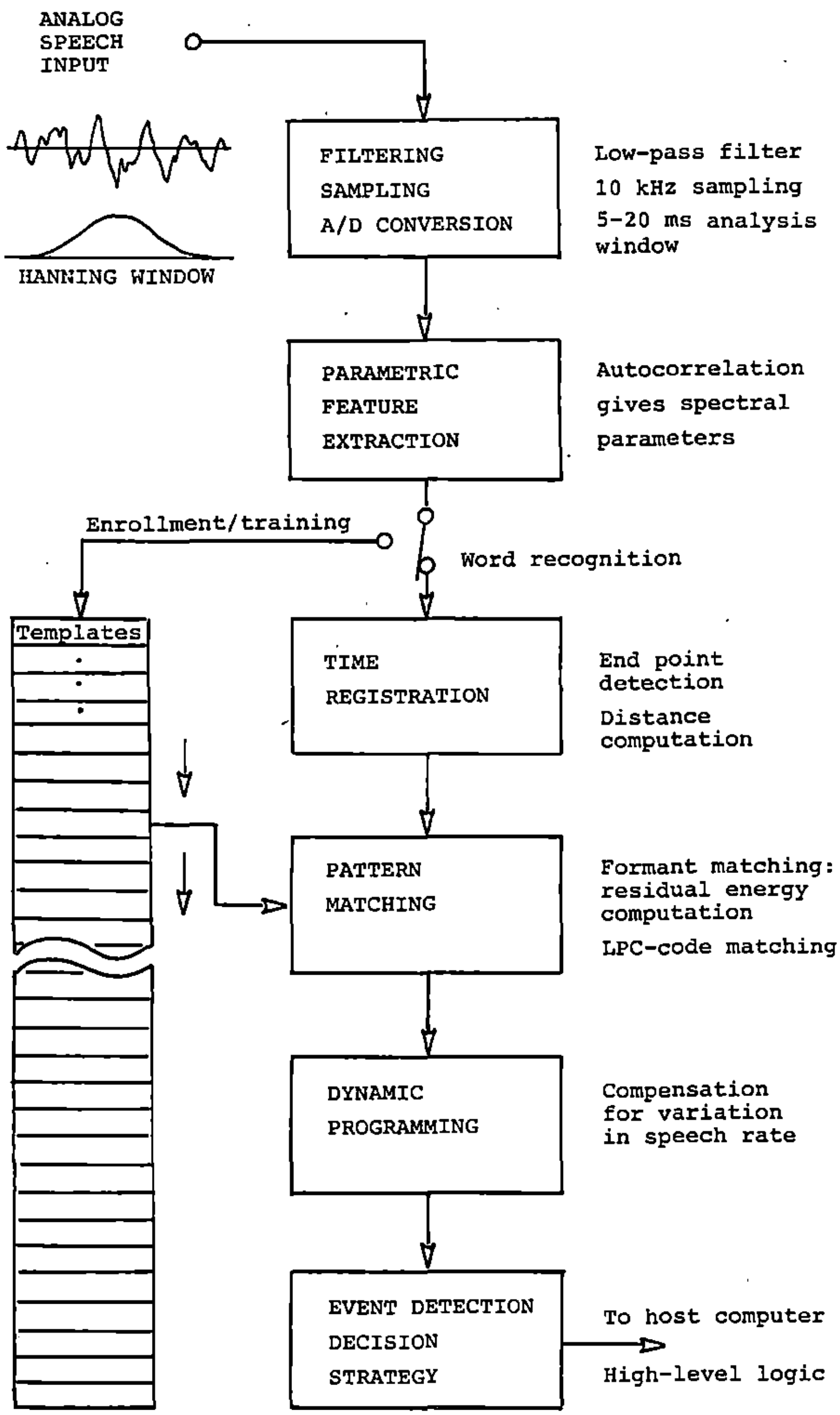

Figure 7 Word recognition using template analysis

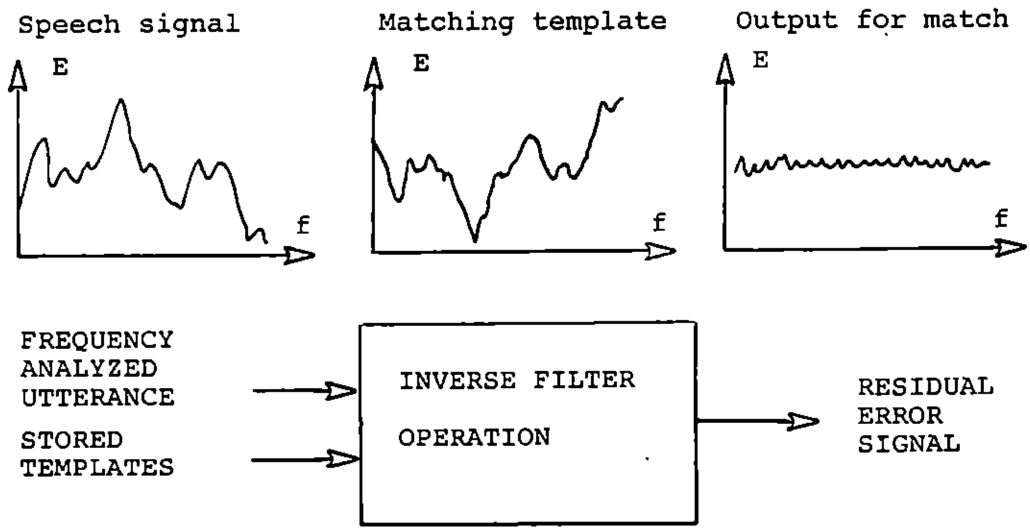

Figure 8 Residual energy calculation

5. Dynamic programming is often used additionally to compensate for variations in the length and timing of the input, i.e., the user's/users' varying speaking rate. The matching in this method is based upon the residual errors for the whole utterance rather than on any of the individual frame residuals. The most popular dynamic programming technique is called Non-Linear Time Warping (DTW) and uses a reference period that is twice the input period. The procedure is computation intensive but accepts non-linear patterns to speed up signal processing and to allow acceptance of speech inputs with a varying time factor of 2:1. An alternative to DTW is linear time compression. Dynamic programming increases processing time significantly but system reliability is greatly enhanced and there is no need for pausing between words (Scott (51)).

6. A decision strategy, that could be written in a high-level language, finally determines if the best match meets the threshold for recognition, as well as how to act upon the recognized speech input. If two templates were found having the residual errors too close together, the utterance will be rejected and the user prompted for more information to minimize the jeopardy of a system substitution mistake.

7. To these steps should be added the enrollment procedure or system training, which keys the recognition to a certain speaker and is done once before the recognizer is put to work. The enrollment creates a set of feature vectors called templates for each word in the system's repertoire. The vectors, which are included in the similarity

measurement as a part of the recognition process, define the spectral shape of the reference pattern for individual vocabulary words. All utterances must be enrolled, and successive repetition of the utterances leads to a significant improvement in performance for the recognizer.

Feature analysis A somewhat different method of speech recognition, called feature analysis, is based on the phonetic features of voice and uses no templates for matching. Instead, reference data are used to help identify features like leading consonant, middle vowel and ending consonant. These systems are speaker-independent, and need no training. The memory size needed is small even for large vocabularies, and the processing and response time is short and almost independent of the vocabulary (Schalk and Van Meir (49)). Figure 9 shows a simplified schematic of feature analysis.

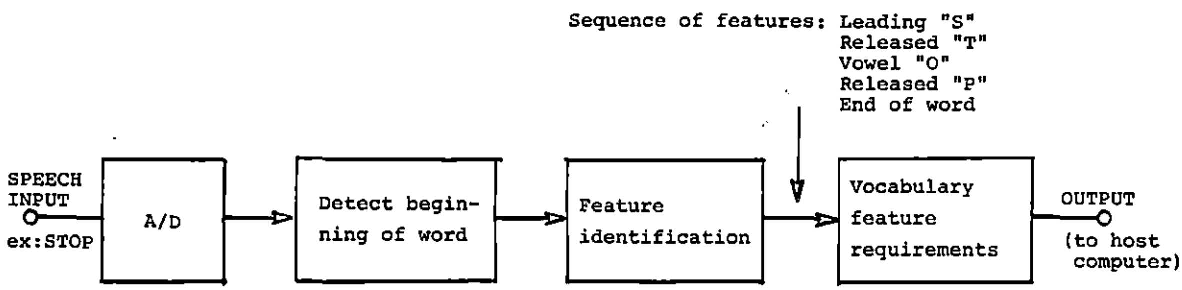

Figure 9 Word recognition using feature analysis

All these steps for any method of speech recognition (and for speech synthesis as well) can be carried out with emphasis on either hardware or software, but as speech recognition is an area undergoing rapid development, the systems that are software-based have an important advantage. For these, the designer can easily encorporate enhancements in the recognition algorithm. Most hardware-based systems tend be obsolete in about three years.

The level of sophistication could also serve as a classification of speech recognition systems. The evaluation criteria are then basically four:

1. Speaker dependency/independency (only one person's voice is recognized or any person's).
2. Limited/unlimited vocabulary.
3. Isolated words/connected speech recognition.
4. Recognition accuracy.

Speaker-independent systems have to face the problem of differences in pronounciation of the same word between different individuals. A study at Bell Laboratories shows that up to 12 different patterns provide a useful description of the ways that English-speaking people utter a given digit. Therefore, when speaker-independent systems using template matching are developed, the multiple patterns that characterize a large population are determined from voluminous measurements on many talkers. Statistical "clustering" analysis (not averaging!) then determines the set of multiple patterns that most nearly characterize the large population (Flanagan (16)).

The recognition accuracy of any product is strongly dependent on the choice of vocabulary, including a careful avoidance of similar-sounding words. Syntactic rules, i.e., grammatical and contextual rules, deciding what limited number of words that can follow the words that have just been recognized, can also greatly improve recogniton accuracy. The highest recognition accuracy, over 99%, is found in word verification systems that are based on a one-word vocabulary, i.e., only one word or phrase is acceptable at a given time. More data can be kept on each word and more processing time can be devoted to that single word. Furthermore, the vocabulary of word verifiers becomes completely disk-based and is limited only by the computer's mass-storage capability.

The top word recognizing products of today recognize continous speech with 95-98% accuracy, if spoken by a selected person choosing from a vocabulary of 400 words maximum.

The applications of reliably functioning voice recognition systems are numerous. This quotation from a representative of Votrax, a leading manufacturer of speech recognition equipment, shows the benefits:

> "Speech recognition does more than leave hands and eyes free. It leaves the operators brain free to focus on the task at hand."

Latest technology

Today it is possible to integrate most of the electronics needed for word recognition on one chip, also commonly including capabilities for voice synthesis, voice coding and voice verification. This is often done using a high-performance single-chip microcomputer optimized for digital signal processing, such as the TMS 320, NEC-7720 or Intel 2920. The TMS 320, for example, can execute 5 million instructions per second (like a 200 ns, 16- by 16-bit multiplication), has 16-bit data I/O words and works with 32 bits internally. Its on-chip ROM-area can be programmed as, for instance, a feature extractor for a LPC-10 model operating in real-time.

Another approach is to construct a specialized voice processing chip. Sharp Corp. offers a CMOS single-chip voice recognizer for about $7.00 a piece. General Instruments recently introduced its SP1000 chip that does both LPC-based speech recognition and synthesis and easily interfaces to a microprocessor (using the Rockwell 6502, no interface circuitry at all is necessary). Price: $7.50 per chip! Voice recognition and synthesis are also found in wristwatches from both Swiss and Japanese manufacturers.

Portable Display Devices

For text output, the Cathode Ray Tube (CRT) is the most cost-effective solution. However, this device is not easily portable nor very energy-efficient. During the 1970s, two types of devices have been developed that to varying extents meet these requirements: Light Emitting Diodes (LEDs) and Liquid Crystal Displays (LCDs) (Kauffman (25); Shepard (53)). Only the latter type will be briefly presented here, though some comparison will be made to LED displays.

Liquid crystal displays

The LCD technique is based upon the change in the light-transmission properties in certain organic "liquid crystals" (an amorphous material) in the presence of an electric field. The liquid crystal material is nonconductive and as the molecular alignment is caused by the electric field only, these devices do not in theory require any current flow. However, some capacitive leakage does occur as AC-voltages are required to prevent electrolysis of the crystal material (LCDs will typically tolerate a DC-component of only 25 mV in the drive signal). Thus LCDs consume about 0.5 $\mu A/cm^2$, which is very little compared to the power consumption of LED displays: 10-20 mA/display segment.

LCDs have response times in the millisecond range, which is much slower than LEDs. The response time gets longer at lower temperatures and at some point, typically -10°to -30°C, LCDs stop functioning altogether due to the material's amorphous character. Thus the operating range for LCDs is commonly -10°to +90°C (LEDs: -40°to +90°C), which, however, is at least as much as the -10°to +70°C temperature range for commercial integrated circuits.

Due to their operating principle, the contrast and visibility for LCD displays increase in high ambient light, while LEDs have the opposite behavior. On the other hand, LCDs have a maximum viewing angle of about 75 degrees compared to the 150-180 degrees for LEDs. The low viewing angle for LCDs is mainly due to the necessary polarization of incoming light before it penetrates the liquid crystal molecules.

Because of these mentioned properties liquid crystal displays are the best, or maybe even the only alternative for low-power applications in light environments like daytime outdoors.

LCD displays, as well as LEDs, seldom fatique and easily meet Original Equipment Manufacturers' (OEM) specifications: a life-time exceeding 50,000 hrs and a decay in performance (the so-called MBTF-ratings) of less than 1% per 10,000 hrs in operation. LCDs can be made quite large and can display any symbol that can be etched.

LCDs are inherently difficult to multiplex (i.e., drive the different symbol segments in a scanning procedure) due to the requirement of an AC drive signal. LEDs, on the other hand, are easy to multiplex as they operate on DC voltage. Furthermore, LCDs are not being integrated with the drive circuitry to the same extent that LED devices are due to manufacturing problems; it is difficult to bond a crystal display to a silicon chip. True integrated LCDs are at least a few years ahead. However, many intelligent devices exist on the market, where a package in addition to the display also contains a controlling microprocessor. Commonly, they accept ASCII-coded information and some large displays also give possibility of Direct Memory Access (DMA). The PCIM-201 used in this project is an example of an intelligent display.

Future development will cause improvements especially in the crystal material. Current LCDs use the so-called Twisted-Nematic (TN) technology, while newer devices may use the new Guest-Host (GH) methods. In GH-displays, a certain color-dye molecule attaches to a host-molecule when a certain electric field is applied. Beside fascilitating colored displays, this technique eliminates the need for polarized light, which will greatly improve the maximum viewing angle.

Of liquid crystal displays the alphanumeric devices are among the most expensive. Depending on the level of intelligence in the display package, the price ranges from $1.00 to $4.00 per character. However, this cost is expected to decrease to 50c - $2.00 per character by the mid 1980s. Appendix F includes some alphanumeric LCDs of interest.

Handicap Aids Using Voice Technology and/or Text Displays

Voice technology opens up vast possibilities of helping vocally or visually impaired people to communicate and study, as well as helping the motor handicapped to voice-control devices of personal need or manage a professional job. The aids under consideration here are all intended for use by handicapped people whose problems are caused by loss of muscular control over the speech mechanism or limbs and not by brain dysfunction. The terminology for all types of communication disorders has been well clarified and arranged in dictionary form by Nicholosi et al. (41).

Because of cost and technological advances (or lack of such) most handicap aids of this category are applications of voice synthesis. Beside the pure message output devices for vocally impaired people (see below), products, or at least prototypes for the visually handicapped have been developed. Examples are talking computer terminals (Ward and Subandhi (64)) or a typewriter that vocally repeats the line (or word) just typed (Brown and Aitchison (3)). Such aids are based on the so called text-to-speech converters that accept ASCII-coded text input and ouput highly intelligible and correctly pronounced speech (Bruckert et al. (4)). An example is the Prose 2000 board from Speech Plus, that has a pronounciation accuracy of more than 99.5%, i.e., only one word for each 200 is mispronounced. Furthermore, of tremendous importance for the blind are systems where an optical reader is hooked up to the text-to-speech converter like the Kurzweil machine. This allows the visually handicapped to read and study almost any book or paper. Some inexpensive text-to-speech converters are listed in Appendix C.

For both vocally/audially and motor handicapped people, multifunction home control systems can be found, for example, from Basic Telecommunications.

Voice recognition technology has been used for wheel-chair control (Rastgar and Devaney (45)). In such devices with a very limited vocabulary, the recognition accuracy can be made very high.

Vocal communication aids

Worldwide, there are about 20 companies manufacturing roughly 40 dedicated devices for non-vocal communication, of which some use synthesized speech (Rosen and Goodenough-Trepagnier (48)). Vanderheiden (61) has given a thorough but today somewhat outdated overview of such devices for severely motor-handicapped people. However, most of these products suffer from an extremely low communication rate. Rates of one word per minute are typical for single-switch devices and rates of more than five words per minute are uncommon. Even though these devices themselves cannot carry all the blame for slow communication, since the rate depends on the user's skill with the device, this is in sharp contrast to normal speech rates, which range from 100 to 200 words per minute. Goodenough-Trepagnier et al. have studied this problem thoroughly and developed systems with a high speech rate as an important feature ((22), (23)).

The major critique against these existing products is, however, their very high price compared to their level of sophistication. While a speech-toy like Texas Instrument's Speak'n Spell sells for a few tens of dollars, the handicap speech aids are found to have prices in the $200-7,000 range. An example is the Phonic Handivoice Mirror 120 from HC Technology. This device measures 90x118x250 mm, weighs 1.75 kg, uses the Votrax SC-01 chip and can speak 888 predefined words and 16 common phrases based on phoneme synthesis, selected with a 3-digit code from a 4x4 keyboard. Its price is close to $3,000, and its smaller relative, The Handivoice 110 (with 373 words selected from a 128-point membrane pad) costs $2,700.

Visual communication aids

Few devices using a text-display in aiding vocally handicapped have been found in the literature. One system was presented in 1978 that was designed to enable aphonic (without speech), motor-deficient patients to communicate (Vartanian and Groesberg (62)). The device displayed alphanumeric messages with up to 20 characters at a time on two opposite-facing screens so that the user and respondent could maintain eye-contact. An attached thermal printer provided a hard copy.

PURPOSE OF PROJECT AND DESIGN SPECIFICATION FOR THE "COMMUNICAIDER"

Purpose of Project

The overall objective behind this thesis has been to attain knowledge in two rapidly developing fields of science and technology: single-chip microcomputers and voice or speech technology. Considering the intention of using this insight for future studies, by the author or others, the purpose of this project has been mainly five-fold:

1. To gain familiarity with different types of voice and text output devices, especially the various methods of voice synthesis and their inherent diverse voice qualities.
2. To design a relatively inexpensive and small communication aid for the vocally impaired.
3. To use a single-chip microcomputer to control the selection and construction of messages, and interface this computer to the output devices.
4. To integrate the various devices into a demonstration unit, still allowing the individual components with or without interface circuitry to be used in other projects where voice/text output could be applied.
5. To create system software that enhances both ease of use and data generation for output messages.

Design Specification for Hardware and Software

The following criteria have been important in the development and evaluation of the "Communicaider":

Hardware

1. Be based as much as possible on existing and tested products to enhance reliability and to save time.
2. Use the least expensive parts for its purpose.
3. Consume a minimum amount of power to enhance future battery operation and portability.

Software

1. Have program and data storage separate (in different EPROMs).
2. Give the user possibility of generating his/her own vocabulary of messages.
3. Have an optional selection of different vocabularies for different situations with easily memorizable codes for each message.
4. Produce messages with a minimum of keystrokes to increase the communication rate.
5. Have the option of either text or voice output.

SYSTEM DESIGN

Overall Design

For the purpose of this project five commercially available (somewhat modified) products have been used: the Z8 BASIC System Controller, the Micromouth and Sweet Talker speech synthesizers, the PCIM-201 liquid crystal display and the Tacti-Mem keyboard. An interconnecting circuit board and a summing audio amplifier have been constructed and the different parts put together as a demonstration unit, called the "Communicaider," rather than a prototype.

The choice of two different speech synthesizers was made for the sake of experiment and for the purpose of showing their diverse voice qualities. However, this choice also makes it possible to "fill in" any word(s) generated by the Sweet Talker in a message primarily produced using the more easily programmed and better sounding Micromouth.

Together, the computer system with its interface circuitry and the speech devices include parts that are redundant or otherwise not needed for the "Communicaider's" intended use. If a commercial product were to be developed, these parts could be "stripped off" from the design.

Integral Parts Used

Z8 BASIC System Controller

The Z8 is a family of single-chip microcomputers developed by Zilog Inc., which also is its main manufacturer. The Z8671 member of this family is a complete microcomputer, where the ROM area has been preprogrammed with a 2K BASIC/Debug interpreter. For more details on this device, see Appendix B or references (70) and (71).

