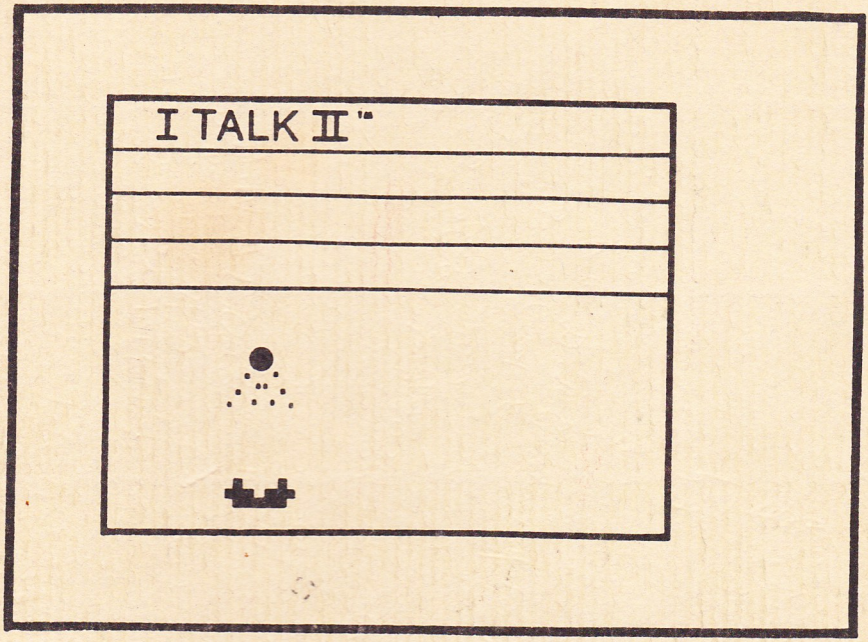


OWNER'S MANUAL

I TALK II™

ITALKII

WARRANTY INFORMATION

RealTime Electronics warrants the accompanying instrument to be free of defects in materials and workmanship for a period of one year from the date of retail purchase. This warranty is extended to the consumer of the instrument.

RealTime will repair or replace the instrument at no charge, if examination at the factory discloses no evidence of abuse, accident, or misuse. All transportation charges are at the expense of the owner. Enclose $2.00 in check or money order with your instrument to cover return shipping charges.

Return of the instrument to the owner, or notification to the owner of repair charges covering non-warranty damage, will be made within 30 days of receipt of the instrument at the factory.

ITALKII

TABLE OF CONTENTS

What To Do After You Open The Box

1.Connect Your ITALKII

Turn off your computer and disk drive. Locate the two blue connectors coming from **ITALKII** labeled "3" and "4". Plug these into the joystick ports on the front of your Atari labeled "3" and "4". Now unplug the black connector that's in your Atari's side serial port, and plug this connector into the back of the **ITALKII**. Plug the remaining **ITALKII** connector into the Atari's serial port. That's it!

You can leave **ITALKII** in place forevermore, and use your cassette and disk drive just like before. If you ever need joystick ports 3 and 4, just unplug those two connectors and leave your **ITALKII** as is. Simple.

2.Listen To Your ITALKII

DISK USERS: With your Basic cartridge installed, boot up SIDE B(reverse side) of the disk you received. A menu will automatically appear. Run the "DEMO" first. We will cover all the other neat things on your disk a little later on.

CASSETTE USERS: On SIDE B of the cassette, at around "120" on your counter (will vary a little from recorder to recorder) is the DEMO program. Type in RUN "C:" to run it.

You might want to run the DEMO a couple of times. Some people have no trouble comprehending **ITALKII**'s "accent", and some people need to listen to it a while to get used to it. In any case, the more you hear it, the easier it is to

understand.

WE HIGHLY RECOMMEND THAT YOU BACK UP YOUR **ITALKII** SOFTWARE IMMEDIATELY. WE CHOSE NOT TO SOFTWARE PROTECT SPECIFICALLY TO ALLOW BACKUPS. DO IT AS SOON AS POSSIBLE. (Disk users, use DOS menu item J,"Duplicate Disk".)

What's So Great About ITALKII?

Glad you asked.

Not only is **ITALKII** easy to install and use (as you just found out), it is also very versatile. If you want booming sound, you can connect the audio plug at the rear of **ITALKII** to the "AUX" input of your stereo amplifier. If you want to vary **ITALKII**'s volume or tone, use the small plastic "screwdriver" included with your unit. While listening to the "DEMO" program, adjust through the small holes labeled "V" and "T" on the bottom of **ITALKII**. It's fun to play around with the tone adjustment. You can make **ITALKII** sound like a chipmunk or a dim-witted giant. Select a voice that you find pleasing. This will be the voice you usually hear. When required, however, you can easily access **ITALKII**'s three other, higher pitched voices.

If you have the inclination, you may want to take a peek inside your **ITALKII**. Just remove the two screws holding the case together and lift off the top cover. Don't poke around in there-just look. We're proud of our well engineered unit. As you can see, **ITALKII** is very aesthetically pleasing, both inside and out. The quality and care that goes into your **ITALKII** speaks for itself.

Speaking of speaking for itself, you won't believe how easy it is to make your programs talk (cassette users, it's not quite this easy - you'll see later). After verifying that your "Disk Dictionary" contains all the words you're going to use (we'll talk about this later on), you

simply write your program. Put all desired speech output in a TALK$. For example, lines 980 and 1000 might read

```
980 TALK$="HELLO.  MY NAME IS
I TALK TOO.  I WILL BE YOUR
GUIDE."
1000 TALK$="YOU ARE IN A LARGE
ROOM, ADORNED WITH PURPLE AND
GOLD TAPESTRIES.  ON THE TABLE
IS AN IRIDESCENT CHALICE."
```

Now all you do is give your program to a utility called the "SENTENCE BUILDER". You'll get back your sentences in translated form - in "ITALKIIese, if you will. You then simply "Enter" into your program the machine language driver called "ITALKII.ENT", and you're ready to go. You'll find that the easiest part of writing your programs is adding speech. More details on this later.

One of **ITALKII**'s best features is the way it runs with Basic. It doesn't tie Basic up at all. **ITALKII** will speak while graphics are running, while sound effects are going, and so on. This makes **ITALKII** perfect for fast action games. Take a look at "WORDBLASTER" and you'll see what we mean.

Wordblaster (24KCassette, 32KDisk)

This arcade style game is a good example of how speech can enhance your programs. In fact, this is one game which is effectively made possible by synthesized speech.

As the game begins, **ITALKII** will ask you to spell a word. Using Joystick 1, shoot the proper letter from the three rows of letters racing across the top of the screen. Each time you make a correct hit, the letter will appear at the very top of the screen, along with a gold star. When you hit an incorrect letter, one gold star will disappear. The object of the game is to get 18 gold stars, at which time you'll get your Wordblaster classification. You can then either stop play, or continue from where you left off.

Wordblaster has several options. The rows of letters at the top of the screen can move at three different speeds, selectable at the game's onset. We'd recommend "SLOW" until you get used to the horizontal scrollong effects.

Disk Users: You can also select your spelling list. You can either utilize an old list, make a new list, or try your hand at some real toughies. The "old list" which comes with your unit is fairly simple. When you get bored with it, or you want to enter your child's weekly spelling list into the game, go for the "enter new list" option. This useful utility lets you enter up to 20 spelling words. Follow the prompts to create your new spelling list. However, prior to entering the list, you must note down the word and its associated encoded phonemes. This is where the Editor

comes in, as you'll soon see.
This new spelling list replaces the "old list" on disk, under the file name "D:WORDS". From that game on, this is your spelling list, unless you either change it or request the "hard words".
Cassette Users: The standard word list will load automatically with your program. Immediately following the Wordblaster program is a list of extremely difficult words. To play these words, **Reset** the program after the second half of the load is finished, then ENTER "C:". If you would like to enter your own words, LIST out lines 9001-9020 in the program. Here's the format:

```
9001 DATA COMPUTER,3Ib<URfgZjI
              spelling   encoded
                word     phoneme
                          string
```

Cassette users, we'd recommend that you make up your word list ahead of time. LIST "C:" the list, then ENTER "C:" it after Wordblaster finishes loading.

You'll notice that Wordblaster loads in two sections. That's done to minimize memory usage - there's an enormous amount of data statements and machine language routines which need not be present in the program's body after they are loaded into memory.

Hints and Comments:

1.Any time the letters are not moving, you can hear the spelling word again by pushing the joystick towards the screen.

2."Lead" your letter when you shoot, particularly on the faster options.

3.Try to avoid shooting near either

edge of the screen, unless you're aiming for the bottom row.

4.If you're stuck, hit the "?" on the bottom row. You'll get the next letter without any penalties whatsoever, but also without any stars.

5.On rare occasions, the projectile's explosion will scatter debris onto a nearby letter, causing Atari to think the wrong letter was hit. So who said life was fair?

6.On equally rare occasions, Atari will be off taking a snooze when you hit the letter. No penalties,no rewards - just shoot again.

7.When you have spelled the word correctly, **ITALKII** will let you know. Your classification at the end depends on the number of bad hits.

Happy Blasting!

Utilities

Following is a description of the powerful tools you have received with your **ITALKII**. These tools make it possible for you to add desired words to your Disk Dictionary and to translate English sentences into **ITALKII**ese.

Cassette users, you have an abbreviated utilities package. We're afraid cassettes and data bases don't make an ideal match. In the following, whenever we talk about saving a word to the Dictionary Disk, interpret that as meaning "Write it down".

EDITOR

Run the Editor program (DISK, side B - - - - CASSETTE,start of side B). The Dictionary Disk it's talking about on the opening screen is on side A of your disk. Insert side A, then choose #1 on the Editor Menu, "Edit New Word".

1.Edit New Word

Let's add a word to your dictionary. How about "LESS"? Type in "LESS", hit the RETURN key.

Now look up the word "LESS" in your Phonetic Speech Dictionary. Those letters following the "LESS" entry are its phonemes, its pieces so to speak. Type in the phonemes, separated by spaces instead of commas:

L EH1 EH3 S

The Editor is now converting the phonemes into an encoded phoneme string - **ITALKII**ese. This code will appear next to TALK$:

H2OO

If you refer to the back of this instruction booklet, you'll see the phonemes and their associated **ITALKII**ese codes. You can easily translate the code "H2OO" and see that it is, indeed, the phonemes for "LESS".

Press "Y",RETURN to listen. Doesn't sound quite right, does it? Sounds like it needs more "S" on the end. Type in "N" for listening, and "Y" for re-editing the phoneme. Edit just as you would in your programs - move the cursor out past the "S" and add another "S". Hit RETURN. Note that the TALK$ is now H2OOO. Listen again. Sounds pretty good. Time to save it to our Disk Dictionary:"N" listen,"N" re-edit,"Y" to save to current dictionary. The word "LESS" is now an official part of your dictionary.

Let's do it again. Select menu item #1, Edit New Word. Enter "LESS" and its phonemes. But this time, enter the phonemes in inverse character (use the key with the Atari logo). Listen to the word, and you'll notice the word sounds higher pitched than before. That's voice#2, as opposed to voice#1 which you heard earlier. In part #5 of this section, we'll discuss voices.

Now answer "N" to listen,"N" to re-edit, and "N" to save to current dictionary. The next prompt is "Save to Different Dictionary (Y/N)?" You'll use this option when you decide to save the word onto a dictionary other than the one you've been using during that editing session. You could conceivably have several different dictionaries - one for general use, one for war games, one for

science fiction games, and so on. When going from one dictionary to the next in an editing session, you should use the "Different Dictionary" option. You would then go back to the "Current Dictionary" option for saving subsequent words onto that dictionary.

Until you become familiar with the phonemes, you'll find the Phonetic Speech Dictionary to be a very valuable guide. Find the word in the Speech Dictionary, enter the phonemes, and then modify the phonemes as required to make the word sound correct. If you enter an invalid phoneme, you'll be notified.

By carefully editing each word going into your dictionary, you can maximize intelligibility. If you can't find your word in the Speech Dictionary, find one with similar sounds and construct your phonemes accordingly. Following are some hints for optimizing your words:

a)Put a short pause (PA0) before words beginning with CH,D,K,P,T,V, or Z.

b)Similarly, put a PA0 after words ending in CH,D,K,P,T,V, or Z.

c)Doubling up on L,M,N,S, and Z helps stress that sound.

d)An ER followed by an R sometimes makes a better "ER" sound.

e)To pluralize words ending in D, add Z S.

f)Putting U1 after an O will sometimes help the "O" sound.

g)Here's one that sometimes helps to put emphasis on the correct syllable: the longer versions on the phonemes, such as EH or UH, seem to produce more emphasis than the shorter forms EH1 or UH1. So you can use two or three short phonemes back to

back if you don't want emphasis on that syllable. The effectiveness of this method varies greatly from word to word.

h)If you want to enter a commonly used phrase in your dictionary, such as "egg sucking dog", separate the words with hyphens, not spaces: "egg-sucking-dog". Spaces are used as word delimiters by the Editor.

You'll be able to store about 3,000 words in your dictionary. That's probably more entries than you will ever need. As you will find out when you add speech to programs, the same common words tend to get used over and over. The 560 word dictionary that came with your **ITALKII** will probably cover most of your requirements.

Don't worry about hurting your dictionary by "overfilling" it. The Editor will let you know when the disk is full.

The last few pages of this manual contain a list of all the words in your Disk Dictionary, with their associated encoded phonemes. Cassette users will find this a very useful reference for incorporating speech into their programs. In fact, cassette users may want to keep updating this list as they edit new words.

2.Edit Old Word (not available on cassette)

Let's say that upon hearing a word in your program, you're not satisfied with the way it sounds. Or else you misspelled the word in your Disk Dictionary and you want to correct it. Select Editor Menu item #2, "Edit Old Word". Simply type in the word - let's use "LESS". The encoded phonemes will be pulled off disk and translated back to phonemes. You can then change the spelling of the word if desired. Answer

"N" to this, and go on to listen. Let's assume that you think there is too much "S" sound. Just re-edit and save to Dictionary, either Current or New. Note that this mode is very effective for pulling words off of one dictionary and relocating them onto another.

3.Print Words (not available on cassette)

If you select item #3 from the Editor Menu, you can see the words in your dictionary. They'll be displayed on the screen, 20 at a time, with their associated encoded phonemes. Any words which begin with a heart-shape have been deleted and can be cleaned up upon exiting (see part 7 of this section).