The Z8671 is used in the Z8 BASIC Computer/Controller board by Micromint Inc., around which the presented speech/text output device has been built. The main parts added to this single-board computer are RWM or EPROM memories (space for up to 6K total on-board memory), an address latch (the Z8 has multiplexed address/data bus), an input buffer for an additional input port, baud-rate switches and, finally, voltage translators for RS-232C serial communication. Thus, the Micromint product is a minimum-chip computer. See Appendix B or references (8), (9), (33), (34) and (35) for more details. Figure 10 shows a photograph of the Z8 BASIC System Controller.

For development of software for the "Communicaider," the Expansion Board and the EPROM Programming Board from Micromint have also been used (references (34) and (35)).

Micromouth speech processor

The Micromouth Speech Synthesizer Interface, its correct name, is likewise produced by Micromint Inc. Its function is to give voice output capabilities easily to either an Apple II or a TRS-80 personal computer. Despite the fact that neither of these computers has been used in this project, the Micromouth was chosen because of availability and because it furnishes on-board power-cicuitry, a frequency generating crystal, an audio amplifier and vocabulary ROMs.

The Micromouth is built around the Digitalker Speech Synthesizer MM54104 chip from National Semiconductor Corp. This device synthesizes

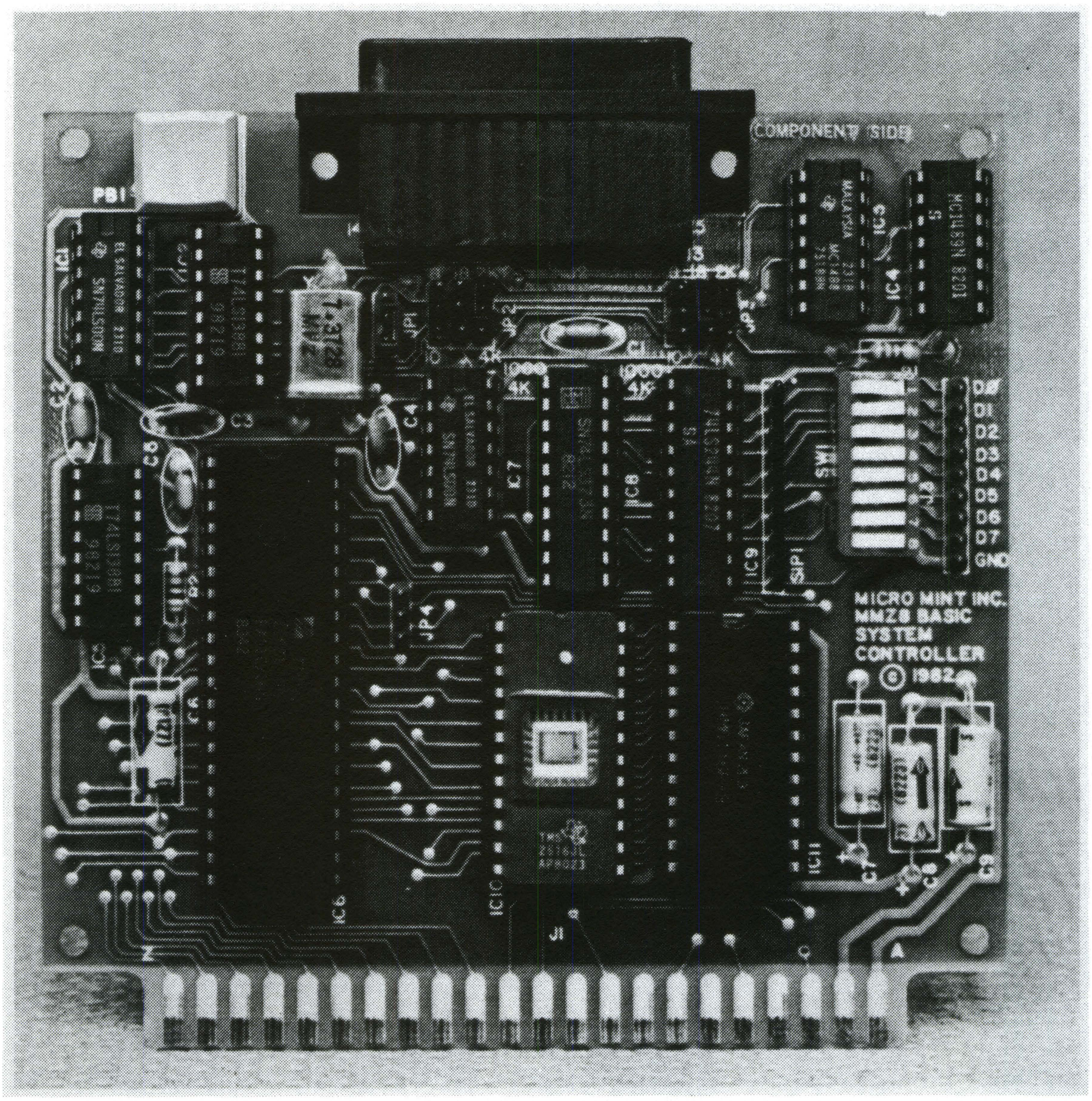

Figure 10   Z8 BASIC System Controller photograph

highly intelligible speech from a predefined, waveform-coded vocabulary that is stored in sets of ROMs. Originally, the Micromouth came with the DT1050 vocabulary chip set, including 143 different utterances (words, letters, digits, tones, etc.) aimed at an industrial control application.

To give more flexibility to the voice output in this project, the DT1057 vocabulary set was added, thus increasing the number of words by 130. For an everyday, personal conversation the selection of utterances is still very limited; but the intent has been more to show the level of speech quality with the Digitalker and how to interface it to the computer, than to create a really versatile set of messages based entirely on this device. Other vocabulary ROMs are avaliable from National, and for a certain application customized vocabularies can be coded and stored. For more details, see references (7), (32) and (38).

Sweet Talker speech synthesizer

The Sweet Talker is another speech synthesizer board from Micromint, intended to give a computer an unlimited vocabulary. Built around the SC-01 phoneme synthesizer chip from Votrax, the board also provides buffered input lines, conversion between CMOS and TTL voltage levels, an RC-circuit for frequency generation and an audio amplifier. The board accepts parallel data that are latched into to the SC-01 chip with a strobe.

The Votrax SC-01 speech synthesizer contains 64 different LPC-coded phonemes, which are accessed by a 6-bit code. In order to decrease the somewhat monotonous and mechanical-sounding voice that phoneme synthesizers produce, which is inherent in this synthesis method, the SC-01 provides automatic inflection of the voice. If this is not enough to produce naturally sounding speech, the two remaining data bits can be manipulated to set the pitch level of voiced phonemes instantaneously. The SC-01 produces intelligble speech at a data rate as low as 70 bits per second. For an interpreted BASIC as the Z8's, however, this is on the upper limit of what the computer can produce, even with a well-written and optimized program. For more details on the Sweet Talker and the SC-01, see references (10) and (63) respectively. Appendix E has a list of the phonemes in the American-English language.

PCIM-201 alphanumeric LCD dot matrix display

The PCIM-201 is a low-cost liquid crystal display for alphanumeric-type text. Sixty-four different ASCII-coded characters can be written on two lines with 16 columns each. In addition to this, the PCIM-201 recognizes a variety of commands for manipulation of the message shown (for example rotation, shifting, blinking and different types of cursors). The cursor can be positioned anywhere on the display and optionally incremented or decremented automaticallly. The display also has a power-down mode and a rapid-load mode. The state of the display device can be obtained by reading a flag register.

In command of the display is a low-power CMOS circuit that provides an automatic refresh for the liquid crystal segments in addition to the text control. The LCD drive voltage is fully temperature compensated and the display contrast is adjustable. The PCIM-201 is bus oriented for direct interface to a microprocessor and the device operates on a single +5 V power supply. Figure 11 shows a photo of the PCIM-201 (mounted on a box for the electrical connections).

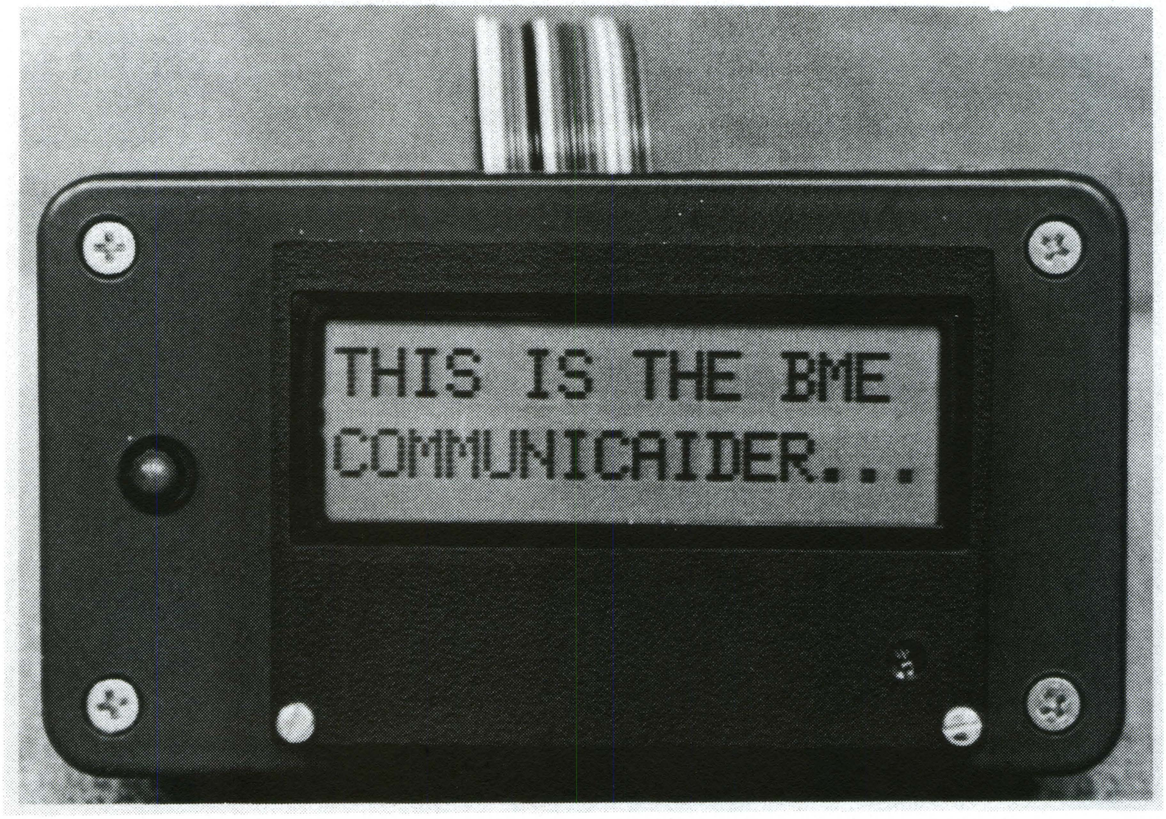

Figure 11 PCIM-201 LCD photograph

RE-AL Tacti-Mem #66(xy)keypad

The Tacti-Mem #66(xy) from RE-AL Inc. is a 4x4 low-cost membrane keypad, designed to be cut and customized in any of seven different keyboard styles. A keystroke's completion is both felt and heard, and the contact bounce is less than 5 ms. Internally, the keypad is arranged into an x-y matrix that gives a unique contact setup on the eight external leads for each key position. These leads have to be monitored by a software routine or connected to a hardware keyboard decoding circuit.

Interface Board

In the "Communicaider," the output peripherals to the Z8 computer, i.e., the Micromouth, Sweet Talker and PCIM-201 mentioned above, have been memory-mapped via the interface board. Addresses HEX A000 through A003 have been chosen, as they do not interfere with other peripherals of the computer. The keyboard, on the other hand, is monitored via port 2 on the Z8.

Through the interface circuitry the Z8 data bus is brought out to all peripherals. However, because this data bus is time-shared with the lower byte of the address bus, logic has been provided to hold the device select signals and catch the data at the right moment after the address decoding has been performed. Furthermore, the SC-01 on the Sweet Talker requires a 100 s set-up strobe pulse, before the data are read. Therefore, a latch saves the data until that peripheral is ready. The Micromouth, on the other hand, requires special logic to address any of the two data chip sets mounted on this board.

The speech output devices each provide a busy signal when they are talking. These lines have been logically ANDed together to a signal that can be sampled by the computer. The status of the LCD display, on the other hand, can be obtained by reading a flag register located in this device. Consequently, the data bus buffer has been made bidirectional. In order to decrease the load on the computer output lines, the address and control lines have likewise been buffered. It may be possible to turn on an LED on the display device to draw attention to a message coming up. Power to the peripherals is finally provided via the interface board.

Figure 12 has a block diagram for the interface board.

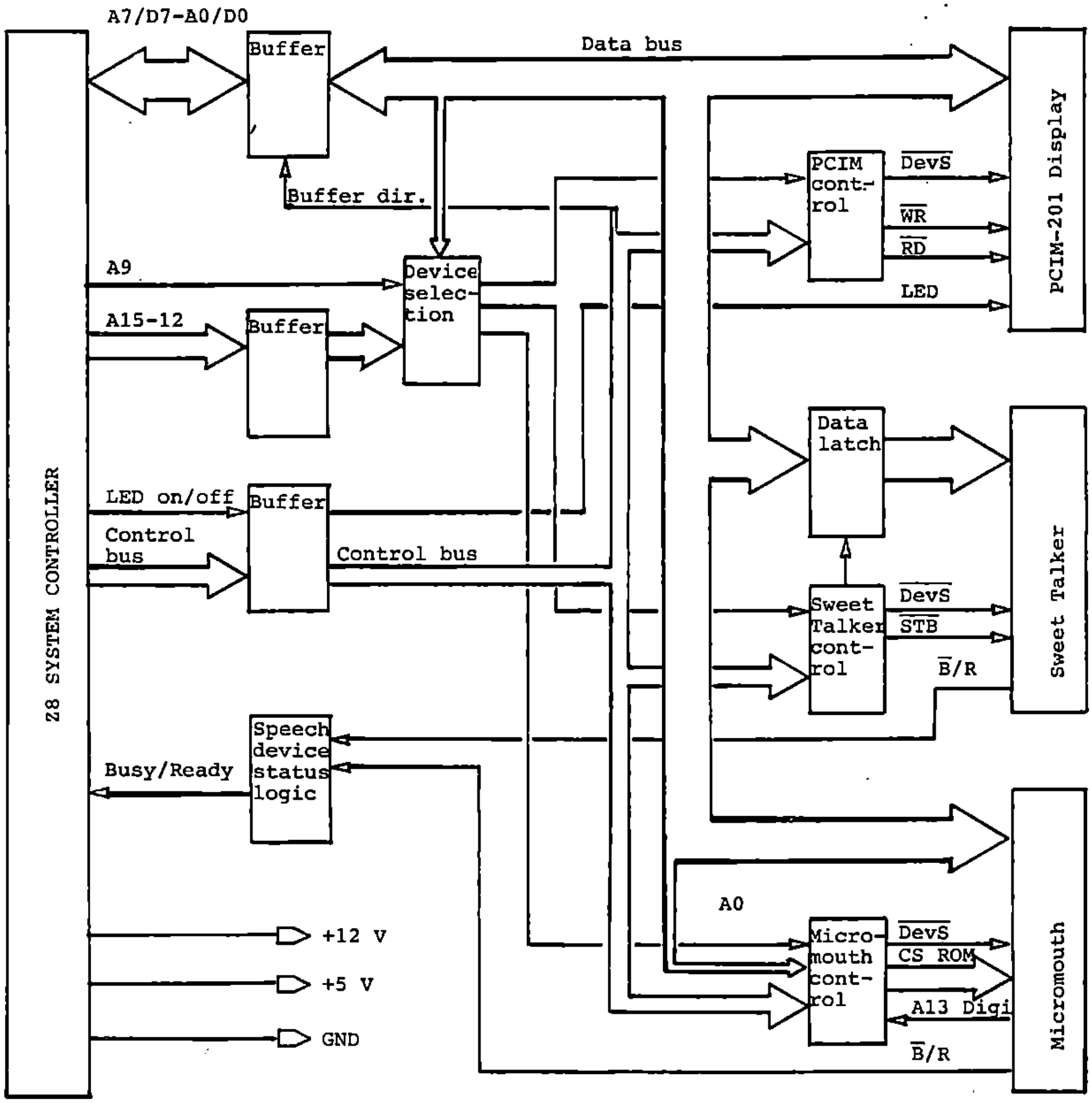

Figure 12 Interface board block diagram

To solve some timing problems, advantage could have been taken of the slow execution speed of interpreted BASIC, but the interface board has been designed to work for high-speed assembly language programs as well. However, it has not yet been possible to verify its proper function for such programs, as sufficient software-development tools have not been available to the author.

DESIGN DETAILS

Interface Board

Components used

In order to reduce power consumption, integrated circuits from the 74HC family (high-speed CMOS) have been used exclusively. Components manufactured with this technology feature the same speed as low-power Schottky (LS-TTL) but the extremely low power consumption of CMOS.

Figure 13 shows the circuit diagram for the interface board and Figure 14 a photograph. It has been designed and built using the following components:

Integrated circuits

1	74HC00	Quad 2-input NAND Gate
1	74HC02	Quad 2-input NOR Gate
1	74HC04	Hex Inverter
1	74HC08	Quad 2-input AND Gate
1	74HC32	Quad 2-input OR Gate
2	74HC74	Dual D-type edge-triggered Flip-Flop with Preset and Clear
1	74HC139	Dual 2-to-4 Line Decoder
1	74HC221	Dual Monostable Multivibrator
1	74HC244	Octal TRI-STATE Buffer
1	74HC245	Octal TRI-STATE Transceiver
1	74HC374	Octal TRI-STATE D-type edge-triggered Flip-Flop

Resistors

1	22 kOhm	timing resistor (R1)
1	14.7kOhm	timimg resistor (R2)
3	4.7 kOhm	pullup resistors (R3, R4 and R5)
1	10 Ohm	current-limiting resistor (R6)
1	47 kOhm	preset resistor (R7)

Capacitors

1	110 pF	timing capacitor (C1)
1	100 nF	timing capacitor (C2)
1	10 uF	preset capacitor (C3)
10	0.1 uF	decoupling capacitors (not shown on the diagram)

Transistor

1	2N2222A	NPN transistor

INTERFACE BETWEEN Z8-BASIC COMPUTER/CONTROLLER AND SPEECH AND DISPLAY OUTPUT DEVICES.