4.Delete Words (not available on cassette)

Let's remove that nasty word "LESS". Select item #4 from the Editor Menu, and type in "LESS". After the word is pulled from the disk, you can answer "Y" to delete it. Do so. Now go back and Print Words (item #3 from the Editor Menu). You'll see that heart-shape we mentioned previously.

5.Set Voices

If you get tired of listening to the same old voice, you can change it. Choose #5 from the Editor Menu. As you can see, Voice#1 is currently pitch level 1 and voice#2 is pitch level 2. Voice#2 is engaged by "inverse" phonemes, as you heard earlier.

Answer "Y" to change voices, and enter in any number from 1 to 4 for Voice#1. Do the same for voice#2. 1 is the lowest pitched and 4 is the highest pitched. All words you listen to subsequently with the Editor will be as per these voices you just

set.

6.Create Dictionary (not available on cassette)

If your old dictionary is full, or you want to create some specialty dictionaries, you'll use item #6 on the Menu. Put a formatted disk in the drive and answer "Y" to create the file called "DICTION.ARY". No words can be put on a disk unless this file has been created.

If you leave DOS off the Dictionary Disk, you'll allow more room for words. Put a copy of "Builder" on your new dictionary (see section on "Builder"), and you're ready to go.

7.Exit

To return to Basic, use item #7 on the Menu. Answer "Y" to clean up deleted words. This clean up does more than just that, however. It reorganizes the words on disk to allow for fastest access time. We'd recommend that you implement this function every 30 new words or so. If your dictionary is getting large (>1000 entries), make sure that the "D:MYFILE.LST" and "D:TALK.LST" residing on the Dictionary Disk aren't so large that they leave no room for your dictionary to be rewritten on disk. The number of free sectors left on your disk must be no smaller than the number of sectors occupied by the Disk Dictionary. A disk error will result if your disk doesn't have enough room for clean up. You may just want to delete unused files before cleaning up a large dictionary.

During clean up, one dot will appear on the display for each dictionary word. When "Exit" is complete, the READY prompt

appears and the Editor program is removed from memory.

SENTENCE BUILDER (not available on cassette)

This utility is located on side A of your disk, along with the dictionary. Since Builder uses a dictionary to do its building, we recommend it be present on all your Disk Dictionaries.

"D:BUILDER" makes it possible to add speech to your programs. Follow these steps:

1.Start your program at line #20.

2.Write your program so that TALK$ sentences reside alone on their program lines. No "GOTO", "PRINT", or anything else on that line other than the TALK$. Write out your TALK$ in standard sentences, such as

```
752 TALK$="GO AHEAD, TRY AND
HIT IT.  YOU CAN DO IT."
```

Spaces between words are ignored by Builder. A comma is treated like a long pause(PA1), and a period, exclamation point, or question mark is treated as a Stop phoneme followed by two long pauses. The words in your TALK$ sentences must appear **exactly** as they do in the Dictionary Disk.

The Disk Dictionary contains the suffixes ED, ES, EY, ING, LY, TEEN, TION, and WARD. You can use these suffixes to construct words not found in your dictionary. For example, "Seventeen" is not in your dictionary. But you can make it like this:


```
1020 TALK$="SEVEN TEEN"
```

3.Once your program is written, "SAVE" a backup copy on one of your development disks. This is always a good practice.

4.Now "LIST" your program onto the Dictionary Disk. We'd recommend that you use the same name everytime, such as "D:MYFILE.LST". This will maximize disk space available for new words.

5.Run "D:BUILDER". When it asks for the name of the file to convert, input your "LIST"ed program name. The next prompt asks if you wish to listen to the TALK$'s as they're translated, one by one. We'd recommend that you do listen. It gives you a chance to change subtleties, such as pauses between words, to increase intelligibility.

6.The Builder will take it from here. It could take a while, depending on both your program length and your dictionary length. Just kick back and relax. If you elect to listen to the sentences and edit as necessary, note that you must enter the **encoded** phoneme (ITALKIIese) in the editing session (refer to the list on the back page of this manual). Builder leaves spaces between the encoded phonemes to simplify editing. When editing, be sure to leave intact either the Stop("o") or Pauses("n") at the end of the TALK$. Without one of these, the last phoneme keeps right on sounding. Try it and you'll see what we mean.

Builder will progress on through the TALK$'s in your program. If it happens upon a word it can't translate, it will leave a space in its translation and inform you of the word. The discrepant words and

their line numbers will be displayed after Builder is finished.

7.Builder has now created a list called "D:TALK.LST", composed of all your translated sentences. Type "NEW", then "ENTER" your original program (it still resides, untouched, on the Dictionary Disk). Enter "D:TALK.LST", and your program now contains **ITALKII**ese instead of English words. The only thing you have left to do now is load the **ITALKII** machine language driver, and call it at the appropriate times.

CASSETTE USERS: Since you don't have a Builder, you have to enter in your TALK$ in **ITALKII**ese. Use the dictionary list at the back of this manual to provide you with the encoded phonemes for the various words you want. Leave spaces between the encoded words:

```
752 TALK$="K2SHSeggnnn E09Y
_O< [2N3o"
```

These spaces are ignored by the machine language driver, but make it easier for you to come back later and re-edit words. Be sure to end all of your TALK$'s with either a Stop("o") or a Pause("3" or "n"). Without one of these, the last phoneme doesn't shut up.

ITALKII DRIVER

This little gem is located on side B of your disk (CASSETTE-side B,around 100). Just **ENTER** "D:ITALKII.DRV" (ENTER "C:") into your program. It occupies lines 1 through 6. It may look like gobblygook to you, but it's music to Atari's ears. In the next section, we'll discuss how to call this routine.

Making Your Programs Talk Back

At this point, you've got ITALKIIese in your program with something called "ITALKII.DRV" located at the program's beginning. Now to make it talk.

As your program executes, it will encounter the various TALK$'s scattered throughout. Call the USR function as follows to engage speech:

```
752 TALK$="K2SHSeggo"
753 X=USR(ADR(DRIVER$),ADR(TALK$),
LEN(TALK$),VOICE1,VOICE2)
```

Don't worry about all that "ADR" and "LEN" stuff in the call - it just passes along certain necessary information to the machine language routine. Simply type that part in just as you see above. What you have to decide on is which pitch voice to use. Enter a number from 1 to 4 for VOICE1, and from 1 to 4 for VOICE2. As we discussed earlier, 1 is the lowest pitch and 4 is the highest pitch. VOICE2 is spoken whenever the machine language routine sees a TALK$ in "inverse". As you see, you can change voice complements at every USR call if you want to. In the programs we've written, however, we tend to use just one set of voices throughout. What we do is put the USR call right near the program's beginning, say line number 8, followed by a "RETURN" statement:

```
8 X=USR(ADR(DRIVER$),ADR(TALK$),
LEN(TALK$),1,2):RETURN
```


Now, whenever we want speech, we just

```
GOSUB 8
```

Pretty easy, huh?

If you call a new TALK$ before the old one is finished, the new one will rudely interrupt. If this is not desired, put in these lines:

```
9 IF LEN(TALK$)>PEEK(207) THEN 9
10 RETURN
```

When no interruptions are desired, just

```
GOSUB 9
```

Note that although the actual talking does not occupy Basic at all, this "waiting to finish" does. So if you are anxious to start the new TALK$, but you've got some graphics happening on the screen, you may want to just "poll" the status of the current TALK$ every so often:

```
1010 GOSUB DRAWLINE
1020  IF  LEN(TALK$)>PEEK(207)  THEN
1010
1030 TALK$="3UIkOOo"
1040 GOSUB 8
1050 GOSUB DRAWLINE
```

Advanced Programmers

This section is for those of you familiar with wierd things like assembly language and interrupt driven routines. Everyone else might want to skip over this section, unless you enjoy being bored to tears.

Tha **ITALKII** driver is a vertical blank interrupt driver routine. When the USR call is executed, the current VVBLKD at address $0224 is stored in page 0, at $CD and $CE. The VVBLKD is then changed to point to the **ITALKII** driver. Every 60th of a second, the driver checks to see if it needs to pull the next element from the TALK$. After finishing with that, it jumps to the routine whose address it had so wisely stored in $CD. When the driver reaches the end of the TALK$, it checks to make sure that its address is still in $0224. If it's not, that means that some other latecomer routine is busily running on 60 HZ interrupt. So good old **ITALKII** allows the other routine to finish up and reset the VVBLKD. Then **ITALKII** resets VVBLKD back to ($CD), and gets itself out of the 60 HZ loop.

If you plan on implementing your own vertical blank interrupt routines, be sure to nest them as described above. That way, some poor unsuspecting routine won't be cheated out of completing its work.

Note that the **ITALKII** driver uses page 0 locations $CB,$CC,$CD,$CE, and $CF. That really only leaves locations $D0 and $D1 for your program. Too bad Atari didn't get to leave more of page 0 open. The 6502's indirect addressing modes are certainly demanding little tykes.

What The Heck Is A Phoneme, Anyway?

Phonemes and words are like notes and music. Just as music can be broken down to a set of notes, so words can be broken down to a set of phonemes. In fact, with a set of about 64 of these little phonemes, virtually any Germanic word and most Romanic words can be synthesized (that's English, French, Italian, and Spanish, to name a few).

Lo and behold, guess how many phonemes **ITALKII** can articulate. Yep-64. This gives **ITALKII** a virtually unlimited vocabulary. Whereas a limited vocabulary is all that's needed by vending machines and elevators, a home computer needs a vast vocabulary. No telling what warped applications will next be forced on the poor, defenseless computer.

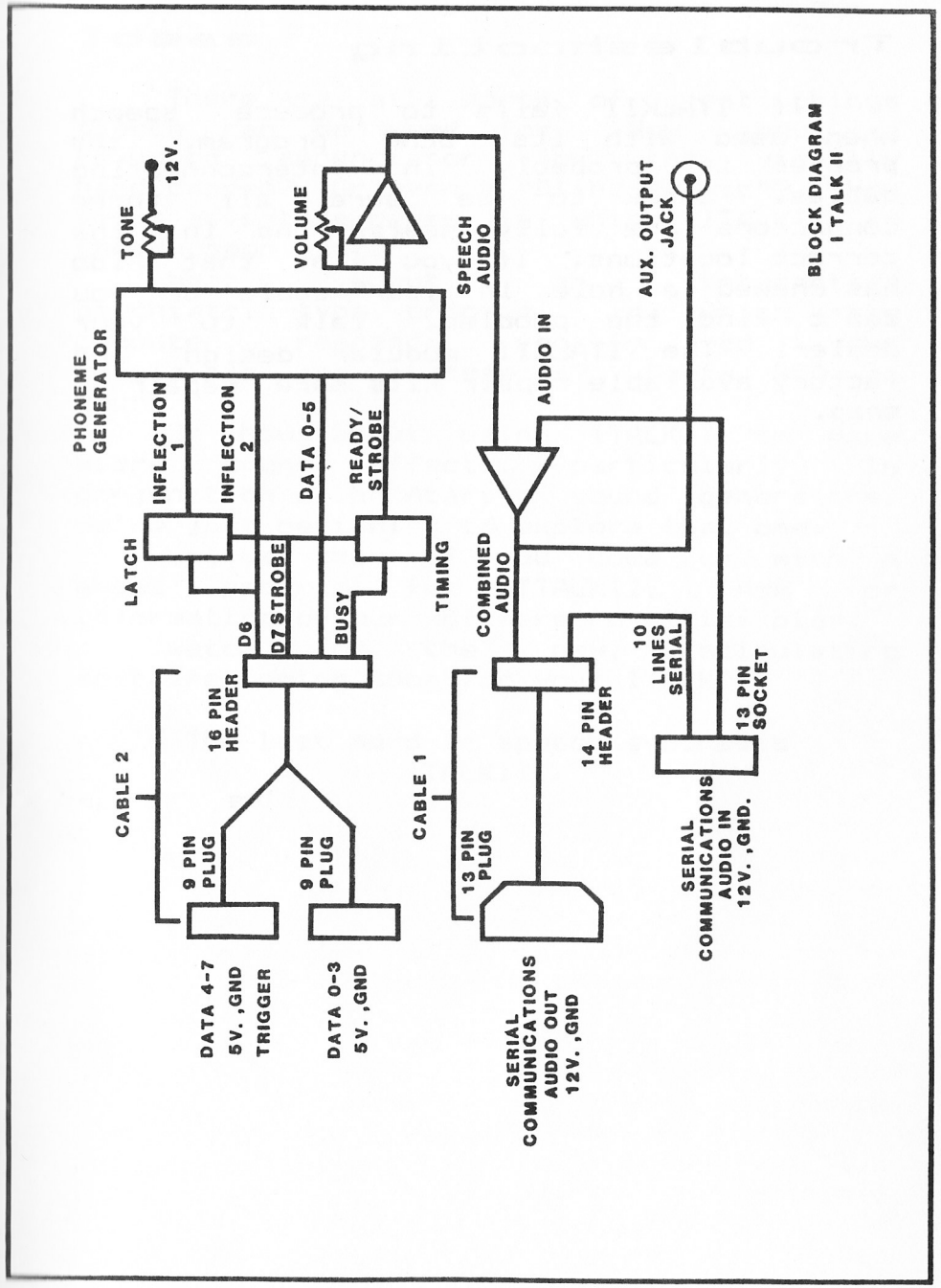
It's The Hardware, And A Lot More

If you're the curious type, you've probably already looked inside the ITALKII. If not, you'll be pleased to hear that there is a lot more inside than the weight would lead you to believe.

The largest component is a speech processor integrated circuit. That circuit modulates a fundamental frequency in a way dictated by phoneme data from your comuter. The output of the integrated circuit is a modulated frequency waveform - speech.

So what does all the rest of the circuitry do? That's where the real magic comes in. Those circuits allow you to use the speech processor without limiting any of the other capabilities of your system. Temporary data storage buffers and hardware/software handshake logic allow the speech synthsizer to operate without disturbing the action on the screen. Protection circuits guard the ITALKII against misconnected cables and component-killing static electricity. The audio stage allows you to adjust speech volume to match the volume of your computer's sound generators. It also adds in the sound track of your cassette player and then outputs the combined signal to your monitor and an auxiliary output jack. The auxiliary output can be connected to any power amplifier to produce booming sound.