CLAS ROLANDER
4 NOV 1983

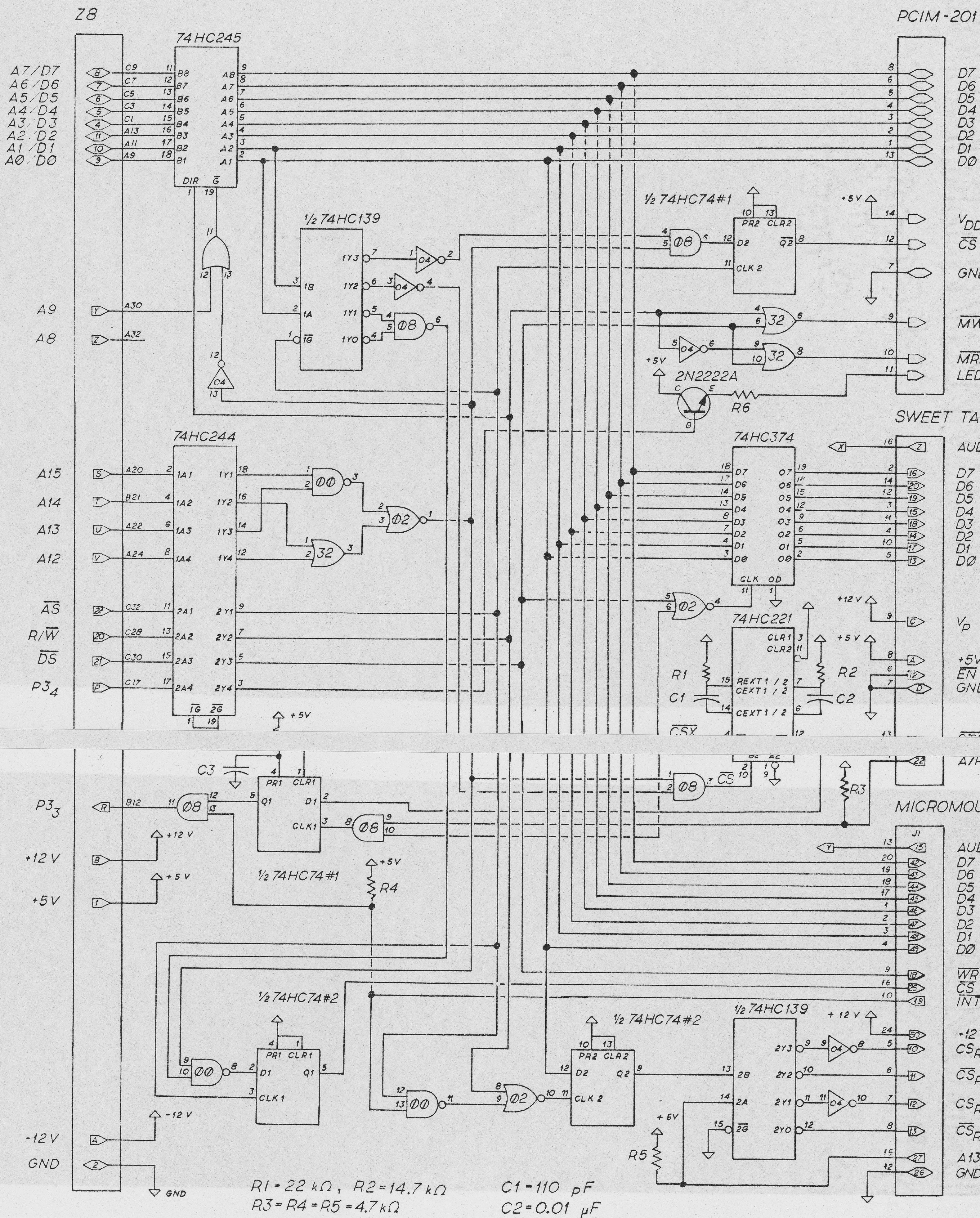

Figure 13 Interface board circuit diagram

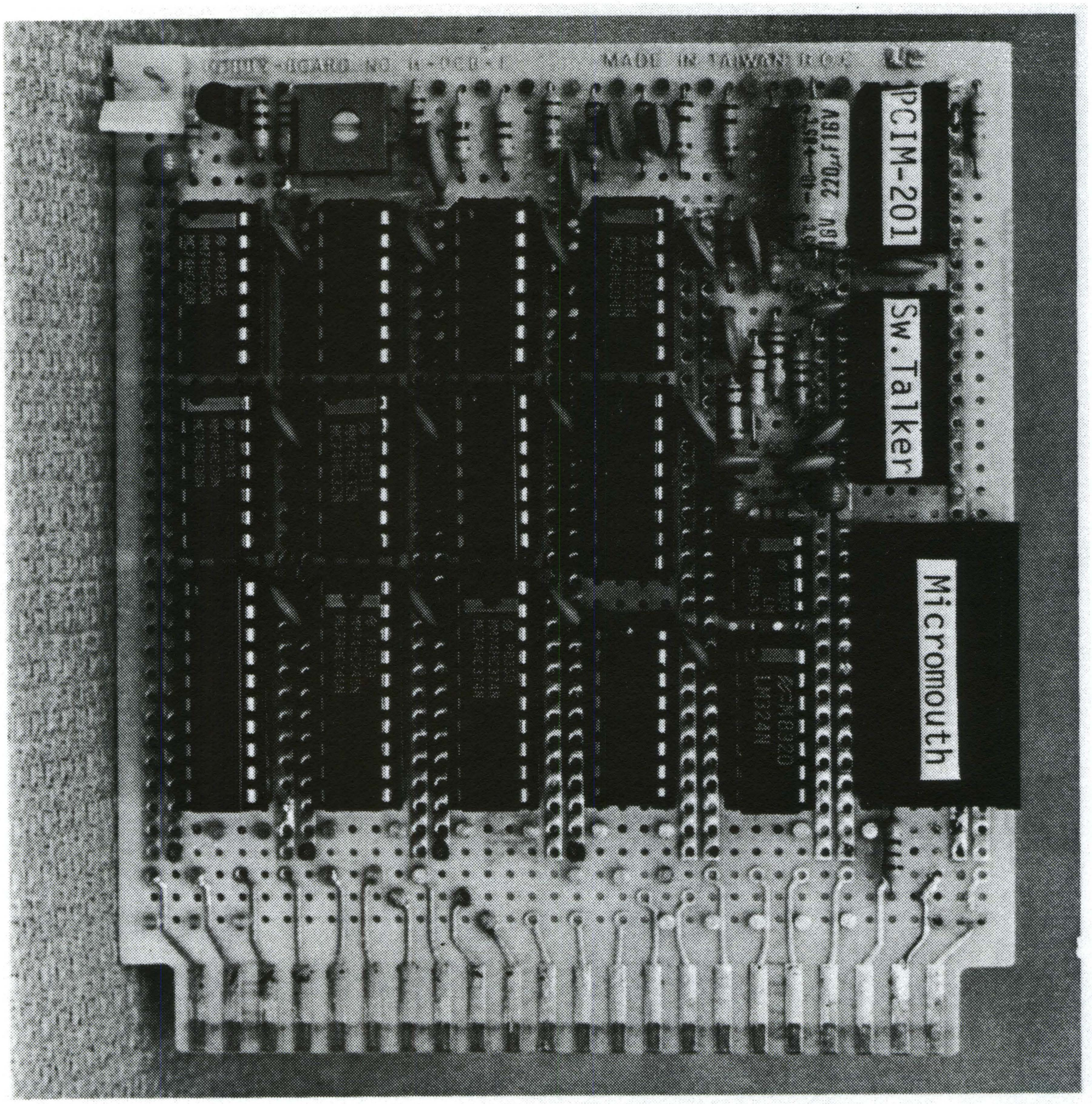

Figure 14 Interface board photograph

Power distribution

The interface board operates on +5 V only, which is provided to the highest numbered pin on each IC. GND is connected to the diagonally opposite pin (number 7, 8 or 10 depending on the size of the IC). The power supply is decoupled at each integrated circuit with 0.1 uF capacitors. Connections for power, GND and decoupling capacitors are not shown on the circuit diagram.

The interface also provides +12 V and GND to the Micromouth board and +5 V, +12 V and GND to the Sweet Talker and +5 V to the PCIM-201 display.

Buffering

The higher nibble of the address byte, the control lines of the Z8-bus and pin 4 of Port 3 are buffered with the 74HC244 monodirectional buffer, which is permanently enabled. The multiplexed data and low byte address bus is buffered with a 74HC245 bidirectional transceiver, whch is enabled via the high byte address decoding circuitry. The direction of the data flow is determined by the state of the R/W line. This two-way buffer makes it possible to read the flag register of the PCIM-201 display if desired.

Address decoding and device selection

Output to the peripheral devices has been memory-mapped in the HEX A000 4K address range, which is the only set of locations that is not also decoded by other parts in the present Z8 BASIC Computer/Controller system from Micromint. The lowest two address bits (A0 and A1) select either of four destinations. The output devices of the "Communicaider" have been given the following addresses:

Micromouth's vocabulary DT1057	HEX A000
Micromouth's vocabulary DT1050	HEX A001
Sweet Talker phoneme synthesizer	HEX A002
PCIM-201 alphanumeric LCD	HEX A003

Actually, only 7 of the 16 bits of the address bus are decoded, thus giving many images for each choice of data destination, as seen in the following decoding scheme where "X" represents a "don't care" and "S" is the device selection.

```
A15 A14 A13 A12 A11 A10 A09 A08 A07 A06 A05 A04 A03 A02 A01 A00
---------------|---------------|---------------|---------------
 1   0   1   0 | X   X   0   X | X   X   X   X | X   X   S   S
```

The decoding of A9, enabling the 74HC245 transceiver, makes it possible to add another 512 I/O devices in the AXXX address range of the Z8 system without changing the "Communicaider" interface.

On the circuit level, the high nibble (HEX A) of the high byte of the address bus is decoded to give an active high "board select." This signal is logically ANDed with the also active high "device/chip select" output from the 74HC139 decoder to enable one of the three destinations (the Micromouth is enabled by both addresses A000 and A001; A0 further selects one of the two vocabulary sets (see below)). Because all three peripherals require longer device/chip select times (CS) than the 270 ns long address strobe (AS) from the Z8, each device select is latched with 74HC74 D-type Flip-Flops or otherwise maintained (see below). The choice of active high board and device chip selects was arbitrary. Active low signals could have been chosen as well, but this would not have decreased the number of ICs on the interface board.

Device-specialized circuitry

Due to different control signal and data timing requirements, some special circuitry have been provided for each output device, as follows:

1. The Micromouth's two different sets of vocabularies (DT1050 and DT1056/1057) are selected with A0. Because the state of this address line is of interest at the time the shared bus brings out the data (or even later), A0 is latched with a 74HC74 Flip-Flop. The speech codes for each vocabulary set are stored in two ROMs on the Micromouth board (i.e., all together four ROMs), and address line 13 from the Digitalker chip selects either of the two ROMs in each chip set. To combine this with two possible vocabulary sets, A0Z8 and A13DIGI are decoded together in a 74HC139 circuit to give one out of four possible ROM selects.

2. The Sweet Talker requires at least a 100 μs strobe pulse to alert the SC-01 chip, after which the data is read on the rising edge of that pulse. Even if the execution of a BASIC-statement takes much longer time (in the order of 20 ms per statement), the BASIC Interpreter uses the address/data bus internally before the next statement is executed. Therefore, a 74HC374 Octal Flip-Flop latches the data when the Sweet Talker is addressed and holds it until the next time that happens. A 2.4 μs pulse from a 74HC221 Monostable Multivibrator gates the Z8 Data Strobe (DS) for the pertinent machine cycle only (an extended external memory write cycle takes 2.2 μs with the clock frequency used by the Z8 BASIC System Controller). The 221 circuit also provides the 100 μs strobe pulse.

3. The PCIM-201 needs no data buffering. Only the device/chip select (CS) and memory write (MWR) have to be held for at least 600 ns. However, as the display device has a flag register that can be read, additional circuitry converts the Z8 R/W control signal into two both active low Read and Write signals, as needed by the PCIM-201. The interface thus provides a CS of at least one machine cycle's length (2.170 μs) and MWR/MRD signals that are 1.090 μs long. Finally, a 2N2222 NPN transistor controlled by Z8 Port 3 pin 4 gates power to an LED on the display box, which can be used to alert the receiver of a message.

Timing diagrams

Figures 15 and 16 show the timing relations for the address, data and control signals when the different speech output peripherals are accessed.

Speech device status information

The activity of any of the two speech output devices can be obtained by the Z8 if a Ready/Busy signal on pin 3 of Port 3 is monitored. The INTR signal from the Micromouth and the A/R signal from the Sweet Talker

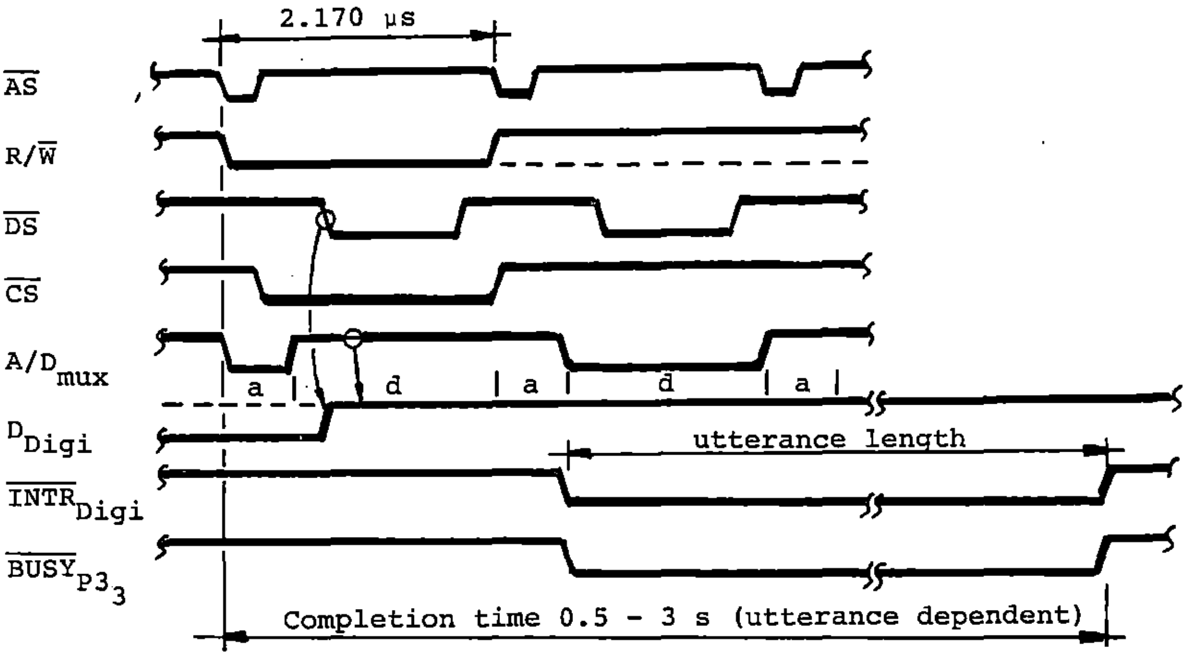

Figure 15 Timing diagram for Micromouth access (address HEX A000 & A001)

could simply be ANDed together and brought to P33. This would be enough for a slowly executing BASIC program. If the devices were to be accessed from an assembly language program, however, the status line from the Sweet Talker would have to go low when the device is addressed and kept low until the phoneme is uttered, so that a currently sounding phoneme would not be "over-written." This security has been provided by an AND Gate and a 74HC74 Flip-Flop that holds the Sweet Talker busy signal. The Preset function of the 74 circuit is used to avoid a lock-up in a low state on power up.

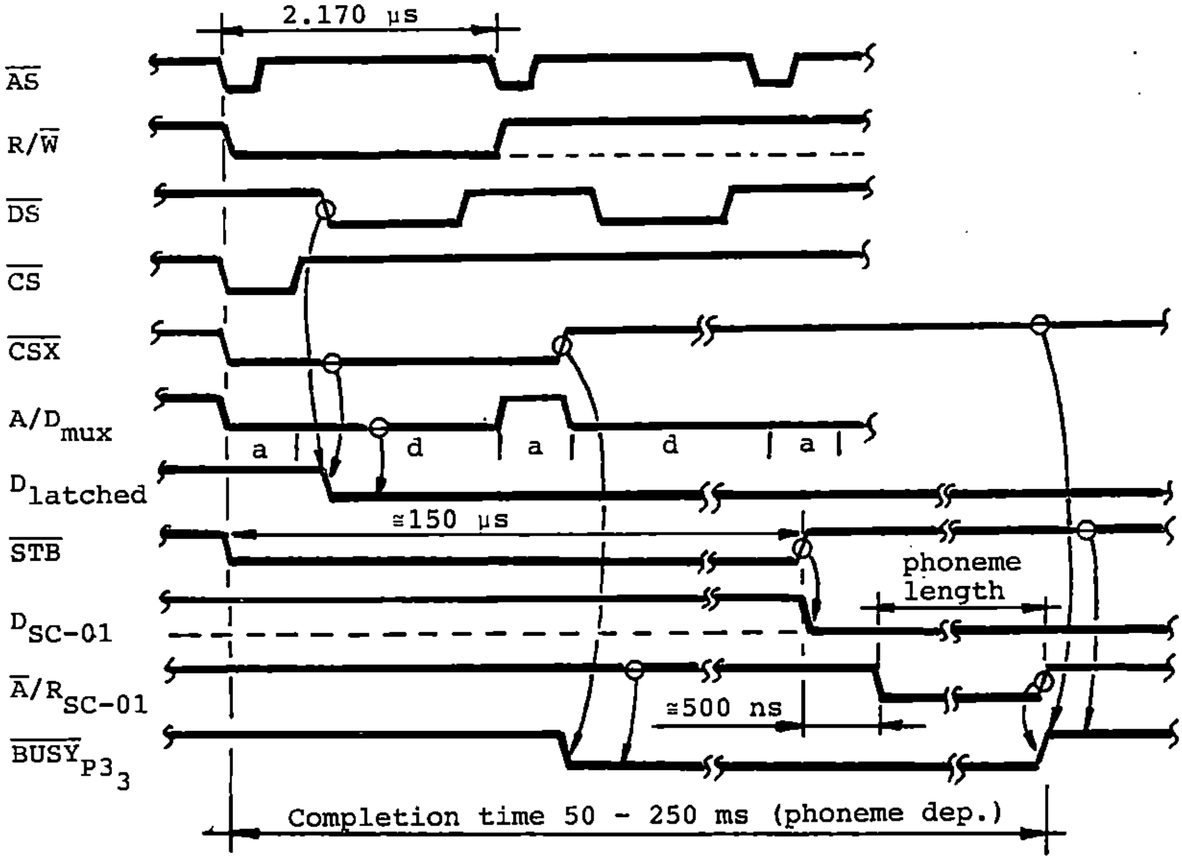

Figure 16 Timing diagram for Sweet Talker access (address HEX A002)

Keyboard

The keyboard is monitored via the Z8 Port 2 that is put in the "open-drain state," i.e., the voltage levels on the pins are determined by the external circuitry that is connected to them. However, the port lines can still sink current up to their maximum capability (2.0 mA). The keyboard is read with the so-called line-reversal technique (Zaks and Lesea (69)), which is a fast a simple method that is especially useful when the keyboard has no common ground line. One port of the computer (or a peripheral interface circuit), in this case Port 2, has to dedicated to the keyboard interface. The only hardware added are eight 4.7 kOhm pullup resistors, as shown in Figure 17, which also shows the line-reversal technique.

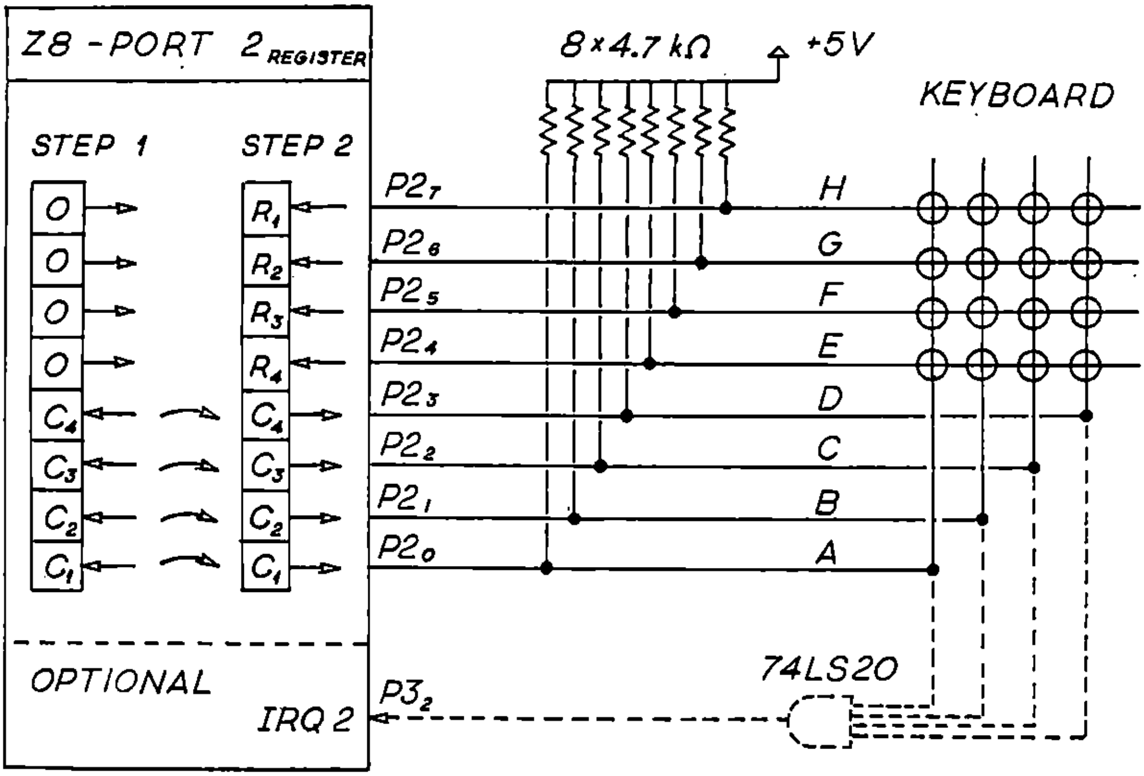

Figure 17 Keyboard connection and reading

Initially, the 8 lines of the computer's Port 2 are configured as 4 input and 4 output lines by loading the proper data pattern into the port direction register (R 246). For example, the keyboard columns are connected to the inputs and the rows to the output lines. If the intial value of the data register is all zeroes, the rows are grounded (the port lines sink the current from the resistors) and a high-level ("ones") voltage appear on the column lines (becauase of the pullup resistors). Whenever a key is pressed on the keyboard, one of the four columns is grounded (step 1). The key closure can be detected either by polling the port or connecting the the column lines to an interrupt line via an AND gate. At this point, the direction of the lines is simply reversed by complementing the contents of the direction register. Only the line for the detected column is now grounded. When subsequently the port is read (step 2), a bit pattern revealing both the row (the high nibble) and the column (the low nibble) of the key is loaded into the port data register. A multiple key closure, i.e., a roll-over problem, can be detected in software when a keyboard code includes "illegal" zeroes (more than two or in an impossible combination).

SYSTEM SOFTWARE AND USER INSTRUCTIONS

Memory Management

The present Z8 computer board from Micromint has two sockets for any combination of supported ROM and RWM. Generally, it is always desirable, when possible, to separate the program and data area for any given system. For this application, it is particularly important to do so as that will allow the user to have different data sets and interchange them easily.

The software for message output from the "Communicaider" is stored in a 2K EPROM to be placed in the HEX 1000-17FF range. Thereby, it will automatically start running on power up, in the way the Z8 computer is configured. The software for data generation, on the other hand, has to be placed on the expansion board in any of the 4K ranges HEX 8000 or 9000, because of its larger program size.

The memory chip for the speech and text messages, however, is always located in the 2K HEX 1800-1FFF range. When data are generated, a RWM chip should be inserted; in operation, the EPROM holding the generated data is placed there instead.

Data Structure

This demonstration version of the "Communicaider" does not allow more than 2 kbytes of data to be stored (see above). Therefore, a primary objective for development of the software and its associated data structure has been not to waste any of this available area. The best way to accomplish this goal is to pack all data contiguously with only a delimiter between the data sets (each being one message), and have pointers to the different sets. The area HEX 1810-18FF has been set aside for these pointers, and in turn has been divided into 3x2 equally sized memory segments. The demonstration unit allows for three different sets of messages (called vocabularies) and the output to be either speech or text. Because of the limited memory bytes available, only 16 (dec) messages of each type can be accommodated, requiring all together 96 data sets (for 3 vocabularies with 16 different messages in text or speech output).

The data sets are stored from location HEX 1900 and up in the order they are generated. The first byte in each set is pointed to by the associated and precisely located 16-bit address value in the pointer array described above. A map of the data memory is shown in Figure 18.

Each data set has a structure as shown in the following example where X takes the value 0-3 (according to the memory-mapped device locations) and Y any HEX value 0-F:

0X YY YY YY YY YY FE 0X YY FE 0X YY YY FF

The first byte (0X) informs the program what output device the message is to start from, a HEX FE selects a change in output device as specified in the following byte, and finally, HEX FF terminates that particular data set. The output routine does not send FE or FF to the peripherals when encountered, but these codes would not be recognized anyway (only Sweet Talker does not read all data bits and sees FF as 3F, which, however, is the code for "silence" at normal pitch setting).

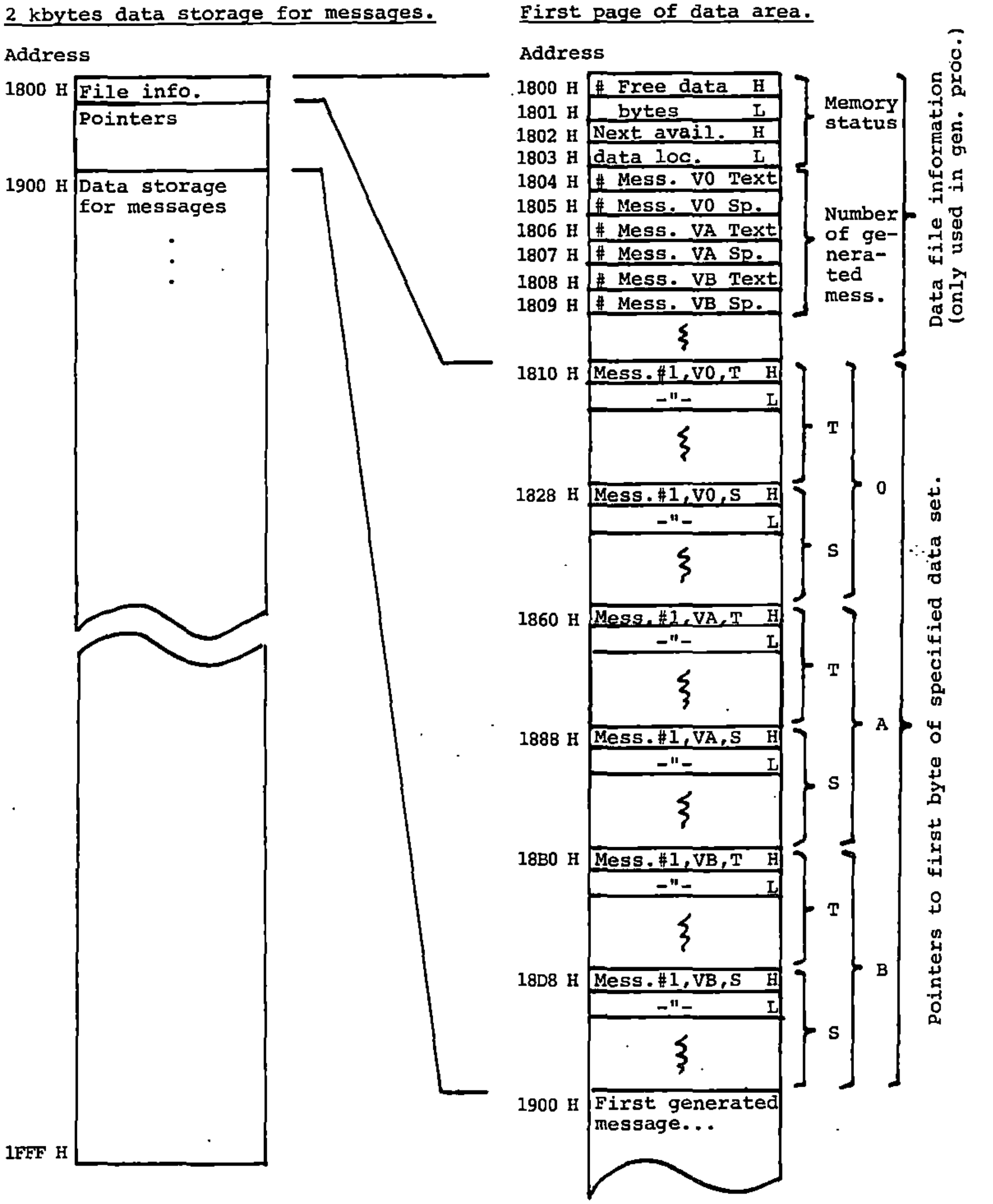

Figure 18 Data memory map

Data Generation Software

The software package for data generation allows the user to create messages for text and voice output and store this data in the most memory-efficient way. Furthermore, data can be easily edited, specific phrases or bytes displayed and changed, given sections of data memory dumped to the CRT monitor or printer, and finally, the current status of the message data area can be shown. The program is written in the compact form supported by the Z8 BASIC interpreter and stored in a 4 kbyte EPROM to be inserted on the expansion board (range HEX 8000 or 9000).

The data generation package is intended to be "user-friendly" and interactive. It prompts the user with a menu of procedures that will be briefly explained in this section. A full listing of the data generation software package appears in Appendix A.

Data area localization

The data for the speech and text messages are to be placed in the HEX 1800-1FFF address range for proper function. However, the package allows the user to select any address range as its target, thus enabling examination and manipulation (where appropriate) of any section of memory. The first location of the target area and its size (in kbytes) is to be specified by the user.

Data area initialization

The first time a data package is to be created, its memory area (HEX 1800, 2K) must be initialized. This procedure loads HEX FF into all locations except the first ten, which are devoted to file information (number of free data bytes, next available byte and counters for the number of messages added to each vocabulary). If the data generation and loading procedure takes place in steps at different times, the data file has to be stored intermediately on tape. When the work is resumed, the data area should, of course, not be initialized.

Data loading sequence

The routines controlling the actual data storage are very interactive and prompt the user for all necessary information. When the vocabulary number, message output type and phrase number have been specified, the location of the pointer for that particular message is calculated. In that two-byte location, the address of the next available free memory byte is stored. Thus, no space is wasted.

The codes for the desired messages are entered in either decimal or hexadecimal form. A HEX FE (dec 254) is used to select a change of output device and HEX FF (dec 255) terminates the input of a particular message. This HEX FF then serves as a delimiter between the data sets stored in memory.

The user has the option of testing the last entered message and deleting it if nonsatisfactory.

Data editing

The entered data can be viewed either phrase by phrase or byte by byte, and a specified data byte can be easily changed. Each time the edited message can be tested until the result is satisfactory. Because of the limited memory space available for the data generation package, however, the routines do not support the deletion of a whole message in one step, nor a memory squeeze or an insertion of a new larger message in the slot of an older, smaller one.

Data dump

Specified sections of memory within the selected area (see above) can be dumped to the CRT monitor or an attached printer. A special cable with a built-in printer enabling relay is needed for an automatic printer dump. The printer baud rate is selected to 300.

Memory status information

Information on the message data file currently being developed can be obtained. In addition to the number of empty memory bytes left and the address of the next empty location, the number of stored messages of each type is shown, i.e., vocabulary 0, A or B and text or speech output.

Message Output Software

The software package that controls the message output gives the user possibility of selecting any of three different vocabularies, the particular message within that vocabulary and the output form (voice or text). A built-in, spoken help-request is also available with one prioritized key-stroke only. The program is stored in a 2K EPROM to be inserted in the HEX 1000-17FF address range. Thus, it starts running automatically on power up. The output routines assume that the message data are located in the HEX 1800-1FFF range. A full listing of the output software is given in Appendix A.

On power up the user is greeted with the message "This is the BME Communicaider." Thereafter, the device waits for an input from the keyboard. On power up or a software reset, vocabulary 0 and speech output are selected by default. Vocabularies A or B are selected with dedicated keys and a return to vocabulary 0 is done by a keyboard "Reset." The output form can be toggled between speech and text with another dedicated key.

Once vocabulary and output form are selected, a certain message is chosen with a two-digit code. With the limited memory storage of the demonstration unit, only messages numbered 1-16 are supported. If the vocabulary and output form are not to be changed in the following message, only its code has to be entered. A number X lower than 10 must be entered as 0X. If a wrong first digit was mistakenly entered it can be changed, but once the second digit is keyed the output begins. The keyboard is read with the so-called line-reversal technique.

The output routine calculates the location of the pointer associated with the selected message. There the address of the first byte of data in that message is to be found. The first byte in a data set holds the code for the output device to be addressed (Micromouth's vocabulary set DT1050 or DT1056/1057, Sweet Talker or PCIM-201) and the following bytes are message data. However, if a HEX FE is encountered, the output device is changed according to the succeeding byte. Finally, upon recognition of a HEX FF, which denotes an end of the message, control is transferred to the keyboard reading routine again.

If a dedicated key (labeled "*") is struck, the "Communicaider" speaks the message "Help, please, help!", preceded by an alarming sequence of tones. This utterance has priority in the software and is output independently of what the output form presently may be set to.

User Instructions

Data generation

In addition to the description of the data generation software, the following instructions should enhance its use:

1. Insert the Data Generation EPROM in any of the expansion board sockets, say HEX 8000, and a 2K RWM in one of the main board sockets. Select its address range to HEX 1800 with the associated jumper. The other socket may be populated or left empty.

2. After power on, set the pointers for start of program and variable storage with the following commands (where % denotes a hexadecimal number):

```
¢8=%8000      { Set program start to HEX 8000 }
@10=%9F       { Locate stack and variable storage in highest
                RWM page }
@12=%9F       { Locate line buffer in highest RWM page }
NEW
```

3. Type RUN and follow the instructions for data generation. Return to BASIC/Debug monitor when done.

4. The first time a set of message data is created, the storage area (HEX 1800-1FFF) has to be initialized. If the procedure is interrupted to be resumed at another time, the hitherto generated data have to be intermediately stored on tape with the commands (see also Z8 Expansion board manual):

```
¢8=%1800
@3A=%4F
GO@%B003,%1800,%1FFF
```

When the procedure is resumed, the data have to be reloaded with the commands:

```
@%32=%FF
GO@%B26C,%1800
```

or

```
GO@%B260        { default loading }
```

5. If the EPROM-board and its programming software package are installed in the system, the data can be immediately stored upon completion of the generation procedure. Otherwise, they have to be stored intermediately on tape as described above.

Message delivery

In addition to the description of the message output software, follow these instructions to deliver the stored messages:

1. Insert the program EPROM in one memory socket addressed HEX 1000 and the data EPROM in the other socket addressed HEX 1800. Install the interface board in the Z8 system and attach the output devices.

2. After the initial greeting message the "Communicaider" is ready to deliver any of the prestored messages. On power on and reset (software or hardware) vocabulary 0 and speech output are enabled. Vocabulary A or B can be selected and the output toggled between speech and text with the respective dedicated keys.

3. A desired message is selected with a two-digit code (between 1 and 16 for this demonstration unit), where a number X less than 10 has to be entered as 0X. Each key has to be released before the next is pushed. A mistake in the first digit can be corrected, but after the second key-stroke has been completed the message output starts.

4. The spoken, prioritized emergency-message "Help, please, help!" is invoked with one key only (labeled "*"), that does not have to be released for the output to start.

A photograph of the assembled demonstration version of the "Communicaider" is shown in Figure 19.

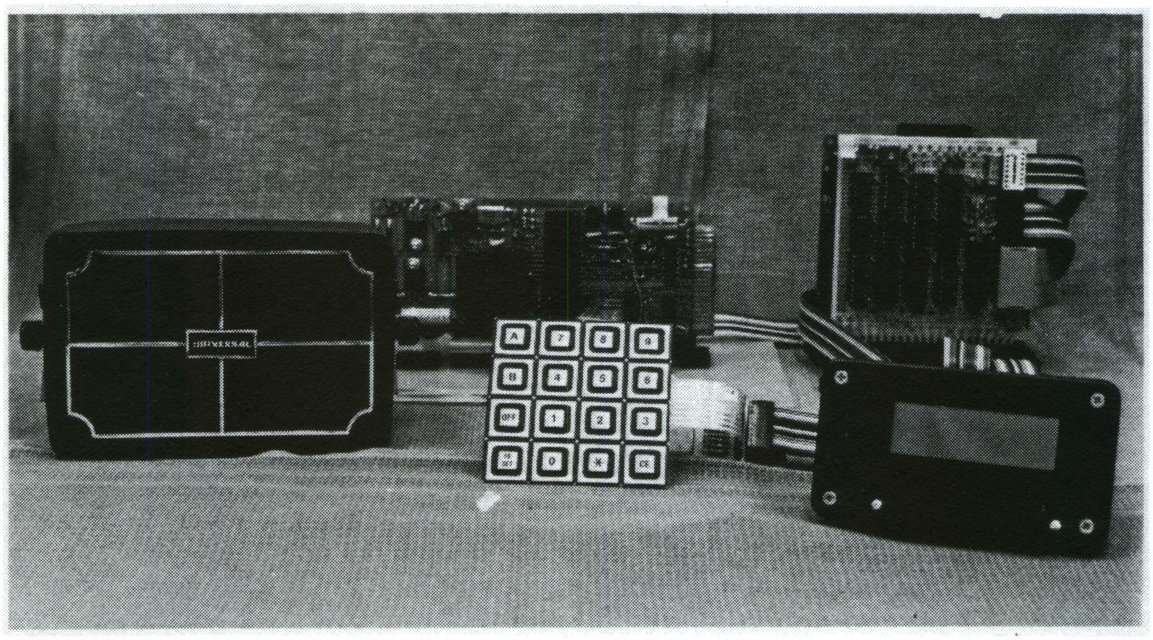

Figure 19 The "Communicaider" demonstration unit photograph

RESULTS AND RECOMMENDATIONS

Experimental Results

This project has dealt with basically two areas of technology: single-chip microcomputers and speech technology. A study of the theory for voice synthesis and recognition has been conducted, including an exhaustive survey of the products of this type that are available on the market. The same type of market examination was done concerning single-chip microcomputers, with special interest in those incorporating BASIC interpreters. The result of this work is found in the appendices, in addition to what is mentioned in the literature review.

On the experimental side, a demonstration unit of a handicap aid for vocally disabled but dextrous people was constructed. The device, called the "Communicaider," fulfills the criteria that were established in the "Design Specification" chapter.

The "Communicaider" is easy to use, once the numbers of the different messages have been memorized. Only a few keystrokes are necessary to produce a message, thus enhancing the communication rate. The voice quality was found to be very good for the Digitalker synthesizer, but not so good for the Votrax device (the option of manually controlled inflection levels was not used). The addition of an LCD alphanumeric display for the option of written messages proved to be an advantageous feature of the device. This could be especially helpful for information where the spelling is important, like the handicapped person's name.

The software routines that have been developed show the possibility of storing program and data separately, conserve data memory as much as possible and give the user the option of having several sets of vocabularies (for example, one for professional use, one for home use and one general). Furthermore, it should be easy for a user to generate his/her own vocabulary, maybe with some help.

The controller of the "Communicaider," the Z8 computer, proved to be a powerful tool for its size and price and relatively easy to use. However, for the control of phoneme-based speech synthesis, the execution speed of interpreted BASIC proved to be too slow, resulting in speech of lower quality. The experiences with integrated circuits from the new low-power HC-family are positive.

In conclusion, it has been shown that if redundant parts of the demonstration version are excluded, a communication aid can be made small in size at a component price of approximately $300.00, much lower than that of present commercially available products. The results of this project should indeed encourage future research and development of these types of handicap aids.

Recommendations for Improvements and Ideas for Future Projects

If future development of the "Communicaider" is to be pursued, by the author or others, the following ideas and recommendations may be of some help:

<u>Technical aspects</u>

1. If software development in BASIC still is desired (for financial or simplicity reasons), stay with the Z8671 as the controlling computer,

but choose a board from another manufacturer that supplies more hardware features as well as software support (for example, the Arcom series (see Appendix B), to which a BASIC compiler including FOR-NEXT loops is also available). The NSC INS8073 offers a better BASIC lanquage but has less hardware features (though a new, improved 16-bit 20 MHz chip including, for instance, A/D, USART and I/O ports is on its way from National for introduction in 1985).

2. Try the new Silicon System phoneme synthesizer chip, the SSI 263, with the powerful text-to-speech-code algorithm that is available for it (see reference 12).

3. Try the second-generation Digitalker II chip (or other LPV synthesizer), that uses predictive coding and needs 400-1000 bits/s.

4. Look for alternative LCD devices and try clip-mounting on a breast pocket and, maybe, wireless transmission of the messages from the hand-held "Communicaider" to the detachable display device.

5. Use a keyboard of higher mechanical quality.

6. Work on improvement of the audio circuitry and search for a good but small loudspeaker.

7. Design the device for battery operation and portability.

8. Use no redundant circuitry in the final design, thus decreasing weight, size, power consumption and cost.

9. Include a telephone adapter to the "Communicaider," so that the vocally impaired person can communicate over the phone as well, especially in case of an emergency.

10. Continue to write the supervising routines in BASIC (for simplicity), but control the actual speech output with assembly language sub-routines (necessary if a synthesizer requiring a higher bit rate, like the SSI 263, is used).

User aspects

1. Make a survey of the phrases most needed by a vocally impaired person.

2. Consider co-operation with a physician/therapist for selection of messages, ergonomic design and practical tests.

3. Increase the possibility even more for the user to easily generate a library of messages of his/her own personal choice, perhaps using the patient's TV as monitor during vocabulary data generation.

BIBLIOGRAPHY

1. Allan, R. 1982. Speech input and output teem with developments. Electronic Design 30, No. 10:67-72.

2. Bond, J. 1983. Two unique methods aid in the recognition of continous speech. Electronic Design 31, No. 11:40-41.

3. Brown, T. J., and R. E. Aitchison. 1982. A Microprocessor Controller for a Personal Typewriter for Visually Handicapped Users. IEEE Transactions on Biomedical Engineering BME-29, No. 7:551-555.

4. Bruckert, E., M. Minow and W. Tetchner. 1983. Three-tiered software and VLSI aid development system to read text aloud. Electronics 56, No. 8:133-138.

5. Bursky, D. 1982. Speech-processor IC gains influence and powerful software. Electronic Design 30, No. 1:40.

6. Carter, J. P. 1983. Electronically Speaking: Computer Speech Generation. Howard Sams & Co., Inc., Indianapolis, IN. 230 pages. ISBN: 0-672-21947-6.

7. Ciarcia, S. 1981. Build a Low-Cost Speech-Synthesizer Interface. Byte 6, No. 6:48-68.

8. Ciarcia, S. 1981. Build a Z8-Based Control Computer with BASIC, Part 1. Byte 6, No. 7:38-47.

9. Ciarcia, S. 1981. Build a Z8-Based Control Computer with BASIC, Part 2. Byte 6, No. 8:50-71.

10. Ciarcia, S. 1981. Build an Unlimited-Vocabulary Speech Synthesizer. Byte 6, No. 9:38-50.

11. Ciarcia, S. 1982. Add Programmable Sound Effects to Your Computer. Byte 7, No. 7:60-72.

12. Ciarcia, S. 1984. Build a Third-Generation Phonetic Speech Synthesizer. Byte 9, No. 3:28-42.

13. Coughlin, R. F., and F. F. Driscoll. 1982. Operational Amplifiers and Linear Integrated Circuits. Prentice-Hall, Inc., Englewood Cliffs, NJ. 376 pages. ISBN: 0-13-637785-8.