The superior performance of ITALKII is due to not only its well engineered electronics, but also to its hardware/software interaction. A lot of technology for such a little box!



Troubleshooting

If ITALKII fails to produce speech when used with its DEMO program, the problem is probably in interconnecting cables. Check to be sure all three connectors are fully seated and in the correct locations. If you find that Fido has chewed a hole in your cable or you can't find the problem, talk to your dealer. The ITALKII modular design and factory available repair kits make repair a snap.

Ideas

There are all sorts of great things speech can be used for. How about a keyboard annunciator for the visually handicapped? Or even a "Blind Basic"? How about adventure games, in which ITALKII is the dungeon master?

What about an interactive psychiatric-type program, in which the computer responds vocally to your complaints and statements? Beats $50 an hour.

Or how about using ITALKII to make wierd sound effects, particularly in conjunction with Atari's sound generators. We're just beginning to explore that one.

Let us know if you come up with a great program for ITALKII. Ask for information on our software royalties plan.

Watch for the new, stimulating software coming soon for your ITALKII.

The last word in speech synthesis
ITALKII.

A	66Q
B	>\Y
C	O\Y
D	N\Y
E	\1Y
F	kMM
G	3NJ\Y
H	6QY3Z@
I	EO9Y
J	3NJO66Y
K	3IO6QY
L	2OSHH
M	21<<
N	21==
O	eVg
P	3U\Y
Q	II3RfX
R	Eaj
S	2100
T	3Z\Y
U	RfX
V	?\Y
W	Nb>SHRfX
X	21II300
Y	JJEO9Y
Z	B\Y

ONE	Jba==
TWO	3Zfgg
THREE	i[1Y
FOUR	Med[3
FIVE	MMEOY??
SIX	O;I300
SEVEN	Ok?:==
EIGHT	PYZ3
NINE	=EOY==
TEN	3Zk==
ELEVEN	2H2O?:==
TWELVE	3ZJ2OH??
THIRTEEN	ijZZ1Y==
FIFTEEN	MM;MZ\Y==
TWENTY	3ZJ2=Z1Y
THIRTY	ijN\
FORTY	Mdd[Z1Y
FIFTY	M;9MZ1Y
SIXTY	O;I3OZ1Y
SEVENTY	O2?:=N1Y
EIGHTY	55YZ1Y
NINETY	=EOY=Z1Y
HUNDRED	KKba=N[9N3
THOUSAND	iESgBO=N3
MILLION	<;HYS==
ZERO	BBQ;[eg
FIRST	Mj[OZ3
SECOND	O2Ia=N3
THIRD	ij[N3
FIFTH	M;9Mi
SUNDAY	Oba==N69Y
MONDAY	<a==N69Y
TUESDAY	3ZfggBBN69Y
WEDNESDAY	J2==BBN69Y
THURSDAY	ij[BBN69Y
FRIDAY	MM[EOYN69Y
SATURDAY	O_SZjN69Y
JANUARY	3NJ_O=Rfg00jY

FEBRUARY	MM1>Rfg11[Y
MARCH	<E[Z@@
APRIL	6YU[aH
MAY	<6QY
JUNE	3NJfgg==
JULY	3NJbHEOY
AUGUST	mL1OZ3
SEPTEMBER	O2OUZZO<>j[
OCTOBER	ESIZeg>j[
NOVEMBER	=eg?2O<>j[
DECEMBER	3NYO2O<>j[
ED	:N3
ES	:BB
EY	1Y
ING	:DL3
LY	H1
TEEN	Z1Y==
TION	AS==
WARD	JjN3
ABORT	b>ed[Z3
ABOVE	b>c??
ADJUST	bNJcOOZ3
AFTER	_OMZj[
AGAIN	cLk==
AHEAD	bKkN3
AIR	ii[[
ALL	mHH
ALONG	bHmDL3
ALSO	mHOFdg
ALWAYS	mHJ55YBB
AM	^O<<
AN	_O==
AND	_O==3N3
ANIMALS	_O=:<SHHBB
ANOTHER	b=bShj[
ANY	k==\
ARE	Eaj
AROUND	b[ES]==3N3
AS	_OBB
ASKED	_OOOI3Z3
AT	_OZ3
ATTENTION	bZk=Ab==
AWAY	bJPY
TH	ii
ACCOMMODATE	cIT<egN6QYZZ3
ALMOST	mH<Vg003Z3
ALARM	bHE[<<
BACK	3>^3I3
BAD	3>^_N3
BE	3>1Y
BECAUSE	3>1IcBB
BEDROOM	3>21N[gg<<
BEEN	3>;==
BEFORE	3>YMdd[
BEGIN	3>Y3LW==
BELOW	3>YHSeg
BETWEEN	3>YZJ1Y==
BICYCLE	>EO9YO;IbHH
BIG	3>;L3
BIRD	3>j[N3
BIRTHDAY	>j[ii3N69Y
BISCUIT	>:900I:Z3
BLACK	3>H_OI3
BLASTER	3>H_OOZj[
BLUE	>Hfgg
BOTH	3>Vgii3

```
BOY             3>VOY
BROWN           >[ESg==
BREAK           3>[6QI3
BUMBLE          3>b<<>aHH
BUT             3>bZ3
BUTTON          3>cZZO==
BY              3>EO9Y
BEGINNER        3>YLW==Sj
CAME            3I6QY<<
CAN             3I_O==
CANCEL          3I_O=OSH
CELEBRATE       OOIH:>[6YZ3
CHANGE          3Z@6QY=NJ3
CHECK           Z@2OI3
CLASSIFICATION          3IH_OO:M:I6QAa==
COLLOQUIALISM           3IaHdgI3J1SH:BS<<
COME            3Ic<<
COMPUTER        Ib<URfgZj[
CONSENSUS       3Ib=Ok=O;OO
CONTINUE        3Ib=ZW=Rfg
CONTROL         3Ib=Z[edH
CORRECT         3IV[2OI3Z3
COSMIC          3ICBB<:I3
COULD           3IGN3
CRASH           3I[_OAA
CREATE          3I[Y6YZ3
CYCLOPS         OEOYIHEUOO
CIRCUIT         O[[I:Z3
COLOR           3IaaHj
DANGER          3N6QY=NJj[
DATE            3N6QZZ3
DAYS            3N69YBB
DECREASE        3N\I[1YOO
DEFEND          3NYMk==NN3
DESTROY         3NYOZ[eS9Q
DID             3N;N3
DIFFERENT       3N;M[O=Z3
DIGITAL         3N;NJ9ZSH
DICTIONARY      3N;I3Aa=OOjY
DISK            3N;OOI3
DO              3Nfgg
DOLLARS         NESHjBB
DOOR            3Ned[
DOWN            3NSEg==
DRIVE           3N[EOY??
DROP            3N[ESU3
DEGREES         3NYL[\BB
DESICCATE       3N2OO;II6QZZ3
DUNGEON         NbS==NJ;==
EACH            \QZ@@
EARTH           j[iii
EAT             \Z3
EMERGENCY       1<jNJ:=OO1Y
EMPTY           k<UZ1Y
END             k=NN3
ENTER           k=Zj[
EQUAL           \I]aHH
EXCELLENT       kI3OaHa=Z3
EXIT            kL3B;9Z3
EXPLODE         2I3OUHVgNN3
EYES            EO9YBB
EVERY           2?[1Y
EDITOR          2N;Zj[
ENTER           k=Zj[
DON'T           3NVg=3Z3
FAIL            MM6Q9SH
FAR             MTS[[
```



```
FAST            M_OOZ3
FEET            M1YZ3
FEW             MRfX
FILE            ME09SH
FIND            MEOY=NN3
FIRE            MEOQj[
FLOWER          MMHEdgJj[
FLY             MHEOY
FOOD            MggNN3
FOR             Med[
FORCE           Med[OO
FORWARD         Mdd[]jN3
FOUND           MES]=N3
FRIEND          M[2O==N3
FROM            M[bS<<
FUEL            MRfgSHH
FUTURE          MRfgZ@j[
FANTASTIC       M_O=Z_OOZ:I3
GALAXY          L^8HaIOO11
GAME            L6QY<<
GALLONS         L_8HS==BB
GET             3LkZ3
GIVE            3LW??
GLOW            3LHeg
GO              3Leeg
GOING           3LV:D3
GOOD            3LGN3
GOT             3LEZ3
GREATER         3L[6Y3Zj[
GREEN           3L[1Y==
GUITAR          3L:ZEaj
GHOST           3LegOOZ3
GRAY            3L[6QY
HAD             K_ON3
HAND            K_O==NN3
HARD            KKE[N3
HAS             KK_OBB
HAVE            3K^S??3
HE              3K1Y
HEART           KKES[ZZ3
HELP            3K2OHU3
HERE            3K19[
HIGH            3KKEOY
HOLE            KKVHH
HOME            3KVg<<
HONEY           Kb==1Y
HOUSE           KK8dgOO
HUMAN           3KRfg<1==
HYPER           KKEOYUj[
HANG            K_9DL3
IF              WMM
IMPORTANT       ;<UV[ZO=Z3
IMPOSTOR        ;<UTOZj[
IN              ;9==
INPUT           ;==UGZ3
INSECT          ;=O2IZ3
INTERRUPT       ;=ZjSbUZ3
INTRUDER        ;=Z[fggNj[
INVALID         ;=?_'H;N3
IS              ;9BB
IT              ;9Z3
INSERT          ;=O[[Z3
INSTRUCTIONS            W=OZ[bI3Aa==BB
JEWEL           3NJfggHH
JUST            3NJbOZ3
```

KEEP	3I\U3
KEY	3I\Y
KNOW	=Feg
KNOWLEDGE	=ESH9NJJ3
LAND	H_0==N3
LARGE	HE[NJJ
LAST	H_0OZ3
LEFT	H2MMZ3
LETTER	H20Zj[
LEVEL	H20?SH
LIFE	HS8YMM
LIGHT	HS8YZ3
LIKE	HS8YII3
LIQUEFY	H;I3]9ME0Y
LOCK	HESI3
LONG	HmDL3
LOST	HmOZ3
LOW	HSVg
LUCK	HbII3
MACHINE	<bA1Y==
MAKE	<6YII3
MAN	<_0==
MANY	<2==\
MAY	<69Y
MAYONNAISE	<69YS==69YBB
ME	<\Y
MEGA	<kLbS
MEMORY	<k<j\
MICRO	<B0Y3I[eg
MIGHT	<B0YZ3
MINUS	<B0Y=000
MINUTES	<;=:Z00
MISSPELL	<:00U20HH
MONSTER	<C=0Zj
MONTH	<b=3ii
MORE	<dd[
MOST	<eg0Z3
MOVE	<gg??
MUCH	<bSZ@
MUST	<bS0Z3
MULTIPLY	<bHZ9UHEY
MY	<E0Y
NAME	=6QY<<
NEAR	=19[[
NEED	=1YN3
NEGATIVE	=2L0Z;??
NEUTRAL	=fgZ[SHH
NEVER	=2?j[
NEW	=fgg
NEXT	==20I30Z3
NIGHT	=S8YZ3
NO	=Feg
NOISE	=eS9QBB
NORMAL	=dd[<SHH
NORTH	=dd[ii
NOT	=ESZ3
NOTICE	=egZ;00
NOW	=ESX
O'CLOCK	dg3IHCI3
OF	c??
OFF	mMM
OFTEN	''M9==
OLD	VHHN3
ON	ES==
ONCE	]b=Z00
ONLY	V=HY

OPEN	eU:==
OR	dd[
ORANGE	dd[;=NJJ
OTHER	bShj[
OUR	Eaj
OUT	S8gZ3
OVER	ed?j[
OXYGEN	ESI309NJ:==
CLASS	3IH^000
CONGRATULATIONS	3Ib=L[_N3JaH6QAS=BB
EXTRATERRESTRIAL	kI0Z[b3Zj[20Z[1bHH
HELLO	K2SHSegg
LIZARD	H;Bj[N3
NICE	=S8Y00
NUMBER	=b<>j
PAPER	U6YUj[
PART	3UE[Z3
PASS	U^000
PASSED	U^000Z3
PAST	U^000Z3
PAYING	3U6QY1=L3
PER	3Uj[
PICK	3U;9II3
PIECE	U1Y00
PLACE	UH6Y00
PLANT	3UH_0=Z3
PLEASE	3UH11BB
PLUS	UHba00
POINT	3UeS9Q=Z3
POINTS	UeS9Q=Z00
POSITIVE	UESB:Z;??
POUND	3UES]=NN3
POWER	3UES]j[
PRESS	3U[k00
PRESSURE	U[2Aj[
PREVENT	U[Y?20=Z3
PROBLEM	U[ES>HS<<
PROCEED	U[e01YN3
PROGRAM	3U[VL[_0<<
PROTECT	U[bZ20IZ3
PULL	UFFH
PURCHASE	U[[4@:00
PURE	3URfj[
PURPLE	3UjUSHH
PUSH	UFfAA
PUT	3UGZ3
QUEEN	3II]]11==
QUESTION	3I]2OZ@S==
RABBIT	[_0>;Z3
RAINBOW	[6QY=3>Vdg
RANGE	[6QY=NJJ
RANK	[_9D3I3
RATING	[6QZ1DL3
READ	[1YN3
READY	[kN1Y
REAL	[\SHH
RECORD	[[20Ij[N3
RED	[2N3
REPAIR	[YUk[[
RESCUE	[20IRfg
REST	[20Z3
RESUSCITATE	[10a0:Z6YZ3
RETURN	[1Zj[==
RIGHT	[S8YZ3
ROW	[Vg
RUN	[b==