14. Dusek, L., T. B. Schalk and M. McMahan. 1983. Voice recognition joins speech on programmable board. Electronics 56, No. 8:128-132.

15. Fields, S. W. 1982. Speech synthesis to be as simple as CP/M. Electronics 56, No. 18:49-50.

16. Flanagan, J. L. 1982. Talking with Computers: Synthesis and Recognition of Speech by Machines. IEEE Transactions on Biomedical Engineering BME-29, No. 4:223-232.

17. Freedman, D. H., and R. Friedman. 1982. Bar-code and voice recognition ease data-entry problems. Mini-Micro Systems 16, No. 7:239-246.

18. Friedlander, C. 1983. Real Time Speech Producing Device. Rocky Mountain Biomedical Symposium, Proc., 1983:111-113.

19. Gallagher, R. T. 1983. Phonemic strategy guides search for multilingual speech-recognition system. Electronics 56, No. 10:88-92.

20. Godin, R. J. 1983. Voice Input and Output, Special Report. Electronics 56, No. 8:126-127.

21. Goldstein, K. 1983. One-chip uC's Become Application Specific. Electronic Engineering Times, April 18/25:47--82.

22. Goodenough-Trepagnier, C., E. Tarry and P. Prather. 1981. Communication Systems for the Non-Vocal Based on Frequent Phoneme Sequences. Journal of Speech and Hearing 24:322-329.

23. Goodenough-Trepagnier, C., and P. Prather. 1982. Derivation of an Efficient Non-Vocal Communication System. Human Factors 24, No. 2:163-172.

24. Helms, G., and S. Petersen. 1982. Portable speech development system creates linear predictive codes. Electronics 55, No. 18:151-156.

25. Kauffman, S. 1981. Displays and Readouts. Electronic Engineering Times, April 27:48-49.

26. Kuecken, J. A. 1983. Talking Computers and Telecommunications. Van Nostrand Reinhold Co., New York, NY. 237 pages. ISBN: 0-442-24721-4.

27. Lerner, E. J. 1982. Products that talk. IEEE Spectrum 19, No. 7:32-37.

28. Lieberman, D. 1983. Data input alternatives. Electronic Products 26, No. 1:47-55.

29. Lineback, J. R. 1983. Voice recognition listens for its cue. Electronics 56, No. 1:110-112.

30. Lubkin, Y. J. 1982. Speech Synthesis and Recognition. Electronic Engineering Times, September 1:33-45.

31. Lubkin, Y. J. 1983. Speech technology: Still a bright future. Electronic Engineering Times, July 4:17-31.

32. Micromint Inc. 1981. The Micromouth Speech Synthesizer Interface, Owner's Manual. Micromint Inc., 917 Midway, Woodmere, NY.

33. Micromint Inc. 1981. Z8 BASIC Computer/Controller, User's Manual. Micromint Inc., 561 Willow Ave., Cedarhurst, NY.

34. Micromint Inc. 1982. Z8 EPROM Promgrammer, User's Manual. Micromint Inc., 561 Willow Ave., Cedarhurst, NY.

35. Micromint Inc. 1982. Z8 Expansion Board, User's Manual. Micromint Inc., 561 Willow Ave., Cedarhurst, NY.

36. Micro Projects. 1982. PSC-1 Programmable Sound Generator, User's Manual. Micro Projects, P. O. Box 420, South Windsor, CT.

37. Motorola Inc. 1979. Linear Integrated Circuits. Motorola Semiconductor Products Inc., Box 20912, Phoenix, AZ.

38. National Semiconductor Corp. 1980. MM54104 Digitalker Speech Synthesis System, Data Sheet. National Semiconductor Corp., 2900 Semiconductor Dr., Santa Clara, CA.

39. National Semiconductor Corp. 1980. Linear Applications Handbook. National Semiconductor Corp., 2900 Semiconductor Dr., Santa Clara, CA.

40. National Semiconductor Corp. 1978. Linear Databook. National Semiconductor Corp., 2900 Semiconductor Dr., Santa Clara, CA.

41. Nicholosi, L., E. Harryman and J. Kresheck. 1983. Terminology of Communication Disorders. William & Wilkins, Baltimore, MD. 319 pages. ISBN: 0-683-06499-1.

42. Nicholson, B. 1983. BASIC-equipped single-board computers function as low-cost controllers. EDN Magazine 28, No. 3:77-82.

43. Parikh, D., S. Charbonneau and S. Barber. 1983. Single-chip speech synthesizers speak well of their algorithms. Electronics 56, No. 15:135-139.

44. Printed Circuits International, Inc. 1983. PCIM-201 Data Sheet. Printed Circuits International, Inc., 1145 Sonora Court, Sunnyvale, CA.

45. Rastgar, F., and M. Devaney. 1983. Speech Recognition for Wheelchair Control. Rocky Mountain Biomedical Symposium, Proc., 1983:109-110.

46. RE-AL, Inc. 1982. RE-AL Tacti-Membrane 66(xy) Data Sheet. RE-AL, Inc., 620 Wheeling Rd., Wheeling, IL.

47. Rosen, M. J., and C. Goodenough-Trepagnier. 1981. Factors Affecting Communication Rate in Non-Vocal Communication Systems. Proc. 4th Annual Conference on Rehabilitation Engineering, 1981:194-196.

48. Rosen, M. J., and C. Goodenough-Trepagnier. 1982. Communication Systems for the Non-Vocal Motor Handicapped: Practise and Prospects. (IEEE) Engineering in Medicine and Biology Magazine 1, No.4:31-35.

49. Schalk, T. B., and E. L. Van Meir. 1983. Terminals, Listen Up, Speech Recognition is a Reality. Computer Design 22, No. 10:97-102.

50. Schalk, T. B., G. A. Frantz and L. Woodson. 1982. Voice synthesis and recognition. Mini-Micro Systems 15, No.12:146-160.

51. Scott, B. L. 1983. Voice Recognition Systems and Strategies. Computer Design 22, No. 1:67-70.

52. Secrest, B., M. Arjmand and M. Ni. 1982. Speech analysis and synthesis become practical on uC chip. Electronic Design 30, No.11:129-136.

53. Shepard, J. D. 1982. LCD/LED Displays. Electronic Engineering Times, September 27:65-66.

54. Teja, E. R. 1982. Voice-output units show improvement in speech quality, vocabulary, price. EDN Magazine 27, No.11:71-80.

55. Teja, E. R. 1982. Designers of voice-recognition systems target speaker-independent, pause-free input. EDN Magazine 27, No. 17:67-72.

56. Teja, E. R. 1983. Speaking of voice I/O... EDN Magazine 28, No. 10:16.

57. Teja, E. R. 1983. Voice-output products see expanding use. EDN Magazine 28, No. 15:54-65.

58. Teja, E. R. 1983. Advancements in recognition methods boost voice-input-device performance. EDN Magazine 28, No. 20:45-54.

59. Texas Instruments Inc. 1981. The TTL Data Book for Design Engineers. Texas Instrument Inc., P.O. Box 225012, Dallas, TX.

60. Tompkins, W. J., and J. G. Webster. 1981. Design of Microcomputer-Based Medical Instrumentation. Prentice-Hall, Inc., Englewood Cliffs, NJ. 435 pages. ISBN: 0-13-201244-8.

61. Vanderheiden, G. C. 1978. Non-Vocal Communication Resource Book. University Park Press, Baltimore, MD. ISBN: 0-8391-1252-1.

62. Vartanian, M. M., and S. W. Groesberg. 1978. A microprocessor-based communicator for aphonic persons. Proc. 20th Annual Conference on Engineering in Medicine and Biology, 1978:319.

63. Votrax. 1980. SC-01 Speech Synthesizer Data Sheet. Votrax Division of Federal Screw Works, 500 Stephenson Highway, Troy, MI.

64. Ward, B., and T. Subandhi. 1983. A Computer Terminal for the Visually Impaired. IEEE Transactions on Biomedical Engineering. BME-30, No. 8:545.

65. Wickersham, F., G. Mullinix and S. McGinness. 1982. Bringing speech to low-volume products. Mini-Micro Systems 15, No.6:219-221.

66. Wickey, D., and L. Kane. 1982. Match vocabulary, speech quality to your voice-output application. EDN Magazine 29, No. 17:185-192.

67. Wilson, D. 1983. Processors Shape Trends in Speech Synthesis, Voice Recognition. Digital Design 13, No. 10:66-73.

68. Witten, I. H. 1982. Principles of Computer Speech. Academic Press, New York, NY. 286 pages. ISBN: 0-12-760760-9.

69. Zaks, R., and A. Lesea. 1979. Microprocessor Interfacing Techniques. Sybex Inc., 2344 Sixth St., Berkeley, CA. 456 pages. ISBN: 0-89588-029-6.

70. Zilog, Inc. 1981. Z8671 Single-Chip BASIC Interpreter, BASIC/DEBUG Software Reference Manual. Zilog, Inc., 1315 Dell Dr., Campbell, CA. Zilog publication ref. no.: 03-3149-02.

71. Zilog, Inc. 1983. Z8 Microcomputer, Technical Manual. Zilog, Inc., 1315 Dell Dr., Campbell, CA. Zilog publication ref. no.: 03-3047-03.

ACKNOWLEDGEMENTS

Initially, I would like to express my deep appreciation for the existence of the Fulbright Program for Educational Exchange between Sweden and the United States, which made my studies at Iowa State University possible.

Further, I owe much gratitude to my major professor, Dr. Curran S. Swift, for his gentle and patient but very inspiring and enthusiastic guidance throughout this project. Thank you and good luck with future work where this study might be of some help.

I would also like to thank the two other members of my committee; Dr. David L. Carlson for many helpful hints concerning electronic design, and Dr. Richard E. Horton for giving me a good start in the area of microcomputers.

This document has been created using a PDP-11 minicomputer and its word processing capability. For the assistance from a "wizard" possessing much experience with this machine, and for his cheerful comments during many long nights in front of the terminal, I would like to thank my fellow BME student Paul Antol, who graduated together with me. However, many other students as well and members of the BME faculty have made my time in Ames a pleasantly memorable and learning experience.

Finally, but not least; studies abroad are much eased by good "ground service" at home. For best practical support in this matter and belief in me, I therefore heartfully thank my parents, Per and Margot Rolander, Sölvesborg, Sweden.

APPENDIX A

SYSTEM SOFTWARE

Listing of program for message output routines

COMMENTS: Line numbers ending in X are not stored in the runtime EPROMs.
Circumflex comes out on the printer as cent sign (¢).

```
100X  REM  ***************************************************************
100X  REM  - VARIABLES -

100X  REM  A = Memory (base) address for output devices
100X  REM  B = Memory (base) address for output data (Data ROM location)
100X  REM  C = Character input from keyboard
100X  REM  D = Data byte in memory
100X  REM  E = Erase routine (Variables = N, P and Q)
100X  REM  F = Fill routine (Variables = N, P and Q)
100X  REM  G = Change output/vocabulary routine (Variables = C, V and S)
100X  REM  H = Halt (delay) routine
100X  REM  I = Output device in greeting message
100X  REM  J = Variable in delay routine
100X  REM  K = Keyboard reading routine (Variables = C and Q)
100X  REM  L = Current line number register
100X  REM  M = Memory data byte
100X  REM  N = Number of keystrokes
100X  REM  O = Output device
100X  REM  P = Phrase number (= entered code)
100X  REM  Q = Local variable in keyboard routine
               and last character entered
100X  REM  R = Ready/Busy check routine
100X  REM  S = Speech flag (speech output ON/OFF)
100X  REM  T = Output routine
100X  REM  U = Local variable in keyboard routine
100X  REM  V = Vocabulary number (Default, A or B)
```

```
100X  REM  ***************************************************************
100X  REM  - INITIALIZATION ROUTINE -

100   REM EMPTY BYTES HEX1000-20
101   @%A001=%47: @%A002=%3F: @%A003=0  { First line number < 255 }
1000  E=1280: F=1260: G=1270: H=1250: I=1003
1001  K=1230: R=1200: T=1210: W=1250
1002  A=%A000: B=%1800: L=%16: @33=0
1003  N=0: V=%10: S=1: P=0
1004  @247=%40: @3=0
1005  O=A+3: @O=%8B: @O=%70: @O=%6E: @O=%6A: @O=%68:
1006  @O=%65: @O=%67: @O=%62: @O=%8A: @O=%00
1007  IF @33=1 GOTO 1500

110X  REM  ***************************************************************
110X  REM  - GREETING MESSAGE ON POWER UP -

1100  J=10: GOSUB H
1101  @3=%10
1102  @I=%76: @O=%54: @O=%48: @O=%49: @O=%53: GOSUB R
1103  I=I+1
1104  @I=%44: @O=%20: GOSUB R
1105  @I=%60: @O=%49: @O=%53: GOSUB R
1106  @I=%44: @O=%20: GOSUB R
1108  @I=%8A: @O=%54: @O=%48: @O=%45: GOSUB R
1109  @I=%44: @O=%20: GOSUB R
1111  @I=%21: @O=%42: GOSUB R
1113  @I=%2C: @O=%4D: GOSUB R
1115  @I=%24: @O=%45: GOSUB R
1116  @I=%46: @O=%20: GOSUB R
1117  I=A+2
1118  @I=%19: @O=%43: GOSUB R
1119  @I=%31: @O=%4F: GOSUB R
1120  @I=%0C: @O=%4D: @O=%4D: GOSUB R
1121  @I=%22: GOSUB R
1122  @I=%36: GOSUB R
1123  @I=%37: @O=%55: GOSUB R
1124  @I=%0D: @O=%4E: GOSUB R
1125  @I=%09: @O=%49: GOSUB R
1126  @I=%19: @O=%43: GOSUB R
1127  @I=%06: @O=%41: GOSUB R
1128  @I=%21: GOSUB R
1129  @I=%29: @O=%49: GOSUB R
1130  @I=%1E: @O=%44: GOSUB R
1131  @I=%3A: @O=%45: @O=%52: GOSUB R
1132  @I=%3F: @O=%2E: @O=%2E: @O=%2E: GOSUB R
1140  J=20: GOSUB H
1141  @O=%63: @3=0: @33=1
1142  GOTO 1500
```

```
120X  REM  ****************************************************************
120X  REM  - READY/BUSY CHECK ROUTINE (SUB R) -

1200  IF AND(@3,8)<>8 GOTO ¢(L)
1201  RET

121X  REM  ****************************************************************
121X  REM  - OUTPUT ROUTINE (SUB T) -

1210  M=¢(B+V+S*40+(P-1)*2)
1211  O=A+@M: M=M+1: IF O<>A+3 GOTO ¢(L)+2
1212  @3=%10: @O=%8A: @O=0: @O=%62
1213  D=@M: M=M+1: IF D=%FE GOTO ¢(L)-2
1214  IF D=%FF GOTO ¢(L)+3
1215  @O=D: IF O<>A+3 GOSUB R
1216  GOTO ¢(L)-3
1217  @3=0
1218  RET

123X  REM  ****************************************************************
123X  REM  - KEYBOARD READING ROUTINE (SUB K) -

1230  @247=%40: U=0
1231  @246=%0F: @2=0: C=@2: IF C=%0F GOTO ¢(L)
1232  @246=%F0: @2=C: C=%FF-@2
1233  IF C=%21 GOTO ¢(L)+10
1234  IF C=%41 GOTO ¢(L)+3
1235  IF AND(C,%81)<>0 GOTO ¢(L)+7
1236  U=AND(C,%0E): Q=AND(C,%70)/16: IF Q=4 Q=3
1237  C=0: IF U=0 GOTO ¢(L)+4
1238  IF U=2 C=4-Q
1239  IF U=4 C=7-Q
1240  IF U=8 C=10-Q
1241  Q=C
1242  @246=%0F: @2=0: IF @2<>%0F GOTO ¢(L)
1243  RET

125X  REM  ****************************************************************
125X  REM  - DELAY ROUTINE (SUB H) -

1250  J=J-1: IF J>0 GOTO ¢(L)
1251  RET

126X  REM  ****************************************************************
126X  REM  - FILL ROUTINE (SUB F) -

1260  P=P*10+C: N=N+1
1261  RET
```

```
127X  REM  ***************************************************************
127X  REM  - CHANGE ROUTINE (SUB G) -

1270  IF C<>%82 GOTO ¢(L)+4
1271  IF S=0 GOTO ¢(L)+2
1272  S=0: GOTO ¢(L)+4
1273  S=1: GOTO ¢(L)+3
1274  IF C=%84 V=%B0
1275  IF C=%88 V=%60
1276  RET

128X  REM  ***************************************************************
128X  REM  - ERASE ROUTINE (SUB E) -

1280  P=(P-Q)/10: N=N-1
1281  RET

130X  REM  ***************************************************************
130X  REM  - "HELP" PRIORITIZED OUTPUT (SUB H)

1300  I=A+1
1301  @I=%41: GOSUB R
1302  @I=%42: GOSUB R
1303  @I=%41: GOSUB R
1304  @I=%42: GOSUB R
1305  @I=%41: GOSUB R
1306  @I=%42: GOSUB R
1307  J=3: GOSUB H
1308  @A=%3A: GOSUB R
1309  @I=%78: GOSUB R
1310  @A=%3A: GOSUB R
1311  RET

150X  REM  ***************************************************************
150X  REM  - MAIN PROGRAM -

1500  GOSUB K
1501  IF C=%21 GOSUB 1300
1502  IF C=%81 GOTO 1003
1503  IF C>=%82 GOSUB G
1504  IF C=%11 GOSUB E
1505  IF C<=9 GOSUB F
1506  IF N<2 GOTO ¢(L)-6
1507  IF P<21 GOSUB T
1508  N=0: P=0: GOTO ¢(L)-8
```