SAME	U6QY<<
SAY	O69Y
SCAN	OI^O==
SCORE	OIddI
SECRET	O1II9Z3
SEE	O1Y
SELL	OkHH
SEND	O2O=N3
SET	OkZ3
SHAKE	AA6QI3
SHIP	A;U3
SHOOT	AfgZ3
SHOOTING	AfgZ:D3
SHOT	AESZ3
SHOULD	AfffN3
SHOW	Aeg
SICK	OO;I3
SIDE	OE0YN3
SINCE	O;=OO
SKELETON	OI2H:Z:==
SLAVE	OH6QY??
SLOW	OHddg
SMALL	O<mHH
SMELL	O<kHH
SMOKE	O<egI3
SO	OFdg
SOME	Oc<<
SORT	OddIZ3
SOUND	OESJ=NN3
SOUTH	OESgii
SPACE	OU6QYOO
SOON	OXg==
SPACESHIP	OU6QYOA;9U3
SPEAK	OU\I3
SPEED	OU1YN3
SPELL	OU2SHH
SPELLING	OU2SHH:DL3
STAR	OZESI
START	OZEIZ3
STEREO	OZ1OI1YeVg
STILL	OZ:HH
STORY	OZVIYY
SUBTRACT	Oba>ZI_OIZ3
SUCH	ObZ@@
SUM	Oc<<
SUN	Oc==
SUPER	OfgUjI
SUPERSEDE	OfgUjO1YNN3
TAKE	3Z6QYI3
TALK	3ZmI3
TELEPHONE	3Z2H:MVg==
TELEVISION	3Z2OHa?:AS==
TELL	3Z2HH
TEMPERATURE	3Z2<UI9Z@jI
TERMINAL	3Zj<O=aH
THAN	h_O==
THAT	h_OZ3
THAT'S	h_OZO
THE	hhbb
THEN	hk==
THERE	h11I
THING	i;9DL3
THINK	ii;9=II3
THIS	h;OO
THROUGH	iIfX
TIME	3ZEOY<<

TO	3Zfgg
TODAY	3ZgN69Y
TOMORROW	3Zfg<EIIeg
TOO	3Zfgg
TORCH	3Zdd[Z@@
TOTAL	3ZegZSHH
TOUCH	3ZcZ@
TRANSPORT	3ZI_O=OUddIZ3
TROUBLE	3ZIb>SHH
TRUE	3ZIfgg
TRY	3ZIE09Y
TRYING	3ZIE09\DL3
TURN	3ZjI==
TYPE	3ZS8YU3
UH	c
UNBELIEVABLE	c=>1RH\?b>bHH
UNDER	aa=Nj
UNIVERSE	Rfg=9?jOO
UNTIL	aa=Z;HH
URGENT	IINJ9=Z3
USE	RfggBB
VECTOR	??2I3ZjI
VERY	?2IY
VIBRATE	??EOY>I6QZ3
VOICE	?eS9QOO
WAIT	J6YZ3
WALK	JCII3
WANT	JES==Z3
WARNING	Jdd[=:DL3
WARP	JddIU3
WAS	JbSBB
WASN'T	JbB==Z3
WATER	JESZjI
WAY	J55Y
WEAK	J1YII3
WEAPON	J11Ub==
WEEK	J1YII3
WEIGH	J55Y
WELCOME	JkHIc<<
WENT	J20=Z3
WEST	J2OOOZ3
WHAT	JSbZ3
WHEN	J20==
WHERE	JO5OI
WHICH	J;9Z@@
WHITE	JS8YZ3
WHY	JE09Y
WILL	J;9HH
WINDOW	J;=Neg
WINTER	J;=ZjI
WISH	J;AA
WITH	J;9ii
WON	JbS==
WORD	JJjIN3
WORK	JjIII3
WORKER	JjIIjI
WORLD	JjSHN3
YEAR	RQ9I
YELLOW	R2OHSVg
YES	RO2OO
YET	R2OZ3
YOU	Rfgg
YOU'LL	RfggSHH
YOUR	YddII
ZONE	BBVg==
LOOK	HF3I3
PAVILION	3US5?;SHRa==
UP	cU3
DAY	3N6QY

PHONEME CODES

Phoneme Symbol	ITALKII "ese"	Duration (ms)	Example Word
EH3	/0	59	jacket
EH2	1	71	enlist
EH1	2	121	heavy
PA0	3	47	no sound
DT	4	47	butter
A2	5	71	make
A1	6	103	pail
ZH	7	90	pleasure
AH2	8	71	honest
I3	9	55	inhibit
I2	:	80	inhibit
I1	;	121	inhibit
M	<	103	mat
N	=	80	sun
B	>	71	bag
V	?	71	van
CH	@	71	chip
SH	A	121	shop
Z	B	71	zoo
AW1	C	146	lawful
NG	D	121	thing
AH1	E	146	father
OO1	F	103	looking
OO	G	185	book
L	H	103	land
K	I	80	trick
J	J	47	judge
H	K	71	hello
G	L	71	get
F	M	103	fast
D	N	55	paid
S	O	90	pass
A	P	185	tame
AY	Q	65	jade
Y1	R	80	yard
UH3	S	47	mission
AH	T	250	mop
P	U	103	past
O	V	185	cold
I	W	185	pin
U	X	185	move
Y	Y	103	any
T	Z	71	tap
R	[	90	red
E	\	185	meet
W	]	80	win
AE	^	185	dad
AE1	_	103	after
AW2	[ctrl].	90	salty
UH2	a	71	about
UH1	b	103	uncle
UH	c	185	cup
O2	d	80	bold
O1	e	121	aboard
IU	f	59	you
U1	g	90	June
THV	h	80	the
TH	i	71	thin
ER	j	146	bird
EH	k	185	ready
E1	l	121	be
AW	m	250	call
PA1	n	185	no sound
STOP	o	47	no sound

REAL TIME

RealTime Electronics
2339 West Obispo Avenue
Mesa, Arizona 85202

GREENBRIER MARKETING
INTERNATIONAL INC.
8225 East Rovey Ave.
Scottsdale,AZ 85253
(602)948-0005

I TA LK II

PHONETIC SPEECH DICTIONARY for the SC-01 SPEECH SYNTHESIZER

INTRODUCTION

VOTRAX® Speech Synthesis Technology produces each of the 45 basic speech sounds, called phonemes. This handy dictionary gives you quick access to the VOTRAX® phoneme sequences used to create approximately 1400 words. Because VOTRAX® speech synthesis uses these basic phonetic sounds, you may program virtually any word in the English language by learning to use and combine the various phoneme codes.

This dictionary is intended for use with the SC-01 Speech Synthesizer. It can also serve as a guide for programming any product containing a VOTRAX® synthesizer.

GENERAL DESCRIPTION OF THE SC-01 CHIP

The SC-01 Speech Synthesizer is a completely self-contained solid state device. This single chip phonetically synthesizes continuous speech, of unlimited vocabulary, from low data rate inputs.

Speech is synthesized by combining phonemes (the building blocks of speech) in the appropriate sequence. The SC-01 Speech Synthesizer contains 64 different phonemes which are accessed by a 6-bit code. It is the proper sequential combination of these phoneme codes that creates continuous speech.

PHONEME DESCRIPTION

Table 1 lists the 64 phonemes produced by the SC-01. Each sound is represented by its VOTRAX® phoneme code and is accompanied by its phoneme symbol and an example. The underlined segments of the example word demonstrate the phoneme use, i.e., sound to be pronounced.

Table 2 provides the phoneme sequences used to produce vowels in the group called dipthongs, (2 vowel sounds in sequence, identified as a single sound, e.g., the long "i" vowel).

Phonetic Programs	
A	A1, AY, Y
a-2	UH2, UH3
able	A1, Y, B, UH3, L
abort	UH1, B, O2, O2, R, T
about	UH1, B, UH2, AH2, U1, T
above	UH1, B, UH1, UH3, V
accept	EH1, K, PAO, S, EH1, EH3, P, T
access	AE1, EH3, K, PAO, S, EH1, EH3, S
account	UH1, K, AH1, UH3, W, N, T
acid	AE1, EH3, S, I1 D
act	AE1, EH3, K, T
active	AE1, EH3, K, T, I1, V
actual	AE1, EH3, K, T, CH, U1, UH3, L
add	AE1, EH3, D
address	AE1, EH3, D, R, EH1, EH3, S
ade	(use "aid" program)
adjust	UH1, D, J, UH1, UH3, S, T
adjucent	UH1, D, J, A1, AY, S, EH3, N, T
advance	AE1, EH3, D, V, AE1, EH3, N, T, S
advise	AE1, EH3, D, V, AH1, EH3, Y, Z
affect	UH1, F, EH1, EH3, K, T
after	AE1, EH3, F, T, ER
again	UH1, G, A2, EH1, N
age	A1, AY, Y, D, J
agent	A1, Y, D, J, EH3, N, T
ahead	UH1, H, EH1, EH3, D
aid	A1, AY, Y, D
air	EH2, EH2, R
alarm	UH1, L, AH1, R, M
alert	UH1, L, ER, R, T
all	AW, L
allocate	AE1, UH3, L, UH2, K, A1, Y, T
allow	UH1, L, AH1, UH3, U1
alpha	AE1, AW2, L, F, UH1
already	AW, L, R, EH1, EH3, D, Y
also	AW, L, S, O1, U1
altitude	AE1, UH3, L, T, I2, T, IU, U1, U1, D
aluminum	UH1, L, IU, U1, M, I3, N, UH1, M
am	AE1, EH3, M
america	UH1, M, EH1, R, I3, K, UH2, UH3
amount	UH1, M, AH1, UH3, W, N, T
amp	AE1, EH3, M, P
amplify	AE1, EH3, M, P, L, I3, F, AH1, EH3, AY
an	AE1, EH3, N
and	AE1, EH3, N, D
angle	AE1, EH3, NG, G, UH3, L
another	UH1, N, UH1, UH3, THV, ER
answer	AE1, EH3, N, S, ER
any	EH2, EH2, N, Y
apostrophe	UH1, P, AH1, UH3, S, T, R, UH3, F, Y
approach	UH1, P, R, O1, U1, T, CH
approve	UH1, P, R, IU, U1, U1, V
approximate	UH1, P, R, AH1, K, PAO, S, EH3, M, I3, T
approximate-2	UH1, P, R, AH1, K, PAO, S, EH3, M, A2, Y, T
april	A1, Y, P, R, UH2, L
architect	AH1, R, K, UH2, T, EH3, EH2, K, T
are	(see "R" program)
area	EH1, EH3, R, Y, UH1
arrive	UH1, R, AH1, EH3, Y, V
arrow	EH1, EH3, R, O1, U1
article	AH1, R, T, EH3, K, UH3, L
as	AE1, EH3, Z
ASCII	AE1, EH3, S, K, Y
ask	AE1, EH3, S, K
assemble	UH1, S, EH1, EH3, M, B,
asset	AE1, EH3, S, EH1, T
assign	UH1, S, AH1, EH3, Y, N
assist	UH1, S, I1, I3, S, T
associate	UH1, S, O1, SH, Y, A1, Y, T
associate-2	UH1, S, O1, SH, Y, I2, T
assume	UH1, S, IU, U1, M
at	AE1, EH3, T
ate	(see "eight" program)
attach	UH1, T, AE1, EH3, T, CH
attempt	UH1, T, EH1, EH3, M, P, T
attend	UH1, T, EH1, EH3, N, D
audio	AW, D, Y, O1, U1
august	AW2, AW2, G, EH2, S, T
authorize	AW2, AW2, TH, ER, AH1, Y, Z
automatic	AW2, AW2, DT, UH3, M, AE1, EH3, DT, I3, K
available	UH1, V, A1, Y, L, UH3, B, UH3, L
average	AE1, EH3, V, R, I1, D, J
avoid	UH1, V, O1, UH3, I3, AY, D
B	B, E1, Y
back	B, AE1, AE1, K
bad	B, AE1, AE1, D
badge	B, AE1, AE1, D, J
bag	B, AE1, AE1, G
balance	B, AE1, AH2, L, I3, N, DT, S
ball	B, AW2, AW1, L
band	B, AE1, EH3, N, D
bank	B, AE1, I3, NG, K
bar	B, AH1, UH3, R
base	B, A1, AY, Y, S
basic	B, A1, Y, S, I2, K
bat	B, AE1, EH3, T
batch	B, AE1, EH3, T, CH
bath	B, AE1, AE1, EH3, TH
battery	B, AE1, EH3, T, ER, Y
be	(use "B" program)
bed	B, EH1, EH3, D
been	B, EH1, EH3, N
beep	B, E1, Y, P
before	B, Y, F, O2, O2, R
begin	B, Y, G, I1, I3, N

bell	B, EH1, UH3, L
below	B, Y, L, UH3, O2, U1
bend	B, EH1, EH3, N, D
best	B, EH1, EH3, S, T
beta	B, A2, A2, AY, T, UH2
better	B, EH1, EH3, T, ER
between	B, Y, T, W, E1, Y, N
bid	B, I1, I3, D
big	B, I1, I3, G
bill	B, I1, I3, L
billion	B, I1, I3, L, Y, UH3, N
bin	B, I1, I3, N
binary	B, AH1, Y, N, EH3, EH3, ER, Y
birthday	B, ER, R, TH, D, A1, I3, Y
bit	B, I1, I3, T
bite	B, UH3, AH2, Y, T
black	B, L, AE1, EH3, K
blank	B, L, AE1, EH3, NG, K
blew	(use "blue" program)
blind	B, L, AH1, EH3, Y, N, D
block	B, L, AH1, UH3, K
blown	B, L, O1, U1, N
blue	B, L, IU, U1, U1
blur	B, L, ER, R
board	B, O1, O2, R, D
bolt	B, O2, O2, L, T
bond	B, AH1, UH3, N, D
book	B, OO1, OO1, K
bored	(use "board" program)
boss	B, AW1, AW2, S
bother	B, AH1, UH3, THV, ER
bottom	B, AH1, UH3, T, UH1, M
bought	B, AW1, AW2, T
box	B, AH1, UH3, K, PAO, S
brace	B, R, A1, Y, S
brain	B, R, A1, Y, N
brake	B, R, A1, Y, K
branch	B, R, AE1, EH3, N, T, CH
bravo	B, R, AH1, UH3, V, O1, U1
break	(use "brake" program)
bridge	B, R, I1, I3, D, J
brief	B, R, AY, Y, F
bright	B, R, UH3, AH2, Y, T
bring	B, R, I1, I3, NG
broke	B, R, O1, U1, K
brought	B, R, AW, T
brown	B, R, AH1, UH3, U1, N
bubble	B, UH1, UH2, B, UH3, L
budget	B, UH1, UH3, D, J, I2, T
bug	B, UH1, UH2, G
build	B, I2, I2, L, D
bus	B, UH1, UH2, S
business	B, I3, I3, Z, N, EH2, S
busy	B, I3, I2, Z, Y
but	B, UH1, UH2, T
button	B, UH1, UH3, T, EH3, N
buy	B, AH1, EH3, I3, Y
by	B, AH1, EH3, I3, Y
bye	B, AH1, EH3, I3, Y
byte	(use "bite" program)
C	S, E1, Y
cable	K, A1, Y, B, UH3, L
calendar	K, AE1, UH3, L, I3, N, D, ER
calibrate	K, AE1, UH3, L, UH3, B, R, A1, Y, T
call	K, AW2, AW1, L
came	K, A1, AY, Y, M
can	K, AE1, EH3, N
cancel	K, AE1, EH3, N, S, UH3, L
capable	K, A1, Y, P, UH3, B, UH3, L
capacitor	K, UH2, P, AE1, EH3, S, EH3, T, ER
capacity	K, UH2, P, AE1, EH3, S, I3, DT, Y
car	K, AH1, UH3, R
card	K, AH1, R, D
care	K, EH3, EH3, ER
carpenter	K, AH1, R, P, I3, N, D, ER
carriage	K, EH2, EH3, R, I1, D, J
carry	K, EH2, EH3, R, Y
carton	K, AH1, R, T, I3, N
case	K, A1, AY, Y, S
cash	K, AE1, EH3, SH
cassette	K, UH1, S, EH1, EH3, T
cassette-2	K, A2, AY, S, EH1, EH2, T
category	K, AE1, EH3, DT, UH3, G, O1, R, Y
catalog	K, AE1, EH3, DT, UH3, L, AW2, AW2, G
caution	K, AW2, AW1, SH, UH3, N
cent	S, EH1, EH3, N, T
center	S, EH1, EH3, N, T, ER
centi	S, EH1, EH3, N, T, I1, I3
centigrade	S, EH1, N, T, I3, G, R, A1, Y, D
certify	S, R, R, T, I3, F, AH1, Y
change	T, CH, A1, AY, Y, N, D, J
character	K, EH1, R, EH1, K, T, ER
charge	T, CH, AH1, R, D, J
charlie	T, CH, AH1, R, L, Y
chart	T, CH, AH1 R, T
check	T, CH, EH1, EH3, K
cheer	T, CH, AY, I2, R
chip	T, CH, I1, I3, P
choice	T, CH, O1, UH3, I3, AY, S
circle	S, ER, R, K, UH3, L
circuit	S, R, R, K, I2, T
city	S, I1, T, Y
claim	K, L, A1, AY, Y, M
class	K, L, AE1, EH3, S
clean	K, L, E1, AY, N
clear	K, L, AY, I3, R
clerk	K, L, ER, K
clip	K, L, I1, I3, P
clock	K, L, AH1, UH3, K
close	K, L, UH3, O1, U1, Z
close-2	K, L, UH3, O2, U1, S
cloud	K, L, AH1, UH3, W, D
coarse	K, O1, O2, R, S
code	K, OO1, O2, U1, D
coin	K, O1, UH3, I3, AY, N

collar	K, AH1, UH3, L, ER
collect	K, UH1, L, EH1, K, T
colon	K, OO1, O2, U1, L, I2, N
color	K, UH2, UH2, L, ER
column	K, AH1, UH3, L, UH3, M
combine	K, UH2, M, B, AH1, EH3, Y, N
comma	K, AH1, UH3, M, UH1
command	K, UH2, M, AE, EH3, N, D
commerce	K, AH1, UH3, M, ER, S
commercial	K, UH1, UH3, M, ER, SH, UH3, L
communicate	K, UH2, M, Y1, IU, U1, N, I3, K A1, Y, T
company	K, UH1, UH3, M, P, EH3, N, Y
compare	K, UH1, UH3, M, P, EH3, EH3, ER
compile	K, UH1, UH3, M, P, AH1, EH3, I3, UH3, L
complete	K, UH1, UH3, M, P, L, AY, Y, T
comply	K, UH1, UH3, M, P, L, AH1, EH3, Y
component	K, UH2, M, P, O2, O1, N, EH2, N, T
computer	K, UH1, M, P, Y1, IU, U1, T, ER
conceal	K, UH1, N, S, E1, AY, L
condense	K, UH1, N, D, EH1, EH3, N, S
condition	K, UH1, N, D, I1, I3, SH, UH3, N
confirm	K, UH1, N, F, ER, R, M
confuse	K, UH1, N, F, Y1, IU, U1, U1, Z
confusion	K, UH1, N, F, Y1, IU, U1, U1, ZH, UH3, N
congratulations	K, UH1, N, G, R, AE1, D, J, UH3, L, A1, AY, SH, UH3, N, Z
connect	K, UH1, N EH1, EH3, K, T
console	K, AH1, UH3, N, S, O1, U1, L
console-2	K, UH1, N, S, O1, O2, L
consult	K, UH1, N, S, UH1, UH2, L, T
consume	K, UH1, N, S, IU, U1, U1, M
contain	K, UH3, UH3, N, T, A1, AY, Y, N
continue	K, UH1, N, T, I1, I3, N, Y1, IU, U1
contract	K, AH1, UH3, N, T, R, AE1, EH3, K, T
contrast	K, AH1, UH3, N, T, R, AE1, EH3, S, T
control	K, UH1, N, T, R, O1, O2, L
convenient	K, UH2, N, V, E1, N, AY, EH3, N, T
copper	K, AH1, UH3, P, ER
copy	K, AH1, UH3, P, Y
correct	K, O2, O2, R, EH1, EH3, K, T
correspond	K, O1, R, I3, S, P, AH1, AH2, N, D
cosine	K, O1, U1, S, AH1, Y, N
cost	K, AW2, AW1, S, T
could	K, IU, IU, OO1, D
count	K, AH1, UH3, W, N, T
country	K, UH1, N, T, R, Y
couple	K, UH3, UH1, P, UH3, L
courage	K, ER, R, I3, D, J
course	K, O1, O2, R, S
court	K, O1, O2, R, T
cover	K, UH1, UH3, V, ER
crane	K, R, A1, AY, Y, N
crash	K, R, AE1, EH3, SH
crease	K, R, E1, Y, S
create	K, R, Y, A1, Y, T
creation	K, R, Y, A1, Y, SH, UH3, N
credit	K, R, EH1, EH3, D, I1, T
crew	K, R, IU, U1, U1
critical	K, R, I1, T, I3, K, UH3, L
cross	K, R, AW, S
crowd	K, R, AH1, UH3, U1, D
cry	K, R, AH1, EH3, I3, Y
cue	(use "Q" program)
cup	K, UH1, UH2, P
curious	K, Y, ER, Y, UH1, S
current	K, ER, R, EH3, N, T
currency	K, ER, R, I2, N, DT, S, Y
curse	K, ER, R, S
curve	K, ER, R, V
customer	K, UH1, UH2, S, T, UH1, M, ER
cut	K, UH1, UH2, T
cycle	S, UH3, AH2, Y, K, UH3, L
D	D, E1, Y
daily	D, A1, AY, Y, L, Y
damage	D, AE1, EH3, M, I1, D, J
danger	D, A1, AY, Y, N, D, J, ER
dark	D, AH1, R, K
dash	D, AE1, EH3, SH
data	D, A1, Y, DT, UH1
date	D, A1, AY, Y, T
day	D, A1, I3, Y
dead	D, EH1, EH3, F
dealer	D, E1, AY, L, ER
dear	D, AY, I3, R
debit	D, EH1, EH3, B, I2, T
debt	D, EH1, EH3, T
december	D, Y, S, EH1, EH3, M, B, ER
decide	D, Y, S, AH1, EH3, Y, D
decimal	D, EH1, S, M, UH3, L
decision	D, Y, S, I1, ZH, UH3, N
decline	D, Y, K, L, AH1, EH3, Y, N
decrease	D, Y, K, R, E1, Y, S
deduct	D, Y, D, UH1, UH2, K, T
deep	D, E1, Y, P
deer	(use "dear" program)
defeat	D, Y, F, E1, AY, T
defend	D, Y, F, EH1, EH3, N, D
defensive	D, Y, F, EH1, EH3, N, S, I1, V
defer	D, E1, F, ER R
deficit	D, EH1, F, I3, S, I1, T
degree	D, Y, G, R, E1, Y
delay	D, I1, L, EH3, A1, Y
delete	D, E1, L, E1, Y, T
deliver	D, Y, L, I1, V, ER
delta	D EH2, EH3, L, T, UH1
demand	D, Y, M, AE1, EH3, N, D

demonstrate	D, EH1, M, UH3, N, S, T, R, A1, Y, T
deny	D, Y, N, AH1, EH3, Y
destroy	D, Y, S, T, R, O1, UH3, I3, AY
detail	D, E, T, EH3, A1, I3, UH3, L
determine	D, Y, T, ER, M, I1, N
device	D, Y, V, UH3, AH2, Y, S
dew	(use "do" program)
diagnostic	D, AH1, AY, I3, G, N, AH1, UH3, S, T, I3, K
dial	D, AH1, EH3, I3, UH3, L
dictionary	D, I1, I3, K, SH, UH3, N, EH3, EH3, ER, Y
did	D, I1, I3, D
die	D, AH1, EH3, Y
diet	D, AH1, EH3, AY, I2, T
differ	D, I1, I3, F, ER
difference	D, I1, F, R, EH3, N, DT, S
different	D, I1, F, R, EH3, N, T
digit	D, I1, D, J, I1, T
digital	D, I1, D, J, I3, T, UH3, L
dime	D, AH1, EH3, Y, M
diode	D, AH1, EH3, AY, O1, U1, D
direct	D, ER, EH1, EH3, K, T
directory	D, ER, EH1, EH3, K, T, ER, Y
dirt	D, ER, R, T
disagree	D, I1, S, UH1, G, R, E1, Y
disappear	D, I1, S, UH1, P, AY, I3, R
disconnect	D, I1, S, K, UH1, N, EH1, EH3, K, T
discuss	D, I1, I3, S, K, UH1, UH2, S
disk	D, I1, I3, S, K
display	D, I1, I3, S, P, L, A1, I3, Y
distance	D, I1, S, T, EH3, N, T, S
divide	D, I1, V, AH1, EH3, Y, D
dividend	D, I1, V, I1, D, EH1, EH3, N, D
division	D, I1, V, I1, ZH, UH3, N
do	D, IU, U1, U1
dock	D, AH1, UH3, K
doctor	D, AH1, UH3, K, T, ER
document	D, AH1, K, Y1, UH3, M, EH3, N, T
does	D, UH2, UH1, Z
dollar	D, AH1, UH3, L, ER
done	D, UH1, UH3, N
door	D, O1, O2, R
double	D, UH3, UH1, B, UH3, L
doubt	D, UH3, AH2, U1, T
down	D, AH1, UH3, U1, N
draft	D, R, AE1, EH3, F, T
draw	D, R, AW
drill	D, R, I1, I3, L
drink	D, R, I1, I3, NG, K
drive	D, R, AH1, EH3, Y, V
drop	D, R, AH1, UH3, P
drum	D, R, UH1, UH2, M
dry	D, R, AH1, EH3, I3, Y
due	(use "do" program)
dump	D, UH1, UH2, M, P
duration	D, ER, R, A1, Y, SH, UH3, N
during	D, ER, R, I1, NG
duty	D, IU, U1, U1, T, Y
dwell	D, W, EH1, EH3, L
E	E1, Y
each	E1, AY, T, CH
ear	E1, I2, R
early	ER, R, L, Y
earn	ER, R, N
east	E1, AY, S, T
easy	E1, AY, Z, Y
echo	EH1, EH3, K, O1, U1
edge	EH1, EH3, D, J
edit	EH1, EH3, D, I2, T
educate	EH1, D, J, U1, K, A1, Y, T
effect	UH1, F, EH1, EH3, K, T
efficient	E1, F, I1, SH, EH3, N, T
effort	EH2, EH3, F, ER, T
eight	A2, A2, Y, T
eighth	A2, A2, Y, DT, DT, TH
eighty	A2, A2, Y, T, Y
either	E1, Y, THV, ER
electric	EH3, L, EH1, K, T, R, I2, K
electrician	EH3, L, EH1, K, PAO, T, R, I1, SH, UH3, N
electronic	EH3, L, EH1, K, T, R, AH1, N, I2, K
elevator	EH1, L, UH3, V, A2, AY, D, ER
eleven	EH1, L, EH1, EH3, V, I2, N
eligible	EH1, L, UH3, D, J, EH3, B, UH3, L
eliminate	EH1, L, I1, M, I1, N, A1, Y, T
else	EH1, EH3, L, S
emit	Y, M, I1, I3, T
employ	EH1, EH3, M, P, L, O1, UH3, I3, AY
empty	EH1, EH3, M, P, T, Y
enable	EH1, N, A1, Y, B, UH3, L
enclose	EH1, EH3, N, K, L, O1, U1, Z
end	EH1, EH3, N, D
engine	EH1, EH3, N, D, J, I1, N
engineer	EH1, N, D, J, I2, N, AY, I1, R
endorse	EH1, EH3, N, D, O2, O2, R, S
english	I1, NG, G, L, I2, SH
enter	EH1, EH3, N, T, ER
entry	EH1, EH3, N, T, R, Y
epsilon	EH1, P, S, UH3, L, AH1, UH3, N
equal	Y, K, W, UH3, L
equipment	E1, K, W, IL, P, M, EH3, N, T
erase	E1, R, A1, Y, S
error	EH3, EH3, EH3, R, ER
escape	EH1, EH3, S, K, A1, AY, Y, P
escrow	EH1, EH3, S, K, R, O1, U1
establish	UH1, S, T, AE1, EH3, B, L, I2, SH
estate	EH1, EH3, S, T, A1, AY, Y, T
estimate	EH1, S, T, EH3, M, I3, T
exact	EH1, EH3, G, PAO, Z, AE1, EH3, K, T