Listing of program for message data generation and storage

```
200X  REM  ***************************************************************
200X  REM  - VARIABLES -

200X  REM  A = Append data routine
200X  REM  B = Base of data memory
200X  REM  C = Input character/number
200X  REM  D = Input/output data byte
200X  REM  E = Erase screen code (ASCII character %1A or %9A)
200X  REM  F = Fill memory routine
200X  REM  G = Speech source selection routine
200X  REM  H = Vocabulary selection routine
200X  REM  I = Input driver
200X  REM  J = Local variable in data loading routine
200X  REM  K = Limit of memory dump
200X  REM  L = Current line number register
200X  REM  M = Memory byte number
200X  REM  N = Counter
200X  REM  O = Output driver
200X  REM  P = Message number (1-16)
200X  REM  Q = Continuation questioning routine
200X  REM  R = Range fault on input message
200X  REM  S = Speech/text flag (0 or 1)
200X  REM  T = Text/speech selection routine
200X  REM  U = Phrase number selection routine
200X  REM  V = Vocabulary offset (%10, %60 or %B0)
200X  REM  W = Data memory size
200X  REM  X = Test message routine
200X  REM  Y = Busy check routine
200X  REM  Z = Message output device
```

```
200X  REM  ************************************************************
200X  REM  - VARIABLE AND REGISTER INITIALIZATION -

2000  E=%1A: I=%54: L=%16: O=%61: @33=0: @247=%40: @%A002=%3F
2001  A=2270: F=2290: G=2250: H=2221: Q=2050
2002  R=2060: T=2231: U=2241: X=2600: Y=2610

201X  REM  ************************************************************
201X  REM  - MAIN PROGRAM FOR DATA GENERATION AND EDITING -

2006  GO@O,E: "":"DATA GENERATION PROCEDURE":""
2007  IF @33=1 GOTO ¢(L)+3
2008  "Data base  address = ";:INPUTB:"Data size (kbytes) = ";:INPUTW
2009  @33=1: W=W*%400: GOTO ¢(L)-3
2010  "   0. Re-locate data area."
2011  "   1. Initialize data area."
2012  "   2. Load text/speech data."
2013  "   3. Show/alter data."
2014  "   4. Dump data to CRT/printer."
2015  "   5. Display memory status."
2016  "   6. Return to monitor.":"": GOTO ¢(L)+2
2017  GOSUB R
2018  "Choice ";: INPUT C
2019  IF C<0 GOTO ¢(L)-2
2020  IF C>6 GOTO ¢(L)-3
2021  IF C=0 GOTO 2000
2022  IF C=6 STOP
2023  GOSUB(2000+C*100)
2024  GOTO ¢(L)-18

205X  REM  ============================================================
205X  REM  - CONTINUATION QUESTIONING ROUTINE (SUB Q) -

2050  "": "¢G": "Done! Hit <RETURN> to continue.": C=USR(I): ""
2051  IF C<>%0D GOTO ¢(L)-1
2052  RET

206X  REM  ============================================================
206X  REM  - SUB R -

2060  "": "¢G": "Selection is out of range. Try again...":""
2061  RET
```

```
210X  REM  ***************************************************************
210X  REM  - DATA AREA INITIALIZATION -

2100  N=0:"":"Wait 2 min."
2101  @(B+N)=%FF: N=N+1: IF N<W GOTO ¢(L)
2102  ¢B=W-%FF: ¢(B+2)=B+%100: N=4
2103  @(B+N)=0: N=N+1: IF N<10 GOTO ¢(L)
2104  GOSUB Q
2105  RET

220X  REM  ***************************************************************
220X  REM  - DATA LOADING ROUTINE -

2200  GO@O,E: "":"DATA LOADING ROUTINE":""
2201  "": "Specify message:": ""
2202  GOSUB H
2203  GOSUB T
2204  GOSUB U
2205  M=¢(B+2): ¢(B+V+S*40+P*2-2)=M: ¢(B+10)=M
2206  D=%03: IF S=1 GOSUB G
2207  GOSUB F
2208  GOSUB A
2209  GOSUB X
2210  "": "Delete entry ? ";:IF USR(I)<>%59 GOTO ¢(L)+2
2211  ¢B=¢B+¢(B+2)-¢(B+10): GOTO ¢(L)+2
2212  J=B+(V-16)/40+S+4: @J=@J+1
2213  "": "Continue ? ";: IF USR(I)=%59 GOTO ¢(L)-13
2214  RET

222X  REM  ===============================================================
222X  REM  - SUB H -

2220  GOSUB R
2221  "   Vocabulary (0,A or B) ? ";: C=USR(I): ""
2222  IF C<>%30 IF C<>%41 IF C<>%42 GOTO ¢(L)-2
2223  IF C=%30 V=%10
2224  IF C=%41 V=%60
2225  IF C=%42 V=%B0
2226  RET

223X  REM  ===============================================================
223X  REM  - SUB T -

2230  GOSUB R
2231  "   Type (T or S) ? ";: C=USR(I): ""
2232  IF C<>%53 IF C<>%54 GOTO ¢(L)-2
2233  IF C=%53 S=1
2234  IF C=%54 S=0
2235  RET
```

```
224X  REM  ==============================================================
224X  REM  - SUB U -

2240  GOSUB R
2241  "   Number ";: INPUT P: ""
2242  IF P<1 GOTO ¢(L)-2
2243  IF P>20 GOTO ¢(L)-3
2244  RET

225X  REM  ==============================================================
225X  REM  - SUB G -

2250  "": Speech output source:": ""
2251  "   1. Digitalker DT1050."
2252  "   2. Digitalker DT1057."
2253  "   3. Votrax phoneme set.": GOTO ¢(L)+2
2254  GOSUB R
2255  "":"Choice ";: INPUT C
2256  IF C<>1 IF C<>2 IF C<>3 GOTO ¢(L)-2
2257  IF C=1 D=%01
2258  IF C=2 D=%00
2259  IF C=3 D=%02
2260  RET

227X  REM  ==============================================================
227X  REM  - SUB A -

2270  "": "Enter message data:": ""
2271  "   (%FF (255) to terminate; %FE (254) to change speech source)"
2272  "": GOTO ¢(L)+3
2273
2274  GOSUB R
2275  "   ",: INPUT D
2276  IF D<0 GOTO ¢(L)-2
2277  IF D>255 GOTO ¢(L)-3
2278  GOSUB F
2279  IF D=%FF GOTO ¢(L)+6
2280  IF D=%FE GOTO ¢(L)+2
2281  GOTO ¢(L)-6
2282  GOSUB G
2283  GOSUB F
2284  GOTO ¢(L)-12
2285  RET
```

```
229X  REM  ============================================================
229X  REM  - SUB F -

2290  ¢M=D: M=M+1: ¢B=¢B-1: ¢(B+2)=¢(B+2)+1: IF M<B+W GOTO ¢(L)+2
2291  "": "¢G": "Memory full!":""
2292  RET

230X  REM  ************************************************************
230X  REM  - DATA EDITING ROUTINE -

2300  GO@O,E: "":"DATA EDITING ROUTINE":""
2301  "   1. Show specific message and test."
2302  "   2. Show/alter specific memory byte and test."
2303  "   3. Quit.":"": GOTO ¢(L)+2
2304  GOSUB R
2305  "Choice ";: INPUT C
2306  IF C<>1 IF C<>2 IF C<>3 GOTO ¢(L)-2
2307  IF C=3 GOTO ¢(L)+3
2308  GOSUB(2300+C*20)
2309  GOTO ¢(L)-9
2310  RET

232X  REM  ============================================================
232X  REM  - MESSAGE DISPLAY ROUTINE -

2320  "": "Specify message!":""
2321  GOSUB H
2322  GOSUB T
2323  GOSUB U
2324  M=¢(B+V+S*40+P*2-2): ¢(B+10)=M: K=M: "": "Location       Data": ""
2325  ""; HEX(D),"",: N=0
2326  D=@M: IF D<%10 "0";
2327  ""; HEX(D); "  ";: N=N+1: M=M+1: IF D<>%FF IF N<16 GOTO ¢(L)-1
2328  IF D<>%FF GOTO ¢(L)-3
2329  "": GOSUB X
2331  GOSUB Q
2332  RET

234X  REM  ============================================================
234X  REM  - BYTE EDITING ROUTINE -

2340  "": "Which memory byte ";: INPUT M
2341  "": "Data in byte "; HEX(M); " is HEX "; HEX(@M); "."
2342  "": "Should data be changed ?";: IF USR(I)<>%50 GOTO ¢(L)+2
2343  "": "New data ";: INPUT D: @M=D
2344  GOSUB X
2345  "": "Continue ? ";: IF USR(I)=%59 GOTO ¢(L)-5
2346  RET
```

```
240X  REM  **************************************************************
240X  REM  - DATA DUMP ROUTINE -

2400  GO@O,E: "":"DATA DUMP ROUTINE":""
2401  GOSUB R
2402  "Start ";: INPUT M
2403  "End   ";: INPUT K
2404
2405
2406  IF M<=K GOTO ¢(L)+2
2407  "¢G": "Error: Start < End. Try again...": "": GOTO ¢(L)-4
2408  "": "Type 'P' for a hardcopy. ";: IF USR(I)<>%50 GOTO ¢(L)+4
2409
2410  @247=%40: @3=%20: @244=64: GO@O,%0C: N=30
2411  N=N-1: IF N>0 GOTO ¢(L)
2412  "":"": "Location      Data"
2413  N=0: GO@O,%0A: GO@O,%0D: ""; HEX(M),"",
2414  D=@M: IF D<%10 "0";
2415  ""; HEX(D); "  ";: N=N+1: M=M+1: IF M>K GOTO ¢(L)+3
2416  IF N=16 GOTO ¢(L)-3
2417  GOTO ¢(L)-3
2418  @244=Y: @3=0
2419  GOSUB Q
2420  RET

250X  REM  **************************************************************
250X  REM  - MEMORY STATUS DISPLAY ROUTINE -

2500  GO@O,E: "":"MEMORY STATUS":""
2501  "Empty bytes = "; ¢B; "(dec)"
2502  "Next empty location = "; HEX(¢(B+2)); "(HEX)"
2503  "":"Number of stored messages": ""
2504  "","","Text","Speech":""
2505  "Vocabulary 0",@(B+4),@(B+5)
2506  "Vocabulary A",@(B+6),@(B+7)
2507  "Vocabulary B",@(B+8),@(B+9)
2508  GOSUB Q
2509  RET

260X  REM  **************************************************************
260X  REM  - MESSAGE TEST ROUTINE -

2600  M=¢(B+10): "": "Message test ? ";: IF USR(I)<>%59 GOTO ¢(L)+8
2601  Z=%A000+@M: M=M+1: IF Z<>%A003 GOTO ¢(L)+2
2602  @Z=%65: @Z=%67: @Z=%62: @Z=%8A: @Z=0: @3=%10
2603  D=@M: M=M+1: IF D=%FE GOTO ¢(L)-2
2604  IF D=%FF GOTO ¢(L)+3
2605  @Z=D: IF Z<>%A003 GOSUB Y
2606  GOTO ¢(L)-3
2607  @3=0
2608  RET
```

```
261X  REM  ***************************************************************
261X  REM  - READY/BUSY CHECK ROUTINE

2610  IF AND(@3,8)<>8 GOTO ¢(L)
2611  RET
```

APPENDIX B

SINGLE-CHIP MICROCOMPUTERS WITH BASIC

Single-chip microcomputers with on-chip BASIC interpreters

National Semiconductor: INS8073

Manufacturer	National Semiconductor Corp.
Name	INS8073 NSC with Tiny BASIC Microinterpreter (a member of the 8070 family)
Address bus	o 16-bit unshared bus for 64K address space. o TTL-compatible lines with TRISTATE mode for DMA operation.
Data bus	o 8-bit unshared bidirectional bus. o TTL-compatible lines with TRISTATE mode for DMA operation.
I/O	o Memory-mapped over the 8-bit parallel data bus only. o No serial I/O in hardware. The BASIC Microinterpreter has a serial I/O routine (using flag F1 for output and SA/INTA interrupt line for input) with baud rates 110, 300, 1200 and 4800. o Three special flag output lines. o All lines can sink 1.6 mA and source 100 uA, all with static charge protection. o I/O controll lines are NRDS, NWDS, NBREQ, NENIN and NENOUT.
Memory	2.5 K ROM (populated with the BASIC interpreter) and 64 bytes of RWM on-chip. In addition to using all 64 bytes, BASIC requires at least an additional 256 bytes external RWM for buffers and variable storage.
Speed	o On-chip clock signal generation for 4, 8 or 16 MHz (1 MHz is minimum). o Assembly instruction execution time averaging 2.2 us (4.5 us at the most) with 4 MHz system clock. o External memory write cycle 1.75 us @ 4 MHz system clock. o The processor can be halted (indefinitely) to fascilitate DMA operations over the MICROBUS or access slow peripherals.

Special features — 16-bit hardware Multiply (37 us) and Divide (47 us) instructions.

Assembly programs

- o Programming model: program counter PC (16-bit), stack pointer SP (16-bit), user stack pointers P1 & P2 (16-bit), accumulator AC (8-bit), extended register ER (8-bit) that can be concatenated with AC, temporary register TR (16-bit) and status register SR (8-bit).
- o 192 different instruction, for instance
- o 8 or 16-bit signed integer arithmetic, logic and stack operations.
- o Single instruction character search.
- o Single instruction ASCII to decimal conversion.
- o 2 vectored, hardware prioritized (A>B) interrupt lines.

Package — 40-pin DIP for operating temperature 0-70°C in comercial version.

Power

- o +5 V @ 100 mA (max -0.5 - 7.0 V).
- o No power-down mode.
- o Fabricated in NMOS technology.

BASIC Interpreter

- o Variables A-Z.
- o 16-bit signed integer arithmetic.
- o Multiple statements on the same program line suported.

- o Command summary:
  NEW expr { establishes pointers for new program (old progam saved) }
  NEW { resets program pointers (old program overwritten) }
  RUN { runs current program }
  CONT { continue execution after break point }
  LIST [expr] { list current program [from line #] }

- o Statement summary:
  REM anything { comments }
  CLEAR { clear variables, disable interrupts, reset stack }
  [LET] var=expr { assignment }
  [LET] STAT=expr { change status register }
  [LET] @factor=expr { store value in memory byte }
  [LET] $factor="string" { store ASCII characters in memory bytes }
  [LET] $factor=$factor { copy memory to memory }
  PRINT expr { print value }
  PRINT "string" { print ASCII string }
  IF expr [THEN] statements { conditional execution }
  FOR var=expr TO expr [STEP expr] { FOR loop

execution (four nested loops OK) }
NEXT var { FOR loop termination }
DO { DO loop initiation (eight nested loops OK) }
UNTIL expr { DO loop termination }
GO TO expr { program control transfer }
GO SUB expr { BASIC subroutine call }
RETURN { return from subroutine }
INPUT var { read value from console }
INPUT $factor { read ASCII string from console }
LINK expr { call assembly routine }
ON 1 or 2 expr { one-level interrupt proceesing in BASIC }
DELAY expr { delay 1-1040 ms }
STOP { terminate program execution }

o Function summary:
AND, OR, NOT { logical operations }
INC(x), DEC(x) { increment or decrement memory location }
MOD(x/y) { modulus (remainder of division) function }
RND(x,y) { random number in interval x,y }
TOP { next available memory byte for new program }
STAT { status register }
@factor { absolute memory/peripheral addressing }

Price $30.00

<u>Zilog: Z8671</u>

Manufacturer	Zilog Inc.
Product	Z8671 with BASIC/Debug Interpreter
Address bus	o 8 or 16-bit for up to 64K memory (62K externally). o Lower byte (if 16-bit address bus) time-shared with data bus. o No TRISTATE mode (for DMA and multiprocessing).
Data bus	8-bit bidirectional time-shared bus on port that can alternative be configured as general I/O port.
I/O	o 32 programmable TTL-compatible I/O lines that can be grouped into four 8-bit ports, of which one can be configured for control of the others (e.g., handshake). I/O can alternatively be memory-mapped, if Port 0 & 1 (or only Port 1) is configured as address bus. o Serial I/O in hardware for full duplex UART with baud rate 110 - 9600, with or without handshake. o Many flag output line possibilities. o I/O control signal: R/W, AS and DS.
Memory	o 2K ROM (for BASIC interpreter) on-chip. o 144 bytes of RWM on-chip: 124 general purpose registers, 16 status registers and 4 I/O registers (data for the 32 I/O lines). o No external RWM necessary for operation (fewer GP-registers available and GOSUB-stack limited (no deeply nested subroutines possible) if on-chip RWM is used only).
Speed	o On-chip clock signal generation. 4, 8 or 12 MHz versions. o Assembly instruction execution time averaging 1.5-2.5 us (no more than 5 us) @ 4 MHz system clock. o External memory write cycle 1.5 us @ 4 MHz system clock. o I/O read/write cycle extension with 500 ns possible. No Halt/Hold mode.
Special features	o Two 8-bit timers/counters with separate prescalers (one generates the baud rate if serial communication is used).
Assembly programs	o Programming model: all registers and external memory locations can be operated on (where it makes sense) in a "dst,src" model. o Five addressing modes: immediate, direct, register indirect, register indexed and program counter

relative.
- o 43 instructions that can be used with any address mode that makes sense.
- o Data can operated on in bits, BCD-coded form, bytes or words (2 bytes).
- o 8-bit arithmetic and logic operations. Decimal adjust operation. 16-bit increment/decrement. Stack operations.
- o Six vectored masked software-programmable prioritized interrupts.

Package — 40-pin DIP for operating temperature 0-70°C for commercial version.

Power
- o +5 V @ 180 mA (max -0.5 - 7.0 V).
- o Power-down mode +3 V @ 10 mA (preserves all internal registers).
- o Fabricated in NMOS and CMOS technology.

BASIC Interpreter
- o Variables A-Z.
- o 16-bit signed integer arithmetic.
- o Several statements on the same program line supported.
- o Byte and word memory references (@ and ¢)

- o Command summary:
  LIST [start,end] { list program [specific lines] }
  NEW { reset program pointers (old program overwritten) }
  RUN { run program }

- o Statement summary:
  GO@ addr,[arg1],[arg2] { call assembly routine that returns no value }
  GOSUB expr { call BASIC subroutine }
  GOTO expr { transfer program control }
  IF expr rel expr [THEN] statement { conditional execution }
  IN var { empty input buffer, then read valu from console }
  INPUT var { ignore input buffer and read value from console }
  [LET] var | fact = expr { assign value to variable or memory location }
  PRINT string | expr { print value or ASCII string }
  REM anything { comments }
  RETURN { return from subroutine }
  STOP { terminate program execution }

- o Function summary:
  AND { logical operation }
  USR(addr,[arg1],[arg2]) { call assembly routine that

returns a value }

Price	\$31.00 in singles. \$22.00 in quantities >100.

Computer boards based on single-chip uCs with BASIC interpreters

As the result of a market survey regarding computer boards based on single-chip microcomputers with on-chip BASIC interpreters, ten manufacturers producing some 20 different computers have been found. These are listed on the following pages, with an overview occuring below. Most of these computer boards use either the Zilog Z8671 or the National INS8073, but there are two that use a 68B09 and a 6502 respectively, both with on-chip BASIC interpreters.

Manufacturer	Product
Arcom Control Systems	ARC40 ARC41 ARC81/82
Basicon	MC-1N Microcontroller
EBV Elektronik	PS73-0
Essex Electronics	Tiny BASIC Computer
HHS Microcontrollers	R8E
Lehman & Associates	SBC 8671 BASIC Controller
Micromint	Z8 BASIC System Controller
Octagon Systems	Sys-1A Sys-2A Sys-3A
Scandia Metric	BAS8
Transwave	K-9000 family: K-9001, K-9002, K-9003 K-8073 family: K-8073, K-8073A, K-8073B

<u>Arcom Control Systems: ARC40</u>

Manufacturer	o Arcom Control Systems (UK)
Product name	o ARC40 Z8 Computer with EPROM Programmer
Intended applications	o Program development for control and instrumentation problems.
CPU	o Z8671 with BASIC/Debug Interpreter (8-bit machine).
System clock	o ?
On-board memory/RWM	o Two 28-pin sockets for 2 or 8K CMOS RWM-chips (totals 16K). Separate power line to enable battery backup.
On-board memory/EPROM	o One 28-pin socket for 2, 4, 8 or 16K EPROM.
Ext. memory expansion	o ?
Remote storage	o On cassette via peripheral board EPIC1.
Total I/O lines	o 40, of which 16 are buffered.
Serial I/O	o Two-way RS-232C via 5-pin connector.
	o Four more RS-232C lines are available for handshake or as a software defined second RS-232 port, also on a 5-pin connector.
Parallel I/O	o Two 8-bit ports of the Z8 provide 16 I/O lines, of which eight are bidirectional.
	o A further 24 I/O lines are provided by a PPI interface chip. Sixteen of these are buffered by 24mA TTL transceivers. These are available when EPROMs are not being programmed. The unbuffered lines from the PPI can be used to control the direction of the buffers, or the second RS-232 port, or used directly.
Miscellaneuos I/O	o -
A/D input	o No, but 12-bit A/D and D/A peripheral board available.
D/A output	o No, but 12-bit A/D and D/A peripheral board available.

Real-time clock	o No
Watch-dog timer	o No
EPROM programming	o On board routines and zero-force insertion socket for 2, 4, 8 and 16K EPROMs.
Miscellaneous features	o Autostart capability.
Power requirements	o +5 V @ 450 mA, +/- 12 V @ 20 mA (and for EPROM programming, 28 to 35 V @ 30 mA). Power supply PSU2T.
Battery backup	o Not on-board (external battery backup to on-board RWM via special line).
Board size	o Single Eurocard 160 mm x 100 mm.
Interface connectors	o ARCbus via DIN41162 Eurocard connector.
	o RS-232C on two 5-pin connectors.
	o 26-pin connector for parallel I/O.
Expansion peripherals	o DI01/DI02/DI03/DI04 Digital I/O with opto-isolators.
	o EPIC1 EPROM progammmer and cassette interface.
	o IOC1 Heavy-duty I/O controller and monitor.
	o IEEE1 IEEE-488/GPIB listener/talker controller.
	o ADA1/ADA12/ADA16 8 and 12-bit A/D and D/A converters with up to 16 differential input channels.
Software support	o XZ8 Cross Assembler for disk-based development systems using CP/M operating system.
	o RZ8 Resident Assembler in 4K EPROM that can be called directly from BASIC. Generated code can be tested in RWM before it is stored in EPROM. Z8 BASIC Compiler ABC1 in 8K EPROM. The ABC1 includes:

1. A 2K run-time package with interrupt handling that allows foreground and background programs to run simultaneously.
2. The 2K BASIC complier that includes FOR-NEXT loops and support for the interrupt handling.
3. The ARC40 Monitor.
4. About 50 utility programs.

o 4K Monitor (can be replaced on-board with BASIC or machine code programs in 2-16K EPROMs).

Price o 169 PS

Arcom Control Systems: ARC41

Manufacturer	o Arcom Control Systems (UK)
Product name	o ARC41 Z8 Computer with clock/battery backup
Intended applications	o Control, measurment and data logging.
CPU	o Z8671 with BASIC/Debug Interpreter (8-bit machine).
System clock	o ?
On-board memory/RWM	o Two sockets for 28-pin 2 or 8K CMOS RWM each. The RWMs are powered from the on-board battery, which is charged whenever the board is externally powered up.
On-board memory/EPROM	o One socket for 28-pin 2, 4, 8 or 16K EPROMs.
Ext. memory expansion	o ?
Remote storage	o On cassette via peripheral board EPIC1.
Total I/O lines	o ?
Serial I/O	o RS-232C, baud rates 110 to 19200.
Parallel I/O	o Two 8-bit ports of the Z8 provide 16 I/O lines, of which eight are bidirectional. Two lines may be taken up with serial I/O.
Miscellaneous I/O	o -
A/D input	o No, but 12-bit A/D and D/A peripheral board available.
D/A output	o No, but 12-bit A/D and D/A perihperal board available.
Real-time clock	o Low-power 146818 real-time clock chip, that provides a full calendar with automatic leap-year calcualtion, periodic interrupts, time of day interrupts and on-chip CMOS RWM. o The clock can drive a reed-relay which switches power either to the rest of the board for "sleep" mode operation, or to an external load. o Control lines to the RWM and clock can be isolated upon receipt of an external signal or if a power-fail is detected.