examine	EH1, EH3, G, PAO, Z, AE1, EH3, M, I1, N
exceed	EH1, EH3, K, PAO, S, E1, Y, D
except	EH1, EH3, K, PAO, S, EH1, EH3, P, T
exchange	EH1, EH3, K, PAO, S, T, CH, A1, AY, Y, N, D, J
execute	EH1, EH3, K, PAO, S, UH3, K, Y1, IU, U1, T
exempt	EH1, EH3, G, PAO, Z, EH1, EH3, M, P, T
exit	EH1, EH3, G, PAO, Z, I1, I3, T
expect	EH1, EH3, K, PAO, S, P, EH1, EH3, K, T
expedite	EH1, EH3, K, PAO, S, P, EH1, EH3, D, UH3, AH2, Y, T
expend	EH1, EH3, K, PAO, S, P, EH1, EH3, N, D
experiment	EH1, K, PAO, S, P, EH1, R, UH3, M, EH3, N, T
exponent	EH1, K, PAO, S, P, O2, O2, N, EH3, N, T
express	EH1, EH3, K, PAO, S, P, R, EH1, S
extension	EH1, EH3, K, PAO, S, T, EH1, EH3, N, SH, UH3, N
F	EH1, EH2, F
face	F, A1, AY, Y, S
facility	F, UH2, S, I1, L, I3, T, Y
fact	F, AE1, EH3, K, T
fahrenheit	F, EH1, R, I2, N, H, UH3, AH2, Y, T
fail	F, A1, AY, I3, UH3, L
fall	F, AW, L
false	F, AW, L, S
familiar	F, UH1, M, I1, L, Y1, ER
far	F, AH1, UH3, R
farad	F, EH3, EH3, ER, AE1, EH3, D
fast	F, AE1, EH3, S, T
fault	F, AW, L, T
feat	(use "feet" program)
feature	F, E1, AY, T, CH, ER
february	F, EH1, B, Y1, IU, W, EH1, R, Y
federal	F, EH1, EH3, D, R, UH3, L
fee	F, E1, Y
feed	F, E1, Y, D
feet	F, E1, Y, T
female	F, AY, Y, M, A1, AY, UH3, L
field	F, E1, AY, UH3, L, D
fifteen	F, I1, I3, F, T, E1, Y, N
fifth	F, I1, I3, F, TH
fifty	F, I1, I3, F, T, Y
file	F, AH1, EH3, I3, UH3, L
fill	F, I1, I3, L
final	F, AH1, Y, N, UH3, L
finance	F, AH1, EH3, Y, N, AE1, EH3, N, S
find	F, AH1, EH3, Y, N, D
finger	F, I1, I3, NG, G, ER
finish	F, I1, N, I1, SH
fire	F, AH1, EH3, AY, R
first	F, ER, R, S, T
fit	F, I1, I3, T
five	F, AH1, EH3, Y, V
fix	F, I1, I3, K, PAO, S
fixture	F, I1, I3, K, PAO, S, T, CH, ER
flash	F, L, AE1, EH3, SH
flat	F, L, AE1, EH3, T
flight	F, L, UH3, AH2, Y, T
flip	F, L, I1, I3, P
floor	F, L, O1, O2, R
flop	F, L, AH1, UH3, P
flow	F, L, O1, U1
fly	F, L, AH1, EH3, Y
fold	F, O2, O2, L, L, D
follow	F, AH1, AW2, L, O1, U1
food	F, U1, U1, D
foot	F, OO1, OO1, T
for	(use "four" program)
fore	(use "four" program)
force	F, O2, O2, R, S
foreman	F, O2, O2, R, M, EH2, N
forget	F, O2, O2, R, G, EH1, EH3, T
forgive	F, O2, O2, R, G, I1, I3, V
form	F, O2, O2, R, M
format	F, O2, O2, R, M, AE1, EH3, T
forty	F, O2, O2, R, T, Y
forward	F, O2, O2, R, W, ER, D
found	F, AH1, UH3, W, N, D
four	F, O1, O2, R
fourth	F, O1, O2, R, TH
fox trot	F, AH1, UH3, K, PAO, S, T, R, AH1, UH3, T
frame	F, R, A1, AY, Y, M
fraud	F, R, AW, D
free	F, R, E1, Y
french	F, R, EH1, EH3, N, T, CH
frequency	F, R, E1, K, W, EH3, N, DT, S, Y
frequent	F, R, E1, K, W, EH3, N, T
friday	F, R, AH1, EH3, Y, D, A1, I3, Y
fright	F, R, UH3, AH2, Y, T
from	F, R, UH1, UH3, M
front	F, R, UH3, UH1, N, T
fruit	F, R, IU, U1, T
fuel	F, Y1, IU, U1, UH3, L
full	F, OO1, L
function	F, UH1, UH2, N, K, SH, UH3, N
fund	F, UH1, UH2, N, D
furnace	F, ER, R, N, EH3, S
further	F, ER, R, THV, ER
future	F, Y1, IU, U1, T, CH, ER
G	D, J, E1, Y
gage	(use "gauge" program)
gain	G, A1, AY, Y, N
gait	(use "gate" program)
gallon	G, AE1, AH2, L, UH3, N
game	G, A1, AY, Y, M
gamma	G, AE1, EH3, M, UH2, UH3
gap	G, AE1, EH3, P

garage	G, UH1, R, AH1, UH3, ZH
gas	G, AE1, EH3, S
gate	G, A1, AY, Y, T
gauge	G, A1, AY, Y, D, J
general	D, J, EH1, EH3, N, ER, UH3, L
generate	D, J, EH1, N, ER, A1, Y, T
gentlemen	D, J, EH1, EH3, N, T, L, M, I2, N
german	D, J, ER, R, M, EH2, N
get	G, EH1, EH3, T
girl	G, ER, R, L
give	G, I1, I3, V
glass	G, L, AE1, EH3, S
glitch	G, L, I1, I3, T, CH
globe	G, L, O1, U1, B
go	G, OO1, O1, U1
golf	G, AW2, AW2, UH3, L, F
good	G, OO1, OO1, D
govern	G, UH1, UH3, V, ER, N
grade	G, R, A1, AY, Y, D
gram	G, R, AE1, EH3, M
grand	G, R, AE1, EH3, N, D
graph	G, R, AE1, EH3, F
grate	(use "great" program)
gray	(use "grey" program)
great	G, R, A1, Y, T
green	G, R, E1, Y, N
greet	G, R, E1, Y, T
grey	G, R, A1, AY, Y
grind	G, R, AH1, EH3, Y, N, D
grocery	G, R, O1, U1, S, ER, Y
ground	G, R, AH1, UH3, W, N, D
group	G, R, U1, U1, P
grow	G, R, O1, U1
guard	G, AH1, R, D
guarantee	G, EH1, R, I3, N, T, E1, Y
guess	G, EH1, EH3, S
H	A1, AY, Y, T, CH
had	H, AE1, EH3, D
half	H, AE1, EH3, F
halt	H, AW, L, T
hammer	H, AE1, EH3, M, ER
hand	H, AE1, EH3, N, D
handle	H, AE1, EH3, N, D, UH3, L
hang	H, AE1, I3, NG
happy	H, AE1, EH3, P, Y
hard	H, AH1, R, D
has	H, AE1, EH3, Z
have	H, AE1, EH3, V
he	H, E1, Y
head	H, EH1, EH3, D
hear	H, AY, I3, R
heart	H, AH1, UH3, R, T
heat	H, E1, AY, T
heavy	H, EH1, V, Y
height	H, UH3, AH2, Y, T
held	H, EH1, UH3, L, D
hello	H, EH1, UH3, L, UH3, O1, U1
help	H, EH1, EH3, L, P
henry	H, EH1, EH3, N, R, Y
her	H, ER
here	(use "hear" program)
hertz	H, R, R, T, S
hex	H, EH1, EH3, K, PAO, S
high	H, AH1, EH3, Y
his	H, J1, I3, Z
hold	H, O2, O2, L, L, D
hole	H, O1, U1, L
home	H, O1, U1, M
hook	H, OO1, OO1, K
host	H, O1, U1, S, T
hot	H, AH1, UH3, T
hotel	H, O1, U1, T, EH2, EH2, L
hour	AH1, UH3, W, ER
house	H, UH3, AH2, U1, S
how	H, AH1, O2, U1
human	H, Y1, IU, U1, U1, M, EH2, N
hundred	H, UH1, UH2, N, D, R, I3, D
hungry	H, UH1, UH2, NG, G, R, Y
I	AH1, EH3, I3, Y
idle	AH1, Y, D, UH3, L
idol	(use "idle" program)
if	I1, I3, F
immediate	I1, I3, M, E1, D, Y, EH3, T
important	I1, I3, M, P, O2, O2, R, T, EH3, N, T
improper	I1, I3, M, P, R, AH1, UH3, P, ER
improve	I1, I3, M, P, R, IU, U1, U1, V
in	I1, I3, N
inch	I1, I3, N, T, CH
include	I1, I3, N, K, L, IU, U1, U1, D
income	I1, I3, N, K, UH1, UH3, M
indep endent	I1, N, D, E1, P, EH2, EH3, N, D, EH3, N, T
index	I1, I3, N, D, EH1, EH3, K, PAO, S
india	I2, I3, N, D, Y, UH2
indicate	I1, N, D, I3, K, A1, Y, T
industrial	I1, I3, N, D, UH1, UH2, S, T, R, AY, UH3, L
inform	I1, I3, N, F, O2, O2, R, M
initial	I1, I3, N, I1, SH, UH3, L
inn	(use "in" program)
input	I1, I3, N, P, OO1, OO1, T
inquire	I1, I3, N, K, W, AH1, EH3, AY, R
insert	I1, N, S, R, R, T
inspect	I1, I3, N, S, P, EH1, EH3, K, T
install	I1, I3, N, S, T, AW, L
instead	I1, I3, N, S, T, EH1, EH3, D
instruct	I1, I3, N, S, T, R, UH1, UH2, K, T
instrument	I1, I3, N, S, T, R, UH1, M, EH1, EH3, N, T
insufficient	I1, N, S, UH2, F, I1, SH, EH3, N, T
insurance	I1, I3, N, SH, ER, R, EH3, N, T, S
interest	I1, N, T, R, EH1, S, T
interface	I1, I3, N, T, ER, F, A1, AY, Y, S
interpret	I1, I3, N, T, ER, P, R, EH3, T
interupt	I1, N, T, ER, UH3, UH1, P, T
intrude	I1, I3, N, T, R, IU, U1, U1, D
invalid	I1, I3, N, V, AE1, AW2, L, I1, D

invent	I1, I3, N, V, EH1, EH3, N, T
inventory	I1, N, V, EH1, N, T, O1, R, Y
invest	I1, I3, N, V, EH1, EH3, S, T
invoice	I1, I3, N, V, O1, UH3, I3, AY, S
irregular	I1, R, EH1, G, Y1, UH3, L, ER
is	I1, I3, Z
it	I1, I3, T
item	AH2, UH3, Y, D, UH3, M
J	D, J, EH3, A1, AY, Y
jack	D, J, AE1, EH3, K
janusry	D, J, AE1, EH3, N, Y1, UI, EH3, EH3, ER, Y
job	D, J, AH1, UH3, B
join	D, J, O1, UH3, I3, AY, N
jolt	D, J, O2, O2, L, T
joy	D, J, O1, UH3, I3, AY
judge	D, J, UH1, UH2, D, J
juliet	D, J, IU, U1, L, Y, EH2, EH3, T
july	D, J, UH1, L, AH1, EH3, Y
jump	D, J, UH1, UH2, M, P
june	D, J, IU, U1, U1, N
K	K, EH3, A1, AY, Y
keep	K, E1, Y, P
kev	K, E1, Y
keyboard	K, AY, Y, B, O1, O2, R, D
kill	K, I1, I3, L
kilo	K, E1, AY, L, UH3, O2, U1
knew	(use "new" program)
knot	(use "not" program)
know	(use "no" program)
knowledge	N, AH1, UH3, L, I3, D, J
L	EH1, EH3, UH3, L
lab	L, AE1, EH3, B
labor	L, A1, Y, B, ER
language	L, AE1, EH3, NG, G, W, I1, D, J
lapse	L, AE1, EH3, P, S
large	L, AH1, R, D, J
last	L, AE1, EH3, S, T
late	L, A1, AY, Y, T
law	L, AW
lead	L, E1, Y, D
led	L, EH1, EH3, D
left	L, EH1, EH3, F, T
leg	L, EH1, EH3, G
legal	L, E1, G, UH3, L
lend	L, EH1, EH3, N, D
length	L, EH1, EH3, NG, TH
less	L, EH1, EH3, S
let	L, EH1, EH3, T
letter	L, EH1, EH3, T, ER
level	L, EH1, EH3, V, UH3, L
life	L, UH3, AH2, Y, F
light	L, UH3, AH2, Y, T
like	L, UH3, AH2, Y, K
lima	L, AY, Y, M, UH1
limit	L, I1, M I1, T
line	L, AH1, EH3, Y, N
linear	L, I2, I3, N, AY, Y, ER
link	L, I1, I3, NG, K
lip	L, I1, I3, P
liquid	L, I1, K, W, I1, D
list	L, I1, I3, S, T
listen	L, I1, I3, S, I2, N
little	L, I1, I3, T, UH3, L
load	L, UH3, O1, U1, D
loan	L, UH3, O1, U1, N
local	L, O2, O2, K, UH3, L
lock	L, AH1, UH3, K
log	L, AW, G
long	L, AW, NG
look	L, OO1, OO1, K
loss	L, AW, S
lost	L, AW, S, T
lot	L, AH1, UH3, T
low	L, O1, U1
M	EH1, EH2, M
machine	M, UH2, SH, E1, Y, N
mail	(use "male" program)
maintenance	M, A1, Y, N, T, EH2, N, EH3, N, DT, S
make	M, A1, AY, Y, K
male	M, A2, A2, AY UH3, L
man	M, AE1, EH3, N
manage	M, AE1, EH3, N, I1, D, J
manual	M, AE1, EH3, N, Y1, U1, UH3, L
manufacture	M, AE1, EH3, N, Y1, U1, F, AE1, EH3, K, T, CH, ER
many	M, EH2, EH2, N, Y
map	M, AE1, EH3, P
march	M, AH1, R, T, CH
margin	M, AH1, UH3, R, D, J, I2, N
mark	M, AH1, R, K
market	M, AH1, R, K, EH3, T
match	M, AE1, EH3, T, CH
mature	M, UH1, T, CH, IU, ER
maximum	M, AE1, EH3, K, PAO, S, EH3, M, UH2, M
may	M, A1, I3, Y
me	M, E1, Y
measure	M, EH3, EH1, ZH, ER
meat	M, E1, AY, T
mechanical	M, UH1, K, AE1, EH3, N, I3, K, UH3, L
media	M, E1, AY, D, Y, UH1
medicine	M, EH2, EH3, D, I3, S, I1, N
medium	M, E1, D, AY, UH1, M
meet	(use "meat" program)
mega	M, EH1, EH3, G, UH2, UH3
member	M, EH1, EH3, M, B, ER
memory	M, EH1, EH3, M, ER, Y
men	M, EH1, EH3, N
merchandise	M, ER, T, CH, EH3, N, D, AH1, EH3, Y, Z
merge	M, ER, R, D, J
message	M, EH1, EH3, S, I2, D, J
metal	M, EH1, EH3, T, UH3, L