Watch-dog timer	o Yes, see above.
EPROM programming	o No, peripheral board EPIC1 must be used.
Miscellaneous features	o "Sleep" mode that reduces power consumption to 20 uA.
	o Relay for switching of external loads.
	o Power-fail detection.
	o Autostart capability.
Power requirements	o +5 V @ 330 mA, +/- 12 V @ 20 mA.
Battery backup	o Yes, for RWM and clock.
Board size	o Single Eurocard 160 mm x 100 mm.
Interface connectors	o ARCbus via DIN 41612 Eurocard connector.
	o DB25S Subminiature connector for serial communication.
Expansion peripherals	o DI01/DI02/DI03/DI04 Digital I/O with opto-isolators.
	o EPIC1 EPROM progammmer and cassette interface.
	o IOC1 Heavy duty I/O controller and monitor.
	o IEEE1 IEEE-488/GPIB listener/talker controller.
	o ADA1/ADA12/ADA16 8 and 12-bit A/D and D/A converters with up to 16 differential input channels.
Software support	o RZ8 Resident Assembler in 4K EPROM that can be called directly from BASIC.
Price	o 166 PS

Arcom Control Systems: ARC42/82

Manufacturer	o Arcom Control Systems (UK)
Product name	o ARC42 and ARC82 minimum-chip Z8 computers
Intended applications	o Prototyping with a minimum chip computer.
CPU	o ARC42: Z8671 with BASIC/Debug Interpreter.
	o ARC82: Z8681 ROM-less Z8 (for machine-code programs).
System clock	o ?
On-board memory/RWM	o 2K of CMOS RWM, expandable to 16K on board.
On-board memory/EPROM	o Two sockets for 2, 4, 8 or 16K EPROMs each.
Ext. memory expansion	o ?
Remote storage	o On cassette via periheral board EPIC1 if ARCbus connector is installed.
Total I/O lines	o 16
Serial I/O	o Yes, in TTL-levels from the Z8.
Parallel I/O	o Yes, unbuffered from the Z8.
Miscellaneous I/O	o -
A/D input	o No
D/A output	o No
Real-time clock	o No
Watch-dog timer	o No
EPROM programming	o No, but EPCI1 can be used if bus connector is installed.
Miscellaneous features	o On-board prototyping area measuring 80 x 100 mm (half the size of a Single Eurocard).
	o Autostart capability.
Power requirements	o +5 V @ 250 mA (if no prototyping circuitry).
Battery backup	o No

Board size	o Single Eurocard 160 x 100 mm.
Interface connectors	o No external connectors, but DIN 41612 connector can easily be added to interface to the ARCbus peripheral boards.
Expansion peripherals	o Only if bus connector is installed. See ARC40/41.
Software support	o Nothing that can run on the ARC42/82. See ARC40/41.
Price	o 85 PS

Basicon: MC-01 Microcontroller

Manufacturer	o Basicon, Inc.
Product name	o MC-1N Microcontroller
Intended applications	o Dedicated control.
CPU	o INS 8073 Tiny BASIC Microinterpreter (8-bit machine).
System clock	o 4 MHz
On-board memory/RWM	o 2K CMOS (2K by 8)
On-board memory/EPROM	o Up to 4K (4K by 8)
Ext. memory expansion	o No
Remote storage	o No
Total I/O lines	o 29
Serial I/O	o RS-232, baud rates 110-4800 (+/- 4 V @ 5 mA).
Parallel I/O	o 24 programmable lines through 8255 PPI.
Miscellaneous I/O	o 3 flag outputs, 2 interrupt inputs.
A/D input	o No
D/A output	o No
Real-time clock	o MM58174 CMOS clock/calendar chip with 0.1 s resolution and interrupt output.
Watch-dog timer	o Yes.
EPROM programming	o On easily attached peripheral board for 2716s and 2732s.
Miscelaneous features	o -
Power requirements	o + 5 V @ 200 mA (- 5 V @ 5 mA for RS-232 generated on-board).
Battery backup	o Clock/calendar chip and RWM wired for battery backup.
Board size	o 3.0" x 4.0"

Interface connectors	o 10-pin header connector for power and RS-232.
	o 50-pin header connector for parallel I/O and EPROM-board attachement.
Expansion peripherals	o EPROM-programming board.
Software support	o Utilities package.
Price	o $139

EBV Elektronik: PS73-0

Manufacturer	o EBV Elektronik (W. Germany)
Product name	o PS73-0
Intended applications	o ?
CPU	o INS8073 Tiny BASIC Microinterpreter (8-bit machine).
System clock	o ?
On-board memory/RWM	o ?
On-board memory/EPROM	o ?
Ext. memory expansion	o ?
Remote storage	o On cassette via on-board interface.
Total I/O lines	o ?
Serial I/O	o ?
Parallel I/O	o ?
Miscellaneuos I/O	o ?
A/D input	o ?
D/A output	o ?
Real-time clock	o ?
Watch-dog timer	o ?
EPROM programming	o ?
Miscellaneous features	o ?
Power requirements	o ?
Battery backup	o ?
Board size	o ?
Interface connectors	o ?
Expansion peripherals	o ?

Software support	o ?
Price	o 690 DM

Essex: Tiny BASIC Computer

Manufacturer	o Essex Electronics Centre (UK)
Product name	o Essex Tiny BASIC Computer
Intended applications	o Dedicated control and software development.
CPU	o INS 8073 Tiny BASIC Microinterpreter (8-bit machine).
System clock	o 4 MHz.
On-board memory/RWM	o 2K of CMOS RWM (256 bytes internally used by the 8073).
On-board memory/EPROM	o Two sockets for up to 8K EPROM (two 2732s or one 2764). PROM decoder.
Ext. memory expansion	o ?
Remote storage	o -
Total I/O lines	o ?
Serial I/O	o RS-232C, baud-rates 110-4800.
Parallel I/O	o 48 programmable I/O lines via two 8255 parallel interfaces.
Miscellaneous I/O	o ?
A/D input	o No
D/A output	o No
Real-time clock	o No, but low-frequency square generator (see below).
Watch-dog timer	o 1-14 sec. adjustable time-out.
EPROM programming	o On board for 2732s and 2764s.
Miscellaneous features	o Crystal oscillator/divider with selectable rates 2, 4 or 8 Hz for slow timing applications (output can be tied to interrupt line).
	o Autostart capability.
Power requirements	o +5 V @ 500 mA (+25 V @ 40 mA during EPROM programming).

Battery backup	o No
Board size	o Single Eurocard 160 x 100 mm.
Interface connectors	o 64-way DIN Eurocard connector to bus.
	o 34-way IDC headers for I/O lines.
Expansion peripherals	o Essex Buffer/Timer Board.
	o Essex Opto Isolator Board.
	o Essex 12-bit A/D & D/A Board.
	o Essex 12-bit A/D Board.
	o Battery Backed Memory Board with real-time clock/calendar.
	o Serial Interface Board (RS422/RS423 and current loop).
	o Essex Speech Board and paged memory.
Software support	o MON Essex Tiny Monitor.
	o ALEX Assembly Language Extension in EPROM (assembler and disassembler).
	o TURBO Tiny Turbo BASIC Compiler.
Price	o 198 PS

<u>HHS Microcontrollers: R8E</u>

Manufacturer	o HHS Microcontrolers
Product name	o R8E
Intended applications	o ?
CPU	o Z8671 with BASIC/Debug Interpreter (8-bit machine).
System clock	o ?
On-board memory/RWM	o 16K (together with ROM/EPROM ?) with battery backup.
On-board memory/EPROM	o 16K (together with ROM/EPROM ?)
Ext. memory expansion	o ?
Remote storage	o ?
Total I/O lines	o ?
Serial I/O	o Yes
Parallel I/O	o Yes
Miscellaneous I/O	o 110 V AC power control ports.
A/D input	o Yes, "sensor input ports."
D/A output	o ?
Real-time clock	o ?
Watch-dog timer	o ?
EPROM programming	o ?
Miscellaneous features	o ?
Power requirements	o ?
Battery backup	o Yes
Board size	o ?
Interface connectors	o ?
Expansion peripherals	o ?

Software support	o ?
Price	o $299

Lehman & Associates: SBC 8671 BASIC Controller

Manufacturer	o Lehman & Associates
Product name	o SBC 8671 BASIC Controller
Intended applications	o Industrial control.
CPU	o Z8671 with BASIC/Debug Interpreter (8-bit machine).
System clock	o 3.68 MHz.
On-board memory/RWM	o Sockets for a total of 48K ROM, RWM or EPROM (8Kx8 static RWM supported).
On-board memory/EPROM	o Sockets for a total of 48K ROM, RWM or EPROM.
Ext. memory expansion	o Total addressable memory 120K (56K program memory and 64K data memory).
Remote storage	o ?
Total I/O lines	o ?
Serial I/O	o RS-232C and opto-isolated current loop. Baud rates 110-9600 with 6 user-programmable handshake lines.
Parallel I/O	o 48 parallel I/O lines grouped into six 8-bit ports.
Miscellaneous I/O	o Four ports (32 lines) have sockets that are pin compatible with industry standard 7400-style opto I/O buffer modules.
A/D input	o No
D/A output	o No
Real-time clock	o No
Watch-dog timer	o No
EPROM programming	o ?
Miscellaneous features	o Six vectored interrupts (two internal, four external).
	o Small prototype area.

Power requirements	o + 5 V @ ? mA (+/- 12 V @ 35 mA for RS-232).
Battery backup	o ?
Board size	o 6.00" x 8.25"
Interface connectors	o Five post (header) connectors for ribbon cables, one of which is a 50-pin module connector containing all Z8671 lines.
Expansion peripherals	o ?
Software support	o Z8 Cross Assembler for CP/M computers (from Microresources).
Price	o $285

Micromint: Z8 BASIC System Controller

Manufacturer	o Micromint Inc.
Product name	o Z8 BASIC System Controller BCC11
Intended applications	o Dedicated control, smart instrumentation.
CPU	o Z8671 with BASIC/Debug Interpreter (8-bit machine).
System clock	o 3.68 MHz (not available on bus).
On-board memory/RWM	o Two sockets for 4K (can be combined with EPROM for a maximum of 6K on-board memory).
On-board memory/EPROM	o Two sockets for up to 6K (2716s and 2732s).
Ext. memory expansion	o Up to 62K total memory supported.
Remote storage	o On cassette via peripheral board.
Total I/O lines	o 24
Serial I/O	o RS-232C, baud-rates 110-9600.
Parallel I/O	o Two parallel 8-bit ports on board of which one is dedicated to input only, the other being bit-programmable.
Miscellaneous I/O	o Memory-mapped input only/baud rate switch port.
A/D input	o No
D/A output	o No
Real-time clock	o No
Watch-dog timer	o No
EPROM programming	o On peripheral EPROM-programming board (the programming software package has to reside on main computer board).
Miscellaneous features	o -
Power requirements	o +5 V @ 250 mA, +/- 12 V @ 30 mA (for RS-232C).
Battery backup	o No
Board size	o 4" x 4.5"

Interface connectors	o 22/44 pin edge-connector to Z8-bus.
	o DB25-S Subminiature female connector for RS-232C.
Expansion peripherals	o Z8 Memory (8K), I/O Expansion (three memory-mapped 8-bit parallel ports) and Cassette Interface Board.
	o Z8 BASIC 16K Memory Expansion Board (with 16K RWM or EPROM).
	o Z8 EPROM programming board.
	o Z8 Serial Expansion Board (for current loop 75-19200 baud).
	o Z8 BASIC A/D Converter (for up to eight channels of 8-bit conversion).
Software support	o Z8 Cross-Assemblers for CP/M computers (from Allen Ashley and Micro Resources).
Price	o $149

Octagon Systems: Sys-1A

Manufacturer	o Octagon Systems
Product name	o Sys-1A
Intended applications	o Prototyping and development.
CPU	o INS8073 Tiny BASIC Microinterpreter (8-bit machine).
System clock	o 4 MHz
On-board memory/RWM	o 4K RWM space (2K x 8, static).
On-board memory/EPROM	o 6K EPROM space (4K if RWM is installed).
Ext. memory expansion	o ?
Remote storage	o -
Total I/O lines	o 29
Serial I/O	o RS-232C, baud-rates 110-4800 (factory setting is 4800).
Parallel I/O	o 24 lines user-configured through one 8255 parallel interface.
Miscellaneous I/O	o -
A/D input	o No
D/A output	o No
Real-time clock	o No
Watch-dog timer	o No
EPROM programming	o On-board
Miscellaneous features	o 1 interrupt vectored to BASIC line.
	o Prototyping area that holds 10 16-pin DIPs/ICs.
	o Autostart capability.
Power requirements	o +5 V @ 300 mA (+25 V @ 30 mA for EPROM programming).
Battery backup	o No

Board size	o 6.5" x 4.5"
Interface connectors	o 22/44 pin edge connector to bus.
	o DB25-S Subminiature connector for RS-232C.
Expansion peripherals	o -
Software support	o Sys-1 Firmware Library (hex screen dump, program move, hex/dec number conversion, serial line print).
	o Sys-1 Utility Libary.
Price	o $245

Octagon Systems: Sys-2A

Manufacturer	o Octagon Systems
Product name	o Sys-2A
Intended applications	o Dedicated control and software development.
CPU	o INS8073 Tiny BASIC Microinterpreter (8-bit machine).
System clock	o 4 MHz (buffered on bus for 20 TTL-loads).
On-board memory/RWM	o 4K RWM space (2K x 8, static).
On-board memory/EPROM	o 4K EPROM space 2716-type (max. 4K if RWM is installed). 5V only EEPROMs may be used.
Ext. memory expansion	o Up to 24K RWM and 26K EPROM.
Remote storage	o -
Total I/O lines	o -
Serial I/O	o RS-232C, baud-rates 110-4800. Half duplex supported.
Parallel I/O	o -
Miscellaneous I/O	o Eight high-current digital outputs sink 500 mA @ 50 V.
	o Eight LSTTL switch closure inputs.
A/D input	o Four channels of 12-bit A/D, 10 ms conversion time, 2.5 V adjustable full scale.
D/A output	o No
Real-time clock	o No
Watch-dog timer	o No
EPROM programming	o On-board
Miscellaneous features	o All I/O lines individually addressable.
	o Interrupt vectored to BASIC program line.
	o Autostart capability.

Power requirements	o +5V @ 300 mA (+25 V @ 30 mA for EPROM programming).
Battery backup	o No
Board size	o 6.5" x 4.5"
Interface connectors	o 22/44 pin edge connector to bus.
	o 34 pin edge connector for I/O expansion.
	o DB25-S Subminiature for serial communication.
Expansion peripherals	?
Software support	Menu driven utility library 1A.
	o Sys-2 Firmware Library.
Price	o $295

Octagon Systems: Sys-3A

Manufacturer	o Octagon Systems
Product name	o Sys-3A
Intended applications	o Dedicated controller.
CPU	o INS8073 Tiny BASIC Microinterpreter (8-bit machine).
System clock	o 4 MHz (buffered on bus for 20 TTL-loads).
On-board memory/RWM	o 4K static.
On-board Memory/EPROM	o 4K (2716 or EEPROM).
Ext. memory expansion	o Up to 24K RWM and 26K EPROM.
Remote storage	o On cassette via expansion peripheral SUP-5.
Total I/O lines	o 46
Serial I/O	o RS-232C, baud-rates 110-4800. Software programmable for full or half duplex.
Parallel I/O	o 24 TTL compatible lines.
Miscellaneous I/O	o Eight of the parallel lines have 50 V @ 500 mA drivers.
	o Three single-bit flag outputs.
	o Two Schmidt trigger inputs.
	o I/O select lines decodes 2K each.
A/D input	o Eight channels: four channels 0-5 V, four channels 0 - +/-5 V or 0-10 V. Conversion time 75 ms.
D/A output	o One 8-bit channel, 0-2.5 V output.
Real-time clock	o No
Watch-dog timer	o No
EPROM programming	o On-board
Miscellaneous features	o Two interrupts vectored to BASIC line numbers.
	o Autostart capability.

Power requirements	o +5 V @ 300 mA (+25 V for EPROM programming). RS-232 voltages generated on-board.
Battery backup	o No
Board size	o ?
Interface connectors	o 22/44 pin edge connector to bus.
	o 56 pin edge connector to parallel I/O.
	o DB25-S Subminiature connector for serial communication.
Expansion peripherals	o SUP-5 Multisupport card.
	o SUP-6 Multifunction card.
	o SUP-7 Opto-relay I/O card.
Software support	o Text editor/assembler on peripheral board SUP-5.
	o Sys-3A Utility Library (more powerful than Sys-1A Utility Library).
Price	o $345

Scandia Metric: BAS8

Manufacturer	o Scandia Metric (Sweden)
Product name	o BAS8
Intended applications	o ?
CPU	o Z8671 with BASIC/Debug Interpreter (8-bit machine).
System clock	o ?
On-board memory/RWM	o 2K
On-board memory/EPROM	o ?
Ext. memory expansion	o ?
Remote storage	o ?
Total I/O lines	o ?
Serial I/O	o Yes
Parallel I/O	o Two parallel ports.
Miscellaneous I/O	o ?
A/D input	o ?
D/A output	o ?
Real-time clock	o ?
Watch-dog timer	o ?
EPROM programming	o On-board programming utilities and EPROM socket.
Miscellaneous features	o ?
Power requirements	o ?
Battery backup	o ?
Board size	o Eurocard
Interface connectors	o DIN connector.
Expansion peripherals	o ?.

Software support	o ?
Price	o 2430 SKR

Transwave: K-9000

Manufacturer	o Transwave Corp
Product name	o K-9000 family: K-9001, K-9002 and K-9003
Intended applications	o -
CPU	o K-9001: INS8073 Tiny BASIC Microinterpreter (XMOS). o K-9002: 68B09 with Tiny BASIC and Monitor. o K-9003: 6502 with Tiny BASIC (CMOS).
System clock	o 4, 8 or 16 MHz.
On-board memory/RWM	o One socket for 2 or 8K (contiguous with external RWM).
On-board memory/EPROM	o One socket for 8 or 16K (contiguous with external EPROM). Firmware/utilities occupy this space if used.
Ext. memory expansion	o ?
Remote storage	o ?
Total I/O lines	o ?
Serial I/O	o RS-232C with handshake and selectable baud-rate 300-19000 (serial communication via bus).
Parallel I/O	o ?
Miscellaneous I/O	o All I/O lines are buffered and the bus DMA-structured.
A/D input	o No, but peripheral board avialable.
D/A output	o No
Real-time clock	o No
Watch-dog timer	o No
EPROM programming	o ?
Miscellaneous features	o Autostart capability. o Multi-tasking capability (master/slave).

	o DMA structured bus with all lines buffered (I/O, address lines, data lines, RS-232C, ART/RC, bus request and control lines).
Power requirements	o +5 V only. +/-12 V for RS-232 generated on board.
Battery backup	o Yes, via bus.
Board size	o 4.0" x 3.35"
Interface connectors	o 50 pin edge connector to bus.
Expansion peripherals	o M-9048 48K Memory Card.
	o C 9002 Communication Card.
	o V 9002 Color Video Generator Card.
	o D 9020/L 16 channel A/D Card.
	o D 9028/L 8 channel Triac Controller.
Software support	o Monitor firmware.
Price	o $199

Transwave: K-8073

Manufacturer	o Transwave Corp.
Product name	o K-8073 family: K-8073, K-8073A and K-8073B XCalibur
Intended applications	o Development and control.
CPU	o INS8073 Tiny BASIC Microinterpreter (8-bit machine).
System clock	o 4 or 8 MHz (A/B only).
On-board memory/RWM	o K-8073: 1K. o K-8073A: 2K expandable to up to 8K on-board. o K-8073B: ?
On-board memory/EPROM	o K-8073: 8K. o K-8073A: 2K autostart EPROM. o K-8073B: ?
Ext. memory expansion	o K-8073A: 40K of which no more than 24K is RWM.
Remote storage	o Cassette interface on-board (via bus).
Total I/O lines	o ?
Serial I/O	o RS-232C, 110-4800 baud with handshake.
Parallel I/O	o Three 8-bit programmable I/O ports via 8255 PPI/W.
Miscellaneous I/O	o ART/RC Master for 128 remote addressable slave units via single wire (A/B only). o Three flag output lines and two input sense lines (A/B only).
A/D input	o No, but peripheral board avaliable.
D/A output	o No
Real-time clock	o Yes, with external battery backup.
Watch-dog timer	o Yes, CPU time-out interrupt possible.