meter	M, E1, Y, T, ER
micro	M, UH3, AH2, AY, K, R, 01, U1
middle	M, I1, I3, D, UH3, L
mike	M, UH3, AH2, Y, K
mile	M, AH1, EH3, I3, UH3, L
mill	M, I1, I3, L
milli	M, I1, I3, L, UH3
million	M, I1, I3, L, Y, UH3, N
mini	M, I2, I2, N, Y
minus	M, AH1, Y, N, EH3, S
minute	M, I1, N, EH3, T
miscellaneous	M, I1, S, UH3, L, A1, AY, N, Y, UH3, S
miss	M, I1, I3, S
mistake	M, I1, I3, S, T, A1, AY, Y, K
mode	M, 01, U1, D
model	M, AH1, UH3, D, UH3, L
module	M, AH1, UH3, D, J, IU, U1, UH3, L
monday	M, UH3, UH1, N, D, A1, I3, Y
money	M, UH3, UH1, N, AY, Y
month	M, UH3, UH1, N, TH
more	M, 02, 02, R
morning	M, 02, 02, R, N, I1, I3, NG
most	M, 01, U1, S, T
motor	M, 01, U1, T, ER
mount	M, AH1, UH3, W, N, T
move	M, U1, U1, V
Mr.	M, I1, S, T, ER
Mrs.	M, I1, S, I2, Z
Ms.	M, I1, I3, Z
much	M, UH1, UH2, T, CH
multi	M, UH2, UH3, L, T, Y
multiple	M, UH1, L, T, EH3, P, UH3, L
multiply	M, UH1, L, T, I3, P, L, AH1, Y
N	EH1, EH2, N
name	N, A1, AY, Y, M
nano	N, AE1, EH3, N, 01, U1
national	N, AE1, EH3, SH, UH3, N, UH3, L
native	N, A1, Y, T, I1, V
near	N, AY, I1, R
neat	N, E1, AY, T
neck	N, EH1, EH3, K
need	N, E1, Y, D
negative	N, EH1, G, EH3, T, I1, V
net	N, EH1, EH3, T
neutral	N, IU, U1, T, R, UH2, L
new	N, IU, U1, U1
next	N, EH1, EH3, K, PAO, S, T
nice	N, UH3, AH2, Y, S
nickel	N, I1, I3, K, UH3, L
night	N, UH3, AH2, Y, T
nine	N, AH1, EH3, Y, N
ninety	N, AH1, EH3, Y, N, T, Y
nineth	N, AH1, Y, N, DT, TH
no	N, 001, 01, U1
noise	N, 01, UH3, I3, AY, Z
none	N, UH1, UH3, N
noon	N, IU, U1, U1, N
normal	N, 02, 02, R, M, UH3, L
north	N, 02, 02, R, TH
not	N, AH1, UH3, T
note	N, 01, U1, T
nothing	N, UH1, TH, I1, I3, NG
notice	N, 01, U1, T, I1, S
notify	N, 01, U1, T, I1, F, AH1, EH3, Y
november	N, 01, U1, V, EH1, EH3, M, B, ER
now	N, AH1, UH3, U1
number	N, UH1, UH2, M, B, ER
nurse	N, ER, R, S
nut	N, UH1, UH2, T
O	02, 01, U1
oar	(use "or" program)
object	UH1, B, D, J, EH1, EH3, K, T
object-2	AH1, UH3, B, D, J, EH2, EH2, K, T
obligation	AH1, B, L, I3, G, A1, Y, SH, UH3, N
obsolete	AH1, UH3, B, S, UH3, L, AY, Y, T
october	AH1, UH3, K, T, O1, U1, B, ER
odd	AH1, UH3, D
of	UH1, UH3, V
off	AW, F
office	AW, F, I1, S
official	UH1, F, I1, SH, UH3, L
often	AW2, AW2, F, I3, N
ohm	O2, O2, U1, M
oil	O1, EH3, I3, UH3, L
old	O2, O2, L, L, D
omega	O1, U1, M, A1, Y, G, UH2
omit	O1, U1, M, I1, I3, T
on	AH1, UH3, N
once	W, UH1, N, T, S
one	W, UH1, UH2, N
only	O1, O2, N, L, Y
open	O1, P, I2, N
operable	AH1, UH3, P, ER, UH3, B, UH3, L
operate	AH1, UH3, P, ER, A1, Y, T
operator	AH1, UH3, P, ER, A1, Y, T, ER
option	AH1, UH3, P, SH, UH3, N
or	O2, O2, R
orange	O2, O2, R, I1, N, D, J
order	O2, O2, R, D, ER
ore	(use "or" program)
original	O2, R, I2, I3, D, J, I3, N, UH3, L
oscar	AH1, UH3, S, K, ER
other	UH1, UH3, THV, ER
ounce	AH1, UH3, W, N, S
out	UH3, AH2, U1, T
oven	UH1, V, I2, N
over	O1, O2, V, ER
oxygen	AH1, UH3, K, PAO, S, I3, D, J, I2, N
own	O1, U1, N
P	P, E1, Y
pack	P, AE1, EH3, K

package	P, AE1, EH3, K, I1, D, J
paid	P, A1, AY, Y, D
pain	P, A1, AY, Y, N
pane	(use "pain" program)
panel	P, AE1, EH3, N, UH3, L
papa	P, AH1, UH3, P, UH3, UH3
paper	P, A1, Y, P, ER
parcel	P, AH1, R, S, UH3, L
paren	P, EH3, EH3, ER, I2, N
part	P, AH1, R, T
partial	P, AH1, R, SH, UH2, L
pass	P, AE1, EH3, S
passed	(use "past" program)
past	P, AE1, EH3, S, T
pat	P, AE1, EH3, T
pattern	P, AE1, EH3, T, ER, N
pause	P, AW, Z
pay	P, A2, A2, AY, Y
pea	(use "P" program)
peace	(use "piece" program)
peak	P, E1, AY, K
peek	(use "peak" program)
percent	P, ER, S, EH1, EH3, N, T
period	P, I1, R, Y, UH2, D
permanent	P, ER, M, EH2, N, EH1, N, T
person	P, ER, S, UH1, N
personal	P, ER, S, UH3, N, UH2, L
personality	P, ER, S, UH3, N, AE1, UH3, L, I3, T, Y
phase	F, A1, AY, Y, Z
phone	F, O1, U1, N
pick	P, I1, I3, K
pico	P, E1, Y, K, O2, U1
piece	P, E1, Y, S
pint	P, AH1, Y, N, T
pipe	P, UH3, AH2, Y, P
place	P, L, A1, AY, Y, S
plain	(use "plane" program)
plan	P, L, AE1, EH3, N
plane	P, L, A1, AY, Y, N
plant	P, L, AE1, EH3, N, T
play	P, L, A1, I3, Y
please	P, L, E1, Y, Z
plot	P, L, AH1, UH3, T
plus	P, L, UH1, UH2, S
pocket	P, AH1, UH3, K, EH3, T
point	P, O1, UH3, I3, AY, N, T
poke	P, O1, U1, K
police	P, UH1, L, AY, Y, S
plain	(use "plane" program)
plan	P, L, AE1, EH3, N
plane	P, L, A1, AY, Y, N
plant	P, L, AE1, EH3, N, T
play	P, L, A1, I3, Y
please	P, L, E1, Y, Z
plot	P, L, AH1, UH3, T
plus	P, L, UH1, UH2, S
pocket	P, AH1, UH3, K, EH3, T
point	P, O1, UH3, I3, AY, N, T
poke	P, O1, U1, K
police	P, UH1, L, AY, Y, S
policy	P, AH1, UH3, L, I3, S, Y
poor	(use "pour" program)
pop	P, AH1, UH3, P
port	P, 02, 02, R, T
position	P, UH1, Z, I1, SH, UH3, N
positive	P, AH1, UH3, Z, I1, T, I1, V
possible	P, AH1, UH3, S, UH3, B, UH2, L
post	P, 01, U1, S, T
potential	P, 01, T, EH1, EH3, N, T, CH, UH3, L
pound	P, AH1, UH3, W, N, D
pour	P, 01, 02, R
power	P, AH1, UH3, W, ER
practice	P, R, AE1, EH3, K, T, I1, S
premium	P, R, AY, Y, M, Y, UH1, M
prepare	P, R, E1, P, EH1, EH3, R
press	P, R, EH1, EH3, S
pressure	P, R, EH1, SH, ER
prevent	P, R, Y, V, EH1, EH3, N, T
previous	P, R, Y, V, Y, UH1, S
price	P, R, UH3, AH2, Y, S
principal	(use "principle" program)
principle	P, R, I1, N, DT, S, UH3, P, UH3, L
print	P, R, I1, I3, N, T
prior	P, R, AH1, Y, ER
priority	P, R, AH1, Y, 01, R, I3, DT, Y
private	P, R, AH1, EH3, Y, V, I3, T
probe	P, R, 01, U1, B
problem	P, R, AH1, UH3, B, L, UH3, M
procedure	P, R, UH1, S, E1, D, J, ER
proceed	P, R, 01, S, E1, Y, D
process	P, R, AH1, UH3, S, EH1, EH3, S
produce	P, R, UH1, D, IU, U1, U1, S
product	P, R, AH1, UH3, D, UH1, UH2, K, T
progress	P, R, AH1, UH3, G, R, EH1, S
profession	P, R, UH1, F, EH1, EH3, SH, UH3, N
profit	P, R, AH1, UH3, F, I1, T
program	P, R, 01, G, R, AE1, EH3, M
project	P, R, AH1, UH3, D, J, EH2, EH2, K, T
PROM	P, R, AH1, UH3, M
promote	P, R, UH1, M, 01, U1, T
propose	P, R, UH1, P, 01, U1, Z
protect	P, R, UH1, T, EH1, EH3, K, T
public	P, UH1, UH3, B, L, I3, K
pull	P, 001, 001, L
pulse	P, UH1, UH2, L, S
punch	P, UH1, UH2, N, T, CH
purpose	P, R, R, P, EH2, S
purchase	P, R, R, DT, CH, I2, S
pure	P, Y1, IU, ER
push	P, 001, IU, SH
put	P, 001, 001, T
Q	K, Y1, IU, U1, U1
qualify	K, W, AW1, L, I1, F, AH1, EH3, Y

quantity	K, W, AH1, N, T, I3, T, Y
quart	K, W, 01, R, T
quarter	K, W, 01, R, T, ER
quebec	K, W, I1, B, EH1, EH3, K
question	K, W, EH1, EH3, S, T, CH, UH3, N
quick	K, W, I1, I3, K
quiet	K, W, AH1, EH3, AY, I2, T
quit	K, W, I1, I3, T
quiz	K, W, I1, I3, Z
quota	K, W, 01, 02, T, UH1
quote	K, W, 01, U1, T
R	AH1, UH2, ER
rail	R, A1, AY, I3, UH3, L
rain	R, A1, AY, Y, N
raise	R, A1, AY, Y, Z
range	R, A1, AY, Y, N, D, J
radio	R, A1, Y, D, Y, 01, U1
rate	R, A1, AY, Y, T
ratio	R, A1, Y, SH, Y, 01, U1
reach	R, E1, Y, T, CH
read	R, E1, Y, D
ready	R, EH1, EH3, D, Y
real	R, E1, AY, L
reason	R, E1, Y, Z, UH1, N
rebate	R, E1, B, A1, Y, T
recall	R, E1, K, AW2, AW1, L
receipt	R, E1, S, AY, Y, T
receive	R, E1, S, E1, Y, V
record	R, E1, K, 02, 02, R, D
record-2	R, EH1, EH3, K, ER, D
red	R, EH1, EH3, D
reel	(use "real" program)
refer	R, E1, F, UH1, UH2, N, D
refuse	R, E1, F, Y1, IU, U1 U1, Z
register	R, EH1, D, J, I1, S, T, ER
regular	R, EH1, G, Y1, IU, L, ER
rein	(use "rain" program)
reject	R, E1, D, J, EH1, EH3, K, T
relay	R, E1, L, A1, I3, Y
release	R, E1, L, E1, AY, S
remain	R, E1, M, A1, AY, Y, N
remove	R, E1, M, U1, U1, V
repair	R, E1, P, EH2, EH2, R
repeat	R, E1, P, E1, AY, T
replace	R, E1, P, L, A1, AY, Y, S
report	R, E1, P, 02, 02, R, T
represent	R, EH1, P, R, I2, Z, EH1, EH3, N, T
request	R, E1, K, W, EH1, EH3, S, T
require	R, E1, K, W, AH1, EH3, AY, R
requisition	R, EH1, K, W, I2, Z, I1, SH, UH3, N
rescue	R, EH1, EH3, S, K, Y1, IU, U1
resemble	R, E1, Z, EH1, EH3, M, B, UH3, L
reset	R, E1, S, EH1, EH3, T
resistor	R, E1, Z, I1, S, T, ER
respect	R, E1, S, P, EH1, EH3, K, T
respond	R, E1, S, P, AH1, UH3, N, D
responsible	R, I2, S, P, AH1, UH3, N, DT, S, UH3, B, UH3, L
rest	R, EH1, EH3, S, T
restrict	R, E1, S, T, R, I1, I3, K, T
result	R, E1, Z, UH1, UH2, L, T
resume	R, E1, Z, IU, U1, U1, M
retail	R, AY, E1, T, EH3, A1, I3, UH3, L
retain	R, E1, T, A1, AY, Y, N
return	R, E1, T, ER, R, N
revision	R, E1, V, I1, ZH, UH3, N
revolve	R, E1, V, AH1, UH3, L, V
ribbon	R, I2, I3, B, UH3, N
right	R, UH3, AH2, Y, T
romeo	R, 01, U1, M, Y, 01, U1
room	R, U1, U1, M
root	R, U1, U1, T
round	R, AH1, UH3, W, N, D
route	R, UH2, AH2, U1, T
row	R, 01, U1
run	R, UH1, UH3, N
rush	R, UH1, UH2, SH
S	EH1, EH2, S
safe	S, A1, AY, Y, F
sail	(use "sale" program)
salary	S, AE1, AH2, L, UH3, R, Y
sale	S, A1, A2, AY, UH3, L
same	S, A1, AY, Y, M
saturday	S, AE1, EH3, T, ER, D, A1, Y
save	S, A1, AY, Y, V
say	S, A1, I3, Y
scan	S, K, AE1, EH3, N
scent	(use "cent" program)
schedule	S, K, EH1, EH3, D, J, IU, U1, L
school	S, K, U1, U1, L
science	S, AH1, I3, Y, EH3, N, DT, S
score	S, K, O2, O2, R
scrap	S, K, R, AE1, EH3, P
screw	S, K, R, IU, U1, U1
sea	(use "C" program)
seat	S, E1, AY, T
second	S, EH1, EH3, K, UH1, N, D
secret	S, E1, K, R, I3, T
section	S, EH1, EH3, K, SH, UH3, N
security	S, EH1, EH3, K, Y, ER, I1, T, Y
see	(use "C" program)
seize	S, E1, Y, Z
select	S, UH1, L, EH1, EH2, K, T
sell	S, EH1, EH3, L
semi	S, EH1, M, AH1, Y
semicolon	S, EH1, M, AH1, Y, K, OO1, O1, L, I2, N
send	S, EH1, EH3, N, D
sent	(use "cent" program)
sentence	S, EH1, N, T, I2, N, DT, S
separate	S, EH1, EH3, P, UH1, R, A1, AY, T
separate-2	S, EH1, EH3, P, R, I2, T

september	S, EH1, EH3, P, T, EH1, EH3, M, B, ER
sequence	S, E1, K, W, EH1, EH3, N, S
serial	S, I1, R, Y, UH3, L
series	S, I1, R, Y, Z
service	S, ER, V, I1, S
set	S, EH1, EH3, T
seven	S, EH1, EH3, V, I2, N
seventh	S, EH1, EH3, V, I2, N, DT, TH
seventy	S, EH1, V, I2, N, D, Y
several	S, EH1, V, ER, UH3, L
sew	(use "so" program)
share	SH, EH3, EH3, ER
sharp	SH, AH1, R, P
shift	SH, I1, I3, F, T
ship	SH, I1, I3, P
shop	SH, AH1, UH3, P
short	SH, O2, O2, R, T
should	SH, IU, IU, IU, D
shunt	SH, UH1, UH2, N, T
shut	SH, UH1, UH2, T
side	S, AH1, EH3, Y, D
sierra	S, E1, I3, EH1, R, UH1
signal	S, I1, I3, G, N, UH3, L
silver	S, I1, I3, L, V, ER
single	S, I1, I3, NG, G, UH3, L
six	S, I1, I3, K, PAO, S
sixth	S, I1, I3, K, PAO, S, TH
sixty	S, I1, I3, K, PAO, T, Y
size	S, AH1, EH3, Y, Z
skin	S, K, I1, I3, N
sky	S, K, AH1, EH3, I3, Y
slang	S, L, AE1, EH3, NG
slash	S, L, AE1, EH3, SH
slave	S, L, A1, AY, Y, V
slip	S, L, I1, I3, P
slow	S, L, O1, U1
small	S, M, AW, L
smell	S, M, EH1, EH3, L
smile	S, M, AH1, EH3, I3, UH3, L
smoke	S, M, O1, U1, K
snow	S, N, OO1, O2, U1
so	S, OO1, O2, U1
soft	S, AW, F, T
sold	S, O2, O2, L, L, D
solid	S, AH1, UH3, L, I1, D
son	(use "sun" program)
some	(use "sum" program)
sorry	S, AW, R, Y
sort	S, O2, O2, R, T
sound	S, AH1, UH3, W, N, D
source	S, O1, O2, R, S
south	S, AH1, UH3, U1, TH
space	S, P, A1, AY, Y, S
spark	S, P, AH1, R, K
speak	S, P, E1, AY, K
special	S, P, EH1, EH3, SH, UH3, L
speed	S, P, E1, Y, D
speech	S, P, E1, Y, T, CH
spell	S, P, EH1, EH3, L
spend	S, P, EH1, EH3, N, D
split	S, P, L, I1, I3, T
spoon	S, P, U1, U1, N
spring	S, P, R, I1, I3, NG
square	S, K, W, EH1, R
stack	S, T, AE1, EH3, K
stair	(use "stare" program)
stand	S, T, AE1, EH3, N, D
standard	S, T, AE1, EH3, N, D, ER, D
star	S, T, AH1, UH3, R
stare	S, T, EH3, EH3, ER
start	S, T, AH1, R, T
state	S, T, A1, AY, Y, T
station	S, T, A1, Y, SH, UH3, N
status	S, T, AE1, EH3, T, I2, S
steal	(use "steel" program)
steel	S, T, E1, Y, L
step	S, T, EH1, EH3, P
stick	S, T, I1, I3, K
stock	S, T, AH1, UH3, K
stop	S, T, AH1, UH3, P
store	S, T, O2, O2, R
strait	(use "straight" program)
straight	S, T, R, A1, AY, Y, T
street	S, T, R, E1, Y, T
stress	S, T, R, EH1, EH3, S
string	S, T, R, I1, I3, NG
structure	S, T, R, UH1, K, T, CH, ER
style	S, T, AH1, EH3, AY, UH3, L
subject	S, UH1, UH2, B, D, J, EH1, EH3, K, T
substitute	S, UH1, UH3, B, S, T, I3, T, IU, U1, T
subtract	S, UH1, UH2, B, T, R, AE1, EH3, K, T
sufficient	S, UH1, F, I1, SH, EH3, N, T
suggest	S, UH1, UH2, G, D, J, EH1, EH3, S, T
suit	S, IU, U1, T
suite	S, W, AY, Y, T
sum	S, UH1, UH2, M
summary	S, UH2, UH2, M, ER, Y
summer	S, UH1, UH2, M, ER
sun	S, UH1, UH2, N
sunday	S, UH1, UH2, N, D, A1, I3, Y
super	S, IU, U1, P, ER
supply	S, UH2, P, L, AH1, Y
surface	S, ER, F, I2, S
surge	S, ER, R, D, J
surgery	S, ER, D, J, ER, Y
surgical	S, ER, D, J, UH3, K, UH3, L
surplus	S, ER, P, L, UH1, S
suspend	S, UH1, S, P, EH1, EH3, N, D
sweep	S, W, E1, Y, P
sweet	(use "suite" program)
switch	S, W, I1, I3, T, CH
syntax	S, I1, N, T, AE1, EH3, K, PAO, S
system	S, I1, S, T, UH3, M

T	T, E1, AY, Y
table	T, A1, Y, B, UH3, L
tail	(use "tale" program)
tale	T, A1, Y, UH3, L
talk	T, AW, K
tangent	T, AE1, EH3, N, D, J, EH3, N, T
target	T, AH1, UH3, R, G, I2, T
tea	(use "T" program
team	T, E1, Y, M
technical	T, EH1, EH3, K, N, I3, K, UH3, L
tee	(use "T" program)
temperature	T, EH1, EH3, M, P, ER, UH1, T, CH, ER
ten	T, EH1, EH3, N
terminal	T, ER, M, EH3, N, UH2, L
test	T, EH1, EH3, S, T
than	THV, EH1, EH3, N
the	THV, UH1, UH3
then	(use "than" program)
theory	TH, AY, I2, R, Y
thin	TH, I1, I3, N
thing	TH, I1, I3, NG
think	TH, I1, I3, NG, K
third	TH, ER, R, D
thirteen	TH, ER, T, T, E1, Y, N
thirty	TH, ER, R, D, Y
thousand	TH, AH1, UH3, U1, Z, EH3, N, D
three	TH, R, E1, Y
threw	(use "through" program)
through	TH, R, IU, U1
thursday	TH, ER, R, Z, D, A1, I3, Y
ticket	T, I1, I3, K, EH3, T
till	T, I1, I3, L
time	T, AH1 EH3, Y, M
tire	T, AH1, EH3, AY, R
title	T, UH3, AH2, Y, T, UH3, L
to	(use "two" program)
today	T, U1, D, A1, I3, Y
toilet	T, O1, EH3, I3, L, I3, T
toll	T, O2, O2, OO1, L
tomorrow	T, U1, M, AH1, R, O1, U1
ton	T, UH1, UH2, N, N
tone	T, O1, U1, N
too	(use "two" program)
tool	T, U1, U1, L
total	T, O1, U1, T, UH3, L
touch	T, UH1, UH3, T, CH
towel	T, AH1, W, UH3, L
trace	T, R, A1, AY, Y, S
trade	T, R, A1, AY, Y, D
train	T, R, A1, AY, Y, N
transact	T, R, AE1, EH3, N, S, AE1, EH3, K, T
transfer	T, R, AE1, EH3, N, S, F, ER
transistor	T, R, AE1, N, Z, I1, S, T, ER
transmit	T, R, AE1, EH3, N, Z, M, I1, I3, T
transport	T, R, AE1, EH3, N, S, P, O2, O2, R, T
transportation	T, R, AE1, N, S, P, ER, T, A1, AY, SH, UH3, N
travel	T, R, AE1, EH3, V, UH3, L
triangle	T, R, AH1, I3, AE1, EH3, NG, G, UH3, L
trouble	T, R, UH3, UH1, B, UH3, L
truck	T, R, UH1, UH2, K
true	T, R, IU, U1, U1
trust	T, R, UH1, UH2, S, T
try	T, R, AH1, EH3, I3, Y
tuesday	T, IU, U1, U1, Z, D, A1, Y
tune	T, IU, U1, U1, N
turn	T, ER, R, N
twelve	T, W, EH1, EH3, UH3, L, V
twenty	T, W, EH1, EH3, N, T, Y
two	T, IU, U1, U1
type	T, UH3, AH2, Y, P
U	Y1, IU, U1, U1
ultra	UH3, UH2, L, T, R, UH1
under	UH2, UH2, N, D, ER
uniform	Y1, IU, U1, N, I3, F, O1, R, M
until	UH2, UH2, N, T, I1, I3, L
up	UH1, UH2, P
urgent	R, R, D, J, I3, N, T
us	UH1, UH2, S
use	Y1, IU, U1, U1, Z
use-2	Y1, IU, U1, S
V	V, E1, AY, Y
vacant	V, A1, Y, K, EH3, N, T
valid	V, AE1, UH3, L, I1, D
vary	(use "very" program)
value	V, AE1, EH3, L, Y1, IU, U1
vendor	V, EH1, EH3, N, D, ER
vent	V, EH1, EH3, N, T
verify	V, EH1, R, I3, F, AH1, EH3, Y
very	V, EH1, R, Y
via	V, E1, AY, UH2, UH3
victor	V, I1, I3, K, T, ER
voice	V, O1, UH3, I3, AY, S
void	V, O1, UH3, I3, AY, D
volt	V, O2, O2, L, T
volume	V, AH1, UH3, L, Y1, IU, U1, M
W	D, UH1, B, UH3, L, Y1, IU, U1
wage	W, A1, AY, Y, D, J
wait	W, A1, AY, Y, T
want	W, AH1, UH3, N, T
was	W, UH1, UH3, Z
wash	W, AW, SH
water	W, AH1, UH3, T, ER
watt	W, AH1, UH3, T
wave	W, A1, AY, Y, V
way	(use "weigh" program)
we	W, E1, Y
weak	(use "week" program)
weapon	W, EH2, EH2, P, UH1, N
wear	(use "where" program)
wednesday	W, EH1, N, Z, D, A1, I3, Y
week	W, E1, Y, K
weigh	W, A2, A2, Y

weight	(use "wait" program)
went	W, EH1, EH3, N, T
west	W, EH1, EH3, S, T
wet	W, EH1, EH3, T
what	W, UH3, UH1, T
wheel	W, E1, Y, L
when	W, EH1, EH3, N
where	W, EH3, A2, EH3, R
which	W, I1, I3, T, CH
while	W, AH1, EH3, I1, UH3, L
whiskey	W, I1, I3, S, K, AY, Y
white	W, UH3, AH2, Y, T
who	H, IU, U1, U1
whole	(use "hole" program)
why	(use "Y" program)
will	W, I1, I3, L
window	W, I1, N, D, O1, U1
winter	W, I1, I3, N, T, ER
wire	W, AH1, EH3, AY, R
with	W, I1, I3, TH
withdraw	W, I1, I3, TH, D, R, AW
without	W, I1, I3, TH, UH2, AH2, U1, T
won	(use "one" program)
word	W, ER, R, D
work	W, ER, R, K
write	(use "right" program)
wrong	R, AW, NG
X	EH1, EH2, K, PAO, S
x-ray	EH1, EH2, K, PAO, S, R, A1, I3, Y
Y	W, AH1, EH3, I3, Y
yankee	Y1, AE1, EH3, NG, K, E1, Y
yard	Y1, AH1, R, D
year	Y1, AY, I3, R
yellow	Y1, EH1, EH3, L, O1, U1
yes	Y1, EH3, EH1, S
yesterday	Y1, EH3, EH1, S, T, ER, D, A1, I3, Y
yet	Y1, EH1, EH3, T
you	(use "U" program)
your	Y, O2, O2, R
you're	(use "your" program)
Z	Z, E1, Y
zap	Z, AE1, EH3, P
zero	Z, AY, I1, R, O1, U1
zone	Z, O1, U1, N
zulu	Z, IU, U1, L, IU, U1

PREFIXES

Prefixes	
con...	K, UH1, N
dis...	D, I1, S
en...	EH1, N
in...	I1, N
non...	N, AH1, UH3, N
pre...	P, R, E1
re...	R, E1
un...	UH1, N

Suffixes	
...d	D
...ed	I2, D
...er	ER
...es	I2, Z
...ful	F, UH3, L
...ing	I2, NG
...less	L, EH2, S
...ly	L, Y
...ment	M, EH3, N, T
...ness	N, EH3, S
...s	S
...t (...ed)	T
...tion (...sion)	SH, UH3, N
...teen	T, E1, Y, N
...ward	W, ER, D
...y	Y
...z (...es)	Z

Reprinted with permission from . . . . . .