EPROM programming	o On-board
Miscellaneous features	o Autostart capability.
	o K-8073A has color/video o generator with graphics and comes in aluminum case.
Power requirements	o +5 V only (+25 V when EPROM programming).
Battery backup	o On K-8073A/B.
Board size	o 4.5" x 6.5"
Interface connectors	o Edge connector to modified STD-bus.
Expansion peripherals	Many, for example:
	o Memory Boards
	o Backplanes.
	o Data Control Modules.
	o I/O Modules.
	o A/D Modules.
	o Remote Sensors.
Software support	o ?
Price	o K-8073: $388
	o K-8073A: $478
	o K-8073B: $684

APPENDIX C

SPEECH TECHNOLOGY PRODUCTS

Recent Interesting Low-cost Speech Synthesis Products

<u>AMERICAN MICROSYSTEMS INC.</u>

S3610	CMOS LPC-10 chip for voice output. 32 words, 17 s high quality speech. 1.4 b/s data rate. 20 k ROM on chip. Switched capacitor filtering. Less than $20 in quantities.
S3620	CMOS LPC-10 chip for uP interfacing. 2 min high quality speech output.

<u>GENERAL INSTRUMENT CORP.</u>

SP0256-AL2	Speech synthesizer chip based on allophone technique. Complete allophone library in internal memory. Handles voice-signals in three forms: LPC-coded, formant coded or allophone coded. Software programmable digital filter. 16 kbit internal ROM for speech instructions. Directly addressing of 491 kbits external ROM. $15.

<u>HITACHI</u>

HD 38880/1	Speech synthesizer chips with on-chip ROM. Parcor technique. 1.2-9.6 kb/s. 38880 can address up to 16 128 kbits ROM's with 10 s of speech in each.
HD 61885	Speech output CMOS IC using Parcor algorithm. Up to 63, data rate 1.2-9.9 kbits/s. 32 kbit ROM, D/A, keyboard and external ROM interface. 5 V power. 26 s of speech.

<u>MICROMINT INC.</u>

Sweet Talker II	Phoneme synthesizer board based on the SSI-263 chip (Votrax SC-02).

NATIONAL SEMICONDUCTOR CORP.

MM 54104	Speech processor for Digitalker. $5.00
Digitalker I	Speech synthesizer chip set with DT 1050.
DT 1050/1052/ 1057	Standard vocabulary set PROMs for Digitalker.
SPC	Speech output IC using waveform coding. Up to 256 utterances from 128 kbit ROM. Programmable frequency generator and varaible gain DAC.
MM54107	Chip for 1-2 kb/s with Adaptive Delta Modulation.
DT3101	Speech processor with external control of pitch and inflection levels. 800 b/s. $20 in 1k quantities.
DTSW-500 DVSS	Digitalker Vocal Selection System: software package for speech development for MM54104 Digitalker chip set. Works with any uC running under CP/M. Package includes a program disk and an archive disk with speech data for 500 words, numbers and the alphabet.
Digitalker II	Second generation of Digitalker chip sets. Uses predictive coding and needs 400-1000 b/s.
DT1060	"COP" Talker Control ROM (16 kbit) designed to allow auditioning of custom generated speech vocabularies through DT 1000 Digital Evaluation board and DT 1058 Epromtalker Board.
DT1058	An EPROM board to be used for storage of speech to be generated with the MM54104 Digitalker Speech Processor.

OKI SEMICONDUCTOR CORP.

MSM 5218	Analyzer/synthesizer using ADPCM for real-time record and playback. Speech analysis and synthesis on the same chip. $4.50 in 100k quantities.

PANASONIC

MN 6401	Versatile speech synthesis chip with 32 k on-chip ROM. Up to 20 s of speech with 63 words. Male/female voice, varying word rate and external ROM address capability.

SANYO SEMICONDUCTOR CORP.

LC-8100	One-chip CMOS speech synthesizer system. 32 kbits ROM on chip, 16 external ROM's can be addressed which gives over 26 min of speech. 2 mA operating power consumption and 1 uA standby current. $3.90 in 10k quantities.

SHARP ELECTRONICS

ADPCM	CMOS microcomputer chip with adaptive pulse code modulation. Samples speech at 8 kHz. 33 instructions.
?	Low-cost single-chip voice recognition system. Up to 24 words in any type of voice recognized with 90% accuracy. Controlled by its on-chip 4-bit CMOS uP. $10-20. To be introduced in 1984.

SILICON SYSTEMS INC.

SSI 263	CMOS integrated circuit that produces synthesized voice, music and sound effects. Uses 64 different phonemes, each with four different duration settings, giving the equivalent of 256 phonemes accessed by 8-bit codes. Also has 32 pitch values, eight inflection speeds, 16 rate settings and 16 amplitude levels. 8-bit bus compatible. Developed by Votrax under the name SC-02. $40 in qunatities of more than 25.

TEXAS INSTRUMENTS

TMS K202	Speech synthesis evaluation kit for 8 or 16-bit systems. 241 words or phrases. Uses TMS 5220 chip.
TMS 5200	Speech synthesis processor for 8-bit systems.
TMS 5220	Speech synthesis processor for 8 or 16-bit systems. 1200 word vocabulary.

TOSHIBA

Parcor chip	CMOS chip with LPC-related Parcor algorithm. ON-chip ROM, can address 8 Mbits external ROM. Data rate 1.2-9.6 kb/s.

VOTRAX

SC-01	CMOS speech synthesizer chip with phoneme (formant) technique. Input 6-bit phoneme code and 2-bit pitch code. Only 70 bits/s required. $5 in OEM quantities.
SC-02	Second-generation phoneme synthesizer chip. Much higher voice quality than the SC-01, but not pin- or software compatible with it. The SC-02 can also sing and synthesize music, but requires a data-rate of 500 bit/s. Manufactured by Silicon Systems Inc. under the name SSI-263. $40 (in quantities >25).

Text-to-speech converters

INFOVOX

SA 101	Text-to-speech synthesis system that works in seven different languages. Phoneme synthesis with an additional software subsystem that modifies the synthesis parameters according to a set of higher level context-dependent rules which gives speech of high quality. An NEC 7720 signal processor for formant synthesis and a Motorola uP for overall control. $1,500.

INTEX MICRO SYSTEMS INC.

Intex-Talker	Text-to-speech synthesizer with 64 programmable inflexion levels. 6502 uP, on-board amplifier and power supply. $295.

MICROMINT

Microvox	ASCII text-to-speech converter with spelling option. $295.

STREET ELECTRONICS CORP.

ECHO Speech Module	Speech synthesizer module. Text-to-speech algorithm. Variable rate of speech, 63 pitch levels. Mixes text-to-speech conversion with prestored words. $100 in OEM quantities.

ECHO GP	Text-to-speech board based on TI 5200. $370.

SWEET MICROSYSTEMS

Mockingboard	ASCII text-to-speech converter and sound generator for the Apple II computer. Spelling function built-in. Several versions available. $199.

Examples on speech recognition products

TEXAS INSTRUMENTS

TMS 32010	Signal processing chip that can be programmed for voice recognition using digital inverse filtering. Recognition of connected speaker-dependent speech with a vocabulary of more than 40 words. The 320101 can also be programmed for voice synthesis.

APPENDIX D

GLOSSARY OF PHONETICS, SPEECH DISORDERS AND SPEECH TECHNOLOGY

Affricate: a consonant resembling a fricative formed by the succession of two consonants such as J=D.ZH.

Allophone: spoken variations in the pronounciation of a phoneme dependent on the phonemes placement within words. An allophone can be regarded as the sound that results when a phoneme is placed in its environment.

Aphonia: loss or absence of voice due to a failure of vibration of the vocal cords.

Auto-correlation: a method of signal processing created by delaying the original signal and then multiplying the delayed signal by the original.

Band-pass filter: a filter that allows only specific segments of the frequency spectrum to be passed.

Broca's area: a portion of the brain in the frontal lobe which relates to language perception and expression.

Cepstrum: the frequency spectrum of a frequency spectrum.

Channel vocoder: parameter-encoding technique that models the frequency spectrum of the speech signal using a bank of band-pass filters.

Chirp: a wideband signal created by a rapid frequency sweep.

Coding tables: tables used to quantize LPC paramters to a limited number of different values.

Cognates: the related pairs of voiced and unvoiced fricative consonants.

Compressed speech: a representation in which some redundant features of the digitized speech have been removed.

Constructive synthesis: a synthesis technique in which the LPC representations of individual sounds are joined to create words or phrases.

Continously variable-slope delta modulation: an extension of the delta modulation technique that allows the amplitude change (slope) to vary.

Continuant: a static speech sound requiring no motion of the vocal tract other than the vocal cords and/or lungs.

Delta modualation: a type of waveform coding in which the differences between individual digitized speech samples are encoded. Data point values are determined by changes from preceding data point amplitudes.

Demisyllable: the region from the middle of one syllable to either of its boundaries.

Digitized speech: a numerical representation of speech in which the amplitude of the speech waveform has been recorded at regular intervals. Speech is typically sampled from 8000 to 12,500 times per second.

Diphthong: a speech sound formed between two spoken vowels.

Dysphonia: an impairment of the larynx which affects the proper voice production.

Excitation: the driving signal to any speech model. In the human model, the movement of the vocal cords.

Fidelity: the accuracy of signal reproduction; lack of distortion.

Formants: resonances in the frequency spectra of voiced speech. Formants appear as bands in a spectrographic display of voiced speech. Formants help to distinguish one sound from another.

Formant synthesis: a parameter-encoding technique that models speech information by tracking the formants of the frequency spectrum.

Frame: the information included within a segment of time in an utterance. A single frame usually represents 10 to 25 msec.

Fricative: s speech sound having a broad frequency spectrum, ususally characterized by a hissing sound.

Fundamental frequency: the lowest frequency component in a harmonically distorted signal.

Homomorphic filter: a filter which passes a desired signal while rejecting all undesired components.

Hononyms: words spelled differently but pronounced identically.

Inflection: a means of "coloring" speech meanings with intentional pitch variations.

Intonation: the melody of pitch over a word or phrase.

Lexicon: a list of features in a language containing phonological, syntactic and semantic features.

Linear Predictive Coding: a parameter-encoding technique that mathematically models the human vocal tract with a digital filter whose controlling parameters change with time. Changes are based on previous speech samples.

Morph: a sound unit that has an associated meaning. Root morphs such as "dog" and "book" are independent; "bound" morphs such as "-s" for the plural or possesive ending must be attached to a root morph.

Nyquist theorem: theorem stating that to preserve fidelity, a signal must be sampled at least at twice its highest frequency component (also called the sampling theorem).

Parameter coding: synthesis technique that reproduces speech by a parametric model. The parameters used are descriptions of unique characteristics of human speech signals.

Phonemes: a set of abstract basic sound units of speech that can be used for writing a language in a systematic and unambiguous way. English has about 40 phonemes: 16 vowel phonemes and 24 consonant phonemes.

Pitch: the predominant frequency of vibration of the vocal cords (or other acoustic source). The physiological vibration correlate to the fundamental frequency of a voice sound.

Prosody: a general term including energy, stress, intonation, pause and duration phenomena in speech.

Plosive: a speech sound consonant also known as stop consonant.

Pulse code modulation: the waveform is sampled at a set rate (for example, 800 times per sec.), and the amplitude of each sample is coded and stored. The signal is reproduced by playing the data back through a digital-to-analog converter.

Recursive filter: a filter which multiply feedback in a lattice configuration.

Reflection coefficient: prediction parameters used for digital filtering in the LPC model to describe the human vocal tract mathematically. The term "LPC-10" indicates that 10 reflection coefficients are used.

Semi-wovel: a group of consonants consisting of the sounds Y, W, R and L. Also known as glide or liquid.

Slope overload: an artifact of delta modulation which causes signal distortion; it occurs when the delta modulator cannot keep up with signal changes.

Spectrogram: a machine-made graphic representation of sounds in terms of their component frequencies. Time is shown on the horisontal axis, frequency on the vertical axis and intensity by darkness of the mark. Also known as "voice-print."

Speech construction: joining synthetic sound units to form words, phrases or sentences.

Stress: the perception of a syllable as louder or more accented then the rest of the syllables in a word.

Synthetic speech: artificially reproduced acoustic signals that are recognizable as human speech.

Unvoiced sounds: sounds produced without vibration of the vocal cords. Unvoiced sounds do not have a periodic sound wave.

Voiced sounds: sounds produced by the vibrations of the vocal cords during articulation. Voiced sounds have a periodic sound wave.

Waveform encoding: a synthetic technique that reproduces speech by reconstructing the original speech waveform.

White noise: a completely random signal waveform that contains all frequencies; acoustically it sounds like a hiss.

APPENDIX E

AMERICAN-ENGLISH PHONEMES

American English phonemes can be continuant or noncontinuant sounds. Continuant sounds are produced by fixed or non-time-varying vocal-tract configuration excited by the appropriate source, and include vowels, voiced and unvoiced fricatives and nasals. Diphtongs, semivowels, affricatives and whispers are noncontinuants produced by a changing vocal-tract configuration. Plosives, or stop consonants, are noncontinuants formed by obstructing the airflow differently within the oral cavity. Nasals are voiced consonants that use the nasal cavity as a resonator.

In the following table the different phonemes are listed and examples of words that they used in are provided. The varieties in length is the main difference between phonemes on the same line, thus the table can also be regarded as a subset of the 128 allophones in the American-English language. Each example of pronounciation (the underlined letter) corresponds to the same relative phoneme entry on that particular line.

Classification	Phoneme symbols	Example words			
Vowels	A A1 A2	tame	pail	make	
	AE AE1	dad	after		
	AH AH1 AH2	mop	honest	father	
	AW AW1 AW2	call	lawfull	salty	
	AY	jade			
	E	meet			
	ER	bird			
	I I1 I2 I3	pin	inhibit	inhibit	inhibit
	O	cold			
	OO OO1	book	looking		
	U U1	move	June		
	UH UH1 UH2 UH3	cup	uncle	about	mission
	Y	any			
Diphtongs	E1	be			
	EH EH1 EH2 EH3	ready	heavy	enlist	jacket
	IU	you			
	O1 O2	aboard	bold		
Semivowels	W	win			
	Y1	yard			
Glides	L	land			
	R	red			
Consonants					
Voiced fricatives	TH THV	thin	the		
	V	van			
	Z	zoo			
	ZH	pleasure			
Unvoiced fricatives	F	fast			
	S	pass			
	SH	shop			
Nasals	M	mat			
	N	sun			
	NG	thing			
Voiced plosives	B	bag			
	D	paid			
	G	get			
Unvoiced plosives	K	trick			
	P	play			
	T DT	tap	butter		
Affricatives	CH (TSH)	chip			
	J (DZH)	judge			
Whisper	H	hello			

APPENDIX F

ALPHANUMERIC LIQIUD CRYSTAL DISPLAYS

Examples on alphanumeric dot matrix LCDs

EPSON AMERICA, INC.

A wide variety of CMOS TTL-compatible alphanumeric (5x7 dot matrix) and graphic LCDs is offered, ranging from 16x1 to 80x4 characters. All displays have a built-in 96-character ASCII character generator and data RWM, and are operated with a 5 V only power supply. The wide viewing angle and high contrast is praised by the manufacturer. Prices from $50 and up.

HITACHI AMERICA, LTD.

A large selection of displays ranging from 1x40 to 4x40 character modules are offered. Power consumption typically a few tens of mW.

INDUSTRIAL ELECTRONICS ENGINEERS (IEE), INC.

Daystar 5 V low-power LCDs with ASCII generators and 8-bit bus interface.

PRINTED CIRCUITS INTERNATIONAL (PCI), INC.

Five standard low-cost modules, from 1x16 to 2x40 characters, offered by PCI. The PCIM-201, for instance, features 64 different ASCII characters, 5 V CMOS bus-oriented drive- and control circuitry with power-down mode, powerful display manipulation and much more.

SEIKO INSTRUMENTS

Some 20 different LCDs with CMOS drive and control circuit. Programmable character generator for up to eight custom figures in addition to 160 ROM-stored characters.

SHARP CORP.

Many different displays from 16x1 to 40x2 are offered by Sharp, featuring built-in character generator and 5 V only power requirements.

APPENDIX G

PRICES OF USED COMPONENTS

One of the design criteria was to use as inexpensive parts as possible for its purpose. The approximate prices for the necessary parts in a "minimum system" version of the "Communicaider" are listed as follows (when ordered in quantities):

Zilog Z8671 microcomputer	$25.00
National MM54104 Digitalker	$12.00
Votrax SC-01 phoneme synthesizer	$44.00
National DT1050 vocabulary set	$35.00
National DT1057 vocabulary set	$25.00
PCIM-201 alphanumeric LCD	$53.00
Program & data EPROMs	$10.00
Interface circuitry	$25.00
Audio circuitry	$ 3.00
Speaker	$10.00
Power supply/battery circuit	$40.00
Enclosure	$10.00
Sum (approximately)	$292.00

APPENDIX H

ADDRESSES TO REFERRED MANUFACTURERS

American Microsystems Inc.	3800 Homestead Rd. Santa Clara, CA 95051 (408) 246-0330
Arcom Control Systems	Unit 6, Robert Davies Ct. Nuffield Rd Industrial Estate Chesterton, Cambridge, UK (0223) 522-642
Basicon, Inc.	7713 SW Nimbus, Bldg. 31 Beaverton, OR 97005 (503) 626-1012
EBV Elektronik	Oberweg 6 8025 Unterhaching West Germany (089) 611 05 1
Epson America, Inc.	3415 Kashiwa Street Torrance, CA 90505 (213) 533-8277
Essex Electronics Centre	University of Essex Wivenhoe Park Colchester CO4 3SQ, UK (0206) 865-089
General Instrument Corp.	Microelectronics Div. 600 W John St. Hicksville, NY 11802 (516) 733-3107
Hitachi America, Ltd.	500 Park Boulevard, Suite 805 Itasca, IL 60143 (312) 773-0700
HHS Microcontrollers	5876 Old State Edinboro, PA 16412 (814) 734-4338
Hitachi	175 Crossways Pk. W Woodbury, NY 11797 (516) 921-7200

IEE	7740 Lemona Ave. Van Nuys, CA 91405 (818) 787-0311
Infovox	Box 7733 S-103 95 STOCKHOLM Sweden 0046-8-14 14 60
Intel Corp.	Microcomputer Div. 2625 Walsh Ave. Santa Clara, CA 95051 (408) 987-4928
	5000 Williams Field Rd. Chandler, AZ 85224 (602) 961-2609
	3065 Bowers Ave. Santa Clara, CA 95051 (408) 987-8080
Intex Micro Systems Inc.	755 West Big Beaver Rd., Suite 1719 Troy, MI 48084 (313) 362-4280
Lehman & Associates	P.O. Box 566 Maumee, Ohio (419)-891-0687
Micromint Inc.	917 Midway Woodmere, NY 11598 (800) 645-3479 (203) 871-6170
	561 Willow Ave. Cedarhurst, NY 11516 (516) 374-6793
National Semiconductor Corp.	2900 Semiconductor Dr. Santa Clara, CA 95051 (408) 721-5000
Octagon Systems	5150 W 80th Ave. Westminster, CO 80030 (303) 426-8540
OKI Semiconductor Inc.	1333 Lawrence Ewpressway Santa Clara, CA 95051 (408) 984-4842

Panasonic	One Panasonic Way Secaucus, NJ 07094 (201) 348-5270
Phonic Ear Inc. (also HC Electronics)	250 Camino Alto Mill Valley, CA 94941 (415) 383-4000
Printed Circuits International (PCI), Inc.	1145 Sonora Court Sunnyvale, CA 94086 (408) 733-4603
Sanyo Semiconductors	7 Pearl Court Allendale, NJ 07401 (201) 825-8080
	1200 W Artesia Blvd. Compton, CA 90220 (213) 537-5830
Scandia Metric AB	Dept. for Systems Box 1307 S-171 25 Solna Sweden
Seiko Instruments	2990 W. Lomita Blvd. Torrance, CA 90505 (213) 530-8777
Sharp Electronics	10 Sharp Plaza POB 588 Paramus, NJ 07652 (201) 265-5600
Silicon Systems Inc.	14351 Myford Rd. Tustin, CA 92680 (714) 979-0941
Street Electronics Corp.	3152 E La Palma Anaheim, CA 92802 (714) 632-9950
	1140 Mark Ave. Carpinteria, CA 93013 (805) 684-4593
Texas Instruments	Semiconductor Group Box 401560,SC-399 Dallas, TX 75240

	Microcomp. Div Box 1443 Houston, TX 77001 (713) 879-2000
Transwave Corp.	Cedar Valley Bldg. Vanderbilt, PA 15486 (412) 628-6370
Votrax (Div of Federal Screw Works)	500 Stephenson Hwy. Troy, MI 48084 (313) 588-2050 (800) 521-1350
Zilog	1315 Dell Ave. Campbell, CA 95008 (408) 370-8000